



**LABORATORIUM PEMBELAJARAN ILMU KOMPUTER**  
**FAKULTAS ILMU KOMPUTER**  
**UNIVERSITAS BRAWIJAYA**

---

BAB : SELEKSI KONDISI  
NAMA : KEYLA HAYU ANJANI  
NIM : 255150407111033  
TANGGAL : 08 SEPTEMBER 2025  
ASISTEN : GRACE WAHYUNI  
NADA MUSYAFFA BILHAQI

---

**Latihan 1**

(bila lebih dari 1 soal maka copy paste poin A-E dan sesuaikan dengan jumlah soalnya)

**A. SOAL**

Buatlah program sebagai berikut dengan menggunakan metode switch case

Menu :

1. menghitung luas dan keliling persegi panjang
2. menghitung luas dan keliling lingkaran
3. menghitung luas dan keliling segitiga

Pilihan anda : 3 Masukkan a : 3

Masukkan b : 4

Masukkan r : 5

Keliling segitiga : 12 cm

Luas segitiga : 6 cm<sup>2</sup>

Pilihan anda : 10

Data tak ditemukan, program dihentikan ...

**B. SOURCE CODE**

```
1 public static void menghitungLuasDanKeliling() {  
2     Scanner input = new Scanner(System.in);  
3  
4     System.out.println("Menu :");  
5     System.out.println("1. Menghitung luas dan  
6     keliling persegi panjang");  
7     System.out.println("2. Menghitung luas dan  
8     keliling lingkaran");  
9     System.out.println("3. Menghitung luas dan  
10    keliling segitiga");  
11    System.out.print("Pilihan anda :");  
12    int pilihan = input.nextInt();  
13  
14    switch (pilihan) {  
15        case 1: System.out.print("Masukan panjang :");  
16        int panjang = input.nextInt();  
17        System.out.print("Masukan lebar :");  
18        int lebar = input.nextInt();  
19  
20    }
```

```

21
22         int kelilingPersegiPanjang = 2 * (panjang + lebar);
23         System.out.println("Keliling persegi panjang : "
24 + kelilingPersegiPanjang);
25         int luasPersegiPanjang = panjang * lebar;
26         System.out.println("Luas persegi panjang : " +
27 luasPersegiPanjang); break;
28
29
30         case 2: System.out.print("Masukan phi :");
31         int phi = input.nextInt();
32         System.out.print("Masukan r :");
33         int r = input.nextInt();
34
35         int kelilingLingkarang = 2 * phi * r;
36         System.out.println("Keliling lingkaran : " +
37 kelilingLingkarang);
38         int luasLingkarang = phi * r * r;
39         System.out.println("Luas lingkaran : " +
40 luasLingkarang); break;
41
42
43         case 3: System.out.print("Masukan alas :");
44         int alas = input.nextInt();
45         System.out.print("Masukan sisi miring :");
46         int sisiMiring = input.nextInt();
47         System.out.print("Masukan tinggi :");
48         int tinggi = input.nextInt();
49
50         int kelilingSegitiga = alas + sisiMiring +
51 tinggi;
52         System.out.println("Keliling segitiga : " +
53 kelilingSegitiga);
54         double luasSegitiga = 0.5f * alas * tinggi;
55         System.out.println("Luas segitiga : " +
56 luasSegitiga); break;
57
58
59         default: System.out.println("Data tak ditemukan,
60 program dihentikan ...");
61         break;
62     }
63

```

### C. PEMBAHASAN

```

public static void main(String[] args) {
    menghitungLuasDanKeliling();
}

```

= Method main adalah titik awal eksekusi program. Di dalamnya, saya memanggil method menghitungLuasDanKeliling() untuk menjalankan proses perhitungan luas dan keliling dari beberapa bentuk geometri.

```

public static void menghitungLuasDanKeliling() {
    Scanner input = new Scanner(System.in);
}

