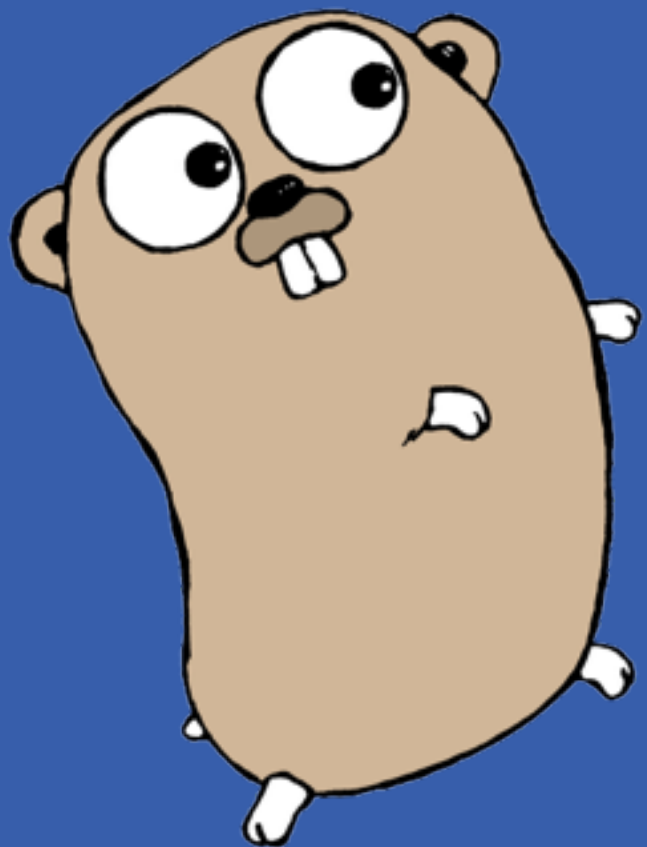
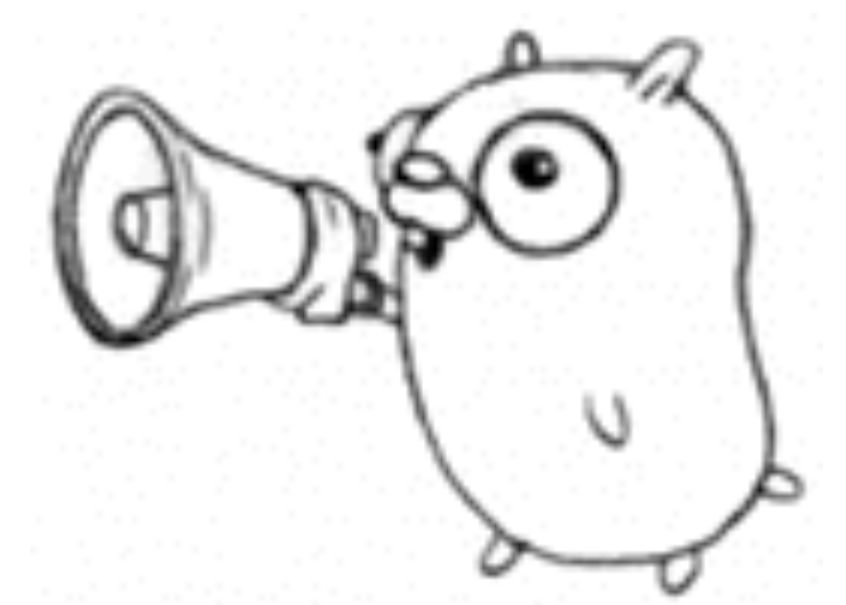


# Go 101



@brianfoshee

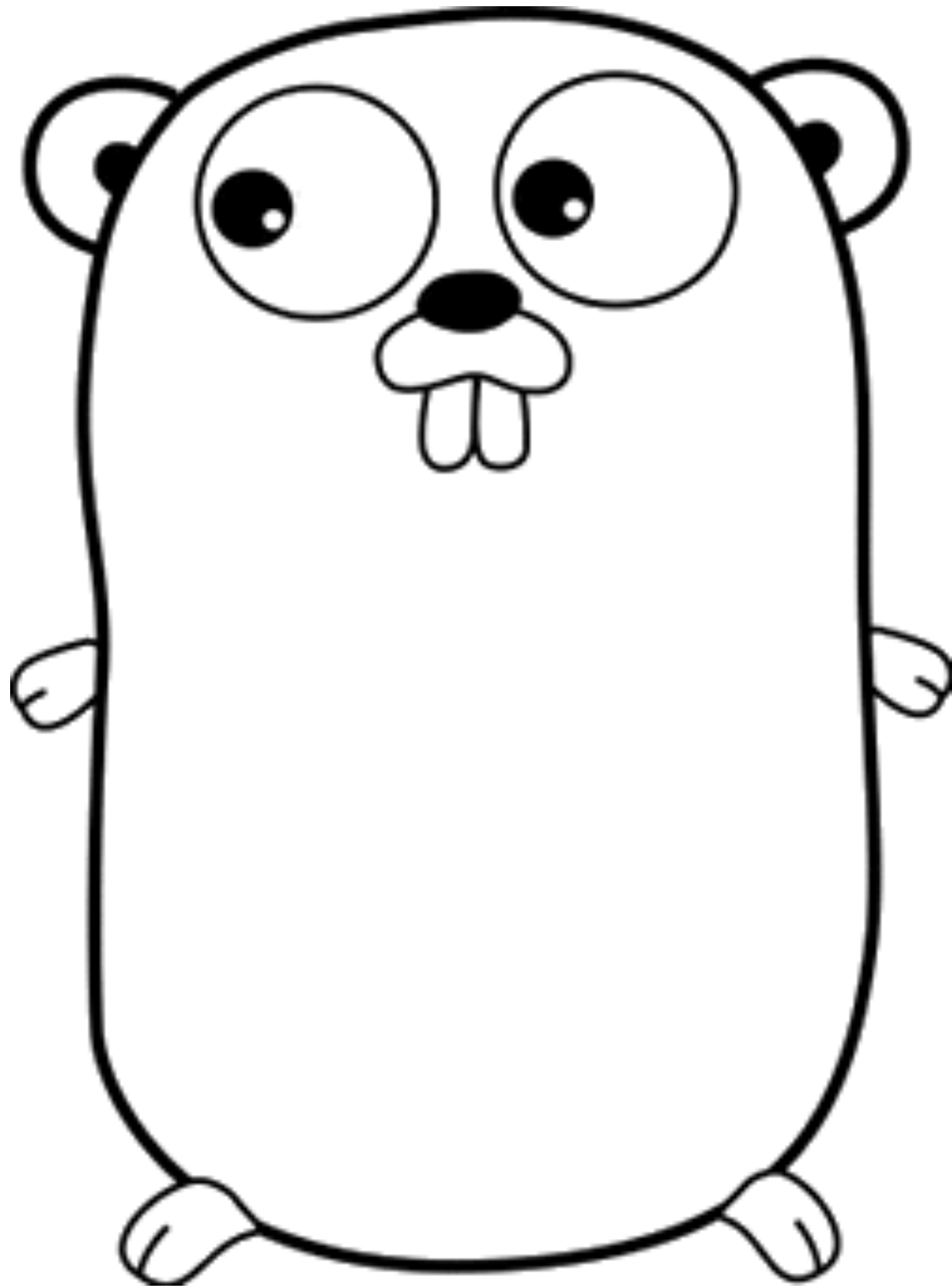


- **What is Go?**
- **Development Environment**
- **Development Tools**
- **Code**

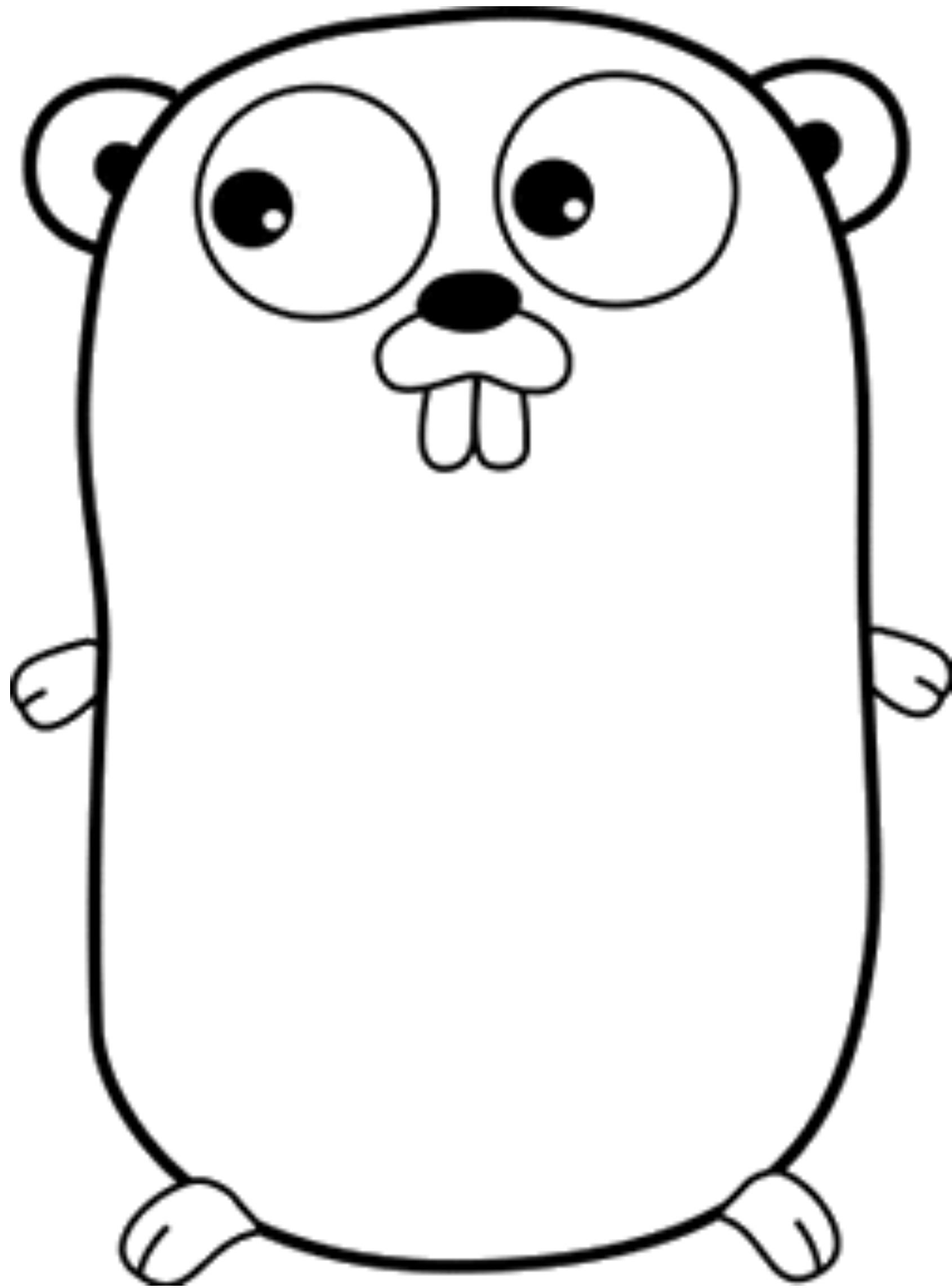
[github.com/brianfoshee/go-101](https://github.com/brianfoshee/go-101)

**“Go is an open source programming language that makes it easy to build simple, reliable, and efficient software.”**

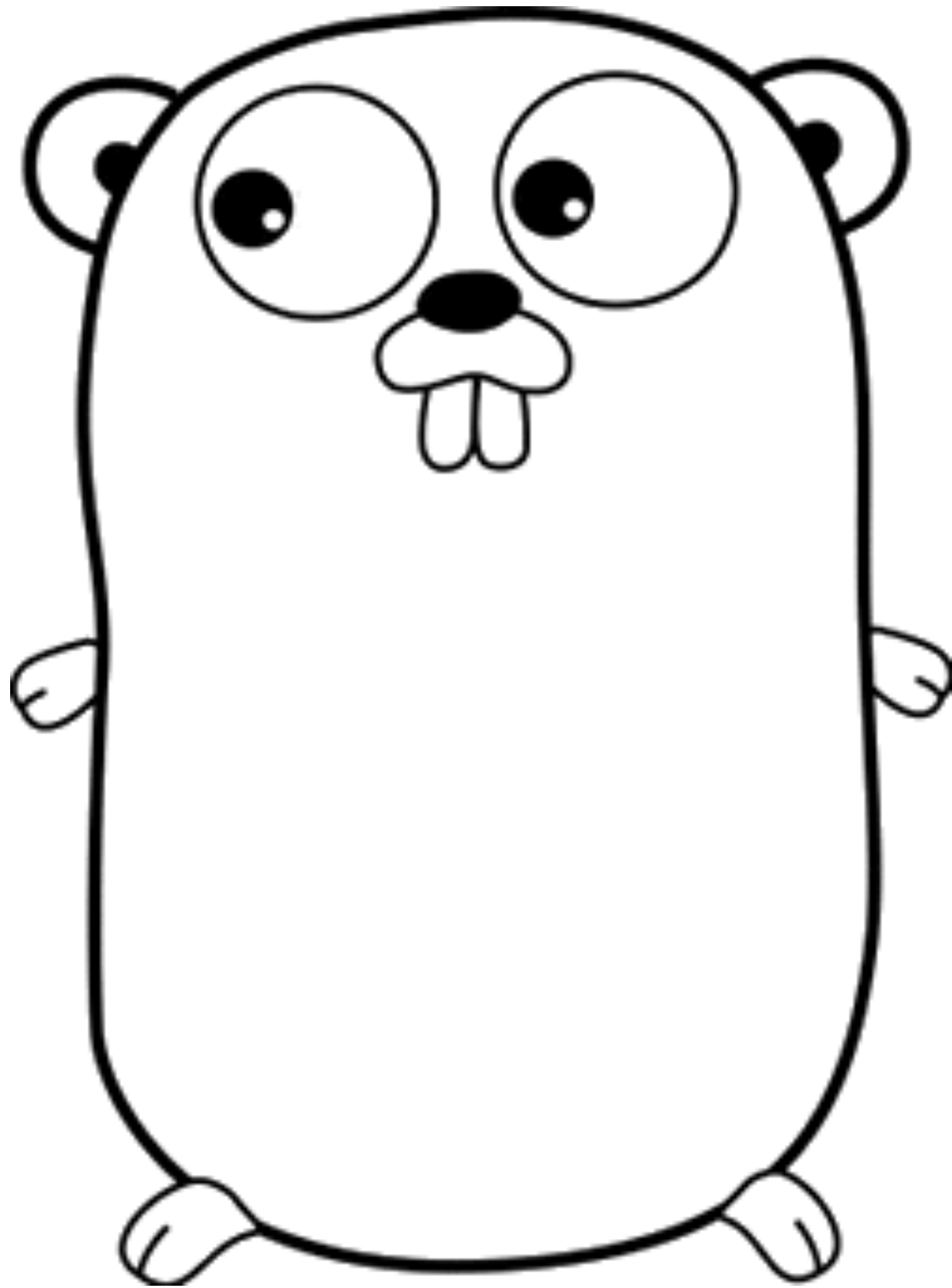
[golang.org](https://golang.org)



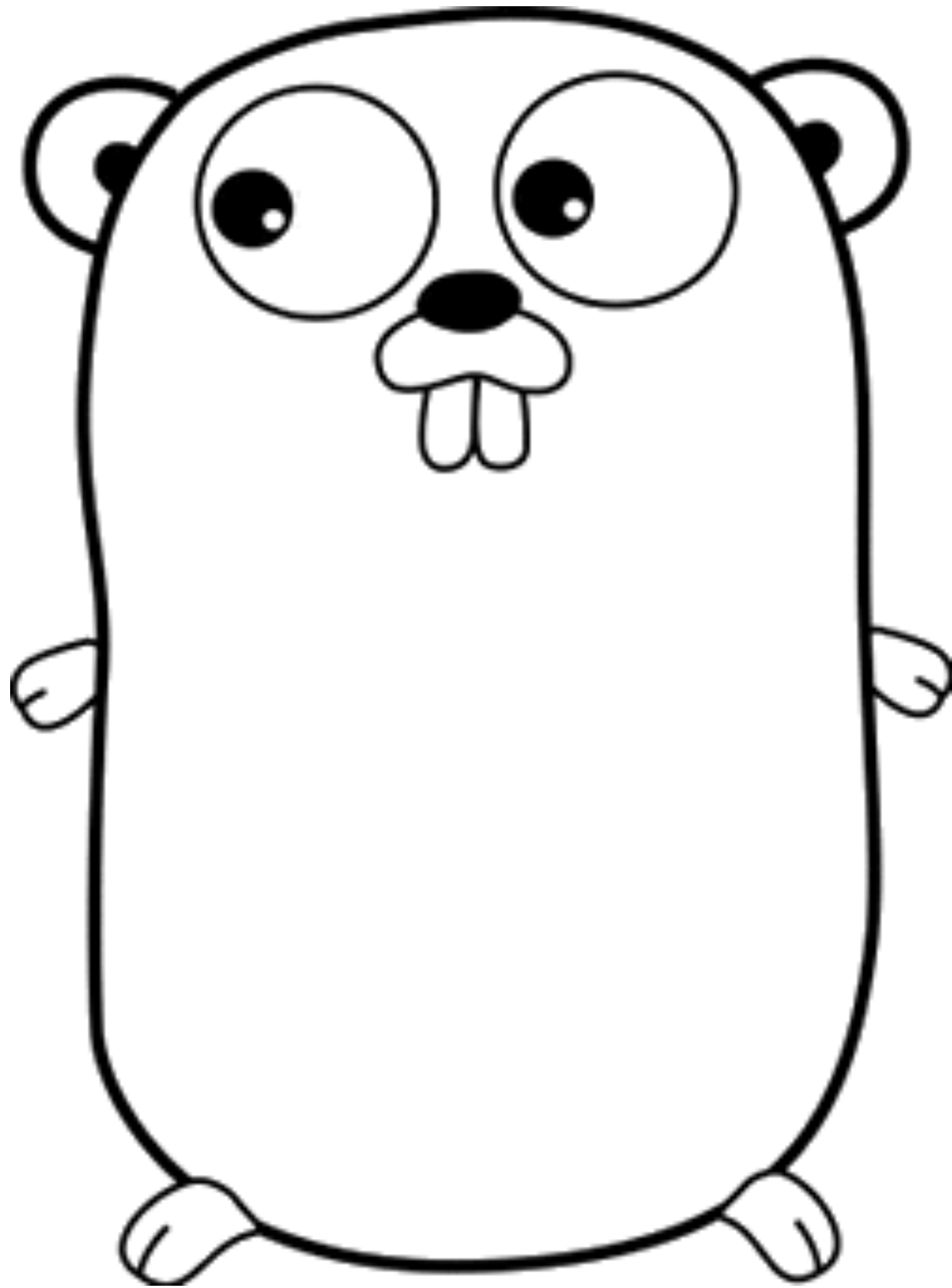
- **Statically Typed**



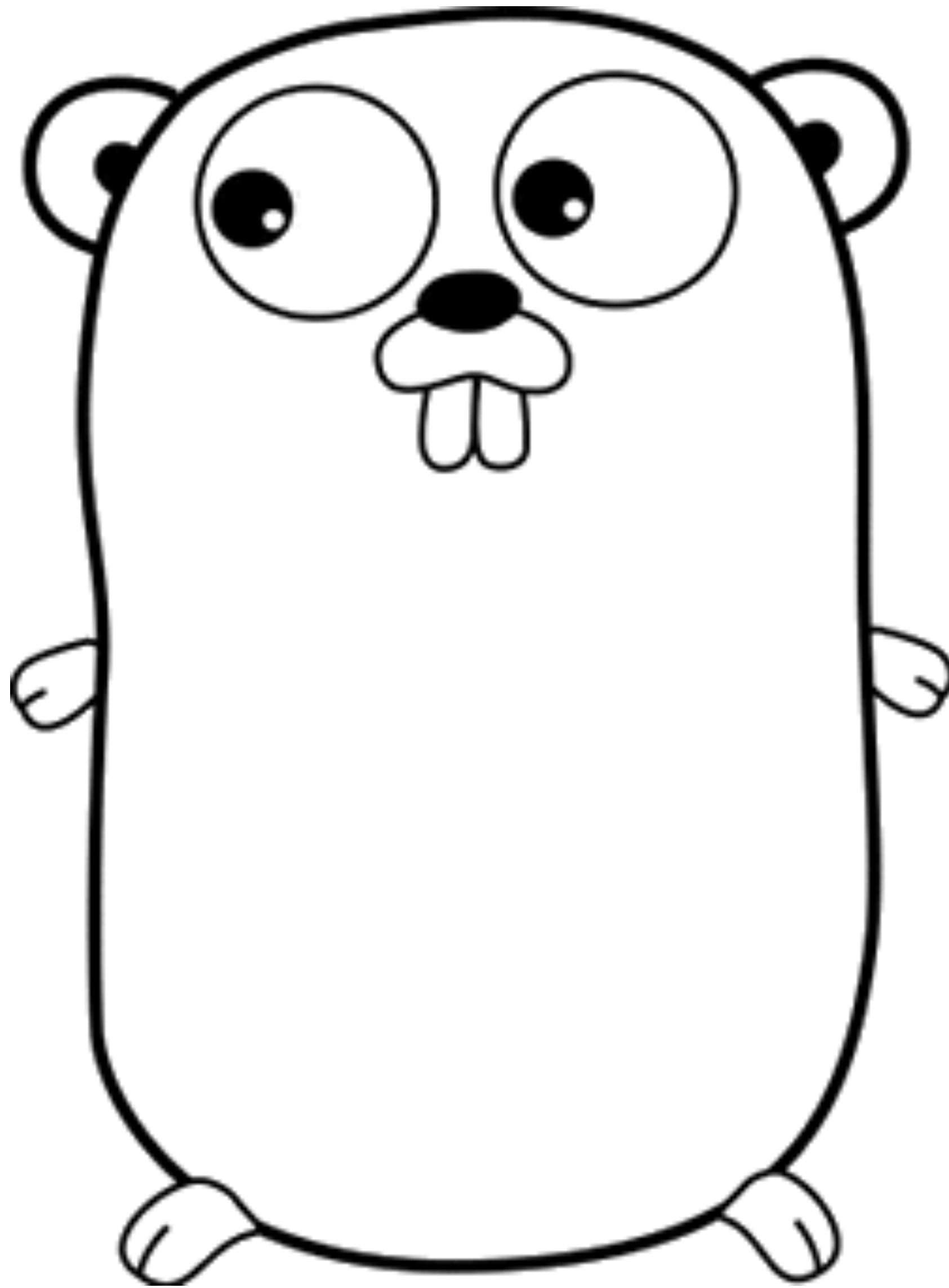
- **Statically Typed**
- **Compiled**



- **Statically Typed**
- **Compiled**
- **Garbage Collected**



- **Statically Typed**
- **Compiled**
- **Garbage Collected**
- **Object Oriented (ish)**



- **Statically Typed**
- **Compiled**
- **Garbage Collected**
- **Object Oriented (ish)**
- **Cross Platform**

