JAVA Programming Assignment - 1

Part A-Theory Questions

1)What is Java and why is it platform-independent

Ans: Java is Programming Language used to interact with computer.

Java is used to give instructions to the Computer.

Ex.English is Language used to interact with humans like JAVA is used
To interact with computer

JAVA is Platform-Indepentant, Because Java code runs on JVM, not directly on the operating system.

To Run java program JVM is must ,Java programs depends on JVM to run.

Java runs on any OS,so it is platform-indepentant.

2)Explain the features of Java.

Ans: 1) Simple :1. java is easy to learn and use.

2.Syntax is clear.

2)Object Oriented: 1.Java uses classes and objects.

2. Helps in code reuse and clear design.

3)Platform-Independent:1.Java runs on any OS(Windows,Linux,Mac).

2.Because Java uses JVM.

4)Portable:1.Java code can be moved easily from one system to another.

2.No changes required.

3)What is difference between JDK ,JRE,and JVM?

Ans: 1) JVM (Java Virtual Machine)

JVM is like a **machine inside your computer** that **runs Java programs**.

What it does:

1)Converts **bytecode** (.class file) into machine code.

2)Executes the program 3)Makes Java **platform independent.**

2)JRE (Java Runtime Environment)

What it is:
JRE is used to **run Java applications**.

It contains:1)JVM

2)Core Java libraries (API)

3)Supporting files

Use case:

1)If you only want to **run** Java programs (not develop)

3) **JDK (Java Development Kit)**

What it is:
JDK is used to **develop and run** Java programs.

It contains:

JRE

- Compiler (javac)
- Debugger
- Other development tools

Use case:

- For **Java developers.**

4)What is bytecode in Java?

Ans: **Bytecode** is the **code made from Java program** that **JVM understands**.

Bytecode is not machine code.Also Bytecode is not human – readable.

Bytecode is a code that JVM understands.

Bytecode is a platform-independent code generated by the Java compiler and executed by the JVM.

5)Explain concept of object-oriented programming.

Ans: **Object-Oriented Programming (OOP)** is a programming approach where a program is built using **objects** instead of only functions or logic.

Main Concepts of OOPS.

1 Class

- A **class** is a blueprint or design of an object.
- Object is created from a class.

2 Object

- An **instance of a class**
- Represents real-world entity.

3 Encapsulation

- Binding **data and methods together**
- Data is protected using access modifiers

4 Inheritance

- One class **acquires properties** of another class
- Helps in **code reuse**

5 Polymorphism

- **One thing, many forms**
- Same method name, different behavior

6 Abstraction

- Showing **only important details**
- Hiding unnecessary details.

Part B – Programming Questions

1)Write a java program to print “Hello.Java”.

Ans:

```
public class HelloJava
{
        public static void main(String args[])
            {
                    System.out.println("helloJava");
             }
}
```

2)Write Program to add two numbers.

Ans:

```
import java.util.Scanner;
public class Addition{
        public static void main(String[] args)
       {
            int a,b;
            Scanner sc=new Scanner(System.in);
            System.out.println("enter first number:");
            a=sc.nextInt();
           System.out.println("enter second number:");
            b=sc.nextInt();

            System.out.println("Addition is:"+(a+b));
           }
}
```

3)Write a Program to check weather a number is even or odd.

Ans:

```
import java.util.Scanner;
public class EvenNumber{
        public static void main(String[] args)
        {
            for(int i=0;i<=10;i++)
            {
                if(i%2==0)
                {
                    System.out.println("even");
                }
                else
                    System.out.println("odd");
            }
    }
}
```

4)Write a program to find the largest of three numbers.

Ans:

```
import java.util.Scanner;
public class MaxNumber2{
    public static void main(String[] args){
        int a,b,c;
        System.out.println("enter numbers:");
        Scanner sc=new Scanner(System.in);
        a=sc.nextInt();
         b=sc.nextInt();
          c=sc.nextInt():
           b=a>b?a:b;
```

```
                b=c>b?c:b;
                System.out.println("max number is:"+b);
        }
}
```

5)Write a program to check if a number is positive,negative or zero.

Ans:

```
import java.util.Scanner;
public class CheckNumber {
        public static void main(String[] args) {
                Scanner sc = new Scanner(System.in);
                System.out.print("Enter a number: ");
                int num = sc.nextInt();
                if (num > 0) {
                        System.out.println("Number is Positive");
                } else if (num < 0) {
                        System.out.println("Number is Negative");
                } else {
                        System.out.println("Number is Zero");
                }
        }
}
```