ANLT5020 – Unit 2 Assignment 1 Tutorial

SAS Studio

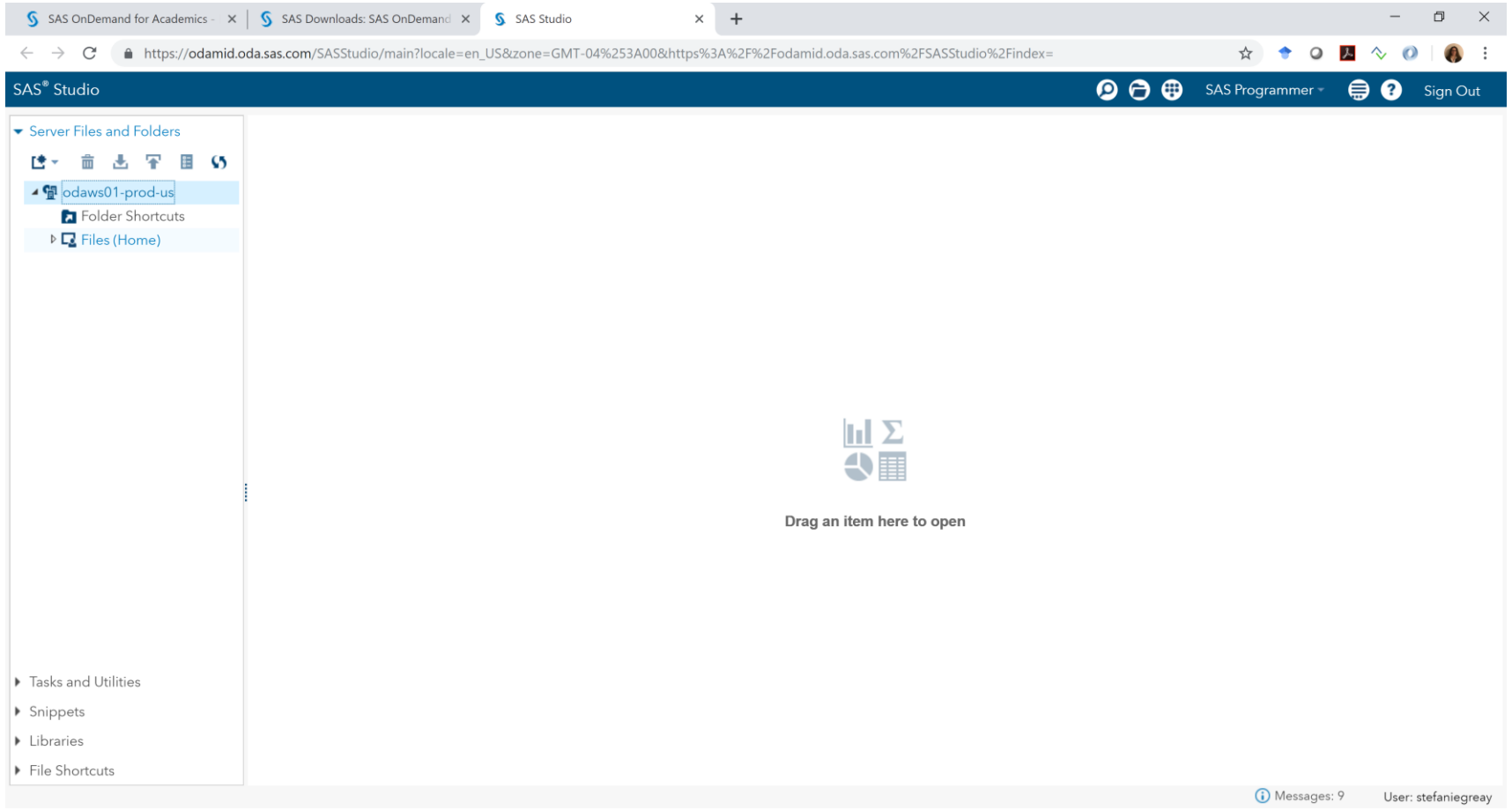
Instructions

- Get your Patient.xlsx data into SAS using an appropriate procedure.
- Record the code used to accomplish this.
- Use PROC CONTENTS to describe the variables in the dataset.
- Run SAS procedures to find errors and/or missing data in numerical (PROC UNIVARIATE) and character variables (PROC FREQ).
- Identify obvious errors and missing data using the output from SAS procedures.
- Use PROC MEANS to calculate before and after heart rate averages.
- Research and summarize several different methods analysts may use to resolve problems of missing data when it poses a threat to quality.

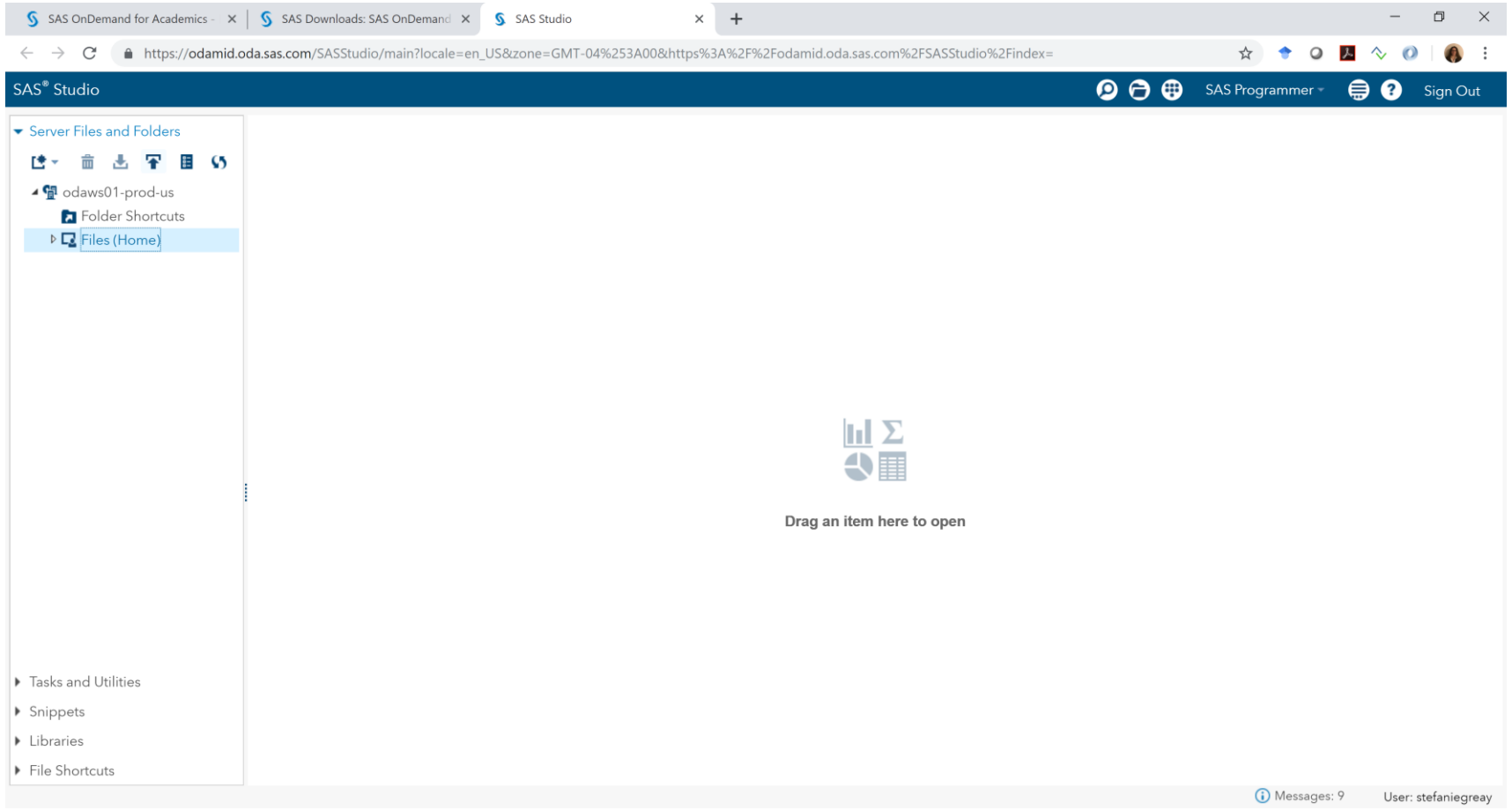
Dataset

- Download the Patient.xlsx file from the course datasets zip file or from the Unit 2 Welcome announcement in the course announcements.

