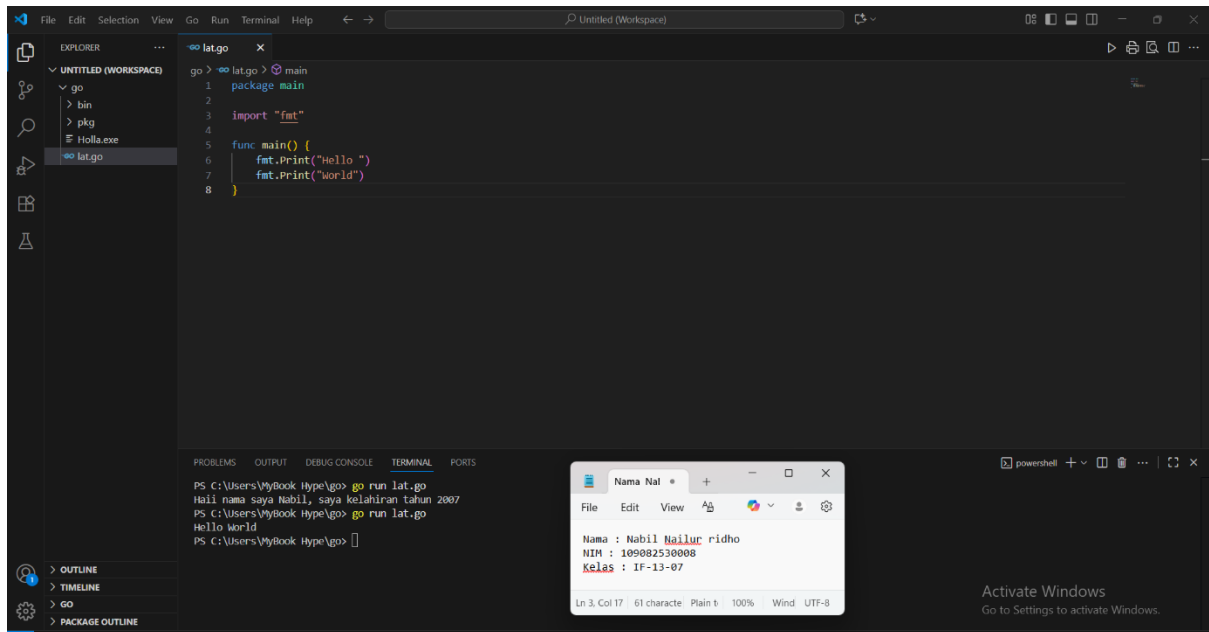


# TUGAS PENDAHULUAN

## INPUT/OUTPUT TIPE DATA DAN VARIABEL

### 1. Jelaskan perbedaan `fmt.Print()`, `fmt.Println()`, dan `fmt.Printf()` di Go!

**`fmt.Print()`** Untuk mencetak argumen secara mentah tanpa spasi otomatis atau baris baru di akhir. Cocok untuk output kontinu tanpa modifikasi.



The screenshot shows the VS Code editor with a Go file named `lat.go`. The code is as follows:

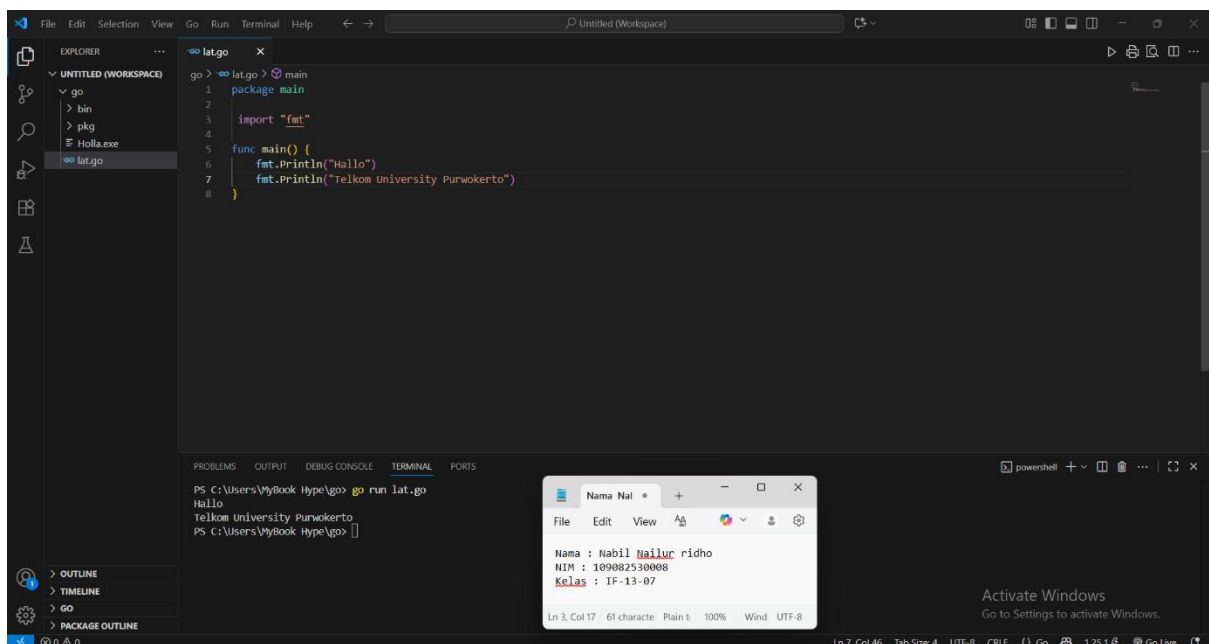
```
1 package main
2
3 import "fmt"
4
5 func main() {
6     fmt.Print("Hello ")
7     fmt.Print("world")
8 }
```

The terminal output shows the result of running the program:

```
PS C:\Users\MyBook Hype\go> go run lat.go
Hail nama saya Nabil, saya kelahiran tahun 2007
PS C:\Users\MyBook Hype\go> go run lat.go
Hello World
PS C:\Users\MyBook Hype\go> []
```

A Notepad window titled "Nama Nal" displays the output of the program, showing the concatenated string "Hail nama saya Nabil, saya kelahiran tahun 2007" and "Hello World".

**`fmt.Println()`** yaitu mirip mirip dengan `fmt.Print`, tapi bedanya yaitu jika `fmt.Println()` itu nanti di outputnya akan menambahkan spasi antar kalimat dan baris baru, agar output yang rapi dan mudah dibaca.



The screenshot shows the VS Code editor with a Go file named `lat.go`. The code is as follows:

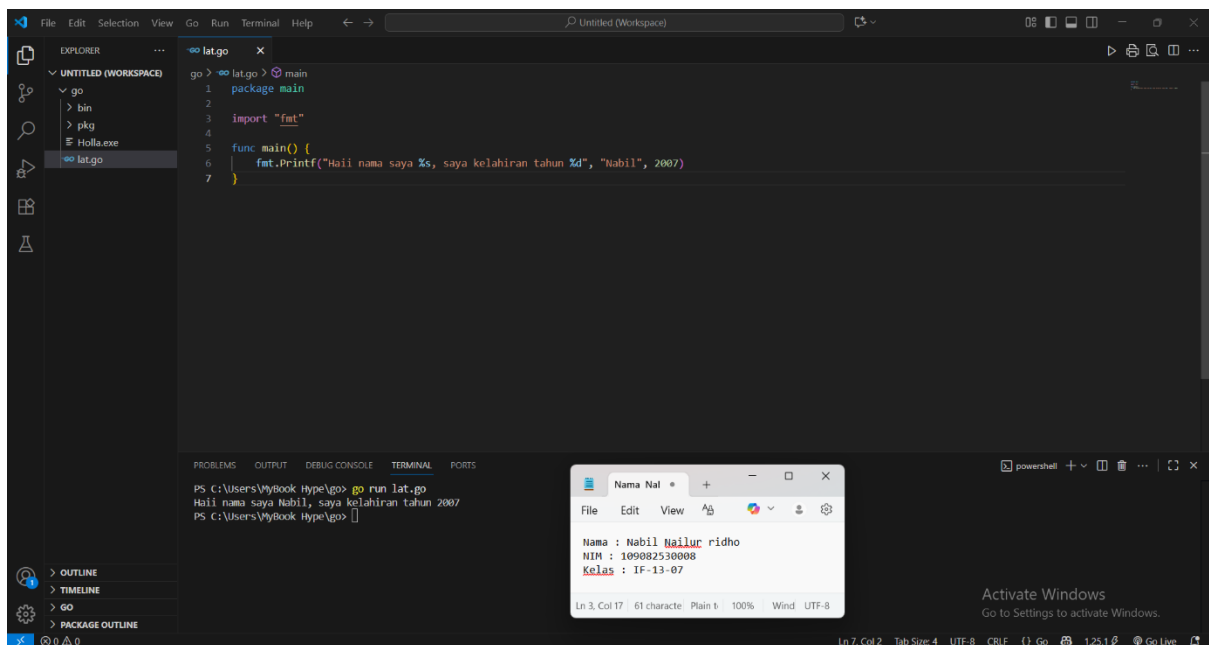
```
1 package main
2
3 import "fmt"
4
5 func main() {
6     fmt.Println("Hailo")
7     fmt.Println("telkom University Purwokerto")
8 }
```

