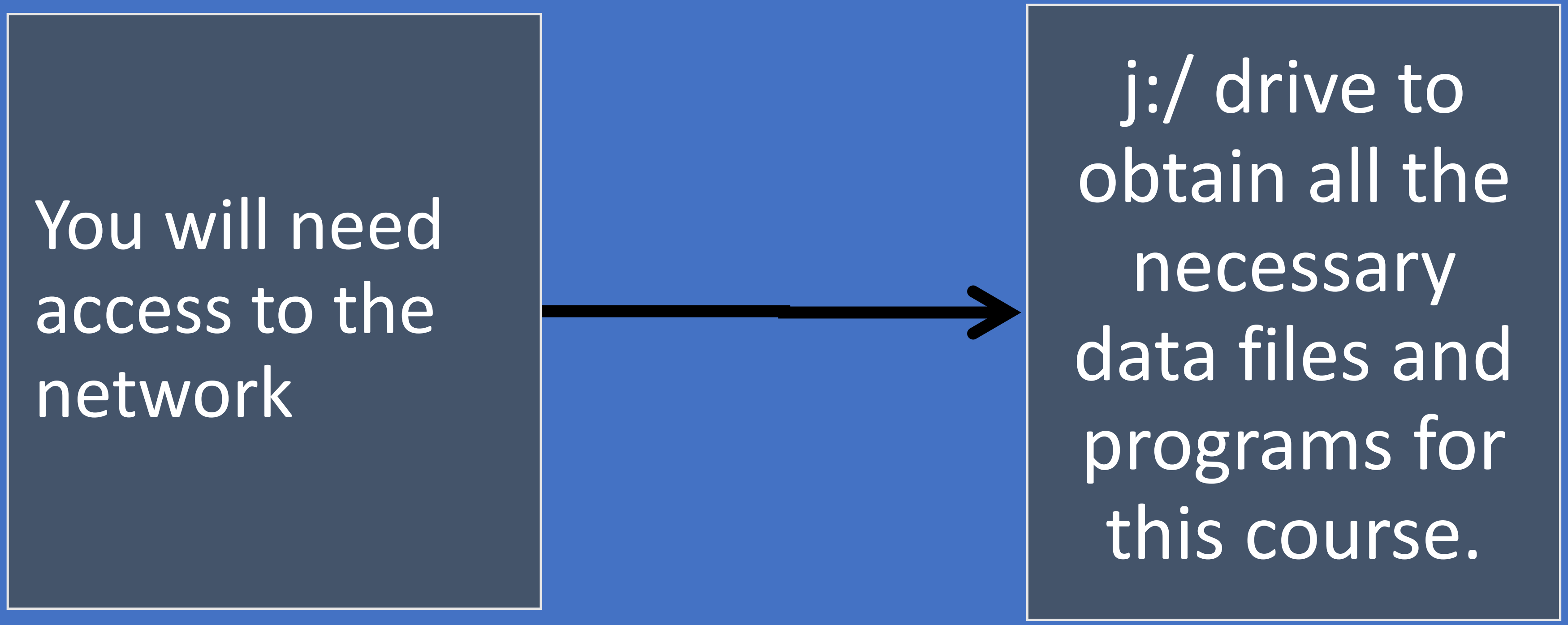
STAT 425 and STAT 625 Statistical Software

Lecture 1
Getting to Know SAS

Saida Zardi

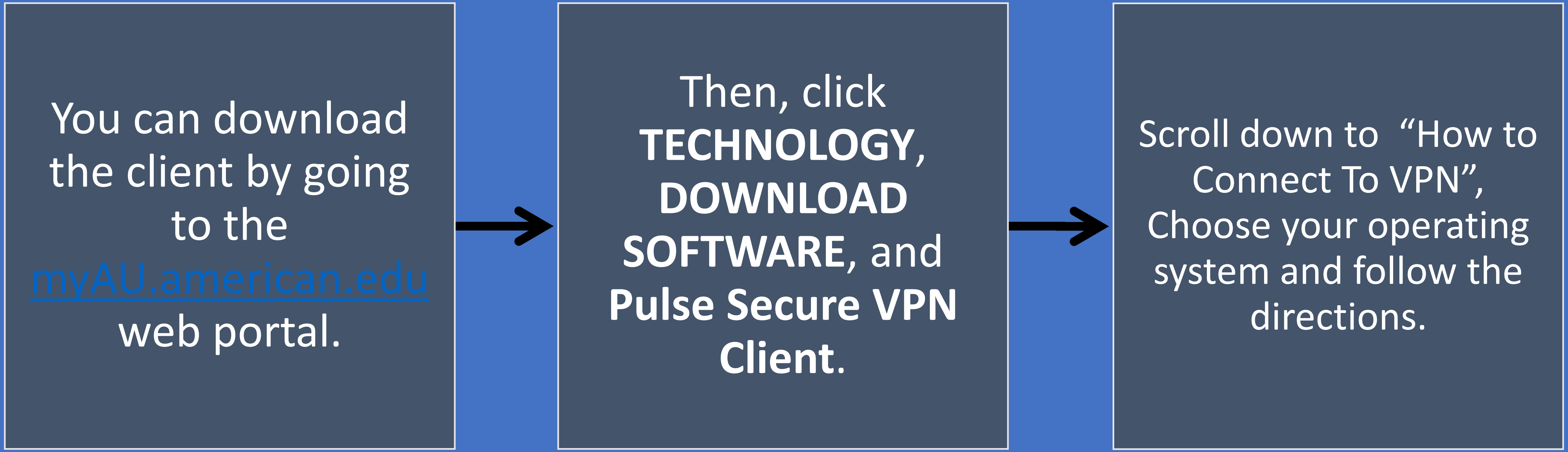
Network drive access from off campus



You will need access to the network

j:/ drive to obtain all the necessary data files and programs for this course.

