Praktikum Pemrograman I
Laporan Pertemuan 10

Nama : Muhammad Faris Fathur Rohman

NPM : 2203040126

Kelas : D

Link Github : https://github.com/Riss27/PP1_223040126_Pertemuan10.git

1. Struktur Stack

```java
public class StrukturStack {
    private int[] array;
    private int capacity;
    private int TOP;
    private int temp;
    public final int MIN = -1;

    public StrukturStack(int capacity) { // Constructor untuk menginisialisasi capacity dan array
        super();
        array = new int[capacity];
        this.capacity = capacity;
        TOP = MIN;
    }

    public void push(int data) { // Method push untuk menambahkan data ke dalam stack
        if (isFull()) {
            System.out.println(x:"Stack penuh");
        } else {
            TOP++;
            array[TOP] = data;
        }
    }
}
```

```java
    public boolean isEmpty() { // Method untuk mengecek apakah stack kosong
    atau tidak
        return (TOP == MIN);
    }

    public boolean isFull() { // Method untuk mengecek apakah stack penuh
    atau tidak
        return TOP == capacity - 1;
    }

    public void displayElements() { // Method untuk menampilkan elemen ke
    layar dari TOP
        System.out.println(x:"Elemen from TOP:");
        if (isEmpty()) {
            System.out.println(x:"Stack kosong");
        } else {
            for (int i = TOP; i >= 0; i--) {
                System.out.print(array[i] + " ");
            }
        }
    }
```

```java
    public int pop() { // Method pop untuk mengeluarkan data dari stack
        if (isEmpty()) {
            System.out.println(x:"Stack kosong");
            return MIN; // Mengembalikan nilai MIN untuk menandakan stack
            kosong
        } else {
            temp = array[TOP]; // Menyimpan nilai TOP ke dalam variabel temp
            TOP--; // Mengurangi TOP
        }
        return temp;
    }


    public int top() { // Method untuk mengecek apakah stack kosong atau tidak
        if (isEmpty()) {
            return MIN;
        } else {
            return array[TOP];
        }
    }
```

```java
    public int size() {  // Method untuk mengecek ukuran stack
        return TOP + 1;
    }
}
```

2. Stack Main

```java
public class StackMain {

    Run | Debug
    public static void main(String[] args) {
        StrukturStack stack = new StrukturStack(capacity:3);

        // Sebelum push
        System.out.println(x:"Sebelum push");
        System.out.println("Size: " + stack.size());
        System.out.println("Empty: " + stack.isEmpty());
        System.out.println("Full: " + stack.isFull());
        System.out.println("Top: " + stack.top());
        stack.displayElements();
        System.out.println();
```

```java
        // Melakukan push 3x
        System.out.println(x:"Melakukan push 3x:");
        stack.push(data:2);
        stack.push(data:4);
        stack.push(data:1);
        System.out.println("Size: " + stack.size());
        System.out.println("Empty: " + stack.isEmpty());
        System.out.println("Full: " + stack.isFull());
        System.out.println("Top: " + stack.top());
        stack.displayElements();
        System.out.println();
```

Output Latihan Push :

```
Sebelum push

Size: 0

Empty: true

Full: false

Top: -1

Elemen from TOP:

Stack kosong


Melakukan push 3x:

Size: 3

Empty: false

Full: true

Top: 1

Elemen from TOP:

1 4 2
```

```java
26              // Melakukan pop 1x
27              System.out.println(x:"\nMelakukan pop 1x:");
28              stack.pop();
29              System.out.println("Size: " + stack.size());
30              System.out.println("Empty: " + stack.isEmpty());
31              System.out.println("Full: " + stack.isFull());
32              System.out.println("Top: " + stack.top());
33              stack.displayElements();
34              System.out.println();
35          }
36      }
```

Output Latihan Pop :

```
Melakukan pop 1x:

Size: 2

Empty: false

Full: false

Top: 4

Elemen from TOP:

4 2
```