

LAPORAN PRAKTIKUM

PEMROGRAMAN BERORIENTASI OBJEK

PERTEMUAN KE-5



Disusun Oleh :

NAMA : Raden Isnawan Argi Aryasatya

NIM : 195410257

JURUSAN : Teknik Informatika

JENJANG : S1

KELAS : TI-5

Laboratorium Terpadu
Sekolah Tinggi Manajemen Informatika Komputer
AKAKOM
YOGYAKARTA

2021

PERTEMUAN KE-5

(PACKAGE, ACCESS MODIFIER, STATIC)

TUJUAN

Dapat menggunakan paket java, dapat membuat dan menggunakan package yang dibuat, dan dapat mengatur hak akses member.

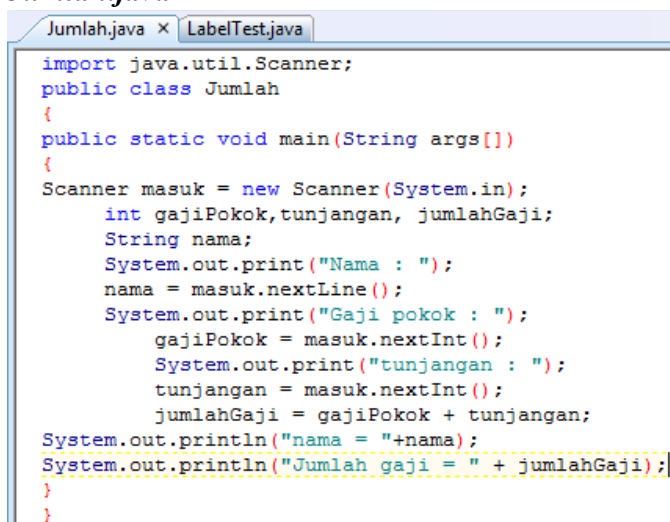
DASAR TEORI

Seperti yang diketahui, Java adalah pemrograman dengan menggunakan banyak kelas. Kelas-kelas tersebut dikelompokkan ke dalam kategori tertentu yang berhubungan disebut dengan packages (paket) . Misalnya saja paket javax.swing.* berarti semua kelas yang berhubungan dengan javax swing berada dalam direktori tersebut. Langkah-langkah untuk membuat kelas yang reusable adalah sebagai berikut:

1. Deklarasikan public class
2. Pilih nama paket, dan tambahkan sebuah package declaration ke file source code untuk deklarasi kelas reusable
3. Kompilasi kelas tersebut sehingga dia ditempatkan di dalam struktur direktori paket yang sesuai.
4. Import reusable class ke dalam sebuah program, dan gunakan kelas tersebut.

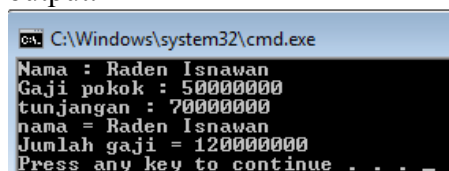
PRAKTIK 1: Menggunakan paket miliknya java.

Jumlah.java



```
import java.util.Scanner;
public class Jumlah
{
    public static void main(String args[])
    {
        Scanner masuk = new Scanner(System.in);
        int gajiPokok, tunjangan, jumlahGaji;
        String nama;
        System.out.print("Nama : ");
        nama = masuk.nextLine();
        System.out.print("Gaji pokok : ");
        gajiPokok = masuk.nextInt();
        System.out.print("tunjangan : ");
        tunjangan = masuk.nextInt();
        jumlahGaji = gajiPokok + tunjangan;
        System.out.println("nama = "+nama);
        System.out.println("Jumlah gaji = " + jumlahGaji);
    }
}
```

output:



```
ca. C:\Windows\system32\cmd.exe
Nama : Raden Isnawan
Gaji pokok : 50000000
tunjangan : 70000000
nama = Raden Isnawan
Jumlah gaji = 120000000
Press any key to continue . . .
```

penjelasan:

Scanner merupakan class yang menyediakan fungsi-fungsi untuk mengambil input dari keyboard. Agar kita bisa menggunakan Scanner, kita perlu mengimpornya ke dalam kode: **import java.util.Scanner;**. Java adalah paket top level/utama, util adalah paket sub, Scanner adalah class di dalam paket util. Setelah itu kita mendeklarasikan kelas dengan modifier public yaitu **public**

class Jumlah. Jadi intinya import ini berfungsi untuk mengimport library Scanner dari java.util. Kita menggunakan public supaya kelas/method/attribute tersebut dapat diakses oleh kelas lain dimanapun, yang menjadikan kelas itu bersifat universal. Lalu deklarasikan **public static void main(String args[]){** untuk mengawali eksekusi. Public, yang berarti metode ini bisa dipanggil dan digunakan didalam Class atau diluar Class. void, yang berarti bahwa metode ini tidak mengirimkan nilai balik. String[], adalah tipe data objek yang menangani serangkaian karakter-karakter yang berjenis array. args, adalah variabel objek.