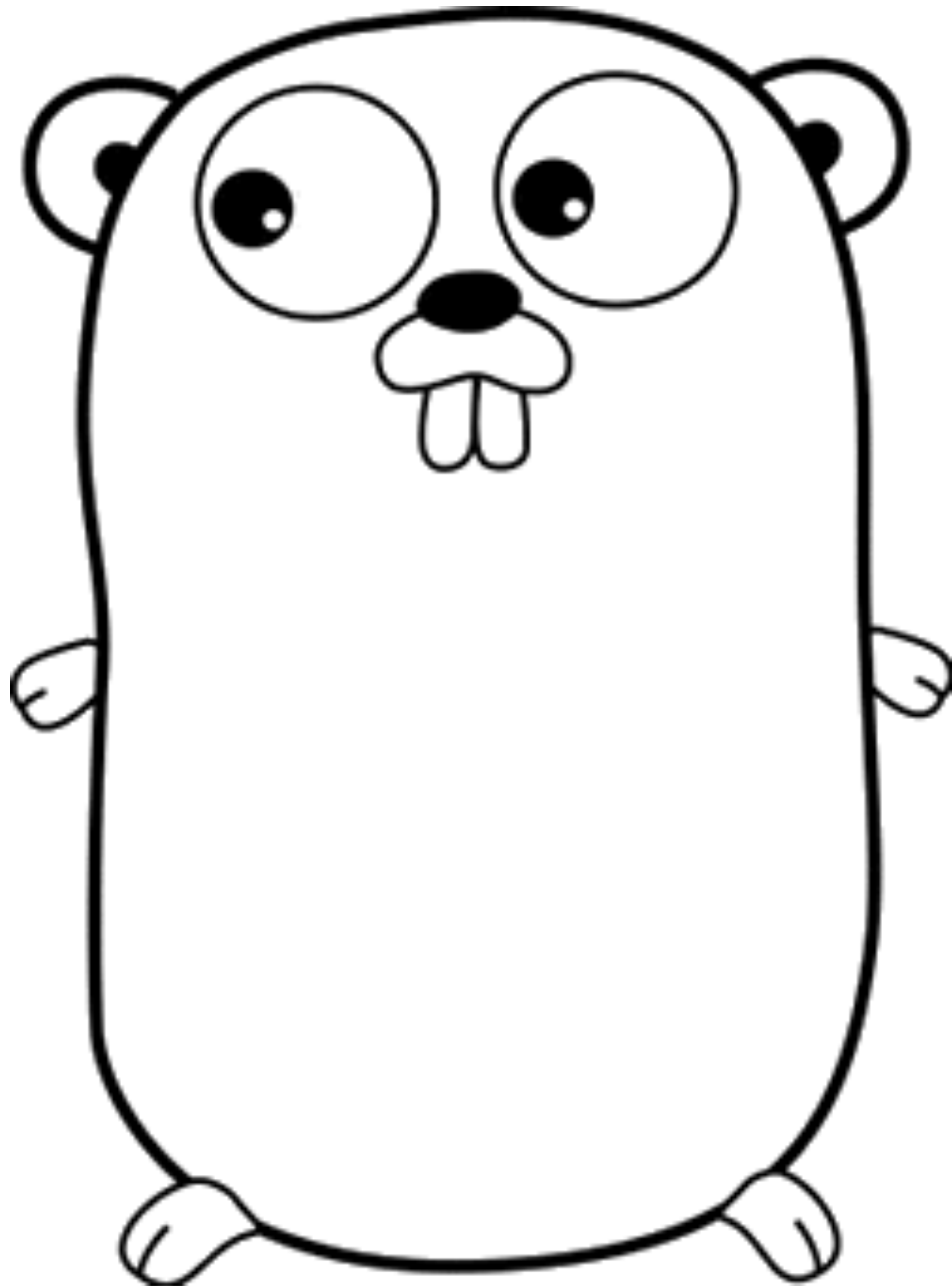


[Go](#) [Gccgo](#) release-branch.go1.5 show only first-class ports

				darwin			freebsd				linux														windows			race			android	openbsd			netbsd	dragonfly	nacl			plan9			solaris								
				386	x64	arm64	386	x64	x64	arm	arm	386	386	386	386	x64	x64	x64	x64	x64	arm	arm	arm	arm64	arm64	mips64	mips64le	ppc64	ppc64le	386	x64	x64	freebsd	linux	windows	arm	386	x64	arm	386	x64	386	amd64p32	arm	386	x64	arm	x64			
71a7647	austin	15 Oct 19:33	[release-branch.g...	ok	ok		ok	ok	ok	fail		ok	ok	ok	ok	ok	ok	ok	ok	fail			ok	ok	fail	fail	fail		ok	ok	ok	ok	ok	ok		ok	ok	fail	ok	ok	fail	ok	fail	fail	ok						
c257dfb	austin	12 Oct 21:50	[release-branch.g...	ok	ok		ok	ok	ok	fail		ok	ok	ok	ok	ok	ok	ok	ok	fail			ok	ok	fail	fail	fail		ok	ok	ok	ok	ok	ok		ok	ok	fail	ok	ok	fail	ok	fail	fail	ok						
6f21d7c	austin	03 Oct 08:00	doc: update relea...	ok	ok		ok	ok	ok	fail		ok	ok	ok	ok	ok	ok	ok	ok	ok	ok	ok	ok	ok	fail	fail	fail		ok	ok	ok	ok	ok	ok		ok	ok	fail	ok	ok	fail	ok	fail	fail	ok						
3c5af9a	cbro	23 Sep 04:10	[release-branch.g...	ok	ok		ok	ok	ok	fail		ok	ok	ok	ok	ok	ok	ok	ok	ok	ok	ok	ok	fail	fail	fail	fail	fail	ok	ok	ok	ok	ok	ok		ok	ok		ok	ok	fail	ok	ok	fail	ok	fail	ok				
f2e4c8b	cbro	09 Sep 00:52	[release-branch.g...		ok	ok	ok	ok	ok			ok	ok	ok	ok	ok	ok	ok	ok				ok	ok		fail	fail	ok	ok	ok	ok	ok	ok	ok		ok	ok		ok	ok		ok	ok		ok	fail	ok				
e7915b2	cbro	09 Sep 00:50	[release-branch.g...		ok	ok	ok	ok	ok			ok	ok	ok	ok	ok	ok	ok	ok				ok	ok		fail	fail	ok	ok	ok	ok	ok	ok	ok		ok	ok		ok	ok		ok	ok		ok	fail	ok				
23c646c	r	08 Sep 23:23	[release-branch.g...		ok	ok	ok	ok	ok			ok	ok	ok	ok	ok	ok	ok	ok				ok	ok		fail	fail	ok	ok	ok	ok	ok	ok	ok		ok	ok		ok	ok		ok	ok		ok	fail	ok				
71387ff	r	08 Sep 23:19	[release-branch.g...		ok	ok	ok	ok	ok			ok	ok	ok	ok	ok	ok	ok	ok				ok	ok		fail	fail	ok	ok	ok	ok	ok	ok	ok		ok	ok		ok	ok		ok	ok		ok	fail	ok				
5e1648d	r	08 Sep 22:51	[release-branch.g...		ok	ok	ok	ok	ok			ok	ok	ok	ok	ok	ok	ok	ok				ok	ok		fail	fail	ok	ok	ok	ok	ok	ok	ok		ok	ok		ok	ok		ok	ok		ok	fail	ok				
7765a5b	bra...	08 Sep 22:51	[release-branch.g...		ok	ok	ok	ok	ok			ok	ok	ok	ok	ok	ok	ok	ok				ok	ok		fail	fail	ok	ok	ok	ok	ok	ok	ok		ok	ok		ok	ok		ok	ok		ok	fail	ok				
878aef8	cbro	08 Sep 22:38	Revert "[release-b...		ok	ok	ok	ok	ok			ok	ok	ok	ok	ok	ok	ok	ok				ok	ok		fail	fail	ok	ok	ok	ok	ok	ok	ok		ok	ok		ok	ok		ok	ok		ok	fail	ok				
f265044	austin	08 Sep 18:02	[release-branch.g...		ok	ok	ok	ok	ok	fail		ok	ok	ok	ok	ok	ok	ok	ok		ok	ok		fail	fail	fail	fail	ok	ok	ok	ok	ok	ok	ok		ok	ok		ok	ok		ok	ok		ok	fail	ok				
c0dd201	bra...	08 Sep 18:01	[release-branch.g...	fail	ok		ok	ok	ok			ok	ok	ok	ok	ok	ok	ok	ok				ok	fail		fail	fail	ok	ok	fail	ok	ok	ok	ok		ok	ok		ok	ok		ok	fail	ok							
a7e30ce	rsc	08 Sep 18:01	[release-branch.g...		ok	ok	ok	ok	ok			ok	ok	ok	ok	ok	ok	ok	ok				fail	fail		fail	fail	ok	ok	fail	ok	ok	ok	ok		ok	ok		ok	ok		ok	ok		ok	fail	ok				
