学习心得

文档（Go Doc）

分享人：Roy

传统艺能（建议）：

- 分享过程中可随时打断提问，但是否回答由分享人斟酌，或在Q&A环节回答

Agenda

- 文档的意义
- 如何生成文档
  - 文档服务器
  - 命令行文档
- 如何阅读文档
  - 官方文档
  - 第三方文档

文档的意义

意义1：解决问题

文档的意义

解决开源软件问题的思路(优先级排序)

1. 阅读官方文档(最权威) 
2. 社区提问(流行的社区)
3. 谷歌搜索(排名靠前)
4. 自己假设验证(test cases)
5. 阅读源代码(问题驱动)

文档的意义

意义2：编程哲学与艺术

文档的意义

1. 模仿包、函数、结构体的设计
2. 模仿编码规范

文档的意义

意义3：系统学习

文档的意义

1. 学习语言核心的常用包用法，比如 encoding/json、strings 等
2. 学习第三方包、框架的使用
3. 有一个大概印象，当需要实现某个功能时，可以再次查阅文档

文档服务器(官方文档)

golang.org/pkg/

Black Lives Matter. Support the Equal Justice Initiative.

Documents Packages The Project Help Blog Play

Packages

Standard library
Other packages
Sub-repositories
Community

Standard library

Name	Synopsis
archive	
tar	Package tar implements access to tar archives.
zip	Package zip provides support for reading and writing ZIP archives.
bufio	Package bufio implements buffered I/O. It wraps an io.Reader or io.Writer object, creating another object (Reader or Writer) that also implements the interface but provides buffering and some help for textual I/O.
builtin	Package builtin provides documentation for Go's predeclared identifiers.
bytes	Package bytes implements functions for the manipulation of byte slices.
cmd	
addr2line	Addr2line is a minimal simulation of the GNU addr2line tool, just enough to support pprof.
api	Binary api computes the exported API of a set of Go packages.
asm	Asm, typically invoked as “go tool asm”, assembles the source file into an object file named for the basename of the argument source file with a .o suffix.
buildid	Buildid displays or updates the build ID stored in a Go package or binary.
cgo	Cgo enables the creation of Go packages that call C code.
compile	Compile, typically invoked as “go tool compile,” compiles a single Go package comprising the files named on the command line.
cover	Cover is a program for analyzing the coverage profiles generated by 'go test -coverprofile=cover.out'.

文档服务器(本地文档)

安装 tools 包

文档服务器(本地文档)

文档服务器(本地文档)

127.0.0.1:6060/pkg/

Go Documentation Server

Search

Packages

Standard library
Other packages
Sub-repositories
Community

Standard library

Name	Synopsis
archive	
tar	Package tar implements access to tar archives.
zip	Package zip provides support for reading and writing ZIP archives.
bufio	Package bufio implements buffered I/O. It wraps an io.Reader or io.Writer object, creating another object (Reader or Writer) that also implements the interface but provides buffering and some help for textual I/O.
builtin	Package builtin provides documentation for Go's predeclared identifiers.
bytes	Package bytes implements functions for the manipulation of byte slices.
compress	
bzip2	Package bzip2 implements bzip2 decompression.
flate	Package flate implements the DEFLATE compressed data format, described in RFC 1951.
gzip	Package gzip implements reading and writing of gzip format compressed files, as specified in RFC 1952.
lzw	Package lzw implements the Lempel-Ziv-Welch compressed data format, described in T. A. Welch, “A Technique for High-Performance Data Compression”, Computer, 17(6) (June 1984), pp 8-19.
zlib	Package zlib implements reading and writing of zlib format compressed data, as specified in RFC 1950.
container	
heap	Package heap provides heap operations for any type that implements heap.Interface.
list	Package list implements a doubly linked list.
ring	Package ring implements operations on circular lists.
context	Package context defines the Context type, which carries deadlines, cancellation signals, and other request-scoped values across API boundaries and between processes.

命令行文档

作用：

1. 类似于很多 linux 命令的 help 参数
2. 查看完整的文档比较麻烦
3. 临时查看某个包、函数、结构体、方法的用法
4. 查看当前包、函数、结构体、方法

命令行文档

生成 strings 包文档

命令行文档

```
luoxiaojundeMacBook-Pro:go luoxiaojun$ go doc strings
package strings // import "strings"

Package strings implements simple functions to manipulate UTF-8 encoded
strings.

For information about UTF-8 strings in Go, see
https://blog.golang.org/strings.

func Compare(a, b string) int
func Contains(s, substr string) bool
func ContainsAny(s, chars string) bool
func ContainsRune(s string, r rune) bool
func Count(s, substr string) int
func EqualFold(s, t string) bool
func Fields(s string) []string
func FieldsFunc(s string, f func(rune) bool) []string
func HasPrefix(s, prefix string) bool
func HasSuffix(s, suffix string) bool
func Index(s, substr string) int
func IndexAny(s, chars string) int
func IndexByte(s string, c byte) int
func IndexFunc(s string, f func(rune) bool) int
func IndexRune(s string, r rune) int
func Join(elems []string, sep string) string
func LastIndex(s, substr string) int
func LastIndexAny(s, chars string) int
func LastIndexByte(s string, c byte) int
func LastIndexFunc(s string, f func(rune) bool) int
func Map(mapping func(rune) rune, s string) string
func Repeat(s string, count int) string
func Replace(s, old, new string, n int) string
func ReplaceAll(s, old, new string) string
func Split(s, sep string) []string
func SplitAfter(s, sep string) []string
func SplitAfterN(s, sep string, n int) []string
func SplitN(s, sep string, n int) []string
func Title(s string) string
func ToLower(s string) string
func ToLowerSpecial(c unicode.SpecialCase, s string) string
func ToTitle(s string) string
func ToTitleSpecial(c unicode.SpecialCase, s string) string
func ToUpper(s string) string
func ToUpperSpecial(c unicode.SpecialCase, s string) string
func ToValidUTF8(s, replacement string) string
func Trim(s, cutset string) string
```

命令行文档

生成 strings 包的 Contains 函数文档

命令行文档

```
luoxiaojundeMacBook-Pro:go luoxiaojun$ go doc strings contains
package strings // import "strings"

func Contains(s, substr string) bool
    Contains reports whether substr is within s.
```

