

# GET SAS

---

```
GET SAS DATA=file [DSET(data set)]  
    [/FORMATS=file]
```

## Example

```
GET SAS DATA='ELECT' .
```

## Overview

GET SAS builds an SPSS-format working data file from a SAS data set or a SAS transport file. A SAS transport file is a sequential file written in SAS transport format and can be created by the SAS export engine available in SAS Release 6.06 or higher or by the EXPORT option on the COPY or XCOPY procedure in earlier versions. GET SAS reads SAS files up to version 8.2.

## Options

**Retrieving User-defined value labels.** For native SAS data sets, you can specify a file on the FORMATS subcommand to retrieve user-defined value labels associated with the data being read. This file must be created by the SAS PROC FORMAT statement, and can only be used for native SAS data sets. For SAS transport files, the FORMATS subcommand is ignored.

**Specifying the Data Set.** You can name a data set contained in a specified SAS file, using DSET on the DATA subcommand. GET SAS reads the specified data set from the SAS file.

## Basic Specification

The basic specification is the DATA subcommand followed by the name of the SAS file to read. By default, the first SAS data set is copied into the working data file and any necessary data conversions are made.

## Syntax Rules

- The subcommand DATA and the SAS file name are required and must be specified first.
- The subcommand FORMATS is optional. This subcommand is ignored for SAS transport files.
- GET SAS does not allow KEEP, DROP, RENAME, and MAP subcommands. To use a subset of the variables, rename them, or display the file content, you can specify the appropriate commands after the SPSS working data file is created.

## Operations

- GET SAS reads data from the specified or default data set contained in the SAS file named on the DATA subcommand.
- Value labels retrieved from a SAS user-defined format are used for variables associated with that format, becoming part of the SPSS dictionary.
- All variables from the SAS data set are included in the working data file, and they are in the same order as in the SAS data set.

## DATA Subcommand

DATA specifies the file that contains the SAS data set to be read.

- DATA is required and must be the first specification on GET SAS.
- The file specification varies from operating system to operating system. Enclosing the filename within apostrophes always works.
- The optional DSET keyword on DATA determines which data set within the specified SAS file is to be read. The default is the first data set.

**DSET (data set)** *Data set to be read.* Specify the name of the data set in parentheses. If the specified data set does not exist in the SAS file, GET SAS displays a message informing you that the data set was not found.

### Example

```
GET SAS DATA='ELECT' DSET(Y1948) .
```

- The SAS file *ELECT* is opened and the data set named *Y1948* is used to build the working file for the SPSS session.

## FORMATS Subcommand

FORMATS specifies the file containing user-defined value labels to be applied to the retrieved data.

- The file specification varies from operating system to operating system. Enclosing the filename within apostrophes always works.
- If FORMATS is omitted, no value labels are available.
- Value labels are only applied to numeric integer values. They are not applied to non-integer numeric values or string variables.
- The file specified on the FORMATS subcommand must be created with the SAS PROC FORMAT statement.
- For SAS transport files, the FORMATS subcommand is ignored.

### Example

```
GET SAS /DATA='ELECT' DSET(Y1948)
  /FORMATS='ELECTFM' .
```

- Value labels read from the SAS file *ELECTFM* are converted to conform to SPSS conventions.

### Creating a Formats File with PROC FORMAT

To create a file containing SAS value labels, run the following program in SAS:

```
libname mylib 'path';  
proc format library = mylib  
  cntlout = mylib.sas_fmfts;  
run;
```

where 'path' is the directory that contains your input data file.

This procedure creates a SAS file in the directory 'path' that has the format information for each SAS data file. In this case, the file will have the name *SAS\_FMFTS.SD2* and be found in the same directory as the input SAS data file.

### SAS to SPSS Data Conversion

Although SAS and SPSS data files have similar attributes, they are not identical. SPSS makes the following conversions to force SAS data sets to comply with SPSS conventions.

#### Variable Names

SAS variable names that do not conform to SPSS variable name rules are converted to valid 8-character SPSS variable names.

#### Variable Labels

SAS variable labels specified on the LABEL statement in the DATA step are used as variable labels in SPSS.

#### Value Labels

SAS value formats that assign value labels are read from the data set specified on the FORMATS subcommand. The SAS value labels are then converted to SPSS value labels in the following manner:

- Labels assigned to single values are retained.
- Labels assigned to a range of values are ignored.
- Labels assigned to SAS keywords LOW, HIGH, and OTHER are ignored.
- Labels assigned to string variables and non-integer numeric values are ignored.
- Labels over 60 characters long are truncated.

### Missing Values

Since SAS has no user-defined missing values, all SAS missing codes are converted to SPSS system-missing values.

### Variable Types

- Both SAS and SPSS allow two basic types of variables: numeric and character string. During conversion, SAS numeric variables become SPSS numeric variables, and SAS string variables become SPSS string variables of the same length.
- Date, time, and datetime SAS variables are converted to equivalent SPSS date, time and datetime variables. All other numeric formats are converted to the default SPSS numeric format.