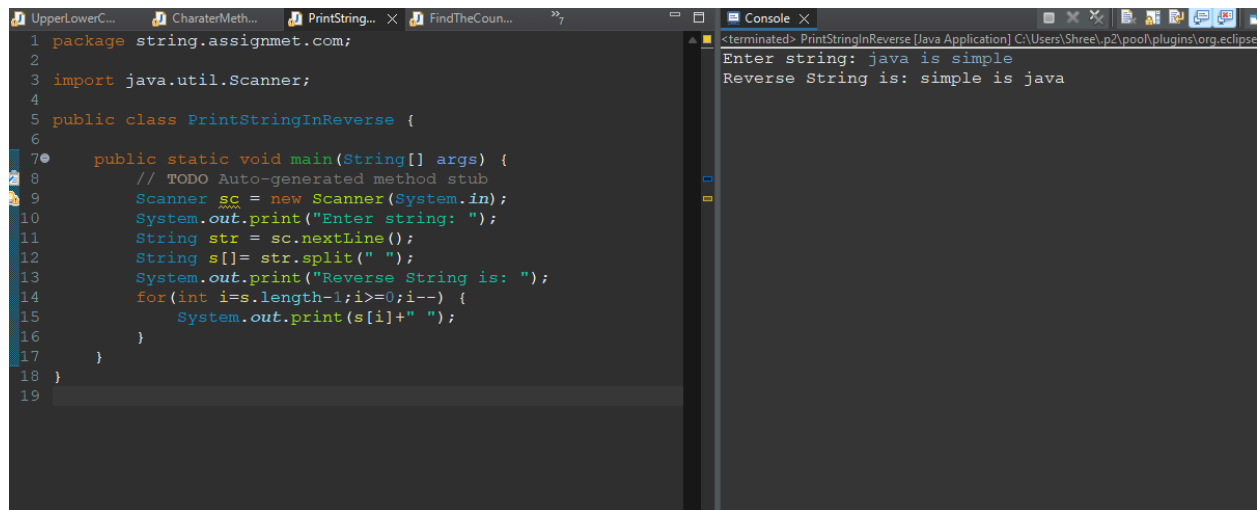


Assignment No:-29

Name:-Suryawanshi Sangramsingh Sambhaji

Batch: - Delta - DCA (Java) 2024 Date:-16/6/2024

Q1.Wap enter a string and print it in a reverse order word by word.



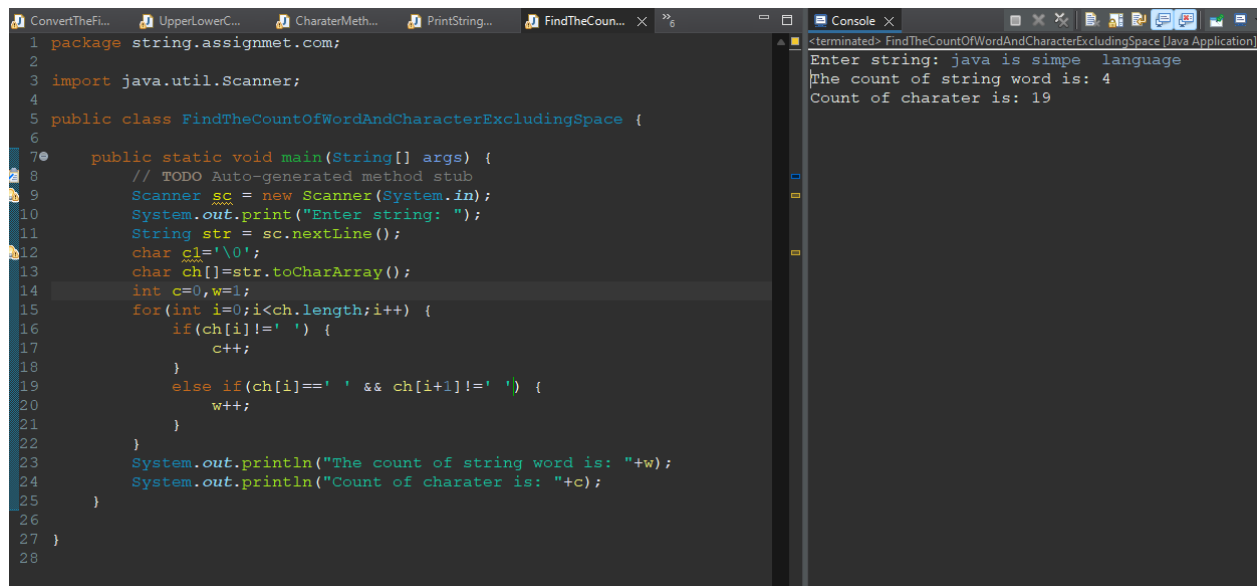
The screenshot shows the Eclipse IDE with a Java file named `PrintStringInReverse.java`. The code uses a `Scanner` to read a string, splits it into words, and then prints them in reverse order. The console output shows the input "java is simple" and the output "Reverse String is: simple is java".

```
1 package string.assignmet.com;
2
3 import java.util.Scanner;
4
5 public class PrintStringInReverse {
6
7     public static void main(String[] args) {
8         // TODO Auto-generated method stub
9         Scanner sc = new Scanner(System.in);
10        System.out.print("Enter string: ");
11        String str = sc.nextLine();
12        String s[] = str.split(" ");
13        System.out.print("Reverse String is: ");
14        for(int i=s.length-1;i>=0;i--) {
15            System.out.print(s[i]+" ");
16        }
17    }
18 }
19
```

Console Output:

```
<terminated> PrintStringInReverse [Java Application] C:\Users\Shree\p2\pool\plugins\org.eclipse
Enter string: java is simple
Reverse String is: simple is java
```

Q2.Wap enter a string and find the count of word and character(excluding space).



