# BioSense Platform User Manual for SAS Studio

April 2020

Center for Surveillance, Epidemiology, and Laboratory Services
Division of Health Informatics and Surveillance

**Technical Assistance:** support.syndromicsurveillance.org

The National Syndromic Surveillance Program (NSSP) promotes and advances development of the cloud-based BioSense Platform, a secure integrated electronic health information system that hosts standardized analytic tools and facilitates collaborative processes. The BioSense Platform is a product of the Centers for Disease Control and Prevention (CDC).

# CONTENTS


## 1 Overview, 1

*First-time Access to SAS Studio, 1*
*Log in to SAS Studio, 1*

## 2 How to Access SAS, 2

*User Permissions and Folder Access on the File Server, 2*
*SAS Studio-specific Folder Shortcuts, 4*

## 3 Basic Navigation, 5

*User Interface, 5*
*The Basics: Write, Run, and Save Code, 6*
*How to Check the Log Panel for Feedback, 7*

## 4 Setup, 8

*First-time Use of SAS Studio, 8*
*Setup Instructions, 8*
*Snippets (Required, Recommended, and Optional), 8*

## 5 How to Work with Data, 12

*Data Sources and Library Names, 12*
*How to View Data Sources in My Libraries, 12*
*Creating Custom Libraries and Folders, 14*
- Create New Folder, 14
- Custom Libnames or Libraries, 15

*Creating and Running Code, 17*
*Saving SAS Code, 19*

## 6 Functionality, 21

*Uploading Data, 21*
*Importing Data, 21*
*Downloading Files, 22*
*Background Submission of SAS Code, 23*

## 7 DataMart Connection and Querying, 24

*SAS Documentation, 24*
*Basic Methods for Connecting to the DataMart, 24*

*How to Select the Appropriate SQL Pass-Through Code, 25*
SAS Data Step Example, 25
SQL Pass-Through Example, 26
Proc SQL Example, 26

# 8 DataMart on the BioSense Platform, 28

*DataMart Partitioning into Tables, 28*
*DataMart Indexes, 28*

# 9 Using ESSENCE APIs in SAS Studio, 29

*Building a Query in ESSENCE, 29*
*API Pulls in SAS Studio, 30*
*Example of an API Pull in SAS Studio, 31*

# 10 Operational Considerations, 33

*File Storage and Retention Policy, 33*
*Fileserver Storage Allocation, 33*
*SAS Memory and Datasets: Out-of-Memory Errors, 33*

## Appendix A: NSSP-provided SAS Code, Macros, and Snippets, 34

*Folder Shortcuts/SAS/programs, 34*
Completeness.sas, 34
Completeness_Exceptions.sas, 34
Datamart_Connect.sas, 34
DataQuality_OnDemand.sas, 35
Data_Connection_Basics.sas, 35
Essence_API_Basic.sas, 35
Essence_API_Advanced.sas, 35
Pilot_Lag.sas, 36
Primary_Facility_Count.sas, 36
*Folder Shortcuts/SAS/macros, 36*
mssqlexe.sas, 36
Nsspdates.sas, 37
Nsspoptions.sas, 37
Prpwonoff.sas, 37
*Folder Shortcuts/SAS/snippets, 38*
Create_Site_Folder_Shortcut.sas, 38
Folder_Create.sas, 38
mssqlexec_snippet.sas, 38
Password_Encode.sas, 39
Site_Info.sas, 39
User_Info.sas, 39

## Appendix B: Fileserver Drive Space Allocation Rules, 40

## Appendix C: Best Practices, 41

# BioSense Platform User Manual for SAS Studio

## 1. Overview

SAS Studio is a customizable, Web browser-based interface for advanced analytics and visualization of complex data. SAS Studio is one of several software tools available on the BioSense Platform.

This manual provides the basics for using your site's data with SAS Studio and will familiarize you with the interface, features, navigation, and functionality. The manual describes how to configure your SAS account so that you can efficiently use SAS Studio to connect to and analyze your site's data.

Browsers that support SAS Studio:
- ✓ Chrome
- ✓ Edge
- ✓ Mozilla Firefox

- First-time users should follow the steps provided to configure their SAS accounts to connect to the BioSense Platform data.
- **Chrome is the preferred browser for SAS Studio**. Functionality might be limited when using other browsers.
- This guide is **not** a SAS programming guide and presumes a moderate level of SAS programming experience to run SAS code and develop SQL queries using SAS Studio.
- SAS Product Support Documents can be found at General SAS Product Support. Check the following link for specific support for SAS Studio.
- A PDF of the *SAS Studio 3.5 User's Guide* can be downloaded from this link: Web Editor UG.

### *First-time Access to SAS Studio*

You must have a BioSense Platform Access & Management Center (AMC) account before a SAS Studio account can be created. **Only site administrators can grant access to SAS Studio**. Site administrators can grant access to SAS Studio by using the Create/Edit User functionality on the Manage Users tab.

Once your site administrator has given you access to SAS Studio, your SAS account will be created within 1 business day, and you will be notified by email from the NSSP mailbox.

Use your AMC username and password to log in to SAS Studio.

### *Log in to SAS Studio*

1. Log in to AMC and click on **SAS Studio** in the Home tab > NSSP Applications section.
2. When the SAS login screen appears (Figure 1), enter your AMC user ID and password.
3. Click **SIGN IN** button.

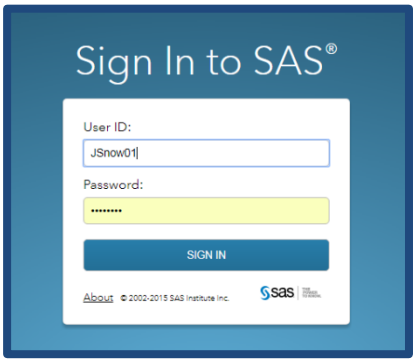



*Figure 1. SAS Sign-in Page*

# 2. How to Access SAS

### *User Permissions and Folder Access on the File Server*

Once you have SAS access, you also have permission to use specific folders on the central file server named "Fileserver." Use the folders on Fileserver to save and access your programs through SAS (Figure 2).
**Note:** This is also true for the RStudio application. RStudio files are saved to the same file server.

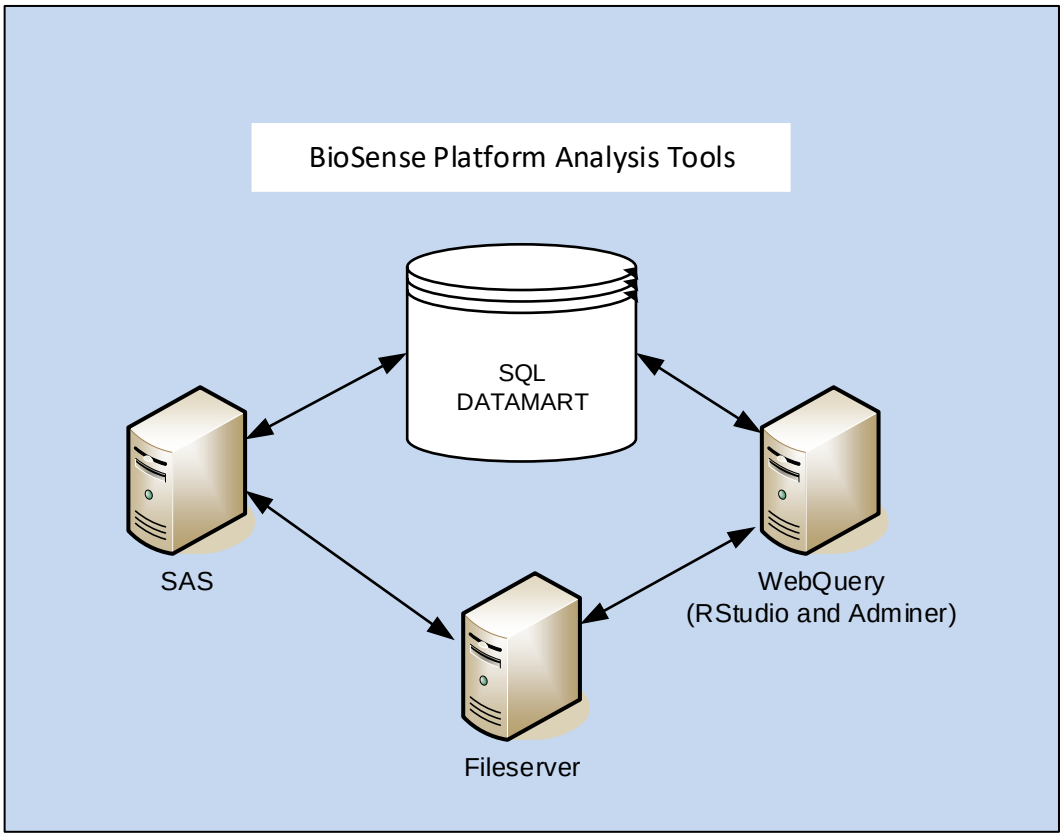



*Figure 2. A secure central file server hosts BioSense Platform tools and applications.*

Users can access several folders (Figure 3 on next page) on the file server, namely, their Files (**Home**), site-specific, and public folders. Your Files (**Home**) folder is sometimes just referred to as your **Home** folder.

- The Files (**Home**) folder is where most programming will be saved. You will find it at the top-level **Server Files and Folders** or Navigator pane under **sas-cmp1**. Please consider creating subfolders under this folder to organize your code and datasets. **Note: The SASUSER.v94 folder is a subfolder in your Files (Home) folder—it is *not* your Home folder.**
- Each site has a dedicated, site-specific folder for sharing programming and data files *securely*. To simplify access, consider mapping a folder shortcut to your site folder. This process will be explained later in Section 4.
- Programming and files saved to the **Public** folder will be shared with all users. This folder is accessible by everyone and is *not* site specific. Any SAS user can create a subfolder in the **Public folder** and write to existing folders. **Everything in the Public folder is readable by *all users from all sites.***

**Caution!** Avoid saving anything in the public folder you do not want others to see (e.g., sensitive data, usernames, and passwords).

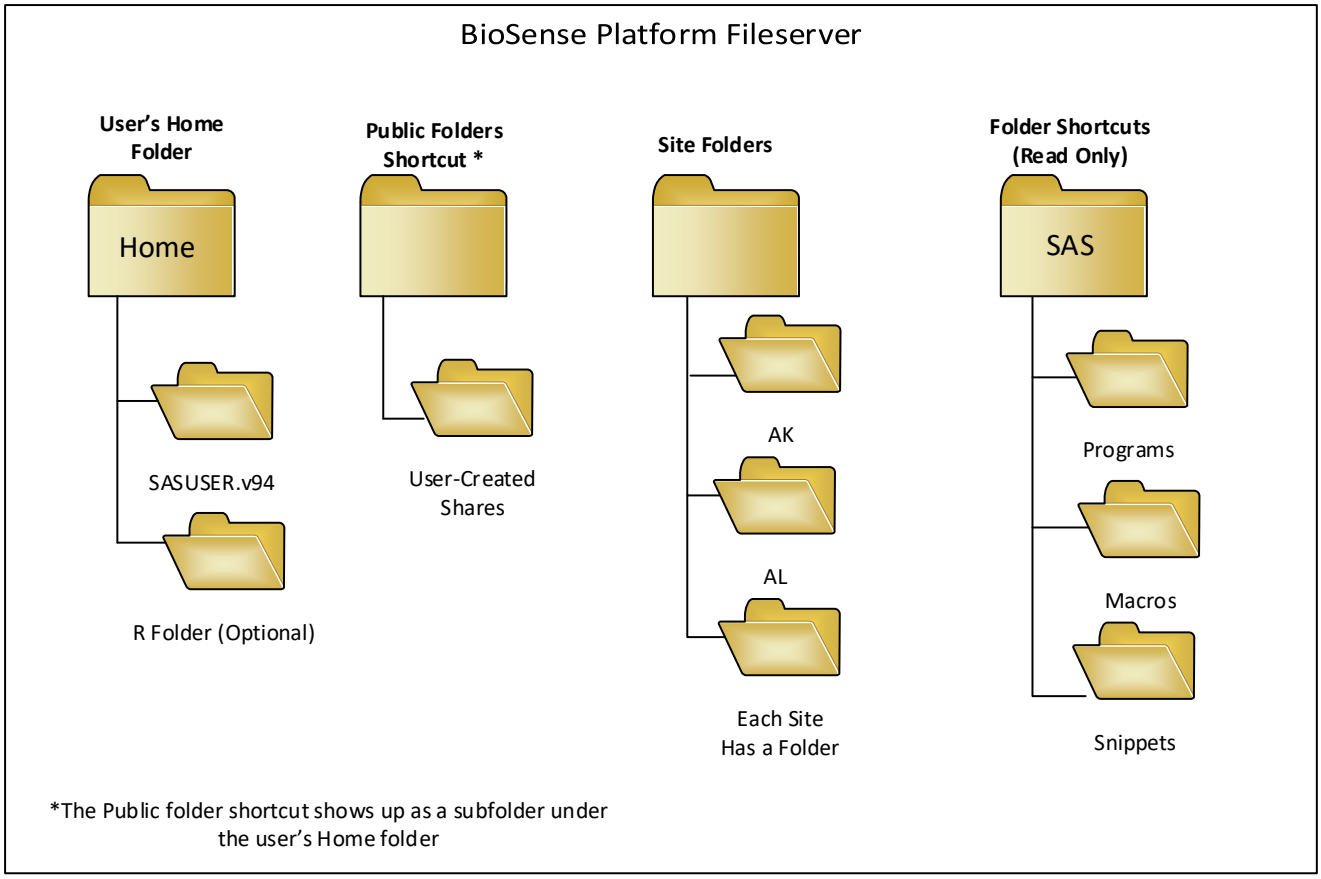



*Figure 3. The file server contains all the folders you and others at your site need for sharing and saving SAS code.*

Figure 4 shows a screenshot of the navigator pane with a new folder you can create (e.g., My_SAS) for storing the SAS code you create or modify.

**Note:** If you use both SAS Studio and RStudio, you will see the Public, R, and sasuser.v94 folders in your Files (**Home**) folder in both SAS Studio and RStudio. We suggest creating separate folders to store your SAS work and R work to better organize your data and code.

