Introductory Tutorial

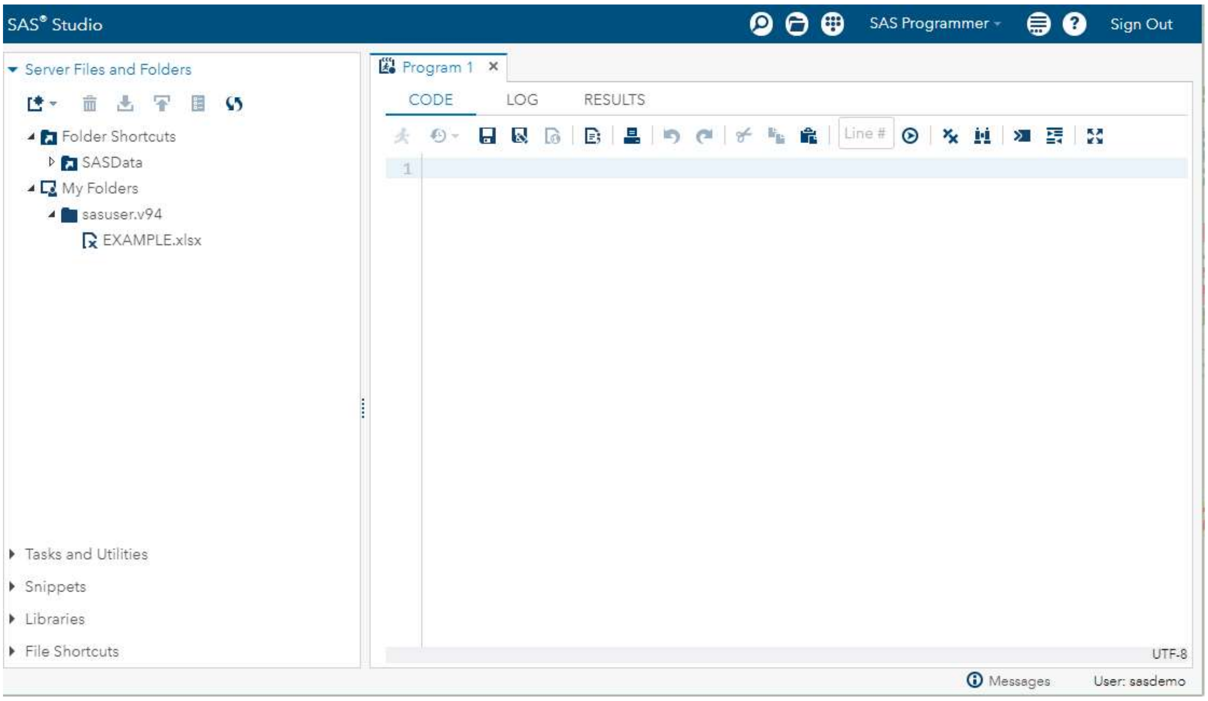
1. Introduction to SAS Studio

In this introductory tutorial the aim is for you to familiarise yourself with the SAS Studio Environment. This tutotial, “Getting Started with SAS Studio”, has been edited from the following tutorial video provided by SAS:

https://www.youtube.com/watch?v=pE5awNW53z8&list=PLVBcK_IpFVi9cajJtRel2uBLbtcLz-WIN&index=6.

You will learn how to view and use data tables, use tasks to quickly generate graphs and statistical analyses and also view the log and code associated with these tasks.

When your first open SAS Studio, you see a navigation pane on the left and a work area on the right as shown below.



The navigation pane provides access to search functionality, folders, and the libraries that contain your files and tables.

When you open data and run tasks, other windows appear in the work area.

Let’s start by viewing a SAS table. Sample data is installed with SAS Studio. Select Libraries, and then expand the SASHELP library. Double-click the **CARS** data table to open it in the table viewer on the right.

The **CARS** table contains 15 columns and 428 rows of data. The table viewer displays one page of data at a time, each page containing 100 rows. You can use the arrows to navigate between pages.

In the columns area of the table viewer, all of the columns are selected by default. You can clear the check box for a column to remove that column from the viewer.

If you'd like to customize the view, you can hide the Columns area, maximise the table viewer, and resize columns as needed. You can also right –click a column heading to filter and sort the data by that column.

If you add a filter, you will be able to see the number of filtered rows at the top. Have a play around with these and see what outcomes you can generate.

a) How many Cars had the Make, Audi (set Make="Audi")?
19

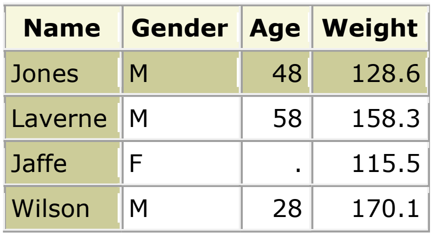
b) Of these, how many had Type equal to Sedan (you will need the AND command)?
13

c) How many cars originated in Asia or the USA (you will need the OR command)?
428

d) How many cars had an MSRP greater than $40,000?
102

SAS enables you to import many data files to create SAS data sets. However, SAS requires all data that it accesses to be in table format. A table is a rectangular arrangement of rows (also called observations) and columns (also called variables).

In the following table, the values Jones, M, 48, and 128.6 make up a single **row** in the table. The values Jones, Laverne, Jaffe, and Wilson make up the **column** Name.



Name	Gender	Age	Weight
Jones	M	48	128.6
Laverne	M	58	158.3
Jaffe	F	.	115.5
Wilson	M	28	170.1

Columns have properties such as name, type, length, label, informat and format. You can view column properties by selecting a column in the Columns window for a data file.

As you will see later, a column's type is important because it affects how the column can be used in a SAS task. A column's type can be either character or numeric.

Character variables, such as Make and Model in the CARS data set, can contain any values. Missing character values are represented by a blank.

Numeric variables, such as Engine Size and Cylinders, can contain only numeric values. Currency, date, and time data are stored as numeric variables. Missing numeric values are represented by a period (.).

2. Querying Data

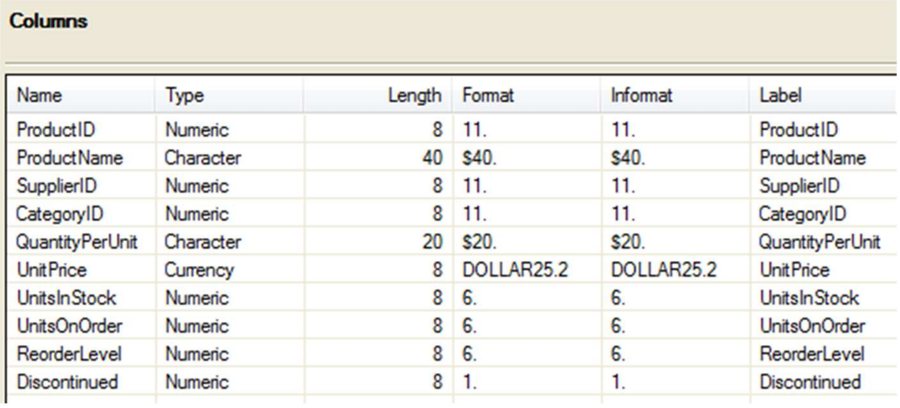
In the following steps you will require two SAS data sets which can be found on in our course Library.

To add this library, select **Libraries, My Libraries**, and then right click to select **New Library**. Enter a name, MS4S08_1 and then the path for this course which is:
~/my_shared_file_links/penny.holborn0

You should now be able to access and view data within this library. The **Products** data set contains information about the products sold by a specialty foods store, and the **Orders** data set contains information about customer orders.

Columns in a SAS data set have specific properties, including the name, type, length, format, informat, and label. It can be helpful to know something about the columns in your data before you use the data in tasks and queries.

View the columns of the **Products** data to see the information as summarised below.

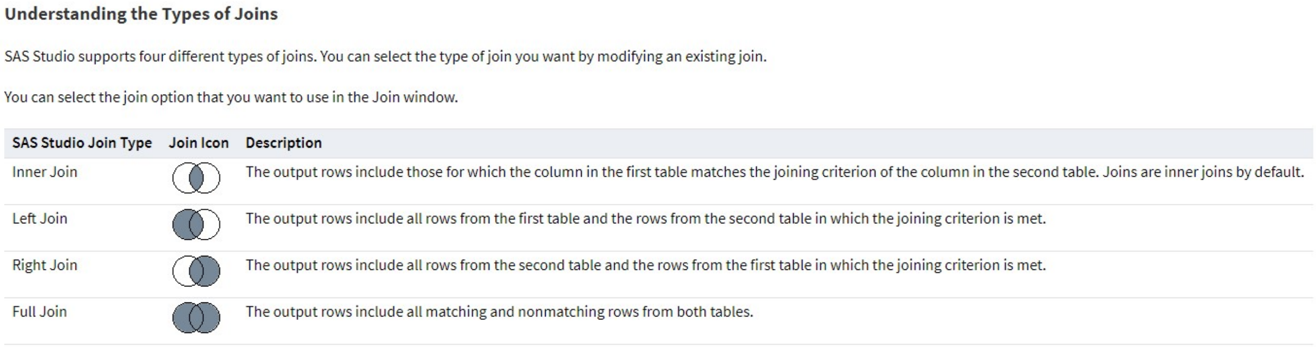


Columns

Name	Type	Length	Format	Informat	Label
ProductID	Numeric	8	11.	11.	ProductID
ProductName	Character	40	$40.	$40.	ProductName
SupplierID	Numeric	8	11.	11.	SupplierID
CategoryID	Numeric	8	11.	11.	CategoryID
QuantityPerUnit	Character	20	$20.	$20.	QuantityPerUnit
UnitPrice	Currency	8	DOLLAR25.2	DOLLAR25.2	UnitPrice
UnitsInStock	Numeric	8	6.	6.	UnitsInStock
UnitsOnOrder	Numeric	8	6.	6.	UnitsOnOrder
ReorderLevel	Numeric	8	6.	6.	ReorderLevel
Discontinued	Numeric	8	1.	1.	Discontinued

You need to join the following tables:

- **Products** - a SAS data set that contains product information such as name, price, and quantity of stock. It also contains category identification numbers.
- **Orders** - a SAS data set that contains data from customer orders.



Understanding the Types of Joins

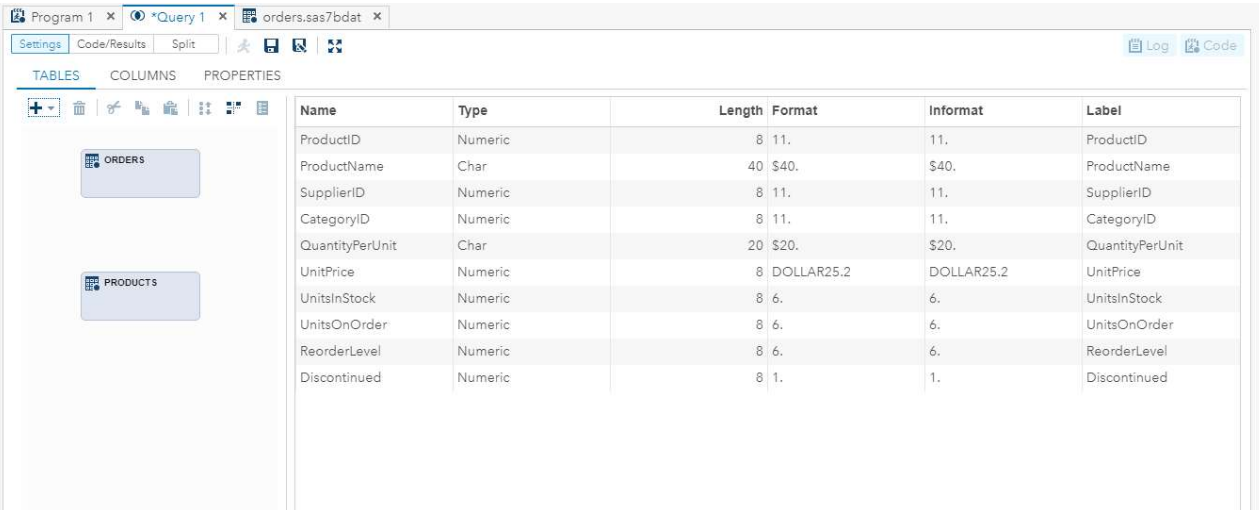
SAS Studio supports four different types of joins. You can select the type of join you want by modifying an existing join.

You can select the join option that you want to use in the Join window.

SAS Studio Join Type	Join Icon	Description
Inner Join	◯	The output rows include those for which the column in the first table matches the joining criterion of the column in the second table. Joins are inner joins by default.
Left Join	◐	The output rows include all rows from the first table and the rows from the second table in which the joining criterion is met.
Right Join	◑	The output rows include all rows from the second table and the rows from the first table in which the joining criterion is met.
Full Join	●	The output rows include all matching and nonmatching rows from both tables.

Under the Utilities window, select **Query**.

Click **+ button** on the toolbar and select **Table**. Select both data sets.



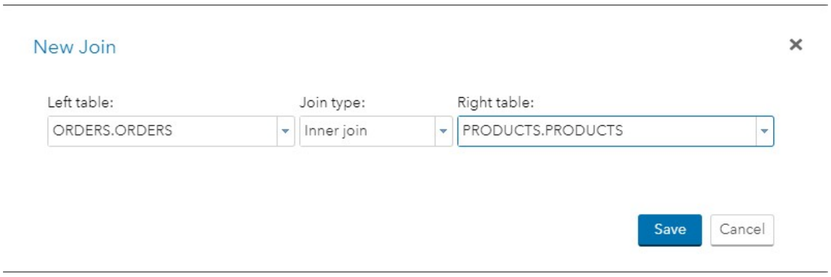
Name	Type	Length	Format	Informat	Label
ProductID	Numeric	8	11.	11.	ProductID
ProductName	Char	40	$40.	$40.	ProductName
SupplierID	Numeric	8	11.	11.	SupplierID
CategoryID	Numeric	8	11.	11.	CategoryID
QuantityPerUnit	Char	20	$20.	$20.	QuantityPerUnit
UnitPrice	Numeric	8	DOLLAR25.2	DOLLAR25.2	UnitPrice
UnitsInStock	Numeric	8	6.	6.	UnitsInStock
UnitsOnOrder	Numeric	8	6.	6.	UnitsOnOrder
ReorderLevel	Numeric	8	6.	6.	ReorderLevel
Discontinued	Numeric	8	1.	1.	Discontinued

Because the Orders table does not contain a variable that has a name that the Query Builder can match in the Products table, you must perform a manual join.

It is known that the **PRODID** field in the Orders data is equivalent to the **ProductID** field in the Product data set.

On the Tables tab of the query window, click the **Add button** on the toolbar again and select Join. The New Join window opens.

Select and **Inner Join** between both tables as follows and click **Save**.



New Join

Left table: ORDERS.ORDERS

Join type: Inner join

Right table: PRODUCTS.PRODUCTS

You can then select the two fields to join as above.

Now we must decide what fields to include in the newly joined data set. Under the **COLUMNS** tab select to keep all previous fields by dragging them over to the selection pane.

You can now select **Run** and view the **OUTPUT DATA**.

By going back to the Query tab and Settings, you can easily change your Query.

Under the **PROPERTIES** tab, change the location to save your newly generated table.

```
%web_drop_table(MyCustom.OrdersProducts);

/* Query code generated for SAS Studio by Common Query Services */

PROC SQL;

CREATE TABLE MyCustom.OrdersProducts

AS

SELECT ORDERS.ORDERID, ORDERS.PRODID, ORDERS.UPRICE, ORDERS.QUANTITY,
PRODUCTS2.ProductID, PRODUCTS2.ProductName, PRODUCTS2.SupplierID, PRODUCTS2.CategoryID,
PRODUCTS2.QuantityPerUnit, PRODUCTS2.UnitPrice, PRODUCTS2.UnitsInStock,
PRODUCTS2.UnitsOnOrder, PRODUCTS2.ReorderLevel, PRODUCTS2.Discontinued

FROM MS4S08_1.ORDERS ORDERS

INNER JOIN MS4S08_1.PRODUCTS2 PRODUCTS2

ON

   ( ORDERS.PRODID = PRODUCTS2.ProductID ) ;

QUIT;

%web_open_table(MyCustom.OrdersProducts);
```

3. Creating a bar chart

In these steps you use the Bar Chart task and select the type of bar chart that you want to create.

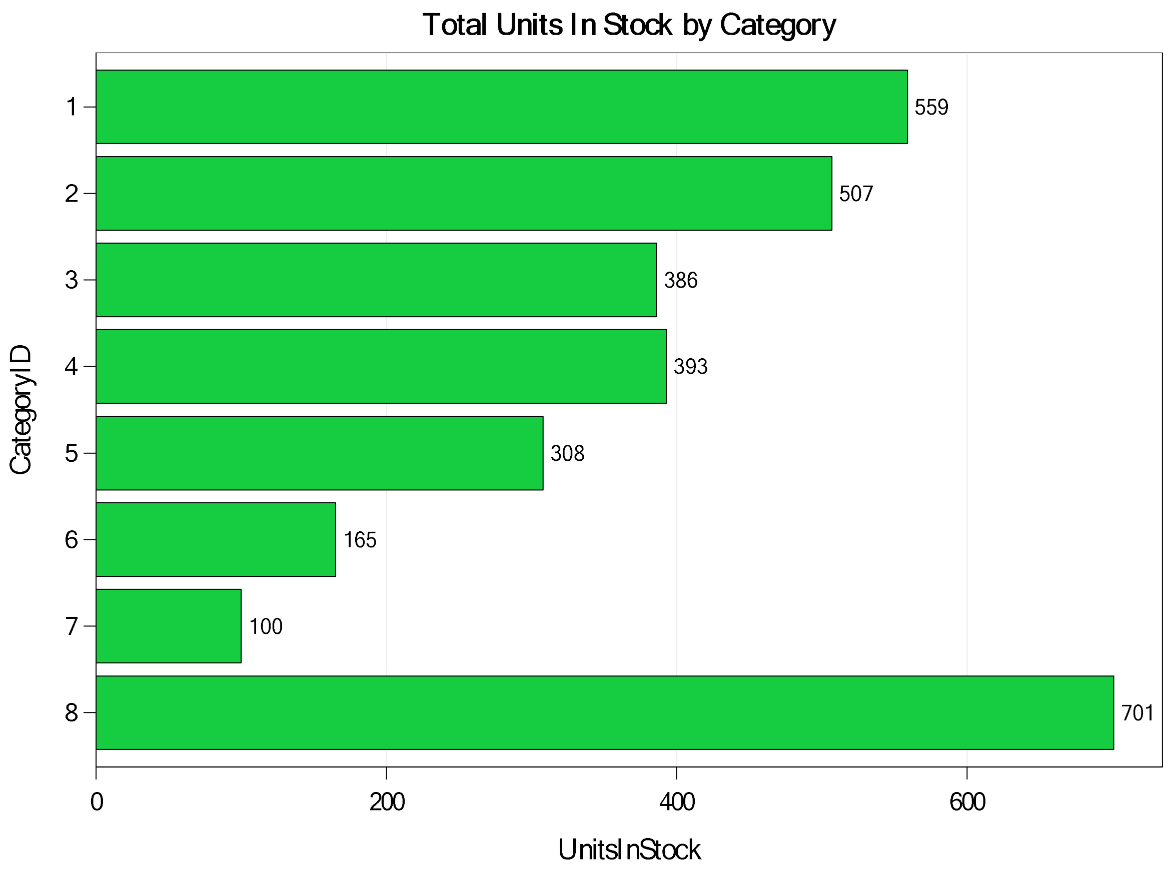
1. From the **My Tasks** menu, select **Tasks** then **Graph** then **Bar Chart**.
2. Under **Data**, select the **Products** data set.

Next you assign columns to roles.

3. Select **CategoryID** and as the **Category variable** role.
4. Select **UnitsInStock** as the measure variable (select Measure = Variable, then select UnitsInStock).

Click **Run** to create the graph.

Play around with the OPTIONS tab to create the chart below and then export it to Microsoft Word.


Total Units In Stock by Category
CategoryID
UnitsInStock