Main.java

```
package LAB_Problem.Problem12;
import java.util.ArrayList;
/**
* Program Name: Main
* Description:
* Author : Keshav Abhishek
* Created On: 22-02-2025
* Organization: C.V. Raman Global University
* ______
* Copyright (c) 2025, All rights reserved.
* ______
*/
class Student {
  String studentId;
  String studentName;
  String studentDept;
  double CGPA;
  public Student() {
  public Student(String studentId, String studentName, String studentDept,
double CGPA) {
    this.studentId = studentId;
    this.studentName = studentName;
    this.studentDept = studentDept;
    this.CGPA = CGPA;
  }
  public String getStudentId() {
    return studentId;
  }
  public void setStudentId(String studentId) {
    this.studentId = studentId;
  }
  public String getStudentName() {
    return studentName;
  }
```

```
public void setStudentName(String studentName) {
    this.studentName = studentName;
  }
  public String getstudentDept() {
    return studentDept;
  }
  public void setstudentDept(String studentDept) {
    this.studentDept = studentDept;
  }
  public double getCGPA() {
    return CGPA;
  public void setCGPA(double cGPA) {
    CGPA = cGPA;
  }
  @Override
  public String toString() {
    return "Student ID: " + studentId + "\n" + "Student Name: " +
studentName + "\n" + "Student Dept: "
         + studentDept + "\n" + "CGPA: " + CGPA + "\n";
}
class StudentSystem {
  ArrayList<Student> students;
  StudentSystem() {
    students = new ArrayList<>();
  }
  public void addStudent(Student stud) {
    students.add(stud);
    System.out.println("Student added: " + stud.getStudentId());
  }
  public void removeViaID(String ID) {
    int prev = students.size();
    students.removeIf(n -> {
       if(ID.equalsIgnoreCase(n.getStudentId())){
         System.out.println("\nDeleted: ");
         System.out.println(n);
         return true;
```

```
return false;
     });
    int curr = students.size();
    if (prev == curr) {
       System.out.println("Invalid ID " + ID);
     }
  }
  public void searchViaName(String Name) {
    boolean flag = false;
    for (int i = 0; i < students.size(); i++) {
       if (Name.equalsIgnoreCase(students.get(i).getStudentName())) {
         System.out.println(students.get(i));
         flag = true;
       }
     }
    if(!flag){
       System.out.printf("\nStudent named with %s doesn't exists...\n",
Name);
  }
  public void displayAll() {
    for (int i = 0; i < students.size(); i++) {
       System.out.println(students.get(i));
     }
  }
}
public class Main {
  public static void main(String[] args) {
    StudentSystem school = new StudentSystem();
    school.addStudent(new Student("S001", "Ramesh", "CSE", 8.9));
    school.addStudent(new Student("S002", "Suresh", "ECE", 8.5));
    school.addStudent(new Student("S003", "Mahesh", "MECH", 8.7));
    school.addStudent(new Student("S004", "Rajesh", "CIVIL", 8.6));
    school.addStudent(new Student("S005", "Naresh", "EEE", 8.8));
    System.out.println("\n\nRemove Via ID");
    school.removeViaID("S005");
    school.removeViaID("S006");
    System.out.println("\n\nSearch Via Name");
    school.searchViaName("Mahesh");
    school.searchViaName("Kalesh");
```

```
System.out.println("\n\nDisplay all students");
    school.displayAll();
}
```

Main2.java

```
package LAB_Problem.Problem12;
import java.util.ArrayList;
/**
* Program Name: Employee Management System
* Description : Manage employee records, including adding, removing,
         and displaying employees with specific conditions.
* Author
           : Keshav Abhishek
* Created On: 22-02-2025
* Organization: C.V. Raman Global University
* ______
* Copyright (c) 2025, All rights reserved.
*/
class EmployeeDetails {
  int empId;
  String empName;
  String empDesg;
  double empSalary;
  String empDept;
  public EmployeeDetails(int empId, String empName, String empDesg, double
empSalary, String empDept) {
    this.empId = empId;
    this.empName = empName;
    this.empDesg = empDesg;
    this.empSalary = empSalary;
    this.empDept = empDept;
  }
  public int getEmpId() {
    return empId;
  }
  public String getEmpName() {
    return empName;
  }
  public String getEmpDesg() {
    return empDesg;
  }
```

```
public double getEmpSalary() {
    return empSalary;
  }
  public String getEmpDept() {
    return empDept;
  }
  @Override
  public String toString() {
    return "Employee ID: " + empId + "\n" +
         "Employee Name: " + empName + "\n" +
         "Designation: " + empDesg + "\n" +
         "Salary: " + empSalary + "\n" +
        "Department: " + empDept + "\n";
  }
}
class EmployeeDatabase {
  ArrayList<EmployeeDetails> employees;
  public EmployeeDatabase() {
    employees = new ArrayList<>();
  }
  public void addEmployee(EmployeeDetails emp) {
    employees.add(emp);
    System.out.println("Employee added: " + emp.getEmpName());
  }
  public void removeEmployeeById(int empId) {
    boolean removed = employees.removeIf(emp -> {
       if (emp.getEmpId() == empId) {
         System.out.println("\nDeleted Employee: \n" + emp);
         return true;
       }
       return false;
    });
    if (!removed) {
       System.out.println("Invalid Employee ID: " + empId);
    }
  }
  public void displayHighSalaryEmployees() {
    System.out.println("\nEmployees with salary > 50000 in CSE Department:
");
```

```
boolean found = false:
    for (EmployeeDetails emp : employees) {
      if (emp.getEmpSalary() > 50000 && "CSE".equalsIgnoreCase(emp.
getEmpDept())) {
         System.out.println(emp);
         found = true;
       }
    }
    if (!found) {
      System.out.println("No employees found with salary > 50000 in CSE
Department.");
    }
  }
  public void displayAllEmployees() {
    System.out.println("\nDisplaying All Employees:");
    for (EmployeeDetails emp : employees) {
      System.out.println(emp);
    }
  }
}
public class Main2 {
  public static void main(String[] args) {
    EmployeeDatabase company = new EmployeeDatabase();
    company.addEmployee(new EmployeeDetails(101, "Arjun", "Software
Engineer", 60000, "CSE"));
    company.addEmployee(new EmployeeDetails(102, "Basant", "Data Scientist",
70000, "CSE"));
    company.addEmployee(new EmployeeDetails(103, "ChandaSuri", "HR Manager",
48000, "HR"));
    company.addEmployee(new EmployeeDetails(104, "Dharmendra", "Network
Engineer", 55000, "IT"));
    company.addEmployee(new EmployeeDetails(105, "Eshan", "Professor",
52000, "CSE"));
    System.out.println("\nRemoving an Employee:");
    company.removeEmployeeById(104);
    company.removeEmployeeById(110);
    System.out.println("\nDisplaying High Salary Employees in CSE:");
    company.displayHighSalaryEmployees();
    System.out.println("\nDisplaying All Employees:");
    company.displayAllEmployees();
  }
```

Sourav.java

```
package LAB_Problem.Problem12;
import java.util.ArrayList;
class Student {
  String studentId;
  String studentName;
  String studentDept;
  double CGPA;
  public Student() {
  public Student(String studentId, String studentName, String studentDept,
double CGPA) {
    this.studentId = studentId;
    this.studentName = studentName;
    this.studentDept = studentDept;
    this.CGPA = CGPA;
  }
  public String getStudentId() {
    return studentId;
  }
  public void setStudentId(String studentId) {
    this.studentId = studentId;
  }
  public String getStudentName() {
    return studentName;
  }
  public void setStudentName(String studentName) {
    this.studentName = studentName;
  }
  public String getStudentDept() {
    return studentDept;
  }
  public void setStudentDept(String studentDept) {
    this.studentDept = studentDept;
  }
```

```
public double getCGPA() {
    return CGPA;
  }
  public void setCGPA(double CGPA) {
    this.CGPA = CGPA;
  }
  @Override
  public String toString() {
    return "Student ID: " + studentId + "\n" + "Student Name: " +
studentName + "\n" + "Student Dept: "
         + studentDept + "\n" + "CGPA: " + CGPA + "\n";
class StudentSystem {
  ArrayList<Student> students;
  StudentSystem() {
    students = new ArrayList<>();
  }
  public void addStudent(Student stud) {
    students.add(stud);
    System.out.println("Student added: " + stud.getStudentId());
  }
  public void removeViaID(String ID) {
    int prev = students.size();
    students.removeIf(n -> {
       if(ID.equalsIgnoreCase(n.getStudentId())){
