# LAPORAN PRAKTIKUM
# TEKNIK PEMROGRAMAN

**DEFENSIVE PROGRAMMING**

MINGGU KE-6


NAMA: FAUZI ISMAIL

NIM: 241524042

KELAS: D4-1B

**PROGRAM STUDI SARJANA TERAPAN**

**TEKNIK INFORMATIKA**

**POLITEKNIK NEGERI BANDUNG**

**2025**

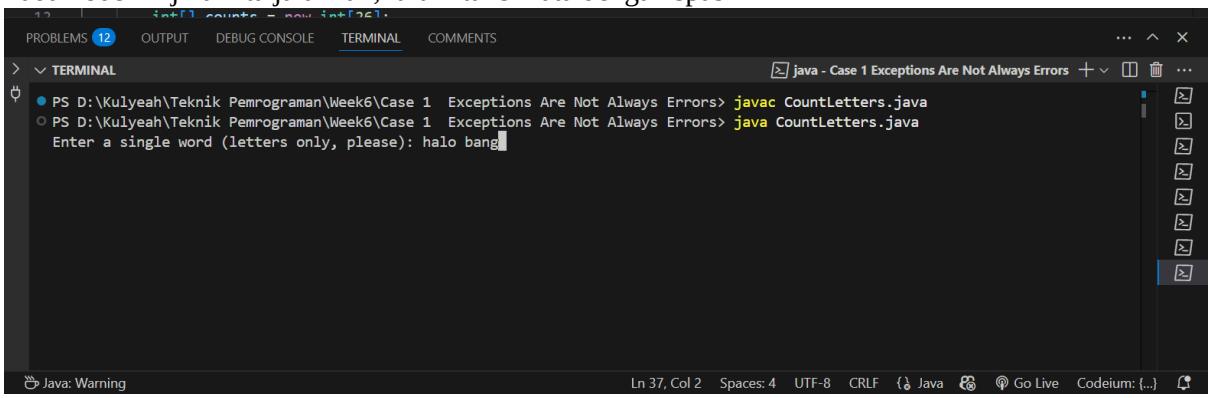
# DAFTAR ISI

https://github.com/mailvlous/teknikPemrograman/tree/main/Week6

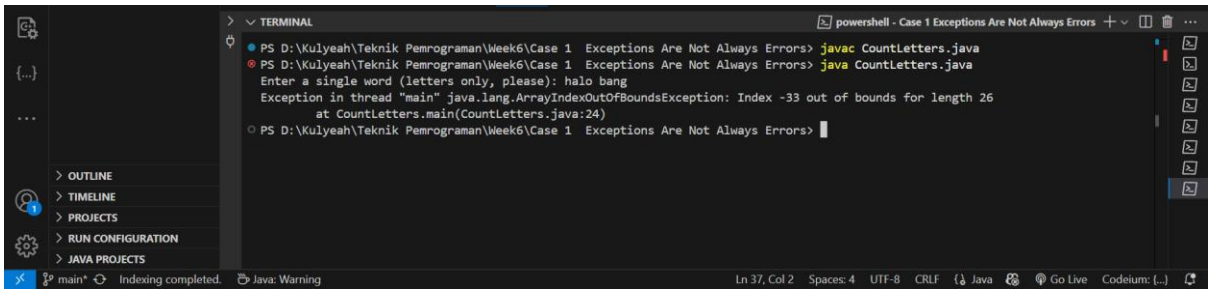
## 1. Case 1 : Exceptions Aren't Always Errors

```java
// ******************************************************************
// CountLetters.java
//
// Reads a words from the standard input and prints the number of
// occurrences of each letter in that word.
//
// ******************************************************************
import java.util.Scanner;

public class CountLetters {
    public static void main(String[] args) {
        int[] counts = new int[26];
        Scanner scan = new Scanner(System.in);

        // Get word from user
        System.out.print("Enter a single word (letters only, please): ");
        String word = scan.nextLine();

        // Convert to all upper case
        word = word.toUpperCase();

        // Count frequency of each letter in string
        for (int i = 0; i < word.length(); i++) {
            counts[word.charAt(i) - 'A']++;
        }

        // Print frequencies
        System.out.println();
        for (int i = 0; i < counts.length; i++) {
            if (counts[i] != 0) {
                System.out.println((char) (i + 'A') + ": " + counts[i]);
            }
        }

        scan.close();
    }
}
```

