%let start\_date=201801;

%let end\_date=201802;

%let start=%sysfunc(inputn(&start\_date.,yymmn6.),date9.);

%let end=%sysfunc(inputn(&end\_date.,yymmn6.),date9.);

/\*%Let start=%Sysfunc(Intnx(Month, "(%sysfunc(inputn(&start\_date.,yymmn6.),date9.)"d, 0, e), date9.);

%Let end=%Sysfunc(Intnx(Month, "(%sysfunc(inputn(&end\_date.,yymmn6.),date9.)"d, 0, e), date9.);

%put &start &end;\*/

option nomlogic nomprint nosource nonotes nosymbolgen; /\*off with log notes\*/

option mlogic mprint source notes symbolgen;

**%macro** ***dates***;

%let intervall=month;/\* loop days(day) month \*/

/\*converts the dates to SAS dates\*/

%let start=%sysfunc(inputn(&start,anydtdte9.));

%let end=%sysfunc(inputn(&end,anydtdte9.));

%put &start &end;

/\*determines the number of months between the two dates\*/

%let dif=%sysfunc(intck(&intervall.,&start,&end));

%put &dif;

%do j=**0** %to &dif;

/\*advances the date i months from the start date and applys the DATE9. format\*/

%let date=%sysfunc(putn(%sysfunc(intnx(&intervall.,&start,&j,b)),date9.));

/\*

n no separator

b blank

d dash

s slash

p period

c colon

date9. tex 15DEC2017

DATE. 15DEC17

DATE.5 15DEC

DATE.11 15-dec-2017

yymmddn8. 20171215

yymmn6. 201712

yymmdd10. 2017-12-15

year4. 2017

Month2. 12

yymmdd4. 1711

yymmdd6. 171101

week.1 lördag

week.2 söndag

week.3 måndag

week.4 tisdag

week.5 onsdag

week.6 torsdag

week.7 fredag

yyqn3. kvartal

WEEKv3. Veckonummer

datetime21.2\*\*/\*/

/\* change format\*/

%let new\_format=%sysfunc(inputn(&date, date9.), yymmddn8.);

%let yyy= %sysfunc(datetime(),datetime21.2);

%let LastMnth=%sysfunc(intnx(month,"&date"d,-1),date9.);

%let datex=%sysfunc(intnx(month, %sysfunc(today()), -1), yymmddn8.);

%let Yesterday=%sysfunc(intnx(day,%sysfunc(today()),-1),date9.);

%Let Rpt\_End\_Dt=%Sysfunc(Intnx(Month, "&date"d, -1, e), date9.);

%let x\_month\_ago=%sysfunc(intnx(month,%sysfunc(today()),-1,same),yymmddn8.);

%let LastMnthx=%sysfunc(intnx(month,"&date"d,-1,same),yymmddn8.);

%let year=%sysfunc(inputn(&date, date9.), year4.);

%let month=%sysfunc(inputn(&date, date9.), Month2.);

%let Prog\_start = %sysfunc(TIME(),time.);

%Let First\_day\_month=%Sysfunc(Intnx(Month, "&date"d, 0, b), date9.);

%Let Second\_day\_month=%Sysfunc(Intnx(day, "%Sysfunc(Intnx(Month, "&date"d, 0, b), date9.)"d, 1, b), date9.);

%Let First\_day\_nextMonth=%Sysfunc(Intnx(Month, "&date"d, 1, b), date9.);

%Let Second\_day\_nextmonth=%Sysfunc(Intnx(day, "%Sysfunc(Intnx(Month, "&date"d, 1, b), date9.)"d, 1, b), date9.);

%let Nr\_monday\_month=%sysfunc(intnx(week.3,%sysfunc(mdy(&month,1,&year)),0,e),date9.);

%let First\_tuesday\_month=%sysfunc(intnx(week.4,%sysfunc(mdy(&month,1,&year)),0,e),date9.);

%let Second\_tuesday\_month=%sysfunc(intnx(week.4,%sysfunc(mdy(&month,1,&year)),1,e),date9.);

%let First\_saturday\_month=%sysfunc(intnx(week.1,%sysfunc(mdy(&month,1,&year)),0,e),date9.);

/\*monday after the first saturday\*/

%Let First\_monday\_afterFist\_saturday=%Sysfunc(Intnx(day, "%sysfunc(intnx(week.1,%sysfunc(mdy(&month,1,&year)),0,e),date9.)"d, 2, e), date9.);

%let Lastday=%sysfunc(intnx(day,"&date"d,-1),date9.);

%let Duration\_days = %sysfunc(intck(day,"&LastMnth"d,"&date"d));

%let Last\_lördag = %sysfunc(putn(%eval(%sysfunc(intnx(week, "&date"D, 0))-1),DATE9.));

%let ReportDate = %sysfunc(intnx(week.6, "&date"d, 0), DATE9.);

%let Birthday=%sysfunc(inputn(%SYSEVALF(%sysfunc(intck(day,"1jun1979"d,"&sysdate."d))/365.25),best32.),16.2);

%let Week=%substr(%sysfunc(inputn(&date, date9.), WEEKv3.),2);

%let Previous\_week=%substr(%sysfunc(inputn(%sysfunc(intnx(day,"&date"d,-7,same),date9.), date9.), WEEKv3.),2);

%let Kvartal=%sysfunc(putn(%substr(%sysfunc(inputn(&date, date9.), yyqn3.),3), z2));

%let Next\_qtr=%sysfunc(putn(%eval(%substr(%sysfunc(inputn(&date, date9.), yyqn3.),3)+1), z2));

/\*%let Duration\_hours = %sysfunc(intck(minute,"&Prog\_start."t,"&Prog\_end."t));

Leding zeros

%let month=%sysfunc(putn(&period, z2));

\*/

/\* fiscal year tex börjar 3 månader innan\*/

%let fiscal\_year=%sysfunc(intnx(month,"&date"d,-3),date9.);

%let Qtr=%sysfunc(inputn(&date, date9.), year4.)\_qtr\_%sysfunc(putn(%substr(%sysfunc(inputn(&date, date9.), yyqn3.),3), z2));

%put &new\_format &Birthday;

%let today\_time=%sysfunc(inputn(&sysdate, date9.), yymmdd10.) %sysfunc(time(),time8.0);

%put &&today\_time.;

%end;

**%mend**;

%***dates***;

/\* data clean\*/

**Data** V1;

set sashelp.class;

Date=**'01sep2016'd**;

format Date yymmddd10.;

V\_=cats('"', age, '"');

Find\_replace=tranwrd(age, **14**, **40**);

Remove\_space=strip(Find\_replace);

kformj\_textc = compbl( kformj\_text); /\*remove leading blanks\*/

compress\_var = compress(Find\_replace);

trim\_var = trim(Find\_replace);

Catx\_space=catx(' ',age,height);

Catx\_comma=catx(', ',age,height);

nospace=cat(age,sex);

char\_num = input(strip(trim\_var),best.);/\*\*\* char to numeric numbers \*\*\*/

char\_var = left(put(char\_num,**6.**)); /\* numeric to strig\*/

Date\_char=PUT(Date, yymmddd10.); /\* date to strig\*/

newvar = input(Date\_char,yymmdd10.);/\* string to date\*/

format newvar yymmddd10.;

Age1=put(age,z2.);/\*leading zeros\*/

**run**;

**data** one;

length before after1a after1b after2a after2b $**25**;

before = 'Hello My Name is Boo';

after1a = compress(before);

after1b = 'HelloMyNameisBoo';

after2a = compress(before,' lo');

after2b = 'HeMyNameisB';

**run**;

**data** date;

format date ddmmyy10.;

do date=**'28dec2015'd** to **'3jan2021'd** by **1**;

output;

end;

**run**;

**data** getweek;

set date;

/\*between dates\*/

Cutoff = intck("month",period,Max\_period);

x=intck('week',intnx('year',date,**0**),date)+**1**;

week=week(date,'v');

day=intnx('day', date , -**7**);

format day date9.;

drop x;

x=intck('week',intnx('year',day,**0**),day)+**1**;

Previous\_week=week(intnx('day', date , -**7**),'v');drop x day;

Sameday\_month=intnx('month1',date,-**1**,'SAMEDAY'); /\*byt -1 och manupulera, eller month7 +7 månader\*/

format Sameday\_month date9.;

Birthday=yrdif(**"1jun1979"d**,**"&sysdate"d**,'actual');

Birthday=int(yrdif(**"1jun1979"d**,**"&sysdate"d**,'actual'));

days=datdif(**"10jan2018"d**,**"&sysdate"d**,'actual');

x=intck('week',intnx('year',Sameday\_month,**0**),Sameday\_month)+**1**;

Previous\_week\_sameday=week(Sameday\_month,'v');

drop x ;

year=intnx('year', date, **1**); /\* +1 year \*/

format year year4.;

x=date;

Kvartal=qtr(x);

day\_intervall = intnx('day', date, -**1**);/\* day intervall\*/

format day\_intervall date9.;

**run**;

idx=**1**;

ID+idx;

\_n\_

**proc** **freq** data=Ucrf\_201712 ;

table period\*mm /norow /\*nofreq\*/ nocol NOPERCENT;

**run**;

**data** test;

set sashelp.class end=eof; sista raden

if eof then output;

**run**;

/\*

http://support.sas.com/documentation/cdl/en/odsug/69832/HTML/default/viewer.htm#p0tdg7fecdruacn1ajo78oqs02vc.htm \*/

**Data** V1;

set Sashelp.Class;

label

sex="test"

age="vvv";

**run**;

%let now = %sysfunc(today(),yymmddd10.);

first.variable (dubblett)

options nofullstimer;

retain now "&sysdate9"d;

/\* cross tab\*/

%let Table=v1; /\* Your dataset\*/;

%LET CT=FTHG; /\* your var to count\*/

%LET OLD\_VAR=DIV; /\* ORIGINAL VAR\*/

%LET NEW\_VAR=DIVX; /\* new var\*/

\*\*\*;

**proc** **summary** data=&table.;

var &CT.;

class &OLD\_VAR. &NEW\_VAR.;

output out=temp sum=;

**run**;

**proc** **sort** data=temp(where=(\_type\_=**3**));

by &NEW\_VAR.;

**run**;

**proc** **transpose** data=temp out=want(drop=\_name\_);

by &NEW\_VAR.;

var &CT.;

id &OLD\_VAR.;

idlabel &OLD\_VAR.;

**run**;

**Proc** **sql** noprint;

select distinct &OLD\_VAR. into : div separated by ' ' from &table.;

**quit**;

**proc** **transpose** data=want out= utfilx

(rename=(col1=Amount \_Name\_=Name));

by &NEW\_VAR.;

var &div.;

**run**;

/\* check the missing\*/

**data** Missing;

set utfilx ;

where amount = **.**;

drop Amount \_label\_;

**run**;

**data** utfil;

set utfilx ;

where amount ne **.**;

drop Amount \_label\_;

**run**;

**proc** **print** data=utfil;

**run**;

\*\*\*end cross tab;s

libname Orion 'C:\Workshop\Data\_PRG2' access=readonly;

/\* summa\*/

**proc****summary** data=Orion.Au\_salesforce nway missing ;

class gender job\_title;

var salary;

output out=CT (drop=\_type\_ \_freq\_) sum=Summa\_salary mean=AVG\_Salary;

**run**;

lo thru in spss

**data** test;

set var\_ue1;

if test<=**5** then newvar=**1**;

else

if test<=**10** then newvar=**2**;

else newvar=**3**;

**run**;

SUBSTRING DATE

**Proc** **sql**;

CREATE TABLE V1 AS

select SUBSTR(PUT(EXTR\_DATUM, YYMMDD10.),**1**,**7**) AS DATUM, EXTR\_DATUM from B\_ue.Train\_ue;

**quit**;

**Data** V1;

set b\_30.Est\_modell\_train\_ue\_30\_ar indsname=source;

datasets=substr(source,**3**,**3**);

lab\_type = translate(scan(source,**2**,'.'),' ','\_'); /\*splitta\*/

Type=strip(scan(lab\_type,**3**,' ')); /\*välj \*/

**run**;

**data** names;

set orion.customers\_ex5;

if upcase(gender)='M' then New\_G='Mr.';

else if

upcase(gender)='F' then New\_G='Ms.';

else if 'Error';

First\_name=strip(scan(name,**2**,','));

last\_name=strip(scan(name,**1**,','));

New\_Name=propcase(strip(catx(' ',New\_G,First\_name,last\_name)));

**run**;

/\* unpivot och Array\*/

**data** test2;

set orion.employee\_donations;

array v1(\*) qtr1-qtr4;

do i=**1** to **4**;

if v1(i)="." then v1(i)=**0**;

end;

**run**;

**proc** **transpose** data=test2 out= small2

(rename=(col1=Amount \_Name\_=Name));

by Employee\_Id;

var qtr1-qtr4;

**run**;

/\* unpivot \*/

**proc** **transpose** data=orion.employee\_donations out= small2

(rename=(col1=Amount \_Name\_=Name));

by Employee\_Id;

var qtr1-qtr4;

**run**;

**data** test2;

set orion.employee\_donations;

if \_numeric\_="." then \_numeric\_=**0**;

**run**;

**proc** **transpose** data=orion.employee\_donations out= small2

(rename=(col1=Amount \_Name\_=Name));

by Employee\_Id;

var qtr1-qtr4;

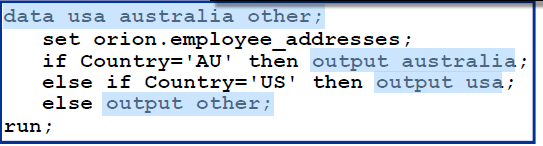
**run**;

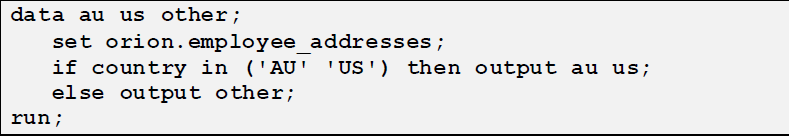
**data** test;

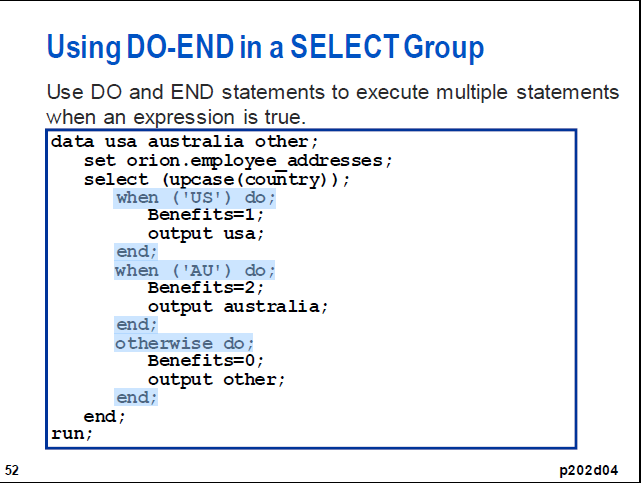
set small2;

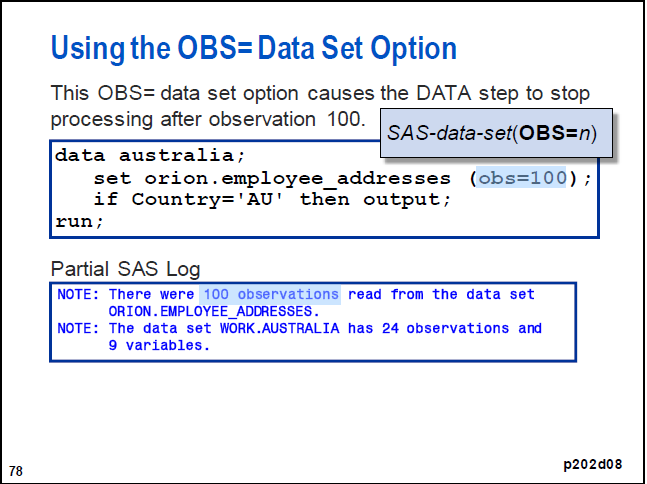
if amount="." then Amount=**0**;

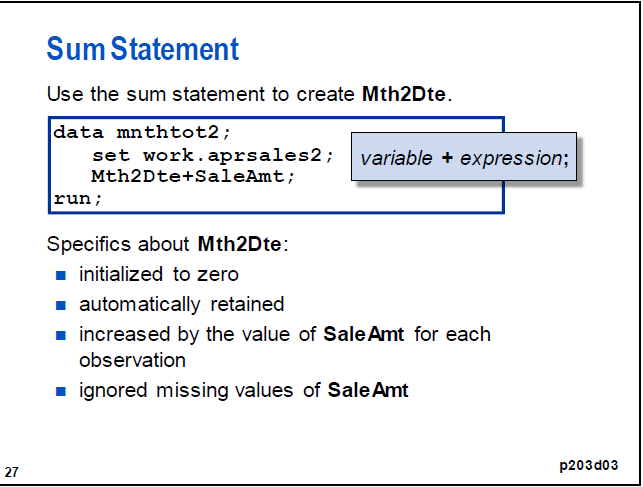
**run**

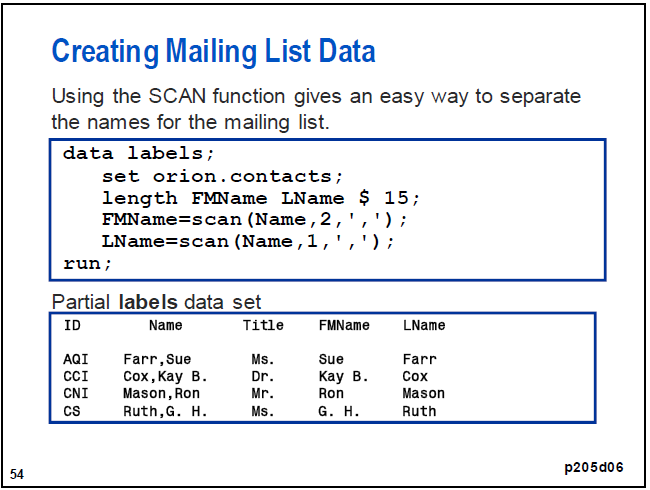


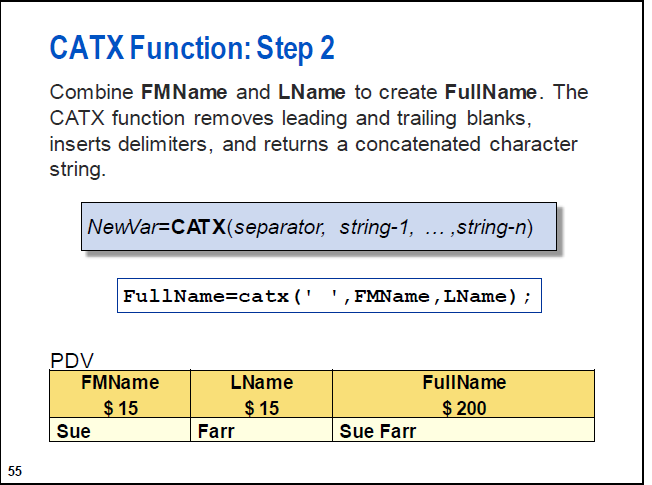


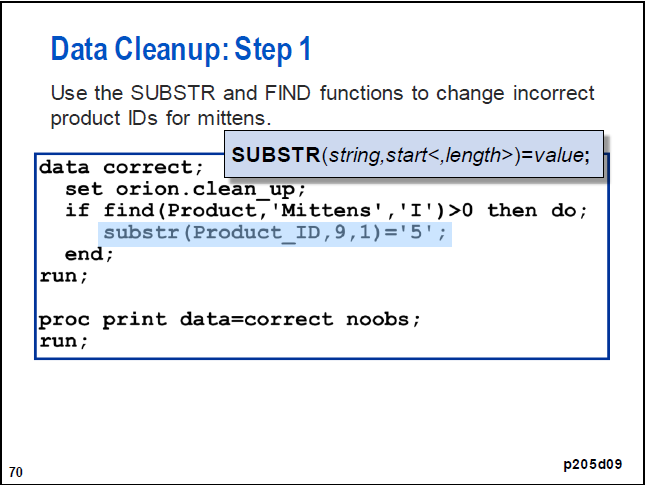












**Example Using a Simple Array in SAS**

Create an ARRAY and use a DO LOOP to SUM the data.

