

MODUL 5

SAS® STUDIO: SAS BASIC PROGRAMMING

THEME DESCRIPTION

Students understand the basic concepts of the SAS language consisting of data steps (for data preparation) and procedures (or PROC) for analytical activities and develop IT knowledge and skills using SAS on data of any size to perform various types of analysis and reporting on big data.

WEEKLY LEARNING OUTCOMES (SUB-LESSONS)

CLO-2-Sub-CLO-5, Able to apply the CRISP-DM framework to provide a systematic and orderly way to carry out data mining projects for knowledge discovery defined in databases as a process for finding useful information and patterns in data-C2.

Through the following learning steps:

1. Preface: DATA & PROC steps and SASHDAT
2. Creating a new SAS program
3. Visual Programmer versus SAS Programmer
4. SAS Studio Auto-Generated Task Code
5. What's in the SAS® OnDemand for Academics Edition?
6. Uploading, Accessing Local Data and Visualized

PRACTICUM SUPPORT

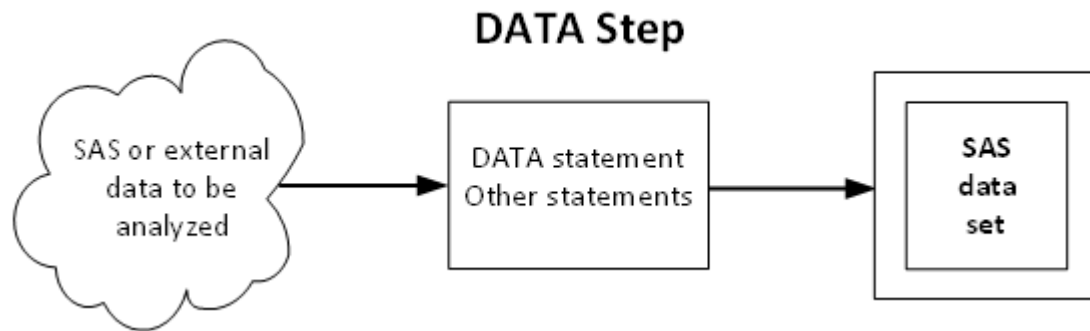
- a. Windows Operating System
- b. (any) Browser Application

PREFACE

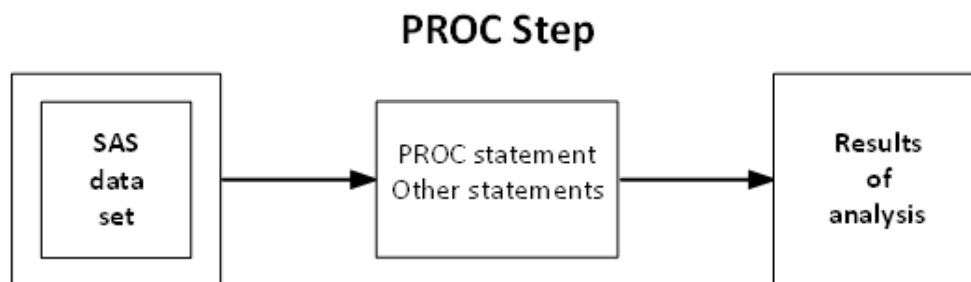
Last week we have practiced data steps and simple steps. The illustrations of these steps describes already in the theory class on "How SAS® Works" I Programming", which consists of data steps and process steps.

SAS Language Elements

- ▶ The SAS language consists of statements, expressions, options, formats, and functions similar to many other programming languages. In SAS, however, you use these elements within one of two groups of SAS statements:
 - ◆ DATA steps
 - ◆ PROC steps
- ▶ **Data steps** are a collection of statements that can help create, modify, and control the output.
- ▶ A DATA step consists of a group of statements in the SAS language that create or manipulate SAS data sets. It is called a DATA step because every step begins with a DATA statement. The following figure shows how a DATA step creates a SAS data set in its simplest form:



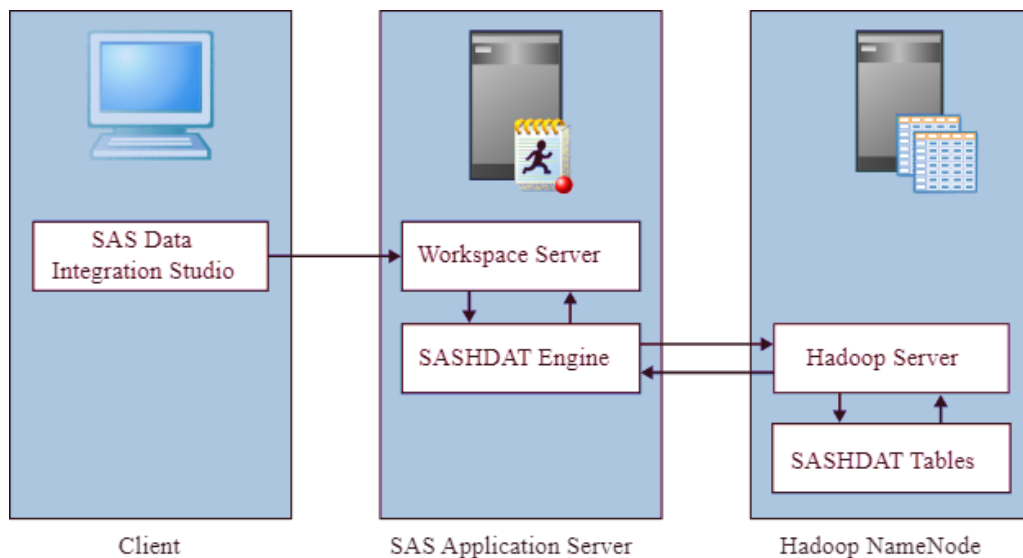
- ▶ Once your data is accessible as a SAS data set, you can analyze the data and write reports by using SAS procedures. The following figure shows how a PROC step can be used to operate on a SAS data set in its simplest form:



- ▶ A group of procedure statements is called a PROC step. Every PROC step begins with a statement whose name begins with the word PROC and is followed by a keyword that describes the purpose of the procedure.
- ▶ Using a PROC step, you can perform simple tasks such as sorting data in a data set (PROC SORT) and printing a data set (PROC PRINT).
- ▶ In addition, you can analyze data in SAS data sets to produce statistics, tables, reports, charts, and plots; create SQL queries; and perform other analyses and operations on your data.

