```
/* ---------------------------------------
Code exported from SAS Enterprise Guide
DATE: vrijdag 23 augustus 2019     TIME: 15:58:32
PROJECT: DataPrep_MA_MClerx

--------------------------------------- */

/* Library assignment for SASApp.DWH_MST */
;
/* Library assignment for SASApp.AIDDWH */
;
/* Library assignment for SASApp.DWH_SPN */
;
/* Library assignment for SASApp._MEST */
Libname _MEST V9 'E:\Users\_KenO_Datascience\010 Mestfraude\02
Netwerkanalyse\Data' ;
/* Library assignment for SASApp.ST_SPN */
;

/* Conditionally delete set of tables or views, if they exists          */
/* If the member does not exist, then no action is performed   */
%macro _eg_conditional_dropds /parmbuff;

        %local num;
        %local stepneeded;
        %local stepstarted;
        %local dsname;
        %local name;

        %let num=1;
        /* flags to determine whether a PROC SQL step is needed */
        /* or even started yet                                  */
        %let stepneeded=0;
        %let stepstarted=0;
        %let dsname= %qscan(&syspbuff,&num,',()');
        %do %while(&dsname ne);
                %let name = %sysfunc(left(&dsname));
                %if %qsysfunc(exist(&name)) %then %do;
                        %let stepneeded=1;
                        %if (&stepstarted eq 0) %then %do;
                                proc sql;
                                %let stepstarted=1;

                        %end;
                                drop table &name;
                %end;

                %if %sysfunc(exist(&name,view)) %then %do;
                        %let stepneeded=1;
                        %if (&stepstarted eq 0) %then %do;
                                proc sql;
                                %let stepstarted=1;
```

```
                        %end;
                                drop view &name;
                %end;
                %let num=%eval(&num+1);
        %let dsname=%qscan(&syspbuff,&num,',()');
        %end;
        %if &stepstarted %then %do;
                quit;
        %end;
%mend _eg_conditional_dropds;


/* Build where clauses from stored process parameters */
%macro _eg_WhereParam( COLUMN, PARM, OPERATOR, TYPE=S, MATCHALL=_ALL_VALUES_,
MATCHALL_CLAUSE=1, MAX= , IS_EXPLICIT=0, MATCH_CASE=1);

  %local q1 q2 sq1 sq2;
  %local isEmpty;
  %local isEqual isNotEqual;
  %local isIn isNotIn;
  %local isString;
  %local isBetween;

  %let isEqual = ("%QUPCASE(&OPERATOR)" = "EQ" OR "&OPERATOR" = "=");
  %let isNotEqual = ("%QUPCASE(&OPERATOR)" = "NE" OR "&OPERATOR" = "<>");
  %let isIn = ("%QUPCASE(&OPERATOR)" = "IN");
  %let isNotIn = ("%QUPCASE(&OPERATOR)" = "NOT IN");
  %let isString = (%QUPCASE(&TYPE) eq S or %QUPCASE(&TYPE) eq STRING );
  %if &isString %then
  %do;
        %if "&MATCH_CASE" eq "0" %then %do;
                %let COLUMN = %str(UPPER%(&COLUMN%));
        %end;
        %let q1=%str(%");
        %let q2=%str(%");
        %let sq1=%str(%');
        %let sq2=%str(%');
  %end;
  %else %if %QUPCASE(&TYPE) eq D or %QUPCASE(&TYPE) eq DATE %then
  %do;
    %let q1=%str(%");
    %let q2=%str(%"d);
        %let sq1=%str(%');
    %let sq2=%str(%');
  %end;
  %else %if %QUPCASE(&TYPE) eq T or %QUPCASE(&TYPE) eq TIME %then
  %do;
    %let q1=%str(%");
    %let q2=%str(%"t);
        %let sq1=%str(%');
    %let sq2=%str(%');
  %end;
```

```
  %else %if %QUPCASE(&TYPE) eq DT or %QUPCASE(&TYPE) eq DATETIME %then
  %do;
    %let q1=%str(%");
    %let q2=%str(%"dt);
        %let sq1=%str(%');
    %let sq2=%str(%');
  %end;
  %else
  %do;
    %let q1=;
    %let q2=;
        %let sq1=;
    %let sq2=;
  %end;

  %if "&PARM" = "" %then %let PARM=&COLUMN;

  %let isBetween = ("%QUPCASE(&OPERATOR)"="BETWEEN" or
"%QUPCASE(&OPERATOR)"="NOT BETWEEN");

  %if "&MAX" = "" %then %do;
    %let MAX = &parm._MAX;
    %if &isBetween %then %let PARM = &parm._MIN;
  %end;

  %if not %symexist(&PARM) or (&isBetween and not %symexist(&MAX)) %then %do;
    %if &IS_EXPLICIT=0 %then %do;
                not &MATCHALL_CLAUSE
        %end;
        %else %do;
            not 1=1
        %end;
  %end;
  %else %if "%qupcase(&&&PARM)" = "%qupcase(&MATCHALL)" %then %do;
    %if &IS_EXPLICIT=0 %then %do;
            &MATCHALL_CLAUSE
        %end;
        %else %do;
            1=1
        %end;
  %end;
  %else %if (not %symexist(&PARM._count)) or &isBetween %then %do;
    %let isEmpty = ("&&&PARM" = "");
    %if (&isEqual AND &isEmpty AND &isString) %then
       &COLUMN is null;
    %else %if (&isNotEqual AND &isEmpty AND &isString) %then
       &COLUMN is not null;
    %else %do;
           %if &IS_EXPLICIT=0 %then %do;
           &COLUMN &OPERATOR
                       %if "&MATCH_CASE" eq "0" %then %do;
                              %unquote(&q1)%QUPCASE(&&&PARM)%unquote(&q2)
```

```
                        %end;
                        %else %do;
                                %unquote(&q1)&&&PARM%unquote(&q2)
                        %end;
           %end;
           %else %do;
               &COLUMN &OPERATOR
                        %if "&MATCH_CASE" eq "0" %then %do;

%unquote(%nrstr(&sq1))%QUPCASE(&&&PARM)%unquote(%nrstr(&sq2))
                        %end;
                        %else %do;

%unquote(%nrstr(&sq1))&&&PARM%unquote(%nrstr(&sq2))
                        %end;
           %end;
       %if &isBetween %then
          AND %unquote(&q1)&&&MAX%unquote(&q2);
    %end;
  %end;
  %else
  %do;
        %local emptyList;
        %let emptyList = %symexist(&PARM._count);
        %if &emptyList %then %let emptyList = &&&PARM._count = 0;
        %if (&emptyList) %then
        %do;
                %if (&isNotin) %then
                   1;
                %else
                        0;
        %end;
        %else %if (&&&PARM._count = 1) %then
    %do;
      %let isEmpty = ("&&&PARM" = "");
      %if (&isIn AND &isEmpty AND &isString) %then
        &COLUMN is null;
      %else %if (&isNotin AND &isEmpty AND &isString) %then
        &COLUMN is not null;
      %else %do;
            %if &IS_EXPLICIT=0 %then %do;
                        %if "&MATCH_CASE" eq "0" %then %do;
                                &COLUMN &OPERATOR
(%unquote(&q1)%QUPCASE(&&&PARM)%unquote(&q2))
                        %end;
                        %else %do;
                                &COLUMN &OPERATOR
(%unquote(&q1)&&&PARM%unquote(&q2))
                        %end;
            %end;
                %else %do;
                    &COLUMN &OPERATOR (
```

```
                        %if "&MATCH_CASE" eq "0" %then %do;

%unquote(%nrstr(&sq1))%QUPCASE(&&&PARM)%unquote(%nrstr(&sq2)))
                        %end;
                        %else %do;

%unquote(%nrstr(&sq1))&&&PARM%unquote(%nrstr(&sq2)))
                        %end;
                %end;
          %end;
    %end;
    %else
    %do;
       %local addIsNull addIsNotNull addComma;
       %let addIsNull = %eval(0);
       %let addIsNotNull = %eval(0);
       %let addComma = %eval(0);
       (&COLUMN &OPERATOR (
       %do i=1 %to &&&PARM._count;
          %let isEmpty = ("&&&PARM&i" = "");
          %if (&isString AND &isEmpty AND (&isIn OR &isNotIn)) %then
          %do;
             %if (&isIn) %then %let addIsNull = 1;
             %else %let addIsNotNull = 1;
          %end;
          %else
          %do;
            %if &addComma %then %do;,%end;
                        %if &IS_EXPLICIT=0 %then %do;
                                %if "&MATCH_CASE" eq "0" %then %do;

%unquote(&q1)%QUPCASE(&&&PARM&i)%unquote(&q2)
                                %end;
                                %else %do;
                                        %unquote(&q1)&&&PARM&i%unquote(&q2)
                                %end;
                        %end;
                        %else %do;
                                %if "&MATCH_CASE" eq "0" %then %do;

%unquote(%nrstr(&sq1))%QUPCASE(&&&PARM&i)%unquote(%nrstr(&sq2))
                                %end;
                                %else %do;

%unquote(%nrstr(&sq1))&&&PARM&i%unquote(%nrstr(&sq2))
                                %end;
                        %end;
            %let addComma = %eval(1);
          %end;
       %end;)
       %if &addIsNull %then OR &COLUMN is null;
       %else %if &addIsNotNull %then AND &COLUMN is not null;
```

```
       %do;)
       %end;
    %end;
  %end;
%mend _eg_WhereParam;


/* ---------------------------------- */
/* MACRO: enterpriseguide             */
/* PURPOSE: define a macro variable   */
/*   that contains the file system    */
/*   path of the WORK library on the  */
/*   server.  Note that different     */
/*   logic is needed depending on the */
/*   server type.                     */
/* ---------------------------------- */
%macro enterpriseguide;
%global sasworklocation;
%local tempdsn unique_dsn path;

%if &sysscp=OS %then %do; /* MVS Server */
        %if %sysfunc(getoption(filesystem))=MVS %then %do;
        /* By default, physical file name will be considered a classic MVS data
set. */
            /* Construct dsn that will be unique for each concurrent session
under a particular account: */
                filename egtemp '&egtemp' disp=(new,delete); /* create a
temporary data set */
                %let tempdsn=%sysfunc(pathname(egtemp)); /* get dsn */
                filename egtemp clear; /* get rid of data set - we only wanted
its name */
                %let unique_dsn=".EGTEMP.%substr(&tempdsn, 1, 16).PDSE";
                filename egtmpdir &unique_dsn
                        disp=(new,delete,delete) space=(cyl,(5,5,50))
                        dsorg=po dsntype=library recfm=vb
                        lrecl=8000 blksize=8004 ;
                options fileext=ignore ;
        %end;
        %else %do;
        /*
                By default, physical file name will be considered an HFS
                (hierarchical file system) file.
                */
                %if "%sysfunc(getoption(filetempdir))"="" %then %do;
                        filename egtmpdir '/tmp';
                %end;
                %else %do;
                        filename egtmpdir "%sysfunc(getoption(filetempdir))";
                %end;
        %end;
        %let path=%sysfunc(pathname(egtmpdir));
    %let sasworklocation=%sysfunc(quote(&path));
```

```
%end; /* MVS Server */
%else %do;
        %let sasworklocation = "%sysfunc(getoption(work))/";
%end;
%if &sysscp=VMS_AXP %then %do; /* Alpha VMS server */
        %let sasworklocation = "%sysfunc(getoption(work))";

%end;
%if &sysscp=CMS %then %do;
        %let path = %sysfunc(getoption(work));
        %let sasworklocation = "%substr(&path, %index(&path,%str( )))";
%end;
%mend enterpriseguide;

%enterpriseguide


/* save the current settings of XPIXELS and YPIXELS */
/* so that they can be restored later               */
%macro _sas_pushchartsize(new_xsize, new_ysize);
        %global _savedxpixels _savedypixels;
        options nonotes;
        proc sql noprint;
        select setting into :_savedxpixels
        from sashelp.vgopt
        where optname eq "XPIXELS";
        select setting into :_savedypixels
        from sashelp.vgopt
        where optname eq "YPIXELS";
        quit;
        options notes;
        GOPTIONS XPIXELS=&new_xsize YPIXELS=&new_ysize;
%mend _sas_pushchartsize;

/* restore the previous values for XPIXELS and YPIXELS */
%macro _sas_popchartsize;
        %if %symexist(_savedxpixels) %then %do;
                GOPTIONS XPIXELS=&_savedxpixels YPIXELS=&_savedypixels;
                %symdel _savedxpixels / nowarn;
                %symdel _savedypixels / nowarn;
        %end;
%mend _sas_popchartsize;


ODS PROCTITLE;
OPTIONS DEV=PNG;
GOPTIONS XPIXELS=0 YPIXELS=0;
FILENAME EGSRX TEMP;
ODS tagsets.sasreport13(ID=EGSRX) FILE=EGSRX
    STYLE=HTMLBlue

STYLESHEET=(URL="file:///D:/applic/SAS/SASHome/x86/SASEnterpriseGuide/7.1/Styles
```

```
/HTMLBlue.css")
    NOGTITLE
    NOGFOOTNOTE
    GPATH=&sasworklocation
    ENCODING=UTF8
    options(rolap="on")
;

/*   START OF NODE: initieer _MEST   */

GOPTIONS ACCESSIBLE;
libname _MEST 'E:\Users\_KenO_Datascience\010 Mestfraude\02
Netwerkanalyse\Data';

GOPTIONS NOACCESSIBLE;
%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;
%LET _SASPROGRAMFILE=;
%LET _SASPROGRAMFILEHOST=;


/*   START OF NODE: Selecteer 2018   */
%LET _CLIENTTASKLABEL='Selecteer 2018';
%LET _CLIENTPROCESSFLOWNAME='preparatie VDM data';
%LET _CLIENTPROJECTPATH='E:\Users\_KenO_Datascience\010 Mestfraude\02
Netwerkanalyse\Datapreparatie.egp';
%LET _CLIENTPROJECTPATHHOST='w1052p0033.cicapp.nl';
%LET _CLIENTPROJECTNAME='Datapreparatie.egp';

GOPTIONS ACCESSIBLE;
%_eg_conditional_dropds(WORK.VDM_18);

PROC SQL;
   CREATE TABLE WORK.VDM_18 AS
   SELECT t1.STATUS_TEKST,
          t1.DATUM_LADEN,
          t1.DATUM_LOSSEN,
          t1.PCS_ID,
          t1.ITEMKEY,
          t1.SOORT_OPGAVE,
          t1.DIENSTSLEUTEL_INZ,
          t1.TYPE_DIENSTSLEUTEL_INZ,
          t1.VDM_NUMMER,
          t1.STATUS,
          t1.DATUM_STATUS,
          t1.DATUM_REGISTRATIE,
          t1.IND_AANTEKENING,
          t1.TYPE_DIENSTSLEUTEL_LEV,
          t1.REGISTRATIENR_SILO_LEV,
```

```
t1.DIENSTSLEUTEL_LEV,
t1.NAAM_LEV,
t1.POSTCODE_PLAATSCODE_LEV,
t1.POSTCODE_LETTERCODE_LEV,
t1.HUISNUMMER_LEV,
t1.MESTCODE_1,
t1.PERCENTAGE_1,
t1.MESTCODE_2,
t1.PERCENTAGE_2,
t1.MESTCODE_3,
t1.PERCENTAGE_3,
t1.MESTCODE_4,
t1.PERCENTAGE_4,
t1.TYPE_DIENSTSLEUTEL_OV_B,
t1.DIENSTSLEUTEL_OV_BETROKKENE,
t1.NAAM_OV_BETROKKENE,
t1.TYPE_DIENSTSLEUTEL_VERV,
t1.DIENSTSLEUTEL_VERV,
t1.NAAM_VERV,
t1.POSTCODE_PLAATSCODE_VERV,
t1.POSTCODE_LETTERCODE_VERV,
t1.HUISNUMMER_VERV,
t1.POSTCODE_PLAATSCODE_LADEN,
t1.POSTCODE_LETTERCODE_LADEN,
t1.TIJDSTIP_UREN_LADEN,
t1.TIJDSTIP_MINUTEN_LADEN,
t1.POSTCODE_PLAATSCODE_LOSSEN,
t1.POSTCODE_LETTERCODE_LOSSEN,
t1.TIJDSTIP_UREN_LOSSEN,
t1.TIJDSTIP_MINUTEN_LOSSEN,
t1.KENTEKEN_VOERTUIG,
t1.KENTEKEN,
t1.KENTEKEN1,
t1.KENTEKEN2,
t1.KENTEKEN3,
t1.REGNR_OPLEGGER_AANHANGER,
t1.CMR_NUMMER,
t1.CODE_MONSTERNEMER,
t1.NR_MONSTERVERPAKKING_1,
t1.NR_MONSTERVERPAKKING_2,
t1.IND_MENGMONSTER_JA,
t1.IND_MENGMONSTER_NEE,
t1.CODE_LABORATORIUM,
t1.TYPE_DIENSTSLEUTEL_AFN,
t1.REGISTRATIENR_SILO_AFN,
t1.DIENSTSLEUTEL_AFN,
t1.NAAM_AFN,
t1.POSTCODE_PLAATSCODE_AFN,
t1.POSTCODE_LETTERCODE_AFN,
t1.HUISNUMMER_AFN,
t1.CODE_OPMERKING_1,
t1.CODE_OPMERKING_2,
```

```
          t1.CODE_OPMERKING_3,
          t1.CODE_OPMERKING_4,
          t1.IND_ONDERTEKEND_LEV,
          t1.IND_ONDERTEKEND_VERV,
          t1.IND_ONDERTEKEND_AFN,
          t1.CREATED_BY,
          t1.LAST_UPDATE_DATE,
          t1.LAST_UPDATED_BY,
          t1.CREATION_DATE,
          t1.AANTAL_KG_FOSF_VERF,
          t1.AANTAL_KG_STIKST_VERF,
          t1.BRON,
          t1.IND_MENGMONGSTER,
          t1.TMG_VDM_NUMMER,
          t1.TMG_ITEMKEY,
          t1.AANTAL_KG_VRACHT,
          t1.IND_MENGMONSTER,
          t1.SOORT_MEST,
          t1.GEWICHT_VRACHT_GESCHAT,
          t1.GEWICHT_VRACHT_GEWOGEN,
          t1.BUNDELNUMMER
      FROM DWH_MST.DM_VDM_VERVOERSBEWIJZEN t1
      WHERE t1.DATUM_LADEN BETWEEN '1jan2018:00:00:00'dt AND
'31dec2018:00:00:00'dt;
QUIT;

GOPTIONS NOACCESSIBLE;


%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: Selecteer regio   */
%LET SYSLAST=WORK.VDM_18;
%LET _CLIENTTASKLABEL='Selecteer regio';
%LET _CLIENTPROCESSFLOWNAME='preparatie VDM data';
%LET _CLIENTPROJECTPATH='E:\Users\_KenO_Datascience\010 Mestfraude\02
Netwerkanalyse\Datapreparatie.egp';
%LET _CLIENTPROJECTPATHHOST='w1052p0033.cicapp.nl';
%LET _CLIENTPROJECTNAME='Datapreparatie.egp';
%LET _SASPROGRAMFILE='';
%LET _SASPROGRAMFILEHOST='';

GOPTIONS ACCESSIBLE;
data VDM_REGIO;
  set VDM_18;

  if anyalpha(POSTCODE_PLAATSCODE_LADEN)=0 then do; /*Output als postcode alleen
```

```
getallen bevat*/
     if input(substr(POSTCODE_PLAATSCODE_LADEN,1,2), best4.) in (52, 54, 55, 56,
57) or
     input(substr(POSTCODE_PLAATSCODE_LADEN,1,3), best4.) in (501, 506, 507,
508, 509, 515, 534, 535, 536, 537, 538,
                                                              539, 580, 581, 582,
583, 584, 596, 597, 598, 600, 602,
                                                              603) or
     POSTCODE_PLAATSCODE_LADEN in (5021, 5022, 5025, 5026, 5037, 5038, 5041,
5042, 5044, 5046, 5048, 5049, 5051, 5052,
                                   5053, 5056, 5057, 5059, 5111, 5131, 5133, 5142
,5145, 5171, 5175, 5856, 5861, 5862,
                                   5863, 5864, 5865, 5866, 5871, 5872, 5921, 5922,
5923, 5924, 5925, 5926, 5927, 5928,
                                   5931, 5935, 5941, 5943, 5944, 5991, 5993, 5995,
6011, 6012, 6013, 6014, 6015, 6081,
                                   6083, 6085, 6086, 6088, 6089, 6091, 6092, 6093,
6095, 6096, 6097, 6099) then output;
  end;
run;

GOPTIONS NOACCESSIBLE;
%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;
%LET _SASPROGRAMFILE=;
%LET _SASPROGRAMFILEHOST=;


/*   START OF NODE: Selecteer varkensmest   */
%LET SYSLAST=WORK.VDM_REGIO;
%LET _CLIENTTASKLABEL='Selecteer varkensmest';
%LET _CLIENTPROCESSFLOWNAME='preparatie VDM data';
%LET _CLIENTPROJECTPATH='E:\Users\_KenO_Datascience\010 Mestfraude\02
Netwerkanalyse\Datapreparatie.egp';
%LET _CLIENTPROJECTPATHHOST='w1052p0033.cicapp.nl';
%LET _CLIENTPROJECTNAME='Datapreparatie.egp';
%LET _SASPROGRAMFILE='';
%LET _SASPROGRAMFILEHOST='';

GOPTIONS ACCESSIBLE;
data WORK.VDM_VARK;
  set WORK.VDM_REGIO;

  GEWICHT_VRACHT=AANTAL_KG_VRACHT/1000;

  /*Combineren postcodecijfers en -letters.*/
  POSTCODELADEN=POSTCODE_PLAATSCODE_LADEN||POSTCODE_LETTERCODE_LADEN;
  POSTCODELOSSEN=POSTCODE_PLAATSCODE_LOSSEN||POSTCODE_LETTERCODE_LOSSEN;
```

```
  POSTCODE_LEV=POSTCODE_PLAATSCODE_LEV||POSTCODE_LETTERCODE_LEV;
  POSTCODE_AFN=POSTCODE_PLAATSCODE_AFN||POSTCODE_LETTERCODE_AFN;
  POSTCODE_VERV=POSTCODE_PLAATSCODE_VERV||POSTCODE_LETTERCODE_VERV;

  BRS_LEV=input(DIENSTSLEUTEL_LEV, best20.);
  BRS_AFN=input(DIENSTSLEUTEL_AFN,best20.);
  BRS_VERV=input(DIENSTSLEUTEL_VERV,best20.);

  if (MESTCODE_1 in (40, 41, 42, 43, 46, 50) or
      MESTCODE_2 in (40, 41, 42, 43, 46, 50) or
          MESTCODE_3 in (40, 41, 42, 43, 46, 50) or
          MESTCODE_4 in (40, 41, 42, 43, 46, 50))
          then output;
run;


GOPTIONS NOACCESSIBLE;
%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;
%LET _SASPROGRAMFILE=;
%LET _SASPROGRAMFILEHOST=;


/*   START OF NODE: TIJDELIJK Selecteer nederlandse laad en los plaats   */
%LET SYSLAST=WORK.VDM_VARK;
%LET _CLIENTTASKLABEL='TIJDELIJK Selecteer nederlandse laad en los plaats';
%LET _CLIENTPROCESSFLOWNAME='preparatie VDM data';
%LET _CLIENTPROJECTPATH='E:\Users\_KenO_Datascience\010 Mestfraude\02
Netwerkanalyse\Datapreparatie.egp';
%LET _CLIENTPROJECTPATHHOST='w1052p0033.cicapp.nl';
%LET _CLIENTPROJECTNAME='Datapreparatie.egp';
%LET _SASPROGRAMFILE='';
%LET _SASPROGRAMFILEHOST='';

GOPTIONS ACCESSIBLE;
data WORK.VDM_VARK_NL;
  set WORK.VDM_VARK;

  /* tijdelijke selectie van nederlandse postcodes */

  if prxmatch('/^\d\d\d\d[a-zA-Z][a-zA-Z]$/',trim(POSTCODELADEN)) AND
         prxmatch('/^\d\d\d\d[a-zA-Z][a-zA-Z]$/',trim(POSTCODELOSSEN))
         then output;
run;


GOPTIONS NOACCESSIBLE;
%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
```

```
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;
%LET _SASPROGRAMFILE=;
%LET _SASPROGRAMFILEHOST=;


/*   START OF NODE: Test dataset maken SAS-VI   */
%LET _CLIENTTASKLABEL='Test dataset maken SAS-VI';
%LET _CLIENTPROCESSFLOWNAME='preparatie VDM data';
%LET _CLIENTPROJECTPATH='E:\Users\_KenO_Datascience\010 Mestfraude\02
Netwerkanalyse\Datapreparatie.egp';
%LET _CLIENTPROJECTPATHHOST='w1052p0033.cicapp.nl';
%LET _CLIENTPROJECTNAME='Datapreparatie.egp';

GOPTIONS ACCESSIBLE;
%_eg_conditional_dropds(WORK.VDM_SASVI);

PROC SQL;
   CREATE TABLE WORK.VDM_SASVI AS
   SELECT t1.VDM_NUMMER,
          /* MAAND_LADEN */
            (month(datepart(datum_LADEN))) LABEL="MAAND_LADEN" AS MAAND_LADEN,
          t1.DATUM_LADEN,
          t1.DATUM_LOSSEN,
          t1.BRS_LEV,
          t1.BRS_AFN,
          t1.BRS_VERV,
          t1.POSTCODELADEN,
          t1.POSTCODELOSSEN
      FROM WORK.VDM_VARK_NL t1
      WHERE t1.BRS_LEV NOT IS MISSING AND t1.BRS_AFN NOT IS MISSING AND
t1.BRS_VERV NOT IS MISSING;
QUIT;

GOPTIONS NOACCESSIBLE;


%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: Selecteer unieke postcode huisnummer   */

GOPTIONS ACCESSIBLE;
%_eg_conditional_dropds(WORK.PCH_UNIEK);

PROC SQL;
   CREATE TABLE WORK.PCH_UNIEK AS
```

```
   SELECT t1.PC6H,
          /* VA_LATITUDE_ADRES */
            (MAX(t1.VA_LATITUDE_ADRES)) AS VA_LATITUDE_ADRES,
          /* VA_LONGITUDE_ADRES */
            (MAX(t1.VA_LONGITUDE_ADRES)) AS VA_LONGITUDE_ADRES
      FROM AIDDWH.DM_BAG t1
      GROUP BY t1.PC6H;
QUIT;

GOPTIONS NOACCESSIBLE;


%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: BRS gegevens informatie   */

GOPTIONS ACCESSIBLE;
%_eg_conditional_dropds(WORK.CVU_BRS);

PROC SQL;
   CREATE TABLE WORK.CVU_BRS AS
   SELECT DISTINCT /* BRS */
                     (input(t1.BRS_NUMMER,11.)) AS BRS,
          t1.AID_NUMMER,
          t1.RELATIENAAM,
          t1.VOORVOEGSEL,
          t1.VOORLETTERS,
          t1.STRAATNAAM,
          t1.HUISNUMMER,
          t1.TOEVOEGING,
          t1.POSTCODE_PLAATSCODE,
          t1.POSTCODE_LETTERCODE,
          t1.PLAATS,
          t1.PROVINCIE,
          t1.DATUM_EINDE
      FROM AIDDWH.CVU_RELATIES_99 t1;
QUIT;

GOPTIONS NOACCESSIBLE;


%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;
```

```
/*   START OF NODE: max AID nummer   */

GOPTIONS ACCESSIBLE;
%_eg_conditional_dropds(WORK.CVU_UNIAID);

PROC SQL;
   CREATE TABLE WORK.CVU_UNIAID AS
   SELECT t1.BRS,
          /* AID_NUMMER */
            (MAX(t1.AID_NUMMER)) FORMAT=12. AS AID_NUMMER,
          t1.RELATIENAAM,
          t1.VOORVOEGSEL,
          t1.VOORLETTERS,
          t1.STRAATNAAM,
          t1.HUISNUMMER,
          t1.TOEVOEGING,
          t1.POSTCODE_PLAATSCODE,
          t1.POSTCODE_LETTERCODE,
          t1.PLAATS,
          t1.PROVINCIE,
          t1.DATUM_EINDE
      FROM WORK.CVU_BRS t1
      GROUP BY t1.BRS,
               t1.RELATIENAAM,
               t1.VOORVOEGSEL,
               t1.VOORLETTERS,
               t1.STRAATNAAM,
               t1.HUISNUMMER,
               t1.TOEVOEGING,
               t1.POSTCODE_PLAATSCODE,
               t1.POSTCODE_LETTERCODE,
               t1.PLAATS,
               t1.PROVINCIE,
               t1.DATUM_EINDE;
QUIT;

GOPTIONS NOACCESSIBLE;


%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: CHECK Dubbele BRS nummers   */
%LET _CLIENTTASKLABEL='CHECK Dubbele BRS nummers';
%LET _CLIENTPROCESSFLOWNAME='Datapreparatie Nodes - BRS';
%LET _CLIENTPROJECTPATH='E:\Users\_KenO_Datascience\010 Mestfraude\02
Netwerkanalyse\Datapreparatie.egp';
```

```
%LET _CLIENTPROJECTPATHHOST='w1052p0033.cicapp.nl';
%LET _CLIENTPROJECTNAME='Datapreparatie.egp';

GOPTIONS ACCESSIBLE;
%_eg_conditional_dropds(WORK.CVU_BRS_DUBBEL);

PROC SQL;
   CREATE TABLE WORK.CVU_BRS_DUBBEL AS
   SELECT DISTINCT t1.BRS,
          /* COUNT_of_BRS */
            (COUNT(t1.BRS)) AS COUNT_of_BRS
      FROM WORK.CVU_BRS t1
      GROUP BY t1.BRS
      ORDER BY COUNT_of_BRS DESC;
QUIT;

GOPTIONS NOACCESSIBLE;


%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: Selecteer unieke lev BRS   */

GOPTIONS ACCESSIBLE;
%_eg_conditional_dropds(WORK.VDM_SASVI_LEV);

PROC SQL;
   CREATE TABLE WORK.VDM_SASVI_LEV AS
   SELECT DISTINCT t1.BRS_LEV AS BRS,
          /* Leverancier */
            ('Leverancier') AS Leverancier
      FROM WORK.VDM_SASVI t1;
QUIT;

GOPTIONS NOACCESSIBLE;


%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: Selecteer unieke afn BRS   */

GOPTIONS ACCESSIBLE;
```

```
%_eg_conditional_dropds(WORK.VDM_SASVI_AFN);

PROC SQL;
   CREATE TABLE WORK.VDM_SASVI_AFN AS
   SELECT DISTINCT t1.BRS_AFN AS BRS,
          /* Afnemer */
            ('Afnemer') AS Afnemer
      FROM WORK.VDM_SASVI t1;
QUIT;

GOPTIONS NOACCESSIBLE;


%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: Selecteer unieke verv BRS   */

GOPTIONS ACCESSIBLE;
%_eg_conditional_dropds(WORK.VDM_SASVI_VERV);

PROC SQL;
   CREATE TABLE WORK.VDM_SASVI_VERV AS
   SELECT DISTINCT t1.BRS_VERV AS BRS,
          /* Vervoerder */
            ('Vervoeder') AS Vervoerder
      FROM WORK.VDM_SASVI t1;
QUIT;

GOPTIONS NOACCESSIBLE;


%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: Plak BRS onder elkaar   */

GOPTIONS ACCESSIBLE;
%_eg_conditional_dropds(WORK.Append_Table);
PROC SQL;
CREATE TABLE WORK.Append_Table AS
SELECT * FROM WORK.VDM_SASVI_LEV
 OUTER UNION CORR
SELECT * FROM WORK.VDM_SASVI_VERV
```

```
 OUTER UNION CORR
SELECT * FROM WORK.VDM_SASVI_AFN
;
Quit;


GOPTIONS NOACCESSIBLE;
%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: BRS in VDM voor SASVI   */

GOPTIONS ACCESSIBLE;
%_eg_conditional_dropds(WORK.VDM_SASVI_UNIBRS);

PROC SQL;
   CREATE TABLE WORK.VDM_SASVI_UNIBRS AS
