Be Prompt! Do it Now!

Creating and Using Prompts in SAS® Enterprise Guide

Ben Cochran, The Bedford Group, Raleigh, NC

ABSTRACT

Several years ago, Enterprise Guide was given the ability to create prompts. The developers at have continually enhanced this process over the last several releases of Enterprise Guide (EG) and today the user can create quite a wide range of prompts to be deployed in the EG environment. This paper takes a look at some of the prompts that can be created.

INTRODUCTION

Prompts enable your programs/tasks to be interactive in that they request input from the user when code is run. Including prompts in your code increases your ability to reuse code and also enables the code to be customized using the value that you enter for the prompt. You can define prompts that require the user to:

1. **select** one or more values from a list or
2. **enter** one or more values when a task, query, program, or stored process is run.

Prompts can be used to provide:

1. the **name of a variable** in order to run a task or
2. a **value** for a filter expression in a query or program.

When the prompt is run, one or more macro variables are automatically generated for each type of prompt.

Without prompts, the code that you run is **static** even though the data might change.
For stored process prompts, you can also create dependencies between prompts, aka cascading prompts.

You can create prompts in any of the following ways:

1. By using the Prompt Manager section of the Resources pane
2. When you are working in the Query Builder
3. When you are writing a SAS program
4. **When you are creating a stored process.**

TYPES OF PROMPTS

There are many types of prompts that you can create in EG, such as:

1. Text
2. Text range
3. Hyperlink
4. Numeric
5. Numeric range
6. Date / Date range
7. Time / Timestamp / Timestamp range
8. Dynamic list - multiple values.
9. Data source / Data source item
10. File or directory
11. Color
12. Data library
13. Variable

There are many ways to populate prompts, such as:

1. Text – contains character strings,
2. User - entered single line of text,
3. User - entered single line of masked text,
4. User - entered multiple lines of text,
5. User - entered multiple values,

6. Static list - single value
7. Static list - multiple values
8. Dynamic list - single value
9. Dynamic list - multiple values.

To create dynamic lists from datasets, the data has to be registered in the metadata.   When a prompt is executed, it can look like any of these...
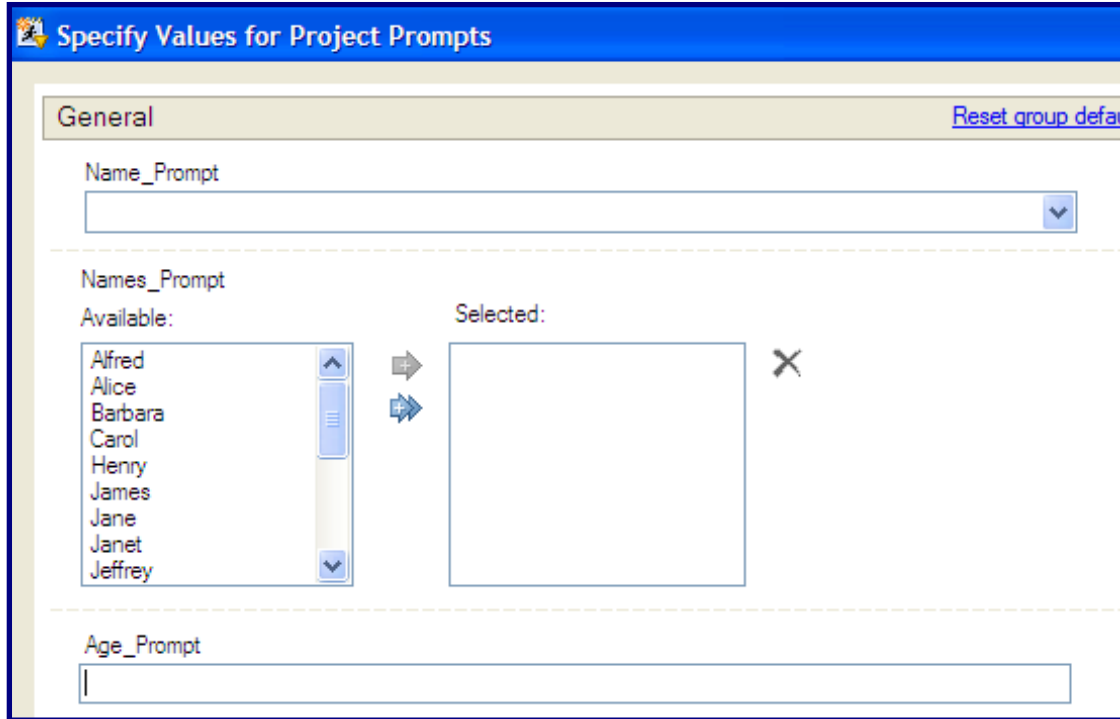

 Figure 1.

The first **Name_Prompt** above is an example of a prompt where the user selects a single value from a static list.
The second **Name_Prompt** above is an example of a prompt where the user selects multiple values from a static list.
The Age_Prompt above is an example of a prompt where the user enters a single line of text.

