go语言

GO语言跟它名字一样是门比较装逼的语言，鲜有菜鸟初学者教程，所以没有语言基础的话，理解起来会有点困难。闲来没事整理了一个GO零基础入门资料。

[安装与简介](http://code.google.com/p/golang-china/wiki/Install)

因为国内老被墙，被墙之后请在网上直接下载，[下载链接](https://code.google.com/p/go/downloads/list)

**开发环境编辑器**

1.Sublime，支持多标签，自动排版，高亮，简单的填补，简单的SHELL直接运行等。

[《Ubuntu下使用Sublime Text 2搭建Go开发环境》](http://www.cnblogs.com/yourihua/archive/2012/06/04/2529333.html)

[《Sublime Text 2搭建Go开发环境（Windows）》](http://www.cnblogs.com/sevenyuan/archive/2013/03/01/2938351.html)

2.vim + gocode

[《Go的Vim的高亮和自动补齐》](http://qizhanming.com/blog/2013/03/14/config-go-vim-highlight-and-autocomplete/)

3.[LiteIDE](https://code.google.com/p/liteide/):只用于GO环境开发的编辑器

4.[免费的golang在线开发环境](https://www.shiyanlou.com/courses/219)：由[实验楼](www.shiyanlou.com)与[StudyGolang](http://stuuydygolang.com)合作提供的在线实验室，可直接通过浏览器访问进行Go语言的学习与编程。

**一、对语言有初步感受**

[Go语言编程](http://www.shiyanlou.com/courses/11)：实验楼的GO语言基础课，手把手教学，极其容易上手。

[《GO指南》](http://go-tour-zh.appsp0t.com/)：简单的介绍，给有一定基础的人查看GO的基本特性语法规则。

**二、系统入门**

[跟无闻学GO语言](http://edu.51cto.com/course/course_id-1762.html)：无闻在51学院上的教学视频教程。

[《GO BY EXAMPLE》](https://gobyexample.com/)：GO的特性全部用简单代码给你展示，方便记不住的时候查询和巩固。

[《GO 标准库》](https://github.com/polaris1119/The-Golang-Standard-Library-by-Example/blob/master/preface.md): Polaris编写，一本有价值的入门书

[《GO WEB编程》](https://github.com/astaxie/build-web-application-with-golang/blob/master/ebook/preface.md):GO在WEB方面的编程应用学习指导

**三、书籍**

[《GO语言编程》](http://pan.baidu.com/share/link?shareid=494827&uk=1983309261)

[《Go并发编程实战》](http://www.ituring.com.cn/book/1525)

**四、社区**

[StudyGolang](http://studygolang.com)

[Golang中国](http://www.golangtc.com)

**五.熟悉API**

[API](http://zh.golanger.com/pkg/):多看官方的GO包类，熟悉各个功能模块接口。

**六.GITHUB上的推荐项目：**

1.GO与MYSQL的结合使用（作为入门熟悉）

<https://github.com/Go-SQL-Driver/MySQL>

<https://github.com/Go-SQL-Driver/MySQL/wiki/Examples>

2.[StudyGolang](https://github.com/studygolang/studygolang):GO语言的网站示范

**网站教程：**

[GO语言编程](https://www.shiyanlou.com/courses/11) and [GO语言开发2048](https://www.shiyanlou.com/courses/42) from [实验楼](https://www.shiyanlou.com)

[Go语言后台应用开发](http://www.ucai.cn/openclass/136) form [优才网](http://www.ucai.cn/)

[Go语言第一课](http://www.imooc.com/learn/345) form [慕课网](http://www.imooc.com/)

**入门书籍**

[《Go 入门指南》](https://github.com/Unknwon/the-way-to-go_ZH_CN)(《The Way to Go》中文版)

[《Go 编程基础》](https://github.com/Unknwon/go-fundamental-programming)

[《学习 Go 语言》](https://github.com/mikespook/Learning-Go-zh-cn)

[《GO 标准库》](https://github.com/polaris1119/The-Golang-Standard-Library-by-Example/blob/master/preface.md): Polaris出品，一本有价值的入门书籍。

[《Go Web 编程》](https://github.com/astaxie/build-web-application-with-golang)

[《Go并发编程实战》](http://www.ituring.com.cn/book/1525) and [《Go命令教程》](https://github.com/hyper-carrot/go_command_tutorial)由[@特价萝卜](http://weibo.com/freej)出品。

**视频资料**

[《Go编程基础》视频教程](http://www.tudou.com/listplay/Rbsk3XMy8fQ.html)：由[@Unknown](http://weibo.com/Obahua?from=myfollow_all) 出品，与书籍配套。

