Part A:

Q 1. What is Java and why is it platform-independent?

Java is programming language used to create software applications.

Java is created by Sun Microsystems,now owned by oracle.

Java is easy to use ,secure, widely used for making web applications, mobile apps, and desktop applications.

Java is called as platform independent because a java program can run on any operating system like Windows, Linux, or macOS.

This is possible because of the java virtual machine(JVM).

Java code is first converted into bytecode.

This bytecode runs on the JVM.

Every operating system has its own JVM.

So the same java program works everywhere.

JVM is platform specific i.e Windows JVM, Linux JVM, macOS JVM.

Q 2: Explain the features of java.

Following are the features of java:

1. Simple:

   Java is easy to learn and understand.

   It does not use complex features like pointers.

2. Object oriented:
   Java is object oriented , means java is based on the concept of objects i.e real world entities.
   Java can represents real world entities as objects.Combining data+ behaviour into a single unit.

Java follows object oriented concepts such as class, object, inheritance, and polymorphism, which make programs easy to manage and reuse.

3. Platform independent:

   Java program can run on any operating system because of the java virtual machine (JVM).

4. Secure:

   Java provides strong security features.
   Java does not allow direct access to memory and runs programs inside a secure environment.

5. Robust:

   Java is strong and reliable.

   It handles errors using exception handling and avoids memory problems.

6. Multithreaded:

   Java supports multithreading, which allows multiple tasks to run at same time.

7. High Performance:

   Java uses the just In Time (JIT) , which improves execution speed.

8. Distributed:

   Java can be used to created distributed applications that work over a network.

Q 3. What is the difference between JDK, JRE, and JVM?

JVM:

JVM is virtual machine.

It runs java bytecode.

It makes java platform independent.

JVM is part of JRE.

JVM is used to run byte code.

JRE:

JRE provides the environment to run java programs.

It contains JVM + core libraries.

It does not contain development toos.

It used runs the java programs.

JDK:

Jdk is used to develop java programs.

It contains JRE + development tools like compiler(Javac).

Programmers need jdk to write and compile code.

Q 4: What is bytecode in Java?

Bytecode is code generated when a java program compiled.

Java source code (.java) is compile by java compiler.

It is converted into bytecode(.class).

Bytecode can run on any machine on which JVM is present.

Bytecode make java platform independent.

Same bytecode runs on any operating system.

It insures write once run anywhere.

Q 5: Explain the concept of object oriented programming.

Object oriented programming is a programming approach in which a program is designed using objects.

Object represents real world entities and contains data and methods.

Basic concepts of oops:

Class:

Class is a blueprint or template used to create object.

Object:

An object is an instance of a class that represents a real world thing.

Encapsulation:

Encapsulation means wrapping data and methods together into a single unit(class).

Inheritance:

Inheritance allows one class to use the properties of another class, which helps in code reuse.

Polymorphism:

Polymorphism means one method ,many forms.

The same method can behave differently in different

situations.

Abstraction:

Abstraction means hiding unnecessary details and showing only important features.

Advantages oops:

Code reusability.

Easy to maintain.

Better security.

Real world problem solving.

Part B:

Q 1: Write a program to print Hello,java.

```
    public class Hello {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        System.out.println("Hello,java");

    }

}
```

Output:

```
    Hello,java
```

Q 2: Write a program to add two numbers.

```
import java.util.Scanner;

public class AdditionOfTwoNumber {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter a two number");
        int a = sc.nextInt();
        int b = sc.nextInt();
        System.out.println("Addition is :"+(a+b));


    }

}
```

```
Output:
    Enter a two number
    10
    20
    Addition is :30
```

Q 3: Write a program to check whether a number is even or odd.

```
import java.util.Scanner;

public class EvenOdd {

    public static void main(String[] args) {
        // TODO Auto-generated method stub

        Scanner sc= new Scanner(System.in);
        System.out.println("Enter a number");
        int n = sc.nextInt();

        if (n%2==0)
            System.out.println("Given no is Even");
        else
            System.out.println("Given no is odd");
    }

}
```

Output :

```
    Enter a number
    23
    Given no is odd
```

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

```
    import java.util.Scanner;
```

```java
public class LargestOfThree {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter a three numbers");
        int a = sc.nextInt();
        int b = sc.nextInt();
        int c = sc.nextInt();

        int largest= (a>b)?(a>c?a:c):(b>c?b:c);
        System.out.println("Largest no.is:"+largest);

    }

}
```

Output:

```
    Enter a three numbers
    10
    20
    30
    Largest no.is:30
```

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

```java
import java.util.Scanner;

public class PositiveOrNegative {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter a number");
        int n = sc.nextInt();

        if(n>0)
            System.out.println("Given no is positive");
        else if(n<0)
            System.out.println("Given no is negative");
```

```
          else
              System.out.println("Given no is zero");

     }

}
```

Output:

```
     Enter a number
     10
     Given no is positive
```