ASSIGNMENT-II

OOP

DeadLine 06-12-2023

You are required to implement Employee Management System. There will be an employee class having following attributes:

Name(String), Father_Name(String), Emp_id(int), dob(date), bps(1-22), job_Type(Private, public or autonomous)

It should have two constructors i.e., default constructor and an overloaded constructor. The attributes in the constructors must be initialized through initializer list. The Employee class should be having following functions i.e., setRecord and showRecord. In the setEmployeeRecord, the data input should be taken from the user while showRecord should print the employee details.

There should a class of Teacher inherited from Employee class and having following attributes i.e.,

Education(String) whose values can be Bachelor, Masters, PhD.

Teaching_level whose values can be school, college, university.

Subject which can be physics, chemistry, maths and cs

The class must have following functions along with its contructors i.e.,

setRecord, showRecord and upDateRecord.

The setRecord function should take input from user while showRecord should display teacher's attributes. The upDateRecord should only be able to update job_type, bps, education and teaching_level attributes.

There should be a class Doctor having following attributes:

Doctor_type can have following values i.e., mbbs, bds and vetnery.

Specialization can be ent, heart, stomach and gyne.

Job_level can be house job, medical officer and senior medical officer.

The class must have following functions along with its contructors i.e.,

setRecord, showRecord and upDateRecord.

The setRecord function should take input from user while showRecord should display teacher's attributes. The upDateRecord should only be able to update job_type, bps and job_level attributes.

Note: You may add more functions as per your requirements.

You need to implement two link lists i.e., one for doctor and other for teacher. To achieve this you may have two classes for the nodes of doctor and teacher i.e.,

TeacherNode and Doctor Node. The TeacherNode can have two datamembers Teacher object and Teacher pointer next node. The same may be implemented for DoctorNode.

You should have EmployeeList class where both the link lists should be created.

You may have attributes as per your requirement.

The EmployeeList should have following functions.

AddNewDoctor

AddNewTeacher

updateEmployeeRecord

SearchEmployeeByID

SearchEmployeeByName

SearchYoungestEmployee

SearchEldestEmployee

SearchDoctorBySpecialization

SearchTeacherByTeachingLevel

DeleteEmployeeByID

DeleteEmployeeByName

In the main function, the user must be given choices and as per user choice action should be performed. The choices are

1) AddEmployee:

When user selects this choice he must be asked whether doctor or teacher should be added. It means a new node in doctor or teacher linklist should be created and their data should be taken from user(setRecord).

2) updateEmployeeRecord

When user selects this choice he must be asked to enter the id of the employee. The id must be searched in both the lists. The list in which the id is found, the node with the given id must be updated.

3) SearchEmployee

When user selects this choice he should be asked to give further choices:

I. SearchEmployeeByID
II. SearchEmployeeByName
III. SearchYoungestEmployee
IV. SearchEldestEmployee
V. SearchDoctorBySpecialization
VI. SearchTeacherByTeachingLevel

When user selects SearchEmployeeByID he must be asked to enter the id of the employee, both the list should be searched for the given id. If found, the record must displayed through corresponding showRecord function.

When user selects SearchEmployeeByName he must be asked to enter the name of the employee, both the list should be searched for the given name. All the doctors or/and teachers with the given name must be displayed.

When user selects SearchYoungestEmployee, The record of the employee with the youngest of the age from both the lists should be displayed.

When user selects SearchEldestEmployee, The record of the employee with the eldest of the age from both the lists should be displayed.

When the user selects SearchDoctorBySpecialization, he must be asked to enter the specialization field. All the details of the doctors having said specialization field must be displayed.

When the user selects SearchTeacherByTeachingLevel, he must be asked to enter the teaching level. All the details of the doctors having said teaching level must be displayed.

4) DeleteEmployee

When user selects this choice, he must be asked to select

I. DeleteEmployeeByID
II. DeleteEmployeeByName

When user selects DeleteEmployeeByID he must be asked to enter the id of the employee. The id must be searched in both the lists. The node from the searched list should be deleted.

When user selects DeleteEmployeeByName he must be asked to enter the name of the employee, both the list should be searched for the given name. All the doctors or/and teachers with the given name must be deleted from the corresponding lists.

**********GOOD LUCK**********