命令行文档

生成 strings 包的 Reader 结构体文档

命令行文档

```
luoxiaojundeMacBook-Pro:go luoxiaojun$ go doc strings reader
package strings // import "strings"

type Reader struct {
        // Has unexported fields.
}
    A Reader implements the io.Reader, io.ReaderAt, io.ByteReader,
    io.ByteScanner, io.RuneReader, io.RuneScanner, io.Seeker, and io.WriterTo
    interfaces by reading from a string. The zero value for Reader operates like
    a Reader of an empty string.

func NewReader(s string) *Reader
func (r *Reader) Len() int
func (r *Reader) Read(b []byte) (n int, err error)
func (r *Reader) ReadAt(b []byte, off int64) (n int, err error)
func (r *Reader) ReadByte() (byte, error)
func (r *Reader) ReadRune() (ch rune, size int, err error)
func (r *Reader) Reset(s string)
func (r *Reader) Seek(offset int64, whence int) (int64, error)
func (r *Reader) Size() int64
func (r *Reader) UnreadByte() error
func (r *Reader) UnreadRune() error
func (r *Reader) WriteTo(w io.Writer) (n int64, err error)
```

命令行文档

```
luoxiaojundeMacBook-Pro:models luoxiaojun$ go doc
package models // import "github.com/luoxiaojun1992/go-skeleton/models"

type BaseModel struct{}
type ModelInterface interface{ ... }
type PhabricatorBugTasks struct{ ... }
type PhabricatorBugTasksHistory struct{ ... }
type QueryBuilder struct{ ... }
```

生成当前包的文档

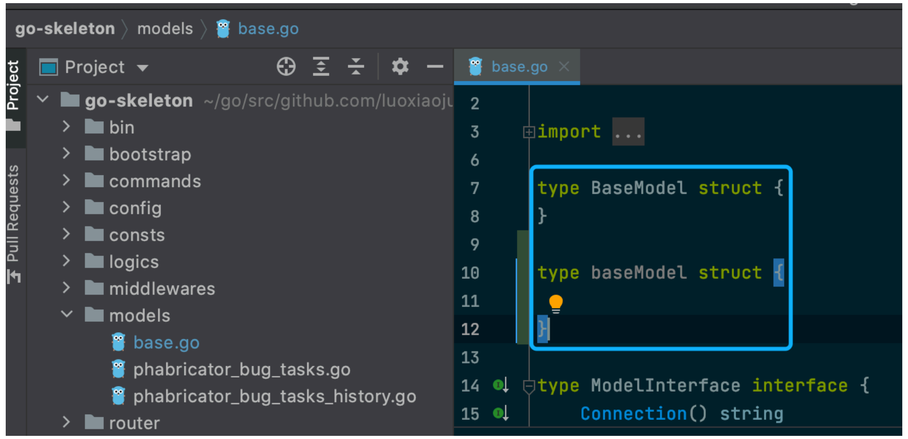
命令行文档

生成当前包的 BaseModel 结构体文档

命令行文档

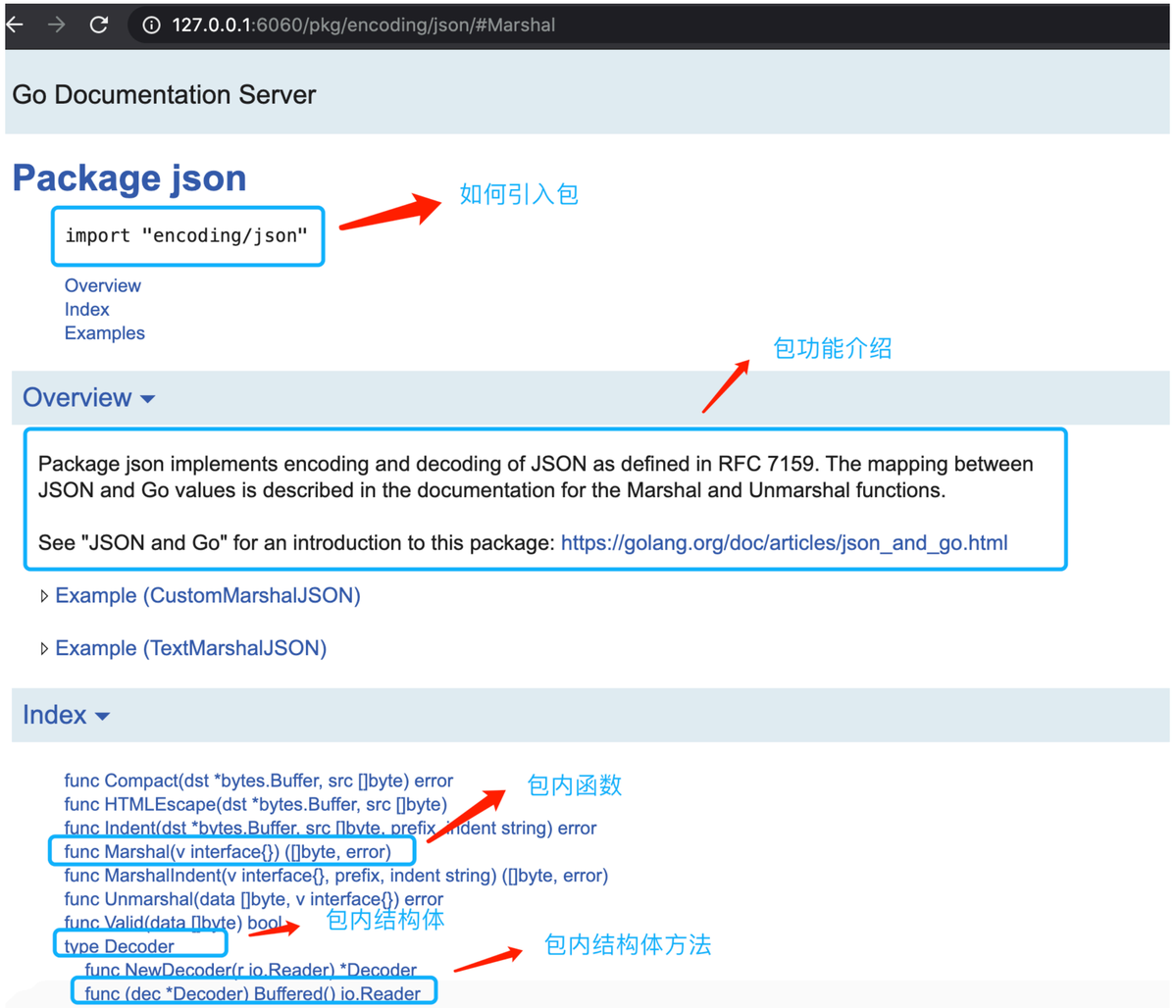
```
luoxiaojundeMacBook-Pro:models luoxiaojun$ go doc basemodel
package models // import "github.com/luoxiaojun1992/go-skeleton/models"

type BaseModel struct {
}

func (baseModel *BaseModel) Query(model ModelInterface) *QueryBuilder
```

命令行文档

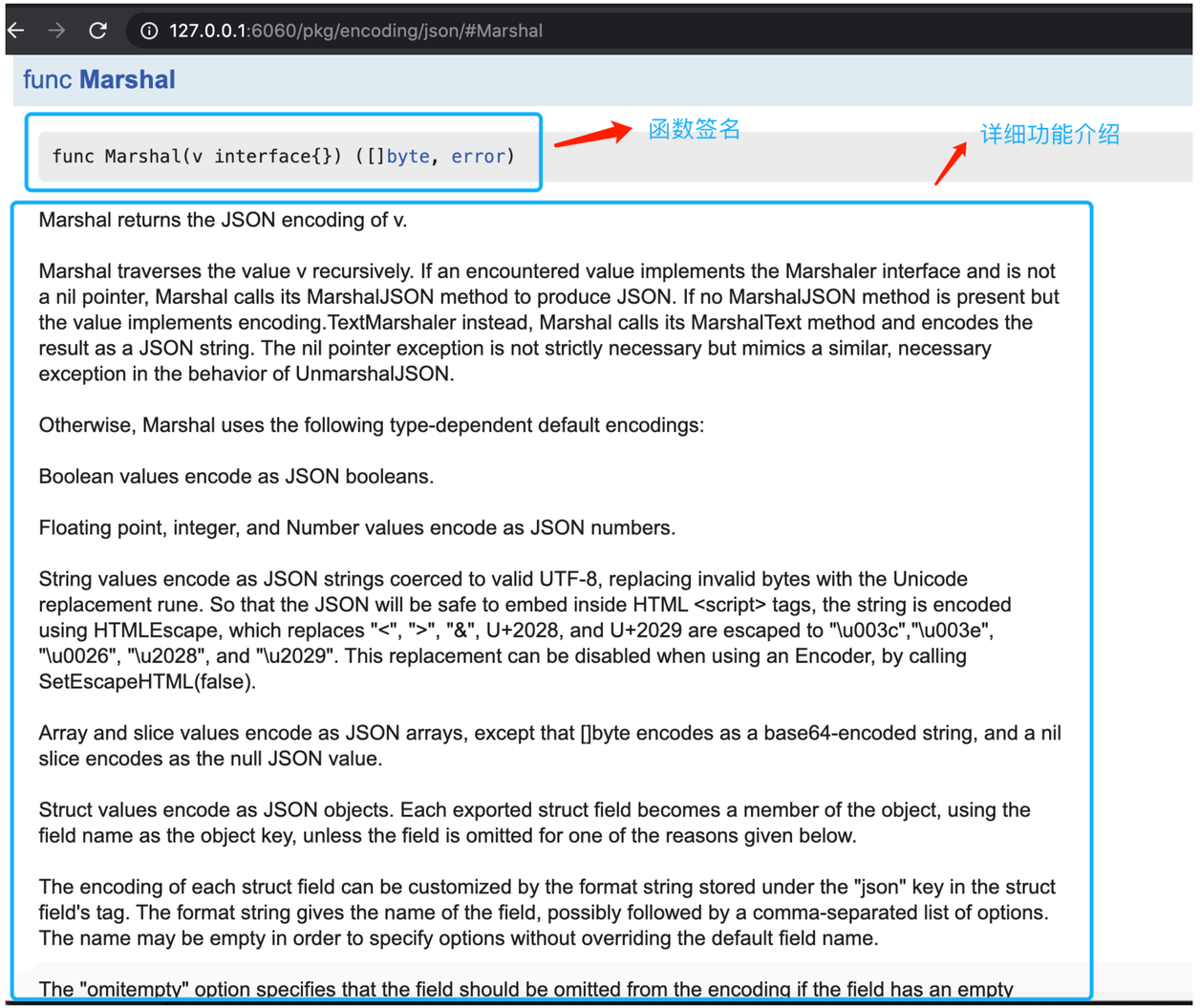


图中有两个结构体，区别是首字母大小写不同，生成首字母大写的结构体文档

官方文档阅读


127.0.0.1:6060/pkg/encoding/json/#Marshal
Go Documentation Server
Package json
import "encoding/json"
如何引入包
Overview
Index
Examples
包功能介绍
Overview
Package json implements encoding and decoding of JSON as defined in RFC 7159. The mapping between JSON and Go values is described in the documentation for the Marshal and Unmarshal functions.
See "JSON and Go" for an introduction to this package: https://golang.org/doc/articles/json_and_go.html
Example (CustomMarshalJSON)
Example (TextMarshalJSON)
Index
func Compact(dst *bytes.Buffer, src []byte) error
func HTMLEscape(dst *bytes.Buffer, src []byte)
func Indent(dst *bytes.Buffer, src []byte, prefix, indent string) error
func Marshal(v interface{}) ([]byte, error)
func MarshalIndent(v interface{}, prefix, indent string) ([]byte, error)
func Unmarshal(data []byte, v interface{}) error
func Valid(data []byte) bool
包内函数
type Decoder
包内结构体
func NewDecoder(r io.Reader) *Decoder
func (dec *Decoder) Buffered() io.Reader
包内结构体方法