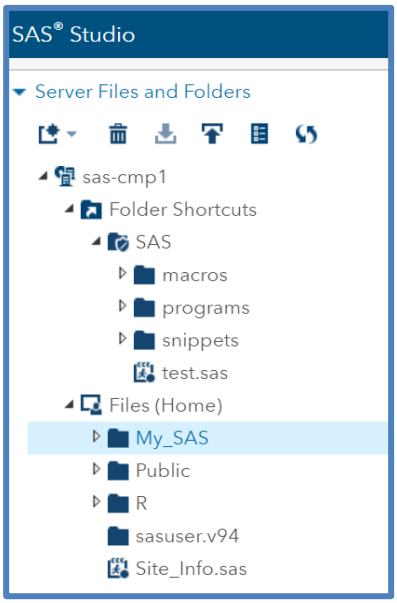



*Figure 4. SAS Studio Navigator Pane*

### *SAS Studio-specific Folder Shortcuts*

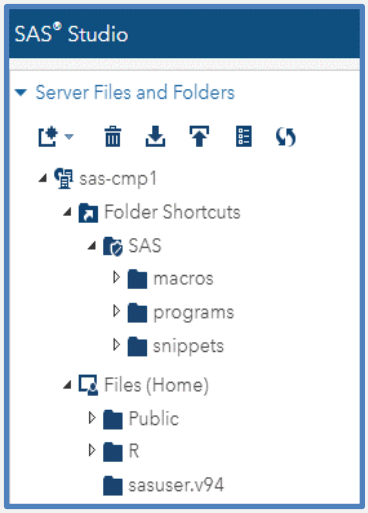
All SAS Studio users have read-only access to the set of global folder shortcuts found in the Folder Shortcuts/SAS folder. This SAS folder can be seen by expanding "Folder Shortcuts" in the Navigator pane. The SAS folder has three subfolders—programs, macros, and snippets—containing SAS programming code to assist users with standard tasks, such as connecting to the DataMart, suppressing the username and password in log files, or for mapping to their site's shared folder.

These folders are used to share helpful programming hints with all users. Only the CDC SAS administrator can save files in these folders.

**How are the subfolders in the Folder Shortcuts/SAS used?**

The macros folder contains:

- SAS-specific macros required by programs in the SAS/Programs folder, and
- SAS macros of general interest to everyone.




Sometimes the NSSP team writes code for everyone's use and saves it in the programs folder. You can open and save these programs to your Files (**Home**) folder. **By default, users cannot save programs in the SAS Folder Shortcuts or its subfolders.** If you want to add a program to SAS Studio that could benefit everyone, please submit a ticket to the NSSP Service Desk and request that the program be considered for inclusion.

Snippets are a few lines of code that are essential or considered useful. Snippets may be small SAS macros or other pieces of code, or they may hold parameter values to be included in SAS programs. You can find these in the snippets folder.

**Tip:** Macros, programs, and snippets can be saved to your Files (**Home**) folder or to other folders within it for modification. Your modified files or the originals can be referenced in your SAS code directly by embedding the folder path and file name.

Detailed information on the purpose of the programs, macros, and snippets is provided in Appendix A.

# 3. Basic Navigation

Several steps are required before you can start using SAS Studio to analyze data. The steps are detailed in Section 4; but first, you need to understand how the SAS Studio User Interface is laid out.

## *User Interface*

You will use the SAS Studio user interface to set up SAS Studio for personal use (see Section 4 for detailed instructions). When you log in to SAS Studio, you will see two panes that make up the user interface (Figure 5). The left-hand pane, the Server Files and Folders or Navigator pane, contains library allocations, folders, folder shortcuts, and saved code. The right-hand pane, called the Editor pane, is a workspace for writing or modifying SAS code, running programs, and examining logs and results once a program has completed.

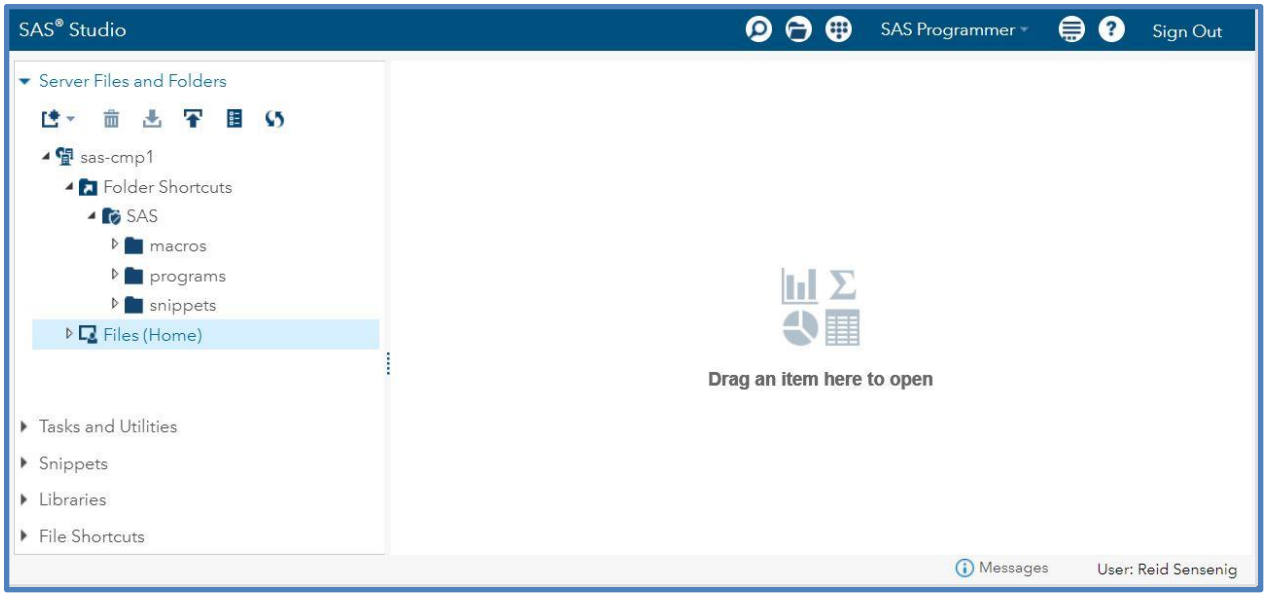



*Figure 5. The SAS interface consists of a navigator pane with files and folders (on the left) and separate Editor pane or workspace (on the right).*

**Note:** You may right-click in the Editor window to open the Editor menu (Figure 6). Here you may create a new SAS program, import data, or create a new query or an XML file

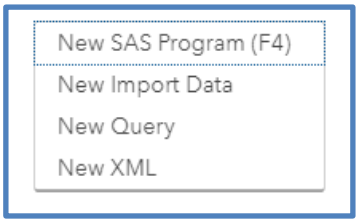



*Figure 6. Editor Menu*

Once you create a new SAS program or open a program, macro, or snippet in the Editor window, the toolbar, as shown in Figure 7, is displayed at the top of the Editor window.

Please note that this toolbar is for SAS programs. Other toolbars are displayed for Import Data, Query, and XML.

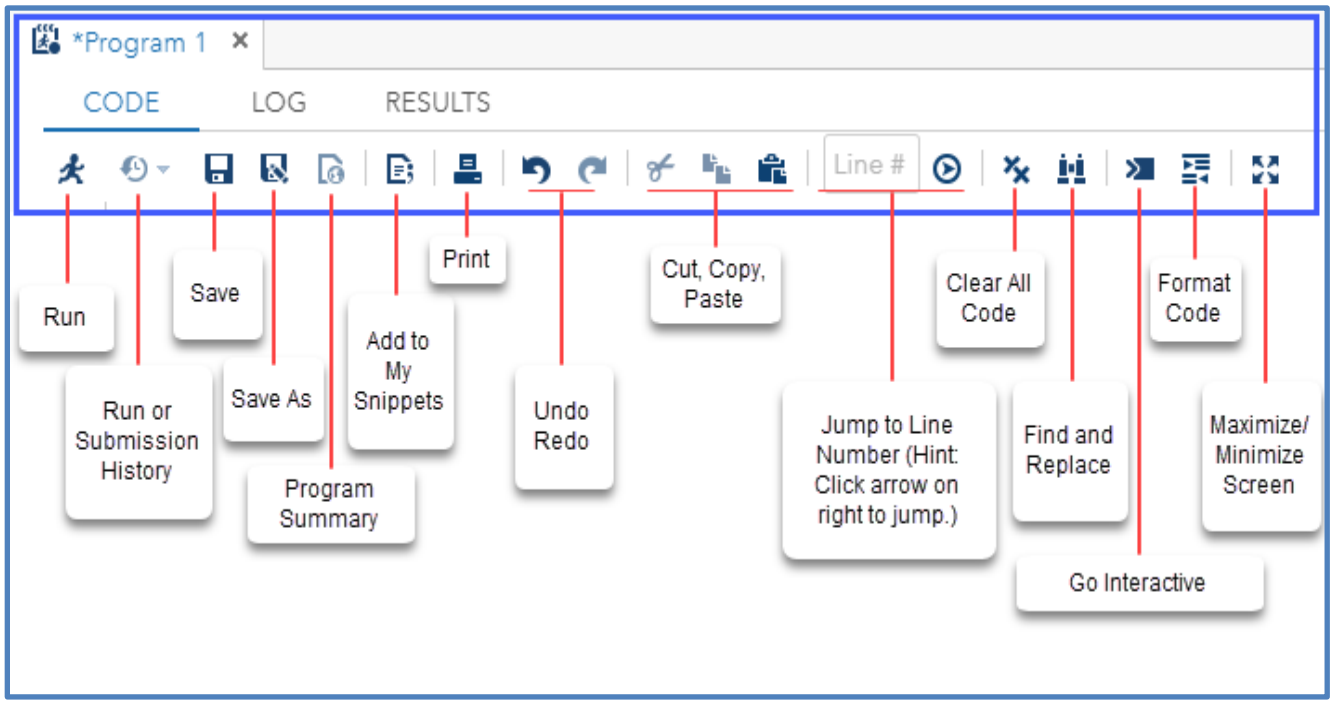



*Figure 7. Toolbar Icons for SAS Programming Code*

## *The Basics: Write, Run, and Save Code*

1. As shown above in Figure 5, you can use the right-hand pane (Editor window) to view existing code or write new code.
2. Click the **Running Man** icon (Figure 8) to execute the code.

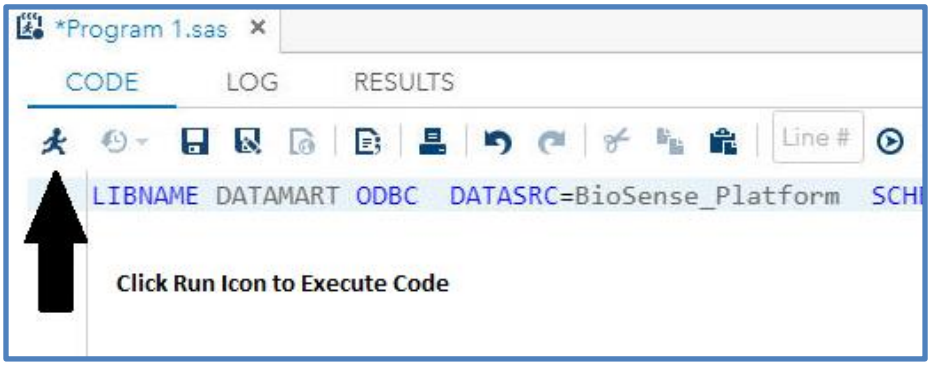



*Figure 8. The Running Man executes the code in the Editor pane.*

3. To save code, click either the **SAVE** button or to save to a specific file name and folder, the **SAVE AS** button (Figure 9).

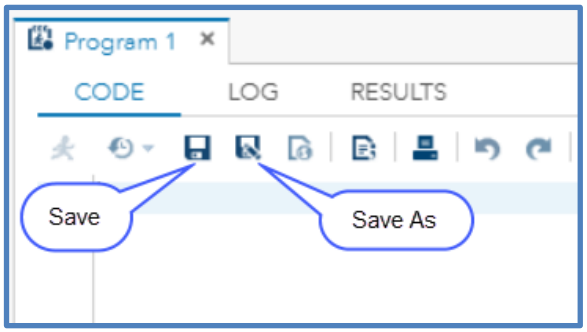



*Figure 9. Select **Save** or **Save As** icon to save the program.*

## *How to Check the Log Panel for Feedback*

The log panel keeps a history of Errors, Warnings, and Notes that you might encounter when running your program (Figure 10). This feedback will be helpful if you need to debug your program.

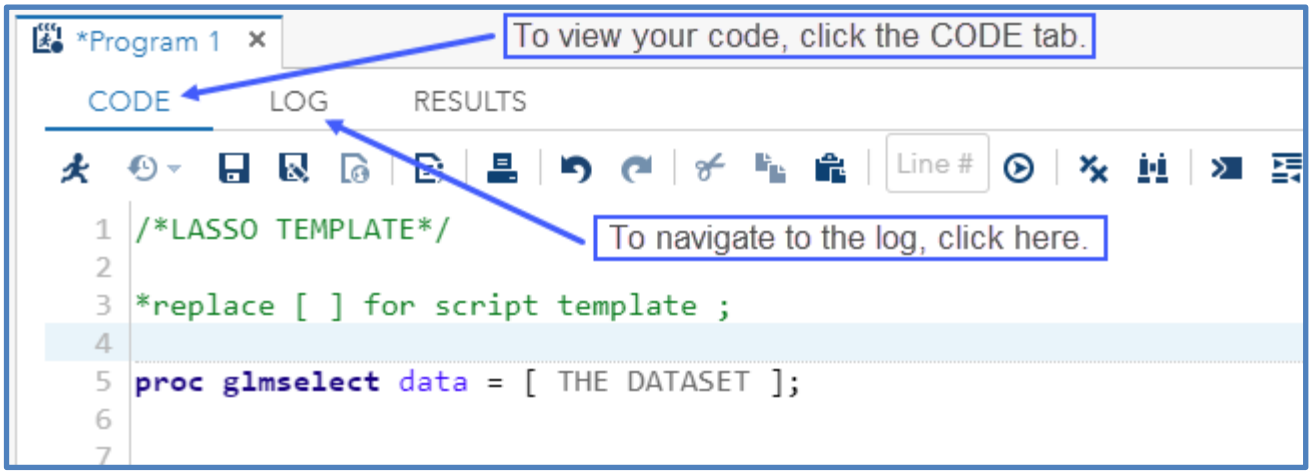



*Figure 10. The Editor pane is always displayed so that you can view code, log, and results.*

# 4. Setup

## *First-time Use of SAS Studio*

Follow these steps the first time you use SAS Studio:

1. In the Folder Shortcuts, expand the **SAS** folder, then the snippets folder.
   Copy **Site_Info.sas** and **User_Info.sas** snippets to your **Home** folder.
2. Modify and execute **Password_Encode.sas** to encrypt your password, then set up the **User_Info.sas** and **Site_Info.sas** snippets. Except for your encrypted password, your site administrator can provide the information needed for these files.
3. Set up and connect to your site-specific folder.
4. Connect to the DataMart library.

