

Herd ing

Gophers













aaroncruz.com



Bring your startup idea to life

I want you to succeed

Not a lot of products become the next big thing. This becomes even harder to achieve when you don't have all of the necessary tools to build, experiment with and market your product. I have those tools and I would love to be the first step on your trip to being a profitable business. I want to give the best return on investment possible.

Start earning money now

Building your MVP can be a real challenge if you don't have a team of developers. I have years of experience building user focused web applications and I would love to build you a revenue producing business this month!

@mraaroncruz

Aaron Cruz
@mraaroncruz

If you're wondering why I look like this, 40%
of my diet is ham

3:54 PM - 3 Oct 2014

github.com/pferdefleisch

The screenshot shows a GitHub user profile for the account "pferdefleisch". The top navigation bar includes links for "Explore", "Gist", "Blog", and "Help". The user's profile picture and name are on the right, along with icons for adding a repository, settings, and more. A central message states: "We are having a problem billing the [REDACTED] organization. Please update your payment method or call your payment provider for details on why the transaction failed. If you don't need access to your private repositories, you can downgrade to the Free plan. We'll try billing [REDACTED] again on February 8, 2015. You can always contact support@github.com with any questions." Below this message, there are links for "News Feed", "Pull Requests", and "Issues".



Search GitHub

Explore Gist Blog Help



pferdefleisch

+ ▾ ☰ ⚙ ↗

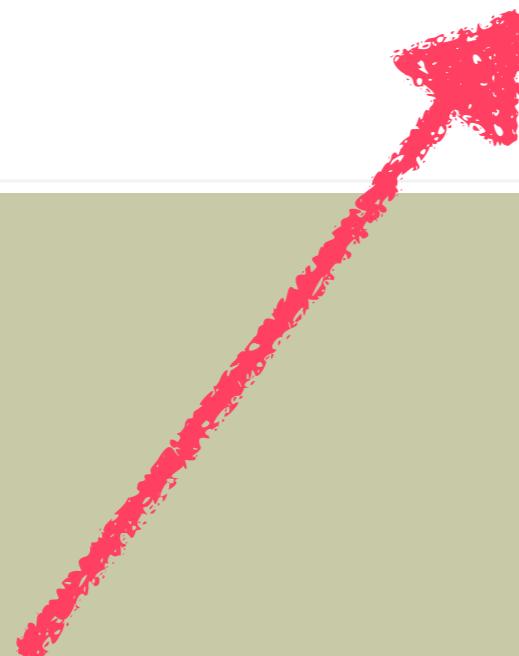
We are having a problem billing the [REDACTED] organization. Please [update your payment method](#) or call your payment provider for details on why the transaction failed. If you don't need access to your private repositories, you can [downgrade to the Free plan](#). We'll try billing [REDACTED] again on February 8, 2015. You can always contact support@github.com with any questions.

pferdefleisch ▾

News Feed

Pull Requests

Issues



A photograph of a soccer player in a white jersey and red gloves scoring a goal. The ball is in the net, and the player's arms are raised in celebration. A young child in a green shirt is visible in the bottom left corner.

Goals



CHESTNUT ST
1600 UNT



STOP

VOLDEMORT

30 Day Challenge

Next API will be written in Go

write idiomatic Go

“an expression of a simple task,
algorithm, or data structure that is *not* a
built-in feature in the programming
language”

-Wikipedia (Programming Idiom)

```
select {
    case msg1 := <- c1:
        fmt.Println(msg1)
    case msg2 := <- c2:
        fmt.Println(msg2)
}
```

```
go func() {
    for {
        select {
        case msg1 := <- c1:
            fmt.Println(msg1)
        case msg2 := <- c2:
            fmt.Println(msg2)
        case <- time.After(time.Second):
            fmt.Println("timeout")
        }
    }
}
```

Repos in list To Do

Description [Edit](#)

Link to typeform

<https://pferdefleisch.typeform.com/to/DySddM>

Golang Repo Links

Here are lists of repos from the most prolific Go project creators

Hashicorp

- [consul](#)
- [terraform](#)
- [serf](#)
- [raft](#)
- [memberlist](#)

gorilla

- [websocket](#)
- [mux](#)
- [context](#)
- [pat](#)

codegangsta

- [cli](#)
- [negroni](#)
- [gin](#)
- [inject](#)
- [go-martini/martini](#)

spf13

- [spf13/hugo](#)

Other

- [bitly/nsq](#)
- [boltdb/bolt](#)
- [coreos/etcd](#)

Add

[Members](#)

[Labels](#)

[Checklist](#)

[Due date](#)

[Attachment](#)

Actions

[Move](#)

[Copy](#)

[Subscribe](#)

[Archive](#)

[Share and more...](#)

Results

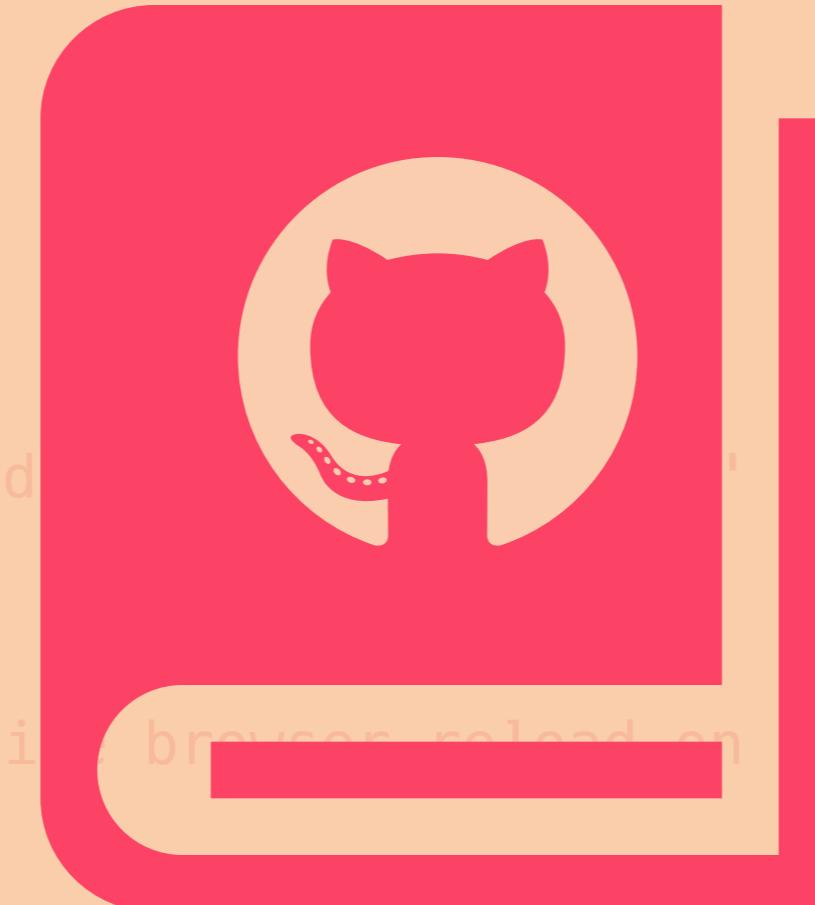
One repo a day is hard

One repo a week

Half hour to an hour per day

How to effectively read

source code



var
var
var
var

//var ServerCmdV *cobra.Command

var

Use
Short
Long

Hugo will render all the files defined in the source directory and serve them up.

 //Run: server,
}

func

 serverCmd

the server on"

 serverCmd

filesystem for changes and

 serverCmd

port to baseURL"

 serverCmd

"watch without enabling live reload on rebuild"

 serverCmd

 serverCmd

(requires --memstats)"

 serverCmd

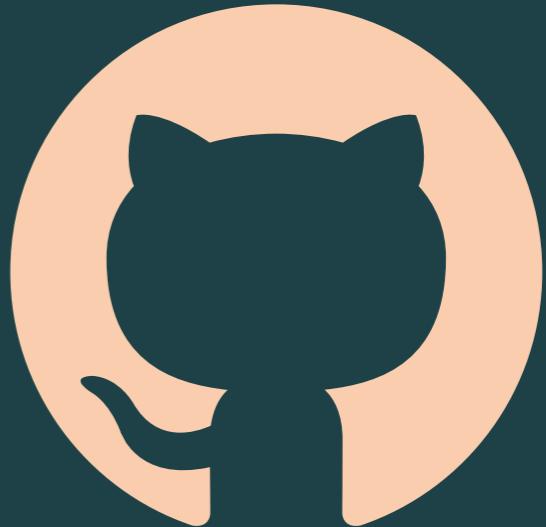
}

Choose a mode

active



passive



Learn the API

Read the docs and README

Get familiar with
codegangsta/cli

github search finds 2,992 results

go get github.com/foo/bar

then open code in editor

sb1m \$GOPATH/src/github.com/foo/bar

Find your entry points

if binary grep for `func main`
for library grep for `package packagename`

Download the Go tools for your editor

vim-go
GoSublime
go-plus for atom

SIDENOTE

if you are not using `goimports`, download
it and set it up now!

“What we call chaos is just patterns we haven’t recognized. What we call random is just patterns we can't decipher.”

-Chuck Palahniuk

Record what you find

I use typeform and trello



BARGAS 6









The Sheep







Give your errors names

```
// ErrTooLarge is passed to panic if memory cannot
// be allocated to store data in a buffer.
var ErrTooLarge = errors.New("bytes.Buffer: too large")
```

```
func makeSlice(n int) []byte {
    // If the make fails, give a known error.
    defer func() {
        if recover() != nil {
            panic(ErrTooLarge)
        }
    }()
    return make([]byte, n)
}
```

golang.org/src/bytes/buffer.go

Factories Everywhere

```
type runner struct {
    bin          string
    args         []string
    writer       io.Writer
    command     *exec.Cmd
    starttime   time.Time
}

func NewRunner(bin string, args ...string) Runner {
    return &runner{
        bin:      bin,
        args:    args,
        writer:  ioutil.Discard,
        starttime: time.Now(),
    }
}
```

github.com/codegangsta/gin/lib/runner.go

The Goats









```
func (client *Client) sendCommand(cmd string, args ...string) (data interface{},  
err error) {  
    // grab a connection from the pool  
    var b []byte  
    c, err := client.popCon()  
    if err != nil {  
        println(err.Error())  
        goto End  
    }  
  
    b = commandBytes(cmd, args...)  
    data, err = client.rawSend(c, b)  
    if err == io.EOF {  
        c, err = client.openConnection()  
        if err != nil {  
            println(err.Error())  
            goto End  
        }  
  
        data, err = client.rawSend(c, b)  
    }  
  
End:  
  
    //add the client back to the queue  
    client.pushCon(c)  
  
    return data, err  
}
```

```
func (client *Client) sendCommand(cmd string, args ...string) (data interface{},  
err error) {  
    // grab a connection from the pool  
    var b []byte  
    c, err := client.popCon()  
    if err != nil {  
        println(err.Error())  
        goto End  
    }  
  
    b = commandBytes(cmd, args...)  
    data, err = client.rawSend(c, b)  
    if err == io.EOF {  
        c, err = client.openConnection()  
        if err != nil {  
            println(err.Error())  
            goto End  
        }  
  
        data, err = client.rawSend(c, b)  
    }  
  
    End:  
  
    //add the client back to the queue  
    client.pushCon(c)  
  
    return data, err  
}
```

Global Globals

```
var dbClient *Client
```

in a different file

```
func API_GetTables(c *gin.Context) {  
    names, err := dbClient.Tables()  
  
    // ...  
}
```

Read Source Code

as your new leisure activity

Look for the patterns

good and bad

Get out of your
comfort zone

but don't forget to set goals

A photograph of a flock of sheep grazing in a field at sunset. The sheep are white with dark faces and legs, scattered across a field of green grass. In the background, rolling hills are silhouetted against a bright orange and yellow sky where the sun is setting. The overall atmosphere is peaceful and pastoral.

Thank you