Quick Start

About the SAS Viya Programming Documentation

This collection of documents provides task-based programming examples, syntax, and concepts for products in SAS Viya, such as:

- SAS Visual Analytics programming
- SAS Visual Statistics programming
- SAS Visual Data Mining and Machine Learning programming

Note: On most hosts, SAS 9.4M5 and later releases are tightly integrated with SAS Viya. See SAS 9.4M5 Integration with SAS Viya in What’s New in Base SAS: Details. (The exceptions are z/OS and 32–bit Windows.)

To search this collection, click 🔍. For search tips, click 🤔, and select Help Tips.

**TIP** If you are viewing this document in a stand-alone context, access the collection before you use the preceding information.

Note: Your site must license and install one or more SAS Viya products to access this functionality.

Orientation

This software supports analytical data preparation, variable transformations, exploratory analysis, analytical modeling, integrated model comparison, and scoring. Here are the main components of SAS Viya:
SAS 9.4M6 is integrated with SAS Viya. SAS programs that are submitted from SAS 9.4M6 clients can call procedures that are unique to SAS Viya, as well as CAS-enabled DATA step code and procedures that have been modified to leverage a CAS server and its tables.

SAS Viya
The third generation of high-performance, in-memory analytics.

SAS Studio
The integrated SAS programming environment.

SAS Cloud Analytic Services (CAS)
The analytic engine. CAS uses high-performance, multithreaded analytic code to rapidly process requests against data of any size.

SAS Visual Analytics
Programming tools that provide baseline functionality, including reporting and basic analytics, such as:
- analytical data preparation
- variable transformations
- exploratory analysis
- descriptive statistics

SAS Visual Statistics
An additional set of advanced analytic functionality that builds on SAS Visual Analytics, such as:
- building predictive models
- integrated model comparison

SAS Visual Data Mining and Machine Learning
An additional set of advanced analytic functionality that builds on SAS Visual Statistics, such as:
- tune machine learning algorithm hyperparameters
- advanced statistical operations
- analyzing complex data

SAS Econometrics
A set of functionality that provides techniques to model complex business and economic scenarios and to analyze the dynamic impact that specific events might have over time.

SAS Visual Forecasting
Provides automatic variable, event, and model selection. It then automatically generates your forecasts.

SAS Visual Text Analytics
A text analytics framework combining text mining, contextual extraction, categorization, sentiment analysis, and search.

SAS Optimization
A set of procedures for exploring models of distribution networks, production systems, resource allocation problems, and scheduling problems using the tools of operations research.

CAS-based procedures run against data that is in CAS. For example, before you can use CAS to work with a SAS data set, you must load that data set into CAS. The following instructions demonstrate the basic mechanics.

**TIP** If you cannot securely sign in and start a CAS session, contact your administrator or see the troubleshooting topic in *SAS Viya Administration: Identity Management.*
Demonstration: Load Personal Data

1. Sign in to SAS Studio.
   a. Open SAS Studio from a URL that is provided by your SAS administrator. For example, you might enter either of the following URLs:
      https://webserver-host-name/SASStudioV (for SAS Studio 5.2, which is available in full deployments)
   b. If you are prompted for credentials, sign in.

2. Start a CAS session.
   a. In the Snippets section of the navigation pane, expand Snippets ➔ SAS Viya Cloud Analytic Services.
   b. Right-click New CAS Session and select Open as program. The snippet opens in the code editor.
   c. In the toolbar, click ⏸️ to run the New CAS Session code.

3. Load a table.
   a. In the navigation pane, right-click Load data to caslib and select Open as program.
   b. In the code editor, edit the SAS data set section so that it looks like this:

      ```sas
      PROC CASUTIL;
      LOAD DATA=sashelp.cars OUTCASLIB="CASUSER"
      CASOUT="demoTable" PROMOTE;
      RUN;
      ```

      **TIP** CASUSER is your personal caslib. It is available across your sessions. You cannot enable other users to access it.
   c. Select the preceding code. In the toolbar, click ⏸️ to run only the four lines of selected code.

4. Verify that you can access the loaded data.
   a. In the navigation pane, right-click Generate SAS librefs for caslibs and select Open as program.
   b. In the toolbar, click ⏸️ to run the code.
   c. In the Libraries section of the navigation pane, expand Libraries ➔ CASUSER.
   d. Double-click demoTable to open it.