         System.out.println("\nDeleted: ");
         System.out.println(n);
         return true;
       return false;
    int curr = students.size();
    if (prev == curr) {
       System.out.println("Invalid ID " + ID);
     }
  }
  public void searchViaName(String Name) {
    boolean flag = false;
```

```
for (int i = 0; i < students.size(); i++) {
       if (Name.equalsIgnoreCase(students.get(i).getStudentName())) {
         System.out.println(students.get(i));
         flag = true;
       }
     }
    if(!flag){
       System.out.printf("\nStudent named with %s doesn't exists...\n",
Name);
  }
  public void displayAll() {
    for (int i = 0; i < students.size(); i++) {
       System.out.println(students.get(i));
    }
  }
}
public class Sourav {
  public static void main(String[] args) {
    StudentSystem school = new StudentSystem();
    school.addStudent(new Student("STD101", "Amit", "CSE", 9.1));
    school.addStudent(new Student("STD102", "Sumit", "ECE", 8.4));
    school.addStudent(new Student("STD103", "Ankit", "MECH", 8.8));
    school.addStudent(new Student("STD104", "Rohit", "CIVIL", 8.5));
    school.addStudent(new Student("STD105", "Mohit", "EEE", 8.9));
    System.out.println("\n\nRemove Via ID");
    school.removeViaID("STD105");
    school.removeViaID("STD106");
    System.out.println("\n\nSearch Via Name");
    school.searchViaName("Ankit");
    school.searchViaName("Vikas");
    System.out.println("\n\nDisplay all students");
    school.displayAll();
  }
}
```

Sourav2.java

```
package LAB_Problem.Problem12;
import java.util.ArrayList;
class EmployeeDetails {
  int empId;
  String empName;
  String empDesg;
  double empSalary;
  String empDept;
  public EmployeeDetails(int empId, String empName, String empDesg, double
empSalary, String empDept) {
    this.empId = empId;
    this.empName = empName;
    this.empDesg = empDesg;
    this.empSalary = empSalary;
    this.empDept = empDept;
  }
  public int getEmpId() {
    return empId;
  }
  public String getEmpName() {
    return empName;
  }
  public String getEmpDesg() {
    return empDesg;
  }
  public double getEmpSalary() {
    return empSalary;
  }
  public String getEmpDept() {
    return empDept;
  }
  @Override
  public String toString() {
    return "Employee ID: " + \text{ empId} + \text{"} \text{"} +
         "Employee Name: " + empName + "\n" +
```

```
"Designation: " + empDesg + "\n" +
         "Salary: " + empSalary + "\n" +
         "Department: " + empDept + "\n";
  }
}
class EmployeeDatabase {
  ArrayList<EmployeeDetails> employees;
  public EmployeeDatabase() {
    employees = new ArrayList<>();
  }
  public void addEmployee(EmployeeDetails emp) {
    employees.add(emp);
    System.out.println("Employee added: " + emp.getEmpName());
  }
  public void removeEmployeeById(int empId) {
    boolean removed = employees.removeIf(emp -> {
       if (emp.getEmpId() == empId) {
         System.out.println("\nDeleted Employee: \n" + emp);
         return true;
       return false;
    });
    if (!removed) {
       System.out.println("Invalid Employee ID: " + empId);
    }
  }
  public void displayHighSalaryEmployees() {
    System.out.println("\nEmployees with salary > 50000 in CSE Department:
");
    boolean found = false;
    for (EmployeeDetails emp : employees) {
       if (emp.getEmpSalary() > 50000 && "CSE".equalsIgnoreCase(emp.
getEmpDept())) {
         System.out.println(emp);
         found = true;
       }
    }
    if (!found) {
       System.out.println("No employees found with salary > 50000 in CSE
Department.");
     }
```

```
}
  public void displayAllEmployees() {
    System.out.println("\nDisplaying All Employees:");
    for (EmployeeDetails emp : employees) {
      System.out.println(emp);
    }
}
public class Sourav2 {
  public static void main(String[] args) {
    EmployeeDatabase company = new EmployeeDatabase();
    company.addEmployee(new EmployeeDetails(101, "Arnaud", "Software
Engineer", 60000, "CSE"));
    company.addEmployee(new EmployeeDetails(102, "Bastien", "Data
Scientist", 70000, "CSE"));
    company.addEmployee(new EmployeeDetails(103, "Chantal", "HR Manager",
48000, "HR"));
    company.addEmployee(new EmployeeDetails(104, "Didier", "Network
Engineer", 55000, "IT"));
    company.addEmployee(new EmployeeDetails(105, "Etienne", "Professor",
52000, "CSE"));
    System.out.println("\nRemoving an Employee:");
    company.removeEmployeeById(104);
    company.removeEmployeeById(110);
    System.out.println("\nDisplaying High Salary Employees in CSE:");
    company.displayHighSalaryEmployees();
    System.out.println("\nDisplaying All Employees:");
    company.displayAllEmployees();
  }
}
```