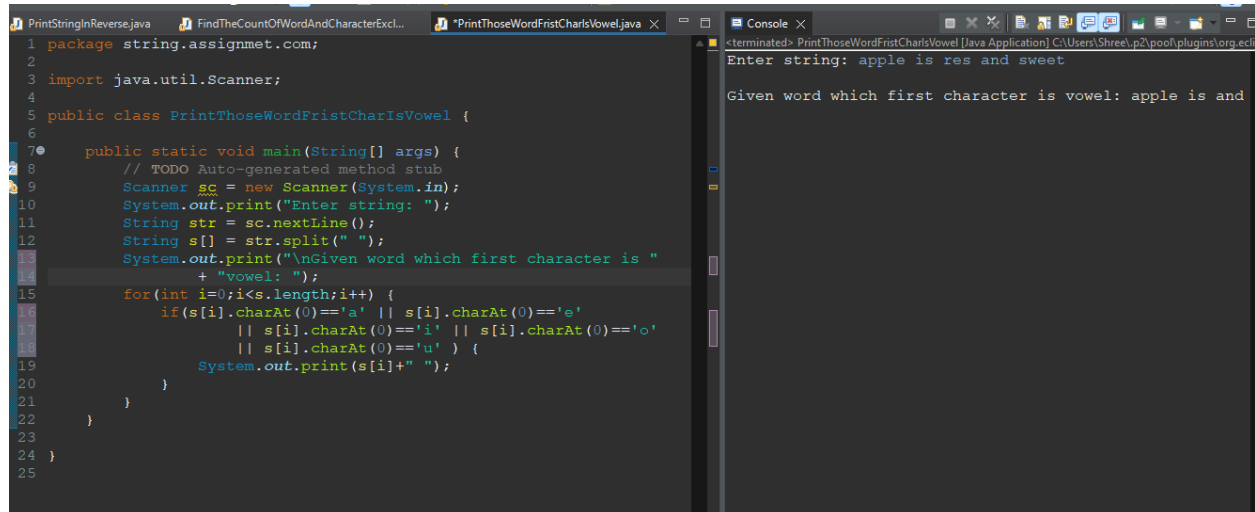
The screenshot shows the Eclipse IDE with a Java file named `FindTheCountOfWordAndCharacterExcludingSpace.java`. The code reads a string and counts the number of words (by splitting on spaces) and the number of characters (excluding spaces). The console output shows the input "java is simpe language" and the output "The count of string word is: 4" and "Count of charater is: 19".

```
1 package string.assignmet.com;
2
3 import java.util.Scanner;
4
5 public class FindTheCountOfWordAndCharacterExcludingSpace {
6
7     public static void main(String[] args) {
8         // TODO Auto-generated method stub
9         Scanner sc = new Scanner(System.in);
10        System.out.print("Enter string: ");
11        String str = sc.nextLine();
12        char c1='\0';
13        char ch[]=str.toCharArray();
14        int c=0,w=1;
15        for(int i=0;i<ch.length;i++) {
16            if(ch[i]!=' ') {
17                c++;
18            }
19            else if(ch[i]==' ' && ch[i+1]!=' ') {
20                w++;
21            }
22        }
23        System.out.println("The count of string word is: "+w);
24        System.out.println("Count of charater is: "+c);
25    }
26 }
27
28
```

Console Output:

```
<terminated> FindTheCountOfWordAndCharacterExcludingSpace [Java Application]
Enter string: java is simpe language
The count of string word is: 4
Count of charater is: 19
```

Q3.Wap input a string and print only those word which first character is vowel.



The screenshot shows an IDE with a Java file named `PrintThoseWordFristCharIsVowel.java`. The code uses a `Scanner` to read a string, splits it into words, and iterates through each word to check if its first character is a vowel (a, i, o, u). The console output shows the input string "apple is res and sweet" and the resulting words "apple is and".