```

= Method ini berisi logika utama untuk menghitung luas dan keliling. Pertama, saya membuat objek Scanner bernama input untuk membaca data dari pengguna melalui keyboard.

```
System.out.println("Menu :");
System.out.println("1. Menghitung luas dan keliling persegi panjang");
System.out.println("2. Menghitung luas dan keliling lingkaran");
System.out.println("3. Menghitung luas dan keliling segitiga");
System.out.print("Pilihan anda :");
int pilihan = input.nextInt();
= Program menampilkan menu pilihan. Nilai yang dimasukkan pengguna disimpan dalam variabel pilihan bertipe int, untuk menentukan bentuk yang akan dihitung.
```

```
switch (pilihan) {
= Struktur switch digunakan untuk memilih blok perhitungan berdasarkan nilai dari variabel pilihan.
```

```
case 1:
System.out.print("Masukan panjang :");
int panjang = input.nextInt();
System.out.print("Masukan lebar :");
int lebar = input.nextInt();
= Program meminta pengguna memasukkan nilai panjang dan lebar. Nilai disimpan dalam variabel panjang dan lebar bertipe int.
```

```
int kelilingPersegiPanjang = 2 * (panjang + lebar);
System.out.println("Keliling persegi panjang : " + kelilingPersegiPanjang);
int luasPersegiPanjang = panjang * lebar;
System.out.println("Luas persegi panjang : " + luasPersegiPanjang);
break;
= Program menghitung keliling dengan rumus  $2 \times (\text{panjang} + \text{lebar})$ , dan luas dengan rumus  $\text{panjang} \times \text{lebar}$ . Hasil disimpan di variabel kelilingPersegiPanjang dan luasPersegiPanjang, lalu ditampilkan ke layar.
```

```
case 2:
System.out.print("Masukan phi :");
int phi = input.nextInt();
System.out.print("Masukan r :");
int r = input.nextInt();
= Program meminta pengguna memasukkan nilai phi dan jari-jari (r). Nilai disimpan dalam variabel phi dan r bertipe int.
```

```
int kelilingLingkaran = 2 * phi * r;
System.out.println("Keliling lingkaran : " + kelilingLingkaran);
int luasLingkaran = phi * r * r;
```

```
System.out.println("Luas lingkaran :" + luasLingkaran);  
break;
```

= Program menghitung keliling dengan rumus  $2 \times \text{phi} \times r$ , dan luas dengan rumus  $\text{phi} \times r \times r$ . Hasil disimpan di variabel kelilingLingkaran dan luasLingkaran, lalu ditampilkan ke layar.

case 3:

```
System.out.print("Masukan alas :");  
int alas = input.nextInt();  
System.out.print("Masukan sisi miring :");  
int sisiMiring = input.nextInt();  
System.out.print("Masukan tinggi :");  
int tinggi = input.nextInt();
```

= Program meminta pengguna memasukkan nilai alas, sisi miring, dan tinggi. Nilai disimpan dalam variabel alas, sisiMiring, dan tinggi bertipe int.

```
int kelilingSegitiga = alas + sisiMiring + tinggi;  
System.out.println("Keliling segitiga :" + kelilingSegitiga);  
double luasSegitiga = 0.5f * alas * tinggi;  
System.out.println("Luas segitiga :" + luasSegitiga);  
break;
```

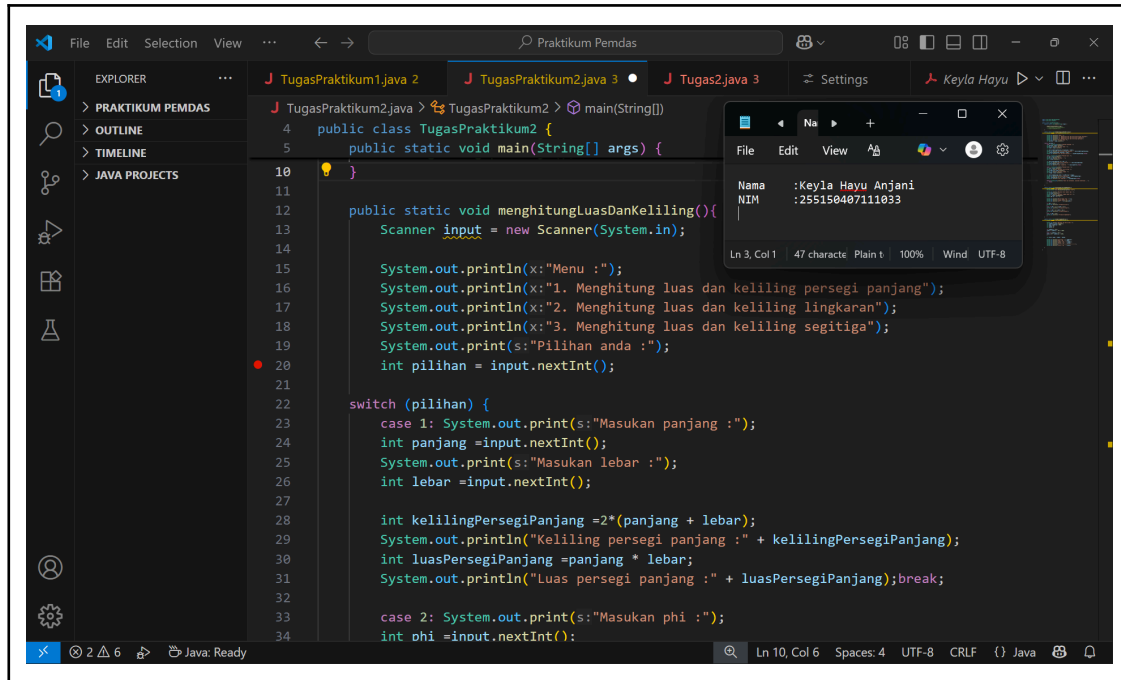
= Program menghitung keliling dengan menjumlahkan ketiga sisi, dan luas dengan rumus  $0,5 \times \text{alas} \times \text{tinggi}$ . Hasil disimpan di variabel kelilingSegitiga dan luasSegitiga, lalu ditampilkan ke layar.

default:

```
System.out.println("Data tak ditemukan, program dihentikan ...");  
break;
```

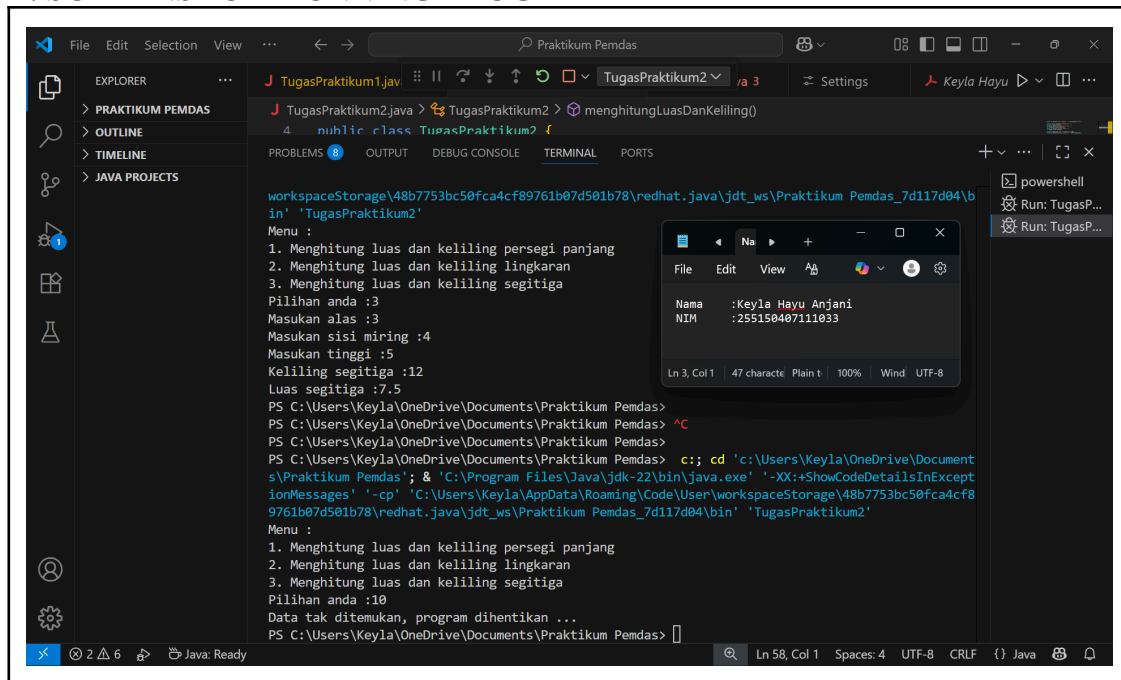
= Jika pengguna memasukkan angka di luar 1–3, program akan menampilkan pesan bahwa data tidak ditemukan dan menghentikan proses.

## D. SCREENSHOT PROGRAM



Gambar 1.1 SS class mobil

## E. SCREENSHOT RUNNING PROGRAM



Gambar 1.1 SS class mobil

## F. SOAL

Untuk menentukan kriteria kegemukan, digunakan IMT (Indeks Massa Tubuh), yang bisa dihitung menggunakan rumus :

$$\text{IMT} = b / t^2$$

b = berat badan (kg)

t = tinggi badan

(m) Kriteria untuk

nilai IMT ditabelkan sebagai berikut :

Nilai IMT	Kriteria
$\text{IMT} \leq 18,5$	Kurus
$18,5 < \text{IMT} \leq 25$	Normal
$25 < \text{IMT} \leq 30$	Gemuk
$\text{IMT} > 30$	Kegemukan

Susun program dengan tampilan sebagai berikut dengan menggunakan metode if-else!

## G. SOURCE CODE

```
1 public static void menentukanKriteriaKegemukan() {
2     Scanner input = new Scanner(System.in);
3     DecimalFormat X=new DecimalFormat("#.##");
4
5     System.out.print("Masukan berat badan (kg) :");
6     int b =input.nextInt();
7     System.out.print("Masukan tinggi badan (m) :");
8     double t =input.nextDouble();
9     double IMT =b / (t * t);
10
11
12     System.out.println("Berat badan (kg) : " + b);
13     System.out.println("Tinggi badan (m) : " + t);
14     System.out.println("IMT : " + X.format(IMT));
15
16     if (IMT < 18.5) {
17         System.out.println("Termasuk kurus");
18     }
19     else if (IMT > 18.5 & IMT <= 25) {
20         System.out.println("Termasuk normal");
21     }
22     else if (IMT > 25 & IMT <= 30) {
23         System.out.println("Termasuk gemuk");
24     }
25     else if (IMT > 30) {
26         System.out.println("Termasuk kegemukan");
27     }
28 }
29
30
31
32
33
34
35
```

## H. PEMBAHASAN

```
public static void main(String[] args) {  
    menentukanKriteriaKegemukan();  
  
}
```

= Method main adalah titik awal eksekusi program. Di dalamnya, saya memanggil method `menentukanKriteriaKegemukan()`.

```
public static void menentukanKriteriaKegemukan() {  
    Scanner input = new Scanner(System.in);  
    DecimalFormat X = new DecimalFormat("#.##");
```

= Pertama, saya membuat objek Scanner bernama input untuk membaca data dari pengguna. Lalu saya membuat objek DecimalFormat bernama X untuk membatasi hasil desimal hanya dua angka di belakang koma.

```
System.out.print("Masukan berat badan (kg) :");  
int b = input.nextInt();
```

= Program meminta pengguna untuk memasukkan berat badan dalam satuan kilogram. Nilai disimpan dalam variabel b bertipe int.

```
System.out.print("Masukan tinggi badan (m) :");  
double t = input.nextDouble();
```

= Program meminta pengguna untuk memasukkan tinggi badan dalam satuan meter. Nilai disimpan dalam variabel t bertipe double agar bisa menangani angka desimal.

```
double IMT = b / (t * t);
```

= Program menghitung Indeks Massa Tubuh (IMT) dengan rumus berat badan / (tinggi badan x tinggi badan). Hasilnya disimpan dalam variabel IMT bertipe double.

```
if (IMT < 18.5) {  
    System.out.println("Termasuk kurus");  
}
```

```
else if (IMT > 18.5 & IMT <= 25) {  
    System.out.println("Termasuk normal");  
}
```

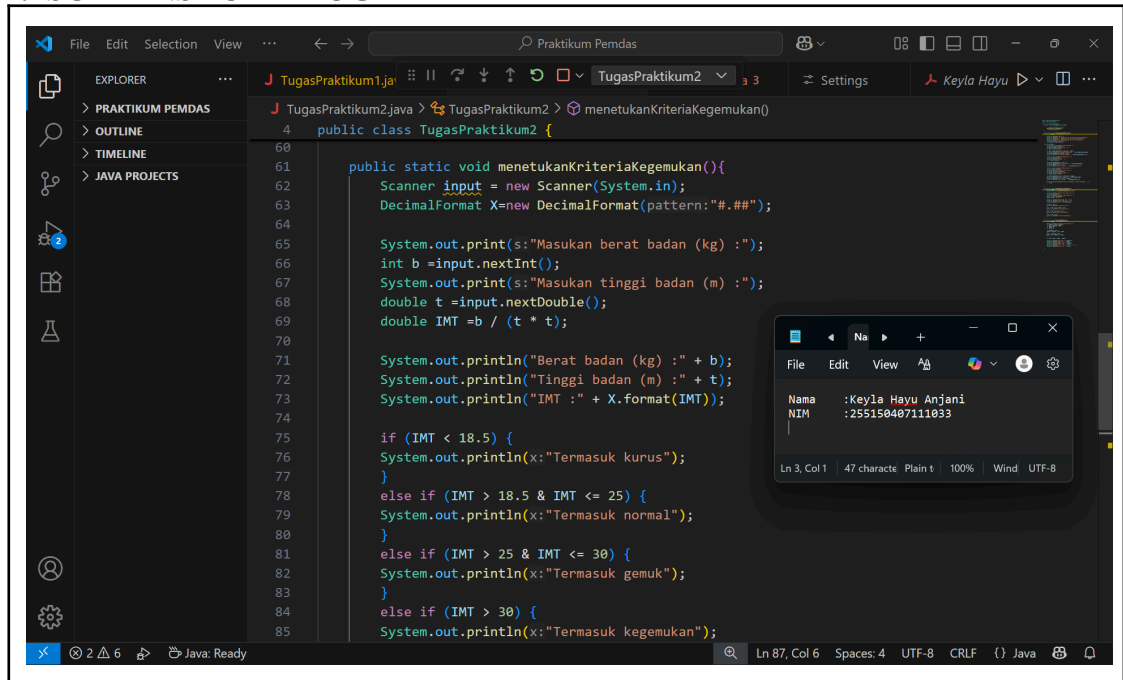
```
else if (IMT > 25 & IMT <= 30) {  
    System.out.println("Termasuk gemuk");  
}
```

```
else if (IMT > 30) {  
    System.out.println("Termasuk kegemukan");  
}
```

= Program menggunakan struktur percabangan if-else untuk menentukan kategori berat badan berdasarkan nilai IMT:

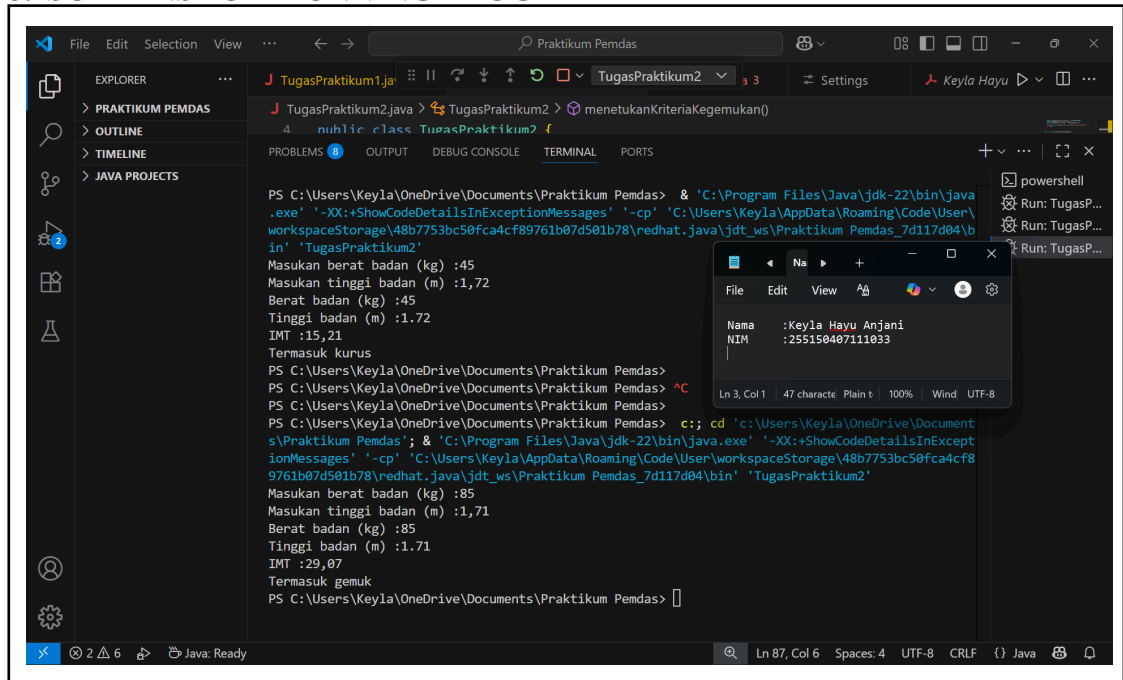
1. Jika IMT kurang dari 18.5, maka termasuk kurus.
2. Jika IMT lebih dari 18.5 dan kurang dari atau sama dengan 25, maka termasuk normal.
3. Jika IMT lebih dari 25 dan kurang dari atau sama dengan 30, maka termasuk gemuk.
4. Jika IMT lebih dari 30, maka termasuk kegemukan.

## I. SCREENSHOT PROGRAM



Gambar 1.1 SS class mobil

## J. SCREENSHOT RUNNING PROGRAM



Gambar 1.1 SS class mobil



## K. SOAL

Susun program untuk masalah pengajian sebagai berikut :

Masukan yang dibutuhkan oleh program adalah : jumlah jam kerja tiap minggu.

Keluaran program adalah : total upah dari pegawai tertentu.

Aturan yang diterapkan adalah :

- Batas kerja maksimal adalah 60 jam / minggu, dengan upah Rp. 5000,- / jam.

Kelebihan jam kerja dari batas maksimum akan dianggap sebagai lembur dengan upah Rp. 6000,- / jam.

- Batas kerja minimal adalah 50 jam / minggu. Apabila pegawai mempunyai jam kerja di bawah batas kerja minimal ini, maka akan dikenakan denda sebesar Rp. 1000, - / jam.

Contoh tampilan :

Jam kerja	:	55
Upah	=	Rp. 275000
Lembur	=	Rp. 0
Denda	=	Rp. 0
Total	=	Rp. 275000

Jam kerja	:	70
Upah	=	Rp. 300000
Lembur	=	Rp. 60000
Denda	=	Rp. 0
Total	=	Rp. 360000

Jam kerja	:	40
Upah	=	Rp. 200000
Lembur	=	Rp. 0
Denda	=	Rp. 10000
Total	=	Rp. 190000

## L. SOURCE CODE

```
1 public static void menghitungUpahKerja() {
2     Scanner input = new Scanner(System.in);
3
4     System.out.print("Masukan jam kerja anda :");
5     int jamKerja =input.nextInt();
6     int upah =jamKerja * 5000;
7     int lembur =0;
8     int denda =0;
9
10
11     if (jamKerja > 60) {
12         lembur= (jamKerja - 60) * 6000;
13     }
14     else if (jamKerja < 50) {
15         denda =(50 - jamKerja) * 1000;
16     }
17
18 }
```

19	int total =upah + lembur - denda;
20	
21	System.out.println("Jam kerja : " + jamKerja);
22	System.out.println("Upah = Rp. " + upah);
23	System.out.println("Lembur = Rp. " + lembur);
24	System.out.println("Denda = Rp. " + denda);
25	
26	
27	System.out.println("-----");
28	System.out.println("Total = Rp. " + total);
29	
30	}
31	

## M. PEMBAHASAN

```
public static void main(String[] args) {
    menghitungUpahKerja();
}
```

= Method main adalah titik awal eksekusi program. Di dalamnya, saya memanggil method menghitungUpahKerja() untuk menjalankan proses penghitungan upah kerja berdasarkan jam kerja.

```
public static void menghitungUpahKerja() {
    Scanner input = new Scanner(System.in);
```

= Method ini berisi logika utama untuk menghitung upah kerja, termasuk lembur dan denda. Pertama, saya membuat objek Scanner bernama input untuk membaca data dari pengguna melalui keyboard.

```
System.out.print("Masukan jam kerja anda :");
int jamKerja = input.nextInt();
```

= Program meminta pengguna untuk memasukkan jumlah jam kerja. Nilai disimpan dalam variabel jamKerja bertipe int.

```
int upah = jamKerja * 5000;
int lembur = 0;
int denda = 0;
```

= Program menghitung upah dasar dengan jamKerja x 5000. Variabel lembur dan denda diinisialisasi dengan nilai 0, karena belum tentu ada lembur atau denda tergantung kondisi jam kerja.

```
if (jamKerja > 60) {
    lembur = (jamKerja - 60) * 6000;
}
else if (jamKerja < 50) {
    denda = (50 - jamKerja) * 1000;
}
```

= Program menggunakan struktur if-else untuk menentukan apakah pekerja mendapatkan lembur atau dikenakan denda:

1. Jika jam kerja lebih dari 60, maka kelebihan jam dihitung sebagai lembur dengan tarif Rp.

6000 per jam.

2. Jika jam kerja kurang dari 50, maka kekurangan jam dikenakan denda sebesar Rp. 1000 per jam.

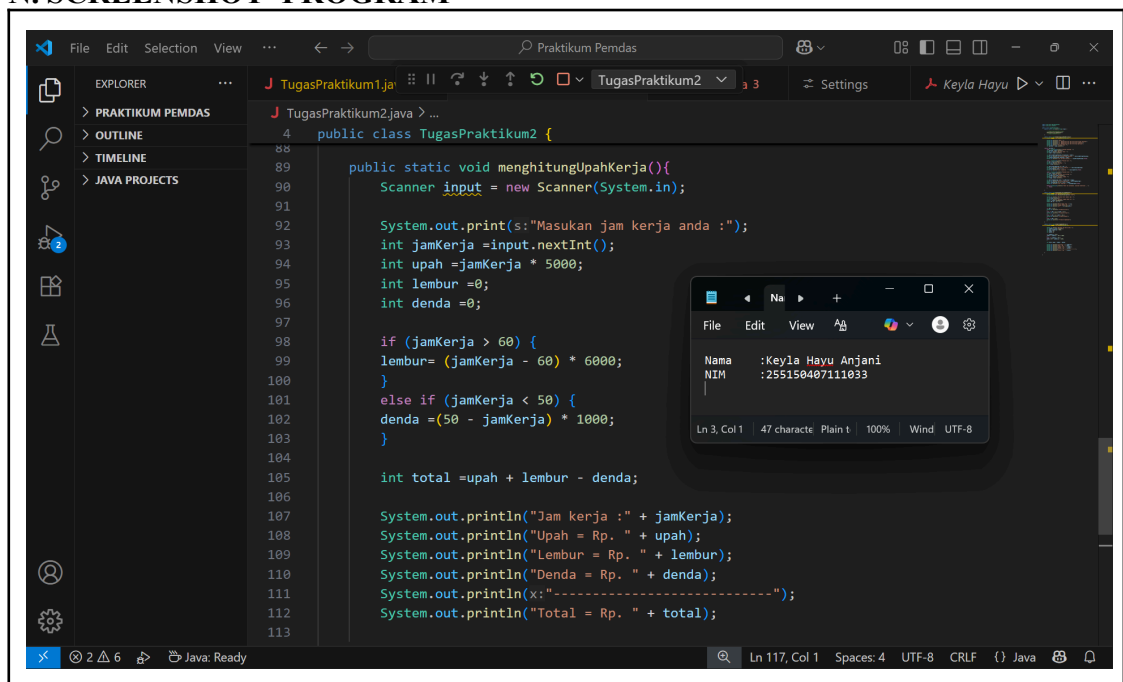
```
int total = upah + lembur - denda;
```

= Program menghitung total upah yang diterima dengan rumus :upah + lembur - denda. Hasil disimpan dalam variabel total bertipe int.

```
System.out.println("Jam kerja :" + jamKerja);
System.out.println("Upah = Rp. " + upah);
System.out.println("Lembur = Rp. " + lembur);
System.out.println("Denda = Rp. " + denda);
System.out.println("-----");
System.out.println("Total = Rp. " + total);
```

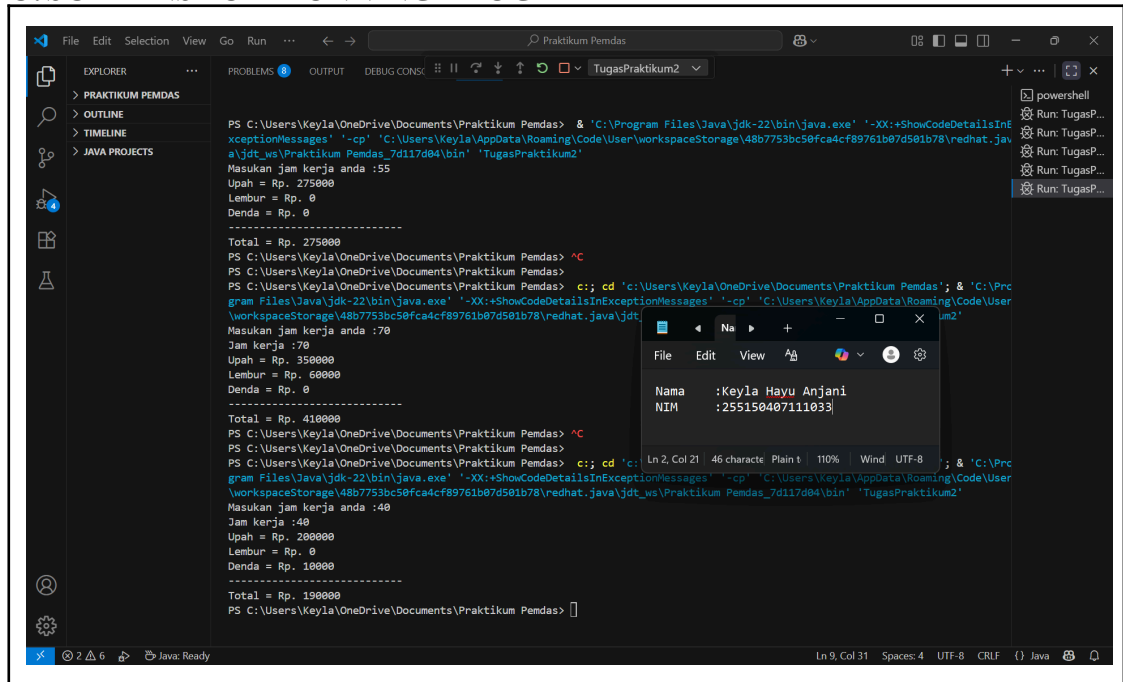
= Program menampilkan semua informasi yang relevan ke layar, yaitu jam kerja, upah, lembur, denda, dan total upah akhir yang diterima oleh pekerja.

## N. SCREENSHOT PROGRAM



Gambar 1.1 SS class mobil

## O.SCREENSHOT RUNNING PROGRAM



Gambar 1.1 SS class mobil