Pada kode ini jika kita jalankan, lalu kita isi kata dengan spasi




Maka hasilnya akan terjadi exception




Exception ini terjadi di bagian

```
23        for (int i = 0; i < word.length(); i++) {
24            counts[word.charAt(i) - 'A']++;
25        }
```

Program akan mengecek setiap huruf yang kemudian dikonversi dalam ASCII kode lalu setelah itu dikurangi oleh ASCII kode dari ‘A’ yaitu 65. Program tak akan mengalami exception jika yang diinput adalah huruf, itu karena dari A = 65 hingga Z = 90 jika dikurangi 65 maka akan menghasilkan lebih dari sama dengan 0. Program akan terjadi exception jika yang diinputkan adalah karakter yang jika diubah kedalam kode ASCII dibawah 65, yang mana jika misal SPACE dengan kode ASCII 32 diinputkan maka 32-65 = -33. Program akan mengalami exception jika hasil operasi negative.

Solusi:
Mari kita buat try catch exception seperti ini

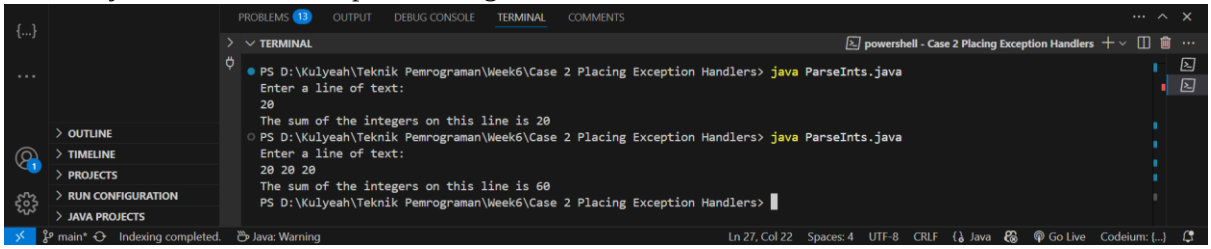
```java
// ****************************************************************
// CountLetters.java
//
// Reads a words from the standard input and prints the number of
// occurrences of each letter in that word.
//
// ****************************************************************
import java.util.Scanner;

public class CountLetters {
    public static void main(String[] args) {
        int[] counts = new int[26];
        Scanner scan = new Scanner(System.in);

        try {
                    // Get word from user
        System.out.print("Enter a single word (letters only, please): ");
        String word = scan.nextLine();

        // Convert to all upper case
        word = word.toUpperCase();

        // Count frequency of each letter in string
        for (int i = 0; i < word.length(); i++) {
            counts[word.charAt(i) - 'A']++;
        }

        // Print frequencies
        System.out.println();
        for (int i = 0; i < counts.length; i++) {
            if (counts[i] != 0) {
                System.out.println((char) (i + 'A') + ": " + counts[i]);
            }
        }
        } catch (ArrayIndexOutOfBoundsException e) {
            System.out.println("Tolong gunakan huruf A-Z");
        }



        scan.close();
    }
}
```

Kita akan buat exception yang jika terkena ArrayIndexOutOfBounds maka program akan memberi perintah untuk “Tolong gunakan huruf A-Z”.

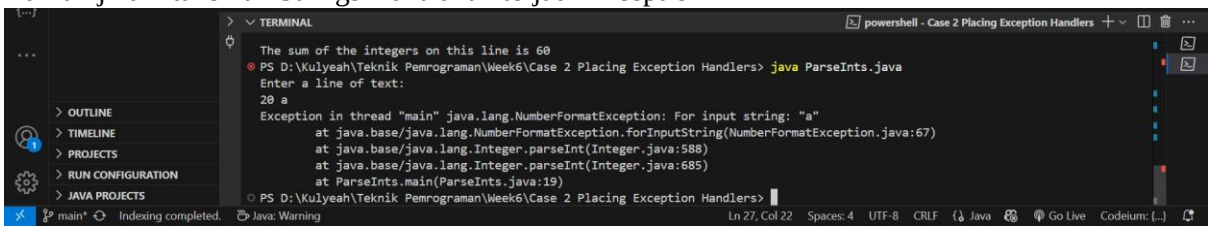
## 2. Case 2 : Placing Exception Handlers

```java
// ************************************************************
// ParseInts.java
//
// Reads a line of text and prints the integers in the line.
//
// ************************************************************
import java.util.Scanner;

public class ParseInts {
    public static void main(String[] args) {
        int val, sum = 0;
        Scanner scan = new Scanner(System.in);

        // Prompt user for input
        System.out.println("Enter a line of text:");
        Scanner scanLine = new Scanner(scan.nextLine());

        while (scanLine.hasNext()) {
            val = Integer.parseInt(scanLine.next());
            sum += val;
        }

        // Print the sum of integers
        System.out.println("The sum of the integers on this line is " + sum);

        // Close scanners
        scan.close();
        scanLine.close();
    }
}
```