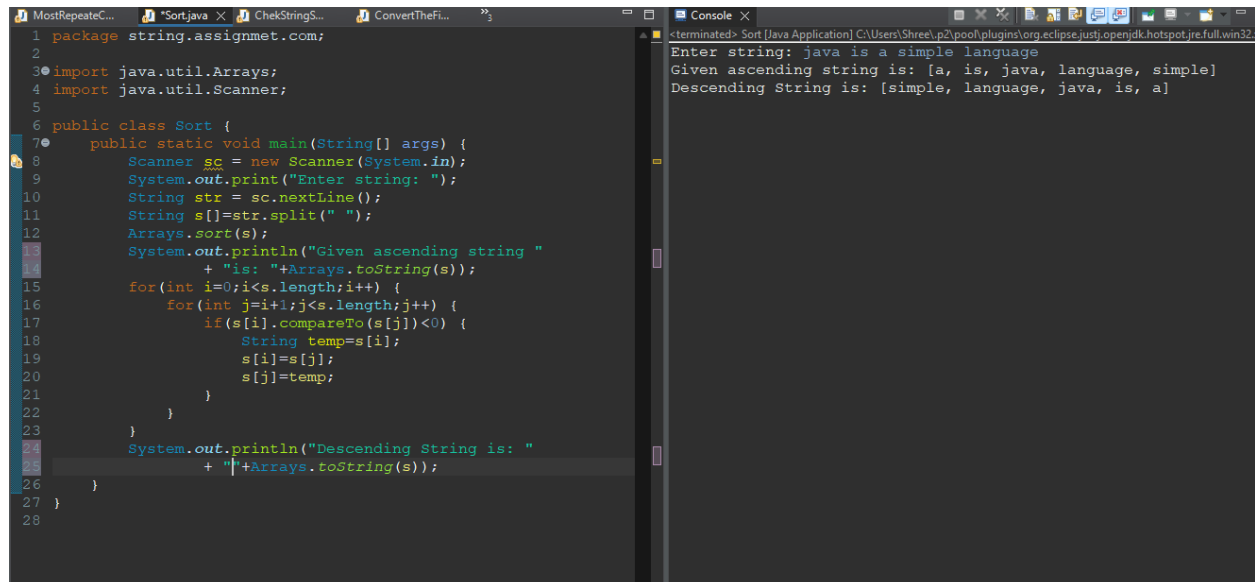
```
1 package string.assignmet.com;
2
3 import java.util.Scanner;
4
5 public class PrintThoseWordFristCharIsVowel {
6
7     public static void main(String[] args) {
8         // TODO Auto-generated method stub
9         Scanner sc = new Scanner(System.in);
10        System.out.print("Enter string: ");
11        String str = sc.nextLine();
12        String s[] = str.split(" ");
13        System.out.print("\nGiven word which first character is "
14            + "vowel: ");
15        for(int i=0;i<s.length;i++) {
16            if(s[i].charAt(0)=='a' || s[i].charAt(0)=='e'
17                || s[i].charAt(0)=='i' || s[i].charAt(0)=='o'
18                || s[i].charAt(0)=='u' ) {
19                System.out.print(s[i]+" ");
20            }
21        }
22    }
23 }
24 }
25 }
```

Console Output:

```
<terminated> PrintThoseWordFristCharIsVowel [Java Application] C:\Users\Shree\p2\pool\plugins\org.ecl...
Enter string: apple is res and sweet

Given word which first character is vowel: apple is and
```

Q4.Wap enter a string and sort each word of string in ascending and descending order by the length of each word.



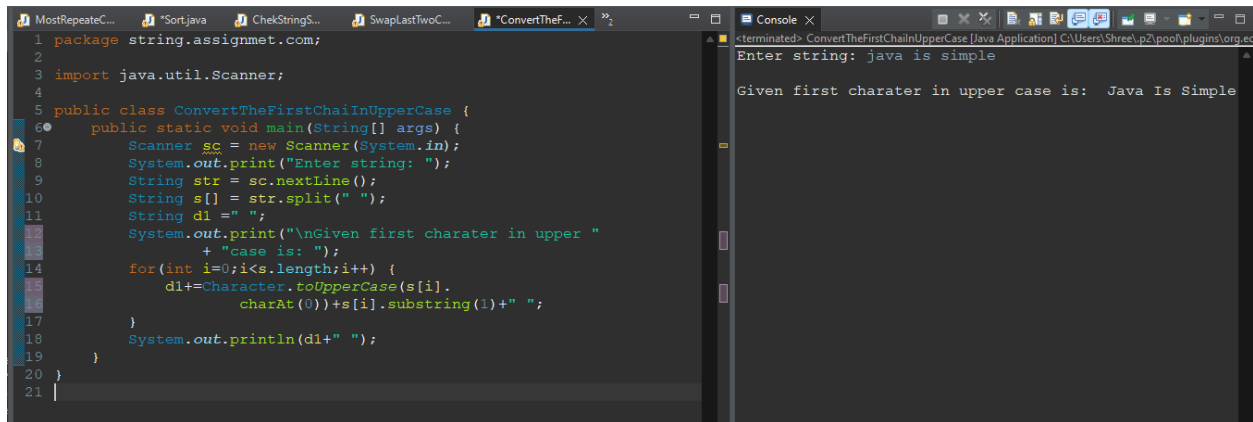
The screenshot shows an IDE with a Java file named `Sort.java`. The code uses a `Scanner` to read a string, splits it into words, and sorts the words in ascending order of length using `Arrays.sort()`. It then prints the sorted words in ascending and descending order. The console output shows the input string "java is a simple language" and the resulting sorted words "a is java language simple" in ascending order and "simple language java is a" in descending order.

```
1 package string.assignmet.com;
2
3 import java.util.Arrays;
4 import java.util.Scanner;
5
6 public class Sort {
7
8     public static void main(String[] args) {
9         Scanner sc = new Scanner(System.in);
10        System.out.print("Enter string: ");
11        String str = sc.nextLine();
12        String s[]=str.split(" ");
13        Arrays.sort(s);
14        System.out.println("Given ascending string "
15            + "is: "+Arrays.toString(s));
16        for(int i=0;i<s.length;i++) {
17            for(int j=i+1;j<s.length;j++) {
18                if(s[i].compareTo(s[j])<0) {
19                    String temp=s[i];
20                    s[i]=s[j];
21                    s[j]=temp;
22                }
23            }
24        }
25        System.out.println("Descending String is: "
26            + " "+Arrays.toString(s));
27    }
28 }
```

Console Output:

```
<terminated> Sort [Java Application] C:\Users\Shree\p2\pool\plugins\org.eclipse.jdt.openide\hotspot.jre.full.win32...
Enter string: java is a simple language
Given ascending string is: [a, is, java, language, simple]
Descending String is: [simple, language, java, is, a]
```

Q5.Wap enter a string in lowercase now you have to convert every first character of the word in upper case.