DATA TIMES;

INPUT TIME1 TIME2 TIME3 TIME4;

ARRAY T[4] TIME1-TIME4;

SUMTIME=0;

DO I=1 TO 4;

SUMTIME=SUMTIME+T(I);

A\_SUM=SUM(OF T(\*));

A\_MEAN=MEAN(OF T(\*));

A\_MIN=MIN(OF Tss(\*));

END;

DATALINES;

22.3 25.3 28.2 30.6

22.8 27.5 33.3 35.8

18.5 26.0 29.0 27.9

22.5 29.3 32.6 33.7

;

RUN; PROC PRINT DATA=TIMES;RUN;

Differences with array

data two;

set pubh.barley;

array base[8]

base\_chol Base\_TG Base\_HDL Base\_LDL

Base\_CRP Base\_HCYS Base\_Ins Base\_Gluc;

array mid[8]

mid\_chol mid\_TG mid\_HDL mid\_LDL

mid\_CRP mid\_HCYS mid\_Ins mid\_Gluc;

array diff[8]

dm\_chol dm\_TG dm\_HDL dm\_LDL

dm\_CRP dm\_HCYS dm\_Ins dm\_Gluc;

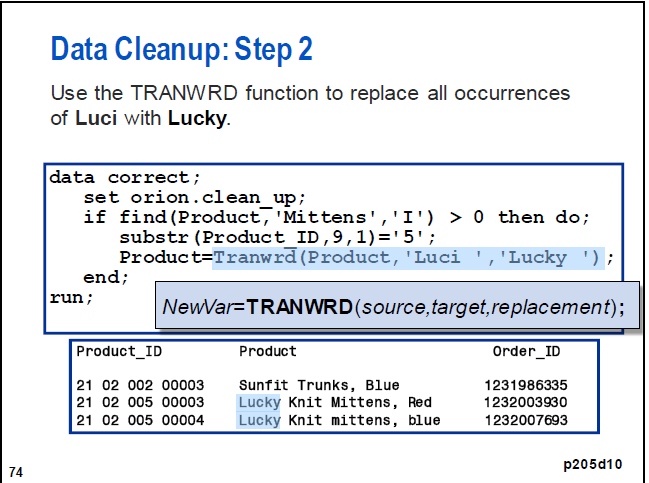
DO i=1 to 8

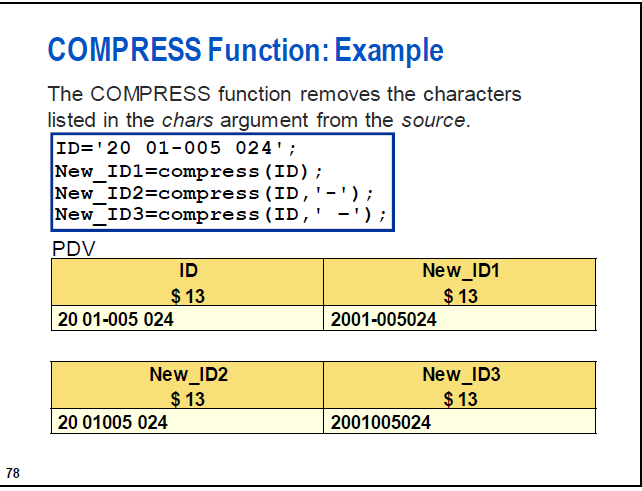
;

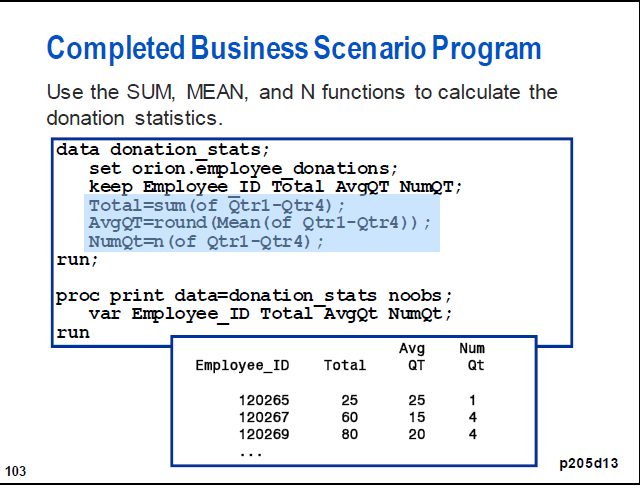
DO-loop does all the subtractions

diff[i] = mid[i] -base[i];

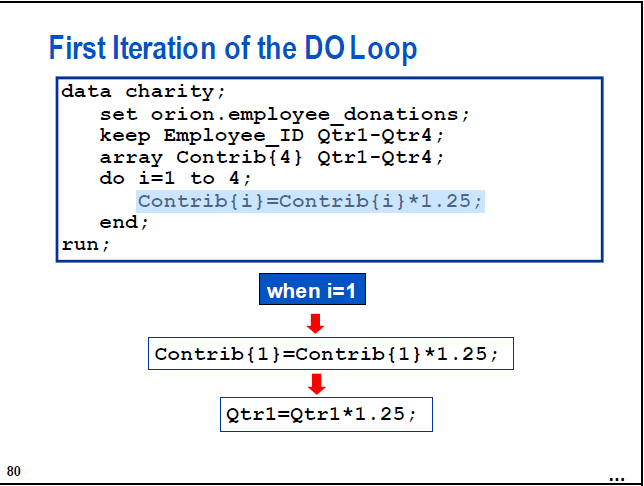
end;

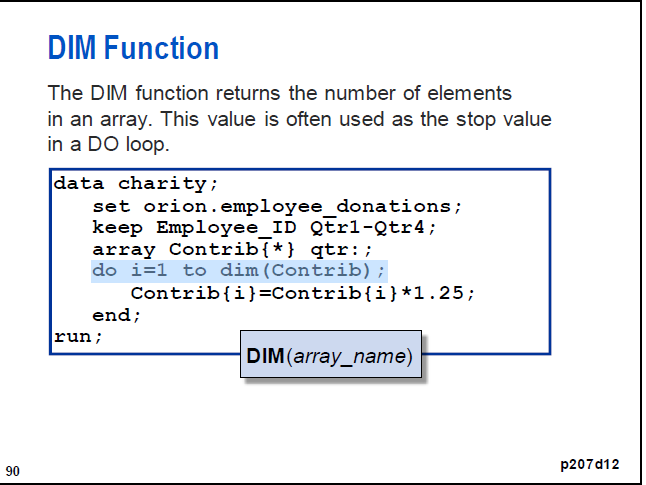


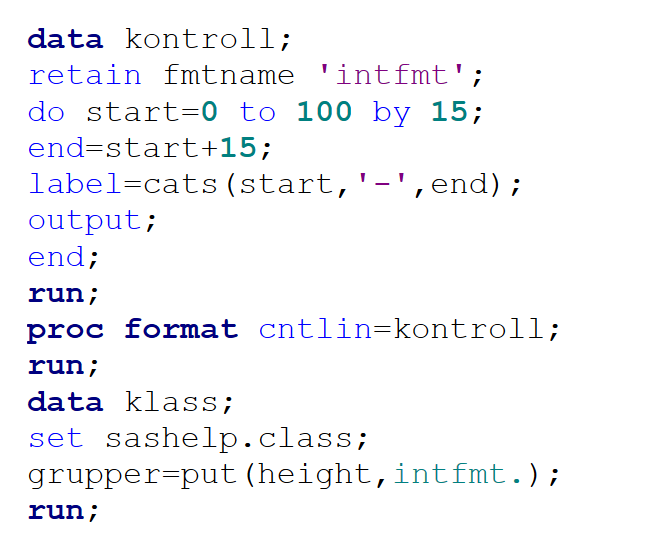












Efficienty sas codes

|  |  |
| --- | --- |
| If age <= 10  then group = "child ";  If age => 11 and age <= 19  then group = "teenager ";  If age => 20 and age <= 29  then group = "young adult";  If age => 30 and age <= 45  then group = "adult ";  If age => 46 and age <= 59  then group = "middle age ";  If age => 60  then group = "senior "; | If age <= 10  then group = "child "; else if age <= 19  then group = "teenager "; else if age <= 29  then group = "young adult"; else if age <= 45  then group = "adult "; else if age <= 59  then group = "middle age "; else group = "senior "; |
|  |  |
|  |  |

Append combine union;

%let fragor\_start="1jun2014"d;

%let fragor\_end="31may2016"d;

%let year\_start = %sysfunc(year(&fragor\_start));

%let year\_end = %sysfunc(year(&fragor\_end));

%let obs=1000;

\* --------------------- Hämta frågor ----------------- ;

**%macro** ***fragor\_iterate\_over\_years\_typ***;

%do year=&year\_start. %to &year\_end.;

\* Hämta frågor;

data fragor\_&year.;

set dwfys.fragor\_&year.