   SELECT DISTINCT t1.BRS
      FROM WORK.APPEND_TABLE t1;
QUIT;

GOPTIONS NOACCESSIBLE;


%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: Bedrijfsinformatie BRS VDM van SASVI   */

GOPTIONS ACCESSIBLE;
%_eg_conditional_dropds(WORK.BEDRIJF_SASVI);

PROC SQL;
   CREATE TABLE WORK.BEDRIJF_SASVI AS
   SELECT t2.BRS,
          t1.AID_NUMMER,
          t1.RELATIENAAM,
          t1.VOORVOEGSEL,
          t1.VOORLETTERS,
          t1.STRAATNAAM,
          t1.HUISNUMMER,
          t1.TOEVOEGING,
          t1.POSTCODE_PLAATSCODE,
          t1.POSTCODE_LETTERCODE,
```

```
          t1.PLAATS,
          t1.PROVINCIE,
          t1.DATUM_EINDE,
          /* PCH6 */
            (compress(t1.POSTCODE_PLAATSCODE || t1.POSTCODE_LETTERCODE ||
put(t1.HUISNUMMER,10.))) LABEL=
             "PCH6 (postcode huisnummer om BAG te koppelen)" AS PCH6
      FROM WORK.CVU_UNIAID t1
           RIGHT JOIN WORK.VDM_SASVI_UNIBRS t2 ON (t1.BRS = t2.BRS);
QUIT;

GOPTIONS NOACCESSIBLE;



%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: Koppel coordinaten   */

GOPTIONS ACCESSIBLE;
%_eg_conditional_dropds(WORK.BEDRIJF_COOR);

PROC SQL;
   CREATE TABLE WORK.BEDRIJF_COOR AS
   SELECT t1.BRS,
          t1.AID_NUMMER,
          t1.RELATIENAAM,
          t1.VOORVOEGSEL,
          t1.VOORLETTERS,
          t1.STRAATNAAM,
          t1.HUISNUMMER,
          t1.TOEVOEGING,
          t1.POSTCODE_PLAATSCODE,
          t1.POSTCODE_LETTERCODE,
          t1.PLAATS,
          t1.PROVINCIE,
          t1.DATUM_EINDE,
          t1.PCH6,
          t2.VA_LATITUDE_ADRES,
          t2.VA_LONGITUDE_ADRES
      FROM WORK.BEDRIJF_SASVI t1
           LEFT JOIN WORK.PCH_UNIEK t2 ON (t1.PCH6 = t2.PC6H);
QUIT;

GOPTIONS NOACCESSIBLE;
```

```
%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: Selecteer inspecties op programma MSW   */

GOPTIONS ACCESSIBLE;
%_eg_conditional_dropds(WORK.INSPECTIES_MSW);

PROC SQL;
   CREATE TABLE WORK.INSPECTIES_MSW AS
   SELECT t1.VTT_IDENTIFICATIE,
          t1.VDT_IDENTIFICATIE,
          t1.VERIFICATIE_PROGRAMMA,
          t1.VMA_IDENTIFICATIE,
          t1.STARTDATUM,
          t1.DATUM_GEREED,
          t1.STATUS_ACTIVITEIT,
          t1.AID_NUMMER,
          t1.BRS_NUMMER,
          t1.VDT_AKKOORD
      FROM DWH_SPN.DM_VTE_ACTIVITEITEN t1
      WHERE t1.VERIFICATIE_PROGRAMMA CONTAINS 'MSW' AND t1.STATUS_ACTIVITEIT =
'Activiteit Gereed';
QUIT;

GOPTIONS NOACCESSIBLE;


%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: Filter op inspecties gedaan na 2016   */

GOPTIONS ACCESSIBLE;
%_eg_conditional_dropds(WORK.INSPECTIES_MSW_2016);

PROC SQL;
   CREATE TABLE WORK.INSPECTIES_MSW_2016 AS
   SELECT t1.VTT_IDENTIFICATIE,
          t1.VDT_IDENTIFICATIE,
          t1.VERIFICATIE_PROGRAMMA,
          t1.VMA_IDENTIFICATIE,
          t1.STARTDATUM,
```

```
          t1.DATUM_GEREED,
          t1.STATUS_ACTIVITEIT,
          t1.AID_NUMMER,
          t1.BRS_NUMMER,
          t1.VDT_AKKOORD
      FROM WORK.INSPECTIES_MSW t1
      WHERE t1.STARTDATUM >= '1Jan2016:0:0:0'dt;
QUIT;

GOPTIONS NOACCESSIBLE;


%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: BRS en AID combi   */

GOPTIONS ACCESSIBLE;
%_eg_conditional_dropds(WORK.BRS_AID_SASVI);

PROC SQL;
   CREATE TABLE WORK.BRS_AID_SASVI AS
   SELECT t1.BRS,
          t1.AID_NUMMER
      FROM WORK.BEDRIJF_SASVI t1;
QUIT;

GOPTIONS NOACCESSIBLE;


%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: Link inspecties met BRS SASVI   */

GOPTIONS ACCESSIBLE;
%_eg_conditional_dropds(WORK.INSPECTIES_BRS_SASVI);

PROC SQL;
   CREATE TABLE WORK.INSPECTIES_BRS_SASVI AS
   SELECT t2.BRS,
          t2.AID_NUMMER,
          t1.VTT_IDENTIFICATIE,
          t1.VDT_IDENTIFICATIE,
```

```
          t1.VERIFICATIE_PROGRAMMA,
          t1.VMA_IDENTIFICATIE,
          t1.STARTDATUM,
          t1.DATUM_GEREED,
          t1.STATUS_ACTIVITEIT,
          t1.AID_NUMMER AS AID_NUMMER1,
          t1.BRS_NUMMER,
          t1.VDT_AKKOORD
      FROM WORK.INSPECTIES_MSW_2016 t1
           RIGHT JOIN WORK.BRS_AID_SASVI t2 ON (t1.AID_NUMMER = t2.AID_NUMMER)
      ORDER BY t2.BRS;
QUIT;

GOPTIONS NOACCESSIBLE;



%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: Aantal inspecties en (niet) akkoord per BRS   */
%LET _CLIENTTASKLABEL='Aantal inspecties en (niet) akkoord per BRS';
%LET _CLIENTPROCESSFLOWNAME='Datapreparatie Nodes - SPIN';
%LET _CLIENTPROJECTPATH='E:\Users\_KenO_Datascience\010 Mestfraude\02
Netwerkanalyse\Datapreparatie.egp';
%LET _CLIENTPROJECTPATHHOST='w1052p0033.cicapp.nl';
%LET _CLIENTPROJECTNAME='Datapreparatie.egp';

GOPTIONS ACCESSIBLE;
%_eg_conditional_dropds(WORK.INSPECTIES_SASVI);

PROC SQL;
   CREATE TABLE WORK.INSPECTIES_SASVI AS
   SELECT t1.BRS,
          /* AANTAL_INSPECTIES */
            ((sum(t1.VDT_AKKOORD = "N")) + (sum(t1.VDT_AKKOORD = "J"))) AS
AANTAL_INSPECTIES,
          /* AANTAL_AKKOORD */
            (sum(t1.VDT_AKKOORD = "J")) AS AANTAL_AKKOORD,
          /* AANTAL_NIET_AKKOORD */
            (sum(t1.VDT_AKKOORD = "N")) AS AANTAL_NIET_AKKOORD
      FROM WORK.INSPECTIES_BRS_SASVI t1
      GROUP BY t1.BRS
      ORDER BY AANTAL_INSPECTIES DESC;
QUIT;

GOPTIONS NOACCESSIBLE;
```

```
%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: Zet BRS onder elkaar   */
%LET _CLIENTTASKLABEL='Zet BRS onder elkaar';
%LET _CLIENTPROCESSFLOWNAME='Risicofactoren ';
%LET _CLIENTPROJECTPATH='E:\Users\_KenO_Datascience\010 Mestfraude\02
Netwerkanalyse\Datapreparatie.egp';
%LET _CLIENTPROJECTPATHHOST='w1052p0033.cicapp.nl';
%LET _CLIENTPROJECTNAME='Datapreparatie.egp';

GOPTIONS ACCESSIBLE;
/* ---------------------------------------------------------------------
   Code generated by SAS Task

   Generated on: Friday, August 23, 2019 at 3:54:28 PM
   By task: Zet BRS onder elkaar

   Input Data: SASApp:_MEST.VDM_SASVI_28_06_2019
   Server:  SASApp
   --------------------------------------------------------------------- */

%_eg_conditional_dropds(WORK.VDM_BRS_ONDERELKAAR,
                WORK.SORTTempTableSorted);
/* ---------------------------------------------------------------------
   Sort data set _MEST.VDM_SASVI_28_06_2019
   --------------------------------------------------------------------- */
PROC SORT
        DATA=_MEST.VDM_SASVI_28_06_2019(KEEP=BRS_LEV BRS_AFN BRS_VERV
VDM_NUMMER)
        OUT=WORK.SORTTempTableSorted
        ;
        BY VDM_NUMMER;
RUN;
PROC TRANSPOSE DATA=WORK.SORTTempTableSorted
        OUT=WORK.VDM_BRS_ONDERELKAAR(LABEL="Transposed
_MEST.VDM_SASVI_28_06_2019")
        PREFIX=BRS
        NAME=BRS_TYPE
        LABEL=BRS_TYPE
;
        BY VDM_NUMMER;
        VAR BRS_LEV BRS_AFN BRS_VERV;

/* ---------------------------------------------------------------------
   End of task code
   --------------------------------------------------------------------- */
```

```
RUN; QUIT;
%_eg_conditional_dropds(WORK.SORTTempTableSorted);
TITLE; FOOTNOTE;


GOPTIONS NOACCESSIBLE;
%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: RF Feestdag en Tijd   */
%LET SYSLAST=ST_SPN.SPN_FEESTDAG;
%LET _CLIENTTASKLABEL='RF Feestdag en Tijd';
%LET _CLIENTPROCESSFLOWNAME='Risicofactoren ';
%LET _CLIENTPROJECTPATH='E:\Users\_KenO_Datascience\010 Mestfraude\02
Netwerkanalyse\Datapreparatie.egp';
%LET _CLIENTPROJECTPATHHOST='w1052p0033.cicapp.nl';
%LET _CLIENTPROJECTNAME='Datapreparatie.egp';
%LET _SASPROGRAMFILE='';
%LET _SASPROGRAMFILEHOST='';

GOPTIONS ACCESSIBLE;
/*      Bepaal of datum van laden of lossen een feestdag is
        en of dit op een raar tijdsstip gebeurt.
        Zo ja dan wordt dit gezien als een risico factor. */

PROC SQL;
        CREATE TABLE WORK.VDM_RF_DATUMTIJD AS
        SELECT a.*,

                /* Risico factor feestdagen */
                CASE WHEN datepart(a.DATUM_LOSSEN)=datepart(b.DATUM) OR
                datepart(a.DATUM_LADEN)=datepart(c.DATUM)
                THEN 1 ELSE 0
                END AS RF_FEESTDAG,

                /* Risico factor tijd */
                CASE WHEN TIMEPART(a.DATUM_LOSSEN) > (23*3600) OR
TIMEPART(a.DATUM_LOSSEN) < (4*3600)
                OR TIMEPART(a.DATUM_LADEN) > (23*3600) OR
TIMEPART(a.DATUM_LADEN) < (4*3600)
                THEN 1 ELSE 0
                END AS RF_TIJD

        FROM _MEST.VDM_SASVI_28_06_2019 a
        LEFT JOIN ST_SPN.SPN_FEESTDAG b ON
datepart(a.DATUM_LOSSEN)=datepart(b.DATUM)
        LEFT JOIN St_SPN.SPN_FEESTDAG c ON
datepart(a.DATUM_LADEN)=datepart(c.DATUM);
```

```
QUIT;

GOPTIONS NOACCESSIBLE;
%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;
%LET _SASPROGRAMFILE=;
%LET _SASPROGRAMFILEHOST=;


/*   START OF NODE: RF VDM Wijzigen   */
%LET _CLIENTTASKLABEL='RF VDM Wijzigen';
%LET _CLIENTPROCESSFLOWNAME='Risicofactoren ';
%LET _CLIENTPROJECTPATH='E:\Users\_KenO_Datascience\010 Mestfraude\02
Netwerkanalyse\Datapreparatie.egp';
%LET _CLIENTPROJECTPATHHOST='w1052p0033.cicapp.nl';
%LET _CLIENTPROJECTNAME='Datapreparatie.egp';

GOPTIONS ACCESSIBLE;
%_eg_conditional_dropds(WORK.VDM_RF_VDMWIJZIG);

PROC SQL;
   CREATE TABLE WORK.VDM_RF_VDMWIJZIG AS
   SELECT t1.VDM_NUMMER,
          t1.DATUM_LADEN,
          t1.DATUM_LOSSEN,
          t1.BRS_LEV,
          t1.BRS_AFN,
          t1.BRS_VERV,
          t1.POSTCODELADEN,
          t1.POSTCODELOSSEN,
          t1.RF_FEESTDAG,
          t1.RF_TIJD,
          /* RF_VDMWIJZIG */
            (case when
            t1.VDM_NUMMER = t2.VDM_NUMMER
            then 1 else 0
            end) AS RF_VDMWIJZIG
      FROM WORK.VDM_RF_DATUMTIJD t1
           LEFT JOIN _MEST.VDM_MET_MUTATIE_2018 t2 ON (t1.VDM_NUMMER =
t2.VDM_NUMMER);
QUIT;

GOPTIONS NOACCESSIBLE;



%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
```

```
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: Link risico factoren aan    */
%LET _CLIENTTASKLABEL='Link risico factoren aan ';
%LET _CLIENTPROCESSFLOWNAME='Risicofactoren ';
%LET _CLIENTPROJECTPATH='E:\Users\_KenO_Datascience\010 Mestfraude\02
Netwerkanalyse\Datapreparatie.egp';
%LET _CLIENTPROJECTPATHHOST='w1052p0033.cicapp.nl';
%LET _CLIENTPROJECTNAME='Datapreparatie.egp';

GOPTIONS ACCESSIBLE;
%_eg_conditional_dropds(WORK.VDM_BRS_RISICOFACTOREN);

PROC SQL;
   CREATE TABLE WORK.VDM_BRS_RISICOFACTOREN AS
   SELECT t1.VDM_NUMMER AS VDM_NUMMER,
          t1.BRS_TYPE AS BRS_TYPE,
          t1.BRS1 AS BRS,
          t2.RF_FEESTDAG,
          t2.RF_TIJD,
          t2.RF_VDMWIJZIG
      FROM WORK.VDM_BRS_ONDERELKAAR t1
           LEFT JOIN WORK.VDM_RF_VDMWIJZIG t2 ON (t1.VDM_NUMMER =
t2.VDM_NUMMER);
QUIT;

GOPTIONS NOACCESSIBLE;



%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: Risicofactoren aantal tot totaal   */
%LET _CLIENTTASKLABEL='Risicofactoren aantal tot totaal';
%LET _CLIENTPROCESSFLOWNAME='Risicofactoren ';
%LET _CLIENTPROJECTPATH='E:\Users\_KenO_Datascience\010 Mestfraude\02
Netwerkanalyse\Datapreparatie.egp';
%LET _CLIENTPROJECTPATHHOST='w1052p0033.cicapp.nl';
%LET _CLIENTPROJECTNAME='Datapreparatie.egp';

GOPTIONS ACCESSIBLE;
%_eg_conditional_dropds(WORK.RISICOFACTOREN_SASVI);

PROC SQL;
   CREATE TABLE WORK.RISICOFACTOREN_SASVI AS
```

```
   SELECT t1.BRS,
          /* RF_FEESTDAG */
            ((SUM(t1.RF_FEESTDAG)) / (COUNT(t1.VDM_NUMMER)) * 100) AS
RF_FEESTDAG,
          /* RF_TIJD */
            ((SUM(t1.RF_TIJD)) / (COUNT(t1.VDM_NUMMER)) * 100) AS RF_TIJD,
          /* RF_VDMWIJZIGING */
            ((SUM(t1.RF_VDMWIJZIG)) / (COUNT(t1.VDM_NUMMER)) * 100) AS
RF_VDMWIJZIGING
      FROM WORK.VDM_BRS_RISICOFACTOREN t1
      GROUP BY t1.BRS;
QUIT;

GOPTIONS NOACCESSIBLE;


%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: Selecteer laad en los berichten 2017/2018   */
%LET _CLIENTTASKLABEL='Selecteer laad en los berichten 2017/2018';
%LET _CLIENTPROCESSFLOWNAME='GPS ontsluiten';
%LET _CLIENTPROJECTPATH='E:\Users\_KenO_Datascience\010 Mestfraude\02
Netwerkanalyse\Datapreparatie.egp';
%LET _CLIENTPROJECTPATHHOST='w1052p0033.cicapp.nl';
%LET _CLIENTPROJECTNAME='Datapreparatie.egp';

GOPTIONS ACCESSIBLE;
%_eg_conditional_dropds(WORK.LAAD_LOS_17_18);

PROC SQL;
   CREATE TABLE WORK.LAAD_LOS_17_18 AS
   SELECT DISTINCT t1.VDM_NUMMER,
          t1.LADEN_GPS_LOCATIE,
          t1.LADEN_GPS_DATUMTIJD,
          t1.LOSSEN_GPS_LOCATIE,
          t1.LOSSEN_GPS_DATUMTIJD
      FROM PMB_LLR.LLR_LAADLOSBERICHTEN t1
      WHERE t1.LADEN_GPS_DATUMTIJD BETWEEN '01jan2017:00:00:00'dt AND
'31dec2018:00:00:00'dt AND t1.COMPLEET = 'J';
QUIT;

GOPTIONS NOACCESSIBLE;
%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;
```

```
/*   START OF NODE: Unieke VDM-GPS meldingen   */
%LET _CLIENTTASKLABEL='Unieke VDM-GPS meldingen';
%LET _CLIENTPROCESSFLOWNAME='GPS ontsluiten';
%LET _CLIENTPROJECTPATH='E:\Users\_KenO_Datascience\010 Mestfraude\02
Netwerkanalyse\Datapreparatie.egp';
%LET _CLIENTPROJECTPATHHOST='w1052p0033.cicapp.nl';
%LET _CLIENTPROJECTNAME='Datapreparatie.egp';

GOPTIONS ACCESSIBLE;
%put ERROR: Unable to get SAS code. Unable to open input data;


GOPTIONS NOACCESSIBLE;



%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: Selecteer unieke complete meldingen   */
%LET _CLIENTTASKLABEL='Selecteer unieke complete meldingen';
%LET _CLIENTPROCESSFLOWNAME='GPS ontsluiten';
%LET _CLIENTPROJECTPATH='E:\Users\_KenO_Datascience\010 Mestfraude\02
Netwerkanalyse\Datapreparatie.egp';
%LET _CLIENTPROJECTPATHHOST='w1052p0033.cicapp.nl';
%LET _CLIENTPROJECTNAME='Datapreparatie.egp';

GOPTIONS ACCESSIBLE;
%put ERROR: Unable to get SAS code. Unable to open input data;


GOPTIONS NOACCESSIBLE;


%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: Splitsen GPS coordinaat   */

GOPTIONS ACCESSIBLE;
data WORK.VDM_GPS_17_18;
  set WORK.VDM_GPS_17_18_1;
```

```
  attrib LADEN_BREEDTEGRAAD length=8 format=z8.5;
  attrib LOSSEN_BREEDTEGRAAD length=8 format=z8.5;
  attrib LADEN_LENGTEGRAAD length=8 format=z8.5;
  attrib LOSSEN_LENGTEGRAAD length=8 format=z8.5;

  /*Kijken of het een geldige GPS locatie is. Er moet zowel een ,N als ,E
inzitten. Als dat niet het
    geval is, wordt de GPS locatie vervangen door een missing.*/
  if find(LADEN_GPS_LOCATIE,',N')= 0 or find(LADEN_GPS_LOCATIE,',E')=0 then
LADEN_GPS_LOCATIE='';
  if find(LOSSEN_GPS_LOCATIE,',N')= 0 or find(LOSSEN_GPS_LOCATIE,',E')=0 then
LOSSEN_GPS_LOCATIE='';

  /*extract de breedtegraad uit de string met de gpslocatie laden*/
  if LADEN_GPS_LOCATIE NE '' then
LADEN_BREEDTEGRAAD=input(substr(LADEN_GPS_LOCATIE,1,find(LADEN_GPS_LOCATIE,'N')-
2),best12.6)/100;
  if LADEN_GPS_LOCATIE NE '' then
LADEN_LENGTEGRAAD=input(substr(LADEN_GPS_LOCATIE,find(LADEN_GPS_LOCATIE,'N')+2,f
ind(LADEN_GPS_LOCATIE,'E')-find(LADEN_GPS_LOCATIE,'N')-3),best12.6)/100;
  if LOSSEN_GPS_LOCATIE NE '' then
LOSSEN_BREEDTEGRAAD=input(substr(LOSSEN_GPS_LOCATIE,1,find(LOSSEN_GPS_LOCATIE,'N
')-2),best12.6)/100;
  if LOSSEN_GPS_LOCATIE NE '' then
LOSSEN_LENGTEGRAAD=input(substr(LOSSEN_GPS_LOCATIE,find(LOSSEN_GPS_LOCATIE,'N')+
2,find(LOSSEN_GPS_LOCATIE,'E')-find(LOSSEN_GPS_LOCATIE,'N')-3),best12.6)/100;

run;



GOPTIONS NOACCESSIBLE;
%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;
%LET _SASPROGRAMFILE=;
%LET _SASPROGRAMFILEHOST=;


/*   START OF NODE: Query Builder (2)   */

GOPTIONS ACCESSIBLE;
%_eg_conditional_dropds(WORK.QUERY_FOR_DM_VTE_ACTIVITEITEN);

PROC SQL;
   CREATE TABLE WORK.QUERY_FOR_DM_VTE_ACTIVITEITEN AS
   SELECT t1.VTT_IDENTIFICATIE,
          t1.VDT_IDENTIFICATIE,
          t1.HANDHAVINGS_ASPECT,
```

```
t1.DOELGROEP,
t1.VERIFICATIE_PROGRAMMA,
t1.TEAM,
t1.VTE_RELATIENUMMER,
t1.GEPLANDE_STARTDAT,
t1.DEADLINE,
t1.STARTDATUM,
t1.DATUM_GEREED,
t1.DATUM_AFGEBROKEN,
t1.GEPLANDE_TIJD,
t1.STATUS_ACTIVITEIT,
t1.WERKELIJKE_TIJD,
t1.MKR_BADGENUMMER,
t1.MKR_NAAM,
t1.OPSPORINGSKOPPELING,
t1.TEAMLEIDER,
t1.AANTAL_MEDEWERKERS,
t1.EERSTE_BEDRIJFSBEZOEK,
t1.LAATSTE_BEDRIJFSBEZOEK,
t1.TIJD_VOOR_BEZOEK,
t1.TIJD_BEZOEK,
t1.TIJD_TUSSEN_BEZOEK,
t1.TIJD_NA_BEZOEK,
t1.TIJD_TOTAAL,
t1.AANTAL_VERSLAGEN,
t1.DATUM_STATUS_VERSLAG,
t1.AANTAL_BEZOEKEN,
t1.AANGEPASTE_DEADLINE,
t1.BEGINDATUM_TDG,
t1.EINDDATUM_TDG,
t1.STT_IDENTIFICATIE,
t1.VTE_NAAM,
t1.VTE_ADRES,
t1.VTE_LAND,
t1.VTE_POSTCODE,
t1.VTE_WOONPLAATS,
t1.VTE_EB_NAAM,
t1.VTE_EB_NUMMER,
t1.AID_NUMMER,
t1.BRS_NUMMER,
t1.REBUSNUMMER,
t1.WDG_PCHNR,
t1.WDG_PCHNRTV,
t1.HPT_VMA_IDENTIFICATIE,
t1.VMA_IDENTIFICATIE,
t1.VDT_STATUS,
t1.VDT_AKKOORD,
t1.VDT_BEGINDATUM,
t1.VDT_EINDDATUM,
t1.TOELICHTING,
t1.AANTAL_VERSLAGEN_VERZONDEN,
t1.AANTAL_VERSLAGEN_DEFINITIEF,
```

```
          t1.AANTAL_VERSLAGEN_CONCEPT,
          t1.AANTAL_VERSLAGEN_VERVALLEN
      FROM DWH_SPN.DM_VTE_ACTIVITEITEN t1
      WHERE t1.VERIFICATIE_PROGRAMMA CONTAINS 'MSW';
QUIT;

GOPTIONS NOACCESSIBLE;


%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: Selecteer unieke postcodes   */

GOPTIONS ACCESSIBLE;
%_eg_conditional_dropds(WORK.BAG_POSTCODE);

PROC SQL;
   CREATE TABLE WORK.BAG_POSTCODE AS
   SELECT DISTINCT t1.POSTCODE,
          /* GEMEENTE */
            (MAX(t1.GEMEENTE)) AS GEMEENTE,
          /* WOONPLAATS */
            (MAX(t1.WOONPLAATS)) AS WOONPLAATS
      FROM AIDDWH.DM_BAG t1
      GROUP BY t1.POSTCODE;
QUIT;

GOPTIONS NOACCESSIBLE;


%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: Koppel gemeente en woonplaats laadlos locatie   */

GOPTIONS ACCESSIBLE;
%_eg_conditional_dropds(WORK.VDM_LAADLOS);

PROC SQL;
   CREATE TABLE WORK.VDM_LAADLOS AS
   SELECT t1.BRS_LADEN,
          t3.GEMEENTE AS GEMEENTE_LADEN,
          t3.WOONPLAATS AS WOONPLAATS_LADEN,
```

```
          t1.POSTCODELADEN,
          t1.BRS_LOSSEN,
          t2.GEMEENTE AS GEMEENTE_LOSSEN,
          t2.WOONPLAATS AS WOONPLAATS_LOSSEN,
          t1.POSTCODELOSSEN
      FROM WORK.VDM_VARK t1
           LEFT JOIN WORK.BAG_POSTCODE t3 ON (t1.POSTCODELADEN = t3.POSTCODE)
           LEFT JOIN WORK.BAG_POSTCODE t2 ON (t1.POSTCODELOSSEN = t2.POSTCODE);
QUIT;

GOPTIONS NOACCESSIBLE;




%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: Aantal transacties tussen BRS   */

GOPTIONS ACCESSIBLE;
%put ERROR: Unable to get SAS code. Unable to open input data;


GOPTIONS NOACCESSIBLE;


%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: Selecteer Best voor visualisatie netwerk   */

GOPTIONS ACCESSIBLE;
%put ERROR: Unable to get SAS code. Unable to open input data;


GOPTIONS NOACCESSIBLE;


%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;
```

```
/*   START OF NODE: Filter and Sort   */

GOPTIONS ACCESSIBLE;
%_eg_conditional_dropds(WORK.FILTER_FOR_BEST_NETWERK);

PROC SQL;
   CREATE TABLE WORK.FILTER_FOR_BEST_NETWERK AS
   SELECT t1.BRS_LADEN,
          t1.GEMEENTE_LADEN,
          t1.WOONPLAATS_LADEN,
          t1.POSTCODELADEN,
          t1.BRS_LOSSEN,
          t1.GEMEENTE_LOSSEN,
          t1.WOONPLAATS_LOSSEN,
          t1.POSTCODELOSSEN,
          t1.AANTAL
      FROM _MEST.BEST_NETWERK t1
      WHERE t1.BRS_LADEN = 100090189;
QUIT;

GOPTIONS NOACCESSIBLE;


%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: Filter and Sort (2)   */

GOPTIONS ACCESSIBLE;
%_eg_conditional_dropds(WORK.FILTER_FOR_BEST_NETWERK_0000);

PROC SQL;
   CREATE TABLE WORK.FILTER_FOR_BEST_NETWERK_0000 AS
   SELECT t1.BRS_LADEN,
          t1.GEMEENTE_LADEN,
          t1.WOONPLAATS_LADEN,
          t1.POSTCODELADEN,
          t1.BRS_LOSSEN,
          t1.GEMEENTE_LOSSEN,
          t1.WOONPLAATS_LOSSEN,
          t1.POSTCODELOSSEN,
          t1.AANTAL
      FROM _MEST.BEST_NETWERK t1
      WHERE t1.BRS_LOSSEN = 100090189;
QUIT;
```

```
GOPTIONS NOACCESSIBLE;


%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: BRS van LEV naar VERV   */

GOPTIONS ACCESSIBLE;
%_eg_conditional_dropds(WORK.LEV_VERV);

PROC SQL;
   CREATE TABLE WORK.LEV_VERV AS
   SELECT t1.BRS_LEV,
          t1.BRS_VERV,
          t1.MAAND_LADEN
      FROM WORK.VDM_SASVI t1;
QUIT;

GOPTIONS NOACCESSIBLE;


%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: Frequentie lev-verv combinaties   */

GOPTIONS ACCESSIBLE;
proc summary data=work.lev_verv nway;
  class brs_lev brs_verv maand_laden;
  output out=lev_verv_freq(drop=_type_);
run;

GOPTIONS NOACCESSIBLE;
%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;
%LET _SASPROGRAMFILE=;
%LET _SASPROGRAMFILEHOST=;


/*   START OF NODE: Hernoem kolommen   */
```

```
GOPTIONS ACCESSIBLE;
%_eg_conditional_dropds(WORK.LEV_VERV_FREQ_COR);

PROC SQL;
   CREATE TABLE WORK.LEV_VERV_FREQ_COR AS
   SELECT t1.BRS_LEV AS Source,
          t1.BRS_VERV AS Target,
          t1.MAAND_LADEN,
          t1._FREQ_ AS Frequentie
      FROM WORK.LEV_VERV_FREQ t1;
QUIT;

GOPTIONS NOACCESSIBLE;


%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: Frequentie AFN en hernoem kolommen   */

GOPTIONS ACCESSIBLE;
%_eg_conditional_dropds(WORK.AFN_FREQ_COR);

PROC SQL;
   CREATE TABLE WORK.AFN_FREQ_COR AS
   SELECT t1.BRS_AFN AS Source,
          t1.MAAND_LADEN,
          /* Frequentie */
            (COUNT(t1.BRS_AFN)) AS Frequentie
      FROM WORK.VDM_SASVI t1
      GROUP BY t1.BRS_AFN,
               t1.MAAND_LADEN;
QUIT;

GOPTIONS NOACCESSIBLE;


%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: BRS van VERV naar AFN   */

GOPTIONS ACCESSIBLE;
```

```
%_eg_conditional_dropds(WORK.VERV_AFN);

PROC SQL;
   CREATE TABLE WORK.VERV_AFN AS
   SELECT t1.BRS_VERV,
          t1.BRS_AFN,
          t1.MAAND_LADEN
      FROM WORK.VDM_SASVI t1;
QUIT;

GOPTIONS NOACCESSIBLE;


%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: Frequentie verv-afn combinaties   */

GOPTIONS ACCESSIBLE;
proc summary data=work.verv_afn nway;
class brs_verv brs_afn maand_laden;
output out=verv_afn_freq(drop=_type_);
run;

GOPTIONS NOACCESSIBLE;
%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;
%LET _SASPROGRAMFILE=;
%LET _SASPROGRAMFILEHOST=;


/*   START OF NODE: Hernoem kolommen   */

GOPTIONS ACCESSIBLE;
%_eg_conditional_dropds(WORK.VERV_AFN_FREQ_COR);

PROC SQL;
   CREATE TABLE WORK.VERV_AFN_FREQ_COR AS
   SELECT t1.BRS_VERV AS Source,
          t1.BRS_AFN AS Target,
          t1.MAAND_LADEN,
          t1._FREQ_ AS Frequentie
      FROM WORK.VERV_AFN_FREQ t1;
QUIT;
```

```
GOPTIONS NOACCESSIBLE;


%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: Append Table   */

GOPTIONS ACCESSIBLE;
%_eg_conditional_dropds(WORK.SOURCE_TARGET_FREQ);
PROC SQL;
CREATE TABLE WORK.SOURCE_TARGET_FREQ AS
SELECT * FROM WORK.LEV_VERV_FREQ_COR
 OUTER UNION CORR
SELECT * FROM WORK.VERV_AFN_FREQ_COR
 OUTER UNION CORR
SELECT * FROM WORK.AFN_FREQ_COR
;
Quit;


GOPTIONS NOACCESSIBLE;
%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: Freq berekening: missende target weggooien   */

GOPTIONS ACCESSIBLE;
%_eg_conditional_dropds(WORK.FREQ_MISSENDWEG);

PROC SQL;
   CREATE TABLE WORK.FREQ_MISSENDWEG AS
   SELECT t1.Source,
          t1.Target,
          t1.MAAND_LADEN,
          t1.Frequentie
      FROM WORK.SOURCE_TARGET_FREQ t1
      WHERE t1.Target NOT IS MISSING;
QUIT;

GOPTIONS NOACCESSIBLE;


%LET _CLIENTTASKLABEL=;
```

```
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: Frequentie berekening: # per source   */

GOPTIONS ACCESSIBLE;
%_eg_conditional_dropds(WORK.FREQ_AANTAL_SOURCE);

PROC SQL;
   CREATE TABLE WORK.FREQ_AANTAL_SOURCE AS
   SELECT t1.Source AS Node,
          /* Aantal node */
            (SUM(t1.Frequentie)) AS 'Aantal node'n
      FROM WORK.FREQ_MISSENDWEG t1
      GROUP BY t1.Source;
QUIT;

GOPTIONS NOACCESSIBLE;


%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: Frequentie berekening: # per target   */

GOPTIONS ACCESSIBLE;
%_eg_conditional_dropds(WORK.FREQ_AANTAL_TARGET);

PROC SQL;
   CREATE TABLE WORK.FREQ_AANTAL_TARGET AS
   SELECT t1.Target AS Node,
          /* Aantal node */
            (SUM(t1.Frequentie)) AS 'Aantal node'n
      FROM WORK.FREQ_MISSENDWEG t1
      GROUP BY t1.Target;
QUIT;

GOPTIONS NOACCESSIBLE;


%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;
```

```
/*   START OF NODE: Append Table (2)   */

GOPTIONS ACCESSIBLE;
%_eg_conditional_dropds(WORK.FREQ_AANTAL_SOURCE_TARGET);
PROC SQL;
CREATE TABLE WORK.FREQ_AANTAL_SOURCE_TARGET AS
SELECT * FROM WORK.FREQ_AANTAL_SOURCE
 OUTER UNION CORR
SELECT * FROM WORK.FREQ_AANTAL_TARGET
;
Quit;


GOPTIONS NOACCESSIBLE;
%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: Frequentie per node (source & target)   */

GOPTIONS ACCESSIBLE;
%_eg_conditional_dropds(WORK.FREQ_AANTAL_NODE);

PROC SQL;
   CREATE TABLE WORK.FREQ_AANTAL_NODE AS
   SELECT t1.Node,
          /* Frequentie */
            (SUM(t1.'Aantal node'n)) AS Frequentie
      FROM WORK.FREQ_AANTAL_SOURCE_TARGET t1
      GROUP BY t1.Node
      ORDER BY t1.Node;
QUIT;

GOPTIONS NOACCESSIBLE;


%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: akkoordpercentage berekenen   */

GOPTIONS ACCESSIBLE;
data nalevingsperc;
```

```
        set inspecties_sasvi;
        perc_akkoord = aantal_akkoord / aantal_inspecties * 100;
run;



GOPTIONS NOACCESSIBLE;
%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;
%LET _SASPROGRAMFILE=;
%LET _SASPROGRAMFILEHOST=;


/*   START OF NODE: Program   */

GOPTIONS ACCESSIBLE;
data naleving_cat;
        set nalevingsperc;
        if perc_akkoord = . then cat_akkoord = "missing";
        if perc_akkoord >= 0 and perc_akkoord <= 10 then cat_akkoord = "0-10";
        if perc_akkoord > 10 and perc_akkoord <= 20 then cat_akkoord = "10-20";
        if perc_akkoord > 20 and perc_akkoord <= 30 then cat_akkoord = "20-30";
        if perc_akkoord > 30 and perc_akkoord <= 40 then cat_akkoord = "30-40";
        if perc_akkoord > 40 and perc_akkoord <= 50 then cat_akkoord = "40-50";
        if perc_akkoord > 50 and perc_akkoord <= 60 then cat_akkoord = "50-60";
        if perc_akkoord > 60 and perc_akkoord <= 70 then cat_akkoord = "60-70";
        if perc_akkoord > 70 and perc_akkoord <= 80 then cat_akkoord = "70-80";
        if perc_akkoord > 80 and perc_akkoord <= 90 then cat_akkoord = "80-90";
        if perc_akkoord > 90 and perc_akkoord <= 100 then cat_akkoord =
"90-100";
run;

GOPTIONS NOACCESSIBLE;
%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;
%LET _SASPROGRAMFILE=;
%LET _SASPROGRAMFILEHOST=;


/*   START OF NODE: Voeg inspectie resultaten toe   */

GOPTIONS ACCESSIBLE;
%_eg_conditional_dropds(WORK.SOURCE_TARGET_FREQ_INSPECTIE);

PROC SQL;
   CREATE TABLE WORK.SOURCE_TARGET_FREQ_INSPECTIE AS
```

```
   SELECT t1.Source,
          t1.Target,
          t1.MAAND_LADEN,
          t1.Frequentie,
          /* Inspectie */
            (case when
            AANTAL_INSPECTIES = 0 then 'Geen inspectie'
            else
            (case when
            AANTAL_AKKOORD =0 then 'Niet akkoord'
            when AANTAL_NIET_AKKOORD=0 then 'Akkoord'
            else 'Akkoord en Niet akkoord'
            end)
            end) AS Inspectie,
          t2.AANTAL_INSPECTIES,
          t2.AANTAL_AKKOORD,
          t2.AANTAL_NIET_AKKOORD,
          t2.perc_akkoord,
          t2.cat_akkoord
      FROM WORK.SOURCE_TARGET_FREQ t1
           LEFT JOIN WORK.NALEVING_CAT t2 ON (t1.Source = t2.BRS);
QUIT;

GOPTIONS NOACCESSIBLE;



%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: Voeg type source toe   */
%LET _CLIENTTASKLABEL='Voeg type source toe';
%LET _CLIENTPROCESSFLOWNAME='Data preparatie SAS VA (Miriam)';
%LET _CLIENTPROJECTPATH='E:\Users\_KenO_Datascience\010 Mestfraude\02
Netwerkanalyse\Datapreparatie.egp';
%LET _CLIENTPROJECTPATHHOST='w1052p0033.cicapp.nl';
%LET _CLIENTPROJECTNAME='Datapreparatie.egp';

GOPTIONS ACCESSIBLE;
%_eg_conditional_dropds(WORK.VDM_INSPECTIES_SASVA);

PROC SQL;
   CREATE TABLE WORK.VDM_INSPECTIES_SASVA AS
   SELECT t1.Source,
          t1.Target,
          t1.MAAND_LADEN,
          t1.Frequentie,
          t2.Type,
```

```
          t1.Inspectie,
          t1.AANTAL_INSPECTIES,
          t1.AANTAL_AKKOORD,
          t1.AANTAL_NIET_AKKOORD,
          t1.perc_akkoord,
          t1.cat_akkoord
      FROM WORK.SOURCE_TARGET_FREQ_INSPECTIE t1
           LEFT JOIN WORK.BRS_TYPE_SASVI t2 ON (t1.Source = t2.BRS);
QUIT;

GOPTIONS NOACCESSIBLE;



%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: Koppel locatie source   */

GOPTIONS ACCESSIBLE;
%_eg_conditional_dropds(WORK.NETWERK_SOURCE_LOC);

PROC SQL;
   CREATE TABLE WORK.NETWERK_SOURCE_LOC AS
   SELECT t1.Source,
          t1.Target,
          t1.MAAND_LADEN,
          t1.Frequentie,
          t1.Type,
          t1.Inspectie,
          t1.AANTAL_INSPECTIES,
          t1.AANTAL_AKKOORD,
          t1.AANTAL_NIET_AKKOORD,
          t1.perc_akkoord,
          t1.cat_akkoord,
          t2.VA_LATITUDE_ADRES AS SOURCE_VA_LATITUDE_ADRES,
          t2.VA_LONGITUDE_ADRES AS SOURCE_VA_LONGITUDE_ADRES
      FROM WORK.VDM_INSPECTIES_SASVA t1
           LEFT JOIN WORK.BEDRIJF_COOR t2 ON (t1.Source = t2.BRS);
QUIT;

GOPTIONS NOACCESSIBLE;



%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
```

```
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: Koppel target locatie   */
%LET _CLIENTTASKLABEL='Koppel target locatie';
%LET _CLIENTPROCESSFLOWNAME='Data preparatie SAS VA (Miriam)';
%LET _CLIENTPROJECTPATH='E:\Users\_KenO_Datascience\010 Mestfraude\02
Netwerkanalyse\Datapreparatie.egp';
%LET _CLIENTPROJECTPATHHOST='w1052p0033.cicapp.nl';
%LET _CLIENTPROJECTNAME='Datapreparatie.egp';

GOPTIONS ACCESSIBLE;
%_eg_conditional_dropds(WORK.NETWERK_LOCATIES);

PROC SQL;
   CREATE TABLE WORK.NETWERK_LOCATIES AS
   SELECT t1.Source,
          t1.Target,
          t1.MAAND_LADEN,
          t1.Frequentie,
          t1.Type,
          t1.Inspectie,
          t1.perc_akkoord,
          t1.cat_akkoord,
          t1.AANTAL_INSPECTIES,
          t1.AANTAL_AKKOORD,
          t1.AANTAL_NIET_AKKOORD,
          t1.SOURCE_VA_LATITUDE_ADRES,
          t1.SOURCE_VA_LONGITUDE_ADRES,
          t2.VA_LATITUDE_ADRES AS TARGET_VA_LATITUDE_ADRES,
          t2.VA_LONGITUDE_ADRES AS TARGET_VA_LONGITUDE_ADRES
      FROM WORK.NETWERK_SOURCE_LOC t1
           LEFT JOIN WORK.BEDRIJF_COOR t2 ON (t1.Target = t2.BRS);
QUIT;

GOPTIONS NOACCESSIBLE;



%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: Koppel overige risicofactoren   */
%LET _CLIENTTASKLABEL='Koppel overige risicofactoren';
%LET _CLIENTPROCESSFLOWNAME='Data preparatie SAS VA (Miriam)';
%LET _CLIENTPROJECTPATH='E:\Users\_KenO_Datascience\010 Mestfraude\02
Netwerkanalyse\Datapreparatie.egp';
```

```
%LET _CLIENTPROJECTPATHHOST='w1052p0033.cicapp.nl';
%LET _CLIENTPROJECTNAME='Datapreparatie.egp';

GOPTIONS ACCESSIBLE;
%_eg_conditional_dropds(WORK.NETWERK_INPUT_VA);

PROC SQL;
   CREATE TABLE WORK.NETWERK_INPUT_VA AS
   SELECT t1.Source,
          t1.Target,
          t1.MAAND_LADEN,
          t1.Frequentie,
          t1.Type,
          t1.Inspectie,
          t1.AANTAL_INSPECTIES,
          t1.AANTAL_AKKOORD,
          t1.AANTAL_NIET_AKKOORD,
          t1.perc_akkoord,
          t1.cat_akkoord,
          t1.SOURCE_VA_LATITUDE_ADRES,
          t1.SOURCE_VA_LONGITUDE_ADRES,
          t1.TARGET_VA_LATITUDE_ADRES,
          t1.TARGET_VA_LONGITUDE_ADRES,
          t2.RF_FEESTDAG AS SOURCE_RF_FEESTDAG,
          t2.RF_TIJD AS SOURCE_RF_TIJD,
          t2.RF_VDMWIJZIGING AS SOURCE_RF_VDMWIJZIGING,
          t3.RF_FEESTDAG AS TARGET_RF_FEESTDAG,
          t3.RF_TIJD AS TARGET_RF_TIJD,
          t3.RF_VDMWIJZIGING AS TARGET_RF_VDMWIJZIGING
      FROM WORK.NETWERK_LOCATIES t1
           LEFT JOIN WORK.RISICOFACTOREN_SASVI t2 ON (t1.Source = t2.BRS)
           LEFT JOIN WORK.RISICOFACTOREN_SASVI t3 ON (t1.Target = t3.BRS);
QUIT;

GOPTIONS NOACCESSIBLE;




%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: Import Data
(Centraliteiten_Inspectiedata_RF_1207.xlsx[Centralities])   */
%LET _CLIENTTASKLABEL='Import Data
(Centraliteiten_Inspectiedata_RF_1207.xlsx[Centralities])';
%LET _CLIENTPROCESSFLOWNAME='Data preparatie SAS VA (Miriam)';
%LET _CLIENTPROJECTPATH='E:\Users\_KenO_Datascience\010 Mestfraude\02
```

```
Netwerkanalyse\Datapreparatie.egp';
%LET _CLIENTPROJECTPATHHOST='w1052p0033.cicapp.nl';
%LET _CLIENTPROJECTNAME='Datapreparatie.egp';

GOPTIONS ACCESSIBLE;
/* ---------------------------------------------------------------------
   Code generated by a SAS task

   Generated on vrijdag 23 August 2019 at 15:44:40
   By task:     Import Data Wizard

   Source file: R:\Mijn Documenten\Data thesis\SNA
   MEST\Centraliteiten_Inspectiedata_RF_1207.xlsx
   Server:      Local File System

   Output data: WORK.CENTRALITEITEN_INSPECTIEDAT_0000
   Server:      SASApp

   Note: In preparation for running the following code, the Import
   Data wizard has used internal routines to transfer the source data
   file from the local file system to SASApp. There is no SAS code
   available to represent this action.