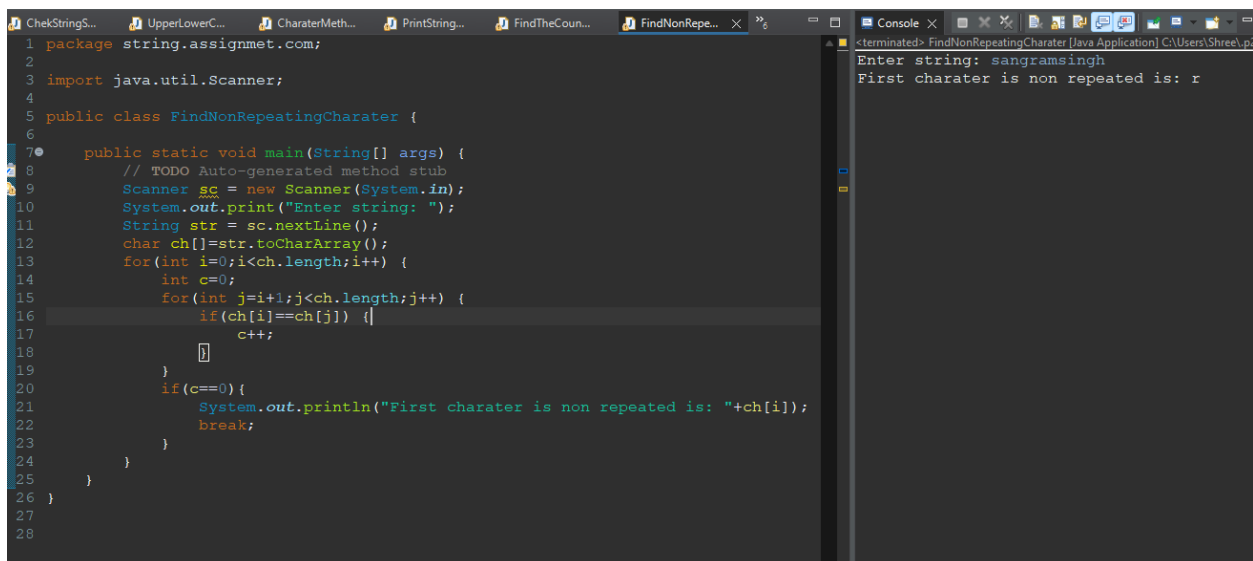


```
1 package string.assignmet.com;
2
3 import java.util.Scanner;
4
5 public class ConvertTheFirstChaiInUpperCase {
6     public static void main(String[] args) {
7         Scanner sc = new Scanner(System.in);
8         System.out.print("Enter string: ");
9         String str = sc.nextLine();
10        String s[] = str.split(" ");
11        String d1 = " ";
12        System.out.print("\nGiven first charater in upper "
13            + "case is: ");
14        for(int i=0;i<s.length;i++) {
15            d1+=Character.toUpperCase(s[i].
16                charAt(0))+s[i].substring(1)+" ";
17        }
18        System.out.println(d1+" ");
19    }
20 }
21
```

Console Output:

```
<terminated> ConvertTheFirstChainUpperCase [Java Application] C:\Users\Shree\p2\poof\plugins\org.e
Enter string: java is simple
Given first charater in upper case is: Java Is Simple
```

Q6.Wap enter a string and print first non-repeating character using class and object.



```
1 package string.assignmet.com;
2
3 import java.util.Scanner;
4
5 public class FindNonRepeatingCharater {
6
7     public static void main(String[] args) {
8         // TODO Auto-generated method stub
9         Scanner sc = new Scanner(System.in);
10        System.out.print("Enter string: ");
11        String str = sc.nextLine();
12        char ch[]=str.toCharArray();
13        for(int i=0;i<ch.length;i++) {
14            int c=0;
15            for(int j=i+1;j<ch.length;j++) {
16                if(ch[i]==ch[j]) {
17                    c++;
18                }
19            }
20            if(c==0){
21                System.out.println("First charater is non repeated is: "+ch[i]);
22                break;
23            }
24        }
25    }
26 }
27
28
```

Console Output:

```
<terminated> FindNonRepeatingCharater [Java Application] C:\Users\Shree\p2
Enter string: sangramsingh
First charater is non repeated is: r
```

Q7.Wap enter a string and check it is anagram or not constructor and using class and object.

```
1 package string.assignmet.com;
2
3 import java.util.Arrays;
4
5 public class ChekStringSsAnagramsOrNot {
6
7
8     ChekStringSsAnagramsOrNot(String str, String str1) {
9         char[] ch = str.toCharArray();
10        char[] chl = str1.toCharArray();
11
12        Arrays.sort(ch);
13        Arrays.sort(chl);
14        String s = new String(ch);
15        String s1 = new String(chl);
16
17        if (s1.equals(s)) {
18            System.out.println("Given strings are anagrams.");
19        } else {
20            System.out.println("Given strings are not anagrams.");
21        }
22    }
23
24    public static void main(String[] args) {
25        Scanner sc = new Scanner(System.in);
26        System.out.print("Enter string 1: ");
27        String str = sc.nextLine();
28        System.out.print("Enter string 2: ");
29        String str1 = sc.nextLine();
30        ChekStringSsAnagramsOrNot c = new ChekStringSsAnagramsOrNot(str, str1);
31    }
32 }
33
```