## *Setup Instructions*

1. Copy the personal info snippets to your Home folder.

   In the Navigator pane on the left, expand the sas-cmp1 > Folder Shortcuts > SAS > snippets folder and right-click on each of the following snippets. In the sub-menu popup, click on Copy To and select your Files (Home) folder to copy these into your Home folder:
   - User_Info.sas
   - Site_Info.sas

2. Run the Password_Encode snippet, and then customize the Info snippets.

## *Snippets (Required, Recommended, and Optional)*

### **Password_Encode.sas** (Required)

Passwords stored on the BioSense Platform are *not* allowed to be saved in clear text. Passwords saved on the BioSense Platform must be encrypted or entered every time you need to make a data connection.

Go to **Folder Shortcuts > SAS > snippets,** and open the **Password_Encode.sas** utility snippet. (You open a snippet by double-clicking it or right-clicking and selecting **Open**.)

Look down several lines and you will see instructions. Follow these instructions closely.

In this line:

```
PROC PWENCODE IN='MyUnencryptedPW1' METHOD=SAS005;
```

replace **MyUnencryptedPW1** with your current AMC unencrypted password. Then click the **Running Man** icon to execute this code.

When you run this code, the view will automatically switch to the **LOG** tab. Once there, scroll down until you see a line beginning {SAS005} followed by a long string of hexadecimals.

**For example:** {SAS005}A61FC901B27CCD53A87111AB1AD8A56E262525E8D2B64A6F

Paste this string to your saved copy of **User_Info.sas** (instructions follow below).

**CAVEAT:** When you close the **Password_Encode.sas** file, you will be asked *"Do you want to save "Password_Encode.sas?"* We recommend that you do NOT save it as it still contains your unencrypted password.

### **User_Info.sas** (Required)

The User_Info snippet contains your username, encrypted password, and site_short_name (e.g., AK, AL, AZ). This snippet is used (i.e., included) in various SAS macros and programs. The long string created when you run **Password_Encode.sas** beginning with {SAS005} is your encrypted/encoded password. Please copy this string to the clipboard. Then open your copy of **User_Info.sas**—the one you saved to your Files **(Home)** folder—and locate the line that reads:

```
%LET PW = %STR({SAS005}XXXXXXXXXXXXX);
or your previously encrypted password
```

Replace the string within the parentheses with the string you had copied to your clipboard. (Be sure to include the {SAS005} prefix.) Before you save User_Info.sas, follow the instructions to enter your SAS UserID (same as your AMC User ID) and site_short_name. (Remember to remove any < > brackets.) When all three lines have been updated, save this file to your Files (**Home**) folder. **Note:** You do NOT need to run this snippet.

**For example:**

```
%LET UserID=Your_AMC_UserID;
%LET PW = %STR({SAS005}A61FC901B27CCD53A87111AB1AD8A56E262525E8D2B64A6F);
%LET site_short_name=XX;
```

When your AMC password expires or is changed, you must re-run the Password_Encode.sas to encrypt your new password. After encrypting it, save the new encryption string in the User_Info.sas snippet in your Home folder. After updating your User_Info.sas snippet, you must run the DataMart_Connect.sas program again. This will connect to the DataMart using your new password. The instructions for running the DataMart_Connect.sas program are shown below.

### **Site_Info.sas** (Required)

This file contains your site-related information that other SAS programs need to process data. Open your copy of the **Site_Info.sas** snippet you saved in your **Home** folder. This code snippet contains five variables or parameters that you must assign values to. If you are unsure of any of these, please contact your site administrator for the correct values. Remove the < > signs and names within them and enter the correct values for your site, then save this file in your **Home** folder.

You must save **Site_Info.sas** to your **Home** folder and make sure its contents are up to date. Once saved, the location for the **Site_Info.sas** snippet should appear with the **User_Info.sas** snippet (Figure 11).

**Note:** Other SAS programs use variables stored in Site_Info.sas (e.g., programs distributed by the NSSP Analytic Data Management team). Blank fields can trigger SAS errors.

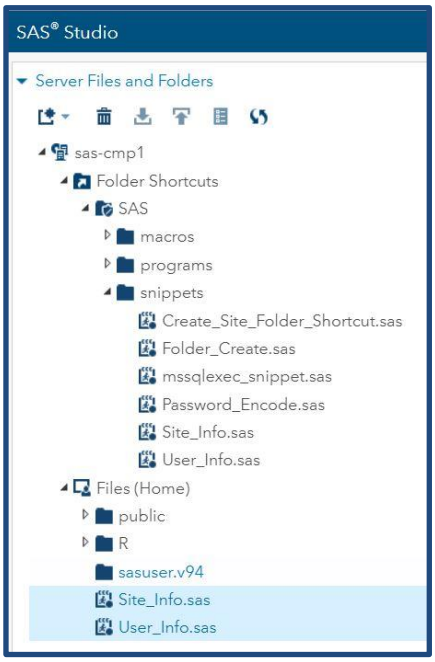



*Figure 11. Save snippet in Home folder.*

### DataMart_Connect.sas (Required)

You will find the **DataMart_Connect.sas** program in the **sas-cmp1 > Folder Shortcuts > SAS > programs** folder. It contains the default SAS LIBNAME statement used to connect your session to the BioSense_Platform DataMart (Figure 12). You may run this program from the global SAS programs folder shortcut.

Notice that this program requires your **User_Info.sas** snippet to access your SAS UserID and encrypted password. Be sure you have saved **User_Info.sas** to your **Home** folder and the information in that snippet is current before running this program.

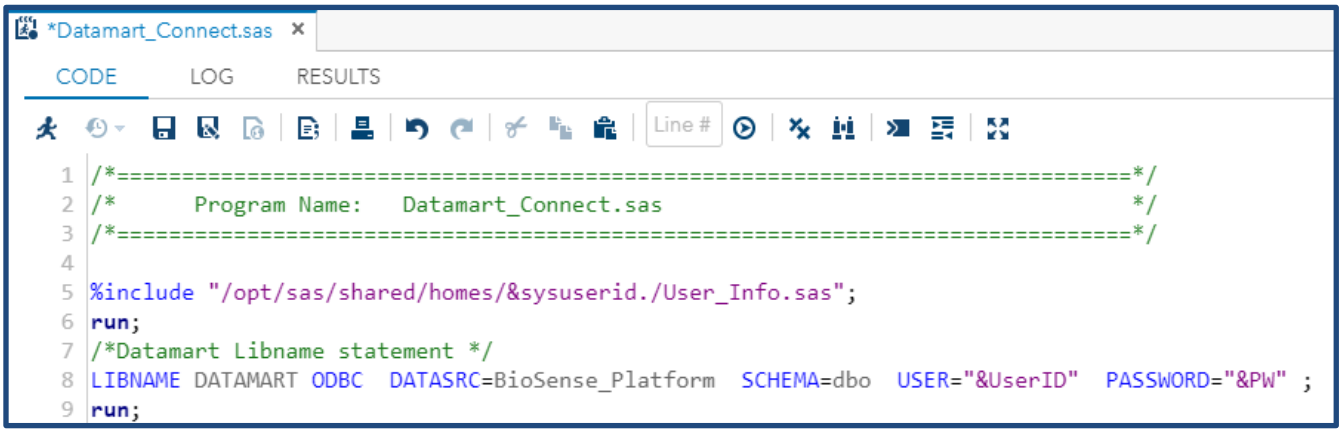



*Figure 12. DataMart_Connect.sas program includes saved values for username and encrypted password from User_Info.sas.*

### Create_Site_Folder_Shortcut.sas (Recommended)

This snippet will connect you to your site's shared folder. After initial access and execution, the shared folder will always be available when you log in to SAS Studio. (This program should only be run one time.)

**Note:** You must run the **Create_Site_Folder_Shortcut.sas** snippet before your site folder will appear.

***Setting Up the Shared Folder***—Follow these steps after you update and save **User_Info.sas** and **Site_Info.sas** in your **Home** folder:

1. Open the **Create_Site_Folder_Shortcut.sas** snippet in the SAS snippets folder, and then run it by clicking the **Running Man** icon.
2. **Refresh the folder view**. You may need to click the **refresh** icon in the Navigator pane if the newly created folder shortcut does not appear as highlighted in Figure 13.

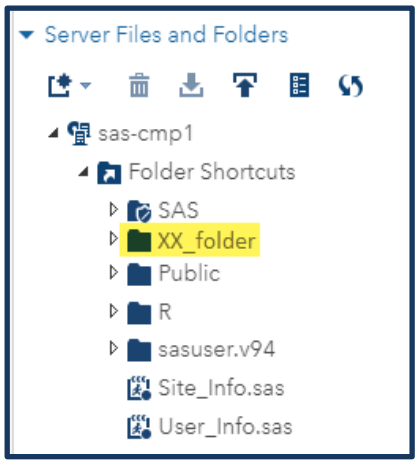



*Figure 13. Upon refresh, site folder should appear under Home folder.*

**REMINDER:** The program and data files saved in your site's shared folder may be run by all SAS users in your site but NOT by users from other sites.

When run, **Create_Site_Folder_Shortcut.sas** creates a link to your site's shared folder on FileServer. This program should only be run once. After this code runs, you will see a new folder under Folder Shortcuts named "XX_folder," where XX stands for your site's short name (e.g., AL, NY, or FL).

On FileServer, your site's shared folder's directory path is "/opt/sas/shared/sites/XX."

## Autoexec.sas for Automatic DataMart Connection (Optional)

This Autoexec file automatically creates SAS data connections whenever you log in to SAS Studio. To activate this feature, your Autoexec file needs to be updated. Once updated, the Autoexec file can call the database connection program stored in the global SAS Programs folder and create the connection at login.

Steps for updating the Autoexec file:

1. In the upper right-hand corner of SAS Studio, click the **More Application Options** button (Figure 14) to select Edit Autoexec File from the drop-down menu.

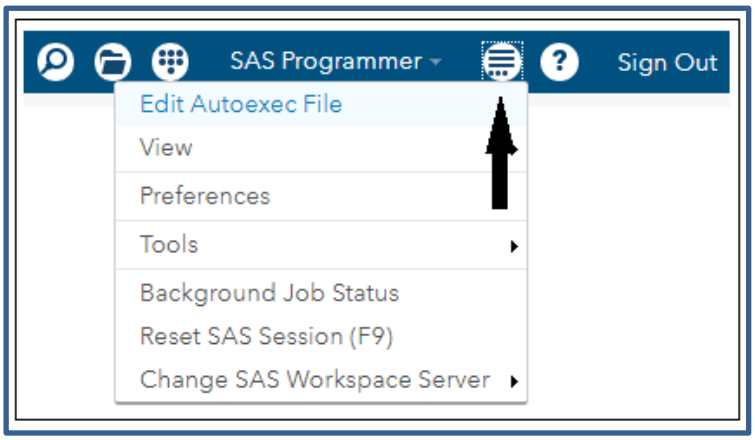



*Figure 14. Click **More Application Options**.*

2. Enter

   **%include "/opt/sas/shared/repository/programs/DataMart_Connect.sas";**

   Be sure to include the double-quote marks and ending semi-colon [;] (Figure 15), and then press **Save**.

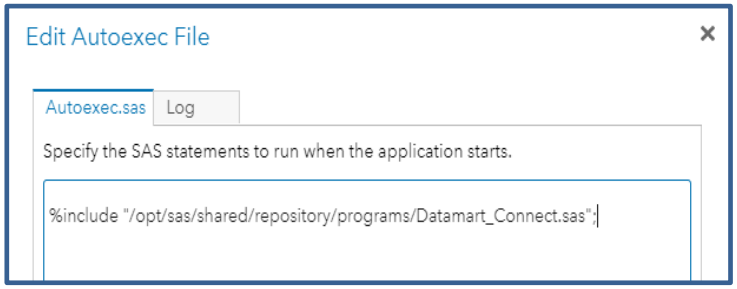



*Figure 15. Enter the location and save.*

On subsequent logins to SAS Studio, the DataMart library will be automatically allocated for use.

This completes the necessary setup steps.

# 5. How to Work with Data

## *Data Sources and Library Names*

Accessibility to data sources is based on your role and site affiliation. Most users can access the BioSense Platform database that contains their site-specific data.

For high performance and interoperability, SAS libraries (data sources) use a Microsoft Open Database Connectivity (ODBC) interface. The ODBC libraries may be predefined, or users may enter a SAS library name—called a Libname. The BioSense_Platform libname statement is included in the DataMart_Connect.sas program that is located in the SAS\programs folder.

The steps for invoking the Libname Statement follow:

1. If you have not already done so, follow the directions in Section 4 to set up the User_Info.sas and Site_Info.sas snippets to your Home folder.
2. Expand the sas-cmp1 > Folder Shortcuts > SAS > programs folder.
3. Double-click on the **DataMart_Connect.sas** program.
4. Click the **Running Man** icon to run DataMart_Connect.sas.

## *How to View Data Sources in My Libraries*

All data sources are stored under My Libraries**.** To explore these data sources, expand the My Libraries section at the bottom of the Navigator pane (Figure 16).

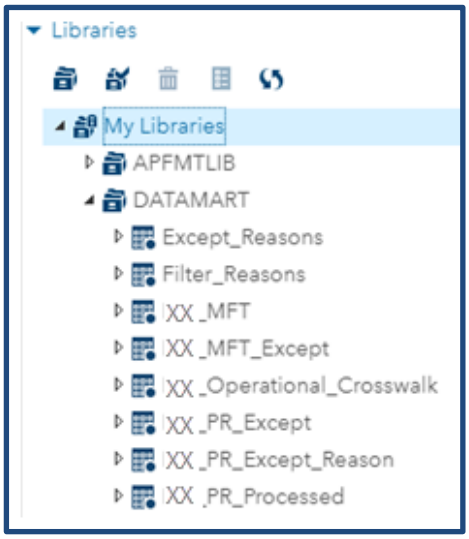



*Figure 16. Users can access files visible under My Libraries > DATAMART.*

Remember to run the DataMart_Connect.sas program to view and access the DataMart library if you haven't configured your autoexec.sas to automatically run it at login.

