

How to Read Delimited Text Files into SAS

This sample program shows you how to read a delimited text file into SAS. A text file is often referred as raw data that can be prepared in a variety of formats (e.g., csv [comma-separated values], tab-delimited, or space-delimited). When reading text files with the DATA step, two statements are used: INFILE and INPUT statements. The INFILE statement is used to specify the physical file being read. You can use a FILENAME statement in conjunction with an INFILE statement (see Example.1). Or, you can specify the full path to the file in the INFILE statement (see example.2). A set of options are available in the INFILE statement: DLM, DSD, and LRECL.

The DLM option allows you to tell SAS what character is used as the delimiter in the text file. If this option is not specified, SAS assumes the delimiter is a space. Common delimiters include commas, vertical pipes, semi-colons, and tabs.

The DSD option has three functions when reading delimited files. The first function is to remove any quotes that surround values in the text file. The second function deals with missing values. When SAS encounters consecutive delimiters in a file, the default action is to treat the delimiters as one unit. If a file has consecutive delimiters, it is usually because there are missing values between them. DSD tells SAS to treat consecutive delimiters separately; therefore, a value that is missing between consecutive delimiters will be read as a missing value when DSD is specified. The third function assumes the delimiter is a comma. If DSD is specified and the delimiter is a comma, the DLM option is not necessary.

The LRECL option should be used when the length of the records in a file is greater than 256 bytes. The input buffer is 256 bytes by default, and records that are greater than this length will be truncated when they are read.

(Source: SAS Technical Support DOC: TS-673).

[Example 1.]

```
FILENAME myfile "/full/path/testfile.csv" DLM=',';
LRECL=400;
```

```
DATA a;
INFILE myfile;
INPUT var1 var2 var3;
DATALINES;
...
```

[Example 2.]

```
DATA a;
INFILE "/full/path/testfile.csv" DLM=',' LRECL=400;
INPUT var1 var2 var3;
DATALINES;
...
```

EXAMPLE:

```
DATA a;
INFILE datalines DLM=',' DSD;
INPUT sex $ age educ;
DATALINES;
F, 40, 20
M, , 16
M, 35, 16
F, ,
, 50, 16
F, 45,
M, 50, 12
M, 42, 16
F, 36, 20
M, 38, 9
;
PROC PRINT;
```

Here are the results of PROC PRINT:

Obs	sex	age	educ
1	F	40	20
2	M	.	16
3	M	35	16
4	F	.	.
5		50	16
6	F	45	.
7	M	50	12
8	M	42	16
9	F	36	20
10	M	38	9