Console Output:

```
<terminated> ChekStringSsAnagramsOrNot [Java Application] C:\Users\Shree\p2\
Enter string 1: silent
Enter string 2: listen
Given strings are anagrams.
```

Q8.Wap enter a string and print most repeated character in string.

```
1 package string.assignmet.com;
2
3 import java.util.Arrays;
4 import java.util.Scanner;
5
6 public class PrintMostRepetedStringCharEle {
7
8     PrintMostRepetedStringCharEle(String str) {
9         char[] ch = str.toCharArray();
10        char c = '\0';
11        char c1 = '\0';
12        int max = 0;
13        for (int i = 0; i < ch.length; i++) {
14            int c2 = 1;
15            for (int j = i + 1; j < ch.length; j++) {
16                if (ch[i] == ch[j] && ch[i] != '\0') {
17                    c2++;
18                    ch[j] = '\0';
19                }
20            }
21            if (c2 > max) {
22                max = c2;
23                c1 = ch[i];
24            }
25        }
26        System.out.println("Given most repeted charater is: " + c1 + " and count is: " + max);
27    }
28
29    public static void main(String[] args) {
30        Scanner sc = new Scanner(System.in);
31        System.out.print("Enter string 1: ");
32        String str = sc.nextLine();
33        PrintMostRepetedStringCharEle c = new PrintMostRepetedStringCharEle(str);
34    }
35 }
36
```

Console Output:

```
<terminated> PrintMostRepetedStringCharEle [Java Application] C:\Users\Shree\p2\poo\plugins\org.eclipse
Enter string 1: java is simple language
Given most repeted charater is: a and count is 4
```

Q9. Write a Java program to create a new string from a given string swapping the last two Characters of the given string. The length of the given string must be two or more.

Sample Output:

The given strings is: string

The string after swap last two characters are: strign

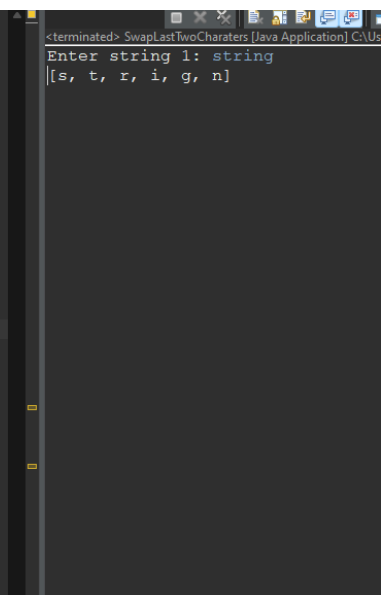
```
package string.assignmet.com;

import java.util.Arrays;
import java.util.Scanner;

public class SwapLastTwoCharaters {

    SwapLastTwoCharaters(String str) {
        if(str.length()<2) {
            System.out.println("Array is too short...");
            return;
        }
        char c[] = str.toCharArray();
        char temp = c[c.length-2];
        c[c.length-2]=c[c.length-1];
        c[c.length-1]=temp;
        System.out.println(Arrays.toString(c));
    }

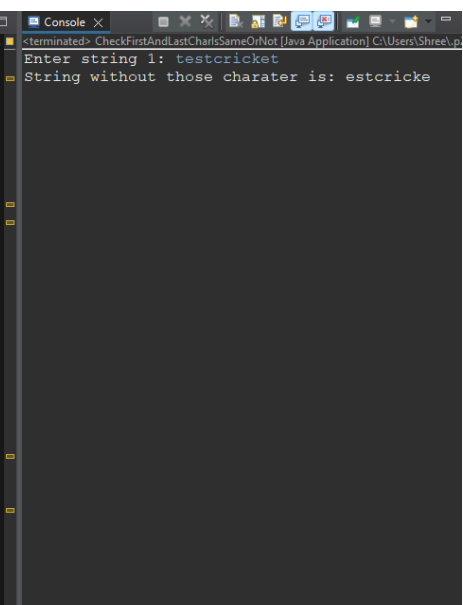
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter string 1: ");
        String str = sc.nextLine();
        SwapLastTwoCharaters c = new SwapLastTwoCharaters(str);
    }
}
```



