

Pertemuan 6

Penerapan Interface dan Exception

A. Interface

Interface adalah satu kontrak dalam wujud koleksi metoda dan deklarasi-deklarasi tetap. Bila suatu kelas menerapkan interface, maka hal tersebut untuk menerapkan semua metoda yang dideklarasikan didalam interface.

Interface sering digunakan sebagai pengganti kelas abstrak walaupun tidak ada implementasi secara default untuk menerima inheritance yaitu, tidak ada field-field dan tidak ada implementasi-implementasi metoda secara default. Seperti kelas-kelas abstrak publik, interface adalah jenis-jenis publik secara khusus, sehingga secara normal dideklarasikan di dalam file-file dengan sendirinya, yaitu dengan nama yang sama sebagai interface dan nama file java.

Interface pada java merupakan satu koleksi perilaku abstrak yang dapat diadopsi oleh kelas manapun tanpa menerima inheritance dari suatu superclass.

```
interface Trayek{
    public static final String PERGI = "Blok M";
    public static final String PULANG = "Ciledug";
    public abstract void tarikpergi();
    public abstract void tarikpulang();}
class Bus implements Trayek {
    private String statusBus = "Ciledug";
    public void tarikpergi(){
        if(this.statusBus.equals(PULANG)){
            this.statusBus = PERGI;
            System.out.println("Narik Ke Blok M");}
        else{
            System.out.println("Sudah Narik Ke Blok M");}
    }
    public void tarikpulang(){
        if(this.statusBus.equals(PERGI)){
            this.statusBus = PULANG;
            System.out.println("Narik Ke Ciledug");}
        else{
            System.out.println("Sudah Narik Ke Ciledug");}
    }
}
```

```

public class Inter {
public static void main(String[] args) {
Bus MetroMini = new Bus();
MetroMini.tarikpergi();
MetroMini.tarikpergi();
MetroMini.tarikpulang();
MetroMini.tarikpulang();}
}

```

B. Exception

Eksepsi adalah suatu indikasi masalah yang terjadi selama pelaksanaan program atau eksekusi program. Nama "Exception" menyiratkan bahwa masalah jarang terjadi jika “aturan” suatu statemen secara normal dilaksanakan dengan tepat, maka eksepsi digunakan untuk “mengatur” masalah yang terjadi. Penanganan eksepsi memberdayakan para programmer untuk membuat aplikasi-aplikasi yang dapat memecahkan atau menangani eksepsi-eksepsi. Dalam banyak kesempatan, menangani Eksepsi mengijinkan suatu program untuk melanjutkan jalannya program, seolah-olah tidak ada masalah.

```

import java.util.Scanner;
public class Excep {
    public static void main(String[] args) {
int a, b;double c=0;
Scanner input = new Scanner(System.in);
System.out.print("Nilai A = "); a = input.nextInt();
System.out.print("Nilai B = "); b = input.nextInt();
try{
c = a / b;}
catch (Exception e){
System.err.println("\nTidak Bisa Dibagi 0");}
System.out.println("\nNilai C = " + c);}
}

```