Demonstration: Provide Shared Data

1. Sign in to SAS Studio and start a CAS session, if you have not already done so.
2 Create a container for shared CAS data.
   a In the Snippets section of the navigation pane, expand Snippets ➔ SAS Viya Cloud Analytic Services.
   b Right-click New caslib for Path and select Open.
   c In the code editor, edit the snippet so that it looks like this:

   ```sas
   CASLIB demoCas PATH="/filePath/" DATASOURCE=(SRCTYPE="path") GLOBAL;
   ``

   Note: Enter a path that is relative to and accessible from your CAS server. You can reference an empty directory.
   d Click to run the code.

   Note: If an error indicates that you do not have permission to create global caslibs, see “Caslib Management Privileges” in SAS Viya Administration: SAS Cloud Analytic Services. Initially, only administrators can add global caslibs. An administrator can enable non-administrators to add global caslibs.

3 Give all users Read access to the new caslib.
   In the SAS Studio 5.2 Enterprise edition, complete these steps:
   a Access the Data page in SAS Environment Manager.
      - Select Manage Environment from the applications menu (≡) and then click in the vertical navigation bar.
   b On the Data page, select the Data Sources tab. Right-click the new caslib, and select Edit authorization.

   TIP Data page functionality is also available in other contexts. See Data Selection Windows and SAS Data Explorer in SAS Data Explorer: User’s Guide.
   c In the Edit Authorization window, adjust the gauge in the Access Level column to increase access for Authenticated Users from No access to Read.
   d Click Save.
   e Go back to SAS Studio.
      - Select Develop SAS Code from the applications menu (≡).

4 Load data to the new caslib.
   a In the navigation pane of SAS Studio, right-click the snippet Load data to caslib and select Open as program.
   b In the code editor, edit the SAS data set section so that it looks like this:

   ```sas
   PROC CASUTIL;
   LOAD DATA=sashelp.cars OUTCASLIB="demoCas"
   CASOUT="demoTable" PROMOTE;
   RUN;
   ``

   c Select the preceding code. In the toolbar, click to run only the four lines of selected code.

5 Verify that other users can see the data. For example, ask them to complete these steps:
a Sign in to SAS Studio, start a CAS session, and run the **Generate SAS librefs for caslibs** snippet.

b In the **Libraries** section of the navigation pane, expand **Libraries ➪ demoCas**.

c Double-click **demoTable** to open it.

---

**Tip: Automatically Connect and Generate Librefs**

For convenience, you can configure SAS Studio to perform the following tasks each time that you sign in:

- Start a CAS session.
- Generate SAS librefs for existing caslibs that have names that are no more than eight characters long.

Complete the following steps:

1. Click **Options** and select **Autoexec file**.
2. On the **Autoexec.sas** tab of the (Edit) Autoexec File window, paste the following code:
   ```sas
   cas casauto;
   caslib _all_ assign;
   ```
3. Run the code.
4. Save the code.

---

**Documentation: References by Task**

<table>
<thead>
<tr>
<th>Task</th>
<th>Refer to:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Do you need to migrate your data to UTF-8?</td>
<td>Migrating Data to UTF-8 for SAS Viya</td>
</tr>
<tr>
<td>Access your data</td>
<td>DATA Step Examples</td>
</tr>
<tr>
<td></td>
<td>Common Tasks for Accessing Data</td>
</tr>
<tr>
<td>Manipulate your data</td>
<td>Data Step Basics</td>
</tr>
<tr>
<td>Prepare, Model, Assess</td>
<td>MDSUMMARY Procedure</td>
</tr>
<tr>
<td></td>
<td>SAS Visual Data Mining and Machine Learning: Procedures</td>
</tr>
<tr>
<td></td>
<td>SAS Visual Statistics: Procedures</td>
</tr>
<tr>
<td>Graph your output data</td>
<td>Introduction to SAS Platform Graphing</td>
</tr>
<tr>
<td>Task</td>
<td>Refer to:</td>
</tr>
<tr>
<td>----------------------------------------------------------------------</td>
<td>---------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Program with the Lua, Python, R, or CASL languages</td>
<td><em>Getting Started with SAS Viya for Lua</em></td>
</tr>
<tr>
<td></td>
<td><em>Getting Started with CASL Programming</em></td>
</tr>
<tr>
<td></td>
<td><em>Getting Started with SAS Viya for Python</em></td>
</tr>
<tr>
<td></td>
<td><em>Getting Started with SAS Viya for R</em></td>
</tr>
<tr>
<td>Learn more about SAS Viya and SAS 9 interactions.</td>
<td><em>An Introduction to SAS Viya Programming</em></td>
</tr>
<tr>
<td></td>
<td><em>SAS Guide to Software Updates and Product Changes</em></td>
</tr>
</tbody>
</table>