

# Rajalakshmi Engineering College

Name: Shivam Jaiswal

Email: 240701499@rajalakshmi.edu.in

Roll no: 240701499

Phone: 7318545479

Branch: REC

Department: CSE - Section 6

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**

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***

```
// You are using Java
import java.util.Scanner;

class LongWordsFinder {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        String sentence = scanner.nextLine();
        String[] words = sentence.split(" ");
        boolean foundLongWord = false;

        for (String word : words) {
            if (word.length() >= 5) {
                System.out.print(word + " ");
                foundLongWord = true;
            }
        }

        if (!foundLongWord) {
            System.out.println("No long words found");
        } else {
            System.out.println();
        }

        scanner.close();
    }
}
```

## 2. Problem Statement

Neha is analyzing text messages to identify words that have repeated characters. A word is considered “repetitive” if any character appears more than once in that word.

Your task is to write a program that extracts all words that contain repeated characters from a given sentence.

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

### ***Input Format***

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

### ***Output Format***

The output prints all words that contain repeated characters separated by a space.

If no word contains repeated characters, print "No repetitive words found".

Refer to the sample output for formatting specifications.

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

Input: letter balloon apple tree

Output: letter balloon apple tree

### ***Answer***

```
// You are using Java
import java.util.Scanner;
import java.util.HashSet;

class RepetitiveWordsFinder {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
```

```

String sentence = scanner.nextLine();
String[] words = sentence.split(" ");
boolean foundRepetitive = false;

for (String word : words) {
    if (hasRepeatedChar(word)) {
        System.out.print(word + " ");
        foundRepetitive = true;
    }
}

if (!foundRepetitive) {
    System.out.println("No repetitive words found");
} else {
    System.out.println();
}

scanner.close();
}

private static boolean hasRepeatedChar(String word) {
    HashSet<Character> chars = new HashSet<>();
    for (char c : word.toCharArray()) {
        if (chars.contains(c)) {
            return true;
        }
        chars.add(c);
    }
    return false;
}

```

**Status :** Correct

**Marks :** 10/10

### 3. Problem Statement

In a college, students are required to create unique usernames for accessing the digital library.

The librarian needs your help to verify whether the usernames entered by students are valid.

A username is considered valid if:

It contains only letters (a–z, A–Z) and digits (0–9). Its length is between 5 and 15 characters (inclusive). It must start with a letter (not a digit).

Your task is to determine whether each username in the list is valid or not.

### ***Input Format***

The first line of input contains an integer T, representing the number of usernames to check.

The next T lines each contain a string S, representing a username.

### ***Output Format***

For each username S, the output print "YES" if it is valid.

Otherwise, the output print "NO".

Refer to the sample output for formatting specifications.

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

Input: 1

Alice123

Output: YES

### ***Answer***

```
// You are using Java
import java.util.Scanner;

class UsernameValidator {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        int T = Integer.parseInt(scanner.nextLine());

        for (int i = 0; i < T; i++) {
            String username = scanner.nextLine();
            if (isValid(username)) {
```

```
        System.out.println("YES");
    } else {
        System.out.println("NO");
    }
}

scanner.close();
}

private static boolean isValid(String username) {
    if (username.length() < 5 || username.length() > 15) {
        return false;
    }

    char firstChar = username.charAt(0);
    if (!Character.isLetter(firstChar)) {
        return false;
    }

    for (char ch : username.toCharArray()) {
        if (!Character.isLetterOrDigit(ch)) {
            return false;
        }
    }

    return true;
}
}
```

**Status : Correct**

**Marks : 10/10**

#### 4. Problem Statement

In a university library, librarians need to track the usage of special characters in students' notes.

To help them, you are asked to write a program that counts the number of specific symbols in each passage of text.

The symbols of interest are:

Exclamation marks (!) Colons (:) Semicolons (;

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

The first line of input contains an integer  $T$ , representing the number of test cases (passages).

Each of the next  $T$  lines contains a single passage of text.

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

For each test case, print three integers separated by spaces, representing the number of exclamation marks, colons, and semicolons in the passage.

The first line of output corresponds to the first passage, the second line to the second passage, and so on.

Refer to the sample output for formatting specifications.

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

Input: 1

Hello! How are you

Output: 1 0 0

#### ***Answer***

```
// You are using Java
import java.util.Scanner;

class SymbolCounter {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        int T = Integer.parseInt(scanner.nextLine());

        for (int i = 0; i < T; i++) {
            String passage = scanner.nextLine();
            int exclamation = 0, colon = 0, semicolon = 0;

            for (char ch : passage.toCharArray()) {
                if (ch == '!') exclamation++;
                else if (ch == ':') colon++;
                else if (ch == ';') semicolon++;
            }

            System.out.println(exclamation + " " + colon + " " + semicolon);
        }
    }
}
```

```
        else if (ch == ':') semicolons++;
    }
    System.out.println(exclamations + " " + colons + " " + semicolons);
}
scanner.close();
}
```

**Status :** Correct

**Marks :** 10/10