THE POWER TO KNOW®

Getting Started with SAS® Studio on SAS® Viya® (Quickstart)

SAS® Tutorial

SAS® Tutorial | Getting Started with SAS® Studio on SAS® Viya® (Quickstart) was developed by Luna Bozeman. Additional contributions were made by Anita Hillhouse, Amy Peters, and Stacey Syphus. Instructional design, editing, and production support was provided by the Learning Design and Development team.

SAS® Tutorial | Getting Started with SAS® Studio on SAS® Viya® (Quickstart)

Course code ESTUGS0C, prepared date 18May2021. ESTUGS0C_001

Getting Started with SAS Studio on SAS Viya

1.1 SAS Studio Demonstrations

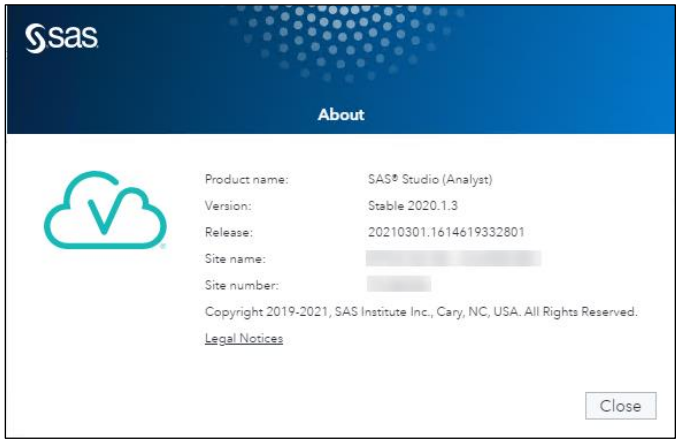
This section contains several demonstrations that use SAS Studio. Here are the topics:

- using programming features in SAS Studio
- working with data in a flow
- using SAS Studio tasks

The demonstrations use data readily available in the **Sashelp** library. However, the starter program and CSV file can be downloaded through the SAS Communities GitHub page.

The demonstration steps are written for SAS Viya 2020.1.3, but they should work for SAS Viya 2020.1.1 and later. The demo has been tested on SAS Viya 2020.1.4, but note that there might be some minor visual differences.

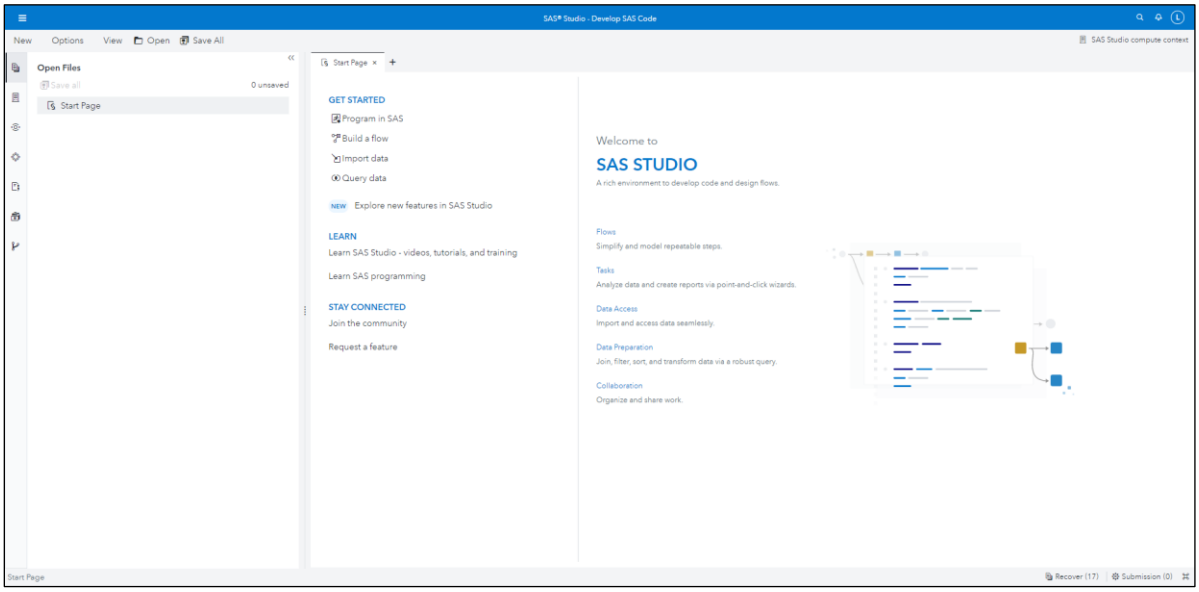
To view your version of SAS Viya, after you sign in, click the user button at the end of the application bar and select **About**.



If you do not have a license for SAS Studio Analyst, for **Product name**, you would see just **SAS Studio**. A license for SAS Studio Analyst is ***not*** required for any of the demonstrations.

Exploring Programming Features

1. Sign in to SAS Studio. The main window of SAS Studio consists of a navigation pane on the left and a work area on the right.



   a. The navigation pane provides easy access to your open files, your folder shortcuts, the file system, SAS content, steps, your tasks and snippets, the libraries that you have access to, your Git repositories, and your file references. The sections that you see depend on settings set by your administrator as well as what is selected in the View menu. These sections are explored throughout the tutorial.

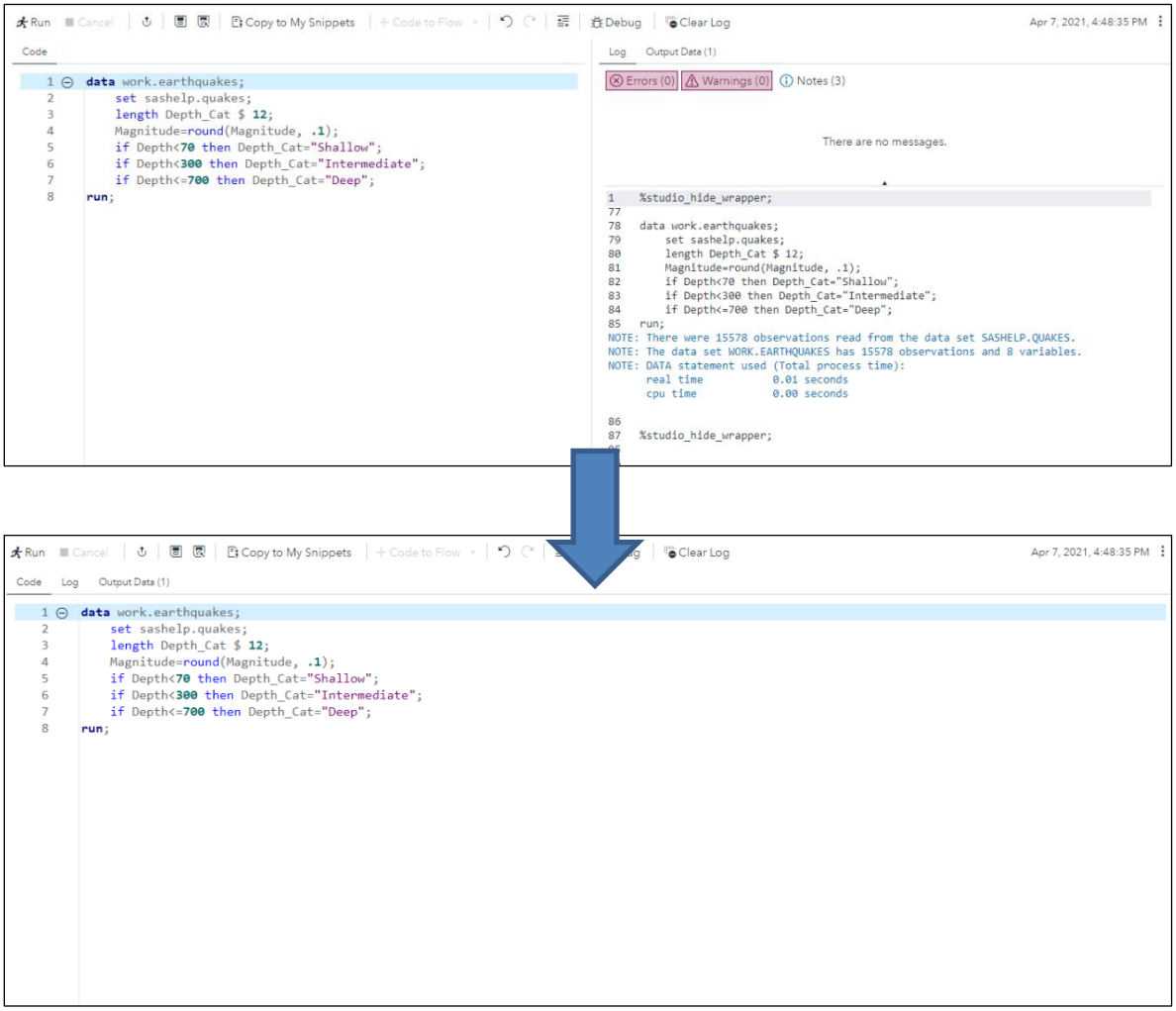
   b. The work area is used to display your data, code, tasks, logs, results, and flows. As you open these items, the windows appear as new tabs in the work area. When you first open SAS Studio, the Start Page tab appears in the work area by default so that you can quickly get started writing a new SAS program, building a flow, importing data, or creating a query.

2. The Explorer section in the navigation pane enables you to access files and folders from your folder shortcuts, your server file system, and your SAS Content Server locations, if available. In the Explorer section, navigate to and then double-click the **Earthquake Depth Categories.sas** program to open the program in a new tab in the work area.

