Workshop 6

Designing classes (5%)

In this workshop, you will design and implement a python class to define the state and behavior of a class of students.

Learning outcomes

Upon successful completion of this workshop, you will have demonstrated the abilities:

1. Design a class and methods for representing a random data series
2. Implement the data series class and its methods to create an object of the class and get basic statistical measures.
3. Design a class and methods to represent Fibonacci series by extending the random data series.
4. Ensure data encapsulation when implementing classes.

Instructions

You are to design and implement python class that stores student’s state and methods to add, update and remove courses from a course data structure. Complete the python program given the following instructions.

docstring part

Fill in the docstring part for each of the class header and each of methods.

Class specifications

Create a python file named w6.py and place the following template code in that file. Implement the class student and the associated methods. A docstring for each function is provided to explain what each method is expected to return.

# -\*- coding: utf-8 -\*-

"""

**BDP200- winter 2020**

**Workshop 6**

**student name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Student id: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Total: /50pts**

**Weight: 5%**

"""

"""

File: student.py

class to manage a student's set of courses and grades.

"""

crsCredits={'BDA100': 1,

'BDP200': 2,

'BTP100': 3,

}

gradePoints= {'A+': 4,

'A': 4,

'B+': 3.5,

'B': 3,

'C+': 2.5,

'C': 2,

'D+': 1.5,

'D': 1,

'F': 0

}

**class Student(object):**

"""Represents a student."""

**def \_\_init\_\_(self, sid, name):** **#[1pts]**

"""initializes the instance variables with student Id sid and student name. """

self.\_sid =

self.\_name=

self.\_grades={} # holds crsCode:grade as key:val pair, initialized as empty

**def enroll(self, crsCode): #[4pts]**

"""returns 1 if crsCode is not in dict grades with adding

the crsCode as key and -1 as grade.

Returns 0 is crsCode is already a key in dict grades

"""

**def setScore(self, crsCode, grade): #[4pts]**

"""

returns 1 if crsCode in dict grades and sets the grade for this course

with the supppied argument grade.

otherwise it returns 0

"""

**def getScore(self, crsCode):** **#[4pts]**

"""

returns the grade for course crsCode

and -1 if crsCode not found in grades.

"""

**def getsid(self): # [1pt]**

""" returns the student id """

**def getName(self): # 1pt**

""" returns the student name """

**def calculateGPA(self):** **# [10pts]**

""" returns the Grade point average (GPA) for a set of courses

given the course credits and the letter grade and grade points associated

with letter grades. Here is an example of how it is calculated.

Examples

Course Credit Score Grade Points

Math 4 A+ 4 x 4.3 = 17.2

Physics 2 B 2 x 3 = 6

English 3 A 3 x 4 = 12

Total 9 NA 35.2

GPA 35.2 / 9 = 3.91

"""

**def \_\_str\_\_(self):** **#[10pts]**

"""

returns a string by concatenating the student name, followed by

the list of courses and the grades and at the end adds pthe GPA.

Within a string concatenation format a float value as follows:

+ "{0:.2f}".format(my\_float\_variable)+

"""

Caller code :

Define a main function right after the class Student definition. Implement the following within the main function:

Student object s1: [5pts]

* Create object s1 of class student with 1, "Adel" respectively for sid and name.
* invoke method enroll(), then setScore on s1 to assign letter grade ‘A’ for crsCode ‘BDA100’
* Invoke methods enroll(), then setScore() on s1 to assign letter grade ‘C’ for crsCode ‘BDP200’.

Student object s1: [5pts]

* Create object s2 of class student with 2, "Fadel" respectively for sid and name.
* invoke method enroll(), then setScore on student object s2 to assign letter grade ‘C’ for crsCode ‘BDA100’
* Invoke methods enroll(), then setScore() on student object s2 to assign letter grade ‘D’ for crsCode ‘BDP200’.

Create a list of student objects and print the student course, grades and the GPA. [5pts]

* Add student objects s1 and s2 to a list. Then loop through the list and call the print function by applying student objects as arguments.

Execution and Output Example

Name: Adel

Course Grade

BDA100 A

BDP200 C

GPA= 2.67

---------------

Name: Fadel

Course Grade

BDA100 C

BDP200 D

GPA= 1.33

---------------

SUBMISSION

Submit your python file w6.py on BB.