PROC IMPORT

PROC IMPORT reads data from an external data source and writes it to a SAS data set. External source should be Excel sheets, CSV, Access files etc...

PROC IMPORT reads data from an external data source and writes it to a SAS data set. Base SAS can import delimited files. In delimited files, a delimiter--such as a blank, comma, or tab--separates columns of data values. If you have license for SAS/ACCESS Interface to PC Files, additional external data sources can include such files as Microsoft Access Database, Excel files, and Lotus spreadsheets.

Syntax:-

```
PROC IMPORT DATAFILE=”RAWDATALOCATION”
                  OUT=LIBREF.DATASET
                  DBMS=SOURCE IDENTIFIER
                  REPLACE;
GETNAMES=YES|NO;
DATAROW=RECORDNO;
GUESSINGROWS=NO OF ROWS
SHEET="SHEET_NAME"
RUN;
```

Examples:-

Datafile=*”File location”*

Specifies the complete path and filename.ext of the input PC file, spreadsheet, or delimited external file. If the name does not include special characters (such as the backslash in a path), lowercase characters, or spaces, you can omit the quotes.

OUT=*SAS-data-set*

Identifies the output SAS data set with either a one- or two-level SAS name (library and member name). If the specified SAS data set does not exist, PROC IMPORT creates it. If you specify a one-level name, PROC IMPORT uses the WORK library.

TABLE=*"tablename"*

Specifies the table name of the input DBMS table. If the name does not include special characters (such as question marks), lowercase characters, or spaces, you can omit the quotes. Note that the DBMS table name may be case-sensitive.

DBMS=*identifier*

Specifies the type of data to import. For example, DBMS=DBF specifies to import a dBASE file. For PC files, spreadsheets, and delimited external files, you do not have to specify DBMS= if the filename specified with DATAFILE= contains a valid extension so that PROC IMPORT can recognize the type of data. For example, PROC IMPORT recognizes the filename ACCOUNTS.WK1 as a Lotus 1 spreadsheet and the filename MYDATA.CSV as a delimited

external file that contains comma-separated data values. PROC IMPORT can recognize the difference between Excel Version 4 and Version 5 spreadsheets when you use the extension .XLS, regardless of whether you specify DBMS=EXCEL, DBMS=EXCEL4, or DBMS=EXCEL5. However, you must specify DBMS=EXCEL97 to import Excel 97 spreadsheets. If you do not specify an identifier or if the extension of the filename is not recognized, an error is returned.
To import a DBMS table, you must specify DBMS= using a valid database product. For example, DBMS=ACCESS imports a Microsoft Access table.

The DBMS= specification can include the values listed in the following table:

Available DBMS Specifications

Identifier	Input Data Source	Extension
ACCESS	Microsoft Access database	.MDB
DBF	dBASE file	.DBF
WK1	Lotus 1 spreadsheet	.WK1
WK3	Lotus 3 spreadsheet	.WK3
WK4	Lotus 4 spreadsheet	.WK4
EXCEL	Excel Version 4 or 5 spreadsheet	.XLS
EXCEL4	Excel Version 4 spreadsheet	.XLS
EXCEL5	Excel Version 5 spreadsheet	.XLS
EXCEL97	Excel 97 spreadsheet	.XLS
DLM	delimited file (default delimiter is a blank)	.*
CSV	delimited file (comma-separated values)	.CSV
TAB	delimited file (tab-delimited values)	.TXT

REPLACE
Overwrites an existing SAS data set. If you do not specify REPLACE, PROC IMPORT does not overwrite an existing data set.

DATAROW=1 to 32767
Starts reading data from the specified row number in the delimited text file.
When GETNAMES=NO, DATAROW must be equal to or greater than 1.
When GETNAMES=YES, DATAROW must be equal to or greater than 2.

DELIMITER

Specifies the delimiter that separates columns of data in the input file. You can specify the delimiter as a single character or as a hexadecimal value. For example, if columns of data are separated by an ampersand, specify DELIMITER='&'. If you do not specify DELIMITER=, the IMPORT procedure assumes that the delimiter is a space.
Valid values are ***anychar|nn|x|space.***

GETNAMES=YES/NO

Specifies whether the IMPORT procedure generate SAS variable names from the data values in the first record in the input file.

GUESSINGROWS=

Specifies the number of rows of the file to scan to determine the appropriate data type and length for the columns. The scan data process scans from row 1 to the number that is specified by the GUESSINGROWS option.
Default rows=20
Valid values is 1 to 32767

Examples:-

```
Proc import
Datafile="C:\Documents and Settings\Administrator\Desktop\sample.xls"
Out=work.Sample dbms=excel;
Run;
```

```
Proc import
Datafile="C:\Documents and Settings\Administrator\Desktop\sample.xls"
Out=work.Sample dbms=Excel;
Run;
```

If dataset (sample) already is there in library use replace option to overwrite on existing dataset.

```
Proc import
Datafile="C:\Documents and Settings\Administrator\Desktop\sample.xls"
Out=work.Sample dbms=excel replace;
Run;
```

By default variable names will come into output dataset from source file
It means Getnames=yes;
If you don't want to read column names use Getnames=no

```
Proc import
Datafile="C:\Documents and Settings\Administrator\Desktop\sample.xls"
Out=work.Sample dbms=excel replace;
Getnames=yes;
Run;
```

When you specify Getnames=no system creates variable names as VAR1 VAR2 VAR3 ….. VARN and original variable from excel come as a first record.
In this case use Datarow=2 to read data properly.

```
Proc import
Datafile="C:\Documents and Settings\Administrator\Desktop\sample.xls"
Out=work.Sample dbms=excel replace;
Getnames=no;
Datarow=2;
Run;
```

To full the data from particular sheet in Excel

```
Proc import
Datafile="C:\Documents and Settings\Administrator\Desktop\sample.xls"
Out=work.Sample dbms=excel replace;
Getnames=no;
Datarow=2;
Sheet="STANSYS";
Run;
```

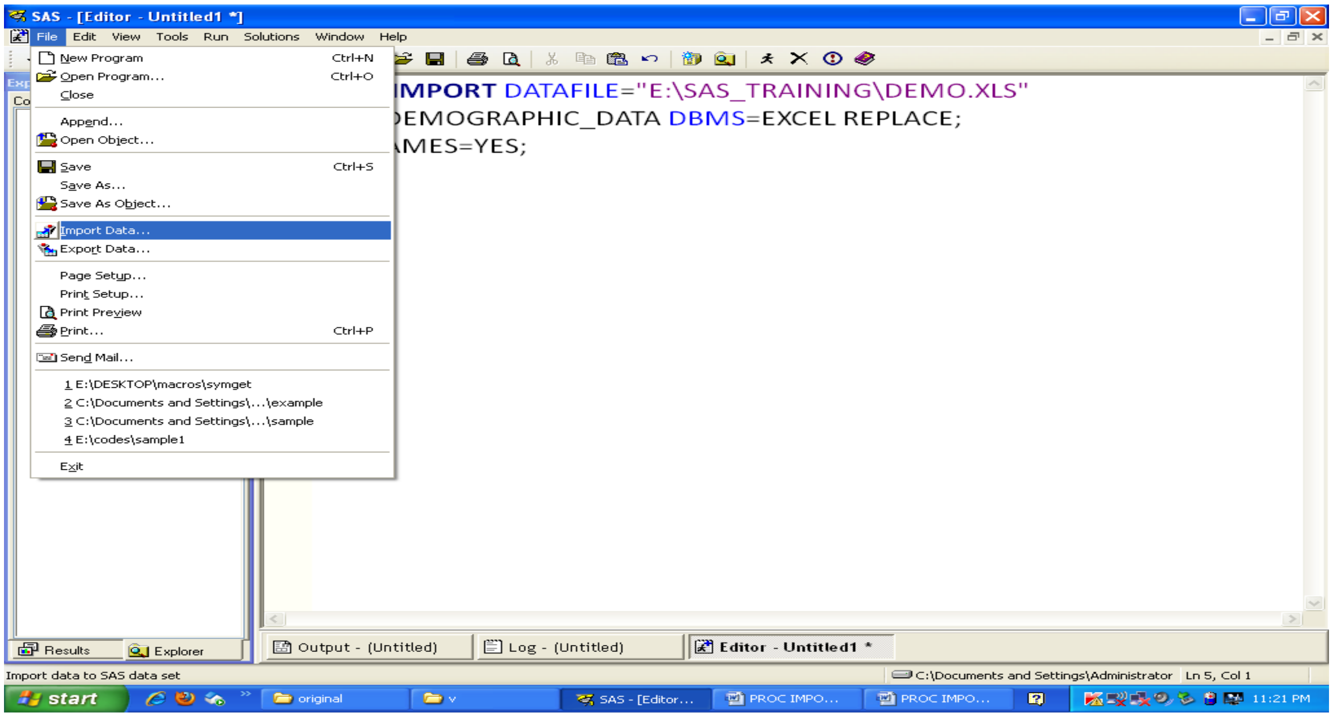
To full the data from MS-Access to SAS

```
Proc import table=employ
            out=work.sample
            dbms=access
            replace;
Database="D:\krishna\sas source\MS-ACCESS\data.mdb";
Run;
```