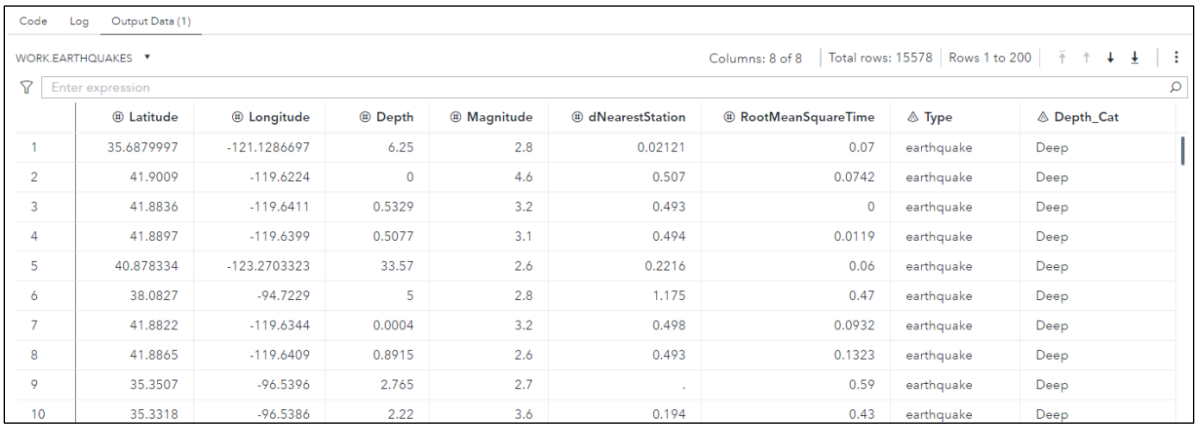
   The goal of this program is to categorize the earthquakes in the **quakes** table from the **Sashelp** library by the depth at which the earthquake occurred. In addition, the magnitude values will be rounded to the nearest decimal point. The results are stored in a new table named **earthquakes** in the **Work** library.

```
data work.earthquakes;
    set sashelp.quakes;
    length Depth_Cat $ 12;
    Magnitude=round(Magnitude, .1);
    if Depth<70 then Depth_Cat="Shallow";
    if Depth<300 then Depth_Cat="Intermediate";
    if Depth<=700 then Depth_Cat="Deep";
run;
```

3. To run the program, click **Run** or press the F3 key on your keyboard. Notice that the default tab layout is **Vertical split**, where the Code tab is on the left, and the Log and (if applicable) the Results and Output Data tabs, are in a tab group on the right. On the program toolbar, click ⋮ (**More options**) and select **Tab layout** ⇨ **Single** to group the tabs into a single tab group.

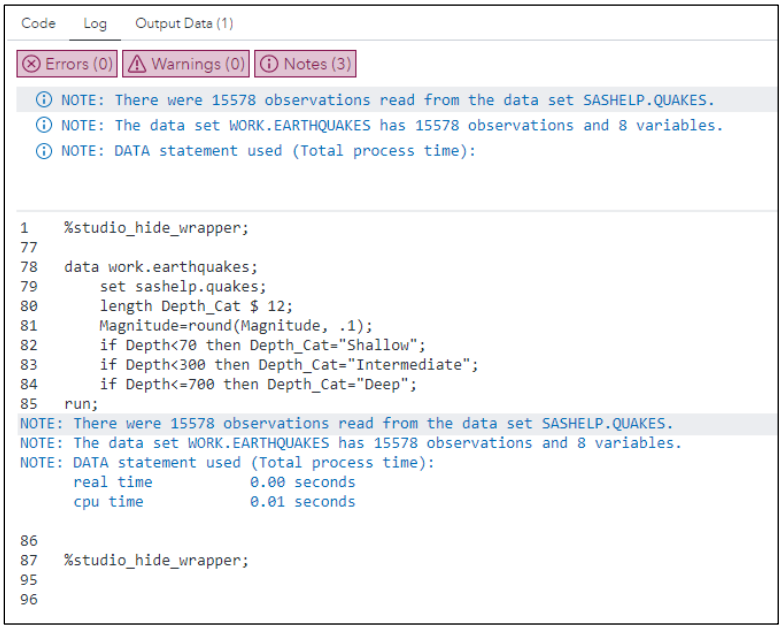


4. Click the **Output Data** tab to view the output **work.earthquakes** table. Notice that **Depth_Cat** has a value of *Deep* for every earthquake.



	Latitude	Longitude	Depth	Magnitude	dNearestStation	RootMeanSquareTime	Type	Depth_Cat
1	35.6879997	-121.1286697	6.25	2.8	0.02121	0.07	earthquake	Deep
2	41.9009	-119.6224	0	4.6	0.507	0.0742	earthquake	Deep
3	41.8836	-119.6411	0.5329	3.2	0.493	0	earthquake	Deep
4	41.8897	-119.6399	0.5077	3.1	0.494	0.0119	earthquake	Deep
5	40.878334	-123.2703323	33.57	2.6	0.2216	0.06	earthquake	Deep
6	38.0827	-94.7229	5	2.8	1.175	0.47	earthquake	Deep
7	41.8822	-119.6344	0.0004	3.2	0.498	0.0932	earthquake	Deep
8	41.8865	-119.6409	0.8915	2.6	0.493	0.1323	earthquake	Deep
9	35.3507	-96.5396	2.765	2.7	.	0.59	earthquake	Deep
10	35.3318	-96.5386	2.22	3.6	0.194	0.43	earthquake	Deep

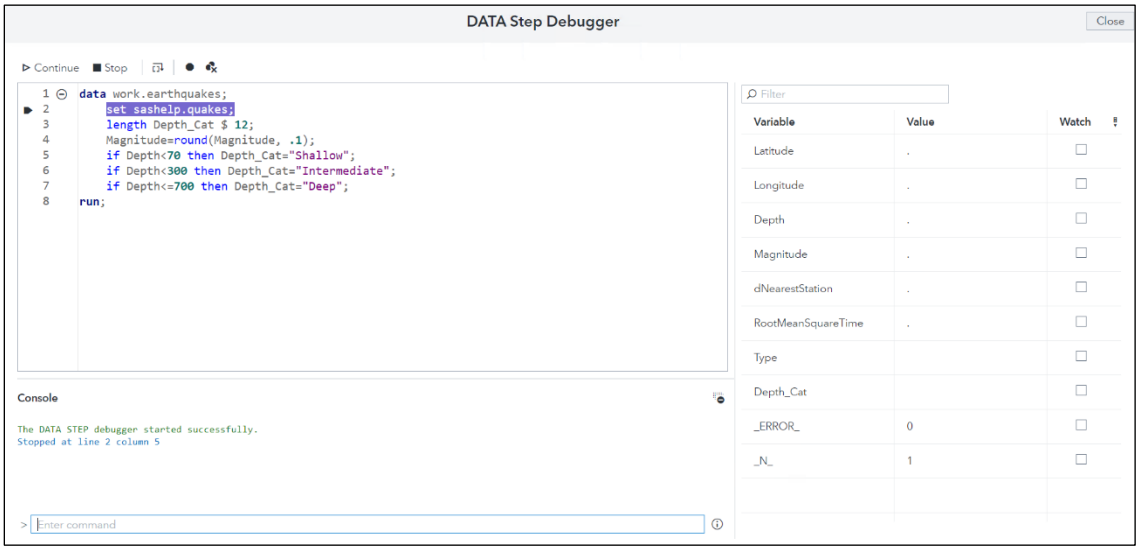
5. Click the **Log** tab to view messages returned from SAS. If necessary, select the **Errors**, **Warnings**, and **Notes** sections to display all messages in the log summary. To see the corresponding full message in the log, select the message in the log summary. There are only three notes, none of which indicate a syntactical error in the program.



This is an example of a logic error—that is, an error that doesn't stop the program from running but produces unexpected results. Use the DATA Step Debugger to understand why the program isn't working as expected.

6. The DATA Step Debugger is a tool that enables you to step through the execution of a DATA step to find logic errors. To enable the DATA Step Debugger, return to the Code tab. Then, on the program toolbar, click **Debug**. All sections of DATA step code in the program are highlighted with a green bar in the margin to indicate that they can be debugged.

   a. Click (**DATA step markers for debugging**) to open the DATA Step Debugger window. The code is on the left, and the currently executing line is highlighted in purple. The list of columns and their current values are displayed on the right.



b. Click (**Step execution for next line**) to execute the highlighted SET statement. The first row from the **quakes** table is read in and the values are displayed to the right. Changes in the column values are displayed in red.

   The LENGTH statement is skipped because it is a compile-time-only statement, which sets the length of **Depth_Cat** to 12 to ensure that the values do not get truncated.

c. The current **Magnitude** value is *2.75*. Click (**Step execution for next line**) to execute the assignment statement to round the **Magnitude** value to the nearest decimal point. The **Magnitude** column is overwritten with a value of *2.8*.

d. The current **Depth** value is *6.25*, which is less than 70. Therefore, the condition in the first IF-THEN statement is true. Click (**Step execution for next line**) twice to execute the statement. The **Depth_Cat** column is assigned a value of *Shallow*.

e. The second IF-THEN statement is highlighted. Again, the current value of **Depth** is *6.25*, which is less than 300. Therefore, the condition in the second IF-THEN statement is also true. Click (**Step execution for next line**) twice to execute the statement. The **Depth_Cat** column is overwritten with a value of *Intermediate*.

f. Similarly, the condition in the third IF-THEN statement is also true. Click (**Step execution for next line**) twice to execute the statement. The **Depth_Cat** column is overwritten with a value of *Deep*.

   When you have multiple IF-THEN statements, SAS tests all conditions in sequence for every row. The last true condition executes the statement that determines the value in the output table. In this example, this means that any earthquake with a **Depth** value less than or equal to *700* is assigned a **Depth_Cat** value of *Deep*. Instead, the conditions were intended to be treated as a hierarchy so that when a true condition is found, SAS executes the statement following the THEN keyword and skips the subsequent IF-THEN statements. To enforce sequential testing, the ELSE keyword can be used in front of the IF-THEN statements, except for the first IF-THEN statement.

g. Click **Close**.

7. On the Code tab, add the ELSE keyword in front of the second and third IF-THEN statements.

```
data work.earthquakes;
    set sashelp.quakes;
    length Depth_Cat $ 12;
    Magnitude=round(Magnitude, .1);
    if Depth<70 then Depth_Cat="Shallow";
    else if Depth<300 then Depth_Cat="Intermediate";
    else if Depth<=700 then Depth_Cat="Deep";
run;
```

8. To verify the effect of the ELSE keyword, click (**DATA step markers for debugging**) to open the DATA Step Debugger window.

   a. Click (**Step execution for next line**) to execute the highlighted SET statement. The first row from the **quakes** table is read in.

   b. Click (**Step execution for next line**) to execute the assignment statement to round the **Magnitude** value to the nearest decimal point.

c. The current **Depth** value is *6.25*, which is less than 70. Therefore, the condition in the first IF-THEN statement is true. Click (**Step execution for next line**) twice to execute the statement. The **Depth_Cat** column is assigned a value of *Shallow*. Because a true condition was met, and the ELSE keyword is used with the remaining conditional processing statements, those statements are skipped.

d. You can continue to use the DATA Step Debugger to process all rows from **sashelp.quakes**. Click **Close**.

9. On the program toolbar, click **Debug** to suppress the debugger icon and the green bar in the margin.

10. Rerun the program. Click the **Output Data** tab to view the updated **earthquakes** table.

WORK.EARTHQUAKES — Columns: 8 of 8 | Total rows: 15578 | Rows 1 to 200

Enter expression

	Latitude	Longitude	Depth	Magnitude	dNearestStation	RootMeanSquareTime	Type	Depth_Cat
1	35.6879997	-121.1286697	6.25	2.8	0.02121	0.07	earthquake	Shallow
2	41.9009	-119.6224	0	4.6	0.507	0.0742	earthquake	Shallow
3	41.8836	-119.6411	0.5329	3.2	0.493	0	earthquake	Shallow
4	41.8897	-119.6399	0.5077	3.1	0.494	0.0119	earthquake	Shallow
5	40.878334	-123.2703323	33.57	2.6	0.2216	0.06	earthquake	Shallow
6	38.0827	-94.7229	5	2.8	1.175	0.47	earthquake	Shallow
7	41.8822	-119.6344	0.0004	3.2	0.498	0.0932	earthquake	Shallow
8	41.8865	-119.6409	0.8915	2.6	0.493	0.1323	earthquake	Shallow
9	35.3507	-96.5396	2.765	2.7	.	0.59	earthquake	Shallow
10	35.3318	-96.5386	2.22	3.6	0.194	0.43	earthquake	Shallow

11. Right-click the **Depth** column and select **Sort** ⇨ **Descending**. Although none of the earthquakes fall under the *Deep* category, several fall under the *Intermediate* category.

    Sorting the table in the table viewer does ***not*** change the sort order of the table. It simply sorts the current view of the table. Any customizations that are applied in the table viewer are not saved with the table.

WORK.EARTHQUAKES — Columns: 8 of 8 | Total rows: 15578 | Rows 1 to 200

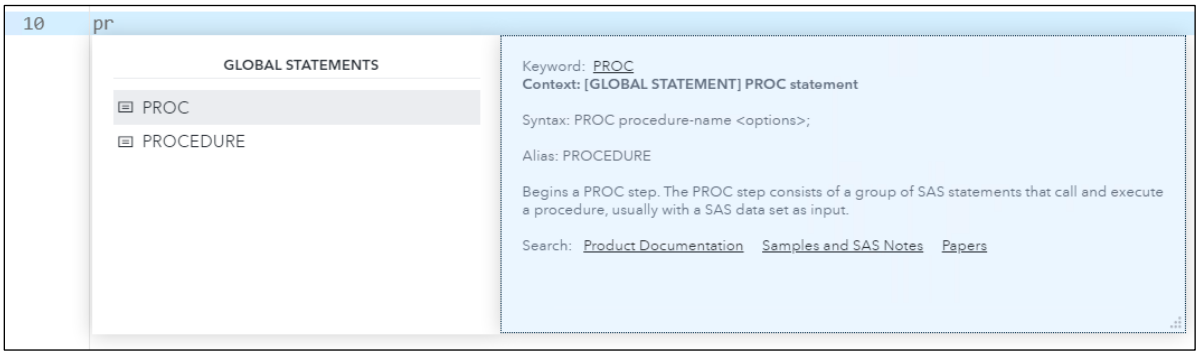
Enter expression

	Latitude	Longitude	Depth ↓	Magnitude	dNearestStation	RootMeanSquareTime	Type	Depth_Cat
1	30.132	-114.705	104.3	4	.	2.45	earthquake	Intermediate
2	42.9745	-109.128	76.2	4.8	0.378	1.08	earthquake	Intermediate
3	42.9944	-109.1146	71.43	3	0.397	0.4	earthquake	Intermediate
4	40.463	-122.214	71	2.8	.	.	earthquake	Intermediate
5	40.0446667	-122.0863333	69.78	2.5	0.1663	0.26	earthquake	Shallow
6	39.8405	-122.0373	62.3	2.5	0.22457882	0.13	earthquake	Shallow
7	36.068	-120.0122	62	2.6	0.08084838	0.18	earthquake	Shallow
8	48.194	-122.634	59.2	3.7	.	.	earthquake	Shallow
9	47.772	-122.557	58.3	4.5	.	.	earthquake	Shallow
10	47.1013	-122.737	58.2	3.5	0.1706799	0.2	earthquake	Shallow

12. To remove the sort, right-click the **Depth** column again and select **Sort** ⇨ **Remove sort**.

13. Return to the Code tab to add a PROC FREQ step to create one-way frequency reports.

14. At the end of the program, begin typing **pr**. Notice that an autocomplete window appears with a list of suggested keywords, and **PROC** is highlighted. The Syntax Help also appears, corresponding to the highlighted keyword with a description of the keyword as well as links to Product Documentation, Samples and SAS Notes, and Papers. You can also hover over any blue keyword in a program to see the Syntax Help for that corresponding keyword. Double-click **PROC** in the autocomplete window to enter the **proc** keyword in the program.



15. Press the spacebar. The autocomplete window appears with a list of procedure names. Type **fr** and press the Enter key to add the **freq** keyword to the program.
16. Press the spacebar. The autocomplete window appears again with a list of valid options for a PROC FREQ statement. Type a **d** and press the Enter key to enter the DATA= option. The autocomplete window displays the output tables referenced in the program. Double-click **work.earthquakes** in the autocomplete window to specify it as the input table to the procedure.
17. Press the spacebar to see the list of valid options for the PROC FREQ statement again. Type **n** and then press the Enter key to enter the NLEVELS option to include a table with the number of distinct values for the columns to be analyzed. Type a **;** (semicolon) to end the PROC FREQ statement. The PROC FREQ statement should appear as below:

```
proc freq data=work.earthquakes nlevels;
```

18. Press the Enter key to advance to the next line in the program. A TABLES statement is used to request one-way frequency tables for columns that you specify. Type **ta**, and when the autocomplete window appears, press the down arrow key on your keyboard to highlight the TABLES keyword, and then press the Enter key to enter it in the program. Press the spacebar.
19. Select the **Libraries** section in the navigation pane to view the list of libraries that you have access to. Then expand **WORK** ⇨ **EARTHQUAKES** to see the list of columns in the **earthquakes** table. Select **Depth_Cat**, hold down the Ctrl key, select **Type**, and then drag and drop the columns into the program to add them to the TABLES statement.

    When you drag a column or table name that is also a reserved word in a database, such as **Type**, or a name that does not conform to traditional SAS naming conventions, SAS Studio automatically encloses the column or table name in quotation marks and adds a lowercase letter **n** to the end to ensure that the name is evaluated correctly by the program. This is known as a *SAS name literal*.

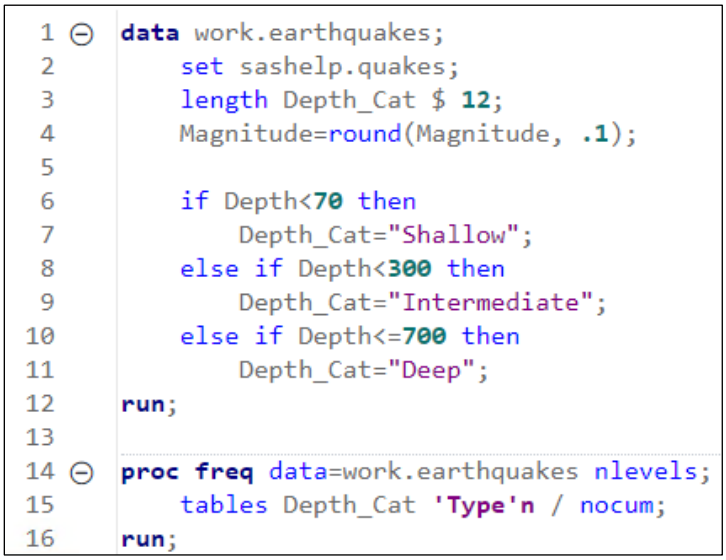
```
10 ⊖ proc freq data=work.earthquakes nlevels;
11    tables Depth_Cat 'Type'n
```

20. Type a space, a **/** (forward slash), and a space again to view a list of options for a TABLES statement. Type an **n** to filter the list of options in the autocomplete window and then double-click **NOCUM** to include the option in the program. The NOCUM option suppresses the display of cumulative frequencies and percentages. Enter a **;** (semicolon) to end the statement.

