Lim Sia

August 13, 2024
Foundations of Programming: Python
Assignment 07

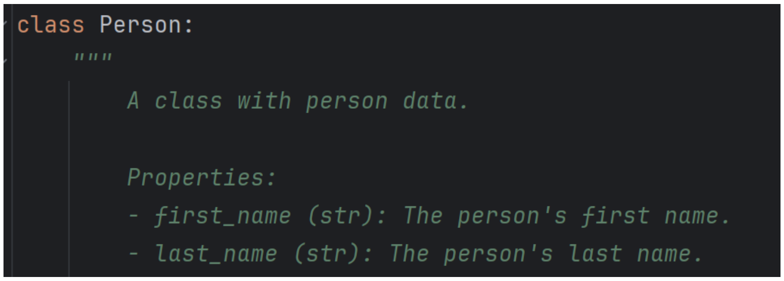
To create sets of data classes

Introduction

Based on assignment 6, in this assignment, we create a Python program that demonstrates using constants, variables, and print statements to display a message about a student's registration for a Python course, but we added set of data classes.

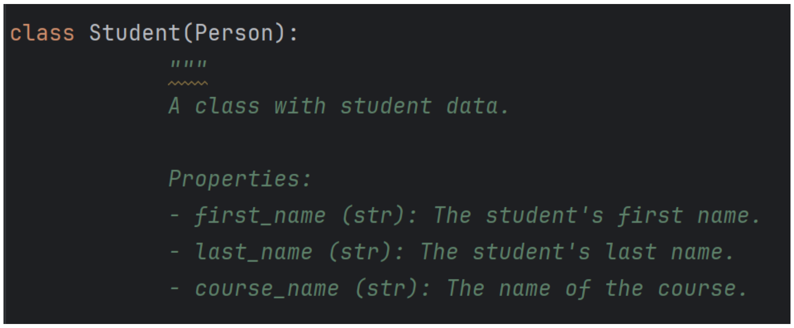
Objects and classes

A class is a blueprint for an object, and an object is an instance of a class with attributes and methods. For this assignment, we are required to create two classes. The first class contains objects with students' first names and last names while the second class extends from the first class, which adds course names to the objects. The benefits of the data classes are: (1) so that each object is isolated from the others and changes made to one object's data therefore, it helps with data integrity; (2) classes serve as templates, where as many objects can be created as needed so that they are reusable, which simplifies code development and maintenance; (3) interaction with objects can be carried out without having to worry about how data is stored or processed internally, therefore it simplifies the complexity of the code.



```
class Person:
    """
        A class with person data.

        Properties:
        - first_name (str): The person's first name.
        - last_name (str): The person's last name.
```

Figure 1. Creating the "Person" class for student names

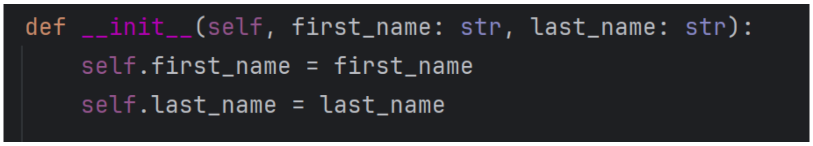


```
class Student(Person):
            """
            A class with student data.

            Properties:
            - first_name (str): The student's first name.
            - last_name (str): The student's last name.
            - course_name (str): The name of the course.
```

Figure 2. Creating the "Student(Person)" class to also include course names

Creating the script using the *Self* keyword

The *Self* keyword is used to refer to data or functions found in an object instance, and not directly in the class. By using the *Self* keyword, it allows access to the attributes and methods of the class, which binds the attributes (i.e. a piece of data, or a characteristic associated with an object) with the given arguments.

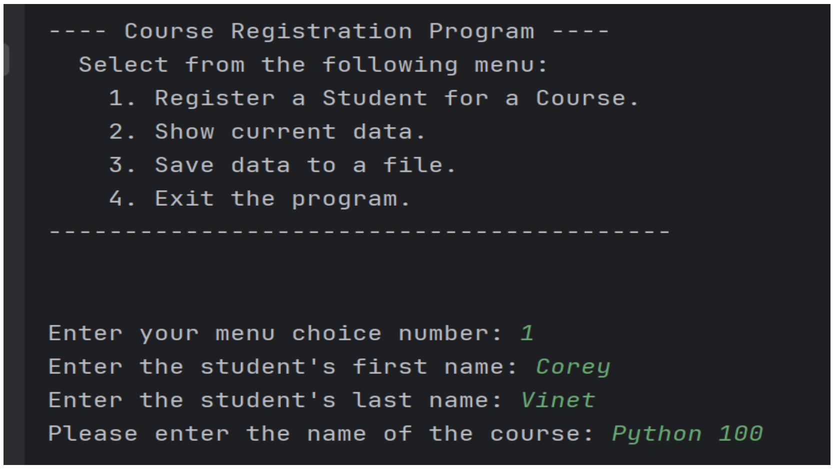


```
def __init__(self, first_name: str, last_name: str):
    self.first_name = first_name
    self.last_name = last_name
```

Figure 3. Creating the script with the "Self" keyword

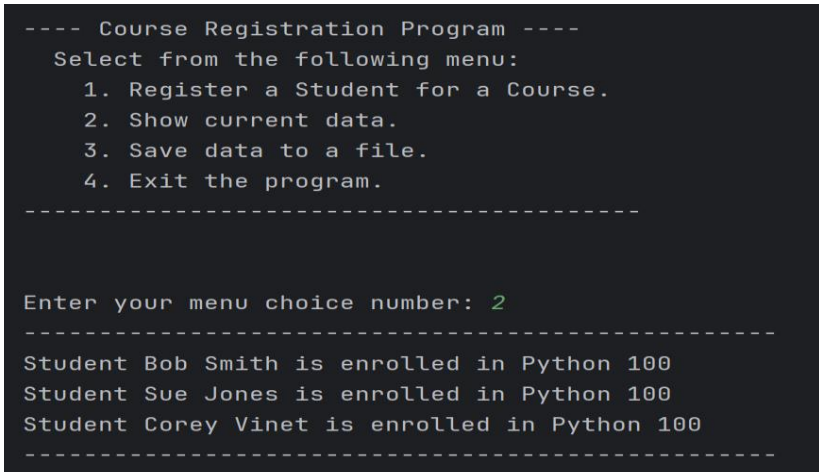
Running the program

Initial errors from running the program included inability to locate the json file, indentations, typos, and various spacings in the scripts. After a series of debugging and testing, the program worked accordingly.



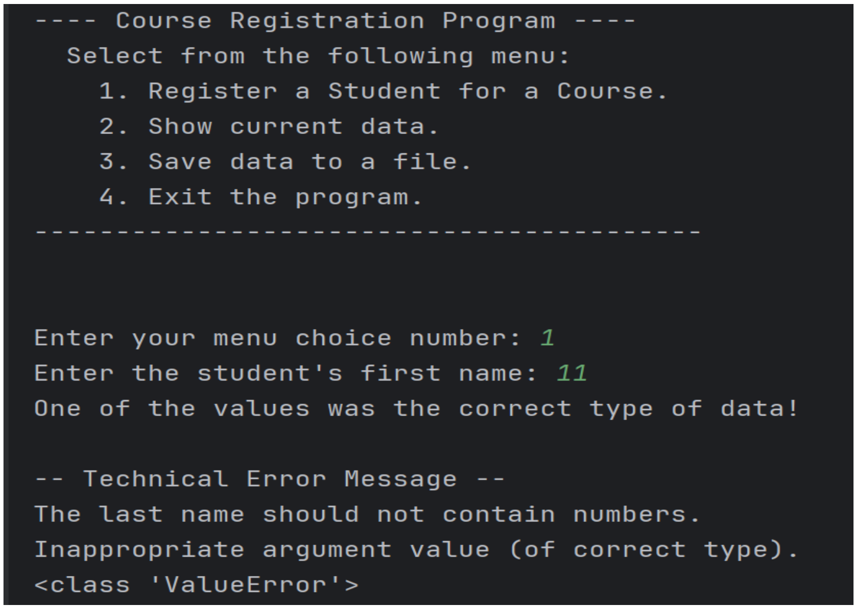
```
---- Course Registration Program ----
  Select from the following menu:
    1. Register a Student for a Course.
    2. Show current data.
    3. Save data to a file.
    4. Exit the program.
-----------------------------------------



Enter your menu choice number: 1
Enter the student's first name: Corey
Enter the student's last name: Vinet
Please enter the name of the course: Python 100
```