All users have access to the default SAS libraries and to their SAS Work library. Temporary SAS datasets are saved by default in the SAS Work library unless you designate a different library or folder. Datasets saved in SAS Work are *not permanent* and will be lost when logging out of SAS.

The DataMart library (Figure 17) will be listed only if you have access to it. The DataMart library is a read-only view into the BioSense_Platform SQL database. Users cannot save datasets to the DataMart library. Each site controls data access to its DataMart library via active directory security settings.

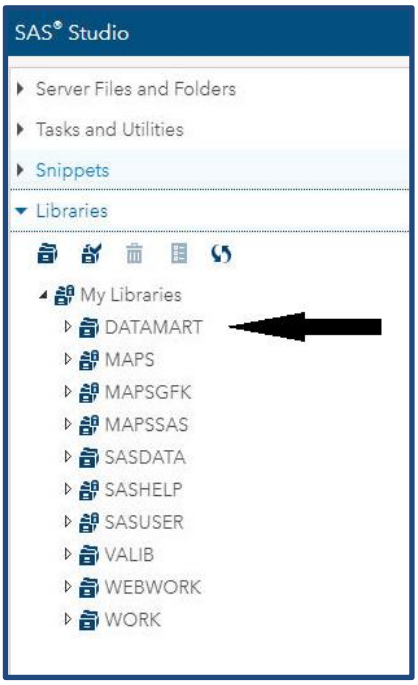



*Figure 17. DataMart is listed only after the DataMart_Connect.sas program runs.*

Here are some tips for viewing data:

- Navigation: Click the **triangles** to expand a dataset. Now you can explore dataset variables and variable properties.
- Right-click on a **variable** to see its properties (e.g., variable type, length).
- Double-click on a **dataset** to view data. Scroll left and right or up and down (Figure 18).

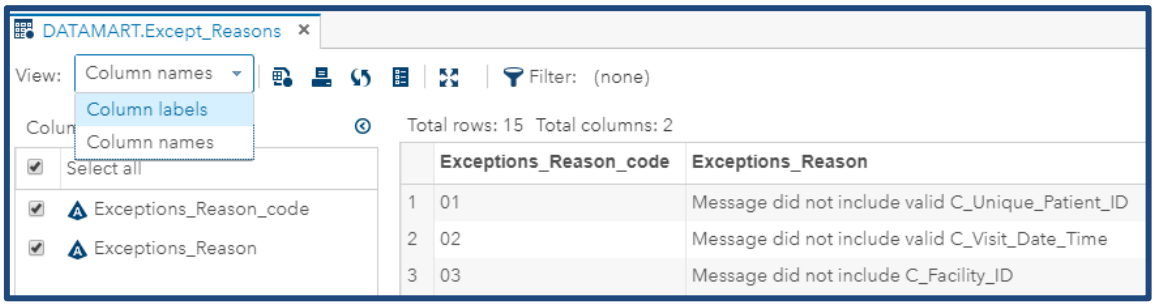



*Figure 18. Select from the View drop-down menu for column names or labels.*

- Once a dataset is open, you will be presented with options for investigating the variable properties, such as defining filters on the dataset.

## *Creating Custom Libraries and Folders*

Users can create and save SAS datasets to any location where they have permission. These locations include the Home folder and subfolders, the shared site folder, and the Public folder.

**Note:** Program and data files saved in the Public folder are viewable and editable by any user with system access. Because the content is readily visible, always use discretion when saving programs or data to this location.

### Create New Folder

To create a new folder, right-click on the **Home** folder, choose **New**, and then select **Folder** (Figure 19).

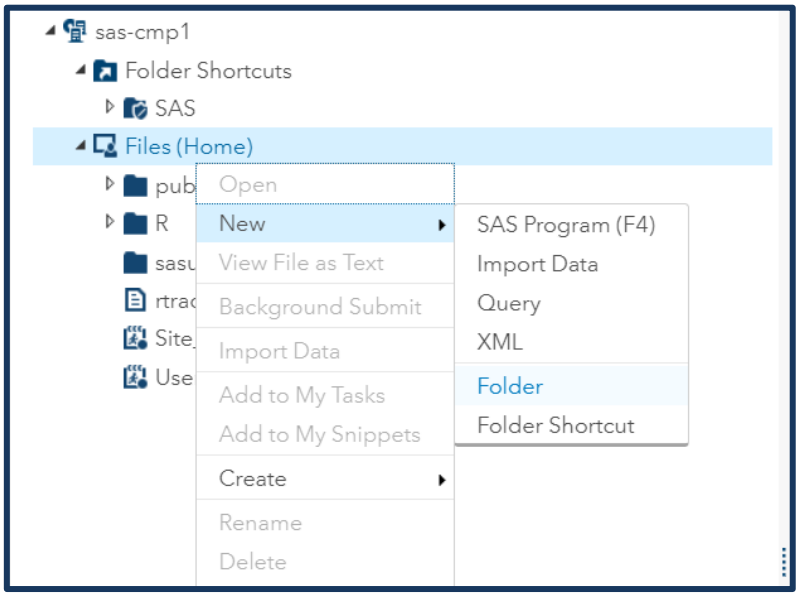



*Figure 19. Navigate to where you want to save the new folder.*

This will open a window indicating the current Location (path) and a field to enter a new folder name.

As an example, Figure 20 shows a user with UserID "nmtestuser" creating a subfolder in the sasuser.v94 folder. Note the Location (Path) is opt/sas/shared/homes/&sysuserid./sasuser.v94 where the value of &sysuserid. is the current user's ID, which in this instance is "nmtestuser."

**Note:** The dot (.) at the end of &sysuserid is required.

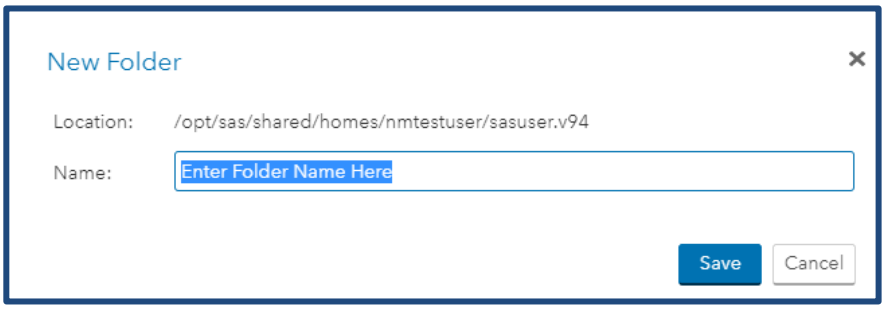



*Figure 20. The New Folder window provides the current Location (path) and field to enter the new folder name.*

Under your Home folder, you can create subfolders to store programs or other files. Figure 21 shows that a folder named Surveillance has been created in the Home folder. The Surveillance folder will be used later to store a SAS dataset.

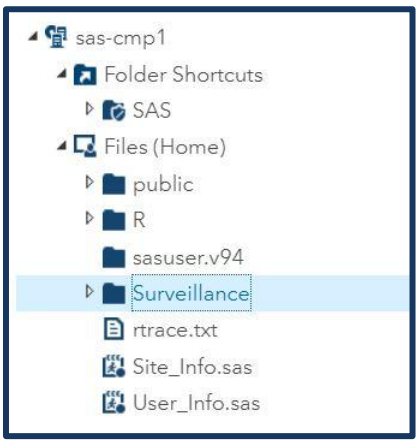



*Figure 21. Surveillance folder was created to contain a SAS dataset.*

## Custom Libnames or Libraries

Users can create custom SAS library names to save SAS datasets for later use. In the following example, a SAS dataset will be saved to the Surveillance folder (shown in Figure 21) and function as a permanent SAS dataset.

To create a custom SAS library name, you'll need to know the Linux server path to the destination folder. To find this path, select the desired destination folder, right-click on the **folder name**, and choose **Properties** (Figure 22). **Note:** You may to have to scroll down to see Properties because it is at the bottom of the sub-menu.

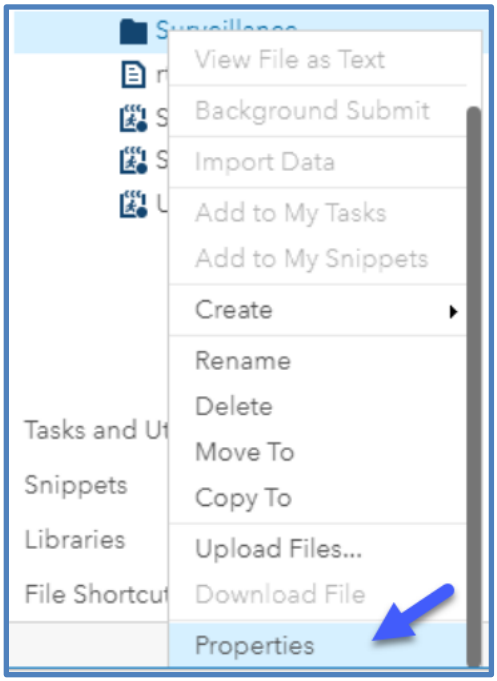



*Figure 22. Right-click and scroll down to select Properties.*

Sample code to extract a dataset and save it to the Surveillance folder is shown on the next page (Figure 24). It requires the path (Location) to the folder where you want to save the dataset (destination folder). Just highlight and copy (Ctrl-c) the location from Folder Properties box (Figure 23) and paste it (Ctrl-v) into the program code as instructed below.

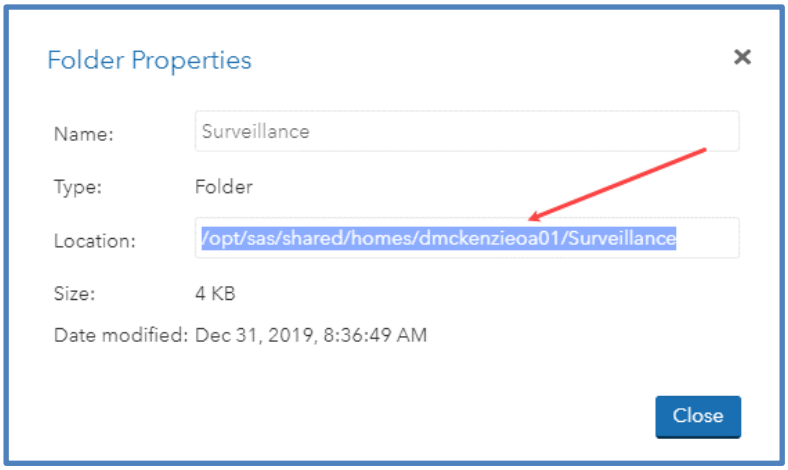
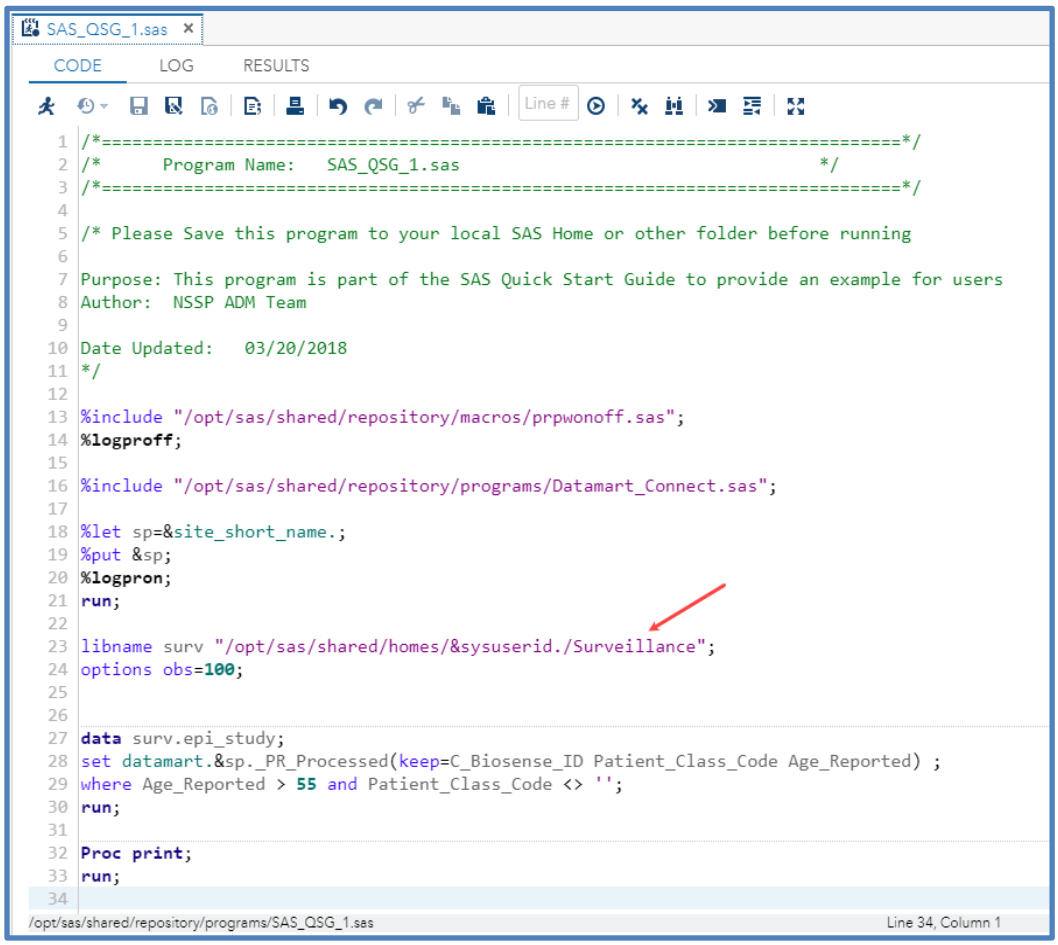



*Figure 23. Example of how to find Linux server path from Folder Properties.*

## *Creating and Running Code*

In the SAS programs folder, locate the SAS_QSG_1.sas code example (Figure 24). Explore the output of a simple data step and Proc print by using the code below. Note that the path to the Surveillance folder for the "surv" library has been supplied in this code (see **red** arrow).




```
/*==========================================================================*/
/*      Program Name:   SAS_QSG_1.sas                                       */
/*==========================================================================*/

/* Please Save this program to your local SAS Home or other folder before running

Purpose: This program is part of the SAS Quick Start Guide to provide an example for users
Author:  NSSP ADM Team

Date Updated:   03/20/2018
*/

%include "/opt/sas/shared/repository/macros/prpwonoff.sas";
%logproff;

%include "/opt/sas/shared/repository/programs/Datamart_Connect.sas";

%let sp=&site_short_name.;
%put &sp;
%logpron;
run;

libname surv "/opt/sas/shared/homes/&sysuserid./Surveillance";
options obs=100;


data surv.epi_study;
set datamart.&sp._PR_Processed(keep=C_Biosense_ID Patient_Class_Code Age_Reported) ;
where Age_Reported > 55 and Patient_Class_Code <> '';
run;

Proc print;
run;
```

*Figure 24. SAS_QSG_1.sas—Sample Program to Create Libname and Extract Data Set from DataMart*

```
libname surv "/opt/sas/shared/homes/&sysuserid./Surveillance";
options obs=100;
```

The above two lines will create the library Reference (libref) named "surv" pointing to your Surveillance subfolder with 100 observations. Note that "&sysuserid." provides your UserID, but you could also paste the path you copied from the Properties screen above.

```
data surv.epi_study;
set datamart.&sp_PR_Processed(keep=C_Biosense_ID Patient_Class_Code Age_Reported) ;
where Age_Reported > 55 and Patient_Class_Code <> '';
run;
Proc print;
run;
```

This code will populate the library "surv" with data from the DataMart based on the site_short_name variable (&sp) and the "where" criteria when this program is executed.

