DSGEN: MAGICALLY GENERATE SAS® DATA SETS, CUSTOMIZED SCREENS AND FORMATS WITHOUT WRITING SOURCE CODE

Bernadette Johnson and Bonnie Duncan
Pharmaceutical Product Development, Inc.

INTRODUCTION

Creating SAS data sets can be a very boring, redundant and time consuming job for experienced SAS programmers. Whereas, less experienced programmers and non-programmers may find this to be a difficult chore. DSGEN was created to alleviate both conditions.

DSGEN, the data set generator, is a menu-driven application that allows users to define data sets, create format catalogs, and customize data access screens with minimal knowledge of the SAS language. At Pharmaceutical Product Development, Inc. (PPD), our clinical data analysts (CDA) choose initial data set names, field names, descriptions and valid values for data fields on a case report form. After reviewing the information with a programmer, the CDA enters the information into DSGEN and creates the data sets.

This paper gives an overview, describes the interface from the user's perspective, and shows, in the programmer's perspective, how the user input is magically converted into SAS language statements.

OVERVIEW

DSGEN guides data set creation by prompting users to supply the necessary information. English phrases such as "field description" are used instead of SAS terminology like "SAS variable label". Field names become the SAS variable names, while lists of valid values are used to create a user defined format catalog. Simple validation programs check for valid informat, format, and SAS variable names and the existence of duplicate field names. The application creates the data sets, as well as the custom data entry screens. The same information is used to write base SAS programs that can be used later to modify or recreate the data sets and format catalog.

FEATURES

User Friendly Interface	The system replaces the SAS terminology with English prompts. Terms like "variable name" and "informat" are replaced with "field name" and "valid value list".
Wide Target Audience	DSGEN is intended to be used by the non-programmer or programmers who are not familiar with SAS. It may also be used as a tool for SAS programmers to make quick work of data base creation.
Menu-Driven	The application takes the user step by step through the process, presenting different menu choices which guide data set creation.
Data Validation	A few data set design guidelines were established. They are: 1) choose valid names or mnemonics for data sets and fields; 2) use discrete valid value lists or data value ranges to aid data integrity; and 3) all fields must have a corresponding description of 40 characters or less. Simple validation programs ensure that all of the above guidelines are followed. Reports allow the user to verify all of the design specifications before the data set is created.

RESULTS

After the data set design is complete, DSGEN produces:

SAS data sets	All of the data sets defined in DSGEN.
SAS screen Catalog	Contains a customized screen for each data set.
SAS format Catalog	Contains user-defined informats and formats.
Maintenance programs	Base SAS programs for each data set to recreate or modify the data set structure. Also, a program to recreate the user-defined format catalog is created.

COMPARISONS TO EXISTING SAS PROGRAMMING TOOLS

DSGEN provides capabilities not readily available through normal SAS programming features:

English prompts for ease of use -
English phases such as "field description" are used instead of SAS terminology like "variable label".

Ability to modify the data set definition -
The SAS/FSP® software data set definition mode gives the user one chance to define the data set. If any of the information needs to be modified, the definition screen cannot be accessed for changes. Also, the position of the variable in the data set cannot be changed. DSGEN allows the user to change all of the data set specifications.

Customized screens -

After creating a data set using the SAS/FSP software definition mode, a data screen is created. Unless you specify the label option, the variable names are displayed on the screen beside each field. DSGEN automatically uses the field descriptions on the customized screen.

Maintenance programs -

DSGEN creates base SAS programs that can be used to modify or recreate the data sets and user-defined format catalog. These maintenance programs also serve as data set design documentation. This is not automatically available from SAS software.

DSGEN DEMONSTRATION

To illustrate DSGEN, a data set will be created to contain contest enrollment information. The data set, DEMOG, will contain the contestant's name, sex, date of birth, age, height, and special code. The field containing sex will be restricted to accept only M or F. Age is limited to 10-12 and 14-18. The actual height will be entered in feet and displayed as short, average or tall. The special code is reformatted to a group assignment. The User's Perspective will show the steps followed by the user, whereas the Programmer's Perspective will show the behind the scenes conversion of user input to SAS language statements.

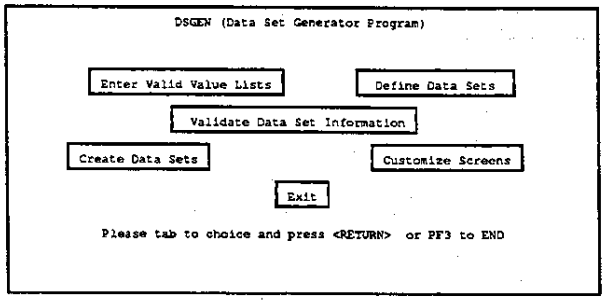
THE USER'S PERSPECTIVE

This section will describe the step by step process of creating data sets, catalogs and screens. The main menu (Figure 1) displays the following features:

1. Enter Valid Value Lists - enter the input and output valid value lists.
2. Define Data sets - enter the data set definitions.
3. Validate Data Set Information - runs programs to verify user input.
4. Create Data Sets - builds user-defined data sets.
5. Customize Screens - generates custom data access screens.
6. Exit - ends session (bye).

Each step is described in detail below.

Figure 1 DSGEN main menu.



Creating the Valid Value List:
Fields can be restricted to accept a list of discrete values or ranges. This allows only correct values to be entered. For example, a field for the contestant's sex should only accept an M or F. If another value is entered, an error message is displayed. Values can be reformatted for output presentation. One of the advantages of reformatting data is to simulate table look-up. For example, state abbreviations can be entered and reformatted to the complete state name. A list that restricts values is an input list, while a list reformatting values constitutes an output list.

Valid value lists are defined through the screen in Figure 2. The list name must begin and end with a non-numeric character. The list type is I for input lists and O for output lists. The data type is either C for character or N for numeric. How the list values are entered is determined by the data type.

Table 1 Sample of valid value list information.

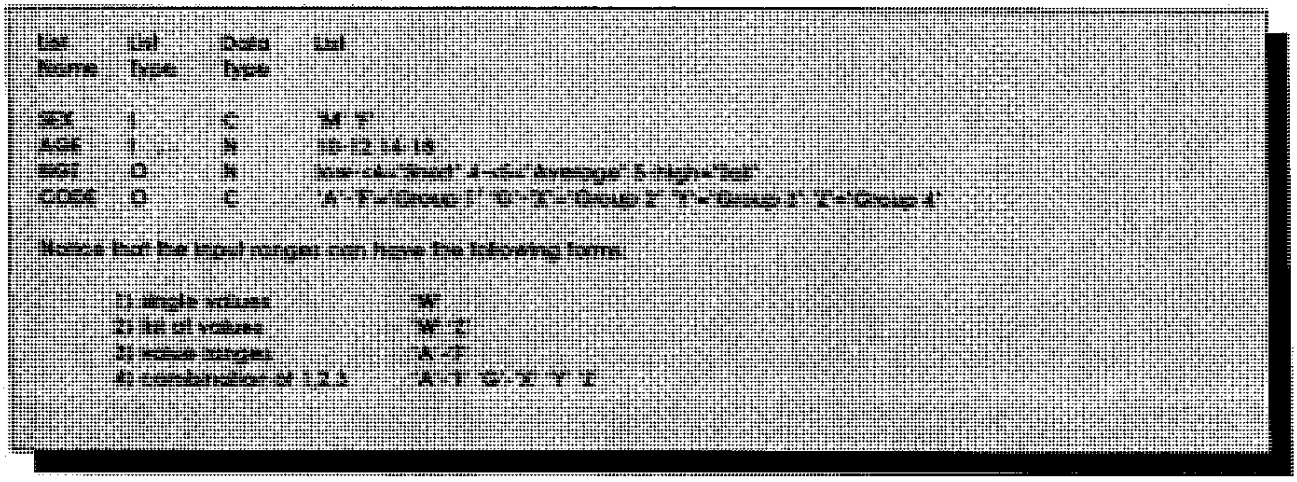


Figure 2 DSGEN: Blank valid value list entry screen.

