

Enhancing Your Skillset with SAS® OnDemand for Academics (ODA) Software

Kirk Paul Lafler, sasNerd

Abstract

The free cloud-based SAS OnDemand for Academics (ODA) software is an exciting development for SAS users and learners around the world! The software includes Base SAS, SAS Studio, SAS/STAT, SAS/GRAPH, SAS/ETS, SAS/OR, SAS/IML, SAS/QC, SAS/CONNECT, SAS Enterprise Miner, and SAS/ACCESS to PC Files. SAS ODA offers users with extensive learning opportunities to enhance skills for career development and advancement using data access, data manipulation, data management, programming techniques, analytics, data visualization, and statistical analysis capabilities. Topics include an introduction and overview of SAS OnDemand for Academics (ODA) software, demonstration of SAS Studio features, and programming examples to showcase this exciting software suite.

Introduction

SAS® OnDemand for Academics (ODA) is amazing software that can be freely used by students, educators, professionals, and anyone who desires to learn how to use SAS software. What does SAS ODA offer? SAS ODA's cloud-based user interface, SAS Studio, users can learn how to access data and perform amazing extract, transform, and load (ETL) activities, data analysis, statistical analysis, reporting and data visualization, and other processing using SAS ODA with a common web browser. This paper introduces how to create a SAS Profile and register to use SAS ODA and SAS Studio; launch SAS Studio; SAS Studio's powerful point-and-click user interface including the Home screen; the Navigation Pane consisting of Files and Folders, Tasks and Utilities, and Libraries; the Work area consisting of the SAS Programmer window; and data access for accessing SAS (SAS7BDAT) data sets, importing tab-delimited text (TSV) data files, comma-separated value (CSV) data files, and Excel (XLSX) data files. SAS ODA and SAS Studio also gives users the ability to access and process JavaScript Object Notation (JSON) data files – the replacement of XML.

Data Set and Data Files Used in the Examples

The example data set and data files presented and used in this paper include the Heart SAS data set, Heart tab-delimited text data file, Heart comma separated values (CSV) data file, Heart Excel (XLSX) data file, and Heart JavaScript Object Notation (JSON) data file. The contents of the Heart data set along with the definition of the various other data files are displayed, below.

File Type Definitions	
SAS Data Set (SAS7BDAT)	A proprietary SAS (SAS7BDAT) data format that contains data values that are created, organized, and stored as a table of rows and columns in a SAS library (e.g., WORK, SASUSER, and User-assigned) where processing is performed by SAS software.
Tab-delimited Text (TSV) Data File	A text data format known as, a tab-separated values (TSV) data file, is created and used by spreadsheet programs and other software. It consists of rows of data values containing one or more fields (or columns) where each field is separated (or delimited) with a tab character.
Comma-separated Values (CSV) Data File	A text data format that contains one or more fields (or columns) where each field is separated (or delimited) with a comma.
Excel (XLSX) Data File	A proprietary Microsoft data format used to format, organize, and compute data in a spreadsheet.
JavaScript Object Notation (JSON)	An open standard data format that is used to transmit web application data.

HEART_MEDCENTER Data File (5 Rows and 5 Variables)

MedCtrID	MedicalCenter	City	State	Zip
CA92101	San Diego Medical Center	San Diego	CA	92101
CA92037	La Jolla Heart Institute	La Jolla	CA	92037
CA90025	Los Angeles Medical Center	Los Angeles	CA	90025
CA94105	San Francisco Medical Center	San Francisco	CA	94105
NV89109	Las Vegas Health Center	Las Vegas	NV	89109

HEART Data File (5,209 Rows and 17 Variables)

Status	Cause of Death	Age CHD Diagnosed	Sex	Age at Start	Height	Weight	Diastolic	Systolic	Metropolitan Relative Weight	Smoking	Age at Death	Cholesterol	Cholesterol Status	Blood Pressure Status	Weight Status	Smoking Status
Dead	Other		Female	29	62.5	140	78	124	121	0	55	.		Normal	Overweight	Non-smoker
Dead	Cancer		Female	41	59.75	194	92	144	183	0	57	181	Desirable	High	Overweight	Non-smoker
Alive			Female	57	62.25	132	90	170	114	10	.	250	High	High	Overweight	Moderate (6-15)
Alive			Female	39	65.75	158	80	128	123	0	.	242	High	Normal	Overweight	Non-smoker
Alive			Male	42	66	156	76	110	116	20	.	281	High	Optimal	Overweight	Heavy (16-25)
Alive			Female	58	61.75	131	92	176	117	0	.	196	Desirable	High	Overweight	Non-smoker
Alive			Female	36	64.75	136	80	112	110	15	.	196	Desirable	Normal	Overweight	Moderate (6-15)
Dead	Other		Male	53	65.5	130	80	114	99	0	77	276	High	Normal	Normal	Non-smoker
Alive			Male	35	71	194	68	132	124	0	.	211	Borderline	Normal	Overweight	Non-smoker
Dead	Cerebral Vascular Disease		Male	52	62.5	129	78	124	106	5	82	284	High	Normal	Normal	Light (1-5)
Alive			Male	39	66.25	179	76	128	133	30	.	225	Borderline	Normal	Overweight	Very Heavy (> 25)
Alive			57 Male	33	64.25	151	68	108	118	0	.	221	Borderline	Optimal	Overweight	Non-smoker
Alive			55 Male	33	70	174	90	142	114	0	.	188	Desirable	High	Overweight	Non-smoker
Alive			79 Male	57	67.25	165	76	128	118	15	.	.		Normal	Overweight	Moderate (6-15)
Alive			66 Male	44	69	155	90	130	105	30	.	292	High	High	Normal	Very Heavy (> 25)
Alive			Female	37	64.5	134	76	120	108	10	.	196	Desirable	Normal	Normal	Moderate (6-15)
Alive			Male	40	66.25	151	72	132	112	30	.	192	Desirable	Normal	Overweight	Very Heavy (> 25)
Dead	Cancer		56 Male	56	67.25	122	72	120	87	15	72	194	Desirable	Normal	Underweight	Moderate (6-15)
Alive			Female	42	67.75	162	96	138	119	1	.	200	Borderline	High	Overweight	Light (1-5)
Dead	Coronary Heart Disease		74 Male	46	66.5	157	84	142	116	30	76	233	Borderline	High	Overweight	Very Heavy (> 25)
Alive			Female	37	66.25	148	78	110	112	15	.	192	Desirable	Optimal	Overweight	Moderate (6-15)
Alive			Female	45	64	147	74	120	119	5	.	209	Borderline	Normal	Overweight	Light (1-5)
Alive			Female	59	65.75	156	74	156	122	0	.	200	Borderline	High	Overweight	Non-smoker
Alive			Female	36	63.75	122	84	132	102	0	.	184	Desirable	Normal	Normal	Non-smoker
Alive			Female	50	67.5	185	88	150	136	15	.	228	Borderline	High	Overweight	Moderate (6-15)
Alive			Female	35	66	123	76	132	93	0	.	150	Desirable	Normal	Normal	Non-smoker
Alive			Male	42	72.25	182	78	136	113	0	.	221	Borderline	Normal	Overweight	Non-smoker
Dead	Coronary Heart Disease		71 Female	49	60.5	153	110	196	140	5	73	221	Borderline	High	Overweight	Light (1-5)

SODA and SAS Studio Features

SAS OnDemand for Academics (SODA) and SAS Studio provide users with tools to enhance learners’ skillsets for the digital world, now and in the future. SODA and SAS Studio offer amazing features for learners everywhere:

- Learning – Anyone interested in learning more about the SAS software has access to analytics, AI, and Machine Learning tools.
- Sharing – Learners can share code, data sets / data files, free cloud services and storage.
- Point-and-click – Learners can access a host of features without the need to program.
- Snippets – Learners can plug in one or more code snippets directly into their code.
- Performance – Use the computing power of the SAS cloud instead of your own machine which means learners do not experience any decrease in performance while SAS code executes.
- Price – Best of all learners have all these features for free.
- Online Learning – To help learners achieve success, there’s a vibrant online learning community.
- Tutorials – “Free” tutorial videos are available to help learners get started painlessly with the features available in SAS OnDemand for Academics and SAS Studio.

For Learners – Access Massive Open Online Content (MOOCs)

Software and resources are available for developing analytical knowledge and include:

- Free teaching and curriculum development content.
- Free e-learning courses:
 - ✓ [SAS® Programming 1](#)
 - ✓ [Statistics 1 – Introduction to ANOVA, Regression, and Logistic Regression](#)
- An online interactive community – <https://communities.sas.com/t5/SAS-Software-for-Learning/bd-p/sas-learning>.
- SAS Tutorial Videos – <https://video.sas.com/category/videos/how-to-tutorials>.
- The SAS Cloud – Upload up to 5GB of data for learning purposes.

SAS® Programming 1 Course: Essentials

The screenshot displays the SAS Training Console interface for the 'SAS® Programming 1: Essentials' course. The page is structured as follows:

- Header:** Includes the SAS logo, 'SUPPORT' text, and navigation icons (USA flag, mail, grid, search).
- Breadcrumbs:** Home > Training
- Navigation:** 'Training Console' with a search bar containing 'Find a course by' and 'Search courses'. A 'My Training' link is also present.
- Right Side:** 'United States' dropdown menu, 'Worldwide Training' link, and 'OFFERS FOR THIS COURSE' section.
- Course Title:** 'SAS® Programming 1: Essentials' with a 'FREE E-LEARNING' badge.
- Course Description:** Explains the course is for users wanting to learn SAS programming for data science, machine learning, and AI. It is a prerequisite for many other SAS courses.
- Prerequisites:** 'SAS Enterprise Guide® 1: Querying and Reporting' course.
- Learning how to:**
 - Use SAS Studio and SAS Enterprise Guide to write and submit SAS programs.
 - Access SAS, Microsoft Excel, and text data.
 - Explore and validate data.
 - Prepare data by subsetting rows and computing new columns.
 - Analyze and report on data.
 - Export data and results to Excel, PDF, and other formats.
 - Use SQL in SAS to query and join tables.
- Who should attend:** Anyone starting to write SAS programs.
- Formats available:**
 - Classroom: 3 days
 - Live Web Classroom: 5 half-day session(s)
 - e-Learning: 21 hours/180 day license
- System Requirements:** System Requirements (linked).
- Buttons:** 'View demo', 'Learn More', 'Earn a Digital Badge', 'Private Team Training'.
- Footer:** 'Ready for this course?', 'View learning path', 'Suggest course date', 'See program details'.

Statistics 1 Course: Introduction to ANOVA, Regression, and Logistic Regression


SUPPORT






Home > Training

Training Console

Find a course by ▼

Search courses 🔍

My Training

 Cart
 800-727-0025
 Contact us

- **Training**
 - ▶ My Training
 - ▶ Find a course
 - ▶ e-learning
 - ▶ Live Web Classes
 - ▶ Locations
 - ▶ Learning formats
 - ▶ Discounts
 - ▶ Free Tutorials
 - ▶ Ask the Expert
 - ▶ Academy for Data Science
 - ▶ Learning Subscription
- SAS Books
- Certification

Statistics 1: Introduction to ANOVA, Regression, and Logistic Regression

FREE E-LEARNING

- Overview
- Prerequisites
- Course Outline

This introductory course is for SAS software users who perform statistical analyses using SAS/STAT software. The focus is on t tests, ANOVA, and linear regression, and includes a brief introduction to logistic regression. This course (or equivalent knowledge) is a prerequisite to many of the courses in the statistical analysis curriculum.

A more advanced treatment of ANOVA and regression occurs in the [Statistics 2: ANOVA and Regression](#) course. A more advanced treatment of logistic regression occurs in the [Categorical Data Analysis Using Logistic Regression](#) course and the [Predictive Modeling Using Logistic Regression](#) course.

This course can help prepare you for the following certification exam(s): [SAS Certified Clinical Trials Programming Using SAS 9](#), [SAS Statistical Business Analysis Using SAS 9: Regression and Modeling](#).