21. Enter a RUN statement on the next line to end the step. The final PROC FREQ step should appear as below:

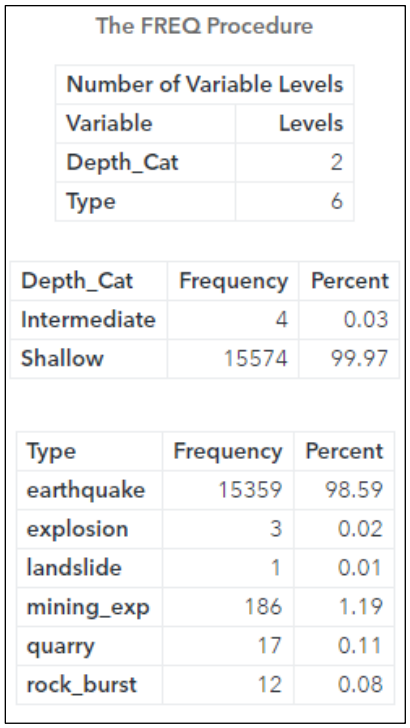
```
proc freq data=work.earthquakes nlevels;
tables Depth_Cat 'Type'n / nocum;
run;
```

22. On the program toolbar, click (**Format code**) to quickly format the entire program.



```
1  data work.earthquakes;
2      set sashelp.quakes;
3      length Depth_Cat $ 12;
4      Magnitude=round(Magnitude, .1);
5
6      if Depth<70 then
7          Depth_Cat="Shallow";
8      else if Depth<300 then
9          Depth_Cat="Intermediate";
10     else if Depth<=700 then
11         Depth_Cat="Deep";
12 run;
13
14 proc freq data=work.earthquakes nlevels;
15     tables Depth_Cat 'Type'n / nocum;
16 run;
```

23. To run just the PROC FREQ step, highlight the step and click **Run** or press the F3 key on your keyboard. The Results tab displays the frequency report.



The FREQ Procedure

Number of Variable Levels	
Variable	Levels
Depth_Cat	2
Type	6

Depth_Cat	Frequency	Percent
Intermediate	4	0.03
Shallow	15574	99.97

Type	Frequency	Percent
earthquake	15359	98.59
explosion	3	0.02
landslide	1	0.01
mining_exp	186	1.19
quarry	17	0.11
rock_burst	12	0.08

24. SAS Viya includes multiple servers to execute SAS code, the two primary servers being the SAS Compute Server and SAS Cloud Analytic Services, or CAS. The code that was submitted was traditional SAS®9 code that you might be used to, and was executed on the SAS Compute Server by default. There is no need to learn new syntax to use the Compute Server. CAS is the high-performance server that performs parallel processing on in-memory data and will likely be used for big data and complex analytics. Often, only very minor code modifications are required in order for programs to run in CAS. In this tutorial, all code is executed on the Compute Server.

    Note: To learn more about running programs on CAS, see the SAS documentation for SAS Viya Programming: Getting Started.

25. To change editor options. including autocomplete, Syntax Help, indention, and more, and to set a default tab layout for all SAS programs, on the main toolbar, select **Options** ⇨ **Preferences**. Under **SAS Programs**, select **Code and Log** and use the **Program tab layout** option to change the default tab layout for all SAS programs. Select **Editors** ⇨ **Editor Options** to change editor options. Click **Cancel** ⇨ **Cancel** without making any changes.

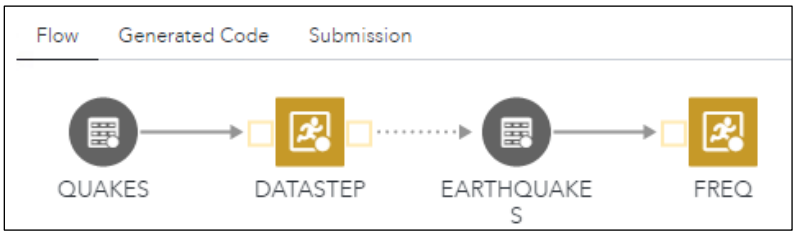
26. To save the program under a different name, on the program toolbar, click (**Save as**). Navigate to and select a folder of your choice, and in the **Name** field, enter **Earthquake Category Frequency**. Select the **Type** drop-down list. You can choose to save the program, the log, or the results, or create a Program Summary page or a SAS Program Package file. A Program Summary page includes information about the program execution, the complete SAS source code, the complete SAS log, and the results. A SAS Program Package contains a snapshot of a SAS program along with its log and HTML results. Use the default **Program** type and click **Save**.

End of Demonstration

Working with Flows

1. A flow in SAS Studio is a visual sequence of operations on data. Data and operations are represented by nodes, and a flow orchestrates these nodes in a series of steps in which the output of one node is the input to another node. Flows can be used to prepare data for reporting and analysis.
2. To start a new SAS session, on the main toolbar, select **Options** ⇨ **Reset SAS Session** ⇨ **Reset**.
3. The previously saved **Earthquake Category Frequency.sas** program can be directly added to a new flow as a SAS program node. However, an alternative is to convert a SAS program to a flow in which the input tables, procedures, and output tables in the program are used to create individual nodes in the flow. To do the latter, on the program toolbar, click ⋮ (**More options**) and select **Create flow from program**. To change the name of the flow, click (**Open**). Navigate to and select a folder of your choice. Then, in the **Name** field, enter **Earthquake Analysis**. Click **OK** ⇨ **OK**.
4. SAS runs the program, and a new flow tab, **Earthquake Analysis**, opens in the work area.

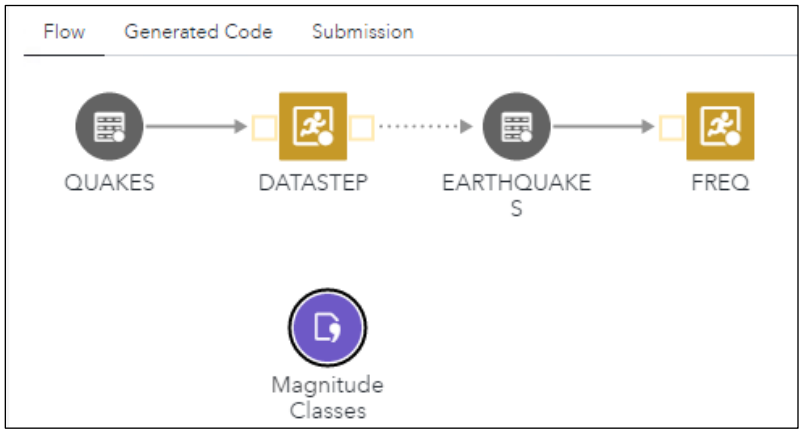
   This flow illustrates that the **quakes** table is used as input to a DATA step to create the **earthquakes** table. Then a PROC FREQ step is used to analyze the **earthquakes** table.



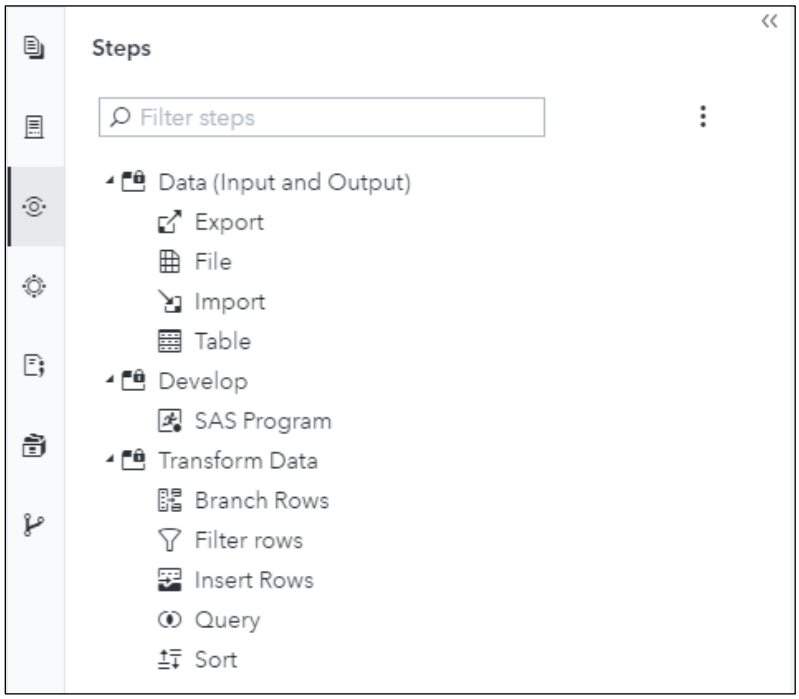
5. The DATASTEP and FREQ nodes are program nodes that contain portions of the original **Earthquake Category Frequency** program. If an existing SAS program is added to a flow, a copy of the code is added to the flow. This means that any changes made to the original SAS program will not affect the code in the flow and vice versa. Select the **FREQ** node to view the code in the node details.
6. Select the **EARTHQUAKES** table node in the flow canvas. The node details display information about the table, such as the properties, the columns, as well as a preview of the data.
7. To further enhance the **earthquakes** table, the table can be joined with a lookup table that maps the magnitudes to descriptive classes. This information is in a CSV file named **Magnitude Classes** and is stored locally, so the file must be uploaded to the server before importing.

```
Magnitude Lower Bound,Magnitude Upper Bound,Class
1,1.9,Micro
2,2.9,Minor
3,3.9,Minor
4,4.9,Light
5,5.9,Moderate
6,6.9,Strong
7,7.9,Major
8,8.9,Great
9,9.9,Great
```

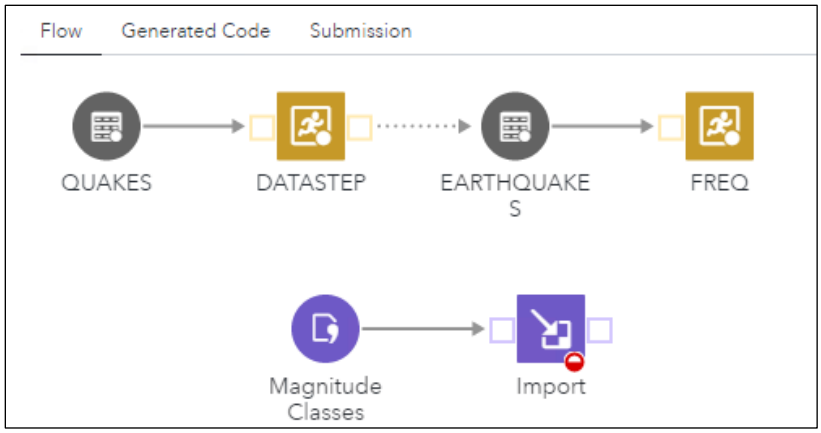
8. In the Explorer section, navigate to and select a folder of interest. Then, on the Explorer toolbar, click (**Upload files**). Click (**Add**) and then navigate to and select the **Magnitude Classes.csv** file. Click **Open** and select **Upload**.
9. Drag the **Magnitude Classes.csv** file from the Explorer section and drop it onto the **Earthquake Analysis** flow canvas.