To import data, you can also use the Import Wizard

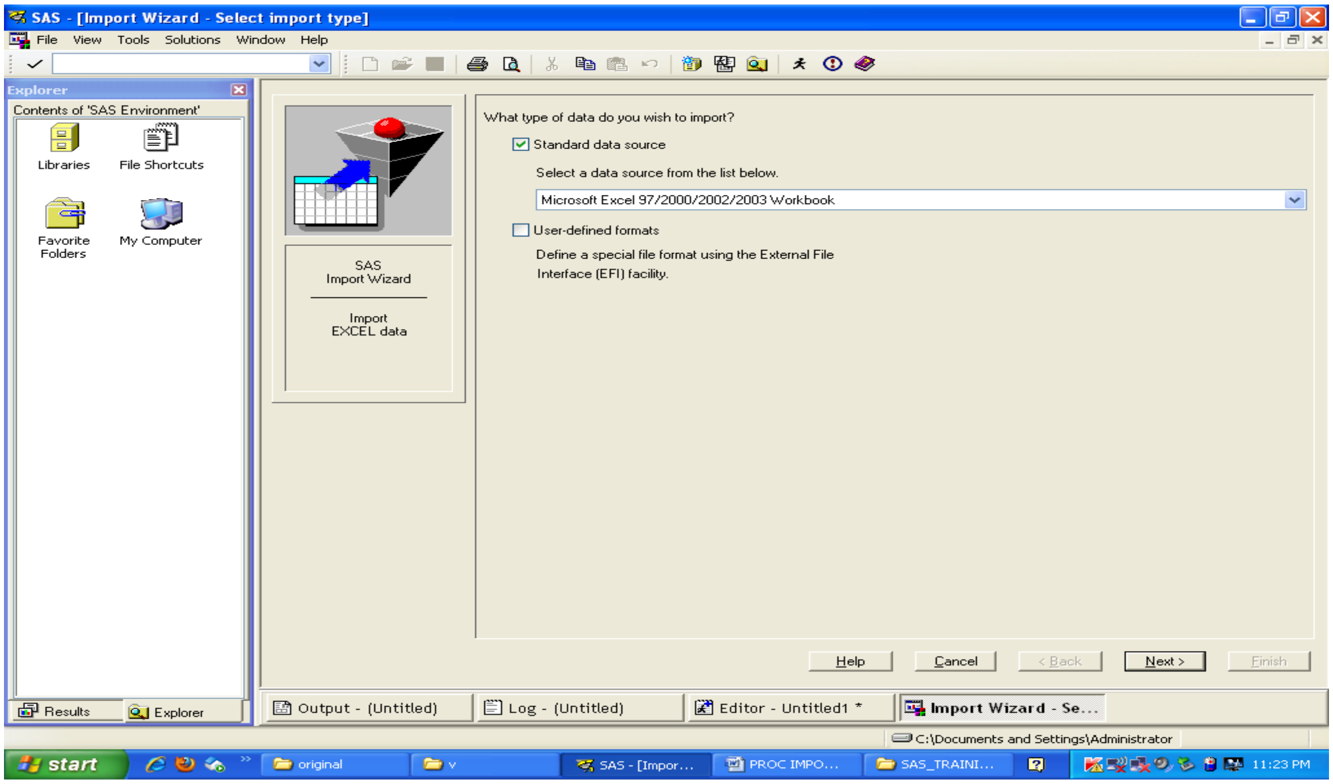
To import data from MS-Excel File to SAS thru import wizard

Click on File (Menu Bar) → Click Import data



```
IMPORT DATAFILE="E:\SAS_TRAINING\DEMO.XLS"
DEMOGRAPHIC_DATA DBMS=EXCEL REPLACE;
AMES=YES;
```

Select format of raw data like excel, csv, ms-access or notepad.



What type of data do you wish to import?

[✓] Standard data source

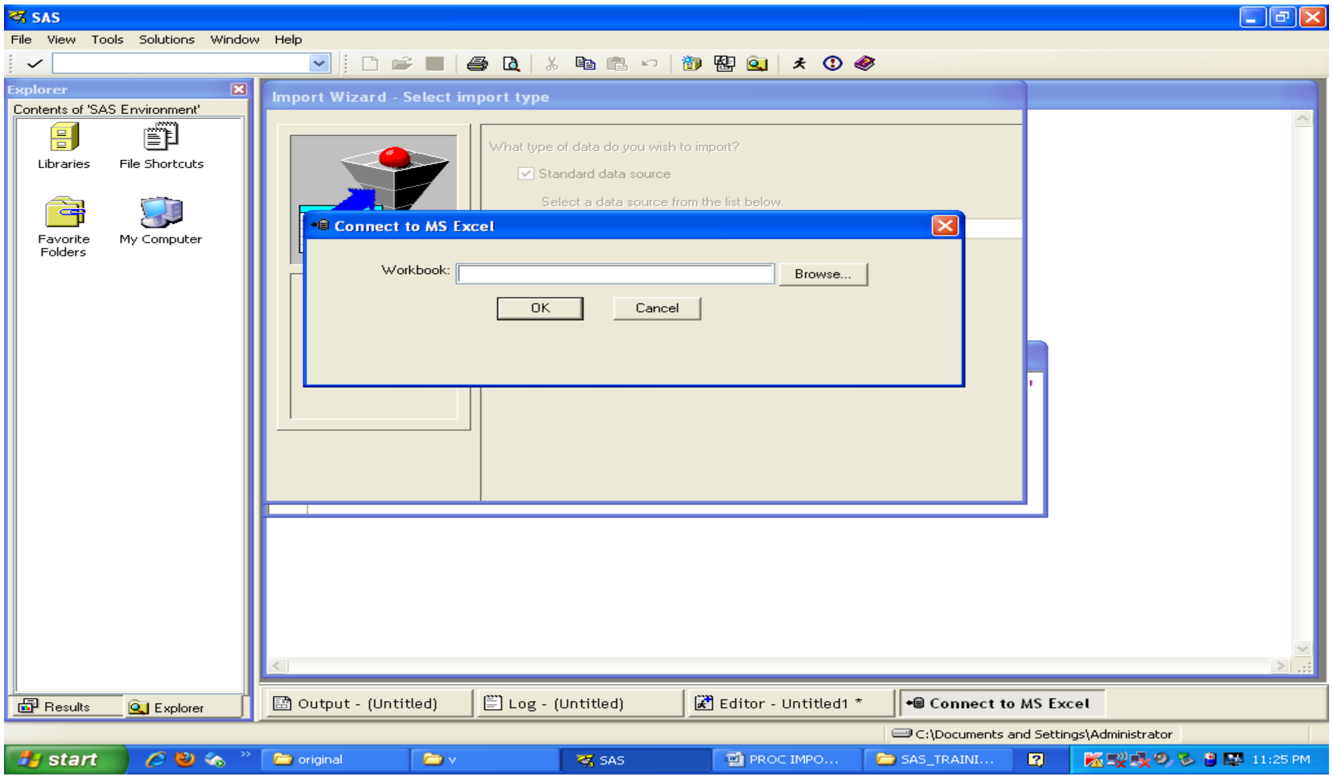
Select a data source from the list below.

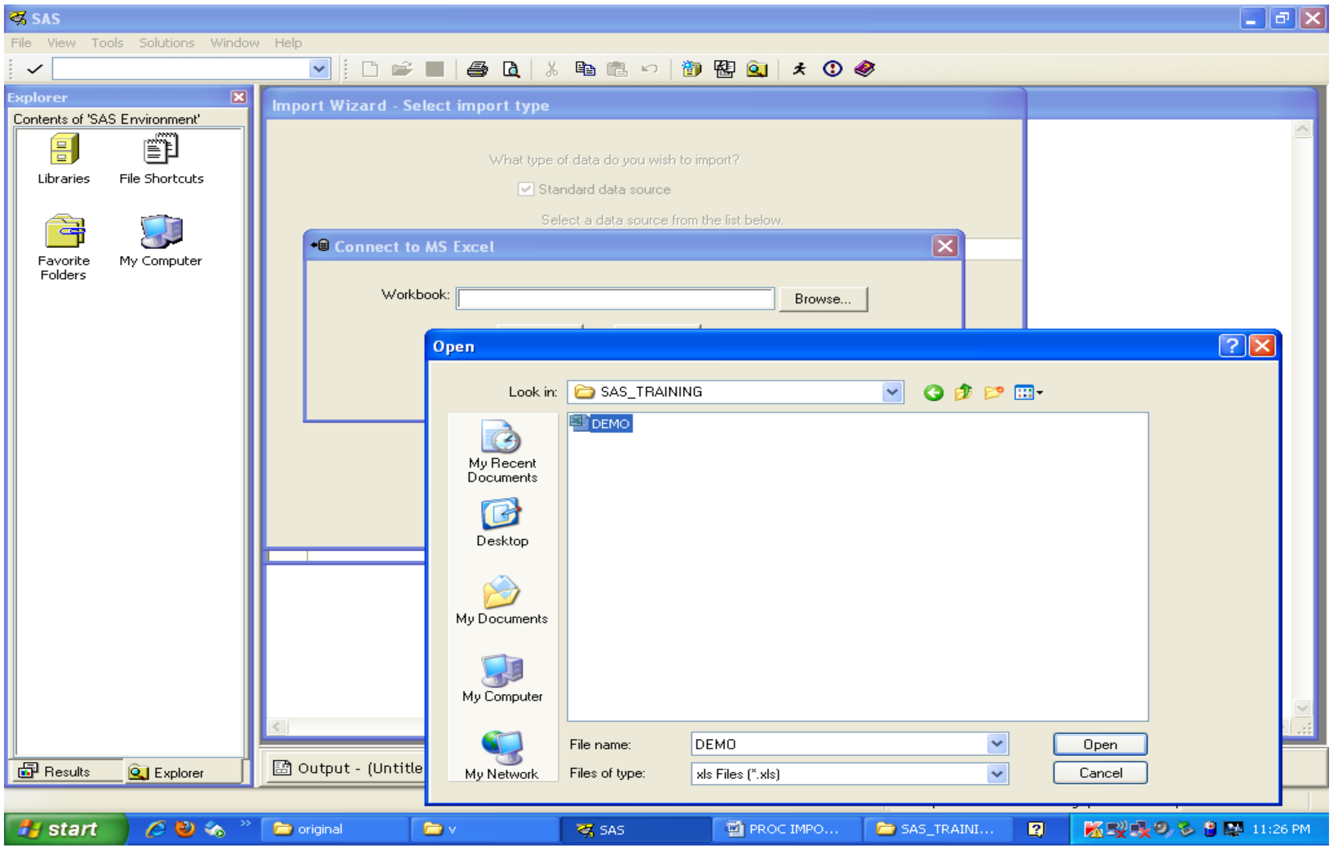
Microsoft Excel 97/2000/2002/2003 Workbook

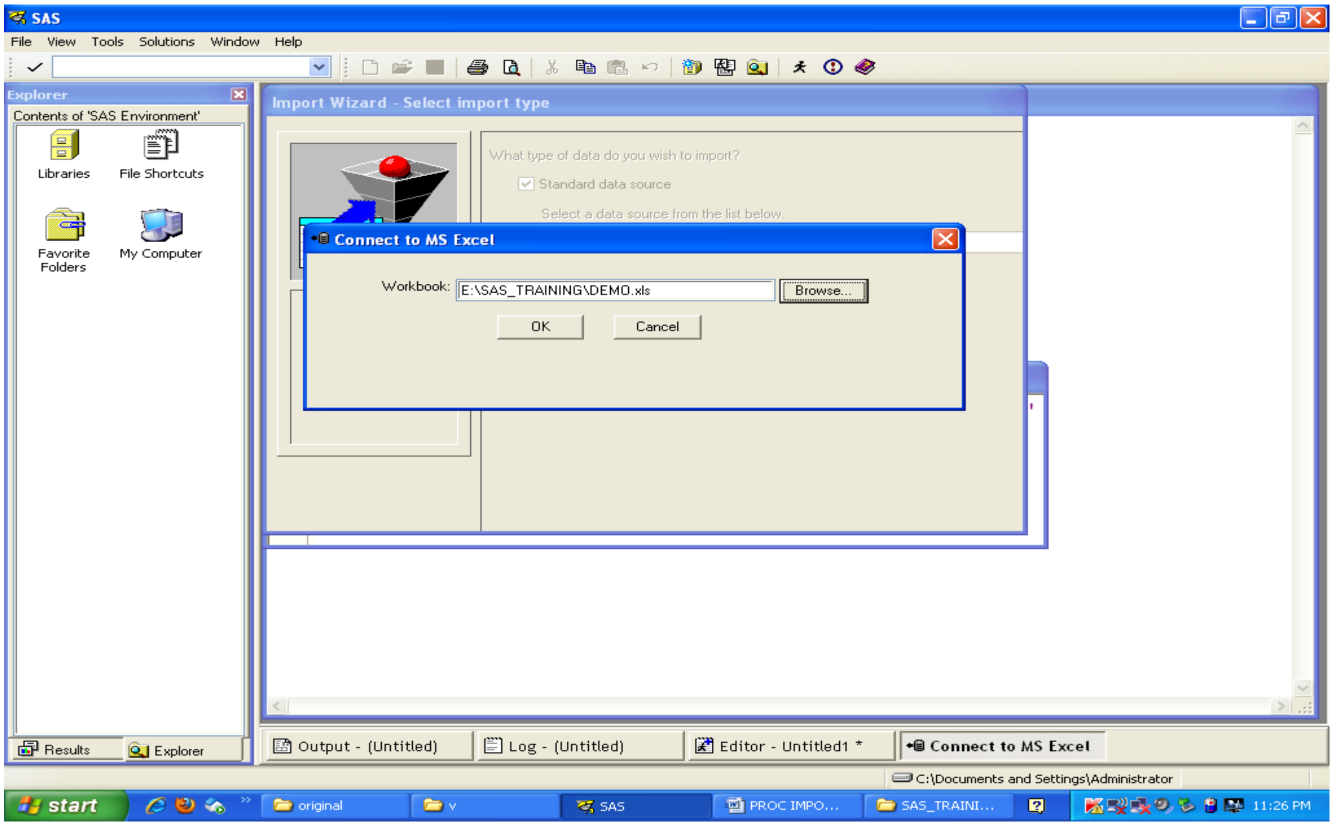
[ ] User-defined formats

Define a special file format using the External File Interface (EFI) facility.