CREATING A DATE PROMPT
There are many ways to create prompts within EG. One was is to go to the menu facility and select **View -->
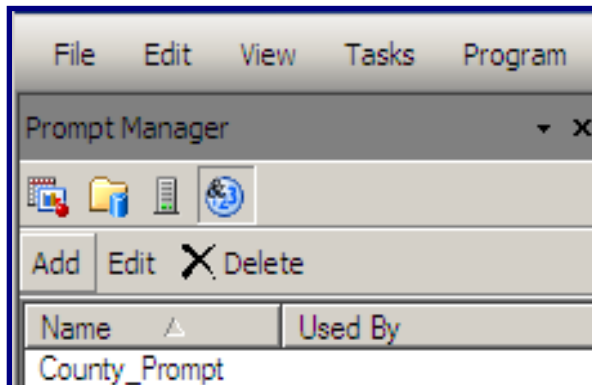Prompt Manager.**   This opens the **Prompt Manager** as seen here...

Figure 2.

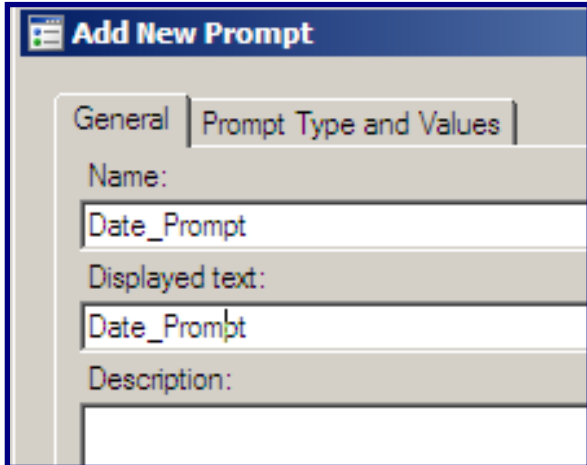Select **Add** to open the Add New Prompt window**.**

Figure 3.

When the Add New Prompt window opens, supply a name and the Displayed Text found on the General tab. Next, select the Prompt Type and Values tab. This is where the majority of the work is done.
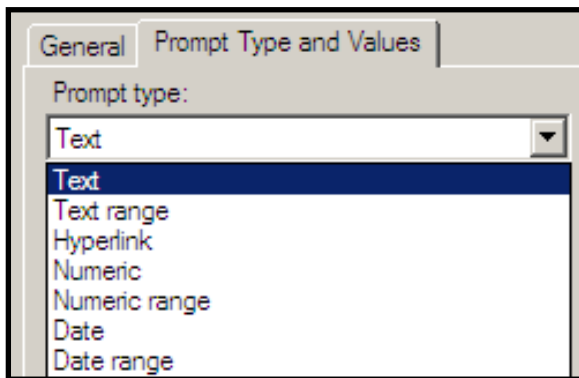

Figure 4.

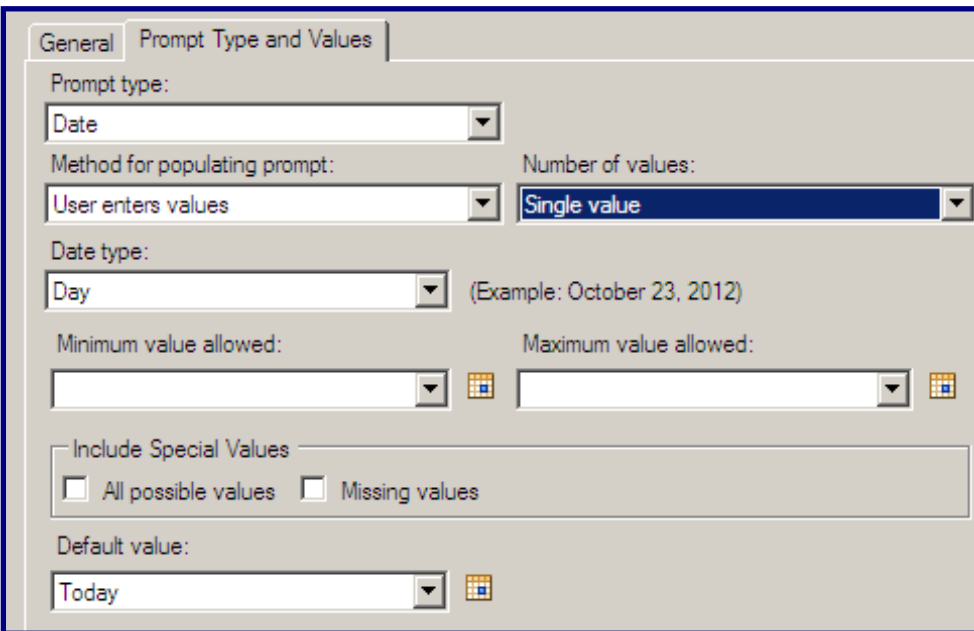From the downward arrow of Prompt Type, select **Date.**

Figure 5.