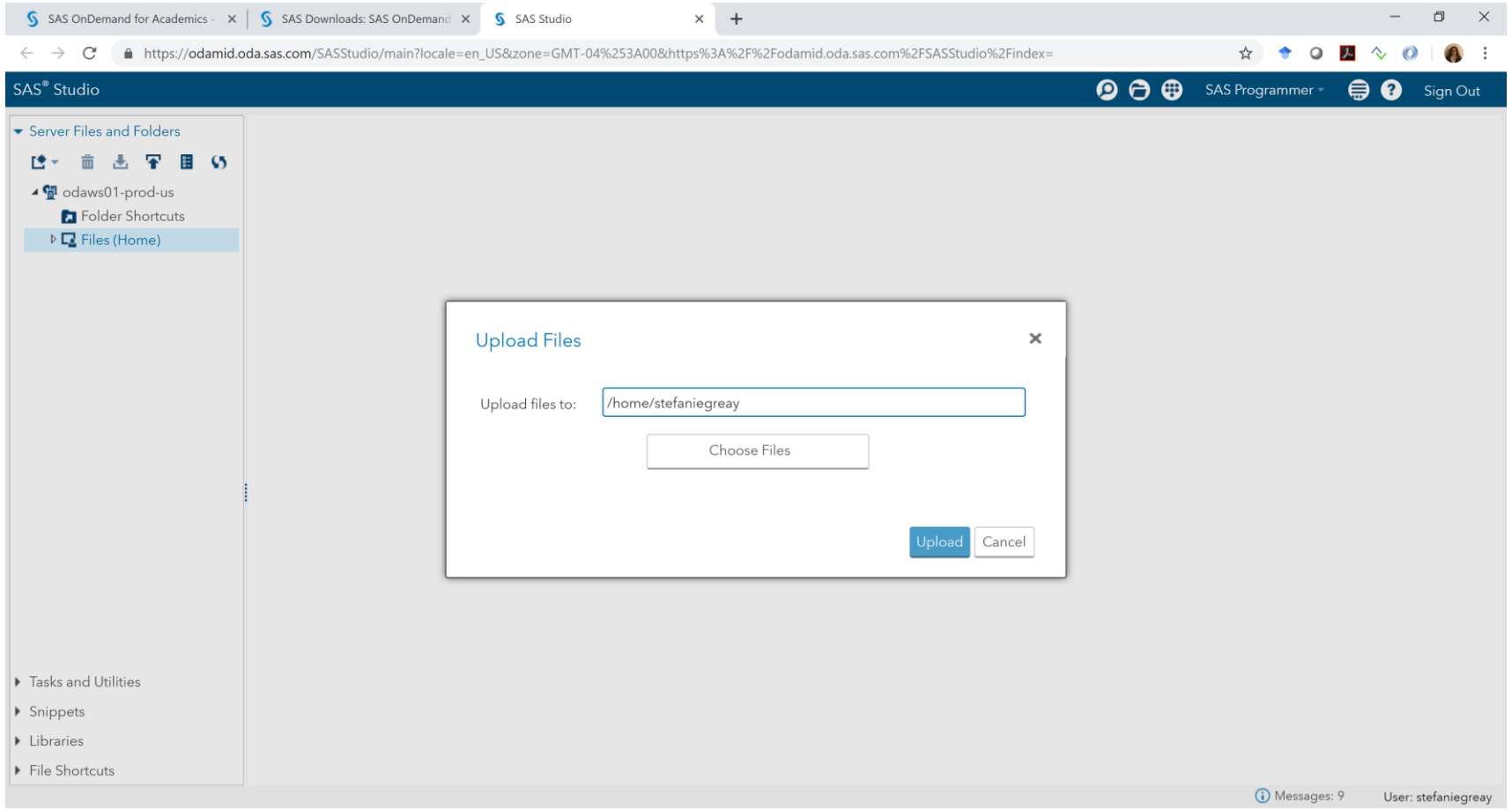
Click on Files(Home)



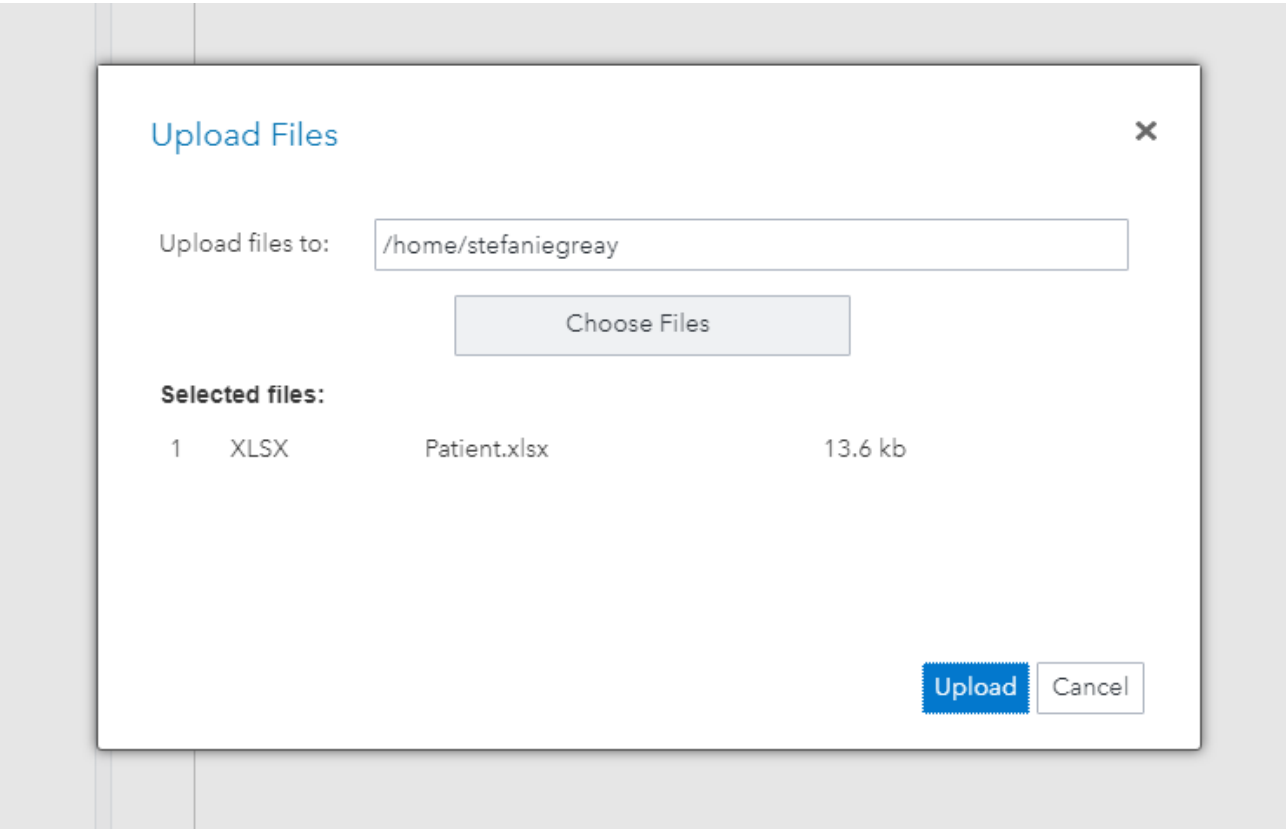
The Upload button will display in dark blue