Browse the location of raw data

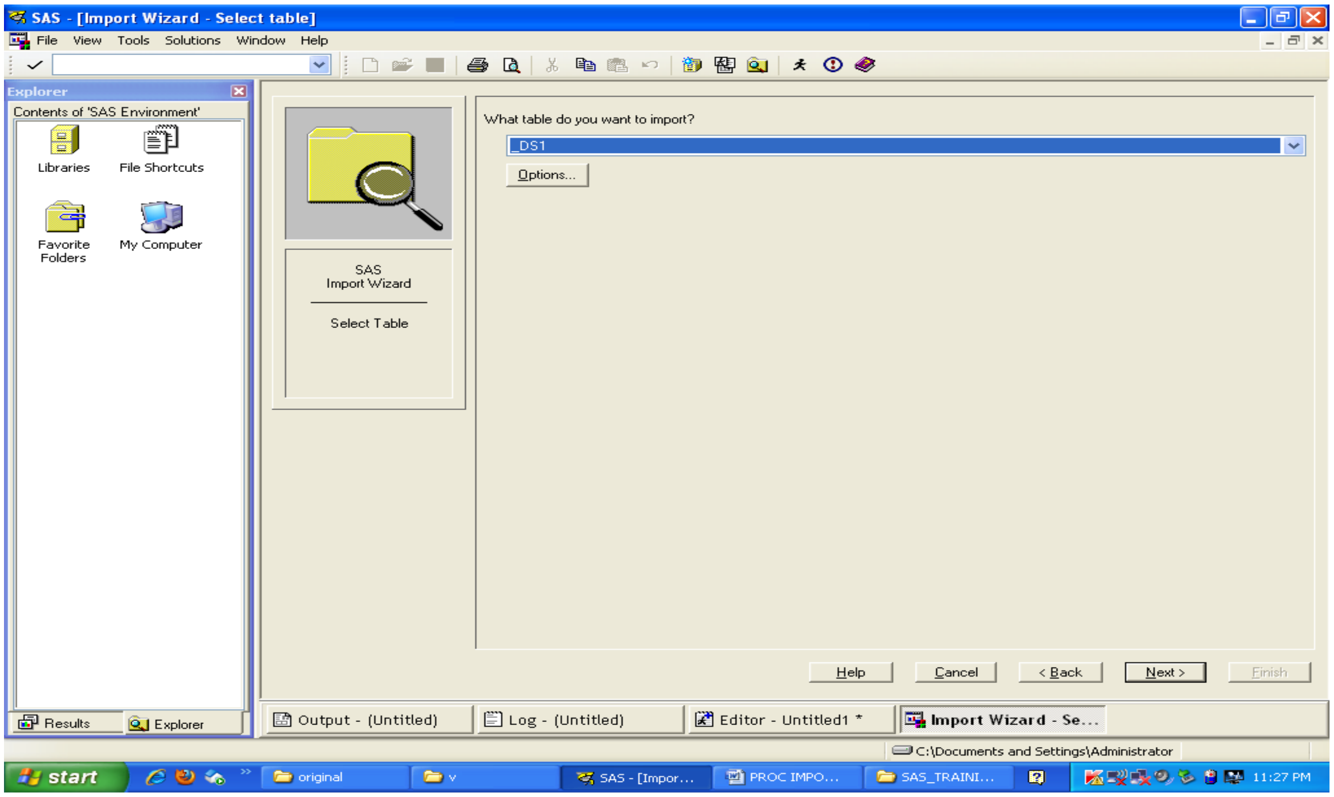





SAS
File View Tools Solutions Window Help
Explorer
Contents of 'SAS Environment'
Libraries
File Shortcuts
Favorite Folders
My Computer
Import Wizard - Select import type
What type of data do you wish to import?
Standard data source
Select a data source from the list below.
Connect to MS Excel
Browse...
OK
Cancel
Results
Explorer
Output - (Untitled)
Log - (Untitled)
Editor - Untitled1 *
Connect to MS Excel
C:\Documents and Settings\Administrator
start
11:26 PM

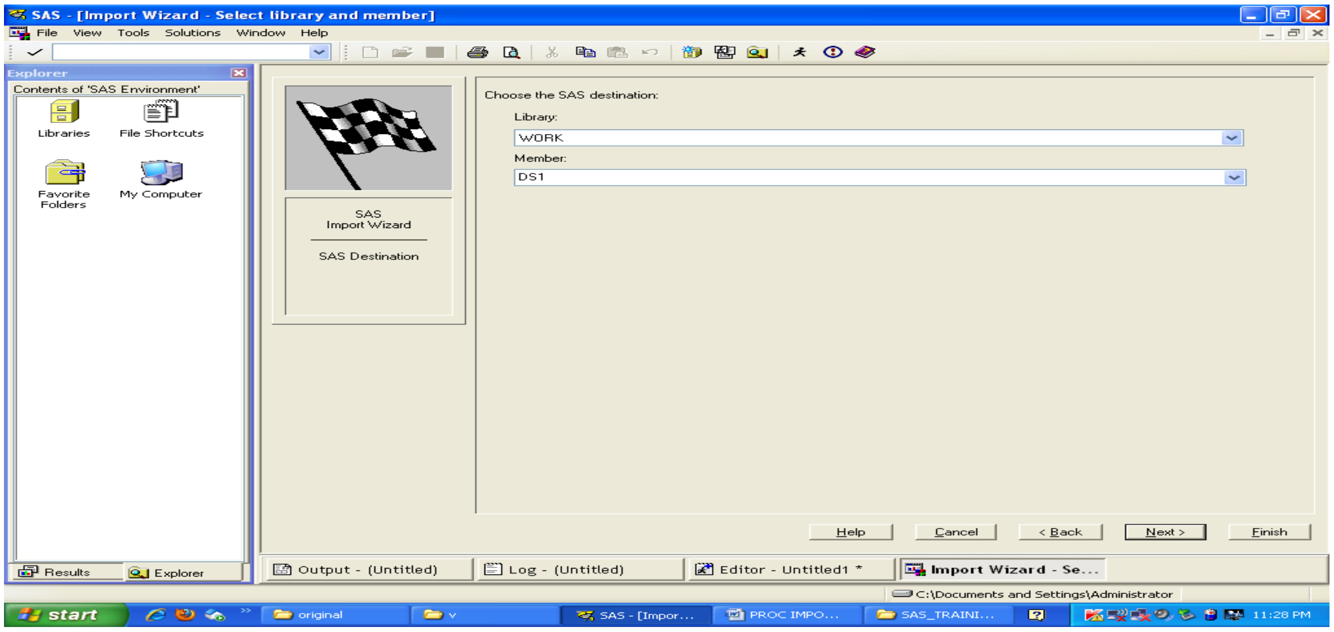
Workbook: E:\SAS_TRAINING\DEMO.xls

Click OK and specify sheet name for of excel sheet



What table do you want to import? _DS1

Specify library name and dataset name, where you are going to create SAS Dataset.



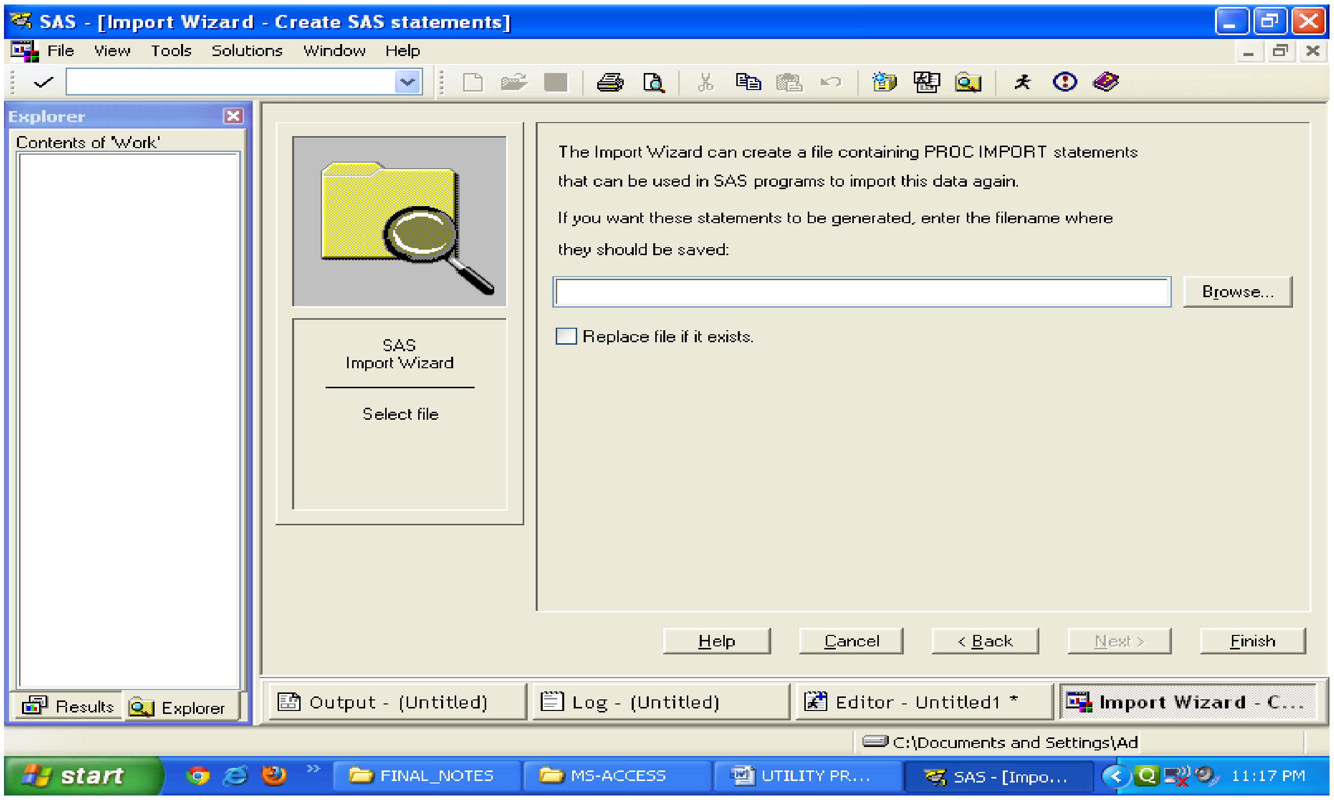
Choose the SAS destination:

Library: WORK

Member: DS1

And click finish

If you want get code which system written don't click finish click next and browse the Location where system should write program lines.