The terminal output shows the result of running the program:

```
PS C:\Users\MyBook Hype\go> go run lat.go
Hailo
telkom University Purwokerto
PS C:\Users\MyBook Hype\go> []
```

A Notepad window titled "Nama Nal" displays the output of the program, showing the concatenated string "Hailo" and "telkom University Purwokerto" on separate lines.

**fmt.Printf()** itu digunakan untuk mencetak data dengan format tertentu menggunakan placeholder, misalnya %d untuk mengambil bilangan bulat, sedangkan %s untuk string, dan %f untuk bilangan desimal. Fungsi ini tidak menambahkan baris baru secara otomatis jadi biasanya diberi dengan tanda \n jika ingin menambah baris baru.



The screenshot shows the Visual Studio Code editor with a Go file named `lat.go`. The code defines a `main` package and a `main` function. Inside the function, `fmt.Printf` is used to print a formatted string. The terminal output shows the result of running the program.

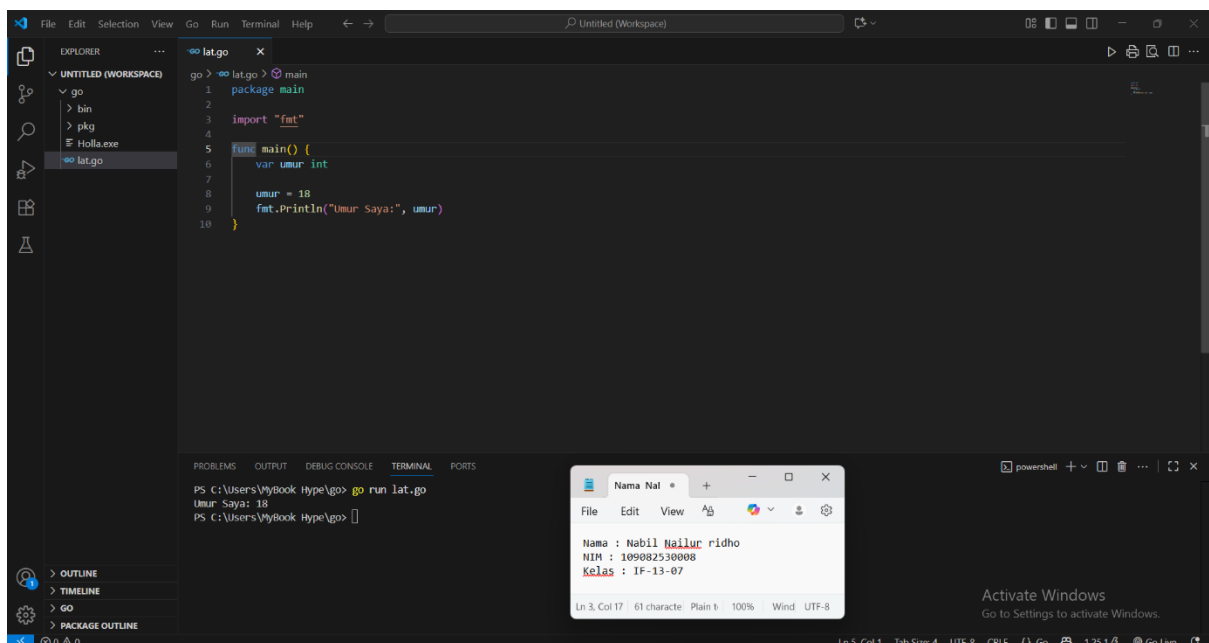
```
go > go lat.go > main
1 package main
2
3 import "fmt"
4
5 func main() {
6     fmt.Printf("Hai nama saya %s, saya kelahiran tahun %d", "Nabil", 2007)
7 }
```

```
PS C:\Users\Mybook Hype\go> go run lat.go
Hai nama saya Nabil, saya kelahiran tahun 2007
PS C:\Users\Mybook Hype\go>
```

## 2. Jelaskan Penggunaan Tipe Data int, float64, bool, dan string di Go!

### Tipe Data int:

Tipe data int ini untuk menyimpan bilangan bulat (tanpa desimal), bisa positif, negatif, atau nol. Lalu ada juga varian int seperti `int32`, `int64` buat ukuran tetap, dan `uint`.