Learn how to

- Generate descriptive statistics and explore data with graphs.
- Perform analysis of variance and apply multiple comparison techniques.
- Perform linear regression and assess the assumptions.
- Use regression model selection techniques to aid in the choice of predictor variables in multiple regression.
- Use diagnostic statistics to assess statistical assumptions and identify potential outliers in multiple regression.
- Use chi-square statistics to detect associations among categorical variables.
- Fit a multiple logistic regression model.
- Score new data using developed models.

Who should attend

Statisticians, researchers, and business analysts who use SAS programming to generate analyses using either continuous or categorical response (dependent) variables

Formats available	Standard Duration		View demo
<ul style="list-style-type: none">  Classroom: 3 days  Live Web Classroom: 6 half-day session(s) System Requirements  e-Learning: FREE 21 hours/180 day license System Requirements 			View demo

United States 
Worldwide Training ▼



LEARN

SAS

Earn a Digital Badge

Private Team Training

Schedule virtually or in-person, at our location or yours.

[Learn more](#)

- STAY INFORMED
- ◆ [Subscribe to e-newsletter](#)
 - ◆ [Read SAS Users blog](#)



A Vibrant SAS® Software Learning Community

SAS Software for Learning Community
Welcome to the exclusive online community for all SAS learners.

Find your SAS answers | This board | Start A Topic

Home > Learn SAS > SAS Software for Learning Community

SAS Explore begins Sept 27! Hundreds of breakout sessions and free training! Register now.

Options | Bookmark | Subscribe | RSS Feed

FAQ for students and independent learners
Learning SAS? This FAQ allows you to access, in one succinct page, many resources you need to start your SAS journey.
[Check it out!](#)

Resources for students and others learning SAS
As you begin your journey to learn SAS, keep these links handy:
 > How to get fast, helpful community answers
 > FAQ for students and independent learners
 > FAQ for educators
 > SAS Viya for Learners (for students & educators only)
 > SAS OnDemand for Academics FAQ (available for everyone)

Post Title	Author	Date	Replies	Views
How do I use a file saved in my library	anthonymsmerlin	yesterday	12	96
i+1 statement in Do Loops	Naive_help	Friday	5	101
Why don't I see my data as a choice when attemptin...	TKC1	Thursday	2	140
Table Error	JoshuaG	2 weeks ago	8	105

SAS Skill Builder for Students:

SAS® Tutorial Videos

Getting Started with SAS Studio

In this video, you get started with programming in SAS Studio. You view a data table, write and submit SAS code, view the log and results, and use interactive features to quickly generate graphs and statistical analyses.

[Learn about SAS Training - Programming path](#)

f t in

CATEGORIES

Trending

Products & Solutions

Analytics in Action

How To Tutorials

SAS Add-In for Microsoft Office

Administration & Architecture

Advanced Analytics

How To Tutorials

1-15 of 591

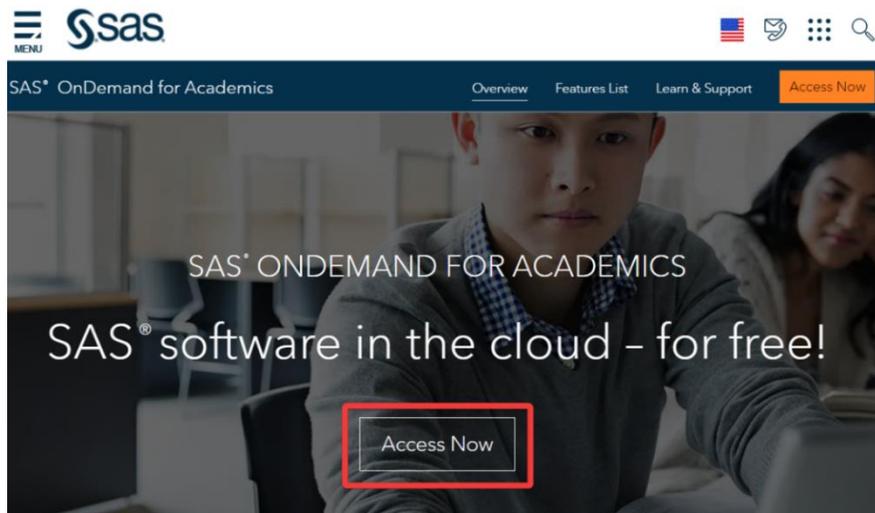
Search

Sort By: Most Viewed

SNAPSHOT DEMO

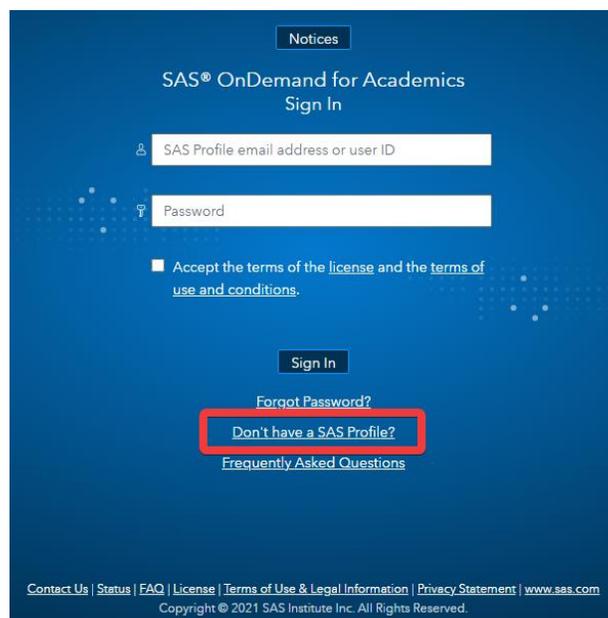
SAS Studio – A Cloud-based Integrated Development Environment (IDE)

SAS OnDemand for Academics (ODA) provides learners and educators with a comprehensive cloud- and web-based user interface called SAS Studio. SAS Studio provides numerous user-friendly features to help users become more productive while using the SAS ODA. To begin, open one of the supported web browsers (e.g., Google Chrome, Mozilla Firefox or Apple Safari) to access SAS ODA by clicking the following hyperlink, https://www.sas.com/en_us/software/on-demand-for-academics.html, and then clicking the “Access Now” as shown, below.

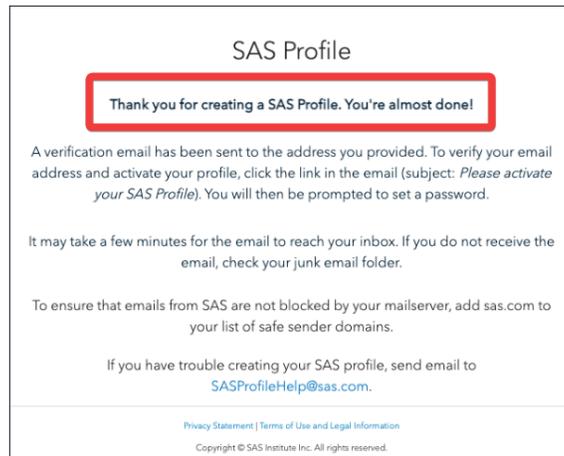


Step 1 – Create a SAS Profile

The SAS OnDemand for Academics (ODA) Sign In dialog window will display as shown, below. Before accessing SAS ODA, you will need to create a SAS Profile. If you are already a SAS user and have set up a SAS profile account, then you can proceed to register to use SAS ODA. By entering your SAS Profile email address or user ID along with your Password in the designated boxes. If you are a new SAS user or have never created a SAS Profile then you will need to click the “Don’t have a SAS Profile?” hyperlink shown, below.

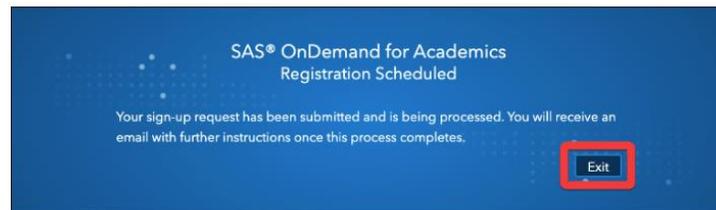
The image shows the SAS OnDemand for Academics Sign In dialog window. At the top is a 'Notices' button. Below it is the title 'SAS OnDemand for Academics Sign In'. There are two input fields: 'SAS Profile email address or user ID' and 'Password'. Below the fields is a checkbox labeled 'Accept the terms of the license and the terms of use and conditions.' with links to 'license' and 'terms of use and conditions'. Below the checkbox is a 'Sign In' button. A red rectangular box highlights the link 'Don't have a SAS Profile?'. Below this link are the links 'Forgot Password?' and 'Frequently Asked Questions'. At the bottom of the dialog are links for 'Contact Us', 'Status', 'FAQ', 'License', 'Terms of Use & Legal Information', 'Privacy Statement', and 'www.sas.com'. The footer text reads 'Copyright © 2021 SAS Institute Inc. All Rights Reserved.'

After entering the requested information to create your SAS Profile, a message will display on your screen, below.



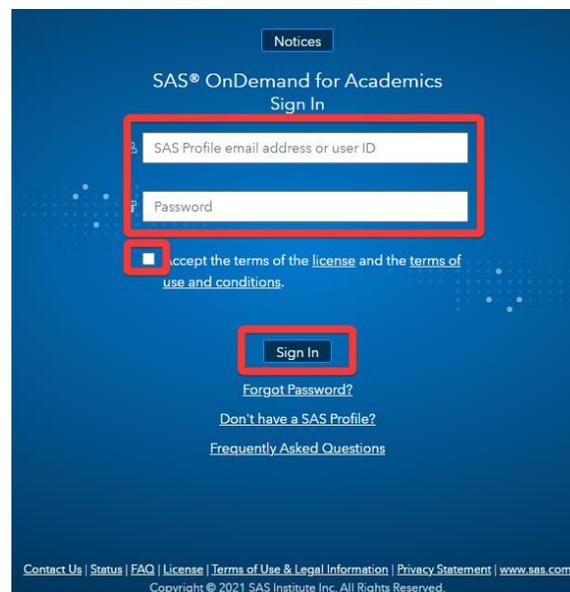
Step 2 – Register to Use SAS OnDemand for Academics (ODA)

After successfully creating a SAS Profile, you can register to use SAS OnDemand for Academics (ODA). You should then return to the SAS OnDemand for Academics (ODA) page where you will be prompted to select your home region and click **Submit**. A confirmation page will then appear like the one shown, below, and finalize the process by clicking the **Exit** button. A follow-up email from SAS will be sent with your User ID so you can then enter this User ID or your email address to access SAS ODA.



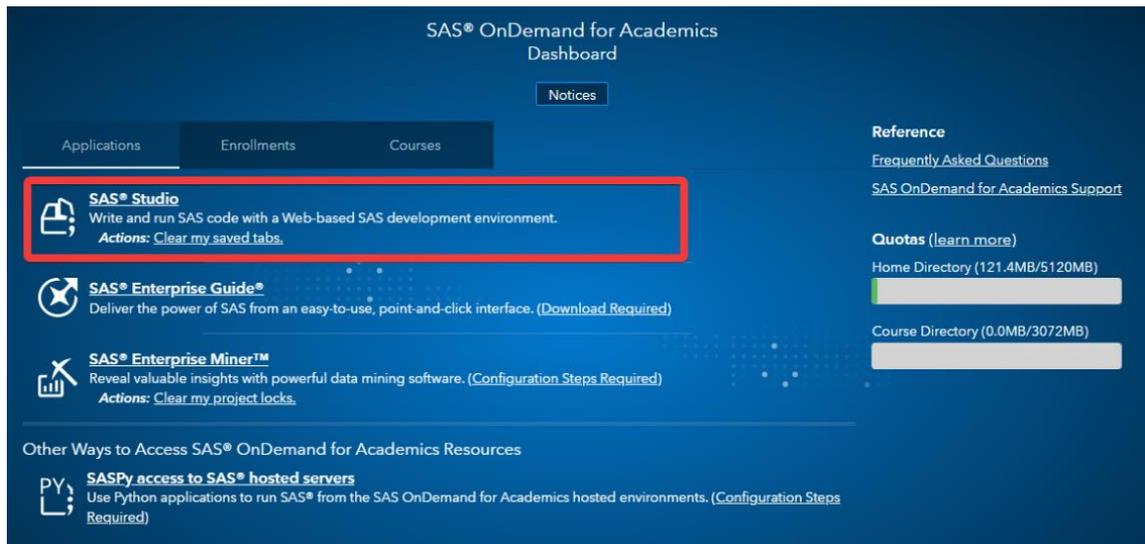
Signing Into SAS OnDemand for Academics (ODA) and Accessing SAS Studio

After successfully registering to use SAS OnDemand for Academics (ODA), you can then sign in with your User ID and password credentials in the appropriate fields, check the box associated with accepting the terms of the license and the terms of use and conditions, and click the **Sign In** button as shown, below.



