```
********************* Code for the seminar 12 Apr 2012 ***************************;

libname Seminar 'C:\Users\ninmal\Desktop\Proc SQL\Database';

******************************************************************************
PROC SQL SYNTAX
******************************************************************************;
PROC SQL;
            Create table as
            SELECT
            FROM
            WHERE;
QUIT;

******************************************************************************
Example 1:

1 a) Select men
1 b) Select Swedish men;
******************************************************************************;
Proc SQL;
            Create table men as
            Select *
            FROM Seminar.PERSON
            WHERE KON=1 and BIRTH_COUNTRY='Sverige';
QUIT;

******************************************************************************
Example 2:

2 a) As / rename variabel and Label
2 b) Sort data "ORDER BY"
******************************************************************************;
Proc SQL;
            Create table men as
            Select LOPNR, KON as SEX LABEL='SEX', BIRTH_DATE, Birth_country
            FROM Seminar.PERSON
            WHERE KON=1 and BIRTH_COUNTRY='Sverige'
            ORDER BY BIRTH_DATE;
QUIT;
```

```
**************************************************************************************************
Example 3

All SAS function is working
Substrn format ;
**************************************************************************************************;
Proc SQL;
            Create table canceyear as
            Select *, input(substrn(DIAGDATE_TEXT,1,4),best4.) as CANCERDIAG_YEAR
format=best4.
            FROM Seminar.CANCER;
QUIT;


**************************************************************************************************
Example 4

Create a new column (new variable) with name, format, label ;
**************************************************************************************************;
Proc SQL;
            Create table deathage as
            Select *, int((Death_date-Birth_date)/365.24) as Death_age format=best4.
            FROM Seminar.PERSON;
QUIT;

**************************************************************************************************
Example 5

Frequency table of cancer incidence per year
**************************************************************************************************;
Proc SQL;
            Create table Nr_cancer as
            Select DIAGYEAR, count(LOPNR) as CANCER_inc LABEL='CANCER INCIDENCE'
            FROM Seminar.CANCER
            WHERE DIAGYEAR>1990
            GROUP BY DIAGYEAR
            ORDER BY CANCER_inc;
QUIT;
```

```
********************************************************************************
Example 6

Select all individuals who have Multiple Cancers

Where statement: look in the input data
Having statement: look in the output data
********************************************************************************;
Proc SQL;
        Create table multi_cancer_incidence as
        Select LOPNR, count(LOPNR) as NR_CANCER LABEL='#CANCER_inc'
        FROM Seminar.CANCER
        GROUP BY LOPNR
        HAVING COUNT(*)>1;
QUIT;




********************************************************************************
Example 7

Join

7a) INNER Join (if a and b)
Same variable name ;
********************************************************************************;
PROC SQL;
        Create table Person_death AS
        SELECT MGR.LOPNR, DEATH.DEATHDATE, DEATH.DEATHYEAR, DEATH.ULORSAK
        FROM SEMINAR.MGR_REG MGR, SEMINAR.Death as DEATH
        WHERE MGR.LOPNR=DEATH.LOPNR;
QUIT;

********************************************************************************
7b)
Same code with merge
(Do not work have to sort)
********************************************************************************;
Data Person_death_merge;
merge SEMINAR.MGR_REG(in=a) SEMINAR.Death(in=b);
by LOPNR;
if a and b;
run;

Proc sort data=SEMINAR.MGR_REG out=sortMGR;by LOPNR;run;
Proc sort data=SEMINAR.Death out=sortDOD;by LOPNR;run;

Data Person_death_merge;
merge SortMGR(in=a keep=LOPNR) SortDOD(in=b);
```

```
by LOPNR;
if a and b;
run;



*****************************************************************************************
7c) Left OUTER Join (if a)
Same variable name
****************************************************************************************;
PROC SQL;
		Create table Person_death AS
		SELECT MGR.LOPNR, DEATH.DEATHDATE, DEATH.DEATHYEAR, DEATH.ULORSAK
		FROM SEMINAR.MGR_REG MGR left JOIN SEMINAR.Death as DEATH ON
		MGR.LOPNR=DEATH.LOPNR;
QUIT;

*****************************************************************************************
7d) Again inner join with "on"
Same variable name
****************************************************************************************;

PROC SQL;
		Create table Person_death AS
		SELECT MGR.LOPNR, DEATH.DEATHDATE, DEATH.DEATHYEAR, DEATH.ULORSAK
		FROM SEMINAR.MGR_REG MGR JOIN SEMINAR.Death as DEATH ON
		MGR.LOPNR=DEATH.LOPNR;
QUIT;

*****************************************************************************************
7e) Inner join with different variable names
****************************************************************************************;
PROC SQL;
		Create table MOTHER_death AS
		SELECT MGR.LOPNRMOR, DEATH.DEATHDATE, DEATH.DEATHYEAR, DEATH.ULORSAK
		FROM SEMINAR.MGR_REG MGR, SEMINAR.Death as DEATH
		WHERE MGR.LOPNRMOR=DEATH.LOPNR;
QUIT;

*****************************************************************************************
Example 8
Many to many Join
Cartesian product

Select all children and their mothers and the education for the mother
A mother can have several children and have one observation per year in the education registry
******************************************************************************************
****************;
PROC SQL;
		Create table Mother_education AS
		SELECT MGR.LOPNR, MGR.LOPNRMOR, EDU.Highest_education, EDU.REGYEAR
		FROM SEMINAR.MGR_REG MGR, SEMINAR.EDUCATION_REG as EDU
```

```
            WHERE MGR.LOPNRMOR=EDU.LOPNR
            ORDER BY LOPNRMOR, LOPNR, REGYEAR;
QUIT;
/*
NOTE: Table WORK.MOTHER_EDUCATION created, with 1268 rows and 7 columns.
*/

*********************************************************************************
Check LOPNRMOR=3974325, have two children with LOPNR=5063934 and LOPNR=6561090
*********************************************************************************;
PROC SQL;
            Create table MOTHER_TEST AS
            SELECT *
            FROM MOTHER_EDUCATION
            WHERE LOPNRMOR=3974325;
QUIT;

*********************************************************************************
Test with merge

Have to sort first
*********************************************************************************;
proc sort data=SEMINAR.MGR_REG out=sortMGR; by lopnrmor;run;
proc sort data=SEMINAR.EDUCATION_REG out=sortedu; by lopnr;run;

Data MOTHER_EDUCATION_Merge;
merge SortMGR(in=a Keep=LOPNRmor) sortEDU(in=b rename=LOPNR=LOPNRMOR);
by LOPNRmor;
if a and b;
run;
/*
NOTE: MERGE statement has more than one data set with repeats of BY values.
NOTE: There were 28222 observations read from the data set WORK.SORTMGR.
NOTE: There were 543377 observations read from the data set SEMINAR.EDUCATION.
NOTE: The data set WORK.TESTTTT has 1234 observations and 6 variables.
*/

*********************************************************************************
Check LOPNRMOR=3974325 have two children with LOPNR=5063934 and LOPNR=6561090
*********************************************************************************;
PROC SQL;
            Create table MOTHER_TEST_merge AS
            SELECT *
            FROM MOTHER_EDUCATION_MERGE
            WHERE LOPNRMOR=3974325;
QUIT;
```