Make the rest of the window look like the one above (Figure 5).   Set the value of Method for populating prompt to: **User enters values.** The rest of the fields contain default values for Prompt Type Date; except for the Default value field. From the downward arrow, set its value to Today.   While we are going to go with the default values mostly, the next figure gives you an idea of what you could use for the Minimum, Maximum and Default Values field.
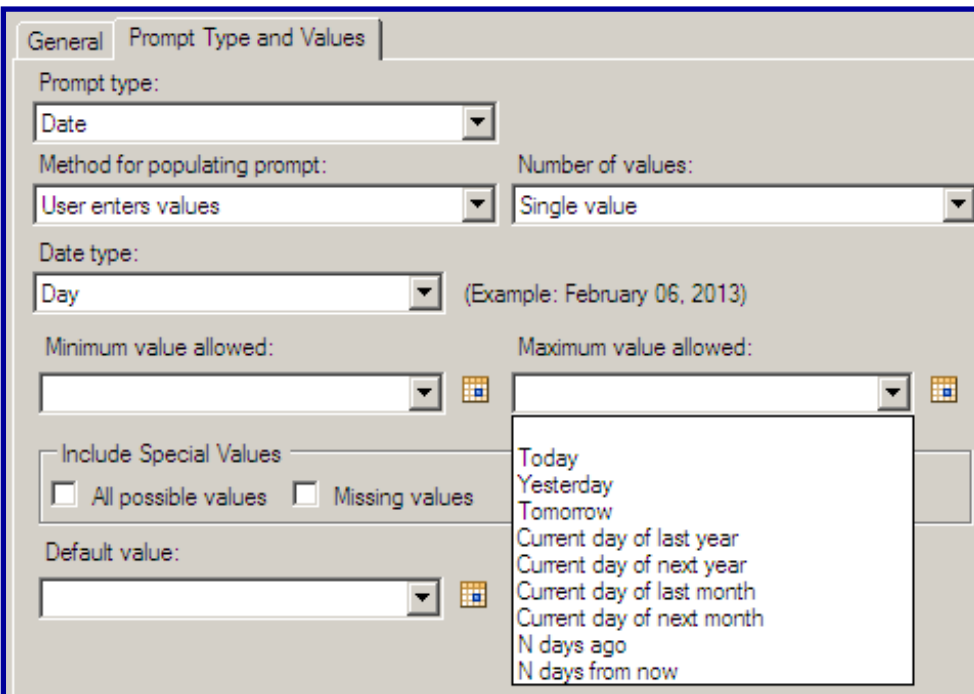

Figure 6.

Select **OK** to complete the creation of the prompt named Date_Prompt.   What we have just done, besides creating a prompt, is we have created a macro variable with the same name as the prompt.   We can now use this macro variable anywhere in the EG project.

Next, open a program window (File --> New --> Program), and write a program where you can use the new macro variable. For example...

4

```
libname workshop 'c:\workshop_EG';


title "First 10 Rows of Enrollment Data set: as of &date_prompt";
proc print data=workshop.enrollment(obs=10);
run;
```

Figure 7.

Close the program window.   From the Process Flow window, right click the program icon and select properties.

When the properties window opens, select Prompts (See Figure 8.). When the Prompts window opens, select **Add.**
This opens the Prompt Selection window. Select the prompt you want to associate with the program (Date_Prompt).
Select OK, then OK again and this will take you back to the process flow window.   When you see the program icon,
it will have a big question mark on it signifying that a prompt has been associated with the program entry (See Figure
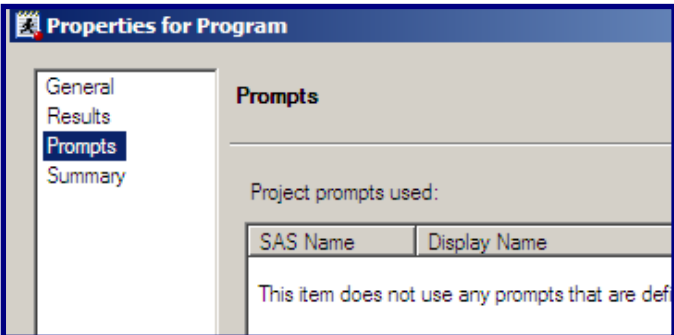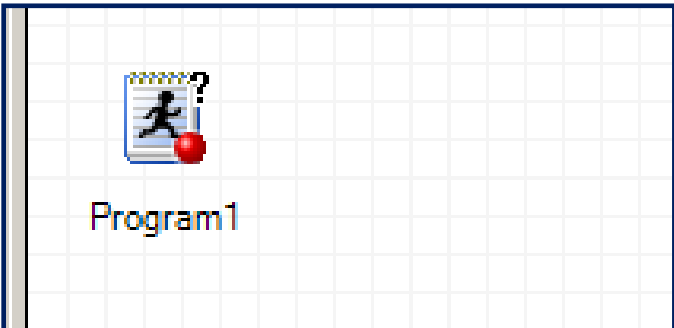9.).

