



Celebrity Makeover: A Fresh and Modern Look for SAS® Enterprise Guide

BASUG April 8, 2020

Amy Peters, SAS

Why Update SAS Enterprise Guide?

Things we can do better

Max 2 tabs

Single monitor

Dated look

Forced to work with a project

SAS Enterprise Guide



Do you want to replace the results from the previous run?

Choosing "No" will save the changes to a new task, named "Program (2)".

Yes

No

Annoying modal dialogs

Enterprise Guide is not able to open this data set in update mode; it will remain, or will be opened, in read-only mode.

The open data operation failed. The following error occurred.
[Error] A lock is not available for SASUSER.MYFAFMMSG.DATA.

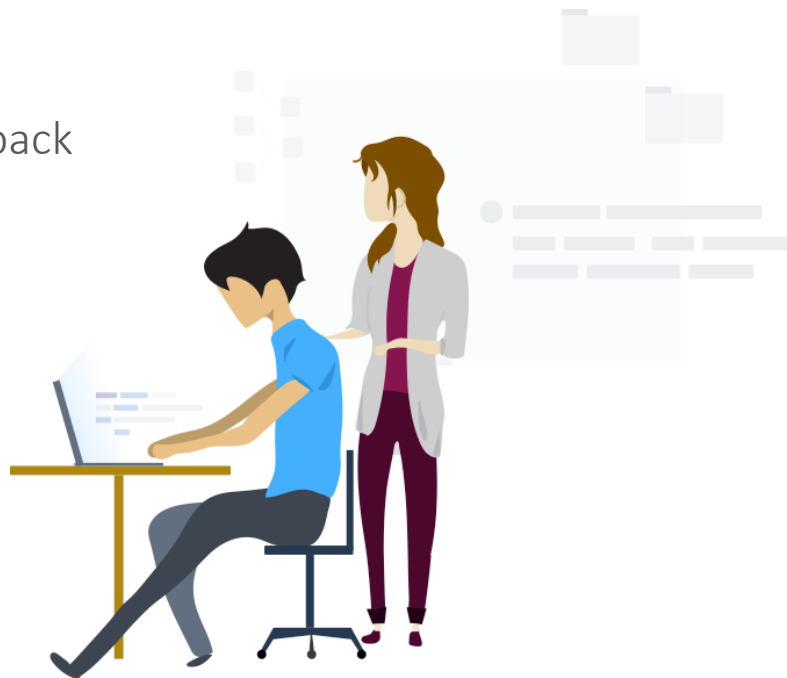
OK

Data locking

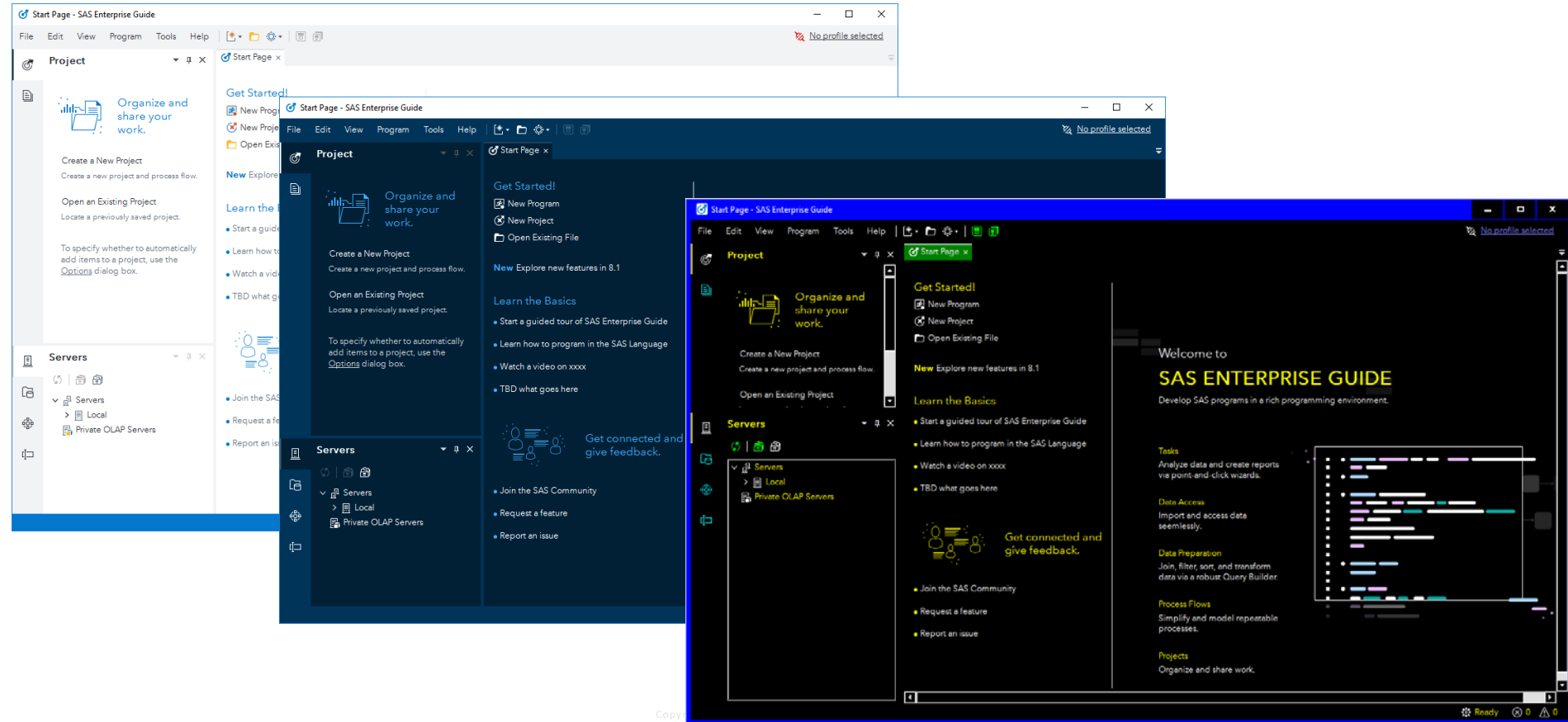
The Plan

Modernize, Yet Keep the Essence

- Why?
 - Supporting the work you do
 - Listening and responding to your feedback
 - Interchangeability with SAS Studio
- How?
 - Attractive, flexible, smart
 - Take advantage of being a desktop app
- When?
 - 8.1 released June 18, 2019
 - 8.2 released November 19, 2019



Meet SAS Enterprise Guide 8

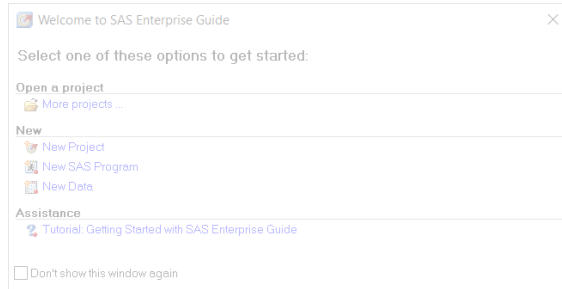




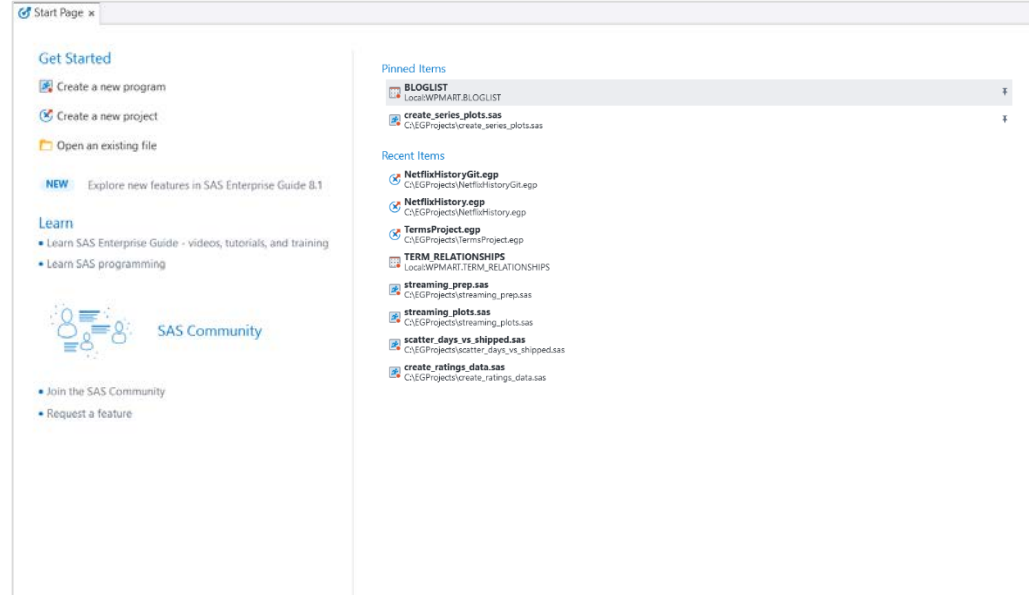
Onboarding

First Impressions Matter

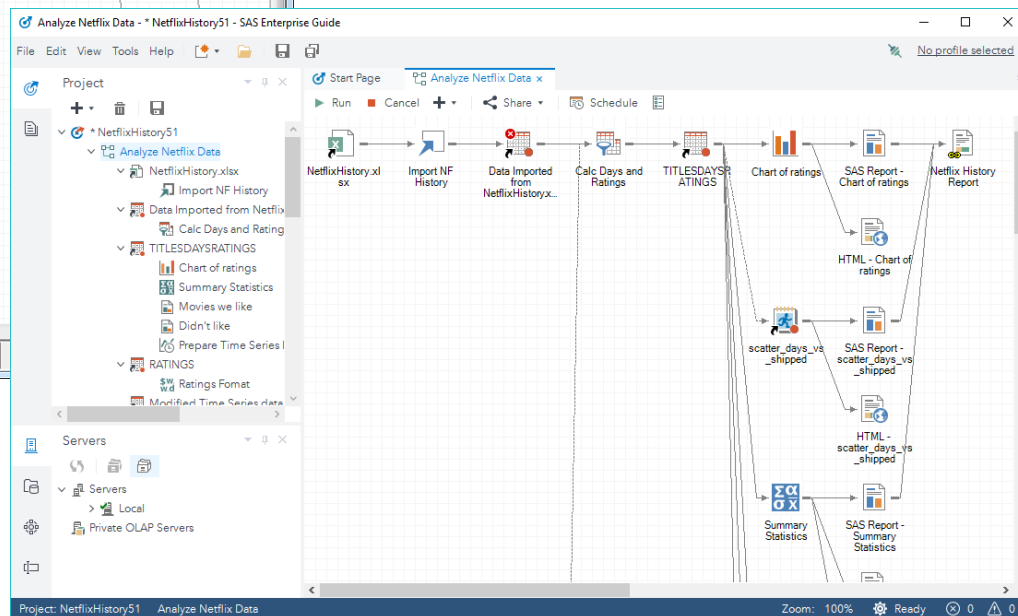
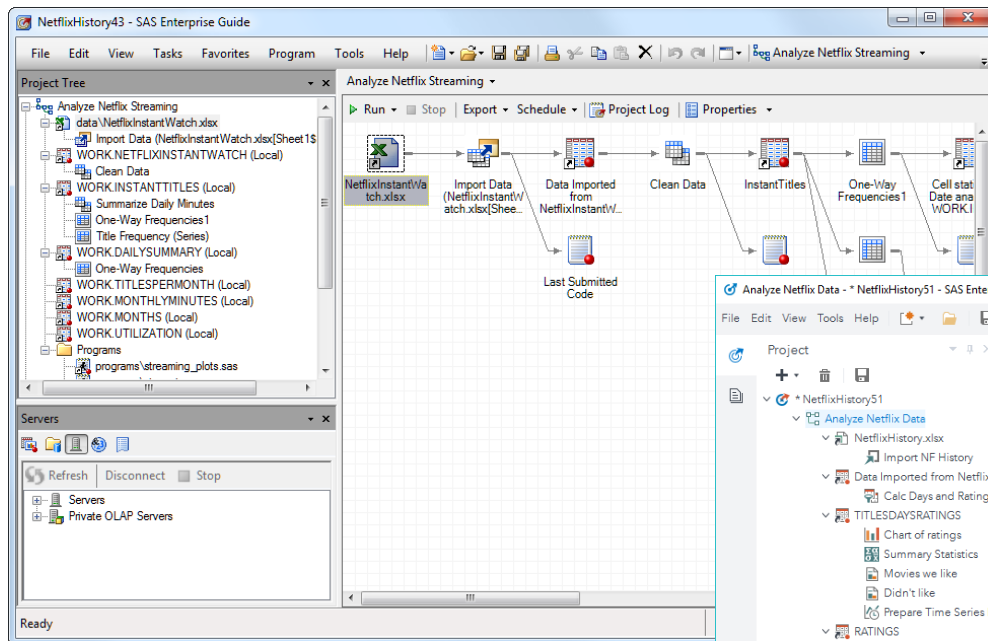
Before



After



Refresh the Look



Refresh the Look

The screenshot shows the SAS Enterprise Guide interface with several annotations in blue boxes and arrows:

- Navigation side bar with panes:** Points to the left sidebar containing the Project, Servers, and Libraries panes.
- Profile connection:** Points to the top right corner where "No profile selected" is displayed.
- workspace area:** Points to the central area containing the Code editor and the Log/Results pane.
- Tools area:** Points to the bottom right area containing the Submission Status and Process FlowServer panes.
- Status bar:** Points to the bottom status bar showing the current project and zoom level.

The Code editor displays the following SAS code:

```
1 /* use a DATA step to subset data test */
2 data localCars;
3   set sashelp.cars(where=(origin="USA"));
4 run;
5
6 title "All USA cars, all variables";
7 /* produces a report of all numeric variables */
8 proc means data=localCars
9   mean stddev median mode n;
10 run;
11
12 title "All USA cars, just MPG numbers";
13 /* narrows the processing to MPG numbers */
14 proc means data=localCars
15   mean stddev median mode n;
16   var mpg_highway mpg_city;
17 run;
18
19 title "All USA cars, just MPG numbers";
20 title2 "by type?";
21 proc means data=localCars
```

The Results pane shows "The MEANS Procedure" with a table of variables and labels:

Variable	Label
MSRP	Engine Size (L)
Invoice	Engine Size (L)
EngineSize	Engine Size (L)
Cylinders	MPG (City)
Horsepower	MPG
MPG_City	MPG (Highway)
MPG_Highway	Weight (LBS)
Weight	Wheelbase (IN)
Wheelbase	Length (IN)
Length	

The Submission Status pane shows the following status information:

Status Information	Run Time	Submitted	Completed
Collecting...		10/13/2018 3:06:08 PM	
Preparing server...			

The Navigation Panes

Most of the panes are trees that contain folders and files. You can do many things from them:

- Create, delete, and rename files and folders.
- Move files and folders with drag and drop.
- Use the context menu to explore all options.



Tip: Drag and drop these panes and use the Layout Guide to help you position them where you like them.



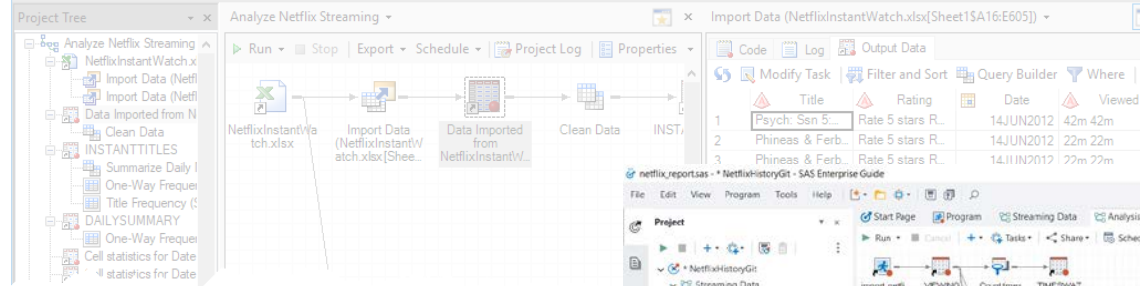
Open Items - At a glance you can see a list of your currently opened files.



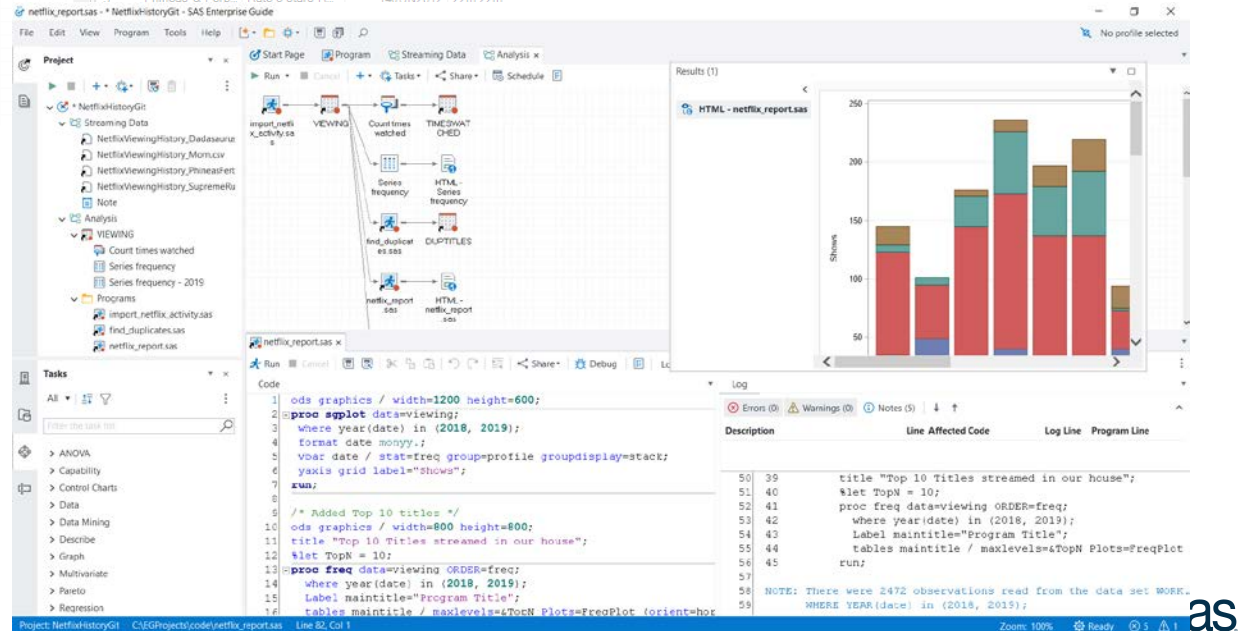
Flexible Layout

Tab groups, floating, docking, oh my!

