

**LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 1**

MODUL IV

Modul 4. I/O, Tipe Data dan Variabel (Latihan 2)



Disusun Oleh :

Julian Saputra / 103112400260

IF-12-05

Asisten Praktikum :

Ayu Susilowati

Noviana Rizki Anisa Putri

Dosen Pengampu :

Yudha Islami Sulistya

PROGRAM STUDI S1 TEKNIK INFORMATIKA

FAKULTAS INFORMATIKA

TELKOM UNIVERSITY PURWOKERTO

2024

TUGAS PENDAHULUAN

A. TUGAS PENDAHULUAN (Soal Latihan pada Modul)

Soal Studi Case

1. Buatlah program yang meminta pengguna memasukkan jari-jari sebuah lingkaran, kemudian menghitung dan menampilkan luas serta keliling lingkaran.

Instruksi:

a. Formula untuk luas lingkaran :

$$L = \pi \times r^2$$

b. Formula untuk keliling lingkaran :

$$K = 2 \times \pi \times r$$

c. Cetak hasil luas dan keliling

Sourcecode

```
package main

import (
    "fmt"
    "math"
)

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

    luas := math.Pi * math.Pow(jariJari, 2)
    keliling := 2 * math.Pi * jariJari

    fmt.Printf("Luas lingkaran dengan jari-jari %.2f adalah: %.2f\n", jariJari, luas)
    fmt.Printf("Keliling lingkaran dengan jari-jari %.2f adalah: %.2f\n", jariJari, keliling)
}
```

Screenshoot Output

```
PS C:\Modul 4> go run "c:\Modul 4\TP\luasdankelilinglingkaran.go"
Masukkan jari-jari lingkaran: 20
Luas lingkaran dengan jari-jari 20.00 adalah: 1256.64
Keliling lingkaran dengan jari-jari 20.00 adalah: 125.66
```

Deskripsi Program

Program ini akan meminta user untuk memasukkan jari-jari sebuah lingkaran, kemudian program akan menghitung luas dan keliling lingkaran berdasarkan rumus yang telah ditentukan. Program menggunakan konstanta π (pi) yang tersedia dalam paket *math* untuk akurasi yang lebih baik.

Pertama keliling, luas lingkaran memiliki rumus $L = \pi \times r^2$, dimana L adalah luas dan r adalah jari jari.

Kedua luas, keliling lingkaran memiliki rumus $K = 2 \times \pi \times r$, dimana K adalah keliling lingkaran.

Soal Studi Case

2. Tulis program Go yang menghitung gaji bulanan karyawan berdasarkan jam kerja. Program harus meminta input jumlah jam kerja dalam seminggu dan upah per jam.

Instruksi:

- Jika jam kerja lebih dari 40 jam per minggu, hitung lembur dengan bayaran 1,5 kali lipat untuk jam lembur.
- Total gaji dihitung sebagai:
$$\text{Total gaji} = \text{jam normal} \times \text{upah per jam} + \text{lembur} \times 1.5 \times \text{upah per jam}$$
- Tampilkan total gaji bulanan.

Sourcecode

```
package main

import (
    "fmt"
)

func main() {
    var jamKerjaPerMinggu, upahPerJam float64

    fmt.Print("Masukkan jumlah jam kerja per minggu: ")
    fmt.Scan(&jamKerjaPerMinggu)
    fmt.Print("Masukkan upah per jam: ")
    fmt.Scan(&upahPerJam)

    var jamNormal, jamLembur float64
    if jamKerjaPerMinggu > 40 {
        jamNormal = 40
        jamLembur = jamKerjaPerMinggu - 40
    } else {
        jamNormal = jamKerjaPerMinggu
        jamLembur = 0
    }

    totalGaji := (jamNormal * upahPerJam * 4) + (jamLembur * 1.5 * upahPerJam * 4)

    fmt.Printf("Total gaji bulanan adalah: %.2f\n", totalGaji)
}
```

Screenshoot Output

```
PS C:\Modul 4> go run "c:\Modul 4\TP\gajibulanankaryawan.go"
Masukkan jumlah jam kerja per minggu: 60
Masukkan upah per jam: 2,4
Total gaji bulanan adalah: 560.00
```

Deskripsi Program

Program akan meminta user untuk memasukkan jumlah jam kerja dalam seminggu dan upah per jam. Lalu, program akan menghitung gaji bulanan karyawan dengan mempertimbangkan lembur jika jam kerja melebihi 40 jam dalam seminggu.

Rumus yang digunakan :

1. Jam normal

- a) Jika jam kerja per minggu ≤ 40 , semua jam dihitung sebagai jam normal.
- b) Jika jam kerja per minggu > 40 , maka 40 jam pertama dihitung sebagai jam normal dan sisanya dihitung sebagai jam lembur.

2. Total gaji diitung :

- a) Total gaji: $(\text{jam normal} \times \text{upah per jam} \times 4) + (\text{jam lembur} \times 1.5 \times \text{upah per jam} \times 4)$

Penjelasan :

Dimana Gaji dibayar setiap bulan, sehingga jam normal dan lembur dikalikan dengan 4 (jumlah minggu dalam sebulan).