SASHDAT (SAS High Performance Data file)

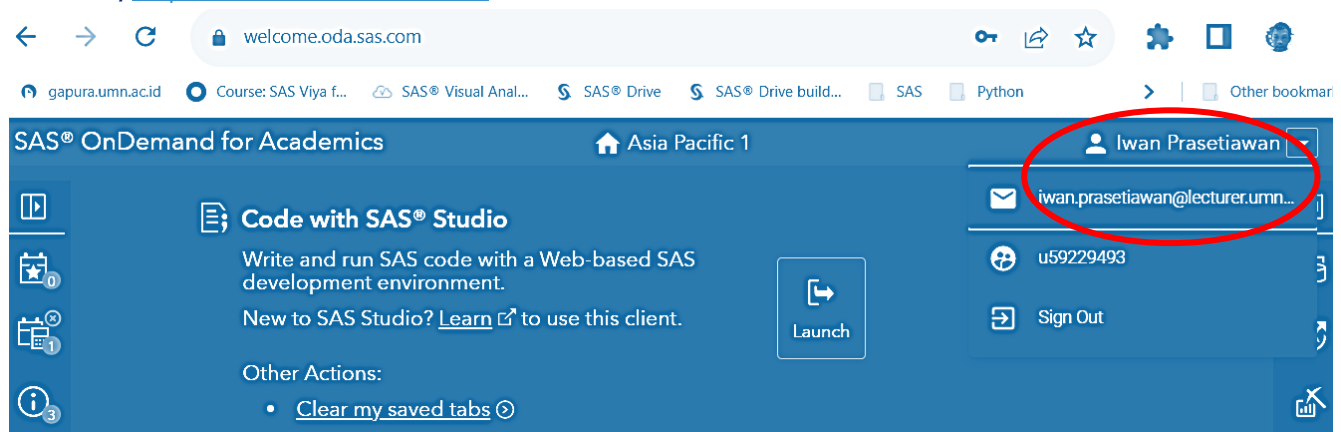
- ▶ The format of an **in-memory table** is called SASHDAT format. SASHDAT is a SAS data storage structure that is designed specifically for high-performance, concurrent data access.
- ▶ A disk file that is in SASHDAT format can easily be loaded as a table without having to do much more than read the header
- ▶ With SASHDAT, your data is broken down into blocks. When saved to HDFS, those blocks are distributed across multiple hosts. When CAS loads SASHDAT from HDFS, the blocks that make up your table are read from the Hadoop DataNode hosts and re-assembled into a single logical table in the CAS memory space.
- ▶ The benefit of using SASHDAT is that it is often the fastest, most-efficient way to (re-)load data into CAS
- ▶ The following figure provides a logical view of connecting to a SASHDAT library (sometimes referred to as SAS Data in HDFS library):



PRACTICUM STEPS

1. Login to SAS Studio

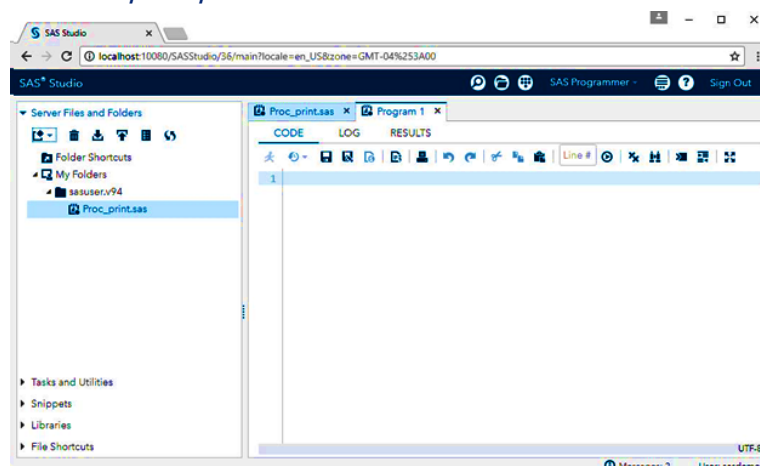
Start, <https://welcome.oda.sas.com>



Click Launch to start SAS® Programming Language using SAS® Studio in the next 2. point below.

2. Creating a new SAS program

- Get into SAS Studio, and on the left-hand pane the Server Files and Folders select the New icon and then select SAS Program(F4).
- SAS Studio allows a user to have multiple programs open at the same time and each program will have their own CODE, LOG, and RESULTS tabs:



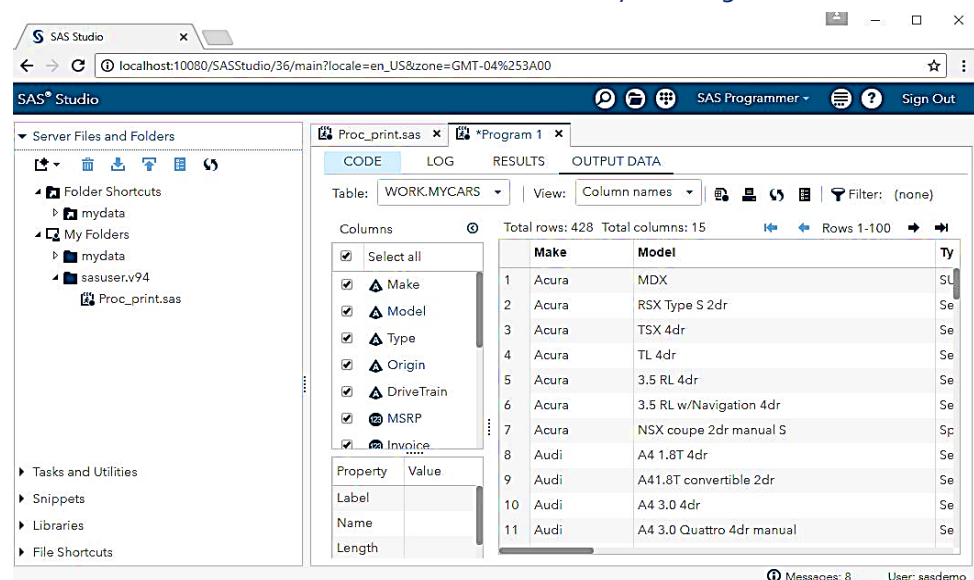
- c. Now, type some data from the SASHELP library into the WORK library using both data step code and a PROC. Type in the following code into the CODE tab of the new Program 1 section:

```

1 /* Data set code to copy a data table */
2 data WORK.MYCARS;
3   set SASHELP.CARS;
4 run;
5
6 /* PROC code to copy a data table */
7 proc copy out=WORK in=SASHELP;
8   select CLASS;
9 run;
10

```

- d. Submit this code and notice that SAS Studio adds an additional tab to your Program 1 section labeled OUTPUT DATA:



The screenshot shows the SAS Studio interface. On the left is the 'Server Files and Folders' pane. The main window has tabs for 'CODE', 'LOG', 'RESULTS', and 'OUTPUT DATA'. The 'OUTPUT DATA' tab is active, showing a table with columns 'Make', 'Model', and 'Type'. The table contains 11 rows of data. Below the table, there is a 'Columns' list with checkboxes for 'Make', 'Model', 'Type', 'Origin', 'DriveTrain', 'MSRP', and 'Invoice'. A 'Property Value' table is also visible at the bottom left of the main window.

