Experiment 4

Name: Mayank Sharma **UID: 22BCS16886**
Branch: BE-CSE **Section: 22BCS_IOT_EPAM_801-B**
Semester:6th **Date of Performance: 17 Feb 2025**
Subject Name: Project Based Learning in Java with Lab **Subject Code: 22CSH-359**

1. **Aim:** Write a Program to perform the basic operations like insert, delete, display and search in list. List contains String object items where these operations are to be performed.

2. **Algorithm:**

Initialize List: Create an `ArrayList<Employee>` to store employee records.

Display Menu:

- Insert a new employee.
- Delete an employee by ID.
- Display all employees.
- Search for an employee by ID.
- Exit the program.

Insert Employee:

- Read employee details from the user.
- Create an `Employee` object and add it to the list.

Delete Employee:

- Search for the employee by ID.
- If found, remove from the list.
- If not found, display an appropriate message.

Display Employees:

- Print all employees with their details.

Search Employee:

- Search for an employee by ID.
- If found, display the employee's details.
- If not found, display a message.

Exit Program:

- Close the scanner and terminate.

3. Implementation/Code:

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

class Employee {
   int empNo;
   String empName;
   String department;
   char designCode;
   int basic;
   int hra;
   int it;

   Employee(int empNo, String empName, String department, char designCode,
int basic, int hra, int it) {
       this.empNo = empNo;
       this.empName = empName;
       this.department = department;
       this.designCode = designCode;
       this.basic = basic;
       this.hra = hra;
       this.it = it;
   }
}

public class problem1 {
   private static List<Employee> employees = new ArrayList<>();
   private static Scanner scanner = new Scanner(System.in);

   public static void main(String[] args) {
       // Initial Employee List
       employees.add(new Employee(1001, "Ashish", "R&D", 'e', 20000, 8000,
3000));
       employees.add(new Employee(1002, "Sushma", "PM", 'c', 30000, 12000,
9000));
       employees.add(new Employee(1003, "Rahul", "Acct", 'k', 10000, 8000,
1000));
```

```
        employees.add(new Employee(1004, "Chahat", "Front Desk", 'r', 12000,
6000, 2000));
        employees.add(new Employee(1005, "Ranjan", "Engg", 'm', 50000, 20000,
20000));
        employees.add(new Employee(1006, "Suman", "Manufacturing", 'e',
23000, 9000, 4400));
        employees.add(new Employee(1007, "Tanmay", "PM", 'c', 29000, 12000,
10000));

        while (true) {
            System.out.println("\nEmployee Management System");
            System.out.println("1. Insert Employee");
            System.out.println("2. Delete Employee");
            System.out.println("3. Display All Employees");
            System.out.println("4. Search Employee by ID");
            System.out.println("5. Exit");
            System.out.print("Enter your choice: ");

            int choice = scanner.nextInt();
            scanner.nextLine(); // Consume newline

            switch (choice) {
                case 1 -> insertEmployee();
                case 2 -> deleteEmployee();
                case 3 -> displayEmployees();
                case 4 -> searchEmployee();
                case 5 -> {
                    System.out.println("Exiting program...");
                    scanner.close();
                    return;
                }
                default -> System.out.println("Invalid choice! Please try
again.");
            }
        }
    }

    private static void insertEmployee() {
        System.out.print("Enter Employee ID: ");
        int empNo = scanner.nextInt();
        scanner.nextLine();

        System.out.print("Enter Employee Name: ");
        String empName = scanner.nextLine();

        System.out.print("Enter Department: ");
        String department = scanner.nextLine();

        System.out.print("Enter Designation Code (e/c/k/r/m): ");
        char designCode = scanner.next().charAt(0);

        System.out.print("Enter Basic Salary: ");
```

```
        int basic = scanner.nextInt();

        System.out.print("Enter HRA: ");
        int hra = scanner.nextInt();

        System.out.print("Enter IT: ");
        int it = scanner.nextInt();

        employees.add(new Employee(empNo, empName, department, designCode,
basic, hra, it));
        System.out.println("Employee added successfully!");
    }

    private static void deleteEmployee() {
        System.out.print("Enter Employee ID to delete: ");
        int empId = scanner.nextInt();

        boolean removed = employees.removeIf(e -> e.empNo == empId);

        if (removed) {
            System.out.println("Employee with ID " + empId + " deleted
successfully.");
        } else {
            System.out.println("No employee found with ID " + empId);
        }
    }

    private static void displayEmployees() {
        if (employees.isEmpty()) {
            System.out.println("No employees in the list.");
            return;
        }

        System.out.printf("%-10s %-15s %-15s %-15s %s%n", "Emp No.", "Emp
Name", "Department", "Designation", "Salary");
        for (Employee e : employees) {
            String designation = getDesignation(e.designCode);
            int salary = e.basic + e.hra + getDA(e.designCode) - e.it;
            System.out.printf("%-10d %-15s %-15s %-15s %d%n", e.empNo,
e.empName, e.department, designation, salary);
        }
    }

    private static void searchEmployee() {
        System.out.print("Enter Employee ID to search: ");
        int empId = scanner.nextInt();

        for (Employee e : employees) {
            if (e.empNo == empId) {
                String designation = getDesignation(e.designCode);
                int salary = e.basic + e.hra + getDA(e.designCode) - e.it;
```

```
                System.out.printf("%-10s %-15s %-15s %-15s %s%n", "Emp No.",
"Emp Name", "Department", "Designation", "Salary");
                System.out.printf("%-10d %-15s %-15s %-15s %d%n", e.empNo,
e.empName, e.department, designation, salary);
                return;
            }
        }
        System.out.println("No employee found with ID: " + empId);
    }

    private static String getDesignation(char code) {
        return switch (code) {
            case 'e' -> "Engineer";
            case 'c' -> "Consultant";
            case 'k' -> "Clerk";
            case 'r' -> "Receptionist";
            case 'm' -> "Manager";
            default -> "Unknown";
        };
    }

    private static int getDA(char code) {
        return switch (code) {
            case 'e' -> 20000;
            case 'c' -> 32000;
            case 'k' -> 12000;
            case 'r' -> 15000;
            case 'm' -> 40000;
            default -> 0;
        };
    }
}
```

4. Output:

```
Employee Management System
1. Insert Employee
2. Delete Employee
3. Display All Employees
4. Search Employee by ID
5. Exit
Enter your choice: 3
Emp No.     Emp Name        Department      Designation     Salary
1001        Ashish          R&D             Engineer        45000
1002        Sushma          PM              Consultant      65000
1003        Rahul           Acct            Clerk           29000
1004        Chahat          Front Desk      Receptionist    31000
1005        Ranjan          Engg            Manager         90000
1006        Suman           Manufacturing   Engineer        47600
1007        Tanmay          PM              Consultant      63000
```

5. **Time Complexity: O(n)**

6. **Space Complexity: O(1)**

7. **Learning Outcomes:**

   i. Gain practical knowledge of fundamental list operations, including **insertion, deletion, search, and display** in Java.
   ii. Learn how to use the **ArrayList** class and its built-in methods (`add()`, `remove()`, `contains()`, etc.) for dynamic data handling.
   iii. Develop logical thinking by implementing a **menu-driven program** that interacts with the user to perform various list operations.