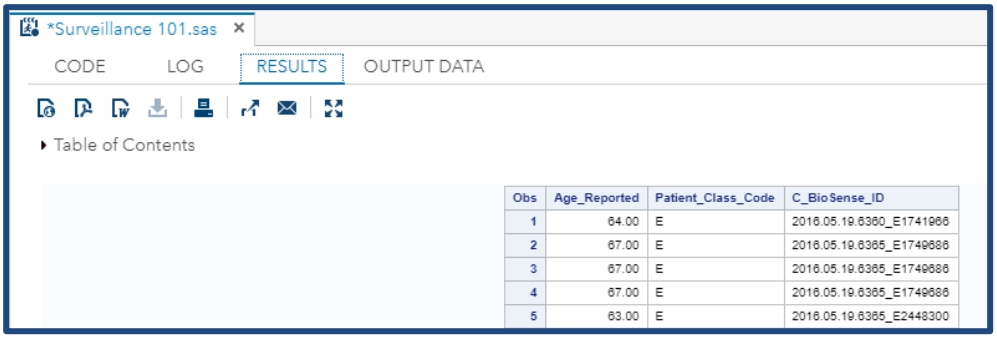
Once the SAS code is executed, a permanent SAS output dataset (epi_study.sas7bdat) will be created in your Surveillance folder (Figure 25).

**Note: SAS and R datasets that have not been accessed for 90 days may be removed from the servers. Users will be notified via email of unused datasets that will be removed if not accessed at the beginning of the following month.**

Users are assigned a maximum amount of disk space for storing datasets. Once the space allocated to a user is exhausted, the SAS job will return an error and not run. To recover storage space, delete previously created SAS datasets. See Appendix B for Drive Space Allocation Rules.

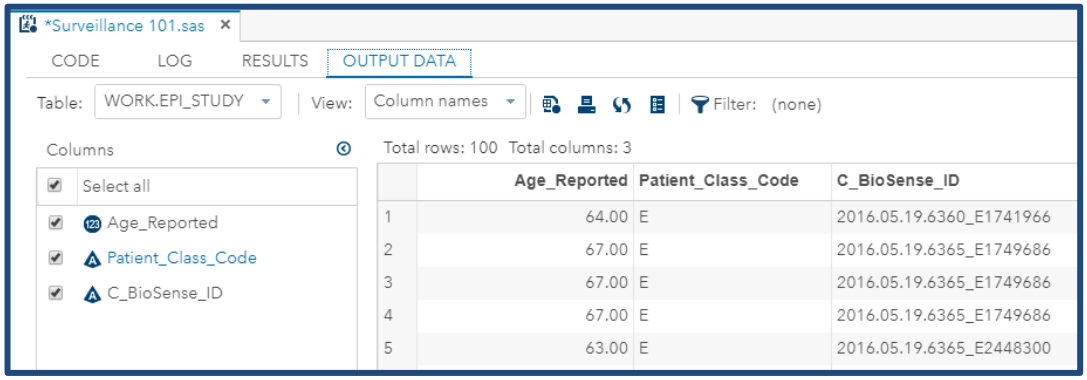
*Figure 25. Save essential datasets and access them every 90 days to avoid auto deletion.*

After the code runs, SAS Studio will display the outcome in the Results window (Figure 26) and Output window (Figure 27). Both views are available to help you understand the output from different perspectives. Click on both tabs to see the differences in presentation.




| Obs | Age_Reported | Patient_Class_Code | C_BioSense_ID |
|---|---|---|---|
| 1 | 64.00 | E | 2016.05.19.6360_E1741966 |
| 2 | 67.00 | E | 2016.05.19.6365_E1749686 |
| 3 | 67.00 | E | 2016.05.19.6365_E1749686 |
| 4 | 67.00 | E | 2016.05.19.6365_E1749686 |
| 5 | 63.00 | E | 2016.05.19.6365_E2448300 |

*Figure 26. The Results window shows data in a clean, crisp table format.*




| | Age_Reported | Patient_Class_Code | C_BioSense_ID |
|---|---|---|---|
| 1 | 64.00 | E | 2016.05.19.6360_E1741966 |
| 2 | 67.00 | E | 2016.05.19.6365_E1749686 |
| 3 | 67.00 | E | 2016.05.19.6365_E1749686 |
| 4 | 67.00 | E | 2016.05.19.6365_E1749686 |
| 5 | 63.00 | E | 2016.05.19.6365_E2448300 |

*Figure 27. SAS Studio defaults to HTML, PDF, and RFT formats, which can be changed by using the Preferences window.*

## *Saving SAS Code*

### Save to My Snippets

The My Snippets folder can be found by expanding the Snippets folder. Do this by scrolling to the bottom of the Navigator (left) pane to view the Snippets main folder. Small sections of code used repeatedly can be saved to the My Snippets subfolder there. Click on the **Add to My Snippets** icon pointed out in Figure 28 to save your code to My Snippets.

**Tip:** The My Snippets folder is a good location to save Libname statements, too.

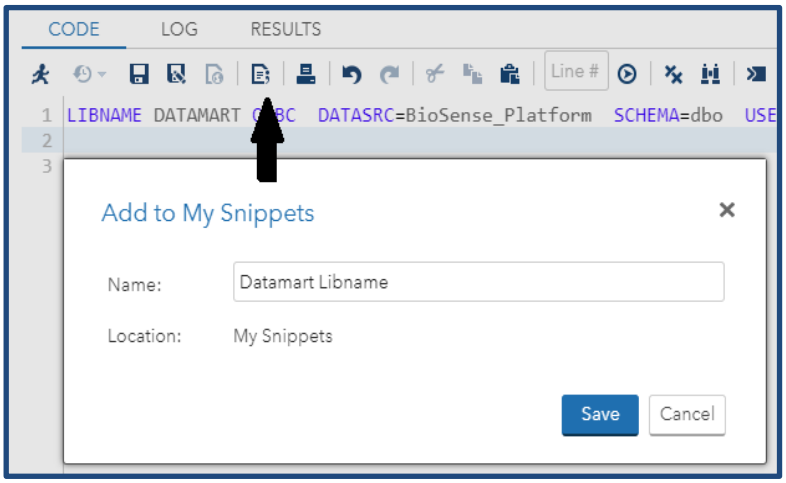



*Figure 28. Select **Add to My Snippets** to save your code to My Snippets folder.*

The Snippets folder, shown near the bottom of the Navigator pane, contains short SAS programs and useful code.

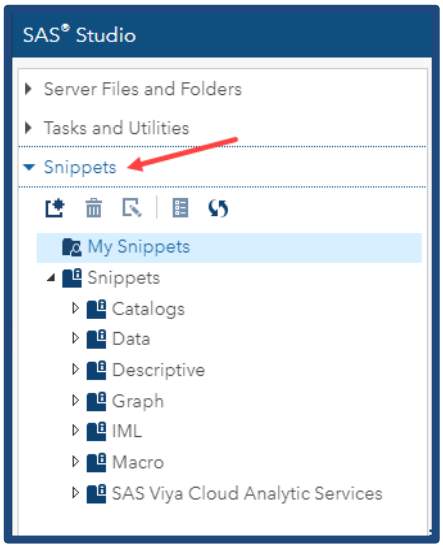



*Figure 29. Expanded View of Snippets Folder*

**Tip:** Take a moment to expand each section under Snippets to see what is available (Figure 29).

### Save to Server Files and Folders

If you repeatedly use large pieces of code (i.e., entire programs or macros), save these files to the server:

1. Click on the **Save** or **Save As** icon from the Editor pane.
2. Navigate to the folder desired; then type the program name in the field. The folder chosen can be the user's Home folder or a subfolder in the Home folder, a shared site folder, or the Public folder.

**Reminder:**

- SAS programs saved to the shared site folder are available to all your site's users. Everyone at your site may delete, modify, or execute these programs.
- SAS programs saved to the Public folder are available and editable by all SAS users across all sites.

# 6. Functionality

## *Uploading Data*

Users may upload data from their desktops or workstations to their Home (or other) folder on the file server.

1. Select the subfolder in your Home folder where your file will be uploaded; then click the **Upload** icon (Figure 30). **Note:** If you do not have a folder named Uploads, you may create one (right-click > New > Folder) or upload your data to your Home folder or a different folder.
2. Click **Choose Files** in the dialogue box (Figure 31).

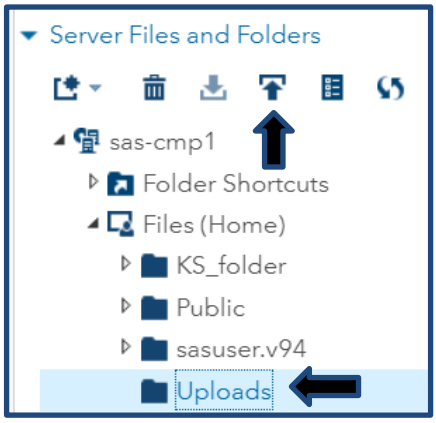



*Figure 30. Upload file(s) to your Uploads or other folder. Click the **Upload** icon to open the dialog box as shown in Figure 31.*

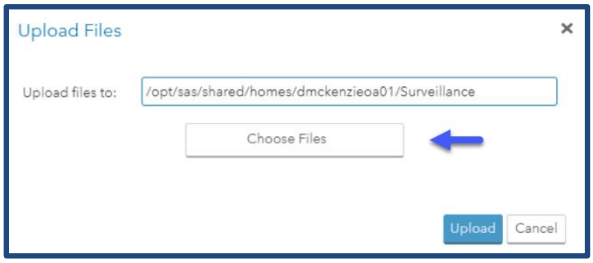



*Figure 31. Select one or more files to upload.*

3. Select the file(s) to upload by double-clicking; or, click the **Open** button.
4. Press **Upload**.

## *Importing Data*

1. Navigate to the file to be imported. This file must be accessible from the directory listing in the Navigator pane.
2. Right-click on the file; then select **Import Data** (Figure 32).
3. Click the **Run** icon.

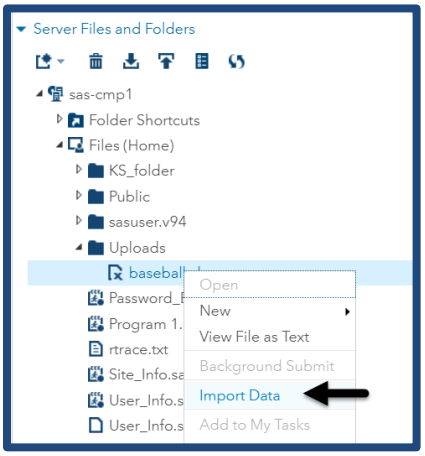



*Figure 32. Navigation for Importing Data*

The resulting report is shown in Figure 33.

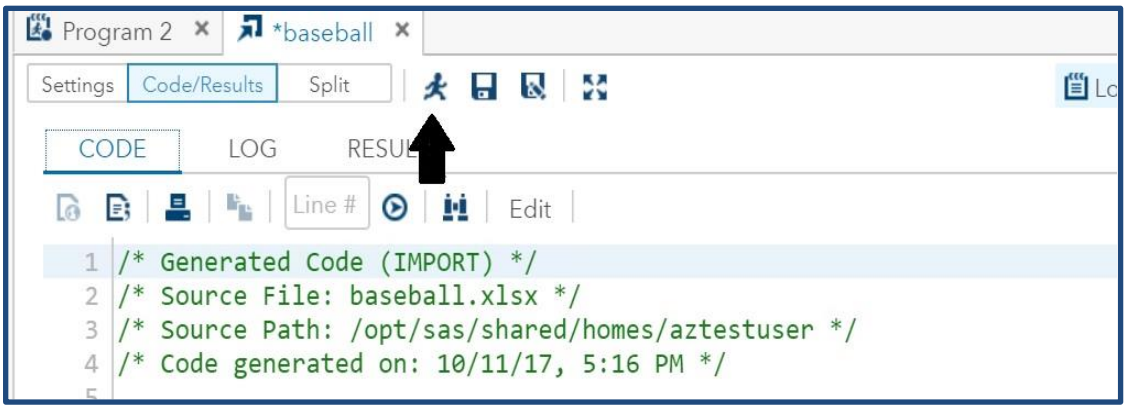



*Figure 33. Report shows imported data.*

## *Downloading Files*

Users can download SAS programs, datasets, Excel, PDF, and other files in their Home folder to their local computers. Users *cannot* download SAS datasets stored in the SAS Work or other SAS libraries.

To download a file, right-click on the file or dataset and select **Download File** (Figure 34)**.** The file will be placed in the user's Downloads folder on the local computer.

**Tip:** If you want to download a SAS Dataset from the SAS Work or other SAS libraries, you will need to use the Export option and export the SAS dataset as a CSV or other file type to your **Home** folder. The Export option copies the dataset from the SAS server to the file server where it can be downloaded.

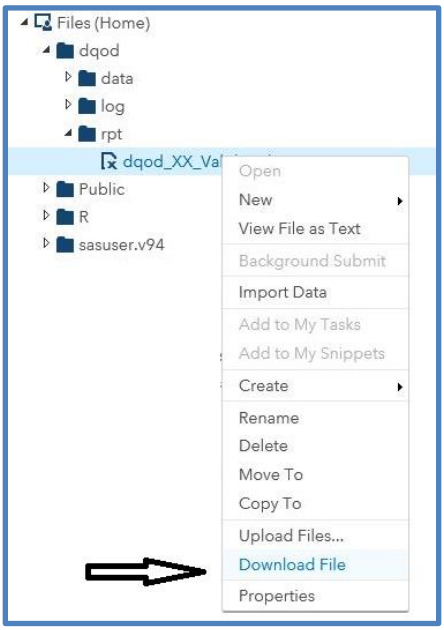



*Figure 34. Downloading a File*

## *Background Submission of SAS Code*

In SAS Studio, you can run programs in background and continue working in SAS Studio while the program runs. This is useful when your program takes a long time to complete.

