


Name : Ayush Anand
UID : 22BCS10841

Java Assignment 4 Easy Level

Problem Statement :

Easy Level: Employee Management System Problem Statement  Write a Java program to implement an ArrayList that stores employee details (ID, Name, and Salary). Allow users to:

Add employees

Update employee details

Remove employees

Search for employees

Key Concepts Used  ArrayList: To store employee objects.

Encapsulation: Employee details are stored in a class with private fields and public getters/setters.

User Interaction: Using Scanner for input/output operations.

How to Run  Navigate to the Easy/ folder.

Compile and run the EmployeeManagement.java file.

Follow the on-screen instructions to manage employee details.

Code :

```
import java.util.*;
```

```
class Employee {
```

```
    private int id;
```

```
    private String name;
```

```
    private double salary;
```

Name : Ayush Anand
UID : 22BCS10841

```
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;  
}  
  
@Override  
public String toString() {  
    return "ID: " + id + ", Name: " + name + ", Salary: " + salary;  
}  
}
```

Name : Ayush Anand
UID : 22BCS10841

```
public class EmployeeManagement {  
    private static final List<Employee> employees = new ArrayList<>();  
    private static final Scanner scanner = new Scanner(System.in);  
  
    public static void addEmployee() {  
        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();  
        employees.add(new Employee(id, name, salary));  
        System.out.println("Employee added successfully!\n");  
    }  
  
    public static void updateEmployee() {  
        System.out.print("Enter Employee ID to update: ");  
        int id = scanner.nextInt();  
        scanner.nextLine();  
        for (Employee emp : employees) {  
            if (emp.getId() == id) {  
                System.out.print("Enter New Name: ");  
                emp.setName(scanner.nextLine());  
                System.out.print("Enter New Salary: ");  
                emp.setSalary(scanner.nextDouble());  
            }  
        }  
    }  
}
```

Name : Ayush Anand
UID : 22BCS10841

```
        System.out.println("Employee updated successfully!\n");
        return;
    }
}

System.out.println("Employee not found!\n");
}

public static void removeEmployee() {
    System.out.print("Enter Employee ID to remove: ");
    int id = scanner.nextInt();
    employees.removeIf(emp -> emp.getId() == id);
    System.out.println("Employee removed successfully!\n");
}

public static void searchEmployee() {
    System.out.print("Enter Employee ID to search: ");
    int id = scanner.nextInt();
    for (Employee emp : employees) {
        if (emp.getId() == id) {
            System.out.println(emp);
            return;
        }
    }
    System.out.println("Employee not found!\n");
}

public static void main(String[] args) {
    while (true) {
        System.out.println("1. Add Employee");
```

Name : Ayush Anand
UID : 22BCS10841

```
        System.out.println("2. Update Employee");
        System.out.println("3. Remove Employee");
        System.out.println("4. Search Employee");
        System.out.println("5. Exit");
        System.out.print("Choose an option: ");
        int choice = scanner.nextInt();
        switch (choice) {
            case 1 -> addEmployee();
            case 2 -> updateEmployee();
            case 3 -> removeEmployee();
            case 4 -> searchEmployee();
            case 5 -> {
                System.out.println("Exiting... Goodbye!");
                return;
            }
            default -> System.out.println("Invalid choice! Try again.\n");
        }
    }
}
```

Name : Ayush Anand
UID : 22BCS10841

OUTPUT:

```
1. Add Employee
2. Update Employee
3. Remove Employee
4. Search Employee
5. Exit
Choose an option: 1
Enter Employee ID: 101
Choose an option: 1
Choose an option: 1
Choose an option: 1
Enter Employee ID: 101
Choose an option: 1
Choose an option: 1
Enter Employee ID: 101
Enter Employee Name: A
Enter Employee Salary: 120000
Employee added successfully!
```

```
1. Add Employee
2. Update Employee
3. Remove Employee
4. Search Employee
5. Exit
Choose an option: 2
Enter Employee ID to update: 101
Enter New Name: AB
Enter New Salary: 140000
Employee updated successfully!
```

```
1. Add Employee
2. Update Employee
3. Remove Employee
4. Search Employee
5. Exit
Choose an option: 3
Enter Employee ID to remove: 101
Employee removed successfully!
```

```
1. Add Employee
2. Update Employee
3. Remove Employee
4. Search Employee
5. Exit
Choose an option: 4
Enter Employee ID to search: 101
Employee not found!
```

```
1. Add Employee
2. Update Employee
3. Remove Employee
4. Search Employee
5. Exit
Choose an option: 5
Exiting... Goodbye!
```

D:\CU STUDY MATERIAL\Sem 6\Java\src\City