Kemudian tulis kode **Scanner masuk = new Scanner(System.in);** yang berfungsi untuk inisiasi object Scanner dengan parameter System.in. Dibawahnya kita deklarasikan beberapa atribut/variabel dengan berbagai tipe data yaitu gajiPokok(integer), tunjangan(integer), jumlahGaji(integer), dan nama (String). Lalu di tahap selanjutnya kita tinggal buat kalimat-kalimat yang akan dicetak beserta kode-kode **input.next** yang berfungsi untuk memerintahkan program untuk menerima input dari console dan mengonversinya ke dalam tipe data yang sesuai. Jika yang diinginkan adalah String maka bisa menggunakan method **nextLine()**, jika ingin input integer dapat menggunakan **nextInt()**, jika ingin input data desimal dapat menggunakan **nextDouble()** dan seterusnya.

Untuk penulisan baris-baris kodenya seperti ini:

```
System.out.print("Nama : ");  
nama = masuk.nextLine();
```

yang artinya kita mencetak "Nama :." lalu diikuti oleh inputan bertipe data String yang sudah diinisialisai dengan variabel nama. Pada inputan ini, saya memasukkan nama "Raden Isnawan" yaitu nama saya sendiri.

```
System.out.print("Gaji pokok : ");  
gajiPokok = masuk.nextInt();
```

yang artinya kita mencetak "Gaji pokok:." lalu diikuti oleh inputan bertipe data integer yang sudah diinisialisai dengan variabel gajiPokok. Pada inputan yang ini saya memasukkan gaji sebesar 50000000

```
System.out.print("tunjangan : ");  
tunjangan = masuk.nextInt();
```

yang artinya kita mencetak "tunjangan:." lalu diikuti oleh inputan bertipe data integer yang sudah diinisialisai dengan variabel tunjangan. Pada inputan tunjangan ini saya masukkan tunjangan sebesar 70000000

Lalu kita deklarasikan **jumlahGaji = gajiPokok + tunjangan** yang artinya jumlahGaji adalah hasil tambah dari gajiPokok dan tunjangan yang tadi sudah saya input.

Yang terakhir, kita tulis dua baris kode ini:

```
System.out.println("nama = "+nama); = mencetak nama yang tadi sudah diinput  
System.out.println("Jumlah gaji = " + jumlahGaji); = mencetak jumlahGaji (hasil tambah dari  
gajiPokok dan tunjangan yang tadi sudah saya input.)
```

****program LabelTest.java di halaman selanjutnya***

LabelTest.java

```
Jumlah.java  LabelTest.java x
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
public class LabelTest extends JFrame {
    private JLabel label;
    public LabelTest()
    {
        super( "Mencoba JLabel" );
        Container container = getContentPane();
        container.setLayout( new FlowLayout() );
        label = new JLabel( "Label dengan text" );
        label.setToolTipText( "Ini adalah label1" );
        container.add( label );
        setSize( 500, 400 );
        setVisible( true );
    }
    public static void main( String args[] )
    {
        LabelTest aplikasi = new LabelTest();
        aplikasi.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    }
}
```

output:



penjelasan:

pertama kita mengimport AWT di dalam program java tersebut. Abstract Windowing Toolkit (AWT), atau disebut juga "Another Windowing Toolkit", adalah pustaka windowing bertujuan umum dan multiplatform serta menyediakan sejumlah kelas untuk membuat GUI di Java. Dengan AWT, dapat membuat window, menggambar, manipulasi gambar, dan komponen seperti Button, Scrollbar, Checkbox, TextField, dan menu pull-down. Fungsi setiap baris kode nya:

- **import java.awt.*;** merupakan fungsi yang digunakan untuk proses inputan dalam program java.
- **import java.awt.event.*;** merupakan fungsi import yang digunakan untuk proses deklarasi dari class event yang berisi komponen.
- **import javax.swing.*;** merupakan fungsi sebagai syarat penggunaan komponen yang ada dalam sistem java.