The Import Wizard can create a file containing PROC IMPORT statements that can be used in SAS programs to import this data again.

If you want these statements to be generated, enter the filename where they should be saved:

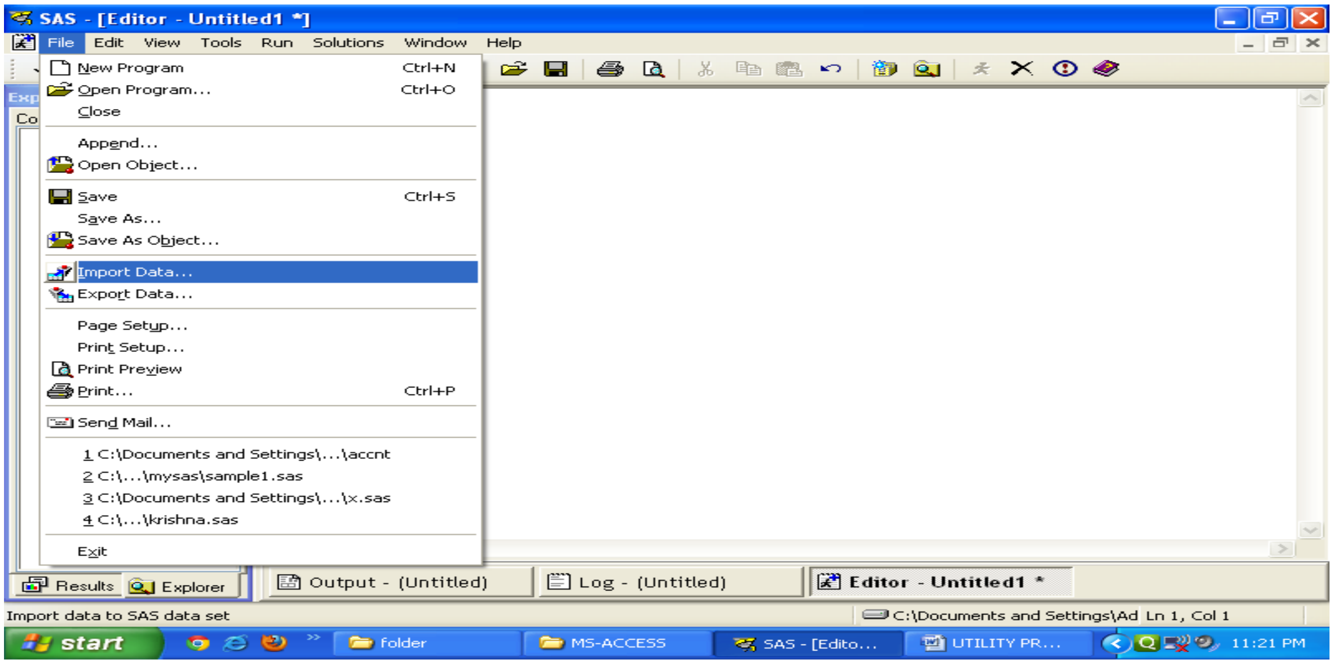
☐ Replace file if it exists.

Click Finish.

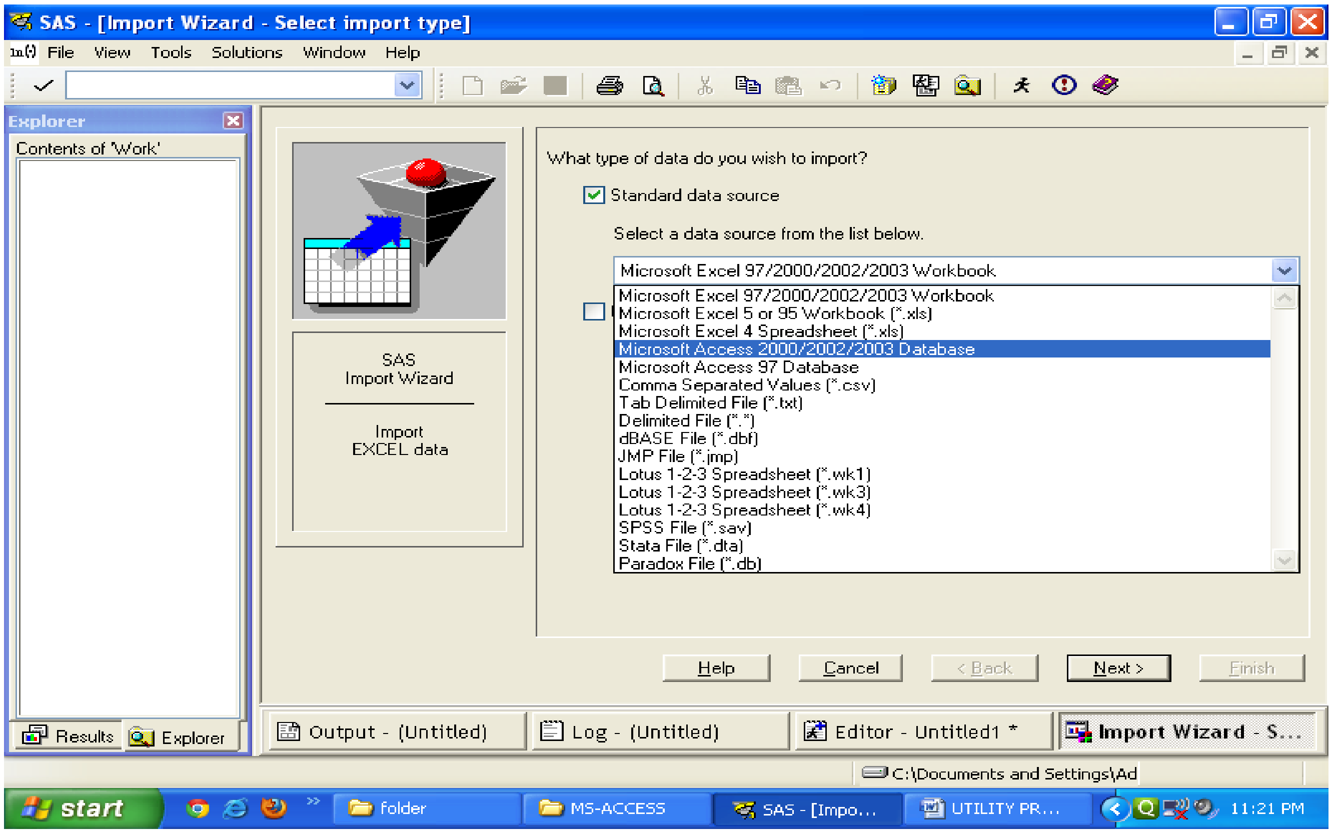
CSV files also we can import same way.

To import data from MS-ACCESS File to SAS thru import wizard

Click on File (Menu Bar) → Click Import data


SAS - [Editor - Untitled1 *]
File Edit View Tools Run Solutions Window Help
New Program Ctrl+N
Open Program... Ctrl+O
Close
Append...
Open Object...
Save Ctrl+S
Save As...
Save As Object...
Import Data...
Export Data...
Page Setup...
Print Setup...
Print Preview
Print... Ctrl+P
Send Mail...
1 C:\Documents and Settings\...\accnt
2 C:\...\mysas\sample1.sas
3 C:\Documents and Settings\...\x.sas
4 C:\...\krishna.sas
Exit
Results Explorer
Output - (Untitled) Log - (Untitled) Editor - Untitled1 *
Import data to SAS data set
C:\Documents and Settings\Ad Ln 1, Col 1

Select Microsoft access database


SAS - [Import Wizard - Select import type]
File View Tools Solutions Window Help
Explorer
Contents of 'Work'
SAS Import Wizard
Import EXCEL data
Help Cancel < Back Next > Finish
Results Explorer
Output - (Untitled) Log - (Untitled) Editor - Untitled1 * Import Wizard - S...

What type of data do you wish to import?

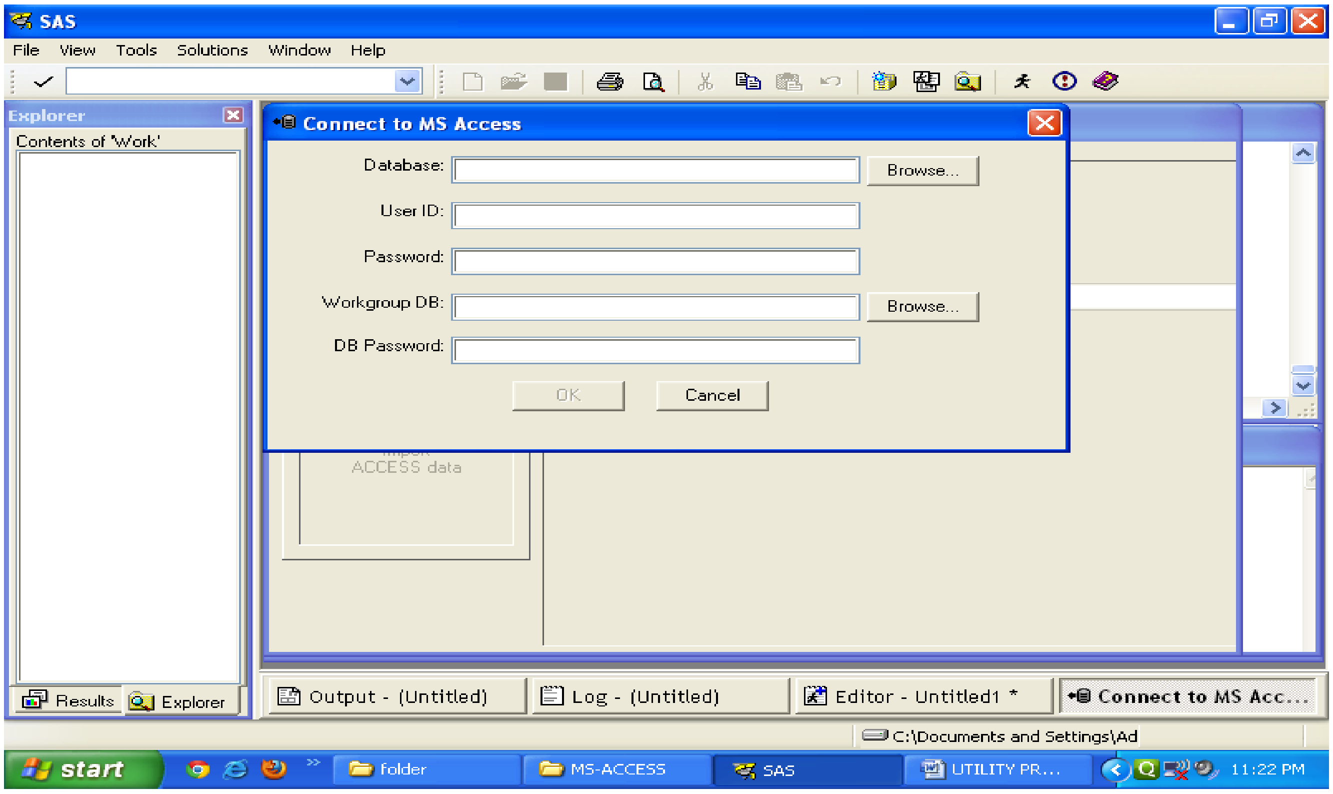
☑ Standard data source

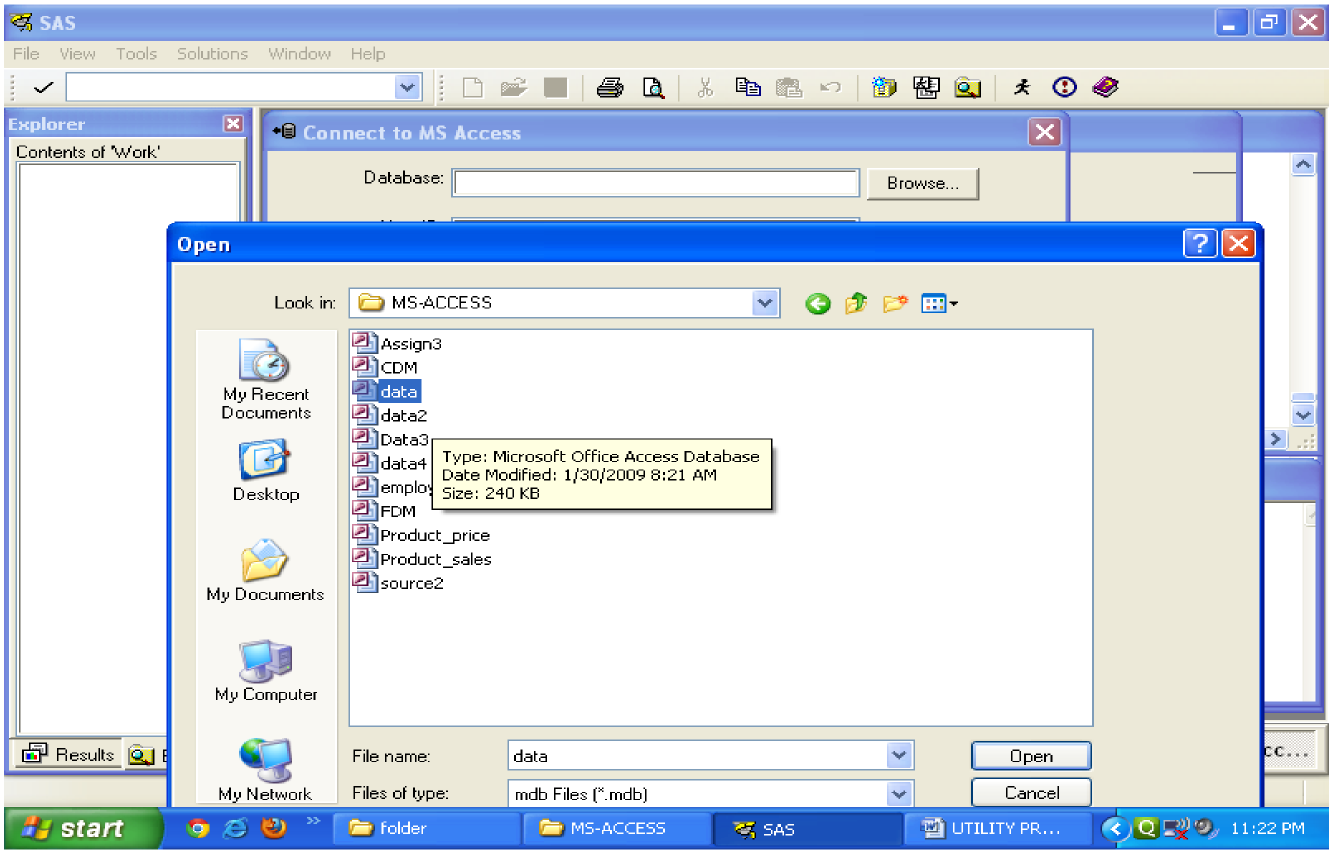
Select a data source from the list below.