Pulse Secure Client (VPN) access



You can download the client by going to the myAU.american.edu web portal.

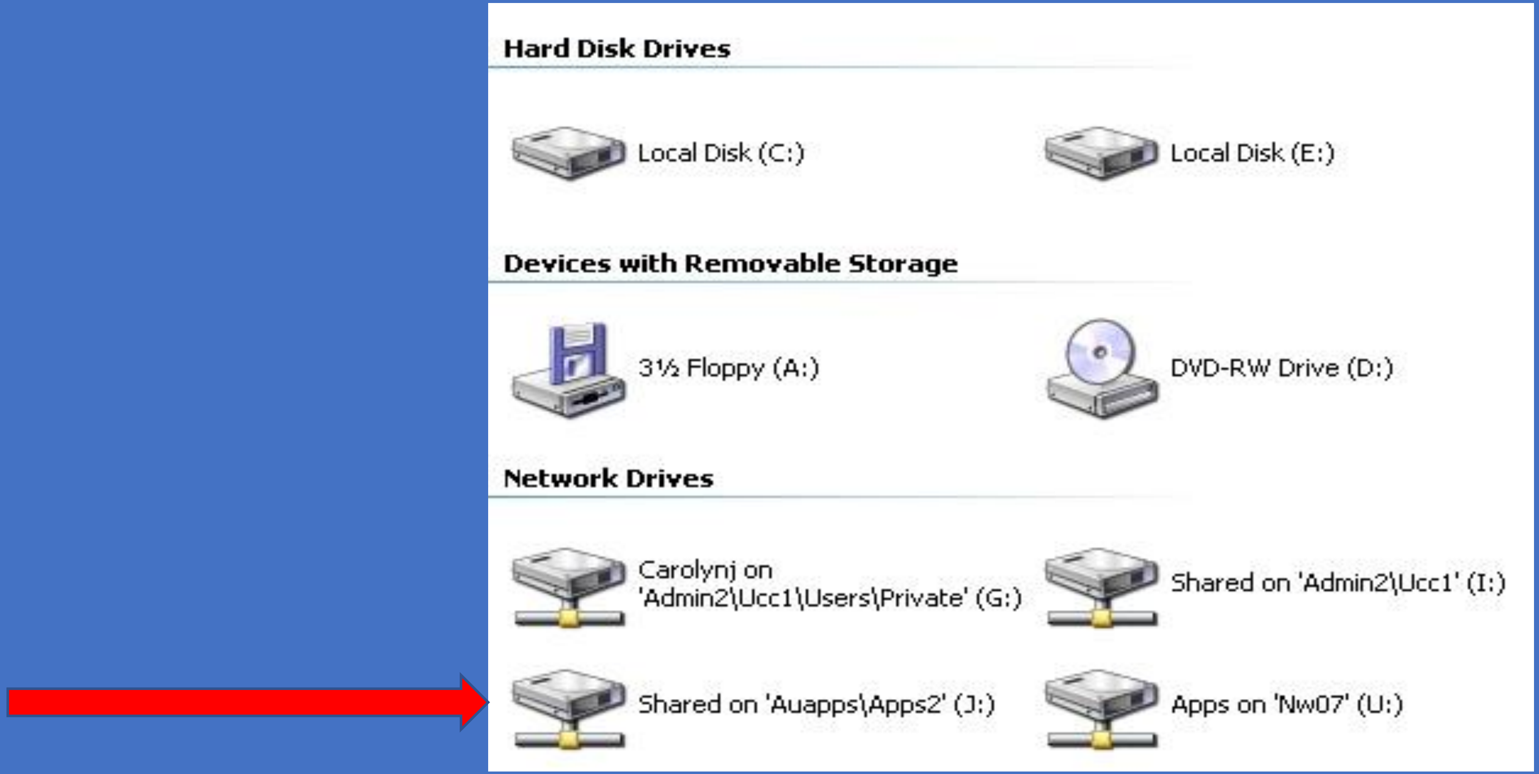
Then, click **TECHNOLOGY, DOWNLOAD SOFTWARE**, and **Pulse Secure VPN Client**.

Scroll down to “How to Connect To VPN”, Choose your operating system and follow the directions.

Note: If you are working from on-campus computers, the Novel client is already installed

Mapping your network drives

Now that you have Pulse Secure VPN access. Your computers will work as if you are on-campus and you will be able to access the J:/ drive:



Hard Disk Drives

Local Disk (C:)

Local Disk (E:)

Devices with Removable Storage

3½ Floppy (A:)

DVD-RW Drive (D:)

Network Drives

Carolynj on 'Admin2\Ucc1\Users\Private' (G:)

Shared on 'Admin2\Ucc1' (I:)

Shared on 'Auapps\Apps2' (J:)

Apps on 'Nw07' (U:)

Note: If you do not have access please contact AU Help desk at 202-885-2550

File types and access

- File types are differentiated by suffix:

  - * .sas – file is a SAS program
  - * .log – file is a SAS log
  - * .lst – file is a SAS list (output)
  - * .sas7bdat – file is a SAS data set

  - * .csv –is a comma separated value file
  - * .xls – file is an Excel spreadsheet

- Double-clicking a file usually opens the application

Access to SAS 9.4

You have free access to SAS 9.4

Please Log-on to

https://apps.american.edu

and gain access to SAS.