(keep=extr\_datum personnr\_orgnr abonnentnr aktsefra slutmall uc\_typ obs=&obs);

where extr\_datum between &fragor\_start. and &fragor\_end. and

aktsefra in ("0", "1", "4");

rename personnr\_orgnr=personnr;

froge\_id = \_n\_;

run;

%end;

**%mend**;

%***fragor\_iterate\_over\_years\_typ***;

**%macro** ***combine***;

data fragor\_tot;

set

%do i = &year\_start. %to &year\_end.;

fragor\_&i.

%end;

;

run;

**%mend**;

%***combine***;

# Append combine union2

%let path\_sas = &svnpgm\5-UCInternt\1-MU\6-UCBlanco\1-UCBLANCOv1\1-Utveckling\1-Modellutv\0-Utredning;

%include "&path\_sas\MACRO trimma namn.sas";

options msglevel=i; /\*writes informative messages to sas-log about index processing\*/

%let fragor\_start="1jun2014"d;

%let fragor\_end="31may2016"d;

%let year\_start = %sysfunc(year(&fragor\_start));

%let year\_end = %sysfunc(year(&fragor\_end));

%put Will use years from &year\_start to &year\_end;

\* --------------------- Hämta frågor ----------------- ;

%let obs=1000;

**%macro** ***fragor\_iterate\_over\_years\_typ***;

**%macro** ***\_***; **%mend** \_;

**data** fragor\_tot; delete; **run**;

%do year=&year\_start. %to &year\_end.;

\* Hämta frågor;

**data** fragor\_&year.;

set dwfys.fragor\_&year.(keep=extr\_datum personnr\_orgnr abonnentnr aktsefra slutmall uc\_typ obs=&obs);

where extr\_datum between &fragor\_start. and &fragor\_end. and aktsefra in ("0", "1", "4");

rename personnr\_orgnr=personnr;

froge\_id = \_n\_;

**run**;

\* Mappa på abonnentnamn;

**proc** **sql**; create table fragor2\_&year. as

select a.\*, b.abonnentnamn

from fragor\_&year. a

left join dwfys.abonnenter\_&year.(idxname=pk\_abonnenter\_&year.) b

on a.abonnentnr = b.abonnentnr and a.extr\_datum=b.extr\_datum

;**quit**;

**data** fragor\_tot; set fragor\_tot fragor2\_&year(in=b);

if b then do;

year=&year.;

%***trimma\_namn***;

end;

**run**;

%end;

**%mend**;

%***fragor\_iterate\_over\_years\_typ***;

Append min version

%let Begin = 160101;

%let Stop = 170601;

%let obs=10;

/\*Ref\_dev\_170701 Resktr.Ref\_dev\_170701\*/

\* --------------------- Hämta frågor ----------------- ;

**%macro** ***fragor\_iterate\_over\_years\_typ***;

%do year=**16** %to **17**;

%\*do Period=&begin. %to &stop. %by 100;

%do Period=**01** %to **12**;

%let month=%sysfunc(putn(&period, z2));

%let p=&year&month.01;

/\*Hämta frågor;

data test&period.;

set Resktr.Ref\_dev\_&period.(keep= personnr civ alder obs=&obs);

froge\_id = \_n\_;

Month=&period.;

run;\*/

Proc sql;

create table test&p as

select "&year&month" as PERIOD, civ,alder,riskprof\_ue, count(\*) as CT from Resktr.Ref\_dev\_&p(obs=&obs)

group by "&year&month", civ, alder,riskprof\_ue;

quit;

%end;

%end;

**%mend**;

%***fragor\_iterate\_over\_years\_typ***;

**%macro** ***combine***;

data test\_tot;

set test1: ; /\*

%do i = &begin. %to &stop. %by 100;

test&i.

%end;

;\*/

run;

**%mend**;

%***combine***;

## Keep missing var

%let seg=20;

%let sysmis=. A B C D E;

**proc** **sql** noprint; select distinct name into :positive\_ar\_vars separated by ' ' from B\_&seg.**.P**re\_deskriptiv\_&seg

where ar > **0**

; **quit**;

**Data** V1;

set b\_&seg.**.m**odellutv\_&seg. (keep=&positive\_ar\_vars.);

array Var\_miss(\*) &positive\_ar\_vars.;

do i=**1** to dim(miss);

if Var\_miss(i)=&sysmis. then do;

Var\_miss=**1**;

i=\*;

end;

end;

**run**;

: betyder börjar med

**data** kund13; set b\_13.train\_kund\_grouped b\_13.test\_kund\_grouped;

keep utfall: rls: namn x: riskprof\_me;

**run**;

skapar Index

**proc** **datasets** library=work noprint;

modify kund13;

index create namn;

**run**;

%let name\_list2table = B C D E F G H I J K L M N O P Q R S T U V W X W Z A;

**%macro** list2table(list=, table=, varname=);

**%macro** ***\_***; **%mend** \_;

\* Find max length;

%let max\_length=0;

%do i=**1** %to %sysfunc(countw(&list));

%let grej=%scan(&list, &i);

%if %length(&grej) > &max\_length %then %let max\_length=%length(&grej);

%end;

%put length &max\_length;

**data** &table;

length &varname $&max\_length;

%do i=**1** %to %sysfunc(countw(&list));

%let grej=%scan(&list, &i);

&varname = "&grej";

output;

%end;

**run**;

**%mend**;

%***list2table***(list=&name\_list2table., table=mytable, varname=Myvariable);

**proc** **sql** noprint;

select distinct Myvariable into :vars separated by " " from mytable;

**quit**;

%put &vars.;

/\*

if alder <= 20 then xalder=20 ;

if 20 < alder <= 21 then xalder=21 ;

if 21 < alder <= 22 then xalder=22 ;

if 22 < alder <= 23 then xalder=23 ;

if 23 < alder <= 24 then xalder=24 ;

if 24 < alder <= 25 then xalder=25 ;

if 25 < alder <= 40 then xalder=40 ;

if 40 < alder then xalder = 41 ;\*/

if civ in (1,4) then xciv=1;

else xciv=2;

s

**data** B\_&seg.**.m**odellutv\_&seg. B\_&seg.**.m**odellutv\_&test.\_&seg.;

keep rls\_&model. &All\_var personnr extr\_datum Dset;

set B\_&model.**.**&train.\_&model. B\_&model.**.**&test.\_&model.

indsname=source;

where riskprof\_&model. not in (**851**,**852**);

Dset=source;

array miss\_var &All\_var.;

do over miss\_var;

if miss\_var=**.** then miss\_var=-**9**;

if miss\_var=**.A** then miss\_var=-**99**;

if miss\_var=**.B** then miss\_var=-**999**;

if miss\_var=**.C** then miss\_var=-**9999**;

if miss\_var=**.D** then miss\_var=-**99999**;

if miss\_var=**.E** then miss\_var=-**999999**;

end;

rename extr\_datum=Sample;

frag7\_12=frag12-frag6; /\*Skapa ny variabel för att undvika att frågor räknas flera gånger för samma person. \*/

frag4\_6=frag6-frag3; /\*Skapa ny variabel för att undvika att frågor räknas flera gånger för samma person. \*/

frag2\_3=frag3-frag1; /\*Skapa ny variabel för att undvika att frågor räknas flera gånger för samma person. \*/

if Dset=upcase("B\_&model..&train.\_&model.") then output B\_&seg.**.m**odellutv\_&seg.;

else output B\_&seg.**.m**odellutv\_&test.\_&seg.;

**run**;

%let name\_list=Train10\_ue Test10\_ue Test10\_oot\_ue Train8\_lfor\_ue;

%let lib=B\_ue;

%let keeps=personnr extr\_datum ;

options symbolgen;

options mcompilenote=all;

options nosource nonotes; /\* suppress much of the log output You will still get warnings and error messages\*/

**%macro** loop\_datasets();

%local i next\_name;

%let i=1;

%do %while (%scan(&name\_list., &i.) ne );

%let next\_name = %scan(&name\_list., &i.);

data &next\_name. ;

set &lib.**.**&next\_name. (keep=&keeps. obs=**3** )

indsname=source;

Dset=source;

run;

%let i = %eval(&i. + 1);

%end;

**%mend**;

%***loop\_datasets***;

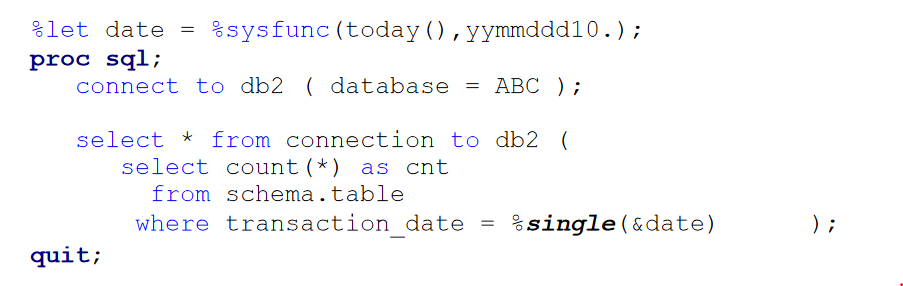
**proc** **datasets** library=WORK KILL NOLIST;

**run**;

**proc** **sql** noprint; select distinct quote(\_name\_) into :vars\_hogsta\_ar separated by ', ' from Hogsta\_AR;

**quit**;

<http://sas.cswenson.com/downloads/macros>



/\* fixar data med “” och ()

/\*example %let=a b\*/

%let data =london paris frag2\_3 v b gggg khk hgg jjj;

%let qlist=("%sysfunc(tranwrd(%sysfunc(compbl(&data)),%str( ),%bquote(",")))");

%let qlist\_comma=%qsysfunc(tranwrd(%sysfunc(compbl(&data)),%str( ),%bquote(,)));

%let qlist\_without\_parentheses="%sysfunc(tranwrd(%sysfunc(compbl(&data)),%str( ),%bquote(",")))";

%let qlist\_with\_parentheses=("%sysfunc(tranwrd(%sysfunc(compbl(&data)),%str( ),%bquote(",")))");

