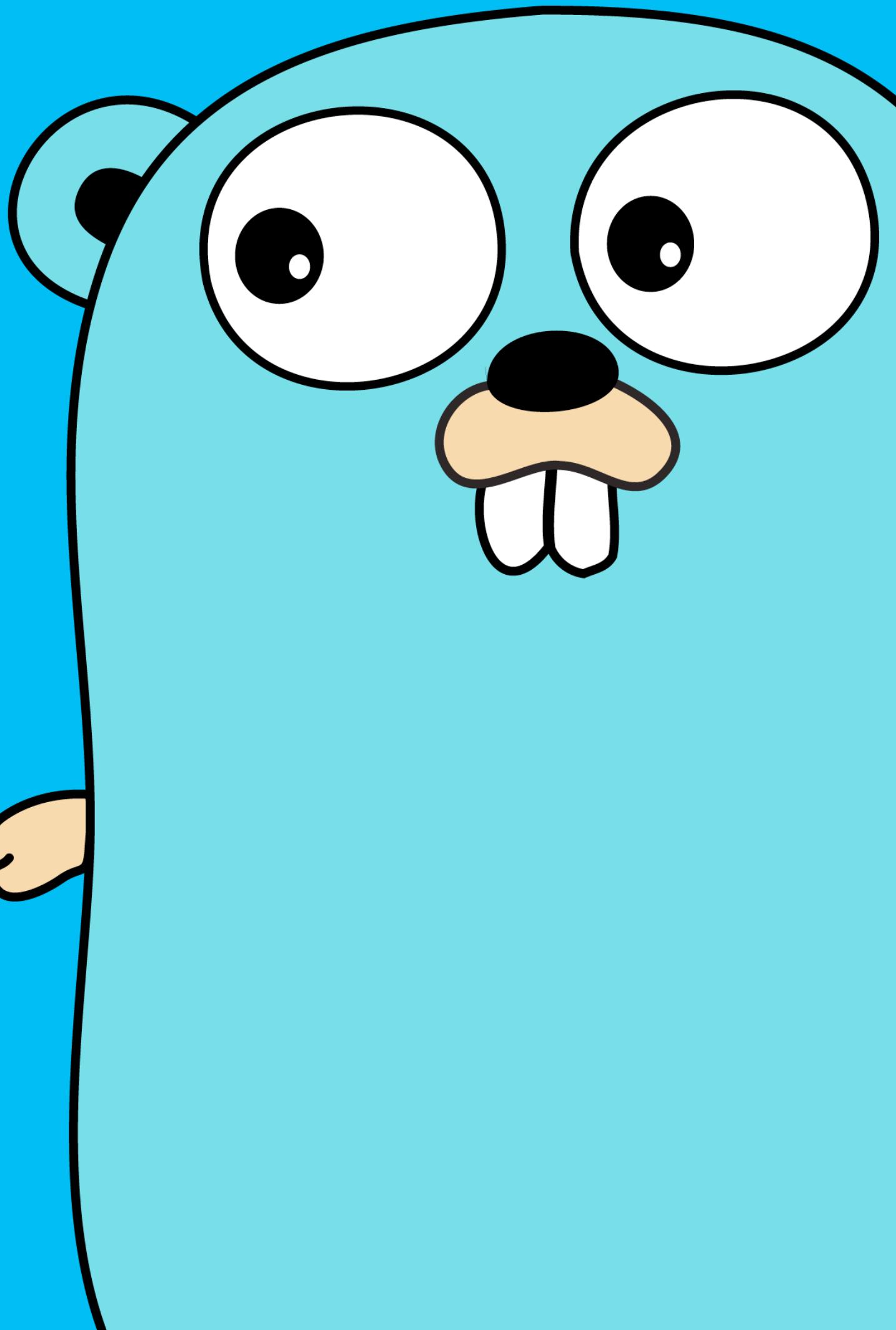


WHAT'S NEW IN GO 1.24

Helsinki Gophers

2025-02-26



ANTTI KUPILA

- Senior Software Engineer at ICEYE
- Helsinki Gophers organizer
- akupila on GitHub