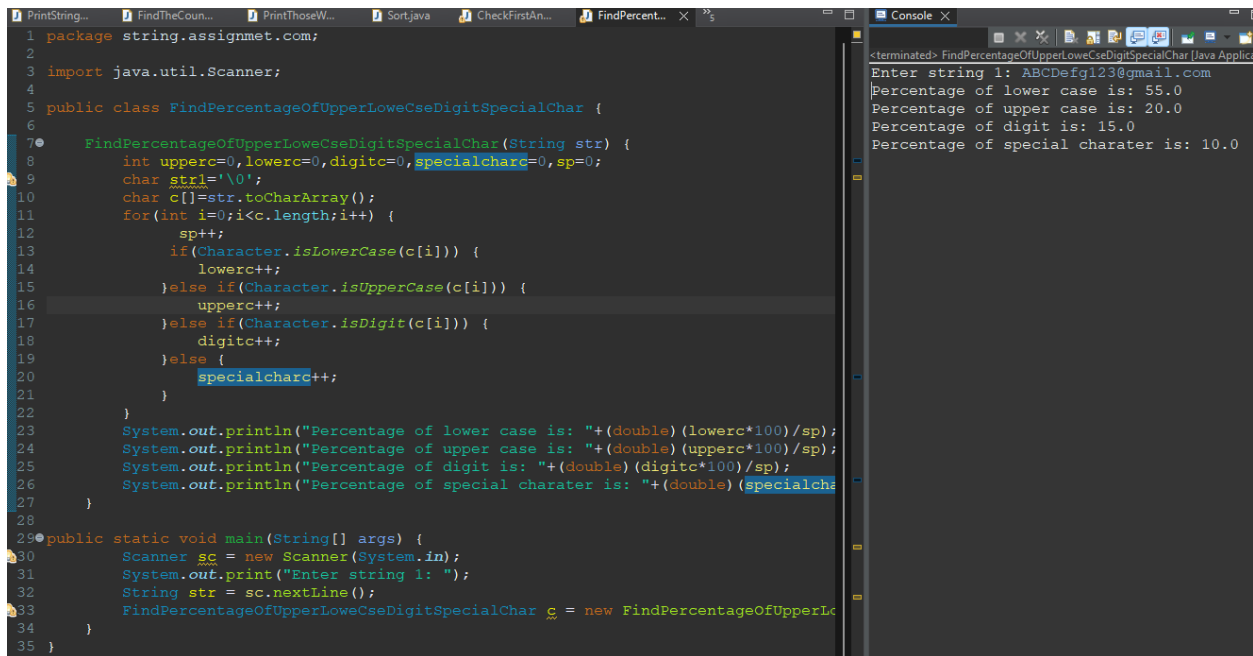
Q10. Write a Java program to read a given string and if the first or last characters are same

Return the string without those characters otherwise return the string unchanged

```
1 package string.assignmet.com;
2
3 import java.util.Arrays;
4 import java.util.Scanner;
5
6 public class CheckFirstAndLastCharIsSameOrNot {
7
8     CheckFirstAndLastCharIsSameOrNot(String str) {
9         char c[] = str.toCharArray();
10        String s="";
11        for(int i=0;i<c.length;i++) {
12            if(c[0]==c[c.length-1]) {
13                String sub=str.substring(1, c.length-1);
14                System.out.println("String without those charater is: "+sub);
15                break;
16            }else {
17                System.out.println("The string is unchanged: ");
18                break;
19            }
20        }
21    }
22
23    public static void main(String[] args) {
24        Scanner sc = new Scanner(System.in);
25        System.out.print("Enter string 1: ");
26        String str = sc.nextLine();
27        CheckFirstAndLastCharIsSameOrNot c = new CheckFirstAndLastCharIsSameOrNot(str);
28    }
29 }
30
31
```



Q11. Wap enter a string and find the percentage of uppercase, lowercase, digits and special characters in a string.



```
1 package string.assignmet.com;
2
3 import java.util.Scanner;
4
5 public class FindPercentageOfUpperLoweCseDigitSpecialChar {
6
7     FindPercentageOfUpperLoweCseDigitSpecialChar(String str) {
8         int upperc=0,lowerc=0,digitc=0,specialcharc=0,sp=0;
9         char strl='\0';
10        char c[]=str.toCharArray();
11        for(int i=0;i<c.length;i++) {
12            sp++;
13            if(Character.isLowerCase(c[i])) {
14                lowerc++;
15            }else if(Character.isUpperCase(c[i])) {
16                upperc++;
17            }else if(Character.isDigit(c[i])) {
18                digitc++;
19            }else {
20                specialcharc++;
21            }
22        }
23        System.out.println("Percentage of lower case is: "+(double)(lowerc*100)/sp);
24        System.out.println("Percentage of upper case is: "+(double)(upperc*100)/sp);
25        System.out.println("Percentage of digit is: "+(double)(digitc*100)/sp);
26        System.out.println("Percentage of special charater is: "+(double)(specialcharc*100)/sp);
27    }
28
29    public static void main(String[] args) {
30        Scanner sc = new Scanner(System.in);
31        System.out.print("Enter string 1: ");
32        String str = sc.nextLine();
33        FindPercentageOfUpperLoweCseDigitSpecialChar c = new FindPercentageOfUpperLoweCseDigitSpecialChar(str);
34    }
35 }
```

Console Output:

```
<terminated> FindPercentageOfUpperLoweCseDigitSpecialChar (Java Application)
Enter string 1: ABCDefg123@gmail.com
Percentage of lower case is: 55.0
Percentage of upper case is: 20.0
Percentage of digit is: 15.0
Percentage of special charater is: 10.0
```