10. Select the **Steps** section in the navigation pane to view the steps that can be added to a flow as nodes. If your site does not have a license for SAS Studio Analyst, you might not have access to all of the steps shown below.



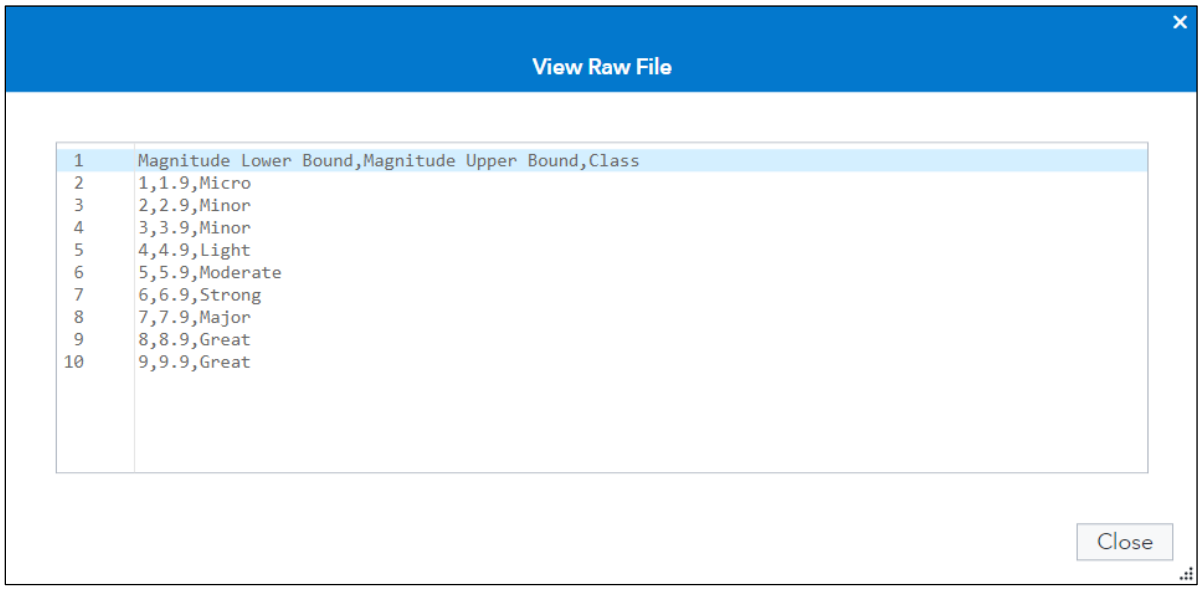
11. To import the file, drag the **Import** step to the right portion of the **Magnitude Classes** file node and drop it when **Connect to output port** appears, or right-click the **Magnitude Classes** file node and select **Add an import**. The **Magnitude Classes** file node is now an input to the Import node. Notice the red unfinished state icon on the Import node. Options and settings will need to be set before the Import node can be run without errors.



12. With the **Import** node selected, in the node details, click (**Maximize preview**).

13. On the Options tab, click (**View raw file**). The first record contains what can be used as column names. The first field contains the magnitude lower bound, followed by the magnitude upper bound. The last field contains the corresponding class description for earthquakes in the specified magnitude range.

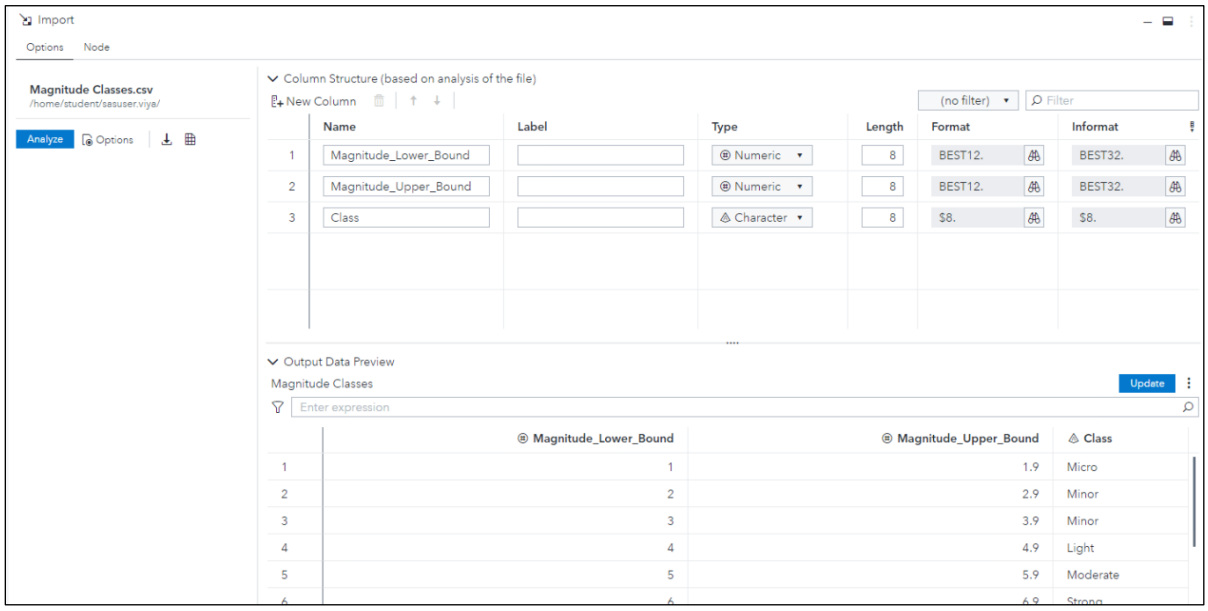
    Click **Close**.



14. Click **Options** to verify the options for the import. Verify that the **Column names are in first row of input file** and **Rename column names to comply with SAS naming conventions** options are selected. The latter replaces the spaces in the column names with underscores. The first row to process should be the second row.

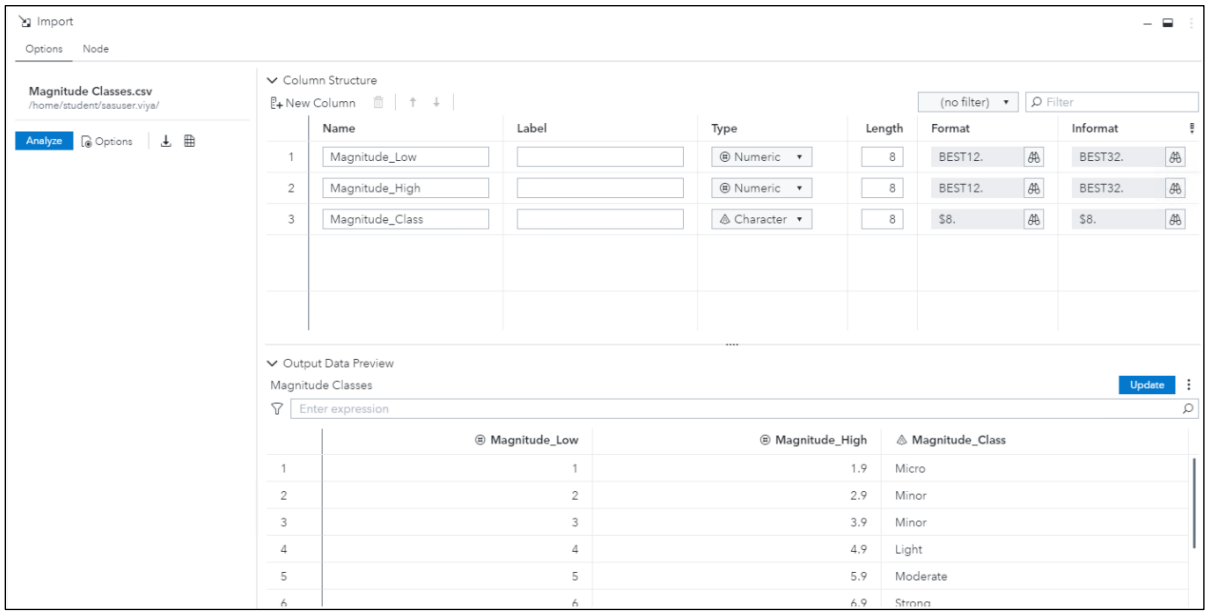
    Click the **Update Options** tab to verify the remaining options and then click **OK**.

15. Click **Analyze** to identify the structure of the file. The three columns are listed with the spaces in the names replaced with underscores. In addition, a preview of the output data is available.



16. Before importing the file, you can make changes to some of the column attributes. Change the column name **Magnitude_Lower_Bound** to **Magnitude_Low**, change **Magnitude_Upper_Bound** to **Magnitude_High**, and change **Class** to **Magnitude_Class**. Verify the remaining attributes.