%put &qlist\_with\_parentheses ;

**proc** **sort** data=v1;

by variable ClassVal0 dset ;

**run**;

**PROC** **EXPAND** data=v4 OUT=v5 METHOD=NONE;

convert goal = lastgame / transformout=(lag **1**);

convert goal = lastgame2 / transformout=(lag **2**);

convert goal = lastgame3 / transformout=(lag **3**);

convert goal = lastgame4 / transformout=(lag **4**);

convert goal = lastgame5 / transformout=(lag **5**);

convert goal = lastgame6 / transformout=(lag **6**);

convert goal = lastgame7 / transformout=(lag **7**);

convert goal = lastgame8 / transformout=(lag **8**);

convert poang = lastpoang / transformout=(lag **1**);

by team season ;

**RUN**;

**proc** **sort** data=yyy nodupkey out=New\_saldo;

by kredittyp Engagemangs\_Datum;

**run**;

**data** t1\_New\_saldo\_1;

set New\_saldo;

by kredittyp Engagemangs\_Datum;

retain startsaldo\_ktyp ;

if first.kredittyp then startsaldo\_ktyp=Saldo\_ktyp;

**run**;

/\*Läs in excelfilen\*/

libname minxls 'c:\temp\Moving\_Average\_last\_6\_Month.xlsx';

**data** test;

  set minxls.'sheet1$'n;

**run**;

libname minxls clear;

/\*Beräkna medelvärden\*/

**data** MovingAvg;

  set test;

  by name;

  array senaste[**6**] \_temporary\_;

  if first.name then

    call missing(of senaste[\*]);

  senaste[mod(\_n\_,**6**)+**1**]=sales;

  MovingAvg=mean(of senaste[\*]);

**run**;

**Proc** **sql**;

create table v3 as

select v1.\*,v2.namn from v1 left join v2 on v1.orgnr=input(v2.orgnr,**10.**);

**quit**;

y=TRANWRD(v1,".",","); /\* search and replace\*/

%let dset=Sashelp.Class;

%let vars=height weight;

**data** Want;

set &dset.(keep=&vars.);

array old\_var &vars.;

array new\_var $**10** v1 v2 ; /\* döp dina nya variabler\*/

do over old\_var;

new\_var=strip(put(old\_var,numx8.5));

end;

**run**;

**Data** V1;

set Risk3.Riskab\_201711( where =(smark\_ny="P" and riskprof=**101**));

**run**;

%let start="1oct2015"d;

%let stop="1nov2016"d;

%let y=%sysfunc(year(&start));

%let m=%sysfunc(month(&start));

%let Z\_m=%sysfunc(putn(&m, z2));

%let new\_date = %sysfunc(intnx(month, &start, 3, begin));

%let y3=%sysfunc(year(&new\_date));

%let m3=%sysfunc(month(&new\_date));

%let Z\_m3=%sysfunc(putn(&m3, z2));

/\*max månad\*/

%let y\_stop=%sysfunc(year(&stop));

%let m\_stop=%sysfunc(month(&stop));

%let Z\_m\_stop=%sysfunc(putn(&m\_stop, z2));

%let min\_ar=&y&z\_m;

%let yymm3=&y3&z\_m3;

%let max\_ar=&y\_stop&z\_m\_stop;

%do ar = &min\_ar. %to &max\_ar.;

%***KONTROLLSIFFRA*** (work.v1,orgnr,o);

**%macro** loopsubs();

proc sql noprint;

%do seg=**101** %to **108**;

%global sub\_&seg;

select distinct sub&seg. into :sub\_&seg. separated by " " from v1;

%end;

quit;

**%mend**;

%***loopsubs***();

**%macro** ***yyy***;

%global yyy;

%let yyy=yes sir;

%put &yyy;

**%mend**;

%***yyy***;

%put &yyy;

**Proc** **sql**;

create table yyy as

select a.\*,mean(a.height, a.weight) as mean

, max(calculated mean) from Sashelp.Class a;

**quit**;

**Proc** **sql**;

create table Kredbar as

select Engagemangs\_Datum,

SUM(CASE WHEN typ="Nischbank" THEN sum\_saldo\_7\_sum END)/**1000000** AS saldo\_Nischbank format **8.0** ,

SUM(CASE WHEN typ="Storbank" THEN sum\_saldo\_7\_sum END)/**1000000** AS saldo\_Storbank format **8.0** ,

SUM(CASE WHEN typ="Finansbolag" THEN sum\_saldo\_7\_sum END)/**1000000** AS saldo\_Finansbolag format **8.0** ,

sum(sum\_saldo\_7\_sum)/**1000000** as Saldo format **8.0**,

sum(calculated saldo\_Nischbank, calculated saldo\_Storbank,calculated saldo\_Finansbolag) as GrandTotal format **8.0**

from test

group by Engagemangs\_Datum;

**quit**;

if omdat<intnx('month',&refdatum,-**24**) then I\_OVERFORT=**0**;

/\*\*\*\*\*loopar proc sql noprint\*/

%let min\_seg=101;

%let max\_seg=102;

**%macro** loopsubs();

proc sql noprint;

select distinct

%do seg=&min\_seg. %to &max\_seg.;

%global sub\_&seg.;

sub&seg. /\*Generera variabellista\*/

%if &seg.^=&max\_seg. %then %do;,%end; /\*Kommatecken mellan namnen\*/

%end;

into

%do seg=&min\_seg. %to &max\_seg.;

:sub\_&seg. separated by " " /\*Generera innehållet i into-satsen\*/

%if &seg.^=&max\_seg. %then %do;,%end; /\*Kommatecken\*/

%end;

from v1;

quit;

**%mend**;

%***loopsubs***();

%put &sub\_101;

**/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*denna laser in 8 ggr, medans ovan laser en gang\*/**

**%macro** loopsubs();

proc sql noprint;

%do seg=**101** %to **108**;

%global sub\_&seg;

select distinct sub&seg. into :sub\_&seg. separated by " " from v1;

%end;

quit;

**%mend**;

%***loopsubs***();

**%macro** ***loop***;

%do ar = &min\_ar. %to &max\_ar.;

%if %substr(&ar, **5**, **2**) = **12** %then %let ar = %eval(&ar + 88);

%end;

**%mend**;

%***loop***;

**%macro** ***combine***;

%do ar = &min\_ar. %to &max\_ar.;

proc append

base = appended

data = &lib.**.A**nsokningBelopp\_&ar. force;

run;

%if %substr(&ar., **5**, **2**) = **12** %then %let ar = %eval(&ar. + 88);

%end;

**%mend** ;

%***combine***;

**proc** **sql** noprint;

select distinct name into :Vars separated by " " from dictionary.columns

where libname="SASHELP" and memname="CLASS" ;

**quit**;

%put &Vars;

**data** master;

do i = **1** to **5**;

j = **10**;

output;

end;

**run**;

**data** add;

do i = **15** to **20**;

j = **200**;

output;

end;

**run**;

/\* Append ADD dataset to MASTER dataset \*/

**proc** **append** base=master data=add;

**run**;

**proc** **print** data=master;

**run**;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*dates\*///////////

<http://www.sascommunity.org/wiki/Macro_Variables_of_Date_and_Time>

<https://v8doc.sas.com/sashtml/lrcon/zenid-63.htm>

<http://apprize.info/programming/sas_3/3.html>

**data** \_null\_;

today = Today();

call symputx('today',put(today, date9.));/\* tex 15DEC2017\*/

call symputx('today1',put(today, yymmddn8.));/\*tex 20171215\*/

call symputx('today2',put(today, yymmdd10.)); /\*tex 2017-12-15\*/

call symputx('rank',put(today, yymmn6.));/\* tex 201712\*/

call symputx('extr\_datum',put(intnx('month',today,**0**,'beginning'),date9.));

call symputx('eng\_dat',put(intnx('month',today,-**2**,'end'),date9.));

call symputx('ar',(year(today))); /\* år\*/

call symputx('period',put(intnx('month',today,**0**,'beginning'),yymmn6.));

call symputx('extr\_datum\_1',put(intnx('month',today,-**1**,'beginning'),date9.));

**Data** V1;

set Temppers.Kredittyp\_: ;

where (Engagemangs\_Datum between **"&start"d** and **"&end"d**) ;

if kredittyp="&kredittyp." and banktypsplit\_text=upcase("&kund.");

format Period yymmn6. lastDay yymmdd10.;

Period=Engagemangs\_Datum;

lastDay=intnx ('month',Engagemangs\_Datum,**0**,'E');

**run**;

**%macro** ***dates***;

/\*

date9. tex 15DEC2017

yymmddn8. 20171215

yymmn6. 201712

yymmdd10. 2017-12-15

year4. 2017

Month2. 12

yymmdd4. 1711\*/

%let date=20171215;

%let sas\_date=%sysfunc(inputn(&date.,yymmdd10.),date9.);

/\* typ 171215\*/

%let yymmdd=%substr(&date,3);

%let LastMnth = %sysfunc(intnx(month,"&sas\_date"d,-12),yymmddn8.);

%put &sas\_date &yymmdd &LastMnth;

**%mend**;

%***dates***;

DATENEW=left(PUT(today, yymmdd10.));

YYYY\_MM=substr(DATENEW,**1**,**7**);

call symputx('manad',YYYY\_MM);

**run**;

%let date=2012-01-01;

%let year= %sysfunc(year("&start"d));

%let sas\_date=%sysfunc(inputn(&date.,yymmdd10.),date9.);

%put &sas\_date;

%let date=201601;

%let sas\_date=%sysfunc(inputn(&date.,yymmn6.),date9.);

%put &sas\_date;

**data** date;

format date ddmmyy10.;

do date=**'01sep2016'd** to **'31dec2017'd** by **1**;

if day(date+**1**)=**1** then output;

DATENEW=PUT(DATE, yymmddd10.);

DATENEW1=substr(DATENEW,**1**,**7**);

xx=compress(tranwrd(DATENEW1, "-", ""));

