Contents

[\* Main Program to run All TC Models (Live version)\*/ 4](#_Toc7770304)

[\* AE\_Main\_Run\_all\_steps\_local.sas ; 8](#_Toc7770305)

[\* APC\_OPROC\_Model\_Main\_Flat\_Files\_1.sas ; 10](#_Toc7770306)

[\* APC\_OPROC\_Model\_Main\_Flat\_Files\_2.sas ; 14](#_Toc7770307)

[\* BPT\_Genaral\_Settings.sas ; 16](#_Toc7770308)

[\* MPP\_Main.sas ; 19](#_Toc7770309)

[\* OPATT\_Model\_Main\_20170512.sas ; 22](#_Toc7770310)

[\* OPROC\_Model\_Main\_20170705.sas ; 25](#_Toc7770311)

[\* OM\_Main\_Run\_all\_steps\_local.sas ; 26](#_Toc7770312)

[\* Unbundled\_Model\_Main\_20170901.sas ; 27](#_Toc7770313)

[\* BPT\_00\_Group 1\_Version 1.sas ; 29](#_Toc7770314)

[\* BPT\_00\_Group 2\_Version 1.sas ; 31](#_Toc7770315)

[\* BPT\_01\_Stroke.sas ; 32](#_Toc7770316)

[\* BPT\_03\_Daycase.sas ; 38](#_Toc7770317)

[\* BPT\_04\_Endoscopy.sas ; 39](#_Toc7770318)

[\* BPT\_05\_FHF.sas ; 40](#_Toc7770319)

[\* BPT\_06\_Outpatients.sas ; 40](#_Toc7770320)

[\* BPT\_07\_Paed\_Ep.sas ; 41](#_Toc7770321)

[\* BPT\_08\_Pleural\_Effusion.sas ; 43](#_Toc7770322)

[\* BPT\_09\_SameDayEmergencyCare.sas ; 46](#_Toc7770323)

[\* BPT\_10\_TIA.sas ; 46](#_Toc7770324)

[\* BPT\_11\_EarlyInflammArthitis.sas ; 48](#_Toc7770325)

[\* Activity\_with\_spell\_flags.sas ; 49](#_Toc7770326)

[\* AEUnitCost\_1.sas ; 50](#_Toc7770327)

[\* AEdataprep.sas ; 50](#_Toc7770328)

[\* AE\_NAUnitCost.sas ; 51](#_Toc7770329)

[\* Pre\_AdmTC.sas ; 52](#_Toc7770330)

[\* AE\_TA\_CostbyHRG.sas ; 52](#_Toc7770331)

[\* AECostBase.sas ; 54](#_Toc7770332)

[\* AEInfEffCNST1.sas ; 54](#_Toc7770333)

[\* AEManualAdjust.sas ; 55](#_Toc7770334)

[\* AEUnitCost\_2.sas ; 57](#_Toc7770335)

[\* AEfilterscope.sas ; 58](#_Toc7770336)

[\* AEUnitCost\_3.sas ; 58](#_Toc7770337)

[\* AEUnitCost\_4.sas ; 59](#_Toc7770338)

[\* AErecodeHRGs.sas ; 61](#_Toc7770339)

[\* AEremoveMFF.sas ; 61](#_Toc7770340)

[\* AEClacNationalAvg.sas ; 62](#_Toc7770341)

[\* AECleanData1.sas ; 62](#_Toc7770342)

[\* AECleanData2.sas ; 63](#_Toc7770343)

[\* A&E\_Create\_Execution\_Instance.sas ; 64](#_Toc7770344)

[\* AECostCrosstab.sas ; 65](#_Toc7770345)

[\* BPT\_AnnexA.sas ; 65](#_Toc7770346)

[\* Compliance\_rates\_for\_SDEC.sas ; 69](#_Toc7770347)

[\* Step3\_5\_Create\_all\_APC\_OPROC\_Excel\_Model\_worksheets\_1 71](#_Toc7770348)

[\* Step3\_5\_Create\_all\_APC\_OPROC\_Excel\_Model\_worksheets\_2 78](#_Toc7770349)

[\* BPT\_LinkedSheet.sas ; 103](#_Toc7770350)

[\* MPP\_Step\_9 Tables\_Lev1.sas ; 110](#_Toc7770351)

[\* MPP\_Step\_10 Tables\_Lev1.sas ; 111](#_Toc7770352)

[\* MPP\_IA.sas ; 118](#_Toc7770353)

[\* MPP\_Step\_1\_Input Tables\_Lev1.sas ; 130](#_Toc7770354)

[\* MPP\_Step\_2\_ Calculation of Parameters\_Lev1.sas ; 134](#_Toc7770355)

[\* MPP\_Aggregations.sas ; 135](#_Toc7770356)

[\* MPP\_Step\_4\_Calculation of Unit Cost\_Lev1.sas ; 149](#_Toc7770357)

[\* MPP\_Step\_5\_1\_QR1 Adjustment\_Lev1.sas ; 151](#_Toc7770358)

[\* MPP\_Step\_5\_2\_CB Adjustment\_Lev1.sas ; 153](#_Toc7770359)

[\* MPP\_Step\_5\_3\_TotalInflation\_Efficiency\_Lev1.sas ; 154](#_Toc7770360)

[\* MPP\_Step\_5\_4\_TotalCNST Adjustment\_Lev1.sas ; 156](#_Toc7770361)

[\* MPP\_Step\_5\_6\_SMF Adjustment\_Lev1.sas ; 171](#_Toc7770362)

[\* MPP\_Step\_5\_8\_CNST1920 Adjustment\_Lev1.sas ; 175](#_Toc7770363)

[\* MPP\_Step\_5\_9\_Inflation\_Efficiency201920\_Lev1.sas ; 177](#_Toc7770364)

[\* MPP\_Step\_6.2\_Calculation of Quantums\_Lev1.sas ; 178](#_Toc7770365)

[\* Maternity\_Trim\_Point\_Calculation 181](#_Toc7770366)

[\* MPP\_Step\_8\_Linked Sheet\_Lev1.sas ; 186](#_Toc7770367)

[\* MPP\_Step\_9 Tables\_Lev1.sas ; 187](#_Toc7770368)

[\* MPP\_Non-Mandatory\_Lev1.sas 190](#_Toc7770369)

[\* Non\_Mandatory\_Trim\_Point\_Calculation(For 1 level HRGs) using default SAS method to calculate Quartiles. 207](#_Toc7770370)

[\* MPP\_Linksheet.sas 211](#_Toc7770371)

[\* OMNationalAvg2.sas 218](#_Toc7770372)

[\* OMMFFAdjust\_1.sas 218](#_Toc7770373)

[\* OMMFFAdjust\_2.sas 219](#_Toc7770374)

[\* OPATT\_1\_13\_Recoding\_of\_501\_560.sas 225](#_Toc7770375)

[\* OPATT\_1\_15\_317\_by\_org.sas 226](#_Toc7770376)

[\* OPATT\_1\_16\_UC\_by\_org\_TFC\_Currency.sas 226](#_Toc7770377)

[\* OPATT\_ 01\_03d.sas 227](#_Toc7770378)

[\* OPATT\_1\_19\_Remapping\_of\_Costs.sas 228](#_Toc7770379)

[\* OPATT\_1\_1\_filter\_for\_scope.sas 228](#_Toc7770380)

[\* OPATT\_01\_01a.sas 229](#_Toc7770381)

[\* OPATT\_02\_01a.sas 229](#_Toc7770382)

[\* OPATT\_1\_21\_Crosstab\_of\_Costs.sas 229](#_Toc7770383)

[\* OPATT\_1\_22\_Make\_table\_of\_Costs\_Crosstab.sas 230](#_Toc7770384)

[\* OPATT\_1\_23\_Crosstab\_Activity.sas ; 231](#_Toc7770385)

[\* OPATT\_1\_24\_Make\_table\_of\_Activity\_Crosstab.sas ; 231](#_Toc7770386)

[\* OPATT\_1\_25\_D\_D\_Exclusions.sas ; 232](#_Toc7770387)

[\* OPATT\_02\_02a.sas ; 232](#_Toc7770388)

[\* OPATT\_1\_27\_Total\_costs\_and\_pcts\_for\_apportionment\_of\_D\_D.sas ; 232](#_Toc7770389)

[\* OPATT\_1\_28\_D\_D\_costs\_to\_remove.sas ; 233](#_Toc7770390)

[\* OPATT\_1\_29\_Revised\_Costs\_following\_removal\_of\_D\_D.sas ; 234](#_Toc7770391)

[\* OPATT\_1\_30\_D\_D\_Limiting\_of\_exclusions\_inc\_additional\_topslice.sas ; 234](#_Toc7770392)

[\* OPATT\_1\_31\_Total\_Topslice\_to\_be\_removed.sas ; 235](#_Toc7770393)

[\* OPATT\_1\_32\_pct\_Topslice.sas ; 236](#_Toc7770394)

[\* OPATT\_1\_33\_Revised\_Costs\_after\_topslice.sas ; 236](#_Toc7770395)

[\* OPATT\_1\_3\_Total\_Costs.sas ; 237](#_Toc7770396)

[\* OPATT\_1\_4\_Regroup.sas ; 237](#_Toc7770397)

[\* OPATT\_1\_5\_Remove\_MFF.sas ; 238](#_Toc7770398)

[\* OPATT\_01\_02a.sas ; 238](#_Toc7770399)

[\* OPATT\_1\_7\_National\_Averages.sas ; 239](#_Toc7770400)

[\* OPATT\_1\_8\_National\_Average\_MFF.sas ; 239](#_Toc7770401)

[\* OPATT\_1\_9\_MFF\_Rescaling.sas ; 240](#_Toc7770402)

[\* OPATT\_2\_2.sas ; 254](#_Toc7770403)

[\* APC\_Prerequisite\_Data.sas ; 264](#_Toc7770404)

[\* APC\_Grubbs\_Test.sas ; 266](#_Toc7770405)

[\* APC\_2550\_Test.sas ; 269](#_Toc7770406)

[\* APC\_Duplicates\_Test.sas ; 271](#_Toc7770407)

[\* APC\_Data\_Clean\_Step\_Final\_data\_clean.sas 277](#_Toc7770408)

[\* Create BPT Excel Model linksheets. 289](#_Toc7770409)

[\* Excel\_Step3\_1 Long Stay Payment 332](#_Toc7770410)

[\* Excel\_Step3\_2 STEP BY STEP TRIMPOINT and LSP 337](#_Toc7770411)

[\* Excel\_Step3\_3 STEP BY STEP OP-DC-EL- 349](#_Toc7770412)

[\* Excel\_Step3\_5 APC\_OPROC\_Model\_Linked\_Sheet 373](#_Toc7770413)

[\* Step3\_5 Create all APC\_OPROC Excel Model worksheets\_1 377](#_Toc7770414)

[\* Step3\_5 Create all APC\_OPROC Excel Model worksheets\_2 387](#_Toc7770415)

[\* Step2\_10\_Calculate\_Unit\_Costs 405](#_Toc7770416)

[\* 2.12 Inclusion of Clinical Negligence Scheme for Trusts(CNST) \*/ 410](#_Toc7770417)

[\* 2.13 Top-Slices 414](#_Toc7770418)

[\* Step2\_14\_Combining low volume activity 417](#_Toc7770419)

[\* Step2\_4\_Data\_Clean 428](#_Toc7770420)

/\*--------------------------------------------------------------\*/

/\*

# \* Main Program to run All TC Models (Live version)\*/

/\* Version: 1.01 \*/

/\* Author: PID \*/

/\* Date: April 2018 \*/

/\*--------------------------------------------------------------\*/

/\* Description: \*/

/\* Run all price models and generate Excel reports: \*/

/\*--------------------------------------------------------------\*/

/\* Models: \*/

/\* 1.AE Model ; \*/

/\* 2.OPROC Model; \*/

/\* 3.APC Model ; \*/

/\* 4.APC\_OPROC Excel Model; \*/

/\* 5.OPATT Model; \*/

/\* 6.Unbundled Model; \*/

/\* 7.Maternity Model; \*/

/\* 8.Other Mandatory Model. \*/

/\*--------------------------------------------------------------\*/

/\*system start time;\*/

%let Start\_Datetime=%sysfunc(DateTIME());

%put Start\_Datetime=&Start\_Datetime;

/\*Version of Smoothing Factor\*/

%let SmoothFVersion=58;

/\*Version of Scaling Factor \*/

%let ScalingFVersion=69;

**data** \_null\_;

attrib guid length=$**36**;

guid=uuidgen();

call symputx('SAS\_Session',guid,'G');

**run**;

/\*system start time;

%let Program\_start=%sysfunc(DateTIME());

**data** \_null\_;

username=sysget('USERNAME');

user\_name=translate(username, "\_", ".");

call symputx('USERNAME',username,'G');

call symputx('user\_name',user\_name,'G');

**run**;

%put USERNAME=&USERNAME.;

%put user\_name=&user\_name.;

/\*Model output sas data \*/

libname Modelout "\\irnarch\sas\_data\Tariff Rebuild\Models for Testing\TC\_Y1920\Model\_Outputs\_SAS\&username.";

%let ExcelOutFolder=\\irnarch\sas\_data\Tariff Rebuild\Models for Testing\TC\_Y1920\Excel\_outputs\&username.;

%let Mainfolder=\\irnarch\sas\_data\Tariff Rebuild\Models for Testing\TC\_Y1920\ModelMain\_SAS\_Codes;

**data** Cash\_in\_Out\_ia\_factors;

set tariff\_r.cashio\_factors\_all\_v69;

rename

CashIO\_POD\_Subch=POD\_Subchapter;

CashIO\_factor\_POD\_Subch=**0**;

**run**;

/\*Clear Modelout library \*/

**proc** **datasets** lib=Modelout

nolist kill;

**quit**;

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\*\*\*\*\*\* Step 1 Run AE Model \*\*\*\*\*\*;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

%include "&Mainfolder.\AE\_Main\_1920\_FinV\_cio\_pl.sas";

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\*\*\*\*\*\* Step 2 Run OPROC Model \*\*\*\*\*\*;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

%include "&Mainfolder.\OPROC\_Model\_Main\_1920\_FinV\_pl.sas";

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\*\*\*\*\*\* Step 3 Run APC Model \*\*\*\*\*\*;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

%include "&Mainfolder.\APC\_Main\_1920\_FinV\_1087\_40\_pl.sas";

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\*\*\*\*\*\* Step 4 Run APC\_OPROC Excel Model \*\*\*\*\*\*;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

%include "&Mainfolder.\APC\_OPROC\_Excel\_Model\_Main\_1920\_FinV\_cio\_SF\_NE\_test\_s118\_ia\_cio\_pl.sas";

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\*\*\*\*\*\* Step 5 Run OPATT Model \*\*\*\*\*\*;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

%include "&Mainfolder.\OPATT\_Model\_Main\_1920\_FinV\_s118\_pl.sas";

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\*\*\*\*\*\* Step 6 Run Unbundled Model \*\*\*\*\*\*;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

%include "&Mainfolder.\Unbundled\_Model\_Main\_1920\_FinV\_s118\_pl\_bpt.sas";

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\*\*\*\*\*\* Step 7 Run Maternity Model \*\*\*\*\*\*;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

/\*%include "&Mainfolder.\MPP\_Main\_1920\_FinV\_1087\_40.sas";\*/

%include "&Mainfolder.\MPP\_Main\_1920\_FinV\_1087\_40\_6levels\_cio\_AIP\_v2.sas";

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\*\*\*\*\*\* Step 8 Run Other Mandatory Model \*\*\*\*\*\*;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

%include "&Mainfolder.\OtherMandatory\_Main\_1920\_FinV\_pl.sas";

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\*\*\*\*\*\* Step 9 Run BPT Model \*\*\*\*\*\*;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

%include "&Mainfolder.\BPT\_Main\_1920\_FinV\_1087\_40\_bpt\_pl\_v2.sas";

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\*\*\*\*\*\* Step 10 Run Excel Final Price output for All Models \*\*\*\*\*\*;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

%include "&Mainfolder.\ExcelOut1\_All\_Models\_Final\_Prices\_FinV\_lvl6.sas";

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\*\*\*\*\*\* Step 11 Run Excel Step by Step Price output for All Models \*\*\*\*\*\*;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

%include "&Mainfolder.\ExcelOut2\_All\_Models\_Prices\_Step\_by Step\_FinV\_SF\_TEST\_lvl6.sas";

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\*\*\*\*\*\* All Program Running time Summary \*\*\*\*\*\*;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

/\*system End time;\*/

%let End\_Datetime=%sysfunc(DateTIME());

%put End\_Datetime=&End\_Datetime.;

**data** system\_running\_time;

Program\_Start=put(&Start\_Datetime.,datetime.);

Program\_End =put(&End\_Datetime.,datetime.) ;

Running\_Time= put(sum(&End\_Datetime.,-&Start\_Datetime.),mmss.);

**run**;

**proc** **sql**;

select Program\_Start, Program\_End, Running\_Time into :Program\_Start, :Program\_End ,:Running\_Time

from system\_running\_time

;

**quit**;

%put "Program Started at : &Program\_Start..";

%put "Program Finished at : &Program\_End..";

%put "Total Running Time: &Running\_Time. minutes. ";

/\*Step 0 Main Run ALL Programs for AE Tariff \*/

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 30 March 2017 ;

\*

# \* AE\_Main\_Run\_all\_steps\_local.sas ;

\* description: ;

\* version: 1.00 ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

%let folder=\\irnarch\SAS\_Data\Tariff Rebuild\Models for Testing\TC\_Y1920\AE\SAS\_Codes;

/\*data work.Reference\_Cost; set tariff\_r.Reference\_Cost\_1415; run;\*/

**PROC** **SQL**;

CREATE table work.Reference\_Cost AS SELECT

RCID,

RC\_Year,

OrganisationCode,

OrganisationName,

CurrencyCode,

CurrencyName,

DepartmentCode,

DepartmentName,

ServiceCode,

ServiceName,

SupplierType,

Bed\_Days as BedDays,

Activity,

Mean,

Actual\_Cost as ActualCost,

Expected\_Cost as ExpectedCost,

Unit\_Cost as UnitCost,

Mapping\_pot as MappingPot

from TARIFF\_R.REFERENCE\_COST\_1617\_PUBLISHED;

**QUIT**;

**data** work.tc\_mff; set tariff\_r.tc\_mff;

WHERE YEAR="1920";

**run**;

**data** work.QUANTUMRECONCILATION\_OUTPUT; set tariff\_r.QUANTUMRECONCILATION\_OUTPUT;

where year="19/20";

**run**;

**data** work.HRG\_recode; set tariff\_r.HRG\_recode; **run**;

%let year\_ae2apc="19/20";

%let id\_ae2apc=2;

%let Cost\_Base\_Adjustment = 0;

**proc** **sql**;

create table n1 as select \* from tariff\_r.cnst

where hrgsubchapter="VB" and year in("17/18" "18/19");

**quit**;

**data** n2; set n1; if year="17/18" then year="17\_18"; else if year="18/19" then year="18\_19"; else year=year;**run**;

**proc** **transpose** data=n2 out=n3; id year; var cnst; **run**;

**data** n4; set n3; cnst=(\_17\_18+**1**)\*(\_18\_19+**1**)-**1**; **run**;

**proc** **sql**;

select cnst into :cnst

from n4;

**quit**;

%let CNST2 = 0.00000;

**data** inf\_eff; set tariff\_r.inflation\_efficiency; **run**;

**proc** **transpose** data=inf\_eff out=inf\_eff2; id year; var inflation\_and\_efficiency\_combine; **run**;

**data** inf\_eff3; set inf\_eff2; ie\_index\_adjst\_fact=(\_17\_18+**1**)\*(\_18\_19+**1**)-**1**; **run**;

**proc** **sql**;

select ie\_index\_adjst\_fact into :ie\_indextn\_adjstmt\_factors

from inf\_eff3;

**quit**;

%let Inf\_Eff=&ie\_indextn\_adjstmt\_factors.;

/\*Smoothing Factor \*/

**PROC** **SQL**;

SELECT Combined\_Uplifts INTO :smf

from tariff\_r.smf\_V&SmoothFVersion.

where subchapter="VB";

**quit**;

%let smf=&smf;

%put smf=&smf;

/\*Scaling Factor \*/

**proc** **sql**;

select SF into :scaling\_factor

from tariff\_r.sf

where year="19/20";

**quit**;

%let scaling\_factor=&scaling\_factor;

%put scaling\_factor=&scaling\_factor;

**data** inf\_eff\_s118; set tariff\_r.inflation\_efficiency; WHERE Year="19\_20"; **run**;

**data** inf\_eff\_s118\_2; set inf\_eff\_s118;inflation\_and\_efficiency\_combine=(Inflation+**1**)\*(Efficiency+**1**)-**1**; **run**;

**proc** **sql**;

select inflation\_and\_efficiency\_combine into :Inf\_Eff\_1920

from inf\_eff\_s118\_2

where ie\_id=**6**;

**quit**;

**proc** **sql**;

select CNST into :CNST\_1920

from tariff\_r.cnst

where Year="19/20" AND hrgsubchapter="VB";

**quit**;

%let CNST2\_1920 = 0;

%include "&folder.\AE Step 1\_1 Preparing data\_0\_2.sas";

%include "&folder.\AE Step 1\_2 Filter scope\_v0\_3.sas";

%include "&folder.\AE Step 1\_3 Recode HRGS\_v0\_3.sas";

%include "&folder.\AE Step 1\_4 Remove MFF\_0\_2.sas";

%include "&folder.\AE Step 1\_5 Calc National Averages.sas";

%include "&folder.\AE Step 1\_6 Clean data first step\_0\_2.sas";

%include "&folder.\AE Step 1\_7 Clean data step 2.sas";

%include "&folder.\AE Step 1\_8 Activity cross tab\_v0\_2.sas";

%include "&folder.\AE Step 1\_9 Cost crosstab.sas";

%include "&folder.\AE Step 1\_10 Apply NA Costs to AD Activity.SAS";

%include "&folder.\AE Step 1\_11 Pre and Post Adm TC.SAS";

%include "&folder.\AE Step 1\_12 Totalcost and activity by HRG.SAS";

\* Previous Excel parts consist of steps 13 - 21;

%include "&folder.\AE Step 1\_ 13 Unit Cost.sas";

%include "&folder.\AE Step 1\_14\_b QR1.sas";

%include "&folder.\AE Step 1\_15 Cost base\_1920.sas";

%include "&folder.\AE Step 1\_16 Infl\_Eff\_CNST this year\_1920.sas";

%include "&folder.\AE Step 1\_17 Manual adjustment implementation\_v0\_2\_1920\_ma.sas";

%include "&folder.\AE Step 1\_17b\_output\_for\_IA\_1920.sas"; /\* to link to IA models for smoothing and scaling \*/

%include "&folder.\AE Step 1\_18 Smoothing UPDATED\_v0\_2\_1920.sas";

%include "&folder.\AE Step 1\_19 Scaling\_v0\_2\_1920.sas";

%include "&folder.\AE Step 1\_20 Infl\_Eff\_CNST next year\_v0\_2\_1920\_cio.sas";

%include "&folder.\AE Step 1\_21 Final price and analysis outputs updated\_v0\_2.sas";/\* final prices and step by step prices\*/

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\*

# \* APC\_OPROC\_Model\_Main\_Flat\_Files\_1.sas ;

\* Author: PID ;

\* date: August 2017 ;

\* description: ;

\* version: 1.00 ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

%let FYear=19/20;

%let Year=1920;

%let Run\_ID=40;

\* param\_HESRunID;

%let param\_HESRunID=40;

%let GrouperRunID =1087;

%let HRG\_MAP\_ID=2;

LIBNAME bprPrep ODBC DATASRC=DSN\_Pbr\_Repository\_New SCHEMA=PreProcessOUT;

LIBNAME bprhesg ODBC DATASRC=DSN\_Pbr\_Repository\_New SCHEMA=HES\_Grouped;

/\*Output log file ;

**proc** **printto**;

/\*SAS code folder \*/

%let folder=\\irnarch\sas\_data\Tariff Rebuild\Models for Testing\TC\_Y1920\APC\SAS\_Codes\Source\_Tariff\_R\V2;

%let outputpath=\\irnarch\sas\_data\Tariff Rebuild\Models for Testing\TC\_Y1920\APC\output\&username.\Summary;

%let outputpath2=\\irnarch\sas\_data\Tariff Rebuild\Models for Testing\TC\_Y1920\APC\output;

%let code\_version=1.00;

/\*Assign SQL input library;

%let lib=WORK;

%let StageLib=APCSTAGE;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\*\*\*\*\*\* Assign SQL view library \*\*\*\*\*\*;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

%let SQLViewLib=SS\_ViewT;

LIBNAME &SQLViewLib. ODBC DATASRC=DSN\_Pbr\_Repository\_New SCHEMA=PreProcessOUT;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\*\*\*\*\*\* All Input tables----------- Start from here \*\*\*\*\*\*;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

%let RefCost\_Selected=Tariff\_R.REFERENCE\_COST\_1617\_PUBLISHED;

**data** TC\_MFF\_Selected;

set Tariff\_R.TC\_MFF;

where Year="&Year.";

Target\_MFF=Uncapped\_MFF;

**run**;

/\*Step\_1\_1 no input table \*/

/\*Step\_1\_2 no input table \*/

/\*Step\_1\_3 no input table \*/

/\*Step\_1\_4 no input table \*/

/\*Step\_1\_5 no input table \*/

/\*Step\_1\_6 no input table \*/

/\*Step\_2\_1 no input table \*/

/\*Step\_2\_3 no input table \*/

/\*Step\_2\_4 no input table \*/

/\*Step2\_5 \*/

/\*Table1 \*/

**data** Tv\_HES\_Adm\_AE;

set &SQLViewLib.**.**'HES\_Adm\_A&E'n;

where run\_ID=&param\_HESRunID.;

**run**;

/\*table2 \*/

**data** AE\_ADM\_COST\_QUANTUM2APC;

set ModelOut.AE\_ADM\_COST\_QUANTUM2APC;

**run**;

/\*Step2\_6 \*/

/\*Table1 \*/

**data** NICE\_HRGs;

set Tariff\_R.NICE\_HRGs;

**run**;

/\*Table2 \*/

**data** NICE\_Costs;

set Tariff\_R.NICE\_Costs;

**run**;

/\*Step2\_7 \*/

/\*Table1\*/

**data** TrimpointsRC\_HRGs;

set Tariff\_R.TrimpointsRC\_HRGs;

**run**;

/\*Table2\*/

**data** Tv\_HES\_FCE\_Spell\_Map\_LoS;

set &SQLViewLib.**.H**ES\_FCE\_Spell\_Map\_LoS;

where Run\_ID = &param\_HESRunID.;

**run**;

/\*Table3\*/

**data** HRG\_ELIGIBILITY;

set Tariff\_R.HRG\_ELIGIBILITY;

**run**;

/\*Table4\*/

**data** Vt\_HES\_Spell\_Counts\_Adm;

set &SQLViewLib.**.H**ES\_Spell\_Counts\_Adm;

where Run\_ID = &param\_HESRunID.;

**run**;

/\*Table5\*/

**data** HRGSELIGIBILITY\_TEMP;

set Tariff\_R.HRGSELIGIBILITY\_TEMP;

**run**;

/\*Step2\_8 \*/

/\*Table1\*/

**data** Vt\_HES\_Count\_Spell\_LoS;

set &SQLViewLib.**.H**ES\_Count\_Spell\_LoS;

where Run\_ID = &param\_HESRunID.;

**run**;

/\*Table2\*/

**data** Vt\_HES\_Trimpoints;

set &SQLViewLib.**.H**ES\_Trimpoints;

where Run\_ID = &param\_HESRunID.;

**run**;

/\*Table3\*/

**data** EligibleHRG\_UZ01Z\_PB03Z;

set Tariff\_R.EligibleHRG\_UZ01Z\_PB03Z;

**run**;

/\*Step2\_9 \*/

/\*Table1\*/

**data** D\_and\_D;

set Tariff\_R.D\_and\_D;

where dd\_id=**3**;

**run**;

/\*Table2\*/

**data** D\_and\_D\_topslice;

set Tariff\_R.D\_and\_D\_topslice;

where dd\_id=**3**;

**run**;

/\*Step\_2\_10 no input table \*/

/\*Step\_2\_11\*/

/\*Table1\*/

**data** SSEM\_Banding;

set Tariff\_R.SSEM\_Banding;

**run**;

/\*Table2\*/

**data** Tv\_SSEM\_Count;

set &SQLViewLib.**.S**SEM\_Count;

where Run\_ID = &param\_HESRunID.;

**run**;

/\*Step\_2\_12\*/

/\*Table1\*/

**data** APC\_CNST\_SubChapter;

set Tariff\_R.APC\_CNST\_SubChapter;

**run**;

/\*Table2\*/

**data** APC\_CNST\_Chapter;

set Tariff\_R.APC\_CNST\_Chapter;

**run**;

/\*Step\_2\_13\*/

/\*Table1\*/

**data** APC\_Topups\_Topslice;

set Tariff\_R.APC\_Topups\_Topslice;

**run**;

/\*Table2\*/

**data** APC\_ICRS;

set Tariff\_R.APC\_ICRS;

**run**;

/\*Table3\*/

**data** ZeroPriceHRG;

set Tariff\_R.ZeroPriceHRG ;

**run**;

/\*Step\_2\_14 no input table \*/

/\*Step\_2\_15\*/

/\*Table1\*/

**data** Uplift;

set Tariff\_R.Uplift;

**run**;

/\*Table2\*/

**data** Scaling;

set Tariff\_R.Scaling;

**run**;

**data** I\_E;

set TARIFF\_R.I\_E;

Inflation\_and\_Efficiency\_Combine=(Inflation+**1**)\*(Efficiency+**1**)-**1**;

**run**;

**data** cnst;

set tariff\_r.cnst;

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\*\*\*\*\*\* All Input tables----------- End here \*\*\*\*\*\*;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

/\*Initialise parameters;

%include "&folder.\APC\_Initialise\_Parameters\_FinV.sas";

/\* start to output log file;

/\*Get prerequisite data;

%include "&folder.\Program APC data Clean\_Step\_1\_1\_Prerequisite\_Data\_FinV.sas";

/\*Do Data Cleaning;

%include "&folder.\Program APC data Clean\_Step\_1\_2\_Grubbs\_Test\_FinV.sas";

%include "&folder.\Program APC data Clean\_Step\_1\_3\_2550\_Test\_FinV.sas";

%include "&folder.\Program APC data Clean\_Step\_1\_4\_Duplicates\_Test\_FinV.sas";

%include "&folder.\Program APC data Clean\_Step\_1\_5 Create flags for EBD\_FinV.sas";

%include "&folder.\Program APC Data Clean\_Step\_1\_6 Final data clean\_FinV.sas";

/\*Run Model Step 2\*/

%include "&folder.\Program\_Step2\_1\_Calculate\_Inlier\_EBD\_Cost\_FinV.sas";

%include "&folder.\Program\_Step2\_3\_Remove\_MFF\_FinV.sas";

%include "&folder.\Program\_Step2\_4\_Data\_Clean\_FinV.sas";

%include "&folder.\Program\_Step2\_5\_Inclusion\_Cost\_in\_AE\_to\_Admission\_FinV.sas";

%include "&folder.\Program\_Step2\_6\_NICE\_Technology\_Appraisals\_FinV.sas";

%include "&folder.\Program\_Step2\_7\_FCE\_to\_Spell\_Cost\_Conversion\_FinV.sas";

%include "&folder.\Program\_Step2\_8\_Calculate\_TrimPoints\_FinV.sas";

%include "&folder.\Program\_Step2\_9\_Removal\_of\_costs\_associated\_with\_DD\_FinV.SAS";

%include "&folder.\Program\_Step2\_10\_Calculate\_Unit\_Costs\_FinV.SAS";

%include "&folder.\Program\_Step2\_11\_Short\_Stay\_Emergency\_Tariff\_Adjust\_FinV.SAS";

%include "&folder.\Program\_Step2\_12\_Inclusion\_of\_CNST\_FinV.SAS";

%include "&folder.\Program\_Step2\_13\_Top\_Slices\_FinV.SAS";

%include "&folder.\Program\_Step2\_14\_Combining low volume activity\_FinV.SAS";

%include "&folder.\Program\_Step2\_15\_Stage2\_Output\_Price\_FinV.SAS";

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\*

# \* APC\_OPROC\_Model\_Main\_Flat\_Files\_2.sas ;

\* Author: PID ;

\* date: August 2017 ;

;

\* version: 1.00 ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

%let param\_HESRunID=29;

\*Output log file ;

**proc** **printto**;

%let folder=\\irnarch\sas\_data\PID\_Proj\_1718\APC\APC\_PIDTest\SAS\_Codes\Source\_FlatFiles;

%let inputpath=\\irnarch\sas\_data\PID\_other\_data;

%let inputpath1=\\irnarch\sas\_data\PID\_sas\_data\otherdata;

%let outputpath=\\irnarch\sas\_data\PID\_Proj\_1718\APC\APC\_PIDTest\output\Source\_FlatFiles\Summary;

%let outputpath2=\\irnarch\sas\_data\PID\_Proj\_1718\APC\APC\_PIDTest\output\Source\_FlatFiles;

%let code\_version=1.00;

\*Assign SQL input library;

%let lib=WORK;

%let SQLInputLib=SQL\_INP;

%let MDSInputLIb=MDS\_INP;

%let siprefco=siprefco;

%let SIP\_CONF=SIP\_CONF;

%let StageLib=APCSTAGE;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\*\*\*\*\*\* Assign SQL view library \*\*\*\*\*\*;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

%let SQLViewLib=SS\_View;

LIBNAME &SQLViewLib. ODBC DATASRC=Pbr1718\_\_28 SCHEMA=dbo;

%let SQLStage=SQL\_STG;

LIBNAME &SQLStage. ODBC DATASRC=Pbr1718\_\_28 SCHEMA=Stage;

LIBNAME &SQLInputLib. ODBC DATASRC=Pbr1718\_\_28 SCHEMA=Input;

LIBNAME &MDSInputLIb. ODBC DATASRC=DSN\_MDS SCHEMA=mdm;

LIBNAME &siprefco. ODBC DATASRC=SAS\_SIP\_UAT\_64 SCHEMA=refcost;

LIBNAME &SIP\_CONF. ODBC DELETE\_MULT\_ROWS=YES UPDATE\_MULT\_ROWS=YES INSERT\_SQL=YES DATASRC=DSN\_DEV\_SAS\_PRICING SCHEMA=sas\_config authdomain='DefaultAuth' ;

libname newhes "&inputpath1.";

/\*Initialise parameters;

%include "&folder.\APC\_Initialise\_Parameters.sas";

/\* start to output log file;

/\*Get prerequisite data;

%include "&folder.\Program APC data Clean\_Step\_1\_1\_Prerequisite\_Data\_NewV.sas";

\*Do Data Cleaning;

%include "&folder.\Program APC data Clean\_Step\_1\_2\_Grubbs\_Test\_NewV.sas";

%include "&folder.\Program APC data Clean\_Step\_1\_3\_2550\_Test\_NewV.sas";

%include "&folder.\Program APC data Clean\_Step\_1\_4\_Duplicates\_Test\_NewV.sas";

%include "&folder.\Program APC data Clean\_Step\_1\_5 Create flags for EBD\_NewV.sas";

%include "&folder.\Program APC Data Clean\_Step\_1\_6 Final data clean\_New.sas";

%let vSummaryStats=NO;

**%macro** Run\_SummaryStats();

%if %upcase(&vSummaryStats.)=YES %then %do;

%include "&folder.\Program APC data Clean\_Step\_1\_7\_Impact\_and\_Stats.sas";

%end;

%else %do;

%put No Summary Statistics output for Data Clean;

%end;

**%mend** Run\_SummaryStats;

%***Run\_SummaryStats***()

%let vDC=NO;

\*Run Stage 2;

**%macro** ***Run\_DC***;

%put vDC=&vDC;

%if %upcase(&vDC.)=YES %then %do;

%include "&folder.\Program\_Step2\_1\_Calculate\_Inlier\_EBD\_Cost\_NewV.sas";

%include "&folder.\Program\_Step2\_3\_Remove\_MFF\_NewV.sas";

%include "&folder.\Program\_Step2\_4\_Data\_Clean\_NewV.sas";

%include "&folder.\Program\_Step2\_5\_Inclusion\_Cost\_in\_AE\_to\_Admission\_NewV.sas";

%include "&folder.\Program\_Step2\_6\_NICE\_Technology\_Appraisals\_NewV.sas";

%include "&folder.\Program\_Step2\_7\_FCE\_to\_Spell\_Cost\_Conversion\_NewV.sas";

%include "&folder.\Program\_Step2\_8\_Calculate\_TrimPoints\_NewV.sas";

%include "&folder.\Program\_Step2\_9\_Removal\_of\_costs\_associated\_with\_DD\_NewV.SAS";

%include "&folder.\Program\_Step2\_10\_Calculate\_Unit\_Costs\_NewV.SAS";

%include "&folder.\Program\_Step2\_10\_1\_Project\_Report\_NewV.SAS";

%include "&folder.\V3DC\_Step2\_10\_2\_excel\_Report\_DC.SAS";

%end;

%else %do;

%include "&folder.\Program\_Step2\_1\_Calculate\_Inlier\_EBD\_Cost\_NewV.sas";

%include "&folder.\Program\_Step2\_3\_Remove\_MFF\_NewV.sas";

%include "&folder.\Program\_Step2\_4\_Data\_Clean\_NewV.sas";

%include "&folder.\Program\_Step2\_5\_Inclusion\_Cost\_in\_AE\_to\_Admission\_NewV.sas";

%include "&folder.\Program\_Step2\_6\_NICE\_Technology\_Appraisals\_NewV.sas";

%include "&folder.\Program\_Step2\_7\_FCE\_to\_Spell\_Cost\_Conversion\_NewV.sas";

%include "&folder.\Program\_Step2\_8\_Calculate\_TrimPoints\_NewV.sas";

%include "&folder.\Program\_Step2\_9\_Removal\_of\_costs\_associated\_with\_DD\_NewV.SAS";

%include "&folder.\Program\_Step2\_10\_Calculate\_Unit\_Costs\_NewV.SAS";

%include "&folder.\Program\_Step2\_10\_1\_Project\_Report\_NewV.SAS";

%include "&folder.\Program\_Step2\_11\_Short\_Stay\_Emergency\_Tariff\_Adjust\_NewV.SAS";

%include "&folder.\Program\_Step2\_12\_Inclusion of Clinical Negligence Scheme for Trusts\_NewV.SAS";

%include "&folder.\Program\_Step2\_13\_Top\_Slices\_NewV.SAS";

%include "&folder.\Program\_Step2\_14\_Combining low volume activity\_NewV.SAS";

%include "&folder.\Program\_Step2\_15\_Stage2\_Output\_Price\_NewV.SAS";

%end;

**%mend** Run\_DC;

%***Run\_DC***;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* Author: PID ;

\* Modified PID

\* Date: July 2018

\* ;

# \* BPT\_Genaral\_Settings.sas ;

\* Description: 1920 ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

/\*All the parameters in BPT models (except for compliance and differential rates that need to be set in the excel file) are set here below.\*/

libname BPTInp\_H "\\irnarch\sas\_data\Tariff Rebuild\Models for Testing\TC\_Y1920\BPT\BPT\_inputSQLData."; /\*this library has HES inputs that we need FOR USE IN ACTIVITY\_SPELLFLAG PROG \*/

libname BPTInput "\\irnarch\sas\_data\Tariff Rebuild\Models for Testing\TC\_Y1920\Model\_Outputs\_SAS\&username."; /\*this library has all other inputs that we need\*/

libname BPTOut "\\irnarch\sas\_data\Tariff Rebuild\Models for Testing\TC\_Y1920\BPT\BPT\_SASOutput\&username."; /\*this library has all the output tables that we need \*/

/\*Settings for Group1:\*/

%let apc\_table\_ne\_price = BPTInput.apc\_oproc\_model\_linkedsheet;

%let currentyear = '19/20';

%let outputpath=\\irnarch\sas\_data\Tariff Rebuild\Models for Testing\TC\_Y1920\BPT\BPT\_ExcelOutput\&username.;

%let outputfile=BPT;

%let currentyear4digits= 1920;

%let BPTinputyear='1920';

/\*Settings for Group2:\*/

%let year\_BPT\_previousyear = '18/19';

/\*Settings for Renal:\*/

%let Access\_Rate =0.8;

%let Price\_Ratio\_NoBPT\_BPT=0.8;

%Let Multiplier\_homeDialysis= 3;

%let inputtableUnbundledBPT= BPTInput.renal\_ckd\_4ia\_Model\_1920;

%let reconcileTargetQuantum = 1; /\*this can be 1 (yes) or 0 (no)\*/

%let RolloverTable= tariff\_r.renal\_1819;

/\*Settings for Outpatients\*/

%let ActivityTable = BPTInput.quantum\_reconciliation\_bysubchap;

%let Opatt\_prices = BPTInput.OPATT\_PRICES\_1920;

%let Opatt\_prices\_step\_by\_step =BPTInput.OPATT\_PRICES\_1920\_STEP\_BY\_STEP;

%let RelFact\_PleuEff = 1.1 ;

/\*Settings for relativity factor pleural effusion\*/

%let YearPaedEp=1920;

%let PreProcessingRunID=40;

%let GrouperRunID=1087;

%let I\_E\_FactorTable=TARIFF\_R.I\_E;

/\*Settings for Early Inflammatory Arthitis\*/

%let N\_TopUpInYear=4145; /\*this parameter is the number of top ups in year (receive full price)\*/

%let Perc\_diff\_TopUp\_NormalPrice= 0.671;/\*Percentage differential between top-up and normal prices\*/

/\*Settings to Create BPT link sheet excel file:\*/

%let BPTExcelOutFolder=\\irnarch\SAS\_Data\Tariff Rebuild\Models for Testing\TC\_Y1920\BPT\BPT\_ExcelOutput\&username.;

libname bptsql "\\irnarch\SAS\_Data\Tariff Rebuild\Models for Testing\TC\_Y1920\BPT\BPT\_inputSQLData";

/\* Import 201617 BPT Manual Adjustment into SAS system,

the user needs to replace the old "1920\_BPT\_manual\_adjustment\_prices"

with the last new year here: T:\Tariff Rebuild\BPT\Version\_1\BPT\_Parameters \*/

**PROC** **IMPORT** OUT=BPTInput.BPT\_ManualAdjustment

DATAFILE= "\\irnarch\sas\_data\Tariff Rebuild\Models for Testing\TC\_Y1920\BPT\BPT\_Parameters\1920\_BPT\_manual\_adjustment\_prices.xlsx"

DBMS=xlsx REPLACE;

**run**;

/\* Import previous year BPT tariffs into SAS system,

the user needs to replace the old "Annex\_A\_national\_prices\_and\_national\_tariff\_workbook"

with the last new year here: T:\Tariff Rebuild\BPT\Version\_1\BPT\_Parameters \*/

**PROC** **IMPORT** OUT=BPT\_previousyears

DATAFILE= "\\irnarch\sas\_data\Tariff Rebuild\Models for Testing\TC\_Y1920\BPT\BPT\_Parameters\BPT\_previous\_year\_NOT\_modify.xlsx"

DBMS=xlsx REPLACE;

**run**;

/\*currency design table\*/

**PROC** **IMPORT** OUT=BPTInput.BPT\_parameter\_all

DATAFILE= "\\irnarch\sas\_data\Tariff Rebuild\Models for Testing\TC\_Y1920\BPT\BPT\_Parameters\CurrencyDesignTable\_AllYears.xlsx"

DBMS=xlsx REPLACE;

**run**;

/\*Activity\_spell\_Flag: without Engagement grouper, like we did for TED,we use an imported table with BPTflag\*/

**PROC** **IMPORT** OUT=BPTInput.BPTflag

DATAFILE= "\\irnarch\sas\_data\Tariff Rebuild\Models for Testing\TC\_Y1920\BPT\BPT\_Parameters\E1920\_BPTflag.xlsx"

DBMS=xlsx REPLACE;

**run**;

**proc** **sql**;

create table BPTInput.APC\_OPROClist

as

select HRG\_Code

from ModelOut.APC\_OPROC\_MODEL\_LINKEDSHEET

;

**quit**;

/\* These are the list of programs that they will run in sequence:\*/

%let folder=\\irnarch\sas\_data\Tariff Rebuild\Models for Testing\TC\_Y1920\BPT\SAS\_Codes\s118\_pl;

%include "&folder.\00\_Group 1\_S118\_integrated.sas";

%include "&folder.\00\_Group 2\_S118\_integrated.sas";

%include "&folder.\00\_Group 2\_S118\_integrated.sas";

%include "&folder.\01\_Stroke\_S118\_integrated.sas";

%include "&folder.\02\_Renal\_modelImplemented\_S118\_integrated.sas";

%include "&folder.\04\_Endoscopy\_S118\_integrated.sas";

%include "&folder.\05\_FHF\_S118\_integrated.sas";

%include "&folder.\06\_Outpatients\_S118\_integrated.sas";

%include "&folder.\07\_Paed\_Ep\_S118\_integrated.sas";

%include "&folder.\08\_Pleural\_Effusion\_S118\_integrated.sas";

%include "&folder.\10\_TIA\_S118\_integrated.sas";

%include "&folder.\11\_EarlyInflammArthritis\_S118\_integrated.sas";

%include "&folder.\Activity\_SpellFlag\_S118\_integrated\_v2.sas";

%include "&folder.\03\_DayCase\_S118\_integrated.sas";

%include "&folder.\Compliance\_Rate\_SDEC\_S118\_integrated\_v2.sas";

%include "&folder.\Spinal\_Surgery.sas";

%include "&folder.\Linked\_sheet\_S118\_integrated.sas";

%include "&folder.\Linked\_sheet\_S118\_integrated.sas";

%include "&folder.\Program BPT link Sheets\_S118\_integrated\_pl.sas";

%include "&folder.\AnnexA\_BPT\_APC\_OPROC\_S118\_integrated.sas";

%include "&folder.\AnnexA\_for\_APC\_BPT\_Replacement\_pl.sas";

**data** BPTInput.Renal\_IAmodel;

set bptout.Renal\_IAmodel;

tariff=tariff/(**1**+&Infla\_Effi\_final\_year\_uplift.);

format tariff comma16.15;

**run**;

;\*/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* Author: PID ;

\* Date: May 2017 ;

\* Modified by: PID ;

\* Date: March 2018 ;

# \* MPP\_Main.sas ;

\* Version: 1.02 ;

\* Description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

/\*Previous SQL and Excel parts consist of steps 1 - 6\*/

%let Year=1920;

%let FYear=19/20;

%let MPP\_CDID=3;

%let Param\_OutlierMinValue = 0.05;

%let Param\_OutlierMaxValue = 20;

/\*According to requests by NHSE, the amount of quantum related to specialised fetal medician should be taken from the total cost at antenatal phase.\*/

%let SFM\_Quantum = 19301973;

/\*According to requests by NHSE, the amount of quantum below should be taken from the total cost at Delivery phase.\*/

%let DELI\_Quantum = 5400000;

%let Postnatal\_Quantum = 0;

/\* ALL levels \*/

LIBNAME bprPrep ODBC DATASRC=DSN\_Pbr\_Repository\_New SCHEMA=PreProcessOUT;

LIBNAME bprhesg ODBC DATASRC=DSN\_Pbr\_Repository\_New SCHEMA=HES\_Grouped;

LIBNAME bprPrPP ODBC DATASRC=DSN\_Pbr\_Repository\_New SCHEMA=PreProcess;

LIBNAME InputMPP base "\\irnarch\sas\_data\Tariff Rebuild\Models for Testing\TC\_Y1920\MPP\1.InputMPP\&username.";

Libname ParMPP base "\\irnarch\sas\_data\Tariff Rebuild\Models for Testing\TC\_Y1920\MPP\2.ParMPP";

%let datafilepath=\\irnarch\sas\_data\Tariff Rebuild\Models for Testing\TC\_Y1920\MPP\2.ParMPP;

%let TCurrenDesign=Currency Design Table v5.xlsx;

%let MPP\_ParameterTable=MPP\_ParameterTable v5.xlsx;

**%macro** ***Decide\_levels***;

/\*Import Currency Design table \*/

proc import out=Currency\_Desing\_1

datafile= "&datafilepath.\&TCurrenDesign."

dbms=xlsx replace;

sheet="Currency Design";

getnames=yes;

run;

/\*Delivery only \*/

proc sql noprint;

create table Currency\_Desing\_2

as

select MPP\_CDID,

Year,

HRG,

Description,

Pathway,

Delivery\_Complications\_Flag

from Currency\_Desing\_1

where Year = "&FYear." and Pathway="Delivery" and MPP\_CDID=&MPP\_CDID.

;

quit;

/\*Count how many levels?\*/

proc sql noprint;

select count(distinct Delivery\_Complications\_Flag) into :n\_level

from Currency\_Desing\_2

;

quit;

%put n\_level=&n\_level.;

%if %eval(&n\_level.)=**2** %then %do;

%let Run\_ID = 40;

%put "Run 2 levles code !!";

%end;

%else %if %eval(&n\_level.)=**6** %then %do;

%let lev=6Lev;

%put lev=&lev.;

%let Run\_ID=40;

%let param\_HESRunID=40;

LIBNAME AGGMPP base "\\irnarch\sas\_data\Tariff Rebuild\Models for Testing\TC\_Y1920\MPP\MPP\_Levels\6\_Levels\3.AGGMPP\&username.";

LIBNAME UCMPP base "\\irnarch\sas\_data\Tariff Rebuild\Models for Testing\TC\_Y1920\MPP\MPP\_Levels\6\_Levels\4.UCMPP\&username.";

LIBNAME PriceMPP base "\\irnarch\sas\_data\Tariff Rebuild\Models for Testing\TC\_Y1920\MPP\MPP\_Levels\6\_Levels\5.PriceMPP\&username.";

libname OutSAS base "\\irnarch\sas\_data\Tariff Rebuild\Models for Testing\TC\_Y1920\MPP\MPP\_Levels\6\_Levels\MPP\_OutputSAS\&username.";

%let inputpath=\\irnarch\sas\_data\Tariff Rebuild\Models for Testing\TC\_Y1920\MPP\1.InputMPP\&username.;

%let ucpath=\\irnarch\sas\_data\Tariff Rebuild\Models for Testing\TC\_Y1920\MPP\MPP\_Levels\6\_Levels\4.UCMPP\&username.;

%let outputpath=\\irnarch\sas\_data\Tariff Rebuild\Models for Testing\TC\_Y1920\MPP\MPP\_Levels\6\_Levels\5.PriceMPP\&username.;

%let folder\_6Levels=\\irnarch\SAS\_Data\Tariff Rebuild\Models for Testing\TC\_Y1920\MPP\MPPCodes\MPP\_Code\_6\_Levles;

%put "run 6 level Program ";

%include "&folder\_6Levels.\MPP\_Step\_1\_Input Tables\_6Levels\_TRNV.sas";

%include "&folder\_6Levels.\MPP\_Step\_2\_Calculation of Parameters\_6Levels\_TRNV.sas";

%include "&folder\_6Levels.\MPP\_Step\_3\_Aggregation of Input Tables\_6Levels\_TRNV.sas";

%include "&folder\_6Levels.\MPP\_Step\_4\_Calculation of Unit Cost\_6Levels\_TRNV.sas";

%include "&folder\_6Levels.\MPP\_Step\_5\_1\_QR1 Adjustment\_6Levels\_TRNV.sas";

%include "&folder\_6Levels.\MPP\_Step\_5\_2\_CB Adjustment\_6Levels\_TRNV.sas";

%include "&folder\_6Levels.\MPP\_Step\_5\_3\_TotalInflation\_Efficiency\_6Levels\_TRNV.sas";

%include "&folder\_6Levels.\MPP\_Step\_5\_4\_TotalCNST Adjustment\_6Levels\_TRNV.sas";

%include "&folder\_6Levels.\MPP\_Step\_5\_5\_QR2\_Adjustment\_6Levels\_TRNV\_AIP.sas";

%include "&folder\_6Levels.\MPP\_Step\_5\_6\_SMF Adjustment\_6Levels\_TRNV\_AIP.sas";

%include "&folder\_6Levels.\MPP\_Step\_5\_7\_Scaling\_Factor\_Adjustment\_6Levels\_TRNV\_CIO.sas";

%include "&folder\_6Levels.\MPP\_Step\_5\_8\_CNST1920 Adjustment\_6Levels\_TRNV.sas";

%include "&folder\_6Levels.\MPP\_Step\_5\_9\_Inflation\_Efficiency201920\_6Levels\_TRNV.sas";

%include "&folder\_6Levels.\MPP\_Step\_6\_MPP Quantums\_FinalPrice\_6Levels\_TRNV.sas";

%include "&folder\_6Levels.\MPP\_Step\_7\_Trimpoints\_6Levels\_TRNV.sas";

%include "&folder\_6Levels.\MPP\_Step\_8\_Linked Sheets\_6Levels\_FinalPrices\_TRNV.sas";

%include "&folder\_6Levels.\MPP\_Step\_9\_Model out for IA Model\_afterQR2\_6Levels\_TRNV\_cio.sas";

%include "&folder\_6Levels.\MPP\_Step\_10\_1\_Output\_Excel\_MPP\_Price\_6Levels\_TRNV\_pl.sas";

%include "&folder\_6Levels.\MPP\_Step\_10\_2\_Output\_Excel\_MPP\_PriceStepbyStep\_6Levels\_TRNV\_pl.sas";

%include "&folder\_6Levels.\Non\_Mandatory\_Step\_1\_Trimpoints\_TRNV.sas";

%include "&folder\_6Levels.\Non-Mandatory\_Step\_2\_TRNV.sas";

%include "&folder\_6Levels.\Non\_Mandatory\_Step\_3\_LinkSheet\_pl.sas";

%end;

%else %if %eval(&n\_level.)=**1** %then %do;

%let Run\_ID = 40;

%put "Run 1 levles code !!";

%end;

%else %do;

%end;

**%mend** Decide\_Levels;

%***Decide\_Levels***;

/\* to reset locations so that main excel for all prices and step by step can run error free \*/

**data** ModelOut.MPPLINKEDSHANTENATALFINALPRICE;

set OutSAS.MPPLINKEDSHANTENATALFINALPRICE;

**run**;

**data** ModelOut.MPPLINKEDSHPOSTNATALFINALPRICE;

set OutSAS.MPPLINKEDSHPOSTNATALFINALPRICE;

**run**;

**data** ModelOut.MPPLINKEDSHDELIVERYHRGFPRICES;

set OutSAS.MPPLINKEDSHDELIVERYHRGFPRICES;

**run**;

**data** ModelOut.MPP\_DELIVERY\_IANDE1920;

set OutSAS.MPP\_DELIVERY\_IANDE1920;

**run**;

**data** ModelOut.MPP\_ANTENATAL\_IANDE1920;

set OutSAS.MPP\_ANTENATAL\_IANDE1920;

**run**;

**data** ModelOut.MPP\_POSTNATAL\_IANDE1920;

set OutSAS.MPP\_POSTNATAL\_IANDE1920;

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 12 May 2017 ;

# \* OPATT\_Model\_Main\_20170512.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

libname HES\_T base "\\irnarch\sas\_data\Tariff Rebuild\SAS source data"; /\*change to Tariff, so from server 20170711 PID \*/

%let folder=\\irnarch\sas\_data\Tariff Rebuild\Models for Testing\TC\_Y1920\OPATT\SAS\_Codes;

%let outputpath=\\irnarch\sas\_data\Tariff Rebuild\Models for Testing\TC\_Y1920\OPATT\output\&username.;

%let Param\_OutlierMinValue = 0.05;

%let Param\_OutlierMaxValue = 20;

%let frontload\_cap=1.5;

%let t\_name=front\_load;

%let Option\_A\_pct=10;

%let Option\_B\_pct=20;

%let Option\_C\_pct=30;

%let Option\_D\_pct=0;

**proc** **import** datafile="\\irnarch\sas\_data\Tariff Rebuild\Models for Testing\TC\_Y1920\OPATT\20170525\_frontload\_options\_by\_tfc.xlsx"

out=frontload\_options\_bytfc

dbms=xlsx replace;

getnames=yes;

sheet=sheet1;

**run**;

**PROC** **SQL**;

CREATE table work.Reference\_Cost AS SELECT

RCID,

RC\_Year,

OrganisationCode,

OrganisationName,

CurrencyCode,

CurrencyName,

DepartmentCode,

DepartmentName,

ServiceCode,

ServiceName,

SupplierType,

Bed\_Days as BedDays,

Activity,

Mean,

Actual\_Cost as ActualCost,

Expected\_Cost as ExpectedCost,

Unit\_Cost as UnitCost,

Mapping\_pot as MappingPot

from TARIFF\_R.REFERENCE\_COST\_1617\_PUBLISHED;

**QUIT**;

**data** work.tc\_mff; set tariff\_r.tc\_mff;

WHERE YEAR="1920";

**run**;

**data** work.OPATT\_MAPPING; set TARIFF\_R.OPATT\_MAPPING; **run**;

**data** work.OPATT\_Curr\_Map; set Tariff\_r.OPATT\_Curr\_Map; **run**;

**data** work.OPATT\_Drugs\_Devices; set tariff\_r.OPATT\_Drugs\_Devices\_1920; where year="1920"

and dd\_id=**3**; **run**;

**data** work.OPATT\_DD\_Topslice; set tariff\_r.OPATT\_DD\_Topslice\_1920; where year=**1920**

and dd\_id=**3**; **run**;

**data** work.\_09nonmandhrgopatt; set Modelout.\_09nonmandhrgopatt; **run**;

**data** work.oproc\_national\_data; set Modelout.oproc\_national\_data\_1920; **run**;

**data** work.exclusions\_hrg\_oproc\_to\_opatt; set Tariff\_R.exclusions\_hrg\_oproc\_to\_opatt;

WHERE YEAR="17/18"; **run**;

**data** work.opatt\_eligibility; set Tariff\_R.opatt\_eligibility\_1920;

where year="19/20";

**run**;

**data** work.oproc\_to\_opatt\_mapping\_new; set Tariff\_R.oproc\_to\_opatt\_mapping\_1920;

where hes\_year=**1617**;

**run**;

**data** work.QUANTUMRECONCILATION\_OUTPUT; set tariff\_r.QUANTUMRECONCILATION\_OUTPUT;

where year="19/20";

**run**;

**proc** **import** datafile="\\irnarch\sas\_data\Tariff Rebuild\SAS source data\20170621\_Adult\_Paed\_mapping.xlsx"

out=further\_adjustment\_adultpaed

dbms=xlsx replace;

getnames=yes;

sheet=sheet1;

**run**;

%LET Cost\_base\_adjustment\_factor=0.000;

**data** inf\_eff; set tariff\_r.inflation\_efficiency; **run**;

**proc** **transpose** data=inf\_eff out=inf\_eff2; id year; var inflation\_and\_efficiency\_combine; **run**;

**data** inf\_eff3; set inf\_eff2; ie\_index\_adjst\_fact=(\_17\_18+**1**)\*(\_18\_19+**1**)-**1**; **run**;

**proc** **sql**;

select ie\_index\_adjst\_fact into :ie\_indextn\_adjstmt\_factors

from inf\_eff3;

**quit**;

%let ie\_indextn\_adjstmt\_factors=&ie\_indextn\_adjstmt\_factors.;

**data** opatt\_smoothing; set tariff\_r.opatt\_smoothing; format smf best22.21; **run**;

/\*Smoothing Factor \*/

**PROC** **SQL**;

SELECT Combined\_Uplifts INTO :Smoothing\_factor

from tariff\_r.smf\_v&SmoothFVersion. where subchapter="WF";

**quit**;

%let Smoothing\_factor=&Smoothing\_factor.;

%put "OPATT Model";

%put Smoothing\_factor=&Smoothing\_factor.;

%LET smoothing\_qr3\_factor=0.00;

/\*proc sql;\*/

/\*Scaling Factor \*/

**proc** **sql**;

select SF into :scaling\_factor

from tariff\_r.sf

where year="19/20";

**quit**;

%let Inflation1920=0;

%let Efficiency1920=0;

\*Get prerequisite data;

%include "&folder.\Opatt\_1\_1\_filter\_for\_scope.sas";

%include "&folder.\Opatt\_1\_3\_Total\_Costs.sas";

%include "&folder.\Opatt\_1\_4\_Regroup\_v2.sas";

%include "&folder.\Opatt\_1\_5\_Remove\_MFF.sas";

%include "&folder.\Opatt\_1\_6.sas";

%include "&folder.\Opatt\_1\_7\_National\_Averages.sas";

%include "&folder.\Opatt\_1\_8\_National\_Average\_MFF.sas";

%include "&folder.\Opatt\_1\_9\_MFF\_Rescaling.sas";

%include "&folder.\Opatt\_1\_10\_Clean\_Data.sas";

%include "&folder.\Opatt\_1\_11.sas";

%include "&folder.\Opatt\_1\_12\_Clean\_Data\_Crosstab.sas";

%include "&folder.\Opatt\_1\_13\_Recoding\_of\_501\_560.sas";

%include "&folder.\Opatt\_1\_13\_Recoding\_of\_501\_560.sas";

%include "&folder.\Opatt\_1\_14\_Providers\_per\_TFC.sas";

%include "&folder.\Opatt\_1\_15\_317\_by\_org.sas";

%include "&folder.\Opatt\_1\_16\_UC\_by\_org\_TFC\_Currency.sas";

%include "&folder.\Opatt\_1\_17.sas";

%include "&folder.\Opatt\_1\_18\_Data\_for\_IM.sas";

%include "&folder.\Opatt\_1\_19\_Remapping\_of\_Costs\_v2.sas";

%include "&folder.\Opatt\_1\_20.sas";

%include "&folder.\Opatt\_1\_21\_Crosstab\_of\_Costs.sas";

%include "&folder.\Opatt\_1\_22\_Make\_table\_of\_Costs\_Crosstab.sas";

%include "&folder.\Opatt\_1\_23\_Crosstab\_Activity.sas";

%include "&folder.\Opatt\_1\_24\_Make\_table\_of\_Activity\_Crosstab.sas";

%include "&folder.\Opatt\_1\_25\_D\_D\_Exclusions.sas";

%include "&folder.\Opatt\_1\_26.sas";

%include "&folder.\Opatt\_1\_27\_Total\_costs\_and\_pcts\_for\_apportionment\_of\_D\_D.sas";

%include "&folder.\Opatt\_1\_28\_D\_D\_costs\_to\_remove.sas";

%include "&folder.\Opatt\_1\_29\_Revised\_Costs\_following\_removal\_of\_D\_D.sas";

%include "&folder.\Opatt\_1\_30\_D\_D\_Limiting\_of\_exclusions\_inc\_additional\_topslice.sas";

%include "&folder.\Opatt\_1\_31\_Total\_Topslice\_to\_be\_removed.sas";

%include "&folder.\Opatt\_1\_32\_pct\_Topslice.sas";

%include "&folder.\Opatt\_1\_33\_Revised\_Costs\_after\_topslice.sas";

%include "&folder.\Opatt\_2\_2.sas";

%include "&folder.\Opatt\_2\_3\_v2\_1920.sas";

%include "&folder.\Opatt\_2\_4.sas";

%include "&folder.\Opatt\_2\_5\_b.sas";

%include "&folder.\Opatt\_2\_6\_b.sas";

%include "&folder.\Opatt\_2\_7\_b.sas";

%include "&folder.\Opatt\_2\_8\_Front\_loading\_i\_v2\_b.sas";

%include "&folder.\Opatt\_2\_9\_Front\_loading\_ii.sas";

%include "&folder.\Opatt\_2\_10\_Further\_adjustment\_v2\_b.sas"; /\*APPLIED EQUALISATION RULE HERE FOR 370=800 PRE QR1 \*/

%include "&folder.\Opatt\_2\_11\_YOY\_Quantum\_PRICES\_v2\_b2.sas";

%include "&folder.\Opatt\_2\_11\_YOY\_Quantum\_PRICES\_v4\_QR1\_CB\_IE\_1920.sas";

%include "&folder.\Opatt\_2\_12\_postQR1\_postMA\_4\_IA\_1920\_manual\_adj.sas";

%include "&folder.\Opatt\_2\_13\_TC\_PRICES\_step\_by\_step\_1920.sas";

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 05 July 2017 ;

# \* OPROC\_Model\_Main\_20170705.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

libname HES\_T base "\\irnarch\sas\_data\Tariff Rebuild\SAS source data"; /\* to change to server ie tariff\_R \*/

%let folder=\\irnarch\sas\_data\Tariff Rebuild\Models for Testing\TC\_Y1920\OPROC\SAS\_Codes;

%let outputpath=\\irnarch\sas\_data\Tariff Rebuild\Models for Testing\TC\_Y1920\OPROC\outputs\&username.;

**PROC** **SQL**;

CREATE table work.Reference\_Cost AS SELECT

RCID,

RC\_Year,

OrganisationCode,

OrganisationName,

CurrencyCode,

CurrencyName,

DepartmentCode,

DepartmentName,

ServiceCode,

ServiceName,

SupplierType,

Bed\_Days as BedDays,

Activity,

Mean,

Actual\_Cost as ActualCost,

Expected\_Cost as ExpectedCost,

Unit\_Cost as UnitCost,

Mapping\_pot as MappingPot

from TARIFF\_R.REFERENCE\_COST\_1617\_PUBLISHED;

**QUIT**;

**data** work.tc\_mff; set tariff\_r.tc\_mff;

WHERE YEAR="1920";

**run**;

**data** work.DI\_Group\_Mapping; set tariff\_r.DI\_Group\_Mapping\_1920;

**run**;

**data** work.DI\_OP\_Split; set tariff\_r.DI\_OP\_Split\_1920;

**RUN**;

**data** work.DI\_Distribution; set tariff\_r.DI\_Distribution\_1920;

**RUN**;

**DATA** WORK.Uplift; SET TARIFF\_R.Uplift;

where year="1920";**RUN**;

**DATA** WORK.HRGs\_zero\_price; SET TARIFF\_R.ZEROPRICEHRG;

WHERE YEAR="1920";

rename Zero\_Price\_HRGs=Zero\_HRG;

**RUN**;

**data** work.HRG\_Eligibility; set tariff\_r.HRG\_Eligibility;

where year="19/20";

**RUN**;

%include "&folder.\Oproc\_1\_1\_filter\_for\_scope.sas";

%include "&folder.\Oproc\_1\_2\_remove\_MFF.sas";

%include "&folder.\Oproc\_1\_3\_Calculates\_national\_averages.sas";

%include "&folder.\Oproc\_1\_4\_Clean\_data.sas";

%include "&folder.\Oproc\_1\_5\_National\_data\_clean\_1920.sas";

%include "&folder.\Oproc\_1\_6\_National\_average\_mff.sas";

%include "&folder.\Oproc\_1\_7\_MFF\_rescaling.sas";

%include "&folder.\Oproc\_1\_8\_Outpatient\_DI\_Costs.sas";

%include "&folder.\Oproc\_1\_9\_MFF\_from\_OPDI\_costs.sas";

%include "&folder.\Oproc\_1\_10\_Map\_DI\_HRGs2Groups.sas";

%include "&folder.\Oproc\_1\_11\_costs\_rebundled2oproc\_hrgs.sas";

%include "&folder.\Oproc\_1\_12\_Map\_DI\_group\_costs2HRGs.sas";

%include "&folder.\Oproc\_1\_13\_rebundle\_DI\_costs2HRGs\_1920.sas";

%include "&folder.\Oproc\_1\_14\_Calculate\_Prices\_1920.sas";

%include "&folder.\Oproc\_1\_15\_09NonMandHRGOPATT\_v2\_1920.sas";

%include "&folder.\Oproc\_1\_16\_Final\_tariff.sas";

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 30 March 2017 ;

# \* OM\_Main\_Run\_all\_steps\_local.sas ;

\* description: ;

\* version: 1.00 ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

%let folder=\\irnarch\SAS\_Data\Tariff Rebuild\Models for Testing\TC\_Y1920\Other\_Mandatory\SAS\_Codes;

%let outputpath=\\irnarch\SAS\_Data\Tariff Rebuild\Models for Testing\TC\_Y1920\Other\_Mandatory\Outputs\&username.;

**PROC** **SQL**;

CREATE table work.Reference\_Cost AS SELECT

RCID,

RC\_Year,

OrganisationCode,

OrganisationName,

CurrencyCode,

CurrencyName,

DepartmentCode,

DepartmentName,

ServiceCode,

ServiceName,

SupplierType,

Bed\_Days as BedDays,

Activity,

Mean,

Actual\_Cost as ActualCost,

Expected\_Cost as ExpectedCost,

Unit\_Cost as UnitCost,

Mapping\_pot as MappingPot

from TARIFF\_R.REFERENCE\_COST\_1617\_PUBLISHED;

**QUIT**;

**data** work.tc\_mff; set tariff\_r.tc\_mff;

WHERE YEAR="1920";

**run**;

**data** other\_mand\_1920; set tariff\_r.hrg\_eligibility;

where year="19/20" and other\_mandatory=**1**; **run**;

libname oth\_mand "\\irnarch\SAS\_Data\Tariff Rebuild\Models for Testing\TC\_Y1920\Model\_Outputs\_SAS\&username.";

**data** xxx(keep=spell\_hrg p6\_oproc p6\_dc p6\_el);

set oth\_mand.STEP\_BY\_STEP\_OP\_DC\_EL\_NE;

where spell\_hrg in("FE35Z", "FE34Z");

**run**;

**data** xxx\_1920;

set xxx;

if spell\_hrg="FZ54Z" then spell\_hrg="FE35Z";

else if spell\_hrg="FZ55Z" then spell\_hrg="FE34Z";

else spell\_hrg=spell\_hrg;

**run**;

**proc** **import** out=work.Other\_mandatory\_model (drop=G) /\* to get cost of reporting \*/

datafile= "\\irnarch\sas\_data\Tariff Rebuild\SAS source data\Other\_mandatory\_model\_1819.xlsx"

dbms=xlsx replace;

**run**;

**data** inf\_eff; set tariff\_r.inflation\_efficiency; **run**;

**proc** **transpose** data=inf\_eff out=inf\_eff2; id year; var inflation\_and\_efficiency\_combine; **run**;

**data** inf\_eff3; set inf\_eff2; ie\_index\_adjst\_fact=(\_17\_18+**1**)\*(\_18\_19+**1**)-**1**; **run**;

**proc** **sql**;

select ie\_index\_adjst\_fact into :ie\_indextn\_adjstmt\_factors

from inf\_eff3;

**quit**;

%let ie\_indextn\_adjstmt\_factors=&ie\_indextn\_adjstmt\_factors.;

**data** inf\_eff\_s118; set tariff\_r.inflation\_efficiency; WHERE Year="19\_20"; **run**;

**data** inf\_eff\_s118\_2; set inf\_eff\_s118;inflation\_and\_efficiency\_combine=(Inflation+**1**)\*(Efficiency+**1**)-**1**; **run**;

**proc** **sql**;

select inflation\_and\_efficiency\_combine into :Infla\_Effi\_final\_year\_uplift

from inf\_eff\_s118\_2

where ie\_id=**6**;

**quit**;

%include "&folder.\OM 1\_1 Load\_Data\_v2.sas";

%include "&folder.\OM 1\_2 Remove\_MFF\_v2.sas";

%include "&folder.\OM 1\_3 National\_Average.sas";

%include "&folder.\OM 1\_4 Data\_Cleaning.sas";

%include "&folder.\OM 1\_5 NA\_Post\_Cleaning.sas";

%include "&folder.\OM 1\_6 NA\_MFF-Adjust.sas";

%include "&folder.\OM 1\_7 Generate\_Tariff.sas";

%include "&folder.\OM\_1\_8\_rollovers\_adjustments\_final\_prices\_1920.sas";

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 01 September 2017 ;

# \* Unbundled\_Model\_Main\_20170901.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

%let folder=\\irnarch\sas\_data\Tariff Rebuild\Models for Testing\TC\_Y1920\Unbundled\SAS\_Codes;

%let outputpath=\\irnarch\sas\_data\Tariff Rebuild\Models for Testing\TC\_Y1920\Unbundled\outputs\&username.;

**PROC** **SQL**;

CREATE table work.Reference\_Cost AS SELECT

RCID,

RC\_Year,

OrganisationCode,

OrganisationName,

CurrencyCode,

CurrencyName,

DepartmentCode,

DepartmentName,

ServiceCode,

ServiceName,

SupplierType,

Bed\_Days as BedDays,

Activity,

Mean,

Actual\_Cost as ActualCost,

Expected\_Cost as ExpectedCost,

Unit\_Cost as UnitCost,

Mapping\_pot as MappingPot

from TARIFF\_R.REFERENCE\_COST\_1617\_PUBLISHED;

**QUIT**;

**data** work.tc\_mff; set tariff\_r.tc\_mff;

WHERE YEAR="1920";

**run**;

**data** work.hrg\_eligibility; set tariff\_r.hrg\_eligibility;

where year="19/20";

**run**;

**data** work.QUANTUMRECONCILATION\_OUTPUT; set tariff\_r.QUANTUMRECONCILATION\_OUTPUT;

where year="19/20";

**run**;

libname testn odbc datasrc=pbr1718\_\_28 schema=input;

**data** work.uplift; set testn.uplift; **run**;

**data** work.unbundled; set tariff\_r.unbundled\_1718; **run**;

**data** work.renal; set tariff\_r.renal\_1718; **run**;

%LET Cost\_base\_adjustment\_factor=0.000;

%LET rc\_qr1\_factor\_RenalCKD=0.00;

**data** inf\_eff; set tariff\_r.inflation\_efficiency; **run**;

**proc** **transpose** data=inf\_eff out=inf\_eff2; id year; var inflation\_and\_efficiency\_combine; **run**;

**data** inf\_eff3; set inf\_eff2; ie\_index\_adjst\_fact=(\_17\_18+**1**)\*(\_18\_19+**1**)-**1**; **run**;

**proc** **sql**;

select ie\_index\_adjst\_fact into :ie\_indextn\_adjstmt\_factors

from inf\_eff3;

**quit**;

/\*Smoothing Factors \*/

/\*SMOOTHING\_factor\_DI\*/

**PROC** **SQL**;

SELECT Combined\_Uplifts INTO :SMOOTHING\_factor\_DI

from tariff\_r.smf\_v&SmoothFVersion. where subchapter="RN";

**quit**;

/\*SMOOTHING\_factor\_Chemo\*/

**PROC** **SQL**;

SELECT Combined\_Uplifts INTO :SMOOTHING\_factor\_Chemo

from tariff\_r.smf\_v&SmoothFVersion. where subchapter="SB";

**quit**;

/\*SMOOTHING\_factor\_Radio\*/

**PROC** **SQL**;

SELECT Combined\_Uplifts INTO :SMOOTHING\_factor\_Radio

from tariff\_r.smf\_v&SmoothFVersion. where subchapter="SC";

**quit**;

/\*Do NOT do Smoothing Factor for Renal\*/

%LET SMOOTHING\_factor\_Renal=0;

/\*Scaling factor \*/

**proc** **sql**;

select SF into :scaling\_factor

from tariff\_r.sf

where year="19/20";

**quit**;

**data** inf\_eff\_s118; set tariff\_r.inflation\_efficiency; WHERE Year="19\_20"; **run**;

**data** inf\_eff\_s118\_2; set inf\_eff\_s118;inflation\_and\_efficiency\_combine=(Inflation+**1**)\*(Efficiency+**1**)-**1**; **run**;

**proc** **sql**;

select inflation\_and\_efficiency\_combine into :Infla\_Effi\_final\_year\_uplift

from inf\_eff\_s118\_2

where ie\_id=**6**;

**quit**;

%include "&folder.\Unbundled\_Radiotherapy\_Chemo\_Scope\_v0\_1\_1.sas";

%include "&folder.\Unbundled\_Radio\_Chemo\_remove\_mff\_v0\_1\_2.sas";

%include "&folder.\Unbundled\_Radio\_Chemo\_National\_average\_v0\_1\_3.sas";

%include "&folder.\Unbundled\_Radio\_Chemo\_Clean\_data\_v0\_1\_4.sas";

%include "&folder.\Unbundled\_Radio\_Chemo\_Nat\_Clean\_Data\_v0\_1\_5.sas";

%include "&folder.\Unbundled\_Radio\_Chemo\_mff\_adjusted\_Nat\_avg\_v0\_1\_6.sas";

%include "&folder.\Unbundled\_Radio\_Chemo\_mff\_rescale\_v0\_1\_7.sas";

%include "&folder.\Unbundled\_Radio\_Chemo\_output\_prices\_v0\_1\_8\_b.sas";

%include "&folder.\Unbundled\_Diagnositic\_Imaging\_Scope\_v0\_2\_1.sas";

%include "&folder.\Unbundled\_DI\_remove\_mff\_v0\_2\_2.sas";

%include "&folder.\Unbundled\_DI\_National\_average\_v0\_2\_3.sas";

%include "&folder.\Unbundled\_DI\_Clean\_data\_v0\_2\_4.sas";

%include "&folder.\Unbundled\_DI\_Nat\_Clean\_Data\_v0\_2\_5.sas";

%include "&folder.\Unbundled\_DI\_mff\_adjusted\_Nat\_avg\_v0\_2\_6.sas";

%include "&folder.\Unbundled\_DI\_mff\_rescale\_v0\_2\_7.sas";

%include "&folder.\Unbundled\_DI\_output\_prices\_v0\_2\_8\_b.sas";

%include "&folder.\Unbundled\_Calc\_QR1\_Radio\_Chem\_DI\_1920.sas";

%include "&folder.\Unbundled\_Post\_QR1\_Radio\_Chemo\_1\_1920.sas";

%include "&folder.\Unbundled\_Post\_QR1\_Radio\_Chemo\_DI\_2\_1920.sas";

%include "&folder.\Unbundled\_Post\_QR1\_MA\_4IA\_MODEL\_1920.sas";

%include "&folder.\Unbundled\_Renal\_CKD\_Scope\_V0\_3\_1\_s118.sas";

%include "&folder.\Unbundled\_Renal\_CKD\_remove\_mff\_v0\_3\_2.sas";

%include "&folder.\Unbundled\_Renal\_CKD\_National\_avg\_v0\_3\_3.sas";

%include "&folder.\Unbundled\_Renal\_CKD\_Clean\_data\_v0\_3\_4.sas";

%include "&folder.\Unbundled\_Renal\_CKD\_Nat\_Clean\_data\_v0\_3\_5.sas";

%include "&folder.\Unbundled\_Renal\_CKD\_mff\_adjusted\_Nat\_avg\_v0\_3\_6.sas";

%include "&folder.\Unbundled\_Renal\_CKD\_mff\_rescale\_V0\_3\_7.sas";

%include "&folder.\Unbundled\_Renal\_CKD\_output\_prices\_v0\_3\_8\_1920\_bpt.sas";

%include "&folder.\Unbundled\_TC\_Prices\_1920\_manual\_adjustment\_s118.sas";

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 02 January 2018 ;

# \* BPT\_00\_Group 1\_Version 1.sas ;

\* description: 1920 ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

**proc** **sql**;

create table bptinput.OPATT\_PRICES as select \* from &Opatt\_prices;

**run**;

**proc** **sql**;

create table bptinput.OPATT\_PRICES\_STEP\_BY\_STEP as select \* from &Opatt\_prices\_step\_by\_step;

**run**;

/\* Step 1. Import parameter table to SAS system

**proc** **sql**;

create table

BPTInput.BPT\_parameter\_currentyear as select \* from BPTInput.BPT\_parameter\_all where year=&BPTinputyear;

**quit**;

/\*Step 2. Retrive APC NE\_prices from 201718 linked table in APC model in SAS system\*/

**proc** **sql**;

create table apc\_prices as

select \* from &apc\_table\_ne\_price where year = &currentyear;

**quit**;

**proc** **sql**;

create table BPTInput.apc\_prices as select \* from apc\_prices;

**quit**;

/\*

Step 3. Conduct tariff calculation by join above two tables together

In this step, three columns will be produced as

-Absolute differential

-Non-BPT Tariff

-BPT Tariff

\*/

/\*

Step 4 Save the Final table into "T:\Tariff Rebuild\BPT\Version 1\BPT\_SASOutput"

The Final table should include those columns as follows:

-HRG code

-HRG name

-HRG/sub-HRG

-BPT Flag

-APC tariffprice\_NE

-%Differential\_of\_ baseprice

-Absolute differential

-Compliancerate

-Non-BPT Tariff

-BPT Tariff\*/

**proc** **sql**;

create table group1 as

select a.HRG\_code, a.Category,

a.HRG\_name,

a.HRG\_or\_SubHRG,

a.BPT\_flag,

a.Perc\_differential\_of\_base\_price,

a.Compliance\_rate,

b.ne\_prices,

round(b.ne\_prices\*a.perc\_differential\_of\_base\_price) as Abs\_Diff,

b.ne\_prices+(calculated Abs\_Diff)\*(**1**-a.Compliance\_rate) as BPT\_tariff,

(Calculated BPT\_Tariff) -(calculated Abs\_Diff) as NonBPT\_Tariff

from BPTInput.BPT\_parameter\_currentyear as a

left join apc\_prices as b

on a.HRG\_code = b.HRG\_code

where category in ('Diabetic ketoacidosis and hypoglycaemia',

'Primary Total Hip & Knee Replacements',

'Heart Failure',

'COPD Exacerbation',

'NSTEMI: Timely access to coronary angiography', 'Emergency Laparotomy'

)

order by a.HRG\_code desc

;

**quit**;

**proc** **sql**;

create table bptout.group1 as

select

Category,

HRG\_code,

HRG\_name,

HRG\_or\_SubHRG,

BPT\_flag,

ne\_prices,

Perc\_differential\_of\_base\_price,

Abs\_Diff,

Compliance\_rate,

NonBPT\_Tariff,

BPT\_tariff

from group1;

**quit**;

/\*created a table in BPToutput\*/

/\*Step 5 EXPORT the Final table into Excel file and save into "T:\Tariff Rebuild\BPT\Version 1\BPT\_ExcelOutput"\*/

**proc** **export** data=BPTOut.group1

outfile="&outputpath.\&outputfile..XLSX"

dbms=XLSX replace;

sheet="00\_group01";

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 02 January 2018 ;

# \* BPT\_00\_Group 2\_Version 1.sas ;

\* description: 1920 ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

/\* Step 2. Join above table with I\_E table which is in SAS system \*/

**proc** **sql**;

create table BPT\_previousyear as select &currentyear as year\_Infl\_Eff,

&year\_BPT\_previousyear as year\_BPT\_previousyear

,\* from BPTInput.BPT\_previousyears

;

**quit**;

**proc** **sql**;

create table BPTInput.BPT\_previousyear as select \* from BPT\_previousyear

where year\_BPT\_previousyear=&year\_BPT\_previousyear;

**quit**;

**proc** **sql**;

create table t as select a.\*, b.Inflation,b.Efficiency,b.Inflation\_and\_Efficiency\_Combine from BPTInput.BPT\_previousyear as a

left join (select \* from tariff\_r.I\_E ) as b

on a.year\_Infl\_Eff= b.year;

**quit**;

/\* Step 3. Conduct tariff calculation \*/

**proc** **sql**;

create table group2 as select \*, round(t.BPT\*(**1**+Inflation)\*(**1**+Efficiency)) as BPT\_current\_year from t

where HRG\_code='n/a';

**quit**;

/\*Step 4. Save the final table into SASOutput folder\*/

**proc** **sql**;

create table BPTout.group2 as select HRG\_code, HRG\_name, BPT as BPT\_last\_year, Inflation, Efficiency, BPT\_Current\_year from group2

where HRG\_code='n/a';

**quit**;

/\* Step 5. Export the final table into ExcelOutput folder \*/

**proc** **export** data=BPTout.group2

outfile="&outputpath.\&outputfile..XLSX"

dbms=XLSX replace;

sheet="00\_group02";

**run**;

\* author: PID ;

\* date: 02 January 2018 ;

# \* BPT\_01\_Stroke.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

/\*1. Importing BPT parameter table\*/

**proc** **sql**;

create table BPT\_parameter\_v2

as select \* from BPTInput.BPT\_PARAMETER\_CURRENTYEAR;

**quit**;

/\*2. Selecting only Stroke HRGs\*/

**proc** **sql**;

create table stroke1 as select \*, Cost\_Imaging+Cost\_stroke as Tot\_Cost\_compliance, (calculated Tot\_Cost\_compliance)\***2** as TotBPT\_diff from BPT\_parameter\_v2

where UPcase(substr(HRG\_name,**1**,**6**))='STROKE';

**quit**;

/\*3. Tariff Calculations\*/

**proc** **sql**;

create table stroke2 as select a.\*, b.NE\_prices,

round(b.NE\_prices-(Tot\_Cost\_compliance\*compliance\_rate)) as Tariff\_no\_cost\_compl,

(totBPT\_diff/(calculated Tariff\_no\_cost\_compl)) as per\_differential\_of\_base\_price

from stroke1 as a

left join apc\_prices as b on a.HRG\_code=b.HRG\_code;

**quit**;

**data** stroke3;

set stroke2 (drop=Perc\_Differential\_of\_base\_price);

format per\_differential\_of\_base\_price **8.2**;

**run**;

/\*4. Created table into BPT library\*/

**proc** **sql**;

create table stroke as select a.\*, Tariff\_no\_cost\_compl\*per\_differential\_of\_base\_price as Abs\_differencial,

Tariff\_no\_cost\_compl+calculated Abs\_differencial\*(**1**-Compliance\_Rate) as BPT\_no\_cost\_compl,

(calculated BPT\_no\_cost\_compl)+tot\_cost\_compliance as BPT\_with\_cost\_compl,

(calculated BPT\_with\_cost\_compl) - (calculated Abs\_differencial) -tot\_cost\_compliance as Quantum\_neutral\_nonBPT,

(calculated BPT\_with\_cost\_compl) - (calculated Quantum\_neutral\_nonBPT) as Diff\_BPT\_nonBPT,

(calculated BPT\_with\_cost\_compl)\*Compliance\_Rate+(**1**-Compliance\_Rate)\*(calculated Quantum\_neutral\_nonBPT) as checkPrices

from stroke3 as a

;

**quit**;

/\*5. Export Excel output\*/

**proc** **sql**;

create table BPTout.stroke as

select

HRG\_code,

HRG\_name,

HRG\_or\_SubHRG,

BPT\_Flag,

NE\_Prices,

Tot\_Cost\_compliance,

Compliance\_rate,

Tariff\_no\_cost\_compl,

per\_differential\_of\_base\_price,

Abs\_differencial,

BPT\_no\_cost\_compl,

BPT\_with\_cost\_compl,

Quantum\_neutral\_nonBPT,

Diff\_BPT\_nonBPT,

checkPrices

from stroke;

**quit**;

**proc** **export** data=BPTout.stroke

outfile="&outputpath.\&outputfile..XLSX"

dbms=XLSX replace;

sheet="01\_Stroke";

**run**;

/\*1. BPT unbundled table\*/

**proc** **sql**;

create table Unbundled\_BPT

as select \*,tariff as Price, Total\_Activity as Activity from &inputtableUnbundledBPT

;

**quit**;

**proc** **sql**;

create table BPTout.renal\_Unbundled\_Extract as select \*,

Price\*Activity as Total\_quantum from Unbundled\_BPT;

**quit**;

/\*Calculating TotalQuantum Unbundled\*/

**Proc** **Sql** noprint;

Select sum(Price\*Activity) into :Total\_quantum\_Unbundled

from BPTout.renal\_Unbundled\_Extract;

**Quit**;

%Put &Total\_quantum\_Unbundled;

/\*Creating table with combined HRGs\*/

**proc** **sql**;

create table BPTout.renal\_CombinedHRG as

select 'LD01A & LD05A and LD02A & LD06A' as HRG\_code,

sum(case when HRG\_code in ('LD01A','LD05A','LD02A','LD06A') then Total\_quantum else **0** end) as Total\_quantum,

sum(case when HRG\_code in ('LD01A','LD05A','LD02A','LD06A') then (Activity) else **0** end) as Total\_activity,

(Calculated Total\_quantum)/(calculated total\_activity) as Combined\_prices,

(calculated Total\_activity)\*&Access\_Rate as BPTactivity\_ComplAchieved,

(calculated Total\_activity)\*(**1**-&Access\_Rate) as NoBPTactivity\_ComplAchieved,

(Calculated Total\_quantum)/((calculated BPTactivity\_ComplAchieved)+(calculated NoBPTactivity\_ComplAchieved\*&Price\_Ratio\_NoBPT\_BPT )) as ProposedFistulaPrice,

(calculated ProposedFistulaPrice)\*&Price\_Ratio\_NoBPT\_BPT as ProposedCatheterPrice,

(calculated BPTactivity\_ComplAchieved)\*(calculated ProposedFistulaPrice) + (calculated NoBPTactivity\_ComplAchieved)\*(calculated ProposedCatheterPrice) as Total\_Cost,

((calculated Total\_cost)/(calculated Total\_quantum))-**1** as ExpectedChangeTotalSpend,

sum(case when HRG\_code in ('LD02A','LD06A') then (Activity) else **0** end) as ActivityFistula,

sum(case when HRG\_code in ('LD01A','LD05A') then (Activity) else **0** end) as ActivityCatheter,

((calculated ActivityFistula)+(calculated ActivityCatheter)) - (calculated Total\_activity) as CheckIfActivityDiff,

(calculated ActivityFistula)/((calculated ActivityFistula)+(calculated ActivityCatheter)) as ActualCompliance

from BPTout.renal\_Unbundled\_Extract as a

union

select 'LD03A & LD07A and LD04A & LD08A' as HRG\_code,

sum(case when HRG\_code in ('LD03A', 'LD07A', 'LD04A','LD08A') then Total\_quantum else **0** end) as Total\_quantum,

sum(case when HRG\_code in ('LD03A', 'LD07A', 'LD04A','LD08A') then Activity else **0** end) as Total\_activity,

(Calculated Total\_quantum)/(calculated total\_activity) as Combined\_prices,

(calculated Total\_activity)\*&Access\_Rate as BPTactivity\_ComplAchieved,

(calculated Total\_activity)\*(**1**-&Access\_Rate) as NoBPTactivity\_ComplAchieved,

(Calculated Total\_quantum)/((calculated BPTactivity\_ComplAchieved)+(calculated NoBPTactivity\_ComplAchieved\*&Price\_Ratio\_NoBPT\_BPT )) as ProposedFistulaPrice,

(calculated ProposedFistulaPrice)\*&Price\_Ratio\_NoBPT\_BPT as ProposedCatheterPrice,

(calculated BPTactivity\_ComplAchieved)\*(calculated ProposedFistulaPrice) + (calculated NoBPTactivity\_ComplAchieved)\*(calculated ProposedCatheterPrice) as Total\_Cost,

((calculated Total\_cost)/(calculated Total\_quantum))-**1** as ExpectedChangeTotalSpend,

sum(case when HRG\_code in ('LD04A','LD08A') then (Activity) else **0** end) as ActivityFistula,

sum(case when HRG\_code in ('LD03A','LD07A') then (Activity) else **0** end) as ActivityCatheter,

((calculated ActivityFistula)+(calculated ActivityCatheter)) - (calculated Total\_activity) as CheckIfActivityDiff,

(calculated ActivityFistula)/((calculated ActivityFistula)+(calculated ActivityCatheter)) as ActualCompliance

from BPTout.renal\_Unbundled\_Extract as b;

**quit**;

/\*Create macro variable for LD01A & LD05A and LD02A & LD06A\*/

**Proc** **Sql** noprint;

Select

ProposedFistulaPrice, ProposedCatheterPrice into :ProposedFistulaPrice\_var01050206,

:ProposedCatPrice\_var01050206

from BPTout.renal\_CombinedHRG

where HRG\_code= 'LD01A & LD05A and LD02A & LD06A';

**Quit**;

%Put &ProposedFistulaPrice\_var01050206;

%Put &ProposedCatPrice\_var01050206;

/\*Create macro variable for LD03A & LD07A and LD04A & LD08A\*/

**Proc** **Sql** noprint;

Select

ProposedFistulaPrice, ProposedCatheterPrice into :ProposedFistulaPrice\_var03070408,

:ProposedCatPrice\_var03070408

from BPTout.renal\_CombinedHRG

where HRG\_code= 'LD03A & LD07A and LD04A & LD08A';

**Quit**;

%Put &ProposedFistulaPrice\_var03070408;

%Put &ProposedCatPrice\_var03070408;

/\*Create macro variable for LD09A, LD10A, LD11A,LD12A, LD13A \*/

**Proc** **Sql** noprint;

Select

price into :price\_var09

from BPTout.renal\_Unbundled\_Extract

where HRG\_code in ( 'LD09A');

**Quit**;

%Put &price\_var09;

**Proc** **Sql** noprint;

Select

price into :price\_var10

from BPTout.renal\_Unbundled\_Extract

where HRG\_code in ( 'LD10A');

**Quit**;

%Put &price\_var10;

**Proc** **Sql** noprint;

Select

price into :price\_var11

from BPTout.renal\_Unbundled\_Extract

where HRG\_code in ( 'LD11A');

**Quit**;

%Put &price\_var11;

**Proc** **Sql** noprint;

Select

price into :price\_var12

from BPTout.renal\_Unbundled\_Extract

where HRG\_code in ( 'LD12A');

**Quit**;

%Put &price\_va12;

**Proc** **Sql** noprint;

Select

price into :price\_var13

from BPTout.renal\_Unbundled\_Extract

where HRG\_code in ( 'LD13A');

**Quit**;

%Put &price\_va13;

**proc** **sql**;

create table RenalStepbyStep as

select

case when hrg\_code in ('LD01A','LD02A') then **1**

when hrg\_code in ('LD03A','LD04A') then **2**

when hrg\_code in ('LD05A','LD06A') then **3**

when hrg\_code in ('LD07A','LD08A') then **4**

when hrg\_code in ('LD09A','LD10A') then **5**

else **.**

end as IDhrg,

hrg\_code,

case when hrg\_name like '%Catheter%' then 'Catheter'

when hrg\_name like '%Fistula%' then 'Fistula'

else 'NULL' end as Access,

case when hrg\_name like '%Home %' then 'Home'

when hrg\_name like '%Peritoneal%' then 'Peritoneal'

else 'NULL' end as Setting,

case

when hrg\_code in ('LD01A','LD05A','LD02A','LD06A') and calculated access='Fistula' then &ProposedFistulaPrice\_var01050206

when hrg\_code in ('LD01A','LD05A','LD02A','LD06A') and calculated access='Catheter' then &ProposedCatPrice\_var01050206

when hrg\_code in ('LD03A','LD07A','LD04A','LD08A') and calculated access='Fistula' then &ProposedFistulaPrice\_var03070408

when hrg\_code in ('LD03A','LD07A','LD04A','LD08A') and calculated access='Catheter' then &ProposedCatPrice\_var03070408

when hrg\_code= 'LD09A' then &price\_var09

when hrg\_code= 'LD10A' then &price\_var10

when hrg\_code= 'LD11A' then &price\_var11

when hrg\_code= 'LD12A' then &price\_var12

when hrg\_code= 'LD13A' then &price\_var13

end as Calculation\_Revised\_Spend,

case when hrg\_code in ('LD09A','LD10A') then &ProposedFistulaPrice\_var01050206 else calculated Calculation\_Revised\_Spend end as AdjustLD09ALD10A\_matchLD02A,

total\_activity as Activity\_RefCost

from bptout.renal\_Unbundled\_Extract

;

**quit**;

**proc** **sql**;

create table RenalSummaryActivity

as

select idhrg, sum(activity\_refcost) as activity\_rc

from RenalStepbyStep

group by

idhrg

;

**quit**;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

**proc** **sql**;

create table BPTout.renal\_RenalStepbyStep as select a.\*,

case when Access='Fistula' then (&Access\_Rate)\*b.activity\_rc

when Access='Catheter' then (**1**-&Access\_Rate)\*b.activity\_rc

when a.idhrg=**.** then Activity\_RefCost

end as ActivityBPTprojected,

AdjustLD09ALD10A\_matchLD02A\*(calculated ActivityBPTprojected) as InitialQuantum ,

case when setting ='Home' then **3** else **1** end as HomeTreatMultiplier,

(calculated ActivityBPTprojected)/(calculated HomeTreatMultiplier) as AdjustedActivity\_duetoMultiplier,

AdjustLD09ALD10A\_matchLD02A\*(calculated HomeTreatMultiplier) as Priceaftermultiplier,

c.Price as MAdjPrice,

case when MAdjPrice=**.** or a.hrg\_code in ('LD11A', 'LD12A') then (calculated Priceaftermultiplier) else MAdjPrice end as FinalPrice\_Aftermadj

from RenalStepbyStep as a

left join RenalSummaryActivity as b on a.idhrg=b.idhrg

left join BPTInput.BPT\_ManualAdjustment as c on a.hrg\_code=c.hrg\_code

where c.Category='Renal';

/\*order by hrg\_code;\*/

**quit**;

/\*Create Quantum Reconciliation Table\*/

**proc** **sql**;

create table BPTout.renal\_QuantumReconciliation as

select

sum(InitialQuantum) as TotalInitQuantum\_ToReconcile,

sum(AdjustedActivity\_duetoMultiplier\*Priceaftermultiplier) as TotalQuantum\_FinalPrice

from BPTout.Renal\_unbundled\_extract as a

left join BPTout.renal\_RenalStepbyStep as b on a.HRG\_code=b.HRG\_code

;

**quit**;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/\*IMPLEMENTING THE MANUAL ADJUSTMENT\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

**Proc** **Sql** noprint;

Select

TotalInitQuantum\_ToReconcile-TotalQuantum\_FinalPrice as checkquantum

into :checkquantum

from BPTout.renal\_QuantumReconciliation

;

**Quit**;

%Put &checkquantum;

**proc** **sql**;

create table BPTout.renal\_RenalStepbyStep as

select \*,

(AdjustLD09ALD10A\_matchLD02A\*HomeTreatMultiplier)\*(**1**+&checkquantum) as FinalPricesPriorMAdj,

MAdjPrice,

FinalPrice\_Aftermadj

from BPTout.renal\_RenalStepbyStep ;

**quit**;

/\*Calculating TotalQuantum after weeekly treatment multiplier\*/

/\*\*/

**Proc** **Sql** noprint;

Select sum(Priceaftermultiplier\*AdjustedActivity\_duetoMultiplier) into :totquantumPriceaftermultiplier

from BPTout.renal\_RenalStepbyStep

Quit;

%Put &totquantumPriceaftermultiplier;

/\*Calculating TotalQuantum after manual adjustment\*/

**Proc** **Sql** noprint;

Select sum(FinalPrice\_Aftermadj\*AdjustedActivity\_duetoMultiplier) into :totquantumPriceaftermadj

from BPTout.renal\_RenalStepbyStep;

**Quit**;

%Put &totquantumPriceaftermadj;

/\*Calculating difference to target quantum\*/

%let Difference\_targetquantum = (&Total\_quantum\_Unbundled/&totquantumPriceaftermadj)-1;

**proc** **sql**;

create table test as select &Difference\_targetquantum

from BPTout.renal\_RenalStepbyStep;

**quit**;

/\*Create final renal table step by step, the final 2 columns are based on the value of reconcileTargetQuantum (this is set up in the general settings)\*/

**proc** **sql**;

create table BPTout.renal\_RenalStepbyStep as

select \*,

case when &reconcileTargetQuantum=**1** then (FinalPrice\_Aftermadj\*(**1**+&Difference\_targetquantum))

when &reconcileTargetQuantum=**0** then FinalPrice\_Aftermadj end as FinalPricAfterMadjSmooth,

case when &reconcileTargetQuantum=**1** then (FinalPrice\_Aftermadj\*(**1**+&Difference\_targetquantum))

when &reconcileTargetQuantum=**0** then FinalPrice\_Aftermadj end as PriceAfterMAj\_NoSmooth

from BPTout.renal\_RenalStepbyStep;

**quit**;

/\*6. Create a Renal table with a structure that can be used for IA loop \*/

**proc** **sql**;

create table t as select hrg\_code, hrg\_name, case when hrg\_code in ('LD09A', 'LD10A') then 'weekly' else Session end as Session from

BPTout.renal\_Unbundled\_Extract;

;

**quit**;

**proc** **sql**;

create table s as select \* from bptout.renal\_renalstepbystep;

**quit**;

**proc** **sql**;

create table bptout.Renal\_IAmodel as select **4** as Renal\_price\_id, &currentyear as year, s.hrg\_code, t.hrg\_name,

s.PriceAfterMAj\_NoSmooth as Tariff,

t.session as Session

from s

left join t on s.hrg\_code=t.hrg\_code

;

**quit**;

/\*5. Export Excel output\*/

**proc** **export** data=BPTout.renal\_Unbundled\_Extract

outfile="&outputpath.\&outputfile..XLSX"

dbms=XLSX replace;

sheet="02a\_renal\_Unbundled\_Extract";

**run**;

**proc** **export** data=BPTout.renal\_CombinedHRG

outfile="&outputpath.\&outputfile..XLSX"

dbms=XLSX replace;

sheet="02b\_renal\_CombinedHRG";

**run**;

**proc** **export** data=BPTout.renal\_RenalStepbyStep

outfile="&outputpath.\&outputfile..XLSX"

dbms=XLSX replace;

sheet="02c\_renal\_RenalStepbyStep";

**run**;

**proc** **export** data=BPTout.renal\_QuantumReconciliation

outfile="&outputpath.\&outputfile..XLSX"

dbms=XLSX replace;

sheet="Check\_renal\_QuantumReconc";

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* author: PID ;

\* date: 02 January 2018 ;

# \* BPT\_03\_Daycase.sas ;

\* description: 1920 ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

**proc** **sql**;

create table DayCase1 as

select

a.HRG\_code,

a.HRG\_name,

a.Perc\_Differential\_of\_base\_price,

a.Compliance\_rate,

a.BPT\_flag,

b.Combined\_DC\_EL\_Prices,

(b.Combined\_DC\_EL\_Prices\*a.Perc\_Differential\_of\_base\_price) as Differential,

(b.Combined\_DC\_EL\_Prices)+((calculated Differential)\*(**1**-a.Compliance\_rate)) as BPTprice\_DC,

(calculated BPTprice\_DC)-(calculated Differential) as nonBPTprice\_EL,

case when a.BPT\_flag<>'n/a' then **.** else c.P1\_DC\_TA end as Activity\_DC,

case when a.BPT\_flag<>'n/a' then **.** else c.P1\_EL\_TA end as Activity\_EL

from BPTInput.BPT\_PARAMETER\_CURRENTYear as a

left join BPTInput.apc\_prices as b on a.hrg\_code=b.hrg\_code

left join &ActivityTable as c on a.hrg\_code=c.hrg

where Category='Daycase';

**quit**;

**proc** **sql**;

create table DayCase2 as

select

a.HRG\_code,

sum(case when a.BPT\_flag<>'n/a' and d.ADM\_SPLIT = 'DC' then d.Spell\_flag\_Cnt else **.** end ) as Flagged\_activity\_DC,

sum(case when a.BPT\_flag<>'n/a' and d.ADM\_SPLIT = 'EL' then d.Spell\_flag\_Cnt else **.** end) as Flagged\_activity\_EL

from BPTInput.BPT\_PARAMETER\_CURRENTYear as a

left join BPTOut.ActivitySpell\_Flags as d on a.hrg\_code=d.RC\_SpellHRG\_Rev and a.BPT\_FLAG=d.LP\_BPT

where Category='Daycase'

group by a.HRG\_code;

**quit**;

**proc** **sql**;

create table BPTout.DayCase

as select a.\*,

b.Flagged\_activity\_DC,

b.Flagged\_activity\_EL,

case when Activity\_DC =**.** then Flagged\_activity\_DC/(Flagged\_activity\_DC+Flagged\_activity\_EL)

when b.Flagged\_activity\_DC=**.** then Activity\_DC/(Activity\_DC+Activity\_EL) end as Actual\_compliance

from DayCase1 as a

left join DayCase2 as b on a.HRG\_code=b.HRG\_code;

**quit**;

**proc** **export** data=BPTout.DayCase

outfile="&outputpath.\&outputfile..XLSX"

dbms=XLSX replace;

sheet="03\_DayCase";

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* author: PID ;

\* date: 02 January 2018 ;

# \* BPT\_04\_Endoscopy.sas ;

\* description: 1920 ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

**proc** **sql**;

create table endoscopy as

select a.HRG\_code, a.HRG\_name, HRG\_or\_SubHRG,a.BPT\_flag, b.Combined\_DC\_EL\_Prices,a.Level3\_Endoscopy\_NonBPT\_PercFull,

b.Combined\_DC\_EL\_Prices\*a.Level3\_Endoscopy\_NonBPT\_PercFull as Abs\_Differenc\_L3,

b.Combined\_DC\_EL\_Prices\*a.Level2\_Endoscopy\_IntermediateBPT as Abs\_Differenc\_L2,

b.Combined\_DC\_EL\_Prices+((**1**-Compliance\_RateTier1-Compliance\_RateTier2)\*(Calculated Abs\_Differenc\_L3))+(Compliance\_RateTier2\*(Calculated Abs\_Differenc\_L2)) as BPT\_L1,

(calculated BPT\_L1)-(calculated Abs\_Differenc\_L3) as nonBPT\_Tariff\_L3,

(calculated BPT\_L1) - (calculated Abs\_Differenc\_L2) as L2\_IntBPT,

b.\*,a.\*

from BPTInput.BPT\_parameter\_currentyear as a

left join BPTInput.APC\_Prices as b on a.hrg\_code=b.hrg\_code

where a.Category='Endoscopy'

;

**proc** **sql**;

create table BPTout.endoscopy as select

HRG\_code,

HRG\_name,

HRG\_or\_SubHRG,

BPT\_flag,

Combined\_DC\_EL\_Prices,

Level3\_Endoscopy\_NonBPT\_PercFull,

Abs\_Differenc\_L3,

nonBPT\_Tariff\_L3,

Level2\_Endoscopy\_IntermediateBPT,

Abs\_Differenc\_L2,

L2\_IntBPT,

BPT\_L1,

(**1**-Compliance\_RateTier1-Compliance\_RateTier2) as Compliance\_RateTier3,

Compliance\_RateTier2,

Compliance\_RateTier1

from endoscopy;

**quit**;

**proc** **export** data=BPTout.endoscopy

outfile="&outputpath.\&outputfile..XLSX"

dbms=XLSX replace;

sheet="04\_Endoscopy";

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* author: PID ;

\* date: 11 January 2018 ;

# \* BPT\_05\_FHF.sas ;

\* description: 1920 ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

**proc** **sql**;

create table BPTout.FHF as

select a.HRG\_code, a.HRG\_name, HRG\_or\_SubHRG,a.BPT\_flag, b.NE\_Prices,a.Cost\_Compliance, a.Compliance\_Rate,

(b.NE\_Prices-(a.Cost\_Compliance\*a.Compliance\_Rate)) as TariffPrice\_ExclComplCost,

(**2**\*a.Cost\_Compliance)/(calculated TariffPrice\_ExclComplCost) as Perc\_Differential\_of\_base\_price,

(calculated Perc\_Differential\_of\_base\_price)\*(calculated TariffPrice\_ExclComplCost) as AbsDifferential,

(calculated TariffPrice\_ExclComplCost)+(calculated AbsDifferential)\*(**1**-a.Compliance\_Rate) as BPTTariff\_ExclComplCost,

(calculated BPTTariff\_ExclComplCost)+a.Cost\_Compliance as BPTprice\_InclComplCost,

((b.NE\_Prices)- (calculated BPTprice\_InclComplCost)\*a.Compliance\_Rate)/(**1**- a.Compliance\_Rate) as QuantumNeutralNonBPTprice

from BPTInput.BPT\_parameter\_currentyear as a

left join BPTInput.APC\_Prices as b on a.hrg\_code=b.hrg\_code

where a.Category='FHF'

;

**proc** **export** data=BPTout.FHF

outfile="&outputpath.\&outputfile..XLSX"

dbms=XLSX replace;

sheet="05\_Fhf";

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* author: PID ;

\* date: 11 January 2018 ;

# \* BPT\_06\_Outpatients.sas ;

\* description: 1920 ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

**proc** **sql**;

create table BPTout.Outpatients as

select a.HRG\_code, a.HRG\_name, HRG\_or\_SubHRG,a.BPT\_flag, b.OPROC\_Tariff, b.Combined\_DC\_EL\_Prices,

c.P1\_OPROC\_TA as APC\_ActivityBaseOPROC,

c.P1\_DC\_TA+ c.P1\_EL\_TA as APC\_ActivityBaseDCEL,

(b.OPROC\_Tariff\*c.P1\_OPROC\_TA)+ (b.Combined\_DC\_EL\_Prices\*(calculated APC\_ActivityBaseDCEL)) as TotalQuantum,

(calculated TotalQuantum)/(APC\_ActivityBaseOPROC+(calculated APC\_ActivityBaseDCEL)) as CombinedPrice,

(calculated CombinedPrice)\*(Perc\_Differential\_of\_base\_price) as Abs\_Differential,

(calculated CombinedPrice)+(calculated Abs\_Differential)\*(**1**-a.Compliance\_Rate) as BPTprice\_OPROC,

a.Compliance\_Rate,

(calculated BPTprice\_OPROC) - (calculated Abs\_Differential) as nonBPT\_minusAbsDiff,

(c.P1\_OPROC\_TA)/((c.P1\_OPROC\_TA+calculated APC\_ActivityBaseDCEL)) as Actual\_compliance

from BPTInput.BPT\_parameter\_currentyear as a

left join BPTInput.APC\_Prices as b on a.hrg\_code=b.hrg\_code

left join &ActivityTable as c on a.hrg\_code=c.hrg

where a.Category='Outpatients'

;

**proc** **export** data=BPTout.Outpatients

outfile="&outputpath.\&outputfile..XLSX"

dbms=XLSX replace;

sheet="06\_Outpatients";

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* author: PID ;

\* date: 19 January 2018 ;

# \* BPT\_07\_Paed\_Ep.sas ;

\* description: 1920 ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

/\*Manual Adjustment TIA \*/

**Proc** **Sql** noprint;

Select price into :ManualAdjPricePaedEp

from BPTInput.BPT\_ManualAdjustment

where Category='Paed\_Ep';

**Quit**;

%Put &ManualAdjPricePaedEp;

/\*Create macro variable \*/

**Proc** **Sql** noprint;

Select

Compliance\_rate format=**8.2** into :Compliance\_Rate\_WF01A

from BPTInput.BPT\_Parameter\_currentyear

where HRG\_code='WF01A';

**Quit**;

%Put &Compliance\_Rate\_WF01A;

/\*Create macro variable \*/

**Proc** **Sql** noprint;

Select

Compliance\_rate format=**8.2** into :Compliance\_Rate\_WF02A

from BPTInput.BPT\_Parameter\_currentyear

where HRG\_code='WF02A';

**Quit**;

%Put &Compliance\_Rate\_WF02A ;

**proc** **sql**;

create table paed\_ep

as select

a.TFC,

a.Tariff\_TFC as Tariff\_TFC,

a.FAs\_prices\_&currentyear4digits. as OPATTprices\_WF01B,

a.FAm\_prices\_&currentyear4digits. as OPATTprices\_WF02B,

a.FUs\_prices\_&currentyear4digits. as OPATTprices\_WF01A,

a.FUm\_prices\_&currentyear4digits. as OPATTprices\_WF02A,

case when a.TFC =**420** then **.** else b.CL\_FAS\_act\_PostA end as OPATTactivity\_WF01B,

case when a.TFC =**420** then **.** else b.CL\_FAM\_act\_PostA end as OPATTactivity\_WF02B,

case when a.TFC =**420** then **.** else b.CL\_FUS\_act\_PostA end as OPATTactivity\_WF01A,

case when a.TFC =**420** then **.** else b.CL\_FUM\_act\_PostA end as OPATTactivity\_WF02A,

case when a.TFC =**420** then **.** else a.FUs\_prices\_&currentyear4digits.\*b.CL\_FUS\_act\_PostA end AS Total\_quantum\_233\_WF01A,

case when a.TFC =**420** then **.** else a.FUm\_prices\_&currentyear4digits.\*b.CL\_FUM\_act\_PostA end AS Total\_quantum\_233\_WF02A,

(calculated OPATTactivity\_WF01A)\*&Compliance\_Rate\_WF01A as Activity\_BPT\_233\_WF01A,

(calculated OPATTactivity\_WF02A)\*&Compliance\_Rate\_WF01A as Activity\_BPT\_233\_WF02A,

(**1**-&Compliance\_Rate\_WF01A)\*(a.FUs\_prices\_&currentyear4digits.)\*(calculated OPATTactivity\_WF01A) as QuantumnonBPT\_WF01A,

(**1**-&Compliance\_Rate\_WF01A)\*(a.FUm\_prices\_&currentyear4digits.)\*(calculated OPATTactivity\_WF02A) as QuantumnonBPT\_WF02A

from bptinput.opatt\_prices as a

left join bptinput.opatt\_prices\_step\_by\_step as b on a.TFC=b.TFC

where a.tfc in (**223**,**420**);

**quit**;

/\*Create macro variable \*/

**Proc** **Sql** noprint;

Select

OPATTprices\_WF01A into :nonBPT\_WF01A

from paed\_ep

where TFC=**420**;

**Quit**;

%Put &nonBPT\_WF01A;

/\*Create macro variable \*/

**Proc** **Sql** noprint;

Select

OPATTprices\_WF02A into :nonBPT\_WF02A

from paed\_ep

where TFC=**420**;

**Quit**;

%Put &nonBPT\_WF02A;

/\*Create macro variable \*/

**Proc** **Sql** noprint;

Select

(Total\_quantum\_233\_WF01A+Total\_quantum\_233\_WF02A) /(Activity\_BPT\_233\_WF01A+Activity\_BPT\_233\_WF02A) as Ratio into :RatioTotQuantumTotAct

from paed\_ep

where TFC=**223**;

**Quit**;

%Put &RatioTotQuantumTotAct;

**proc** **sql**;

create table BPTOut.PaedEp as select \*,

case when TFC= **223** then &nonBPT\_WF01A end as nonBPT\_WF01A,

case when TFC= **223** then &nonBPT\_WF02A end as nonBPT\_WF02A,

case when TFC= **223** then &ManualAdjPricePaedEp/&RatioTotQuantumTotAct end as BPTManAdj\_WF01A,

case when TFC= **223** then &ManualAdjPricePaedEp/&RatioTotQuantumTotAct end as BPTManAdj\_WF02A,

&RatioTotQuantumTotAct\*(calculated BPTManAdj\_WF01A) as BPT\_WF01A,

&RatioTotQuantumTotAct\*(calculated BPTManAdj\_WF02A) as BPT\_WF02A

from paed\_ep

;

**quit**;

**proc** **export** data=BPTOut.PaedEp

outfile="&outputpath.\&outputfile..XLSX"

dbms=XLSX replace;

sheet="07\_Paedep";

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* author: PID ;

\* date: 11 January 2018 ;

# \* BPT\_08\_Pleural\_Effusion.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

**proc** **sql** number outobs=**10**;

create table PlEff as

select a.HRG\_code, a.HRG\_name from BPTInput.BPT\_parameter\_currentyear as a where Category ='Pleural Effusion'

union

select 'DZ16N' as HRG\_code, 'Pleural Effusion with Single Intervention, with CC Score 0-5' as HRG\_name from BPTInput.BPT\_parameter\_currentyear as b

order by HRG\_code

;

**quit**;

**data** PlEff\_v2;

ID=\_n\_;

set PlEff;

**run**;

**proc** **sql**;

create table PlEff\_v3 as select a.\*,

case when ID=**1** then 'SSEM'

when ID=**2** then 'NE'

when ID=**3** then 'DC/EL'

when ID=**4** then 'DC/EL' end

as Setting, b.Compliance\_Rate,

case when a.HRG\_code = 'DZ16N' then **.** else c.P1\_DC\_TA end as SpellDC ,

case when a.HRG\_code = 'DZ16N' then **.** else c.P1\_EL\_TA end as SpellEL,

case when (Calculated Setting) = 'SSEM' then **0** else d.SSEM\_pct end as SSEM\_ptc,

case when a.HRG\_Code in ('YD05Z','YD04Z') then d.Combined\_DC\_EL\_Prices

when a.HRG\_Code ='DZ16N' and (calculated Setting)='NE' then d.NE\_Prices

when a.HRG\_Code ='DZ16N' and (calculated Setting)='SSEM' then (case when a.HRG\_Code ='DZ16N' and (calculated Setting)='NE' then d.NE\_Prices end)\*(calculated SSEM\_ptc) end

as APC\_basePrice,

case when (calculated Setting) ='DC/EL' then (calculated SpellDC)+(calculated SpellEL)

when (calculated Setting) ='NE' then c.P1\_NE\_AT\_less\_SSEM

when (calculated Setting) ='SSEM' then c.P1\_SSEM end as APC\_baseActivity,

(calculated APC\_baseActivity)\*(calculated APC\_basePrice) as Total\_quantum\_actual

from PlEff\_v2 as a

left join Bptinput.BPT\_parameter\_currentyear as b on a.HRG\_code=b.HRG\_code

left join &ActivityTable as c on a.HRG\_code=c.HRG

left join &apc\_table\_ne\_price as d on a.HRG\_code=d.HRG\_code

;

**quit**;

**proc** **sql**;

create table t as select sum(case when Setting <> 'DC/EL' then APC\_baseActivity\*Compliance\_Rate else **0** end) as SumProdNESSEM,

sum(case when Setting = 'DC/EL' then APC\_baseActivity else **0** end) as ActDCEL

from PlEff\_v3;

**quit**;

/\*Create macro variable \*/

**Proc** **Sql** noprint;

Select

SumProdNESSEM into :SumProd\_ActCompl\_NE\_SSEM

from t

Quit;

%Put &SumProd\_ActCompl\_NE\_SSEM;

**Proc** **Sql** noprint;

Select

ActDCEL into :ActDCEL

from t

Quit;

%Put &ActDCEL;

**proc** **sql**;

create table PlEff\_v4 as

select \*,

case when Setting ='DC/EL' then Compliance\_Rate\*APC\_baseActivity +(&SumProd\_ActCompl\_NE\_SSEM\*APC\_baseActivity)/(&ActDCEL)

when Setting <> 'DC/EL' then (**1**-Compliance\_Rate)\*(APC\_baseActivity) end as Target\_activity,

APC\_baseprice\*(calculated Target\_activity) as TotalQuantum\_withCompl

from PlEff\_v3;

**quit**;

**proc** **sql**;

create table intermidiatetable as select sum(TotalQuantum\_withCompl)/sum(Target\_activity) as ImpliedSinglePrice

from PlEff\_v4;

**quit**;

/\*Create macro variable \*/

**Proc** **Sql** noprint;

Select

ImpliedSinglePrice into :ImpliedSinglePrice

from intermidiatetable;

**Quit**;

%Put &ImpliedSinglePrice;

**proc** **sql**;

create table PlEff\_v5 as

select \*,

&ImpliedSinglePrice ,

APC\_baseprice/&ImpliedSinglePrice as ActualPrice\_Rel

from PlEff\_v4

;

**quit**;

**proc** **sql**;

create table PlEff\_v5 as

select \*,

&ImpliedSinglePrice as ImpliedSinglePrice,

APC\_baseprice/&ImpliedSinglePrice as ActualPrice\_Rel

from PlEff\_v4

;

**quit**;

/\*Create macro variable \*/

**Proc** **Sql** noprint;

Select

ActualPrice\_Rel into :ActualPrice\_Rel\_NE\_DZ16N

from PlEff\_v5

where HRG\_code ='DZ16N' and Setting = 'NE'

;

**Quit**;

%Put &ActualPrice\_Rel\_NE\_DZ16N;

**proc** **sql**;

create table PlEff\_v6 as select \*,

case when HRG\_code ='DZ16N' and Setting = 'NE' then &ActualPrice\_Rel\_NE\_DZ16N

when HRG\_code <>'DZ16N' then &ActualPrice\_Rel\_NE\_DZ16N \*&RelFact\_PleuEff

when HRG\_code ='DZ16N' and Setting = 'SSEM' then &ActualPrice\_Rel\_NE\_DZ16N \*SSEM\_ptc end

as Target\_Rel,

Target\_activity\*(calculated Target\_Rel) as Target\_CostWeights

from PlEff\_v5;

**quit**;

/\*Create macro variable \*/

**Proc** **Sql** noprint;

Select

sum(TotalQuantum\_withCompl)/sum(Target\_CostWeights) into :New\_baseprice

from PlEff\_v6

;

**Quit**;

%Put &New\_baseprice;

**proc** **sql**;

create table BPTOut.PleuralEff as select \*,

case when HRG\_code ='DZ16N' and Setting = 'NE' then &ActualPrice\_Rel\_NE\_DZ16N

when HRG\_code <>'DZ16N' then &ActualPrice\_Rel\_NE\_DZ16N \*&RelFact\_PleuEff

when HRG\_code ='DZ16N' and Setting = 'SSEM' then &ActualPrice\_Rel\_NE\_DZ16N \*SSEM\_ptc end

as Target\_Rel,

Target\_activity\*(calculated Target\_Rel) as Target\_CostWeights,

&New\_baseprice as New\_baseprice,

&New\_baseprice\*(calculated Target\_Rel) as Final\_BPT\_Price

from PlEff\_v6;

**quit**;

**proc** **sql**;

create table BPTOut.PleuralEff as

select HRG\_code,

HRG\_name,

Setting,

case when APC\_basePrice =**.** then **0** else APC\_basePrice end as APC\_basePrice,

SSEM\_ptc,

APC\_baseActivity,

case when Total\_quantum\_actual=**.** then **0** else Total\_quantum\_actual end as Total\_quantum\_actual,

Compliance\_rate as TargetRateforProp,

Target\_activity,

case when TotalQuantum\_withCompl=**.** then **0** else TotalQuantum\_withCompl end as TotalQuantum\_withCompl,

ImpliedSinglePrice,

case when ActualPrice\_Rel=**.** then **0** else ActualPrice\_Rel end as ActualPrice\_Rel ,

Target\_Rel,

Target\_CostWeights,

New\_baseprice,

Final\_BPT\_Price

from BPTOut.PleuralEff;

**quit**;

**proc** **export** data=BPTOut.PleuralEff

outfile="&outputpath.\&outputfile..XLSX"

dbms=XLSX replace;

sheet="08\_PleuralEffusion";

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* author: PID ;

\* date: 02 January 2018 ;

# \* BPT\_09\_SameDayEmergencyCare.sas ;

\* description: 1920 ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

**proc** **sql**;

create table BPTout.SameDayEmCare as

select a.HRG\_code,

a.HRG\_name,

HRG\_or\_SubHRG,

a.BPT\_flag,

b.NE\_Prices as APCbasetariff\_NE,

c.P1\_NE\_TA as Activity\_NE,

c.P1\_SSEM as Activity\_SSEM,

c.P18\_Long\_Stay\_Payment\***100**/b.NE\_Prices as PercIncreasetoderiveBPT,

c.P18\_Long\_Stay\_Payment\*b.NE\_Prices/b.NE\_Prices as AbsoluteDifferential,

d.Percentile75th\_rnd as Compliance\_Rate,

b.NE\_Prices+((calculated AbsoluteDifferential)\*(**1**-d.Percentile75th\_rnd/**100**)) as BPTprice\_SameDayEM,

(calculated BPTprice\_SameDayEM)-(calculated AbsoluteDifferential) as nonBPTpriceNE\_quantumneautral

from BPTInput.BPT\_parameter\_currentyear as a

left join BPTInput.APC\_Prices as b on a.hrg\_code=b.hrg\_code

left join &ActivityTable as c on a.hrg\_code=c.hrg

left join BPTout.ComplRateSDEC\_percentile as d on a.hrg\_code=d.hrg

where a.Category='Same Day Emergency Care'

;

**proc** **export** data=BPTout.SameDayEmCare

outfile="&outputpath.\&outputfile..XLSX"

dbms=XLSX replace;

sheet="09\_SameDayEmergencyCare";

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* author: PID ;

\* date: 02 January 2018 ;

# \* BPT\_10\_TIA.sas ;

\* description: 1920 ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

/\*Manual Adjustment TIA \*/

**Proc** **Sql** noprint;

Select price format=**15.** into :ManualAdj\_TIA

from BPTInput.BPT\_ManualAdjustment

where Category='TIA';

**Quit**;

%Put &ManualAdj\_TIA;

/\*Create macro variable \*/

**Proc** **Sql** noprint;

Select

Inflation\_and\_Efficiency\_Combine into :IE\_combined

from &I\_E\_FactorTable where Year=&currentyear;

**Quit**;

%Put &IE\_combined;

**proc** **sql**;

create table tia1\_v1

as select HRG\_code format=$15., HRG\_name as TF\_name,

b.FAs\_prices\_&currentyear4digits. as OPATTprices\_WF01B,

b.FAm\_prices\_&currentyear4digits. as OPATTprices\_WF02B,

c.CL\_FAS\_act\_PostA as OPATTactivity\_WF01B,

c.CL\_FAM\_act\_PostA as OPATTactivity\_WF02B,

((b.FAs\_prices\_&currentyear4digits.\*c.CL\_FAS\_act\_PostA) +(b.FAm\_prices\_&currentyear4digits.\*c.CL\_FAM\_act\_PostA ))/(c.CL\_FAS\_act\_PostA+c.CL\_FAM\_act\_PostA) as Modelled\_Price,

a.compliance\_rate

from BPTInput.BPT\_parameter\_currentyear as a

left join bptinput.opatt\_prices as b on input(a.HRG\_code,**6.**)=b.TFC

left join bptinput.opatt\_prices\_step\_by\_step as c on input(a.HRG\_code,**6.**)=c.TFC

where a.Category = 'TIA' and a.HRG\_code='329';

**quit**;

**proc** **sql**;

create table bptout.Tia\_Table2

as select

a.HRG\_code format=$15.,

a.HRG\_name as TF\_name,

b.BPT,

&IE\_combined as IE\_combined,

b.BPT\*(**1**+&IE\_combined) as TariffAdj

from BPTInput.BPT\_parameter\_currentyear as a

left join BPTInput.BPT\_previousyear as b on a.hrg\_code=b.hrg\_code

where a.Category = 'TIA' and a.HRG\_code='TopUp';

**quit**;

/\*Create macro variable \*/

**Proc** **Sql**;

Select

TariffAdj into :TariffAdj

from bptout.Tia\_Table2 ;

**Quit**;

**proc** **sql**;

create table bptout.Tia\_Table1

as select \*,

Modelled\_Price-compliance\_rate\*&TariffAdj as Modelled\_Prices\_QN,

(calculated Modelled\_Prices\_QN)\*&ManualAdj\_TIA as FinalPrice

from tia1\_v1

;

**quit**;

**proc** **export** data= bptout.Tia\_Table1

outfile="&outputpath.\&outputfile..XLSX"

dbms=XLSX replace;

sheet="10\_Tia\_Table1";

**run**;

**proc** **export** data= bptout.Tia\_Table2

outfile="&outputpath.\&outputfile..XLSX"

dbms=XLSX replace;

sheet="10\_Tia\_Table2";

**run**;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* author: PID ;

\* date: 19 January 2018 ;

# \* BPT\_11\_EarlyInflammArthitis.sas ;

\* description: 1920 ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

**proc** **sql**;

create table bptout.EIA\_table\_1

as select

a.TFC,

a.Tariff\_TFC as Tariff\_TFC,

a.FAs\_prices\_&currentyear4digits. as OPATTprices\_WF01B,

a.FAm\_prices\_&currentyear4digits. as OPATTprices\_WF02B,

a.FUs\_prices\_&currentyear4digits. as OPATTprices\_WF01A,

a.FUm\_prices\_&currentyear4digits. as OPATTprices\_WF02A,

b.CL\_FAS\_act\_PostA as OPATTactivity\_WF01B,

b.CL\_FAM\_act\_PostA as OPATTactivity\_WF02B,

b.CL\_FUS\_act\_PostA as OPATTactivity\_WF01A,

b.CL\_FUM\_act\_PostA as OPATTactivity\_WF02A,

(b.CL\_FAS\_act\_PostA)+(b.CL\_FAM\_act\_PostA)+(b.CL\_FUS\_act\_PostA)+(b.CL\_FUM\_act\_PostA) as OPATT\_TotalActivity,

a.FAs\_prices\_&currentyear4digits.\*b.CL\_FAS\_act\_PostA AS Price\_quantum\_410\_WF01B,

a.FAm\_prices\_&currentyear4digits.\*b.CL\_FAM\_act\_PostA AS Price\_quantum\_410\_WF02B,

a.FUs\_prices\_&currentyear4digits.\*b.CL\_FUS\_act\_PostA AS Price\_quantum\_410\_WF01A,

a.FUm\_prices\_&currentyear4digits.\*b.CL\_FUM\_act\_PostA AS Price\_quantum\_410\_WF02A,

(calculated Price\_quantum\_410\_WF01B)+(calculated Price\_quantum\_410\_WF02B)+(calculated Price\_quantum\_410\_WF01A)+(calculated Price\_quantum\_410\_WF02A) as OPATT\_TotalQuantum

from bptinput.opatt\_prices as a

left join bptinput.opatt\_prices\_step\_by\_step as b on a.TFC=b.TFC

where a.tfc=**410**;

**quit**;

**proc** **sql**;

create table bptout.EIA\_table\_2

as select

'410' as TFC,

&N\_TopUpInYear as N\_TopUpsInYear,

(b.CL\_FAS\_act\_PostA+b.CL\_FAM\_act\_PostA)-&N\_TopUpInYear as Activity\_NotMeeting\_NotBPT,

(a.FAs\_prices\_&currentyear4digits.\*b.CL\_FAS\_act\_PostA)+(a.FAm\_prices\_&currentyear4digits.\*b.CL\_FAM\_act\_PostA) as TotalSpendingRemainsConstant,

(calculated TotalSpendingRemainsConstant)/(&N\_TopUpInYear+((calculated Activity\_NotMeeting\_NotBPT)\*&Perc\_diff\_TopUp\_NormalPrice)) as TopUpandTFCPrice,

(calculated TopUpandTFCPrice)- a.FAs\_prices\_&currentyear4digits. as TopUp\_Value,

(calculated TopUpandTFCPrice)\*&Perc\_diff\_TopUp\_NormalPrice as Normal\_TFC\_Price,

&N\_TopUpInYear\*(calculated TopUp\_Value) as Unbundled\_Quantum,

(&N\_TopUpInYear\*(calculated TopUpandTFCPrice))+((calculated Activity\_NotMeeting\_NotBPT)\*(calculated Normal\_TFC\_Price)) as Quantum\_Check,

(((a.FAs\_prices\_&currentyear4digits.\*b.CL\_FAS\_act\_PostA )+(a.FAm\_prices\_&currentyear4digits.\*b.CL\_FAM\_act\_PostA)+(a.FUs\_prices\_&currentyear4digits.\*b.CL\_FUS\_act\_PostA)+(a.FUm\_prices\_&currentyear4digits.\*b.CL\_FUM\_act\_PostA)) /\*total quantum\*/

-((calculated Normal\_TFC\_Price)\*(&N\_TopUpInYear+(calculated Activity\_NotMeeting\_NotBPT))

+(a.FUs\_prices\_&currentyear4digits.\*b.CL\_FUS\_act\_PostA) /\*price quantum\_WF01A\*/

+(a.FUm\_prices\_&currentyear4digits.\*b.CL\_FUM\_act\_PostA)))/\*price quantum\_WF01B\*/

/

(((a.FAs\_prices\_&currentyear4digits.\*b.CL\_FAS\_act\_PostA )+(a.FAm\_prices\_&currentyear4digits.\*b.CL\_FAM\_act\_PostA)+(a.FUs\_prices\_&currentyear4digits.\*b.CL\_FUS\_act\_PostA)+(a.FUm\_prices\_&currentyear4digits.\*b.CL\_FUM\_act\_PostA)))

/\*total quantum\*/

as Check\_QuantumShift

from bptinput.opatt\_prices as a

left join bptinput.opatt\_prices\_step\_by\_step as b on a.TFC=b.TFC

where a.tfc=**410**;

**quit**;

**proc** **export** data=BPTOut.EIA\_table\_1

outfile="&outputpath.\&outputfile..XLSX"

dbms=XLSX replace;

sheet="11\_EIA\_table1";

**run**;

**proc** **export** data=BPTOut.EIA\_table\_2

outfile="&outputpath.\&outputfile..XLSX"

dbms=XLSX replace;

sheet="11\_EIA\_table2";

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* author: PID ;

\* date: 19 January 2018 ;

# \* Activity\_with\_spell\_flags.sas ;

\* description: 1920 ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

**proc** **sql**;

create table ActivitySpell\_Flags as

SELECT trim(apc.RC\_SpellHRG\_Rev)||trim(apc.ADM\_SPLIT)||trim(bp.BPTflag) as Concatenation,

trim(apc.RC\_SpellHRG\_Rev)||trim(ADM\_SPLIT) as LU,

apc.RC\_SpellHRG\_Rev as RC\_SpellHRG\_Rev,

case when apc.RC\_SPELLHRG\_REV ='JA20F' and substr(gr.PROC\_01,**1**,**2**) = 'T8' then 'BP32'

when apc.RC\_SPELLHRG\_REV ='JA20F' and substr(gr.PROC\_01,**1**,**3**) ='B27' then 'B28'

when bp.BPTflag is null then '' else bp.BPTflag end as LP\_BPT,

apc.adm\_split,

count(RC\_SpellHRG\_Rev) as Spell\_flag\_Cnt

FROM

bptsql.StageKeyfields as apc

INNER JOIN

BPTINPUT.bptflag as bp

ON apc.RC\_SpellHRG\_Rev = bp.HRgcode

left join

bptsql.groupeddata as gr

on apc.epikey=gr.epikey

where apc.SpellFlag = **1**

GROUP BY

calculated Concatenation,

calculated LU,

apc.RC\_SpellHRG\_Rev,

calculated LP\_BPT,

apc.adm\_split

order by LU,

RC\_SpellHRG\_Rev

;

**quit**;

**proc** **sql**;

create table BPTOut.ActivitySpell\_Flags as select \* from ActivitySpell\_Flags;

**quit**;

**proc** **export** data=BPTOut.ActivitySpell\_Flags

outfile="&outputpath.\&outputfile..XLSX"

dbms=XLSX replace;

sheet="ActivitySpellFlags";

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 05 April 2017 ;

# \* AEUnitCost\_1.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* AEUnitCost;

\* Calculate Unit Costs at start of excel model;

**%macro** AEUnitCost(schema=);

proc sql;

create table AE\_Unit\_Cost as

select HRG,

SumOfTotalActivity,

SumOfTotalCost,

SumOfTotalCost / SumOfTotalActivity as Unit\_Cost

from AE\_Total\_Activity\_Cost;

quit;

\*/ Added output of data ;

%let outputfile=AE\_Unit\_Cost;

%let outputpath=\\irnarch\sas\_data\PID\AE\AE\_outputs;

proc export data=WORK.AE\_Unit\_Cost

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

run;

**%mend** AEUnitCost;

%***AEUnitCost***(schema=sas\_config)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 03 March 2017 ;

# \* AEdataprep.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* Connects to SQL by ODBC to get non acute ref cost data

\* Formats to get data needed for A&E model;

**%macro** AEdataprep(schema=);

proc sql; create table AEalldata as select \* from work.reference\_cost; quit;

PROC SQL;

CREATE TABLE AEdata1 AS

SELECT

t1.organisationcode as FK\_ORGS\_PROV\_ID,

t1.suppliertype as SUPPLIER\_TYPE,

t1.departmentcode as DEPARTMENT,

t1.servicecode as SERVICE,

t1.currencycode as CURRENCY,

t1.unitcost as UNIT\_COST,

t1.activity as ACTIVITY\_P1

FROM AEalldata t1

WHERE t1.DEPARTMENTcode='EM' AND t1.CURRENCYcode Not Like 'uz01z'

ORDER BY t1.DEPARTMENTcode;

QUIT;

**%mend** AEdataprep;

%***AEdataprep***(schema=sas\_config)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 03 March 2017 ;

# \* AE\_NAUnitCost.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

/\*NA Unit Cost

Calculates the unit cost of A&E attendances that do not result in an admission

Then Applies Non-Admission unit costs to all attendances;

**%macro** AE\_NAUnitCost(schema=);

proc SQL;

create table AE01\_06A as

select in\_5a.HRG,

in\_5b.'T01NA'n / in\_5a.'T01NA'n as EM\_T01NA\_UC,

in\_5b.'T03NA'n / in\_5a.'T03NA'n AS EM\_T03NA\_UC,

in\_5b.'T02NA'n / in\_5a.'T02NA'n AS EM\_T02NA\_UC

from AE01\_05AA in\_5a inner join AE01\_05BB in\_5b on in\_5a.HRG = in\_5b.HRG ;

Quit;

proc sql;

create table AE01\_06b as

SELECT

A.\*,

EM\_T01\_TC + EM\_T03\_TC + EM\_T02\_TC AS Total\_Cost from(

SELECT

in\_5a.HRG,

COALESCE(in\_5a.T03A,**0**)+COALESCE(in\_5a.T03NA,**0**)+COALESCE(in\_5a.T01A,**0**)+COALESCE(in\_5a.T01NA,**0**)+COALESCE(in\_5a.T02A,**0**)+COALESCE(in\_5a.T02NA,**0**) AS Total\_Activity,

COALESCE((EM\_T01NA\_UC\*in\_5a.T01A),**0**)+COALESCE(in\_5b.T01NA,**0**) AS EM\_T01\_TC,

COALESCE((EM\_T03NA\_UC\*in\_5a.T03A),**0**)+COALESCE(in\_5b.T03NA,**0**) AS EM\_T03\_TC,

COALESCE((EM\_T02NA\_UC\*in\_5a.T02A),**0**)+COALESCE(in\_5b.T02NA,**0**) AS EM\_T02\_TC

from AE01\_05AA in\_5a inner join AE01\_05BB in\_5b on in\_5a.HRG = in\_5b.HRG

INNER JOIN AE01\_06A in\_06a ON in\_5B.HRG = in\_06a.HRG ) AS A;

QUIT;

**%mend** AE\_NAUnitCost;

%***AE\_NAUnitCost***(schema=sas\_config)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 30 March 2017 ;

# \* Pre\_AdmTC.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* Pre\_AdmTC ;

\* Extracts the total cost of A&E attendances pre-admission ;

\* Then does the same for the post admission A&E attendences;

\* Finally calculates difference between pre and post admission costs, to establish total cost of non-admissions ;

**%macro** Pre\_AdmTC(schema=);

proc sql;

create table AE01\_07a as

SELECT Sum(Total\_Cost) AS PreAdm\_TC

FROM AE\_National;

quit;

proc sql;

create table AE01\_07b as

SELECT Sum(Total\_Cost) AS PostAdm\_TC

FROM AE01\_06b ;

quit;

proc sql;

create table AE\_AD\_NE\_Tariff as

select PreAdm\_TC - PostAdm\_TC AS AE\_AD\_TC

FROM AE01\_07a, AE01\_07b ;

quit;

data Modelout.AE\_AD\_NE\_Tariff; set work.AE\_AD\_NE\_Tariff; run;

proc sql;

create table AE\_Adm\_Cost as

SELECT PreAdm\_TC - PostAdm\_TC AS AE\_AD\_TC

FROM AE01\_07a, AE01\_07b ;

QUIT ;

data Modelout.ae\_adm\_cost\_quantum2apc; set AE\_Adm\_Cost; id=&id\_ae2apc.; Year=&year\_ae2apc.; run;

proc sql; create table Modelout.ae\_adm\_cost\_quantum2apc as select id, year, ae\_ad\_tc from

Modelout.ae\_adm\_cost\_quantum2apc; quit;

**%mend** Pre\_AdmTC;

%***Pre\_AdmTC***(schema=sas\_config)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 30 March 2017 ;

# \* AE\_TA\_CostbyHRG.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* AE\_TA\_CostbyHRG;

\* Total Activity and Cost by HRG;

\* Summary of costs by HRG;

**%macro** AE\_TA\_CostbyHRG(schema=);

proc sql;

create table AE\_Total\_Activity\_Cost as

SELECT

HRG,

Sum(Total\_Activity) AS SumOfTotalActivity,

Sum(Total\_Cost) AS SumOfTotalCost

FROM AE01\_06b

GROUP BY HRG ;

QUIT;

\*/ Added output of data ;

%let outputfile=AE\_TC\_by\_HRG;

%let outputpath=\\irnarch\sas\_data\PID\AE\AE\_outputs;

proc export data=WORK.AE\_Total\_Activity\_Cost

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

run;

**%mend** AE\_TA\_CostbyHRG;

%***AE\_TA\_CostbyHRG***(schema=sas\_config)

**data** xxx; set work.QUANTUMRECONCILATION\_OUTPUT; **run**;

**data** yyy; set AE\_Unit\_Cost; **run**;

**proc** **sql**; create table aaa as select "A\_E" FORMAT=$20. as POD, sum(sumoftotalcost) as Quantum4qr from yyy; **quit**;

**PROC** **SQL**;

CREATE TABLE B2 AS SELECT A.\*, b.quantumrecvalues,

(b.quantumrecvalues/A.quantum4qr)-**1** as qr\_factor1 FROM AAA AS A

LEFT JOIN xxx AS B

ON A.POD=B.POD

where b.pod="A\_E"/

**QUIT**;

**proc** **sql**;

select qr\_factor1 into :qr1\_factor

from b2

where POD="A\_E"

;

**quit**;

%let qr1\_factor=&qr1\_factor.;

**%macro** AE\_QR1(schema=, QR1factor=&qr1\_factor.);

PROC SQL;

create table AE\_QR1 as

select HRG,

SumOfTotalActivity,

SumOfTotalCost,

Unit\_Cost,

&QR1factor as QR1\_factor,

Unit\_Cost \* (**1** + &QR1factor.) as Modelled\_Tariff\_QR1

from AE\_Unit\_Cost ;

quit;

**%mend** AE\_QR1;

%***AE\_QR1***(schema=sas\_config)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 05 April 2017 ;

# \* AECostBase.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

;

**%macro** AECostBase(schema=/\*, Cost\_Base\_Adjustment = 0\*/);

PROC SQL;

create table AE\_CostBase as

select HRG,

SumOfTotalActivity,

SumOfTotalCost,

Unit\_Cost,

QR1\_factor format=**13.12**,

Modelled\_Tariff\_QR1,

&Cost\_Base\_Adjustment as Cost\_Base\_Adjustment,

Modelled\_Tariff\_QR1 \* (**1** + &Cost\_Base\_Adjustment) as Prices\_QR1\_and\_CB

from AE\_QR1 ;

quit;

**%mend** AECostBase;

%***AECostBase***(schema=sas\_config)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 05 April 2017 ;

# \* AEInfEffCNST1.sas ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

**%macro** AEInfEffCNST1(schema= /\*, Inf\_Eff = -0.006211334 , CNST = 0.039785733, CNST2 = 0.00000\*/);

PROC SQL;

create table AE\_Inf\_Eff\_CNST1\_mo as

select HRG,

SumOfTotalActivity,

SumOfTotalCost,

Unit\_Cost,

QR1\_factor,

Modelled\_Tariff\_QR1,

Cost\_Base\_Adjustment,

Prices\_QR1\_and\_CB,

&Inf\_Eff as Inf\_Eff,

&CNST as CNST,

/ (**1**+&Inf\_Eff)\*(**1**+&CNST)-**1** as Total\_Adjust,

Prices\_QR1\_and\_CB \* ((**1**+&Inf\_Eff)\*(**1**+&CNST)) as Prices\_after\_uplifts1

from AE\_CostBase ;

quit;

data AE\_Inf\_Eff\_CNST1; set AE\_Inf\_Eff\_CNST1\_mo; if HRG in("VB10Z" "VB11Z") then Prices\_after\_uplifts1=Prices\_QR1\_and\_CB\*((**1**+&Inf\_Eff)\*(**1**+&CNST2));

else Prices\_after\_uplifts1=Prices\_after\_uplifts1;

run;

data AE\_Inf\_Eff\_CNST1\_mo; set AE\_Inf\_Eff\_CNST1;

if HRG in("VB10Z" "VB11Z") then CNST=&CNST2; else CNST=&CNST;

if HRG in("VB10Z" "VB11Z") then Total\_Adjust=(**1**+&Inf\_Eff)\*(**1**+&CNST2)-**1**; else Total\_Adjust=Total\_Adjust;

run;

**%mend** AEInfEffCNST1;

%***AEInfEffCNST1***(schema=sas\_config)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 05 April 2017 ;

# \* AEManualAdjust.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

Libname PostMA base "\\irnarch\sas\_data\Tariff Rebuild\Model outputs pre\_MA\AnE\FY1920\MA\_Data";

**%macro** AE\_Manual\_Adjust(schema=);

DATA MA\_AnE\_Input;

SET PostMA.MA\_AnE\_Input;

RUN;

DATA MA\_AnE\_Output;

SET PostMA.MA\_AnE\_Output;

RUN;

proc sql;

create table new1

as select

a.hrg,

d.hrg\_name,

a.\*,

B.HRG\_CODE,

B.PRICE\_DECISION,

c.PostMA\_PRICE

from AE\_Inf\_Eff\_CNST1\_mo as a

left join MA\_AnE\_Input as b

on a.hrg=b.HRG\_CODE

LEFT JOIN MA\_AnE\_Output as c

on a.hrg=c.HRG\_CODE

left join tariff\_r.a\_e\_1718 as d

on a.hrg=d.hrg\_code

;

quit;

data new2(drop=PostMA\_Price);

set new1;

if price\_decision="Adjusted" then Prices\_after\_MA=PostMA\_Price;

else Prices\_after\_MA=Prices\_after\_uplifts1;

format Prices\_after\_MA Comma7.0;

quantum\_pre\_MA=SumOfTOTALACTIVITY\*Prices\_after\_uplifts1;

quantum\_post\_MA=SumOfTOTALACTIVITY\*Prices\_after\_MA;

run;

PROC SQL;

SELECT MIN(PRICES\_AFTER\_MA) AS LOWEST\_P INTO :LATE\_CHANGE

FROM new2;

QUIT;

data late\_change;

set new2;

if hrg="VB99Z" THEN PRICES\_AFTER\_MA=&LATE\_CHANGE.;

ELSE PRICES\_AFTER\_MA=PRICES\_AFTER\_MA;

quantum\_pre\_MA=SumOfTOTALACTIVITY\*Prices\_after\_uplifts1;

quantum\_post\_MA=SumOfTOTALACTIVITY\*Prices\_after\_MA;

run;

DATA new2;

SET late\_change;

RUN;

proc sql;

create table new3 as select sum(quantum\_pre\_MA) as qtyprema, sum(quantum\_post\_MA) as qtypostma,

sum(quantum\_pre\_MA)/sum(quantum\_post\_MA) as qr2\_ma from new2;

quit;

proc sql;

select qr2\_ma into :qr2\_factor

from new3;

quit;

%let rc\_qr2\_factor=&qr2\_factor.;

data AE\_Manual\_adjust(drop=hrg\_code); set new2;

rc\_qr2\_factor=&rc\_qr2\_factor.;

Prices\_after\_QR2=Prices\_after\_MA\*rc\_qr2\_factor;

format Prices\_after\_QR2 comma15.0;

format QR1\_factor percent7.2;

run;

**%mend** AE\_Manual\_Adjust;

%***AE\_Manual\_Adjust***(schema=sas\_config)

**proc** **sql**;

create table AE\_Prices\_4\_IA\_model as select **4** format=**8.** as ae\_price\_id, "19/20" format=$8. as year,

hrg as hrg\_code, hrg\_name, prices\_after\_qr2 as type\_1\_2\_prices format=comma16.15,

MIN(prices\_after\_qr2) as type\_3\_prices format=comma16.15 FROM AE\_Manual\_adjust;

**quit**;

**data** Modelout.AE\_Prices\_4\_IA\_model;

set AE\_Prices\_4\_IA\_model;

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 05 April 2017 ;

# \* AEUnitCost\_2.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

**%macro** AE\_Smoothing(schema=, Smoothing\_Factor=);

PROC SQL;

create table AE\_Smoothing as

select t1.\*,

&SMF. as Smoothing\_Factor format=percentn8.2,

t1.Prices\_after\_QR2 \* (**1** + &SMF.) as Prices\_after\_Smoothing

from AE\_Manual\_adjust as t1

;

Quit;

%let outputfile=AE\_Smoothing;

%let outputpath=\\irnarch\sas\_data\PID\AE\AE\_outputs;

proc export data=WORK.AE\_Smoothing

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

run;

**%mend** AE\_Smoothing;

%***AE\_Smoothing***(schema=sas\_config)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID

\* Date: 05 April 2017 ; ;

\* AEUnitCost.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* AEUnitCost;

\* Calculate Unit Costs at start of excel model;

**%macro** AE\_Scaling(schema=/\*, Scaling\_factor=\*/);

PROC SQL;

create table AE\_Scaling as

select \*,

&Scaling\_factor. as Scaling\_factor format=percentn8.2,

Prices\_after\_Smoothing \* (**1**+ &Scaling\_factor.) as Prices\_after\_scaling

from AE\_Smoothing;

Quit;

\*/ Added output of data ;

%let outputfile=AE\_Scaling;

%let outputpath=\\irnarch\sas\_data\PID\AE\AE\_outputs;

proc export data=WORK.AE\_Scaling

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

run;

**%mend** AE\_Scaling;

%***AE\_Scaling***(schema=sas\_config)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 30 March 2017 ;

# \* AEfilterscope.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

**%macro** AEfilterscope(schema=);

proc sql; create table AEprov as select \* from work.tc\_mff; quit;

PROC SQL;

CREATE TABLE WORK.AEscope AS

SELECT t1.rc\_code as Provider\_Code,

t2.SERVICE,

t2.CURRENCY,

t2.UNIT\_COST,

t2.ACTIVITY\_P1,

(t2.UNIT\_COST \* t2.ACTIVITY\_P1) AS TotalCost

FROM WORK.AEDATA1 t2

INNER JOIN WORK.AEPROV t1 ON (t2.FK\_ORGS\_PROV\_ID = t1.RC\_Code)

WHERE t2.SERVICE NOT CONTAINS 'T04' AND t2.SERVICE NOT CONTAINS 'UZ01Z' AND t2.DEPARTMENT = 'EM' AND

t2.SUPPLIER\_TYPE = 'OWN';

QUIT;

PROC SQL;

CREATE TABLE WORK.AEdata01\_01 AS

SELECT sum(Unit\_cost) as Unitcost,

sum(Activity\_P1) as Activity,

sum(TotalCost) as TotalCosts

FROM WORK.AEscope;

QUIT;

**%mend** AEfilterscope;

%***AEfilterscope***(schema=sas\_config)

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 05 April 2017 ;

# \* AEUnitCost\_3.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* AEUnitCost;

\* Calculate Unit Costs at start of excel model;

/\* START CIO \*/

**data** r0;

set AE\_Scaling;

**run**;

**proc** **sql**;

create table r1

as select a.\*, b.\* from r0 as a

left join Cash\_in\_Out\_ia\_factors as b

on substr(a.hrg,**1**,**2**)=b.Subchapter;

**quit**;

**data** r9(drop=POD\_SUBCHAPTER SUBCHAPTER CASHIO\_FACTOR\_POD\_SUBCH final\_CASHIO\_FACTOR);

set r1;

PRICES\_AFTER\_SCALING\_IA=PRICES\_AFTER\_SCALING\*(**1**+FINAL\_CASHIO\_FACTOR);

**run**;

**DATA** AE\_Scaling(DROP=PRICES\_AFTER\_SCALING\_IA);

SET r9 (DROP=PRICES\_AFTER\_SCALING);

PRICES\_AFTER\_SCALING=PRICES\_AFTER\_SCALING\_IA;

**RUN**;

**%macro** AEInfEffCNST2(schema=);

PROC SQL;

create table AE\_Inf\_Eff\_CNST2\_mo as

select \*,

&Inf\_Eff\_1920. as Inf\_Ef\_1920,

&CNST\_1920. as CNST\_1920 ,

Prices\_after\_scaling \*(**1**+&Inf\_Eff\_1920.)\*(**1**+&CNST\_1920.) as Prices\_after\_adjustments2

from AE\_Scaling

order by HRG;

Quit;

data AE\_Inf\_Eff\_CNST2; set AE\_Inf\_Eff\_CNST2\_mo; if HRG in("VB10Z" "VB11Z"/\* "VB99Z" \*/) then Prices\_after\_adjustments2=Prices\_after\_scaling\*((**1**+&Inf\_Eff\_1920.)\*(**1**+&CNST2\_1920.));

else Prices\_after\_adjustments2=Prices\_after\_adjustments2;

if HRG in("VB10Z" "VB11Z"/\* "VB99Z" \*/) then CNST\_1920=&CNST2\_1920.;

else CNST\_1920=&CNST\_1920.; run;

\*/ Added output of data ;

%let outputfile=AE\_Inf\_Eff\_CNST2;

%let outputpath=\\irnarch\sas\_data\PID\AE\AE\_outputs;

proc export data=WORK.AE\_Inf\_Eff\_CNST2

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

run;

**%mend** AEInfEffCNST2;

%***AEInfEffCNST2***(schema=sas\_config)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 05 April 2017 ;

# \* AEUnitCost\_4.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* AEUnitCost;

\* Calculate Unit Costs at start of excel model;

**proc** **sql**;

create table w1 as select \* from AE\_Inf\_Eff\_CNST2;

**quit**;

**proc** **sql**;

select prices\_after\_adjustments2 into :vb99\_equalise\_vb11

from AE\_Inf\_Eff\_CNST2

where HRG="VB11Z";

**quit**;

**data** w2;

set w1;

if HRG="VB99Z" then prices\_after\_adjustments2=&vb99\_equalise\_vb11.;

else prices\_after\_adjustments2=prices\_after\_adjustments2;

**run**;

**data** AE\_Inf\_Eff\_CNST2;

set w2;

**run**;

**%macro** AE\_Final\_Prices(schema=);

proc sql;

create table AE\_Final\_Type1n2 as

select \*,

Prices\_after\_adjustments2 as Final\_Prices\_Type1n2 FORMAT=COMMA7.0

from AE\_Inf\_Eff\_CNST2;

QUIT;

proc sql;

create table AE\_Final\_Prices as

SELECT HRG,

hrg\_name,

Final\_Prices\_Type1n2,

MIN(Final\_Prices\_Type1n2) as Final\_Prices\_Type3 FORMAT=COMMA7.0 FROM AE\_Final\_Type1n2;

quit;

/\*data work.AE\_Final\_Prices;\*/

data modelout.AE\_Final\_Prices;

set AE\_Final\_Prices;

run;

proc sql;

create table AE\_Prices\_STEP\_BY\_STEP as

SELECT \*,

Final\_Prices\_Type1n2,

MIN(Final\_Prices\_Type1n2) as Final\_Prices\_Type3 FORMAT=COMMA7.0 FROM AE\_Final\_Type1n2;

quit;

/\*data work.AE\_Prices\_STEP\_BY\_STEP;\*/

data modelout.AE\_Prices\_STEP\_BY\_STEP;

set AE\_Prices\_STEP\_BY\_STEP;

format Inf\_Eff CNST Total\_Adjust percent7.2;

format QR1\_factor percent14.11;

run;

**%mend** AE\_Final\_Prices;

%***AE\_Final\_Prices***(schema=sas\_config)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 03 March 2017 ;

# \* AErecodeHRGs.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\*Get the recode data

\*Join recode data to AEScope

\*Dead on Arrival = VB09Z. Type 2 (Non 24hr) and Type 3 (Minor Injury Service) recode ;

**%macro** AErecodeHRGs(schema=);

proc sql; create table AE\_HRG\_Recode as select \* from work.HRG\_recode; quit;

Proc Sql;

Create table WORK.AE01\_02A

as SELECT distinct

sd.Provider\_code,

sd.service,

CASE WHEN currency = 'VBDOA' THEN 'VB09Z' ELSE Recode END as HRG\_code,/\*

Sum(sd.Activity\_p1) AS SumOfActivity,

Sum(sd.TotalCost) AS SumOfTotalCost

FROM

AE\_HRG\_Recode as in\_06

INNER JOIN AEScope sd

ON (in\_06.HRG = sd.currency)

AND (in\_06.Department = sd.service)

GROUP BY sd.Provider\_code, sd.service,

CASE WHEN currency = 'VBDOA' THEN 'VB09Z' ELSE Recode END;

quit;

**%mend** AErecodeHRGs;

%***AErecodeHRGs***(schema=sas\_config)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 30 March 2017 ;

# \* AEremoveMFF.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\*/

--02b: Remove MFF

--MFF Removed to exclude location-specific uplifts ;

\*Join last tabel to provider table

;

**%macro** AEremoveMFF(schema=);

PROC SQL;

CREATE TABLE AE\_Recode\_MFF AS

SELECT t2.Provider\_Code,

t2.SERVICE,

t2.HRG\_code,

t2.SumOfActivity AS Activity,

t2.SumOfTotalCost/t2.SumOfActivity AS UnitCost,

t2.SumOfTotalCost AS TotalCost,

t2.SumOfTotalCost/t2.SumOfActivity /t1.Uncapped\_MFF AS MFF\_Adj\_UC,

t2.SumOfTotalCost /t1.Capped\_MFF AS MFF\_Adj\_TC,

t2.SumOfTotalCost/t2.SumOfActivity/t1.Capped\_MFF AS MFF\_Adj\_UC\_Capped,

t2.SumOfTotalCost/t1.Capped\_MFF AS MFF\_Adj\_TC\_Capped

FROM AE01\_02A as t2

INNER JOIN AEprov t1 ON (t2.Provider\_Code = t1.RC\_Code);

QUIT;

**%mend** AEremoveMFF;

%***AEremoveMFF***(schema=sas\_config)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 03 March 2017 ;

# \* AEClacNationalAvg.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\*Calculates national average unit costs for use in "1\_5 Clean Data first step" ;

**%macro** AEClacNationalAvg(schema=);

PROC SQL;

CREATE TABLE AE01\_03 AS

SELECT t2.SERVICE,

t2.HRG\_code,

Sum(t2.Activity) AS Total\_Activity,

Sum(t2.MFF\_Adj\_TC) AS Total\_MFF\_Adj\_TC,

Sum(t2.MFF\_Adj\_TC)/ Sum(t2.Activity) AS National\_Average\_UC

FROM AE\_Recode\_MFF as t2

GROUP BY t2.service, t2.HRG\_code;

QUIT;

**%mend** AEClacNationalAvg;

%***AEClacNationalAvg***(schema=sas\_config)

\*

--03: National Averages

--Calculates national average unit costs for use in "03b: Clean Data"

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 03 April 2017 ;

# \* AECleanData1.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* Excludes any records with MFF Adj UCs <0.05 or >20 times the national average UC ;

\* Needs PARAMATERISING as HARD CODED RIGHT NOW ;

**%macro** AECleanData1(schema=);

%let Param\_OutlierMinValue = 0.05;

%let Param\_OutlierMaxValue = 20;

Proc Sql;

Create table stage\_AE\_Clean

as SELECT

ae.Provider\_Code,

ae.Service AS Department,

ae.HRG\_code AS HRG,

ae.Activity,

ae.MFF\_Adj\_TC,

MFF\_adj\_TC/activity AS MFF\_Adj\_UC,

in\_03.National\_Average\_UC,

CASE WHEN ((&param\_OutlierMinValue.=**0** OR &param\_OutlierMinValue. IS NULL) AND (&param\_OutlierMaxValue. =**0** OR &param\_OutlierMaxValue. IS NULL)) THEN **0**

ELSE

CASE WHEN (MFF\_adj\_TC/activity)/National\_Average\_UC <&param\_OutlierMinValue. OR (MFF\_adj\_TC/activity)/National\_Average\_UC>&param\_OutlierMaxValue.

