Assignment No:-34

Name:-Suryawanshi Sangramsingh Sambhaji

Batch: - Delta - DCA (Java) 2024 Date:-25/6/2024

TOP STRING QUESTIONS FOR INTERVIEW AND LOGIC BUILDING.

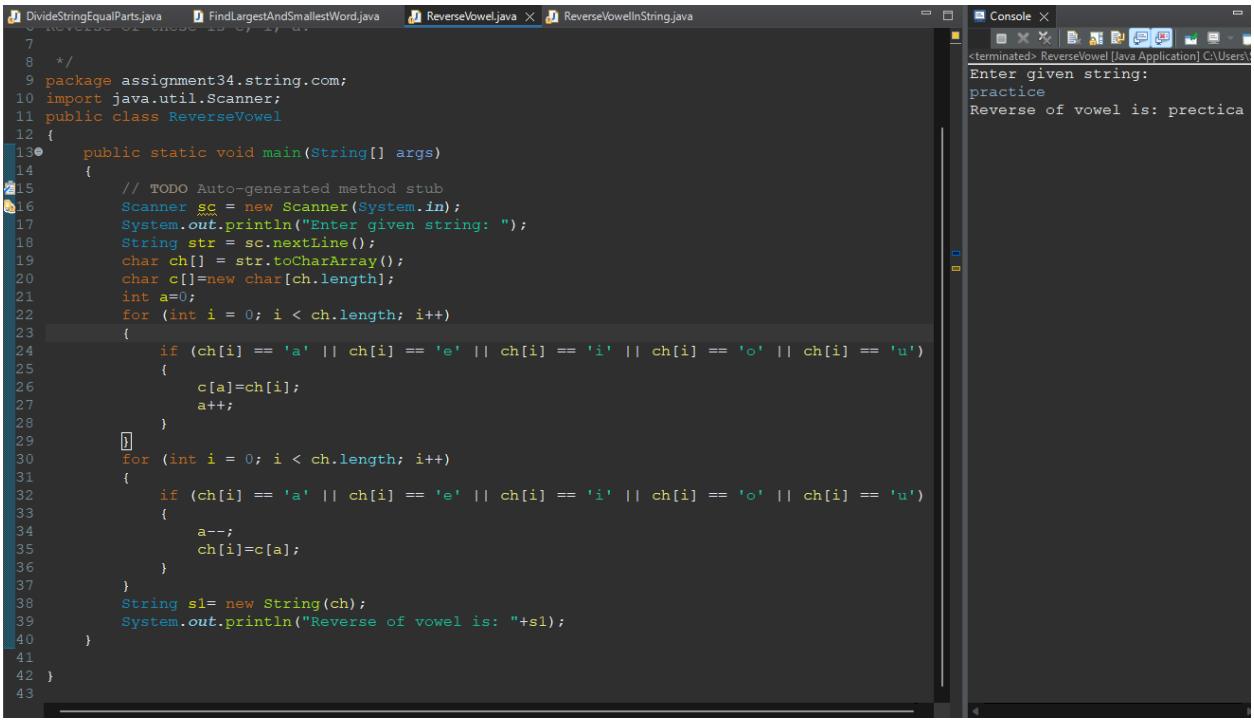
Q1. Reverse the vowels only (Flipkart)

S = "practice"

Output: prectica

Explanation: The vowels are a, i, e

Reverse of these is e, i, a.




```java
 */
package assignment34.string.com;
import java.util.Scanner;
public class ReverseVowel
{
    public static void main(String[] args)
    {
        // TODO Auto-generated method stub
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter given string: ");
        String str = sc.nextLine();
        char ch[] = str.toCharArray();
        char c[]=new char[ch.length];
        int a=0;
        for (int i = 0; i < ch.length; i++)
        {
            if (ch[i] == 'a' || ch[i] == 'e' || ch[i] == 'i' || ch[i] == 'o' || ch[i] == 'u')
            {
                c[a]=ch[i];
                a++;
            }
        }
        for (int i = 0; i < ch.length; i++)
        {
            if (ch[i] == 'a' || ch[i] == 'e' || ch[i] == 'i' || ch[i] == 'o' || ch[i] == 'u')
            {
                a--;
                ch[i]=c[a];
            }
        }
        String s1= new String(ch);
        System.out.println("Reverse of vowel is: "+s1);
    }

}
```

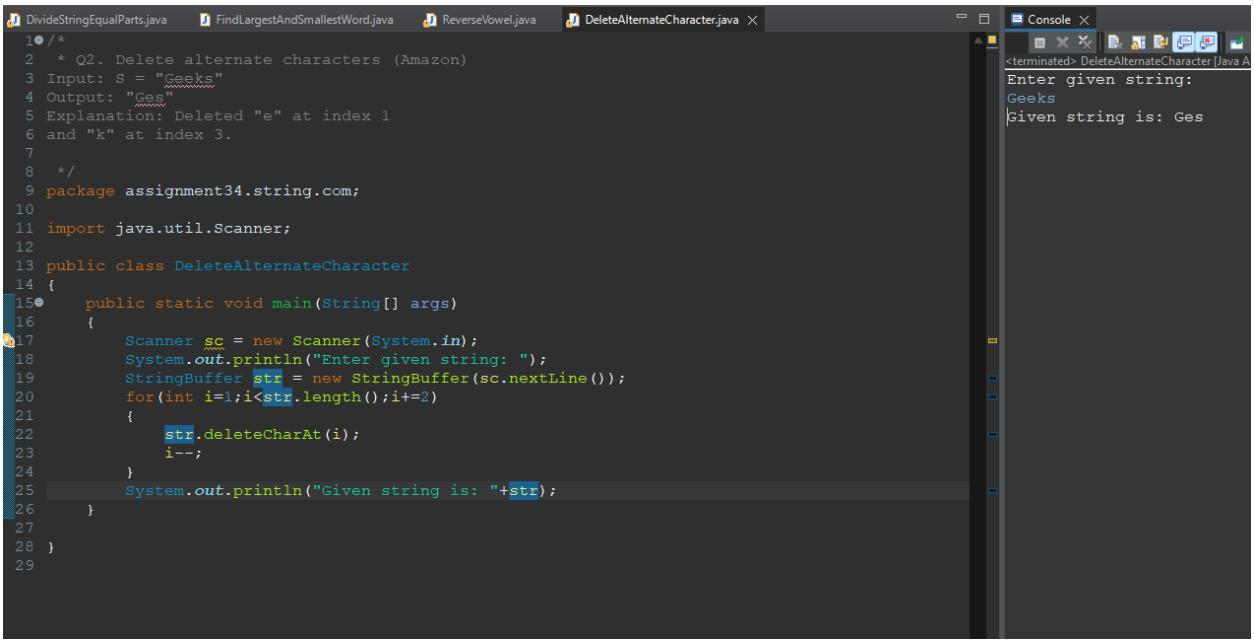
Q2. Delete alternate characters (Amazon)

Input: S = "Geeks"

Output: "Ges"

Explanation: Deleted "e" at index 1

and "k" at index 3.




```java
/*
 * Q2. Delete alternate characters (Amazon)
Input: S = "Geeks"
Output: "Ges"
Explanation: Deleted "e" at index 1
and "k" at index 3.

 */
package assignment34.string.com;

import java.util.Scanner;

public class DeleteAlternateCharacter
{
    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter given string: ");
        StringBuffer str = new StringBuffer(sc.nextLine());
        for(int i=1;i<str.length();i+=2)
        {
            str.deleteCharAt(i);
            i--;
        }
        System.out.println("Given string is: "+str);
    }

}
```

Q3. Extract the integers (Zoho)

Input:

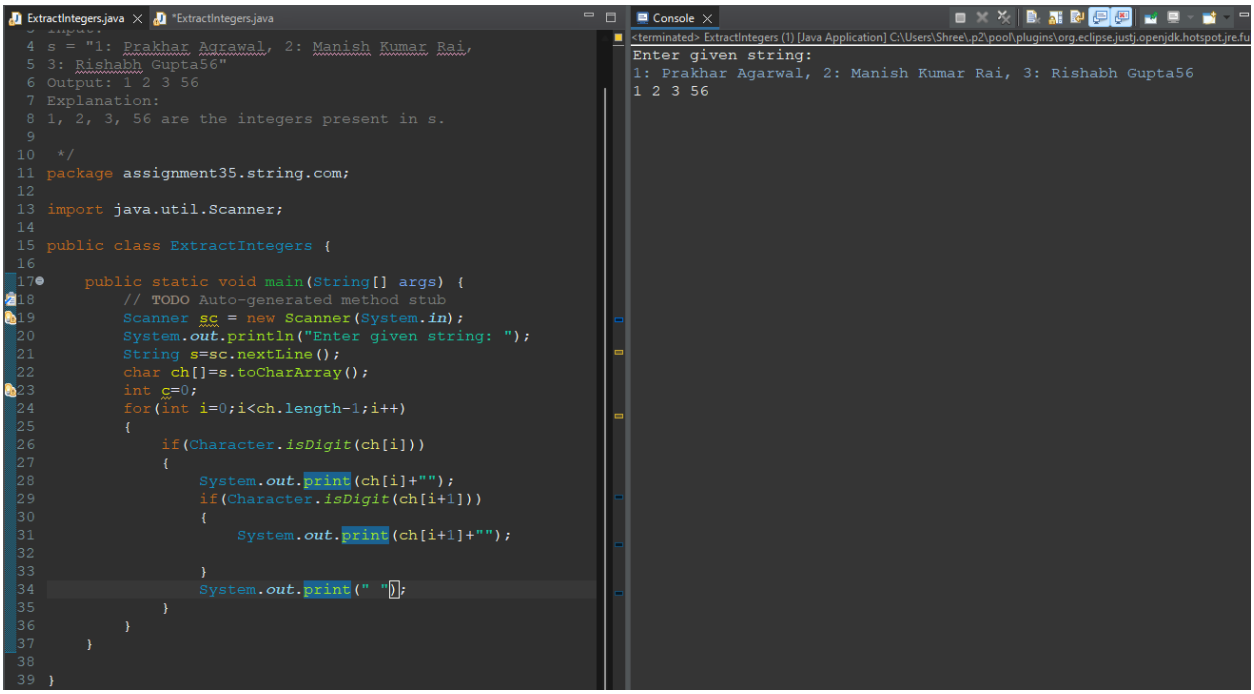
s = "1: Prakhar Agrawal, 2: Manish Kumar Rai,

3: Rishabh Gupta56"

Output: 1 2 3 56

Explanation:

1, 2, 3, 56 are the integers present in s.

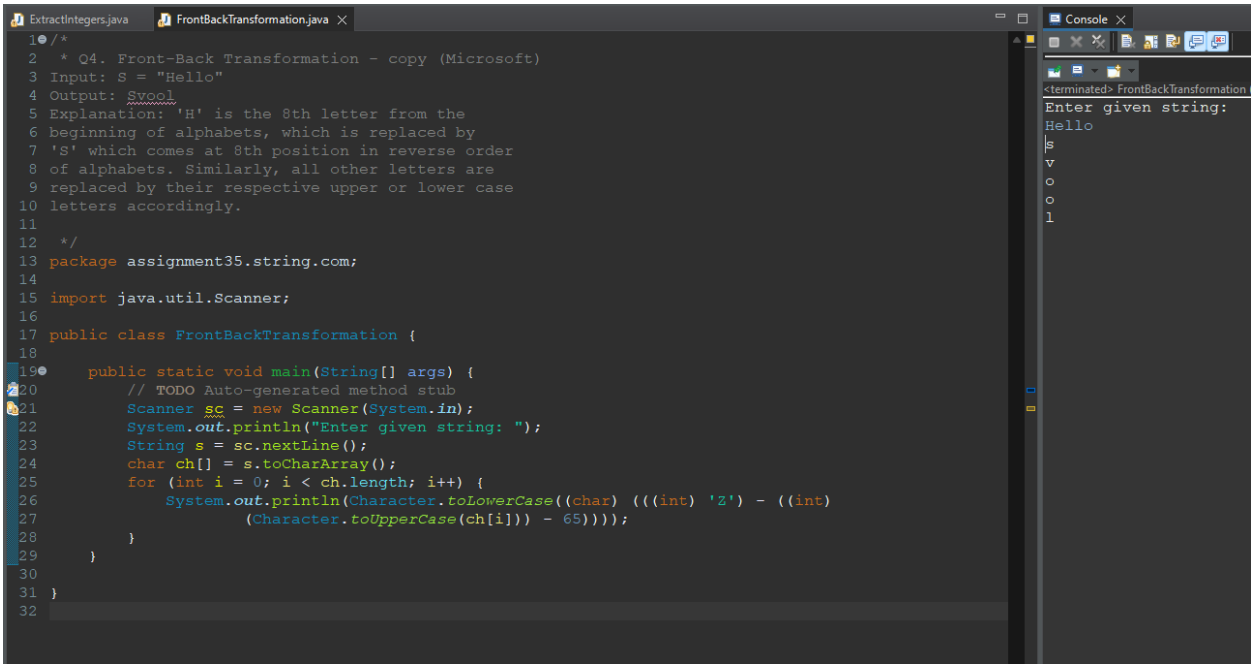


Q4. Front-Back Transformation - copy (Microsoft)

Input: S = "Hello"

Output: Svool

Explanation: 'H' is the 8th letter from the beginning of alphabets, which is replaced by 'S' which comes at 8th position in reverse order of alphabets. Similarly, all other letters are replaced by their respective upper or lower case letters accordingly.



Q5. Uncommon characters (Zoho)

Input:

A = geeksforgeeks

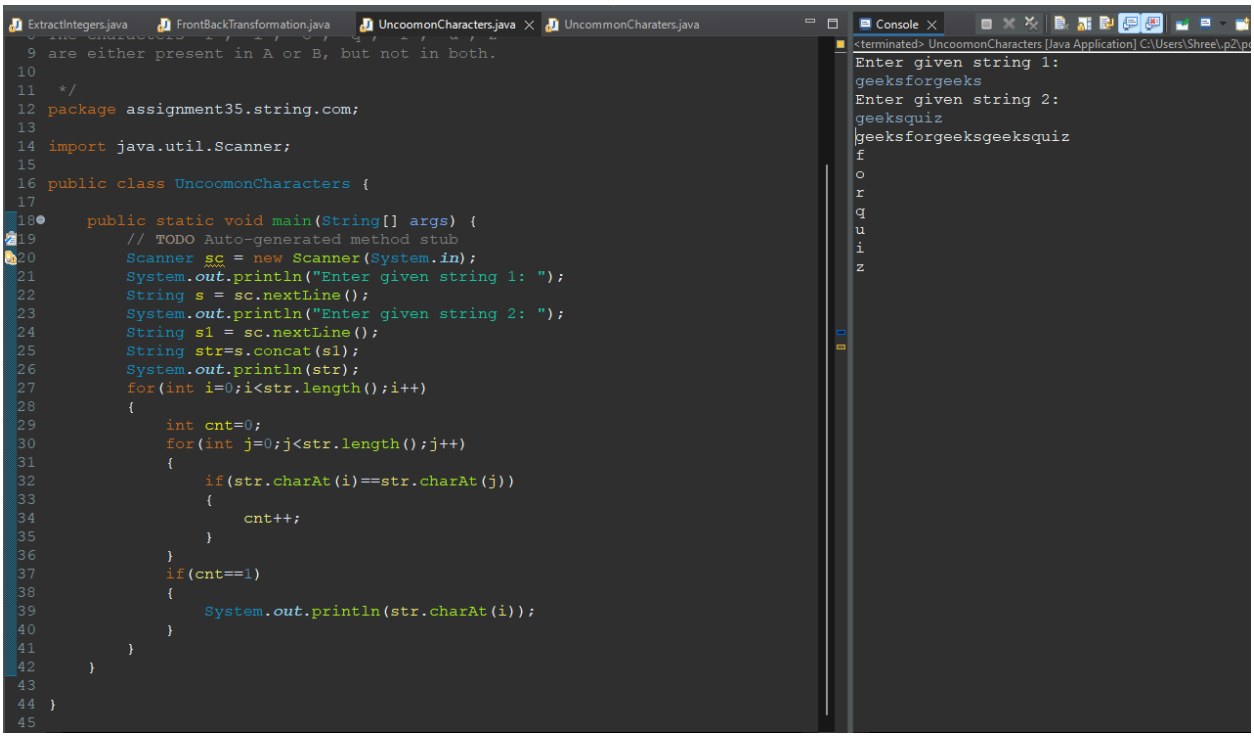
B = geeksquiz

Output: fioqruz

Explanation:

The characters 'f', 'i', 'o', 'q', 'r', 'u','z'

are either present in A or B, but not in both.



Q6. Remove common characters and concatenate (Oracle)

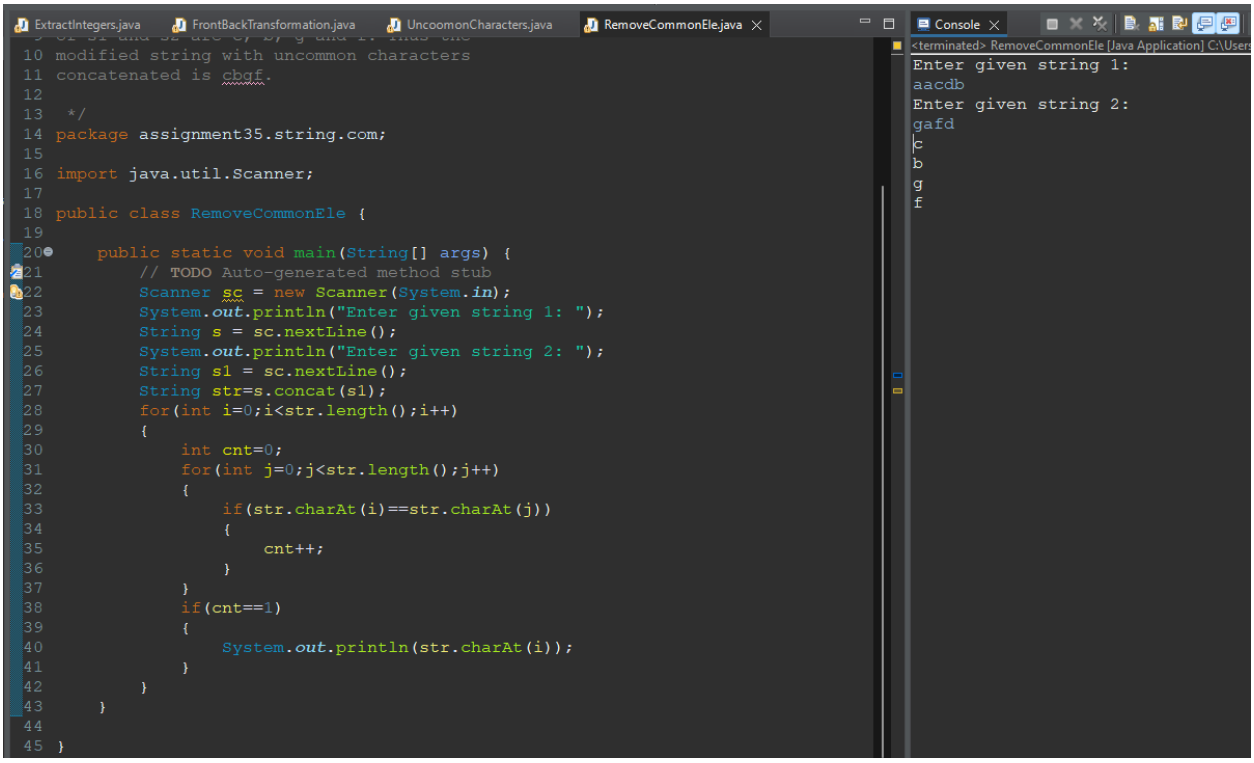
Input:

s1 = aacdb

s2 = gafd

Output: cbgf

Explanation: The common characters of s1 and s2 are: a, d. The uncommon characters of s1 and s2 are c, b, g and f. Thus the modified string with uncommon characters concatenated is cbgf.



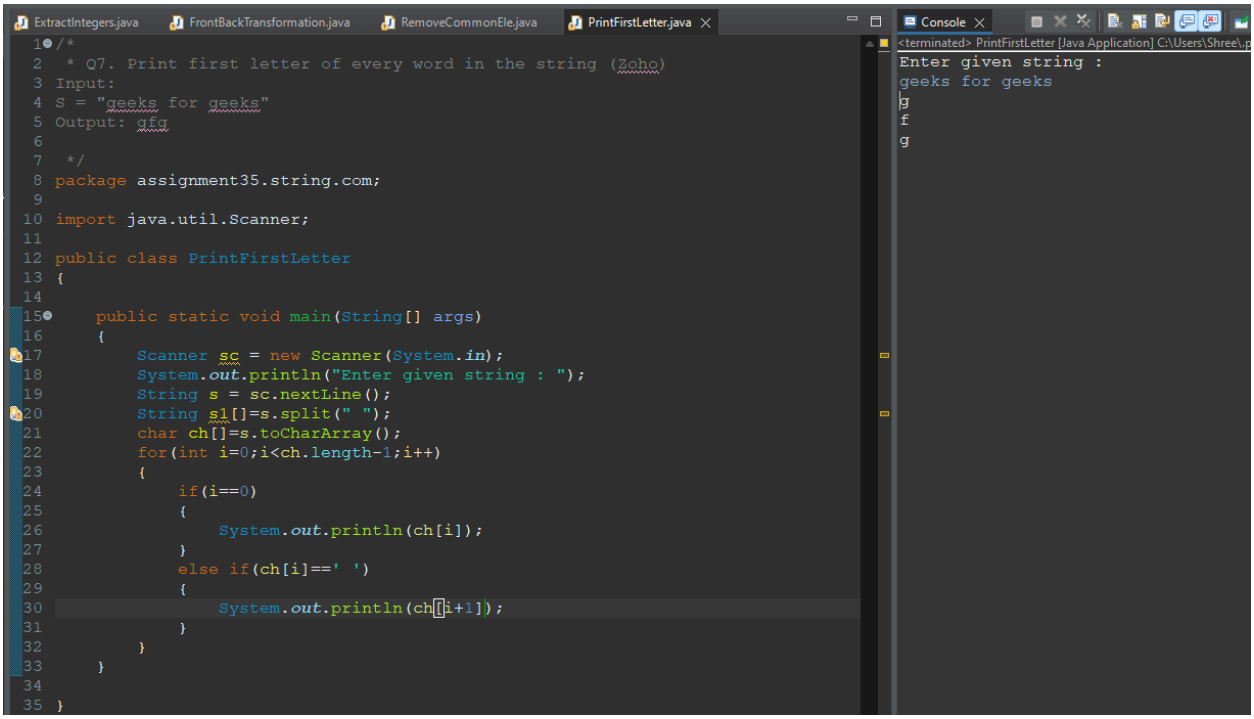

```
modified string with uncommon characters
concatenated is cbgf.

 */
package assignment35.string.com;

import java.util.Scanner;

public class RemoveCommonEle {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter given string 1: ");
        String s = sc.nextLine();
        System.out.println("Enter given string 2: ");
        String s1 = sc.nextLine();
        String str=s.concat(s1);
        for(int i=0;i<str.length();i++)
        {
            int cnt=0;
            for(int j=0;j<str.length();j++)
            {
                if(str.charAt(i)==str.charAt(j))
                {
                    cnt++;
                }
            }
            if(cnt==1)
            {
                System.out.println(str.charAt(i));
            }
        }
    }

}
```