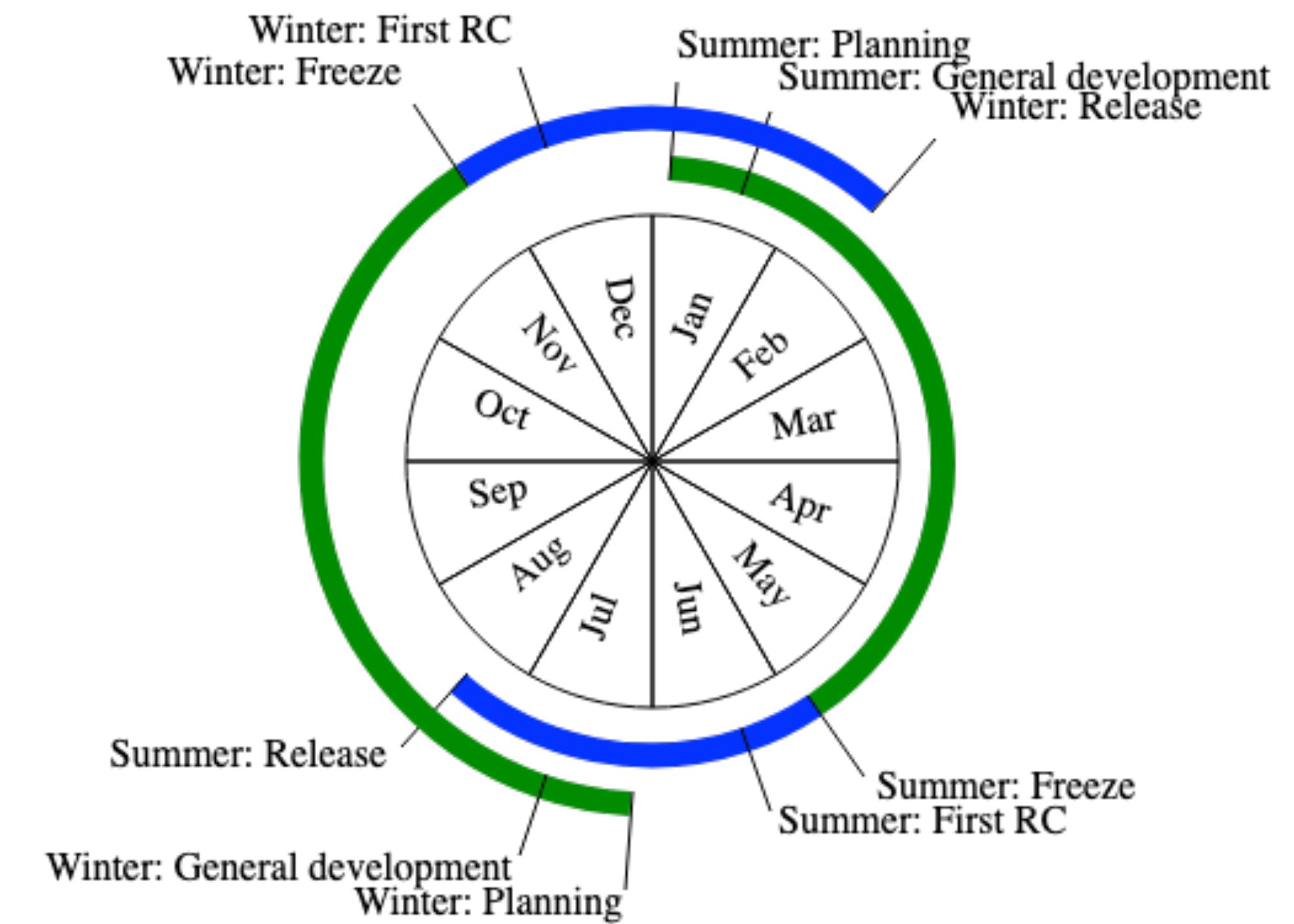
AGENDA

- Go release cycle & proposal process
- A tip on gotip
- 10 things in 1.24

RELEASE CYCLE

- **January / July week:** Planning for release begins
- **May / November:** Release freeze begins
- **June / December:** Release candidate 1 issued
- **August / February:** Release issued

<https://go.dev/s/release>



proposal: review meeting minutes #33502

New issue 

 Open



rsc opened on Aug 7, 2019 · edited by rsc

Edits  ...

The [proposal review group meets regularly](#) (roughly weekly) to review pending proposal issues and move them along in the [proposal process](#).

Review consists primarily of checking that discussion is ongoing, commenting on issues to move discussion along, summarizing long discussions, CC'ing experts, and accepting or closing proposals when the discussion on the GitHub issue has reached a clear consensus about the outcome.

Review also includes closing proposals that are untenable (for example, because the changes are [backwards-incompatible](#) or violate key design goals of the language or packages).

This meta-issue records minutes of the proposal review meetings as issue comments, so that they can be cross-linked easily with the relevant issues. (This meta-issue is for minutes *only*; comments that are not meeting minutes will be deleted.)

  **golang** locked and limited conversation to collaborators on Aug 7, 2019

  rsc changed the title **placeholder** proposal: review meeting minutes on Aug 7, 2019

  **gopherbot** added this to the [Proposal](#) milestone on Aug 7, 2019

  **gopherbot** added [Proposal](#) on Aug 7, 2019



rsc on Aug 7, 2019 · edited by rsc

Edits  Contributor Author ...

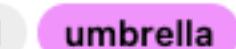
2019-08-06 / [@rsc](#), [@griesemer](#), [@andybons](#)

-  [cmd/fix: automate migrations for simple deprecations #32816](#) cmd/fix: automate migrations for simple deprecations
 - discussion ongoing (no action taken)
-  [proposal: cmd/go: add build tags for 32-bit and 64-bit architectures #33388](#) cmd/go: add build tags for 32-bit and 64-bit architectures
 - retitled, commented
-  [proposal: cmd/go: add package search functionality #33203](#) cmd/go: add package search functionality
 - commented

Assignees

No one assigned

Labels

Type

No type

Projects

No projects

Milestone

 **Proposal**

No due date

Relationships

None yet

Development

 Open in Workspace

No branches or pull requests

Notifications

Customize

 Unsubscribe

You're receiving notifications because you're subscribed to this thread.

Participants



- **testing/synctest**: new package for testing concurrent code [#67434](#)
 - hold for experience



rsc 2 days ago

Contributor Author ...

2025-02-12 / [@adonovan](#), [@andybons](#), [@bradfitz](#), [@cherrymui](#), [@ianlancetaylor](#), [@rolandshoemaker](#), [@rsc](#)

Discussions (not yet proposals)

- discussion: spec: reduce error handling boilerplate using ? [#71460](#)
- encoding/json/v2 [#63397](#)
- codereview: should the Go project stop importing GitHub PRs? [#61182](#)
- memory regions [#70257](#)
- gabyhelp feedback [#67901](#)
- metrics for the standard library [#61638](#)
- x/oscar feedback [#68490](#)
- net/http: move HTTP/2 into the standard library [#60746](#)
- gonew: templates for new modules [#61669](#)

Accepted

- crypto/rsa: allow hash.Hash for OAEP and MGF1 to be specified independently for encryption [#65716](#)
 - no change in consensus; accepted

Declined

- none

Likely Accept

- go/parser: deprecate parser.ParseDir (which returns deprecated type ast.Package) [#71122](#)
 - likely accept; last call for comments
- spec: remove notion of core types [#70128](#)
 - likely accept; last call for comments

Likely Decline

- none

Active

- all: remove support for windows/arm [#71671](#)
 - added to minutes
- cmd/go: add `work` package pattern matching all packages in work modules [#71294](#)
 - commented
- cmd/go: add go mod verify -tag [#68669](#)

LIKE AND SUBSCRIBE

ISSUE #33502



SNEAK PEEK: GOTIP



```
~ > go install golang.org/dl/gotip@latest
~ > gotip download
Updating the go development tree...
From https://go.googlesource.com/go
 * branch           master    -> FETCH_HEAD
HEAD is now at c62c69dd crypto/internal/fips140/subtle: provide riscv64 assembly implementation for xorBytes
Building Go cmd/dist using /nix/store/bva4mymi5b2lhvvbccx8ipws89rar1d8-go-1.23.4/share/go. (go1.23.4 darwin/arm64)
Building Go toolchain1 using /nix/store/bva4mymi5b2lhvvbccx8ipws89rar1d8-go-1.23.4/share/go.
Building Go bootstrap cmd/go (go_bootstrap) using Go toolchain1.
Building Go toolchain2 using go_bootstrap and Go toolchain1.
Building Go toolchain3 using go_bootstrap and Go toolchain2.
Building packages and commands for darwin/arm64.
---
Installed Go for darwin/arm64 in /Users/akupila/sdk/gotip
Installed commands in /Users/akupila/sdk/gotip/bin
Success. You may now run 'gotip'!
~ > go version
go version go1.23.4 darwin/arm64
~ > gotip version
go version devel go1.25-c62c69dd Fri Feb 14 20:58:15 2025 -0800 darwin/arm64
~ >
```

10 THINGS IN 1.24



GO TOOL

● ● ● nvim

```
19 package user
18
17 import "fmt"
16
15 type State int
14
13 const (
12     StateSignedUp State = iota
11     StateEmailVerified
10     StateBanned
9     StateDeactivated
8 )
7
6 type User struct {
5     Email string
4     State State
3 }
2
1 func (u User) String() string {
20 |     return fmt.Sprintf("%s (%v)", u.Email, u.State) // alice@example.com (0) ←
1 }
```

● ● ●

nvim

```
4 package user
3
2 import "fmt"
1
5 //go:generate stringer -type=State -trimprefix=State ←
1
2 type State int
3
4 const (
5     StateSignedUp State = iota
6     StateEmailVerified
7     StateBanned
8     StateDeactivated
9 )
10
11 type User struct {
12     Email string
13     State State
14 }
15
16 func (u User) String() string {
17     return fmt.Sprintf("%s (%v)", u.Email, u.State)
18 }
```

z z z z z z z z z z

5,54 All



.../dev/go124talk/tool

```
~/dev/go124talk/tool > go generate
user.go:5: running "stringer": exec: "stringer": executable file not found in $PATH
~/dev/go124talk/tool > go install golang.org/x/tools/cmd/stringer@latest
go: downloading golang.org/x/tools v0.30.0
go: downloading golang.org/x/sync v0.11.0
go: downloading golang.org/x/mod v0.23.0
~/dev/go124talk/tool > go generate
~/dev/go124talk/tool > █
```

```
1 // Code generated by "stringer -type=State -trimprefix=State"; DO NOT EDIT.
1
2 package user
3
4 import "strconv"
5
6 func _() {
7     // An "invalid array index" compiler error signifies that the constant values have changed.
8     // Re-run the stringer command to generate them again.
9     var x [1]struct{}
10    _ = x[StateSignedUp-0]
11    _ = x[StateEmailVerified-1]
12    _ = x[StateBanned-2]
13    _ = x[StateDeactivated-3]
14 }
15
16 const _State_name = "SignedUpEmailVerifiedBannedDeactivated"
17
18 var _State_index = [...]uint8{0, 8, 21, 27, 38}
19
20 func (i State) String() string {
21     if i < 0 || i >= State(len(_State_index)-1) {
22         return "State(" + strconv.FormatInt(int64(i), 10) + ")"
23     }
24     return _State_name[_State_index[i]:_State_index[i+1]]
25 }
```

```
z z z z z z
```



nvim

```
21 package user
20
19 import "fmt"
18
17 //go:generate stringer -type=State -trimprefix=State
16
15 type State int
14
13 const (
12     StateSignedUp State = iota
11     StateEmailVerified
10     StateBanned
9     StateDeactivated
8 )
7
6 type User struct {
5     Email string
4     State State
3 }
2
1 func (u User) String() string {
22     return fmt.Sprintf("%s (%v)", u.Email, u.State) // alice@example.com (EmailVerified) ←
1 }
```

```
z z z z z z z z z
```

