Emily Durfey

June 8, 2021

Foundations of Python Programming

Assignment 09

GitHub URL: https://github.com/edurfey/IntrotoProg-Python-Mod09

Modules in Python

Introduction

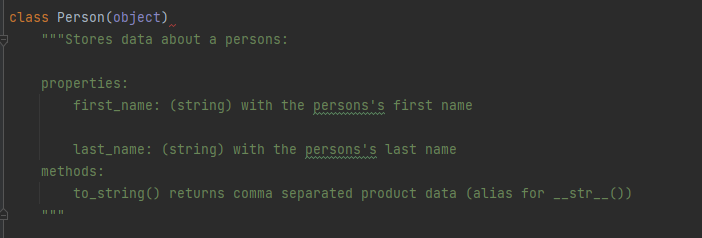
In this paper, I will describe the steps taken to create Python scripts to define modules. These scripts are housed in individual files within PyCharm and combined within a main file. The output is captured on a text file.

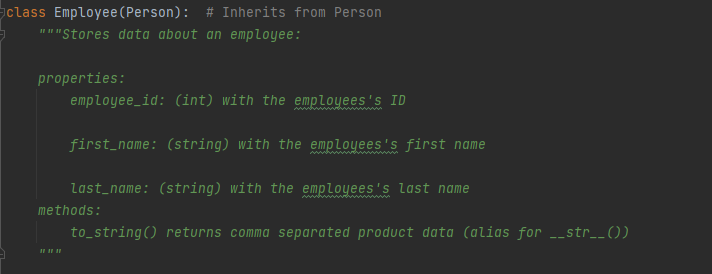
Creating the Files

I began creating my scripts by creating a new project in PyCharm labelled “Assignment09”. Once this had been set up, I created several new Python files that would be used to build the main script. The script templates were provided by the instructor for each section, which were tremendously helpful to use as I worked through the module. Each section is described in greater detail below.

Data Classes Python File

The first file, “Data Classes”, features several scripts that allow the user to add participant names to the text file and save the data. This section features a combination of several different types of scripts, including “def”, “if/else”, and “return”. During this section, I was able to adjust the template and update the Python file to include a “Person” class, which would be used in the main Python script. The class additions are featured below in Figures 9.1 and 9.2:





Test Harness Python File

The second file, “Test Harness”, was used to create the main testing module for the main Python script. This section first allows the user to import information based on the other scripts created and displays an error message if the program cannot match the data provided. This file also features a variety of scripts (similar to the previous Python file) that prompt the user to save the data and instruct the program to display the information if needed. It also allows the user to test the input and output at the end of the file.

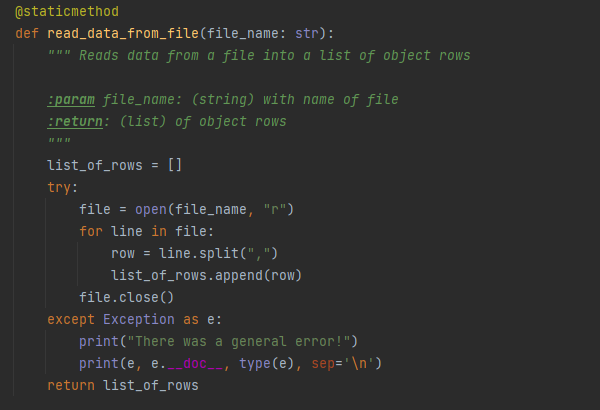
The testing methods in this section are outlined below in Figure 9.3:



Processing Classes Python File

The third file, “Processing Classes”, builds on the previous Python files and uses multiple classes in the code in order to both save new data and allow the file to read the new information. This section features several types of scripts, including “try”, “return”, and “def”, and it also incorporates an exception error message should the user enter the incorrect information. There are also arrays of data that can be used to define the rows of information exported to the text file.

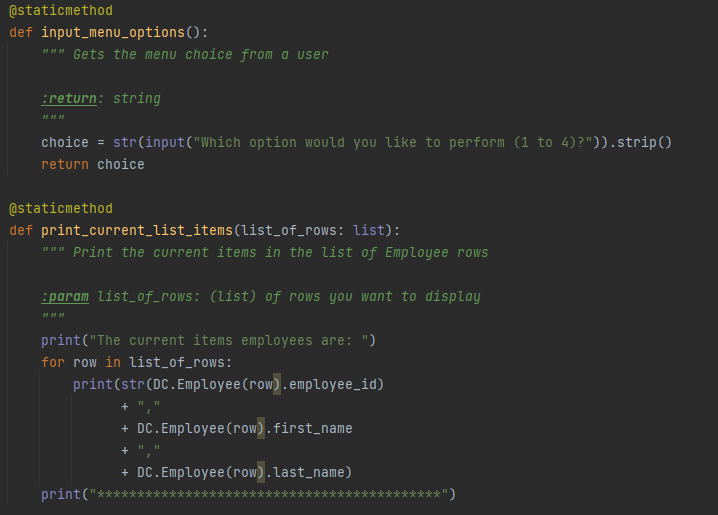
Part of this script is outlined below in Figure 9.4:



IO Classes Python File

The final individual Python file, “IO Classes”, features scripts similar to the ones introduced in earlier modules. They prompt the user to choose from a menu of options in order to perform the task they wish to move forward with. There are four different choices available, and the scripts feature a mixture of scripts such as “print”, “def”, and “return”. If the user chooses an incorrect option, there is also an error message built into the code to inform them that an issue has occurred. This file also allows the user to import the data classes file as part of the main Python file.

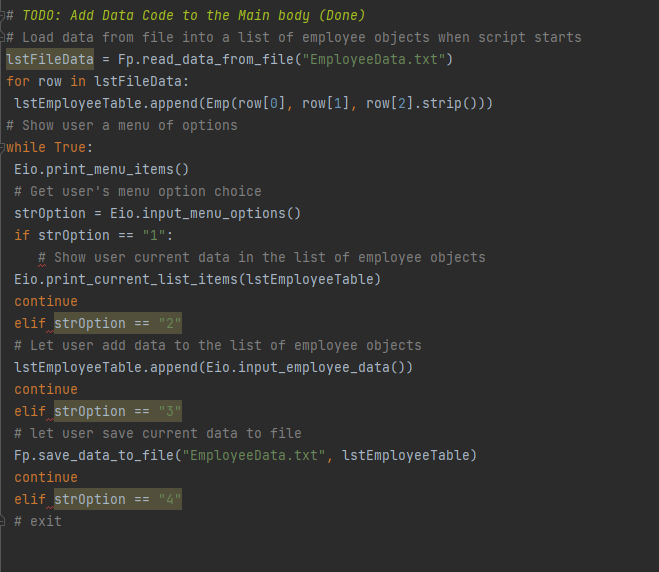
Part of this file is displayed below in Figure 9.5:



Main Python File

The main Python file incorporates each of the four separate Python files created earlier, and it imports the information into a user-friendly, physical text file (“EmployeeData”). The options created for this script are similar to the ones in the “IO Classes” file and display the choices available to the user through a series of if/elif statements. Once the user has added their data, they are able to save their updates and exit the file if they choose. They can also continue to add more information to the file if they desire.

The script in this section is displayed below in Figure 9.6:



Summary

By using the examples provided in the textbook, Module 9 video, and supplemental materials, I was able to create a series of individual Python files that can be combined to create a main Python file/more sophisticated text file. I enjoyed working through this assignment as it allowed for a degree of creativity and approaching text file creation with a new approach; I look forward to working on the final module of the class and eventually build on this information by enrolling in the Python certificate program.