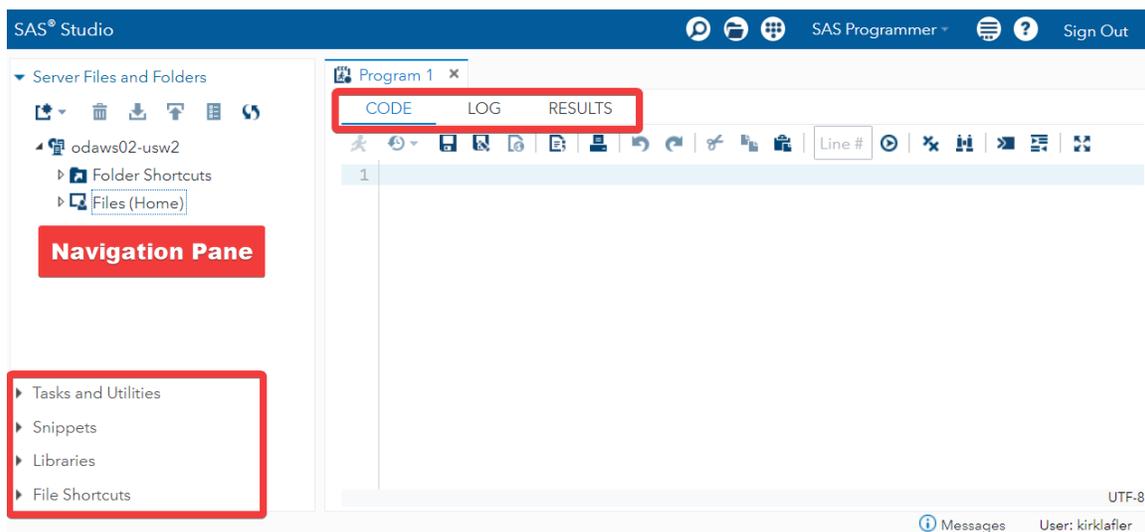
SAS OnDemand for Academics (SODA) Dashboard

The SAS ODA dashboard will then display with important information about your account including permissions, enrollments, courses, self-help references, and storage space quotas. When ready, click the **SAS Studio** hyperlink shown, below.



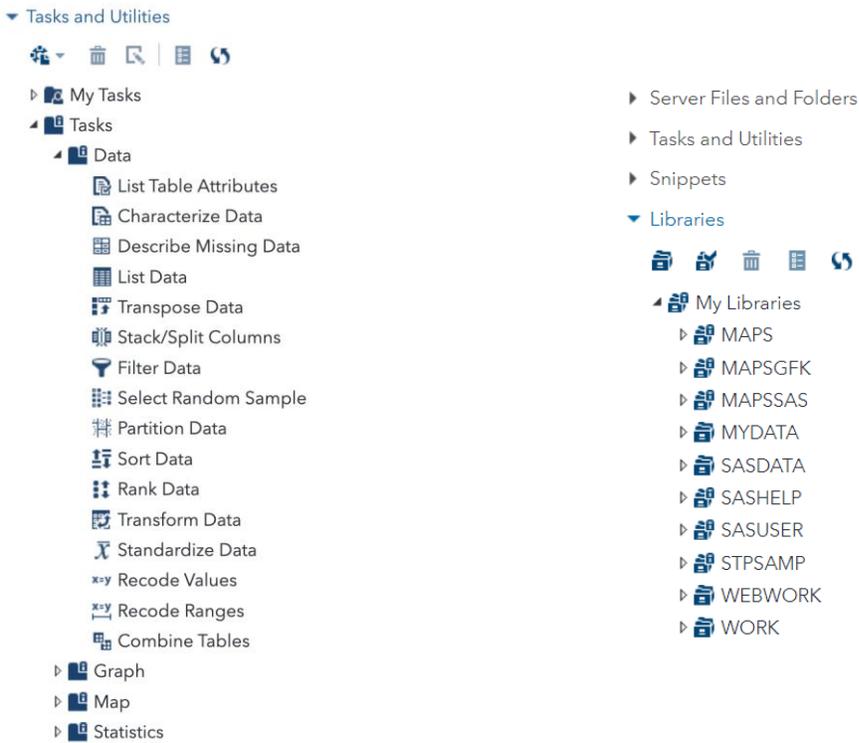
SAS Studio User Interface

SAS Studio's powerful and easy-to-use interface provides users with a comprehensive integrated development environment (IDE). The SAS Studio interface is divided into several parts that help make user interaction easier, Navigation pane, and Work area more convenient. Let's explore the different parts of SAS Studio to better understand what they're used for. After signing into SAS Studio, **Server Files and Folders** provide users with the ability to upload local data files. There are four more dropdown menus below Server Files and Folders, two of which will be emphasized, **Tasks and Utilities**, and **Libraries**.



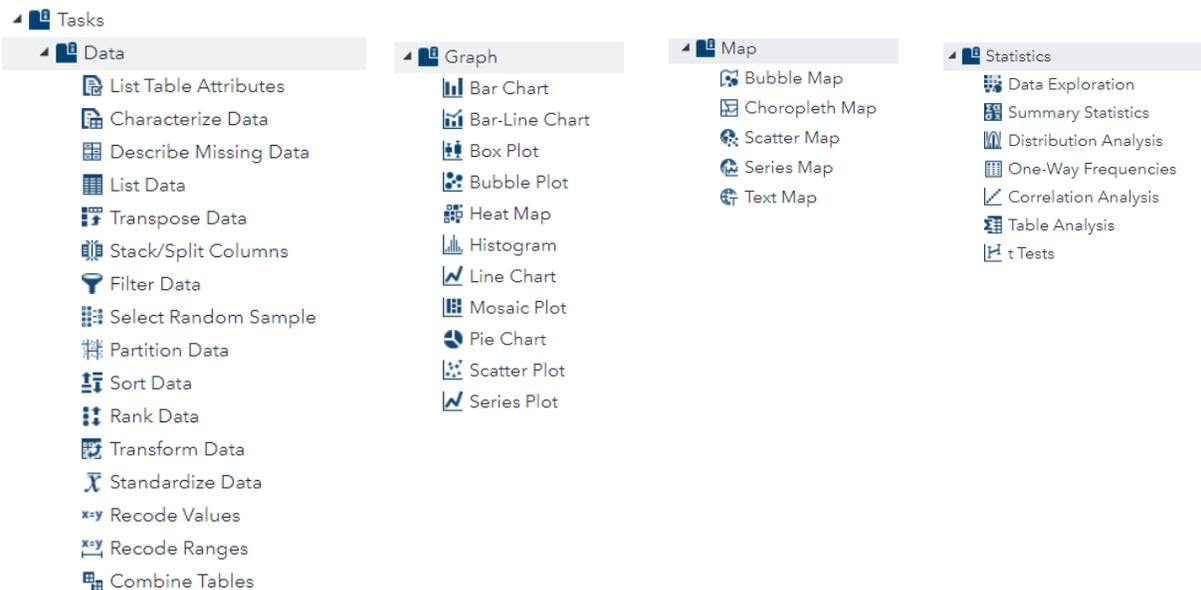
Tasks – SAS Studio Navigation Pane – Data and Libraries

When clicking on the Navigation pane's drop-down arrow next to Tasks, more options expand as shown, below. SAS Studio's built-in point-and-click interface helps make working with SAS data sets, text-delimited data files, CSV data files, Excel data files, JSON data files, and program code easier with a powerful toolkit of predefined tasks that enable users to list table attributes, characterize data, describe missing data, and much more. access data sources, perform data analytics, and several other tasks.



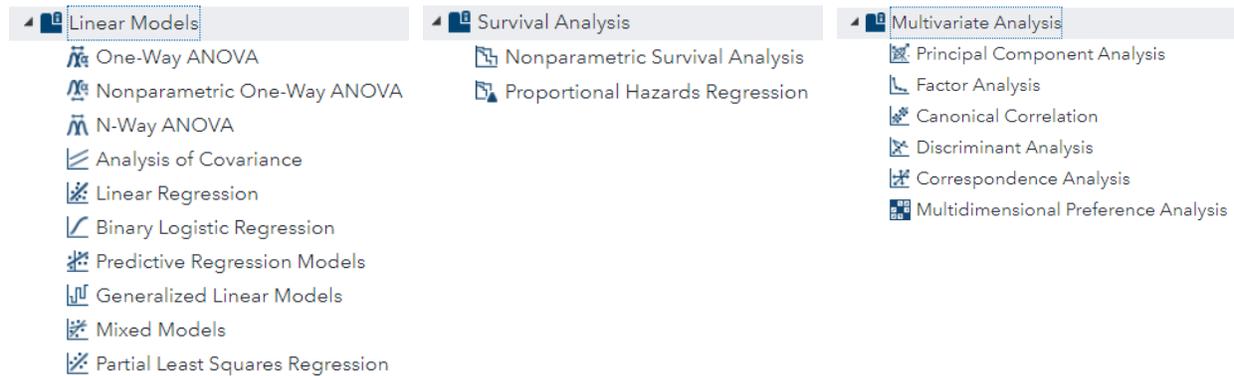
Tasks – Data, Graph, Map, and Statistics

SAS Studio supports data, graph, map, and statistics tasks as appears, below.



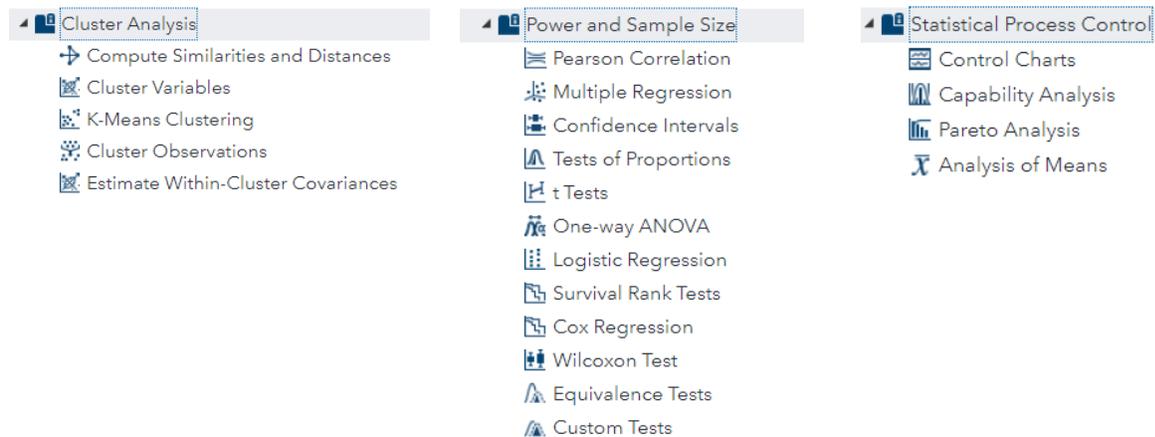
Tasks – Linear Models, Survival, and Multivariate Analysis

SAS Studio supports linear models, survival, and multivariate analysis tasks as appears, below.



Tasks – Cluster Analysis, Power and Sample Size, and Statistical Process Control

SAS Studio supports cluster analysis, power and sample size, and statistical process tasks as appears, below.



Tasks – Combinatorics and Probability, Data Mining, and Econometrics

SAS Studio supports combinatorics and probability, data mining, and econometrics tasks as appears, below.



