

LAPORAN PRAKTIKUM
Algoritma Pemrograman

MODUL 5 & 6
FOR-LOOP



Disusun oleh:

Nama:Nafisah Salsabila

NIM:109082500063

S1IF-13-04

PROGRAM STUDI S1 INFORMATIKA
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO

2025

LATIHAN KELAS – GUIDED

1. Guided 1

Source Code

```
package main

import "fmt"

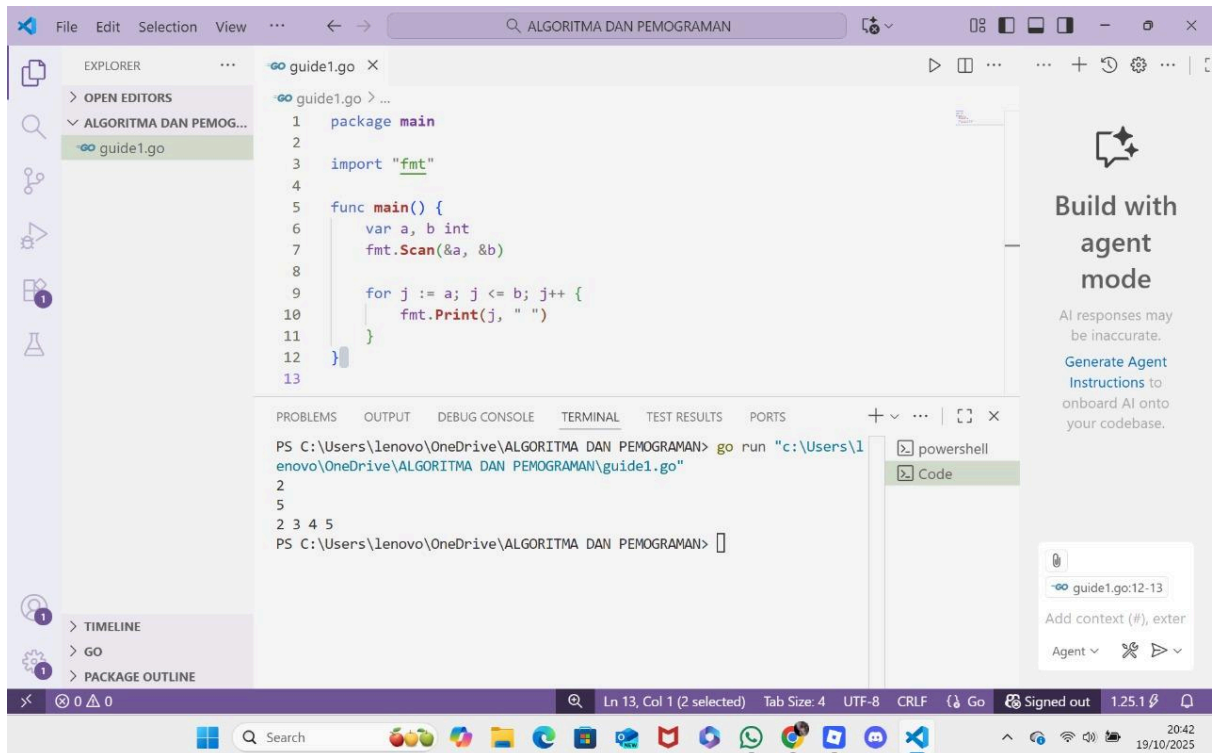
func main() {
    var a, b int
    fmt.Scan(&a, &b)

    for j := a; j <= b; j++ {
        fmt.Print(j, " ")
    }
}
```

Input: 2 5

Output: 2 3 4 5

Screenshoot program



Deskripsi program

Program ini digunakan untuk menampilkan deretan bilangan bulat dari nilai a sampai dengan b .

Pengguna memasukkan dua bilangan bulat a dan b dengan syarat $a \leq b$.

Program akan menampilkan semua bilangan mulai dari a hingga b secara berurutan dengan pemisah spasi.

Perulangan `for` digunakan untuk menambah nilai dari a hingga mencapai b , dan setiap nilai dicetak di layar menggunakan perintah `fmt.Print()`.

2. Guided 2

Source Code

```
package main

import "fmt"

func main() {

    var j, alas, tinggi, n int
    var luas float64

    fmt.Scan(&n)

    for j = 1; j <= n; j++ {
```

```

        fmt.Scan(&alas, &tinggi)

        luas = 0.5 * float64(alas * tinggi)

        fmt.Println(luas)

    }

}

```

Input:

```

3

12 32

231 234

43 34

```

Output:

```

192

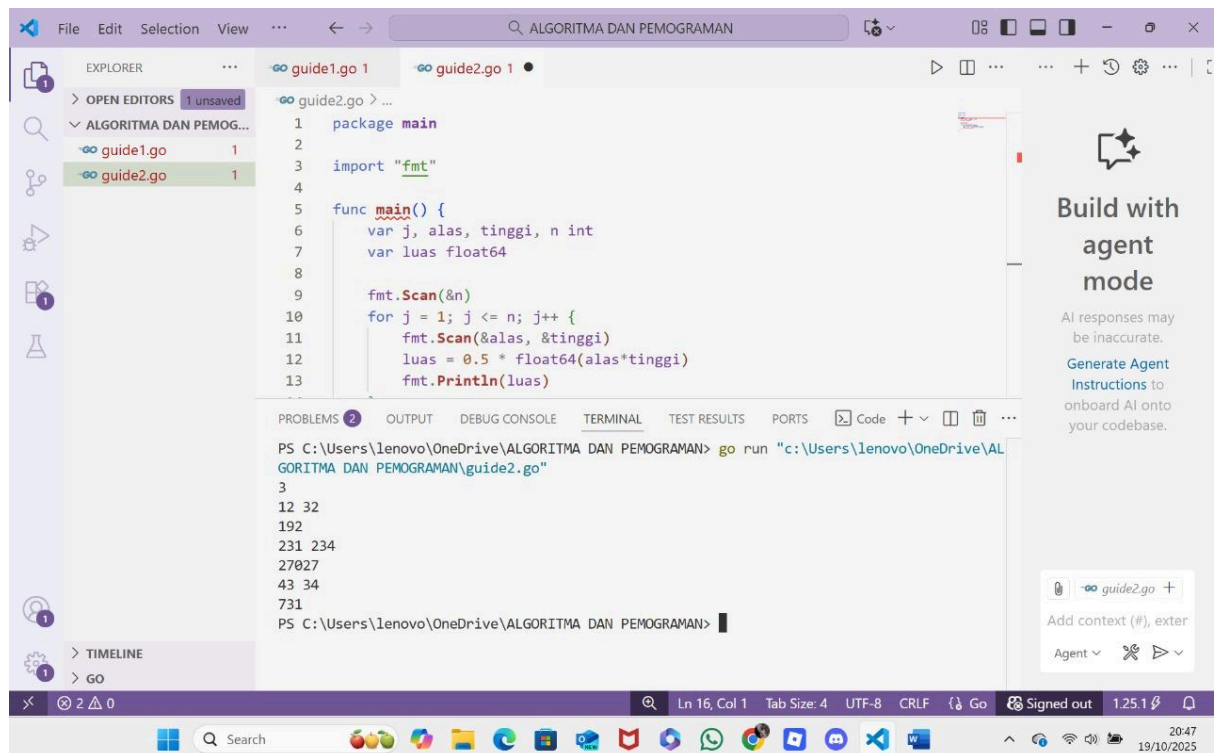
27027

731

I

```

Screenshoot program



Deskripsi program

Program ini berfungsi untuk menghitung dan menampilkan luas beberapa segitiga berdasarkan data alas dan tinggi yang dimasukkan oleh pengguna.

Masukan pertama berupa bilangan bulat n yang menunjukkan banyaknya segitiga.

Kemudian pengguna memasukkan n pasang nilai alas dan tinggi.

Untuk setiap pasangan, program menghitung luas segitiga menggunakan rumus:

$$L = 0.5 \times \text{alas} \times \text{tinggi}$$

Perulangan for digunakan untuk memproses data setiap segitiga.

3. Guided 3

Source Code

```
package main

import "fmt"

func main() {
    var j, v1, v2 int
    var hasil int

    fmt.Scan(&v1, &v2)
    hasil = 0

    for j = 1; j <= v2; j++ {
        hasil = hasil + v1
    }

    fmt.Println(hasil)
}
```

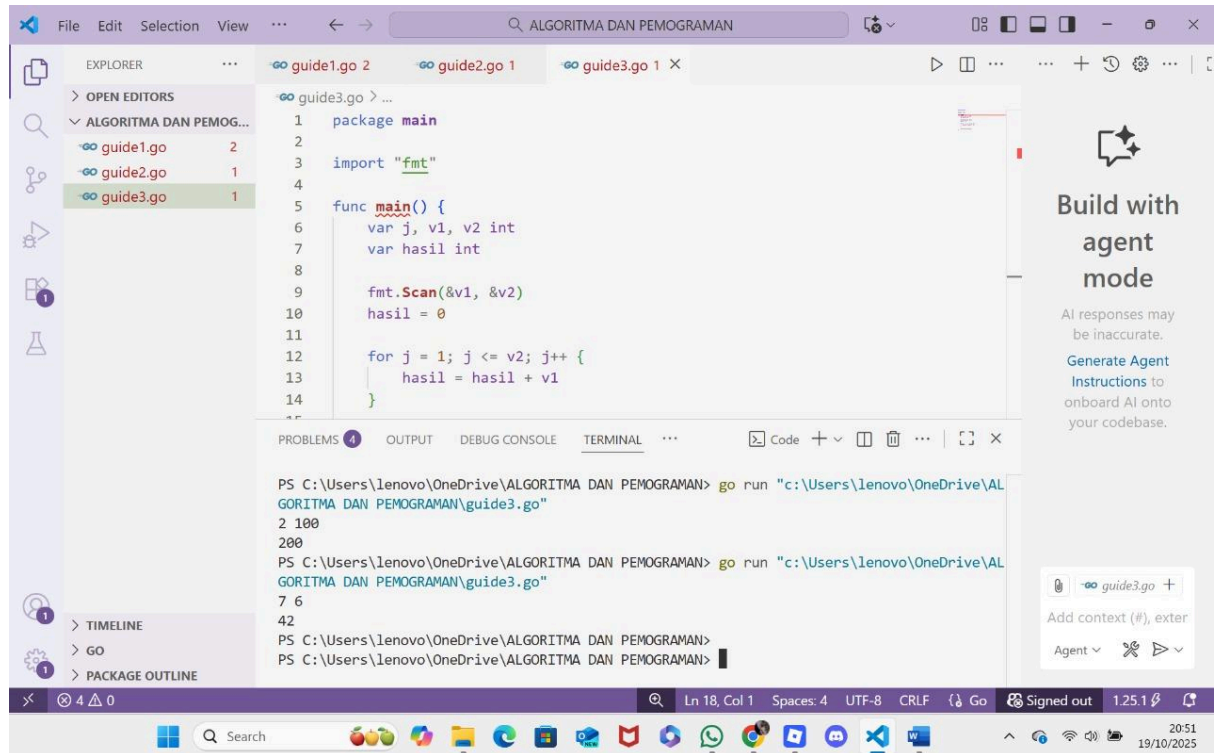
Input: 2 100

Output: 200

Input: 7 6

Output: 42

Screenshoot program



Deskripsi program

Program ini digunakan untuk menghitung hasil perkalian dua bilangan bulat positif tanpa menggunakan operator kali (*).

Program membaca dua bilangan dari input, lalu melakukan proses penjumlahan berulang untuk mendapatkan hasil perkalian.

Nilai pertama (v1) akan dijumlahkan sebanyak nilai kedua (v2) kali menggunakan perulangan for.

Hasil akhir penjumlahan tersebut merupakan hasil perkalian dari kedua bilangan yang dimasukkan.

TUGAS

1. Tugas 1

Source code

```
package main

import "fmt"
```

```

func main() {

    var n, hasil int

    fmt.Scan(&n)

    hasil = 0

    for i := 1; i <= n; i++ {

        hasil = hasil + i

    }

    fmt.Println(hasil)

}

```

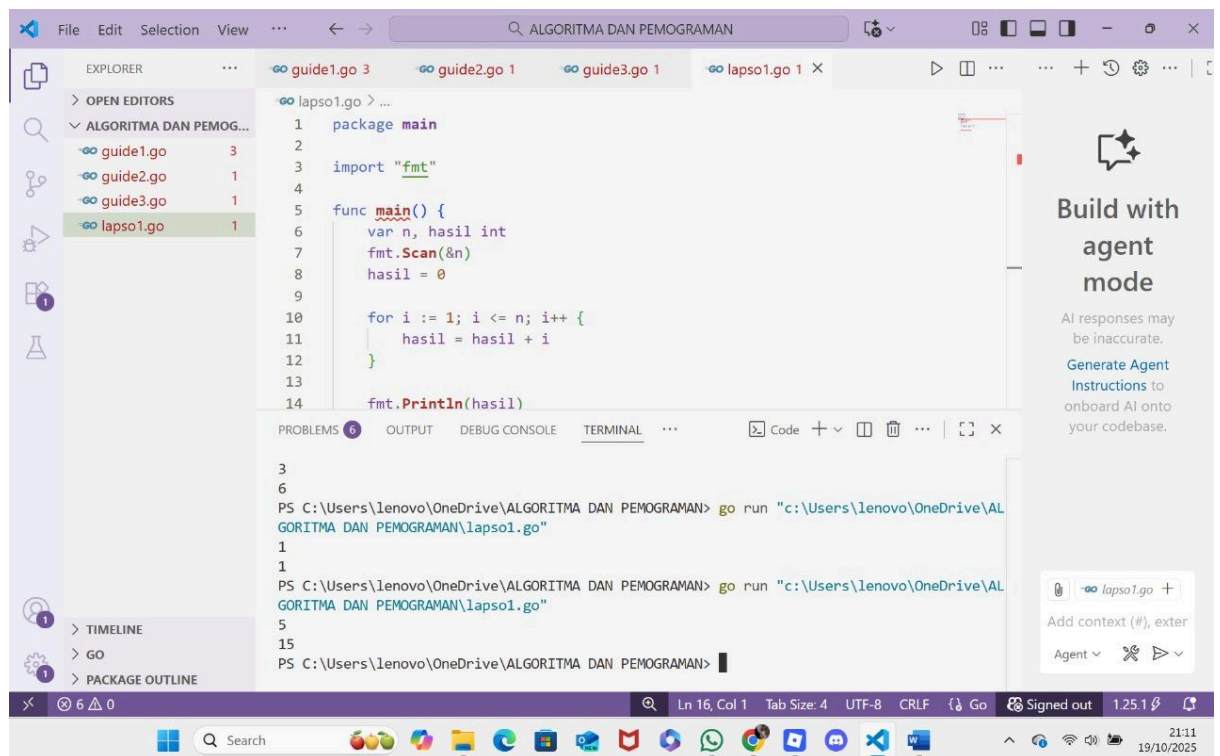
Input:

3 1 5

Output:

6 1 15

Screenshoot program



Deskripsi program

Program ini digunakan untuk menghitung jumlah bilangan dari 1 sampai n. Program membaca input berupa bilangan bulat positif n. Variabel hasil diinisialisasi dengan nilai 0. Menggunakan perulangan for dari 1 hingga n, setiap nilai i dijumlahkan ke hasil. Setelah perulangan

2. Tugas 2

Source code

```
package main
import (
    "fmt"
    "math"
)

func main() {
    var n, jari, tinggi int
    var volume float64

    fmt.Scan(&n)
    for i := 1; i <= n; i++ {
        fmt.Scan(&jari, &tinggi)
        volume = (1.0 / 3.0) * math.Pi * math.Pow(float64(jari), 2) *
float64(tinggi)
        fmt.Println(volume)
    }
}
```

Input:

1
3 4

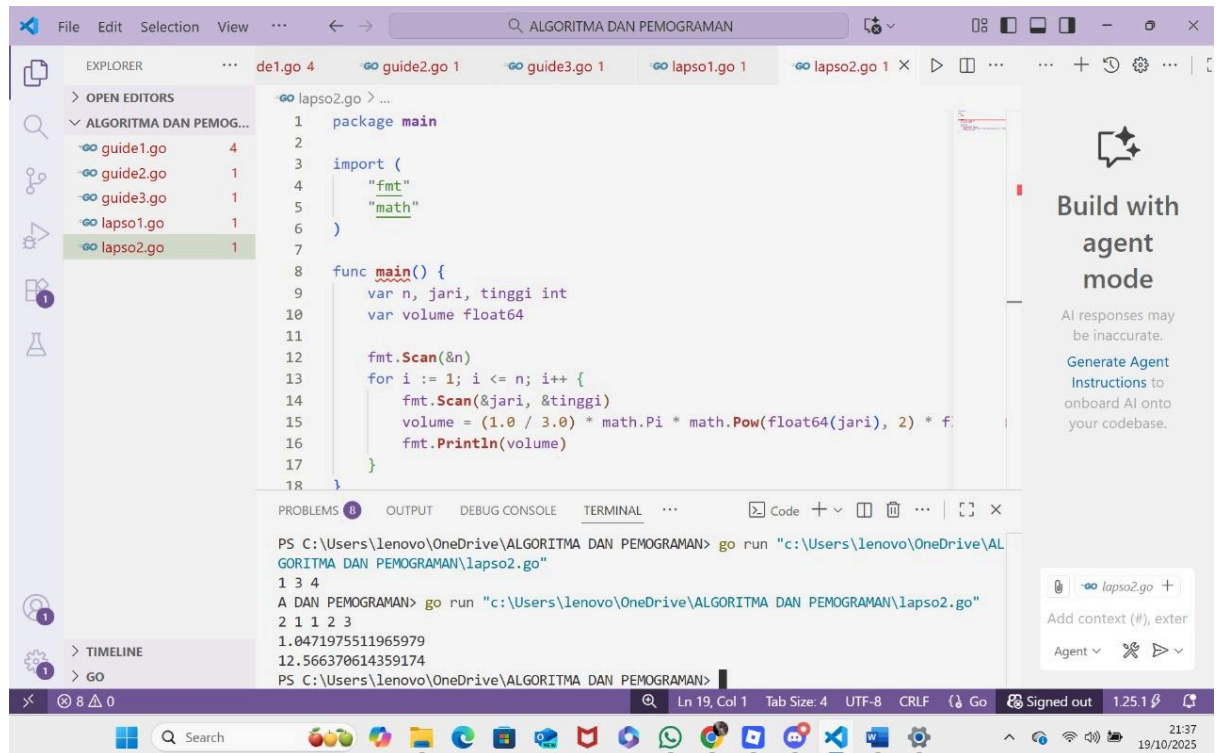
3
1 1
2 2
3 3

Output:

37.69911184307752

1.0471975511965976
28.274333882308138

Screenshoot program



Deskripsi program

Program ini menghitung volume beberapa kerucut menggunakan rumus:

$$V = \frac{1}{3} \pi r^2 t$$

Input n menunjukkan berapa banyak data kerucut yang akan dihitung. Untuk setiap kerucut, input jari (r) dan tinggi (t). Rumus volume dihitung menggunakan fungsi `math.Pow()` dan konstanta `math.Pi`. Volume tiap kerucut ditampilkan di baris terpisah.

3. Tugas 3

Source code

```
package main
import "fmt"

func main() {
    var a, b, hasil int
    fmt.Scan(&a, &b)

    hasil = 1
    for i := 1; i <= b; i++ {
        hasil = hasil * a
    }

    fmt.Println(hasil)
```

```
}
```

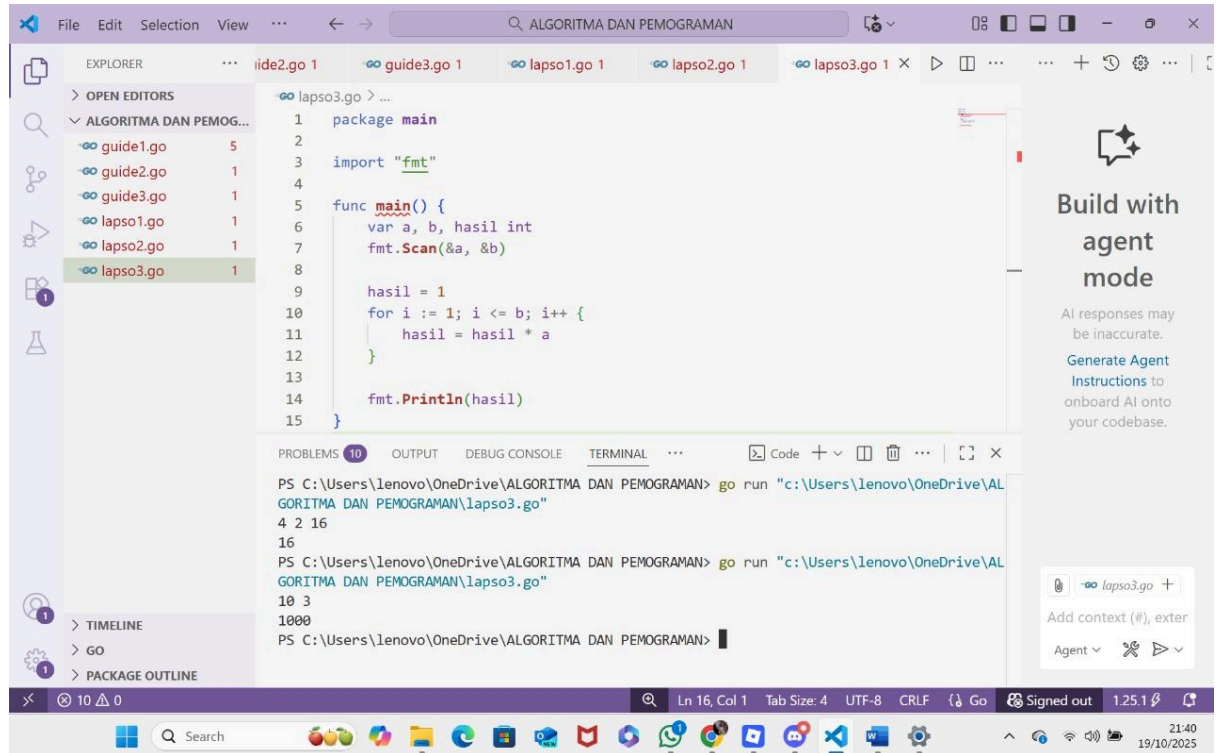
Input:

4 2 16

Output:

10 3 1000

Screenshoot program



```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var a, b, hasil int
7     fmt.Scan(&a, &b)
8
9     hasil = 1
10    for i := 1; i <= b; i++ {
11        hasil = hasil * a
12    }
13
14    fmt.Println(hasil)
15 }
```

PS C:\Users\lenovo\OneDrive\ALGORITMA DAN PEMOGGRAMAN> go run "c:\Users\lenovo\OneDrive\ALGORITMA DAN PEMOGGRAMAN\lapso3.go"

4 2 16

10 3

1000

PS C:\Users\lenovo\OneDrive\ALGORITMA DAN PEMOGGRAMAN>

Deskripsi program

Program ini menghitung pemangkatan bilangan (misalnya) menggunakan perulangan, bukan fungsi siap pakai. Input dua bilangan bulat positif, a (basis) dan b (pangkat). Inisialisasi hasil = 1. Ulangi b kali, setiap kali kalikan hasil dengan a. Setelah selesai, tampilkan hasil akhir.

Tugas Pendahuluan

Soal:

1. Jelaskan perbedaan `fmt.Print()`, `fmt.Println()`, dan `fmt.Printf()` di Go!
2. Jelaskan penggunaan tipe data `int`, `float64`, `bool`, dan `string` di Go!
3. Bagaimana Cara mendeklarasikan variabel dengan kata kunci `var` dan dengan acara singkat? Berikan contoh! Hint `":="`
4. Apa perbedaan antara operator `==` dan `=` dalam bahasa Go?
5. Buatlah program Go sederhana untuk meminta input nama kalian, lalu menampilkan nama.

Jawaban:

1.

- `fmt.Print()` adalah untuk mencetak teks ke layar tanpa menambah baris baru.

→ Contoh:

Go:

```
fmt.Print("Halo ")  
fmt.Print("Billa")
```

Output:

Halo Billa

- `fmt.Println()` → mencetak teks lalu otomatis pindah baris setelahnya.

→ Contoh:

Go:

```
fmt.Println("Halo")
```

```
fmt.Println("Billa")
```

Output:

```
Halo  
Billa
```

- `fmt.Println()` adalah mencetak teks lalu otomatis pindah baris setelahnya.

→ Contoh:

Go:

```
fmt.Printf("Nilai saya: %.2f", 90.555)
```

Output:

```
Nilai saya: 90.56
```

2. `int` → untuk bilangan bulat, misalnya 10, -7, 0.

`float64` → untuk bilangan desimal, misalnya 3.14, 5.0.

`bool` → untuk nilai benar/salah, yaitu `true` atau `false`.

`string` → untuk teks, misalnya "Halo Dunia".

Contoh:

```
var umur int = 18  
var tinggi float64 = 168.5  
var sehat bool = true  
var nama string = "Billa"
```

3. Dengan var:

```
var nama string = "Billa"  
var umur int = 18
```

Dengan cara singkat (:=):

```
nama := "Billa"  
umur := 18
```

Ini otomatis menentukan tipe data berdasarkan nilainya (string, int, dll).

4. = untuk memberikan nilai (assignment) ke variabel.

Contoh: x = 5

== untuk membandingkan dua nilai.

Contoh: x == 5 (hasilnya true kalau x memang 5)

5. Program sederhana untuk memasukkan nama:

```
package main  
import "fmt"  
  
func main() {  
    var nama string  
    fmt.Print("Masukkan nama kamu: ")  
    fmt.Scan(&nama)  
    fmt.Println("Halo,", nama, "selamat datang di Golang!")  
}
```

Output:

Masukkan nama kamu: Billa

Halo, Billa selamat datang di Golang