The Background Submit option provides this function.
To perform a Background Submit:

1. Save changes to SAS code into your Home or another folder.
2. Right-click on the **saved program** and then click on **Background Submit** (Figure 35).

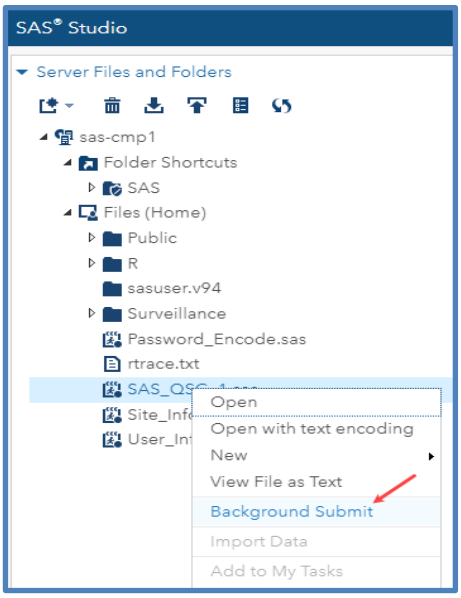



*Figure 35. After navigating to a program, select **Background Submit**.*

When a job is submitted for background processing, SAS Studio will display a Messages box in the lower right-hand corner of the screen to let you know the job is running (Figure 36). You can monitor this box by clicking the **Message** icon to see job status and access the SAS Log (Figure 37).

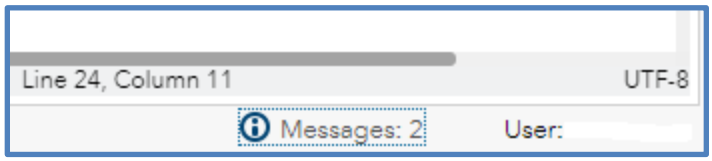



*Figure 36. Messages pop-up indicates the program is running.*

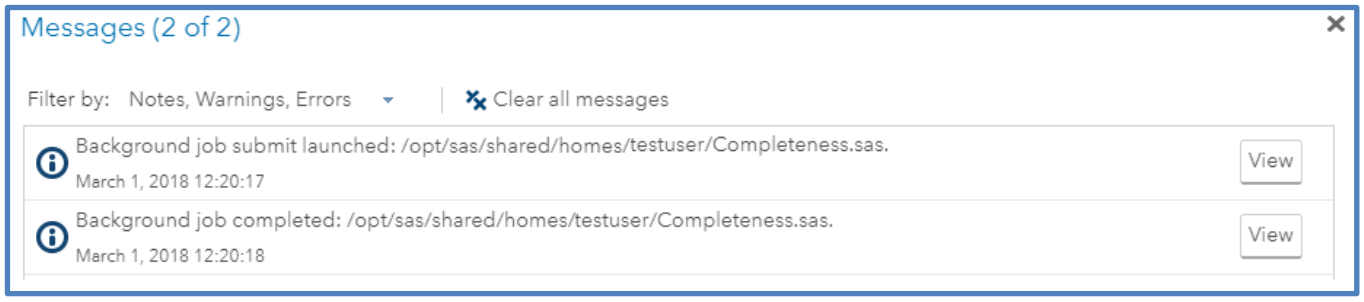



*Figure 37. Click on the **Message** icon to see job status and access log.*

# 7. DataMart Connection and Querying

## *SAS Documentation*

If you have questions about SAS programming or the options mentioned in this manual, please refer to these SAS resources:

- SAS v9.4 documentation: http://support.sas.com/software/
- *SAS Studio User's Guide*: http://support.sas.com/documentation/cdl/en/webeditorug/68828/PDF/default/webeditorug.pdf

## *Basic Methods for Connecting to the DataMart*

The SAS Studio Folder Shortcuts/SAS/programs folder contains a Data_Connection_Basics.sas program with three examples for connecting to the DataMart and querying data.

The three basic methods are:

- SAS Data Step,
- SQL Pass-Through, and
- Proc SQL.

While all SAS users are familiar with the SAS Data Step as it is a foundational concept, SQL Pass-Through and Proc SQL are not as well known or understood. There are differences between the three options and how they operate with SQL databases and SAS.

What follows is a high-level view of these three methods.

### SAS Data Step

The SAS Data Step enables the SAS programmer to make full use of SAS features and functions. When a SAS Data Step is submitted, the SAS system will try to translate as much of the SAS code as possible to SQL. SAS will then pull the requested data from the SQL DataMart into the SAS server where SAS will perform the SAS operations that it could not translate. When performing this SAS to SQL translation, SAS generates an optimized database query to pull data as efficiently as possible from SQL. Afterward, SAS resources complete the data step operations.

### SQL Pass-Through

SQL Pass-Through is an explicit pass-through to the SQL server, and SQL will execute the query as written. The pass-through procedure provides a tool to quickly access and manipulate data found in the relational views in the BioSense Platform DataMart. Data can be easily retrieved from the DataMart and written into SAS data structures as SAS datasets for further processing. Since the SQL pass-through is an explicit pass-through, it is not optimized by SAS or by SQL. This means that an inefficient or complicated SQL query will take longer to run on the SQL server and take more resources to return results to SAS for further processing.

### Proc SQL

SQL is a query language that can be implemented differently across vendors—*even when vendors adhere to the ANSI standard*. Each vendor's SQL implementation is a little different. Proc SQL is the SAS interpretation of SQL and makes use of the SAS *Libname* reference to perform an "implicit SQL pass-through." SAS will try to execute Proc SQL directly on the SQL database and perform as many calculations as possible in the SQL server rather than on the SAS server. Since Proc SQL is the SAS version of SQL there may be SQL options or functions available in other platforms that are not available in the SAS version.

### *How to Select the Appropriate SQL Pass-Through Code*

When using SAS to query data on the BioSense Platform, you should select the tools best suited for the task. Some situations require Proc SQL or SQL pass-through; for example, SQL variable names with more than 32 characters. In those instances, SQL implicit pass-through using Proc SQL lets you rename a SQL variable by using the "as" feature. In some cases, SQL Pass-Through is the best option—for example, when testing a query with limited observations (using TOP(50), etc.) in the SELECT statement. If you were to limit observations in the normal Proc SQL statement, the query would not return the desired results.

Keep in mind that the DataMart is a shared resource for all BioSense Platform users. Complicated queries can take substantial time to execute and should be avoided, if possible. Please endeavor to be efficient and keep queries to the DataMart simple. Always try to break down queries into simpler Proc SQL or SAS Data Steps. This will enable SAS to optimize the queries and, when needed, perform further joins and operations using SAS data steps and functions.

Here are examples showing each how each procedure could be used. Note that the code in all three examples returns the same dataset.

### SAS Data Step Example

You can use a SAS data step program, as shown in the Data_Connection_Basics.sas, to produce the same output as the SQL Pass-Through and Proc SQL procedures. **Note:** This program *only works* if the DataMart_Connect.sas program is included and run first.

```
options source source2 mprint mlogic symbolgen notes nocenter errors=1 compress=yes;
%logproff;
%include "/opt/sas/shared/repository/programs/Datamart_Connect.sas";
run;                        /* Datamart_Connect.sas runs first       */
%logpron;

%let sp=&site_short_name.;
%let stdate = '01Jan2018'd;
%let endate = '31Jan2018'd;

data d1;
set datamart.&sp._PR_Processed (keep=C_Biosense_Facility_ID Arrived_Date);
where Arrived_Date between &stdate. and &endate.;
run;

proc freq data=d1;
tables C_Biosense_Facility_ID /out=d2;
run;

data demo3 (keep=C_Biosense_Facility_ID  Facility_Name Count) ;
length C_Biosense_Facility_ID 8 Facility_Name $255 Count 8;
merge d2 (in=x) datamart.&sp._MFT (in=y where=(Primary_Facility='Y'));
by C_Biosense_Facility_ID;
if x;
run;

proc print data=demo3;run;
```

### SQL Pass-Through Example

Following is a sample of a SAS SQL pass-through from the **Data_Connection_Basics.sas** program. This procedure passes the SQL code directly to the SQL Server. The macro variables defined for your User ID (&UserID), password (&PW), and site short name (&sp) are included from your **User_Info.sas** snippet. (See line 6 of the code.)

```
options  source source2 mprint mlogic symbolgen notes nocenter errors=1 compress=yes;
proc sql noprint;
%include "/opt/sas/shared/repository/macros/prpwonoff.sas";
%logproff;
%include "/opt/sas/shared/homes/%scan(&sysuserid.,1,@)/User_Info.sas";          /*  ← Line 6  - User_Info.sas */
connect to odbc (datasrc='BioSense_Platform' user="&UserID." password="&PW.");
%logpron;

/* In this example, we are created a "shorter named" macro, &sp, that is set to the site you specified in the &site_short_name
  macro in User_Info.sas.

  Set your start and ending dates in &stdate and &endate. */
%let sp=&site_short_name.;
%let stdate = '2018-01-01';
%let endate = '2018-01-31';

create table demo1 as
select *
/* The section below is the SQL Pass Through */
from connection to odbc
(
select a.C_Biosense_Facility_ID, Facility_Name, count(*) as Records
from &sp._PR_Processed a inner join &sp._MFT b on a.C_Biosense_Facility_ID=b.C_Biosense_Facility_ID
where Primary_Facility='Y' and Arrived_Date between %bquote(&stdate.) and %bquote(&endate.)
group by a.C_Biosense_Facility_ID, Facility_Name);
disconnect from odbc;
quit;

proc print data=demo1;
run;
```

### Proc SQL Example

You can run a Proc SQL program using the code below. This code includes the **Datamart_Connect.sas** program to connect to the DataMart. This Proc SQL program produces the same output as the data step and SQL pass-through procedures. However, this program *only works* if the **DataMart_Connect.sas** program is run first.

```
%include "/opt/sas/shared/repository/macros/prpwonoff.sas";
%logproff;
%include "/opt/sas/shared/repository/programs/Datamart_Connect.sas";
%logpron;
options source source2 mprint mlogic symbolgen notes nocenter errors=1 compress=yes;

run;
%let sp=&site_short_name.;
%let stdate = '01Jan2018'd;
%let endate = '31Jan2018'd;

proc sql noprint;
create table demo2 as
```

```
select a.C_Biosense_Facility_ID, Facility_Name, count(*) as Records
from datamart.&sp._PR_Processed a inner join datamart.&sp._MFT b on
a.C_Biosense_Facility_ID=b.C_Biosense_Facility_ID
where Primary_Facility='Y' and Arrived_Date between &stdate. and &endate.
group by a.C_Biosense_Facility_ID, Facility_Name;
quit;

proc print data=demo2;
run;
```

**Best Practice:** Pull whole columns or records for small datasets, like the MFT table, but **do not** pull whole columns or records for large datasets like XX_PR_Processed, XX_PR_Raw, and similar tables. If you create a program using these tables, include only relevant fields in your code and, if possible, use indexed columns when subsetting. (See DataMart Table Variables with Indexes table in Section 8 below.)

# 8. DataMart on the BioSense Platform

## *DataMart Partitioning into Tables*

The DataMart is a set of SQL views partitioned by site. In general, users have access to the following tables (XX will be replaced by your site abbreviation or site short name).

- XX_MFT
- XX_MFT_Except
- XX_Operational_Crosswalk
- XX_PR_Except
- XX_PR_Except_Reason
- XX_PR_Processed
- XX_PR_Raw
- XX_Site_Contacts
- XX_ST_Except
- XX_ST_Except_Reason
- XX_ST_Processed
- XX_ST_Raw
- Except_Reasons
- Filter_Reasons

## *DataMart Indexes*

Indexes can expedite database searches. Normally, a table without indexes must search an entire table linearly. But, when an index is added, the query will only search a subset of data, which increases efficiency. (**Note:** Indexes were added to the columns below because users typically want to analyze most columns.)

Imagine that you need all license plate numbers for Toyotas in a lot. A search with no indexes would be the equivalent of checking if each car is a Toyota and recording the license plate number each time it is. But if there is an index for *manufacturer*, only Toyotas would be shown to you in the lot, decreasing query time significantly.

**Table 1. DataMart Table Variables with Indexes**

| Column Name | Production | | Staging | |
|---|---|---|---|---|
| | Processed | Raw | Processed | Raw |
| Arrived_Date_Time | Yes | Yes | Yes | Yes |
| Arrived_Date | Yes | Yes | Yes | Yes |
| Create_Raw_Date_Time | No | Yes | No | Yes |
| Feed_Name | Yes | Yes | Yes | Yes |
| File_Name | Yes | Yes | Yes | No |
| Message_ID | Yes | Yes | Yes | Yes |
| Message_Status | No | Yes | No | Yes |
| Update_Processed | No | Yes | No | Yes |
| C_Visit_Date_Time | Yes | No | Yes | No |
| C_Visit_Date | Yes | No | Yes | No |
| C_BioSense_Facility_ID | Yes | No | Yes | No |
| C_BioSense_ID | Yes | No | Yes | No |
| C_Facility_ID | Yes | No | Yes | No |
| Message_Date_Time | Yes | No | Yes | No |
| Message_Date | Yes | No | Yes | No |
| C_Unique_Patient_ID | Yes | No | Yes | No |
| Legacy_Flag | Yes | No | Yes | No |
| Create_Processed_Date_Time | Yes | No | Yes | No |
| Processed_ID | Yes | No | Yes | No |
| Site_ID | Yes | No | Yes | No |
| Update_Essence | Yes | No | Yes | No |
| Legacy_Row_Number | No | No | Yes | No |

**Arrived_Date_Time versus Arrived_Date**

Arrived_Date is derived from Arrived_Date_Time data—but it uses *only* the date portion, which makes processing faster.

Users should see improved performance for queries using Arrived_Date compared with functionally identical queries using Arrived_Date_Time.

# 9. Using ESSENCE APIs in SAS Studio

You can use ESSENCE data directly by calling the essapi.sas macro from within your SAS programs. This macro uses the ESSENCE-generated URL when you run a query (see the *BioSense Platform Quick Start Guide to Using ESSENCE* for more information).

