Intro to Golang

August 2018

Google Cloud

Topics

Hello World (9 to Noon)	*Workstation Setup (go, ide), Your First App, Language Overview and Exercises.*
Interfaces and Concurrency (1:30 - 3:30)	*Develop a basic understanding of two of the most valuable features in golang.*
Golang Challenge (3:30 - 5)	*Write some code for fame and prizes.*
Next Steps (30 Minutes)	*References, books, blogs and getting help.*

Google Cloud

Workstation Setup

Google Cloud

Setup Go

Install Go for your platform from https://golang.org/dl/

Setup you folder structure https://golang.org/doc/code.html

(Your Home Folder) / (gopath folder)   /src
/pkg
/bin

This training will do all of its work under the src folder.

Setup Ide

Atom - https://atom.io

Visual Studio Code - https://code.visualstudio.com/

Goland - https://www.jetbrains.com/go/specials/go/go.html?dclid=CJzE5LDG4NwCFeVuwQodWy4Pow

LiteIde - https://github.com/visualfc/liteide

VIM - https://github.com/fatih/vim-go

Emacs - https://github.com/dominikh/go-mode.el

Explore Go Command

The commands are:

bug	start a bug report
build	compile packages and dependencies
clean	remove object files and cached files
doc	show documentation for package or symbol
env	print Go environment information
fix	update packages to use new APIs
fmt	gofmt (reformat) package sources
generate	generate Go files by processing source
get	download and install packages and dependencies
install	compile and install packages and dependencies
list	list packages or modules
mod	module maintenance
run	compile and run Go program
test	test packages
tool	run specified go tool
version	print Go version
vet	report likely mistakes in packages

Getting the Samples Code and Slides

Commands:

cd $GOPATH/src/

git clone https://github.com/goog-lukemc/go-train

Slide and docs are in $GOPATH/src/assets

Source is in $GOPATH/src/go-train/<item>

Background

History

Who:

Robert Griesemer
Rob Pike
Ken Thompson

Why:

Combine the easy of a dynamically type language with the safety of the statically type system.
https://golang.org/doc/faq#Is_Go_an_object-oriented_language
https://golang.org/doc/effective_go.html
https://golang.org/doc/code.html

Where are we now:

Go 1.10 - 1.11 Aug 2018
https://golang.org/dl/

More Info:

https://talks.golang.org/2012/splash.article
https://tip.golang.org/doc/go1.11
https://golang.org/doc/devel/release.html
https://talks.golang.org/2015/gophercon-goevolution.slide#8
https://golang.org/doc/faq

What’s Cool about Go (Top 3)

Concurrency: (More After Lunch)

- Concurrency is not parallelism (https://www.youtube.com/watch?v=cN_DpYBzKso)
- Concurrency is about having the best design to maximize parallelism if it is available.

Interfaces:

Covered after lunch

Portability:

- Go is not runtime interpreted (There is nothing to install on the target to execute a go program)
- A simple build switch can build the executable for any supported platform.

Google Cloud

Coders vs Developers

Coders

Writing code so solve the problem in front of you. Slinging Code - Having fun!

Developers

Writing code to solve a problem for generic reuse. Write small - Having fun!

Idiomatic

What is this anyways? - https://golang.org/doc/effective_go.html

Google Cloud

Hello World Time

01

~/go-train/hello

Review - main.go (In Editor)

Build - main.go (In Editor)

Build for any platform from any platform.

```
Pi:
env GOOS=linux GOARCH=arm go build -v main.go -o program-arm
MAC:
env GOOS=darwin GOARCH=amd64 go build -v main.go -o program-mac-amd64
Windows:
env GOOS=windows GOARCH=amd64 go build -v main.go -o program-windows-amd64.exe
```

Running the file and flow testing:
go run main.go

~/go-train/hello_flag

Review - main.go (In Editor)

Build - main.go (In Editor)

Build for any platform from any platform.

```
Pi:
env GOOS=linux GOARCH=arm go build -v main.go -o program-arm
MAC:
env GOOS=darwin GOARCH=amd64 go build -v main.go -o program-mac-amd64
Windows:
env GOOS=windows GOARCH=amd64 go build -v main.go -o program-windows-amd64.exe
```

Running the file and flow testing:
go run main.go

Google Cloud

~/go-train/hello_struct

Review - main.go (In Editor)

Build - main.go (In Editor)

Build for any platform from any platform.

```
Pi:
env GOOS=linux GOARCH=arm go build -v main.go -o program-arm
MAC:
env GOOS=darwin GOARCH=amd64 go build -v main.go -o program-mac-amd64
Windows:
env GOOS=windows GOARCH=amd64 go build -v main.go -o program-windows-amd64.exe
```

Running the file and flow testing:

```
go build
go run main.go
```

The Basics

02

Google Cloud

errors ~/go-train/errors

Review - main.go (In Editor)

Build for any platform from any platform.

```
Pi:
env GOOS=linux GOARCH=arm go build -v main.go -o program-arm
MAC:
env GOOS=darwin GOARCH=amd64 go build -v main.go -o program-mac-amd64
Windows:
env GOOS=windows GOARCH=amd64 go build -v main.go -o program-windows-amd64.exe
```

Running the file and flow testing:
go run main.go

Google Cloud

basic ~/go-train/basics

Review - main.go (In Editor)

Build for any platform from any platform.

Pi:

```
env GOOS=linux GOARCH=arm go build -v main.go -o program-arm
```

MAC:

```
env GOOS=darwin GOARCH=amd64 go build -v main.go -o program-mac-amd64
```

Windows:

```
env GOOS=windows GOARCH=amd64 go build -v main.go -o program-windows-amd64.exe
```

Running the file and flow testing:
go run main.go

basic ~/go-train/asciicoolness

Review - main.go (In Editor)

Build for any platform from any platform.

```
Pi:
env GOOS=linux GOARCH=arm go build -v main.go -o program-arm
MAC:
env GOOS=darwin GOARCH=amd64 go build -v main.go -o program-mac-amd64
Windows:
env GOOS=windows GOARCH=amd64 go build -v main.go -o program-windows-amd64.exe
```

Running the file and flow testing:
go run main.go

basic ~/go-train/basichttpserver

Review - main.go (In Editor)

Build for any platform from any platform.

```
Pi:
env GOOS=linux GOARCH=arm go build -v main.go -o program-arm
MAC:
env GOOS=darwin GOARCH=amd64 go build -v main.go -o program-mac-amd64
Windows:
env GOOS=windows GOARCH=amd64 go build -v main.go -o program-windows-amd64.exe
```

Running the file and flow testing:
go run main.go

03

Lunch time

Google Cloud

Concurrency

04

concurrency ~/go-train/basichttpserver

Review - main.go (In Editor)

Build for any platform from any platform.

```
Pi:
env GOOS=linux GOARCH=arm go build -v main.go -o program-arm
MAC:
env GOOS=darwin GOARCH=amd64 go build -v main.go -o program-mac-amd64
Windows:
env GOOS=windows GOARCH=amd64 go build -v main.go -o program-windows-amd64.exe
```

Running the file and flow testing:
go run main.go

concurrency ~/go-train/sametime

Review - main.go (In Editor)

Build for any platform from any platform.

```
Pi:
env GOOS=linux GOARCH=arm go build -v main.go -o program-arm
MAC:
env GOOS=darwin GOARCH=amd64 go build -v main.go -o program-mac-amd64
Windows:
env GOOS=windows GOARCH=amd64 go build -v main.go -o program-windows-amd64.exe
```

Running the file and flow testing:
go run main.go

Testing

05

Google Cloud

Let’s Make Something TDD Style for prizes and fame.

Survey:
https://goo.gl/forms/tj22IlUtigE0DfHP2

We Done

06

Google Cloud