

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

**MODUL 3**  
**[NAMA MODUL]**



**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

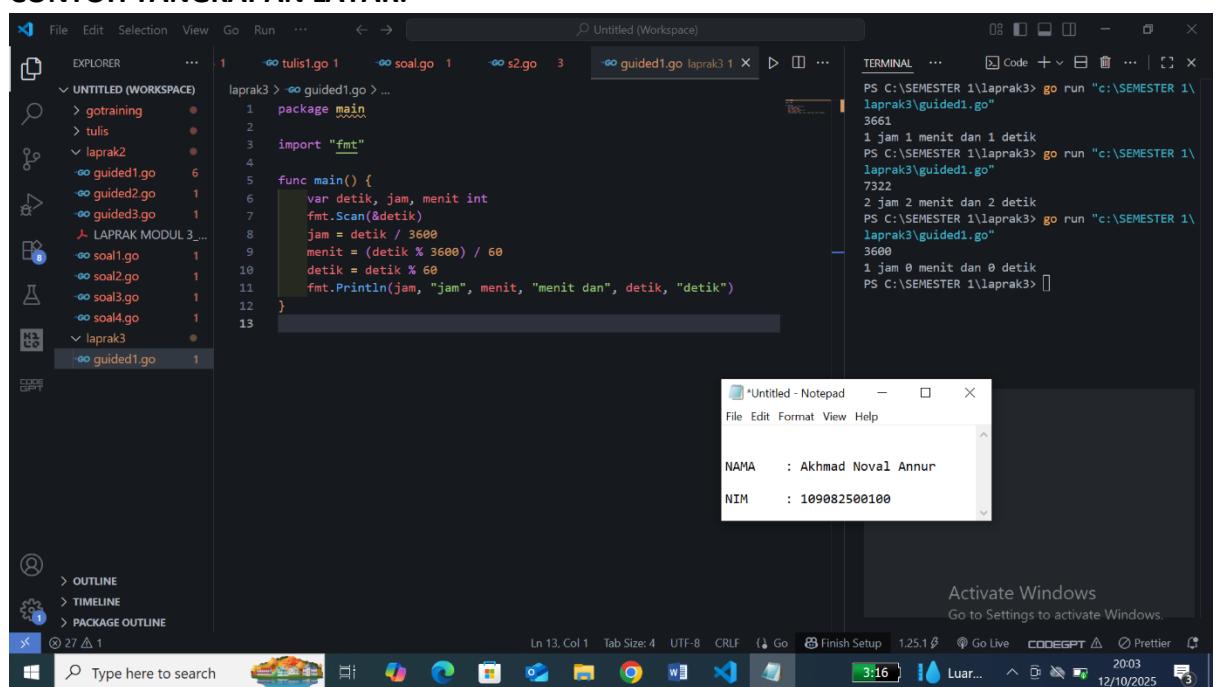
import "fmt"

func main() {
    var detik, jam, menit int
    fmt.Scan(&detik)
    jam = detik / 3600
    menit = (detik % 3600) / 60
    detik = detik % 60
    fmt.Println(jam, "jam", menit, "menit dan", detik,
    "detik")
}
```

#### Screenshot program

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

CONTOH TANGKAPAN LAYAR:



### **Deskripsi program**

Program ini berfungsi untuk mengonversi satuan waktu dari detik menjadi bentuk yang lebih mudah dipahami, yaitu jam, menit, dan detik. Ketika program dijalankan, pengguna diminta memasukkan sebuah bilangan bulat yang mewakili jumlah total detik. Program kemudian memproses input tersebut melalui beberapa tahapan perhitungan. Pertama, program membagi total detik dengan angka 3600 untuk menentukan jumlah jam, karena satu jam terdiri atas 3600 detik. Selanjutnya, sisa pembagian tersebut dihitung kembali untuk menentukan jumlah menit dengan membaginya dengan 60, sebab satu menit terdiri atas 60 detik. Akhirnya, sisa dari operasi tersebut menjadi nilai detik yang tersisa setelah dikonversi ke jam dan menit. Semua hasil perhitungan tersebut disimpan ke dalam variabel terpisah dan ditampilkan ke layar dalam format kalimat seperti “2 jam 30 menit dan 45 detik”. Program ini menunjukkan penerapan konsep integer division dan modulus, serta memperlihatkan bagaimana proses dekomposisi nilai dapat dilakukan secara bertahap untuk menghasilkan representasi waktu yang lebih informatif. Selain itu, program ini memperkuat pemahaman tentang tipe data numerik dan penggunaan operator aritmetika dalam bahasa Go.

## 2. Guided 2

### Source Code

```
package main

import "fmt"

func main() {
    var bilangan, d1, d2, d3 int
    fmt.Scan(&bilangan)
    d1 = bilangan / 100
    d2 = bilangan % 100 / 10
    d3 = bilangan % 100 % 10
    fmt.Println(d1 <= d2 && d2 <= d3)
}
```

### Screenshot program

The screenshot shows a Windows desktop environment. In the center is a Microsoft Visual Studio Code (VS Code) window. The Explorer sidebar on the left shows a workspace named 'UNTITLED (WORKSPACE)' containing files like 'soal.go', 's2.go', 'guided1.go', 'guided2.go' (which contains the provided Go code), and 'guided3.go'. The terminal window on the right shows command-line output from running the program. Below the VS Code window is a Notepad window displaying two lines of text: 'NAMA : Akhmad Noval Annur' and 'NIM : 109082500100'. At the bottom of the screen is the Windows taskbar with various pinned icons.

```
PS C:\SEMESTER 1\laprak3> go run "c:\SEMESTER 1\laprak3\guided2.go"
362
false
PS C:\SEMESTER 1\laprak3> go run "c:\SEMESTER 1\laprak3\guided2.go"
259
true
PS C:\SEMESTER 1\laprak3> go run "c:\SEMESTER 1\laprak3\guided2.go"
189
true
PS C:\SEMESTER 1\laprak3>
```

### Deskripsi program

Program ini digunakan untuk menentukan apakah sebuah bilangan tiga digit tersusun dalam urutan menaik berdasarkan nilai tiap digitnya. Saat dijalankan, pengguna diminta memasukkan satu bilangan bulat tiga digit, misalnya 123. Program kemudian memecah bilangan tersebut menjadi tiga digit terpisah menggunakan operasi pembagian dan modulus.

Digit pertama didapat dengan membagi bilangan dengan 100, digit kedua diperoleh dari sisa bagi 100 kemudian dibagi 10, dan digit ketiga didapat dari sisa pembagian tersebut. Setelah ketiga digit disimpan ke variabel d1, d2, dan d3, program menggunakan ekspresi logika  $d1 \leq d2 \ \&\& \ d2 \leq d3$  untuk memeriksa apakah digit-digit tersebut menaik atau setidaknya tidak menurun. Jika kondisi ini terpenuhi, maka program mencetak nilai true, sedangkan jika tidak, hasilnya adalah false. Logika AND ( $\&\&$ ) memastikan kedua kondisi harus benar agar hasil akhir bernilai benar. Program ini menekankan pemahaman terhadap konsep penguraian bilangan menjadi digit-digit penyusunnya, penggunaan operator pembanding, serta penerapan logika boolean untuk menentukan kebenaran suatu pernyataan. Melalui program ini, pengguna dapat memahami bagaimana memecah masalah sederhana menjadi langkah-langkah logis dalam algoritma.

### 3. Guided 3

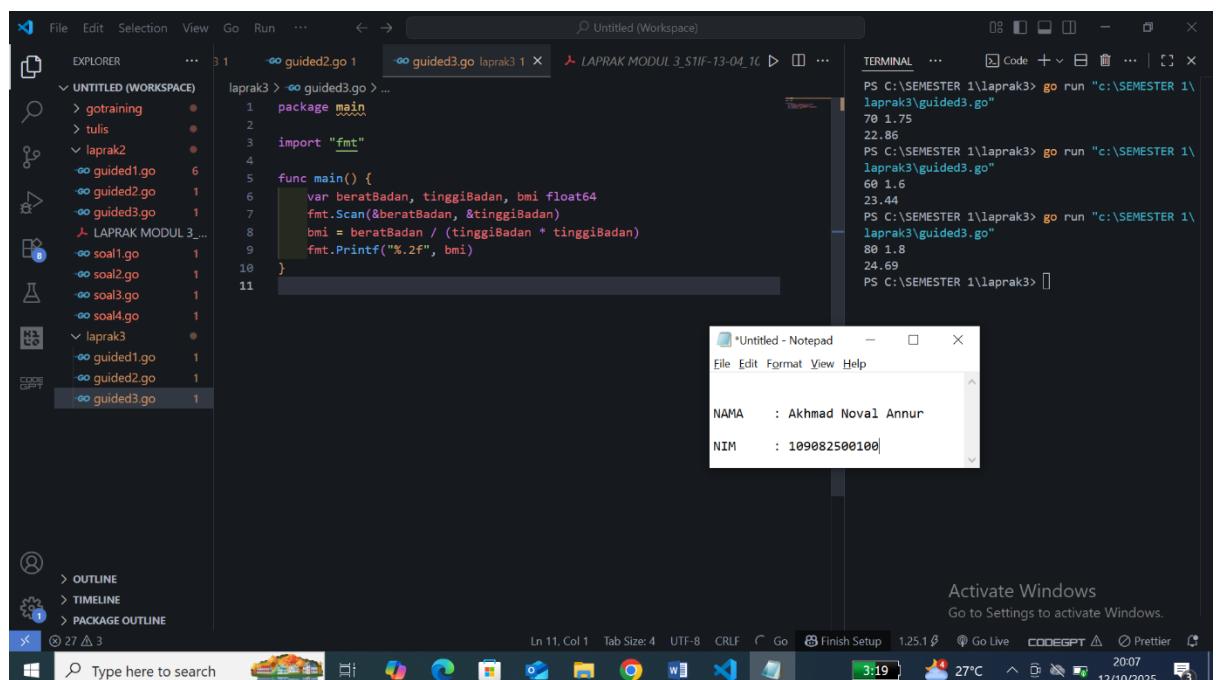
#### Source Code

```
package main

import "fmt"

func main() {
    var beratBadan, tinggiBadan, bmi float64
    fmt.Scan(&beratBadan, &tinggiBadan)
    bmi = beratBadan / (tinggiBadan * tinggiBadan)
    fmt.Printf("%.2f", bmi)
}
```

#### Screenshot program



#### Deskripsi program

Program ini berfungsi untuk menghitung nilai Body Mass Index (BMI) seseorang berdasarkan berat dan tinggi badan yang dimasukkan oleh pengguna. Saat program dijalankan, pengguna diminta memasukkan berat badan dalam kilogram dan tinggi badan dalam meter. Setelah itu, program menghitung nilai BMI dengan rumus berat badan dibagi kuadrat tinggi badan. Hasil perhitungan kemudian ditampilkan ke layar dengan dua angka di belakang koma agar lebih rapi dan mudah dibaca. Program ini membantu memahami bagaimana cara

menggunakan operasi matematika dalam pemrograman untuk menyelesaikan perhitungan sederhana seperti menentukan nilai BMI.

# TUGAS

## 1. Tugas 1

### Source code

```
package main

import "fmt"

func main() {
    var h,d int

    fmt.Print("Masukkan Total Belanja: ")
    fmt.Scan(&h)

    fmt.Print("Masukkan Diskon(dalam persen): ")
    fmt.Scan(&d)

    d = h * d/100

    akhir := h - d

    fmt.Printf("Harga setelah diskon adalah %d",akhir)
}
```

### Screenshot program

The screenshot shows a Windows desktop environment with several windows open:

- TERMINAL**: A terminal window titled "Untitled (Workspace)" showing Go code and its execution output. The output includes:

```
PS C:\SEMESTER 1\laprak3> go run "c:\SEMESTER 1\laprak3\tugas1.go"
Masukkan Total Belanja: 100000
Masukkan Diskon(dalam persen): 10
Harga setelah diskon adalah 90000
PS C:\SEMESTER 1\laprak3> go run "c:\SEMESTER 1\laprak3\tugas1.go"
Masukkan Total Belanja: 200000
Masukkan Diskon(dalam persen): 20
Harga setelah diskon adalah 160000
PS C:\SEMESTER 1\laprak3> go run "c:\SEMESTER 1\laprak3\tugas1.go"
Masukkan Total Belanja: 150000
Masukkan Diskon(dalam persen): 15
Harga setelah diskon adalah 127500
PS C:\SEMESTER 1\laprak3>
```
- Code Editor**: An IDE window titled "Untitled (Workspace)" showing the Go source code for "tugas1.go". The code is identical to the one shown in the terminal.
- Notepad**: A window titled "Untitled - Notepad" containing student information:

NAMA :	Akhmad Noval Annur
NIM :	109082500100

### Deskripsi program

Program ini berfungsi untuk menghitung harga akhir suatu pembelian setelah diterapkan diskon dalam bentuk persentase. Pertama, program menampilkan pesan agar pengguna memasukkan total harga belanja dan besar diskon yang diberikan. Nilai total belanja dan diskon tersebut disimpan dalam variabel bertipe integer. Program kemudian menghitung besarnya potongan harga dengan mengalikan total belanja dengan nilai diskon, lalu membaginya dengan 100 agar hasilnya sesuai dengan persentase yang diberikan. Setelah nilai potongan diperoleh, program menghitung harga akhir dengan mengurangkan total harga awal dan nilai potongan tersebut. Hasil akhirnya ditampilkan ke layar dalam bentuk kalimat yang menunjukkan harga yang harus dibayar oleh pembeli. Program ini menekankan konsep input-output handling, operasi aritmetika dasar seperti perkalian, pembagian, dan pengurangan, serta pentingnya urutan eksekusi dalam penyelesaian masalah. Selain itu, program ini juga mencerminkan bagaimana logika perhitungan diskon digunakan dalam transaksi komersial seperti di aplikasi kasir atau e-commerce.

## 2. Tugas 2

### Source code

```
package main

import "fmt"

func main() {
    var n,tb float64

    fmt.Print("Masukkan Nilai BMI: ")
    fmt.Scan(&n)

    fmt.Print("Masukkan Tinggi Badan(dalam meter): ")
    fmt.Scan(&tb)

    bb := n * (tb*tb)

    fmt.Printf("Berat Badan: %.0f",bb)
}
```

### Screenshot program

The screenshot shows a Windows desktop environment with a Visual Studio Code (VS Code) instance open. The VS Code interface includes:

- EXPLORER** sidebar showing files and folders: `guided3.go`, `tugas1.go`, `tugas2.go`, `tugas3.go`, `laprak3` folder containing `guided1.go`, `guided2.go`, `guided3.go`, `tugas1.go`, `tugas2.go`, and `tugas3.go`.
- TERMINAL** tab showing command-line output:

```
PS C:\SEMESTER 1\laprak3> go run "c:\SEMESTER 1\laprak3\tugas2.go"
Masukkan Nilai BMI: 22.85 1.75
Masukkan Tinggi Badan(dalam meter): Berat Badan: 70
PS C:\SEMESTER 1\laprak3> go run "c:\SEMESTER 1\laprak3\tugas2.go"
Masukkan Nilai BMI: 23.43 1.6
Masukkan Tinggi Badan(dalam meter): Berat Badan: 60
PS C:\SEMESTER 1\laprak3> go run "c:\SEMESTER 1\laprak3\tugas2.go"
Masukkan Nilai BMI: 24.69 1.8
Masukkan Tinggi Badan(dalam meter): Berat Badan: 80
PS C:\SEMESTER 1\laprak3>
```

A separate Notepad window titled "Untitled - Notepad" displays student information:

NAMA :	Akhmad Noval Annur
NIM :	109082500100

The taskbar at the bottom shows various pinned icons including File Explorer, Edge browser, FileZilla, and others.

### **Deskripsi program**

Program ini digunakan untuk menghitung berat badan seseorang berdasarkan nilai BMI (Body Mass Index) dan tinggi badan. Program meminta pengguna memasukkan nilai BMI dan tinggi badan dalam meter. Setelah kedua data diterima, program melakukan perhitungan berat badan menggunakan rumus Berat Badan= BMI x (Tinggi Badan). Tinggi badan dikalikan dengan dirinya sendiri untuk menghasilkan nilai kuadrat, lalu dikalikan dengan nilai BMI yang dimasukkan pengguna. Hasil perhitungan disimpan dalam variabel bertipe float64 untuk menjaga ketelitian nilai desimal. Program kemudian menampilkan hasilnya dengan format dua angka di belakang koma agar mudah dibaca. Program ini menggambarkan penerapan perhitungan matematis yang umum digunakan dalam bidang kesehatan dan memperkenalkan cara menerjemahkan formula ke dalam bentuk algoritmik. Selain itu, program ini menunjukkan pemanfaatan tipe data numerik, operator aritmetika, dan format keluaran yang terstruktur dalam bahasa Go.

### 3. Tugas 3

#### Source code

```
package main

import (
    "fmt"
    "math"
)

func main() {
    var ax, ay float64
    var bx, by float64
    var cx, cy float64

    fmt.Print("Masukkan x dan y untuk titik A: ")
    fmt.Scan(&ax, &ay)
    fmt.Print("Masukkan x dan y untuk titik B: ")
    fmt.Scan(&bx, &by)
    fmt.Print("Masukkan x dan y untuk titik C: ")
    fmt.Scan(&cx, &cy)

    ab := math.Sqrt(math.Pow(bx-ax, 2) + math.Pow(by-
ay, 2))
    bc := math.Sqrt(math.Pow(cx-bx, 2) + math.Pow(cy-
by, 2))
    ca := math.Sqrt(math.Pow(ax-cx, 2) + math.Pow(ay-
cy, 2))

    sisitp := ab
    if bc > sisitp {
        sisitp = bc
    }
    if ca > sisitp {
        sisitp = ca
    }

    fmt.Printf("%.2f\n", sisitp)
}
```

#### Screenshot program

The screenshot shows a Windows desktop environment with a terminal window open in the background. The terminal window displays the command "PS C:\SEMESTER 1\laprak3> go run "c:\SEMESTER 1\laprak3\tugas3.go"" and its output, which includes prompts for inputting coordinates for points A, B, and C, and the resulting calculations for distances AB, BC, and CA.

The foreground shows a code editor window titled "Untitled (Workspace)" containing a Go program named "tugas3.go". The code imports the "math" package and defines a main function that reads coordinates from the user and prints the longest side of the triangle formed by those points. A Notepad window is also visible, showing personal information.

## Deskripsi program

Program ini dibuat untuk menentukan sisi terpanjang dari sebuah segitiga berdasarkan tiga titik yang dimasukkan oleh pengguna. Program meminta pengguna untuk mengisi nilai koordinat x dan y pada masing-masing titik A, B, dan C. Setelah semua data dimasukkan, program memproses nilai-nilai tersebut untuk menghitung jarak antar titik dan menentukan panjang setiap sisi segitiga.

Setelah ketiga sisi diketahui, program membandingkannya untuk menemukan sisi mana yang memiliki panjang paling besar. Hasil akhirnya berupa nilai sisi terpanjang yang kemudian ditampilkan kepada pengguna. Program ini memperlihatkan bagaimana logika perbandingan dan pemrosesan data dapat digunakan untuk menyelesaikan persoalan yang berkaitan dengan bentuk geometri. Selain itu, program ini juga mengajarkan cara memanfaatkan fungsi-fungsi matematika dalam bahasa Go untuk membantu menyelesaikan permasalahan dengan cara yang sistematis dan efisien.

## TUGAS PENDAHULUAN

### 1. Soal 1

**Pemberi soal :** Ichsan Maulana Muhammad-109082500093

Sebuah program digunakan untuk menghitung total harga setelah mendapatkan diskon dengan persentase tertentu.

Masukan:

Terdiri dari dua baris.

- Baris pertama berisi bilangan bulat yang menyatakan total belanja.
- Baris kedua berisi bilangan bulat yang menyatakan besar diskon dalam persen.

Keluaran:

Satu baris berisi total harga yang harus dibayar setelah mendapatkan diskon.

#### Source code

```
package main

import "fmt"

func main() {
    var n1, n2, n3 float64
    fmt.Print("Masukkan nilai 1: ")
    fmt.Scan(&n1)
    fmt.Print("Masukkan nilai 2: ")
    fmt.Scan(&n2)
    fmt.Print("Masukkan nilai 3: ")
    fmt.Scan(&n3)

    rata := (n1 + n2 + n3) / 3
    fmt.Printf("Rata-rata: %.2f\n", rata)

    if rata >= 60 {
        fmt.Println("Keterangan: Lulus")
    } else {
        fmt.Println("Keterangan: Tidak Lulus")
    }
}
```

#### Screenshot program

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows a tree view of files under "UNTITLED (WORKSPACE)".
  - laprak3 > soal1.go
  - laprak3 > soal2.go
  - laprak3 > soal3.go
  - laprak3 > tugas1.go
  - laprak3 > tugas2.go
  - laprak3 > tugas3.go
- Editor:** The main editor pane displays the content of the "soal1.go" file.

```
package main

import "fmt"

func main() {
    var n1, n2, n3 float64
    fmt.Print("Masukkan nilai 1: ")
    fmt.Scan(&n1)
    fmt.Print("Masukkan nilai 2: ")
    fmt.Scan(&n2)
    fmt.Print("Masukkan nilai 3: ")
    fmt.Scan(&n3)

    rata := (n1 + n2 + n3) / 3
    fmt.Printf("Rata-rata: %.2f\n", rata)

    if rata >= 60 {
        fmt.Println("Keterangan: Lulus")
    } else {
        fmt.Println("Keterangan: Tidak Lulus")
    }
}
```
- Terminal:** The terminal pane on the right shows the output of running the "soal1.go" file.

```
PS C:\SEMESTER 1\laprak3> go run "c:\SEMESTER 1\laprak3\soal1.go"
Masukkan nilai 1: 80
Masukkan nilai 2: 75
Masukkan nilai 3: 90
Rata-rata: 81.67
Keterangan: Lulus
PS C:\SEMESTER 1\laprak3> go run "c:\SEMESTER 1\laprak3\soal1.go"
Masukkan nilai 1: 50
Masukkan nilai 2: 60
Masukkan nilai 3: 55
Rata-rata: 55.00
Keterangan: Tidak Lulus
PS C:\SEMESTER 1\laprak3>
```
- Bottom Status Bar:** Shows the current file path (LAPRAK MODUL 3\_S1IF-13), line number (Ln 22, Col 2), tab size (Tab Size: 4), encoding (UTF-8), line endings (CRLF), search status (Go), file status (Finish Setup), and system status (1.25.1, Go Live, CODEGPT, Prettier).

## Deskripsi program

Program ini dibuat untuk menghitung nilai rata-rata dari tiga mata pelajaran dan menentukan apakah seorang siswa dinyatakan lulus atau tidak. Ketika program dijalankan, pengguna diminta memasukkan tiga nilai ujian. Setelah nilai dimasukkan, program menjumlahkan ketiganya lalu menghitung rata-rata dengan membaginya menjadi tiga bagian yang sama. Hasil rata-rata tersebut ditampilkan dengan dua angka di belakang koma agar lebih rapi. Selanjutnya, program menggunakan pernyataan if-else untuk memberikan keterangan kelulusan berdasarkan nilai rata-rata yang diperoleh. Jika rata-rata sama dengan atau lebih dari 60, maka siswa dinyatakan lulus, sedangkan jika kurang dari 60, dinyatakan tidak lulus. Program ini membantu memahami penggunaan operasi aritmetika, logika percabangan, serta format tampilan hasil dalam pemrograman Go.

## 2.Soal 2

**Pemberi soal :** Ichsan Maulana Muhammad-109082500093

Buatlah program untuk menghitung konsumsi air harian seseorang berdasarkan berat badan yang dimasukkan. Setiap 1 kilogram berat badan membutuhkan 35 ml air per hari.

Masukan:

Satu bilangan bulat yang menyatakan berat badan seseorang dalam kilogram.

Keluaran:

Satu bilangan yang menunjukkan jumlah air yang perlu dikonsumsi per hari dalam satuan mililiter.

### Source code

```
package main

import "fmt"

func main() {
    var berat, air int
    fmt.Print("Masukkan berat badan (kg) : ")
    fmt.Scan(&berat)

    air = berat * 35
    fmt.Printf("Kebutuhan air harian anda adalah: %d\n", air)
}
```

### Screenshoot program

The screenshot shows a VS Code interface with the following details:

- File Explorer:** Shows a tree view of files under "UNTITLED (WORKSPACE)".
  - laprak3 > **soal2.go**
  - laprak3 > **soal2.go** > ...
  - laprak3 > **soal1.go** laprak3 1
  - laprak3 > **soal2.go** laprak3 1 x
- Code Editor:** Displays the content of **soal2.go**. The code calculates daily water needs based on body weight.

```
package main

import "fmt"

func main() {
    var berat, air int
    fmt.Print("Masukkan berat badan (kg): ")
    fmt.Scan(&berat)

    air = berat * 35
    fmt.Printf("Kebutuhan air harian anda adalah: %d ml\n", air)
}
```
- Terminal:** Shows command-line output for running the code and calculating water needs.
- Bottom Status Bar:** Shows file path (C:\SEMESTER 1\laprak3\soal2.go), line 13, column 0, tab size 4, encoding UTF-8, CRLF, Go Live, CODEGPT, Prettier, and system status (1:56, 27°C, 12/10/2025).

## Deskripsi program

Program ini digunakan untuk menghitung kebutuhan air harian seseorang berdasarkan berat badannya. Ketika program dijalankan, pengguna diminta memasukkan berat badan dalam satuan kilogram. Nilai tersebut kemudian diproses untuk menentukan jumlah air yang harus diminum per hari. Program menggunakan logika sederhana dengan mengalikan berat badan dengan angka 35 untuk mendapatkan hasil dalam satuan mililiter. Nilai akhir ditampilkan ke layar dengan kalimat informatif yang menunjukkan total konsumsi air per hari. Program ini membantu memahami konsep dasar pemrosesan input pengguna, operasi aritmetika sederhana, serta cara menampilkan hasil perhitungan secara terformat dalam bahasa Go.