Period1=input(strip(xx),best.); /\* string till numeric\*/

drop DATENEW DATENEW1 xx;

end;

**run**;

/\*\*\*\*\*\*\* loop\_days\*\*\*\*\*\*\*\*\*/

%let start=1jan2017;

%let end=28feb2017;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

%let indata\_lib=Dwfys;

%let lib=temppers;

**%macro** ***date\_loop\_days***;

/\*converts the dates to SAS dates\*/

%let start=%sysfunc(inputn(&start,anydtdte9.));

%let end=%sysfunc(inputn(&end,anydtdte9.));

%put &start &end;

/\*determines the number of months between the two dates\*/

%let dif=%sysfunc(intck(day,&start,&end));

%put &dif;

%do j=**0** %to &dif;

/\*advances the date i months from the start date and applys the DATE9. format\*/

%let date=%sysfunc(putn(%sysfunc(intnx(day,&start,&j,b)),date9.));

%put &date;

%let year= %sysfunc(year("&date"d));

Proc sql;

create table yyy\_&date. as

select distinct

a.extr\_datum,a.Abonnentnr,a.Kontonr,a.Engagemangs\_Datum,a.Kredittyp,a.personnr,a.saldo

from &indata\_lib.**.k**onton\_&year. (idxname=extr\_datum obs=&obs.) a

where a.extr\_datum=**"&date"d** and a.saldo>**0**

and a.Konto\_Slutdatum in (**"1jan1900"d**,**.**)

and a.limit\_sw ne "3" and a.saldo\_sw ne "3" and

a.Engagemangs\_Datum=intnx("month",a.extr\_datum,-**2**,"e");

quit;

%end;

**%mend**;

%***date\_loop\_days***;

**Data** V1;

set Jp.Jp210\_1706 (keep=orgnr lanj anstj);

Lan=put(lanj,z2.);

**run**;

/\*\*\*\*week (u starts on sundays while v starts on monday\*/

**proc** **sql** noprint;

select distinct min(Period1), max(Period1) into

:min\_ar separated by ' ',

:max\_ar separated by ' ' from date;

**quit**;

;

\*\*\*\*\*\*\*\*\*\*\*from string to datum\*/

**%macro** from\_chardate\_sasdate (datename,new\_date);

if &datename. ne " " then do;

length mo da yr &new\_date. **8**;

mo=input(substr(&datename.,**6**,**2**),**8.**);

da=input(substr(&datename.,**9**,**2**),**8.**);

yr=input(substr(&datename.,**1**,**4**),**8.**);

if mo >= **1** and mo <= **12** then mo1=**1**;

if da >= **1** and da <= **31** then da1=**1**;

if da1 ne **1** or mo1 ne **1** then &new\_date.=**.** ;

else

&new\_date.=mdy(mo,da,yr);

format &new\_date. yymmddd10. /\*mmddyy10.\*/;

drop mo da yr da1 mo1;

end;

run;

**%mend**;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*kollar dublett\*/

**proc** **sort** data=Temppers.Kreditbarometern\_201710 out=test;

by kontonr ;

**run**;

**data** a; set test;

by kontonr ;

if first.kontonr and last.kontonr then dublett=**0**;

else dublett=**1**;

**run**;

**Data** V1x;

set a;

where dublett=**1**;

**run**;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

**proc** **sort**

data = test

out = duplicates

uniqueout = singles /\*få ut endast unika\*/

nouniquekey;

by var1;

**run**;

\*samma som ovan;

**Proc** **sql**;

create table yyy as

select var1 from Test

group by var1

having count(var1)=**1** ;

**quit**;

%let Mail\_to= ronny.sentongo@uc.se;

%let Prog\_end = %sysfunc(TIME(),time.);

%let nobs = %nlobs(&outfile.);

%let Duration = %sysfunc(intck(minute,"&Prog\_start."t,"&Prog\_end."t));

filename outbox email

to="&Mail\_to."

type='text/html'

subject="Filen är klar"

from="&Mail\_to.";

data \_null\_;

file outbox

to=("&Mail\_to." ) ;

from="&Mail\_to.";

PUT "&outfile. uppdaterad med &nobs. poster. Uppdateringstid ca &duration.min";

put "Startade &prog\_start. och slutade &sysday. &Prog\_end";

run;

/\* skickar mail till mig själv\*/

/\*högst upp\*/

**data** \_null\_;

time = TIME();

call symputx('Prog\_start',put(time,time.));

**run**;

%let Mail\_to= ronny.sentongo@uc.se;

%let t\_time = %sysfunc(TIME(),time.);

**data** \_null\_;

today = today();

call symputx('Duration',intck('minute',**"&Prog\_start."t**,**"&t\_time."t**));

call symputx('today',put(today,yymmdd10.));

**run**;

ods html body=temp rs=none;

**proc** **contents** data=sashelp.class position varnum;

title "Program started at &prog\_start and finished &sysday:

&today, &t\_time duration approx. &duration minutes";

**run**;

ods html close;

ODS LISTING;

filename temp email to="&Mail\_to"

subject="Filen är klar"

type="text/html"

from="&Mail\_to.";

**proc** **datasets** library=WORK KILL NOLIST;

**run**;

%let start=01jun2008;

%let end=01dec2009;

/\*data just\_these\_dates;

do date=200806,200810;

output;

end;

run;\*/

**%macro** ***date\_loop***;

/\*converts the dates to SAS dates\*/

%let start=%sysfunc(inputn(&start,anydtdte9.));

%let end=%sysfunc(inputn(&end,anydtdte9.));

/\*determines the number of months between the two dates\*/

%let dif=%sysfunc(intck(month,&start,&end));

%do i=**0** %to &dif;

/\*advances the date i months from the start date and applys the DATE9. format\*/

%let date=%sysfunc(putn(%sysfunc(intnx(month,&start,&i,b)),date9.));

%put &date;

data \_null\_;

call symputx('Month',put(**"&date"d**, yymmn6.));/\* tex 201601\*/

call symputx('Month\_01',put(**"&date"d**, yymmddn8.));/\* tex 20160101\*/

call symputx('Month\_2',put(intnx('month',**"&date"d**,-**2**,'e'),yymmn6.)); /\*Engagemangs\_Datum\*/

call symputx('Year',(year(**"&date"d**))); /\* år\*/

run;

/\*data tmp;

set just\_these\_dates

(where=(date = (&month.))) ;

run;

%let nobs = %nlobs(tmp);

%if &nobs ne 0 %then %do;\*/

Data V1\_&month.;

set sashelp.class (obs=**4**);

month=**"&date"d**;

format month date9.;

period=&Month.;

period2=&Month\_2.;

Year=&year.;

Month\_01=&Month\_01.;

run;

/\*%end;\*/

%end;

**%mend**;

%***date\_loop***;

**proc** **sql**;

select memname into :dellist separated by ' ' from dictionary.tables

where libname=upcase("&libname.") and nobs=**0**;

**quit**;

%put &dellist;

**proc** **datasets** lib=&libname. memtype=data nolist;

delete &dellist.;

**quit**;

**proc** **sort** data=&lib.**.P**rog&mm.\_&ar.;

by riskprof modell;

**run**;

**proc** **logistic** data=&lib.**.P**rog&mm.\_&ar. (where=(riskprof not in (**109**,**110**)));

by riskprof modell;

class &perf.;

model &perf.=&prognos.;

ods output Association=v2 (where=(label2="Somers' D"));/\* SomersD = AR i SAS cvalue2\*/

**run**;

proc sql;  
create table X(drop=patient\_b) as  
select distinct \*  
from patient a  
left join encounter(rename=(patient=patient\_b)) b  
on a.patient=b.patient\_b  
;  
quit;

**/\*mycalendar\*/**

**PROC** **IMPORT** OUT= datasets\_betas

DATAFILE= "&MU.\0-Ovrigt\Övrigt\Ronny\Kopia av MyCalendar.xlsx"

DBMS=xlsx REPLACE ;

sheet="Calendar";

GETNAMES=YES;

**RUN**;

**Data** temppers.MyCalendar;

format date yymmdd10.;

set datasets\_betas (drop=MonthNumber QuarterNumber);

**run**;

**proc** **datasets** library=temppers;

modify MyCalendar;

index create date;

**run**;

/\*workingdays\*/

**PROC** **IMPORT** OUT= v1

DATAFILE= "&MU.\0-Ovrigt\Övrigt\Ronny\Kopia av MyCalendar.xlsx"

DBMS=xlsx REPLACE ;

sheet="weekdays";

GETNAMES=YES;

**RUN**;

**PROC** **IMPORT** OUT= v2

DATAFILE= "&MU.\0-Ovrigt\Övrigt\Ronny\Kopia av MyCalendar.xlsx"

DBMS=xlsx REPLACE ;

sheet="holidays";

GETNAMES=YES;

**RUN**;

**Data** weekdays;

format date yymmdd10.;

set v1 (drop=b c d e f g);

**run**;

**Data** Holidays;

format date yymmdd10.;

set v2 ;

**run**;

**Proc** **sql**;

create table yyy as

select a.\*,b.date as Holiday from weekdays a left join Holidays b on a.date=b.date;

**quit**;

**Data** temppers.MyCalendar\_Workingdays;

set yyy (where=(Holiday=**.**));

drop Holiday;

**run**;

**proc** **datasets** library=temppers;

modify MyCalendar\_Workingdays;

index create date;

**run**;

**proc** **sql**;

select memname into :dellist separated by ' ' from dictionary.tables

where libname=upcase("&libname.") and nobs=**0**;

**quit**;

%put &dellist;

**proc** **datasets** lib=&libname. memtype=data nolist;

delete &dellist.;

**quit**;

**data** date;

format date yymmdd10.;

do date=**'28feb2015'd** to **'31jan2019'd** by **1**;

output;

end;

**run**;

**data** getweek;

set date;

