Columbia College
Vancouver, Canada

Introduction to Computer Science and Programming 1
CSCI120

Chapter11: OOP

Sample Coding

Note: This document has been designed and developed as part of an initiative for creating an OER (Open Education Resource) package for the course CSCI 120 at Columbia College.

Please contact Alireza.davoodi@gmail.com for any comment, modification, and questions.

Terms of use: Please feel free to customize this document as needed

Last Modified: July 2022

Requirements

- Please use meaningful name for your variables and functions
- Try to reuse your solutions as much as possible.

Problem0:

- Create a Cricle class and initialize it with radius. Make two instance methods getArea and getCircumference inside this class. Implement the methods.
- No need to write test class or main function for this.

Problem1:

- Create a Temprature class. Make two class/static methods :
  1. convertFahrenheit - It will take Celsius and will convert and print it into Fahrenheit.
  2. convertCelsius - It will take Fahrenheit and will convert it into Celsius.
- Write a test class for this one and test it (Look at problem 3 and 4 for example)

Problem2:

- Create a Student class and initialize it with coursesGrades, name and id number. Make methods to :
  1. Display - It should display name and id number of the student.
  2. AddCourseGrade – It adds the grades of the courses to the list of courses.
  3. getAverage – it calculates the average (GPA) of the student based on the grades of the student.
- No need to write test class or main function for this.

Problem3:

- Define a class with a static method that receives a list of numbers and calculate the average of the numbers.
- Add another static method the above class that receives a string and returns the reverse of it.

Problem4:

- Define and design a class that represents the time. (hours and minutes, AM and PM). Also add an instance method called addMinutes, that receives a number which represent a number (that represents a minute) and adds the time to current time the instance is

showing. For instance if the object represent 3:31 pm then you call addMinutes(30) then the new time is going to be 3:31 + 30 (minute) = 4:01 PM.

Good Luck ☺