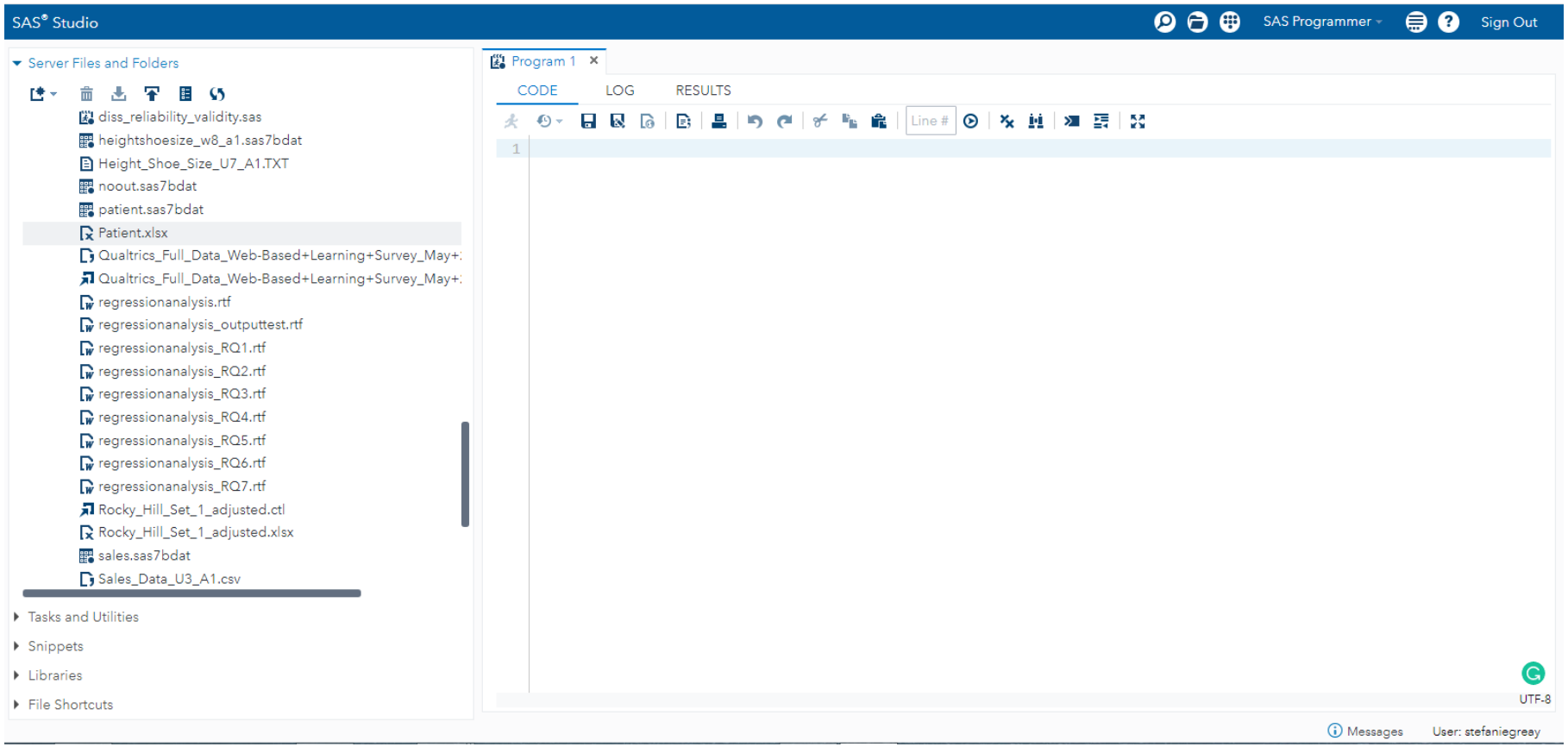
You can create a folder at this point, if you wish, or simply upload to your home directory.



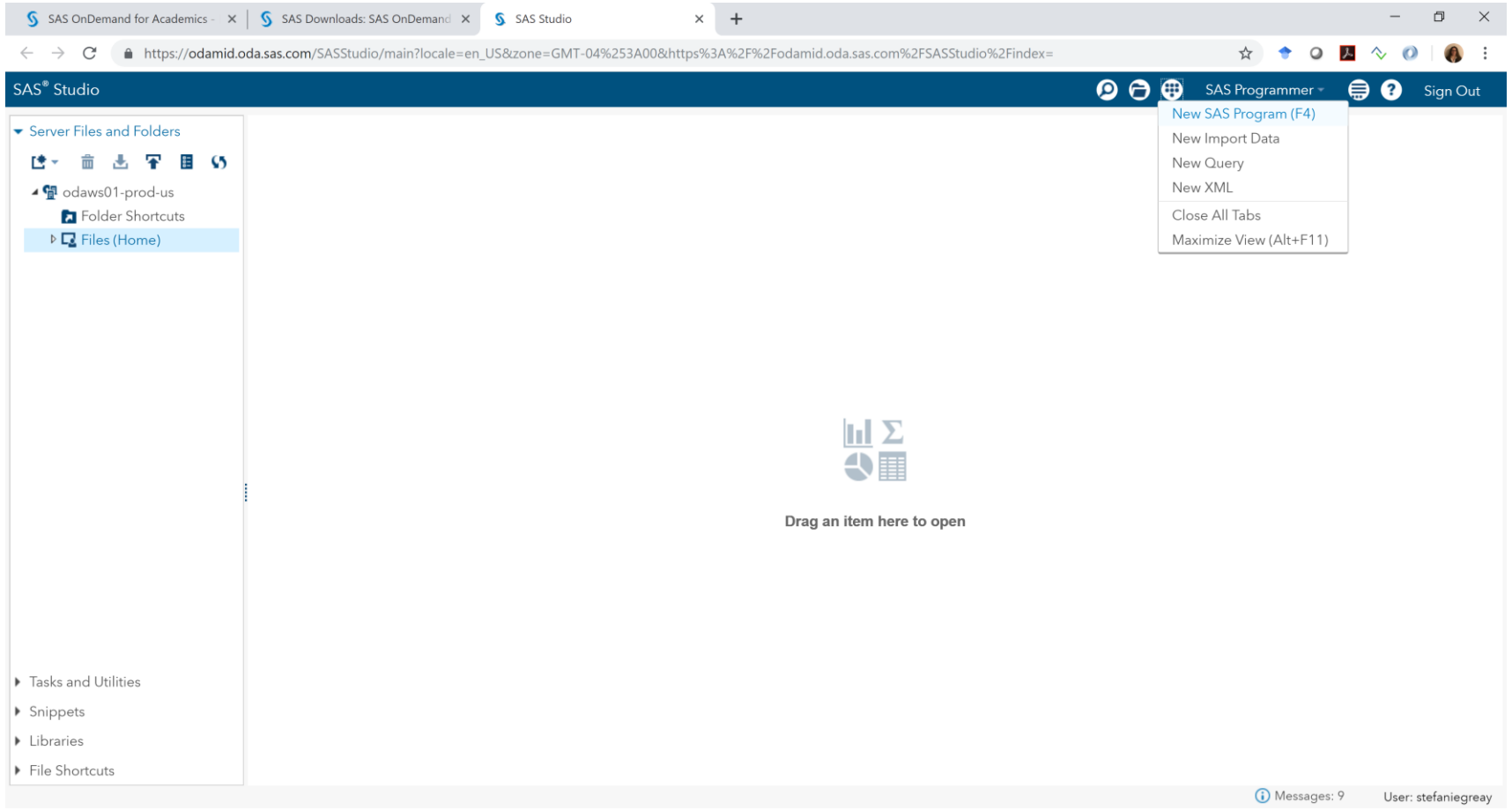
Select “Choose Files” to browse your computer for the dataset you want to upload. Once the dataset has been selected, click “Upload.”



