

LAPORAN PRAKTIKUM PEMROGRAMAN DASAR

BAB 2.5.5 Tugas Praktikum

Nama : Kusindra Aji Rabbany

Program Studi : Teknologi Informasi

Kelas : B

Asisten 1 : Dionisius Seraf Saputra

Asisten 2 : Malika Zahro Nadia Aulia

JAWABAN

1. A. Flowchart dan/atau Pseudocode

```
Mulai
  Tampilkan "Menu:"
  Tampilkan "1. menghitung luas dan keliling persegi panjang"
  Tampilkan "2. menghitung luas dan keliling lingkaran"
  Tampilkan "3. menghitung luas dan keliling segitiga"
  Tampilkan "\nPilihan anda: "
  Baca pilihan

  Jika pilihan == 1
    Tampilkan "Masukkan panjang: "
    Baca panjang
    Tampilkan "Masukkan lebar: "
    Baca lebar
    Hitung luas persegi panjang = panjang * lebar
    Hitung keliling persegi panjang = 2 * (panjang + lebar)
    Tampilkan keliling dan luas persegi panjang
  Jika pilihan == 2
    Tampilkan "Masukkan jari-jari: "
    Baca jari-jari
    Hitung luas lingkaran =  $\pi * \text{jari-jari} * \text{jari-jari}$ 
    Hitung keliling lingkaran =  $2 * \pi * \text{jari-jari}$ 
    Tampilkan keliling dan luas lingkaran
  Jika pilihan == 3
    Tampilkan "Masukkan a: "
    Baca sisi a
    Tampilkan "Masukkan b: "
    Baca sisi b
    Tampilkan "Masukkan c: "
    Baca sisi c
    Hitung keliling segitiga =  $a + b + c$ 
    Hitung s = keliling segitiga / 2
    Hitung luas segitiga =  $\sqrt{s * (s - a) * (s - b) * (s - c)}$ 
    Tampilkan keliling dan luas segitiga
  Jika tidak ada pilihan yang valid
```

Tampilkan "Menu tidak ada."

Tutup input

Selesai

B. Implementasi

```
1  import java.util.Scanner;
2
3  public class HitungBangun {
4      public static void main(String[] args) {
5          Scanner input = new Scanner(System.in);
6
7          System.out.println("Menu:");
8          System.out.println("1. menghitung luas dan keliling persegi panjang");
9          System.out.println("2. menghitung luas dan keliling lingkaran");
10         System.out.println("3. menghitung luas dan keliling segitiga");
11         System.out.print("\nPilihan anda: ");
12         int pilihan = input.nextInt();
13
14         switch (pilihan) {
15             case 1:
16                 System.out.print("Masukkan panjang: ");
17                 double p = input.nextDouble();
18                 System.out.print("Masukkan lebar: ");
19                 double l = input.nextDouble();
20                 double luasPP = p * l;
21                 double kelilingPP = 2 * (p + l);
22                 System.out.println("\nKeliling persegi panjang\t: " + kelilingPP + "
cm");
23                 System.out.println("Luas persegi panjang\t\t: " + luasPP + " cm^2");
24                 break;
25
26             case 2:
27                 System.out.print("Masukkan jari-jari: ");
28                 double r = input.nextDouble();
29                 double luasLingkaran = Math.PI * r * r;
30                 double kelilingLingkaran = 2 * Math.PI * r;
31                 System.out.println("\nKeliling lingkaran\t: " + kelilingLingkaran + "
cm");
32                 System.out.println("Luas lingkaran\t\t: " + luasLingkaran + " cm^2");
33                 break;
34
35             case 3:
36                 System.out.print("Masukkan a: ");
37                 double a = input.nextDouble();
38                 System.out.print("Masukkan b: ");
```

```

39         double b = input.nextDouble();
40         System.out.print("Masukkan r: ");
41         double c = input.nextDouble();
42         double kelilingSegitiga = a + b + c;
43         double s = kelilingSegitiga / 2;
44         double luasSegitiga = Math.sqrt(s * (s - a) * (s - b) * (s - c));
45         System.out.println("\nKeliling segitiga\t: " + kelilingSegitiga + "
cm");
46         System.out.println("Luas segitiga\t\t: " + luasSegitiga + " cm^2");
47         break;
48
49     default:
50         System.out.println("Menu tidak ada.");
51         break;
52     }
53
54     input.close();
55 }