Before: Max 2 tabs and single monitor



After: Unlimited tabs & multiple monitors



Working with Code, Log, Results

At the Same Time!

After: View ALL

Before: Choose which ONE to view

Program

Program* Log Output Data (4) Results

Refresh Export Send To Create Publish Properties

Percent utilization

Obs	Make	Model	Type	Origin	DriveTrain	MSRP
1	Audi	A6 4.2 Quattro 4dr	Sedan	Europe	All	\$49,690
2	Audi	A8 L Quattro 4dr	Sedan	Europe	All	\$69,190
3	Audi	S4 Quattro	Sedan	Europe	All	\$48,040

* Program x

Run Cancel [Icons] Share Debug Local

Code Results (1) Output Data (4)

```
1 proc sort data=sashelp.cars out=cars;
2 by make;
3 run;
4 data testme;
5 set cars(where=(cylinders eq 8)) nobs=numObs;
6 format dollarsPerHorse dollar20.2 runningAverageHorsepower=runningHorses/_n_;
7 retain runningHorses;
8 by make;
9 if _n_ eq 1 then runningHorses=0; else runningHorses=runningHorses+1;
10 runningAverageHorsepower=runningHorses/_n_;
11 dollarsPerHorse=invoice/Horsepower;
12 run;
13 proc print;
14 run;
15 data _null_;
16 y=(0.1 + 0.2) + 0.3;
17 z=0.1 + (0.2 + 0.3);
18 if y eq z then do;
19 put "equal";
20 end;
21 run;
22 data a;
23 set sashelp.class;
24 if sex eq "F";
25 output;
26 run;
27 data b;
28 set sashelp.class;
29 where sex eq "F";
30 run;
```

Obs	Make	Model	Type	Origin	DriveTrain	MSRP
1	Audi	A6 4.2 Quattro 4dr	Sedan	Europe	All	\$49,690
2	Audi	A8 L Quattro 4dr	Sedan	Europe	All	\$69,190
3	Audi	S4 Quattro 4dr	Sedan	Europe	All	\$48,040
4	Audi	RS 6 4dr	Sports	Europe	Front	\$84,800
5	Audi	S4 Avant Quattro	Wagon	Europe	All	\$49,090
6	BMW	X5 4.4i	SUV	Europe	All	\$52,195
7	BMW	545iA 4dr	Sedan	Europe	Rear	\$54,995
8	BMW	745i 4dr	Sedan	Europe	Rear	\$69,195
9	BMW	745i i 4dr	Sedan	Europe	Rear	\$73,195

Log

Errors (0) Warnings (0) Notes (17) ↓ ↑

```
101
102 NOTE: There were 9 observations read from the
103       WHERE sex='F';
104 NOTE: The data set WORK.B has 9 observations.
105 NOTE: DATA statement used (Total process time
106       real time           0.01 seconds
107       cpu time             0.01 seconds
108
109
110 57
111 58          %LET _CLIENTTASKLABEL=;
112 59          %LET _CLIENTPROCESSFLOWNAME=;
```



Working with Code, Log, Results

Drag a tab then use the Layout Guide to help you position the tab

The screenshot displays the SAS Enterprise Guide interface with the following components:

- Code Editor:** Contains SAS code for generating a stacked bar chart and a frequency plot.
- Log Window:** Shows the execution log, including a note about the SGPLOT procedure and the number of observations read.
- Results Window:** Displays a stacked bar chart titled "Shows" and a frequency plot titled "Program Title".

Code Editor Content:

```
1 ods graphics / width=1200 height=600;
2 proc sgplot data=viewing;
3   where year(date) in (2018, 2019);
4   format date monyy.;
5   vbar date / stat=freq group=profile groupdisplay=stack;
6   yaxis grid label="Shows";
7 run;
8
9 /* Added Top 10 titles */
10 ods graphics / width=800 height=800;
11 title "Top 10 Titles streamed in our house";
12 slet TopN = 10;
13 proc freq data=viewing ORDER=freq;
14   where year(date) in (2018, 2019);
15   label maintitle="Program Title";
16   tables maintitle / maxlevels=&TopN Plots=FreqPlot (orient=horizontal scale=pe);
17 run;
```

Log Window Content:

```
37 34 yaxis grid label="Shows";
38 25 run;
39
40 NOTE: PROCEDURE SGPLOT used (Total process time):
41   real time      0.31 seconds
42   cpu time       0.11 seconds
43
44 NOTE: There were 2472 observations read from the data set WORK.VIEWING.
45   WHERE YEAR(date) in (2018, 2019);
46
47 36
48 37 /* Added Top 10 titles */
```

Results Window Content:

The Results window displays two plots:

- Shows:** A stacked bar chart showing the frequency of shows for different years (2018, 2019). The y-axis is labeled "Shows" and ranges from 0 to 250. The x-axis represents different shows.
- Program Title:** A horizontal frequency plot showing the distribution of program titles. The x-axis represents the program title and the y-axis represents the frequency.

Working with Code, Log, Results

Or use preset layouts

The screenshot displays the SAS Enterprise Guide interface for a project named 'netflix_reports.sas'. The left sidebar shows a tree view with 'Project' expanded, listing 'Open Items', 'Servers', 'SAS Folders', 'Tasks', 'Prompt Manager', 'Submission Status', 'Start Page', and 'Search'. The main window is divided into three panes: 'Code', 'Log', and 'Results (1)'. The 'Code' pane shows SAS code for a PROC SGPLOT, including a WHERE clause for the years 2018 and 2019, and a FREQPLOT statement. The 'Log' pane shows the execution log, including the PROC SGPLOT execution time and the number of observations read. The 'Results' pane displays a stacked bar chart titled 'Shows' on the y-axis, showing the frequency of shows across different categories. The chart has a red base layer and a teal top layer. The status bar at the bottom indicates 'Project: NetflixHistoryGit' and 'C:\EGProjects\code\netflix_reports.sas'.

```
/* width=1200 height=600;
data=viewing;
date) in (2018, 2019);
monyy.;
stat=freq group=profile groupdisplay=stack;
label="Shows";

10 titles */
/* width=800 height=800;
house";

N Plots=FreqPlot (orient=horizontal scale=percent);

where year
Label main
tables main
run;

Log
Errors (0) Warnings (0)

Description Line Affected Code Log Line Program Line

37 34 yaxis grid label="Shows";
38 35 run;
39
40 NOTE: PROCEDURE SGPLOT used (Total process time):
41 real time 0.31 seconds
42 cpu time 0.11 seconds
43
44 NOTE: There were 2472 observations read from the data set WORK.VIEWING.
45 WHERE YEAR(date) in (2018, 2019);
46
47 36
48 37 /* Added Top 10 titles */
```

Category	Red Layer (Frequency)	Teal Layer (Frequency)	Total Frequency
1	120	20	140
2	100	10	110
3	150	30	180
4	140	50	190
5	140	50	190
6	90	10	100
7	80	90	170
8	130	10	140
9	120	10	130
10	10	10	20

Float a Tab Within the App or To Another Monitor

The screenshot shows the SAS interface with a tab labeled 'BLOGLIST'. A context menu is open over the tab, displaying the following options: Close, Close others, Close all, Full screen, Float, New vertical tab group, and New horizontal tab group. The 'Float' option is highlighted by the mouse cursor. The background shows a table with columns: blog_id, name, last_updated, and public.

blog_id	name	last_updated	public
1	2 SAS Voices	07JUN2019:15:42:10	1
2	3 JMP Blog	22AUG2017:13:48:07	0
3	4 The Business Forecasting Deal	27MAR2019:13:58:47	1
4	5 The Text Frontier	11NOV2016:19:29:13	1
5	6 A Shot in the Arm	20SEP2016:14:12:48	1
6	7 SAS Analytics U Blog	30MAY2018:19:11:52	1

