Experiment -1.1

Create an application to save the employee information using arrays having following fields:-

empid[],depName[],empDes,empName[],dateJoin[],basic[],hra[],it[], Des Codes [].

Tasks:-

(a) Salary should be calculated as (Basic + HRA + DA - IT)

(b) Printing designation and da according to employee designation.

Student Name: **UID:**
Branch: BE CSE -BD **Section/Group**
Semester: 4th **Date of Performance:**
Subject Name Project Based Learning in Java Lab **Code: 22E-20CSP-287**

1. **Aim/Overview of the practical:**
   Create an application to save the employee information using arrays having following fields:- empid[ ], depName[ ], empDes, empName[ ], dateJoin[ ], hra[ ], it[ ], DesCodes[ ], basic[ ].

2. **Task to be done:**
   (a) Salary should be calculated as (Basic + HRA + DA – IT)
   (b) Printing designation and da according to employee designation

3. Algorithm/Flowchart (For programming based labs)**:**

1. Start.
2. Declare the variables.
3. Create a function Employee.
4. Create a method da to check the designation codes using switch case.
5. Create a method salary to calculate the salary.
6. Create a method details to print the details.
7. Call the main function to print the desired details.
8. End.

4. Steps for experiment/practical:

```
import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;
public class Employee
{
    String empId;
    String depName;
    String empDesignation;
    String empName;
    String dateJoin;
    int basic;
```

```
    int hra;
    int it;
    char designationCode;
    public static int da;
    public Employee(String empId, String depName, String empDesignation,
String empName, String dateJoin, int basic, int hra, int it, char
designationCode)
    {
        this.empId = empId;
        this.depName = depName;
        this.empDesignation = empDesignation;
        this.empName = empName;
        this.dateJoin = dateJoin;
        this.basic = basic;
        this.hra = hra;
        this.it = it;
        this.designationCode = designationCode;
    }
    public static int da(char designationCode)
    {
        switch(designationCode)
        {
            case 'e':
            {
                da = 20000;
                break;
            }
            case 'c':
            {
                da = 32000;
```

```
                break;
            }
            case 'k':
            {
                da = 12000;
                break;
            }
            case 'r':
            {
                da = 15000;
                break;
            }
            case 'm':
            {
                da = 40000;
                break;
            }
            default:
            throw new IllegalStateException("Unexpected value: " +
designationCode);
        }
        return da;
    }
    public static int salary(int basic,int hra,int da,int it)
    {
        int salary = basic+hra+da-it;
        return salary;
    }
    public static void details(String empId,String empName,String
depName,String empDesignation,int salary)
```

```
    {
        System.out.println("Emp Id:\t"+empId);
        System.out.println("Employee Name:\t"+empName);
        System.out.println("Department:\t"+depName);
        System.out.println("Designation:\t"+empDesignation);
        System.out.println("Salary:\t\t"+salary);
    }
    public  static void  main(String[] args) throws IOException
    {
        BufferedReader bufferedReader=new BufferedReader(new
InputStreamReader(System.in));
        String empId;int c=0;
        Employee[] employees=new Employee[3];
        employees[0] =new
Employee("1","HM","Manager","Vishal","1/04/2021",20000,8000,3000,'e' );
        employees[1] =new
Employee("2","PM","Consultant","Ish","23/08/2012",30000,12000,9000,'c');
        employees[2] =new
Employee("3","Acct","Clerk","Viyan","12/11/2008",10000,8000,1000,'k');

        System.out.println("Enter the employee ID ");
        empId = bufferedReader.readLine();
        for(int i=0;i<3;i++)
        {
            if(employees[i].empId.equals(empId))
            {
                c = 1;
                int salary =
salary(employees[i].basic,employees[i].hra,da(employees[i].designationCo
de),employees[i].designationCode);
```

```
                details(employees[i].empId,employees[i].empName,employee
s[i].depName,employees[i].empDesignation,salary);
                break;
            }
        }
        if(c!=1)
        System.out.println("Entered employee ID not found");
    }
}
```

5. Result/Output/Writing Summary:

```
Enter the employee ID
1
Emp Id: 1
Employee Name:  Vishal
Department:     HM
Designation:    Manager
Salary:         47899
```

```
Enter the employee ID
2
Emp Id: 2
Employee Name:  Ish
Department:     PM
Designation:    Consultant
Salary:         73901
```

```
Enter the employee ID
3
Emp Id: 3
Employee Name:  Viyan
Department:     Acct
Designation:    Clerk
Salary:         29893
```

Learning outcomes (What I have learnt):

1. Learn about getter and setter method.

2. Learn about factory method.

3. Learn how to make code more efficient and maintainable.

4. Learn how to implemented OOP in Java.

5. Learn how to use exceptional handling in Java Applications.

Evaluation Grid (To be created as per the SOP and Assessment guidelines by the faculty):

Sr. No.	Parameters	Marks Obtained	Maximum Marks
1.			
2.			
3.			