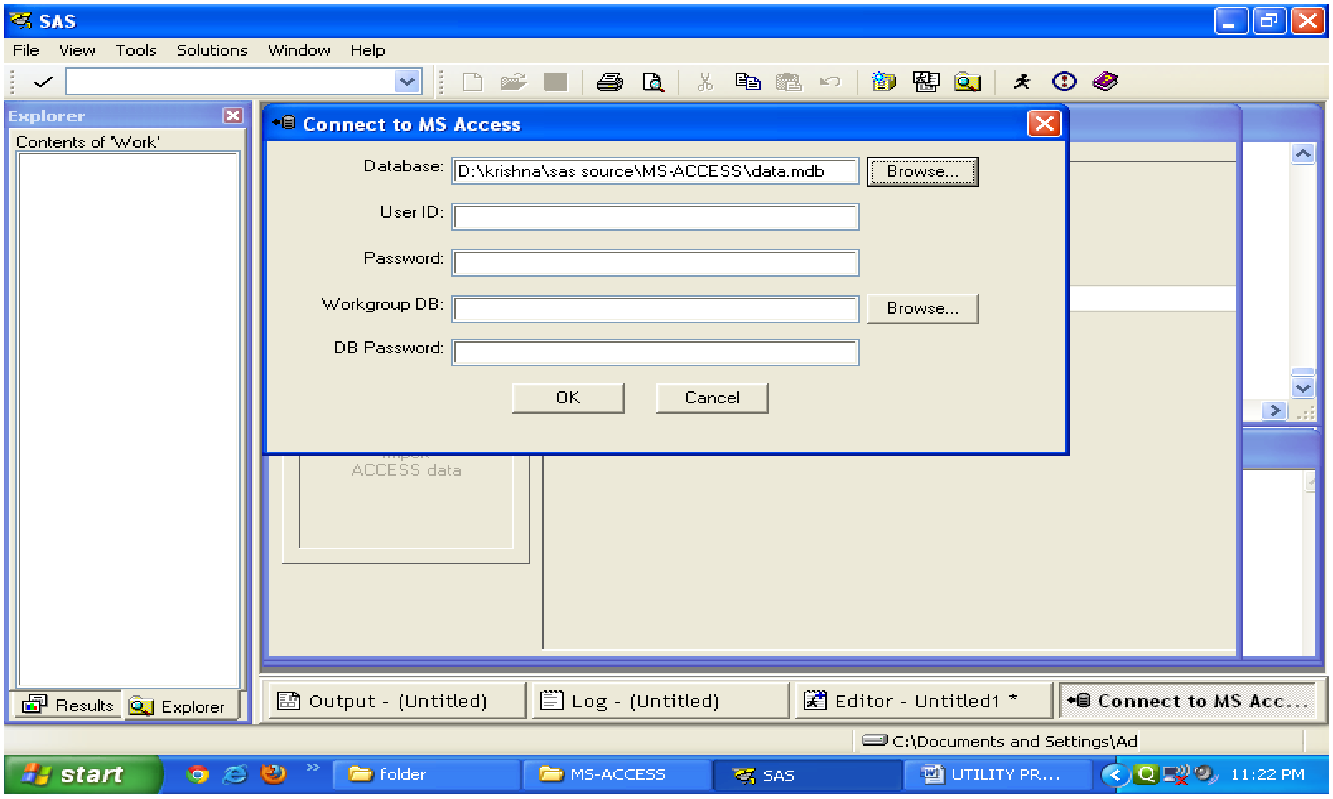
Microsoft Excel 97/2000/2002/2003 Workbook

- Microsoft Excel 97/2000/2002/2003 Workbook
- Microsoft Excel 5 or 95 Workbook (*.xls)
- Microsoft Excel 4 Spreadsheet (*.xls)
- Microsoft Access 2000/2002/2003 Database
- Microsoft Access 97 Database
- Comma Separated Values (*.csv)
- Tab Delimited File (*.txt)
- Delimited File (*.*)
- dBASE File (*.dbf)
- JMP File (*.jmp)
- Lotus 1-2-3 Spreadsheet (*.wk1)
- Lotus 1-2-3 Spreadsheet (*.wk3)
- Lotus 1-2-3 Spreadsheet (*.wk4)
- SPSS File (*.sav)
- Stata File (*.dta)
- Paradox File (*.db)

Browse access file location


SAS
File View Tools Solutions Window Help
Explorer
Contents of 'Work'
Connect to MS Access
Database:
Browse...
User ID:
Password:
Workgroup DB:
Browse...
DB Password:
OK
Cancel
ACCESS data
Results
Explorer
Output - (Untitled)
Log - (Untitled)
Editor - Untitled1 *
Connect to MS Acc...
C:\Documents and Settings\Ad
start
folder
MS-ACCESS
SAS
UTILITY PR...
11:22 PM


SAS
File View Tools Solutions Window Help
Explorer
Contents of 'Work'
Connect to MS Access
Database:
Browse...
Open
Look in: MS-ACCESS
Assign3
CDM
data
data2
Data3
data4
employ
FDM
Product_price
Product_sales
source2
Type: Microsoft Office Access Database
Date Modified: 1/30/2009 8:21 AM
Size: 240 KB
My Recent Documents
Desktop
My Documents
My Computer
My Network
File name: data
Files of type: mdb Files (*.mdb)
Open
Cancel
Results
start
folder
MS-ACCESS
SAS
UTILITY PR...
11:22 PM


Connect to MS Access

Database: D:\krishna\sas source\MS-ACCESS\data.mdb

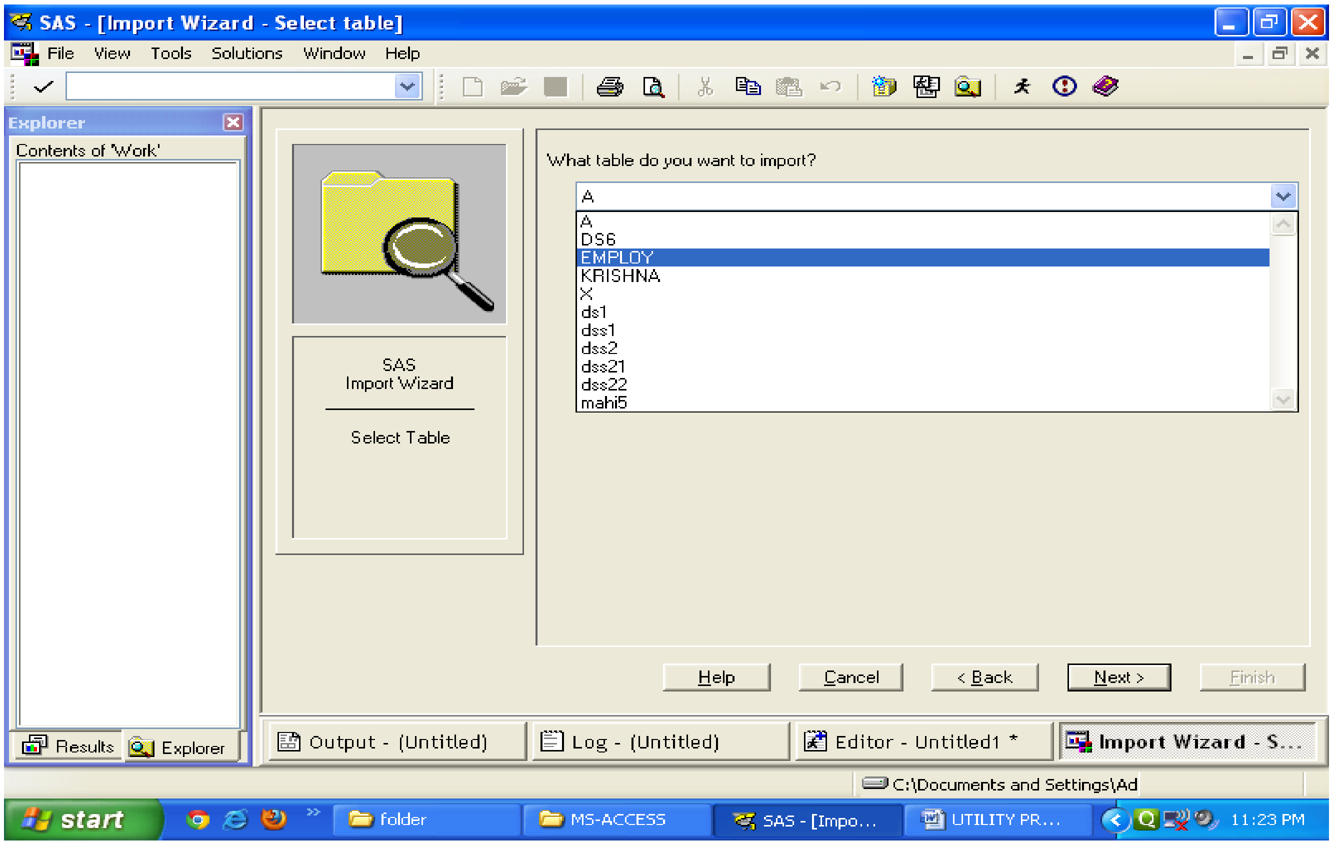
User ID:

Password:

Workgroup DB:

DB Password:

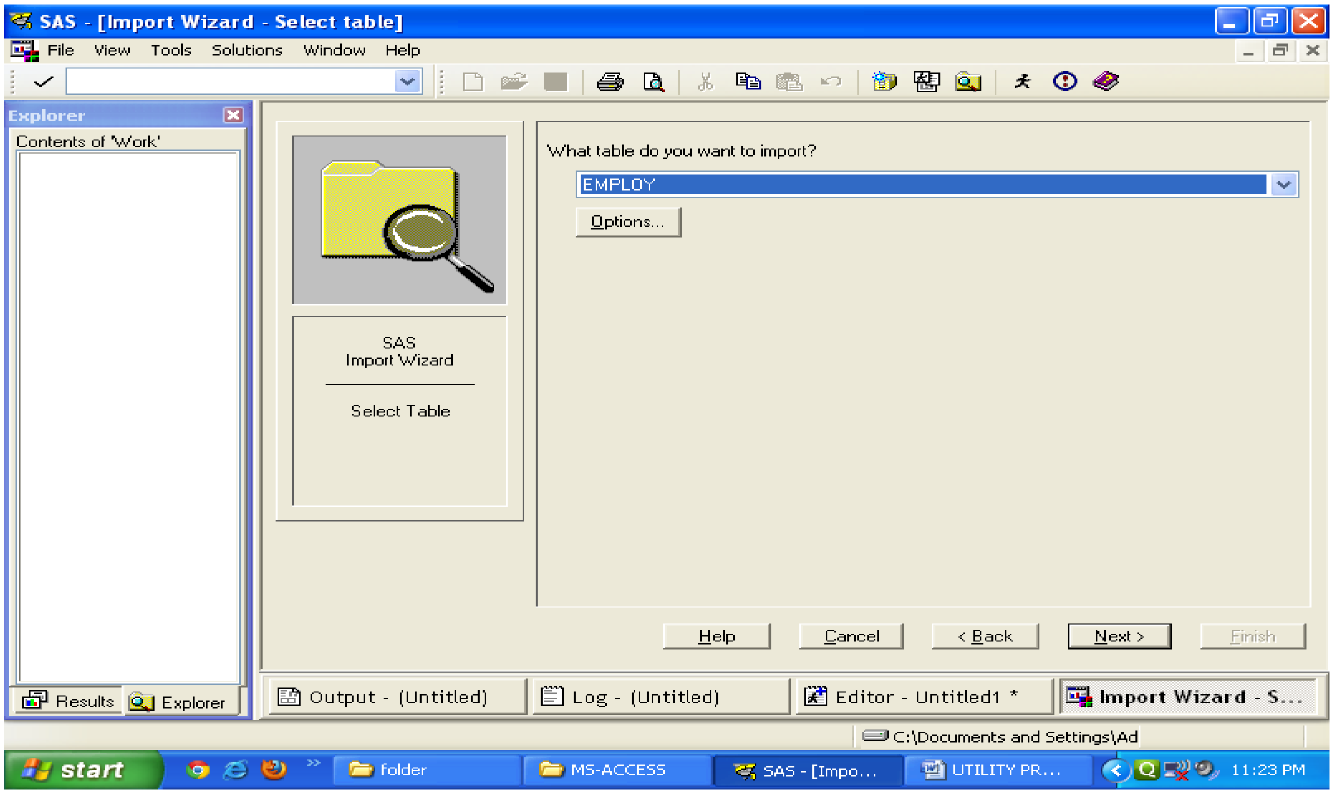
Specify the table from database which we are pulling to sas


SAS - [Import Wizard - Select table]

What table do you want to import?

A

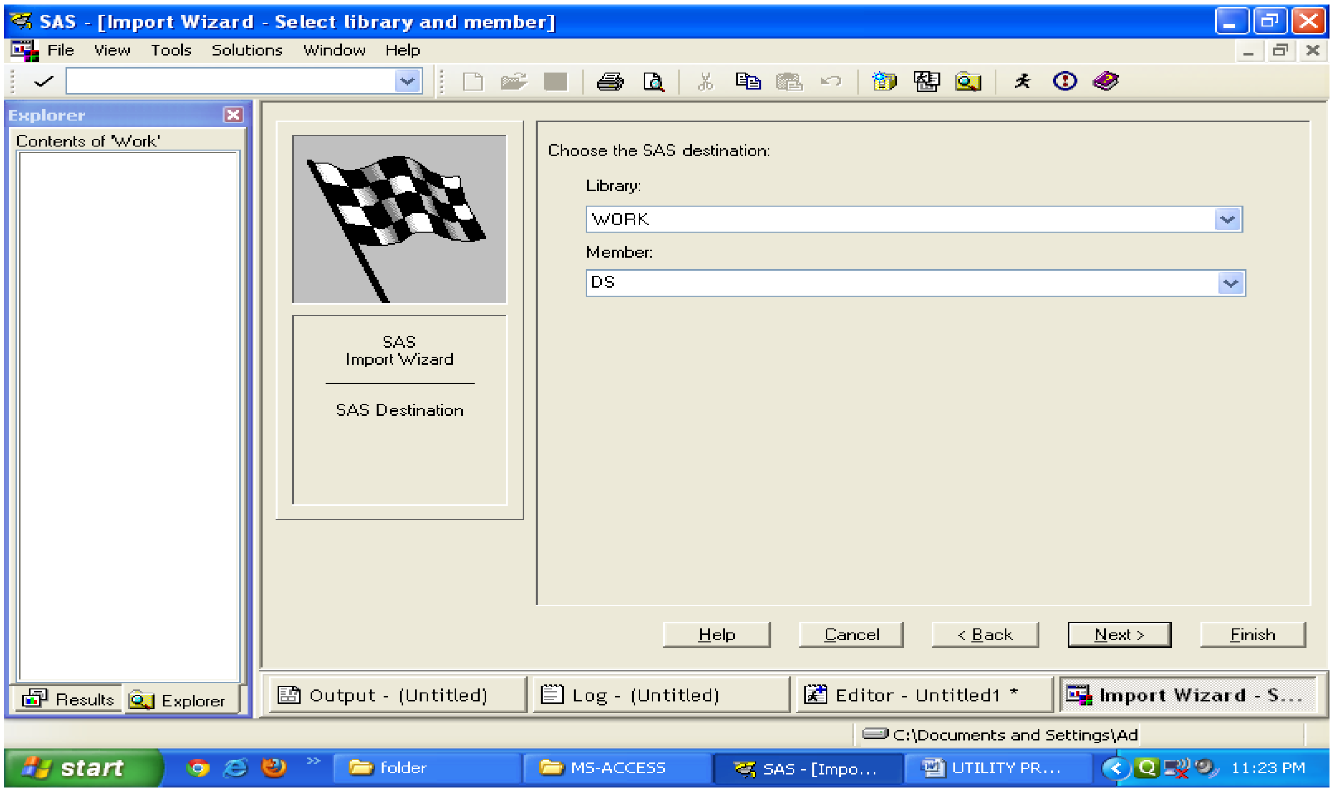
A
DS6
EMPLOY
KRISHNA
X
ds1
dss1
dss2
dss21
dss22
mahi5


SAS - [Import Wizard - Select table]

What table do you want to import?

EMPLOY

Specify library name and dataset name to where sas creates sas dataset.


SAS - [Import Wizard - Select library and member]

Choose the SAS destination:

Library: WORK

Member: DS

Specify the location where system writes program for the import process

When you run the IMPORT procedure, it reads the input file and writes the data to a SAS data set. The SAS variable definitions are based on the input records.

When the IMPORT procedure reads a delimited file, it generates a DATA step code to import the data.

```
Data WORK.Dataset;
      %let _EFIERR_ = 0; /* set the ERROR detection macro variable */
Infile 'C:\Documents and Settings\Administrator\Desktop\Demo.csv' delimiter = ','
                                             MISSOVER DSD lrecl=32767 firstobs=2;
      Informat EMP_ID $5. ;
      Informat EMP_NAME $10. ;
      Informat SEX $1. ;
      Informat AGE best32. ;
      Informat EDUCATION $2. ;
      Informat HEIGHT $7. ;
      Informat WEIGHT best32. ;
      Informat BLOOD_GROUP $4. ;
      Informat DOB DATE9. ;
      Informat DOJ DATE9. ;
      Informat ZONE $6. ;
      Informat ZONAL_HEAD $13. ;
      Informat REGION $11. ;
      Informat REGIONAL_MANAGER $22. ;
      Informat INCOME best32. ;
      Format EMP_ID $5. ;
      Format EMP_NAME $10. ;
      Format SEX $1. ;
      Format AGE best12. ;
      Format EDUCATION $2. ;
      Format HEIGHT $7. ;
      Format WEIGHT best12. ;
      Format BLOOD_GROUP $4. ;
      Format DOB DATE9. ;
      Format DOJ DATE9. ;
      Format ZONE $6. ;
      Format ZONAL_HEAD $13. ;
      Format REGION $11. ;
      Format REGIONAL_MANAGER $22. ;
      Format INCOME best12. ;
Input EMP_ID $
      EMP_NAME $
      SEX $
      AGE
      EDUCATION $
      HEIGHT $
      WEIGHT
      BLOOD_GROUP $
      DOB
      DOJ
      ZONE $
      ZONAL_HEAD $
      REGION $
      REGIONAL_MANAGER $
      INCOME
      ;
If _ERROR_ then call symputx('_EFIERR_',1); /*set ERROR detection macro variable*/
Run;
```

Demo.csv