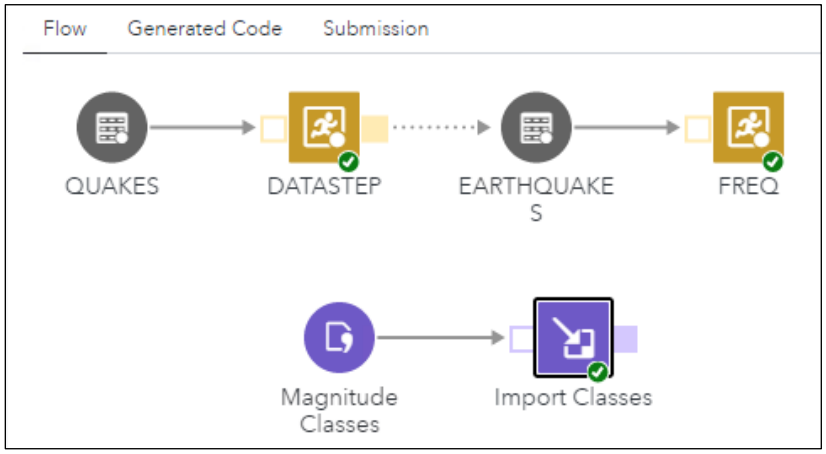
    Click **Update** to view an updated preview of the output data.



17. Click the **Node** tab. In the **Name** box, enter **Import Classes**.
18. Click — (**Minimize preview**).

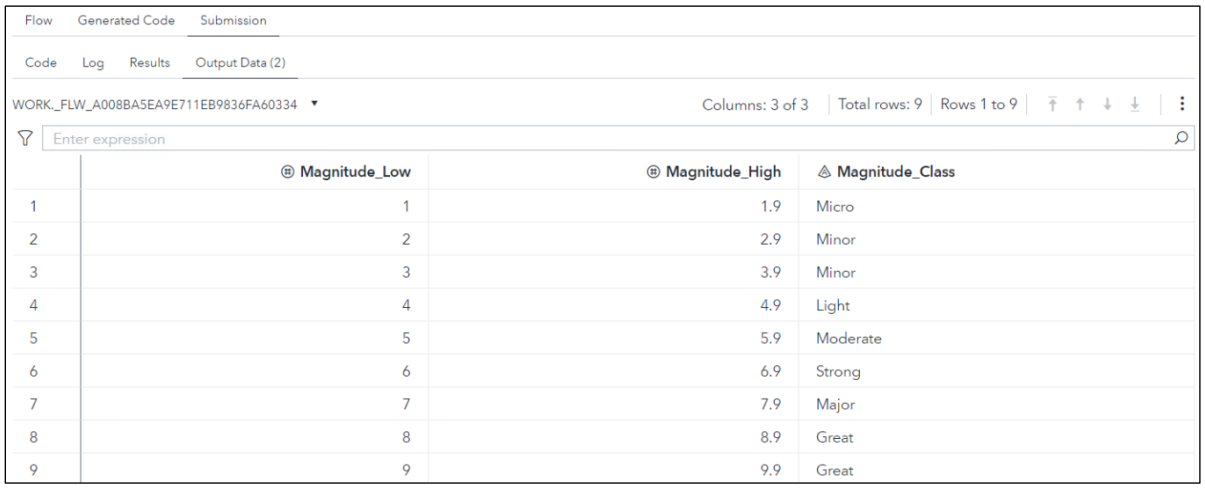
19. On the flow toolbar, click ▷ **Run** to run the entire flow.

    Notice that the output port of the Import node is filled in, indicating that the node ran successfully and that data is available from the output port.



20. Click the **Generated Code** tab and click **Refresh** to view the generated code for all nodes. Click the **Submission** tab to view the submitted code and resulting log, results, and output data. Click the **Output Data** tab to view the imported table. If necessary, use the drop-down list to select the imported table.

    Because a Table node was not connected to the output port of the Import node, a temporary table is created in the **Work** library. We explore how to specify the name and location of the output table with an upcoming example.

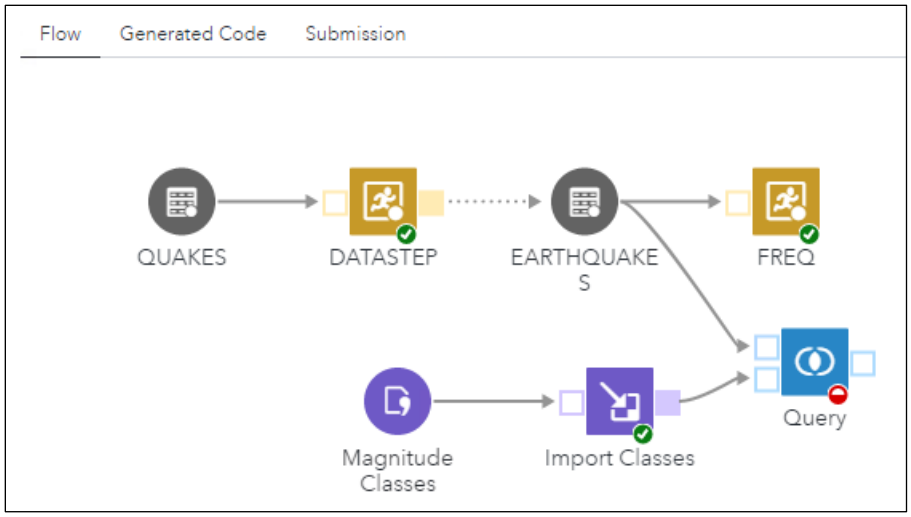


	Magnitude_Low	Magnitude_High	Magnitude_Class
1	1	1.9	Micro
2	2	2.9	Minor
3	3	3.9	Minor
4	4	4.9	Light
5	5	5.9	Moderate
6	6	6.9	Strong
7	7	7.9	Major
8	8	8.9	Great
9	9	9.9	Great

21. Return to the Flow tab.

22. To start a query to join the **earthquakes** table with the imported lookup table, first, right-click **EARTHQUAKES** and select **Add a query** to connect the table to an input port of a new Query node.

23. To connect the imported lookup table to the Query node, select and hold the output port of the **Import Classes** node and then drag it toward the Query node. When a second input port appears on the Query node, place your cursor on the second input port and release. Alternatively, you can right-click the **Query** node and select **Add input port** before connecting the imported lookup table to the second input port of the Query node.



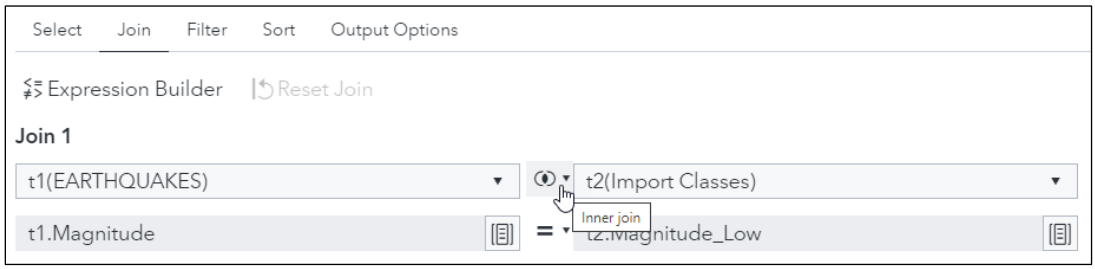
24. To specify a name and location for the resulting output table of the query, right-click the output port of the Query node and select **Add a table**.

25. Select the **Table** node and click (**Restore preview**). In the node details, on the Table Properties tab, next to the **Library** box, click (**Library**). Select the **WORK** library, and in the **Table** box, enter **earthquakes_class**. Click **OK**.

26. Select the **Query** node to specify options for the query. In the node details, click (**Maximize preview**). Both tables are listed in the Columns area, and all columns from both tables can be used in the query.

27. Click the **Join** tab. In the join indicator between the two tables, verify that (**Inner join**) is selected to include only matching rows in the output.

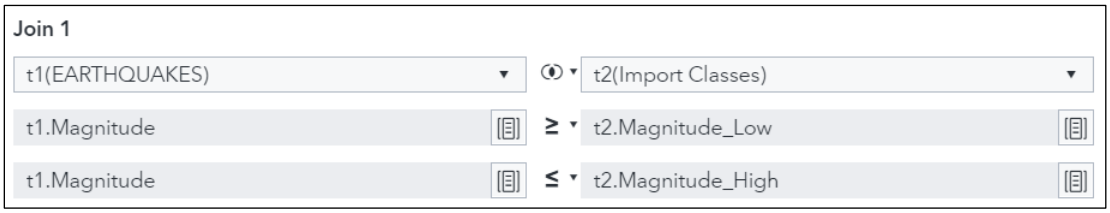
    By default, SAS Studio attempts to join the tables by matching columns that have the same name and type. If there are no matching columns, the tables are joined using the first column from each table.



28. Rows from the two tables should be matched based on the magnitude range defined by **Magnitude_Low** and **Magnitude_High**. If the **Magnitude** value of the earthquakes falls between the range, then it is considered a match, and the corresponding rows from each table are combined into a single row. This requires two join conditions.

    a. On the first join condition, to the left of the operator, click (**Column**). Select **Magnitude** and then click **OK**. Change the operator to ≥ (**Greater than or equal to**). To the right of the operator, verify that **t2.Magnitude_Low** is listed. The table alias names (**t1** and **t2**) might differ depending on the order in which the tables were added to the query.

    b. Next to the first join condition, click (**Add a condition**). If you do not see the button, hover over the first join condition and the button will appear. On the second join condition, to the left of the operator, click (**Column**). Select **Magnitude** and then click **OK**. Change the operator to ≤ (**Less than or equal to**). To the right of the operator, click (**Column**). Select **Magnitude_High** and then click **OK**.