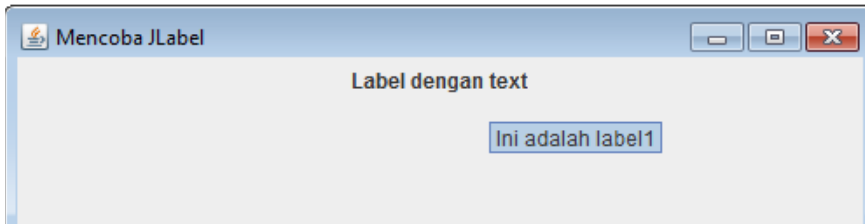
Setelah itu kita deklarasi kelas yang memiliki modifier public yaitu **public class LabelTest extends JFrame**. Kenapa harus diikuti dengan extends JFrame? Karena LabelTest adalah kelas turunan dari kelas JFrame yang mewarisi metoda terakses baik dari kelas JFrame itu sendiri maupun dari superclass berlapis lain dari mana kelas JFrame diturunkan. Beberapa metoda yang diturunkan adalah setLocation, setVisible, setTitle, setSize dan setDefaultCloseOperation.

Kemudian kita deklarasikan JLabel bernama label dengan modifier private (Menyatakan bahwa kelas tersebut tidak dapat diakses sama sekali oleh kelas lain bahkan juga tidak dapat diturunkan). Lalu kita buat konstruktor bernama LabelTest(). Di dalamnya ada super("Mencoba JLabel"). Kenapa super? Karena super memanggil constructor dari superclass yang memiliki argumen-argumen yang sesuai. Public class LabelTest() memanggil constructor dari superclass-nya yaitu class JFrame pada baris kode **public class LabelTest extends JFrame**.

container container = getContentPane(); merupakan fungsi pembentukkan objek (keyword)

tanpa proses instansiasi. `getContentPane()` mengatur properti dari obyek content pane. Lalu kita buat layout di dalam container dengan baris kode **`container.setLayout(new FlowLayout());`**. Kelas `FlowLayout` adalah layout manager yang cukup sederhana. `FlowLayout` adalah default layout manager untuk kelas `Applet`, `Panel` dan `JPanel`. `FlowLayout()` Mengkonstruksi obyek `FlowLayout` dengan alignment memusat dan ukuran celah (gap) horisontal dan vertikal adalah 5 pixel.

Kemudian kita membuat label baru yang sudah diinisialisasi dengan class `JLabel` yang akan mengeluarkan kalimat "Label dengan text" dengan kode **`label = new JLabel("Label dengan text");`**. Setelah itu kita akan deklarasikan tooltip dengan baris kode **`label.setToolTipText("Ini adalah label1");`**. Kalimat itu hanya akan muncul jika kita mengarahkan kursor ke kalimat yang dimiliki/berada di label, seperti pada gambar dibawah ini:



Lalu,

`container.add(label);` berfungsi untuk menambahkan label
`setSize(500, 400);` menentukan ukuran jendela
`setVisible(true);` menampilkan jendela ke layar

Kemudian buat objek bernama “aplikasi” dengan kode **`LabelTest aplikasi = new LabelTest();`**

Untuk menutup jendela frame, kita dapat menggunakan metoda `setDefaultCloseOperation` dari kelas `JFrame` dan memberi nilai untuk parameter operation konstanta statik `EXIT_ON_CLOSE`.

PRAKTIK 2: Membuat sebuah paket sederhana

```
paket.java x
package mahasiswa;
public class paket
{
    public static void isiPaket(){
        System.out.println("ini adalah hasil import");
    }
}
```

penjelasan:

Package digunakan pada Java sebenarnya untuk memudahkan mengorganisir file dari class. Package ini merupakan mekanisme dari encapsulation suatu kelompok atau grup yang terdiri dari class – class, sub packages dan juga interfaces. Hal ni akan memberi kemudahan, karena banyak penerapan dari program java menggunakan konsep hirarki untuk mengatur file-file dari class dan source.

Yang perlu diperhatikan disini adalah pada saat pembuatan class. Sebelum suatu class dapat digunakan ulang melalui pernyataan import, maka class tersebut harus di simpan di dalam package. Langkah-langkah membuat class di dalam package adalah sebagai berikut:

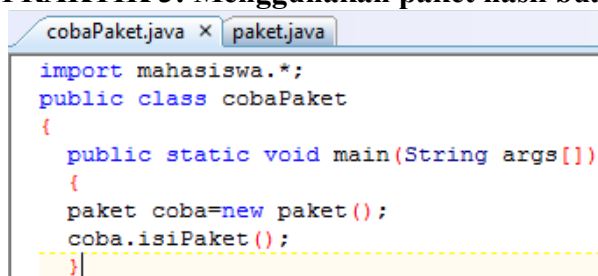
1. Mendeklarasikan class sebagai public. Jika class tidak dideklarasikan demikian, maka class tersebut hanya dapat digunakan oleh class lainnya yang berada dalam package yang sama.
2. Memilih nama yang unik untuk package tersebut dan menambahkan deklarasi package pada file source code untuk deklarasi class yang digunakan ulang
3. Melakukan compile class

4. Meng-import class yang akan digunakan ulang di dalam program dilanjutkan dengan penggunaan class.

Pada baris pertama dan kedua script program tersebut, class packet berada di dalam package mahasiswa. Hal yang penting dalam membuat package adalah mengatur struktur direktori pada disk kita dan disesuaikan dengan nama package yang kita pilih. Jika package kita hanya memiliki satu nama seperti program di atas yaitu mahasiswa, maka kita hanya perlu membuat satu direktori untuk package tersebut.

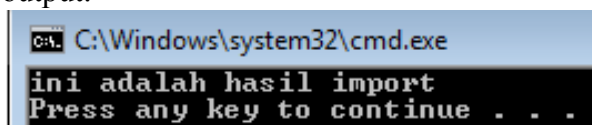
Lalu yang terakhir kita buat method bernama isiPaket() dengan baris kode **public static void isiPaket()**, lalu cetak kalimat dengan **System.out.println("ini adalah hasil import");**. Output tidak bisa dikeluarkan karena perlu main method dan kelas lain untuk memunculkan outputnya.

PRAKTIK 3: Menggunakan paket hasil buatan sendiri



```
cobaPaket.java x paket.java
import mahasiswa.*;
public class cobaPaket
{
    public static void main(String args[])
    {
        paket coba=new paket();
        coba.isiPaket();
    }
}
```

output:



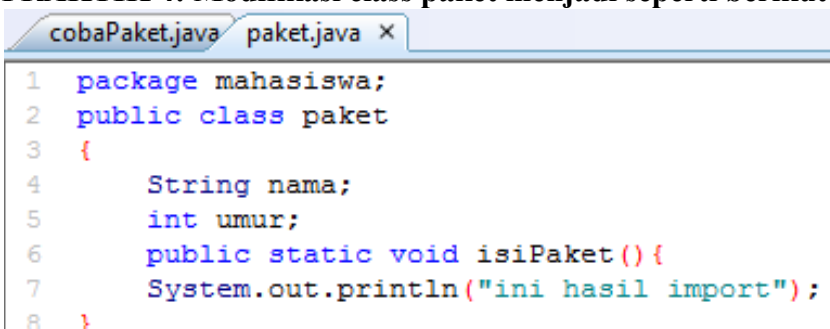
```
C:\Windows\system32\cmd.exe
ini adalah hasil import
Press any key to continue . . .
```

penjelasan:

dalam script program di atas, bisa kita lihat bahwa kita mengimport package mahasiswa. Import merupakan kata kunci untuk melibatkan class-class lain yang terhimpun di dalam package yang akan kita import. Pelibatan kelas dari package dapat hanya sebagian class (sub class) atau semuanya. Pada program di atas, kita mengimport semua class yaitu dengan kode **import mahasiswa.*;**

program tersebut memiliki class yang memiliki modifier public yaitu cobaPaket. Lalu deklarasikan **public static void main(String args[])** untuk mengawali eksekusi. Di dalamnya, kita buat objek bernama “coba” dengan membuat baris kode **paket coba=new paket();**. Kemudian kita inisialisasi objek coba dengan method isiPaket yaitu dengan menuliskan **coba.isiPaket();**, sehingga kalimat yang berada di dalam isiPaket() yaitu “ini adalah hasil import” berhasil dicetak di output program.

PRAKTIK 4: Modifikasi class paket menjadi seperti berikut



```
cobaPaket.java x paket.java x
1 package mahasiswa;
2 public class paket
3 {
4     String nama;
5     int umur;
6     public static void isiPaket(){
7         System.out.println("ini hasil import");
8     }
}
```

```

9  public void setData(int umur, String nama){
10     this.umur = umur;
11     this.nama = nama;
12 }
13     public void info(){
14         System.out.println("Nama : "+nama);
15         System.out.println("Umur : "+umur);
16     }
17 }

```

penjelasan:

Modifikasi yang kita lakukan adalah menambahkan method-method baru. Yang pertama kita membuat method `setData` yang memiliki parameter `int umur` dan `String nama` yang ditulis dengan baris kode **`public void setData(int umur, String nama)`**. Lalu untuk mengakses variabel/atribut yang kita buat tadi, kita perlu menginisialisasinya dengan **`this.umur = umur;`** dan **`this.nama = nama;`**.

Yang terakhir kita buat sebuah method bernama `info()` dengan cara menulis **`public void info()`**, yang di dalamnya kita mencetak output yaitu berupa kalimat ditambah variabel nama dan umur:

`System.out.println("Nama : "+nama);`

`System.out.println("Umur : "+umur);`

PRAKTIK 5: Panggil method tersebut dari class `cobaPaket`!

```

cobaPaket.java x paket.java
import mahasiswa.*;
public class cobaPaket
{
    public static void main(String args[])
    {
        paket coba=new paket ();
        coba.isiPaket ();
        coba.info ();
    }
}

```

Output:

```

C:\Windows\system32\cmd.exe
ini hasil import
Nama : null
Umur : 0
Press any key to continue . . .

```

penjelasan:

dalam `cobaPaket.java`, kita tambahkan inisialisasi objek `coba` yaitu dengan method `info()` milik class `paket`. Hasilnya adalah yang pertama muncul kalimat dari method `isiPaket()` yaitu “ini hasil import” lalu dibawahnya muncul kalimat + variabel/atribut beserta value milik method `info()` yaitu:

Nama : null

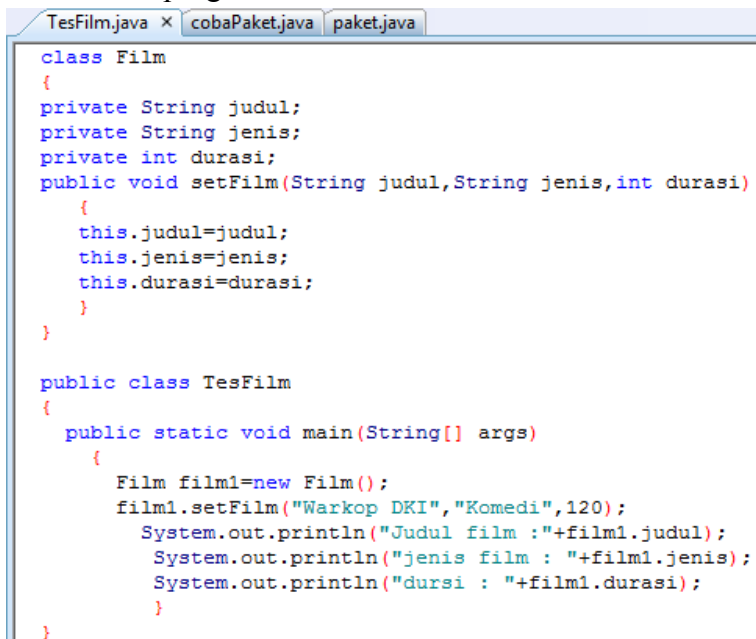
Umur : 0

Kenapa Nama memiliki value `null` dan Umur memiliki value `0`? Tentu saja karena di class `paket`, kedua variabel/atribut tersebut tidak diberi value apapun sehingga hasil yang muncul adalah hasil default yaitu `null` untuk variabel `String` dan `0` untuk variabel `integer`.

**Latihan ada di halaman selanjutnya*

LATIHAN

1. Ketikkan program berikut



```
TesFilm.java x cobaPaket.java paket.java
class Film
{
    private String judul;
    private String jenis;
    private int durasi;
    public void setFilm(String judul,String jenis,int durasi)
    {
        this.judul=judul;
        this.jenis=jenis;
        this.durasi=durasi;
    }
}

public class TesFilm
{
    public static void main(String[] args)
    {
        Film film1=new Film();
        film1.setFilm("Warkop DKI","Komedi",120);
        System.out.println("Judul film :"+film1.judul);
        System.out.println("jenis film : "+film1.jenis);
        System.out.println("dursi : "+film1.durasi);
    }
}
```

penjelasan:

pertama kita membuat class Film yang fungsinya untuk mendeklarasikan kelas bernama Film yang gunanya untuk mengumpulkan fungsi atau atribut yaitu private String judul; private String jenis; dan private int durasi;. Lalu kita membuat method setFilm yang memiliki parameter String judul, String jenis, int durasi yang ditulis dengan baris kode **public void setFilm(String judul,String jenis,int durasi)**. Lalu untuk mengakses variabel/atribut yang kita buat tadi, kita perlu menginisialisasikannya dengan this.judul=judul; this.jenis=jenis; this.durasi=durasi;.

Lalu kita buat public class TesFilm yang fungsinya sebagai kelas yang menampung pembuatan objek serta proses outputnya. Pertama kita tulis **public static void main(String args[])** untuk mengawali eksekusi. Di dalamnya kita membuat objek pada kelas Film bernama “film1” yang ditulis dengan **Film film1=new Film()**;. Setelah itu kita inisialisasikan objek film1 dengan method setFilm dengan memasukkan value di dalam setiap atributnya yaitu dengan baris kode **film1.setFilm("Warkop DKI","Komedi",120)**;. Lalu outputnya kita cetak dengan baris-baris kode berikut:

```
System.out.println("Judul film :"+film1.judul);
```

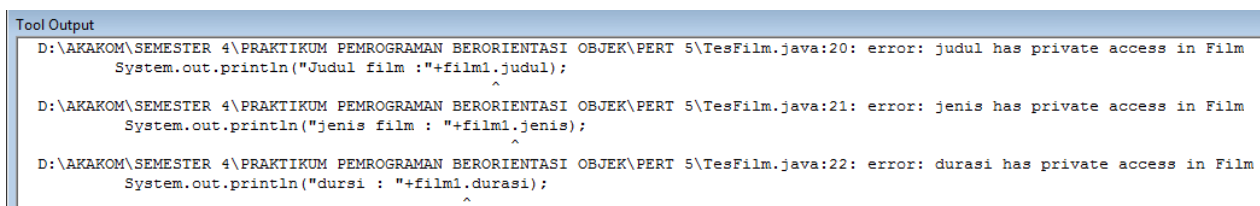
```
System.out.println("jenis film : "+film1.jenis);
```

```
System.out.println("dursi : "+film1.durasi);
```

Soal:

a) Apa yang terjadi saat class TesFilm dikompilasi? Mengapa demikian?

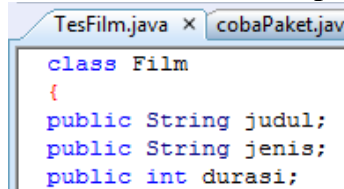
Yang terjadi adalah program tidak bisa dijalankan seperti gambar di bawah ini:



```
Tool Output
D:\AKAKOM\SEMESTER 4\PRAKTIKUM PEMROGRAMAN BERORIENTASI OBJEK\PERT 5\TesFilm.java:20: error: judul has private access in Film
    System.out.println("Judul film :"+film1.judul);
                                   ^
D:\AKAKOM\SEMESTER 4\PRAKTIKUM PEMROGRAMAN BERORIENTASI OBJEK\PERT 5\TesFilm.java:21: error: jenis has private access in Film
    System.out.println("jenis film : "+film1.jenis);
                                   ^
D:\AKAKOM\SEMESTER 4\PRAKTIKUM PEMROGRAMAN BERORIENTASI OBJEK\PERT 5\TesFilm.java:22: error: durasi has private access in Film
    System.out.println("dursi : "+film1.durasi);
                                   ^
```

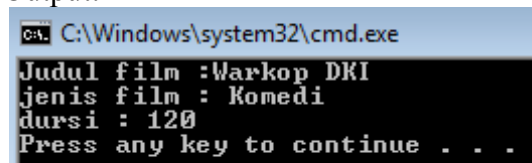

keterangan di gambar itu adalah setiap atribut di kelas Film memiliki private access sehingga tidak bisa diakses oleh kelas TesFilm. Variable dan method yang diberikan hak akses private hanya bisa diakses oleh class itu sendiri, dan data-data tersebut tidak bisa diwariskan pada subclass atau class lainnya. Agar class lain bisa mengakses variable atau method tersebut maka perlu dibuatkan method yang mempunyai hak akses public (package yang berbeda) atau protected (pada package yang sama), tetapi di script program di atas tidak ada method khusus yang punya hak akses public.

b) Ubah modifier atribut pada class Film menjadi public ! jalankan dan jelaskan hasilnya !



```
TesFilm.java x cobaPaket.jav
class Film
{
    public String judul;
    public String jenis;
    public int durasi;
```

Output:



```
C:\Windows\system32\cmd.exe
Judul film :Warkop DKI
jenis film : Komedi
dursi : 120
Press any key to continue . . .
```

penjelasan:

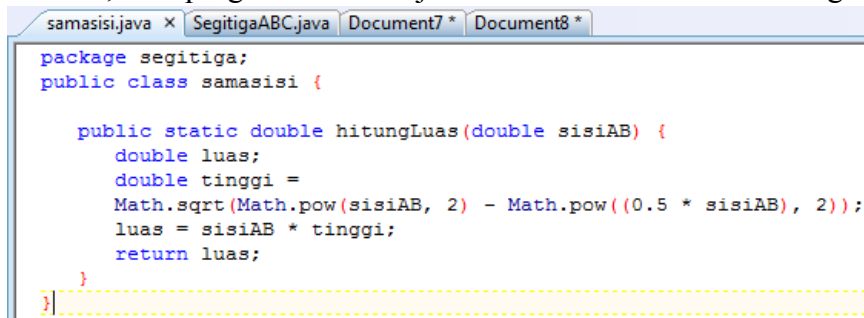
Program TesFilm.java sudah bisa dijalankan karena modifier setiap atribut di dalam class Film sudah diganti dengan public. Public digunakan pada atribut-atribut tersebut agar dapat diakses oleh seluruh class didalam package yang sama atau diluar package yang berbeda, modifier jenis ini memiliki tingkat akses yang sangat luas, hingga seluruh sumber daya dapat diakses oleh class manapun tanpa batasan. Sehingga, program dapat berjalan dengan semestinya.

TUGAS

Buatlah program dengan menggunakan paket java yang lain !

Saya membuat program dengan 3 package yang terdiri dari user-defined package (segitiga), dan built-in package (javax.swing.JOptionPane dan java.text.DecimalFormat;). Program java yang saya buat ini berfungsi untuk menghitung luas segitiga sama sisi.

Pertama, buat program samasisi.java di dalam folder direktori segitiga



```
samasisi.java x SegitigaABC.java Document7 * Document8 *
package segitiga;
public class samasisi {

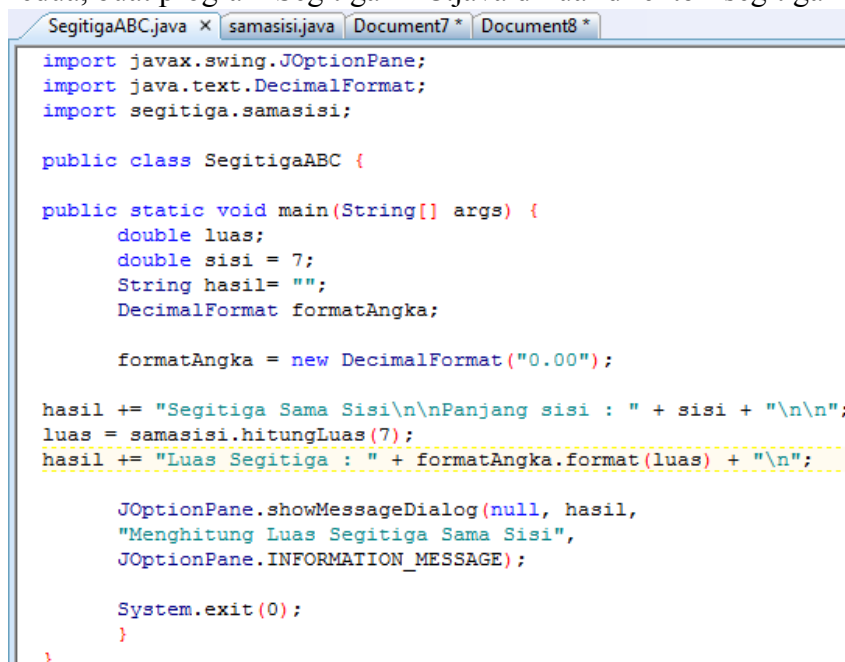
    public static double hitungLuas(double sisiAB) {
        double luas;
        double tinggi =
            Math.sqrt(Math.pow(sisiAB, 2) - Math.pow((0.5 * sisiAB), 2));
        luas = sisiAB * tinggi;
        return luas;
    }
}
```

penjelasan:

Pertama, kita membuat package user-defined bernama segitiga. Setelah itu kita mendeklarasikan class yang memiliki modifier public bernama samasisi. Di dalamnya kita membuat suatu method

bernama `hitungLuas` yang memiliki parameter `double sisiAB`, kita melakukannya dengan menulis baris kode **`public static double hitungLuas(double sisiAB) {`**. Di dalamnya kita deklarasi atribut `luas` dan atribut `tinggi`, kedua-duanya bertipe data `double`. Atribut `tinggi` memiliki kalkulasi di dalamnya yaitu **`Math.sqrt(Math.pow(sisiAB, 2) - Math.pow((0.5 * sisiAB), 2));`**. Method `Math.sqrt()` digunakan untuk mencari hasil dari akar kuadrat sebuah angka. Fungsi ini membutuhkan setidaknya 1 argumen, yaitu angka yang akan dihitung. Method `Math.pow()` berfungsi untuk mencari hasil pemangkatan. Fungsi ini membutuhkan 2 buah argumen. Perhitungannya tidak usah saya jelaskan karena inti pertemuan kali ini bukan menghitung-hitung, yang penting paham tentang package. Kemudian `luas` sama dengan `sisiAB` dikali dengan `tinggi` yang sudah kita hitung tadi. Yang terakhir kita `return luas` supaya perhitungannya terkalkulasi.

Kedua, buat program `SegitigaABC.java` di luar direktori `segitiga`



```

SegitigaABC.java x samasisi.java Document7 * Document8 *
import javax.swing.JOptionPane;
import java.text.DecimalFormat;
import segitiga.samasisi;

public class SegitigaABC {

    public static void main(String[] args) {
        double luas;
        double sisi = 7;
        String hasil = "";
        DecimalFormat formatAngka;

        formatAngka = new DecimalFormat("0.00");

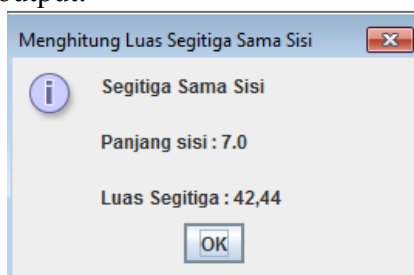
        hasil += "Segitiga Sama Sisi\n\nPanjang sisi : " + sisi + "\n\n";
        luas = samasisi.hitungLuas(7);
        hasil += "Luas Segitiga : " + formatAngka.format(luas) + "\n";

        JOptionPane.showMessageDialog(null, hasil,
            "Menghitung Luas Segitiga Sama Sisi",
            JOptionPane.INFORMATION_MESSAGE);

        System.exit(0);
    }
}

```

output:



penjelasan:

`import javax.swing.JOptionPane;`

`JOptionPane` merupakan sebuah kelas yang menyediakan jendela dialog. `JOptionPane` bisa kita gunakan untuk mengambil input, menampilkan informasi, menampilkan pesan error, menampilkan dialog konfirmasi, dan lain-lain. Sebelum menggunakan `JOptionPane`, kita harus mengimpornya terlebih dahulu ke dalam program: `import javax.swing.`

`import java.text.DecimalFormat;`

`DecimalFormat` berfungsi untuk mengontrol display dari leading dan trailing zeros, prefixes dan suffixes, grouping(ribuan) dan decimal. Dengan `DecimalFormat` inilah kita akan mendapatkan flexibility dalam mem-format number/angka.

import segitiga.samasisi;
kita mengimport package segitiga dan class samasisi

Lalu kita deklarasi kelas Segitiga ABC. Di dalamnya terdapat method main yaitu **public static void main(String[] args)** untuk mengawali eksekusi program. Di dalamnya ada beberapa variabel/atribut yaitu double luas; double sisi = 7; String hasil= ""; DecimalFormat formatAngka;. Kemudian kita membuat objek dari kelas DecimalFormat sekaligus membuat value nya yaitu ("0.00")

Kita hitung panjang sisi dengan
hasil += "Segitiga Sama Sisi\n\nPanjang sisi : " + sisi + "\n\n";

Kita hitung luas ($\frac{1}{2} \times \text{alas} \times \text{tinggi}$) dengan
hasil += "Luas Segitiga : " + formatAngka.format(luas) + "\n";

showMessageDialog() : untuk menampilkan pesan/informasi, yaitu (null, hasil, "Menghitung Luas Segitiga Sama Sisi", JOptionPane.INFORMATION_MESSAGE). Argumen pertama untuk menentukan posisi dialog, argumen kedua adalah untuk menampilkan hasil perhitungan tadi di dalam jendela dialog, argumen ketiga untuk menentukan judul jendela dialog, dan argumen keempat akan mengeluarkan simbol informasi seperti tanda pentung.

Terakhir, kita beri exit pada jendela dialog dengan **System.exit(0);**

KESIMPULAN LAPORAN

Di dalam laporan ini, kita banyak belajar tentang package, access modifier, dan static. Yang paling banyak dipelajari adalah package terutama package swing dan package yang kita buat sendiri. Di dalam package swing, ada beberapa komponen yaitu:

Jlabel Area dimana teks yang tidak bisa diedit atau sebuah icon dapat ditampilkan

JTextField Area dimana user input data dari keyboard. Area ini juga bisa menampilkan informasi

JButton Area yang memicu sebuah event ketika diklik dengan mouse

JCheckBox Komponen GUI yang bisa dipilih atau tidak dipilih

JPanel Sebuah kontainer dalam mana komponen dapat ditempatkan atau diorganisasi

Struktur Package:

```
package nm_package;  
import registration.processing.*;  
import java.util.List;  
import java.lang.*; //imported by default  
class NmClass {  
/* details of NmClass */  
}
```

Untuk modifier pada laporan ini, kita menggunakan dua yaitu public dan private. Sekian.

Terima Kasih