

**LAPORAN PRAKTIKUM**  
**Algoritma Pemrograman**

**MODUL No. 2**  
**I/O, Tipe Data & Variabel**



**Disusun oleh:**

**Leonardo Farriz Garcya**

**109082530036**

**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

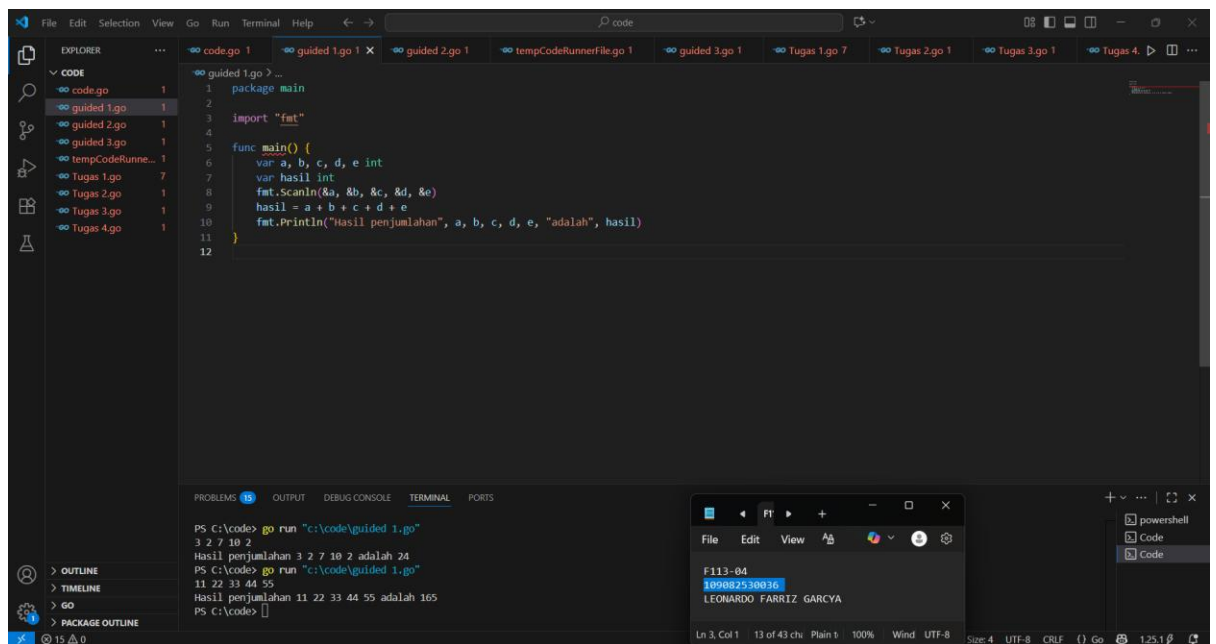
    fmt.Scanln(&a, &b, &c, &d, &e)

    hasil = a + b + c + d + e

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

}
```

### Screenshoot program



## Deskripsi program

### 1.package main

- Bahwa program ini adalah program utama yang bisa langsung dijalankan.
- Tanpa main, kode tidak bisa dieksekusi sebagai aplikasi.

### 2.import "fmt"

- Format (fmt), digunakan untuk input/output standar.
- Seperti "fmt.Scanln" : untuk membaca input dari user.  
"fmt.Println" : menampilkan output ke layar.

### 3.func main () { ... }

- Fungsi **main** adalah untuk titik awal eksekusi program Go.
- Semua intruksi akan berjalan dari fungsi ini.

### 4.var a, b, c, d, e int

- Menyatakan satu variabel integer ( int ) yang akan menampung input bilangan bulat dari user.

### 5.var hasil int

- Menyatakan satu variabel integer bernama **hasil** untuk menyimpan penjumlahan.

### 6.fmt.Scanln(&a, &b, &c, &d, &e

- Membaca input dari user berupa 5 bilangan bulat.
- **&a** artinya alamat memori variabel **a** dan nilai input akan disimpan ke variabel **a**.
- User harus memasukan 5 angka **dipisahkan spasi** lalu tekan Enter.

### 7.hasil = a + b + c + d + e

- Untuk menjumlahkan kelima bilangan tersebut.
- Hasilnya akan disimpan ke variabel hasil.

### 8.fmt.Println("Hasil penjumlahan", a, b, c, d, e, "adalah", hasil )

- Untuk mencetak hasil ke layar dengan format

### 9.Cara Menjalankannya

1. Simpan kode ke file
2. Buka terminal " Ctrl J "
3. Lalu jalankan
4. Masukan angka, Input 1: 3 2 7 10 2
5. Lalu akan muncul di terminal, Output 1: Hasil penjumlahan 3 2 7 10 2 adalah 24
6. Lalu masukan Input 2: 11 22 33 44 55

7. Dan akan muncul Output 2: Hasil penjumlahan 11 22 33 44 55 adalah 165

## 2. Guided 2

### Source Code

```
package main

import "fmt"

func main() {

    var x, fx float64

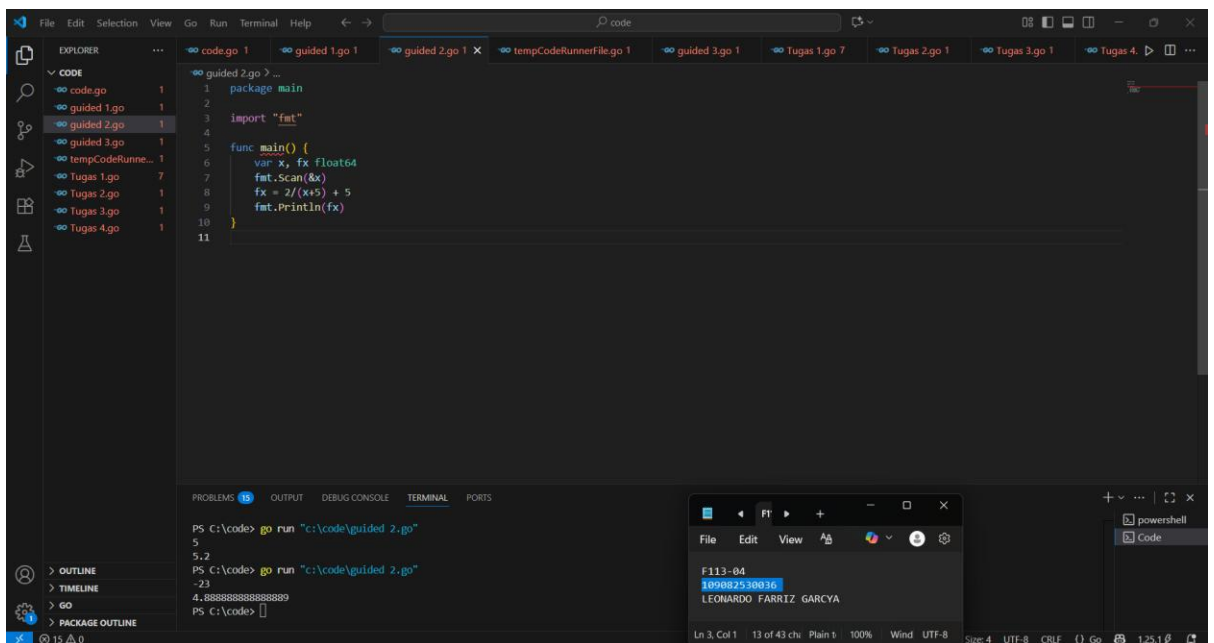
    fmt.Scan(&x)

    fx = 2/(x+5) + 5

    fmt.Println(fx)

}
```

### Screenshoot program



## Deskripsi program

1.package main

- Program ini adalah program utama yang bisa langsung dijalankan.
- Tanpa main, kode tidak bisa dieksekusi sebagai aplikasi.

2.import "fmt"

- Format (fmt), digunakan untuk input/output standar.
- Seperti "fmt.Scanln" : untuk membaca input dari user.  
"fmt.Println" : menampilkan output ke layar.

3.func main () { ... }

- Fungsi **main** adalah untuk titik awal eksekusi program Go.
- Semua intruksi akan berjalan dari fungsi ini.

4.var x, fx float64

- Menyatakan dua variable tipe **float64**(bilangan pecahan).
- Seperti **x** :untuk menampung input bilangan dari user.  
**fx** :untuk menampung hasil perhitungan fungsi f(x).

5.fmt.Scan(&x)

- Untuk membaca input dari pengguna, lalu membaca ke variabel **x**.
- Tanda **&** artinya menyimpan nilai input langsung ke alamat memori variabel **x**.

6.fx = 2/(x+5) + 5

- Untuk menghitung persamaan matematika sesuai dengan soal.

7.fmt.Println(fx)

- Untuk menampilkan nilai **fx** ke layar.
- Outputnya akan berupa bilangan desimal(floating point)

8. Cara Menjalankannya

### Disclaimer

**Input 1 (soal 1)**

**Output 1 (hasil soal 1)**

**Input 2 (soal 2)**

**Output 2 (hasil soal 2)**

1. Simpan kode ke file

2. Buka terminal " **Ctrl J** "

3. Lalu jalankan

4. Masukkan angka, Input 1: **5**

5. Lalu akan muncul di terminal, Output 1 (hasilnya): **5,2**

6. Dan masukan soal berikutnya(Input ke 2, Input 2: **-23**

7. Lalu akan muncul di terminal, Output 2 (hasilnya):

**4.888888888888889**

### 3. Guided 3

#### Source Code

```
// file name: ascii.go

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)

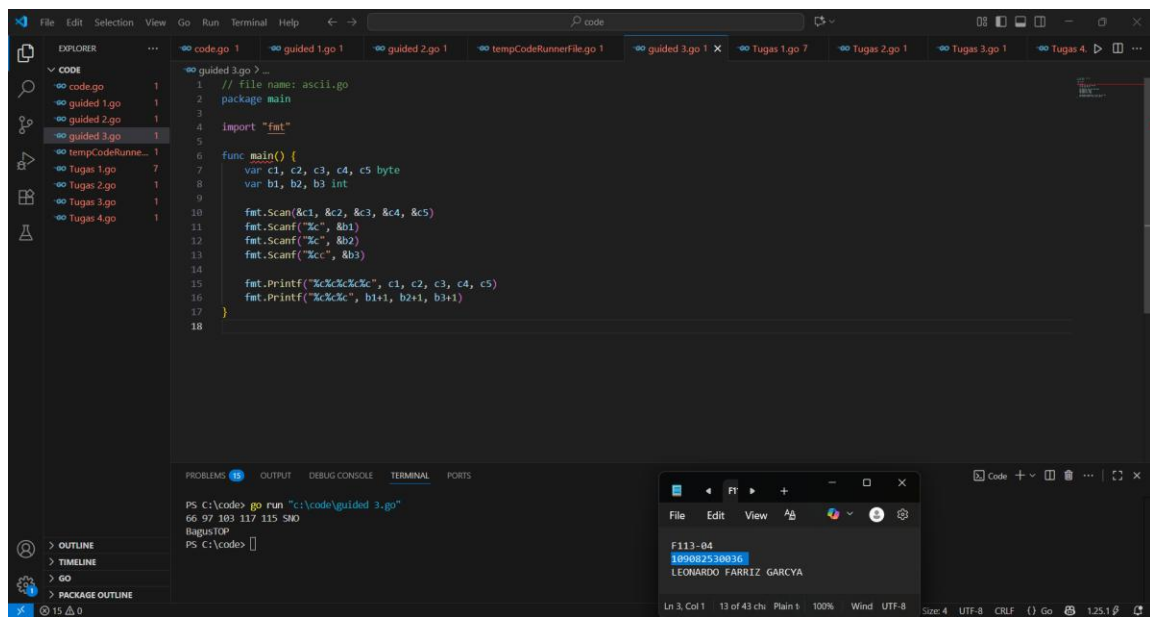
    fmt.Scanf("%cc", &b3)

    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



The screenshot shows the Visual Studio Code interface. The Explorer panel on the left lists several files, including 'guided 3.go'. The main editor displays the content of 'guided 3.go', which is a Go program. The program defines a package 'main', imports the 'fmt' package, and contains a 'main' function. The function declares five byte variables (c1-c5) and three integer variables (b1-b3). It uses 'fmt.Scanf' to read input from the user and 'fmt.Printf' to print the results. The output panel at the bottom shows the command 'go run "C:\code\guided 3.go"' being executed, with the output '66 97 103 117 115 SMO' and 'BagusTOP'.

```
1 // file name: ascii.go
2 package main
3
4 import "fmt"
5
6 func main() {
7     var c1, c2, c3, c4, c5 byte
8     var b1, b2, b3 int
9
10    fmt.Scan(&c1, &c2, &c3, &c4, &c5)
11    fmt.Printf("%c", &b1)
12    fmt.Printf("%c", &b2)
13    fmt.Printf("%c", &b3)
14
15    fmt.Printf("%c%c%c%c", c1, c2, c3, c4, c5)
16    fmt.Printf("%c%c", b1+1, b2+1, b3+1)
17 }
18
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\code> go run "C:\code\guided 3.go"

66 97 103 117 115 SMO

BagusTOP

PS C:\code>

## Deskripsi program

### 1.package main

- Program ini adalah program utama yang bisa langsung dijalankan.
- Tanpa main, kode tidak bisa dieksekusi sebagai aplikasi.

### 2.import "fmt"

- Format (fmt), digunakan untuk input/output standar.
- Seperti "fmt.Scanln" : untuk membaca input dari user.
- "fmt.Println" : menampilkan output ke layar.

### 3.func main () { ... }

- Fungsi **main** adalah untuk titik awal eksekusi program Go.
- Semua intruksi akan berjalan dari fungsi ini.

### 4. var c1,...,c5 byte

- Merancang 5 variabel untuk menampung nilai ASCII(0-255).

### 5. var b1,b2,b3 int

- Merancang 3 variabel untuk menampung pembacaan karakter dari baris kedua.
- Menyimpan karakter dalam **int** juga dimungkinkan saat dicetak dengan **%c**, nilai integer akan dianggap kode Unicode/ASCII dan dicetak sebagai karakter.

### 6.fmt.Scanf(&c1, &c2, &c3, &c4, &c5)

- Untuk membaca 5 token angka dari input dan dipisah spasi.
- Seperti: 66 97 103 117 115.

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

- Dipanggil 3 kali. Berusaha membaca satu karakter tiap pemanggilan.
- Format string "%c" pada Scanf itu sebagai membaca 1 karakter mentah.
- Untuk nilai yang di input kedalam **b1** samapai **b3** disimpan pada variabel bertipe integer.
- Berusaha membaca satu karakter tiap pemanggilan.

8. fmt.Printf("%c%c%c%c%c", c1, c2, c3, c4, c5)

- %c mencetak nilai numerik sebagai karakter Unicode dengan kode tersebut
- Jika c1..c5 berisi 66,97,103,117,115, maka akan mencetak Bagus (B=66, a=97, g=103, u=117, s=115).

9. fmt.Printf("%c%c%c", b1+1, b2+1, b3+1)

- Menambahkan 1 ke setiap nilai integer lalu mencetaknya sebagai karakter. Misal bila b1='S' (83) maka b1+1 = 84 → %c mencetak 'T'.

#### 10. Cara Menjalankannya

1. Simpan kode ke file
2. Buka terminal " Ctrl J "
3. Lalu jalankan
4. Masukkan angka, 66 97 103 117 115 SNO
5. Lalu akan muncul di terminal " BAGUS TOP "

## 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

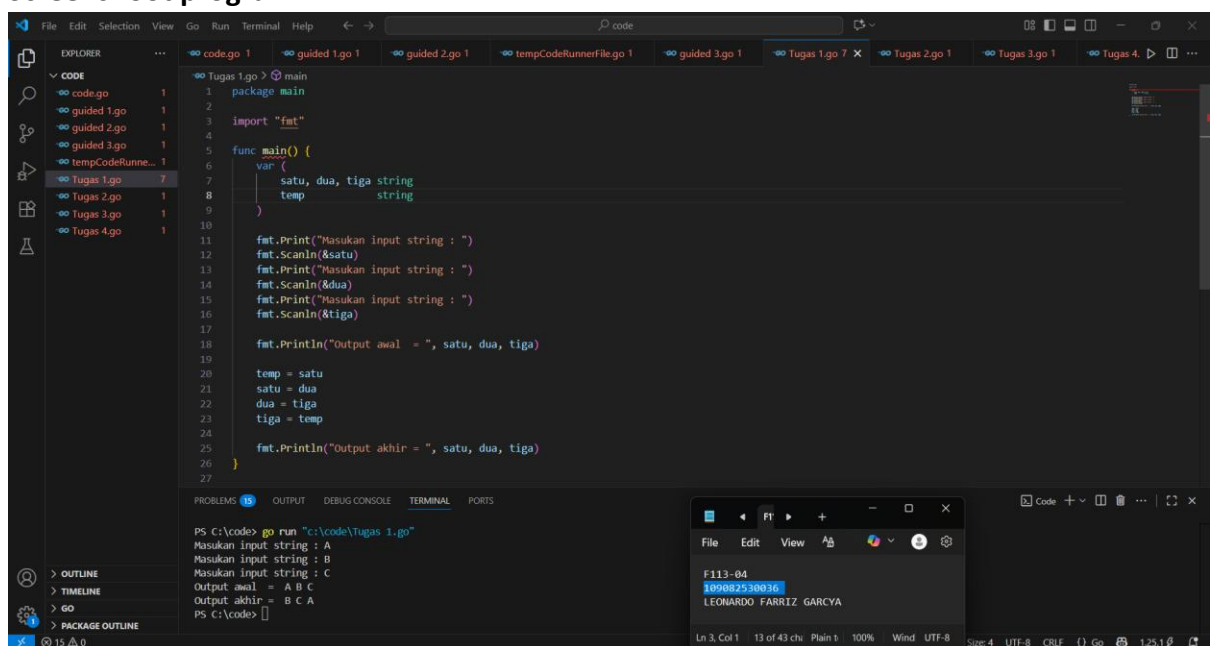
    tiga = temp


    fmt.Println("Output akhir = ", satu, dua, tiga)

}

```

## Screenshoot program



## Deskripsi program

1.package main

- Program ini adalah program utama yang bisa langsung dijalankan.
- Tanpa main, kode tidak bisa dieksekusi sebagai aplikasi.

2.import "fmt"

- Format (fmt), digunakan untuk input/output standar.
- Seperti "fmt.Scanln" : untuk membaca input dari user.  
"fmt.Println" : menampilkan output ke layar.

3.func main () { ... }

- Fungsi **main** adalah untuk titik awal eksekusi program Go.
- Semua intruksi akan berjalan dari fungsi ini.

4. var (

satu, dua, tiga string

- Satu, dua, tiga, digunakan untuk menampung input dari pengguna.

5. temp string

- Digunakan sebagai variabel sementara (helper) saat menukar isi variabel.

6. fmt.Print("Masukan input string : ")

- Menampilkan teks ke layar tanpa newline (agar input ditulis di baris yang sama).

7.fmt.Scanln(&satu)

fmt.Scanln(&dua)

fmt.Scanln(&tiga)

- Tanda **&** digunakan untuk memberikan alamat variabel (satu)/(dua)/(tiga) agar Scanln bisa menuliskan hasil input ke sana.

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

- Menampilkan nilai dari variabel satu, dua, dan tiga sebelum dilakukan pertukaran.

9. temp = satu

- Menyimpan nilai satu ke dalam variabel sementara temp.
- Hal ini dilakukan supaya nilai satu tidak hilang ketika diganti dengan nilai lain.

10. satu = dua

- Mengganti isi variabel satu dengan nilai dari variabel dua.Begitu juga dengan dua = tiga

11. tiga = temp

- Mengganti isi variabel tiga dengan nilai dari variabel sementara **temp** (yang sebelumnya menyimpan nilai satu).

12. `fmt.Println("Output akhir = ", satu, dua, tiga)`

- Menampilkan nilai **satu**, **dua**, dan **tiga** setelah dilakukan pertukaran.

- jika input awal **A B C**, maka hasil akhir: Output akhir = **B C A**.

13. Cara Menjalankannya

1. Simpan kode ke file
2. Buka terminal " Ctrl J "
3. Lalu jalankan
4. Masukkan A, B, C satu persatu
5. Dan di terminal akan keluar:  
Output awal = A B C  
Output akhir = B C A

## 2. Tugas 2

### Source code

```
package main

import "fmt"

func main() {

    var nama, nim, kelas string

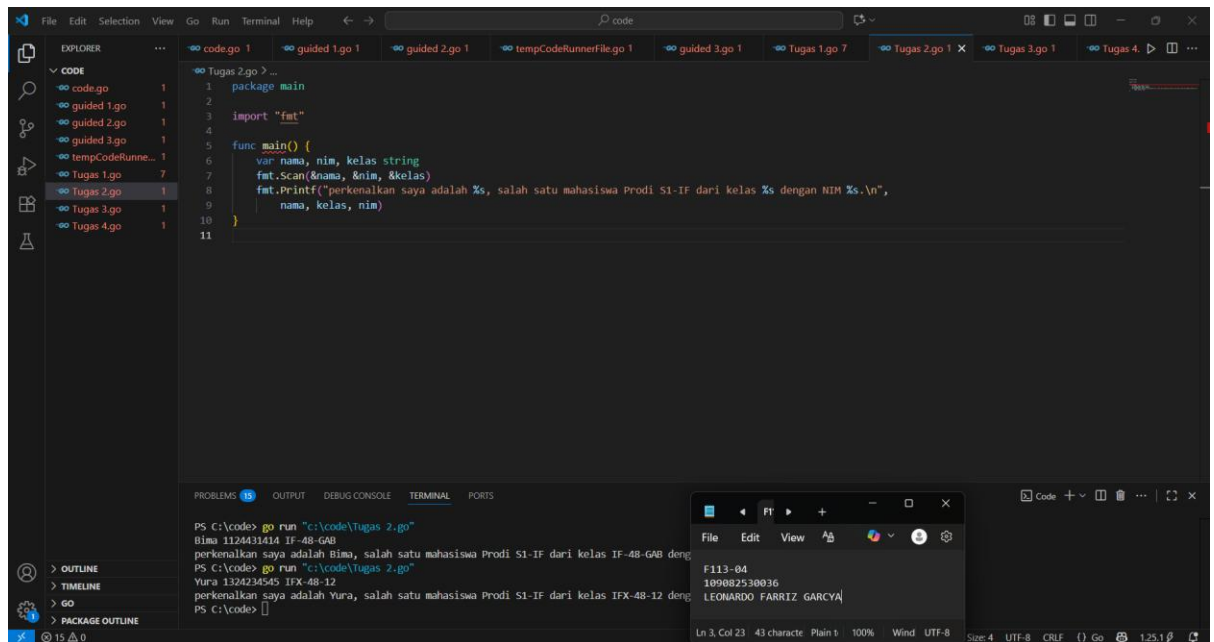
    fmt.Scan(&nama, &nim, &kelas)

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

        nama, kelas, nim)

}
```

### Screenshoot program



## Deskripsi program

### 1.package main

- Program ini adalah program utama yang bisa langsung dijalankan.
- Tanpa main, kode tidak bisa dieksekusi sebagai aplikasi.

### 2.import "fmt"

- Format (fmt), digunakan untuk input/output standar.
- Seperti "fmt.Scanln" : untuk membaca input dari user.
- "fmt.Println" : menampilkan output ke layar.

### 3.func main () { ... }

- Fungsi **main** adalah untuk titik awal eksekusi program Go.
- Semua intruksi akan berjalan dari fungsi ini.

### 4. var nama, nim, kelas string

- Menyatakan tiga variabel bertipe string bernama nama, nim, dan kelas.

### 5.fmt.Scan(&nama, &nim, &kelas)

- perintah untuk membaca tiga input dari keyboard yang dipisahkan spasi, lalu menyimpannya ke variabel **nama**, **nim**, dan **kelas**.
- Tanda **&** dipakai agar **Scan** bisa menulis langsung ke alamat variabel tersebut

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

- **fmt.Printf** digunakan untuk mencetak output dengan format tertentu.

- Di dalam tanda kutip punya **placeholder %s**, artinya nanti akan diganti dengan nilai variabel string yang diberikan.
- Urutannya sesuai dengan argumen setelah string format:  
**%s** pertama diganti nilai **nama**  
**%s** kedua diganti nilai **kelas**  
**%s** ketiga diganti nilai **nim**
- **\n** di akhir menambahkan baris baru setelah teks ditampilkan.

#### 7. Cara Menjalankannya

1. Simpan kode ke file
2. Buka terminal " Ctrl J "
3. Lalu jalankan
4. Masukkan Input 1: Bima 1124431414 IF-48-GAB
5. Lalu akan muncul di terminal, Output 1 (hasilnya):  
perkenalkan saya adalah Bima, salah satu mahasiswa Prodi S1-IF dari kelas IF-48-GAB dengan NIM 1124431414.
6. Dan masukan soal berikutnya (Input ke 2, Input 2: Yura 1324234545 IFX-48-12)
7. Lalu akan muncul di terminal, Output 2 (hasilnya):  
perkenalkan saya adalah Yura, salah satu mahasiswa Prodi S1-IF dari kelas IFX-48-12 dengan NIM 1324234545.

### 3. Tugas 3

#### Source code

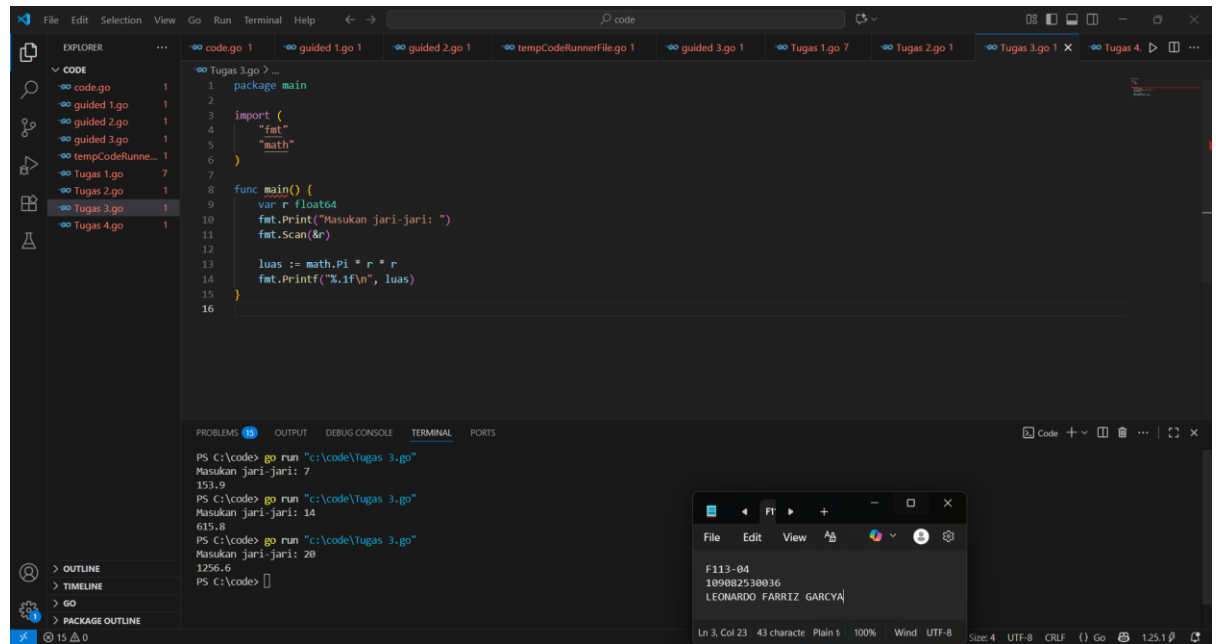
```
package main

import (
    "fmt"
    "math"
)

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

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

## Screenshoot program



## Deskripsi program

### 1. package main

- Program ini adalah program utama yang bisa langsung dijalankan.
- Tanpa main, kode tidak bisa dieksekusi sebagai aplikasi.

### 2. import "fmt"

- Format (fmt), digunakan untuk input/output standar.
- Seperti "fmt.Scanln" : untuk membaca input dari user.  
"fmt.Println" : menampilkan output ke layar.

### 3. import ( "fmt" "math" )

- Mengimpor paket **fmt** (input/output) dan **math** (konstanta dan fungsi matematika). **math** dipakai untuk math.Pi.

### 4. func main () { ... }

- Fungsi **main** adalah untuk titik awal eksekusi program Go.
- Semua intruksi akan berjalan dari fungsi ini.

### 5. var r float64

- Mendeklarasikan variabel r bertipe float64. Nilai awalnya adalah zero value 0.0 sampai diisi.

### 6. fmt.Print("Masukan jari-jari: ")

- Mencetak prompt tanpa newline sehingga input user bisa diketik di baris yang sama.

### 7. fmt.Scan(&r)

- Membaca satu token dari standar input dan menuliskannya ke **r**.
- **&r** adalah alamat variabel (pointer) yang dibutuhkan **Scan**. Jika user mengetik 7 atau 7.5, r akan berisi 7.0 atau 7.5. Jika input tidak valid, Scan mengembalikan error (kode ini tidak memeriksa error).

```
8. luas := math.Pi * r * r
```

- Menghitung luas lingkaran dengan rumus  $\pi \cdot r^2$ .
- **Math.Pi** adalah konstanta  $\pi$  (tipe float64).
- Membuat variabel baru luas dan menyimpannya (tipe inferred float64).
- Operasi **r \* r** mengalikan jari-jari dua kali (lebih cepat dan idiomatik dibanding **math.Pow(r, 2)** untuk pangkat 2).

```
9. fmt.Printf("%.1f\n", luas)
```

- Mencetak nilai luas dengan format satu angka di belakang koma (**%.1f**) lalu newline. Nilai akan dibulatkan sesuai aturan formatting Go.

#### 10. Cara Menjalankannya

1. Simpan kode ke file
2. Buka terminal " Ctrl J "
3. Lalu jalankan
4. Masukan Input 1: 7
5. Lalu akan muncul di terminal, Output 1 (hasilnya): 153.9
6. Masukan Input 2: 14
7. Lalu akan muncul di terminal, Output 2 (hasilnya): 615.8
8. Masukan Input 3: 28
9. Lalu akan muncul di terminal, Output 3 (hasilnya): 1256.6

### 4. Tugas 3

#### Source code

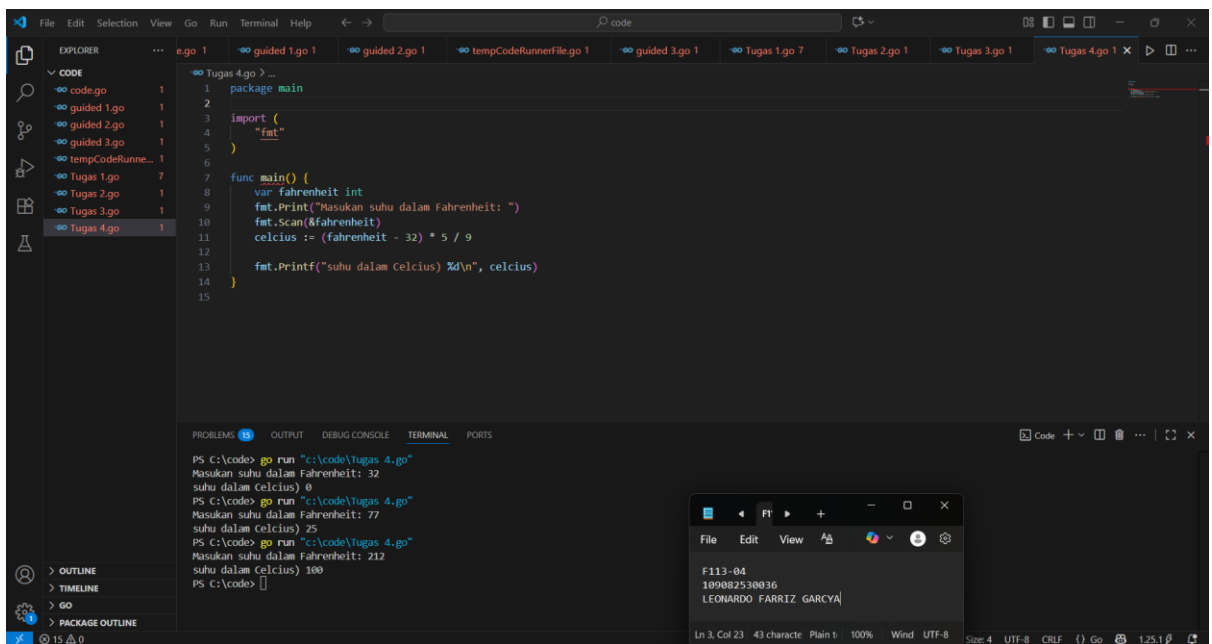
```
package main

import (
    "fmt"
)

func main() {
    var fahrenheit int
    fmt.Print("Masukan suhu dalam Fahrenheit: ")
    fmt.Scan(&fahrenheit)
    celcius := (fahrenheit - 32) * 5 / 9

    fmt.Printf("suhu dalam Celcius) %d\n", celcius)
}
```

## Screenshoot program



## Deskripsi program

### 1.package main

- Program ini adalah program utama yang bisa langsung dijalankan.
- Tanpa main, kode tidak bisa dieksekusi sebagai aplikasi.

### 2.import "fmt"

- Format (fmt), digunakan untuk input/output standar.
- Seperti "fmt.Scanln" : untuk membaca input dari user.  
"fmt.Println" : menampilkan output ke layar.

### 3.func main () { ... }

- Fungsi **main** adalah untuk titik awal eksekusi program Go.
- Semua intruksi akan berjalan dari fungsi ini.

### 3. var fahrenheit int

- Menyatakan variabel fahrenheit bertipe int. Nilai awal (zero value) = 0.

### 4. fmt.Print("Masukan suhu dalam Fahrenheit: ")

- Mencetak prompt tanpa newline sehingga kursor tetap pada baris yang sama dan user dapat mengetik di situ.

### 5. fmt.Scan(&fahrenheit)

- Membaca token dari standar input (keyboard) dan menyimpannya ke **fahrenheit**.
- **&fahrenheit** memberi alamat variabel (pointer) agar **Scan** dapat menulis nilai ke variabel tersebut.



- **fmt.Scan** membaca token yang dipisah whitespace; jika input bukan angka, **Scan** akan gagal – kode saat ini tidak memeriksa error kembalian.

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

- Menghitung konversi Celsius menggunakan rumus matematika **C= (F - 32) \* 5 / 9.**

- Karena semua operand bertipe int, seluruh operasi **menggunakan aritmetika integer**: hasil pembagian dibulatkan ke bawah (truncated).

- Operator dan urutan evaluasi: tanda kurung dievaluasi dulu (**fahrenheit - 32**), kemudian dikalikan **\* 5**, lalu dibagi **/ 9**.

```
7. fmt.Printf("suhu dalam Celcius) %d\n", celcius)
```

- Untuk Mencetak teks format ke layar. **%d** adalah verb untuk mencetak **int**.

- **\n** menambahkan newline. (string mengandung " ) " sebelum **%d** itu hanya karakter biasa dalam output.)

#### 8. Cara Menjalankannya

1. Simpan kode ke file
2. Buka terminal " Ctrl J "
3. Lalu jalankan
4. Masukkan Input 1: 32
5. Lalu akan muncul di terminal, Output 1 (hasilnya): 0
6. Masukkan Input 2: 77
7. Lalu akan muncul di terminal, Output 2 (hasilnya): 25
8. Masukkan Input 3: 212
9. Lalu akan muncul di terminal, Output 3 (hasilnya): 100

## TUGAS PENDAHULUAN

### 1. Jelaskan perbedaan antara

`fmt.Print()`, `fmt.Println()`, `fmt.Printf()` di GO!

- `fmt.Print()` : berfungsi sebagai mencetak teks atau nilai ke layar tanpa baris baru otomatis di akhir. Pada outputnya tidak menambahkan spasi otomatis di antara argumen, kecuali kita menuliskannya sendiri, dan jika kita menginginkan baris baru kita harus menambahkan `\n` secara manual.

- `fmt.Printf()` : mencetak teks dengan format string serta kita juga bisa atur placeholder untuk string, angka, desimal, dan sebagainya. Jika kita menginginkan baris baru harus menambahkan `\n` secara manual, untuk outputnya spasi atau tidaknya antar argumen diatur di format.

- `fmt.Println()` : berfungsi sebagai Mencetak teks atau nilai ke layar dengan spasi otomatis antar argumen dan baris baru otomatis di akhir, dan sama seperti `fmt.Print()` tidak pakai format placeholder.

2. Jelaskan penggunaan tipe data `int`, `float64`, `bool`, dan `string` di Go.

- Integer merupakan tipe data bilangan bulat untuk ukuran bisa 32 bit atau 64 bit tergantung arsitektur, berguna untuk menyimpan angka seperti umur, jumlah barang, dll.

- `Float64` merupakan tipe data pecahan atau decimal dengan presisi 64 bit. Berguna untuk menyimpan nilai pecahan atau decimal seperti, tinggi badan, nilai rata rata, dll.

- `Bool` merupakan tipe data Boolean yang hanya menyimpan `true` dan `false`, kegunaannya seperti logika program, kondisi `if/else`, perbandingan.

- `String` merupakan tipe data teks, kegunaannya seperti menyimpan nama, alamat, dll

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

Dengan penggunaan `var` caranya yaitu, sebutkan `var`, nama variable, tipe data, nilai (opsional).

Dengan deklarasi singkat menggunakan deklarasi + inisialisasi.

Contoh:

- `nama := "ALIP"`
- `umur := 35`

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

- `" = "` digunakan untuk memberikan nilai ke dalam variable, membaca sebagai "diisi dengan" atau "assign".

- `" == "` disebut equality cek karena berfungsi sebagai mengecek apakah dua nilai sama yang nantinya menghasilkan `true/false`.

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

#### Source code

```
package main

import (
    "fmt"
)
```

```
func main() {  
  
    var a, b, c string  
  
    fmt.Print("Masukan nama Anda ")  
    fmt.Scan(&a, &b, &c)  
    fmt.Println("Halo", a, b, c)  
}
```

## Screenshoot program