Figure 8.

Figure 9.

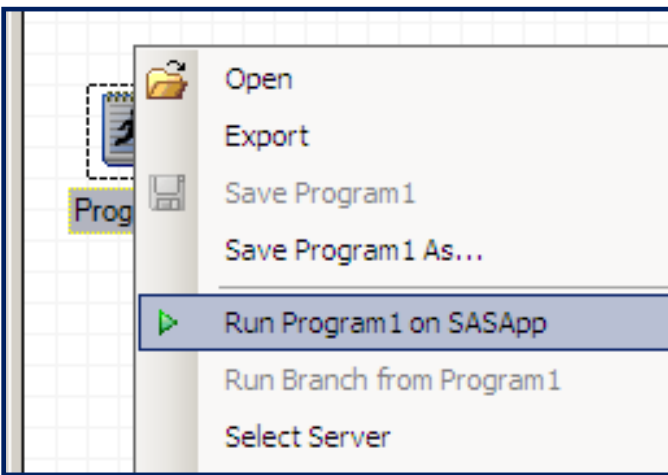Next, right click on the Program1 icon and select **Run Program1 on ...**

5

Figure 10.

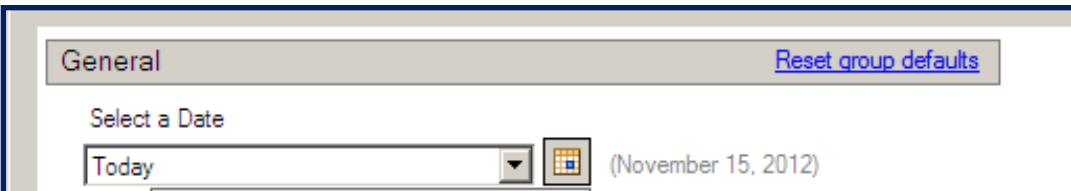When the program runs, the prompt appears (See Figure 11.).

Figure 11.

Select the icon to just to the right of the field where Today is being displayed. This opens a calendar for you to select a specific date. (See Figure 12.).
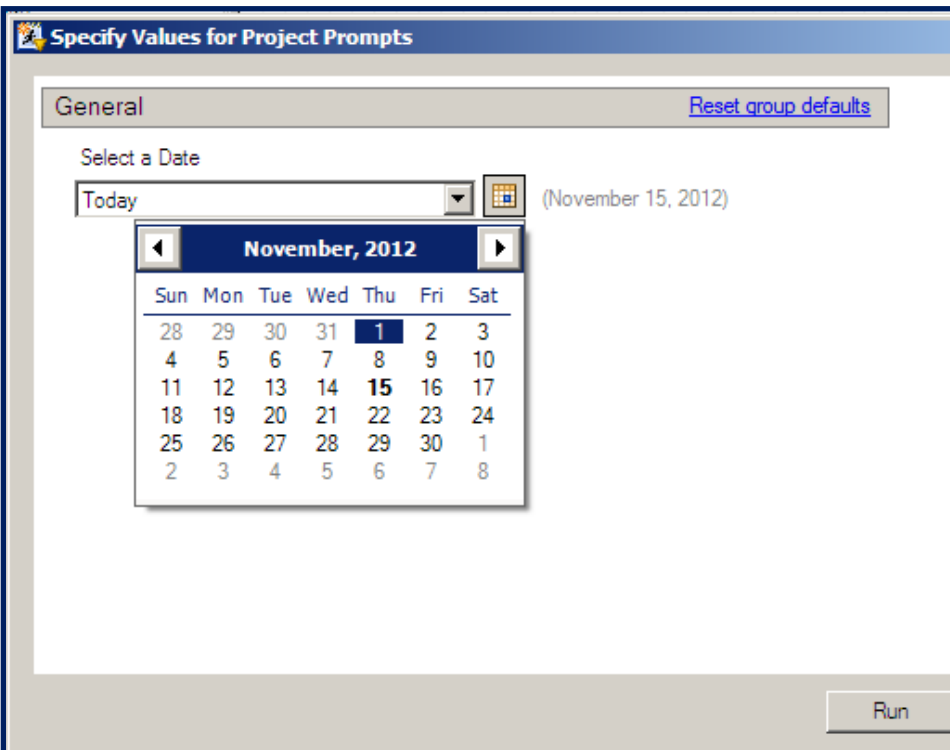

Figure 12.

Scroll to select a date, then select Run in the lower right hand corner of the window. The next thing that appears is the output. Only the first 7 observations of the output are shown. See Figure 13.

Figure 13.

CREATING A DATE RANGE PROMPT
Similar to a Date Prompt is the Date Range prompt. This will be our second type of prompt that will be created. From within the EG environment, select View --> Prompt Manager --> Add.

When the Edit Prompt window opens, supply the name Date_Range on the General tab. Next, select the **Prompt Type and Values** tab.