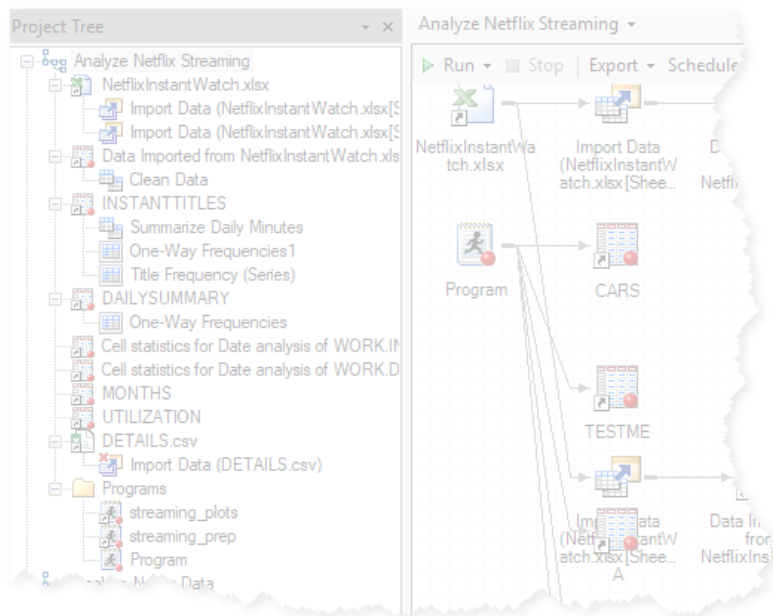
The screenshot shows the SAS interface with a tab labeled 'BLOGLIST'. A context menu is open over the tab, displaying the following options: Close, Close others, Close all, and Dock as tabbed document. The 'Dock as tabbed document' option is highlighted by the mouse cursor. The background shows a table with columns: path, registered, last_updated, and public.

path	registered	last_updated	public
com/	10JUN2011:12:52:59	07JUN2019:15:42:10	1
/content/jmp/	10JUN2011:12:54:06	22AUG2017:13:48:07	0
/content/forecasting/	10JUN2011:14:12:56	27MAR2019:13:58:47	1
/content/text-mining/	10JUN2011:14:19:41	11NOV2016:19:29:13	1
/content/hls/	10JUN2011:14:24:18	20SEP2016:14:12:48	1
/content/academic/	10JUN2011:14:32:02	30MAY2018:19:11:52	1
/content/insurance/	10JUN2011:14:45:58	21SEP2016:18:01:11	1

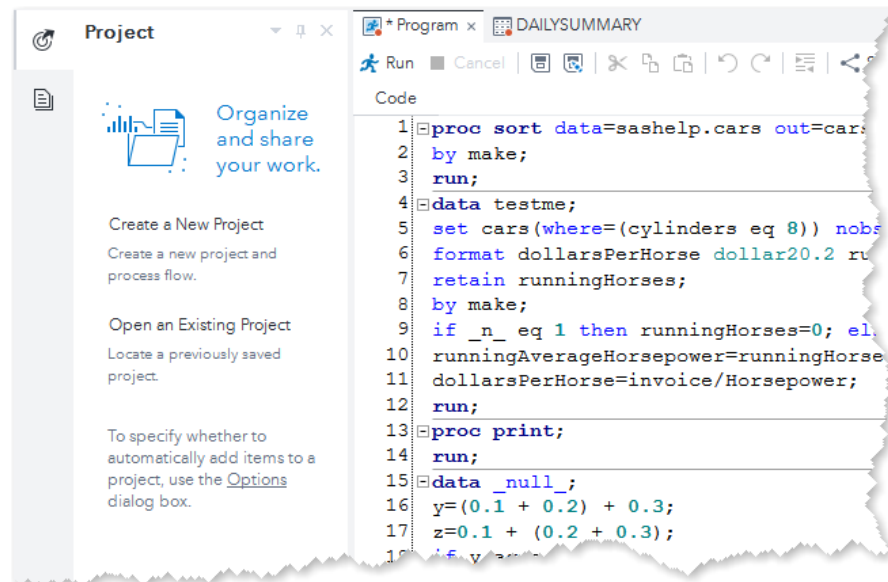
Working with a Project or NOT!

Program-centric Development

Before: Project required!

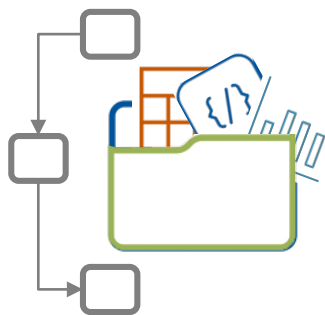


After: Project optional!



Adding Files to a Project

Automatically



I love the way EG adds files to projects and process flows for me



Project and Process Flows

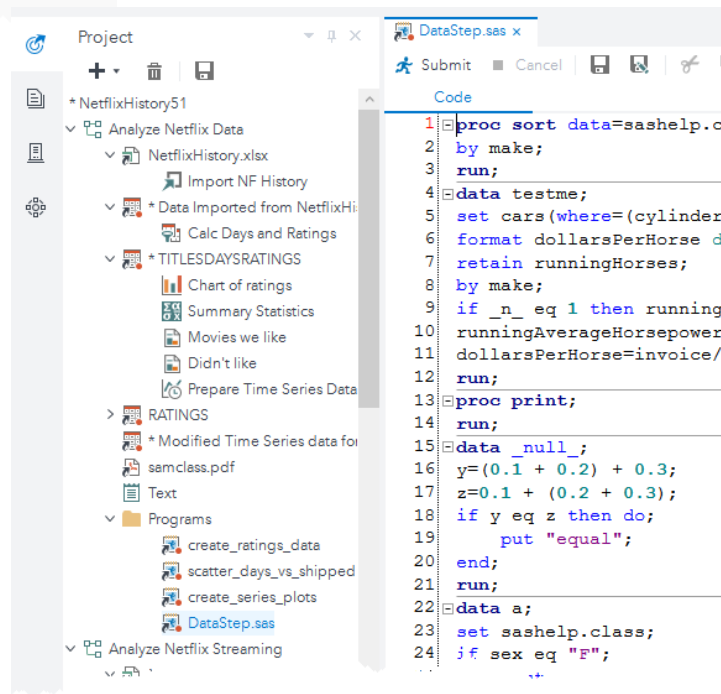
[Reset](#)

