DEPARTMENT OF
COMPUTER SCIENCE & ENGINEERING
Discover. Learn. Empower.

Expriment 1.1

Nmae: Siddhant Garg

UID: 22BCS10547

Branch: BE-CSE

Section: KRG-IOT-2 B

Semester: 6th

DOP: 11/1/25

Subject: Java

Subject Code: 22CSH-359

Aim:

Develop a small java application, which accepts employee id from the command prompt and displays the details as output. Salary is calculated as Basic+HRA+DA-IT.

Objective:

To develop a functional application that effectively utilizes arrays to store, manage, and retrieve employee information, enabling efficient data organization and manipulation within the application.

Code:

```
import java.util.Scanner;
class Employee {
    int empno;
    String empn;
    String dc;
    String dept;
    int b;
    int hra;
    int it;

    Employee(int e, String n, String d, String de, int ba, int h, int i) {
        empno = e;
        empn = n;
```

```
    dc = d;
    dept = de;
    b = ba;
    hra = h;
    it = i;
    switch(dc){
        case "e":
            dc="Engineer";
            b+=20000;
            break;
        case "c":
            dc="Consultant";
            b+=32000;
            break;
        case "k":
            dc="Clerk";
            b+=12000;
            break;
        case "r":
            dc="Receptionist";
            b+=15000;
            break;
        default:
            break;
    }
}

void dEmployee() {
```

```
        System.out.printf("Emp No: %d   Emp Name: %s    Department: %s  Designation: %s Salary: %d\n",
empno, empn,
                dept, dc, b+hra-it);
    }
}


public class Main {
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        System.out.print("Enter the no. of employees you want to enter: ");
        int n = input.nextInt();
        Employee[] emp = new Employee[n];
        int count = 0;
        while (true) {
            System.out.println("Enter 1 to add employee");
            System.out.println("Enter 2 to view employees");
            System.out.println("Enter 3 to exit");
            int c = input.nextInt();
            switch (c) {
                case 1:
                    if (count < n) {
                        System.out.print("Enter employee ID: ");
                        int empno = input.nextInt();
                        String empn = input.nextLine();
                        System.out.print("Enter employee name: ");
                        empn = input.nextLine();
                        System.out.print("Enter Joining Date: ");
                        String jd = input.nextLine();
```

```
            System.out.print("Enter desig code: ");
            String dc = input.nextLine();
            System.out.print("Enter department: ");
            String dept = input.nextLine();
            System.out.print("Enter Basic: ");
            int b = input.nextInt();
            System.out.print("Enter HRA: ");
            int hra = input.nextInt();
            System.out.print("Enter IT: ");
            int it = input.nextInt();
            emp[count] = new Employee(empno, empn, dc, dept, b, hra, it);
            count += 1;
        } else {
            System.out.println("List full");
        }
        break;
    case 2:
        System.out.println("Enter the emp no: ");
        int find = input.nextInt();
        int flag = 0;
        for (int i = 0; i < n; i++) {
            if (find == emp[i].empno) {
                emp[i].dEmployee();
                flag = 1;
                break;
            }
        }
        if (flag == 0) {
```

```
                System.out.println("Employee not found");
            }
            break;
        case 3:
            return;
        default:
            break;
      }
    }
  }
}
```

Output:

```
Enter the no. of employees you want to enter: 2
Enter 1 to add employee
Enter 2 to view employees
Enter 3 to exit
1
Enter employee ID: 1001
Enter employee name: sidd
Enter Joining Date: 15/4/6
Enter desig code: e
Enter department: PM
Enter Basic: 20000
Enter HRA: 5000
Enter IT: 2000
Enter 1 to add employee
Enter 2 to view employees
Enter 3 to exit
2
Enter the emp no:
1001
Emp No: 1001   Emp Name: sidd    Department: PM  Designation: Engineer Salary: 43000
Enter 1 to add employee
Enter 2 to view employees
Enter 3 to exit
2
Enter the emp no:
1002
Employee not found
Enter 1 to add employee
Enter 2 to view employees
Enter 3 to exit
3
PS C:\Users\siddh\Documents\Java>
```

Learning Outcomes:

1. Demonstrate: Apply key concepts to real-world scenarios to showcase understanding.
2. Analyze: Critically evaluate information, identify patterns, and draw meaningful conclusions.
3. 3. Create: Develop original work, including presentations, reports, or projects, to exhibit comprehension and skills.
4. 4. Communicate: Convey ideas and findings effectively through oral and written communication. 5. Collaborate: Contribute to group projects and exhibit strong teamwork capabilities in a collaborative environment.