22,87-90

All



.../dev/go124talk/tool

```
~/dev/go124talk/tool > go get -tool golang.org/x/tools/cmd/stringer@latest ←
go: added golang.org/x/mod v0.23.0
go: added golang.org/x/sync v0.11.0
go: added golang.org/x/tools v0.30.0
~/dev/go124talk/tool > cat go.mod
module github.com/akupila/go124

go 1.24

tool golang.org/x/tools/cmd/stringer

require (
    golang.org/x/mod v0.23.0 // indirect
    golang.org/x/sync v0.11.0 // indirect
    golang.org/x/tools v0.30.0 // indirect
)
~/dev/go124talk/tool > █
```



```
4 package user
3
2 import "fmt"
1
5 //go:generate go tool stringer -type=State -trimprefix=State ←
1
2 type State int
3
4 const (
5     StateSignedUp State = iota
6     StateEmailVerified
7     StateBanned
8     StateDeactivated
9 )
10
11 type User struct {
12     Email string
13     State State
14 }
15
16 func (u User) String() string {
17     return fmt.Sprintf("%s (%v)", u.Email, u.State) // alice@example.com (EmailVerified)
18 }
```

z z z z z z z z z

● ● ●

.../dev/go124talk/tool

*

```
~/dev/go124talk/tool > go tool
addr2line
asm
buildid
cgo
compile
covdata
cover
doc
fix
link
nm
objdump
pack
pprof
preprofile
test2json
trace
vet
golang.org/x/tools/cmd/stringer
~/dev/go124talk/tool > █
```

