



Packages in Go

# PACKAGE SYNTAX

```
package pkg
```

```
// code
```



# PACKAGE STRUCTURE

package network

tcp.go

udp.go

network.go

package main

main.go

start.go

init.go



# PACKAGE TYPES



## ■ **main (executable)**

```
go run *.go
```

```
go build -o exec
```








**func main**

## ■ **non main (non executable)**

```
import "github.com/gophertuts/go-basics/pkg"
```

# PACKAGE NAMING & STRUCTURE

 src/network	
<code>package network</code>	
 src/net	
<code>package network</code>	
 src/net_err	
<code>package net_err</code>	
 src/netErr	
<code>package netErr</code>	
 src/utls	
<code>package utls</code>	

 src/pkg	
 src/sub_pkg	
 src/sub_pkg	
 src/sub_pkg	
 src/network	
network.go	
tcp.go	
udp.go	



# PACKAGE IMPORTS

```
package pkg
```

```
import "github.com/gophertuts/go-basics/net"
```

HOST

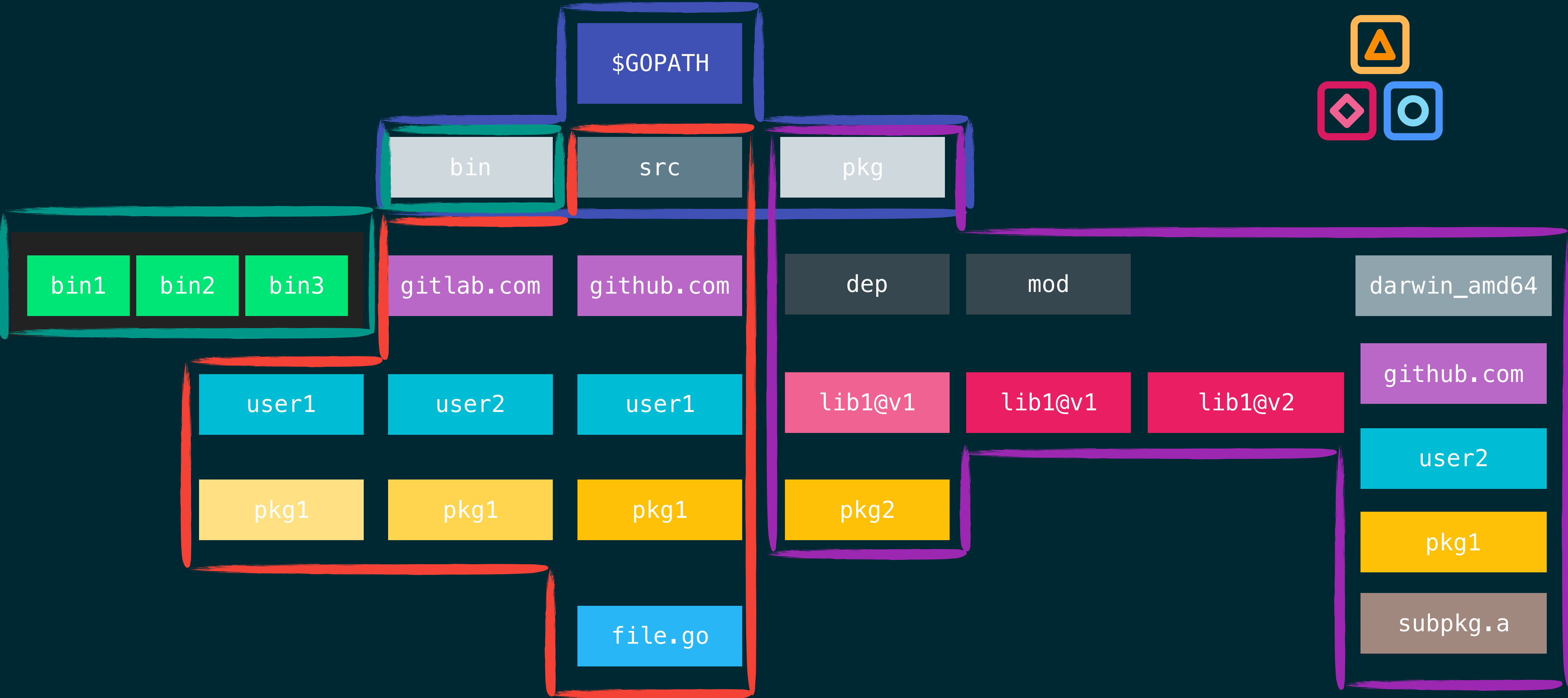
USER/ORGANIZATION

PROJECT/REPO

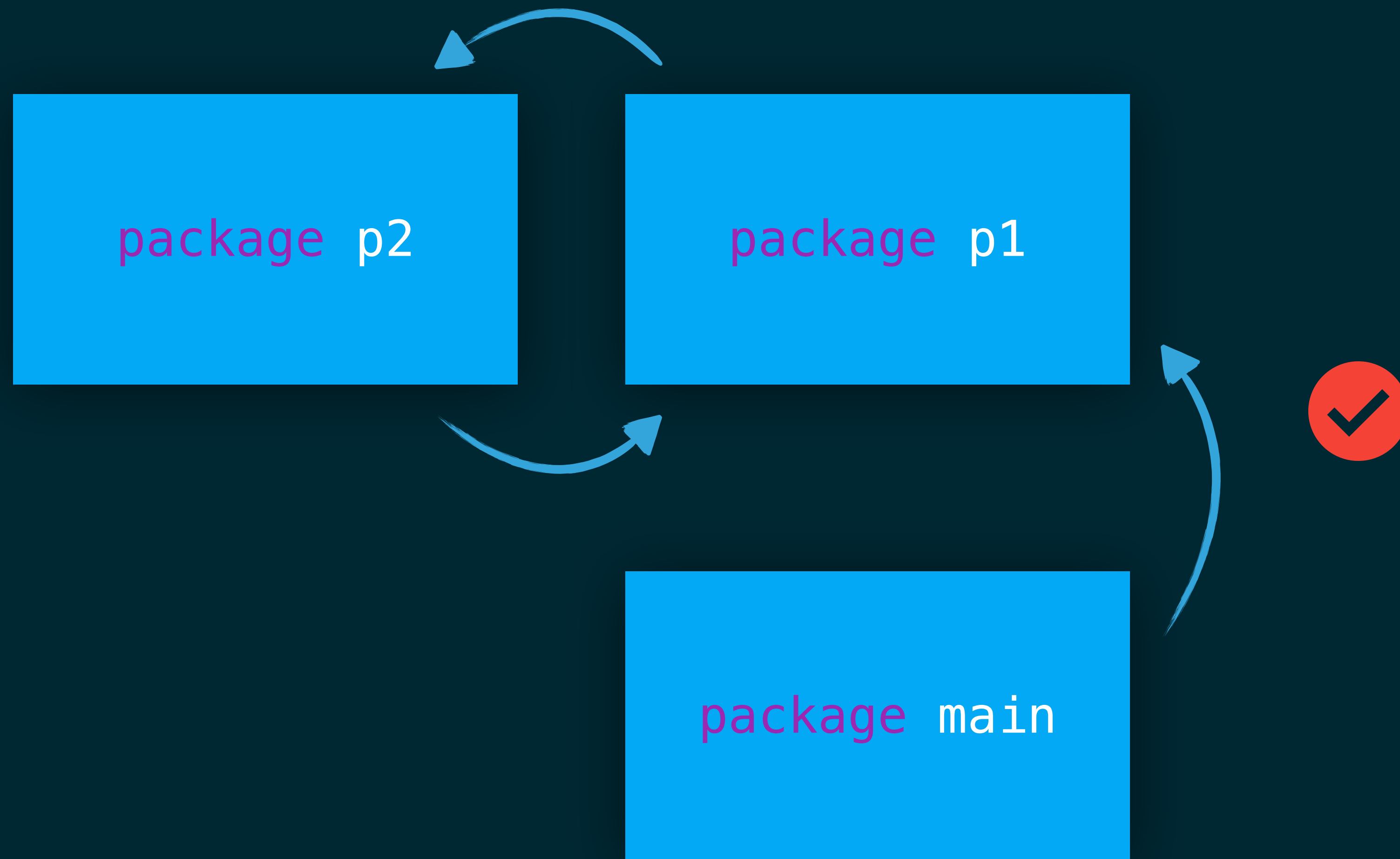
PACKAGE



# \$GOPATH



# CROSS REFERENCE





# FILE GLOBALS & LOCALS

## Locals

package name

imports

## Globals

constants

variables

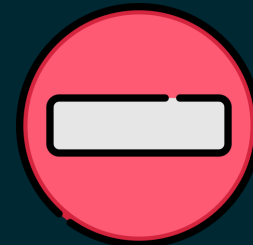
functions



# TEST FILES

p1\_test.go

package p1



p2\_test.go

package p2



# PACKAGE VISIBILITY (SCOPE)



## Exported

---

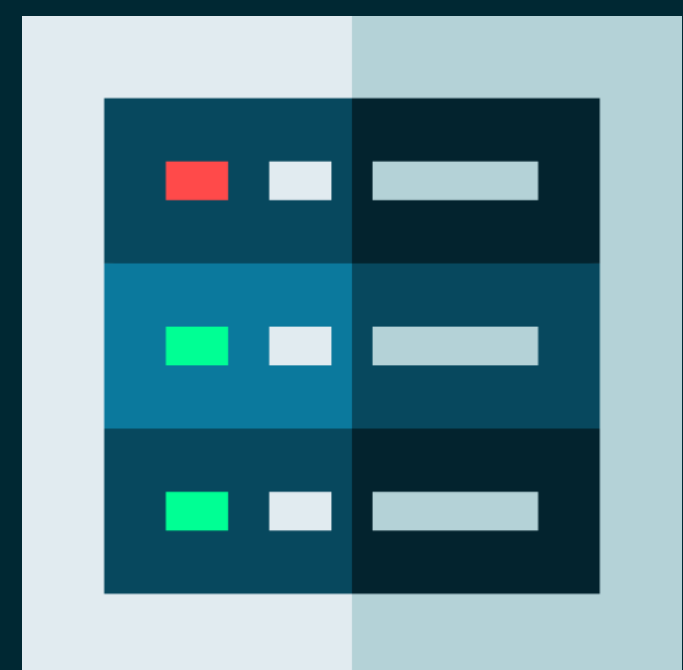
Visible & accessible in all files  
of the same package

## unExported

---

Visible & accessible inside the package, and  
inaccessible outside of it

# GO GET



github.com



steevehook



repo



`$GOPATH/src`

```
go get ./...
```