For the **Prompt Type,** select Date range, then make the rest of the window look like this, then select **OK.**
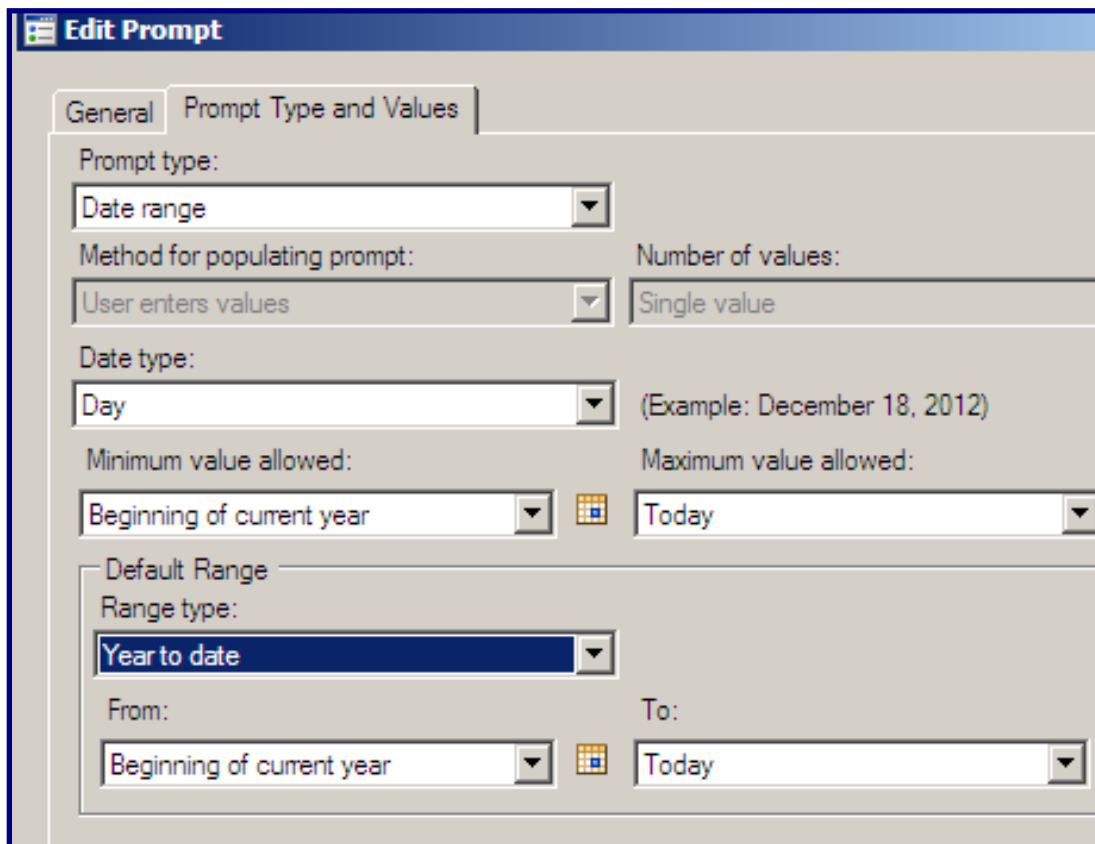
Figure 14.

This prompt creates two macro variables: **&Date_Range_Max** and **&Date_Range_Min.** Next, open a new program window and use them in a program. Own here. nly the TITLE statement is shown here. See Figure 15.
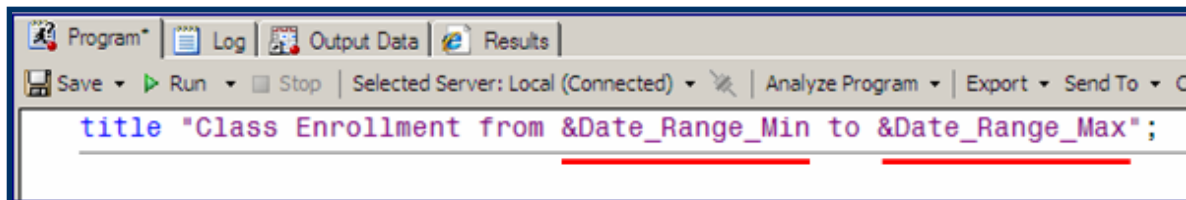

Figure 15.

Close the program window. Right click on the program icon in the Process Flow window and select properties. Select **Prompts** and when the **Select Prompts** window opens**,** select the **Data_Range** prompt**,** then **Add,** then select **OK.** When the **Process Flow** window reappears**,** the program icon has a **'?'** embedded in it indicating that a prompt has been associated with it.
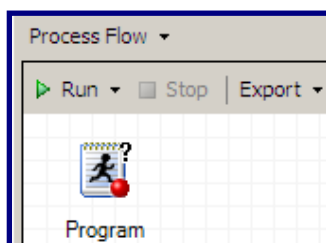

Figure 16.