Q7. Print first letter of every word in the string (Zoho)

Input:

S = "geeks for geeks"

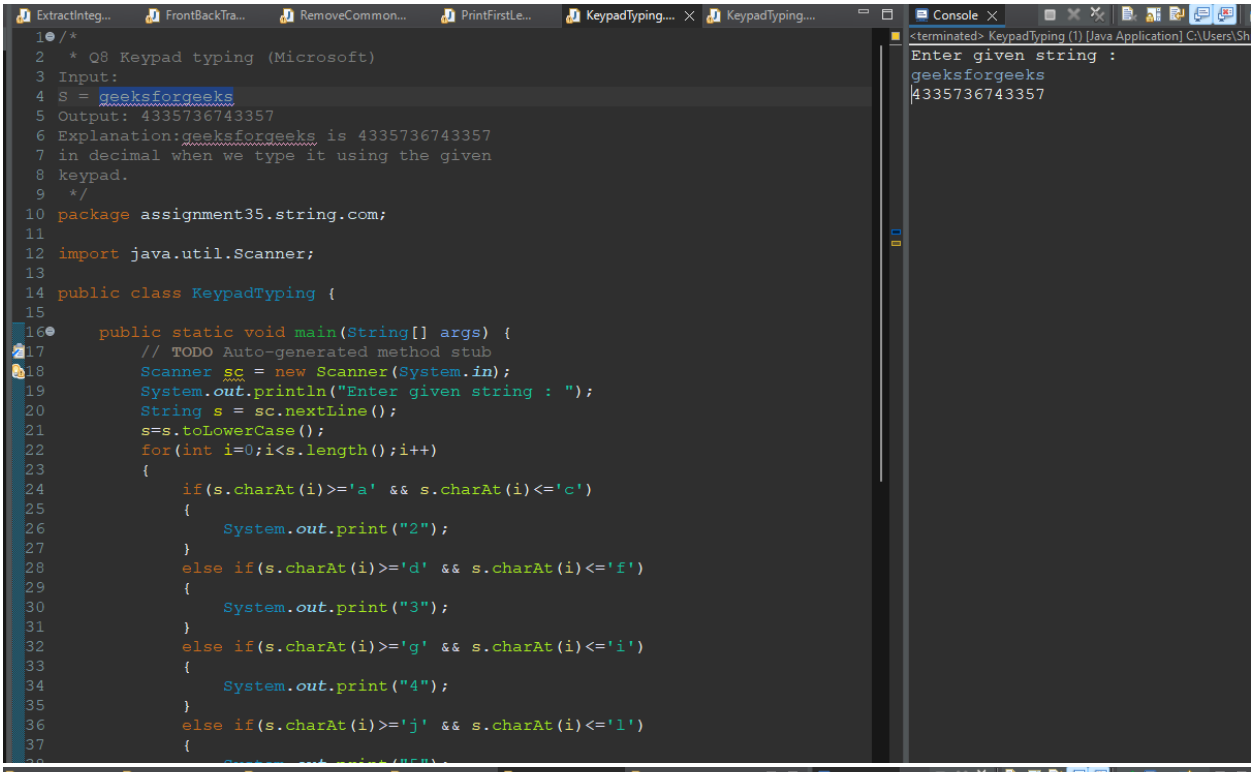
Output: gfg



Q8 Keypad typing (Microsoft)

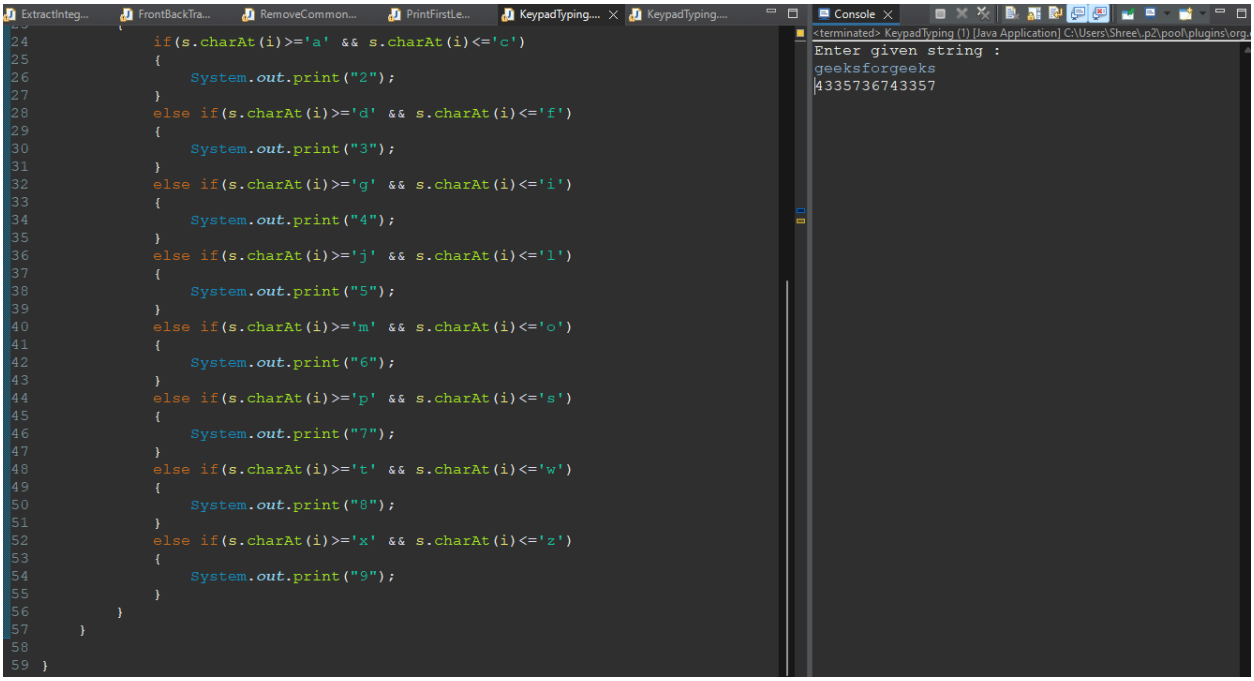
Input: S = geeksforgeeks Output: 4335736743357

Explanation:geeksforgeeks is 4335736743357 in decimal when we type it using the given keypad.



```
/*
 * Q8 Keypad typing (Microsoft)
Input:
S = geeksforgeeks
Output: 4335736743357
Explanation:geeksforgeeks is 4335736743357
in decimal when we type it using the given
keypad.
 */
package assignment35.string.com;

import java.util.Scanner;

public class KeypadTyping {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter given string : ");
        String s = sc.nextLine();
        s=s.toLowerCase();
        for(int i=0;i<s.length();i++)
        {
            if(s.charAt(i)>='a' && s.charAt(i)<='c')
            {
                System.out.print("2");
            }
            else if(s.charAt(i)>='d' && s.charAt(i)<='f')
            {
                System.out.print("3");
            }
            else if(s.charAt(i)>='g' && s.charAt(i)<='i')
            {
                System.out.print("4");
            }
            else if(s.charAt(i)>='j' && s.charAt(i)<='l')
            {
```

```
Console
<terminated> KeypadTyping (1) [Java Application] C:\Users\Sh
Enter given string :
geeksforgeeks
4335736743357
```



```
            if(s.charAt(i)>='a' && s.charAt(i)<='c')
            {
                System.out.print("2");
            }
            else if(s.charAt(i)>='d' && s.charAt(i)<='f')
            {
                System.out.print("3");
            }
            else if(s.charAt(i)>='g' && s.charAt(i)<='i')
            {
                System.out.print("4");
            }
            else if(s.charAt(i)>='j' && s.charAt(i)<='l')
            {
                System.out.print("5");
            }
            else if(s.charAt(i)>='m' && s.charAt(i)<='o')
            {
                System.out.print("6");
            }
            else if(s.charAt(i)>='p' && s.charAt(i)<='s')
            {
                System.out.print("7");
            }
            else if(s.charAt(i)>='t' && s.charAt(i)<='w')
            {
                System.out.print("8");
            }
            else if(s.charAt(i)>='x' && s.charAt(i)<='z')
            {
                System.out.print("9");
            }
        }
    }

}
```

```
Console
<terminated> KeypadTyping (1) [Java Application] C:\Users\Shree\.p2\pool\plugins\org.
Enter given string :
geeksforgeeks
4335736743357
```

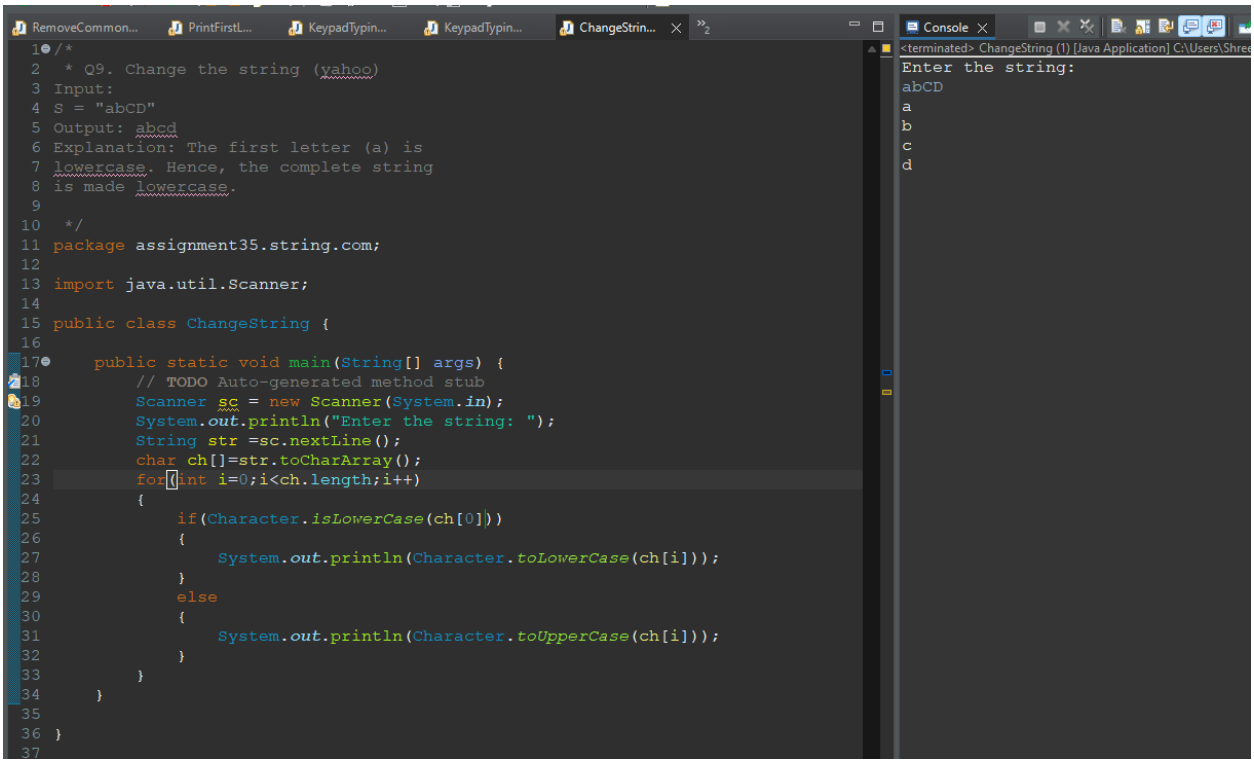
Q9. Change the string (yahoo)

Input:

S = "abCD"

Output: abcd

Explanation: The first letter (a) is lowercase. Hence, the complete string is made lowercase.



```
/*
 * Q9. Change the string (yahoo)
Input:
S = "abCD"
Output: abcd
Explanation: The first letter (a) is
lowercase. Hence, the complete string
is made lowercase.

 */
package assignment35.string.com;

import java.util.Scanner;

public class ChangeString {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the string: ");
        String str =sc.nextLine();
        char ch[]=str.toCharArray();
        for(int i=0;i<ch.length;i++)
        {
            if(Character.isLowerCase(ch[0]))
            {
                System.out.println(Character.toLowerCase(ch[i]));
            }
            else
            {
                System.out.println(Character.toUpperCase(ch[i]));
            }
        }
    }

}
```

```
Enter the string:
abCD
a
b
c
d
```

Q10. Good or Bad string (Amazon)

Input:

S = aeioup??

