

## STRINGS SOLUTIONS

Solution 1:

```
import java.util.*;  
  
public class Solution {  
    public static void main(String[] args) {  
        String str = new Scanner(System.in).next();  
        int count = 0;  
  
        for(int i=0; i<str.length(); i++) {  
            char ch = str.charAt(i);  
            if(ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' || ch == 'u') {  
                count++;  
            }  
        }  
        System.out.println("count of vowels is :" + count);  
    }  
}
```

COLLEGE

Solution 2: Output will be :

false true

(If you need an explanation, please r-ewatch the video about how Strings work in memory?)

Solution 3 : Output will be :

ApnaCoege

Following are some methods in Java which are used to replace characters:

String	<b>replace(char oldChar, char newChar)</b> Returns a new string resulting from replacing all occurrences of oldChar in this string with newChar.
String	<b>replace(CharSequence target, CharSequence replacement)</b> Replaces each substring of this string that matches the literal target sequence with the specified literal replacement sequence.
String	<b>replaceAll(String regex, String replacement)</b> Replaces each substring of this string that matches the given regular expression with the given replacement.
String	<b>replaceFirst(String regex, String replacement)</b> Replaces the first substring of this string that matches the given regular expression with the given replacement.

er.dm.zer03@gmail.com

**Solution 4:**

```
import java.util.Arrays;
public class Solution {
    public static void main(String[] args) {
        String str1 = "earth";
        String str2 = "heart";

        //Convert Strings to lowercase. Why? so that we don't have to check
separately for lower & uppercase.
        str1 = str1.toLowerCase();
        str2 = str2.toLowerCase();

        // First check - if the lengths are the same
        if(str1.length() == str2.length()) {
            // convert strings into char array
            char[] str1charArray = str1.toCharArray();
            char[] str2charArray = str2.toCharArray();
            // sort the char array
            Arrays.sort(str1charArray);
            Arrays.sort(str2charArray);
            // if the sorted char arrays are same or identical then the strings are
anagram
            boolean result = Arrays.equals(str1charArray, str2charArray);
            if(result) {
                System.out.println(str1 + " and " + str2 + " are anagrams of each
other.");
            } else {
                System.out.println(str1 + " and " + str2 + " are not anagrams of
each other.");
            }
        } else {
            // case when lengths are not equal
            System.out.println(str1 + " and " + str2 + " are not anagrams of each
other.");
        }
    }
}
```

er.dlm.zer03@gmail.com