KING SAUD UNIVERSITY COLLEGE OF COMPUTER AND INFORMATION SCIENCES COMPUTER SCIENCE DEPARTMENT		
CSC 111: Introduction to Programming with Java	Final Lab	1st Semester 1441/1442

Name (Arabic):…… ID:……..
Section#…… Serial#:……..
PC#:…… Lab#……

INSTRUCTIONS:

1- Create a **folder** on the **desktop** with your full name **firstname_lastname**
2- The first three lines of your code should specify your **full name, your ID and your lab section or time** as comments.
3- Use the same variable names in the UML, and meaningful names for other variables.
4- Reuse code whenever possible.
5- The duration of the exam: **2 hour 45 minutes**
6- Lab instructor email:

We want to implement a program to manage gym classes. The program should be able to create a gym class, register trainees in the class and display class information.

1. **Create a class GymClass according to the following UML:**

GymClass
- title : String - coach: String - capacity: int - day: String - traineesID: String[] + num: int
+ GymClass (String t, String c, int cap, String d) + addTrainee(String id):boolean +display(): void +isFull(): boolean + calculateIncome(): double +getters/setters as necessary

Where:

- **title:** The title of the class.
- **coach:** The name of the coach.
- **capacity:** The capacity of the class. The maximum number of trainees who can register in this class.
- **day:** The day of the class.
- **traineesID []**: An array that holds the IDs of trainees enrolled in the class.
- **num:** The number of trainees registered in the class. **Hint:** It points to the first empty location in the array *traineesID*. Every time a *trainee* is added successfully, **num** is incremented by 1.

- **GymClass (String t, String c, int cap, String d):** a non-default constructor to initialize attributes (*title,coach,capcity,day*) to given values.
    - The the size of the *traineesID* array should equal to the *capacity*.
    - The program should validate if *day* is a valid day. All week days are valid except "Friday", if it is entered by the user, "Wednesday" will be assigned to *day*.
    - Assume the user will enter correct day names.
- **addTrainee(String id): boolean:** The method adds an *id* of trainee to the first empty location in the array and returns true if the operation was completed successfully, and false otherwise.
    - The *id* is successfully added if the array is not full.
    - id format is:***CATEGORY_XXX*** where ***CATEGORY*** is: S, G or V and **XXX** are digits.
    - *note: assume user will enter correct format*
    - A validation step should be performed before to check if *id* is valid.
    - An ID is valid if it start with S , G or V.
    - Your program should ask the user to enter a valid *id* until a valid one is entered by the user.
- **calculateIncome(): double:** Calculate and return the total income of the class based on registered trainees. The prices are as follow:

Trainee	Price
S	100
G	120
V	170

- **isFull():boolean:** returns True when the class is full and False otherwise.
- **display():** displays the Class information in the following form:
  Class: <title in 30 col width, right justified>
  Coach: <coach in 35 col width, left justified>
  This class is full <if GymClass is full> **OR You can book this class** <if GymClass is not full>

2. Create an application class Gym that contains the following:

a. checkEnrollment(String traineeID): int

- This method takes as input a ***traineeID*** String. The method returns the location of *traineeID* in *traineesID* array or -1 if not found.

b. Create object *spinning* from *GymClass* with the following information.

Title	Coach	capacity	Day
Spinning	Hadeel	15	Sunday

c. In the main:

- Read the id of 3 *trainees* and add them to array of *trainees* in *spinning*
- Create another **GymClass object (call it** *spinning2)* , copy the following information from object *spinning : coach , capacity* . The title is spinning2 and day is *Wednesday*.
- Display the number of trainees enrolled in object *spinning.*
- Check if a *trainee* with the id "*G_111*" exists in object *spinning*. Display an appropriate message.
- Display a message indicating whether the object *spinning* achieved the target or not. The target is achieved if the total income from the class is equal to or more than 1000 SAR**.**
- Display the information of object *spinning2*.

Sample Run:

```
Add three trainee to gym class
Trainee 1
S_111
Trainee added successfully
Trainee 2
X_222
Invalid id , please enter a correct one
V_222
Trainee added successfully
Trainee 3
V_333
Trainee added successfully
Number of trainee in Spinning class = 3
Trainee G_111 is not enrolled
Class Spinning didn't hit target
Class :                     Spinning 2
Coach : Hadeel
You can book this class
```

Class Math	Class String
int/long/double/float **abs** (int/long/double/float x) long **round**(double x) double **ceil**(double x) double **floor**(double x) double **exp**(double x) double **log**(double x) double **log10**(double x) double **sqrt**(double x) double **pow**(double x, double y) int/long/double/float **min** (int/long/double/float x, int/long/double/float y) int/long/double/float **max** (int/long/double/float x, int/long/double/float y) double **sin**(double x) double **cos**(double x) double **tan**(double x)	int **length**() String **toLowerCase**() String **toUpperCase**() String **concat**(String str) String **replace** (char CharToBeReplaced, char CharReplacedWith) char **charAt**(int index) int **indexOf** (char ch) int **indexOf** (char ch, int pos) int **indexOf** (String str) int **indexOf** (String str, int pos) String **substring** (int beginIndex) String **substring** (int beginIndex, int endIndex) boolean **equals** (String str) int **compareTo** (String str)
	Class Character
	boolean **isDigit**(char ch) boolean **isLetter**(char ch) boolean **isLowerCase**(char ch) boolean **isUpperCase**(char ch) char **toLowerCase**(char ch) char **toUpperCase**(char ch)