Using the SAS Studio Interface: Supplement to *The Little SAS Book, Fifth Edition*

There are several interfaces that you can use to access SAS, including the SAS windowing environment (sometimes called Display Manager), SAS Enterprise Guide, and SAS Studio. SAS University Edition and SAS OnDemand for Academics both use the SAS Studio interface. For people who are using SAS Studio with *The Little SAS Book*, this document replaces sections 1.6 through 1.12, which cover the SAS windowing environment. This document also lists relevant changes to other sections, and gives additional information about accessing files from within SAS Studio. This document was prepared using the single-user edition of SAS Studio 3.4.

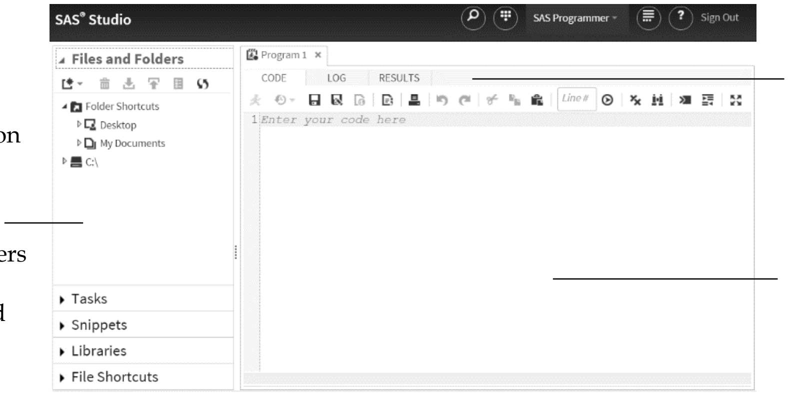
Part 1: Replacement Sections for SAS Studio Users

1.6 Windows and Commands in SAS Studio

SAS Studio is a browser-based interface that links to a SAS server. That SAS server could be your own personal computer, a virtual machine running on your computer, or a remote server. Regardless of how you use SAS Studio, the basic windows are the same.

The SAS Studio interface

When you first open SAS Studio, you will see the navigation pane on the left, and work area on the right. At the top of the window you can choose either Visual Programmer mode or SAS Programmer mode. If you choose SAS Programmer mode, then the work area will show the Program window with the Code, Log, and Results tabs. Other windows also appear in the work area, but are closed by default. The following figure shows the home screen for a SAS Studio session in SAS Programmer mode showing the navigation pane on the left and the Program window in the work area on the right.



Navigation pane showing the Files and Folders section expanded

Code, Log, and Results tabs

Work area showing the Code tab open in the Program window

Navigation pane The navigation pane on the left gives you easy access to your files and SAS libraries as well as tasks, code snippets, and file shortcuts. You can expand a section of the navigation pane by clicking on it. The single-user edition of SAS Studio (as shown here) has a section named Files and Folders, while SAS University Edition and SAS OnDemand for Academics have a section named Server Files and Folders instead.

Program window The Program window appears in the work area on the right and is where you write and submit your SAS programs and view your results. There are three tabs in this window:

> **Code** The Code tab contains a text editor. You can use the editor to type, edit, and run SAS programs as well as edit other text files such as raw data files.
>
> **Log** The Log tab contains notes about your SAS session, and after you run a SAS program, any notes, errors, or warnings associated with your program as well as the program statements themselves will appear in the Log tab.
>
> **Results** If your program generates any printable results, then they will appear in the Results tab.

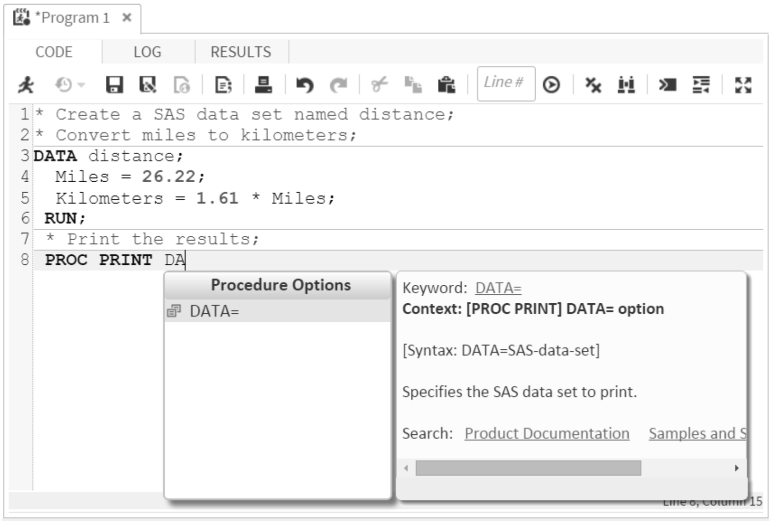
Toolbar The toolbar is located just below the tabs and gives you quick access to icons for many operations. The toolbar is context sensitive, so the available icons will change depending on which windows are open in the work area.

Controlling your windows You can change the relative sizes of the navigation pane and work area by dragging the bar that separates them to the right or left. You can also maximize the work area by clicking the Maximize view icon on the toolbar. With the work area maximized, you will not see the navigation pane. To return the windows to their original configuration, click the Exit maximized view icon .

1.7 Writing and Submitting a Program in SAS Studio

In SAS Studio, you write SAS programs using the code editor and then submit the code to the SAS server that is associated with the SAS Studio session. The SAS server processes the SAS code and returns the results back to your SAS Studio session. The SAS server can be on your local computer or on a remote computer.

Getting your program into the code editor The first thing you need to do is get your program into the editor in the Code tab. You can type your program into the editor, or you can copy and paste programs from another location. You can also open SAS programs that you previously saved by locating them in the Files and Folders (or Server Files and Folders) section of the navigation pane and double-clicking the name. By default, the editor in the Code tab has syntax sensitive help that pops up while you type. The following figure shows a simple SAS program being entered into the editor. You can see that the letters DA were typed on the last line. A help window has popped up showing syntax for the option DATA=. SAS Studio will give you lists of possible keywords and the associated syntax, and when appropriate, will even show you lists of data set names that you can include in your code.



```
* Create a SAS data set named distance;
* Convert miles to kilometers;
DATA distance;
  Miles = 26.22;
  Kilometers = 1.61 * Miles;
 RUN;
 * Print the results;
 PROC PRINT DA
```

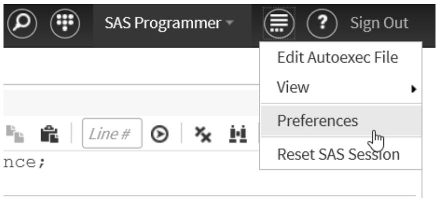
Keyword: DATA=
Context: [PROC PRINT] DATA= option

[Syntax: DATA=SAS-data-set]

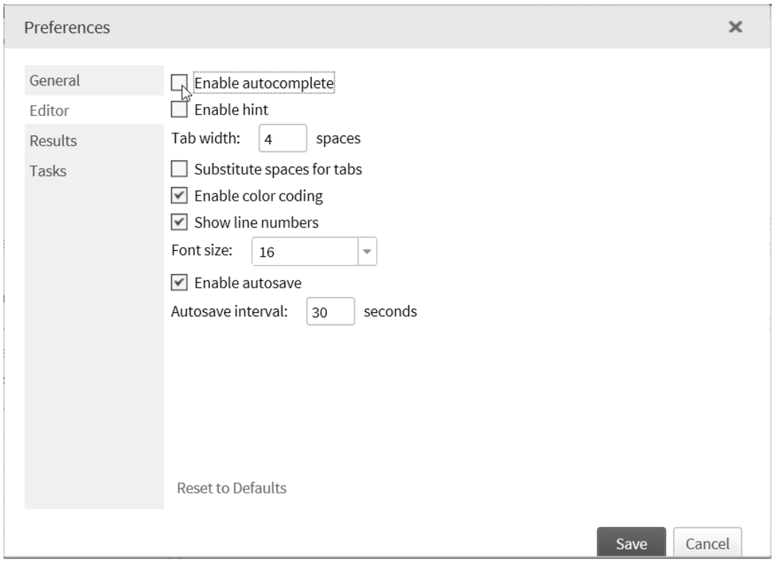
Specifies the SAS data set to print.

Search: Product Documentation   Samples and S

Turning off the syntax help While the syntax help can be very useful, especially if you are new to SAS programming, not everyone likes to see so many pop-up windows. You can turn the syntax help off in the Preferences window. Open the Preferences window by clicking the More application options icon ☰ and selecting **Preferences**.



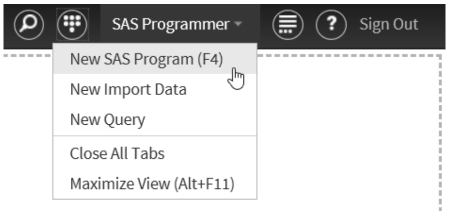
Then in the Preferences window, click Editor in the selection pane on the left and uncheck **Enable autocomplete**. Click **Save** to save the changes.