官方文档阅读


127.0.0.1:6060/pkg/encoding/json/#Marshal
func Marshal
func Marshal(v interface{}) ([]byte, error)
函数签名
详细功能介绍
Marshal returns the JSON encoding of v.
Marshal traverses the value v recursively. If an encountered value implements the Marshaler interface and is not a nil pointer, Marshal calls its MarshalJSON method to produce JSON. If no MarshalJSON method is present but the value implements encoding.TextMarshaler instead, Marshal calls its MarshalText method and encodes the result as a JSON string. The nil pointer exception is not strictly necessary but mimics a similar, necessary exception in the behavior of UnmarshalJSON.
Otherwise, Marshal uses the following type-dependent default encodings:
Boolean values encode as JSON booleans.
Floating point, integer, and Number values encode as JSON numbers.
String values encode as JSON strings coerced to valid UTF-8, replacing invalid bytes with the Unicode replacement rune. So that the JSON will be safe to embed inside HTML <script> tags, the string is encoded using HTMLEscape, which replaces "<", ">", "&", U+2028, and U+2029 are escaped to "\u003c","\u003e", "\u0026", "\u2028", and "\u2029". This replacement can be disabled when using an Encoder, by calling SetEscapeHTML(false).
Array and slice values encode as JSON arrays, except that []byte encodes as a base64-encoded string, and a nil slice encodes as the null JSON value.
Struct values encode as JSON objects. Each exported struct field becomes a member of the object, using the field name as the object key, unless the field is omitted for one of the reasons given below.
The encoding of each struct field can be customized by the format string stored under the "json" key in the struct field's tag. The format string gives the name of the field, possibly followed by a comma-separated list of options. The name may be empty in order to specify options without overriding the default field name.
The "omitempty" option specifies that the field should be omitted from the encoding if the field has an empty

官方文档阅读

127.0.0.1:6060/pkg/encoding/json/#Marshal

Pointer values encode as the value pointed to. A nil pointer encodes as the null JSON value.

Interface values encode as the value contained in the interface. A nil interface value encodes as the null JSON value.

Channel, complex, and function values cannot be encoded in JSON. Attempting to encode such a value causes Marshal to return an UnsupportedTypeError.

代码示例

JSON cannot represent cyclic data structures and Marshal does not handle them. Passing cyclic structures to Marshal will result in an error.

▾ Example

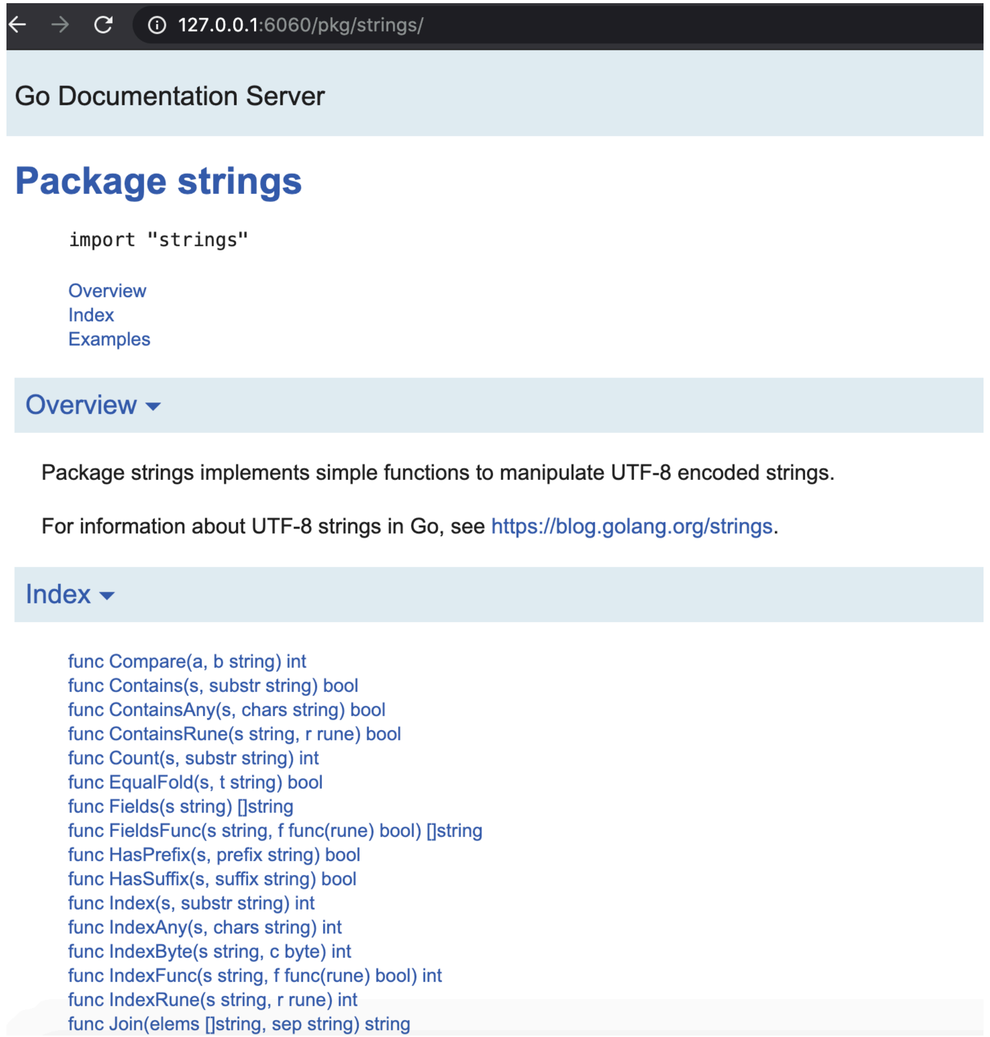
Code:

```
type ColorGroup struct {
    ID     int
    Name   string
    Colors []string
}
group := ColorGroup{
    ID:     1,
    Name:   "Reds",
    Colors: []string{"Crimson", "Red", "Ruby", "Maroon"},
}
b, err := json.Marshal(group)
if err != nil {
    fmt.Println("error:", err)
}
os.Stdout.Write(b)
```

Output:

```
{"ID":1,"Name":"Reds","Colors":["Crimson","Red","Ruby","Maroon"]}
```

官方文档阅读


127.0.0.1:6060/pkg/strings/
Go Documentation Server
Package strings
import "strings"
Overview
Index
Examples
Overview
Package strings implements simple functions to manipulate UTF-8 encoded strings.
For information about UTF-8 strings in Go, see https://blog.golang.org/strings.
Index
func Compare(a, b string) int
func Contains(s, substr string) bool
func ContainsAny(s, chars string) bool
func ContainsRune(s string, r rune) bool
func Count(s, substr string) int
func EqualFold(s, t string) bool
func Fields(s string) []string
func FieldsFunc(s string, f func(rune) bool) []string
func HasPrefix(s, prefix string) bool
func HasSuffix(s, suffix string) bool
func Index(s, substr string) int
func IndexAny(s, chars string) int
func IndexByte(s string, c byte) int
func IndexFunc(s string, f func(rune) bool) int
func IndexRune(s string, r rune) int
func Join(elems []string, sep string) string

