

# Rajalakshmi Engineering College

Name: giri prasad

Email: 240701147@rajalakshmi.edu.in

Roll no:

Phone: 9150391064

Branch: REC

Department: CSE - Section 10

Batch: 2028

Degree: B.E - CSE

Scan to verify results



## 2024\_28\_III\_OOPS Using Java Lab

### **REC\_2028\_OOPS using Java\_Week 4\_CY**

Attempt : 1

Total Mark : 40

Marks Obtained : 40

#### **Section 1 : Coding**

##### **1. Problem Statement**

Meera is practicing her English vocabulary. She wants to focus on words that have more vowels in them, as they help improve her pronunciation. She decides to extract only those words from a sentence that contain at least two vowels.

Your task is to help Meera by writing a program that finds such words from the given sentence.

##### ***Input Format***

The input contains a string representing the sentence.

##### ***Output Format***

The output prints all the words that contain at least two vowels, separated by a space.

If no such word exists, print "No words with two vowels".

Refer to the sample output for formatting specifications.

### ***Sample Test Case***

Input: This is an example sentence

Output: example sentence

### ***Answer***

```
import java.util.Scanner;
```

```
class VowelWords {  
    public static void main(String[] args) {  
        Scanner sc = new Scanner(System.in);  
        String sentence = sc.nextLine();  
        String[] words = sentence.split(" ");  
        String vowels = "aeiouAEIOU";  
        String result = "";  
        for (int i = 0; i < words.length; i++) {  
            int count = 0;  
            for (int j = 0; j < words[i].length(); j++) {  
                if (vowels.indexOf(words[i].charAt(j)) != -1) {  
                    count++;  
                }  
            }  
            if (count >= 2) {  
                if (result.length() > 0) result += " ";  
                result += words[i];  
            }  
        }  
        if (result.length() == 0) System.out.println("No words with two vowels");  
        else System.out.println(result);  
    }  
}
```

**Status : Correct**

**Marks : 10/10**

## 2. Problem Statement

Anjali is preparing a report on text complexity. She wants to identify all words in a sentence that contain at least one digit so she can analyze numeric mentions.

Your task is to write a program that extracts and prints all words containing at least one digit from a given sentence.

If no such word exists, print "No words with digits found".

### ***Input Format***

The input contains a single line containing a sentence with multiple words.

### ***Output Format***

The output prints all words containing at least one digit separated by a space.

If no word contains a digit, print "No words with digits found".

Refer to the sample output for formatting specifications.

### ***Sample Test Case***

Input: The model X100 and Y200 are available

Output: X100 Y200

### ***Answer***

```
import java.util.Scanner;
class WordsWithDigits {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        String sentence = sc.nextLine();
        String[] words = sentence.split(" ");
        String result = "";
        for (String word : words) {
            if (containsDigit(word)) {
                result += word + " ";
            }
        }
    }
}
```

```

    }
    if (result.equals("")) {
        System.out.println("No words with digits found");
    } else {
        System.out.println(result.trim());
    }
}

private static boolean containsDigit(String word) {
    for (int i = 0; i < word.length(); i++) {
        if (Character.isDigit(word.charAt(i))) return true;
    }
    return false;
}
}

```

**Status :** Correct

**Marks :** 10/10

### 3. Problem Statement

A bookstore wants to analyze the titles of books to determine their longest word in each title. This helps in designing banners and covers.

Your task is to write a program that, given a sentence (book title), finds and prints the longest word. If multiple words have the same maximum length, print the first one.

#### ***Input Format***

The input contains a single line containing a sentence representing the book title.

#### ***Output Format***

The output prints a string representing the longest word in the sentence (book title).

Refer to the sample output for formatting specifications.

#### ***Sample Test Case***

**Input:** The Chronicles of Narnia  
**Output:** Chronicles

**Answer**

```
import java.util.Scanner;
class LongestWord {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        String sentence = sc.nextLine();
        String[] words = sentence.split(" ");
        String longest = "";
        for (String word : words) {
            if (word.length() > longest.length()) {
                longest = word;
            }
        }
        System.out.println(longest);
    }
}
```

**Status :** Correct

**Marks :** 10/10

#### 4. Problem Statement

Riya is preparing for a vocabulary test. Her teacher told her to focus on long words in her practice sentences, specifically words that have at least 5 letters.

Riya wants to write a program that will help her identify such words quickly.

Your task is to help Riya by printing all the words in a given sentence that have a length greater than or equal to 5.

If no such word exists, display "No long words found".

***Input Format***

The input contains a single line containing a sentence with multiple words.

***Output Format***

The output prints all words having length  $\geq 5$ , separated by a space.

If no such word is found, print "No long words found".

Refer to the sample output for formatting specifications.

### ***Sample Test Case***

Input: The quick brown fox jumps over the lazy dog

Output: quick brown jumps

### ***Answer***

```
import java.util.*;
```

```
class Main {  
    public static void main(String[] args) {  
        Scanner sc = new Scanner(System.in);  
        String sentence = sc.nextLine();  
        String[] words = sentence.split(" ");  
        boolean found = false;  
        for (String word : words) {  
            if (word.length() >= 5) {  
                System.out.print(word + " ");  
                found = true;  
            }  
        }  
        if (!found) {  
            System.out.print("No long words found");  
        }  
    }  
}
```

***Status : Correct***

***Marks : 10/10***