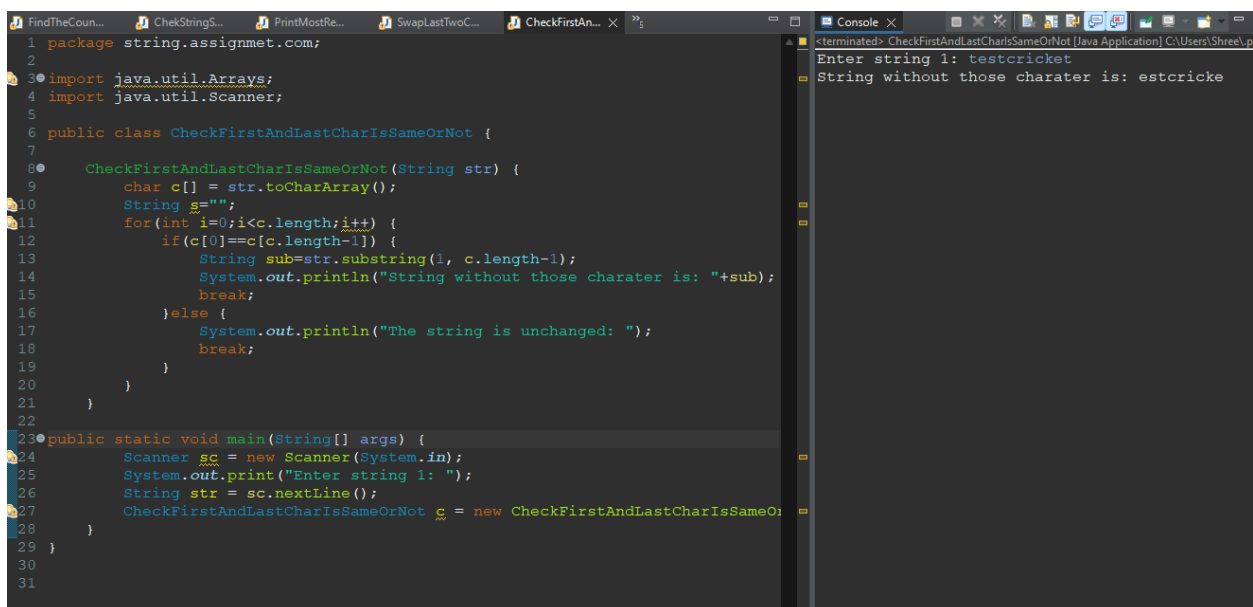
Q12. Write a Java program to read a given string and if the first or last characters are same

return the string without those characters otherwise return the string unchanged.

Sample Output:

The given strings is: testcricket

The new string is: estcricke



```
1 package string.assignmet.com;
2
3 import java.util.Arrays;
4 import java.util.Scanner;
5
6 public class CheckFirstAndLastCharIsSameOrNot {
7
8     CheckFirstAndLastCharIsSameOrNot(String str) {
9         char c[] = str.toCharArray();
10        String s="";
11        for(int i=0;i<c.length;i++) {
12            if(c[0]==c[c.length-1]) {
13                String sub=str.substring(1, c.length-1);
14                System.out.println("String without those charater is: "+sub);
15                break;
16            }else {
17                System.out.println("The string is unchanged: ");
18                break;
19            }
20        }
21    }
22
23    public static void main(String[] args) {
24        Scanner sc = new Scanner(System.in);
25        System.out.print("Enter string 1: ");
26        String str = sc.nextLine();
27        CheckFirstAndLastCharIsSameOrNot c = new CheckFirstAndLastCharIsSameOrNot(str);
28    }
29 }
30
31 }
```

Console Output:

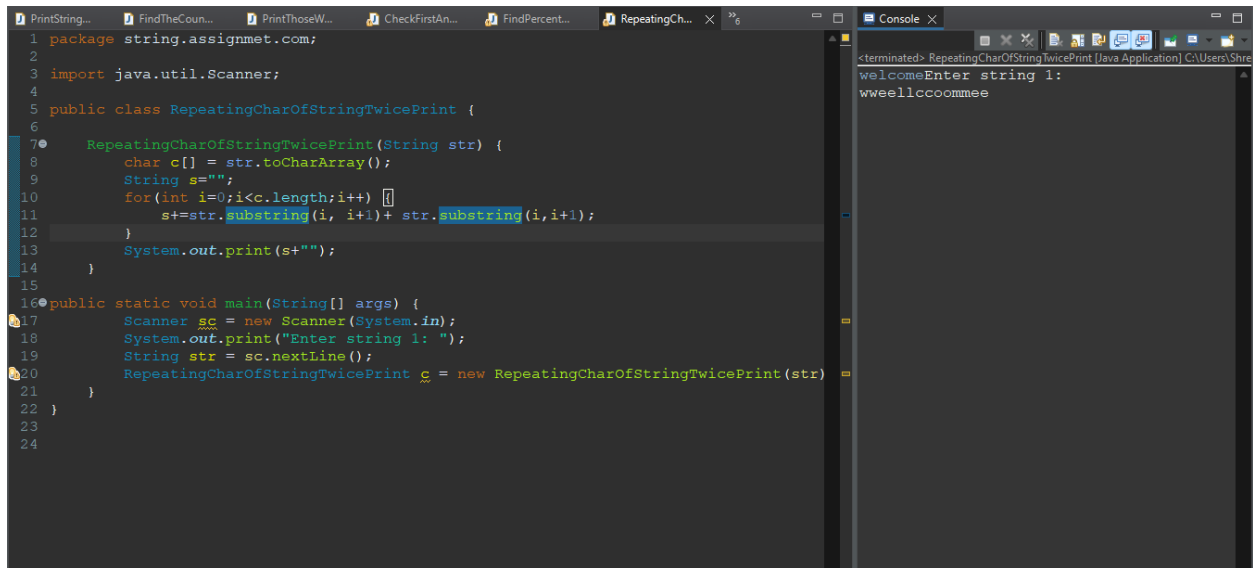
```
<terminated> CheckFirstAndLastCharIsSameOrNot (Java Application) C:\Users\Shree\p
Enter string 1: testcricket
String without those charater is: estcricke
```

Q13. Write a Java program to create a new string repeating every character twice of a given string.

