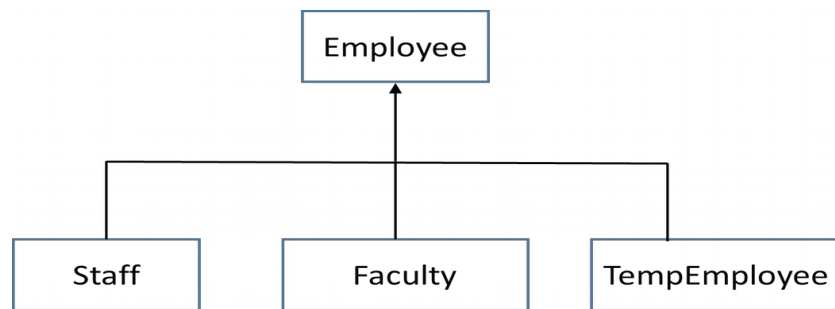


CS 202: IT Workshop I

Assignment IV (GG3)



Employee class:

Employee class will have following attributes: name, dateOfBirth, empID, dateOfJoining, etc.

Staff class:

Staff class will have following attributes: sectionName, designation, etc.

Staff will have a method that will check whether designation of two staffs are same or not. [**Hint:** pass object as parameter to the method]

Faculty class:

Faculty class will have departmentName, noOfPublications, etc.

Faculty will have a method that will check whether department of two faculties are same or not. [**Hint:** pass object as parameter to the method]

TempEmployee class:

TempEmployee class will have sectionName, lastWorkingDay, etc.

TempEmployee will have a method that will display his/her total number of working days.

a) Implement the above scenario. You may assume additional instance variables, additional methods, static fields, etc. if needed / to make the application more realistic. Variable type and method return may be assumed appropriately.

b) Override a method setDetails() to take values to the attributes from keyboard.

c) Override another method showDetails(String joiningMonth) to display all the information of all the employees who has joined in a particular month.

d) Every class should support a parameterized constructor that will set the values to the instance variables of appropriate objects.

e) Create a public class and design a nested menu-driven interface to i) create objects of different classes (Staff, Faculty, TempEmployee), ii) display details (based on empID, based on joiningYear), iii) setDetails, etc.

Hint: To handle date related things (e.g. `dateOfBirth`, `dateOfJoining`), you may use additional packages supported by Java.