GoLang - Getting started

signageOS

Why Go?

- Make it fast
- Make it less consuming CPU/Memory
- Reduce Cloud costs
- Inspire in different programming world with other mindset
- Use existing Go code
- Use existing C code
- Be cool
- Be smart

signageOS

Installation

- https://golang.org/doc/install
- https://marketplace.visualstudio.com/items?itemName=golang.Go
- Or https://www.jetbrains.com/go/promo

```
misak113@keira ~ (go-lecture-1)
$ tar -C /usr/local -xzf go1.15.6.linux-amd64.tar.gz
$ export PATH=$PATH:/usr/local/go/bin
$ go version
```

Cheat sheet

- https://github.com/signageos/go-lecture-1

```
misak113@keira ~ (go-lecture-1)
$ git clone https://github.com/signageos/go-lecture-1.git
$ cd go-lecture-1
$
$ # Decrypt example code
$ go run show-example.go 01 cow
$ # Run your first go code
$ go run 01/ex-print.go

$ # go mod init github.com/signageos/go-lecture-1
```

Cycle problem

```
misak113@keira ~ (go-lecture-1)

$ code 02/my-cycle.go
$ go run 02/my-cycle.go
```

Cycle solution

```
misak113@keira ~ (go-lecture-1)

$ go run show-example.go 02 dog
$ go run 02/ex-cycle.go
```

package + const + func problem

```
misak113@keira ~ (go-lecture-1)

$ code 03/my-pkg.go
$ go run 03/my-pkg.go
```

package + const + func solution

```
misak113@keira ~ (go-lecture-1)

$ go run show-example.go 03 duck
$ go run 03/ex-pkg.go
```

mod + private + init + var problem

```
misak113@keira ~ (go-lecture-1)

$ code 04/my-mod.go
$ go run 04/my-mod.go
```

mod + private + init + var solution

```
misak113@keira ~ (go-lecture-1)

$ go run show-example.go 04 god
$ go run 04/ex-mod.go
```

array + map problem

```
misak113@keira ~ (go-lecture-1)

$ code 05/my-arr.go
$ go run 05/my-arr.go
```

array + map solution

```
misak113@keira ~ (go-lecture-1)

$ go run show-example.go 05 monkey
$ go run 05/ex-arr.go
```

interface + struct + “class” problem

```
misak113@keira ~ (go-lecture-1)

$ code 06/my-oop.go
$ go run 06/my-oop.go
```

interface + struct + “class” solution

```
misak113@keira ~ (go-lecture-1)

$ go run show-example.go 06 big
$ go run 06/ex-oop.go
```

C problem

```
misak113@keira ~ (go-lecture-1)

$ code 07/my-cgo.go
$ go run 07/my-cgo.go
```

C solution

```
misak113@keira ~ (go-lecture-1)

$ go run show-example.go 07 wow
$ go run 07/ex-cgo.go
```

Tests + Lint + Build problem

```
misak113@keira ~ (go-lecture-1)

$ code 08/my-build.go
$ code 08/my-build_test.go
$ go run 08/my-build.go
$ go test ./08/...
```

Tests + Lint + Build solution

```
misak113@keira ~ (go-lecture-1)

$ go run show-example.go 08 noway
$ go run 08/ex-build.go
$ go test ./08/...
$ go test -bench=. ./08/...
$ go build ./08/ex-build.go
$ go run golang.org/x/lint/golint ./...
$ go run github.com/mgechev/revive -formatter stylish ./...
```

Inspiration

- https://github.com/docker/cli
- https://vsupalov.com/go-folder-structure/
- https://www.wolfe.id.au/2020/03/10/how-do-i-structure-my-go-project/

- https://golang.org/doc/
- https://gobyexample.com/
- Google: “golang …”

OVERALL TEST SCORE
40 OUT OF 42 TEST SCENARIOS PASSED

95 %

signageOS

Michael Zabka

QUESTIONS?

signageOS