官方文档阅读


127.0.0.1:6060/pkg/strings/#Contains
func Contains
func Contains(s, substr string) bool
Contains reports whether substr is within s.
Example
Code:
fmt.Println(strings.Contains("seafood", "foo"))
fmt.Println(strings.Contains("seafood", "bar"))
fmt.Println(strings.Contains("seafood", ""))
fmt.Println(strings.Contains("", ""))
Output:
true
false
true
true

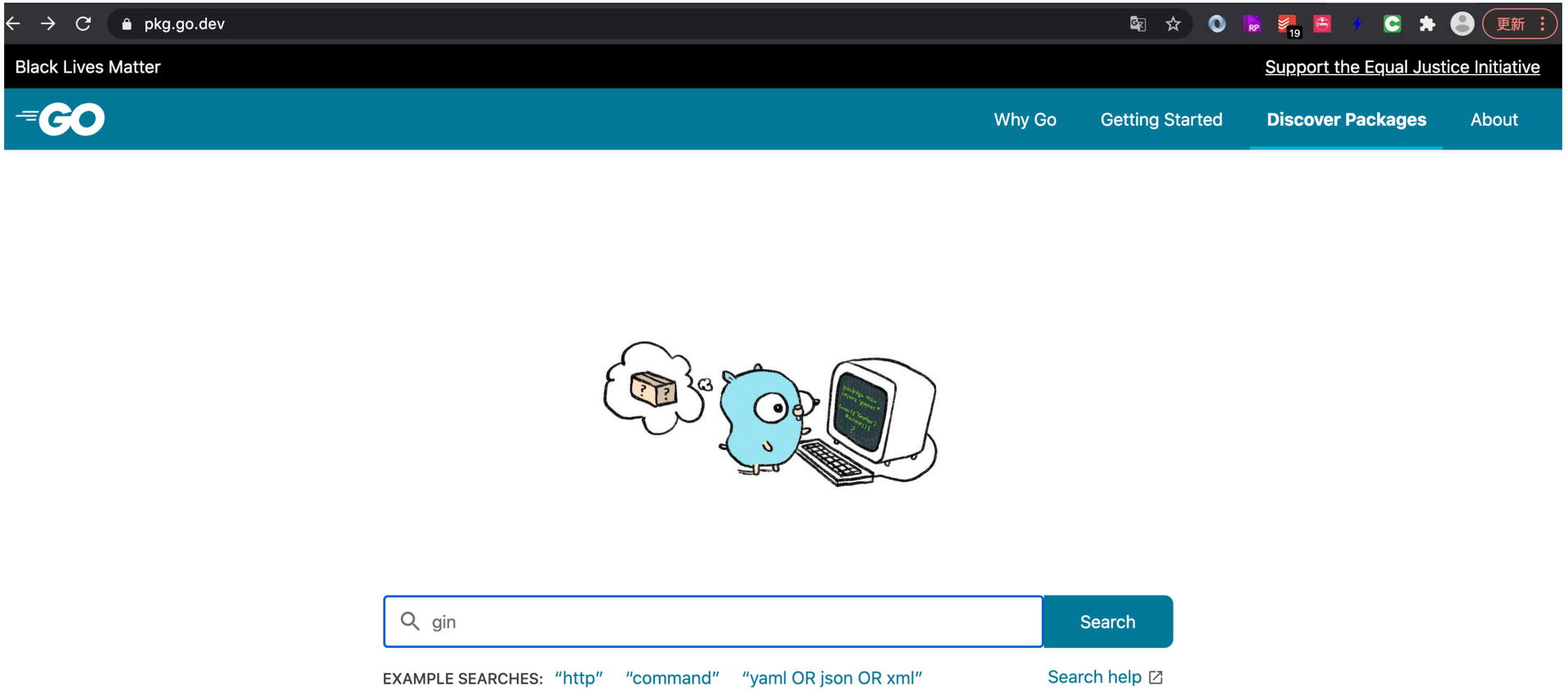
官方文档阅读

常见问题：

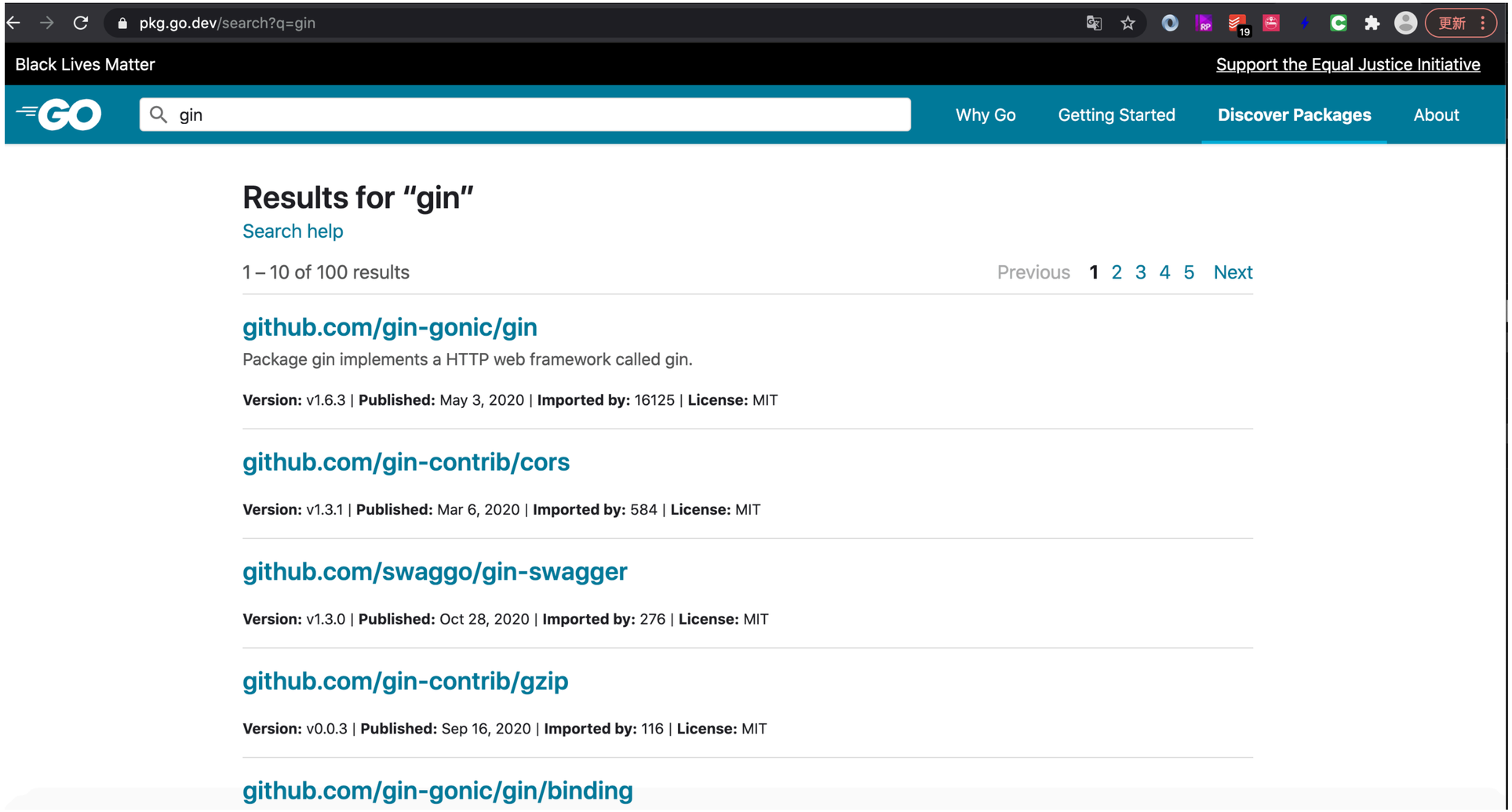
1. 英文看不懂？谷歌或者有道翻译
2. 功能描述看不懂？写测试代码，运行一下

欢迎补充。。。

第三方文档阅读


pkg.go.dev
更新
Black Lives Matter
Support the Equal Justice Initiative
GO
Why Go
Getting Started
Discover Packages
About
gin
Search
EXAMPLE SEARCHES: “http” “command” “yaml OR json OR xml”
Search help

第三方文档阅读


pkg.go.dev/search?q=gin
更新
Black Lives Matter
Support the Equal Justice Initiative
GO
gin
Why Go
Getting Started
Discover Packages
About
Results for “gin”
Search help
1 – 10 of 100 results
Previous 1 2 3 4 5 Next
github.com/gin-gonic/gin
Package gin implements a HTTP web framework called gin.
Version: v1.6.3 | Published: May 3, 2020 | Imported by: 16125 | License: MIT
github.com/gin-contrib/cors
Version: v1.3.1 | Published: Mar 6, 2020 | Imported by: 584 | License: MIT