Figure 4. A student registered to the student list



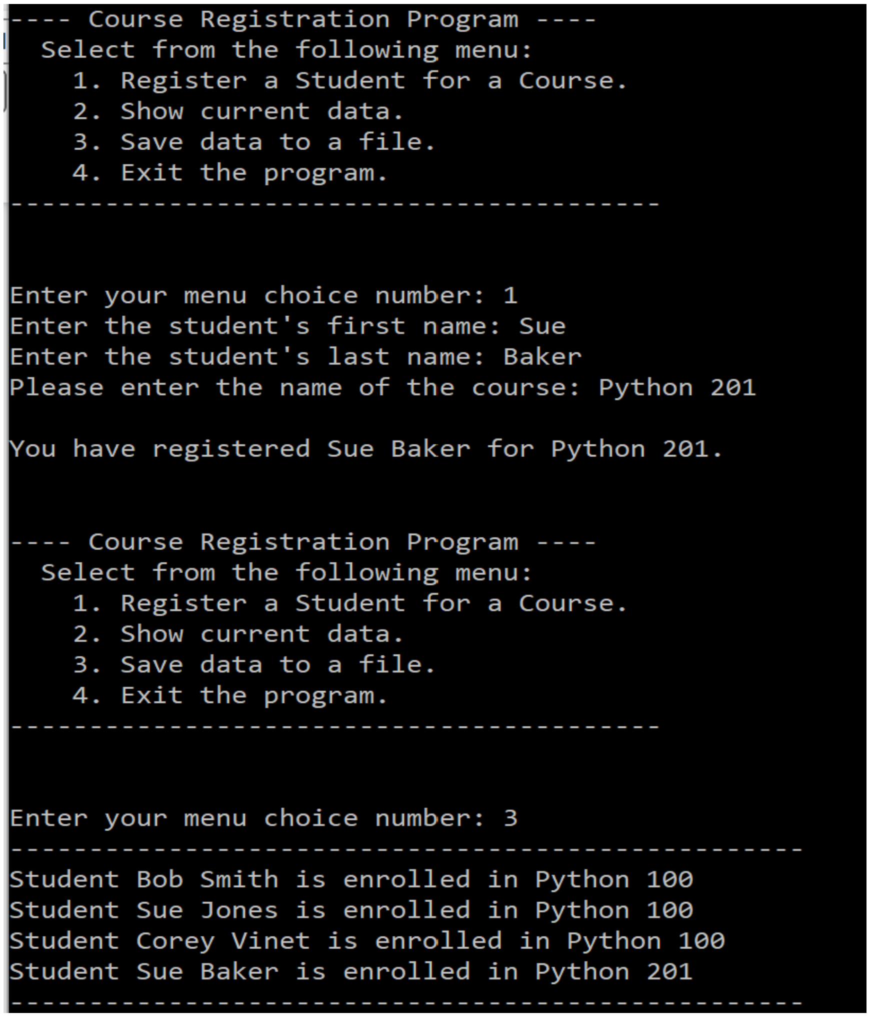
```
---- Course Registration Program ----
  Select from the following menu:
    1. Register a Student for a Course.
    2. Show current data.
    3. Save data to a file.
    4. Exit the program.
-----------------------------------------



Enter your menu choice number: 2
--------------------------------------------------
Student Bob Smith is enrolled in Python 100
Student Sue Jones is enrolled in Python 100
Student Corey Vinet is enrolled in Python 100
--------------------------------------------------
```

