

Forum: [Hack Forums](#)

Thread: [Using stringtokenizer to count the number of vowels for each word](#)

Axanas - May 26, 2017, 7:29 pm

Hey everyone. I'm trying to use StringTokenizer to:

- 1) count the number of vowels in each word of a sentence
- 2) output the word that has the most vowels

I could write a countVowel method that counts the vowels at a particular index position, but I don't think I'm allowed to use index positions with StringTokenizer? Do I need to convert the string into an array, or is there a simple way of doing this with StringTokenizer?

Untraceable_Pro - May 27, 2017, 12:25 am

Count the number of vowels in each word. Have a variable which holds the index of the word with max vowels and another variable which holds the no of vowels in that word. Now use a loop to go through each word , count vowels in each word .

if vowels is greater than the word with max vowel , update both variables.

randomized patterns - May 27, 2017, 1:12 am

Should have checked java docs,

<https://docs.oracle.com/javase/7/docs/api/java/util/StringTokenizer.html>

```
Code StringTokenizer st = new StringTokenizer("this is a test");
while (st.hasMoreTokens()) {
    System.out.println(st.nextToken());
}
```

However, StringTokenizer seems to be deprecated, it is encouraged to use split or regex.

Your implementation should look something like this with the split method.

Code public class Vowel{

```
    public static void main(String[] args){
        String[] tokens = "This is a test Ackeiouuu bam AAAAAAaaaaAAAAA".split("\\s");
        String mostVowels = null;
        int vowelCount = 0;

        for(int i = 0; i < tokens.length; i++){
            int temp = tokens[i].replaceAll("[^aeiouAEIOU]", "").length(); //regex magic
            if(temp > vowelCount){
                vowelCount = temp;
                mostVowels = tokens[i];
            }
        }

        System.out.println("Most vowels: " + mostVowels);
    }
}
```

This is a very minimalistic implementation that should suit your needs.

randomized patterns - May 27, 2017, 7:14 am

(May 27, 2017 05:50 AM)XAKE? Wrote: [\[CITE\]](#) Based on Oracle website, StringTokenizer is a deprecated class and you are supposed to use .split() method instead, you would do it like this:

- Get a sentence input from user
- Assign the input to a string array using .split() and ' ' (space) as a delimiter
- Declare max vowels variable and initialize it to 0
- Loop through your array of strings and then loop through each of the words and count vowels, you can do something like word.charAt(x).matches("[aoeiu]"), and if this returns true increase the int variable that holds number of vowels in current word
- If your variable that holds number of vowels in current word is greater than max vowels variable you declared in step 3, assign this new value to the max vowels variable.

Print out the max number of vowels in the single word and the word itself if needed. If you have problems doing this, reply here and I will post a complete code.

EDIT:

Here is your code, note that it doesn't deal with 2 or more words with same vowel count, it will return the first word with max vowel count, if you need this as well let me know:

Code package com.company;

import java.util.Scanner;

```
public class Main {
    public static void main(String[] args) {

        Scanner in = new Scanner(System.in);
        System.out.print("Enter a sentence: ");
        String sentence = in.nextLine();
        String[] words = sentence.split(" ");

        int maxVowels = 0;
        String wordWithMaxVowels = null;

        for (int x = 0; x < words.length; x++){
            int tempVowels = 0;
            for (int y = 0; y < words[x].length(); y++){
                if ((words[x].charAt(y) + "").matches("[AOEIUaoeiu]"))
                    tempVowels++;

                if (tempVowels > maxVowels) {
                    maxVowels = tempVowels;
                    wordWithMaxVowels = words[x];
                }
            }
        }

        System.out.println("Word with most vowels is: \"\" + wordWithMaxVowels + "\" and vowel count
```

in the given word is " + maxVowels);

```
}  
}
```

PASTEBIN (for better readability): <https://pastebin.com/GdErBesH>

Sample output: <http://prntscr.com/fcowj6>

This is not how you should use a regular expression.

randomized patterns - May 27, 2017, 8:22 am

(May 27, 2017 07:47 AM)XAKE? Wrote: [\[CITE\]](#) Please do go into more details.
Okay.

