

LAPORAN PRAKTIKUM
Algoritma Pemrograman

MODUL 2
I/O, TIPE DATA & VARIABEL



Disusun oleh:
EDWARD ABIMAS SURYA HATTA
109082500171
S1IF-13-04

PROGRAM STUDI S1 INFORMATIKA
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO
2025

LATIHAN KELAS – GUIDED

1. Guided 1

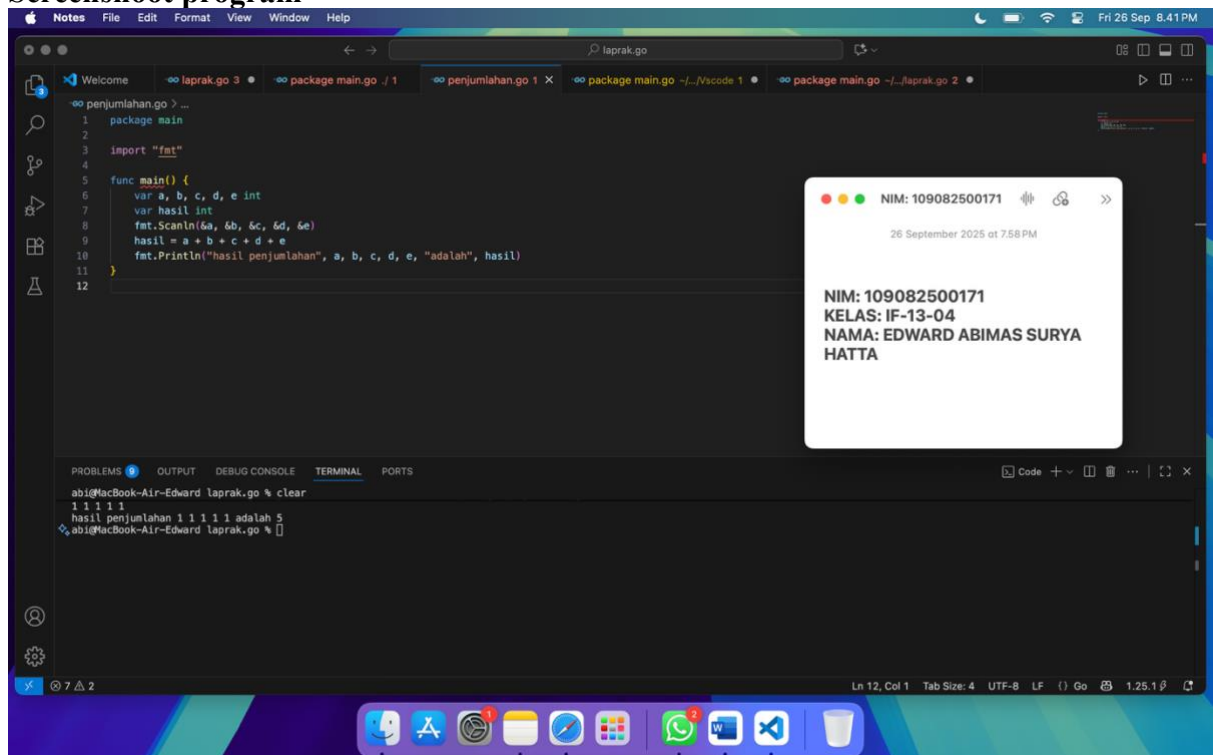
Source Code

```
package main

import "fmt"

func main() {
    var a, b, c, d, e int
    var hasil int
    fmt.Scanln(&a, &b, &c, &d, &e)
    hasil = a + b + c + d + e
    fmt.Println("hasil penjumlahan", a, b, c, d, e, "adalah", hasil)
}
```

Screenshoot program



Deskripsi program

- package main: Mendeklarasikan bahwa ini adalah program utama yang dapat dieksekusi secara langsung
- import "fmt": Mengimpor package fmt (format) untuk operasi input/output standar seperti membaca dari keyboard dan menulis ke layar

- `var a, b, c, d, e int`: Mendeklarasikan 5 variabel (a, b, c, d, e) dengan tipe data integer secara bersamaan. Setiap variabel akan diinisialisasi dengan nilai default 0
- `var hasil int`: Mendeklarasikan variabel hasil untuk menyimpan total penjumlahan
- `fmt.Scanln(&a, &b, &c, &d, &e)`: Membaca input dari pengguna dalam satu baris, dipisahkan oleh spasi. Operator &(ampersand) digunakan sebagai pointer untuk memberikan alamat memori variabel kepada fungsi `Scanln` sehingga nilai dapat disimpan langsung ke variabel tersebut
- `hasil = a + b + c + d + e`: Melakukan operasi aritmatika penjumlahan dari kelima variabel dan menyimpan hasilnya ke variabel hasil
- `fmt.Printf("hasil penjumlahan", a, b, c, d, e, "adalah", hasil)`: Menampilkan hasil ke layar. Namun ada **kesalahan sintaks** di sini karena tidak menggunakan format specifier yang benar
- `}`: Menutup fungsi main

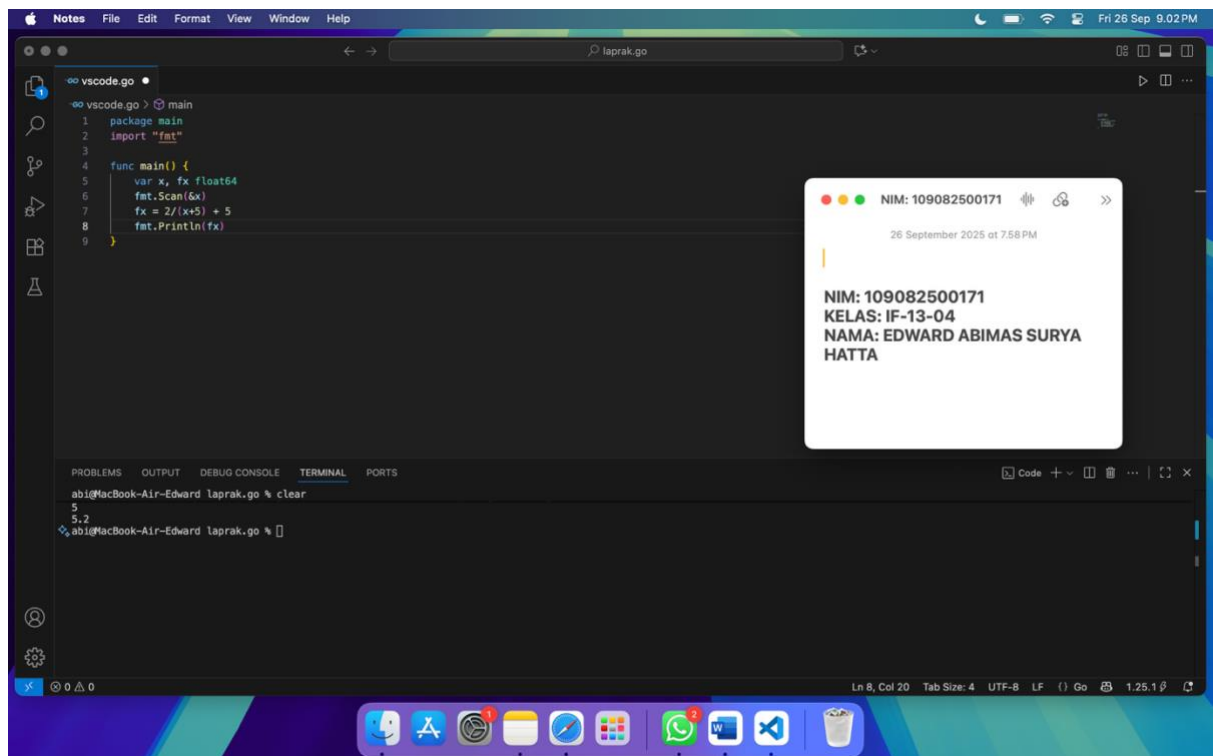
2. Guided 2

Source Code

```
package main
import "fmt"

func main() {
    var x, fx float64
    fmt.Scan(&x)
    fx = 2/(x+5) + 5
    fmt.Println(fx)
}
```

Screenshoot program



Deskripsi program

- package main - Mendeklarasikan bahwa ini adalah program utama yang dapat dieksekusi
- import "fmt" - Mengimpor package fmt untuk operasi input/output
- func main() { - Fungsi utama yang akan dijalankan pertama kali
- var x, fx float64 - Mendeklarasikan dua variabel x dan fx dengan tipe data float64 (bilangan desimal)
- fmt.Scan(&x) - Membaca input dari user dan menyimpannya ke variabel x (menggunakan pointer &x)
- $fx = 2/(x+5) + 5$ - Melakukan perhitungan matematis: $fx = 2 \div (x+5) + 5$
- fmt.Println(fx) - Menampilkan hasil perhitungan fx ke layar dengan baris baru
- } - Menutup fungsi main

3. Guided 3

Source Code

```
package main

import "fmt"

func main() {
    var c1, c2, c3, c4, c5 byte
    var b1, b2, b3 int
}
```

```

fmt.Scan(&c1, &c2, &c3, &c4, &c5)

fmt.Scanf("%c", &b1)

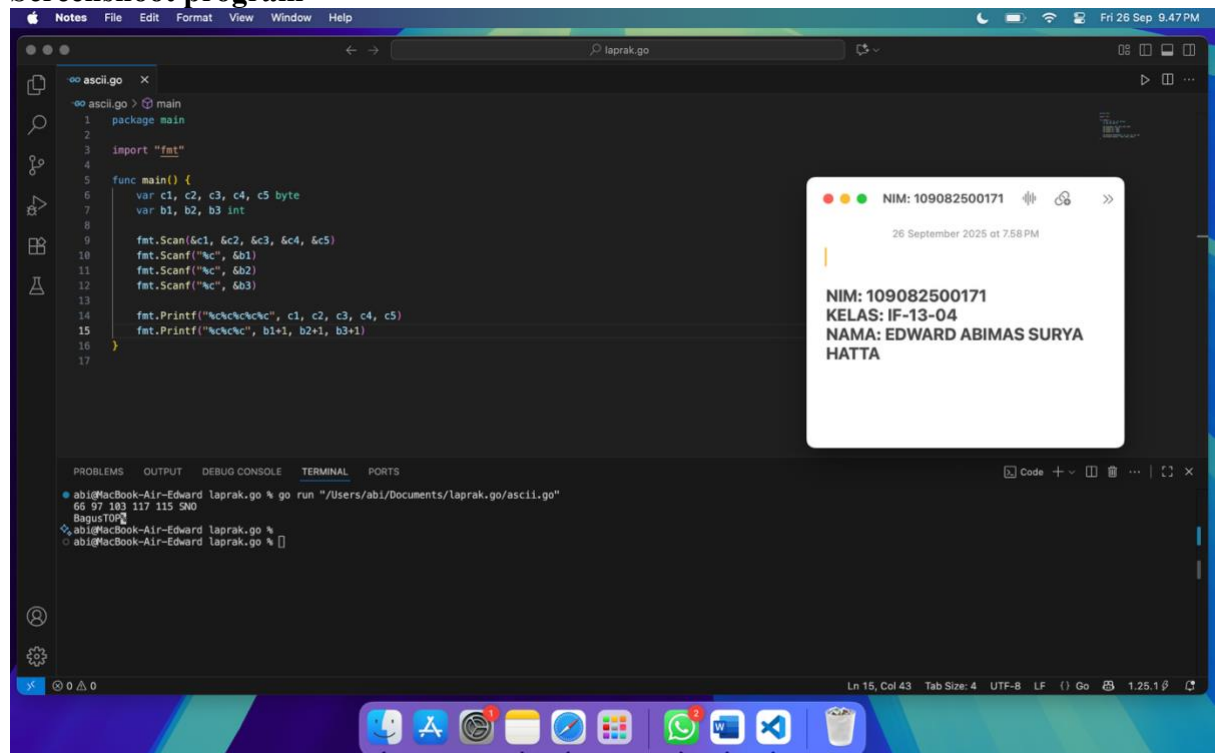
fmt.Scanf("%c", &b2)

fmt.Scanf("%c", &b3)


fmt.Printf("%c%c%c%c%c", c1, c2, c3, c4, c5)
fmt.Printf("%c%c%c", b1+1, b2+1, b3+1)
}

```

Screenshoot program



Deskripsi program

- package main - Deklarasi package utama
- import "fmt" - Import package untuk I/O
- func main() { - Fungsi utama
- var c1, c2, c3, c4, c5 byte - Deklarasi 5 variabel bertipe byte (untuk menyimpan karakter)
- var b1, b2, b3 int - Deklarasi 3 variabel bertipe int
- fmt.Scan(&c1, &c2, &c3, &c4, &c5) - Membaca 5 input sekaligus dan menyimpan ke c1-c5
- fmt.Scanf("%c", &b1) - Membaca 1 karakter dengan format %c ke b1
- fmt.Scanf("%c", &b2) - Membaca 1 karakter dengan format %c ke b2
- fmt.Scanf("%c", &b3) - Membaca 1 karakter dengan format %c ke b3

- `fmt.Printf("%c%c%c%c%c", c1, c2, c3, c4, c5)` - Menampilkan c1-c5 sebagai karakter
- `fmt.Printf("%c%c", b1+1, b2+1, b3+1)` - Menampilkan b1-b3 yang sudah ditambah 1 sebagai karakter
- `}` - Menutup fungsi main

TUGAS

1. Tugas 1

Source code

```
package main

import "fmt"

func main() {
    var (
        satu, dua, tiga string
        temp      string
    )

    fmt.Print("Masukkan input String: ")
    fmt.Scan(&satu)
    fmt.Print("Masukkan Input String: ")
    fmt.Scan(&dua)
    fmt.Print("Masukkan Input String: ")
    fmt.Scan(&tiga)
    fmt.Println("output awal = " + satu + " " + dua + " " + tiga)
    temp = satu
    satu = dua
    dua = tiga
    tiga = temp

    fmt.Println("Output akhir = " + satu + " " + dua + " " + tiga)
}
```

Screenshoot program

```
5 func main() {
6     var (
7     )
8
9     fmt.Print("Masukkan Input String: ")
10    fmt.Scan(&satu)
11    fmt.Print("Masukkan Input String: ")
12    fmt.Scan(&dua)
13    fmt.Print("Masukkan Input String: ")
14    fmt.Scan(&tiga)
15    fmt.Println("output awal = " + satu + " " + dua + " " + tiga)
16    temp = satu
17    satu = dua
18    dua = tiga
19    tiga = temp
20
21    fmt.Println("Output akhir = " + satu + " " + dua + " " + tiga)
22 }
23
24
```

abir@MacBook-Air-Edward laprak.go % CLEAR
Masukkan Input String: a
Masukkan Input String: b
Masukkan Input String: c
output awal = a b c
Output akhir = b c a
abir@MacBook-Air-Edward laprak.go %

Deskripsi program

- package main - Deklarasi package utama
- import "fmt" - Import package fmt
- func main() { - Fungsi utama
- var (- Mulai blok deklarasi variabel berkelompok
- satu, dua, tiga string - Deklarasi 3 variabel string
- temp string - Variabel temporary untuk proses swap
-) - Akhir blok deklarasi
- fmt.Print("Masukkan input String: ") - Menampilkan prompt untuk input pertama
- fmt.Scan(&satu) - Membaca input ke variabel satu
- Baris 10-13: Mengulangi proses input untuk variabel dua dan tiga
- fmt.Println("output awal = " + satu + " " + dua + " " + tiga) - Menampilkan nilai awal
- temp = satu - Menyimpan nilai satu ke temp
- satu = dua - Memindahkan nilai dua ke satu
- dua = tiga - Memindahkan nilai tiga ke dua
- tiga = temp - Memindahkan nilai temp (asal satu) ke tiga
- fmt.Println("Output akhir = " + satu + " " + dua + " " + tiga) - Menampilkan hasil akhir
- } - Menutup fungsi main

2. Tugas 2

Source code

```
package main

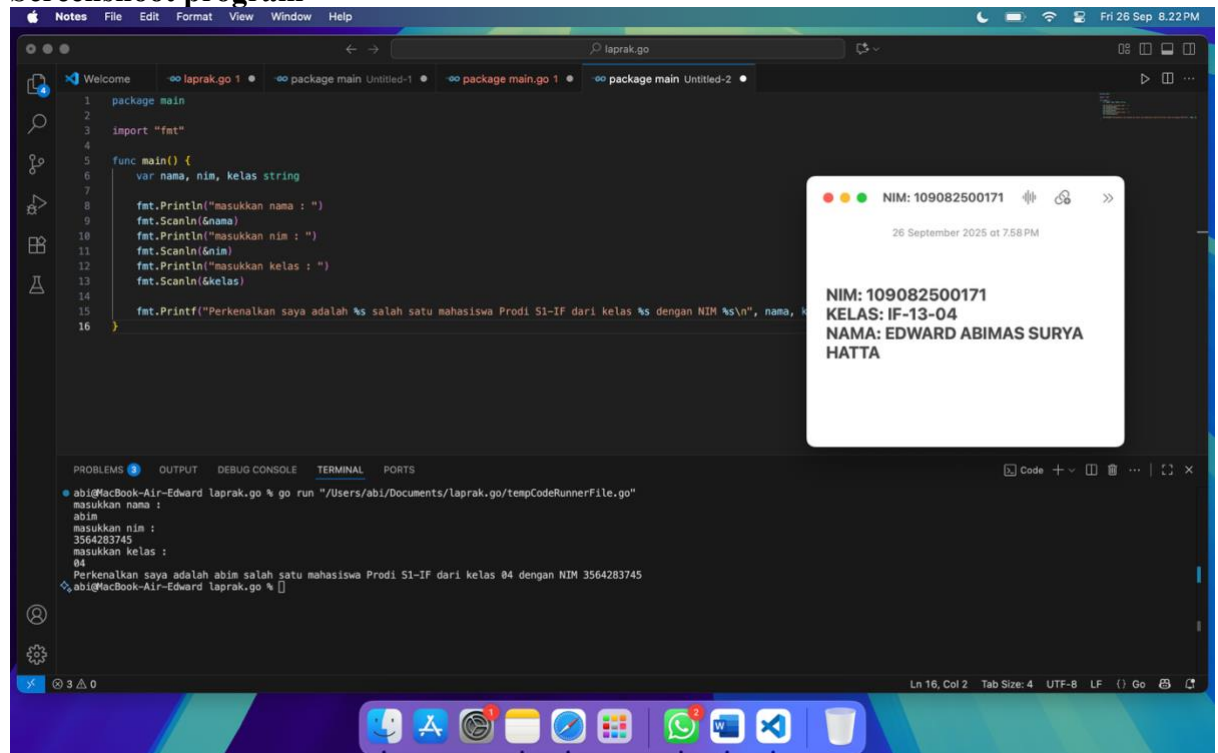
import "fmt"

func main() {
    var nama, nim, kelas string

    fmt.Println("masukkan nama : ")
    fmt.Scanln(&nama)
    fmt.Println("masukkan nim : ")
    fmt.Scanln(&nim)
    fmt.Println("masukkan kelas : ")
    fmt.Scanln(&kelas)

    fmt.Printf("Perkenalkan saya adalah %s salah satu mahasiswa Prodi S1-IF dari kelas %s dengan NIM %s\n", nama, kelas, nim)
}
```

Screenshoot program



Deskripsi program

- package main - Deklarasi package utama

- `import "fmt"` - Import package `fmt`
- `func main() {` - Fungsi utama
- `var nama, nim, kelas string` - Deklarasi 3 variabel string sekaligus
- `fmt.Print("masukkan nama : ")` - Prompt untuk input nama
- `fmt.Scanln(&nama)` - Membaca seluruh baris input (termasuk spasi) ke variabel `nama`
- `fmt.Print("masukkan nim : ")` - Prompt untuk input NIM
- `fmt.Scanln(&nim)` - Membaca input NIM
- `fmt.Print("masukkan kelas : ")` - Prompt untuk input kelas
- `fmt.Scanln(&kelas)` - Membaca input kelas
- `fmt.Printf("Perkenalkan saya adalah %s salah satu mahasiswa Prodi SI dari kelas %s dengan NIM %s\n", nama, kelas, nim)` - Menampilkan kalimat perkenalan dengan format placeholder `%s`
- `}` - Menutup fungsi `main`

3. Tugas 3

Source code

```
package main

import "fmt"

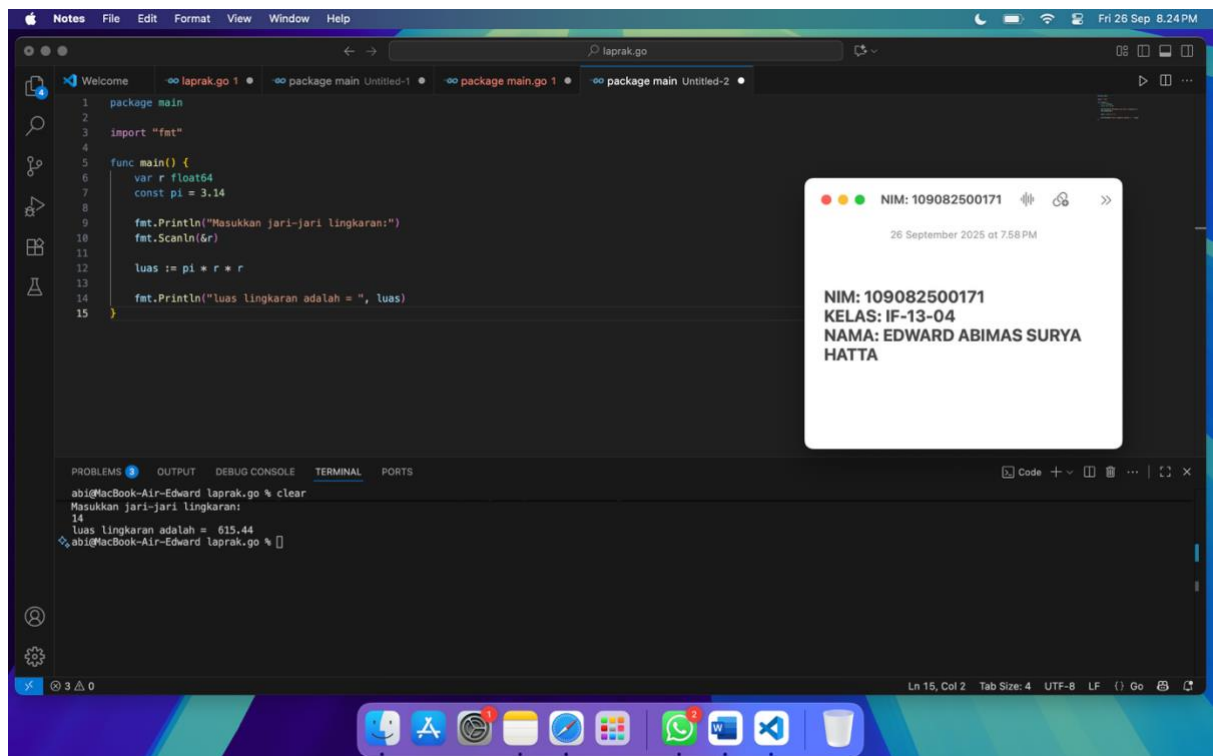
func main() {
    var r float64
    const pi = 3.14

    fmt.Println("Masukkan jari-jari lingkaran:")
    fmt.Scanln(&r)

    luas := pi * r * r

    fmt.Println("luas lingkaran adalah = ", luas)
}
```

Screenshoot program



Deskripsi program

- package main - Deklarasi package utama
- import "fmt" - Import package fmt
- func main() { - Fungsi utama
- var r float64 - Deklarasi variabel r (radius) bertipe float64
- const pi = 3.14 - Deklarasi konstanta pi dengan nilai 3.14
- fmt.Print("Masukkan jari-jari lingkaran: ") - Prompt input
- fmt.Scanln(&r) - Membaca input radius
- luas := pi * r * r - Menghitung luas dengan rumus $\pi \times r^2$ menggunakan short declaration
- fmt.Println("luas lingkaran adalah ", luas) - Menampilkan hasil perhitungan
- } - Menutup fungsi main

4.Tugas 4

Source code

```
package main

import "fmt"

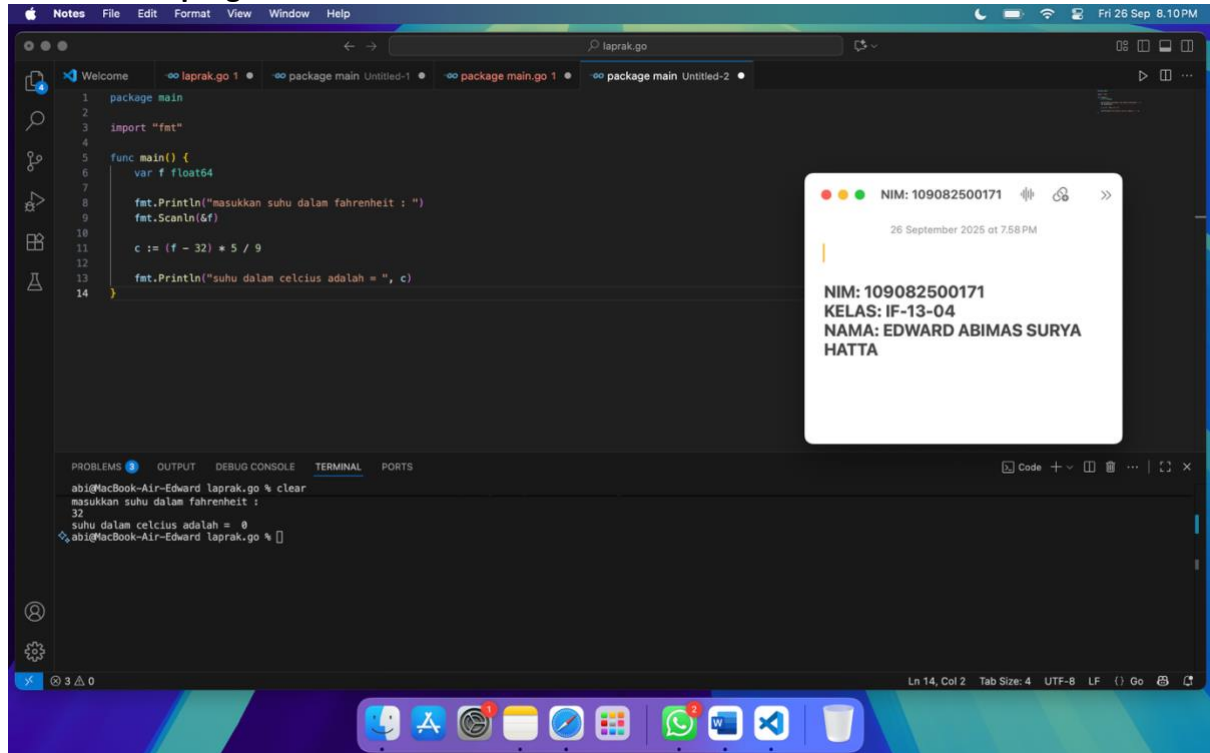
func main() {
    var f float64

    fmt.Println("masukkan suhu dalam fahrenheit : ")
    fmt.Scanln(&f)
```

```
c := (f - 32) * 5 / 9

fmt.Println("suhu dalam celcius adalah = ", c)
}
```

Screenshoot program



Deskripsi program

- package main - Deklarasi package utama
- import "fmt" - Import package fmt
- func main() { - Fungsi utama
- var f float64 - Deklarasi variabel f (fahrenheit) bertipe float64
- fmt.Print("Masukkan suhu dalam fahrenheit : ") - Prompt input
- fmt.Scanln(&f) - Membaca input suhu dalam Fahrenheit
- c := (f - 32) * 5 / 9 - Konversi Fahrenheit ke Celsius dengan rumus $C = (F - 32) \times \frac{5}{9}$
- fmt.Println("suhu dalam celsius adalah = ", c) - Menampilkan hasil konversi
- } - Menutup fungsi main

TUGAS PENDAHULUAN

Tugas Pendahuluan

Tugas pendahuluan wajib dikerjakan untuk semua praktikan dan dikumpulkan bersamaan dengan laporan praktikum (laprak), dengan deadline yang sama.

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.

1. Perbedaan `fmt.Print()`, `fmt.Println()`, dan `fmt.Printf()` di Go

`fmt.Print()`:

- Mencetak output tanpa menambahkan baris baru di akhir
- Tidak menambahkan spasi antar argumen jika berupa string
- Contoh: `fmt.Print("Hello", "World")` → output: HelloWorld

`fmt.Println()`:

- Mencetak output dan menambahkan baris baru (`\n`) di akhir
- Menambahkan spasi antar argumen
- Contoh: `fmt.Println("Hello", "World")` → output: Hello World\n

`fmt.Printf()`:

- Mencetak output dengan format tertentu (formatted output)
- Menggunakan format specifier seperti `%s`, `%d`, `%f`, dll.
- Tidak menambahkan baris baru secara otomatis
- Contoh: `fmt.Printf("Nama: %s, Umur: %d", "John", 25)` → output: Nama: John, Umur: 25

2. Penggunaan Tipe Data di Go

`int`:

- Tipe data bilangan bulat
- Ukuran bergantung pada arsitektur sistem (32-bit atau 64-bit)
- Contoh: `var umur int = 25`

float64:

- Tipe data bilangan pecahan presisi ganda (64-bit)
- Dapat menyimpan angka desimal
- Contoh: `var tinggi float64 = 175.5`

bool:

- Tipe data boolean yang hanya memiliki dua nilai: true atau false
- Contoh: `var aktif bool = true`

string:

- Tipe data untuk menyimpan teks/karakter
- Menggunakan tanda kutip ganda
- Contoh: `var nama string = "John Doe"`

3. Cara Mendeklarasikan Variabel dengan var dan Short Declaration

Menggunakan kata kunci var:

go

```
var nama string = "John"
```

```
var umur int = 25
```

```
var aktif bool = true
```

// Atau tanpa nilai awal (akan menggunakan zero value)

```
var nama string
```

```
var umur int
```

Menggunakan short declaration (:=):

go

```
nama := "John"
```

```
umur := 25
```

```
aktif := true
```

Perbedaan:

- `var` dapat digunakan di level package dan function
- `:=` hanya dapat digunakan di dalam function
- `:=` secara otomatis menentukan tipe data berdasarkan nilai yang diberikan
- `var` memungkinkan deklarasi tanpa inisialisasi

4. Perbedaan Operator == dan = dalam Go

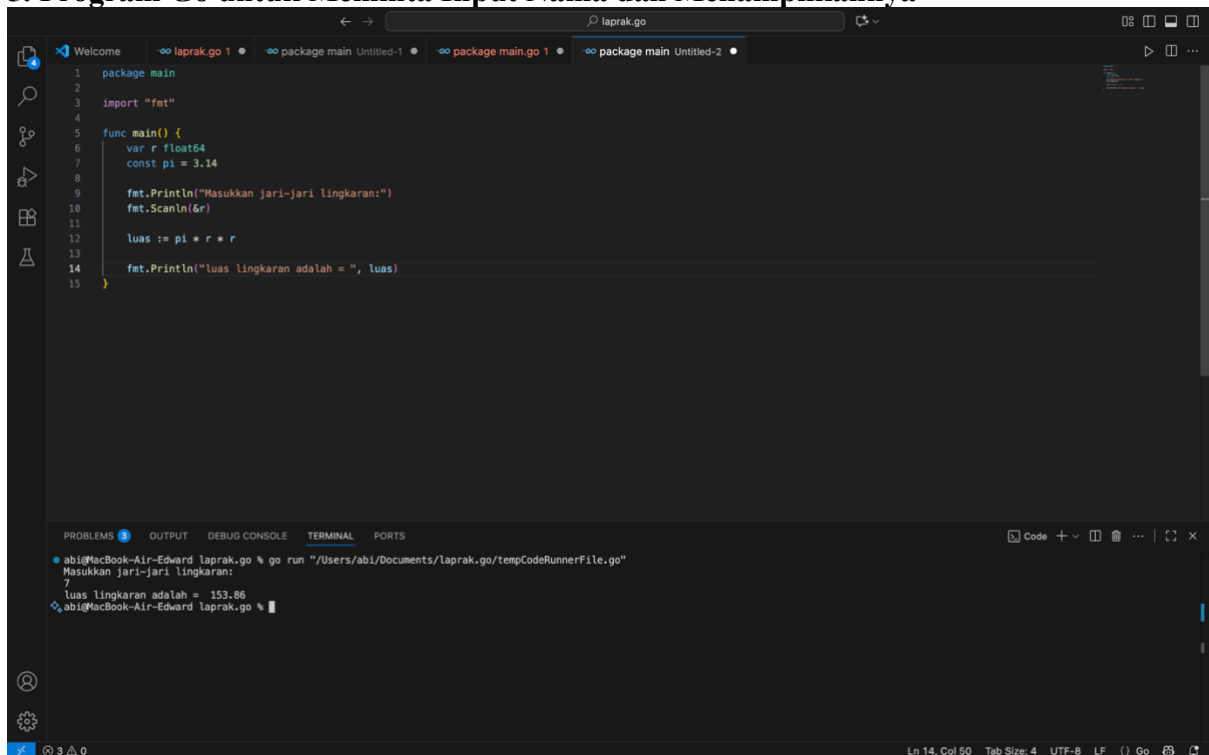
Operator = (Assignment):

- Digunakan untuk memberikan nilai kepada variabel
- Merupakan operator penugasan
- Contoh: `nama = "John"` (memberikan nilai "John" ke variabel nama)

Operator == (Comparison):

- Digunakan untuk membandingkan dua nilai
- Mengembalikan nilai boolean (true/false)
- Contoh: `nama == "John"` (mengecek apakah nilai nama sama dengan "John")

5. Program Go untuk Meminta Input Nama dan Menampilkannya



```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var r float64
7     const pi = 3.14
8
9     fmt.Println("Masukkan jari-jari lingkaran:")
10    fmt.Scanln(&r)
11
12    luas := pi * r * r
13
14    fmt.Println("luas lingkaran adalah = ", luas)
15 }
```

The screenshot shows a Go program in a code editor with a dark theme. The code calculates the area of a circle by taking a radius input from the user. The output window at the bottom shows the execution results: the prompt 'Masukkan jari-jari lingkaran:' is followed by the user input '7', and the program outputs 'luas lingkaran adalah = 153.86'.