Nikita Daharia
June 16, 2022
Foundations of Programming: Python
Assignment 09
https://github.com/nikitadaharia/nikitadaharia-IntroToProg-Python-Mod09/upload

Introduction:

This paper aims to demonstrate the python programming skills I have learned in this course. I will be organizing code into script modules that can be imported into the main script. The three modules we had for importing were DataClasses, ProcessingClasses, and IOClasses. The latest version of PyCharm Community version is used for this assignment on a Mac OS.

Organizing code into script modules

I started by creating a new sub-folder called Assignment09 inside of the _PythonClass folder (created in Module 01) in the Documents folder on a Mac OS. Then I created a new project in PyCharm that uses the _PythonClass\Assignment08 folder as its location.

After that, I downloaded the Module 9 listings and saved the modules DataClasses, ProcessingClasses, IOClasses, and Employee.txt in the Assignment09 folder along with my main file.

The main script includes the learning from the previous assignments to organize code into script modules.

```python
# TODO: Import Modules
if __name__ == "__main__":
    import DataClasses as DC  # data classes
    import ProcessingClasses as P  # processing classes
    import IOClasses as IO  # IO classes
else:
    raise Exception("This file was not created to be imported")

# Main Body of Script  ---------------------------------------------------- #
# TODO: Add Data Code to the Main body. Done.
lstEmployees = []  # list of lists
strFile = "EmployeeData.txt"
strChoice = ""
strStatus = ""
lstObjects = [] # list of employee objects

# Load data from file into a list of employee objects when script starts
try:
    lstEmployees = P.FileProcessor.read_data_from_file(strFile)
    for item in lstEmployees:
        emp_obj = DC.Employee(item[0], item[1], item[2].strip())  # reconfiguring list into objects
        lstObjects.append(emp_obj)

except FileNotFoundError as e:
    print("File not found.")
except Exception as e:
    print(e, e.__doc__, type(e), sep='\n')
```

Figure 1. Snippet of "Main.py" i.e., the main body of the script code.

Now, run the python script in PyCharm.

```
Which option would you like to perform? [1 to 4] - 2

What is the employee Id? - 3
What is the employee First Name? - James
What is the employee Last Name? - Potter

3,James,Potter <class 'DataClasses.Employee'>

        Menu of Options
        1) Show current employee data
        2) Add new employee data.
        3) Save employee data to File
        4) Exit program


Which option would you like to perform? [1 to 4] - 2

What is the employee Id? - 4
What is the employee First Name? - Lily
What is the employee Last Name? - Potter

4,Lily,Potter <class 'DataClasses.Employee'>
```

Figure 2. Snippet of output saving new employee data displayed in PyCharm Shell after running the python script.

Run the script on the Terminal window

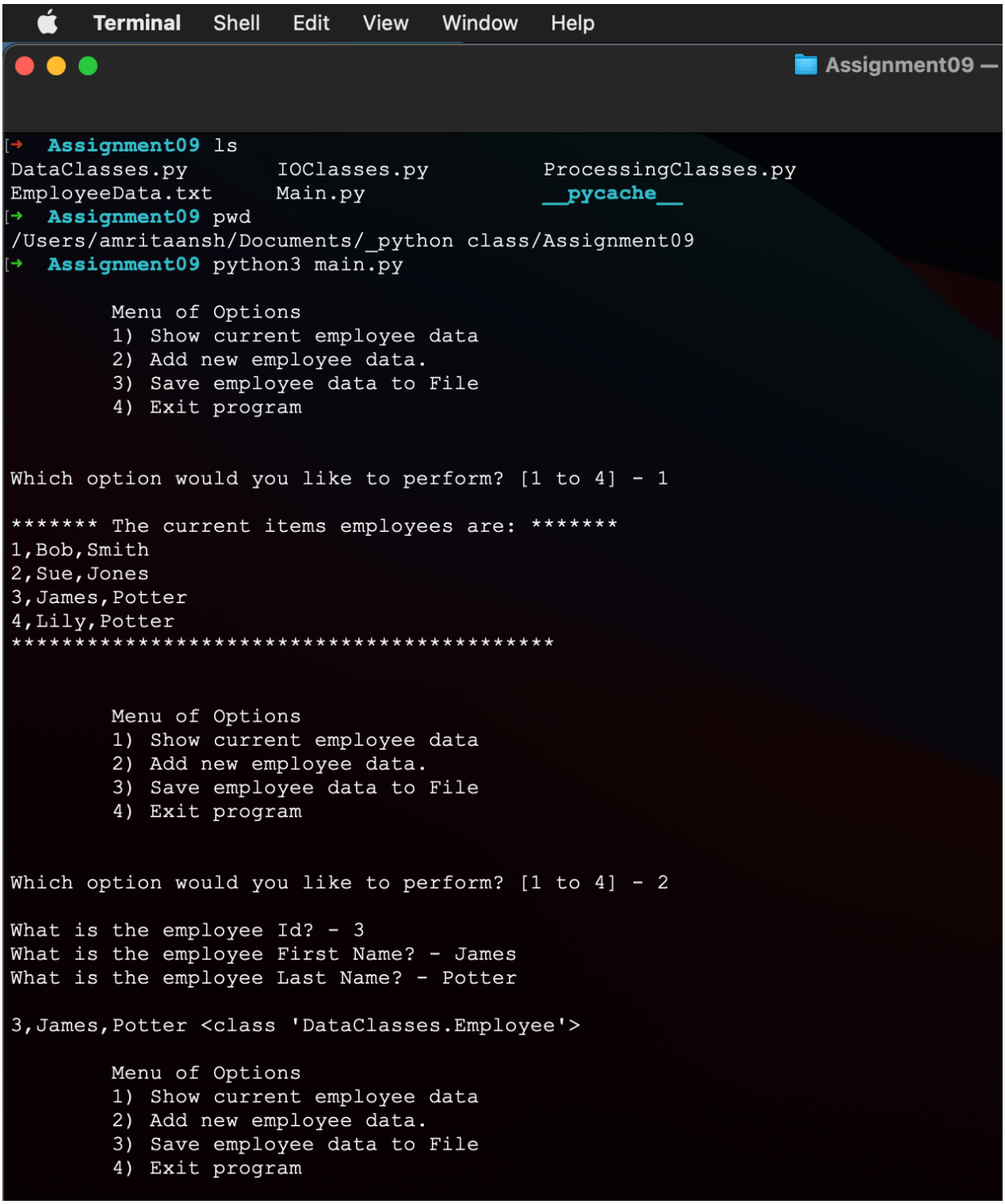

Figure 3. Output displayed in Terminal window after running the python script.

Lastly, the text file Employees.txt was located and opened that it worked and the updated data was saved.

Summary:

Python is a simple yet powerful language programming language that runs on Windows, Linus/Unix, and Mac OS. I used PyCharm to create a python script organizing code into script modules that can be imported into the main script. The script was run both in PyCharm and Terminal . Finally, the code was verified by locating the text file products.txt.