**LAPORAN PRAKTIKUM**

**ALGORITMA DAN PEMROGRAMAN 1**

**MODUL 3**

**“TIPE DATA & VARIABEL”**



**DISUSUN OLEH:**

**Keishin Naufa Alfaridzhi**

**103112400061**

**S1 IF-12-01**

**DOSEN:**

**Yohani Setiya Rafika Nur, M. Kom.**

**PROGRAM STUDI S1 TEKNIK INFORMATIKA**

**FAKULTAS INFORMATIKA**

**TELKOM UNIVERSITY PURWOKERTO**

**2024/2025**

**DASAR TEORI**

1. **Bahasa Yang Digunakan**

Pada praktikum ini bahasa pemrograman yang digunakan adalah bahasa pemrograman Go, sesuai dengan modul yang menjadi acuan praktikum. Golang (atau) Go adalah bahasa pemrograman baru, yang mulai dilirik oleh para developer karena kelebihan-kelebihan yang dimilikinya. Sudah banyak Perusahaan besar yang menggunakan bahasa ini untuk produk-produk mereka hingga di level production.

1. **Komentar**

Komentar biasa dimanfaatkan untuk menyisipkan catatan pada kode program, menulis penjelasan atau deskripsi mengenai suatu blok kode, atau bisa juga digunakan untuk me-remark kode (men-non-aktifkan kode yang tidak digunakan). Komentar akan diabaikan Ketika kompilasi maupun eksekusi program.

Ada 2 jenis komentar di Golang, yaitu inline dan multiline.

1. **Komentar Inline**

Penulisan komentar jenis ini diawali dengan tanda *double slash* (//) lalu diikuti pesan komentarnya. Komentar inline hanya berlaku untuk satu baris pesan saja. Jika pesan komentar lebih dari satu baris, maka tanda *double slash* harus ditulis lagi di baris selanjutnya.

1. **Komentar Multiline**

Komentar yang cukup panjang akan lebih rapi jika ditulis menggunakan teknik komentar multiline. Ciri dari komentar jenis ini adalah penulisannya diawali dengan tanda (/\*) dan diakhiri (\*/).

1. **Variabel**

Golang mengadopsi 2 jenis penulisan variabel, yang dituliskan tipe data-nya dan yang tidak. Kedua cara tersebut intinya adalah sama, pembedanya hanyalah cara penulisannya saja. Untuk penulisan variabel dengan tipe data, keyword *var* digunakan untuk deklarasi variabel kemudian diakhiri dengan tipe data misalnya *string*. Kemudian untuk penulisan variabel tanpa tipe data, variabel dideklarasikan dengan menggunakan metode type inference. Penandanya tipe data tidak dituliskan pada saat deklarasi. Pada penggunaan metode ini, operand (=) harus diganti dengan (:=) dan keyword *var* dihilangkan.

Golang memiliki aturan unik yang tidak dimiliki bahasa lain, yaitu tidak boleh ada satupun variabel yang menganggur. Artinya, semua variabel yang dideklarasikan harus digunakan. Jika terdapat variabel yang tidak digunakan tapi dideklarasikan, program akan gagal dikompilasi. Untuk mengatasi itu, golang memiliki variabel yaitu underscore. Underscore ( \_ ) adalah predefined variabel yang bisa dimanfaatkan untuk menampung nilai yang tidak dipakai.

1. **Tipe Data**

Golang mengenal beberapa jenis tipe data, diantaranya adalah tipe data numerik (decimal dan non-desimal), string, dan boolean.

1. Tipe Data Numerik Non-Desimal (uint, int)
2. Tipe Data Numerik Desimal (float64, float32)
3. Tipe Data Bool (true, false)
4. Tipe Data String (string, “ “)
5. **Operator Aritmatika**

Operator aritmatika merupakan operator yang digunakan untuk operasi yang sifatnya perhitungan. Golang mendukung beberapa operator aritmatika standar, yaitu:

1. Penjumlahan (+)
2. Pengurangan (-)
3. Perkalian (\*)
4. Pembagian (/)
5. Modulus atau sisa hasil pembagian (%)
6. **Seleksi Kondisi**

Seleksi kondisi pada program berguna untuk mengontrol sebuah blok kode yang akan dieksekusi. Yang dijadikan acuan oleh selksi kondisi adalah nilai bertipe bool, bisa berasal dari variabel, ataupun hasil operasi perbandingan. Nilai tersebut menentukan blok kode mana yang akan dieksekusi. Go memiliki 2 macam keyword untuk selesksi kondisi, yaitu if else dan switch.

1. **If Expression**

If adalah salah satu kata kunci yang digunakan dalam percabangan. Percabangan artinya kitabisa mengeksekusi kode program tertentu ketika suatu kondisi terpenuhi. Hampir semua bahasa pemrograman mendukung if expression.

1. **Else if expression**

Terkadang kita butuh membuat beberapa kondisi. Kasus seperti ini dapat menggunakan else if expression. If mendukung short statement sebelum kondisi. Hal ini sangat cocok untuk membuat statement yang sederhana sebelum melakukan pengecekan terhadap kondisi.

**CONTOH SOAL**

1. Buatlah program dalam Bahasa Go untuk menghitung volume kubus berdasarkan panjang sisinya.

**Masukan** terdiri dari suatu bilangan bulat positif yang menyatakan panjang sisi kubus.

**Keluaran** berupa bilangan yang menyatakan volume dari kubus.

**Source Code:**

package main

import "fmt"

func main() {

    var sisi, volume *float64*

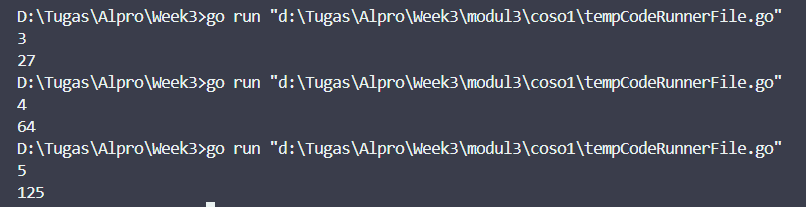
    fmt.Scan(&sisi)

    volume = (sisi \* sisi \* sisi)

    fmt.Print(volume)

}

Output:



1. Buatlah program dalam Bahasa Go yang digunakan untuk menghitung luas segitiga, apabila diketahui panjang alas dan tinggi dari segitiga.

**Masukan** terdiri dari dua bilangan bulat positif yang menyatakan panjang alas dan tinggi dari segitiga.

**Keluaran** berupa bilangan yang menyatakan luas dari segitiga

**Source Code:**

package main

import "fmt"

func main() {

    var alas, tinggi, luas *float64*

    fmt.Print("Masukkan Alas : ")

    fmt.Scan(&alas)

    fmt.Print("Tinggi : ")

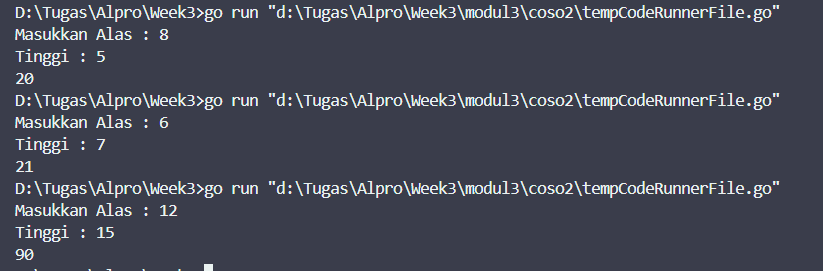
    fmt.Scan(&tinggi)

    luas = (alas \* tinggi / 2)

    fmt.Print(luas)

}

Output:



1. Sebuah program digunakan untuk menghitung konversi mata uang dari IDR ke Dolar US dengan Kurs 15,000 IDR / USD. Masukan terdiri dari bilangan bulat yang menyatakan uang dalam satuan IDR. Keluaran terdiri dari bilangan yang menyatakan uang dalam satuan USD.

**Source Code:**

package main

import "fmt"

func main() {

    var rupiah, dollar *float64*

    fmt.Print("Masukkan nominal rupiah : ")

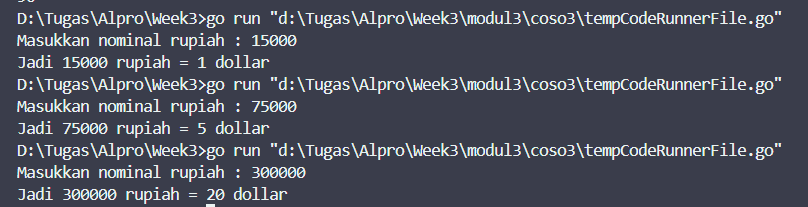
    fmt.Scan(&rupiah)

    dollar = (rupiah / 15000)

    fmt.Print("Jadi ", rupiah, " rupiah = ", dollar, " dollar")

}

Output:



**LATIHAN SOAL**

1. Latihan1

Source Code:

package main

import "fmt"

func main() {

    var x, fx *float64*

    fmt.Print("Masukkan nilai x : ")

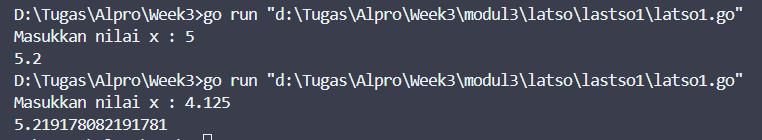
    fmt.Scan(&x)

    fx = (2.0 / (x + 5.0)) + 5

    fmt.Print(fx)

}

Output:



Deskripsi Program:

Program untuk mengetahui nilai dari fungsi f(x) dengan memasukkan nilai bertipe data bilangan rill.

1. Latihan2

Source Code:

package main

import (

    "fmt"

    "math"

)

func main() {

    var (

        r            *int*

        volume, luas *float64*

    )

    fmt.Print("Masukkan jari-jari bola : ")

    fmt.Scanln(&r)

    rFloat := *float64*(r)

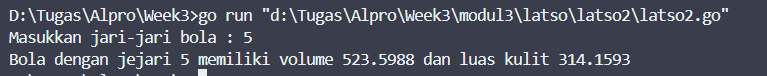
    volume = (4.0 / 3.0) \* math.Pi \* (rFloat \* rFloat \* rFloat)

    luas = 4 \* math.Pi \* (rFloat \* rFloat)

    fmt.Printf("Bola dengan jejari %v memiliki volume %.4f dan luas kulit %.4f", r, volume, luas)

}

Output:



Deskripsi Program:

Program untuk mencari luas kulit dan volume bola dengan memasukkan nilai jari-jari bertipe data int kemudian melakukan casting merubah tipe data jari-jari dari integer menjadi float64. Kemudian melakukan output dengan fmt.Printf(“%.4f”) untuk format output, membulatkan decimal hanya menjadi 4 digit dibelakang koma.

1. Latihan3

Source Code:

package main

import "fmt"

func main() {

    var tahun *int*

    var kabisat *bool*

    fmt.Print("Masukkan tahun : ")

    fmt.Scanln(&tahun)

    if tahun%4 == 0 && (tahun%100 != 0 || tahun%400 == 0) {

        kabisat = true

    } else {

        kabisat = false

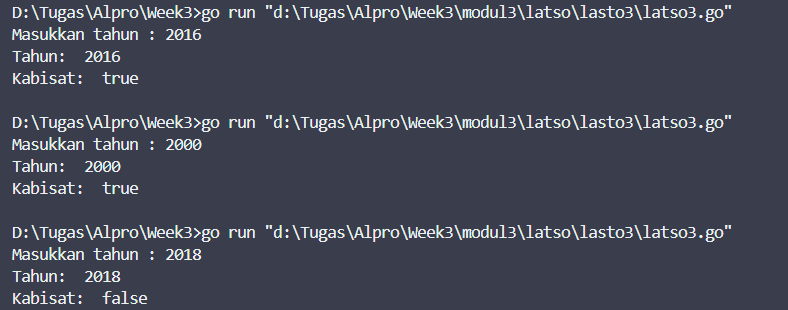
    }

    fmt.Println("Tahun: ", tahun)

    fmt.Println("Kabisat: ", kabisat)

}

Output:



Deskripsi Program:

Menentukan nilai Boolean dari tahun yang diinput. Dengan melakukan pengecekan dengan menggunakan if, else statement. Menggunakan 2 cek, Pengecekan pertama dengan tahun dimodulo dengan 4 kemudian melakukan cek nilai, jika sama dengan 0 maka true DAN lanjut cek tahun dimodulo dengan 100, jika tidak sama dengan 0 maka true atau cek tahun dimodulo dengan 400, jika sama dengan 0 maka true. Jika 2 cek tersebut tidak terpenuhi, maka kabisat bernilai false.

1. Latihan4

Source Code:

package main

import "fmt"

func main() {

    var (

        celcius, reamur, fahrenheit, kelvin *float64*

    )

    fmt.Print("Temperatur Celcius: ")

    fmt.Scan(&celcius)

*// celcius = (fahrenheit - 32) \* (5.0 / 9.0)*

    fahrenheit = (celcius \* (9.0 / 5.0)) + 32

    reamur = celcius \* (4.0 / 5.0)

    kelvin = (fahrenheit + 459.67) \* 5 / 9

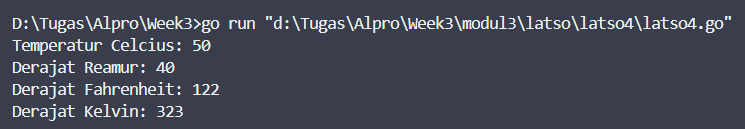
    fmt.Printf("Derajat Reamur: %.0f\n", reamur)

    fmt.Printf("Derajat Fahrenheit: %.0f\n", fahrenheit)

    fmt.Printf("Derajat Kelvin: %.0f\n", kelvin)

}

Output:



Deskripsi Program:

Konversi suhu dari celcius ke reamur, fahrenheit, dan kelvin. Dengan memasukkan nilai bilangan riil.

**DAFTAR PUSTAKA**

Noval Agung Prayogo. *Dasar Pemrograman Golang.* Diakses pada 01 Oktober 2024. <https://dasarpemrogramangolang.novalagung.com>

Annisa Nur Isnaeni. *Golang — Seleksi Kondisi.* Diakses pada 01 Oktober 2024.

https://medium.com/@annisaisna/golang-seleksi-kondisi-f988ead004b4