☐ Enable autocomplete
☐ Enable hint
Tab width: 4 spaces
☐ Substitute spaces for tabs
☑ Enable color coding
☑ Show line numbers
Font size: 16
☑ Enable autosave
Autosave interval: 30 seconds

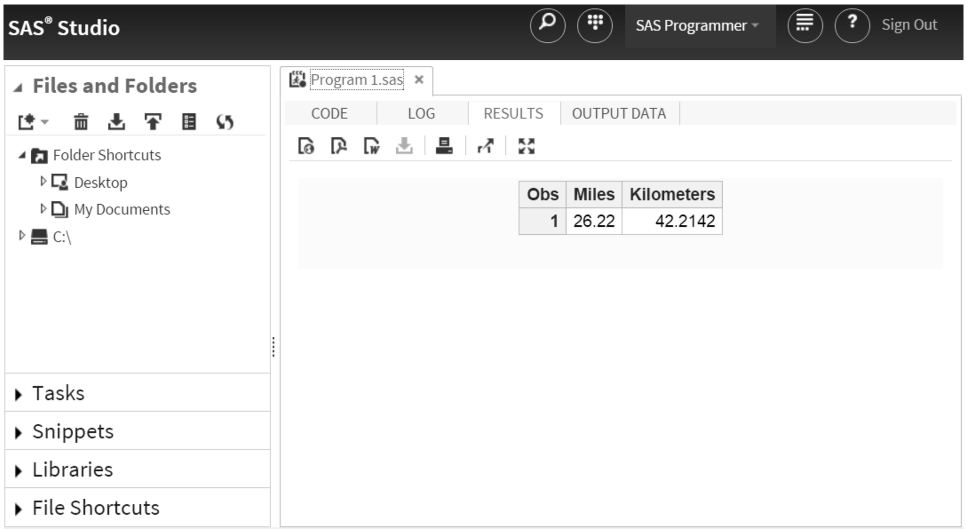
Now the pop-up windows will not appear automatically, but you can still access them as you type by holding down the Control key <ctrl> and pressing the space bar.

Opening a new Program window If you want to open a new Program window, click the New options icon at the top of SAS Studio and select **New SAS Program** from the pop-up menu. This will open an empty Program window in the work area.



Submitting your program Once your program appears in the editor, you submit it (either the whole program, or a part you have highlighted) by clicking the Run icon on the toolbar.

Viewing the SAS log and results After submitting your program, you can view your SAS log in the Log tab, the results (if any) in the Results tab, and any data sets that were generated by the SAS program in the Output Data tab. If your program created results, then the Results tab will appear on top as shown in the following figure.



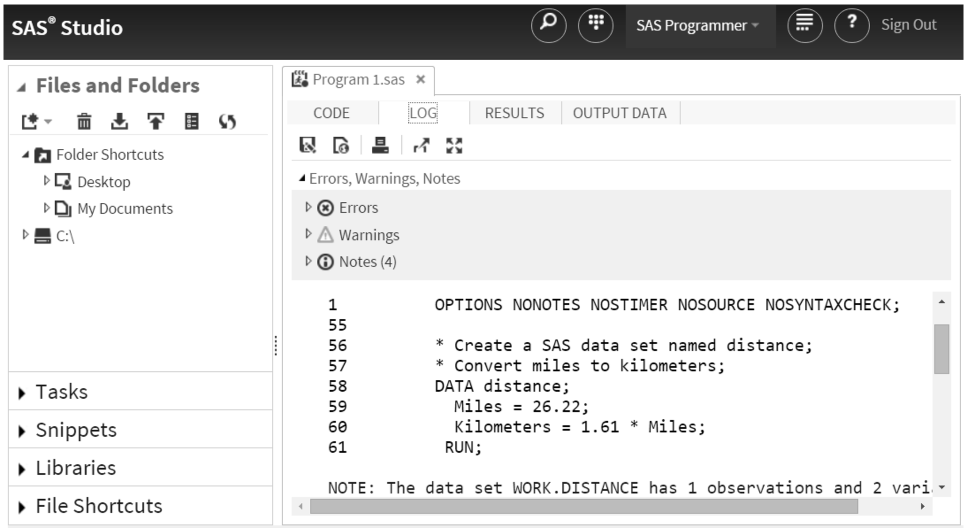
Obs	Miles	Kilometers
1	26.22	42.2142

To view the SAS log, code, results, or output data sets, simply click the appropriate tab in the Program window. The Log, Results, and Output Data tabs will only show the results of the code you most recently submitted. However, you can access previously submitted code in the current SAS Studio session and the associated SAS log and results through the Submission history icon on the toolbar for the Code tab.

1.8 Reading the SAS Log

Every time you submit a SAS program, SAS writes messages in your log. Many SAS programmers ignore the SAS log and go straight to the output. That's understandable, but dangerous. It is possible—and sooner or later it happens to all of us—to get bogus results that look fine in the output. The only way to know they are bad is to check the SAS log. Just because it runs doesn't mean it's right.

Where to find the SAS log Open the SAS log by clicking the Log tab in the Program window. The following figure shows the Log tab after running a simple program. The bottom portion of the Log tab shows the complete SAS log. The top part of the Log tab has nodes that tell you how many errors, warnings, and notes are in the log. If you click the nodes, then you can see the errors, warnings, or notes listed in a condensed format. Clicking any of these will take you directly to that error, warning, or note in the SAS log below.



```
1          OPTIONS NONOTES NOSTIMER NOSOURCE NOSYNTAXCHECK;
55
56         * Create a SAS data set named distance;
57         * Convert miles to kilometers;
58         DATA distance;
59           Miles = 26.22;
60           Kilometers = 1.61 * Miles;
61          RUN;

NOTE: The data set WORK.DISTANCE has 1 observations and 2 vari
```

What the log contains People tend to think of the SAS log as either a rehash of their program or as just a lot of gibberish. OK, we admit, there is some technical trivia in the SAS log, but there is also plenty of important information. Here is a simple program that converts miles to kilometers and prints the result:

```
* Create a SAS data set named distance;
* Convert miles to kilometers;
DATA distance;
   Miles = 26.22;
   Kilometers = 1.61 * Miles;
RUN;
* Print the results;
PROC PRINT DATA = distance;
RUN;
```

If you run this program, SAS will produce a log similar to this:

```
❶ 1      OPTIONS NONOTES NOSTIMER NOSOURCE NOSYNTAXCHECK;
  55
❷ 56     * Create a SAS data set named distance;
  57     * Convert miles to kilometers;
  58     DATA distance;
  59        Miles = 26.22;
  60        Kilometers = 1.61 * Miles;
  61     RUN;

❸ NOTE: The data set WORK.DISTANCE has 1 observations and 2 variables.
❹ NOTE: DATA statement used (Total process time):
        real time           0.03 seconds
        cpu time            0.03 seconds

❷ 62     * Print the results;
  63     PROC PRINT DATA = distance;
  64     RUN;
  NOTE: There were 1 observations read from the data set WORK.DISTANCE
❹ NOTE: PROCEDURE PRINT used (Total process time):
        real time           0.01 seconds
        cpu time            0.00 seconds
```

The SAS log above is a blow-by-blow account of how SAS executes the program.

❶ When you submit programs in SAS Studio, extra statements are executed automatically to ensure that your program runs properly in the SAS Studio environment. The log shows one of these automatic statements. To see all the statements submitted, open the Preferences window by clicking the More application options icon ☰ and selecting **Preferences. T**hen check **Show generated code in the SAS log** in the General tab of the Preferences window and resubmit your program.

❷ It contains the original program statements with line numbers added on the left.

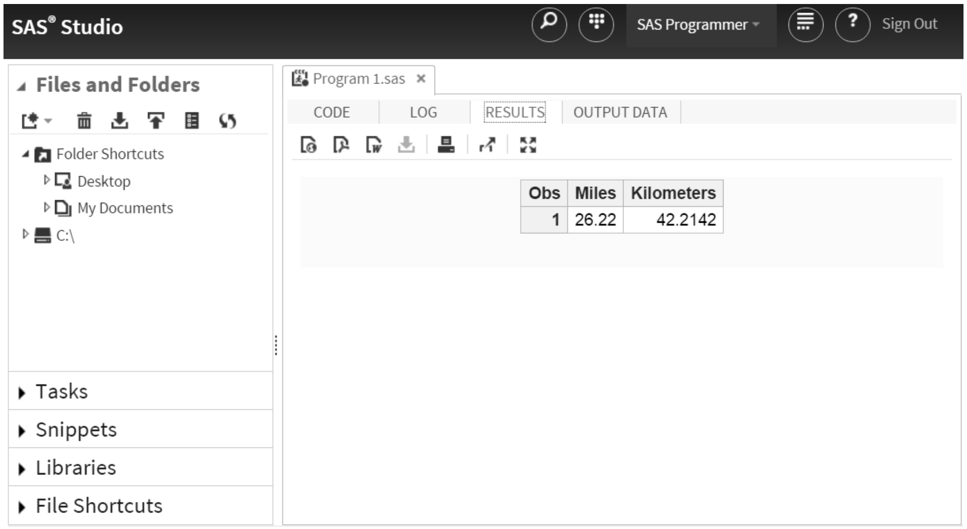
❸ The DATA step is followed by a note containing the name of the SAS data set created (WORK.DISTANCE), and the number of observations (1) and variables (2). A quick glance is enough to assure you that you did not lose any observations or accidentally create a lot of unwanted variables.