Jika kita jalankan dan coba inputkan integer semua akan normal



Namun jika kita isikan Strings maka akan terjadi Exception

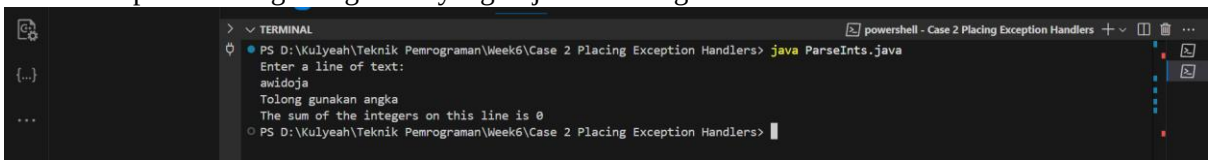



Itu terjadi karena kita menginputkan format yang tidak sesuai dengan tipe data variable val, sum, maka akan terjadi NumberFormatException

Solusi:
Mari kita buat try catch Exceptionnya

```java
// ************************************************************
// ParseInts.java
//
// Reads a line of text and prints the integers in the line.
//
// ************************************************************
import java.util.Scanner;

public class ParseInts {
    public static void main(String[] args) {
        int val, sum = 0;
        Scanner scan = new Scanner(System.in);

        // Prompt user for input
        System.out.println("Enter a line of text:");
        String line = scan.nextLine();


        Scanner scanLine = new Scanner(line);

        while (scanLine.hasNext()) {
            try {
                val = Integer.parseInt(scanLine.next());
                sum += val;
            } catch (NumberFormatException e) {
                System.out.println("Tolong gunakan angka");
            }
        }

        // Print the sum of integers
        System.out.println("The sum of the integers on this line is " + sum);

        // Close scanners
        scan.close();
        scanLine.close();
    }
}
```

Jika kita inputkan ulang string maka yang terjadi sekarang adalah



# 3. Case 3 : Throwing Exceptions

Pada file Factorials.java

```java
import java.util.Scanner;

Codeium: Refactor | Explain
public class Factorials {
Run main | Debug main | Run | Debug | Codeium: Refactor | Explain | Generate Javadoc | ×
public static void main(String[] args) {
    String keepGoing = "y";
    Scanner scan = new Scanner(System.in);

    while (keepGoing.equals(anObject:"y") || keepGoing.equals(anObject:"Y")) {
        System.out.print(s:"Enter an integer: ");
        int val = scan.nextInt();
        System.out.println("Factorial(" + val + ") = " + MathUtils.factorial(val));
        System.out.print(s:"Another factorial? (y/n) ");
        keepGoing = scan.next();
    }
}
}