Review the Stat425-004/STAT 625-004 Syllabus and Announcements on Blackboard

Lecture 1:
Introduction to SAS

Learning Outcome

- Understanding the SAS Windowing environment
- Understanding the Basics of the SAS Language
- Understanding the SAS Libraries

What is SAS ?

- SAS stood originally for ***Statistical Analysis System***
- It began in the late 6os and in the 70s
- It evolved into a very powerful, general purpose programing language, widely used.
- It is a collection of modules that are used to analyze and process data

Recommended Books and Other SAS Documentation

- **Books**:
  - The Little SAS Book: A primer, 4th Edition. By L .Delwiche and S. Slaughter.SAS Publishing.
  - Learning SAS® by Example: A Programmer's Guide, Second Edition. By Ron Cod SAS Publishing
  - http://www.sas.com/apps/pubscat/bookdetails.jsp?catid=1&pc=60864
- **SAS Documentation**
  - SAS Press

    http://www.sas.com/apps/pubscat/booklist.jsp?attr=category&val=SAS+Press

Understanding your SAS Interface

- SAS Windowing environment

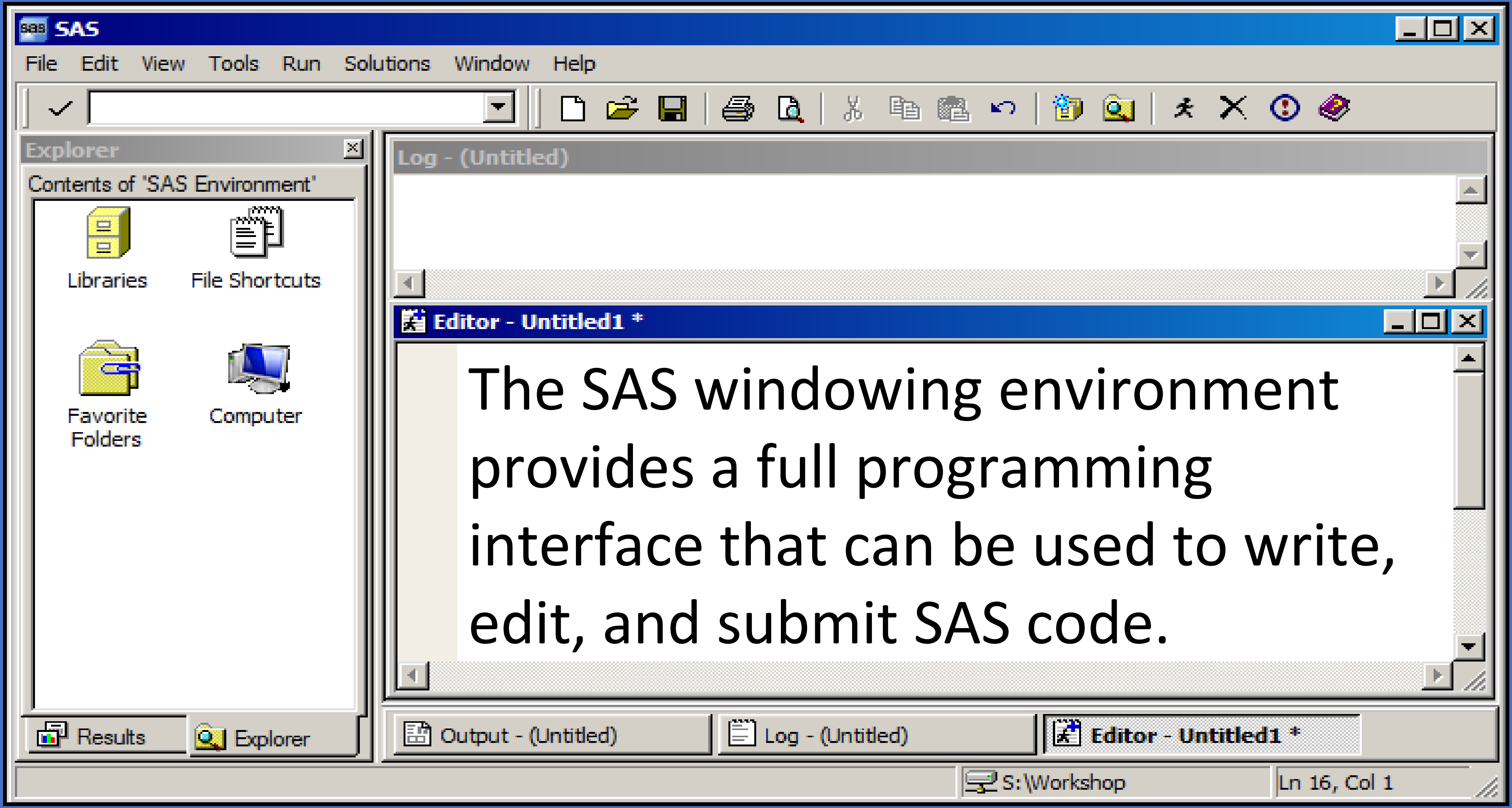
This is the interrface we will be using for this cource

- SAS Entreprise Guide

- SAS Studio

SAS Windowing Environment

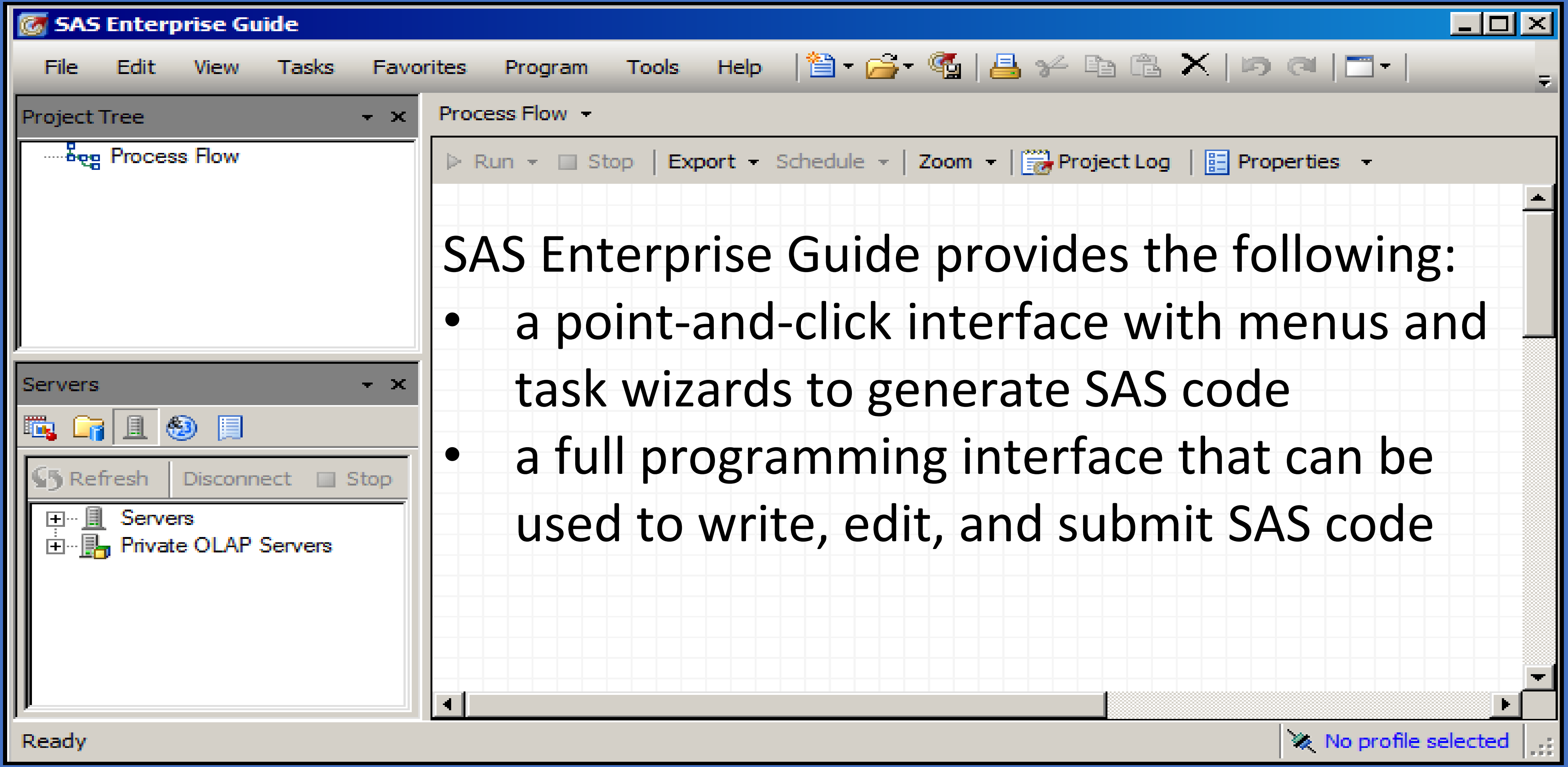
It’s an application that is accessed from different operating environments



The SAS windowing environment provides a full programming interface that can be used to write, edit, and submit SAS code.

SAS Entreprise Guide

It’s a client application that is accessed from the Windows operating environment

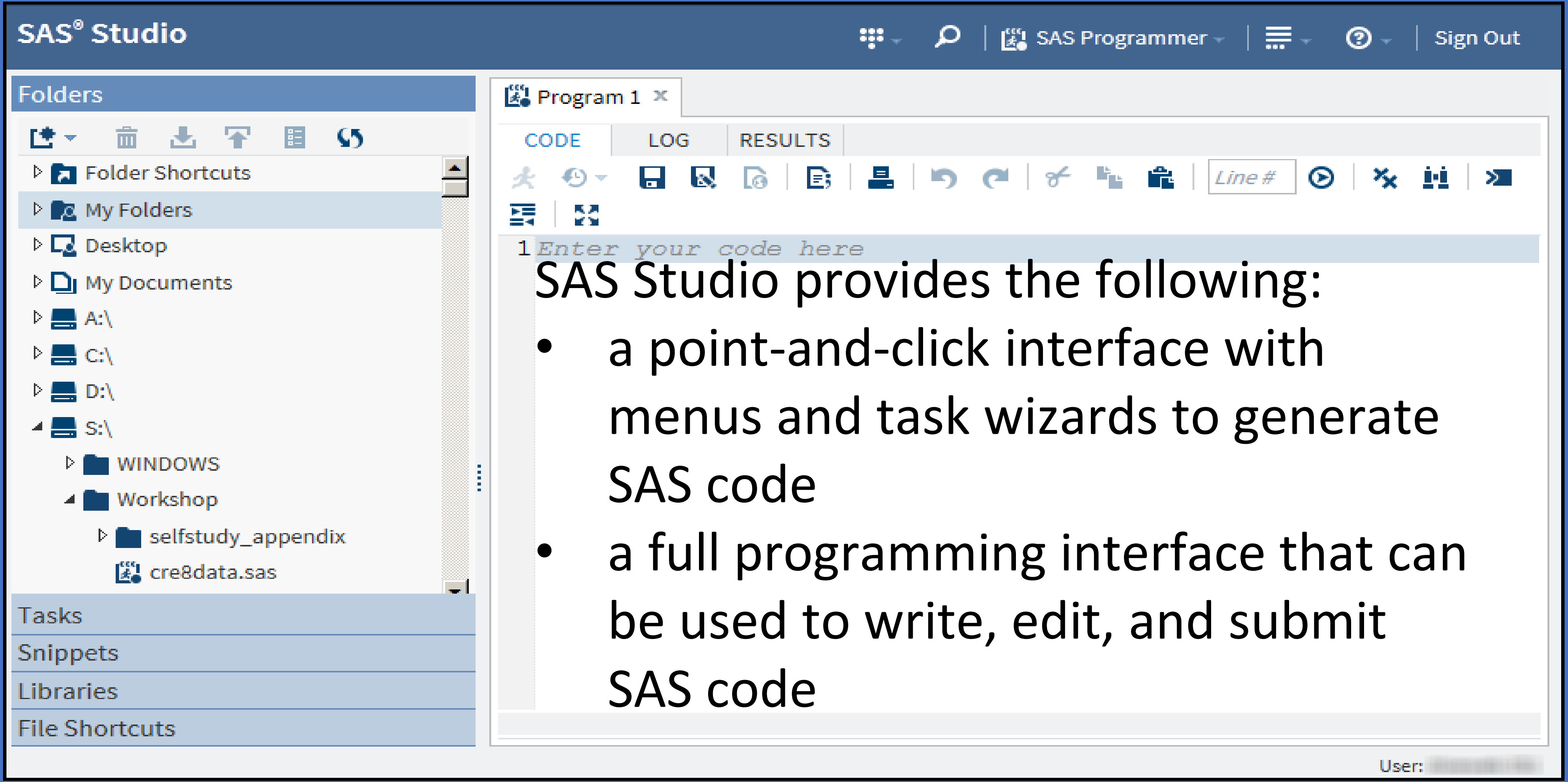


SAS Enterprise Guide provides the following:

- a point-and-click interface with menus and task wizards to generate SAS code
- a full programming interface that can be used to write, edit, and submit SAS code

SAS Studio

It’s a web client that is accessed through an HTML5-compliant web browser.

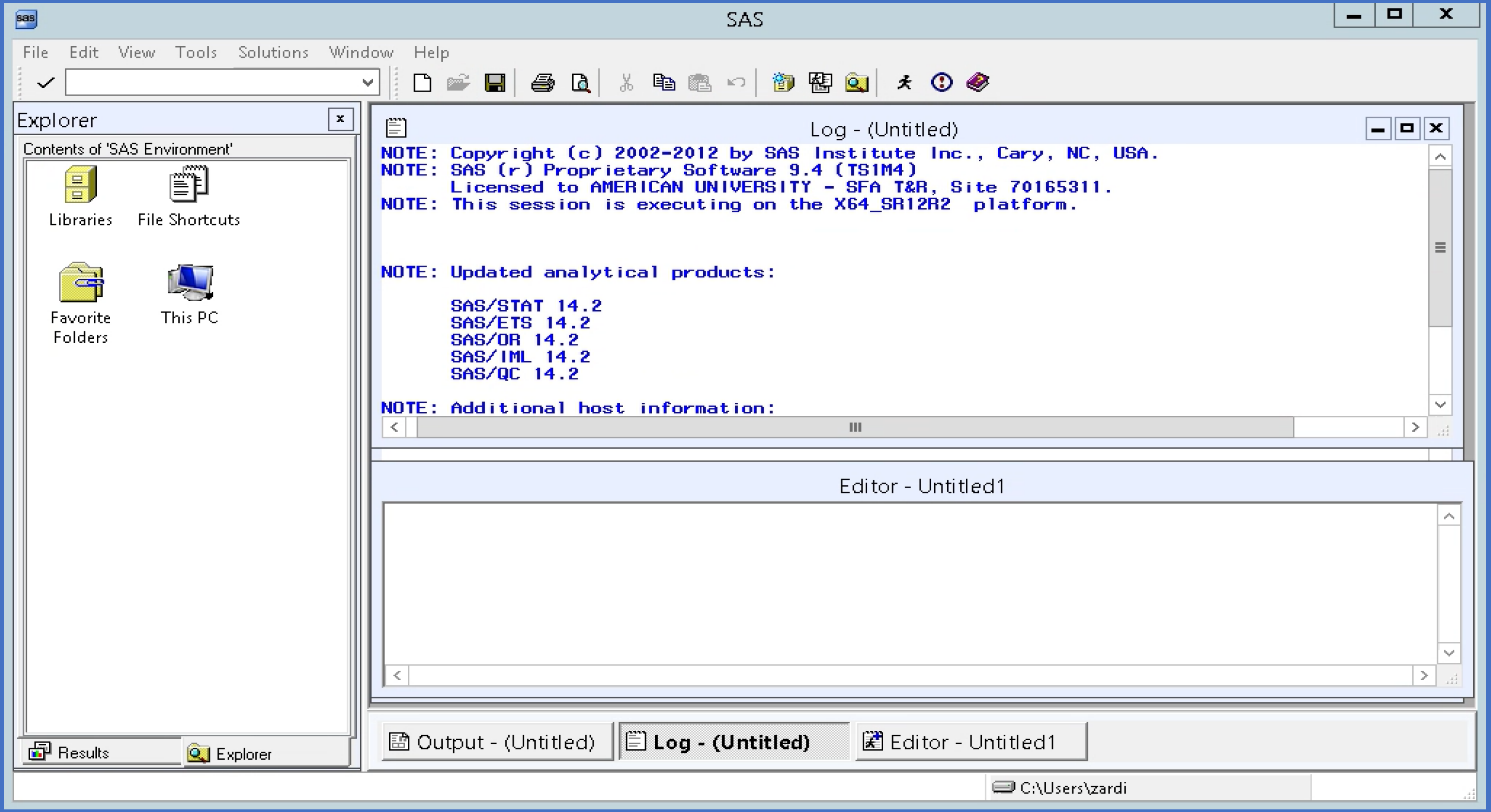


SAS Studio provides the following:

- a point-and-click interface with menus and task wizards to generate SAS code
- a full programming interface that can be used to write, edit, and submit SAS code

Opening SAS v9.4

When you start your SAS v9.4, you will find five SAS Windows:



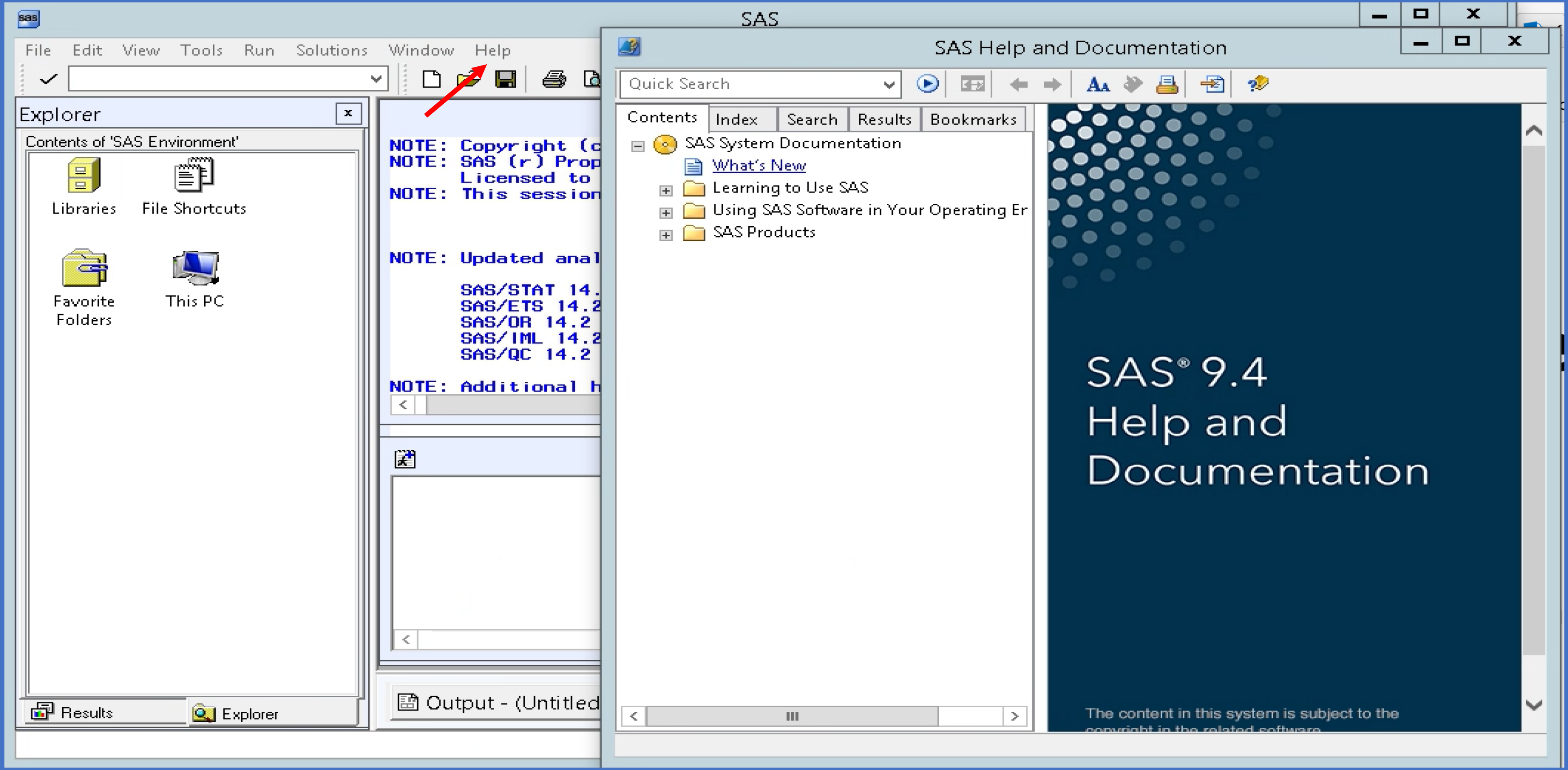
```
NOTE: Copyright (c) 2002-2012 by SAS Institute Inc., Cary, NC, USA.
NOTE: SAS (r) Proprietary Software 9.4 (TS1M4)
      Licensed to AMERICAN UNIVERSITY - SFA T&R, Site 70165311.
NOTE: This session is executing on the X64_SR12R2  platform.



NOTE: Updated analytical products:

      SAS/STAT 14.2
      SAS/ETS 14.2
      SAS/OR 14.2
      SAS/IML 14.2
      SAS/QC 14.2

NOTE: Additional host information:
```

- Editor: Enables you to enter, edit and submit SAS programs
- Log: Displays messages about your SAS programs
- Output: enables you to browse the output from your SAS program
- Results: helps you navigate and manage your SAS programs
- Explorer: enables you to view and manage your SAS files

More details about this later.

