JAVA LAB PROGRAM

Submitted by,

Tamilvanan B.

2018503566.

MO Batch.

09-22-2020.

1. Write a Java program that tracks the presence of more than three consecutive 0's (Zero's) in a string. The program should display the following output

1) Identify the occurrence of more than three consecutive zeros in a string. Count the number of such consecutive occurrence and number of zeros in it.

2) Check if there are any leading or trailing consecutive occurrences in the string. Display the count.

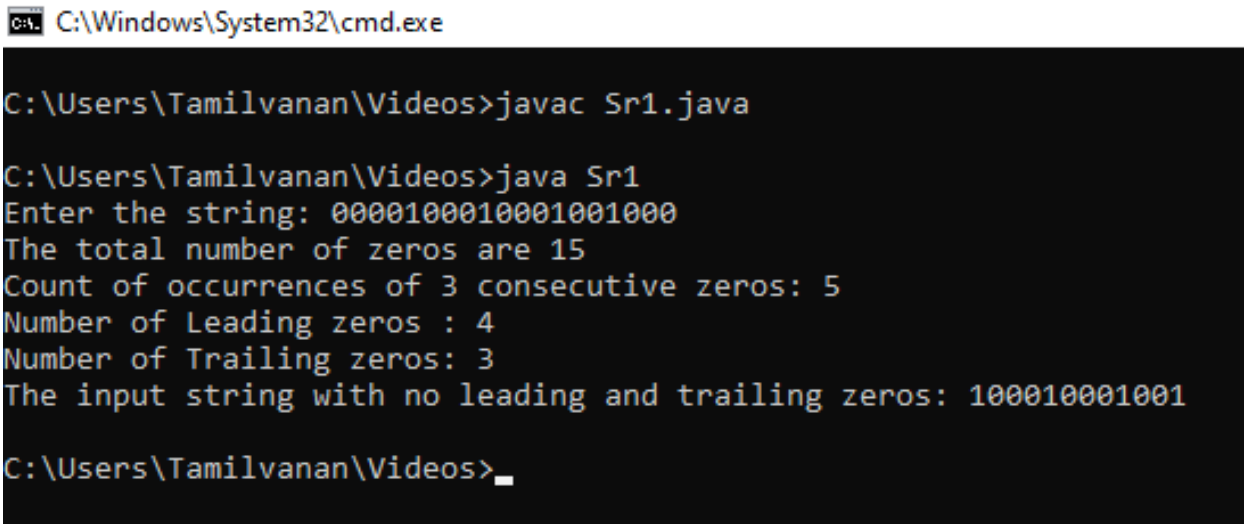
3) Use StringBuffer method to remove such consecutive zero occurrences and display the string

Program:

```
import java.util.Scanner;
public class Sr1{
    public static void main(String... args) {
        Scanner sc = new Scanner(System.in);
        String in;
        int c = 0;
        System.out.print("Enter the string: ");
        in = sc.nextLine();
        int count = 0, l_c = 0, t_c = 0;
        for(int i = 0; i < in.length(); i++){
            if(in.charAt(i) == '0'){
                c ++;
            }
        }
        for(int i = 2; i < in.length(); i++) {
            if(in.charAt(i) == '0' && in.charAt(i - 1) == '0' &&
in.charAt(i - 2) == '0') {
                count++;
            }
        }
        for(int i = 0; i < in.length(); i++) {
            if(in.charAt(i) == '1') {
                break;
            }
            l_c++;
        }
        for(int j = in.length() - 1; j >= 0; j--) {
            if(in.charAt(j) == '1') {
                break;
            }
            t_c++;
```

```
        }
        System.out.println("The total number of zeros are " +
String.valueOf(c));
        System.out.println("Count of occurrences of 3 consecutive zeros: "
+ count);
        System.out.println("Number of Leading zeros : " + l_c);
        System.out.println("Number of Trailing zeros: " + t_c);
        StringBuffer s = new StringBuffer(in);
        System.out.println("The input string with no leading and trailing
0s: " + s.delete(0, l_c).delete(s.length() - t_c, s.length()));
        sc.close();
    }
}
```

Output:



2. Write a java program that accepts two strings as input. The program takes first three characters from first string and last two characters from second string and concatenates into the third string. It then compares the third string with the other two input string.

The program should include following key features:

1. Identify and use appropriate string methods for extracting characters from first and second string

2. Use string methods for Concatenation

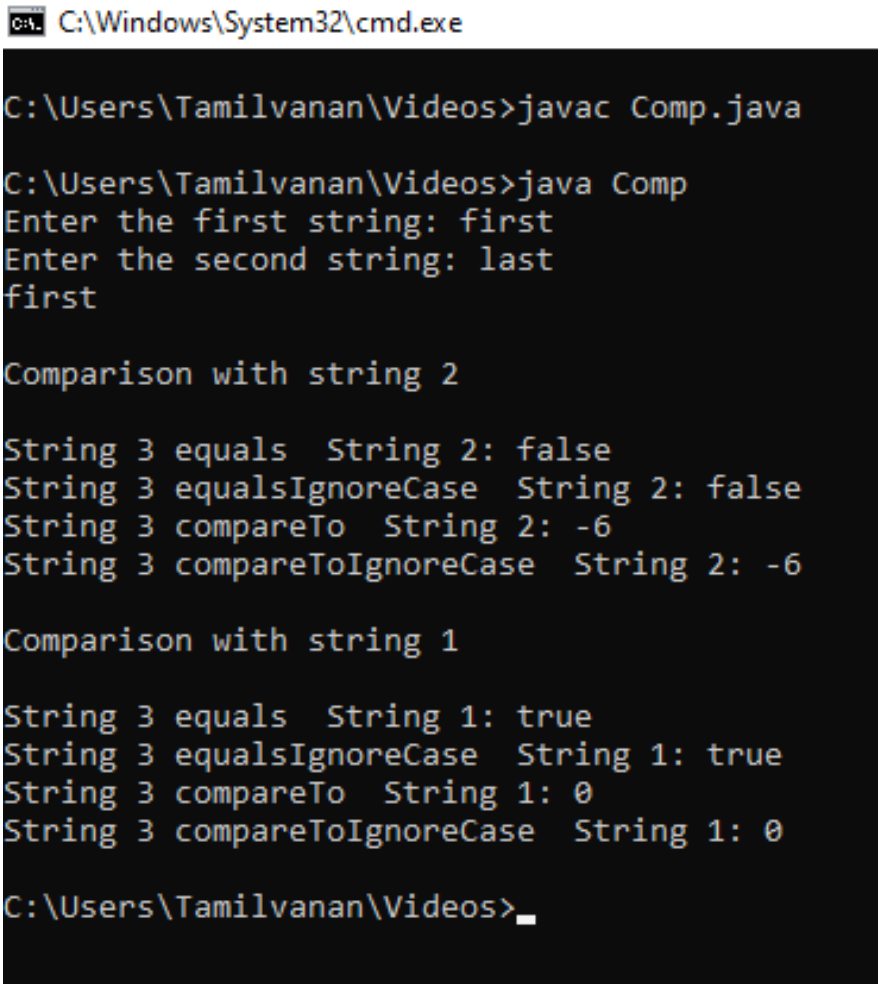
3. Perform String comparison using equals(), equalsIgnoreCase, compareTo() and compareToIgnoreCase() method.

Program:

```
import java.util.Scanner;
public class Comp{
    public static void main(String... args){
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter the first string: ");
        String s1 = sc.next();
        System.out.print("Enter the second string: ");
        String s2 = sc.next();
        String s3 = "";
        s3 = s3 + s1.substring(0,3) + s2.substring(s2.length() - 2,
s2.length());
        System.out.println(s3);
        System.out.println("\nComparison with string 2\n");
        System.out.println("String 3 equals to String 2: " +
s3.equals(s2));
        System.out.println("String 3 equalsIgnoreCase to String 2: " +
s3.equalsIgnoreCase(s2));
        System.out.println("String 3 compareTo to String 2: " +
String.valueOf(s3.compareTo(s2)));
        System.out.println("String 3 compareToIgnoreCase to String 2: " +
String.valueOf(s3.compareToIgnoreCase(s2)));
        System.out.println("\nComparison with string 1\n");
        System.out.println("String 3 equals to String 1: " +
s3.equals(s1));
        System.out.println("String 3 equalsIgnoreCase to String 1: " +
s3.equalsIgnoreCase(s1));
        System.out.println("String 3 compareTo to String 1: " +
String.valueOf(s3.compareTo(s1)));
        System.out.println("String 3 compareToIgnoreCase to String 1: " +
String.valueOf(s3.compareToIgnoreCase(s1)));
```

```
    }
}
```

Output:


C:\Windows\System32\cmd.exe
C:\Users\Tamilvanan\Videos>javac Comp.java
C:\Users\Tamilvanan\Videos>java Comp
Enter the first string: first
Enter the second string: last
first
Comparison with string 2
String 3 equals String 2: false
String 3 equalsIgnoreCase String 2: false
String 3 compareTo String 2: -6
String 3 compareToIgnoreCase String 2: -6
Comparison with string 1
String 3 equals String 1: true
String 3 equalsIgnoreCase String 1: true
String 3 compareTo String 1: 0
String 3 compareToIgnoreCase String 1: 0
C:\Users\Tamilvanan\Videos>

3. You have two versions of welcome note for a program. Write a Java program to compare the two welcome notes lexicographically. The two welcome notes are lexicographically equal if they show similarity in length and contain the same characters in the same positions.

Program:

```
import java.util.Scanner;
public class Comp3{
    public static void main(String[] args){
        Scanner obj=new Scanner(System.in);
        System.out.print("Enter the first string: ");
        String msg1=obj.nextLine();
        System.out.print("Enter the second string: ");
        String msg2=obj.nextLine();
        if(msg1.compareTo(msg2)==0)
         System.out.println("The strings are lexicographically equal");
        else
            System.out.println("The strings are not lexicographically
equal");
    }
}
```

Output: