Experiment 4

Name: Rohit Kumar
UID: 22BCS16160
Branch: BE-CSE
Section: EPAM-801-B
Semester:6th
Date : 17/02/2025
Subject: PBLJ
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. Objective:

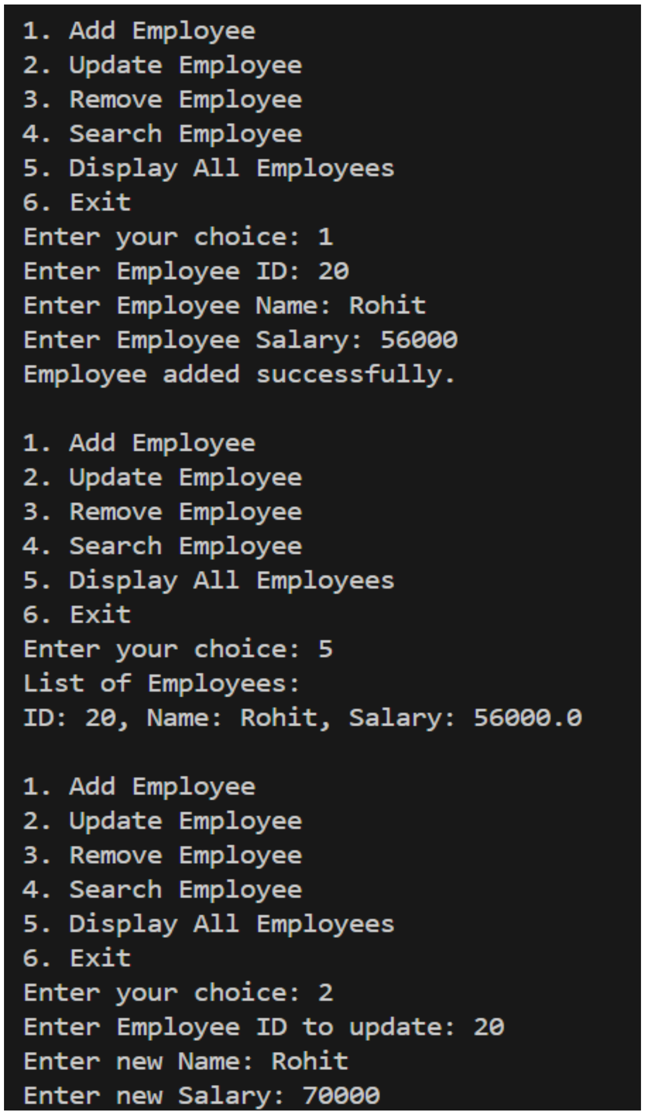
The objective of this program is to perform basic operations (insert, search, delete, display) on a list of strings using Java. It demonstrates the use of ArrayList, user input handling, and control structures for efficient data management

