EXPERIMENT 1.1

Student Name: Harshita Mishra   **UID: 22BCS80329**

Branch: BE CSE   **Section/Group: DL-902(B)**

Semester: 6th   **Date of Performance: 09.01.2025**

Subject Name: Project Based Learning in Java   **Subject Code: 22CSH-359**

1. **Aim: Create an application to save the employee information using arrays.**

2. **Objective:** To create a 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: Emp No Emp Name Department Designation and Salary.

3. **Program Code:**

```java
import java.util.Scanner;

public class Exp1{

    public static void main(String[] args){

        String[][] employees = {

            {"1001", "Ashish", "01/04/2009", "e", "R&D", "20000", "8000", "3000"},

            {"1002", "Sushma", "23/08/2012", "c", "PM", "30000", "12000", "9000"},

            {"1003", "Rahul", "12/11/2008", "k", "Accounts", "10000", "8000", "1000"},

            {"1004", "Chahat", "29/01/2013", "r", "Front Desk", "12000", "6000", "2000"},

            {"1005", "Ranjan", "16/07/2005", "m", "Engineering", "50000", "20000", "20000"},

            {"1006", "Suman", "01/01/2000", "e", "Manufacturing", "23000", "9000", "4400"},

            {"1007", "Tanmay", "12/06/2006", "c", "PM", "29000", "12000", "10000"},

        };

        Scanner scanner = new Scanner(System.in);
```

```java
System.out.println("Enter Employee ID: ");

String empId = scanner.nextLine();

    boolean found = false;

    for(String[] emp : employees){

        if(emp[0].equals(empId)){

            found=true;

            String empName = emp[1];

            String dept = emp[4];

            String desigCode = emp[3];

            String joinDate = emp[2];

            int basic = Integer.parseInt(emp[5]);

            int hra = Integer.parseInt(emp[6]);

            int it = Integer.parseInt(emp[7]);

            String designation = "";

            int da=0;

            switch(desigCode){

                case "e":

                designation="Engineer";

                da=20000;

                break;

                case "c":

                designation="Consultant";

                da=32000;
```

```java
                    case "k":

                    designation="Clerk";

                    da=12000;

                    break;

                    case "r":

                    designation="Receptionist";

                    da=15000;

                    break;

                    case "m":

                    designation="Manager";

                    da=40000;

                    break;

                    default:

                    System.out.println("Invalid Designation Code.");

                    return;

                }

            int salary = basic+hra+it+da;

            System.out.println("Emp Id   Emp Name   Department   Designation   Salary");

            System.out.printf("%s        %s        %s        %s        %d%n", empId, empName,
        dept, designation, salary);

            break;

        }
```

```java
        if(!found){

            System.out.println("There is no Employee with employee id "+empId);

        }}

}
```

4. Output:

```
Enter Employee ID:
1003
Emp Id      Emp Name      Department      Designation      Salary
1003          Rahul         Accounts          Clerk         31000
Enter Employee ID:
123
There is no Employee with employee id 123
PS D:\P. Languages\JAVA files\PBLJ>
```

5. Learning Outcomes:

- **Understanding Data Structures**:
  Learn to organize and manage data efficiently using 2D arrays, enabling structured storage and retrieval of employee details.
- **Implementing Conditional Logic**:
  Develop the ability to use `switch-case` constructs to handle conditional operations like mapping designation codes to designations and calculating allowances.
- **Enhancing User Interaction**:
  Gain experience in designing interactive console-based applications that accept user inputs and produce dynamic outputs.
- **Mastering Input Handling**:
  Acquire proficiency in using the `Scanner` class for reading and validating user inputs in Java programs.
- **Real-world Salary Calculation Logic**:
  Understand the practical implementation of business logic, such as salary computation involving basic pay, allowances, deductions, and other components.