```

C. Tampilan Keluaran (Screenshot)

```

• → benakusindras-macbook ~/kuliah/Semester 1/Pemrograman Dasar/Modul 2 java HitungBangun.java
Menu:
1. menghitung luas dan keliling persegi panjang
2. menghitung luas dan keliling lingkaran
3. menghitung luas dan keliling segitiga

Pilihan anda: 2
Masukkan jari-jari: 10

Keliling lingkaran      : 62.83185307179586 cm
Luas lingkaran          : 314.1592653589793 cm^2
• → benakusindras-macbook ~/kuliah/Semester 1/Pemrograman Dasar/Modul 2 java HitungBangun.java
Menu:
1. menghitung luas dan keliling persegi panjang
2. menghitung luas dan keliling lingkaran
3. menghitung luas dan keliling segitiga

Pilihan anda: 1
Masukkan panjang: 10
Masukkan lebar: 10

Keliling persegi panjang : 40.0 cm
Luas persegi panjang     : 100.0 cm^2

```

D. Pembahasan dan Analisis

Kode implementasi ini digunakan untuk menghitung luas dan keliling berbagai bentuk geometri, yaitu persegi panjang, lingkaran, dan segitiga. Data input untuk masing-masing bentuk diambil dari pengguna melalui perangkat input berupa keyboard, dan kemudian dilakukan perhitungan berdasarkan rumus matematika yang sesuai.

- **Baris ke-1:** Menggunakan kelas Scanner untuk menerima input dari pengguna. Scanner input = new Scanner(System.in); adalah langkah pertama untuk memulai interaksi dengan pengguna, di mana input dimasukkan secara langsung melalui konsol.

- **Baris ke-2 hingga ke-5:** Program menampilkan menu pilihan kepada pengguna, yang memungkinkan mereka memilih jenis bangun datar yang ingin dihitung luas dan kelilingnya. Menu ini menyediakan 3 opsi, yaitu:
 1. Menghitung luas dan keliling persegi panjang
 2. Menghitung luas dan keliling lingkaran
 3. Menghitung luas dan keliling segitiga
- **Baris ke-6:** Menggunakan `input.nextInt()` untuk membaca pilihan yang dimasukkan oleh pengguna. Pilihan ini kemudian akan menentukan bangun datar mana yang akan dihitung.

Analisis Perhitungan Masing-Masing Bangun Datar

Persegi Panjang:

- **Baris ke-9 hingga ke-11:** Program meminta pengguna untuk memasukkan panjang dan lebar persegi panjang.
- **Baris ke-12 hingga ke-13:** Luas dan keliling dihitung menggunakan rumus:
 - $\text{Luas} = \text{panjang} * \text{lebar}$
 - $\text{Keliling} = 2 * (\text{panjang} + \text{lebar})$
- **Baris ke-14 hingga ke-15:** Hasil perhitungan ditampilkan ke pengguna dalam format yang jelas.

Lingkaran:

- **Baris ke-17:** Program meminta pengguna untuk memasukkan jari-jari lingkaran.
- **Baris ke-18 hingga ke-19:** Luas dan keliling lingkaran dihitung menggunakan rumus:
 - $\text{Luas} = \pi * r^2$
 - $\text{Keliling} = 2 * \pi * r$
- **Baris ke-20 hingga ke-21:** Hasil perhitungan ditampilkan ke pengguna.

Segitiga:

- **Baris ke-23 hingga ke-25:** Program meminta pengguna untuk memasukkan panjang ketiga sisi segitiga (a, b, dan c).
- **Baris ke-26 hingga ke-27:** Keliling segitiga dihitung dengan menjumlahkan panjang ketiga sisi, dan rumus semiperimeter $s = (a + b + c) / 2$ digunakan untuk menghitung luas segitiga menggunakan rumus Heron:
 - $\text{Keliling} = a + b + c$
 - $\text{Luas} = \sqrt{s * (s - a) * (s - b) * (s - c)}$
- **Baris ke-28 hingga ke-29:** Hasil perhitungan keliling dan luas segitiga ditampilkan.

Analisis Kasus dan Pengaruh Terhadap Hasil

- **Jika Baris ke-3 Tidak Ada** (Misalnya, jika input pilihan tidak diberikan atau terjadi kesalahan dalam memilih opsi):
 - Program akan masuk ke bagian default dari switch-case, dan menampilkan pesan "Menu tidak ada." Hal ini menunjukkan bahwa input dari pengguna tidak valid, dan sistem akan menginformasikan pengguna untuk memilih dengan benar.

- **Jika Baris ke-4 Tidak Ada** (Misalnya, jika kode yang meminta input untuk panjang dan lebar persegi panjang atau untuk jari-jari lingkaran dihilangkan):
 - Jika data yang diperlukan untuk menghitung luas dan keliling tidak diberikan, maka hasil perhitungan tidak bisa dilakukan, yang berarti aplikasi akan memberikan pesan kesalahan atau tidak ada output sama sekali. Oleh karena itu, penting untuk selalu meminta data yang diperlukan sebelum melakukan perhitungan.

E. Refleksi Diri

Setelah mengerjakan ini, saya belajar menggunakan switch case bahasa pemrograman java dan menggunakannya sebagai percabangan dalam membuat menu.

2. A. Flowchart dan/atau Pseudocode

```

Mulai
Deklarasikan input untuk membaca data pengguna
Deklarasikan df (DecimalFormat) untuk format angka dengan dua angka desimal

Tampilkan "Masukkan berat badan (kg):"
Baca input berat badan sebagai b

Tampilkan "Masukkan tinggi badan (m):"
Baca input tinggi badan sebagai t

Hitung IMT = berat badan / (tinggi badan * tinggi badan)

Tentukan kategori kriteria berdasarkan nilai IMT:
  Jika IMT <= 18.5
    Set kriteria = "Kurus"
  Jika IMT > 18.5 dan IMT <= 25
    Set kriteria = "Normal"
  Jika IMT > 25 dan IMT <= 30
    Set kriteria = "Gemuk"
  Jika IMT > 30
    Set kriteria = "Kegemukan"

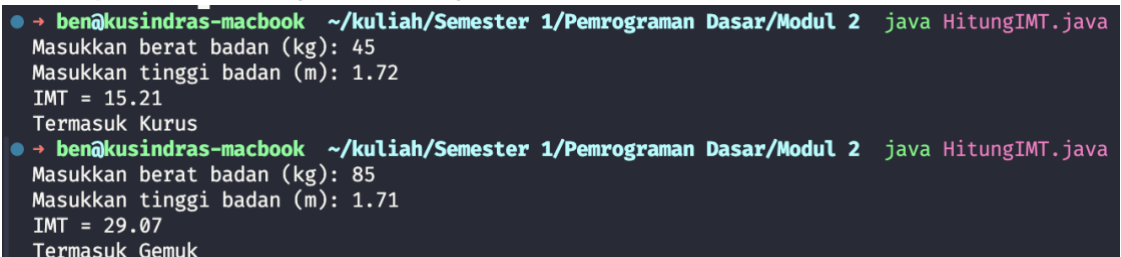
Tampilkan IMT dengan format dua angka desimal
Tampilkan "Termasuk" dan kategori kriteria

Tutup input
Selesai
  
```

B. Implementasi

```
1  import java.text.DecimalFormat;
2  import java.util.Scanner;
3
4  public class HitungIMT {
5      public static void main(String[] args) {
6          Scanner input = new Scanner(System.in);
7          DecimalFormat df = new DecimalFormat("#.00");
8
9          System.out.print("Masukkan berat badan (kg): ");
10         double b = input.nextDouble();
11         System.out.print("Masukkan tinggi badan (m): ");
12         double t = input.nextDouble();
13
14         double imt = b / (t * t);
15
16         String kriteria;
17         if (imt <= 18.5) {
18             kriteria = "Kurus";
19         } else if (imt <= 25) {
20             kriteria = "Nromal";
21         } else if (imt <= 30) {
22             kriteria = "Gemuk";
23         } else {
24             kriteria = "Kegemukan";
25         }
26
27         System.out.println("IMT = " + df.format(imt));
28         System.out.println("Termasuk " + kriteria);
29
30         input.close();
31     }
32 }
```

C. Tampilan Keluaran (Screenshot)



```
benakusindras-macbook ~/kuliah/Semester 1/Pemrograman Dasar/Modul 2 java HitungIMT.java
Masukkan berat badan (kg): 45
Masukkan tinggi badan (m): 1.72
IMT = 15.21
Termasuk Kurus
benakusindras-macbook ~/kuliah/Semester 1/Pemrograman Dasar/Modul 2 java HitungIMT.java
Masukkan berat badan (kg): 85
Masukkan tinggi badan (m): 1.71
IMT = 29.07
Termasuk Gemuk
```

D. Pembahasan dan Analisis

1. Deklarasi Scanner dan DecimalFormat:

- Pada baris pertama, `Scanner input = new Scanner(System.in);` digunakan untuk membaca input dari pengguna, yaitu berat badan dan tinggi badan.
- `DecimalFormat df = new DecimalFormat("#.00");` digunakan untuk memformat hasil IMT (Indeks Massa Tubuh) agar hanya menampilkan dua angka desimal.

2. Membaca Input:

- Program meminta pengguna untuk memasukkan berat badan dalam kilogram dan tinggi badan dalam meter.
- Input ini kemudian dibaca dan disimpan dalam variabel `b` (berat badan) dan `t` (tinggi badan).

3. Perhitungan IMT:

- IMT dihitung dengan rumus: $IMT = \text{berat badan} / \text{tinggi badan}^2$
- Hasil perhitungan IMT disimpan dalam variabel `imt`.

4. Menentukan Kriteria IMT:

- Program kemudian menentukan kategori IMT berdasarkan rentang nilai IMT:
 - Jika $IMT \leq 18.5$, maka kriteria adalah "Kurus".
 - Jika $18.5 < IMT \leq 25$, maka kriteria adalah "Normal".
 - Jika $25 < IMT \leq 30$, maka kriteria adalah "Gemuk".
 - Jika $IMT > 30$, maka kriteria adalah "Kegemukan".

5. Menampilkan Hasil:

- Hasil IMT ditampilkan dengan dua angka desimal menggunakan `df.format(imt)`.
- Kriteria IMT yang sesuai juga ditampilkan kepada pengguna.

6. Menutup Scanner:

- Program menutup scanner untuk mencegah kebocoran memori.

Analisis:

• Keunggulan:

- Program ini mudah digunakan untuk menghitung IMT berdasarkan berat badan dan tinggi badan pengguna. Pengguna hanya perlu memasukkan dua data, dan hasil IMT serta kategorinya langsung ditampilkan.

• Pentingnya Format:

- Format angka desimal yang ditampilkan menggunakan `DecimalFormat` memastikan hasil IMT dibaca dengan tepat, yakni dua angka desimal.

• Validasi Input:

- Program ini belum melakukan validasi untuk memastikan bahwa input yang diberikan valid (misalnya, tidak negatif atau nol). Dalam pengembangan lebih lanjut, sebaiknya ditambahkan validasi untuk input yang lebih baik.

E. Refleksi Diri

Setelah mengerjakan ini, saya dapat memahami sistem penghitungan Body Mass Index (BMI) dan dapat mengimplementasikannya dengan coding menggunakan bahasa pemrograman java.

3. A. Flowchart dan/atau Pseudocode

```
Mulai
Deklarasikan input untuk membaca data pengguna
Deklarasikan variabel upah per jam = 5000, upah lembur = 6000, denda per jam = 1000

Tampilkan "Jam kerja : "
Baca input jamKerja

Inisialisasi variabel upah = 0, lembur = 0, denda = 0, total = 0

Jika jamKerja > 60
```

```

    upah = 60 * upahPerJam
    lembur = (jamKerja - 60) * upahLembur
    denda = 0
    Jika tidak, jika jamKerja >= 50
        upah = jamKerja * upahPerJam
        lembur = 0
        denda = 0
    Jika tidak
        upah = jamKerja * upahPerJam
        lembur = 0
        denda = (50 - jamKerja) * dendaPerJam

total = upah + lembur - denda

Tampilkan "Upah = Rp. " + upah
Tampilkan "Lembur = Rp. " + lembur
Tampilkan "Denda = Rp. " + denda
Tampilkan "-----"
Tampilkan "Total = Rp. " + total

Tutup input
Selesai

```

B. Implementasi

```

1  import java.util.Scanner;
2
3  public class Penggajian {
4      public static void main(String[] args) {
5          Scanner input = new Scanner(System.in);
6
7          System.out.print("Jam kerja : ");
8          int jamKerja = input.nextInt();
9
10         int upahPerJam = 5000;
11         int upahLembur = 6000;
12         int dendaPerJam = 1000;
13
14         int upah = 0;
15         int lembur = 0;
16         int denda = 0;
17         int total = 0;
18
19         if (jamKerja > 60) {
20             upah = 60 * upahPerJam;
21             lembur = (jamKerja - 60) * upahLembur;
22             denda = 0;
23         } else if (jamKerja >= 50) {
24             upah = jamKerja * upahPerJam;
25             lembur = 0;
26             denda = 0;
27         } else {

```



```

28     upah = jamKerja * upahPerJam;
29     lembur = 0;
30     denda = (50 - jamKerja) * dendaPerJam;
31 }
32
33     total = upah + lembur - denda;
34
35     System.out.println("Upah   = Rp. " + upah);
36     System.out.println("Lembur = Rp. " + lembur);
37     System.out.println("Denda  = Rp. " + denda);
38     System.out.println("-----");
39     System.out.println("Total  = Rp. " + total);
40
41     input.close();
42 }
43 }

```

C. Tampilan Keluaran (Screenshot)

```

→ ben@kusindras-macbook ~/kuliah/Semester 1/Pemrograman Dasar/Modul 2 java Penggajian.java
Jam kerja : 55
Upah   = Rp. 275000
Lembur = Rp. 0
Denda  = Rp. 0
-----
Total  = Rp. 275000
→ ben@kusindras-macbook ~/kuliah/Semester 1/Pemrograman Dasar/Modul 2 java Penggajian.java
Jam kerja : 70
Upah   = Rp. 300000
Lembur = Rp. 60000
Denda  = Rp. 0
-----
Total  = Rp. 360000
→ ben@kusindras-macbook ~/kuliah/Semester 1/Pemrograman Dasar/Modul 2 java Penggajian.java
Jam kerja : 40
Upah   = Rp. 200000
Lembur = Rp. 0
Denda  = Rp. 10000
-----
Total  = Rp. 190000

```

D. Pembahasan dan Analisis (Singkat & Padat)

Program ini digunakan untuk menghitung total penghasilan pekerja berdasarkan jam kerja, dengan memperhitungkan **upah**, **lembur**, dan **denda**.

- **Baris 1:** Membuat scanner untuk membaca input pengguna.
- **Baris 2:** Meminta pengguna memasukkan jumlah jam kerja dan menyimpannya di jamKerja.
- **Baris 3:** Menetapkan tarif upah per jam, lembur, dan denda, serta inisialisasi variabel upah, lembur, denda, dan total.
- **Baris 4:** Menghitung upah, lembur, dan denda berdasarkan jam kerja:
 - Jika jam > 60: upah = 60 jam normal, sisanya lembur, denda = 0.
 - Jika jam 50–60: upah = jam kerja normal, lembur = 0, denda = 0.
 - Jika jam < 50: upah = jam kerja normal, lembur = 0, denda = kekurangan jam × tarif denda.
- **Baris 5:** Total gaji = upah + lembur – denda.
- **Baris 6:** Menampilkan upah, lembur, denda, dan total ke layar.

- **Baris 7:** Menutup scanner untuk mencegah kebocoran memori.
- Program menghitung upah sesuai jam kerja, lembur hanya untuk jam > 60, denda hanya untuk jam < 50.
- Untuk lebih aman, bisa ditambahkan validasi input negatif atau tidak logis.

E. Refleksi Diri

Setelah mengerjakan ini, saya dapat mengetahui sistem perhitungan bertingkat dan mengimplementasikannya dalam bahasa pemrograman java.

Kesimpulan

Dari percobaan ketiga program ini, dapat disimpulkan:

1. **Tujuan percobaan tercapai:** Semua program berhasil menghitung nilai sesuai tujuan masing-masing, yaitu:
 - Hitung Bangun → menghitung luas dan keliling bangun datar.
 - Hitung IMT → menghitung Indeks Massa Tubuh dan mengkategorikannya.
 - Penggajian → menghitung total gaji berdasarkan jam kerja, termasuk upah, lembur, dan denda.
2. **Hubungan dengan dasar teori:**
 - Program menggunakan rumus matematika dan logika percabangan yang sesuai dengan teori:
 - Luas dan keliling bangun datar berdasarkan rumus geometris.
 - IMT dihitung dengan berat / tinggi² dan dikategorikan sesuai standar kesehatan.
 - Upah, lembur, dan denda dihitung berdasarkan aturan jam kerja dan tarif per jam.
3. **Faktor yang memengaruhi hasil percobaan:**
 - Input yang diberikan pengguna sangat menentukan hasil akhir.
 - Kesalahan memasukkan data (misal jam kerja negatif, tinggi atau berat 0) dapat menghasilkan output yang tidak valid.

Program-program ini berjalan sesuai tujuan dan teori, hasil dapat diprediksi dengan benar, dan akurasi bergantung pada input yang valid.