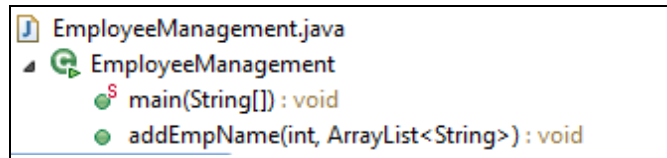


Refer below sample problem with solution to solve remaining problems. Ensure same approach is followed. Use exactly same class name and method names as specified in the outline (image) provided along with the problem statement. Any mistake on these instructions may result into invalid submission of the assignments even if logic is 100% correct.

1. Write a program to store the names of the employees present in an organization. You can use an ArrayList to store the names of all the employees.



```
import java.util.ArrayList;
import java.util.Scanner;

public class EmployeeManagement {

    public static void main(String[] args)
    {
        ArrayList<String> empNameList = new ArrayList<String>();

        Scanner sc = new Scanner(System.in);

        System.out.println("How many names do you want to add to the system?");
        int num = sc.nextInt();

        EmployeeManagement empMgmt = new EmployeeManagement();
        empMgmt.addEmpName(num, empNameList);
    }

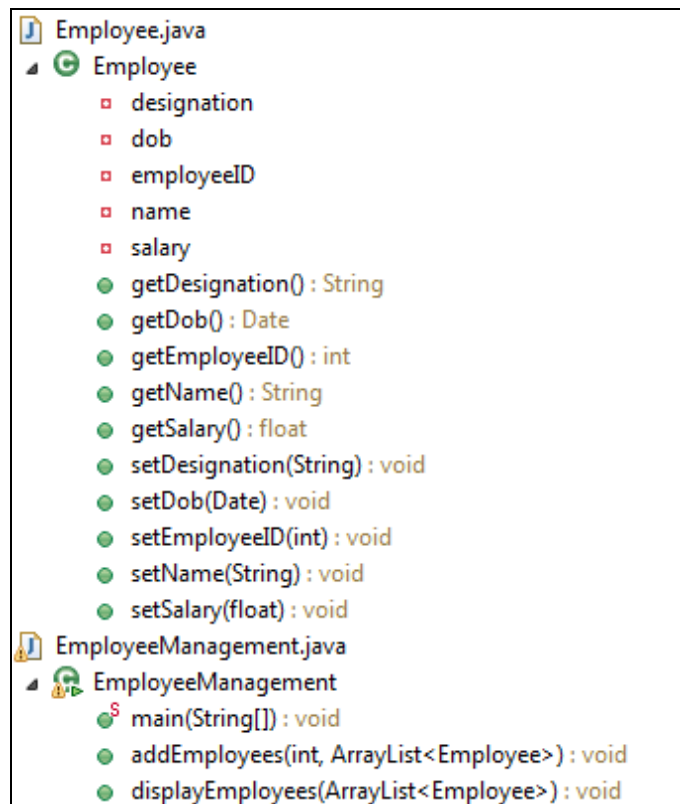
    public void addEmpName(int numEmp, ArrayList<String> empList)
    {
        Scanner sc = new Scanner(System.in);

        for(int i = 1; i<=numEmp; i++)
        {
            System.out.println("Enter the name of the employee");
            String name = sc.nextLine();
            empList.add(name);
        }
    }
}
```

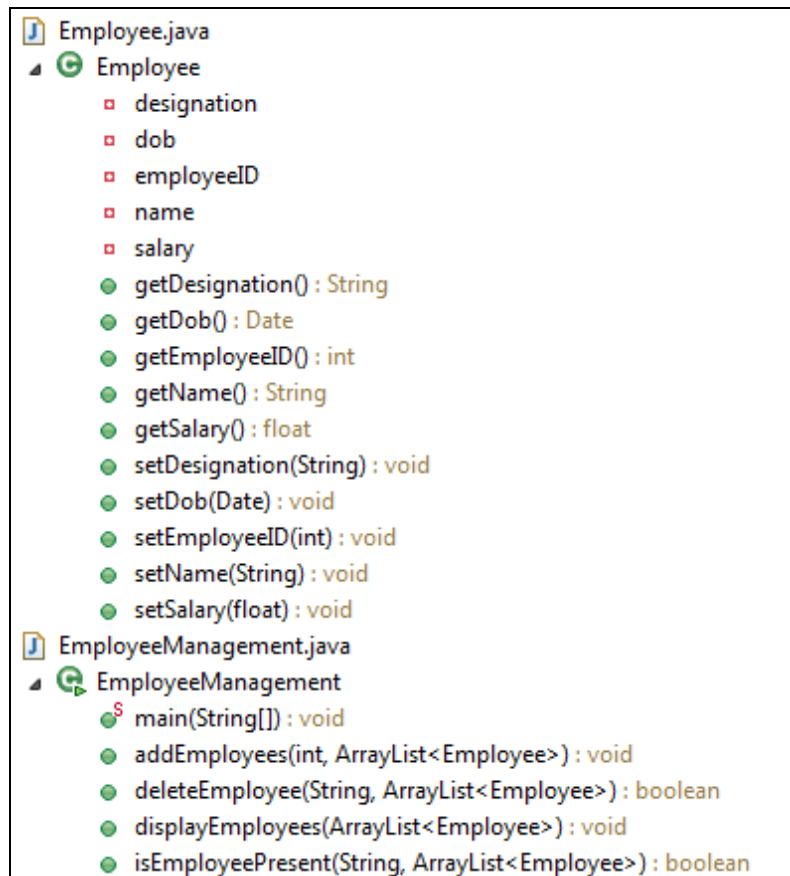
Sample Output:

```
How many names do you want to add to the system?  
2  
Enter the name of the employee  
Saurabh  
Enter the name of the employee  
Sumit
```

2. Write a program to store all the details of the employees present in an organization. The details of the employee include employee id, name, date of birth, salary and designation of the employee. You can use an ArrayList to store the names of all the employees. After adding the details of the employees also add a feature to print the details of all the employees that were added. The printing of details should be done with the use of for each loop and also by using an iterator.

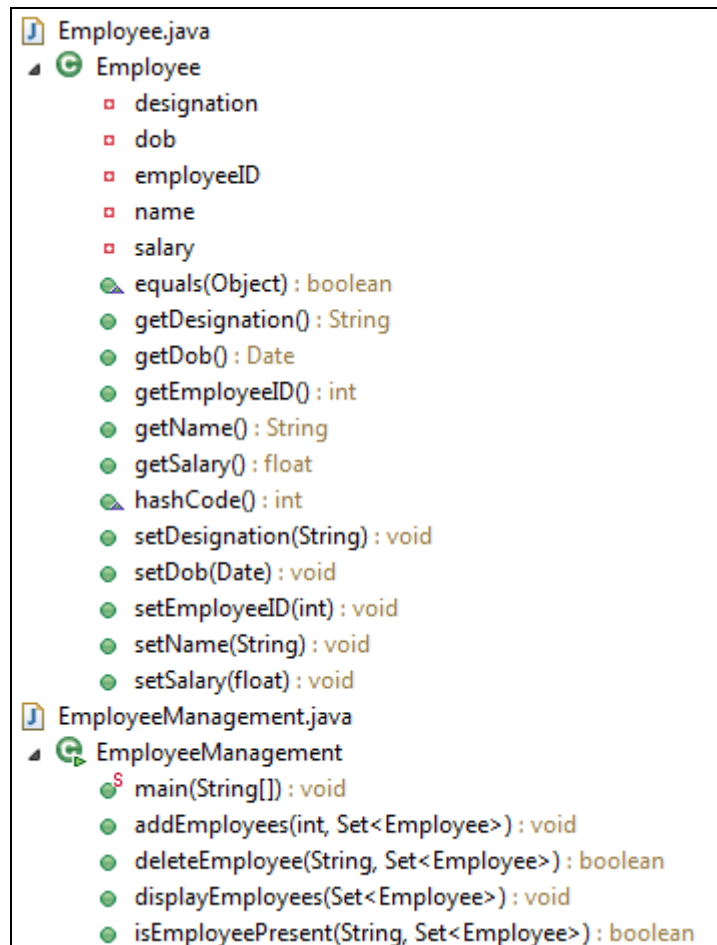


3. Modify the above program such that the end user is given a privilege to check whether and employee exists or not and delete an employee by entering the name of the employee.



4. Modify the above program such that if the end user tries to add a duplicate employee, then he is not allowed to do so. All other operations remain same.

Note: If name of two employees are same then it should be considered as duplicate



5. Using a Map write a program to map the details of an employee with the hobbies of the employee. The program should take the details of the employee and his/her hobbies as input and then store them as key/value pair in the map. Once the addition of details is complete, then the user should be able to list down all the employees, along with their hobbies and also should be able to view a particular employee's (by giving employee id as input), his/her details and his/her hobbies . Also the user should be able to delete an employee based on the id of the employee.

