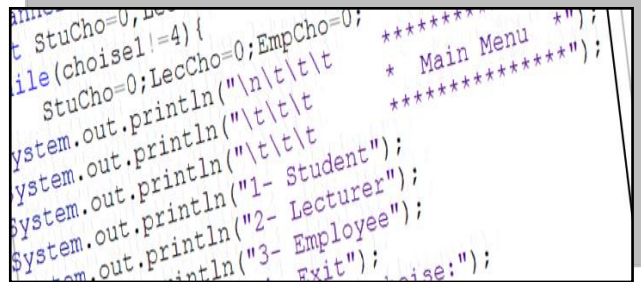


Due Date: 9⁰⁰ – 11⁰⁰ , 09th May 2017

Java Project

1. Create a **Person** class that should have the following protected data members (attributes): Name, Email & Description. Define a constructor with arguments (Name, Email) and also define getter methods for the attributes (Name, Email & Description)
2. Create a **Student** class that extends the Person class. As well as storing the students name and email, also store their course grade (e.g A, B, C) in a member variable. The grade should be accessible via a `getGrade` method. class student Overrides the `getDescription` methods. For the implementation of `getDescription` return a message along the lines of “The grade of student is”, substituting the student actual grade.
3. Create a **Lecturer** class that extends the Person class. This class should also store the subject that the lecturer teaches. Add a `getSubject` method, and implement `getDescription` so that it returns a suitable message, e.g. “Teaches Biology”.
4. Create a third class, **Employee** that extends the Person class. This should also store the name of the department the Employee works in (available via `getDepartment`). Again, `getDescription` should return a suitable message.



5. Create class **PersonViewer** as following:



```
public class PersonViewer
{
    public void view(Person person)
    {
        System.out.println("Person...");
        System.out.println("Name: " + person.getName());
        System.out.println("Email: " + person.getEmail());
        System.out.println("Description: " + person.getDescription());
        System.out.println("\n");
    }
}
```

6. Create a class called **PersonViewerTest**. Implement a main method that
 - a. Creates a **PersonViewer** object using the provided classes.
 - b. Creates instances of the **Lecturer**, **Employee** and **Student** classes and invokes the view method of the **PersonViewer** on each of them.
7. Create a subclass of the **PersonViewer** object that has the following modifications:
 - a. Overrides the view method, and uses the `instanceof` test to determine the actual type of the object (e.g. **Employee**), and then casts it appropriately.

- Create three methods called `viewPerson`, that vary by their input parameters. I.e. create one that accepts an `Employee` another that accepts a `Lecturer` and a third method that accepts a `Student`.
- Implement these methods so they write out to the console all information available about the objects. E.g. for an `Employee` write out its name, email and department.
- Finally add a main method to `EnhancedPersonViewer` that creates instances of each different type of person, and then invokes the `view` method on each.

<p>Main Menu</p> <p>1- Student 2- Lecturer 3- Employee 4- Exit</p> <p>Enter your Choice: 1</p> <p>A) Add Student C) Calculate Average D) Display Students R) Return the Menu</p>	<p>Enter your Choice: 2</p> <p>A) Add Lecturer D) Display Lecturers R) Return the Menu</p> <p>Enter your Choice: 3</p> <p>A) Add Employee C) Calculate yearly salary D) Display Employees R) Return the Menu</p> <p>Press (r / R) key to return to the main menu.</p>
--	--

Important Notes:

-  The Last day of submission is 9,10/5/2017 ⌚ 9 - 11
-  You must submit the following :
All source files + All class files. *Your work must be in a folder, its name is your name .*
- ⌚ If a student copies the Project of another student, he/she will be assigned a zero grade for the Project.
- ♂ ♀ Maximum & Minimum group project 2 Student.
- 📅 The date of discussion Lab 5 , 6 , 7

شعبة	التاريخ	الوقت	رقم المختبر
طالبات شعبة 1	9/5/2017	10 - 9	Lab 5 , 6
طالبات شعبة 2	9/5/2017	12 - 11	Lab 5 , 6
طلاب شعبة الاحد طلاب شعبة الثلاثاء	10/5/2017	9 - 8	Lab 5 , 6 ,7
طلاب شعبة الاثنين	10/5/2017	11 - 10	Lab 5 , 6 ,7
طلاب شعبة الاربعاء	10/5/2017	1 - 12	Lab 5 , 6 ,7



Web Site: WWW.PALINFONET.COM