❹ Both DATA and PROC steps produce a note about the computer resources used. At first you probably won't care in the least. But if you run on a multi-user system or have long jobs with large data sets, these statistics may start to pique your interest. If you ever find yourself wondering why your job takes so long to run, a glance at the SAS log will tell you which steps are the culprits.

If there were error messages, they would appear in the log, indicating where SAS got confused and what action it took. You may also find warnings and other types of notes that sometimes indicate errors and other times just provide useful information. Chapter 11 discusses several of the more common errors SAS users encounter.

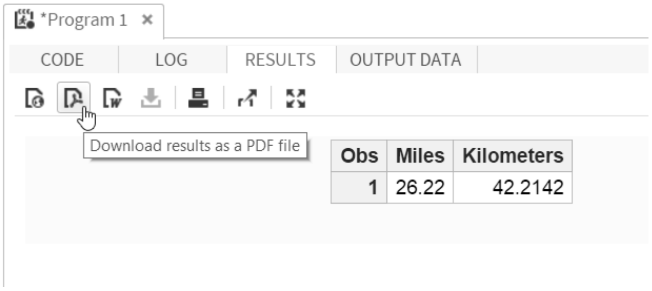
1.9 Viewing Your Results

To view results of any submitted SAS procedures in your program, click the Results tab of the Program window. The following shows the Results tab after submitting the program from the previous section that converts miles to kilometers and prints the result.



Obs	Miles	Kilometers
1	26.22	42.2142

You can print your results, or save your results to a file in either HTML, PDF, or RTF format by clicking the appropriate icon on the toolbar.



Obs	Miles	Kilometers
1	26.22	42.2142

If your program created any SAS data sets, you can view them by clicking the Output Data tab. If more than one data set was created, select the desired data set from the **Table** pull-down menu on the toolbar. The following shows the data set DISTANCE that was created by the program in the previous section.

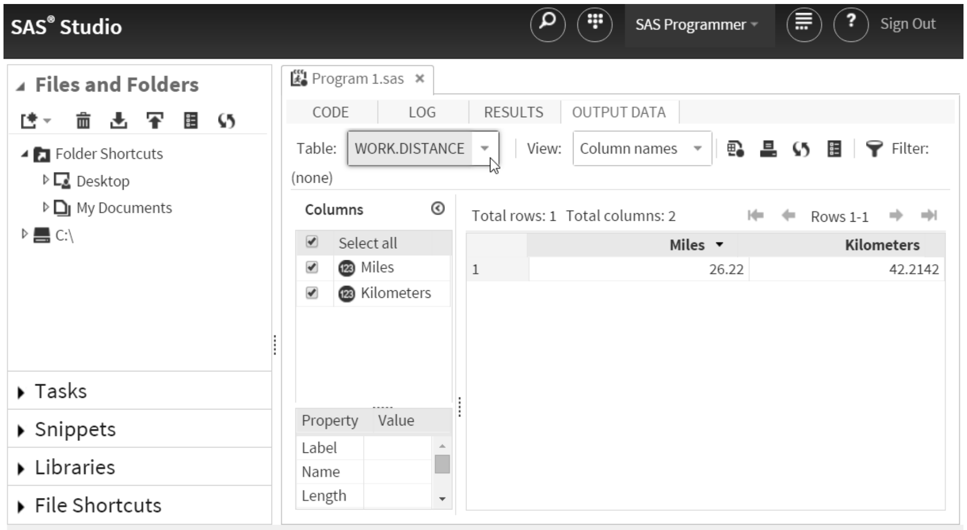


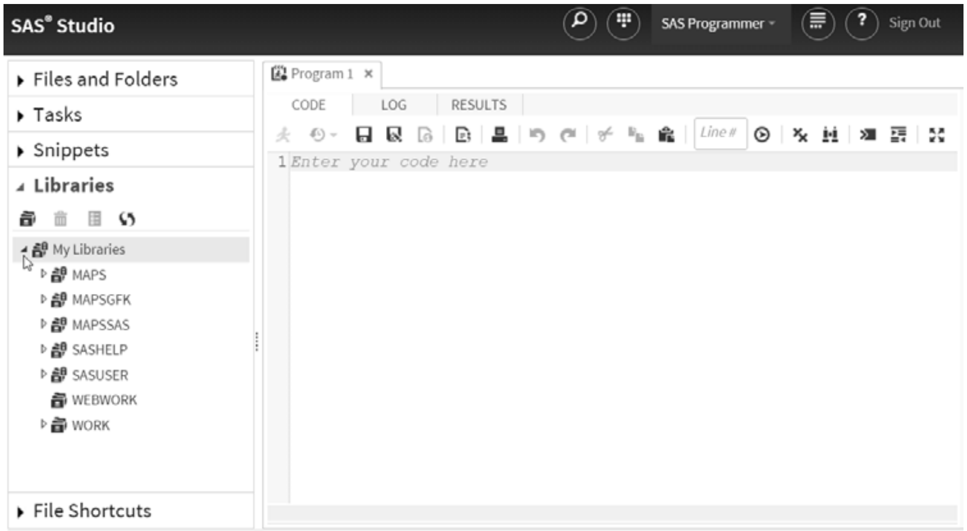
Table: WORK.DISTANCE View: Column names
(none)

Total rows: 1 Total columns: 2

	Miles	Kilometers
1	26.22	42.2142

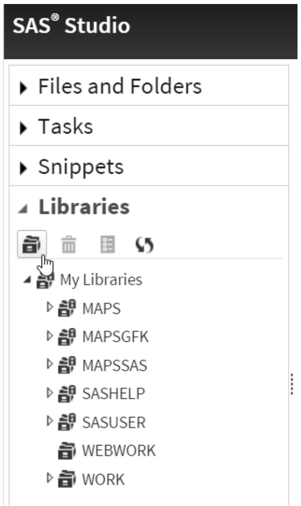
1.10 SAS Data Libraries

Before you can use a SAS data set, you have to tell SAS where to find it. You do that by setting up a SAS library. A SAS library is simply a location where SAS data sets (as well as other types of SAS files) are stored. To set up a library, all you have to do is make up a name for your library and tell SAS where it is. There are several ways to do this, including using the LIBNAME statement (covered in sections 2.19 to 2.20) and using the navigation pane in SAS Studio.

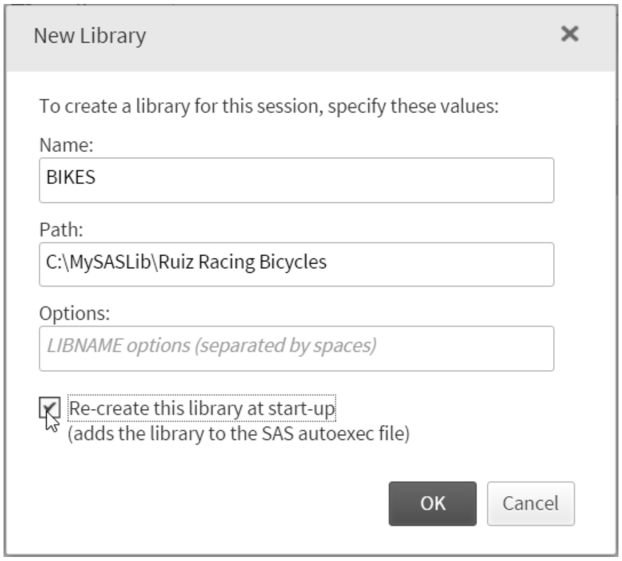


The Libraries section of the navigation pane To view the libraries defined for your session, select **Libraries** in the navigation pane on the left, then select **My Libraries**. You will see at least three libraries: SASHELP, SASUSER, and WORK. You may have other libraries for specific SAS products (such as the MAPS library for SAS/GRAPH software), or libraries that have been set up by you or someone you work with. The SASHELP library contains information that controls your SAS session along with sample SAS data sets. The WORK library is a temporary storage location for SAS data sets. It is also the default library. If you create a SAS data set without specifying a library, SAS will put it in the WORK library, and then delete it when you end your session. The SASUSER library is read-only in SAS University Edition and SAS OnDemand for Academics. In other editions of SAS Studio, you may be able to store SAS data sets, SAS programs, and other SAS files in the SASUSER library. However, many people prefer to create a new library for their SAS files.

Creating a new library You can create new SAS libraries using the New Library window. To open this window, click the New library icon in the Libraries section of the navigation pane.



In the New Library window, type the name of the library you want to create. This name is called a libref, which is short for library reference. A libref must be 8 characters or fewer; start with a letter or underscore; and contain only letters, numerals, or underscores. In the following window, the name BIKES has been typed in as the libref. Type the path for your folder in SAS Studio. If you don't know the path, you can use another method described below. If you want SAS to define your library reference every time you start SAS Studio, then check the **Re-create this library at start-up** box. Click **OK** and then your new library reference will appear under My Libraries in the Libraries section of the navigation pane.



New Library

To create a library for this session, specify these values:

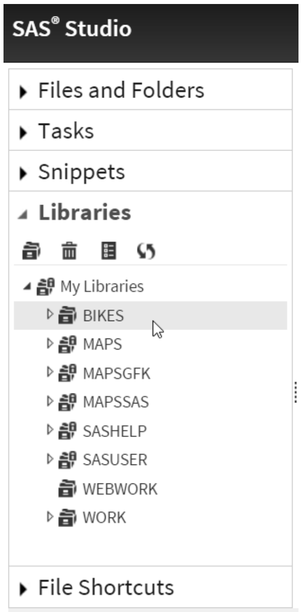
Name: BIKES

Path: C:\MySASLib\Ruiz Racing Bicycles

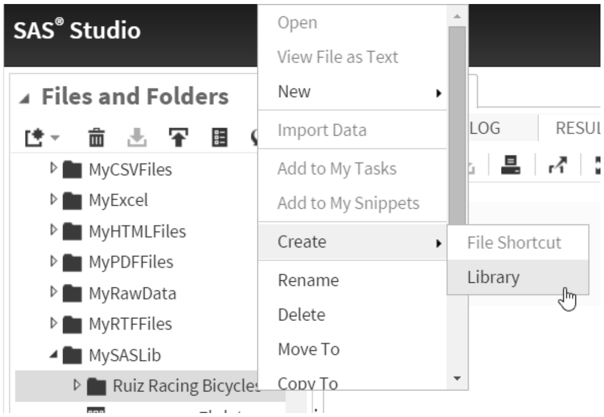
Options: *LIBNAME options (separated by spaces)*

☑ Re-create this library at start-up
(adds the library to the SAS autoexec file)

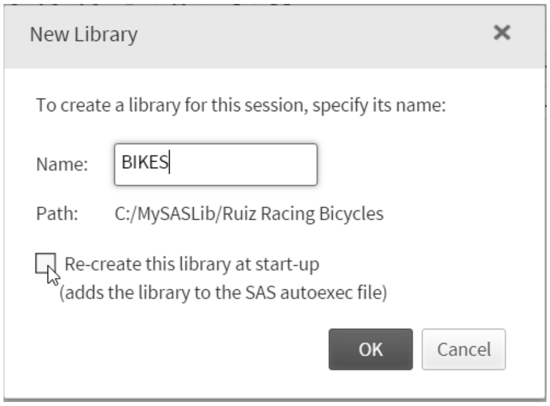
Here is the navigation pane showing the newly created BIKES library.



You can also access the New Library window by clicking Files and Folders (or Server Files and Folders) in the navigation pane on the left and double-clicking the desired folder in the navigation pane. Then right-click the folder and choose **Create ▶ Library** from the pop-up menu.



When you open the New Library window this way, the Path field is already filled in with the path for the folder you selected.


New Library

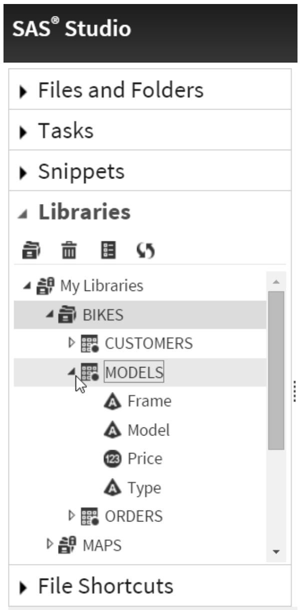
To create a library for this session, specify its name:

Name: BIKES

Path: C:/MySASLib/Ruiz Racing Bicycles

□ Re-create this library at start-up
(adds the library to the SAS autoexec file)

Once your library is defined, you can view a list of the SAS data sets in the library and the variables in the data sets by clicking them in the Libraries section of the navigation pane. The following shows the Models data set in the BIKES library. There are four variables in the Models data set: Frame, Model, Price, and Type.

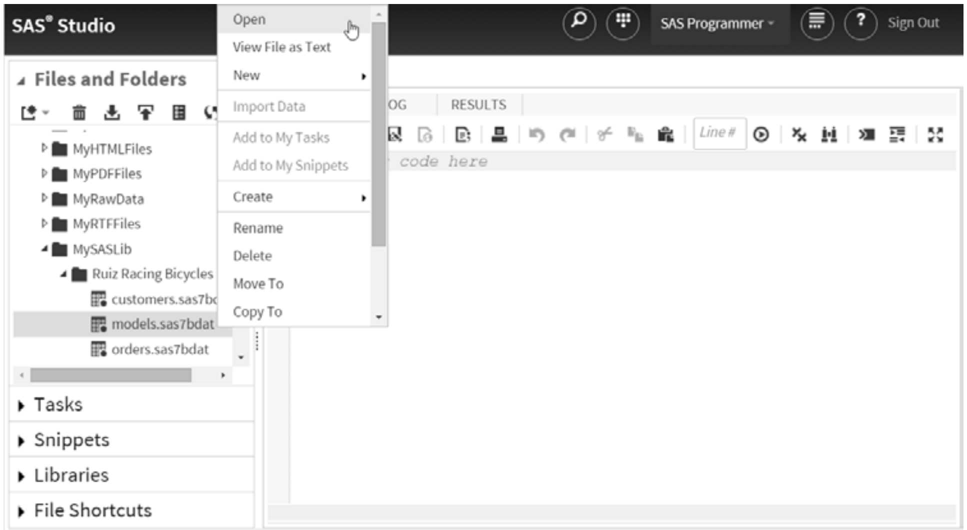


Note that you can also see SAS data set names in the Files and Folders (or Server Files and Folders) section of the navigation pane, but you cannot expand them to see the variables names.

1.11 Viewing Data Sets

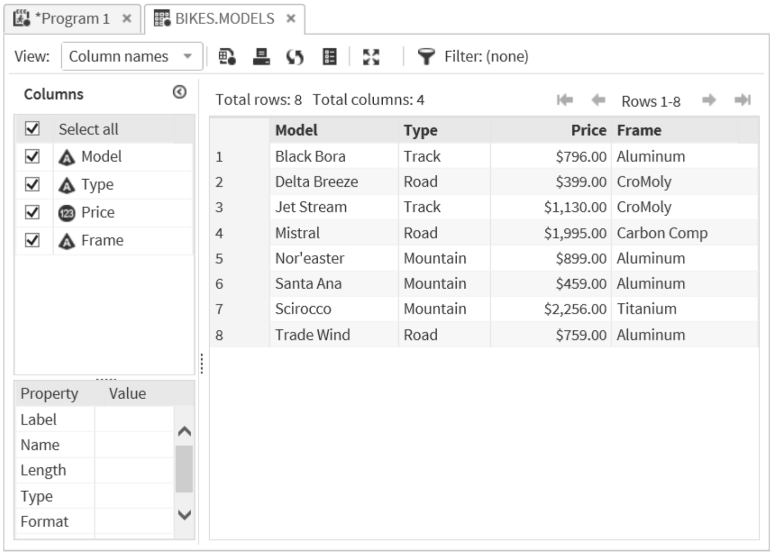
In addition to listing your current libraries and creating new libraries, you can also use the navigation pane to open SAS data sets for viewing. When you are writing programs it is always a good idea to check the data sets you create to make sure they are correct. The SAS Studio table viewer is one way to check your data.

To open a SAS data set in the table viewer, right-click the data set name in the Files and Folders (or Server Files and Folders) section of the navigation pane and select **Open**.



You can also open data sets from the Files and Folders (or Server Files and Folders) section of the navigation pane by double-clicking the data set name or by dragging the data set to the work area. If the data set is in a SAS library, you can open it by double-clicking the data set name in the Libraries section of the navigation pane. In the Libraries section you can expand SAS data sets to view variable names, and drag and drop variable or data set names into your SAS program. (If you don't yet have any SAS data sets of your own, you can view sample data sets that are provided with SAS in the SASHELP library. The CLASS data set in the SASHELP library is a good one to view.)

The SAS Studio table viewer The table viewer allows you to browse, filter, and sort SAS data sets. The following shows the data set named MODELS from the BIKES library. The column names appear in the panel on the left, and the data appear on the right. Initially data for all columns are displayed, but you can hide columns if you like by unchecking them in the panel on the left. The table viewer will display up to 100 observations (also called rows) of data at one time. To see different observations, use the arrows in the top right corner of the table viewer to go forward and backward in your data set.



View: Column names Filter: (none)

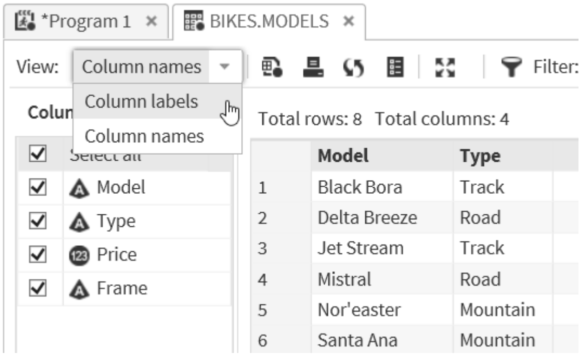
☑	Select all
☑	Model
☑	Type
☑	Price
☑	Frame

Property	Value
Label	
Name	
Length	
Type	
Format	

Total rows: 8 Total columns: 4

	Model	Type	Price	Frame
1	Black Bora	Track	$796.00	Aluminum
2	Delta Breeze	Road	$399.00	CroMoly
3	Jet Stream	Track	$1,130.00	CroMoly
4	Mistral	Road	$1,995.00	Carbon Comp
5	Nor'easter	Mountain	$899.00	Aluminum
6	Santa Ana	Mountain	$459.00	Aluminum
7	Scirocco	Mountain	$2,256.00	Titanium
8	Trade Wind	Road	$759.00	Aluminum

Changing column headings By default, the table viewer displays variable names for column headings. Sometimes you may want to see the variable labels instead of the names. To do this, click the down arrow next to **Column names** and select **Column labels**.



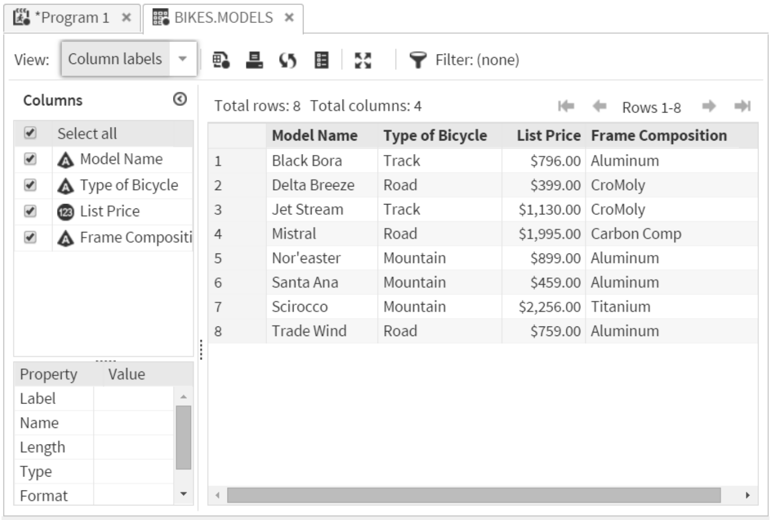
View: Column names

Total rows: 8 Total columns: 4

☑	Select all
☑	Model
☑	Type
☑	Price
☑	Frame

	Model	Type
1	Black Bora	Track
2	Delta Breeze	Road
3	Jet Stream	Track
4	Mistral	Road
5	Nor'easter	Mountain
6	Santa Ana	Mountain

Here is the MODELS SAS data set showing the column (also called variable) labels instead of the names.



*Program 1 | BIKES.MODELS

View: Column labels

Filter: (none)

Columns

☑ Select all
☑ Model Name
☑ Type of Bicycle
☑ List Price
☑ Frame Compositi

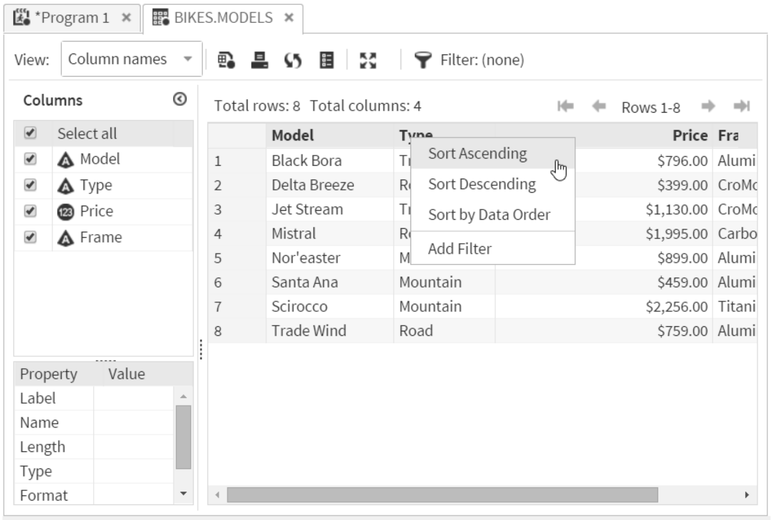
Total rows: 8 Total columns: 4

Rows 1-8

	Model Name	Type of Bicycle	List Price	Frame Composition
1	Black Bora	Track	$796.00	Aluminum
2	Delta Breeze	Road	$399.00	CroMoly
3	Jet Stream	Track	$1,130.00	CroMoly
4	Mistral	Road	$1,995.00	Carbon Comp
5	Nor'easter	Mountain	$899.00	Aluminum
6	Santa Ana	Mountain	$459.00	Aluminum
7	Scirocco	Mountain	$2,256.00	Titanium
8	Trade Wind	Road	$759.00	Aluminum

Property	Value
Label	
Name	
Length	
Type	
Format	

Column options If you right-click a column heading, a pop-up menu will appear. You can sort the data set by ascending, descending, or data order (the order in which observations are stored in the data set), and you can add a filter to the data view. You can also change the sort order by simply clicking a column heading.



*Program 1 | BIKES.MODELS

View: Column names

Filter: (none)

Columns

☑ Select all
☑ Model
☑ Type
☑ Price
☑ Frame

Total rows: 8 Total columns: 4

Rows 1-8

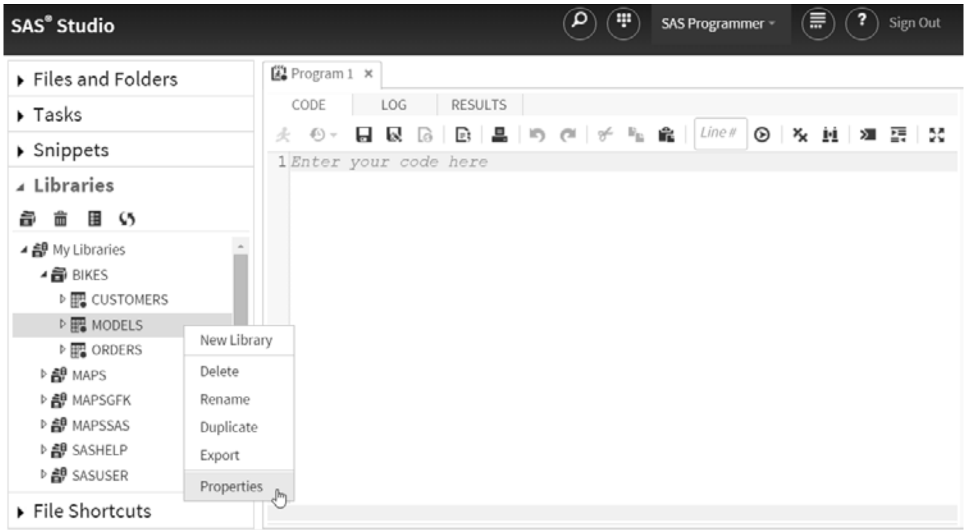
	Model	Type	Price	Fra
1	Black Bora	Tr	$796.00	Alumi
2	Delta Breeze	R	$399.00	CroMc
3	Jet Stream	Tr	$1,130.00	CroMc
4	Mistral	R	$1,995.00	Carbo
5	Nor'easter	M	$899.00	Alumi
6	Santa Ana	Mountain	$459.00	Alumi
7	Scirocco	Mountain	$2,256.00	Titani
8	Trade Wind	Road	$759.00	Alumi

Property	Value
Label	
Name	
Length	
Type	
Format	

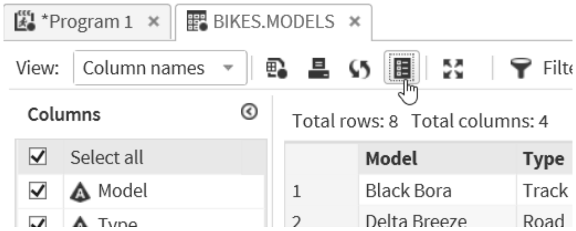
1.12 Viewing the Properties of Data Sets

The Table Properties window for a SAS data set contains some very useful information, such as the date and time the data set was created, the number of observations, all the variable names, and the attributes of the variables. This information is similar to the output produced by the CONTENTS procedure described in section 2.22.

Opening the Table Properties window To open the Table Properties window, click **Libraries** in the navigation pane on the left and then click the library containing your data set. SAS will list all the data sets in the library. Right-click the data set, and select **Properties** from the pop-up menu.



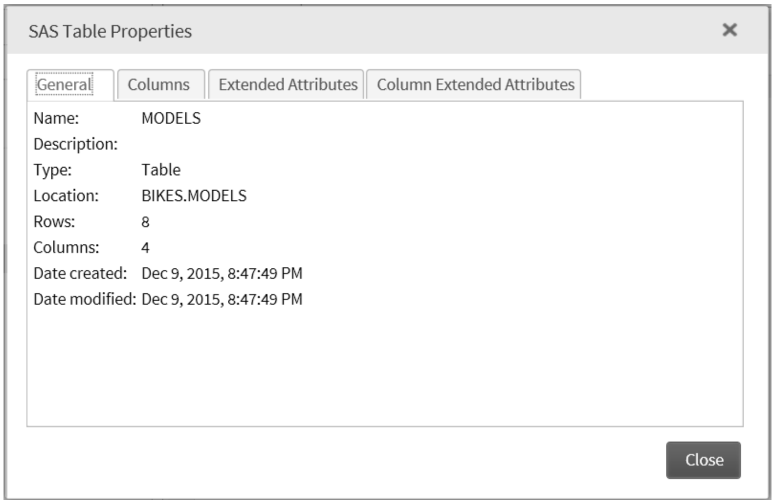
Or, if the table is already open, then you can click the Table properties icon ▤ on the toolbar to open the Table Properties window.



Total rows: 8 Total columns: 4

	Model	Type
1	Black Bora	Track
2	Delta Breeze	Road

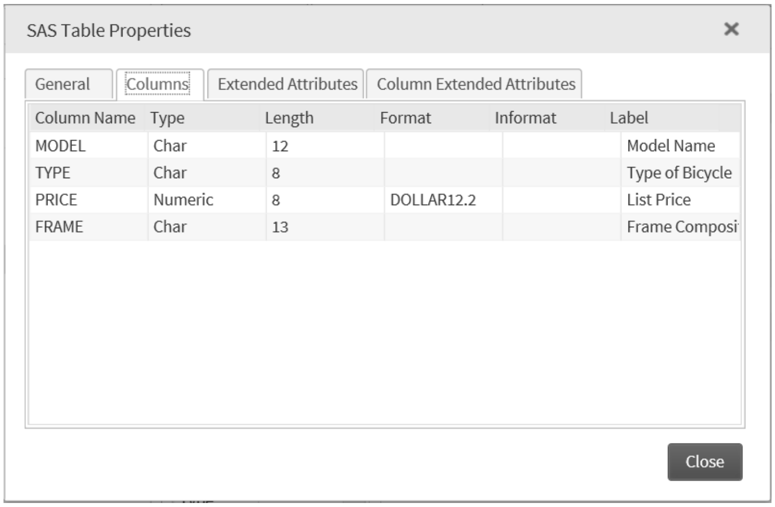
General tab The General tab displays information about the data set such as the date it was created and the number of rows (or observations) and columns (or variables).



SAS Table Properties

Name:	MODELS
Description:	
Type:	Table
Location:	BIKES.MODELS
Rows:	8
Columns:	4
Date created:	Dec 9, 2015, 8:47:49 PM
Date modified:	Dec 9, 2015, 8:47:49 PM

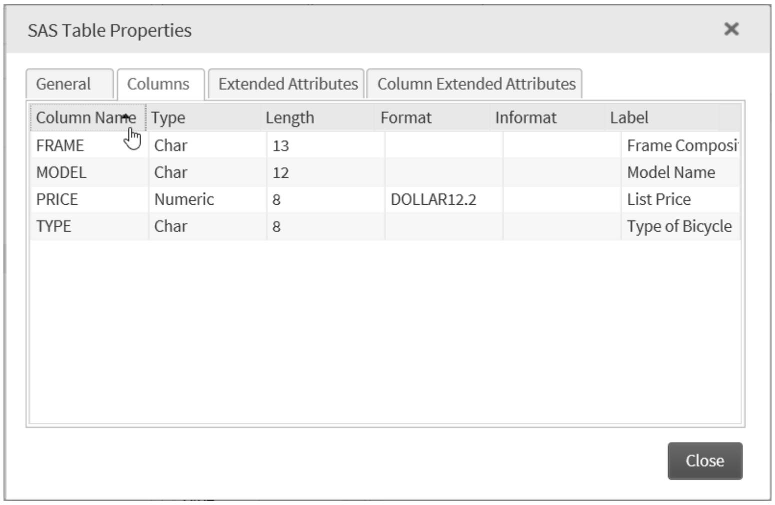
Columns tab If you click the Columns tab, SAS will display information about the columns (or variables) in that data set. The variable name, type, and length are displayed along with any formats or informats assigned to the variable. The variable labels, if any, are also displayed in this window.



SAS Table Properties

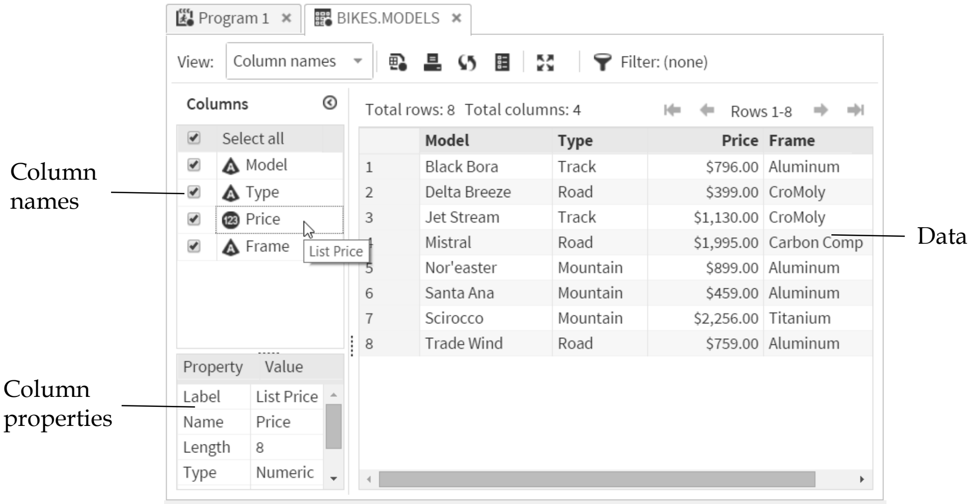
Column Name	Type	Length	Format	Informat	Label
MODEL	Char	12			Model Name
TYPE	Char	8			Type of Bicycle
PRICE	Numeric	8	DOLLAR12.2		List Price
FRAME	Char	13			Frame Composi

If you have lots of variables in your data set, using the sort feature can make your work easier. You can sort any of these columns alphabetically by clicking the column heading. The following window shows the variables sorted by column name.



Column Name	Type	Length	Format	Informat	Label
FRAME	Char	13			Frame Composi
MODEL	Char	12			Model Name
PRICE	Numeric	8	DOLLAR12.2		List Price
TYPE	Char	8			Type of Bicycle

Column properties are also displayed in the work area below the list of variables when you click the name of the column (variable) in the selection pane on the left side of the work area.



Total rows: 8 Total columns: 4

	Model	Type	Price	Frame
1	Black Bora	Track	$796.00	Aluminum
2	Delta Breeze	Road	$399.00	CroMoly
3	Jet Stream	Track	$1,130.00	CroMoly
4	Mistral	Road	$1,995.00	Carbon Comp
5	Nor'easter	Mountain	$899.00	Aluminum
6	Santa Ana	Mountain	$459.00	Aluminum
7	Scirocco	Mountain	$2,256.00	Titanium
8	Trade Wind	Road	$759.00	Aluminum

Property	Value
Label	List Price
Name	Price
Length	8
Type	Numeric

Part 2: Notes on Other Sections for SAS Studio Users

1.13 Using SAS System Options

There is no equivalent to the SAS System Options window in SAS Studio; however, the other three methods for setting SAS system options described in this section do work in SAS Studio.

2.2 Entering Data with the Viewtable Window

SAS Studio uses the SAS Studio table viewer instead of the Viewtable window. At the time this document was written, the SAS Studio table viewer did not provide a way to enter data directly into a SAS data set.

2.3 Reading Files with the Import Wizard

There is no Import Wizard in SAS Studio; however, there is a task that can help you write PROC IMPORT code. Access the task by right-clicking the file you wish to import in the Files and Folders (or Server Files and Folders) section of the navigation pane and selecting **Import Data** from the pop-up menu. SAS Studio also provides code snippets for importing data.

2.4 Telling SAS Where to Find Your Raw Data

SAS University Edition and SAS OnDemand for Academics access SAS via a Linux (a UNIX-like operating system) server. So if you are using SAS Studio in one of these ways, then the file paths in your INFILE statements need to be written in the UNIX style. See part 3 of this document for more information.

2.19 Using Permanent SAS Data Sets with LIBNAME Statements

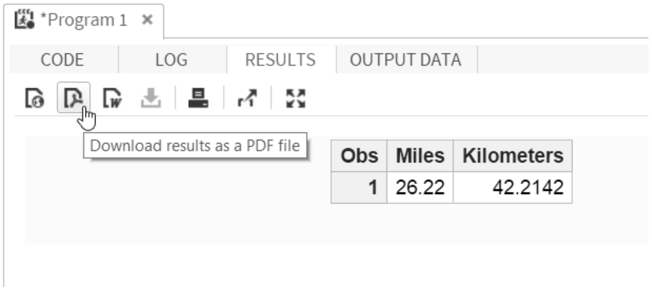
SAS University Edition and SAS OnDemand for Academics use SAS Studio to access SAS running on Linux. So if you are using SAS Studio in one of these ways, then the file paths in your LIBNAME statements need to be written in the UNIX style. See part 3 of this document for more information.

2.20 Using Permanent SAS Data Sets by Direct Referencing

SAS University Edition and SAS OnDemand for Academics use SAS Studio to access SAS running on Linux. So if you are using SAS Studio in one of these ways, then the file path to your SAS data set needs to be written in the UNIX style. See part 3 of this document for more information.

5.1 Concepts of the Output Delivery System

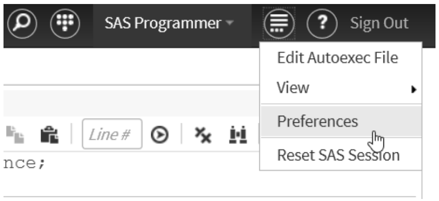
By default, SAS Studio displays your results using the HTML5 destination instead of the HTML destination. In addition, the PDF and RTF destinations are open by default. Only the HTML5 results are displayed in SAS Studio, but you can download any of these results by clicking the icon for HTML , PDF , or RTF in the toolbar in the Results tab.



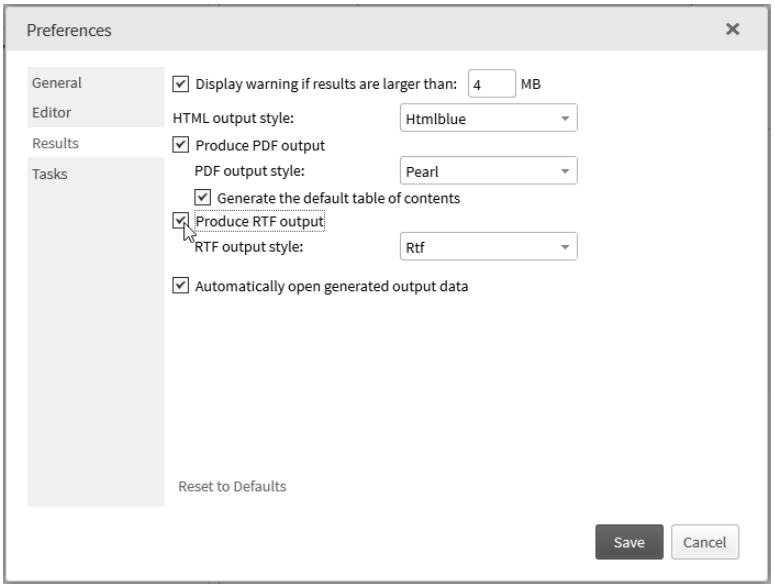
Obs	Miles	Kilometers
1	26.22	42.2142

The default style template for HTML5 output in SAS Studio is HTMLBLUE, for PDF it is PEARL, and for RTF it is RTF.

You can turn off HTML, PDF, or RTF output, or change the style for your results using the Preferences window. To open the Preferences window, click the More application options icon ☰ and select **Preferences**.



Then in the Preferences window click Results in the selection pane on the left. To turn off a particular type of output, uncheck the corresponding box. To change a style template, click the down arrow next to the corresponding output style and select a style from the pull-down list.



☑ Display warning if results are larger than: 4 MB

HTML output style: Htmlblue

☑ Produce PDF output

PDF output style: Pearl

☑ Generate the default table of contents

☑ Produce RTF output

RTF output style: Rtf

☑ Automatically open generated output data

5.4 Creating Text Output

There are no options in SAS Studio to create text output automatically. However, you can use ODS LISTING statements with a FILE = option to write to a text file like this:

```
ODS LISTING FILE = 'c:\MyTextFiles\AnnualReport.lst';

SAS procedure statements

RUN;
ODS LISTING CLOSE;
```

It is a good idea to include a statement to close the LISTING destination. Otherwise you may not be able to open your text file in another application. You can view your text file in SAS Studio by double-clicking the filename in the Files and Folders (or Server Files and Folders) section of the navigation pane, or by dragging the filename to the work area.

5.5 Creating HTML Output

By default, SAS Studio renders your results using the HTML5 destination instead of the HTML destination. HTML5 is the fifth revision to the HTML standard markup language. However, you can still create HTML files and control the style using ODS HTML statements with a FILE = option like this:

```
ODS HTML FILE = 'c:\MyHTMLFiles\AnnualReport.html'
      STYLE = BARRETTSBLUE;

SAS procedure statements

RUN;
ODS HTML CLOSE;
```

You can change the style for the default HTML5 output that is displayed in SAS Studio using the Preferences window as described in the notes for section 5.1.

10.2 Writing Files Using the Export Wizard

There is no Export Wizard in SAS Studio, but you can still write PROC EXPORT code, and SAS Studio provides code snippets for generating data.

11.6 Note: Invalid Data

Sometimes there are issues reading raw data files created on a Mac or Windows computer when you are using SAS University Edition or SAS OnDemand for Academics. This happens when the end-of-line character in the raw data file is not what these UNIX-based environments expect. If you find that you are getting notes about invalid data when reading raw data files, then try adding the TERMSTR= option to your INFILE statement.

For raw data files that originated from a Mac, use TERMSTR=CR as in this example for SAS University Edition:

```
INFILE '/folders/myfolders/President.dat' TERMSTR = CR;
```

For raw data files that originated from Windows, use TERMSTR=CRLF as in this example for SAS University Edition:

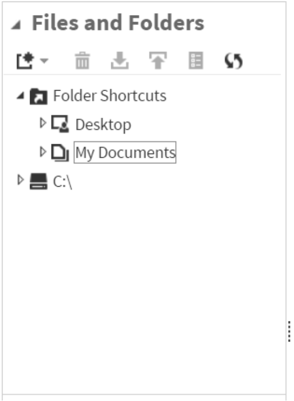
```
INFILE '/folders/myfolders/President.dat' TERMSTR = CRLF;
```

Part 3: Accessing Files

How you access your files from SAS Studio depends on which version of SAS Studio you are using. This section discusses how to access files in the single-user edition of SAS Studio, SAS OnDemand for Academics, and SAS University Edition.

Single-User Edition

Because the single-user edition of SAS Studio is installed on the same computer as the SAS server, accessing files is straightforward. The following shows the Files and Folders section of the navigation pane for Windows.



You have direct access to all files stored on your local computer. You can click C:\ to see files and folders contained on the C drive, or you can use the defined Folder Shortcuts.

Examples When specifying file names and paths in INFILE and LIBNAME statements, simply use the operating system path. For example, if you have a file called President.dat in the MyRawData folder on the C drive (Windows), then the INFILE statement for that file would be

```
INFILE 'c:\MyRawData\President.dat';
```

Here is an example of a LIBNAME statement that defines a SAS library named PLANTS that points to the MySASLib folder on the C drive (Windows):

```
LIBNAME plants 'c:\MySASLib';
```

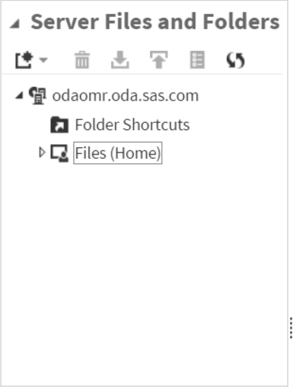
Here is an example of how you would reference a SAS data set named MAGNOLIA in the MySASLib folder on the C drive (Window) using direct referencing.

```
DATA 'c:\MySASLib\magnolia';
```

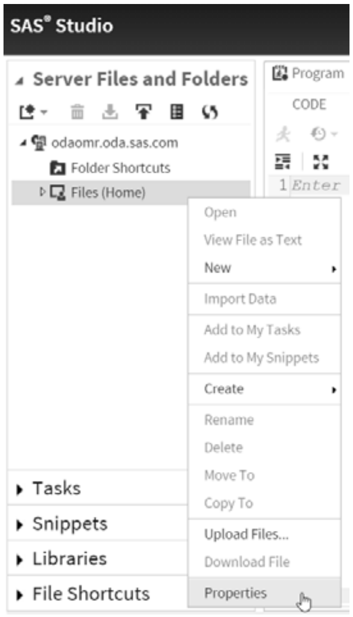
SAS OnDemand for Academics

SAS OnDemand for Academics uses SAS Studio in a Linux server environment hosted by SAS Institute. When you set up your account, you are given storage space for your files on the server. If you have any data files or SAS programs stored on your local computer that you want to use with SAS OnDemand for Academics, then you must upload the files to the server. Click the Upload files icon in the Server Files and Folders section of the navigation pane to upload files to your account.

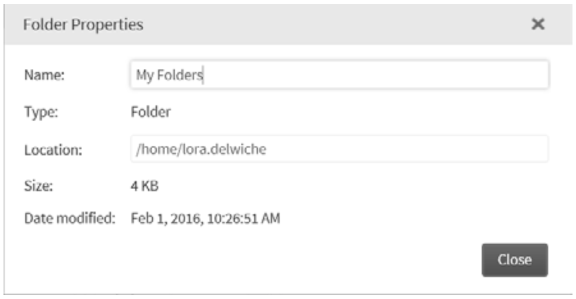
Because the SAS OnDemand for Academics version of SAS Studio runs on Linux servers, all file references must use the UNIX style. The following shows the Server Files and Folders section of the navigation pane.



There is a Folder Shortcuts section where you can set up your own shortcuts to files stored on the server. The Files (Home) section is the storage location assigned to you when your OnDemand account was created. To find the UNIX style path for your storage location (home folder), open the Folder Properties window by right-clicking **Files(Home)** and selecting **Properties**.



The path for your home folder is listed in the Location field of the Properties window. In the following example the path for this user's home folder is /home/lora.delwiche.



Name: My Folders

Type: Folder

Location: /home/lora.delwiche

Size: 4 KB

Date modified: Feb 1, 2016, 10:26:51 AM

Examples When specifying file names and paths in INFILE and LIBNAME statements, use UNIX style paths. For example, if you have a file called President.dat in your home folder and the path to your home folder is /home/myuserid, then the INFILE statement for that file would be

```
    INFILE '/home/myuserid/President.dat';
```

Or you could use the ~ shortcut to your home folder as in

```
    INFILE '~/President.dat';
```

Here are examples of LIBNAME statements that define a SAS library named PLANTS that points to the MySASLib folder created in your home folder:

```
LIBNAME plants '/home/myuserid/MySASLib';

LIBNAME plants '~/MySASLib';
```

Here are examples of how you would reference a SAS data set named MAGNOLIA in the MySASLib folder of your home folder using direct referencing.

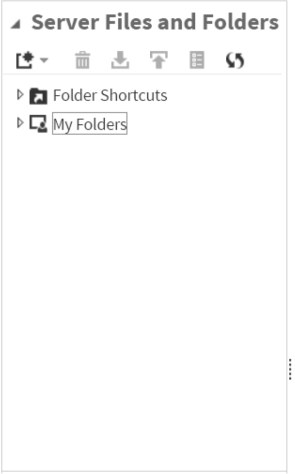
```
DATA '/home/myuserid/MySASLib/magnolia';

DATA '~/MySASLib/magnolia';
```

It is important to remember that file and folder names in UNIX-based environments are case sensitive. That means that the file President.dat is not the same as president.dat. If you are getting messages saying SAS cannot find your files, make sure the case matches.

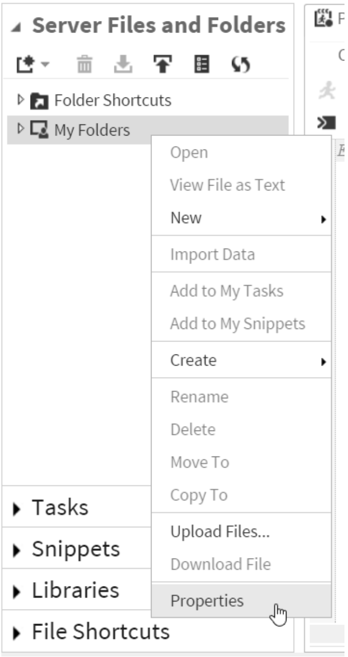
SAS University Edition

The SAS server for SAS University Edition runs on a virtual Linux machine installed on your local computer. You open SAS Studio in a browser on your local computer, and SAS Studio connects to the virtual machine. Think of the virtual machine as a box sitting inside your computer. The SAS server sitting inside that box can't see any files on your local computer unless you share the folder containing the files with the virtual machine. When you set up the virtual machine on your computer and install SAS University Edition, you set up one shared folder by default. The default shared folder is named myfolders and is stored in the SASUniversityEdition folder. You create these folders when you install SAS University Edition. Files that you want to use with SAS University Edition should be stored in the myfolders folder. It is possible to set up more than one shared folder, but this has to be done in the virtualization software (such as VMware Player or Oracle VirtualBox).

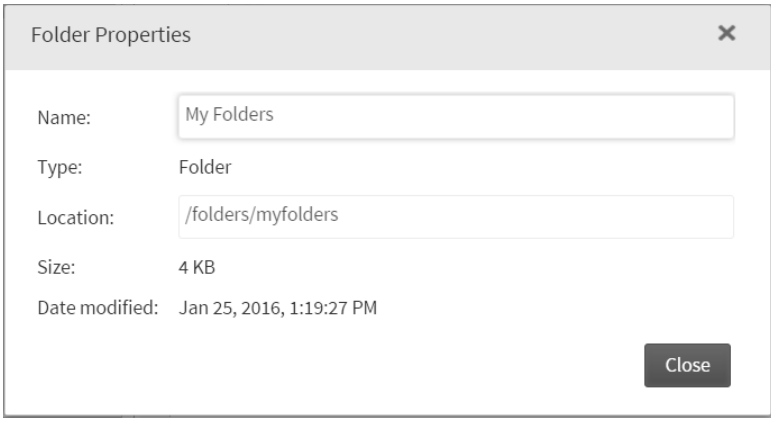
The following shows the Server Files and Folders section of the navigation pane in SAS University Edition.



The My Folders folder is linked to the default shared folder that was set up when you installed SAS University Edition. But because SAS University Edition runs on a virtual Linux machine, when you reference files stored in the myfolders folder, you need to use UNIX style paths. The UNIX style path that SAS Studio uses for this folder is /folders/myfolders. You can see this when you open the Folder Properties window by right-clicking **My Folders** and selecting **Properties**.



The Folder Properties window shows that the location for My Folders is /folders/myfolders.


Folder Properties

Name:	My Folders
Type:	Folder
Location:	/folders/myfolders
Size:	4 KB
Date modified:	Jan 25, 2016, 1:19:27 PM

Note that this is not the path (such as C:\SASUniversityEdition\myfolders for Windows) that you would use to access the same files from your local computer. This is because you are running in a Linux environment once you are inside the virtual machine and SAS Studio has mapped the path /folders/myfolders to your myfolders folder.

Examples for files in myfolders When specifying file names and paths in INFILE and LIBNAME statements, use UNIX style paths. For example, if you have a file called President.dat in your myfolders folder, then the INFILE statement for that file would be

```
INFILE '/folders/myfolders/President.dat';
```

Here is an example of a LIBNAME statement that defines a SAS library named PLANTS that points to the MySASLib folder created in your myfolders folder:

```
LIBNAME plants '/folders/myfolders/MySASLib';
```

Here is an example of how you would reference a SAS data set named MAGNOLIA in the MySASLib folder of your myfolders folder using direct referencing.

```
DATA '/folders/myfolders/MySASLib/magnolia';
```

In SAS University Edition, file and path names may be case sensitive depending on your installation. That means that the file President.dat is not the same as president.dat. If you are getting messages saying SAS cannot find your files, make sure the case matches.

Example for files in other shared folders If you set up additional shared folders through your virtualization software, then those are accessed through folder shortcuts. The path to those folders is /folders/myshortcuts/*shortcutname* where *shortcutname* is the name you gave the shared folder in the virtualization software.

For example, if you set up a shared folder and gave it the name MyRawData, then the INFILE statement to read a file named President.dat in that folder would be:

```
INFILE '/folders/myshortcuts/MyRawData/President.dat';
```

If you are unsure of the path for any folder shortcut you can always find the location in the Folder Properties window for the folder shortcut.