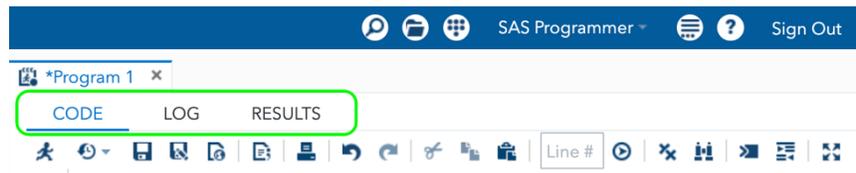
Another Navigation Pane drop-down is Libraries. A SAS library is a collection of one or more SAS data sets that are stored, referenced, and processed by SAS software. Specifically, the SASHELP library stores a variety of data sets for students, faculty, and SAS learners to explore and learn from. We will demonstrate using the SASHELP library and the HEART data set in several examples in this paper.

Program Window: Code, Log, and Results

The SAS Studio Program window provides users with Code, Log, and Results tabs. A brief description of each tab appears, below.

CODE Editor

SAS Studio includes a color-coded, syntax-checking editor for editing new or existing SAS programs. The editor includes a number of features such as autocompletion, automatic formatting, and pop-up syntax help. With the code editor, you can write, run, and save SAS programs.

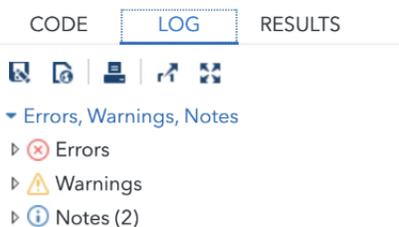


There are a number of icons in the Code tab toolbar. Most of these icons will display tooltips or their functionality when hovering the mouse on them. Below are descriptions of some commonly used SAS Studio-specific icons:

Icon	Tooltip	Execution
	Run all or selected codes	Executes all lines or highlighted lines of codes in the Code window.
	Submission history	Displays a history of executed statements and will rerun the code once selected on the previous statement.
	Save program	Save all codes.
	Program summary	An HTML file that opens in a separate browser tab includes information about the program execution, the complete SAS source code, the complete SAS log, and the results.
	Clear all code	Clears all code in the current program’s code editor.

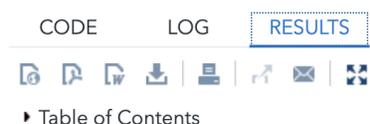
LOG

It is crucial to develop a routine habit of checking the Log tab after each code execution as it is a tremendous tool for helping users during troubleshooting. After executing the program code, the SAS Log tab provides useful information about Errors, Warnings, and Notes in corresponding red, yellow, and blue colors.



RESULTS

By clicking the Results tab, you can view any output results from output-producing procedures. SAS software automatically produces HyperText Markup Language (HTML) results as the “default” output format, along with any graphical, tabular, and statistical information when it be requested, as shown, below.



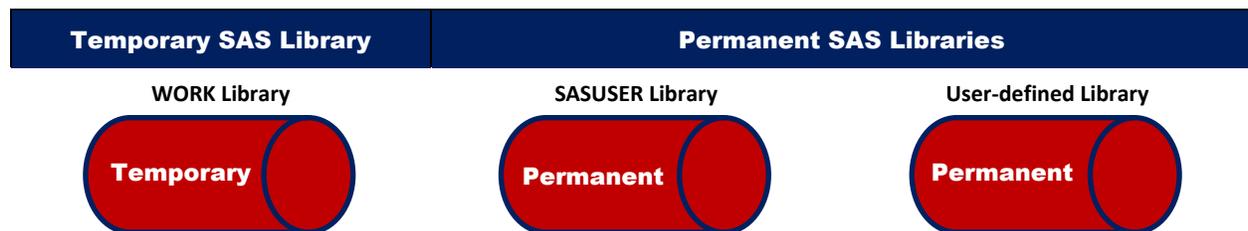
The MEANS Procedure

Analysis Variable : Smoking							
Sex	Smoking Status	N Obs	N	Mean	Std Dev	Minimum	Maximum
Female	Heavy (16-25)	339	339	20.3834808	1.3325105	20.0000000	25.0000000
	Light (1-5)	422	422	4.1279621	1.6535358	1.0000000	5.0000000
	Moderate (6-15)	340	340	12.6764706	2.4974393	10.0000000	15.0000000
	Non-smoker	1682	1682	0	0	0	0
	Very Heavy (> 25)	73	73	33.9726027	4.7110172	30.0000000	45.0000000
Male	Heavy (16-25)	707	707	20.7001414	1.7363103	20.0000000	25.0000000
	Light (1-5)	157	157	4.4649682	1.3659254	1.0000000	5.0000000
	Moderate (6-15)	236	236	12.9449153	2.4653202	10.0000000	15.0000000
	Non-smoker	819	819	0	0	0	0
	Very Heavy (> 25)	398	398	36.7336683	7.7107287	30.0000000	60.0000000

Temporary versus Permanent SAS Data Sets

In the SAS world, the location of your data is everything. This concept is essential for SAS users to understand when using SAS OnDemand for Academics (ODA), or any other SAS product. But what does it mean? Data can be stored on a variety of fixed or removable storage devices including CDs, DVDs, Blu-ray, USB flash drives, tape, external hard drives, NAS storage, and in the cloud. The data access demonstrations presented in this paper use data that is stored in the cloud.

Another important concept that users should become familiar with is which SAS library a data set is stored in. The library a SAS data set is stored in determines if the data set is temporary or permanent. If this sounds just a bit confusing, then the good news is that, in time and with practice, your comfort level working with temporary and permanent data sets will become second. The SAS WORK library is classified as temporary, and all temporary SAS data sets are automatically removed (or deleted) at the end of a SAS session. A SAS data set that is stored in either the SASUSER library or in a user-defined folder in SAS Studio is classified as permanent and, as a result, is accessible even after the end of a SAS session, from one session to another, or until the SAS data set is removed (or deleted).

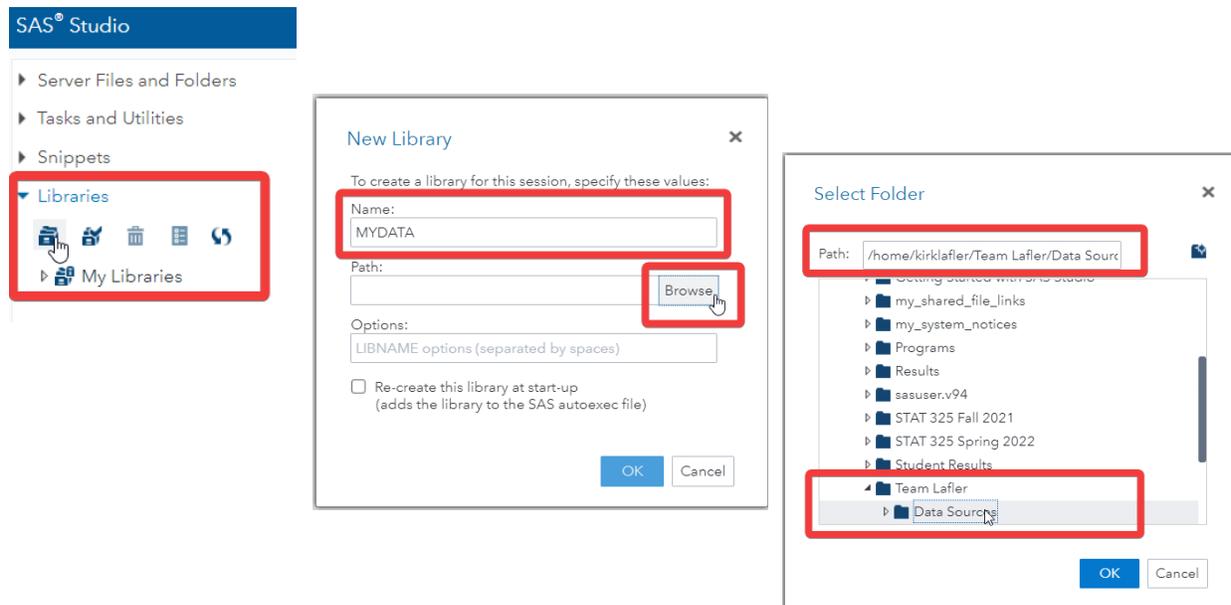


SAS Studio's Point-and-Click Navigation

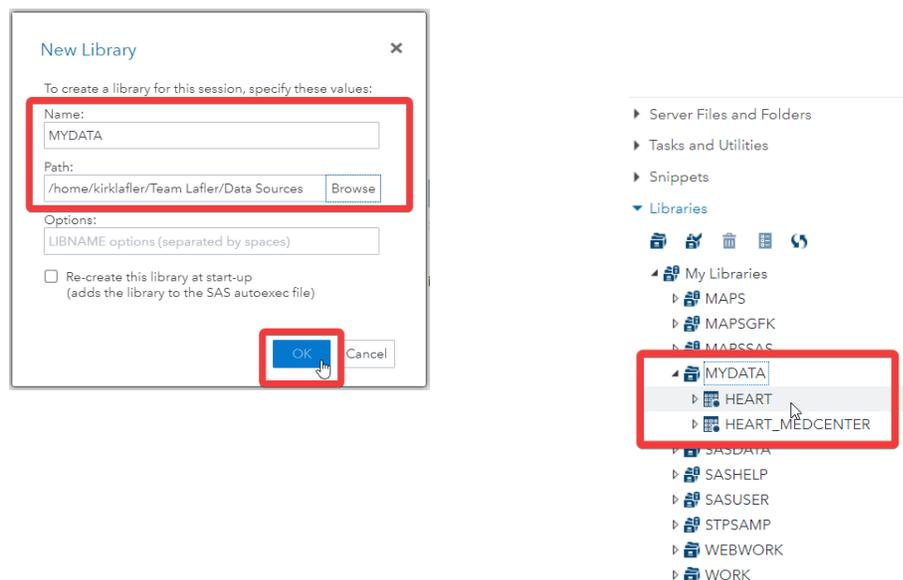
SAS Studio's point-and-click navigation provides users with a powerful, flexible, and easy to use approach to auto-generating SAS code for all types of SAS processing. The objective of this paper is to demonstrate the many capabilities that SAS OnDemand for Academics (ODA) and SAS Studio offers users including creating new SAS libraries; establishing library references (LIBREFs); uploading SAS data sets, tab-delimited, CSV, and Excel data files in the cloud; importing tab-delimited, CSV, and Excel data files to SAS data sets using tasks and utilities; and producing results using the Navigation pane.

Assigning a New SAS Library

Using the Navigation pane's point-and-click features, select **Libraries** → **New Library icon** → **Import Data** to Using the Navigation pane's point-and-click features, users can assign a new SAS library, a libref, and the path to where the data is in the cloud, as shown, below.

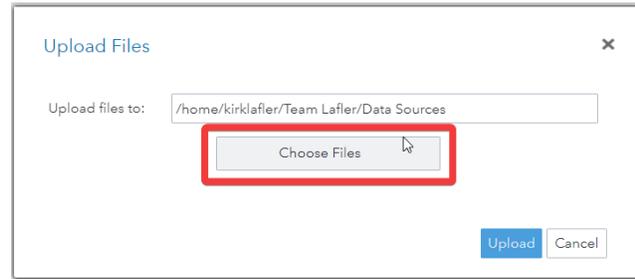
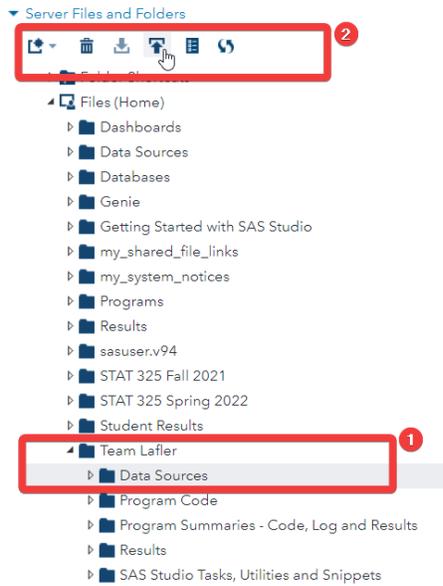


User-assigned library references (LIBREFs) along with their specific paths were specified using the **New Library** window. Specifically, the LIBREF, **MYDATA**, along with its path to identify where the Heart and Heart_MedCenter data sets are stored in the cloud were assigned, as shown, below.

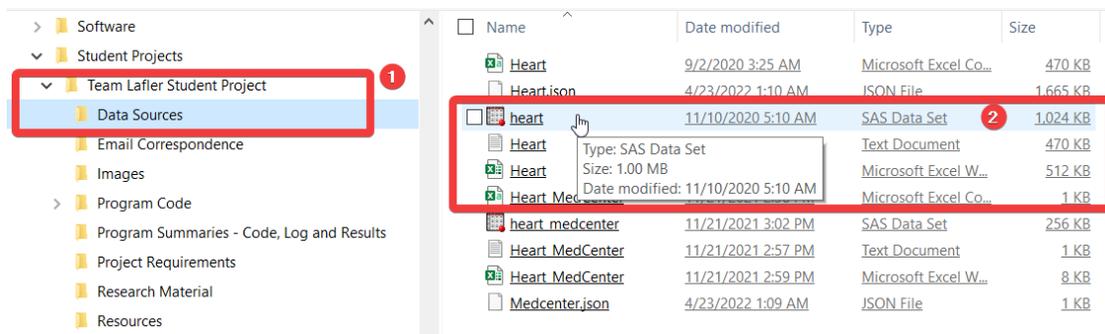


Uploading SAS Data Sets and Other Data Files to the Cloud

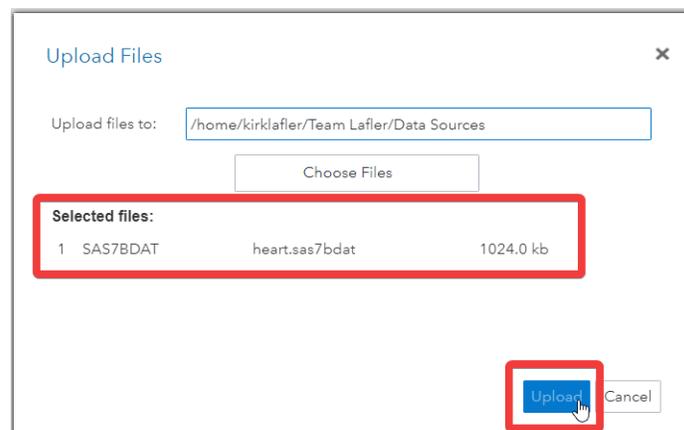
To upload SAS data sets and data files to SAS Studio in the cloud, use the following point-and-click steps, as shown, below. In step #1, click the desired folder / sub-folder where you want a SAS data set or data file uploaded to. Then, in step #2, click the Upload control tool to display the **Upload Files** window. Then, click the **Choose Files** button, as shown, below.



After clicking the **Choose Files** button, navigate to where your data is stored (step #1), and then select the SAS data set you want to upload (step #2), as shown, below.



After selecting the SAS data set from the list of data files you want uploaded to the cloud, the **Upload Files** window will then display the name of the selected data set. Finally, clicking the **Upload** button launches the upload process, as shown, below.



Data Access Using SAS Studio

SAS Studio's point-and-click approach uses the Navigation pane as a relatively easy and flexible way to access SAS data sets and data files, automatically generate program code, and run (or execute) program code using SAS ODA software. We'll guide you through the steps to access permanent and temporary SAS data sets and data files residing in the cloud; create new SAS data sets; produce results including reports, tables, statistics, and charts using SAS Studio's point-and-click approach. We'll explore the data access steps for four different types of data files:

- SAS (SAS7BDAT) Data Sets
- Tab-delimited Text Data Files
- Comma-separated Values (CSV) Data Files
- Excel (XLSX) Data Files

Data Access and SAS (SAS7BDAT) Data Sets

SAS (SAS7BDAT) data sets can be accessed using SAS OnDemand for Academics (ODA) and SAS Studio, a proprietary data format, developed by SAS Institute. Since a SAS data set is already structured in the desired format, it only needs to be uploaded to the desired library in the cloud (see point-and-click instructions in previous section), and then it can be accessed directly using the pre-defined point-and-click tasks specified in this paper. SAS data sets contain data values that are created, organized, and stored as a table of rows and columns in a SAS library (e.g., WORK, SASUSER, and User-assigned). In our next example, we'll use SAS Studio's point-and-click Navigation pane and SAS ODA to auto-generate the code to access the Heart SAS data set and produce summary statistics results with PROC MEANS, as shown, below.

The screenshot displays the SAS Studio interface. On the left is the 'Server Files and Folders' pane, and on the right is the 'Summary Statistics' task configuration pane. Red circles with numbers 1 through 11 highlight key steps in the process:

1. Click on 'Tasks and Utilities' in the left navigation pane.
2. Click on 'Tasks' in the left navigation pane.
3. Click on 'Statistics' in the left navigation pane.
4. Click on 'Summary Statistics' in the left navigation pane.
5. Enter the data set name 'MYDATA.HEART' in the 'DATA' field.
6. Select the variable 'Smoking' in the '*Analysis variables:' list.
7. Select the variable 'Smoking_Status' in the 'Classification variables:' list.
8. Select the variable 'Status' in the 'Group analysis by:' list.
9. Click the 'Run' button (a red square with a white play icon).
10. Click the 'Code/Results' tab.
11. Click the 'Split' button.

The auto-generated code produced **PROC SORT** and **PROC MEANS** statements from the selections made using the SAS Studio Navigation pane, as shown, below. To run the auto-generated code, click the “**running**” icon, (step #11) on the previous page.

```

CODE    LOG    RESULTS
[Icons] Line # [Icons] Edit
1  /*
2  *
3  * Task code generated by SAS Studio 3.8
4  *
5  * Generated on '8/15/22, 7:51 AM'
6  * Generated by 'kirklafler'
7  * Generated on server 'ODAWS01-USW2.ODA.SAS.COM'
8  * Generated on SAS platform 'Linux LIN X64 3.10.0-1062.9.1.e17.x86_64'
9  * Generated on SAS version '9.04.01M6P11072018'
10 * Generated on browser 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit
11 * Generated on web client 'https://odamid-usw2.oda.sas.com/SASStudio/main?loc
12 *
13 */
14
15 ods noproctitle;
16 ods graphics / imagemap=on;
17
18 proc sort data=MYDATA.HEART out=WORK.TempSorted2236;
19     by Status;
20 run;
21
22 proc means data=WORK.TempSorted2236 chartype mean std min max n vardef=df;
23     var Smoking;
24     class Smoking_Status;
25     by Status;
26 run;
27
28 proc datasets library=WORK noprint;
29     delete TempSorted2236;
30 run;
    
```

The summary statistics results are produced in the **Results** tab, as shown, below.

CODE LOG RESULTS

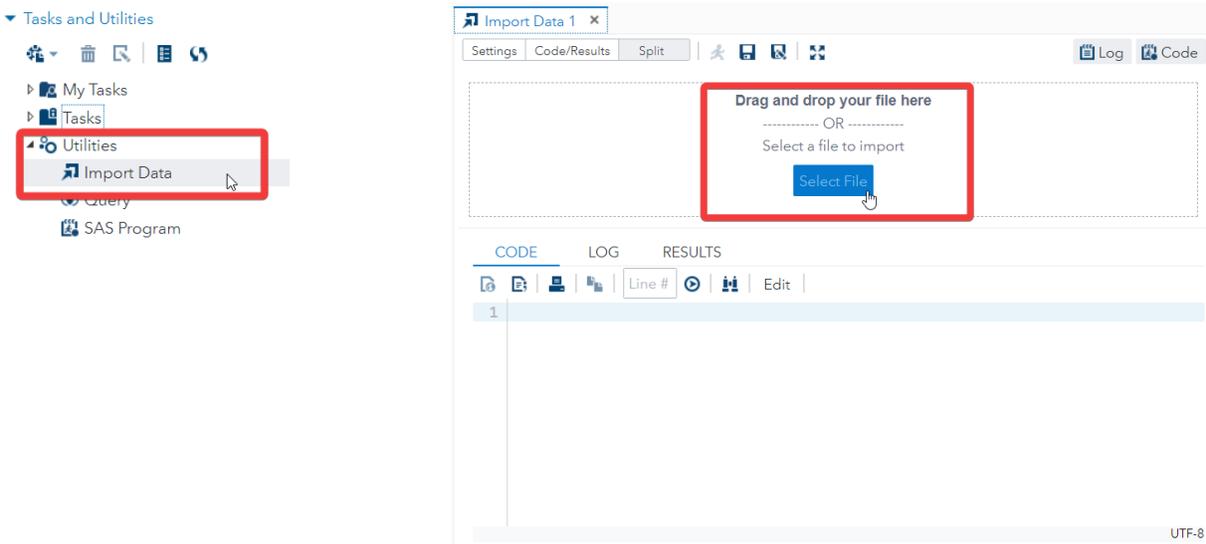
Table of Contents

Status=Alive						
Analysis Variable : Smoking						
Smoking Status	N Obs	Mean	Std Dev	Minimum	Maximum	N
Heavy (16-25)	603	20.5389718	1.5518899	20.0000000	25.0000000	603
Light (1-5)	392	4.1734694	1.6216282	1.0000000	5.0000000	392
Moderate (6-15)	363	12.8236915	2.4823778	10.0000000	15.0000000	363
Non-smoker	1610	0	0	0	0	1610
Very Heavy (> 25)	42	34.1666667	4.6743549	30.0000000	40.0000000	42

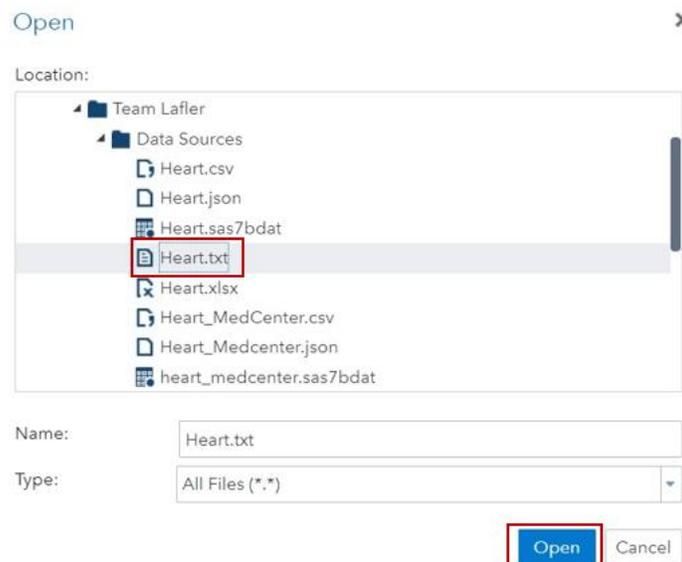
Status=Dead						
Analysis Variable : Smoking						
Smoking Status	N Obs	Mean	Std Dev	Minimum	Maximum	N
Heavy (16-25)	443	20.6772009	1.7129000	20.0000000	25.0000000	443
Light (1-5)	187	4.3155080	1.5105098	1.0000000	5.0000000	187
Moderate (6-15)	213	12.7230047	2.4958997	10.0000000	15.0000000	213
Non-smoker	891	0	0	0	0	891
Very Heavy (> 25)	31	33.7096774	4.8248907	30.0000000	45.0000000	31

Data Access and Tab-delimited (TSV) Text Data Files

Using the Navigation pane's point-and-click features, select **Tasks and Utilities** → **Utilities** → **Import Data** to auto-generate the PROC IMPORT code to access tab-delimited (TSV) text data files, as shown, below.