Right click on the Program icon and select **Run Program on ...** When the program runs, the Prompt appears**.**
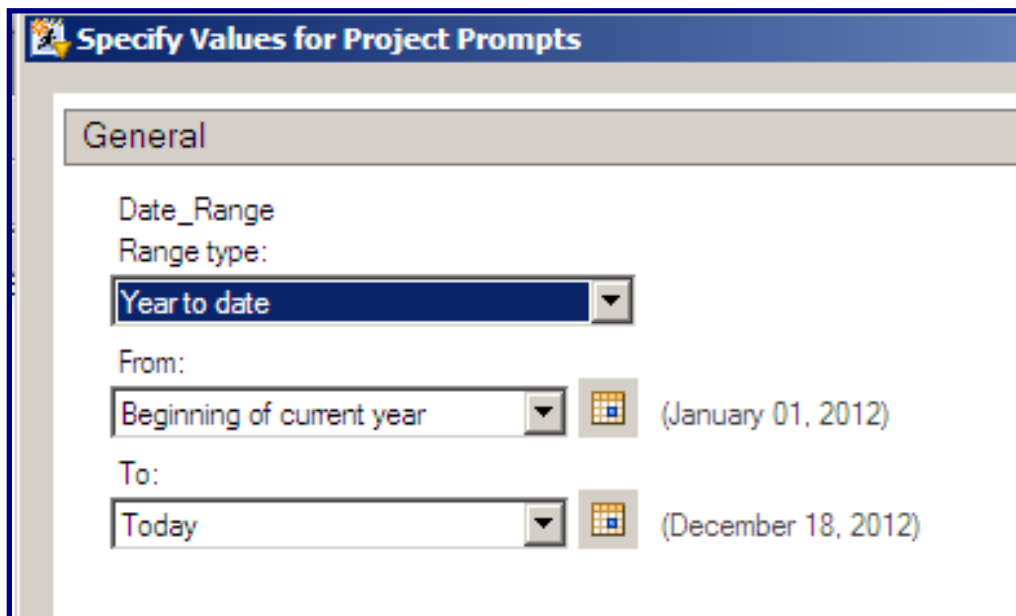
8

Figure 17.

Select a beginning and ending date to define the date range, then select the Run statement at the lower left corner of the prompt. The program puts the date range in the title.

Figure 18.

The macro variables that contain the date return the values in a DATE9 form. They are character strings just as they appear in the title of the above output. if you want to use these values in some sort of program filter you might need to consider doing the following to convert the character values into SAS date values.

```
Beg_Date="&date_range_min"d;
End_Date="&date_range_max"d;
```

CREATING A PROMPT TO FILTER A SAS DATASET
This example will illustrate the act of using a prompt to filter a SAS dataset. Specifically, the SASHELP.CLASS dataset is used in this example. To get started, open the Prompt Manager by selecting View --> Prompt Manager --> Add. When the Add a New Prompt window opens, select the **General** tab and supply a name like **Gender_Prompter** and then select the **Prompt Type and Values** tab. Make the screen look like the one in Figure 19.).
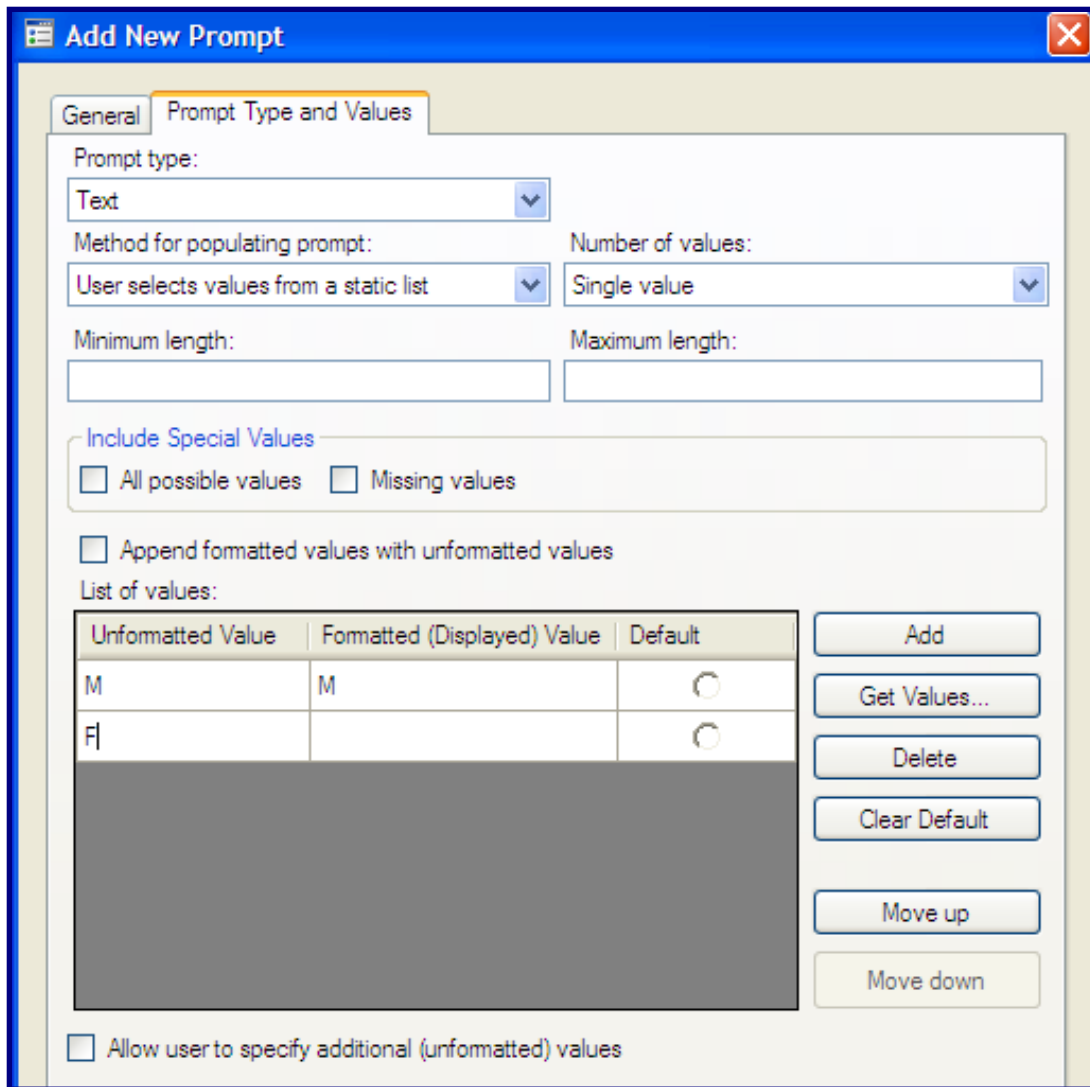
 Figure 18.

