

<p style="text-align: center;"><b>Tugas 5</b>  <b>Pemrograman Jaringan (CSH4V3)</b>          -----  <b>Semester Ganjil 2019 - 2020</b>  <b>Dosen: Aulia Arif Wardana, S.Kom., M.T. (UIW)</b></p>		
<p style="text-align: center;"><i>Berdo'alah sebelum mengerjakan. Dilarang berbuat curang.          Tugas ini untuk mengukur kemampuan anda, jadi kerjakan dengan sepenuh hati.          Selamat belajar, semoga sukses !</i></p>		
<p><b>Nama Mahasiswa:</b></p> <p style="text-align: center;"><b>Mia Amelia</b></p>	<p><b>NIM:</b></p> <p style="text-align: center;"><b>1301164310</b></p>	<p><b>Nilai:</b></p> <p style="text-align: center;">.....</p>
<p><b>Nama Mahasiswa:</b></p> <p style="text-align: center;"><b>Muhammad Baihaqy</b></p>	<p><b>NIM:</b></p> <p style="text-align: center;"><b>1301164557</b></p>	<p><b>Nilai:</b></p> <p style="text-align: center;">.....</p>
<p><b>Nama Mahasiswa:</b></p> <p style="text-align: center;"><b>M Thariq Sunu R</b></p>	<p><b>NIM:</b></p> <p style="text-align: center;"><b>1301150775</b></p>	<p><b>Nilai:</b></p> <p style="text-align: center;">.....</p>

<p><b>Siapkan tools berikut sebelum mengerjakan:</b></p>
<ol style="list-style-type: none"> <li>1. Go Programming Language (<a href="https://golang.org/dl/">https://golang.org/dl/</a>).</li> <li>2. Visual Studio Code (<a href="https://code.visualstudio.com/">https://code.visualstudio.com/</a>) atau LiteIDE (<a href="https://github.com/visualfc/liteide">https://github.com/visualfc/liteide</a>).</li> <li>3. Harus menggunakan linux dengan distro fedora (<a href="https://getfedora.org/id/workstation/">https://getfedora.org/id/workstation/</a>).</li> <li>4. Buatlah git repository pada <a href="https://github.com/">https://github.com/</a> kemudian push semua kode dan hasil laporan anda ke dalam repository github yang sudah anda buat.</li> <li>5. Lakukan instalasi flatbuffer (<a href="https://google.github.io/flatbuffers/">https://google.github.io/flatbuffers/</a>) untuk mengerjakan salah satu tugas pada modul ini.</li> <li>6. Kumpulkan link repository github tersebut sebagai tanda bahwa anda mengerjakan tugas modul ini.</li> <li>7. Link repository harus berbeda untuk setiap tugasnya. Buatlah markdown yang rapi di setiap repository tugas yang anda kumpulkan.</li> <li>8. Printscreen program harus dari desktop kelompok anda sendiri, dan harus dari linux yang sudah diinstall. Jika tidak, maka harus mengulang pengerjaan tugasnya.</li> <li>9. Jangan lupa untuk menuliskan NAMA dan NIM pada laporan.</li> <li>10. Laporan berbentuk PDF dan dikumpulkan pada link repository github beserta kodenya.</li> <li>11. Walaupun tugas berkelompok tapi pengumpulan link github harus individu, jika tidak mengumpulkan maka dianggap tidak mengerjakan.</li> </ol>



Nama:	NIM:	Nilai:
-------	------	--------

### Soal No 1 (JSON Marshal)

```
package main

import (
    "encoding/json"
    "fmt"
)

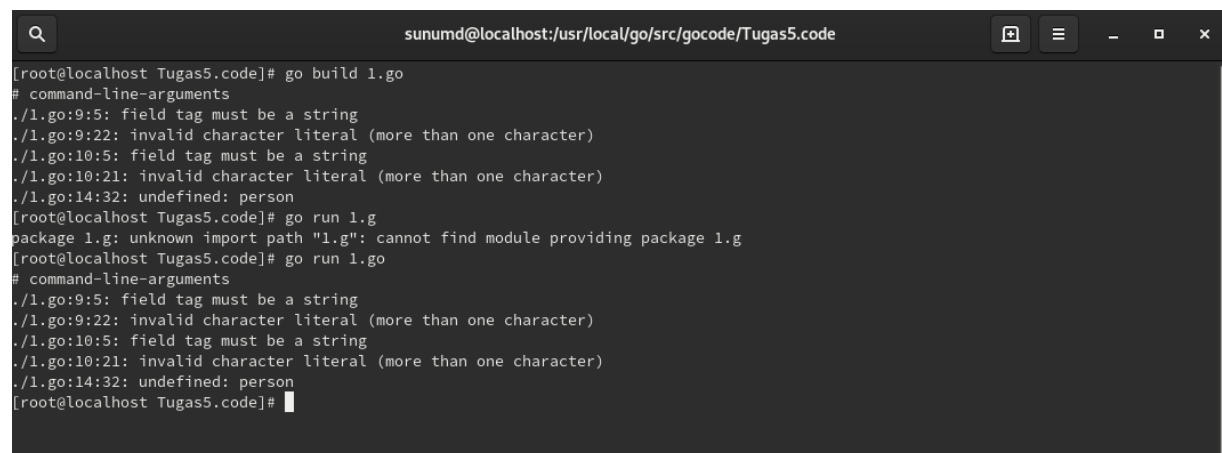
type Person struct {
    FirstName string `json:"firstName"`
    LastName  string `json:"lastName"`
}

func main() {
    bytes, err := json.Marshal(Person{
        FirstName: "John",
        LastName:  "Dow",
    })
    if err != nil {
        panic(err)
    }

    fmt.Println(string(bytes))
}
```

Jalankan program diatas, apakah outputnya (berikan printscreen) dan jelaskan cara kerjanya!

Jawaban:



```
sunumd@localhost:/usr/local/go/src/gocode/Tugas5.code
[root@localhost Tugas5.code]# go build 1.go
# command-line-arguments
./1.go:9:5: field tag must be a string
./1.go:9:22: invalid character literal (more than one character)
./1.go:10:5: field tag must be a string
./1.go:10:21: invalid character literal (more than one character)
./1.go:14:32: undefined: person
[root@localhost Tugas5.code]# go run 1.g
package 1.g: unknown import path "1.g": cannot find module providing package 1.g
[root@localhost Tugas5.code]# go run 1.go
# command-line-arguments
./1.go:9:5: field tag must be a string
./1.go:9:22: invalid character literal (more than one character)
./1.go:10:5: field tag must be a string
./1.go:10:21: invalid character literal (more than one character)
./1.go:14:32: undefined: person
[root@localhost Tugas5.code]#
```

terdapat error strng char

Nama:	NIM:	Nilai:
-------	------	--------

--

## Soal No 2 (JSON Unmarshal)

```
package main

import (
    "encoding/json"
    "fmt"
)

type Person struct {
    FirstName string `json:"firstName"`
    LastName  string `json:"lastName"`
}

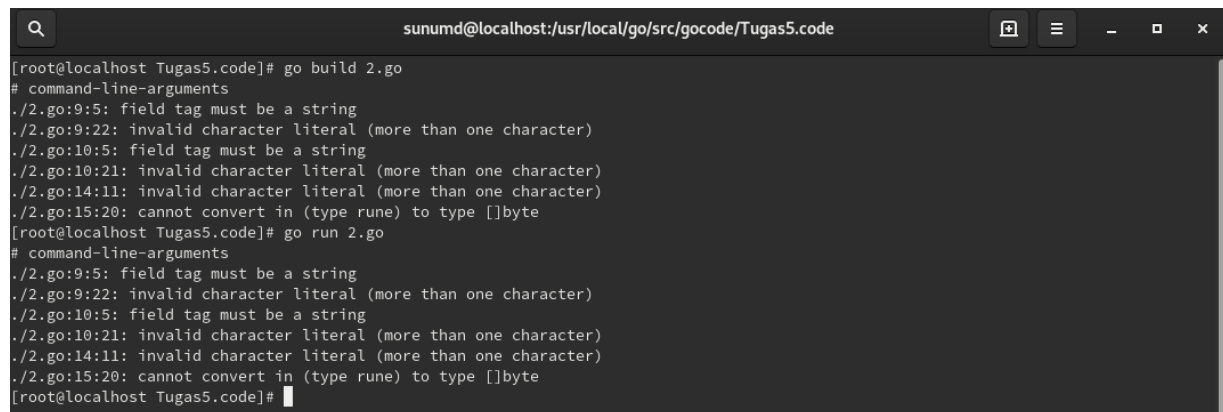
func main() {
    in := `{"firstName":"John","lastName":"Dow"}`
    bytes := []byte(in)

    var p Person
    err := json.Unmarshal(bytes, &p)
    if err != nil {
        panic(err)
    }

    fmt.Printf("%+v", p)
}
```

Jalankan program diatas, apakah outputnya (berikan printscreen) dan jelaskan cara kerjanya!