1d11801	vin...	08 Sep 18:01	[release-branch.g...		ok	ok	ok	ok	ok			ok	ok	ok	ok	ok	ok	ok	ok				ok	ok		fail		ok	ok	fail	ok	ok	ok	ok		ok	ok		ok	ok		ok	ok		ok	fail	ok				
0507802	uli...	08 Sep 18:01	[release-branch.g...		ok	fail	ok	ok	ok			ok	ok	ok	ok	ok	ok	ok	ok				ok	ok		fail		ok	ok	ok	ok	ok	ok	ok		ok	ok		ok	ok		ok	ok		ok	fail	ok				
23ef1e1	austin	08 Sep 06:04	[release-branch.g...		ok	ok	ok	ok	ok	fail		ok	ok	ok	ok	ok	ok	ok	ok	fail	ok	ok	ok	ok		fail	fail	ok	ok	ok	ok	ok	ok	ok		ok	ok		ok	ok		ok	ok	fail	ok	fail	ok				
f7d7403	austin	08 Sep 06:04	[release-branch.g...		ok	ok	ok	ok	ok	fail		ok	ok	ok	ok	ok	ok	ok	ok	fail	ok	ok	ok	ok		fail	fail	ok	ok	ok	ok	ok	ok	ok		ok	ok		ok	ok		ok	ok	fail	ok	fail	ok				
e4e5992	austin	08 Sep 06:04	[release-branch.g...		ok	ok	ok	ok	ok	fail		ok	ok	ok	ok	ok	ok	ok	ok		ok	ok	ok	ok		fail	fail	ok	ok	fail	ok	ok	ok	ok		ok	ok		ok	ok		ok	ok	fail	ok	fail	ok				
13d03fa	ale...	08 Sep 05:41	[release-branch.g...		ok	ok	ok	ok	ok	fail		ok	ok	ok	ok	ok	ok	ok	ok				ok	ok		fail	fail	ok	ok	ok	ok	ok	ok	ok		ok	ok		ok	ok		ok	ok	fail	ok	fail	ok				
97ec0a8	didi...	08 Sep 05:41	[release-branch.g...		ok	ok	ok	ok	ok	fail		ok	ok	ok	ok	ok	ok	ok	ok				ok	ok		fail	fail	ok	ok	ok	ok	ok	ok	ok		fail	ok		ok	ok		ok	ok		ok	fail	ok				
2dfb0eb	iant	08 Sep 01:52	[release-branch.g...		ok	fail	ok	ok	ok	fail		ok	ok	ok	ok	ok	ok	ok	ok	fail	ok	ok	ok	ok		fail	fail	ok	ok	ok	ok	ok	ok	ok		ok	ok		ok	ok		ok	ok	fail	fail	fail	ok				
d4f1309	minux	06 Sep 01:14	cmd/link/internal/l...		ok	ok	ok	ok	ok	fail		ok	ok	ok	ok	ok	ok	ok	ok	fail	ok	ok	ok	ok		fail	fail	ok	ok	ok	ok	ok	ok	ok		ok	ok		ok	ok		ok	ok	fail	ok	fail	ok				
c20b8e1	dave	04 Sep 02:10	[release-branch.g...		ok	ok	ok	ok	ok	fail		ok	ok	ok	ok	ok	ok	ok	ok				ok	ok				ok	ok	ok	ok	ok	ok	ok		ok	ok		ok	ok		ok	ok		ok	fail	ok				
5aa3ba8	minux	03 Sep 08:53	[release-branch.g...	fail	ok	ok	ok	ok	ok	fail		ok	ok	ok	ok	ok	ok	ok	ok	fail	ok	ok	ok					ok	ok	ok	ok	ok	ok	ok		ok	ok		ok	ok		ok	ok	fail	ok	fail	ok				
a1350a1	adg	03 Sep 03:17	[release-branch.g...		ok	ok	ok	ok	ok	fail		ok	ok	ok	ok	ok	ok	ok	ok	fail	ok	ok	ok					ok	ok	ok	ok	ok	ok	ok		ok	ok		ok	ok		ok	ok	fail	ok	fail	ok				
00dbc5a	adg	03 Sep 03:02	[release-branch.g...		ok	ok	ok	ok	ok	fail		ok	ok	ok	ok	ok	ok	ok	ok	fail	ok	ok	ok					ok	ok	ok	ok	ok	ok	ok		ok	ok		ok	ok		ok	ok	fail	ok	fail	ok				
bb03def	rsc	19 Aug 05:04	[release-branch.g...		ok	ok	ok	ok	ok			ok	ok	ok	ok	ok	ok	ok	ok				ok	ok	ok		fail	ok	ok	fail	ok	ok	ok	ok		fail	ok		ok	ok		ok	ok	fail	ok	ok					
a30edb8	rsc	19 Aug 04:39	[release-branch.g...		ok	ok	ok	ok	ok			ok	ok	ok	ok	ok	ok	ok	ok				ok	ok	ok		fail	ok	ok	fail	ok	ok	ok	ok		ok	ok		fail	ok		ok	ok	ok	ok	ok					
0d20a61	adg	06 Aug 04:06	[release-branch.g...		ok	ok	ok	ok	ok	fail		ok	ok	ok	ok	ok	ok	fail	ok	ok	ok	ok					ok	ok	ok	fail	ok	ok	ok	ok		ok	ok		ok	ok		ok	ok	fail	ok	ok					

[newer](#) [older](#) [latest](#)





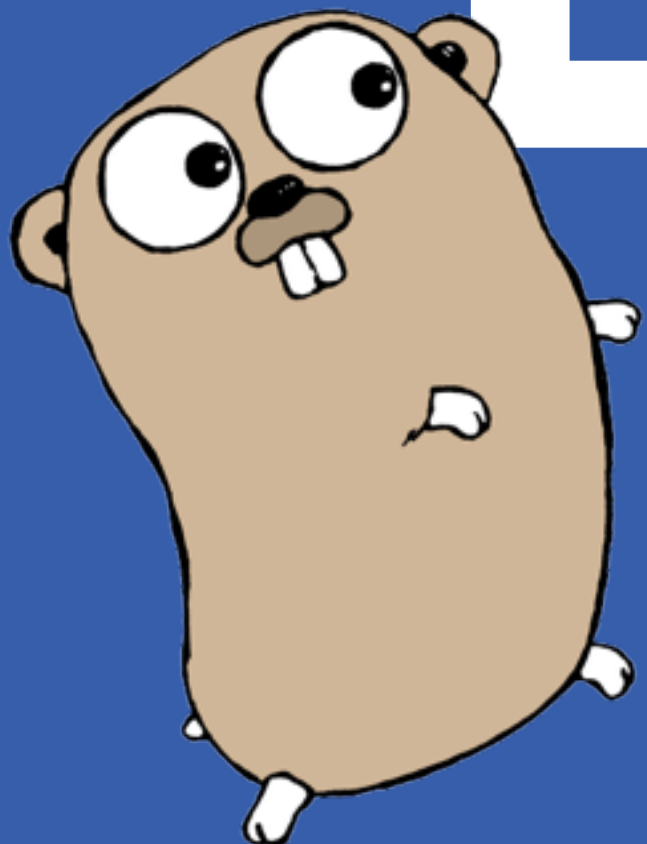
- **Statically Typed**
- **Compiled**
- **Garbage Collected**
- **Object Oriented (ish)**
- **Cross Platform**
- **Concurrent**

```
package main
```

```
import "fmt"
```

```
func main() {  
    fmt.Println("Hello, world!")  
}
```

# Development Environment





# The Go Programming Language

[Documents](#)[Packages](#)[The Project](#)[Help](#)[Blog](#)

## Downloads

After downloading a binary release suitable for your system, please follow the [installation instructions](#).

If you are building from source, follow the [source installation instructions](#).

See the [release history](#) for more information about Go releases.

### Featured downloads

#### Microsoft Windows

*Windows XP or later, Intel 64-bit processor*

**[go1.5.1.windows-amd64.msi](#)** (64MB)

SHA1: 0a439f49b546b82f85adf84a79bbf40de2b3d5ba

#### Apple OS X

*OS X 10.8 or later, Intel 64-bit processor*

**[go1.5.1.darwin-amd64.pkg](#)** (74MB)

SHA1: 857b77a85ba111af1b0928a73cca52136780a75d

#### Linux

*Linux 2.6.23 or later, Intel 64-bit processor*

**[go1.5.1.linux-amd64.tar.gz](#)** (74MB)

SHA1: 46eecd290d8803887dec718c691cc243f2175fe0

#### Source

**[go1.5.1.src.tar.gz](#)** (11MB)

SHA1: 0df564746d105f4180c2b576a1553ebca9d9a124

# Workspace

- ***src/*** contains Go source files organized into packages (one package per directory)
- ***pkg/*** contains package objects
- ***bin/*** contains executable commands



To give you an idea of how a workspace looks in practice, here's an example:

```
bin/
  hello                # command executable
  outyet               # command executable
pkg/
  linux_amd64/
    github.com/golang/example/
      stringutil.a     # package object
src/
  github.com/golang/example/
    .git/              # Git repository metadata
    hello/
      hello.go         # command source
    outyet/
      main.go          # command source
      main_test.go     # test source
    stringutil/
      reverse.go       # package source
      reverse_test.go  # test source
```

This workspace contains one repository (example) comprising two commands (hello and outyet) and one library (stringutil).

