

Ridho Fariha Lubis
IT Perbankan Backend
No.Reg : 0568281020-7

Tugas Pertemuan 8 : Pengenalan Golang

Golang Script :

```
package main

import (
    "encoding/json"
    "io/ioutil"
    "log"
    "mux"
    "net/http"
)

//Inputan data
type Inputan struct {
    Panjang int `json:"panjang"`
    Lebar    int `json:"lebar"`
    Tinggi   int `json:"tinggi"`
}

//Hasil Hitungan
type Hasil struct {
    JenisBangun string `json:"bangun"`
    Volume       int    `json:"volume"`
}

func main() {
    router := mux.NewRouter()
    router.HandleFunc("/api/hitung-volume", Volume)
    log.Fatal(http.ListenAndServe(":8020", router))
}

//Volume Search
func Volume(w http.ResponseWriter, r *http.Request) {

    var hasilHitung Hasil
    var input Inputan

    body, err := ioutil.ReadAll(r.Body) //mengambil data dari inputan
    form body
    json.Unmarshal(body, &input)        //me-replace data json inputan
    dengan isi yang ada di dalam body

    hasilHitung = Hasil{
        JenisBangun: input.FindBangun(), //cari jenis bangunan
        Volume:       input.RumusVolume(), //cari volume
    }

    result, err := json.Marshal(hasilHitung) //print hasilperhitungan
    dalam bentuk json

    if err == nil {
        w.Header().Set("Content-Type", "application/json")
        w.Write(result)
    }
}

//RumusVolume Bangunan
func (i *Inputan) RumusVolume() int {
    return i.Panjang * i.Lebar * i.Tinggi
}

//FindBangun type
func (i *Inputan) FindBangun() string {
```

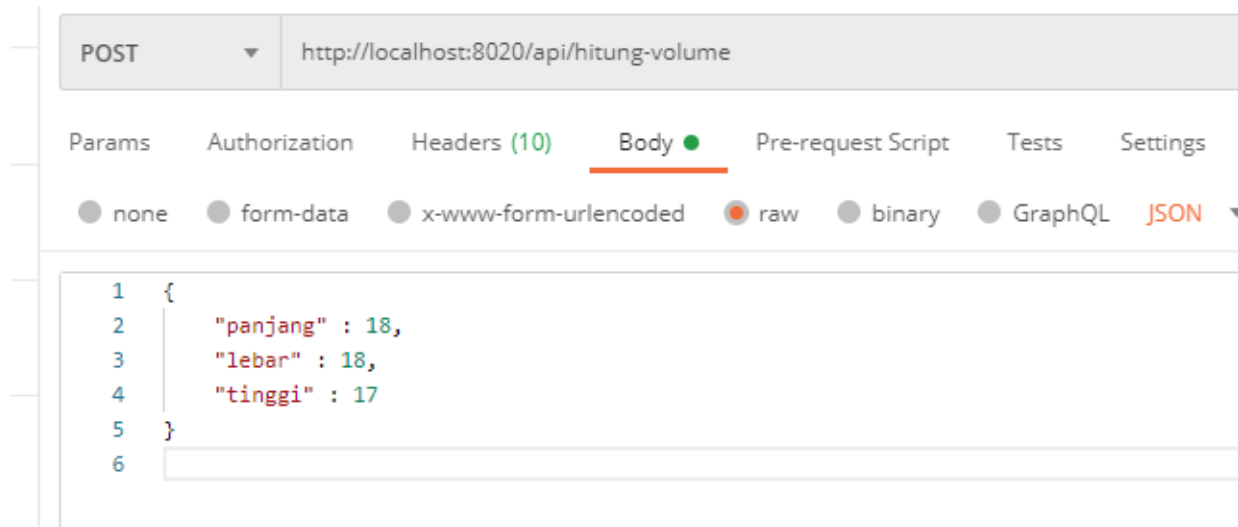
```
        if i.Panjang == i.Lebar && i.Panjang == i.Tinggi && i.Lebar ==  
i.Tinggi {  
            return "Kubus"  
        }  
        return "Balok"  
    }  
}
```

Untuk mendapatkan hasil balok atau kubus, saya tentukan berdasrkan inputannya, jika nilai panjang,lebar, dan tinggi nya sama, maka dipastikan itu adalah bangun kubus, sedangkan jika ada salah satu nilainya berbeda, maka dipastikan itu adalah balok

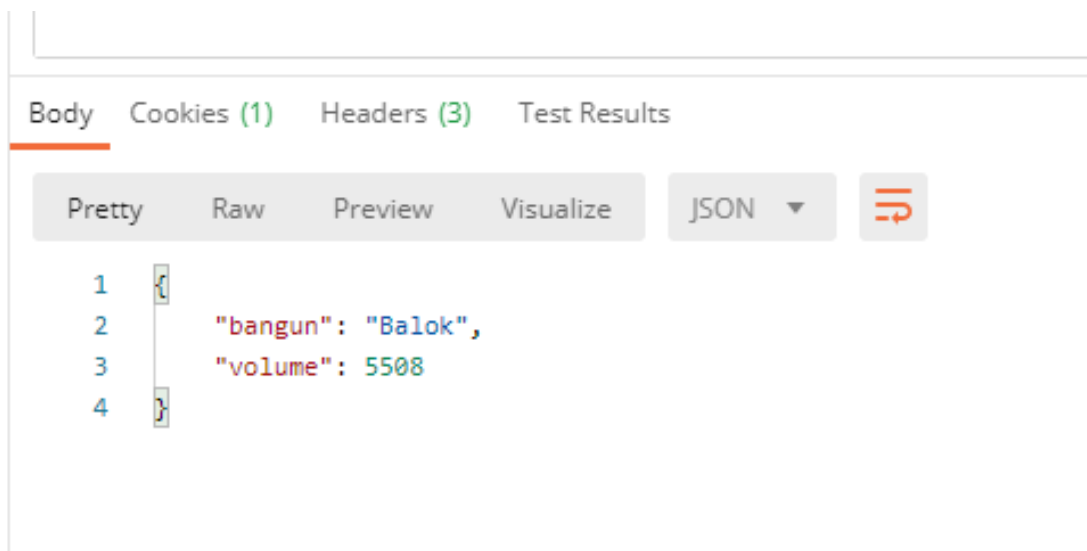
RESULT :

Balok :

Input :



Hasil :



Kubus :

Input :

Untitled Request

POST

http://localhost:8020/api/hitung-volume

Params

Authorization

Headers (10)

Body

Pre-request Script

Tests

Settings

none

form-data

x-www-form-urlencoded

raw

binary

GraphQL

JSON

1

{

2

"panjang" : 20,

3

"lebar" : 20,

4

"tinggi" : 20

5

}

6

Hasil :

Body

Cookies (1)

Headers (3)

Test Results

Pretty

Raw

Preview

Visualize

JSON

1

{

2

"bangun": "Kubus",

3

"volume": 8000

4

}