29. Click the **Select** tab. Columns must be added from the Columns area to the Select tab to be included in the output table. To include all columns from the **earthquakes** table, drag **t1 (EARTHQUAKES)** onto the Select tab. Alternatively, double-click the table name or right-click the table name and select **Add columns**.

30. Instead of adding the **Magnitude_Class** column from the imported lookup table, create an enhanced version that lists the magnitude class description, the magnitude lower bound, and the magnitude upper bound (for example, **Moderate : 5 to 5.9**). In the Columns area, select **Calculated Column**.

    a. The expression for the new column can be directly typed into the expression area, or it can be built by selecting functions, columns, and operators.

    b. Click the **Functions** tab. Expand the **Character** folder and double-click **CATX**. The CATX function concatenates strings and inserts delimiters between each string. The delimiter is specified in the first argument, and the strings to concatenate follow. Replace the first argument with **' '** to specify a blank as the delimiter.

    c. Highlight **string2**. Click the **Data** tab and, if necessary, expand **Tables** ⇨ **t2 (Import Classes)**. Double-click **Magnitude_Class** to replace **string2**. The expression should appear as below:

```
CATX(' ',t2.Magnitude_Class)
```

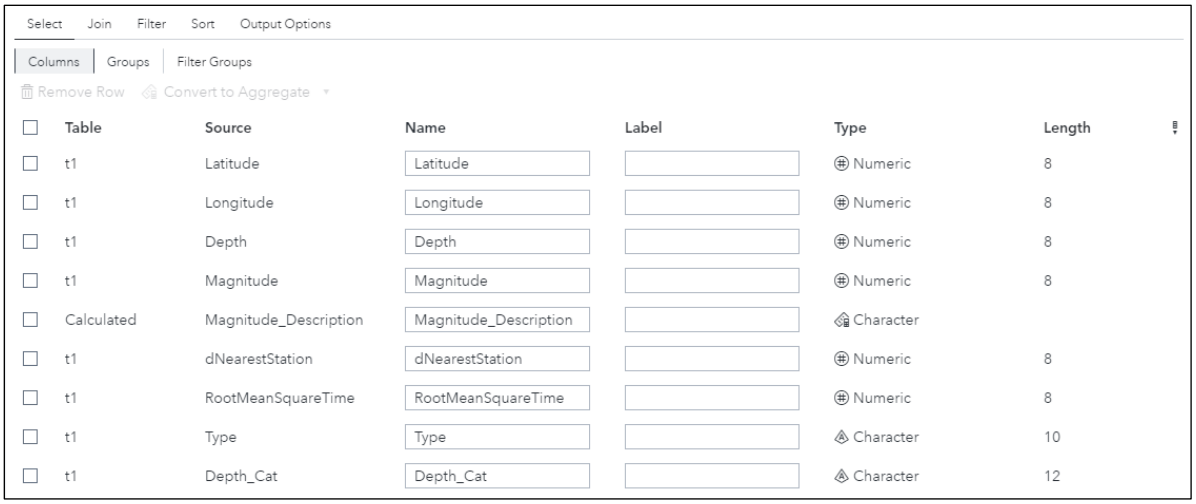
    d. Select or enter a comma and type **':'**. Select or enter a comma and double-click **Magnitude_Low**. Select or enter a comma and type **'to'**. Select or enter a comma and double-click **Magnitude_High**. The final expression should appear as below:

```
CATX(' ',t2.Magnitude_Class,':',t2.Magnitude_Low,'to',
     t2.Magnitude_High)
```

    e. On the Properties tab, in the **Column name** box, enter **Magnitude_Description**. Verify that **Add new calculated column to the Select tab** is selected.

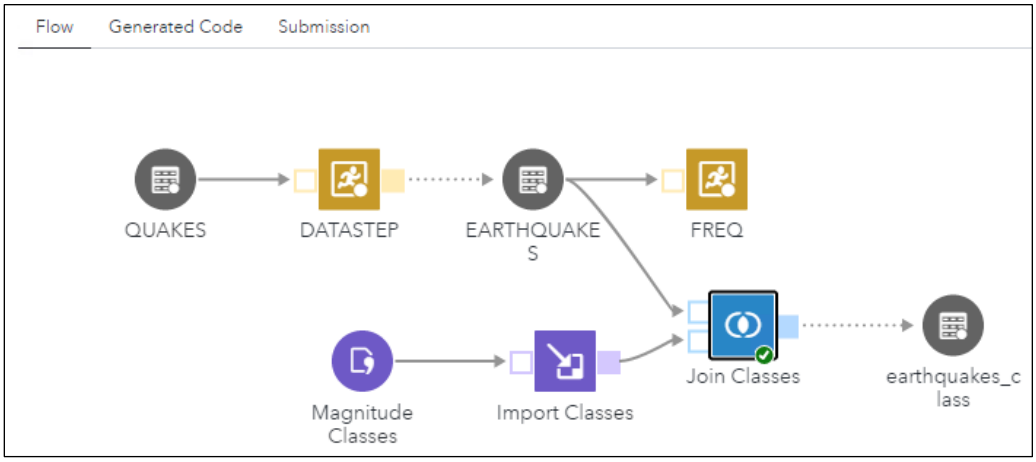
    f. Click **Save**.

31. The new **Magnitude_Description** column is automatically added to the end of the Select tab. The order of the columns on the Select tab determines the order of the columns in the output table. Drag **Magnitude_Description** to directly after **Magnitude**. You can alternatively right-click a column and select **Move to top**, **Move to bottom**, **Move up**, or **Move down** to rearrange the order.



32. Click the **Filter** tab to include only rows with a **Type** value of *earthquake*.
    a. In the Columns area, expand **t1 (EARTHQUAKES)**. Drag **Type** onto the Filter tab, or right-click **Type** and select **Add to filter**.
    b. Next to the **Type** column on the Filter tab, click (**Set a filter on a column**).
    c. In the Add Filter window, verify that **Equal to** is selected in the **Condition** drop-down list.
    d. You can directly type a value in the **Value** box, or retrieve a value from the column. To do the latter, next to the **Value** box, click (**Lookup value**). Click **Get Values**. Select **earthquake** and click **OK**.
    e. Verify that the **Match case** check box is ***not*** selected to make the filter case insensitive. Verify that the **Quote strings** check box is selected and select **Filter**.
33. Click the **Node** tab. In the **Name** box, enter **Join Classes**.
34. Click (**Restore preview**).

35. To run only the query, select the **Join Classes** node, and then, on the flow toolbar, click (**Run a single selected node**). Alternatively, right-click the **Join Classes** node and select **Run node**.



36. Select the **earthquakes_class** node and, in the node details, click the **Preview Data** tab. To easily view more rows and columns, in the node details, click (**Maximize preview**) and, in the upper right corner of the navigation pane, click (**Hide pane**).

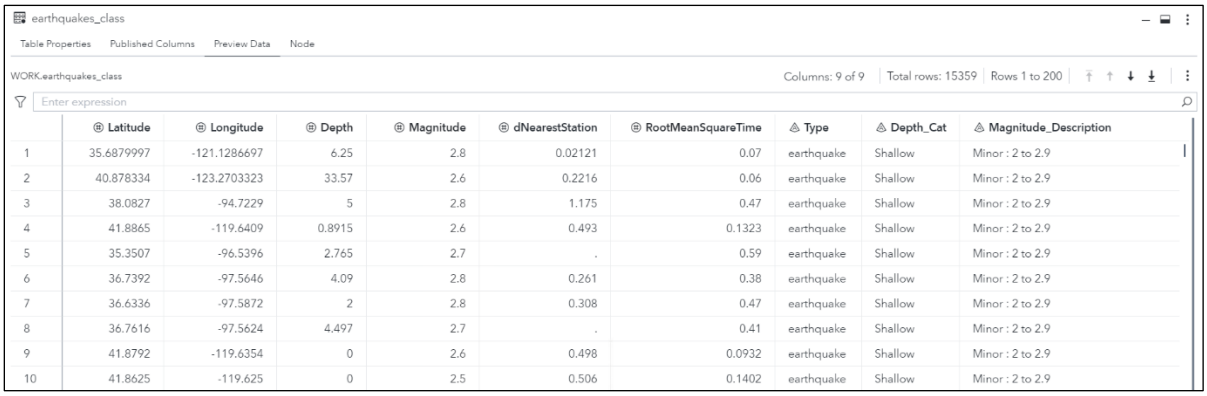

earthquakes_class

Table Properties | Published Columns | Preview Data | Node

WORK.earthquakes_class — Columns: 9 of 9 | Total rows: 15359 | Rows 1 to 200

Enter expression

	Latitude	Longitude	Depth	Magnitude	dNearestStation	RootMeanSquareTime	Type	Depth_Cat	Magnitude_Description
1	35.6879997	-121.1286697	6.25	2.8	0.02121	0.07	earthquake	Shallow	Minor : 2 to 2.9
2	40.878334	-123.2703323	33.57	2.6	0.2216	0.06	earthquake	Shallow	Minor : 2 to 2.9
3	38.0827	-94.7229	5	2.8	1.175	0.47	earthquake	Shallow	Minor : 2 to 2.9
4	41.8865	-119.6409	0.8915	2.6	0.493	0.1323	earthquake	Shallow	Minor : 2 to 2.9
5	35.3507	-96.5396	2.765	2.7	.	0.59	earthquake	Shallow	Minor : 2 to 2.9
6	36.7392	-97.5646	4.09	2.8	0.261	0.38	earthquake	Shallow	Minor : 2 to 2.9
7	36.6336	-97.5872	2	2.8	0.308	0.47	earthquake	Shallow	Minor : 2 to 2.9
8	36.7616	-97.5624	4.497	2.7	.	0.41	earthquake	Shallow	Minor : 2 to 2.9
9	41.8792	-119.6354	0	2.6	0.498	0.0932	earthquake	Shallow	Minor : 2 to 2.9
10	41.8625	-119.625	0	2.5	0.506	0.1402	earthquake	Shallow	Minor : 2 to 2.9