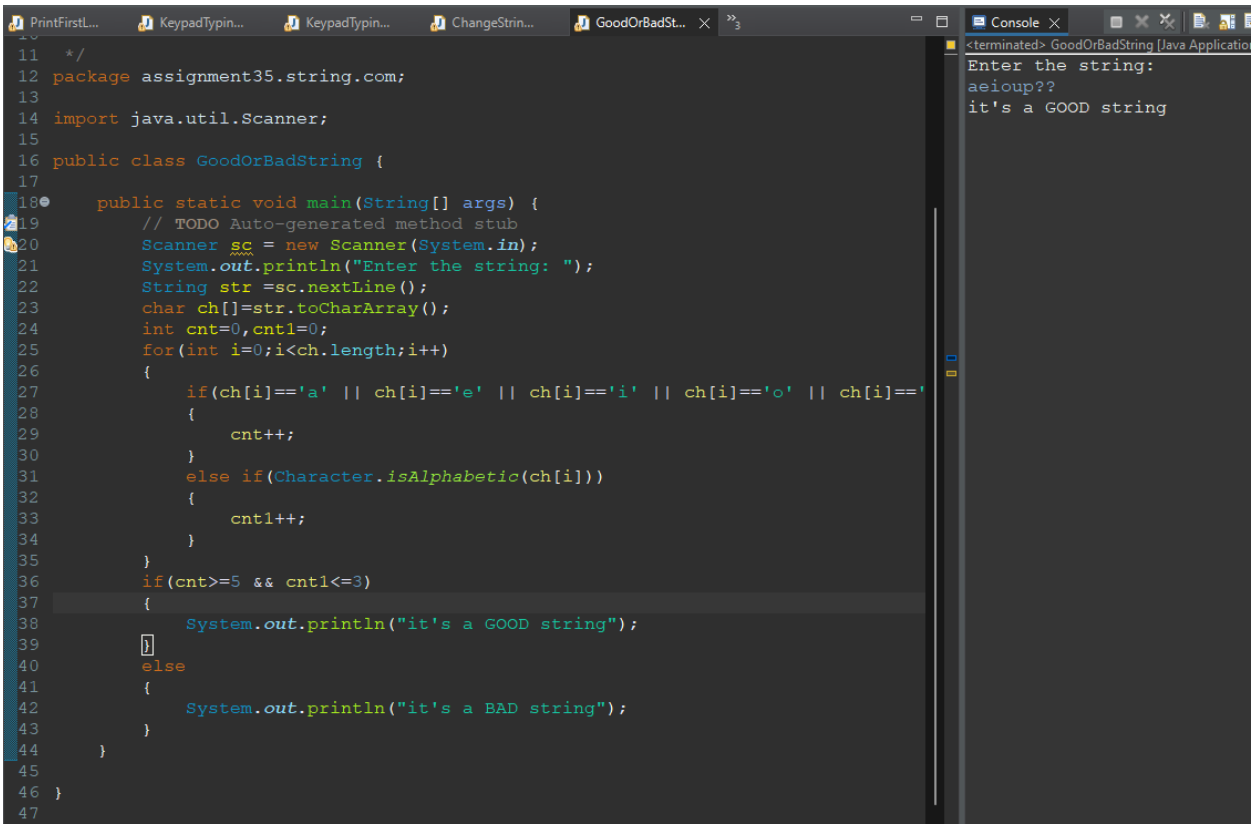
Output:

1

Explanation: The String doesn't contain more than

3 consonants or more than 5 vowels together. So,

it's a GOOD string.



```
 */
package assignment35.string.com;

import java.util.Scanner;

public class GoodOrBadString {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the string: ");
        String str =sc.nextLine();
        char ch[]=str.toCharArray();
        int cnt=0,cnt1=0;
        for(int i=0;i<ch.length;i++)
        {
            if(ch[i]=='a' || ch[i]=='e' || ch[i]=='i' || ch[i]=='o' || ch[i]=='
            {
                cnt++;
            }
            else if(Character.isAlphabetic(ch[i]))
            {
                cnt1++;
            }
        }
        if(cnt>=5 && cnt1<=3)
        {
            System.out.println("it's a GOOD string");
        }
        else
        {
            System.out.println("it's a BAD string");
        }
    }

}
```

```
<terminated> GoodOrBadString [Java Applicatio
Enter the string:
aeioup??
it's a GOOD string
```

Q11. Twice Counter

Input:

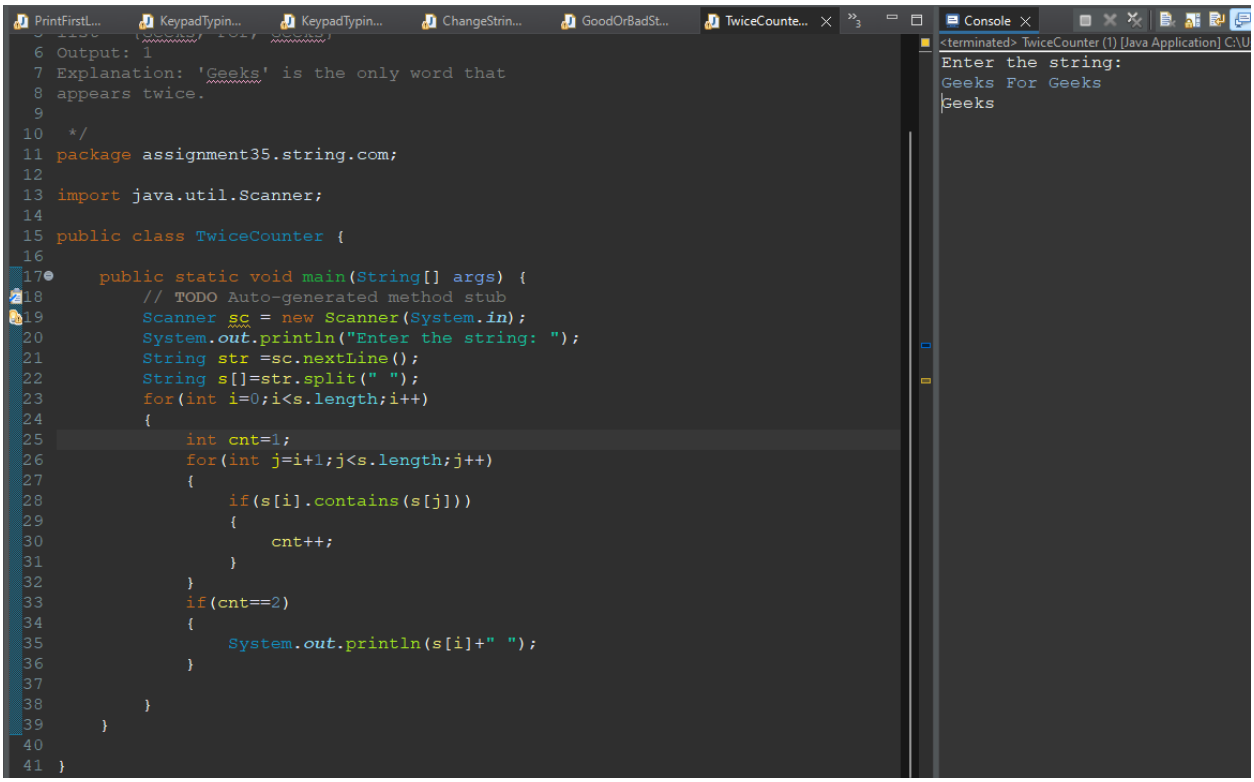
N = 3

list = {Geeks, For, Geeks}

Output: 1

Explanation: 'Geeks' is the only word that

appears twice.



Q12. Rearrange a string

Example 1:

Input: S = "AC2BEW3"

Output: "ABCEW5"

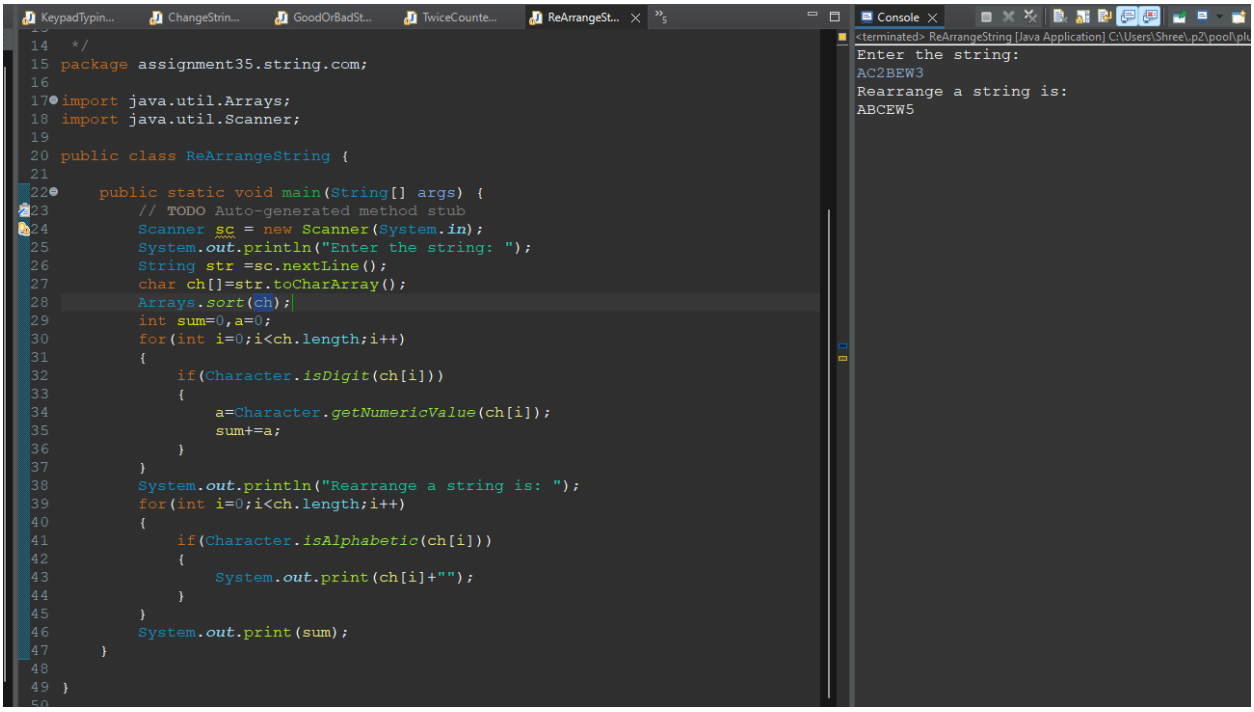
Explanation: 2 + 3 = 5 and we print all alphabets in the lexicographical order.