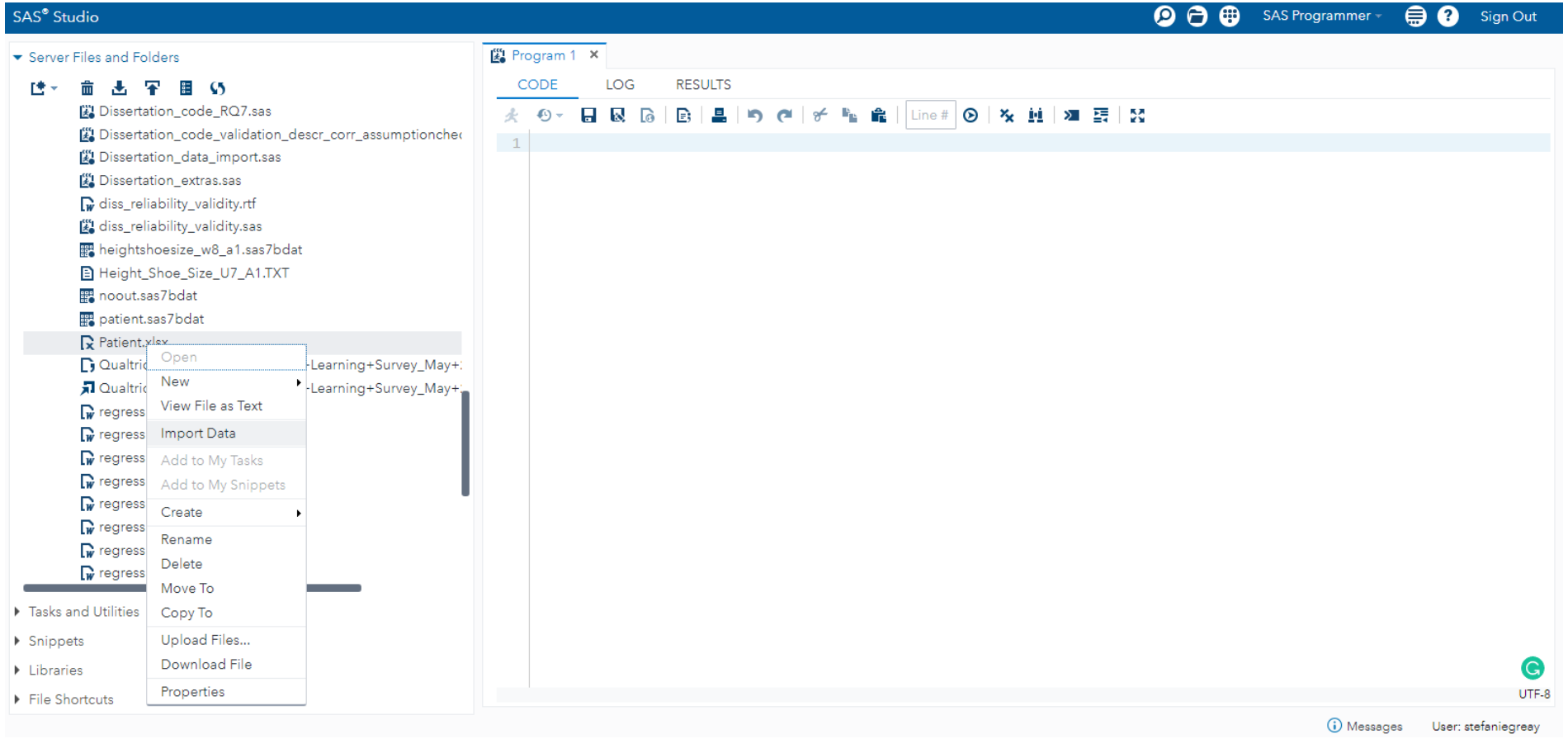
You will be able to view your files by clicking on “Files(Home)” to verify that your file successfully uploaded.



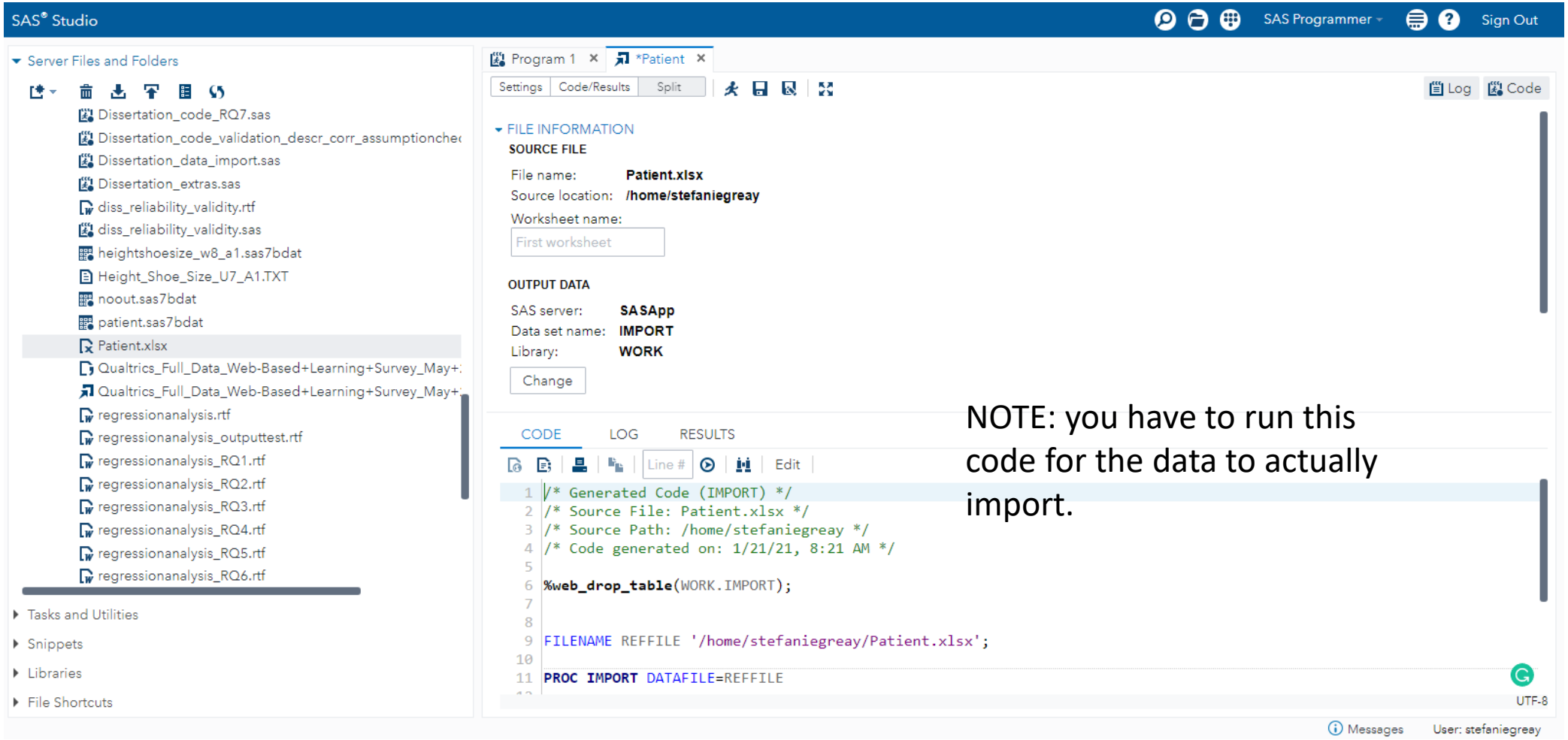
To get started with the SAS portion of the Unit 2 Assignment 1 assignment, start a new SAS program.



Import the dataset into a SAS dataset format (from the current xlsx format)

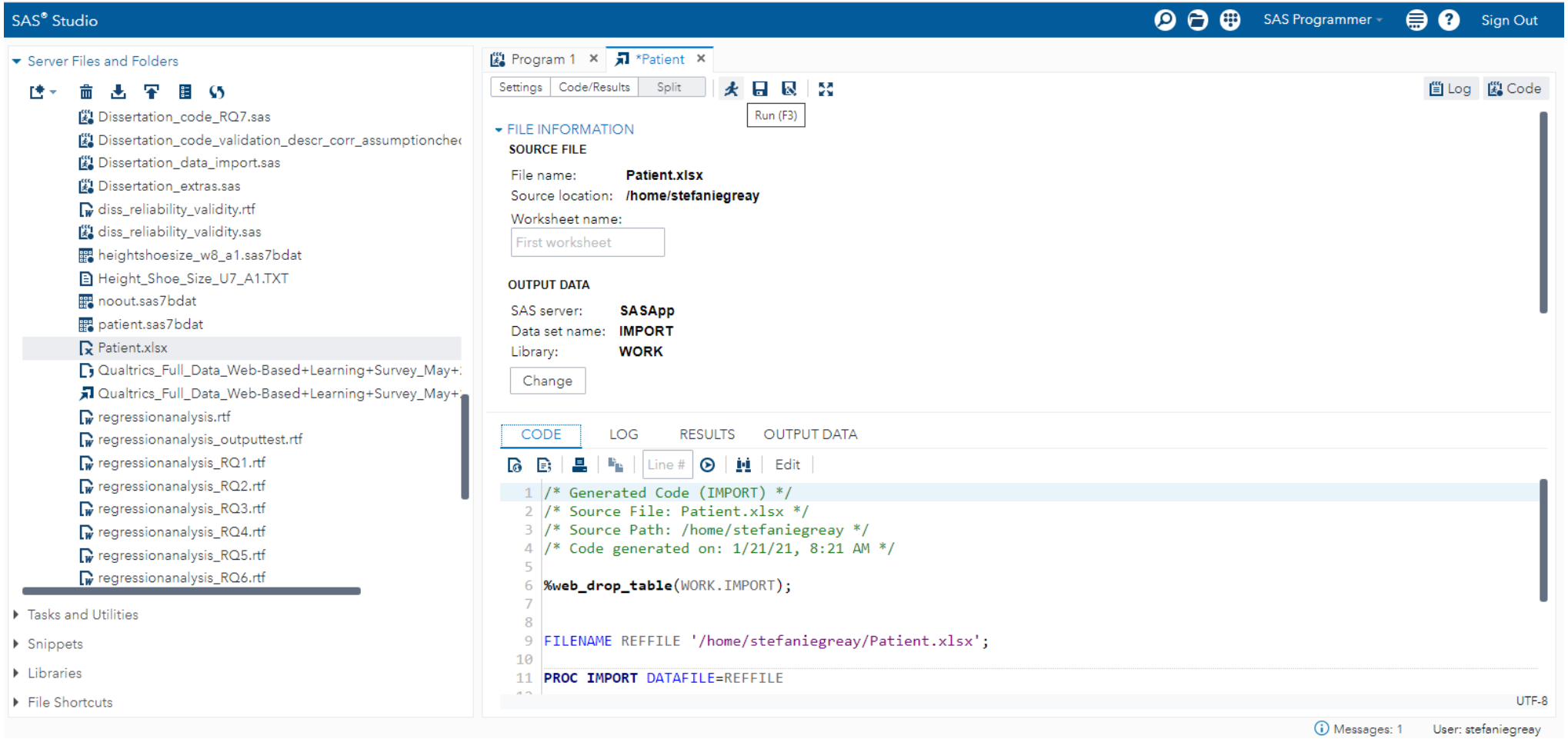


The Proc Import code will be written for you (save this as a template to use for future imports!)

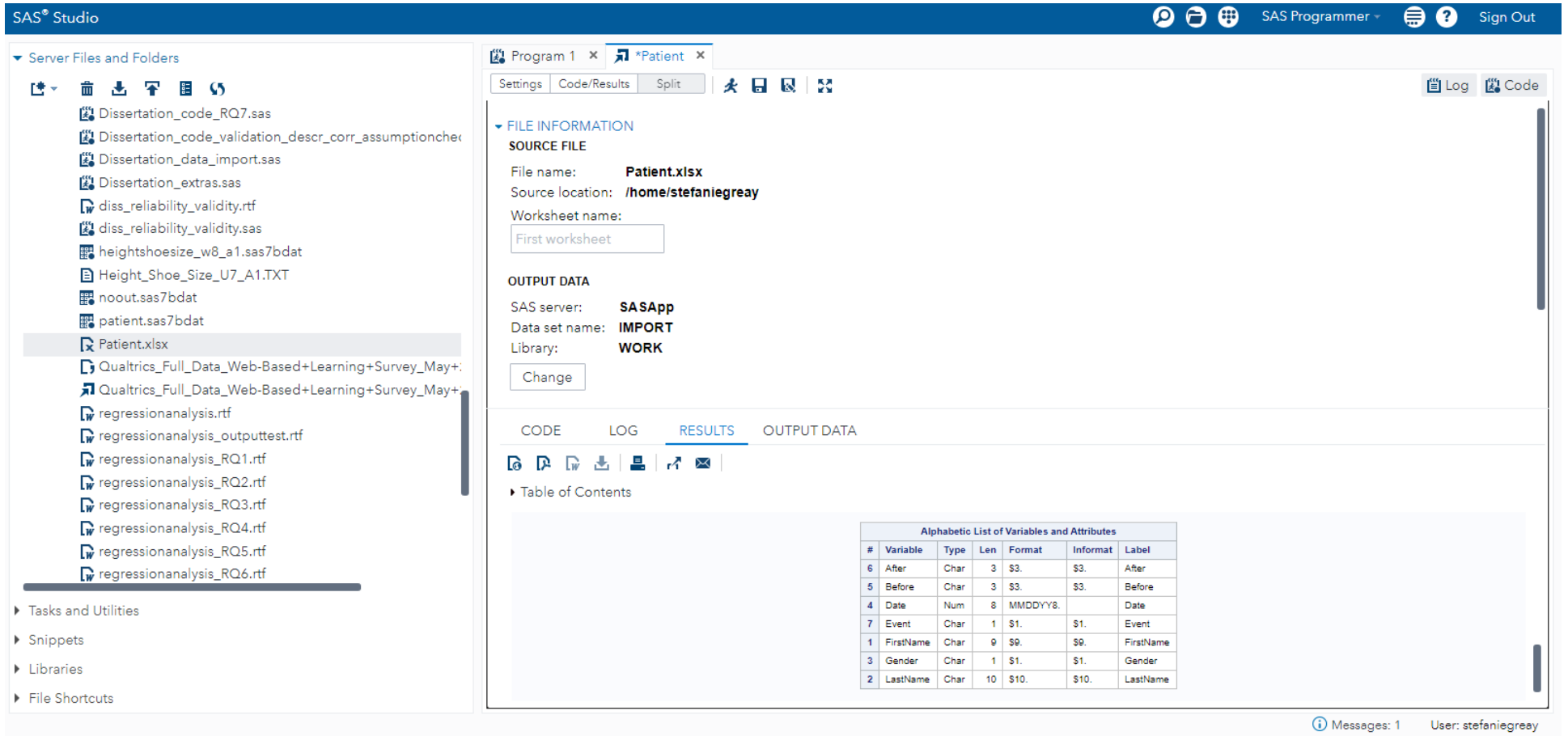


NOTE: you have to run this code for the data to actually import.

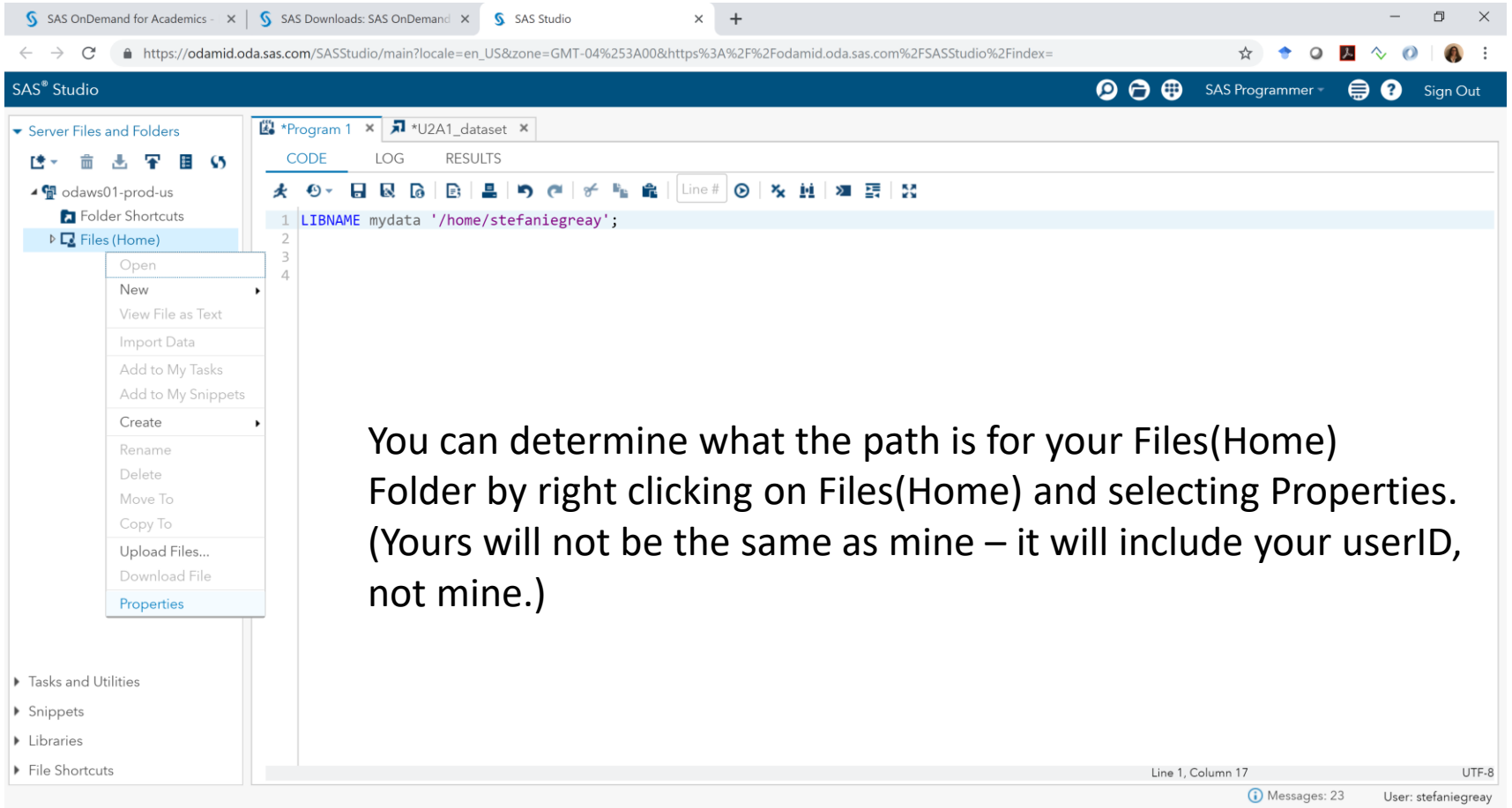
To run the code, click the icon that looks like a guy running.



