JAVA PROGRAMMING ASSIGNMENT – 6

PART A – THEORY QUESTIONS

1 : **What is the main method's signature and why is it static?**

main() method is the starting point from where

the JVM starts the execution of a Java program.

JVM will not execute the code if the program is missing the main method.

Hence, it is one of the most important methods of Java,

and having a proper understanding of it is very important.

Static:

It is a keyword that is when associated with a method,

making it a class-related method.

The main() method is static so that JVM can invoke it without instantiating the class.

This also saves the unnecessary wastage of memory which

would have been used by the object declared only for calling the main() method by the JVM.

Q2 **: What are wrapper classes in Java?**

In Java, wrapper classes allow primitive data types to be represented as objects.

Each wrapper class encapsulates a corresponding primitive value inside an object

(e.g., Integer for int, Double for double).

Java provides wrapper classes for all eight

primitive data types to support object-based operations.

Q3 : **Difference between equals() and == operator.**

both equals() and == are used to compare two entities,
but they serve different purposes and behave differently
depending on the data type involved.

Equality (==) Operator : *Compares if two references point
to the same memory location.
*Primitives and object references.
*Cannot be overridden.
*Compares memory addresses.

equals() Method : *Compares the content of objects.
*Working on only object.
*Can be overridden in custom classes.
*Default Compares memory addresses.

Q4 : **What is the final keyword used for?**

*Final variables hold a value that cannot be reassigned after initialization.
*Final methods cannot be overridden by subclasses.
*Final classes cannot be extended.
*Initialization rules require that a final variable
must be assigned exactly once either at declaration
or inside constructors or initializer blocks.
*Reference final variables cannot change which object they
point to though the internal state of the object can change.

*Static final variables represent constants shared across all objects.

*Blank final variables are declared without initialization

and must be assigned once before use.

Local final variables inside methods must be initialized within their block.

Q5 : **Difference between final, finally, and finalize.**

Final :The final keyword in Java is used to restrict modification

by preventing variable reassignment, method overriding,

and class inheritance.

Finally :The final keyword in Java is used to restrict modification

by preventing variable reassignment,

method overriding, and class inheritance.

finalize : The finalize() method is called by the Garbage Collector

just before an object is removed from memory. It allows us to

perform clean up activity. Once the finalize() method completes,

Garbage Collector destroys that object.finalize method is present in the Object class.

PART B – PRACTICAL QUESTIONS

Code 1: **Write a program to find frequency of each character in a String**.

```
import java.util.Scanner;
public class CharFrequency {
public static void main(String[] args) {
```

```
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter a string: ");
        String str = sc.nextLine();
        for (int i = 0; i < str.length(); i++) {
            char ch = str.charAt(i);
            int count = 0;
            for (int j = 0; j < str.length(); j++) {
                if (str.charAt(j) == ch) {
                    count++;
                }
            }
            System.out.println(ch + " : " + count);
        }
    }
}
```

O/p : Enter a string: aacbbb

a : 2

a : 2

c : 1

b : 3

b : 3

b : 3

Code 2**: Write a program to remove whitespace from a String.**

```
import java.util.Scanner;
public class RemoveWhitespace {
 public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);
    System.out.print("Enter a string: ");
    String str = sc.nextLine();
     String result = str.replace(" ", "");
    System.out.println("String without whitespace: " + result);
  }
}
```

O/P:Enter a string: Hello World

String without whitespace: HelloWorld

Code 3:**Write a program to remove duplicates from a String.**

```
    import java.util.Scanner;
    public class RemoveDuplicates {
     public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);
    System.out.print("Enter a string: ");
    String str = sc.nextLine();
    String result = "";
    for (int i = 0; i < str.length(); i++) {
        if (!result.contains(str.charAt(i) + "")) {
            result += str.charAt(i);
        }
    }
```

```
        System.out.println("After removing duplicates: " + result);
    }
}
```

O/P: Enter a string:hello

After removing duplicates: helo

Code 4:**Write a program to find the first non-repeated character in a String**.

```
import java.util.Scanner;
public class FirstNonRepeated {
 public static void main(String[] args) {
  Scanner sc = new Scanner(System.in);
 System.out.print("Enter a string: ");
 String str = sc.nextLine();
 for (int i = 0; i < str.length(); i++) {
    char ch = str.charAt(i);
    int count = 0;
    for (int j = 0; j < str.length(); j++) {
       if (str.charAt(j) == ch) {
          count++;
       }
    }

    if (count == 1) {
       System.out.println("First non-repeated character: " + ch);
       break;
```

```
        }
      }
   }
}
```

O/P : Enter a string:swiss

First non-repeated character: w

Code 5 **: Write a program to find substring in a String.**

```
import java.util.Scanner;
public class FindSubstring {
public static void main(String[] args) {
 Scanner sc = new Scanner(System.in);

 System.out.print("Enter the main string: ");
 String mainString = sc.nextLine();

 System.out.print("Enter the substring: ");
 String subString = sc.nextLine();

 if (mainString.contains(subString)) {
    System.out.println("Substring found!");
 } else {
    System.out.println("Substring not found!");
 }
   }
}
```

O/P :Enter the main string: Hello World

Enter the substring: World

Substring found!