

Bonus [OPTIONAL] Assignment

Deadline: 11.59 pm 13th July 2022 **[Agreed and Requested by you guys]**

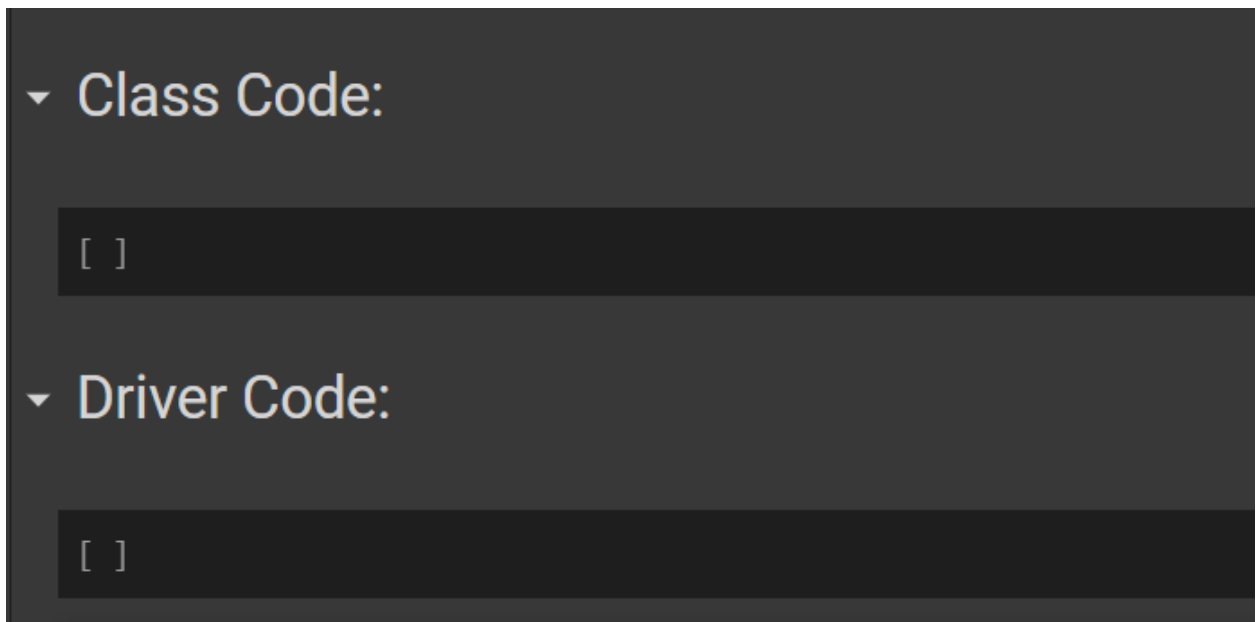
Submission link: <https://forms.gle/U7JENricx9rzNDLr6>

Reward: Reinstatement of Partial Marking **[if >=50% of students actually try]**

Read each step carefully and execute them carefully. In this assignment, you will be writing both class and driver code. **The whole assignment is open-ended, meaning there is no reason for 2 submissions to be exactly the same. You can use additional print statements wherever you feel necessary.**

The student class has come back to haunt you.

1. Open a blank Jupyter Notebook / Google Colab file.
2. Make 2 “CODE” Segments in the following way:



All of your classes will be in the “Class Code” section and all of your Driver code (making objects, calling methods) will be in the Driver Code section.

3. Write a class Student such that:
 - a. Mandatory Parameters: Name, ID,
 - b. Optional Parameters: gender, DOB
 - c. Make a new variable “uniqueKey” which concatenates a student's name + ID.
This will not be passed as a parameter, you have to generate this inside constructor yourself and assign it to a “self.uniqueKey” variable.

Implement the following checks in the constructor:

- a. ID must have length of 8 digits. Otherwise print “Student ID must have 8 digits”
 - b. Name must have more than 1 word, and the words will be separated by space. Any single name will not be allowed. E.g: “Robin” is not allowed but “Robin Hood” and “Robin Hood Jr” is allowed. If any invalid name is given, print “Invalid name given”
4. Make **at least 4 objects of the Student class**, to demonstrate how your code works for each scenario. E.g:
 - a. One object to show it takes the mandatory parameters
 - b. One object to show it takes both mandatory + optional parameters
 - c. One object to show it does not take invalid ID
 - d. One object to show it does not take invalid name

5. Write a class Course. What should be the variables of Course class so **that it does not become specific to any particular student?** E.g: Saving the “grade achieved” or “semester completed” to a course object makes it specific to a particular student. Your “Course” class must have **at least 3 variables, and none of the variables should be student-specific. Write an appropriate constructor method.**

6. Write a AddStudent() method in your Course class so that it takes a Student Object, and semester completed as the parameters and saves it to an existing set of student objects. {Hint: use a dictionary to save the Student object and semester completed together}

E.g: if CSE111 is a **Course** object, and StudentA is a **Student** object, then the driver code will look like this:

CSE111.AddStudent(StudentA, “Fall2016”)

This code will add StudentA to CSE111’s student list, and will also save “Fall2016” as his semester completed.

6. Make 3 objects of your Course class.

7. In one of your Course objects, add the 2 valid student objects you made in Question 4 using the AddStudent() method.