☒ When an item is opened automatically add to the project

General

☐ Enable automatic file new projects

Option to automatically add items to a project

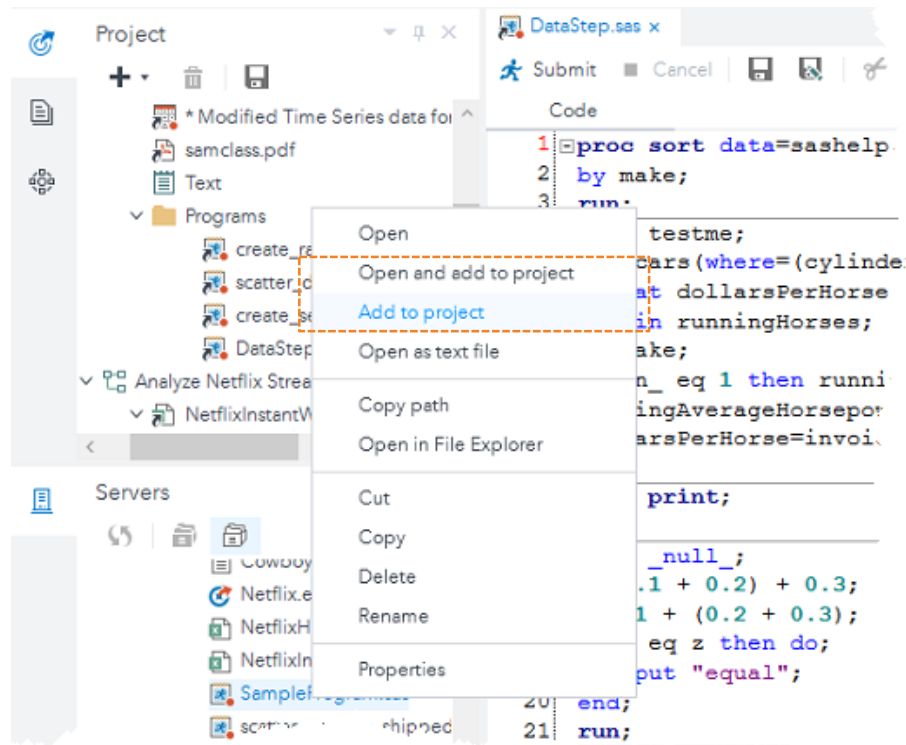


Add Files to a Project

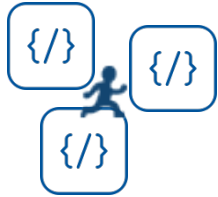
Manually



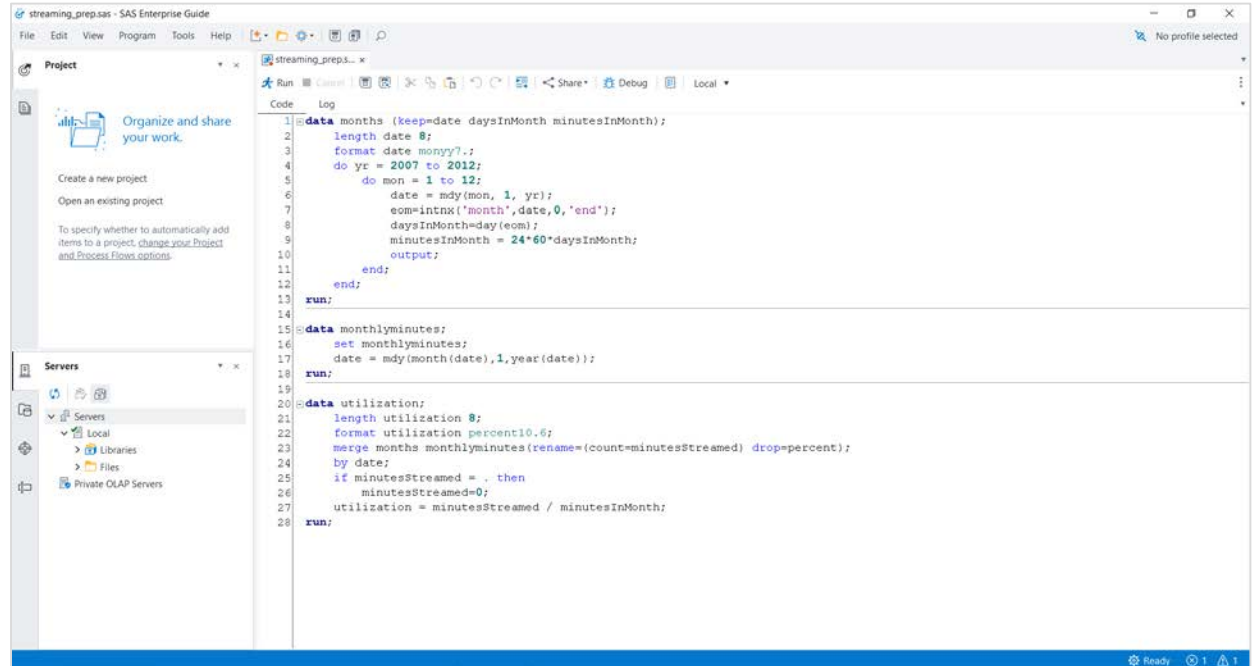
I love the idea of a projects and process flows, but I want to add files myself



No Project!



I just want to program – not
interested in projects or
process flows



Items That Don't Require a Project

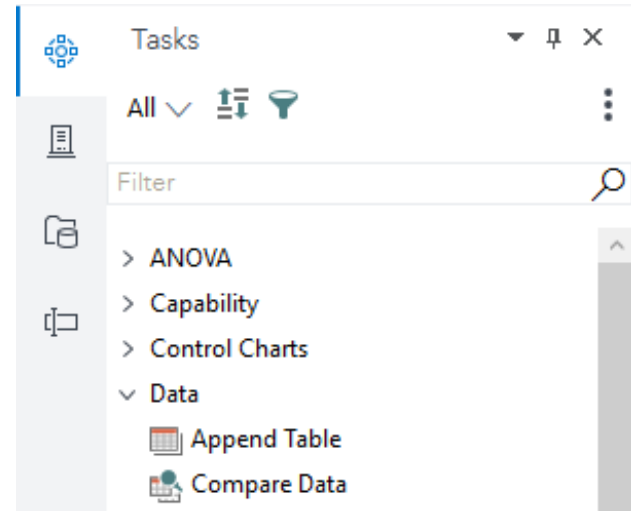
These items do not require a project. They can be opened and edited in a 'project-less' environment.

- Programs
- Data
- OLAP Cubes
- SAS Report Results
- SAS Visual Analytics Reports
- Logs

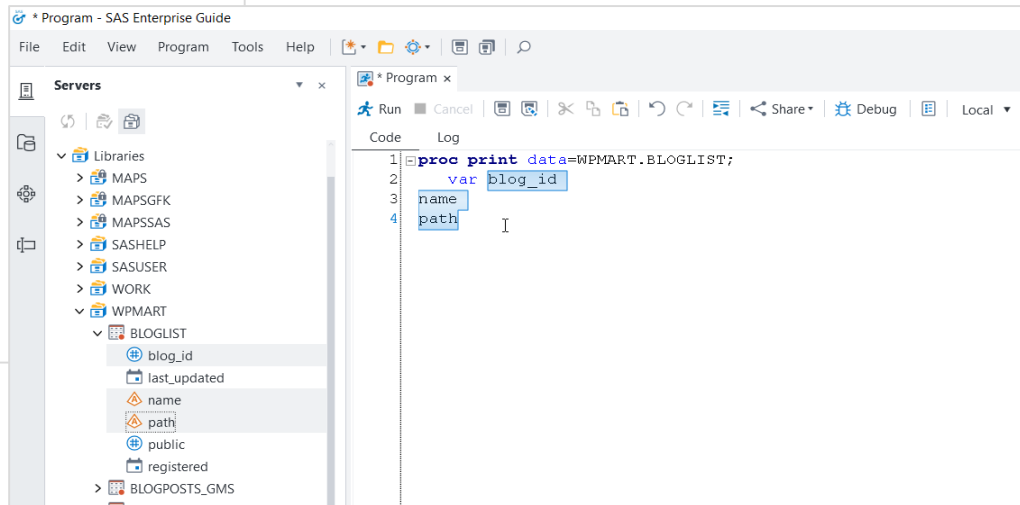
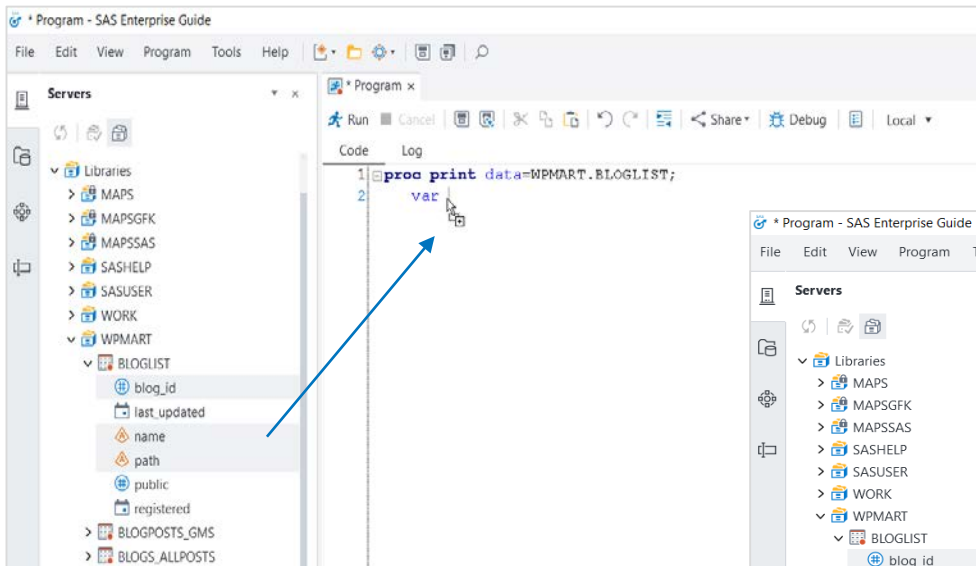


Working with Tasks

- Tasks now available from the Tasks pane
- Query Builder and Import are automatic favorites
- Still need a project to use tasks
- SAS Studio tasks are available



Drag and Drop Data Set Names and Variables into a Program



Integration with Git

The screenshot displays the SAS Git interface with the following components:

- Git Repositories:** A sidebar on the left showing 'GitEG' as the current repository.
- Start Page:** The main header area with tabs for 'Start Page' and 'GitEG x'. It includes buttons for 'Clone', 'Add', 'Commit (0)', and 'History'.
- Graph:** A visual representation of the commit history on the left side of the commit list.
- Commit List:** A table of commits with columns for Message, Author, Date, and Commit ID.
- Commit Details:** A section below the commit list showing the author, date, and commit hash for the selected commit.
- Diff View:** A section on the right showing the differences between the current commit and the previous one.