Example 2:

Input: S = "ACCBA10D2EW30"

Output: "AABCCDEW6"

Explanation: 0+1+2+3 = 6 and we print all alphabets in the lexicographical order.



```
 */
package assignment35.string.com;

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

public class ReArrangeString {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the string: ");
        String str =sc.nextLine();
        char ch[]=str.toCharArray();
        Arrays.sort(ch);
        int sum=0,a=0;
        for(int i=0;i<ch.length;i++)
        {
            if(Character.isDigit(ch[i]))
            {
                a=Character.getNumericValue(ch[i]);
                sum+=a;
            }
        }
        System.out.println("Rearrange a string is: ");
        for(int i=0;i<ch.length;i++)
        {
            if(Character.isAlphabetic(ch[i]))
            {
                System.out.print(ch[i]+"");
            }
        }
        System.out.print(sum);
    }

}
```

```
Enter the string:
AC2BEW3
Rearrange a string is:
ABCEW5
```

Q13. Easy string (google)

Example 1:

Input: S = "aaABBb"

Output: "3a3b"

Explanation: As 'a' appears 3 times

consecutively and 'b' also 3 times,

and 'b' and 'B' considered as same.

Example 2:

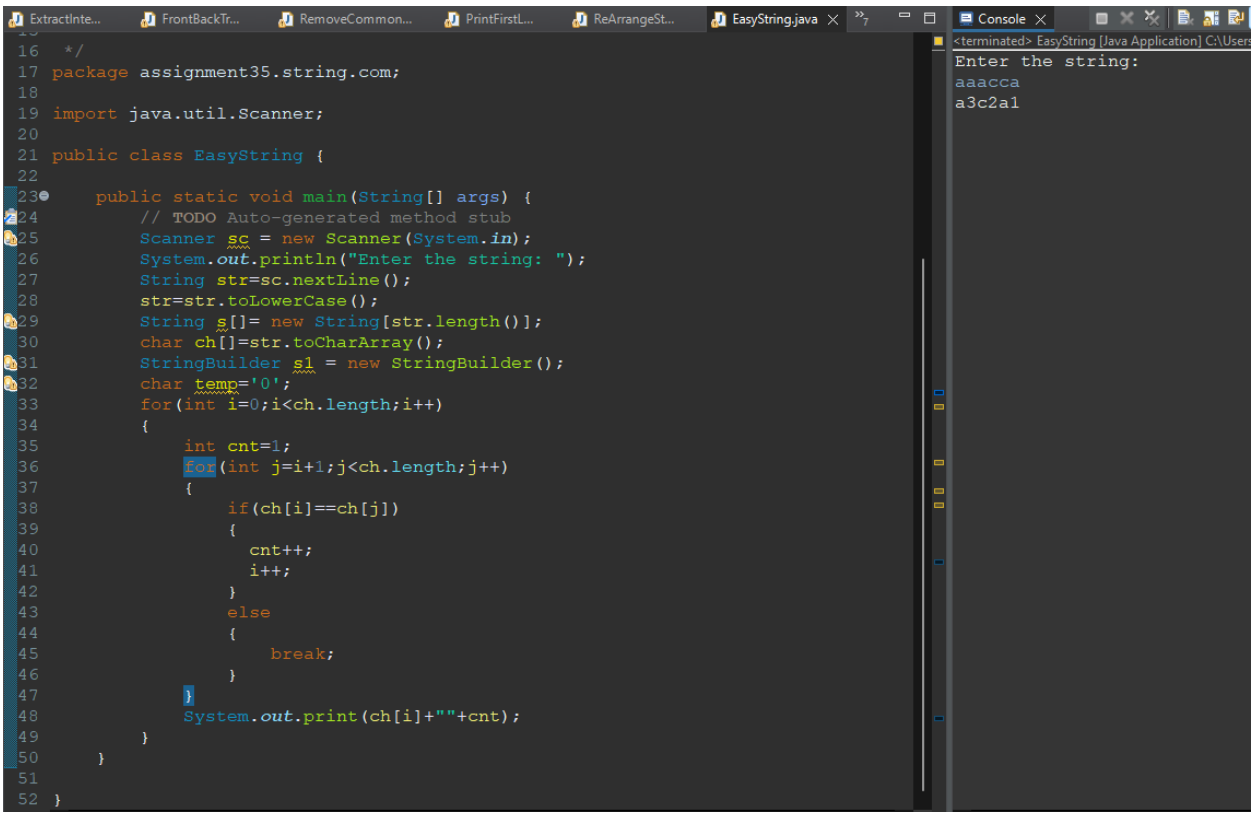
Input: S = "aaacca"

Output: "3a2c1a"

Explanation: As 'a' appears 3 times

consecutively and 'c' also 2 times,

and then 'a' 1 time.




```java
 */
package assignment35.string.com;

import java.util.Scanner;

public class EasyString {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the string: ");
        String str=sc.nextLine();
        str=str.toLowerCase();
        String s[]= new String[str.length()];
        char ch[]=str.toCharArray();
        StringBuilder s1 = new StringBuilder();
        char temp='0';
        for(int i=0;i<ch.length;i++)
        {
            int cnt=1;
            for(int j=i+1;j<ch.length;j++)
            {
                if(ch[i]==ch[j])
                {
                  cnt++;
                  i++;
                }
                else
                {
                    break;
                }
            }
            System.out.print(ch[i]+""+cnt);
        }
    }

}
```

Q14. Special array reversal (google)

Example 1:

Input: S = "A&B"

Output: "B&A"

Explanation: As we ignore '&' and

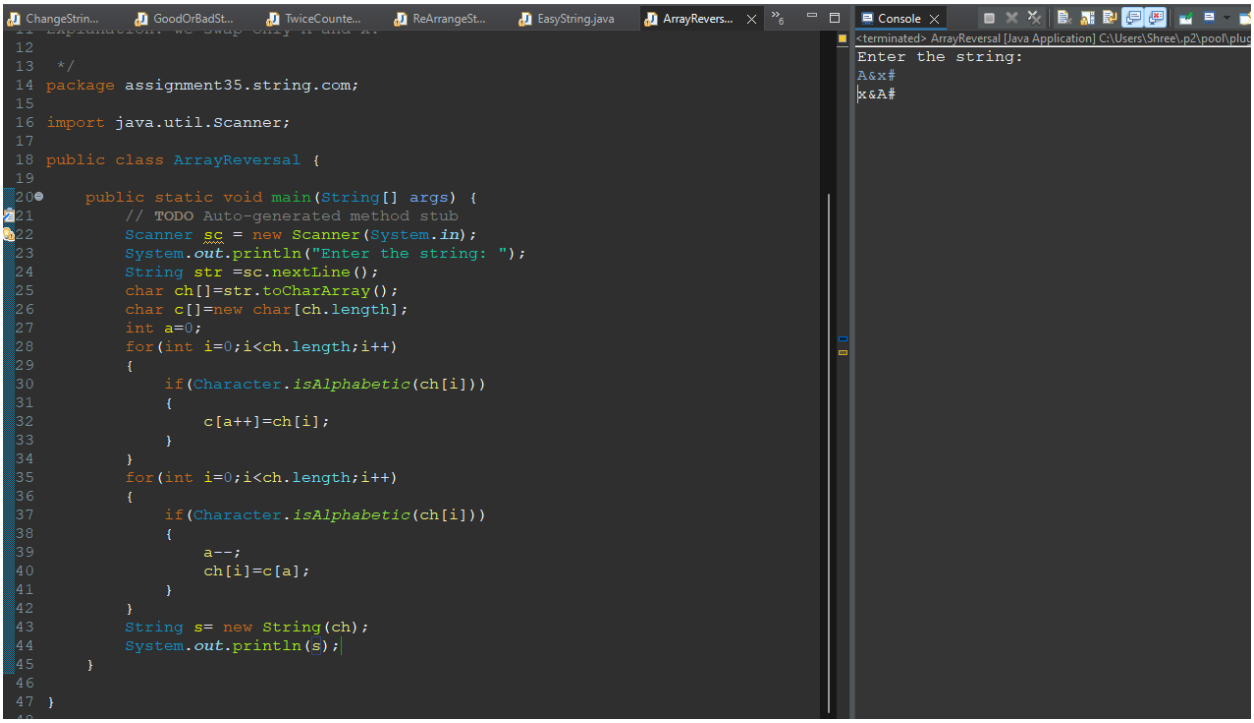
then reverse, so answer is "B&A".

Example 2:

Input: S = "A&x#

Output: "x&A#"

Explanation: we swap only A and x.



Q15. Find largest word in dictionary (MicroSoft)

Example 1:

Input: d = {"ale", "apple", "monkey", "plea"}

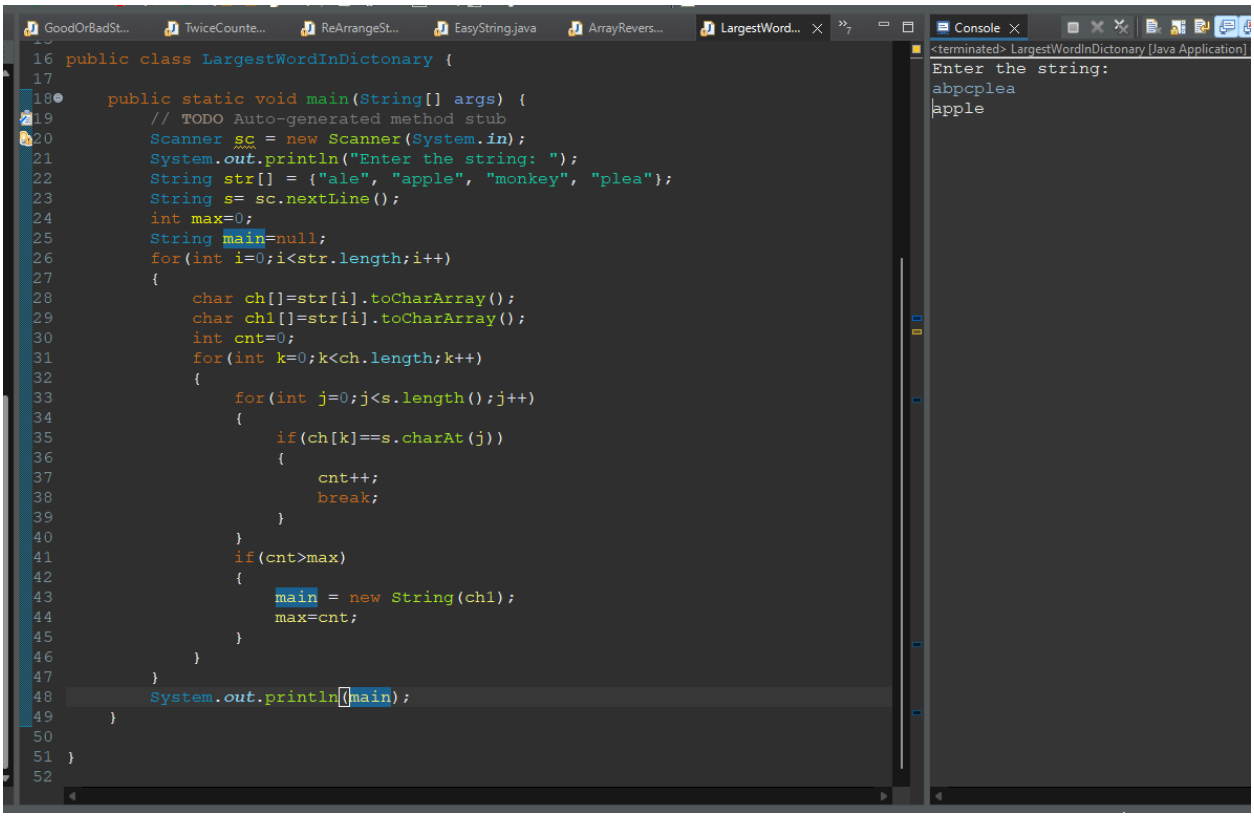
S = "abpcplea"

