

## Introduction to SAS/FSP<sup>®</sup> in Version 8

Terry Fain, RAND, Santa Monica, California  
 Cyndie Gareleck, RAND, Santa Monica, California

### ABSTRACT

SAS/FSP is a set of procedures used to perform full-screen interactive data entry, data editing, data retrieval, and data query. This tutorial provides a basic overview of SAS/FSP software under version 8 of the SAS<sup>®</sup> System<sup>1</sup>. Topics include:

- creating a new SAS data set
- customizing data screens
- data entry
- data query
- browsing SAS data sets

The focus of this tutorial is on PROC FSEDIT although other procedures such as PROC FSBROWSE and PROC FSVIEW will be briefly discussed. SAS Component Language (SCL) is considered an advanced topic and will not be covered in this tutorial.

### INTRODUCTION TO SAS/FSP

SAS/FSP contains five procedures: FSEDIT, FSBROWSE, FSLIST, FSVIEW, and FSLETTER. This tutorial will concentrate on the first four procedures, with emphasis on FSEDIT. A description of the five procedures follows.

1. **FSEDIT**: Used for creating and editing SAS data sets and data views created with SAS/ACCESS<sup>2</sup> as well as for data entry. May also be used for data query. Data is displayed by observation.
2. **FSBROWSE**: Similar to FSEDIT without the data editing features (data set is read-only). May also be used to browse data views created with SAS/ACCESS and PROC SQL.
3. **FSVIEW**: Similar to FSEDIT but data are displayed as a table with rows and columns instead of by observation.
4. **FSLIST**: Used to view external files.
5. **FSLETTER**: Used to create letters and other reports.

### INTRODUCTION TO PROC FSEDIT

PROC FSEDIT is one of the most powerful procedures contained in SAS/FSP software. It is an interactive procedure used to create new SAS data sets, edit existing SAS data sets, enter data, and for ad hoc queries.

#### Syntax

FSEDIT is invoked from the Program Editor Window<sup>3</sup> by a PROC statement which ends with a semicolon.<sup>4</sup> VAR and LABEL statements may also be used, but are optional.

```
PROC FSEDIT [options];
VAR variables;
LABEL variable = 'label'... ;
FORMAT... ;
INFORMAT... ;
WHERE5... ;
```

**PROC Statement.** The PROC statement initiates the FSEDIT procedure. Frequently used options include:

1. **NEW=SASdataset.** Creates a new SAS data set. Optionally, can be used with **LIKE=SASdataset**, which copies an existing data structure when creating a new SAS data set.
2. **DATA=SASdataset.** Names an existing SAS data set for editing.
3. **SCREEN=libref.catalog.** Names the SAS catalog where customized screen information will be stored. If this option is omitted, a customized screen will not be saved.
4. **Keys=keys-entry.** Contains function key assignments. Ignored when the SCREEN= option is specified.
5. **LABEL.** Specifies that variable labels be used to identify variables. Ignored if SCREEN= specified.
6. **ADD/NOADD.** **ADD** adds a new observation, **NOADD** prevents adding new observations.
7. **NODEL.** Prevents deletion of observations.
8. **MOD.** Opens FSEDIT screen modification menu window.
9. **OBS=n.** Specifies the number of observations. Not valid if a WHERE statement is specified.
10. **PRINTALL.** Prints the data set by observation.
11. **NC=n.** Specifies the display column width of FSEDIT window. Not valid if SCREEN= specified.
12. **NR=n.** Specifies row height of FSEDIT window. Not valid if SCREEN= specified.
13. **TAB=n.** Specifies tab interval. Not valid if SCREEN= specified.
14. **DEBUG.** Turns on the SCL source-level debugger, which provides step-by-step assistance in resolving errors in SCL programs.
15. **PRINTFILE=fileref.** Specifies the fileref of an external file to which letters are to be written by the FSEDIT procedure.
16. **LETTER=SAS-catalog** or **LETTER=libref.catalog.letter-name.LETTER.** Names a SAS catalog that contains letters, or a specific letter within a SAS catalog.
17. **SEND=letter-name.** Sends the specified letter for the LETTER= catalog for each observation in your DATA= data set. FSEDIT is not actually launched by this option.

Within SAS there are multiple ways to create a new data set. You can use a SAS data step, PROC FSVIEW, or you can use PROC FSEDIT to create a new data set.

### CREATING A NEW SAS DATA SET WITH PROC FSEDIT

To create a new data set, submit statements such as the following from the Program Editor window.

```

Program Editor - (Untitled)
libname new ' ';
proc fseedit new=new.demo1 screen=new.demo1scr;
run;
    
```

Figure 1—Invoking PROC FSEDIT

This code will invoke PROC FSEDIT in data definition mode as shown in Figure 2. Note that the procedure and file name are indicated at the top of the window. Also note that the display may vary depending on the window manager being used. The screens shown in this paper come from Microsoft® Windows NT® Workstation 4.0.

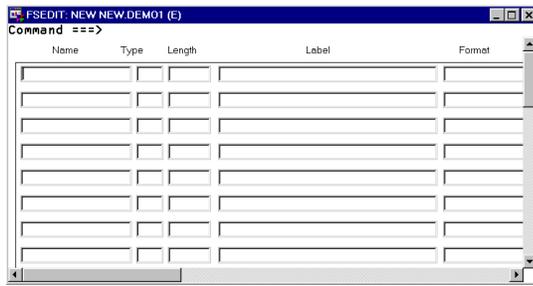


Figure 2—Data Definition Mode

You create a data set by entering definition information as shown in Figure 3 below. Note the following rules for defining variables:

- Variable names must follow SAS naming conventions.
- Character variables must be assigned “type” c or \$.
- Each variable should be assigned a length.<sup>6</sup> Eight is the default.
- You may assign a label, format, or informat. To view the Informat column, from the View menu, select Informat.

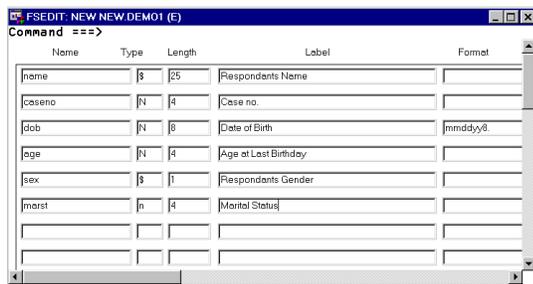


Figure 3—Data Definition Mode

After you are finished entering the information, type End on the Command line, or press the End function key to execute. A data entry screen will appear similar to the one below.

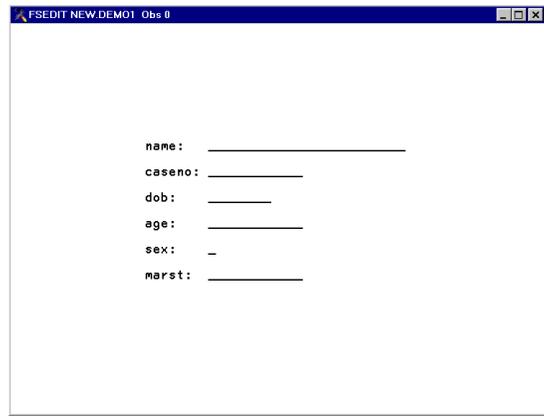


Figure 4—Default Screen

### Screen Customization

You may customize your screen display to, for example, resemble a survey instrument. Modifications to the screen will be stored in the screen catalog. In our example, the catalog will be saved as DEMO1SCR.SAS7BCAT in the library referenced by the LIBNAME statement.

To enter Screen Modification mode, from the Tools menu, select Modify Screen... A dialog box like the one shown in Figure 5 will appear to allow you to assign a password.

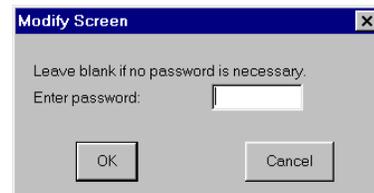


Figure 5—Password Dialog Box

Enter a password if you wish. Click OK to close the dialog box. A screen like the one in Figure 6 will appear. If you have not specified a libref to which your screen will be saved, you will also see the message:

WARNING: Screen modifications will not be saved.

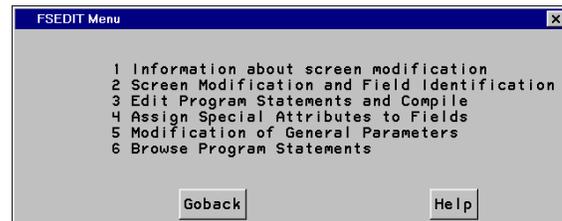


Figure 6—Screen Modification Mode

We will concentrate primarily on Options 2 (Screen Modification and Field Identification), 4 (Assign Special Attributes to Fields), and 5 (Modification of General Parameters), because that's where most of the customization features (with the exception of SCL) are to be found.

### Option 2: Screen Modification and Field Identification

Click Option 2, Screen Modification and Field Identification, to select it. A screen similar to the default data entry screen will appear.

Use the mouse to select, cut, and paste as needed. You can write over text, add lines, delete lines, move text, or delete text. You can also add comments, labels, and color. Note that each field<sup>7</sup> must have a space before and after it. Use underscores to create data entry fields. Figure 7 below is an example of a screen after customization.

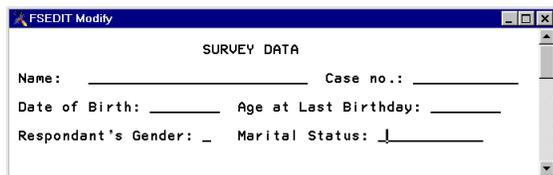


Figure 7—Sample Screen After Customization

To exit and save your changes, click the close box (X) in the upper right corner of the window. When asked if you created any computational fields, respond N.<sup>8</sup> These fields are created using SCL only. Messages at the bottom of the window will then ask you to identify the newly moved fields to the system. When prompted to identify a field, place the cursor anywhere on that field and press <CR>.<sup>9</sup> When you are finished the following message will appear at the bottom of your screen:

NOTE: All fields are identified.

Click the close box to return to the Modification Mode menu.

**Option 4: Assign Special Attributes to Fields**

You should now be back at the Modification Mode menu (Figure 4). Select Option 4, Assign Special Attributes to Fields. At the top center of the frame, you'll see the identifier of the type of attribute you're looking at. From the View menu, select Next Attribute (or Last Attribute) to access attribute screens. Table 1 summarizes the 13 attribute frames, in order.

Table 1 Field Attributes

Attribute	Description
INITIAL	Sets initial values for fields.
MAXIMUM	Sets the maximum value allowed for fields.
MINIMUM	Sets the minimum value allowed for fields.
REQUIRED	Specifies fields that require a value.
CAPS	Specifies fields to be automatically capitalized.
FCOLOR	Specifies the color of each entry field.
ECOLOR	Specifies a color for flagging erroneous data.
FATTR	Assigns special highlighting attributes.
EATTR	Assigns special highlighting attributes for flagging erroneous data.
PAD	Assigns pad characters for blank entry fields.
PROTECT	Specifies protected fields (fields that cannot be edited).
JUSTIFY	Specifies how the values are aligned; left, right, or centered.
NODISPLAY	Specifies which fields should not be visible.
NOAUTOSKIP	Controls whether cursor skips to next field when you type a character in the last position of a field.
NOAUTOBLANK	Allows numeric values to be updated by entering them left justified in a field without manually blanking out the rest of the field.

After you have finished selecting the field attributes, from the View menu, select End to return to Screen Modification screen.

**Modification of General Parameters**

Select Option 5, Modification of General Parameters. Your screen will resemble Figure 8 below.<sup>10</sup> Use this option to modify the screen size, color, and override options, as well as to set defaults for text searches. Changes will take effect after you exit Screen Modification mode and return to the data entry screen.

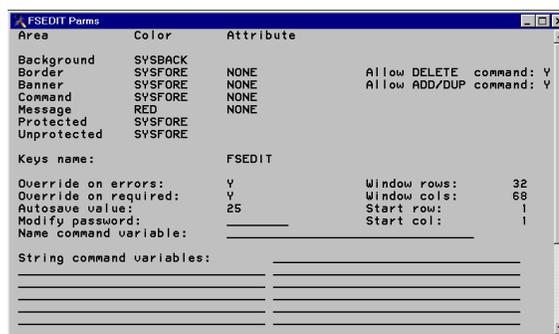


Figure 8—Sample General Parameters

After you set the parameters, click the close box to return to Screen Modification screen. Close this screen to enter data entry mode.

## Data Entry Mode

When you enter data entry mode your customized display will appear on your screen. The file name and observation number will appear at the top of the frame.

The screenshot shows a window titled 'FSEDIT NEW.DEMO1 Obs 0'. Inside the window, the text 'SURVEY DATA' is centered. Below it, there are several input fields: 'Name: \_\_\_\_\_ Case no.: \_\_\_\_\_', 'Date of Birth: \_\_\_\_\_ Age at Last Birthday: \_\_\_\_\_', and 'Respondant's Gender: \_ Marital Status: \_\_\_\_\_'.

Figure 9—Data Entry Screen

To add an observation either press the function key<sup>11</sup> assigned to ADD, or from the Edit menu, select Add New Record. You may also add records by duplicating an existing record and making changes as needed. To duplicate a record, press the appropriate function key, or from the Edit menu, select Duplicate record. Note the following:

- Use the <Tab> key to move to and from the fields on the data entry screen.
- If you have no data to enter into a REQUIRED field, or attempt to enter invalid data into the data field, from the Edit menu, select Override to leave the observation.

## DATA QUERY

There are many ways to search for data within PROC FSEDIT. For example, if you know the observation number, from the View menu, select Go to observation number... and enter the number in the dialog box that appears.

In many cases you will want to query the data based on information other than observation number. Using the Search menu, you can specify any of the following criteria for your query:

- a set of conditions (Where<sup>12</sup>... and Where Also...)
- the value of a field (Name... and Locate...)
- part of the value of a field (Name... and Locate...)
- the value of more than one field (Find...)
- embedded text strings<sup>13</sup> (String... and Search For Value...)

For example, to look for a record with name Cyndie and case number "1", select Find, which allows you to look for the value of more than one field.

To find an embedded string, e.g., "Cyn", use the String and Search for Value commands.

Note also that you can specify default String variables when you modify general parameters.

## DATA EDITING

Editing is done in "real time." Make a backup of your data before you begin the editing process.

- To edit within an observation, type in the new value over the old value.

- To cancel changes within an observation, issue the CANCEL command before moving to the next observation or issuing the SAVE command.

The DELETE command may not physically remove observations from a file. To physically remove deleted records, create a new data set with a DATA step or with any SAS procedure that updates files in place, e.g., PROC SORT.

## PROC FSVIEW

PROC FSVIEW is a newer PROC similar to FSEDIT and FSBROWSE, but more flexible.

- In FSEDIT, data are displayed by observation. In FSVIEW, data are displayed as rows and columns (see Figure 10 below).
- FSVIEW can be used for either editing or browsing. The default mode is BROWSE; use the MODIFY or EDIT option in the PROC FSVIEW statement to edit.
- You must exit FSEDIT before sorting. In FSVIEW, however, you can sort in place from within the procedure. The display will be refreshed to show the newly sorted data.
- In both FSVIEW and FSEDIT, the observation number is displayed for uncompressed files. However, for compressed files, row information is only displayed in FSVIEW.
- FSVIEW allows concurrent windows. The SWAP command moves the cursor to the next concurrent window and activates that window.
- FSVIEW enables you to define formulas that perform computations and otherwise manipulate variables. Formulas are SAS expressions that calculate values for variables in the FSVIEW window. You can define formulas for variables in the data set, or you can create computed variables to display the formula results. Formulas are defined in the FSVIEW define window.
- FSVIEW also allows the creation of computational fields.<sup>14</sup>

Obs	name	caseno	dob	age	sex
1	CYNDIE GARELECK	1	.	.	2
2	TERRY FAIN	2	.	.	1

Figure 10—Example of FSVIEW Screen

## CONCLUSION

Under Version 8, SAS/FSP remains a powerful data management tool which can be used for creating new SAS data sets, data entry, and data query. It is easy to use and easy to learn. SCL can be used to enhance your application, although it is not really needed in most instances.

## SUMMARY OF CHANGES TO SAS/FSP UNDER VERSION 8

Most of the changes to SAS/FSP under version 8 are a reflection of changes to the entire SAS System, rather than to SAS/FSP per se. The most egregious changes are:

- Virtually all menus are different. The Locals and Globals menus no longer exist. Much of their functionality is now

found under the Tools menu. Many tasks are now listed under a different menu than before.

- The data definition screen has a new format.

SAS and SAS/FSP are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. Microsoft and Windows NT are registered trademarks of Microsoft Corporation in the USA and other countries. © indicates USA registration.

Comments and questions may be sent to either of the authors:

Terry Fain  
 RAND  
 1700 Main Street  
 Santa Monica CA 90407  
 Phone (310) 393-0411 x7804  
 FAX (310) 451-7063  
 fain@rand.org

Cyndie Gareleck  
 RAND  
 1700 Main Street  
 Santa Monica CA 90407  
 Phone (310) 393-0411 x7815  
 FAX (310) 393-4818  
 cyndie@rand.org

---

<sup>1</sup> A summary of the changes to SAS/FSP under Version 8 is given at the end of the paper.

<sup>2</sup> Views created with PROC SQL are read-only and not available to FSEDIT.

<sup>3</sup> SAS/FSP may also be accessed via SAS/ASSIST.

<sup>4</sup> FSEDIT (and FSVIEW) may also be invoked from the COMMAND line for existing data sets, but not to create a new data set.

<sup>5</sup> The WHERE statement is called a permanent WHERE clause. It cannot be changed without ending the procedure. We will discuss temporary WHERE clauses when we talk about data query.

<sup>6</sup> "Length" is the number of bytes of internal storage, not screen display length.

<sup>7</sup> Except fields that begin in the first column or end in the last column.

<sup>8</sup> If your machine beeps, make sure you are not in INSERT mode.

<sup>9</sup> Not all fields must appear on the screen. To keep a variable from appearing on the screen, enter **UNWANTED** on the command line when asked to identify its location.

<sup>10</sup> Under Version 8, SAS uses mostly system default colors rather than specifying colors as in Version 6.

<sup>11</sup> To see or modify key definitions, from the Tools menu, select Options, then from the Options submenu, select Keys. If ADD is not listed, add it.

<sup>12</sup> Character data only.

<sup>13</sup> This is called a temporary WHERE clause. It can be canceled by clicking on Search, Where Undo.

<sup>14</sup> The computational fields are only available in the FSVIEW window. To save them as part of your data set, you cannot overwrite the current file; you must create a new one. To create a new data set, select Save As... from the File menu.