Faculty of Computers & Information Technology

Programming language 1

جامعة الأقصى
AL-AQSA UNIVERSITY

School Management System project

Use OOP principles that you have learned so far to design and write a java program to School Management System.

Features :

- Add Employee (Principal/ Teacher) details.
- Add Student details.
- Add Subjects details.
- Display salaries of Employees.
- Finding the number of students registered in a particular subject
- Sava All details about Teachers, Students and Subjects to binary file with name "project.dat".

Classes of School Management System:

1. School class [that contains main method].
2. Employee class.
3. Principal class.
4. Teacher class.
5. Student class.
6. Subject class.

Explain the project

Employee class should contains:

1. Fields : Employee Name , Employee id, Employee Address , Employee phone number , Employee Email, Employee basicSalary, Employee liveExpensive.
2. Constructors: no-args constructor, constructor that takes all fields and store it in the class.
3. Mutators && Accessors methods
4. toString() to print Employee details.
5. Abstract method to calculate salary to employee. As name getSalary method.

Principal class that inherit Employee class should contains:

1. Fields : PrincipalBonus(علاوة المدير)
2. Constructors: no-args constructor, constructor that takes all fields and store it in the class.
3. Mutators && Accessors methods
4. toString() to print Principal details.
5. Calculate a total salary to Principal. As name getSalary method.

Teacher class that inherit Employee class should contains:

1. Fields : number of classes he teaches (classNo).
2. Constructors: no-args constructor, constructor that takes all fields and store it in the class and copy constructor .
3. Mutators && Accessors methods
4. toString() to print Teacher details.
5. calculate a total salary to Teacher. As name getSalary() method.

Notes :

- The living Expensive is only 10% of the basic salary
- The teacher's salary is the basic salary in addition to the living Expensive, and 20 dinars are added to it for every class he teaches.
- The Principal’s salary is the basic salary in addition to the living Expensive and the Principal’s Bonus .

Student class should contains:

1. Fields : Student id ,Student Name , Student level.
2. Constructors: no-args constructor, constructor that takes all fields and store it in the class and copy constructor .
3. Mutators && Accessors methods
4. toString() to print Student details.

Subject class should contains:

1. Fields : Subject Name , Teacher object , Student object.
2. Constructors: constructor that takes all fields and store it in the.
3. Mutators && Accessors methods
4. toString() to print Subject details.

School class [that contains main method].

1. Create arraylist OR array of Students to store object of students.
2. Create object of Principal.
3. Create arraylist OR array of Teacher to store object of teachers.
4. Create arraylist OR array of Subject to store object of subjects.

When the program runs, the following choices appear.

1- Add Employee
2- Add Student.
3- Add Subject.
4- Show Subjects.
5- Show Employees Salaries.
6- Count of student in any subject.
7- Save data in file
8- Exit.

Enter your choice :

Notes :

- When choose 1 or 5 show display menu to choose 1. Principal 2. Teacher .
- When choose 3 to add Subject:

- Display names of exists Teachers and students.
- should choose object of student and object of Teacher form exiting objects only.(Pass any object of student and object of teacher that does not exist. The course is not added)

- when choose 6 ask user to enter subject to know How many students are registered in this subject?

- When choose 7 All Students details , All Teachers details and All Subjects details(Subject Name , Teacher Name ,Student Name) store in binary file.
- Exit of project when choose 8 only
- When you run the project again It is read all the objects stored in the file and deal with them

Important Notes:

- This is a SOLO project, means that any two copies will result in a ZERO

- Each student is required to submit :
  - UML Diagram.
  - Project [all the project folder with the file inside should be compressed and submitted as .zip or .rar].
  -
- Write your triple name and your ID university in each class in first line and last line. (name and id as a note in each class)

- Each student will be discussed in the project, where 30% of the project's grade will be calculated on Discussion.

وفقكم الله وأنار دربكم