

### EXPERIMENT NO. 3

**Name : Bhanudas Patil**

**Roll : 40**

**Batch : B**

**DATE OF EXPERIMENT : 02/08/2023**

**DATE OF SUBMISSION : 02/08/2023**

#### **CODE:**

```
import java.util.*;
class Employee
{
public static void main(String[] args)
{
int empid;
String empName;
int Basicsal;
int da;
int hra;
int incometax;
int gross;
Scanner in = new Scanner(System.in);
System.out.println("Enter the Emp Id : ");
empid= in.nextInt();
System.out.println("Enter the Basic Salary : ");
Basicsal=in.nextInt();
System.out.println("Enter the DA :");
da=in.nextInt();
System.out.println("Enter the HRA :");
hra=in.nextInt();
System.out.println("Enter the Income tax :");
incometax=in.nextInt();
```

```

System.out.println("Enter the Name : ");
empName=in.next();
gross=Basicsal+da+hra+incometax;
System.out.println("The basic sal is :, "+gross);
System.out.println("Emlpoyee ID:"+empid);
System.out.println("Basic salary :"+Basicsal);
System.out.println("DA :"+da);
System.out.println("HRA : "+hra);
System.out.println("Income Tax :"+incometax);
System.out.println("Employee name :"+empName);
}
}

```

OUTPUT:

The screenshot shows a Linux desktop environment with a terminal window titled "Employee.java". The terminal is split into two panes. The left pane shows the source code of the Java program, and the right pane shows the output of the program after compilation and execution.

**Source Code (Left Pane):**

```

1 import java.util.*;
2 class Employee
3 {
4     public static void main(String[] args)
5     {
6         int empId;
7         String empName;
8         int Basicsal;
9         int da;
10        int hra;
11        int incometax;
12        int gross;
13        Scanner in = new Scanner(System.in);
14        System.out.println("Enter the Emp Id : ");
15        empId= in.nextInt();
16        System.out.println("Enter the Basic Salary : ");
17        Basicsal=in.nextInt();
18        System.out.println("Enter the DA :");
19        da=in.nextInt();
20        System.out.println("Enter the HRA :");
21        hra=in.nextInt();
22        System.out.println("Enter the Income tax :");
23        incometax=in.nextInt();
24        System.out.println("Enter the Name : ");
25        empName=in.next();
26        gross=Basicsal+da+hra+incometax;
27        System.out.println("The basic sal is :, "+gross);
28        System.out.println("Emlpoyee ID:"+empid);
29        System.out.println("Basic salary :"+Basicsal);
30        System.out.println("DA :"+da);
31        System.out.println("HRA : "+hra);
32        System.out.println("Income Tax :"+incometax);
33        System.out.println("Employee name :"+empName);
34    }
35 }
36
37

```

**Output (Right Pane):**

```

dbit@complab2-1: ~
dbit@complab2-1:~$ javac Employee.java
dbit@complab2-1:~$ java Employee
Enter the Emp Id :
20221230
Enter the Basic Salary :
45000
Enter the DA :
1200000
Enter the HRA :
100000
Enter the Income tax :
30000
Enter the Name :
Bhanudas Patil
The basic sal is :, 1375000
Emlpoyee ID:20221230
Basic salary :45000
DA :1200000
HRA : 100000
Income Tax :30000
Employee name :Bhanudas
dbit@complab2-1:~$

```

### **Q1. What is Java Virtual Machine?**

A Java virtual machine (JVM) is a virtual machine that enables a computer to run java programs as well as programs written in other languages that are also compiled to java bytecode. The JVM is detailed by a specification that formally describes what is required in a JVM implementation.

The Java virtual machine is an abstract (virtual) computer defined by a specification. It is a part of java runtime environment. Every Java virtual machine implementation must have a bootstrap class loader that is capable of loading trusted classes, as well as an extension class loader or application class loader. The Java virtual machine specification does not specify how a class loader should locate classes.

### **Q2.Explain branching and looping.**

Loops In Java are used for executing a block of statements continuously until a particular condition is satisfied.

For example, if we want to print numbers from 1-10 we have two ways.

1- To write `System.out.println()` and put the numbers and print it 10 times.

2-We can simply use loops which will automatically print the numbers from 1 to 100 or whatever is the last digit or condition.

There are 3 types of loops in java-:

1-While loop

2-For loop

3-Do while

Break and continue statements are known as branching statements or jump statements. These statements can be used inside any loop(for,do-while, while).

These statements are used to skip some statements or immediately terminate the loop when the condition is satisfied.

### **Q3. Different data types used in java**

There are two types of datatypes in java i.e. Primitive and Non-Primitive data types  
primitive datatype includes,

Boolean, char, byte, short, int, long, float and double.

Non-primitive datatype include Classes, Interfaces, and Arrays.