github.com/swaggo/gin-swagger
Version: v1.3.0 | Published: Oct 28, 2020 | Imported by: 276 | License: MIT
github.com/gin-contrib/gzip
Version: v0.0.3 | Published: Sep 16, 2020 | Imported by: 116 | License: MIT
github.com/gin-gonic/gin/binding

第三方文档阅读

pkg.go.dev/github.com/gin-gonic/gin

Black Lives Matter — Support the Equal Justice Initiative

Search for a package | Why Go | Getting Started | Discover Packages | About

Discover Packages > github.com/gin-gonic/gin

gin package module

Version: v1.6.3 LATEST | Published: May 3, 2020 | License: MIT | Imports: 32 | Imported by: 16159

module 版本，点击后可选择指定版本

Jump to ...

README

- Gin Web Framework
- Contents
- Installation
- Quick start
- Benchmarks
- Gin v1. stable
- Build with
- API Examples
- Testing
- Users
- Documentation

点击后跳转到包文档

README

Gin Web Framework

build failing | codecov 99% | go report A+ | reference | gitter join chat | used by 9.7k projects | code helpers 51 | release v1.6.3 | TODOs 7

Gin is a web framework written in Go (Golang). It features a martini-like API with performance that is up to 40 times faster thanks to httprouter. If you need performance and good productivity, you will love Gin.

