UNIVERSITY
OF THE PEOPLE

CS-1103 Programming 2

Sana Ur Rehman

Programming Assignment Unit 1

2/1/25

```
package assignment_1;

import java.util.Scanner;

import java.util.HashMap;


public class TextAnalysisTool {

    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);


        // User Input

        System.out.println("Please enter a paragraph or a lengthy text:");

        String inputText = scanner.nextLine();


        // Character Count

        int charCount = inputText.length();

        System.out.println("Total number of characters: " + charCount);


        // Word Count

        String[] words = inputText.split("\\s+");
```

```
int wordCount = words.length;

System.out.println("Total number of words: " + wordCount);



// Most Common Character

HashMap<Character, Integer> charFrequency = new HashMap<>();

for (char c : inputText.toLowerCase().toCharArray()) {

    charFrequency.put(c, charFrequency.getOrDefault(c, 0) + 1);

}

char mostCommonChar = ' ';

int maxCount = 0;

for (char c : charFrequency.keySet()) {

    if (charFrequency.get(c) > maxCount) {

        maxCount = charFrequency.get(c);

        mostCommonChar = c;

    }

}

System.out.println("Most common character: " + mostCommonChar);
```

```
// Character Frequency

System.out.println("Please enter a character to find its frequency:");

char charInput = scanner.next().toLowerCase().charAt(0);

int charFrequencyCount = charFrequency.getOrDefault(charInput, 0);

System.out.println("Frequency of '" + charInput + "': " + charFrequencyCount);



// Word Frequency

System.out.println("Please enter a word to find its frequency:");

String wordInput = scanner.next().toLowerCase();

int wordFrequencyCount = 0;

for (String word : words) {

    if (word.toLowerCase().equals(wordInput)) {

        wordFrequencyCount++;

    }

}

System.out.println("Frequency of \"" + wordInput + "\": " + wordFrequencyCount);



// Unique Words
```

```
        HashMap<String, Integer> uniqueWords = new HashMap<>();

        for (String word : words) {

            uniqueWords.put(word.toLowerCase(), uniqueWords.getOrDefault(word.toLowerCase(),
0) + 1);

        }

        System.out.println("Number of unique words: " + uniqueWords.size());



        scanner.close();

    }

}
```

nt_1/TextAnalysisTool.java - Eclipse IDE

Search Project Run Window Help

TextAnalysisTool.java

```java
package assignment_1;
import java.util.Scanner;
import java.util.HashMap;

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

        // User Input
        System.out.println("Please enter a paragraph or a lengthy text:");
        String inputText = scanner.nextLine();

        // Character Count
        int charCount = inputText.length();
        System.out.println("Total number of characters: " + charCount);

        // Word Count
        String[] words = inputText.split("\\s+");
        int wordCount = words.length;
        System.out.println("Total number of words: " + wordCount);

        // Most Common Character
        HashMap<Character, Integer> charFrequency = new HashMap<>();
        for (char c : inputText.toLowerCase().toCharArray()) {
            charFrequency.put(c, charFrequency.getOrDefault(c, 0) + 1);
        }
        char mostCommonChar = ' ';
        int maxCount = 0;
        for (char c : charFrequency.keySet()) {
            if (charFrequency.get(c) > maxCount) {
                maxCount = charFrequency.get(c);
                mostCommonChar = c;
            }
        }
        System.out.println("Most common character: " + mostCommonChar);

        // Character Frequency
        System.out.println("Please enter a character to find its frequency:");
        char charInput = scanner.next().toLowerCase().charAt(0);
        int charFrequencyCount = charFrequency.getOrDefault(charInput, 0);
        System.out.println("Frequency of '" + charInput + "': " + charFrequencyCount);

```

Problems Javadoc Declaration Console

<terminated> TextAnalysisTool [Java Application] C:\Users\Yahya\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_23.0.2.v20250131-0604\jre\bin\javaw.exe (Feb 1, 2025, 11:34:56 a.m. – 11:35:21 a.m. elapsed: 0:00:25.859) [pid: 8408]

```
Please enter a paragraph or a lengthy text:
Custom exceptions enhance code readability and maintenance. Creating specific exceptions for different scenarios helps other developers understand and handle errors appropriately. For example, creating an
Writing effective exception handling code requires balance. Over-catching exceptions can hide important problems, while under-catching might leave your program vulnerable to crashes. The key lies in ident
Total number of words: 39
Most common character: e
Please enter a character to find its frequency:

Frequency of 'w': 2
Please enter a word to find its frequency:
Frequency of "effective": 0
Number of unique words: 35
```

Writable Smart Insert 15 : 72 : 504