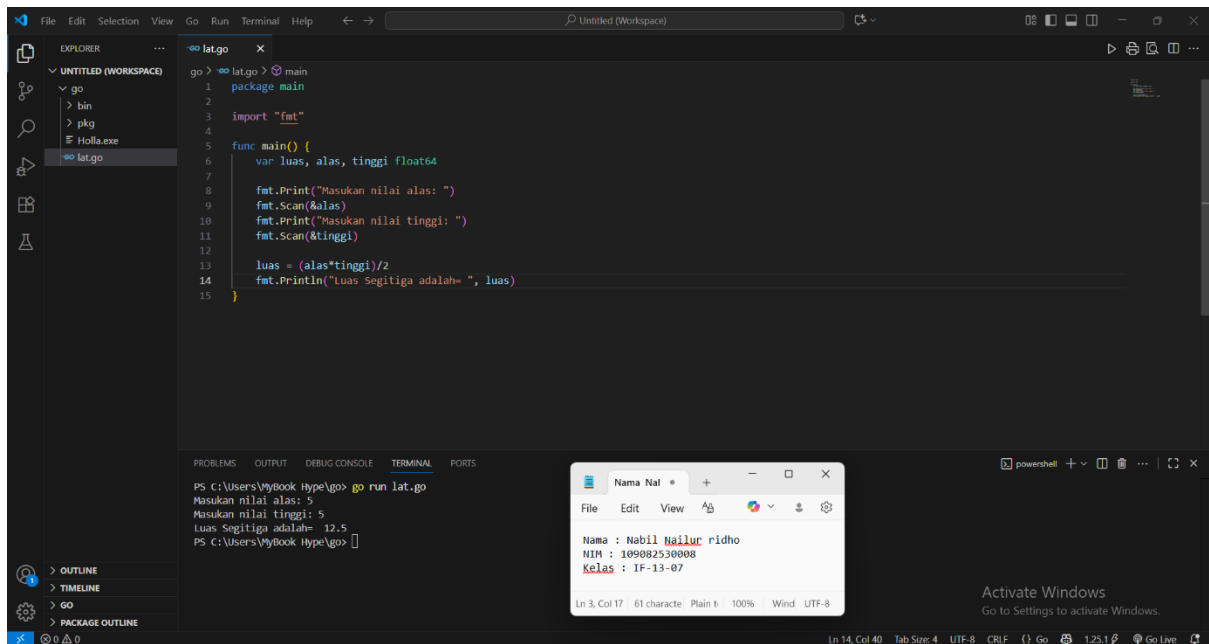
The screenshot shows the Visual Studio Code editor with a Go file named `lat.go`. The code defines a `main` package and a `main` function. Inside the function, a variable `umur` of type `int` is declared and assigned the value 18. Then, `fmt.Println` is used to print the value of `umur`. The terminal output shows the result of running the program.

```
go > go lat.go > main
1 package main
2
3 import "fmt"
4
5 func main() {
6     var umur int
7
8     umur = 18
9     fmt.Println("Umur Saya:", umur)
10 }
```

```
PS C:\Users\Mybook Hype\go> go run lat.go
Umur Saya: 18
PS C:\Users\Mybook Hype\go>
```

## Tipe Data float64:

Tipe data buat nyimpen bilangan desimal (bilangan real) float64 adalah yang paling umum di gunakan pada program Go, ada juga float32 buat ukuran lebih kecil.

