

LAPORAN PRAKTIKUM
Algoritma Pemrograman

MODUL 2
I/O, TIPE DATA & VARIABEL



Disusun oleh:

AKHMAD NOVAL ANNUR

109082500100

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)

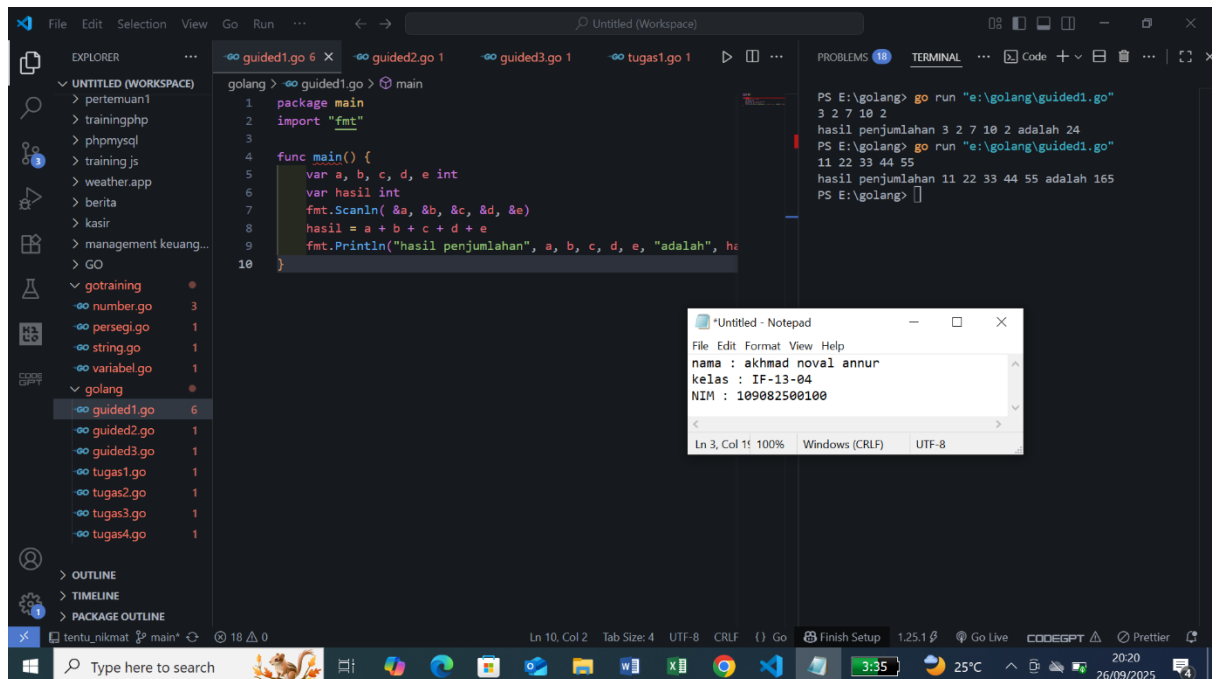
    hasil = a + b + c + d + e

    fmt.Println("hasil penjumlahan", a, b, c, d, e, "adalah",
hasil)
}
```

Screenshoot program

//tambahkan tangkapan layar dari program (boleh lebih dari 1 jika diperlukan)

CONTOH TANGKAPAN LAYAR:



Deskripsi program

Program ini dibuat untuk melakukan operasi penjumlahan terhadap lima angka yang dimasukkan oleh pengguna. Di awal program, lima variable a,b,c,d dan e

dideklarasikan bertipe `int` untuk menyimpan input dari pengguna, serta satu variable `hasil` yang berfungsi sebagai penampung hasil perhitungan. Setelah itu, program menggunakan fungsi `fmt.Scanln()` untuk membaca lima angka yang dimasukkan melalui keyboard secara berurutan.

Setelah semua angka tersimpan, program menjumlahkan kelima nilai tersebut dan menyimpannya ke dalam variable `hasil`. Langkah terakhir adalah menampilkan hasil perhitungan ke layar menggunakan `fmt.Println()` di mana program tidak hanya menampilkan hasil akhir tetapi juga memperlihatkan angka-angka yang diinput sebelumnya. Melalui program sederhana ini, kita dapat memahami konsep dasar seperti deklarasi variabel, proses input-output, serta operasi aritmetika dasar dalam bahasa Go.

2. Guided 2

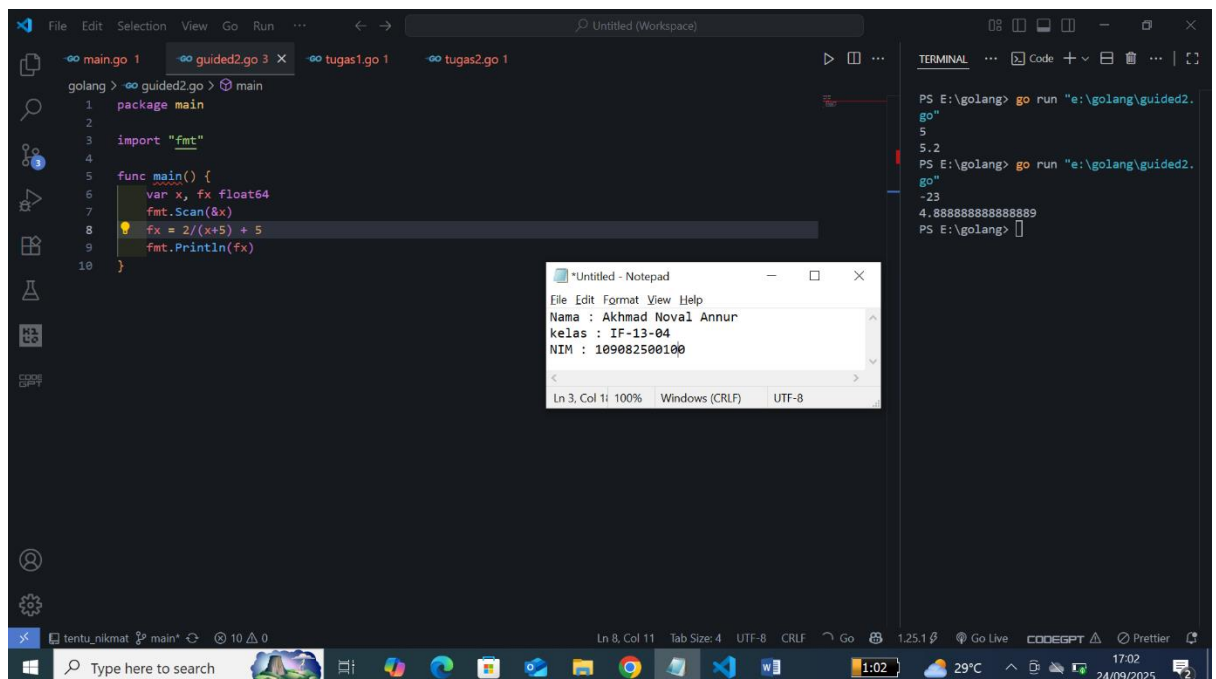
Source Code

```
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

