

## Experiment 1

**Student Name: Deepanshu**

**UID: 22BCS15133**

**Branch: BE-CSE**

**Section/Group: DL-902/B**

**Semester: Sixth**

**Date of Performance: 10/01/2025**

**Subject Name: Project Based Learning in Java with Lab**

**Subject Code: 22CSH-359**

**1. Aim:** Given the following table containing information about employees of an organization, develop a small java application, which accepts employee id from the command prompt and displays the following details as output:

### **2. Objective:**

- **Input Employee ID:** Accept an employee ID from the user using the command line or console input.
- **Search Database:** Locate the corresponding employee record in the predefined array of employee data.
- **Map Details:** Retrieve and map employee attributes, including designation and DA, using a switch-case structure.
- **Calculate Salary:** Compute the total salary as Basic + HRA + DA - IT using extracted employee data.
- **Output Result:** Display employee details if found, or print an error message for invalid employee ID input.

### **3. Code:**

```
import java.util.Scanner;  
  
public class Project1 {  
    public static void main(String[] args) {
```

```
Employee[] employees = {  
    new Employee(1001, "Ashish", "R&D", "e", 20000, 8000, 3000),  
    new Employee(1002, "Sushma", "PM", "c", 30000, 12000, 9000),  
    new Employee(1003, "Rahul", "Acct", "k", 10000, 8000, 1000),  
    new Employee(1004, "Chahat", "Front Desk", "r", 12000, 6000, 2000),  
    new Employee(1005, "Ranjan", "Engg", "m", 50000, 20000, 20000),  
    new Employee(1006, "Suman", "Manufacturing", "e", 23000, 9000, 4400),  
    new Employee(1007, "Tanmay", "PM", "c", 29000, 12000, 10000)  
};  
  
Scanner scanner = new Scanner(System.in);  
System.out.print("Enter Employee ID: ");  
int empId = scanner.nextInt();  
boolean found = false;  
  
for (Employee emp : employees) {  
    if (emp.getEmpNo() == empId) {  
        found = true;  
        int da = getDA(emp.getDesigCode());  
        int salary = emp.getBasic() + emp.getHRA() + da - emp.getIT();  
        System.out.println("Emp No. Emp Name Department Designation Salary");  
        System.out.printf("%d %s %s %s %d\n", emp.getEmpNo(),  
emp.getEmpName(), emp.getDepartment(),  
        getDesignation(emp.getDesigCode()), salary);  
        break;  
    }  
}
```

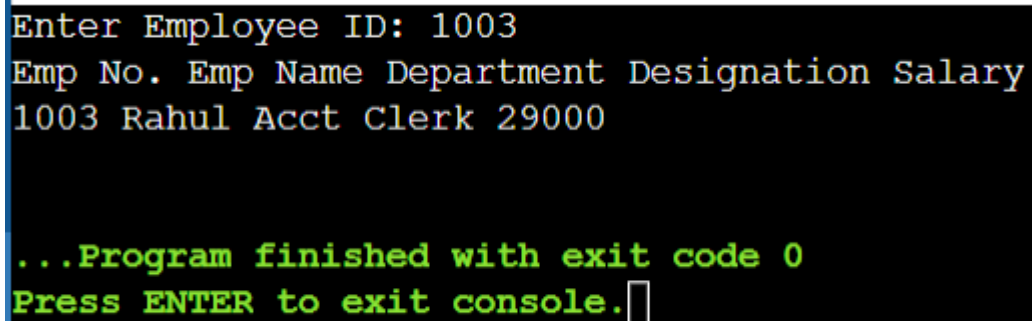
```
        if (!found) {  
            System.out.println("There is no employee with empId : " + empId);  
            scanner.close();  
        }  
private static int getDA(String desigCode) {  
    switch (desigCode) {  
        case "e": return 20000;  
        case "c": return 32000;  
        case "k": return 12000;  
        case "r": return 15000;  
        case "m": return 40000;  
        default: return 0;  
    }  
}  
  
private static String getDesignation(String desigCode) {  
    switch (desigCode) {  
        case "e": return "Engineer";  
        case "c": return "Consultant";  
        case "k": return "Clerk";  
        case "r": return "Receptionist";  
        case "m": return "Manager";  
        default: return "Unknown";  
    }  
}
```

```
}
```

```
class Employee {  
    private int empNo;  
    private String empName;  
    private String department;  
    private String desigCode;  
    private int basic;  
    private int HRA;  
    private int IT;  
  
    public Employee(int empNo, String empName, String department, String desigCode, int  
basic, int HRA, int IT) {  
        this.empNo = empNo;  
        this.empName = empName;  
        this.department = department;  
        this.desigCode = desigCode;  
        this.basic = basic;  
        this.HRA = HRA;  
        this.IT = IT;  
    }  
  
    public int getEmpNo() { return empNo; }  
    public String getEmpName() { return empName; }  
    public String getDepartment() { return department; }  
    public String getDesigCode() { return desigCode; }
```

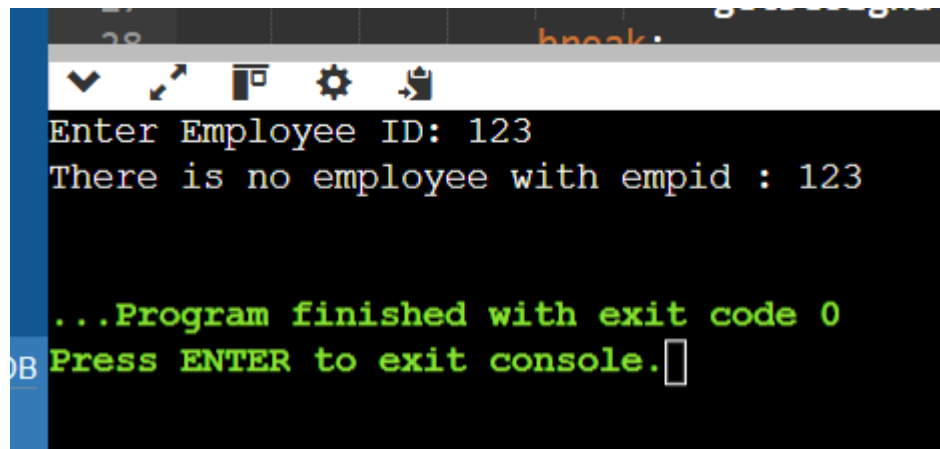
```
public int getBasic() { return basic; }  
public int getHRA() { return HRA; }  
public int getIT() { return IT; }  
}
```

#### 4. Output



```
Enter Employee ID: 1003  
Emp No. Emp Name Department Designation Salary  
1003 Rahul Acct Clerk 29000  
  
...Program finished with exit code 0  
Press ENTER to exit console.
```

Fig 1: Employee ID - 1003



```
Enter Employee ID: 123  
There is no employee with empid : 123  
  
...Program finished with exit code 0  
Press ENTER to exit console.
```

Fig 2: Employee ID - 123

## 5. Learning Outcomes:

- **Understand Array Data Handling:** Learn to work with arrays to store and retrieve structured employee data.
- **Develop Conditional Logic Skills:** Gain proficiency in using switch-case statements for mapping designations and allowances.
- **Enhance Problem-Solving Abilities:** Apply logical steps to compute salary and handle both valid and invalid inputs.
- **Improve Error Handling:** Learn to manage scenarios where the user provides incorrect or non-existent employee IDs.
- **Strengthen Output Formatting:** Understand techniques to display results in a clear and organized tabular format.