```

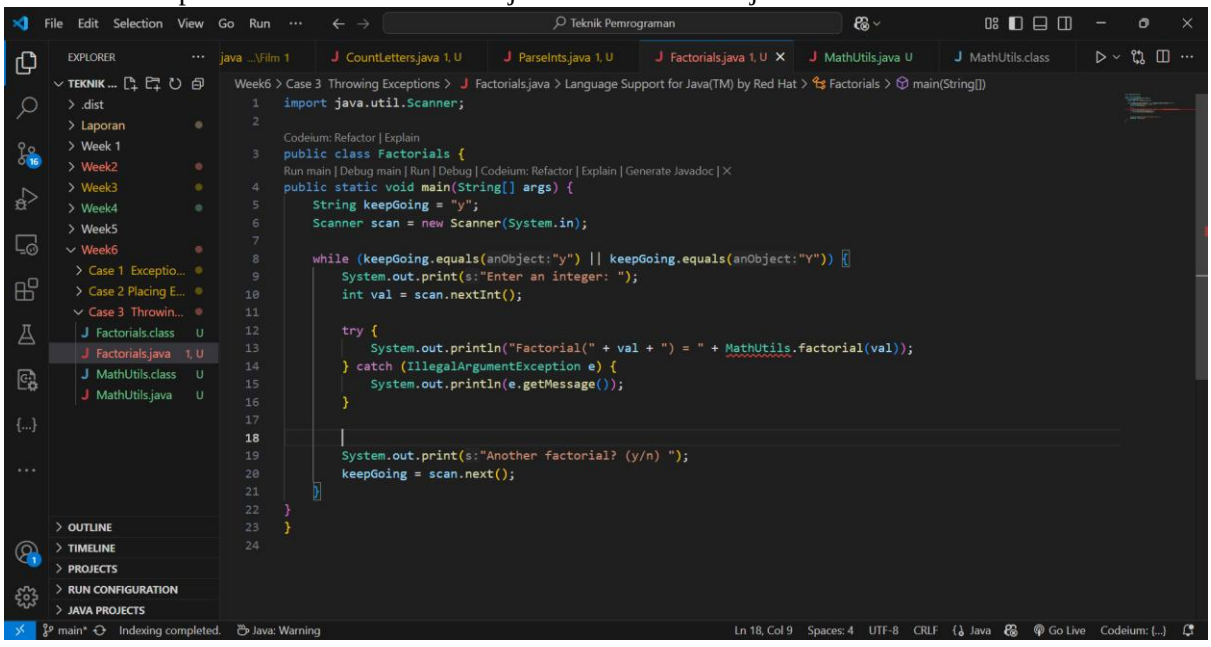
Pada file MathUtils.java

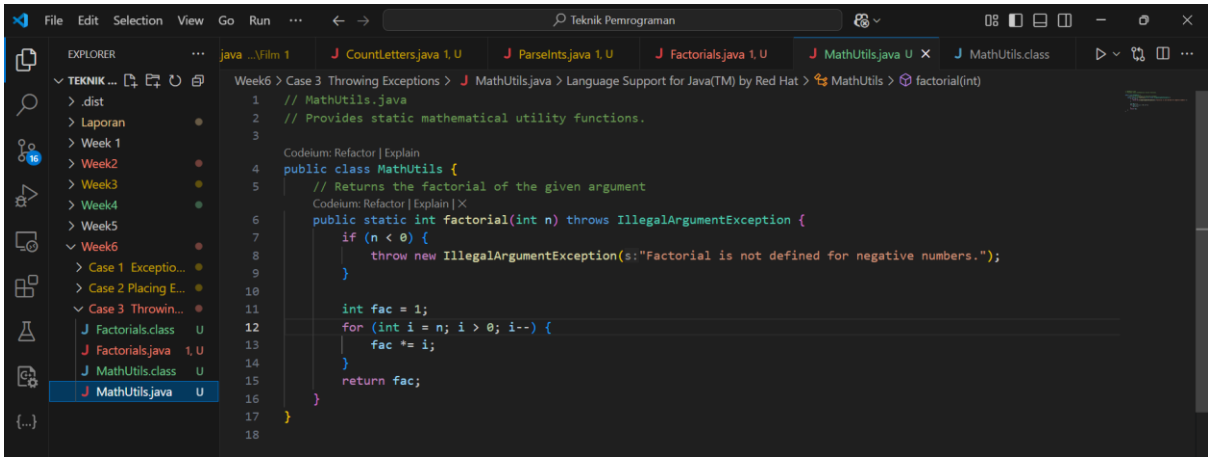
```java
Codeium: Refactor | Explain
public class MathUtils {
    // Returns the factorial of the given argument
    Codeium: Refactor | Explain | ×
    public static int factorial(int n) {
        int fac = 1;
        for (int i = n; i > 0; i--) {
            fac *= i;
        }
        return fac;
    }
}

```

Dalam soal dikatakan bahwa jika kita menginputkan angka negative maka akan selalu menghasilkan 1. Untuk itu mari kita modifikasi agar throws an Exception.

Solusi:

Kita buat Exception baru di file MathUtils.java serta Factorials.java






Sekarang jika kita menginputkan angka negative maka akan menghasilkan Excerption: