**LAPORAN PRAKTIKUM ALGORITMA**

**DAN PEMROGRAMAN 1**

**MODUL [No.4]**

**[Tipe data & variabel]**

**Sebuah gambar berisi logo, teks, simbol, Grafis

Konten yang dihasilkan AI mungkin salah.**

**Disusun oleh:**

**[Denis Ramadhani]**

**[109082500060]**

**S1IF-13-[02]**

**Asisten Praktikum**

Adithana dharma putra

Alfin Ilham Berlianto

**PROGRAM STUDI S1 INFORMATIKA**

**FAKULTAS INFORMATIKA**

**TELKOM UNIVERSITY PURWOKERTO**

**2025**

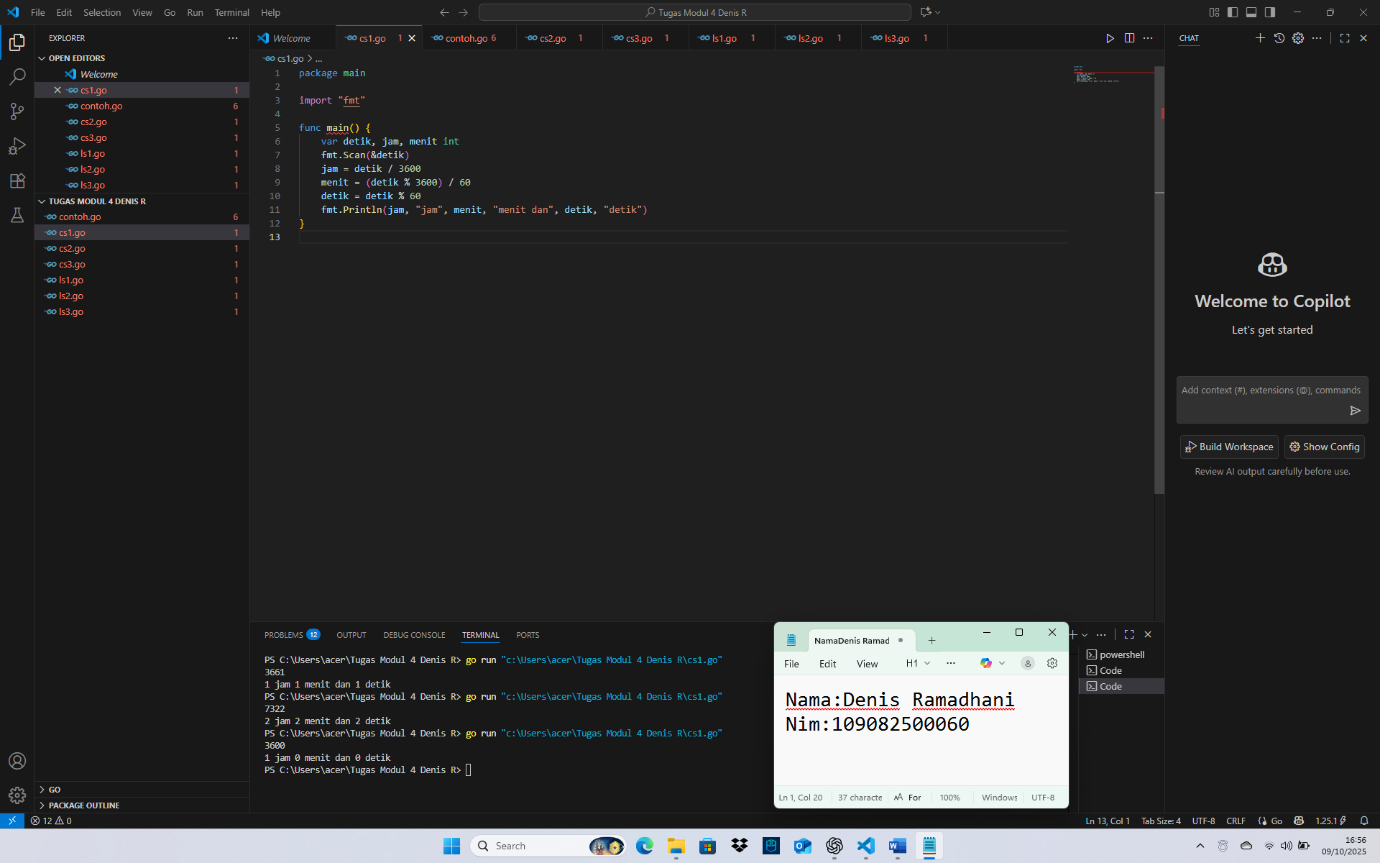
**LATIHAN KELAS – GUIDED**

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")  } |

**Screenshoot program**

****

**Deskripsi program**

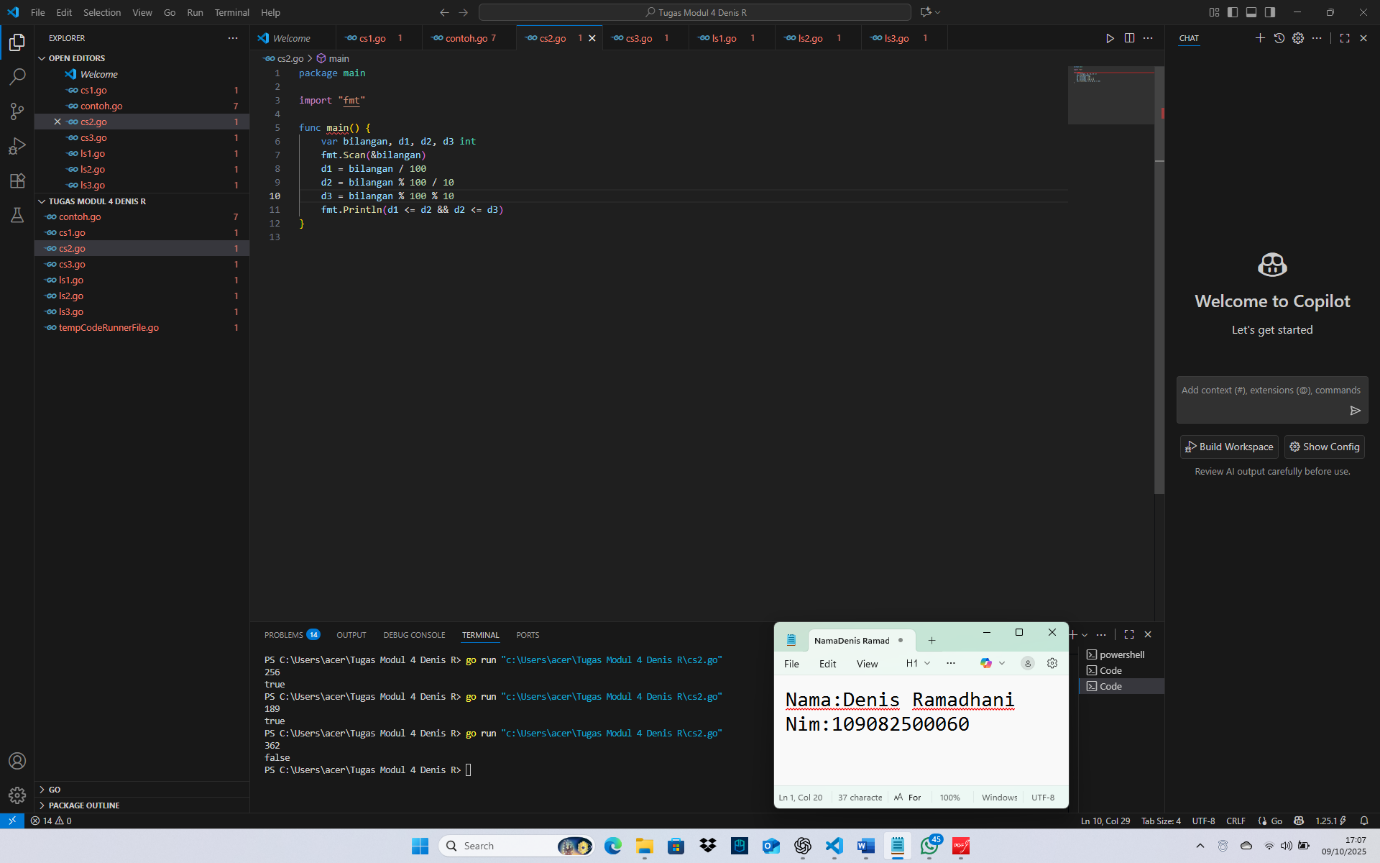
Kode itu buat ngubah detik jadi jam, menit, sama detik. Jadi pertama program minta input jumlah detik dari user lewat fmt.Scan. Habis itu dia bagi detik sama 3600 buat dapetin jam, terus sisa dari bagi itu dipakai buat hitung menit dengan (detik % 3600) / 60. Sisanya lagi dipakai buat detik terakhir pakai detik % 60. Terakhir, hasilnya ditampilin pake fmt.Println biar keluar kayak “1 jam 1 menit dan 1 detik