Message	Author	Date	Commit ID
✓ master origin/master complex manipulations for step 2	ChicopeeAmy	2/7/2020 1:58:40 PM	f657cca
origin/EngineReporting final bar panel	SGF	2/7/2020 1:57:51 PM	acbcd6b
updates to bar panels	SGF	2/7/2020 1:57:01 PM	d194e94
first reports	SGF	2/7/2020 1:47:31 PM	f898f1d
Merge 'FETCH_HEAD' into branch master	Merge	2/7/2020 1:10:14 PM	dadfba4
first stab at gas data set manipulation	ChicopeeAmy	2/7/2020 1:06:10 PM	72104d8
more complex manipulation	SGF	2/7/2020 12:47:39 PM	19c360c
initial basic data prep for cars	SGF	2/7/2020 12:43:22 PM	2617e2e
Merge 'FETCH_HEAD' into branch master	Merge	2/7/2020 12:31:18 PM	fbeb32f
origin/StudioOnlyWork more separate Studio work	SGF	1/31/2020 2:57:54 PM	84858c0
continued EG work	ChicopeeAmy	1/31/2020 2:54:18 PM	8d764df
different sashelp.class work	SGF	1/31/2020 2:52:50 PM	0c5a2f0

Author: ChicopeeAmy | Date: 1/31/2020 2:54:18 PM | Commit hash: 8d764dfb2db5a4f4f39b9824669a8479b1aa52c

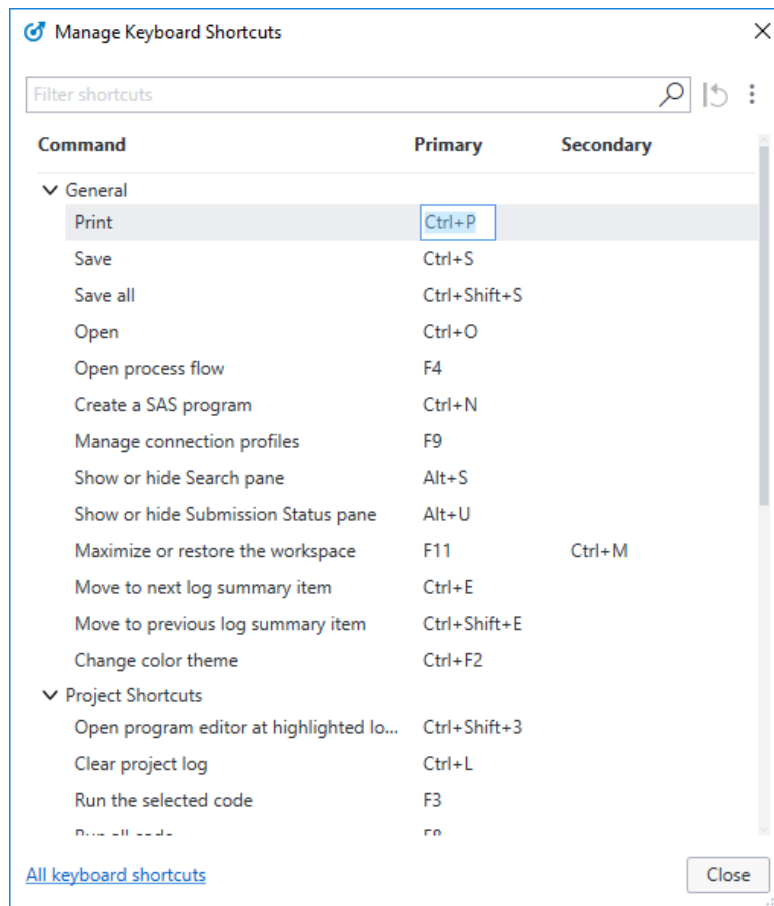
continued EG work

even more EG changes@055bf5

- enginesize.sas

```
1 diff --git a/enginesize.sas b/enginesize.sas
2 index 5c073af..e7fe305 100644
3 --- a/enginesize.sas
4 +++ b/enginesize.sas
5 ----- Difference (@@ -5,4 +5,5 @@) -----
6      *additions made in EG;
7      *another EG change;
8      *and even more EG changes;
9 +    *continuing to work in EG;
10 run;
11
```

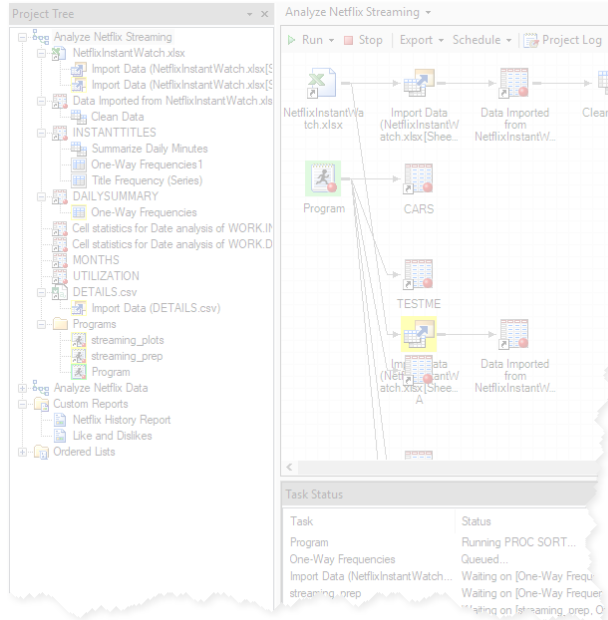
Managing Keyboard Shortcuts



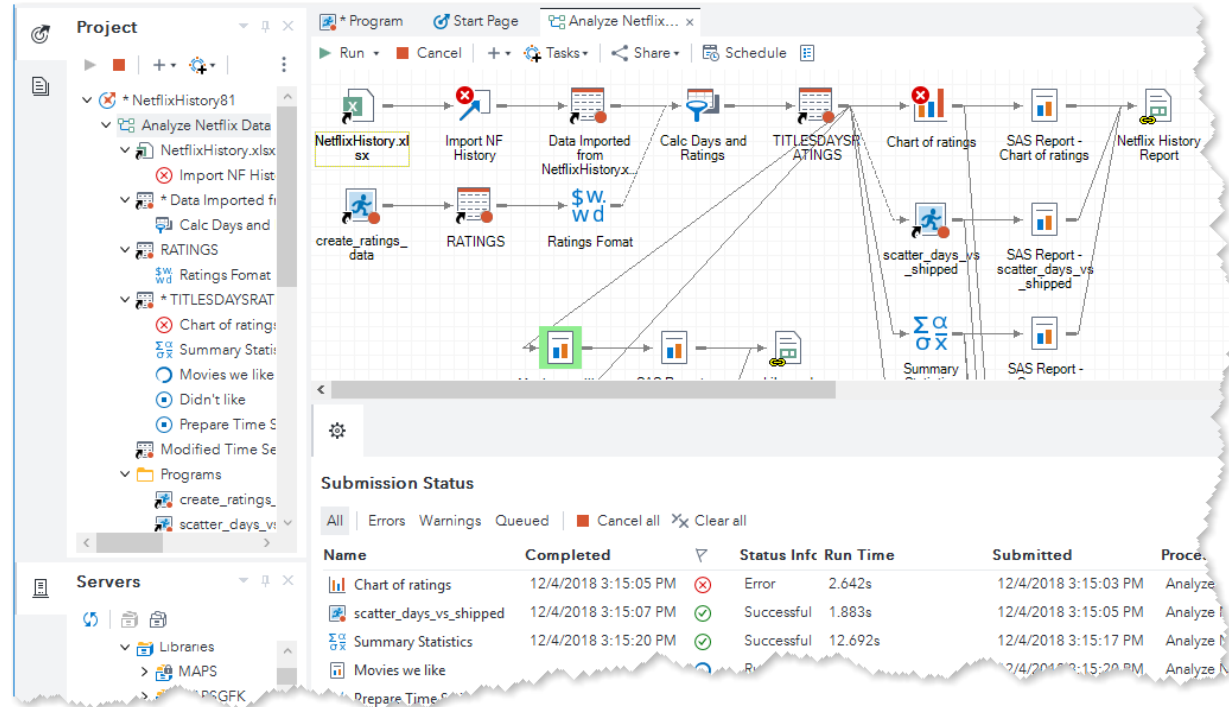
What's Happening?

Submission Feedback

Before: Some status information

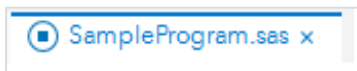


After: Knowing status at a glance

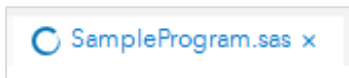


Running Multiple Items without Locking the UI

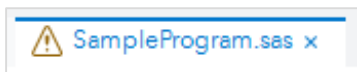
Status icons



Program is queued in the order of submission.



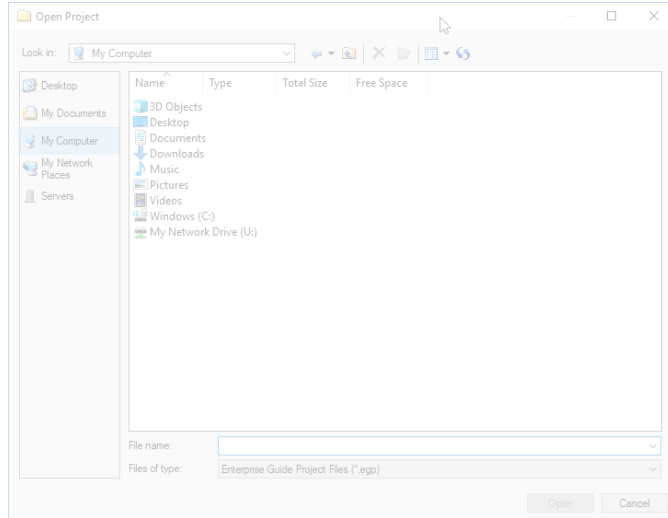
Program is in the process of being submitted.



