

Modul 5

Packaging

A) Pokok Bahasan

1. Package
2. Import
3. Access modifier Default

B) Tujuan

1. Mahasiswa Mampu Mengelola Class Dengan Package

C) Dasar Teori

1. Packaging

Packaging adalah konsep membungkus sekumpulan class ke dalam suatu wadah yang dinamakan package sehingga class-class tersebut menjadi terstruktur dan mudah diakses

Package tersebut juga harus disusun berdasarkan direktori/folder

Keyword `package` harus ditaruh paling atas sebelum deklarasi class

Syntax :
`package [nama_package];`

Contoh :
`package hello;
public class Hi {

}`

2. Import Class

Class yang sudah dikelompokkan berdasarkan package-nya dapat dipakai oleh class lain menggunakan keyword `import`

Untuk import semua class di dalam package tersebut, bisa menggunakan simbol asteriks (*) sebagai pengganti nama class

Syntax :
`import [nama_package].[nama_class];
import [nama_package].*;`

Contoh :
`import java.util.Scanner;`

```
import java.lang.*;
```

3. Import Static

Apabila kita hanya ingin menggunakan static attribute atau method dari suatu class, maka kita bisa menggunakan keyword `import static`

Syntax :

```
import static [nama_package].[nama_class].[atribut/method];
```

Contoh :

```
import static java.lang.System.out;
```

4. Access Modifier Default

Sebelumnya kita telah membahas access modifier public dan private. Tidak seperti access modifier lainnya, access modifier default tidak memiliki syntax khusus

Untuk menggunakannya, cukup deklarasikan class, attribute, atau method tanpa keyword public, private dan protected

Class, attribute, method yang menggunakan access modifier default hanya dapat diakses oleh class dengan package yang sama

Contoh :

```
// class dengan access modifier default
class Hello {
    // attribute dengan access modifier default
    int jumlahSapa = 0;

    // method dengan access modifier default
    void sapa() {
        this.jumlahSapa++;
    }
}
```

D) Praktik

Buatlah direktori dengan susunan sebagai berikut

```
/main
  Main.java
/bangun
  /datar
    PersegiPanjang.java
    Segitiga.java
  /ruang
    Kubus.java
    Balok.java
  Alarm.java
```

file : bangun/Alarm.java

```
package bangun;

public class Alarm {

    public Alarm() {
        System.out.println("Alarm Berhasil Dibuat");
    }

    public static String waktuBangun = "09:00";
    public static void bangunlah() {
        System.out.println("Bangunlah Saudaraku, ini sudah Waktunya");
    }
}
```

file : bangun/datar/Segitiga.java

```
package bangun.datar;

public class Segitiga {
    double alas;
    double tinggi;

    public Segitiga(double alas, double tinggi) {
        this.alas = alas;
        this.tinggi = tinggi;
    }

    public double hitungLuas() {
        return (this.alas * this.tinggi) / 2;
    }
}
```

file : bangun/ruang/Balok.java

```
package bangun.ruang;

public class Balok {
    double panjang;
    double lebar;
    double tinggi;

    public Balok(double panjang, double lebar, double tinggi) {
        this.panjang = panjang;
        this.lebar = lebar;
        this.tinggi = tinggi;
    }

    public double hitungVolume() {
```

```

        return this.panjang * this.lebar * this.tinggi;
    }
}

```

file : bangun/ruang/Kubus.java

```

package bangun.ruang;

public class Kubus {
    double sisi;

    public Kubus(double sisi) {
        this.sisi = sisi;
    }

    public double hitungVolume() {
        return Math.pow(this.sisi, 3);
    }
}

```

file : main/Main.java

```

package main;
import bangun.datar.Segitiga;
import bangun.ruang.*;
import static bangun.Alarm.waktuBangun;
import static bangun.Alarm.bangunlah;
import static java.lang.System.out;

public class Main {

    public static void main(String []args) {
        Segitiga s1 = new Segitiga(10, 5);
        // System.out.println( s1.alas ); // Error
        out.println( s1.hitungLuas() );

        Balok b1 = new Balok(2, 3, 5);
        out.println( b1.hitungVolume() );

        Kubus k1 = new Kubus(5);
        out.println( k1.hitungVolume() );

        out.println( waktuBangun );
        bangunlah();

        // Alarm a1 = new Alarm(); // Error : Alarm belum di Import
        bangun.Alarm a1 = new bangun.Alarm();
    }
}

```