

# GOLANG FOR INTERMEDIATE

## ACTIVITY 1

### (HTTP Request Pada Golang)

- Silahkan download ZIP Project yang tersedia
- Ekstrak ZIP Project tersebut kemudian rename menjadi **Nama\_NPM\_Pert4** (nama folder tidak boleh ada spasi). Contoh : **ilhamhmmd\_13116429\_Pert4**
- Kemudian pindahkan folder project ke dalam **GOPATH** yaitu **c://golang/src/**
- Buka project menggunakan Code Editor yang tersedia, disarankan menggunakan VS Code.

Lalu pastikan seluruh **import model** pada file go yang berada di dalam **folder test** sesuai dengan nama folder projek.

```
import (  
    "nama_npm_pert4/model"  
    "testing"  
)
```

- Didalam folder **Nama\_NPM\_Pert4** buatlah 1 buah folder yaitu **handler**
- Pada **folder handler** silahkan buat file go berikut ini :

**handler.go**

**(POIN 20)**

```
package handler  
  
import (  
    "nama_npm_pert4/model" //sesuaikan dengan nama folder (case sensitive)  
    "database/sql"  
    "fmt"  
    "net/http"  
    "strings"  
)
```

```

var username, password, host, namaDB, defaultDB string
var db *sql.DB
var err error

func init() {
    username = "CPC[noPC]" //Misal : CPC21
    password = "lepkom@123"
    host = "dbms.lepkom.f4.com"
    namaDB = "db_npm" //Misal : db_13116429
    defaultDB = "mysql"
}

func API(w http.ResponseWriter, r *http.Request) {
    db, err = model.Connect(username, password, host, namaDB)
    if err != nil {
        return
    }
    defer db.Close()

    w.Header().Set("Content-Type", "text-html; charset=utf-8; application/json")
    dataURL := strings.Split(fmt.Sprintf("%s", r.URL.Path), "/")
    switch dataURL[2] {
    case "mahasiswa":
        switch r.Method {
        case "GET":
            HandlerMahasiswaGet(w, r)
        case "POST":
            HandlerMahasiswaPost(w, r)
        case "PUT":
            HandlerMahasiswaPut(w, r)
        case "DELETE":
            HandlerMahasiswaDelete(w, r)
        default:

```

```

        w.Write([]byte("method tidak ditemukan"))
    }
    default:
        w.Write([]byte("request tidak ditemukan"))
    }
}

```

## handlerMahasiswa.go (POIN 20)

```

package handler

import (
    "nama_npm_pert4/model" //sesuaikan dengan nama folder (case sensitive)
    "encoding/json"
    "io/ioutil"
    "net/http"
)

func HandlerMahasiswaGet(w http.ResponseWriter, r *http.Request) {

    var data interface{}
    var err error

    npm := r.URL.Query()["npm"]
    if len(npm) != 0 {
        data, err = model.GetMahasiswa(db, npm[0])
    } else {
        data, err = model.GetAllMahasiswa(db)
    }

    if err != nil {
        http.Error(w, err.Error(), http.StatusInternalServerError)
    }
}

```

```

    jsonData, _ := json.Marshal(data)
    w.Write(jsonData)
}

func HandlerMahasiswaPost(w http.ResponseWriter, r *http.Request) {
    defer r.Body.Close()
    body, err := ioutil.ReadAll(r.Body)
    if err != nil {
        http.Error(w, err.Error(), http.StatusInternalServerError)
        return
    }
    var data model.Mahasiswa
    if err = json.Unmarshal(body, &data); err != nil {
        http.Error(w, err.Error(), http.StatusInternalServerError)
        return
    }
    if err = data.Insert(db); err != nil {
        http.Error(w, err.Error(), http.StatusInternalServerError)
        return
    }
    jsonData, err := json.Marshal(data)
    if err != nil {
        http.Error(w, err.Error(), http.StatusInternalServerError)
    }
    w.Write(jsonData)
}

func HandlerMahasiswaDelete(w http.ResponseWriter, r *http.Request) {

    npm := r.URL.Query()["npm"]

    if len(npm) != 0 {
        data := model.Mahasiswa{NPM: npm[0]}
    }
}

```

```

        if err := data.Delete(db); err != nil {
            http.Error(w, "ID tidak ditemukan", http.StatusBadRequest)
            return
        }
        w.Write([]byte("Data telah dihapus"))
    } else {
        http.Error(w, "ID tidak ditemukan", http.StatusInternalServerError)
    }
}

func HandlerMahasiswaPut(w http.ResponseWriter, r *http.Request) {

    npm := r.URL.Query()["npm"]

    if len(npm) == 0 {
        http.Error(w, "ID tidak ditemukan", http.StatusBadRequest)
        return
    }

    defer r.Body.Close()
    body, err := ioutil.ReadAll(r.Body)
    if err != nil {
        http.Error(w, err.Error(), http.StatusInternalServerError)
        return
    }

    jsonmap := make(map[string]interface{})
    err = json.Unmarshal(body, &jsonmap)
    if err != nil {
        http.Error(w, err.Error(), http.StatusInternalServerError)
        return
    }
}

```

```

data := model.Mahasiswa{NPM: npm[0]}

err = data.Update(db, jsonmap)
if err != nil {
    http.Error(w, err.Error(), http.StatusInternalServerError)
    return
}

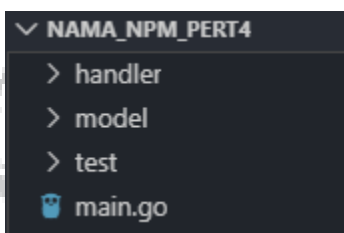
result, err := model.GetMahasiswa(db, npm[0])

if err != nil {
    http.Error(w, err.Error(), http.StatusInternalServerError)
    return
}

jsonData, err := json.Marshal(result)
if err != nil {
    http.Error(w, err.Error(), http.StatusInternalServerError)
    return
}
w.Write(jsonData)
}

```

- Kemudian buatlah file bernama main.go **diluar semua folder!**



## Main.go (POIN 10)

```
package main

import (
    "nama_npm_pert4/handler" //sesuaikan dengan nama folder (case sensitive)
    "log"
    "net/http"
)

func main() {
    http.HandleFunc("/api/", handler.API)

    //Ganti 2 digit akhir port dengan 2 digit akhir NPM anda
    log.Println("localhost : 8029")
    http.ListenAndServe(":8029", nil)
}
```

- Setelah itu silahkan jalankan Unit Testing **database\_test.go** terlebih dahulu

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS C:\laragon\www\go\src\nama_npm_pert3> cd test
PS C:\laragon\www\go\src\nama_npm_pert3\test> go test -run TestDatabase
PASS
ok      nama_npm_pert3/test      0.950s
PS C:\laragon\www\go\src\nama_npm_pert3\test> █
```

Kemudian cek pada Mysql atau PHPMYADMIN bahwa database telah berhasil dibuat oleh Unit Testing secara otomatis

Note : Apabila muncul pesan *database exist*, abaikan dan lanjutkan ke point testing berikutnya.

- Kemudian jalankan Unit Testing **mahasiswa\_test.go**

```
PS C:\laragon\www\go\src\nama_npm_pert3\test> go test -run TestMahasiswa
&{12345678 Abdi Teh Ayeuna 3KA20}
&{19283746 Doremi Budi 4KA20}
&{44444444 DoBud 4KA21}
PASS
ok      nama_npm_pert3/test      2.088s
```

Kemudian cek pada Mysql atau PHPMYADMIN bahwa table **mahasiswa** telah berhasil dibuat oleh Unit Testing beserta record-nya secara otomatis

- Silahkan jalankan program berikut dengan perintah berikut :

```
PS C:\laragon\www\go\src\nama_npm_pert4> go run main.go
```

Dan akses <http://127.0.0.1:8029/api/mahasiswa> (sesuaikan portnya)

- Selanjutnya silahkan lanjutkan dengan membuat file handler lainnya

handler :

- **handlerMatkul.go** (POIN 25)
- **handlerNilai.go** (POIN 25)

Penjelasan penuh dan kode lainnya pada Activity ini terdapat di video

**Selamat Mengerjakan Calon Orang-Orang Sukses,  
Jangan Lupa Kurung Kurawalnya Harus Pas Hiya Hiya ☺**