Program ini bertujuan untuk membaca data diri mahasiswa berupa nama, NIM, dan kelas, lalu menampilkannya dalam bentuk kalimat perkenalan. Pertama, tiga variabel bertipe string (nama, nim dan kelas) dideklarasikan untuk menyimpan input dari pengguna. Program kemudian meminta pengguna memasukkan ketiga data tersebut menggunakan `fmt.Scanf()` secara berurutan.

Setelah data berhasil dibaca, program menyusunnya ke dalam format kalimat dan menampilkannya di layar, misalnya: “Perkenalkan saya adalah [nama], salah satu mahasiswa Prodi S1-IF dari kelas [kelas] dengan NIM [nim].” Program ini melatih

cara mengambil input string dari pengguna, memproses data, dan menampilkan output dalam format kalimat yang lebih informatif dan komunikatif.

3. Guided 3

Source Code

```
package main

import "fmt"

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

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

    var str string
    fmt.Scan(&str)

    b1 = str[0]
    b2 = str[1]
    b3 = str[2]

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

Screenshoot program

The screenshot shows a Visual Studio Code editor with a Go file named `guided3.go`. The code defines a `main` function that reads five bytes and a string, then prints them with their ASCII values. A terminal window shows the command `go run "e:\golang\guided3.go"` and its output: `66 97 103 117 115`, `SNO`, `Bagus`, `TOP`, and `PS E:\golang>`. A Notepad window is also open, displaying student information: `Nama : Akhmad Noval Annur`, `kelas : IF-13-04`, and `NIM : 109082500100`.

```
1 package main
2 import "fmt"
3
4 func main() {
5     var c1, c2, c3, c4, c5 byte
6     var b1, b2, b3 byte
7
8     fmt.Scan(&c1, &c2, &c3, &c4, &c5)
9
10    var str string
11    fmt.Scan(&str)
12
13    b1 = str[0]
14    b2 = str[1]
15    b3 = str[2]
16
17    fmt.Printf("%c%c%c%c%c\n", c1, c2, c3, c4, c5)
18    fmt.Printf("%c%c%c\n", b1+1, b2+1, b3+1)
19 }
```

```
PS E:\golang> go run "e:\golang\guided3.go"
66 97 103 117 115
SNO
Bagus
TOP
PS E:\golang>
```

```
File Edit Format View Help
Name : Akhmad Noval Annur
kelas : IF-13-04
NIM : 109082500100
```

Deskripsi program

Program ini bertujuan untuk memproses dan memanipulasi karakter serta string yang dimasukkan oleh pengguna. Pertama, lima variabel `c1` hingga `c5` dideklarasikan sebagai `byte` untuk menyimpan karakter satu per satu, sedangkan tiga variabel `b1`, `b2`, dan `b3` digunakan untuk menyimpan hasil pengolahan karakter dari string. Program meminta pengguna memasukkan lima karakter terlebih dahulu, kemudian dilanjutkan dengan memasukkan sebuah string.

Setelah input diterima, tiga karakter pertama dari string diambil dan disimpan ke dalam `b1`, `b2`, dan `b3`. Program selanjutnya mencetak lima karakter awal yang dimasukkan serta mencetak tiga karakter dari string yang telah ditambahkan dengan nilai 1 pada kode ASCII-nya. Program ini membantu memahami cara membaca input bertipe karakter, melakukan manipulasi data menggunakan kode ASCII, serta menampilkan hasil pemrosesan ke layar.

TUGAS

1. Tugas 1

Source code

```
package main

import "fmt"

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

    fmt.Print("Masukan input string: ")
    fmt.Scanln(&satu)

    fmt.Print("Masukan input string: ")
    fmt.Scanln(&dua)

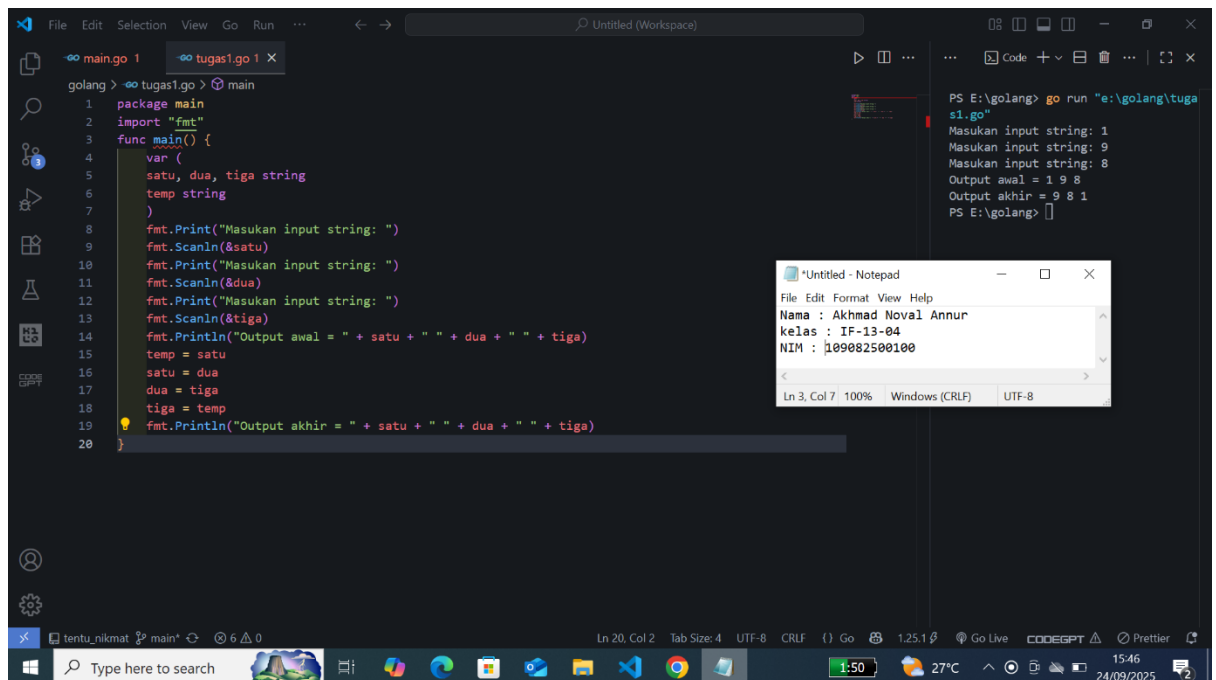
    fmt.Print("Masukan input string: ")
    fmt.Scanln(&tiga)

    fmt.Println("Output awal = " + satu + " " + dua + " " + tiga)

    temp = satu
    satu = dua
    dua = tiga
    tiga = temp

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


Screenshoot program



The screenshot shows a Go program in a VS Code editor. The program is named `tugas1.go` and is located in the `main` package. It imports the `fmt` package and defines a `main` function. Inside the `main` function, three string variables (`satu`, `dua`, `tiga`) are declared, and a temporary string variable (`temp`) is also declared. The program prompts the user to enter three strings, one by one, and stores them in the `satu`, `dua`, and `tiga` variables. It then prints the initial output of the three strings. Finally, it prints the output after swapping the values of `satu` and `dua` using the `temp` variable as a temporary storage.

```
1 package main
2 import "fmt"
3 func main() {
4     var (
5         satu, dua, tiga string
6         temp string
7     )
8     fmt.Println("Masukan input string: ")
9     fmt.Scanln(&satu)
10    fmt.Println("Masukan input string: ")
11    fmt.Scanln(&dua)
12    fmt.Println("Masukan input string: ")
13    fmt.Scanln(&tiga)
14    fmt.Println("Output awal = " + satu + " " + dua + " " + tiga)
15    temp = satu
16    satu = dua
17    dua = tiga
18    tiga = temp
19    fmt.Println("Output akhir = " + satu + " " + dua + " " + tiga)
20 }
```

The terminal output shows the execution of the program. It prompts the user to enter three strings: "1", "9", and "8". The initial output is "1 9 8". After swapping the values of `satu` and `dua`, the final output is "9 8 1".

Deskripsi program

Program ini digunakan untuk menukar posisi tiga buah string yang dimasukkan oleh pengguna. Di awal, tiga variabel bertipe string (`satu`, `dua`, dan `tiga`) dideklarasikan untuk menyimpan input, sedangkan variabel `temp` digunakan sebagai penampung sementara saat proses pertukaran berlangsung. Program meminta pengguna untuk memasukkan tiga string secara bergantian, kemudian menampilkan urutan awal dari ketiga string tersebut.

Selanjutnya, proses pertukaran dilakukan dengan memindahkan nilai dari variabel `satu` ke variabel lainnya menggunakan `temp` sebagai perantara. Setelah proses pertukaran selesai, program menampilkan hasil akhir string dengan urutan yang telah berubah. Program ini menunjukkan cara kerja pertukaran nilai antar variabel dan melatih pemahaman dalam mengelola data string secara dinamis.

2. Tugas 2

Source code

```
package main

import (
    "fmt"
)

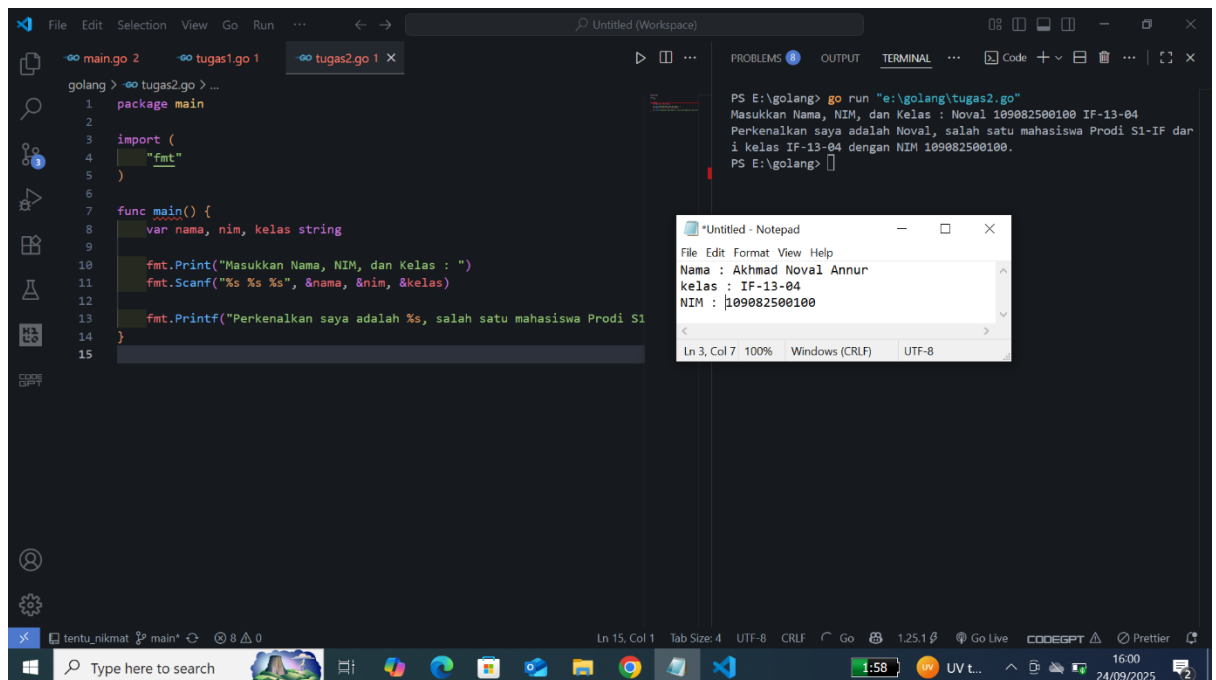
func main() {
    var nama, nim, kelas string

    fmt.Print("Masukkan Nama, NIM, dan Kelas : ")
    fmt.Scanf("%s %s %s", &nama, &nim, &kelas)

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

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

Screenshoot program



The screenshot shows a Go program in VS Code and its execution output. The program is located in `tugas2.go` and uses `fmt` to read user input and print a formatted introduction. The terminal shows the command `go run "e:\golang\tugas2.go"` and the output of the program. A Notepad window shows the input provided by the user.

```
golang > main.go 2  tugas1.go 1  tugas2.go 1 X
1 package main
2
3 import (
4     "fmt"
5 )
6
7 func main() {
8     var nama, nim, kelas string
9
10    fmt.Print("Masukkan Nama, NIM, dan Kelas : ")
11    fmt.Scanf("%s %s %s", &nama, &nim, &kelas)
12
13    fmt.Printf("Perkenalkan saya adalah %s, salah satu mahasiswa Prodi S1
14
15
```

```
PS E:\golang> go run "e:\golang\tugas2.go"
Masukkan Nama, NIM, dan Kelas : Noval 109082500100 IF-13-04
Perkenalkan saya adalah Noval, salah satu mahasiswa Prodi S1-IF dar
i kelas IF-13-04 dengan NIM 109082500100.
PS E:\golang>
```

```
File Edit Format View Help
Nama : Akhmad Noval Annur
kelas : IF-13-04
NIM : 109082500100
Ln 3, Col 7 100% Windows (CRLF) UTF-8
```

Deskripsi program

Program ini bertujuan untuk membaca data diri mahasiswa berupa nama, NIM, dan kelas, lalu menampilkannya dalam bentuk kalimat perkenalan. Pertama, tiga variabel bertipe string (nama, nim, dan kelas) dideklarasikan untuk menyimpan input dari pengguna. Program kemudian meminta pengguna memasukkan ketiga data tersebut menggunakan `fmt.Scanf()` secara berurutan.

Setelah data berhasil dibaca, program menyusunnya ke dalam format kalimat dan menampilkannya di layar, misalnya: “Perkenalkan saya adalah [nama], salah satu mahasiswa Prodi S1-IF dari kelas [kelas] dengan NIM [nim].” Program ini melatih cara mengambil input string dari pengguna, memproses data, dan menampilkan output dalam format kalimat yang lebih informatif dan komunikatif.

3. Tugas 3

Source code

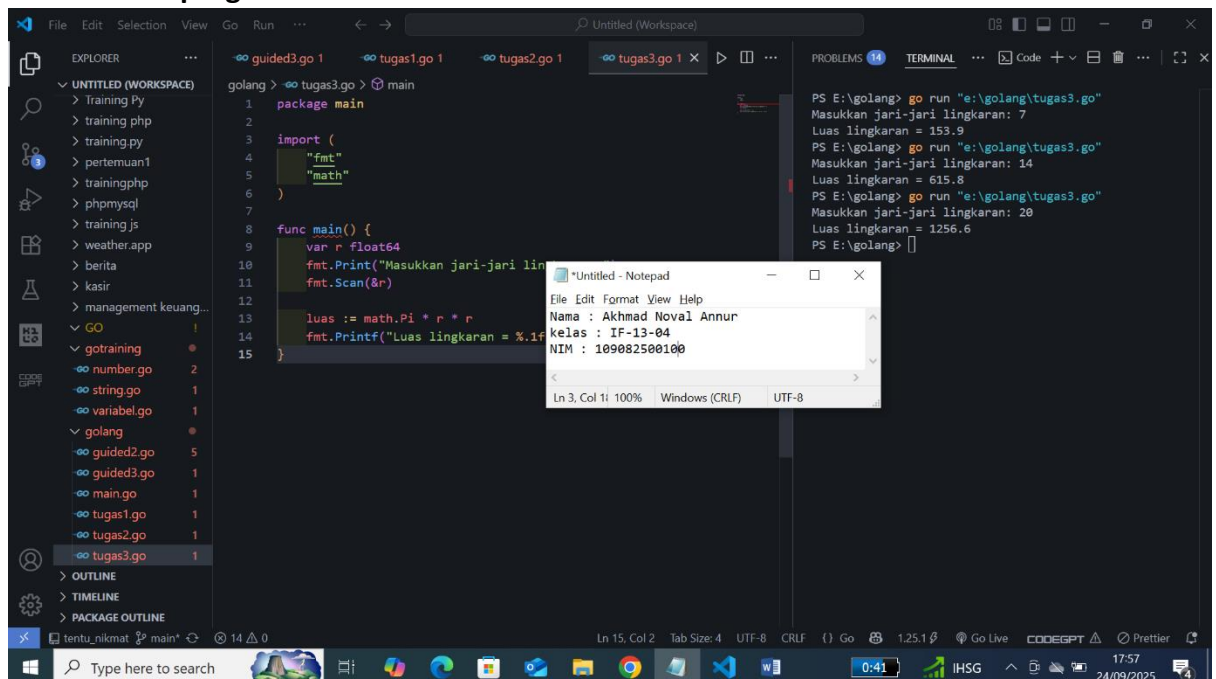
```
package main

import (
    "fmt"
    "math"
)

func main() {
    var r float64
    fmt.Print("Masukkan jari-jari lingkaran: ")
    fmt.Scan(&r)

    luas := math.Pi * r * r
    fmt.Printf("Luas lingkaran = %.1f\n", luas)
}
```

Screenshoot program



Deskripsi program

Program ini digunakan untuk menghitung luas lingkaran berdasarkan jari-jari yang dimasukkan oleh pengguna. Variabel `r` bertipe `float64` dideklarasikan untuk menyimpan nilai jari-jari, kemudian program meminta pengguna memasukkan nilai tersebut melalui fungsi `fmt.Scan()`. Setelah nilai diterima, luas lingkaran dihitung menggunakan rumus $\pi * r * r$ dengan bantuan konstanta `math.Pi`.

Hasil perhitungan kemudian ditampilkan ke layar dengan satu angka di belakang koma menggunakan `fmt.Printf()`. Program ini memberikan pemahaman tentang cara menggunakan pustaka `math`, mengoperasikan nilai pecahan dalam perhitungan matematika, dan menampilkan hasil perhitungan dalam format yang lebih presisi

4. Tugas 4

Source code

```
package main

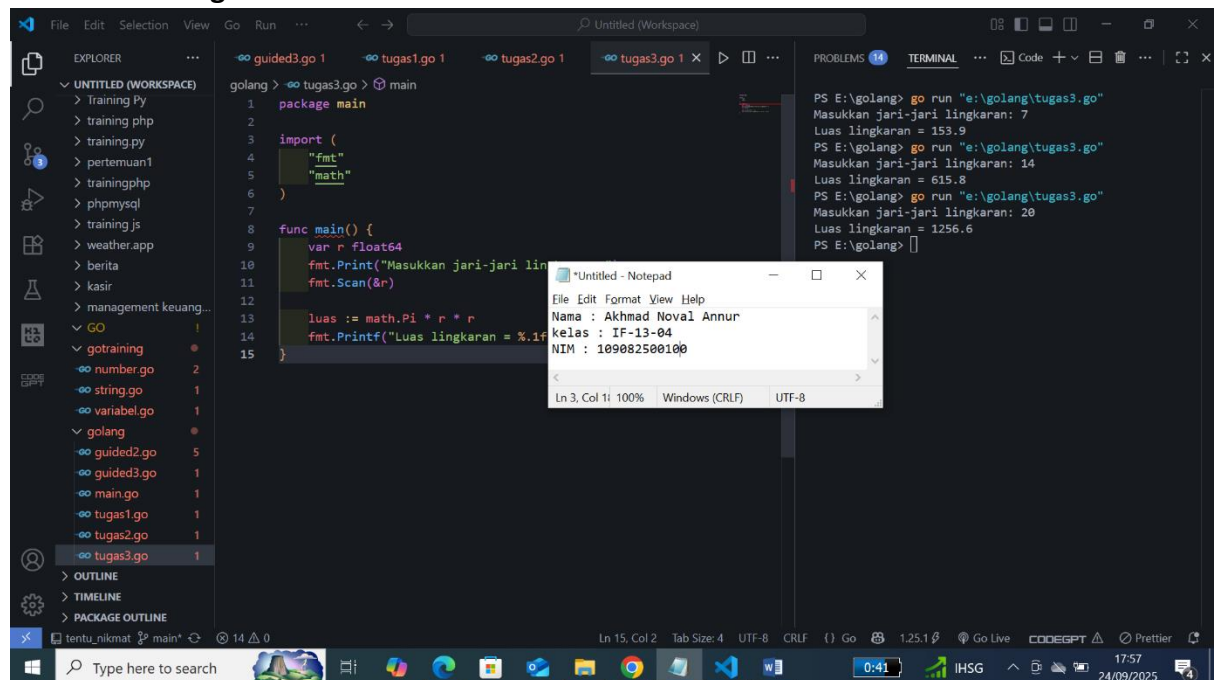
import "fmt"

func main() {
    var fahrenheit float64
    fmt.Print("Masukkan suhu dalam Fahrenheit: ")
    fmt.Scanln(&fahrenheit)

    celcius := (fahrenheit - 32) * 5 / 9

    fmt.Printf("Suhu dalam Celcius: %.2f\n", celcius)
}
```

Screenshot Program



Deskripsi program

Program ini berfungsi untuk mengonversi suhu dari satuan Fahrenheit ke Celcius. Pertama, variabel fahrenheit bertipe float64 dideklarasikan untuk menyimpan nilai input suhu dari pengguna. Program kemudian meminta pengguna memasukkan suhu dalam Fahrenheit melalui fungsi `fmt.Scanln()`.

Setelah nilai suhu diterima, program melakukan konversi ke Celcius menggunakan rumus $(fahrenheit - 32) * 5 / 9$ dan menyimpannya ke dalam variabel

celcius. Hasil konversi ditampilkan ke layar dengan dua angka di belakang koma menggunakan `fmt.Printf()`. Program ini membantu memahami penggunaan operasi aritmetika dasar untuk konversi satuan, serta cara menampilkan hasil perhitungan dalam format yang lebih rapi.

TUGAS PENDAHULUAN

1. Jelaskan perbedaan `fmt.Print()`, `fmt.Println()`, `fmt.Printf()` di go!

Ketiga fungsi tersebut merupakan fungsi untuk menampilkan output ke layar, tetapi memiliki cara kerja yang berbeda. Fungsi `fmt.Print()` digunakan untuk mencetak teks atau nilai variabel tanpa menambahkan baris baru di akhir output, sehingga jika ada perintah `Print()` berikutnya, hasilnya akan ditulis di baris yang sama. Fungsi `fmt.Println()` digunakan untuk mencetak teks atau nilai variabel dengan menambahkan baris baru di akhir output, sehingga perintah berikutnya akan tampil di baris selanjutnya. Sementara itu, `fmt.Printf()` digunakan untuk mencetak teks dengan format tertentu menggunakan placeholder seperti `%d` untuk integer, `%f` untuk float, atau `%s` untuk string. Fungsi ini sangat berguna ketika kita ingin menampilkan teks dan variabel secara terstruktur dalam satu kalimat.

2. Jelaskan penggunaan data `int`, `float64`, `bool`, dan `string` di go!

Tipe data `int` digunakan untuk menyimpan bilangan bulat, baik positif maupun negatif, misalnya 10, -5, atau 0. Tipe data `float64` digunakan untuk menyimpan bilangan desimal atau pecahan, seperti 3.14 atau -0.75, dan biasanya dipakai dalam operasi matematika yang memerlukan presisi. Tipe data `bool` menyimpan nilai logika berupa `true` atau `false`, dan sering digunakan dalam percabangan kondisi atau perbandingan logika. Terakhir, tipe data `string` digunakan untuk menyimpan teks atau rangkaian karakter, misalnya "Halo Dunia" atau "Pemrograman Go". Pemahaman terhadap tipe data ini penting karena setiap tipe memiliki fungsi dan kegunaan yang berbeda sesuai kebutuhan program.

3. Bagaimana cara mendeklarasikan variable dengan kata kunci var dan dengan acara singkat? Berikan contoh! Hint “:=”

Dalam bahasa Go, variabel dapat dideklarasikan menggunakan kata kunci var dengan menyebutkan nama variabel dan tipe datanya. umur := 20 Misalnya:

```
umur := 20
```

Cara ini cocok digunakan jika kita ingin memisahkan proses deklarasi dan pemberian nilai. Selain itu, Go juga menyediakan cara singkat menggunakan operator := yang secara otomatis mendeteksi tipe data berdasarkan nilai yang diberikan. Contohnya:

Dalam contoh tersebut, Go akan secara otomatis mengenali bahwa umur bertipe int. Cara singkat ini sering dipakai untuk membuat kode menjadi lebih ringkas dan mudah dibaca.

4. Apa perbedaan antara operator == dan = dalam Bahasa Go?

Operator == adalah operator pembandingan yang digunakan untuk memeriksa apakah dua nilai sama. Hasil dari operasi ini adalah nilai boolean (true atau false). Misalnya, a == b akan bernilai true jika a dan b memiliki nilai yang sama. Sementara itu, operator = adalah operator penugasan yang digunakan untuk memberikan nilai ke sebuah variabel. Contohnya, a = 10 berarti kita memberikan nilai 10 ke dalam variabel a. Perbedaan utama keduanya adalah == digunakan dalam perbandingan kondisi, sedangkan = digunakan untuk menyimpan nilai.

5. Buatlah program Go sederhana untuk meminta input nama kalian , lalu menampilkan nama.

Berikut contoh program sederhana menggunakan bahasa Go untuk meminta pengguna memasukkan nama, kemudian menampilkannya kembali ke layar: package main

```
Package main
```

```
import "fmt"
```

```
func main() {
```

```
    var nama string
```

```
    fmt.Print("Masukkan nama Anda: ")
```

```
    fmt.Scanln(&nama)
```

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

```
}
```

Penjelasan: Program di atas menggunakan variabel nama bertipe string untuk menyimpan input dari pengguna. Fungsi `fmt.Print()` digunakan untuk menampilkan pesan agar pengguna memasukkan nama, sedangkan `fmt.Scanln()` berfungsi untuk membaca input tersebut. Setelah itu, program mencetak sapaan menggunakan `fmt.Println()` dengan menampilkan nama yang telah dimasukkan.