THEN **1** ELSE **0** END

END /\*--Replaced value 0.05 with @param\_OutlierMinValue AND value 20 with @param\_OutlierMaxValue - 18/12/2014\*/ as Exclude

FROM AE\_Recode\_MFF as ae

INNER JOIN AE01\_03 in\_03

ON ae.HRG\_code = in\_03.HRG\_code

AND ae.Service = in\_03.Service

WHERE

(CASE WHEN ((&param\_OutlierMinValue.=**0** OR &param\_OutlierMinValue. IS NULL) AND (&param\_OutlierMaxValue. =**0** OR &param\_OutlierMaxValue. IS NULL)) THEN **0**

ELSE

CASE WHEN (MFF\_adj\_TC/activity)/National\_Average\_UC <&param\_OutlierMinValue. OR (MFF\_adj\_TC/activity)/National\_Average\_UC>&param\_OutlierMaxValue. THEN **1**

ELSE **0** END

/\* --Replaced value 0.05 with @param\_OutlierMinValue AND value 20 with @param\_OutlierMaxValue - 18/12/2014\*/

END

) = **0**;

quit;

data AE\_Clean1; set stage\_AE\_Clean; run;

**%mend** AECleanData1;

%***AECleanData1***(schema=sas\_config)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 05 April 2017 ;

# \* AECleanData2.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* National Data - Clean2 ;

\*Aggregates "CleanData1" to department and HRG level;

\* step1 create national post cleaning prices ;

**%macro** AECleanData2(schema=);

proc sql;

create table AE\_National as

SELECT Department,

HRG,

Sum(Activity) AS Total\_Activity,

Sum(MFF\_Adj\_UC\*Activity) AS Total\_Cost

FROM AE\_Clean1

GROUP BY

Department,

HRG;

quit;

proc sql;

create table AE01\_04 as

SELECT Department,

HRG,

Sum(Activity) AS Total\_Activity,

Sum(MFF\_Adj\_UC\*Activity) AS Total\_Cost

FROM AE\_Clean1

GROUP BY

Department,

HRG;

quit;

**%mend** AECleanData2;

%***AECleanData2***(schema=sas\_config)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 03 March 2017 ;

# \* A&E\_Create\_Execution\_Instance.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* AEActivityCrosstab;

\*Might be an output for metrics engine or excel etc;

\*definately consider recoding in SAS at end ;

\* The proc transpose needs checking here...once data polulates through;

**%macro** AEActivityCrosstab(schema=);

Proc Sort data=AE\_National;

by HRG;

RUN;

PROC TRANSPOSE data=AE\_National out=WORK.AE01\_05a;

by HRG;

id Department;

var Total\_Activity;

RUN;

PROC SQL;

CREATE TABLE AE01\_05AA AS

SELECT t1.HRG,

t1.\_NAME\_,

t1.T01A,

t1.T01NA,

t1.T02A,

t1.T02NA,

t1.T03A,

t1.T03NA

FROM WORK.AE01\_05A t1

ORDER BY t1.HRG;

QUIT;

**%mend** AEActivityCrosstab;

%***AEActivityCrosstab***(schema=sas\_config)

;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 30 March 2017 ;

# \* AECostCrosstab.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\*AECostCrosstab ;

\*Tabulates total cost by HRG and Department ;

**%macro** AECostCrosstab(schema=);

PROC TRANSPOSE data=AE\_National out=AE01\_05b;

by HRG;

id Department;

var Total\_Cost;

RUN;

PROC SQL;

CREATE TABLE AE01\_05BB AS

SELECT t1.HRG,

t1.\_NAME\_,

t1.T01A,

t1.T01NA,

t1.T02A,

t1.T02NA,

t1.T03A,

t1.T03NA

FROM WORK.AE01\_05B t1

ORDER BY t1.HRG;

QUIT;

**%mend** AECostCrosstab;

%***AECostCrosstab***(schema=sas\_config)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 02 March 2018 ;

# \* BPT\_AnnexA.sas ;

\* description: 1920 ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

/\*This code is used to create the annex A APC OPROC table for the BPT columns \*/

**proc** **sql**;

create table Annex\_A\_APC\_OPROC

as

select

a.\*,

b.Area\_BPT as BPT\_area,

b.HRG\_or\_SubHRG,

b.BPT\_flag,

b.POD\_BPT,

b.SUS\_BPT\_nonBPT,

b.Price\_information

from BPTInput.APC\_OPROClist as a

inner join BPTInput.BPT\_PARAMETER\_CURRENTYEAR as b

on a.HRG\_code=b.HRG\_code;

**quit**;

**proc** **sql**;

create table Annex\_A\_APC\_OPROC2

as

select HRG\_code,

case when HRG\_or\_SubHRG is null then '-' else HRG\_or\_SubHRG end as BPT\_HRGsubHRG,

case when BPT\_area is null then '-' else BPT\_area end as BPT\_area,

case when POD\_BPT is null then '-' else POD\_BPT end as POD\_BPT,

case when SUS\_BPT\_nonBPT is null then '-' else SUS\_BPT\_nonBPT end as SUS\_BPT\_nonBPT,

Price\_information,

case when BPT\_flag is null then '-' else BPT\_flag end as BPT\_flag

from Annex\_A\_APC\_OPROC

;

**quit**;

**proc** **sql**;

create table lnk

as select a.HRG\_code, non\_best\_practice\_tariff as nonBPT,best\_practice\_tariff as bpt, 'table\_1b' as type, '' as procedure from bptout.lnk\_1b\_stroke\_table2 as a

union

(select b.HRG\_code, '' as nonbpt,best\_practice\_tariff as bpt, 'table\_2a1' as type, '' as procedure from bptout.lnk\_2a1\_renal\_hemod\_hospital\_t1 as b)

union

(select c.HRG\_code ,'' as nonbpt,best\_practice\_tariff as bpt, 'table\_2a2' as type, '' as procedure from bptout.lnk\_2a2\_renal\_hemod\_satellite\_t2 as c)

union

(select d.HRG\_code, '' as nonbpt,best\_practice\_tariff as bpt, 'table\_2a3' as type, '' as procedure from bptout.lnk\_2a3\_renal\_hemod\_home\_t3 as d)

union

(select e.HRG\_code, '' as nonbpt,price as bpt, 'table\_2b' as type, '' as procedure from bptout.lnk\_2b\_renal\_peritoneal\_t4 as e)

union

(select f.HRG\_code, elective\_nonbestpractice\_tariff as nonbpt, best\_practice\_day\_case\_tariff as bpt, 'table\_3' as type, procedure as procedure from bptout.lnk\_3\_day\_case as f)

union

(select g.HRG\_code, non\_bpt\_non\_elective\_tariff as nonbpt, bpt\_non\_elective\_tariff as bpt, 'table\_4' as type, '' as procedure from bptout.lnk\_4\_diabete as g)

union

(select i.code, lev\_3\_non\_best\_practice\_tariff as nonbpt, lev\_1\_best\_practice\_tariff as bpt, 'table\_6' as type, '' as procedure from bptout.lnk\_6\_endoscopy\_procedure as i)

union

(select l.code, non\_best\_practice\_tariff as nonbpt, best\_practice\_tariff as bpt, 'table\_7' as type, '' as procedure from bptout.lnk\_7\_Fragility\_hip\_fracture as l)

union

(select n.HRG\_code, nonbest\_pract\_dc\_ord\_el\_tariff as nonbpt, best\_pract\_outpatient\_tariff as bpt, 'table\_9' as type, '' as procedure from bptout.lnk\_9\_Outpatients as n )

union

(select o.HRG\_code, '' as nonbpt, best\_practice\_tariff as bpt, 'table\_10' as type, '' as procedure from bptout.lnk\_10\_Paediatric\_diabetes as o)

union

(select r.HRG\_code, '' as nonbpt, day\_case\_best\_practice\_tariff as bpt, 'table\_13a' as type, '' as procedure from bptout.lnk\_13a\_PleuralEffusion as r)

union

(select s.HRG\_code, '' as nonbpt, day\_case\_best\_practice\_tariff as bpt, 'table\_13b' as type, '' as procedure from bptout.lnk\_13b\_PleuralEffusion as s)

union

(select t.HRG\_code, non\_best\_practice\_tariff as nonbpt, best\_practice\_tariff as bpt, 'table\_14' as type, '' as procedure from bptout.lnk\_14\_Primary\_hipknee\_replac as t)

union

(select u.HRG\_code, non\_elective\_non\_bpt\_tariff as nonbpt, same\_day\_emergency\_care\_bpt as bpt, 'table\_15' as type, '' as procedure from bptout.LNK\_15\_SAME\_DAY\_EM\_CARE as u)

union

(select aa.HRG\_code, non\_bpt\_non\_elective\_tariff as nonbpt, bpt\_non\_elective\_tariff as bpt, 'table\_17' as type, '' as procedure from bptout.lnk\_17\_heartfailure as aa)

union

(select bb.HRG\_code, non\_bpt\_non\_elective\_tariff as nonbpt, bpt\_non\_elective\_tariff as bpt, 'table\_18' as type, '' as procedure from bptout.lnk\_18\_COPD\_Exacerbation as bb)

union

(select cc.HRG\_code, non\_bpt\_non\_elective\_tariff as nonbpt, bpt\_non\_elective\_tariff as bpt, 'table\_19' as type, '' as procedure from bptout.lnk\_19\_NSTEMI as cc)

union

(select dd.HRG\_code, non\_bpt\_non\_elective\_tariff as nonbpt, bpt\_non\_elective\_tariff as bpt, 'table\_20' as type, '' as procedure from bptout.lnk\_20\_EM\_LAPAROTOMY as dd);

**quit**;

**proc** **sql**;

create table BPTout.Annex\_A\_APC\_OPROC

as select \*,

case when BPT\_Area = 'Outpatients' then **1**

when a.HRG\_code in ('YD04Z', 'YD05Z') then **2**

when BPT\_Area = 'Daycase' then **3** else **0** end as Condition\_index,

case when calculated Condition\_index = **1** then 'in Annex A to replace column D (outpatient proc tariff) with BPT prices and column E (

Combined dc/ ordinary el) with nonBPT prices'

when calculated Condition\_index = **2** then ' in Annex A to replace column E (combined day case elective) using BPT prices'

when calculated Condition\_index = **3** then 'in Annex A to remove the values in the (combined daycase el. column) and the split values for DayCase column F (BPt prices) and elective column G (non bpt prices) separated'

else '' end as Condition\_Information

from Annex\_A\_APC\_OPROC2 as a

left join lnk as b

on a.HRG\_code=b.HRG\_code;

**quit**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* APC\_Initialise\_Parameters.sas

\* date: 8 Oct 2015 ;

;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\*Define a guid for the SAS Session i.e. for the execution;

**data** \_null\_;

attrib guid length=$**36**;

guid=uuidgen();

call symputx('SAS\_Session',guid,'G');

**run**;

\*system start time;

%let Program\_start=%sysfunc(DateTIME());

**data** \_null\_;

username=sysget('USERNAME');

user\_name=translate(username, "\_", ".");

call symputx('USERNAME',username,'G');

call symputx('user\_name',user\_name,'G');

**run**;

**data** \_null\_;

cdate = "&SYSDATE9";

ctime = "&SYSTIME";

time\_string = cdate||"\_"||translate(ctime, "\_", ":");

call symputx('logtime', time\_string,'G');

**run**;

\*macro to indicate whether the model run must do Grubbs;

**proc** **sql** noprint;

select Grubbs into :vGrubbs

from APCPARM.GrubbsTable

;

**quit**;

\*macro to indicate whether model include Day Care Patients;

**proc** **sql** noprint;

select DC into :vDC from APCPARM.DCTable;

**quit**;

\*macro to indicate whether to use provider list supplied by the user;

**proc** **sql** noprint;

select include\_Provider into :InputProv

from APCPARM.INC\_PROVIDER

;

**quit**;

\*macro to indicate summary Stats;

**proc** **sql** noprint;

select SummaryStats into :vSummaryStats

from APCPARM.SUMMARYSTATSTABLE

;

**quit**;

\*Get SIP File ID - macro containg the fileid to get correct (period) reference cost data;

**proc** **sql** noprint;

select SIPFileID into :SIP\_FileID

from APCPARM.SIPFILEID

;

**quit**;

%put SIP\_FileID=&SIP\_FileID;

\*Get MDS\_Refdata Verson ID;

**proc** **sql** noprint;

select MDSVersionID into :MDS\_refdata\_version\_id

from APCPARM.MDSVERSIONID

;

**quit**;

\*\*\* List values for all the global macro variables \*\*\*;

%put SAS\_Session: &SAS\_Session.;

%put program\_start=&program\_start.;

%put USERNAME=&USERNAME.;

%put user\_name=&user\_name.;

%put logtime=&logtime.;

%put vGrubbs=&vGrubbs.;

%put vDC=&vDC.;

%put InputProv=&InputProv.;

%put SIP\_FileID=&SIP\_FileID.;

%put param\_HESRunID=&param\_HESRunID;

%put MDS\_refdata\_version\_id=&MDS\_refdata\_version\_id.;

%put vSummaryStats=&vSummaryStats;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* author: PID ;

\* date: 19 January 2018 ;

# \* Compliance\_rates\_for\_SDEC.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

/\*This code has been copied from a previous implementation in sql.

This code will run before the Same Day Emergency Care model is created because it is used to create the compliance rate for them.

\*/

**proc** **sql**;

create table ot as SELECT

MAX(GrouperRunID) as GrouperRunID,

MAX(RowNumber) as RowNumber,

'APC' as POD,

PROCODET as ProviderCode,

PBR\_SPELL\_ID as SpellID,

CASE

WHEN MAX(ADMIMETH) IN ('11','12','13') and MAX(CLASSPAT) = **2** and MAX(SpellLOS) = **0** THEN 'DC'

WHEN MAX(ADMIMETH) IN ('11','12','13') and ((MAX(CLASSPAT) = **3** and MAX(SpellLOS) = **0**) OR (MAX(CLASSPAT) = **4** and MAX(SpellLOS) IN(**0**,**1**))) THEN 'RE'

WHEN MAX(ADMIMETH) IN ('11','12','13') THEN 'EL'

WHEN MAX(ADMIMETH) LIKE ('2%') THEN 'EM'

WHEN MAX(CLASSPAT) = **8** OR MAX(ADMIMETH) IN ('98','99') THEN 'NA'

ELSE 'NE'

END as PODSubType,

MAX(SpellHRG) as HRG,

MAX(GPPRAC) as GPPracticeCode,

MAX(TRETSPEF) as TFC,

CASE

WHEN MAX(SpellLOS) = **0** THEN **1**

ELSE **0**

END as ZeroLengthSpell,

CASE

WHEN MAX(SpellLOS) > **0** THEN **1**

ELSE **0**

END as NonZeroLengthSpell,

MAX(SpellLOS) as Duration,

MAX(CLASSPAT) as CLASSPAT,

MAX(ADMINCAT) as ADMINCAT,

MAX(STARTAGE) as STARTAGE

FROM bptsql.groupeddata

WHERE GrouperRunID = &GrouperRunID and SpellHRG IN ('GC17K','AA31D','AA31E','HD26G','SA04K','SA04L','SA09K','SA09L','HE21F','HE21G','HE31F','HE31G','HE41C','HE41D','HE51G','HE51H','DZ15Q','DZ15R','LB16J','LB16K','JD07H','JD07J','JD07K','EB12B','EB12C','EB13C','EB13D','LB15E','LB20F','LB20G','DZ11V','YQ51C','YQ51D','YQ51E','AA26F','AA26G','AA26H','EB08E','HE11G','HE11H','DZ22Q','CB02E','CB02F','DZ09P','DZ09Q','LB40G','WH04E','EB07C','EB07D','EB07E','AA29F','LA04Q','LA04R','LA04S', 'FD01H','FD01J')

GROUP BY

PROCODET

,PBR\_SPELL\_ID;

**quit** ;

**proc** **sql**;

create table BPTOUT.ComplRateSDEC\_sql as

SELECT substr(ProviderCode,**1**,**3**) as providercode,

HRG,

sum(ZeroLengthSpell) AS CountZeroLOS,

sum(NonZeroLengthSpell) AS CountNonZeroLOS,

(sum(NonZeroLengthSpell)+sum(ZeroLengthSpell)) AS TotalActivity,

(calculated CountZeroLOS)\***100**/(calculated TotalActivity) as Perc\_SDEC

FROM ot

Where PODSubType = 'NE' or PODSubType = 'EM'

GROUP BY calculated providercode , HRG

ORDER BY HRG;

**quit**;

**proc** **sql**;

create table t1 as

select

distinct HRG,

Perc\_SDEC

from BPTOUT.ComplRateSDEC\_sql;

**quit**;

**proc** **sql**;

create table t2 as

select

HRG,

sum(CountZeroLOS) as sumCountZeroLOS,

sum(TotalActivity) as sumActivity

from BPTOUT.ComplRateSDEC\_sql

group by HRG

;

**quit**;

**proc** **univariate** data=t1 noprint;

var Perc\_SDEC;

by hrg;

output out=perc\_table1 pctlpts=**75** pctlpre=P;

**run**;

**proc** **sql**;

create table BPTout.ComplRateSDEC\_percentile as

select a.HRG,

(a.P75)/**100** as Percentile75th,

round(a.P75) as Percentile75th\_rnd,

b.sumCountZeroLOS\***100**/b.sumActivity as WeightedAvg,

round(b.sumCountZeroLOS\***100**/b.sumActivity ) as WeightedAvg\_rnd,

(b.sumCountZeroLOS\***100**/b.sumActivity )\***1.1** as WeightedAvg\_10perc,

round((b.sumCountZeroLOS\***100**/b.sumActivity )\***1.1** ) as WeightedAvg\_10perc\_rnd

from perc\_table1 as a

left join t2 as b on a.HRG=b.HRG

;

**quit**;

**proc** **export** data=BPTOUT.ComplRateSDEC\_sql

outfile="&outputpath.\&outputfile..XLSX"

dbms=XLSX replace;

sheet="ComplRateSDEC\_sql";

**run**;

**proc** **export** data=BPTout.ComplRateSDEC\_percentile

outfile="&outputpath.\&outputfile..XLSX"

dbms=XLSX replace;

sheet="ComplRateSDEC\_percentile";

**run**;

/\*--------------------------------------------------------------\*/

# \* Step3\_5\_Create\_all\_APC\_OPROC\_Excel\_Model\_worksheets\_1

/\* Version: 1.02 \*/

/\* Author: PID \*/

/\* Date: September 2017 \*/

/\*--------------------------------------------------------------\*/

/\* Description: \*/

/\* Generate an excel file which contains 5 tabs for APC results:\*/

/\* 1.STEP BY STEP OP-DC-EL-NE; \*/

/\* 2.Long Stay Payment; \*/

/\* 3.STEP BY STEP TRIMPOINT and LSP; \*/

/\* 4.QUANTUM RECONCILIATION BY Sub Chapter \*/

/\* 5.APC OPROC Model LinkSheet \*/

/\*--------------------------------------------------------------\*/

/\*ODS to create Excel workbook output with multiple sheets\*/

ods \_all\_ close;

ods tagsets.ExcelXP path="&ExcelOutFolder." file="19\_20 All Models Final Prices.xml" style=Printer;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\*\*\*\*\*\* Step 1 Run APC\_OPROC Model \*\*\*\*\*\*;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

/\*Step 1-- APC\_OPROC\_Model\_Linked\_Sheet Start from here \*/

ods tagsets.ExcelXP options(sheet\_name='1 APC & OPROC'

autofilter='NO' embedded\_titles ="yes" autofit\_height ="yes" frozen\_headers="yes" frozen\_headers="2"

FROZEN\_ROWHEADERS='3'

absolute\_column\_width='6,8,50,10,10,10,10,10,10,10,10,10,10,10');

**proc** **print** data = ModelOUt.APC\_OPROC\_MODEL\_LINKEDSHEET noobs label split= '\*';

var Year HRG\_Code HRG\_Name /style(header)=[background=#b3ffff];

var OPROC\_Tariff Combined\_DC\_EL\_Prices DC\_Prices EL\_Prices /style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###.##"];

var EL\_TRIM\_POINTS /style(header)=[background=#b3ffff];

var NE\_Prices /style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###.##"];

var NE\_TRIM\_POINTS /style(header)=[background=#b3ffff];

var EBD\_Prices /style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###.##"];

var SSEM\_applicable /style(header)=[background=#b3ffff];

var SSEM\_Pct /style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:#,##0.0%"];

var SSEM\_Price /style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###.##"];

label

HRG\_Code="HRG Code\_\_"

OPROC\_Tariff="Outpatient \*procedure \*tariff (£)\_\_"

Combined\_DC\_EL\_Prices="Combined day \*case /ordinary \*elective spell \*tariff (£)\_"

DC\_Prices="Day case \*spell tariff (£)\_\_"

EL\_Prices="Ordinary \*elective \*spell tariff (£)\_\_"

EL\_TRIM\_POINTS="Ordinary \*elective long \*stay trim point \*(days)\_\_"

NE\_Prices="Non-elective \*spell tariff (£)\_\_"

NE\_TRIM\_POINTS="Non-elective \*long stay \*trim point \*(days)\_\_"

EBD\_Prices="Per day long\* stay payment \*(for days exceeding \*trim point) (£)\_\_"

SSEM\_applicable="Reduced short \*stay \*emergency \*tariff \*applicable?\_\_"

SSEM\_Pct="% applied in \*calculation of \*reduced short \*stay \*emergency \*tariff\_\_"

SSEM\_Price="Reduced short \*stay emergency \*tariff (£)\_\_"

;

**run**;

/\*Step 1-- APC\_OPROC\_Model\_Linked\_Sheet End here \*/

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\*\*\*\*\*\* Step 2 Run OPATT Model \*\*\*\*\*\*;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

ods tagsets.ExcelXP options(sheet\_name='2 OP Attendances'

autofilter='NO' embedded\_titles ="yes" autofit\_height ="yes" frozen\_headers="yes" frozen\_headers="2"

FROZEN\_ROWHEADERS='1'

absolute\_column\_width='10,50,10,10,10,10');

**proc** **print** data = ModelOUt.OPATT\_PRICES\_1920 noobs label split= '\*';

var TFC Tariff\_TFC FAs\_prices\_1920 FAm\_prices\_1920 FUs\_prices\_1920 FUm\_prices\_1920 /style(header)=[background=#b3ffff];

;

label

TFC="Treatment function code\_\_"

Tariff\_TFC="Treatment function description\_\_"

FAs\_prices\_1920="WF01B \*First Attendance - \*Single Professional\_\_"

FAm\_prices\_1920="WF02B \*First Attendance - \*Multi Professional\_\_"

FUs\_prices\_1920="WF01A \*Follow Up Attendance - \*Single Professional\_\_"

FUm\_prices\_1920="WF02A \*Follow Up Attendance - \*Multi Professional\_\_"

;

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\*\*\*\*\*\* Step 3 Run A&E Model \*\*\*\*\*\*;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

ods tagsets.ExcelXP options(sheet\_name='3 A&E'

autofilter='NO' embedded\_titles ="yes" autofit\_height ="yes" frozen\_headers="yes" frozen\_headers="2"

FROZEN\_ROWHEADERS='1'

absolute\_column\_width='10,50,10,10');

**proc** **print** data = ModelOUt.AE\_FINAL\_PRICES noobs label split= '\*';

var HRG HRG\_name Final\_Prices\_Type1n2 Final\_Prices\_Type3 /style(header)=[background=#b3ffff];

;

label

HRG="HRG code"

HRG\_name="HRG name"

Final\_Prices\_Type1n2="Tariff (£) \*Type 1 and 2 \*Departments\_\_"

Final\_Prices\_Type3="Tariff (£) \*Type 3 \*Departments\_\_"

;

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\*\*\*\*\*\* Step 4 Run Unbundeled Model \*\*\*\*\*\*;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

ods tagsets.ExcelXP options(sheet\_name='4 Unbundled Services' sheet\_interval = 'NONE'

autofilter='NO' embedded\_titles ="yes" autofit\_height ="yes"

absolute\_column\_width='12,10,50,10,10');

Title "1.Direct access and outpatient diagnostic imaging services ";

**proc** **print** data = ModelOUt.UNB\_DI\_RD\_PRICES\_1920 noobs label split= '\*' ;

var department currency HRG\_name /style(header)=[background=#b3ffff];

var final\_prices Cost\_of\_reporting /style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###"];

label

department="Department"

currency="HRG Code"

HRG\_name="HRG Name"

final\_prices="Tariff \*(including cost \*of reporting) \*(£)\_\_"

Cost\_of\_reporting="Cost of reporting\*(£)\_\_"

;

**run**;

Title;

Title "2.Direct access and outpatient nuclear medicine services ";

**proc** **print** data = ModelOUt.UNB\_DI\_RN\_PRICES\_1920 noobs label split= '\*' ;

var department currency HRG\_name /style(header)=[background=#b3ffff];

var final\_prices Cost\_of\_reporting /style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###"];

label

department="Department"

currency="HRG Code"

HRG\_name="HRG Name"

final\_prices="Tariff \*(including cost \*of reporting) \*(£)\_\_"

Cost\_of\_reporting="Cost of reporting\*(£)\_\_"

;

**run**;

Title;

Title "3.Unbundled chemotherapy delivery ";

**proc** **print** data = ModelOUt.UNB\_CHEMO\_PRICES\_1920 noobs label split= '\*' ;

var DEPARTMENT CURRENCY HRG\_name /style(header)=[background=#b3ffff];

var final\_prices /style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###"];

label

DEPARTMENT="Department"

currency="HRG Code"

HRG\_name="HRG Name"

final\_prices="Tariff (£)"

;

**run**;

Title;

Title "4.Unbundled external beam radiotherapy ";

**proc** **print** data = ModelOut.UNB\_RADIO\_PRICES\_1920 noobs label split= '\*' ;

var DEPARTMENT CURRENCY HRG\_name /style(header)=[background=#b3ffff];

var final\_prices /style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###"];

label

DEPARTMENT="Department"

currency="HRG Code"

HRG\_name="HRG Name"

final\_prices="Tariff (£)"

;

**run**;

Title;

**data** empty;

\_=" ";

**run**;

ods tagsets.ExcelXP options(sheet\_name='Empty' sheet\_interval = 'table');

**proc** **print** data =empty noobs ;

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\*\*\*\*\*\* Step 5 Maternity Model \*\*\*\*\*\*;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

ods tagsets.ExcelXP options(sheet\_name='5 Maternity' sheet\_interval = 'NONE'

autofilter='NO' embedded\_titles ="yes" autofit\_height ="yes"

absolute\_column\_width='6,50,10,10,10,10,10,10,10,10,10,10,10,10,10');

Title " ";

Title "1. Non-delivery phases";

Title;

Title "1a. Antenatal phase";

**proc** **print** data = ModelOut.MPPLINKEDSHANTENATALFINALPRICE noobs label split= '\*' ;

var code Name /style(header)=[background=#b3ffff];

var Tariff / style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###"];

label

Tariff="Tariff (£)";

**run**;

Title "1b. Postnatal phase";

**proc** **print** data = ModelOut.MPPLINKEDSHPOSTNATALFINALPRICE noobs label split= '\*' ;

var code Name /style(header)=[background=#b3ffff];

var Tariff / style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###"];

label

Tariff="Tariff (£)";

**run**;

Title "2. Mandatory delivery admitted patient and outpatient procedure prices for maternity services";

**proc** **print** data = ModelOut.MPPLINKEDSHDELIVERYHRGFPRICES noobs label split= '\*' ;

var level /style(header)=[background=#b3ffff];

var

Combined\_DC\_EL\_NE /style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###,###"];

var

Day\_Case\_Tariff

Ordinary\_Elective\_Tariff

Ordinary\_elective\_LS\_trimpoint /style(header)=[background=#b3ffff];

var

Non\_elective\_tariff /style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###,###"];

var

Non\_elective\_LS\_trimpoint /style(header)=[background=#b3ffff];

var Long\_stay\_payment /style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###,###"];

var

Reduced\_SSE\_tariff\_Applic

Pct\_Reduced\_SSE\_tariff\_Applic

Reduced\_SSE\_tariff

/style(header)=[background=#b3ffff];

label

level="Level"

Combined\_DC\_EL\_NE="Combined day \*case / ordinary \*elective spell \*tariff (£)"

Day\_Case\_Tariff="Day case spell \*tariff (£)"

Ordinary\_Elective\_Tariff="Ordinary elective spell tariff (£)"

Ordinary\_elective\_LS\_trimpoint="Ordinary elective \*long stay trimpoint \*(days)"

Non\_elective\_tariff="Non-elective \*spell tariff (£)"

Non\_elective\_LS\_trimpoint="Non-elective \*long stay \*trimpoint (days)"

Long\_stay\_payment="Per day long \*stay payment \*(for days exceeding \*trimpoint) (£)"

Reduced\_SSE\_tariff\_Applic="Reduced short \*stay \*emergency \*tariff \*applicable?"

Pct\_Reduced\_SSE\_tariff\_Applic="% applied in \*calculation of \*reduced short \*stay \*emergency \*tariff "

Reduced\_SSE\_tariff="Reduced \*short stay \*emergency \*tariff (£)"

;

**run**;

Title;

ods tagsets.ExcelXP options(sheet\_name='Empty2' sheet\_interval = 'table');

**proc** **print** data =empty noobs ;

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\*\*\*\*\*\* Step 6 Run Other Mandatory Model \*\*\*\*\*\*;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

ods tagsets.ExcelXP options(sheet\_name='6 Other Mandatory' sheet\_interval = 'NONE'

autofilter='NO' embedded\_titles ="yes" autofit\_height ="yes"

absolute\_column\_width='30,20,40,10,10,10,10,10,10,10,10,10,10,10');

Title "Other mandatory prices";

Title "1.Direct access services";

**proc** **print** data = ModelOut.OM\_DIRECT\_ACCESS noobs label split= '\*' ;

var Type HRG\_Code HRG\_Name

/style(header)=[background=#b3ffff];

var Tariff /style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###"];

label

HRG\_Code="HRG code"

HRG\_Name="HRG\_Code name"

Tariff="Tariff (£)\_\_";

**run**;

Title "2.Rehabilitation post discharge";

**proc** **print** data = ModelOUt.OM\_REHAB\_POST\_DISCHARGE noobs label split= '\*' ;

var Description /style(header)=[background=#b3ffff];

var Tariff /style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###"];

label

Tariff="Tariff (£)";

**run**;

Title "3.Cystic fibrosis";

**proc** **print** data = ModelOUt.OM\_CYSTIC\_FIBROSIS noobs label split= '\*' ;

var Band /style(header)=[background=#b3ffff];

var Tariff /style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###"];

label

Tariff="Tariff (£)";

**run**;

Title "4.Looked after children's health assessments";

**proc** **print** data = ModelOUt.OM\_CHILDREHEALTHASSESSMENT noobs label split= '\*' ;

var Task /style(header)=[background=#b3ffff];

var Tariff /style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###"];

label

Tariff="Tariff (£)";

**run**;

**quit**;

Title;

/\*Empty table 3 \*/

ods tagsets.ExcelXP options(sheet\_name='Empty3' sheet\_interval = 'table');

**proc** **print** data =empty noobs ;

**run**;

Title ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\*\*\*\*\*\* Step 7 Run APC\_OPROC\_BPT Model \*\*\*\*\*\*;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

ods tagsets.ExcelXP options(sheet\_name='APC\_OPROC\_Annex\_A'

autofilter='NO' embedded\_titles ="yes" autofit\_height ="yes" frozen\_headers="yes" frozen\_headers="2"

FROZEN\_ROWHEADERS='1'

absolute\_column\_width='8,50,10,10,10,10,10,10,10,10,10,10,10,10,20,20,20,20,8');

Title "Admitted patient care & outpatient procedure prices 2019/20";

**proc** **print** data = ModelOUt.APC\_OPROC\_BPT\_ANNEXT\_A noobs label split= '\*';

var HRG\_Code HRG\_Name /style(header)=[background=#b3ffff];

var OPROC\_Tariff Combined\_DC\_EL\_Prices DC\_Prices EL\_Prices /style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###"];

var EL\_TRIM\_POINTS /style(header)=[background=#b3ffff];

var NE\_Prices /style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###"];

var NE\_TRIM\_POINTS /style(header)=[background=#b3ffff];

var EBD\_Prices /style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###"];

var SSEM\_applicable /style(header)=[background=#b3ffff];

var SSEM\_Pct /style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:#,##0.0%"];

var SSEM\_Price /style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###"];

var BPT\_HRGsubHRG/style(header)=[background=#b3ffff];

var BPT\_area/style(header)=[background=#b3ffff];

var POD\_BPT/style(header)=[background=#b3ffff];

var SUS\_BPT\_nonBPT/style(header)=[background=#b3ffff];

var Price\_information/style(header)=[background=#b3ffff];

var BPT\_flag/style(header)=[background=#b3ffff];

;

label

HRG\_Code="HRG Code"

OPROC\_Tariff="Outpatient \*procedure \*tariff (£)"

Combined\_DC\_EL\_Prices="Combined day \*case /ordinary \*elective spell \*tariff (£)"

DC\_Prices="Day case \*spell tariff (£)"

EL\_Prices="Ordinary \*elective \*spell tariff (£)"

EL\_TRIM\_POINTS="Ordinary \*elective long \*stay trim point \*(days)"

NE\_Prices="Non-elective \*spell tariff (£)"

NE\_TRIM\_POINTS="Non-elective \*long stay \*trim point \*(days)"

EBD\_Prices="Per day long\* stay payment \*(for days exceeding \*trim point) (£)"

SSEM\_applicable="Reduced short \*stay \*emergency \*tariff \*applicable?"

SSEM\_Pct="% applied in \*calculation of \*reduced short \*stay \*emergency \*tariff"

SSEM\_Price="Reduced short \*stay emergency \*tariff (£)"

BPT\_HRGsubHRG=" BPT applies to HRG \*or sub-HRG level"

BPT\_area="Area BPT Name applies\*(see also tab '6a.BPTs')"

POD\_BPT="Where BPT applies:\*NE = Non-elective spell tariff \*DC/EL = Day case/ordinary elective spell tariff

\*OP = OPROC tariff"

SUS\_BPT\_nonBPT="SUS will automate which \*BPT price(BPT or \*non-BPT price)"

Price\_information="The price automated by SUS"

BPT\_flag="BPT Flag"

;

**run**;

Title;

ods tagsets.ExcelXP close;

ods listing;

**data** P\_Finihsed;

Congratulations="Excel report is Ready";

Location="\\irnarch\sas\_data\Tariff Rebuild\Models for demo\TC models\Final\_Output\_Excel\ ";

Filename="19\_20 All Models Final Prices.xml.xml";

**run**;

**proc** **sql**;

Title "Congratulations ! Excel report is ready Now!! ";

Title1 "Excel report location:\\irnarch\sas\_data\Tariff Rebuild\Models for demo\TC models\Final\_Output\_Excel\ ";

Title1 "File Name:19\_20 All Models Final Prices.xml.";

Title "Have a nice day!!";

Title1;

Title;

select \* from P\_Finihsed

;

**quit**;

# \* Step3\_5\_Create\_all\_APC\_OPROC\_Excel\_Model\_worksheets\_2

/\* Version: 1.02 \*/

/\* Author: PID \*/

/\* Date: September 2017 \*/

/\*--------------------------------------------------------------\*/

/\* Description: \*/

/\* Generate an excel file which contains 5 tabs for APC results:\*/

/\* 1.STEP BY STEP OP-DC-EL-NE; \*/

/\* 2.Long Stay Payment; \*/

/\* 3.STEP BY STEP TRIMPOINT and LSP; \*/

/\* 4.QUANTUM RECONCILIATION BY Sub Chapter \*/

/\* 5.APC OPROC Model LinkSheet \*/

/\*--------------------------------------------------------------\*/

/\*ODS to create Excel workbook output with multiple sheets\*/

ods \_all\_ close;

ods tagsets.ExcelXP path="&ExcelOutFolder." file="19\_20 All Models Prices Step by Step.xml" style=Printer;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\*\*\*\*\*\* Step 1 Run APC\_OPROC Model \*\*\*\*\*\*;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

/\*Step 1\_1-- STEP BY STEP OP-DC-EL-NE Start from here \*/

/\*Create and format STEP BY STEP OP-DC-EL-NE tab\*/

ods tagsets.ExcelXP options(sheet\_name='1\_1.APC&OPROC OP-DC-EL-NE'

suppress\_bylines='yes'

frozen\_headers='1'

FROZEN\_ROWHEADERS='4'

autofit\_height='Yes'

/\*PID ADDED SIX 10's B4 15\*/ absolute\_column\_width='6, 6, 7, 50, 8, 8, 8, 7, 8, 7, 7, 10, 10, 10,10, 10, 10, 10, 10, 10, 10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,14,14,14,14'

pages\_fitwidth='1'

pages\_fitheight='9'

FitToPage="yes"

autofilter='1-70');

/\*Define STEP BY STEP OP-DC-EL-NE tab output variables\*/

**proc** **print** data=Modelout.STEP\_BY\_STEP\_OP\_DC\_EL\_NE split='\*' n noobs;

var Spell\_HRG SpellHRGChapter SpellHRGSubchapter HRG\_Name OPROC\_currency\_applicable

/ style(Column)=[background=#e6ffff tagattr='format:#,##'] style(header)=[background=#e6ffff];

var P0\_OPROC P0\_DC P0\_EL P0\_NE / style(Column)=[background=#e6e6ff ] style(header)=[background=#e6e6ff];

var P1\_OPROC P1\_DC P1\_EL P1\_NE / style(Column)=[background=#c2c2a3 tagattr="format:###,###,###.##"] style(header)=[background=#c2c2a3];

var QR1/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.0000%"] style(header)=[background=#FFFF00] ;

var P3\_OPROC P3\_DC P3\_EL P3\_NE / style(Column)=[background=#a5bcd9 tagattr="format:###,###,###.##"] style(header)=[background=#a5bcd9];

var Cost\_base\_adjustment/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00];

var P5\_OPROC P5\_DC P5\_EL P5\_NE / style(Column)=[background=#d1e2b2 tagattr="format:###,###,###.##"] style(header)=[background=#d1e2b2];

var P6\_OPROC P6\_DC P6\_EL P6\_NE / style(Column)=[background=#ffe6e6 tagattr="format:###,###,###.##"] style(header)=[background=#ffe6e6];

var P7\_OPROC P7\_DC P7\_EL P7\_NE / style(Column)=[background=#b2d7e2 tagattr="format:###,###,###.##"] style(header)=[background=#b2d7e2];

var P8\_OPROC P8\_DC P8\_EL P8\_NE / style(Column)=[background=#c0b2e2 tagattr="format:###,###,###.##"] style(header)=[background=#c0b2e2];

var P9\_QR2/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.0000%"] style(header)=[background=#FFFF00];

var P10\_OPROC P10\_DC P10\_EL P10\_NE / style(Column)=[background=#AFEEEE tagattr="format:###,###,###.##"] style(header)=[background=#AFEEEE ];

var SMF/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%" ] style(header)=[background=#FFFF00];

var P12\_OPROC P12\_DC P12\_EL P12\_NE / style(Column)=[background=#E9967A tagattr="format:###,###,###.##"] style(header)=[background=#E9967A];

var P13\_QR3/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%" ] style(header)=[background=#FFFF00];

var P14\_OPROC P14\_DC P14\_EL P14\_NE / style(Column)=[background=#87CEFA tagattr="format:###,###,###.##"] style(header)=[background=#87CEFA];

var P15\_SCF / style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00];

var P16\_OPROC P16\_DC P16\_EL P16\_NE / style(Column)=[background=#c2c2a4 tagattr="format:###,###,###.##"] style(header)=[background=#c2c2a4];

/\* PID ADDED RE CASH IN CASH OUT STARTS HERE \*/

var cash\_io\_factor\_ne / style(Column)=[background=#FFFF00 tagattr="format:#,###.##"] style(header)=[background=#FFFF00];

var cash\_io\_factor\_all / style(Column)=[background=#FFFF00 tagattr="format:#,###.##"] style(header)=[background=#FFFF00];

var P16\_OPROC\_CIO P16\_DC\_CIO P16\_EL\_CIO P16\_NE\_CIO / style(Column)=[background=#c2c2a4 tagattr="format:###,###,###.##"] style(header)=[background=#c2c2a4];

/\* PID ADDED RE CASH IN CASH OUT ENDS HERE \*/

var P17\_Inflation /style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00];

var P18\_OPROC P18\_DC P18\_EL P18\_NE / style(Column)=[background=#ffe6e9 tagattr="format:###,###,###.##" ] style(header)=[background=#ffe6e9];

var P19\_Efficiency/style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00];

var P20\_OPROC P20\_DC P20\_EL P20\_NE / style(Column)=[background=#b8dc6f tagattr="format:###,###,###.##" ] style(header)=[background=#b8dc6f];

var P21\_19\_20\_CNST/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00];

var P22\_OPROC P22\_DC P22\_EL P22\_NE / style(Column)=[background=#ccffff tagattr="format:###,###,###.##" ] style(header)=[background=#ccffff];

label

Spell\_HRG="Spell\*HRG\_\_"

SpellHRGChapter ="Spell HRG\*chapter\_\_"

SpellHRGSubchapter="Spell HRG\*sub - \*chapter\_\_"

HRG\_Name="HRG\*name"

OPROC\_currency\_applicable="OPROC\*currency\*applicable\_\_"

P0\_OPROC="P0\*Exception flag:\*1 = HRG manual adj, 0 = no manual adj \*OPROC Tariff\_\_"

P0\_DC = "P0\*Exception flag:\*1 = HRG manual adj, 0 = no manual adj \*DC Tariff\_\_"

P0\_EL = "P0\*Exception flag:\*1 = HRG manual adj, 0 = no manual adj \*EL Tariff\_\_"

P0\_NE= "P0\*Exception flag:\*1 = HRG manual adj, 0 = no manual adj \*NE Tariff\_\_"

P1\_OPROC="P1\*SQL\_Out\*OPROC Tariff\_\_"

P1\_DC = "P1\*SQL\_Output\*DC Tariff\_\_"

P1\_EL = "P1\*SQL\_Output\*EL Tariff\_\_"

P1\_NE= "P1\*SQL\_Output\*NE Tariff\_\_"

QR1="P2\*Reference costs quantum reconciliation factor (QR1)\_\_"

P3\_OPROC="P3\*Modelled prices after QR1 factor\*OPROC Tariff\_\_"

P3\_DC = "P3\*Modelled prices after QR1 factor\*DC Tariff\_\_"

P3\_EL = "P3\*Modelled prices after QR1 factor\*EL Tariff\_\_"

P3\_NE= "P3\*Modelled prices after QR1 factor\*NE Tariff\_\_"

Cost\_base\_adjustment="P4\*Cost base adjustment (CB)\_\_"

P5\_OPROC="P5\*Prices after CB\*implementation\*OPROC Tariff\_\_"

P5\_DC = "P5\*Prices after CB\*implementation\*DC Tariff\_\_"

P5\_EL = "P5\*Prices after CB\*implementation\*EL Tariff\_\_"

P5\_NE = "P5\*Prices after CB\*implementation\*NE Tariff\_\_"

P6\_OPROC="P6\*Prices after inflation, \*efficiency and CNST adjustments \*to a 18/19 financial year basis\*OPROC Tariff\_\_"

P6\_DC = "P6\*Prices after inflation, \*efficiency and CNST adjustments \*to a 18/19 financial year basis\*DC Tariff\_\_"

P6\_EL = "P6\*Prices after inflation, \*efficiency and CNST adjustments \*to a 18/19 financial year basis\*EL Tariff\_\_"

P6\_NE = "P6\*Prices after inflation, \*efficiency and CNST adjustments \*to a 18/19 financial year basis\*NE Tariff\_\_"

P7\_OPROC="P7\*Manual Adjustments\*OPROC Tariff\_\_"

P7\_DC = "P7\*Manual Adjustments\*DC Tariff\_\_"

P7\_EL = "P7\*Manual Adjustments\*EL Tariff\_\_"

P7\_NE = "P7\*Manual Adjustments\*NE Tariff\_\_"

P8\_OPROC="P8\*Prices before QR2 implementation\*OPROC Tariff\_\_"

P8\_DC = "P8\*Prices before QR2 implementation\*DC Tariff\_\_"

P8\_EL = "P8\*Prices before QR2 implementation\*EL Tariff\_\_"

P8\_NE = "P8\*Prices before QR2 implementation\*NE Tariff\_\_"

P9\_QR2="P9\*Manual adjustments quantum reconciliation factor (QR2)\_\_"

P10\_OPROC="P10\*Prices after QR2 implementation\*OPROC Tariff\_\_"

P10\_DC = "P10\*Prices after QR2 implementation\*DC Tariff\_\_"

P10\_EL = "P10\*Prices after QR2 implementation\*EL Tariff\_\_"

P10\_NE = "P10\*Prices after QR2 implementation\*NE Tariff\_\_"

SMF="P11\*Smoothing Factor adj (SMF)\_\_"

P12\_OPROC="P12\*Prices after SMF adj\*OPROC Tariff\_\_"

P12\_DC = "P12\*Prices after SMF adj\*DC Tariff\_\_"

P12\_EL = "P12\*Prices after SMF adj\*EL Tariff\_\_"

P12\_NE = "P12\*Prices after SMF adj\*NE Tariff\_\_"

P13\_QR3="P13\*Final quantum reconciled prices (QR3)\_\_"

P14\_OPROC="P14\*Prices after QR3 implementation\*OPROC Tariff\_\_"

P14\_DC = "P14\*Prices after QR3 implementation\*DC Tariff\_\_"

P14\_EL = "P14\*Prices after QR3 implementation\*EL Tariff\_\_"

P14\_NE = "P14\*Prices after QR3 implementation\*NE Tariff\_\_"

P15\_SCF="P15\*Scaling Factor Adj (SCF)\_\_"

P16\_OPROC="P16\*Final prices before 18/19 cost uplift assumptions\*OPROC Tariff\_\_"

P16\_DC = "P16\*Final prices before 18/19 cost uplift assumptions\*DC Tariff\_\_"

P16\_EL = "P16\*Final prices before 18/19 cost uplift assumptions\*EL Tariff\_\_"

P16\_NE = "P16\*Final prices before 18/19 cost uplift assumptions\*NE Tariff\_\_"

/\* PID ADDED COLUMNS FOR CASH\_IN\_CASH\_OUT STARTS HERE \*/

cash\_io\_factor\_ne="P16\*Cash in Cash out factor\*Non Elective only\_\_"

cash\_io\_factor\_all="P16\*Cash in Cash out factor\*All APC\_OPROC PODS\_\_"

P16\_OPROC\_CIO= "P16\_CIO\*Final prices after Cash in Cash out\*before 18/19 cost uplift assumptions\*OPROC Tariff\_\_"

P16\_DC\_CIO= "P16\_CIO\*Final prices after Cash in Cash out\*before 18/19 cost uplift assumptions\*DC Tariff\_\_"

P16\_EL\_CIO= "P16\_CIO\*Final prices after Cash in Cash out\*before 18/19 cost uplift assumptions\*EL Tariff\_\_"

P16\_NE\_CIO= "P16\_CIO\*Final prices after Cash in Cash out\*before 18/19 cost uplift assumptions\*NE Tariff\_\_"

/\* PID ADDED COLUMNS FOR CASH\_IN\_CASH\_OUT ENDS HERE \*/

P17\_Inflation="P17\*19/20 Inflation\_\_"

P18\_OPROC="P18\*Final Prices in 19/20 prices including inflation\*OPROC Tariff\_\_"

P18\_DC = "P18\*Final Prices in 19/20 prices including inflation\*DC Tariff\_\_"

P18\_EL = "P18\*Final Prices in 19/20 prices including inflation\*EL Tariff\_\_"

P18\_NE = "P18\*Final Prices in 19/20 prices including inflation\*NE Tariff\_\_"

P19\_Efficiency="P19\*19/20 Efficiency Factor\_\_"

P20\_OPROC="P20\*Final Prices in 19/20 prices, including inflation and efficiency assumptions\*OPROC Tariff\_\_"

P20\_DC = "P20\*Final Prices in 19/20 prices, including inflation and efficiency assumptions\*DC Tariff\_\_"

P20\_EL = "P20\*Final Prices in 19/20 prices, including inflation and efficiency assumptions\*EL Tariff\_\_"

P20\_NE = "P20\*Final Prices in 19/20 prices, including inflation and efficiency assumptions\*NE Tariff\_\_"

P21\_19\_20\_CNST="P21\*19/20 CNST\_\_"

P22\_OPROC="P22\*FINAL PRICES in 19/20 prices, including inflation, efficiency and CNST assumptions\*OPROC Tariff\_\_"

P22\_DC = "P22\*FINAL PRICES in 19/20 prices, including inflation, efficiency and CNST assumptions\*DC Tariff\_\_"

P22\_EL = "P22\*FINAL PRICES in 19/20 prices, including inflation, efficiency and CNST assumptions\*EL Tariff\_\_"

P22\_NE = "P22\*FINAL PRICES in 19/20 prices, including inflation, efficiency and CNST assumptions\*NE Tariff\_\_"

;

**run**;

/\*Step 1\_1-- STEP BY STEP OP-DC-EL-NE End here \*/

/\*Step 1\_2-- Long Stay Payment Start from here \*/

ods tagsets.ExcelXP options(sheet\_name='1\_2.APC&OPROC Long Stay Payment'

autofilter='NO' embedded\_titles ="yes" autofit\_height ="yes" frozen\_headers="yes" frozen\_headers="2"

absolute\_column\_width='6,50,6,8,8,8,3,10,10,10,10,10,10');

**proc** **print** data = Modelout.Long\_Stay\_Payment noobs label split= '\*';

var HRG HRG\_Name Chapter Proportion\_Children\_Activity Congenital\_HRG\_Flag AG18\_YearsandUnder\_HRG\_Flag LU /style(header)=[background=#ecffb3];

var CLEAN\_EBD\_ACT / style(header)=[background=#ecffb3] style(Column)=[ tagattr="format:###,###,###"] ;

var CLEAN\_EBD\_TC LSP\_UC Limit\_within\_100\_to\_500 / style(header)=[background=#ecffb3] style(Column)=[ tagattr="format:###,###,###.##"] ;

var Zero\_price\_HRG / style(header)=[background=#ecffb3] ;

var Long\_Stay\_Payment / style(header)=[background=#ecffb3] style(Column)=[ tagattr="format:###,###,###.##"];

;

label

HRG="HRG \_\_"

HRG\_Name="HRG Name\_\_"

Chapter="Chapter \_\_"

Proportion\_Children\_Activity="Proportion\*of children\*activity\_\_"

Congenital\_HRG\_Flag="Congenital \*HRG flag\_\_"

AG18\_YearsandUnder\_HRG\_Flag="18 years \*and under \*HRG flag\_\_"

LU="LU\_\_"

CLEAN\_EBD\_ACT="CLEAN EBD \*ACT \_\_"

CLEAN\_EBD\_TC="CLEAN EBD \*TC \_\_"

LSP\_UC="LSP UC\_\_"

Limit\_within\_100\_to\_500="Limiting within \*£100 to £500 \_\_"

Zero\_price\_HRG="Zero priced HRG \*flag (1 if the HRG is zero-priced, 0 \*otherwise)\_\_"

Long\_Stay\_Payment="Long Stay Payments\_\_"

;

**run**;

/\*Step 1\_2-- Long Stay Payment End here \*/

/\*Step 1\_3-- STEP BY STEP TRIMPOINT and LSP------ Start from here \*/

/\*Create and format STEP BY STEP TRIMPOINT and LSP tab\*/

ods tagsets.ExcelXP options(sheet\_name='1\_3.APC&OPROC TRIMPOINT and LSP'

suppress\_bylines='yes'

frozen\_headers='1'

FROZEN\_ROWHEADERS='4'

autofit\_height='Yes'

absolute\_column\_width='6, 6, 7, 50, 8, 8, 8, 7, 8, 7, 7, 10, 10, 10,10, 10, 10, 10, 10, 10, 10,10,10,10'

pages\_fitwidth='1'

pages\_fitheight='9'

FitToPage="yes"

autofilter='1-66');

/\*Define STEP BY STEP TRIMPOINT and LSP tab output variables\*/

**proc** **print** data=Modelout.STEP\_BY\_STEP\_TRIMPOINT\_and\_LSP split='\*' n noobs;

var Spell\_HRG SpellHRGChapter SpellHRGSubChapter HRG\_Description / style(Column)=[background=#e6ffff tagattr='format:#,##'] style(header)=[background=#e6ffff];

/\*Part1\*/

var EL\_Trim NE\_Trim Long\_Stay\_Payment / style(Column)=[background=#ffb699 /\*tagattr='format:#,##'\*/] style(header)=[background=#ffb699];

/\*Part2 \*/

var EL\_Trim\_5DRule NE\_Trim\_5DRule / style(Column)=[background=#ffb6c1 ] style(header)=[background=#ffb6c1];

/\*Part 3\*/

var Manual\_TP\_EL Manual\_TP\_NE / style(Column)=[background=#cfe2cf] style(header)=[background=#cfe2cf] ;

/\*Part 4 \*/

var DC\_EL\_SemiAuto NE\_SemiAuto / style(Column)=[background=#dabcf6 ] style(header)=[background=#dabcf6];

/\*Part 5 \*/

var Fin\_EL Fin\_NE / style(Column)=[background=#afeeee ] style(header)=[background=#afeeee];

/\*Part 6 \*/

var Initial\_longStayPayment / style(Column)=[background=#e6ffff ] style(header)=[background=#e6ffff];

/\*Part 7 \*/

var LSP\_after\_QR1 / style(Column)=[background=#fff5e6 ] style(header)=[background=#fff5e6];

/\*Part 8 \*/

var LSP\_after\_CB / style(Column)=[background=#88cc00 ] style(header)=[background=#88cc00];

/\*Part 9 \*/

var LSP\_Uplift18\_19 / style(Column)=[background=#efdcc3 ] style(header)=[background=#efdcc3];

/\* var LSP\_SCF/ style(Column)=[background=#efdcc3 ] style(header)=[background=#efdcc3];\*/

/\*Part 10 \*/

var LSP\_QR2/ style(Column)=[background=#ffe6e9 ] style(header)=[background=#ffe6e9];

/\*Part 11 \*/

var LSP\_SMF/ style(Column)=[background=#bbd0f7] style(header)=[background=#bbd0f7 ];

/\*Part 12 \*/

var LSP\_QR3 / style(Column)=[background=#9fc69f] style(header)=[background=#9fc69f];

/\*Part 13 \*/

var LSP\_SCF / style(Column)=[background=#c69f6c] style(header)=[background=#c69f6c];

/\*Part 14 \*/

var LSP\_1920Basis / style(Column)=[background=#dcd8f3 ] style(header)=[background=#dcd8f3];

label

Spell\_HRG="Spell\*HRG\_\_"

SpellHRGChapter="Spell\*HRG\*chapter\_\_"

SpellHRGSubChapter="Spell HRG\*subchapter\_\_"

HRG\_Description="HRG description"

EL\_Trim="P1-SQL output\*EL\_Trim\_\_"

NE\_Trim="P1-SQL output\*NE\_Trim\_\_"

Long\_Stay\_Payment="P1-SQL output\*Long\_Stay\_Payment\_\_"

EL\_Trim\_5DRule="P2-Minimum 5 day rule\*EL\_Trim\_\_"

NE\_Trim\_5DRule="P2-Minimum 5 day rule\*NE\_Trim\_\_"

Manual\_TP\_EL="P3-2019/20 Manual\*EL\_Trim\_\_"

Manual\_TP\_NE="P3-2019/20 Manual\*NE\_Trim\_\_"

DC\_EL\_SemiAuto="P4-2019/20\*Semi-Automated\*EL\_Trim\_\_"

NE\_SemiAuto="P4-2019/20\*Semi-Automated\*NE\_Trim\_\_"

Fin\_EL="P5-Final trim points\*EL\_Trim\_\_"

Fin\_NE="P5-Final trim points\*NE\_Trim\_\_"

Initial\_longStayPayment="P6-Initial long stay\*payments\*(LSP)\_\_"

LSP\_after\_QR1="P7-LSP after QR1\*implementation\_\_"

LSP\_after\_CB="P8-Long stay \*payments after CB adjustment\*implementation\_\_"

LSP\_Uplift18\_19="P9-Modelled LSP\*prices adjusted to \*a 18/19 basis\_\_"

LSP\_QR2="P10-Long stay payments\*after QR2 \*implementation \*adjustment \*implementation\_\_"

LSP\_SMF="P11-Long stay payments\* after SMF\*adjustment \*implementation\_\_"

LSP\_QR3="P12-Long stay\*payments after\* QR3\*implementation\_\_"

LSP\_SCF="P13-Long stay\* payments after\* SCF\* implementation\_\_"

LSP\_1920Basis="P14-Long stay\*payments on a 19/20 \*basis (applying inflation, \*efficiency \*and CNST)\_\_"

/\*LSP\_SCF="Long stay\*payments after\* Scaling Factor\* implementation\_\_"\*/

;

**run**;

/\*Step 1\_3-- STEP BY STEP TRIMPOINT and LSP----------------- End here \*/

/\*Step 1\_4-- QUANTUM RECONCILIATION BY Sub Chapter-------------------------------- Start from here \*/

/\*Create and format QUANTUM RECONCILIATION BY Sub Chapter tab\*/

ods tagsets.ExcelXP options(sheet\_name='1\_4.APC&OPROC QUANTUM RECONCILIATION BY SubChapter'

suppress\_bylines='yes'

frozen\_headers='1'

FROZEN\_ROWHEADERS='2'

autofit\_height='Yes'

absolute\_column\_width='6, 50, 7, 7, 8, 8, 8, 7, 8, 7, 7, 10, 10, 10,10, 10, 10, 10, 10, 10, 10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10,'

pages\_fitwidth='1'

pages\_fitheight='9'

FitToPage="yes"

autofilter='1-170');

/\*Define STEP BY STEP QUANTUM RECONCILIATION BY Sub Chapter variables\*/

**proc** **print** data=Modelout.QUANTUM\_RECONCILIATION\_BySubChap split='\*' n noobs;

var HRG Description OPROC\_currency\_applicable / style(Column)=[background=#e6ffff tagattr='format:#,##'] style(header)=[background=#e6ffff];

var P0\_OPROC P0\_DC P0\_EL P0\_NE / style(header)=[background=#eda891];

var P1\_OPROC\_TA P1\_DC\_TA P1\_EL\_TA P1\_NE\_TA P1\_NE\_AT\_less\_SSEM P1\_LSP\_days P1\_SSEM / style(header)=[background=#eae395] style(Column)=[ tagattr="format:###,###,###"];

/\* Part 2 \*/

var P2\_OPROC P2\_DC P2\_EL P2\_NE P2\_LSP / style(header)=[background=#f0e6ff] style(Column)=[ tagattr="format:###,###,###.##"];

var P2\_SSEM\_Banding P2\_SSEM\_Pct/style(header)=[background=#f0e6ff];

var P2\_SSEM\_Tariff / style(header)=[background=#f0e6ff] style(Column)=[ tagattr="format:###,###,###.##"];

/\* Part 3 \*/

var P3\_Sub\_Chapter /style(header)=[background=#f0e6ff];

var P3\_OPROC\_quantum\_by\_HRG P3\_APC\_Quantum\_by\_HRG /style(header)=[background=#f0e6ff] style(Column)=[ tagattr="format:###,###,###.##"];

var P3\_Sub\_\_Chapter /style(header)=[background=#f0e6ff] ;

var P3\_OPROC\_quantum\_by\_SubChapter P3\_APC\_quantum\_by\_SubChapter P3\_APC\_OPROC\_quantum\_by\_chapter /style(header)=[background=#f0e6ff] style(Column)=[ tagattr="format:###,###,###.##"];

/\* Part 4 \*/

var P4\_OPROC P4\_DC P4\_EL P4\_NE P4\_Long\_Stay\_Payment /style(header)=[background=#ffe6e9] style(Column)=[ tagattr="format:###,###,###.##"];

var P4\_SSEM\_Banding P4\_SSEM\_Pct /style(header)=[background=#ffe6e9];

var P4\_SSEM\_Tariff /style(header)=[background=#ffe6e9] style(Column)=[ tagattr="format:###,###,###.##"];

/\*Part 5 \*/

var P5\_OPROC P5\_DC P5\_EL P5\_NE P5\_Long\_Stay\_Payment /style(header)=[background=#**99e600**] style(Column)=[ tagattr="format:###,###,###.##"];

var P5\_SSEM\_Banding P5\_SSEM\_Pct /style(header)=[background=#**99e600**];

var P5\_SSEM\_Tariff /style(header)=[background=#**99e600**] style(Column)=[ tagattr="format:###,###,###.##"];

/\*Part 6 \*/

var P6\_Sub\_Chapter /style(header)=[background=#ffe6e9];

var P6\_OPROC\_Quantum\_by\_HRG P6\_APC\_Quantum\_by\_HRG P6\_Sub\_\_Chapter P6\_OPROC\_quantum\_by\_SubChapter

P6\_APC\_quantum\_by\_SubChapter P6\_APC\_OPROC\_quantum\_by\_chapter /style(header)=[background=#ffe6e9] style(Column)=[ tagattr="format:###,###,###.##"];

/\*Part 7 \*/

var P7\_Sub\_Chapter /style(header)=[background=#**99e600**];

var P7\_OPROC\_Quantum\_by\_HRG P7\_APC\_Quantum\_by\_HRG /style(header)=[background=#**99e600**] style(Column)=[ tagattr="format:###,###,###.##"];

var P7\_Sub\_\_Chapter /style(header)=[background=#**99e600**];

var P7\_OPROC\_quantum\_by\_SubChapter P7\_APC\_quantum\_by\_SubChapter P7\_APC\_OPROC\_quantum\_by\_chapter /style(header)=[background=#**99e600**] style(Column)=[ tagattr="format:###,###,###.##"];

/\*Part 8 \*/

var P8\_Sub\_Chapter /style(header)=[background=#e6ffff];

var P8\_OPROC\_Quantum\_by\_HRG P8\_APC\_Quantum\_by\_HRG /style(header)=[background=#e6ffff] style(Column)=[ tagattr="format:###,###,###.##"];

var P8\_Sub\_\_Chapter /style(header)=[background=#e6ffff] ;

var P8\_OPROC\_quantum\_by\_SubChapter P8\_APC\_quantum\_by\_SubChapter P8\_APC\_OPROC\_quantum\_by\_chapter

/style(header)=[background=#e6ffff] style(Column)=[ tagattr="format:###,###,###.##"];

/\*Part 9 \*/

var P9\_Chapter P9\_Chapter\_Sum P9\_APC\_OPROC\_QR2 /style(header)=[background=#**669900**];

/\*Part 10 \*/

var P10\_HRG P10\_Chapter /style(header)=[background= #936c39];

var P10\_APCOPROC\_subchapter\_QR2 P10\_OPROC P10\_DC P10\_EL P10\_NE P10\_Long\_Stay\_Payment /style(header)=[background= #936c39] style(Column)=[ tagattr="format:###,###,###.##"];

var P10\_SSEM\_Banding P10\_SSEM\_Pct /style(header)=[background= #936c39];

var P10\_SSEM\_Tariff /style(header)=[background= #936c39] style(Column)=[ tagattr="format:###,###,###.##"];

/\*Part 11 \*/

var P11\_Sub\_Chapter /style(header)=[background= #936c39];

var P11\_OPROC\_Quantum\_by\_HRG P11\_APC\_Quantum\_by\_HRG P11\_APC\_OPROC\_Quantum\_by\_HRG /style(header)=[background= #936c39] style(Column)=[ tagattr="format:###,###,###.##"];

var P11\_Sub\_\_Chapter /style(header)=[background= #936c39];

var P11\_OPROC\_quantum\_by\_SubChapter P11\_APC\_quantum\_by\_SubChapter P11\_APC\_OPROC\_quantum\_by\_chapter/style(header)=[background= #936c39] style(Column)=[ tagattr="format:###,###,###.##"];

/\*Part 112 \*/

var P112\_HRG P112\_Chapter /style(header)=[background=#ffcc99];

var P112\_OPROC P112\_DC P112\_EL P112\_NE P112\_Long\_Stay\_Payment /style(header)=[background=#ffcc99] style(Column)=[ tagattr="format:###,###,###.##"];

var P112\_SSEM\_Banding P112\_SSEM\_Pct /style(header)=[background=#ffcc99];

var P112\_SSEM\_Tariff /style(header)=[background=#ffcc99] style(Column)=[ tagattr="format:###,###,###.##"];

/\*Part 12 \*/

var P12\_Sub\_Chapter /style(header)=[background=#ffcc99];

var P12\_OPROC\_Quantum\_by\_HRG P12\_APC\_Quantum\_by\_HRG P12\_APC\_OPROC\_Quantum\_by\_HRG /style(header)=[background=#ffcc99] style(Column)=[ tagattr="format:###,###,###.##"];

var P12\_Sub\_\_Chapter /style(header)=[background=#ffcc99];

var P12\_OPROC\_quantum\_by\_SubChapter

P12\_APC\_quantum\_by\_SubChapter P12\_APC\_OPROC\_quantum\_by\_chapter/style(header)=[background=#ffcc99] style(Column)=[ tagattr="format:###,###,###.##"];

/\*Part 13 \*/

var P13\_Chapter P13\_Chapter\_Sum P13\_APC\_OPROC\_QR3 /style(header)=[background=#5bd75b];

/\*Part 14 \*/

var P14\_HRG P14\_Chapter P14\_APCOPROC\_subchapter\_QR3/style(header)=[background=#e6ffff];

var P14\_OPROC P14\_DC P14\_EL P14\_NE P14\_Long\_Stay\_Payment /style(header)=[background=#e6ffff] style(Column)=[ tagattr="format:###,###,###.##"];

var P14\_SSEM\_Banding P14\_SSEM\_Pct /style(header)=[background=#e6ffff];

var P14\_SSEM\_Tariff /style(header)=[background=#e6ffff] style(Column)=[ tagattr="format:###,###,###.##"];

/\*Part 15 \*/

var P15\_Sub\_Chapter /style(header)=[background=#ffe6e9];

var P15\_OPROC\_Quantum\_by\_HRG P15\_APC\_Quantum\_by\_HRG P15\_APC\_OPROC\_Quantum\_by\_HRG /style(header)=[background=#ffe6e9] style(Column)=[ tagattr="format:###,###,###.##"];

var P15\_Sub\_\_Chapter /style(header)=[background=#ffe6e9];

var P15\_OPROC\_quantum\_by\_SubChapter

P15\_APC\_quantum\_by\_SubChapter P15\_APC\_OPROC\_quantum\_by\_chapter /style(header)=[background=#ffe6e9] style(Column)=[ tagattr="format:###,###,###.##"];

/\*Part 16 \*/

var P16\_HRG P16\_Chapter/style(header)=[background=#cce6ff];

var P16\_OPROC P16\_DC P16\_EL P16\_NE P16\_Long\_Stay\_Payment /style(header)=[background=#cce6ff] style(Column)=[ tagattr="format:###,###,###.##"];

var P16\_SSEM\_Banding P16\_SSEM\_Pct /style(header)=[background=#cce6ff];

var P16\_SSEM\_Tariff /style(header)=[background=#cce6ff] style(Column)=[ tagattr="format:###,###,###.##"];

/\*Part 17 \*/

var P17\_Sub\_Chapter/style(header)=[background=#f2ffcc];

var P17\_OPROC\_Quantum\_by\_HRG P17\_APC\_Quantum\_by\_HRG P17\_APC\_OPROC\_Quantum\_by\_HRG/style(header)=[background=#f2ffcc] style(Column)=[ tagattr="format:###,###,###.##"];

var P17\_Sub\_\_Chapter/style(header)=[background=#f2ffcc];

var P17\_OPROC\_quantum\_by\_SubChapter

P17\_APC\_quantum\_by\_SubChapter P17\_APC\_OPROC\_quantum\_by\_chapter P17\_APC\_Inlier\_Quantum /style(header)=[background=#f2ffcc] style(Column)=[ tagattr="format:###,###,###.##"];

var P172\_Sub\_Chapter/style(header)=[background=#f2ffcc];

var P172\_OPROC\_QuantSubchap\_eCNST P172\_APC\_QuantSubchap\_inlier P172\_APC\_OPROC\_QuanSubchap\_Inl/style(header)=[background=#f2ffcc] style(Column)=[ tagattr="format:###,###,###.##"];

/\*Part 18 \*/

var P18\_HRG P18\_Chapter/style(header)=[background=#d5f6f6];

var P18\_OPROC P18\_DC P18\_EL P18\_NE P18\_Long\_Stay\_Payment/style(header)=[background=#d5f6f6] style(Column)=[ tagattr="format:###,###,###.##"];

var P18\_SSEM\_Banding P18\_SSEM\_Pct/style(header)=[background=#d5f6f6];

var P18\_SSEM\_Tariff/style(header)=[background=#d5f6f6] style(Column)=[ tagattr="format:###,###,###.##"];

/\*Part 19 \*/

var P19\_Sub\_Chapter/style(header)=[background=#aed75b];

var P19\_OPROC\_Quantum\_by\_HRG P19\_APC\_Quantum\_by\_HRG P19\_APC\_OPROC\_Quantum\_by\_HRG/style(header)=[background=#aed75b] style(Column)=[ tagattr="format:###,###,###.##"];

var P19\_Sub\_\_Chapter/style(header)=[background=#aed75b];

var P19\_OPROC\_quantum\_by\_SubChapter

P19\_APC\_quantum\_by\_SubChapter P19\_APC\_OPROC\_quantum\_by\_chapter/style(header)=[background=#aed75b] style(Column)=[ tagattr="format:###,###,###.##"];

label

Description="HRG description"

P0\_OPROC="P0\*HRGs eligible for QR2\*1 = eligible, \*0 = not eligible\* OPROC\_"

P0\_DC = "P0\*HRGs eligible for QR2\*1 = eligible, \*0 = not eligible\* DC \_\_"

P0\_EL = "P0\*HRGs eligible for QR2\*1 = eligible, \*0 = not eligible\* EL \_\_"

P0\_NE= "P0\*HRGs eligible for QR2\*1 = eligible, \*0 = not eligible\* NE \_\_"

P1\_OPROC\_TA="P1\*Total activity (spells)\*OPROC\_"

P1\_DC\_TA="P1\*Total activity (spells)\*DC \_\_"

P1\_EL\_TA="P1\*Total activity (spells)\*EL \_\_"

P1\_NE\_TA="P1\*Total activity (spells)\*NE \_\_"

P1\_NE\_AT\_less\_SSEM="P1\*NE activities less SSEM\* (this only applies for HRGs that \*are eligible for\* SSEM\_)"

P1\_LSP\_days="P1\*LSP days \_\_"

P1\_SSEM="P1\*SSEM activities\_\_"

P2\_OPROC="P2\*Modelled prices pre \*quantum reconciliation (QR1)\*OPROC \_"

P2\_DC="P2\*Modelled prices pre \*quantum reconciliation (QR1)\*DC \_"

P2\_EL="P2\*Modelled prices pre \*quantum reconciliation (QR1)\*EL \_"

P2\_NE="P2\*Modelled prices pre \*quantum reconciliation (QR1)\*NE \_"

P2\_LSP="P2\*Modelled prices pre \*quantum reconciliation (QR1)\*LSP \_"

P2\_SSEM\_Banding="P2\*Modelled prices pre \*quantum reconciliation (QR1)\*SSEM\_Banding\_"

P2\_SSEM\_Pct="P2\*Modelled prices pre \*quantum reconciliation (QR1)\*SSEM % \_\_"

P2\_SSEM\_Tariff="P2\*Modelled prices pre \*quantum reconciliation (QR1)\*SSEM Tariff \_\_"

P3\_Sub\_Chapter="P3\*Total quantum by \*subchapter pre quantum \*reconciliation (QR1)\*Sub Chapter \_"

P3\_OPROC\_quantum\_by\_HRG="P3\*Total quantum by \*subchapter pre quantum \*reconciliation (QR1)\*OPROC quantum by HRG \_"

P3\_APC\_Quantum\_by\_HRG="P3\*Total quantum by \*subchapter pre quantum \*reconciliation (QR1)\*APC quantum by HRG \_"

P3\_Sub\_\_Chapter="P3\*Total quantum by \*subchapter pre quantum \*reconciliation (QR1)\*Sub-Chapter \_"

P3\_OPROC\_quantum\_by\_SubChapter="P3\*Total quantum by \*subchapter pre quantum \*reconciliation (QR1)\* OPROC quantum by chapter\_"

P3\_APC\_quantum\_by\_SubChapter="P3\*Total quantum by \*subchapter pre quantum \*reconciliation (QR1)\*APC quantum by chapter\_"

P3\_APC\_OPROC\_quantum\_by\_chapter="P3\*Total quantum by \*subchapter pre quantum \*reconciliation (QR1)\*APC and OPROC \*quantum by chapter\_"

P4\_OPROC="P4\*Prices post inflation,\*efficiency and CNST adjustments\*(prices before manual\* adjustments)\*OPROC \_"

P4\_DC="P4\*Prices post inflation,\*efficiency and CNST adjustments\*(prices before manual\* adjustments)\*DC \_"

P4\_EL="P4\*Prices post inflation,\*efficiency and CNST adjustments\*(prices before manual\* adjustments)\*EL \_"

P4\_NE="P4\*Prices post inflation,\*efficiency and CNST adjustments\*(prices before manual\* adjustments)\*NE \_"

P4\_Long\_Stay\_Payment="P4\*Prices post inflation,\*efficiency and CNST adjustments\*(prices before manual\* adjustments)\*Long Stay Payment \_"

P4\_SSEM\_Banding="P4\*Prices post inflation,\*efficiency and CNST adjustments\*(prices before manual\* adjustments)\*SSEM Banding \_"

P4\_SSEM\_Pct="P4\*Prices post inflation,\*efficiency and CNST adjustments\*(prices before manual\* adjustments)\*SSEM % \_"

P4\_SSEM\_Tariff="P4\*Prices post inflation,\*efficiency and CNST adjustments\*(prices before manual\* adjustments)\*SSEM Tariff \_\_"

P5\_OPROC="P5\*Prices after \*manual adjustments\* adjustments)\*OPROC \_"

P5\_DC="P5\*Prices after \*manual adjustments\* adjustments)\*DC \_"

P5\_EL="P5\*Prices after \*manual adjustments\* adjustments)\*EL \_"

P5\_NE="P5\*Prices after \*manual adjustments\* adjustments)\*NE \_"

P5\_Long\_Stay\_Payment="P5\*Prices after \*manual adjustments\* adjustments)\*Long Stay Payment \_"

P5\_SSEM\_Banding="P5\*Prices after \*manual adjustments\* adjustments)\*SSEM Banding \_"

P5\_SSEM\_Pct="P5\*Prices after \*manual adjustments\* adjustments)\*SSEM % \_"

P5\_SSEM\_Tariff="P5\*Prices after \*manual adjustments\* adjustments)\*SSEM Tariff\_\_"

P6\_Sub\_Chapter="P6\*Total quantum by subchapter\*before manual adjustment\*Sub\_Chapter"

P6\_OPROC\_Quantum\_by\_HRG="P6\*Total quantum by subchapter\*before manual adjustment\*OPROC Quantum by HRG\_\_"

P6\_APC\_Quantum\_by\_HRG="P6\*Total quantum by subchapter\*before manual adjustment\*APC Quantum by HRG\_\_"

P6\_Sub\_\_Chapter="P6\*Total quantum by subchapter\*before manual adjustment\*Subchapter\_\_"

P6\_OPROC\_quantum\_by\_SubChapter="P6\*Total quantum by subchapter\*before manual adjustment\*OPROC quantum\*by chapter\_\_"

P6\_APC\_quantum\_by\_SubChapter="P6\*Total quantum by subchapter\*before manual adjustment\*APC quantum by chapter\_\_"

P6\_APC\_OPROC\_quantum\_by\_chapter="P6\*Total quantum by subchapter\*before manual adjustment\*APC and OPROC\*quantum by chapter\_\_"

P7\_Sub\_Chapter="P7\*Total quantum by\*subchapter after\*manual adjustment\*Sub-Chapter\_\_"

P7\_OPROC\_Quantum\_by\_HRG="P7\*Total quantum by\*subchapter after\*manual adjustment\*OPROC Quantum by HRG\_\_"

P7\_APC\_Quantum\_by\_HRG="P7\*Total quantum by\*subchapter after\*manual adjustment\*APC Quantum by HRG\_\_"

P7\_Sub\_\_Chapter="P7\*Total quantum by\*subchapter after\*manual adjustment\*Sub\_Chapter\_\_"

P7\_OPROC\_quantum\_by\_SubChapter="P7\*Total quantum by\*subchapter after\*manual adjustment\*OPROC quantum\* by chapter\_\_"

P7\_APC\_quantum\_by\_SubChapter="P7\*Total quantum by\*subchapter after\*manual adjustment\*APC quantum\* by chapter\_\_"

P7\_APC\_OPROC\_quantum\_by\_chapter="P7\*Total quantum by\*subchapter after\*manual adjustment\*APC and OPROC\*quantum by chapter\_\_"

P8\_Sub\_Chapter="P8\*Total quantum by\*subchapter after manual\*adjustment excluding HRGs\*that incorporate\*OPATT prices\*Sub-Chapter\_\_"

P8\_OPROC\_Quantum\_by\_HRG="P8\*Total quantum by\*subchapter after manual\*adjustment excluding HRGs\*that incorporate\*OPROC Quantum by HRG\_\_"

P8\_APC\_Quantum\_by\_HRG="P8\*Total quantum by\*subchapter after manual\*adjustment excluding HRGs\*that incorporate\*APC Quantum by HRG\_\_"

P8\_Sub\_\_Chapter="P8\*Total quantum by\*subchapter after manual\*adjustment excluding HRGs\*that incorporate\*OPATT prices\*Sub\_chapter\_\_"

P8\_OPROC\_quantum\_by\_SubChapter="P8\*Total quantum by\*subchapter after manual\*adjustment excluding HRGs\*that incorporate\*OPATT prices\*OPROC quantum\*by chapter\_\_"

P8\_APC\_quantum\_by\_SubChapter="P8\*Total quantum by\*subchapter after manual\*adjustment excluding HRGs\*that incorporate\*OPATT prices\*APC quantum by chapter\_\_"

P8\_APC\_OPROC\_quantum\_by\_chapter="P8\*Total quantum by\*subchapter after manual\*adjustment excluding HRGs\*that incorporate\*OPATT prices\*APC and OPROC\*quantum by chapter\_\_"

P9\_Chapter="P9\*Quantum reconciliation\*by chapter (QR2)\*Chapter \_"

P9\_Chapter\_Sum="P9\*Quantum reconciliation\*by chapter (QR2)\*Chapter sum\_\_"

P9\_APC\_OPROC\_QR2="P9\*Quantum reconciliation\*by chapter (QR2)\*APC and OPROC\*QR2 \_\_"

P10\_HRG="P10\*Prices after quantum\*reconciliation by\*chapter (QR2)\*HRG \_\_"

P10\_Chapter="P10\*Prices after quantum\*reconciliation by\*chapter (QR2)\*Chapter \_\_"

P10\_APCOPROC\_subchapter\_QR2="P10\*Prices after quantum\*reconciliation by\*chapter (QR2)\*APC and OPROC\*subchapter QR2 factor\_\_"

P10\_OPROC="P10\*Prices after quantum\*reconciliation by\*chapter (QR2)\*OPROC \_\_"

P10\_DC="P10\*Prices after quantum\*reconciliation by\*chapter (QR2)\*DC \_\_"

P10\_EL="P10\*Prices after quantum\*reconciliation by\*chapter (QR2)\*EL \_\_"

P10\_NE="P10\*Prices after quantum\*reconciliation by\*chapter (QR2)\*NE \_\_"

P10\_Long\_Stay\_Payment="P10\*Prices after quantum\*reconciliation by\*chapter (QR2)\*Long Stay Payment\_\_"

P10\_SSEM\_Banding="P10\*Prices after quantum\*reconciliation by\*chapter (QR2)\*SSEM Banding\_\_"

P10\_SSEM\_Pct="P10\*Prices after quantum\*reconciliation by\*chapter (QR2)\*SSEM % \_"

P10\_SSEM\_Tariff="P10\*Prices after quantum\*reconciliation by\*chapter (QR2)\*SSEM Tariff\_\_"

P11\_Sub\_Chapter="P11\*Total quantum by\*subchapter after quantum\*adjustment by \*chapter (QR2)-Subchapter\_\_"

P11\_OPROC\_Quantum\_by\_HRG="P11\*Total quantum by\*subchapter after quantum\*adjustment by \*chapter (QR2)-OPROC Quantum\*by HRG\_\_"

P11\_APC\_Quantum\_by\_HRG="P11\*Total quantum by\*subchapter after quantum\*adjustment by \*chapter (QR2)-APC Quantum\*by HRG\_\_"

P11\_APC\_OPROC\_Quantum\_by\_HRG="P11\*Total quantum by\*subchapter after quantum\*adjustment by \*chapter (QR2)-APC and OPROC\*Quantum by HRG\_\_"

P11\_Sub\_\_Chapter="P11\*Total quantum by\*subchapter after quantum\*adjustment by \*chapter (QR2)-Sub\_chapter\_\_"

P11\_OPROC\_quantum\_by\_SubChapter="P11\*Total quantum by\*subchapter after quantum\*adjustment by \*chapter (QR2)-OPROC Quantum\*by subchapter\_\_"

P11\_APC\_quantum\_by\_SubChapter="P11\*Total quantum by\*subchapter after quantum\*adjustment by \*chapter (QR2)-APC Quantum\*by subchapter\_\_"

P11\_APC\_OPROC\_quantum\_by\_chapter="P11\*Total quantum by\*subchapter after quantum\*adjustment by \*chapter (QR2)-OPROC and APC \*Quantum\*by subchapter\_\_"

P112\_HRG="P112\*Prices after \*implementing SMF\*HRG \_\_"

P112\_Chapter="P112\*Prices after \*implementing SMF\*Chapter \_\_"

P112\_OPROC="P112\*Prices after \*implementing SMF\*OPROC \_\_"

P112\_DC="P112\*Prices after \*implementing SMF\*DC \_\_"

P112\_EL="P112\*Prices after \*implementing SMF\*EL \_\_"

P112\_NE ="P112\*Prices after \*implementing SMF\*NE \_\_"

P112\_Long\_Stay\_Payment="P112\*Prices after \*implementing SMF\*Long Stay Payment \_\_"

P112\_SSEM\_Banding="P112\*Prices after \*implementing SMF\*SSEM Banding \_\_"

P112\_SSEM\_Pct="P112\*Prices after \*implementing SMF\*SSEM % \_\_"

P112\_SSEM\_Tariff="P112\*Prices after \*implementing SMF\*SSEM Tariff\_\_"

P12\_Sub\_Chapter="P12\*Total quantum \*by chapter after \*implementing SMF\*Sub-chapter"

P12\_OPROC\_Quantum\_by\_HRG="P12\*Total quantum \*by chapter after \*implementing SMF\*OPROC Quantum by HRG\_\_"

P12\_APC\_Quantum\_by\_HRG="P12\*Total quantum \*by chapter after \*implementing SMF\*APC Quantum by HRG\_\_"

P12\_APC\_OPROC\_Quantum\_by\_HRG="P12\*Total quantum \*by chapter after \*implementing SMF\*OPROC and APC \*Quantum by HRG\_\_"

P12\_Sub\_\_Chapter="P12\*Total quantum \*by chapter after \*implementing SMF\*Subchapter"

P12\_OPROC\_quantum\_by\_SubChapter="P12\*Total quantum \*by chapter after \*implementing SMF\*OPROC Quantum \*by subchapter"

P12\_APC\_quantum\_by\_SubChapter="P12\*Total quantum \*by chapter after \*implementing SMF\*APC Quantum \*by subchapter"

P12\_APC\_OPROC\_quantum\_by\_chapter="P12\*Total quantum \*by chapter after \*implementing SMF\*APC and OPROC\*Quantum \*by subchapter"

P13\_Chapter="P13\*Quantum reconciliation\*by chapter (QR3)\*-Chapter \_\_"

P13\_Chapter\_Sum="P13\*Quantum reconciliation\*by chapter (QR3)\*-Chapter Sum\_\_"

P13\_APC\_OPROC\_QR3="P13\*Quantum reconciliation\*by chapter (QR3)\*-APC and OPROC QR3\_\_"

P14\_HRG="P14\*Prices after implementing QR3\*HRG\_\_"

P14\_Chapter="P14\*Prices after implementing QR3\*Chapter\_\_"

P14\_APCOPROC\_subchapter\_QR3="P14\*Prices after implementing QR3\*APC and OPROC\*subchapter QR3 factor\_\_"

P14\_OPROC="P14\*Prices after implementing QR3\*OPROC\_\_"

P14\_DC="P14\*Prices after implementing QR3\*DC\_\_"

P14\_EL="P14\*Prices after implementing QR3\*EL\_\_"

P14\_NE="P14\*Prices after implementing QR3\*NE\_\_"

P14\_Long\_Stay\_Payment="P14\*Prices after implementing QR3\*Long Stay Payment\_\_"

P14\_SSEM\_Banding="P14\*Prices after implementing QR3\*SSEM Banding\_\_"

P14\_SSEM\_Pct="P14\*Prices after implementing QR3\*SSEM % \_"

P14\_SSEM\_Tariff="P14\*Prices after implementing QR3\*SSEM Tariff\_\_"

P15\_Sub\_Chapter="P15\*Total quantum by \*chapter after \*implementing QR3\*-Subchapter\_\_"

P15\_OPROC\_Quantum\_by\_HRG="P15\*Total quantum by \*chapter after \*implementing QR3\*OPROC Quantum by HRG\_\_"

P15\_APC\_Quantum\_by\_HRG="P15\*Total quantum by \*chapter after \*implementing QR3\*APC Quantum by HRG\_\_"

P15\_APC\_OPROC\_Quantum\_by\_HRG="P15\*Total quantum by \*chapter after \*implementing QR3\*APC and OPROC\*Quantum by HRG\_\_"

P15\_Sub\_\_Chapter="P15\*Total quantum by \*chapter after \*implementing QR3\*-Sub\_chapter\_\_"

P15\_OPROC\_quantum\_by\_SubChapter="P15\*Total quantum by \*chapter after \*implementing QR3\*OPROC Quantum \*by subchapter\_\_"

P15\_APC\_quantum\_by\_SubChapter="P15\*Total quantum by \*chapter after \*implementing QR3\*APC Quantum\*by subchapter\_\_"

P15\_APC\_OPROC\_quantum\_by\_chapter="P15\*Total quantum by \*chapter after \*implementing QR3\*APC and OPROC\*Quantum\*by subchapter\_\_"

P16\_HRG="P16\*Prices after Scaling\*HRG\_\_"

P16\_Chapter="P16\*Prices after Scaling\*Chapter\_\_"

P16\_OPROC="P16\*Prices after Scaling\*OPROC\_\_"

P16\_DC="P16\*Prices after Scaling\*DC\_\_"

P16\_EL="P16\*Prices after Scaling\*EL\_\_"

P16\_NE="P16\*Prices after Scaling\*NE\_\_"

P16\_Long\_Stay\_Payment="P16\*Prices after Scaling\*Long Stay Payment\_\_"

P16\_SSEM\_Banding="P16\*Prices after Scaling\*SSEM Banding\_\_"

P16\_SSEM\_Pct="P16\*Prices after Scaling\*SSEM % \_"

P16\_SSEM\_Tariff="P16\*Prices after Scaling\*SSEM Tariff \_"

P17\_Sub\_Chapter="P17\*Total quantum by chapter\*after implementing scaling\*Subchapter\_\_"

P17\_OPROC\_Quantum\_by\_HRG="P17\*Total quantum by chapter\*after implementing scaling\*OPROC Quantum by HRG\_"

P17\_APC\_Quantum\_by\_HRG="P17\*Total quantum by chapter\*after implementing scaling\*APC Quantum by HRG\_"

P17\_APC\_OPROC\_Quantum\_by\_HRG="P17\*Total quantum by chapter\*after implementing scaling\*APC and OPROC \*Quantum by HRG\_"

P17\_Sub\_\_Chapter="P17\*Total quantum by chapter\*after implementing scaling\*Sub-Chapter\_"

P17\_OPROC\_quantum\_by\_SubChapter="P17\*Total quantum by chapter\*after implementing scaling\*OPROC Quantum \*by subchapter\_"

P17\_APC\_quantum\_by\_SubChapter="P17\*Total quantum by chapter\*after implementing scaling\*APC Quantum \*by subchapter\_"

P17\_APC\_OPROC\_quantum\_by\_chapter="P17\*Total quantum by chapter\*after implementing scaling\*APC and OPROC Quantum \*by subchapter\_"

P17\_APC\_Inlier\_Quantum="P17\*Total quantum by chapter\*after implementing scaling\*APC and OPROC Quantum \*APC Inlier Quantum\_"

P172\_Sub\_Chapter="P172\*For CNST (Uplifts \*excluding CNST applied)\*Subchapter\_\_"

P172\_OPROC\_QuantSubchap\_eCNST="P172\*For CNST (Uplifts \*excluding CNST applied)\*OPROC Quantum by subchapter\*(Uplifted to 19/20, \*excluding CNST)\_\_"

P172\_APC\_QuantSubchap\_inlier="P172\*For CNST (Uplifts \*excluding CNST applied)\*APC Quantum by subchapter\*(Inlier Only - uplifted \*to 19/20, \*excluding CNST)\_\_"

P172\_APC\_OPROC\_QuanSubchap\_Inl="P172\*For CNST (Uplifts \*excluding CNST applied)\*APC and OPROC \*Quantum by subchapter\*(Inlier only - uplifted \*to 19/20, \*excluding CNST)\_\_"

P18\_HRG="P18\*Final prices \*after all uplifts\*HRG\_\_"

P18\_Chapter="P18\*Final prices \*after all uplifts\*Chapter\_\_"

P18\_OPROC="P18\*Final prices \*after all uplifts\*OPROC\_\_"

P18\_DC="P18\*Final prices \*after all uplifts\*DC\_\_"

P18\_EL="P18\*Final prices \*after all uplifts\*EL\_\_"

P18\_NE="P18\*Final prices \*after all uplifts\*NE\_\_"

P18\_Long\_Stay\_Payment="P18\*Final prices \*after all uplifts\*Long Stay Payment\_\_"

P18\_SSEM\_Banding="P18\*Final prices \*after all uplifts\*SSEM Banding\_\_"

P18\_SSEM\_Pct="P18\*Final prices \*after all uplifts\*SSEM % \_"

P18\_SSEM\_Tariff="P18\*Final prices \*after all uplifts\*SSEM Tariff \_"

P19\_Sub\_Chapter="P19\*Final total quantum \*by chapter after implementing\*all uplifts\*-SubChapter \_\_"

P19\_OPROC\_Quantum\_by\_HRG="P19\*Final total quantum \*by chapter after implementing\*all uplifts\*OPROC Quantum by HRG"

P19\_APC\_Quantum\_by\_HRG="P19\*Final total quantum \*by chapter after implementing\*all uplifts\*APC Quantum by HRG"

P19\_APC\_OPROC\_Quantum\_by\_HRG="P19\*Final total quantum \*by chapter after implementing\*all uplifts\*APC and OPROC\*Quantum by HRG"

P19\_Sub\_\_Chapter="P19\*Final total quantum \*by chapter after implementing\*all uplifts\*-Sub\_Chapter \_\_"

P19\_OPROC\_quantum\_by\_SubChapter="P19\*Final total quantum \*by chapter after implementing\*all uplifts\*OPROC Quantum \*by subchapter \_\_"

P19\_APC\_quantum\_by\_SubChapter="P19\*Final total quantum \*by chapter after implementing\*all uplifts\*APC Quantum \*by subchapter \_\_"

P19\_APC\_OPROC\_quantum\_by\_chapter="P19\*Final total quantum \*by chapter after implementing\*all uplifts\*APC and OPROC \*Quantum \*by subchapter \_\_"

;

**run**;

/\*Step 1\_4--QUANTUM RECONCILIATION BY Sub Chapter----------------------- End here \*/

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\*\*\*\*\*\* Step 2 Run OPATT Model \*\*\*\*\*\*;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

ods tagsets.ExcelXP options(sheet\_name='2.OPATT Price Step by Step'

suppress\_bylines='yes'

frozen\_headers='1'

FROZEN\_ROWHEADERS='2'

autofit\_height='Yes'

absolute\_column\_width='6, 30,10, 10,10, 10, 10, 10, 10, 10, 10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10,'

pages\_fitwidth='1'

pages\_fitheight='9'

FitToPage="yes"

autofilter='1-170');

/\*Define STEP BY STEP QUANTUM RECONCILIATION BY Sub Chapter variables\*/

**proc** **print** data=ModelOut.OPATT\_PRICES\_1920\_STEP\_BY\_STEP split='\*' noobs;

/\*Part0 \*/

var TFC Tariff\_TFC ;

/\*Part1 \*/

var CL\_FAS\_uc CL\_FAM\_uc CL\_FUS\_uc CL\_FUM\_uc / style(Column)=[background=#f4f4d7 tagattr='format:#,##'] style(header)=[background=#f4f4d7];

/\*Part2 \*/

var CL\_FAS\_UnitCost CL\_FAM\_UnitCost CL\_FUS\_UnitCost CL\_FUM\_UnitCost / style(Column)=[background=#ffe6ff tagattr='format:#,##'] style(header)=[background=#ffe6ff];

/\*Part 3 \*/

var CL\_FAS\_UnitCost\_PostA CL\_FAM\_UnitCost\_PostA CL\_FUS\_UnitCost\_PostA CL\_FUM\_UnitCost\_PostA/ style(Column)=[background=#eeeedd tagattr='format:#,##'] style(header)=[background=#eeeedd];

/\*Part4 \*/

var fas\_paed\_lt\_adult fam\_paed\_lt\_adult fus\_paed\_lt\_adult fum\_paed\_lt\_adult / style(Column)=[background=#ccf2ff tagattr='format:#,##'] style(header)=[background=#ccf2ff];

/\*Part5\*/

var Quantum/ style(Column)=[background=#e6ffe6 tagattr="format:###,###,###,###.###"] style(header)=[background=#e6ffe6];

var rc\_qr1\_factor/ style(Column)=[background=#ffffcc ] style(header)=[background=#ffffcc];

var FAs\_prices\_QR1 FAm\_prices\_QR1 FUs\_prices\_QR1 FUm\_prices\_QR1/ style(Column)=[background=#e0e0d1 ] style(header)=[background=#e0e0d1];

/\*Part 6 \*/

Var Cost\_base\_adjustment\_factor/ style(Column)=[background=#ffffb3 ] style(header)=[background=#ffffb3];

var FAs\_prices\_CB FAm\_prices\_CB FUs\_prices\_CB FUm\_prices\_CB/ style(Column)=[background=#ffe6e6 ] style(header)=[background=#ffe6e6];

/\*Part 7 \*/

var ie\_indextn\_adjstmt\_factors/ style(Column)=[background=#ffffb3 ] style(header)=[background=#ffffb3];

var WF01B\_FA\_Single\_PreMA WF02B\_FA\_Multi\_PreMA WF01A\_FU\_Single\_PreMA WF02A\_FU\_Multi\_PreMA/ style(Column)=[background=#dabcf6 ] style(header)=[background=#dabcf6];

/\*Part 11 \*/

var CL\_FAS\_act\_PostA CL\_FAM\_act\_PostA CL\_FUS\_act\_PostA CL\_FUM\_act\_PostA/ style(Column)=[background=#99ff99 tagattr="format:###,###,###" ] style(header)=[background=#99ff99];

/\*Part 12 \*/

var QUANTUM\_PRE\_MA/ style(Column)=[background=#e6ffe6 tagattr="format:###,###,###.##"] style(header)=[background=#e6ffe6];

var WF01B\_FA\_Single\_PostMA WF02B\_FA\_Multi\_PostMA WF01A\_FU\_Single\_PostMA WF02A\_FU\_Multi\_PostMA/ style(Column)=[background=#ffccff ] style(header)=[background=#ffccff];

var QUANTUM\_Post\_MA/ style(Column)=[background=#e6ffe6 tagattr="format:###,###,###.##"] style(header)=[background=#e6ffe6];

/\*Part 13 \*/

var ma\_qr2\_factor/ style(Column)=[background=#ffff80 ] style(header)=[background=#ffff80];

/\*Part 14 \*/

var FAs\_prices\_QR2 FAm\_prices\_QR2 FUs\_prices\_QR2 FUm\_prices\_QR2/ style(Column)=[background=#adad85 ] style(header)=[background=#adad85];

/\*Part 14 \*/

var SMOOTHING\_factor/ style(Column)=[background=#ffa64d ] style(header)=[background=#ffa64d];

var FAs\_prices\_SF FAm\_prices\_SF FUs\_prices\_SF FUm\_prices\_SF/ style(Column)=[background=#ccffcc ] style(header)=[background=#ccffcc];

/\*part 15 \*/

var smoothing\_qr3\_factor/ style(Column)=[background=#ffff80 ] style(header)=[background=#ffff80];

VAR FAs\_prices\_qr3 FAm\_prices\_qr3 FUs\_prices\_qr3 FUm\_prices\_qr3/ style(Column)=[background=#e6ccff ] style(header)=[background=#e6ccff];

/\*part 16 \*/

var Scaling\_factor / style(Column)=[background=#ffff80 ] style(header)=[background=#ffff80];

var FAs\_prices\_SCALING FAm\_prices\_SCALING FUs\_prices\_SCALING FUm\_prices\_SCALING / style(Column)=[background=#f5f5dc ] style(header)=[background=#f5f5dc];

/\*Part 17 \*/

var Inflation1920/ style(Column)=[background=#ffff80 ] style(header)=[background=#ffff80];

var FAs\_prices\_Inflation1920 FAm\_prices\_Inflation1920 FUs\_prices\_Inflation1920 FUm\_prices\_Inflation1920/ style(Column)=[background=#b3ffff] style(header)=[background=#b3ffff];

var Efficiency1920 / style(Column)=[background=#ffff80 ] style(header)=[background=#ffff80];

var FAs\_prices\_1920 FAm\_prices\_1920 FUs\_prices\_1920 FUm\_prices\_1920/ style(Column)=[background=#b3ffb3] style(header)=[background=#b3ffb3];

label

Tariff\_TFC="Treatment function name"

/\*Part 1 \*/

CL\_FAS\_uc="SQL prices \*FAS\_\_"

CL\_FAM\_uc="SQL prices \*FAM\_\_"

CL\_FUS\_uc="SQL prices \*FUS\_\_"

CL\_FUM\_uc="SQL prices \*FUM\_\_"

/\*Part2 \*/

CL\_FAS\_UnitCost="OPROC cost \*allocation \*FAS\_\_"

CL\_FAM\_UnitCost="OPROC cost \*allocation \*FAM\_\_"

CL\_FUS\_UnitCost="OPROC cost \*allocation \*FUS\_\_"

CL\_FUM\_UnitCost="OPROC cost \*allocation \*FUM\_\_"

/\*Part3 \*/

CL\_FAS\_UnitCost\_PostA="Front-loading \*FAS\_\_"

CL\_FAM\_UnitCost\_PostA="Front-loading \*FAM\_\_"

CL\_FUS\_UnitCost\_PostA="Front-loading \*FUS\_\_"

CL\_FUM\_UnitCost\_PostA="Front-loading \*FUM\_\_"

/\*Part 4 \*/

fas\_paed\_lt\_adult="Adj for low volume, \*paediatric-adult \*relativities \*FAS\_\_"

fam\_paed\_lt\_adult="Adj for low volume, \*paediatric-adult \*relativities \*FAM\_\_"

fus\_paed\_lt\_adult="Adj for low volume, \*paediatric-adult \*relativities \*FUS\_\_"

fum\_paed\_lt\_adult="Adj for low volume, \*paediatric-adult \*relativities \*FUM\_\_"

/\*Part 5\*/

Quantum="Quantum \*Pre QR1 factor\_\_"

rc\_qr1\_factor="Reference costs \*quantum \*reconciliation \*factor\*QR1\_\_"

FAs\_prices\_QR1="Prices after QR1 \*FAS\_\_"

FAm\_prices\_QR1="Prices after QR1 \*FAM\_\_"

FUs\_prices\_QR1="Prices after QR1 \*FUS\_\_"

FUm\_prices\_QR1="Prices after QR1 \*FUM\_\_"

/\*Part 6\*/

Cost\_base\_adjustment\_factor="Cost base \*adjustment factor \*CB\_\_"

FAS\_prices\_CB="Prices after \*CB adjustment \* FAS\_\_"

FAM\_prices\_CB="Prices after \*CB adjustment \* FAM\_\_"

FUS\_prices\_CB="Prices after \*CB adjustment \* FUS\_\_"

FUM\_prices\_CB="Prices after \*CB adjustment \* FUM\_\_"

/\*Part 7 \*/

ie\_indextn\_adjstmt\_factors="Indexation adjustment \*factors to 18/19 \*Inflation \*& Efficiency\_\_"

/\*no column from 7-10 \*/

WF01B\_FA\_Single\_PreMA="Pre-Manually adjusted \*Prices\*FAS\_\_"

WF02B\_FA\_Multi\_PreMA="Pre-Manually adjusted \*Prices\*FAM\_\_"

WF01A\_FU\_Single\_PreMA="Pre-Manually adjusted \*Prices\*FUS\_\_"

WF02A\_FU\_Multi\_PreMA= "Pre-Manually adjusted \*Prices\*FUM\_\_"

/\*Part 11 Activities \*/

CL\_FAS\_act\_PostA="OPATT \*Activites \*FAS\_\_"

CL\_FAM\_act\_PostA="OPATT \*Activites \*FAM\_\_"

CL\_FUS\_act\_PostA="OPATT \*Activites \*FUS\_\_"

CL\_FUM\_act\_PostA="OPATT \*Activites \*FUM\_\_"

QUANTUM\_PRE\_MA= "Quantum \*pre - Manual \*adjustment\_\_"

/\*Part 12 \*/

WF01B\_FA\_Single\_PostMA="Manually adjusted \*Prices\*FAS\_\_"

WF02B\_FA\_Multi\_PostMA ="Manually adjusted \*Prices\*FAM\_\_"

WF01A\_FU\_Single\_PostMA="Manually adjusted \*Prices\*FUS\_\_"

WF02A\_FU\_Multi\_PostMA= "Manually adjusted \*Prices\*FUM\_\_"

QUANTUM\_Post\_MA="Quantum \*post - Manual \*adjustment\_\_"

/\*Part 13 \*/

ma\_qr2\_factor="Manual adjustments \*reconciliation \*factor \*QR2\_\_"

/\*Part 14 \*/

FAs\_prices\_QR2="Prices after \*QR2 adjustment \*FAS\_\_"

FAm\_prices\_QR2="Prices after \*QR2 adjustment \*FAM\_\_"

FUs\_prices\_QR2="Prices after \*QR2 adjustment \*FUS\_\_"

FUm\_prices\_QR2="Prices after \*QR2 adjustment \*FUM\_\_"

/\*Part 14 \*/

SMOOTHING\_factor="Smoothing Factor \*SMF\_\_"

FAs\_prices\_SF="Prices after \*applying \*smoothing \*factors\*FAS\_\_"

FAm\_prices\_SF="Prices after \*applying \*smoothing \*factors\*FAM\_\_"

FUs\_prices\_SF="Prices after \*applying \*smoothing \*factors\*FUS\_\_"

FUm\_prices\_SF="Prices after \*applying \*smoothing \*factors\*FUM\_\_"

/\*Part 15 \*/

smoothing\_qr3\_factor="Smoothing quantum \*neutrality check \*QR3\_\_"

FAs\_prices\_qr3="Prices after QR3 \*(validation check, \*smoothing is quantum \*neutral so QR3 should \*be equal to 0%)\*FAS\_\_"

FAm\_prices\_qr3="Prices after QR3 \*(validation check, \*smoothing is quantum \*neutral so QR3 should \*be equal to 0%)\*FAM\_\_"

FUs\_prices\_qr3="Prices after QR3 \*(validation check, \*smoothing is quantum \*neutral so QR3 should \*be equal to 0%)\*FUS\_\_"

FUm\_prices\_qr3="Prices after QR3 \*(validation check, \*smoothing is quantum \*neutral so QR3 should \*be equal to 0%)\*FUM\_\_"

/\*Part 16 \*/

Scaling\_factor="Scaling Factor\_\_"

FAs\_prices\_SCALING="Prices after \*scaling factor\_\*FAS\_\_"

FAm\_prices\_SCALING="Prices after \*scaling factor\_\*FAM\_\_"

FUs\_prices\_SCALING="Prices after \*scaling factor\_\*FUS\_\_"

FUm\_prices\_SCALING="Prices after \*scaling factor\_\*FUM\_\_"

/\*Part 17 \*/

Inflation1920="19/20 Inflation"

FAs\_prices\_Inflation1920="Prices including \*19/20 cost \*uplift Inflation \*FAS\_\_"

FAm\_prices\_Inflation1920="Prices including \*19/20 cost \*uplift Inflation \*FAM\_\_"

FUs\_prices\_Inflation1920="Prices including \*19/20 cost \*uplift Inflation \*FUS\_\_"

FUm\_prices\_Inflation1920="Prices including \*19/20 cost \*uplift Inflation \*FUM\_\_"

Efficiency1920="19/20 Efficiency \*Factor"

FAs\_prices\_1920="Final Prices including \*19/20 cost uplifts \*Inflation & Efficiency \*FAS\_\_"

FAm\_prices\_1920="Final Prices including \*19/20 cost uplifts \*Inflation & Efficiency \*FAM\_\_"

FUs\_prices\_1920="Final Prices including \*19/20 cost uplifts \*Inflation & Efficiency \*FUS\_\_"

FUm\_prices\_1920="Final Prices including \*19/20 cost uplifts \*Inflation & Efficiency \*FUM\_\_"

;

**run**;

**quit**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\*\*\*\*\*\* Step 3 Run A&E Model \*\*\*\*\*\*;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

ods tagsets.ExcelXP options(sheet\_name='3.A&E Price Step by Step'

suppress\_bylines='yes'

frozen\_headers='1'

FROZEN\_ROWHEADERS='2'

autofit\_height='Yes'

absolute\_column\_width='6, 50, 10, 10, 10,10, 10, 10, 10, 10, 10, 10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10,'

pages\_fitwidth='1'

pages\_fitheight='9'

FitToPage="yes"

autofilter='1-28');

/\*Define STEP BY STEP QUANTUM RECONCILIATION BY Sub Chapter variables\*/

**proc** **print** data=Modelout.AE\_PRICES\_STEP\_BY\_STEP /\*split='\*'\*/ noobs;

**quit**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\* Step 4 Run Unbundeled Model

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ods tagsets.ExcelXP options(sheet\_name='4.1 UNB\_DI\_RD\_STEP\_BY\_STEP'

suppress\_bylines='yes'

frozen\_headers='1'

FROZEN\_ROWHEADERS='2'

autofit\_height='Yes'

absolute\_column\_width='10 ,6, 50, 7, 7, 8, 8, 8, 7, 8, 7, 7, 10, 10, 10,10, 10, 10, 10, 10, 10, 10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10,'

pages\_fitwidth='1'

pages\_fitheight='9'

FitToPage="yes"

autofilter='1-28');

/\*Define STEP BY STEP QUANTUM RECONCILIATION BY Sub Chapter variables\*/

**proc** **print** data=Modelout.UNB\_DI\_RD\_STEP\_BY\_STEP\_1920 /\*split='\*'\*/ noobs;

**quit**;

ods tagsets.ExcelXP options(sheet\_name='4.2 UNB\_DI\_RN\_STEP\_BY\_STEP'

suppress\_bylines='yes'

frozen\_headers='1'

FROZEN\_ROWHEADERS='2'

autofit\_height='Yes'

absolute\_column\_width='10,6, 50, 7, 7, 8, 8, 8, 7, 8, 7, 7, 10, 10, 10,10, 10, 10, 10, 10, 10, 10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10,'

pages\_fitwidth='1'

pages\_fitheight='9'

FitToPage="yes"

autofilter='1-28');

/\*Define STEP BY STEP QUANTUM RECONCILIATION BY Sub Chapter variables\*/

**proc** **print** data=Modelout.UNB\_DI\_RN\_STEP\_BY\_STEP\_1920 /\*split='\*'\*/ noobs;

**quit**;

ods tagsets.ExcelXP options(sheet\_name='4.3 UNB\_CHEMO\_STEP\_BY\_STEP'

suppress\_bylines='yes'

frozen\_headers='1'

FROZEN\_ROWHEADERS='2'

autofit\_height='Yes'

absolute\_column\_width='10,6, 50, 7, 7, 8, 8, 8, 7, 8, 7, 7, 10, 10, 10,10, 10, 10, 10, 10, 10, 10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10,'

pages\_fitwidth='1'

pages\_fitheight='9'

FitToPage="yes"

autofilter='1-33');

/\*Define STEP BY STEP QUANTUM RECONCILIATION BY Sub Chapter variables\*/

**proc** **print** data=Modelout.UNB\_CHEMO\_STEP\_BY\_STEP\_1920 /\*split='\*'\*/ noobs;

**quit**;

ods tagsets.ExcelXP options(sheet\_name='4.4 UNB\_RADIO\_STEP\_BY\_STEP'

suppress\_bylines='yes'

frozen\_headers='1'

FROZEN\_ROWHEADERS='2'

autofit\_height='Yes'

absolute\_column\_width='10, 6,50, 7, 7, 8, 8, 8, 7, 8, 7, 7, 10, 10, 10,10, 10, 10, 10, 10, 10, 10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10,'

pages\_fitwidth='1'

pages\_fitheight='9'

FitToPage="yes"

autofilter='1-35');

/\*Define STEP BY STEP QUANTUM RECONCILIATION BY Sub Chapter variables\*/

**proc** **print** data=Modelout.UNB\_RADIO\_STEP\_BY\_STEP\_1920 /\*split='\*'\*/ noobs;

**quit**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\*\*\*\*\*\* Step 5 Run Maternity Model \*\*\*\*\*\*;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

ods tagsets.ExcelXP options(sheet\_name='5.1 MPP\_DELIVERY\_PRICES\_STEPBYSTEP'

suppress\_bylines='yes'

frozen\_headers='1'

FROZEN\_ROWHEADERS='2'

autofit\_height='Yes'

absolute\_column\_width='6, 10, 10, 10, 10, 10,10, 10, 10, 10, 10, 10, 10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10,'

pages\_fitwidth='1'

pages\_fitheight='9'

FitToPage="yes"

autofilter='1-21');

/\*Define STEP BY STEP QUANTUM RECONCILIATION BY Sub Chapter variables\*/

/\*Define STEP BY STEP QUANTUM RECONCILIATION BY Sub Chapter variables\*/

**proc** **print** data=OutSAS.MPP\_DELIVERY\_IANDE1920 split='\*' noobs;

var level/style(header)=[background=#b3ffff];;

var Total\_Activity/style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###,###"];

var Total\_TC/style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###,###"];

var Unit\_Cost/style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###,###"];

var QR1\_Adjustment/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Delivery\_QR1/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var CB\_Adjustment/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Delivery\_CB / style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var Total\_I\_E\_Adjustment/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Delivery\_Total\_I\_E/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var Total\_CNST/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Uplift\_Total\_Adjust/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Delivery\_Total\_CNST/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var Prices\_Delivery\_PostMA/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var QR2\_Adjustment/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Delivery\_QR2/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var CashInOut\_Delivery\_Factor/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Delivery\_CashInOut/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var SMF/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Delivery\_SMF/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var Scaling\_Factor/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Delivery\_ScalingFact/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var CNST1920/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Delivery\_CNST1920/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var I\_E\_1920Adjustment/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Delivery\_FinalPrices/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

;

label

level="Delivery Phase\*Level"

Inlier\_Activity="APC Activity\_\_"

OPROC\_Activity="OPROC Activity\_\_"

Comm\_Activity="Community Activity\_\_"

Total\_Activity="Total Activity\_\_"

Inlier\_TC="APC Total Cost\_\_"

OPROC\_TC="OPROC Total Cost\_\_"

Comm\_TC="CommunityTotal Cost\_\_"

Total\_TC="Total Cost\_\_"

Unit\_Cost="Unit Cost\_\_"

QR1\_Adjustment="Reference costs \*reconciliation \*factor (QR1)\_\_"

Prices\_Delivery\_QR1="Prices \*after QR1 \*adjustment \*(£)\_\_"

CB\_Adjustment="Cost base \*adjustment \*(CB)\_\_"

Prices\_Delivery\_CB="Prices after \*implementing \*CB factors\_\_"

Total\_I\_E\_Adjustment="Inflation and \*Efficiency \*2017/18 and \*2018/19\_\_"

Prices\_Delivery\_Total\_I\_E="Prices after \*implementing \*Inflation \*and Efficiency factors\_\_"

Total\_CNST="Total CNST up to 18/19 price levels"

Uplift\_Total\_Adjust="Total Adjustment\_\_"

Prices\_Delivery\_Total\_CNST="Modelled prices\*adjusted to a\*18/19 base (£)\_\_"

Prices\_Delivery\_PostMA="Manually adjusted \*prices before \*QR2 adjustment\_\_"

QR2\_Adjustment="Manual \*adjustments \*reconciliation \*factor (QR2)\_\_"

Prices\_Delivery\_QR2="Prices after \*implementing QR2 \*factor\_\_"

CashInOut\_Delivery\_Factor="Cash in cash out\* adjustment factor\_\_"

Prices\_Delivery\_CashInOut="Price after \*Cash in Cash out \*adjustment\_\_"

SMF="Smoothing \*Factor \*(SMF)\_\_"

Prices\_Delivery\_SMF="Prices after \*implementing the \*Smoothing \*factor\_\_"

Scaling\_Factor="Scaling \*factor \*(SCF)\_"

Prices\_Delivery\_ScalingFact="Prices after \*implementing the \*Scaling \*factor\_\_"

CNST1920="CNST \*2019/20\_\_"

Prices\_Delivery\_CNST1920="Prices after \*implementing \*CNST \*factor\_\_"

I\_E\_1920Adjustment="Inflation and\* Efficiency \*2019/2020\_\_"

Prices\_Delivery\_FinalPrices="Price after \*applying \*prospective \*adjustments\_\_"

;

**run**;

**quit**;

ods tagsets.ExcelXP options(sheet\_name='5.2 MPP\_ANTENATAL\_PRICES\_STEPBYSTEP'

suppress\_bylines='yes'

frozen\_headers='1'

FROZEN\_ROWHEADERS='1'

autofit\_height='Yes'

absolute\_column\_width='30,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10'

pages\_fitwidth='1'

pages\_fitheight='9'

FitToPage="yes"

autofilter='1-21');

/\*Define STEP BY STEP QUANTUM RECONCILIATION BY Sub Chapter variables\*/

**proc** **print** data=ModelOut.MPP\_ANTENATAL\_IANDE1920 split='\*' noobs;

var Level / style(Column)=[background=#e6ffff ] style(header)=[background=#e6ffff];

var Act\_assumed/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var UC\_Antenatal/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var QR1\_Adjustment/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Antenatal\_QR1/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var CB\_Adjustment/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Antenatal\_CB / style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var Total\_I\_E\_Adjustment/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Antenatal\_Total\_I\_E/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var Total\_CNST/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00];

var Uplift\_Total\_Adjust/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Antenatal\_Total\_CNST/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var CashInOut\_Antenatal\_Factor/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Antenatal\_CashInOut/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var Prices\_Antenatal\_PostMA/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var QR2\_Adjustment/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Antenatal\_QR2/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var SMF/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Antenatal\_SMF/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var Scaling\_Factor/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Antenatal\_ScalingFact/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var I\_E\_1920Adjustment/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Antenatal\_FinalPrices/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

;

label

Level="Antenatal Phase"

Act\_assumed="Act Assumed\_"

UC\_Antenatal="Unit Cost \*(£)\_\_"

QR1\_Adjustment="Reference costs \*reconciliation \*factor (QR1)\_\_"

Prices\_Antenatal\_QR1="Prices \*after QR1 \*adjustment \*(£)\_\_"

CB\_Adjustment="Cost base \*adjustment \*(CB)\_\_"

Prices\_Antenatal\_CB="Prices after \*implementing \*CB factors\_\_"

Total\_I\_E\_Adjustment="Inflation and \*Efficiency \*2017/18 and \*2018/19\_\_"

Prices\_Antenatal\_Total\_I\_E="Prices after \*implementing \*Inflation \*and Efficiency factors\_\_"

Total\_CNST="Total CNST up to 18/19 price levels\_\_"

Uplift\_Total\_Adjust="Total Adjustment\_\_"

Prices\_Antenatal\_Total\_CNST="Modelled prices \*adjusted to a \*18/19 base (£)\_\_"

CashInOut\_Antenatal\_Factor="Cash in cash out\*Adjustment factor\_\_"

Prices\_Antenatal\_CashInOut="Price after \*cash in cash out \*adjustment\_\_"

Prices\_Antenatal\_PostMA="Manually adjusted \*prices before \*QR2 adjustment\_\_"

QR2\_Adjustment="Manual \*adjustments \*reconciliation \*factor (QR2)\_\_"

Prices\_Antenatal\_QR2="Prices after \*implementing QR2 \*factor\_\_"

SMF="Smoothing \*Factor \*(SMF)\_\_"

Prices\_Antenatal\_SMF="Prices after \*implementing the \*Smoothing \*factor\_\_"

Scaling\_Factor="Scaling \*factor \*(SCF)\_"

Prices\_Antenatal\_ScalingFact="Prices after \*implementing the \*Scaling \*factor\_\_"

I\_E\_1920Adjustment="Inflation and\* Efficiency \*2019/2020\_\_"

Prices\_Antenatal\_FinalPrices="Price after \*applying \*prospective \*adjustments\_\_"

;

**quit**;

ods tagsets.ExcelXP options(sheet\_name='5.3 MPP\_POSTNATAL\_PRICES\_STEPBYSTEP'

suppress\_bylines='yes'

frozen\_headers='1'

FROZEN\_ROWHEADERS='1'

autofit\_height='Yes'

absolute\_column\_width='30,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10'

pages\_fitwidth='1'

pages\_fitheight='9'

FitToPage="yes"

autofilter='1-21');

/\*Define STEP BY STEP QUANTUM RECONCILIATION BY Sub Chapter variables\*/

**proc** **print** data=ModelOut.MPP\_POSTNATAL\_IANDE1920 split='\*' noobs;

var Level / style(Column)=[background=#e6ffff ] style(header)=[background=#e6ffff];

var Act\_assumed/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var UC\_Postnatal/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var QR1\_Adjustment/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Postnatal\_QR1/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var CB\_Adjustment/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Postnatal\_CB / style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var Total\_I\_E\_Adjustment/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Postnatal\_Total\_I\_E/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var Total\_CNST/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00];

var Uplift\_Total\_Adjust/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Postnatal\_Total\_CNST/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var CashInOut\_Postnatal\_Factor/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Postnatal\_CashInOut/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var Prices\_Postnatal\_PostMA/style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var QR2\_Adjustment/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Postnatal\_QR2/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var SMF/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Postnatal\_SMF/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var Scaling\_Factor/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Postnatal\_ScalingFact/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var I\_E\_1920Adjustment/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Postnatal\_FinalPrices/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

label

Level="Postnatal Phase"

Act\_assumed="Act Assumed\_"

UC\_Postnatal="Unit Cost \*(£)\_\_"

QR1\_Adjustment="Reference costs \*reconciliation \*factor (QR1)\_\_"

Prices\_Postnatal\_QR1="Prices \*after QR1 \*adjustment \*(£)\_\_"

CB\_Adjustment="Cost base \*adjustment \*(CB)\_\_"

Prices\_Postnatal\_CB="Prices after \*implementing \*CB factors\_\_"

Total\_I\_E\_Adjustment="Inflation and \*Efficiency \*2017/18 and \*2018/19\_\_"

Prices\_Postnatal\_Total\_I\_E="Prices after \*implementing \*Inflation \*and Efficiency factors\_\_"

Total\_CNST="Total CNST up to 18/19 price levels\_\_"

Uplift\_Total\_Adjust="Total Adjustment\_\_"

Prices\_Postnatal\_Total\_CNST="Modelled prices \*adjusted to a \*18/19 base (£)\_\_"

CashInOut\_Postnatal\_Factor="Cash in cash out\*Adjustment factor\_\_"

Prices\_Postnatal\_CashInOut="Price after \*cash in cash out \*adjustment\_\_"

Prices\_Postnatal\_PostMA="Manually adjusted \*prices before \*QR2 adjustment\_\_"

QR2\_Adjustment="Manual \*adjustments \*reconciliation \*factor (QR2)\_\_"

Prices\_Postnatal\_QR2="Prices after \*implementing QR2 \*factor\_\_"

SMF="Smoothing \*Factor \*(SMF)\_\_"

Prices\_Postnatal\_SMF="Prices after \*implementing the \*Smoothing \*factor\_\_"

Scaling\_Factor="Scaling \*factor \*(SCF)\_"

Prices\_Postnatal\_ScalingFact="Prices after \*implementing the \*Scaling \*factor\_\_"

I\_E\_1920Adjustment="Inflation and\* Efficiency \*2019/2020\_\_"

Prices\_Postnatal\_FinalPrices="Price after \*applying \*prospective \*adjustments\_\_"

;

**run**;

**quit**;

ods tagsets.ExcelXP close;

ods listing;

**data** P\_Finished;

Congratulations="Excel report is Ready";

Location="\\irnarch\sas\_data\Tariff Rebuild\Models for demo\TC models\Final\_Output\_Excel\ ";

Filename="19\_20 All Models Prices Step by Step.xml";

**run**;

**proc** **sql**;

Title "Congratulations ! Excel report is ready Now!! ";

Title1 "Excel report location:\\irnarch\sas\_data\Tariff Rebuild\Models for demo\TC models\Final\_Output\_Excel\ ";

Title1 "File Name:19\_20 All Models Prices Step by Step.xml.";

Title "Have a nice day!!";

Title1;

Title;

select \* from P\_Finished

;

**quit**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* author: PID ;

\* date: 02 January 2018 ;

# \* BPT\_LinkedSheet.sas ;

\* description: 1920 ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

/\*It is creating all the tables that are used in the BPT linkedsheet\*/

/\*1. ACUTE STROKE \*/

**proc** **sql**;

create table bptout.lnk\_1a\_stroke\_table1

as select distinct

**1** as ID,

'Rapid brain imaging' as Best\_practice\_tariff\_add\_p,

Cost\_Imaging\***3** as additional\_payment

from

bptinput.BPT\_PARAMETER\_CURRENTYEAR

where Category ='Stroke'

union

select distinct

**2** as ID,

'Direct admission and 90% of spell spent in an acute stroke unit' as Best\_practice\_tariff\_add\_p,

Cost\_Stroke\***3** as additional\_payment

from

bptinput.BPT\_PARAMETER\_CURRENTYEAR

where Category ='Stroke'

union

select distinct

**3** as ID,

'Alteplase' as Best\_practice\_tariff\_add\_p,

Additional\_payment

from

BPTInput.BPT\_previousyear

where HRG\_name='Alteplase';

**quit**;

**proc** **sql**;

create table bptout.lnk\_1b\_stroke\_table2

as select

HRG\_code,

HRG\_name,

Quantum\_neutral\_nonBPT as Non\_best\_practice\_tariff,

BPT\_with\_cost\_compl as Best\_practice\_tariff,

BPT\_Flag

from

bptout.stroke;

**quit**;

/\*2. ADULT RENAL DIALYSIS\*/

/\*2A hemodialysis\*/

/\*\*/

**proc** **sql**;

create table bptout.lnk\_2a1\_renal\_hemod\_hospital\_t1

as select

a.HRG\_code,

b.HRG\_name,

round(PriceAfterMAj\_NoSmooth) as Best\_practice\_tariff

from

bptout.renal\_renalstepbystep as a

left join bptout.renal\_unbundled\_extract as b on a.hrg\_code=b.hrg\_code

where a.hrg\_code in ('LD01A','LD02A','LD03A','LD04A');

**quit**;

**proc** **sql**;

create table bptout.lnk\_2a2\_renal\_hemod\_satellite\_t2

as select

a.HRG\_code,

b.HRG\_name,

round(PriceAfterMAj\_NoSmooth) as Best\_practice\_tariff

from

bptout.renal\_renalstepbystep as a

left join bptout.renal\_unbundled\_extract as b on a.hrg\_code=b.hrg\_code

where a.hrg\_code in ('LD05A','LD06A','LD07A','LD08A');

**quit**;

**proc** **sql**;

create table bptout.lnk\_2a3\_renal\_hemod\_home\_t3

as select

a.HRG\_code,

b.HRG\_name,

round(PriceAfterMAj\_NoSmooth) as Best\_practice\_tariff

from

bptout.renal\_renalstepbystep as a

left join bptout.renal\_unbundled\_extract as b on a.hrg\_code=b.hrg\_code

where a.hrg\_code in ('LD09A','LD10A');

**quit**;

/\*2b peritoneal dialysis\*/

**proc** **sql**;

create table bptout.lnk\_2b\_renal\_peritoneal\_t4

as select

a.HRG\_code,

b.HRG\_name,

round(PriceAfterMAj\_NoSmooth) as Price

from

bptout.renal\_renalstepbystep as a

left join bptout.renal\_unbundled\_extract as b on a.hrg\_code=b.hrg\_code

where a.hrg\_code in ('LD11A','LD12A','LD13A');

**quit**;

/\*3. DAY CASE \*/

**proc** **sql**;

create table bptout.lnk\_3\_day\_case

as select distinct

a.HRG\_code,

a.HRG\_name,

b.Surgical\_subspeciality, Category,

case when b.Procedure= 'Excision\*' then 'Axillary clearance' else b.Procedure end as Procedure,

BPTprice\_DC as Best\_practice\_day\_case\_tariff,

nonBPTprice\_EL as Elective\_nonbestpractice\_tariff,

HRG\_or\_subHRG as HRG\_or\_sub\_HRG\_level,

a.BPT\_Flag

from bptout.daycase as a

left join

bptinput.bpt\_parameter\_currentyear as b on a.HRG\_code=b.HRG\_code

where ( calculated Procedure <> 'Mastectomy' and a.BPT\_flag <>'BP28'

or ( calculated Procedure <> 'Axillary clearance' and a.BPT\_flag <>'BP32') )and Category='Daycase'

;

**quit**;

/\*4. Diabetic ketoacidosis and hypoglycaemia\*/

**proc** **sql**;

create table bptout.lnk\_4\_diabete

as select

a.HRG\_code,

a.HRG\_name,

round(a.BPT\_Tariff) as BPT\_Non\_elective\_tariff,

round(a.BPT\_Tariff\*b.SSEM\_pct) as BPT\_Reduced\_SSEM\_tariff,

round(a.nonBPT\_Tariff) as Non\_BPT\_Non\_elective\_tariff,

round(a.nonBPT\_Tariff\*b.SSEM\_pct) as Non\_BPT\_Reduced\_SSEM\_tariff,

HRG\_or\_subHRG as HRG\_or\_sub\_HRG\_level,

BPT\_Flag,

b.SSEM\_pct as Percentage\_reduction\_SSEM\_tariff

from bptout.group1 as a

left join

bptinput.apc\_prices as b on a.HRG\_code=b.HRG\_code

where substr(a.HRG\_code,**1**,**2**) in ('KB');

**quit**;

/\*5. Early inflammatory arthritis\*/

**proc** **sql**;

create table bptout.lnk\_5\_earlyinflammatoryarthritis

as select 'n/a' as code,

'Early inflammatory arthritis 6 standards of care' as HRG\_name,

TopUp\_Value as Additional\_payment

from BPTout.EIA\_table\_2

;

**quit**;

/\*6. Enndoscopy procedure\*/

**proc** **sql**;

create table bptout.lnk\_6\_endoscopy\_procedure

as select HRG\_code as code,

HRG\_name,

BPT\_L1 as Lev\_1\_Best\_practice\_tariff,

L2\_IntBPT as Lev\_2\_Interm\_Bestpractice\_tariff,

nonBPT\_Tariff\_L3 as Lev\_3\_Non\_best\_practice\_tariff,

HRG\_or\_subHRG as HRG\_or\_sub\_HRG\_level,

BPT\_flag

from BPTout.endoscopy;

**quit**;

/\*7. FHF\*/

**proc** **sql**;

create table bptout.lnk\_7\_Fragility\_hip\_fracture

as select HRG\_code as code,

HRG\_name,

QuantumNeutralNonBPTprice as Non\_best\_practice\_tariff,

BPTprice\_inclcomplcost as Best\_practice\_tariff,

BPTprice\_inclcomplcost-QuantumNeutralNonBPTprice as Additional\_payment,

HRG\_or\_subHRG as HRG\_or\_sub\_HRG\_level,

BPT\_flag

from BPTout.fhf;

**quit**;

/\*8. Major trauma\*/

**proc** **sql**;

create table lnk\_8\_Major\_Trauma

as select

HRG\_name as Level,

BPT\_current\_year as Best\_practice\_tariff

from BPTout.group2

where substr(HRG\_name,**1**,**3**) in ('Lev')

order by HRG\_name;

**quit**;

**proc** **sql**;

create table bptout.lnk\_8\_Major\_Trauma

as select

monotonic() as ID,

Level,

Best\_practice\_tariff

from lnk\_8\_Major\_Trauma

where substr(Level,**1**,**3**) in ('Lev')

order by Level;

**quit**;

/\*9. Outpatients\*/

**proc** **sql**;

create table bptout.lnk\_9\_Outpatients

as select

a.HRG\_code,

a.HRG\_name,

b.Procedure,

a.BPTprice\_OPROC as Best\_pract\_outpatient\_tariff,

a.NonBPT\_minusAbsDiff as Nonbest\_pract\_dc\_ord\_el\_tariff,

a.HRG\_or\_subHRG as HRG\_or\_sub\_HRG\_level,

a.BPT\_flag as BPT\_flag

/\*\*/

from BPTout.outpatients as a

left join BPTInput.Bpt\_parameter\_currentyear as b on a.HRG\_code=b.HRG\_code

;

**quit**;

/\*9. Outpatients\*/

**proc** **sql**;

create table bptout.lnk\_9\_Outpatients

as select

a.HRG\_code,

a.HRG\_name,

b.Procedure,

a.BPTprice\_OPROC as Best\_pract\_outpatient\_tariff,

a.NonBPT\_minusAbsDiff as Nonbest\_pract\_dc\_ord\_el\_tariff,

a.HRG\_or\_subHRG as HRG\_or\_sub\_HRG\_level,

a.BPT\_flag as BPT\_flag

/\*\*/

from BPTout.outpatients as a

left join BPTInput.Bpt\_parameter\_currentyear as b on a.HRG\_code=b.HRG\_code

;

**quit**;

/\*10. Paediatric diabetes year of care\*/

**proc** **sql**;

create table bptout.lnk\_10\_Paediatric\_diabetes

as select

HRG\_code,

HRG\_name,

BPT\_current\_year as Best\_practice\_tariff

/\*\*/

from BPTout.group2

where substr(HRG\_name,**1**,**10**)='Paediatric'

;

**quit**;

/\*11. Paediatric epilepsy\*/

**proc** **sql**;

create table bptout.lnk\_11\_Paediatric\_epilepsy

as select

TFC as Treatment\_function\_code,

Tariff\_TFC as Treatment\_function\_name,

round(nonBPT\_WF01A) as Non\_best\_practice\_tariffWF01A,

round(nonBPT\_WF02A) as Non\_best\_practice\_tariffWF02A,

BPT\_WF01A as Best\_practice\_tariffWF01A ,

BPT\_WF02A as Best\_practice\_tariffWF02A

from BPTout.paedep

where TFC=**223**

;

**quit**;

/\*12. Parkinson's disease\*/

**proc** **sql**;

create table bptout.lnk\_12\_Parkinson

as select

HRG\_code,

HRG\_name,

BPT\_current\_year as Best\_practice\_tariff

/\*\*/

from BPTout.group2

where substr(HRG\_name,**1**,**9**)='Parkinson'

;

**quit**;

/\*13. Pleural Effusion\*/

**proc** **sql**;

create table bptout.lnk\_13a\_PleuralEffusion

as select

a.HRG\_code,

a.HRG\_name,

round(a.Final\_BPT\_Price) as Day\_case\_best\_practice\_tariff,

b.HRG\_or\_subHRG as HRG\_or\_sub\_HRG\_level,

BPT\_flag

/\*\*/

from BPTout.pleuraleff as a

left join BPTinput.BPT\_PArameter\_currentyear as b on a.HRG\_code=b.HRG\_code

where a.HRG\_code in ('YD05Z','YD04Z')

;

**quit**;

**proc** **sql**;

create table bptout.lnk\_13b\_PleuralEffusion

as select

a.HRG\_code,

a.HRG\_name,

round(a.Final\_BPT\_Price) as Day\_case\_best\_practice\_tariff,

case when c.SSEM\_pct =**.** then 'n/a' end as ReducedSSEM\_NonBestpract\_tariff,

b.HRG\_or\_subHRG as HRG\_or\_sub\_HRG\_level,

BPT\_flag,

case when calculated ReducedSSEM\_NonBestpract\_tariff='n/a' then **.** end as Percentage\_reduction\_SSEM\_tariff

/\*\*/

from BPTout.pleuraleff as a

left join BPTinput.BPT\_PArameter\_currentyear as b on a.HRG\_code=b.HRG\_code

left join

bptinput.apc\_prices as c on a.HRG\_code=c.HRG\_code

where a.HRG\_code in ('DZ16N') and a.Setting='NE'

;

**quit**;

/\*14. Primary total hip and knee replacements\*/

**proc** **sql**;

create table bptout.lnk\_14\_Primary\_hipknee\_replac

as select

a.HRG\_code,

a.HRG\_name,

round(nonBPT\_tariff) as Non\_best\_practice\_tariff,

round(BPT\_tariff) as Best\_practice\_tariff,

b.HRG\_or\_subHRG as HRG\_or\_sub\_HRG\_level,

b.BPT\_flag

from BPTout.group1 as a

left join BPTinput.BPT\_PArameter\_currentyear as b on a.HRG\_code=b.HRG\_code

where a.Category in ('Primary Total Hip & Knee Replacements')

;

**quit**;

/\*15. Same day emergency care\*/

**proc** **sql**;

create table bptout.LNK\_15\_SAME\_DAY\_EM\_CARE

as select

a.HRG\_code,

a.HRG\_name,

b.Clinical\_Scenario,

a.BPTprice\_SameDayEM as Same\_day\_emergency\_care\_BPT,

a.nonBPTpriceNE\_quantumneautral as Non\_elective\_non\_BPT\_tariff,

b.HRG\_or\_subHRG as HRG\_or\_sub\_HRG\_level,

b.BPT\_Flag

from BPTout.SameDayEmCare as a

left join BPTinput.BPT\_PArameter\_currentyear as b on a.HRG\_code=b.HRG\_code

where b.Category='Same Day Emergency Care'

;

**quit**;

/\*16. Transient ischaemic attack\*/

**proc** **sql**;

create table bptout.lnk\_16a\_Tia

as select

HRG\_code as Treatment\_function,

tf\_name as Treatment\_function\_name,

round(FinalPrice) as WF01B\_First\_Att\_Single\_Prof,

round(FinalPrice) as WF02B\_First\_Att\_Multi\_Prof

from BPTout.tia\_table1

;

**quit**;

**proc** **sql**;

create table bptout.lnk\_16b\_Tia

as select

monotonic() as ID,

tf\_name as Best\_practice\_tariff\_adjustments,

round(tariffadj) as Tariff\_adjustment

from BPTout.tia\_table2

;

**quit**;

/\*17. Heart Failure\*/

**proc** **sql**;

create table bptout.lnk\_17\_heartfailure

as select

a.HRG\_code,

a.HRG\_name,

round(a.BPT\_Tariff) as BPT\_Non\_elective\_tariff,

round(a.BPT\_Tariff\*b.SSEM\_pct) as BPT\_Reduced\_SSEM\_tariff,

round(a.nonBPT\_Tariff) as Non\_BPT\_Non\_elective\_tariff,

round(a.nonBPT\_Tariff\*b.SSEM\_pct) as Non\_BPT\_Reduced\_SSEM\_tariff,

HRG\_or\_subHRG as HRG\_or\_sub\_HRG\_level,

BPT\_Flag,

b.SSEM\_pct as Percentage\_reduction\_SSEM\_tariff

from bptout.group1 as a

left join

bptinput.apc\_prices as b on a.HRG\_code=b.HRG\_code

where substr(a.HRG\_code,**1**,**2**) in ('EB');

**quit**;

/\*18. COPD Exacerbation\*/

**proc** **sql**;

create table bptout.lnk\_18\_COPD\_Exacerbation

as select

a.HRG\_code,

a.HRG\_name,

round(a.BPT\_Tariff) as BPT\_Non\_elective\_tariff,

round(a.BPT\_Tariff\*b.SSEM\_pct) as BPT\_Reduced\_SSEM\_tariff,

round(a.nonBPT\_Tariff) as Non\_BPT\_Non\_elective\_tariff,

round(a.nonBPT\_Tariff\*b.SSEM\_pct) as Non\_BPT\_Reduced\_SSEM\_tariff,

HRG\_or\_subHRG as HRG\_or\_sub\_HRG\_level,

BPT\_Flag,

b.SSEM\_pct as Percentage\_reduction\_SSEM\_tariff

from bptout.group1 as a

left join

bptinput.apc\_prices as b on a.HRG\_code=b.HRG\_code

where substr(a.HRG\_code,**1**,**2**) in ('DZ');

**quit**;

/\*19. NSTEMI\*/

**proc** **sql**;

create table bptout.lnk\_19\_NSTEMI

as select

a.HRG\_code,

a.HRG\_name,

round(a.BPT\_Tariff) as BPT\_Non\_elective\_tariff,

round(a.nonBPT\_Tariff) as Non\_BPT\_Non\_elective\_tariff,

HRG\_or\_subHRG as HRG\_or\_sub\_HRG\_level,

BPT\_Flag

from bptout.group1 as a

left join

bptinput.apc\_prices as b on a.HRG\_code=b.HRG\_code

where substr(a.HRG\_code,**1**,**2**) in ('EY');

**quit**;

/\*20. EMERGENCY LAPAROTOMY for 1920\*/

**proc** **sql**;

create table bptout.lnk\_20\_EM\_LAPAROTOMY

as select

a.HRG\_code,

a.HRG\_name,

round(a.BPT\_Tariff) as BPT\_Non\_elective\_tariff,

round(a.nonBPT\_Tariff) as Non\_BPT\_Non\_elective\_tariff,

HRG\_or\_subHRG as HRG\_or\_sub\_HRG\_level,

BPT\_Flag

from bptout.group1 as a

left join

bptinput.apc\_prices as b on a.HRG\_code=b.HRG\_code

where Category ='Emergency Laparotomy';

**quit**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 05 May 2018 ;

\* Modified by: PID ;

\* ;

# \* MPP\_Step\_9 Tables\_Lev1.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

/\*ODS to create Excel workbook output with multiple sheets\*/

ods \_all\_ close;

ods tagsets.ExcelXP path="\\irnarch\sas\_data\Tariff Rebuild\Models for Testing\TC\_Y1920\Excel\_outputs\&username." file="19\_20 Maternity Model Final Prices 6 Levels .xml" style=Printer;

ods tagsets.ExcelXP options(sheet\_name='Maternity Final prices ' sheet\_interval = 'NONE'

autofilter='NO' embedded\_titles ="yes" autofit\_height ="yes"

absolute\_column\_width='15,20,10,10,10,10,10,10,10,10,10,10,10,10,10');

Title " ";

Title "1. Non-delivery phases";

Title;

Title "1a. Antenatal phase";

**proc** **print** data = OutSAS.MPPLINKEDSHANTENATALFINALPRICE noobs label split= '\*' ;

var code Name Tariff

/style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###,###"];

label

Tariff="Tariff (£)";

**run**;

Title "1b. Postnatal phase";

**proc** **print** data = OutSAS.MPPLINKEDSHPOSTNATALFINALPRICE noobs label split= '\*' ;

var code Name Tariff

/style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###,###"];

label

Tariff="Tariff (£)";

**run**;

Title "2. Mandatory delivery admitted patient and outpatient procedure prices for maternity services";

**proc** **print** data = OutSAS.MPPLINKEDSHDELIVERYHRGFPRICES noobs label split= '\*' ;

var level /style(header)=[background=#b3ffff];

var

Combined\_DC\_EL\_NE /style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###,###"];

var

Day\_Case\_Tariff

Ordinary\_Elective\_Tariff

Ordinary\_elective\_LS\_trimpoint /style(header)=[background=#b3ffff];

var

Non\_elective\_tariff /style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###,###"];

var

Non\_elective\_LS\_trimpoint /style(header)=[background=#b3ffff];

var Long\_stay\_payment /style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###,###"];

var

Reduced\_SSE\_tariff\_Applic

Pct\_Reduced\_SSE\_tariff\_Applic

Reduced\_SSE\_tariff

/style(header)=[background=#b3ffff];

label

level="Level"

Combined\_DC\_EL\_NE="Combined day \*case / ordinary \*elective spell \*tariff (£)"

Day\_Case\_Tariff="Day case spell \*tariff (£)"

Ordinary\_Elective\_Tariff="Ordinary elective spell tariff (£)"

Ordinary\_elective\_LS\_trimpoint="Ordinary elective \*long stay trimpoint \*(days)"

Non\_elective\_tariff="Non-elective \*spell tariff (£)"

Non\_elective\_LS\_trimpoint="Non-elective \*long stay \*trimpoint (days)"

Long\_stay\_payment="Per day long \*stay payment \*(for days exceeding \*trimpoint) (£)"

Reduced\_SSE\_tariff\_Applic="Reduced short \*stay \*emergency \*tariff \*applicable?"

Pct\_Reduced\_SSE\_tariff\_Applic="% applied in \*calculation of \*reduced short \*stay \*emergency \*tariff "

Reduced\_SSE\_tariff="Reduced \*short stay \*emergency \*tariff (£)"

;

**run**;

Title;

ods tagsets.ExcelXP close;

ods listing;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 05 May 2018 ;

\* Modified by: PID ;

\* ;

# \* MPP\_Step\_10 Tables\_Lev1.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

/\*ODS to create Excel workbook output with multiple sheets\*/

ods \_all\_ close;

ods tagsets.ExcelXP path="\\irnarch\sas\_data\Tariff Rebuild\Models for Testing\TC\_Y1920\Excel\_outputs\&username." file="19\_20 Maternity Model Prices Step by Step 6 Levels.xml" style=Printer;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\*\*\*\*\*\* Step 5 Run Maternity Model \*\*\*\*\*\*;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

ods tagsets.ExcelXP options(sheet\_name='5.1 MPP\_DELIVERY\_PRICES\_STEPBYSTEP'

suppress\_bylines='yes'

frozen\_headers='1'

FROZEN\_ROWHEADERS='1'

autofit\_height='Yes'

absolute\_column\_width='20, 10, 10, 10, 10, 10,10, 10, 10, 10, 10, 10, 10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10,'

pages\_fitwidth='1'

pages\_fitheight='9'

FitToPage="yes"

autofilter='1-21');

/\*Define STEP BY STEP QUANTUM RECONCILIATION BY Sub Chapter variables\*/

/\*Define STEP BY STEP QUANTUM RECONCILIATION BY Sub Chapter variables\*/

**proc** **print** data=OutSAS.MPP\_DELIVERY\_IANDE1920 split='\*' noobs;

var level/style(header)=[background=#b3ffff];;

var Total\_Activity/style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###,###"];

var Total\_TC/style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###,###"];

var Unit\_Cost/style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###,###"];

var QR1\_Adjustment/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Delivery\_QR1/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var CB\_Adjustment/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Delivery\_CB / style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var Total\_I\_E\_Adjustment/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Delivery\_Total\_I\_E/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var Total\_CNST/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Uplift\_Total\_Adjust/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Delivery\_Total\_CNST/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var Prices\_Delivery\_PostMA/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var QR2\_Adjustment/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Delivery\_QR2/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var CashInOut\_Delivery\_Factor/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Delivery\_CashInOut/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var SMF/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Delivery\_SMF/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var Scaling\_Factor/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Delivery\_ScalingFact/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var CNST1920/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Delivery\_CNST1920/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var I\_E\_1920Adjustment/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Delivery\_FinalPrices/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

;

label

level="Delivery Phase\*Level"

Inlier\_Activity="APC Activity\_\_"

OPROC\_Activity="OPROC Activity\_\_"

Comm\_Activity="Community Activity\_\_"

Total\_Activity="Total Activity\_\_"

Inlier\_TC="APC Total Cost\_\_"

OPROC\_TC="OPROC Total Cost\_\_"

Comm\_TC="CommunityTotal Cost\_\_"

Total\_TC="Total Cost\_\_"

Unit\_Cost="Unit Cost\_\_"

QR1\_Adjustment="Reference costs \*reconciliation \*factor (QR1)\_\_"

Prices\_Delivery\_QR1="Prices \*after QR1 \*adjustment \*(£)\_\_"

CB\_Adjustment="Cost base \*adjustment \*(CB)\_\_"

Prices\_Delivery\_CB="Prices after \*implementing \*CB factors\_\_"

Total\_I\_E\_Adjustment="Inflation and \*Efficiency \*2017/18 and \*2018/19\_\_"

Prices\_Delivery\_Total\_I\_E="Prices after \*implementing \*Inflation \*and Efficiency factors\_\_"

Total\_CNST="Total CNST up to 18/19 price levels"

Uplift\_Total\_Adjust="Total Adjustment\_\_"

Prices\_Delivery\_Total\_CNST="Modelled prices\*adjusted to a\*18/19 base (£)\_\_"

Prices\_Delivery\_PostMA="Manually adjusted \*prices before \*QR2 adjustment\_\_"

QR2\_Adjustment="Manual \*adjustments \*reconciliation \*factor (QR2)\_\_"

Prices\_Delivery\_QR2="Prices after \*implementing QR2 \*factor\_\_"

CashInOut\_Delivery\_Factor="Cash in cash out\* adjustment factor\_\_"

Prices\_Delivery\_CashInOut="Price after \*Cash in Cash out \*adjustment\_\_"

SMF="Smoothing \*Factor \*(SMF)\_\_"

Prices\_Delivery\_SMF="Prices after \*implementing the \*Smoothing \*factor\_\_"

Scaling\_Factor="Scaling \*factor \*(SCF)\_"

Prices\_Delivery\_ScalingFact="Prices after \*implementing the \*Scaling \*factor\_\_"

CNST1920="CNST \*2019/20\_\_"

Prices\_Delivery\_CNST1920="Prices after \*implementing \*CNST \*factor\_\_"

I\_E\_1920Adjustment="Inflation and\* Efficiency \*2019/2020\_\_"

Prices\_Delivery\_FinalPrices="Price after \*applying \*prospective \*adjustments\_\_"

;

**run**;

**quit**;

ods tagsets.ExcelXP options(sheet\_name='5.2 MPP\_ANTENATAL\_PRICES\_STEPBYSTEP'

suppress\_bylines='yes'

frozen\_headers='1'

FROZEN\_ROWHEADERS='1'

autofit\_height='Yes'

absolute\_column\_width='10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10'

pages\_fitwidth='1'

pages\_fitheight='9'

FitToPage="yes"

autofilter='1-21');

/\*Define STEP BY STEP QUANTUM RECONCILIATION BY Sub Chapter variables\*/

**proc** **print** data=OutSAS.MPP\_ANTENATAL\_IANDE1920 split='\*' noobs;

var Level / style(Column)=[background=#e6ffff ] style(header)=[background=#e6ffff];

var Act\_assumed /style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var UC\_Antenatal/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var QR1\_Adjustment/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Antenatal\_QR1/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var CB\_Adjustment/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Antenatal\_CB / style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var Total\_I\_E\_Adjustment/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Antenatal\_Total\_I\_E/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

Var Total\_CNST/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Uplift\_Total\_Adjust/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Antenatal\_Total\_CNST/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var CashInOut\_Antenatal\_Factor/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Antenatal\_CashInOut/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var Prices\_Antenatal\_PostMA/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var QR2\_Adjustment/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Antenatal\_QR2/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var SMF/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Antenatal\_SMF/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var Scaling\_Factor/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Antenatal\_ScalingFact/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var I\_E\_1920Adjustment/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Antenatal\_FinalPrices/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

;

label

Level="Antenatal Phase"

Act\_assumed="Activities"

UC\_Antenatal="Unit Cost \*(£)\_\_"

QR1\_Adjustment="Reference costs \*reconciliation \*factor (QR1)\_\_"

Prices\_Antenatal\_QR1="Prices \*after QR1 \*adjustment \*(£)\_\_"

CB\_Adjustment="Cost base \*adjustment \*(CB)\_\_"

Prices\_Antenatal\_CB="Prices after \*implementing \*CB factors\_\_"

Total\_I\_E\_Adjustment="Inflation and \*Efficiency \*2017/18 and \*2018/19\_\_"

Prices\_Antenatal\_Total\_I\_E="Prices after \*implementing \*Inflation \*and Efficiency factors\_\_"

Total\_CNST="Total CNST up to 18/19 price levels"

Uplift\_Total\_Adjust="Total Adjustment\_\_"

Prices\_Antenatal\_Total\_CNST="Modelled prices\*adjusted to a\*18/19 base (£)\_\_"

CashInOut\_Antenatal\_Factor="Cash in cash out\*Adjustment factor\_\_"

Prices\_Antenatal\_CashInOut="Price after \*cash in cash out \*adjustment\_\_"

Prices\_Antenatal\_PostMA="Manually adjusted\*prices before\*QR2 adjustment\_\_"

QR2\_Adjustment="Manual \*adjustments \*reconciliation \*factor (QR2)\_\_"

Prices\_Antenatal\_QR2="Prices after \*implementing QR2 \*factor\_\_"

SMF="Smoothing \*Factor \*(SMF)\_\_"

Prices\_Antenatal\_SMF="Prices after \*implementing the \*Smoothing \*factor\_\_"

Scaling\_Factor="Scaling \*factor \*(SCF)\_"

Prices\_Antenatal\_ScalingFact="Prices after \*implementing the \*Scaling \*factor\_\_"

I\_E\_1920Adjustment="Inflation and\* Efficiency \*2019/2020\_\_"

Prices\_Antenatal\_FinalPrices="Price after \*applying \*prospective \*adjustments\_\_"

;

**quit**;

ods tagsets.ExcelXP options(sheet\_name='5.3 MPP\_POSTNATAL\_PRICES\_STEPBYSTEP'

suppress\_bylines='yes'

frozen\_headers='1'

FROZEN\_ROWHEADERS='1'

autofit\_height='Yes'

absolute\_column\_width='10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10'

pages\_fitwidth='1'

pages\_fitheight='9'

FitToPage="yes"

autofilter='1-21');

/\*Define STEP BY STEP QUANTUM RECONCILIATION BY Sub Chapter variables\*/

**proc** **print** data=OutSAS.MPP\_POSTNATAL\_IANDE1920 split='\*' noobs;

var Level / style(Column)=[background=#e6ffff ] style(header)=[background=#e6ffff];

var Act\_assumed /style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var UC\_Postnatal/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var QR1\_Adjustment/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Postnatal\_QR1/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var CB\_Adjustment/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Postnatal\_CB / style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var Total\_I\_E\_Adjustment/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Postnatal\_Total\_I\_E/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

Var Total\_CNST/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Uplift\_Total\_Adjust/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Postnatal\_Total\_CNST/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var CashInOut\_Postnatal\_Factor/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Postnatal\_CashInOut/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var Prices\_Postnatal\_PostMA/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var QR2\_Adjustment/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Postnatal\_QR2/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var SMF/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Postnatal\_SMF/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var Scaling\_Factor/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Postnatal\_ScalingFact/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

var I\_E\_1920Adjustment/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var Prices\_Postnatal\_FinalPrices/ style(Column)=[background=#b3ffb3 tagattr="format:###,###,###,###"] style(header)=[background=#b3ffb3] ;

label

Level="Postnatal Phase"

Act\_assumed="Activities"

UC\_Postnatal="Unit Cost \*(£)\_\_"

QR1\_Adjustment="Reference costs \*reconciliation \*factor (QR1)\_\_"

Prices\_Postnatal\_QR1="Prices \*after QR1 \*adjustment \*(£)\_\_"

CB\_Adjustment="Cost base \*adjustment \*(CB)\_\_"

Prices\_Postnatal\_CB="Prices after \*implementing \*CB factors\_\_"

Total\_I\_E\_Adjustment="Inflation and \*Efficiency \*2017/18 and \*2018/19\_\_"

Prices\_Postnatal\_Total\_I\_E="Prices after \*implementing \*Inflation \*and Efficiency factors\_\_"

Total\_CNST="Total CNST up to 18/19 price levels"

Uplift\_Total\_Adjust="Total Adjustment\_\_"

Prices\_Postnatal\_Total\_CNST="Modelled prices\*adjusted to a\*18/19 base (£)\_\_"

CashInOut\_Postnatal\_Factor="Cash in cash out\*Adjustment factor\_\_"

Prices\_Postnatal\_CashInOut="Price after \*cash in cash out \*adjustment\_\_"

Prices\_Postnatal\_PostMA="Manually adjusted\*prices before\*QR2 adjustment\_\_"

QR2\_Adjustment="Manual \*adjustments \*reconciliation \*factor (QR2)\_\_"

Prices\_Postnatal\_QR2="Prices after \*implementing QR2 \*factor\_\_"

SMF="Smoothing \*Factor \*(SMF)\_\_"

Prices\_Postnatal\_SMF="Prices after \*implementing the \*Smoothing \*factor\_\_"

Scaling\_Factor="Scaling \*factor \*(SCF)\_"

Prices\_Postnatal\_ScalingFact="Prices after \*implementing the \*Scaling \*factor\_\_"

I\_E\_1920Adjustment="Inflation and\* Efficiency \*2019/2020\_\_"

Prices\_Postnatal\_FinalPrices="Price after \*applying \*prospective \*adjustments\_\_"

;

**quit**;

ods tagsets.ExcelXP close;

ods listing;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 28 November 2017 ;

# \* MPP\_IA.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

/\* 1. Calculate current year quantum for Provider at delivery phase \*/

/\* create HES based activity by provider and HRG for delivery phase \*/

%let Run\_ID=33;

**proc** **sql**;

create table MPP\_IA\_Provider\_HRG\_Activity\_1 as

SELECT a.PROCODETD as Provider

,a.RC\_SpellHRG\_Rev as HRG

,a.ADM as Admission

,sum(a.SPELLFLAG) as Activity

FROM

bprPrPP.Stage\_HES\_KeyFields as a

WHERE

a.Run\_ID=&Run\_ID. and

(a.Revised\_Excluded ='N'

AND substr(a.PROCODETD,**1**,**1**) in ('5','R','T'))

AND a.SpellFlag <>**0**

AND

a.RC\_SpellHRG\_Rev like 'NZ%'

and

a.ADM <> 'UX'

group by a.PROCODETD,a.RC\_SpellHRG\_Rev, a.ADM

order by a.PROCODETD,a.RC\_SpellHRG\_Rev, a.ADM

;

**quit**;

/\* create a pivot table \*/

**proc** **sort** data=MPP\_IA\_Provider\_HRG\_Activity\_1 out=MPP\_IA\_Provider\_HRG\_Activity\_1;

by Provider HRG Admission;

**run**;

**PROC** **TRANSPOSE** data=MPP\_IA\_Provider\_HRG\_Activity\_1 out=MPP\_IA\_Provider\_HRG\_Activity\_2;

by PROvider HRG;

id Admission;

var Activity;

**RUN**;

**Data** MPP\_IA\_Provider\_HRG\_Activity(drop=\_Name\_);

Set MPP\_IA\_Provider\_HRG\_Activity\_2;

Activity=sum(NE,DC,EL);

**Run**;

/\*Retrive data from reference cost database\*/

**data** TC\_MFF\_Selected;

set Tariff\_R.TC\_MFF;

where Year="&Year.";

Target\_MFF=Uncapped\_MFF;

**run**;

**proc** **sql**;

create table work.MPP\_IA\_Provider\_HRG\_EBD as

SELECT rfa.FK\_ORGS\_PROV\_ID AS PROVIDER,

pt.Cluster\_Name as Provider\_Type,

rfa.CURRENCY AS HRG,

pt.RC\_Name,

Sum(rfa.EXCESS\_BED\_DAYS\_ACTIVITY) AS EBD\_ACT

FROM Tariff\_R.REFCOSTACUTE\_1920 as rfa

INNER JOIN TC\_MFF\_Selected as pt

ON rfa.FK\_ORGS\_PROV\_ID = pt.RC\_Code

WHERE rfa.SUPPLIER\_TYPE='OWN' and (rfa.CURRENCY like ('NZ3%')or rfa.CURRENCY like ('NZ4%') or rfa.CURRENCY like ('NZ5%'))

GROUP BY rfa.FK\_ORGS\_PROV\_ID, pt.Cluster\_Name,pt.RC\_Name,rfa.CURRENCY

ORDER BY rfa.FK\_ORGS\_PROV\_ID

;

**quit**;

/\* Pending Inlier activity and EBD activity together \*/

**proc** **sql**;

create table MPP\_IA\_Provider\_HRG\_Combine\_ACT as

select

a.Provider,

b.Provider\_Type,

b.RC\_Name,

a.HRG,

a.Activity,

b.EBD\_ACT

from work.MPP\_IA\_Provider\_HRG\_Activity as a

Inner join work.MPP\_IA\_Provider\_HRG\_EBD as b

on a.HRG=b.HRG and a.Provider=b.Provider

order by a.HRG, a.Provider

;

**quit**;

/\* Export the Provider\_HRG activity table above \*/

%let outputfile=MPP\_IA\_Provider\_HRG\_Combine\_ACT;

**proc** **export** data=work.MPP\_IA\_Provider\_HRG\_Combine\_ACT

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

/\* Calculate the quantum for the current year \*/

/\* Combine the Inlier Activity table above with tariff linked table to calculate inlier\_quantum \*/

/\* Reconcile the number of activity into the totalbirths \*/

**proc** **sql** noprint;

select Total\_Activity into :Total\_Activity

from MPP\_DELIVERY\_ACTIVITY\_3

;

**quit**;

%let TotalBirth=&Total\_Activity;

%put TotalBirth=&Total\_Activity;

**proc** **sql**;

create table MPP\_IA\_Provider\_quantum\_1 as

select

a.Provider

,a.Provider\_Type

,a.RC\_Name

,a.HRG

,a.Activity AS Delivery\_ACT

,a.EBD\_ACT as EBD\_ACT

,b.Combined\_DC\_EL\_NE as Price\_delivery

,b.Long\_stay\_payment as Price\_EBD

from MPP\_IA\_Provider\_HRG\_Combine\_ACT as a

left join OUTSAS.MPP\_LINKEDSH\_DELIVERY\_HRG\_PRICES as b

on a.HRG=b.HRG\_Code

group by a.Provider,a.Provider\_Type,a.RC\_Name,a.HRG

order by a.Provider

;

**quit**;

**proc** **sql** noprint;

select sum(Delivery\_ACT) into :TotalAct\_Delivery

from MPP\_IA\_Provider\_quantum\_1

;

**quit**;

%let TotalDelivery=&TotalAct\_Delivery;

**proc** **sql**;

create table MPP\_IA\_Provider\_quantum\_3 as

select

Provider

,Provider\_Type

,RC\_Name

,sum(Delivery\_ACT\*Price\_delivery\*&TotalBirth/&TotalDelivery) as Quantum\_Delivery

from MPP\_IA\_Provider\_quantum\_1

group by Provider,Provider\_Type,RC\_Name

order by Provider

;

**quit**;

/\* Combine the EBD Activity table above with tariff linked table to calculate EBD\_quantum \*/

**proc** **sql** noprint;

select Total\_Activity into :Total\_Activity

from MPP\_DELIVERY\_ACTIVITY\_4

;

**quit**;

%let TotalEBD=&Total\_Activity;

**proc** **sql** noprint;

select sum(EBD\_ACT) into :TotalAct\_EBD

from MPP\_IA\_Provider\_quantum\_1

;

**quit**;

%let TotalACT\_EBD=&TotalAct\_EBD;

**proc** **sql**;

create table MPP\_IA\_Provider\_quantum\_EBD as

select

Provider

,Provider\_Type

,RC\_Name

,sum(EBD\_ACT\*Price\_EBD\*&TotalEBD/&TotalACT\_EBD) as Quantum\_EBD

from MPP\_IA\_Provider\_quantum\_1

group by Provider,Provider\_Type,RC\_Name

order by Provider

;

**quit**;

/\* Join inlier quantum with EBD quantum \*/

**proc** **sql**;

create table MPP\_IA\_Provider\_Delivery as

select

a.Provider

,a.Provider\_Type

,a.RC\_Name

,a.Quantum\_Delivery

,b.Quantum\_EBD

from MPP\_IA\_Provider\_quantum\_3 as a

left join MPP\_IA\_Provider\_quantum\_EBD as b

on a.Provider=b.Provider

group by a.Provider,a.Provider\_Type,a.RC\_Name

order by a.Provider

;

**quit**;

/\* 2. Calculate quantum at antenatal phase of by using current year prices \*/

/\* Create Activity table for antenatal phase \*/

**proc** **sql** noprint;

select Value format = Best12.8 into :Upliftbirths

from PARMPP.MPP\_STILLBIRTHS\_PARA

;

**quit**;

**proc** **sql** noprint;

select Antenatal format = Best12.8 into :ACT\_percent\_ST

from PARMPP.MPP\_ACTIVITY\_PARA

where Level in ('Standard')

;

**quit**;

**proc** **sql** noprint;

select Antenatal format = Best12.8 into :ACT\_percent\_Intm

from PARMPP.MPP\_ACTIVITY\_PARA

where Level in ('Intermediate')

;

**quit**;

**proc** **sql** noprint;

select Antenatal format = Best12.8 into :ACT\_percent\_Ints

from PARMPP.MPP\_ACTIVITY\_PARA

where Level in ('Intensive')

;

**quit**;

%let Upliftbirths=&Upliftbirths;

%put Upliftbirths=&Upliftbirths;

%let ACT\_percent\_ST=&ACT\_percent\_ST;

%put ACT\_percent\_ST=&ACT\_percent\_ST;

%let ACT\_percent\_Intm =&ACT\_percent\_Intm;

%put ACT\_percent\_Intm =&ACT\_percent\_Intm;

%let ACT\_percent\_Ints =&ACT\_percent\_Ints;

%put ACT\_percent\_Ints =&ACT\_percent\_Ints;

**proc** **sql**;

create table MPP\_IA\_Provider\_antenatal\_1 as

select

Provider

,sum(Activity)as Activity

from MPP\_IA\_PROVIDER\_HRG\_ACTIVITY

group by Provider

;

**quit**

;

**proc** **sql**;

create table MPP\_IA\_Provider\_antenatal\_2 as

select

Provider,

Activity\*(**1**+&Upliftbirths)\*&ACT\_percent\_ST as Act\_antenatal\_standard,

Activity\*(**1**+&Upliftbirths)\*&ACT\_percent\_Intm as Act\_antenatal\_Intemediate,

Activity\*(**1**+&Upliftbirths)\*&ACT\_percent\_Ints as Act\_antenatal\_Intensive

from MPP\_IA\_Provider\_antenatal\_1

;

**quit**;

/\* Calculation quantum at antenatal phase by multiplying activity above with current year prices \*/

/\* Retrive current year prices from linked table produced in Step 8 \*/

**proc** **sql** noprint;

select Tariff format = Best12.8 into :Price\_crrntyr\_ST

from OUTSAS.MPP\_LINKEDSH\_ANTENATAL

where Name in ('Standard')

;

**quit**;

**proc** **sql** noprint;

select Tariff format = Best12.8 into :Price\_crrntyr\_INTM

from OUTSAS.MPP\_LINKEDSH\_ANTENATAL

where Name in ('Intermediate')

;

**quit**;

**proc** **sql** noprint;

select Tariff format = Best12.8 into :Price\_crrntyr\_INTS

from OUTSAS.MPP\_LINKEDSH\_ANTENATAL

where Name in ('Intensive')

;

**quit**;

%let Price\_crrntyr\_ST=&Price\_crrntyr\_ST;

%put Price\_crrntyr\_ST=&Price\_crrntyr\_ST;

%let Price\_crrntyr\_INTM =&Price\_crrntyr\_INTM;

%put Price\_crrntyr\_INTM =&Price\_crrntyr\_INTM;

%let Price\_crrntyr\_INTS=&Price\_crrntyr\_INTS;

%put Price\_crrntyr\_INTS=&Price\_crrntyr\_INTS;

/\* Calculate current year quantum at antenatal phase \*/

**proc** **sql**;

create table MPP\_IA\_Provider\_antenatal\_3 as

select

Provider,

Act\_antenatal\_standard\*&Price\_crrntyr\_ST as Quant\_antenatal\_crrntyr\_STD,

Act\_antenatal\_Intemediate\*&Price\_crrntyr\_INTM as Quant\_antenatal\_crrntyr\_INTM,

Act\_antenatal\_Intensive\*&Price\_crrntyr\_INTS as Quant\_antenatal\_crrntyr\_INTS

from MPP\_IA\_Provider\_antenatal\_2

;

**quit**;

**proc** **sql**;

create table MPP\_IA\_Provider\_antenatal\_4 as

select

Provider,

Quant\_antenatal\_crrntyr\_STD,

Quant\_antenatal\_crrntyr\_INTM,

Quant\_antenatal\_crrntyr\_INTS,

Quant\_antenatal\_crrntyr\_STD+Quant\_antenatal\_crrntyr\_INTM+Quant\_antenatal\_crrntyr\_INTS as TotalQuant\_antenatal

from MPP\_IA\_Provider\_antenatal\_3

;

**quit**;

/\* 4. Calculate quantum at postnatal phase of by using current year prices \*/

/\* Create Activity table for postnatal phase \*/

**proc** **sql** noprint;

select Postnatal format = Best12.8 into :ACT\_percent\_ST

from PARMPP.MPP\_ACTIVITY\_PARA

where Level in ('Standard')

;

**quit**;

**proc** **sql** noprint;

select Postnatal format = Best12.8 into :ACT\_percent\_Intm

from PARMPP.MPP\_ACTIVITY\_PARA

where Level in ('Intermediate')

;

**quit**;

**proc** **sql** noprint;

select Postnatal format = Best12.8 into :ACT\_percent\_Ints

from PARMPP.MPP\_ACTIVITY\_PARA

where Level in ('Intensive')

;

**quit**;

%let ACT\_percent\_ST=&ACT\_percent\_ST;

%put ACT\_percent\_ST=&ACT\_percent\_ST;

%let ACT\_percent\_Intm =&ACT\_percent\_Intm;

%put ACT\_percent\_Intm =&ACT\_percent\_Intm;

%let ACT\_percent\_Ints =&ACT\_percent\_Ints;

%put ACT\_percent\_Ints =&ACT\_percent\_Ints;

**proc** **sql**;

create table MPP\_IA\_Provider\_postnatal\_1 as

select

Provider

,sum(Activity)as Activity

from MPP\_IA\_PROVIDER\_HRG\_ACTIVITY

group by Provider

;

**quit**

;

**proc** **sql**;

create table MPP\_IA\_Provider\_postnatal\_2 as

select

Provider,

Activity\*&ACT\_percent\_ST as Act\_postnatal\_standard,

Activity\*&ACT\_percent\_Intm as Act\_postnatal\_Intemediate,

Activity\*&ACT\_percent\_Ints as Act\_postnatal\_Intensive

from MPP\_IA\_Provider\_postnatal\_1

;

**quit**;

/\* Calculation quantum at postnatal phase by multiplying activity above with current year prices \*/

/\* Retrive current year prices from linked table produced in Step 8 \*/

**proc** **sql** noprint;

select Tariff format = Best12.8 into :Price\_crrntyr\_ST

from OUTSAS.MPP\_LINKEDSH\_POSTNATAL

where Name in ('Standard')

;

**quit**;

**proc** **sql** noprint;

select Tariff format = Best12.8 into :Price\_crrntyr\_INTM

from OUTSAS.MPP\_LINKEDSH\_POSTNATAL

where Name in ('Intermediate')

;

**quit**;

**proc** **sql** noprint;

select Tariff format = Best12.8 into :Price\_crrntyr\_INTS

from OUTSAS.MPP\_LINKEDSH\_POSTNATAL

where Name in ('Intensive')

;

**quit**;

%let Price\_crrntyr\_ST=&Price\_crrntyr\_ST;

%put Price\_crrntyr\_ST=&Price\_crrntyr\_ST;

%let Price\_crrntyr\_INTM =&Price\_crrntyr\_INTM;

%put Price\_crrntyr\_INTM =&Price\_crrntyr\_INTM;

%let Price\_crrntyr\_INTS=&Price\_crrntyr\_INTS;

%put Price\_crrntyr\_INTS=&Price\_crrntyr\_INTS;

/\* Calculate current year quantum at postnatal phase \*/

**proc** **sql**;

create table MPP\_IA\_Provider\_postnatal\_3 as

select

Provider,

Act\_postnatal\_standard\*&Price\_crrntyr\_ST as Quant\_postnatal\_crrntyr\_STD,

Act\_postnatal\_Intemediate\*&Price\_crrntyr\_INTM as Quant\_postnatal\_crrntyr\_INTM,

Act\_postnatal\_Intensive\*&Price\_crrntyr\_INTS as Quant\_postnatal\_crrntyr\_INTS

from MPP\_IA\_Provider\_postnatal\_2

;

**quit**;

**proc** **sql**;

create table MPP\_IA\_Provider\_postnatal\_4 as

select

Provider,

Quant\_postnatal\_crrntyr\_STD,

Quant\_postnatal\_crrntyr\_INTM,

Quant\_postnatal\_crrntyr\_INTS,

Quant\_postnatal\_crrntyr\_STD+Quant\_postnatal\_crrntyr\_INTM+Quant\_postnatal\_crrntyr\_INTS as TotalQuant\_postnatal

from MPP\_IA\_Provider\_postnatal\_3

;

**quit**;

/\* Join the quantum table of three phase into one quantum table \*/

**proc** **sql**;

create table MPP\_IA\_Provider\_currtyr\_Quantum as

select

a.Provider

,a.Provider\_Type

,a.RC\_Name

,a.Quantum\_Delivery

,a.Quantum\_EBD

,b.TotalQuant\_antenatal

,c.TotalQuant\_postnatal

,a.Quantum\_Delivery+a.Quantum\_EBD+b.TotalQuant\_antenatal+c.TotalQuant\_postnatal as MPP\_IA\_Provider\_crryrQuant

from MPP\_IA\_Provider\_Delivery as a

left join MPP\_IA\_Provider\_antenatal\_4 as b

on a.Provider=b.Provider

left join MPP\_IA\_Provider\_postnatal\_4 as c

on a.Provider=c.Provider

;

**quit**;

/\* Export the MPP\_IA\_Provider\_currtyr\_Quantum table above \*/

%let outputfile=MPP\_IA\_Provider\_currtyr\_Quantum;

**proc** **export** data=work.MPP\_IA\_Provider\_currtyr\_Quantum

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

/\*5. Calculate Quantum of current year at delivery phase for Commissioners\*/

/\* Create HES based activity by Commissioner and HRG for delivery phase \*/

**proc** **sql**;

create table MPP\_IA\_Commissioner\_HRG\_Act\_1 as

SELECT substr(a.PURCODE,**1**,**3**) as Commissioner,

a.PROCODETD as Provider,

a.RC\_SpellHRG\_Rev as HRG

,sum(a.SPELLFLAG) as Activity

FROM

bprPrPP.Stage\_HES\_KeyFields as a

WHERE

a.Run\_ID=&Run\_ID. and

(a.Revised\_Excluded ='N'

AND substr(a.PURCODE,**1**,**1**) in ('0','1','9'))

AND a.SpellFlag <>**0**

AND

(a.RC\_SpellHRG\_Rev like 'NZ3%'or a.RC\_SpellHRG\_Rev like 'NZ4%' or a.RC\_SpellHRG\_Rev like 'NZ5%')

and

a.ADM <> 'UX'

group by substr(a.PURCODE,**1**,**3**), a.PROCODETD, a.RC\_SpellHRG\_Rev

order by substr(a.PURCODE,**1**,**3**), a.PROCODETD, a.RC\_SpellHRG\_Rev

;

**quit**;

/\* Join EBD\_activity into the inlier\_activity table \*/

**proc** **sql**;

create table MPP\_IA\_Commissioner\_HRG\_Act\_2 as

SELECT distinct

a.Commissioner,

a.HRG,

a.Activity,

b.EBD\_ACT as EBD\_Activity

FROM

MPP\_IA\_Commissioner\_HRG\_Act\_1 as a

left join work.MPP\_IA\_Provider\_HRG\_EBD as b

on a.HRG=b.HRG and a.Provider=b.Provider

;

**quit**;

**proc** **sql**;

create table MPP\_IA\_Commissioner\_HRG\_Act\_3 as

SELECT

Commissioner,

HRG,

sum(Activity) as Activity,

sum(EBD\_Activity) as EBD\_Activity

from

MPP\_IA\_Commissioner\_HRG\_Act\_2

group by Commissioner, HRG

;

**quit**;

**data** MPP\_IA\_Commissioner\_HRG\_Act\_4;

set MPP\_IA\_Commissioner\_HRG\_Act\_3;

if(EBD\_Activity=**.**) then EBD\_Activity=**0**;

**run**;

/\* Join activity table with Price linked table preparing for Quantum calculation\*/

**proc** **sql**;

create table MPP\_IA\_Commissioner\_HRG\_Act\_5 as

SELECT

a.Commissioner,

a.HRG,

a.Activity,

a.EBD\_Activity,

b.Combined\_DC\_EL\_NE as Price\_delivery,

b.Long\_stay\_payment as Price\_EBD

from

MPP\_IA\_Commissioner\_HRG\_Act\_4 as a

left join OUTSAS.MPP\_LINKEDSH\_DELIVERY\_HRG\_PRICES as b

on a.HRG=b.HRG\_Code

group by a.Commissioner,a.HRG

order by a.Commissioner

;

**quit**;

/\* Reconcile the number of activity back to targeted figures and then calculation quantum at delivery phase\*/

**proc** **sql** noprint;

select sum(Activity) into :Total\_Activity\_C

from MPP\_IA\_COMMISSIONER\_HRG\_ACT\_4

;

**quit**;

%let Total\_Activity\_C=&Total\_Activity\_C;

%put Total\_Activity\_C=&Total\_Activity\_C;

**proc** **sql** noprint;

select Total\_Activity into :Total\_Activity

from MPP\_DELIVERY\_ACTIVITY\_3

;

**quit**;

%let TotalBirth=&Total\_Activity;

%put TotalBirth=&Total\_Activity;

**proc** **sql** noprint;

select Total\_Activity into :Total\_Activity

from MPP\_DELIVERY\_ACTIVITY\_4

;

**quit**;

%let TotalEBD=&Total\_Activity;

**proc** **sql** noprint;

select sum(EBD\_Activity) into :Total\_EBD\_Act\_C

from MPP\_IA\_COMMISSIONER\_HRG\_ACT\_4

;

**quit**;

%let Total\_EBD\_Act\_C=&Total\_EBD\_Act\_C;

%put Total\_EBD\_Act\_C=&Total\_EBD\_Act\_C;

**proc** **sql**;

create table MPP\_IA\_Commssn\_Quant\_Delivery\_1 as

SELECT

Commissioner,

HRG,

Activity,

EBD\_Activity,

Price\_delivery,

Price\_EBD,

Activity\*Price\_delivery\*&TotalBirth/&Total\_Activity\_C as Quantum\_Inlier\_C,

EBD\_Activity\*Price\_EBD\*&TotalEBD/&Total\_EBD\_Act\_C as Quantum\_EBD\_C,

Activity\*Price\_delivery\*&TotalBirth/&Total\_Activity\_C+EBD\_Activity\*Price\_EBD\*&TotalEBD/&Total\_EBD\_Act\_C as Total\_Quantum\_Delivery\_C

from

MPP\_IA\_Commissioner\_HRG\_Act\_5

;

**quit**;

/\*Group the table above into Commision level\*/

**proc** **sql**;

create table MPP\_IA\_Commssn\_Quant\_Delivery\_2 as

SELECT

Commissioner,

sum(Quantum\_Inlier\_C) as Quantum\_Inlier\_C ,

sum(Quantum\_EBD\_C) as Quantum\_EBD\_C,

sum(Total\_Quantum\_Delivery\_C) as Total\_Quant\_Delivery\_C

from

MPP\_IA\_Commssn\_Quant\_Delivery\_1

group by Commissioner

;

**quit**;

/\* 6. Calculate Quantum at current year for Antenatal phase at Commissioner level\*/

/\* Aggregate activity from HRG level into Commissioner level \*/

**proc** **sql**;

create table MPP\_IA\_Commissioner\_HRG\_Act\_6 as

SELECT

Commissioner,

sum(Activity\*&TotalBirth/&Total\_Activity\_C) as Act\_Inlier\_C,

sum(EBD\_Activity\*&TotalEBD/&Total\_EBD\_Act\_C) as Act\_EBD\_C,

sum(Activity\*&TotalBirth/&Total\_Activity\_C+EBD\_Activity\*&TotalEBD/&Total\_EBD\_Act\_C) as Total\_Act\_Delivery\_C

from

MPP\_IA\_Commissioner\_HRG\_Act\_5

group by Commissioner

;

**quit**;

/\* Adjust activity number into those at antenatal phase by uplifting births and dividing into three levels \*/

**proc** **sql** noprint;

select Value format = Best12.8 into :Upliftbirths

from PARMPP.MPP\_STILLBIRTHS\_PARA

;

**quit**;

**proc** **sql** noprint;

select Antenatal format = Best12.8 into :ACT\_percent\_ST

from PARMPP.MPP\_ACTIVITY\_PARA

where Level in ('Standard')

;

**quit**;

**proc** **sql** noprint;

select Antenatal format = Best12.8 into :ACT\_percent\_Intm

from PARMPP.MPP\_ACTIVITY\_PARA

where Level in ('Intermediate')

;

**quit**;

**proc** **sql** noprint;

select Antenatal format = Best12.8 into :ACT\_percent\_Ints

from PARMPP.MPP\_ACTIVITY\_PARA

where Level in ('Intensive')

;

**quit**;

%let Upliftbirths=&Upliftbirths;

%put Upliftbirths=&Upliftbirths;

%let ACT\_percent\_ST=&ACT\_percent\_ST;

%put ACT\_percent\_ST=&ACT\_percent\_ST;

%let ACT\_percent\_Intm =&ACT\_percent\_Intm;

%put ACT\_percent\_Intm =&ACT\_percent\_Intm;

%let ACT\_percent\_Ints =&ACT\_percent\_Ints;

%put ACT\_percent\_Ints =&ACT\_percent\_Ints;

**proc** **sql**;

create table MPP\_IA\_Commissioner\_HRG\_Act\_7 as

select

Commissioner,

Total\_Act\_Delivery\_C\*(**1**+&Upliftbirths)\*&ACT\_percent\_ST as Act\_antenatal\_standard\_C,

Total\_Act\_Delivery\_C\*&ACT\_percent\_Intm as Act\_antenatal\_Intemediate\_C,

Total\_Act\_Delivery\_C\*&ACT\_percent\_Ints as Act\_antenatal\_Intensive\_C

from

MPP\_IA\_Commissioner\_HRG\_Act\_6

;

**quit**;

/\* Caculate quantum at antenatal phase for Commissioner \*/

/\* Retrive current year prices from linked table produced in Step 8 \*/

**proc** **sql** noprint;

select Tariff format = Best12.8 into :Price\_crrntyr\_ST

from OUTSAS.MPP\_LINKEDSH\_ANTENATAL

where Name in ('Standard')

;

**quit**;

**proc** **sql** noprint;

select Tariff format = Best12.8 into :Price\_crrntyr\_INTM

from OUTSAS.MPP\_LINKEDSH\_ANTENATAL

where Name in ('Intermediate')

;

**quit**;

**proc** **sql** noprint;

select Tariff format = Best12.8 into :Price\_crrntyr\_INTS

from OUTSAS.MPP\_LINKEDSH\_ANTENATAL

where Name in ('Intensive')

;

**quit**;

%let Price\_crrntyr\_ST=&Price\_crrntyr\_ST;

%put Price\_crrntyr\_ST=&Price\_crrntyr\_ST;

%let Price\_crrntyr\_INTM =&Price\_crrntyr\_INTM;

%put Price\_crrntyr\_INTM =&Price\_crrntyr\_INTM;

%let Price\_crrntyr\_INTS=&Price\_crrntyr\_INTS;

%put Price\_crrntyr\_INTS=&Price\_crrntyr\_INTS;

/\* Calculate current year quantum at antenatal phase for Commissioner\*/

**proc** **sql**;

create table MPP\_IA\_Commssn\_Quant\_antenatal\_1 as

select

Commissioner,

Act\_antenatal\_standard\_C\*&Price\_crrntyr\_ST as Quant\_antenatal\_crrntyr\_STD\_C,

Act\_antenatal\_Intemediate\_C\*&Price\_crrntyr\_INTM as Quant\_antenatal\_crrntyr\_INTM\_C,

Act\_antenatal\_Intensive\_C\*&Price\_crrntyr\_INTS as Quant\_antenatal\_crrntyr\_INTS\_C

from MPP\_IA\_Commissioner\_HRG\_Act\_7

;

**quit**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 05 May 2017 ;

\* Modified by: PID ;

\* Date: May 2018 ;

# \* MPP\_Step\_1\_Input Tables\_Lev1.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

/\* Produces input tables used in tariff calculation for Maternity Pathway Payment (MPP)

Steps\*/

%let RefCost\_Selected=Tariff\_R.REFERENCE\_COST\_1617\_PUBLISHED;

**data** TC\_MFF\_Selected;

set Tariff\_R.TC\_MFF;

where Year="&Year.";

Target\_MFF=Uncapped\_MFF;

**run**;

/\*1) Creates APC\_Input tables by retriving data from stage table in APC model file \*/

/\*Input source table comes from APC Model: APCSTAGE.APC\_HRGLEVELDATA

APC Model step 2.4

\*/

**proc** **sql**;

create table work.MPP\_APC\_Input as

SELECT

HRG,

ADMISSION,

CLEAN\_INLIER\_ACT AS INLIER\_ACT,

CLEAN\_INLIER\_TC AS INLIER\_COST,

CLEAN\_EBD\_ACT AS EBD\_ACT,

CLEAN\_EBD\_TC AS EBD\_COST,

CLEAN\_TC,

Clean\_Inlier\_UC

FROM apcstage.APC\_HRGLEVELDATA as rfa

WHERE HRG like ('NZ%')

ORDER BY HRG

;

**quit**;

%let outputfile=MPP\_APC\_Input;

**proc** **export** data=work.MPP\_APC\_Input

outfile="&inputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

/\*2) Creates OPROC\_Input tables by retriving data from stage table in OPROC model file \*/

**proc** **sql**;

create table MPP\_OPROC\_Input as

SELECT

CURRENCYcode as HRG,

TOTAL\_ACT AS ACT,

TC

FROM

tariff\_r.OPROC\_National\_Data\_1920

WHERE

HRG like ('NZ%')

ORDER BY HRG

;

**quit**;

%let outputfile=MPP\_OPROC\_Input;

**proc** **export** data=work.MPP\_OPROC\_Input

outfile="&inputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

/\*3) Creates Community\_Input tables by retriving data from RCNonAcute dataset \*/

**proc** **sql**;

create table MPP\_Community\_Input as

SELECT

a.CurrencyCode as HRG,

a.Department\_code as DepartmentCode,

a.Org\_Code as OrganisationCode,

a.Activity,

a.Unit\_Cost/ b.Target\_MFF as UC\_Excl\_MFF,

a.Activity\*(Calculated UC\_Excl\_MFF) as TC

FROM

Tariff\_R.REFCOSTNONACUTE\_FLAGED\_1920 as a

left join TC\_MFF\_Selected as b

on a.Org\_Code=b.Tariff\_Code

WHERE

CurrencyCode in ('ASNNS','N01A','N01P') or

CurrencyCode like 'NZ%' and Department\_code in ('CHS')

order by HRG

;

**quit**;

%let outputfile=MPP\_Community\_Input;

**proc** **export** data=work.MPP\_Community\_Input

outfile="&inputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

/\*4) Creates OPATT\_Input tables by retriving data from stage tables in OPATT model file \*/

**proc** **sql**;

create table MPP\_OPATT\_Input\_ACT as

SELECT

Tariff\_TFC\_code,

\*

FROM

STAGE\_OPATT\_ACTIVITY\_CROSSTAB

WHERE

Tariff\_TFC\_code in ('501','560')

;

**quit**;

**proc** **sql**;

create table MPP\_OPATT\_Input\_Cost as

SELECT

Tariff\_TFC\_code,

\*

from

output\_OPATT\_FinalTariff

WHERE

Tariff\_TFC\_code in ('501','560')

;

**quit**;

**proc** **sql**;

create table MPP\_OPATT\_Input

as

select

a.Tariff\_TFC\_code,

a.Total\_Activity,

b.Total\_Costs,

a.CL\_FAM as CL\_FAM\_ACT,

a.CL\_FAS as CL\_FAS\_ACT,

a.CL\_FUM as CL\_FUM\_ACT,

a.CL\_FUS as CL\_FUS\_ACT,

b.CL\_FAM as CL\_FAM\_TC,

b.CL\_FAS as CL\_FAS\_TC,

b.CL\_FUM as CL\_FUM\_TC,

b.CL\_FUS as CL\_FUS\_TC

from MPP\_OPATT\_Input\_ACT as a

left join MPP\_OPATT\_Input\_Cost as b

on a.Tariff\_TFC\_code=b.Tariff\_TFC\_code

;

**quit**;

%let outputfile=MPP\_OPATT\_Input;

**proc** **export** data=work.MPP\_OPATT\_Input

outfile="&inputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

/\*Retrive tables from HES dataset \*/

**proc** **sql**;

create table MPP\_HES\_Activity\_1 as

SELECT a.RC\_SpellHRG\_Rev as HRG

,a.ADM as Admission

,sum(a.SPELLFLAG) as Activity

FROM

bprPrPP.Stage\_HES\_KeyFields as a

WHERE

a.Run\_ID=&Run\_ID. and

(a.Revised\_Excluded ='N'

AND substr(a.PROCODETD,**1**,**1**) in ('5','R','T'))

AND a.SpellFlag <>**0**

AND

a.RC\_SpellHRG\_Rev like 'NZ%'

and

a.ADM <> 'UX'

group by a.RC\_SpellHRG\_Rev, a.ADM

order by a.RC\_SpellHRG\_Rev, a.ADM

;

**quit**;

**PROC** **TRANSPOSE** data=MPP\_HES\_Activity\_1 out=MPP\_HES\_Activity\_1;

by HRG;

id Admission;

var Activity;

**RUN**;

/\* Sum up activity for DC and EL\*/

**proc** **sql**;

create table MPP\_HES\_Activity\_2 as

select HRG

, case DC

when **.** then **0**

else DC

end as DC

,EL

,NE

from work.MPP\_hes\_activity\_1

;

**quit**;

**proc** **sql**;

create table MPP\_HES\_Activity\_3 as

select HRG

,sum(DC,EL) as Elective\_Activity

,sum(NE) as Non\_Elective\_Activity

from work.MPP\_hes\_activity\_2

group by HRG

order by HRG

;

**quit**;

/\* Export results \*/

%let outputfile=MPP\_HES\_Activity;

**proc** **export** data=work.MPP\_HES\_Activity\_3

outfile="&inputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 05 May 2017 ;

\* Modified by: PID ;

\* Date: May 2018 ;

\* version: 1.02 ;

# \* MPP\_Step\_2\_ Calculation of Parameters\_Lev1.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* Step 2: Produces parameterisation used in tariff calculation for Maternity Pathway Payment (MPP)

1) Import Currency Design table

2) Import Assumptions - Still Births percentage table

3) Import Assumptions - Activity percentages between three tiers

4) Import Assumptions - Unit Cost percentages between three tiers

5) Import Price Factor table

;

/\*1) Import Currency Design table\*/

**proc** **import** out=MPP\_Par\_CDTable\_1

datafile= "&datafilepath.\&TCurrenDesign."

dbms=xlsx replace;

sheet="Currency Design";

getnames=yes;

**run**;

**proc** **sql**;

create table ParMPP.MPP\_Par\_CDTable

as

select

MPP\_CDID,

Year,

HRG,

Description,

Pathway,

Delivery\_Complications\_Flag

from MPP\_Par\_CDTable\_1

where Year = "&FYear." and MPP\_CDID=&MPP\_CDID.

;

**quit**;

/\*2) Import Still Births percentage table\*/

**proc** **import** out=MPP\_StillBirths\_Para\_1

datafile= "&datafilepath.\&MPP\_ParameterTable."

dbms=xlsx replace;

sheet="stillbirths\_para";

getnames=yes;

**run**;

**data** ParMPP.MPP\_StillBirths\_Para;

set MPP\_StillBirths\_Para\_1;

where Year = "&FYear.";

**run**;

/\*3) Import Assumptions - Activity percentages between three tiers\*/

**proc** **import** out=MPP\_Activity\_Para\_1

datafile= "&datafilepath.\&MPP\_ParameterTable."

dbms=xlsx replace;

sheet="activity\_para";

getnames=yes;

**run**;

**data** ParMPP.MPP\_Activity\_Para;

set MPP\_Activity\_Para\_1;

where Year = "&FYear.";

**run**;

/\* 4)Import Unit Cost percentages between three tiers\*/

**proc** **import** out=MPP\_UnitCost\_Para\_1

datafile= "&datafilepath.\&MPP\_ParameterTable."

dbms=xlsx replace;

sheet="unitcost\_para";

getnames=yes;

**run**;

**data** ParMPP.MPP\_UnitCost\_Para;

set MPP\_UnitCost\_Para\_1;

where Year = "&FYear.";

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 05 May 2017 ;

\* Modified by: PID ;

\* Date: May 2018 ;

# \* MPP\_Aggregations.sas ;

\* description:

;

\*

\* ---PID changed the method of calculating activity for level 6 on 24 May 2018

according to NHSE request.;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* Step 3: Aggregates inputs into two/three tiers by using assumptions to prepare for unit cost calculation

1) Total Births table

1.1 Calculates the total births for APC

1.2 Calculates the total births for OPROC

1.3 Calculates the total births for Community

1.4 Sum up three tables above to the total births

2) Deliver phase: Calculation of total activity and total cost

2.1 Aggregates APC\_input tables for Delivery phase

2.2 Aggregates OPROC input tables for Delivery phase

2.3 Aggregates Community input tables for Delivery phase

2.4 Sum up three aggregated tables above to calculate total activity and total cost in Delivery phase

2.5 Sum up three aggregated tables above and then filter the data to get total activity and cost for those with CC

2.6 Calculates total activity and cost for those without CC

2.7 Calculate proportions of RC\_activity between those HRGs with CC and without CC

2.8 Splits HES based activity(see section 1.4) into with CC and without CC by using the proportions above

2.9 Creates a summary table by adding the EBD table

3) Antenatal phase: Calculation of total activity and total cost

3.1 Calculates two kinds of activity (Total Assumed and Total Weighted)at three levels by using assumptions

3.2 Calculates total costs at three levels by using assumptions

3.3 Create a summary table to get total costs at antenatal phase

4) Postnatal phase: Calculation of total activity and total cost

4.1 Calculates two kinds of activity (Total Assumed and Total Weighted)at three levels by using assumptions

4.2 Calculates total costs at three levels by using assumptions in Postnatal phase

4.3 Creates a summary table to get total costs at postnatal phase

;

/\* 1) Creates Total Births table \*/

/\* 1.1 Calculates the total births for APC\*/

**proc** **sql**;

create table TotalBirths\_APC\_1 as

select

(b.Delivery\_Complications\_Flag) as CCFlag,

sum(a.Elective\_Activity)+sum(a.Non\_Elective\_Activity) as APC\_Births\_1

from work.MPP\_HES\_Activity\_3 as a

left join ParMPP.MPP\_Par\_CDTable as b

on a.HRG = b.HRG

WHERE

b.Pathway in ("Delivery")

group by b.Delivery\_Complications\_Flag

;

**quit**;

**proc** **sql**;

create table TotalBirths\_APC\_4

as

select sum(APC\_Births\_1) as APC\_Births,

**1** as ID

from TotalBirths\_APC\_1

;

**quit**;

/\* 1.2 Calculates the total births for OPROC \*/

**proc** **sql**;

create table TotalBirths\_OPROC\_1 as

select

(b.Delivery\_Complications\_Flag) as CCFlag,

sum(a.Act) as OPROC\_Births

from work.MPP\_OPROC\_Input as a

left join ParMPP.MPP\_Par\_CDTable as b

on a.HRG = b.HRG

WHERE

b.Pathway in ("Delivery")

group by b.Delivery\_Complications\_Flag;

**quit**;

**proc** **sql**;

create table TotalBirths\_OPROC\_4

as

select sum(OPROC\_Births) as OPROC\_Births,

**1** as ID

from TotalBirths\_OPROC\_1

;

**quit**;

/\* 1.3 Calculates the total births for Community \*/

**proc** **sql**;

create table TotalBirths\_Comm\_1 as

select

(b.Delivery\_Complications\_Flag) as CCFlag,

sum(a.Activity) as Comm\_Births

from WORK.mpp\_community\_input as a

left join ParMPP.MPP\_Par\_CDTable as b

on a.HRG = b.HRG

WHERE

b.Pathway in ("Delivery")

group by b.Delivery\_Complications\_Flag;

;

**quit**;

**proc** **sql**;

create table TotalBirths\_Comm\_4

as

select sum(Comm\_Births) as Comm\_Births,

**1** as ID

from TotalBirths\_Comm\_1

;

**quit**;

/\* 1.4 Sum up three tables above to the total births \*/

**proc** **sql**;

create table AGGMPP.TotalBirths\_5 as

select

a.ID,

a.APC\_Births,

b.OPROC\_Births,

c.Comm\_Births,

sum(a.APC\_Births,b.OPROC\_Births,c.Comm\_Births) as Totalbirths

from TotalBirths\_APC\_4 as a

left join TotalBirths\_OPROC\_4 as b

on a.ID = b.ID

left join TotalBirths\_Comm\_4 as c

on a.ID = c.ID

group by a.ID

;

**quit**;

%let outputfile=MPP\_TotalBirths;

**proc** **export** data=AGGMPP.TotalBirths\_5

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

/\*----------------- on 23 May 2018 According to request by NHSE, the method below is suspended.

2) Calculates total activity and total cost for Delivery phase\*/

/\* 2.1 Aggregates APC\_input tables for Delivery phase\*/

/\* An Important Note: In Excel based MPP model, the Total Cost in Delivery phase is the total cost included EBD costs,

which is inconsistent with the number of activity used in calculation of unit cost. This is an inherited error. The error has been corrected

proc sql;

create table APC\_AGG\_Delivery as

SELECT

b.Delivery\_Complications\_Flag as CCFlag,

sum(a.Inlier\_Act) as Inlier\_Activity,

sum(a.Inlier\_Cost) as Inlier\_TC,

sum(a.EBD\_ACT) as EBD\_ACT ,

sum(a.EBD\_Cost) as EBD\_TC

FROM

work.mpp\_apc\_input as a

left join ParMPP.MPP\_Par\_CDTable as b

on a.HRG = b.HRG

WHERE

b.Pathway in ("Delivery")

group BY b.Delivery\_Complications\_Flag;

quit;

/\* 2.2 Aggregates OPROC input tables for Delivery phase

proc sql;

create table OPROC\_AGG\_Delivery as

SELECT

b.Delivery\_Complications\_Flag as CCFlag,

sum(a.Act) as OPROC\_Activity,

sum(a.TC) as OPROC\_TC

FROM

work.mpp\_oproc\_input as a

left join ParMPP.MPP\_Par\_CDTable as b

on a.HRG = b.HRG

WHERE

b.Pathway in ("Delivery")

group BY b.Delivery\_Complications\_Flag;

quit;

/\* 2.3 Aggregates Community input tables for Delivery phase

proc sql;

create table Comm\_AGG\_Delivery as

SELECT

b.Delivery\_Complications\_Flag as CCFlag,

sum(a.Activity) as Comm\_Activity,

sum(a.TC) as Comm\_TC

FROM

WORK.MPP\_COMMUNITY\_INPUT as a

left join ParMPP.MPP\_Par\_CDTable as b

on a.HRG = b.HRG

WHERE

(a.HRG in ('ASNNS','N01A','N01P') or

a.HRG like 'NZ%' and

a.DepartmentCode in ('CHS'))

and b.Pathway in ("Delivery")

group BY CCFlag

order by CCFlag;

quit;

/\* 2.4 Sum up three aggregated tables above to calculate total activity and total cost in Delivery phase

proc sql;

create table MPP\_Delivery\_Activity\_1 as

select distinct

a.CCFlag,

a.Inlier\_TC,

a.Inlier\_Activity,

a.EBD\_ACT,

a.EBD\_TC,

b.OPROC\_Activity,

b.OPROC\_TC,

c.Comm\_Activity,

c.Comm\_TC

from APC\_AGG\_Delivery as a

left join OPROC\_AGG\_Delivery as b

on a.CCFlag = b.CCFlag

left join Comm\_AGG\_Delivery as c

on a.CCFlag = c.CCFlag

order by a.CCFlag

;

quit;

proc sql;

create table MPP\_Delivery\_Activity\_2

as

select

put(a.CCFlag,8.) as CCFlag format=$30. length=30,

a.Inlier\_Activity,

a.OPROC\_Activity,

a.Comm\_Activity,

sum(a.Inlier\_Activity,a.OPROC\_Activity,a.Comm\_Activity) as Total\_Activity,

a.Inlier\_TC,

a.OPROC\_TC,

a.Comm\_TC,

sum(a.Inlier\_TC,a.OPROC\_TC,a.Comm\_TC) as Total\_TC

from MPP\_Delivery\_Activity\_1 as a

order by a.CCFlag

;

quit;

/\*1.5 Inliner Summary

proc sql;

create table MPP\_Delivery\_Activity\_3

as

select distinct

"Total Deliveries (Inlier)" as CCFlag format=$30. length=30,

sum(a.Inlier\_Activity) as Inlier\_Activity,

sum(a.OPROC\_Activity) as OPROC\_Activity,

sum(a.Comm\_Activity) as Comm\_Activity,

sum(a.Total\_Activity) as Total\_Activity,

sum(a.Inlier\_TC) as Inlier\_TC,

sum(a.OPROC\_TC) as OPROC\_TC,

sum(a.Comm\_TC) as Comm\_TC,

sum(Total\_TC) as Total\_TC

from MPP\_Delivery\_Activity\_2 as a

;

quit;

/\* 1.6 EBD summary

proc sql;

create table EBD\_Delivery\_TC

as

select

CCFlag,

EBD\_ACT,

EBD\_TC

from APC\_AGG\_Delivery

order by CCFlag

;

quit;

proc sql;

create table MPP\_Delivery\_Activity\_4

as

select

"Excess Bed Days" as CCFlag format=$30. length=30,

sum(EBD\_ACT) as Total\_Activity,

sum(EBD\_TC) as Total\_TC

from

EBD\_Delivery\_TC

;

quit;

/\* Combined all above

data MPP\_Summ\_Delivery;

set

MPP\_Delivery\_Activity\_2

MPP\_Delivery\_Activity\_3

MPP\_Delivery\_Activity\_4

;

run;

%let outputfile=MPP\_Summ\_Delivery;

proc export data=MPP\_Summ\_Delivery

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

run;

\*/

/\* 2.7 Calculate proportions of RC\_activity between those HRGs with CC and without CC.

The proportions are to be used in split HES based total births into withCC and without CC in the next stage\*/

/\* Total\_Activity for calculation

proc sql noprint;

select Total\_Activity into :Total\_Activity

from MPP\_Summ\_Delivery

where CCFlag="Total Deliveries (Inlier)"

;

quit;

%let Total\_Activity=&Total\_Activity.;

/\*Totalbirths for calculation

proc sql noprint;

select Totalbirths into :Totalbirths

from AGGMPP.TotalBirths\_5

;

quit;

%let Totalbirths=&Totalbirths;

/\*Calculated Proportion and HES\_Activity

proc sql;

create table RC\_AGG\_Delivery\_1 as

SELECT

a.\*,

a.Total\_Activity/&Total\_Activity. as Proportion,

&Totalbirths\*(calculated Proportion) as HES\_Activity

from MPP\_Summ\_Delivery as a

where CCFlag^="Excess Bed Days"

;

quit;

/\* RC Hes -- Inlier

proc sql;

create table RC\_AGG\_Delivery\_2

as

select

CCFlag,

Total\_Activity as RC\_Delivery\_Activity,

Proportion,

HES\_Activity as HES\_Delivery\_Activity ,

Total\_TC as TC\_Delivery

from RC\_AGG\_Delivery\_1

where CCFlag^="Total Deliveries (Inlier)"

;

quit;

/\*RC Hes -- EBD only

proc sql;

create table RC\_AGG\_Delivery\_3

as

select

CCFlag,

. as RC\_Delivery\_Activity,

. as Proportion,

Total\_Activity as HES\_Delivery\_Activity ,

Total\_TC as TC\_Delivery

from MPP\_Summ\_Delivery

where CCFlag="Excess Bed Days"

;

quit;

data AGGMPP.MPP\_RC\_Summ\_Delivery;

set RC\_AGG\_Delivery\_2

RC\_AGG\_Delivery\_3

;

run;

%let outputfile=MPP\_RC\_Summ\_Delivery;

proc export data=AGGMPP.MPP\_RC\_Summ\_Delivery

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

run;

/\*---------End-----\*/

/\*-----------------------------24/05/2018 According to request by NHSE, we use same method as level 1 to generate APC\_activity--------\*/

/\* 1.1 Calculates the APC Activity for Delivery\*/

**proc** **sql**;

create table APC\_HES\_Delivery\_Activity as

select

a.HRG,

sum(a.Elective\_Activity,a.Non\_Elective\_Activity) as APC\_Activity

from work.MPP\_HES\_Activity\_3 as a

left join ParMPP.MPP\_Par\_CDTable as b

on a.HRG = b.HRG

WHERE

b.Pathway in ("Delivery")

order by a.HRG

;

**quit**;

**proc** **sql**;

create table APC\_Delivery\_TC as

SELECT

a.HRG,

sum(a.Inlier\_Cost) as Inlier\_TC,

sum(a.Inlier\_ACT) as Inlier\_Activity

FROM

work.mpp\_apc\_input as a

left join ParMPP.MPP\_Par\_CDTable as b

on a.HRG = b.HRG

WHERE

b.Pathway in ("Delivery")

group by a.HRG

order by a.HRG

;

**quit**;

**proc** **sql**;

create table APC\_Deli\_Act\_TC

as

select

a.HRG,

a.APC\_Activity,

b.Inlier\_TC,

b.Inlier\_Activity

from APC\_HES\_Delivery\_Activity as a

left join

APC\_Delivery\_TC as b

on a.HRG=b.HRG

order by a.HRG

;

**quit**;

**proc** **sql**;

create table EBD\_Delivery\_TC as

SELECT

a.HRG,

sum(a.EBD\_ACT) as Total\_Act,

sum(a.EBD\_Cost) as Total\_TC

FROM

work.mpp\_apc\_input as a

left join ParMPP.MPP\_Par\_CDTable as b

on a.HRG = b.HRG

WHERE

b.Pathway in ("Delivery")

group by a.HRG

order by a.HRG

;

**quit**;

/\* 1.2 Calculates the total activity and cost in delivery phase for OPROC \*/

**proc** **sql**;

create table OPROC\_Delivery as

select

a.HRG,

a.TC as OPROC\_TC,

a.Act as OPROC\_Activity

from work.MPP\_OPROC\_Input as a

left join ParMPP.MPP\_Par\_CDTable as b

on a.HRG = b.HRG

WHERE

b.Pathway in ("Delivery")

;

**quit**;

/\* 1.3 Calculates the activity and cost in Delivery phase for Community \*/

**proc** **sql**;

create table Comm\_Delivery as

select

a.HRG,

sum(a.UC\_Excl\_MFF\* a.Activity) as Comm\_TC,

sum( a.Activity) as Comm\_Activity

from WORK.mpp\_community\_input as a

left join ParMPP.MPP\_Par\_CDTable as b

on a.HRG = b.HRG

WHERE

b.Pathway in ("Delivery")

group by a.HRG

order by a.HRG;

;

**quit**;

/\* 1.4 Sum up three tables above to the total Activity for Delivery \*/

**proc** **sql**;

create table MPP\_Delivery\_Activity\_1 as

select distinct

a.HRG format=$30. length=**30**,

a.APC\_Activity,

a.Inlier\_TC,

a.Inlier\_Activity,

b.OPROC\_Activity,

b.OPROC\_TC,

c.Comm\_Activity,

c.Comm\_TC

from APC\_Deli\_Act\_TC as a

left join OPROC\_Delivery as b

on a.HRG = b.HRG

left join Comm\_Delivery as c

on a.HRG = c.HRG

order by a.HRG

;

**quit**;

**proc** **sql**;

create table MPP\_Delivery\_Activity\_2

as

select

a.HRG format=$30. length=**30**,

a.APC\_Activity,

a.OPROC\_Activity,

a.Comm\_Activity,

sum(a.APC\_Activity,a.OPROC\_Activity,a.Comm\_Activity) as Total\_Activity,

a.Inlier\_TC,

a.OPROC\_TC,

a.Comm\_TC,

sum(a.Inlier\_TC,a.OPROC\_TC,a.Comm\_TC) as Total\_TC

from MPP\_Delivery\_Activity\_1 as a

order by a.HRG

;

**quit**;

/\*1.5 Inliner Summary \*/

**proc** **sql**;

create table MPP\_Delivery\_Activity\_3

as

select

put(b.Delivery\_Complications\_Flag,**8.**) as Level length=**30**,

sum(a.APC\_Activity) as APC\_Activity,

sum(a.OPROC\_Activity) as OPROC\_Activity,

sum(a.Comm\_Activity) as Comm\_Activity,

sum(a.Total\_Activity) as Total\_Activity,

sum(a.Inlier\_TC) as Inlier\_TC,

sum(a.OPROC\_TC) as OPROC\_TC,

sum(a.Comm\_TC) as Comm\_TC,

sum(Total\_TC) as Total\_TC

from MPP\_Delivery\_Activity\_2 as a

left join ParMPP.MPP\_Par\_CDTable as b

on a.HRG = b.HRG

WHERE

b.Pathway in ("Delivery")

group by Delivery\_Complications\_Flag

;

**quit**;

/\* 1.6 EBD summary \*/

**proc** **sql**;

create table MPP\_Delivery\_Activity\_4

as

select

"Excess Bed Days" as Level format=$30. length=**30**,

sum(Total\_Act) as Total\_Activity,

sum(Total\_TC) as Total\_TC

from

EBD\_Delivery\_TC

;

**quit**;

**data** MPP\_Summ\_Delivery;

set

MPP\_Delivery\_Activity\_3

MPP\_Delivery\_Activity\_4

;

**run**;

%let outputfile=MPP\_Summ\_Delivery;

**proc** **export** data=MPP\_Summ\_Delivery

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

/\*------------End---------------------\*/

/\* 3) Calculates total activity and costs at three levels for MPP in Antenatal phase\*/

/\* 3.1 Calculates two kinds of activity (Total Assumed and Total Weighted)at three levels by using assumptions\*/

**proc** **sql**;

create table Activity\_Antenatal\_1 as

select

a.ID,

a.totalbirths,

sum(a.totalbirths)\*(**1**+b.value) as Act\_assumed

from AGGMPP.totalbirths\_5 as a

right join ParMPP.MPP\_stillbirths\_para as b

on a.ID=b.MPP\_SBID

group by value

;

**quit**;

**proc** **sql**;

create table Activity\_Antenatal\_2 as

select

a.ID,

b.Level,

b.Antenatal,

a.Act\_assumed as Totact\_assumed,

a.Act\_assumed\*antenatal as Act\_assumed

from work.Activity\_Antenatal\_1 as a

right join ParMPP.MPP\_activity\_para as b

on a.ID=b.MPP\_ACTID

Order by Level

;

**quit**;

**proc** **sql**;

create table AGGMPP.Act\_Antenatal as

select

a.ID,

a.Level,

a.Totact\_assumed,

a.Act\_assumed,

sum(a.Act\_assumed\*b.Antenatal) as Totact\_weighted,

a.Act\_assumed\*b.Antenatal as Act\_weighted

from work.Activity\_Antenatal\_2 as a

right join ParMPP.MPP\_unitcost\_para as b

on a.Level=b.Level

;

**quit**;

%let outputfile=MPP\_Act\_Antenatal;

**proc** **export** data=AGGMPP.Act\_Antenatal

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

/\* 3.2 Calculates total costs at three levels by using assumptions\*/

/\* Calculates APC\_total costs at antenatal phase\*/

**proc** **sql**;

create table APC\_AGG\_Antenatal as

SELECT

b.Delivery\_Complications\_Flag as CCFlag,

sum(a.Clean\_TC) as APC\_TC

FROM

MPP\_APC\_Input as a

left join ParMPP.MPP\_Par\_CDTable as b

on a.HRG = b.HRG

WHERE

b.Pathway in ("Antenatal")

group BY CCFlag;

**quit**;

/\* Calculate OPROC\_total costs at antenatal phase\*/

**proc** **sql**;

create table OPROC\_AGG\_Antenatal as

SELECT

b.Delivery\_Complications\_Flag as CCFlag,

sum(a.TC) as OPROC\_TC

FROM

MPP\_OPROC\_Input as a

left join ParMPP.MPP\_Par\_CDTable as b

on a.HRG = b.HRG

WHERE

b.Pathway in ("Antenatal")

group BY b.Delivery\_Complications\_Flag

;

**quit**;

/\* Calculates Community\_total costs at antenatal phase\*/

**proc** **sql**;

create table Comm\_AGG\_Antenatal as

SELECT

b.Delivery\_Complications\_Flag as CCFlag,

sum(a.TC) as Comm\_TC

FROM

MPP\_Community\_Input as a

left join ParMPP.MPP\_Par\_CDTable as b

on a.HRG = b.HRG

WHERE

(a.HRG in ('ASNNS','N01A','N01P') or

a.HRG like 'NZ%' and

a.DepartmentCode in ('CHS'))

and b.Pathway in ("Antenatal")

group BY CCFlag;

**quit**;

/\* Calculates OPATT\_total costs at antenatal phase\*/

**proc** **sql**;

create table OPATT\_AGG\_Antenatal as

Select

**.** as CCFlag,

sum(a.Total\_Costs) as OPATT\_TC

From MPP\_OPATT\_Input as a

group by CCFlag

;

**quit**;

/\* 3.3 Creates a summary table to get total costs at antenatal phase\*/

/\*Note: According to requests byNHSE, the amount of quantum related to specialised fetal medician should be taken from the total cost at antenatal phase.\*/

**proc** **sql**;

create table AGGMPP.TC\_AGG\_Antenatal as

SELECT

case a.CCFlag

when **.** then **1**

else a.CCFlag

end as CCFlag,

sum(a.APC\_TC,b.OPROC\_TC,c.Comm\_TC,d.OPATT\_TC) as TC\_Antenatal

from work.APC\_AGG\_Antenatal as a

left join work.OPROC\_AGG\_Antenatal as b

on a.CCFlag=b.CCFlag

left join work.Comm\_AGG\_Antenatal as c

on a.CCFlag=c.CCFlag

left join work.OPATT\_AGG\_Antenatal as d

on a.CCFlag=d.CCFlag

;

**quit**;

%let outputfile=MPP\_TC\_AGG\_Antenatal;

**proc** **export** data=AGGMPP.TC\_AGG\_Antenatal

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

/\* 4) Calculates total activity and costs at three levels for MPP in Postnatal phase\*/

/\* 4.1 Calculates two kinds of activity (Total Assumed and Total Weighted)at three levels by using assumptions\*/

**proc** **sql**;

create table Activity\_Postnatal\_1 as

select

ID

,sum(Totalbirths) as Totalbirths

from AGGMPP.totalbirths\_5

group by ID

;

**quit**;

**proc** **sql**;

create table Activity\_Postnatal\_2 as

select

a.ID as CCFlag,

b.Level,

a.Totalbirths as Totact\_assumed,

a.Totalbirths\*b.Postnatal as Act\_assumed

from Activity\_Postnatal\_1 as a

left join ParMPP.MPP\_activity\_para as b

on a.ID=b.MPP\_ACTID

group by ID

order by b.Level

;

**quit**;

**proc** **sql**;

create table AGGMPP.Act\_Postnatal as

select

a.CCFlag,

a.Level,

a.Totact\_assumed,

a.Act\_assumed,

sum(a.Act\_assumed\*b.Postnatal) as Totact\_weighted,

a.Act\_assumed\*b.Postnatal as Act\_weighted

from work.Activity\_Postnatal\_2 as a

left join ParMPP.MPP\_unitcost\_para as b

on a.Level=b.Level

order by a.Level

;

**quit**;

%let outputfile=MPP\_Act\_Postnatal;

**proc** **export** data=AGGMPP.Act\_Postnatal

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

/\* 4.2 Calculate total costs at three levels by using assumptions in Postnatal phase\*/

/\* Calculate APC\_total costs at postnatal phase\*/

**proc** **sql**;

create table AGGMPP.APC\_AGG\_Postnatal as

SELECT

b.Delivery\_Complications\_Flag as CCFlag,

sum(a.CLEAN\_TC) as APC\_TC

FROM

work.MPP\_APC\_Input as a

left join ParMPP.MPP\_Par\_CDTable as b

on a.HRG = b.HRG

WHERE

b.Pathway in ("Postnatal")

group BY CCFlag;

**quit**;

/\* Calculates OPROC\_total costs at postnatal phase\*/

**proc** **sql**;

create table AGGMPP.OPROC\_AGG\_Postnatal as

SELECT

b.Delivery\_Complications\_Flag as CCFlag,

sum(a.TC) as OPROC\_TC

FROM

MPP\_OPROC\_Input as a

left join ParMPP.MPP\_Par\_CDTable as b

on a.HRG= b.HRG

WHERE

b.Pathway in ("Postnatal")

group BY CCFlag;

**quit**;

/\* Calculates Community\_total costs at postnatal phase\*/

**proc** **sql**;

create table AGGMPP.Comm\_AGG\_Postnatal as

SELECT

b.Delivery\_Complications\_Flag as CCFlag,

sum(a.TC) as Comm\_TC

FROM

WORK.MPP\_Community\_Input as a

left join ParMPP.MPP\_Par\_CDTable as b

on a.HRG = b.HRG

WHERE

(a.HRG in ('ASNNS','N01A','N01P') or

a.HRG like 'NZ%' and

a.DepartmentCode in ('CHS'))

and b.Pathway in ("Postnatal")

group BY CCFlag;

**quit**;

/\* 4.3 Creates a summary table to get total costs at postnatal phase\*/

**proc** **sql**;

create table AGGMPP.TC\_AGG\_Postnatal as

SELECT

case a.CCFlag

when **.** then **1**

else a.CCFlag

end as CCFlag,

sum(a.APC\_TC,b.OPROC\_TC,c.Comm\_TC) as TC\_Postnatal

from AGGMPP.APC\_AGG\_Postnatal as a

left join AGGMPP.OPROC\_AGG\_Postnatal as b

on a.CCFlag=b.CCFlag

left join AGGMPP.Comm\_AGG\_Postnatal as c

on a.CCFlag=c.CCFlag

;

**quit**;

%let outputfile=MPP\_TC\_AGG\_Postnatal;

**proc** **export** data=AGGMPP.TC\_AGG\_Postnatal

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 05 May 2017 ;

\* Modified by: PID ;

\* Date: May 2018 ;

\* version: 1.02 ;

# \* MPP\_Step\_4\_Calculation of Unit Cost\_Lev1.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* Step 4: Calculates unit costs for Maternity Pathway Payment (MPP)

1) Calculate Unit Cost for Delivery phase

2) Calculate Unit Cost for Antenatal phase

3) Calculate Unit Cost for Postnatal phase

;

/\*%macro MPP\_UnitCost(schema=);\*/

/\* 1)Calculate Unit Cost for Delivery phase\*/

/\* Calculate Inlier unit cost \*/

**proc** **sql**;

create table MPP\_Delivery\_UC

as

select

a.\*,

a.Total\_TC/a.Total\_Activity as Unit\_Cost

from MPP\_Summ\_Delivery as a

;

**quit**;

**data** UCMPP.MPP\_Delivery\_UC;

set MPP\_Delivery\_UC;

**run**;

/\* export the table\*/

%let outputfile=MPP\_Delivery\_UC;

**proc** **export** data=MPP\_Delivery\_UC

outfile="&ucpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

/\* 2) Calculate Unit Cost for Antenatal phase\*/

**proc** **sql**;

create table UC\_Antenatal\_1 as

SELECT

a.CCFlag,

b.Level,

b.Act\_assumed,

b.Act\_weighted,

a.TC\_Antenatal,

a.TC\_Antenatal/sum(b.Act\_weighted)\*c.Antenatal as UC\_Antenatal

from AGGMPP.TC\_AGG\_Antenatal as a

right join AGGMPP.Act\_Antenatal as b

on a.CCFlag=b.ID

left join ParMPP.MPP\_unitcost\_para as c

on b.Level=c.Level

order by Level

;

**quit**;

/\* Rank three tiers from Standard, intermediate to intensive\*/

**proc** **sort** data=UC\_Antenatal\_1 out=UCMPP.UC\_Antenatal;

by descending level;

**run**;

/\* export the table\*/

%let outputfile=MPP\_UC\_Antenatal;

**proc** **export** data=UCMPP.UC\_Antenatal

outfile="&ucpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

/\* 3) Calculate Unit Cost for Postnatal phase\*/

**proc** **sql**;

create table UC\_Postnatal\_1 as

SELECT

a.CCFlag,

b.Level,

b.Act\_assumed,

b.Act\_weighted,

a.TC\_Postnatal,

a.TC\_Postnatal/sum(b.Act\_weighted)\*c.Postnatal as UC\_Postnatal

from AGGMPP.TC\_AGG\_Postnatal as a

right join AGGMPP.Act\_Postnatal as b

on a.CCFlag=b.CCFlag

left join ParMPP.MPP\_unitcost\_para as c

on b.Level=c.Level

order by Level

;

**quit**;

/\* Rank three tiers from Standard, intermediate to intensive\*/

**proc** **sort** data=UC\_Postnatal\_1 out=UCMPP.UC\_Postnatal;

by descending level;

**run**;

/\* export the table\*/

%let outputfile=MPP\_UC\_Posnatal;

**proc** **export** data=UCMPP.UC\_Postnatal

outfile="&ucpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 05 June 2017 ;

\* Modified by: PID ;

\* Date: May 2018 ;

\* version: 1.02 ;

# \* MPP\_Step\_5\_1\_QR1 Adjustment\_Lev1.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* MPP\_QR1;

\* Calculate QR1 as in excel model;

\*Currently blank;

/\*Get QuantumRecValues value \*/

**Proc** **sql** noprint;

select QuantumRecValues into :QuantumRecValues

from Tariff\_R.QUANTUMRECONCILATION\_OUTPUT

where Year="&FYear." and POD="MatCashIO"

;

%let QuantumRecValues=&QuantumRecValues;

%put QuantumRecValues=&QuantumRecValues;

**quit**;

/\*calculate Pre\_QR1\_Quantum \*/

/\*Delivery \*/

**proc** **sql**;

select sum(Total\_TC) format=Best21.20 into :PreQR1Quan\_Delivery

from UCMPP.MPP\_Delivery\_UC

;

%let PreQR1Quan\_Delivery=&PreQR1Quan\_Delivery;

%put PreQR1Quan\_Delivery=&PreQR1Quan\_Delivery;

select distinct TC\_Antenatal format=Best21.20 into :PreQR1Quan\_Antenata

from UCMPP.UC\_Antenatal

;

%let PreQR1Quan\_Antenata=&PreQR1Quan\_Antenata;

%put PreQR1Quan\_Antenata=&PreQR1Quan\_Antenata;

select distinct TC\_Postnatal format=Best21.20 into :PreQR1Quan\_Postnatal

from UCMPP.UC\_POSTNATAL

;

%let PreQR1Quan\_Postnatal=&PreQR1Quan\_Postnatal;

%put PreQR1Quan\_Postnatal=&PreQR1Quan\_Postnatal;

;

**Quit**;

**data** MPP\_QR1;

PreQR1Quan\_Delivery=&PreQR1Quan\_Delivery. ;

PreQR1Quan\_Antenata=&PreQR1Quan\_Antenata.;

PreQR1Quan\_Postnatal=&PreQR1Quan\_Postnatal.;

Pre\_QR1\_Quantum=sum(PreQR1Quan\_Delivery,PreQR1Quan\_Antenata,PreQR1Quan\_Postnatal);

QuantumRecValues=&QuantumRecValues.;

QR1\_Adjustment=(QuantumRecValues/Pre\_QR1\_Quantum)-**1**;

**run**;

**proc** **sql** noprint;

select QR1\_Adjustment into :QR1\_Adjustment

from MPP\_QR1

;

**quit**;

/\* 1.QR1 adjustment for Delivery phase\*/

**PROC** **SQL**;

create table MPP\_Delivery\_QR1 as

select

a.\*,

&QR1\_Adjustment as QR1\_Adjustment,

Unit\_Cost \* (**1** + &QR1\_Adjustment) as Prices\_Delivery\_QR1

from MPP\_Delivery\_UC as a ;

**quit**;

\*/ Added output of data ;

%let outputfile=MPP\_Delivery\_QR1;

**proc** **export** data=WORK.MPP\_Delivery\_QR1

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

/\* 2. QR1 adjustment for Antenatal phase\*/

**PROC** **SQL**;

create table MPP\_Antenatal\_QR1 as

select

CCFlag as ID

,Level

,Act\_assumed

,UC\_Antenatal

,&QR1\_Adjustment as QR1\_Adjustment

,UC\_Antenatal \* (**1** + &QR1\_Adjustment) as Prices\_Antenatal\_QR1

from UCMPP.UC\_Antenatal;

**quit**;

\*/ Added output of data ;

%let outputfile=MPP\_Antenatal\_QR1;

**proc** **export** data=WORK.MPP\_Antenatal\_QR1

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

/\* 3. QR1 adjustment for Postnatal phase\*/

**PROC** **SQL**;

create table MPP\_Postnatal\_QR1 as

select

CCFlag as ID

,Level

,Act\_assumed

,UC\_Postnatal

,&QR1\_Adjustment as QR1\_Adjustment

,UC\_Postnatal \* (**1** + &QR1\_Adjustment) as Prices\_Postnatal\_QR1

from UCMPP.UC\_Postnatal;

**quit**;

\*/ Added output of data ;

%let outputfile=MPP\_Postnatal\_QR1;

**proc** **export** data=WORK.MPP\_Postnatal\_QR1

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 05 June 2017 ;

\* Modified by: PID ;

\* Date: May 2018 ;

\* version: 1.02 ;

# \* MPP\_Step\_5\_2\_CB Adjustment\_Lev1.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

%let CB\_Adjustment = 0;

/\* 1.Cost Base adjustment for Delivery phase\*/

**PROC** **SQL**;

create table MPP\_Delivery\_CB as

select

a.\*,

&CB\_Adjustment as CB\_Adjustment,

Prices\_Delivery\_QR1 \* (**1** + &CB\_Adjustment) as Prices\_Delivery\_CB

from MPP\_Delivery\_QR1 as a;

**quit**;

\*/ Added output of data ;

%let outputfile=MPP\_Delivery\_CB;

**proc** **export** data=WORK.MPP\_Delivery\_CB

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

/\* 2. Cost Base adjustment for Antenatal phase\*/

**PROC** **SQL**;

create table MPP\_Antenatal\_CB as

select

ID

,Level

,Act\_assumed

,UC\_Antenatal

,QR1\_Adjustment

,Prices\_Antenatal\_QR1

,&CB\_Adjustment as CB\_Adjustment

,Prices\_Antenatal\_QR1 \* (**1** + &CB\_Adjustment) as Prices\_Antenatal\_CB

from MPP\_Antenatal\_QR1;

**quit**;

\*/ Added output of data ;

%let outputfile=MPP\_Antenatal\_CB;

**proc** **export** data=WORK.MPP\_Antenatal\_CB

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

/\* 3. Case Base adjustment for Postnatal phase\*/

**PROC** **SQL**;

create table MPP\_Postnatal\_CB as

select

ID

,Level

,Act\_assumed

,UC\_Postnatal

,QR1\_Adjustment

,Prices\_Postnatal\_QR1

,&CB\_Adjustment as CB\_Adjustment

,Prices\_Postnatal\_QR1 \* (**1** + &CB\_Adjustment) as Prices\_Postnatal\_CB

from MPP\_Postnatal\_QR1;

**quit**;

\*/ Added output of data ;

%let outputfile=MPP\_Postnatal\_CB;

**proc** **export** data=WORK.MPP\_Postnatal\_CB

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* Author: PID ;

\* Date: 05 June 2017 ;

\* Modified by: PID ;

\* Date: May 2018 ;

\* version: 1.02 ;

# \* MPP\_Step\_5\_3\_TotalInflation\_Efficiency\_Lev1.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

/\* calcualtion --Inflation\_and\_Efficiency\_Combine -- Start from here \*/

/\* --------------------------Note- The calculation will be changed if we have new uplift factors ---------------------------- \*/

/\* Transpose print adjustment table \*/

**PROC** **TRANSPOSE** DATA=Tariff\_R.I\_E

OUT=Trans\_I\_E (drop=Source )

PREFIX=Year

NAME=Source

LABEL=Adjustment\_Type

;

ID Year;

VAR Inflation Efficiency Inflation\_and\_Efficiency\_Combine;

**RUN**; **QUIT**;

/\*Calculate uplift total \*/

**data** PriceAdj\_InflaEffici;

set Trans\_I\_E;

label Adjustment\_Type =Adjustment\_Type;

TotalUplift18\_19=(Year17\_18+**1**)\*(Year18\_19+**1**)-**1**;

label TotalUplift18\_19="Total uplift up to 18/19 price levels";

**run**;

**proc** **sql** noprint;

select TotalUplift18\_19 into :TotalUplift18\_19

from PriceAdj\_InflaEffici

where Adjustment\_Type="Inflation and Efficiency Combined"

;

**quit**;

%let Total\_I\_E\_Adjustment=&TotalUplift18\_19;

%put Total\_I\_E\_Adjustment=&Total\_I\_E\_Adjustment;

/\* calculation --Inflation\_and\_Efficiency\_Combine -- End here \*/

**PROC** **SQL**;

create table MPP\_Delivery\_Total\_I\_E as

select

a.\*,

&Total\_I\_E\_Adjustment as Total\_I\_E\_Adjustment,

Prices\_Delivery\_CB\*(**1**+&Total\_I\_E\_Adjustment) as Prices\_Delivery\_Total\_I\_E

from MPP\_Delivery\_CB as a

;

**quit**;

\*/ Added output of data ;

%let outputfile=MPP\_Delivery\_Total\_I\_E;

**proc** **export** data=WORK.MPP\_Delivery\_Total\_I\_E

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

/\* 2. Total\_I\_E adjustment for Antenatal phase\*/

**PROC** **SQL**;

create table MPP\_Antenatal\_Total\_I\_E as

select

a.\*

,&Total\_I\_E\_Adjustment. as Total\_I\_E\_Adjustment,

Prices\_Antenatal\_CB\*(**1**+&Total\_I\_E\_Adjustment) as Prices\_Antenatal\_Total\_I\_E

from work.MPP\_Antenatal\_CB as a

order by a.UC\_Antenatal

;

**quit**;

\*/ Added output of data ;

%let outputfile=MPP\_Antenatal\_Total\_I\_E;

**proc** **export** data=WORK.MPP\_Antenatal\_Total\_I\_E

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

/\* 3. Total\_I\_E adjustment for Postnatal phase\*/

**PROC** **SQL**;

create table MPP\_Postnatal\_Total\_I\_E as

select

a.\*,

&Total\_I\_E\_Adjustment. as Total\_I\_E\_Adjustment,

Prices\_Postnatal\_CB\*(**1**+&Total\_I\_E\_Adjustment) as Prices\_Postnatal\_Total\_I\_E

from work.MPP\_Postnatal\_CB as a

order by a.UC\_Postnatal

;

**quit**;

\*/ Added output of data ;

%let outputfile=MPP\_Postnatal\_Total\_I\_E;

**proc** **export** data=WORK.MPP\_Postnatal\_Total\_I\_E

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* Author: PID ;

\* Date: 05 June 2017 ;

\* Modified by: PID ;

\* Date: May 2018 ;

\* version: 1.02 ;

# \* MPP\_Step\_5\_4\_TotalCNST Adjustment\_Lev1.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* MPP\_TotalCNST adjustment for Delivery phase only;

/\*%macro MPP\_Total\_CNST(schema=);\*/

/\* Total\_CNST adjustment for Delivery phase\*/

/\* Retrive CNST\_Maternity factors from the master file\*/

/\* Note -- This calcualtion should be updated annually \*/

**Proc** **sql**;

create table MPP\_CNST\_Factor1 as

select

**1** as ID

,**1**+CNST as CNST1617

from Tariff\_R.cnst as a

where HRGSubchapter = 'Maternity\*' and Year ="16/17"

;

**quit**;

**Proc** **sql**;

create table MPP\_CNST\_Factor2 as

select

**1** as ID

,(**1**+CNST) as CNST1718

from Tariff\_R.cnst

where HRGSubchapter = 'Maternity\*' and Year ="17/18"

;

**quit**;

**Proc** **sql**;

create table MPP\_CNST\_Factor3 as

select

**1** as ID

,(**1**+CNST) as CNST1819

from Tariff\_R.cnst

where HRGSubchapter = 'Maternity\*' and Year ="18/19"

;

**quit**;

**Proc** **sql**;

create table MPP\_CNST\_Factor4 as

select

**1** as ID

,(**1**+CNST) as CNST1920

from Tariff\_R.cnst

where HRGSubchapter = 'Maternity\*' and Year ="19/20"

;

**quit**;

**data** MPP\_CNST\_factor;

merge MPP\_CNST\_Factor1 MPP\_CNST\_Factor2 MPP\_CNST\_Factor3 MPP\_CNST\_Factor4;

**run**;

/\* Adjusts prices by total\_CNST factor\*/

**PROC** **SQL**;

create table MPP\_Delivery\_Total\_CNST\_1 as

select

a.\*

from MPP\_Delivery\_Total\_I\_E as a

;

**quit**;

**PROC** **SQL**;

create table MPP\_Delivery\_Total\_CNST\_2 as

select

a.\*,

b.CNST1718\*b.CNST1819-**1** as Total\_CNST,

(calculated Total\_CNST+ Total\_I\_E\_Adjustment) as Uplift\_Total\_Adjust,

a.Prices\_Delivery\_CB\*(Calculated Uplift\_Total\_Adjust+**1**) as Prices\_Delivery\_Total\_CNST

from MPP\_Delivery\_Total\_CNST\_1 as a,

MPP\_CNST\_factor as b

where a.Level <> ('Excess Bed Days')

;

**quit**;

/\* CNST factor doesn't apply to EBD price\*/

**PROC** **SQL**;

create table MPP\_Delivery\_Total\_CNST\_3 as

select

a.\*,

**0** as Total\_CNST,

(calculated Total\_CNST+ Total\_I\_E\_Adjustment) as Uplift\_Total\_Adjust,

a.Prices\_Delivery\_CB\*(Calculated Uplift\_Total\_Adjust+**1**) as Prices\_Delivery\_Total\_CNST

from MPP\_Delivery\_Total\_CNST\_1 as a,

MPP\_CNST\_factor as b

where a.Level = ('Excess Bed Days')

;

**quit**;

/\* join two tables above together\*/

**data** MPP\_Delivery\_Total\_CNST;

set MPP\_Delivery\_Total\_CNST\_2 MPP\_Delivery\_Total\_CNST\_3;

**run**;

**data** outsas.MPP\_Delivery\_Total\_CNST;

set MPP\_Delivery\_Total\_CNST;

**run**;

\*/ Added output of data ;

%let outputfile=MPP\_Delivery\_Total\_CNST;

**proc** **export** data=outsas.MPP\_Delivery\_Total\_CNST

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

/\*Antenatal phase\*/

**proc** **sql**;

create table MPP\_Antenatal\_Total\_CNST\_1

as

select a.\*

from MPP\_Antenatal\_Total\_I\_E as a

;

**quit**;

**PROC** **SQL**;

create table MPP\_Antenatal\_Total\_CNST as

select

a.\*,

**0** as Total\_CNST,

(calculated Total\_CNST+ Total\_I\_E\_Adjustment) as Uplift\_Total\_Adjust,

a.Prices\_Antenatal\_CB\*(Calculated Uplift\_Total\_Adjust+**1**) as Prices\_Antenatal\_Total\_CNST

from MPP\_Antenatal\_Total\_CNST\_1 as a,

MPP\_CNST\_factor as b

;

**quit**;

**data** outsas.MPP\_Antenatal\_Total\_CNST;

set MPP\_Antenatal\_Total\_CNST;

**run**;

\*/ Added output of data ;

%let outputfile=MPP\_Antenatal\_Total\_CNST;

**proc** **export** data=outsas.MPP\_Antenatal\_Total\_CNST

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

/\*Postnatal phase\*/

**proc** **sql**;

create table MPP\_Postnatal\_Total\_CNST\_1

as

select a.\*

from MPP\_Postnatal\_Total\_I\_E as a

;

**quit**;

**PROC** **SQL**;

create table MPP\_Postnatal\_Total\_CNST as

select

a.\*,

**0** as Total\_CNST,

(calculated Total\_CNST+ Total\_I\_E\_Adjustment) as Uplift\_Total\_Adjust,

a.Prices\_Postnatal\_CB\*(Calculated Uplift\_Total\_Adjust+**1**) as Prices\_Postnatal\_Total\_CNST

from MPP\_Postnatal\_Total\_CNST\_1 as a,

MPP\_CNST\_factor as b

;

**quit**;

**data** outsas.MPP\_Postnatal\_Total\_CNST;

set MPP\_Postnatal\_Total\_CNST;

**run**;

\*/ Added output of data ;

%let outputfile=MPP\_Postnatal\_Total\_CNST;

**proc** **export** data=outsas.MPP\_Postnatal\_Total\_CNST

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

/\*\*/

/\*\*/

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 05 June 2017 ;

\* Modified by: PID ;

\* Date: May 2018

\* version: 1.02 ;

\* MPP\_Step\_5\_5\_QR2 Adjustment\_Lev1NV.sas :

\* ;

\* PID CREATE CODES FOR AUTO-CALCULATION OF QR2 ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

LIBNAME MAMPP base "\\irnarch\sas\_data\Tariff Rebuild\Model outputs pre\_MA\MPP\FY1920\_Final\_V3\Version8\6\_Levels\data";

/\* ----------------------------------------------------------- QR2 Calculation ------------ Start from here\*/

/\*------------------------below is the process of calculation QR2 factors for three phases------01/03/2018 \*/

/\* Step 1: export bef\_MA prices and activity to Excel files \*/

/\*Note: Amend codes to add activity into the input tables for antenatal and postnatal phases\*/

/\*Step 1 ---------------- Create excel file before Manual Adjustment ------------------------- Start from here \*/

/\*%let outputfile=MPP\_BefMA\_Prices;\*/

/\*export Prices for EWG consultations\*/

**proc** **sql**;

create table work.BefMA\_prices\_Antenatal

as

select

ID,

Level,

Prices\_Antenatal\_Total\_CNST

from work.MPP\_ANTENATAL\_TOTAL\_CNST

;

**quit**;

/\*Postnatal\*/

**proc** **sql**;

create table work.BefMA\_prices\_Postnatal

as

select

ID,

Level,

Prices\_Postnatal\_Total\_CNST

from work.MPP\_POSTNATAL\_TOTAL\_CNST

;

**quit**;

/\*Delivery\*/

**proc** **sql**;

create table work.BefMA\_prices\_Delivery

as

select

Level,

Total\_Activity,

Prices\_Delivery\_Total\_CNST

from work.MPP\_DELIVERY\_TOTAL\_CNST

;

**quit**;

/\*Step 1 ---------------- Create excel file before Manual Adjustment ------------------------- End here \*/

/\* ----------------- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* -------------------\*/

/\*Part 2: Implement Cash\_In\_Out and Caclulate CashInOutFactor ---Antenatal------------------------ Start from here \*/

/\* ----------------- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* -------------------\*/

/\*Cash\_in\_Cash\_Out table \*/

**data** MPP\_CashInOut\_Table\_1(drop=i);

Model="Maternity";

do i=**1** to **3**;

order =i;

output;

end;

**run**;

**data** MPP\_CashInOut\_Table;

length ID $20.;

format ID $20.;

retain Model;

set MPP\_CashInOut\_Table\_1;

if order=**1** then

do;

ID="Antenatal";

CashIO\_Amount=&SFM\_Quantum.;

end;

if order=**2** then

do;

ID="Postnatal";

CashIO\_Amount=&Postnatal\_Quantum.;

end;

if order=**3** then

do ;

ID="Delivery";

CashIO\_Amount=&DELI\_Quantum.;

end;

**run**;

/\*Part 2.1 ---Cash-in\_Cash\_out---------Antenatal ----------------- Start from Here\*/

**proc** **sql**;

select CashIO\_Amount into :Cashinout\_Antenatal

from MPP\_CashInOut\_Table

where ID="Antenatal"

;

**quit**;

**proc** **sql**;

create table MPP\_CashIO\_Antenatal\_1

as

select

"Antenatal" as Phase ,

a.Level,

a.Act\_assumed as Activities,

b.Prices\_Antenatal\_Total\_CNST as Prices\_Antenatal\_PreMA,

a.Act\_assumed\*b.Prices\_Antenatal\_Total\_CNST as BefMA\_Quantum\_Antenatal

from Aggmpp.act\_antenatal as a

left join work.MPP\_ANTENATAL\_TOTAL\_CNST as b

on a.level=b.level

;

**quit**;

/\*Apply Cash\_In\_Out \*/

**proc** **sql**;

create table MPP\_CashIO\_Antenatal\_2

as

select sum(BefMA\_Quantum\_Antenatal) as BefMA\_Antenatal\_TotQuan,

calculated BefMA\_Antenatal\_TotQuan-&Cashinout\_Antenatal. as Cashinout\_Antenatal\_TotQuan,

calculated Cashinout\_Antenatal\_TotQuan/calculated BefMA\_Antenatal\_TotQuan -**1** as MPP\_Antenatal\_CashInOut\_Factor

from MPP\_CashIO\_Antenatal\_1

;

**quit**;

**proc** **sql**;

select MPP\_Antenatal\_CashInOut\_Factor into :Antenatal\_CashInOut\_Factor

from MPP\_CashIO\_Antenatal\_2;

**quit**;

**proc** **sql**;

create table MPP\_Antenatal\_CashInOut

as

select a.\*,

&Antenatal\_CashInOut\_Factor. as CashInOut\_Antenatal\_Factor,

Prices\_Antenatal\_Total\_CNST\*(**1**+&Antenatal\_CashInOut\_Factor.) as Prices\_Antenatal\_CashInOut

from outsas.MPP\_Antenatal\_Total\_CNST as a

;

**quit**;

/\*Part 2.1 ---Cash\_in\_Cash\_Out---------Antenatal ----------------- End Here\*/

/\*Step 3: Calculate QR2

price PostMA = Price\_CashinOut

\*/

**proc** **sql**;

create table MPP\_QR2\_Antenatal\_1

as

select

"Antenatal" as Phase ,

a.Level,

a.Act\_assumed as Activities,

b.Prices\_Antenatal\_CashInOut,

a.Act\_assumed\*b.Prices\_Antenatal\_CashInOut as CashInOut\_Quantum\_Antenatal,

b.Prices\_Antenatal\_CashInOut as Prices\_Antenatal\_PostMA,

a.Act\_assumed\*b.Prices\_Antenatal\_CashInOut as AftMA\_Quantum\_Antenatal

from Aggmpp.act\_antenatal as a

left join work.MPP\_Antenatal\_CashInOut as b

on a.level=b.level

left join MAMPP.PostMA\_Price\_Antenantal as c

on a.level=c.level

;

**quit**;

**proc** **sql**;

create table MPP\_QR2\_Antenatal\_2

as

select

Phase,

sum(Activities) as Activities,

sum(CashInOut\_Quantum\_Antenatal) as CashInOut\_Quantum,

sum(AftMA\_Quantum\_Antenatal) as AftMA\_Quantum,

(calculated CashInOut\_Quantum/calculated AftMA\_Quantum)-**1** as MPP\_QR2

from work.MPP\_QR2\_Antenatal\_1

group by Phase

;

**quit**;

**proc** **sql**;

select MPP\_QR2 into :MPP\_QR2\_Antenatal

from MPP\_QR2\_Antenatal\_2

;

**quit**;

/\*Part 2.2 ---QR2---------Antenatal ----------------- Start from Here\*/

/\* 2.2. QR2 adjustment for Antenatal phase

Prices\_Antenatal\_CashInOut=Prices\_Antenatal\_PostMA

\*/

**PROC** **SQL**;

create table MPP\_Antenatal\_PostMA as

select

a.\*,

a.Prices\_Antenatal\_CashInOut as Prices\_Antenatal\_PostMA

from work.MPP\_ANTENATAL\_CASHINOUT as a

left join MAMPP.POSTMA\_PRICE\_ANTENANTAL as b

on a.Level=b.Level

;

**quit**;

**PROC** **SQL**;

create table MPP\_Antenatal\_QR2 as

select

a.\*,

&MPP\_QR2\_Antenatal. as QR2\_Adjustment,

Prices\_Antenatal\_PostMA \* (**1** + &MPP\_QR2\_Antenatal. ) as Prices\_Antenatal\_QR2

from MPP\_Antenatal\_PostMA as a

;

**quit**;

**data** OutSAS.MPP\_Antenatal\_QR2;

set MPP\_Antenatal\_QR2;

**run**;

\*/ Added output of data ;

%let outputfile=MPP\_Antenatal\_QR2;

**proc** **export** data=MPP\_Antenatal\_QR2

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

/\*Part 2.2 ---QR2---------Antenatal ----------------- End Here\*/

/\* ----------------- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* -------------------\*/

/\*Part 2: Implement Cash\_In\_Out and Caclulate CashInOutFactor ---Antenatal------------------------ End here \*/

/\* ----------------- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* -------------------\*/

/\* ----------------- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* -------------------\*/

/\*Part3--- Cash\_in\_out + QR2 ------------------ Postnatal--------------------------- Start from here \*/

/\* ----------------- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* -------------------\*/

/\*Part 3.1 - Cash in out-----------Postnatal----------------- Start from Here\*/

**proc** **sql**;

select CashIO\_Amount into :Cashinout\_Postnatal

from MPP\_CashInOut\_Table

where ID="Postnatal"

;

**quit**;

**proc** **sql**;

create table MPP\_CashIO\_Postnatal\_1

as

select

"Postnatal" as Phase ,

a.Level,

a.Act\_assumed as Activities,

b.Prices\_Postnatal\_Total\_CNST as Prices\_Postnatal\_PreMA,

a.Act\_assumed\*b.Prices\_Postnatal\_Total\_CNST as BefMA\_Quantum\_Postnatal

from Aggmpp.act\_Postnatal as a

left join work.MPP\_Postnatal\_TOTAL\_CNST as b

on a.level=b.level

;

**quit**;

**proc** **sql**;

create table MPP\_CashIO\_Postnatal\_2

as

select sum(BefMA\_Quantum\_Postnatal) as BefMA\_Postnatal\_TotQuan,

calculated BefMA\_Postnatal\_TotQuan-&Cashinout\_Postnatal. as Cashinout\_Postnatal\_TotQuan,

calculated Cashinout\_Postnatal\_TotQuan/calculated BefMA\_Postnatal\_TotQuan -**1** as MPP\_Postnatal\_CashInOut\_Factor

from MPP\_CashIO\_Postnatal\_1

;

**quit**;

**proc** **sql**;

select MPP\_Postnatal\_CashInOut\_Factor into :Postnatal\_CashInOut\_Factor

from MPP\_CashIO\_Postnatal\_2;

**quit**;

**proc** **sql**;

create table MPP\_Postnatal\_CashInOut

as

select a.\*,

&Postnatal\_CashInOut\_Factor. as CashInOut\_Postnatal\_Factor,

Prices\_Postnatal\_Total\_CNST\*(**1**+&Postnatal\_CashInOut\_Factor.) as Prices\_Postnatal\_CashInOut

from outsas.MPP\_Postnatal\_Total\_CNST as a

;

**quit**;

/\*Part 3.1 -----Cash in out-------Postnatal----------------- End Here\*/

/\*Part 3.2-----QR2------- Postnatal -----------------Start from here\*/

/\* import PostMA\_Price table \*/

/\*proc import out=MAMPP.PostMA\_Price\_Postnantal\*/

/\* datafile= "\\irnarch\sas\_data\Tariff Rebuild\Model outputs pre\_MA\MPP\Y1920\_2\MPP\_PostMA\_Prices.xlsx"\*/

/\* dbms=xlsx replace;\*/

/\* sheet="Postnatal";\*/

/\* getnames=yes;\*/

/\*run;\*/

/\*2) Calculate QR2

price PostMA = Price\_CashinOut

\*/

**proc** **sql**;

create table MPP\_QR2\_Postnatal\_1

as

select

"Postnatal" as Phase ,

a.Level,

a.Act\_assumed as Activities,

b.Prices\_Postnatal\_CashInOut ,

a.Act\_assumed\*b.Prices\_Postnatal\_CashInOut as CashInOut\_Quantum\_Postnatal,

/\*c.PostMAPrices\_Postnatal as Prices\_Postnatal\_PostMA,\*/

b.Prices\_Postnatal\_CashInOut as Prices\_Postnatal\_PostMA,

a.Act\_assumed\*b.Prices\_Postnatal\_CashInOut as AftMA\_Quantum\_Postnatal

from Aggmpp.act\_postnatal as a

left join work.MPP\_POSTNATAL\_CashInOut as b

on a.level=b.level

left join MAMPP.PostMA\_Price\_Postnatal as c

on a.level=c.level

;

**quit**;

**proc** **sql**;

create table MPP\_QR2\_Postnatal\_2

as

select

Phase,

sum(Activities) as Activities,

sum(CashInOut\_Quantum\_Postnatal) as CashInOut\_Quantum,

sum(AftMA\_Quantum\_Postnatal) as AftMA\_Quantum,

calculated CashInOut\_Quantum/calculated AftMA\_Quantum-**1** as MPP\_QR2

from work.MPP\_QR2\_Postnatal\_1

group by Phase

;

**quit**;

**proc** **sql**;

select MPP\_QR2 into :MPP\_QR2\_Postnatal

from MPP\_QR2\_Postnatal\_2

;

**quit**;

/\*Part 3.2-----Cash\_in\_Out------- Postnatal -----------------Start from here\*/

/\* 3. QR2 adjustment for Postnatal phase

Prices\_Postnatal\_PostMA=Prices\_Postnatal\_CashInOut

\*/

**PROC** **SQL**;

create table MPP\_Postnatal\_PostMA as

select

a.\*,

a.Prices\_Postnatal\_CashInOut as Prices\_Postnatal\_PostMA

from work.MPP\_POSTNATAL\_CASHINOUT as a

left join MAMPP.POSTMA\_PRICE\_Postnatal as b

on a.Level=b.Level

;

**quit**;

**PROC** **SQL**;

create table MPP\_Postnatal\_QR2 as

select

a.\*,

&MPP\_QR2\_Postnatal. as QR2\_Adjustment,

Prices\_Postnatal\_PostMA \* (**1** + &MPP\_QR2\_Postnatal.) as Prices\_Postnatal\_QR2

from work.MPP\_Postnatal\_PostMA as a

;

**quit**;

**data** OutSAS.MPP\_Postnatal\_QR2;

set MPP\_Postnatal\_QR2;

**run**;

\*/ Added output of data ;

%let outputfile=MPP\_Postnatal\_QR2;

**proc** **export** data=WORK.MPP\_Postnatal\_QR2

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

/\*3.2 Caclulate QR2 ------Postnatal---------------------- End here\*/

/\* ----------------- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* -------------------\*/

/\*Part3--- Cash\_in\_out + QR2 ------------------ Postnatal----------------------------------- End here \*/

/\* ----------------- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* -------------------\*/

/\* ----------------- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* -------------------\*/

/\*Part4 --- QR2 + Cash in out ---------------- Delivery ---------- Start from here \*/

/\* ----------------- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* -------------------\*/

/\*Part 4.1-----Calculate QR2-------- Delivery -----------------------Start from here\*/

**proc** **sql**;

create table MPP\_QR2\_Delivery\_1

as

select

"Delivery" as Phase ,

a.Level,

a.Total\_Activity as Activities,

a.Prices\_Delivery\_Total\_CNST as BefMAPrice\_Delivery,

a.Total\_Activity\*a.Prices\_Delivery\_Total\_CNST as BefMA\_Quantum\_Delivery,

b.PostMAPrices\_Delivery,

a.Total\_Activity\*b.PostMAPrices\_Delivery as AftMA\_Quantum\_Delivery

from MPP\_Delivery\_TOTAL\_CNST as a

left join MAMPP.PostMA\_Price\_Delivery as b

on a.Level=b.level

Where a.Level ^="Excess Bed Days"

;

**quit**;

**proc** **sql**;

create table MPP\_QR2\_Delivery\_2

as

select

Phase,

sum(Activities) as Activities,

sum(BefMA\_Quantum\_Delivery) as BefMA\_Quantum,

sum(AftMA\_Quantum\_Delivery) as AftMA\_Quantum,

calculated BefMA\_Quantum/calculated AftMA\_Quantum-**1** as MPP\_QR2

from work.MPP\_QR2\_Delivery\_1

group by Phase

;

**quit**

;

**proc** **sql**;

select MPP\_QR2 into :MPP\_QR2\_Delivery

from MPP\_QR2\_Delivery\_2

;

**quit**;

/\*EBD\*/

**proc** **sql**;

create table MPP\_QR2\_EBD

as

select

"Excess Bed Days" as Phase ,

a.Total\_Activity as Activities,

**.** as BefMA\_Quantum,

**.** as AftMA\_Quantum,

**0** as MPP\_QR2

from MPP\_DELIVERY\_TOTAL\_CNST as a

Where a.Level ="Excess Bed Days"

;

**quit**;

**proc** **sql**;

select MPP\_QR2 into :MPP\_QR2\_EBD

from MPP\_QR2\_EBD

;

**quit**;

**data** MPP\_Delivery\_Total\_CNST;

set MPP\_Delivery\_Total\_CNST;

order=\_N\_;

**run**;

**PROC** **SQL**;

create table MPP\_Delivery\_PostMA as

select

a.\*,

b.PostMAPrices\_Delivery as Prices\_Delivery\_PostMA

from work.MPP\_Delivery\_Total\_CNST as a

left join MAMPP.POSTMA\_PRICE\_Delivery as b

on a.level=b.level

order by a.order

;

**quit**;

**PROC** **SQL**;

create table MPP\_Delivery\_QR2 as

select distinct

a.\*

,&MPP\_QR2\_Delivery. as QR2\_Adjustment

,Prices\_Delivery\_PostMA \* (**1** + &MPP\_QR2\_Delivery. ) as Prices\_Delivery\_QR2

from MPP\_Delivery\_PostMA as a

Where a.level ^="Excess Bed Days"

order by order

;

**quit**;

**PROC** **SQL**;

create table MPP\_EBD\_QR2 as

select distinct

a.\*

,&MPP\_QR2\_EBD. as QR2\_Adjustment

,Prices\_Delivery\_PostMA \* (**1** + &MPP\_QR2\_EBD.) as Prices\_Delivery\_QR2

from MPP\_Delivery\_PostMA as a

Where a.level ="Excess Bed Days"

order by order

;

**quit**;

**data** MPP\_Delivery\_QR2(drop=order);

set MPP\_Delivery\_QR2

MPP\_EBD\_QR2;

**run**;

**data** OutSAS.MPP\_Delivery\_QR2;

set MPP\_Delivery\_QR2;

**run**;

\*/ Added output of data ;

%let outputfile=MPP\_Delivery\_QR2;

**proc** **export** data=WORK.MPP\_Delivery\_QR2

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

/\*Part 4.1-----Calculate QR2-------- Delivery -----------------------Start from here\*/

/\*Part 4.2-----Calculate Cash in Out -------- Delivery -----------------------Start from here\*/

**proc** **sql**;

select CashIO\_Amount into :Cashinout\_Delivery

from MPP\_CashInOut\_Table

where ID="Delivery"

;

**quit**;

/\*4.2.1 - Cash in Out for first 30 HRGs -- Start from here \*/

**proc** **sql**;

create table MPP\_CashIO\_Delivery\_1

as

select

"Delivery" as Phase ,

a.level,

a.Total\_Activity ,

a.Prices\_Delivery\_QR2,

a.Total\_Activity\*a.Prices\_Delivery\_QR2 as QR2\_Quantum\_Delivery,

a.Prices\_Delivery\_QR2 as Prices\_Delivery\_CashInOut,

a.Total\_Activity\*a.Prices\_Delivery\_QR2 as CashIO\_Quantum\_Delivery

from MPP\_Delivery\_QR2 as a

where level not in ("Excess Bed Days")

;

**quit**;

**proc** **sql**;

select sum(QR2\_Quantum\_Delivery) into :QR2\_Delivery\_TotQuan\_lvls456

from MPP\_CashIO\_Delivery\_1

where find(level,"4","i") or find(level,"5","i") or find(level,"6","i");

**quit**;

**data** a1;

set MPP\_CashIO\_Delivery\_1;

if find(level, "4","i") then Delivery\_CashIO\_Factor\_ratio=**1**-&Cashinout\_Delivery./&QR2\_Delivery\_TotQuan\_lvls456.;

else if find(level, "5","i") then Delivery\_CashIO\_Factor\_ratio=**1**-&Cashinout\_Delivery./&QR2\_Delivery\_TotQuan\_lvls456.;

else if find(level, "6","i") then Delivery\_CashIO\_Factor\_ratio=**1**-&Cashinout\_Delivery./&QR2\_Delivery\_TotQuan\_lvls456.;

else Delivery\_CashIO\_Factor\_ratio=**1**;

**run**;

**proc** **sql**;

create table MPP\_CashIO\_Delivery\_2

as

select

\*,

Delivery\_CashIO\_Factor\_ratio -**1** as Delivery\_CashIO\_Factor

from a1

;

**quit**;

/\*Final Cash\_IO Table 2 -- First 30 HRGS \*/

**proc** **sql**;

create table MPP\_Delivery\_CashInOut\_1

as

select a.\*,

b.Delivery\_CashIO\_Factor as CashInOut\_Delivery\_Factor,

a.Prices\_Delivery\_QR2\*(**1**+b.Delivery\_CashIO\_Factor) as Prices\_Delivery\_CashInOut

from MPP\_Delivery\_QR2 as a

left join MPP\_CashIO\_Delivery\_2 as b

on a.level=b.level

where a.level not in ("Excess Bed Days")

;

**quit**;

/\*4.2.1 - Cash in Out for first 30 HRGs -- End here \*/

/\*4.2.3 - Cash in Out for EBD -- Start from here \*/

**proc** **sql**;

create table MPP\_CashIO\_Delivery\_EBD\_1

as

select

"Delivery" as Phase ,

a.level,

a.Total\_Activity ,

a.Prices\_Delivery\_QR2,

a.Total\_Activity\*a.Prices\_Delivery\_QR2 as EBD\_QR2\_Quantum\_Delivery,

a.Prices\_Delivery\_QR2 as Prices\_Delivery\_CashInOut,

a.Total\_Activity\*a.Prices\_Delivery\_QR2 as EBDCashIO\_Quantum\_Delivery

from MPP\_Delivery\_QR2 as a

where level in ("Excess Bed Days")

;

**quit**;

**proc** **sql**;

create table MPP\_CashIO\_Delivery\_EBD\_2

as

select

sum(EBD\_QR2\_Quantum\_Delivery) as EBD\_QR2\_Delivery\_TotQuan,

sum(EBDCashIO\_Quantum\_Delivery) as EBD\_CashIO\_Delivery\_TotQuan,

calculated EBD\_QR2\_Delivery\_TotQuan /Calculated EBD\_CashIO\_Delivery\_TotQuan -**1** as EBD\_Delivery\_CashIO\_Factor

from MPP\_CashIO\_Delivery\_EBD\_1

;

**quit**;

**proc** **sql**;

select EBD\_Delivery\_CashIO\_Factor into :EBD\_Delivery\_CashIO\_Factor

from MPP\_CashIO\_Delivery\_EBD\_2

;

**quit**;

/\*Final Cash\_IO Table 3 -- EBD \*/

**proc** **sql**;

create table MPP\_Delivery\_CashInOut\_EBD

as

select a.\*,

&EBD\_Delivery\_CashIO\_Factor. as CashInOut\_Delivery\_Factor,

Prices\_Delivery\_QR2\*(**1**+&EBD\_Delivery\_CashIO\_Factor.) as Prices\_Delivery\_CashInOut

from MPP\_Delivery\_QR2 as a

where level in ("Excess Bed Days")

;

**quit**;

/\*4.2.3 - Cash in Out for EBD -- End here \*/

/\*Final Cash\_In\_Cash\_Out table -- Delivery \*/

**data** MPP\_Delivery\_CashInOut;

set

MPP\_Delivery\_CashInOut\_1

MPP\_Delivery\_CashInOut\_EBD

;

**run**;

**data** OutSAS.MPP\_Delivery\_CashInOut;

set MPP\_Delivery\_CashInOut;

**run**;

\*/ Added output of data ;

%let outputfile=MPP\_Delivery\_CashInOut;

**proc** **export** data=WORK.MPP\_Delivery\_CashInOut

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

/\*Part 4.2-----Calculate Cash in Out -------- Delivery -----------------------End here\*/

/\* ----------------- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* -------------------\*/

/\*Part4 --- QR2 + Cash in out ---------------- Delivery ---------- ------ End here \*/

/\* ----------------- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* -------------------\*/

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 05 June 2017 ;

\* Modified by: PID ;

\* Date: May 2018 ;

\* version: 1.02 ;

# \* MPP\_Step\_5\_6\_SMF Adjustment\_Lev1.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* MPP\_SMF;

\* Apply SCF to Prices;

/\*-------------------------------------Note ----------------------------------------- YC Oct 2017 ------- \*/

/\*Hard codeed year=17/18 change to &FYear. for real data \*/

/\*-------------------------------------Note ----------------------------------------- YC Oct 2017 ------- \*/

/\*%macro MPPSMF(schema=);\*/

/\*Get Smoothing Factor from Tariff\_R \*/

/\*Re run Maternity Model update Tariff\_R.MATERNITY\_SMOOTHING table code in the project to update Smoothing Factor

location: T:\Tariff Rebuild\Models for Testing\TC\_Model\_Setup\TC\_SmoothingFactor\_update

\*/

/\* 1.SMF adjustment for Delivery phase\*/

**PROC** **SQL**;

create table MPP\_Delivery\_SMF as

select

a.\*

,b.SMF

,a.Prices\_Delivery\_CashInOut\*(**1**+b.SMF) as Prices\_Delivery\_SMF

from work.MPP\_Delivery\_CashInOut as a,

Tariff\_R.maternity\_smoothing as b

where b.Year="&FYear."

;

**quit**;

**data** OutSAS.MPP\_Delivery\_SMF;

set MPP\_Delivery\_SMF;

**run**;

/\* Added output of data ;\*/

%let outputfile=MPP\_Delivery\_SMF;

**proc** **export** data=WORK.MPP\_Delivery\_SMF

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

/\* 2. SMF adjustment for Antenatal phase\*/

**PROC** **SQL**;

create table MPP\_Antenatal\_SMF as

select

a.\*

,b.SMF

,a.Prices\_Antenatal\_QR2\*(**1**+b.SMF) as Prices\_Antenatal\_SMF

from work.MPP\_Antenatal\_QR2 as a,

Tariff\_R.maternity\_smoothing as b

where b.Year="&FYear."

order by a.UC\_Antenatal

;

**quit**;

**data** OutSAS.MPP\_Antenatal\_SMF;

set MPP\_Antenatal\_SMF;

**run**;

\*/ Added output of data ;

%let outputfile=MPP\_Antenatal\_SMF;

**proc** **export** data=MPP\_Antenatal\_SMF

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

/\* 3. SMF adjustment for Postnatal phase\*/

**PROC** **SQL**;

create table MPP\_Postnatal\_SMF as

select

a.\*

,b.SMF

,a.Prices\_Postnatal\_QR2\*(**1**+b.SMF) as Prices\_Postnatal\_SMF

from work.MPP\_Postnatal\_QR2 as a,

Tariff\_R.maternity\_smoothing as b

where b.Year="&FYear."

order by a.UC\_Postnatal

;

**quit**;

**data** OutSAS.MPP\_Postnatal\_SMF;

set MPP\_Postnatal\_SMF;

**run**;

\*/ Added output of data ;

%let outputfile=MPP\_Postnatal\_SMF;

**proc** **export** data=MPP\_Postnatal\_SMF

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

/\* Scaling adjustment for Delivery phase\*/

/\* Retrive Scaling factors from the master file\*/

**proc** **sql** noprint;

select SF into :SCF

from Tariff\_R.SF

where year="&FYear"

;

**quit**;

%let SCF=&SCF;

%put SCF=&SCF;

**proc** **sql**;

select final\_cashio\_factor into :cashio\_factor\_NZ

from cash\_in\_out\_ia\_factors

where POD\_Subchapter="Maternity\_NZ";

**QUIT**;

/\* Adjusts prices by total\_CNST factor\*/

**PROC** **SQL**;

create table MPP\_Delivery\_Scaling as

select

a.\*,

&SCF. as Scaling\_Factor,

a.Prices\_Delivery\_SMF\*(**1**+&SCF.) as Prices\_Delivery\_ScalingFact

from MPP\_Delivery\_SMF as a

;

**quit**;

/\*&cashio\_factor\_NZ.\*/

**DATA** MPP\_Delivery\_Scaling;

SET MPP\_Delivery\_Scaling;

Prices\_Delivery\_ScalingFact=Prices\_Delivery\_ScalingFact\*(**1**+&cashio\_factor\_NZ.);

**RUN**;

**data** outsas.MPP\_Delivery\_Scaling;

set MPP\_Delivery\_Scaling;

**run**;

\*/ Added output of data ;

%let outputfile=MPP\_Delivery\_Scaling;

**proc** **export** data=WORK.MPP\_Delivery\_Scaling

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

/\*Antenatal phase\*/

**PROC** **SQL**;

create table MPP\_Antenatal\_Scaling as

select

a.\*,

&SCF. as Scaling\_Factor,

a.Prices\_Antenatal\_SMF\*(**1**+&SCF.) as Prices\_Antenatal\_ScalingFact

from MPP\_Antenatal\_SMF as a

;

**quit**;

/\*&cashio\_factor\_NZ.\*/

**DATA** MPP\_Antenatal\_Scaling;

SET MPP\_Antenatal\_Scaling;

Prices\_Antenatal\_ScalingFact=Prices\_Antenatal\_ScalingFact\*(**1**+&cashio\_factor\_NZ.);

**RUN**;

**data** outsas.MPP\_Antenatal\_Scaling;

set MPP\_Antenatal\_Scaling;

**run**;

\*/ Added output of data ;

%let outputfile=MPP\_Antenatal\_Scaling;

**proc** **export** data=outsas.MPP\_Antenatal\_Scaling

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

/\*Postnatal phase\*/

**PROC** **SQL**;

create table MPP\_Postnatal\_Scaling as

select

a.\*,

&SCF. as Scaling\_Factor,

a.Prices\_Postnatal\_SMF\*(**1**+&SCF.) as Prices\_Postnatal\_ScalingFact

from MPP\_Postnatal\_SMF as a

;

**quit**;

/\*&cashio\_factor\_NZ.\*/

**DATA** MPP\_Postnatal\_Scaling;

SET MPP\_Postnatal\_Scaling;

Prices\_Postnatal\_ScalingFact=Prices\_Postnatal\_ScalingFact\*(**1**+&cashio\_factor\_NZ.);

**RUN**;

**data** outsas.MPP\_Postnatal\_Scaling;

set MPP\_Postnatal\_Scaling;

**run**;

\*/ Added output of data ;

%let outputfile=MPP\_Postnatal\_Scaling;

**proc** **export** data=outsas.MPP\_Postnatal\_Scaling

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* Author: PID ;

\* date: 05 June 2017 ;

\* Modified by: PID ;

\* Date: May 2018 ;

\* version: 1.02 ;

# \* MPP\_Step\_5\_8\_CNST1920 Adjustment\_Lev1.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* MPP\_SCF;

\* Apply CNST1920 to Prices;

/\*CNST1920 adjustment doesn't apply to Antenatal and Postnatal phases\*/

/\* Note : -------------- will be changed when new CNST \*/

/\* 1.CNST adjustment for Delivery phase\*/

**PROC** **SQL**;

create table MPP\_Delivery\_CNST1920\_1 as

select

a.\*

,b.CNST1920-**1** as CNST1920

,a.Prices\_Delivery\_ScalingFact\*(**1**+(b.CNST1920-**1**)) as Prices\_Delivery\_CNST1920

from work.MPP\_Delivery\_Scaling as a,

work.MPP\_CNST\_FACTOR as b

where a.level ^=('Excess Bed Days')

;

**quit**;

/\* CNST doestn't apply to EBD prices\*/

**PROC** **SQL**;

create table MPP\_Delivery\_CNST1920\_2 as

select

a.\*

,**0** as CNST1920

,a.Prices\_Delivery\_ScalingFact as Prices\_Delivery\_CNST1920

from work.MPP\_Delivery\_Scaling as a,

work.MPP\_CNST\_factor as b

where a.level =('Excess Bed Days')

;

**quit**;

/\* add two tables above together\*/

**data** MPP\_Delivery\_CNST1920;

set MPP\_Delivery\_CNST1920\_1 MPP\_Delivery\_CNST1920\_2;

**run**;

**data** outsas.MPP\_Delivery\_CNST1920;

set MPP\_Delivery\_CNST1920;

**run**;

\*/ Added output of data ;

%let outputfile=MPP\_Delivery\_CNST1920;

**proc** **export** data=WORK.MPP\_Delivery\_CNST1920

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

/\*Antenatal phase\*/

**data** MPP\_Antenatal\_CNST1920;

set MPP\_Antenatal\_Scaling;

**run**;

**data** outsas.MPP\_Antenatal\_CNST1920;

set MPP\_Antenatal\_CNST1920;

**run**;

/\*Postnatal phase\*/

**data** MPP\_Postnatal\_CNST1920;

set MPP\_Postnatal\_Scaling;

**run**;

**data** outsas.MPP\_Postnatal\_CNST1920;

set MPP\_Postnatal\_CNST1920;

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 05 June 2017 ;

\* Modified by: PID ;

\* Date: May 2018 ;

\* version: 1.02 ;

# \* MPP\_Step\_5\_9\_Inflation\_Efficiency201920\_Lev1.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* MPP\_IandE1920;

\* Apply MPP\_IandE1920 to Prices;

/\*%macro MPP\_IandE1920(schema=);\*/

/\* Calculate total I\_E\_1920 factors\*/

**proc** **sql**;

create table I\_E\_1920 as

select

**1** as ID

,**1**+Inflation\_and\_Efficiency\_Combine as IE\_1

from Tariff\_R.i\_e

where Year="&FYear."

;

**quit**;

/\* 1.MPP\_IandE1920 adjustment for Delivery phase\*/

**PROC** **SQL**;

create table MPP\_Delivery\_IandE1920 as

select

a.\*

,b.IE\_1-**1** as I\_E\_1920Adjustment

,a.Prices\_Delivery\_CNST1920\*b.IE\_1 as Prices\_Delivery\_FinalPrices

from MPP\_Delivery\_CNST1920 as a,

work.I\_E\_1920 as b

;

**quit**;

**data** outsas.MPP\_Delivery\_IandE1920;

set MPP\_Delivery\_IandE1920;

**run**;

\*/ Added output of data ;

%let outputfile=MPP\_Delivery\_IandE1920;

**proc** **export** data=WORK.MPP\_Delivery\_IandE1920

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

/\* 2. MPP\_IandE1920 adjustment for Antenatal phase\*/

**PROC** **SQL**;

create table MPP\_Antenatal\_IandE1920 as

select

a.\*

,b.IE\_1-**1** as I\_E\_1920Adjustment

,a.Prices\_Antenatal\_ScalingFact\*b.IE\_1 as Prices\_Antenatal\_FinalPrices

from MPP\_Antenatal\_CNST1920 as a

left join work.I\_E\_1920 as b

on a.ID=b.ID

order by UC\_Antenatal

;

**quit**;

**data** outsas.MPP\_Antenatal\_IandE1920;

set MPP\_Antenatal\_IandE1920;

**run**;

\*/ Added output of data ;

%let outputfile=MPP\_Antenatal\_IandE1920;

**proc** **export** data=MPP\_Antenatal\_IandE1920

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

/\* 3. IandE1920 adjustment for Postnatal phase\*/

**PROC** **SQL**;

create table MPP\_Postnatal\_IandE1920 as

select

a.\*

,b.IE\_1-**1** as I\_E\_1920Adjustment

,a.Prices\_Postnatal\_ScalingFact\*b.IE\_1 as Prices\_Postnatal\_FinalPrices

from work.MPP\_Postnatal\_CNST1920 as a

left join work.I\_E\_1920 as b

on a.ID=b.ID

order by UC\_Postnatal

;

**quit**;

**data** outsas.MPP\_Postnatal\_IandE1920;

set MPP\_Postnatal\_IandE1920;

**run**;

\*/ Added output of data ;

%let outputfile=MPP\_Postnatal\_IandE1920;

**proc** **export** data=MPP\_Postnatal\_IandE1920

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 05 June 2017 ;

\* Modified by: PID ;

\* Date: May 2018 ;

\* version: 1.02 ;

# \* MPP\_Step\_6.2\_Calculation of Quantums\_Lev1.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

/\*Antenatal \*/

**proc** **sql**;

create table MPP\_Quan\_Antenatal\_FinalPrice

as

select

"Antenatal" as Phase ,

a.Level format=$20. length=**20**,

a.Act\_assumed as Activities,

b.UC\_Antenatal as Unit\_Cost ,

a.Act\_assumed\*b.UC\_Antenatal as Pre\_QR1\_Quantum,

a.Act\_assumed\*b.Prices\_Antenatal\_QR1 as Post\_QR1\_Quantum,

a.Act\_assumed\*b.Prices\_Antenatal\_Total\_I\_E as Post\_Uplift\_Quantum,

a.Act\_assumed\*b.Prices\_Antenatal\_Total\_CNST as Post\_CNST\_Quantum,

a.Act\_assumed\*b.Prices\_Antenatal\_CashInOut as CashInOut\_Quantum,

a.Act\_assumed\*b.Prices\_Antenatal\_PostMA as Post\_MA\_Quantum,

a.Act\_assumed\*b.Prices\_Antenatal\_QR2 as Post\_QR2\_Quantum,

a.Act\_assumed\*b.Prices\_Antenatal\_SMF as Post\_SMF\_Quantum,

a.Act\_assumed\*b.Prices\_Antenatal\_ScalingFact as Post\_SCF\_Quantum,

a.Act\_assumed\*b.Prices\_Antenatal\_FinalPrices as Final\_Quantum

from Aggmpp.act\_antenatal as a

left join MPP\_Antenatal\_IandE1920 as b

on a.level=b.level

;

**quit**;

/\*Postnatal \*/

**proc** **sql**;

create table MPP\_Quan\_Postnatal\_FinalPrice

as

select

"Postnatal" as Phase ,

a.Level format=$20. length=**20**,

a.Act\_assumed as Activities,

b.UC\_Postnatal as Unit\_Cost,

a.Act\_assumed\*b.UC\_Postnatal as Pre\_QR1\_Quantum,

a.Act\_assumed\*b.Prices\_Postnatal\_QR1 as Post\_QR1\_Quantum,

a.Act\_assumed\*b.Prices\_Postnatal\_Total\_I\_E as Post\_Uplift\_Quantum,

a.Act\_assumed\*b.Prices\_Postnatal\_Total\_CNST as Post\_CNST\_Quantum,

a.Act\_assumed\*b.Prices\_Postnatal\_PostMA as Post\_MA\_Quantum,

a.Act\_assumed\*b.Prices\_Postnatal\_CashInOut as CashInOut\_Quantum,

a.Act\_assumed\*b.Prices\_Postnatal\_QR2 as Post\_QR2\_Quantum,

a.Act\_assumed\*b.Prices\_Postnatal\_SMF as Post\_SMF\_Quantum,

a.Act\_assumed\*b.Prices\_Postnatal\_ScalingFact as Post\_SCF\_Quantum,

a.Act\_assumed\*b.Prices\_Postnatal\_FinalPrices as Final\_Quantum

from Aggmpp.act\_Postnatal as a

left join MPP\_POSTNATAL\_IandE1920 as b

on a.level=b.level

;

**quit**;

/\*Delivery \*/

**proc** **sql**;

create table MPP\_Quan\_Delivery\_FinalPrice

as

select

"Delivery" as Phase ,

a.Level format=$20. length=**20**,

a.Total\_Activity as Activities,

a.Unit\_Cost ,

a.Total\_Activity\*Unit\_Cost as Pre\_QR1\_Quantum,

a.Total\_Activity\*a.Prices\_Delivery\_QR1 as Post\_QR1\_Quantum,

a.Total\_Activity\*a.Prices\_Delivery\_Total\_I\_E as Post\_Uplift\_Quantum,

a.Total\_Activity\*a.Prices\_Delivery\_Total\_CNST as Post\_CNST\_Quantum,

a.Total\_Activity\*a.Prices\_Delivery\_PostMA as Post\_MA\_Quantum,

a.Total\_Activity\*a.Prices\_Delivery\_QR2 as Post\_QR2\_Quantum,

a.Total\_Activity\*a.Prices\_Delivery\_CashInOut as CashInOut\_Quantum,

a.Total\_Activity\*a.Prices\_Delivery\_SMF as Post\_SMF\_Quantum,

a.Total\_Activity\*a.Prices\_Delivery\_ScalingFact as Post\_SCF\_Quantum,

a.Total\_Activity\*a.Prices\_Delivery\_FinalPrices as Final\_Quantum

from MPP\_DELIVERY\_IANDE1920 as a

;

**quit**;

**data** MPP\_Quan\_Summary\_AntPost\_1;

set

MPP\_Quan\_Antenatal\_FinalPrice

MPP\_Quan\_Postnatal\_FinalPrice

;

**run**;

**proc** **sql** noprint;

create table MPP\_Quan\_Summary\_AntPost\_2

as

select

"Total" as Phase ,

"\_" as Level format=$20. length=**20**,

**.** as Activities,

**.** as Unit\_Cost,

sum(Pre\_QR1\_Quantum) as Pre\_QR1\_Quantum,

sum(Post\_QR1\_Quantum) as Post\_QR1\_Quantum,

sum(Post\_Uplift\_Quantum) as Post\_Uplift\_Quantum,

sum(Post\_CNST\_Quantum) as Post\_CNST\_Quantum,

sum(Post\_MA\_Quantum) as Post\_MA\_Quantum,

sum(Post\_QR2\_Quantum) as Post\_QR2\_Quantum,

Sum(CashInOut\_Quantum) as CashInOut\_Quantum,

sum(Post\_SMF\_Quantum) as Post\_SMF\_Quantum,

sum(Post\_SCF\_Quantum) as Post\_SCF\_Quantum,

sum(Final\_Quantum) as Final\_Quantum

from

MPP\_Quan\_Summary\_AntPost\_1

;

**quit**;

**data** MPP\_Quan\_Summary\_AntPost;

set

MPP\_Quan\_Summary\_AntPost\_1

MPP\_Quan\_Summary\_AntPost\_2

;

**run**;

**data** OutSAS.MPP\_Quan\_Summary\_AntPost6Levels;

set MPP\_Quan\_Summary\_AntPost;

**run**;

\*/ Added output of data ;

%let outputfile=MPP\_Quan\_Summary\_AntPost6Levels;

**proc** **export** data=OutSAS.MPP\_Quan\_Summary\_AntPost6Levels

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

**proc** **sql** noprint;

create table MPP\_Deliv\_Quan\_Sum\_FinalPrice\_1

as

select

"Total" as Phase ,

"\_" as Level format=$20. length=**20**,

**.** as Activities,

**.** as Unit\_Cost,

sum(Pre\_QR1\_Quantum) as Pre\_QR1\_Quantum,

sum(Post\_QR1\_Quantum) as Post\_QR1\_Quantum,

sum(Post\_Uplift\_Quantum) as Post\_Uplift\_Quantum,

sum(Post\_CNST\_Quantum) as Post\_CNST\_Quantum,

sum(Post\_MA\_Quantum) as Post\_MA\_Quantum,

sum(Post\_QR2\_Quantum) as Post\_QR2\_Quantum,

sum(CashInOut\_Quantum) as CashInOut\_Quantum,

sum(Post\_SMF\_Quantum) as Post\_SMF\_Quantum,

sum(Post\_SCF\_Quantum) as Post\_SCF\_Quantum,

sum(Final\_Quantum) as Final\_Quantum

from

MPP\_Quan\_Delivery\_FinalPrice

;

**quit**;

**data** MPP\_Deliv\_Quan\_Sum\_FinalPrice;

set

MPP\_Quan\_Delivery\_FinalPrice

MPP\_Deliv\_Quan\_Sum\_FinalPrice\_1

;

**run**;

**data** OutSAS.MPP\_Deliv\_QuanSum\_6Levels;

set MPP\_Deliv\_Quan\_Sum\_FinalPrice;

**run**;

\*/ Added output of data ;

%let outputfile=MPP\_Deliv\_QuanSum\_6Levels;

**proc** **export** data=OutSAS.MPP\_Deliv\_QuanSum\_6Levels

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

/\*--------------------------------------------------------------\*/

# \* Maternity\_Trim\_Point\_Calculation

/\* Version: 1.02 \*/

/\* Author: PID \*/

/\* Date: May 2018 \*/

/\* Calculates Trim Points for Maternity Model (6 levles) \*/

/\* Updated by PID. for Flag table from Flat file in \*/

/\* Tariff\_R library \*/

/\*--------------------------------------------------------------\*/

/\* Input: \*/

/\* (1)[DSN\_Pbr\_Repository\_New].[PreProcess].Stage\_HES\_KeyFields \*/

/\* \*/

/\* Output: \*/

/\*(1)MP\_TrimPointCal\_Final \*/

/\* \*/

/\* \*/

/\*--------------------------------------------------------------\*/

/\*Get MP HRG table for Trim Point Calculation \*/

**proc** **sql** noprint;

create table MP\_HRG\_DeliveryFlags

as

select

HRG,

Description,

Pathway,

Delivery\_Complications\_Flag

from ParMPP.MPP\_PAR\_CDTABLE as a

WHERE Pathway="Delivery"

order by HRG

;

**quit**;

/\*Create MP HRG list \*/

**proc** **sql** noprint;

create table T\_MP\_uniqu\_HRG

as

select distinct

'"' as leftquote,

HRG,

'"' as rightquote,

strip(calculated leftquote)||strip(HRG)||strip(calculated rightquote) as Q\_HRG

from

MP\_HRG\_DeliveryFlags

order by HRG

;

**quit**;

/\*MP HRGs: output unique HRG List \*/

**proc** **sql** noprint;

select Q\_HRG into :HRGlist\_MP separated by ','

from T\_MP\_uniqu\_HRG

;

**quit**;

%put HRGlist\_MP=&HRGlist\_MP.;

/\*MP HRGs: get data from SQL server for Selecteds MP HRGs \*/

**proc** **sql** noprint;

create table HES\_KeyFields\_Data\_MP

as

select

RC\_SpellHRG\_Rev as HRG,

ADM,

Run\_ID,

Revised\_Excluded,

spellflag,

PROCODETD,

Spell\_LoS\_2yr

FROM bprPrPP.Stage\_HES\_KeyFields

where Run\_ID = &Run\_ID.

and Revised\_Excluded = 'N' and spellflag = **1** and substr(PROCODETD,**1**,**1**) in ('5','R','T') and adm<>"UX"

and RC\_SpellHRG\_Rev in (&HRGlist\_MP.)

order by RC\_SpellHRG\_Rev, adm

;

**quit**;

/\*add Delivery\_Complications\_Flag \*/

**proc** **sql** noprint;

create table Data\_for\_Cal\_MP\_Trimpoint\_1

as

select

a.\*,

b.Delivery\_Complications\_Flag

from HES\_KeyFields\_Data\_MP as a

inner join MP\_HRG\_DeliveryFlags as b

on A.HRG=b.HRG

order by a.HRG, a.adm

;

**quit**;

/\*create Amtype\_Adm\_table \*/

**proc** **sql** noprint;

create table Admtype\_Adm\_Table

(Rn num format=**8.0**,

Adm\_type char(**12**),

Adm char(**12**)

);

INSERT INTO Admtype\_Adm\_Table

values( **1**,'DC/EL','DC')

values( **1**,'DC/EL','EL')

values( **2**,'NE','NE')

values( **3**,'DC/EL/NE','DC')

values( **3**,'DC/EL/NE','EL')

values( **3**,'DC/EL/NE','NE' )

values( **4**,'DC','DC')

values( **5**,'EL','EL')

;

**quit**;

/\*MP HRGs: create Full list of HRG\_HRG\_Admtype\_adm \*/

**proc** **sql** noprint;

create table T\_MP\_HRG\_AdmType\_Adm

as select distinct

a.HRG ,

b.adm\_type,

b.adm

from T\_MP\_uniqu\_HRG as a ,admtype\_adm\_Table as b

order by a.HRG, b.adm\_type

;

**quit**;

/\*add ADM\_Type to data for Trimpoint calculation \*/

**proc** **sql** noprint;

create table Data\_for\_Cal\_MP\_Trimpoint

as

select

a.\*,

b.adm\_Type

from

Data\_for\_Cal\_MP\_Trimpoint\_1 as a

full join

T\_MP\_HRG\_AdmType\_Adm as b

on a.HRG=b.HRG and a.adm=b.adm

where a.adm is not null

order by a.Delivery\_Complications\_Flag, b.adm\_Type,a.HRG

;

**quit**;

/\*Calculate the Trim Points \*/

**proc** **means** data=Data\_for\_Cal\_MP\_Trimpoint Pctldef=**4** missing MAxdec=**1**

N Mean MEDIAN nmiss P25 P50 P75 noprint;

by Delivery\_Complications\_Flag adm\_type;

output out=MP\_TrimPointStats N=N Mean=Mean Median=Median nmiss=NMiss P25=P25 P50=P50 P75=P75;

var Spell\_LoS\_2yr;

**run**;

**proc** **sql** noprint;

create table MP\_TrimPointCal\_1 as

select

&Run\_ID as Run\_ID,

Delivery\_Complications\_Flag,

adm\_type,

(\_FREQ\_-coalesce(nmiss,**0**)) as Valid,

NMiss as Missing label="nmiss",

coalesce(Mean,**0**) as Mean label="Mean" format=**8.2**,

coalesce(Median,**0**) as Median label="Median" format=**8.2**,

coalesce(P25,**0**) as P25 label="P25" format=**8.2**,

coalesce(P50,**0**) as P50 label="P50" format=**8.2**,

coalesce(P75,**0**) as P75 label="P75" format=**8.2**,

coalesce(P75 + ((P75 - P25) \* **1.5**),**0**) as Trimpoint\_Cal format=**8.1**,

case when round( Calculated Trimpoint\_Cal)=**0** then **1**

else round(Calculated Trimpoint\_Cal,**1**)

end as TrimPoint

from MP\_TrimPointStats

order by Delivery\_Complications\_Flag, adm\_type

;

**quit**;

/\*Apply minimum 5 days rule to trim point \*/

**data** MP\_TrimPointCal;

set MP\_TrimPointCal\_1;

if TrimPoint<**5** then TrimPoint=**5**;

**run**;

**proc** **sql** noprint;

create table MP\_TrimPointCal\_forTranspose

as

select

Run\_ID,

Delivery\_Complications\_Flag,

Adm\_type,

TrimPoint

from MP\_TrimPointCal

where Adm\_type in ("DC/EL","NE", "DC/EL/NE")

order by Delivery\_Complications\_Flag,Adm\_type

;

**quit**;

**PROC** **TRANSPOSE** DATA= MP\_TrimPointCal\_forTranspose

OUT=OutSAS.MP\_TrimPoint\_&lev.(drop=Source)

PREFIX=TrimPoint\_

NAME=Source

LABEL=Label

;

BY Run\_ID Delivery\_Complications\_Flag;

ID Adm\_type;

VAR TrimPoint;

**RUN**;

**QUIT**;

%let outputfile=OutSAS.MP\_TrimPoint\_&lev.;

%let outputName=MPP\_TrimPoint;

**proc** **export** data=&outputfile.

outfile="&outputpath.\&outputName..csv"

dbms=csv

replace;

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 05 June 2017 ;

\* Modified by: PID ;

\* Date: May 2018 ;

\* version: 1.02 ;

# \* MPP\_Step\_8\_Linked Sheet\_Lev1.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

/\* 1.Non-delivery phases\*/

/\*1.a Antenatal phase \*/

**proc** **sql**;

create table OutSAS.MPPlinkedShAntenatalFinalPrice

as

select

"NA" as code,

Level as Name,

round(Prices\_Antenatal\_FinalPrices,**1.0**) as Tariff format=comma8.

from OutSAS.MPP\_Antenatal\_IandE1920 as a

;

**quit**;

/\*1.b Postnatal phase \*/

**proc** **sql**;

create table OutSAS.MPPlinkedShPostnatalFinalPrice

as

select

"NA" as code,

Level as Name,

round(Prices\_Postnatal\_FinalPrices,**1.0**) as Tariff format=comma8.

from OutSAS.MPP\_POSTNATAL\_IANDE1920 as a

;

**quit**;

/\*3.Mandatory delivery admitted patient and outpatient procedure prices for maternity services \*/

**proc** **sql** noprint;

select Prices\_Delivery\_FinalPrices into :EDB\_Price

from OutSAS.MPP\_DELIVERY\_IANDE1920

where level="Excess Bed Days"

;

**quit**;

%let EDB\_Price=&EDB\_Price.;

**proc** **sql**;

create table MPP\_Par\_CDTable\_X

as

select a.\*,

put(a.Delivery\_Complications\_Flag,**8.**) as Delivery\_Complications\_Flag\_C

from ParMPP.MPP\_Par\_CDTable as a

where a.Year="&FYear."

;

**quit**;

**proc** **sql**;

create table OutSAS.MPPlinkedShDeliveryHRGFPrices

as

select distinct

a.Delivery\_Complications\_Flag as level,

round(b.Prices\_Delivery\_FinalPrices,**1.**) as Combined\_DC\_EL\_NE format=comma8.,

"-" as Day\_Case\_Tariff,

"-" as Ordinary\_Elective\_Tariff,

c.TrimPoint\_DC\_EL\_NE as Ordinary\_elective\_LS\_trimpoint,

round(b.Prices\_Delivery\_FinalPrices,**1.**) as Non\_elective\_tariff format=comma8.,

c.TrimPoint\_DC\_EL\_NE as Non\_elective\_LS\_trimpoint,

round(&EDB\_Price.,**1.**) as Long\_stay\_payment format=comma8.,

"No" as Reduced\_SSE\_tariff\_Applic,

"-" as Pct\_Reduced\_SSE\_tariff\_Applic,

"-" as Reduced\_SSE\_tariff

from MPP\_Par\_CDTable\_X as a

left join OutSAS.MPP\_DELIVERY\_IANDE1920 as b

on a.Delivery\_Complications\_Flag\_C=b.level

left join OutSAS.MP\_TrimPoint\_&lev. as c

on a.Delivery\_Complications\_Flag=c.Delivery\_Complications\_Flag

where a.Year="&FYear." and a.Pathway="Delivery"

order by a.Delivery\_Complications\_Flag

;

**quit**;

\*/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 05 May 2018 ;

\* Modified by: PID ;

\* ;

# \* MPP\_Step\_9 Tables\_Lev1.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

**proc** **sql**;

create table u9 as select a.\*, b.prices\_delivery\_cashinout

from MPP\_DELIVERY\_QR2 as a left join MPP\_Delivery\_CashInOut as b

on a.level=b.level;

**quit**;

**data** MPP\_DELIVERY\_QR2;

set u9(drop=prices\_delivery\_qr2);

rename prices\_delivery\_cashinout=prices\_delivery\_qr2;

**run**;

**proc** **sql**;

create table MPP\_Antenatal\_afterQR2

as

select

"Antenatal" as ID,

a.Level,

a.Prices\_Antenatal\_QR2 as Prices\_QR2

from MPP\_ANTENATAL\_QR2 as a

;

**quit**;

**proc** **sql**;

create table MPP\_Postnatal\_afterQR2

as

select

"Postnatal" as ID,

a.Level,

a.Prices\_Postnatal\_QR2 as Prices\_QR2

from MPP\_POSTNATAL\_QR2 as a

;

**quit**;

**data** MPP\_Pathway\_afterQR2;

set

MPP\_Antenatal\_afterQR2

MPP\_Postnatal\_afterQR2

;

**run**;

/\* 1.Delivery phase \*/

**proc** **sql**;

select Prices\_Delivery\_QR2 into :EDB\_Price\_QR2

from MPP\_DELIVERY\_QR2

where level="Excess Bed Days"

;

**quit**;

%let EDB\_Price\_QR2=&EDB\_Price\_QR2.;

%put EDB\_Price\_QR2=&EDB\_Price\_QR2.;

**proc** **sql**;

create table MPP\_Par\_CDTable\_X

as

select a.\*,

put(a.Delivery\_Complications\_Flag,**8.**) as Delivery\_Complications\_Flag\_C

from ParMPP.MPP\_Par\_CDTable as a

where a.Year="&FYear."

;

**quit**;

**proc** **sql**;

create table MPP\_linkedShQR2\_Delivery

as

select distinct

a.level,

**0** as Outpatient\_procedure\_tariff,

a.Prices\_Delivery\_QR2 as Combined\_DC\_EL\_NE

from MPP\_DELIVERY\_QR2 as a

left join MPP\_Par\_CDTable\_X as b

on a.level=b.Delivery\_Complications\_Flag\_C

where a.level^="Excess Bed Days"

order by a.level

;

**quit**;

/\* 2.Non-delivery phases\*/

/\*2.a Antenatal phase \*/

**proc** **sql**;

create table MPP\_linkedShQR2\_Antenatal

as

select

"Antenatal phase" as HRG\_Code length=**20** format=$20.,

a.level as HRG\_Name length=**150** format=$150.,

a.Prices\_Antenatal\_QR2 as Tariff,

**0** as Outpatient\_procedure\_tariff,

**.** as DC\_EL\_Tariff,

**0** as DC\_Tariff,

**0** as EL\_Tariff,

**.** as EL\_Trimpoint,

**.** as NE\_Tariff,

**.** as NE\_Trimpoint,

**.** as EBD\_payment,

" " as SSEM\_applicable,

" " as calculation\_SSEM,

**0** as SSEM\_tariff,

"Antenatal phase "||a.level as Lookup\_Combination

from MPP\_ANTENATAL\_QR2 as a

;

**quit**;

/\*2.b Postnatal phase \*/

**proc** **sql**;

create table MPP\_linkedShQR2\_Postnatal

as

select

"Postnatal phase " as HRG\_Code length=**20** format=$20.,

a.level as HRG\_Name length=**150** format=$150.,

a.Prices\_Postnatal\_QR2 as Tariff,

**0** as Outpatient\_procedure\_tariff,

**.** as DC\_EL\_Tariff,

**0** as DC\_Tariff,

**0** as EL\_Tariff,

**.** as EL\_Trimpoint,

**.** as NE\_Tariff,

**.** as NE\_Trimpoint,

**.** as EBD\_payment,

" " as SSEM\_applicable,

" " as calculation\_SSEM,

**0** as SSEM\_tariff,

"Postnatal phase"||a.level as Lookup\_Combination

from MPP\_POSTNATAL\_QR2 as a

;

**quit**;

/\*3.Mandatory delivery admitted patient and outpatient procedure prices for maternity services \*/

**proc** **sql**;

create table MPP\_linkedShQR2\_deli\_HRGPrices

as

select distinct

"Delivery Phase" as HRG\_Code,

a.Delivery\_Complications\_Flag\_C as HRG\_Name ,

**.** as Tariff,

**0** as Outpatient\_procedure\_tariff,

b.Combined\_DC\_EL\_NE as DC\_EL\_Tariff,

**0** as DC\_Tariff,

**0** as EL\_Tariff,

c.TrimPoint\_DC\_EL\_NE as EL\_Trimpoint,

b.Combined\_DC\_EL\_NE as NE\_Tariff,

c.TrimPoint\_DC\_EL\_NE as NE\_Trimpoint,

&EDB\_Price\_QR2. as EBD\_payment,

" " as SSEM\_applicable,

" " as calculation\_SSEM,

**0** as SSEM\_tariff,

calculated HRG\_Code||a.Delivery\_Complications\_Flag\_C as Lookup\_Combination

from MPP\_Par\_CDTable\_X as a

left join MPP\_linkedShQR2\_Delivery as b

on a.Delivery\_Complications\_Flag\_C=b.level

left join OutSAS.MP\_TrimPoint\_&lev. as c

on a.Delivery\_Complications\_Flag=c.Delivery\_Complications\_Flag

where a.Pathway="Delivery"

order by a.Delivery\_Complications\_Flag\_C

;

**quit**;

**data** MPP\_linkedShQR2\_combine;

set

MPP\_linkedShQR2\_Antenatal

MPP\_linkedShQR2\_Postnatal

MPP\_linkedShQR2\_deli\_HRGPrices

;

**run**;

**data** OutSAS.Maternity\_Price\_4\_IA\_Model\_&lev.;

set MPP\_linkedShQR2\_combine;

**run**;

**data** modelout.Maternity\_Price\_4\_IA\_Model\_&lev.;

set OutSAS.Maternity\_Price\_4\_IA\_Model\_&lev.;;

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* ;

\* Date: 06 May 2018 ;

# \* MPP\_Non-Mandatory\_Lev1.sas

\* /17/09/2018 According to request from NHSE, the tariff of home birth is

produced by rolling over 1819 prices ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

/\* Produces Non-Mandatory tariffs for Maternity Pathway Payment (MPP)

Steps\*/

libname TR "\\irnarch\sas\_data\Tariff Rebuild\Maternity Pathway\version8\MPP\_Tariff\_R\SourceData\NonMandatory";

%let outputfile2=Non\_mand\_MPP;

/\* Produces Non-mandatory Non-delivery admitted patient and outpatient procedure prices for maternity services

\*/

/\*APC tariffs\*/

**proc** **sql**;

create table MPP\_Nonmand\_APC\_1 as

SELECT

a.Spell\_HRG as HRG,

a.HRG\_Name,

a.EL\_Tariff,

a.EL\_Trim,

a.NE\_Tariff,

a.NE\_Trim,

a.Longstay\_Payment,

a.SSEM\_Eligible,

a.SSEM\_Tariff,

a.SSNDS\_Eligible

FROM

TR.apc\_tariff\_non\_mand as a

left join ParMPP.MPP\_Par\_CDTable as b

on a.Spell\_HRG=b.HRG

where a.Spell\_HRG Like ('NZ%') and b.Pathway ^="Delivery"

order by a.Spell\_HRG

;

**quit**

;

**proc** **sql**;

create table MPP\_Nonmand\_APC\_2 as

select

a.HRG,

a.HRG\_Name,

b.Elective\_Activity,

b.Non\_Elective\_Activity,

a.EL\_Tariff,

a.NE\_Tariff,

(a.EL\_Tariff\*b.Elective\_Activity+b.Non\_Elective\_Activity\*a.NE\_Tariff)/(b.Elective\_Activity+b.Non\_Elective\_Activity) as Combined\_ELNE\_Tariffs,

a.Longstay\_Payment,

a.SSEM\_Eligible,

a.SSEM\_Tariff,

a.SSNDS\_Eligible

FROM

MPP\_Nonmand\_APC\_1 as a

left join

MPP\_HES\_ACTIVITY\_3 as b

on a.HRG=b.HRG

;

**quit**

;

/\*OPROC tariffs\*/

**proc** **sql**;

create table MPP\_Nonmand\_OPROC\_1 as

select

\*

from

TR.oproc\_final\_tariff\_non\_mand

;

**quit**;

/\*Join in the APC and OPROC tables \*/

**proc** **sql**;

create table MPP\_Nonmand\_APROC\_1 as

select

a.HRG,

a.HRG\_Name,

b.TOTAL\_ACT as OPROC\_Activity,

a.Elective\_Activity,

a.Non\_Elective\_Activity,

b.OPROC\_Tariff,

a.EL\_Tariff,

a.NE\_Tariff,

a.Combined\_ELNE\_Tariffs,

a.Longstay\_Payment,

a.SSEM\_Eligible,

a.SSEM\_Tariff

from MPP\_Nonmand\_APC\_2 as a

left join MPP\_Nonmand\_OPROC\_1 as b

on a.HRG=b.HRG

;

**quit**;

/\*Uplifting adjustment\*/

**PROC** **TRANSPOSE** DATA=Tariff\_R.I\_E

OUT=Trans\_I\_E (drop=Source )

PREFIX=Year

NAME=Source

LABEL=Adjustment\_Type

;

ID Year;

VAR Inflation Efficiency Inflation\_and\_Efficiency\_Combine;

**RUN**; **QUIT**;

/\*Calculate uplift total \*/

**data** PriceAdj\_InflaEffici;

set Trans\_I\_E;

label Adjustment\_Type =Adjustment\_Type;

TotalUplift18\_19=(Year17\_18+**1**)\*(Year18\_19+**1**)-**1**;

label TotalUplift18\_19="Total uplift up to 18/19 price levels";

**run**;

**proc** **sql** noprint;

select TotalUplift18\_19 into :TotalUplift18\_19

from PriceAdj\_InflaEffici

where Adjustment\_Type="Inflation and Efficiency Combined"

;

**quit**;

%let Total\_I\_E\_Adjustment=&TotalUplift18\_19;

%put Total\_I\_E\_Adjustment=&Total\_I\_E\_Adjustment;

**proc** **sql**;

create table MPP\_Nonmand\_APROC\_2 as

select

HRG,

HRG\_Name,

OPROC\_Activity,

Elective\_Activity,

Non\_Elective\_Activity,

OPROC\_Tariff,

EL\_Tariff,

NE\_Tariff,

Combined\_ELNE\_Tariffs,

&Total\_I\_E\_Adjustment as Total\_I\_E\_Adjustment,

OPROC\_Tariff\*(**1**+&Total\_I\_E\_Adjustment) as OPROC\_IE\_Tariffs,

Combined\_ELNE\_Tariffs\*(**1**+&Total\_I\_E\_Adjustment) as Combined\_IE\_Tariffs,

Longstay\_Payment,

SSEM\_Eligible,

SSEM\_Tariff

from MPP\_Nonmand\_APROC\_1

;

**quit**;

/\*Smoothing adjustment\*/

**proc** **sql**;

select SMF into:SMF

from Tariff\_R.maternity\_smoothing

where Year="&FYear."

;

**quit**;

**proc** **sql**;

create table MPP\_Nonmand\_APROC\_3 as

select

HRG,

HRG\_Name,

OPROC\_Activity,

Elective\_Activity,

Non\_Elective\_Activity,

OPROC\_Tariff,

EL\_Tariff,

NE\_Tariff,

Combined\_ELNE\_Tariffs,

Total\_I\_E\_Adjustment,

OPROC\_IE\_Tariffs,

Combined\_IE\_Tariffs,

&SMF as SMF,

OPROC\_IE\_Tariffs\*(**1**+&SMF) as OPROC\_SMF\_Tariffs,

Combined\_IE\_Tariffs\*(**1**+&SMF) as Combined\_SMF\_Tariffs,

Longstay\_Payment,

SSEM\_Eligible,

SSEM\_Tariff

from MPP\_Nonmand\_APROC\_2

;

**quit**;

/\*Scaling factor adjustment\*/

**proc** **sql**;

select SF into:SF

from Tariff\_R.SF

where year="&FYear"

;

**quit**;

**proc** **sql**;

create table MPP\_Nonmand\_APROC\_4\_SF as

select

HRG,

HRG\_Name,

OPROC\_Activity,

Elective\_Activity,

Non\_Elective\_Activity,

OPROC\_Tariff,

round(EL\_Tariff) as EL\_Tariff,

round(NE\_Tariff) as NE\_Tariff,

round(Combined\_ELNE\_Tariffs) as Combined\_ELNE\_Tariffs,

Total\_I\_E\_Adjustment,

round(OPROC\_IE\_Tariffs) as OPROC\_IE\_Tariffs,

round (Combined\_IE\_Tariffs) as Combined\_IE\_Tariffs,

SMF,

round(OPROC\_SMF\_Tariffs) as OPROC\_SMF\_Tariffs,

round(Combined\_SMF\_Tariffs) as Combined\_SMF\_Tariffs,

&SF as SF,

round(OPROC\_SMF\_Tariffs\*(**1**+&SF)) as OPROC\_SF\_Tariffs,

round(Combined\_SMF\_Tariffs\*(**1**&SF)) as Combined\_SF\_Tariffs,

round(Longstay\_Payment) as Longstay\_Payment,

SSEM\_Eligible,

round(SSEM\_Tariff) as SSEM\_Tariff

from MPP\_Nonmand\_APROC\_3

;

**quit**;

/\* Note: PID add MPP\_IandE1920 adjustment on 13/12/2018\*/

**PROC** **SQL**;

create table MPP\_Nonmand\_APROC\_4 as

select

a.HRG,

a.HRG\_Name,

a.OPROC\_Activity,

a.Elective\_Activity,

a.Non\_Elective\_Activity,

a.OPROC\_Tariff,

a.EL\_Tariff,

a.NE\_Tariff,

a.Combined\_ELNE\_Tariffs,

a.Total\_I\_E\_Adjustment,

a.OPROC\_IE\_Tariffs,

a.Combined\_IE\_Tariffs,

a.SMF,

a.OPROC\_SMF\_Tariffs,

a.Combined\_SMF\_Tariffs,

a.SF,

a.OPROC\_SF\_Tariffs,

a.Combined\_SF\_Tariffs

,b.IE\_1-**1** as I\_E\_1920Adjustment

,a.OPROC\_SF\_Tariffs\*b.IE\_1 as OPROC\_Tariffs

,a.Combined\_SF\_Tariffs\*b.IE\_1 as Combined\_Tariffs,

a.Longstay\_Payment,

a.SSEM\_Eligible,

a.SSEM\_Tariff

from MPP\_Nonmand\_APROC\_4\_SF as a,

work.I\_E\_1920 as b

;

**quit**;

/\* According to request by NHSE on 25 May 2018, we implement equalise NZ20B with NZ21B to deal with illogical relativity. \*/

**proc** **sql**;

create table MPP\_Nonmand\_APROC\_MA\_1 as

select

HRG,

HRG\_Name,

(Elective\_Activity+Non\_Elective\_Activity) as combined\_ACT,

Combined\_Tariffs

from MPP\_Nonmand\_APROC\_4

where (HRG="NZ21Z") or (HRG="NZ22Z")

;

**quit**;

**proc** **sql**;

select combined\_ACT into:combined\_ACT\_NZ21Z

from MPP\_Nonmand\_APROC\_MA\_1

where HRG="NZ21Z"

;

**quit**;

**proc** **sql**;

select combined\_ACT into:combined\_ACT\_NZ22Z

from MPP\_Nonmand\_APROC\_MA\_1

where HRG="NZ22Z"

;

**quit**;

**proc** **sql**;

select Combined\_Tariffs into:combined\_Tariff\_NZ21Z

from MPP\_Nonmand\_APROC\_MA\_1

where HRG="NZ21Z"

;

**quit**;

**proc** **sql**;

select Combined\_Tariffs into:combined\_Tariff\_NZ22Z

from MPP\_Nonmand\_APROC\_MA\_1

where HRG="NZ22Z"

;

**quit**;

**Data** MA\_NZ21and22;

HRG="NZ21Z&NZ22Z";

MA\_NZ2122=(&combined\_ACT\_NZ21Z\*&combined\_Tariff\_NZ21Z+&combined\_ACT\_NZ22Z\*&combined\_Tariff\_NZ22Z)/(&combined\_ACT\_NZ21Z+&combined\_ACT\_NZ22Z);

**run**;

**proc** **sql**;

select MA\_NZ2122 into:MA\_NZ2122

from MA\_NZ21and22

;

**quit**;

**proc** **export** data=MPP\_Nonmand\_APROC\_4

outfile="&outputpath.\&outputfile2..XLSX"

dbms=XLSX replace;

sheet="Non\_mandatory\_APROC\_Stepbystep";

**run**;

/\*Create the table for export to Excel version\*/

**proc** **sql**;

create table MPP\_Nonmand\_APROC\_5\_1 as

select

a.HRG,

a.HRG\_Name,

OPROC\_Tariffs,

Combined\_Tariffs,

b.DC\_EL\_NE as Long\_Stay\_TrimPoint,

Longstay\_Payment,

SSEM\_Eligible,

SSEM\_Tariff

from MPP\_Nonmand\_APROC\_4 as a

left join OutSAS.TRANS\_TRIMPOINT\_NM\_&lev. as b

on a.HRG=b.HRG

order by a.HRG

;

**quit**;

/\* Replace modelled prices with manually adjusted prices for NZ21Z adn NZ22Z\*/

**data** MPP\_Nonmand\_APROC\_5;

set MPP\_Nonmand\_APROC\_5\_1;

if HRG="NZ21Z" then Combined\_Tariffs=round(&MA\_NZ2122,**1**);

if HRG="NZ22Z" then Combined\_Tariffs=round(&MA\_NZ2122,**1**);

if HRG="NZ20A" then OPROC\_Tariffs="";

**run**;

/\*Export the table\*/

**proc** **export** data=MPP\_Nonmand\_APROC\_5

outfile="&outputpath.\&outputfile2..XLSX"

dbms=XLSX replace;

sheet="Non\_mandatory\_APC\_OPROC";

**run**;

/\*Non-mandatory outpatient attendance prices for maternity services\*/

**proc** **sql**;

create table MPP\_Nonmand\_OPATT\_1 as

select

Tariff\_TFC\_code,

case

when Tariff\_TFC\_code="501" then "Obstetrics"

when Tariff\_TFC\_code="560" then "Midwifery Service"

end as Treatment\_Function\_Name,

CL\_FAS\_ACT,

CL\_FAM\_ACT,

CL\_FUS\_ACT,

CL\_FUM\_ACT,

CL\_FAS\_TC,

CL\_FAM\_TC,

CL\_FUS\_TC,

CL\_FUM\_TC,

CL\_FAS\_TC/CL\_FAS\_ACT as CL\_FAS\_UC,

CL\_FAM\_TC/CL\_FAM\_ACT as CL\_FAM\_UC,

CL\_FUS\_TC/CL\_FUS\_ACT as CL\_FUS\_UC,

CL\_FUM\_TC/CL\_FUM\_ACT as CL\_FUM\_UC

from MPP\_OPATT\_Input

;

**quit**;

/\*Uplifting adjustment\*/

**PROC** **TRANSPOSE** DATA=Tariff\_R.I\_E

OUT=Trans\_I\_E (drop=Source )

PREFIX=Year

NAME=Source

LABEL=Adjustment\_Type

;

ID Year;

VAR Inflation Efficiency Inflation\_and\_Efficiency\_Combine;

**RUN**; **QUIT**;

/\*Calculate uplift total \*/

**data** PriceAdj\_InflaEffici;

set Trans\_I\_E;

label Adjustment\_Type =Adjustment\_Type;

TotalUplift18\_19=(Year17\_18+**1**)\*(Year18\_19+**1**)-**1**;

label TotalUplift18\_19="Total uplift up to 18/19 price levels";

**run**;

**proc** **sql** noprint;

select TotalUplift18\_19 into :TotalUplift18\_19

from PriceAdj\_InflaEffici

where Adjustment\_Type="Inflation and Efficiency Combined"

;

**quit**;

%let Total\_I\_E\_Adjustment=&TotalUplift18\_19;

%put Total\_I\_E\_Adjustment=&Total\_I\_E\_Adjustment;

**proc** **sql**;

create table MPP\_Nonmand\_OPATT\_2 as

select

Tariff\_TFC\_code,

case

when Tariff\_TFC\_code="501" then "Obstetrics"

when Tariff\_TFC\_code="560" then "Midwifery Service"

end as Treatment\_Function\_Name,

CL\_FAM\_ACT,

CL\_FAS\_ACT,

CL\_FUM\_ACT,

CL\_FUS\_ACT,

CL\_FAM\_TC,

CL\_FAS\_TC,

CL\_FUM\_TC,

CL\_FUS\_TC,

CL\_FAM\_UC,

CL\_FAS\_UC,

CL\_FUM\_UC,

CL\_FUS\_UC,

&Total\_I\_E\_Adjustment as Total\_I\_E\_Adjustment,

CL\_FAM\_UC\*(**1**+&Total\_I\_E\_Adjustment) as FAM\_Total\_I\_E\_Adjustment,

CL\_FAS\_UC\*(**1**+&Total\_I\_E\_Adjustment) as FAS\_Total\_I\_E\_Adjustment,

CL\_FUM\_UC\*(**1**+&Total\_I\_E\_Adjustment) as FUM\_Total\_I\_E\_Adjustment,

CL\_FUS\_UC\*(**1**+&Total\_I\_E\_Adjustment) as FUS\_Total\_I\_E\_Adjustment

from MPP\_Nonmand\_OPATT\_1

;

**quit**;

/\*Manual adjustment

The adjustment is based on EWG feedback\*/

**proc** **sql**;

create table MPP\_Nonmand\_OPATT\_MA\_1 as

select

Tariff\_TFC\_code,

case

when Tariff\_TFC\_code="501" then "Obstetrics"

when Tariff\_TFC\_code="560" then "Midwifery Service"

end as Treatment\_Function\_Name,

CL\_FAM\_ACT,

CL\_FAS\_ACT,

CL\_FUM\_ACT,

CL\_FUS\_ACT,

CL\_FAM\_TC,

CL\_FAS\_TC,

CL\_FUM\_TC,

CL\_FUS\_TC,

CL\_FAM\_UC,

CL\_FAS\_UC,

CL\_FUM\_UC,

CL\_FUS\_UC,

&Total\_I\_E\_Adjustment as Total\_I\_E\_Adjustment,

FAM\_Total\_I\_E\_Adjustment,

FAS\_Total\_I\_E\_Adjustment,

FUM\_Total\_I\_E\_Adjustment,

FUS\_Total\_I\_E\_Adjustment,

FAM\_Total\_I\_E\_Adjustment as FAS\_MA,

FAS\_Total\_I\_E\_Adjustment as FAM\_MA,

FUS\_Total\_I\_E\_Adjustment as FUS\_MA,

FUM\_Total\_I\_E\_Adjustment as FUM\_MA

from MPP\_Nonmand\_OPATT\_2

where Tariff\_TFC\_code="560"

;

**quit**;

**proc** **sql**;

create table MPP\_Nonmand\_OPATT\_MA\_2 as

select

Tariff\_TFC\_code,

case

when Tariff\_TFC\_code="501" then "Obstetrics"

when Tariff\_TFC\_code="560" then "Midwifery Service"

end as Treatment\_Function\_Name,

CL\_FAM\_ACT,

CL\_FAS\_ACT,

CL\_FUM\_ACT,

CL\_FUS\_ACT,

CL\_FAM\_TC,

CL\_FAS\_TC,

CL\_FUM\_TC,

CL\_FUS\_TC,

CL\_FAM\_UC,

CL\_FAS\_UC,

CL\_FUM\_UC,

CL\_FUS\_UC,

&Total\_I\_E\_Adjustment as Total\_I\_E\_Adjustment,

FAM\_Total\_I\_E\_Adjustment,

FAS\_Total\_I\_E\_Adjustment,

FUM\_Total\_I\_E\_Adjustment,

FUS\_Total\_I\_E\_Adjustment,

FAS\_Total\_I\_E\_Adjustment as FAS\_MA,

FAM\_Total\_I\_E\_Adjustment as FAM\_MA,

FUS\_Total\_I\_E\_Adjustment as FUS\_MA,

FUM\_Total\_I\_E\_Adjustment as FUM\_MA

from MPP\_Nonmand\_OPATT\_2

where Tariff\_TFC\_code="501"

;

**quit**;

**data** MPP\_Nonmand\_OPATT\_3;

set MPP\_Nonmand\_OPATT\_MA\_2 MPP\_Nonmand\_OPATT\_MA\_1;

**run**;

/\*Smoothing adjustment\*/

**proc** **sql**;

create table MPP\_Nonmand\_OPATT\_4 as

select

Tariff\_TFC\_code,

case

when Tariff\_TFC\_code="501" then "Obstetrics"

when Tariff\_TFC\_code="560" then "Midwifery Service"

end as Treatment\_Function\_Name,

CL\_FAS\_ACT,

CL\_FAM\_ACT,

CL\_FUS\_ACT,

CL\_FUM\_ACT,

CL\_FAS\_TC,

CL\_FAM\_TC,

CL\_FUS\_TC,

CL\_FUM\_TC,

CL\_FAS\_UC,

CL\_FAM\_UC,

CL\_FUS\_UC,

CL\_FUM\_UC,

&Total\_I\_E\_Adjustment as Total\_I\_E\_Adjustment,

FAS\_Total\_I\_E\_Adjustment,

FAM\_Total\_I\_E\_Adjustment,

FUS\_Total\_I\_E\_Adjustment,

FUM\_Total\_I\_E\_Adjustment,

FAS\_MA,

FAM\_MA,

FUS\_MA,

FUM\_MA,

&SMF as SMF,

FAS\_MA\*(**1**+&SMF)as FAS\_SMF,

FAM\_MA\*(**1**+&SMF)as FAM\_SMF,

FUS\_MA\*(**1**+&SMF)as FUS\_SMF,

FUM\_MA\*(**1**+&SMF)as FUM\_SMF

from MPP\_Nonmand\_OPATT\_3

;

**quit**;

/\*Scaling factor adjustment\*/

**proc** **sql**;

create table MPP\_Nonmand\_OPATT\_5\_SF as

select

Tariff\_TFC\_code,

case

when Tariff\_TFC\_code="501" then "Obstetrics"

when Tariff\_TFC\_code="560" then "Midwifery Service"

end as Treatment\_Function\_Name,

CL\_FAS\_ACT,

CL\_FAM\_ACT,

CL\_FUS\_ACT,

CL\_FUM\_ACT,

CL\_FAS\_TC,

CL\_FAM\_TC,

CL\_FUS\_TC,

CL\_FUM\_TC,

CL\_FAS\_UC,

CL\_FAM\_UC,

CL\_FUS\_UC,

CL\_FUM\_UC,

&Total\_I\_E\_Adjustment as Total\_I\_E\_Adjustment,

FAS\_Total\_I\_E\_Adjustment,

FAM\_Total\_I\_E\_Adjustment,

FUS\_Total\_I\_E\_Adjustment,

FUM\_Total\_I\_E\_Adjustment,

FAS\_MA,

FAM\_MA,

FUS\_MA,

FUM\_MA,

SMF,

FAS\_SMF,

FAM\_SMF,

FUS\_SMF,

FUM\_SMF,

&SF as SF,

FAS\_SMF\*(**1**+&SF)as FAS\_SF,

FAM\_SMF\*(**1**+&SF)as FAM\_SF,

FUS\_SMF\*(**1**+&SF)as FUS\_SF,

FUM\_SMF\*(**1**+&SF)as FUM\_SF

from MPP\_Nonmand\_OPATT\_4

;

**quit**;

/\* MPP\_IandE1920 adjustment\*/

**PROC** **SQL**;

create table MPP\_Nonmand\_OPATT\_5 as

select

a.\*

,b.IE\_1-**1** as I\_E\_1920Adjustment

,a.FAS\_SF\*b.IE\_1 as FAS\_FinalPrices

,a.FAM\_SF\*b.IE\_1 as FAM\_FinalPrices

,a.FUS\_SF\*b.IE\_1 as FUS\_FinalPrices

,a.FUM\_SF\*b.IE\_1 as FUM\_FinalPrices

from MPP\_Nonmand\_OPATT\_5\_SF as a,

work.I\_E\_1920 as b

;

**quit**;

**proc** **export** data=MPP\_Nonmand\_OPATT\_5

outfile="&outputpath.\&outputfile2..XLSX"

dbms=XLSX replace;

sheet="Non\_mandatory\_OPATT\_Stepbystep";

**run**;

/\*Prepare a table to export\*/

**proc** **sql**;

create table MPP\_Nonmand\_OPATT\_6 as

select

Tariff\_TFC\_code,

case

when Tariff\_TFC\_code="501" then "Obstetrics"

when Tariff\_TFC\_code="560" then "Midwifery Service"

end as Treatment\_Function\_Name,

FAS\_FinalPrices as WF01B\_FAS,

FAM\_FinalPrices as WF02B\_FAM,

FUS\_FinalPrices as WF01A\_FUS,

FUM\_FinalPrices as WF02A\_FUM

from MPP\_Nonmand\_OPATT\_5

;

**quit**;

/\*Export the table\*/

**proc** **export** data=MPP\_Nonmand\_OPATT\_6

outfile="&outputpath.\&outputfile2..XLSX"

dbms=XLSX replace;

sheet="Non\_mandatory\_OPATT";

**run**;

/\*Non-mandatory community prices for maternity services\*/

**proc** **sql**;

create table MPP\_Nonmand\_Community\_1 as

SELECT

HRG,

case

when HRG="N01A" then "Antenatal Visits"

when HRG="N01P" then "Postnatal Visits"

end as Name,

sum(Activity) as Activity,

sum(TC) as TC,

calculated TC/calculated Activity as UnitCost

from

work.MPP\_Community\_Input

where

HRG like'N01%'

group by HRG

;

**quit**;

**PROC** **TRANSPOSE** DATA=Tariff\_R.I\_E

OUT=Trans\_I\_E (drop=Source )

PREFIX=Year

NAME=Source

LABEL=Adjustment\_Type

;

ID Year;

VAR Inflation Efficiency Inflation\_and\_Efficiency\_Combine;

**RUN**; **QUIT**;

/\*Calculate uplift total \*/

**data** PriceAdj\_InflaEffici;

set Trans\_I\_E;

label Adjustment\_Type =Adjustment\_Type;

TotalUplift18\_19=(Year17\_18+**1**)\*(Year18\_19+**1**)-**1**;

label TotalUplift18\_19="Total uplift up to 18/19 price levels";

**run**;

**proc** **sql** noprint;

select TotalUplift18\_19 into :TotalUplift18\_19

from PriceAdj\_InflaEffici

where Adjustment\_Type="Inflation and Efficiency Combined"

;

**quit**;

%let Total\_I\_E\_Adjustment=&TotalUplift18\_19;

%put Total\_I\_E\_Adjustment=&Total\_I\_E\_Adjustment;

/\*Adjusted by total uplift factor\*/

**proc** **sql**;

create table MPP\_Nonmand\_Community\_2 as

SELECT

HRG,

Name,

Activity,

TC,

UnitCost,

&Total\_I\_E\_Adjustment as Total\_I\_E\_Adjustment,

UnitCost\*(**1**+&Total\_I\_E\_Adjustment) as Prices\_Nonmand\_Total\_I\_E

from

work.MPP\_Nonmand\_Community\_1

;

**quit**;

/\*Smoothing Factor\*/

**proc** **sql**;

create table MPP\_Nonmand\_Community\_3 as

SELECT

HRG,

Name,

Activity,

TC,

UnitCost,

Total\_I\_E\_Adjustment,

Prices\_Nonmand\_Total\_I\_E,

&SMF as SMF,

Prices\_Nonmand\_Total\_I\_E\*(**1**+&SMF) as Prices\_SMF

from

work.MPP\_Nonmand\_Community\_2

;

**quit**;

/\*Scaling factor adjustment\*/

**proc** **sql**;

create table MPP\_Nonmand\_Community\_4\_SF as

SELECT

HRG,

Name,

Activity,

TC,

UnitCost,

Total\_I\_E\_Adjustment,

Prices\_Nonmand\_Total\_I\_E,

SMF,

Prices\_SMF,

&SF as SF,

Prices\_SMF\*(**1**+&SF) as Prices\_SF

from

work.MPP\_Nonmand\_Community\_3

;

**quit**;

/\* MPP\_IandE1920 adjustment\*/

**PROC** **SQL**;

create table MPP\_Nonmand\_Community\_4 as

select

a.\*

,b.IE\_1-**1** as I\_E\_1920Adjustment

,a.Prices\_SF\*b.IE\_1 as Non\_mand\_prices\_comm

from MPP\_Nonmand\_Community\_4\_SF as a,

work.I\_E\_1920 as b

;

**quit**;

**proc** **export** data=MPP\_Nonmand\_Community\_4

outfile="&outputpath.\&outputfile2..XLSX"

dbms=XLSX replace;

sheet="Non\_mandatory\_Comm\_Stepbystep";

**run**;

/\*Prepare for exporting Table\*/

**proc** **sql**;

create table MPP\_Nonmand\_Community\_5 as

SELECT

HRG,

Name,

Non\_mand\_prices\_comm

from

work.MPP\_Nonmand\_Community\_4

;

**quit**;

/\*Export the table\*/

**proc** **export** data=MPP\_Nonmand\_Community\_5

outfile="&outputpath.\&outputfile2..XLSX"

dbms=XLSX replace;

sheet="Non\_mandatory\_Community";

**run**;

/\*Table Non-mandatory antenatal assessment visit for maternity services\*/

/\*import 2018/19 tariffs\*/

**proc** **import** out=Nonmand\_assevisit

datafile= "&datafilepath.\1819NonMand\_Anatenatal\_Visit"

dbms=xlsx replace;

sheet="1819\_Nonmand\_assevisit";

getnames=yes;

**run**;

**proc** **import** out=Nonmand\_antvisit

datafile= "&datafilepath.\1819NonMand\_Anatenatal\_Visit"

dbms=xlsx replace;

sheet="1819\_Nonmand\_antvisit";

getnames=yes;

**run**;

/\*Calculate relativity and tariffs for assessment visits\*/

/\*standard\*/

**proc** **sql**;

select Tariffs into:st\_tariff\_antenatal

from Nonmand\_assevisit

where Name="Standard"

;

**quit**;

**proc** **sql**;

select Tariff format=**8.0** into:st\_tariff\_Community

from Nonmand\_antvisit

where Name="Antenatal Visits"

;

**quit**;

**proc** **sql**;

select UnitCost into: ante\_UC\_Community

from MPP\_Nonmand\_Community\_1

where HRG="N01A"

;

**quit**;

**Data** C\_1;

Name="Standard";

Relativity=&st\_tariff\_antenatal/&st\_tariff\_Community;

Tariff=Relativity\*&ante\_UC\_Community;

**run**;

/\*Intermediate\*/

**proc** **sql**;

select Tariffs into:st\_tariff\_intermediate

from Nonmand\_assevisit

where Name="Intermediate"

;

**quit**;

**proc** **sql**;

select Tariffs into:st\_tariff\_standard

from Nonmand\_assevisit

where Name="Standard"

;

**quit**;

**proc** **sql**;

select Tariff into:Tariff\_s

from C\_1

where Name="Standard"

;

**quit**;

**Data** C\_2;

Name="Intermediate";

Relativity=&st\_tariff\_intermediate/&st\_tariff\_standard;

Tariff=Relativity\*&Tariff\_s;

**run**;

/\*Intensive\*/

**proc** **sql**;

select Tariffs into:st\_tariff\_intensive

from Nonmand\_assevisit

where Name="Intensive"

;

**quit**;

**proc** **sql**;

select Tariffs into:st\_tariff\_standard

from Nonmand\_assevisit

where Name="Standard"

;

**quit**;

**proc** **sql**;

select Tariff into:Tariff\_s

from C\_1

where Name="Standard"

;

**quit**;

**Data** C\_3;

Name="Intensive";

Relativity=&st\_tariff\_intensive/&st\_tariff\_standard;

Tariff=Relativity\*&Tariff\_s;

**run**;

/\*Pending three tables together\*/

**data** Non\_Mand\_ant\_assessmentvisit;

set C\_1 C\_2 C\_3;

**run**;

/\*Uplift adjustment\*/

**proc** **sql**;

create table Non\_Mand\_table11\_assvst\_1

as select

Name,

Relativity,

Tariff,

&Total\_I\_E\_Adjustment as Total\_I\_E\_Adjustment,

Tariff\*(**1**+&Total\_I\_E\_Adjustment) as Prices\_table11\_Total\_I\_E

from Non\_Mand\_ant\_assessmentvisit

;

**quit**

;

/\*Smoothing adjustment\*/

**proc** **sql**;

create table Non\_Mand\_table11\_assvst\_2

as select

Name,

Relativity,

Tariff,

Total\_I\_E\_Adjustment,

Prices\_table11\_Total\_I\_E,

&SMF as SMF,

Prices\_table11\_Total\_I\_E\*(**1**+&SMF) as Prices\_SMF

from Non\_Mand\_table11\_assvst\_1

;

**quit**

;

/\*Scaling adjustment\*/

**proc** **sql**;

create table Non\_Mand\_table11\_assvst\_3\_SF

as select

Name,

Relativity,

Tariff,

Total\_I\_E\_Adjustment,

Prices\_table11\_Total\_I\_E,

SMF,

Prices\_SMF,

&SF as SF,

Prices\_SMF\*(**1**+&SF) as Prices\_SF

from Non\_Mand\_table11\_assvst\_2

;

**quit**

;

/\* MPP\_IandE1920 adjustment\*/

**PROC** **SQL**;

create table Non\_Mand\_table11\_assvst\_3 as

select

a.\*

,b.IE\_1-**1** as I\_E\_1920Adjustment

,a.Prices\_SF\*b.IE\_1 as Tariffs

from Non\_Mand\_table11\_assvst\_3\_SF as a,

work.I\_E\_1920 as b

;

**quit**;

**proc** **export** data=Non\_Mand\_table11\_assvst\_3

outfile="&outputpath.\&outputfile2..XLSX"

dbms=XLSX replace;

sheet="Non\_mandatory\_asst\_Stepbystep";

**run**;

/\*Prepare for exporting table 11\*/

**proc** **sql**;

create table Non\_Mand\_table11\_assvst\_4

as select

Name,

Tariffs

from Non\_Mand\_table11\_assvst\_3

;

**quit**

;

/\*Export the table 10\*/

**proc** **export** data=Non\_Mand\_table11\_assvst\_4

outfile="&outputpath.\&outputfile2..XLSX"

dbms=XLSX replace;

sheet="Non\_mandatory\_Assessment";

**run**;

/\*17/09/2018 According to request from NHSE, the tariff of home birth is

produced by rolling over 1819 prices \*/

/\*import 2018/19 tariffs\*/

**proc** **import** out=Homebirth

datafile= "&datafilepath.\1819NonMand\_Anatenatal\_Visit"

dbms=xlsx replace;

sheet="1819\_mand\_delivery";

getnames=yes;

**run**;

/\*create homebirth price table\*/

**proc** **sql**;

create table Homebirth\_1

as select

case

when Name="Without complications and co-morbidities" then "Homebirth"

end as Name,

Tariff

from work.HOMEBIRTH

WHERE Name="Without complications and co-morbidities"

;

**quit**

;

/\*Smoothing adjustment\*/

**proc** **sql**;

create table Homebirth\_2

as select

\*,

&SMF as SMF,

Tariff\*(**1**+&SMF) as Prices\_SMF

from work.Homebirth\_1

;

**quit**

;

/\*Scaling adjustment\*/

**proc** **sql**;

create table Homebirth\_3\_SF

as select

\*

,&SF as SF,

Prices\_SMF\*(**1**+&SF) as Prices\_SF

from Homebirth\_2

;

**quit**

;

/\* CNST adjustment for Homebirth price\*/

**PROC** **SQL**;

create table Homebirth\_3\_CNST as

select

a.\*

,b.CNST1920-**1** as CNST1920

,a.Prices\_SF\*(**1**+(b.CNST1920-**1**)) as Prices\_CNST1920

from Homebirth\_3\_SF as a,

work.MPP\_CNST\_FACTOR as b

;

**quit**;

/\* MPP\_IandE1920 adjustment\*/

**PROC** **SQL**;

create table Homebirth\_3 as

select

a.\*

,b.IE\_1-**1** as I\_E\_1920Adjustment

,a.Prices\_CNST1920\*b.IE\_1 as Tariffs

from Homebirth\_3\_CNST as a,

work.I\_E\_1920 as b

;

**quit**;

/\*Export the table\*/

**PROC** **SQL**;

create table Homebirth\_4 as

select

Name

,Tariffs

from Homebirth\_3

;

**quit**;

**proc** **export** data=Homebirth\_4

outfile="&outputpath.\&outputfile2..XLSX"

dbms=XLSX replace;

sheet="Homebirth";

**run**;

# \* Non\_Mandatory\_Trim\_Point\_Calculation(For 1 level HRGs) using default SAS method to calculate Quartiles.

/\* Version: 1.01 \*/

/\* Author: PID \*/

/\* Date: May 2018 \*/

/\*----------------------------------------------------------------\*/

/\* Input: \*/

/\* (1)PreProcess.Stage\_HES\_KeyFields \*/

/\* \*/

/\* \*/

/\* Output: \*/

/\* (1)OutSAS.Trans\_TrimPoint\_NM\_&lev. \*/

/\* \*/

/\* \*/

/\*----------------------------------------------------------------\*/

/\* ALL levels \*/

/\*Create Non\_Mandatory HRG list \*/

**data** Non\_Mand\_HRGS;

input HRG $ ;

datalines;

NZ16Z

NZ17A

NZ17B

NZ18A

NZ18B

NZ19A

NZ19B

NZ20A

NZ20B

NZ21Z

NZ22Z

NZ24A

NZ24B

NZ25Z

NZ26A

NZ26B

NZ27Z

;

**run**;

/\*Combined HRGs: create unique HRG list for Combined HRGs \*/

**proc** **sql**;

create table T\_CB\_uniqu\_HRG

as

select distinct

'"' as leftquote,

HRG,

'"' as rightquote,

strip(calculated leftquote)||strip(HRG)||strip(calculated rightquote) as Q\_HRG

from

Non\_Mand\_HRGS

order by HRG

;

**quit**;

/\*Combined HRGs: output unique HRG List \*/

**proc** **sql** noprint;

select Q\_HRG into :HRGlist\_CB separated by ','

from T\_CB\_uniqu\_HRG

;

**quit**;

%put HRGlist\_CB=&HRGlist\_CB.;

/\*create Adm and Adm\_Type table \*/

**proc** **sql**;

create table Admtype\_Adm\_Table

(Rn num format=**8.0**,

Adm\_type char(**12**),

Adm char(**12**)

);

INSERT INTO Admtype\_Adm\_Table

values( **1**,'DC/EL','DC')

values( **1**,'DC/EL','EL')

values( **2**,'NE','NE')

values( **3**,'DC/EL/NE','DC')

values( **3**,'DC/EL/NE','EL')

values( **3**,'DC/EL/NE','NE' )

values( **4**,'DC','DC')

values( **5**,'EL','EL')

;

**quit**;

/\* Create a full list of HRG\_AdmType\_Adm list \*/

**proc** **sql**;

create table T\_HRG\_AdmType\_Adm

as select distinct

a.HRG ,

b.adm\_type,

b.adm

from Non\_Mand\_HRGS as a ,admtype\_adm\_Table as b

order by a.HRG, b.adm\_type

;

**quit**;

/\*get data from bprPrPP.Stage\_HES\_KeyFields \*/

**proc** **sql**;

create table HES\_KeyFields\_Data

as

select

RC\_SpellHRG\_Rev as HRG,

ADM,

Run\_ID,

Revised\_Excluded,

spellflag,

PROCODETD,

Spell\_LoS\_2yr

FROM bprPrPP.Stage\_HES\_KeyFields

where RC\_SpellHRG\_Rev in ( &HRGlist\_CB.) and Run\_ID = &Run\_ID.

and Revised\_Excluded = 'N' and spellflag = **1** and substr(PROCODETD,**1**,**1**) in ('5','R','T') and adm<>"UX"

order by RC\_SpellHRG\_Rev, adm

;

**quit**;

/\*Prepare data for TP Calculation\*/

**proc** **sql**;

create table Data\_for\_Cal\_Trimpoint\_NM

as

select a.HRG,

a.adm\_type,

a.adm,

&Run\_ID as Run\_ID,

b.Spell\_LoS\_2yr

from T\_HRG\_AdmType\_Adm as a

left join

HES\_KeyFields\_Data as b

on A.HRG=b.HRG and a.adm=b.adm

order by a.HRG, a.adm\_Type

;

**quit**;

/\*Calculate the Trim Points \*/

**proc** **means** data=Data\_for\_Cal\_Trimpoint\_NM missing MAxdec=**1**

N Mean MEDIAN nmiss P25 P50 P75 noprint;

by HRG adm\_type;

output out=TrimPointStats\_NM N=N Mean=Mean Median=Median nmiss=NMiss P25=P25 P50=P50 P75=P75;

var Spell\_LoS\_2yr;

**run**;

/\* Trim Point Output table\*/

**proc** **sql**;

create table TrimPointCal\_NM\_1 as

select

&Run\_ID as Run\_ID,

HRG,

adm\_type,

(\_FREQ\_-coalesce(nmiss,**0**)) as Valid,

NMiss as Missing label="nmiss",

coalesce(Mean,**0**) as Mean label="Mean" format=**8.2**,

coalesce(Median,**0**) as Median label="Median" format=**8.2**,

coalesce(P25,**0**) as P25 label="P25" format=**8.2**,

coalesce(P50,**0**) as P50 label="P50" format=**8.2**,

coalesce(P75,**0**) as P75 label="P75" format=**8.2**,

coalesce(P75 + ((P75 - P25) \* **1.5**),**0**) as Trimpoint\_Cal format=**8.1**,

case when round( Calculated Trimpoint\_Cal)=**0** then **1**

else round(Calculated Trimpoint\_Cal)

end as TrimPoint\_B

from TrimPointStats\_NM

where adm\_type^="DC"

order by HRG, adm\_type

;

**quit**;

/\*Apply minimum 5 days rule to trim point \*/

**data** TrimPointCal\_NM;

set TrimPointCal\_NM\_1;

TrimPoint=TrimPoint\_B;

if TrimPoint<**5** then TrimPoint=**5**;

**run**;

**proc** **sql**;

create table trim\_point\_NM\_for\_Trans

as

select

Run\_ID,

HRG,

Adm\_type,

TrimPoint

from TrimPointCal\_NM

order by HRG, Adm\_type

;

**quit**;

**PROC** **TRANSPOSE** DATA=trim\_point\_NM\_for\_Trans

OUT=Trans\_TrimPoint\_NM\_1

NAME=Source

LABEL=Label

;

BY HRG;

ID Adm\_type;

VAR TrimPoint;

COPY Run\_ID;

**RUN**; **QUIT**;

**proc** **sql** noprint;

create table Trans\_TrimPoint\_NM

as

select

a.Run\_ID,

a.HRG,

/\*a.DC,\*/

a.DC\_EL,

a.DC\_EL\_NE,

a.EL,

a.NE

from Trans\_TrimPoint\_NM\_1 as a

where source="TrimPoint"

order by HRG

;

**quit**;

**data** OutSAS.Trans\_TrimPoint\_NM\_&lev.;

set Trans\_TrimPoint\_NM;

**run**;

%let outputfile=OutSAS.Trans\_TrimPoint\_NM\_&lev.;

%let outputName=MPP\_TrimPoint\_NM\_&lev.;

**proc** **export** data=&outputfile.

outfile="&outputpath.\&outputName..csv"

dbms=csv

replace;

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* Author: PID ;

\* ;

\* Date: May 2018 ;

# \* MPP\_Linksheet.sas

\* 17/09/2018 According to request from NHSE, the tariff of home birth is

produced by rolling over 1819 prices ;;

\* Version: 1.02 ;

\* Description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

/\*ODS to create Excel workbook output with multiple sheets\*/

ods \_all\_ close;

ods tagsets.ExcelXP path="\\irnarch\sas\_data\Tariff Rebuild\Models for Testing\TC\_Y1920\Excel\_outputs\&username." file="19\_20 Non\_Mandatory\_Report.xml" style=Printer;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\*\*\*\*\*\* Step 5 Run Maternity Model \*\*\*\*\*\*;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

ods tagsets.ExcelXP options(sheet\_name='Non\_mandatory\_APC\_OPROC'

suppress\_bylines='yes'

frozen\_headers='1'

FROZEN\_ROWHEADERS='1'

autofit\_height='Yes'

absolute\_column\_width='10, 40, 10, 10, 10, 10,10, 10, 10, 10, 10, 10, 10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10,'

pages\_fitwidth='1'

pages\_fitheight='9'

FitToPage="yes"

autofilter='1-21');

/\*Define STEP BY STEP QUANTUM RECONCILIATION BY Sub Chapter variables\*/

/\*MPP\_Nonmand\_APROC\*/

/\*Define STEP BY STEP QUANTUM RECONCILIATION BY Sub Chapter variables\*/

**proc** **print** data=MPP\_Nonmand\_APROC\_5 split='\*' noobs;

var HRG HRG\_Name/style(header)=[background=#b3ffff];

var OPROC\_Tariffs

Combined\_Tariffs

Long\_Stay\_TrimPoint

Longstay\_Payment / style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###"];

var SSEM\_Eligible/style(header)=[background=#b3ffff];

var SSEM\_Tariff/ style(header)=[background=#b3ffff] ;

;

label

HRG='HRG code'

HRG\_Name='HRG name'

OPROC\_Tariffs='Outpatient \*procedure \*(£)\_\_'

Combined\_Tariffs='Combined day case /\*ordinary elective / \*non-elective spell \*(£)\_\_'

Long\_Stay\_TrimPoint='Long stay trim \*point \*(days)\_\_'

Longstay\_Payment='Per day long stay \*payment (for days \*exceeding trim \*point) \*(£)\_\_'

SSEM\_Eligible='Reduced short stay \*emergency tariff \*applicable?\_\_'

SSEM\_Tariff='Reduced short stay \*emergency tariff \*(£)\_\_'

;

**run**;

/\*MPP\_Nonmand\_OPATT\*/

ods tagsets.ExcelXP options(sheet\_name='MPP\_Nonmand\_OPATT'

suppress\_bylines='yes'

frozen\_headers='1'

FROZEN\_ROWHEADERS='1'

autofit\_height='Yes'

absolute\_column\_width='10, 40, 10, 10, 10, 10,10, 10, 10, 10, 10, 10, 10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10,'

pages\_fitwidth='1'

pages\_fitheight='9'

FitToPage="yes"

autofilter='1-21');

/\*Define STEP BY STEP QUANTUM RECONCILIATION BY Sub Chapter variables\*/

/\*Define STEP BY STEP QUANTUM RECONCILIATION BY Sub Chapter variables\*/

**proc** **print** data=MPP\_NONMAND\_OPATT\_6 split='\*' noobs;

var Tariff\_TFC\_code Treatment\_Function\_Name/style(header)=[background=#b3ffff];

var WF01B\_FAS

WF02B\_FAM

WF01A\_FUS

WF02A\_FUM/ style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###"];

label

Tariff\_TFC\_code="Treatment function"

Treatment\_Function\_Name="Treatment function name"

WF01B\_FAS="WF01B\*First Attendance - \*Single Professional\*(£)\_\_"

WF02B\_FAM="WF02B\*First Attendance - \*Multi Professional\*(£)\_\_"

WF01A\_FUS="WF01A\*Follow Up Attendance - \*Single Professional\*(£)\_\_"

WF02A\_FUM="WF02A\*Follow Up Attendance - \*Multi Professional\*(£)\_\_"

;

**run**;

/\* Non\_mandatory\_Community \*/

ods tagsets.ExcelXP options(sheet\_name='MPP\_Nonmand\_Community'

suppress\_bylines='yes'

frozen\_headers='1'

FROZEN\_ROWHEADERS='1'

autofit\_height='Yes'

absolute\_column\_width='10, 40, 10, 10, 10, 10,10, 10, 10, 10, 10, 10, 10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10,'

pages\_fitwidth='1'

pages\_fitheight='9'

FitToPage="yes"

autofilter='1-21');

/\*Define STEP BY STEP QUANTUM RECONCILIATION BY Sub Chapter variables\*/

/\*Define STEP BY STEP QUANTUM RECONCILIATION BY Sub Chapter variables\*/

**proc** **print** data=MPP\_Nonmand\_Community\_5 split='\*' noobs;

var HRG Name/style(header)=[background=#b3ffff];

var Non\_mand\_prices\_comm/ style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###"];

label

HRG='HRG code'

Name='HRG name'

Non\_mand\_prices\_comm="Tariff for \*Community\*(£)\_\_"

;

**run**;

/\*Non-mandatory antenatal assessment visit for maternity services\*/

ods tagsets.ExcelXP options(sheet\_name='Nonmand\_assevisit'

suppress\_bylines='yes'

frozen\_headers='1'

FROZEN\_ROWHEADERS='1'

autofit\_height='Yes'

absolute\_column\_width='10, 40, 10, 10, 10, 10,10, 10, 10, 10, 10, 10, 10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10,'

pages\_fitwidth='1'

pages\_fitheight='9'

FitToPage="yes"

autofilter='1-21');

/\*Define STEP BY STEP QUANTUM RECONCILIATION BY Sub Chapter variables\*/

/\*Define STEP BY STEP QUANTUM RECONCILIATION BY Sub Chapter variables\*/

**proc** **print** data=Non\_Mand\_table11\_assvst\_4 split='\*' noobs;

var Name/style(header)=[background=#b3ffff];

var Tariffs/ style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###"];

label

Tariffs="Tariff for \*antenatal assessment \*visit\*(£)\_\_"

;

**run**;

/\*Homebirth\*/

ods tagsets.ExcelXP options(sheet\_name='Homebirth'

suppress\_bylines='yes'

frozen\_headers='1'

FROZEN\_ROWHEADERS='1'

autofit\_height='Yes'

absolute\_column\_width='10, 40, 10, 10, 10, 10,10, 10, 10, 10, 10, 10, 10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10,'

pages\_fitwidth='1'

pages\_fitheight='9'

FitToPage="yes"

autofilter='1-21');

**proc** **print** data=Homebirth\_4 split='\*' noobs;

var Name/style(header)=[background=#b3ffff];

var Tariffs/ style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###"];

label

Tariffs="Tariff for \*Homebirth \*visit\*(£)\_\_"

;

**run**;

/\*Non\_mandatory\_APROC\_Step by step\*/

/\*MPP\_Nonmand\_APROC\_4 \*/

ods tagsets.ExcelXP close;

ods listing;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 26 April 2017 ;

\* OMLoadData.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* Connects to SQL by ODBC to get non acute ref cost data

\* Formats to get data needed for OM model

\* Connects to SQL by ODBC to get the provider table (to be replaced by SAS connection;

**%macro** OMLoadData(schema=);

proc sql; create table OMalldata as select \* from work.Reference\_Cost WHERE DEPARTMENTcode="DA"; quit;

proc sql;

create table OM\_Filter\_For\_Scope as

select

organisationcode as Prov\_Code

,DEPARTMENTcode as department

,SERVICEcode as service

,CURRENCYcode as currency

,UNITCOST as UC

,Activity

,(UNITCOST \* ACTIVITY) as TOTAL\_COST

from OMalldata

WHERE DEPARTMENTcode = "DA";

quit;

\*/ Added output of data ;

%let outputfile=OM\_Filter\_For\_Scope;

proc export data=WORK.OM\_Filter\_For\_Scope

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

run;

**%mend** OMLoadData;

%***OMLoadData***(schema=sas\_config)

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 26 April 2017 ;

\* OMremoveMFF.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* Joins provider and scope tables

\* Removes MFF values ;

**%macro** OMremoveMFF(schema=);

proc sql;

create table OM\_Remove\_MFF as

select

ffs.Prov\_Code,

ffs.DEPARTMENT,

ffs.SERVICE,

ffs.CURRENCY,

ffs.UC,

ffs.Activity,

ffs.TOTAL\_COST,

ffs.UC / p.uncapped\_MFF as UC\_Target\_MFF,

ffs.TOTAL\_COST / p.uncapped\_MFF as TC\_Target\_MFF,

ffs.TOTAL\_COST / p.Capped\_MFF as TC\_Payment\_MFF

from OM\_Filter\_For\_Scope ffs inner join work.tc\_mff p on ffs.prov\_code = p.RC\_Code;

quit;

\*/ Added output of data ;

%let outputfile=OM\_Remove\_MFF;

proc export data=WORK.OM\_Remove\_MFF

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

run;

**%mend** OMremoveMFF;

%***OMremoveMFF***(schema=sas\_config)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 26 April 2017 ;

\* OMNationalAvg.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* Calulcates national averages;

**%macro** OMNationalAvg(schema=);

proc sql;

create table OM\_nat\_avg as

select department,

service,

currency,

sum( UC\_Target\_MFF \* ACTIVITY) / sum(ACTIVITY) as National\_Avg

from OM\_Remove\_MFF

group by department, service , currency;

quit;

\*/ Added output of data ;

%let outputfile=OM\_nat\_avg;

proc export data=WORK.OM\_nat\_avg

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

run;

**%mend** OMNationalAvg;

%***OMNationalAvg***(schema=sas\_config)

; \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 26 April 2017 ;

\* OMDataClean.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* Cleans the data using the 5 and 20 rules, no grubbs no other data cleaning ;

**%macro** OMDataClean(schema=);

proc sql;

create table OM\_clean\_data as

select

rmff.Prov\_Code,

rmff.DEPARTMENT,

rmff.SERVICE,

rmff.CURRENCY,

rmff.UC,

rmff.Activity,

rmff.TOTAL\_COST,

rmff.UC\_Target\_MFF,

rmff.TC\_Target\_MFF,

rmff.TC\_Payment\_MFF,

CASE WHEN (rmff.UC\_Target\_MFF / na.National\_Avg)<**0.05** OR (rmff.UC\_Target\_MFF / na.National\_Avg) > **20** THEN **1** ELSE **0** END as Anomaly

from OM\_Remove\_MFF rmff inner join OM\_nat\_avg na on

rmff.currency = na.currency and

rmff.service = na.service and

rmff.department = na.department

where

(case when rmff.UC\_Target\_MFF / na.National\_Avg < **0.05** or rmff.UC\_Target\_MFF / na.National\_Avg > **20**

then **1** else **0** end) = **0**;

quit;

\*/ Added output of data ;

%let outputfile=OM\_clean\_data;

proc export data=WORK.OM\_clean\_data

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

run;

**%mend** OMDataClean;

%***OMDataClean***(schema=sas\_config)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 26 April 2017 ;

\* OMNationalAvg2.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* Re Calulcates national averages post cleaning;

**%macro** OMNationalAvg2(schema=);

proc sql;

create table OM\_nat\_avg2 as

select department,

service,

currency,

sum (activity) as Total\_Activity,

sum(UC\_target\_MFF \* activity) as Total\_cost,

sum( UC\_Target\_MFF \* ACTIVITY) / sum(ACTIVITY) as HRG\_UC\_Target\_MFF

from OM\_clean\_data

group by department, service , currency;

quit;

\*/ Added output of data ;

%let outputfile=OM\_nat\_avg2;

proc export data=WORK.OM\_nat\_avg2

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

run;

**%mend** OMNationalAvg2;

%***OMNationalAvg2***(schema=sas\_config)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 26 April 2017 ;

\* OMMFFAdjust\_1.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

**%macro** OMMFFAdjust(schema=);

proc sql;

create table OM\_MFF\_IMPACT as

select sum (Total\_cost) as Inc\_MFF,

sum (TC\_target\_MFF) as Exec\_MFF,

sum (Total\_Cost) / sum(TC\_Target\_MFF) as NA\_MFF\_ADJ

from OM\_Remove\_MFF;

quit;

\*/ Added output of data ;

%let outputfile=OM\_MFF\_IMPACT;

proc export data=WORK.OM\_MFF\_IMPACT

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

run;

proc sql;

create table OM\_MFF\_Rescale as

select

sum (TC\_target\_MFF) as Exec\_MFF,

sum(TC\_Payment\_MFF) as Exec\_MFF\_capped,

SUM (TC\_Payment\_MFF) / SUM (TC\_Target\_MFF) as MFF\_RESCALE

from OM\_Remove\_MFF;

quit;

\*/ Added output of data ;

%let outputfile=OM\_MFF\_Rescale;

%let outputpath=\\irnarch\sas\_data\PID\Other mandatory\OM\_outputs;

proc export data=WORK.OM\_MFF\_Rescale

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

run;

**%mend** OMMFFAdjust;

%***OMMFFAdjust***(schema=sas\_config)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 26 April 2017 ;

\* OMMFFAdjust\_2.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

**%macro** OMMFFAdjust(schema=);

proc sql;

create table OM\_MFF\_IMPACT as

select sum (Total\_cost) as Inc\_MFF,

sum (TC\_target\_MFF) as Exec\_MFF,

sum (Total\_Cost) / sum(TC\_Target\_MFF) as NA\_MFF\_ADJ

from OM\_Remove\_MFF;

quit;

\*/ Added output of data ;

%let outputfile=OM\_MFF\_IMPACT;

proc export data=WORK.OM\_MFF\_IMPACT

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

run;

proc sql;

create table OM\_MFF\_Rescale as

select

sum (TC\_target\_MFF) as Exec\_MFF,

sum(TC\_Payment\_MFF) as Exec\_MFF\_capped,

SUM (TC\_Payment\_MFF) / SUM (TC\_Target\_MFF) as MFF\_RESCALE

from OM\_Remove\_MFF;

quit;

\*/ Added output of data ;

%let outputfile=OM\_MFF\_Rescale;

proc export data=WORK.OM\_MFF\_Rescale

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

run;

**%mend** OMMFFAdjust;

%***OMMFFAdjust***(schema=sas\_config)

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 26 April 2017 ;

\* OMNATIONALTARIFF.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

**%macro** OMNATIONALTARIFF(schema=);

proc sql;

create table OM\_NATIONAL\_TARIFF\_Direct\_Access as

select

nt.department,

nt.service,

nt.currency,

nt.total\_activity,

nt.total\_cost,

nt.HRG\_UC\_target\_MFF as HRG\_UC,

nt.HRG\_UC\_target\_MFF \* MFF\_rescale as Tariff

from OM\_MFF\_Rescale, OM\_nat\_avg2 as nt

order by nt.department, nt.service, nt.currency;

quit;

\*/ Added output of data ;

%let outputfile=OM\_NATIONAL\_TARIFF\_Direct\_Access;

proc export data=WORK.OM\_NATIONAL\_TARIFF\_Direct\_Access

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

run;

**%mend** OMNATIONALTARIFF;

%***OMNATIONALTARIFF***(schema=sas\_config)

**proc** **sql**; create table t0 as select monotonic() as list, \* from work.Other\_mandatory\_model;

**quit**;

**data** t0\_1920; set t0;

if complexity\_resource\_group="FZ54Z" then complexity\_resource\_group="FE35Z";

else if complexity\_resource\_group="FZ55Z" then complexity\_resource\_group="FE34Z";

else complexity\_resource\_group=complexity\_resource\_group;

**run**;

**Data** yyy; set WORK.OM\_NATIONAL\_TARIFF\_Direct\_Access;

where CURRENCY IN("DZ55Z" "DZ59Z");

**run**;

**Data** yyy2; set WORK.OM\_NATIONAL\_TARIFF\_Direct\_Access/\*yyy\*/;

if CURRENCY="DZ55Z" then currency0="DZ35Z";

ELSE if currency="DZ59Z" then currency0="DZ44Z";

ELSE CURRENCY0=CURRENCY0;

**run**;

**proc** **sql**;

create table a1 as select a.\*, y.currency, y.total\_activity, y.total\_cost, y.hrg\_uc, y.tariff,

x.spell\_hrg, x.p6\_el

from t0\_1920/\*tariff\_r.other\_mandatory\_model\_1617\*/ as a

left join yyy2 as y on a.complexity\_resource\_group=y.currency

left join xxx\_1920 as x on a.complexity\_resource\_group=x.spell\_hrg

order by list/\*complexity\_resource\_group\*/

/\*pod other\_mandatory\_type\*/

;

**quit**;

**data** a2(drop=list currency spell\_hrg p6\_el); set a1;

if p6\_el ne **.** then hrg\_uc=p6\_el; ELSE hrg\_uc=hrg\_uc;

if currency="DZ55Z" then complexity\_resource\_group=currency;

ELSE if currency="DZ59Z" then complexity\_resource\_group=currency;

ELSE COMPLEXITY\_RESOURCE\_GROUP=COMPLEXITY\_RESOURCE\_GROUP;

**run**;

**data** a3; set a2;

if complexity\_resource\_group="FE35Z" then prices=hrg\_uc;

ELSE if complexity\_resource\_group="FE34Z" then prices=hrg\_uc;

ELSE if complexity\_resource\_group="DZ55Z" then prices=hrg\_uc;

ELSE if complexity\_resource\_group="DZ59Z" then prices=hrg\_uc;

else prices=tariff;

IF COMPLEXITY\_RESOURCE\_GROUP NOTIN("FE35Z" "FE34Z") THEN inflation\_efficiency=&ie\_indextn\_adjstmt\_factors.;

else inflation\_efficiency=**.**;

IF COMPLEXITY\_RESOURCE\_GROUP NOTIN("FE35Z" "FE34Z") THEN prices\_after\_IE=prices\*(**1**+&ie\_indextn\_adjstmt\_factors.);

else prices\_after\_IE=prices;

if total\_activity ne **.** then cost\_quantum=total\_activity\*prices\_after\_IE;

else cost\_quantum=**.**;

FORMAT inflation\_efficiency percentn7.2;

format cost\_quantum comma9.2;

**run**;

/\* START HERE - MA PID added - Dec 07 2018 \*/

**data** a300;

set a3;

Year="19/20";

if complexity\_resource\_group="DZ55Z" then prices\_after\_IE2=**51**;

else if complexity\_resource\_group="DZ59Z" then prices\_after\_IE2=**42**;

else prices\_after\_IE2=prices\_after\_IE;

**run**;

**data** a301;

set a300;

if total\_activity ne **.** then cost\_quantum\_post\_ma=total\_activity\*PRICES\_after\_ie2;

else cost\_quantum\_post\_ma=**.**;

**run**;

**proc** **sql**;

select sum(cost\_quantum)/sum(cost\_quantum\_post\_ma) into :rc\_qr2\_factor

from a301

where total\_activity ne **.**;

**quit**;

**DATA** A4

; SET A301;

if total\_activity ne **.** then rc\_qr2\_factor=&rc\_qr2\_factor.;

else rc\_qr2\_factor=**1**;

PRICES\_after\_qr2=prices\_after\_IE2\*rc\_qr2\_factor;

if total\_activity ne **.** then cost\_quantum\_post\_qr2=total\_activity\*PRICES\_after\_qr2;

else cost\_quantum\_post\_qr2=**.**;

**run**;

**data** Other\_Mandatory\_step\_by\_step(drop=om\_id year);

set a4;

if find(complexity\_resource\_group, "FE3","i")=**0** then inflation\_efficiency\_1920=&Infla\_Effi\_final\_year\_uplift.;

else inflation\_efficiency\_1920=**.**;

if find(complexity\_resource\_group, "FE3","i")=**0** then Final\_Prices/\*prices\_after\_adjustment\*/=PRICES\_after\_qr2\*(**1**+&Infla\_Effi\_final\_year\_uplift.);

else Final\_Prices/\*prices\_after\_adjustment\*/=PRICES\_after\_qr2;

format Final\_Prices/\*prices\_after\_adjustment\*/ comma7.0;

format inflation\_efficiency\_1920 percentn7.5;

**run**;

**data** ModelOut.Other\_Mandatory\_step\_by\_step;

set Other\_Mandatory\_step\_by\_step;

**run**;

**data** Other\_Mandatory\_prices\_1920(keep=Year Other\_Mandatory\_type POD Complexity\_Resource\_Group Final\_Prices/\*Prices\_after\_adjustment\*/);

set ModelOut.Other\_Mandatory\_step\_by\_step;

**run**;

**data** ModelOut.Other\_Mandatory\_prices\_1920;

set Other\_Mandatory\_prices\_1920;

**run**;

/\*Table 1 OM\_Direct\_Access \*/

**data** OM\_Direct\_Access\_1;

set Other\_Mandatory\_prices\_1920;

where POD="APC" and Complexity\_Resource\_Group in ("FE35Z","FE34Z","DZ55Z","DZ59Z");

rename Complexity\_Resource\_Group=HRG\_Code;

if Complexity\_Resource\_Group in ("FE35Z","FE34Z") then Type="Flexible Sigmoidoscopy";

if Complexity\_Resource\_Group in ("DZ55Z","DZ59Z") then Type="Airflow Studies";

rename Other\_Mandatory\_Type=HRG\_Name;

**run**;

**proc** **sql**;

create table ModelOut.OM\_Direct\_Access

as

select

a.Type,

a.HRG\_Code,

a.HRG\_Name,

a.Final\_Prices as Tariff

from OM\_Direct\_Access\_1 as a

;

**quit**;

/\*Table 2 Rehabilitation post discharge \*/

**Data** ModelOut.OM\_Rehab\_Post\_Discharge(drop=Other\_Mandatory\_Type POD);

set Other\_Mandatory\_prices\_1920;

where Other\_Mandatory\_Type="Rehabilitation Post Discharge";

rename

Complexity\_Resource\_Group=Description

Final\_Prices = Tariff

;

**Run**;

/\*Table 3 Cystic fibrosis \*/

**Data** ModelOut.OM\_Cystic\_fibrosis(drop= POD Other\_Mandatory\_Type);

set Other\_Mandatory\_prices\_1920;

where Other\_Mandatory\_Type="Cystic Fibrosis";

rename

Complexity\_Resource\_Group=Band

Final\_Prices = Tariff

;

**run**;

/\*Table 4 Looked after children's health assessments \*/

**Data** ModelOut.OM\_ChildreHealthAssessment(drop= POD Other\_Mandatory\_Type);

set Other\_Mandatory\_prices\_1920;

where Other\_Mandatory\_Type="Children's health assessments";

rename

Complexity\_Resource\_Group=Task

Final\_Prices = Tariff

;

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 05 March 2017 ;

\* OPATT\_1\_10\_Clean\_Data.sas ;

\* summary: Compares provider UC to nat av and for those >20x or <1/20th of ;

\* the nat av, sets data to 0 ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

%let outputfile=Stage\_OPATT\_CleanData;

**proc** **sql**; create table Stage\_OPATT\_CleanData as

SELECT

in\_02a.Org\_Code,

in\_02a.Dept\_code,

in\_02a.Currency\_code,

in\_02a.Tariff\_TFC\_code,

in\_02a.Activity,

in\_02B.Average,

in\_02a.MFF\_Adj\_UC,

CASE WHEN ((&param\_OutlierMinValue=**0** OR &param\_OutlierMinValue IS NULL) AND (&param\_OutlierMaxValue =**0** OR &param\_OutlierMaxValue IS NULL)) THEN Activity

ELSE

CASE WHEN MFF\_adj\_uc/Average<&param\_OutlierMinValue Or MFF\_adj\_uc/Average>&param\_OutlierMaxValue THEN **0**

ELSE Activity END

END/\*--Replaced value 0.05 with @param\_OutlierMinValue AND value 20 with @param\_OutlierMaxValue - 18/12/2014\*/as clean\_activity,

CASE WHEN ((&param\_OutlierMinValue=**0** OR &param\_OutlierMinValue IS NULL) AND (&param\_OutlierMaxValue =**0** OR &param\_OutlierMaxValue IS NULL)) THEN MFF\_adj\_tc

ELSE

CASE WHEN MFF\_adj\_uc/Average<&param\_OutlierMinValue Or MFF\_adj\_uc/Average>&param\_OutlierMaxValue THEN **0**

ELSE MFF\_adj\_tc END

END/\*--Replaced value 0.05 with @param\_OutlierMinValue AND value 20 with @param\_OutlierMaxValue - 18/12/2014 \*/as clean\_tc

FROM

Stage\_OPATT01\_02a in\_02a

INNER JOIN Stage\_OPATT01\_02B in\_02B

ON (in\_02a.Currency\_code = in\_02B.Currency\_code)

AND (in\_02a.Dept\_code = in\_02B.Dept\_code)

AND (in\_02a.Tariff\_TFC\_code = in\_02B.Tariff\_TFC\_code)

ORDER BY

in\_02a.Org\_Code, in\_02a.Tariff\_TFC\_code;

**quit**;

**proc** **export** data=&outputfile.

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 05 March 2017 ;

\* OPATT\_xxx\_xxx\_xxx.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

**proc** **sql**; create table Valid\_OPATT01\_03a as

SELECT

Org\_Code,

SUM(clean\_activity) AS clean\_activity,

SUM(clean\_tc) AS clean\_tc

FROM Stage\_OPATT\_CleanData

GROUP BY Org\_Code;

**quit**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 05 March 2017 ;

\* OPATT\_1\_12\_Clean\_Data\_Crosstab.sas ;

\* summary: ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

%let outputfile=Stage\_OPATT01\_03B;

**proc** **sort** data=Stage\_OPATT\_CleanData out=Stage\_OPATT\_CleanDataq;

by org\_code dept\_code tariff\_tfc\_code;

**run**;

**PROC** **TRANSPOSE** data=Stage\_OPATT\_CleanDataq out=Stage\_OPATT01\_03B (drop=\_name\_);

by org\_code dept\_code tariff\_tfc\_code;

id Currency\_code;

var clean\_activity;

**RUN**;

**proc** **export** data=&outputfile.

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 09 March 2017 ;

\* OPATT\_1\_13\_Recoding\_of\_501\_560.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

%let outputfile=Stage\_OPATT01\_03C;

**Proc** **sql**; create table Stage\_OPATT01\_03C (compress=yes) as

SELECT

Org\_Code,

Currency\_code,

Dept\_code,

tariff\_tfc\_code,

CASE WHEN tariff\_tfc\_code="501" THEN "CL" ELSE

(CASE WHEN tariff\_tfc\_code = "560" THEN "CL" ELSE Dept\_Code END) ENd as Recoding\_of\_501\_560,

clean\_activity,

clean\_tc

FROM Stage\_OPATT\_CleanData

ORDER BY Org\_Code, tariff\_tfc\_code;

**quit**;

**proc** **export** data=&outputfile.

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 09 March 2017 ;

\* OPATT\_1\_14\_Providers\_per\_TFC.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

%let outputfile=stage\_OPATT\_Providers\_TFC;

**proc** **sql**; create table stage\_OPATT\_Providers\_TFC as

SELECT

tariff\_tfc\_code,

Currency\_code,

Count(Org\_Code) AS CountOfOrg\_Code

FROM STAGE\_OPATT01\_03C

WHERE Recoding\_of\_501\_560="CL"

GROUP BY tariff\_tfc\_code, Currency\_code;

**quit**;

**proc** **export** data=&outputfile.

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 09 March 2017 ;

\* OPATT\_1\_15\_317\_by\_org.sas ;

\* summary: Used to determine impact of excluding organisations ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

%let outputfile=Stage\_OPATT01\_03ciii;

**Proc** **sql**; create table Stage\_OPATT01\_03ciii as

SELECT

Org\_Code,

Currency\_code,

tariff\_tfc\_code,

Recoding\_of\_501\_560,

clean\_activity,

clean\_tc

FROM Stage\_OPATT01\_03c

WHERE tariff\_tfc\_code="317" AND Recoding\_of\_501\_560="CL";

**QUIT**;

**proc** **export** data=&outputfile.

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 09 March 2017 ;

\* OPATT\_1\_16\_UC\_by\_org\_TFC\_Currency.sas ;

\* summary: Unit costs by org, TFC, Currency ;

\* For calculating the IM for the new 14 TFCs ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

%let outputfile=stage\_OPATT\_Clean\_WO\_Provider;

**proc** **sql**; Create table stage\_OPATT\_Clean\_WO\_Provider as

SELECT

Currency\_code,

tariff\_tfc\_code,

Recoding\_of\_501\_560 AS Dept\_Code,

Sum(clean\_activity) AS Act,

Sum(clean\_tc) AS TC

FROM Stage\_OPATT01\_03c

GROUP BY

Currency\_code,

tariff\_tfc\_code,

Recoding\_of\_501\_560

HAVING

tariff\_tfc\_code<>'BMT'

And tariff\_tfc\_code<>'CMD'

And tariff\_tfc\_code<>'CSB'

And tariff\_tfc\_code<>'FPC'

And tariff\_tfc\_code<> 'DAP'

And tariff\_tfc\_code<>'H/A'

And tariff\_tfc\_code<>'TCM'

AND tariff\_tfc\_code<>'999'

ORDER BY Currency\_code, tariff\_tfc\_code;

**quit**;

**proc** **export** data=&outputfile.

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 09 March 2017 ;

\* OPATT\_ 01\_03d.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

**Proc** **Sql**; create table VALID\_OPATT01\_03d as

SELECT

SUM(ACT) AS ACT,

SUM(TC) as TC

FROM stage\_OPATT\_Clean\_WO\_Provider;

**quit**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 09 March 2017 ;

\* OPATT\_1\_18\_Data\_for\_IM.sas ;

\* summary: Group recoded and cleaned data to provide activity for impact modelling ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

%let outputfile=Stage\_OPATT\_Provider;

**Proc** **sql**; create table Stage\_OPATT\_Provider as

SELECT

Org\_Code,

Currency\_code,

tariff\_tfc\_code,

Recoding\_of\_501\_560 AS Dept\_Code,

Sum(clean\_activity) AS SumOfclean\_activity

FROM Stage\_OPATT01\_03c

GROUP BY Org\_Code, Currency\_code, tariff\_tfc\_code, Recoding\_of\_501\_560

ORDER BY Currency\_code, tariff\_tfc\_code

;

**quit**;

**proc** **export** data=&outputfile.

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 09 March 2017 ;

\* OPATT\_1\_19\_Remapping\_of\_Costs.sas ;

\* summary: Allocating costs to relevant attendance types ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

/\*--01c\_Data\_OP\_2 upto & incl D&D excl\*/

/\*\*/

/\*--01a: Remapping of Costs\*/

/\*--Allocating costs to relevant attendance types\*/

%let outputfile=Stage\_OPATT02\_01a;

**Proc** **sql**; create table Stage\_OPATT02\_01a as

SELECT

trim(Dept\_Code)||trim(Mapping) AS Combined,

in\_13a.tariff\_tfc\_code,

in\_13a.Act,

in\_13a.TC

FROM

work.OPATT\_Curr\_Map as in\_04

INNER JOIN Stage\_OPATT\_Clean\_WO\_Provider as in\_13a

ON in\_04.Currency\_Code = in\_13a.Currency\_code

ORDER BY

trim(Dept\_Code)||trim(Mapping),

in\_13a.tariff\_tfc\_code;

**quit**;

**proc** **export** data=&outputfile.

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 05 May 2017 ;

\* OPATT\_1\_1\_filter\_for\_scope.sas ;

\* summary: Remove PMS+ and non-NHS data, and NF2F attendances ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

%let outputfile=stage\_OPATT\_ScopeData;

**proc** **sql**; create table stage\_OPATT\_ScopeData as

SELECT

in\_01b.organisationcode AS Org\_Code,

in\_01b.SUPPLIERTYPE AS Supplier\_type\_code,

in\_01b.DEPARTMENTcode AS Dept\_code,

in\_01b.SERVICEcode AS Service\_code,

in\_01b.CURRENCYcode AS Currency\_code,

in\_01b.UNITCOST AS Unit\_cost,

in\_01b.ACTIVITY AS Activity\_1

FROM

work.Reference\_Cost as in\_01b

INNER JOIN work.tc\_mff as in\_00

ON in\_01b.organisationcode=

in\_00.RC\_Code

WHERE

in\_01b.SUPPLIERTYPE='OWN' and /\* --Removed PH 11/02/2015 -- reset SPK 27/11/2015 \*/

(in\_01b.DEPARTMENTcode='CL'

Or in\_01b.DEPARTMENTcode='NCL')

AND (in\_01b.CURRENCYcode Not Like '%C'

And in\_01b.CURRENCYcode Not Like '%D')

ORDER BY in\_01b.organisationcode, in\_01b.SERVICEcode;

**quit**;

**proc** **export** data=&outputfile.

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 05 May 2017 ;

\* OPATT\_01\_01a.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

**proc** **sql**; create table valid\_OPATT01\_01a as

SELECT

Org\_Code,

AVG(UNIT\_COST) AS Unit\_Cost,

SUM(Activity\_1) AS Activity

FROM stage\_OPATT\_ScopeData

GROUP BY Org\_Code;

**quit**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 09 March 2017 ;

\* OPATT\_02\_01a.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

**Proc** **Sql**; create table valid\_OPATT02\_01a as

SELECT

tariff\_tfc\_code,

SUM(ACT) AS ACT,

SUM(TC) AS TC

FROM Stage\_OPATT02\_01a

GROUP BY tariff\_tfc\_code;

**quit**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 09 March 2017 ;

\* OPATT\_1\_21\_Crosstab\_of\_Costs.sas ;

\* summary: Costs by Attendance type ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

%let outputfile=Stage\_OPATT02\_01B;

**proc** **sort** data=Stage\_OPATT02\_01a out=Stage\_OPATT02\_01aq;

by tariff\_tfc\_code;

**run**;

**PROC** **TRANSPOSE** data=Stage\_OPATT02\_01aq out=Stage\_OPATT02\_01Bq;

by tariff\_tfc\_code;

id Combined;

var TC;

**RUN**;

**proc** **sql**; create table Stage\_OPATT02\_01B as

SELECT

tariff\_tfc\_code,

coalesce(CL\_FAM,**0**) + coalesce(CL\_FAS,**0**) + coalesce(CL\_FUM,**0**) + coalesce(CL\_FUS,**0**) + coalesce(NCL\_FAM,**0**)

+ coalesce(NCL\_FAS,**0**) + coalesce(NCL\_FUM,**0**) + coalesce(NCL\_FUS,**0**) as Total\_Costs,

coalesce(CL\_FAM,**0**) as CL\_FAM,

coalesce(CL\_FAS,**0**) as CL\_FAS,

coalesce(CL\_FUM,**0**) as CL\_FUM,

coalesce(CL\_FUS,**0**) as CL\_FUS,

coalesce(NCL\_FAM,**0**) as NCL\_FAM,

coalesce(NCL\_FAS,**0**) as NCL\_FAS,

coalesce(NCL\_FUM,**0**) as NCL\_FUM,

coalesce(NCL\_FUS,**0**) as NCL\_FUS

FROM Stage\_OPATT02\_01Bq;

**quit**;

**proc** **export** data=&outputfile.

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 09 March 2017 ;

\* OPATT\_1\_22\_Make\_table\_of\_Costs\_Crosstab.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

%let outputfile=stage\_OPATT\_Cost\_Crosstab;

**Proc** **sql**; create table stage\_OPATT\_Cost\_Crosstab as

SELECT \*

FROM Stage\_OPATT02\_01B;

**quit**;

**proc** **export** data=&outputfile.

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 09 March 2017 ;

# \* OPATT\_1\_23\_Crosstab\_Activity.sas ;

\* summary: Activity by attendance type ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

%let outputfile=Stage\_OPATT02\_01D;

**proc** **sort** data=Stage\_OPATT02\_01a out=Stage\_OPATT02\_01aq;

by tariff\_tfc\_code;

**run**;

**PROC** **TRANSPOSE** data=Stage\_OPATT02\_01aq out=Stage\_OPATT02\_01Dq;

by tariff\_tfc\_code;

id Combined;

var ACT;

**RUN**;

**PROC** **SQL**; CREATE TABLE Stage\_OPATT02\_01D AS

SELECT

tariff\_tfc\_code,

coalesce(CL\_FAM,**0**) + coalesce(CL\_FAS,**0**) + coalesce(CL\_FUM,**0**) + coalesce(CL\_FUS,**0**) + coalesce(NCL\_FAM,**0**)

+ coalesce(NCL\_FAS,**0**) + coalesce(NCL\_FUM,**0**) + coalesce(NCL\_FUS,**0**) as Total\_Activity,

coalesce(CL\_FAM,**0**) as CL\_FAM,

coalesce(CL\_FAS,**0**) as CL\_FAS,

coalesce(CL\_FUM,**0**) as CL\_FUM,

coalesce(CL\_FUS,**0**) as CL\_FUS,

coalesce(NCL\_FAM,**0**) as NCL\_FAM,

coalesce(NCL\_FAS,**0**) as NCL\_FAS,

coalesce(NCL\_FUM,**0**) as NCL\_FUM,

coalesce(NCL\_FUS,**0**) as NCL\_FUS

FROM Stage\_OPATT02\_01Dq;

**QUIT**;

**proc** **export** data=&outputfile.

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 09 March 2017 ;

# \* OPATT\_1\_24\_Make\_table\_of\_Activity\_Crosstab.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

%let outputfile=stage\_OPATT\_Activity\_Crosstab;

**proc** **sql**; create table stage\_OPATT\_Activity\_Crosstab as

SELECT \*

FROM Stage\_OPATT02\_01D;

**quit**;

**proc** **sql**; create table valid\_OPATT02\_01e as

SELECT \*

FROM stage\_OPATT\_Activity\_Crosstab;

**quit**;

**proc** **export** data=&outputfile.

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 09 March 2017 ;

# \* OPATT\_1\_25\_D\_D\_Exclusions.sas ;

\* summary: Total Drug & Device Exclusions & relevant TFC ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

%let outputfile=Stage\_OPATT02\_02a;

**Proc** **sql**; Create table CTE\_A as

SELECT TFC,SUM(COALESCE(DandD\_Total,**0**)) AS DandD\_Total

FROM work.OPATT\_Drugs\_Devices

GROUP BY TFC;

**Quit**;

**proc** **sql**; create table Stage\_OPATT02\_02a as

SELECT TFC,COALESCE(DandD\_Total,**0**)/National\_Average AS Total\_D\_D

FROM CTE\_A as in\_05a,

Stage\_OPATT\_Nat\_Avg\_MFF as in\_12a

WHERE (COALESCE(DandD\_Total,**0**)) /National\_Average<>**0**;

**quit**;

**proc** **export** data=&outputfile.

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 09 March 2017 ;

# \* OPATT\_02\_02a.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

**proc** **sql**; create table valid\_OPATT02\_02a as

SELECT

TFC,

SUM(TOTAL\_d\_d) AS Total\_D\_D

FROM Stage\_OPATT02\_02a

GROUP BY TFC;

**quit**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 09 March 2017 ;

# \* OPATT\_1\_27\_Total\_costs\_and\_pcts\_for\_apportionment\_of\_D\_D.sas ;

\* summary: Total costs and %'s for apportionment of D&D ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

%let outputfile=Stage\_OPATT02\_02b;

**proc** **sql**; create table Stage\_OPATT02\_02b as

SELECT B.\*,

CL\_FAM\_App+CL\_FAS\_App+CL\_FUM\_App+CL\_FUS\_App AS TOTAL\_App

FROM

(

SELECT a.\*,

case when Total\_CL\_Costs = **0** then **0** else CL\_FAM/Total\_CL\_Costs end as CL\_FAM\_App,

case when Total\_CL\_Costs = **0** then **0** else CL\_FAS/Total\_CL\_Costs end as CL\_FAS\_App,

case when Total\_CL\_Costs = **0** then **0** else CL\_FUM/Total\_CL\_Costs end as CL\_FUM\_App,

case when Total\_CL\_Costs = **0** then **0** else CL\_FUS/Total\_CL\_Costs end as CL\_FUS\_App

FROM (

SELECT tariff\_tfc\_code,

Total\_Costs,

coalesce(CL\_FAM,**0**)+ coalesce(CL\_FAS,**0**)+coalesce(CL\_FUM,**0**)+coalesce(CL\_FUS,**0**) AS Total\_CL\_Costs,

coalesce(CL\_FAM,**0**) as CL\_FAM, coalesce(CL\_FAS,**0**) as CL\_FAS, coalesce(CL\_FUM,**0**) as CL\_FUM, coalesce(CL\_FUS,**0**) as CL\_FUS

FROM stage\_OPATT\_Cost\_Crosstab

) AS A

) AS b

;

**quit**;

**proc** **export** data=&outputfile.

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 09 March 2017 ;

# \* OPATT\_1\_28\_D\_D\_costs\_to\_remove.sas ;

\* summary: Apply exclusions to apportionment ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

%let outputfile=Stage\_OPATT02\_02C;

**Proc** **sql**; create table Stage\_OPATT02\_02C as

SELECT

out\_01a.tariff\_tfc\_code,

out\_01a.Total\_Costs,

in\_02a.Total\_D\_D,

COALESCE(Total\_D\_D,**0**)\*COALESCE(CL\_FAM\_App,**0**) AS From\_CL\_FAM,

COALESCE(Total\_D\_D,**0**)\*COALESCE(CL\_FAS\_App,**0**) AS From\_CL\_FAS,

COALESCE(Total\_D\_D,**0**)\*COALESCE(CL\_FUM\_App,**0**) AS From\_CL\_FUM,

COALESCE(Total\_D\_D,**0**)\*COALESCE(CL\_FUS\_App,**0**) AS From\_CL\_FUS

FROM

stage\_OPATT\_Cost\_Crosstab as out\_01a

LEFT JOIN Stage\_OPATT02\_02a in\_02a

ON out\_01a.tariff\_tfc\_code = in\_02a.TFC

LEFT JOIN Stage\_OPATT02\_02b in\_02b ON out\_01a.tariff\_tfc\_code = in\_02b.tariff\_tfc\_code;

**quit**;

**proc** **export** data=&outputfile.

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 09 March 2017 ;

# \* OPATT\_1\_29\_Revised\_Costs\_following\_removal\_of\_D\_D.sas ;

\* summary: Cost following removal of drugs and devices ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

%let outputfile=Stage\_OPATT02\_02D;

**Proc** **sql**; create table Stage\_OPATT02\_02D as

SELECT

A.\*,

COALESCE(CL\_FAM\_Rev\_Costs,**0**)+COALESCE(CL\_FAS\_Rev\_Costs,**0**)+COALESCE(CL\_FUM\_Rev\_Costs,**0**)+COALESCE(CL\_FUS\_Rev\_Costs,**0**)+COALESCE(NCL\_FAM\_Rev\_Costs,**0**)+COALESCE(NCL\_FAS\_Rev\_Costs,**0**)+COALESCE(NCL\_FUM\_Rev\_Costs,**0**)+COALESCE(NCL\_FUS\_Rev\_Costs,**0**) AS Total\_Revised\_Costs

FROM (

SELECT

in\_02C.tariff\_tfc\_code,

COALESCE(CL\_FAM,**0**)-COALESCE(From\_CL\_FAM,**0**) AS CL\_FAM\_Rev\_Costs,

COALESCE(CL\_FAS,**0**)-COALESCE(From\_CL\_FAS,**0**) AS CL\_FAS\_Rev\_Costs,

COALESCE(CL\_FUM,**0**)-COALESCE(From\_CL\_FUM,**0**) AS CL\_FUM\_Rev\_Costs,

COALESCE(CL\_FUS,**0**)-COALESCE(From\_CL\_FUS,**0**) AS CL\_FUS\_Rev\_Costs,

out\_01a.NCL\_FAM AS NCL\_FAM\_Rev\_Costs,

out\_01a.NCL\_FAS AS NCL\_FAS\_Rev\_Costs,

out\_01a.NCL\_FUM AS NCL\_FUM\_Rev\_Costs,

out\_01a.NCL\_FUS AS NCL\_FUS\_Rev\_Costs

FROM

stage\_OPATT\_Cost\_Crosstab as out\_01a

INNER JOIN Stage\_OPATT02\_02C in\_02C

ON out\_01a.tariff\_tfc\_code = in\_02C.tariff\_tfc\_code

) AS A;

**quit**;

**proc** **export** data=&outputfile.

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 09 March 2017 ;

# \* OPATT\_1\_30\_D\_D\_Limiting\_of\_exclusions\_inc\_additional\_topslice.sas ;

\* summary: Check that D&D exclusions dont exceed 50% ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

%let outputfile=Stage\_OPATT02\_02E;

**Proc** **sql**; create table Stage\_OPATT02\_02E as

SELECT

A.\*,

COALESCE(CL\_FAM\_Limit,**0**)

+COALESCE(CL\_FAS\_Limit,**0**)

+COALESCE(CL\_FUM\_Limit,**0**)

+COALESCE(CL\_FUS\_Limit,**0**) AS Total\_CL\_Revised\_Costs,

(COALESCE(CL\_FAM\_Rev\_Costs,**0**)

+COALESCE(CL\_FAS\_Rev\_Costs,**0**)

+COALESCE(CL\_FUM\_Rev\_Costs,**0**)

+COALESCE(CL\_FUS\_Rev\_Costs,**0**)) -

(COALESCE(CL\_FAM\_Limit,**0**)+COALESCE(CL\_FAS\_Limit,**0**)+COALESCE(CL\_FUM\_Limit,**0**)+COALESCE(CL\_FUS\_Limit,**0**))

AS Limiting\_top\_slice

FROM (

SELECT

in\_02D.tariff\_tfc\_code,

in\_02D.CL\_FAM\_Rev\_Costs

,in\_02D.CL\_FAS\_Rev\_Costs

,in\_02D.CL\_FUM\_Rev\_Costs

,in\_02D.CL\_FUS\_Rev\_Costs,

CASE WHEN CL\_FAM = **0** OR CL\_FAM IS NULL THEN **0**

ELSE CASE WHEN COALESCE(CL\_FAM\_Rev\_Costs,**0**) / COALESCE(CL\_FAM,**0**) < **0.5**

THEN **0.5**\*COALESCE(CL\_FAM,**0**)

ELSE COALESCE(CL\_FAM\_Rev\_Costs,**0**) END

END as CL\_FAM\_Limit,

CASE WHEN CL\_FAS = **0** OR CL\_FAS IS NULL THEN **0**

ELSE CASE WHEN COALESCE(CL\_FAS\_Rev\_Costs,**0**) / COALESCE(CL\_FAS,**0**) < **0.5**

THEN **0.5**\*COALESCE(CL\_FAS,**0**)

ELSE COALESCE(CL\_FAS\_Rev\_Costs,**0**) END

END as CL\_FAS\_Limit,

CASE WHEN CL\_FUM = **0** OR CL\_FUM IS NULL THEN **0**

ELSE CASE WHEN COALESCE(CL\_FUM\_Rev\_Costs,**0**) / COALESCE(CL\_FUM,**0**) < **0.5**

THEN **0.5**\*COALESCE(CL\_FUM,**0**)

ELSE COALESCE(CL\_FUM\_Rev\_Costs,**0**) END

END as CL\_FUM\_Limit,

CASE WHEN CL\_FUS = **0** OR CL\_FUS IS NULL THEN **0**

ELSE CASE WHEN COALESCE(CL\_FUS\_Rev\_Costs,**0**) / COALESCE(CL\_FUS,**0**) < **0.5**

THEN **0.5**\*COALESCE(CL\_FUS,**0**)

ELSE COALESCE(CL\_FUS\_Rev\_Costs,**0**) END

END as CL\_FUS\_Limit

FROM

Stage\_OPATT02\_02D in\_02D

INNER JOIN stage\_OPATT\_Cost\_Crosstab as out\_01a

ON in\_02D.tariff\_tfc\_code = out\_01a.tariff\_tfc\_code ) AS A;

**quit**;

**proc** **export** data=&outputfile.

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 09 March 2017 ;

# \* OPATT\_1\_31\_Total\_Topslice\_to\_be\_removed.sas ;

\* summary: Amounts to be top-sliced ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

%let outputfile=Stage\_OPATT02\_02f;

**Proc** **Sql**; Create table Stage\_OPATT02\_02f as

SELECT

Sum(in\_02e.Total\_CL\_Revised\_Costs) AS SumOfTotal\_CL\_Revised\_Costs,

Sum(abs(in\_02e.Limiting\_top\_slice)) AS Limit\_TS,

in\_05b.Top\_Slice AS Additional\_TS,

in\_12a.National\_Average,

Sum(abs(in\_02e.Limiting\_top\_slice)) +(in\_05b.Top\_Slice/National\_Average) AS Top\_slice

FROM

work.OPATT\_DD\_Topslice AS in\_05b,

Stage\_OPATT\_Nat\_Avg\_MFF as in\_12a,

Stage\_OPATT02\_02D in\_02D INNER JOIN Stage\_OPATT02\_02E in\_02e ON in\_02D.tariff\_tfc\_code = in\_02e.tariff\_tfc\_code

GROUP BY in\_05b.Top\_Slice, in\_12a.National\_Average;

**QUIT**;

**proc** **export** data=&outputfile.

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 09 March 2017 ;

# \* OPATT\_1\_32\_pct\_Topslice.sas ;

\* summary: Top-slice as a % of overall costs ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

%let outputfile=Stage\_OPATT02\_02g;

**Proc** **sql**; create table Stage\_OPATT02\_02g as

SELECT (Top\_slice)/(SumofTotal\_CL\_Revised\_Costs) AS Pct\_Top\_slice

FROM Stage\_OPATT02\_02f;

**quit**;

**proc** **export** data=&outputfile.

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 09 March 2017 ;

# \* OPATT\_1\_33\_Revised\_Costs\_after\_topslice.sas ;

\* summary: Revised costs following removal of D&D and top-slice ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

%let outputfile=output\_OPATT\_FinalTariff;

**Proc** **Sql**; Create table output\_OPATT\_FinalTariff as

SELECT

A.\*,

COALESCE(CL\_FAM,**0**)+COALESCE(CL\_FAS,**0**)+COALESCE(CL\_FUM,**0**)+COALESCE(CL\_FUS,**0**) AS Total\_Costs

FROM (

SELECT

in\_02E.tariff\_tfc\_code,

COALESCE(CL\_FAM\_Limit,**0**)-(COALESCE(CL\_FAM\_Limit,**0**)\*(Pct\_Top\_Slice)) AS CL\_FAM,

COALESCE(CL\_FAS\_Limit,**0**)-(COALESCE(CL\_FAS\_Limit,**0**)\*(Pct\_Top\_Slice)) AS CL\_FAS,

COALESCE(CL\_FUM\_Limit,**0**)-(COALESCE(CL\_FUM\_Limit,**0**)\*(Pct\_Top\_Slice)) AS CL\_FUM,

COALESCE(CL\_FUS\_Limit,**0**)-(COALESCE(CL\_FUS\_Limit,**0**)\*(Pct\_Top\_Slice)) AS CL\_FUS

FROM

Stage\_OPATT02\_02E in\_02E, Stage\_OPATT02\_02g ) AS A;

**quit**;

**proc** **export** data=&outputfile.

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 05 March 2017 ;

# \* OPATT\_1\_3\_Total\_Costs.sas ;

\* summary: Total costs associated with activity within scope ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

%let outputfile=stage\_OPATT01\_01B;

**proc** **sql**; create table stage\_OPATT01\_01B as

SELECT

Org\_Code,

Dept\_code,

Currency\_code,

Service\_code,

Unit\_cost,

Activity\_1,

(Unit\_Cost\*Activity\_1) AS Total\_Costs

FROM stage\_OPATT\_ScopeData

ORDER BY

Org\_Code, Service\_code;

**quit**;

**proc** **export** data=&outputfile.

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 05 March 2017 ;

# \* OPATT\_1\_4\_Regroup.sas ;

\* summary: Regroup to 3 character TFC codes ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

%let outputfile=stage\_OPATT01\_01C;

**DATA** OPATT\_MAPPING;

SET work.OPATT\_MAPPING;

TFC2=PUT(TARIFF\_TFC, **3.**);

DROP TARIFF\_TFC;

RENAME TFC2=TARIFF\_TFC;

**RUN**;

**proc** **sql**; create table stage\_OPATT01\_01C as

SELECT

in\_01B.Org\_Code,

in\_01B.Dept\_code,

in\_01B.Currency\_code,

in\_03.Tariff\_TFC AS Tariff\_TFC\_code,

Sum(in\_01B.Activity\_1) AS Activity,

Sum(in\_01B.Total\_Costs) AS SumOfTotal\_Costs

FROM

WORK.OPATT\_MAPPING as in\_03

INNER JOIN stage\_OPATT01\_01B in\_01B

ON in\_03.Tariff\_TFC = in\_01B.Service\_code

GROUP BY

in\_01B.Org\_Code,

in\_01B.Dept\_code,

in\_01B.Currency\_code,

in\_03.Tariff\_TFC

ORDER BY

in\_01B.Org\_Code,

in\_01B.Currency\_code,

in\_03.Tariff\_TFC;

**quit**;

**proc** **export** data=&outputfile.

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 05 March 2017 ;

# \* OPATT\_1\_5\_Remove\_MFF.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

%let outputfile=stage\_OPATT01\_02a;

**proc** **sql**; create table stage\_OPATT01\_02a as

SELECT

in\_01C.Org\_Code,

in\_01C.Dept\_code,

in\_01C.Currency\_code,

in\_01C.Tariff\_TFC\_code,

in\_01C.Activity,

(Sumoftotal\_costs/Activity) AS Unit\_Cost,

in\_01C.SumOfTotal\_Costs AS Total\_Costs,

((Sumoftotal\_costs/Activity)/uncapped\_MFF) AS MFF\_Adj\_UC,

(SumofTotal\_Costs/uncapped\_MFF) AS MFF\_Adj\_TC,

(SumofTotal\_Costs/Capped\_MFF) AS MFF\_Adj\_TC\_Capped

FROM stage\_OPATT01\_01C in\_01C

INNER JOIN work.tc\_mff as in\_00

ON in\_01C.Org\_Code = in\_00.RC\_Code

ORDER BY

in\_01C.Org\_Code,

in\_01C.Currency\_code,

in\_01C.Tariff\_TFC\_code;

**quit**;

**proc** **export** data=&outputfile.

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 05 March 2017 ;

# \* OPATT\_01\_02a.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

**proc** **sql**; create table valid\_OPATT01\_02a as

SELECT

Org\_Code,

SUM(Activity) as Activity,

AVG(Unit\_Cost) as unit\_cost,

AVG(Total\_Costs) as total\_costs,

SUM(mff\_adj\_uc) as mff\_adj\_uc,

sum(mff\_adj\_tc) as mff\_adj\_tc,

sum(MFF\_Adj\_TC\_Capped) as mff\_adj\_tc\_capped

FROM stage\_OPATT01\_02a

GROUP BY Org\_Code;

**quit**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 05 March 2017 ;

# \* OPATT\_1\_7\_National\_Averages.sas ;

\* summary: Calculates national averages for use in 07: Clean Data ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

%let outputfile=stage\_OPATT01\_02B;

**proc** **sql**; create table stage\_OPATT01\_02B as

SELECT

Dept\_code,

Currency\_code,

Tariff\_TFC\_code,

Sum(Activity) AS SumOfActivity,

Sum(MFF\_Adj\_TC) AS SumOfMFF\_Adj\_TC,

Sum(MFF\_Adj\_TC)/Sum(Activity) AS Average

FROM stage\_OPATT01\_02a

GROUP BY

Dept\_code,

Currency\_code,

Tariff\_TFC\_code

ORDER BY

Currency\_code,

Tariff\_TFC\_code;

**quit**;

**proc** **export** data=&outputfile.

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 05 March 2017 ;

# \* OPATT\_1\_8\_National\_Average\_MFF.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

%let outputfile=stage\_OPATT\_Nat\_Avg\_MFF;

**proc** **sql**; create table stage\_OPATT\_Nat\_Avg\_MFF as

SELECT

Sum(Total\_Costs) AS Inc\_MFF,

Sum(MFF\_Adj\_TC) AS Exc\_MFF,

(Sum(Total\_Costs)/Sum(MFF\_Adj\_TC)) AS National\_Average

FROM Stage\_OPATT01\_02a;

**quit**;

**proc** **export** data=&outputfile.

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: 05 March 2017 ;

# \* OPATT\_1\_9\_MFF\_Rescaling.sas ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

%let outputfile=Stage\_OPATT\_MFF\_Rescaling;

**proc** **sql**; create table Stage\_OPATT\_MFF\_Rescaling as

SELECT

Sum(MFF\_Adj\_TC) AS Uncapped\_TC,

Sum(MFF\_Adj\_TC\_Capped) AS Capped\_TC,

(Sum(MFF\_Adj\_TC\_Capped)/Sum(MFF\_Adj\_TC)) AS MFF\_Rebasing

FROM

Stage\_OPATT01\_02a;

**quit**;

**proc** **export** data=&outputfile.

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

**data** new\_t2c\_2\_3;

set front\_loaded;

TFC2=input(TFC, **8.**);

drop TFC;

rename TFC2=TFC;

**run**;

**proc** **sql**; create table newer\_t3 as select tfc, \* from new\_t2c\_2\_3; **quit**;

**PROC** **SQL**;

CREATE TABLE newer\_t3d AS SELECT A.TFC, B.TARIFF\_TFC, B.PAeD\_PAIRINGS, a.\* from newer\_t3 a right join

further\_adjustment\_adultpaed b

on a.tfc=b.tfc;

**quit**;

**DATA** newer\_t3E;

SET newer\_t3d;

if FIND(TARIFF\_TFC, "PAEDIA", "i") then adult\_child="C";

ELSE adult\_child="A";

if paed\_pairings="" then paed\_pairings="999"; else paed\_pairings=paed\_pairings;

if tariff\_tfc="Paediatric Gastrointestinal Surgery" and tfc=**.** then tfc=**213**;

else if tariff\_tfc="Global Trust Codes" and tfc=**.** then tfc=**999**;

else tfc=tfc;

**run**;

**PROC** **SQL**;

CREATE TABLE newer\_t3F

AS SELECT TFC, TARIFF\_TFC, PAED\_PAIRINGS, adult\_child, cl\_fas\_uc, cl\_fam\_uc, cl\_fus\_uc, cl\_fum\_uc, cl\_fas\_unitcost, cl\_fam\_unitcost, cl\_fus\_unitcost,

cl\_fum\_unitcost, cl\_fas\_act\_postA, cl\_fam\_act\_postA,cl\_fUs\_act\_postA, cl\_fUm\_act\_postA, CL\_FAs\_TC\_FINALA, CL\_FAm\_TC\_FINALA, CL\_FUs\_TC\_FINALA, CL\_FUm\_TC\_FINALA,

CL\_FAs\_UNITCOST\_POSTA, CL\_FAm\_UNITCOST\_POSTA, CL\_FUs\_UNITCOST\_POSTA, CL\_FUm\_UNITCOST\_POSTA FROM newer\_t3E

ORDER BY PAED\_PAIRINGS, ADULT\_CHILD;

**QUIT**;

**DATA** newer\_t3G;

SET newer\_t3F;

if paed\_pairings="999" then paed\_pairings="";

else paed\_pairings=paed\_pairings;

IF cl\_faS\_act\_postA LT **50** THEN LowVol\_FAS=(CL\_FAS\_TC\_FINALA+CL\_FUS\_TC\_FINALA)/(cl\_faS\_act\_postA+cl\_fUS\_act\_postA);

ELSE IF cl\_fUS\_act\_postA LT **50** THEN LowVol\_FAS=(CL\_FAS\_TC\_FINALA+CL\_FUS\_TC\_FINALA)/(cl\_faS\_act\_postA+cl\_fUS\_act\_postA);

ELSE LowVol\_FAS=CL\_FAs\_UNITCOST\_POSTA;

IF cl\_fam\_act\_postA LT **50** THEN LowVol\_FAM=(CL\_FAM\_TC\_FINALA+CL\_FUM\_TC\_FINALA)/(cl\_fam\_act\_postA+cl\_fUm\_act\_postA);

ELSE IF cl\_fUm\_act\_postA LT **50** THEN LowVol\_FAM=(CL\_FAM\_TC\_FINALA+CL\_FUM\_TC\_FINALA)/(cl\_fam\_act\_postA+cl\_fUm\_act\_postA);

ELSE LowVol\_FAM=CL\_FAM\_UNITCOST\_POSTA;

IF cl\_faS\_act\_postA LT **50** THEN LowVol\_FuS=(CL\_FAS\_TC\_FINALA+CL\_FUS\_TC\_FINALA)/(cl\_faS\_act\_postA+cl\_fUS\_act\_postA);

ELSE IF cl\_fUS\_act\_postA LT **50** THEN LowVol\_FuS=(CL\_FAS\_TC\_FINALA+CL\_FUS\_TC\_FINALA)/(cl\_faS\_act\_postA+cl\_fUS\_act\_postA);

ELSE LowVol\_FuS=CL\_Fus\_UNITCOST\_POSTA;

IF cl\_fUm\_act\_postA LT **50** THEN LowVol\_FUM=(CL\_FAM\_TC\_FINALA+CL\_FUM\_TC\_FINALA)/(cl\_fam\_act\_postA+cl\_fUm\_act\_postA);

ELSE IF cl\_fUm\_act\_postA LT **50** THEN LowVol\_FUM=(CL\_FAM\_TC\_FINALA+CL\_FUM\_TC\_FINALA)/(cl\_fam\_act\_postA+cl\_fUm\_act\_postA);

ELSE LowVol\_FUM=CL\_FUM\_UNITCOST\_POSTA;

IF LowVol\_FAs LT LowVol\_Fus THEN FAS\_gtE\_FUM=((LowVol\_FAs\*cl\_fas\_act\_postA)+(LowVol\_FUs\*cl\_fUs\_act\_postA))/(cl\_fas\_act\_postA+cl\_fUs\_act\_postA);

ELSE FAS\_gtE\_FUM=LowVol\_FAS;

IF LowVol\_FAM LT LowVol\_FuM THEN FAM\_gtE\_FUM=((LowVol\_FAM\*cl\_fam\_act\_postA)+(LowVol\_FUM\*cl\_fUm\_act\_postA))/(cl\_fam\_act\_postA+cl\_fUm\_act\_postA);

ELSE FAM\_gtE\_FUM=LowVol\_FAM;

IF LowVol\_FAs LT LowVol\_Fus THEN FUS\_ltE\_FAS=((LowVol\_FAs\*cl\_fas\_act\_postA)+(LowVol\_FUs\*cl\_fUs\_act\_postA))/(cl\_fas\_act\_postA+cl\_fUs\_act\_postA);

IF LowVol\_FAM LT LowVol\_FuM THEN FUM\_ltE\_FAM=((LowVol\_FAM\*cl\_fam\_act\_postA)+(LowVol\_FUM\*cl\_fUm\_act\_postA))/(cl\_fam\_act\_postA+cl\_fUm\_act\_postA);

ELSE FUM\_ltE\_FAM=LowVol\_FUM;

IF FAM\_GTE\_FUM^=**.** and FAM\_GTE\_FUM LT FAS\_gtE\_FUM THEN FAS\_ngt\_FAM=((FAM\_GTE\_FUM\*cl\_fam\_act\_postA)+(FAS\_gtE\_FUM\*cl\_faS\_act\_postA))/(cl\_fam\_act\_postA+cl\_faS\_act\_postA);

else FAS\_ngt\_FAM=FAS\_GTE\_FUM;

IF FAM\_GTE\_FUM ^=**.** and FAM\_GTE\_FUM LT FAS\_gtE\_FUM THEN FAM\_nlt\_FAS=((FAM\_GTE\_FUM\*cl\_fam\_act\_postA)+(FAS\_gtE\_FUM\*cl\_faS\_act\_postA))/(cl\_fam\_act\_postA+cl\_faS\_act\_postA);

else FAM\_nlt\_FAS=FAM\_GTE\_FUM;

IF FUM\_ltE\_FAM^=**.** and FUM\_ltE\_FAM LT FUS\_ltE\_FAS THEN FUS\_ngt\_FUM=((FUM\_ltE\_FAM\*cl\_fUm\_act\_postA)+(FUS\_ltE\_FAS\*cl\_fUS\_act\_postA))/(cl\_fUm\_act\_postA+cl\_fUS\_act\_postA);

else FUS\_ngt\_FUM=FUS\_ltE\_FAS;

IF FUM\_ltE\_FAM^=**.** and FUM\_ltE\_FAM LT FUS\_ltE\_FAS THEN FUM\_nlt\_FUS=((FUM\_ltE\_FAM\*cl\_fUm\_act\_postA)+(FUS\_ltE\_FAS\*cl\_fUS\_act\_postA))/(cl\_fUm\_act\_postA+cl\_fUS\_act\_postA);

else FUM\_nlt\_FUS=FUM\_ltE\_FAM;

/\* POLICY QUESTIONS RAISED HERE, IN CONDITIONS BELOW BECAUSE OF ANOMALY IN EXCEL FORMULA RESULTS WITH REGARDS TO IF ACTIVITY AND UNIT PRICE ARE 0 FOR EXAMPLE TFC 345\*/

if FAM\_nlt\_FAS=**.** then FAM\_nlt\_FAS=**0**; else FAM\_nlt\_FAS=FAM\_nlt\_FAS;

if FUM\_nlt\_FUS=**.** then FUM\_nlt\_FUS=**0**; else FUM\_nlt\_FUS=FUM\_nlt\_FUS;

IF FAM\_nlt\_FAS=**0** or FAM\_nlt\_FAS GT **2**\*FAS\_ngt\_FAM THEN FASx2\_lt\_FAM=((FAM\_nlt\_FAS\*cl\_fam\_act\_postA)+(FAS\_ngt\_FAM\*cl\_faS\_act\_postA))/(**2**\*cl\_fam\_act\_postA+cl\_faS\_act\_postA);

else FASx2\_lt\_FAM=FAS\_ngt\_FAM;

IF FAM\_nlt\_FAS=**0** or FAM\_nlt\_FAS>**2**\*FAS\_ngt\_FAM THEN FAM\_gt\_2xFAS=(((FAM\_nlt\_FAS\*cl\_fam\_act\_postA)+(FAS\_ngt\_FAM\*cl\_faS\_act\_postA))/(**2**\*cl\_fam\_act\_postA+cl\_faS\_act\_postA))\***2**;

else FAM\_gt\_2xFAS=FAM\_nlt\_FAS;

IF FUM\_nlt\_FUS=**0** or FUM\_nlt\_FUS GT **2**\*FUS\_ngt\_FUM THEN FUSx2\_lt\_FUM=((FUM\_nlt\_FUS\*cl\_fUm\_act\_postA)+(FUS\_ngt\_FUM\*cl\_fUS\_act\_postA))/(**2**\*cl\_fUm\_act\_postA+cl\_fUS\_act\_postA);

else FUSx2\_lt\_FUM=FUS\_ngt\_FUM;

IF FUM\_nlt\_FUS=**0** or FUM\_nlt\_FUS GT **2**\*FUS\_ngt\_FUM THEN FUM\_gt\_2xFUS=(((FUM\_nlt\_FUS\*cl\_fUm\_act\_postA)+(FUS\_ngt\_FUM\*cl\_fUS\_act\_postA))/(**2**\*cl\_fUm\_act\_postA+cl\_fUS\_act\_postA))\***2**;

else FUM\_gt\_2xFUS=FUM\_nlt\_FUS;

**run**;

**PROC** **SQL**;

CREATE TABLE newer\_t3H AS SELECT MONOTONIC() AS list\_nos, \* from newer\_t3G;

**quit**;

**data** newer\_t3i;

set newer\_t3H;

where list\_nos le **56**; **run**;

**proc** **transpose** data=newer\_t3i out=newer\_t3ib;

by Paed\_Pairings;

id adult\_child;

var fam\_gt\_2xfas FASx2\_lt\_FAM FUM\_gt\_2xFUS FUSx2\_lt\_FUM;

**run**;

**proc** **sql**; create table newer\_t3ib2 as select monotonic() as what\_nos, \* from newer\_t3ib; **quit**;

**proc** **transpose** data=newer\_t3i out=newer\_t3ic;

by Paed\_Pairings;

id adult\_child;

var CL\_FAM\_act\_PostA CL\_FAS\_act\_PostA CL\_FUM\_act\_PostA CL\_FUS\_act\_PostA;

**run**;

**data** newer\_t3id( rename=a=AA rename=c=CC); set newer\_t3ic; **run**;

**proc** **sql**; create table newer\_t3id2 as select monotonic() as what\_nos, \* from newer\_t3id; **quit**;

**PROC** **SQL**; CREATE TABLE NEW\_JOIN1 AS SELECT \* from newer\_t3ib2 AS a left join newer\_t3id2 as b on a.what\_nos=b.what\_nos; **quit**;

**data** newer\_t3j;

set NEW\_JOIN1;

if c=**.** then c=**0**; if cc=**.** then cc=**0**;

if A>C then paed\_lt\_adult=((a\*aa)+(c\*cc))/(aa+cc); else paed\_lt\_adult=a;

**run**;

**proc** **transpose** data=newer\_t3j out=newer\_t3k;

by Paed\_Pairings;

id \_name\_;

var paed\_lt\_adult;

**run**;

**data** newer\_t3j2;

set NEW\_JOIN1;

if A>C then paed\_lt\_adult=((a\*aa)+(c\*cc))/(aa+cc); else paed\_lt\_adult=c;

**run**;

**proc** **transpose** data=newer\_t3j2 out=newer\_t3k2;

by Paed\_Pairings;

id \_name\_;

var paed\_lt\_adult;

**run**;

**data** NEW\_set1 (rename=fam\_gt\_2xfas=fam\_paed\_lt\_adult rename=FASx2\_lt\_FAM=fas\_paed\_lt\_adult

rename=FUM\_gt\_2xFUS=fum\_paed\_lt\_adult rename=FUSx2\_lt\_FUM=fus\_paed\_lt\_adult);

set newer\_t3k newer\_t3k2;

**run**;

**proc** **sort** data=new\_set1 out=new\_set2;

by paed\_pairings;

**run**;

**proc** **sql**; create table new\_set2a as select monotonic() as list\_nos, Paed\_Pairings, \_name\_, fas\_paed\_lt\_adult,

fam\_paed\_lt\_adult, fus\_paed\_lt\_adult, fum\_paed\_lt\_adult from new\_set2; **quit**;

**proc** **sql**; create table new\_join2 as select a.\*, b.\* from newer\_t3i as a left join new\_set2a as b on a.list\_nos=b.list\_nos; **quit**;

**data** newer\_t3i2;

set newer\_t3H;

where list\_nos gt **56**;

\_name\_="";

fas\_paed\_lt\_adult=FASx2\_lt\_FAM;

fam\_paed\_lt\_adult=fam\_gt\_2xfas;

fus\_paed\_lt\_adult=FUSx2\_lt\_FUM;

fum\_paed\_lt\_adult=FUM\_gt\_2xFUS;

**run**;

**data** NEW\_set3;

set new\_join2 newer\_t3i2;

**run**;

**data** test;

set new\_set3;

where tfc in(**659** **346** **344** **342** **723** **663** **721** **328** **345** **106** **213**);

**run**;

**Proc** **sql**; create table hrgs\_to\_hrgs\_mandatory3 as select distinct tfc, mandatory from hrgs\_to\_hrgs\_mandatory2; **quit**;

**data** hrgs\_to\_hrgs\_mandatory4(drop=mandatory rename=mandatory\_yes=Mandatory); set hrgs\_to\_hrgs\_mandatory3; if mandatory="Yes" then Mandatory\_Yes=**1**; else Mandatory\_Yes=**0**; **run**;

**data** hrgs\_to\_hrgs\_mandatory5;

set hrgs\_to\_hrgs\_mandatory4;

TFC2=input(TFC, **8.**);

drop TFC;

rename TFC2=TFC;

**run**;

**proc** **sql**; create table further\_adjustment\_done as select a.list\_nos, a.TFC, a.Tariff\_TFC, a.Paed\_Pairings, a.Adult\_Child, b.mandatory, a.\* from NEW\_set3 as a

left join hrgs\_to\_hrgs\_mandatory5 as b on a.tfc=b.tfc;

**quit**;

**proc** **sort** data=further\_adjustment\_done out=further\_adjustment\_done2;

by list\_nos;

**run**;

**data** further\_adjusted(drop=list\_nos \_name\_);

set further\_adjustment\_done2;

**run**;

**data** further\_adjusted2; set further\_adjusted; where mandatory=**1**; **run**;

**data** further\_adjusted3; set further\_adjusted2;

Quantum=((fam\_paed\_lt\_adult\*cl\_fam\_act\_postA)+(fas\_paed\_lt\_adult\*cl\_faS\_act\_postA)+(fum\_paed\_lt\_adult\*cl\_fum\_act\_postA)+(fus\_paed\_lt\_adult\*cl\_fuS\_act\_postA));

format Quantum comma12.0;

**run**;

**proc** **export** data=further\_adjusted3

outfile="&outputpath.\further\_adjusted3.csv"

dbms=csv

replace;

**run**;

/\* Calculating YOY Quantum \*/

**data** YOY\_Quantum\_wip; set further\_adjusted3;

YOY\_Quantum=((fam\_paed\_lt\_adult\*cl\_fam\_act\_postA)+(fas\_paed\_lt\_adult\*cl\_faS\_act\_postA)+(fum\_paed\_lt\_adult\*cl\_fum\_act\_postA)+(fus\_paed\_lt\_adult\*cl\_fuS\_act\_postA));

**run**;

**data** xxx; set work.QUANTUMRECONCILATION\_OUTPUT; **run**;

**proc** **sql**; create table YOY\_Quantum\_final as select "OPATT" FORMAT=$20. as POD, sum(YOY\_Quantum) as YOY\_Quantum\_all from YOY\_Quantum\_wip

where tariff\_tfc^="Diagnostic Imaging"

group by mandatory;

**quit**;

**DATA** AAA; set YOY\_Quantum\_final;quantum4qr=YOY\_Quantum\_all; **run**;

**PROC** **SQL**;

CREATE TABLE B2 AS SELECT A.\*, b.quantumrecvalues, (b.quantumrecvalues/A.quantum4qr)-**1** as qr\_factor1 FROM AAA AS A

LEFT JOIN xxx AS B

ON A.POD=B.POD

where B.pod="OPATT"

**QUIT**;

**proc** **sql**;

select qr\_factor1 into :qr1\_factor

from b2

where POD="OPATT"

;

**quit**;

%let rc\_qr1\_factor=&qr1\_factor.;

**DATA** adjust\_4\_DI; SET further\_adjusted3;

if TFC=**812** THEN fam\_paed\_lt\_adult=**0**; else fam\_paed\_lt\_adult=fam\_paed\_lt\_adult;

if TFC=**812** THEN fas\_paed\_lt\_adult=**0**; else fas\_paed\_lt\_adult=fas\_paed\_lt\_adult;

if TFC=**812** THEN fum\_paed\_lt\_adult=**0**; else fum\_paed\_lt\_adult=fum\_paed\_lt\_adult;

if TFC=**812** THEN fus\_paed\_lt\_adult=**0**; else fus\_paed\_lt\_adult=fus\_paed\_lt\_adult;

**RUN**;

/\* Calculating Final Prices \*/

**data** adjust\_4\_DI0;

set adjust\_4\_DI;

rc\_qr1\_factor=&rc\_qr1\_factor.;

FAs\_prices\_QR1=fas\_paed\_lt\_adult\*(**1**+rc\_qr1\_factor);

FAm\_prices\_QR1=fam\_paed\_lt\_adult\*(**1**+rc\_qr1\_factor);

FUs\_prices\_QR1=fus\_paed\_lt\_adult\*(**1**+rc\_qr1\_factor);

FUm\_prices\_QR1=fum\_paed\_lt\_adult\*(**1**+rc\_qr1\_factor);

Cost\_base\_adjustment\_factor=&Cost\_base\_adjustment\_factor.;

FAs\_prices\_CB=FAs\_prices\_QR1\*(**1**+Cost\_base\_adjustment\_factor);

FAm\_prices\_CB=FAm\_prices\_QR1\*(**1**+Cost\_base\_adjustment\_factor);

FUs\_prices\_CB=FUs\_prices\_QR1\*(**1**+Cost\_base\_adjustment\_factor);

FUm\_prices\_CB=FUm\_prices\_QR1\*(**1**+Cost\_base\_adjustment\_factor);

ie\_indextn\_adjstmt\_factors=&ie\_indextn\_adjstmt\_factors.;

FAs\_prices\_IE=FAs\_prices\_CB\*(**1**+ie\_indextn\_adjstmt\_factors);

FAm\_prices\_IE=FAm\_prices\_CB\*(**1**+ie\_indextn\_adjstmt\_factors);

FUs\_prices\_IE=FUs\_prices\_CB\*(**1**+ie\_indextn\_adjstmt\_factors);

FUm\_prices\_IE=FUm\_prices\_CB\*(**1**+ie\_indextn\_adjstmt\_factors);

**run**;

**data** adjust\_4\_di1(DROP=FAs\_prices\_IE FAm\_prices\_IE Fus\_prices\_IE Fum\_prices\_IE);

set adjust\_4\_di0;

rename FAs\_prices\_IE=WF01B\_FA\_Single\_PreMA; rename FAm\_prices\_IE=WF02B\_FA\_Multi\_PreMA;

rename Fus\_prices\_IE=WF01A\_FU\_Single\_PreMA; rename Fum\_prices\_IE=WF02A\_FU\_Multi\_PreMA;

**run**;

**data** adjust\_4\_DI2; set adjust\_4\_DI1; if tfc=**329** then WF01A\_FU\_Single\_PreMA=**.**; if tfc=**329** then WF02A\_FU\_Multi\_PreMA=**.**;**run**;

/\* added 2 programs below to equalise tfc 370 nand 800 prices \*/

**proc** **sql**;

select WF01B\_FA\_Single\_preMA, WF02B\_FA\_Multi\_preMA, WF01A\_FU\_Single\_preMA, WF02A\_FU\_Multi\_preMA

into :WF01B\_FA\_Single\_preMA, :WF02B\_FA\_Multi\_preMA, :WF01A\_FU\_Single\_preMA, :WF02A\_FU\_Multi\_preMA

from adjust\_4\_DI2

where tfc=**370**;

**quit**;

**data** adjust\_4\_DI2; set adjust\_4\_DI2;

if tfc=**800** then WF01B\_FA\_Single\_preMA=&WF01B\_FA\_Single\_preMA.; else WF01B\_FA\_Single\_preMA=WF01B\_FA\_Single\_preMA;

if tfc=**800** then WF02B\_FA\_Multi\_preMA=&WF02B\_FA\_Multi\_preMA.; else WF02B\_FA\_Multi\_preMA=WF02B\_FA\_Multi\_preMA;

if tfc=**800** then WF01A\_FU\_Single\_preMA=&WF01A\_FU\_Single\_preMA.; else WF01A\_FU\_Single\_preMA=WF01A\_FU\_Single\_preMA;

if tfc=**800** then WF02A\_FU\_Multi\_preMA=&WF02A\_FU\_Multi\_preMA.; else WF02A\_FU\_Multi\_preMA=WF02A\_FU\_Multi\_preMA;

**run**;

**DATA** adjust\_4\_DI3; SET adjust\_4\_DI2;

QUANTUM\_PRE\_MA=(coalesce(WF01B\_FA\_Single\_PreMA,**0**)\*cl\_fas\_act\_postA)+(coalesce(WF02B\_FA\_Multi\_PreMA,**0**)\*cl\_fam\_act\_postA)+(coalesce(WF01A\_FU\_Single\_PreMA,**0**)\*cl\_fus\_act\_postA)+(coalesce(WF02A\_FU\_Multi\_PreMA,**0**)\*cl\_fum\_act\_postA);

**RUN**;

/\* Calculating YOY Quantum \*/

**data** YOY\_Quantum\_wip; set further\_adjusted3;

YOY\_Quantum=((fam\_paed\_lt\_adult\*cl\_fam\_act\_postA)+(fas\_paed\_lt\_adult\*cl\_faS\_act\_postA)+(fum\_paed\_lt\_adult\*cl\_fum\_act\_postA)+(fus\_paed\_lt\_adult\*cl\_fuS\_act\_postA));

**run**;

**data** xxx; set WORK.QUANTUMRECONCILATION\_OUTPUT; **run**;

**proc** **sql**; create table YOY\_Quantum\_final as select "OPATT" FORMAT=$20. as POD, sum(YOY\_Quantum) as YOY\_Quantum\_all from YOY\_Quantum\_wip

where tariff\_tfc^="Diagnostic Imaging"

group by mandatory;

**quit**;

**DATA** AAA; set YOY\_Quantum\_final;quantum4qr=YOY\_Quantum\_all; **run**;

**PROC** **SQL**;

CREATE TABLE B2 AS SELECT A.\*, b.quantumrecvalues, (b.quantumrecvalues/A.quantum4qr)-**1** as qr\_factor1 FROM AAA AS A

LEFT JOIN xxx AS B

ON A.POD=B.POD

where B.pod="OPATT";

**QUIT**;

**proc** **sql**;

select qr\_factor1 into :qr1\_factor

from b2

where POD="OPATT"

;

**quit**;

%let rc\_qr1\_factor=&qr1\_factor.;

/\* Calculating Final Prices \*/

**DATA** adjust\_4\_DI; SET further\_adjusted3;

if TFC=**812** THEN fam\_paed\_lt\_adult=**0**; else fam\_paed\_lt\_adult=fam\_paed\_lt\_adult;

if TFC=**812** THEN fas\_paed\_lt\_adult=**0**; else fas\_paed\_lt\_adult=fas\_paed\_lt\_adult;

if TFC=**812** THEN fum\_paed\_lt\_adult=**0**; else fum\_paed\_lt\_adult=fum\_paed\_lt\_adult;

if TFC=**812** THEN fus\_paed\_lt\_adult=**0**; else fus\_paed\_lt\_adult=fus\_paed\_lt\_adult;

**RUN**;

**proc** **sql**; create table OPATT\_PRICES\_QR1 as select **4** format=**8.** as opatt\_price\_id, "19/20" format=$10. as Year, TFC, Tariff\_TFC as TFC\_description,

cl\_fam\_act\_postA, cl\_fas\_act\_postA, cl\_fum\_act\_postA, cl\_fus\_act\_postA /\* NEWLY added by PID \*/,

fas\_paed\_lt\_adult\*(**1**+&rc\_qr1\_factor.) as WF01B\_FA\_Single1/\*FAS\_prices\_1718\*/ FORMAT=**8.**, fam\_paed\_lt\_adult\*(**1**+&rc\_qr1\_factor.) as WF02B\_FA\_Multi1/\*FAM\_prices\_1718\*/ FORMAT=**8.**,

fus\_paed\_lt\_adult\*(**1**+&rc\_qr1\_factor.) as WF01A\_FU\_Single1/\*FUS\_prices\_1718\*/ FORMAT=**8.**, fum\_paed\_lt\_adult\*(**1**+&rc\_qr1\_factor.) as WF02A\_FU\_Multi1/\*FUM\_prices\_1718\*/ FORMAT=**8.**

from adjust\_4\_DI

order by TFC;

**quit**;

**data** OPATT\_PRICES\_4IA\_CB;

set OPATT\_PRICES\_QR1;

Cost\_base\_adjustment\_factor=&Cost\_base\_adjustment\_factor.;

WF01B\_FA\_Single2=WF01B\_FA\_Single1\*(**1**+Cost\_base\_adjustment\_factor);

WF02B\_FA\_Multi2=WF02B\_FA\_Multi1\*(**1**+Cost\_base\_adjustment\_factor);

WF01A\_FU\_Single2=WF01A\_FU\_Single1\*(**1**+Cost\_base\_adjustment\_factor);

WF02A\_FU\_Multi2=WF02A\_FU\_Multi1\*(**1**+Cost\_base\_adjustment\_factor);

**run**;

**proc** **sql**; create table OPATT\_PRICES\_4IA as select opatt\_price\_id, Year, TFC, TFC\_description,

cl\_fas\_act\_postA, cl\_fam\_act\_postA, cl\_fus\_act\_postA, cl\_fum\_act\_postA /\* NEWLY added by PID \*/,

WF01B\_FA\_Single2\*(**1**+&ie\_indextn\_adjstmt\_factors.) as WF01B\_FA\_Single\_preMA/\*FAS\_prices\_1718\*/ FORMAT=**8.**, WF02B\_FA\_Multi2\*(**1**+&ie\_indextn\_adjstmt\_factors.) as WF02B\_FA\_Multi\_preMA/\*FAM\_prices\_1718\*/ FORMAT=**8.**,

WF01A\_FU\_Single2\*(**1**+&ie\_indextn\_adjstmt\_factors.) as WF01A\_FU\_Single\_preMA/\*FUS\_prices\_1718\*/ FORMAT=**8.**, WF02A\_FU\_Multi2\*(**1**+&ie\_indextn\_adjstmt\_factors.) as WF02A\_FU\_Multi\_preMA/\*FUM\_prices\_1718\*/ FORMAT=**8.**

from OPATT\_PRICES\_4IA\_CB

order by TFC;

**quit**;

**data** OPATT\_PRICES\_4IA; set OPATT\_PRICES\_4IA; if tfc=**329** then WF01A\_FU\_Single\_preMA=**.**; if tfc=**329** then WF02A\_FU\_Multi\_preMA=**.**;**run**;

/\* added 2 programs below to equalise tfc 370 nand 800 prices \*/

**proc** **sql**;

select WF01B\_FA\_Single\_preMA, WF02B\_FA\_Multi\_preMA, WF01A\_FU\_Single\_preMA, WF02A\_FU\_Multi\_preMA

into :WF01B\_FA\_Single\_preMA, :WF02B\_FA\_Multi\_preMA, :WF01A\_FU\_Single\_preMA, :WF02A\_FU\_Multi\_preMA

from OPATT\_PRICES\_4IA

where tfc=**370**;

**quit**;

**data** OPATT\_PRICES\_4IA; set OPATT\_PRICES\_4IA;

if tfc=**800** then WF01B\_FA\_Single\_preMA=&WF01B\_FA\_Single\_preMA.; else WF01B\_FA\_Single\_preMA=WF01B\_FA\_Single\_preMA;

if tfc=**800** then WF02B\_FA\_Multi\_preMA=&WF02B\_FA\_Multi\_preMA.; else WF02B\_FA\_Multi\_preMA=WF02B\_FA\_Multi\_preMA;

if tfc=**800** then WF01A\_FU\_Single\_preMA=&WF01A\_FU\_Single\_preMA.; else WF01A\_FU\_Single\_preMA=WF01A\_FU\_Single\_preMA;

if tfc=**800** then WF02A\_FU\_Multi\_preMA=&WF02A\_FU\_Multi\_preMA.; else WF02A\_FU\_Multi\_preMA=WF02A\_FU\_Multi\_preMA;

**run**;

/\* lines added stopped here \*/

**DATA** OPATT\_PRICES\_4IA; SET OPATT\_PRICES\_4IA;

QUANTUM\_PRE\_MA=(coalesce(WF01B\_FA\_Single\_preMA,**0**)\*cl\_fas\_act\_postA)+(coalesce(WF02B\_FA\_Multi\_preMA,**0**)\*cl\_fam\_act\_postA)+(coalesce(WF01A\_FU\_Single\_preMA,**0**)\*cl\_fus\_act\_postA)+(coalesce(WF02A\_FU\_Multi\_preMA,**0**)\*cl\_fum\_act\_postA);

**RUN**;

Libname PostMA base "\\irnarch\sas\_data\Tariff Rebuild\Model outputs pre\_MA\opatt\FY\_1920\_s118\data";

**DATA** MA\_opatt\_Input;

SET PostMA.MA\_opatt\_Input;

**RUN**;

**DATA** MA\_opatt\_Output;

SET PostMA.MA\_opatt\_Output;

**RUN**;

**proc** **sql**;

create table new1 as select

a.\*,

b.FAS,

b.FAM,

b.FUS,

b.FUM,

c.PostMA\_FAS,

c.PostMA\_FAM,

c.PostMA\_FUS,

c.PostMA\_FUM

from OPATT\_PRICES\_4IA as a

left join MA\_opatt\_Input as b

on a.tfc=b.tfc

LEFT JOIN MA\_opatt\_Output

as c

on

a.tfc=c.tfc;

**quit**;

**data** new2;

set new1;

if fas="Manually adjusted" or fam="Manually adjusted" or fus="Manually adjusted" or fum="Manually adjusted"

then MA=**1**; ELSE MA=**0**;

**run**;

**data** new3;

set new2;

if FAS="Manually adjusted" then WF01B\_FA\_Single\_postMA=PostMA\_FAS; else WF01B\_FA\_Single\_postMA=WF01B\_FA\_Single\_preMA;

IF FAM="Manually adjusted" THEN WF02B\_FA\_Multi\_postMA=PostMA\_FAM; else WF02B\_FA\_Multi\_postMA=WF02B\_FA\_Multi\_preMA;

IF FUS="Manually adjusted" THEN WF01A\_FU\_Single\_postMA=PostMA\_FUS; else WF01A\_FU\_Single\_postMA=WF01A\_FU\_Single\_preMA;

IF FUM="Manually adjusted" THEN WF02A\_FU\_Multi\_postMA=PostMA\_FUM; else WF02A\_FU\_Multi\_postMA=WF02A\_FU\_Multi\_preMA;

**RUN**;

**DATA** new4(DROP=FAS FAM FUS FUM PostMA\_FAS PostMA\_FAM PostMA\_FUS PostMA\_FUM MA);

SET new3;

QUANTUM\_Post\_MA=(coalesce(WF01B\_FA\_Single\_postMA,**0**)\*cl\_fas\_act\_postA)+(coalesce(WF02B\_FA\_Multi\_postMA,**0**)\*cl\_fam\_act\_postA)+(coalesce(WF01A\_FU\_Single\_postMA,**0**)\*cl\_fus\_act\_postA)+(coalesce(WF02A\_FU\_Multi\_postMA,**0**)\*cl\_fum\_act\_postA);

**RUN**;

**proc** **sql**; create table new5 as select sum(quantum\_pre\_ma) as TOTAL\_QUANTUM\_PRE\_MA, sum(quantum\_post\_ma) as TOTAL\_QUANTUM\_POST\_MA from new4; **quit**;

**DATA** NEW6; SET NEW5; QR2\_MA=(TOTAL\_QUANTUM\_PRE\_MA-TOTAL\_QUANTUM\_POST\_MA)/TOTAL\_QUANTUM\_POST\_MA;

**RUN**;

**proc** **sql**;

select QR2\_MA into :qr2\_factor

from new6;

**quit**;

%let rc\_qr2\_factor=&qr2\_factor.;

**data** OPATT\_PRICES\_4\_IA\_model (drop=cl\_fas\_act\_postA cl\_fam\_act\_postA cl\_fus\_act\_postA

cl\_fum\_act\_postA rc\_qr2\_factor quantum\_pre\_ma quantum\_post\_ma

WF01B\_FA\_Single\_postMA WF01B\_FA\_Single\_preMA WF02B\_FA\_Multi\_postMA WF02B\_FA\_Multi\_preMA

WF01A\_FU\_Single\_postMA WF01A\_FU\_Single\_preMA WF02A\_FU\_Multi\_postMA WF02A\_FU\_Multi\_preMA);

set new4;

rc\_qr2\_factor=&rc\_qr2\_factor.;

WF01B\_FA\_Single=WF01B\_FA\_Single\_postMA\*(**1**+rc\_qr2\_factor);

WF02B\_FA\_Multi=WF02B\_FA\_Multi\_postMA\*(**1**+rc\_qr2\_factor);

WF01A\_FU\_Single=WF01A\_FU\_Single\_postMA\*(**1**+rc\_qr2\_factor);

WF02A\_FU\_Multi=WF02A\_FU\_Multi\_postMA\*(**1**+rc\_qr2\_factor);

format WF01B\_FA\_Single WF02B\_FA\_Multi WF01A\_FU\_Single WF02A\_FU\_Multi comma16.15;

FORMAT rc\_qr2\_factor percentn7.2;

**run**;

**proc** **export** data=OPATT\_PRICES\_4\_IA\_model

outfile="&outputpath.\Prices\_4\_IA\_MODELS.csv"

dbms=csv

replace;

**run**;

**data** modelout.OPATT\_PRICES\_4\_IA\_model;

set OPATT\_PRICES\_4\_IA\_model;

**run**;

**DATA** OPATT\_PRICES\_4\_IA\_model2;

SET ModelOut.OPATT\_PRICES\_4\_IA\_model;

TFC2=put(TFC,**3.**);

drop TFC;

RENAME TFC2=TFC;

**RUN**;

**proc** **sql**; create table ModelOut.OPATT\_PRICES\_4\_IA\_model as select opatt\_price\_id, year, TFC, tfc\_description,

WF01B\_FA\_SINGLE, WF02B\_FA\_MULTI, WF01A\_FU\_SINGLE, WF02A\_FU\_MULTI from OPATT\_PRICES\_4\_IA\_model2;

**quit**;

Libname PostMA base "\\irnarch\sas\_data\Tariff Rebuild\Model outputs pre\_MA\opatt\FY\_1920\_Final\data";

**DATA** MA\_opatt\_Input;

SET PostMA.MA\_opatt\_Input;

**RUN**;

**DATA** MA\_opatt\_Output;

SET PostMA.MA\_opatt\_Output;

**RUN**;

**proc** **sql**;

create table new\_1 as select a.\*, b.FAS,b.FAM,b.FUS,b.FUM,c.PostMA\_FAS, c.PostMA\_FAM,c.PostMA\_FUS,c.PostMA\_FUM from adjust\_4\_DI3 as a

left join MA\_opatt\_Input as b

on a.tfc=b.tfc

LEFT JOIN MA\_opatt\_Output

as c

on

a.tfc=c.tfc;

**quit**;

**data** new\_2;

set new\_1;

if fas="Manually adjusted" or fam="Manually adjusted" or fus="Manually adjusted" or fum="Manually adjusted"

then MA=**1**; ELSE MA=**0**;

**run**;

**data** new\_3;

set new\_2;

if FAS="Manually adjusted" then WF01B\_FA\_Single\_postMA=PostMA\_FAS; else WF01B\_FA\_Single\_postMA=WF01B\_FA\_Single\_preMA;

IF FAM="Manually adjusted" THEN WF02B\_FA\_Multi\_postMA=PostMA\_FAM; else WF02B\_FA\_Multi\_postMA=WF02B\_FA\_Multi\_preMA;

IF FUS="Manually adjusted" THEN WF01A\_FU\_Single\_postMA=PostMA\_FUS; else WF01A\_FU\_Single\_postMA=WF01A\_FU\_Single\_preMA;

IF FUM="Manually adjusted" THEN WF02A\_FU\_Multi\_postMA=PostMA\_FUM; else WF02A\_FU\_Multi\_postMA=WF02A\_FU\_Multi\_preMA;

**RUN**;

**DATA** new\_4(DROP=FAS FAM FUS FUM PostMA\_FAS PostMA\_FAM PostMA\_FUS PostMA\_FUM MA);

SET new\_3;

QUANTUM\_Post\_MA=(coalesce(WF01B\_FA\_Single\_postMA,**0**)\*cl\_fas\_act\_postA)+(coalesce(WF02B\_FA\_Multi\_postMA,**0**)\*cl\_fam\_act\_postA)+(coalesce(WF01A\_FU\_Single\_postMA,**0**)\*cl\_fus\_act\_postA)+(coalesce(WF02A\_FU\_Multi\_postMA,**0**)\*cl\_fum\_act\_postA);

**RUN**;

**proc** **sql**; create table new\_5 as select sum(quantum\_pre\_ma) as TOTAL\_QUANTUM\_PRE\_MA, sum(quantum\_post\_ma) as TOTAL\_QUANTUM\_POST\_MA from new\_4; **quit**;

**DATA** NEW\_6; SET NEW\_5; QR2\_MA=(TOTAL\_QUANTUM\_PRE\_MA-TOTAL\_QUANTUM\_POST\_MA)/TOTAL\_QUANTUM\_POST\_MA;

**RUN**;

**proc** **sql**;

select QR2\_MA into :qr2\_factor

from new\_6;

**quit**;

%let rc\_qr2\_factor=&qr2\_factor.;

**data** new\_7;

set new\_4;

ma\_qr2\_factor=&rc\_qr2\_factor.;

FAs\_prices\_QR2=WF01B\_FA\_Single\_postMA\*(**1**+ma\_qr2\_factor);

FAm\_prices\_QR2=WF02B\_FA\_Multi\_postMA\*(**1**+ma\_qr2\_factor);

FUs\_prices\_QR2=WF01A\_FU\_Single\_postMA\*(**1**+ma\_qr2\_factor);

FUm\_prices\_QR2=WF02A\_FU\_Multi\_postMA\*(**1**+ma\_qr2\_factor);

SMOOTHING\_factor=&SMOOTHING\_factor.;

FAs\_prices\_SF=FAs\_prices\_QR2\*(**1**+SMOOTHING\_factor);

FAm\_prices\_SF=FAm\_prices\_QR2\*(**1**+SMOOTHING\_factor);

FUs\_prices\_SF=FUs\_prices\_QR2\*(**1**+SMOOTHING\_factor);

FUm\_prices\_SF=FUm\_prices\_QR2\*(**1**+SMOOTHING\_factor);

smoothing\_qr3\_factor=&smoothing\_qr3\_factor.;

FAs\_prices\_qr3=FAs\_prices\_SF\*(**1**+smoothing\_qr3\_factor);

FAm\_prices\_qr3=FAm\_prices\_SF\*(**1**+smoothing\_qr3\_factor);

FUs\_prices\_qr3=FUs\_prices\_SF\*(**1**+smoothing\_qr3\_factor);

FUm\_prices\_qr3=FUm\_prices\_SF\*(**1**+smoothing\_qr3\_factor);

Scaling\_factor=&Scaling\_factor.;

FAs\_prices\_SCALING=FAs\_prices\_qr3\*(**1**+Scaling\_factor);

FAm\_prices\_SCALING=FAm\_prices\_qr3\*(**1**+Scaling\_factor);

FUs\_prices\_SCALING=FUs\_prices\_qr3\*(**1**+Scaling\_factor);

FUm\_prices\_SCALING=FUm\_prices\_qr3\*(**1**+Scaling\_factor);

Inflation1920=&Inflation1920.;

FAs\_prices\_Inflation1920=FAs\_prices\_SCALING\*(**1**+Inflation1920);

FAm\_prices\_Inflation1920=FAm\_prices\_SCALING\*(**1**+Inflation1920);

FUs\_prices\_Inflation1920=FUs\_prices\_SCALING\*(**1**+Inflation1920);

FUm\_prices\_Inflation1920=FUm\_prices\_SCALING\*(**1**+Inflation1920);

Efficiency1920=&Efficiency1920.;

FAs\_prices\_1920=FAs\_prices\_Inflation1920\*(**1**+Efficiency1920);

FAm\_prices\_1920=FAm\_prices\_Inflation1920\*(**1**+Efficiency1920);

FUs\_prices\_1920=FUs\_prices\_Inflation1920\*(**1**+Efficiency1920);

FUm\_prices\_1920=FUm\_prices\_Inflation1920\*(**1**+Efficiency1920);

format rc\_qr1\_factor Cost\_base\_adjustment\_factor ie\_indextn\_adjstmt\_factors ma\_qr2\_factor SMOOTHING\_factor smoothing\_qr3\_factor /\*Scaling\_factor\*/

Inflation1920 Efficiency1920 percentn7.2;

format Scaling\_factor percentn8.2;

format fam\_prices\_1920 fas\_prices\_1920 fum\_prices\_1920 fus\_prices\_1920 comma7.2;

Format CL\_FAS\_act\_PostA CL\_FAM\_act\_PostA CL\_FUS\_act\_PostA CL\_FUM\_act\_PostA Comma12.0;/\* formatted 20171010 \*/

Format Quantum QUANTUM\_Pre\_MA QUANTUM\_Post\_MA comma22.3;

**run**;

**DATA** OPATT\_Prices\_1920\_STEP\_BY\_STEP (drop=Paed\_Pairings adult\_child Mandatory CL\_FAM\_tc\_FinalA CL\_FAS\_tc\_FinalA CL\_FUM\_tc\_FinalA CL\_FUS\_tc\_FinalA

LowVol\_FAM LowVol\_FAS LowVol\_FUM LowVol\_FUS FAM\_gtE\_FUM FAS\_gtE\_FUM FUM\_ltE\_FAM FUS\_ltE\_FAS

FAM\_nlt\_FAS FAS\_ngt\_FAM FUM\_nlt\_FUS FUS\_ngt\_FUM FAM\_gt\_2xFAS FASx2\_lt\_FAM FUM\_gt\_2xFUS FUSx2\_lt\_FUM /\*Quantum\*/);

set new\_7;

**run**;

**proc** **sort** data=OPATT\_Prices\_1920\_STEP\_BY\_STEP;

by TFC;

**RUN**;

**proc** **export** data=OPATT\_Prices\_1920\_STEP\_BY\_STEP

outfile="&outputpath.\OPATT\_Prices\_1920\_STEP\_BY\_STEP.csv"

dbms=csv

replace;

**run**;

**data** OPATT\_PRICES\_1920(KEEP=TFC TARIFF\_TFC FAM\_PRICES\_1920 FAS\_PRICES\_1920 FUM\_PRICES\_1920 FUS\_PRICES\_1920);

SET OPATT\_Prices\_1920\_STEP\_BY\_STEP;

**RUN**;

**proc** **export** data=OPATT\_Prices\_1920

outfile="&outputpath.\OPATT\_Prices\_1920.csv"

dbms=csv

replace;

**run**;

**data** ModelOut.OPATT\_PRICES\_1920;

set OPATT\_PRICES\_1920;

**run**;

**data** ModelOut.OPATT\_Prices\_1920\_STEP\_BY\_STEP;

set OPATT\_Prices\_1920\_STEP\_BY\_STEP;

format CL\_FAS\_act\_PostA CL\_FAM\_act\_PostA CL\_FUS\_act\_PostA CL\_FUM\_act\_PostA comma8.0;

format Quantum QUANTUM\_PRE\_MA QUANTUM\_Post\_MA Comma22.3;

**run**;

Libname PostMA base "\\irnarch\sas\_data\Tariff Rebuild\Model outputs pre\_MA\opatt\FY\_1920\_s118\data";

**DATA** MA\_opatt\_Input;

SET PostMA.MA\_opatt\_Input;

**RUN**;

**DATA** MA\_opatt\_Output;

SET PostMA.MA\_opatt\_Output;

**RUN**;

**proc** **sql**;

create table new\_1 as select a.\*, b.FAS,b.FAM,b.FUS,b.FUM,c.PostMA\_FAS, c.PostMA\_FAM,c.PostMA\_FUS,c.PostMA\_FUM from adjust\_4\_DI3 as a

left join MA\_opatt\_Input as b

on a.tfc=b.tfc

LEFT JOIN MA\_opatt\_Output

as c

on

a.tfc=c.tfc;

**quit**;

**data** new\_2;

set new\_1;

if fas="Manually adjusted" or fam="Manually adjusted" or fus="Manually adjusted" or fum="Manually adjusted"

then MA=**1**; ELSE MA=**0**;

**run**;

**data** new\_3;

set new\_2;

if FAS="Manually adjusted" then WF01B\_FA\_Single\_postMA=PostMA\_FAS; else WF01B\_FA\_Single\_postMA=WF01B\_FA\_Single\_preMA;

IF FAM="Manually adjusted" THEN WF02B\_FA\_Multi\_postMA=PostMA\_FAM; else WF02B\_FA\_Multi\_postMA=WF02B\_FA\_Multi\_preMA;

IF FUS="Manually adjusted" THEN WF01A\_FU\_Single\_postMA=PostMA\_FUS; else WF01A\_FU\_Single\_postMA=WF01A\_FU\_Single\_preMA;

IF FUM="Manually adjusted" THEN WF02A\_FU\_Multi\_postMA=PostMA\_FUM; else WF02A\_FU\_Multi\_postMA=WF02A\_FU\_Multi\_preMA;

**RUN**;

**DATA** new\_4(DROP=FAS FAM FUS FUM PostMA\_FAS PostMA\_FAM PostMA\_FUS PostMA\_FUM MA);

SET new\_3;

QUANTUM\_Post\_MA=(coalesce(WF01B\_FA\_Single\_postMA,**0**)\*cl\_fas\_act\_postA)+(coalesce(WF02B\_FA\_Multi\_postMA,**0**)\*cl\_fam\_act\_postA)+(coalesce(WF01A\_FU\_Single\_postMA,**0**)\*cl\_fus\_act\_postA)+(coalesce(WF02A\_FU\_Multi\_postMA,**0**)\*cl\_fum\_act\_postA);

**RUN**;

**proc** **sql**; create table new\_5 as select sum(quantum\_pre\_ma) as TOTAL\_QUANTUM\_PRE\_MA, sum(quantum\_post\_ma) as TOTAL\_QUANTUM\_POST\_MA from new\_4; **quit**;

**DATA** NEW\_6; SET NEW\_5; QR2\_MA=(TOTAL\_QUANTUM\_PRE\_MA-TOTAL\_QUANTUM\_POST\_MA)/TOTAL\_QUANTUM\_POST\_MA;

**RUN**;

**proc** **sql**;

select QR2\_MA into :qr2\_factor

from new\_6;

**quit**;

%let rc\_qr2\_factor=&qr2\_factor.;

/\* PID added code below re CASH IO \*/

**proc** **sql**;

select final\_cashio\_factor into :cashio\_factor\_WF

from cash\_in\_out\_ia\_factors

where POD\_Subchapter="OPATT\_WF";

**QUIT**;

**data** new\_7;

set new\_4;

ma\_qr2\_factor=&rc\_qr2\_factor.;

FAs\_prices\_QR2=WF01B\_FA\_Single\_postMA\*(**1**+ma\_qr2\_factor);

FAm\_prices\_QR2=WF02B\_FA\_Multi\_postMA\*(**1**+ma\_qr2\_factor);

FUs\_prices\_QR2=WF01A\_FU\_Single\_postMA\*(**1**+ma\_qr2\_factor);

FUm\_prices\_QR2=WF02A\_FU\_Multi\_postMA\*(**1**+ma\_qr2\_factor);

SMOOTHING\_factor=&SMOOTHING\_factor.;

FAs\_prices\_SF=FAs\_prices\_QR2\*(**1**+SMOOTHING\_factor);

FAm\_prices\_SF=FAm\_prices\_QR2\*(**1**+SMOOTHING\_factor);

FUs\_prices\_SF=FUs\_prices\_QR2\*(**1**+SMOOTHING\_factor);

FUm\_prices\_SF=FUm\_prices\_QR2\*(**1**+SMOOTHING\_factor);

smoothing\_qr3\_factor=&smoothing\_qr3\_factor.;

FAs\_prices\_qr3=FAs\_prices\_SF\*(**1**+smoothing\_qr3\_factor);

FAm\_prices\_qr3=FAm\_prices\_SF\*(**1**+smoothing\_qr3\_factor);

FUs\_prices\_qr3=FUs\_prices\_SF\*(**1**+smoothing\_qr3\_factor);

FUm\_prices\_qr3=FUm\_prices\_SF\*(**1**+smoothing\_qr3\_factor);

Scaling\_factor=&Scaling\_factor.;

FAs\_prices\_SCALING=FAs\_prices\_qr3\*(**1**+Scaling\_factor);

FAm\_prices\_SCALING=FAm\_prices\_qr3\*(**1**+Scaling\_factor);

FUs\_prices\_SCALING=FUs\_prices\_qr3\*(**1**+Scaling\_factor);

FUm\_prices\_SCALING=FUm\_prices\_qr3\*(**1**+Scaling\_factor);

/\*&cashio\_factor\_WF.\*/

FAs\_prices\_SCALING=FAs\_prices\_SCALING\*(**1**+&cashio\_factor\_WF.);

FAm\_prices\_SCALING=FAm\_prices\_SCALING\*(**1**+&cashio\_factor\_WF.);

FUs\_prices\_SCALING=FUs\_prices\_SCALING\*(**1**+&cashio\_factor\_WF.);

FUm\_prices\_SCALING=FUm\_prices\_SCALING\*(**1**+&cashio\_factor\_WF.);

Inflation1920=&Inflation1920.;

FAs\_prices\_Inflation1920=FAs\_prices\_SCALING\*(**1**+Inflation1920);

FAm\_prices\_Inflation1920=FAm\_prices\_SCALING\*(**1**+Inflation1920);

FUs\_prices\_Inflation1920=FUs\_prices\_SCALING\*(**1**+Inflation1920);

FUm\_prices\_Inflation1920=FUm\_prices\_SCALING\*(**1**+Inflation1920);

Efficiency1920=&Efficiency1920.;

FAs\_prices\_1920=FAs\_prices\_Inflation1920\*(**1**+Efficiency1920);

FAm\_prices\_1920=FAm\_prices\_Inflation1920\*(**1**+Efficiency1920);

FUs\_prices\_1920=FUs\_prices\_Inflation1920\*(**1**+Efficiency1920);

FUm\_prices\_1920=FUm\_prices\_Inflation1920\*(**1**+Efficiency1920);

format rc\_qr1\_factor Cost\_base\_adjustment\_factor ie\_indextn\_adjstmt\_factors ma\_qr2\_factor SMOOTHING\_factor smoothing\_qr3\_factor /\*Scaling\_factor\*/

Inflation1920 Efficiency1920 percentn7.2;

format Scaling\_factor percentn8.2;

format fam\_prices\_1920 fas\_prices\_1920 fum\_prices\_1920 fus\_prices\_1920 comma7.2;

Format CL\_FAS\_act\_PostA CL\_FAM\_act\_PostA CL\_FUS\_act\_PostA CL\_FUM\_act\_PostA Comma12.0;

Format Quantum QUANTUM\_Pre\_MA QUANTUM\_Post\_MA comma22.3;

**run**;

**DATA** OPATT\_Prices\_1920\_STEP\_BY\_STEP (drop=Paed\_Pairings adult\_child Mandatory CL\_FAM\_tc\_FinalA CL\_FAS\_tc\_FinalA CL\_FUM\_tc\_FinalA CL\_FUS\_tc\_FinalA

LowVol\_FAM LowVol\_FAS LowVol\_FUM LowVol\_FUS FAM\_gtE\_FUM FAS\_gtE\_FUM FUM\_ltE\_FAM FUS\_ltE\_FAS

FAM\_nlt\_FAS FAS\_ngt\_FAM FUM\_nlt\_FUS FUS\_ngt\_FUM FAM\_gt\_2xFAS FASx2\_lt\_FAM FUM\_gt\_2xFUS FUSx2\_lt\_FUM /\*Quantum\*/);

set new\_7;

**run**;

**proc** **sort** data=OPATT\_Prices\_1920\_STEP\_BY\_STEP;

by TFC;

**RUN**;

**proc** **export** data=OPATT\_Prices\_1920\_STEP\_BY\_STEP

outfile="&outputpath.\OPATT\_Prices\_1920\_STEP\_BY\_STEP.csv"

dbms=csv

replace;

**run**;

**data** OPATT\_PRICES\_1920(KEEP=TFC TARIFF\_TFC FAM\_PRICES\_1920 FAS\_PRICES\_1920 FUM\_PRICES\_1920 FUS\_PRICES\_1920);

SET OPATT\_Prices\_1920\_STEP\_BY\_STEP;

**RUN**;

**proc** **export** data=OPATT\_Prices\_1920

outfile="&outputpath.\OPATT\_Prices\_1920.csv"

dbms=csv

replace;

**run**;

**data** ModelOut.OPATT\_PRICES\_1920;

set OPATT\_PRICES\_1920;

**run**;

**data** ModelOut.OPATT\_Prices\_1920\_STEP\_BY\_STEP;

set OPATT\_Prices\_1920\_STEP\_BY\_STEP;

format CL\_FAS\_act\_PostA CL\_FAM\_act\_PostA CL\_FUS\_act\_PostA CL\_FUM\_act\_PostA comma8.0;

format Quantum QUANTUM\_PRE\_MA QUANTUM\_Post\_MA Comma22.3;

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* author: PID ;

\* date: May 2017 ;

# \* OPATT\_2\_2.sas ;

\* summary: xxxxxx ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

**Proc** **sql**; create table t1st as select

a.hrg , b.total\_act as total\_activities, b.tc as total\_costs,

b.total\_act as revised\_total\_act, b.tc as revised\_tc, c.flag

from work.\_09nonmandhrgopatt as a

left join work.oproc\_national\_data as b

on a.hrg=b.currencyCODE

left join work.exclusions\_hrg\_oproc\_to\_opatt as c

on a.hrg=c.hrg;

**quit**;

**data** t2nd;

set t1st;

if flag=**1** then revised\_total\_act=**0**;

else revised\_total\_act=revised\_total\_act;

if flag=**1** then revised\_tc=**0**;

else revised\_tc=revised\_tc;

**run**;

**data** work.opatt\_eligibility; set work.opatt\_eligibility;**run**;

**data** MANDATORY (rename=tariff\_tfc\_code=TFC

rename=WF01B\_First\_Attendance\_\_\_Single=FAS

rename=WF02B\_First\_Attendance\_\_\_Multi\_P=FAM

rename=WF01A\_Follow\_Up\_Attendance\_\_\_Sin=FUS

rename=WF02A\_Follow\_Up\_Attendance\_\_\_Mul=FUM);

set work.opatt\_eligibility (keep=tariff\_tfc\_code WF01B\_First\_Attendance\_\_\_Single

WF02B\_First\_Attendance\_\_\_Multi\_P WF01A\_Follow\_Up\_Attendance\_\_\_Sin WF02A\_Follow\_Up\_Attendance\_\_\_Mul);

**run**;

**data** hrgs\_to\_tfcs2;

set work.oproc\_to\_opatt\_mapping\_new;

TFC2=put(TFC, **3.**);

drop TFC;

rename TFC2=TFC;

**run**;

**data** MANDATORY;

set work.MANDATORY;

TFC2=put(TFC, **3.**);

drop TFC;

rename TFC2=TFC;

**run**;

**proc** **sql**; create table hrgs\_to\_hrgs\_mandatory as select

a.\*, b.tfc as tfc2, b.\* from hrgs\_to\_tfcs2 as a left join mandatory as b

on a.tfc=b.tfc;**quit**;

**data** hrgs\_to\_hrgs\_mandatory2; set hrgs\_to\_hrgs\_mandatory; if (FUS=**1** or fum=**1** or fas=**1** or fam=**1**) then Mandatory="Yes";

else mandatory="No"; **run**;

**proc** **sql**; create table hrgs\_to\_tfcs2 as select

tfc, sum(WF01A) as SumOfWF01A\_Activity, sum(WF01B) as SumOfWF01B\_Activity,

sum(WF02A) as SumOfWF02A\_Activity, sum(WF02B) as SumOfWF02B\_Activity,

sum(WF01A)+sum(WF01B)+sum(WF02A)+sum(WF02B) as SumOfTotal\_Activity, Mandatory from hrgs\_to\_hrgs\_mandatory2

GROUP BY TFC, mandatory;

**quit**;

/\* validation check \*/

**data** test1; set hrgs\_to\_tfcs2; where tfc notin("199" "264" "343" "424"); **run**;

**data** test1b; set hrgs\_to\_tfcs2; where tfc in("199" "264" "343" "424"); **run**;

**proc** **sql**; create table hrgs\_to\_tfcs2\_pct\_split as select \*, sum(sumoftotal\_activity) as totalactivity,

sumofWF01A\_activity/sum(sumoftotal\_activity) as FUS\_impact, sumofWF01B\_activity/sum(sumoftotal\_activity) as FAS\_impact,

sumofWF02A\_activity/sum(sumoftotal\_activity) as FUM\_impact, sumofWF02B\_activity/sum(sumoftotal\_activity) as FAM\_impact from hrgs\_to\_tfcs2

where tfc notin("199" "264" "343" "424"); **quit**;/\* alternative approach should be used to eliminate these TFCs \*/

**data** hrgs\_to\_tfcs\_pct\_split2;

set hrgs\_to\_tfcs2\_pct\_split;

rename sumofWF02B\_activity=FAM;

rename sumofWF01B\_activity=FAS;

rename sumofWF02A\_activity=FUM;

rename SumOfWF01A\_Activity=FUS;

FORMAT FAM\_impact FAS\_impact FUM\_impact FUS\_impact percent7.1;

format SumOfTotal\_Activity totalactivity comma10.0;

**run**;

**data** hrgs\_to\_tfcs\_pct\_split; set hrgs\_to\_tfcs\_pct\_split2;

format FAM FAS FUM FUS comma10.0;**run**;

/\* validation check \*/

**data** test2; set hrgs\_to\_tfcs\_pct\_split; where tfc notin("199" "264" "343" "424"); **run**;

**proc** **sql**; create table allocate\_oproc\_cost3 as

select a.tariff\_tfc\_code as TFC, a.CL\_FAM/b.CL\_FAM as CL\_FAM\_uc, a.CL\_FAS/b.CL\_FAS as CL\_FAS\_uc,

a.CL\_FUM/b.CL\_FUM as CL\_FUM\_uc, a.CL\_FUS/b.CL\_FUS as CL\_FUS\_uc from output\_OPATT\_FinalTariff as a

left join stage\_OPATT\_Activity\_Crosstab as b

on a.tariff\_tfc\_code=b.tariff\_tfc\_code;

**quit**;

**DATA** allocate\_oproc\_cost2; SET allocate\_oproc\_cost3;

if TFC="812" THEN cl\_fam\_uc=**0**; else cl\_fam\_uc=cl\_fam\_uc;

if TFC="812" THEN cl\_fas\_uc=**0**; else cl\_fas\_uc=cl\_fas\_uc;

if TFC="812" THEN cl\_fum\_uc=**0**; else cl\_fum\_uc=cl\_fum\_uc;

if TFC="812" THEN cl\_fus\_uc=**0**; else cl\_fus\_uc=cl\_fus\_uc;

**RUN**;

**proc** **sql**; create table allocate\_oproc\_cost as select \* from allocate\_oproc\_cost2 as a

left join hrgs\_to\_tfcs\_pct\_split as b

on a.tfc=b.tfc; **quit**;

**proc** **sql**; create table t3rd as select

"Yes" format=$3. as mandatory, sum(revised\_total\_act) as oproc\_act, SUM(REVISED\_TC) AS OPROC\_TC from t2nd;

**quit**;

**proc** **sql**; create table t4th as select

"No" format=$3. as mandatory, sum(revised\_total\_act) as oproc\_act, SUM(REVISED\_TC) AS OPROC\_TC from t2nd;

**quit**;

**proc** **sql**; create table t5th as select

"" format=$3. as mandatory, sum(revised\_total\_act) as oproc\_act, SUM(REVISED\_TC) AS OPROC\_TC from t2nd;

**quit**;

**data** t6th; set t3rd t4th t5th; **run**;

**proc** **sql**; create table allocate\_oproc\_cost4 as select a.\*, b.oproc\_act, B.OPROC\_TC from allocate\_oproc\_cost as a

full join t6th as b on a.mandatory=b.mandatory

order by TFC;

**quit**;

**proc** **sql**;

select distinct oproc\_act into :oproc\_act

from allocate\_oproc\_cost4;

**quit**;

**proc** **sql**;

select distinct oproc\_tc into :oproc\_tc

from allocate\_oproc\_cost4;

**quit**;

**data** allocate\_oproc\_cost5;

set allocate\_oproc\_cost4;

CL\_FAM\_act=FAM\_Impact\*&oproc\_act.;CL\_FAM\_act=ROUND(CL\_FAM\_act);

CL\_FAS\_act=FAS\_Impact\*&oproc\_act.;CL\_FAS\_act=ROUND(CL\_FAS\_act);

CL\_FUM\_act=FUM\_Impact\*&oproc\_act.;CL\_FUM\_act=ROUND(CL\_FUM\_act);

CL\_FUS\_act=FUS\_Impact\*&oproc\_act.;CL\_FUS\_act=ROUND(CL\_FUS\_act);

CL\_FAM\_TC=FAM\_Impact\*&oproc\_TC.;

CL\_FAS\_TC=FAS\_Impact\*&oproc\_TC.;

CL\_FUM\_TC=FUM\_Impact\*&oproc\_TC.;

CL\_FUS\_TC=FUS\_Impact\*&oproc\_TC.;

FORMAT CL\_FAM\_act CL\_FAS\_act CL\_FUM\_act CL\_FUS\_act /\*CL\_FAM\_TC CL\_FAS\_TC CL\_FUM\_TC CL\_FUS\_TC\*/ COMMA7.0;

**run**;

**DATA** allocate\_oproc\_cost5; SET allocate\_oproc\_cost5;

if TFC="812" THEN cl\_fam\_tc=**0**; else cl\_fam\_tc=cl\_fam\_tc;

if TFC="812" THEN cl\_fas\_tc=**0**; else cl\_fas\_tc=cl\_fas\_tc;

if TFC="812" THEN cl\_fum\_tc=**0**; else cl\_fum\_tc=cl\_fum\_tc;

if TFC="812" THEN cl\_fus\_tc=**0**; else cl\_fus\_tc=cl\_fus\_tc;

**RUN**;**proc** **sql**; create table allocate\_oproc\_cost6 as

select a.\*, coalesce(a.CL\_FAM\_act,**0**)+coalesce(b.CL\_FAM,**0**) as CL\_FAM\_act\_PostA, coalesce(a.CL\_FAS\_ACT,**0**)+coalesce(b.CL\_FAS,**0**) as CL\_FAS\_act\_PostA,

coalesce(a.CL\_FUM\_act,**0**)+coalesce(b.CL\_FUM,**0**) as CL\_FUM\_act\_PostA, coalesce(a.CL\_FUS\_ACT,**0**)+coalesce(b.CL\_FUS,**0**) as CL\_FUS\_act\_PostA,

coalesce(a.CL\_FAM\_tc,**0**)+coalesce(c.CL\_FAM,**0**) as CL\_FAM\_tc\_PostA, coalesce(a.CL\_FAS\_tc,**0**)+coalesce(c.CL\_FAS,**0**) as CL\_FAS\_tc\_PostA,

coalesce(a.CL\_FUM\_tc,**0**)+coalesce(c.CL\_FUM,**0**) as CL\_FUM\_tc\_PostA, coalesce(a.CL\_FUS\_tc,**0**)+coalesce(c.CL\_FUS ,**0**)as CL\_FUS\_tc\_PostA

from allocate\_oproc\_cost5 as a

left join stage\_OPATT\_Activity\_Crosstab as b

on a.tfc=b.tariff\_tfc\_code

left join output\_OPATT\_FinalTariff as c

on a.tfc=c.tariff\_tfc\_code;

**quit**;

**DATA** allocate\_oproc\_cost6; SET allocate\_oproc\_cost6;

if TFC="812" THEN CL\_FAM\_tc\_PostA=**0**; else CL\_FAM\_tc\_PostA=CL\_FAM\_tc\_PostA;

if TFC="812" THEN CL\_FAS\_tc\_PostA=**0**; else CL\_FAS\_tc\_PostA=CL\_FAS\_tc\_PostA;

if TFC="812" THEN CL\_FUM\_tc\_PostA=**0**; else CL\_FUM\_tc\_PostA=CL\_FUM\_tc\_PostA;

if TFC="812" THEN CL\_FUS\_tc\_PostA=**0**; else CL\_FUS\_tc\_PostA=CL\_FUS\_tc\_PostA;

**RUN**;

**data** allocate\_oproc\_cost6a; set allocate\_oproc\_cost6;

FORMAT CL\_FAM\_act\_PostA CL\_FAS\_act\_PostA CL\_FUM\_act\_PostA CL\_FUS\_act\_PostA

CL\_FAM\_TC\_PostA CL\_FAS\_TC\_PostA CL\_FUM\_TC\_PostA CL\_FUS\_TC\_PostA

COMMA12.0;

**run**;

**data** allocate\_oproc\_cost7;

set allocate\_oproc\_cost6a;

if CL\_FAM\_act\_PostA ne **0** then CL\_FAM\_UnitCost=CL\_FAM\_TC\_PostA/CL\_FAM\_act\_PostA; else CL\_FAM\_UnitCost=**0**;

if CL\_FAS\_act\_PostA ne **0** then CL\_FAS\_UnitCost=CL\_FAS\_TC\_PostA/CL\_FAS\_act\_PostA; else CL\_FAS\_UnitCost=**0**;

if CL\_FUM\_act\_PostA ne **0** then CL\_FUM\_UnitCost=CL\_FUM\_TC\_PostA/CL\_FUM\_act\_PostA; else CL\_FUM\_UnitCost=**0**;

if CL\_FUS\_act\_PostA ne **0** then CL\_FUS\_UnitCost=CL\_FUS\_TC\_PostA/CL\_FUS\_act\_PostA; else CL\_FUS\_UnitCost=**0**;

CL\_FAM\_UC\_CHANGE=CL\_FAM\_UNITCOST/CL\_FAM\_UC-**1**;

CL\_FAS\_UC\_CHANGE=CL\_FAS\_UNITCOST/CL\_FAS\_UC-**1**;

CL\_FUM\_UC\_CHANGE=CL\_FUM\_UNITCOST/CL\_FUM\_UC-**1**;

CL\_FUS\_UC\_CHANGE=CL\_FUS\_UNITCOST/CL\_FUS\_UC-**1**;

if mandatory="Yes" then MAND=**1**; else MAND=**0**;

FORMAT

CL\_FAM\_UnitCost CL\_FAS\_UnitCost CL\_FUM\_UnitCost CL\_FUS\_UnitCost COMMA12.0;

FORMAT CL\_FAM\_UC\_CHANGE CL\_FAS\_UC\_CHANGE CL\_FUM\_UC\_CHANGE CL\_FUS\_UC\_CHANGE PERCENT7.1;

**run**;

%let outputfile=allocate\_oproc\_cost7;

**proc** **export** data=&outputfile.

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

**data** front\_load(keep=TFC cl\_faS\_uc cl\_faM\_uc cl\_fuS\_uc cl\_fuM\_uc CL\_FAS\_act\_PostA CL\_FAM\_act\_PostA CL\_FUS\_act\_PostA CL\_FUM\_act\_PostA CL\_FAS\_tc\_PostA CL\_FAM\_tc\_PostA CL\_FUS\_tc\_PostA CL\_FUM\_tc\_PostA

CL\_FAS\_unitcost CL\_FAM\_unitcost CL\_FUS\_unitcost CL\_FUM\_unitcost);

set allocate\_oproc\_cost7;

**run**;

%let pct0=0;

%let pct10=10;%let pct20=20;%let pct30=30;%let pct40=40;%let pct50=50;

%let pct60=60;%let pct70=70;%let pct80=80;%let pct90=90;%let pct100=100;

**%macro** frontload(pct);

data &t\_name.\_&pct.pct;

set front\_load;

if CL\_FAS\_tc\_PostA ne **0** then cl\_faS\_initial\_frontload=**1**+(CL\_FUS\_tc\_PostA\*&pct./**100**)/CL\_FAS\_tc\_PostA; else cl\_faS\_initial\_frontload=**1**;

if CL\_FAM\_tc\_PostA ne **0** then cl\_faM\_initial\_frontload=**1**+(CL\_FUM\_tc\_PostA\*&pct./**100**)/CL\_FAM\_tc\_PostA; else cl\_faM\_initial\_frontload=**1**;

cl\_fuS\_initial\_frontload=**1**-(&pct./**100**);

cl\_fuM\_initial\_frontload=**1**-(&pct./**100**);

if cl\_faS\_initial\_frontload le &frontload\_cap. then cl\_faS\_revised\_frontload=cl\_faS\_initial\_frontload; else cl\_faS\_revised\_frontload=&frontload\_cap.;

if cl\_faM\_initial\_frontload le &frontload\_cap. then cl\_faM\_revised\_frontload=cl\_faM\_initial\_frontload; else cl\_faM\_revised\_frontload=&frontload\_cap.;

cl\_fuS\_revised\_frontload=cl\_fuS\_initial\_frontload;

cl\_fuM\_revised\_frontload=cl\_fum\_initial\_frontload;

CL\_FAS\_tc\_ResultantA=CL\_FAS\_tc\_PostA\*cl\_faS\_revised\_frontload;

CL\_FAM\_tc\_ResultantA=CL\_FAM\_tc\_PostA\*cl\_faM\_revised\_frontload;

CL\_FUS\_tc\_ResultantA=CL\_FUS\_tc\_PostA\*cl\_fuS\_revised\_frontload;

CL\_FUM\_tc\_ResultantA=CL\_FUM\_tc\_PostA\*cl\_fuM\_revised\_frontload;

DIFFERENCE\_IN\_TC=(CL\_FAM\_tc\_PostA+CL\_FAS\_tc\_PostA+CL\_FUM\_tc\_PostA+CL\_FUS\_tc\_PostA)/(CL\_FAM\_tc\_ResultantA+CL\_FAS\_tc\_ResultantA+CL\_FUM\_tc\_ResultantA+CL\_FUS\_tc\_ResultantA);

CL\_FAS\_tc\_FinalA=CL\_FAS\_tc\_ResultantA\*DIFFERENCE\_IN\_TC;

CL\_FAM\_tc\_FinalA=CL\_FAM\_tc\_ResultantA\*DIFFERENCE\_IN\_TC;

CL\_FUS\_tc\_FinalA=CL\_FUS\_tc\_ResultantA\*DIFFERENCE\_IN\_TC;

CL\_FUM\_tc\_FinalA=CL\_FUM\_tc\_ResultantA\*DIFFERENCE\_IN\_TC;

if CL\_FAS\_act\_PostA ne **0** then CL\_FAS\_UnitCost\_PostA=CL\_FAS\_tc\_FinalA/CL\_FAS\_act\_PostA; else CL\_FAS\_UnitCost\_PostA=**0**;

if CL\_FAM\_act\_PostA ne **0** then CL\_FAM\_UnitCost\_PostA=CL\_FAM\_tc\_FinalA/CL\_FAM\_act\_PostA; else CL\_FAM\_UnitCost\_PostA=**0**;

if CL\_FUS\_act\_PostA ne **0** then CL\_FUS\_UnitCost\_PostA=CL\_FUS\_tc\_FinalA/CL\_FUS\_act\_PostA; else CL\_FUS\_UnitCost\_PostA=**0**;

if CL\_FUM\_act\_PostA ne **0** then CL\_FUM\_UnitCost\_PostA=CL\_FUM\_tc\_FinalA/CL\_FUM\_act\_PostA; else CL\_FUM\_UnitCost\_PostA=**0**;

if CL\_FAS\_unitcost ne **0** then CL\_FAS\_UnitCost\_change=CL\_FAS\_UnitCost\_PostA/CL\_FAS\_unitcost-**1**; else CL\_FAS\_UnitCost\_change=**0**;

if CL\_FAM\_unitcost ne **0** then CL\_FAM\_UnitCost\_change=CL\_FAM\_UnitCost\_PostA/CL\_FAM\_unitcost-**1**; else CL\_FAM\_UnitCost\_change=**0**;

if CL\_FUS\_unitcost ne **0** then CL\_FUS\_UnitCost\_change=CL\_FUS\_UnitCost\_PostA/CL\_FUS\_unitcost-**1**; else CL\_FUS\_UnitCost\_change=**0**;

if CL\_FUM\_unitcost ne **0** then CL\_FUM\_UnitCost\_change=CL\_FUM\_UnitCost\_PostA/CL\_FUM\_unitcost-**1**; else CL\_FUM\_UnitCost\_change=**0**;

format cl\_fam\_initial\_frontload cl\_fas\_initial\_frontload cl\_fum\_initial\_frontload cl\_fus\_initial\_frontload

cl\_fam\_revised\_frontload cl\_fas\_revised\_frontload cl\_fum\_revised\_frontload cl\_fus\_revised\_frontload DIFFERENCE\_IN\_TC

CL\_FAM\_UnitCost\_change CL\_FAS\_UnitCost\_change CL\_FUM\_UnitCost\_change CL\_FUS\_UnitCost\_change percent7.1;

format CL\_FAM\_tc\_ResultantA CL\_FAS\_tc\_ResultantA CL\_FUM\_tc\_ResultantA CL\_FUS\_tc\_ResultantA

CL\_FAM\_tc\_FinalA CL\_FAS\_tc\_FinalA CL\_FUM\_tc\_FinalA CL\_FUS\_tc\_FinalA

CL\_FAM\_UnitCost\_PostA CL\_FAS\_UnitCost\_PostA CL\_FUM\_UnitCost\_PostA CL\_FUS\_UnitCost\_PostA Comma12.0;

Pct=&pct.;

run;

%let outputfile=&t\_name.\_&pct.pct;

**%mend**;

%***frontload***(&pct0.)

%***frontload***(&pct10.)

%***frontload***(&pct20.)

%***frontload***(&pct30.)

%***frontload***(&pct40.)

%***frontload***(&pct50.)

%***frontload***(&pct60.)

%***frontload***(&pct70.)

%***frontload***(&pct80.)

%***frontload***(&pct90.)

%***frontload***(&pct100.)

**data** new\_t;

set

front\_load\_0pct

front\_load\_10pct front\_load\_20pct front\_load\_30pct front\_load\_40pct front\_load\_50pct front\_load\_60pct front\_load\_70pct

front\_load\_80pct front\_load\_90pct front\_load\_100pct;

**run**;

**proc** **sort** data=new\_t out=new\_t2;

by tfc;

**run**;

**proc** **export** data=new\_t2

outfile="&outputpath.\new\_t2.csv"

dbms=csv

replace;

**run**;

**data** frontload\_options\_bytfc; set frontload\_options\_bytfc;

TFC2=put(TFC, **3.**);

drop TFC;

rename TFC2=TFC;

**run**;

**proc** **sql**; create table new\_t2b as select a.\*, b.option from new\_t2 as a

left join frontload\_options\_bytfc as b

on a.tfc=b.tfc;

**quit**;

**data** front\_loaded; set new\_t2b;

where option="A" and pct=&Option\_A\_pct. or

(option="B" and pct=&Option\_B\_pct.) or

(option="C" and pct=&Option\_C\_pct.)

OR (option="D" and pct=&Option\_D\_pct.)

;

**run**;

**proc** **export** data=front\_loaded

outfile="&outputpath.\front\_loaded.csv"

dbms=csv

replace;

**run**;

/\*--06c: Map DI HRGs to Groups\*/

/\*--Groups UBDI costs and activity for mapping\*/

/\*\*/

**proc** **sql**;

create table Stage\_OPROC01\_06c as

SELECT

in\_06b.DEPARTMENTcode,

in\_06b.SERVICEcode,

in\_04a.Group,

Sum(in\_06b.ACT) AS SumOfACT,

Sum(in\_06b.TC\_Target\_MFF) AS SumOfTC\_Target\_MFF

FROM

WORK.DI\_Group\_Mapping as in\_04a

INNER JOIN Stage\_OPROC01\_06b in\_06b

ON in\_04a.HRG = in\_06b.CURRENCYcode

GROUP BY

in\_06b.DEPARTMENTcode,

in\_06b.SERVICEcode,

in\_04a.Group;

**quit**;

/\*--06d: Select costs to be rebundled into OPROC HRGs\*/

/\*--Applies OPATT/OPROC split to DI costs to ensure not double-counted\*/

/\*\*/

**proc** **sql**;

create table Stage\_OPROC01\_06d as

SELECT

in\_06c.Group,

SumofACT\*PROC\_\_ AS ACT,

SumofTC\_Target\_MFF\*PROC\_\_ AS TC

FROM

Stage\_OPROC01\_06c in\_06c

INNER JOIN WORK.DI\_OP\_Split as in\_04c

ON in\_06c.Group = in\_04c.Group;

**quit**;

/\*--06e: Map DI group costs to HRGs\*/

/\*--Maps UBDI groups to OPROC HRGs\*/

/\*\*/

**proc** **sql**;

create table stage\_OPROC01\_06e as

SELECT

in\_04b.HRG,

Sum(TC\*proportion) AS OPROC\_DI\_COST

FROM

WORK.DI\_Distribution as in\_04b

INNER JOIN Stage\_OPROC01\_06d in\_06d

ON in\_04b.Group = in\_06d.Group

GROUP BY in\_04b.HRG;

**quit**;

/\*--06f: Rebundle DI costs into HRGs\*/

/\*--Rebundles mapped costs into HRGs\*/

**proc** **sql**;

create table Modelout.stage\_OPROC\_DI\_HRG\_1920 as

SELECT

\_04a.DEPARTMENTcode,

\_04a.SERVICEcode,

\_04a.CURRENCYcode,

\_04a.TOTAL\_ACT,

\_04a.TC,

in\_06e.OPROC\_DI\_COST,

coalesce(TC,**0**)+coalesce(OPROC\_DI\_Cost,**0**) AS REVISED\_TC

FROM

Modelout.OPROC\_National\_Data\_1920 \_04a

LEFT JOIN stage\_OPROC01\_06e in\_06e

ON \_04a.CURRENCYcode = in\_06e.HRG;

**quit**;

/\*LIBNAME inp\_upl ODBC DATASRC=Pbr1718\_\_28 SCHEMA=input;\*/

/\*--07: Calculate Prices\*/

/\*--Costs and activity from 06F and is adjusted for MFF Rescale and Uplifted\*/

/\*\*/

**proc** **sql**;

create table stage\_OPROC01\_07 as

SELECT

DH.DEPARTMENTcode,

DH.SERVICEcode,

DH.CURRENCYcode,

DH.TOTAL\_ACT,

DH.REVISED\_TC,

REVISED\_TC/TOTAL\_ACT AS UC, (REVISED\_TC/TOTAL\_ACT)\*MFF\_Rescale\*UPLIFT AS TARIFF

FROM

Stage\_OPROC\_MFF\_Rescale as out\_02,

WORK.Uplift as in\_99,

Modelout.stage\_OPROC\_DI\_HRG\_1920 as DH

ORDER BY

DH.DEPARTMENTcode,

DH.SERVICEcode,

DH.CURRENCYcode;

**quit**;

**Proc** **sql**;

create table Modelout.\_09NonMandHRGOPATT as

SELECT

FT.CURRENCYcode AS HRG,

FT.TOTAL\_ACT,

FT.REVISED\_TC AS TC

FROM WORK.HRG\_Eligibility AS HE INNER JOIN stage\_OPROC01\_07 AS FT ON HE.HRG = FT.CURRENCYcode

WHERE HE.outpatient\_procedure\_tariff=**0**;

**quit**;

**pROC** **SQL**;

CREATE TABLE output\_OPROC\_Final\_Tariff AS

SELECT

in\_07.CURRENCYcode AS HRG,

in\_07.TOTAL\_ACT,

in\_07.REVISED\_TC,

case when Zero\_HRG Is Null THEN TARIFF else **0** end as OPROC\_Tariff

FROM

work.HRG\_Eligibility as in\_02

INNER JOIN (work.HRGs\_zero\_price as in\_03

RIGHT JOIN stage\_OPROC01\_07 in\_07

ON in\_03.Zero\_HRG = in\_07.CURRENCYcode)

ON in\_02.HRG = in\_07.CURRENCYcode

WHERE

in\_02.outpatient\_procedure\_tariff=**1**

ORDER BY

in\_07.CURRENCYcode;

**quit**;

**data** ModelOut.output\_OPROC\_Final\_Tariff;

set output\_OPROC\_Final\_Tariff;

**run**;

**data** OPROC\_prices\_4\_IA\_model;

set output\_OPROC\_Final\_Tariff;

**run**;

**data** ModelOut.OPROC\_prices\_4\_IA\_model;

set OPROC\_prices\_4\_IA\_model;

**run**;

**proc** **export** data=OPROC\_prices\_4\_IA\_model

outfile="&outputpath.\Oproc\_output\_4\_IA.csv"

dbms=csv

replace;

**run**;

/\*--01: Filter for Scope\*/

/\*--Filter to remove PMS and non-NHS Data\*/

**proc** **sql**;

create table Stage\_OPROC\_ScopeData as

SELECT

in\_01b.organisationcode AS ORG\_CODE,

in\_01b.DEPARTMENTcode,

"OPROC" AS SERVICEcode,

in\_01b.CURRENCYcode,

Sum(UNITCOST\*in\_01b.ACTIVITY)/Sum(in\_01b.ACTIVITY) AS UC,

SUM(in\_01b.ACTIVITY) AS ACT,

SUM(UNITCOST\*ACTIVITY) AS TC

FROM

work.Reference\_Cost

as in\_01b

INNER JOIN WORK.tc\_mff as in\_00

ON in\_01b.organisationcode =

in\_00.RC\_Code

WHERE

((in\_01b.DEPARTMENTcode)="OPROC")

AND ((in\_01b.SUPPLIERTYPE) = "OWN")

GROUP BY

in\_01b.organisationcode,

in\_01b.DEPARTMENTcode,

/\*--"OPROC", \*/

in\_01b.CURRENCYcode

ORDER BY ORG\_CODE,DEPARTMENTcode,CURRENCYcode;

**quit**;

/\*--02: Remove MFF\*/

**Proc** **sql**;

create table Stage\_OPROC\_MFF as

SELECT

sd.ORG\_CODE,

sd.DEPARTMENTcode,

sd.SERVICEcode,

sd.CURRENCYcode,

sd.UC,

sd.TC,

UC/uncapped\_MFF AS UC\_Target\_MFF,

sd.ACT,

(TC/uncapped\_MFF) AS TC\_Target\_MFF,

(TC/in\_00.Capped\_MFF) AS TC\_Payment\_MFF

FROM

work.tc\_mff as in\_00

INNER JOIN Stage\_OPROC\_ScopeData as sd

ON in\_00.RC\_Code = sd.ORG\_CODE;

**quit**;

/\*--03a: National Averages\*/

/\*--Calculates national averages for use in "03b: Clean Data"\*/

/\*\*/

**PROC** **SQL**;

CREATE TABLE Stage\_OPROC\_National\_Avg AS

SELECT

DEPARTMENTCODE,

SERVICECODE,

CURRENCYCODE,

Sum(UC\_Target\_MFF\*ACT)/Sum(ACT) AS NA

FROM Stage\_OPROC\_MFF

GROUP BY

DEPARTMENTCODE,

SERVICECODE,

CURRENCYCODE;

**QUIT**;

/\*--03b: Clean Data\*/

/\*--Excludes any records with unit costs <0.05 or >20 times the national average\*/

%let Param\_OutlierMinValue = 0.05;

%let Param\_OutlierMaxValue = 20;

**proc** **sql**;

create table Stage\_OPROC\_Clean\_Data as

SELECT

mf.\* ,

CASE WHEN ((&param\_OutlierMinValue=**0** OR &param\_OutlierMinValue IS NULL) AND (&param\_OutlierMaxValue=**0** OR &param\_OutlierMaxValue IS NULL)) THEN **0**

ELSE

CASE WHEN UC\_Target\_MFF/NA<&param\_OutlierMinValue OR UC\_Target\_MFF/NA>&param\_OutlierMaxValue

THEN **1** ELSE **0** END

END /\*--Replaced value 0.05 with @param\_OutlierMinValue AND value 20 with @param\_OutlierMaxValue - 18/12/2014\*/ as Exclude

FROM

Stage\_OPROC\_MFF mf

INNER JOIN Stage\_OPROC\_National\_Avg na ON

(mf.CURRENCYcode = na.CURRENCYcode)

AND (mf.SERVICEcode = na.SERVICEcode)

AND (mf.DEPARTMENTcode = na.DEPARTMENTcode)

WHERE

(CASE WHEN ((&param\_OutlierMinValue=**0** OR &param\_OutlierMinValue IS NULL) AND (&param\_OutlierMaxValue=**0** OR &param\_OutlierMaxValue IS NULL)) THEN **0**

ELSE

CASE WHEN UC\_Target\_MFF/NA<&param\_OutlierMinValue OR UC\_Target\_MFF/NA>&param\_OutlierMaxValue THEN **1**

ELSE **0** END

END /\*--Replaced value 0.05 with @param\_OutlierMinValue AND value 20 with @param\_OutlierMaxValue - 18/12/2014\*/

) = **0**;

**quit**;

/\*--04a: National Data - Clean\*/

/\*--Aggregates "03b: Clean Data" to national HRG levels\*/

**proc** **sql**;

create table tariff\_r.OPROC\_National\_Data\_1920 as

SELECT

DEPARTMENTcode,

SERVICEcode,

CURRENCYcode,

Sum(ACT) AS TOTAL\_ACT,

Sum(UC\_Target\_MFF\*ACT) AS TC,

Sum(UC\_Target\_MFF\*ACT)/Sum(ACT) AS HRG\_UC\_Target\_MFF

FROM Stage\_OPROC\_Clean\_Data

GROUP BY

DEPARTMENTcode,

SERVICEcode,

CURRENCYcode;

**quit**;

/\*--04a: National Data - Clean\*/

/\*--Aggregates "03b: Clean Data" to national HRG levels\*/

**proc** **sql**;

create table Modelout.OPROC\_National\_Data\_1920 as

SELECT

DEPARTMENTcode,

SERVICEcode,

CURRENCYcode,

Sum(ACT) AS TOTAL\_ACT,

Sum(UC\_Target\_MFF\*ACT) AS TC,

Sum(UC\_Target\_MFF\*ACT)/Sum(ACT) AS HRG\_UC\_Target\_MFF

FROM Stage\_OPROC\_Clean\_Data

GROUP BY

DEPARTMENTcode,

SERVICEcode,

CURRENCYcode;

**quit**;

/\* PID added 20180830 FOR MPP CODE INPUT \*/

**data** tariff\_r.OPROC\_NATIONAL\_DATA\_1920;

set modelout.OPROC\_NATIONAL\_DATA\_1920;

**run**;

/\*--05a: National Average MFF\*/

/\*--Calculates national average Market Forces Factor value\*/

**proc** **sql**;

create table Stage\_OPROC\_National\_MFF as

SELECT

Sum(TC) AS INC\_MFF,

Sum(TC\_Target\_MFF) AS EXC\_MFF,

Sum(TC) / (Sum(TC\_Target\_MFF)) as nat\_avg

FROM Stage\_OPROC\_MFF mff;

**quit**;

/\*--05b: MFF Rescaling\*/

/\*--Rebases the 2010/11 MFF to take account of the capped 2013/14 MFF\*/

**proc** **sql**;

create table Stage\_OPROC\_MFF\_Rescale as

SELECT

Sum(TC\_Target\_MFF) AS Uncapped\_TC,

Sum(TC\_Payment\_MFF) AS Capped\_TC,

(Sum(TC\_Payment\_MFF)/Sum(TC\_Target\_MFF)) AS MFF\_Rescale

FROM Stage\_OPROC\_MFF;

**quit**;

/\*--06a: Outpatient DI Costs\*/

/\*--Filter outpatient diagnostic imaging data \*/

/\*\*/

**proc** **sql**;

create table stage\_OPROC01\_06a as

SELECT

in\_01b.organisationcode AS ORG\_CODE,

in\_01b.DEPARTMENTcode,

in\_01b.SERVICEcode,

in\_01b.CURRENCYcode,

in\_01b.UNITCOST AS UC,

in\_01b.ACTIVITY AS ACT,

UNITCOST\*ACTIVITY AS TC

FROM

work.Reference\_Cost as in\_01b

INNER JOIN work.tc\_mff as in\_00

ON in\_01b.organisationcode

= in\_00.RC\_Code

WHERE

in\_01b.DEPARTMENTcode ="IMAGOP"

AND in\_01b.SUPPLIERTYPE Like "OWN"; /\* --Removed PH 11/02/15 -- reset SPK 27/11/2015\*/

**quit**;

/\*--06b: Remove MFF from OPDI costs\*/

/\*--MFF removed to exclude location-specific uplifts\*/

/\*\*/

**proc** **sql**;

create table Stage\_OPROC01\_06b as

SELECT

in\_06a.ORG\_CODE,

in\_06a.DEPARTMENTcode,

in\_06a.SERVICEcode,

in\_06a.CURRENCYcode,

in\_06a.UC,

in\_06a.TC,

UC/Uncapped\_MFF AS UC\_Target\_MFF,

in\_06a.ACT,

(TC/Uncapped\_MFF) AS TC\_Target\_MFF,

(TC/Capped\_MFF) AS TC\_Payment\_MFF

FROM stage\_OPROC01\_06a in\_06a INNER JOIN

work.tc\_mff as in\_00

ON in\_06a.ORG\_CODE = in\_00.RC\_Code;

**quit**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

# \* APC\_Prerequisite\_Data.sas ;

\* date: August 2017 ;

\* description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

**%macro** ***check\_IncProvd***;

%if %sysfunc(exist(work.providerIn\_F)) %then %do;

proc datasets lib=work nodetails nolist;

delete providerIn\_F;

run;

%end;

%if %upcase(&InputProv.)=YES %then %do;

proc sql noprint;

Create view work.providerIn\_F as

select a.SHA\_Code,

a.SHA\_Cluster,

a.RC\_Code,

a.RC\_Name,

a.Tariff\_Code,

a.Cluster\_Code,

a.Cluster\_Name,

a.Target\_MFF,

a.Capped\_MFF\_201213,

a.Capped\_MFF\_201314

from work.providerIn\_MDS a

inner join

APCPARM.Provider as b

on a.RC\_Code =b.ProvCD

where upcase(b.Include)="YES"

and a.version\_id=&MDS\_refdata\_version\_id.

order by a.RC\_Code

;

quit;

%end;

%else %do;

data work.providerIn\_F /view=work.providerIn\_F;

set work.providerIn\_MDS;

run;

%end;

**%mend** check\_IncProvd;

**proc** **sql** noprint;

create table &lib.**.A**PC\_Refcost\_Unclean\_Flat(compress=yes) as

select

OrganisationCode as PROCODET,

DepartmentCode as POD,

ServiceCode as TFC,

CurrencyCode as HRG4P\_RC,

Unit\_Cost format=nlmnlgbp23.2,

Activity,

Bed\_days,

Mean format=nlmnlgbp23.2,

Actual\_Cost format=nlmnlgbp23.2,

Expected\_Cost format=nlmnlgbp23.2,

Mapping\_Pot,

SupplierType as SUPPLIER\_TYPE

from &RefCost\_Selected.

where DepartmentCode in ( "DC", "EL","EL\_XS", "NEL" ,"NEL\_XS" , "NES")

order by OrganisationCode

;

**quit**;

/\*Remove MFF \*/

**proc** **sql** noprint;

create table &lib.**.t**able\_remove\_MFF\_F as

select

b.Year,

a.PROCODET,

a.POD,

a.TFC,

a.HRG4P\_RC,

a.Unit\_Cost,

a.Activity,

a.Bed\_days,

a.Mean,

a.Actual\_cost,

a.Expected\_cost,

a.Mapping\_pot,

b.Target\_MFF,

b.Cluster\_Name,

(a.Unit\_Cost/b.Target\_MFF) as Un\_Cost\_RemMFF,

/\* (a.Unit\_Cost/b.Capped\_MFF) as Un\_Cost\_RemMFF,\*/

log(calculated Un\_Cost\_RemMFF) as Ln\_Un\_CostNoMFF,

substr(a.HRG4P\_RC,**1**,**1**) as Chapter,

substr(a.HRG4P\_RC,**1**,**2**) as SubChapter,

A.SUPPLIER\_TYPE

from work.APC\_Refcost\_Unclean\_Flat as a

inner join TC\_MFF\_Selected as b

/\*inner join &SQLInputLib..ProviderTable as b\*/

/\*inner join work.providerIn\_MDS as b\*/

on a.PROCODET=b.RC\_Code

;

**quit**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* Author: PID ;

\* Date: 16 Oct 2015 ;

# \* APC\_Grubbs\_Test.sas ;

\* Description: Finalises "Program APC Data Clean\_ Step\_1\_2 Do Grubbs Test SAS data.sas" ;

\* program ;

\* Implements iterative Grubbs testing to remove outliers ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

**%macro** doGrubbs(alpha=,inDS=,outDS=,key=,var=,min\_seg\_size=,flgName=);

\*sort variable that be will tested in ascending order for each key (segment);

proc sort data=&inDS. out=work.sorted\_inputDataset(compress=yes);

by &key. &var.;

run;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* determine the last variable in the key e.g. if key=HRG4P\_RC POD then the last variable is POD and add single ;

\* quotes around key varaibles and separate with comma for use by hash table code e.g. 'HRG4P\_RC','POD' ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

data \_null\_;

whitespace\_cnt=sum(count(compbl("&key.")," "),**1**); \*Assumption made that no variable can have spaces i.e. not in the form of 'Var WithSpace'n;

varlast=scan(compbl("&key."),whitespace\_cnt);

hashKey=cats("'",tranwrd("&key."," ","','"),"'");

call symputx('lastKeyVar',varlast,'L');

call symputx('hashKey',hashKey,'L');

run;

\*create a ranking(ordering number) for each key (segment);

data work.sorted\_inputDataset(compress=yes);

set work.sorted\_inputDataset;

by &key.;

if first.&lastKeyVar. then do;

\_\_order\_id=**1**;

end;

else do;

\_\_order\_id+**1**;

end;

run;

\*Get the stats for each per key (segment);

proc summary data=work.sorted\_inputDataset(keep=&key. &var.) nway missing;

by &key.;

var &var.;

output out=stats(where=(\_\_N>=&min\_seg\_size.)) sum=\_\_SUM mean=\_\_MEAN uss=\_\_USS n=\_\_N std=\_\_STD;

run;

\*add the critical value for each key (segment);

data work.all\_stats(drop=t2);

set work.stats;

t2=tinv(&alpha./(**2**\*\_\_N),\_\_N-**2**);

gcrit2=((\_\_N-**1**)/sqrt(\_\_N))\*sqrt(t2\*t2/(\_\_N-**2**+t2\*t2));

run;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Logic used to implement iterative Grubbes test \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* ;

\* load all records into memory using a hash table ;

\* do loop per key(segment) ;

\* get first(min) and last(max) observation from the hash table ;

\* get z-scores for these values ;

\* do Grubbs Test ;

\* IF record fails the test then do following: ;

\* (a) Output the outlier ;

\* (b) Recalculate stats ;

\* 1. output to outliers table ;

\* 2. N=SUM(N,-1) ;

\* 3. MEAN=SUM((MEAN\*sum(N,-1)),-OUTLIER)/N ;

\* 4. USS=USS-(outlier\*outlier) ;

\* 5. STD=((USS-(N\*MEAN\*MEAN))/(N-1))\*\*0.5 ;

\* (c) Remove outlier from Hash table ;

\* (d) Repeat process ;

\* ELSE pass then stop iteration for this key (segment) ;

\* ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

data work.Outliers\_Detected(keep=&key. \_\_order\_id);

attrib i length=**8**;

set work.all\_stats;

if \_n\_=**1** then do;

if **0** then set work.sorted\_inputDataset;

declare hash hData(dataset:"work.sorted\_inputDataset");

hData.defineKey(&hashKey.,'\_\_order\_id');

hData.defineData("&var.");

hData.defineDone();

end;

outlier\_flg=**1**;

upper\_order\_id=\_\_N;

lower\_order\_id=**1**;

i=**0**;

do until(outlier\_flg=**0**);

if \_\_N<&min\_seg\_size. then outlier\_flg=**0**;

outlier=**.**;

\_\_order\_id=upper\_order\_id;

rc=hData.find();

\_\_upper=&var.;

\_\_order\_id=lower\_order\_id;

rc=hData.find();

\_\_lower=&var.;

lower\_z\_score=abs(\_\_lower-\_\_MEAN)/\_\_STD;

upper\_z\_score=abs(\_\_upper-\_\_MEAN)/\_\_STD;

if lower\_z\_score > upper\_z\_score then do;

if lower\_z\_score > gcrit2 then do;

outlier=\_\_lower;

\_\_order\_id=lower\_order\_id;

rc=hData.remove();

lower\_order\_id=sum(lower\_order\_id,**1**);

output work.Outliers\_Detected;

end;

else do;

outlier\_flg=**0**;

end;

end;

else do;

if upper\_z\_score > gcrit2 then do;

outlier=\_\_upper;

\_\_order\_id=upper\_order\_id;

rc=hData.remove();

upper\_order\_id=sum(upper\_order\_id,-**1**);

output work.Outliers\_Detected;

end;

else do;

outlier\_flg=**0**;

end;

end;

if outlier\_flg=**1** then do;

\_\_N=SUM(\_\_N,-**1**);

if \_\_N=**2** then do;

outlier\_flg=**0**;

\*stop looking for ouliers as calculating the critical values uses 2 degrees of Freedom therefore cannot divide by 0;

end;

else do;

\_\_MEAN=SUM((\_\_MEAN\*sum(\_\_N,**1**)),-OUTLIER)/\_\_N;

\_\_USS=\_\_USS-(outlier\*outlier);

\_\_STD=((\_\_USS-(\_\_N\*\_\_MEAN\*\_\_MEAN))/(\_\_N-**1**))\*\***0.5**;

t2=tinv(&alpha./(**2**\*\_\_N),\_\_N-**2**);

gcrit2=((\_\_N-**1**)/sqrt(\_\_N))\*sqrt(t2\*t2/(\_\_N-**2**+t2\*t2));

end;

end;

end;

run;

proc sort data=work.Outliers\_Detected;

by &key. \_\_order\_id;

run;

data &outDS.(drop=\_\_order\_id rename=(HRG4P\_RC=HRG) compress=yes);

merge work.sorted\_inputDataset(in=a) work.Outliers\_Detected(in=b);

by &key. \_\_order\_id;

if b then &flgName.=**1**;

else &flgName.=**0**;

run;

\*do clean up to preserve disk space;

proc datasets lib=work nodetails nolist ;

delete stats;

delete all\_stats;

delete sorted\_inputDataset;

delete Outliers\_Detected;

quit;

**%mend** doGrubbs;

**%macro** applyGrubbs(doGrubbs=&vGrubbs.);

%if %sysfunc(exist(work.Data\_After\_Grubbs)) %then %do;

proc datasets lib=work nodetails nolist;

delete Data\_After\_Grubbs;

run;

%end;

%if &doGrubbs. = YES %then %do;

%***doGrubbs***(alpha=**0.05**,inDS=work.table\_remove\_MFF\_F,outDS=work.Data\_After\_Grubbs,key=HRG4P\_RC POD,var=Ln\_Un\_CostNoMFF,min\_seg\_size=**3**,flgName=flaggrubbs)

%end;

%else %do;

data work.Data\_After\_Grubbs/view=work.Data\_After\_Grubbs;

set work.table\_remove\_MFF\_F;

flaggrubbs=**0**;

rename HRG4P\_RC=HRG;

run;

%end;

**%mend** applyGrubbs;

%***applyGrubbs***()

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* Author: PID ;

\* Date: 15 Oct 2015 ;

# \* APC\_2550\_Test.sas ;

\* Description: Optimises "Program APC Data Clean\_Step\_1\_3 Do\_2550 data clean.sas" program ;

\* Algorithm: The 2550 flag is set to a value of 2 if an Organisation has 25 percentage ;

\* of it's records with unit costs (MFF removed) more than 1.5 times the ;

\* National average and 25 percentage of it's records with unit costs ;

\* (MFF removed) less than 0.5 times the National average. ;

\* Note: The units costs - MFF removed are at a HRG/POD grain. ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

**%macro** do2550Test(inDS=,outDS=,flgName=);

proc sort data=&inDS. out=work.sorted\_inDS(compress=yes);

by PROCODET HRG POD;

run;

\*Get the average unit cost with MFF removed at a HRG/POD grain and determine the number of records per organisation;

proc summary data=work.sorted\_inDS missing;

class PROCODET HRG POD;

var Un\_Cost\_RemMFF;

weight ACTIVITY;

output out=work.Av\_Unit\_Cost\_Summ(where=(\_type\_=**3**) drop=PROCODET \_FREQ\_) mean=average\_unit\_cost;

output out=work.Count\_Provider\_Summ(where=(\_type\_=**4**) keep=PROCODET count\_provider \_type\_) N=count\_provider;

run;

\*Determine the flg2550 for each Organisation as per above alogrithm;

data work.org\_with\_2550\_flg(keep=PROCODET flag2550 average\_unit\_cost);

set work.sorted\_inDS(keep=PROCODET HRG POD Un\_Cost\_RemMFF);

by PROCODET;

retain tot\_ab\_50pc tot\_bel\_50pc;

if \_n\_=**1** then do;

if **0** then set work.Av\_Unit\_Cost\_Summ(drop=\_type\_);

declare hash hDataAverage(dataset:"work.Av\_Unit\_Cost\_Summ");

hDataAverage.defineKey("HRG","POD");

hDataAverage.defineData("average\_unit\_cost");

hDataAverage.defineDone();

if **0** then set work.Count\_Provider\_Summ(drop=\_type\_);

declare hash hDataCount(dataset:"work.Count\_Provider\_Summ");

hDataCount.defineKey("PROCODET");

hDataCount.defineData("count\_provider");

hDataCount.defineDone();

end;

if first.PROCODET then do;

tot\_ab\_50pc=**0**;

tot\_bel\_50pc=**0**;

end;

rc=hDataAverage.find(); \*get the average\_unit\_cost for the HRG/POD;

if rc = **0** then do;

if Un\_Cost\_RemMFF > (**1.5**\*average\_unit\_cost) then tot\_ab\_50pc=sum(tot\_ab\_50pc,**1**);

else if Un\_Cost\_RemMFF < (**0.5**\*average\_unit\_cost) then tot\_bel\_50pc=sum(**1**,tot\_bel\_50pc);

end;

if last.PROCODET then do;

rc=hDataCount.find(); \*get the number of records for the organisation;

RATIO\_ab\_50pc\_TO\_record\_count=tot\_ab\_50pc/count\_provider;

RATIO\_bel\_50pc\_TO\_record\_count=tot\_bel\_50pc/count\_provider;

if RATIO\_ab\_50pc\_TO\_record\_count > **.25** and RATIO\_bel\_50pc\_TO\_record\_count >**.25** then do;

flag2550=**2**;

end;

else do;

flag2550=**0**;

end;

output;

end;

run;

\*Add the flg2550 to the main dataset;

data &outDS.(compress=yes);

merge work.sorted\_inDS (in=a) work.org\_with\_2550\_flg(in=b);

by PROCODET;

run;

\*do clean up to preserve disk space;

proc datasets lib=work nodetails nolist ;

delete Av\_Unit\_Cost\_Summ;

delete Count\_Provider\_Summ;

delete org\_with\_2550\_flg;

delete sorted\_inDS;

quit;

**%mend** do2550Test;

%***do2550Test***(inDS=work.Data\_After\_Grubbs,outDS=work.data\_GrabFlag\_2550Flag,flgName=flag2550)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* Author: PID ;

\* Updated: PID

\* Date: August 2017 ;

# \* APC\_Duplicates\_Test.sas ;

\* Description: Optimises "Program APC Data Clean\_Step\_1\_4 Create duplicate flags.sas" ;

\* program. ;

\* Updated: Added process to send the cleaned data with flags to IT. (Aug 2017) ;

\* Algorithm: The duplicate flag (flagduplicate) for an organisation is set to a value of ;

\* 4 if more than 75 percent of the organisations records have identical values;

\* for the POD x Service Code x Unit Cost with MFF removed, combination ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

**%macro** doDuplicateTest(inDS=,outDS=,flgName=);

proc sort data=&inDS. out=work.sorted\_inDS(compress=yes);

by PROCODET POD TFC Un\_Cost\_RemMFF;

run;

data work.org\_with\_duplicateFlag(keep=PROCODET &flgName. provider\_count dup\_percentage);

set work.sorted\_inDS;

by PROCODET POD TFC Un\_Cost\_RemMFF;

if first.PROCODET then do; \*New organisation - reset counters;

provider\_count=**0**;

tot\_dup\_count=**0**;

end;

if first.Un\_Cost\_RemMFF then do;

combination\_count=**0**; \*New combination for the organisation- reset counter;

end;

provider\_count+**1**; \*Count number of records per organisation;

combination\_count+**1**; \*Count number of records for the POD x TFC x Un\_Cost\_RemMFF combination for the organisation;

if last.Un\_Cost\_RemMFF then do;

if combination\_count > **1** then do; \*Encountered a duplicate combination of POD x TFC x Un\_Cost\_RemMFF for the organisation - increment counter;

tot\_dup\_count+combination\_count;

end;

end;

if last.PROCODET then do;

dup\_percentage=tot\_dup\_count/provider\_count;

if dup\_percentage > **0.75** then &flgName.=**4**;

else &flgName.=**0**;

output;

end;

run;

\*Merge duplicate flag and create bitwise clean\_flag;

proc sql noprint;

create table &outDS.(compress=yes) as

select

a.PROCODET as Org\_Code,

a.POD as Department\_code,

a.TFC as Service\_code,

a.HRG as Currency\_code,

a.Chapter,

a.SubChapter,

a.Unit\_Cost,

a.Activity,

a.Bed\_Days,

a.Mean,

a.Actual\_Cost,

a.Expected\_Cost,

a.Mapping\_Pot,

a.Target\_MFF,

a.Un\_Cost\_RemMFF,

a.Ln\_Un\_CostNoMFF,

a.Cluster\_Name as orgtype,

a.average\_unit\_cost,

a.flaggrubbs,

a.flag2550,

b.flagduplicate,

A.SUPPLIER\_TYPE, /\* PID 20160223 - ADDED \*/

sum (a.flaggrubbs,a.flag2550,b.flagduplicate) as clean\_flag

from work.sorted\_inDS a, work.org\_with\_duplicateFlag b

where a.PROCODET=b.PROCODET

;

quit;

\*do clean up to preserve disk space;

proc datasets lib=work nodetails nolist ;

delete org\_with\_duplicateFlag;

delete sorted\_inDS;

quit;

**%mend** doDuplicateTest;

%***doDuplicateTest***(inDS=work.data\_GrabFlag\_2550Flag,outDS=work.data\_GrabF\_2550F\_dupF\_cleanf,flgName=flagduplicate)

**PROC** **SQL**;

CREATE TABLE PPP AS SELECT \* FROM work.data\_GrabF\_2550F\_dupF\_cleanf

WHERE SERVICE\_CODE="100" AND CURRENCY\_CODE="JC43A" AND DEPARTMENT\_CODE="DC";

**QUIT**;

**proc** **sql** noprint;

create table work.Data\_for\_Matrix\_Engine as

select

Org\_Code,

Supplier\_type,

Department\_code,

upcase(Service\_code) as Service\_code,

Currency\_code,

Unit\_Cost,

Activity,

Bed\_Days,

Mean,

Actual\_Cost,

Expected\_Cost,

Mapping\_Pot,

clean\_flag as cleaning\_flag

from work.data\_GrabF\_2550F\_dupF\_cleanf

order by Org\_Code,Department\_code,Service\_code,Currency\_code

;

**quit**;

/\*Non Acute data \*/

**proc** **sql** noprint;

create table None\_APC\_Refcost\_cleanFlag as

select

OrganisationCode as Org\_Code,

SupplierType as Supplier\_type,

DepartmentCode as Department\_code,

upcase(ServiceCode) as Service\_code,

CurrencyCode,

Unit\_Cost format=nlmnlgbp23.2,

Activity ,

Bed\_days,

Mean format=nlmnlgbp23.2 ,

Actual\_cost format=nlmnlgbp23.2,

Expected\_cost format=nlmnlgbp23.2,

Mapping\_pot,

**0** as Cleaning\_flag

from &RefCost\_Selected.

where DepartmentCode not in ( "DC", "EL","EL\_XS", "NEL" ,"NEL\_XS" , "NES")

order by OrganisationCode,DepartmentCode,ServiceCode,CurrencyCode

;

**quit**;

**data** Tariff\_R.RefCostNonAcute\_flaged\_1920;

set None\_APC\_Refcost\_cleanFlag;

**run**;

/\*As same as PID cleaned refcost send to IT \*/

**data** RefCost\_data\_to\_infoserve\_1;

set Data\_for\_Matrix\_Engine

None\_APC\_Refcost\_cleanFlag

;

**run**;

**proc** **sql**;

create table RefCost\_data\_to\_infoserve(drop=Bed\_Days Rename=(BedDays=Bed\_Days))

as

select

Org\_Code,

Supplier\_type,

Department\_code,

Service\_code,

Currency\_code,

Unit\_Cost,

Activity,

case when

Bed\_Days=**.** then **0**

else Bed\_Days

end as BedDays,

Mean,

Actual\_Cost,

Expected\_Cost,

Mapping\_Pot,

cleaning\_flag

from RefCost\_data\_to\_infoserve\_1

;

**quit**;

**proc** **export** data=work.RefCost\_data\_to\_infoserve

outfile="&outputpath.\RefCost\_data\_to\_infoserve.csv"

dbms=csv

replace;

**run**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* Author: PID ;

\* Date: August 2017 ;

\* APC\_2550\_Test.sas ;

\* Description: Apply duplicate rules and create Flags ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

/\*Apply duplicate rules and create Flags \*/

/\*EL\_NEL\_EL\_XS\_NEL\_XS table \*/

**proc** **sql** noprint;

create table &lib.**.T**able\_EL\_NEL\_EL\_ELXS\_NELXS\_All(compress=yes) as

select

a.Org\_Code,

a.Supplier\_type,

a.Currency\_code,

a.Department\_code,

a.Service\_code,

a.Unit\_Cost ,

a.Un\_Cost\_RemMFF,

a.Activity ,

a.Bed\_Days,

a.flaggrubbs ,

a.flag2550,

a.flagduplicate

from

&lib.**.d**ata\_GrabF\_2550F\_dupF\_cleanf a

where (a.Department\_code="EL" or a.Department\_code="EL\_XS" or a.Department\_code="NEL" or a.Department\_code="NEL\_XS")

and a.Supplier\_type="OWN"

order by a.Org\_Code,a.Currency\_code,a.Department\_code,service\_code

;

**quit**;

/\*Get data for EL and NEL only \*/

**proc** **sql** noprint;

create table &lib.**.T**able\_EL\_NEL\_only(compress=yes) as

select

a.Org\_Code,

a.Supplier\_type,

a.Currency\_code,

a.Department\_code,

a.Service\_code,

a.Unit\_Cost ,

a.Un\_Cost\_RemMFF,

a.Activity ,

a.Bed\_Days,

a.flaggrubbs ,

a.flag2550,

a.flagduplicate,

A.SUPPLIER\_TYPE /\* PID 20160223 - ADDED \*/

from

&lib.**.T**able\_EL\_NEL\_EL\_ELXS\_NELXS\_All a

where a.Department\_code="EL" or a.Department\_code="NEL"

order by a.Org\_Code,a.Currency\_code,a.Department\_code,service\_code

;

**quit**;

/\*Get data for EL\_XS and NEL\_XS only \*/

**proc** **sql** noprint;

create table &lib.**.T**able\_ELXS\_NELXS(compress=yes) as

select

a.Org\_Code,

a.Supplier\_type,

a.Currency\_code,

case

when a.Department\_code="EL\_XS" then "EL"

when a.Department\_code="NEL\_XS" then "NEL"

End as DepartmentCode,

a.Department\_code,

a.Service\_code,

a.Unit\_Cost as EBD\_unit\_cost, /\* changed August 2017 ---PID\*/

a.Activity as EBD\_act,

a.flaggrubbs as FlagGrubbsEBD,

a.flag2550 as Flag2550EBD,

a.flagduplicate as FlagDuplicatedEBD,

A.SUPPLIER\_TYPE /\* PID 20160223 - ADDED \*/

from

&lib.**.T**able\_EL\_NEL\_EL\_ELXS\_NELXS\_All a

where a.Department\_code="EL\_XS" or a.Department\_code="NEL\_XS"

order by a.Org\_Code,a.Currency\_code,DepartmentCode,service\_code

;

**quit**;

/\*EL\_NEL reshape \*/

**proc** **sql** noprint;

create table &lib.**.E**L\_NEL\_reshape(compress=yes)

as

select

a.Org\_Code,

a.Supplier\_type,

a.Currency\_code,

a.Department\_code,

a.Service\_code,

a.Unit\_Cost ,

a.Un\_Cost\_RemMFF,

a.Activity ,

a.Bed\_Days,

a.flaggrubbs ,

a.flag2550,

a.flagduplicate,

b.EBD\_unit\_cost,b.EBD\_act,b.FlagGrubbsEBD,b.Flag2550EBD,b.FlagDuplicatedEBD

from

&lib.**.T**able\_EL\_NEL\_only a

left join

&lib.**.T**able\_ELXS\_NELXS b

on a.Org\_code=b.Org\_code and

a.Department\_code=b.DepartmentCode and

a.Currency\_code=b.Currency\_code and

a.Service\_code=b.Service\_code

order by a.Org\_code, a.Department\_code, a.Service\_code,a.Currency\_code

;

**quit**;

/\*Get data for DC and NES only \*/

**proc** **sql** noprint;

create table &lib.**.T**able\_DC\_NES\_All(compress=yes) as

select

a.Org\_Code,

a.Supplier\_type,

a.Currency\_code,

a.Department\_code,

a.Service\_code,

a.Unit\_Cost ,

a.Un\_Cost\_RemMFF,

a.Activity ,

a.Bed\_Days,

a.flaggrubbs ,

a.flag2550,

a.flagduplicate,

A.SUPPLIER\_TYPE /\* PID 20160223 - ADDED \*/,

**0** as EBD\_unit\_cost,

**0** as EBD\_act,

**0** as FlagGrubbsEBD,

**0** as Flag2550EBD,

**0** as FlagDuplicatedEBD

from

&lib.**.d**ata\_GrabF\_2550F\_dupF\_cleanf a

where (a.Department\_code="DC" or a.Department\_code="NES" ) and a.Supplier\_type="OWN" /\*added August 2017 --PID \*/

order by a.Org\_Code,a.Currency\_code,a.Department\_code,service\_code

;

**quit**;

/\*Append EL\_NEL\_DC\_NES together \*/

**data** &lib.**.D**ata\_EL\_NEL\_DC\_NES(compress=yes);

set &lib.**.E**L\_NEL\_reshape;

**run**;

**proc** **append** base=&lib.**.D**ata\_EL\_NEL\_DC\_NES data=&lib.**.T**able\_DC\_NES\_All;

**run**;

**data** &lib.**.D**ata\_EL\_NEL\_DC\_NES\_flag\_for(compress=yes);

set &lib.**.D**ata\_EL\_NEL\_DC\_NES;

flag\_for\_grubbs=sum(flaggrubbs,flaggrubbsebd);

flag\_for\_2550=sum(flag2550,flag2550ebd);

flag\_for\_duplicate=sum(flagduplicate,FlagDuplicatedEBD);

**run**;

**proc** **sql** noprint;

create table count\_ALL\_before\_clean as

select count(\*) as count\_All\_before\_clean from

&lib.**.D**ata\_EL\_NEL\_DC\_NES\_flag\_for

;

**quit**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* Author: PID ;

\* Date: August 2017 ;

\* APC\_Data\_Clean\_Step\_Final\_data\_clean.sas ;

\* Description: ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\*Apply duplicate rules and create Flags;

\* Grubbs cleaning rule- removed if flag\_for\_grubbs>0;

\* PID - CHECKING FOR AND THEN REMOVING ROWS WHERE GRUBBS IS FLAGGED AS GREATER THAN 0;

**data** &lib.**.D**ata\_in\_removed\_FlagGrubbs(compress=yes);

set &lib.**.D**ata\_EL\_NEL\_DC\_NES\_flag\_for;

where flag\_for\_grubbs>**0**;

**run**;

**data** &lib.**.D**ata\_after\_removed\_FlagGrubbs(compress=yes);

set &lib.**.D**ata\_EL\_NEL\_DC\_NES\_flag\_for;

where flag\_for\_grubbs<=**0**;

**run**;

\*Report for Grubbs;

**proc** **sql** noprint;

create table &lib.**.R**pt\_DataClean\_Removed\_grubbs\_1(drop=POD2 compress=yes) as

select

substr(Department\_code,**1**,**2**) as POD2,

catx('\_',Currency\_code,calculated POD2) as HRG\_POD,

Currency\_code as HRG,

Department\_code as POD,

Activity,

Un\_Cost\_RemMFF

from

&lib.**.D**ata\_after\_removed\_FlagGrubbs

where POD<>'NEL\_XS' and POD<>'EL\_XS'

order by HRG\_POD,HRG,POD

;

**quit**;

**proc** **sql** noprint;

create table &lib.**.R**pt\_DataClean\_Removed\_grubbs(compress=yes) as

select HRG\_POD,

HRG,

POD,

sum(Activity) as Sum\_Activity,

sum(Un\_Cost\_RemMFF\*Activity)/sum(Activity) as Avg\_UCost\_RemMFF

from &lib.**.R**pt\_DataClean\_Removed\_grubbs\_1

group by HRG\_POD,HRG,POD

;

**quit**;

%let outputfile=Rpt\_DataClean\_Removed\_grubbs;

**proc** **export** data=&lib.**.**&outputfile.

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

\*2550 cleaning rule;

\* PID - CHECKING FOR AND THEN REMOVING ROWS WHERE 2550 IS FLAGGED AS GREATER THAN 0 FROM DATASET AFTER REMOVED FLAG GRUBBS;

**data** &lib.**.D**ata\_in\_removed\_Flag2550(compress=yes);

set &lib.**.D**ata\_after\_removed\_FlagGrubbs;

where flag\_for\_2550>**0**;

**run**;

**proc** **sql** noprint;

create table &lib.**.D**ata\_after\_removed\_Flag2550(compress=yes) as

select distinct(a.\*)

from

&lib.**.D**ata\_after\_removed\_FlagGrubbs a

where flag\_for\_2550<=**0**

order by Org\_Code, currency\_code

;

**run**;

\*Report for 2550;

**proc** **sql** noprint;

create table &lib.**.R**pt\_DataClean\_Removed\_2550\_1(drop=POD2 compress=yes) as

select

substr(Department\_code,**1**,**2**) as POD2,

catx('\_',Currency\_code,calculated POD2) as HRG\_POD,

Currency\_code as HRG,

Department\_code as POD,

Activity,

Un\_Cost\_RemMFF

from

&lib.**.D**ata\_after\_removed\_Flag2550

where POD<>'NEL\_XS' and POD<>'EL\_XS'

order by HRG\_POD,HRG,POD

;

**quit**;

**proc** **sql** noprint;

create table &lib.**.R**pt\_DataClean\_Removed\_2550(compress=yes) as

select HRG\_POD,

HRG,

POD,

sum(Activity) as Sum\_Activity,

sum(Un\_Cost\_RemMFF\*Activity)/sum(Activity) as Avg\_UCost\_RemMFF

from &lib.**.R**pt\_DataClean\_Removed\_2550\_1

group by HRG\_POD,HRG,POD

;

**quit**;

%let outputfile=Rpt\_DataClean\_Removed\_2550;

**proc** **export** data=&lib.**.**&outputfile.

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

\* PID - COUNT OF ROWS GROUPED BY ORG\_CODE, CURRENCY\_CODE AND DEPARTMENT\_CODE, AFTER REMOVED GRUBBS AND 2550 STAGE;

**proc** **sql** noprint;

create table count\_org\_code\_forFlag2550(compress=yes) as

select Org\_Code, currency\_code,Department\_code,count(\*) as count\_orgCode

from &lib.**.D**ata\_after\_removed\_FlagGrubbs

group by Org\_Code, currency\_code,Department\_code

order by Org\_Code, currency\_code,Department\_code

;

**quit**;

\* Duplicate cleaning Rule;

\* PID - CHECKING FOR AND THEN REMOVING ROWS WHERE DUPLICATES ARE FLAGGED ATER REMOVED FLAG 2550;

**data** &lib.**.D**ata\_in\_removed\_Flagduplicate(compress=yes);

set &lib.**.D**ata\_after\_removed\_Flag2550;

where flag\_for\_duplicate>**0**;

**run**;

**proc** **sql** noprint;

create table &lib.**.D**ata\_after\_removed\_Flagduplicate(compress=yes) as

select distinct(a.\*)

from

&lib.**.D**ata\_after\_removed\_Flag2550 a

where flag\_for\_duplicate<=**0**

order by Org\_Code, currency\_code,Department\_code

;

**run**;

\*Report for duplicate;

**proc** **sql** noprint;

create table &lib.**.R**pt\_DataClean\_Removed\_dup\_1(drop=POD2) as

select

substr(Department\_code,**1**,**2**) as POD2,

catx('\_',Currency\_code,calculated POD2) as HRG\_POD,

Currency\_code as HRG,

Department\_code as POD,

Activity,

Un\_Cost\_RemMFF

from

&lib.**.D**ata\_after\_removed\_Flagduplicate

where POD<>'NEL\_XS' and POD<>'EL\_XS'

order by HRG\_POD,HRG,POD

;

**quit**;

**proc** **sql** noprint;

create table &lib.**.R**pt\_DataClean\_Removed\_DUP(compress=yes) as

select HRG\_POD,

HRG,

POD,

sum(Activity) as Sum\_Activity,

sum(Un\_Cost\_RemMFF\*Activity)/sum(Activity) as Avg\_UCost\_RemMFF

from &lib.**.R**pt\_DataClean\_Removed\_dup\_1

group by HRG\_POD,HRG,POD

;

**quit**;

%let outputfile=Rpt\_DataClean\_Removed\_DUP;

**proc** **export** data=&lib.**.**&outputfile.

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

\* PID - COUNT OF ROWS GROUPED BY ORG\_CODE, CURRENCY\_CODE AND DEPARTMENT\_CODE, AFTER ALL CLEANING STAGES;

**proc** **sql** noprint;

create table count\_org\_code\_forFlagduplicate as

select Org\_Code, currency\_code,Department\_code,count(\*) as count\_orgCode

from &lib.**.D**ata\_after\_removed\_Flagduplicate

group by Org\_Code, currency\_code,Department\_code

order by Org\_Code, currency\_code,Department\_code

;

**quit**;

/\*Final\_Cleaned\_Refcost \*/

**proc** **sql** noprint;

create table &lib.**.F**inal\_APC\_Cleaned\_Refcost(compress=yes) as

select

"&Year." as YEARORDER,

Org\_Code as FK\_ORGS\_PROV\_ID,

Supplier\_type as SUPPLIER\_TYPE,

case

when Department\_code="EL" Then "EI"

when Department\_code="NEL" Then "NEI\_L"

when Department\_code="NES" Then "NEI\_S"

ELSE Department\_code

end as DEPARTMENT,

Service\_code as SERVICE,

Currency\_code as CURRENCY,

Unit\_Cost as UNIT\_COST,

Un\_Cost\_RemMFF,

Activity as FCE,

Bed\_Days as BED\_DAYS,

EBD\_unit\_cost as EXCESS\_BED\_DAYS\_UNIT\_COST,

EBD\_act as EXCESS\_BED\_DAYS\_ACTIVITY

from &lib.**.D**ata\_after\_removed\_Flagduplicate

order by CURRENCY,FK\_ORGS\_PROV\_ID

;

**quit**;

\* PID - CREATING NEW TABLE REASSIGNING BLANK VALUES FOR EXCESS BED DAYS UNIT COST TO 0 ;

**data** &lib.**.F**inal\_APC\_Cleaned\_Refcost\_F(compress=yes);

set &lib.**.F**inal\_APC\_Cleaned\_Refcost;

if EXCESS\_BED\_DAYS\_UNIT\_COST=**.** then EXCESS\_BED\_DAYS\_UNIT\_COST=**0**;

if EXCESS\_BED\_DAYS\_ACTIVITY=**.** then EXCESS\_BED\_DAYS\_ACTIVITY=**0**;

if BED\_DAYS=**.** then BED\_DAYS=**0**;

**run**;

**data** test1; set Final\_APC\_Cleaned\_Refcost\_F; where find(currency, "FE", "i"); **run**;

**data** tariff\_R.RefCostAcute\_1920;

set &lib.**.F**inal\_APC\_Cleaned\_Refcost\_F;

**run**;

\*Report for duplicate ;

**proc** **sql** noprint;

create table &lib.**.R**pt\_DataClean\_Removed\_ALL\_1(drop=POD2) as

select

substr(DEPARTMENT,**1**,**2**) as POD2,

catx('\_',CURRENCY,calculated POD2) as HRG\_POD,

Currency as HRG,

DEPARTMENT as POD,

FCE,

UNIT\_COST

from

&lib.**.F**inal\_APC\_Cleaned\_Refcost\_F

where POD<>'NEL\_XS' and POD<>'EL\_XS'

order by HRG\_POD,HRG,POD

;

**quit**;

**proc** **sql** noprint;

create table &lib.**.R**pt\_DataClean\_Removed\_ALL as

select HRG\_POD,

HRG,

POD,

sum(FCE) as Sum\_Activity,

sum(UNIT\_COST\*FCE)/sum(FCE) as Avg\_UCost\_RemMFF

from &lib.**.R**pt\_DataClean\_Removed\_ALL\_1

group by HRG\_POD,HRG,POD

;

**quit**;

%let outputfile=Rpt\_DataClean\_Removed\_ALL;

**proc** **export** data=&lib.**.**&outputfile.

outfile="&outputpath.\&outputfile..csv"

dbms=csv

replace;

**run**;

\* PID - COUNT OF THE ROWS IN THE TABLE ;

**proc** **sql** noprint;

create table count\_F as

select count(\*) as count\_Final from

&lib.**.F**inal\_APC\_Cleaned\_Refcost\_F

;

**quit**;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\* Author: PID ;

\* Date: 16 Oct 2015 ;

\* APC\_Cleaning\_Impact\_and\_Stats.sas ;

\* Description: ;

\* ;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

**proc** **summary** data=work.data\_GrabF\_2550F\_dupF\_cleanf missing;

var actual\_cost;

class Org\_Code flaggrubbs flag2550 flagduplicate;

output out=work.before\_clean(where=(\_type\_ in(**0** **8**)) rename=\_freq\_=activity\_b4\_clean)sum=quantum\_b4\_clean mean=mean\_cost std=std\_cost median=median\_cost q1=lower\_quartile q3=upper\_quartile;

output out=work.after\_grubbs(where=(flaggrubbs=**0** and \_type\_ in(**4** **12**)) rename=\_freq\_=activity\_post\_grubbs)sum=quantum\_post\_grubbs mean=mean\_cost std=std\_cost median=median\_cost q1=lower\_quartile q3=upper\_quartile;

output out=work.after\_2550(where=(flaggrubbs=**0** and flag2550=**0** and \_type\_ in(**6** **14**)) rename=\_freq\_=activity\_post\_2550)sum=quantum\_post\_2550 mean=mean\_cost std=std\_cost median=median\_cost q1=lower\_quartile q3=upper\_quartile;

output out=work.after\_dups(where=(flaggrubbs=**0** and flag2550=**0** and flagduplicate=**0**) rename=\_freq\_=activity\_post\_dups)sum=quantum\_post\_dups mean=mean\_cost std=std\_cost median=median\_cost q1=lower\_quartile q3=upper\_quartile;

**run**;

**data** work.activity(drop=flaggrubbs flag2550 flagduplicate \_type\_ quantum\_b4\_clean quantum\_post\_grubbs quantum\_post\_2550 quantum\_post\_dups mean\_cost std\_cost median\_cost lower\_quartile upper\_quartile);

merge work.before\_clean work.after\_grubbs work.after\_2550 work.after\_dups;

by org\_code;

if org\_code="" then org\_code="TOTAL";

**run**;

**data** work.quantum(drop=flaggrubbs flag2550 flagduplicate \_type\_ activity\_b4\_clean activity\_post\_grubbs activity\_post\_2550 activity\_post\_dups mean\_cost std\_cost median\_cost lower\_quartile upper\_quartile);

merge work.before\_clean work.after\_grubbs work.after\_2550 work.after\_dups;

by org\_code;

if org\_code="" then org\_code="TOTAL";

**run**;

**data** work.activity\_figures;

SET work.activity;

percentage\_activity\_REMOVED=(-((activity\_post\_dups/activity\_b4\_clean)-**1**));

if activity\_post\_dups in(**.** **0**) then percentage\_activity\_removed=**1**;

format percentage\_activity\_REMOVED percent7.2;

label

org\_code= 'Provider code'

activity\_b4\_clean= 'Number of observations in 13-14 raw RC data'

activity\_post\_grubbs= 'Number of observations in 13-14 RC data after applying the Grubbs cleaning rule'

activity\_post\_2550= 'Number of observations in 13-14 RC data after applying the 2550 cleaning rule'

activity\_post\_dups= 'Number of observations in 13-14 RC data after applying the duplicates cleaning rule'

percentage\_activity\_REMOVED= '% of data removed';

**RUN**;

**DATA** work.QUANTUM\_FIGURES;

SET work.QUANTUM;

PERCENTAGE\_QUANTUM\_REMOVED=(-((quantum\_post\_dups/quantum\_b4\_clean)-**1**));

if quantum\_post\_dups in(**.** **0**) then percentage\_quantum\_removed=**1**;

format PERCENTAGE\_QUANTUM\_REMOVED percent7.2;

label

ORG\_CODE = 'Provider code'

quantum\_b4\_clean = 'Total quantum in 13-14 raw RC data'

quantum\_post\_grubbs = 'Total quantum in 13-14 RC data after applying the Grubbs cleaning rule'

quantum\_post\_2550 = 'Total quantum in 13-14 RC data after applying the 2550 cleaning rule'

quantum\_post\_dups = 'Total quantumin 13-14 RC data after applying the duplicates cleaning rule'

PERCENTAGE\_QUANTUM\_REMOVED= '% of quantum removed';

**RUN**;

/\* 20151026 ADDED CODE BLOCK BELOW, ADDING COLUMN WITH MFF VALUES USED \*/

**proc** **sql**;

create table work.activity\_figures2 as

select a.\*, b.Target\_MFF from work.activity\_figures as a

left join providerIn\_MDS as b

on a.org\_code=b.RC\_Code

;

**quit**;

**proc** **export** data=work.activity\_figures2

outfile="&outputpath.\DataClean\_impact\_activity.csv"

dbms=csv

replace;

**run**;

**proc** **export** data=work.QUANTUM\_FIGURES

outfile="&outputpath.\DataClean\_impact\_QUANTUM.csv"

dbms=csv

replace;

**run**;

**Proc** **Sort** Data=work.data\_GrabF\_2550F\_dupF\_cleanf Out =work.after\_pod(compress=yes);

By currency\_code;

**Run**;

**proc** **summary** data=work.after\_pod missing;

var actual\_cost;

class currency\_code department\_code flaggrubbs flag2550 flagduplicate;

/\* by currency\_code;\*/

output out=work.before\_pod\_clean(where=(\_type\_ =**24**) rename=\_freq\_=activity\_b4\_clean)sum=quantum\_b4\_clean mean=mean\_cost std=std\_cost median=median\_cost q1=lower\_quartile q3=upper\_quartile;

output out=work.after\_pod\_grubbs(where=(flaggrubbs=**0** and \_type\_=**28**) rename=\_freq\_=activity\_post\_grubbs)sum=quantum\_post\_grubbs mean=mean\_cost std=std\_cost median=median\_cost q1=lower\_quartile q3=upper\_quartile;

output out=work.after\_pod\_2550(where=(flaggrubbs=**0** and flag2550=**0** and \_type\_=**30**) rename=\_freq\_=activity\_post\_2550)sum=quantum\_post\_2550 mean=mean\_cost std=std\_cost median=median\_cost q1=lower\_quartile q3=upper\_quartile;

output out=work.after\_pod\_dups(where=(flaggrubbs=**0** and flag2550=**0** and flagduplicate=**0** and \_type\_=**31**) rename=\_freq\_=activity\_post\_dups)sum=quantum\_post\_dups mean=mean\_cost std=std\_cost median=median\_cost q1=lower\_quartile q3=upper\_quartile;

**run**;

**data** work.quantum\_pod(drop=flaggrubbs flag2550 flagduplicate \_type\_ activity\_b4\_clean activity\_post\_grubbs activity\_post\_2550 activity\_post\_dups mean\_cost std\_cost median\_cost lower\_quartile upper\_quartile);

merge work.before\_pod\_clean work.after\_pod\_grubbs work.after\_pod\_2550 work.after\_pod\_dups;

by currency\_code department\_code;

**run**;

**PROC** **SQL**;

CREATE TABLE work.quantum\_pod2 as

SELECT DEPARTMENT\_CODE, currency\_code, quantum\_b4\_clean, quantum\_post\_grubbs, quantum\_post\_2550, quantum\_post\_dups FROM work.QUANTUM\_POD;

**QUIT**;

**DATA** work.QUANTUM\_POD\_FIGURES;

SET work.QUANTUM\_POD2;

PERCENTAGE\_QUANTUM\_REMOVED=(-((quantum\_post\_dups/quantum\_b4\_clean)-**1**));

if quantum\_post\_dups in(**.** **0**) then percentage\_quantum\_removed=**1**;

format PERCENTAGE\_QUANTUM\_REMOVED percent7.2;

label

DEPARTMENT\_CODE = 'Department code'

HRG = 'HRG'

quantum\_b4\_clean = 'Total quantum in 13-14 raw RC data'

quantum\_post\_grubbs = 'Total quantum in 13-14 RC data after applying the Grubbs cleaning rule'

quantum\_post\_2550 = 'Total quantum in 13-14 RC data after applying the 2550 cleaning rule'

quantum\_post\_dups = 'Total quantumin 13-14 RC data after applying the duplicates cleaning rule'

PERCENTAGE\_QUANTUM\_REMOVED= '% of quantum removed';

**RUN**;

**proc** **export** data=work.QUANTUM\_POD\_figures

outfile="&outputpath.\DataClean\_impact\_QUANTUM\_POD.csv"

dbms=csv

replace;

**run**;

**proc** **summary** data=work.data\_GrabF\_2550F\_dupF\_cleanf missing;

var actual\_cost;

class Currency\_code flaggrubbs flag2550 flagduplicate;

output out=work.stats\_b4clean\_HRG(where=(\_type\_ in(**0** **8**)) rename=\_freq\_=activity\_b4\_clean)sum=quantum\_b4\_clean mean=mean\_cost std=std\_cost median=median\_cost q1=lower\_quartile q3=upper\_quartile;

output out=work.stats\_postgrubbs\_HRG(where=(flaggrubbs=**0** and \_type\_ in(**4** **12**)) rename=\_freq\_=activity\_post\_grubbs)sum=quantum\_post\_grubbs mean=mean\_cost std=std\_cost median=median\_cost q1=lower\_quartile q3=upper\_quartile;

output out=work.stats\_post2550\_HRG(where=(flaggrubbs=**0** and flag2550=**0** and \_type\_ in(**6** **14**)) rename=\_freq\_=activity\_post\_2550)sum=quantum\_post\_2550 mean=mean\_cost std=std\_cost median=median\_cost q1=lower\_quartile q3=upper\_quartile;

output out=work.stats\_postdups\_HRG(where=(flaggrubbs=**0** and flag2550=**0** and flagduplicate=**0**) rename=\_freq\_=activity\_post\_dups)sum=quantum\_post\_dups mean=mean\_cost std=std\_cost median=median\_cost q1=lower\_quartile q3=upper\_quartile;

**run**;

**proc** **summary** data=work.data\_GrabF\_2550F\_dupF\_cleanf missing;

var actual\_cost;

class Department\_code flaggrubbs flag2550 flagduplicate;

output out=work.stats\_b4clean\_POD(where=(\_type\_ in(**0** **8**)) rename=\_freq\_=activity\_b4\_clean)sum=quantum\_b4\_clean mean=mean\_cost std=std\_cost median=median\_cost q1=lower\_quartile q3=upper\_quartile;

output out=work.stats\_postgrubbs\_POD(where=(flaggrubbs=**0** and \_type\_ in(**4** **12**)) rename=\_freq\_=activity\_post\_grubbs)sum=quantum\_post\_grubbs mean=mean\_cost std=std\_cost median=median\_cost q1=lower\_quartile q3=upper\_quartile;

output out=work.stats\_post2550\_POD(where=(flaggrubbs=**0** and flag2550=**0** and \_type\_ in(**6** **14**)) rename=\_freq\_=activity\_post\_2550)sum=quantum\_post\_2550 mean=mean\_cost std=std\_cost median=median\_cost q1=lower\_quartile q3=upper\_quartile;

output out=work.stats\_postdups\_POD(where=(flaggrubbs=**0** and flag2550=**0** and flagduplicate=**0**) rename=\_freq\_=activity\_post\_dups)sum=quantum\_post\_dups mean=mean\_cost std=std\_cost median=median\_cost q1=lower\_quartile q3=upper\_quartile;

**run**;

**proc** **summary** data=work.data\_GrabF\_2550F\_dupF\_cleanf missing;

var actual\_cost;

class Service\_code flaggrubbs flag2550 flagduplicate;

output out=work.stats\_b4clean\_Service(where=(\_type\_ in(**0** **8**)) rename=\_freq\_=activity\_b4\_clean)sum=quantum\_b4\_clean mean=mean\_cost std=std\_cost median=median\_cost q1=lower\_quartile q3=upper\_quartile;

output out=work.stats\_postgrubbs\_Service(where=(flaggrubbs=**0** and \_type\_ in(**4** **12**)) rename=\_freq\_=activity\_post\_grubbs)sum=quantum\_post\_grubbs mean=mean\_cost std=std\_cost median=median\_cost q1=lower\_quartile q3=upper\_quartile;

output out=work.stats\_post2550\_Service(where=(flaggrubbs=**0** and flag2550=**0** and \_type\_ in(**6** **14**)) rename=\_freq\_=activity\_post\_2550)sum=quantum\_post\_2550 mean=mean\_cost std=std\_cost median=median\_cost q1=lower\_quartile q3=upper\_quartile;

output out=work.stats\_postdups\_Service(where=(flaggrubbs=**0** and flag2550=**0** and flagduplicate=**0**) rename=\_freq\_=activity\_post\_dups)sum=quantum\_post\_dups mean=mean\_cost std=std\_cost median=median\_cost q1=lower\_quartile q3=upper\_quartile;

**run**;

**data** work.stats\_postdups\_org(drop=flaggrubbs flag2550 flagduplicate \_type\_);

set work.after\_dups;

if org\_code="" then org\_code="TOTAL";

**run**;

**data** work.stats\_postdups\_HRG2(drop=flaggrubbs flag2550 flagduplicate \_type\_);

set work.stats\_postdups\_HRG;

if CURRENCY\_code="" then CURRENCY\_code="TOTAL";

**run**;

**data** work.stats\_postdups\_POD2(drop=flaggrubbs flag2550 flagduplicate \_type\_);

set work.stats\_postdups\_POD;

if department\_code="" then department\_code="TOTAL";

**run**;

**data** work.stats\_postdups\_Service2(drop=flaggrubbs flag2550 flagduplicate \_type\_);

set work.stats\_postdups\_Service;

if service\_code="" then service\_code="TOTAL";

**run**;

**proc** **export** data=work.stats\_postdups\_org

outfile="&outputpath.\DataClean\_stats\_postdups\_org.csv"

dbms=csv

replace;

**run**;

**proc** **export** data=work.stats\_postdups\_HRG2

outfile="&outputpath.\DataClean\_stats\_postdups\_HRG.csv"

dbms=csv

replace;

**run**;

**proc** **export** data=work.stats\_postdups\_POD2

outfile="&outputpath.\DataClean\_stats\_postdups\_POD.csv"

dbms=csv

replace;

**run**;

**proc** **export** data=work.stats\_postdups\_Service2

outfile="&outputpath.\DataClean\_stats\_postdups\_Service.csv"

dbms=csv

replace;

**run**;

**data** activity\_band\_calc;

set work.activity;

percentage\_activity=(-((activity\_post\_dups/activity\_b4\_clean)-**1**));

if activity\_post\_dups in(**.** **0**) then percentage\_activity=**1**;

**run**;

**data** activity\_band\_calc2;

set activity\_band\_calc;

where

org\_code^="TOTAL";

**run**;

**data** activity\_band;

length band $**30**;

format band $30.;

set activity\_band\_calc2;

if percentage\_activity=**0** then do band="0%"; order=**2**; end;

else if percentage\_activity gt **0** and percentage\_activity lt **.01** then do band="< 1%"; order=**3**; end;

else if percentage\_activity ge **.01** and percentage\_activity lt **.02** then do band="1-2%"; order=**4**; end;

else if percentage\_activity ge **.02** and percentage\_activity lt **.03** then do band="2-3%"; order=**5**; end;

else if percentage\_activity ge **.03** and percentage\_activity lt **.04** then do band="3-4%"; order=**6**; end;

else if percentage\_activity ge **.04** and percentage\_activity lt **.05** then do band="4-5%"; order=**7**; end;

else if percentage\_activity ge **.05** and percentage\_activity lt **.06** then do band="5-6%"; order=**8**; end;

else if percentage\_activity ge **.06** and percentage\_activity lt **.07** then do band="6-7%"; order=**9**; end;

else if percentage\_activity ge **.07** and percentage\_activity lt **.08** then do band="7-8%"; order=**10**; end;

else if percentage\_activity ge **.08** and percentage\_activity lt **.09** then do band="8-9%"; order=**11**; end;

else if percentage\_activity ge **.09** and percentage\_activity lt **.10** then do band="9-10%"; order=**12**; end;

else if percentage\_activity ge **.1** and percentage\_activity lt **.20** then do band="10-19%"; order=**13**; end;

else if percentage\_activity ge **.2** and percentage\_activity lt **.30** then do band="20-29%"; order=**14**; end;

else if percentage\_activity ge **.3** and percentage\_activity lt **.40** then do band="30-39%"; order=**15**; end;

else if percentage\_activity ge **.4** and percentage\_activity lt **.50** then do band="40-49%"; order=**16**; end;

else if percentage\_activity ge **.5** and percentage\_activity lt **.60** then do band="50-59%"; order=**17**; end;

else if percentage\_activity ge **.6** and percentage\_activity lt **.70** then do band="60-69%"; order=**18**; end;

else if percentage\_activity ge **.7** and percentage\_activity lt **.80** then do band="70-79%"; order=**19**; end;

else if percentage\_activity ge **.8** and percentage\_activity lt **.90** then do band="80-89%"; order=**20**; end;

else if percentage\_activity ge **.9** and percentage\_activity lt **1** then do band="90-99%"; order=**21**; end;

else if percentage\_activity ge **1** then do band="100%"; order=**22**; end;

else if percentage\_activity=**.** then do band="error"; order=**1**; end;

**run**;

**data** activity\_banding;

set work.activity\_band;

if band ="" then do band="0%"; order=**2**; end;

**run**;

**PROC** **FORMAT**;

VALUE typefmt

**1**="error"

**2**="0%"

**3**="< 1%"

**4**="1-2%"

**5**="2-3%"

**6**="3-4%"

**7**="4-5%"

**8**="5-6%"

**9**="6-7%"

**10**="7-8%"

**11**="8-9%"

**12**="9-10%"

**13**="10-19%"

**14**="20-29%"

**15**="30-39%"

**16**="40-49%"

**17**="50-59%"

**18**="60-69%"

**19**="70-79%"

**20**="80-89%"

**21**="90-99%"

**22**="100%"

;

**RUN**;

**PROC** **FREQ** DATA =work.activity\_banding noprint;

TABLES order/nocum out=work.dataclean\_activityband(rename=order=percentage\_change);

FORMAT order typefmt.;

**RUN**;

**proc** **export** data=work.dataclean\_activityband

outfile="&outputpath.\DataClean\_Impact\_activity\_banding.csv"

dbms=csv

replace;

**run**;

**data** quantum\_band\_calc;

set work.QUANTUM;

PERCENTAGE\_QUANTUM=(-((quantum\_post\_dups/quantum\_b4\_clean)-**1**));

if quantum\_post\_dups in(**.** **0**) then percentage\_quantum=**1**;

**run**;

**data** quantum\_band\_calc2;

set quantum\_band\_calc;

where

org\_code^="TOTAL";

**run**;

**data** quantum\_band;

length band $**30**;

format band $30.;

set quantum\_band\_calc2;

if percentage\_quantum ge **0** and percentage\_quantum le **0.000000000000001** then do band="0%"; order=**2**; end;

else if percentage\_quantum gt **0.000000000000001** and percentage\_quantum lt **.01** then do band="< 1%"; order=**3**; end;

else if percentage\_quantum ge **.01** and percentage\_quantum lt **.02** then do band="1-2%"; order=**4**; end;

else if percentage\_quantum ge **.02** and percentage\_quantum lt **.03** then do band="2-3%"; order=**5**; end;

else if percentage\_quantum ge **.03** and percentage\_quantum lt **.04** then do band="3-4%"; order=**6**; end;

else if percentage\_quantum ge **.04** and percentage\_quantum lt **.05** then do band="4-5%"; order=**7**; end;

else if percentage\_quantum ge **.05** and percentage\_quantum lt **.06** then do band="5-6%"; order=**8**; end;

else if percentage\_quantum ge **.06** and percentage\_quantum lt **.07** then do band="6-7%"; order=**9**; end;

else if percentage\_quantum ge **.07** and percentage\_quantum lt **.08** then do band="7-8%"; order=**10**; end;

else if percentage\_quantum ge **.08** and percentage\_quantum lt **.09** then do band="8-9%"; order=**11**; end;

else if percentage\_quantum ge **.09** and percentage\_quantum lt **.10** then do band="9-10%"; order=**12**; end;

else if percentage\_quantum ge **.1** and percentage\_quantum lt **.20** then do band="10-19%"; order=**13**; end;

else if percentage\_quantum ge **.2** and percentage\_quantum lt **.30** then do band="20-29%"; order=**14**; end;

else if percentage\_quantum ge **.3** and percentage\_quantum lt **.40** then do band="30-39%"; order=**15**; end;

else if percentage\_quantum ge **.4** and percentage\_quantum lt **.50** then do band="40-49%"; order=**16**; end;

else if percentage\_quantum ge **.5** and percentage\_quantum lt **.60** then do band="50-59%"; order=**17**; end;

else if percentage\_quantum ge **.6** and percentage\_quantum lt **.70** then do band="60-69%"; order=**18**; end;

else if percentage\_quantum ge **.7** and percentage\_quantum lt **.80** then do band="70-79%"; order=**19**; end;

else if percentage\_quantum ge **.8** and percentage\_quantum lt **.90** then do band="80-89%"; order=**20**; end;

else if percentage\_quantum ge **.9** and percentage\_quantum lt **1** then do band="90-99%"; order=**21**; end;

else if percentage\_quantum ge **1** then do band="100%"; order=**22**; end;

else if percentage\_quantum=**.** then do band="error"; order=**1**; end;

**run**;

**data** work.quantum\_banding;

set quantum\_band;

if band ="" then do band="0%"; order=**2**; end;

**run**;

**PROC** **FREQ** DATA =work.quantum\_banding noprint;

TABLES order/nocum out=work.dataclean\_quantumband(rename=order=percentage\_change);

FORMAT order typefmt.;

**RUN**;

**proc** **export** data=work.dataclean\_quantumband

outfile="&outputpath.\DataClean\_Impact\_quantum\_banding.csv"

dbms=csv

replace;

**run**;

**data** pod\_band\_calc;

set work.QUANTUM\_POD2;

PERCENTAGE\_QUANTUM=(-((quantum\_post\_dups/quantum\_b4\_clean)-**1**));

if quantum\_post\_dups in(**.** **0**) then percentage\_quantum=**1**;

**run**;

**data** pod\_band;

length band $**30**;

format band $30.;

set pod\_band\_calc;

if percentage\_quantum ge **0** and percentage\_quantum le **0.000000000000001** then do band="0%"; order=**2**; end;

else if percentage\_quantum gt **0.000000000000001** and percentage\_quantum lt **.01** then do band="< 1%"; order=**3**; end;

else if percentage\_quantum ge **.01** and percentage\_quantum lt **.02** then do band="1-2%"; order=**4**; end;

else if percentage\_quantum ge **.02** and percentage\_quantum lt **.03** then do band="2-3%"; order=**5**; end;

else if percentage\_quantum ge **.03** and percentage\_quantum lt **.04** then do band="3-4%"; order=**6**; end;

else if percentage\_quantum ge **.04** and percentage\_quantum lt **.05** then do band="4-5%"; order=**7**; end;

else if percentage\_quantum ge **.05** and percentage\_quantum lt **.06** then do band="5-6%"; order=**8**; end;

else if percentage\_quantum ge **.06** and percentage\_quantum lt **.07** then do band="6-7%"; order=**9**; end;

else if percentage\_quantum ge **.07** and percentage\_quantum lt **.08** then do band="7-8%"; order=**10**; end;

else if percentage\_quantum ge **.08** and percentage\_quantum lt **.09** then do band="8-9%"; order=**11**; end;

else if percentage\_quantum ge **.09** and percentage\_quantum lt **.10** then do band="9-10%"; order=**12**; end;

else if percentage\_quantum ge **.1** and percentage\_quantum lt **.20** then do band="10-19%"; order=**13**; end;

else if percentage\_quantum ge **.2** and percentage\_quantum lt **.30** then do band="20-29%"; order=**14**; end;

else if percentage\_quantum ge **.3** and percentage\_quantum lt **.40** then do band="30-39%"; order=**15**; end;

else if percentage\_quantum ge **.4** and percentage\_quantum lt **.50** then do band="40-49%"; order=**16**; end;

else if percentage\_quantum ge **.5** and percentage\_quantum lt **.60** then do band="50-59%"; order=**17**; end;

else if percentage\_quantum ge **.6** and percentage\_quantum lt **.70** then do band="60-69%"; order=**18**; end;

else if percentage\_quantum ge **.7** and percentage\_quantum lt **.80** then do band="70-79%"; order=**19**; end;

else if percentage\_quantum ge **.8** and percentage\_quantum lt **.90** then do band="80-89%"; order=**20**; end;

else if percentage\_quantum ge **.9** and percentage\_quantum lt **1** then do band="90-99%"; order=**21**; end;

else if percentage\_quantum ge **1** then do band="100%"; order=**22**; end;

else if percentage\_quantum=**.** then do band="error"; order=**1**; end;

**run**;

**data** work.pod\_banding;

set pod\_band;

if band ="" then do band="0%"; order=**2**; end;

**run**;

**PROC** **FREQ** DATA =work.pod\_banding noprint;

TABLES order/nocum out=work.dataclean\_quantumPODband(rename=order=percentage\_change);

FORMAT order typefmt.;

**RUN**;

**proc** **export** data=work.dataclean\_quantumPODband

outfile="&outputpath.\DataClean\_Impact\_quantum\_POD\_banding.csv"

dbms=csv

replace;

**run**;

**proc** **datasets** lib=work nodetails nolist ;

delete before\_clean;

delete after\_grubbs;

delete after\_2550;

delete after\_dups;

delete activity;

delete activity\_figures;

delete QUANTUM\_FIGURES;

delete QUANTUM;

delete quantum\_pod;

delete quantum\_pod2;

delete QUANTUM\_POD\_figures;

delete before\_pod\_clean;

delete after\_pod\_2550;

delete after\_pod\_dups;

delete after\_pod\_grubbs;

delete stats\_b4clean\_hrg;

delete stats\_post2550\_hrg;

delete stats\_postdups\_hrg;

delete stats\_postgrubbs\_hrg;

delete stats\_b4clean\_pod;

delete stats\_post2550\_pod;

delete stats\_postdups\_pod;

delete stats\_postgrubbs\_pod;

delete stats\_b4clean\_Service;

delete stats\_postgrubbs\_Service;

delete stats\_post2550\_Service;

delete stats\_postdups\_Service;

delete stats\_postdups\_org;

delete stats\_postdups\_HRG2;

delete stats\_postdups\_POD2;

delete stats\_postdups\_Service2;

**quit**;

/\*--------------------------------------------------------------\*/

\* Create BPT Excel Model linksheets. \*/

/\* Version: 1.02 \*/

/\* Author: PID \*/

/\* Date: Feb 2018 \*/

/\*--------------------------------------------------------------\*/

/\* Description: \*/

/\* Generate an excel file which contains a linked sheet for all \*/

/\* BPT results: \*/

/\* 1.Acute stroke care; \*/

/\* 2.Adult renal dialysis; \*/

/\* 3.Day cases; \*/

/\* 4.Diabetic ketoacidosis and hypoglycaemia; \*/

/\* 5.Early inflammatory arthritis; \*/

/\* 6.Endoscopy procedures; \*/

/\* 7.Fragility hip fracture; \*/

/\* 8.Major trauma; \*/

/\* 9.Outpatient procedures; \*/

/\* 10.Paediatric diabetes year of care; \*/

/\* 11.Paediatric epilepsy; \*/

/\* 12.Parkinson's disease; \*/

/\* 13.Pleural effusion; \*/

/\* 14.Primary total hip and knee replacements; \*/

/\* 15.Same day emergency care; \*/

/\* 16.Transient ischaemic attack; \*/

/\* 17.Heart Failure; \*/

/\* 18.COPD Exacerbation; \*/

/\* 19.NSTEMI: Timely access to coronary angiography; \*/

/\* 20.Emergency Laparotomy;

This program needs to run after the Linked\_sheet\_v2\_TED\_integrated program because it is using their tables. \*/

/\*--------------------------------------------------------------\*/

/\*ODS to create Excel workbook output with multiple sheets\*/

ods \_all\_ close;

ods tagsets.ExcelXP path="&BPTExcelOutFolder." file="2019\_20 Best Practice Tariffs.xml" style=Printer;

ods tagsets.ExcelXP options(sheet\_name='BPT Linked\_Sheet' sheet\_interval = 'NONE'

autofilter='NO' embedded\_titles ="yes" embedded\_footnotes='yes' autofit\_height ="yes" ascii\_dots="no"

absolute\_column\_width='10,60,20,20,20,20,20,20,20,10,10,10,10,10'

);

ods escapechar='^';

/\*Report header --Start from here \*/

**Data** text;

input Empty $8. Text & $100.;

datalines;

2019-20 tariff - Best Practice Tariffs

;

**run**;

**proc** **report** data=text nowd noheader

style(column)={font\_weight=bold font\_size=**14**pt JUST=c};

column Empty text;

**run**;

/\*Report header --End here \*/

title2 JUSTIFY=left '1. Acute stroke care ';

title4 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "The BPT is made up of four components: a base tariff and three conditional top-up payments.The base tariff is payable to all activity irrespective of whether the characteristics of best practice were met.";

title5 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "The three conditional payments are conditional on meeting best practice characteristics described in the guidance. ";

title6 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "The base tariff and the conditional top-up payment apply at the HRG level. ";

title7 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "Short stay emergency adjustment does not apply. Long stay payments and specialist top-ups apply where appropriate. ";

title8 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "SUS PbR will apply the base tariff to the HRGs below. SUS PbR will only apply the conditional payment for alteplase (where coded) and not those for the other two characteristics of best practice (see guidance). ";

title9 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "The alteplase conditional payment is payable for all spells with core HRGs listed below and unbundled HRG XD07Z.";

title10 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" " ";

/\* Table 1.Acute stroke care \*/

\*Title "1.Acute stroke care";

Title10 j=l "1a. Best practice tariff additional payments";

**proc** **print** data = BPTOut.LNK\_1A\_STROKE\_TABLE1 noobs label split= '\*' ;

var ID Best\_practice\_tariff\_add\_p

/style(header)=[background=#b3ffff];

var additional\_payment/style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###"];

label

Best\_practice\_tariff\_add\_p="Best practice tariff additional payments"

additional\_payment="Additional payments (£)"

;

**run**;

title2;

title4;

title5;

title6;

title7;

title8;

title9;

Title10 j=l "1b. Non-best practice tariff payments";

**proc** **print** data = BPTOut.LNK\_1B\_STROKE\_TABLE2 noobs label split= '\*' ;

var HRG\_code HRG\_name /style(header)=[background=#b3ffff];

var Non\_best\_practice\_tariff Best\_practice\_tariff/style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###"];

var BPT\_Flag /style(header)=[background=#b3ffff];

label

HRG\_code="HRG code"

HRG\_name="HRG name"

Non\_best\_practice\_tariff="Non-best practice tariff (£)"

Best\_practice\_tariff="Best practice tariff (£) \*(excl. alteplase)"

BPT\_Flag="BPT Flag \*(see BPT Flag sheet)"

;

**run**;

/\* Table 2.Adult renal dialysis \*/

Title2 j=l "2. Adult renal dialysis";

title5 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "The tariff prices apply at the HRG level.";

title6 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "The BPTs are set to encourage the adoption of clinical best practice with respect to vascular access. ";

title7 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "Please note that the mandatory tariffs for Peritoneal Dialysis are not classed as BPTs, however they are included in this sheet for ease of reference.";

title8 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "The HRGs are generated by data items from the National Renal Dataset, listed in the guidance. Organisations need to establish local reporting processes to support payment of the tariffs..";

title9 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" " ";

Title10 j=l "2a. Haemodialysis\_Hospital";

**proc** **print** data = BPTOut.lnk\_2a1\_renal\_hemod\_hospital\_t1 noobs label split= '\*' ;

var HRG\_Code HRG\_Name /style(header)=[background=#b3ffff];

var Best\_practice\_tariff /style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###"];

label

HRG\_Code="HRG code"

HRG\_Name="HRG\_Code name"

Best\_practice\_tariff="Best practice tariff (£)\*(per session)";

**run**;

Title2;

Title3 j=l "2a. Haemodialysis\_Satellite";

**proc** **print** data = BPTOut.LNK\_2A2\_RENAL\_HEMOD\_SATELLITE\_T2 noobs label split= '\*' ;

var HRG\_Code HRG\_Name /style(header)=[background=#b3ffff];

var Best\_practice\_tariff /style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###"];

label

HRG\_Code="HRG code"

HRG\_Name="HRG\_Code name"

Best\_practice\_tariff="Best practice tariff (£)\*(per session)";

**run**;

Title3 j=l "2a. Haemodialysis\_Home";

**proc** **print** data = BPTOut.LNK\_2A3\_RENAL\_HEMOD\_HOME\_T3 noobs label split= '\*' ;

var HRG\_Code HRG\_Name /style(header)=[background=#b3ffff];

var Best\_practice\_tariff /style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###"];

label

HRG\_Code="HRG code"

HRG\_Name="HRG\_Code name"

Best\_practice\_tariff="Best practice tariff (£)\*(per week)";

**run**;

Title3 j=l "2b. Peritoneal dialysis";

**proc** **print** data = BPTOut.LNK\_2B\_RENAL\_PERITONEAL\_T4 noobs label split= '\*' ;

var hrg\_code HRG\_name

/style(header)=[background=#b3ffff];

var Price

/style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###"];

label

HRG\_Code="HRG code"

HRG\_Name="HRG\_Code name"

Price="Price (£) \* (Per day)";

**run**;

/\*Table 3.Day cases \*/

Title2 j=l "3. Day cases";

**proc** **print** data = BPTOut.LNK\_3\_DAY\_CASE noobs label split= '\*' ;

var hrg\_code HRG\_name Surgical\_SubSpeciality Procedure /style(header)=[background=#b3ffff];

var Best\_practice\_day\_case\_tariff Elective\_nonbestpractice\_tariff

/style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###"];

var HRG\_or\_sub\_HRG\_level BPT\_Flag/style(header)=[background=#b3ffff];

label

HRG\_Code="HRG code"

HRG\_Name="HRG\_Code name"

Surgical\_SubSpeciality="Surgical sub speciality"

Procedure="Procedure"

Best\_practice\_day\_case\_tariff="Best practice \*day case tariff (£)"

Elective\_nonbestpractice\_tariff="Elective non-best \*practice tariff (£)"

HRG\_or\_sub\_HRG\_level="BPT applies to:\*HRG or sub-HRG level"

BPT\_Flag="BPT applies to:\*BPT Flag(see BPT Flag sheet)"

;

title5 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "The BPT for each procedure is made up of a pair of prices: one applied to day case admissions, the other to ordinary elective admissions.";

title6 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "For some procedures the BPT will apply to the HRG and for others it will apply at the Sub-HRG level with the use of a BPT flag to capture the relevant activity within the associated HRGs. ";

title7 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "Where the BPT applies at the Sub-HRG level, with the use of a BPT flag, there will also be a conventional tariff applicable to the HRG. The conventional tariff is to reimburse activity unrelated to the BPT within the same HRG. The BPT is mandatory for activity Specialist top-ups apply where appropriate.";

title8 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "SUS PbR will automate payment by: (a) generating the relevant flag, where required (b) applying relevant prices (either the BPT price or the conventional price).";

title9 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" " ";

FootNote1 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "\*Excision of breast includes quadrantectomy, partial excision, any other excision.";

**run**;

/\*Table 4.Diabetic ketoacidosis and hypoglycaemia\*/

Title2 j=l "4. Diabetic ketoacidosis and hypoglycaemia";

title5 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "The BPT will apply at the Sub-HRG level to emergency admissions.";

title6 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "SUS PbR will automate payment of the base tariff to activity eligible for the BPT, with conditional top-up for best practice to be paid locally subject to the relevant criteria being met.";

title7 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "As the BPT applies at the Sub-HRG level, with the use of a BPT flag, there will also be a conventional tariff applicable to the HRG. The conventional tariff is to reimburse activity unrelated to the BPT within the same HRG. The BPT is mandatory for activity identified by the flag and is therefore not optional. ";

FootNote1;

**proc** **print** data = BPTOut.LNK\_4\_DIABETE noobs label split= '\*' ;

var hrg\_code HRG\_name /style(header)=[background=#b3ffff];

var BPT\_Non\_elective\_tariff BPT\_Reduced\_SSEM\_tariff Non\_BPT\_Non\_elective\_tariff Non\_BPT\_Reduced\_SSEM\_tariff

/style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###"];

var HRG\_or\_sub\_HRG\_level BPT\_Flag /style(header)=[background=#b3ffff];

var Percentage\_reduction\_SSEM\_tariff /style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:#,##0.0%"];

label

HRG\_Code="HRG code"

HRG\_Name="HRG\_Code name"

BPT\_Non\_elective\_tariff="Best practice tariff:\*Non-elective tariff (£)"

BPT\_Reduced\_SSEM\_tariff="Best practice tariff:\*Reduced short stay \*emergency tariff (£)"

Non\_BPT\_Non\_elective\_tariff="Non-best practice tariff:\*Non-elective tariff (£)"

Non\_BPT\_Reduced\_SSEM\_tariff="Non-best practice tariff:\*Reduced short stay \*emergency tariff (£)"

HRG\_or\_sub\_HRG\_level="HRG or sub-HRG level"

BPT\_Flag="BPT Flag\*(see BPT Flag sheet) "

Percentage\_reduction\_SSEM\_tariff="% applied in calculation of \*reduced short stay emergency tariff(£) "

;

**run**;

/\*Table 5.Early inflammatory arthritis \*/

Title2 j=l "5. Early inflammatory arthritis";

**proc** **print** data = BPTOut.LNK\_5\_EARLYINFLAMMATORYARTHRITIS noobs label split= '\*' ;

var Code HRG\_name

/style(header)=[background=#b3ffff];

var Additional\_payment

/style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###"];

label

Code="Code"

HRG\_Name="Best practice tariff additional payment"

Additional\_payment ="Additional payment(£)";

title5 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "The BPT is a single conditional top-up payment payable for each newly referred patient who receives all six characteristics of best practice (see guidance).";

title6 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "Where satisfied that patients have achieved the best practice criteria, commissioners should make manual adjustments by applying the conditional top-up payment for each newly referred patient who received all six standards of care.";

title7 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "Providers will continue to be paid the national tariff or locally agreed prices for all relevant activity.";

title8 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "SUS+ will not apply the conditional top-up";

**run**;

/\*Table 6.Endoscopy procedures \*/

Title2 j=l "6. Endoscopy procedures";

title5 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "The BPT will apply at the HRG level to all daycase and elective activity.";

title6 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "SUS PbR will automate payment of the best practice tariff to all activity eligible for the BPT. Where the criteria is not met commissioners should reclaim any over payments from providers not achieving level 1 (see guidance).";

Footnote1;

**proc** **print** data = BPTOut.LNK\_6\_ENDOSCOPY\_PROCEDURE noobs label split= '\*' ;

var code HRG\_name /style(header)=[background=#b3ffff];

var Lev\_1\_Best\_practice\_tariff Lev\_2\_Interm\_Bestpractice\_tariff Lev\_3\_Non\_best\_practice\_tariff

/style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###"];

var HRG\_or\_sub\_HRG\_level BPT\_Flag /style(header)=[background=#b3ffff];

label

Code="HRG code"

HRG\_Name="HRG name"

Lev\_1\_Best\_practice\_tariff="Level 1 \*Best practice tariff \*(£)"

Lev\_2\_Interm\_Bestpractice\_tariff="Level 2 \*Intermediate Best practice tariff \*(£)"

Lev\_3\_Non\_best\_practice\_tariff="Level 3 \*Non-best practice tariff \*(£)"

HRG\_or\_sub\_HRG\_level="HRG or sub-HRG level"

BPT\_Flag="BPT Flag (see BPT Flag sheet)"

;

**run**;

/\*Table 7.Fragility hip fracture \*/

Title2 j=l "7. Fragility hip fracture";

title5 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "The BPT is made up of two components: a base tariff and a conditional top-up payment. SUS PbR will automate payment of the base tariff to all activity irrespective of whether the characteristics of best practice are met. The conditional top-up is payable locally if all ";

title6 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "The base tariff and the conditional top-up apply at the Sub-HRG level with the use of 'BPT flag BP01' to capture the relevant activity within the associated HRGs.";

title7 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "For each of the HRGs there will be a conventional tariff applicable to the HRG. The conventional price is to reimburse the costs of the activity unrelated to the BPT within the same HRG. The BPT is mandatory for activity identified by the flag and is therefore not optional. ";

title8 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "Short stay emergency adjustment does not apply. Long stay payments and specialist top-ups do apply where appropriate.";

**proc** **print** data = BPTOut.LNK\_7\_FRAGILITY\_HIP\_FRACTURE noobs label split= '\*' ;

var code HRG\_name /style(header)=[background=#b3ffff];

var Non\_best\_practice\_tariff Best\_practice\_tariff Additional\_payment

/style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###"];

var HRG\_or\_sub\_HRG\_level BPT\_Flag /style(header)=[background=#b3ffff];

label

Code="HRG code"

HRG\_Name="HRG name"

Non\_best\_practice\_tariff="Non-best practice tariff (£)"

Best\_practice\_tariff="Best practice tariff (£)"

Additional\_payment="Additional payment (£)"

HRG\_or\_sub\_HRG\_level="HRG or sub-HRG level"

BPT\_Flag="BPT Flag (see BPT Flag sheet)"

;

**run**;

/\*Table 8.Major trauma \*/

Title2 j=l "8. Major trauma";

title5 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "The BPT is made up of two levels of payment differentiated by the Injury Severity Score (ISS) of the patient, conditional on a set of criteria (see guidance) and paid in addition to the core spell tariff.";

title6 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "Per patient, only one level of payment can be paid to only one major trauma centre. The BPT applies to adults and children.";

title7 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "Long stay payments and specialist top-ups apply, where appropriate, to the core spell tariff.";

title8 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "SUS PbR will not apply the BPT. Organisations will need to use the TARN database to support the payment. ";

**proc** **print** data = BPTOut.LNK\_8\_MAJOR\_TRAUMA noobs label split= '\*' ;

var ID Level /style(header)=[background=#b3ffff];

var Best\_practice\_tariff

/style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###"];

label

Level="Level"

Best\_practice\_tariff="Best practice tariff (£)"

;

**run**;

/\*Table 9.Outpatient procedures \*/

Title2 j=l "9. Outpatient procedures";

title5 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "For the diagnostic procedures, the BPT is made up of a pair of prices for each procedure: one applied to outpatient setting, the other to ordinary and day case elective admissions.";

title6 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "The BPT for all three procedures apply at the HRG level.";

title7 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "SUS PbR will automate payment of the applicable prices.";

**proc** **print** data = BPTOut.LNK\_9\_OUTPATIENTS noobs label split= '\*' ;

var HRG\_code HRG\_name Procedure

/style(header)=[background=#b3ffff];

var Best\_pract\_outpatient\_tariff Nonbest\_pract\_dc\_ord\_el\_tariff

/style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###"];

var HRG\_or\_sub\_HRG\_level BPT\_flag/style(header)=[background=#b3ffff];

label

HRG\_code="HRG code"

HRG\_name="HRG name"

Procedure="Procedure"

Best\_pract\_outpatient\_tariff="Best practice outpatient procedure tariff (£)"

Nonbest\_pract\_dc\_ord\_el\_tariff="Non-best practice combined\* day case / ordinary elective \*spell tariff (£)"

HRG\_or\_sub\_HRG\_level="HRG or sub-HRG level"

BPT\_flag="BPT Flag(see BPT Flag sheet)"

;

**run**;

/\*Table 10.Paediatric diabetes year of care \*/

Title2 j=l "10. Paediatric diabetes year of care";

title5 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "The BPT is an annual payment that covers inpatient and outpatient care compliant with best practice specification (see guidance) from the date of discharge from hospital after the initial diagnosis of diabetes is made, until the young person is transferred to adult services at the age of 19.";

title6 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "SUS PbR will not apply the BPT.";

**proc** **print** data = BPTOut.LNK\_10\_PAEDIATRIC\_DIABETES noobs label split= '\*' ;

var HRG\_code HRG\_name

/style(header)=[background=#b3ffff];

var Best\_practice\_tariff

/style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###"];

label

HRG\_code="Treatment function"

HRG\_name="Treatment function name"

Best\_practice\_tariff="Best practice tariff (£)"

;

**run**;

/\*Table 11.Paediatric epilepsy \*/

Title2 j=l "11. Paediatric epilepsy";

title5 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "The BPT is payable per attendance to follow-up activity captured by the paediatric epilepsy treatment function code (TFC 223), which was introduced to capture activity delivered in line with best practice.";

title6 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "SUS PbR will apply the best practice tariff price. Organisations will need to make adjustments locally for activity which is coded to TFC 223 but does not meet the criteria. Activity not meeting best practice should be paid the outpatient attendance tariff for paediatrics (TFC 420).";

title7 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "The paediatric TFC 420 price (base price) is included below for information.";

**proc** **print** data = BPTOut.LNK\_11\_PAEDIATRIC\_EPILEPSY noobs label split= '\*' ;

var Treatment\_function\_code Treatment\_function\_name

/style(header)=[background=#b3ffff];

var Non\_best\_practice\_tariffWF01A Non\_best\_practice\_tariffWF02A Best\_practice\_tariffWF01A Best\_practice\_tariffWF02A

/style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###"];

label

Treatment\_function\_code="Treatment function"

Treatment\_function\_name="Treatment function name"

Non\_best\_practice\_tariffWF01A="Non-best practice tariff:\* WF01A\*Follow Up Attendance - Single Professional(£)"

Non\_best\_practice\_tariffWF02A="Non-best practice tariff:\* WF02A\*Follow Up Attendance - Multi Professional(£)"

Best\_practice\_tariffWF01A="Best practice tariff:\*WF01A\* Follow Up Attendance - Single Professional (£)"

Best\_practice\_tariffWF02A="Best practice tariff:\*WF02A\* Follow Up Attendance - Multi Professional (£)"

;

**run**;

/\*Table 12.Parkinson's disease \*/

Title2 j=l "12. Parkinson's disease";

title5 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "The BPT is an annual payment for activity delivered in line with the best practice criteria.";

title6 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "The activity associated with Parkinson's disease is currently captured as part of the neurology TFC (TFC 400). Without a discrete TFC it is necessary for eligible activity to be excluded and paid locally (see guidance)."

;

**proc** **print** data =BPTOut.LNK\_12\_PARKINSON noobs label split= '\*' ;

var HRG\_code HRG\_name

/style(header)=[background=#b3ffff];

var Best\_practice\_tariff

/style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###"];

label

HRG\_code="Treatment function"

HRG\_name="Treatment function name"

Best\_practice\_tariff="Best practice tariff (£)"

;

**run**;

/\*Table 13.Pleural effusion \*/

Title2 j=l "13. Pleural effusion";

title5 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "The BPT has been designed to incentivise a shift in activity away from non-elective admissions to pleural effusions being performed as a planned elective basis under ultrasound control (see guidance). ";

title6 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "The BPT applies at HRG level. SUS+ will automate payment. ";

title7 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "The non-elective tariff for DZ16N 'Pleural Effusion with Single Intervention, with CC Score 0-5' has been reduced to incentivise a shift of activity to day case and fund best practice.";

/\*\*/

/\*Title7 j=l "13a. ";\*/

**proc** **print** data = BPTOut.LNK\_13A\_PLEURALEFFUSION noobs label split= '\*' ;

var HRG\_code HRG\_name

/style(header)=[background=#b3ffff];

var Day\_case\_best\_practice\_tariff

/style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###"];

var HRG\_or\_sub\_HRG\_level BPT\_flag

/style(header)=[background=#b3ffff];

label

HRG\_code="HRG code"

HRG\_name="HRG name"

Day\_case\_best\_practice\_tariff="BPT applies to:\*Day case best practice tariff (£)"

HRG\_or\_sub\_HRG\_level="BPT applies to:\*HRG or sub-HRG level"

BPT\_flag="BPT applies to:\*BPT Flag (see BPT Flag sheet)"

;

**run**;

/\*Table 14.Primary total hip and knee replacements \*/

Title2 j=l "14. Primary total hip and knee replacements";

title5 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "The BPT applies at the HRG level to elective admissions. ";

title6 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "SUS PbR will automate payment of the BPT price for all relevant activity. Where the criteria is not met commissioners should reclaim any over payments from providers down to the base tariff (see guidance).";

**proc** **print** data = BPTOut.LNK\_14\_PRIMARY\_HIPKNEE\_REPLAC noobs label split= '\*' ;

var HRG\_code HRG\_name

/style(header)=[background=#b3ffff];

var Non\_best\_practice\_tariff Best\_practice\_tariff

/style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###"];

var HRG\_or\_sub\_HRG\_level BPT\_Flag

/style(header)=[background=#b3ffff];

label

HRG\_code="HRG code"

HRG\_name="HRG name"

Non\_best\_practice\_tariff="BPT applies to:\*Non-best practice tariff (£)"

Best\_practice\_tariff="BPT applies to:\*Best practice tariff (£)"

HRG\_or\_sub\_HRG\_level="BPT applies to:\*HRG or sub-HRG level"

BPT\_Flag="BPT applies to:\*BPT flag (see BPT Flag sheet)"

;

**run**;

/\*Table 15.Same day emergency care \*/

Title2 j=l "15. Same day emergency care";

**proc** **print** data = BPTOut.LNK\_15\_SAME\_DAY\_EM\_CARE noobs label split= '\*' ;

var HRG\_code HRG\_name Clinical\_Scenario

/style(header)=[background=#b3ffff];

var Same\_day\_emergency\_care\_BPT Non\_elective\_non\_BPT\_tariff

/style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###"];

var HRG\_or\_sub\_HRG\_level BPT\_Flag

/style(header)=[background=#b3ffff];

label

HRG\_code="HRG code"

HRG\_name="HRG name"

Clinical\_Scenario="Clinical scenario"

Same\_day\_emergency\_care\_BPT="Same day emergency care BPT (£)\*(LOS = 0 days)"

Non\_elective\_non\_BPT\_tariff="Non-elective non-BPT tariff (£)\*(LOS > 0 days)"

HRG\_or\_sub\_HRG\_level="BPT applies to:\*HRG or sub-HRG level"

BPT\_Flag="BPT applies to:\*BPT flag\*(see BPT Flag sheet)"

;

title4 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "The BPT for each clinical scenario is made up of a pair of prices: one applied to emergency admissions with a zero day length of stay, the other to emergency admissions with a stay of 1 or more days. ";

title5 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "For around half of the scenarios, the BPT will apply at HRG level and for the remaining scenarios, the BPT will apply at the Sub-HRG level with the use of a BPT flag to capture the relevant activity within the associated HRGs. In both cases SUS PbR will generate a BPT flag in order to automate payment. ";

title6 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "Where the BPT applies at the Sub-HRG level, there will be a conventional tariff applicable to the HRG. The conventional tariff is to reimburse the costs of the activity unrelated to the BPT within the same HRG. The BPT is mandatory for activity identified by the flag and is therefore not optional. ";

title7 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "For this BPT a flag is also used to identify the appropriate method of admission for the activity under the scope of the BPT. ";

title8 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "Short stay emergency adjustment does not apply. Long stay payments and specialist top-ups do apply where appropriate. Activity within the scope of the BPT are included in the marginal rate emergency rule. ";

title9 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "SUS PbR will automate payment by: (a) generating the relevant flag, (b) applying relevant price. ";

FootNote1 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "\* for these HRGs the BPT flag is not required to identify specific clinical activity within the scope of a HRG(OPCS and/or ICD ), but is still needed to identify the method of admission. ";

FootNote2 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "\*\* Includes: First seizure and seizure in known epileptic from NHS Institute’s Directory of Ambulatory Emergency Care in Adults. ";

**run**;

/\*Table 16.Transient ischaemic attack \*/

Title2 j=l "16. Transient ischaemic attack";

title3 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "The BPT is made up of two components: a best practice tariff and a conditional top-up payment. Both components are conditional on meeting best practice characteristics though they are payable separately. ";

title4 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "SUS PbR will (a) apply the BPT price to activity coded under the TFC 329 (b) unbundle MRI scans (where coded) and apply the relevant unbundled diagnostic payment (c) prevent generation of an outpatient procedure e.g. where 24 hour ECGs are performed, when recorded against the TIA TFC 329.";

title5 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "The BPT applies to a TFC in the non-admitted setting (outpatients). Activity that does not meet best practice must not be recorded against TFC 329.";

Title7 j=l "16a. ";

FootNote2;

FootNote1;

**proc** **print** data = BPTOut.LNK\_16A\_TIA noobs label split= '\*' ;

var Treatment\_function Treatment\_function\_name

/style(header)=[background=#b3ffff];

var WF01B\_First\_Att\_Single\_Prof WF02B\_First\_Att\_Multi\_Prof

/style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###"];

label

Treatment\_function="Treatment function"

Treatment\_function\_name="Treatment function name"

WF01B\_First\_Att\_Single\_Prof="WF01B\*First Attendance - Single Professional (£)"

WF02B\_First\_Att\_Multi\_Prof="WF02B\*First Attendance - Multi Professional (£)"

;

**run**;

Title2;

Title7 j=l "16b. ";

**proc** **print** data = BPTOut.LNK\_16B\_TIA noobs label split= '\*' ;

var ID Best\_practice\_tariff\_adjustments

/style(header)=[background=#b3ffff];

var Tariff\_adjustment

/style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###"];

label

Best\_practice\_tariff\_adjustments="Best practice tariff adjustments"

Tariff\_adjustment="Tariff adjustment (£)"

;

**run**;

/\*Table 17.Heart Failure \*/

Title2 j=l "17. Heart Failure";

title3 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "The BPT applies at the HRG level to relevant non-elective admissions. ";

title4 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "The BPT is made up of two components: a base tariff and a conditional top-up payment. SUS PbR will automate payment of the base tariff to all activity irrespective of whether the characteristics of best practice are met. The conditional top-up is payable locally if all defined characteristics are achieved (see guidance).";

title5 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "The short stay emergency adjustment continues to apply (see guidance).";

**proc** **print** data = BPTOut.LNK\_17\_HEARTFAILURE noobs label split= '\*' ;

var HRG\_code HRG\_name

/style(header)=[background=#b3ffff];

var Non\_BPT\_Non\_elective\_tariff Non\_BPT\_Reduced\_SSEM\_tariff BPT\_Non\_elective\_tariff BPT\_Reduced\_SSEM\_tariff

/style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###"];

var HRG\_or\_sub\_HRG\_level BPT\_Flag

/style(header)=[background=#b3ffff];

var Percentage\_reduction\_SSEM\_tariff /style(header)=[background=#b3ffff] style(Column)=[tagattr="format:#,##0.0%"];

label

HRG\_code="HRG code"

HRG\_name="HRG name"

Non\_BPT\_Non\_elective\_tariff="Non-best practice tariff:\*Non-elective tariff (£)"

Non\_BPT\_Reduced\_SSEM\_tariff="Non-best practice tariff:\*Reduced short stay emergency tariff (£)"

BPT\_Non\_elective\_tariff="Best practice tariff:\*Non-elective tariff (£)"

BPT\_Reduced\_SSEM\_tariff="Best practice tariff:\*Reduced short stay emergency tariff (£)"

HRG\_or\_sub\_HRG\_level="HRG or sub-HRG level"

BPT\_Flag="BPT Flag(see BPT Flag sheet)"

Percentage\_reduction\_SSEM\_tariff="% applied in calculation of reduced short stay emergency tariff "

;

**run**;

/\*Table 18.COPD Exacerbation \*/

Title2 j=l "18. COPD Exacerbation";

title3 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "The BPT applies at the HRG level to all non-elective admissions.";

title4 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "The BPT is made up of two components: a base tariff and a conditional top-up payment. SUS PbR will automate payment of the base tariff to all activity irrespective of whether the characteristics of best practice are met. The conditional top-up is payable locally if all defined characteristics are achieved (see guidance). ";

title5 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "The short stay emergency adjustment continues to apply (see guidance). ";

**proc** **print** data = BPTOut.LNK\_18\_COPD\_EXACERBATION noobs label split= '\*' ;

var HRG\_code HRG\_name

/style(header)=[background=#b3ffff];

var Non\_BPT\_Non\_elective\_tariff Non\_BPT\_Reduced\_SSEM\_tariff BPT\_Non\_elective\_tariff BPT\_Reduced\_SSEM\_tariff

/style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###"];

var HRG\_or\_sub\_HRG\_level BPT\_Flag/style(header)=[background=#b3ffff];

var Percentage\_reduction\_SSEM\_tariff/style(header)=[background=#b3ffff] style(Column)=[tagattr="format:#,##0.0%"];

label

HRG\_code="HRG code"

HRG\_name="HRG name"

Non\_BPT\_Non\_elective\_tariff="Non-best practice tariff:\*Non-elective tariff (£)"

Non\_BPT\_Reduced\_SSEM\_tariff="Non-best practice tariff:\*Reduced short stay emergency tariff (£)"

BPT\_Non\_elective\_tariff="Best practice tariff:\*Non-elective tariff (£)"

BPT\_Reduced\_SSEM\_tariff="Best practice tariff:\*Reduced short stay emergency tariff (£)"

HRG\_or\_sub\_HRG\_level="HRG or sub-HRG level"

BPT\_Flag="BPT Flag (see BPT Flag sheet)"

Percentage\_reduction\_SSEM\_tariff="% applied in calculation of reduced short stay emergency tariff "

;

**run**;

/\*Table 19.NSTEMI: Timely access to coronary angiography \*/

Title2 j=l "19. NSTEMI:Timely access to coronary angiography";

title3 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "The BPT applies at sub-HRG level to all non-elective admissions. SUS PbR will automate payment of the base tariff. Where satisfied that providers have achieved the best practice criteria, commissioners should make payment of the conditional top-up to the best practice tariff.";

title4 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "As the BPT applies at the Sub-HRG level, with the use of a BPT flag, there will be a conventional tariff applicable to the HRG. The conventional tariff is to reimburse the costs of the activity unrelated to the BPT within the same HRG. ";

title5 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "The BPT is mandatory for activity identified by the flag and is therefore not optional.";

**proc** **print** data = BPTOut.LNK\_19\_NSTEMI noobs label split= '\*' ;

var HRG\_code HRG\_name

/style(header)=[background=#b3ffff];

var Non\_BPT\_Non\_elective\_tariff BPT\_Non\_elective\_tariff

/style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###"];

var HRG\_or\_sub\_HRG\_level BPT\_Flag /style(header)=[background=#b3ffff];

label

HRG\_code="HRG code"

HRG\_name="HRG name"

Non\_BPT\_Non\_elective\_tariff="Non-best practice tariff (£)"

BPT\_Non\_elective\_tariff="Best practice tariff (£)"

HRG\_or\_sub\_HRG\_level="HRG Level"

BPT\_Flag="BPT Flag(see BPT Flag sheet)"

;

**run**;

/\*Table 20.Emergency Laparotomy \*/

Title2 j=l "20. Emergency Laparotomy";

title3 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "The BPT applies at the HRG level to all non-elective admissions.";

title4 JUSTIFY=left FONT=Arial HEIGHT=**12**pt BCOLOR="#ffff80" "The BPT is made up of two components: a base tariff and a conditional top-up payment. SUS PbR will automate payment of the base tariff to all activity irrespective of whether the characteristics of best practice are met. The conditional top-up is payable locally if all defined characteristics are achieved (see guidance). ";

**proc** **print** data = BPTOut.LNK\_20\_EM\_LAPAROTOMY noobs label split= '\*' ;

var HRG\_code HRG\_name

/style(header)=[background=#b3ffff];

var Non\_BPT\_Non\_elective\_tariff

/\*Non\_BPT\_Reduced\_SSEM\_tariff \*/

BPT\_Non\_elective\_tariff

/\*BPT\_Reduced\_SSEM\_tariff \*/

/style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###"];

var HRG\_or\_sub\_HRG\_level BPT\_Flag/style(header)=[background=#b3ffff];

label

HRG\_code="HRG code"

HRG\_name="HRG name"

Non\_BPT\_Non\_elective\_tariff="Non-best practice tariff:\*Non-elective tariff (£)"

BPT\_Non\_elective\_tariff="Best practice tariff:\*Non-elective tariff (£)"

HRG\_or\_sub\_HRG\_level="HRG or sub-HRG level"

BPT\_Flag="BPT Flag (see BPT Flag sheet)"

;

**run**;

Title2 ;

ods tagsets.ExcelXP close;

ods listing;

Options missing = **0**;

/\*APC\_OPROC ModelOut table \*/

**data** APC\_OPROC\_MODEL\_LINKEDSHEET;

set Modelout.APC\_OPROC\_MODEL\_LINKEDSHEET;

**run**;

/\*BPT ModelOut table \*/

**data** BPT\_ANNEX\_A\_APC\_OPROC;

set BPTout.ANNEX\_A\_APC\_OPROC;

**run**;

/\*Create Link sheet for Annex\_A \*/

**Proc** **sql**;

create table APC\_OPROC\_Annext\_A\_1

as

select

a.\*,

b.\*

from APC\_OPROC\_MODEL\_LINKEDSHEET as a

left join BPT\_ANNEX\_A\_APC\_OPROC as b

on a.HRG\_Code=b.HRG\_Code

order by a.HRG\_Code

;

**quit**;

/\*1.for Outpatient codes (LB14Z, LB72A,MA10Z, MA31Z, MA32Z)I have to replace column D (OPROC\_Tariff)with BPT prices and

column E(Combined\_DC\_EL\_Prices)

with nonBPT prices.

2.for Pleural Effusion codes (YD04Z, YD05Z) I have to modify column E combined day case elective using BPT prices.

3.where the BPT areas is DayCase we need to remove the values in the combined daycase el. columnand

the split values for DayCase column F (BPt prices) and elective column G (non bpt prices) separated.

\*/

**data** APC\_OPROC\_Annext\_A\_2;

set APC\_OPROC\_Annext\_A\_1;

if HRG\_Code in ("LB14Z", "LB72A","MA10Z", "MA31Z", "MA32Z")

then do;

OPROC\_Tariff=BPT;

Combined\_DC\_EL\_Prices=NonBPT;

End;

if HRG\_Code in ("YD04Z", "YD05Z")

then do;

Combined\_DC\_EL\_Prices=BPT;

End;

if BPT\_area="Daycase"

then do;

Combined\_DC\_EL\_Prices=**.**;

DC\_Prices=BPT;

EL\_Prices=NonBPT;

End;

**run**;

/\*No Duplcated HRG \*/

**proc** **sql**;

create table APC\_OPROC\_Annext\_A\_3\_1

as

select a.\*,count(HRG\_Code) as Count

from APC\_OPROC\_Annext\_A\_2 as a

group by HRG\_Code

having count(HRG\_Code)=**1**

;

**quit**;

/\*Duplcated HRG \*/

**proc** **sql**;

create table APC\_OPROC\_Annext\_A\_3\_2\_1

as

select a.\*,count(HRG\_Code) as Count

from APC\_OPROC\_Annext\_A\_2 as a

group by HRG\_Code

having count(HRG\_Code)>**1** and type="table\_3"

;

**quit**;

/\*special treatment for duplicated HRGS --------------------------------------- Start from here\*/

/\*3 HRGs: FF51E, JA20F and LB15E \*/

/\*Case 1 - FF51E \*/

**proc** **sql**;

create table APC\_OPROC\_Annext\_A\_3\_2\_1\_FF51E\_1

as

select

Year,

HRG\_Code,

HRG\_Name,

OPROC\_Tariff,

Combined\_DC\_EL\_Prices,

DC\_Prices,

EL\_Prices,

EL\_TRIM\_POINTS,

NE\_Prices,

NE\_TRIM\_POINTS,

EBD\_Prices,

SSEM\_applicable,

SSEM\_Pct,

SSEM\_Price,

BPT\_HRGsubHRG,

BPT\_area,

POD\_BPT,

SUS\_BPT\_nonBPT,

Price\_information,

BPT\_flag,

nonBPT,

bpt,

type,

procedure,

Condition\_index,

Condition\_Information,

Count

from

APC\_OPROC\_Annext\_A\_3\_2\_1

where HRG\_Code="FF51E"

;

**quit**;

**proc** **sql**;

select BPT\_area,

BPT\_flag,

BPT\_HRGsubHRG

into :BPT\_Area\_1,

:BPT\_flag\_1,

:BPT\_HRGsubHRG\_1

from APC\_OPROC\_Annext\_A\_3\_2\_1\_FF51E\_1

where BPT\_area="Emergency Laparotomy"

;

**quit**;

**proc** **sql**;

create table APC\_OPROC\_Annext\_A\_3\_2\_1\_FF51E\_2

as

select

Year,

HRG\_Code,

HRG\_Name,

OPROC\_Tariff,

Combined\_DC\_EL\_Prices,

DC\_Prices,

EL\_Prices,

EL\_TRIM\_POINTS,

NE\_Prices,

NE\_TRIM\_POINTS,

EBD\_Prices,

SSEM\_applicable,

SSEM\_Pct,

SSEM\_Price,

strip("&BPT\_HRGsubHRG\_1.")||"/"||strip(BPT\_HRGsubHRG) as BPT\_HRGsubHRG2,

strip(BPT\_area)||"/"||trim("&BPT\_Area\_1.") as BPT\_area2,

POD\_BPT,

SUS\_BPT\_nonBPT,

Price\_information,

BPT\_flag||"/"||trim("&BPT\_flag\_1.") as BPT\_flag2,

nonBPT,

bpt,

type,

procedure,

Condition\_index,

Condition\_Information,

Count

from

APC\_OPROC\_Annext\_A\_3\_2\_1\_FF51E\_1

where BPT\_area="Daycase"

;

**quit**;

**proc** **sql**;

create table APC\_OPROC\_Annext\_A\_3\_2\_1\_FF51E\_F

as

select

Year,

HRG\_Code,

HRG\_Name,

OPROC\_Tariff,

Combined\_DC\_EL\_Prices,

DC\_Prices,

EL\_Prices,

EL\_TRIM\_POINTS,

NE\_Prices,

NE\_TRIM\_POINTS,

EBD\_Prices,

SSEM\_applicable,

SSEM\_Pct,

SSEM\_Price,

BPT\_HRGsubHRG2 as BPT\_HRGsubHRG,

BPT\_area2 as BPT\_area,

POD\_BPT,

SUS\_BPT\_nonBPT,

Price\_information,

BPT\_flag2 as BPT\_flag,

nonBPT,

bpt,

type,

procedure,

Condition\_index,

Condition\_Information,

Count

from

APC\_OPROC\_Annext\_A\_3\_2\_1\_FF51E\_2

;

**quit**;

/\*Case 2 - JA20F \*/

**proc** **sql**;

create table APC\_OPROC\_Annext\_A\_3\_2\_1\_JA20F\_1

as

select

Year,

HRG\_Code,

HRG\_Name,

OPROC\_Tariff,

Combined\_DC\_EL\_Prices,

DC\_Prices,

EL\_Prices,

EL\_TRIM\_POINTS,

NE\_Prices,

NE\_TRIM\_POINTS,

EBD\_Prices,

SSEM\_applicable,

SSEM\_Pct,

SSEM\_Price,

BPT\_HRGsubHRG,

BPT\_area,

POD\_BPT,

SUS\_BPT\_nonBPT,

Price\_information,

BPT\_flag,

nonBPT,

bpt,

type,

procedure,

Condition\_index,

Condition\_Information,

Count

from

APC\_OPROC\_Annext\_A\_3\_2\_1

where HRG\_Code="JA20F" and procedure="Axillary clearance"

;

**quit**;

**proc** **sql**;

select BPT\_area,

BPT\_flag

into :BPT\_Area\_1,

:BPT\_flag\_1

from APC\_OPROC\_Annext\_A\_3\_2\_1\_JA20F\_1

where BPT\_flag="BP32"

;

**quit**;

**proc** **sql**;

create table APC\_OPROC\_Annext\_A\_3\_2\_1\_JA20F\_2

as

select

Year,

HRG\_Code,

HRG\_Name,

OPROC\_Tariff,

Combined\_DC\_EL\_Prices,

DC\_Prices,

EL\_Prices,

EL\_TRIM\_POINTS,

NE\_Prices,

NE\_TRIM\_POINTS,

EBD\_Prices,

SSEM\_applicable,

SSEM\_Pct,

SSEM\_Price,

BPT\_HRGsubHRG,

BPT\_area,

POD\_BPT,

SUS\_BPT\_nonBPT,

Price\_information,

BPT\_flag||"/"||trim("&BPT\_flag\_1.") as BPT\_flag2,

nonBPT,

bpt,

type,

procedure,

Condition\_index,

Condition\_Information,

Count

from

APC\_OPROC\_Annext\_A\_3\_2\_1\_JA20F\_1

where BPT\_flag="BP28"

;

**quit**;

**proc** **sql**;

create table APC\_OPROC\_Annext\_A\_3\_2\_1\_JA20F\_F

as

select

Year,

HRG\_Code,

HRG\_Name,

OPROC\_Tariff,

Combined\_DC\_EL\_Prices,

DC\_Prices,

EL\_Prices,

EL\_TRIM\_POINTS,

NE\_Prices,

NE\_TRIM\_POINTS,

EBD\_Prices,

SSEM\_applicable,

SSEM\_Pct,

SSEM\_Price,

BPT\_HRGsubHRG,

BPT\_area,

POD\_BPT,

SUS\_BPT\_nonBPT,

Price\_information,

BPT\_flag2 as BPT\_flag,

nonBPT,

bpt,

type,

procedure,

Condition\_index,

Condition\_Information,

Count

from

APC\_OPROC\_Annext\_A\_3\_2\_1\_JA20F\_2

;

**quit**;

/\*Case 3 - LB15E\*/

**proc** **sql**;

create table APC\_OPROC\_Annext\_A\_3\_2\_1\_LB15E\_1

as

select

Year,

HRG\_Code,

HRG\_Name,

OPROC\_Tariff,

Combined\_DC\_EL\_Prices,

DC\_Prices,

EL\_Prices,

EL\_TRIM\_POINTS,

NE\_Prices,

NE\_TRIM\_POINTS,

EBD\_Prices,

SSEM\_applicable,

SSEM\_Pct,

SSEM\_Price,

BPT\_HRGsubHRG,

BPT\_area,

POD\_BPT,

SUS\_BPT\_nonBPT,

Price\_information,

BPT\_flag,

nonBPT,

bpt,

type,

procedure,

Condition\_index,

Condition\_Information,

Count

from

APC\_OPROC\_Annext\_A\_3\_2\_1

where HRG\_Code="LB15E"

;

**quit**;

**proc** **sql** noprint;

select BPT\_area,

BPT\_flag

into :BPT\_Area\_1,

:BPT\_flag\_1

from APC\_OPROC\_Annext\_A\_3\_2\_1\_LB15E\_1

where BPT\_flag="BP82"

;

**quit**;

**proc** **sql** noprint;

create table APC\_OPROC\_Annext\_A\_3\_2\_1\_LB15E\_2

as

select

Year,

HRG\_Code,

HRG\_Name,

OPROC\_Tariff,

Combined\_DC\_EL\_Prices,

DC\_Prices,

EL\_Prices,

EL\_TRIM\_POINTS,

NE\_Prices,

NE\_TRIM\_POINTS,

EBD\_Prices,

SSEM\_applicable,

SSEM\_Pct,

SSEM\_Price,

BPT\_HRGsubHRG,

strip(BPT\_area)||"/"||trim("&BPT\_Area\_1.") as BPT\_area2,

POD\_BPT,

SUS\_BPT\_nonBPT,

Price\_information,

BPT\_flag||"/"||trim("&BPT\_flag\_1.") as BPT\_flag2,

nonBPT,

bpt,

type,

procedure,

Condition\_index,

Condition\_Information,

Count

from

APC\_OPROC\_Annext\_A\_3\_2\_1\_LB15E\_1

where BPT\_flag="BP92"

;

**quit**;

**proc** **sql** noprint;

create table APC\_OPROC\_Annext\_A\_3\_2\_1\_LB15E\_F

as

select

Year,

HRG\_Code,

HRG\_Name,

OPROC\_Tariff,

Combined\_DC\_EL\_Prices,

DC\_Prices,

EL\_Prices,

EL\_TRIM\_POINTS,

NE\_Prices,

NE\_TRIM\_POINTS,

EBD\_Prices,

SSEM\_applicable,

SSEM\_Pct,

SSEM\_Price,

BPT\_HRGsubHRG,

BPT\_area2 as BPT\_area ,

POD\_BPT,

SUS\_BPT\_nonBPT,

Price\_information,

BPT\_flag2 as BPT\_flag,

nonBPT,

bpt,

type,

procedure,

Condition\_index,

Condition\_Information,

Count

from

APC\_OPROC\_Annext\_A\_3\_2\_1\_LB15E\_2

;

**quit**;

/\*Combine all duplicated HRGs \*/

**data** APC\_OPROC\_Annext\_A\_3\_2;

Length BPT\_HRGsubHRG $20. BPT\_flag $40.;

set

APC\_OPROC\_Annext\_A\_3\_2\_1\_FF51E\_F

APC\_OPROC\_Annext\_A\_3\_2\_1\_JA20F\_F

APC\_OPROC\_Annext\_A\_3\_2\_1\_LB15E\_F

;

**run**;

/\*special treatment for duplicated HRGS --------------------------------------- End here\*/

/\*combine Non\_duplicated and Duplicated HRGs \*/

**data** APC\_OPROC\_Annext\_A\_3;

Length BPT\_HRGsubHRG $20. BPT\_flag $40.;

set

APC\_OPROC\_Annext\_A\_3\_1

APC\_OPROC\_Annext\_A\_3\_2

;

**run**;

/\*Zero prce HRGS\*/

**data** ZeroPriceHRG;

set tariff\_R.ZeroPriceHRG;

where year="1920"

;

**run**;

**data** HRG\_Elighbility;

set tariff\_r.hrg\_eligibility;

where year="19/20";

**run**;

**proc** **sql** noprint;

create table ZeroPriceHRG\_Name\_TP

as

select

a.Zero\_Price\_HRGs as HRG\_Code,

b.Description as HRG\_Name,

case when c.DC\_EL<**5** then **5**

else c.DC\_EL

End as EL\_TRIM\_POINTS,

case when c.NE<**5** then **5**

Else c.NE end as NE\_TRIM\_POINTS

from ZeroPriceHRG as a

left join HRG\_Elighbility as b

on a.Zero\_Price\_HRGs=b.HRG

left join BPrprep.HES\_Trimpoints(where=(Run\_ID=&Run\_ID)) as c

on a.Zero\_Price\_HRGs=c.HRG

;

**quit**;

**proc** **sql** noprint;

create table ZeroPriceHRG\_APC\_OPROC\_BPT

as

select

"19/20" as Year,

HRG\_Code,

HRG\_Name,

**0** as OPROC\_Tariff,

**0** as Combined\_DC\_EL\_Prices,

**.** as DC\_Prices,

**.** as EL\_Prices,

EL\_TRIM\_POINTS,

**0** as NE\_Prices,

NE\_TRIM\_POINTS,

**0** as EBD\_Prices,

"NO" as SSEM\_applicable,

**.** as SSEM\_Pct,

**.** as SSEM\_Price,

" " as BPT\_HRGsubHRG,

" " as BPT\_area,

" " as POD\_BPT,

" " as SUS\_BPT\_nonBPT,

" " as Price\_information,

" " as BPT\_flag

from

ZeroPriceHRG\_Name\_TP

;

**quit**;

/\*combine all APC with zero price \*/

**data** APC\_OPROC\_Annext\_A\_4;

Length BPT\_flag $40.;

set

APC\_OPROC\_Annext\_A\_3

ZeroPriceHRG\_APC\_OPROC\_BPT

;

**run**;

/\*Final report \*/

**proc** **sql**;

create table ModelOut.APC\_OPROC\_BPT\_Annext\_A

as

select

Year,

HRG\_Code,

HRG\_Name,

OPROC\_Tariff,

Combined\_DC\_EL\_Prices,

DC\_Prices,

EL\_Prices,

EL\_TRIM\_POINTS,

NE\_Prices,

NE\_TRIM\_POINTS,

EBD\_Prices,

SSEM\_applicable,

SSEM\_Pct,

SSEM\_Price,

BPT\_HRGsubHRG,

BPT\_area,

POD\_BPT,

SUS\_BPT\_nonBPT,

Price\_information,

BPT\_flag

from APC\_OPROC\_Annext\_A\_4

order by HRG\_Code

;

**quit**;

/\*--------------------------------------------------------------\*/

\* Excel\_Step3\_4 QUANTUM RECONCILIATION BY Sub Chapter \*/

/\* Version: 1.03 \*/

/\* Author: PID \*/

/\* Date: September 2017 \*/

/\*--------------------------------------------------------------\*/

/\* Input: \*/

/\* \*/

/\* (1)[input].HRGsEligibility \*/

/\* (2)APCSTAGE.APC\_Tariff \*/

/\* (3)[Stage].APC02\_05C \*/

/\* (4)[SQLView].V\_SSEM\_Count \*/

/\* \*/

/\* Output: \*/

/\* (1)Modelout.QUANTUM\_RECONCILIATION\_BySubChap \*/

/\* \*/

/\* Description: \*/

/\* Create required data for Step3: Quantum Reconciliation \*/

/\* by Sub Chapter \*/

/\* \*/

/\* \*/

/\*--------------------------------------------------------------\*/

/\*get OPROC Tariff \*/

**proc** **sql** noprint;

create table OPROC\_Eligible\_Tariff

as select

a.HRG,

case

when b.OPROC\_Tariff=**.** then **0**

else b.OPROC\_Tariff

End as Tariff

from NHSE\_Currency\_specification\_SAS as a

left join

ModelOut.OPROC\_PRICES\_4\_IA\_MODEL as b

on a.HRG =b.HRG

where a.Outpatient\_procedure\_tariff=**1**

order by a.HRG

;

**Quit**;

/\*Part 0 -- HRG Flags ------------ Start from here \*/

**proc** **sql** noprint;

create table QRbySC\_Part0

as

select

a.HRG ,

a.Description ,

case

when a.Outpatient\_procedure\_tariff=**1** then a.HRG

else " "

end as OPROC\_currency\_applicable,

case

when (calculated OPROC\_currency\_applicable is not null) and (c.OPROC\_Tariff is null) then **0**

when (calculated OPROC\_currency\_applicable is not null) and (c.OPROC\_Tariff is not null ) then c.OPROC\_Tariff

else **.**

End as P2\_OPROC label="P2\_OPROC",

b.EL\_Tariff as P2\_DC label="P2\_DC" ,

b.EL\_Tariff as P2\_EL label="P2\_EL" ,

b.NE\_Tariff as P2\_NE label="P2\_NE",

case

when Calculated P2\_OPROC is not null then **1**

else **.**

End as P0\_OPROC,

case

when P2\_DC is not null then **1**

else **.**

End as P0\_DC,

case

when P2\_EL is not null then **1**

else **.**

End as P0\_EL,

case

when P2\_NE is not null then **1**

else **.**

End as P0\_NE

from

NHSE\_Currency\_specification\_SAS as a

left join

&StageLib.**.A**PC\_Tariff as b

on a.HRG=b.SPELL\_HRG

left join ModelOut.OPROC\_PRICES\_4\_IA\_MODEL as c /\* OPROC Tariff \*/

on a.HRG=c.HRG

order by a.HRG

;

**quit**;

/\*Part 0 -- HRG Flags ------------ End here \*/

/\*Part 1 -- Total activity spells ------------ Start from here \*/

**proc** **sql** noprint;

create table HES\_Spell\_Activitie\_DC

as select \*

from bprPrep.HES\_Spell\_Counts\_Adm

where Run\_ID=&Run\_ID. and ADM="DC"

order by SPELL\_HRG

;

**quit**;

**proc** **sql** noprint;

create table HES\_Spell\_Activitie\_EL

as select \*

from bprPrep.HES\_Spell\_Counts\_Adm

where Run\_ID=&Run\_ID. and ADM="EL"

order by SPELL\_HRG

;

**quit**;

**proc** **sql** noprint;

create table HES\_Spell\_Activitie\_NE

as select \*

from bprPrep.HES\_Spell\_Counts\_Adm

where Run\_ID=&Run\_ID. and ADM="NE"

order by SPELL\_HRG

;

**quit**;

**proc** **sql** noprint;

create table QRbySC\_Part1\_1

as

select

a.HRG,

a.Description,

a.OPROC\_currency\_applicable,

a.P0\_OPROC,

a.P0\_DC,

a.P0\_EL,

a.P0\_NE,

case

when (a.OPROC\_currency\_applicable is not null) and (b.OPROC\_Tariff is null) then **0**

when (a.OPROC\_currency\_applicable is not null) and (b.OPROC\_Tariff is not null ) then b.TOTAL\_ACT

else **.**

End as P1\_OPROC\_TA ,

b.TOTAL\_ACT as P1\_OPROC\_TA,

c.SPELLFLAG as P1\_DC\_TA,

d.SPELLFLAG as P1\_EL\_TA,

e.SPELLFLAG as P1\_NE\_TA

from

QRbySC\_Part0 as a

left join

ModelOut.OPROC\_PRICES\_4\_IA\_MODEL as b

on a.HRG=b.HRG

left join HES\_Spell\_Activitie\_DC as c

on a.HRG=c.SPELL\_HRG

left join HES\_Spell\_Activitie\_EL as d

on a.HRG=d.SPELL\_HRG

left join HES\_Spell\_Activitie\_NE as e

on a.HRG=e.SPELL\_HRG

order by a.HRG

;

**quit**;

**proc** **sql** noprint;

create table APCHRGsEligibility

as

select

HRG\_Code,

Eligible\_for\_SSEM,

SSEM\_Banding

from tariff\_r.HRGSELIGIBILITY\_TEMP

where Year="&Year."

order by HRG\_Code

;

**quit**;

**proc** **sql** noprint;

create table APC\_SumEBDS

as

select

SPELL\_HRG,

sum(EBDS) as EBDS

from &Stagelib.**.A**PC02\_05C

group by SPELL\_HRG

order by SPELL\_HRG

;

**quit**;

**proc** **sql** noprint;

create table APC\_SSEM

as

select \*

from &SQLViewLib.**.S**SEM\_Count

where Run\_ID=&Run\_ID.

order by Spell\_HRG

;

**quit**;

**proc** **sql** noprint;

Create table QRbySC\_Part1

as

select

a.HRG,

a.Description,

a.OPROC\_currency\_applicable,

a.P0\_OPROC,

a.P0\_DC,

a.P0\_EL,

a.P0\_NE,

a.P1\_OPROC\_TA,

a.P1\_DC\_TA,

a.P1\_EL\_TA,

a.P1\_NE\_TA,

case

when b.Eligible\_for\_SSEM=**1** then sum(a.P1\_NE\_TA,-d.SSEM)

else a.P1\_NE\_TA

end as P1\_NE\_AT\_less\_SSEM,

c.EBDS as P1\_LSP\_days,

d.SSEM as P1\_SSEM

from QRbySC\_Part1\_1 as a

left join

APCHRGsEligibility as b

on a.HRG=b.HRG\_Code

left join APC\_SumEBDS as c

on a.HRG=c.SPELL\_HRG

left join APC\_SSEM as d

on a.HRG=d.Spell\_HRG

order by a.HRG

;

**quit**;

/\*Part 1 -- Total activity spells ------------ End here \*/

/\*Part2 -- Modelled prices pre quantum reconciliation (QR1) -- Start from here \*/

**proc** **sql** noprint;

create table QRbySC\_Part2\_1

as

select

a.HRG,

a.Description,

a.OPROC\_currency\_applicable,

a.P0\_OPROC,

a.P0\_DC,

a.P0\_EL,

a.P0\_NE,

a.P1\_OPROC\_TA,

a.P1\_DC\_TA,

a.P1\_EL\_TA,

a.P1\_NE\_TA,

a.P1\_NE\_AT\_less\_SSEM,

a.P1\_LSP\_days,

a.P1\_SSEM,

b.P2\_OPROC,

b.P2\_DC,

b.P2\_EL,

b.P2\_NE,

c.Initial\_longStayPayment as P2\_LSP,

case

when d.Eligible\_for\_SSEM=**1** then d.SSEM\_Banding

else **0**

End as P2\_SSEM\_Banding,

case

when Calculated P2\_SSEM\_Banding =**0** then **0**

when Calculated P2\_SSEM\_Banding=**.** then **.**

else e.Tariff\_Pct

End as P2\_SSEM\_Pct,

case

when Calculated P2\_SSEM\_Banding =**0** then **.**

when Calculated P2\_SSEM\_Banding=**.** then **.**

else (calculated P2\_SSEM\_Pct\*P2\_NE)

End as P2\_SSEM\_Tariff

from QRbySC\_Part1 as a

left join QRBYSC\_PART0 as b

on a.HRG=b.HRG

left join ModelOut.STEP\_BY\_STEP\_TRIMPOINT\_AND\_LSP as c

on a.HRG=c.SPELL\_HRG

left join APCHRGsEligibility as d

on a.HRG=d.HRG\_Code

left join Tariff\_R.SSEM\_BANDING(where=(Year="&Year.")) as e

on d.SSEM\_Banding=e.Band

order by a.HRG

;

**quit**;

/\*Add row number \*/

**proc** **sql** noprint;

create table QRbySC\_Part2

as

select

monotonic() as row\_no,

a.\*

from QRbySC\_Part2\_1 as a

order by row\_no

;

**quit**;

/\*Part2 -- Modelled prices pre quantum reconciliation (QR1) -- End here \*/

/\*Part3 -- Total quantum by subchapter pre quantum reconciliation (QR1) Start from here \*/

**proc** **sql** noprint;

create table QRbySC\_Part3\_1

as

select

a.row\_no,

a.HRG,

a.Description,

a.OPROC\_currency\_applicable,

a.P0\_OPROC,

a.P0\_DC,

a.P0\_EL,

a.P0\_NE,

a.P1\_OPROC\_TA,

a.P1\_DC\_TA,

a.P1\_EL\_TA,

a.P1\_NE\_TA,

a.P1\_NE\_AT\_less\_SSEM,

a.P1\_LSP\_days,

a.P1\_SSEM,

a.P2\_OPROC,

a.P2\_DC,

a.P2\_EL,

a.P2\_NE,

a.P2\_LSP,

a.P2\_SSEM\_Banding,

a.P2\_SSEM\_Pct,

a.P2\_SSEM\_Tariff,

substr(a.HRG,**1**,**2**) as P3\_Sub\_Chapter,

a.P1\_OPROC\_TA\*a.P2\_OPROC as P3\_OPROC\_quantum\_by\_HRG ,

case

when a.P2\_SSEM\_Tariff=**0** then sum(a.P2\_DC\*a.P1\_DC\_TA, a.P2\_EL\*a.P1\_EL\_TA,a.P2\_NE\*a.P1\_NE\_AT\_less\_SSEM,a.P1\_LSP\_days\*a.P2\_LSP)

else sum(a.P2\_DC\*a.P1\_DC\_TA, a.P2\_EL\*a.P1\_EL\_TA, a.P2\_NE\*a.P1\_NE\_AT\_less\_SSEM,a.P2\_LSP\*a.P1\_LSP\_days, a.P2\_SSEM\_Tariff\*a.P1\_SSEM)

end as P3\_APC\_Quantum\_by\_HRG

from QRbySC\_Part2 as a

order by a.row\_no

;

**quit**;

/\*sub-chapter table \*/

**proc** **sql** noprint;

create table APC\_Sub\_Chapter

as

select HRGSubchapter

from Tariff\_R.CNST

where year="&FYear." and HRGSubchapter^="Maternity\*"

order by HRGSubchapter

;

**quit**;

/\*OPROC quantum by subchapter\*/

**proc** **sql** noprint;

create table OPROC\_quantum\_by\_subchapter

as

select

P3\_Sub\_Chapter,

sum(P3\_OPROC\_quantum\_by\_HRG) as OPROC\_quantum\_by\_SubChapter

from QRbySC\_Part3\_1

group by P3\_Sub\_Chapter

order by P3\_Sub\_Chapter

;

**quit**;

/\*APC quantum by subchapter\*/

**proc** **sql** noprint;

create table APC\_quantum\_by\_subchapter

as

select

P3\_Sub\_Chapter,

sum(P3\_APC\_Quantum\_by\_HRG) as APC\_quantum\_by\_SubChapter

from QRbySC\_Part3\_1

group by P3\_Sub\_Chapter

order by P3\_Sub\_Chapter

;

**quit**;

**proc** **sql** noprint;

create table QRbySC\_Part3\_2

as

select

a.HRGSubchapter as P3\_Sub\_\_Chapter,

b.OPROC\_quantum\_by\_SubChapter as P3\_OPROC\_quantum\_by\_SubChapter,

c.APC\_quantum\_by\_SubChapter as P3\_APC\_quantum\_by\_SubChapter,

sum(b.OPROC\_quantum\_by\_SubChapter,c.APC\_quantum\_by\_SubChapter) as P3\_APC\_OPROC\_quantum\_by\_chapter

from APC\_Sub\_Chapter as a

left join

OPROC\_quantum\_by\_subchapter as b

on a.HRGSubchapter=b.P3\_Sub\_Chapter

left join

APC\_quantum\_by\_subchapter as c

on a.HRGSubchapter=c.P3\_Sub\_Chapter

order by a.HRGSubchapter

;

**quit**;

/\* Total Summary \*/

**proc** **sql** noprint;

create table QRbySC\_Part3\_2\_2

as

select

"Total" as P3\_Sub\_\_Chapter length=**10**,

sum(P3\_OPROC\_quantum\_by\_SubChapter) as P3\_OPROC\_quantum\_by\_SubChapter,

sum(P3\_APC\_quantum\_by\_SubChapter) as P3\_APC\_quantum\_by\_SubChapter,

sum(P3\_APC\_OPROC\_quantum\_by\_chapter) as P3\_APC\_OPROC\_quantum\_by\_chapter

from QRbySC\_Part3\_2

;

**quit**;

/\*combine detail and summary \*/

**proc** **append** base=QRbySC\_Part3\_2 data=QRbySC\_Part3\_2\_2

;

**run**;

**quit**;

/\*add row\_no \*/

**proc** **sql** noprint;

create table QRbySC\_Part3\_3

as

select

monotonic() as row\_no,

a.\*

from QRbySC\_Part3\_2 as a

order by row\_no

;

**quit**;

/\*Join two table by row\_no \*/

**proc** **sql** noprint;

create table QRbySC\_Part3

as

select

a.row\_no,

a.HRG,

a.Description,

a.OPROC\_currency\_applicable,

a.P0\_OPROC,

a.P0\_DC,

a.P0\_EL,

a.P0\_NE,

a.P1\_OPROC\_TA,

a.P1\_DC\_TA,

a.P1\_EL\_TA,

a.P1\_NE\_TA,

a.P1\_NE\_AT\_less\_SSEM,

a.P1\_LSP\_days,

a.P1\_SSEM,

a.P2\_OPROC,

a.P2\_DC,

a.P2\_EL,

a.P2\_NE,

a.P2\_LSP,

a.P2\_SSEM\_Banding,

a.P2\_SSEM\_Pct,

a.P2\_SSEM\_Tariff,

a.P3\_Sub\_Chapter,

a.P3\_OPROC\_quantum\_by\_HRG,

a.P3\_APC\_Quantum\_by\_HRG,

b.P3\_Sub\_\_Chapter,

b.P3\_OPROC\_quantum\_by\_SubChapter,

b.P3\_APC\_quantum\_by\_SubChapter,

b.P3\_APC\_OPROC\_quantum\_by\_chapter

from QRbySC\_Part3\_1 as a

left join QRbySC\_Part3\_3 as b

on a.row\_no=b.row\_no

order by a.row\_no

;

**quit**;

/\*Part3 -- Total quantum by subchapter pre quantum reconciliation (QR1) End here \*/

/\*Part4- Prices post inflation, efficiency and CNST adjustments (prices before manual adjustments)-- Start from here \*/

**proc** **sql** noprint;

create table QRbySC\_Part4

as

select

a.row\_no,

a.HRG,

a.Description,

a.OPROC\_currency\_applicable,

a.P0\_OPROC,

a.P0\_DC,

a.P0\_EL,

a.P0\_NE,

a.P1\_OPROC\_TA,

a.P1\_DC\_TA,

a.P1\_EL\_TA,

a.P1\_NE\_TA,

a.P1\_NE\_AT\_less\_SSEM,

a.P1\_LSP\_days,

a.P1\_SSEM,

a.P2\_OPROC,

a.P2\_DC,

a.P2\_EL,

a.P2\_NE,

a.P2\_LSP,

a.P2\_SSEM\_Banding,

a.P2\_SSEM\_Pct,

a.P2\_SSEM\_Tariff,

a.P3\_Sub\_Chapter,

a.P3\_OPROC\_quantum\_by\_HRG,

a.P3\_APC\_Quantum\_by\_HRG,

a.P3\_Sub\_\_Chapter,

a.P3\_OPROC\_quantum\_by\_SubChapter,

a.P3\_APC\_quantum\_by\_SubChapter,

a.P3\_APC\_OPROC\_quantum\_by\_chapter,

b.P6\_OPROC as P4\_OPROC,

b.P6\_DC as P4\_DC,

b.P6\_EL as P4\_EL,

b.P6\_NE as P4\_NE,

c.LSP\_Uplift18\_19 as P4\_Long\_Stay\_Payment,

case

when d.Eligible\_for\_SSEM=**1** then d.SSEM\_Banding

else **0**

End as P4\_SSEM\_Banding,

case

when Calculated P4\_SSEM\_Banding =**0** then **0**

when Calculated P4\_SSEM\_Banding=**.** then **.**

else e.Tariff\_Pct

End as P4\_SSEM\_Pct,

case

when Calculated P4\_SSEM\_Banding =**0** then **.**

when Calculated P4\_SSEM\_Banding=**.** then **.**

else (calculated P4\_SSEM\_Pct\*P4\_NE)

End as P4\_SSEM\_Tariff

from QRbySC\_Part3 as a

left join Modelout.STEP\_BY\_STEP\_OP\_DC\_EL\_NE as b

on a.HRG=b.SPELL\_HRG

left join Modelout.STEP\_BY\_STEP\_TRIMPOINT\_AND\_LSP as c

on a.HRG=c.SPELL\_HRG

left join APCHRGsEligibility as d

on a.HRG=d.HRG\_Code

left join Tariff\_R.SSEM\_BANDING(where=(Year="&Year.")) as e

on d.SSEM\_Banding=e.Band

order by a.row\_no

;

**quit**;

/\*Part4- Prices post inflation, efficiency and CNST adjustments (prices before manual adjustments)-- End here \*/

/\*Part5- Prices after manual adjustments -- Start from here \*/

**proc** **sql** noprint;

create table QRbySC\_Part5

as

select

a.row\_no,

a.HRG,

a.Description,

a.OPROC\_currency\_applicable,

a.P0\_OPROC,

a.P0\_DC,

a.P0\_EL,

a.P0\_NE,

a.P1\_OPROC\_TA,

a.P1\_DC\_TA,

a.P1\_EL\_TA,

a.P1\_NE\_TA,

a.P1\_NE\_AT\_less\_SSEM,

a.P1\_LSP\_days,

a.P1\_SSEM,

a.P2\_OPROC,

a.P2\_DC,

a.P2\_EL,

a.P2\_NE,

a.P2\_LSP,

a.P2\_SSEM\_Banding,

a.P2\_SSEM\_Pct,

a.P2\_SSEM\_Tariff,

a.P3\_Sub\_Chapter,

a.P3\_OPROC\_quantum\_by\_HRG,

a.P3\_APC\_Quantum\_by\_HRG,

a.P3\_Sub\_\_Chapter,

a.P3\_OPROC\_quantum\_by\_SubChapter,

a.P3\_APC\_quantum\_by\_SubChapter,

a.P3\_APC\_OPROC\_quantum\_by\_chapter,

a.P4\_OPROC,

a.P4\_DC,

a.P4\_EL,

a.P4\_NE,

a.P4\_Long\_Stay\_Payment,

a.P4\_SSEM\_Banding,

a.P4\_SSEM\_Pct ,

a.P4\_SSEM\_Tariff,

b.P8\_OPROC as P5\_OPROC,

b.P8\_DC as P5\_DC,

b.P8\_EL as P5\_EL,

b.P8\_NE as P5\_NE,

c.LSP\_Uplift18\_19 as P5\_Long\_Stay\_Payment,

a.P4\_SSEM\_Banding as P5\_SSEM\_Banding,

a.P4\_SSEM\_Pct as P5\_SSEM\_Pct,

case

when P4\_SSEM\_Banding =**0** then **.**

when P4\_SSEM\_Banding=**.** then **.**

else (P4\_SSEM\_Pct\*b.P8\_NE)

End as P5\_SSEM\_Tariff

from QRbySC\_Part4 as a

left join Modelout.STEP\_BY\_STEP\_OP\_DC\_EL\_NE as b

on a.HRG=b.SPELL\_HRG

left join Modelout.STEP\_BY\_STEP\_TRIMPOINT\_AND\_LSP as c

on a.HRG=c.SPELL\_HRG

order by a.row\_no

;

**quit**;

/\*Part5- Prices after manual adjustments -- End here \*/

/\* Part6 - Total quantum by subchapter before manual adjustment -- Start from here \*/

**proc** **sql** noprint;

create table QRbySC\_Part6\_1

as

select

a.row\_no,

a.HRG,

a.Description,

a.OPROC\_currency\_applicable,

a.P0\_OPROC,

a.P0\_DC,

a.P0\_EL,

a.P0\_NE,

a.P1\_OPROC\_TA,

a.P1\_DC\_TA,

a.P1\_EL\_TA,

a.P1\_NE\_TA,

a.P1\_NE\_AT\_less\_SSEM,

a.P1\_LSP\_days,

a.P1\_SSEM,

a.P2\_OPROC,

a.P2\_DC,

a.P2\_EL,

a.P2\_NE,

a.P2\_LSP,

a.P2\_SSEM\_Banding,

a.P2\_SSEM\_Pct,

a.P2\_SSEM\_Tariff,

a.P3\_Sub\_Chapter,

a.P3\_OPROC\_quantum\_by\_HRG,

a.P3\_APC\_Quantum\_by\_HRG,

a.P3\_Sub\_\_Chapter,

a.P3\_OPROC\_quantum\_by\_SubChapter,

a.P3\_APC\_quantum\_by\_SubChapter,

a.P3\_APC\_OPROC\_quantum\_by\_chapter,

a.P4\_OPROC,

a.P4\_DC,

a.P4\_EL,

a.P4\_NE,

a.P4\_Long\_Stay\_Payment,

a.P4\_SSEM\_Banding,

a.P4\_SSEM\_Pct ,

a.P4\_SSEM\_Tariff,

a.P5\_OPROC,

a.P5\_DC,

a.P5\_EL,

a.P5\_NE,

a.P5\_Long\_Stay\_Payment,

a.P5\_SSEM\_Banding,

a.P5\_SSEM\_Pct,

a.P5\_SSEM\_Tariff ,

a.P3\_Sub\_Chapter as P6\_Sub\_Chapter,

coalesce(a.P1\_OPROC\_TA\*a.P4\_OPROC,**.**) as P6\_OPROC\_Quantum\_by\_HRG ,

case

when a.P4\_SSEM\_Tariff=**0** then sum(a.P4\_DC\*a.P1\_DC\_TA, a.P4\_EL\*a.P1\_EL\_TA,a.P4\_NE\*a.P1\_NE\_AT\_less\_SSEM,a.P4\_Long\_Stay\_Payment\*a.P1\_LSP\_days)

else sum(a.P4\_DC\*a.P1\_DC\_TA, a.P4\_EL\*a.P1\_EL\_TA,a.P4\_NE\*a.P1\_NE\_AT\_less\_SSEM,a.P4\_Long\_Stay\_Payment\*a.P1\_LSP\_days, a.P4\_SSEM\_Tariff\*a.P1\_SSEM)

end as P6\_APC\_Quantum\_by\_HRG

from QRbySC\_Part5 as a

order by a.row\_no

;

**quit**;

/\*sub-chapter table \*/

/\*APC\_Sub\_Chapter \*/

/\*OPROC quantum by subchapter\*/

**proc** **sql** noprint;

create table P6\_OPROC\_quantum\_by\_subchapter

as

select

P6\_Sub\_Chapter,

sum(P6\_OPROC\_quantum\_by\_HRG) as P6\_OPROC\_quantum\_by\_SubChapter

from QRbySC\_Part6\_1

group by P6\_Sub\_Chapter

order by P6\_Sub\_Chapter

;

**quit**;

/\*APC quantum by subchapter\*/

**proc** **sql** noprint;

create table P6\_APC\_quantum\_by\_subchapter

as

select

P6\_Sub\_Chapter,

sum(P6\_APC\_Quantum\_by\_HRG) as P6\_APC\_quantum\_by\_SubChapter

from QRbySC\_Part6\_1

group by P6\_Sub\_Chapter

order by P6\_Sub\_Chapter

;

**quit**;

**proc** **sql** noprint;

create table QRbySC\_Part6\_2

as

select

a.HRGSubchapter as P6\_Sub\_\_Chapter,

b.P6\_OPROC\_quantum\_by\_SubChapter ,

c.P6\_APC\_quantum\_by\_SubChapter ,

sum(b.P6\_OPROC\_quantum\_by\_SubChapter,c.P6\_APC\_quantum\_by\_SubChapter) as P6\_APC\_OPROC\_quantum\_by\_chapter

from APC\_Sub\_Chapter as a

left join

P6\_OPROC\_quantum\_by\_subchapter as b

on a.HRGSubchapter=b.P6\_Sub\_Chapter

left join

P6\_APC\_quantum\_by\_subchapter as c

on a.HRGSubchapter=c.P6\_Sub\_Chapter

order by a.HRGSubchapter

;

**quit**;

/\* Total Summary \*/

**proc** **sql** noprint;

create table QRbySC\_Part6\_2\_2

as

select

"Total" as P6\_Sub\_\_Chapter length=**10**,

sum(P6\_OPROC\_quantum\_by\_SubChapter) as P6\_OPROC\_quantum\_by\_SubChapter,

sum(P6\_APC\_quantum\_by\_SubChapter) as P6\_APC\_quantum\_by\_SubChapter,

sum(P6\_APC\_OPROC\_quantum\_by\_chapter) as P6\_APC\_OPROC\_quantum\_by\_chapter

from QRbySC\_Part6\_2

;

**quit**;

/\*combine detail and summary \*/

**proc** **append** base=QRbySC\_Part6\_2 data=QRbySC\_Part6\_2\_2

;

**run**;

**quit**;

**proc** **sql** noprint;

create table QRbySC\_Part6\_3

as

select

monotonic() as row\_no,

a.\*

from QRbySC\_Part6\_2 as a

order by row\_no

;

**quit**;

/\*Join two table by row\_no \*/

**proc** **sql** noprint;

create table QRbySC\_Part6

as

select

a.row\_no,

a.HRG,

a.Description,

a.OPROC\_currency\_applicable,

a.P0\_OPROC,

a.P0\_DC,

a.P0\_EL,

a.P0\_NE,

a.P1\_OPROC\_TA,

a.P1\_DC\_TA,

a.P1\_EL\_TA,

a.P1\_NE\_TA,

a.P1\_NE\_AT\_less\_SSEM,

a.P1\_LSP\_days,

a.P1\_SSEM,

a.P2\_OPROC,

a.P2\_DC,

a.P2\_EL,

a.P2\_NE,

a.P2\_LSP,

a.P2\_SSEM\_Banding,

a.P2\_SSEM\_Pct,

a.P2\_SSEM\_Tariff,

a.P3\_Sub\_Chapter,

a.P3\_OPROC\_quantum\_by\_HRG,

a.P3\_APC\_Quantum\_by\_HRG,

a.P3\_Sub\_\_Chapter,

a.P3\_OPROC\_quantum\_by\_SubChapter,

a.P3\_APC\_quantum\_by\_SubChapter,

a.P3\_APC\_OPROC\_quantum\_by\_chapter,

a.P4\_OPROC,

a.P4\_DC,

a.P4\_EL,

a.P4\_NE,

a.P4\_Long\_Stay\_Payment,

a.P4\_SSEM\_Banding,

a.P4\_SSEM\_Pct ,

a.P4\_SSEM\_Tariff,

a.P5\_OPROC,

a.P5\_DC,

a.P5\_EL,

a.P5\_NE,

a.P5\_Long\_Stay\_Payment,

a.P5\_SSEM\_Banding,

a.P5\_SSEM\_Pct,

a.P5\_SSEM\_Tariff,

a.P6\_Sub\_Chapter,

a.P6\_OPROC\_Quantum\_by\_HRG format=Best12.10,

a.P6\_APC\_Quantum\_by\_HRG format=best12.10,

b.P6\_Sub\_\_Chapter,

b.P6\_OPROC\_quantum\_by\_SubChapter format=best12.10 ,

b.P6\_APC\_quantum\_by\_SubChapter format=best12.10,

b.P6\_APC\_OPROC\_quantum\_by\_chapter

from QRbySC\_Part6\_1 as a

left join QRbySC\_Part6\_3 as b

on a.row\_no=b.row\_no

order by a.row\_no

;

**quit**;

/\* Part6 - Total quantum by subchapter before manual adjustment -- End here \*/

/\*Join two table by row\_no \*/

**proc** **sql** noprint;

create table Modelout.QUANTUM\_RECONCILIATION\_BySubChap

as

select

a.row\_no,

a.HRG,

a.Description,

a.OPROC\_currency\_applicable,

a.P0\_OPROC,

a.P0\_DC,

a.P0\_EL,

a.P0\_NE,

a.P1\_OPROC\_TA,

a.P1\_DC\_TA,

a.P1\_EL\_TA,

a.P1\_NE\_TA,

a.P1\_NE\_AT\_less\_SSEM,

a.P1\_LSP\_days,

a.P1\_SSEM,

a.P2\_OPROC,

a.P2\_DC,

a.P2\_EL,

a.P2\_NE,

a.P2\_LSP,

a.P2\_SSEM\_Banding,

a.P2\_SSEM\_Pct format=Percent8.0,

a.P2\_SSEM\_Tariff,

a.P3\_Sub\_Chapter,

a.P3\_OPROC\_quantum\_by\_HRG,

a.P3\_APC\_Quantum\_by\_HRG,

a.P3\_Sub\_\_Chapter,

a.P3\_OPROC\_quantum\_by\_SubChapter,

a.P3\_APC\_quantum\_by\_SubChapter,

a.P3\_APC\_OPROC\_quantum\_by\_chapter,

a.P4\_OPROC,

a.P4\_DC,

a.P4\_EL,

a.P4\_NE,

a.P4\_Long\_Stay\_Payment,

a.P4\_SSEM\_Banding,

a.P4\_SSEM\_Pct format=Percent8.0,

a.P4\_SSEM\_Tariff,

a.P5\_OPROC,

a.P5\_DC,

a.P5\_EL,

a.P5\_NE,

a.P5\_Long\_Stay\_Payment,

a.P5\_SSEM\_Banding,

a.P5\_SSEM\_Pct format=Percent8.0,

a.P5\_SSEM\_Tariff,

a.P6\_Sub\_Chapter,

a.P6\_OPROC\_Quantum\_by\_HRG ,

a.P6\_APC\_Quantum\_by\_HRG,

a.P6\_Sub\_\_Chapter,

a.P6\_OPROC\_quantum\_by\_SubChapter ,

a.P6\_APC\_quantum\_by\_SubChapter ,

a.P6\_APC\_OPROC\_quantum\_by\_chapter

from QRbySC\_Part6 as a

order by a.row\_no

;

**quit**;

/\*Part-19 - Final total quantum by chapter after implementing all uplifts- End here \*/

/\*Delete temp tables \*/

**proc** **datasets** noprint;

delete

QRbySC\_Part1\_1

QRbySC\_Part2\_1

QRbySC\_Part3\_1

QRbySC\_Part3\_2

QRbySC\_Part3\_3

QRbySC\_Part6\_1

QRbySC\_Part6\_2

QRbySC\_Part6\_2\_2

QRbySC\_Part6\_3

;

**run**;

**quit**;

libname Savedata "\\irnarch\sas\_data\PID\_Proj\_1718\APC\APC\_PIDTest\sasdata";

%let SQLInputLib=SQL\_INP;

LIBNAME &SQLInputLib. ODBC DATASRC=Pbr1718\_\_28 SCHEMA=Input;

LIBNAME bprRefCo ODBC DATASRC=DSN\_Pbr\_Repository\_New SCHEMA=RefCost;

/\*Part 1------------------------------ Grubbs test QA -- Start from here \*/

/\* Grubbs test results \*/

**proc** **import** datafile="\\irnarch\sas\_data\PID\_Proj\_1718\APC\APC\_PIDTest\data\Grubbs flag.csv"

out=PID\_GrubFlag\_1 dbms=csv replace;

getnames=Yes;

**run**;

**data** PID\_GrubFlag;

set PID\_GrubFlag\_1;

servicecodeM=put(serviceCode, **3.**);

**run**;

/\*SAS Flat after grab \*/

**data** Savedata.sas\_Flat\_after\_Grabb;

set Data\_After\_Grubbs;

**run**;

**proc** **sql**;

create table Savedata.Grub\_compare\_Flat

as

select

a.SUPPLIER\_TYPE as SupplierType\_A,

a.PROCODET as OrgCode\_A,

a.POD as departmentcode\_A,

a.TFC as ServiceCode\_A,

a.HRG as HRG\_A,

b.suppliertype as suppliertype\_B,

b.orgcode as orgcode\_B,

b.departmentcode as departmentcode\_B,

b.servicecode as servicecode\_B,

b.hrg as hrg\_B,

a.flaggrubbs as flaggrubbs\_A,

b.grubbs as grubbs\_B,

a.flaggrubbs-b.grubbs as grubbs\_dif,

a.\*

from Savedata.sas\_Flat\_after\_Grabb as a

left join PID\_GrubFlag as b

on

a.SUPPLIER\_TYPE=b.suppliertype and

a.PROCODET=b.orgcode and

a.POD=b.departmentcode and

a.TFC=b.servicecodeM and

a.HRG=b.hrg

;

**quit**;

/\*Part 1------------------------------ Grubbs test QA -- End here \*/

/\*Part 2------------------------------ QA :Flat file RefCost with Clean flag to IT -- Start from here \*/

/\*SAS data to IT (refer to) step 1-4 \*/

**data** Savedata.SASFlat\_RefCost\_to\_infoserve;

set RefCost\_data\_to\_infoserve;

**run**;

/\* sent to is

Savedata.PID\_RC\_DATA\_TO\_IT

\*/

**proc** **sql**;

create table clean\_flag\_diff

as

select

a.Org\_code as Org\_code\_A,

a.Supplier\_type as Supplier\_type\_A,

a.Department\_code as Department\_code\_A,

a.Service\_code as Service\_code\_A,

a.Currency\_code as Currency\_code\_A,

b.Org\_Code as Org\_Code\_B,

b.SUPPLIER\_TYPE as SUPPLIER\_TYPE\_B,

b.Department\_code as Department\_code\_B,

b.Service\_code as Service\_code\_B,

b.Currency\_code as Currency\_code\_B,

a.Unit\_cost as Unit\_cost\_A,

a.Cleaning\_flag as Cleaning\_flag\_A,

b.Unit\_Cost as Unit\_Cost\_B,

b.cleaning\_flag as cleaning\_flag\_B,

Case

when a.Unit\_cost = b.Unit\_cost then "Y"

Else "N"

end as flag\_compare,

a.Activity as Activity\_A,

a.Bed\_days as Bed\_days\_A,

a.Mean as Mean\_A,

a.Actual\_cost as Actual\_cost\_A,

a.Expected\_cost as Expected\_cost\_A,

a.Mapping\_pot as Mapping\_pot\_A,

b.Unit\_Cost as Unit\_Cost\_B,

b.Activity as Activity\_B,

b.Bed\_days as Bed\_days\_B,

b.Mean as Mean\_B,

b.Actual\_Cost as Actual\_Cost\_B,

b.Expected\_Cost as Expected\_Cost\_B,

b.Mapping\_Pot as Mapping\_Pot\_B

from

Savedata.PID\_RC\_DATA\_TO\_IT as a

left join

Savedata.SASFlat\_RefCost\_to\_infoserve as b

on a.Org\_code=b.Org\_code

and a.Supplier\_type=b.Supplier\_type

and a.Department\_code=b.Department\_code

and a.Service\_code=b.Service\_code

and a.Currency\_code=b.Currency\_code

;

**quit**;

/\*Part 2------------------------------ QA :RefCost with Clean flag to IT -- End here \*/

/\* Part 3 ------------------------------------ Acute Refrence cost QA --------------- Start from here \*/

/\* Refcost from SAS data clean \*/

**proc** **sql**;

create table Savedata.SAS\_Flat\_APC\_Cleaned\_Refcost\_F

as

select

YEARORDER,

**.** as FK\_PERIODS\_ID,

FK\_ORGS\_PROV\_ID,

SUPPLIER\_TYPE,

DEPARTMENT,

SERVICE,

CURRENCY,

UNIT\_COST,

Un\_Cost\_RemMFF,

FCE,

BED\_DAYS,

EXCESS\_BED\_DAYS\_UNIT\_COST,

EXCESS\_BED\_DAYS\_ACTIVITY

from Final\_APC\_Cleaned\_Refcost\_F

order by FK\_ORGS\_PROV\_ID, DEPARTMENT, SERVICE, CURRENCY

;

**run**;

**data** Savedata.SAS\_Flat\_APC\_Cleaned\_Refcost\_F;

set Savedata.SAS\_Flat\_APC\_Cleaned\_Refcost\_F;

if DEPARTMENT="EI" then DEPARTMENT="EL";

if DEPARTMENT="NEI\_L" then DEPARTMENT="NEL";

if DEPARTMENT="NEI\_S" then DEPARTMENT="NES";

**run**;

/\* Refcost from SQL server data clean \*/

**proc** **sql**;

create table SQL\_RefCost\_Acute

as

select a.\*

from &SQLInputLib.**.R**efCostAcute as a

order by FK\_ORGS\_PROV\_ID, DEPARTMENT, SERVICE, CURRENCY

;

**quit**;

**data** SQL\_RefCost\_Acute;

set SQL\_RefCost\_Acute;

if BED\_DAYS=**.** then BED\_DAYS=**0**;

**run**;

/\*Compare with Refcost from SAS vs. Refcost from SQL server (input.RefCostAcute ) \*/

**proc** **sql**;

create table RefCostAcute\_Diff

as

select

a.YEARORDER as YEARORDER\_A,

a.FK\_PERIODS\_ID as FK\_PERIODS\_ID\_A,

a.FK\_ORGS\_PROV\_ID as FK\_ORGS\_PROV\_ID\_A,

a.SUPPLIER\_TYPE as SUPPLIER\_TYPE\_A,

a.DEPARTMENT as DEPARTMENT\_A,

a.SERVICE as SERVICE\_A,

a.CURRENCY as CURRENCY\_A,

b.FK\_ORGS\_PROV\_ID as FK\_ORGS\_PROV\_ID\_B,

b.SUPPLIER\_TYPE as SUPPLIER\_TYPE\_B,

b.DEPARTMENT as DEPARTMENT\_B,

b.SERVICE as SERVICE\_B,

b.CURRENCY as CURRENCY\_B,

a.UNIT\_COST as UNIT\_COST\_A,

b.UNIT\_COST as UNIT\_COST\_B,

Case

when a.Unit\_cost = b.Unit\_cost then "Y"

Else "N"

end as UC\_compare,

a.FCE as FCE\_A,

b.FCE as FCE\_B,

Case

when a.FCE =b.FCE then "Y"

Else "N"

end as FCE\_compare,

a.BED\_DAYS as BED\_DAYS\_A,

b.BED\_DAYS as BED\_DAYS\_B,

Case

when a.BED\_DAYS=b.BED\_DAYS then "Y"

Else "N"

end as BED\_DAYS\_compare,

a.EXCESS\_BED\_DAYS\_UNIT\_COST as EXCESS\_BED\_DAYS\_UNIT\_COST\_A,

b.EXCESS\_BED\_DAYS\_UNIT\_COST as EXCESS\_BED\_DAYS\_UNIT\_COST\_B,

Case

when a.EXCESS\_BED\_DAYS\_UNIT\_COST=b.EXCESS\_BED\_DAYS\_UNIT\_COST then "Y"

Else "N"

end as EBD\_UNIT\_COSTcompare,

a.EXCESS\_BED\_DAYS\_ACTIVITY as EXCESS\_BED\_DAYS\_ACTIVITY\_A,

b.EXCESS\_BED\_DAYS\_ACTIVITY as EXCESS\_BED\_DAYS\_ACTIVITY\_B,

Case

when a.EXCESS\_BED\_DAYS\_ACTIVITY=b.EXCESS\_BED\_DAYS\_ACTIVITY then "Y"

Else "N"

end as EBD\_Activity\_compare,

b.YEARORDER as YEARORDER\_B,

b.FK\_PERIODS\_ID as FK\_PERIODS\_ID\_B

from Savedata.SAS\_Flat\_APC\_Cleaned\_Refcost\_F as a

left join SQL\_RefCost\_Acute as b

on a.FK\_ORGS\_PROV\_ID=b.FK\_ORGS\_PROV\_ID

and a.SUPPLIER\_TYPE=b.SUPPLIER\_TYPE

and a.DEPARTMENT=b.DEPARTMENT

and a.SERVICE=b.SERVICE

and a.CURRENCY=b.CURRENCY

order by a.FK\_ORGS\_PROV\_ID, a.DEPARTMENT, a.SERVICE, a.CURRENCY

;

**quit**;

/\*Compare the results between SAS Data Clean and SQL server data \*/

**proc** **compare** base=Savedata.SAS\_Flat\_APC\_Cleaned\_Refcost\_F compare=SQL\_RefCost\_Acute;

var FK\_ORGS\_PROV\_ID SERVICE DEPARTMENT CURRENCY UNIT\_COST FCE BED\_DAYS EXCESS\_BED\_DAYS\_UNIT\_COST EXCESS\_BED\_DAYS\_ACTIVITY YEARORDER SUPPLIER\_TYPE;

with FK\_ORGS\_PROV\_ID SERVICE DEPARTMENT CURRENCY UNIT\_COST FCE BED\_DAYS EXCESS\_BED\_DAYS\_UNIT\_COST EXCESS\_BED\_DAYS\_ACTIVITY YEARORDER SUPPLIER\_TYPE;

**run**;

**proc** **sql**;

select count(\*) as numberofrow

from &SQLInputLib.**.R**efCostAcute;

**quit**;

**proc** **sql**;

select count(\*) as numberofrow

from Savedata.SAS\_Flat\_APC\_Cleaned\_Refcost\_F;

**quit**;

/\* Part 3 --- Acute Refrence cost QA --------------- End here \*/

/\*--------------------------------------------------------------\*/

\* Excel\_Step3\_1 Long Stay Payment \*/

/\* Version: 1.02 \*/

/\* Author: PID \*/

/\* Date: July 2017 \*/

/\*--------------------------------------------------------------\*/

/\* Input: \*/

/\* (1)[HES\_Grouped].[APC\_GroupedData] \*/

/\* (2)Tariff\_R.HRG\_4\_PROPOSED\_NEWCD\_ELIGI \*/

/\* (3)Tariff\_R.NHSE\_PRULES\_EQUALIZATION\_HRG4 \*/

/\* (4)SQLStage.APC\_NICE \*/

/\* \*/

/\* Output: \*/

/\* (1)Modelout.Long\_Stay\_Payment \*/

/\* \*/

/\* Description: \*/

/\* Create required data for Step1: Long Stay Payment \*/

/\* \*/

/\* \*/

/\*--------------------------------------------------------------\*/

%let Cost\_base\_adjustment=0.00;

/\* Create table Y18\_and\_under\_proportion\_by\_HRG \*/

**proc** **sql** noprint;

create table T18andUnderproportionbyHRG\_1

as

select

SpellHRG,

SUM(

CASE WHEN (STARTAGE < **19**) THEN **1**

ELSE **0** END

) AS AG18\_or\_under,

SUM(

CASE WHEN (STARTAGE > **18**) THEN **1**

ELSE **0** END

) AS AG19\_or\_over,

COUNT(Epikey) AS Activity

FROM bprhesg.APC\_GroupedData

WHERE (GrouperRunID = &GrouperRunID. AND Excluded = 'N' AND

(substr(LEFT(PROCODET),**1**,**1**) IN ('R','T'))

AND ADMIMETH <> '98' AND ADMIMETH <> '99')

and STARTAGE < **120**

Group bY SpellHRG

Order by SPellHRG

;

**quit**;

**proc** **sql** noprint;

create table T18andUnderproportionbyHRG

as

select

SpellHRG,

AG18\_or\_under,

AG19\_or\_over,

Activity,

AG18\_or\_under / Activity as Proportion /\*format=8.2\*/

from T18andUnderproportionbyHRG\_1

order by SpellHRG

;

**quit**;

/\*Import NHSE\_Currency\_specification table -----------Start from here \*/

/\*Note: SB97Z is a special case which should be list in final table \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*\*/

**proc** **sql**;

create table NHSE\_Currency\_specificationSAS\_1

as

select a.\*

from Tariff\_R.HRG\_ELIGIBILITY as a

where a.Area="APC & OPROC" and Year="&FYear."

and Combined\_day\_case\_\_\_ordinary\_ele= **1** and Non\_elective\_spell\_tariff =**1** /\*Added 29/08/2017 --PID for Mandatory IP=1 in Excel model\*/

;

**quit**;

**data** NHSE\_Currency\_specification\_SAS;

set NHSE\_Currency\_specificationSAS\_1

;

**run**;

**data** NHSE\_Currency\_specification\_SAS;

set NHSE\_Currency\_specification\_SAS;

if HRG="EY12A" then Outpatient\_procedure\_tariff=**1**; /\*No reference cost for EY12A for FY14\_15 \*/

/\*if HRG in (&ZeroPriceHRGlist.) then Outpatient\_procedure\_tariff=1;\*/

**run**;

/\*import NHSE\_Currency\_specification\_SAS table end here\*/

**proc** **sql** noprint;

create table T\_LSP\_part1 as

select

a.HRG,

a.Description as HRG\_Name,

substr(a.HRG,**1**,**1**) as Chapter format=$1.,

b.Proportion as Proportion\_Children\_Activity,

case

when find(a.Description, "Congenital", "i")>**0** then **1**

Else **0**

End as Congenital\_HRG\_Flag,

case

when (b.Proportion+ calculated Congenital\_HRG\_Flag)>=**1** or (substr(a.HRG,**1**,**1**)="P") then **1**

else **0**

End as AG18\_YearsandUnder\_HRG\_Flag,

strip(calculated Chapter)||strip(put(calculated AG18\_YearsandUnder\_HRG\_Flag,**2.**)) as LU,

c.Zero\_Price\_Tariff as Zero\_price\_HRG

from NHSE\_Currency\_specification\_SAS as a

left join T18andUnderproportionbyHRG as b

on a.HRG=b.SpellHRG

left join Tariff\_R.nhse\_equalization\_rules\_1920 (where=(Year="&FYear.")) as c /\*PID ADDED \*/

on a.HRG=c.HRG

order by a.HRG

;

**quit**;

**proc** **sql**;

create table APC\_Nice\_EBD\_ACT\_TC

as

select

a.HRG,

sum(a.CLEAN\_EBD\_ACT) as CLEAN\_EBD\_ACT,

sum(a.CLEAN\_EBD\_TC) as CLEAN\_EBD\_TC

from &StageLib.**.A**PC\_NICE as a

group by a.HRG

order by HRG

;

**quit**;

/\*Add EBD\_ACT\_TC to Main table \*/

**proc** **sql** noprint;

create table T\_LSP\_part2

as

select

a.HRG,

a.HRG\_Name,

a.Chapter,

a.Proportion\_Children\_Activity,

a.Congenital\_HRG\_Flag,

a.AG18\_YearsandUnder\_HRG\_Flag,

a.LU,

b.CLEAN\_EBD\_ACT,

b.CLEAN\_EBD\_TC,

a.Zero\_price\_HRG

from T\_LSP\_part1 as a

left join APC\_Nice\_EBD\_ACT\_TC as b

on a.HRG=b.HRG

order by a.HRG

;

**quit**;

/\*LSP unit cost calculation \*/

**proc** **sql** noprint;

create table LSP\_Unit\_cost

as

select

a.LU,

coalesce(sum(a.CLEAN\_EBD\_TC)/sum(a.CLEAN\_EBD\_ACT),**.**) as LSP\_UC

from T\_LSP\_part2 as a

group by a.LU

order by a.LU

;

**quit**;

/\* add LSP unit cost to main table\*/

**proc** **sql** noprint;

create table T\_LSP\_part3

as

select

a.HRG,

a.HRG\_Name,

a.Chapter,

a.Proportion\_Children\_Activity,

a.Congenital\_HRG\_Flag,

a.AG18\_YearsandUnder\_HRG\_Flag,

a.LU,

a.CLEAN\_EBD\_ACT,

a.CLEAN\_EBD\_TC,

b.LSP\_UC,

a.Zero\_price\_HRG

from T\_LSP\_part2 as a

left join LSP\_Unit\_cost as b

on a.LU=b.LU

order by

a.HRG

;

**quit**;

**data** T\_LSP\_part3\_lag;

set T\_LSP\_part3;

lag\_LSP\_UC=lag(LSP\_UC) ;

**run**;

/\*Limiting within £100 to £500 \*/

**proc** **sql**;

create table T\_LSP\_part4

as

select

a.HRG,

a.HRG\_Name,

a.Chapter,

a.Proportion\_Children\_Activity,

a.Congenital\_HRG\_Flag,

a.AG18\_YearsandUnder\_HRG\_Flag,

a.LU,

a.CLEAN\_EBD\_ACT,

a.CLEAN\_EBD\_TC,

a.LSP\_UC,

case

when a.LSP\_UC=**.** then lag\_LSP\_UC

when a.LSP\_UC>=**500** then **500**

when **0**<a.LSP\_UC<=**100** then **100**

else LSP\_UC

End as Limit\_within\_100\_to\_500 ,

a.Zero\_price\_HRG

from T\_LSP\_part3\_lag as a

order by

a.HRG

;

**quit**;

/\*Zero price adjustment - produce Long Stay Payment table \*/

**proc** **sql** noprint;

create table Modelout.Long\_Stay\_Payment

as

select

a.HRG,

a.HRG\_Name,

a.Chapter,

a.Proportion\_Children\_Activity,

a.Congenital\_HRG\_Flag,

a.AG18\_YearsandUnder\_HRG\_Flag,

a.LU,

a.CLEAN\_EBD\_ACT ,

a.CLEAN\_EBD\_TC ,

a.LSP\_UC,

a.Limit\_within\_100\_to\_500,

a.Zero\_price\_HRG,

case

when a.Zero\_price\_HRG=**1** then **0**

else a.Limit\_within\_100\_to\_500

end as Long\_Stay\_Payment

from T\_LSP\_part4 as a

order by a.HRG

;

**quit**;

/\*--------------------------------------------------------------\*/

\* Excel\_Step3\_2 STEP BY STEP TRIMPOINT and LSP \*/

/\* Version: 1.02 \*/

/\* Author: PID \*/

/\* Date: September 2017 \*/

/\*--------------------------------------------------------------\*/

/\* Input: \*/

/\* (1)APCSTAGE.APC\_Tariff \*/

/\* (2)Tariff\_R.APC\_OPROC\_SMOOTHING \*/

/\* \*/

/\* \*/

/\* \*/

/\* Output: \*/

/\* (1)Modelout.STEP\_BY\_STEP\_TRIMPOINT\_and\_LSP \*/

/\* \*/

/\* Description: \*/

/\* Create required data for Step2: STEP BY STEP TRIMPOINT \*/

/\* and Long Stay Payment \*/

/\* \*/

/\* \*/

/\*--------------------------------------------------------------\*/

/\*Part1-"SQL output for trimpoints and modelled long stay payments" -- Start from here \*/

/\*Import NHSE\_Currency\_specification table \*/

/\*Note: SB97Z is a special case which should be list in final table \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

libname MATrimP "\\irnarch\sas\_data\Tariff Rebuild\Model outputs pre\_MA\Trimpoints";

/\*Calculate QR1 factor ---------- Start from here \*/

/\* total APC+OPROC Quans from QR programm run --numerator -- on Top\*/

**proc** **sql** noprint;

select QuantumRecValues format=Best12.4 into :QR\_PRun\_APC

from Tariff\_R.QUANTUMRECONCILATION\_OUTPUT

where Year="&FYear." and POD="APC"

;

**quit**;

**proc** **sql** noprint;

select QuantumRecValues format=Best12.4 into :QR\_PRun\_OPROC

from Tariff\_R.QUANTUMRECONCILATION\_OUTPUT

where Year="&FYear." and POD="OPROC"

;

**quit**;

/\* total APC+OPROC Quans -- denominator -- at bottom\*/

**proc** **sql** noprint;

select OP\_APC\_TotalQT format=Best12.4,

OPROC\_TotalQT format=Best12.4,

APC\_TotalQT format=Best12.4

into :OPROCAPC\_TotalQT\_Bot,

:OPROC\_TotalQT\_Bot,

:APC\_TotalQT\_Bot

from Tariff\_R.APC\_QR1\_CAL\_1920

;

**quit**;

**data** QR1\_F\_Calculation;

QR\_PRun\_APC=&QR\_PRun\_APC.;

QR\_PRun\_OPROC=&QR\_PRun\_OPROC.;

QR\_PRunTotal\_Top=sum(QR\_PRun\_APC,QR\_PRun\_OPROC);

OPROC\_TotalQT\_Bot=&OPROC\_TotalQT\_Bot;

APC\_TotalQT\_Bot=&APC\_TotalQT\_Bot;

OPROCAPC\_TotalQT\_Bot=&OPROCAPC\_TotalQT\_BoT;

QR1\_Factor=(QR\_PRunTotal\_Top/OPROCAPC\_TotalQT\_Bot-**1**);

format QR1\_Factor Best12.8;

**run**;

**proc** **sql** noprint;

select QR1\_Factor into :QR1\_Factor

from QR1\_F\_Calculation;

**quit**;

/\*Calculate QR1 factor ---------- End here \*/

/\*NHSE\_Currency\_specification\_SAS table \*/

**proc** **sql** noprint;

create table TP\_and\_LSP\_Part1 as

select

a.HRG as Spell\_HRG,

substr(a.HRG,**1**,**1**) as SpellHRGChapter format=$1.,

substr(a.HRG,**1**,**2**) as SpellHRGSubChapter format=$2.,

a.Description as HRG\_Description,

b.EL\_Trim,

b.NE\_Trim,

c.Long\_Stay\_Payment

from NHSE\_Currency\_specification\_SAS as a

left join

&StageLib.**.A**PC\_Tariff as b

on a.HRG=b.SPELL\_HRG

left join Modelout.Long\_Stay\_Payment as c

on a.HRG=c.HRG

order by a.HRG

;

**quit**;

/\*Part1-"SQL output for trimpoints and modelled long stay payments" -- End here \*/

/\*Part2-"Minimum 5 day rule for trim points"- start from here \*/

**proc** **sql**;

create table TP\_and\_LSP\_Part2

as

select

a.Spell\_HRG,

a.SpellHRGChapter,

a.SpellHRGSubChapter,

a.HRG\_Description,

a.EL\_Trim,

a.NE\_Trim,

a.Long\_Stay\_Payment,

case

when a.EL\_Trim<**5** then **5**

Else a.EL\_Trim

End as EL\_Trim\_5DRule,

case

when a.NE\_Trim<**5** then **5**

Else a.NE\_Trim

End as NE\_Trim\_5DRule

from TP\_and\_LSP\_Part1 as a

order by a.Spell\_HRG

;

**quit**;

/\*Part2-"Minimum 5 day rule for trim points"- End here \*/

/\*Part3-"Manual Trimpoint" - Start From here \*/

**proc** **sql** noprint;

create table TP\_and\_LSP\_Part3

as

select

a.Spell\_HRG,

a.SpellHRGChapter,

a.SpellHRGSubChapter,

a.HRG\_Description,

a.EL\_Trim,

a.NE\_Trim,

a.Long\_Stay\_Payment,

a.EL\_Trim\_5DRule,

a.NE\_Trim\_5DRule,

b.MAtrim\_EL as Manual\_TP\_EL,

b.MAtrim\_NE as Manual\_TP\_NE

from TP\_and\_LSP\_Part2 as a

left join MATrimP.MA\_TRIMPOINTS\_OUTPUT as b

on a.Spell\_HRG=b.HRGCode

order by a.Spell\_HRG

;

**quit**;

/\*Part3-"Manual Trimpoint" - End here \*/

/\*Part4-"Semi-Automated Trimpoint" -Start from here \*/

/\* Create MA\_TrimPoint\_Semi-Automated table --- Start \*/

**proc** **sql** noprint;

create table TP\_and\_LSP\_Part4

as

select

a.Spell\_HRG,

a.SpellHRGChapter,

a.SpellHRGSubChapter,

a.HRG\_Description,

a.EL\_Trim,

a.NE\_Trim,

a.Long\_Stay\_Payment,

a.EL\_Trim\_5DRule,

a.NE\_Trim\_5DRule,

a.Manual\_TP\_EL,

a.Manual\_TP\_NE,

b.Semi\_MAtrim\_EL as DC\_EL\_SemiAuto,

b.Semi\_MAtrim\_NE as NE\_SemiAuto

from TP\_and\_LSP\_Part3 as a

left join MATrimP.MA\_TRIMPOINTS\_OUTPUT as b

on a.Spell\_HRG=b.HRGCode

order by a.Spell\_HRG

;

**quit**;

/\* Create MA\_TrimPoint\_Semi-Automated table --- End \*/

/\*Part4-"Semi-Automated Trimpoint" -End here \*/

/\*Part5-"Final trim point" -Start from here \*/

**proc** **sql** noprint;

create table TP\_and\_LSP\_Part5

as

select

Spell\_HRG,

SpellHRGChapter,

SpellHRGSubChapter,

HRG\_Description,

EL\_Trim,

NE\_Trim,

Long\_Stay\_Payment,

EL\_Trim\_5DRule,

NE\_Trim\_5DRule,

Manual\_TP\_EL,

Manual\_TP\_NE,

DC\_EL\_SemiAuto,

NE\_SemiAuto,

Finaltrim\_EL as Fin\_EL,

Finaltrim\_NE as Fin\_NE

from TP\_and\_LSP\_Part4 as a

left join MATrimP.MA\_TRIMPOINTS\_OUTPUT as b

on a.Spell\_HRG=b.HRGCode

order by a.Spell\_HRG

;

**quit**;

/\*Part5-"Final trim point" - End here \*/

/\* Part6 - "Initial long stay payments (LSP)" -- Start from here \*/

**proc** **sql** noprint;

create table TP\_and\_LSP\_Part6

as

select

a.Spell\_HRG,

a.SpellHRGChapter,

a.SpellHRGSubChapter,

a.HRG\_Description,

a.EL\_Trim,

a.NE\_Trim,

a.Long\_Stay\_Payment,

a.EL\_Trim\_5DRule,

a.NE\_Trim\_5DRule,

a.Manual\_TP\_EL,

a.Manual\_TP\_NE,

a.DC\_EL\_SemiAuto,

a.NE\_SemiAuto,

a.Fin\_EL,

a.Fin\_NE,

a.Long\_Stay\_Payment as Initial\_longStayPayment

from TP\_and\_LSP\_Part5 as a

left join Modelout.Long\_Stay\_Payment as b

on a.SPell\_HRG=b.HRG

order by a.Spell\_HRG

;

**quit**;

/\* Part6 - "Initial long stay payments (LSP)" -- End here \*/

/\*Part7-"LSP after QR1 implementation" -- Start from here \*/

**proc** **sql** noprint;

create table TP\_and\_LSP\_Part7\_LSP\_after\_QR1

as

select

a.Spell\_HRG,

a.SpellHRGChapter,

a.SpellHRGSubChapter,

a.HRG\_Description,

a.EL\_Trim,

a.NE\_Trim,

a.Long\_Stay\_Payment,

a.EL\_Trim\_5DRule,

a.NE\_Trim\_5DRule,

a.Manual\_TP\_EL,

a.Manual\_TP\_NE,

a.DC\_EL\_SemiAuto,

a.NE\_SemiAuto,

a.Fin\_EL,

a.Fin\_NE,

a.Initial\_longStayPayment,

a.Initial\_longStayPayment\*(**1**+&QR1\_Factor.) /\*format=8.0\*/ as LSP\_after\_QR1

from TP\_and\_LSP\_Part6 as a

order by a.Spell\_HRG

;

**quit**;

/\*Part7-"LSP after QR1 implementation" -- End here \*/

/\*Part8 -"Long stay payments after CB adjustment implementation" -- Start from here \*/

**proc** **sql** noprint;

create table TP\_and\_LSP\_Part8\_LSP\_after\_CB

as

select

a.Spell\_HRG,

a.SpellHRGChapter,

a.SpellHRGSubChapter,

a.HRG\_Description,

a.EL\_Trim,

a.NE\_Trim,

a.Long\_Stay\_Payment,

a.EL\_Trim\_5DRule,

a.NE\_Trim\_5DRule,

a.Manual\_TP\_EL,

a.Manual\_TP\_NE,

a.DC\_EL\_SemiAuto,

a.NE\_SemiAuto,

a.Fin\_EL,

a.Fin\_NE,

a.Initial\_longStayPayment,

a.LSP\_after\_QR1,

a.LSP\_after\_QR1\*(**1**+&Cost\_base\_adjustment.) /\*format=8.0\*/ as LSP\_after\_CB

from TP\_and\_LSP\_Part7\_LSP\_after\_QR1 as a

order by a.Spell\_HRG

;

**quit**;

/\*Part8 -"Long stay payments after CB adjustment implementation" -- End here \*/

/\*Part9-"Modelled LSP prices adjusted to a 18/19 basis " -- Start from here \*/

/\*Part9-" " -- Start from here \*/

/\* Transpose print adjustment table \*/

**PROC** **TRANSPOSE** DATA=Tariff\_R.I\_E

OUT=Trans\_I\_E (drop=Source )

PREFIX=Year

NAME=Source

LABEL=Adjustment\_Type

;

ID Year;

VAR Inflation Efficiency Inflation\_and\_Efficiency\_Combine;

**RUN**; **QUIT**;

/\*Calculate uplift total \*/

**data** PriceAdj\_InflaEffici;

set Trans\_I\_E;

label Adjustment\_Type =Adjustment\_Type;

TotalUplift18\_19=(Year17\_18+**1**)\*(Year18\_19+**1**)-**1**;

label TotalUplift18\_19="Total uplift up to 18/19 price levels";

**run**;

**proc** **sql** noprint;

select TotalUplift18\_19 into :InfEffTotalUplift18\_19

from PriceAdj\_InflaEffici

where Adjustment\_Type="Inflation and Efficiency Combined"

;

**quit**;

%let InfEffTotalUplift18\_19=&InfEffTotalUplift18\_19;

%put InfEffTotalUplift18\_19=&InfEffTotalUplift18\_19;

**proc** **sql** noprint;

create table TP\_and\_LSP\_Part9\_LSP\_adj\_1819

as

select

a.Spell\_HRG,

a.SpellHRGChapter,

a.SpellHRGSubChapter,

a.HRG\_Description,

a.EL\_Trim,

a.NE\_Trim,

a.Long\_Stay\_Payment,

a.EL\_Trim\_5DRule,

a.NE\_Trim\_5DRule,

a.Manual\_TP\_EL,

a.Manual\_TP\_NE,

a.DC\_EL\_SemiAuto,

a.NE\_SemiAuto,

a.Fin\_EL,

a.Fin\_NE,

a.Initial\_longStayPayment,

a.LSP\_after\_QR1,

a.LSP\_after\_CB,

a.LSP\_after\_CB\*(**1**+&InfEffTotalUplift18\_19.) /\*format=8.0\*/ as LSP\_Uplift18\_19

from TP\_and\_LSP\_Part8\_LSP\_after\_CB as a

order by a.Spell\_HRG

;

**quit**;

/\*Part9-"Modelled LSP prices adjusted to a 18/19 basis " -- End here \*/

/\* Part9\_N " FY1920 --- \*/

/\*Part 9\_N -"Long stay payments after SCF(Scaling Factor Adj) implementation"- Start from here \*/

**proc** **sql** noprint;

select SF into :SCF

from Tariff\_R.SF

where year="&FYear"

;

**quit**;

%let SCF=&SCF;

%put SCF=&SCF;

**proc** **sql** noprint;

create table TP\_LSP\_Part9\_N\_LSP\_after\_SCF

as

select

a.Spell\_HRG,

a.SpellHRGChapter,

a.SpellHRGSubChapter,

a.HRG\_Description,

a.EL\_Trim,

a.NE\_Trim,

a.Long\_Stay\_Payment,

a.EL\_Trim\_5DRule,

a.NE\_Trim\_5DRule,

a.Manual\_TP\_EL,

a.Manual\_TP\_NE,

a.DC\_EL\_SemiAuto,

a.NE\_SemiAuto,

a.Fin\_EL,

a.Fin\_NE,

a.Initial\_longStayPayment,

a.LSP\_after\_QR1,

a.LSP\_after\_CB,

a.LSP\_Uplift18\_19,

a.LSP\_Uplift18\_19\*(**1**+&SCF.) as LSP\_SCF

from TP\_and\_LSP\_Part9\_LSP\_adj\_1819 as a

order by a.Spell\_HRG

;

**quit**;

**data** Modelout.STEP\_BY\_STEP\_TRIMPOINT\_and\_LSP;

set TP\_LSP\_Part9\_N\_LSP\_after\_SCF;

**run**;

/\*Part 9\_N -"Long stay payments after SCF(Scaling Factor Adj) implementation"- End here \*/

/\*---------------------------------------------------------------------------------------------------\*/

/\*Part 10-"Long stay payments after QR2 implementation adjustment implementation " -- Start from here \*/

/\*import QR 2 table \*/

**PROC** **IMPORT** OUT= APC\_OPROC\_QR2

DATAFILE= "\\irnarch\sas\_data\PID\_Proj\_1718\APC\APC\_Excel\_Model\inputdata\APC\_OPROC\_QR2\_Factor.xlsx"

DBMS=xlsx REPLACE ;

**run**;

**proc** **sql** noprint;

create table TP\_and\_LSP\_Part10\_LSP\_after\_QR2

as

select

a.Spell\_HRG,

a.SpellHRGChapter,

a.SpellHRGSubChapter,

a.HRG\_Description,

a.EL\_Trim,

a.NE\_Trim,

a.Long\_Stay\_Payment,

a.EL\_Trim\_5DRule,

a.NE\_Trim\_5DRule,

a.Manual\_TP\_EL,

a.Manual\_TP\_NE,

a.DC\_EL\_SemiAuto,

a.NE\_SemiAuto,

a.Fin\_EL,

a.Fin\_NE,

a.Initial\_longStayPayment,

a.LSP\_after\_QR1,

a.LSP\_after\_CB,

a.LSP\_Uplift18\_19,

a.LSP\_Uplift18\_19\*(**1**+b.APC\_OPROC\_QR2) /\*format=8.0\*/ as LSP\_QR2

from TP\_and\_LSP\_Part9\_LSP\_adj\_1819 as a

left join APC\_OPROC\_QR2 as b

on a.Spell\_HRG=b.HRG

order by a.Spell\_HRG

;

**quit**;

/\*Part 10-"Long stay payments after QR2 implementation adjustment implementation " -- End here \*/

/\*Part 11-"Long stay payments after SMF adjustment implementation " -- Start from here \*/

**proc** **sql** noprint;

create table TP\_LSP\_Part10\_LSP\_include\_SMF

as

select

a.\*,

b.SMF

from

TP\_and\_LSP\_Part10\_LSP\_after\_QR2 as a

left join

Tariff\_R.APC\_OPROC\_SMOOTHING as b

on a.Spell\_HRG=b.HRG

and b.Year=&Year.

;

**quit**;

**proc** **sql** noprint;

Create table TP\_LSP\_Part11\_LSP\_after\_SMF

as

select

a.Spell\_HRG,

a.SpellHRGChapter,

a.SpellHRGSubChapter,

a.HRG\_Description,

a.EL\_Trim,

a.NE\_Trim,

a.Long\_Stay\_Payment,

a.EL\_Trim\_5DRule,

a.NE\_Trim\_5DRule,

a.Manual\_TP\_EL,

a.Manual\_TP\_NE,

a.DC\_EL\_SemiAuto,

a.NE\_SemiAuto,

a.Fin\_EL,

a.Fin\_NE,

a.Initial\_longStayPayment,

a.LSP\_after\_QR1,

a.LSP\_after\_CB,

a.LSP\_Uplift18\_19,

a.LSP\_QR2,

a.LSP\_QR2\*(**1**+a.SMF ) as LSP\_SMF

from TP\_LSP\_Part10\_LSP\_include\_SMF as a

order by a.Spell\_HRG

;

**quit**;

/\*Part 11-"Long stay payments after SMF adjustment implementation " -- End here \*/

/\*Part 12-"Long stay payments after QR3 implementation" --- Start from here \*/

**PROC** **IMPORT** OUT= APC\_OPROC\_QR3

DATAFILE= "\\irnarch\sas\_data\PID\_Proj\_1718\APC\APC\_Excel\_Model\inputdata\APC\_OPROC\_QR3.xlsx"

DBMS=xlsx REPLACE ;

**run**;

**proc** **sql** noprint;

create table TP\_LSP\_Part12\_LSP\_after\_QR3

as

select

a.Spell\_HRG,

a.SpellHRGChapter,

a.SpellHRGSubChapter,

a.HRG\_Description,

a.EL\_Trim,

a.NE\_Trim,

a.Long\_Stay\_Payment,

a.EL\_Trim\_5DRule,

a.NE\_Trim\_5DRule,

a.Manual\_TP\_EL,

a.Manual\_TP\_NE,

a.DC\_EL\_SemiAuto,

a.NE\_SemiAuto,

a.Fin\_EL,

a.Fin\_NE,

a.Initial\_longStayPayment,

a.LSP\_after\_QR1,

a.LSP\_after\_CB,

a.LSP\_Uplift18\_19,

a.LSP\_QR2,

a.LSP\_SMF,

a.LSP\_SMF\*(**1**+b.APC\_OPROC\_QR3 ) as LSP\_QR3

from

TP\_LSP\_Part11\_LSP\_after\_SMF as a

left join

APC\_OPROC\_QR3 as b

on a.Spell\_HRG=b.HRG

order by a.Spell\_HRG

;

**quit**;

/\*Part 12-"Long stay payments after QR3 implementation" --- End here \*/

/\*Part 13-"Long stay payments after SCF(Scaling Factor Adj) implementation"- Start from here \*/

**proc** **sql** noprint;

select SF into :SCF

from Tariff\_R.SF

where year="&FYear"

;

**quit**;

%let SCF=&SCF;

%put SCF=&SCF;

**proc** **sql** noprint;

create table TP\_LSP\_Part13\_LSP\_after\_SCF

as

select

a.Spell\_HRG,

a.SpellHRGChapter,

a.SpellHRGSubChapter,

a.HRG\_Description,

a.EL\_Trim,

a.NE\_Trim,

a.Long\_Stay\_Payment,

a.EL\_Trim\_5DRule,

a.NE\_Trim\_5DRule,

a.Manual\_TP\_EL,

a.Manual\_TP\_NE,

a.DC\_EL\_SemiAuto,

a.NE\_SemiAuto,

a.Fin\_EL,

a.Fin\_NE,

a.Initial\_longStayPayment,

a.LSP\_after\_QR1,

a.LSP\_after\_CB,

a.LSP\_Uplift18\_19,

a.LSP\_QR2,

a.LSP\_SMF,

a.LSP\_QR3,

a.LSP\_QR3\*(**1**+&SCF.) as LSP\_SCF /\*format=8.0\*/

from TP\_LSP\_Part12\_LSP\_after\_QR3 as a

order by a.Spell\_HRG

;

**quit**;

/\*Part 13-"Long stay payments after SCF(Scaling Factor Adj) implementation"- End here \*/

/\*Part 14 - "Long stay payments on a 17/18 basis (applying inflation, efficiency and CNST)" - Start from here \*/

**proc** **sql** noprint;

select Inflation\_and\_Efficiency\_Combine into :I\_E\_combine

from TARIFF\_R.I\_E

where Year="&FYear."

;

**quit**;

%let I\_E\_combine=&I\_E\_combine.;

%put I\_E\_combine=&I\_E\_combine;

**proc** **sql** noprint;

create table STEP\_BY\_STEP\_TRIMPOINT\_and\_LSP

as

select

a.Spell\_HRG,

a.SpellHRGChapter,

a.SpellHRGSubChapter,

a.HRG\_Description,

a.EL\_Trim,

a.NE\_Trim,

a.Long\_Stay\_Payment,

a.EL\_Trim\_5DRule,

a.NE\_Trim\_5DRule,

a.Manual\_TP\_EL,

a.Manual\_TP\_NE,

a.DC\_EL\_SemiAuto,

a.NE\_SemiAuto,

a.Fin\_EL,

a.Fin\_NE,

a.Initial\_longStayPayment,

a.LSP\_after\_QR1,

a.LSP\_after\_CB,

a.LSP\_Uplift18\_19,

a.LSP\_QR2,

a.LSP\_SMF,

a.LSP\_QR3,

a.LSP\_SCF ,

a.LSP\_SCF\*(**1**+&I\_E\_combine.) as LSP\_1920Basis

from TP\_LSP\_Part13\_LSP\_after\_SCF as a

order by a.Spell\_HRG

;

**quit**;

/\*Part 14 - "Long stay payments on a 17/18 basis (applying inflation, efficiency and CNST)" - End here \*/

/\*--------------------------------------------------------------\*/

\* Excel\_Step3\_3 STEP BY STEP OP-DC-EL-NE \*/

/\* Version: 1.03 \*/

/\* Author: PID \*/

/\* Date: September 2017 \*/

/\*--------------------------------------------------------------\*/

/\* Input: \*/

/\* (1)StageLib..APC\_Tariff \*/

/\* (2)Tariff\_R.I\_E \*/

/\* (3)Tariff\_R.CNST \*/

/\* (4)Tariff\_R.HRG\_ELIGIBILITY \*/

/\* (5)StageLib.OPROC01\_07 \*/

/\* (6)Tariff\_R.APC\_OPROC\_SMOOTHING \*/

/\* (7)Tariff\_R.OPROC\_PRICES\_4\_IA\_MODEL \*/

/\* \*/

/\* Output: \*/

/\* (1)Modelout.STEP\_BY\_STEP\_OP\_DC\_EL\_NE \*/

/\* \*/

/\* Description: \*/

/\* Create required data for Step4:STEP BY STEP OP-DC-EL-NE \*/

/\*--------------------------------------------------------------\*/

**proc** **sql**;

create table OPROC\_Eligible\_Tariff

as select

a.HRG,

case

when b.OPROC\_Tariff=**.** then **0**

else b.OPROC\_Tariff

End as Tariff

from NHSE\_Currency\_specification\_SAS as a

left join

ModelOut.OPROC\_PRICES\_4\_IA\_MODEL as b

on a.HRG =b.HRG

where a.Outpatient\_procedure\_tariff=**1**

order by a.HRG

;

**Quit**;

/\*Part 1 Modelled prices plus SQL output - Start here \*/

**proc** **sql** noprint;

create table TF1\_after\_P1\_SQL\_Output

as

select

distinct

a.HRG as SPELL\_HRG,

substr(a.HRG,**1**,**1**) as SpellHRGChapter,

substr(a.HRG,**1**,**2**) as SpellHRGSubchapter,

a.Description as HRG\_Name,

case

when a.Outpatient\_procedure\_tariff=**1** then a.HRG

else " "

end as OPROC\_currency\_applicable,

case

when (calculated OPROC\_currency\_applicable is not null) and (c.OPROC\_Tariff is null) then **0**

when (calculated OPROC\_currency\_applicable is not null) and (c.OPROC\_Tariff is not null ) then c.OPROC\_Tariff

else **.**

End as P1\_OPROC label="P1\_OPROC",

b.EL\_Tariff as P1\_DC label="P1\_DC" ,

b.EL\_Tariff as P1\_EL label="P1\_EL" ,

b.NE\_Tariff as P1\_NE label="P1\_NE" from

NHSE\_Currency\_specification\_SAS as a

left join

&StageLib.**.A**PC\_Tariff as b

on a.HRG=b.SPELL\_HRG

left join Modelout.OPROC\_PRICES\_4\_IA\_MODEL as c /\* OPROC Tariff \*/

on a.HRG=c.HRG

order by a.HRG

;

**quit**;

/\*Part 2 Modelled prices plus Reference costs quantum reconciliation factor (QR1) - Start here \*/

**proc** **sql** noprint;

create table TF2\_Includs\_QR1

as

select

SPELL\_HRG,

SpellHRGChapter,

SpellHRGSubchapter,

HRG\_Name,

OPROC\_currency\_applicable,

P1\_OPROC ,

P1\_DC,

P1\_EL ,

P1\_NE ,

&QR1\_Factor. as QR1

from

TF1\_after\_P1\_SQL\_Output as a

order by SPELL\_HRG

;

**quit**;

/\*Part 2 Modelled prices plus Reference costs quantum reconciliation factor (QR1) - End here \*/

/\*Part3 Modelled prices after QR1 factor -Start from here \*/

**proc** **sql** noprint;

create table TF3\_after\_QR1

as

select

SPELL\_HRG,

SpellHRGChapter,

SpellHRGSubchapter,

HRG\_Name,

OPROC\_currency\_applicable,

P1\_OPROC,

P1\_DC ,

P1\_EL ,

P1\_NE,

QR1 ,

coalesce(P1\_OPROC\*(**1**+QR1), **.**) as P3\_OPROC ,

coalesce(P1\_DC\*(**1**+QR1),**0**) as P3\_DC ,

coalesce(P1\_EL\*(**1**+QR1),**0**) as P3\_EL,

coalesce(P1\_NE\*(**1**+QR1),**0**) as P3\_NE

from

TF2\_Includs\_QR1

order by SPELL\_HRG

;

**quit**;

/\*Part3 Modelled prices after QR1 factor -End here \*/

/\*Part4 Modelled price plus Cost base adjustment (CB) -Start here \*/

**proc** **sql** noprint;

create table TF4\_includs\_CB

as

select

SPELL\_HRG,

SpellHRGChapter,

SpellHRGSubchapter,

HRG\_Name,

OPROC\_currency\_applicable,

P1\_OPROC ,

P1\_DC ,

P1\_EL ,

P1\_NE ,

QR1,

P3\_OPROC ,

P3\_DC ,

P3\_EL ,

P3\_NE ,

&Cost\_base\_adjustment. as Cost\_base\_adjustment

from

TF3\_after\_QR1

order by SPELL\_HRG

;

**quit**;

/\*Part4 Modelled price plus Cost base adjustment (CB) -End here \*/

/\*Part5 Modelled price after Cost base adjustment (CB) -Start from here \*/

**proc** **sql** noprint;

create table TF5\_after\_CB

as

select

SPELL\_HRG,

SpellHRGChapter,

SpellHRGSubchapter,

HRG\_Name,

OPROC\_currency\_applicable,

P1\_OPROC ,

P1\_DC ,

P1\_EL ,

P1\_NE,

QR1 ,

P3\_OPROC ,

P3\_DC ,

P3\_EL ,

P3\_NE ,

Cost\_base\_adjustment,

coalesce(P3\_OPROC\*(**1**+Cost\_base\_adjustment),**.**) as P5\_OPROC ,

coalesce(P3\_DC\*(**1**+Cost\_base\_adjustment),**0**) as P5\_DC ,

coalesce(P3\_EL\*(**1**+Cost\_base\_adjustment),**0**) as P5\_EL ,

coalesce(P3\_NE\*(**1**+Cost\_base\_adjustment),**0**) as P5\_NE

from

TF4\_includs\_CB

order by SPELL\_HRG

;

**quit**;

/\*Part5 Modelled price after Cost base adjustment (CB) -End here \*/

/\*Part 6 Prices after inflation, efficiency and CNST adjustments to a 16/17 financial year basis - Start from here \*/

/\* Transpose print adjustment table \*/

**PROC** **TRANSPOSE** DATA=Tariff\_R.I\_E

OUT=Trans\_I\_E (drop=Source )

PREFIX=Year

NAME=Source

LABEL=Adjustment\_Type

;

ID Year;

VAR Inflation Efficiency Inflation\_and\_Efficiency\_Combine;

**RUN**; **QUIT**;

/\*Calculate uplift total \*/

**data** PriceAdj\_InflaEffici;

set Trans\_I\_E;

label Adjustment\_Type =Adjustment\_Type;

TotalUplift18\_19=(Year17\_18+**1**)\*(Year18\_19+**1**)-**1**;

label TotalUplift18\_19="Total uplift up to 18/19 price levels";

**run**;

**proc** **sql** noprint;

select TotalUplift18\_19 into :InfEffTotalUplift18\_19

from PriceAdj\_InflaEffici

where Adjustment\_Type="Inflation and Efficiency Combined"

;

**quit**;

%let InfEffTotalUplift18\_19=&InfEffTotalUplift18\_19;

%put InfEffTotalUplift18\_19=&InfEffTotalUplift18\_19;

/\* CNST table \*/

**proc** **sql** noprint;

create table CNST\_Y14\_15

as select

HRGSubchapter,

CNST as Y2014\_15

from Tariff\_R.CNST

where year="14/15"

order by HRGSubchapter

;

**quit**;

**proc** **sql** noprint;

create table CNST\_Y15\_16

as select

HRGSubchapter,

CNST as Y2015\_16

from Tariff\_R.CNST

where year="15/16"

order by HRGSubchapter

;

**quit**;

**proc** **sql** noprint;

create table CNST\_Y16\_17

as select

HRGSubchapter,

CNST as Y2016\_17

from Tariff\_R.CNST

where year="16/17"

order by HRGSubchapter

;

**quit**;

**proc** **sql** noprint;

create table CNST\_Y17\_18

as select

HRGSubchapter,

CNST as Y2017\_18

from Tariff\_R.CNST

where year="17/18"

order by HRGSubchapter

;

**quit**;

**proc** **sql** noprint;

create table CNST\_Y18\_19

as select

HRGSubchapter,

CNST as Y2018\_19

from Tariff\_R.CNST

where year="18/19"

order by HRGSubchapter

;

**quit**;

**proc** **sql** noprint;

create table CNST\_Y19\_20

as select

HRGSubchapter,

CNST as Y2019\_20

from Tariff\_R.CNST

where year="19/20"

order by HRGSubchapter

;

**quit**;

**proc** **sql** noprint;

create table CNST\_AllYears

as

select

a.HRGSubchapter,

a.Y2014\_15 label="Y2014\_15",

b.Y2015\_16 label="Y2015\_16",

c.Y2016\_17 label="Y2016\_17",

d.Y2017\_18 label="Y2017\_18",

e.Y2018\_19 label="Y2018\_19",

(**1**+b.Y2015\_16)\*(**1**+c.Y2016\_17)-**1** as CNSTTotaluplift1617 label="Total uplift up to 16/17 price levels",

(**1**+d.Y2017\_18)\*(**1**+e.Y2018\_19)-**1** as CNSTTotaluplift1819 label="Total uplift up to 18/19 price levels"

from

CNST\_Y14\_15 as a

left join

CNST\_Y15\_16 as b

on a.HRGSubchapter=b.HRGSubchapter

left join

CNST\_Y16\_17 as c

on a.HRGSubchapter=c.HRGSubchapter

left join

CNST\_Y17\_18 as d

on a.HRGSubchapter=d.HRGSubchapter

left join

CNST\_Y18\_19 as e

on a.HRGSubchapter=e.HRGSubchapter

order by a.HRGSubchapter

;

**quit**;

/\*join CNST uplift factor \*/

**proc** **sql** noprint;

create table TF6\_P6\_01

as

select a.\*,

b.CNSTTotaluplift1819,

&InfEffTotalUplift18\_19 as InfEffTotalUplift18\_19

from TF5\_after\_CB as a

left join

CNST\_AllYears as b

on a.SpellHRGSubchapter=b.HRGSubchapter

order by

a.SPELL\_HRG,

a.SpellHRGSubchapter

;

**quit**;

/\*Apply Total\_adjustment of inflation, efficiency , and Total adjustment of CNST, get Adjusted Tariff \*/

**proc** **sql**;

create table TF6\_price\_afternfeffiCNST

as

select

Spell\_HRG,

SpellHRGChapter,

SpellHRGSubchapter,

HRG\_Name,

OPROC\_currency\_applicable,

P1\_OPROC,

P1\_DC,

P1\_EL,

P1\_NE,

QR1,

P3\_OPROC,

P3\_DC,

P3\_EL,

P3\_NE,

Cost\_base\_adjustment,

P5\_OPROC,

P5\_DC,

P5\_EL,

P5\_NE,

coalesce((**1**+InfEffTotalUplift18\_19)\*(**1**+CNSTTotaluplift1819)\*P5\_OPROC ,**.**) as P6\_OPROC,

coalesce((**1**+InfEffTotalUplift18\_19)\*(**1**+CNSTTotaluplift1819)\*P5\_DC,**0**) as P6\_DC,

coalesce((**1**+InfEffTotalUplift18\_19)\*(**1**+CNSTTotaluplift1819)\*P5\_EL,**0**) as P6\_EL ,

coalesce((**1**+InfEffTotalUplift18\_19)\*(**1**+CNSTTotaluplift1819)\*P5\_NE,**0**) as P6\_NE from TF6\_P6\_01

order by Spell\_HRG, SpellHRGChapter

;

**quit**;

/\*Part 6 Prices after inflation, efficiency and CNST adjustments to a 16/17 financial year basis - End here \*/

/\*Part 6\_N price after Scaling factor -- start from here \*/

**proc** **sql** noprint;

select SF into :SCF

from Tariff\_R.SF

where year="&FYear"

;

**quit**;

%let SCF=&SCF;

%put SCF=&SCF;

**proc** **sql**;

create table TF6N\_price\_afterScaling

as

select

Spell\_HRG,

SpellHRGChapter,

SpellHRGSubchapter,

HRG\_Name,

OPROC\_currency\_applicable,

P1\_OPROC,

P1\_DC,

P1\_EL,

P1\_NE,

QR1,

P3\_OPROC,

P3\_DC,

P3\_EL,

P3\_NE,

Cost\_base\_adjustment,

P5\_OPROC,

P5\_DC,

P5\_EL,

P5\_NE,

P6\_OPROC,

P6\_DC ,

P6\_EL ,

P6\_NE ,

&SCF as Scaling\_Factor,

P6\_OPROC\*(**1**+&SCF) as P6N\_SF\_OPROC,

P6\_DC\*(**1**+&SCF) as P6N\_SF\_DC,

P6\_EL\*(**1**+&SCF) as P6N\_SF\_EL,

P6\_NE\*(**1**+&SCF) as P6N\_SF\_NE

from TF6\_price\_afternfeffiCNST

;

**quit**;

/\*Part 7 Manual Adjustments --- Start from here \*/

/\*import Manual Adjustment data \*/

libname MA\_lib "\\irnarch\sas\_data\Tariff Rebuild\Model outputs pre\_MA\apc\_oproc";

**proc** **sql**;

create table TF7\_price\_after\_MA

as

select

a.Spell\_HRG,

a.SpellHRGChapter,

a.SpellHRGSubchapter,

a.HRG\_Name,

a.OPROC\_currency\_applicable,

a.P1\_OPROC,

a.P1\_DC,

a.P1\_EL,

a.P1\_NE,

a.QR1,

a.P3\_OPROC,

a.P3\_DC,

a.P3\_EL,

P3\_NE,

a.Cost\_base\_adjustment,

a.P5\_OPROC,

a.P5\_DC,

a.P5\_EL,

a.P5\_NE,

a.P6\_OPROC,

a.P6\_DC,

a.P6\_EL,

a.P6\_NE,

a.Scaling\_Factor,

a.P6N\_SF\_OPROC,

a.P6N\_SF\_DC,

a.P6N\_SF\_EL,

a.P6N\_SF\_NE,

/\* keep the value as before \*/

a.P6\_OPROC as P7\_OPROC label="P7\_OPROC",

a.P6\_DC as P7\_DC label="P7\_DC",

a.P6\_EL as P7\_EL label="P7\_EL" ,

a.P6\_NE as P7\_NE label="P7\_NE"

from TF6N\_price\_afterScaling as a

left join MA\_lib.POST\_MA\_APC\_N\_OPROC as b

on a.Spell\_HRG=b.HRG

order by a.Spell\_HRG

;

**quit**;

/\*Part 7 Manual Adjustments --- End \*/

/\* Part 8. Prices before QR2 implementation - Start from Here \*/

**proc** **sql**;

create table TF8\_price\_before\_Q2\_1

as

select

a.Spell\_HRG,

a.SpellHRGChapter,

a.SpellHRGSubchapter,

a.HRG\_Name,

a.OPROC\_currency\_applicable,

a.P1\_OPROC,

a.P1\_DC,

a.P1\_EL,

a.P1\_NE,

a.QR1,

a.P3\_OPROC,

a.P3\_DC,

a.P3\_EL,

a.P3\_NE,

a.Cost\_base\_adjustment,

a.P5\_OPROC,

a.P5\_DC,

a.P5\_EL,

a.P5\_NE,

a.P6\_OPROC,

a.P6\_DC,

a.P6\_EL,

a.P6\_NE,

a.Scaling\_Factor,

a.P6N\_SF\_OPROC,

a.P6N\_SF\_DC,

a.P6N\_SF\_EL,

a.P6N\_SF\_NE,

a.P7\_OPROC ,

a.P7\_DC ,

a.P7\_EL ,

a.P7\_NE ,

coalesce(a.P7\_OPROC,a.P6\_OPROC) as P8\_OPROC ,

coalesce(a.P7\_DC,a.P6\_DC) as P8\_DC ,

coalesce(a.P7\_EL,a.P6\_EL) as P8\_EL ,

coalesce(a.P7\_NE,a.P6\_NE) as P8\_NE

from TF7\_price\_after\_MA as a

order by Spell\_HRG

;

**quit**;

**proc** **sql**;

create table TF8\_price\_before\_Q2

as

select

a.Spell\_HRG,

a.SpellHRGChapter,

a.SpellHRGSubchapter,

a.HRG\_Name,

a.OPROC\_currency\_applicable,

case when P5\_OPROC=P8\_OPROC then **0**

else **1**

end as P0\_OPROC,

case when P5\_DC=P8\_DC then **0**

else **1**

end as P0\_DC,

case when P5\_EL=P8\_EL then **0**

else **1**

end as P0\_EL,

case when P5\_NE=P8\_NE then **0**

else **1**

end as P0\_NE,

a.P1\_OPROC,

a.P1\_DC,

a.P1\_EL,

a.P1\_NE,

a.QR1,

a.P3\_OPROC,

a.P3\_DC,

a.P3\_EL,

a.P3\_NE,

a.Cost\_base\_adjustment,

a.P5\_OPROC,

a.P5\_DC,

a.P5\_EL,

a.P5\_NE,

a.P6\_OPROC,

a.P6\_DC,

a.P6\_EL,

a.P6\_NE,

a.Scaling\_Factor,

a.P6N\_SF\_OPROC,

a.P6N\_SF\_DC,

a.P6N\_SF\_EL,

a.P6N\_SF\_NE,

a.\*

from TF8\_price\_before\_Q2\_1 as a

order by Spell\_HRG

;

**quit**;

/\* Part 8. Prices before QR2 implementation - End Here \*/

**data** Modelout.STEP\_BY\_STEP\_OP\_DC\_EL\_NE;

set TF8\_price\_before\_Q2;

**run**;

/\* Part9. Manual adjustments quantum reconciliation factor (QR2)- Start here \*/

/\*import QR 2 table \*/

**PROC** **IMPORT** OUT= APC\_OPROC\_QR2

DATAFILE= "\\irnarch\sas\_data\PID\_Proj\_1718\APC\APC\_Excel\_Model\inputdata\APC\_OPROC\_QR2\_Factor.xlsx"

DBMS=xlsx REPLACE ;

**run**;

**proc** **sql** noprint;

create table TF9\_MA\_includes\_QR2

as

select

a.\*,

b.APC\_OPROC\_QR2 as P9\_QR2 label="P9\_QR2 " format=Percentn8.2

from

TF8\_price\_before\_Q2 as a

left join

APC\_OPROC\_QR2 as b

on a.Spell\_HRG=b.HRG

order by a.Spell\_HRG

;

**quit**;

/\*Part10.Prices after QR2 implementation -Start from here \*/

**proc** **sql** noprint;

create table TF10\_MA\_Price\_after\_QR2

as

select

a.Spell\_HRG,

a.SpellHRGChapter,

a.SpellHRGSubchapter,

a.HRG\_Name,

a.OPROC\_currency\_applicable,

a.P1\_OPROC,

a.P1\_DC,

a.P1\_EL,

a.P1\_NE,

a.QR1,

a.P3\_OPROC,

a.P3\_DC,

a.P3\_EL,

a.P3\_NE,

a.Cost\_base\_adjustment,

a.P5\_OPROC,

a.P5\_DC,

a.P5\_EL,

a.P5\_NE,

a.P6\_OPROC,

a.P6\_DC,

a.P6\_EL,

a.P6\_NE,

a.P7\_OPROC ,

a.P7\_DC ,

a.P7\_EL ,

a.P7\_NE ,

a.P8\_OPROC ,

a.P8\_DC ,

a.P8\_EL ,

a.P8\_NE ,

a.P9\_QR2 /\* format=Percentn8.2\*/,

a.P8\_OPROC as P10\_OPROC ,

a.P8\_DC as P10\_DC ,

a.P8\_EL as P10\_EL ,

a.P8\_NE as P10\_NE

from

TF9\_MA\_includes\_QR2 as a

order by a.Spell\_HRG

;

**quit**;

/\*Part10.Prices after QR2 implementation -End here \*/

/\*Part11.Smoothing Factor adj (SMF) \* - Start from here\*/

**proc** **sql** noprint;

create table TF11\_includes\_SMF

as

select

a.\*,

b.SMF from

TF10\_MA\_Price\_after\_QR2 as a

left join

Tariff\_R.APC\_OPROC\_SMOOTHING(where=(Year=&Year.) )as b

on a.SPELL\_HRG=b.HRG

order by a.SPELL\_HRG

;

**quit**;

/\*Part11.Smoothing Factor adj (SMF) \* - End here\*/

/\*Part 12 Prices after SMF adjustment - Start from here \*/

**proc** **sql** noprint;

create table TF12\_Price\_after\_SMF

as

select

a.Spell\_HRG,

a.SpellHRGChapter,

a.SpellHRGSubchapter,

a.HRG\_Name,

a.OPROC\_currency\_applicable,

case when P5\_OPROC=P8\_OPROC then **0**

else **1**

end as P0\_OPROC,

case when P5\_DC=P8\_DC then **0**

else **1**

end as P0\_DC,

case when P5\_EL=P8\_EL then **0**

else **1**

end as P0\_EL,

case when P5\_NE=P8\_NE then **0**

else **1**

end as P0\_NE,

a.P1\_OPROC,

a.P1\_DC,

a.P1\_EL,

a.P1\_NE,

a.QR1,

a.P3\_OPROC,

a.P3\_DC,

a.P3\_EL,

a.P3\_NE,

a.Cost\_base\_adjustment,

a.P5\_OPROC,

a.P5\_DC,

a.P5\_EL,

a.P5\_NE,

a.P6\_OPROC,

a.P6\_DC,

a.P6\_EL,

a.P6\_NE,

a.P7\_OPROC ,

a.P7\_DC ,

a.P7\_EL ,

a.P7\_NE ,

a.P8\_OPROC,

a.P8\_DC,

a.P8\_EL,

a.P8\_NE,

a.P9\_QR2,

a.P10\_OPROC ,

a.P10\_DC ,

a.P10\_EL ,

a.P10\_NE ,

a.SMF,

coalesce(a.P10\_OPROC\*(**1**+a.SMF),**.**) as P12\_OPROC ,

coalesce(a.P10\_DC\*(**1**+a.SMF),**.**) as P12\_DC ,

coalesce(a.P10\_EL\*(**1**+a.SMF),**.**) as P12\_EL ,

coalesce(a.P10\_NE\*(**1**+a.SMF),**.**) as P12\_NE

from TF11\_includes\_SMF as a

order by a.Spell\_HRG

;

**quit**;

/\*Part 12 Prices after SMF adjustment - End here \*/

/\*Part 13 Final quantum reconciled prices (QR3) -Start from here \*/

**PROC** **IMPORT** OUT= APC\_OPROC\_QR3

DATAFILE= "\\irnarch\sas\_data\PID\_Proj\_1718\APC\APC\_Excel\_Model\inputdata\APC\_OPROC\_QR3.xlsx"

DBMS=xlsx REPLACE ;

**run**;

%let APC\_OPROC\_QR3=0;

**proc** **sql** noprint;

create table TF13\_price\_includes\_QR3

as

select

a.\*,

&APC\_OPROC\_QR3. as P13\_QR3 label="P13\_QR3"

from

TF12\_Price\_after\_SMF as a

;

**quit**;

/\*Part 13 Final quantum reconciled prices (QR3) -End here \*/

/\*Part 14 Prices after QR3 implementation - Start from here \*/

**proc** **sql** noprint;

create table TF14\_Price\_after\_QR3

as

select

a.Spell\_HRG,

a.SpellHRGChapter,

a.SpellHRGSubchapter,

a.HRG\_Name,

a.OPROC\_currency\_applicable,

a.P1\_OPROC,

a.P1\_DC,

a.P1\_EL,

a.P1\_NE,

a.QR1,

a.P3\_OPROC,

a.P3\_DC,

a.P3\_EL,

a.P3\_NE,

a.Cost\_base\_adjustment,

a.P5\_OPROC,

a.P5\_DC,

a.P5\_EL,

a.P5\_NE,

a.P6\_OPROC,

a.P6\_DC,

a.P6\_EL,

a.P6\_NE,

a.P7\_OPROC ,

a.P7\_DC ,

a.P7\_EL ,

a.P7\_NE ,

a.P8\_OPROC,

a.P8\_DC,

a.P8\_EL,

a.P8\_NE,

a.P9\_QR2,

a.P10\_OPROC ,

a.P10\_DC ,

a.P10\_EL ,

a.P10\_NE ,

a.SMF,

a.P12\_OPROC ,

a.P12\_DC ,

a.P12\_EL ,

a.P12\_NE ,

a.P13\_QR3,

coalesce(a.P12\_OPROC\*(**1**+a.P13\_QR3),**.**) as P14\_OPROC ,

coalesce(a.P12\_DC\*(**1**+a.P13\_QR3),**.**) as P14\_DC ,

coalesce(a.P12\_EL\*(**1**+a.P13\_QR3),**.**) as P14\_EL ,

coalesce(a.P12\_NE\*(**1**+a.P13\_QR3),**.**) as P14\_NE

from TF13\_price\_includes\_QR3 as a

order by a.Spell\_HRG

;

**quit**;

/\*Part 14 Prices after QR3 implementation - End here \*/

/\*Part 15 Scaling Factor Adj (SCF) -Start from here\*/

**proc** **sql** noprint;

select SF into :SCF

from Tariff\_R.SF

where year="&FYear"

;

**quit**;

%let SCF=&SCF;

%put SCF=&SCF;

**proc** **sql** noprint;

Create table TF15\_price\_includes\_SCF as

select a.\*,

&SCF as P15\_SCF label="P15\_SCF"

from TF14\_Price\_after\_QR3 as a

;

**quit**;

/\*Part 15 Scaling Factor Adj (SCF) -End here\*/

/\*Part 16 Final prices before 16/17 cost uplift assumptions -- Start from here \*/

**proc** **sql**;

create table TF16\_price\_before\_CostUplift

as

select

Spell\_HRG,

SpellHRGChapter,

SpellHRGSubchapter,

HRG\_Name,

OPROC\_currency\_applicable,

P1\_OPROC,

P1\_DC,

P1\_EL,

P1\_NE,

QR1,

P3\_OPROC,

P3\_DC,

P3\_EL,

P3\_NE,

Cost\_base\_adjustment,

P5\_OPROC,

P5\_DC,

P5\_EL,

P5\_NE,

P6\_OPROC,

P6\_DC,

P6\_EL,

P6\_NE,

P7\_OPROC,

P7\_DC,

P7\_EL,

P7\_NE,

P8\_OPROC,

P8\_DC,

P8\_EL,

P8\_NE,

P9\_QR2,

P10\_OPROC,

P10\_DC,

P10\_EL,

P10\_NE,

SMF,

P12\_OPROC,

P12\_DC,

P12\_EL,

P12\_NE,

P13\_QR3,

P14\_OPROC,

P14\_DC,

P14\_EL,

P14\_NE,

P15\_SCF,

coalesce(P14\_OPROC\*(**1**+P15\_SCF),**.**) as P16\_OPROC ,

coalesce(P14\_DC\*(**1**+P15\_SCF),**.**) as P16\_DC ,

coalesce(P14\_EL\*(**1**+P15\_SCF),**.**) as P16\_EL ,

coalesce(P14\_NE\*(**1**+P15\_SCF),**.**) as P16\_NE

from TF15\_price\_includes\_SCF

order by Spell\_HRG

;

**quit**;

/\*Part 16 Final prices before 16/17 cost uplift assumptions -- End here \*/

/\*Part 17 Year 17/18 Inflation start from here \*/

**proc** **sql** noprint;

select Inflation into :Inflation

from Tariff\_R.I\_E

where year="&FYear"

;

**quit**;

%let Inflation=&Inflation;

%put Inflation=&Inflation;

**proc** **sql** noprint;

Create table TF17\_price\_and\_Inflation as

select a.\*,

&Inflation as P17\_Inflation label="P17\_Inflation"

from TF16\_price\_before\_CostUplift as a

;

**quit**;

/\*Part 17 Year 17/18 Inflation End here \*/

/\*Part 18 Final Prices in 17/18 prices including inflation - Start from here\*/

**proc** **sql** noprint;

create table TF18\_price\_including\_Inflation

as

select

Spell\_HRG,

SpellHRGChapter,

SpellHRGSubchapter,

HRG\_Name,

OPROC\_currency\_applicable,

P1\_OPROC,

P1\_DC,

P1\_EL,

P1\_NE,

QR1,

P3\_OPROC,

P3\_DC,

P3\_EL,

P3\_NE,

Cost\_base\_adjustment,

P5\_OPROC,

P5\_DC,

P5\_EL,

P5\_NE,

P6\_OPROC,

P6\_DC,

P6\_EL,

P6\_NE,

P7\_OPROC,

P7\_DC,

P7\_EL,

P7\_NE,

P8\_OPROC,

P8\_DC,

P8\_EL,

P8\_NE,

P9\_QR2,

P10\_OPROC,

P10\_DC,

P10\_EL,

P10\_NE,

SMF,

P12\_OPROC,

P12\_DC,

P12\_EL,

P12\_NE,

P13\_QR3,

P14\_OPROC,

P14\_DC,

P14\_EL,

P14\_NE,

P15\_SCF,

P16\_OPROC ,

P16\_DC ,

P16\_EL ,

P16\_NE ,

P17\_Inflation,

coalesce(P16\_OPROC\*(**1**+P17\_Inflation),**.**) as P18\_OPROC ,

coalesce(P16\_DC\*(**1**+P17\_Inflation),**.**) as P18\_DC ,

coalesce(P16\_EL\*(**1**+P17\_Inflation),**.**) as P18\_EL ,

coalesce(P16\_NE\*(**1**+P17\_Inflation),**.**) as P18\_NE

from

TF17\_price\_and\_Inflation

order by Spell\_HRG

;

**quit**;

/\*Part 18 Final Prices in 19/20 prices including inflation - End here\*/

/\*Part 19 year 19/20 Efficiency Factor - Start from here \*/

**proc** **sql** noprint;

select Efficiency into :Efficiency

from Tariff\_R.I\_E

where year="&FYear"

;

**quit**;

%let Efficiency=&Efficiency;

%put Efficiency=&Efficiency;

**proc** **sql** noprint;

Create table TF19\_price\_and\_Efficiency as

select a.\*,

&Efficiency as P19\_Efficiency label="P19\_Efficiency" /\* format=Percentn8.2\*/

from TF18\_price\_including\_Inflation as a

;

**quit**;

/\*Part 19 year 19/20 Efficiency Factor - End here \*/

/\*Part 20 Final Prices in 19/20 prices, including inflation and efficiency assumptions -Start from here \*/

**proc** **sql** noprint;

create table TF20\_price\_inc\_InflationEffi

as

select

Spell\_HRG,

SpellHRGChapter,

SpellHRGSubchapter,

HRG\_Name,

OPROC\_currency\_applicable,

P1\_OPROC,

P1\_DC,

P1\_EL,

P1\_NE,

QR1,

P3\_OPROC,

P3\_DC,

P3\_EL,

P3\_NE,

Cost\_base\_adjustment,

P5\_OPROC,

P5\_DC,

P5\_EL,

P5\_NE,

P6\_OPROC,

P6\_DC,

P6\_EL,

P6\_NE,

P7\_OPROC,

P7\_DC,

P7\_EL,

P7\_NE,

P8\_OPROC,

P8\_DC,

P8\_EL,

P8\_NE,

P9\_QR2,

P10\_OPROC,

P10\_DC,

P10\_EL,

P10\_NE,

SMF,

P12\_OPROC,

P12\_DC,

P12\_EL,

P12\_NE,

P13\_QR3,

P14\_OPROC,

P14\_DC,

P14\_EL,

P14\_NE,

P15\_SCF,

P16\_OPROC ,

P16\_DC ,

P16\_EL ,

P16\_NE ,

P17\_Inflation,

P18\_OPROC ,

P18\_DC ,

P18\_EL ,

P18\_NE,

P19\_Efficiency,

coalesce(P18\_OPROC\*(**1**+P19\_Efficiency),**.**) as P20\_OPROC /\* format=comma8.0\*/,

coalesce(P18\_DC\*(**1**+P19\_Efficiency),**.**) as P20\_DC /\* format=comma8.0\*/,

coalesce(P18\_EL\*(**1**+P19\_Efficiency),**.**) as P20\_EL /\* format=comma8.0\*/,

coalesce(P18\_NE\*(**1**+P19\_Efficiency),**.**) as P20\_NE /\* format=comma8.0\*/

from

TF19\_price\_and\_Efficiency

order by Spell\_HRG

;

**quit**;

/\*Part 20 Final Prices in 19/20 prices, including inflation and efficiency assumptions -End here \*/

/\*Part 21 Year 19/20 CNST -Start from here \*/

**proc** **sql** noprint;

create table TF21\_price\_and\_CNST

as

select a.\*,

b.Y2019\_20 as P21\_19\_20\_CNST label="P21\_19\_20\_CNST" /\*format=Percentn8.2\*/

from TF20\_price\_inc\_InflationEffi as a

left join

CNST\_Y19\_20 as b

on a.SpellHRGSubchapter=b.HRGSubchapter

order by a.Spell\_HRG,a.SpellHRGSubchapter

;

**quit**;

/\*Part 21 Year 17/18 CNST -End here \*/

/\*Part22 FINAL PRICES in 17/18 prices, including inflation, efficiency and CNST assumptions- Start from here \*/

**proc** **sql** noprint;

create table STEP\_BY\_STEP\_OP\_DC\_EL\_NE

as

select

Spell\_HRG,

SpellHRGChapter,

SpellHRGSubchapter,

HRG\_Name,

OPROC\_currency\_applicable,

case when P5\_OPROC=P8\_OPROC then **0**

else **1**

end as P0\_OPROC,

case when P5\_DC=P8\_DC then **0**

else **1**

end as P0\_DC,

case when P5\_EL=P8\_EL then **0**

else **1**

end as P0\_EL,

case when P5\_NE=P8\_NE then **0**

else **1**

end as P0\_NE,

P1\_OPROC,

P1\_DC,

P1\_EL,

P1\_NE,

QR1 ,

P3\_OPROC,

P3\_DC,

P3\_EL,

P3\_NE,

Cost\_base\_adjustment,

P5\_OPROC,

P5\_DC,

P5\_EL,

P5\_NE,

P6\_OPROC,

P6\_DC,

P6\_EL,

P6\_NE,

P7\_OPROC,

P7\_DC,

P7\_EL,

P7\_NE,

P8\_OPROC,

P8\_DC,

P8\_EL,

P8\_NE,

P9\_QR2,

P10\_OPROC,

P10\_DC,

P10\_EL,

P10\_NE,

SMF,

P12\_OPROC,

P12\_DC,

P12\_EL,

P12\_NE,

P13\_QR3,

P14\_OPROC,

P14\_DC,

P14\_EL,

P14\_NE,

P15\_SCF,

P16\_OPROC ,

P16\_DC ,

P16\_EL ,

P16\_NE ,

P17\_Inflation,

P18\_OPROC ,

P18\_DC ,

P18\_EL ,

P18\_NE,

P19\_Efficiency,

P20\_OPROC ,

P20\_DC ,

P20\_EL ,

P20\_NE,

P21\_19\_20\_CNST,

coalesce(P20\_OPROC\*(**1**+P21\_19\_20\_CNST),**.**) as P22\_OPROC ,

coalesce(P20\_DC\*(**1**+P21\_19\_20\_CNST),**.**) as P22\_DC ,

coalesce(P20\_EL\*(**1**+P21\_19\_20\_CNST),**.**) as P22\_EL ,

coalesce(P20\_NE\*(**1**+P21\_19\_20\_CNST),**.**) as P22\_NE

from TF21\_price\_and\_CNST

order by Spell\_HRG

;

**quit**;

/\*Part22 FINAL PRICES in 17/18 prices, including inflation, efficiency and CNST assumptions- End here \*/

/\*--------------------------------------------------------------\*/

\* Excel\_Step3\_5 APC\_OPROC\_Model\_Linked\_Sheet \*/

/\* Version: 1.02 \*/

/\* Author: PID \*/

/\* Date: September 2017 \*/

/\*--------------------------------------------------------------\*/

/\* Input: \*/

/\* Total 4 Output datasets for Step1 to Step4. \*/

/\* \*/

/\* \*/

/\* \*/

/\* Output: \*/

/\* (1)ModelOut.APC\_OPROC\_Model\_LinkedSheet \*/

/\* \*/

/\* Description: \*/

/\* Create APC\_OPROC Model Linked Sheet \*/

/\* \*/

/\* \*/

/\*--------------------------------------------------------------\*/

/\*NOte: Need consider BPT Model \*/

/\* Linksheet before QR1\*/

**Proc** **sql**;

create table APC\_OPROC\_LinkSheet\_part1

as

select

a.year,

a.HRG as HRG\_Code label= "HRG\_Code",

a.Description as HRG\_Name label ="HRG\_Name",

b.P6\_OPROC as OPROC\_Tariff Label="Outpatient procedure tariff (£)" ,

case

when b.P6\_DC =b.P6\_EL then b.P6\_DC

else b.P6\_EL /\*should be BPT price here \*/

End as Combined\_DC\_EL\_Prices,

case

when b.P6\_DC =b.P6\_EL then **0**

else b.P6\_DC /\*should be BPT price here \*/

End as DC\_Prices,

case

when b.P6\_DC =b.P6\_EL then **0**

else b.P6\_EL /\*should be BPT price here \*/

End as EL\_Prices,

b.P6\_NE as NE\_Prices label="Non-elective spell tariff (£)"

from NHSE\_CURRENCY\_SPECIFICATION\_SAS as a

left join Modelout.STEP\_BY\_STEP\_OP\_DC\_EL\_NE as b

on a.HRG=b.SPELL\_HRG

order by HRG

;

**quit**;

**proc** **sql**;

create table APC\_OPROC\_LinkSheet\_part2

as

select a.\*,

b.EL\_Trim\_5DRule as EL\_TRIM\_POINTS label="Ordinary elective long stay trim point (days)",

b.NE\_Trim\_5DRule as NE\_TRIM\_POINTS label="Non-elective long stay trim point (days)",

b.LSP\_Uplift18\_19 as EBD\_Prices label="Per day long stay payment (for days exceeding trim point) (£)"

from APC\_OPROC\_LinkSheet\_part1 as a

left join Modelout.STEP\_BY\_STEP\_TRIMPOINT\_AND\_LSP as b

on a.HRG\_Code=b.SPELL\_HRG

order by a.HRG\_Code

;

**quit**;

**proc** **sql**;

create table ModelOut.APC\_OPROC\_Model\_LinkedSheet

as

select

a.Year,

a.HRG\_Code,

a.HRG\_Name,

a.OPROC\_Tariff,

a.Combined\_DC\_EL\_Prices,

a.DC\_Prices,

a.EL\_Prices,

a.EL\_TRIM\_POINTS,

a.NE\_Prices,

a.NE\_TRIM\_POINTS,

a.EBD\_Prices,

case

when b.P4\_SSEM\_Banding=**0** then "NO"

else "YES"

End as SSEM\_applicable label="Reduced short stay emergency tariff applicable?",

case

when calculated SSEM\_applicable ="YES" then b.P4\_SSEM\_Pct

Else **.**

end as SSEM\_Pct,

case

when calculated SSEM\_applicable ="YES" then a.NE\_Prices\*b.P4\_SSEM\_Pct

Else **.**

end as SSEM\_Price

from APC\_OPROC\_LinkSheet\_part2 as a

left join Modelout.quantum\_reconciliation\_bysubchap as b

on a.HRG\_Code=b.HRG

order by a.HRG\_Code

;

**quit**;

/\*--------------------------------------------------------------\*/

\* Excel\_Step3\_5 APC\_OPROC\_Model\_Linked\_Sheet \*/

/\* Version: 1.02 \*/

/\* Author: PID \*/

/\* Date: September 2017 \*/

/\*--------------------------------------------------------------\*/

/\* Input: \*/

/\* Total 4 Output datasets for Step1 to Step4. \*/

/\* \*/

/\* \*/

/\* \*/

/\* Output: \*/

/\* (1)ModelOut.APC\_OPROC\_Model\_LinkedSheet \*/

/\* \*/

/\* Description: \*/

/\* Create APC\_OPROC Model Linked Sheet \*/

/\* \*/

/\* \*/

/\*--------------------------------------------------------------\*/

/\*NOte: Need consider BPT Model \*/

**Proc** **sql**;

create table APC\_OPROC\_LinkSheet\_part1

as

select

a.year,

a.HRG as HRG\_Code label= "HRG\_Code",

a.Description as HRG\_Name label ="HRG\_Name",

b.P22\_OPROC as OPROC\_Tariff Label="Outpatient procedure tariff (£)" ,

case

when b.P22\_DC =b.P22\_EL then b.P22\_DC

else b.P22\_EL /\*should be BPT price here \*/

End as Combined\_DC\_EL\_Prices,

case

when b.P22\_DC =b.P22\_EL then **0**

else b.P22\_DC /\*should be BPT price here \*/

End as DC\_Prices,

case

when b.P22\_DC =b.P22\_EL then **0**

else b.P22\_EL /\*should be BPT price here \*/

End as EL\_Prices,

b.P22\_NE as NE\_Prices label="Non-elective spell tariff (£)"

from NHSE\_CURRENCY\_SPECIFICATION\_SAS as a

left join Modelout.STEP\_BY\_STEP\_OP\_DC\_EL\_NE as b

on a.HRG=b.SPELL\_HRG

order by HRG

;

**quit**;

**proc** **sql**;

create table APC\_OPROC\_LinkSheet\_part2

as

select a.\*,

b.Fin\_EL as EL\_TRIM\_POINTS label="Ordinary elective long stay trim point (days)",

b.Fin\_NE as NE\_TRIM\_POINTS label="Non-elective long stay trim point (days)",

b.LSP\_1920Basis as EBD\_Prices label="Per day long stay payment (for days exceeding trim point) (£)"

from APC\_OPROC\_LinkSheet\_part1 as a

left join Modelout.STEP\_BY\_STEP\_TRIMPOINT\_AND\_LSP as b

on a.HRG\_Code=b.SPELL\_HRG

order by a.HRG\_Code

;

**quit**;

**proc** **sql**;

create table ModelOut.APC\_OPROC\_Model\_LinkedSheet

as

select

a.Year,

a.HRG\_Code,

a.HRG\_Name,

a.OPROC\_Tariff,

a.Combined\_DC\_EL\_Prices,

a.DC\_Prices,

a.EL\_Prices,

a.EL\_TRIM\_POINTS,

a.NE\_Prices,

a.NE\_TRIM\_POINTS,

a.EBD\_Prices,

case

when b.P10\_SSEM\_Banding=**0** then "NO"

else "YES"

End as SSEM\_applicable label="Reduced short stay emergency tariff applicable?",

case

when calculated SSEM\_applicable ="YES" then b.P10\_SSEM\_Pct

Else **.**

end as SSEM\_Pct,

case

when calculated SSEM\_applicable ="YES" then a.NE\_Prices\*b.P10\_SSEM\_Pct

Else **.**

end as SSEM\_Price

from APC\_OPROC\_LinkSheet\_part2 as a

left join Modelout.quantum\_reconciliation\_bysubchap as b

on a.HRG\_Code=b.HRG

order by a.HRG\_Code

;

**quit**;

/\*--------------------------------------------------------------\*/

\* Excel\_Step3\_5 APC\_OPROC\_Model\_Linked\_Sheet \*/

/\* Version: 1.02 \*/

/\* Author: PID \*/

/\* Date: September 2017 \*/

/\*--------------------------------------------------------------\*/

/\* Input: \*/

/\* Total 4 Output datasets for Step1 to Step4. \*/

/\* \*/

/\* \*/

/\* \*/

/\* Output: \*/

/\* (1)ModelOut.APC\_OPROC\_Model\_LinkedSheet \*/

/\* \*/

/\* Description: \*/

/\* Create APC\_OPROC Model Linked Sheet \*/

/\* \*/

/\* \*/

/\*--------------------------------------------------------------\*/

/\*NOte: Need consider BPT Model \*/

/\* Linksheet before QR1\*/

**Proc** **sql**;

create table APC\_OPROC\_LinkSheet\_part1

as

select

a.year,

a.HRG as HRG\_Code label= "HRG\_Code",

a.Description as HRG\_Name label ="HRG\_Name",

b.P6N\_SF\_OPROC as OPROC\_Tariff Label="Outpatient procedure tariff (£)" ,

case

when b.P6N\_SF\_DC =b.P6N\_SF\_EL then b.P6N\_SF\_DC

else b.P6N\_SF\_EL /\*should be BPT price here \*/

End as Combined\_DC\_EL\_Prices,

case

when b.P6N\_SF\_DC =b.P6N\_SF\_EL then **0**

else b.P6N\_SF\_DC /\*should be BPT price here \*/

End as DC\_Prices,

case

when b.P6N\_SF\_DC =b.P6N\_SF\_EL then **0**

else b.P6N\_SF\_EL /\*should be BPT price here \*/

End as EL\_Prices,

b.P6N\_SF\_NE as NE\_Prices label="Non-elective spell tariff (£)"

from NHSE\_CURRENCY\_SPECIFICATION\_SAS as a

left join Modelout.STEP\_BY\_STEP\_OP\_DC\_EL\_NE as b

on a.HRG=b.SPELL\_HRG

order by HRG

;

**quit**;

**proc** **sql**;

create table APC\_OPROC\_LinkSheet\_part2

as

select a.\*,

b.EL\_Trim\_5DRule as EL\_TRIM\_POINTS label="Ordinary elective long stay trim point (days)",

b.NE\_Trim\_5DRule as NE\_TRIM\_POINTS label="Non-elective long stay trim point (days)",

b.LSP\_SCF as EBD\_Prices label="Per day long stay payment (for days exceeding trim point) (£)"

from APC\_OPROC\_LinkSheet\_part1 as a

left join Modelout.STEP\_BY\_STEP\_TRIMPOINT\_AND\_LSP as b

on a.HRG\_Code=b.SPELL\_HRG

order by a.HRG\_Code

;

**quit**;

**proc** **sql**;

create table ModelOut.APC\_OPROC\_Model\_LinkedSheet

as

select

a.Year,

a.HRG\_Code,

a.HRG\_Name,

a.OPROC\_Tariff,

a.Combined\_DC\_EL\_Prices,

a.DC\_Prices,

a.EL\_Prices,

a.EL\_TRIM\_POINTS,

a.NE\_Prices,

a.NE\_TRIM\_POINTS,

a.EBD\_Prices,

case

when b.P4\_SSEM\_Banding=**0** then "NO"

else "YES"

End as SSEM\_applicable label="Reduced short stay emergency tariff applicable?",

case

when calculated SSEM\_applicable ="YES" then b.P4\_SSEM\_Pct

Else **.**

end as SSEM\_Pct,

case

when calculated SSEM\_applicable ="YES" then a.NE\_Prices\*b.P4\_SSEM\_Pct

Else **.**

end as SSEM\_Price

from APC\_OPROC\_LinkSheet\_part2 as a

left join Modelout.quantum\_reconciliation\_bysubchap as b

on a.HRG\_Code=b.HRG

order by a.HRG\_Code

;

**quit**;

**Proc** **sql**;

create table APC\_OPROC\_4\_IA\_part1

as

select

a.year,

a.HRG as HRG\_Code label= "HRG\_Code",

a.Description as HRG\_Name label ="HRG\_Name",

b.P6\_OPROC as OPROC\_Tariff Label="Outpatient procedure tariff (£)" ,

case

when b.P6\_DC =b.P6\_EL then b.P6\_DC

else b.P6\_EL /\*should be BPT price here \*/

End as Combined\_DC\_EL\_Prices,

case

when b.P6\_DC =b.P6\_EL then **0**

else b.P6\_DC /\*should be BPT price here \*/

End as DC\_Prices,

case

when b.P6\_DC =b.P6\_EL then **0**

else b.P6\_EL /\*should be BPT price here \*/

End as EL\_Prices,

b.P6\_NE as NE\_Prices label="Non-elective spell tariff (£)"

from NHSE\_CURRENCY\_SPECIFICATION\_SAS as a

left join Modelout.STEP\_BY\_STEP\_OP\_DC\_EL\_NE as b

on a.HRG=b.SPELL\_HRG

order by HRG

;

**quit**;

**proc** **sql**;

create table APC\_OPROC\_4\_IA\_part2

as

select a.\*,

b.EL\_Trim\_5DRule as EL\_TRIM\_POINTS label="Ordinary elective long stay trim point (days)",

b.NE\_Trim\_5DRule as NE\_TRIM\_POINTS label="Non-elective long stay trim point (days)",

b.LSP\_Uplift18\_19 as EBD\_Prices label="Per day long stay payment (for days exceeding trim point) (£)"

from APC\_OPROC\_4\_IA\_part1 as a

left join Modelout.STEP\_BY\_STEP\_TRIMPOINT\_AND\_LSP as b

on a.HRG\_Code=b.SPELL\_HRG

order by a.HRG\_Code

;

**quit**;

**proc** **sql**;

create table ModelOut.APC\_OPROC\_4\_IA\_MODEL

as

select

a.Year,

a.HRG\_Code,

a.HRG\_Name,

a.OPROC\_Tariff,

a.Combined\_DC\_EL\_Prices,

a.DC\_Prices,

a.EL\_Prices,

a.EL\_TRIM\_POINTS,

a.NE\_Prices,

a.NE\_TRIM\_POINTS,

a.EBD\_Prices,

case

when b.P4\_SSEM\_Banding=**0** then "NO"

else "YES"

End as SSEM\_applicable label="Reduced short stay emergency tariff applicable?",

case

when calculated SSEM\_applicable ="YES" then b.P4\_SSEM\_Pct

Else **.**

end as SSEM\_Pct,

case

when calculated SSEM\_applicable ="YES" then a.NE\_Prices\*b.P4\_SSEM\_Pct

Else **.**

end as SSEM\_Price

from APC\_OPROC\_4\_IA\_part2 as a

left join Modelout.quantum\_reconciliation\_bysubchap as b

on a.HRG\_Code=b.HRG

order by a.HRG\_Code

;

**quit**;

/\*--------------------------------------------------------------\*/

\* Step3\_5 Create all APC\_OPROC Excel Model worksheets\_1

/\* Version: 1.02 \*/

/\* Author: PID \*/

/\* Date: September 2017 \*/

/\*--------------------------------------------------------------\*/

/\* Description: \*/

/\* Generate an excel file which contains 5 tabs for APC results:\*/

/\* 1.STEP BY STEP OP-DC-EL-NE; \*/

/\* 2.Long Stay Payment; \*/

/\* 3.STEP BY STEP TRIMPOINT and LSP; \*/

/\* 4.QUANTUM RECONCILIATION BY Sub Chapter \*/

/\* 5.APC OPROC Model LinkSheet \*/

/\*--------------------------------------------------------------\*/

/\*ODS to create Excel workbook output with multiple sheets\*/

ods \_all\_ close;

ods tagsets.ExcelXP path="\\irnarch\sas\_data\Tariff Rebuild\Models for demo\WIP\TC\_FY1920\Excel\_outputs" file="APC\_OPROC model.xml" style=Printer;

/\*Step 1-- Long Stay Payment Start from here \*/

ods tagsets.ExcelXP options(sheet\_name='Long Stay Payment'

autofilter='NO' embedded\_titles ="yes" autofit\_height ="yes" frozen\_headers="yes" frozen\_headers="2"

absolute\_column\_width='6,50,6,8,8,8,3,10,10,10,10,10,10');

**proc** **print** data = Modelout.Long\_Stay\_Payment noobs label split= '\*';

var HRG HRG\_Name Chapter Proportion\_Children\_Activity Congenital\_HRG\_Flag AG18\_YearsandUnder\_HRG\_Flag LU /style(header)=[background=#ecffb3];

var CLEAN\_EBD\_ACT / style(header)=[background=#ecffb3] style(Column)=[ tagattr="format:###,###,###"] ;

var CLEAN\_EBD\_TC LSP\_UC Limit\_within\_100\_to\_500 / style(header)=[background=#ecffb3] style(Column)=[ tagattr="format:###,###,###.##"] ;

var Zero\_price\_HRG / style(header)=[background=#ecffb3] ;

var Long\_Stay\_Payment / style(header)=[background=#ecffb3] style(Column)=[ tagattr="format:###,###,###.##"];

;

label

HRG="HRG \_\_"

HRG\_Name="HRG Name\_\_"

Chapter="Chapter \_\_"

Proportion\_Children\_Activity="Proportion\*of children\*activity\_\_"

Congenital\_HRG\_Flag="Congenital \*HRG flag\_\_"

AG18\_YearsandUnder\_HRG\_Flag="18 years \*and under \*HRG flag\_\_"

LU="LU\_\_"

CLEAN\_EBD\_ACT="CLEAN EBD \*ACT \_\_"

CLEAN\_EBD\_TC="CLEAN EBD \*TC \_\_"

LSP\_UC="LSP UC\_\_"

Limit\_within\_100\_to\_500="Limiting within \*£100 to £500 \_\_"

Zero\_price\_HRG="Zero priced HRG \*flag (1 if the HRG is zero-priced, 0 \*otherwise)\_\_"

Long\_Stay\_Payment="Long Stay Payments\_\_"

;

**run**;

/\*Step 1-- Long Stay Payment End here \*/

/\*Step 2-- STEP BY STEP TRIMPOINT and LSP------ Start from here \*/

/\*Create and format STEP BY STEP TRIMPOINT and LSP tab\*/

ods tagsets.ExcelXP options(sheet\_name='STEP BY STEP TRIMPOINT and LSP'

suppress\_bylines='yes'

frozen\_headers='1'

FROZEN\_ROWHEADERS='4'

autofit\_height='Yes'

absolute\_column\_width='6, 6, 7, 50, 8, 8, 8, 7, 8, 7, 7, 10, 10, 10,10, 10, 10, 10, 10, 10, 10,10,10,10'

pages\_fitwidth='1'

pages\_fitheight='9'

FitToPage="yes"

autofilter='1-66');

/\*Define STEP BY STEP TRIMPOINT and LSP tab output variables\*/

**proc** **print** data=Modelout.STEP\_BY\_STEP\_TRIMPOINT\_and\_LSP split='\*' n noobs;

var Spell\_HRG SpellHRGChapter SpellHRGSubChapter HRG\_Description / style(Column)=[background=#e6ffff tagattr='format:#,##'] style(header)=[background=#e6ffff];

/\*Part1\*/

var EL\_Trim NE\_Trim Long\_Stay\_Payment / style(Column)=[background=#ffb699] style(header)=[background=#ffb699];

/\*Part2 \*/

var EL\_Trim\_5DRule NE\_Trim\_5DRule / style(Column)=[background=#ffb6c1 ] style(header)=[background=#ffb6c1];

/\*Part 6 \*/

var Initial\_longStayPayment / style(Column)=[background=#e6ffff ] style(header)=[background=#e6ffff];

/\*Part 7 \*/

var LSP\_after\_QR1 / style(Column)=[background=#fff5e6 ] style(header)=[background=#fff5e6];

/\*Part 8 \*/

var LSP\_after\_CB / style(Column)=[background=#88cc00 ] style(header)=[background=#88cc00];

/\*Part 9 \*/

var LSP\_Uplift18\_19 / style(Column)=[background=#efdcc3 ] style(header)=[background=#efdcc3];

label

Spell\_HRG="Spell\*HRG\_\_"

SpellHRGChapter="Spell\*HRG\*chapter\_\_"

SpellHRGSubChapter="Spell HRG\*subchapter\_\_"

HRG\_Description="HRG description"

EL\_Trim="P1-SQL output\*EL\_Trim\_\_"

NE\_Trim="P1-SQL output\*NE\_Trim\_\_"

Long\_Stay\_Payment="P1-SQL output\*Long\_Stay\_Payment\_\_"

EL\_Trim\_5DRule="P2-Minimum 5 day rule\*EL\_Trim\_\_"

NE\_Trim\_5DRule="P2-Minimum 5 day rule\*NE\_Trim\_\_"

Manual\_TP\_EL="P3-2019/20 Manual\*EL\_Trim\_\_"

Manual\_TP\_NE="P3-2019/20 Manual\*NE\_Trim\_\_"

DC\_EL\_SemiAuto="P4-2019/20\*Semi-Automated\*EL\_Trim\_\_"

NE\_SemiAuto="P4-2019/20\*Semi-Automated\*NE\_Trim\_\_"

Fin\_EL="P5-Final trim points\*EL\_Trim\_\_"

Fin\_NE="P5-Final trim points\*NE\_Trim\_\_"

Initial\_longStayPayment="P6-Initial long stay\*payments\*(LSP)\_\_"

LSP\_after\_QR1="P7-LSP after QR1\*implementation\_\_"

LSP\_after\_CB="P8-Long stay \*payments after CB adjustment\*implementation\_\_"

LSP\_Uplift18\_19="P9-Modelled LSP\*prices adjusted to \*a 18/19 basis\_\_"

LSP\_QR2="P10-Long stay payments\*after QR2 \*implementation \*adjustment \*implementation\_\_"

LSP\_SMF="P11-Long stay payments\* after SMF\*adjustment \*implementation\_\_"

LSP\_QR3="P12-Long stay\*payments after\* QR3\*implementation\_\_"

LSP\_SCF="P13-Long stay\* payments after\* SCF\* implementation\_\_"

LSP\_1920Basis="P14-Long stay\*payments on a 19/20 \*basis (applying inflation, \*efficiency \*and CNST)\_\_"

;

**run**;

/\*Step 2-- STEP BY STEP TRIMPOINT and LSP----------------- End here \*/

/\*Step 3-- QUANTUM RECONCILIATION BY Sub Chapter-------------------------------- Start from here \*/

/\*Create and format QUANTUM RECONCILIATION BY Sub Chapter tab\*/

ods tagsets.ExcelXP options(sheet\_name='QUANTUM RECONCILIATION BY SubChapter'

suppress\_bylines='yes'

frozen\_headers='1'

FROZEN\_ROWHEADERS='3'

autofit\_height='Yes'

absolute\_column\_width='6, 50, 7, 7, 8, 8, 8, 7, 8, 7, 7, 10, 10, 10,10, 10, 10, 10, 10, 10, 10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10,'

pages\_fitwidth='1'

pages\_fitheight='9'

FitToPage="yes"

autofilter='1-170');

/\*Define STEP BY STEP QUANTUM RECONCILIATION BY Sub Chapter variables\*/

**proc** **print** data=Modelout.QUANTUM\_RECONCILIATION\_BySubChap split='\*' n noobs;

var HRG Description OPROC\_currency\_applicable / style(Column)=[background=#e6ffff tagattr='format:#,##'] style(header)=[background=#e6ffff];

var P0\_OPROC P0\_DC P0\_EL P0\_NE / style(header)=[background=#eda891];

var P1\_OPROC\_TA P1\_DC\_TA P1\_EL\_TA P1\_NE\_TA P1\_NE\_AT\_less\_SSEM P1\_LSP\_days P1\_SSEM / style(header)=[background=#eae395] style(Column)=[ tagattr="format:###,###,###"];

/\* Part 2 \*/

var P2\_OPROC P2\_DC P2\_EL P2\_NE P2\_LSP / style(header)=[background=#f0e6ff] style(Column)=[ tagattr="format:###,###,###.##"];

var P2\_SSEM\_Banding P2\_SSEM\_Pct/style(header)=[background=#f0e6ff];

var P2\_SSEM\_Tariff / style(header)=[background=#f0e6ff] style(Column)=[ tagattr="format:###,###,###.##"];

/\* Part 3 \*/

var P3\_Sub\_Chapter /style(header)=[background=#f0e6ff];

var P3\_OPROC\_quantum\_by\_HRG P3\_APC\_Quantum\_by\_HRG /style(header)=[background=#f0e6ff] style(Column)=[ tagattr="format:###,###,###.##"];

var P3\_Sub\_\_Chapter /style(header)=[background=#f0e6ff] ;

var P3\_OPROC\_quantum\_by\_SubChapter P3\_APC\_quantum\_by\_SubChapter P3\_APC\_OPROC\_quantum\_by\_chapter /style(header)=[background=#f0e6ff] style(Column)=[ tagattr="format:###,###,###.##"];

/\* Part 4 \*/

var P4\_OPROC P4\_DC P4\_EL P4\_NE P4\_Long\_Stay\_Payment /style(header)=[background=#ffe6e9] style(Column)=[ tagattr="format:###,###,###.##"];

var P4\_SSEM\_Banding P4\_SSEM\_Pct /style(header)=[background=#ffe6e9];

var P4\_SSEM\_Tariff /style(header)=[background=#ffe6e9] style(Column)=[ tagattr="format:###,###,###.##"];

/\*Part 6 \*/

var P6\_Sub\_Chapter /style(header)=[background=#ffe6e9];

var P6\_OPROC\_Quantum\_by\_HRG P6\_APC\_Quantum\_by\_HRG P6\_Sub\_\_Chapter P6\_OPROC\_quantum\_by\_SubChapter

P6\_APC\_quantum\_by\_SubChapter P6\_APC\_OPROC\_quantum\_by\_chapter /style(header)=[background=#ffe6e9] style(Column)=[ tagattr="format:###,###,###.##"];

label

Description="HRG description"

P0\_OPROC="P0\*HRGs eligible for QR2\*1 = eligible, \*0 = not eligible\* OPROC\_"

P0\_DC = "P0\*HRGs eligible for QR2\*1 = eligible, \*0 = not eligible\* DC \_\_"

P0\_EL = "P0\*HRGs eligible for QR2\*1 = eligible, \*0 = not eligible\* EL \_\_"

P0\_NE= "P0\*HRGs eligible for QR2\*1 = eligible, \*0 = not eligible\* NE \_\_"

P1\_OPROC\_TA="P1\*Total activity (spells)\*OPROC\_"

P1\_DC\_TA="P1\*Total activity (spells)\*DC \_\_"

P1\_EL\_TA="P1\*Total activity (spells)\*EL \_\_"

P1\_NE\_TA="P1\*Total activity (spells)\*NE \_\_"

P1\_NE\_AT\_less\_SSEM="P1\*NE activities less SSEM\* (this only applies for HRGs that \*are eligible for\* SSEM\_)"

P1\_LSP\_days="P1\*LSP days \_\_"

P1\_SSEM="P1\*SSEM activities\_\_"

P2\_OPROC="P2\*Modelled prices pre \*quantum reconciliation (QR1)\*OPROC \_"

P2\_DC="P2\*Modelled prices pre \*quantum reconciliation (QR1)\*DC \_"

P2\_EL="P2\*Modelled prices pre \*quantum reconciliation (QR1)\*EL \_"

P2\_NE="P2\*Modelled prices pre \*quantum reconciliation (QR1)\*NE \_"

P2\_LSP="P2\*Modelled prices pre \*quantum reconciliation (QR1)\*LSP \_"

P2\_SSEM\_Banding="P2\*Modelled prices pre \*quantum reconciliation (QR1)\*SSEM\_Banding\_"

P2\_SSEM\_Pct="P2\*Modelled prices pre \*quantum reconciliation (QR1)\*SSEM % \_\_"

P2\_SSEM\_Tariff="P2\*Modelled prices pre \*quantum reconciliation (QR1)\*SSEM Tariff \_\_"

P3\_Sub\_Chapter="P3\*Total quantum by \*subchapter pre quantum \*reconciliation (QR1)\*Sub Chapter \_"

P3\_OPROC\_quantum\_by\_HRG="P3\*Total quantum by \*subchapter pre quantum \*reconciliation (QR1)\*OPROC quantum by HRG \_"

P3\_APC\_Quantum\_by\_HRG="P3\*Total quantum by \*subchapter pre quantum \*reconciliation (QR1)\*APC quantum by HRG \_"

P3\_Sub\_\_Chapter="P3\*Total quantum by \*subchapter pre quantum \*reconciliation (QR1)\*Sub-Chapter \_"

P3\_OPROC\_quantum\_by\_SubChapter="P3\*Total quantum by \*subchapter pre quantum \*reconciliation (QR1)\* OPROC quantum by chapter\_"

P3\_APC\_quantum\_by\_SubChapter="P3\*Total quantum by \*subchapter pre quantum \*reconciliation (QR1)\*APC quantum by chapter\_"

P3\_APC\_OPROC\_quantum\_by\_chapter="P3\*Total quantum by \*subchapter pre quantum \*reconciliation (QR1)\*APC and OPROC \*quantum by chapter\_"

P4\_OPROC="P4\*Prices post inflation,\*efficiency and CNST adjustments\*(prices before manual\* adjustments)\*OPROC \_"

P4\_DC="P4\*Prices post inflation,\*efficiency and CNST adjustments\*(prices before manual\* adjustments)\*DC \_"

P4\_EL="P4\*Prices post inflation,\*efficiency and CNST adjustments\*(prices before manual\* adjustments)\*EL \_"

P4\_NE="P4\*Prices post inflation,\*efficiency and CNST adjustments\*(prices before manual\* adjustments)\*NE \_"

P4\_Long\_Stay\_Payment="P4\*Prices post inflation,\*efficiency and CNST adjustments\*(prices before manual\* adjustments)\*Long Stay Payment \_"

P4\_SSEM\_Banding="P4\*Prices post inflation,\*efficiency and CNST adjustments\*(prices before manual\* adjustments)\*SSEM Banding \_"

P4\_SSEM\_Pct="P4\*Prices post inflation,\*efficiency and CNST adjustments\*(prices before manual\* adjustments)\*SSEM % \_"

P4\_SSEM\_Tariff="P4\*Prices post inflation,\*efficiency and CNST adjustments\*(prices before manual\* adjustments)\*SSEM Tariff \_\_"

P5\_OPROC="P5\*Prices after \*manual adjustments\* adjustments)\*OPROC \_"

P5\_DC="P5\*Prices after \*manual adjustments\* adjustments)\*DC \_"

P5\_EL="P5\*Prices after \*manual adjustments\* adjustments)\*EL \_"

P5\_NE="P5\*Prices after \*manual adjustments\* adjustments)\*NE \_"

P5\_Long\_Stay\_Payment="P5\*Prices after \*manual adjustments\* adjustments)\*Long Stay Payment \_"

P5\_SSEM\_Banding="P5\*Prices after \*manual adjustments\* adjustments)\*SSEM Banding \_"

P5\_SSEM\_Pct="P5\*Prices after \*manual adjustments\* adjustments)\*SSEM % \_"

P5\_SSEM\_Tariff="P5\*Prices after \*manual adjustments\* adjustments)\*SSEM Tariff\_\_"

P6\_Sub\_Chapter="P6\*Total quantum by subchapter\*before manual adjustment\*Sub\_Chapter"

P6\_OPROC\_Quantum\_by\_HRG="P6\*Total quantum by subchapter\*before manual adjustment\*OPROC Quantum by HRG\_\_"

P6\_APC\_Quantum\_by\_HRG="P6\*Total quantum by subchapter\*before manual adjustment\*APC Quantum by HRG\_\_"

P6\_Sub\_\_Chapter="P6\*Total quantum by subchapter\*before manual adjustment\*Subchapter\_\_"

P6\_OPROC\_quantum\_by\_SubChapter="P6\*Total quantum by subchapter\*before manual adjustment\*OPROC quantum\*by chapter\_\_"

P6\_APC\_quantum\_by\_SubChapter="P6\*Total quantum by subchapter\*before manual adjustment\*APC quantum by chapter\_\_"

P6\_APC\_OPROC\_quantum\_by\_chapter="P6\*Total quantum by subchapter\*before manual adjustment\*APC and OPROC\*quantum by chapter\_\_"

P7\_Sub\_Chapter="P7\*Total quantum by\*subchapter after\*manual adjustment\*Sub-Chapter\_\_"

P7\_OPROC\_Quantum\_by\_HRG="P7\*Total quantum by\*subchapter after\*manual adjustment\*OPROC Quantum by HRG\_\_"

P7\_APC\_Quantum\_by\_HRG="P7\*Total quantum by\*subchapter after\*manual adjustment\*APC Quantum by HRG\_\_"

P7\_Sub\_\_Chapter="P7\*Total quantum by\*subchapter after\*manual adjustment\*Sub\_Chapter\_\_"

P7\_OPROC\_quantum\_by\_SubChapter="P7\*Total quantum by\*subchapter after\*manual adjustment\*OPROC quantum\* by chapter\_\_"

P7\_APC\_quantum\_by\_SubChapter="P7\*Total quantum by\*subchapter after\*manual adjustment\*APC quantum\* by chapter\_\_"

P7\_APC\_OPROC\_quantum\_by\_chapter="P7\*Total quantum by\*subchapter after\*manual adjustment\*APC and OPROC\*quantum by chapter\_\_"

P8\_Sub\_Chapter="P8\*Total quantum by\*subchapter after manual\*adjustment excluding HRGs\*that incorporate\*OPATT prices\*Sub-Chapter\_\_"

P8\_OPROC\_Quantum\_by\_HRG="P8\*Total quantum by\*subchapter after manual\*adjustment excluding HRGs\*that incorporate\*OPROC Quantum by HRG\_\_"

P8\_APC\_Quantum\_by\_HRG="P8\*Total quantum by\*subchapter after manual\*adjustment excluding HRGs\*that incorporate\*APC Quantum by HRG\_\_"

P8\_Sub\_\_Chapter="P8\*Total quantum by\*subchapter after manual\*adjustment excluding HRGs\*that incorporate\*OPATT prices\*Sub\_chapter\_\_"

P8\_OPROC\_quantum\_by\_SubChapter="P8\*Total quantum by\*subchapter after manual\*adjustment excluding HRGs\*that incorporate\*OPATT prices\*OPROC quantum\*by chapter\_\_"

P8\_APC\_quantum\_by\_SubChapter="P8\*Total quantum by\*subchapter after manual\*adjustment excluding HRGs\*that incorporate\*OPATT prices\*APC quantum by chapter\_\_"

P8\_APC\_OPROC\_quantum\_by\_chapter="P8\*Total quantum by\*subchapter after manual\*adjustment excluding HRGs\*that incorporate\*OPATT prices\*APC and OPROC\*quantum by chapter\_\_"

P9\_Chapter="P9\*Quantum reconciliation\*by chapter (QR2)\*Chapter \_"

P9\_Chapter\_Sum="P9\*Quantum reconciliation\*by chapter (QR2)\*Chapter sum\_\_"

P9\_APC\_OPROC\_QR2="P9\*Quantum reconciliation\*by chapter (QR2)\*APC and OPROC\*QR2 \_\_"

P10\_HRG="P10\*Prices after quantum\*reconciliation by\*chapter (QR2)\*HRG \_\_"

P10\_Chapter="P10\*Prices after quantum\*reconciliation by\*chapter (QR2)\*Chapter \_\_"

P10\_APCOPROC\_subchapter\_QR2="P10\*Prices after quantum\*reconciliation by\*chapter (QR2)\*APC and OPROC\*subchapter QR2 factor\_\_"

P10\_OPROC="P10\*Prices after quantum\*reconciliation by\*chapter (QR2)\*OPROC \_\_"

P10\_DC="P10\*Prices after quantum\*reconciliation by\*chapter (QR2)\*DC \_\_"

P10\_EL="P10\*Prices after quantum\*reconciliation by\*chapter (QR2)\*EL \_\_"

P10\_NE="P10\*Prices after quantum\*reconciliation by\*chapter (QR2)\*NE \_\_"

P10\_Long\_Stay\_Payment="P10\*Prices after quantum\*reconciliation by\*chapter (QR2)\*Long Stay Payment\_\_"

P10\_SSEM\_Banding="P10\*Prices after quantum\*reconciliation by\*chapter (QR2)\*SSEM Banding\_\_"

P10\_SSEM\_Pct="P10\*Prices after quantum\*reconciliation by\*chapter (QR2)\*SSEM % \_"

P10\_SSEM\_Tariff="P10\*Prices after quantum\*reconciliation by\*chapter (QR2)\*SSEM Tariff\_\_"

P11\_Sub\_Chapter="P11\*Total quantum by\*subchapter after quantum\*adjustment by \*chapter (QR2)-Subchapter\_\_"

P11\_OPROC\_Quantum\_by\_HRG="P11\*Total quantum by\*subchapter after quantum\*adjustment by \*chapter (QR2)-OPROC Quantum\*by HRG\_\_"

P11\_APC\_Quantum\_by\_HRG="P11\*Total quantum by\*subchapter after quantum\*adjustment by \*chapter (QR2)-APC Quantum\*by HRG\_\_"

P11\_APC\_OPROC\_Quantum\_by\_HRG="P11\*Total quantum by\*subchapter after quantum\*adjustment by \*chapter (QR2)-APC and OPROC\*Quantum by HRG\_\_"

P11\_Sub\_\_Chapter="P11\*Total quantum by\*subchapter after quantum\*adjustment by \*chapter (QR2)-Sub\_chapter\_\_"

P11\_OPROC\_quantum\_by\_SubChapter="P11\*Total quantum by\*subchapter after quantum\*adjustment by \*chapter (QR2)-OPROC Quantum\*by subchapter\_\_"

P11\_APC\_quantum\_by\_SubChapter="P11\*Total quantum by\*subchapter after quantum\*adjustment by \*chapter (QR2)-APC Quantum\*by subchapter\_\_"

P11\_APC\_OPROC\_quantum\_by\_chapter="P11\*Total quantum by\*subchapter after quantum\*adjustment by \*chapter (QR2)-OPROC and APC \*Quantum\*by subchapter\_\_"

P112\_HRG="P112\*Prices after \*implementing SMF\*HRG \_\_"

P112\_Chapter="P112\*Prices after \*implementing SMF\*Chapter \_\_"

P112\_OPROC="P112\*Prices after \*implementing SMF\*OPROC \_\_"

P112\_DC="P112\*Prices after \*implementing SMF\*DC \_\_"

P112\_EL="P112\*Prices after \*implementing SMF\*EL \_\_"

P112\_NE ="P112\*Prices after \*implementing SMF\*NE \_\_"

P112\_Long\_Stay\_Payment="P112\*Prices after \*implementing SMF\*Long Stay Payment \_\_"

P112\_SSEM\_Banding="P112\*Prices after \*implementing SMF\*SSEM Banding \_\_"

P112\_SSEM\_Pct="P112\*Prices after \*implementing SMF\*SSEM % \_\_"

P112\_SSEM\_Tariff="P112\*Prices after \*implementing SMF\*SSEM Tariff\_\_"

P12\_Sub\_Chapter="P12\*Total quantum \*by chapter after \*implementing SMF\*Sub-chapter"

P12\_OPROC\_Quantum\_by\_HRG="P12\*Total quantum \*by chapter after \*implementing SMF\*OPROC Quantum by HRG\_\_"

P12\_APC\_Quantum\_by\_HRG="P12\*Total quantum \*by chapter after \*implementing SMF\*APC Quantum by HRG\_\_"

P12\_APC\_OPROC\_Quantum\_by\_HRG="P12\*Total quantum \*by chapter after \*implementing SMF\*OPROC and APC \*Quantum by HRG\_\_"

P12\_Sub\_\_Chapter="P12\*Total quantum \*by chapter after \*implementing SMF\*Subchapter"

P12\_OPROC\_quantum\_by\_SubChapter="P12\*Total quantum \*by chapter after \*implementing SMF\*OPROC Quantum \*by subchapter"

P12\_APC\_quantum\_by\_SubChapter="P12\*Total quantum \*by chapter after \*implementing SMF\*APC Quantum \*by subchapter"

P12\_APC\_OPROC\_quantum\_by\_chapter="P12\*Total quantum \*by chapter after \*implementing SMF\*APC and OPROC\*Quantum \*by subchapter"

P13\_Chapter="P13\*Quantum reconciliation\*by chapter (QR3)\*-Chapter \_\_"

P13\_Chapter\_Sum="P13\*Quantum reconciliation\*by chapter (QR3)\*-Chapter Sum\_\_"

P13\_APC\_OPROC\_QR3="P13\*Quantum reconciliation\*by chapter (QR3)\*-APC and OPROC QR3\_\_"

P14\_HRG="P14\*Prices after implementing QR3\*HRG\_\_"

P14\_Chapter="P14\*Prices after implementing QR3\*Chapter\_\_"

P14\_APCOPROC\_subchapter\_QR3="P14\*Prices after implementing QR3\*APC and OPROC\*subchapter QR3 factor\_\_"

P14\_OPROC="P14\*Prices after implementing QR3\*OPROC\_\_"

P14\_DC="P14\*Prices after implementing QR3\*DC\_\_"

P14\_EL="P14\*Prices after implementing QR3\*EL\_\_"

P14\_NE="P14\*Prices after implementing QR3\*NE\_\_"

P14\_Long\_Stay\_Payment="P14\*Prices after implementing QR3\*Long Stay Payment\_\_"

P14\_SSEM\_Banding="P14\*Prices after implementing QR3\*SSEM Banding\_\_"

P14\_SSEM\_Pct="P14\*Prices after implementing QR3\*SSEM % \_"

P14\_SSEM\_Tariff="P14\*Prices after implementing QR3\*SSEM Tariff\_\_"

P15\_Sub\_Chapter="P15\*Total quantum by \*chapter after \*implementing QR3\*-Subchapter\_\_"

P15\_OPROC\_Quantum\_by\_HRG="P15\*Total quantum by \*chapter after \*implementing QR3\*OPROC Quantum by HRG\_\_"

P15\_APC\_Quantum\_by\_HRG="P15\*Total quantum by \*chapter after \*implementing QR3\*APC Quantum by HRG\_\_"

P15\_APC\_OPROC\_Quantum\_by\_HRG="P15\*Total quantum by \*chapter after \*implementing QR3\*APC and OPROC\*Quantum by HRG\_\_"

P15\_Sub\_\_Chapter="P15\*Total quantum by \*chapter after \*implementing QR3\*-Sub\_chapter\_\_"

P15\_OPROC\_quantum\_by\_SubChapter="P15\*Total quantum by \*chapter after \*implementing QR3\*OPROC Quantum \*by subchapter\_\_"

P15\_APC\_quantum\_by\_SubChapter="P15\*Total quantum by \*chapter after \*implementing QR3\*APC Quantum\*by subchapter\_\_"

P15\_APC\_OPROC\_quantum\_by\_chapter="P15\*Total quantum by \*chapter after \*implementing QR3\*APC and OPROC\*Quantum\*by subchapter\_\_"

P16\_HRG="P16\*Prices after Scaling\*HRG\_\_"

P16\_Chapter="P16\*Prices after Scaling\*Chapter\_\_"

P16\_OPROC="P16\*Prices after Scaling\*OPROC\_\_"

P16\_DC="P16\*Prices after Scaling\*DC\_\_"

P16\_EL="P16\*Prices after Scaling\*EL\_\_"

P16\_NE="P16\*Prices after Scaling\*NE\_\_"

P16\_Long\_Stay\_Payment="P16\*Prices after Scaling\*Long Stay Payment\_\_"

P16\_SSEM\_Banding="P16\*Prices after Scaling\*SSEM Banding\_\_"

P16\_SSEM\_Pct="P16\*Prices after Scaling\*SSEM % \_"

P16\_SSEM\_Tariff="P16\*Prices after Scaling\*SSEM Tariff \_"

P17\_Sub\_Chapter="P17\*Total quantum by chapter\*after implementing scaling\*Subchapter\_\_"

P17\_OPROC\_Quantum\_by\_HRG="P17\*Total quantum by chapter\*after implementing scaling\*OPROC Quantum by HRG\_"

P17\_APC\_Quantum\_by\_HRG="P17\*Total quantum by chapter\*after implementing scaling\*APC Quantum by HRG\_"

P17\_APC\_OPROC\_Quantum\_by\_HRG="P17\*Total quantum by chapter\*after implementing scaling\*APC and OPROC \*Quantum by HRG\_"

P17\_Sub\_\_Chapter="P17\*Total quantum by chapter\*after implementing scaling\*Sub-Chapter\_"

P17\_OPROC\_quantum\_by\_SubChapter="P17\*Total quantum by chapter\*after implementing scaling\*OPROC Quantum \*by subchapter\_"

P17\_APC\_quantum\_by\_SubChapter="P17\*Total quantum by chapter\*after implementing scaling\*APC Quantum \*by subchapter\_"

P17\_APC\_OPROC\_quantum\_by\_chapter="P17\*Total quantum by chapter\*after implementing scaling\*APC and OPROC Quantum \*by subchapter\_"

P17\_APC\_Inlier\_Quantum="P17\*Total quantum by chapter\*after implementing scaling\*APC and OPROC Quantum \*APC Inlier Quantum\_"

P172\_Sub\_Chapter="P172\*For CNST (Uplifts \*excluding CNST applied)\*Subchapter\_\_"

P172\_OPROC\_QuantSubchap\_eCNST="P172\*For CNST (Uplifts \*excluding CNST applied)\*OPROC Quantum by subchapter\*(Uplifted to 17/18, \*excluding CNST)\_\_"

P172\_APC\_QuantSubchap\_inlier="P172\*For CNST (Uplifts \*excluding CNST applied)\*APC Quantum by subchapter\*(Inlier Only - uplifted \*to 17/18, \*excluding CNST)\_\_"

P172\_APC\_OPROC\_QuanSubchap\_Inl="P172\*For CNST (Uplifts \*excluding CNST applied)\*APC and OPROC \*Quantum by subchapter\*(Inlier only - uplifted \*to 17/18, \*excluding CNST)\_\_"

P18\_HRG="P18\*Final prices \*after all uplifts\*HRG\_\_"

P18\_Chapter="P18\*Final prices \*after all uplifts\*Chapter\_\_"

P18\_OPROC="P18\*Final prices \*after all uplifts\*OPROC\_\_"

P18\_DC="P18\*Final prices \*after all uplifts\*DC\_\_"

P18\_EL="P18\*Final prices \*after all uplifts\*EL\_\_"

P18\_NE="P18\*Final prices \*after all uplifts\*NE\_\_"

P18\_Long\_Stay\_Payment="P18\*Final prices \*after all uplifts\*Long Stay Payment\_\_"

P18\_SSEM\_Banding="P18\*Final prices \*after all uplifts\*SSEM Banding\_\_"

P18\_SSEM\_Pct="P18\*Final prices \*after all uplifts\*SSEM % \_"

P18\_SSEM\_Tariff="P18\*Final prices \*after all uplifts\*SSEM Tariff \_"

P19\_Sub\_Chapter="P19\*Final total quantum \*by chapter after implementing\*all uplifts\*-SubChapter \_\_"

P19\_OPROC\_Quantum\_by\_HRG="P19\*Final total quantum \*by chapter after implementing\*all uplifts\*OPROC Quantum by HRG"

P19\_APC\_Quantum\_by\_HRG="P19\*Final total quantum \*by chapter after implementing\*all uplifts\*APC Quantum by HRG"

P19\_APC\_OPROC\_Quantum\_by\_HRG="P19\*Final total quantum \*by chapter after implementing\*all uplifts\*APC and OPROC\*Quantum by HRG"

P19\_Sub\_\_Chapter="P19\*Final total quantum \*by chapter after implementing\*all uplifts\*-Sub\_Chapter \_\_"

P19\_OPROC\_quantum\_by\_SubChapter="P19\*Final total quantum \*by chapter after implementing\*all uplifts\*OPROC Quantum \*by subchapter \_\_"

P19\_APC\_quantum\_by\_SubChapter="P19\*Final total quantum \*by chapter after implementing\*all uplifts\*APC Quantum \*by subchapter \_\_"

P19\_APC\_OPROC\_quantum\_by\_chapter="P19\*Final total quantum \*by chapter after implementing\*all uplifts\*APC and OPROC \*Quantum \*by subchapter \_\_"

;

**run**;

/\*Step 3--QUANTUM RECONCILIATION BY Sub Chapter----------------------- End here \*/

/\*Step 4-- STEP BY STEP OP-DC-EL-NE Start from here \*/

/\*Create and format STEP BY STEP OP-DC-EL-NE tab\*/

ods tagsets.ExcelXP options(sheet\_name='STEP BY STEP OP-DC-EL-NE'

suppress\_bylines='yes'

frozen\_headers='1'

FROZEN\_ROWHEADERS='6'

autofit\_height='Yes'

absolute\_column\_width='6, 6, 7, 50, 8, 8, 8, 7, 8, 7, 7, 10, 10, 10,10, 10, 10, 10, 10, 10, 10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,14,14,14,14'

pages\_fitwidth='1'

pages\_fitheight='9'

FitToPage="yes"

autofilter='1-70');

/\*Define STEP BY STEP OP-DC-EL-NE tab output variables\*/

**proc** **print** data=Modelout.STEP\_BY\_STEP\_OP\_DC\_EL\_NE split='\*' n noobs;

var Spell\_HRG SpellHRGChapter SpellHRGSubchapter HRG\_Name OPROC\_currency\_applicable

/ style(Column)=[background=#e6ffff tagattr='format:#,##'] style(header)=[background=#e6ffff];

var P0\_OPROC P0\_DC P0\_EL P0\_NE / style(Column)=[background=#e6e6ff] style(header)=[background=#e6e6ff];

var P1\_OPROC P1\_DC P1\_EL P1\_NE / style(Column)=[background=#c2c2a3 tagattr="format:###,###,###.##"] style(header)=[background=#c2c2a3];

var QR1/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var P3\_OPROC P3\_DC P3\_EL P3\_NE / style(Column)=[background=#a5bcd9 tagattr="format:###,###,###.##"] style(header)=[background=#a5bcd9];

var Cost\_base\_adjustment/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00];

var P5\_OPROC P5\_DC P5\_EL P5\_NE / style(Column)=[background=#d1e2b2 tagattr="format:###,###,###.##"] style(header)=[background=#d1e2b2];

var P6\_OPROC P6\_DC P6\_EL P6\_NE / style(Column)=[background=#ffe6e6 tagattr="format:###,###,###.##"] style(header)=[background=#ffe6e6];

label

Spell\_HRG="Spell\*HRG\_\_"

SpellHRGChapter ="Spell HRG\*chapter\_\_"

SpellHRGSubchapter="Spell HRG\*sub - \*chapter\_\_"

HRG\_Name="HRG\*name"

OPROC\_currency\_applicable="OPROC\*currency\*applicable\_\_"

P0\_OPROC="P0\*Exception flag:\*1 = HRG manual adj, 0 = no manual adj \*OPROC Tariff\_\_"

P0\_DC = "P0\*Exception flag:\*1 = HRG manual adj, 0 = no manual adj \*DC Tariff\_\_"

P0\_EL = "P0\*Exception flag:\*1 = HRG manual adj, 0 = no manual adj \*EL Tariff\_\_"

P0\_NE= "P0\*Exception flag:\*1 = HRG manual adj, 0 = no manual adj \*NE Tariff\_\_"

P1\_OPROC="P1\*SQL\_Out\*OPROC Tariff\_\_"

P1\_DC = "P1\*SQL\_Output\*DC Tariff\_\_"

P1\_EL = "P1\*SQL\_Output\*EL Tariff\_\_"

P1\_NE= "P1\*SQL\_Output\*NE Tariff\_\_"

QR1="P2\*Reference costs quantum reconciliation factor (QR1)\_\_"

P3\_OPROC="P3\*Modelled prices after QR1 factor\*OPROC Tariff\_\_"

P3\_DC = "P3\*Modelled prices after QR1 factor\*DC Tariff\_\_"

P3\_EL = "P3\*Modelled prices after QR1 factor\*EL Tariff\_\_"

P3\_NE= "P3\*Modelled prices after QR1 factor\*NE Tariff\_\_"

Cost\_base\_adjustment="P4\*Cost base adjustment (CB)\_\_"

P5\_OPROC="P5\*Prices after CB\*implementation\*OPROC Tariff\_\_"

P5\_DC = "P5\*Prices after CB\*implementation\*DC Tariff\_\_"

P5\_EL = "P5\*Prices after CB\*implementation\*EL Tariff\_\_"

P5\_NE = "P5\*Prices after CB\*implementation\*NE Tariff\_\_"

P6\_OPROC="P6\*Prices after inflation, \*efficiency and CNST adjustments \*to a 18/19 financial year basis\*OPROC Tariff\_\_"

P6\_DC = "P6\*Prices after inflation, \*efficiency and CNST adjustments \*to a 18/19 financial year basis\*DC Tariff\_\_"

P6\_EL = "P6\*Prices after inflation, \*efficiency and CNST adjustments \*to a 18/19 financial year basis\*EL Tariff\_\_"

P6\_NE = "P6\*Prices after inflation, \*efficiency and CNST adjustments \*to a 18/19 financial year basis\*NE Tariff\_\_"

P7\_OPROC="P7\*Manual Adjustments\*OPROC Tariff\_\_"

P7\_DC = "P7\*Manual Adjustments\*DC Tariff\_\_"

P7\_EL = "P7\*Manual Adjustments\*EL Tariff\_\_"

P7\_NE = "P7\*Manual Adjustments\*NE Tariff\_\_"

P8\_OPROC="P8\*Prices before QR2 implementation\*OPROC Tariff\_\_"

P8\_DC = "P8\*Prices before QR2 implementation\*DC Tariff\_\_"

P8\_EL = "P8\*Prices before QR2 implementation\*EL Tariff\_\_"

P8\_NE = "P8\*Prices before QR2 implementation\*NE Tariff\_\_"

P9\_QR2="P9\*Manual adjustments quantum reconciliation factor (QR2)\_\_"

P10\_OPROC="P10\*Prices after QR2 implementation\*OPROC Tariff\_\_"

P10\_DC = "P10\*Prices after QR2 implementation\*DC Tariff\_\_"

P10\_EL = "P10\*Prices after QR2 implementation\*EL Tariff\_\_"

P10\_NE = "P10\*Prices after QR2 implementation\*NE Tariff\_\_"

SMF="P11\*Smoothing Factor adj (SMF)\_\_"

P12\_OPROC="P12\*Prices after SMF adj\*OPROC Tariff\_\_"

P12\_DC = "P12\*Prices after SMF adj\*DC Tariff\_\_"

P12\_EL = "P12\*Prices after SMF adj\*EL Tariff\_\_"

P12\_NE = "P12\*Prices after SMF adj\*NE Tariff\_\_"

P13\_QR3="P13\*Final quantum reconciled prices (QR3)\_\_"

P14\_OPROC="P14\*Prices after QR3 implementation\*OPROC Tariff\_\_"

P14\_DC = "P14\*Prices after QR3 implementation\*DC Tariff\_\_"

P14\_EL = "P14\*Prices after QR3 implementation\*EL Tariff\_\_"

P14\_NE = "P14\*Prices after QR3 implementation\*NE Tariff\_\_"

P15\_SCF="P15\*Scaling Factor Adj (SCF)\_\_"

P16\_OPROC="P16\*Final prices before 18/19 cost uplift assumptions\*OPROC Tariff\_\_"

P16\_DC = "P16\*Final prices before 18/19 cost uplift assumptions\*DC Tariff\_\_"

P16\_EL = "P16\*Final prices before 18/19 cost uplift assumptions\*EL Tariff\_\_"

P16\_NE = "P16\*Final prices before 18/19 cost uplift assumptions\*NE Tariff\_\_"

P17\_Inflation="P17\*17/18 Inflation\_\_"

P18\_OPROC="P18\*Final Prices in 19/20 prices including inflation\*OPROC Tariff\_\_"

P18\_DC = "P18\*Final Prices in 19/20 prices including inflation\*DC Tariff\_\_"

P18\_EL = "P18\*Final Prices in 19/20 prices including inflation\*EL Tariff\_\_"

P18\_NE = "P18\*Final Prices in 19/20 prices including inflation\*NE Tariff\_\_"

P19\_Efficiency="P19\*19/20 Efficiency Factor\_\_"

P20\_OPROC="P20\*Final Prices in 19/20 prices, including inflation and efficiency assumptions\*OPROC Tariff\_\_"

P20\_DC = "P20\*Final Prices in 19/20 prices, including inflation and efficiency assumptions\*DC Tariff\_\_"

P20\_EL = "P20\*Final Prices in 19/20 prices, including inflation and efficiency assumptions\*EL Tariff\_\_"

P20\_NE = "P20\*Final Prices in 19/20 prices, including inflation and efficiency assumptions\*NE Tariff\_\_"

P21\_19\_20\_CNST="P21\*19/20 CNST\_\_"

P22\_OPROC="P22\*FINAL PRICES in 19/20 prices, including inflation, efficiency and CNST assumptions\*OPROC Tariff\_\_"

P22\_DC = "P22\*FINAL PRICES in 19/20 prices, including inflation, efficiency and CNST assumptions\*DC Tariff\_\_"

P22\_EL = "P22\*FINAL PRICES in 19/20 prices, including inflation, efficiency and CNST assumptions\*EL Tariff\_\_"

P22\_NE = "P22\*FINAL PRICES in 19/20 prices, including inflation, efficiency and CNST assumptions\*NE Tariff\_\_"

;

**run**;

/\*Step 4-- STEP BY STEP OP-DC-EL-NE End here \*/

/\*Step 5-- APC\_OPROC\_Model\_Linked\_Sheet Start from here \*/

ods tagsets.ExcelXP options(sheet\_name='Linked Sheet'

autofilter='NO' embedded\_titles ="yes" autofit\_height ="yes" frozen\_headers="yes" frozen\_headers="2"

FROZEN\_ROWHEADERS='3'

absolute\_column\_width='6,8,50,10,10,10,10,10,10,10,10,10,10,10');

**proc** **print** data = ModelOUt.APC\_OPROC\_MODEL\_LINKEDSHEET noobs label split= '\*';

var Year HRG\_Code HRG\_Name /style(header)=[background=#b3ffff];

var OPROC\_Tariff Combined\_DC\_EL\_Prices DC\_Prices EL\_Prices /style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###.##"];

var EL\_TRIM\_POINTS /style(header)=[background=#b3ffff];

var NE\_Prices /style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###.##"];

var NE\_TRIM\_POINTS /style(header)=[background=#b3ffff];

var EBD\_Prices /style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###.##"];

var SSEM\_applicable /style(header)=[background=#b3ffff];

var SSEM\_Pct /style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:#,##0.00%"];

var SSEM\_Price /style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###.##"];

label

HRG\_Code="HRG Code\_\_"

OPROC\_Tariff="Outpatient \*procedure \*tariff (£)\_\_"

Combined\_DC\_EL\_Prices="Combined day \*case /ordinary \*elective spell \*tariff (£)\_"

DC\_Prices="Day case \*spell tariff (£)\_\_"

EL\_Prices="Ordinary \*elective \*spell tariff (£)\_\_"

EL\_TRIM\_POINTS="Ordinary \*elective long \*stay trim point \*(days)\_\_"

NE\_Prices="Non-elective \*spell tariff (£)\_\_"

NE\_TRIM\_POINTS="Non-elective \*long stay \*trim point \*(days)\_\_"

EBD\_Prices="Per day long\* stay payment \*(for days exceeding \*trim point) (£)\_\_"

SSEM\_applicable="Reduced short \*stay \*emergency \*tariff \*applicable?\_\_"

SSEM\_Pct="% applied in \*calculation of \*reduced short \*stay \*emergency \*tariff\_\_"

SSEM\_Price="Reduced short \*stay emergency \*tariff (£)\_\_"

;

**run**;

/\*Step 5-- APC\_OPROC\_Model\_Link\_Sheet End here \*/

**quit**;

ods tagsets.ExcelXP close;

ods listing;

**data** P\_Finihsed;

Congratulations="Excel report is Ready";

Location="\\irnarch\sas\_data\PID\_Proj\_1718\APC\APC\_FY1920\Excel\_outputs\ ";

Filename="SASOUT 17-18 APC\_OPROC model.xml";

**run**;

**proc** **sql**;

Title "Congratulations ! Excel report is ready Now!! ";

Title1 "Excel report location:\\irnarch\sas\_data\PID\_Proj\_1718\APC\APC\_FY1920\Excel\_outputs\ ";

Title1 "File Name:SASOUT 17-18 APC\_OPROC model.xml.";

Title "Have a nice day!!";

Title1;

Title;

select \* from P\_Finihsed

;

**quit**;

/\*--------------------------------------------------------------\*/

\* Step3\_5 Create all APC\_OPROC Excel Model worksheets\_2

/\* Version: 1.02 \*/

/\* Author: PID \*/

/\* Date: September 2017 \*/

/\*--------------------------------------------------------------\*/

/\* Description: \*/

/\* Generate an excel file which contains 5 tabs for APC results:\*/

/\* 1.STEP BY STEP OP-DC-EL-NE; \*/

/\* 2.Long Stay Payment; \*/

/\* 3.STEP BY STEP TRIMPOINT and LSP; \*/

/\* 4.QUANTUM RECONCILIATION BY Sub Chapter \*/

/\* 5.APC OPROC Model LinkSheet \*/

/\*--------------------------------------------------------------\*/

/\*ODS to create Excel workbook output with multiple sheets\*/

ods \_all\_ close;

ods tagsets.ExcelXP path="\\irnarch\sas\_data\Tariff Rebuild\Models for demo\WIP\TC\_FY1920\Excel\_outputs" file="APC\_OPROC model.xml" style=Printer;

/\*Step 1-- Long Stay Payment Start from here \*/

ods tagsets.ExcelXP options(sheet\_name='Long Stay Payment'

autofilter='NO' embedded\_titles ="yes" autofit\_height ="yes" frozen\_headers="yes" frozen\_headers="2"

absolute\_column\_width='6,50,6,8,8,8,3,10,10,10,10,10,10');

**proc** **print** data = Modelout.Long\_Stay\_Payment noobs label split= '\*';

var HRG HRG\_Name Chapter Proportion\_Children\_Activity Congenital\_HRG\_Flag AG18\_YearsandUnder\_HRG\_Flag LU /style(header)=[background=#ecffb3];

var CLEAN\_EBD\_ACT / style(header)=[background=#ecffb3] style(Column)=[ tagattr="format:###,###,###"] ;

var CLEAN\_EBD\_TC LSP\_UC Limit\_within\_100\_to\_500 / style(header)=[background=#ecffb3] style(Column)=[ tagattr="format:###,###,###.##"] ;

var Zero\_price\_HRG / style(header)=[background=#ecffb3] ;

var Long\_Stay\_Payment / style(header)=[background=#ecffb3] style(Column)=[ tagattr="format:###,###,###.##"];

;

label

HRG="HRG \_\_"

HRG\_Name="HRG Name\_\_"

Chapter="Chapter \_\_"

Proportion\_Children\_Activity="Proportion\*of children\*activity\_\_"

Congenital\_HRG\_Flag="Congenital \*HRG flag\_\_"

AG18\_YearsandUnder\_HRG\_Flag="18 years \*and under \*HRG flag\_\_"

LU="LU\_\_"

CLEAN\_EBD\_ACT="CLEAN EBD \*ACT \_\_"

CLEAN\_EBD\_TC="CLEAN EBD \*TC \_\_"

LSP\_UC="LSP UC\_\_"

Limit\_within\_100\_to\_500="Limiting within \*£100 to £500 \_\_"

Zero\_price\_HRG="Zero priced HRG \*flag (1 if the HRG is zero-priced, 0 \*otherwise)\_\_"

Long\_Stay\_Payment="Long Stay Payments\_\_"

;

**run**;

/\*Step 1-- Long Stay Payment End here \*/

/\*Step 2-- STEP BY STEP TRIMPOINT and LSP------ Start from here \*/

/\*Create and format STEP BY STEP TRIMPOINT and LSP tab\*/

ods tagsets.ExcelXP options(sheet\_name='STEP BY STEP TRIMPOINT and LSP'

suppress\_bylines='yes'

frozen\_headers='1'

FROZEN\_ROWHEADERS='4'

autofit\_height='Yes'

absolute\_column\_width='6, 6, 7, 50, 8, 8, 8, 7, 8, 7, 7, 10, 10, 10,10, 10, 10, 10, 10, 10, 10,10,10,10'

pages\_fitwidth='1'

pages\_fitheight='9'

FitToPage="yes"

autofilter='1-66');

/\*Define STEP BY STEP TRIMPOINT and LSP tab output variables\*/

**proc** **print** data=Modelout.STEP\_BY\_STEP\_TRIMPOINT\_and\_LSP split='\*' n noobs;

var Spell\_HRG SpellHRGChapter SpellHRGSubChapter HRG\_Description / style(Column)=[background=#e6ffff tagattr='format:#,##'] style(header)=[background=#e6ffff];

/\*Part1\*/

var EL\_Trim NE\_Trim Long\_Stay\_Payment / style(Column)=[background=#ffb699]style(header)=[background=#ffb699];

/\*Part2 \*/

var EL\_Trim\_5DRule NE\_Trim\_5DRule / style(Column)=[background=#ffb6c1 ] style(header)=[background=#ffb6c1];

/\*Part 3\*/

var Manual\_TP\_EL Manual\_TP\_NE / style(Column)=[background=#cfe2cf] style(header)=[background=#cfe2cf] ;

/\*Part 4 \*/

var DC\_EL\_SemiAuto NE\_SemiAuto / style(Column)=[background=#dabcf6] style(header)=[background=#dabcf6];

/\*Part 5 \*/

var Fin\_EL Fin\_NE / style(Column)=[background=#afeeee ] style(header)=[background=#afeeee];

/\*Part 6 \*/

var Initial\_longStayPayment / style(Column)=[background=#e6ffff ] style(header)=[background=#e6ffff];

/\*Part 7 \*/

var LSP\_after\_QR1 / style(Column)=[background=#fff5e6 ] style(header)=[background=#fff5e6];

/\*Part 8 \*/

var LSP\_after\_CB / style(Column)=[background=#88cc00 ] style(header)=[background=#88cc00];

/\*Part 9 \*/

var LSP\_Uplift18\_19 / style(Column)=[background=#efdcc3 ] style(header)=[background=#efdcc3];

/\*Part 10 \*/

var LSP\_QR2/ style(Column)=[background=#ffe6e9 ] style(header)=[background=#ffe6e9];

/\*Part 11 \*/

var LSP\_SMF/ style(Column)=[background=#bbd0f7] style(header)=[background=#bbd0f7 ];

/\*Part 12 \*/

var LSP\_QR3 / style(Column)=[background=#9fc69f] style(header)=[background=#9fc69f];

/\*Part 13 \*/

var LSP\_SCF / style(Column)=[background=#c69f6c] style(header)=[background=#c69f6c];

/\*Part 14 \*/

var LSP\_1920Basis / style(Column)=[background=#dcd8f3 ] style(header)=[background=#dcd8f3];

label

Spell\_HRG="Spell\*HRG\_\_"

SpellHRGChapter="Spell\*HRG\*chapter\_\_"

SpellHRGSubChapter="Spell HRG\*subchapter\_\_"

HRG\_Description="HRG description"

EL\_Trim="P1-SQL output\*EL\_Trim\_\_"

NE\_Trim="P1-SQL output\*NE\_Trim\_\_"

Long\_Stay\_Payment="P1-SQL output\*Long\_Stay\_Payment\_\_"

EL\_Trim\_5DRule="P2-Minimum 5 day rule\*EL\_Trim\_\_"

NE\_Trim\_5DRule="P2-Minimum 5 day rule\*NE\_Trim\_\_"

Manual\_TP\_EL="P3-2019/20 Manual\*EL\_Trim\_\_"

Manual\_TP\_NE="P3-2019/20 Manual\*NE\_Trim\_\_"

DC\_EL\_SemiAuto="P4-2019/20\*Semi-Automated\*EL\_Trim\_\_"

NE\_SemiAuto="P4-2019/20\*Semi-Automated\*NE\_Trim\_\_"

Fin\_EL="P5-Final trim points\*EL\_Trim\_\_"

Fin\_NE="P5-Final trim points\*NE\_Trim\_\_"

Initial\_longStayPayment="P6-Initial long stay\*payments\*(LSP)\_\_"

LSP\_after\_QR1="P7-LSP after QR1\*implementation\_\_"

LSP\_after\_CB="P8-Long stay \*payments after CB adjustment\*implementation\_\_"

LSP\_Uplift18\_19="P9-Modelled LSP\*prices adjusted to \*a 18/19 basis\_\_"

LSP\_QR2="P10-Long stay payments\*after QR2 \*implementation \*adjustment \*implementation\_\_"

LSP\_SMF="P11-Long stay payments\* after SMF\*adjustment \*implementation\_\_"

LSP\_QR3="P12-Long stay\*payments after\* QR3\*implementation\_\_"

LSP\_SCF="P13-Long stay\* payments after\* SCF\* implementation\_\_"

LSP\_1920Basis="P14-Long stay\*payments on a 19/20 \*basis (applying inflation, \*efficiency \*and CNST)\_\_"

;

**run**;

/\*Step 2-- STEP BY STEP TRIMPOINT and LSP----------------- End here \*/

/\*Step 3-- QUANTUM RECONCILIATION BY Sub Chapter-------------------------------- Start from here \*/

/\*Create and format QUANTUM RECONCILIATION BY Sub Chapter tab\*/

ods tagsets.ExcelXP options(sheet\_name='QUANTUM RECONCILIATION BY SubChapter'

suppress\_bylines='yes'

frozen\_headers='1'

FROZEN\_ROWHEADERS='3'

autofit\_height='Yes'

absolute\_column\_width='6, 50, 7, 7, 8, 8, 8, 7, 8, 7, 7, 10, 10, 10,10, 10, 10, 10, 10, 10, 10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10,'

pages\_fitwidth='1'

pages\_fitheight='9'

FitToPage="yes"

autofilter='1-170');

/\*Define STEP BY STEP QUANTUM RECONCILIATION BY Sub Chapter variables\*/

**proc** **print** data=Modelout.QUANTUM\_RECONCILIATION\_BySubChap split='\*' n noobs;

var HRG Description OPROC\_currency\_applicable / style(Column)=[background=#e6ffff tagattr='format:#,##'] style(header)=[background=#e6ffff];

var P0\_OPROC P0\_DC P0\_EL P0\_NE / style(header)=[background=#eda891];

var P1\_OPROC\_TA P1\_DC\_TA P1\_EL\_TA P1\_NE\_TA P1\_NE\_AT\_less\_SSEM P1\_LSP\_days P1\_SSEM / style(header)=[background=#eae395] style(Column)=[ tagattr="format:###,###,###"];

/\* Part 2 \*/

var P2\_OPROC P2\_DC P2\_EL P2\_NE P2\_LSP / style(header)=[background=#f0e6ff] style(Column)=[ tagattr="format:###,###,###.##"];

var P2\_SSEM\_Banding P2\_SSEM\_Pct/style(header)=[background=#f0e6ff];

var P2\_SSEM\_Tariff / style(header)=[background=#f0e6ff] style(Column)=[ tagattr="format:###,###,###.##"];

/\* Part 3 \*/

var P3\_Sub\_Chapter /style(header)=[background=#f0e6ff];

var P3\_OPROC\_quantum\_by\_HRG P3\_APC\_Quantum\_by\_HRG /style(header)=[background=#f0e6ff] style(Column)=[ tagattr="format:###,###,###.##"];

var P3\_Sub\_\_Chapter /style(header)=[background=#f0e6ff] ;

var P3\_OPROC\_quantum\_by\_SubChapter P3\_APC\_quantum\_by\_SubChapter P3\_APC\_OPROC\_quantum\_by\_chapter /style(header)=[background=#f0e6ff] style(Column)=[ tagattr="format:###,###,###.##"];

/\* Part 4 \*/

var P4\_OPROC P4\_DC P4\_EL P4\_NE P4\_Long\_Stay\_Payment /style(header)=[background=#ffe6e9] style(Column)=[ tagattr="format:###,###,###.##"];

var P4\_SSEM\_Banding P4\_SSEM\_Pct /style(header)=[background=#ffe6e9];

var P4\_SSEM\_Tariff /style(header)=[background=#ffe6e9] style(Column)=[ tagattr="format:###,###,###.##"];

/\*Part 5 \*/

var P5\_OPROC P5\_DC P5\_EL P5\_NE P5\_Long\_Stay\_Payment /style(header)=[background=#**99e600**] style(Column)=[ tagattr="format:###,###,###.##"];

var P5\_SSEM\_Banding P5\_SSEM\_Pct /style(header)=[background=#**99e600**];

var P5\_SSEM\_Tariff /style(header)=[background=#**99e600**] style(Column)=[ tagattr="format:###,###,###.##"];

/\*Part 6 \*/

var P6\_Sub\_Chapter /style(header)=[background=#ffe6e9];

var P6\_OPROC\_Quantum\_by\_HRG P6\_APC\_Quantum\_by\_HRG P6\_Sub\_\_Chapter P6\_OPROC\_quantum\_by\_SubChapter

P6\_APC\_quantum\_by\_SubChapter P6\_APC\_OPROC\_quantum\_by\_chapter /style(header)=[background=#ffe6e9] style(Column)=[ tagattr="format:###,###,###.##"];

/\*Part 7 \*/

var P7\_Sub\_Chapter /style(header)=[background=#**99e600**];

var P7\_OPROC\_Quantum\_by\_HRG P7\_APC\_Quantum\_by\_HRG /style(header)=[background=#**99e600**] style(Column)=[ tagattr="format:###,###,###.##"];

var P7\_Sub\_\_Chapter /style(header)=[background=#**99e600**];

var P7\_OPROC\_quantum\_by\_SubChapter P7\_APC\_quantum\_by\_SubChapter P7\_APC\_OPROC\_quantum\_by\_chapter /style(header)=[background=#**99e600**] style(Column)=[ tagattr="format:###,###,###.##"];

/\*Part 8 \*/

var P8\_Sub\_Chapter /style(header)=[background=#e6ffff];

var P8\_OPROC\_Quantum\_by\_HRG P8\_APC\_Quantum\_by\_HRG /style(header)=[background=#e6ffff] style(Column)=[ tagattr="format:###,###,###.##"];

var P8\_Sub\_\_Chapter /style(header)=[background=#e6ffff] ;

var P8\_OPROC\_quantum\_by\_SubChapter P8\_APC\_quantum\_by\_SubChapter P8\_APC\_OPROC\_quantum\_by\_chapter

/style(header)=[background=#e6ffff] style(Column)=[ tagattr="format:###,###,###.##"];

/\*Part 9 \*/

var P9\_Chapter P9\_Chapter\_Sum P9\_APC\_OPROC\_QR2 /style(header)=[background=#**669900**];

/\*Part 10 \*/

var P10\_HRG P10\_Chapter /style(header)=[background= #936c39];

var P10\_APCOPROC\_subchapter\_QR2 P10\_OPROC P10\_DC P10\_EL P10\_NE P10\_Long\_Stay\_Payment /style(header)=[background= #936c39] style(Column)=[ tagattr="format:###,###,###.##"];

var P10\_SSEM\_Banding P10\_SSEM\_Pct /style(header)=[background= #936c39];

var P10\_SSEM\_Tariff /style(header)=[background= #936c39] style(Column)=[ tagattr="format:###,###,###.##"];

/\*Part 11 \*/

var P11\_Sub\_Chapter /style(header)=[background= #936c39];

var P11\_OPROC\_Quantum\_by\_HRG P11\_APC\_Quantum\_by\_HRG P11\_APC\_OPROC\_Quantum\_by\_HRG /style(header)=[background= #936c39] style(Column)=[ tagattr="format:###,###,###.##"];

var P11\_Sub\_\_Chapter /style(header)=[background= #936c39];

var P11\_OPROC\_quantum\_by\_SubChapter P11\_APC\_quantum\_by\_SubChapter P11\_APC\_OPROC\_quantum\_by\_chapter/style(header)=[background= #936c39] style(Column)=[ tagattr="format:###,###,###.##"];

/\*Part 112 \*/

var P112\_HRG P112\_Chapter /style(header)=[background=#ffcc99];

var P112\_OPROC P112\_DC P112\_EL P112\_NE P112\_Long\_Stay\_Payment /style(header)=[background=#ffcc99] style(Column)=[ tagattr="format:###,###,###.##"];

var P112\_SSEM\_Banding P112\_SSEM\_Pct /style(header)=[background=#ffcc99];

var P112\_SSEM\_Tariff /style(header)=[background=#ffcc99] style(Column)=[ tagattr="format:###,###,###.##"];

/\*Part 12 \*/

var P12\_Sub\_Chapter /style(header)=[background=#ffcc99];

var P12\_OPROC\_Quantum\_by\_HRG P12\_APC\_Quantum\_by\_HRG P12\_APC\_OPROC\_Quantum\_by\_HRG /style(header)=[background=#ffcc99] style(Column)=[ tagattr="format:###,###,###.##"];

var P12\_Sub\_\_Chapter /style(header)=[background=#ffcc99];

var P12\_OPROC\_quantum\_by\_SubChapter

P12\_APC\_quantum\_by\_SubChapter P12\_APC\_OPROC\_quantum\_by\_chapter/style(header)=[background=#ffcc99] style(Column)=[ tagattr="format:###,###,###.##"];

/\*Part 13 \*/

var P13\_Chapter P13\_Chapter\_Sum P13\_APC\_OPROC\_QR3 /style(header)=[background=#5bd75b];

/\*Part 14 \*/

var P14\_HRG P14\_Chapter P14\_APCOPROC\_subchapter\_QR3/style(header)=[background=#e6ffff];

var P14\_OPROC P14\_DC P14\_EL P14\_NE P14\_Long\_Stay\_Payment /style(header)=[background=#e6ffff] style(Column)=[ tagattr="format:###,###,###.##"];

var P14\_SSEM\_Banding P14\_SSEM\_Pct /style(header)=[background=#e6ffff];

var P14\_SSEM\_Tariff /style(header)=[background=#e6ffff] style(Column)=[ tagattr="format:###,###,###.##"];

/\*Part 15 \*/

var P15\_Sub\_Chapter /style(header)=[background=#ffe6e9];

var P15\_OPROC\_Quantum\_by\_HRG P15\_APC\_Quantum\_by\_HRG P15\_APC\_OPROC\_Quantum\_by\_HRG /style(header)=[background=#ffe6e9] style(Column)=[ tagattr="format:###,###,###.##"];

var P15\_Sub\_\_Chapter /style(header)=[background=#ffe6e9];

var P15\_OPROC\_quantum\_by\_SubChapter

P15\_APC\_quantum\_by\_SubChapter P15\_APC\_OPROC\_quantum\_by\_chapter /style(header)=[background=#ffe6e9] style(Column)=[ tagattr="format:###,###,###.##"];

/\*Part 16 \*/

var P16\_HRG P16\_Chapter/style(header)=[background=#cce6ff];

var P16\_OPROC P16\_DC P16\_EL P16\_NE P16\_Long\_Stay\_Payment /style(header)=[background=#cce6ff] style(Column)=[ tagattr="format:###,###,###.##"];

var P16\_SSEM\_Banding P16\_SSEM\_Pct /style(header)=[background=#cce6ff];

var P16\_SSEM\_Tariff /style(header)=[background=#cce6ff] style(Column)=[ tagattr="format:###,###,###.##"];

/\*Part 17 \*/

var P17\_Sub\_Chapter/style(header)=[background=#f2ffcc];

var P17\_OPROC\_Quantum\_by\_HRG P17\_APC\_Quantum\_by\_HRG P17\_APC\_OPROC\_Quantum\_by\_HRG/style(header)=[background=#f2ffcc] style(Column)=[ tagattr="format:###,###,###.##"];

var P17\_Sub\_\_Chapter/style(header)=[background=#f2ffcc];

var P17\_OPROC\_quantum\_by\_SubChapter

P17\_APC\_quantum\_by\_SubChapter P17\_APC\_OPROC\_quantum\_by\_chapter P17\_APC\_Inlier\_Quantum /style(header)=[background=#f2ffcc] style(Column)=[ tagattr="format:###,###,###.##"];

var P172\_Sub\_Chapter/style(header)=[background=#f2ffcc];

var P172\_OPROC\_QuantSubchap\_eCNST P172\_APC\_QuantSubchap\_inlier P172\_APC\_OPROC\_QuanSubchap\_Inl/style(header)=[background=#f2ffcc] style(Column)=[ tagattr="format:###,###,###.##"];

/\*Part 18 \*/

var P18\_HRG P18\_Chapter/style(header)=[background=#d5f6f6];

var P18\_OPROC P18\_DC P18\_EL P18\_NE P18\_Long\_Stay\_Payment/style(header)=[background=#d5f6f6] style(Column)=[ tagattr="format:###,###,###.##"];

var P18\_SSEM\_Banding P18\_SSEM\_Pct/style(header)=[background=#d5f6f6];

var P18\_SSEM\_Tariff/style(header)=[background=#d5f6f6] style(Column)=[ tagattr="format:###,###,###.##"];

/\*Part 19 \*/

var P19\_Sub\_Chapter/style(header)=[background=#aed75b];

var P19\_OPROC\_Quantum\_by\_HRG P19\_APC\_Quantum\_by\_HRG P19\_APC\_OPROC\_Quantum\_by\_HRG/style(header)=[background=#aed75b] style(Column)=[ tagattr="format:###,###,###.##"];

var P19\_Sub\_\_Chapter/style(header)=[background=#aed75b];

var P19\_OPROC\_quantum\_by\_SubChapter

P19\_APC\_quantum\_by\_SubChapter P19\_APC\_OPROC\_quantum\_by\_chapter/style(header)=[background=#aed75b] style(Column)=[ tagattr="format:###,###,###.##"];

label

Description="HRG description"

P0\_OPROC="P0\*HRGs eligible for QR2\*1 = eligible, \*0 = not eligible\* OPROC\_"

P0\_DC = "P0\*HRGs eligible for QR2\*1 = eligible, \*0 = not eligible\* DC \_\_"

P0\_EL = "P0\*HRGs eligible for QR2\*1 = eligible, \*0 = not eligible\* EL \_\_"

P0\_NE= "P0\*HRGs eligible for QR2\*1 = eligible, \*0 = not eligible\* NE \_\_"

P1\_OPROC\_TA="P1\*Total activity (spells)\*OPROC\_"

P1\_DC\_TA="P1\*Total activity (spells)\*DC \_\_"

P1\_EL\_TA="P1\*Total activity (spells)\*EL \_\_"

P1\_NE\_TA="P1\*Total activity (spells)\*NE \_\_"

P1\_NE\_AT\_less\_SSEM="P1\*NE activities less SSEM\* (this only applies for HRGs that \*are eligible for\* SSEM\_)"

P1\_LSP\_days="P1\*LSP days \_\_"

P1\_SSEM="P1\*SSEM activities\_\_"

P2\_OPROC="P2\*Modelled prices pre \*quantum reconciliation (QR1)\*OPROC \_"

P2\_DC="P2\*Modelled prices pre \*quantum reconciliation (QR1)\*DC \_"

P2\_EL="P2\*Modelled prices pre \*quantum reconciliation (QR1)\*EL \_"

P2\_NE="P2\*Modelled prices pre \*quantum reconciliation (QR1)\*NE \_"

P2\_LSP="P2\*Modelled prices pre \*quantum reconciliation (QR1)\*LSP \_"

P2\_SSEM\_Banding="P2\*Modelled prices pre \*quantum reconciliation (QR1)\*SSEM\_Banding\_"

P2\_SSEM\_Pct="P2\*Modelled prices pre \*quantum reconciliation (QR1)\*SSEM % \_\_"

P2\_SSEM\_Tariff="P2\*Modelled prices pre \*quantum reconciliation (QR1)\*SSEM Tariff \_\_"

P3\_Sub\_Chapter="P3\*Total quantum by \*subchapter pre quantum \*reconciliation (QR1)\*Sub Chapter \_"

P3\_OPROC\_quantum\_by\_HRG="P3\*Total quantum by \*subchapter pre quantum \*reconciliation (QR1)\*OPROC quantum by HRG \_"

P3\_APC\_Quantum\_by\_HRG="P3\*Total quantum by \*subchapter pre quantum \*reconciliation (QR1)\*APC quantum by HRG \_"

P3\_Sub\_\_Chapter="P3\*Total quantum by \*subchapter pre quantum \*reconciliation (QR1)\*Sub-Chapter \_"

P3\_OPROC\_quantum\_by\_SubChapter="P3\*Total quantum by \*subchapter pre quantum \*reconciliation (QR1)\* OPROC quantum by chapter\_"

P3\_APC\_quantum\_by\_SubChapter="P3\*Total quantum by \*subchapter pre quantum \*reconciliation (QR1)\*APC quantum by chapter\_"

P3\_APC\_OPROC\_quantum\_by\_chapter="P3\*Total quantum by \*subchapter pre quantum \*reconciliation (QR1)\*APC and OPROC \*quantum by chapter\_"

P4\_OPROC="P4\*Prices post inflation,\*efficiency and CNST adjustments\*(prices before manual\* adjustments)\*OPROC \_"

P4\_DC="P4\*Prices post inflation,\*efficiency and CNST adjustments\*(prices before manual\* adjustments)\*DC \_"

P4\_EL="P4\*Prices post inflation,\*efficiency and CNST adjustments\*(prices before manual\* adjustments)\*EL \_"

P4\_NE="P4\*Prices post inflation,\*efficiency and CNST adjustments\*(prices before manual\* adjustments)\*NE \_"

P4\_Long\_Stay\_Payment="P4\*Prices post inflation,\*efficiency and CNST adjustments\*(prices before manual\* adjustments)\*Long Stay Payment \_"

P4\_SSEM\_Banding="P4\*Prices post inflation,\*efficiency and CNST adjustments\*(prices before manual\* adjustments)\*SSEM Banding \_"

P4\_SSEM\_Pct="P4\*Prices post inflation,\*efficiency and CNST adjustments\*(prices before manual\* adjustments)\*SSEM % \_"

P4\_SSEM\_Tariff="P4\*Prices post inflation,\*efficiency and CNST adjustments\*(prices before manual\* adjustments)\*SSEM Tariff \_\_"

P5\_OPROC="P5\*Prices after \*manual adjustments\* adjustments)\*OPROC \_"

P5\_DC="P5\*Prices after \*manual adjustments\* adjustments)\*DC \_"

P5\_EL="P5\*Prices after \*manual adjustments\* adjustments)\*EL \_"

P5\_NE="P5\*Prices after \*manual adjustments\* adjustments)\*NE \_"

P5\_Long\_Stay\_Payment="P5\*Prices after \*manual adjustments\* adjustments)\*Long Stay Payment \_"

P5\_SSEM\_Banding="P5\*Prices after \*manual adjustments\* adjustments)\*SSEM Banding \_"

P5\_SSEM\_Pct="P5\*Prices after \*manual adjustments\* adjustments)\*SSEM % \_"

P5\_SSEM\_Tariff="P5\*Prices after \*manual adjustments\* adjustments)\*SSEM Tariff\_\_"

P6\_Sub\_Chapter="P6\*Total quantum by subchapter\*before manual adjustment\*Sub\_Chapter"

P6\_OPROC\_Quantum\_by\_HRG="P6\*Total quantum by subchapter\*before manual adjustment\*OPROC Quantum by HRG\_\_"

P6\_APC\_Quantum\_by\_HRG="P6\*Total quantum by subchapter\*before manual adjustment\*APC Quantum by HRG\_\_"

P6\_Sub\_\_Chapter="P6\*Total quantum by subchapter\*before manual adjustment\*Subchapter\_\_"

P6\_OPROC\_quantum\_by\_SubChapter="P6\*Total quantum by subchapter\*before manual adjustment\*OPROC quantum\*by chapter\_\_"

P6\_APC\_quantum\_by\_SubChapter="P6\*Total quantum by subchapter\*before manual adjustment\*APC quantum by chapter\_\_"

P6\_APC\_OPROC\_quantum\_by\_chapter="P6\*Total quantum by subchapter\*before manual adjustment\*APC and OPROC\*quantum by chapter\_\_"

P7\_Sub\_Chapter="P7\*Total quantum by\*subchapter after\*manual adjustment\*Sub-Chapter\_\_"

P7\_OPROC\_Quantum\_by\_HRG="P7\*Total quantum by\*subchapter after\*manual adjustment\*OPROC Quantum by HRG\_\_"

P7\_APC\_Quantum\_by\_HRG="P7\*Total quantum by\*subchapter after\*manual adjustment\*APC Quantum by HRG\_\_"

P7\_Sub\_\_Chapter="P7\*Total quantum by\*subchapter after\*manual adjustment\*Sub\_Chapter\_\_"

P7\_OPROC\_quantum\_by\_SubChapter="P7\*Total quantum by\*subchapter after\*manual adjustment\*OPROC quantum\* by chapter\_\_"

P7\_APC\_quantum\_by\_SubChapter="P7\*Total quantum by\*subchapter after\*manual adjustment\*APC quantum\* by chapter\_\_"

P7\_APC\_OPROC\_quantum\_by\_chapter="P7\*Total quantum by\*subchapter after\*manual adjustment\*APC and OPROC\*quantum by chapter\_\_"

P8\_Sub\_Chapter="P8\*Total quantum by\*subchapter after manual\*adjustment excluding HRGs\*that incorporate\*OPATT prices\*Sub-Chapter\_\_"

P8\_OPROC\_Quantum\_by\_HRG="P8\*Total quantum by\*subchapter after manual\*adjustment excluding HRGs\*that incorporate\*OPROC Quantum by HRG\_\_"

P8\_APC\_Quantum\_by\_HRG="P8\*Total quantum by\*subchapter after manual\*adjustment excluding HRGs\*that incorporate\*APC Quantum by HRG\_\_"

P8\_Sub\_\_Chapter="P8\*Total quantum by\*subchapter after manual\*adjustment excluding HRGs\*that incorporate\*OPATT prices\*Sub\_chapter\_\_"

P8\_OPROC\_quantum\_by\_SubChapter="P8\*Total quantum by\*subchapter after manual\*adjustment excluding HRGs\*that incorporate\*OPATT prices\*OPROC quantum\*by chapter\_\_"

P8\_APC\_quantum\_by\_SubChapter="P8\*Total quantum by\*subchapter after manual\*adjustment excluding HRGs\*that incorporate\*OPATT prices\*APC quantum by chapter\_\_"

P8\_APC\_OPROC\_quantum\_by\_chapter="P8\*Total quantum by\*subchapter after manual\*adjustment excluding HRGs\*that incorporate\*OPATT prices\*APC and OPROC\*quantum by chapter\_\_"

P9\_Chapter="P9\*Quantum reconciliation\*by chapter (QR2)\*Chapter \_"

P9\_Chapter\_Sum="P9\*Quantum reconciliation\*by chapter (QR2)\*Chapter sum\_\_"

P9\_APC\_OPROC\_QR2="P9\*Quantum reconciliation\*by chapter (QR2)\*APC and OPROC\*QR2 \_\_"

P10\_HRG="P10\*Prices after quantum\*reconciliation by\*chapter (QR2)\*HRG \_\_"

P10\_Chapter="P10\*Prices after quantum\*reconciliation by\*chapter (QR2)\*Chapter \_\_"

P10\_APCOPROC\_subchapter\_QR2="P10\*Prices after quantum\*reconciliation by\*chapter (QR2)\*APC and OPROC\*subchapter QR2 factor\_\_"

P10\_OPROC="P10\*Prices after quantum\*reconciliation by\*chapter (QR2)\*OPROC \_\_"

P10\_DC="P10\*Prices after quantum\*reconciliation by\*chapter (QR2)\*DC \_\_"

P10\_EL="P10\*Prices after quantum\*reconciliation by\*chapter (QR2)\*EL \_\_"

P10\_NE="P10\*Prices after quantum\*reconciliation by\*chapter (QR2)\*NE \_\_"

P10\_Long\_Stay\_Payment="P10\*Prices after quantum\*reconciliation by\*chapter (QR2)\*Long Stay Payment\_\_"

P10\_SSEM\_Banding="P10\*Prices after quantum\*reconciliation by\*chapter (QR2)\*SSEM Banding\_\_"

P10\_SSEM\_Pct="P10\*Prices after quantum\*reconciliation by\*chapter (QR2)\*SSEM % \_"

P10\_SSEM\_Tariff="P10\*Prices after quantum\*reconciliation by\*chapter (QR2)\*SSEM Tariff\_\_"

P11\_Sub\_Chapter="P11\*Total quantum by\*subchapter after quantum\*adjustment by \*chapter (QR2)-Subchapter\_\_"

P11\_OPROC\_Quantum\_by\_HRG="P11\*Total quantum by\*subchapter after quantum\*adjustment by \*chapter (QR2)-OPROC Quantum\*by HRG\_\_"

P11\_APC\_Quantum\_by\_HRG="P11\*Total quantum by\*subchapter after quantum\*adjustment by \*chapter (QR2)-APC Quantum\*by HRG\_\_"

P11\_APC\_OPROC\_Quantum\_by\_HRG="P11\*Total quantum by\*subchapter after quantum\*adjustment by \*chapter (QR2)-APC and OPROC\*Quantum by HRG\_\_"

P11\_Sub\_\_Chapter="P11\*Total quantum by\*subchapter after quantum\*adjustment by \*chapter (QR2)-Sub\_chapter\_\_"

P11\_OPROC\_quantum\_by\_SubChapter="P11\*Total quantum by\*subchapter after quantum\*adjustment by \*chapter (QR2)-OPROC Quantum\*by subchapter\_\_"

P11\_APC\_quantum\_by\_SubChapter="P11\*Total quantum by\*subchapter after quantum\*adjustment by \*chapter (QR2)-APC Quantum\*by subchapter\_\_"

P11\_APC\_OPROC\_quantum\_by\_chapter="P11\*Total quantum by\*subchapter after quantum\*adjustment by \*chapter (QR2)-OPROC and APC \*Quantum\*by subchapter\_\_"

P112\_HRG="P112\*Prices after \*implementing SMF\*HRG \_\_"

P112\_Chapter="P112\*Prices after \*implementing SMF\*Chapter \_\_"

P112\_OPROC="P112\*Prices after \*implementing SMF\*OPROC \_\_"

P112\_DC="P112\*Prices after \*implementing SMF\*DC \_\_"

P112\_EL="P112\*Prices after \*implementing SMF\*EL \_\_"

P112\_NE ="P112\*Prices after \*implementing SMF\*NE \_\_"

P112\_Long\_Stay\_Payment="P112\*Prices after \*implementing SMF\*Long Stay Payment \_\_"

P112\_SSEM\_Banding="P112\*Prices after \*implementing SMF\*SSEM Banding \_\_"

P112\_SSEM\_Pct="P112\*Prices after \*implementing SMF\*SSEM % \_\_"

P112\_SSEM\_Tariff="P112\*Prices after \*implementing SMF\*SSEM Tariff\_\_"

P12\_Sub\_Chapter="P12\*Total quantum \*by chapter after \*implementing SMF\*Sub-chapter"

P12\_OPROC\_Quantum\_by\_HRG="P12\*Total quantum \*by chapter after \*implementing SMF\*OPROC Quantum by HRG\_\_"

P12\_APC\_Quantum\_by\_HRG="P12\*Total quantum \*by chapter after \*implementing SMF\*APC Quantum by HRG\_\_"

P12\_APC\_OPROC\_Quantum\_by\_HRG="P12\*Total quantum \*by chapter after \*implementing SMF\*OPROC and APC \*Quantum by HRG\_\_"

P12\_Sub\_\_Chapter="P12\*Total quantum \*by chapter after \*implementing SMF\*Subchapter"

P12\_OPROC\_quantum\_by\_SubChapter="P12\*Total quantum \*by chapter after \*implementing SMF\*OPROC Quantum \*by subchapter"

P12\_APC\_quantum\_by\_SubChapter="P12\*Total quantum \*by chapter after \*implementing SMF\*APC Quantum \*by subchapter"

P12\_APC\_OPROC\_quantum\_by\_chapter="P12\*Total quantum \*by chapter after \*implementing SMF\*APC and OPROC\*Quantum \*by subchapter"

P13\_Chapter="P13\*Quantum reconciliation\*by chapter (QR3)\*-Chapter \_\_"

P13\_Chapter\_Sum="P13\*Quantum reconciliation\*by chapter (QR3)\*-Chapter Sum\_\_"

P13\_APC\_OPROC\_QR3="P13\*Quantum reconciliation\*by chapter (QR3)\*-APC and OPROC QR3\_\_"

P14\_HRG="P14\*Prices after implementing QR3\*HRG\_\_"

P14\_Chapter="P14\*Prices after implementing QR3\*Chapter\_\_"

P14\_APCOPROC\_subchapter\_QR3="P14\*Prices after implementing QR3\*APC and OPROC\*subchapter QR3 factor\_\_"

P14\_OPROC="P14\*Prices after implementing QR3\*OPROC\_\_"

P14\_DC="P14\*Prices after implementing QR3\*DC\_\_"

P14\_EL="P14\*Prices after implementing QR3\*EL\_\_"

P14\_NE="P14\*Prices after implementing QR3\*NE\_\_"

P14\_Long\_Stay\_Payment="P14\*Prices after implementing QR3\*Long Stay Payment\_\_"

P14\_SSEM\_Banding="P14\*Prices after implementing QR3\*SSEM Banding\_\_"

P14\_SSEM\_Pct="P14\*Prices after implementing QR3\*SSEM % \_"

P14\_SSEM\_Tariff="P14\*Prices after implementing QR3\*SSEM Tariff\_\_"

P15\_Sub\_Chapter="P15\*Total quantum by \*chapter after \*implementing QR3\*-Subchapter\_\_"

P15\_OPROC\_Quantum\_by\_HRG="P15\*Total quantum by \*chapter after \*implementing QR3\*OPROC Quantum by HRG\_\_"

P15\_APC\_Quantum\_by\_HRG="P15\*Total quantum by \*chapter after \*implementing QR3\*APC Quantum by HRG\_\_"

P15\_APC\_OPROC\_Quantum\_by\_HRG="P15\*Total quantum by \*chapter after \*implementing QR3\*APC and OPROC\*Quantum by HRG\_\_"

P15\_Sub\_\_Chapter="P15\*Total quantum by \*chapter after \*implementing QR3\*-Sub\_chapter\_\_"

P15\_OPROC\_quantum\_by\_SubChapter="P15\*Total quantum by \*chapter after \*implementing QR3\*OPROC Quantum \*by subchapter\_\_"

P15\_APC\_quantum\_by\_SubChapter="P15\*Total quantum by \*chapter after \*implementing QR3\*APC Quantum\*by subchapter\_\_"

P15\_APC\_OPROC\_quantum\_by\_chapter="P15\*Total quantum by \*chapter after \*implementing QR3\*APC and OPROC\*Quantum\*by subchapter\_\_"

P16\_HRG="P16\*Prices after Scaling\*HRG\_\_"

P16\_Chapter="P16\*Prices after Scaling\*Chapter\_\_"

P16\_OPROC="P16\*Prices after Scaling\*OPROC\_\_"

P16\_DC="P16\*Prices after Scaling\*DC\_\_"

P16\_EL="P16\*Prices after Scaling\*EL\_\_"

P16\_NE="P16\*Prices after Scaling\*NE\_\_"

P16\_Long\_Stay\_Payment="P16\*Prices after Scaling\*Long Stay Payment\_\_"

P16\_SSEM\_Banding="P16\*Prices after Scaling\*SSEM Banding\_\_"

P16\_SSEM\_Pct="P16\*Prices after Scaling\*SSEM % \_"

P16\_SSEM\_Tariff="P16\*Prices after Scaling\*SSEM Tariff \_"

P17\_Sub\_Chapter="P17\*Total quantum by chapter\*after implementing scaling\*Subchapter\_\_"

P17\_OPROC\_Quantum\_by\_HRG="P17\*Total quantum by chapter\*after implementing scaling\*OPROC Quantum by HRG\_"

P17\_APC\_Quantum\_by\_HRG="P17\*Total quantum by chapter\*after implementing scaling\*APC Quantum by HRG\_"

P17\_APC\_OPROC\_Quantum\_by\_HRG="P17\*Total quantum by chapter\*after implementing scaling\*APC and OPROC \*Quantum by HRG\_"

P17\_Sub\_\_Chapter="P17\*Total quantum by chapter\*after implementing scaling\*Sub-Chapter\_"

P17\_OPROC\_quantum\_by\_SubChapter="P17\*Total quantum by chapter\*after implementing scaling\*OPROC Quantum \*by subchapter\_"

P17\_APC\_quantum\_by\_SubChapter="P17\*Total quantum by chapter\*after implementing scaling\*APC Quantum \*by subchapter\_"

P17\_APC\_OPROC\_quantum\_by\_chapter="P17\*Total quantum by chapter\*after implementing scaling\*APC and OPROC Quantum \*by subchapter\_"

P17\_APC\_Inlier\_Quantum="P17\*Total quantum by chapter\*after implementing scaling\*APC and OPROC Quantum \*APC Inlier Quantum\_"

P172\_Sub\_Chapter="P172\*For CNST (Uplifts \*excluding CNST applied)\*Subchapter\_\_"

P172\_OPROC\_QuantSubchap\_eCNST="P172\*For CNST (Uplifts \*excluding CNST applied)\*OPROC Quantum by subchapter\*(Uplifted to 17/18, \*excluding CNST)\_\_"

P172\_APC\_QuantSubchap\_inlier="P172\*For CNST (Uplifts \*excluding CNST applied)\*APC Quantum by subchapter\*(Inlier Only - uplifted \*to 17/18, \*excluding CNST)\_\_"

P172\_APC\_OPROC\_QuanSubchap\_Inl="P172\*For CNST (Uplifts \*excluding CNST applied)\*APC and OPROC \*Quantum by subchapter\*(Inlier only - uplifted \*to 17/18, \*excluding CNST)\_\_"

P18\_HRG="P18\*Final prices \*after all uplifts\*HRG\_\_"

P18\_Chapter="P18\*Final prices \*after all uplifts\*Chapter\_\_"

P18\_OPROC="P18\*Final prices \*after all uplifts\*OPROC\_\_"

P18\_DC="P18\*Final prices \*after all uplifts\*DC\_\_"

P18\_EL="P18\*Final prices \*after all uplifts\*EL\_\_"

P18\_NE="P18\*Final prices \*after all uplifts\*NE\_\_"

P18\_Long\_Stay\_Payment="P18\*Final prices \*after all uplifts\*Long Stay Payment\_\_"

P18\_SSEM\_Banding="P18\*Final prices \*after all uplifts\*SSEM Banding\_\_"

P18\_SSEM\_Pct="P18\*Final prices \*after all uplifts\*SSEM % \_"

P18\_SSEM\_Tariff="P18\*Final prices \*after all uplifts\*SSEM Tariff \_"

P19\_Sub\_Chapter="P19\*Final total quantum \*by chapter after implementing\*all uplifts\*-SubChapter \_\_"

P19\_OPROC\_Quantum\_by\_HRG="P19\*Final total quantum \*by chapter after implementing\*all uplifts\*OPROC Quantum by HRG"

P19\_APC\_Quantum\_by\_HRG="P19\*Final total quantum \*by chapter after implementing\*all uplifts\*APC Quantum by HRG"

P19\_APC\_OPROC\_Quantum\_by\_HRG="P19\*Final total quantum \*by chapter after implementing\*all uplifts\*APC and OPROC\*Quantum by HRG"

P19\_Sub\_\_Chapter="P19\*Final total quantum \*by chapter after implementing\*all uplifts\*-Sub\_Chapter \_\_"

P19\_OPROC\_quantum\_by\_SubChapter="P19\*Final total quantum \*by chapter after implementing\*all uplifts\*OPROC Quantum \*by subchapter \_\_"

P19\_APC\_quantum\_by\_SubChapter="P19\*Final total quantum \*by chapter after implementing\*all uplifts\*APC Quantum \*by subchapter \_\_"

P19\_APC\_OPROC\_quantum\_by\_chapter="P19\*Final total quantum \*by chapter after implementing\*all uplifts\*APC and OPROC \*Quantum \*by subchapter \_\_"

;

**run**;

/\*Step 3--QUANTUM RECONCILIATION BY Sub Chapter----------------------- End here \*/

/\*Step 4-- STEP BY STEP OP-DC-EL-NE Start from here \*/

/\*Create and format STEP BY STEP OP-DC-EL-NE tab\*/

ods tagsets.ExcelXP options(sheet\_name='STEP BY STEP OP-DC-EL-NE'

suppress\_bylines='yes'

frozen\_headers='1'

FROZEN\_ROWHEADERS='6'

autofit\_height='Yes'

absolute\_column\_width='6, 6, 7, 50, 8, 8, 8, 7, 8, 7, 7, 10, 10, 10,10, 10, 10, 10, 10, 10, 10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,14,14,14,14'

pages\_fitwidth='1'

pages\_fitheight='9'

FitToPage="yes"

autofilter='1-70');

/\*Define STEP BY STEP OP-DC-EL-NE tab output variables\*/

**proc** **print** data=Modelout.STEP\_BY\_STEP\_OP\_DC\_EL\_NE split='\*' n noobs;

var Spell\_HRG SpellHRGChapter SpellHRGSubchapter HRG\_Name OPROC\_currency\_applicable

/ style(Column)=[background=#e6ffff tagattr='format:#,##'] style(header)=[background=#e6ffff];

var P0\_OPROC P0\_DC P0\_EL P0\_NE / style(Column)=[background=#e6e6ff ] style(header)=[background=#e6e6ff];

var P1\_OPROC P1\_DC P1\_EL P1\_NE / style(Column)=[background=#c2c2a3 tagattr="format:###,###,###.##"] style(header)=[background=#c2c2a3];

var QR1/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00] ;

var P3\_OPROC P3\_DC P3\_EL P3\_NE / style(Column)=[background=#a5bcd9 tagattr="format:###,###,###.##"] style(header)=[background=#a5bcd9];

var Cost\_base\_adjustment/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00];

var P5\_OPROC P5\_DC P5\_EL P5\_NE / style(Column)=[background=#d1e2b2 tagattr="format:###,###,###.##"] style(header)=[background=#d1e2b2];

var P6\_OPROC P6\_DC P6\_EL P6\_NE / style(Column)=[background=#ffe6e6 tagattr="format:###,###,###.##"] style(header)=[background=#ffe6e6];

var P7\_OPROC P7\_DC P7\_EL P7\_NE / style(Column)=[background=#b2d7e2 tagattr="format:###,###,###.##"] style(header)=[background=#b2d7e2];

var P8\_OPROC P8\_DC P8\_EL P8\_NE / style(Column)=[background=#c0b2e2 tagattr="format:###,###,###.##"] style(header)=[background=#c0b2e2];

var P9\_QR2/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00];

var P10\_OPROC P10\_DC P10\_EL P10\_NE / style(Column)=[background=#AFEEEE tagattr="format:###,###,###.##"] style(header)=[background=#AFEEEE ];

var SMF/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%" ] style(header)=[background=#FFFF00];

var P12\_OPROC P12\_DC P12\_EL P12\_NE / style(Column)=[background=#E9967A tagattr="format:###,###,###.##"] style(header)=[background=#E9967A];

var P13\_QR3/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%" ] style(header)=[background=#FFFF00];

var P14\_OPROC P14\_DC P14\_EL P14\_NE / style(Column)=[background=#87CEFA tagattr="format:###,###,###.##"] style(header)=[background=#87CEFA];

var P15\_SCF / style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00];

var P16\_OPROC P16\_DC P16\_EL P16\_NE / style(Column)=[background=#c2c2a4 tagattr="format:###,###,###.##"] style(header)=[background=#c2c2a4];

var P17\_Inflation /style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00];

var P18\_OPROC P18\_DC P18\_EL P18\_NE / style(Column)=[background=#ffe6e9 tagattr="format:###,###,###.##" ] style(header)=[background=#ffe6e9];

var P19\_Efficiency/style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00];

var P20\_OPROC P20\_DC P20\_EL P20\_NE / style(Column)=[background=#b8dc6f tagattr="format:###,###,###.##" ] style(header)=[background=#b8dc6f];

var P21\_19\_20\_CNST/ style(Column)=[background=#FFFF00 tagattr="format:#,##0.00%"] style(header)=[background=#FFFF00];

var P22\_OPROC P22\_DC P22\_EL P22\_NE / style(Column)=[background=#ccffff tagattr="format:###,###,###.##" ] style(header)=[background=#ccffff];

label

Spell\_HRG="Spell\*HRG\_\_"

SpellHRGChapter ="Spell HRG\*chapter\_\_"

SpellHRGSubchapter="Spell HRG\*sub - \*chapter\_\_"

HRG\_Name="HRG\*name"

OPROC\_currency\_applicable="OPROC\*currency\*applicable\_\_"

P0\_OPROC="P0\*Exception flag:\*1 = HRG manual adj, 0 = no manual adj \*OPROC Tariff\_\_"

P0\_DC = "P0\*Exception flag:\*1 = HRG manual adj, 0 = no manual adj \*DC Tariff\_\_"

P0\_EL = "P0\*Exception flag:\*1 = HRG manual adj, 0 = no manual adj \*EL Tariff\_\_"

P0\_NE= "P0\*Exception flag:\*1 = HRG manual adj, 0 = no manual adj \*NE Tariff\_\_"

P1\_OPROC="P1\*SQL\_Out\*OPROC Tariff\_\_"

P1\_DC = "P1\*SQL\_Output\*DC Tariff\_\_"

P1\_EL = "P1\*SQL\_Output\*EL Tariff\_\_"

P1\_NE= "P1\*SQL\_Output\*NE Tariff\_\_"

QR1="P2\*Reference costs quantum reconciliation factor (QR1)\_\_"

P3\_OPROC="P3\*Modelled prices after QR1 factor\*OPROC Tariff\_\_"

P3\_DC = "P3\*Modelled prices after QR1 factor\*DC Tariff\_\_"

P3\_EL = "P3\*Modelled prices after QR1 factor\*EL Tariff\_\_"

P3\_NE= "P3\*Modelled prices after QR1 factor\*NE Tariff\_\_"

Cost\_base\_adjustment="P4\*Cost base adjustment (CB)\_\_"

P5\_OPROC="P5\*Prices after CB\*implementation\*OPROC Tariff\_\_"

P5\_DC = "P5\*Prices after CB\*implementation\*DC Tariff\_\_"

P5\_EL = "P5\*Prices after CB\*implementation\*EL Tariff\_\_"

P5\_NE = "P5\*Prices after CB\*implementation\*NE Tariff\_\_"

P6\_OPROC="P6\*Prices after inflation, \*efficiency and CNST adjustments \*to a 18/19 financial year basis\*OPROC Tariff\_\_"

P6\_DC = "P6\*Prices after inflation, \*efficiency and CNST adjustments \*to a 18/19 financial year basis\*DC Tariff\_\_"

P6\_EL = "P6\*Prices after inflation, \*efficiency and CNST adjustments \*to a 18/19 financial year basis\*EL Tariff\_\_"

P6\_NE = "P6\*Prices after inflation, \*efficiency and CNST adjustments \*to a 18/19 financial year basis\*NE Tariff\_\_"

P7\_OPROC="P7\*Manual Adjustments\*OPROC Tariff\_\_"

P7\_DC = "P7\*Manual Adjustments\*DC Tariff\_\_"

P7\_EL = "P7\*Manual Adjustments\*EL Tariff\_\_"

P7\_NE = "P7\*Manual Adjustments\*NE Tariff\_\_"

P8\_OPROC="P8\*Prices before QR2 implementation\*OPROC Tariff\_\_"

P8\_DC = "P8\*Prices before QR2 implementation\*DC Tariff\_\_"

P8\_EL = "P8\*Prices before QR2 implementation\*EL Tariff\_\_"

P8\_NE = "P8\*Prices before QR2 implementation\*NE Tariff\_\_"

P9\_QR2="P9\*Manual adjustments quantum reconciliation factor (QR2)\_\_"

P10\_OPROC="P10\*Prices after QR2 implementation\*OPROC Tariff\_\_"

P10\_DC = "P10\*Prices after QR2 implementation\*DC Tariff\_\_"

P10\_EL = "P10\*Prices after QR2 implementation\*EL Tariff\_\_"

P10\_NE = "P10\*Prices after QR2 implementation\*NE Tariff\_\_"

SMF="P11\*Smoothing Factor adj (SMF)\_\_"

P12\_OPROC="P12\*Prices after SMF adj\*OPROC Tariff\_\_"

P12\_DC = "P12\*Prices after SMF adj\*DC Tariff\_\_"

P12\_EL = "P12\*Prices after SMF adj\*EL Tariff\_\_"

P12\_NE = "P12\*Prices after SMF adj\*NE Tariff\_\_"

P13\_QR3="P13\*Final quantum reconciled prices (QR3)\_\_"

P14\_OPROC="P14\*Prices after QR3 implementation\*OPROC Tariff\_\_"

P14\_DC = "P14\*Prices after QR3 implementation\*DC Tariff\_\_"

P14\_EL = "P14\*Prices after QR3 implementation\*EL Tariff\_\_"

P14\_NE = "P14\*Prices after QR3 implementation\*NE Tariff\_\_"

P15\_SCF="P15\*Scaling Factor Adj (SCF)\_\_"

P16\_OPROC="P16\*Final prices before 18/19 cost uplift assumptions\*OPROC Tariff\_\_"

P16\_DC = "P16\*Final prices before 18/19 cost uplift assumptions\*DC Tariff\_\_"

P16\_EL = "P16\*Final prices before 18/19 cost uplift assumptions\*EL Tariff\_\_"

P16\_NE = "P16\*Final prices before 18/19 cost uplift assumptions\*NE Tariff\_\_"

P17\_Inflation="P17\*17/18 Inflation\_\_"

P18\_OPROC="P18\*Final Prices in 19/20 prices including inflation\*OPROC Tariff\_\_"

P18\_DC = "P18\*Final Prices in 19/20 prices including inflation\*DC Tariff\_\_"

P18\_EL = "P18\*Final Prices in 19/20 prices including inflation\*EL Tariff\_\_"

P18\_NE = "P18\*Final Prices in 19/20 prices including inflation\*NE Tariff\_\_"

P19\_Efficiency="P19\*19/20 Efficiency Factor\_\_"

P20\_OPROC="P20\*Final Prices in 19/20 prices, including inflation and efficiency assumptions\*OPROC Tariff\_\_"

P20\_DC = "P20\*Final Prices in 19/20 prices, including inflation and efficiency assumptions\*DC Tariff\_\_"

P20\_EL = "P20\*Final Prices in 19/20 prices, including inflation and efficiency assumptions\*EL Tariff\_\_"

P20\_NE = "P20\*Final Prices in 19/20 prices, including inflation and efficiency assumptions\*NE Tariff\_\_"

P21\_19\_20\_CNST="P21\*19/20 CNST\_\_"

P22\_OPROC="P22\*FINAL PRICES in 19/20 prices, including inflation, efficiency and CNST assumptions\*OPROC Tariff\_\_"

P22\_DC = "P22\*FINAL PRICES in 19/20 prices, including inflation, efficiency and CNST assumptions\*DC Tariff\_\_"

P22\_EL = "P22\*FINAL PRICES in 19/20 prices, including inflation, efficiency and CNST assumptions\*EL Tariff\_\_"

P22\_NE = "P22\*FINAL PRICES in 19/20 prices, including inflation, efficiency and CNST assumptions\*NE Tariff\_\_"

;

**run**;

/\*Step 4-- STEP BY STEP OP-DC-EL-NE End here \*/

/\*Step 5-- APC\_OPROC\_Model\_Linked\_Sheet Start from here \*/

ods tagsets.ExcelXP options(sheet\_name='Linked Sheet'

autofilter='NO' embedded\_titles ="yes" autofit\_height ="yes" frozen\_headers="yes" frozen\_headers="2"

FROZEN\_ROWHEADERS='3'

absolute\_column\_width='6,8,50,10,10,10,10,10,10,10,10,10,10,10');

**proc** **print** data = ModelOUt.APC\_OPROC\_MODEL\_LINKEDSHEET noobs label split= '\*';

var Year HRG\_Code HRG\_Name /style(header)=[background=#b3ffff];

var OPROC\_Tariff Combined\_DC\_EL\_Prices DC\_Prices EL\_Prices /style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###.##"];

var EL\_TRIM\_POINTS /style(header)=[background=#b3ffff];

var NE\_Prices /style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###.##"];

var NE\_TRIM\_POINTS /style(header)=[background=#b3ffff];

var EBD\_Prices /style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###.##"];

var SSEM\_applicable /style(header)=[background=#b3ffff];

var SSEM\_Pct /style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:#,##0.00%"];

var SSEM\_Price /style(header)=[background=#b3ffff] style(Column)=[ tagattr="format:###,###,###.##"];

label

HRG\_Code="HRG Code\_\_"

OPROC\_Tariff="Outpatient \*procedure \*tariff (£)\_\_"

Combined\_DC\_EL\_Prices="Combined day \*case /ordinary \*elective spell \*tariff (£)\_"

DC\_Prices="Day case \*spell tariff (£)\_\_"

EL\_Prices="Ordinary \*elective \*spell tariff (£)\_\_"

EL\_TRIM\_POINTS="Ordinary \*elective long \*stay trim point \*(days)\_\_"

NE\_Prices="Non-elective \*spell tariff (£)\_\_"

NE\_TRIM\_POINTS="Non-elective \*long stay \*trim point \*(days)\_\_"

EBD\_Prices="Per day long\* stay payment \*(for days exceeding \*trim point) (£)\_\_"

SSEM\_applicable="Reduced short \*stay \*emergency \*tariff \*applicable?\_\_"

SSEM\_Pct="% applied in \*calculation of \*reduced short \*stay \*emergency \*tariff\_\_"

SSEM\_Price="Reduced short \*stay emergency \*tariff (£)\_\_"

;

**run**;

/\*Step 5-- APC\_OPROC\_Model\_Link\_Sheet End here \*/

**quit**;

ods tagsets.ExcelXP close;

ods listing;

**data** P\_Finihsed;

Congratulations="Excel report is Ready";

Location="\\irnarch\sas\_data\PID\_Proj\_1718\APC\APC\_FY1920\Excel\_outputs\ ";

Filename="SASOUT 17-18 APC\_OPROC model.xml";

**run**;

**proc** **sql**;

Title "Congratulations ! Excel report is ready Now!! ";

Title1 "Excel report location:\\irnarch\sas\_data\PID\_Proj\_1718\APC\APC\_FY1920\Excel\_outputs\ ";

Title1 "File Name:SASOUT 17-18 APC\_OPROC model.xml.";

Title "Have a nice day!!";

Title1;

Title;

select \* from P\_Finihsed

;

**quit**;

/\*Step 2.10\_1 Project\_Report \*/

/\*--------------------------------------------------------------\*/

\* Step2\_10\_1\_Project\_Report \*/

/\* Version: 1.01 \*/

/\* Coding: PID \*/

/\* Date: August 2015 \*/

/\* Updated: Feb 2016 \*/

/\* \*/

/\*--------------------------------------------------------------\*/

/\*Calculate trim points \*/

**proc** **sql** noprint;

create table APC02\_05b\_4Trans\_TP as

select distinct

SPELL\_HRG,

ADM,

TP

from

&StageLib.**.A**PC02\_05b

where ADM<>"DCRA"

order by SPELL\_HRG,ADM

;

**quit**;

/\* \*--------> diffrent method used (proc transpose ) vs pivot in SQL server \*/

**proc** **transpose** data=APC02\_05b\_4Trans\_TP out=HRG\_TrimPoints (drop=\_name\_);

by SPELL\_HRG;

id ADM;

**run**;

**proc** **sql** noprint;

create table &StageLib.**.H**RG\_Trimpoints

as

select

SPELL\_HRG,

EL as EL\_Trimpoints,

NE as NE\_Trimpoints

from

HRG\_Trimpoints

order by SPELL\_HRG

;

**quit**;

/\*Calculate EBDDs \*/

**proc** **sql** noprint;

create table APC02\_05C\_4Trans\_EBDS as

select distinct

SPELL\_HRG,

ADM,

FCE\_EBD\_UC\_CAPPED

from

&StageLib.**.A**PC02\_05C

where ADM<>"DCRA"

order by SPELL\_HRG,ADM

;

**quit**;

/\* \*--------> diffrent method used (proc transpose ) vs pivot in SQL server \*/

**proc** **transpose** data=APC02\_05C\_4Trans\_EBDS out=HRG\_EBDS (drop=\_name\_);

by SPELL\_HRG;

id ADM;

**run**;

**proc** **sql** noprint;

create table &StageLib.**.H**RG\_EBDS

as

select

SPELL\_HRG,

NE as NE\_EBDs format **8.2**

from

HRG\_EBDS

order by SPELL\_HRG

;

**quit**;

**proc** **sql** noprint;

create table &StageLib.**.P**rj\_Report\_APC\_Tariff (drop= EL\_ACT) as

select distinct

a.SPELL\_HRG as HRG\_Code label "HRG\_Code",

b.Description as HRG\_Name,/\* Added 20160302 by PID \*/

a.EL\_ACT label "Combined day case / ordinary elective spell tariff (£)",

a.EL\_UC as Ordinary\_Elective\_Spell\_Tariff format **8.2**,

a.NE\_UC as None\_Elective\_Spell\_Tariff format **8.2**,

c.EL\_Trimpoints,

c.NE\_Trimpoints,

d.NE\_EBDs

from &StageLib.**.A**PC\_UNIT\_COSTS as a

inner join Tariff\_R.HRG\_ELIGIBILITY as b

on a.SPELL\_HRG=b.HRG /\* Added 20160302 by PID \*/

inner join &StageLib.**.H**RG\_Trimpoints as c

on a.SPELL\_HRG=c.SPELL\_HRG

inner join &StageLib.**.H**RG\_EBDs as d

on a.SPELL\_HRG=d.SPELL\_HRG

where b.Mandatory\_Price\_anywhere\_in\_NT\_=**1** and b.Year="&Year."

order by a.SPELL\_HRG

;

**quit**;

%let StepNo=Step2\_10\_1;

%let id=DC;

%let outputfile1=Prj\_Report\_APC\_Tariff;

**proc** **export** data=&StageLib.**.**&outputfile1.

outfile="&outputpath.\&outputfile1..csv"

dbms=csv

replace;

**run**;

FILENAME DelFile1 "&outputpath2.\DC\_NO\Prj\_Report\_APC.xls"; /\* PID in \*/

**DATA** \_NULL\_ ;

rc = FDELETE('DelFile1');

**RUN** ;

FILENAME DelFile1 CLEAR ;

**data** Prj\_Report\_APC ;

set &StageLib.**.P**rj\_Report\_APC\_Tariff;

label

HRG\_Code= "HRG code"

HRG\_Name= "HRG name"

Ordinary\_Elective\_Spell\_Tariff= "Ordinary^ elective spell^ tariff (£)"

None\_Elective\_Spell\_Tariff ="Non-elective^ spell tariff (£)"

EL\_Trimpoints= "Ordinary^ elective long^ stay^ trimpoint^ (days)"

NE\_Trimpoints= "Non-elective^ long stay^ trimpoint^ (days)"

NE\_EBDs= "Per day^ long stay payment^ (for days ^exceeding ^trimpoint) ^(£)"

;

**run**;

**proc** **template**;

define style styles.MysansPrinter;

parent = styles.sansPrinter;

style myheader from header /

background = #CEF6EC

font\_size = **9**pt

just = Center;

end;

**run**;

**quit**;

ods listing close;

ods tagsets.ExcelXP path="&outputpath2.\DC\_NO" file="Prj\_Report\_APC.xls" style=MysansPrinter

options (autofilter='NO' embedded\_titles ="yes" autofit\_height ="yes" frozen\_headers="yes" frozen\_headers="2");

ods tagsets.ExcelXP options(sheet\_name='APC\_Tariff' absolute\_column\_width='12,14');

**proc** **print** data = Prj\_Report\_APC noobs label split= '^';

var HRG\_Code

HRG\_Name

Ordinary\_Elective\_Spell\_Tariff

None\_Elective\_Spell\_Tariff

EL\_Trimpoints

NE\_Trimpoints

NE\_EBDs /style(Header)=myheader

;

**run**;

ods tagsets.ExcelXP close;

ods listing;

/\*part 2 output all results from all steps \*/

FILENAME DelFile "&outputpath2.\DC\_NO\APC\_Tariff.xlsx"; /\* PID in \*/

**DATA** \_NULL\_ ;

rc = FDELETE('DelFile');

**RUN** ;

FILENAME DelFile CLEAR ;

libname excelf2 XLSX "&outputpath2.\DC\_NO\APC\_Tariff.xlsx"; /\* PID in \*/

**data** excelf2.Rpt\_2\_1\_APC\_ScopeData\_UnitCost;

set &StageLib.**.R**pt\_2\_1\_APC\_ScopeData\_UnitCost;

**run**;

**data** excelf2.Rpt\_2\_3\_APC01\_03a\_UnitCost;

set &StageLib.**.R**pt\_2\_3\_APC01\_03a\_UnitCost;

**run**;

/\*Step2\_4 outputs \*/

**data** excelf2.APC\_HRGLevelData;

set &StageLib.**.A**PC\_HRGLevelData;

**run**;

/\*Step2\_5 outputs \*/

**data** excelf2.APC\_AandE\_Adm;

set &StageLib.**.A**PC\_AandE\_Adm;

**run**;

/\*Step2\_6 outputs \*/

**data** excelf2.APC\_NICE;

set &StageLib.**.A**PC\_NICE;

**run**;

/\*Step2\_7 outputs \*/

**data** excelf2.APC\_SpellData;

set &StageLib.**.A**PC\_SpellData;

**run**;

/\*Step2\_8 outputs \*/

**data** excelf2.APC\_EBD;

set &StageLib.**.A**PC\_EBD;

**run**;

**data** excelf2.APC\_EBD\_UC;

set &StageLib.**.A**PC\_EBD\_UC;

**run**;

/\*Step2\_9 outputs \*/

**data** excelf2.APC\_Act\_Costs;

set &StageLib.**.A**PC\_Act\_Costs;

**run**;

**data** excelf2.APC\_DandD\_topslice;

set &StageLib.**.A**PC\_DandD\_topslice;

**run**;

/\*Step2\_10 outputs \*/

**data** excelf2.APC\_Unit\_Costs;

set &StageLib.**.A**PC\_Unit\_Costs;

**run**;

/\*close the excel file \*/

libname excelf2 CLEAR;

/\*Step 2.10 Calculate unit costs \*/

/\*--------------------------------------------------------------\*/

\* Step2\_10\_Calculate\_Unit\_Costs \*/

/\* Version: 1.01 \*/

/\* Coding: PID \*/

/\* Date: August 2015 \*/

/\* -------------------------------------------------------------\*/

/\* Input: \*/

/\* (1)Stage.APC\_Act\_Costs (from Step2\_9) \*/

/\* \*/

/\* Output: \*/

/\* (1) Stage.Stage.APC\_Unit\_Costs \*/

/\* \*/

/\* \*/

/\*--------------------------------------------------------------\*/

/\* \*/

/\* Output variables from Stage.APC\_Unit\_Costs: \*/

/\* (1)SPELL\_HRG \*/

/\* (2)EL\_ACT \*/

/\* (3)NE\_ACT \*/

/\* (4)EL\_TC \*/

/\* (5)NE\_TC \*/

/\* (6)EL\_UC \*/

/\* (7)NE\_UC \*/

/\* \*/

/\*--------------------------------------------------------------\*/

\*Note: param\_HESRunID moved to initialise parameters;

\*%let param\_HESRunID=10;

**proc** **sql** noprint;

create table APC\_APC03\_01A\_4Trans as

select distinct

SPELL\_HRG,

ADM,

SPELL\_ACT

from &StageLib.**.A**PC\_Act\_Costs

where ADM in ("EL","NE")

order by SPELL\_HRG,ADM

;

**quit**;

/\* \*--------> diffrent method used (proc transpose ) vs pivot in SQL server \*/

**proc** **transpose** data=APC\_APC03\_01A\_4Trans out=&StageLib.**.A**PC03\_01a (drop=\_name\_);

by SPELL\_HRG;

id ADM;

**run**;

/\*--Query: 01b: Costs

--Admission crosstab of total

\*/

**proc** **sql** noprint;

create table APC03\_01b\_4Trans as

select distinct

SPELL\_HRG,

ADM,

REVISED\_TC

from &StageLib.**.A**PC\_Act\_Costs

where ADM in ("EL","NE")

order by SPELL\_HRG,ADM

;

**quit**;

/\* \*--------> diffrent method used (proc transpose ) vs pivot in SQL server \*/

**proc** **transpose** data=APC03\_01b\_4Trans out=&StageLib.**.A**PC03\_01b (drop=\_name\_);

by SPELL\_HRG;

id ADM;

**run**;

/\*--Query: 01c: Unit Costs

--Calculates Unit Cost

\*/

**proc** **sql** noprint;

Create table &StageLib.**.A**PC\_Unit\_Costs as

SELECT

A.\*,

EL\_TC/EL\_ACT AS EL\_UC,

NE\_TC/NE\_ACT AS NE\_UC

FROM

(

SELECT

T01a.SPELL\_HRG,

T01a.EL AS EL\_ACT,

T01a.NE AS NE\_ACT,

T01b.EL AS EL\_TC,

T01b.NE AS NE\_TC

FROM

&StageLib.**.A**PC03\_01a AS T01a

INNER JOIN &StageLib.**.A**PC03\_01b AS T01b

ON T01a.SPELL\_HRG = T01b.SPELL\_HRG) AS A

order by SPELL\_HRG

;

**quit**;

/\*output final dataset to csv file \*/

%let StepNo=Step2\_10;

%let outputfile1=APC\_Unit\_Costs;

**proc** **export** data=&StageLib.**.**&outputfile1.

outfile="&outputpath.\&StepNo.\_&outputfile1..csv"

dbms=csv

replace;

**run**;

/\*Step 2.11 Short Stay Energency Tariff(SSEM) adjustment \*/

/\*--------------------------------------------------------------\*/

/\* Program Name: Step2\_11\_Short\_Stay\_Emergency\_Tarrif\_Adjust \*/

/\* Version: 1.01 \*/

/\* Coding: PID \*/

/\* Date: October 2015 \*/

/\* Code updated : PID August 2017-- for [Stage].APC03\_02b \*/

/\* -------------------------------------------------------------\*/

/\* Input: \*/

/\* (1)Stage.APC\_Act\_Costs (from Step2\_9) \*/

/\* (2 ) Tariff\_R.HRGSELIGIBILITY\_TEMP PID 20171218 \*/

/\* Output: \*/

/\* (1) Stage.Stage.APC\_Unit\_Costs \*/

/\* \*/

/\* \*/

/\*--------------------------------------------------------------\*/

/\* \*/

/\* Output variables from Stage.APC\_Unit\_Costs: \*/

/\* (1)SPELL\_HRG \*/

/\* (2)EL\_ACT \*/

/\* (3)NE\_ACT \*/

/\* (4)EL\_TC \*/

/\* (5)NE\_TC \*/

/\* (6)EL\_UC \*/

/\* (7)NE\_UC \*/

/\* \*/

/\*--------------------------------------------------------------\*/

/\*

--02 SSEM

--Query: 02a: Apply SSEM banding to HRGs

--Short Stay bandings for mandatory HRGs

\*/

**proc** **sql** noprint;

create table &StageLib.**.A**PC03\_02a as

SELECT distinct

in\_01b.HRG\_Code AS HRG,

in\_01b.Eligible\_for\_SSEM AS ELIGIBLE,

in\_07.Band AS BAND,

in\_07.ALoS,

in\_07.TARIFF\_Pct,

in\_01b.Mandatory\_IP AS MAND\_IP

FROM

Tariff\_R.HRGSELIGIBILITY\_TEMP AS in\_01b

INNER JOIN Tariff\_R.SSEM\_Banding as in\_07

ON in\_01b.SSEM\_Banding = in\_07.Band

WHERE

(((in\_01b.Mandatory\_IP)=**1** and in\_07.Year="&Year." and in\_01b.Year="&Year." AND SUBSTR(in\_01b.HRG\_CODE,**1**,**2**)^="NZ"))

ORDER BY

in\_01b.HRG\_Code

;

**quit**;

/\* -----------------------------------Note : YC - Nov 2017 \*/

/\*Need update when there are values in SSEM\_banding table in Tariff\_R.HRG\_Eligibility \*/

/\*

proc sql noprint;

create table &StageLib..APC03\_02a as

SELECT

in\_01b.HRG,

in\_01b.SSEM\_Eligible AS ELIGIBLE,

in\_07.Band AS BAND,

in\_07.ALoS,

in\_07.TARIFF\_Pct,

in\_01b.Mandatory\_Price\_anywhere\_in\_NT\_ AS MAND\_IP

FROM

\*/

/\*&SQLInputLib..HRGsEligibility AS in\_01b \*/

/\*

Tariff\_R.HRG\_Eligibility AS in\_01b

\*/

/\*INNER JOIN &SQLInputLib..SSEM\_Banding as in\_07\*/

/\*

INNER JOIN Tariff\_R.SSEM\_Banding as in\_07

ON in\_01b.SSEM\_Banding = in\_07.Band

WHERE \*/

/\*(((in\_01b.Mandatory\_IP)=1)) \*/ /\* YC - Nov 2017\*/

/\*

in\_01b.Mandatory\_Price\_anywhere\_in\_NT\_=1 and in\_01b.Year="&Year." and in\_01b.HRG\_MAP\_ID=1

ORDER BY

in\_01b.HRG

;

quit;

\*/

/\* Query 02b: Apply SSEM reductions

Uplifts non-elective unit costs to take account of short stay emergency structure

\*/

**proc** **sql** noprint;

create table &StageLib.**.A**PC03\_02b

as select F.\*,

CASE

WHEN ( F.eligible=**1** And F.band>**1**) THEN (SSEM\_SPELL \*TARIFF\_Pct\*ADJ\_NE\_UC)

END as SSEM\_Q

from

(select

E.\*,

**0** AS TC\_CHECK

from

(

select

D.\*,

CASE WHEN NE\_TC = **.** THEN **.**

ELSE D.ADJ\_NE\_UC\*D.NE\_ACT

END as ADJ\_NE\_TC

from

(

select

C.\*,

CASE WHEN C.ELIGIBLE = **1** THEN COALESCE(C.NE\_UC/C.SSEM\_ADJ,**0**,**.**)

ELSE C.NE\_UC

END as ADJ\_NE\_UC

from (

SELECT

B.\*,

Case

when B.SSEM\_PCT^=**.** then B.SSEM\_PCT \* B.TARIFF\_Pct +(**1**-B.SSEM\_PCT)

Else **.**

End AS SSEM\_ADJ

from (

Select

A.\*,

CASE WHEN NE\_ACT = **.** THEN **.**

ELSE COALESCE(SSEM\_SPELL/ NE\_ACT ,**0**)

END as SSEM\_PCT

FROM (

SELECT

uc.SPELL\_HRG AS HRG,

T02a.ELIGIBLE,

T02a.TARIFF\_Pct,

coalesce(NE\_UC,**0**,**.**) AS RAW\_NE\_UC,

in\_05.SSEM AS SSEM\_SPELL,

uc.NE\_ACT,

uc.NE\_UC,

uc.NE\_TC,

T02a.band

FROM

&StageLib.**.A**PC\_Unit\_Costs as uc

LEFT JOIN &SQLViewLib.**.S**SEM\_Count(where=(Run\_ID = &param\_HESRunID.)) as in\_05

ON (uc.SPELL\_HRG = in\_05.SPELL\_HRG )

LEFT JOIN &StageLib.**.A**PC03\_02a AS T02a

ON uc.SPELL\_HRG = T02a.HRG

) AS A

)AS B

)AS C

)AS D

) AS E

)AS F

ORDER BY HRG

;

**quit**;

/\*

--Query 02b: Apply SSEM reductions

--Uplifts non-elective unit costs to take account of short stay emergency structure

/\*output final dataset to csv file \*/

%let StepNo=Step2\_11;

/\*%let outputfile1=&StageLib..SSEM\_Tariff\_Adjust;\*/

%let outputfile1=APC03\_02b;

**proc** **export** data=&StageLib.**.**&outputfile1.

outfile="&outputpath.\&StepNo.\_SSEM\_Tariff\_Adjustment.csv"

dbms=csv

replace;

**run**;

/\*Step 2.12 Inclusion of Clinical Negligence Scheme for Trusts(CNST) \*/

/\*--------------------------------------------------------------\*/

# \* 2.12 Inclusion of Clinical Negligence Scheme for Trusts(CNST) \*/

/\* Version: 1.01 \*/

/\* Coding: PID \*/

/\* Date: October 2015 \*/

/\* -------------------------------------------------------------\*/

/\* Input: \*/

/\* (1)Stage.APC03\_02b (from Step2\_11) \*/

/\* (2)Stage.Stage.APC\_Unit\_Costs \*/

/\* (3)Input.APC\_CNST\_SubChapter \*/

/\* (4)Input.APC\_CNST\_Chapter \*/

/\* Output: \*/

/\* (1) Stage.APC\_Post\_CNST \*/

/\* (2) Stage.APC03\_03e \*/

/\* \*/

/\*--------------------------------------------------------------\*/

/\* \*/

/\* Output variables from Stage.APC\_Post\_CNST: \*/

/\* (1)SPELL\_HRG \*/

/\* (2)CHAPTER \*/

/\* (3)SUB\_CHAPTER \*/

/\* (4)EL\_ACT \*/

/\* (5)NE\_ACT \*/

/\* (6)EL \*/

/\* (7)NE \*/

/\* (8)REV\_TC \*/

/\* Output variables from Stage.APC03\_03e: \*/

/\* \*/

/\*--------------------------------------------------------------\*/

/\*03 CNST

\*03a: Sub-chapters (CNST)

\*/

**proc** **sql** noprint;

create table &StageLib.**.A**PC\_Post\_SSEM\_CNST as

select

uc.SPELL\_HRG,

substr(uc.SPELL\_HRG,**1**,**1**) AS CHAPTER,

substr(uc.SPELL\_HRG,**1**,**2**) AS SUB\_CHAPTER,

uc.EL\_TC,

T02b.ADJ\_NE\_TC AS NE\_TC,

coalesce(uc.EL\_TC,**0**)+coalesce(T02b.ADJ\_NE\_TC,**0**) AS TC,

uc.EL\_ACT,

uc.NE\_ACT

from

&StageLib.**.A**PC\_Unit\_Costs as uc

left join

&StageLib.**.A**PC03\_02b as T02b

ON uc.SPELL\_HRG = T02b.HRG

ORDER BY uc.SPELL\_HRG

;

**quit**;

/\*

-- 03b1: SubChapter Total Costs

\*/

**proc** **sql** noprint;

create table &StageLib.**.A**PC03\_03b1

as

select

in\_08a.SubChapter,

in\_08a.CNST,

Sum(out\_01.TC) AS SubChapter\_TC

FROM

/\*&SQLInputLib..APC\_CNST\_SubChapter as in\_08a \*/ /\*YC - Nov 2017 \*/

Tariff\_R.APC\_CNST\_SubChapter as in\_08a

INNER JOIN &StageLib.**.A**PC\_Post\_SSEM\_CNST AS out\_01

on in\_08a.SubChapter = out\_01.SUB\_CHAPTER

where in\_08a.year="&Year."

GROUP BY

in\_08a.SubChapter,

in\_08a.CNST

;

**quit**;

/\*-- 03b2: Chapter Total Costs \*/

/\*-- Zero for 2013/14 - All costs at Sub-Chapter level \*/

**proc** **sql** noprint;

create table &StageLib.**.A**PC03\_03b2 as

SELECT

in\_08b.Chapter,

in\_08b.CNST,

Sum(out\_01.TC) AS Chapter\_TC

FROM

&StageLib.**.A**PC\_Post\_SSEM\_CNST AS out\_01

/\*INNER JOIN &SQLInputLib..APC\_CNST\_Chapter as in\_08b\*/

INNER JOIN Tariff\_R.APC\_CNST\_Chapter(where=( Year="&Year.")) as in\_08b

ON out\_01.CHAPTER = in\_08b.Chapter

GROUP BY

in\_08b.Chapter, in\_08b.CNST

;

**quit**;

/\*--03c1: SubChapter CNST Increases\*/

**proc** **sql** noprint;

create table &StageLib.**.A**PC03\_03c1

as

select

SubChapter,

CNST,

SubChapter\_TC,

CASE

WHEN coalesce(SubChapter\_TC,**0**)>**0** THEN (CNST+SubChapter\_TC)/SubChapter\_TC

ELSE **0** END AS CNST\_SubCh\_Inc

FROM

&StageLib.**.A**PC03\_03b1

;

**quit**;

/\*--03c2: Chapter CNST Increases

--Zero for 2013/14 - All at Sub-Chapter level

\*/

**proc** **sql** noprint;

create table &StageLib.**.A**PC03\_03c2 as

SELECT

T03b2.Chapter,

T03b2.CNST,

T03b2.Chapter\_TC,

(CNST+Chapter\_TC)/Chapter\_TC AS CNST\_Ch\_Inc

FROM &StageLib.**.A**PC03\_03b2 AS T03b2

;

**quit**;

\*--03d: Apply CNST Increases;

**proc** **sql** noprint;

Create table &StageLib.**.A**PC\_Post\_CNST

as

SELECT

A.\*,

( COALESCE(EL,**0**) + COALESCE(NE,**0**)) as REV\_TC

FROM (

SELECT

out\_01.SPELL\_HRG,

out\_01.CHAPTER,

out\_01.SUB\_CHAPTER,

out\_01.EL\_ACT,

out\_01.NE\_ACT,

CASE WHEN CNST\_Ch\_Inc IS NOT NULL THEN CNST\_Ch\_Inc \* EL\_TC

ELSE CNST\_SubCh\_Inc \* EL\_TC

END as EL ,

CASE WHEN CNST\_Ch\_Inc IS NOT NULL THEN CNST\_Ch\_Inc \* NE\_TC

ELSE CNST\_SubCh\_Inc \* NE\_TC

END as NE

FROM

&StageLib.**.A**PC\_Post\_SSEM\_CNST AS out\_01

LEFT JOIN &StageLib.**.A**PC03\_03c2 AS T03c2

ON out\_01.CHAPTER = T03c2.Chapter

LEFT JOIN &StageLib.**.A**PC03\_03c1 AS T03c1

ON out\_01.SUB\_CHAPTER = T03c1.SubChapter

) AS A

ORDER BY

SPELL\_HRG

;

**quit**;

**data** &StageLib.**.A**PC\_Post\_CNST;

set &StageLib.**.A**PC\_Post\_CNST;

where

find(spell\_hrg, "NZ", "i")=**0**;

**run**;

\*--03e: Summed TC by Admission;

**proc** **sql** noprint;

create table &StageLib.**.A**PC03\_03e

as

SELECT

Sum(NE) AS NE\_TC,

Sum(EL) AS EL\_TC,

Sum(REV\_TC) AS TC

FROM &StageLib.**.A**PC\_Post\_CNST

;

**quit**;

/\*output final dataset to csv file \*/

%let StepNo=Step2\_12;

%let outputfile1=APC\_Post\_CNST;

**proc** **export** data=&StageLib.**.**&outputfile1.

outfile="&outputpath.\&StepNo.\_&outputfile1..csv"

dbms=csv

replace;

**run**;

/\*Step 2.13 Top-slices \*/

/\*--------------------------------------------------------------\*/

\* 2.13 Top-Slices \*/

/\* \*/

/\* Version: 1.01 \*/

/\* Coding: PID \*/

/\* Date: November 2015 \*/

/\* -------------------------------------------------------------\*/

/\* Input: \*/

/\*(1)Stage.APC03\_03e \*/

/\*(2)input.'APC\_Topups Topslice'n \*/

/\*(3)stage.National\_MFF \*/

/\*(4)Input.APC\_ICRS \*/

/\*(5)input.'APC\_D&D\_topslice'n \*/

/\*(6)Stage.APC\_Post\_CNST \*/

/\* \*/

/\* Output: \*/

/\* (1)Stage.APC\_Final\_Unit\_Costs \*/

/\* \*/

/\* \*/

/\*--------------------------------------------------------------\*/

/\* \*/

/\* Output variables from Stage.APC\_Final\_Unit\_Costs: \*/

/\* \*/

/\*(1)SPELL\_HRG \*/

/\*(2)NE\_UC \*/

/\*(3)EL\_UC \*/

/\*(4)NE\_ACT \*/

/\*(5)EL\_ACT \*/

/\*--------------------------------------------------------------\*/

\*--04a: Topslice for Specialist Top-Ups;

\*--Calculates % topslice for specialised service top-ups, by admission;

**proc** **sql**;

create table &StageLib.**.A**PC03\_04a as

SELECT

(NE\_TC-(Spec\_NE/NAT\_MFF))/NE\_TC AS NE\_TS,

(EL\_TC-((Spec\_EL+Spec\_DC)/NAT\_MFF))/EL\_TC AS EL\_TS

FROM

&StageLib.**.A**PC03\_03e as a,

Tariff\_R.APC\_Topups\_Topslice(where=(Year="&Year.")) as b,

&StageLib.**.N**ational\_MFF as c

;

**quit**;

\*04b: Topslice for ICRS;

\* Calculates % non-elective topslice for ICRS;

**proc** **sql** noprint;

create table &StageLib.**.A**PC03\_04B

AS

SELECT

(NE\_TC-(ICRS/NAT\_MFF))/NE\_TC AS ICRS\_TS

FROM

Tariff\_R.APC\_ICRS(where=(Year="&Year.")) as a,

&StageLib.**.A**PC03\_03e as b,

&StageLib.**.N**ational\_MFF as c

;

**quit**;

\*--04c: D&D topslice;

\*--Calculates % topslice for drugs and device, by admission, as well as topslice from limiting cost removal;

**proc** **sql** noprint;

create table &StageLib.**.A**PC03\_04c

as

SELECT

((TC)-LIMIT\_TS-DandD\_TS)/(TC) AS ALL

FROM

&StageLib.**.A**PC03\_03e,

&StageLib.**.A**PC\_DandD\_topslice

;

**quit**;

\*--04d: Compound Topslices;

\*--Calculates compound topslices by admission;

**proc** **sql** noprint;

create table &StageLib.**.A**PC03\_04D

as

SELECT

NE\_TS\*ICRS\_TS\*ALL AS NE\_Compound\_TS,

EL\_TS\*ALL AS EL\_DC\_Compound\_TS

FROM

&StageLib.**.A**PC03\_04a,

&StageLib.**.A**PC03\_04B,

&StageLib.**.A**PC03\_04c

;

**quit**;

\*--04e: Total Costs after Topslices;

\*--Applies topslices to costs, by HRG and admission;

**proc** **sql** noprint;

create table &StageLib.**.A**PC03\_04e

as

SELECT

cnst.SPELL\_HRG,

NE\*NE\_Compound\_TS AS NE\_TC,

EL\*EL\_DC\_Compound\_TS AS EL\_TC,

cnst.NE\_ACT,

cnst.EL\_ACT

FROM &StageLib.**.A**PC03\_04D,

&StageLib.**.A**PC\_Post\_CNST as cnst

ORDER BY cnst.SPELL\_HRG

;

**quit**;

\*--05 - FINAL UNIT COSTS;

\*--05: Unit Costs;

**proc** **sql** noprint;

create table &StageLib.**.A**PC\_Final\_Unit\_Costs

as

SELECT

T04e.SPELL\_HRG,

CASE WHEN Zero\_Price\_HRGs IS NOT null OR NE\_ACT = **0** THEN **0**

ELSE NE\_TC/NE\_ACT END as NE\_UC ,

CASE WHEN Zero\_Price\_HRGs IS NOT null OR EL\_ACT = **0** THEN **0**

ELSE EL\_TC/EL\_ACT END as EL\_UC ,

T04e.NE\_ACT,

T04e.EL\_ACT

FROM

&StageLib.**.A**PC03\_04e AS T04e

LEFT JOIN Tariff\_R.ZeroPriceHRG (where=(Year="&Year.")) AS IN\_03

ON T04e.SPELL\_HRG = IN\_03.Zero\_Price\_HRGs

;

**quit**;

/\*output final dataset to csv file \*/

%let StepNo=Step2\_13;

%let outputfile1=APC\_Final\_Unit\_Costs;

**proc** **export** data=&StageLib.**.**&outputfile1.

outfile="&outputpath.\&StepNo.\_&outputfile1..csv"

dbms=csv

replace;

**run**;

/\*Step 2.14 Combining low volume activity \*/

/\* \*/

/\*--------------------------------------------------------------\*/

\* Step2\_14\_Combining low volume activity \*/

/\* \*/

/\* Version: 1.01 \*/

/\* Coding: PID \*/

/\* Date: November 2015 \*/

/\* -------------------------------------------------------------\*/

/\* Input: \*/

/\* (1)Stage.APC\_Final\_Unit\_Costs \*/

/\* (2)Stage.APC\_EBD\_UC (From Step 2.8) \*/

/\* Output: \*/

/\* (1)Stage.APC04\_01 \*/

/\* \*/

/\* \*/

/\*--------------------------------------------------------------\*/

/\* \*/

/\* Output variables from Stage.APC04\_01: \*/

/\*(1)SPELL\_HRG \*/

/\*(2)NE\_ACT \*/

/\*(3)EL\_ACT \*/

/\*(4)TOTAL\_ACT \*/

/\*(5)NE\_TC \*/

/\*(6)EL\_TC \*/

/\*(7)EL\_UC \*/

/\*(8)NE\_UC \*/

/\*(9)HRG \*/

/\*(10)FCE\_EBD\_UC\_CAPPED \*/

/\*(11)TC \*/

/\*(12)COMBINE \*/

/\*(13)EL\_COMB\_TC \*/

/\*(14)NE\_COMB\_TC \*/

/\*(15)EL\_COMB\_UC \*/

/\*(16)NE\_COMB\_UC \*/

/\*(17)EBD\_UC \*/

/\* \*/

/\*--------------------------------------------------------------\*/

/\*

\*--Query 01: Combine low volume activity;

\*--Only used where no data for an admission method - No combining of EL/NE data in 2013/14. Stage left in for potential future use;

\*/

**proc** **sql** noprint;

create table &StageLib.**.C**ombine\_Low\_volume\_Activity

as

SELECT

b.\*,

CASE WHEN COMBINE = **1** THEN TC/**2**

ELSE EL\_TC END as EL\_COMB\_TC,

CASE WHEN COMBINE = **1** THEN TC/**2**

ELSE NE\_TC END as NE\_COMB\_TC,

CASE WHEN COMBINE = **1** THEN

CASE WHEN (TOTAL\_ACT Is Null) Or (TOTAL\_ACT=**0**) THEN **0**

ELSE TC/TOTAL\_ACT

END

ELSE EL\_UC END as EL\_COMB\_UC,

CASE WHEN COMBINE = **1** THEN

CASE WHEN TOTAL\_ACT Is Null Or TOTAL\_ACT=**0** THEN **0**

ELSE TC/TOTAL\_ACT END

ELSE NE\_UC END as NE\_COMB\_UC,

CASE WHEN HRG = 'UZ01Z' OR HRG='PB03Z' OR (HRG IS NULL) THEN **0**

ELSE FCE\_EBD\_UC\_CAPPED END as EBD\_UC

FROM (

SELECT a.\*,

coalesce(EL\_TC,**0**)+coalesce(NE\_TC,**0**) AS TC,

CASE WHEN (TOTAL\_ACT Is Null) Or (EL\_ACT Is Null) Or (NE\_ACT IS NULL) THEN **1**

ELSE **0** END as COMBINE

FROM

(SELECT

in\_08.SPELL\_HRG,

in\_08.NE\_ACT,

in\_08.EL\_ACT,

coalesce(NE\_ACT,**0**)+coalesce(EL\_ACT,**0**) AS TOTAL\_ACT,

(NE\_UC\*NE\_ACT) AS NE\_TC,

(EL\_UC\*EL\_ACT) AS EL\_TC,

EL\_UC,

NE\_UC,

HRG,

FCE\_EBD\_UC\_CAPPED

FROM

&StageLib.**.A**PC\_Final\_Unit\_Costs as in\_08

LEFT JOIN &StageLib.**.A**PC\_EBD\_UC as in\_07

ON in\_08.SPELL\_HRG = in\_07.HRG

) AS a

) AS b

ORDER BY SPELL\_HRG

;

**quit**;

**DATA** XXX; SET &StageLib.**.A**PC\_Final\_Unit\_Costs; **RUN**;

/\*output final dataset to csv file \*/

%let StepNo=Step2\_14;

%let outputfile1=Combine\_Low\_volume\_Activity;

**proc** **export** data=&StageLib.**.**&outputfile1.

outfile="&outputpath.\&StepNo.\_&outputfile1..csv"

dbms=csv

replace;

**run**;

/\*Step 2.15 Step2\_15\_Stage2\_Output\_Price \*/

/\* \*/

/\*--------------------------------------------------------------\*/

/\* Program Name: Step2\_15\_Stage2\_Output\_Price \*/

/\* \*/

/\* Version: 1.01 \*/

/\* Coding: PID \*/

/\* Date: November 2015 \*/

/\* -------------------------------------------------------------\*/

/\* Input: \*/

/\* (1)Stage.APC04\_01 \*/

/\* (2)Input.Uplift \*/

/\* (3)Input.Scaling \*/

/\* (4)Stage.National\_MFF \*/

/\* (5)Input.SSEM\_Banding \*/

/\* (6)Input.V\_HES\_Trimpoints \*/

/\* (7)Input.EligibleHRG\_UZ01Z\_PB03Z \*/

/\* \*/

/\* Output: \*/

/\* (1) Stage.APC\_Tariff \*/

/\* \*/

/\* \*/

/\*--------------------------------------------------------------\*/

/\* \*/

/\* Output variables from Stage.APC\_Tariff: \*/

/\*(1)SPELL\_HRG \*/

/\*(2)HRG\_Name \*/

/\*(3)EL\_Tariff \*/

/\*(4)EL\_Trim \*/

/\*(5)NE\_Tariff \*/

/\*(6)NE\_Trim \*/

/\*(7)Longstay\_Payment \*/

/\*(8)SSEM\_Eligible \*/

/\*(9)SSEM\_Tariff \*/

/\*(10)SSNDS\_Eligible \*/

/\* \*/

/\*--------------------------------------------------------------\*/

\*--02: Prices;

\*--Calculates prices and brings together other structural and pricing information;

**proc** **sql** noprint;

select UPLIFT into :UPLIFT

from Tariff\_R.Uplift

where Year="&Year."

;

%let UPLIFT=&UPLIFT;

%put UPLIFT=&UPLIFT;

select EL\_SCALE,NE\_SCALE

into :EL\_SCALE,:NE\_SCALE

from Tariff\_R.Scaling

where Year="&Year."

;

%let EL\_SCALE=&EL\_SCALE;

%let NE\_SCALE=&NE\_SCALE;

%put EL\_SCALE=&EL\_SCALE;

%put NE\_SCALE=&NE\_SCALE;

select NAT\_MFF ,MFF\_REBASE

into :NAT\_MFF ,:MFF\_REBASE

from &StageLib.**.N**ational\_MFF

;

%let NAT\_MFF =&NAT\_MFF;

%let MFF\_REBASE=&MFF\_REBASE;

**quit**;

**proc** **sql** noprint;

create table &StageLib.**.A**PC04\_02

as

select a.\*,

CASE WHEN a.SSEM\_Eligible = 'Yes' then (a.NE\_COMB\_UC \* a.UPLIFT \* a.NE\_SCALE \* a.MFF\_REBASE \* a.Tariff\_Pct )

ELSE **0** END as SSEM\_Tariff

FROM(

select

SPELL\_HRG,

(EL\_COMB\_UC\*&UPLIFT.\*&EL\_SCALE.\*&MFF\_REBASE.) as EL\_Tariff,

CASE WHEN

(CASE WHEN COMBINE=**1** THEN DC\_EL\_NE

ELSE DC\_EL

END) < **0** THEN **5**

ELSE (

CASE WHEN COMBINE=**1** THEN DC\_EL\_NE

ELSE DC\_EL

END)

END as EL\_Trim,

(NE\_COMB\_UC\*&UPLIFT.\*&NE\_SCALE.\*&MFF\_REBASE.) as NE\_Tariff,

CASE WHEN

(CASE WHEN COMBINE=**1** THEN DC\_EL\_NE

ELSE NE

END) < **0** THEN **5**

ELSE (

CASE WHEN COMBINE=**1** THEN DC\_EL\_NE

ELSE NE

END)

END as NE\_Trim,

CASE WHEN HRG\_Code='LA08E' Or HRG\_Code='SB97Z' Or HRG\_Code='SC97Z' THEN **0**

Else EBD\_UC\*&UPLIFT.

END as Longstay\_Payment ,

CASE WHEN Eligible\_for\_SSEM=**1** And Band>**1** THEN 'Yes'

ELSE 'No' END as SSEM\_Eligible ,

NE\_COMB\_UC,

&UPLIFT. as UPLIFT,

&NE\_SCALE as NE\_SCALE,

&MFF\_REBASE. as MFF\_REBASE,

Tariff\_Pct

from

&StageLib.**.C**ombine\_Low\_volume\_Activity as APC04\_01

/\* ------------------------------------------------------------ Need to change to HRGELIGIBILITY when new data comes ------------------\*/

INNER JOIN Tariff\_R.HRGSELIGIBILITY\_TEMP (where=(year="&Year.")) as in\_01a

ON APC04\_01.SPELL\_HRG = in\_01a.HRG\_Code

inner join Tariff\_R.SSEM\_Banding(where=(year="&Year.")) as in\_02

ON in\_02.Band = in\_01a.SSEM\_Banding

INNER JOIN &SQLViewLib.**.H**ES\_Trimpoints as in\_05

ON in\_01a.HRG\_Code = in\_05.HRG and in\_05.Run\_ID=&param\_HESRunID.

) as a

ORDER BY a.SPELL\_HRG

;

**quit**;

\*--03: Add UZ01Z and PB03Z to prices;

\*--Combines main tariff prices with those for UZ01Z and PB03Z;

**proc** **sql** noprint;

create table &StageLib.**.A**PC04\_03

as

SELECT

Spell\_HRG,

EL\_Tariff,

EL\_Trim,

NE\_Tariff,

NE\_Trim,

Longstay\_Payment,

SSEM\_Eligible,

SSEM\_Tariff

FROM &StageLib.**.A**PC04\_02

UNION

SELECT

SpellHRG as Spell\_HRG,

EL\_Tariff,

EL\_Trim,

NE\_Tariff,

NE\_Trim,

Longstay\_Payment,

SSEM\_Eligible,

SSEM\_Tariff

FROM

Tariff\_R.EligibleHRG\_UZ01Z\_PB03Z

where Year="&Year."

;

**quit**;

\*--04: Final Prices ;

\*--Maketable of union query above;

**proc** **sql** noprint;

create table &StageLib.**.A**PC\_Tariff

as

SELECT

T03a.Spell\_HRG,

T01a.HRG\_Name,

T03a.EL\_Tariff,

T03a.EL\_Trim,

T03a.NE\_Tariff,

T03a.NE\_Trim,

T03a.Longstay\_Payment,

T03a.SSEM\_Eligible,

T03a.SSEM\_Tariff,

case when Eligible\_for\_Specialist\_Top\_ups = **0** then 'No'

ELSE 'Yes' END as SSNDS\_Eligible

FROM

&StageLib.**.A**PC04\_03 as T03a

LEFT JOIN Tariff\_R.HRGSELIGIBILITY\_TEMP(where=( Year="&Year.")) as T01a

ON T03a.Spell\_HRG=T01a.HRG\_Code

order by T03a.Spell\_HRG

;

**quit**;

%let StepNo=Step2\_15;

%let outputfile1=APC\_Tariff;

**proc** **export** data=&StageLib.**.**&outputfile1.

outfile="&outputpath.\&StepNo.\_&outputfile1..csv"

dbms=csv

replace;

**run**;

/\* PID added 20151110\*/

FILENAME DelFile2 "&outputpath2.\DC\_NO\Prj\_Report\_APC17.xls"; /\* PID in \*/

**DATA** \_NULL\_ ;

rc = FDELETE('DelFile2');

**RUN** ;

FILENAME DelFile2 CLEAR ;

/\*part 2 output all results from all steps - PID DELETED \*/

/\*Step 2.1 Calculate Inlier and Excess Bed Day (EDB) Cost \*/

/\*--------------------------------------------------------------\*/

/\* Program Name: Step2\_1\_Calculate\_Inlier\_EBD\_Cost \*/

/\* Version: 1.01 \*/

/\* Author: PID \*/

/\* Date: July 2015 \*/

/\* -------------------------------------------------------------\*/

/\* Input: \*/

/\* (1) Input.RefCostAcute \*/

/\* (2) Input.ProviderTable \*/

/\* Output: \*/

/\* Stage.APC\_ScopeData \*/

/\*--------------------------------------------------------------\*/

/\* \*/

/\* Output variables from Stage.APC\_ScopeData: \*/

/\*(1)HRG\_POD \*/

/\*(2)HRG \*/

/\*(3)ADMISSION \*/

/\*(4)PROVIDER \*/

/\*(5)Chapter \*/

/\*(6)SubChapter \*/

/\*(7)INLIER\_ACT \*/

/\*(8)INLIER\_COST \*/

/\*(9)EBD\_ACT \*/

/\*(10)EBD\_COST \*/

/\*--------------------------------------------------------------\*/

%let &vDC=NO;

**%macro** ***check\_DC***;

%put vDC=&vDC;

%if %upcase(&vDC.)=YES %then %let vlen=5;

%else %let vlen=2;

%put vlen=&vlen;

proc sql noprint;

create table work.APC\_ScopeData\_1 (drop=ADMISSION\_Full)

as

SELECT distinct

rfa.CURRENCY AS HRG,

CASE strip(rfa.DEPARTMENT)

WHEN 'EI' THEN 'EL'

when 'NEI\_L' then 'NE'

WHEN 'NEI\_S' then 'NE'

WHEN 'RCRA' then 'DC'

ELSE DEPARTMENT

END AS ADMISSION\_Full informat $20. format $20. length **20**,

substr(calculated ADMISSION\_Full,**1**,&vlen.) as ADMISSION informat $20. format $20. length **20**,

catx("\_",HRG,calculated ADMISSION) as HRG\_POD informat $20. format $20. length **20**,

substr(HRG,**1**,**1**) as Chapter,

substr(HRG,**1**,**2**) as SubChapter,

rfa.FK\_ORGS\_PROV\_ID AS PROVIDER,

Sum(rfa.FCE) AS INLIER\_ACT,

Sum(rfa.UNIT\_COST\*FCE) AS INLIER\_COST ,

Sum(rfa.EXCESS\_BED\_DAYS\_ACTIVITY) AS EBD\_ACT,

Sum(rfa.EXCESS\_BED\_DAYS\_UNIT\_COST \*EXCESS\_BED\_DAYS\_ACTIVITY) AS EBD\_COST informat **20.6** format **20.6**

from &lib.**.F**INAL\_APC\_CLEANED\_REFCOST\_F rfa

inner join TC\_MFF\_Selected as pt

ON rfa.FK\_ORGS\_PROV\_ID = pt.RC\_Code

Group by rfa.CURRENCY,ADMISSION,rfa.FK\_ORGS\_PROV\_ID

order by rfa.CURRENCY, ADMISSION,rfa.FK\_ORGS\_PROV\_ID

;

quit;

**%mend** check\_DC;

%***check\_DC***;

/\*remove the duplicated records \*/

**proc** **sql** noprint;

create table &StageLib.**.A**PC\_ScopeData

as

select distinct

HRG\_POD,

HRG,

ADMISSION,

PROVIDER,

Chapter,

SubChapter,

INLIER\_ACT,

INLIER\_COST,

EBD\_ACT,

EBD\_COST

from APC\_ScopeData\_1

order by HRG\_POD, HRG, ADMISSION

;

**quit**;

%let StepNo=Step2\_1;

/\*Reporting \*/

**proc** **means** data=&StageLib.**.A**PC\_ScopeData noprint;

var INLIER\_ACT INLIER\_COST;

by HRG\_POD;

Output OUT=&StageLib.**.R**pt\_APC\_ScopeData\_Sum (drop=\_TYPE\_ \_FREQ\_)

Sum= Inlier\_ACT\_Total Inlier\_Cost\_TC ;

**run**;

**proc** **sql** noprint;

create table &StageLib.**.R**pt\_2\_1\_APC\_ScopeData\_UnitCost as

select distinct

HRG\_POD,

substr(hrg\_pod,**1**,**1**) as Chapter,

substr(hrg\_pod,**1**,**2**) as Subchapter,

Inlier\_Cost\_TC ,

Inlier\_ACT\_Total,

CASE

WHEN Inlier\_ACT\_Total IS NULL THEN **0**

ELSE Inlier\_Cost\_TC/Inlier\_ACT\_Total

END as Inlier\_Unit\_Cost

from &StageLib.**.R**pt\_APC\_ScopeData\_Sum

order by HRG\_POD,Chapter,Subchapter

;

**quit**;

/\*output final dataset to csv file \*/

%let outputfile3=Rpt\_2\_1\_APC\_ScopeData\_UnitCost;

**proc** **export** data=&StageLib.**.**&outputfile3.

outfile="&outputpath.\&StepNo.\_&outputfile3..csv"

dbms=csv

replace;

**run**;

/\*Step2\_3\_Removing\_MFF\*/

/\*Step 2.3 Removing Market Forces Factor (MFF)

/\*--------------------------------------------------------------\*/

/\* Program Name: Step2\_3\_Removing\_MFF \*/

/\* Version: 1.01 \*/

/\* Author: PID \*/

/\* Date: July 2015 \*/

/\* -------------------------------------------------------------\*/

/\* Input: \*/

/\* (1) Stage.APC\_ScopeData (from Step 2\_1) \*/

/\* (2) Input.ProviderTable \*/

/\* Output: \*/

/\* (1)Stage.APC01\_03a \*/

/\* (2)Stage.National\_MFF \*/

/\*--------------------------------------------------------------\*/

/\* \*/

/\* Output variables : \*/

/\* Stage.APC01\_03a: \*/

/\*(1)PROVIDER \*/

/\*(2)HRG (CURRENCY) \*/

/\*(3)ADMISSION \*/

/\*(4)INLIER\_ACT \*/

/\*(5)INLIER\_TC \*/

/\*(6)INLIER\_TC\_Target\_MFF \*/

/\*(7)EBD\_ACT \*/

/\*(8)EBD\_TC \*/

/\*(9)EBD\_TC\_Target\_MFF \*/

/\*(10)INLIER\_TC\_Payment\_MFF \*/

/\*Stage.National\_MFF: \*/

/\* (1)NAT\_MFF \*/

/\* (2)MFF\_REBASE \*/

/\*--------------------------------------------------------------\*/

/\*---\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*---\*/

/\*PLEASE DO NOT MAKE CHANGE AFTER THIS LINE \*/

/\*---\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*---\*/

/\* 03 - MARKET FORCES FACTOR \*/

/\*Prepare the input data \*/

**proc** **sql** noprint;

Create table APC\_ScopeData\_TC

as select distinct

HRG,

ADMISSION,

PROVIDER,

INLIER\_ACT,

INLIER\_COST as INLIER\_TC,

EBD\_ACT,

EBD\_COST as EBD\_TC

from &StageLib.**.A**PC\_ScopeData

order by HRG, ADMISSION, PROVIDER

;

**quit**;

/\*03a: Remove provider MFF \*/

**proc** **sql** noprint;

create table APC01\_03a\_1 as

SELECT distinct

tc.HRG,

tc.ADMISSION,

tc.PROVIDER,

tc.INLIER\_ACT,

tc.INLIER\_TC,

(tc.INLIER\_TC/pt.Target\_MFF) AS INLIER\_TC\_Target\_MFF,

tc.EBD\_ACT,

tc.EBD\_TC,

(tc.EBD\_TC/pt.Target\_MFF) AS EBD\_TC\_Target\_MFF,

tc.INLIER\_TC/pt.Capped\_MFF AS INLIER\_TC\_Payment\_MFF

FROM

APC\_ScopeData\_TC as tc

INNER JOIN TC\_MFF\_Selected as pt

ON tc.PROVIDER = pt.RC\_Code

order by tc.HRG ,tc.ADMISSION

;

**quit**;

**proc** **sql** noprint;

create table &StageLib.**.A**PC01\_03a as

select

catx("\_",tc.HRG,tc.ADMISSION) as HRG\_POD informat $20. format $20. length **20**,

tc.HRG,

tc.ADMISSION,

tc.PROVIDER,

substr(HRG,**1**,**1**) as Chapter,

substr(HRG,**1**,**2**) as SubChapter,

tc.INLIER\_ACT,

tc.INLIER\_TC,

tc.INLIER\_TC\_Target\_MFF,

tc.EBD\_ACT,

tc.EBD\_TC,

tc.EBD\_TC\_Target\_MFF,

tc.INLIER\_TC\_Payment\_MFF

from APC01\_03a\_1 as tc

order by HRG\_POD

;

**quit**;

/\*03b: Calculate national MFF \*/

**proc** **sql** noprint;

create table &StageLib.**.N**ational\_MFF as

SELECT

Sum(Inlier\_TC)/Sum(INLIER\_TC\_Target\_MFF) AS NAT\_MFF,

Sum(INLIER\_TC\_Target\_MFF)/Sum(INLIER\_TC\_Payment\_MFF) AS MFF\_REBASE

FROM &StageLib.**.A**PC01\_03a

;

**quit**;

/\*Reporting \*/

**proc** **means** data=&StageLib.**.A**PC01\_03a noprint;

var INLIER\_ACT INLIER\_TC\_Target\_MFF;

by HRG\_POD;

Output OUT=&StageLib.**.R**pt\_APC01\_03a\_Sum (drop=\_TYPE\_ \_FREQ\_)

Sum= Inlier\_ACT\_Total INLIER\_TC\_Target\_MFF\_Total ;

**run**;

**proc** **sql** noprint;

create table &StageLib.**.R**pt\_2\_3\_APC01\_03a\_UnitCost as

select distinct

HRG\_POD,

substr(hrg\_pod,**1**,**1**) as Chapter,

substr(hrg\_pod,**1**,**2**) as Subchapter,

INLIER\_TC\_Target\_MFF\_Total,

Inlier\_ACT\_Total,

CASE

WHEN Inlier\_ACT\_Total IS NULL THEN **0**

ELSE INLIER\_TC\_Target\_MFF\_Total/Inlier\_ACT\_Total

END as INLIER\_TC\_Target\_MFF\_Unit\_Cost

from &StageLib.**.R**pt\_APC01\_03a\_Sum

order by HRG\_POD,Chapter,Subchapter

;

**quit**;

%let StepNo=Step2\_3;

%let outputfile4=Rpt\_2\_3\_APC01\_03a\_UnitCost;

**proc** **export** data=&StageLib.**.**&outputfile4.

outfile="&outputpath.\&StepNo.\_&outputfile4..csv"

dbms=csv

replace;

**run**;

/\*Step2\_4\_Data\_Clean\*/

/\*Step 2.4 Data Clean

/\*--------------------------------------------------------------\*/

\* Step2\_4\_Data\_Clean \*/

/\* Version: 1.01 \*/

/\* Author: PID \*/

/\* Date: July 2015 \*/

/\* -------------------------------------------------------------\*/

/\* Input: \*/

/\* Stage.APC01\_03a (from Step2\_3 ) \*/

/\* \*/

/\* Output: \*/

/\*(1)Stage.APC01\_04a \*/

/\*(2)Stage.APC\_CleanData \*/

/\*(3)Stage.APC\_HRG\_LevelData \*/

/\*--------------------------------------------------------------\*/

/\* \*/

/\* Output variables : \*/

/\*Stage.APC01\_04a: \*/

/\*(1)HRG, \*/

/\*(2)ADMISSION, \*/

/\*(3)INLIER\_NA, \*/

/\*(4)EBD\_NA \*/

/\*Stage.APC\_CleanData: \*/

/\*(1) HRG \*/

/\*(2)ADMISSION \*/

/\*(3)PROVIDER \*/

/\*(4)INLIER\_ACT \*/

/\*(5)INLIER\_NA \*/

/\*(6)INLIER\_PA \*/

/\*(7)EBD\_ACT \*/

/\*(8)EBD\_NA \*/

/\*(9)EBD\_PA \*/

/\*(10)EBD\_TC\_Target\_MFF \*/

/\*(11)CLEAN\_INLIER\_FLAG \*/

/\*(12)CLEAN\_INLIER\_ACT \*/

/\*(13)CLEAN\_INLIER\_PA \*/

/\*(14)CLEAN\_INLIER\_TC \*/

/\*(15)CLEAN\_EBD\_FLAG \*/

/\*(16)CLEAN\_EBD\_ACT \*/

/\*(17)CLEAN\_EBD\_PA \*/

/\*(18)CLEAN\_EBD\_TC \*/

/\*(19)CLEAN\_TC \*/

/\* Stage.APC\_HRG\_LevelData: \*/

/\*(1)HRG \*/

/\*(2)ADMISSION \*/

/\*(3)Clean\_INLIER\_ACT \*/

/\*(5)Clean\_INLIER\_TC \*/

/\*(6)Clean\_EBD\_ACT \*/

/\*(8)Clean\_EBD\_TC \*/

/\*(9)Clean\_TC \*/

/\*--------------------------------------------------------------\*/

%let Param\_OutlierMinValue = 5;

%let Param\_OutlierMaxValue = 20;

**proc** **sql** ;

create table Dis\_APC01\_03a as

select distinct

ADMISSION,

HRG,

Provider,

INLIER\_ACT,

INLIER\_TC,

INLIER\_TC\_Target\_MFF,

EBD\_ACT,

EBD\_TC,

EBD\_TC\_Target\_MFF

from

&StageLib.**.A**PC01\_03a

order by Admission, HRG

;

**quit**;

**proc** **sql** noprint;

create table &StageLib.**.A**PC01\_04a (drop=INLIER\_NA\_C EBD\_NA\_C)

as

SELECT distinct

T03a.ADMISSION,

T03a.HRG,

CASE

WHEN sum(INLIER\_ACT) IS NULL THEN **0**

WHEN sum(INLIER\_ACT) = **0** THEN **0**

ELSE

Sum(INLIER\_TC\_Target\_MFF) / Sum(INLIER\_ACT)

END as INLIER\_NA\_C,

CASE

when calculated INLIER\_NA\_C =**.** then **0**

ELSE calculated INLIER\_NA\_C

END as INLIER\_NA,

CASE

WHEN sum(EBD\_ACT) IS NULL THEN **0**

WHEN sum(EBD\_ACT) = **0** THEN **0**

ELSE

Sum(EBD\_TC\_Target\_MFF) / Sum(EBD\_ACT)

END as EBD\_NA\_C,

CASE

when calculated EBD\_NA\_C =**.** then **0**

ELSE calculated EBD\_NA\_C

END as EBD\_NA

FROM Dis\_APC01\_03a AS T03a

GROUP BY T03a.ADMISSION,T03a.HRG

;

**quit**;

/\*--RE work of 04b: Clean Data \*/

**proc** **sql** noprint;

Create table &StageLib.**.A**PC\_CleanData as

SELECT

d.\*,

COALESCE(CLEAN\_INLIER\_TC,**0**) + COALESCE(CLEAN\_EBD\_TC,**0**) as CLEAN\_TC

FROM

(SELECT

c.\*,

CLEAN\_EBD\_PA \* CLEAN\_EBD\_ACT as CLEAN\_EBD\_TC

FROM

(select

B.\*,

CASE

WHEN CLEAN\_EBD\_FLAG = **0**

THEN EBD\_PA

ELSE **0**

END as CLEAN\_EBD\_PA ,

CASE

WHEN CLEAN\_EBD\_FLAG = **0**

THEN EBD\_ACT

ELSE **0**

END as CLEAN\_EBD\_ACT

from

(select

a.\*,

CASE

when EBD\_PA > **0**

THEN

CASE

WHEN &param\_OutlierMaxValue=**0** OR &param\_OutlierMaxValue IS NULL

THEN **0**

ELSE

/\* Outlier Change Start -- 18/12/2014 \*/

CASE WHEN

(EBD\_PA > EBD\_NA\*&param\_OutlierMaxValue) /\* value 20 replaced with @param\_OutlierMaxValue parameter\*/

OR (EBD\_PA < EBD\_NA/&param\_OutlierMaxValue) /\* value 20 replaced with @param\_OutlierMaxValue parameter\*/

OR (INLIER\_ACT<>**0** AND INLIER\_ACT Is Not Null AND EBD\_ACT/INLIER\_ACT > **365**)

THEN **0**

ELSE **0**

END

END

ELSE **0**

/\* Outlier Change End \*/

END as CLEAN\_EBD\_FLAG

from

(select

z.\*,

z.CLEAN\_INLIER\_PA \* z.CLEAN\_INLIER\_ACT as CLEAN\_INLIER\_TC ,

CASE

WHEN EBD\_ACT= **0** OR EBD\_ACT IS null THEN **0**

ELSE EBD\_TC\_Target\_MFF / EBD\_ACT

END as EBD\_PA

FROM

(select

y.\*,

CASE

WHEN y.CLEAN\_INLIER\_FLAG = **0**

THEN y.INLIER\_PA

ELSE **0**

END as CLEAN\_INLIER\_PA ,

CASE

WHEN y.CLEAN\_INLIER\_FLAG = **0**

THEN y.INLIER\_ACT

ELSE **0** END as CLEAN\_INLIER\_ACT

from

(select

x.\*,

/\* Outlier Change Start 18/12/2014 \*/

CASE WHEN &param\_OutlierMaxValue=**0** OR &param\_OutlierMaxValue IS NULL THEN **0**

ELSE

CASE

WHEN x.INLIER\_PA > **0** AND x.INLIER\_PA > (x.INLIER\_NA\*&param\_OutlierMaxValue) THEN **0**

WHEN x.INLIER\_PA > **0** AND x.INLIER\_PA < (x.INLIER\_NA/&param\_OutlierMaxValue) THEN **0**

ELSE **0**

END

END as CLEAN\_INLIER\_FLAG

/\* value 20 replaced with &param\_OutlierMaxValue parameter \*/

/\* Outlier Change Start \*/

from

(select

T03a.PROVIDER,

T03a.ADMISSION,

T03a.HRG,

T04a.INLIER\_NA,

T04a.EBD\_NA,

T03a.EBD\_ACT,

T03a.EBD\_TC\_Target\_MFF,

T03a.INLIER\_ACT,

CASE

WHEN T03a.INLIER\_ACT = **0** OR T03a.INLIER\_ACT IS NULL

THEN **0**

ELSE T03a.INLIER\_TC\_Target\_MFF / T03a.INLIER\_ACT

END as INLIER\_PA

FROM Dis\_APC01\_03a AS T03a

INNER JOIN &StageLib.**.A**PC01\_04a AS T04a

ON (T03a.ADMISSION = T04a.ADMISSION)

AND (T03a.HRG = T04a.HRG)) as x

) as y

) as z

) as a

) as b

) as c

) as d

order by d.HRG, d.ADMISSION

;

**quit**;

/\*04c-Aggregate Clean Data to HRG Provider\*/

**proc** **sql** noprint;

create table APC\_HRGLevelData\_1 as

SELECT distinct

HRG,

ADMISSION,

sum(coalesce(CLEAN\_INLIER\_ACT,**0**)) AS CLEAN\_INLIER\_ACT,

Sum(CASE HRG

WHEN 'UZ01Z' THEN **0**

ELSE coalesce(CLEAN\_INLIER\_TC,**0**)

END) as CLEAN\_INLIER\_TC,

Sum(coalesce(CLEAN\_EBD\_ACT,**0**)) AS CLEAN\_EBD\_ACT,

Sum(CASE HRG

WHEN 'UZ01Z' THEN **0**

ELSE coalesce(CLEAN\_EBD\_TC,**0**)

END) AS CLEAN\_EBD\_TC,

Sum(CASE HRG

WHEN 'UZ01Z' THEN **0**

ELSE coalesce(CLEAN\_TC,**0**)

END) AS CLEAN\_TC

FROM &StageLib.**.A**PC\_CleanData

GROUP BY HRG, ADMISSION

ORDER BY HRG, ADMISSION

;

**quit**;

**proc** **sql** noprint;

create table &StageLib.**.A**PC\_HRGLevelData as

select distinct

catx("\_",HRG,ADMISSION) as HRG\_POD informat $20. format $20. length **20**,

HRG,

ADMISSION,

substr(HRG,**1**,**1**) as Chapter,

substr(HRG,**1**,**2**) as SubChapter,

CLEAN\_INLIER\_ACT,

CLEAN\_INLIER\_TC,

CLEAN\_EBD\_ACT,

CLEAN\_EBD\_TC,

CLEAN\_TC,

(CLEAN\_INLIER\_TC / CLEAN\_INLIER\_ACT) as Clean\_Inlier\_UC

from

APC\_HRGLevelData\_1

order by HRG\_POD, HRG,ADMISSION,Chapter,SubChapter

;

**quit**;

%let StepNo=Step2\_4;

%let outputfile3=APC\_HRGLevelData;

**proc** **export** data=&StageLib.**.**&outputfile3.

outfile="&outputpath.\&StepNo.\_&outputfile3..csv"

dbms=csv

replace;

**run**;

/\*Step 2.5 Inclusion of cost in A\_E leading to Admission \*/

/\*--------------------------------------------------------------\*/

/\* Step2\_5\_Inclusion\_Cost\_in\_AE\_to\_Admission \*/

/\* Version: 1.01 \*/

/\* Coding: PID \*/

/\* Date: July 2015 \*/

/\* -------------------------------------------------------------\*/

/\* Input: \*/

/\* (1)Input.HES\_Adm\_A\_E \*/

/\* (2)Input.'A\_E\_Adm\_Cost'n \*/

/\* (3)Stage.APC\_HRG\_LevelData (from Step3\_3) \*/

/\* \*/

/\* Output: \*/

/\*(1) Stage.APC02\_01a; \*/

/\*(2)Stage.APC02\_01b; \*/

/\*(3)Stage.APC02\_01c; \*/

/\*(4)Stage.APC\_A\_E\_Adm \*/

/\*--------------------------------------------------------------\*/

/\* \*/

/\* Output variables from Stage.APC\_A\_E\_Adm: \*/

/\*(1)HRG\_POD \*/

/\*(2)HRG \*/

/\*(3)ADMISSION \*/

/\*(4)Chapter \*/

/\*(5)SubChapter \*/

/\*(6)CLEAN\_INLIER\_ACT \*/

/\*(7)CLEAN\_INLIER\_TC\_with\_AandE \*/

/\*(8)CLEAN\_EBD\_ACT \*/

/\*(9)CLEAN\_EBD\_TC \*/

/\*(10)CLEAN\_TC \*/

/\*(3)CLEAN\_INLIER\_ACT \*/

/\*(4)CLEAN\_INLIER\_TC\_with\_AandE \*/

/\*(5)CLEAN\_EBD\_ACT \*/

/\*(6)CLEAN\_EBD\_TC \*/

/\*(7)CLEAN\_TC \*/

/\*--------------------------------------------------------------\*/

/\*---\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*---\*/

/\*PLEASE DO NOT MAKE CHANGE AFTER THIS LINE \*/

/\*---\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*---\*/

/\*--02\_Base\_APC \*/

/\* step1. Calculating A\_E inlier costs proportions \*/

**data** t\_HES\_Adm\_AE;

set &SQLViewLib.**.**'HES\_Adm\_A&E'n;

where run\_ID=&param\_HESRunID.;

**run**;

/\*--01a: A\_E Proportions by HRG \*/

**proc** **sql** noprint;

create table &StageLib.**.A**PC02\_01a as

select

FCE\_HRG,

NE\_FCE,

A\_E\_FCE,

Case

when NE\_FCE =**0** then **0**

when NE\_FCE =**.** then **0**

Else A\_E\_FCE/NE\_FCE

End as AandE\_Pct

from

t\_HES\_Adm\_AE

order by FCE\_HRG

;

**quit**;

/\*--01b: Total A\_E FCE Costs\*/

**proc** **sql** noprint;

Create table &StageLib.**.A**PC02\_01b as

select distinct

a.HRG,

a.ADMISSION,

b.AandE\_Pct,

CASE WHEN (a.ADMISSION NOT LIKE 'NE' or b.AandE\_Pct Is Null )THEN **0**

ELSE a.CLEAN\_INLIER\_TC\*b.AandE\_Pct

END as AandE\_TC

from &StageLib.**.A**PC\_HRGLevelData as a

left join

&StageLib.**.A**PC02\_01a as b

on a.HRG=b.FCE\_HRG

order by a.HRG, a.ADMISSION

;

**quit**;

/\*--01c: Total Cost -\*/

**proc** **sql** noprint;

Create table &StageLib.**.A**PC02\_01c as

select

sum(AandE\_TC) as AandE\_TC

from &StageLib.**.A**PC02\_01b

;

**quit**;

/\*Step2. A\_E costs allocation to HRGS \*/

/\*01d- apportion AandE Addmission Costs \*\*/

**proc** **sql** noprint;

create table APC\_AandE\_Adm\_1

as

SELECT

a.\*,

coalesce(a.CLEAN\_INLIER\_TC\_with\_AandE,**0**)+coalesce(a.CLEAN\_EBD\_TC,**0**) AS CLEAN\_TC

FROM

(SELECT

hrg.HRG,

hrg.ADMISSION,

hrg.CLEAN\_INLIER\_ACT,

CASE WHEN hrg.ADMISSION ='NE'

THEN hrg.CLEAN\_INLIER\_TC+(in\_02.AE\_AD\_TC \* (T01b.AandE\_TC/ T01C.AandE\_TC))

ELSE hrg.CLEAN\_INLIER\_TC

END as CLEAN\_INLIER\_TC\_with\_AandE ,

hrg.CLEAN\_EBD\_ACT ,

hrg.CLEAN\_EBD\_TC

FROM

ModelOut.AE\_ADM\_COST\_QUANTUM2APC(where =(year="&FYear.")) as in\_02,

&StageLib.**.A**PC02\_01c AS T01C ,

&StageLib.**.A**PC\_HRGLevelData as hrg

INNER JOIN &StageLib.**.A**PC02\_01b AS T01b

ON (hrg.ADMISSION = T01b.ADMISSION)

AND (hrg.HRG = T01b.HRG)

) AS a

ORDER BY a.HRG, a.ADMISSION

;

**quit**;

**proc** **sql** noprint;

create table &StageLib.**.A**PC\_AandE\_Adm as

select distinct

catx("\_",HRG,ADMISSION) as HRG\_POD informat $20. format $20. length **20**,

HRG,

ADMISSION,

substr(HRG,**1**,**1**) as Chapter,

substr(HRG,**1**,**2**) as SubChapter,

CLEAN\_INLIER\_ACT,

CLEAN\_INLIER\_TC\_with\_AandE,

CLEAN\_EBD\_ACT,

CLEAN\_EBD\_TC,

CLEAN\_TC,

(CLEAN\_INLIER\_TC\_with\_AandE / CLEAN\_INLIER\_ACT) as Clean\_Inlier\_UC\_with\_AandE

from APC\_AandE\_Adm\_1

order by HRG\_POD, HRG,ADMISSION,Chapter,SubChapter

;

**quit**;

/\*output final dataset to csv file \*/

%let StepNo=Step2\_5;

%let outputfile1=APC\_AandE\_Adm;

**proc** **export** data=&StageLib.**.**&outputfile1.

outfile="&outputpath.\&StepNo.\_&outputfile1..csv"

dbms=csv

replace;

**run**;

/\*Step 2.6 NICE Technology Appraisals \*/

/\*--------------------------------------------------------------\*/

/\* Step2\_6\_NICE\_Technology\_Appraisals \*/

/\* Version: 1.01 \*/

/\* Coding: PID \*/

/\* Date: August 2015 \*/

/\* -------------------------------------------------------------\*/

/\* Input: \*/

/\* (1)Stage.APC\_A&E\_Adm (from Step3\_5) \*/

/\* (2)Input.NICE\_Costs \*/

/\* (3)Input.NICE\_HRGs \*/

/\* \*/

/\* Output: \*/

/\* \*/

/\* Stage.APC\_NICE \*/

/\*--------------------------------------------------------------\*/

/\* \*/

/\* Output variables from Stage.APC\_NICE: \*/

/\*(1)HRG\_POD \*/

/\*(2)HRG \*/

/\*(3)ADMISSION \*/

/\*(4)Chapter \*/

/\*(5)SubChapter \*/

/\*(6)CLEAN\_INLIER\_ACT \*/

/\*(7)CLEAN\_EBD\_ACT \*/

/\*(8)CLEAN\_INLIER\_TC\_WITH\_NICE \*/

/\*(9)CLEAN\_EBD\_TC \*/

/\*(10)CLEAN\_TC \*/

/\*--------------------------------------------------------------\*/

/\* 02a: NICE - IP Activity \*/

**proc** **sql** noprint;

create table &StageLib.**.A**PC02\_02a as

SELECT distinct

HRG,

Sum(CASE ADMISSION

WHEN 'EL' THEN CLEAN\_INLIER\_ACT

ELSE **0**

END ) AS EL\_ACT,

Sum(CASE ADMISSION

WHEN 'NE' THEN CLEAN\_INLIER\_ACT

ELSE **0**

END ) AS NE\_ACT,

Sum(CASE ADMISSION

WHEN 'DC' THEN CLEAN\_INLIER\_ACT

ELSE **0**

END ) AS DC\_ACT

FROM &StageLib.**.A**PC\_AandE\_Adm

GROUP BY HRG

;

**quit**;

/\*

--02b: NICE - HRG Activity split by Admission

\*/

**proc** **sql** noprint;

create table &StageLib.**.A**PC02\_02b as

SELECT

A.\*,

CASE TOTAL\_ACT

WHEN **0** THEN **0**

ELSE EL\_ACT/TOTAL\_ACT

END AS EL\_PCT,

CASE TOTAL\_ACT

WHEN **0** THEN **0**

ELSE NE\_ACT/TOTAL\_ACT

END AS NE\_PCT,

CASE TOTAL\_ACT

WHEN **0** THEN **0**

ELSE DC\_ACT/TOTAL\_ACT

END AS DC\_PCT

FROM

(SELECT

HRG,

EL\_ACT,

NE\_ACT,

DC\_ACT,

COALESCE(EL\_ACT,**0**)+COALESCE(NE\_ACT,**0**)+COALESCE(DC\_ACT,**0**) AS TOTAL\_ACT

FROM &StageLib.**.A**PC02\_02a

) AS a

order by HRG

;

**quit**;

/\*--02c: NICE - Activity by adjustment type \*/

**proc** **sql** noprint;

create table &StageLib.**.A**PC02\_02c as

SELECT

in\_03a.Type,

Sum(T02b.TOTAL\_ACT) AS TOTAL\_ACT\_FOR\_TYPE

FROM

Tariff\_R.NICE\_HRGs(where=(Year="&Year.")) as in\_03a

INNER JOIN &StageLib.**.A**PC02\_02b AS T02b

ON in\_03a.HRG = T02b.HRG

GROUP BY

in\_03a.Type

;

**quit**;

/\*--02d: NICE - Split costs by HRG\*/

**proc** **sql** noprint;

create table &StageLib.**.A**PC02\_02d as

SELECT distinct

T02b.HRG,

T02b.TOTAL\_ACT,

T02c.TOTAL\_ACT\_FOR\_TYPE,

TOTAL\_ACT/TOTAL\_ACT\_FOR\_TYPE AS ACT\_PCT,

(TOTAL\_ACT/TOTAL\_ACT\_FOR\_TYPE)\*Inclusion AS NICE\_COST

FROM

&StageLib.**.A**PC02\_02c AS T02c

INNER JOIN Tariff\_R.NICE\_HRGs as in\_03a

ON T02c.Type = in\_03a.Type

INNER JOIN &StageLib.**.A**PC02\_02b AS T02b

ON in\_03a.HRG = T02b.HRG

INNER JOIN Tariff\_R.NICE\_Costs AS in\_03b

ON in\_03b.Type = in\_03b.Type

where in\_03a.Year="&Year." and in\_03b.Year="&Year."

order by T02b.HRG

;

**quit**;

/\*--02e: NICE - Apportion costs by HRG and Admission\*/

**proc** **sql** noprint;

create table &StageLib.**.A**PC02\_02e

as

SELECT

T02d.HRG,

NICE\_COST\*EL\_PCT AS EL\_NICE,

NICE\_COST\*NE\_PCT AS NE\_NICE,

NICE\_COST\*DC\_PCT AS DC\_NICE

FROM

&StageLib.**.A**PC02\_02b AS T02b

INNER JOIN &StageLib.**.A**PC02\_02d AS T02d

ON T02b.HRG = T02d.HRG

order by T02d.HRG

;

**quit**;

/\*--02f: Total Inlier Costs inc NICE\*/

**proc** **sql** noprint;

create table &StageLib.**.A**PC02\_02f as

SELECT

a.\*,

COALESCE (CLEAN\_INLIER\_TC\_WITH\_AandE, **0**) + COALESCE(NICE\_APP,**0**) AS CLEAN\_INLIER\_TC\_WITH\_NICE

FROM (

SELECT

out\_01.HRG,

out\_01.ADMISSION,

out\_01.CLEAN\_INLIER\_ACT,

out\_01.CLEAN\_INLIER\_TC\_WITH\_AandE,

CASE WHEN T02e.HRG IS NULL THEN **0**

ELSE CASE

WHEN out\_01.ADMISSION = 'EL' THEN COALESCE( T02e.EL\_NICE,**0**)

WHEN out\_01.ADMISSION = 'NE' THEN COALESCE( T02e.NE\_NICE,**0**)

WHEN out\_01.ADMISSION= 'DC' THEN COALESCE( T02e.DC\_NICE,**0**)

END

END as NICE\_APP ,

out\_01.CLEAN\_EBD\_ACT,

out\_01.CLEAN\_EBD\_TC

FROM

&StageLib.**.A**PC\_AandE\_Adm AS out\_01

LEFT JOIN &StageLib.**.A**PC02\_02e AS T02e

ON out\_01.HRG = T02e.HRG

) AS A

ORDER BY HRG, ADMISSION

;

**quit**;

/\*--02g: Total Costs inc NICE\*/

**proc** **sql** noprint;

create table APC\_NICE\_1 as

SELECT

HRG,

ADMISSION,

CLEAN\_INLIER\_ACT,

CLEAN\_INLIER\_TC\_WITH\_NICE,

CLEAN\_EBD\_ACT,

CLEAN\_EBD\_TC,

coalesce(CLEAN\_INLIER\_TC\_WITH\_NICE,**0**)+coalesce(CLEAN\_EBD\_TC,**0**) AS CLEAN\_TC

FROM &StageLib.**.A**PC02\_02f

ORDER BY HRG,ADMISSION

;

**quit**;

**proc** **sql** noprint;

create table &StageLib.**.A**PC\_NICE as

select

catx("\_",HRG,ADMISSION) as HRG\_POD informat $20. format $20. length **20**,

HRG,

ADMISSION,

substr(HRG,**1**,**1**) as Chapter,

substr(HRG,**1**,**2**) as SubChapter,

CLEAN\_INLIER\_ACT,

CLEAN\_INLIER\_TC\_WITH\_NICE,

CLEAN\_EBD\_ACT,

CLEAN\_EBD\_TC,

CLEAN\_TC,

(CLEAN\_INLIER\_TC\_WITH\_NICE / CLEAN\_INLIER\_ACT) as Clean\_inlier\_UC\_with\_Nice

from APC\_NICE\_1

order by HRG\_POD,HRG,ADMISSION,Chapter,SubChapter

;

**quit**;

/\*output final dataset to csv file \*/

%let StepNo=Step2\_6;

%let outputfile1=APC\_NICE;

**proc** **export** data=&StageLib.**.**&outputfile1.

outfile="&outputpath.\&StepNo.\_&outputfile1..csv"

dbms=csv

replace;

**run**;

/\* Step2.7 FCE to Spell Cost Conversion \*/

/\*--------------------------------------------------------------\*/

/\* Step2\_7\_FCE\_to\_Spell\_Cost\_Conversion \*/

/\* Version: 1.01 \*/

/\* Coding: PID \*/

/\* Date: August 2015 \*/

/\* -------------------------------------------------------------\*/

/\* Input: \*/

/\* (1)Stage.APC\_NICE (from Step3\_6) \*/

/\* (2)Stabe.APC\_FCE\_UC\_incEBD \*/

/\* (3)Input.TrimpointsRC\_HRGs \*/

/\* (4)View.HES\_FCE\_Spell\_Map\_LoS \*/

/\* (5)View.HES\_Spell\_Counts\_Adm \*/

/\* (6)Input.HRGsEligibility \*/

/\* Output: \*/

/\* \*/

/\* Stage.APC02\_03C \*/

/\*--------------------------------------------------------------\*/

/\* \*/

/\* Output variables from Stage.APC02\_03C: \*/

/\*(1)SPELL\_HRG \*/

/\*(2)ADM \*/

/\*(3)SPELL\_TC \*/

/\*(4)SPELL\_ACT \*/

/\*--------------------------------------------------------------\*/

\*Note: param\_HESRunID moved to initialise parameters;

\*%let param\_HESRunID=10;

/\*--03a: FCE Level Unit Costs (inc EBDs) \*/

**proc** **sql** noprint;

create table &StageLib.**.A**PC\_FCE\_UC\_incEBD

as

SELECT

HRG,

ADMISSION,

CLEAN\_INLIER\_ACT,

CLEAN\_INLIER\_TC\_WITH\_NICE,

CLEAN\_EBD\_TC,

CLEAN\_TC,

CASE

WHEN CLEAN\_INLIER\_ACT IS NULL OR CLEAN\_INLIER\_ACT = **0** THEN **0**

ELSE (COALESCE(CLEAN\_INLIER\_TC\_WITH\_NICE,**0**) / COALESCE(CLEAN\_INLIER\_ACT,**0**))

END as UC\_exc\_EBD,

CASE

WHEN CLEAN\_INLIER\_ACT IS NULL OR CLEAN\_INLIER\_ACT = **0** THEN **0**

ELSE CLEAN\_TC / COALESCE (CLEAN\_INLIER\_ACT,**0**)

END as UC\_inc\_EBD,

CASE

WHEN CLEAN\_EBD\_TC IS NULL OR CLEAN\_EBD\_TC = **0** OR CLEAN\_INLIER\_ACT IS NULL OR CLEAN\_INLIER\_ACT = **0**

THEN **0**

ELSE CLEAN\_EBD\_TC/CLEAN\_EBD\_ACT

END as EBD\_UC

FROM &StageLib.**.A**PC\_NICE

ORDER BY HRG, ADMISSION

;

**quit**;

/\*--03b: FCE to Spell Conversion\*/

**proc** **sql** noprint;

Create table &StageLib.**.A**PC02\_03b as

select

in\_09c.SPELL\_HRG ,

in\_09c.ADM,

Sum(in\_09c.FCE\_ACTIVITY) AS FCE\_ACT,

Sum(in\_09c.FCE\_ACTIVITY\*out\_03.UC\_exc\_EBD) AS SPELL\_INLIER,

Sum(

CASE WHEN ADM <> 'DC' AND in\_10.FCE\_trim < in\_09c.EPI\_LoS

THEN (in\_09c.EPI\_LoS -in\_10.FCE\_trim)\* in\_09c.FCE\_ACTIVITY\* out\_03.EBD\_UC

ELSE **0** END) as HES\_FCE\_EBD

FROM

Tariff\_R.TrimpointsRC\_HRGs(where=(Year="&Year." )) AS in\_10

INNER JOIN &SQLViewLib.**.H**ES\_FCE\_Spell\_Map\_LoS (where=( Run\_ID = &param\_HESRunID.)) as in\_09c

ON (in\_10.HRG = in\_09c.FCE\_HRG)

LEFT JOIN &StageLib.**.A**PC\_FCE\_UC\_incEBD as out\_03

ON ((in\_09c.FCE\_HRG = out\_03.HRG) AND (in\_09c.ADM = out\_03.ADMISSION))

GROUP BY

in\_09c.SPELL\_HRG,

in\_09c.ADM

HAVING

(((in\_09c.SPELL\_HRG) Is Not Null)

AND ((in\_09c.ADM)<>'UX'))

ORDER BY

in\_09c.SPELL\_HRG,

in\_09c.ADM;

**quit**;

**proc** **sql** noprint;

create table &StageLib.**.A**PC02\_03C as

SELECT

in\_09d.SPELL\_HRG ,

CASE

WHEN T03b.ADM = 'DC' THEN 'EL'

ELSE T03b.ADM

END as ADM,

Sum(coalesce(T03b.SPELL\_INLIER,**0**)+ coalesce(HES\_FCE\_EBD,**0**)) AS SPELL\_TC,

Sum(coalesce(in\_09d.SPELLFLAG,**0**)) AS SPELL\_ACT

FROM

Tariff\_R.HRG\_ELIGIBILITY as in\_05

INNER JOIN &StageLib.**.A**PC02\_03b AS T03b

ON (in\_05.HRG = T03b.SPELL\_HRG)

INNER JOIN &SQLViewLib.**.H**ES\_Spell\_Counts\_Adm(where=( Run\_ID = &param\_HESRunID.)) as in\_09d

ON (T03b.ADM = in\_09d.ADM AND T03b.SPELL\_HRG = in\_09d.SPELL\_HRG)

WHERE

in\_05.Combined\_day\_case\_\_\_ordinary\_ele=**1**

AND in\_05.Year="&FYear."

GROUP BY

in\_09d.SPELL\_HRG,

CASE

WHEN T03b.ADM = 'DC' THEN 'EL'

ELSE T03b.ADM

END

;

**quit**;

/\*--03cc: Spell Costs and Activity ---\*/

**proc** **sql** noprint;

create table

&StageLib.**.A**PC\_DcEl\_Split

as

SELECT

in\_09d.SPELL\_HRG as SPELL\_HRG,

T03b.ADM,

Sum(coalesce(T03b.SPELL\_INLIER,**0**)+ coalesce(T03b.HES\_FCE\_EBD,**0**) )AS SPELL\_TC,

Sum(coalesce(in\_09d.SPELLFLAG,**0**)) AS SPELL\_ACT

FROM

Tariff\_R.HRGSELIGIBILITY\_TEMP as in\_05

INNER JOIN &StageLib.**.A**PC02\_03b AS T03b

ON (in\_05.HRG\_Code = T03b.SPELL\_HRG)

INNER JOIN &SQLViewLib.**.H**ES\_Spell\_Counts\_Adm as in\_09d

ON (T03b.ADM = in\_09d.ADM AND T03b.SPELL\_HRG = in\_09d.SPELL\_HRG)

WHERE

in\_05.Mandatory\_IP=**1** and in\_05.Year="&Year." AND in\_09d.Run\_ID = &param\_HESRunID.

GROUP BY

in\_09d.SPELL\_HRG,T03b.ADM

;

**quit**;

/\*--04a: Spell Maketable \*/

**proc** **sql** noprint;

create table &StageLib.**.A**PC\_SpellData as

SELECT

distinct

SPELL\_HRG,

ADM,

SPELL\_TC,

SPELL\_ACT

FROM &StageLib.**.A**PC02\_03C

ORDER BY SPELL\_HRG, ADM

;

**quit**;

/\*output final dataset to csv file \*/

%let StepNo=Step2\_7;

%let outputfile1=APC\_SpellData;

**proc** **export** data=&StageLib.**.**&outputfile1.

outfile="&outputpath.\&StepNo.\_&outputfile1..csv"

dbms=csv

replace;

**run**;

/\*Step 2.8 Calculate Trim Points \*/

/\*--------------------------------------------------------------\*/

/\* Step2\_8\_Calculate\_TrimPoints \*/

/\* Version: 1.02 \*/

/\* Coding: PID \*/

/\* Date: August 2015 by PID \*/

/\* Code Updated: August 2017 -- for [Stage].APC02\_04c \*/

/\* -------------------------------------------------------------\*/

/\* Input: \*/

/\* (1)Stage.APC\_SpellData (from Step2\_7) \*/

/\* (2)Stage.APC\_HRGLevelData; \*/

/\* (3)View.V\_HES\_Count\_Spell\_LoS; \*/

/\* (4)View.V\_HES\_Trimpoints \*/

/\* \*/

/\* Output: \*/

/\* (1) Stage.APC\_EBD \*/

/\* (2) Stage.APC\_EBD\_UC \*/

/\* \*/

/\*--------------------------------------------------------------\*/

/\* \*/

/\* Output variables from Stage.APC\_EBD: \*/

/\*(1)SPELL\_HRG \*/

/\*(2)ADM \*/

/\*(3)SPELL\_ACT \*/

/\*(4)SPELL\_TC \*/

/\*(5)EBD\_TC \*/

/\*(6)INLIER\_TC \*/

/\* \*/

/\* Output variables from Stage.APC\_EBD\_UC: \*/

/\*(1)HRG \*/

/\*(2)FCE\_EBD\_UC\_CAPPED \*/

/\*--------------------------------------------------------------\*/

\*Note: param\_HESRunID moved to initialise parameters;

;

/\*--04b: Crosstab Activity \*/

**proc** **sql** noprint;

create table APC\_SPellData\_4Trans as

select distinct

SPELL\_HRG,

ADM,

SPELL\_ACT

from &StageLib.**.A**PC\_SpellData

quit;

/\* \*--------> diffrent method used (proc transpose ) vs pivot in SQL server \*/

**proc** **transpose** data=APC\_SPellData\_4Trans out=&StageLib.**.A**PC02\_04b (drop=\_name\_);

by SPELL\_HRG;

id ADM;

**run**;

**proc** **sql** noprint;

create table &StageLib.**.A**PC02\_04c

as

SELECT

SPELL\_HRG,

EL,

NE,

CASE

WHEN coalesce(EL,**0**)^=**0** and coalesce(EL,**0**) < **50** THEN **1**

ELSE

CASE

WHEN coalesce(NE,**0**)^=**0** and coalesce(NE,**0**)<**50** THEN **1**

ELSE

CASE

WHEN sum(coalesce(NE,**0**), coalesce(EL,**0**)) < **150** THEN **1**

ELSE **0**

END

END

END as Combine

FROM &StageLib.**.A**PC02\_04b

order by SPELL\_HRG

;

**quit**;

/\*--05a: EBD UC\*/

**proc** **sql** noprint;

create table

&StageLib.**.A**PC02\_05a as

SELECT

A.\*,

CASE

WHEN FCE\_EBD\_UC > **500** THEN **500**

ELSE

CASE

WHEN FCE\_EBD\_UC< **100** THEN **100**

ELSE FCE\_EBD\_UC

END

END as FCE\_EBD\_UC\_CAPPED

FROM

(SELECT distinct

substr(HRG,**1**,**1**) AS CHAPTER,

CASE

WHEN SUM(coalesce (CLEAN\_EBD\_ACT,**0**)) = **0** THEN **0**

ELSE SUM(coalesce (CLEAN\_EBD\_TC,**0**)) / SUM(coalesce (CLEAN\_EBD\_ACT,**0**))

END as FCE\_EBD\_UC

FROM &StageLib.**.A**PC\_HRGLevelData

GROUP BY substr(HRG,**1**,**1**)

) AS A

ORDER BY CHAPTER

;

**quit**;

**proc** **sql** noprint;

create table &StageLib.**.A**PC02\_05b

as

SELECT

SPELL\_HRG,

Chapter,

ADM,

TP,

SUM(

CASE

WHEN SpellLoS\_1YR > TP THEN (SpellLoS\_1YR-TP)\*SPELLFLAG

ELSE **0**

END ) as EBDs

FROM (

SELECT

in\_09b.SPELL\_HRG,

substr(in\_09e.HRG,**1**,**1**) AS Chapter,

in\_09b.ADM,

CASE

WHEN Combine=**1** THEN in\_09e.DC\_EL\_NE

ELSE

CASE WHEN strip(in\_09b.ADM) = 'EL' THEN in\_09e.DC\_EL

ELSE in\_09e.NE

END

END as TP,

in\_09b.SpellLoS\_1YR,

in\_09b.SPELLFLAG

FROM

&StageLib.**.A**PC02\_04c AS T04c

INNER JOIN &SQLViewLib.**.H**ES\_Count\_Spell\_LoS AS in\_09b

ON T04c.SPELL\_HRG = in\_09b.SPELL\_HRG

INNER JOIN &SQLViewLib.**.H**ES\_Trimpoints as in\_09e

ON in\_09b.SPELL\_HRG = in\_09e.HRG

WHERE in\_09b.Run\_ID = &param\_HESRunID. AND in\_09e.Run\_ID = &param\_HESRunID.

) AS A

GROUP BY

SPELL\_HRG,

Chapter,

ADM,

TP

HAVING

strip(ADM) <>'DC' And strip(ADM)<>'UX'

ORDER BY SPELL\_HRG, CHAPTER, ADM

;

**quit**;

/\*--05c: EBD total costs by HRG & admission \*/

**proc** **sql** noprint;

create TABLE &StageLib.**.A**PC02\_05C as

SELECT distinct

T05B.SPELL\_HRG,

T05B.ADM,

T05A.FCE\_EBD\_UC\_CAPPED,

T05B.EBDs,

FCE\_EBD\_UC\_CAPPED\*EBDs AS EBD\_TC

FROM

&StageLib.**.A**PC02\_05a AS T05A

INNER JOIN

&StageLib.**.A**PC02\_05B AS T05B

ON T05A.CHAPTER = T05B.Chapter

ORDER BY T05B.SPELL\_HRG, T05B.ADM

;

**quit**;

/\*--05d: Remove spell EBD TC from inlier TC\*/

**proc** **sql** noprint;

Create table &StageLib.**.A**PC\_EBD as

SELECT

catx("\_",out\_04.SPELL\_HRG,out\_04.ADM) as SPELL\_HRG\_POD informat $20. format $20. length **20**,

out\_04.SPELL\_HRG,

out\_04.ADM,

out\_04.SPELL\_ACT,

out\_04.SPELL\_TC,

T05C.EBD\_TC,

coalesce(SPELL\_TC,**0**)- coalesce(EBD\_TC,**0**) AS INLIER\_TC

FROM

&StageLib.**.A**PC02\_05C AS T05C

RIGHT JOIN &StageLib.**.A**PC\_SpellData AS out\_04

ON T05C.SPELL\_HRG = out\_04.SPELL\_HRG AND T05C.ADM = out\_04.ADM

ORDER BY out\_04.SPELL\_HRG,ADM

;

**quit**;

/\*--05e: EBD unit cost\*/

**proc** **sql** noprint;

create table &StageLib.**.A**PC\_EBD\_UC as

SELECT distinct

T05B.SPELL\_HRG AS HRG,

T05A.FCE\_EBD\_UC\_CAPPED

FROM

&StageLib.**.A**PC02\_05A AS T05A

INNER JOIN &StageLib.**.A**PC02\_05B AS T05B

ON T05A.CHAPTER = T05B.Chapter

Order by T05B.SPELL\_HRG, T05A.FCE\_EBD\_UC\_CAPPED

;

**quit**;

/\*output final dataset to csv file \*/

%let StepNo=Step2\_8;

%let outputfile1=APC\_EBD;

**proc** **export** data=&StageLib.**.**&outputfile1.

outfile="&outputpath.\&StepNo.\_&outputfile1..csv"

dbms=csv

replace;

**run**;

/\*output final dataset to csv file \*/

%let outputfile2=APC\_EBD\_UC;

**proc** **export** data=&StageLib.**.**&outputfile2.

outfile="&outputpath.\&StepNo.\_&outputfile2..csv"

dbms=csv

replace;

**run**;

/\*Step 2.9 Removal of costs associated with DD(Drugs and Device) \*/

/\*--------------------------------------------------------------\*/

/\* Step2\_9a\_Removal\_Costs\_with\_DD \*/

/\* Version: 1.01 \*/

/\* Coding: PID \*/

/\* Date: August 2015 \*/

/\* -------------------------------------------------------------\*/

/\* Input: \*/

/\* (1)Input.D&D; \*/

/\* (2)Stage.National\_MFF; \*/

/\* (3)Stage.APC\_Spell\_Data; \*/

/\* (4)Stage.APC\_EBD \*/

/\* \*/

/\* Output: \*/

/\* (1)Stage.APC\_Act\_Costs; \*/

/\* (2)Stage.APC\_DandD\_topslice \*/

/\* \*/

/\*--------------------------------------------------------------\*/

/\* \*/

/\* Output variables from Stage.APC\_Act\_Costs: \*/

/\*(1)SPELL\_HRG \*/

/\*(2)ADM \*/

/\*(3)SPELL\_ACT \*/

/\*(4)REVISED\_TC \*/

/\* \*/

/\*Output variables from Stage.APC\_DandD\_topslice: \*/

/\*(1)LIMIT\_TS \*/

/\*(2)DandD\_TS \*/

/\* \*/

/\*--------------------------------------------------------------\*/

\*Note: param\_HESRunID moved to initialise parameters;

**proc** **sql** noprint;

create table &StageLib.**.A**PC02\_07a as

SELECT distinct

HRG\_Code,

CASE WHEN Admission='DC' THEN 'EL'

ELSE Admission

END as ADM ,

Sum(DandD\_Total) AS DandD\_GROUPED

FROM Tariff\_R.D\_and\_D

where Year="&Year."

GROUP BY

HRG\_Code, ADM

order by

HRG\_Code, ADM

;

**quit**;

/\*--07b: Remove MFF from D&D \*/

**proc** **sql** noprint;

select NAT\_MFF informat **20.9** format **20.9**

into :NAT\_MFF

from &StageLib.**.N**ational\_MFF

;

**quit**;

%let NAT\_MFF=&NAT\_MFF;

%put NAT\_MFF=&NAT\_MFF;

**proc** **sql** noprint;

create table &StageLib.**.A**PC02\_07b as

SELECT

T07A.HRG\_Code,

T07A.ADM,

(DandD\_GROUPED/&NAT\_MFF.) AS DandD\_exc\_MFF

FROM

&StageLib.**.A**PC02\_07a AS T07A

order by HRG\_Code, ADM

;

**quit**;

/\*--07c: Limit Cost Removal to 50% \*/

**proc** **sql** noprint;

create table &StageLib.**.A**PC02\_07C

as

SELECT distinct

A.\*,

CASE WHEN ADM = 'OP' THEN TC\_exc\_DandD

ELSE

CASE WHEN TC\_exc\_DandD<Cal\_50Pct\_TC THEN Cal\_50Pct\_TC

ELSE TC\_exc\_DandD

END

END as LIMITED\_TC ,

CASE WHEN TC\_exc\_DandD <Cal\_50Pct\_TC THEN **1**

ELSE **0**

END as FLAG

FROM

(

SELECT

out\_04.SPELL\_HRG,

out\_04.ADM,

out\_04.SPELL\_ACT,

T07b.DandD\_exc\_MFF,

out\_05a.EBD\_TC,

out\_04.SPELL\_TC,

coalesce(INLIER\_TC,**0**)-coalesce(DandD\_exc\_MFF,**0**) AS TC\_exc\_DandD,

(out\_04.SPELL\_TC/**2**) AS Cal\_50Pct\_TC

FROM

&StageLib.**.A**PC\_SpellData as out\_04

LEFT JOIN &StageLib.**.A**PC02\_07b AS T07b

ON out\_04.ADM = T07b.ADM AND out\_04.SPELL\_HRG = T07b.HRG\_Code

LEFT JOIN &StageLib.**.A**PC\_EBD as out\_05a

ON out\_04.ADM = out\_05a.ADM AND out\_04.SPELL\_HRG = out\_05a.SPELL\_HRG

) AS A

ORDER BY SPELL\_HRG, ADM

;

**quit**;

/\*--07d: Output from Limited D&D \*/

**proc** **sql** noprint;

create TABLE &StageLib.**.A**PC\_Act\_Costs as

SELECT

catx("\_",SPELL\_HRG,ADM) as SPELL\_HRG\_POD informat $20. format $20. length **20**,

SPELL\_HRG,

ADM,

SPELL\_ACT,

LIMITED\_TC AS REVISED\_TC

FROM &StageLib.**.A**PC02\_07C

order by SPELL\_HRG, ADM

;

**quit**;

/\* --07e: Calculate Limiting Topslice \*/

**proc** **sql** noprint;

create table &StageLib.**.A**PC02\_07e as

SELECT

Sum(LIMITED\_TC)-(Sum(SPELL\_TC)-Sum(EBD\_TC)-Sum(DandD\_exc\_MFF)) AS DandD\_Limit\_Slice

FROM &StageLib.**.A**PC02\_07C

;

**quit**;

/\*--07f: D&D Topslices \*/

/\*preparing the data \*/

**proc** **sql** noprint;

select TOTAL into :TOTAL

FROM Tariff\_R.D\_and\_D\_topslice

where Year="&Year."

;

**quit**;

%let total=&total;

%put total=&total;

**proc** **sql** noprint;

create table &StageLib.**.A**PC\_DandD\_topslice as

SELECT

T07e.DandD\_Limit\_Slice AS LIMIT\_TS,

(&TOTAL./&NAT\_MFF.) AS DandD\_TS

FROM &StageLib.**.A**PC02\_07e as T07e

;

**quit**;

%let StepNo=Step2\_9;

%let outputfile1=APC\_Act\_Costs;

**proc** **export** data=&StageLib.**.**&outputfile1.

outfile="&outputpath.\&StepNo.\_&outputfile1..csv"

dbms=csv

replace;

**run**;

**data** Chemo\_pre\_qr1\_44; set Chemo\_pre\_qr1\_4;

prices\_pre\_qr1=price\_weighted;

drop price\_weighted;

quantum=prices\_pre\_qr1\*total\_activity;

**run**;

**data** Radio\_pre\_qr66; set Radio\_pre\_qr6;

prices\_pre\_qr1=final\_prices\_neutral;

drop final\_prices\_neutral;

quantum=prices\_pre\_qr1\*total\_activity;

**run**;

**data** output\_Total\_Act\_Cost\_D\_Im; set output\_Total\_Act\_Cost\_D\_Im; **run**;

**data** t1 (keep=currency quantum); set Chemo\_pre\_qr1\_44; **run**;

**data** t2 (keep=currency quantum); set Radio\_pre\_qr66; **run**;

**data** t3; set output\_Total\_Act\_Cost\_D\_Im; quantum=total\_cost; keep currency quantum; **run**;

**data** t4;

set t3 t1 t2;

**run**;

**proc** **sql**; create table unb\_pre\_qr1 as select a.\*, b.hrg as hrg\_code from t4 a

right join work.hrg\_eligibility as b on a.currency=b.hrg

where b.Unbundled\_\_tariff\_\_including\_cos=**1**;

**quit**;

**proc** **sql**; create table t5 as select "Unbundled" FORMAT=$20. as POD, sum(quantum) as quantum4qr from unb\_pre\_qr1;

**quit**;

**data** xxx; set work.QUANTUMRECONCILATION\_OUTPUT; **run**;

**PROC** **SQL**;

CREATE TABLE B2

AS SELECT A.\*, b.quantumrecvalues, (b.quantumrecvalues/A.quantum4qr)-**1** as qr\_factor1

FROM t5 AS A

LEFT JOIN xxx AS B

ON strip(upcase(A.POD))=strip(upcase(B.POD))

where B.pod="UNBUNDLED"

**QUIT**;

/\*UNBUNDLED\*/

**proc** **sql**;

select qr\_factor1 into :qr1\_factor

from b2

;

**quit**;

%let rc\_qr1\_factor=&qr1\_factor.;

/\* DIAGNOSTIC IMAGING \*/

/\* -- Filter Non\_Acute Table \*/

/\*1718 Changes 1 - Codes DIAGIMDA, DIAGIMOP, DIAGIMOTH in RC1314 have been replaced with IMAGDA, IMAGOP, IMAGOTH respectively in RC1415.\*/

/\* --Removed Service from Output \*/

**PROC** **SQL**;

CREATE TABLE stage\_Imaging\_Scope

AS SELECT organisationcode as FK\_ORGS\_PROV\_ID

,'IMAG' as DEPARTMENTcode

,CASE WHEN rcna.DEPARTMENTcode = 'IMAGDA' THEN 'DA'

WHEN rcna.DEPARTMENTcode= 'IMAGOP' THEN 'OP'

ELSE 'Remove' END AS SERVICEcode

,CURRENCYcode

,UNITCOST as UC

,activity as ACTIVITY\_P1

,(UNITCOST\*ACTIVITY) as TOTAL\_COST

from work.Reference\_Cost rcna

WHERE (rcna.DEPARTMENTcode='IMAGDA' or rcna.DEPARTMENTcode='IMAGOP') AND

rcna.SUPPLIERTYPE='OWN';

**QUIT**;

/\*--- Clean Data > ( NA \* 20 < org or NA/20 > org) \*/

**PROC** **SQL**;

CREATE TABLE Clean\_Data

AS SELECT

MFF\_Adjusted.\*,

CASE WHEN UC\_Target\_MFF/NA<**0.05** OR UC\_Target\_MFF/NA>**20**

THEN **1** ELSE **0** END AS anomaly

FROM stage\_Imaging\_Scope\_MFF\_Adj AS MFF\_Adjusted INNER JOIN National\_Average ON

MFF\_Adjusted.CURRENCYcode = National\_Average.CURRENCY AND

MFF\_Adjusted.DEPARTMENTcode = National\_Average.DEPARTMENT AND

MFF\_Adjusted.Servicecode = National\_Average.Service

WHERE

(CASE WHEN UC\_Target\_MFF/NA<**0.05** OR UC\_Target\_MFF/NA>**20**

THEN **1** ELSE **0** END) = **0**;

**QUIT**;

/\* -- National Average MFF adjustment \*/

**proc** **sql**;

CREATE TABLE MFF\_IMPACT

as SELECT SUM(TOTAL\_COST) as INC\_MFF, SUM(TC\_Target\_MFF) as EXEC\_MFF, SUM(TOTAL\_COST)/SUM(TC\_Target\_MFF) as NA\_MFF\_ADJ

FROM stage\_Imaging\_Scope\_MFF\_Adj;

**quit**;

/\* -- MFF RESCALE \*/

**proc** **sql**;

CREATE TABLE MFF\_RESCALE

as SELECT SUM(TC\_Target\_MFF) as EXEC\_MFF, SUM(TC\_Payment\_MFF) as EXEC\_MFF\_CAPPED,

SUM(TC\_Payment\_MFF)/SUM(TC\_Target\_MFF) as MFF\_RESCALE

FROM stage\_Imaging\_Scope\_MFF\_Adj;

**quit**;

/\* ---- National Average \*/

**PROC** **SQL**;

CREATE TABLE National\_Average

AS SELECT DEPARTMENTcode as DEPARTMENT, SERVICEcode as SERVICE, CURRENCYcode as CURRENCY, Sum(UC\_Target\_MFF\*ACTIVITY\_P1)/Sum(ACTIVITY\_P1) AS NA

FROM stage\_Imaging\_Scope\_MFF\_Adj

GROUP BY DEPARTMENT, Service, CURRENCY;

**QUIT**;

/\* -- Recreate National Average (combining DA/OP SERVICES HERE) \*/

**PROC** **SQL**;

CREATE TABLE NATIONAL\_DATA\_CLEAN

as SELECT

DEPARTMENTcode as department,

CURRENCYcode as currency,

SUM(ACTIVITY\_p1) as Total\_Activity, Sum(UC\_Target\_MFF\*ACTIVITY\_p1) as TOTAL\_COST,

SUM(UC\_Target\_MFF\*ACTIVITY\_p1)/Sum(ACTIVITY\_p1) AS HRG\_UC\_Target\_MFF

FROM Clean\_Data

GROUP BY

DEPARTMENT,

CURRENCY;

**quit**;

/\* -- Generate Tariff \*/

/\* ---\* 0910 Unit cost\*/

**proc** **sql**;

CREATE TABLE output\_Total\_Act\_Cost\_D\_Im

as SELECT distinct nt.DEPARTMENT, nt.CURRENCY, nt.TOTAL\_ACTIVITY format=COMMA15.0, nt.TOTAL\_COST format=COMMA15.2,

ROUND(nt.HRG\_UC\_Target\_MFF,**.01**) as HRG\_UC,

HRG\_UC\_Target\_MFF\*MFF\_RESCALE format=comma8. as TARIFF

FROM MFF\_RESCALE, work.Uplift, NATIONAL\_DATA\_CLEAN nt

ORDER BY DEPARTMENT, CURRENCY;

**quit**;

**PROC** **SQL**;

CREATE TABLE stage\_Imaging\_Scope\_MFF\_Adj

AS SELECT stage\_Imaging\_Scope.\*, UC/p.UNCAPPED\_MFF AS UC\_TARGET\_MFF, TOTAL\_COST/p.UNCAPPED\_MFF AS TC\_TARGET\_MFF, TOTAL\_COST/p.Capped\_MFF AS TC\_PAYMENT\_MFF

FROM stage\_Imaging\_Scope INNER JOIN work.tc\_mff p ON stage\_Imaging\_Scope.FK\_ORGS\_PROV\_ID = p.RC\_Code;

**QUIT**;

Libname PostMA base "\\irnarch\sas\_data\Tariff Rebuild\Model outputs pre\_MA\unbundled";

**DATA** MA\_unbundled\_Input;

SET PostMA.MA\_unbundled\_Input;

**RUN**;

**DATA** MA\_UNBUNDLED\_Output;

SET PostMA.MA\_UNBUNDLED\_Output;

**RUN**;

**proc** **sql**;

create table new1 as select a.\*, B.HRG\_CODE,B.PRICE\_DECISION,c.PostMA\_PRICE from Unb\_4IA\_DI\_Chemo\_Radio\_Prices as a

left join MA\_UNBUNDLED\_Input as b

on a.HRG\_CODE=b.HRG\_CODE

LEFT JOIN MA\_UNBUNDLED\_Output

as c

on

a.HRG\_CODE=c.HRG\_CODE;

**quit**;

**data** new2;

set new1;

IF currency="" then currency=hrg\_code; else currency=currency;

Unb\_Price\_ID=**3**;

Year="19/20";

if price\_decision="Adjusted"

then Tariff=Tariff; ELSE Tariff=Tariff;

**run**;

**data** new3;

set new2 (DROP=PRICE\_DECISION POSTMA\_PRICE HRG\_CODE);

rename currency=HRG\_CODE;

**RUN**;

**data** Chemo\_post\_qr1;

set Chemo\_pre\_qr1\_4;

rc\_qr1\_factor=&rc\_qr1\_factor.;

modelled\_TARIFF\_withQR1=Price\_weighted\*(**1**+rc\_qr1\_factor);

TOTAL\_COST\_preQR1=TOTAL\_ACTIVITY\*Price\_weighted;

TOTAL\_COST\_postQR1=TOTAL\_ACTIVITY\*modelled\_TARIFF\_withQR1;

Format modelled\_TARIFF\_withQR1 Comma7.0;

Format TOTAL\_COST\_preQR1 TOTAL\_COST\_postQR1 Comma15.2;

**run**;

**data** Chemo\_post\_qr\_ie\_cb; set Chemo\_post\_qr1;

Cost\_base\_adjustment\_factor=&Cost\_base\_adjustment\_factor.;

TOTAL\_price\_postQR1\_cb=modelled\_TARIFF\_withQR1\*(**1**+Cost\_base\_adjustment\_factor);

ie\_indextn\_adjstmt\_factors=&ie\_indextn\_adjstmt\_factors.;

final\_prices\_chemo=TOTAL\_price\_postQR1\_cb\*(**1**+ie\_indextn\_adjstmt\_factors);

format final\_prices\_chemo comma15.0;

format rc\_qr1\_factor Cost\_base\_adjustment\_factor ie\_indextn\_adjstmt\_factors percentn7.2;

**run**;

**Data** Radio\_post\_qr\_ie\_cb;

set Radio\_pre\_qr6;

rc\_qr1\_factor=&rc\_qr1\_factor.;

modelled\_tariff\_qr1=final\_prices\_neutral\*(**1**+rc\_qr1\_factor);

Cost\_base\_adjustment\_factor=&Cost\_base\_adjustment\_factor.;

prices\_after\_qr1\_cb=modelled\_tariff\_qr1\*(**1**+Cost\_base\_adjustment\_factor);

ie\_indextn\_adjstmt\_factors=&ie\_indextn\_adjstmt\_factors.;

final\_prices\_radio=prices\_after\_qr1\_cb\*(**1**+ie\_indextn\_adjstmt\_factors);

format final\_prices\_radio comma15.0;

format rc\_qr1\_factor Cost\_base\_adjustment\_factor ie\_indextn\_adjstmt\_factors percentn7.2;

**run**;

**proc** **sql**; create table Chemo\_prices as select **3** format=**8.** as Unb\_price\_id, "19/20" format=$10. as Year, department, currency as hrg\_code, final\_prices\_chemo as tariff from Chemo\_post\_qr\_ie\_cb; **quit**;

**proc** **sql**; create table RADIO\_prices as select **3** format=**8.** as Unb\_price\_id, "19/20" format=$10. as Year, department, currency as hrg\_code, final\_prices\_RADIO as tariff from RADIO\_post\_qr\_ie\_cb; **quit**;

**DATA** CHEMO\_RADIO\_PRICES;

SET Chemo\_prices Radio\_prices;

**run**;

**proc** **sql**;

create table Unb\_4IA\_Chemo\_Radio\_Prices as select a.unb\_price\_id, a.year, a.hrg\_code, b.description as hrg\_name, a.tariff from CHEMO\_RADIO\_PRICES as a

left join work.hrg\_eligibility as b

on a.hrg\_code=b.hrg

where b.Unbundled\_\_tariff\_\_including\_cos=**1**;

**quit**;

**data** Diag\_Imag\_post\_qr1;

set output\_Total\_Act\_Cost\_D\_Im;

rc\_qr1\_factor=&rc\_qr1\_factor.;

modelled\_TARIFF\_withQR1=TARIFF\*(**1**+rc\_qr1\_factor);

TOTAL\_COST\_postQR1=TOTAL\_ACTIVITY\*modelled\_TARIFF\_withQR1;

Format modelled\_TARIFF\_withQR1 Comma7.0;

Format TOTAL\_COST\_postQR1 Comma15.2;

**run**;

**data** DI\_post\_qr1\_CB\_IE; set Diag\_Imag\_post\_qr1;

Cost\_base\_adjustment\_factor=&Cost\_base\_adjustment\_factor.;

TOTAL\_price\_postQR1\_cb=modelled\_TARIFF\_withQR1\*(**1**+Cost\_base\_adjustment\_factor);

ie\_indextn\_adjstmt\_factors=&ie\_indextn\_adjstmt\_factors.;

final\_prices\_DI=TOTAL\_price\_postQR1\_cb\*(**1**+ie\_indextn\_adjstmt\_factors);

format final\_prices\_DI comma15.0;

format rc\_qr1\_factor Cost\_base\_adjustment\_factor ie\_indextn\_adjstmt\_factors percentn7.2;

**run**;

**data** unb\_t1(rename=hrg=hrg\_code); set work.hrg\_eligibility; /\* use hrg\_eligibility instead \*/

where Unbundled\_\_tariff\_\_including\_cos=**1** and (find(hrg, "RD", "i") or find(hrg, "RN", "i")); **run**;

**proc** **sql**;

create table DI\_post\_qr1\_CB\_IE2 as select b.\*,/\*hrg\_code\*/c.cost\_of\_reporting

, a.description as hrg\_name, a.hrg\_code from unb\_t1 as a

left join DI\_post\_qr1\_CB\_IE as b

on b.currency/\*hrg\_code\*/=a.hrg\_code

left join work.unbundled as c

on a.hrg\_code=c.hrg\_code;

**quit**;

**proc** **sql**; create table DI\_prices as select **3** format=**8.** as Unb\_price\_id, "19/20" format=$10. as Year, department, currency as hrg\_code, final\_prices\_DI as tariff from DI\_post\_qr1\_CB\_IE; **quit**;

**DATA** DI\_CHEMO\_RADIO\_PRICES;

SET DI\_PRICES CHEMO\_RADIO\_PRICES;

currency=hrg\_code;

drop hrg\_code;

**RUN**;

**proc** **sql**;

create table Unb\_4IA\_DI\_Chemo\_Radio\_Prices as select b.unb\_price\_id, b.year, b.currency/\*hrg\_code\*/, a.description as hrg\_name, b.tariff, c.cost\_of\_reporting, a.hrg as hrg\_code from work.hrg\_eligibility as a

left join DI\_CHEMO\_RADIO\_PRICES as b /\* use hrg\_eligibility instead of wor.unbundled \*/

on b.currency/\*hrg\_code\*/=a.hrg

left join work.unbundled as c

on a.hrg=c.hrg\_code

where a.Unbundled\_\_tariff\_\_including\_cos=**1**;

**quit**;

/\* RADIOTHERAPY & CHEMOTHERAPY \*/

\*/

**proc** **sql**;

create table Stage\_RadioTherapy\_Scope as SELECT organisationcode as FK\_ORGS\_PROV\_ID, DEPARTMENTcode as DEPARTMENT, SERVICEcode as SERVICE, CURRENCYcode as CURRENCY, UNITCOST as UC, activity as ACTIVITY\_p1, (UNITCOST \* activity) as TOTAL\_COST

from work.Reference\_Cost rcna

WHERE (rcna.DEPARTMENTcode IN ('CHEMD', 'CHEMP', 'RADP', 'RADT')) AND

rcna.SUPPLIERTYPE = 'OWN' and service notin ('IP','Oth');

**quit**;

/\*Step 4

Clean Data > ( NA \* 20 < org or NA/20 > org) \*/

**proc** **sql**;

CREATE TABLE Clean\_Data

as SELECT

stage\_RadioTherapy\_Scope\_MFF\_Adj.\*,

CASE WHEN UC\_Target\_MFF/NA<**0.05** OR UC\_Target\_MFF/NA>**20**

THEN **1** ELSE **0** END as anomaly

FROM stage\_RadioTherapy\_Scope\_MFF\_Adj INNER JOIN National\_Average ON

stage\_RadioTherapy\_Scope\_MFF\_Adj.CURRENCY = National\_Average.CURRENCY AND

stage\_RadioTherapy\_Scope\_MFF\_Adj.DEPARTMENT = National\_Average.DEPARTMENT AND

stage\_RadioTherapy\_Scope\_MFF\_Adj.Service = National\_Average.Service

WHERE

(CASE WHEN UC\_Target\_MFF/NA<**0.05** OR UC\_Target\_MFF/NA>**20**

THEN **1** ELSE **0** END) = **0** and NA <> **0**;

**quit**;

/\*Step 6

National Average MFF adjustment \*/

**Proc** **sql**;

CREATE TABLE MFF\_IMPACT

as SELECT SUM(TOTAL\_COST) as INC\_MFF, SUM(TC\_Target\_MFF) as EXEC\_MFF, SUM(TOTAL\_COST)/SUM(TC\_Target\_MFF) as NA\_MFF\_ADJ

FROM stage\_RadioTherapy\_Scope\_MFF\_Adj;

**quit**;

/\*Step 7

MFF RESCALE \*/

**proc** **sql**;

CREATE TABLE MFF\_RESCALE

as SELECT SUM(TC\_Target\_MFF) as EXEC\_MFF, SUM(TC\_Payment\_MFF) as EXEC\_MFF\_CAPPED,

SUM(TC\_Payment\_MFF)/SUM(TC\_Target\_MFF) as MFF\_RESCALE

FROM stage\_RadioTherapy\_Scope\_MFF\_Adj;

**quit**;

/\*Step 3

National Average \*/

**proc** **sql**;

CREATE TABLE National\_Average

as SELECT DEPARTMENT, SERVICE, CURRENCY, Sum(UC\_Target\_MFF\*ACTIVITY\_p1)/Sum(ACTIVITY\_p1) AS NA

FROM stage\_RadioTherapy\_Scope\_MFF\_Adj

GROUP BY DEPARTMENT, SERVICE, CURRENCY;

**quit**;

/\*Step 5

Recreate National Average (combining DA/OP SERVICES HERE) \*/

**Proc** **Sql**;

CREATE TABLE NATIONAL\_DATA\_CLEAN

as SELECT

(CASE WHEN DEPARTMENT IN ('CHEMD', 'CHEMP') then 'CHEMTHPY'

WHEN DEPARTMENT IN ('RADP', 'RADT') then 'RADTHPY' END) AS DEPARTMENT,

CURRENCY,

SUM(ACTIVITY\_p1) as Total\_Activity, Sum(UC\_Target\_MFF\*ACTIVITY\_p1) as TOTAL\_COST,

Sum(UC\_Target\_MFF\*ACTIVITY\_p1)/Sum(ACTIVITY\_p1) AS HRG\_UC\_Target\_MFF

FROM Clean\_Data

GROUP BY

(CASE WHEN DEPARTMENT IN ('CHEMD', 'CHEMP') then 'CHEMTHPY'

WHEN DEPARTMENT IN ('RADP', 'RADT') then 'RADTHPY' END),

CURRENCY;

**quit**;

/\*Step 8

Generate Tariff \*/

**Proc** **Sql**;

CREATE TABLE output\_Total\_Act\_Cost\_Chemo\_R

as SELECT distinct nt.DEPARTMENT, nt.CURRENCY, nt.TOTAL\_ACTIVITY format=COMMA15.0, nt.TOTAL\_COST format=COMMA15.2,

ROUND(nt.HRG\_UC\_Target\_MFF,**.01**) as HRG\_UC,

HRG\_UC\_Target\_MFF\*MFF\_RESCALE format=comma8. as TARIFF

FROM MFF\_RESCALE, work.Uplift, NATIONAL\_DATA\_CLEAN nt

ORDER BY DEPARTMENT, CURRENCY;

**quit**;

**data** Chemo\_pre\_qr1;

set output\_Total\_Act\_Cost\_Chemo\_R;

where Currency in(

"SB11Z"

"SB12Z"

"SB13Z"

"SB14Z"

"SB15Z"

"SB17Z"

);

**run**;

**data** Chemo\_pre\_qr1\_2;

set Chemo\_pre\_qr1;

if currency="SB11Z" then weighted\_total\_activity=TOTAL\_ACTIVITY\***0.8**;

else if currency="SB12Z" then weighted\_total\_activity=TOTAL\_ACTIVITY\***1.0**;

else if currency="SB13Z" then weighted\_total\_activity=TOTAL\_ACTIVITY\***2.0**;

else if currency="SB14Z" then weighted\_total\_activity=TOTAL\_ACTIVITY\***3.0**;

else if currency="SB15Z" then weighted\_total\_activity=TOTAL\_ACTIVITY\***2.0**;

else if currency="SB17Z" then weighted\_total\_activity=TOTAL\_ACTIVITY\***0.0**;

**run**;

**data** Chemo\_pre\_qr1\_2;

set Chemo\_pre\_qr1;

if currency="SB11Z" then weight=**0.8**;

else if currency="SB12Z" then weight=**1.0**;

else if currency="SB13Z" then weight=**2.0**;

else if currency="SB14Z" then weight=**3.0**;

else if currency="SB15Z" then weight=**2.0**;

else if currency="SB17Z" then weight=**0.0**;

weighted\_total\_activity=total\_activity\*weight;

**run**;

**proc** **sql**; create table Chemo\_pre\_qr1\_3 as select \*, sum(total\_cost)/sum(weighted\_total\_activity) as avg\_uc from Chemo\_pre\_qr1\_2;

**quit**;

**data** Chemo\_pre\_qr1\_4; set Chemo\_pre\_qr1\_3; Price\_weighted=avg\_uc\*weight;

format Price\_weighted comma7.0;

**run**;

**data** Radio\_pre\_qr1; set output\_Total\_Act\_Cost\_Chemo\_R;

where find(currency, "SC", "i");

**run**;

**data** Radio\_pre\_qr1b; set Radio\_pre\_qr1;

if currency in("SC29Z" "SC56Z") then uc\_used\_in\_calc=**0**; else uc\_used\_in\_calc=tariff;

if currency in("SC29Z" "SC56Z") then revised\_quantum=**0**; else revised\_quantum=total\_cost;

if currency in("SC29Z" "SC56Z") then zero\_price\_applies="Yes"; else zero\_price\_applies="No";

**run**;

**proc** **sql**; create table Radio\_pre\_qr2 as select \*, sum(total\_cost) as sum1 from Radio\_pre\_qr1b

where currency in(

"SC21Z"

"SC22Z"

"SC23Z"

"SC24Z"

"SC25Z"

"SC29Z"

"SC31Z"

);

**quit**;

**proc** **sql**; create table Radio\_pre\_qr3 as select \*, sum(revised\_quantum) as sum2 from Radio\_pre\_qr2

where currency in(

"SC21Z"

"SC22Z"

"SC23Z"

"SC24Z"

"SC25Z"

"SC29Z"

"SC31Z"

);

**quit**;

**proc** **sql**; create table Radio\_pre\_qr4 as select \*, sum(total\_cost) as sum1 from Radio\_pre\_qr1b

where currency in(

"SC40Z"

"SC41Z"

"SC42Z"

"SC43Z"

"SC44Z"

"SC45Z"

"SC46Z"

"SC47Z"

"SC48Z"

"SC49Z"

"SC50Z"

"SC51Z"

"SC52Z"

"SC56Z"

);

**quit**;

**proc** **sql**; create table Radio\_pre\_qr5 as select \*, sum(revised\_quantum) as sum2 from Radio\_pre\_qr4

where currency in(

"SC40Z"

"SC41Z"

"SC42Z"

"SC43Z"

"SC44Z"

"SC45Z"

"SC46Z"

"SC47Z"

"SC48Z"

"SC49Z"

"SC50Z"

"SC51Z"

"SC52Z"

"SC56Z"

);

**quit**;

**Data** Radio\_pre\_qr6;

set Radio\_pre\_qr3 Radio\_pre\_qr5;

Zero\_price\_adj=sum1/sum2;

format Zero\_price\_adj PERCENT7.0;

Neutral\_Quantum=revised\_quantum\*Zero\_price\_adj;

final\_prices\_neutral=Neutral\_Quantum/total\_activity;

format Neutral\_Quantum final\_prices\_neutral Comma15.0;

**run**;

/\*Step 2

Remove MFF\*/

**proc** **sql**;

CREATE TABLE stage\_RadioTherapy\_Scope\_MFF\_Adj

\*/

as SELECT scope.\*, UC/p.uncapped\_MFF as uc\_target\_mff, TOTAL\_COST/p.uncapped\_MFF as tc\_target\_mff, TOTAL\_COST/p.Capped\_MFF as tc\_payment\_mff

FROM Stage\_RadioTherapy\_Scope AS scope INNER JOIN work.tc\_mff p ON scope.FK\_ORGS\_PROV\_ID = p.RC\_Code;

**quit**;

/\* --- Clean Data > ( NA \* 20 < org or NA/20 > org) \*/

**proc** **sql**;

CREATE TABLE Clean\_Data

as SELECT Remove\_MFF.\*, National\_Average.NA, CASE WHEN UC\_Target\_MFF/NA<**0.05** OR UC\_Target\_MFF/NA>**20**

THEN **1** ELSE **0** END as anomaly

FROM stage\_Renal\_CKD\_Scpe\_MFF\_Adj as Remove\_MFF INNER JOIN National\_Average ON

Remove\_MFF.CURRENCY = National\_Average.CURRENCY AND

Remove\_MFF.DEPARTMENT = National\_Average.DEPARTMENT AND

Remove\_MFF.Service = National\_Average.Service

WHERE

(CASE WHEN UC\_Target\_MFF/NA<**0.05** OR UC\_Target\_MFF/NA>**20**

THEN **1** ELSE **0** END) = **0**;

**quit**;

/\* -- National Average MFF adjustment \*/

**proc** **sql**;

CREATE TABLE MFF\_IMPACT

as SELECT SUM(TOTAL\_COST) as INC\_MFF, SUM(TC\_Target\_MFF) as EXEC\_MFF, SUM(TOTAL\_COST)/SUM(TC\_Target\_MFF) as NA\_MFF\_ADJ

FROM

stage\_Renal\_CKD\_Scpe\_MFF\_Adj;

**quit**;

**Proc** **sql**;

CREATE TABLE MFF\_RESCALE

as SELECT SUM(TC\_Target\_MFF) as EXEC\_MFF, SUM(TC\_Payment\_MFF) as EXEC\_MFF\_CAPPED,

SUM(TC\_Payment\_MFF)/SUM(TC\_Target\_MFF) as MFF\_RESCALE

FROM stage\_Renal\_CKD\_Scpe\_MFF\_Adj;

**quit**;

**Proc** **sql**;

CREATE TABLE National\_Average

as SELECT DEPARTMENT, Service, CURRENCY, Sum(UC\_Target\_MFF\*ACTIVITY\_p1)/Sum(ACTIVITY\_p1) AS NA

FROM stage\_Renal\_CKD\_Scpe\_MFF\_Adj

GROUP BY DEPARTMENT, Service, CURRENCY;

**quit**;

**pROC** **SQL**;

CREATE TABLE NATIONAL\_DATA\_CLEAN

AS SELECT

DEPARTMENT,

CURRENCY,

SUM(ACTIVITY\_p1) as Total\_Activity, Sum(UC\_Target\_MFF\*ACTIVITY\_p1) as TOTAL\_COST,

Sum(UC\_Target\_MFF\*ACTIVITY\_p1)/Sum(ACTIVITY\_p1) AS HRG\_UC\_Target\_MFF

FROM Clean\_Data

GROUP BY

DEPARTMENT, CURRENCY;

**quit**;

**proc** **sql**;

CREATE TABLE output\_Ttl\_Act\_Cst\_Renal\_CKD2

as SELECT distinct nt.DEPARTMENT, nt.CURRENCY, nt.TOTAL\_ACTIVITY, nt.TOTAL\_COST,

nt.HRG\_UC\_Target\_MFF as UC,

(HRG\_UC\_Target\_MFF\*MFF\_RESCALE) as TARIFF2

FROM MFF\_RESCALE, work.Uplift, NATIONAL\_DATA\_CLEAN nt

WHERE

nt.CURRENCY like '%A'

ORDER BY DEPARTMENT, CURRENCY;

**quit**;

**DATA** output\_Ttl\_Act\_Cst\_Renal\_CKD (DROP=TARIFF2); SET output\_Ttl\_Act\_Cst\_Renal\_CKD2;

TARIFF=TARIFF2;

FORMAT TOTAL\_ACTIVITY TARIFF COMMA15.0;

FORMAT TOTAL\_COST UC COMMA15.2; **RUN**;

**data** Renal\_CKD\_post\_qr1;

set output\_Ttl\_Act\_Cst\_Renal\_CKD;

rc\_qr1\_factor=&rc\_qr1\_factor\_RenalCKD.;

modelled\_TARIFF\_withQR1=TARIFF\*(**1**+rc\_qr1\_factor);

TOTAL\_COST\_postQR1=TOTAL\_ACTIVITY\*modelled\_TARIFF\_withQR1;

Format modelled\_TARIFF\_withQR1 Comma7.0;

Format TOTAL\_COST\_postQR1 Comma15.2;

**run**;

**data** Renal\_CKD\_post\_qr1\_cb\_ie; set Renal\_CKD\_post\_qr1;

Cost\_base\_adjustment\_factor=&Cost\_base\_adjustment\_factor.;

TOTAL\_price\_postQR1\_cb=modelled\_TARIFF\_withQR1\*(**1**+Cost\_base\_adjustment\_factor);

ie\_indextn\_adjstmt\_factors=&ie\_indextn\_adjstmt\_factors.;

final\_prices\_renal\_CKD=TOTAL\_price\_postQR1\_cb\*(**1**+ie\_indextn\_adjstmt\_factors);

format final\_prices\_renal\_CKD comma15.0;

format rc\_qr1\_factor Cost\_base\_adjustment\_factor ie\_indextn\_adjstmt\_factors percentn7.2;

**run**;

**data** renal\_CKD\_PRICES\_B4IA(keep=renal\_price\_id Year department hrg\_code total\_activity tariff);

set Renal\_CKD\_post\_qr1\_cb\_ie(drop=tariff);

rename currency=hrg\_code;

rename final\_prices\_renal\_CKD=tariff;

renal\_price\_ID=**4**;

Year="19/20";

**run**;

**proc** **sql**; create table renal\_CKD\_4IA\_Model

as select a.renal\_price\_ID, a.Year, a.hrg\_code, b.hrg\_name, a.tariff, b.Session, a.total\_activity from renal\_CKD\_PRICES\_B4IA as a

left join work.renal as b /\* use hrg\_eligibility table instead \*/

on a.hrg\_code=b.hrg\_code;

**quit**;

**data** ModelOut.renal\_CKD\_4IA\_Model\_1920;

set renal\_CKD\_4IA\_Model;

format tariff comma16.15;

**run**;

**data** renal\_4\_BPT\_Model\_1920;

set renal\_CKD\_4IA\_Model;

format tariff comma16.15;

**run**;

**data** ModelOut.renal\_4\_BPT\_Model\_1920;

set renal\_CKD\_4IA\_Model;

format tariff comma16.15; **run**;

/\* --SELECT \* FROM #MFF\_RESCALE \*/

**proc** **sql**;

CREATE TABLE output\_Ttl\_Act\_Cst\_Renal\_CKD2

as SELECT distinct nt.DEPARTMENT, nt.CURRENCY, nt.TOTAL\_ACTIVITY, nt.TOTAL\_COST,

nt.HRG\_UC\_Target\_MFF as UC,

(HRG\_UC\_Target\_MFF\*MFF\_RESCALE) as TARIFF2

FROM MFF\_RESCALE, work.Uplift, NATIONAL\_DATA\_CLEAN nt

WHERE

nt.CURRENCY like '%A'

ORDER BY DEPARTMENT, CURRENCY;

**quit**;

**DATA** output\_Ttl\_Act\_Cst\_Renal\_CKD (DROP=TARIFF2); SET output\_Ttl\_Act\_Cst\_Renal\_CKD2;

TARIFF=TARIFF2;

FORMAT TOTAL\_ACTIVITY TARIFF COMMA15.0;

FORMAT TOTAL\_COST UC COMMA15.2; **RUN**;

**data** Renal\_CKD\_post\_qr1;

set output\_Ttl\_Act\_Cst\_Renal\_CKD;

rc\_qr1\_factor=&rc\_qr1\_factor\_RenalCKD.;

modelled\_TARIFF\_withQR1=TARIFF\*(**1**+rc\_qr1\_factor);

TOTAL\_COST\_postQR1=TOTAL\_ACTIVITY\*modelled\_TARIFF\_withQR1;

Format modelled\_TARIFF\_withQR1 Comma7.0;

Format TOTAL\_COST\_postQR1 Comma15.2;

**run**;

**data** Renal\_CKD\_post\_qr1\_cb\_ie; set Renal\_CKD\_post\_qr1;

Cost\_base\_adjustment\_factor=&Cost\_base\_adjustment\_factor.;

TOTAL\_price\_postQR1\_cb=modelled\_TARIFF\_withQR1\*(**1**+Cost\_base\_adjustment\_factor);

ie\_indextn\_adjstmt\_factors=&ie\_indextn\_adjstmt\_factors.;

final\_prices\_renal\_CKD=TOTAL\_price\_postQR1\_cb\*(**1**+ie\_indextn\_adjstmt\_factors);

format final\_prices\_renal\_CKD comma15.0;

format rc\_qr1\_factor Cost\_base\_adjustment\_factor ie\_indextn\_adjstmt\_factors percentn7.2;

**run**;

**data** renal\_CKD\_PRICES\_B4IA(keep=renal\_price\_id Year department hrg\_code total\_activity tariff);

set Renal\_CKD\_post\_qr1\_cb\_ie(drop=tariff);

rename currency=hrg\_code;

rename final\_prices\_renal\_CKD=tariff;

renal\_price\_ID=**4**;

Year="19/20";

**run**;

**proc** **sql**; create table renal\_CKD\_4IA\_Model

as select a.renal\_price\_ID, a.Year, a.hrg\_code, b.hrg\_name, a.tariff, b.Session, a.total\_activity from renal\_CKD\_PRICES\_B4IA as a

left join work.renal as b /\* use hrg\_eligibility table instead \*/

on a.hrg\_code=b.hrg\_code;

**quit**;

**data** ModelOut.renal\_CKD\_4IA\_Model\_1920;

set renal\_CKD\_4IA\_Model;

tariff=tariff\*(**1**+&Infla\_Effi\_final\_year\_uplift.);

format tariff comma16.15;

**run**;

**data** renal\_4\_BPT\_Model\_1920;

set renal\_CKD\_4IA\_Model;

tariff=tariff\*(**1**+&Infla\_Effi\_final\_year\_uplift.);

format tariff comma16.15; **run**;

/\* PID added below 20180425 \*/

**data** ModelOut.renal\_4\_BPT\_Model\_1920;

set renal\_CKD\_4IA\_Model;

tariff=tariff\*(**1**+&Infla\_Effi\_final\_year\_uplift.);

format tariff comma16.15;

**run**;

**Proc** **sql**;

CREATE TABLE stage\_Renal\_CKD\_Scpe\_MFF\_Adj

as SELECT stage\_Renal\_CKD\_Scope.\*, UC/p.uncapped\_MFF as uc\_target\_mff, TOTAL\_COST/p.uncapped\_MFF as tc\_target\_mff, TOTAL\_COST/p.Capped\_MFF as tc\_payment\_mff

FROM stage\_Renal\_CKD\_Scope INNER JOIN work.tc\_mff p ON

stage\_Renal\_CKD\_Scope.FK\_ORGS\_PROV\_ID = p.RC\_Code;

**quit**;

/\*-- =============================================

Change History -

-- Filter Non\_Acute Table

--Change for 2016/17. Added departments 'RENALAKI' and 'RENALCKD' to filter renal costs. (05/01/2015)

--Change for 2016/17. Removed [SUPPLIER\_TYPE] like 'own'. (05/01/2015)

--Uncommented 'OWN' filter 27/11/2015

18/01/2016 DA Fix for Defect 8029 - Added step to drop table [stage].[Renal\_CKD\_Scope\_MFF\_Adjusted]

-- =============================================\*/

**Proc** **sql**;

CREATE TABLE stage\_Renal\_CKD\_Scope

as SELECT organisationcode as FK\_ORGS\_PROV\_ID, DEPARTMENTcode as DEPARTMENT, SERVICEcode as SERVICE, CURRENCYcode as CURRENCY,

UNITCOST as UC, ACTIVITY as ACTIVITY\_P1, (UNITCOST \* ACTIVITY) as TOTAL\_COST

FROM work.Reference\_Cost rcna

WHERE

(rcna.organisationcode not like 'RHW' AND rcna.organisationcode not like 'RBD')

AND

rcna.DEPARTMENTCODE in ('RENALCKD')

AND

rcna.SUPPLIERTYPE = 'OWN' ;

**quit**;

/\*-- =============================================

-- Filter Non\_Acute Table

--Change for 2016/17. Added departments 'RENALAKI' and 'RENALCKD' to filter renal costs. (05/01/2015)

--Change for 2016/17. Removed [SUPPLIER\_TYPE] like 'own'. (05/01/2015)

--Uncommented 'OWN' filter 27/11/2015

18/01/2016 DA Fix for Defect 8029 - Added step to drop table [stage].[Renal\_CKD\_Scope\_MFF\_Adjusted]

-- =============================================\*/

/\*BEGIN\*/

**Proc** **sql**;

CREATE TABLE stage\_Renal\_CKD\_Scope

as SELECT organisationcode as FK\_ORGS\_PROV\_ID, DEPARTMENTcode as DEPARTMENT, SERVICEcode as SERVICE, CURRENCYcode as CURRENCY,

UNITCOST as UC, ACTIVITY as ACTIVITY\_P1, (UNITCOST \* ACTIVITY) as TOTAL\_COST

FROM work.Reference\_Cost rcna

WHERE

rcna.DEPARTMENTCODE in ('RENALCKD')/\* -- added new renal departments\*/

AND

rcna.SUPPLIERTYPE = 'OWN' /\*-- supplier type deleted as doesn't exist in 13/14 ref costs -- condition included SPK 27/11/2015 \*/;

**quit**;

**proc** **sql**;

create table new1 as select a.Department, a.Currency, a.hrg\_name, a.\*, B.HRG\_CODE,B.PRICE\_DECISION,c.PostMA\_PRICE from DI\_post\_qr1\_CB\_IE2 as a

left join MA\_UNBUNDLED\_Input as b

on a.HRG\_CODE=b.HRG\_CODE

LEFT JOIN MA\_UNBUNDLED\_Output

as c

on

a.HRG\_CODE=c.HRG\_CODE;

**quit**;

**DATA** NEW1A; SET NEW1; WHERE FIND(HRG\_CODE, "RD", "i"); **run**; **data** NEW1B; SET NEW1; WHERE FIND(HRG\_CODE, "RN", "i"); **run**;

**data** new2a(drop=PostMA\_PRICE);

set new1a;

IF currency="" then currency=hrg\_code; else currency=currency;

IF DEPARTMENT="" then DEPARTMENT="IMAG"; else DEPARTMENT=DEPARTMENT;

if price\_decision="Adjusted" then Price=Final\_Prices\_DI;

else Price=Final\_Prices\_DI;

if price\_decision="Adjusted" then price\_decision="Modelled";

else Price\_decision=Price\_decision;

format Price Comma7.0;

quantum\_pre\_MA=TOTAL\_ACTIVITY\*final\_prices\_DI;

quantum\_post\_MA=total\_activity\*Price;

**run**;

**proc** **sql**; create table new3a as select sum(quantum\_pre\_MA)/sum(quantum\_post\_MA) as qr2\_ma from new2a; **quit**;

**proc** **sql**;

select qr2\_ma into :qr2\_factor

from new3a;

**quit**;

%let rc\_qr2\_factor=&qr2\_factor.;

**data** rd\_UNBUNDLED\_PRICES\_4\_IA\_model; set new2a;

rc\_qr2\_factor=&rc\_qr2\_factor.;

TOTAL\_price\_postQR2\_cb=PRICE\*rc\_qr2\_factor;

**run**;

**data** ModelOut.rd\_UNBUNDLED\_PRICES4\_IA\_1920;

set rd\_UNBUNDLED\_PRICES\_4\_IA\_model;

**run**;

**proc** **sql**;

create table rd\_prices as select **4** format=**8.** as Unb\_price\_id,

"19/20" format=$10. as Year, department, currency as hrg\_code, hrg\_name,

TOTAL\_price\_postQR2\_cb as tariff format=**8.**, cost\_of\_reporting

from rd\_UNBUNDLED\_PRICES\_4\_IA\_model; **quit**;

/\* ---------------------- 1 ----------------------------------------------\*/

**data** Unb\_DI\_RD\_STEP\_BY\_STEP (drop=hrg\_code CofR); set new2a(rename=cost\_of\_reporting=CofR);

rc\_qr2\_factor=&rc\_qr2\_factor.;

TOTAL\_price\_postQR2\_cb=PRICE\*rc\_qr2\_factor;

SMOOTHING\_factor=&SMOOTHING\_factor\_DI.;

prices\_SF=TOTAL\_price\_postQR2\_cb\*(**1**+SMOOTHING\_factor);

Scaling\_factor=&Scaling\_factor.;

prices\_after\_SCALING=prices\_SF\*(**1**+Scaling\_factor);

Infla\_Effi\_final=&Infla\_Effi\_final\_year\_uplift.;

final\_prices=prices\_after\_SCALING\*(**1**+Infla\_Effi\_final);

Cost\_of\_reporting=CofR;

format TOTAL\_price\_postQR2\_cb prices\_SF prices\_after\_SCALING final\_prices comma15.0;

format rc\_qr1\_factor SMOOTHING\_factor /\*Scaling\_factor\*/ Infla\_Effi\_final percentn7.2;

format Scaling\_factor percentn8.2;

**run**;

**data** ModelOut.Unb\_DI\_RD\_STEP\_BY\_STEP\_1920;

set Unb\_DI\_RD\_STEP\_BY\_STEP;

**run**;

**data** Unb\_DI\_RD\_PRICES (keep=Department Currency Hrg\_name final\_prices cost\_of\_reporting); set Unb\_DI\_RD\_STEP\_BY\_STEP; **run**;

**proc** **export** data=Unb\_DI\_RD\_STEP\_BY\_STEP

outfile="&outputpath.\UB\_DI\_RD\_Prices\_1920.csv"

dbms=csv

replace;

**run**;

**data** ModelOut.Unb\_DI\_RD\_PRICES\_1920;

set Unb\_DI\_RD\_PRICES;

**run**;

**data** new2b(drop=PostMA\_PRICE);

set new1b;

IF currency="" then currency=hrg\_code;else currency=currency;

IF DEPARTMENT="" then DEPARTMENT="IMAG";else DEPARTMENT=DEPARTMENT;

if price\_decision="Adjusted" then Price=Final\_Prices\_DI;

else Price=Final\_Prices\_DI;

if price\_decision="Adjusted" then price\_decision="Modelled";

else Price\_decision=Price\_decision;

format Price Comma7.0;

quantum\_pre\_MA=TOTAL\_ACTIVITY\*final\_prices\_DI;

quantum\_post\_MA=total\_activity\*Price;

**run**;

**proc** **sql**; create table new3b as select sum(quantum\_pre\_MA)/sum(quantum\_post\_MA) as qr2\_ma from new2b; **quit**;

**proc** **sql**;

select qr2\_ma into :qr2\_factor

from new3b;

**quit**;

%let rc\_qr2\_factor=&qr2\_factor.;

/\* ---------------------- 2 ----------------------------------------------\*/

**data** rn\_UNBUNDLED\_PRICES\_4\_IA\_model; set new2b;

rc\_qr2\_factor=&rc\_qr2\_factor.;

TOTAL\_price\_postQR2\_cb=PRICE\*rc\_qr2\_factor;**run**;

**data** ModelOut.rn\_UNBUNDLED\_PRICES\_4\_IA\_model;

set rn\_UNBUNDLED\_PRICES\_4\_IA\_model;

**run**;

**proc** **sql**;

create table rn\_prices as select **4** format=**8.** as Unb\_price\_id,

"19/20" format=$10. as Year, department, currency as hrg\_code, hrg\_name,

TOTAL\_price\_postQR2\_cb as tariff format=**8.**, cost\_of\_reporting

from rn\_UNBUNDLED\_PRICES\_4\_IA\_model; **quit**;

**data** Unb\_DI\_RN\_STEP\_BY\_STEP(drop=hrg\_code CofR); set new2b(rename=cost\_of\_reporting=CofR);

rc\_qr2\_factor=&rc\_qr2\_factor.;

TOTAL\_price\_postQR2\_cb=PRICE\*rc\_qr2\_factor;

SMOOTHING\_factor=&SMOOTHING\_factor\_DI.;

prices\_SF=TOTAL\_price\_postQR2\_cb\*(**1**+SMOOTHING\_factor);

Scaling\_factor=&Scaling\_factor.;

prices\_after\_SCALING=prices\_SF\*(**1**+Scaling\_factor);

Infla\_Effi\_1718=&Infla\_Effi\_final\_year\_uplift.;

final\_prices=prices\_after\_SCALING\*(**1**+Infla\_Effi\_1718);

Cost\_of\_reporting=CofR;

format TOTAL\_price\_postQR2\_cb prices\_SF prices\_after\_SCALING final\_prices comma15.0;

format rc\_qr1\_factor SMOOTHING\_factor /\*Scaling\_factor\*/ Infla\_Effi\_1718 percentn7.2;

format Scaling\_factor percentn8.2;

**run**;

**data** ModelOut.Unb\_DI\_RN\_STEP\_BY\_STEP\_1920;

set Unb\_DI\_RN\_STEP\_BY\_STEP;

**run**;

**data** Unb\_DI\_RN\_PRICES (keep=Department Currency Hrg\_name final\_prices cost\_of\_reporting); set Unb\_DI\_RN\_STEP\_BY\_STEP; **run**;

**data** ModelOut.Unb\_DI\_RN\_PRICES\_1920;

set Unb\_DI\_RN\_PRICES;

**run**;

**proc** **export** data=Unb\_DI\_RN\_STEP\_BY\_STEP

outfile="&outputpath.\UB\_DI\_RN\_Prices\_1920.csv"

dbms=csv

replace;

**run**;

**proc** **sql**;

create table new1c as select a.department, a.currency, d.description as hrg\_name, a.\*, B.HRG\_CODE,B.PRICE\_DECISION,c.PostMA\_PRICE from Radio\_post\_qr\_ie\_cb as a

left join MA\_UNBUNDLED\_Input as b

on a.currencyb.HRG\_CODE

LEFT JOIN MA\_UNBUNDLED\_Output

as c

on

a.currency=c.HRG\_CODE

left join work.hrg\_eligibility as d

on a.currency=d.hrg

where d.Unbundled\_\_tariff\_\_including\_cos=**1**;

**quit**;

**data** new2c(drop=PostMA\_PRICE);

set new1c;

IF currency="" then currency=hrg\_code;else currency=currency;

if price\_decision="Adjusted" then Price=Final\_Prices\_Radio;

else Price=Final\_Prices\_Radio;

if price\_decision="Adjusted" then price\_decision="Modelled";

else Price\_decision=Price\_decision;

format Price Comma7.0;

quantum\_pre\_MA=TOTAL\_ACTIVITY\*final\_prices\_radio;

quantum\_post\_MA=total\_activity\*Price;

**run**;

**proc** **sql**; create table new3c as select sum(quantum\_pre\_MA)/sum(quantum\_post\_MA) as qr2\_ma from new2c; **quit**;

**proc** **sql**;

select qr2\_ma into :qr2\_factor

from new3c;

**quit**;

%let rc\_qr2\_factor=&qr2\_factor.;

/\* ---------------------- 3 ----------------------------------------------\*/

**data** rad\_UNBUNDLED\_PRICES\_4\_IA\_model; set new2c;

rc\_qr2\_factor=&rc\_qr2\_factor.;

TOTAL\_price\_postQR2\_cb=PRICE\*rc\_qr2\_factor;cost\_of\_reporting=**0**;**run**;

**data** ModelOut.rad\_UNBUNDLED\_PRICES\_4\_IA\_model;

set rad\_UNBUNDLED\_PRICES\_4\_IA\_model;

**run**;

**proc** **sql**;

create table rad\_prices as select **4** format=**8.** as Unb\_price\_id,

"19/20" format=$10. as Year, department, currency as hrg\_code, hrg\_name,

TOTAL\_price\_postQR2\_cb as tariff format=**8.**, cost\_of\_reporting

from rad\_UNBUNDLED\_PRICES\_4\_IA\_model; **quit**;

**data** Unb\_RADIO\_STEP\_BY\_STEP; set new2c;

rc\_qr2\_factor=&rc\_qr2\_factor.;

TOTAL\_price\_postQR2\_cb=PRICE\*rc\_qr2\_factor;

SMOOTHING\_factor=&SMOOTHING\_factor\_Radio.;

prices\_SF=TOTAL\_price\_postQR2\_cb\*(**1**+SMOOTHING\_factor);

Scaling\_factor=&Scaling\_factor.;

prices\_after\_SCALING=prices\_SF\*(**1**+Scaling\_factor);

Infla\_Effi\_final=&Infla\_Effi\_final\_year\_uplift.;

final\_prices=prices\_after\_SCALING\*(**1**+Infla\_Effi\_final);

format TOTAL\_price\_postQR2\_cb prices\_SF prices\_after\_SCALING final\_prices comma15.0;

format rc\_qr1\_factor SMOOTHING\_factor /\*Scaling\_factor\*/ Infla\_Effi\_final percentn7.2;

format Scaling\_factor percentn8.2;

**run**;

**data** ModelOut.Unb\_RADIO\_STEP\_BY\_STEP\_1920;

set Unb\_RADIO\_STEP\_BY\_STEP;

**run**;

**data** Unb\_RADIO\_PRICES (keep=Department Currency Hrg\_name final\_prices); set Unb\_RADIO\_STEP\_BY\_STEP; **run**;

**data** ModelOut.Unb\_RADIO\_PRICES\_1920;

set Unb\_RADIO\_PRICES;

**run**;

**proc** **export** data=Unb\_RADIO\_STEP\_BY\_STEP

outfile="&outputpath.\UB\_Radio\_Prices\_1920.csv"

dbms=csv

replace;

**run**;

**proc** **sql**;

create table new1d as select a.department, a.currency, d.description as hrg\_name, a.\*, B.HRG\_CODE, B.PRICE\_DECISION,c.PostMA\_PRICE from Chemo\_post\_qr\_ie\_cb as a

left join MA\_UNBUNDLED\_Input as b

on a.currency/\*a.HRG\_CODE\*/=b.HRG\_CODE

LEFT JOIN MA\_UNBUNDLED\_Output

as c

on

a.currency/\*a.HRG\_CODE\*/=c.HRG\_CODE

left join work.hrg\_eligibility as d

on a.currency=d.hrg

where d.Unbundled\_\_tariff\_\_including\_cos=**1**;

**quit**;

**data** new2d(drop=PostMA\_PRICE);

set new1d;

IF currency="" then currency=hrg\_code;else currency=currency;

if price\_decision="Adjusted" then Price=Final\_Prices\_Chemo;

else Price=Final\_Prices\_Chemo;

if price\_decision="Adjusted" then price\_decision="Modelled";

else Price\_decision=Price\_decision;

format Price Comma7.0;

quantum\_pre\_MA=TOTAL\_ACTIVITY\*final\_prices\_Chemo;

quantum\_post\_MA=total\_activity\*Price;

**run**;

**proc** **sql**; create table new3d as select sum(quantum\_pre\_MA)/sum(quantum\_post\_MA) as qr2\_ma from new2d; **quit**;

**proc** **sql**;

select qr2\_ma into :qr2\_factor

from new3d;

**quit**;

%let rc\_qr2\_factor=&qr2\_factor.;

/\* ---------------------- 4 ----------------------------------------------\*/

**data** che\_UNBUNDLED\_PRICES\_4\_IA\_model; set new2d;

rc\_qr2\_factor=&rc\_qr2\_factor.;

TOTAL\_price\_postQR2\_cb=PRICE\*rc\_qr2\_factor;cost\_of\_reporting=**0**;**run**;

**data** ModelOut.che\_UNBUNDLED\_PRICES4\_IA\_1920;

set che\_UNBUNDLED\_PRICES\_4\_IA\_model;

**run**;

**proc** **sql**;

create table che\_prices as select **4** format=**8.** as Unb\_price\_id,

"19/20" format=$10. as Year, department, currency as hrg\_code, hrg\_name,

TOTAL\_price\_postQR2\_cb as tariff format=**8.**, cost\_of\_reporting

from che\_UNBUNDLED\_PRICES\_4\_IA\_model; **quit**;

**data** Unb\_CHEMO\_STEP\_BY\_STEP; set new2d;

rc\_qr2\_factor=&rc\_qr2\_factor.;

TOTAL\_price\_postQR2\_cb=PRICE\*rc\_qr2\_factor;

SMOOTHING\_factor=&SMOOTHING\_factor\_Chemo.;

prices\_SF=TOTAL\_price\_postQR2\_cb\*(**1**+SMOOTHING\_factor);

Scaling\_factor=&Scaling\_factor.;

prices\_after\_SCALING=prices\_SF\*(**1**+Scaling\_factor);

Infla\_Effi\_final=&Infla\_Effi\_final\_year\_uplift.;

final\_prices=prices\_after\_SCALING\*(**1**+Infla\_Effi\_final);

format TOTAL\_price\_postQR2\_cb prices\_SF prices\_after\_SCALING final\_prices comma15.0;

format rc\_qr1\_factor SMOOTHING\_factor /\*Scaling\_factor\*/ Infla\_Effi\_final percentn7.2;

format Scaling\_factor percentn8.2;

**run**;

**data** ModelOut.Unb\_CHEMO\_STEP\_BY\_STEP\_1920;

set Unb\_CHEMO\_STEP\_BY\_STEP;

**run**;

**data** Unb\_chemo\_PRICES (keep=Department Currency Hrg\_name final\_prices); set Unb\_chemo\_STEP\_BY\_STEP; **run**;

**data** ModelOut.Unb\_chemo\_PRICES\_1920;

set Unb\_chemo\_PRICES;

**run**;

**proc** **export** data=Unb\_chemo\_STEP\_BY\_STEP

outfile="&outputpath.\UB\_Chemo\_Prices\_1920.csv"

dbms=csv

replace;

**run**;

/\* ---------------------- 5 ----------------------------------------------\*/

**data** UNBUNDLED\_PRICES\_4\_IA\_model;

set rd\_prices rn\_prices rad\_prices che\_prices;

format tariff comma16.15; /\* PID added 20171019 this line to keep decimal places to 15 places \*/

**run**;

**Data** ModelOut.UNBUNDLED\_PRICES\_4\_IA\_model;

set UNBUNDLED\_PRICES\_4\_IA\_model;

**run**;

**proc** **export** data=UNBUNDLED\_PRICES\_4\_IA\_model

outfile="&outputpath.\Unbundled\_Prices\_1920\_4IA\_MODELS.csv"

dbms=csv

replace;

**run**;

**proc** **sql**;

create table new1e as select a.\*,/\*hrg\_name\*/ B.HRG\_CODE,B.PRICE\_DECISION,c.PostMA\_PRICE from Renal\_CKD\_post\_qr1\_cb\_ie as a

left join MA\_UNBUNDLED\_Input as b

on a.currency/\*a.HRG\_CODE\*/=b.HRG\_CODE

LEFT JOIN MA\_UNBUNDLED\_Output

as c

on

a.currency/\*a.HRG\_CODE\*/=c.HRG\_CODE

**quit**;

**data** new2e(drop=PostMA\_PRICE);

set new1e;

IF currency="" then currency=hrg\_code;else currency=currency;

if price\_decision="Adjusted" then Price=Final\_Prices\_Renal\_CKD;

else Price=Final\_Prices\_Renal\_CKD;

if price\_decision="Adjusted" then price\_decision="Modelled";

else Price\_decision=Price\_decision;

format Price Comma7.0;

quantum\_pre\_MA=TOTAL\_ACTIVITY\*final\_prices\_Renal\_CKD;

quantum\_post\_MA=total\_activity\*Price;

**run**;

**proc** **sql**; create table new3e as select sum(quantum\_pre\_MA)/sum(quantum\_post\_MA) as qr2\_ma from new2e; **quit**;

**proc** **sql**;

select qr2\_ma into :qr2\_factor

from new3e;

**quit**;

%let rc\_qr2\_factor=&qr2\_factor.;

/\* ---------------------- 5 ----------------------------------------------\*/

**data** Unb\_RENAL\_STEP\_BY\_STEP; set new2e;

rc\_qr2\_factor=&rc\_qr2\_factor.;

TOTAL\_price\_postQR2\_cb=PRICE\*rc\_qr2\_factor;

SMOOTHING\_factor=&SMOOTHING\_factor\_Renal.;

prices\_SF=TOTAL\_price\_postQR2\_cb\*(**1**+SMOOTHING\_factor);

Scaling\_factor=&Scaling\_factor.;

prices\_after\_SCALING=prices\_SF\*(**1**+Scaling\_factor);

Infla\_Effi\_final=&Infla\_Effi\_final\_year\_uplift.;

final\_prices=prices\_after\_SCALING\*(**1**+Infla\_Effi\_final);

format TOTAL\_price\_postQR2\_cb prices\_SF prices\_after\_SCALING final\_prices comma15.0;

format rc\_qr1\_factor SMOOTHING\_factor /\*Scaling\_factor\*/ Infla\_Effi\_final percentn7.2;

format Scaling\_factor percentn8.2;

**run**;

**data** ModelOut.Unb\_RENAL\_STEP\_BY\_STEP\_1920;

set Unb\_RENAL\_STEP\_BY\_STEP;

**run**;

**data** Unb\_Renal\_PRICES(keep=Department Currency Hrg\_name final\_prices); set Unb\_Renal\_STEP\_BY\_STEP; **run**;

**proc** **export** data=Unb\_RENAL\_STEP\_BY\_STEP

outfile="&outputpath.\UB\_RenalCKD\_Prices\_1920.csv"

dbms=csv

replace;

**run**;

**data** ModelOut.Unb\_Renal\_PRICES\_1920;

set Unb\_Renal\_PRICES;

**run**;

Libname UBMA "\\irnarch\sas\_data\Tariff Rebuild\Model outputs pre\_MA\Unbundled\FY1920\_s118\data";

**proc** **sql**;

create table new1 as select a.Department, a.Currency, a.hrg\_name, a.\*, B.HRG\_CODE,B.PRICE\_DECISION,c.PostMA\_PRICE from DI\_post\_qr1\_CB\_IE2 as a

left join UBMA.MA\_UNBUNDLED\_Input as b

on a.HRG\_CODE=b.HRG\_CODE

LEFT JOIN UBMA.MA\_UNBUNDLED\_Output

as c

on

a.HRG\_CODE=c.HRG\_CODE;

**quit**;

**DATA** NEW1A;

SET NEW1;

WHERE FIND(HRG\_CODE, "RD", "i");

**run**;

**data** NEW1B;

SET NEW1;

WHERE FIND(HRG\_CODE, "RN", "i");

**run**;

**data** new2a(drop=PostMA\_PRICE);

set new1a;

IF currency="" then currency=hrg\_code; else currency=currency;

IF DEPARTMENT="" then DEPARTMENT="IMAG"; else DEPARTMENT=DEPARTMENT;

if price\_decision="Adjusted" then Price=PostMA\_PRICE;

else Price=Final\_Prices\_DI;

format Price Comma7.0;

quantum\_pre\_MA=TOTAL\_ACTIVITY\*final\_prices\_DI;

quantum\_post\_MA=total\_activity\*Price;

**run**;

**proc** **sql**; create table new3a as select sum(quantum\_pre\_MA)/sum(quantum\_post\_MA) as qr2\_ma from new2a; **quit**;

**proc** **sql**;

select qr2\_ma into :qr2\_factor

from new3a;

**quit**;

%let rc\_qr2\_factor=&qr2\_factor.;

**data** rd\_UNBUNDLED\_PRICES\_4\_IA\_model; set new2a;

rc\_qr2\_factor=&rc\_qr2\_factor.;

TOTAL\_price\_postQR2\_cb=PRICE\*rc\_qr2\_factor;

**run**;

**data** ModelOut.rd\_UNBUNDLED\_PRICES4\_IA\_1920;

set rd\_UNBUNDLED\_PRICES\_4\_IA\_model;

**run**;

**proc** **sql**;

create table rd\_prices as select **4** format=**8.** as Unb\_price\_id,

"19/20" format=$10. as Year, department, currency as hrg\_code, hrg\_name,

TOTAL\_price\_postQR2\_cb as tariff format=**8.**, cost\_of\_reporting

from rd\_UNBUNDLED\_PRICES\_4\_IA\_model; **quit**;

**proc** **sql**;

select final\_cashio\_factor into :cashio\_factor\_RD

from cash\_in\_out\_ia\_factors

where POD\_Subchapter="Unbundled\_RD";

**QUIT**;

/\* ---------------------- 1 ----------------------------------------------\*/

**data** Unb\_DI\_RD\_STEP\_BY\_STEP (drop=hrg\_code CofR); set new2a(rename=cost\_of\_reporting=CofR);

rc\_qr2\_factor=&rc\_qr2\_factor.;

TOTAL\_price\_postQR2\_cb=PRICE\*rc\_qr2\_factor;

SMOOTHING\_factor=&SMOOTHING\_factor\_DI.;

prices\_SF=TOTAL\_price\_postQR2\_cb\*(**1**+SMOOTHING\_factor);

Scaling\_factor=&Scaling\_factor.;

prices\_after\_SCALING=prices\_SF\*(**1**+Scaling\_factor);

/\*&cashio\_factor\_RD.\*/

prices\_after\_SCALING=prices\_after\_SCALING\*(**1**+&cashio\_factor\_RD.);

Infla\_Effi\_final=&Infla\_Effi\_final\_year\_uplift.;

final\_prices=prices\_after\_SCALING\*(**1**+Infla\_Effi\_final);

Cost\_of\_reporting=CofR;

format TOTAL\_price\_postQR2\_cb prices\_SF prices\_after\_SCALING final\_prices comma15.0;

format rc\_qr1\_factor SMOOTHING\_factor /\*Scaling\_factor\*/ Infla\_Effi\_final percentn7.2;

format Scaling\_factor percentn8.2;

**run**;

**data** ModelOut.Unb\_DI\_RD\_STEP\_BY\_STEP\_1920;

set Unb\_DI\_RD\_STEP\_BY\_STEP;

**run**;

**data** Unb\_DI\_RD\_PRICES (keep=Department Currency Hrg\_name final\_prices cost\_of\_reporting); set Unb\_DI\_RD\_STEP\_BY\_STEP; **run**;

**proc** **export** data=Unb\_DI\_RD\_STEP\_BY\_STEP

outfile="&outputpath.\UB\_DI\_RD\_Prices\_1920.csv"

dbms=csv

replace;

**run**;

**data** ModelOut.Unb\_DI\_RD\_PRICES\_1920;

set Unb\_DI\_RD\_PRICES;

**run**;

**data** new2b(drop=PostMA\_PRICE);

set new1b;

IF currency="" then currency=hrg\_code;else currency=currency;

IF DEPARTMENT="" then DEPARTMENT="IMAG";else DEPARTMENT=DEPARTMENT;

if price\_decision="Adjusted" then Price=PostMA\_PRICE;

else Price=Final\_Prices\_DI;

format Price Comma7.0;

quantum\_pre\_MA=TOTAL\_ACTIVITY\*final\_prices\_DI;

quantum\_post\_MA=total\_activity\*Price;

**run**;

**proc** **sql**; create table new3b as select sum(quantum\_pre\_MA)/sum(quantum\_post\_MA) as qr2\_ma from new2b; **quit**;

**proc** **sql**;

select qr2\_ma into :qr2\_factor

from new3b;

**quit**;

%let rc\_qr2\_factor=&qr2\_factor.;

/\* ---------------------- 2 ----------------------------------------------\*/

**data** rn\_UNBUNDLED\_PRICES\_4\_IA\_model; set new2b;

rc\_qr2\_factor=&rc\_qr2\_factor.;

TOTAL\_price\_postQR2\_cb=PRICE\*rc\_qr2\_factor;**run**;

**data** ModelOut.rn\_UNBUNDLED\_PRICES\_4\_IA\_model;

set rn\_UNBUNDLED\_PRICES\_4\_IA\_model;

**run**;

**proc** **sql**;

create table rn\_prices as select **4** format=**8.** as Unb\_price\_id,

"19/20" format=$10. as Year, department, currency as hrg\_code, hrg\_name,

TOTAL\_price\_postQR2\_cb as tariff format=**8.**, cost\_of\_reporting

from rn\_UNBUNDLED\_PRICES\_4\_IA\_model; **quit**;

**proc** **sql**;

select final\_cashio\_factor into :cashio\_factor\_RN

from cash\_in\_out\_ia\_factors

where POD\_Subchapter="Unbundled\_RN";

**QUIT**;

**data** Unb\_DI\_RN\_STEP\_BY\_STEP(drop=hrg\_code CofR); set new2b(rename=cost\_of\_reporting=CofR);

rc\_qr2\_factor=&rc\_qr2\_factor.;

TOTAL\_price\_postQR2\_cb=PRICE\*rc\_qr2\_factor;

SMOOTHING\_factor=&SMOOTHING\_factor\_DI.;

prices\_SF=TOTAL\_price\_postQR2\_cb\*(**1**+SMOOTHING\_factor);

Scaling\_factor=&Scaling\_factor.;

prices\_after\_SCALING=prices\_SF\*(**1**+Scaling\_factor);

/\*&cashio\_factor\_RN.\*/

prices\_after\_SCALING=prices\_after\_SCALING\*(**1**+&cashio\_factor\_RN.);

Infla\_Effi\_1718=&Infla\_Effi\_final\_year\_uplift.;

final\_prices=prices\_after\_SCALING\*(**1**+Infla\_Effi\_1718);

Cost\_of\_reporting=CofR;

format TOTAL\_price\_postQR2\_cb prices\_SF prices\_after\_SCALING final\_prices comma15.0;

format rc\_qr1\_factor SMOOTHING\_factor /\*Scaling\_factor\*/ Infla\_Effi\_1718 percentn7.2;

format Scaling\_factor percentn8.2;

**run**;

**data** ModelOut.Unb\_DI\_RN\_STEP\_BY\_STEP\_1920;

set Unb\_DI\_RN\_STEP\_BY\_STEP;

**run**;

**data** Unb\_DI\_RN\_PRICES (keep=Department Currency Hrg\_name final\_prices cost\_of\_reporting); set Unb\_DI\_RN\_STEP\_BY\_STEP; **run**;

**data** ModelOut.Unb\_DI\_RN\_PRICES\_1920;

set Unb\_DI\_RN\_PRICES;

**run**;

**proc** **export** data=Unb\_DI\_RN\_STEP\_BY\_STEP

outfile="&outputpath.\UB\_DI\_RN\_Prices\_1920.csv"

dbms=csv

replace;

**run**;

**proc** **sql**;

create table new1c as select a.department, a.currency, d.description as hrg\_name, a.\*, B.HRG\_CODE,B.PRICE\_DECISION,c.PostMA\_PRICE from Radio\_post\_qr\_ie\_cb as a

left join UBMA.MA\_UNBUNDLED\_Input as b

on a.currency/\*a.HRG\_CODE\*/=b.HRG\_CODE

LEFT JOIN UBMA.MA\_UNBUNDLED\_Output

as c

on

a.currency/\*a.HRG\_CODE\*/=c.HRG\_CODE

left join work.hrg\_eligibility as d

on a.currency=d.hrg

where d.Unbundled\_\_tariff\_\_including\_cos=**1**;

**quit**;

**data** new2c(drop=PostMA\_PRICE);

set new1c;

IF currency="" then currency=hrg\_code;else currency=currency;

if price\_decision="Adjusted" then Price=PostMA\_PRICE;

else Price=Final\_Prices\_Radio;

format Price Comma7.0;

quantum\_pre\_MA=TOTAL\_ACTIVITY\*final\_prices\_radio;

quantum\_post\_MA=total\_activity\*Price;

**run**;

**proc** **sql**; create table new3c as select sum(quantum\_pre\_MA)/sum(quantum\_post\_MA) as qr2\_ma from new2c; **quit**;

**proc** **sql**;

select qr2\_ma into :qr2\_factor

from new3c;

**quit**;

%let rc\_qr2\_factor=&qr2\_factor.;

/\* ---------------------- 3 ----------------------------------------------\*/

**data** rad\_UNBUNDLED\_PRICES\_4\_IA\_model; set new2c;

rc\_qr2\_factor=&rc\_qr2\_factor.;

TOTAL\_price\_postQR2\_cb=PRICE\*rc\_qr2\_factor;cost\_of\_reporting=**0**;**run**;

**data** ModelOut.rad\_UNBUNDLED\_PRICES\_4\_IA\_model;

set rad\_UNBUNDLED\_PRICES\_4\_IA\_model;

**run**;

**proc** **sql**;

create table rad\_prices as select **4** format=**8.** as Unb\_price\_id,

"19/20" format=$10. as Year, department, currency as hrg\_code, hrg\_name,

TOTAL\_price\_postQR2\_cb as tariff format=**8.**, cost\_of\_reporting

from rad\_UNBUNDLED\_PRICES\_4\_IA\_model; **quit**;

**proc** **sql**;

select final\_cashio\_factor into :cashio\_factor\_SC

from cash\_in\_out\_ia\_factors

where POD\_Subchapter="Unbundled\_SC";

**QUIT**;

**data** Unb\_RADIO\_STEP\_BY\_STEP; set new2c;

rc\_qr2\_factor=&rc\_qr2\_factor.;

TOTAL\_price\_postQR2\_cb=PRICE\*rc\_qr2\_factor;

SMOOTHING\_factor=&SMOOTHING\_factor\_Radio.;

prices\_SF=TOTAL\_price\_postQR2\_cb\*(**1**+SMOOTHING\_factor);

Scaling\_factor=&Scaling\_factor.;

prices\_after\_SCALING=prices\_SF\*(**1**+Scaling\_factor);

prices\_after\_SCALING=prices\_after\_SCALING\*(**1**+&cashio\_factor\_SC.);

Infla\_Effi\_final=&Infla\_Effi\_final\_year\_uplift.;

final\_prices=prices\_after\_SCALING\*(**1**+Infla\_Effi\_final);

format TOTAL\_price\_postQR2\_cb prices\_SF prices\_after\_SCALING final\_prices comma15.0;

format rc\_qr1\_factor SMOOTHING\_factor /\*Scaling\_factor\*/ Infla\_Effi\_final percentn7.2;

format Scaling\_factor percentn8.2;

**run**;

**data** ModelOut.Unb\_RADIO\_STEP\_BY\_STEP\_1920;

set Unb\_RADIO\_STEP\_BY\_STEP;

**run**;

**data** Unb\_RADIO\_PRICES (keep=Department Currency Hrg\_name final\_prices); set Unb\_RADIO\_STEP\_BY\_STEP; **run**;

**data** ModelOut.Unb\_RADIO\_PRICES\_1920;

set Unb\_RADIO\_PRICES;

**run**;

**proc** **export** data=Unb\_RADIO\_STEP\_BY\_STEP

outfile="&outputpath.\UB\_Radio\_Prices\_1920.csv"

dbms=csv

replace;

**run**;

**proc** **sql**;

create table new1d as select a.department, a.currency, d.description as hrg\_name, a.\*, B.HRG\_CODE, B.PRICE\_DECISION,c.PostMA\_PRICE from Chemo\_post\_qr\_ie\_cb as a

left join UBMA.MA\_UNBUNDLED\_Input as b

on a.currency/\*a.HRG\_CODE\*/=b.HRG\_CODE

LEFT JOIN UBMA.MA\_UNBUNDLED\_Output

as c

on

a.currency/\*a.HRG\_CODE\*/=c.HRG\_CODE

left join work.hrg\_eligibility as d

on a.currency=d.hrg

where d.Unbundled\_\_tariff\_\_including\_cos=**1**;

**quit**;

**data** new2d(drop=PostMA\_PRICE);

set new1d;

IF currency="" then currency=hrg\_code;else currency=currency;

if price\_decision="Adjusted" then Price=PostMA\_PRICE;

else Price=Final\_Prices\_Chemo;

format Price Comma7.0;

quantum\_pre\_MA=TOTAL\_ACTIVITY\*final\_prices\_Chemo;

quantum\_post\_MA=total\_activity\*Price;

**run**;

**proc** **sql**; create table new3d as select sum(quantum\_pre\_MA)/sum(quantum\_post\_MA) as qr2\_ma from new2d; **quit**;

**proc** **sql**;

select qr2\_ma into :qr2\_factor

from new3d;

**quit**;

%let rc\_qr2\_factor=&qr2\_factor.;

/\* ---------------------- 4 ----------------------------------------------\*/

**data** che\_UNBUNDLED\_PRICES\_4\_IA\_model; set new2d;

rc\_qr2\_factor=&rc\_qr2\_factor.;

TOTAL\_price\_postQR2\_cb=PRICE\*rc\_qr2\_factor;cost\_of\_reporting=**0**;**run**;

**data** ModelOut.che\_UNBUNDLED\_PRICES4\_IA\_1920;

set che\_UNBUNDLED\_PRICES\_4\_IA\_model;

**run**;

**proc** **sql**;

create table che\_prices as select **4** format=**8.** as Unb\_price\_id,

"19/20" format=$10. as Year, department, currency as hrg\_code, hrg\_name,

TOTAL\_price\_postQR2\_cb as tariff format=**8.**, cost\_of\_reporting

from che\_UNBUNDLED\_PRICES\_4\_IA\_model; **quit**;

**proc** **sql**;

select final\_cashio\_factor into :cashio\_factor\_SB

from cash\_in\_out\_ia\_factors

where POD\_Subchapter="Unbundled\_SB";

**QUIT**;

**data** Unb\_CHEMO\_STEP\_BY\_STEP; set new2d;

rc\_qr2\_factor=&rc\_qr2\_factor.;

TOTAL\_price\_postQR2\_cb=PRICE\*rc\_qr2\_factor;

SMOOTHING\_factor=&SMOOTHING\_factor\_Chemo.;

prices\_SF=TOTAL\_price\_postQR2\_cb\*(**1**+SMOOTHING\_factor);

Scaling\_factor=&Scaling\_factor.;

prices\_after\_SCALING=prices\_SF\*(**1**+Scaling\_factor);

prices\_after\_SCALING=prices\_after\_SCALING\*(**1**+&cashio\_factor\_SB.);

Infla\_Effi\_final=&Infla\_Effi\_final\_year\_uplift.;

final\_prices=prices\_after\_SCALING\*(**1**+Infla\_Effi\_final);

format TOTAL\_price\_postQR2\_cb prices\_SF prices\_after\_SCALING final\_prices comma15.0;

format rc\_qr1\_factor SMOOTHING\_factor /\*Scaling\_factor\*/ Infla\_Effi\_final percentn7.2;

format Scaling\_factor percentn8.2;

**run**;

**data** ModelOut.Unb\_CHEMO\_STEP\_BY\_STEP\_1920;

set Unb\_CHEMO\_STEP\_BY\_STEP;

**run**;

**data** Unb\_chemo\_PRICES (keep=Department Currency Hrg\_name final\_prices); set Unb\_chemo\_STEP\_BY\_STEP; **run**;

**data** ModelOut.Unb\_chemo\_PRICES\_1920;

set Unb\_chemo\_PRICES;

**run**;

**proc** **export** data=Unb\_chemo\_STEP\_BY\_STEP

outfile="&outputpath.\UB\_Chemo\_Prices\_1920.csv"

dbms=csv

replace;

**run**;

/\* ---------------------- 5 ----------------------------------------------\*/

**data** UNBUNDLED\_PRICES\_4\_IA\_model;

set rd\_prices rn\_prices rad\_prices che\_prices;

format tariff comma16.15; /\* PID added 20171019 this line to keep decimal places to 15 places \*/

**run**;

**Data** ModelOut.UNBUNDLED\_PRICES\_4\_IA\_model;

set UNBUNDLED\_PRICES\_4\_IA\_model;

**run**;

**proc** **export** data=UNBUNDLED\_PRICES\_4\_IA\_model

outfile="&outputpath.\Unbundled\_Prices\_1920\_4IA\_MODELS.csv"

dbms=csv

replace;

**run**;

**proc** **sql**;

create table new1e as select a.\*,/\*hrg\_name\*/ B.HRG\_CODE,B.PRICE\_DECISION,c.PostMA\_PRICE from Renal\_CKD\_post\_qr1\_cb\_ie as a

left join UBMA.MA\_UNBUNDLED\_Input as b

on a.currency/\*a.HRG\_CODE\*/=b.HRG\_CODE

LEFT JOIN UBMA.MA\_UNBUNDLED\_Output

as c

on

a.currency/\*a.HRG\_CODE\*/=c.HRG\_CODE

**quit**;

**data** new2e(drop=PostMA\_PRICE);

set new1e;

IF currency="" then currency=hrg\_code;else currency=currency;

if price\_decision="Adjusted" then Price=PostMA\_PRICE;

else Price=Final\_Prices\_Renal\_CKD;

format Price Comma7.0;

quantum\_pre\_MA=TOTAL\_ACTIVITY\*final\_prices\_Renal\_CKD;

quantum\_post\_MA=total\_activity\*Price;

**run**;

**proc** **sql**; create table new3e as select sum(quantum\_pre\_MA)/sum(quantum\_post\_MA) as qr2\_ma from new2e; **quit**;

**proc** **sql**;

select qr2\_ma into :qr2\_factor

from new3e;

**quit**;

%let rc\_qr2\_factor=&qr2\_factor.;

/\* ---------------------- 5 ----------------------------------------------\*/

**data** Unb\_RENAL\_STEP\_BY\_STEP; set new2e;

rc\_qr2\_factor=&rc\_qr2\_factor.;

TOTAL\_price\_postQR2\_cb=PRICE\*rc\_qr2\_factor;

SMOOTHING\_factor=&SMOOTHING\_factor\_Renal.;

prices\_SF=TOTAL\_price\_postQR2\_cb\*(**1**+SMOOTHING\_factor);

Scaling\_factor=&Scaling\_factor.;

prices\_after\_SCALING=prices\_SF\*(**1**+Scaling\_factor);

Infla\_Effi\_final=&Infla\_Effi\_final\_year\_uplift.;

final\_prices=prices\_after\_SCALING\*(**1**+Infla\_Effi\_final);

format TOTAL\_price\_postQR2\_cb prices\_SF prices\_after\_SCALING final\_prices comma15.0;

format rc\_qr1\_factor SMOOTHING\_factor /\*Scaling\_factor\*/ Infla\_Effi\_final percentn7.2;

format Scaling\_factor percentn8.2;

**run**;

**data** ModelOut.Unb\_RENAL\_STEP\_BY\_STEP\_1920;

set Unb\_RENAL\_STEP\_BY\_STEP;

**run**;

**data** Unb\_Renal\_PRICES(keep=Department Currency Hrg\_name final\_prices); set Unb\_Renal\_STEP\_BY\_STEP; **run**;

**proc** **export** data=Unb\_RENAL\_STEP\_BY\_STEP

outfile="&outputpath.\UB\_RenalCKD\_Prices\_1920.csv"

dbms=csv

replace;

**run**;

**data** ModelOut.Unb\_Renal\_PRICES\_1920;

set Unb\_Renal\_PRICES;

**run**;

\_