## *Building a Query in ESSENCE*

Briefly, here are the steps to build and run a query:

1. After logging in to AMC, click the **ESSENCE** link, then log in to ESSENCE.
2. Click the **Query Portal** button on the ESSENCE Home page (Figure 38) to open the Query Wizard.

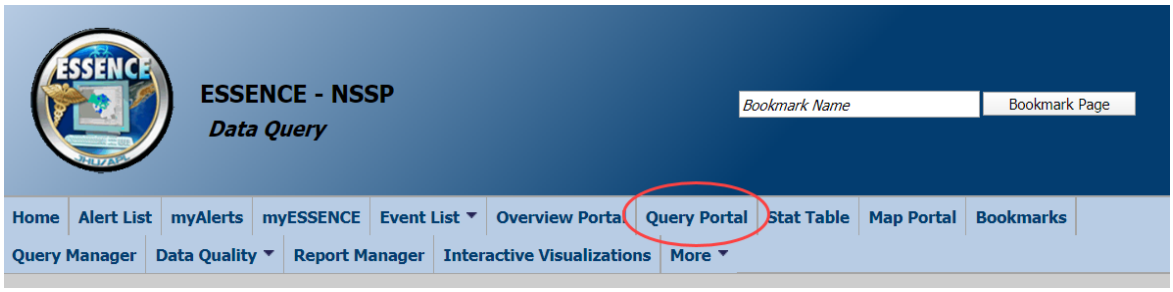



*Figure 38. ESSENCE Home Page*

3. Select parameters for your query:
   a. In the Query Wizard, select the Configuration Options for Datasource, Time Resolution, etc. (Figure 39), and then select from the Available Query Fields.

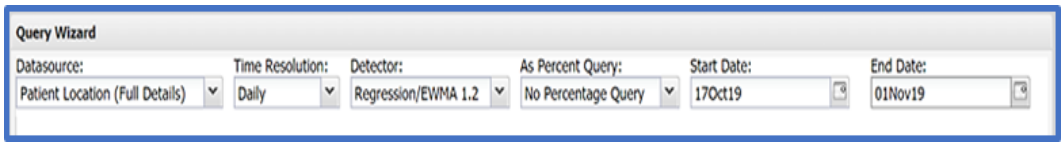



*Figure 39. ESSENCE Query Wizard*

   b. Click on the **Table Builder** button, then scroll down and select the fields from the Table Configuration Options.

4. After selecting the fields that you want in the table, scroll down again and you will see the **Create Table** and **Download to Excel** buttons. Below the buttons, you will see the field names you selected and the number of cells (rows) that your query will return.
5. When satisfied with the expected results, click the **Create Table** button.
6. Once your query executes, the table with your results will be displayed.
7. Review the results to see that your query selected the expected data.
8. If so, look in the upper left of the web page and click on the **plus sign** by Query Options. This will expand the selection.
9. Click on the **API URLs** button under Query Options.

10. The Table Builder API URLs pop-up box will be displayed, and you can select the CSV or JSON URL (Figure 40). To do this, click inside the desired URL, then type Ctrl-a > Ctrl-c to copy the entire URL.

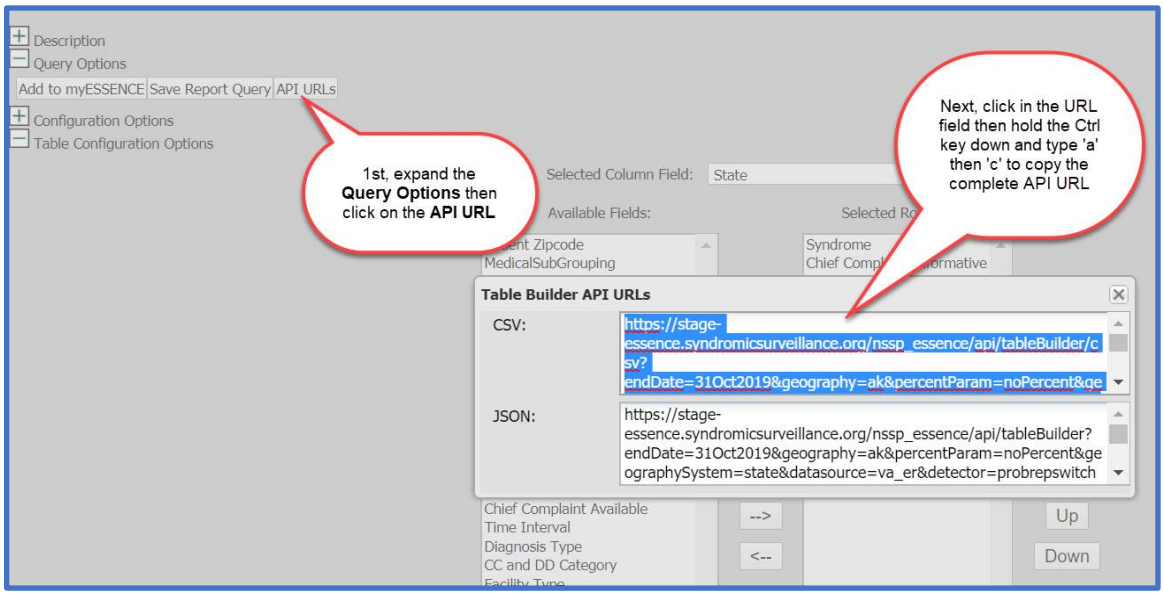



*Figure 40. Copy the API URL to your clipboard.*

## *API Pulls in SAS Studio*

After you copy the API URL (API Pull) to your computer's clipboard (Step 10 above), you can proceed directly to SAS Studio to update your own SAS program. Or, you can use the Essence_API_Basic.sas (or _Advanced) SAS program in the Folder Shortcuts/SAS/programs folder.

**Note:** If you are not going the API URL immediately, you may paste it to a Word or Notepad document and save it for later use.

1. In your browser, click on the **Home** tab in AMC, then select **SAS Studio**.
2. Log in to SAS Studio. Open the Folder Shortcuts/SAS/programs folder; right-click on **Essence_API_Basic.sas** and copy it to your SAS Home folder.

   **Note:** There is a program named Essence_API_Advanced.sas in the same programs folder, and that program offers additional parameters.
3. In your SAS Studio Home folder, open your copy of Essence_API_Basic.sas and paste (Ctrl-v) the API URL into your program where indicated by the program's internal documentation (see the example code that follows).
4. Click the **Running Man** icon to execute the program. This program calls the essapi.sas macro as mentioned above. This will execute the query URL in ESSENCE and return the results to your SAS Studio Home folder under the data subfolder.

**The program will accept all ESSENCE API output types and produce a SAS dataset that you can use in SAS Studio. All you need to do is paste the ESSENCE API URL into the program, provide the name of the SAS dataset to create, and provide the ESSENCE API output option being called (JSON, CSV, or PNG).**

*Example of an API Pull in SAS Studio*

Here is an example of the part of the Essence_API_Basic.sas program code where you drop (paste) the API URL you copied from ESSENCE Query Options and the program's output options. Please be aware that the URL shown is for illustration and is **not valid**.

**Note:** The single quote marks above and below the URL should remain on separate lines, as shown:

```
/* Drop Essence URL Here between the single quotes replacing <essence URL>
*/
%let essurl=
'
essence.syndromicsurveillance.org/nssp_essence/api/tableBuilder/csv?endDate=31Jan2020&geograp
hy=xx&percentParam=noPercent&geographySystem=state&datasource=xx_er&detector=probrepswitc
h&startDate=17Jan2020&timeResolution=daily&medicalGroupingSystem=essencesyndromes&userId=n
nnn&aqtTarget=TableBuilder&rowFields=ageCDCILI&rowFields=medicalGroupingessencesyndromes&c
olumnField=geographystate
'
;
/* SAS Output File Location will be a folder named 'data' in user's home
folder */
%let home=/opt/sas/shared/homes/%scan(&sysuserid.,1,@);
%include "&home./User_Info.sas";
%let ddir=&home./data;
libname  My_API_Lib  "&ddir";
/* --CHANGED default 'sasout' to 'My_API_Lib' */
%let essapiout= csv;     /* NA=Let essapi decide or csv, png, json. No
brackets, lower case.
--CHANGED default 'NA' to 'csv' */
%let fn= API_Out_1;   /* FN= SAS Output and files.
--CHANGED default 'ESS_API_Out' to 'API_Out_1' */
```

On the next page is a screenshot (Figure 41) of the full Essence_API_Basic.sas program with additional comments to indicate where you can make changes.

```
/*==============================================================================*/
/*              Program Name:   Essence_API_Basic.sas
/*==============================================================================
Purpose: Executes the ESSENCE API connection to enable SAS users to query the ESSENCE
application data. This is the simple example.

Author:        Paul McMurray
Date Updated:   6/12/2019
Update Log:
Dependencies:  The User_Info.sas program must be stored in the users HOME folder with the
               UserID and Password populated
Directions: Save this code to your home directory.
    1) Log into ESSENCE and create the desired query.
    2) Copy the ESSENCE API URL and replace the <essence URL> below with the API URL.
    3) The URL must be on one line by itself. KEEP THE SINGLE QUOTES above and below the URL.
    4) Choose output type: csv, png or json or NA.  The NA option will use the output
    type in the URL. The other options will transform the API into your desired output type
    5) Choose the output filename (fn) for the output type
==========================================================================================*/


%include "/opt/sas/shar                                                    ;
/* Drop Essence URL He                                                  g <essence URL>    */
%let essurl=
'
<Essence URL>
'
;
/* SAS Output File Locati                                                in user's home folder */
%let home=/opt/sas/shar
%include "&home./User
%let ddir=&home./d
libname  sasout    &ddir";


%let essapiout= NA;      /* NA=Let essapi decide or csv, png, json. No brackets, lower case */
%let fn= ESS_API_Out;    /* FN= SAS Output and files. Change if needed*/
%include "/opt/sas/shared/repository/macros/Essapi.sas";
%essapi;
run;
```

*From Directions (above)*
Step 2 & 3. Paste your API URL over "<Essence URL>"

You can change the name of the Library to something meaningful

Step 4. Use "NA" or choose Output Type

Step 5. Change the file name, if needed

*Figure 41. Essence_API_Basic.sas Program with Comments*

# 10. Operational Considerations

## *File Storage and Retention Policy*

SAS Studio creates and uses two types of datasets: Temporary or Work datasets and permanent datasets. SAS Work datasets are stored on SAS servers and deleted once a user logs out of SAS Studio. When permanent SAS datasets are saved to a user's Home folder or shared site folder, the datasets are saved on a file server separate from the SAS servers. This file server (Fileserver) is a shared resource for the NSSP BioSense Platform user community that everyone can use to store output datasets and programming.

On the first day of each month, a scheduled job will examine file metadata on the file server and identify files over 10 Gigabytes (GB) in size that have not been accessed, used, or updated in the last 90 days. System users will receive an email notifying them of files marked for deletion, and users will have 10 days to log in to SAS or RStudio to access or update the flagged files. Files that have not been accessed or updated within 10 days of the emailed notice will be moved to an offline storage location and deleted on the first day of the following month.

If users are not able to access or update their files within the 10-day window and want to retain the files, the users should submit a service desk ticket (support.syndromicsurveillance.org) requesting the files be restored to their Home folder. The special request to restore the file(s) must be received before the first day of the following month. **Once files have been deleted from the server, these files cannot be recovered.**

## *Fileserver Storage Allocation*

Fileserver limits each user to a maximum of 500 GB of drive space. A user may temporarily exceed the 500 GB limit by up to 20 percent, however, if the 500 GB limit is exceeded, the user will receive a warning message and have 7 days to reduce storage. See Appendix B for detailed rules.

**Users are encouraged to be good stewards of their SAS dataset storage on Fileserver and only save datasets that are essential.**

If you attempt to save a dataset that exceeds the allocated space for your Home folder and your site's shared folder, try freeing space by deleting old or unused files. Other alternatives for managing dataset size follow:

- Use SAS KEEP and DROP dataset options to retain variables needed ONLY for processing or analysis.
- Consider *not keeping* long text fields such as Chief Complaint or Admit Reason.
- Use the SAS COMPRESS option to reduce the size of the dataset.

## *SAS Memory and Datasets: Out-of-Memory Errors*

The SAS servers have limited memory. When using Proc Print or other SAS tools to visualize data, SAS Studio will load the data into memory for display. If the dataset is too large, SAS Studio will generate an out-of-memory error.

An out-of-memory error may also be triggered if the dataset being analyzed is too large for SAS to hold in memory.

In situations like these, try adjusting the code to limit dataset size or running the program in the background (see Background Submission of SAS Code). You may also attempt to run the program when fewer people are using the system.

# Appendix A: NSSP-provided SAS Code, Macros, and Snippets

*This appendix is a library of SAS code, macros, and snippets developed by the National Syndromic Surveillance Program (NSSP). In the macros, programs, and snippets folders, all files are set to read-only, however, you can copy or save them to your Home folder.*

## *Folder Shortcuts/SAS/programs*

SAS code may need to be updated to run for your site. Please save code to your Home or other folder before making changes. The following programs can be found under Folder Shortcuts/SAS/programs:

### Completeness.sas

*Purpose*: This program creates a visit-level count/percent completeness report for select variables. *Percent completeness* of a column is based on the percent of visits that have a value in the column that is also found in at least one message or record associated with that visit.

The program will:

- Output an Excel file that contains the count and percent completeness for a designated period.
- Create directories for the programs, report, and data. (The program directory is optional.)

*User Input*: Users must provide the start and end dates for the analysis period beginning about line 28 of the SAS program.

- Start date variable: %let stdt='2018-01-01';
- End date variable: %let endt='2018-02-01';

### Completeness_Exceptions.sas

*Purpose*: This program creates a count of excepted (error) records.

This program will create an Excel file that can be downloaded to your local computer

*User Input*: Users must provide the start and end dates for the analysis period beginning about line 18 of the SAS program.

- Start date variable: %let stdt='2020-01-01';
- End date variable: %let endt='2020-01-31';
- The user may also designate Production (PR) or Staging (ST) environment, with Production being the default.

### Datamart_Connect.sas

*Purpose*: Executes the DataMart Connection statement so that SAS users can query their DataMart data. When run successfully, the DATAMART library will show up under My Libraries. Users must always run this program to connect to the BioSense Platform database on the DataMart. For details, review the description in Section 4, Setup, about the **Datamart_Connect.sas** program.

