

TUGAS : EXCEPTION
MINGGU PERTEMUAN 9
PEMROGRAMAN BERORIENTASI OBJEK (PR)



Disusun oleh :
Muhammad Wildan Gumilang (231511087)

PROGRAM DIPLOMA III TEKNIK INFORMATIKA
JURUSAN TEKNIK KOMPUTER DAN INFORMATIKA
POLITEKNIK NEGERI BANDUNG

DAFTAR ISI

DAFTAR ISI	2
Studi Kasus 1.....	3
Studi Kasus 2.....	4
Studi Kasus 3.....	5
Link Repository GitHub.....	7

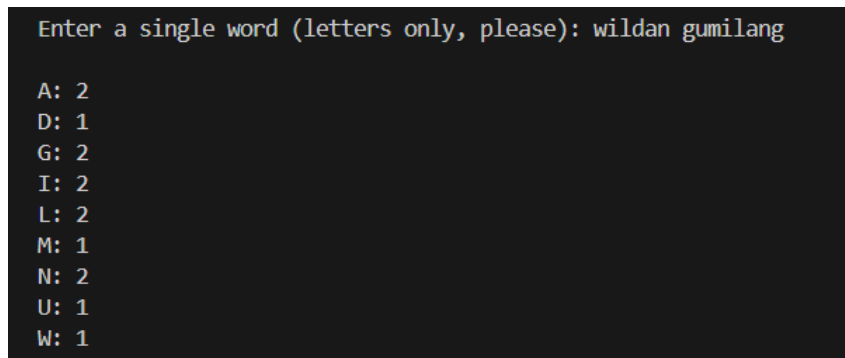
Studi Kasus 1

CountLetters.java

```
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 the string
        for (int i = 0; i < word.length(); i++) {
            try {
                counts[word.charAt(i) - 'A']++;
            } catch (ArrayIndexOutOfBoundsException e) {
                System.out.println("Not a letter: " + word.charAt(i));
            }
        }
        // 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]);
            }
        }
    }
}
```

Output :

- Menambahkan try catch :



```
Enter a single word (letters only, please): wildan gumilang
A: 2
D: 1
G: 2
I: 2
L: 2
M: 1
N: 2
U: 1
W: 1
```

- Menambahkan print dan menampilkan karakter yang selain masuk ke pengecualian pada catch :

```

11 (!?) { java CountLetters }
Enter a single word (letters only, please): wildan1gumilang2
Not a letter: 1
Not a letter: 2

A: 2
D: 1
G: 2
I: 2
L: 2
M: 1
N: 2
U: 1
W: 1
PS. D:\wg\Kuliah\semester_3\PRO\praktek\pertemuan9\src>

```

Komentar :

Pada program in, modifikasi yang dilakukan adalah menambahkan try catch di dalam loop pertama (for) yang menghitung frekuensi huruf. Modifikasi ini dilakukan untuk menangkap pengecualian dari `ArrayIndexOutOfBoundsException` yang bisa terjadi jika ada input non huruf dari pengguna. Dengan menambahkan error handling ini, program tidak akan langsung berhenti saat ada karakter non huruf. Didalam try catch tersebut juga diambahkan pesan yang dapat menampilkan karakter yang non huruf.

Studi Kasus 2

ParseInt.java

```

import java.util.Scanner;
public class ParseInts {
    public static void main(String[] args) {
        int val, sum = 0;
        Scanner scan = new Scanner(System.in);
        String line;
        // Get a line of text from the user
        System.out.println("Enter a line of text");
        Scanner scanLine = new Scanner(scan.nextLine());
        // Parse and sum integers from the line
        while (scanLine.hasNext()) {
            try {
                val = Integer.parseInt(scanLine.next());
                sum += val;
            } catch (NumberFormatException e) {

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

```

Output :

- Try diluar loop

```
(#?) { java -jar scines }  
Enter a line of text  
10 dog and 1 fish  
The sum of the integers on this line is 10  
PS D:\wg\Kuliah\semester_3\PPG\praktek\pertemuan
```

- Try catch didalam loop

<pre>(#?) { java -jar scines } Enter a line of text We have 2 dogs and 1 cat. The sum of the integers on this line is 3 PS D:\wg\Kuliah\semester_3\PPG\praktek\pertemuan</pre>	<pre>(#?) { java -jar scines } Enter a line of text 10 dog and 1 fish The sum of the integers on this line is 11 PS D:\wg\Kuliah\semester_3\PPG\praktek\pertemuan</pre>
--	---

Komentar :

Pada program ini, beberapa modifikasi yang dilakukan adalah :

1. Menambahkan try catch di luar loop while, sehingga program akan berhenti padasaat terdapat kesalahan pada input dari user yang bukan angka. Oleh karena itu program hanya akan menghitung angka sampai kesalahan pertama ditemukan dan tidak akan menghitung setelah kesalahan tersebut.
2. Menambahlan try catch di dalam loop while, sehingga program menangkap pengecualian NumberFormatException setiap ditemukan kesalahan. Jadi dalam program ini dapat lanjut ke iterasi selanjutnya dan menghitung semua yang ada di dalam input, sehingga menghitung jumlah dari semua angka yang ada.

Studi Kasus 3

Factorials.java

```
import java.util.Scanner;  
  
public class Factorials {  
    public static void main(String[] args) {  
        String keepGoing = "y";  
        Scanner scan = new Scanner(System.in);  
  
        // Loop until the user decides to stop  
        while (keepGoing.equals("y") || keepGoing.equals("Y")) {  
            System.out.print("Enter an integer: ");  
            int val = scan.nextInt();  
  
            try {  
                System.out.println("Factorial(" + val + ") = " + MathUtils.factorial(val));  
            } catch (IllegalArgumentException e) {  
                System.out.println("Error: " + e.getMessage());  
            }  
        }  
    }  
}
```

```

        // Ask if the user wants to continue
        System.out.print("Another factorial? (y/n) ");
        keepGoing = scan.next();
    }
}
}

```

MathUtils.java

```

public class MathUtils {
    public static int factorial(int n) throws IllegalArgumentException {
        if (n < 0) {
            throw new IllegalArgumentException("Faktorial tidak didefinisikan untuk
bilangan bulat negatif.");
        }
        if (n > 16) {
            throw new IllegalArgumentException("Faktorial terlalu besar untuk bilangan
bulat (overflow untuk n > 16)");
        }
        int fac = 1;

        // Calculate factorial
        for (int i = n; i > 0; i--) {
            fac *= i;
        }

        return fac;
    }
}

```

Output :

```

Another factorial? (y/n) y
Enter an integer: -5
Factorial(-5) = 1
Another factorial? (y/n) y
Enter an integer: 17
Factorial(17) = -288522240
Another factorial? (y/n) n
PS D:\wq\Kuliah\semester 3\PRO\pr
1 ($?) { java Factorials }
Enter an integer: -5
Error: Faktorial tidak didefinisikan untuk bilangan bulat negatif.
Another factorial? (y/n) y
Enter an integer: 17
Error: Faktorial terlalu besar untuk bilangan bulat (overflow untuk n > 16)
Another factorial? (y/n)

```

Komentar :

Pada program Factorials.java dan MathUtils.java, dilakukan modifikasi pada metode factorial, ditambahkan pengecekan untuk parameter n, yang dimana jika n negatif atau lebih besar dari 16, program akan melakukan throws IllegalArgumentException. Pada metode main,

try catch ditambahkan saat memanggil factorial, sehingga jika terjadi exception pesan error akan muncul dan program akan tetap berjalan.

Link Repository GitHub

<https://github.com/WildanGumilang/PBO-praktek.git>