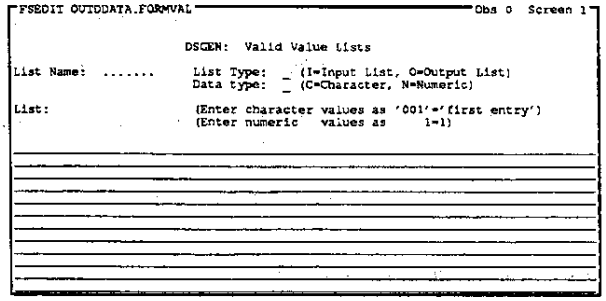
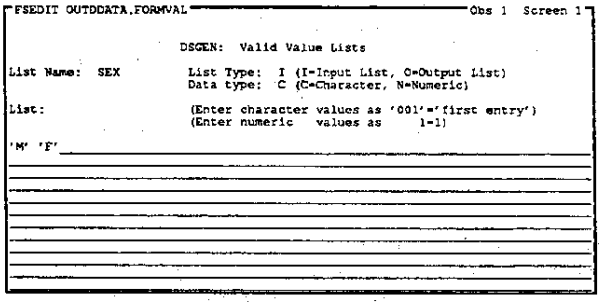


Figure 3 shows an example of an input list to restrict field values. A field for the contestant's sex will be defined on our data set. To limit field values to M and F, an input list named SEX is created. Notice that the list type is I for Input List, and the data type is C to denote a character field.

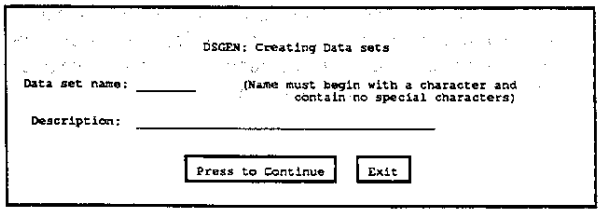
Figure 3 DSGEN: Completed valid value list entry screen.



For this example, the valid value lists that have been created are shown in Table 1.

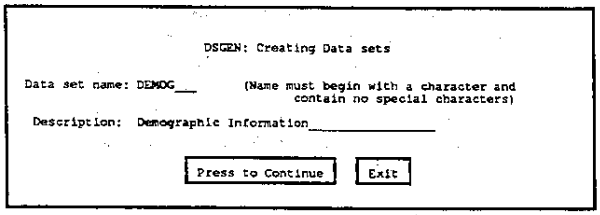
Defining Data Sets:
To define data sets, a user provides the data set name and a brief description of the data set (Figure 4). Individual fields are also defined.

Figure 4 DSGEN choose data set screen.



The example data set name is DEMOG with a description of Demographic Information (Figure 5).

Figure 5 DSGEN data set DEMOG chosen.



Each field is defined along with its description, type, and length. Also, valid value lists are assigned to restrict input values or format output values (Figure 6). If a question mark is entered in the type, input value list or output value list fields, you will be presented with a list of valid choices for that field. Notice that the field TYPE can have the values 'D' for a SAS date field or 'T' to denote a SAS time field. When the field type is 'D' or 'T', the user can select standard SAS date and time value lists.

Figure 6 Blank DSGEN data set definition screen.

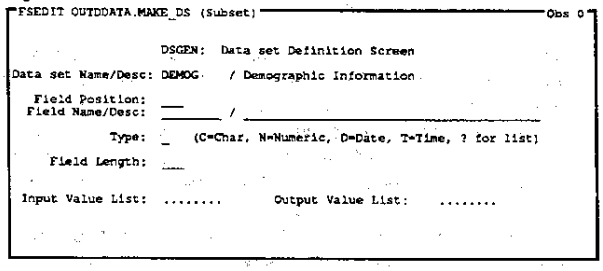


Figure 7 shows an example of the definition for the field SEX as it would be entered on this screen. Notice that the defined input value list SEX is used to restrict the input values for the field.

Figure 7 Completed DSGEN data set definition screen.

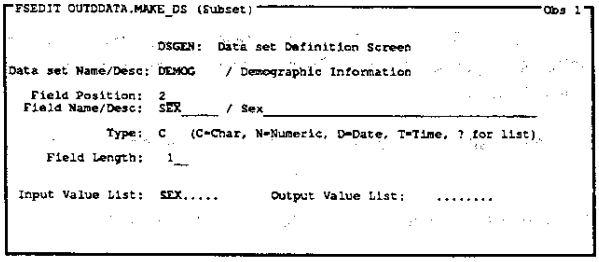


Table 2 lists the fields defined for the DEMOG data set.

Validating Definitions:
On the main menu, press the button labeled "Validate Data Set Information" to submit programs to check the valid value lists and the data set definitions.

Table 2 Sample of data set definitions for DEMOG data set.

Position	Field Name	Field Description	Type	Field Length	Input Value List	Output Value List
1	NAME	Contestant's Name	C	20		
2	SEX	Sex	C	1	SEX	
3	DOB	Date of Birth	D	8	MMDDYY.	MMDDYY.
4	AGE	Age	N	2	AGE	
5	HEIGHT	Height (in feet)	N	3		HGT
6	GROUP	Group Code	C	3		GRP

Creating Data Sets:
Press the button labeled "Create Data Sets" to generate the defined data sets and the format catalog.

Customizing the Data Access Screen

Select 'Customize Screens' on the main menu, to customize a data access screen. Enter the data set name on Figure 8 to access the screen (Figure 9). The screen will automatically be labeled with the field descriptions.

Figure 8 DSGEN: Data access screen

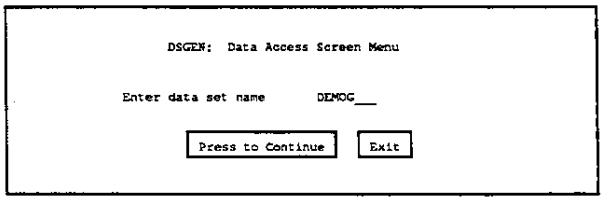
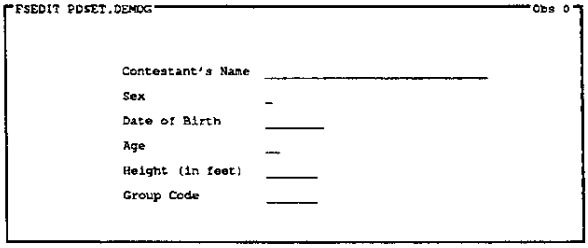
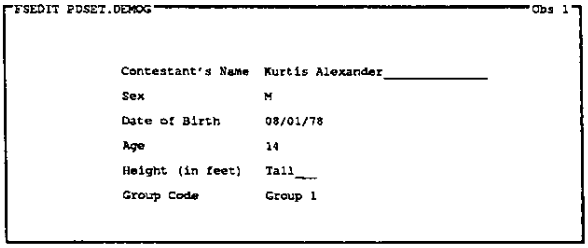


Figure 9 Customized data access screen for DEMOG data set.



The data set has been created and is ready for data entry. The first contestant's name is Kurtis Alexander. The information is entered as M (Sex), 080178 (Date of Birth), 14 (Age), 5.5 (Height), and A (Group Code). Notice in Figure 10 that the height and group code is reformatted to Tall and Group 1 respectively. The date of birth is reformatted to put slashes between the month, day and year. If invalid information is entered into the fields SEX or AGE, the system displays the following message: 'ERROR: Data value is not valid. Please reenter.'.

Figure 10 Using the DEMOG data access screen.



THE PROGRAMMER'S PERSPECTIVE

Creating the Format Catalog:
The "Enter Valid Value Lists" selection in Figure 1 prompts for information to create the format catalog. The user-friendly prompts and the more familiar SAS terminology for defining format catalogs are listed below.

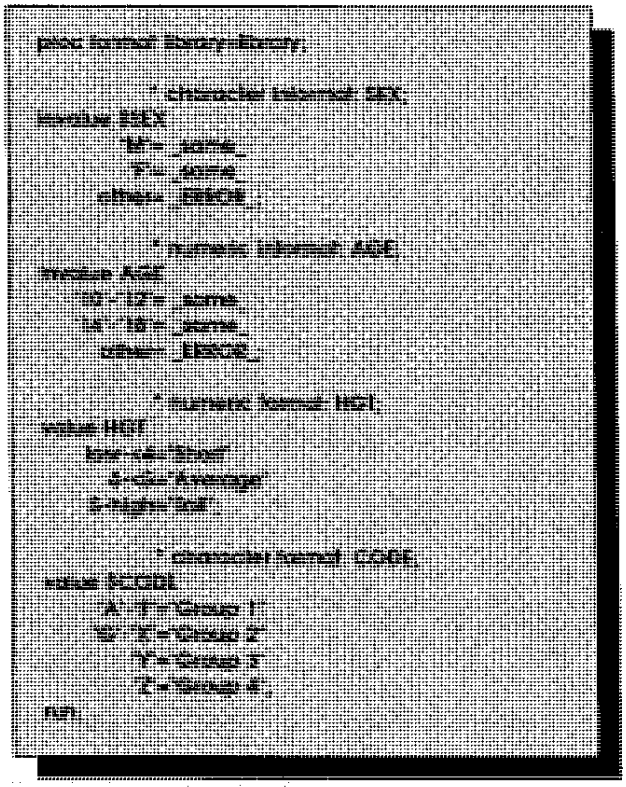
DSGEN Prompt	Format Catalog Term
List Name	Format or informat name
List Type	Type of format:Input list(I)=informat entry, output list(O) = format entry
Data Type	Type of Format: C=Character, N=Numeric
List	Input ranges and formatted values

The list type and date type information are combined to determine the type of format entry.

List Type	Data Type	Resulting Format Entry
I(Input List)	N(numeric)	numeric informat
I(Input List)	C(character)	character informat
O(Output List)	N(numeric)	numeric format
O(Output List)	C(character)	character format

The valid value list in Table 1 is converted to the base SAS program shown in Table 3.

Table 3 Base SAS program to create the format catalog.



Defining Data Sets:
Figures 2 through 7 prompt the user to supply information that will be used to create the data set. The user-friendly prompts and the more familiar SAS terminology for defining data sets are listed below.

DSGEN Prompt	Data Set Definition Terms
Data Set Name	Data Set Name
Description	Data Set Label
Field Position	Variable Position in the Data Set
Field Name	Variable Name
Field Description	Variable Label
Field Type	Variable Type (N=Numeric, C=Character)
Field Length	Variable Length
Input Value List	Informat Name
Output Value List	Format Name

The base SAS program in Table 4 is run to create the DEMOG data set.

Customize Screens:
Data access screens are created for each data set. Select the "Customize Screens" option in Figure 1. After entering the data set name in Figure 8, the default screen is displayed (Figure 9). This screen can be further modified to meet the user's specification. Easily modified screen attributes include: screen text, field length, initial, maximum and minimum values, required fields and field protection and justification.

CONCLUSION

With DSGEN, users no longer need programming knowledge to create SAS data sets. Creating data sets using DSGEN has been challenging and rewarding for the non-programmers at PPD. Programmers act as data set design consultants instead of building the data sets. This shift in responsibility frees programmers to work on other tasks. DSGEN creates base SAS maintenance programs used to make changes in the data set structure or format catalog specifications. The investment in building the application has been justified by the increase in end user satisfaction and the decrease in the programmer data set development time.

DSGEN was developed using Release 6.06 of the SAS System. Base SAS, SAS/AF®, and SAS/FSP software and Screen Control Language (SCL) were used to create the application on a Digital Equipment Corporation VAX™ machine using version 5 of the VMS™ operating system.

Bernadette Johnson Bonnie Duncan
Pharmaceutical Product Development, Inc.
1400 Perimeter Park Drive, Suite 100
Morrisville, NC 27560
Telephone (919) 380-2000 FAX (919) 380-2022

Pharmaceutical Product Development, Inc. is a full-service contract research organization located near Research Triangle Park, NC.

Presented at the SAS Users Group International (SUGI) Conference, May 1993.

Table 4 Base SAS program to create DEMOG data set.