*

BINARY CACHING

● ● ●

~/dev/example

~/dev/example main > go run -x ./cmd/api



go run -x ./cmd/api

```
packagefile golang.org/x/text/internal=/Users/akupila/Library/Caches/go-build/6a/6a35a1829dc23f7255ebf2db9f668b5b  
76074bf7213b52e1aeb1a5eacea0a48f-d  
packagefile golang.org/x/text/internal/language=/Users/akupila/Library/Caches/go-build/56/56c9ad65171c7dc86f86616  
c529789604c1acc1804e5c565cd9000ee9048d898-d  
packagefile golang.org/x/text/internal/language/compact=/Users/akupila/Library/Caches/go-build/e6/e652d9771c3c491  
349dcb3485f69d3133516812b9433dad75f994ba69bc78d64-d  
packagefile golang.org/x/text/unicode/bidi=/Users/akupila/Library/Caches/go-build/37/377706f9a3b8c1c8f0ff26df981f  
5b11e8a2619d74a035ca73783d45308fdb1d-d  
packagefile crypto/internal/nistec/fiat=/Users/akupila/Library/Caches/go-build/6b/6b5c37413a68a0e40c91a4f62b94697  
a97bdbda85d461f5e6333d93350be45f5-d  
packagefile embed=/Users/akupila/Library/Caches/go-build/f2/f2d81a160d9e12d3df85de40032f4bd90cb439b5c399694590a4d  
8b81e83e0d6-d  
packagefile golang.org/x/text/internal/tag=/Users/akupila/Library/Caches/go-build/d0/d091bf4b7d6d3561c5bac8a31400  
47c9755b3f79cc55a1a4d763fd39d7651952-d  
modinfo "0w\xaf\f\x92t\b\x02A\xe1\xc1\ a\xe6\xd6\x18\xe6path\tgithub.com/func/func/cmd/api\nmod\tgithub.com/func/f  
unc\t(devel)\t\ndep\tgithub.com/google/uuid\tv1.6.0\th1:NIvaJDM0sjHA8n1jAhLSgrAzy1Hgr+hNrb57e+94F0=\n\ndep\tgithub  
.com/jackc/pgpassfile\tv1.0.0\th1:/6Hmqy13Ss2zCq62VdNG8tM1wchn8zjSG0BJ6icpsIM=\n\ndep\tgithub.com/jackc/pgservicefi  
le\tv0.0.0-20240606120523-5a60cdf6a761\th1:iCEnooe7Ulw0QYpKFhBabPMi4aNfoODPEFNiAnClxo=\n\ndep\tgithub.com/jackc/pg  
x/v5\tv5.7.1\th1:x7SYsPBYDkHDksogeSmZZ5xzThcTgRz++I5E+ePFUcs=\n\ndep\tgithub.com/jackc/puddle/v2\tv2.2.2\th1:PR8nw+  
E/1w0GLuRFSmioY6UooMp6KJv0/61nB7icHo=\n\ndep\tgithub.com/lmittmann/tint\tv1.0.5\th1:NQclAut0fYsqz2F1Lenue6OoWCajs5  
wJcP3DfWVpePw=\n\ndep\tgithub.com/matttn/go-isatty\tv0.0.20\th1:xfD0iDuEKnDkl03q4limB+vH+GxLEtL/jb4xVJSWWEY=\n\ndep\tg  
ithub.com/peterbourgon/ff/v4\tv4.0.0-alpha.4\th1:aiqS8aBlF9PsAKeMddMSfbwp3sm0NCn3U08QfUg0Z7Y=\n\ndep\tgolang.org/x/  
crypto\tv0.27.0\th1:GXm2NjJrPaiv/h1tb2UH8QfgC/h0f/+z0p6PT8o1w7A=\n\ndep\tgolang.org/x/sync\tv0.8.0\th1:3NFvSEYkUoMi  
fnESzZl15y791HH1qU2xm6eCJU5ZPXQ=\n\ndep\tgolang.org/x/sys\tv0.25.0\th1:r+8e+loiHxRqhXVl6ML1n03l1+oFoWbnlu2Ehimmi34=  
\n\ndep\tgolang.org/x/text\tv0.18.0\th1:XvMDiNzPAI0jr17s6W9lcaIhGUfU0RdGCNsuLmPG224=\n\ncbuild\t-buildmode=exe\ncbuild\t  
-compiler=gc\ncbuild\tCGO_ENABLED=1\ncbuild\tCGO_CFLAGS=\ncbuild\tCGO_CPPFLAGS=\ncbuild\tCGO_CXXFLAGS=\ncbuild\tCGO_L  
DFLAGS=\ncbuild\tGOARCH=arm64\ncbuild\tGOOS=darwin\ncbuild\tGOARM64=v8.0\n\xf92C1\x86\x18 r\x00\x82B\x10A\x16\xd8\xf  
2"  
EOF  
mkdir -p $WORK/b001/exe/  
cd .  
GOROOT='/nix/store/bva4mymi5b2lhvvbccx8ipws89rar1d8-go-1.23.4/share/go' /nix/store/bva4mymi5b2lhvvbccx8ipws89rar1  
d8-go-1.23.4/share/go/pkg/tool/darwin_arm64/link -o $WORK/b001/exe/api -importcfg $WORK/b001/importcfg.link -s -w  
-buildmode=pie -buildid=6sKHNknLNYSLi0KxBtW1/shH01PhVqfw_oW46miPu/exi2a9eClQInCoGcXK2x/6sKHNknLNYSLi0KxBtW1 -ext  
ld=clang /Users/akupila/Library/Caches/go-build/da/da78f351f7f5510864a391f7e11b6fe44b7f3013682c053bb70ceda255b15b  
9c-d  
$WORK/b001/exe/api  
15:46:37.276 INF Ready to accept traffic addr=127.0.0.1:8080
```



gotip run -x ./cmd/api

```
~/dev/example main > gotip run -x ./cmd/api
WORK=/var/folders/xg/4nzhzzm568v1_47836d97c200000gn/T/go-build1166156704
/Users/akupila/Library/Caches/go-build/7a/7a8100d32895eb84b5b990f0ea54a33e4dcc65ff7bcf32da997d33cfa8ba880a-d/api
15:46:57.115 INF Ready to accept traffic addr=127.0.0.1:8080
```

CRYPTO PACKAGES

Discover Packages > Standard library > crypto > mlkem

mlkem

Version: go1.24.0

Details Repository Links

Jump to ...

Documentation

- Overview
- Index
- Constants
- Variables
- Functions
- Types

Source Files

Search packages or symbols

Why Go Learn Docs Packages Community

Discover Packages > Standard library > crypto > hkdf

hkdf

Version: go1.24.0

Details Repository Links

Jump to ...

Documentation

- Overview
- Index
- Constants
- Variables
- Functions
- Types

Source Files

Search packages or symbols

Why Go Learn Docs Packages Community

Discover Packages > Standard library > crypto > pbkdf2

pbkdf2

Version: go1.24.0

Details Repository Links

Jump to ...

Documentation

- Overview
- Index
- Constants
- Variables
- Functions
- Types

Source Files

Search packages or symbols

Why Go Learn Docs Packages Community

Discover Packages > Standard library > crypto > sha3

sha3

Version: go1.24.0 Latest | Published: Feb 11, 2025 | License: BSD-3-Clause | Imports: 4 | Imported by: 11

Details Repository Links

Valid go.mod file | Redistributable license | Tagged version | Stable version | Learn more about best practices

cs.opensource.google/go/go

Report a Vulnerability

Search packages or symbols

Why Go Learn Docs Packages Community

Discover Packages > Standard library > crypto > mlkem

mlkem

Version: go1.24.0

Details Repository Links

Jump to ...

Documentation

- Overview
- Index
- Constants
- Variables

Source Files

Search packages or symbols

Why Go Learn Docs Packages Community

OS.OPENROOT

OMITZERO

json package - encoding/json - X + https://pkg.go.dev/encoding/json

Version: go1.24.0 Latest | Published: Feb 11, 2025 | License: BSD-3-Clause | Imports: 17 | Imported by: 1,301,844

Jump to ... f

Documentation

- Overview
- Index
- Constants
- Variables
- Functions
- Types

Source Files

```
// the field is omitted from the object if its value is empty,
// as defined above.
Field int `json:"myName,omitempty"`

// Field appears in JSON as key "Field" (the default), but
// the field is skipped if empty.
// Note the leading comma.
Field int `json:",omitempty"`

// Field is ignored by this package.
Field int `json:"-"`

// Field appears in JSON as key "-".
Field int `json:"-,"`
```

The "omitempty" option specifies that the field should be omitted from the encoding if the field has a zero value, according to rules:

- 1) If the field type has an "IsZero() bool" method, that will be used to determine whether the value is zero.
- 2) Otherwise, the value is zero if it is the zero value for its type.

If both "omitempty" and "omitempty" are specified, the field will be omitted if the value is either empty or zero (or both).

The "string" option signals that a field is stored as JSON inside a JSON-encoded string. It applies only to fields of string, floating point, integer, or boolean types. This extra level of encoding is sometimes used when communicating with JavaScript programs:

```
Int64String int64 `json:",string"`
```

The key name will be used if it's a non-empty string consisting of only Unicode letters, digits, and ASCII punctuation except quotation marks, backslash, and comma.

Embedded struct fields are usually marshaled as if their inner exported fields were fields in the outer struct, subject to the usual Go visibility rules amended as described in the next paragraph. An anonymous struct field with a name given in its JSON tag is treated as having that name, rather than being anonymous. An anonymous struct field of interface type is treated the same as having that type as its name, rather than being anonymous.

The Go visibility rules for struct fields are amended for JSON when deciding which field to marshal or unmarshal. If there are multiple fields at the same level, and that level is the least nested (and would therefore be the nesting level selected by the usual Go

```
● ● ● nvim

10 package main
11
12 import (
13     "encoding/json"
14     "fmt"
15     "time"
16 )
17
18 type User struct {
19     Email      string      `json:"email"`
20     Verified   time.Time   `json:"verified,omitempty"` ←
21 }
22
23 func main() {
24     u := User{Email: "alice@example.com"}
25
26     j, _ := json.MarshalIndent(u, "", "  ")
27     fmt.Println(string(j))
28 }
```

● ● ●

.../dev/go124talk/omitzero

```
~/dev/go124talk/omitzero > go run .
{
  "email": "alice@example.com",
  "verified": "0001-01-01T00:00:00Z"
}
~/dev/go124talk/omitzero > █
```

```
10 package main
 9
 8 import (
 7     "encoding/json"
 6     "fmt"
 5     "time"
 4 )
 3
 2 type User struct {
 1     Email      string      `json:"email"`
 11    Verified time.Time `json:"verified,omitempty"` ←
 1 }
 2
 3 func main() {
 4     u := User{Email: "alice@example.com"}
 5
 6     j, _ := json.MarshalIndent(u, "", "  ")
 7     fmt.Println(string(j))
 8 }
```

11,48-51

All



.../dev/go124talk/omitzero

```
~/dev/go124talk/omitzero > go run .
{
  "email": "alice@example.com",
  "verified": "0001-01-01T00:00:00Z"
}
~/dev/go124talk/omitzero > go run .
{
  "email": "alice@example.com"
}
~/dev/go124talk/omitzero > █
```

T.CONTEXT()

testing package - testing - Go X + https://pkg.go.dev/testing#T.Context Version: go1.24.0 Latest | Published: Feb 11, 2025 | License: BSD-3-Clause | Imports: 26 | Imported by: 96,119

Jump to ... f Documentation

Overview Index Constants Variables Functions Types type B type BenchmarkResult type Cover type CoverBlock type F type InternalBenchmark type InternalExample type InternalFuzzTarget type InternalTest type M type PB type T (t) Chdir(dir) (c) Cleanup(f) (c) Context()

func (*T) Context

```
func (c *T) Context() context.Context
```

Context returns a context that is canceled just before Cleanup-registered functions are called.

Cleanup functions can wait for any resources that shut down on Context.Done before the test or benchmark completes.

func (*T) Deadline

```
func (t *T) Deadline() (deadline time.Time, ok bool)
```

Deadline reports the time at which the test binary will have exceeded the timeout specified by the -timeout flag.

The ok result is false if the -timeout flag indicates "no timeout" (0).

func (*T) Error

```
func (c *T) Error(args ...any)
```

Error is equivalent to Log followed by Fail.

func (*T) Errorf

```
func (c *T) Errorf(format string, args ...any)
```

Errorf is equivalent to Logf followed by Fail.

func (*T) Fail

```
func (c *T) Fail()
```

Fail marks the function as having failed but continues execution.

func (*T) FailNow

BLOOP()

● ● ●

```
11 package fibonacci
10
9 func Fibonacci(n int) int {
8     if n < 2 {
7         return n
6     }
5     a, b := 0, 1
4     for i := 2; i <= n; i++ {
3         a, b = b, a+b
2     }
1     return b
12 }
```

nvim

```
5 package fibonacci
4
3 import "testing"
2
1 func BenchmarkFibonacci(b *testing.B) {
6     for range b.N {
1         Fibonacci(32)
2     }
3 }
```

6,1

All

```
~/dev/go124talk/fibonacci > go test -bench=. -gcflags="-m"
# talk/fibonacci [talk/fibonacci.test]
./fibonacci.go:3:6: can inline Fibonacci
./fibonacci_test.go:5:6: can inline BenchmarkFibonacci
./fibonacci_test.go:7:12: inlining call to Fibonacci
./fibonacci_test.go:5:25: b does not escape
# talk/fibonacci.test
_testmain.go:37:6: can inline init.0
<autogenerated>:1: inlining call to reflect.flag.kind
<autogenerated>:1: inlining call to reflect.flag.kind
<autogenerated>:1: inlining call to reflect.flag.mustBe
<autogenerated>:1: inlining call to reflect.flag.kind
<autogenerated>:1: inlining call to reflect.flag.mustBe
<autogenerated>:1: inlining call to reflect.flag.kind
<autogenerated>:1: inlining call to reflect.flag.mustBeAssignable
<autogenerated>:1: inlining call to reflect.flag.mustBeAssignable
<autogenerated>:1: inlining call to reflect.flag.mustBeExported
<autogenerated>:1: inlining call to reflect.flag.mustBeExported
<autogenerated>:1: inlining call to reflect.flag.ro
<autogenerated>:1: inlining call to reflect.flag.ro
_testmain.go:43:42: testdeps.TestDeps{} escapes to heap
<autogenerated>:1: &reflect.ValueError{...} escapes to heap
<autogenerated>:1: &reflect.ValueError{...} escapes to heap
goos: darwin
goarch: arm64
pkg: talk/fibonacci
cpu: Apple M2 Pro
```

nvim

```
11 package fibonacci
10
9 func Fibonacci(n int) int {
8     if n < 2 {
7         return n
6     }
5     a, b := 0, 1
4     for i := 2; i <= n; i++ {
3         a, b = b, a+b
2     }
1     return b
12 }
```

```
5 package fibonacci
4
3 import "testing"
2
1 func BenchmarkFibonacci(b *testing.B) {
6     for b.Loop() {
1         Fibonacci(32)
2     }
3 }
```

6,1 All

```
~/dev/go124talk/fibonacci > go test -bench=. -gcflags="-m"
# talk/fibonacci [talk/fibonacci.test]
./fibonacci.go:3:6: can inline Fibonacci
./fibonacci_test.go:6:12: inlining call to testing.(*B).Loop
./fibonacci_test.go:5:25: b does not escape
# talk/fibonacci.test
_testmain.go:37:6: can inline init.0
<autogenerated>:1: inlining call to reflect.flag.kind
<autogenerated>:1: inlining call to reflect.flag.kind
<autogenerated>:1: inlining call to reflect.flag.mustBe
<autogenerated>:1: inlining call to reflect.flag.kind
<autogenerated>:1: inlining call to reflect.flag.mustBe
<autogenerated>:1: inlining call to reflect.flag.kind
<autogenerated>:1: inlining call to reflect.flag.mustBeAssignable
<autogenerated>:1: inlining call to reflect.flag.mustBeAssignable
<autogenerated>:1: inlining call to reflect.flag.mustBeExported
<autogenerated>:1: inlining call to reflect.flag.mustBeExported
<autogenerated>:1: inlining call to reflect.flag.ro
<autogenerated>:1: inlining call to reflect.flag.ro
_testmain.go:43:42: testdeps.TestDeps{} escapes to heap
<autogenerated>:1: &reflect.ValueError{...} escapes to heap
<autogenerated>:1: &reflect.ValueError{...} escapes to heap
goos: darwin
goarch: arm64
pkg: talk/fibonacci
cpu: Apple M2 Pro
BenchmarkFibonacci-12          73770682          16.22 ns/op
```

SYNCTEST



fg

```
21 package retry
20
19 import (
18     "context"
17     "time"
16 )
15
14 // Func executes the given function until it returns nil.
13 // If it returns an error, it will be called again with
12 // exponential backoff until it succeeds or the context
11 // is cancelled.
10 func Func(ctx context.Context, fn func() error) error {
9     sleep := 25 * time.Millisecond
8     for {
7         if err := fn(); err == nil {
6             return nil
5         }
4         select {
3         case <-ctx.Done():
2             return ctx.Err()
1         case <-time.After(sleep):
22             // Try again
1             sleep *= 2
2             continue
3         }
4     }
5 }
```

```
24 package retry
23
22 import (
21     "fmt"
20     "testing"
19 )
18
17 func TestFunc(t *testing.T) {
16     var calls int
15     fn := func() error {
14         calls++
13         if calls >= 10 {
12             return nil
11         }
10         return fmt.Errorf("error")
9     }
8
7     err := Func(t.Context(), fn)
6     if err != nil {
5         t.Fatal(err)
4     }
3     if calls != 10 {
2         t.Fatalf("calls=%d", calls)
1     }
25     t.Logf("Called %d times", calls)
1 }
```

```
● ● ● .../dev/go124talk/retry
~/dev/go124talk/retry > go test -v
=== RUN TestFunc
    retry_test.go:25: Called 10 times
--- PASS: TestFunc (12.78s) ←
PASS
ok      talk/retry    13.013s
~/dev/go124talk/retry > █
```