Output: "apple"

Explanation: After deleting "b", "c"

"a" S became "apple" which is present

in d.




```java
public class LargestWordInDictonary {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the string: ");
        String str[] = {"ale", "apple", "monkey", "plea"};
        String s= sc.nextLine();
        int max=0;
        String main=null;
        for(int i=0;i<str.length;i++)
        {
            char ch[]=str[i].toCharArray();
            char ch1[]=str[i].toCharArray();
            int cnt=0;
            for(int k=0;k<ch.length;k++)
            {
                for(int j=0;j<s.length();j++)
                {
                    if(ch[k]==s.charAt(j))
                    {
                        cnt++;
                        break;
                    }
                }
                if(cnt>max)
                {
                    main = new String(ch1);
                    max=cnt;
                }
            }
        }
        System.out.println(main);
    }

}
```

Q16. Odd Even Problem (oracle)

Given a string S of lowercase english characters, find out whether the summation of X and Y is

even or odd, where X is the count of characters which occupy even positions in english

alphabets and have positive even frequency, and Y is the count of characters which occupy odd

positions in english alphabets and have positive odd frequency.

Note: Positive means greater than zero.

Example 1:

Input: S = "abbbcc"

Output: "ODD"

Explanation: X = 0 and Y = 1 so (X + Y) is

ODD. 'a' occupies 1st place(odd) in english

alphabets and its frequency is odd(1), 'b'

occupies 2nd place(even) but its frequency

is odd(3) so it doesn't get counted and 'c'

occupies 3rd place(odd) but its frequency

is even(2) so it also doesn't get counted.

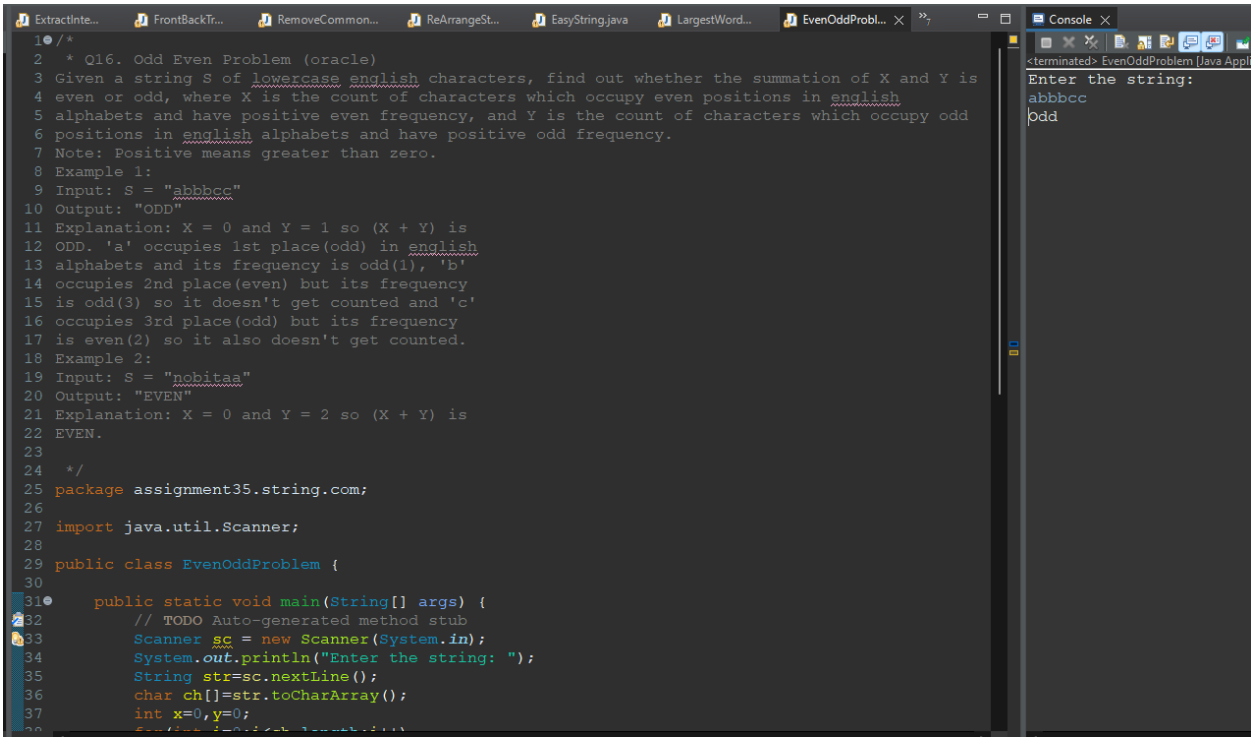
Example 2:

Input: S = "nobitaa"

Output: "EVEN"

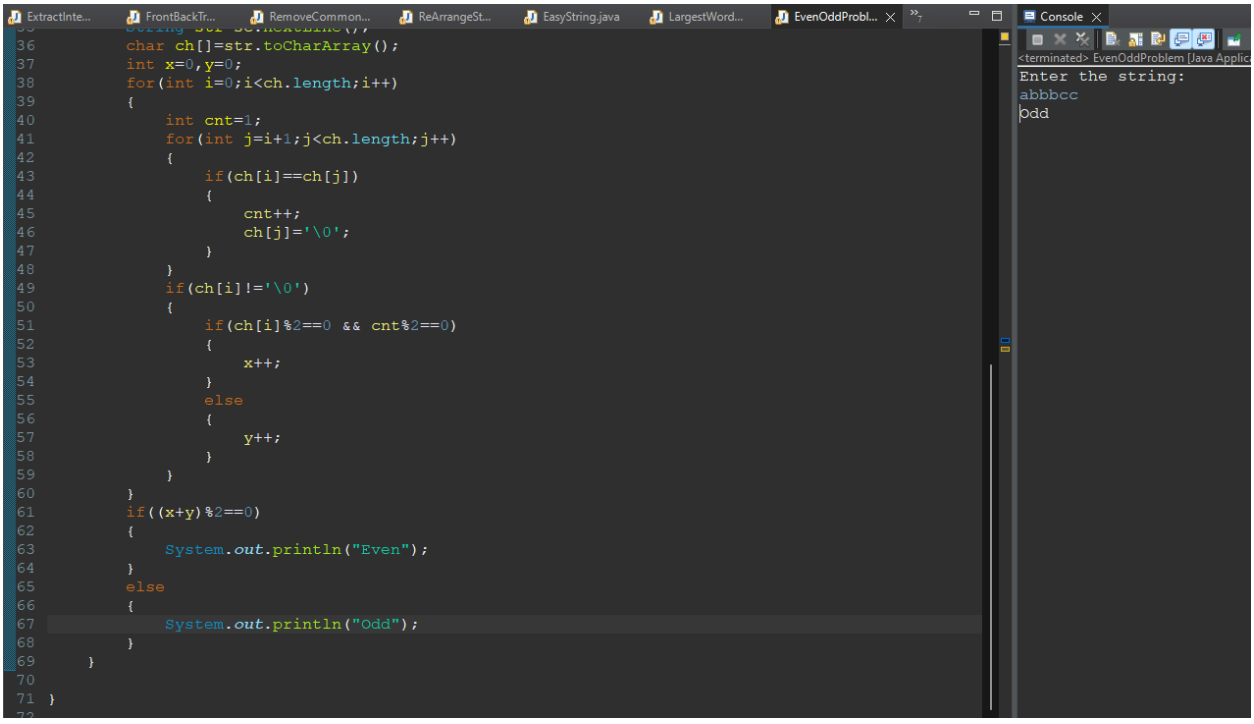
Explanation: X = 0 and Y = 2 so (X + Y) is

EVEN.



```
/*
 * Q16. Odd Even Problem (oracle)
Given a string S of lowercase english characters, find out whether the summation of X and Y is
even or odd, where X is the count of characters which occupy even positions in english
alphabets and have positive even frequency, and Y is the count of characters which occupy odd
positions in english alphabets and have positive odd frequency.
Note: Positive means greater than zero.
Example 1:
Input: S = "abbbcc"
Output: "ODD"
Explanation: X = 0 and Y = 1 so (X + Y) is
ODD. 'a' occupies 1st place(odd) in english
alphabets and its frequency is odd(1), 'b'
occupies 2nd place(even) but its frequency
is odd(3) so it doesn't get counted and 'c'
occupies 3rd place(odd) but its frequency
is even(2) so it also doesn't get counted.
Example 2:
Input: S = "nobitaa"
Output: "EVEN"
Explanation: X = 0 and Y = 2 so (X + Y) is
EVEN.

 */
package assignment35.string.com;

import java.util.Scanner;

public class EvenOddProblem {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the string: ");
        String str=sc.nextLine();
        char ch[]=str.toCharArray();
        int x=0,y=0;
```

```
Console
<terminated> EvenOddProblem [Java Appl
Enter the string:
abbbcc
odd
```



```
        char ch[]=str.toCharArray();
        int x=0,y=0;
        for(int i=0;i<ch.length;i++)
        {
            int cnt=1;
            for(int j=i+1;j<ch.length;j++)
            {
                if(ch[i]==ch[j])
                {
                    cnt++;
                    ch[j]='\0';
                }
            }
            if(ch[i]!='\0')
            {
                if(ch[i]%2==0 && cnt%2==0)
                {
                    x++;
                }
                else
                {
                    y++;
                }
            }
        }
        if((x+y)%2==0)
        {
            System.out.println("Even");
        }
        else
        {
            System.out.println("Odd");
        }
    }

}
```

```
Console
<terminated> EvenOddProblem [Java Applic
Enter the string:
abbbcc
odd
```

Q17. Replace a word

Example 1:Input: S = "xxforxx xx for xx" ,oldW = "xx" ,ewW = "Geeks"

Output: "geeksforgeeks geeks for geeks"

Explanation: Replacing each "xx" with" Geeks" in S.