☢ [go] ls

bin pkg src

☢ [go] pwd

/Users/brian/Code/go

☢ [go] ls src

9fans.net

code.google.com

golang.org

azul3d.org

gary.burd.info

gopkg.in

bitbucket.org

github.com

gopkg.com

☢ [go] ls src/github.com/brianfoshee

app-analytics

autoingestion

gosolar

aquaponics-data

concurrency

hello

acquire

go-101

mosaic



```
export GOPATH=$HOME/Code/go  
PATH=$PATH:$GOPATH/bin  
PATH=$PATH:/usr/local/go/bin
```



## Package names

The first statement in a Go source file must be

```
package name
```

where *name* is the package's default name for imports. (All files in a package must use the same *name*.)

Go's convention is that the package name is the last element of the import path: the package imported as "crypto/rot13" should be named rot13.

Executable commands must always use package main.

There is no requirement that package names be unique across all packages linked into a single binary, only that the import paths (their full file names) be unique.

```
package main
```

```
import (  
    "flag"  
    "fmt"
```

```
    "github.com/brianfoshee/autoingestion/properties"  
)
```

```
func main() {  
    var fn string  
    flag.StringVar(&fn, "file", "autoingestion.properties", "Pro  
    flag.Parse()  
  
    p := properties.NewPropertiesFromFile(fn)  
  
    fmt.Printf("Email: %s Password: %s\n", p.Email, p.Password)  
}
```



```
package properties
```

```
...
```

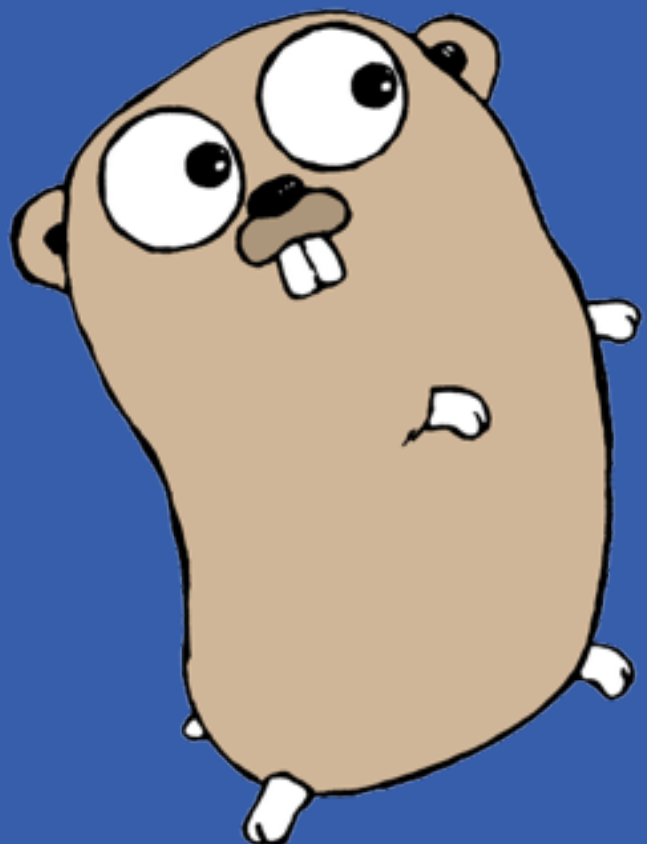
```
type Properties struct {  
    Email    string  
    Password string  
}
```

```
func NewPropertiesFromFile(name string) Properties {  
    p := Properties{}  
    p.fromFile(name)  
    return p  
}
```

```
func (p *Properties) fromFile(name string) {  
    ...  
}
```

```
☢ [~] go get github.com/gorilla/websocket
```

# Development Tools



go build



go install

go fmt

go vet

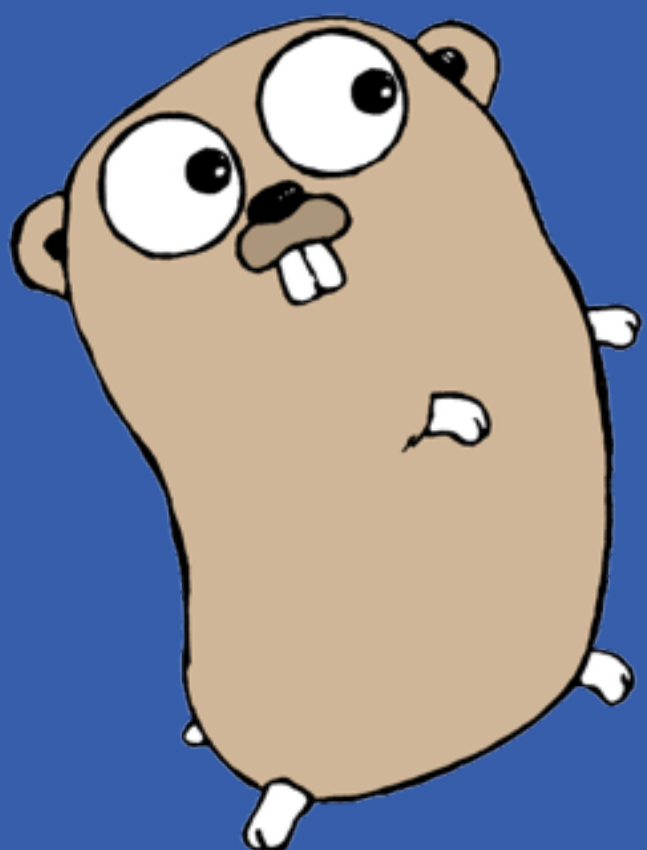
golint

goimports

godoc

go test

# Code





- [golang.org/doc/code.html](http://golang.org/doc/code.html)
- [golang.org/doc/effective\\_go.html](http://golang.org/doc/effective_go.html)
- [golang.org/cmd/go](http://golang.org/cmd/go)
- [tour.golang.org](http://tour.golang.org)
- [play.golang.org](http://play.golang.org)
- [blog.golang.org](http://blog.golang.org)
- [talks.golang.org](http://talks.golang.org)
- [golang.org/pkg](http://golang.org/pkg)
- [reddit.com/r/golang](http://reddit.com/r/golang)
- [forum.golangbridge.org](http://forum.golangbridge.org)
- [bit.ly/go-slack-signup](http://bit.ly/go-slack-signup)
- [Gophercon 2015 Videos](#)