1. **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)  } |

**Screenshoot program**

****

**Deskripsi program**

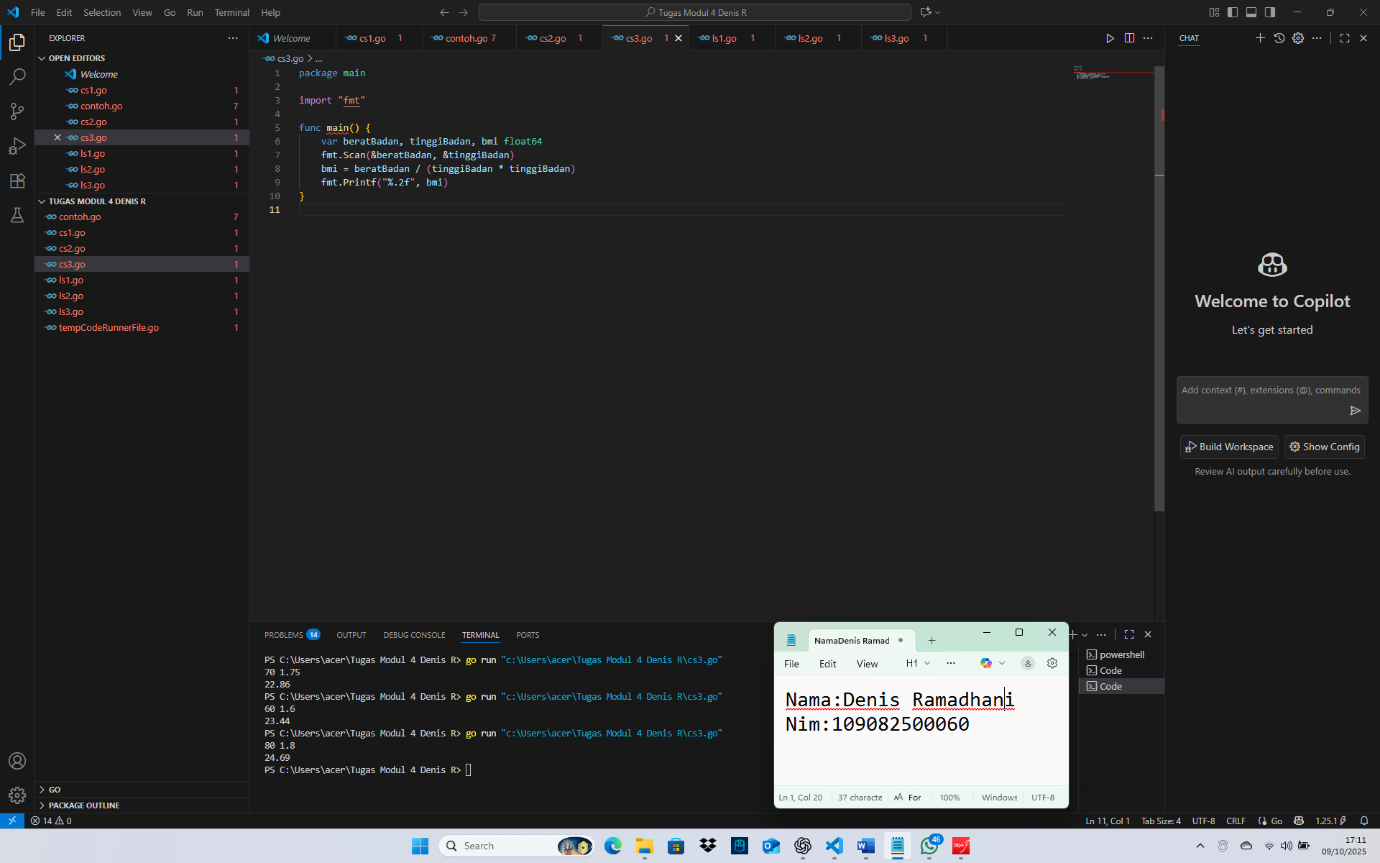
**Kode di gambar itu fungsinya buat ngecek apakah tiga digit angka berurutan dari kecil ke besar. Jadi pertama user masukin angka tiga digit lewat fmt.Scan. Terus d1, d2, dan d3 diambil dari masing-masing digitnya — d1 digit ratusan, d2 puluhan, d3 satuan. Caranya pakai operasi bagi dan sisa bagi (/ dan %). Setelah itu dicek kondisinya di fmt.Println(d1 < d2 && d2 < d3), artinya bakal tampil true kalau urutan digitnya naik (misal 123), dan false kalau enggak (misal 362).**

1. **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)  } |

**Screenshoot program**

****

**Kode di gambar itu buat ngitung BMI (Body Mass Index) dari berat dan tinggi badan. Pertama, program nyiapin tiga variabel beratBadan, tinggiBadan, dan bmi yang tipenya float64 biar bisa nyimpen angka desimal. Terus pakai fmt.Scan buat baca input berat dan tinggi dari user. Rumus BMI-nya ada di baris bmi = beratBadan / (tinggiBadan \* tinggiBadan), yaitu berat badan dibagi tinggi badan kuadrat (tingginya dalam meter). Hasil akhirnya ditampilin pakai fmt.Printf("%.2f", bmi) biar keluar dua angka di belakang koma.**

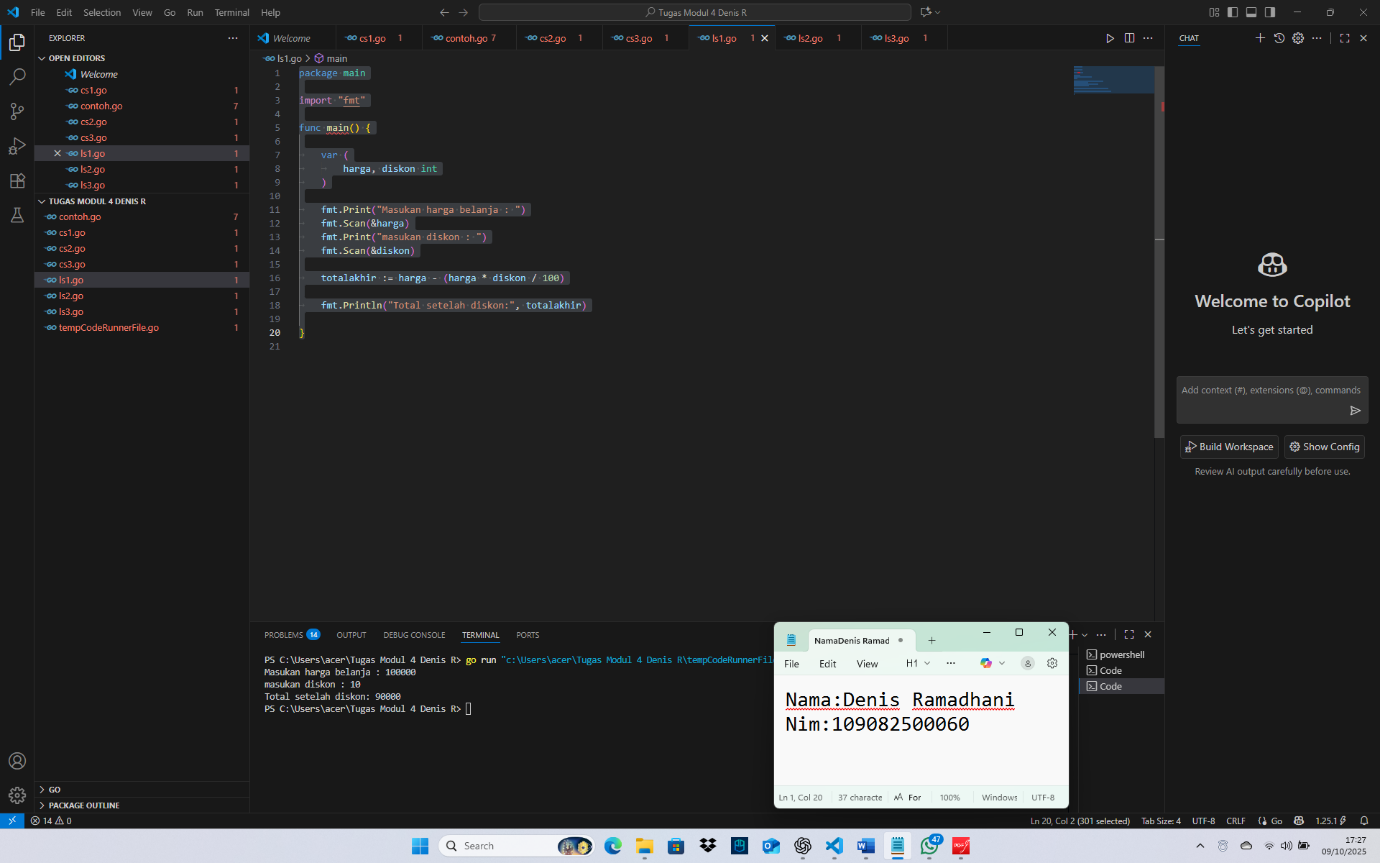
**TUGAS**

1. **Tugas 1**

**Source code**

|  |
| --- |
| package main  import "fmt"  func main() {      var (          harga, diskon int      )      fmt.Print("Masukan harga belanja : ")      fmt.Scan(&harga)      fmt.Print("masukan diskon : ")      fmt.Scan(&diskon)      totalakhir := harga - (harga \* diskon / 100)      fmt.Println("Total setelah diskon:", totalakhir)  } |

**Screenshoot program**

****

**Deskripsi program**

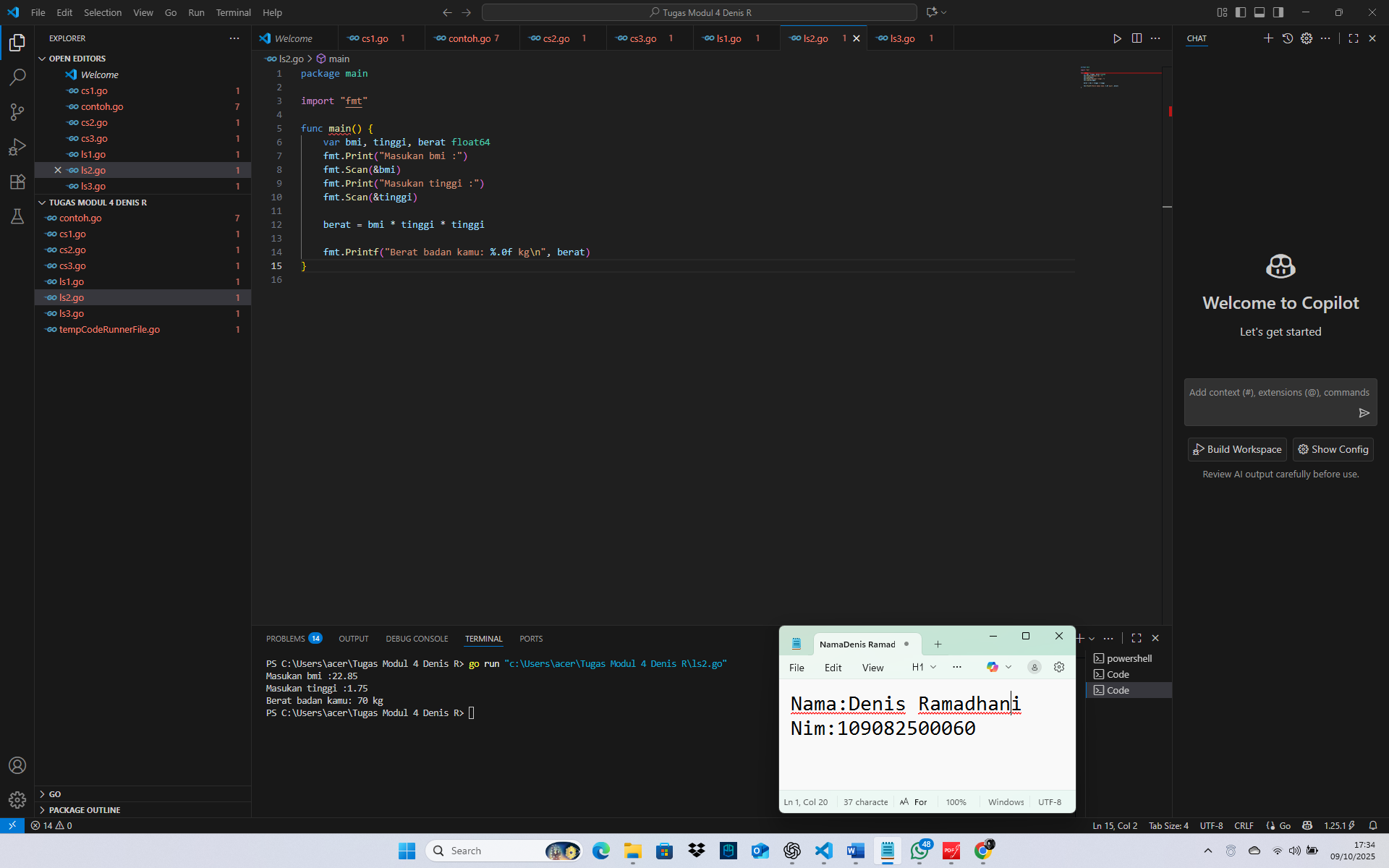
**Kode gambar itu buat ngitung total harga setelah dikasih diskon. Jadi pertama program nyiapin variabel harga dan diskon bertipe int. Terus pakai fmt.Scan buat input harga belanja dan persentase diskon dari user. Setelah itu rumusnya totalakhir = harga - (harga \* diskon / 100) buat ngurangin harga awal sama potongan diskonnya. Terakhir hasilnya ditampilin lewat fmt.Println("Total setelah diskon:", totalakhir). Jadi misal harga 100000 dan diskon 10, hasilnya jadi 90000.**

1. **Tugas 2**

**Source code**

|  |
| --- |
| package main  import "fmt"  func main() {      var bmi, tinggi, berat float64      fmt.Print("Masukan bmi :")      fmt.Scan(&bmi)      fmt.Print("Masukan tinggi :")      fmt.Scan(&tinggi)      berat = bmi \* tinggi \* tinggi      fmt.Printf("Berat badan kamu: %.0f kg\n", berat)  } |

**Screenshoot program**



**Deskripsi program**

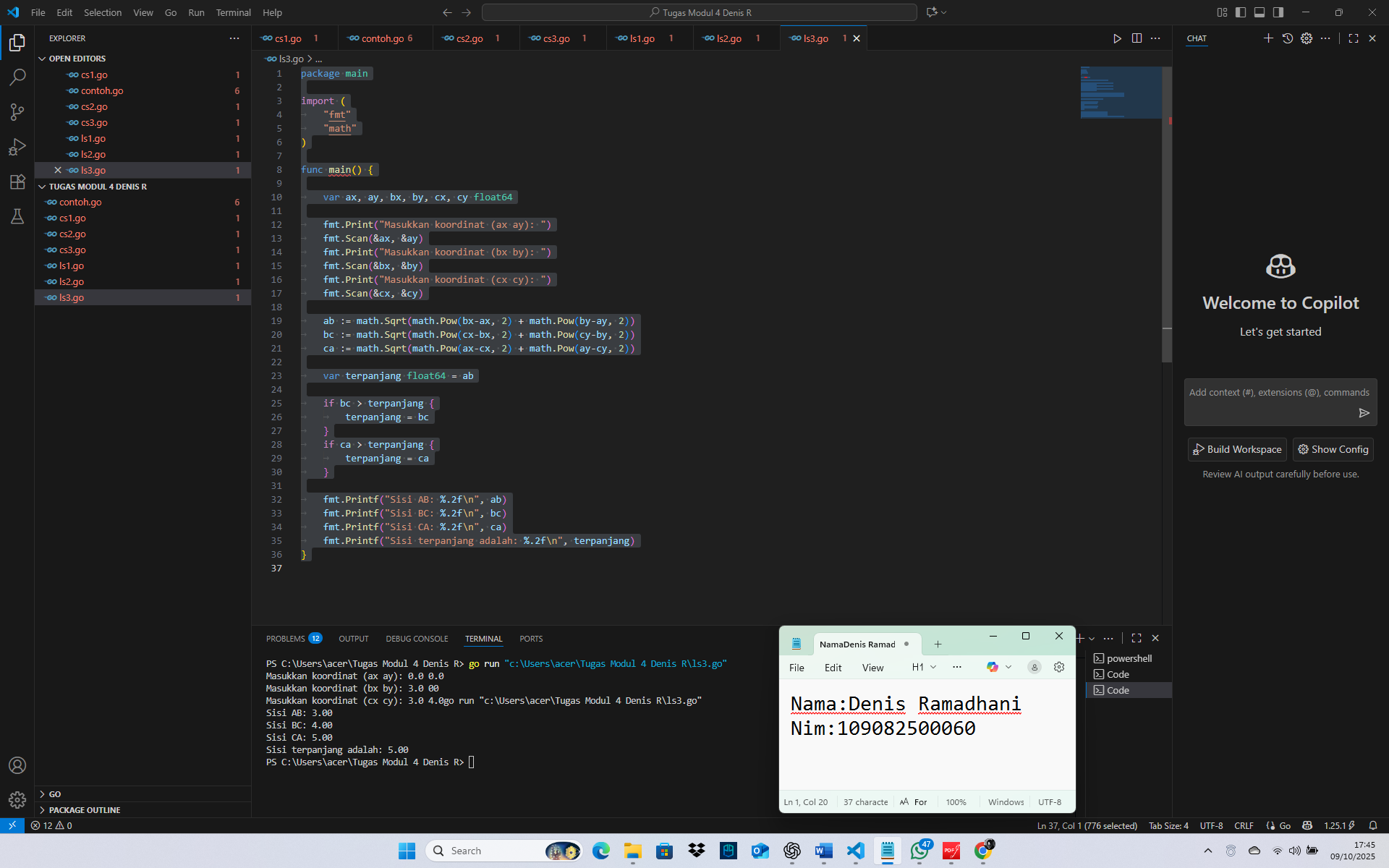
**Jadi, pertama program minta input nilai bmi dan tinggi lewat fmt.Scan. Kedua variabel ini bertipe float64 biar bisa nyimpen angka desimal. Rumusnya berat = bmi \* tinggi \* tinggi, karena dari rumus asli BMI (berat / tinggi²), kalau dibalik jadi berat = BMI × tinggi². Terakhir hasilnya ditampilin pakai fmt.Printf biar keluar rapi dua angka di belakang koma, misalnya Berat badan kamu: 70.00 kg**

1. **Tugas 3**

**Source code**

|  |
| --- |
| package main  import (      "fmt"      "math"  )  func main() {      var ax, ay, bx, by, cx, cy float64      fmt.Print("Masukkan koordinat (ax ay): ")      fmt.Scan(&ax, &ay)      fmt.Print("Masukkan koordinat (bx by): ")      fmt.Scan(&bx, &by)      fmt.Print("Masukkan koordinat (cx cy): ")      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))      var terpanjang float64 = ab      if bc > terpanjang {          terpanjang = bc      }      if ca > terpanjang {          terpanjang = ca      }      fmt.Printf("Sisi AB: %.2f\n", ab)      fmt.Printf("Sisi BC: %.2f\n", bc)      fmt.Printf("Sisi CA: %.2f\n", ca)      fmt.Printf("Sisi terpanjang adalah: %.2f\n", terpanjang)  } |

**Screenshoot program**



**Deskripsi program**

**program minta input koordinat A, B, dan C (masing-masing punya x dan y). Nilai itu disimpan di variabel ax, ay, bx, by, cx, cy. Lalu pakai rumus jarak dua titik (sqrt((x2 - x1)² + (y2 - y1)²)) buat ngitung tiga sisi: AB, BC, dan CA.  
Setelah itu dicek satu-satu: kalau bc lebih besar dari ab, berarti sisi terpanjangnya sementara adalah BC. Kalau ca lebih besar lagi, ganti jadi CA.  
Terakhir, program nampilin semua panjang sisi dengan dua angka di belakang koma dan nyebutin sisi yang paling panjang.**

**Hasil di terminal juga udah bener — semua sisi ditampilin dan yang paling panjang ketemu (5.00).**