### DataQuality_OnDemand.sas

*Purpose*: Initializes global variables (e.g., start_date, end_date) and starts program dqodmain.sas. The dqodmain.sas program calls multiple other programs depending on configuration and type of report selected (see sas-cmp1/Folder Shortcuts/programs/DQOD_secondary_programs for possible programs called by dqodmain.sas).

Copy this program to your Home folder, then do the following:

1. Set the site_id and site_short_name to the values for your site.
2. Set the Start and End Date variables to the date range over which you would like to see Data Quality (DQ) Reports.
   **CAVEAT**: Consider setting the date range to no more than a few days at first to get a sense of run time.
3. OPTIONAL: On LINE 42, you can change dolog=N to dolog=Y if you want detailed logs saved.
4. Set the code for the report type (the default is Completeness report).

### Data_Connection_Basics.sas

*Purpose*: Provides code to demonstrate the various ways of accessing NSSP data (see Section 7. DataMart Connection and Querying: Basic Methods for Connecting to the DataMart for details).

### Essence_API_Basic.sas

*Purpose*: This is a basic program for users who want to simply copy an ESSENCE API URL, paste it into a program, and run it with standard settings.

Copy this program to your Home folder, then go to ESSENCE, run a query, and copy the API URL (see Section 9 for instructions). Then paste it into this program and click the **Running Man** button to execute it.

**Key Points for Users:**

- In this SAS program, the ESSENCE API URL must be on one single line with single quotes on the line above and below it.
- Output data is written to a folder named “data” in the user’s Home folder.
- The output dataset type defaults to the type contained in the API URL. CSV, PNG, or JSON may be explicitly specified. NA indicates the default output will be used (see %let essapiout= NA;).
- The output dataset name defaults to “ESS_API_Out” in the created data subfolder. User may specify a different file name (see **%let** fn= ESS_API_Out;).

### Essence_API_Advanced.sas

*Purpose:* This advanced version of the ESSENCE API program supports repeated use and customization.
**Note:** Save a copy of this program to your Home folder before using it.

**Key Points for Users:**

- In this SAS program, the ESSENCE API must be on one single line with single quotes on the line above and below.
- Output data is written to a folder named “data” in the user’s Home folder.

- The output dataset type defaults to the type contained in the API URL. CSV, PNG, or JSON may be specified (%let essapiout= NA; is the default).
- The output dataset name defaults to ESS_API_Out in the created data subfolder. User may specify a different file name (see %let fn= ESS_API_Out;).
- Date range is specified by modifying the startdate and enddate variables (DDMMMYY).

By default, the API query will return the same fields as selected on the Data Details webpage. However, this can be overridden by specifying one or multiple field parameters. For example, to return only the Age and ChiefComplaintParsed fields, add &field=age&field=ChiefComplaintParsed to the URL.

### Pilot_Lag.sas

*Purpose*: This program allows you to adjust the lag period. Lag period refers to the time between a message being sent and its arrival on the BioSense Platform.

Lag periods are calculated in days by Feed and Facility. Locate the **WHERE** clause in the SQL query (line 62 currently) and insert the desired start date (Arrived_Date_Time). When using the CAST() function, enter the date as a string in the 'YYYY-MM-DD' format including the single quote marks as follows:

```
where Primary_Facility='Y' and cast (a.Arrived_Date_time as date) >= '2020-01-01'
```

### Primary_Facility_Count.sas

*Purpose*: This program uses SQL pass-through to query for primary facilities, name, Facility_ids and counts. Locate the **WHERE** clause in the code (currently on line 41) and replace the 'YYYY-MM-DD' with the desired Arrived_Date. When using the CAST() function, please enter the date as a string in the 'YYYY-MM-DD' format including the single quote marks as follows:

```
where Primary_Facility='Y' and Arrived_Date = '2020-01-01'
```

## *Folder Shortcuts/SAS/macros*

### mssqlexe.sas

*Purpose*: This macro lets novice users submit SQL code to the MS SQL Server using SAS SQL Pass-Through—all without learning the coding details. A basic knowledge of SAS macro coding is assumed, and the user must make sure that references to MSSQL tables are fully qualified.

Here is an example of a "fully qualified" reference for the XX_PR_Processed table:

biosense_platform.dbo.XX_PR_Processed. Note that "biosense_platform.dbo." is the required prefix to fully qualify a table name in the MS SQL Server.

**Note:** This SAS macro is used in the mssqlexec_snippet.sas snippet that is discussed in the section below. You do not need to save the mssqlexe.sas macro to your **Home** folder, since it is called directly from the SAS\Macros folder. Do not modify it because it is used by other programs and snippets.

### Nsspdates.sas

*Purpose*: Creates SAS macro dates for SQL Server and SAS processing when macro variables for start and end dates (***bst*** and ***ben***) are set. This macro also sets SAS options for maximum logging from macros and listing of source code. This macro is to be executed from within your SAS program.

To use the Nsspdates.sas macro, copy these lines from the comments in the macro or this document and embed them in your SAS program. Replace the YYYYMMDD date strings in the %let statements below with the desired start and end dates:

```
%include '/opt/sas/shared/repository/macros/nsspoptions.sas';
%let bst= YYYYMMDD ;                                    /* Start Date, e.g., 20191201 */
%let ben= YYYYMMDD ;                                    /* End Date, e.g.,  2019123   */
%include '/opt/sas/shared/repository/macros/nsspdates.sas';
run;
```

**Note:** Remember to include the last line ("run;") when you copy the above. It is required.

### Nsspoptions.sas

*Purpose*: This macro turns on SAS macro logging and listing of source code. Examine the internal documentation in this macro for the specific options that are turned on and what they do.

To use this macro, paste the following line into your SAS program:

```
%include '/opt/sas/shared/repository/macros/nsspoptions.sas';
```

**Note:** If you have already copied the macro Nsspdates.sas to your program, this macro was included.

### Prpwonoff.sas

*Purpose*: This is the "print password on/off" macro. The Prpwonoff.sas macro suppresses printing of username and password to the SAS log. This macro is in almost every provided program.

You do not need to save or copy this program to your Home folder. To use this macro, simply insert the following code into your SAS program:

```
%include "/opt/sas/shared/repository/macros/prpwonoff.sas";  /* this is the macro code */
%logproff;        /* turns off the password printing        */
%logpron;         /* turns back on the password printing */
```

- Place the %logproff macro *before* the line of code that contains the username and password or the macros for those variables.
- Place the %logpron macro *after* the line of code that contains the username and password or the macros for those variables.

### *Folder Shortcuts/SAS/snippets*

SAS code may need to be updated to run for your site. Please save code to your Home or other folder before making changes. The following programs can be found under Folder Shortcuts/SAS/snippets:

#### Create_Site_Folder_Shortcut.sas

*Purpose*: This snippet creates a link to the site-shared folder on FileServer. This snippet should only be run one time when you set up SAS Studio.

This snippet requires that you have saved either the User_Info.sas or Site_Info.sas snippet in your Home folder and have populated your site's short name there.

**CAVEAT**: Programming and data files saved in your site's shared folder may be run by all users on your site. Users from other sites will not have access.

#### Folder_Create.sas

*Purpose*: This snippet will create folders within SAS code. Users can copy and paste this snippet into their programs and change the names of the folders to be created. This snippet is already included in some NSSP-provided SAS programs. If the user only wants to create a Reports folder in the Home folder, the default "%let dir" entry will work.

```
%let dir = /opt/sas/shared/homes/&sysuserid./reports;
```

#### mssqlexec_snippet.sas

*Purpose*: This snippet demonstrates how to use the mssqlexe.sas macro in your own programs.
It executes the mssqlexe.sas macro referenced in the SAS Macros section and provides the framework to supply the values (parameters) required.

To use, you should save this snippet to your **Home** folder and modify:

- The start (bst) and end (ben) dates
- If desired (Figure 42), you can update the internal %macro sqlselect to point to the Production or Staging database (PR or ST) and the Processed or Exceptions table.

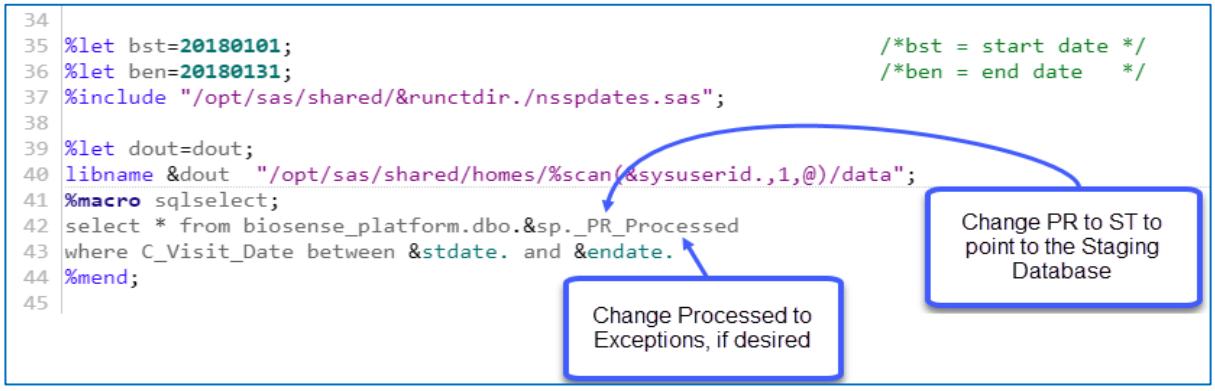



*Figure 42. mssqlexec.sas snippet—Change Database or Record Type*

### Password_Encode.sas

*Purpose*: This snippet is used to encrypt your SAS password so that it may be safely stored in your User_Info.sas file for use by other SAS macros, programs, and snippets.

Run this program when you set up SAS Studio and when your password changes (see Password_Encode.sas—Required for details).

### Site_Info.sas

*Purpose*: This snippet provides parameters used by other programs, macros, and other snippets. Save the Site_Info.sas snippet to the Home folder and fill in the user's Site ID number, site name, site short name, and other variables (see Site_Info.sas—Required for details).

Most NSSP-provided SAS programs require this short snippet to minimize user-required changes when running the provided programs and rely on Site_Info.sas for this information.

### User_Info.sas

*Purpose*: This snippet provides parameters used by programs, macros, and other snippets. Save the User_Info.sas snippet to the Home folder and fill in the user's BioSense Platform username and encrypted password (see **Password_Encode.sas** snippet).

Most NSSP-provided SAS programs require this short program to minimize user-required changes when running the provided programs.

# Appendix B: Fileserver Drive Space Allocation Rules

## *File Storage*

SAS Studio creates and uses two types of datasets: Temporary or Work datasets and permanent datasets.

SAS Work datasets are stored on SAS servers and deleted once a user logs out of SAS Studio. When permanent SAS datasets are saved to a user's Home folder or shared Site folder, the datasets are saved on a file server separate from the SAS servers. This file server is a shared resource for the NSSP BioSense Platform user community that everyone can use to store output datasets and programming files.

## *Drive Space*

Each user is allocated 500 Gigabytes (GB) of drive space on Fileserver. This allocation can be used to store files in the user's **Home** folder and their shared site folder.

All files a user creates count as part of the user's allocated storage, including:

- Files created *manually* by writing programs; saving existing macros, programs, and snippets; or uploading files; and
- Files created *programmatically* by pulling data from other sources, such as ESSENCE.

When users exceed the 500 GB maximum, they will receive a warning message and will have 7 days to reduce their total storage below the maximum. During this grace period, the users will continue to receive warning messages but will not be prevented from creating or extending files.

This space allocation maximum has a buffer, so a user may temporarily use up to 600 GB if work in progress requires more data or workspace than expected. However, 600 GB is a hard limit and, if exceeded, further attempts to write to the disk will fail with an error message, even if the grace period has not yet expired.

If users exceeding the 500 GB limit allow the 7-day grace period to expire without reducing their storage to under 500 GB, they will *not be permitted* to create new files or extend existing files until space usage is adequately reduced.

Effectively, there is no limit to the size of your shared site folder. If each user who contributes files to the shared site folder remains under the individual drive space allocation, this folder can be added to as required. This means that individual users do not need to know about the space usage of other users in their site before writing data or saving files to the shared site folder.

Here are two examples of how drive space allocation works.

- You have 300 GB of files stored in your Home folder and a 500 GB quota across all storage folders. Thinking you can avoid hitting your quota, you copy 250 GB of data to the shared site folder. Because your total usage is now 550 GB and exceeds your 500 GB quota, the system sends you a warning. With only 50 GB remaining in your maximum allowed storage (hard limit), if you continue to write data to the shared site folder or to your Home folder, you will likely exceed this hard limit of 600 GB. You cannot write more data once you reach 600 GB.
- You and two others are using the same shared site folder, and everyone has fewer than 100 GB of data stored in individual Home folders. If we each write a program that saves 300 GB of data to our shared site folder, we would <u>each</u> have allocated space usage of 400 GB (which is under our individual limits), but our shared site folder would total 900 GB of data. This is not a quota violation.

# Appendix C: Best Practices

- Because drive space is limited, only save datasets that are essential. You are assigned a maximum amount of disk space for storing datasets and other files. When a user's allocated space is exhausted, the SAS job will return an error and not run. To recover storage space, delete previously created SAS datasets.

- Run programs with large datasets in background or at times when others are unlikely to run analyses.

- Always remember to re-encrypt your AMC password when you change it; then update User_Info.sas file with your new encrypted password.

- Your site's shared folder (XX_folder) is available to all users who are members of your site, whereas the Public folder is available to members of any site. Keep this in mind when saving a dataset or other files to a shared folder.

- When working with a large dataset, try adjusting your SAS code to limit dataset size or running the program in the background to prevent out-of-memory errors. Try running the program when fewer people are using the system.

- Only your site administrator can grant access to SAS Studio.

  **Note:** When a site administrator adds SAS to a user's account, there is a 24-hour lag from adding access to the user's profile in AMC to access being granted. If site administrators enable SAS for themselves or another user, please consider that there will be a wait before access is available.

  You will receive an email from the NSSP team when setup is complete. If you cannot access SAS Studio after 24 hours or did not receive the email, please put in a support request at

  support.syndromicsurveillance.org

- Here are popular browsers in general use with our recommendations for SAS Studio:
  - Google Chrome—First choice—recommended/supported;
  - Microsoft Edge—Alternate;
  - Firefox—Alternate (not for use by CDC users);
  - Microsoft Internet Explorer (IE) has limited functionality and is <u>not recommended</u>;
  - Apple Safari is not supported.