The point of regular expressions is for defining patterns to search for strings, in this case, vowels in a String (or as your inferior technical prowess refers to, a Sentence). The whole point of regex is to use your expression on a String, not individual characters in a String. It's counter intuitive. The expression you use (you know, the same one I used in my post but that's besides the point, there's only so many vowels so I expect people to use the same expression) would be able to find all vowels in ANY given string, so why use it on one character at a time?

The code you posted was literally a more autistic version of what I posted. And if you attempted to improve your implementation to be any less autistic, it would be a carbon copy of my implementation. So why post shit code when someone has already helped? It's pointless, if anything you are trying to teach OP bad programming practices.

randomized patterns - May 27, 2017, 9:13 am

(May 27, 2017 08:55 AM)XAKE? Wrote: [\[CITE\]](#) So much anger from a smelly tiny vagina, phew.

I didn't bother reading your code I never do, especially when a username is something that 5 year old girl would call herself. You didn't even satisfy what the OP said code should do, your code just outputs the word with most vowels. I smell so much hatred from your post I believe you need to be in a mental institution. Now apart from helping OP I wish to help you so I'll share this here just for you:

<https://www.zocdoc.com/psychiatrists>

I believe in you, you can make it, just because you're special doesn't mean you don't have equal rights ! ? ? ?

you "inferior" ball of diarrhea. :thereSholdBeAVomitEmoji:
Did I touch a nerve?

(May 27, 2017 08:55 AM)XAKE? Wrote: [\[CITE\]](#) You didn't even satisfy what the OP said code should do, your code just outputs the word with most vowels.

Your reading comprehension skills are very weak:

(May 26, 2017 07:29 PM)Axanas Wrote: [\[CITE\]](#) 1) count the number of vowels in each word of a sentence

2) output the word that has the most vowels

Un-surprisingly, my code does exactly what OP asked for, output the word that has the most vowels.

(May 27, 2017 08:55 AM)XAKE? Wrote: [\[CITE\]](#) Oh and sentence refers to the users input, end user

This is a programming section of a forum. I am going to assume you are in highschool, and have never been in an professional environment. When speaking to other programmers, you use technical vocabulary. "Sentence" is vague, and not technical. Use that in any sort of environment where you are speaking to other programmers and you will be laughed at, like you are right now.

randomized patterns - May 27, 2017, 9:25 am

You are not speaking to end users, you're speaking to people in the Java section of a programming sub-forum. When you use technical vocabulary in half your sentences, then "I just started programming two weeks ago" vocabulary in others, it just means you are an idiot.

<https://reddit.com/r/programminghorror>

(May 27, 2017 01:12 AM)Cat Girl Wrote: [\[CITE\]](#) Should have checked java docs,

```
Code StringTokenizer st = new StringTokenizer("this is a test");
    while (st.hasMoreTokens()) {
        System.out.println(st.nextToken());
    }
```

Your implementation should look something like this with the split method.

```
public static void main(String[] args){
    String[] tokens = "This is a test Ackeiouuu bam AAAAAAaaaaA AAAA".split("\\s");
    String mostVowels = null;
    int vowelCount = 0;

    for(int i = 0; i < tokens.length; i++){
        int temp = tokens[i].replaceAll("[^aeiouAEIOU]", "").length(); //regex magic
        if(temp > vowelCount){
            vowelCount = temp;
            mostVowels = tokens[i];
        }
    }
}
```

```
        System.out.println("Most vowels: " + mostVowels);  
    }  
}
```

This is a very minimalistic implementation that should suit your needs.
Thank you!

I should have added though: This is for an exercise at school, the teacher wants us to use `StringTokenizer` (although I know it's legacy). I'm not sure how to do it without using `split`.

EDIT: So I can help myself learn, what does this line do?

Code `int temp = tokens[i].replaceAll("[^aeiouAEIOU]", "").length();` //regex magic
It replaces the letters that aren't equal to vowels? I'll look this up. :P