fg

```
21 package retry
20
19 import (
18     "context"
17     "time"
16 )
15
14 // Func executes the given function until it returns nil.
13 // If it returns an error, it will be called again with
12 // exponential backoff until it succeeds or the context
11 // is cancelled.
10 func Func(ctx context.Context, fn func() error) error {
9     sleep := 25 * time.Millisecond
8     for {
7         if err := fn(); err == nil {
6             return nil
5         }
4         select {
3         case <-ctx.Done():
2             return ctx.Err()
1         case <-time.After(sleep):
22             // Try again
1             sleep *= 2
2             continue
3         }
4     }
5 }
```

```
18 package retry
17
16 import (
15     "fmt"
14     "testing"
E 13     "testing/synctest"
12 )
11
10 func TestFunc(t *testing.T) {
9     var calls int
8     fn := func() error {
7         calls++
6         if calls >= 10 {
5             return nil
4         }
3         return fmt.Errorf("error")
2     }
1
19 synctest.Run(func() { ←
1     err := Func(t.Context(), fn)
2     if err != nil {
3         t.Fatal(err)
4     }
5     if calls != 10 {
6         t.Fatalf("calls=%d", calls)
7     }
8     t.Logf("Called %d times", calls)
9 })
10 }
```

```
● ● ● .../dev/go124talk/retry
~/dev/go124talk/retry > go test -v
# talk/retry
package talk/retry (test)
    imports testing/synctest: build constraints exclude all Go files in /Users/akupila/go/pkg/mod/golang.org/toolchain
@v0.0.1-go1.24.0.darwin-arm64/src/testing/synctest
FAIL    talk/retry [setup failed]
~/dev/go124talk/retry > GOEXPERIMENT=synctest go test -v
*** RUN  TestFunc
    retry_test.go:27: Called 10 times
--- PASS: TestFunc (0.00s) ←
PASS
ok      talk/retry      0.231s
~/dev/go124talk/retry > █
```

ENCODING.APPENDBINARY

encoding package - encoding - + https://pkg.go.dev/encoding#BinaryAppender Version: go1.24.0 Latest | Published: Feb 11, 2025 | License: BSD-3-Clause | Imports: 0 | Imported by: 15,769

Jump to ... f

Documentation

- Overview
- Index
- Constants
- Variables
- Functions
- Types
 - type [BinaryAppender](#)
 - type [BinaryMarshaler](#)
 - type [BinaryUnmarshaler](#)
 - type [TextAppender](#)
 - type [TextMarshaler](#)
 - type [TextUnmarshaler](#)
- Source Files
- Directories

type BinaryAppender added in go1.24.0

```
type BinaryAppender interface {
    // AppendBinary appends the binary representation of itself to the end of b
    // (allocating a larger slice if necessary) and returns the updated slice.
    //
    // Implementations must not retain b, nor mutate any bytes within b[:len(b)].
    AppendBinary(b []byte) ([]byte, error)
}
```

BinaryAppender is the interface implemented by an object that can append the binary representation of itself. If a type implements both [BinaryAppender](#) and [BinaryMarshaler](#), then v.MarshalBinary() must be semantically identical to v.AppendBinary(nil).

type BinaryMarshaler

```
type BinaryMarshaler interface {
    MarshalBinary() (data []byte, err error)
}
```

BinaryMarshaler is the interface implemented by an object that can marshal itself into a binary form.

MarshalBinary encodes the receiver into a binary form and returns the result.

type BinaryUnmarshaler

```
type BinaryUnmarshaler interface {
    UnmarshalBinary(data []byte) error
}
```

BinaryUnmarshaler is the interface implemented by an object that can unmarshal a binary representation of itself.

UnmarshalBinary must be able to decode the form generated by MarshalBinary. UnmarshalBinary must copy the data if it wishes to retain the data after returning.

type TextAppender added in go1.24.0

```
type TextAppender interface {
```

WEAK POINTERS

weak package - weak - Go Pack X + https://pkg.go.dev/weak#Make Version: go1.24.0 Latest | Published: Feb 11, 2025 | License: BSD-3-Clause | Imports: 3 | Imported by: 5

weak package standard library

Jump to ... f

Documentation

Overview

Package weak provides ways to safely reference memory weakly, that is, without preventing its reclamation.

Index

type Pointer

```
func Make[T any](ptr *T) Pointer[T]
func (p Pointer[T]) Value() *T
```

Constants

This section is empty.

Variables

This section is empty.

Functions

This section is empty.

Types

type Pointer

```
type Pointer[T any] struct {
    // contains filtered or unexported fields
}
```

Pointer is a weak pointer to a value of type T.

Just like regular pointers, Pointer may reference any part of an object, such as a field of a struct or an element of an array. Objects that are only pointed to by weak pointers are not considered reachable, and once the object becomes unreachable, [Pointer.Value](#).

PHEW, THAT'S ENOUGH FOR NOW.

THANKS!



BUT READ THE RELEASE NOTES FOR MORE:

