import sas — Import SAS files			

Description

import sas reads into memory a version 7 or higher SAS (.sas7bdat) file. It can also import SAS value labels from a .sas7bcat file. import sas can import up to 32,766 variables at one time (up to 2,048 variables in Stata/BE). If your SAS file contains more variables than this, you can break up the SAS file into multiple Stata datasets. You can also import SAS value labels from a .sas7bcat file.

Quick start

Import the SAS file myfile.sas7bdat into Stata

     import sas myfile

Same as above, but replace the data in memory

     import sas myfile, clear

Same as above, but import only variables x1 and x2

     import sas x1 x2 using myfile, clear

Import data from SAS file myfile and value labels from file labels.sas7bcat

     import sas myfile, bcat(labels)

Menu

File > Import > SAS data (*.sas7bdat)

Syntax

Load a SAS file (*.sas7bdat)

> import sas [using] *filename* [, *options*]

Load a subset of a SAS file (*.sas7bdat)

> import sas [*namelist*] [*if*] [*in*] using *filename* [, *options*]

If *filename* is specified without an extension, .sas7bdat is assumed. If *filename* contains embedded spaces, enclose it in double quotes.

namelist specifies SAS variable names to be imported.

options	Description
bcat(*filename*$_{vl}$)	load value labels defined in *filename*$_{vl}$ into memory
case(<u>lower</u> | <u>upper</u> | <u>preserve</u>)	read variable names as lowercase or uppercase; the default is to preserve the case
clear	replace data in memory
encoding("*encoding*")	specify the file encoding; see help encodings

collect is allowed; see [U] **11.1.10 Prefix commands**.
encoding() does not appear in the dialog box.

Options

bcat(*filename*$_{vl}$) specifies that the value labels defined in *filename*$_{vl}$ be loaded into memory along with the dataset. If *filename*$_{vl}$ is specified without an extension, .sas7bcat is assumed. If *filename*$_{vl}$ contains embedded spaces, enclose it in double quotes.

> SAS does not assign value labels to variables; therefore, you must use the label values command to assign the value labels to specific variables after importing them.

case(lower \| upper \| preserve) specifies the case of the variable names after import. The default is case(preserve).

clear specifies that it is okay to replace the data in memory, even though the current data have not been saved to disk.

The following option is available with import sas but is not shown in the dialog box:

encoding("*encoding*") specifies the encoding of the file. If your file has an incorrect encoding specified in the file header, you can use this option to specify the correct encoding. See help encodings for details.

Remarks and examples

import sas reads into memory version 7 or higher SAS (.sas7bdat) files. If a SAS variable name from the file does not conform to a Stata variable name, a generic v# name will be assigned, and the original variable name will be stored as a characteristic for the variable. If a SAS variable label is too long, it will be truncated to 80 characters. The original variable label will be stored as a variable characteristic. If a SAS data label is too long, it will be truncated to 80 characters, and the original label will be stored as a data characteristic.

▷ Example 1: Importing a SAS file into Stata

We can import SAS files into Stata, either by selecting the entire file or by selecting subsets of the data, with `import sas`. For example, we have the SAS file `auto.sas7bdat`, which contains data on automobiles, and we have value labels for these data stored in `formats.sas7bcat`. Below, we demonstrate how to import these data into Stata. To follow along, download these files to your working directory by typing the `copy` commands below:

```
. copy https://www.stata.com/sampledata/auto.sas7bdat auto.sas7bdat
. copy https://www.stata.com/sampledata/formats.sas7bcat formats.sas7bcat
```

To load the file `auto.sas7bdat` into Stata's memory, we type

```
. import sas auto.sas7bdat
(12 vars, 74 obs)
```

We can instead import only the variables `make`, `weight`, and `foreign` from `auto.sas7bdat`. We use the `bcat()` option to add the value labels defined in the `formats.sas7bcat` file and the `clear` option to replace the data in memory without saving them.

```
. import sas make weight foreign using auto, bcat(formats) clear
(3 vars, 74 obs)
. list in 1/5
```

	make	weight	foreign
1.	AMC Concord	2930	0
2.	AMC Pacer	3350	0
3.	AMC Spirit	2640	0
4.	Buick Century	3250	0
5.	Buick Electra	4080	0

We list the value labels that we imported using `label list`

```
. label list
ORIGIN:
           0 Domestic
           1 Foreign
```

`ORIGIN` contains value labels for the variable `foreign`. We need to use the `label values` command to apply this label to `foreign`. Then, we `save` the data with these labels attached.

```
. label values foreign ORIGIN
. list in 1/5
```

	make	weight	foreign
1.	AMC Concord	2930	Domestic
2.	AMC Pacer	3350	Domestic
3.	AMC Spirit	2640	Domestic
4.	Buick Century	3250	Domestic
5.	Buick Electra	4080	Domestic

```
. save myauto
file myauto.dta saved
```

◁

Stored results

import sas stores the following in r():

Scalars
  r(N)   number of observations imported
  r(k)   number of variables imported

Also see

[D] **import sasxport5** — Import and export data in SAS XPORT Version 5 format

[D] **import sasxport8** — Import and export data in SAS XPORT Version 8 format

[D] **import** — Overview of importing data into Stata