3. Implementation/Code:

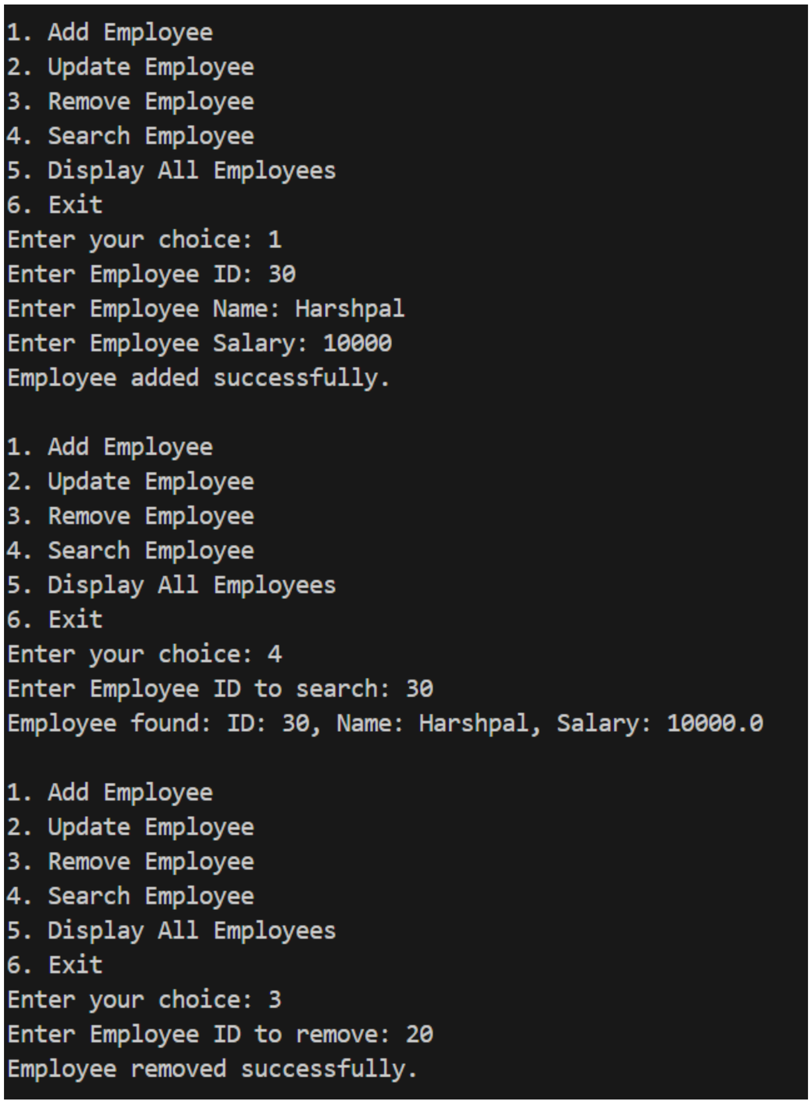
```
import java.util.*;

class Employee {
    private int id;
    private String name;
    private double salary;

    public Employee(int id, String name, double salary) {
        this.id = id;
        this.name = name;
        this.salary = salary;
    }
    public int getId() {
        return id;
    }
    public String getName() {
        return name;
    }
    public double getSalary() {
        return salary;
    }
    public void setName(String name) {
        this.name = name;
    }
    public void setSalary(double salary) {
        this.salary = salary;
    }
    public String toString() {
        return "ID: " + id + ", Name: " + name + ", Salary: " + salary;
    }
}

public class EmployeeManagementSystem {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        List<Employee> employeeList = new ArrayList<>();

        while (true) {
            System.out.println("\n1. Add Employee");
            System.out.println("2. Update Employee");
            System.out.println("3. Remove Employee");
            System.out.println("4. Search Employee");
            System.out.println("5. Display All Employees");
            System.out.println("6. Exit");
            System.out.print("Enter your choice: ");
            int choice = scanner.nextInt();
            scanner.nextLine();

            switch (choice) {
                case 1:
                    System.out.print("Enter Employee ID: ");
                    int id = scanner.nextInt();
                    scanner.nextLine();
                    System.out.print("Enter Employee Name: ");
                    String name = scanner.nextLine();
                    System.out.print("Enter Employee Salary: ");
                    double salary = scanner.nextDouble();

                    Employee employee = new Employee(id, name, salary);
                    employeeList.add(employee);
                    System.out.println("Employee added successfully.");
                    break;

                case 2:
                    System.out.print("Enter Employee ID to update: ");
                    int updateId = scanner.nextInt();
                    scanner.nextLine(); // Consume newline

                    boolean updated = false;
                    for (Employee emp : employeeList) {
                        if (emp.getId() == updateId) {
                            System.out.print("Enter new Name: ");
                            String newName = scanner.nextLine();
                            System.out.print("Enter new Salary: ");
                            double newSalary = scanner.nextDouble();
                            emp.setName(newName);
                            emp.setSalary(newSalary);
                            System.out.println("Employee updated successfully.");
                            updated = true;
                            break;
                        }
                    }
                    if (!updated) {
                        System.out.println("Employee with ID " + updateId + " not found.");
                    }
                    break;

                case 3:
                    System.out.print("Enter Employee ID to remove: ");
                    int removeId = scanner.nextInt();

                    boolean removed = false;
                    Iterator<Employee> iterator = employeeList.iterator();
                    while (iterator.hasNext()) {
                        Employee emp = iterator.next();
                        if (emp.getId() == removeId) {
                            iterator.remove();
                            System.out.println("Employee removed successfully.");
                            removed = true;
                            break;
                        }
```

```
                }
                if (!removed) {
                    System.out.println("Employee with ID " + removeId + " not found.");
                }
                break;

            case 4:
                System.out.print("Enter Employee ID to search: ");
                int searchId = scanner.nextInt();

                boolean found = false;
                for (Employee emp : employeeList) {
                    if (emp.getId() == searchId) {
                        System.out.println("Employee found: " + emp);
                        found = true;
                        break;
                    }
                }
                if (!found) {
                    System.out.println("Employee with ID " + searchId + " not found.");
                }
                break;

            case 5:
                System.out.println("List of Employees:");
                for (Employee emp : employeeList) {
                    System.out.println(emp);
                }
                break;

            case 6:
                System.out.println("Exiting...");
                scanner.close();
                return;

            default:
                System.out.println("Invalid choice. Please try again.");
        }
    }
  }
}
```

4. Output:



```
1. Add Employee
2. Update Employee
3. Remove Employee
4. Search Employee
5. Display All Employees
6. Exit
Enter your choice: 1
Enter Employee ID: 20
Enter Employee Name: Rohit
Enter Employee Salary: 56000
Employee added successfully.

1. Add Employee
2. Update Employee
3. Remove Employee
4. Search Employee
5. Display All Employees
6. Exit
Enter your choice: 5
List of Employees:
ID: 20, Name: Rohit, Salary: 56000.0

1. Add Employee
2. Update Employee
3. Remove Employee
4. Search Employee
5. Display All Employees
6. Exit
Enter your choice: 2
Enter Employee ID to update: 20
Enter new Name: Rohit
Enter new Salary: 70000
```



```
1. Add Employee
2. Update Employee
3. Remove Employee
4. Search Employee
5. Display All Employees
6. Exit
Enter your choice: 1
Enter Employee ID: 30
Enter Employee Name: Harshpal
Enter Employee Salary: 10000
Employee added successfully.

1. Add Employee
2. Update Employee
3. Remove Employee
4. Search Employee
5. Display All Employees
6. Exit
Enter your choice: 4
Enter Employee ID to search: 30
Employee found: ID: 30, Name: Harshpal, Salary: 10000.0

1. Add Employee
2. Update Employee
3. Remove Employee
4. Search Employee
5. Display All Employees
6. Exit
Enter your choice: 3
Enter Employee ID to remove: 20
Employee removed successfully.
```

5. Learning Outcomes:

- Understanding ArrayList Operations – Learn how to insert, search, delete, and display elements in an ArrayList.
- User Input Handling – Gain experience in handling user input using the Scanner class.
- Control Structures – Implement decision-making using switch-case and loops for menu-driven programs.
- Exception Handling Awareness – Learn to handle potential input errors, such as invalid choices.
- Practical Java Application – Develop a real-world application demonstrating list manipulation and dynamic data storage