   --------------------------------------------------------------------- */

/* ---------------------------------------------------------------------
   This DATA step reads the data values from a temporary text file
   created by the Import Data wizard. The values within the temporary
   text file were extracted from the Excel source file.
   --------------------------------------------------------------------- */

DATA WORK.CENTRALITEITEN_INSPECTIEDAT_0000;
    LENGTH
        'Item Type'n     $ 4
        ID                 8
        'Betweenness-centraliteit'n   8
        'Ranking Betweenness'n   8
        'Closeness-centraliteit'n   8
        'Ranking Closeness'n   8
        'Stress-centraliteit'n   8
        'Ranking Stress'n   8
        Bereikcentraliteit   8
        'Ranking Reach'n   8
        AANTAL_AKKOORD     8
        AANTAL_NIET_AKKOORD   8
        AANTAL_INSPECTIES   8
        AANTAL_INSPECTIES_ZONDER_0 $ 14
        NALEVINGSPERCENTAGE   8
        Inspectie        $ 23
        RF_FEESTDAG        8
        RF_TIJD            8
        RF_VDMWIJZIGING    8
        Type             $ 29 ;
```

```
FORMAT
    'Item Type'n     $CHAR4.
    ID               BEST12.
    'Betweenness-centraliteit'n COMMA12.4
    'Ranking Betweenness'n COMMA12.
    'Closeness-centraliteit'n COMMA12.4
    'Ranking Closeness'n COMMA12.
    'Stress-centraliteit'n COMMA12.3
    'Ranking Stress'n F12.
    Bereikcentraliteit COMMA12.4
    'Ranking Reach'n COMMA12.
    AANTAL_AKKOORD   BEST12.
    AANTAL_NIET_AKKOORD BEST12.
    AANTAL_INSPECTIES BEST12.
    AANTAL_INSPECTIES_ZONDER_0 $CHAR14.
    NALEVINGSPERCENTAGE PERCENT9.2
    Inspectie        $CHAR23.
    RF_FEESTDAG      PERCENT10.3
    RF_TIJD          PERCENT10.3
    RF_VDMWIJZIGING  PERCENT10.3
    Type             $CHAR29. ;
INFORMAT
    'Item Type'n     $CHAR4.
    ID               BEST12.
    'Betweenness-centraliteit'n COMMA12.
    'Ranking Betweenness'n COMMA12.
    'Closeness-centraliteit'n COMMA12.
    'Ranking Closeness'n COMMA12.
    'Stress-centraliteit'n COMMA12.
    'Ranking Stress'n BEST12.
    Bereikcentraliteit COMMA12.
    'Ranking Reach'n COMMA12.
    AANTAL_AKKOORD   BEST12.
    AANTAL_NIET_AKKOORD BEST12.
    AANTAL_INSPECTIES BEST12.
    AANTAL_INSPECTIES_ZONDER_0 $CHAR14.
    NALEVINGSPERCENTAGE PERCENT9.
    Inspectie        $CHAR23.
    RF_FEESTDAG      PERCENT10.
    RF_TIJD          PERCENT10.
    RF_VDMWIJZIGING  PERCENT10.
    Type             $CHAR29. ;
INFILE 'E:\saswork\_TD11368_W1052P0033_\#LN00263'
    LRECL=228
    ENCODING="WLATIN1"
    TERMSTR=CRLF
    DLM='7F'x
    MISSOVER
    DSD ;
INPUT
    'Item Type'n     : $CHAR4.
    ID               : BEST32.
```

```
        'Betweenness-centraliteit'n : BEST32.
        'Ranking Betweenness'n : BEST32.
        'Closeness-centraliteit'n : BEST32.
        'Ranking Closeness'n : BEST32.
        'Stress-centraliteit'n : BEST32.
        'Ranking Stress'n : BEST32.
        Bereikcentraliteit : BEST32.
        'Ranking Reach'n : BEST32.
        AANTAL_AKKOORD   : BEST32.
        AANTAL_NIET_AKKOORD : BEST32.
        AANTAL_INSPECTIES : BEST32.
        AANTAL_INSPECTIES_ZONDER_0 : $CHAR14.
        NALEVINGSPERCENTAGE : BEST32.
        Inspectie        : $CHAR23.
        RF_FEESTDAG      : BEST32.
        RF_TIJD          : BEST32.
        RF_VDMWIJZIGING  : BEST32.
        Type             : $CHAR29. ;
RUN;


GOPTIONS NOACCESSIBLE;
%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: Toevoegen Centraliteiten aan Node   */

GOPTIONS ACCESSIBLE;
%_eg_conditional_dropds(WORK.QUERY_FOR_CENTRALITEITEN_INSPECT);

PROC SQL;
   CREATE TABLE WORK.QUERY_FOR_CENTRALITEITEN_INSPECT AS
   SELECT t1.ID,
          t2.Frequentie,
          t1.'Betweenness-centraliteit'n,
          t1.'Closeness-centraliteit'n,
          t1.'Stress-centraliteit'n,
          t1.Bereikcentraliteit
      FROM WORK.CENTRALITEITEN_INSPECTIEDAT_0000 t1
           RIGHT JOIN WORK.FREQ_AANTAL_NODE t2 ON (t1.ID = t2.Node);
QUIT;

GOPTIONS NOACCESSIBLE;



%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
```

```
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: Unieke BRS afnemers   */

GOPTIONS ACCESSIBLE;
%_eg_conditional_dropds(WORK.VDM_AFN);

PROC SQL;
   CREATE TABLE WORK.VDM_AFN AS
   SELECT DISTINCT t1.BRS_AFN
      FROM _MEST.VDM_SASVI_28_06_2019 t1;
QUIT;

GOPTIONS NOACCESSIBLE;


%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: Afnemer BRS informatie   */
LIBNAME TMP00002 "E:\Users\_KenO_Datascience\010 Mestfraude\02
Netwerkanalyse\Output voor SAS-VI";



GOPTIONS ACCESSIBLE;
%put ERROR: Unable to get SAS code. Unable to open input data;


GOPTIONS NOACCESSIBLE;



%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: Unieke BRS vervoerders   */

GOPTIONS ACCESSIBLE;
%_eg_conditional_dropds(WORK.VDM_VERV);
```

```
PROC SQL;
   CREATE TABLE WORK.VDM_VERV AS
   SELECT DISTINCT t1.BRS_VERV
      FROM _MEST.VDM_SASVI_28_06_2019 t1;
QUIT;

GOPTIONS NOACCESSIBLE;


%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: Vervoerder BRS informatie   */
LIBNAME TMP00002 "E:\Users\_KenO_Datascience\010 Mestfraude\02
Netwerkanalyse\Output voor SAS-VI";



GOPTIONS ACCESSIBLE;
%put ERROR: Unable to get SAS code. Unable to open input data;


GOPTIONS NOACCESSIBLE;



%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: Unieke BRS leverancier   */

GOPTIONS ACCESSIBLE;
%_eg_conditional_dropds(WORK.VDM_LEV);

PROC SQL;
   CREATE TABLE WORK.VDM_LEV AS
   SELECT DISTINCT t1.BRS_LEV
      FROM _MEST.VDM_SASVI_28_06_2019 t1;
QUIT;

GOPTIONS NOACCESSIBLE;


%LET _CLIENTTASKLABEL=;
```

```
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: Leverancier BRS informatie   */
LIBNAME TMP00002 "E:\Users\_KenO_Datascience\010 Mestfraude\02
Netwerkanalyse\Output voor SAS-VI";



GOPTIONS ACCESSIBLE;
%put ERROR: Unable to get SAS code. Unable to open input data;


GOPTIONS NOACCESSIBLE;



%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;


/*   START OF NODE: BRS met type   */

GOPTIONS ACCESSIBLE;
%_eg_conditional_dropds(WORK.BRS_TYPE_SASVI);

PROC SQL;
   CREATE TABLE WORK.BRS_TYPE_SASVI AS
   SELECT t1.BRS,
          /* Type */
            (catx("/",catx("/",t2.Leverancier,t3.Vervoerder),t4.Afnemer)) AS
Type
      FROM WORK.VDM_SASVI_UNIBRS t1
           LEFT JOIN WORK.VDM_SASVI_LEV t2 ON (t1.BRS = t2.BRS)
           LEFT JOIN WORK.VDM_SASVI_VERV t3 ON (t1.BRS = t3.BRS)
           LEFT JOIN WORK.VDM_SASVI_AFN t4 ON (t1.BRS = t4.BRS)
      ORDER BY t2.Leverancier DESC,
               t3.Vervoerder DESC,
               t4.Afnemer DESC;
QUIT;

GOPTIONS NOACCESSIBLE;
```

```
%LET _CLIENTTASKLABEL=;
%LET _CLIENTPROCESSFLOWNAME=;
%LET _CLIENTPROJECTPATH=;
%LET _CLIENTPROJECTPATHHOST=;
%LET _CLIENTPROJECTNAME=;

;*';*";*/;quit;run;
ODS _ALL_ CLOSE;
```