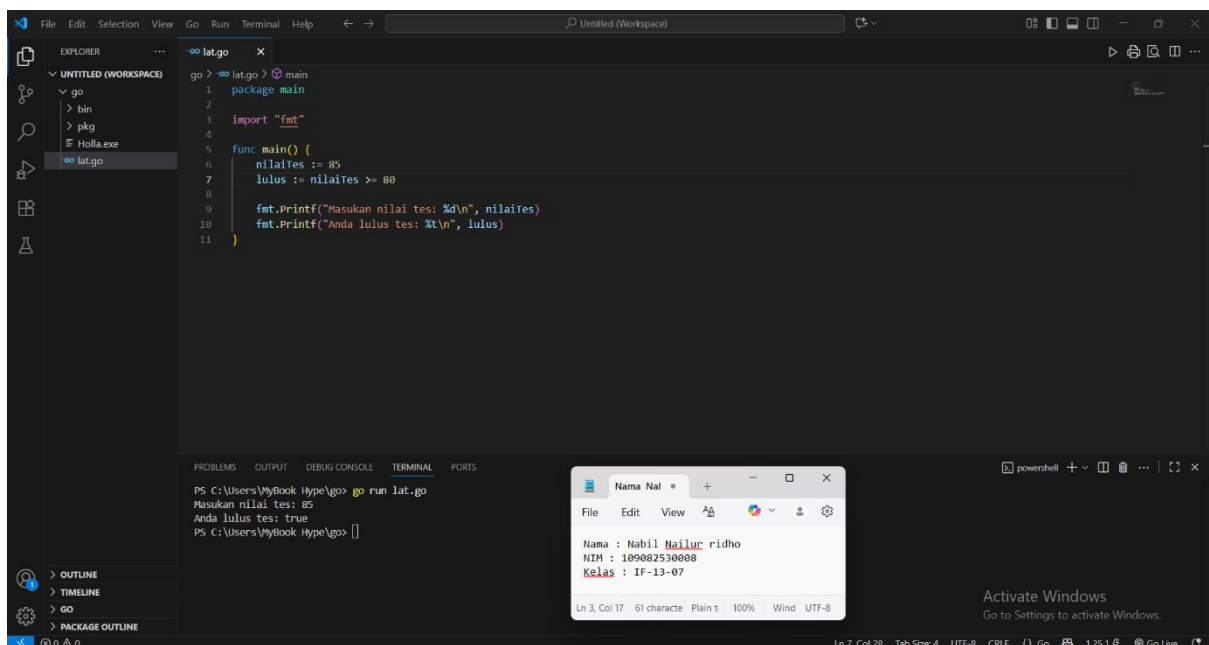


```
go > latgo > main
1 package main
2
3 import "fmt"
4
5 func main() {
6     var luas, alas, tinggi float64
7
8     fmt.Print("Masukan nilai alas: ")
9     fmt.Scan(&alas)
10    fmt.Print("Masukan nilai tinggi: ")
11    fmt.Scan(&tinggi)
12
13    luas = (alas*tinggi)/2
14    fmt.Println("Luas Segitiga adalah= ", luas)
15 }
```

```
PS C:\Users\MyBook Hype\go> go run lat.go
Masukan nilai alas: 5
Masukan nilai tinggi: 5
Luas Segitiga adalah= 12.5
PS C:\Users\MyBook Hype\go> []
```

## Tipe Data bool:

Tipe data yang cuma bisa nyimpen dua nilai yaitu apakah nilai dari ini true (benar) atau false (salah), dan nggak ada nilai lain seperti null.

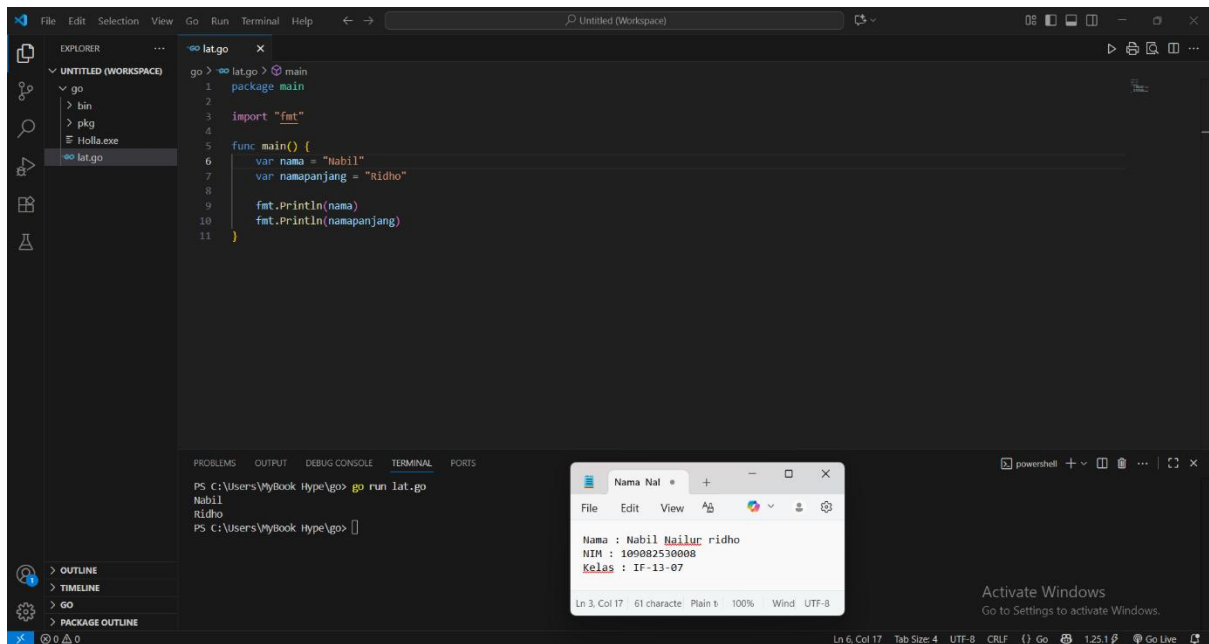


```
go > latgo > main
1 package main
2
3 import "fmt"
4
5 func main() {
6     nilaites := 85
7     lulus := nilaites >= 80
8
9     fmt.Printf("Masukan nilai tes: %d\n", nilaites)
10    fmt.Printf("Anda lulus tes: %t\n", lulus)
11 }
```

```
PS C:\Users\MyBook Hype\go> go run lat.go
Masukan nilai tes: 85
Anda lulus tes: true
PS C:\Users\MyBook Hype\go> []
```

## Tipe Data string:

Tipe data string ini buat menyimpan nama, teks atau urutan karakter, kalimat pada output pemrograman bahasa Golang.



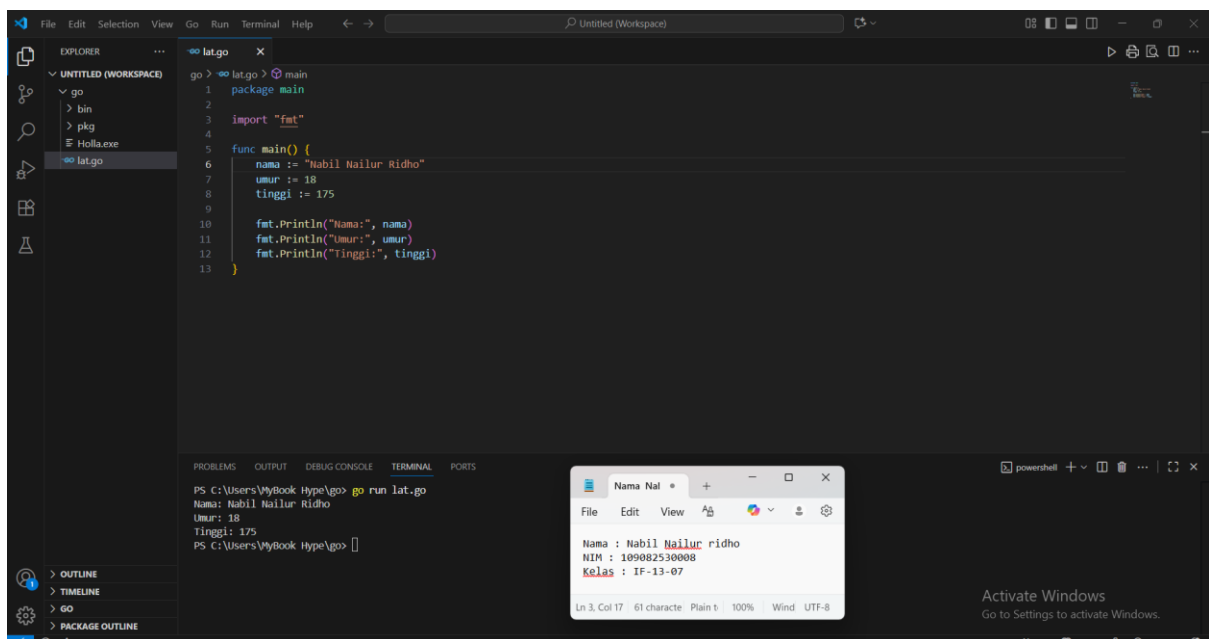
The screenshot shows the Visual Studio Code editor with a Go file named `lat.go`. The code defines a `main` package and a `main` function. Inside the function, two variables are declared: `nama` with the value "Nabil" and `namapanjang` with the value "ridho". The function then prints both variables using `fmt.Println`. The terminal output shows the command `go run lat.go` being executed, resulting in the output: `Nabil` and `ridho`. A small window titled "Nama Nal" is also visible, displaying the same output.

```
go > go run lat.go
1 package main
2
3 import "fmt"
4
5 func main() {
6     var nama = "Nabil"
7     var namapanjang = "ridho"
8
9     fmt.Println(nama)
10    fmt.Println(namapanjang)
11 }
```

```
PS C:\Users\MyBook Hype\go> go run lat.go
Nabil
ridho
PS C:\Users\MyBook Hype\go>
```

### 3. Bagaimana cara mendlarasikan variable dengan kata kunci var dan dengan acara singkat? Berikah contoh! Hint “:=”

Cara deklarasi variabel pakai dengan memakai `:=`, harus sebut tipe seperti `var umur` atau contoh: `var umur = 18`, yang mana outputnya menghasilkan "18" itu sendiri. Contoh lainnua adalah jika kita ingin mengeluarkan atau outputnya nama kita yaitu contoh: `nama = nabil`, yang mana output nya udah pasti “nabil” tidak akan berubah menjadi kata kata lain.



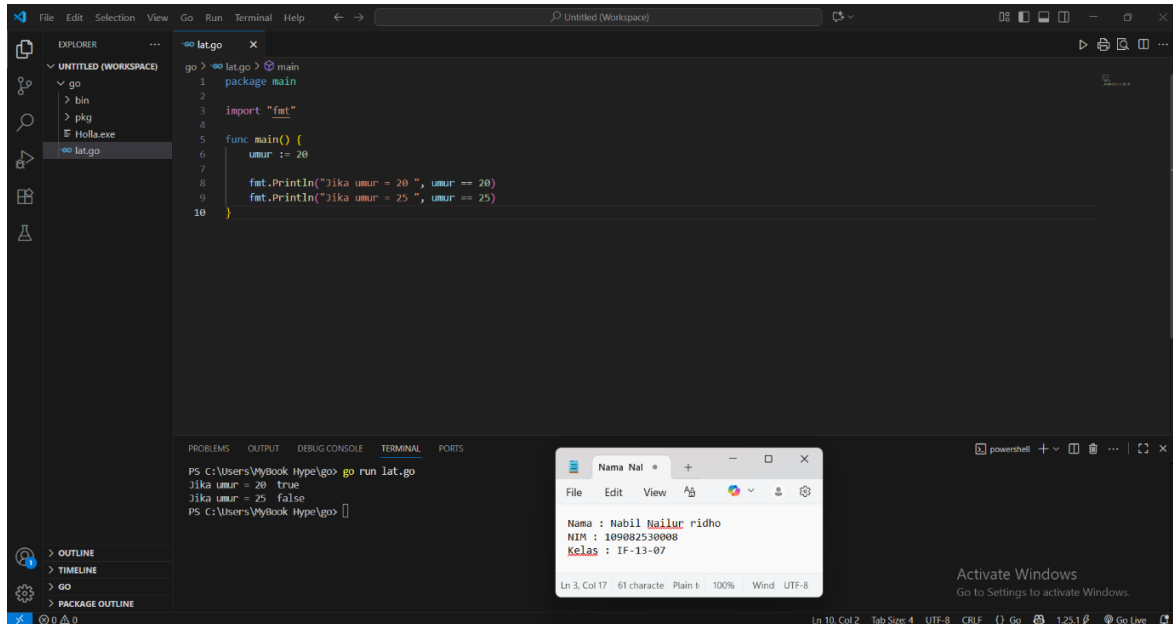
The screenshot shows the Visual Studio Code editor with a Go file named `lat.go`. The code defines a `main` package and a `main` function. Inside the function, three variables are declared using the `:=` operator: `nama` with the value "Nabil Nailur Ridho", `umur` with the value 18, and `tinggi` with the value 175. The function then prints each variable using `fmt.Println`. The terminal output shows the command `go run lat.go` being executed, resulting in the output: `Nama: Nabil Nailur Ridho`, `Umur: 18`, and `Tinggi: 175`. A small window titled "Nama Nal" is also visible, displaying the same output.

```
go > go run lat.go
1 package main
2
3 import "fmt"
4
5 func main() {
6     nama := "Nabil Nailur Ridho"
7     umur := 18
8     tinggi := 175
9
10    fmt.Println("Nama:", nama)
11    fmt.Println("Umur:", umur)
12    fmt.Println("Tinggi:", tinggi)
13 }
```

```
PS C:\Users\MyBook Hype\go> go run lat.go
Nama: Nabil Nailur Ridho
Umur: 18
Tinggi: 175
PS C:\Users\MyBook Hype\go>
```