Example 2:

Input:

S = "india is the xx country"

oldW = "xx"

newW = "best"

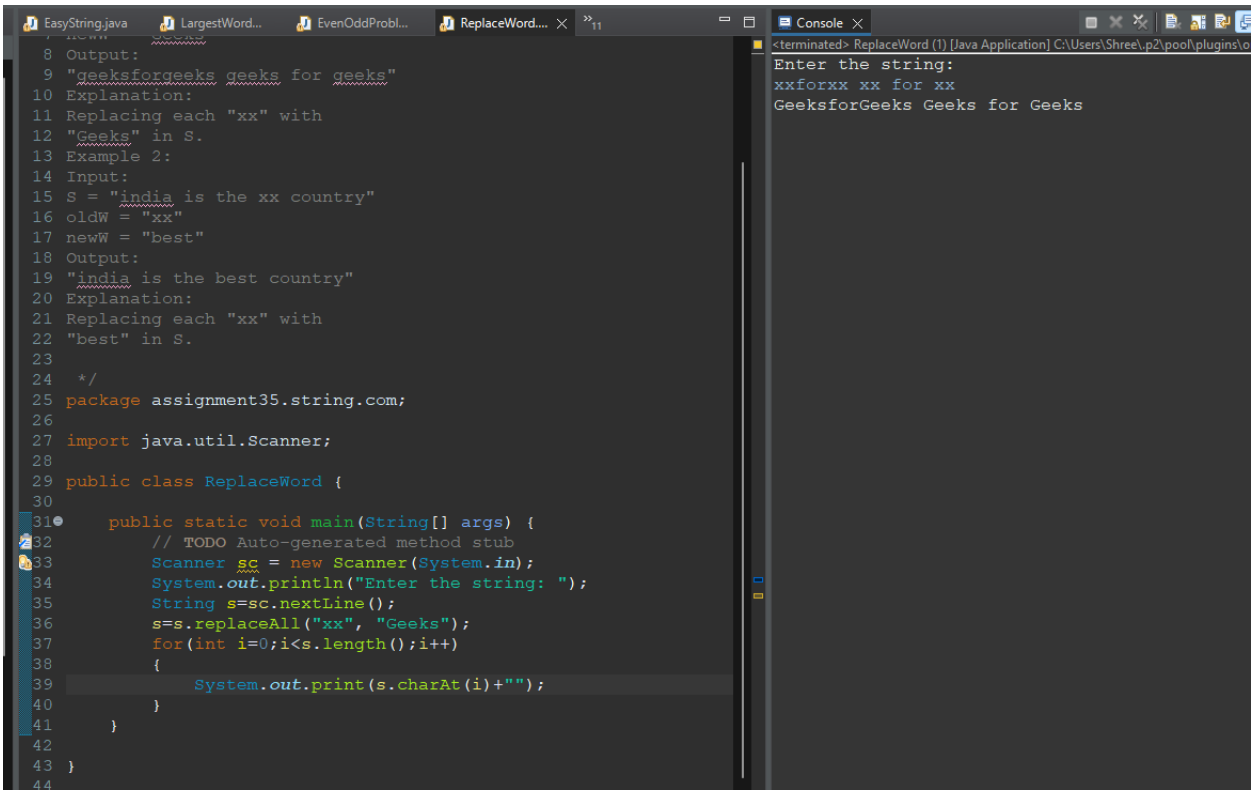
Output:

"india is the best country"

Explanation:

Replacing each "xx" with

"best" in S.



Q18. Ordering of strings (walmart)

You will be given N number of strings. You have to find the lexicographically smallest string and

the lexicographically largest string among these strings.

Example 1:

Input:

N = 3

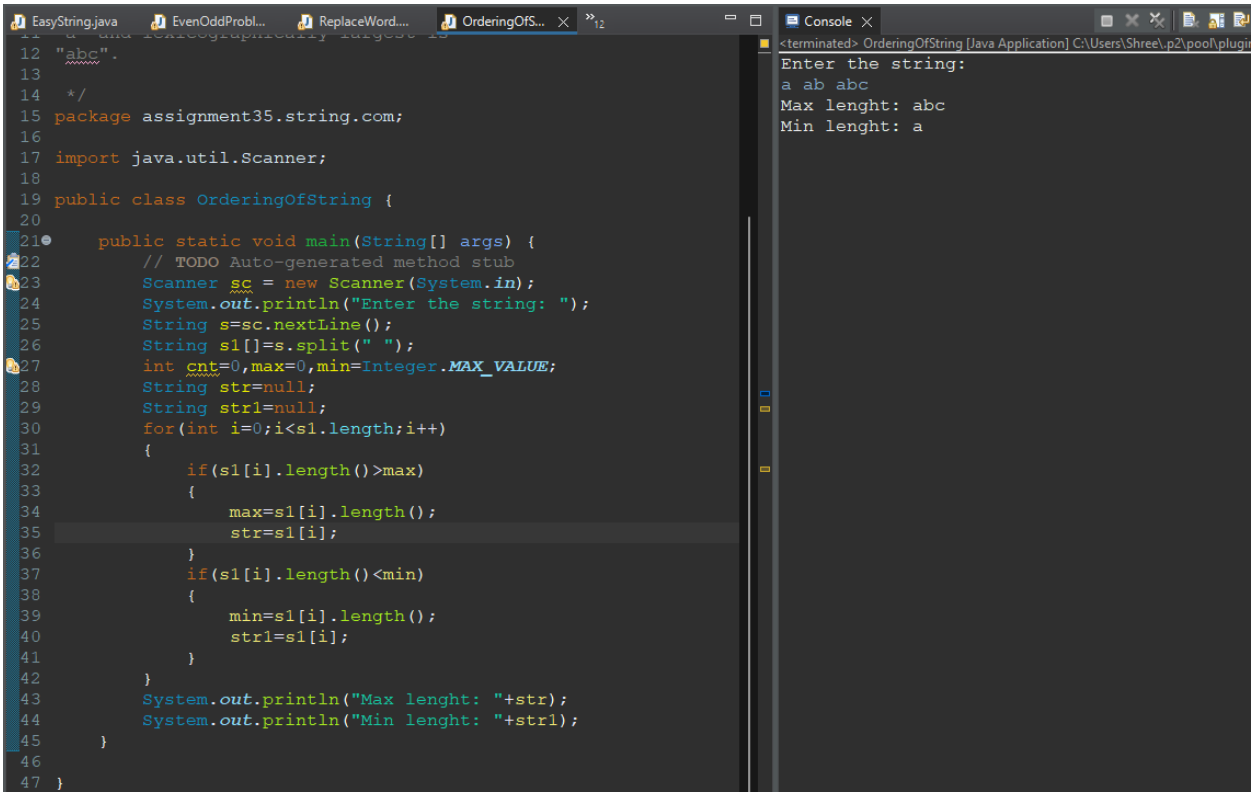
strings = a , ab , abc

Output: a abc

Explanation: Lexicographically smallest is

"a" and lexicographically largest is

"abc".




```
"abc".

 */
package assignment35.string.com;

import java.util.Scanner;

public class OrderingOfString {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the string: ");
        String s=sc.nextLine();
        String s1[]=s.split(" ");
        int cnt=0,max=0,min=Integer.MAX_VALUE;
        String str=null;
        String str1=null;
        for(int i=0;i<s1.length;i++)
        {
            if(s1[i].length()>max)
            {
                max=s1[i].length();
                str=s1[i];
            }
            if(s1[i].length()<min)
            {
                min=s1[i].length();
                str1=s1[i];
            }
        }
        System.out.println("Max lenght: "+str);
        System.out.println("Min lenght: "+str1);
    }

}
```

Q19. Same characters in two strings

Given two strings A and B of equal length, find how many times the corresponding position in

the two strings hold exactly the same character. The comparison should not be case sensitive.

Example 1:

Input:

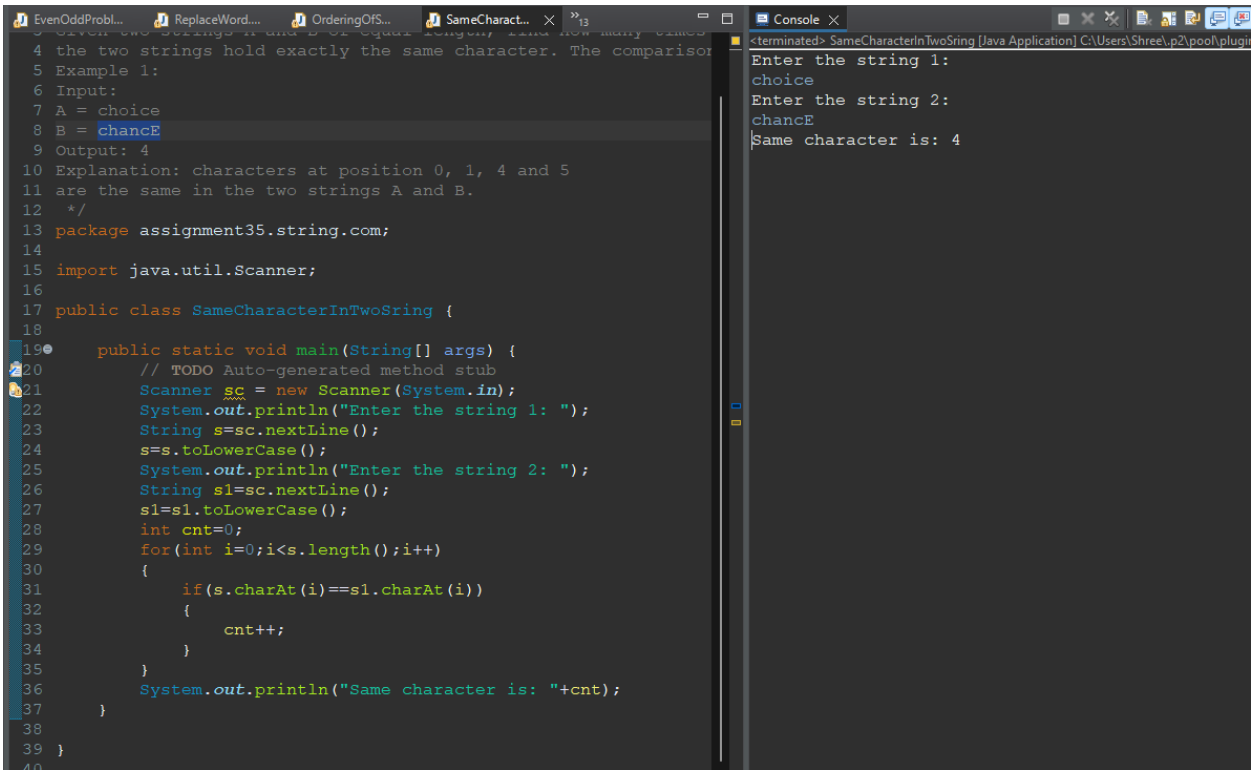
A = choice

B = chancE

Output: 4

Explanation: characters at position 0, 1, 4 and 5

are the same in the two strings A and B.



```
 4 the two strings hold exactly the same character. The comparisor
 5 Example 1:
 6 Input:
 7 A = choice
 8 B = chancE
 9 Output: 4
10 Explanation: characters at position 0, 1, 4 and 5
11 are the same in the two strings A and B.
12  */
13 package assignment35.string.com;
14
15 import java.util.Scanner;
16
17 public class SameCharacterInTwoSring {
18
19     public static void main(String[] args) {
20         // TODO Auto-generated method stub
21         Scanner sc = new Scanner(System.in);
22         System.out.println("Enter the string 1: ");
23         String s=sc.nextLine();
24         s=s.toLowerCase();
25         System.out.println("Enter the string 2: ");
26         String s1=sc.nextLine();
27         s1=s1.toLowerCase();
28         int cnt=0;
29         for(int i=0;i<s.length();i++)
30         {
31             if(s.charAt(i)==s1.charAt(i))
32             {
33                 cnt++;
34             }
35         }
36         System.out.println("Same character is: "+cnt);
37     }
38
39 }
```

```
<terminated> SameCharacterInTwoSring [Java Application] C:\Users\Shree\.p2\pool\plugi
Enter the string 1:
choice
Enter the string 2:
chancE
Same character is: 4
```

Q20. Maximum number of characters between any two same character

Given a string containing lower and uppercase alphabets, the task is to find the maximum number of characters between any two same characters in the string.

Example 1:

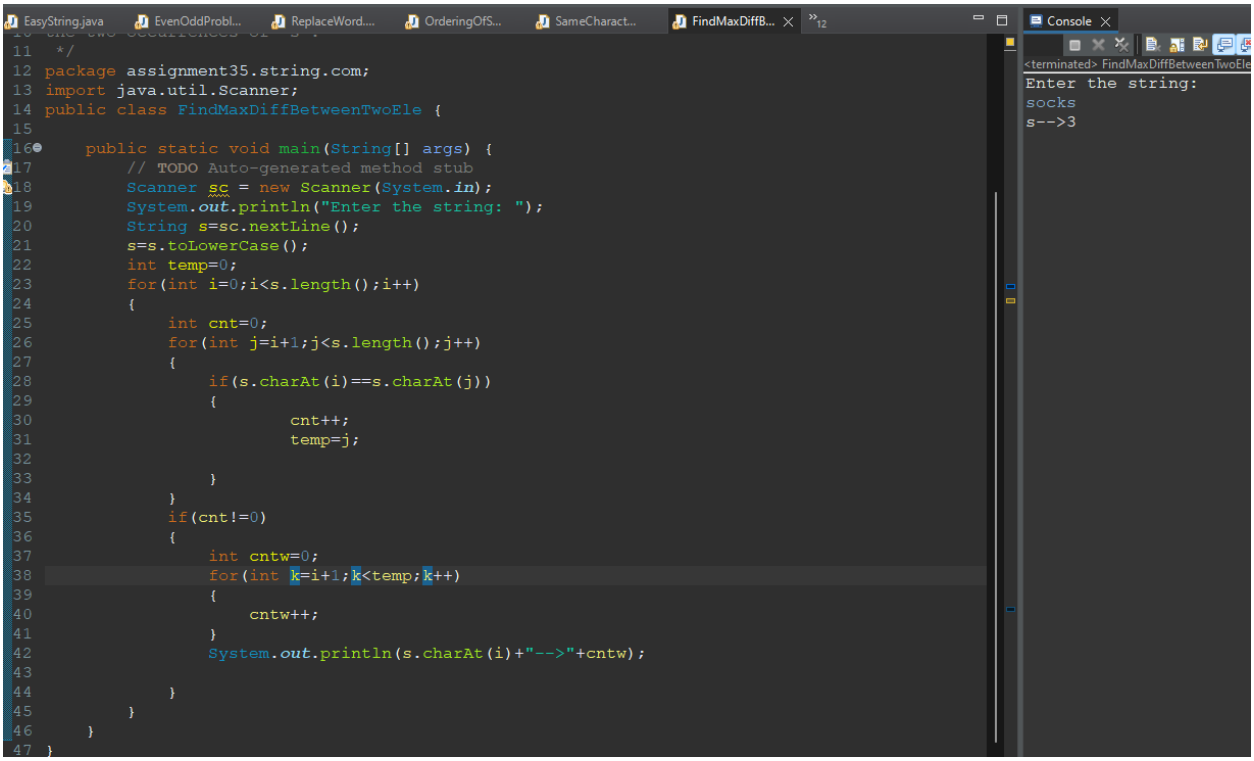
Input:

S = "socks"

Output: 3

Explanation: There are 3 characters between

the two occurrences of 's'.



```
 */
package assignment35.string.com;
import java.util.Scanner;
public class FindMaxDiffBetweenTwoEle {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the string: ");
        String s=sc.nextLine();
        s=s.toLowerCase();
        int temp=0;
        for(int i=0;i<s.length();i++)
        {
            int cnt=0;
            for(int j=i+1;j<s.length();j++)
            {
                if(s.charAt(i)==s.charAt(j))
                {
                        cnt++;
                        temp=j;

                }
            }
            if(cnt!=0)
            {
                int cntw=0;
                for(int k=i+1;k<temp;k++)
                {
                    cntw++;
                }
                System.out.println(s.charAt(i)+"-->"+cntw);

            }
        }
    }
}
```

```
Console
<terminated> FindMaxDiffBetweenTwoEle
Enter the string:
socks
s-->3
```

Q21. Last Match

Given two strings A and B, you need to find the last occurrence (1 based indexing) of string B in

string A.

Example 1:

Input:

A = abcdefghijklghifghsd

B = ghi

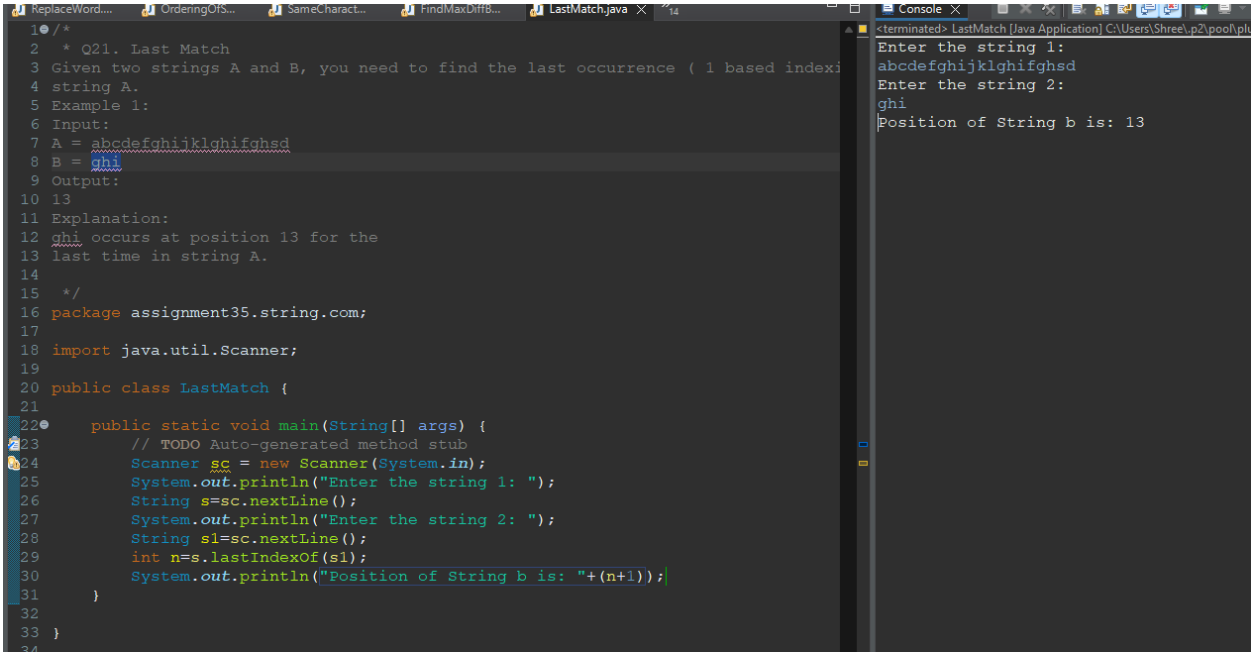
Output:

13

Explanation:

ghi occurs at position 13 for the

last time in string A.




```
/*
 * Q21. Last Match
Given two strings A and B, you need to find the last occurrence ( 1 based indexi
string A.
Example 1:
Input:
A = abcdefghijklghifghsd
B = ghi
Output:
13
Explanation:
ghi occurs at position 13 for the
last time in string A.

 */
package assignment35.string.com;

import java.util.Scanner;

public class LastMatch {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the string 1: ");
        String s=sc.nextLine();
        System.out.println("Enter the string 2: ");
        String s1=sc.nextLine();
        int n=s.lastIndexOf(s1);
        System.out.println("Position of String b is: "+(n+1));
    }

}
```

Q22. Difficulty of sentence

Given a sentence as a string S. Calculate difficulty of a given sentence.

Difficulty of sentence is defined as 5*(number of hard words) + 3*(number of easy words). A

word in the given string is considered hard if it has 4 consecutive consonants or the number of

consonants is more than the number of vowels. Else the word is easy.

Note: uppercase and lowercase characters are the same.

Example

Input: S = "Difficulty of sentence"

Output: 13

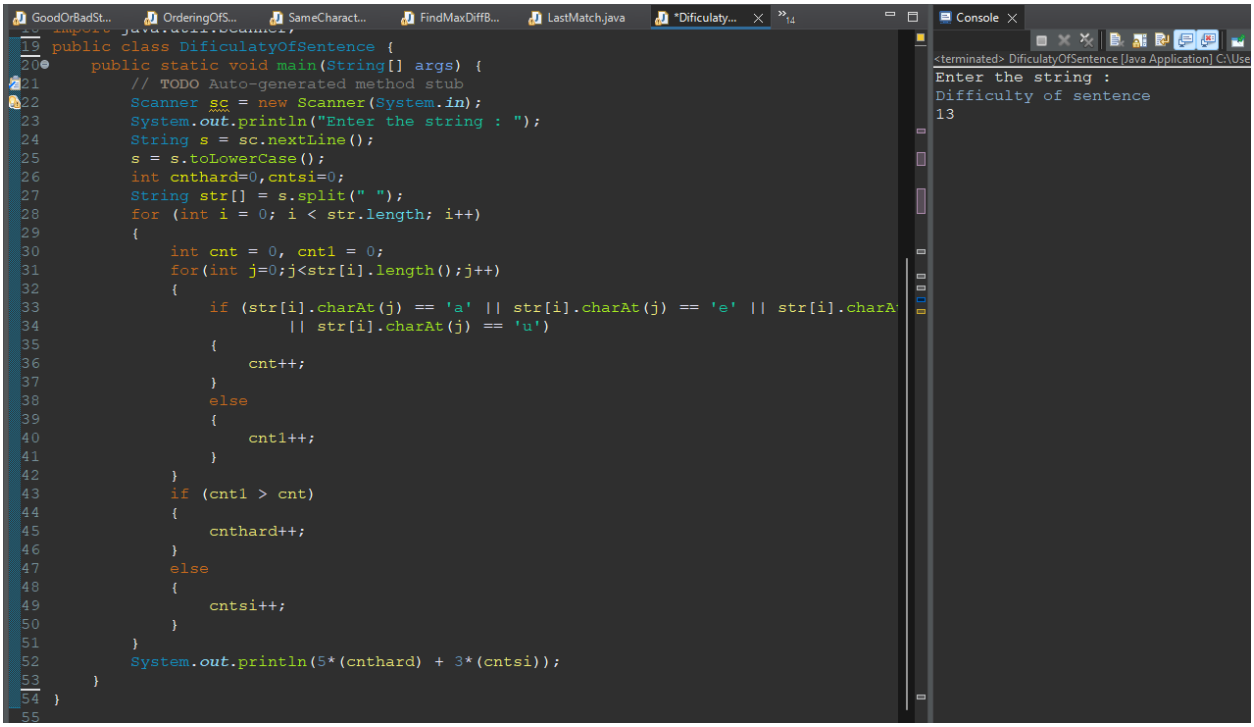
Explanation: 2 hard words + 1 easy word

Example

Input: S = "I am good"

Output: 9

Explanation: 3 easy word




```java
public class DificulatyOfSentence {
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the string : ");
        String s = sc.nextLine();
        s = s.toLowerCase();
        int cnthard=0,cntsi=0;
        String str[] = s.split(" ");
        for (int i = 0; i < str.length; i++)
        {
            int cnt = 0, cnt1 = 0;
            for(int j=0;j<str[i].length();j++)
            {
                if (str[i].charAt(j) == 'a' || str[i].charAt(j) == 'e' || str[i].charA
                        || str[i].charAt(j) == 'u')
                {
                    cnt++;
                }
                else
                {
                    cnt1++;
                }
            }
            if (cnt1 > cnt)
            {
                cnthard++;
            }
            else
            {
                cntsi++;
            }
        }
        System.out.println(5*(cnthard) + 3*(cntsi));
    }
}
```