Contents

Expand ▾

Documentation

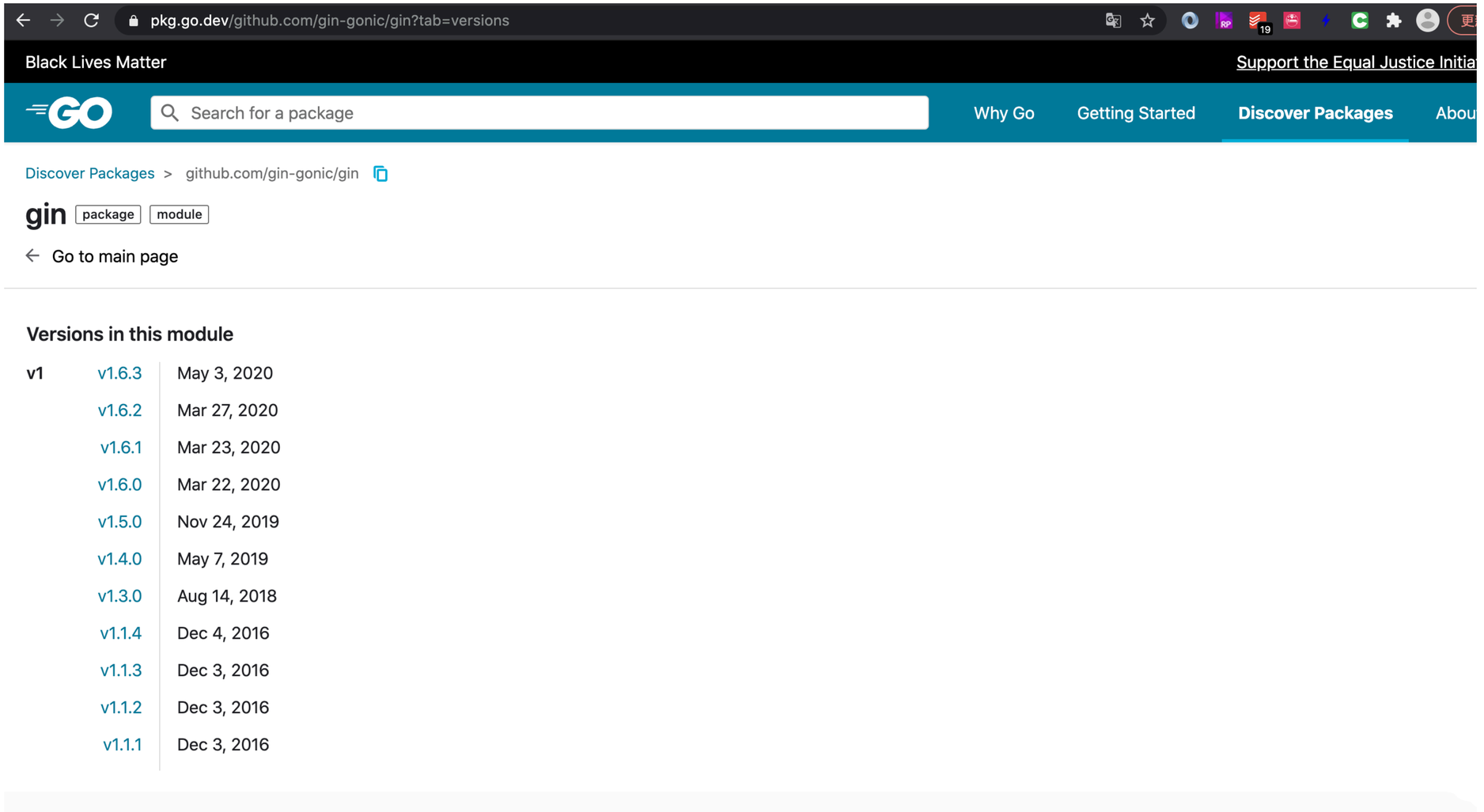
Details Learn more

- Valid go.mod file
- Redistributable license
- Tagged version
- Stable version

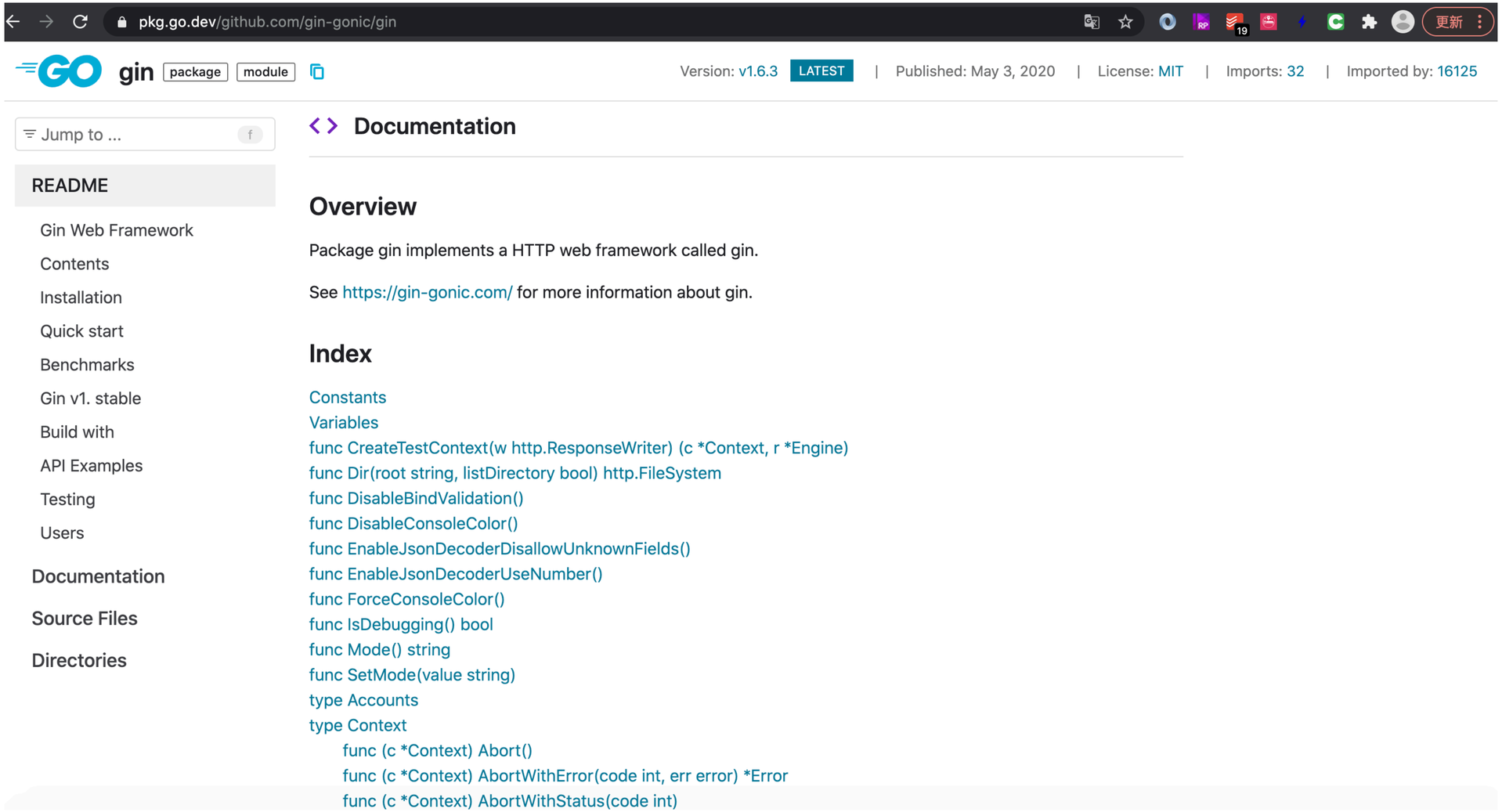
Repository

https://github.com/gin-gonic/gin

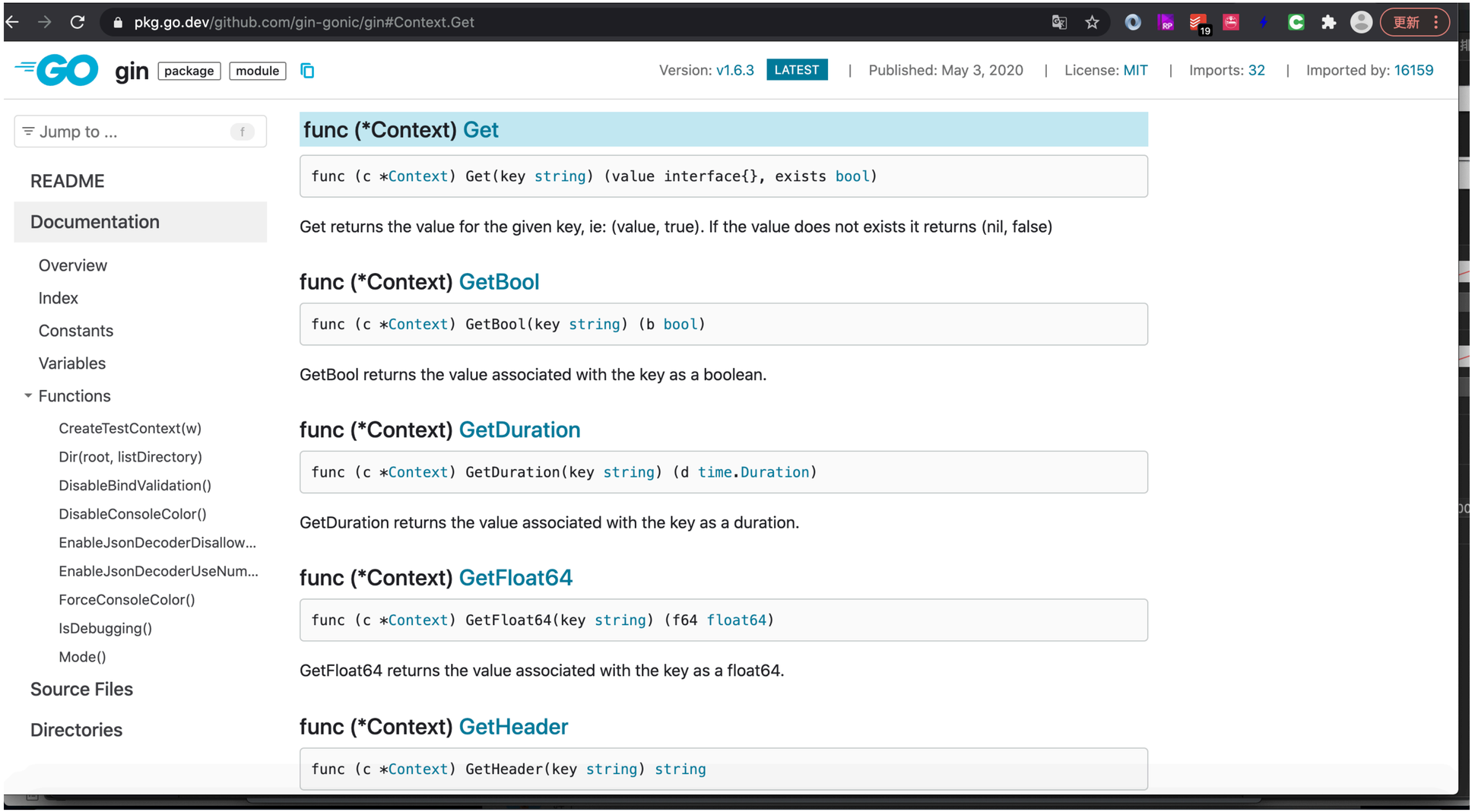
第三方文档阅读


pkg.go.dev/github.com/gin-gonic/gin?tab=versions
Black Lives Matter
Support the Equal Justice Initia
Search for a package
Why Go
Getting Started
Discover Packages
Abou
Discover Packages > github.com/gin-gonic/gin
gin
package
module
Go to main page
Versions in this module
v1
v1.6.3 May 3, 2020
v1.6.2 Mar 27, 2020
v1.6.1 Mar 23, 2020
v1.6.0 Mar 22, 2020
v1.5.0 Nov 24, 2019
v1.4.0 May 7, 2019
v1.3.0 Aug 14, 2018
v1.1.4 Dec 4, 2016
v1.1.3 Dec 3, 2016
v1.1.2 Dec 3, 2016
v1.1.1 Dec 3, 2016

第三方文档阅读


pkg.go.dev/github.com/gin-gonic/gin
更新
gin package module
Version: v1.6.3 LATEST | Published: May 3, 2020 | License: MIT | Imports: 32 | Imported by: 16125
Jump to ... f
README
Gin Web Framework
Contents
Installation
Quick start
Benchmarks
Gin v1. stable
Build with
API Examples
Testing
Users
Documentation
Source Files
Directories
Documentation
Overview
Package gin implements a HTTP web framework called gin.
See https://gin-gonic.com/ for more information about gin.
Index
Constants
Variables
func CreateTestContext(w http.ResponseWriter) (c *Context, r *Engine)
func Dir(root string, listDirectory bool) http.FileSystem
func DisableBindValidation()
func DisableConsoleColor()
func EnableJsonDecoderDisallowUnknownFields()
func EnableJsonDecoderUseNumber()
func ForceConsoleColor()
func IsDebugging() bool
func Mode() string