37. On the flow toolbar, click (**Save**) to save the flow.

End of Demonstration

Using Tasks

1. SAS Studio has several features to help generate SAS code: snippets and tasks. Snippets are lines of commonly used code or text that you can save and reuse. SAS Studio is shipped with several code snippets, and these can be found in the Snippets section in the navigation pane. Tasks are point-and-click interfaces to SAS procedures. Unlike steps, tasks cannot be directly added to a flow, but the code generated by the task can be copied into a flow. Use the Bubble Map task to create a geographical map of earthquake locations with bubble sizes representing the magnitude.
2. Select the **Tasks** section in the navigation pane. Expand **Visualize Data** ⇨ **Map** and then double-click **Bubble Map**. The Bubble Map task opens in a new tab in the work area.
3. In the upper right corner of the navigation pane, click (**Hide pane**).
4. On the Data tab, make the following changes:
   a. Next to the **Data** box, click (**Select a table**). Select the **WORK** library and then select the **EARTHQUAKES_CLASS** table. Click **OK**.
   b. To include only earthquakes with a magnitude of 5 or higher, under the table name, click **Filter**.
      1) Double-click **Magnitude** to include it in the expression area.
      2) Click (**Is greater than or equal to**).
      3) Enter **5**. The final expression should appear as below:

```
Magnitude>= 5
```

      4) Click **Save**.
   c. Roles determine how columns are used in the task. Roles with a red asterisk require a column assignment. To assign a column to a role, click (**Select columns**) next to the role. Make the following role assignments:

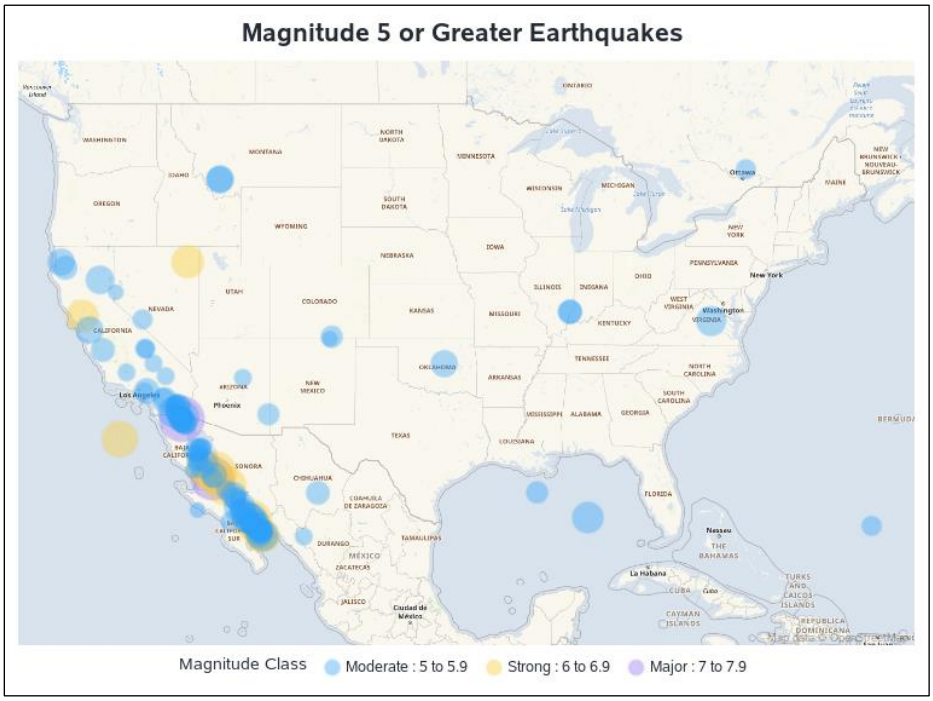
Role	Column
Latitude	**Latitude**
Longitude	**Longitude**
Bubble size	**Magnitude**
Group	**Magnitude_Description**

   d. Verify that the **Include base map layer** check box is selected and that **OpenStreetMap** is selected. You can alternatively select **Esri map** to specify a URL to a specific Esri map that you would like to use. As you make changes to the task options, the code on the Code tab is updated.
5. Click the **Appearance** tab and make the following changes:
   a. If necessary, expand the **Legend** section. Verify that the **Generate plot legend** check box is selected and, in the **Label** box, enter **Magnitude Class**.
   b. If necessary, expand the **Plot** section and increase the transparency to **0.63**.

c. Expand the **Title and Footnote** section. In the **Title** box, enter **Magnitude 5 or Greater Earthquakes** and increase the font size to **15**.

d. Expand the **Graph Size** section. Increase the width to **8** inches and height to **6** inches.

6. To run the task, click **Run** or press the F3 key on your keyboard. View the results on the Results tab.

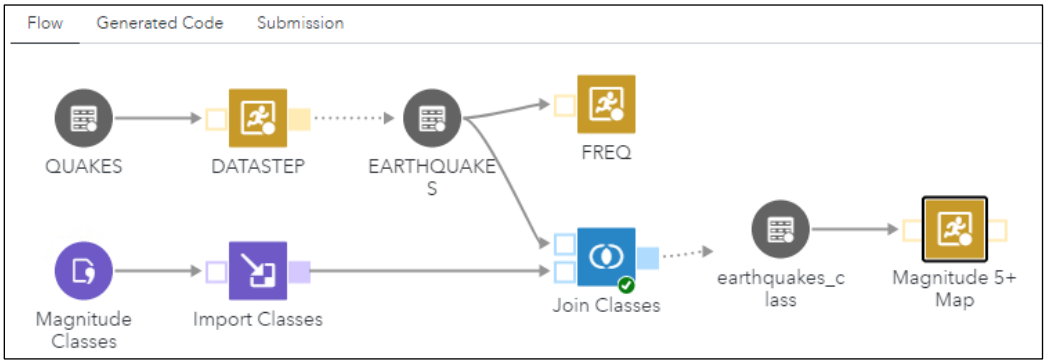


7. To save the options specified in the task, on the task toolbar, click (**Save as**). Navigate to and select a folder of your choice and, in the **Name** field, enter **Magnitude 5+ Map**. Click **Save**.

8. To copy the code generated by the task to the **Earthquake Analysis** flow, on the task toolbar, select **Code to Flow** ⇨ **Earthquake Analysis**.

9. Click the **Earthquake Analysis.flw** tab and, if necessary, click (**Restore preview**) to view both the flow canvas and node details. The code generated by the Bubble Map task is added as a program node to the flow.

10. To connect the **earthquakes_class** table to the **Magnitude 5+ Map** program node, select and hold the right edge of the **earthquakes_class** table node, and then drag it to connect the arrow to the input port of the **Magnitude 5+ Map** program node. To quickly optimize the layout of the flow, on the flow toolbar, click (**Arrange nodes**).



11. Select the **Magnitude 5+ Map** program node and, in the node details, click the **Node** tab. Notice the **Input Ports and Macro Variables** and **Output Ports and Macro Variables** sections. A macro variable stores text that is substituted in your code when it is run. You can specify names for the macro variables storing the table names connected to the input and output ports of a program node and then use those macro variables in the code. Using macro variables to reference tables makes it easy to reuse your flow with other tables. If other tables have the same structure as the current table, you could simply swap out the table without making any adjustments to the code.

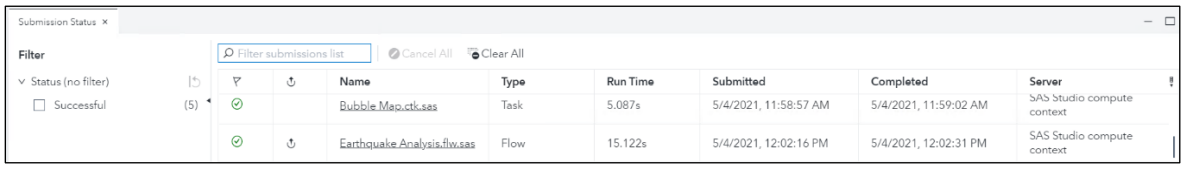
    Use the default macro variable names.

12. Click the **Code** tab. In the PROC SGMAP statement, in the PLOTDATA= option, replace **WORK.EARTHQUAKES_CLASS** with **&_input1** to reference the input port macro variable. Macro variables are referenced by preceding the macro variable name with an ampersand. When the code is run, SAS replaces **&_input1** with the name of the table connected at the input port. The PROC SGMAP step should appear as shown below:

```
proc sgmap plotdata=&_input1(where=(Magnitude>=5));
    openstreetmap;
    title height=15pt 'Magnitude 5 or Greater Earthquakes';
    bubble x=Longitude y=Latitude size=Magnitude/
           group=Magnitude_Description transparency=0.63
           name="bubblePlot";
    keylegend "bubblePlot" / title='Magnitude Class';
run;
```

13. Click — (**Minimize preview**). To run the entire flow, on the flow toolbar, you can click ▷ **Run**. However, an alternative is to use the background submit feature. The background submit feature enables you to run SAS programs, queries, tasks, or flows in the background while you continue to use SAS Studio. To run the flow in the background, on the flow toolbar, click (**Submit the flow in the background using another session**). You can also right-click a program, query, task, or flow in the Explorer section in the navigation pane and select **Background submit**.

14. From the main menu, select **View** ⇨ **Submission Status** to open the Submission Status window. Verify that a green check appears when the flow completes execution. You can select an entry in the Submission Status window to return to an earlier version of a program, task, query, or flow and view the associated log. Close the Submission Status window.



15. On the flow toolbar, click (**Save**) to save the flow.

16. Right-click the **Start Page** tab and select **Close others** to close all tabs expect for the Start Page tab.

End of Demonstration