/\* year.1 dec year.2 jan\*/

Financial\_Year = IntNx("Year.4", date, **0**, "END");

Financial\_Year1 = IntNx("Year.4", date, **1**, "END");

day\_intervall = intnx('day', Financial\_Year, -**5**);

/\*antag att från 25-25e\*/

DATENEW=PUT(Financial\_Year, yymmddd10.);

DATENEW1=substr(DATENEW,**1**,**7**);

xx=compress(DATENEW1||"-25");

Newdate\_25 = input(xx,yymmdd10.);

drop DATENEW DATENEW1 xx;

format day\_intervall Financial\_Year1 Financial\_Year yymmdd10.;

**run**;

let

Källa = Folder.Files("C:\Users\rso363\RONSEN\Football\excel"),

#"Borttagna kolumner" = Table.RemoveColumns(Källa,{"Date accessed", "Date modified", "Date created", "Attributes", "Folder Path"}),

#"Lägg till egen" = Table.AddColumn(#"Borttagna kolumner", "Import", each Excel.Workbook([Content])),

#"Borttagna kolumner1" = Table.RemoveColumns(#"Lägg till egen",{"Content"}),

#"Expanderad Import" = Table.ExpandTableColumn(#"Borttagna kolumner1", "Import", {"Data", "Item"}, {"Data", "Item"}),

#"Omdöpta kolumner" = Table.RenameColumns(#"Expanderad Import",{{"Item", "Sheet\_name"}}),

#"Lägg till egen1" = Table.AddColumn(#"Omdöpta kolumner", "Noheaders", each Table.PromoteHeaders([Data]))

in

#"Lägg till egen1"

let

Source = Folder.Files("L:\2-UCRF\1-UCRFRiskIII\0-Ovrigt\1-AdhocUtredningar\Diverse utredningspunkter UCRF 2017\Export\_Från\_SAS\AB\csv"),

#"Added Custom" = Table.AddColumn(Source, "Custom", each Table.AddColumn(Source, "Custom", each Table.PromoteHeaders(Csv.Document([Content],[Delimiter=";", Encoding=1252]))))

in

#"Added Custom"

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*moving average excel\*/

=IF(ISBLANK(sum(abonnentnr[Saldo]));BLANK();

CALCULATE(sum(abonnentnr[Saldo]);

DATESINPERIOD(abonnentnr[Manad];

LASTDATE(abonnentnr[Manad]);-3;MONTH)))

**proc** **datasets** library=WORK KILL NOLIST;

**run**;

%let alder=34;

%let postnr=16843;

%let mm\_dd=0705;

%let kön=1; /\* 0=man 1=kvinna\*/

**Data** V1;

set Resktr.Ref\_dev\_180101 (keep=personnr alder kon tink4 postnr);

where kon=&kön. and alder=&alder. and postnr="&postnr."

and substr(personnr,**4**,**4**)="&mm\_dd.";

Lon=tink4/**12**;

**run**;

/\* if exist\*/

%let min\_start=201801;

%let max\_end=202001;

%let min\_year=%substr(&min\_start.,1,4);

%let max\_year=%substr(&max\_end.,1,4);

%put &min\_year &max\_year;

option nomlogic nomprint nosource nonotes nosymbolgen; /\*off with log notes\*/

**%macro** dates(nr=);

%let start\_date=&min\_start.;

%let end\_date=&max\_end.;

%let start=%sysfunc(inputn(&start\_date.,yymmn6.),date9.);

%let end=%sysfunc(inputn(&end\_date.,yymmn6.),date9.);

%let intervall=month;/\* loop days(day) month \*/

/\*converts the dates to SAS dates\*/

%let start=%sysfunc(inputn(&start,anydtdte9.));

%let end=%sysfunc(inputn(&end,anydtdte9.));

%put &start &end;

/\*determines the number of months between the two dates\*/

%let dif=%sysfunc(intck(&intervall.,&start,&end));

%put &dif;

%do j=**0** %to &dif;

/\*advances the date i months from the start date and applys the DATE9. format\*/

%let date=%sysfunc(putn(%sysfunc(intnx(&intervall.,&start,&j,b)),date9.));

%let test=%sysfunc(inputn(&date, date9.), year4.);

%let Qtr=%sysfunc(inputn(&date, date9.), year4.)\_qtr\_%sysfunc(putn(%substr(%sysfunc(inputn(&date, date9.), yyqn3.),3), z2));

%let Kvartal=%sysfunc(putn(%substr(%sysfunc(inputn(&date, date9.), yyqn3.),3), z2));

%if &Kvartal.=&nr. %then %do;

Data V1;

set date;

where date=**"&date."d**;

Test=**"&date."d**;

format test yymmdd10.;

Kvartal=&Kvartal.;

run;

proc append

base = appended\_&qtr.

data = v1 force;

run;

%end;

%end;

**%mend**;

%***dates***(nr=**04**);

**%macro** test(Dsetname);

%do ar=&min\_year. %to &max\_year.;

%do i=**1** %to **4**;

%if %length(&i.) = **1** %then %let i=0&i.;

%let dsn = &Dsetname.&i;

%if %sysfunc(exist(&dsn.)) %then %do;

**proc** **append**

base=Finale

data=&dsn. force;

**run**;

%end;

%end;

%end;

**%mend** test;

%***test***(Appended\_&ar.\_qtr\_);

**%macro** ***test***;

%let start\_date=&min\_start.;

%let end\_date=&max\_end.;

%let start=%sysfunc(inputn(&start\_date.,yymmn6.),date9.);

%let end=%sysfunc(inputn(&end\_date.,yymmn6.),date9.);

%let intervall=month;/\* loop days(day) month \*/

/\*converts the dates to SAS dates\*/

%let start=%sysfunc(inputn(&start,anydtdte9.));

%let end=%sysfunc(inputn(&end,anydtdte9.));

%put &start &end;

/\*determines the number of months between the two dates\*/

%let dif=%sysfunc(intck(&intervall.,&start,&end));

%put &dif;

%do j=**0** %to &dif;

/\*advances the date i months from the start date and applys the DATE9. format\*/

%let date=%sysfunc(putn(%sysfunc(intnx(&intervall.,&start,&j,b)),date9.));

%let new\_format=%sysfunc(inputn(&date, date9.), yymmdd4.);

%let dsn = Ss.Ss020\_&new\_format.;

%if %sysfunc(exist(&dsn.)) %then %do;

Data V1\_&new\_format.;

set Ss.Ss020\_&new\_format. (obs=**10**);

run;

%put &new\_format;

%end;

%end;

**%mend**;

%***test***;

/\*macro in and not in operator\*/

options minoperator mindelimiter=",";

**%macro** in(test=);

%let list=A,B,C;

%if &test. in (&list.) %then %do;

Data V1;

set sashelp.class (obs=**5**);

IN="IN";

run;

%end;

%else %if not (&test. in &list.) %then %do;

Data V2;

set sashelp.class (obs=**10**);

IN="NOT IN";

run;

%end;

**%mend**;

%***in***(test=A B C);

**%macro** ***ok***;

Data tmp;

set log2;

where l\_r like 'ERROR%' or l\_r like 'WARNING%';

run;

%let nobs = %nlobs(tmp);

%if &nobs. = **0** %then %do;

Data A\_errors\_warnings;

Syntax="OK";

run;

%end;

%else %if &nobs. ne **0** %then %do;

Data A\_errors\_warnings;

set tmp;

Rename l\_r=Log\_info;

run;

%end;

**%mend**;

%***ok***;

**%macro** iterm(beg,end);

%let lst=a b c d e f g h i j k l m n o p q r s t u v w x y z;

%let start=%sysfunc(indexc(%sysfunc(compress(&lst.)),&beg.));

%let finish=%sysfunc(indexc(%sysfunc(compress(&lst.)),&end.));

%do i = &start. %to &finish.;

%put %scan(&lst.,&i.);

%let alfa=%scan(&lst.,&i.);

Data Alfabets\_&alfa.;

Id\_alfa="&alfa.";

Id=&i.;

run;

Data V1;

set Alfabets\_:;

run;

%end;

**%mend**;

/\* Just pass in starting and ending value \*/

%***iterm***(a,z);

**proc** **datasets** lib=work memtype=data nolist;

delete Alfabets\_:;

**quit**;

%let name\_list =1 2 4 6 13 29 32;

**%macro** ***do\_loop\_non\_sequential***;

%let cnt = %sysfunc(CountW(&name\_list.));

%do loop\_name\_list = **1** %to &cnt.;

%let Next\_name=%scan(&name\_list,&loop\_name\_list.);

Data V&Next\_name.;

set sashelp.class;

test="&Next\_name";

run;

%end;

**%mend**;

%***do\_loop\_non\_sequential***;

%let name\_list =201605 201612 201702 201801;

option nomlogic nomprint nosource nonotes nosymbolgen;

**%macro** ***do\_loop\_non\_sequential\_dates***;

%let cnt = %sysfunc(CountW(&name\_list.));

%do loop\_name\_list = **1** %to &cnt.;

%let Next\_name=%scan(&name\_list,&loop\_name\_list.);

%let year=%sysfunc(inputn(&Next\_name.,yymmn6.),year4.); /\*\*/

%Let extr\_datum=%Sysfunc(Intnx(Month, "(%sysfunc(inputn(&Next\_name.,yymmn6.),date9.)"d, 0, b), date9.);

%Let datum=%Sysfunc(Intnx(Month, "(%sysfunc(inputn(&Next\_name.,yymmn6.),date9.)"d, 0, e), yymmdd10.);

%put &year &datum &extr\_datum;

Data V\_&Next\_name.;

set Dwfys.Konton\_&year. (where=(extr\_datum=**"&extr\_datum."d**) keep=extr\_datum obs=**10**);

test="&Next\_name";

Datum="&datum.";

run;

%end;

**%mend**;

%***do\_loop\_non\_sequential\_dates***;