Sample Output:

The given string is: welcome

The new string is: ww e e l l c c o o m m e e

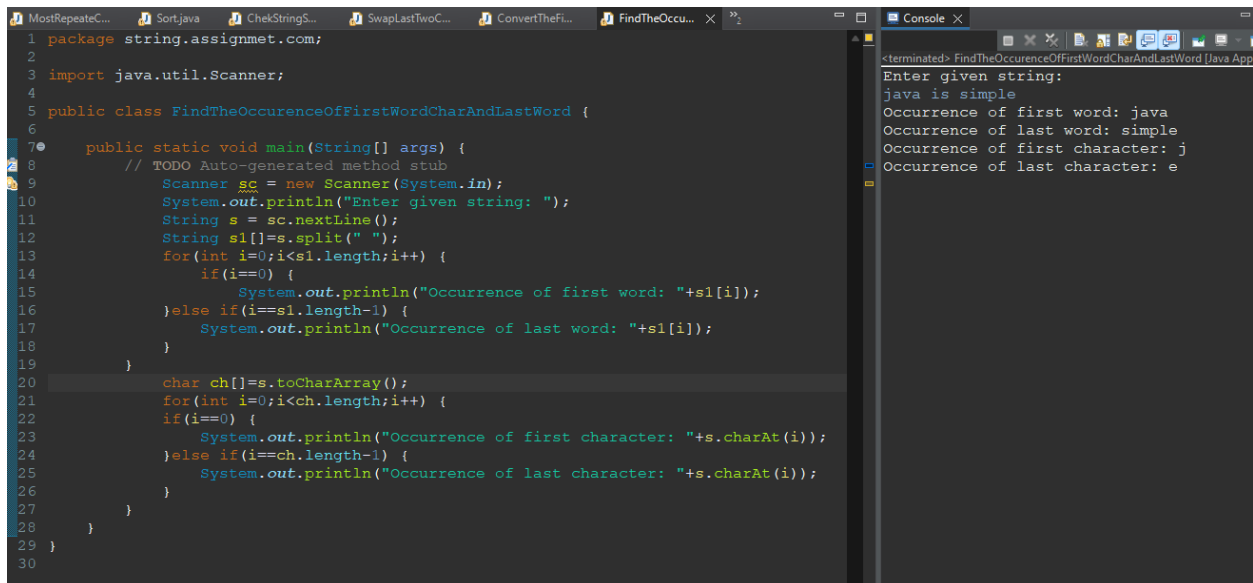


```
1 package string.assignmet.com;
2
3 import java.util.Scanner;
4
5 public class RepeatingCharOfStringTwicePrint {
6
7     RepeatingCharOfStringTwicePrint(String str) {
8         char c[] = str.toCharArray();
9         String s="";
10        for(int i=0;i<c.length;i++) {
11            s+=str.substring(i, i+1)+ str.substring(i,i+1);
12        }
13        System.out.print(s+"");
14    }
15
16    public static void main(String[] args) {
17        Scanner sc = new Scanner(System.in);
18        System.out.print("Enter string 1: ");
19        String str = sc.nextLine();
20        RepeatingCharOfStringTwicePrint c = new RepeatingCharOfStringTwicePrint(str);
21    }
22 }
23
24
```

Console output:

```
<terminated> RepeatingCharOfStringTwicePrint (Java Application) C:\Users\Shre
welcomeEnter string 1:
ww e e l l c c o o m m e e
```

Q14.Wap enter a string and find the occurrence of first word, first character, last word, last character in string by using class and object.



```
1 package string.assignmet.com;
2
3 import java.util.Scanner;
4
5 public class FindTheOccurrenceOfFirstWordCharAndLastWord {
6
7     public static void main(String[] args) {
8         // TODO Auto-generated method stub
9         Scanner sc = new Scanner(System.in);
10        System.out.println("Enter given string: ");
11        String s = sc.nextLine();
12        String s1[]=s.split(" ");
13        for(int i=0;i<s1.length;i++) {
14            if(i==0) {
15                System.out.println("Occurrence of first word: "+s1[i]);
16            }else if(i==s1.length-1) {
17                System.out.println("Occurrence of last word: "+s1[i]);
18            }
19        }
20        char ch[]=s.toCharArray();
21        for(int i=0;i<ch.length;i++) {
22            if(i==0) {
23                System.out.println("Occurrence of first character: "+s.charAt(i));
24            }else if(i==ch.length-1) {
25                System.out.println("Occurrence of last character: "+s.charAt(i));
26            }
27        }
28    }
29 }
30
```

Console output:

```
<terminated> FindTheOccurrenceOfFirstWordCharAndLastWord (Java App
Enter given string:
java is simple
Occurrence of first word: java
Occurrence of last word: simple
Occurrence of first character: j
Occurrence of last character: e
```