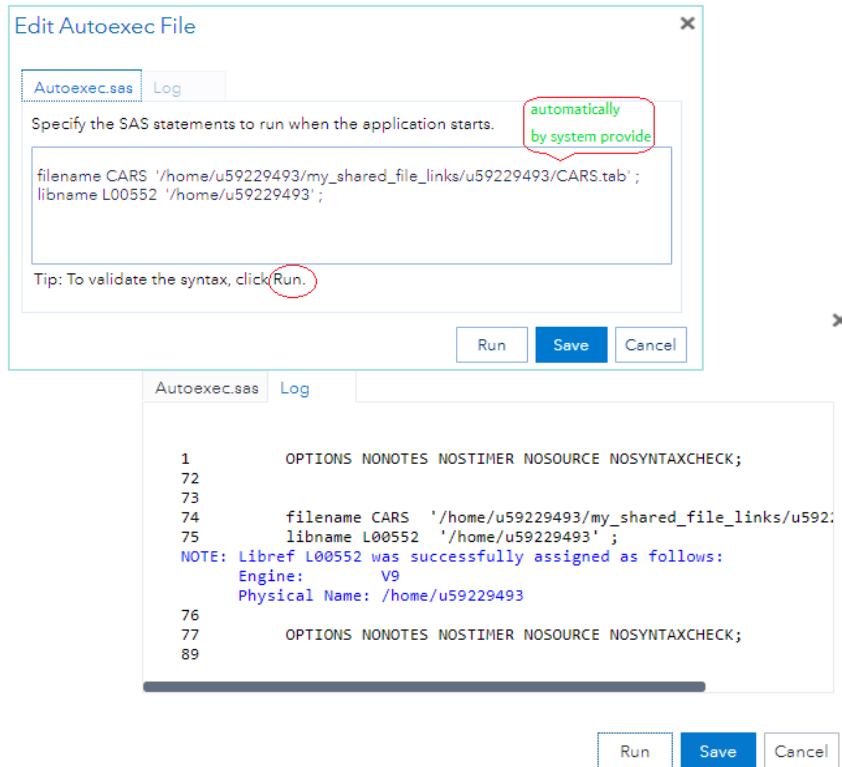
	Make	Model	Type
1	Acura	MDX	SUV
2	Acura	RSX Type S 2dr	Sedan
3	Acura	TSX 4dr	Sedan
4	Acura	TL 4dr	Sedan
5	Acura	3.5 RL 4dr	Sedan
6	Acura	3.5 RL w/Navigation 4dr	Sedan
7	Acura	NSX coupe 2dr manual S	Sports
8	Audi	A4 1.8T 4dr	Sedan
9	Audi	A41.8T convertible 2dr	Sedan
10	Audi	A4 3.0 4dr	Sedan
11	Audi	A4 3.0 Quattro 4dr manual	Sedan

- e. The SAS Studio user can toggle between both of the datasets created by this code with the Table drop-down list. This tab also shows the columns or variables within the tables and shows the user what the rows look like within the datasets

The AUTOEXEC file

- There is a special SAS file called autoexec.sas. This file allows an individual user or a server administrator to put any type of SAS executable statement in it, and whenever the individual starts a SAS session or, for a server, whenever the server is started, then the code in this file executes automatically.
- On a server, this makes it easy to set up consistent libraries that are named the same for multiple users, which saves time and resources because there is only one connection/pointer needed for a particular data source, instead of multiple libraries from multiple users making multiple connections to a single data source.
- Likewise, at the individual user level, this empowers the user to set up libraries, create data, create user-defined formats, and/or create SAS macros that they themselves use all the time.

- d. SAS Studio provides the user with access to their own autoexec file through the More application options icon located in the top right-hand corner to the left of the Help question mark icon. Select the icon for More application options and select Edit Autoexec File.
- e. There are two tabs Autoexec.sas and Log.
- f. The Autoexec.sas tab is simply a special code pane, and the Log allows you to debug any code that is put in the Autoexec.sas pane and Run.



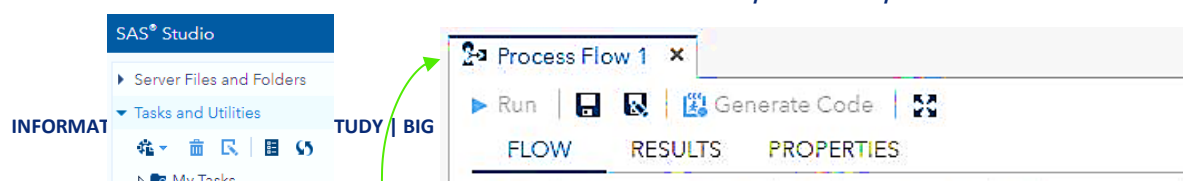
- g. Save your program to **IS429 Week#5 A NIM yourName**.

3. Visual Programmer versus SAS Programmer

- ▶ In our IS429 BDA practicum class, the reader will use the SAS Programmer perspective.
- ▶ However, SAS Studio also provides a Visual Programmer perspective, which is designed for users who would prefer to work with process flows in a project-based environment. In the upper right-hand corner of SAS Studio, right-click on SAS Programmer and then switch to the Visual Programmer perspective:

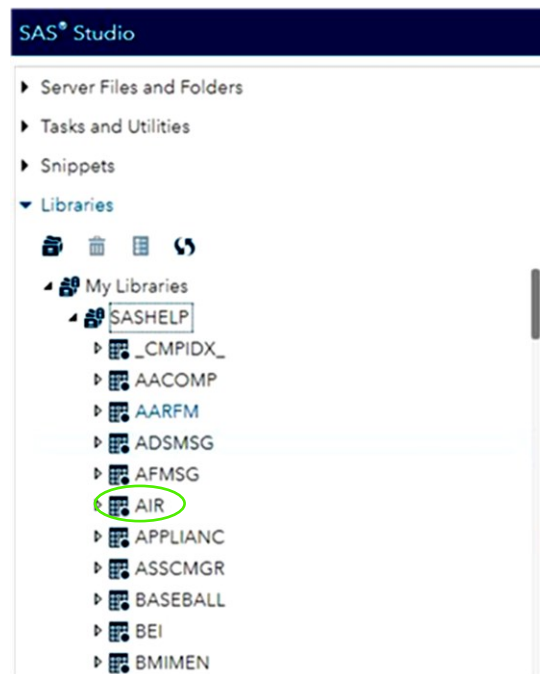


- ▶ Notice how the right-hand side of SAS Studio changes from a Program 1 section with CODE, LOG, and RESULTS tabs to a Process Flow 1 section with FLOW, RESULTS, and PROPERTIES tabs:

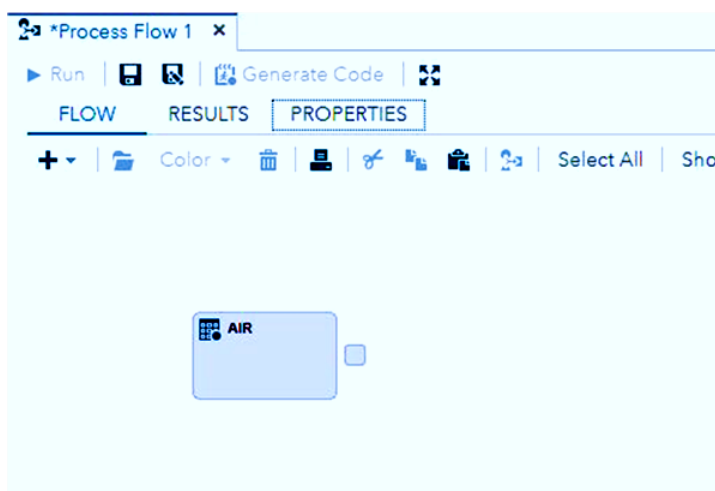


4. SAS Studio Auto-Generated Task Code

- ▶ When you are already in the Visual Programmer perspective work area, the reader can now build a process flow using drag and drop functionality.
- a. On the left-hand side of SAS Studio, select/expand Libraries and select/expand SASHELP, which will provide the user with a list of default datasets that come with SAS:

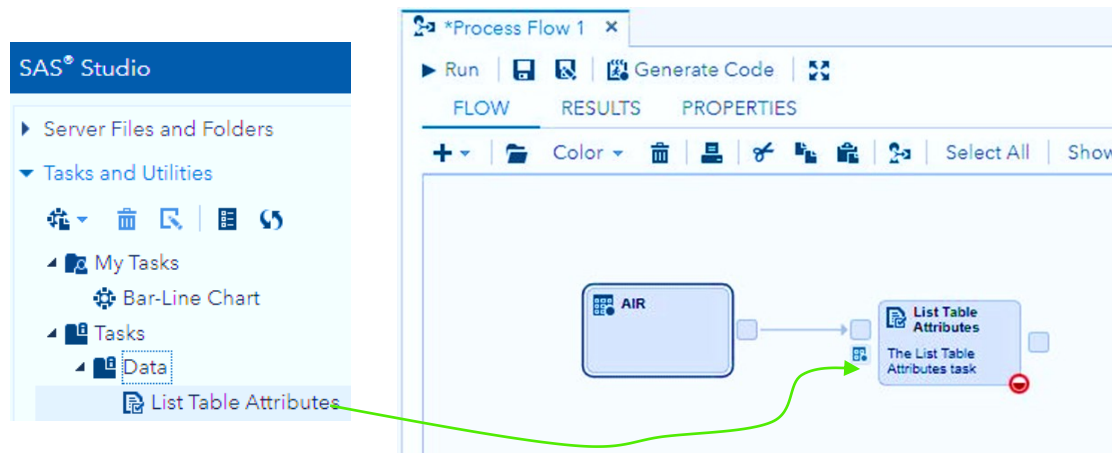


- b. Left-click and hold down on the AIR dataset and drag this over to the Process Flow 1 section and drop it :

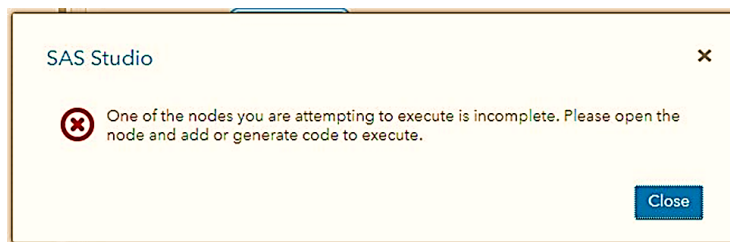


Notice a block representing the AIR dataset is created on the process flow. **This is not a copy of the AIR dataset**, but simply a **graphical representation of the table** within the process flow that the reader is building.

- c. To continue building a simple process flow, the user should now select/expand Tasks and Utilities on the left-hand side of SAS Studio and select/expand Tasks.
- d. As the user did for dragging and dropping the SASHELP. AIR dataset onto the process flow, drag and **drop the List Table Attributes task** to the right of the AIR table and then connect the two icons by left-clicking and holding on the control point, the little square, on the right side of the AIR icon, and drag the arrow over to connect it to the control point on the left side of the List Table Attributes icon:

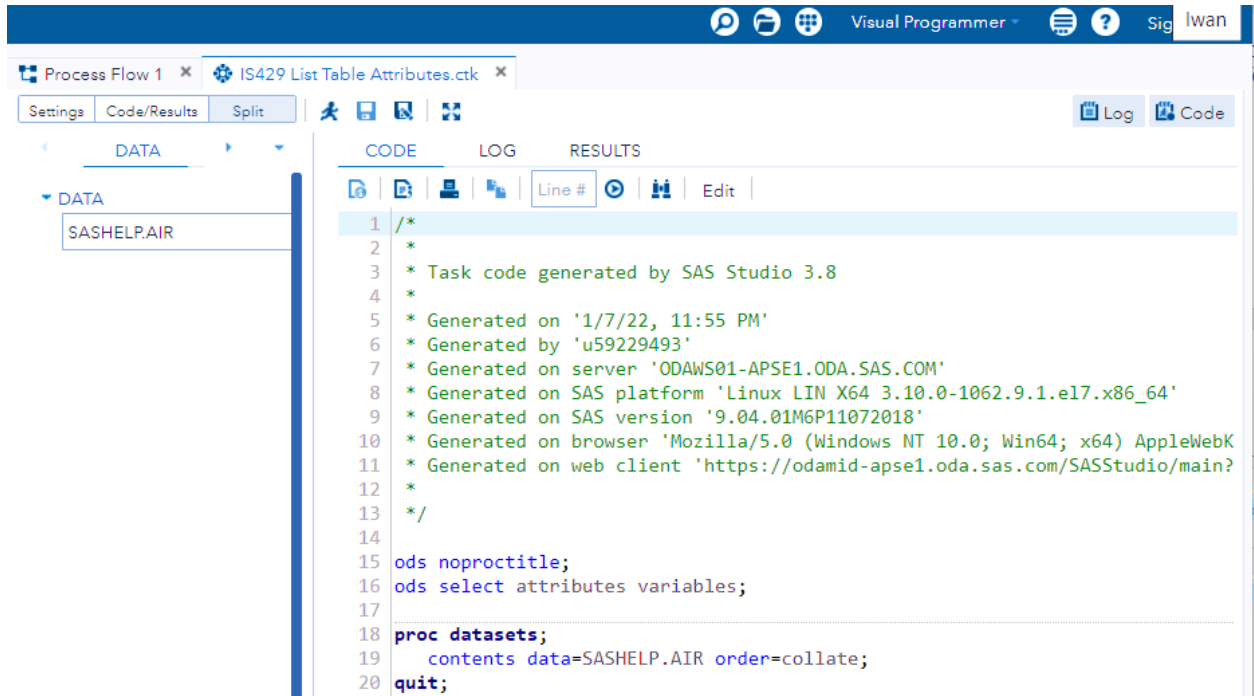


- e. The Run icon of the Process Flow 1 section should have ungrayed now that a table has been connected to a task; however, you will **notice** the task has a **red half-filled circle icon in the lower right-hand corner**.
- f. The red circle means that the task's properties have not yet been generated or manually added.
- g. Select the Run icon and the user should see the following popup window:



- h. **To resolve this situation, select Close and then right-click the List Table Attributes icon and select Open.**

- i. Once the user does this, **SAS Studio generates the necessary code for this task** and the reader should see the following window:



The screenshot shows the SAS Studio Visual Programmer interface. The left pane displays the 'DATA' tab with a tree view containing 'SASHELP.AIR'. The main pane shows the 'CODE' tab with the following SAS code:

```

1 /*
2 *
3 * Task code generated by SAS Studio 3.8
4 *
5 * Generated on '1/7/22, 11:55 PM'
6 * Generated by 'u59229493'
7 * Generated on server 'ODAWS01-APSE1.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
11 * Generated on web client 'https://odamid-apse1.oda.sas.com/SASStudio/main?'
12 *
13 */
14
15 ods noproctitle;
16 ods select attributes variables;
17
18 proc datasets;
19     contents data=SASHELP.AIR order=collate;
20 quit;
  
```

- j. Select Process Flow 1 in the upper left-corner to return to the process flow and now select Run.
 k. Get back again into Program tab and select Run
 l. Select the LOG tab, and it will provide the status and time of each executable step in the flow:

NOTE: PROCEDURE DATASETS used (Total process time):

real time	0.02 seconds
user cpu time	0.03 seconds
system cpu time	0.00 seconds
memory	1695.31k
OS Memory	26020.00k
Timestamp	01/07/2022 05:44:51 PM
Step Count	42
Page Faults	0
Page Reclaims	217
Page Swaps	0
Voluntary Context Switches	2
Involuntary Context Switches	0
Block Input Operations	0
Block Output Operations	24

- m. Select the RESULTS tab, and it will provide the table of contents and List Table Attributes:

▼ Table of Contents

▢ The Datasets Procedure

▢ SASHELP.AIR

▢ Attributes

▢ Variables


Data Set Name	SASHELP.AIR	Observations	144
Member Type	DATA	Variables	2
Engine	V9	Indexes	0
Created	10/25/2018 09:15:49	Observation Length	16
Last Modified	10/25/2018 09:15:49	Deleted Observations	0
Protection		Compressed	NO
Data Set Type		Sorted	NO
Label	airline data (monthly: JAN49-DEC80)		
Data Representation	SOLARIS_X86_64, LINUX_X86_64, ALPHA_TRU64, LINUX_IA64		
Encoding	us-ascii ASCII (ANSI)		

Alphabetic List of Variables and Attributes					
#	Variable	Type	Len	Format	Label
2	AIR	Num	8		international airline travel (thousands)
1	DATE	Num	8	MONYY.	

- n. When the List Table Attributes task opens this time, instead of the CODE tab being active and seeing the SAS Studio auto-generated code, the RESULTS tab is active and displays the output from the code that ran successfully when the user ran the process flow.
- o. Save your process flow to **IS429 Week#5 B NIM yourName**


5. What's in the SAS® OnDemand for Academics Edition?

- a. Switch SAS Studio back to the SAS Programmer perspective using the drop-down list in the upper right-hand corner
- b. Since SAS is an analytic with many modules and solutions, let's see which pieces of the SAS environment are included in the SAS® OnDemand for Academics Edition.
- c. Return to the Program 1 section and clear all the code using the Xx icon on the CODE tab toolbar. Type the following code in the CODE tab:


SAS Programmer ▼

*Program 1 x

CODE
LOG
RESULTS

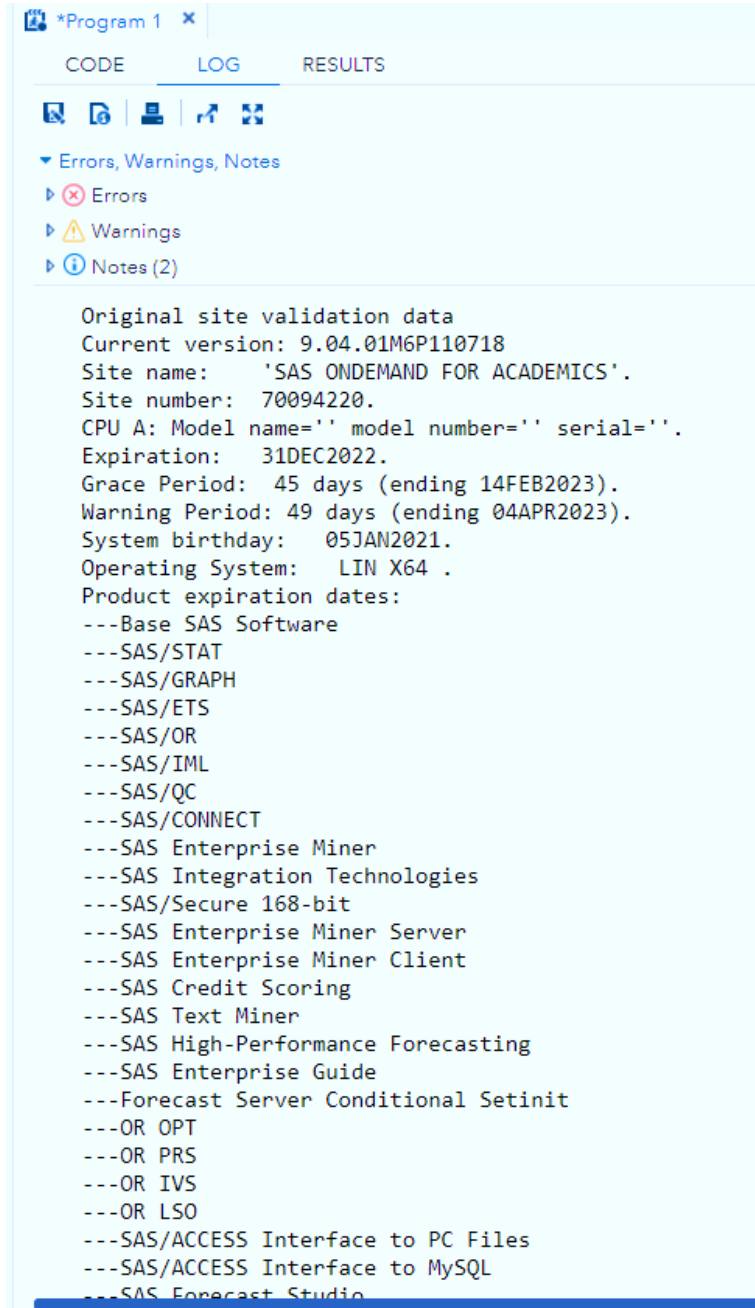


```

1  /* This lists the SAS foundation products that are installed on your system */
2  /* along with their version numbers.                                     */
3  proc product_status;
4  run;
5
6  /* This lists the SAS products that have been licensed and the date through */
7  /* which the licensed software is valid.                                     */
8  proc setinit;
9  run;
10

```

- d. Click and highlight only the proc product_status; run; part of the code and submit this by selecting the running man icon from the CODE tab toolbar.



The screenshot shows the SAS Studio interface with the LOG tab selected. The log contains the following text:

```


Original site validation data
Current version: 9.04.01M6P110718
Site name: 'SAS ONDEMAND FOR ACADEMICS'.
Site number: 70094220.
CPU A: Model name='' model number='' serial=''.
Expiration: 31DEC2022.
Grace Period: 45 days (ending 14FEB2023).
Warning Period: 49 days (ending 04APR2023).
System birthday: 05JAN2021.
Operating System: LIN X64 .
Product expiration dates:
---Base SAS Software
---SAS/STAT
---SAS/GRAPH
---SAS/ETS
---SAS/OR
---SAS/IML
---SAS/QC
---SAS/CONNECT
---SAS Enterprise Miner
---SAS Integration Technologies
---SAS/Secure 168-bit
---SAS Enterprise Miner Server
---SAS Enterprise Miner Client
---SAS Credit Scoring
---SAS Text Miner
---SAS High-Performance Forecasting
---SAS Enterprise Guide
---Forecast Server Conditional Setinit
---OR OPT
---OR PRS
---OR IVS
---OR LSO
---SAS/ACCESS Interface to PC Files
---SAS/ACCESS Interface to MySQL
---SAS Forecast Studio

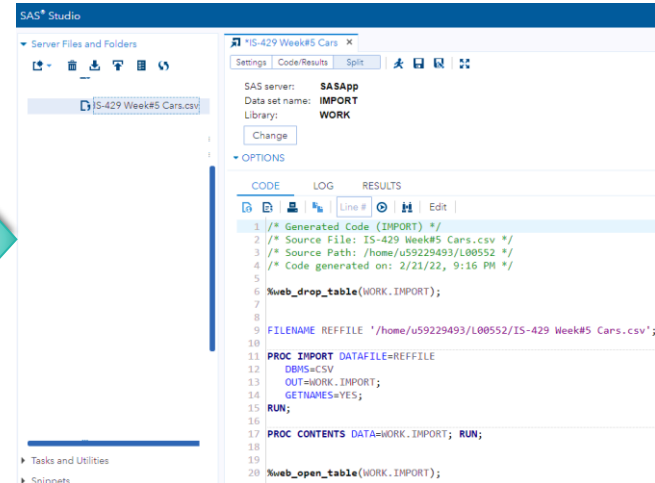
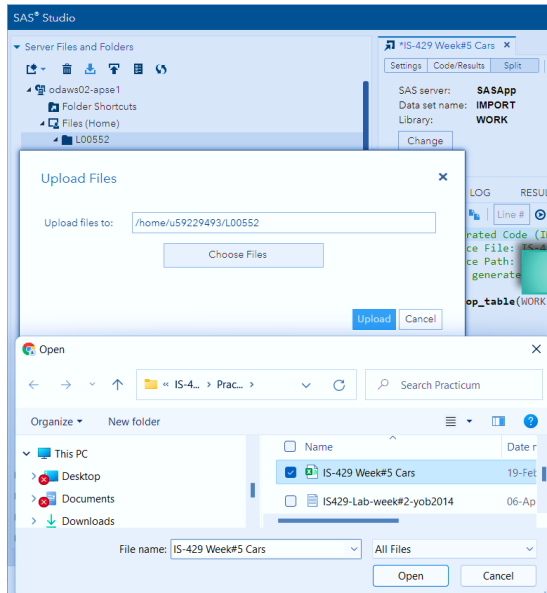
```

- e. As the results show, the SAS® OnDemand for Academics Edition some contains custom versions of BASE SAS, SAS/STAT® software, SAS/ETS® software, SAS/IML® software, High Performance Suite, and SAS/ACCESS® Interface to PC files based on the SAS 9.4 release.

6. Uploading, Accessing Local Data and Visualized

a. Uploading Local Data

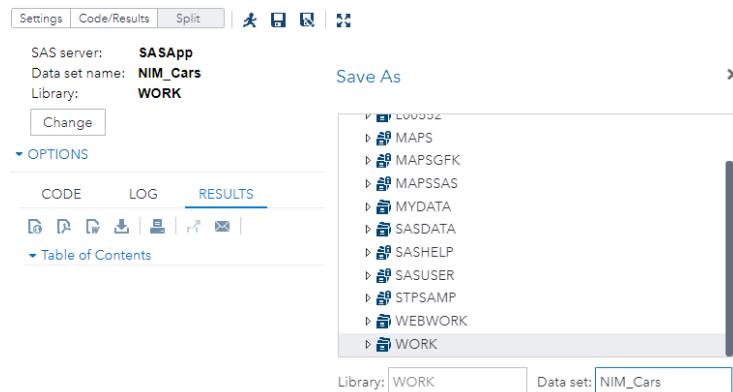
- 1) Back again to SAS® Studio and return to the SAS Programmer's worksheet.
- 2) In SAS Studio, select the folder that you want to upload data to. Expand Files (Home) and then select an existing of your sasuser folder.
- 3) Select the folder that you want to upload data to. Click  or right click, then upload your cars.csv file.



- 4) The results of the upload will bring up the cars.csv file in the folder you selected.
- 5) This SAS software that we use will create the program for us: as you can see on the bottom right of the "option" program code that has been generated by the system automatically.
- 6) Now, take a look at the contents of your uploaded csv file, what happens? is it a SASHDAT table file?
- 7) Certainly not: because the uploaded cars file is still in its original data form, namely text csv format.

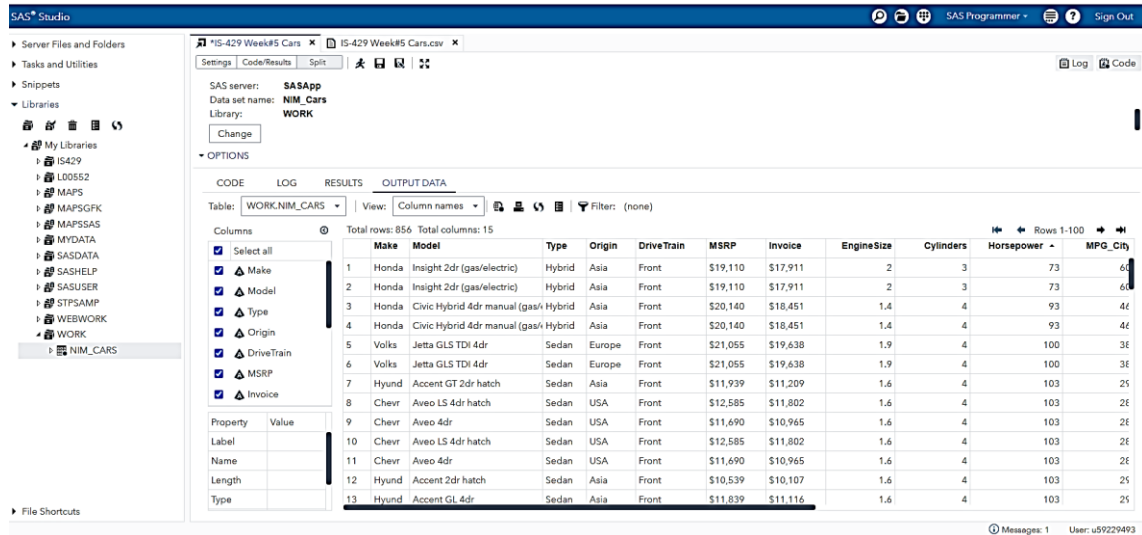
b. Importing your Data into SASHDAT

- 1) Import Data to SASHDAT
- 2) Now you import the cars.csv into the SASHDAT table in SAS CAS, how to right click on the cars.csv file and select import, name your car table to NIM_Cars.



NIM_Cars table perched on your WORK library.

- 3) Now you have a SASHDAT



SAS Studio interface showing a data table with 13 rows and 11 columns. The columns are Make, Model, Type, Origin, DriveTrain, MSRP, Invoice, EngineSize, Cylinders, Horsepower, and MPG_City. The data includes various car models like Honda Insight, Honda Civic Hybrid, Volkswagen Jetta, and Chevrolet Aveo.

	Make	Model	Type	Origin	DriveTrain	MSRP	Invoice	EngineSize	Cylinders	Horsepower	MPG_City
1	Honda	Insight 2dr (gas/electric)	Hybrid	Asia	Front	\$19,110	\$17,911	2	3	73	60
2	Honda	Insight 2dr (gas/electric)	Hybrid	Asia	Front	\$19,110	\$17,911	2	3	73	60
3	Honda	Civic Hybrid 4dr manual (gas/electric)	Hybrid	Asia	Front	\$20,140	\$18,451	1.4	4	93	46
4	Honda	Civic Hybrid 4dr manual (gas/electric)	Hybrid	Asia	Front	\$20,140	\$18,451	1.4	4	93	46
5	Volksw	Jetta GLS TDI 4dr	Sedan	Europe	Front	\$21,055	\$19,638	1.9	4	100	36
6	Volksw	Jetta GLS TDI 4dr	Sedan	Europe	Front	\$21,055	\$19,638	1.9	4	100	36
7	Hyund	Accent GT 2dr hatch	Sedan	Asia	Front	\$11,939	\$11,209	1.6	4	103	25
8	Chevr	Aveo LS 4dr hatch	Sedan	USA	Front	\$12,585	\$11,802	1.6	4	103	26
9	Chevr	Aveo 4dr	Sedan	USA	Front	\$11,690	\$10,965	1.6	4	103	26
10	Chevr	Aveo LS 4dr hatch	Sedan	USA	Front	\$12,585	\$11,802	1.6	4	103	26
11	Chevr	Aveo 4dr	Sedan	USA	Front	\$11,690	\$10,965	1.6	4	103	26
12	Hyund	Accent 2dr hatch	Sedan	Asia	Front	\$10,539	\$10,107	1.6	4	103	25
13	Hyund	Accent GL 4dr	Sedan	Asia	Front	\$11,039	\$11,116	1.6	4	103	25

c. Accessing and Visualized your Data

Next, you create a simple graph in the form of a barchart as shown in the output image C.

1) Do you have to code the program? Of course not, the SAS system will create it for you and just click-drag through the **"bar-line chart"** option on the **Task** with the following format menu:

- ▶ Data = WORK_NIM_CARS
- ▶ Category=Type, Sub-Category=Origin
- ▶ Bar Variable=MPG_City, Statistics=Mean
- ▶ Line Variable=MPG_Highway, Statistics=Mean
- ▶ For coloring and texting creations, you can use the Appearance menu.

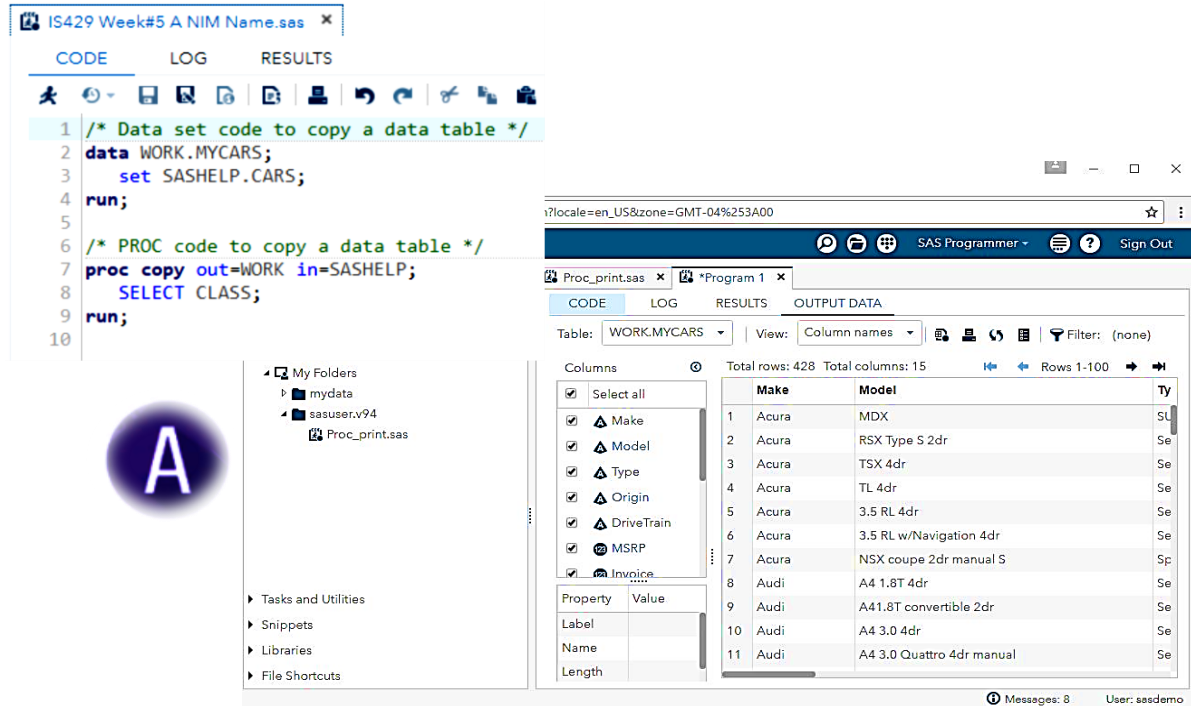
2) If the result is finished as a C output image, save this visualization program with the naming format IS-429 Lab Week#5C Visualized NIM.sas

3) You can add other graphic forms or useful information content, please improvise.

🔄 Finally, today's practicum is over, collect the SAS programs and SAS process in IS-429 BDA Week#5 NIM yourName. zip format and submit them to e-Learning IS-429 BDA Week#5.

RESULTS/ OUTPUT

A. Creating a new SAS program



The screenshot displays the SAS Studio interface. The main window shows a SAS program with the following code:

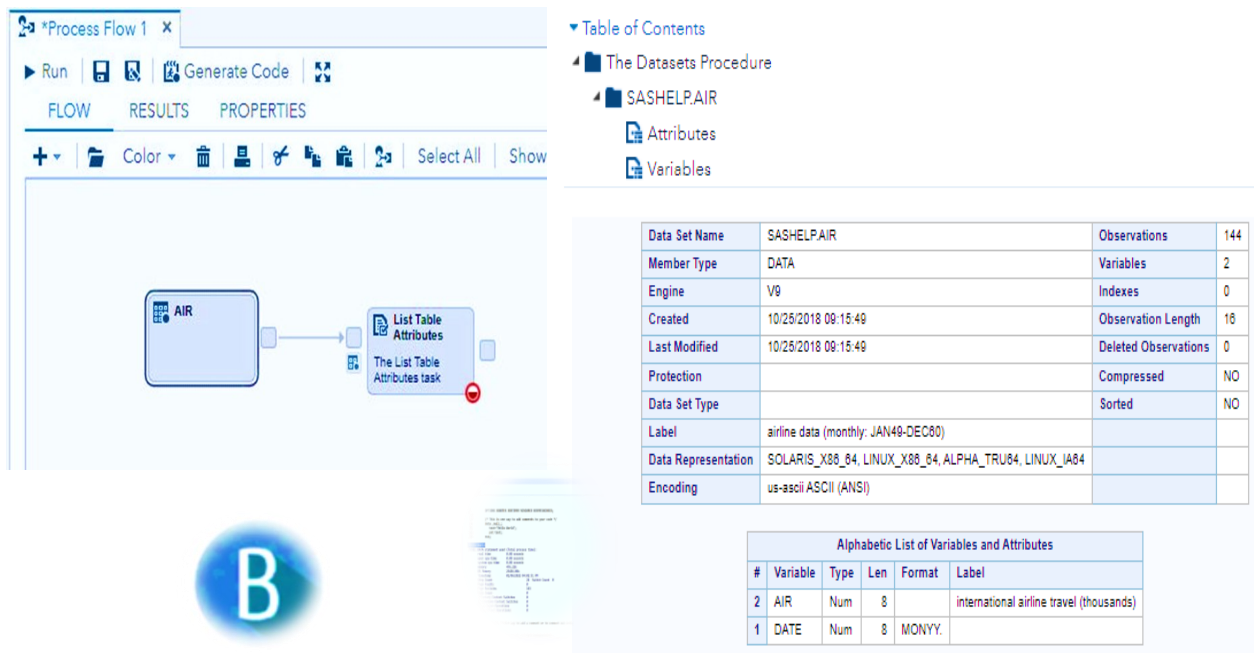
```

1 /* Data set code to copy a data table */
2 data WORK.MYCARS;
3   set SASHELP.CARS;
4 run;
5
6 /* PROC code to copy a data table */
7 proc copy out=WORK in=SASHELP;
8   SELECT CLASS;
9 run;
10

```

The interface includes a toolbar with various icons for file operations, a file explorer on the left showing the project structure, and a results/output window on the right. The results window shows the output of the PROC COPY statement, displaying a table with columns Make, Model, and Type, and rows 1 through 11.

B. SAS Studio Auto-Generated Task Code



The screenshot displays the SAS Studio interface for a task named 'AIR'. The task is connected to 'List Table Attributes'. The results window shows the output of the task, including a table of dataset properties and an alphabetic list of variables and attributes.

Table of Contents

- The Datasets Procedure
 - SASHELP.AIR
 - Attributes
 - Variables

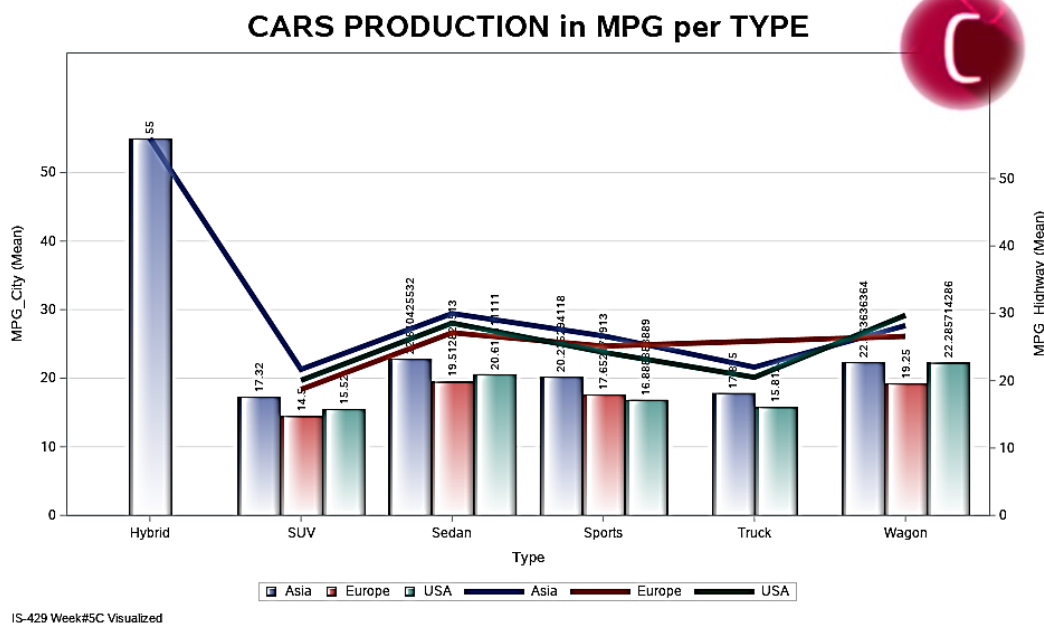
Dataset Properties Table:

Property	Value
Data Set Name	SASHELP.AIR
Member Type	DATA
Engine	V9
Created	10/26/2018 09:15:49
Last Modified	10/26/2018 09:15:49
Protection	
Data Set Type	Sorted
Label	airline data (monthly: JAN49-DEC80)
Data Representation	SOLARIS_X86_64, LINUX_X86_64, ALPHA_TRU64, LINUX_IA64
Encoding	us-ascii ASCII (ANSI)
Observations	144
Variables	2
Indexes	0
Observation Length	16
Deleted Observations	0
Compressed	NO

Alphabetic List of Variables and Attributes

#	Variable	Type	Len	Format	Label
2	AIR	Num	8		international airline travel (thousands)
1	DATE	Num	8	MONYY.	

C. Accessing and Visualized your Data



The
End

REFERENCE

1. David Pope. 2017. Big Data Analytics with SAS. Packt Publishing Ltd. Birmingham, UK.
2. [SAS® Support | Documentation](#)
3. Other additional references are excerpts from various Online Learning/websites.