The Help Facility

Click on Help then SAS Help Documentation: This is an important resource and you need to accustom yourself with it.



```
NOTE: Copyright (c
NOTE: SAS (r) Prop
      Licensed to
NOTE: This session


NOTE: Updated anal

      SAS/STAT 14.
      SAS/ETS 14.2
      SAS/OR 14.2
      SAS/IML 14.2
      SAS/QC 14.2

NOTE: Additional h
```

The Basics of the SAS Language

SAS statement:

- It is used to perform a particular operation in a SAS program; or to provide information to a SAS program.
- They are free-format: meaning that they can begin and end anywhere on a line, that one statement can continue over several lines, and that several statements can be on the same line.
- Blank or special characters separate words in a SAS statement.
- You can specify SAS statements in uppercase or lowercase.
- In most situations, text that is enclosed in quotation marks is case sensitive.

SAS statements can be specified:

- in uppercase or lowercase.
- In most situations, text that is enclosed in quotation marks is case sensitive.
- Usually begins with an identifying keyword
- **always end with a *semicolon***.

There are two types of SAS statements:

1. statements that are used in DATA and PROC steps
2. statements that are global in scope and can be used anywhere in a SAS program

Global Statements

- Global statements are used anywhere in a SAS program
- They stay in effect until changed or canceled, or until the SAS session ends.
- Some common global statements are: TITLE, LIBNAME, OPTIONS, and FOOTNOTE.

DATA Step

- The DATA step creates or modifies data.
- For a DATA step, the input can be of several types: such as raw data or a SAS data set.
- The output from a DATA step can be of several types, such as a SAS data set or a report.
- A SAS data set is a data file that is formatted in a way that SAS can understand.

For example, DATA steps can do the following:

- compute values
- put the data into a SAS data set
- produce new SAS data sets by subsetting, supersetting, merging, and updating existing data sets
- check for and correct errors in your data

PROC Step

A PROC (short for procedure) step:

- analyzes data
- produces output or manages SAS files
- A SAS data set is usually the input for a PROC step
- The output can be of several types, for example it could be a report or an updated SAS data set.

PROC steps could be used to do the following:

- create a report that lists the data
- produce descriptive statistics
- create a summary report
- produce plots and charts

A simple example of a SAS Program

In this example of a SAS program, an existing SAS data set is used to create a new SAS data set containing a subset of the original data set.

Then, it prints a listing of the new data set using the procedure: PROC PRINT.

```
data sasuser.admit2;
    set sasuser.admit;
    where age>39;
run;
proc print data=sasuser.admit2;
run;
```

Data Step

Proc Step

This sample program displys in the **Editor** window. It contains a Data Step and a Proc Step.

- The DATA step starts with a DATA statement, which begins with the keyword ***DATA***
- The DATA step produces a new SAS data set.

Only the observations with an age value greater than 39 are written to the new SAS data set.

- The PROC step starts with a **PROC statement**, which begins with the keyword ***PROC***: it prints the new data

Structure of a SAS Program

Statements	Sample Program Code
DATA statement	data sasuser.admit2;
SET statement	Set sasuser.admit;
Other statements	Where age > 39;
RUN statement	Run;
PROC PRINT statement	proc print data=sasuser.admit2;
RUN statement	Run;

Note:

Between the steps of a SAS program, the RUN statement is not required.

Nonetheless, it’s a best practice to use a RUN statement, because it makes the SAS program easier to read and the LOG easier to understand when debugging.

It can make the SAS program easier to read and the SAS LOG easier to understand when debugging.