Make the Prompt Type: **Text.**   Make the Method for populating prompt: **User selects values from a static list.**
Make the Number of values: **Single values.** Select the Add button in the middle right part of the screen. This button
is used to build the 'static list'.   Type 'M' then select Add again. Then type 'F' to complete the list of values in the list
as M and F.   Finally select **Ok.**   The new prompt (Gender_Prompter) appears in the list of prompts in the
Prompt Manager**.**

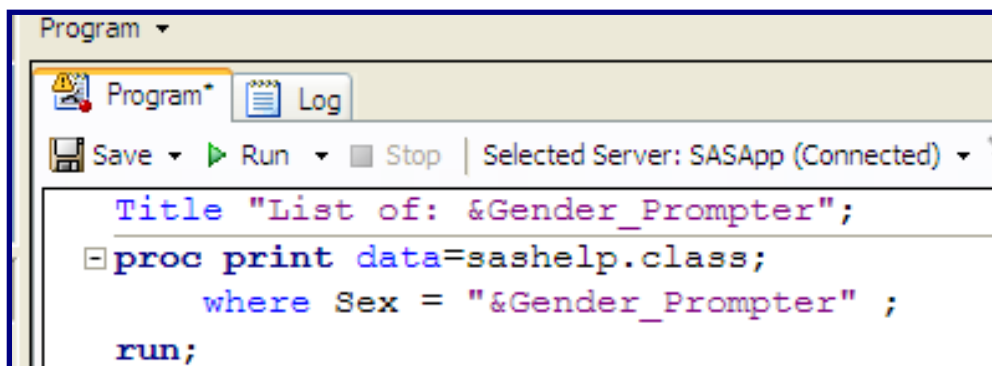Open a program window (File --> New --> Program) and write the following:

Figure 19.

Notice the two locations of &Gender_Prompter.   Close the program window.   In the Process Flow window, right click on the program icon. Go through the same process as before to associate the prompt with the program.   Then when the program runs, the prompt appears.
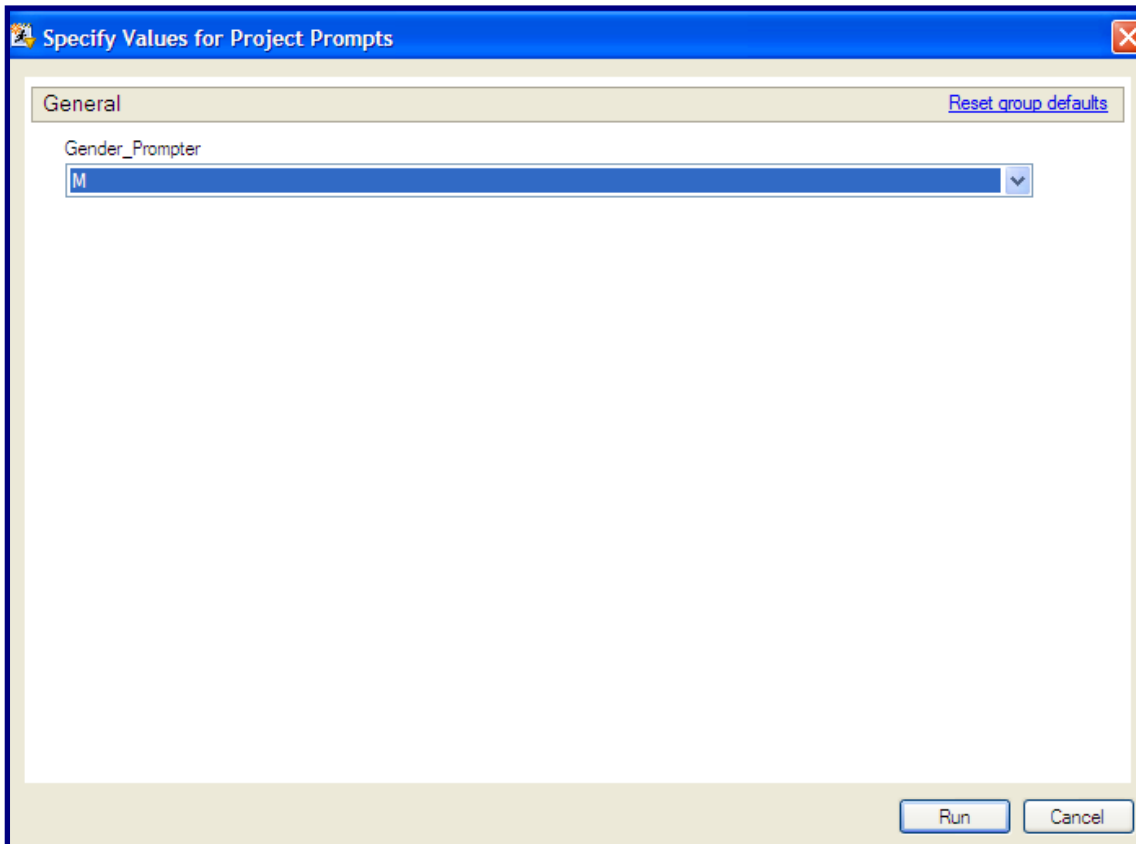

Figure 20.

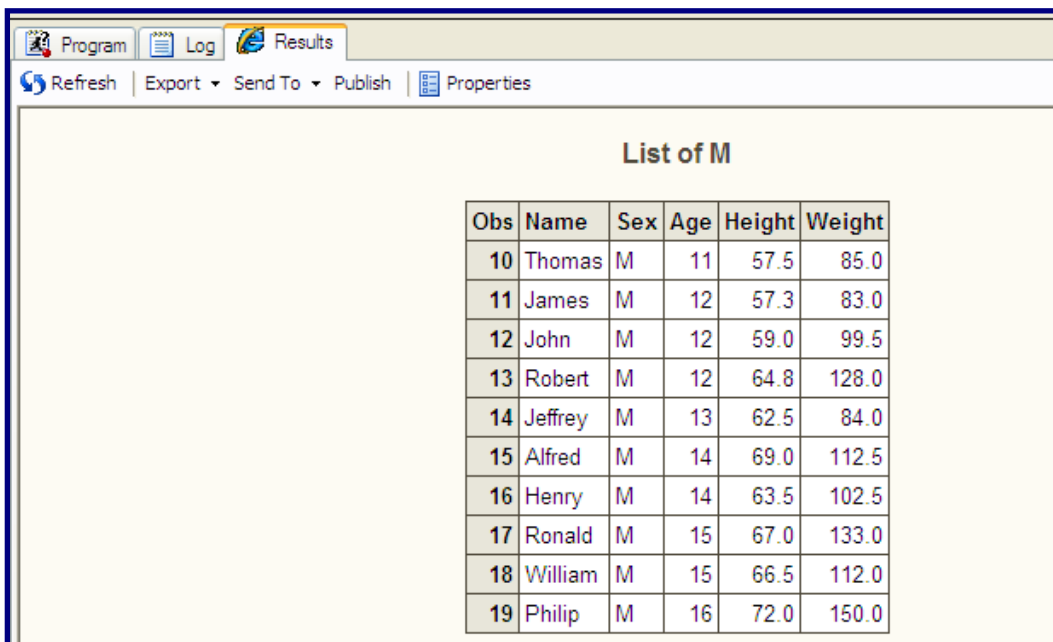Make a selection from the prompt then select **Run.** View the output.

Figure 21.

The output shows the effects of the prompt.

CONCLUSION

Prompts are fairly simple to create, but have very powerful effects on your SAS programs. Prompts can also be used in the Query Builder in EG. Examples of these uses will be shown in the presentation at SGF.   It is the hope of the author that the reader of this paper has learned some valuable information about how to create and use prompts.

ACKNOWLEDGMENTS

As always, I would like to thank the people in the Technical Support department of SAS Institute for their kind and helpful assistance for this past year.   Their knowledge and contributions have helped to make this presentation possible.

CONTACT INFORMATION

If you have any questions or comments, the author can be reached at:
> Ben Cochran
> The Bedford Group
> 3224 Bedford Avenue
> Raleigh, NC 27607
> Work Phone: 919.741.0370
> Email: bencochran@nc.rr.com

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.