Figure 5. Data collected in the student list



```
 ---- Course Registration Program ----
   Select from the following menu:
     1. Register a Student for a Course.
     2. Show current data.
     3. Save data to a file.
     4. Exit the program.
 -----------------------------------------



 Enter your menu choice number: 1
 Enter the student's first name: 11
 One of the values was the correct type of data!

 -- Technical Error Message --
 The last name should not contain numbers.
 Inappropriate argument value (of correct type).
 <class 'ValueError'>
```

Figure 6. Tested the error message when numbers were entered for student's first name

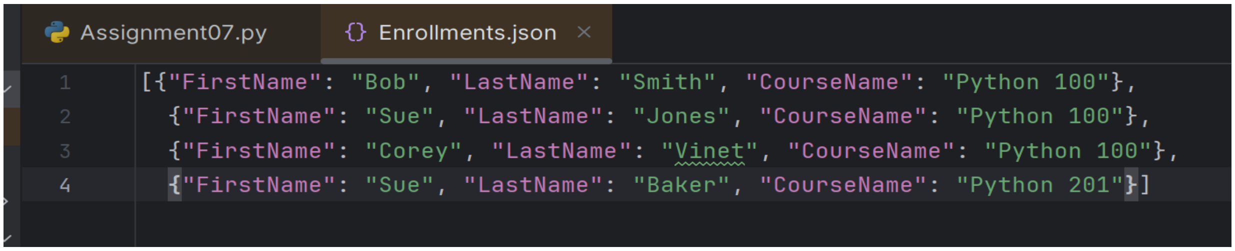


```
---- Course Registration Program ----
  Select from the following menu:
    1. Register a Student for a Course.
    2. Show current data.
    3. Save data to a file.
    4. Exit the program.
-----------------------------------------



Enter your menu choice number: 1
Enter the student's first name: Sue
Enter the student's last name: Baker
Please enter the name of the course: Python 201

You have registered Sue Baker for Python 201.


---- Course Registration Program ----
  Select from the following menu:
    1. Register a Student for a Course.
    2. Show current data.
    3. Save data to a file.
    4. Exit the program.
-----------------------------------------



Enter your menu choice number: 3
--------------------------------------------------
Student Bob Smith is enrolled in Python 100
Student Sue Jones is enrolled in Python 100
Student Corey Vinet is enrolled in Python 100
Student Sue Baker is enrolled in Python 201
--------------------------------------------------
```

Figure 7. The program runs well on the command prompt

Saving the collected data to the *JSON* file and posting to GitHub

Working with the Json file has been tricky for this assignment, when I had to go back and check on the file all the time in order to make sure that it showed up under the correct directory in order for my program to see the file. Posting to GitHub was straightforward after the previous two assignments.



```
[{"FirstName": "Bob", "LastName": "Smith", "CourseName": "Python 100"},
  {"FirstName": "Sue", "LastName": "Jones", "CourseName": "Python 100"},
  {"FirstName": "Corey", "LastName": "Vinet", "CourseName": "Python 100"},
  {"FirstName": "Sue", "LastName": "Baker", "CourseName": "Python 201"}]
```

Figure 8. Data shown in the Enrollments.json file

Summary

The assignment notes and the video helped a lot with this assignment. After the previous assignments and the errors I had ran across, while getting more familiar with how *PyCharm* works, programming is starting to makes more sense to me. I look forward to learning more on *Python*.