For this particular dataset, because there is a . for missing values, two columns imported as character instead of numeric. We will have to fix this.

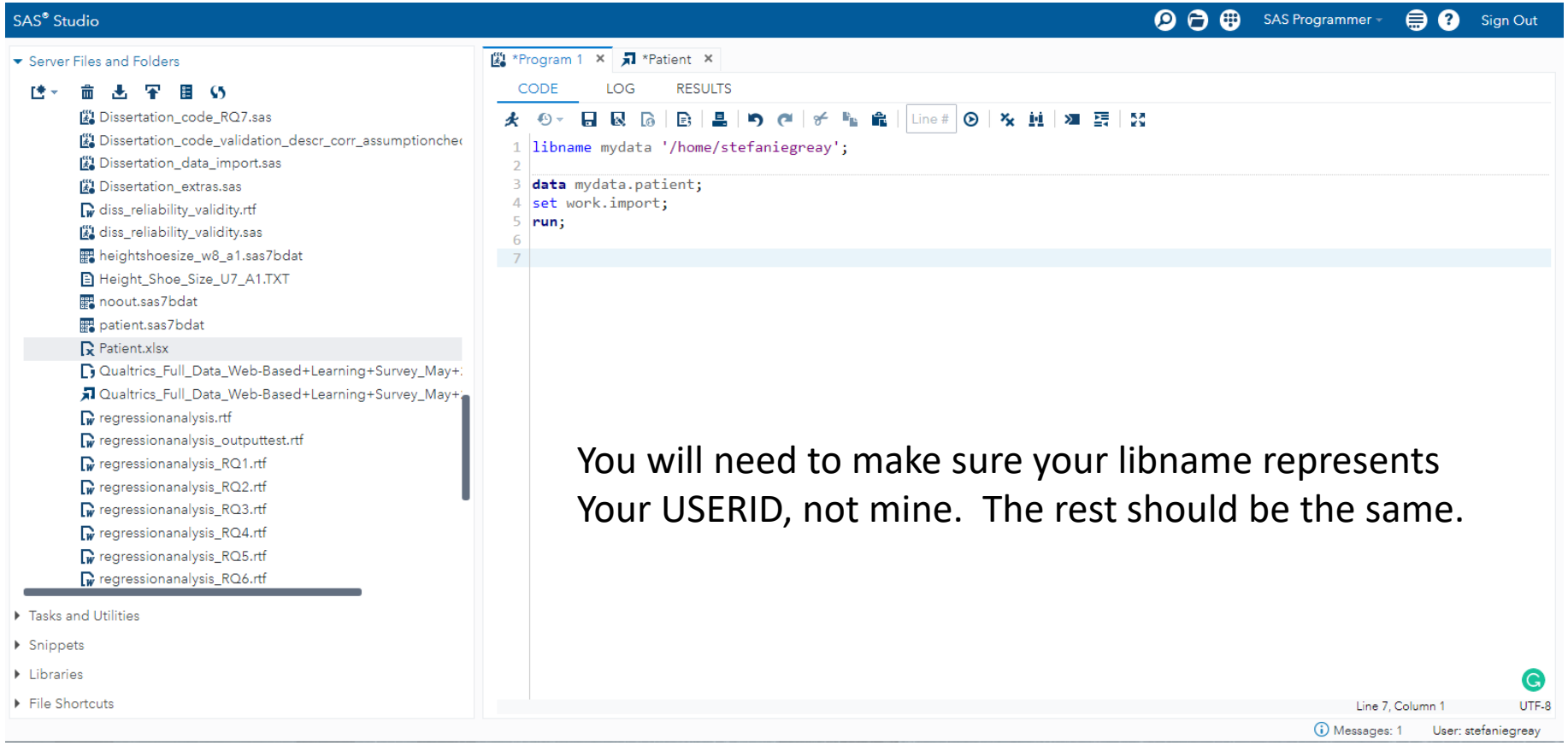


To create a SAS Library for your Files(Home) folder, you need to use a libname statement

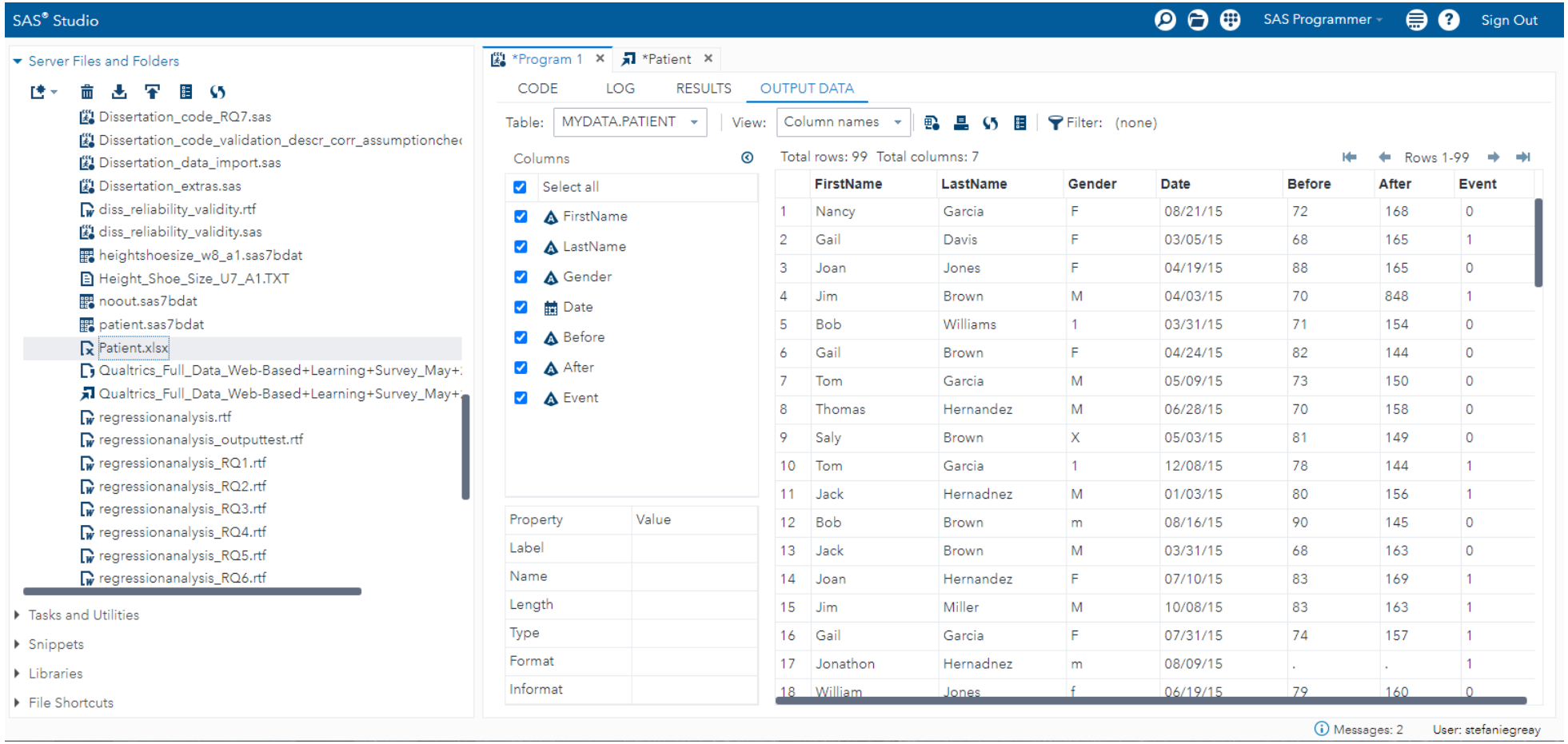


You can determine what the path is for your Files(Home) Folder by right clicking on Files(Home) and selecting Properties. (Yours will not be the same as mine – it will include your userID, not mine.)

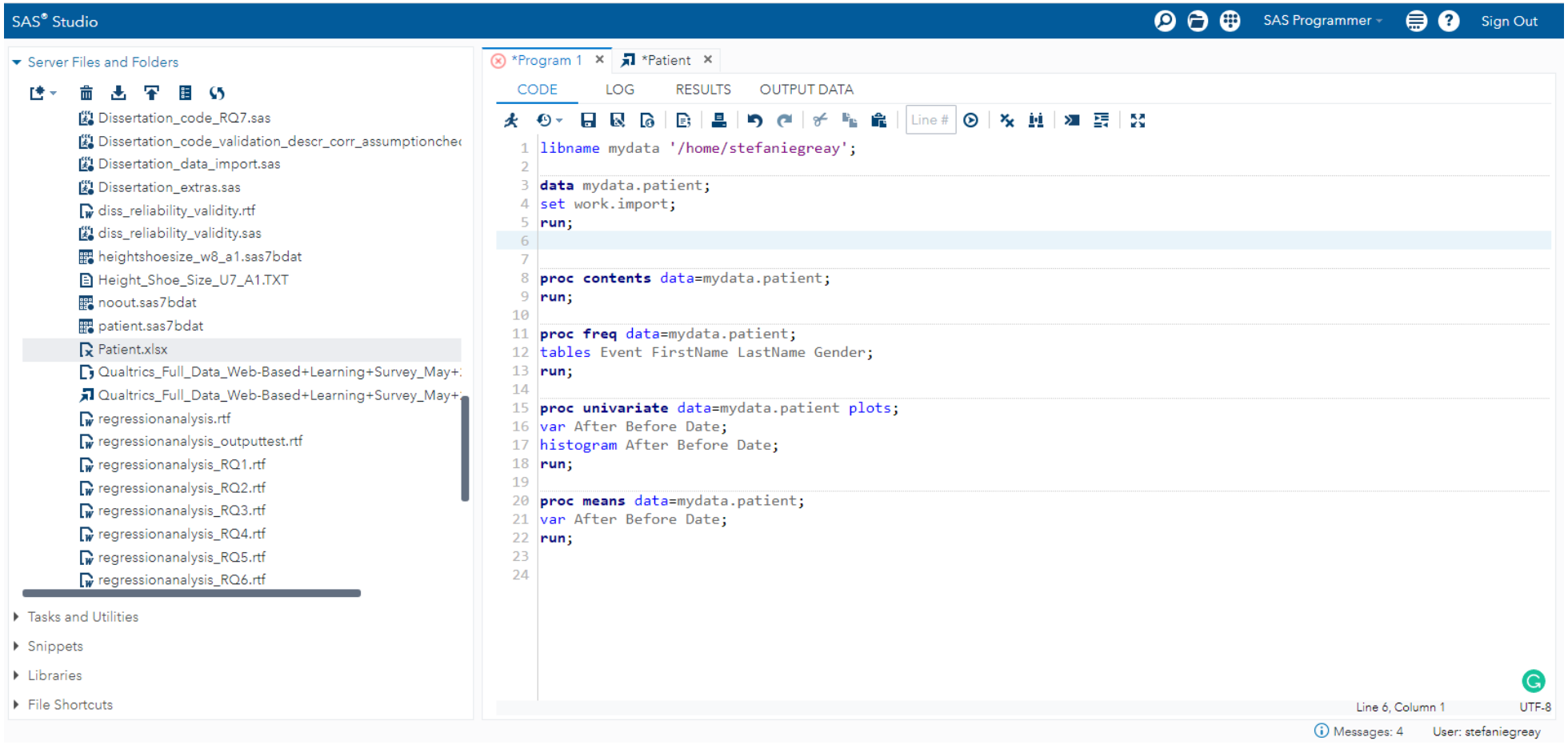
Save the temporary SAS dataset created by the import to your library using the following sample code.



When you run the code, you will see the dataset in the ouput data window and can verify its success.



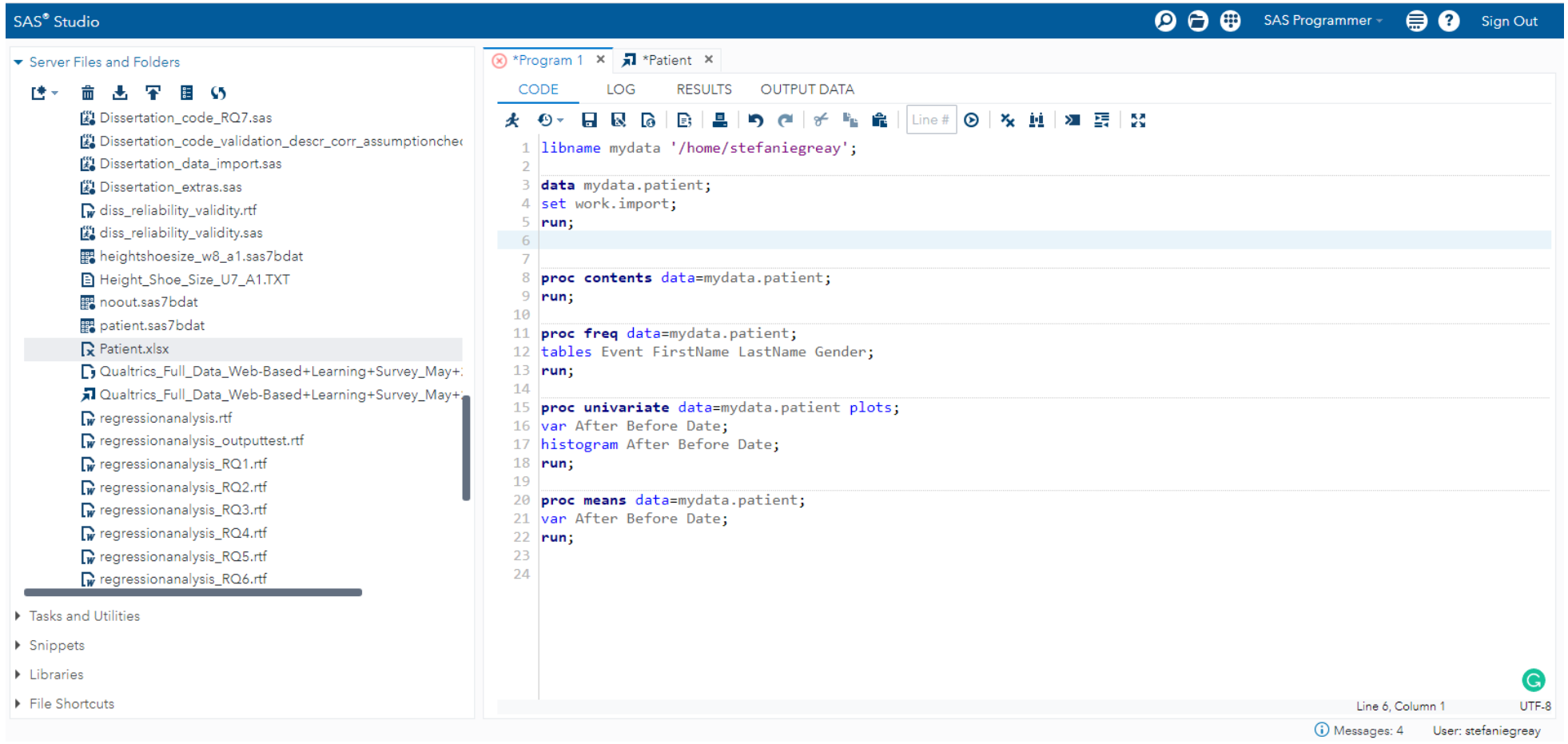
You can now run any procedures against that dataset via the code window.