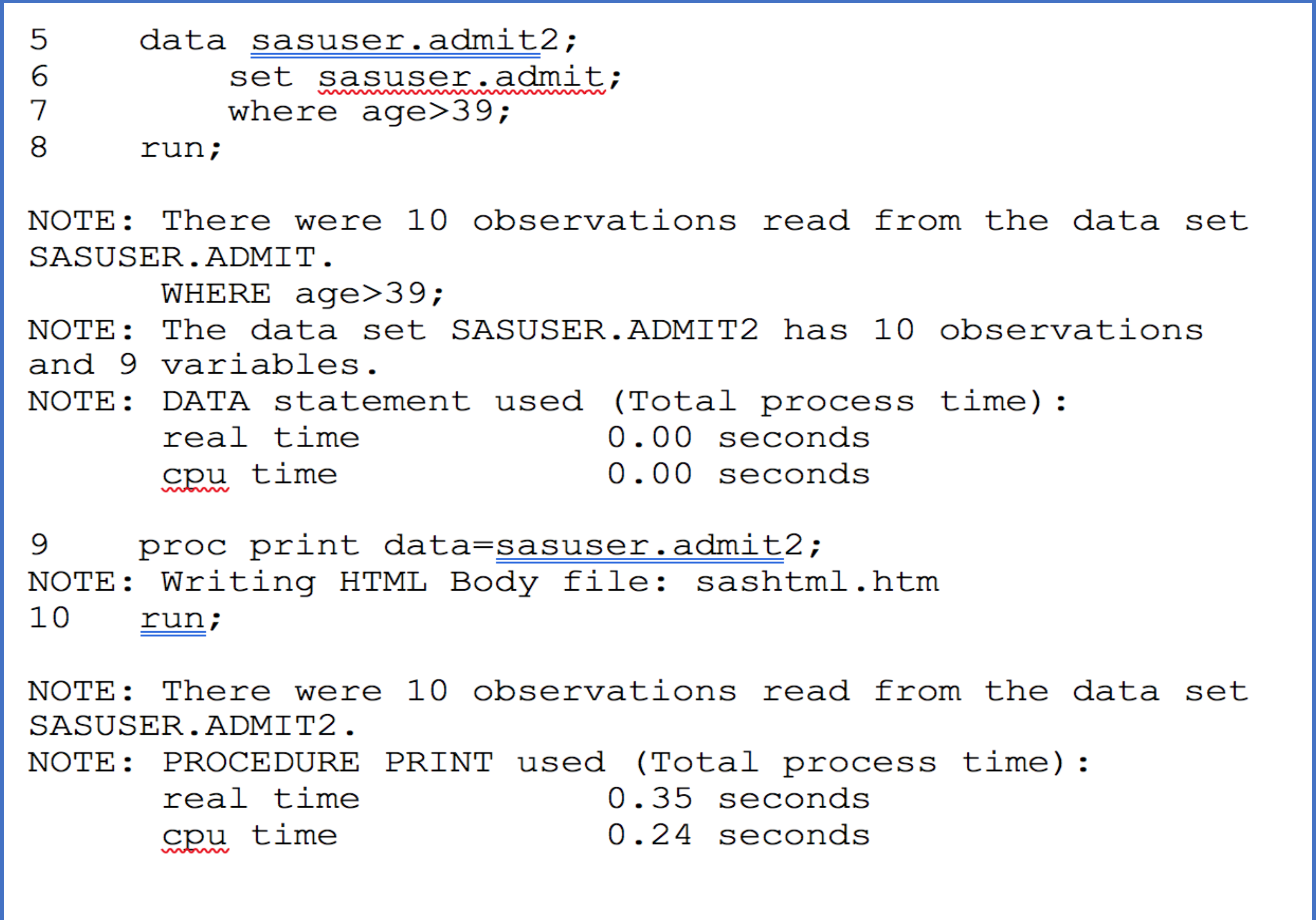
Log Messages

The Log messages appear in the **LOG** window.

The **SAS log** collects messages about the processing of SAS programs and about any errors that occur.

- Each time a step is executed, SAS generates a log of the processing activities and the results of the processing.
- you get separate sets of messages for each step in the program.

Log Messages of the Sample Program



```
5    data sasuser.admit2;
         set sasuser.admit;
7        where age>39;
8    run;

NOTE: There were 10 observations read from the data set
SASUSER.ADMIT.
      WHERE age>39;
NOTE: The data set SASUSER.ADMIT2 has 10 observations
and 9 variables.
NOTE: DATA statement used (Total process time):
      real time                0.00 seconds
      cpu time                 0.00 seconds

9    proc print data=sasuser.admit2;
NOTE: Writing HTML Body file: sashtml.htm
10   run;

NOTE: There were 10 observations read from the data set
SASUSER.ADMIT2.
NOTE: PROCEDURE PRINT used (Total process time):
      real time                0.35 seconds
      cpu time                 0.24 seconds
```

Results of Processing

The DATA Step

Suppose you submit the sample program below:

```
data sasuser.admit2;
   set sasuser.admit;
   where age>39;
run;
```

When the program is processed, it creates a new SAS data set (sasuser.admit2) containing only those observations with age values greater than 39.

The DATA step creates a new data set and produces messages in the SAS log, but it does not create a report or other output.

The PROC Step

If you add a PROC PRINT step to this same example, the program produces the same new data set as before, but it also creates the following report, which is displayed in HTML:

```
data sasuser.admit2;
    set sasuser.admit;
    where age>39;
run;
proc print data=sasuser.admit2;
run;
```

SAS Libraries

A SAS library contains one or more files that are defined, recognized, and accessible by SAS, and that are referenced and stored as a unit.

One special type of file is called a catalog.

In SAS libraries, catalogs function much like subfolders for grouping other members.

Predefined SAS Libraries

By default, SAS defines several libraries for you:

- **Sashelp**: a permanent library that contains sample data and other files that control how SAS works at your site. This is a Read-Only library.
- **Sasuser**: a permanent library that contains SAS files in the Profile catalog and that stores your personal settings. This is also a convenient place to store your own files.
- **Work**: a temporary library for files that do not need to be saved from session to session.

Additionnal Libraries

To define a library:

- Assign a library name to it and specify the location of the files, such as a directory path.
- Specify an engine, which is a set of internal instructions that SAS uses for writing to and reading from files in a library.
- You can define SAS libraries using programming statements.

How SAS Files Are Stored?

- A SAS library is the highest level of organization for information within SAS.

For instance, in the Windows environment, a library is typically a group of SAS files in the same folder or directory.

Permanent SAS Libraries

Permanent SAS libraries are available to you during subsequent SAS sessions.

To store files permanently in a SAS library:

- specify a library name other than the default library name Work.
- For example, by specifying the library name sasuser when you create a file, you specify that the file is to be stored in a permanent SAS library until you delete it.

Temporary SAS Libraries

Temporary SAS libraries last only for the current SAS session.

- If you do not specify a library name when you create a file, the file is stored in the temporary SAS library.
- At the end of the session, the temporary library and all of its files are deleted.
- if you specify the library name **Work**, then the files will be deleted at the end of the session.