

Name : Utsav Vijay Gavli

## Asssignment 1

### Part 1: Introduction to Java

1. What is Java? Explain its significance in modern software development.
2. List and explain the key features of Java.
3. What is the difference between compiled and interpreted languages? Where does Java fit in?
4. Explain the concept of platform independence in Java.
5. What are the various applications of Java in the real world?

### Part 2: History of Java

1. Who developed Java and when was it introduced?
2. What was Java initially called? Why was its name changed?
3. Describe the evolution of Java versions from its inception to the present.
4. What are some of the major improvements introduced in recent Java versions?
5. How does Java compare with other programming languages like C++ and Python in terms of evolution and usability?

### Part 3: Data Types in Java

1. Explain the importance of data types in Java.
2. Differentiate between primitive and non-primitive data types.
3. List and briefly describe the eight primitive data types in Java.
4. Provide examples of how to declare and initialize different data types.
5. What is type casting in Java? Explain with an example.
6. Discuss the concept of wrapper classes and their usage in Java.
7. What is the difference between static and dynamic typing? Where does Java stand?

Write a Java program to declare and initialize all eight primitive data types and print their values.

```
// 1. Program to declare and initialize all primitive data types
public class PrimitiveDataTypes {
    public static void main(String[] args) {
        byte b = 10;
        short s = 20;
        int i = 30;
        long l = 40000L;
        float f = 5.75f;
        double d = 6.98765;
        char c = 'A';
        boolean bool = true;

        System.out.println("Byte: " + b);
        System.out.println("Short: " + s);
        System.out.println("Int: " + i);
        System.out.println("Long: " + l);
        System.out.println("Float: " + f);
        System.out.println("Double: " + d);
        System.out.println("Char: " + c);
        System.out.println("Boolean: " + bool);
    }
}
```

Output:

```
Output
Byte: 10
Short: 20
Int: 30
Long: 40000
Float: 5.75
Double: 6.98765
Char: A
Boolean: true

=== Code Execution Successful ===
```

Write a Java program that takes two integers as input and performs all arithmetic operations on them

```
1 // 2. Program to perform all arithmetic operations on two integers
2 import java.util.Scanner;
3
4 public class ArithmeticOperations {
5     public static void main(String[] args) {
6         Scanner scanner = new Scanner(System.in);
7         System.out.print("Enter first number: ");
8         int num1 = scanner.nextInt();
9         System.out.print("Enter second number: ");
10        int num2 = scanner.nextInt();
11
12        System.out.println("Addition: " + (num1 + num2));
13        System.out.println("Subtraction: " + (num1 - num2));
14        System.out.println("Multiplication: " + (num1 * num2));
15        System.out.println("Division: " + (num1 / num2));
16        System.out.println("Modulus: " + (num1 % num2));
17
18        scanner.close();
19    }
20 }
```

## Output

```
Enter first number: 12
Enter second number: 23
Addition: 35
Subtraction: -11
Multiplication: 276
Division: 0
Modulus: 12
```

```
=== Code Execution Successful ===
```

Implement a Java program to demonstrate implicit and explicit type casting.

```
1 // 3. Program to demonstrate implicit and explicit type casting
2 public class TypeCastingDemo {
3     public static void main(String[] args) {
4         // Implicit Type Casting
5         int num = 100;
6         double doubleNum = num; // Automatic conversion from int to
           double
7         System.out.println("Implicit Type Casting: " + doubleNum);
8
9         // Explicit Type Casting
10        double d = 9.78;
11        int intNum = (int) d; // Manual conversion from double to int
12        System.out.println("Explicit Type Casting: " + intNum);
13    }
14 }
15
```

## Output

Implicit Type Casting: 100.0

Explicit Type Casting: 9

=== Code Execution Successful ===

Create a Java program that converts a given integer to a double and vice versa using wrapper classes.

```
// 4. Program to Convert an Integer to a Double and vice versa using  
wrapper classes  
public class WrapperClassConversion {  
    public static void main(String[] args) {  
        // Integer to Double Conversion  
        Integer intVal = 42;  
        Double doubleVal = intVal.doubleValue();  
        System.out.println("Integer to Double: " + doubleVal);  
  
        // Double to Integer Conversion  
        Double doubleNum = 98.76;  
        Integer intNum = doubleNum.intValue();  
        System.out.println("Double to Integer: " + intNum);  
    }  
}
```

## Output

Integer to Double: 42.0

Double to Integer: 98

=== Code Execution Successful ===

Write a Java program to swap two numbers using a temporary variable and without using a temporary variable.

```
public class SwapNumbers {  
    public static void main(String[] args) {  
        Scanner scanner = new Scanner(System.in);  
        System.out.print("Enter first number: ");  
        int a = scanner.nextInt();  
        System.out.print("Enter second number: ");  
        int b = scanner.nextInt();  
  
        // Swapping using a temporary variable  
        int temp = a;  
        a = b;  
        b = temp;  
        System.out.println("After swapping (using temp variable): a  
            = " + a + ", b = " + b);  
  
        // Swapping without using a temporary variable  
        a = a + b;  
        b = a - b;  
        a = a - b;  
        System.out.println("After swapping (without temp variable):  
            a = " + a + ", b = " + b);  
  
        scanner.close();  
    }  
}
```

## Output

```
Enter first number: 12  
Enter second number: 21  
After swapping (using temp variable): a = 21, b = 12  
After swapping (without temp variable): a = 12, b = 21  
  
=== Code Execution Successful ===
```



Develop a program that takes user input for a character and prints whether it is a vowel or consonant.

```
// 6. Program to check whether a character is a vowel or consonant
import java.util.Scanner;

public class VowelOrConsonant {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter a character: ");
        char ch = scanner.next().charAt(0);

        ch = Character.toLowerCase(ch);
        if (ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' || ch == 'u') {
            System.out.println(ch + " is a vowel.");
        } else if (Character.isLetter(ch)) {
            System.out.println(ch + " is a consonant.");
        } else {
            System.out.println("Invalid input. Please enter an alphabet.");
        }

        scanner.close();
    }
}
```

## Output

```
Enter a character: a
a is a vowel.
```

```
=== Code Execution Successful ===
```

Create a Java program to check whether a given number is even or odd using command-line arguments.

```
// 7. Program to check whether a number is even or odd using command
-line arguments
public class EvenOrOdd {
    public static void main(String[] args) {
        if (args.length != 1) {
            System.out.println("Usage: java EvenOrOdd <number>");
            return;
        }

        int num = Integer.parseInt(args[0]);

        if (num % 2 == 0) {
            System.out.println(num + " is even.");
        } else {
            System.out.println(num + " is odd.");
        }
    }
}
```

## Output

Usage: java EvenOrOdd <number>

=== Code Execution Successful ===



#### Part 4: Java Development Kit (JDK)

1. What is JDK? How does it differ from JRE and JVM?
2. Explain the main components of JDK.
3. Describe the steps to install JDK and configure Java on your system.
4. Write a simple Java program to print "Hello, World!" and explain its structure.
5. What is the significance of the PATH and CLASSPATH environment variables in Java?
6. What are the differences between OpenJDK and Oracle JDK?
7. Explain how Java programs are compiled and executed.
8. What is Just-In-Time (JIT) compilation, and how does it improve Java performance?
9. Discuss the role of the Java Virtual Machine (JVM) in program execution