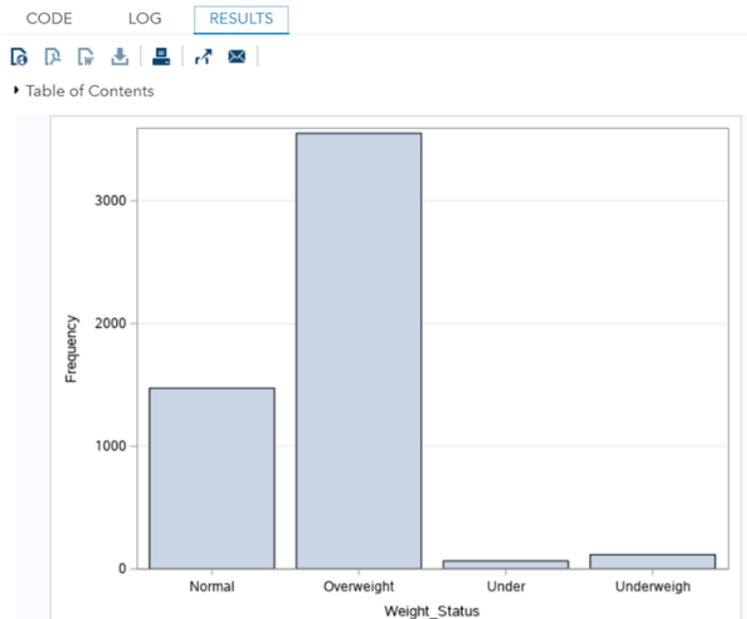


After clicking “**Tasks and Utilities**” and then “**Utilities**” from the Navigation pane, you can select “**Import Data**” to open a new window where you can select a file to import. In this demonstration we’ll select the **Heart.txt** file, as shown, below.



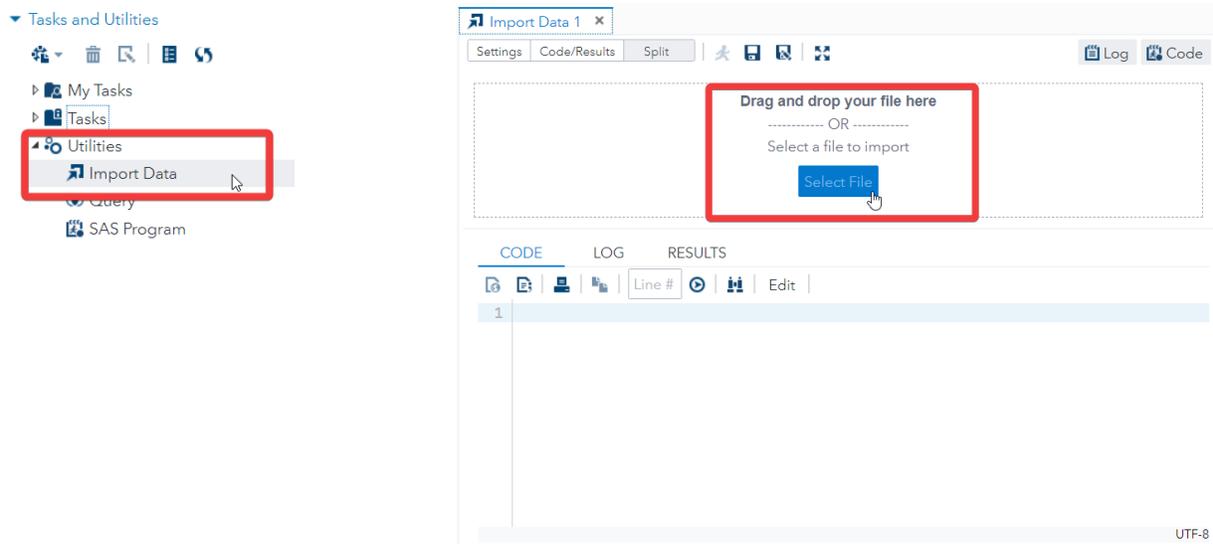
After opening the file, the code is automatically generated. By default, the SAS data set temporarily exists in the WORK library as a name of IMPORT. You can customize the location and name of the SAS data set by clicking **Edit** allowing you to change the code, as shown, below. Here we rename the SAS file that we’re about to create, **IMPORT_FROM_TXT**, under the **WORK** (or temporary) library. Also, for the tab-delimited text data files, you need to specify “**TAB**” as the DBMS identifier. After making the changes, you can run the code, as shown, below.

The vertical bar chart results are produced in the **Results** tab, as shown, below.

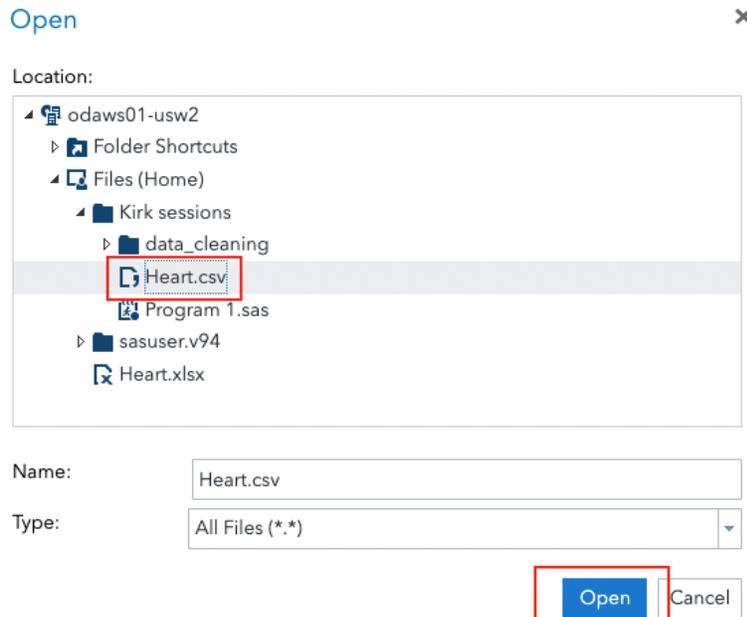


Data Access and Comma-separated Values (CSV) Data Files

Using the Navigation pane’s point-and-click features, select **Tasks and Utilities** → **Utilities** → **Import Data** to auto-generate the PROC IMPORT code to access tab-delimited text data files, as shown, below.



After clicking “**Tasks and Utilities**” and then “**Utilities**” from the Navigation pane, you can select “**Import Data**” to open a new window where you can select a file to import. In this demonstration we’ll select the **Heart.csv** file and click the **Open** button, as shown, below.



After opening the file, the code is automatically generated. By default, the SAS data set temporarily exists in the WORK library as a name of IMPORT. You can customize the location and name of the SAS data set by clicking **Edit** allowing you to change the code, as shown, below.

```
CODE LOG RESULTS
1 /* Generated Code (IMPORT) */
2 /* Source File: Heart.csv */
3 /* Source Path: /home/kirklafler/Team Lafler/Data Sources */
4 /* Code generated on: 8/16/22, 3:13 AM */
5
6 %web_drop_table(WORK.IMPORT);
7
8
9 FILENAME REFFILE '/home/kirklafler/Team Lafler/Data Sources/Heart.csv';
10
11 PROC IMPORT DATAFILE=REFFILE
12   DBMS=CSV
13   OUT=WORK.IMPORT;
14   GETNAMES=YES;
15 RUN;
16
17 PROC CONTENTS DATA=WORK.IMPORT; RUN;
18
19
20 %web_open_table(WORK.IMPORT);
```

We then rename the SAS data set in the Code tab from WORK.IMPORT to, **IMPORT_FROM_CSV**, in the **WORK** (or temporary) SAS library. Also, for the comma-separated values (CSV) data file, you will want to specify “**CSV**” as the DBMS identifier and rename the data set in the PROC CONTENTS to produce detailed metadata information. After making the changes, you can run the code, as shown, below.

```

CODE    LOG    RESULTS  OUTPUT DATA
[Log Icon] [Refresh] [Save] [Print] [Copy] [Paste] [Undo] [Redo] [Home] [End] [Line #] [Zoom In] [Zoom Out] [Fullscreen]
4  /* Generated Code (IMPORT) */
5  /* Source File: Heart.csv */
6  /* Source Path: /home/kirklafler/Team Lafler/Data Sources */
7  /* Code generated on: 8/16/22, 5:00 AM */
8
9  %web_drop_table(WORK.IMPORT);
10
11 FILENAME REFFILE '/home/kirklafler/Team Lafler/Data Sources/Heart.csv' ;
12 PROC IMPORT DATAFILE=REFFILE
13             DBMS=CSV
14             OUT=WORK.IMPORT_Heart_CSV ;
15 GETNAMES=YES ;
16 RUN ;
17 PROC CONTENTS DATA=WORK.IMPORT_Heart_CSV ;
18 RUN ;
19
20
21 %web_open_table(WORK.IMPORT);
22

```

By clicking the **Log** tab, you'll be able to view essential information about your running program code including the number of observations and variables contained in the new SAS data set, and any Notes, Warnings, and Errors that were produced from running the code.

```

NOTE: WORK.IMPORT_HEART_CSV data set was successfully created.
NOTE: The data set WORK.IMPORT_HEART_CSV has 5209 observations and 18 variables.
NOTE: PROCEDURE IMPORT used (Total process time):
      real time           0.13 seconds
      user cpu time       0.09 seconds
      system cpu time     0.01 seconds
      memory              10258.25k
      OS Memory          42920.00k
      Timestamp           08/16/2022 12:55:04 PM
      Step Count          31  Switch Count  8
      Page Faults         0
      Page Reclaims      5572
      Page Swaps          0
      Voluntary Context Switches  70
      Involuntary Context Switches  3
      Block Input Operations  0
      Block Output Operations 1864

```

By clicking the **Results** tab, you'll be able to view detailed metadata information describing essential information about the SAS data set, WORK.IMPORT_HEART_CSV, including the number of observations and variables contained in the new SAS data set, list of columns (or variables), and much more valuable information, as shown, below.

CODE LOG RESULTS OUTPUT DATA

[Refresh] [Save] [Print] [Copy] [Paste] [Undo] [Redo] [Home] [End] [Line #] [Zoom In] [Zoom Out] [Fullscreen]

► Table of Contents

The CONTENTS Procedure			
Data Set Name	WORK.IMPORT_HEART_CSV	Observations	5209
Member Type	DATA	Variables	18
Engine	V9	Indexes	0
Created	08/16/2022 05:55:04	Observation Length	168
Last Modified	08/16/2022 05:55:04	Deleted Observations	0
Protection		Compressed	NO
Data Set Type		Sorted	NO
Label			
Data Representation	SOLARIS_X86_64, LINUX_X86_64, ALPHA_TRU64, LINUX_IA64		
Encoding	utf-8 Unicode (UTF-8)		

Engine/Host Dependent Information	
Data Set Page Size	131072
Number of Data Set Pages	7
First Data Page	1
Max Obs per Page	779
Obs in First Data Page	754
Number of Data Set Repairs	0
Filename	/saswork/SAS_work9F590001524F_odaws03-usw2.oda.sas.com/SAS_work1A8B0001524F_odaws03-usw2.oda.sas.com/import_heart_csv.sas7bdat
Release Created	9.0401M6
Host Created	Linux
Inode Number	1610618467
Access Permission	rw-r--r--
Owner Name	kirklafler
File Size	1MB
File Size (bytes)	1048576

Alphabetic List of Variables and Attributes					
#	Variable	Type	Len	Format	Informat
13	AgeAtDeath	Num	8	BEST12.	BEST32.
6	AgeAtStart	Num	8	BEST12.	BEST32.
4	AgeCHDdiag	Num	8	BEST12.	BEST32.
16	BP_Status	Char	7	\$7.	\$7.
15	Chol_Status	Char	10	\$10.	\$10.
14	Cholesterol	Num	8	BEST12.	BEST32.
3	DeathCause	Char	25	\$25.	\$25.
9	Diastolic	Num	8	BEST12.	BEST32.
7	Height	Num	8	BEST12.	BEST32.
11	MRW	Num	8	BEST12.	BEST32.
1	MedCtrlID	Char	7	\$7.	\$7.
5	Sex	Char	6	\$6.	\$6.
12	Smoking	Num	8	BEST12.	BEST32.
18	Smoking_Status	Char	17	\$17.	\$17.
2	Status	Char	5	\$5.	\$5.
10	Systolic	Num	8	BEST12.	BEST32.
8	Weight	Num	8	BEST12.	BEST32.
17	Weight_Status	Char	10	\$10.	\$10.

By clicking the **Output Data** tab, you'll be able to view detailed metadata information including the SAS data set's column names and column attributes (e.g., column type, column length, column labels, informat, and format information), as shown, below.

CODE LOG RESULTS **OUTPUT DATA**

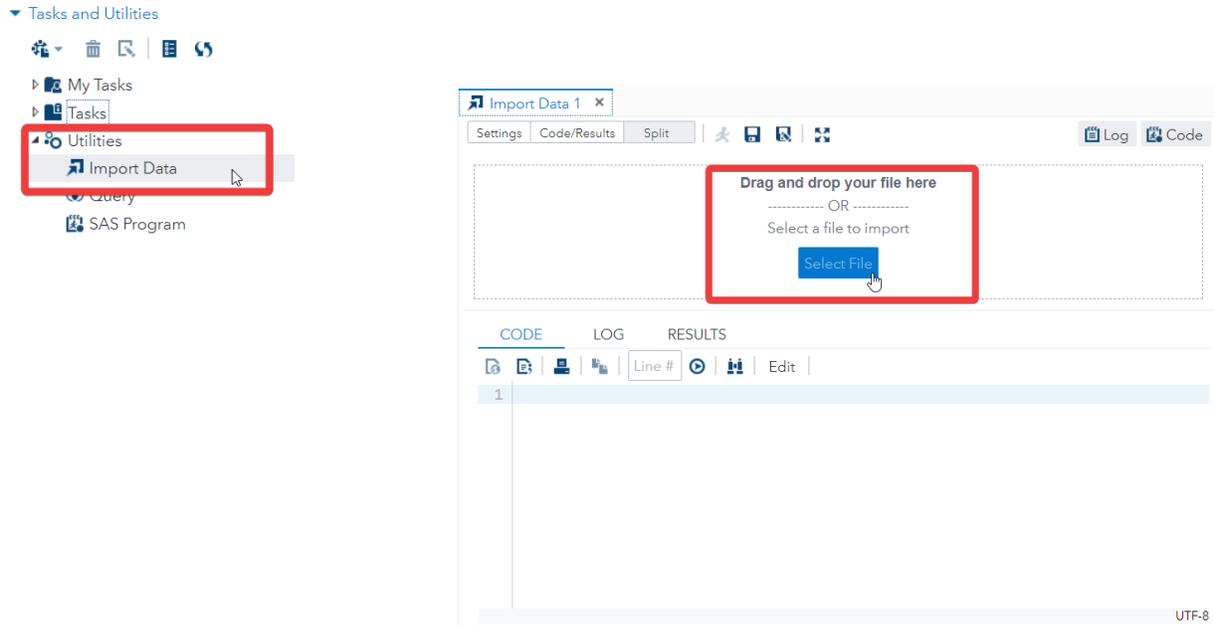
Table: WORK.IMPORT_HEART_CSV | View: Column names | Filter: (none)

Columns: Total rows: 5209 Total columns: 18

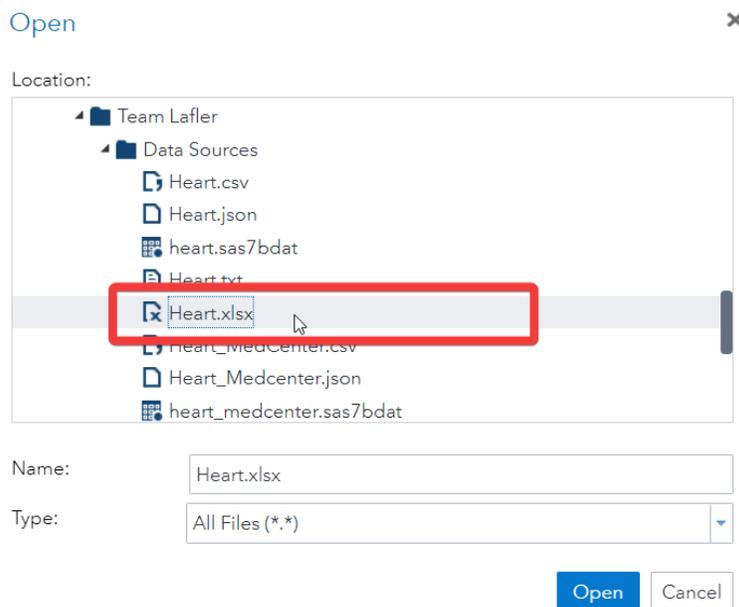
	MedCtrlID	Status	DeathCause	AgeCHDdiag	Sex	AgeAtStart	Height
1	CA94105	Dead	Other	.	Female	29	62.5
2	CA94105	Dead	Cancer	.	Female	41	59.75
3	CA94105	Alive		.	Female	57	62.25
4	CA92307	Alive		.	Female	39	65.75
5	CA90025	Alive		.	Male	42	66
6	CA92307	Alive		.	Female	58	61.75
7	CA94105	Alive		.	Female	36	64.75
8	CA90025	Dead	Other	.	Male	53	65.5
9	CA92307	Alive		.	Male	35	71
10	CA90025	Dead	Cerebral Vascular Disease	.	Male	52	62.5
11	NV89109	RIP		.	Male	39	66.25

Data Access and Excel (XLSX) Data Files

An Excel (XLSX) data file is a proprietary data format, developed by Microsoft, and is used to format, organize, and compute data in a spreadsheet. Using the Navigation pane's point-and-click features, select **Tasks and Utilities** → **Utilities** → **Import Data** to auto-generate the PROC IMPORT code to access the Excel (XLSX) data file, as shown, below.



After clicking “**Tasks and Utilities**” and then “**Utilities**” from the Navigation pane, you can select “**Import Data**” to open a new window where you can select a file to import. In this demonstration we’ll select the **Heart.xlsx** file and click the **Open** button, as shown, below.



After opening the file, the code is automatically generated. By default, the SAS data set temporarily exists in the WORK library as a name of IMPORT. You can customize the location and name of the SAS data set by clicking **Edit** allowing you to change the code, as shown, below.

```

CODE LOG RESULTS
1 /* Generated Code (IMPORT) */
2 /* Source File: Heart.xlsx */
3 /* Source Path: /home/kirklafler/Team Lafler/Data Sources */
4 /* Code generated on: 8/16/22, 4:43 AM */
5
6 %web_drop_table(WORK.IMPORT);
7
8
9 FILENAME REFFILE '/home/kirklafler/Team Lafler/Data Sources/Heart.xlsx';
10
11 PROC IMPORT DATAFILE=REFFILE
12     DBMS=XLSX
13     OUT=WORK.IMPORT;
14     GETNAMES=YES;
15 RUN;
16
17 PROC CONTENTS DATA=WORK.IMPORT; RUN;
18
19
20 %web_open_table(WORK.IMPORT);
    
```

We then rename the SAS data set in the Code tab from WORK.IMPORT to **WORK.IMPORT_FROM_XLSX**, in the **WORK** (or temporary) SAS library. Also, for the Excel (XLSX) data file, you will want to specify **"XLSX"** as the DBMS identifier and rename the data set in the PROC CONTENTS to produce metadata information. You can then run the code, as shown, below.

```

CODE LOG RESULTS
1 /* Generated Code (IMPORT) */
2 /* Source File: Heart.xlsx */
3 /* Source Path: /home/kirklafler/Team Lafler/Data Sources */
4 /* Code generated on: 8/16/22, 4:43 AM */
5
6 %web_drop_table(WORK.IMPORT);
7
8
9 FILENAME REFFILE '/home/kirklafler/Team Lafler/Data Sources/Heart.xlsx' ;
10
11 PROC IMPORT DATAFILE=REFFILE
12     DBMS=XLSX
13     OUT=WORK.IMPORT_Heart_XLSX ;
14     GETNAMES=YES ;
15 RUN ;
16
17 PROC CONTENTS DATA=WORK.IMPORT_Heart_XLSX ;
18 RUN ;
19
20
21 %web_open_table(WORK.IMPORT);
22
23
    
```

By clicking the **Log** tab, you'll be able to view essential information about the program code including the number of observations and variables contained in the new SAS data set, and any Notes, Warnings, and Errors that were produced.

```

NOTE: The import data set has 5209 observations and 18 variables.
NOTE: WORK.IMPORT_HEART_XLSX data set was successfully created.
NOTE: PROCEDURE IMPORT used (Total process time):
      real time           0.92 seconds
      user cpu time       0.92 seconds
      system cpu time     0.00 seconds
      memory              3503.96k
      OS Memory           37040.00k
      Timestamp           08/16/2022 01:44:18 PM
      Step Count          58
      Step CPU Time       0.00 seconds
      Step Memory         3503.96k
      Step OS Memory      37040.00k
      Step Timestamp      08/16/2022 01:44:18 PM
      Step Count          58
      Step CPU Time       0.00 seconds
      Step Memory         3503.96k
      Step OS Memory      37040.00k
      Step Timestamp      08/16/2022 01:44:18 PM
      Page Faults         0
      Page Reclaims       747
      Page Swaps           0
      Voluntary Context Switches 16
      Involuntary Context Switches 1
      Block Input Operations 1072
      Block Output Operations 1544
    
```

By clicking the **Results tab**, you'll be able to view detailed metadata information describing essential information about the SAS data set, WORK.IMPORT_HEART_XLSX, including the number of observations and variables contained in the new SAS data set, list of columns (or variables), and much more valuable information, as shown, below.

The CONTENTS Procedure

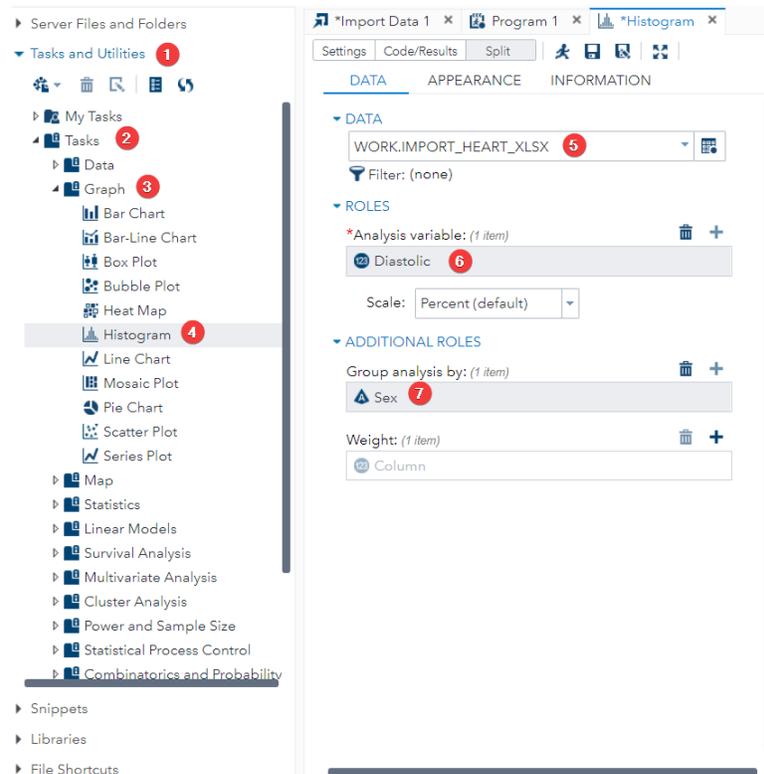
Data Set Name	WORK.IMPORT_HEART_XLSX	Observations	5209
Member Type	DATA	Variables	18
Engine	V9	Indexes	0
Created	08/14/2022 05:30:56	Observation Length	136
Last Modified	08/14/2022 05:30:56	Deleted Observations	0
Protection		Compressed	NO
Data Set Type		Sorted	NO
Label			
Data Representation	SOLARIS_X86_64, LINUX_X86_64, ALPHA_TRU64, LINUX_IA64		
Encoding	utf-8 Unicode (UTF-8)		

Engine/Host Dependent Information	
Data Set Page Size	131072
Number of Data Set Pages	6
First Data Page	1
Max Obs per Page	962
Obs in First Data Page	930
Number of Data Set Repairs	0
Filename	/saswork/SAS_workD13800004D28_odaws01-usw2.oda.sas.com/SAS_work308E00004D28_odaws01-usw2.oda.sas.com/import_heart_xlsx.sas7bdat
Release Created	9.0401M6
Host Created	Linux
Inode Number	1610687682
Access Permission	rw-r--r--
Owner Name	kirklafier
File Size	896KB
File Size (bytes)	917504

Alphabetic List of Variables and Attributes						
#	Variable	Type	Len	Format	Informat	Label
13	AgeAtDeath	Char	2	\$2.	\$2.	AgeAtDeath
6	AgeAtStart	Num	8	BEST.		AgeAtStart
4	AgeCHDdiag	Char	2	\$2.	\$2.	AgeCHDdiag
16	BP_Status	Char	7	\$7.	\$7.	BP_Status
15	Chol_Status	Char	10	\$10.	\$10.	Chol_Status
14	Cholesterol	Char	3	\$3.	\$3.	Cholesterol
3	DeathCause	Char	25	\$25.	\$25.	DeathCause
9	Diastolic	Num	8	BEST.		Diastolic
7	Height	Char	5	\$5.	\$5.	Height
11	MRW	Char	3	\$3.	\$3.	MRW
1	MedCtrlID	Char	7	\$7.	\$7.	MedCtrlID
5	Sex	Char	6	\$6.	\$6.	Sex
12	Smoking	Char	2	\$2.	\$2.	Smoking
18	Smoking_Status	Char	17	\$17.	\$17.	Smoking_Status
2	Status	Char	5	\$5.	\$5.	Status
10	Systolic	Num	8	BEST.		Systolic
8	Weight	Char	3	\$3.	\$3.	Weight
17	Weight_Status	Char	11	\$11.	\$11.	Weight_Status

Now you can do further analysis with the SAS data set by using the point-of-click features as well. To illustrate, we will create a **Histogram** to visualize Diastolic as the analysis variable grouped by Sex distributed across the observations with several clicks.

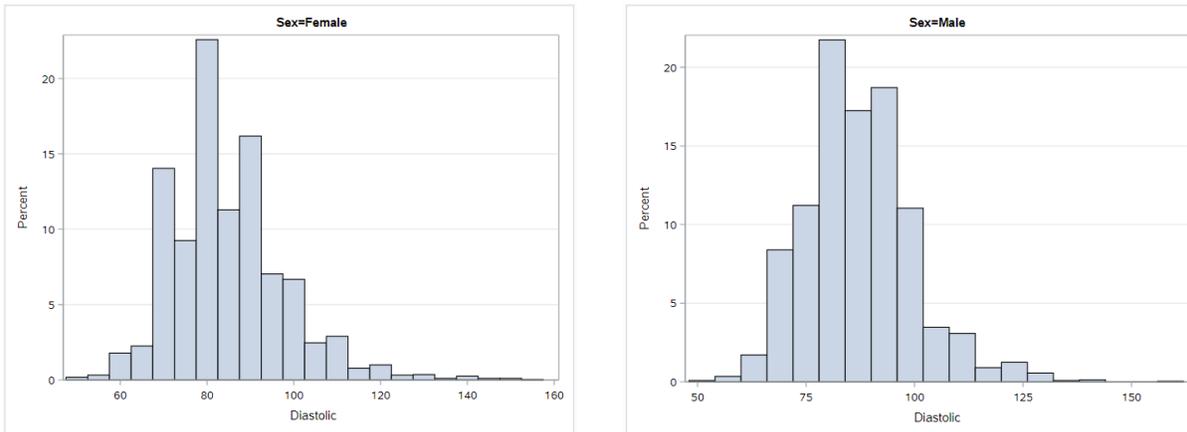
We selected "**Histogram**" under the **Tasks and Utilities** → **Tasks** → **Histogram**. After choosing Histogram, a new window opens, where you can click the small "**table**" button to select a table. Here we choose the newly imported **IMPORT_FROM_XLSX** table, and run the auto-generated code, as shown, below.



```

CODE    LOG    RESULTS
[Icons] Line # [Icons] Edit |
1 /*
2 *
3 * Task code generated by SAS Studio 3.8
4 *
5 * Generated on '8/14/22, 5:54 AM'
6 * Generated by 'kirklafler'
7 * Generated on server 'ODAWS02-USW2.ODA.SAS.COM'
8 * Generated on SAS platform 'Linux LIN X64 3.10.0-1062.9.1.el7.x86_64'
9 * Generated on SAS version '9.04.01M6P11072018'
10 * Generated on browser 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML
11 * Generated on web client 'https://odamid-usw2.oda.sas.com/SASstudio/main?locale=en_US&
12 *
13 */
14
15 ods graphics / reset width=6.4in height=4.8in imagemap;
16
17 proc sort data=WORK.IMPORT_HEART_XLSX out=_HistogramTaskData;
18   by Sex;
19 run;
20
21 proc sgplot data=_HistogramTaskData;
22   by Sex;
23   histogram Diastolic /;
24   yaxis grid;
25 run;
26
27 ods graphics / reset;
28
29 proc datasets library=WORK noprint;
30   delete _HistogramTaskData;
31 run;
    
```

The Histogram results are produced in the **Results** tab, as shown, below.



Data Access and JSON Data Files

JavaScript Object Notation (JSON) is very popular around the world. Compared to XML, JSON file sizes are typically much smaller, easier to read, and considerably faster to load data than XML. In fact, JSON has been quickly replacing XML as the “go-to” data format. Since Team Lafler has worked with JSON, we’ll provide a brief introduction of how to create a JSON data file using PROC JSON and create a SAS data set by reading and processing a JSON data file with a SAS DATA step in SAS ODA and SAS Studio. Although there may be a task or utility available in SAS Studio to auto-generate either DATA step or PROC JSON code for the purpose of creating a SAS data set and/or a JSON data file, we weren’t able to find it at the time of developing this paper. Instead, we’ll show you what a JSON data file looks like; demonstrate how to read, process, and create a SAS data set from a JSON data file; and how to create a JSON data file from a SAS data set.

Creating a JSON Data File Using SAS ODA and SAS Studio

In the following example, we’ll create a JSON data file using PROC JSON in SAS ODA. The SAS data set specified as input is, Heart_MedCenter, as shown, below.

Code:

```
libname mydata '/home/kirklafler/Team Lafler/Data Sources' ;

/* Produce metadata using PROC CONTENTS */
proc contents data = mydata.heart_medcenter ;
run ;

/* Create a JSON Data File from the SAS Heart_MedCenter Data Set */
proc json out = "/home/kirklafler/Team Lafler/Results/Heart_MedCenter.json" ;
  export mydata.heart_medcenter ;
run ;
```

PROC CONTENTS Results:

The CONTENTS Procedure			
Data Set Name	MYDATA.HEART_MEDCENTER	Observations	5
Member Type	DATA	Variables	5
Engine	V9	Indexes	0
Created	11/21/2021 17:02:03	Observation Length	74
Last Modified	11/21/2021 17:02:03	Deleted Observations	0
Protection		Compressed	NO
Data Set Type		Sorted	NO
Label			
Data Representation	SOLARIS_X86_64, LINUX_X86_64, ALPHA_TRU64, LINUX_IA64		
Encoding	utf-8 Unicode (UTF-8)		

PROC CONTENTS Results (continued):

Engine/Host Dependent Information	
Data Set Page Size	131072
Number of Data Set Pages	1
First Data Page	1
Max Obs per Page	1767
Obs in First Data Page	5
Number of Data Set Repairs	0
Filename	/home/kirklafler/Team Laffer/Data Sources/heart_medcenter.sas7bdat
Release Created	9.0401M6
Host Created	Linux
Inode Number	4314274826
Access Permission	rw-r--r--
Owner Name	kirklafler
File Size	256KB
File Size (bytes)	262144

Alphabetic List of Variables and Attributes			
#	Variable	Type	Len
3	City	Char	20
1	MedCtrID	Char	7
2	MedicalCenter	Char	40
4	State	Char	2
5	Zip	Char	5

PROC JSON Results:

```
{
  "SASJSONExport": "1.0 PRETTY",
  "SASTableData+MEDCENTER": [
    {
      "MedCtrID": "CA92101",
      "MedicalCenter": "San Diego Medical Center",
      "City": "San Diego",
      "State": "CA",
      "Zip": "92101"
    },
    {
      "MedCtrID": "CA92037",
      "MedicalCenter": "La Jolla Heart Institute",
      "City": "La Jolla",
      "State": "CA",
      "Zip": "92037"
    },
    {
      "MedCtrID": "CA90025",
      "MedicalCenter": "Los Angeles Medical Center",
      "City": "Los Angeles",
      "State": "CA",
      "Zip": "90025"
    },
    {
      "MedCtrID": "CA94105",
      "MedicalCenter": "San Francisco Medical Center",
      "City": "San Francisco",
      "State": "CA",
      "Zip": "94105"
    },
    {
      "MedCtrID": "NV89109",
      "MedicalCenter": "Las Vegas Health Center",
      "City": "Las Vegas",
      "State": "NV",
      "Zip": "89109"
    }
  ]
}
```

Creating a SAS Data Set from a JSON Data File Using SAS ODA and SAS Studio

In the next example, we'll create a SAS data set from a JSON data file using a DATA step in SAS ODA. The SAS data set is called, Heart_MedCenter_JSON, as shown, below.

Code:

```
filename myjson "/home/kirklafler/Team Lafler/Data Sources/Heart_Medcenter.json" ;

data WORK.Heart_MedCenter_JSON ;
  infile myjson lrecl=99999999 dlm="{}[[:,]" dsd ;
  input
    @'"MedCtrID":' MedCtrID : $7.
    @'"MedicalCenter":' MedicalCenter : $40.
    @'"City":' City : $20.
    @'"State":' State : $2.
    @'"Zip":' Zip : $5.
  @@
;
run;

proc print data=WORK.Heart_MedCenter_JSON N ;
run ;
```

PROC PRINT Results:

Obs	MedCtrID	MedicalCenter	City	State	Zip
1	CA92101	San Diego Medical Center	San Diego	CA	92101
2	CA92037	La Jolla Heart Institute	La Jolla	CA	92037
3	CA90025	Los Angeles Medical Center	Los Angeles	CA	90025
4	CA94105	San Francisco Medical Center	San Francisco	CA	94105
5	NV89109	Las Vegas Health Center	Las Vegas	NV	89109
N = 5					

Conclusion

With SAS® OnDemand for Academics (ODA) and SAS Studio, students, faculty, and anyone who wants to learn SAS software's many features has access to a full-blown version of SAS software. The primary objective was to demonstrate SAS ODA's cloud-based user-friendly interface, SAS Studio and its point-and-click features using the Navigation pane, to access a variety of data files including SAS (SAS7BDAT) data sets, tab-delimited text (TSV) data files, comma-separated values (CSV) data files, and Excel (XLSX) data files. Another very important data file that is replacing XML data files throughout the industry is, JavaScript Object Notation (JSON). We also demonstrated converting a JSON data file, which is widely used around the world, to a SAS data set using a step-by-step programming approach using SAS ODA and SAS Studio.

Acknowledgments

The author wishes to thank the 2022 SESUG Conference Committee, particularly the Learning SAS II (HOW) Section Chairs, Andrea Lewton and Nat Wooding, for accepting my abstract, paper, and hands-on workshop (HOW); the 2022 SESUG Executive Committee for organizing and supporting a "live" conference event; SAS Institute Inc. for providing SAS users with wonderful software; and SAS users everywhere for being the nicest people anywhere!

Trademark Citations

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration. Other brand and product names are trademarks of their respective companies.

Author Bio

Kirk Paul Lafler is a SAS consultant, application developer, and programmer; lecturer and adjunct professor at San Diego State University; an advisor and adjunct professor at the University of California San Diego Extension; and teaches SAS, SQL, Python and Excel courses, workshops, and webinars to users around the world. Kirk has been a SAS user since 1979 and is the author of several books including, PROC SQL: Beyond the Basics Using SAS, Third Edition (SAS Press. 2019) along with papers and articles on a variety of SAS topics. Kirk has also been selected as an Invited speaker, educator, keynote, and section leader at SAS conferences; and is the recipient of 27 “Best” contributed paper, hands-on workshop (HOW), and poster awards.

Comments and suggestions can be sent to:

Kirk Paul Lafler

SAS® / SQL / Python Consultant, Application Developer, Programmer, Data Analyst, Educator and Author

E-mail: KirkLafler@cs.com

LinkedIn: <https://www.linkedin.com/in/KirkPaulLafler/>

Twitter: @sasNerd