#### 4. Apa perbedaan antara operator == dan = dalam bahasa Go?

Operator “=” adalah untuk assignment, atau mengisi atau mengubah nilai dalam sebuah variabel, Sementara “==” adalah operator perbandingan untuk mengecek kesamaan, dan juga bisa untuk menentukan (true/false). Jadi, operator “=” untuk mengisi nilai suatu variabel, dan operator “==” untuk membandingkan suatu variabel.



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

##### Source Code

```
package main

import "fmt"

func main() {

    var nama string

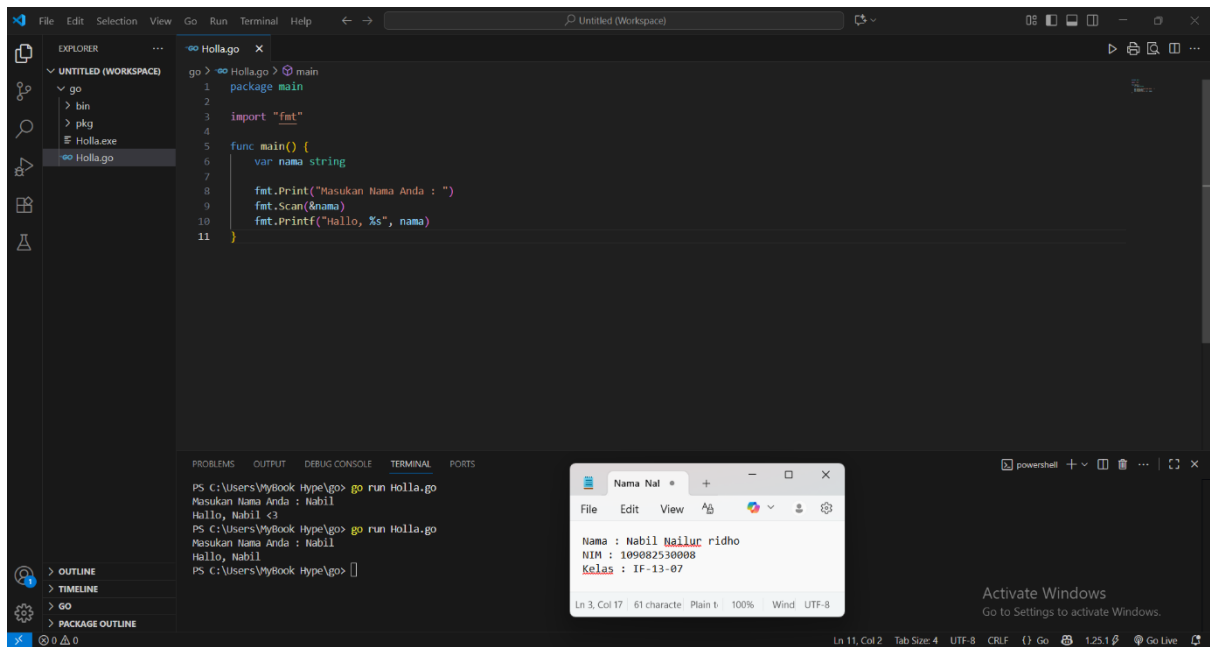
    fmt.Print("Masukan Nama Anda : ")

    fmt.Scan(&nama)

    fmt.Printf("Hallo, %s", nama)
```

```
}
```

## Screenshoot program



## Deskripsi Program:

Program Go ini untuk menampilkan input nama kita sendiri menggunakan `fmt.Print`, lalu menampilkan output nya di terminal yang menghasilkan nama yang di isi sama kita sendiri dengan mennggunakan `fmt.Scan` dan juga `fmt.Printf`.