Jawaban:



The screenshot shows a terminal window with the title 'sunumd@localhost:/usr/local/go/src/gocode/Tugas5.code'. The user runs 'go build 2.go' and 'go run 2.go'. Both commands result in the same error output: '# command-line-arguments ./2.go:9:5: field tag must be a string ./2.go:9:22: invalid character literal (more than one character) ./2.go:10:5: field tag must be a string ./2.go:10:21: invalid character literal (more than one character) ./2.go:14:11: invalid character literal (more than one character) ./2.go:15:20: cannot convert in (type rune) to type []byte'.

```
[root@localhost Tugas5.code]# go build 2.go
# command-line-arguments
./2.go:9:5: field tag must be a string
./2.go:9:22: invalid character literal (more than one character)
./2.go:10:5: field tag must be a string
./2.go:10:21: invalid character literal (more than one character)
./2.go:14:11: invalid character literal (more than one character)
./2.go:15:20: cannot convert in (type rune) to type []byte
[root@localhost Tugas5.code]# go run 2.go
# command-line-arguments
./2.go:9:5: field tag must be a string
./2.go:9:22: invalid character literal (more than one character)
./2.go:10:5: field tag must be a string
./2.go:10:21: invalid character literal (more than one character)
./2.go:14:11: invalid character literal (more than one character)
./2.go:15:20: cannot convert in (type rune) to type []byte
[root@localhost Tugas5.code]#
```

Terdapat error string char

**Soal No 3 (Flatbuffer dan Protocol Buffer)**

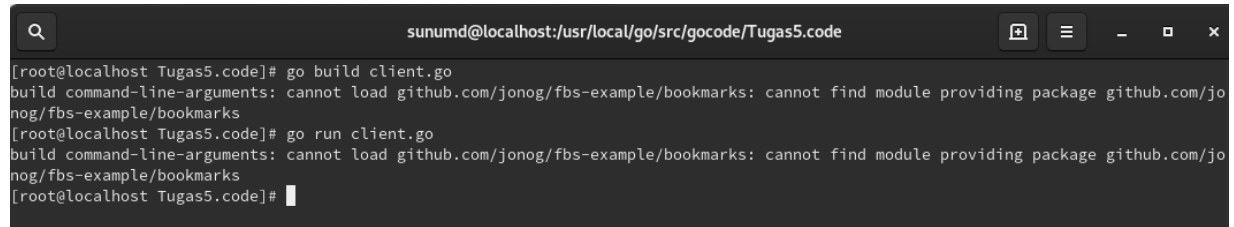
Jalankan program pada repository github berikut: <https://github.com/jonog/grpc-flatbuffers-example>

Berikan analisis berupa:

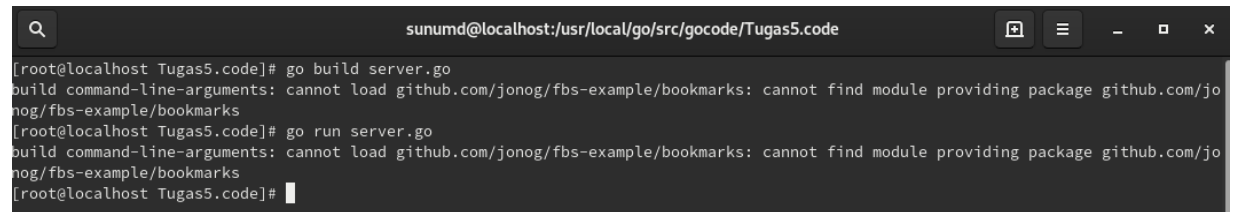
1. Apakah outputnya (berikan printscreen)!
2. Jelaskan cara kerjanya dan buatlah diagram FSMnya!
3. Analisis perbedaan dari protocol buffer dan flatbuffer!

Jawaban:

1.



```
sunumd@localhost:usr/local/go/src/gocode/Tugas5.code
[root@localhost Tugas5.code]# go build client.go
build command-line-arguments: cannot load github.com/jonog/fbs-example/bookmarks: cannot find module providing package github.com/jonog/fbs-example/bookmarks
[root@localhost Tugas5.code]# go run client.go
build command-line-arguments: cannot load github.com/jonog/fbs-example/bookmarks: cannot find module providing package github.com/jonog/fbs-example/bookmarks
[root@localhost Tugas5.code]#
```



```
sunumd@localhost:usr/local/go/src/gocode/Tugas5.code
[root@localhost Tugas5.code]# go build server.go
build command-line-arguments: cannot load github.com/jonog/fbs-example/bookmarks: cannot find module providing package github.com/jonog/fbs-example/bookmarks
[root@localhost Tugas5.code]# go run server.go
build command-line-arguments: cannot load github.com/jonog/fbs-example/bookmarks: cannot find module providing package github.com/jonog/fbs-example/bookmarks
[root@localhost Tugas5.code]#
```

terjadi error saat mencari modul pada package dari github

2.

Cara kerja hampir sama dengan rekursif, namun disini yang disinkronkan adalah pengambilan package modul pada client dan server dari github

3.

Perbedaan dari protokol buffer dengan flatbuffer adalah pada kecepatan serialization yaitu oleh protokol buffer