/\*\* This macro will allow you to step through the alphabet on a %DO loop.\*\*/

**%macro** iterm(beg,end);

%do i = %sysfunc(rank(&beg)) %to %sysfunc(rank(&end));

%put %sysfunc(byte(&i));

%let j=%sysfunc(byte(&i));

Data \_Alfa\_&j.;

Alfa="&j";

run;

Data Alfa;

set \_Alfa:;

run;

%end;

proc datasets lib=work memtype=data nolist;

delete \_Alfa\_:;

quit;

**%mend**;

/\* Just pass in starting and ending value \*/

%***iterm***(A,Z);

%let Array\_name=Array\_name;

%let Array\_vars=age height;

%let Outfile=v1;

**Data** &Outfile.;

set sashelp.class;

array &Array\_name. &Array\_vars.;

do over &Array\_name.;

A\_SUM=SUM(OF &Array\_name.(\*));

A\_MEAN=MEAN(OF &Array\_name.(\*));

A\_MIN=MIN(OF &Array\_name.(\*));

end;

**run**;

**%macro** summstat(dsin=,dsout=,classvar=, vvars=,outprt=NOPRINT );

proc summary data=&dsin nway &outprt;

class &classvar;

%if (&vvars ne ) %then %do;

var &vvars;

output out=&dsout (drop=\_type\_ \_freq\_) /\*n= mean= std= median= min= max=\*/ sum= /autoname;

%end;

%else %do;

output out=&dsout (drop=\_type\_ rename=(\_freq\_=N));

%end;

run;

/\*vvars variable(r) som ska summeras, classvar groupering\*/

**%mend** summstat;

%***summstat***(dsin=&Outfile.,dsout=v2,classvar=sex,vvars=&Array\_vars. A\_SUM, outprt=);

/\* proc sql noprint;\*/

/\* proc sql noprint;\*/

**%macro** sql\_noprint(dset=,tabell\_name=,macro\_name=);

%global &macro\_name;

proc sql noprint;

select distinct &tabell\_name. into

:&macro\_name. separated by " " from &dset.;

quit;

**%mend**;

%***sql\_noprint***(dset=sashelp.class,tabell\_name=age,macro\_name=var\_age);

%***sql\_noprint***(dset=sashelp.class,tabell\_name=sex,macro\_name=var\_sex);

%put &var\_age &var\_sex;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* columns \*

**%macro** find\_column\_names(lib=,tabell=, table=, varname=);

**%macro** ***\_***; **%mend** \_;

**proc** **sql** noprint;

select distinct name into :Vars separated by " " from dictionary.columns

where libname=upcase("&lib.") and memname=upcase("&tabell") ;

**quit**;

\* Find max length;

%let max\_length=0;

%do i=**1** %to %sysfunc(countw(&vars));

%let grej=%scan(&vars, &i);

%if %length(&grej) > &max\_length %then %let max\_length=%length(&grej);

%end;

%put length &max\_length;

**data** &table;

length &varname $&max\_length;

%do i=**1** %to %sysfunc(countw(&vars));

%let grej=%scan(&vars, &i);

&varname = "&grej";

output;

%end;

**run**;

**%mend**;

%***find\_column\_names***(lib=sashelp,tabell=class,table=test, varname=Myvariable);

%***find\_column\_names***(lib=sashelp,tabell=Citimon,table=test1, varname=Myvariable);

%***find\_column\_names***(lib=risk3,tabell=Riskab\_201801,table=test2, varname=Myvariable);

/\* kör skapad tabell för selekteringar\*/

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* tables \*

**%macro** find\_table\_names(lib=, table=, varname=);

**%macro** ***\_***; **%mend** \_;

**proc** **sql** noprint;

select distinct memname into :vars separated by " " from dictionary.tables

where libname=upcase("&lib.");

**quit**;

\* Find max length;

%let max\_length=0;

%do i=**1** %to %sysfunc(countw(&vars));

%let grej=%scan(&vars, &i);

%if %length(&grej) > &max\_length %then %let max\_length=%length(&grej);

%end;

%put length &max\_length;

**data** &table;

length &varname $&max\_length;

%do i=**1** %to %sysfunc(countw(&vars));

%let grej=%scan(&vars, &i);

&varname = "&grej";

output;

%end;

**run**;

**Data** &table.;

set &table.;

lib="&lib..";

Dset=cat(lib,&varname.);

drop lib;

**run**;

**%mend**;

%***find\_table\_names***(lib=sashelp,table=test, varname=Myvariable);

%***find\_table\_names***(lib=temppers,table=test1, varname=Myvariable);

\* Skapa en tabell grupperade variabler;

**proc** **sql** noprint;

create table vars as

select distinct name from dictionary.columns

where libname="WORK" and memname="DATA" and substr(name,**1**,**1**)="x" ;

**quit**;

/\* delete empty tables\*/

**%macro** delete\_empty\_tables(lib=);

**%macro** ***\_***; **%mend** \_;

**proc** **sql** noprint;

select memname into :dellist separated by " " from dictionary.tables

where libname=upcase("&lib.") and nobs=**0**;

**quit**;

**proc** **datasets** lib=&lib. memtype=data nolist;

delete &dellist.;

**quit**;

**%mend**;

%***delete\_empty\_tables***(lib=work);

%***delete\_empty\_tables***(lib=test);

**proc** **datasets** library=WORK KILL NOLIST;

**run**;

%let name\_list =201701 201708 201501;

option nomlogic nomprint nosource nonotes nosymbolgen;

**%macro** ***do\_loop\_non\_sequential\_dates***;

**%macro** ***\_***; **%mend** \_;

%let cnt = %sysfunc(CountW(&name\_list.));

%do loop\_name\_list = **1** %to &cnt.;

%let Next\_name=%scan(&name\_list,&loop\_name\_list.);

%let tidpunkt=%sysfunc(inputn(&Next\_name.,yymmn6.),yymmdd6.);

%let Observationstidpunkt=%sysfunc(inputn(&Next\_name.,yymmn6.),yymmddn8.);

**data** blanco\_&Next\_name.;

set resktr.ref\_dev\_&tidpunkt.(where=(sumsaldo\_BLANCO\_engdat0 >**0** )

keep= personnr ucrp\_me sumsaldo\_BLANCO\_engdat0 ant\_BLANCO\_engdat0 tink4 obs=**10**);

Observationstidpunkt=&Observationstidpunkt.;

**run**;

%end;

**%mend**;

%***do\_loop\_non\_sequential\_dates***;

/\*&lib.713.train\_me\_713\_36m

data temp;

set &lib..Train\_&modell.\_&seg.;

n=ranuni(8);

proc sort data=temp;

by n;

data v1;

set temp nobs=nobs;

if \_n\_<=.3\*nobs;

drop n;

run;

Data &lib..Train\_&modell.\_&seg.;

set v1;

run;

\*/

%let segment=700 711 712 713 720 751 752;

**%macro** ***remove***;

**%macro** ***\_***; **%mend** \_;

%let name\_list =&segment.;

%let yyy=xxx;

%let cnt = %sysfunc(CountW(&name\_list.));

%do loop\_name\_list = **1** %to &cnt.;

%let Next\_name=%scan(&name\_list,&loop\_name\_list.);

**proc** **datasets** lib=gdpr&Next\_name. memtype=data nolist;

delete Est\_final Model\_final pe\_final type3\_final;

**quit**;

%end;

**%mend**;

%***remove***;

/\*group by in SAS keeping all variables\*/

**proc** **sort** data=Temppers.Inp\_hss\_201501(where=(period ne **.**)) out=Inp\_hss\_201501;

by personnr descending period ;

**run**;

**data** Inp\_hss\_201501\_adj;

set Inp\_hss\_201501;

by personnr;

retain Max\_period;

if first.personnr then do;

Max\_period=period;

format Max\_period yymmdd10.;

end;

Cutoff = intck("month",period,Max\_period);

if Cutoff<=**12**;

drop Cutoff Max\_period;

**run**;

/\*group by in SAS keeping all variables loop\*/

**proc** **datasets** lib=WORK memtype=data nolist;

delete Inp\_hss\_adj;

**quit**; /\*avoid duplicates in append \*/

**%macro** ***inp***;

**%macro** ***\_***; **%mend** \_;

option nomlogic nomprint nosource nonotes nosymbolgen; /\*off with log notes\*/

%let start\_date=201401;

%let end\_date=201501;

%let start=%sysfunc(inputn(&start\_date.,yymmn6.),date9.);

%let end=%sysfunc(inputn(&end\_date.,yymmn6.),date9.);

%let intervall=month;/\* loop days(day) month \*/

/\*converts the dates to SAS dates\*/

%let start=%sysfunc(inputn(&start,anydtdte9.));

%let end=%sysfunc(inputn(&end,anydtdte9.));

/\*determines the number of months between the two dates\*/

%let dif=%sysfunc(intck(&intervall.,&start,&end));

%put &dif;

%do j=**0** %to &dif %by **12**;

/\*advances the date i months from the start date and applys the DATE9. format\*/

%let date=%sysfunc(putn(%sysfunc(intnx(&intervall.,&start,&j,b)),date9.));

%let ar=%sysfunc(inputn(&date, date9.), yymmn6.);

%put &ar;

**proc** **sort** data=gdpr.Inp\_hss\_&ar.(where=(period ne **.**)) out=Inp\_hss\_&ar.;

by personnr descending period ;

**run**;

**data** Inp\_hss\_&ar.\_;

set Inp\_hss\_&ar.;

by personnr;

retain Max\_period;

if first.personnr then do;

Max\_period=period;

format Max\_period yymmdd10.;

end;

Cutoff = intck("month",period,Max\_period);

if Cutoff<=**12**;

drop Cutoff Max\_period;

**run**;

**proc** **append**

base = Inp\_hss\_adj

data = Inp\_hss\_&ar.\_ force;

**run**;

%end;

option mlogic mprint source notes symbolgen;

**%mend**;

%***inp***;