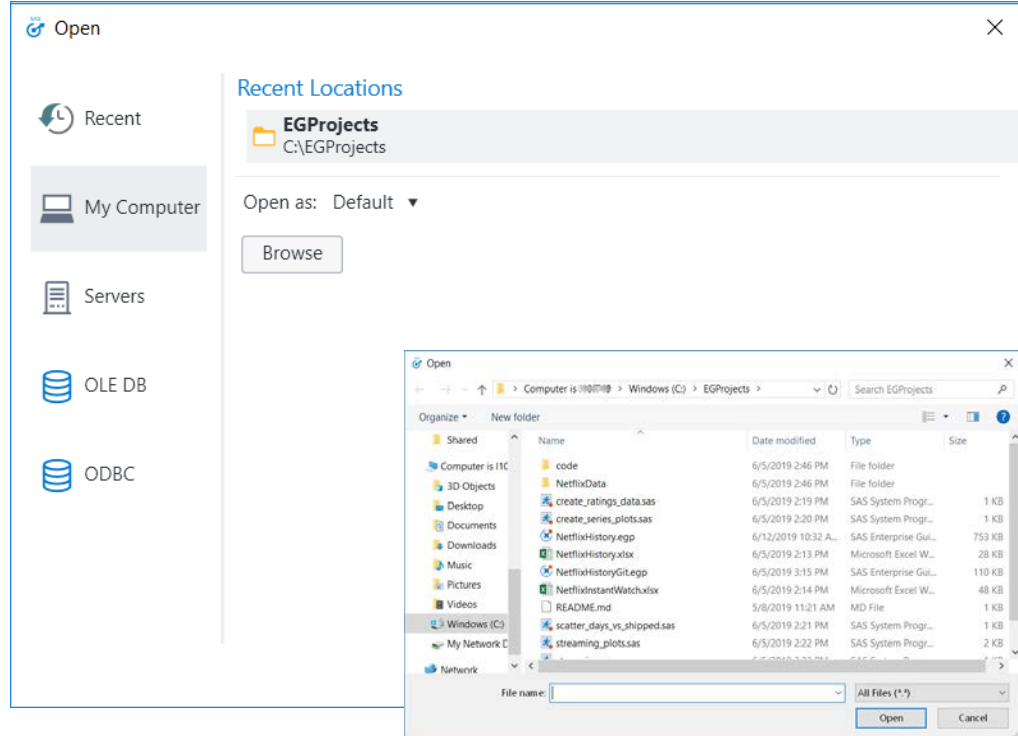
Status icons will show you, at a quick glance, if your program ran with errors or warnings.

Opening Files

Before



After



Output Data Performance Options

- New options for controlling what output data to open automatically

When generating tables in a program, automatically open:

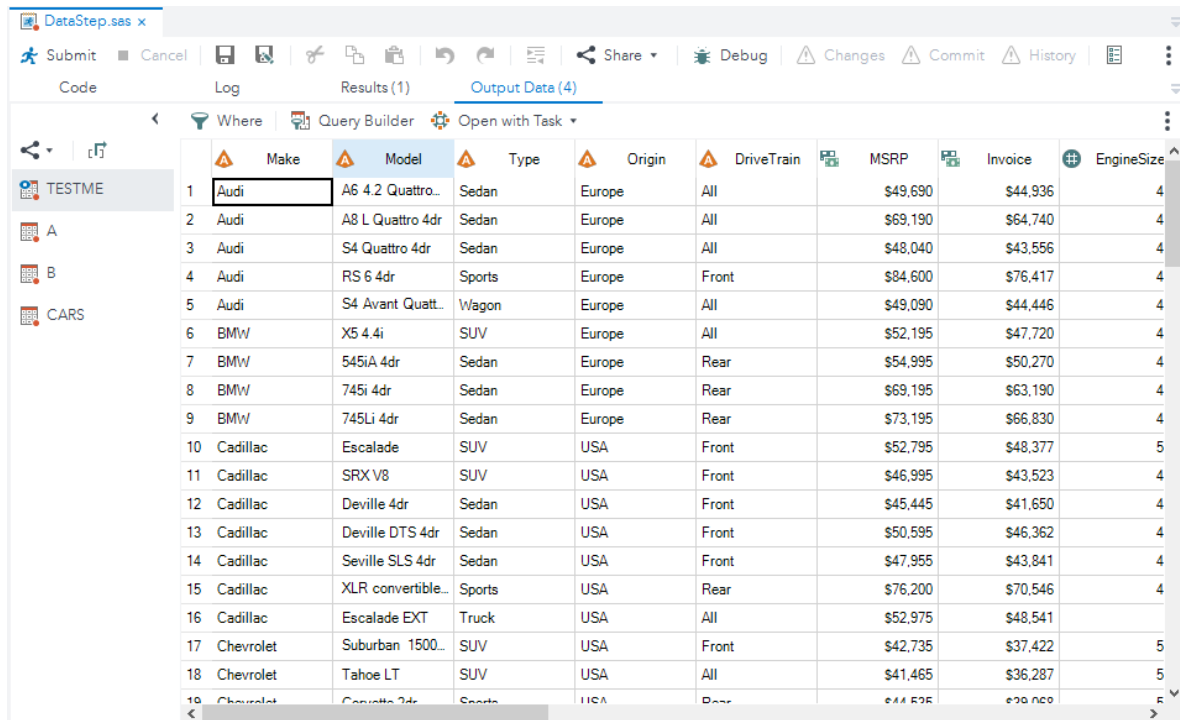
Last table ▾

☒ Unlock open data after inactivity (minutes):

3

☐ Unlock all open data before running code

☐ Automatically refresh open data tabs after running code



The screenshot shows the SAS DataStep interface. The 'Output Data (4)' tab is active, displaying a table of car data. The table has columns: Make, Model, Type, Origin, DriveTrain, MSRP, Invoice, and EngineSize. The data is sorted by Make and Model. The first row is highlighted.

	Make	Model	Type	Origin	DriveTrain	MSRP	Invoice	EngineSize
1	Audi	A6 4.2 Quattro...	Sedan	Europe	All	\$49,690	\$44,936	4
2	Audi	A8 L Quattro 4dr	Sedan	Europe	All	\$69,190	\$64,740	4
3	Audi	S4 Quattro 4dr	Sedan	Europe	All	\$48,040	\$43,556	4
4	Audi	RS 6 4dr	Sports	Europe	Front	\$84,600	\$76,417	4
5	Audi	S4 Avant Quatt...	Wagon	Europe	All	\$49,090	\$44,446	4
6	BMW	X5 4.4i	SUV	Europe	All	\$52,195	\$47,720	4
7	BMW	545iA 4dr	Sedan	Europe	Rear	\$54,995	\$50,270	4
8	BMW	745i 4dr	Sedan	Europe	Rear	\$69,195	\$63,190	4
9	BMW	745Li 4dr	Sedan	Europe	Rear	\$73,195	\$66,830	4
10	Cadillac	Escalade	SUV	USA	Front	\$52,795	\$48,377	5
11	Cadillac	SRX V8	SUV	USA	Front	\$46,995	\$43,523	4
12	Cadillac	Deville 4dr	Sedan	USA	Front	\$45,445	\$41,650	4
13	Cadillac	Deville DTS 4dr	Sedan	USA	Front	\$50,595	\$46,362	4
14	Cadillac	Seville SLS 4dr	Sedan	USA	Front	\$47,955	\$43,841	4
15	Cadillac	XLR convertible...	Sports	USA	Rear	\$76,200	\$70,546	4
16	Cadillac	Escalade EXT	Truck	USA	All	\$52,975	\$48,541	4
17	Chevrolet	Suburban 1500...	SUV	USA	Front	\$42,735	\$37,422	5
18	Chevrolet	Tahoe LT	SUV	USA	All	\$41,465	\$36,287	5
19	Chevrolet	Camaro 3dr	Sports	USA	Rear	\$44,535	\$39,069	5

Search

- Search panel that can toggle on and off
- Search for items in a project and in the Open Files pane

The screenshot displays the SAS Enterprise Guide interface. The main workspace shows a project workflow for 'Analyze Netflix Data'. The workflow starts with 'NetflixHistory.xlsx' and proceeds through steps like 'Import NF History', 'Data Imported from Netflix...', 'Calc Days and Ratings', 'TITLES DAY RATINGS', 'Chart of ratings', and 'HTML - Chart of ratings'. Other steps include 'create_ratings_data.sas', 'RATINGS', 'Ratings Format', 'Movies we like', 'SAS Report - Movies we like', 'Like and Dislikes', 'Summary Statistics', and 'HTML - Summary Statistics'.

On the left, the 'Project' pane shows a tree view of the project structure, including 'NetflixHistory', 'Analyze Netflix Data', 'NetflixHistory.xlsx', 'Import NF History', 'Data Imported from NetflixHistory', 'Calc Days and Ratings', 'RATINGS', 'Ratings Format', 'TITLES DAY RATINGS', 'Chart of ratings', 'Summary Statistics', 'Movies we like', 'Didn't like', and 'Prepare Time Series Data'.

At the bottom, the 'Search' panel is open, showing a search for 'Netflix'. The search results are displayed in a table with columns for 'Name', 'Type', and 'Process Flow'.

Name	Type	Process Flow
Data Imported from NetflixHistory.xlsx	Data	Analyze Netflix Data
Data Imported from NetflixInstantWatch.xlsx	Data	Analyze Netflix Streaming
Calc Days and Ratings	Query	Analyze Netflix Data
Clean Data	Query	Analyze Netflix Streaming
Import NF History	Task	Analyze Netflix Data
Import Data (NetflixInstantWatch.xlsx[Sheet1\$A16:E605])	Task	Analyze Netflix Streaming
Log	Log	Analyze Netflix Data
Icon	Icon	Analyze Netflix Data

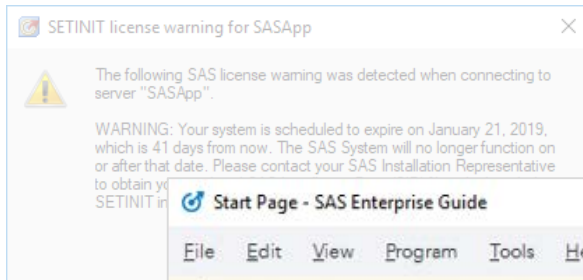


Notifications

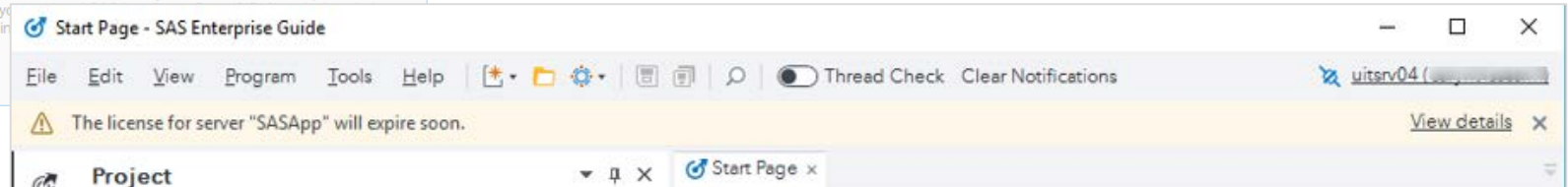
What do you really need to know?

- Get out of your way so you can continue working
- Better messaging

Before



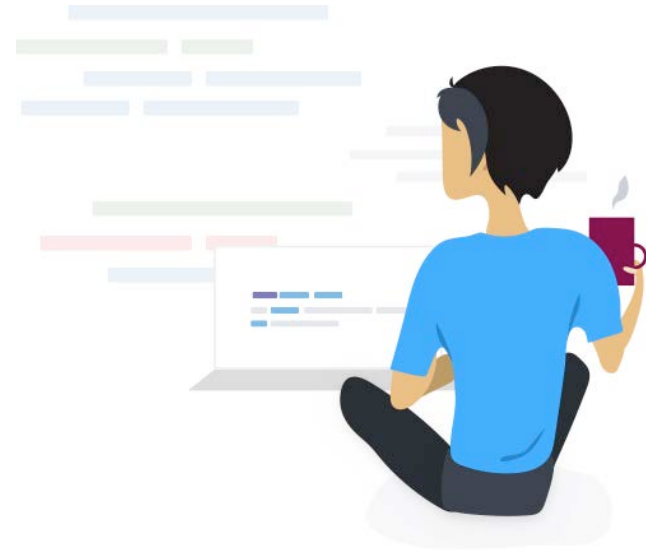
After





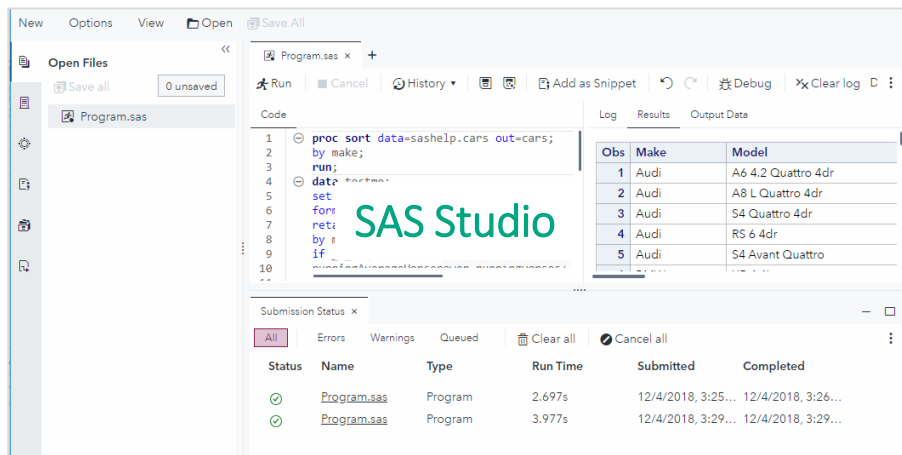
Behind the Scenes

- Always responsive
 - Gracefully handle server unresponsiveness
- More responsible with open data
 - Not locking as long
 - Not opening when unnecessary
- Improved handling of large project logs
- Product help is now online!



SAS Enterprise Guide and SAS Studio

Future Forward



The screenshot shows the SAS Studio web interface. On the left is a sidebar with icons for Open Files, Save all, and Program.sas. The main area has a menu bar (New, Options, View, Open, Save All) and a toolbar (Run, Cancel, History, Add as Snippet, Debug, Clear log). The Code pane contains SAS code for sorting data. The Log pane shows a table of results. The Submission Status pane at the bottom shows a table of program submissions.

Code

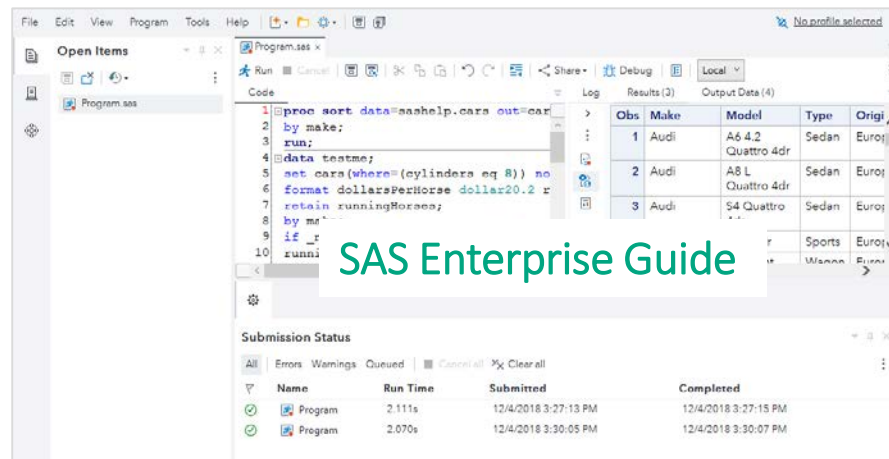
```
1 proc sort data=sashelp.cars out=cars;  
2 by make;  
3 run;  
4 data testme;  
5 set cars;  
6 format dollarsPerHorse dollar20.2;  
7 retain runningHorse;  
8 if _type_ = 'Sedan' then runningHorse = 1;  
9 if _type_ = 'SUV' then runningHorse = 2;  
10 if _type_ = 'Truck' then runningHorse = 3;
```

Log

Obs	Make	Model
1	Audi	A6 4.2 Quattro 4dr
2	Audi	A8 L Quattro 4dr
3	Audi	S4 Quattro 4dr
4	Audi	RS 6 4dr
5	Audi	S4 Avant Quattro

Submission Status

Status	Name	Type	Run Time	Submitted	Completed
✓	Program.sas	Program	2.697s	12/4/2018, 3:25...	12/4/2018, 3:26...
✓	Program.sas	Program	3.977s	12/4/2018, 3:29...	12/4/2018, 3:29...



The screenshot shows the SAS Enterprise Guide desktop application. It features a menu bar (File, Edit, View, Program, Tools, Help) and a toolbar. The Open Items pane on the left shows Program.sas. The main area has a menu bar (Run, Cancel, History, Add as Snippet, Debug, Local) and a toolbar. The Code pane contains SAS code for sorting data. The Log pane shows a table of results. The Submission Status pane at the bottom shows a table of program submissions.

Code

```
1 proc sort data=sashelp.cars out=cars;  
2 by make;  
3 run;  
4 data testme;  
5 set cars(where=(cylinders eq 8));  
6 format dollarsPerHorse dollar20.2;  
7 retain runningHorse;  
8 if _type_ = 'Sedan' then runningHorse = 1;  
9 if _type_ = 'SUV' then runningHorse = 2;  
10 if _type_ = 'Truck' then runningHorse = 3;
```

Log

Obs	Make	Model	Type	Origin
1	Audi	A6 4.2 Quattro 4dr	Sedan	Euro
2	Audi	A8 L Quattro 4dr	Sedan	Euro
3	Audi	S4 Quattro	Sedan	Euro

Submission Status

Name	Run Time	Submitted	Completed
Program	2.111s	12/4/2018 3:27:13 PM	12/4/2018 3:27:15 PM
Program	2.070s	12/4/2018 3:30:05 PM	12/4/2018 3:30:07 PM



Join the Discussion

Tell us what you like, don't like, what more you need

On SAS Communities:

- SAS Enterprise Guide community – find it under Business Intelligence
- SASware ballot ideas





sas.com