func SetMode(value string)
type Accounts
type Context
func (c *Context) Abort()
func (c *Context) AbortWithError(code int, err error) *Error
func (c *Context) AbortWithStatus(code int)

第三方文档阅读


pkg.go.dev/github.com/gin-gonic/gin#Context.Get
更新
gin
package
module
Version: v1.6.3
LATEST
Published: May 3, 2020
License: MIT
Imports: 32
Imported by: 16159
Jump to ...
README
Documentation
Overview
Index
Constants
Variables
Functions
CreateTestContext(w)
Dir(root, listDirectory)
DisableBindValidation()
DisableConsoleColor()
EnableJsonDecoderDisallow...
EnableJsonDecoderUseNum...
ForceConsoleColor()
IsDebugging()
Mode()
Source Files
Directories
func (*Context) Get
func (c *Context) Get(key string) (value interface{}, exists bool)
Get returns the value for the given key, ie: (value, true). If the value does not exists it returns (nil, false)
func (*Context) GetBool
func (c *Context) GetBool(key string) (b bool)
GetBool returns the value associated with the key as a boolean.
func (*Context) GetDuration
func (c *Context) GetDuration(key string) (d time.Duration)
GetDuration returns the value associated with the key as a duration.
func (*Context) GetFloat64
func (c *Context) GetFloat64(key string) (f64 float64)
GetFloat64 returns the value associated with the key as a float64.
func (*Context) GetHeader
func (c *Context) GetHeader(key string) string

第三方文档阅读


gowalker.org
首页 帮助
Twitter GitHub
搜索 738,760 个 Go 语言项目
Go Walker 是一个可以在线生成并浏览 Go 项目 API 文档的 Web 服务器，目前已支持包括 GitHub 等代码托管平台。
请输入项目路径或关键字
项目浏览记录
github.com/gomodule/redigo/redis
2021/3/18下午1:30:15
网站由 @Unknwon 建设，并由 Macaron 和 Xorm 提供动力。
Powered by

Q & A

谢谢!