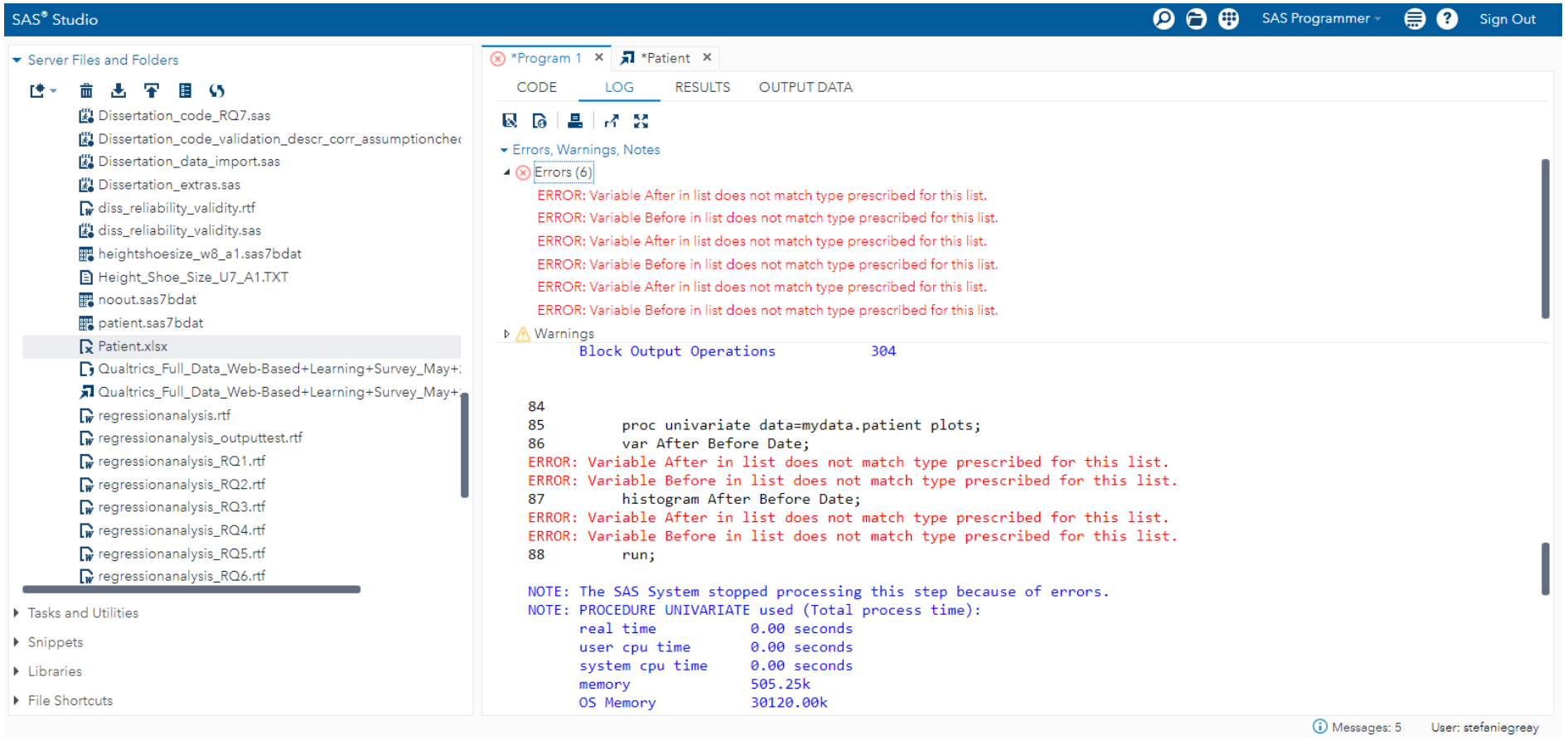
Sample Code

```
libname mydata '/home/stefaniegreay';

data mydata.patient;
set work.import;
run;


proc contents data=mydata.patient;
run;

proc freq data=mydata.patient;
tables Event FirstName LastName Gender;
run;

proc univariate data=mydata.patient plots;
var After Before Date;
histogram After Before Date;
run;

proc means data=mydata.patient;
var After Before Date;
run;
```


Once you run the code, you will notice a red x next to the program. This means we need to check the logfile for the errors.



Once you run the code, you will notice a red x next to the program. This means we need to check the logfile for the errors. All of the errors are shown in red text.



ERRORS

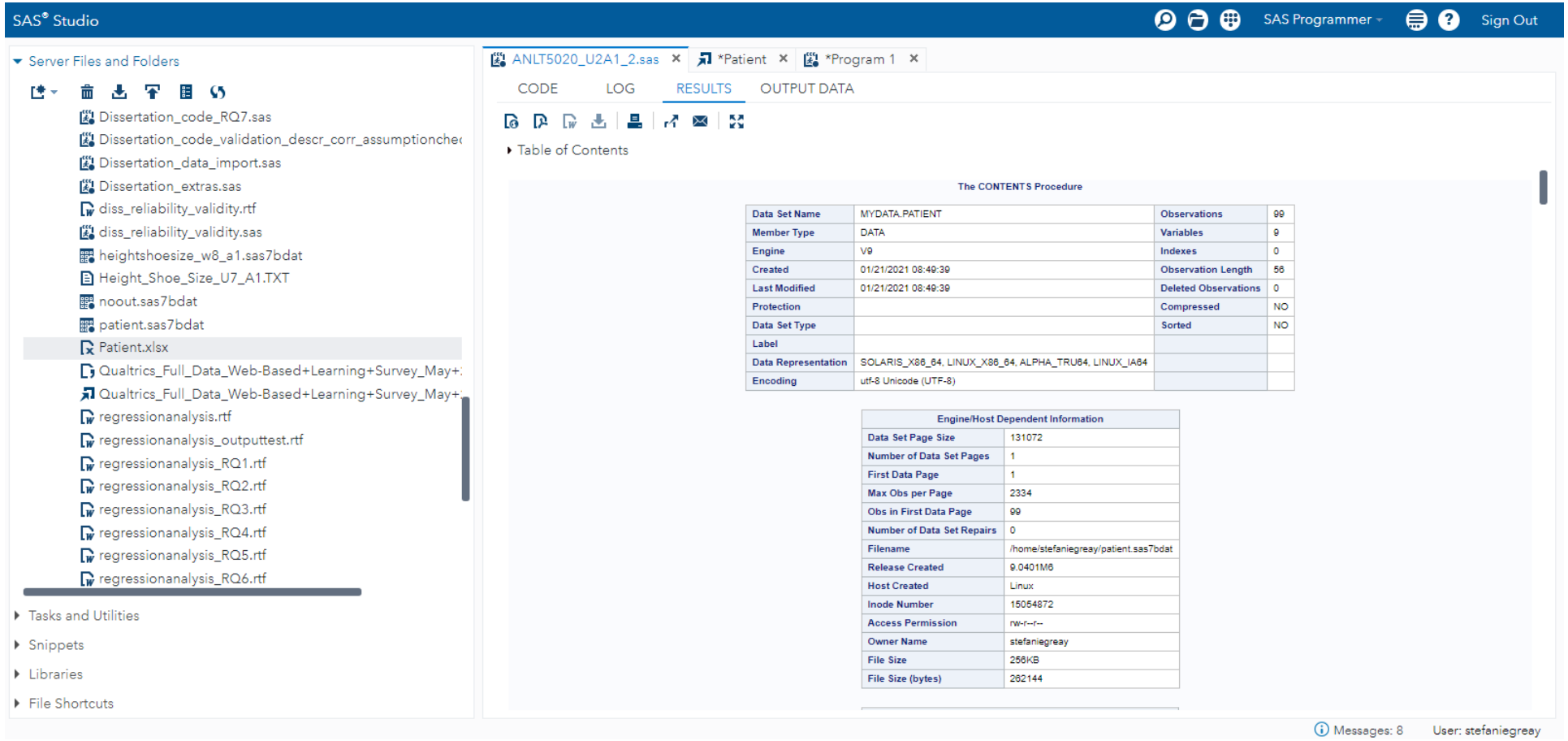
- The errors here are indicating that our data was imported in the wrong format. (I pointed this out in previous slides/screens, but we haven't fixed it yet. Not fixing it causes this type of error when we go to use those variables that should be numeric, but came in as character.)
- The solution is to convert the data type of those variables to numeric. We can do that using a data step in our code in SAS.
- We can write a new data step or do it in the same data step we already have to make our dataset permanent.

Sample Code

```
data mydata.patient;
set work.import;
Aftern = input(After, 3.);
Beforen = input(Before, 3.);
run;
```

- We are creating two new variables Aftern and Beforen that will be numeric. We are doing so using the “input” function in SAS with a format of 3.
- We will need to change the name of the variables used in the proc univariate and proc means to use these two new numeric variables.

We can now run our updated program and review the results.



Sample Code

```
libname mydata '/home/stefaniegreay';

data mydata.patient;
set work.import;
Aftern = input(After, 3.);
Beforen = input(Before, 3.);
run;


proc contents data=mydata.patient;
run;

proc freq data=mydata.patient;
tables Event FirstName LastName Gender;
run;

proc univariate data=mydata.patient plots;
var Aftern Beforen Date;
histogram Aftern Beforen Date;
run;

proc means data=mydata.patient;
var Aftern Beforen Date;
run;
```