[《Go Web基础》视频教程](http://www.tudou.com/listplay/I2YCDzcJxYI.html)

[《Go名库讲解》视频教程](http://www.tudou.com/listplay/m1AWOKxI9nQ.html/)

**社区交流**

[StudyGolang](http://studygolang.com/)

[Golang中国](http://www.golangtc.com/)

[Go友团](http://golanghome.com/)

**其他补充**

[Go学习笔记](https://github.com/qyuhen/book) by 雨痕大神

[Golang在线开发环境](https://www.shiyanlou.com/courses/219) 由[Go语言中文网](http://studygolang.com/)和[实验楼](https://www.shiyanlou.com)共同打造，通过浏览器直接访问Golang在线开发环境，进行Go编程。

<http://books.studygolang.com/go-patterns/> go设计模式

<https://github.com/robfig/cron> golang定时器模块，超强！

<https://github.com/qiniu/py/blob/develop/README.md> go调用python

<https://www.golangtc.com/t/52cbcced421aa9215f000001> 七牛golang python绑定包需要改进的三个方面

<http://blog.csdn.net/Alsmile/article/details/48290223> GOPATH ,GOROOT,GOBIN 目录

<http://blog.csdn.net/cmbug/article/details/49339341> Golang import 包问题相关详解

<https://www.golangtc.com/download/package> 下载golang包

<http://mp.weixin.qq.com/s/w6hHkme-JwuDgEv95XLptA> 深度解析gRPC以及京东分布式服务框架跨语言实战

<https://github.com/Tencent/mars> Mars 是微信官方的跨平台跨业务的终端基础组件。JAVA编写

<http://www.jianshu.com/c/dc78f0ef880e> 一个人的go简书笔记

<http://studygolang.com/p/govendor> govendor 包管理

<http://www.infoq.com/cn/articles/golang-package-management> Golang的包管理之道

<https://golangtc.com/packages> go包国内镜像

<http://studygolang.com/articles/8104> golang年度使用总结，简洁不简单

<http://www.infoq.com/cn/news/2015/02/nsq-distributed-message-platform/> NSQ：分布式的实时消息平台，亿级访问

go 的不错框架 etcd、consul、groupcache、nsq, gin,beego,等

服务发现管理：Consul和ZooKeeper，Doozerd，Etcd的区别

[https://github.com/tim1020/godaemon#readme](https://github.com/tim1020/godaemon" \l "readme) 要让你的go应用支持daemon很简单，只需导入godaemon包即可，无需再调用任何方法，WINDOWS不可用，应为里面用到了syscall模块

<https://github.com/bitly/go-simplejson>

<http://studygolang.com/articles/345>

<http://xiaorui.cc/2016/03/06/golang%E8%A7%A3%E6%9E%90%E5%88%9B%E5%BB%BA%E5%A4%8D%E6%9D%82%E5%B5%8C%E5%A5%97%E7%9A%84json%E6%95%B0%E6%8D%AE/> 文章说到 Golang的json序列化果真是静态语言的范呀。各种的麻烦，各种的标准。 用pyhon的时候毫无顾忌，各种的数据各种的塞。 各个功能模块需要传递信息，一般是需要序列化的，常用的序列化模式有json、msgpack啥的。

这里就先扯扯golang的json模块。 对于相当复杂又各种嵌套的json数据，应该如何解析成go内置的数据类型？ 需要你写一堆的关联struct的。

golang json里的struct变量首字母需要大写的，如果给你的json是小写咋办？在type后面跟着别名就可以了，格式是 json:"字段名", 下面是反序列化的方法，这里我们介绍一个超级简单的反序列化模块 。simplejson ! 一个高度封装的json库，你不需要针对嵌套的json写各种interface及struct 。

<https://github.com/mitchellh/mapstructure> go map和struct转换

**<http://studygolang.com/articles/3950> [go语言 map转成struct结构（调用第三方包）](http://blog.csdn.net/zistxym/article/details/43534331)**

**可以方便的把 json string转换成map,把map和struct之间转换**

<http://www.cnblogs.com/yjf512/archive/2012/12/02/2798505.html> fnotify go跨平台实现监控文件夹

<http://www.infoq.com/cn/articles/inotify-linux-file-system-event-monitoring/> Inotify: 高效、实时的Linux文件系统事件监控框架

<https://github.com/lxn/walk> A Windows GUI toolkit for the Go

<http://studygolang.com/articles/2960?fr=sidebar> Go与GUI——GO语言的图形界面Walk

<http://blog.csdn.net/qq_35267530/article/details/74536646> **[Golang调用Python](http://blog.csdn.net/qq_35267530/article/details/74536646)**

<https://github.com/sbinet/go-python> naive go bindings to the CPython C-API

<https://github.com/go-python/gopy>

<https://github.com/qiniu/py> Golang bindings to the CPython C-API

<https://blog.gopheracademy.com/advent-2015/gopy/> gopy: extending CPython with Go

<http://studygolang.com/articles/547> Go 语言编写 CPython 扩展 goPy

<https://github.com/google/grumpy> 把python代码编译成GO代码

<https://www.kubernetes.org.cn/> **[Kubernetes 稳定版发布两年，代码提交次数破5万](https://www.kubernetes.org.cn/2492.html) 容器服务**

<http://www.infoq.com/cn/articles/Kubernetes-system-architecture-introduction> kubernetes 容器架构 谷歌开源,Kubernetes是Google开源的容器集群管理系统，其提供应用部署、维护、 扩展机制等功能，利用Kubernetes能方便地管理跨机器运行容器化的应用，其主要功能如下

<https://github.com/xiaonanln/goworld> go写的一个分布式游戏服务器，最近都有更新，比较活跃

<https://github.com/shiyanhui/hero> go模板引擎

[https://github.com/SlinSo/goTemplateBenchmark#full-featured-template-engines-2](https://github.com/SlinSo/goTemplateBenchmark" \l "full-featured-template-engines-2) go模板引擎性能测试，分别比较 预编译和不预编译的，Hero是一个高性能、强大并且易用的go模板引擎，工作原理是把模板预编译为go代码。Hero目前已经在[bthub.io](http://bthub.io/)的线上环境上使用

// Hero需要goimports处理生成的go代码，所以需要安装goimports. go get golang.org/x/tools/cmd/goimports

<https://github.com/flike/kingshard/blob/master/README_ZH.md> 目前已有上二十家公司在生产环境使用kingshard作为MySQL代理

<http://www.workerman.net/> Workerman是一款**纯PHP开发**的**开源高性能**的**PHP socket 服务器框架**

<http://www.swoole.com/> PHP的异步、并行、高性能网络通信引擎，使用纯C语言编写，提供了[PHP语言的异步多线程服务器](http://www.swoole.com/wiki/page/p-server.html)，[异步TCP/UDP网络客户端](http://www.swoole.com/wiki/page/p-client.html)，[异步MySQL](http://www.swoole.com/wiki/page/517.html)，[异步Redis](http://www.swoole.com/wiki/page/p-redis.html)，[数据库连接池](https://github.com/swoole/framework/blob/master/tests/async_mysql.php)，[AsyncTask](http://www.swoole.com/wiki/page/134.html)，[消息队列](http://wiki.swoole.com/wiki/page/289.html)，[毫秒定时器](http://www.swoole.com/wiki/page/244.html)，[异步文件读写](http://www.swoole.com/wiki/page/183.html)，[异步DNS查询](http://www.swoole.com/wiki/page/186.html)。 Swoole内置了[Http/WebSocket服务器端](http://www.swoole.com/wiki/page/326.html)/[客户端](http://www.swoole.com/wiki/page/p-http_client.html)、[Http2.0服务器端](http://www.swoole.com/wiki/page/326.html)。

<https://docs.gitea.io/zh-cn/> Gitea 是一个自己托管的Git服务程序。他和GitHub, Bitbucket or Gitlab等比较类似。他是从 [Gogs](http://gogs.io/) 发展而来

[http://colobu.com/2017/06/21/json-tricks-in-Go/#more](http://colobu.com/2017/06/21/json-tricks-in-Go/" \l "more) go中处理json小技巧

<https://github.com/json-iterator/go> A high-performance 100% compatible drop-in replacement of "encoding/json" 取代标准库的json库,Jsoniter 的 Golang 版本可以比标准库（encoding/json）**快 6 倍**之多。而且这个性能是在不使用代码生成的前提下获得的。

<http://jsoniter.com/index.cn.html>

go 技术牛人博客

<http://colobu.com/> 鸟我 rpcx 作者

<http://tonybai.com/> tonybai

<http://1234n.com/?tag/Go/> 达达

<http://fuxiaohei.me/> 傅小黑

<http://www.cnblogs.com/phpgo/> 不灭的炎

<http://www.cnblogs.com/golove/p/5909541.html> golang reflect包详解

<http://studygolang.com/articles/1256> Go的reflect机制和reflect包

<http://www.jianshu.com/p/f7f930152482?utm_campaign=hugo&utm_medium=reader_share&utm_content=note&utm_source=qq> **GO语言：json转map**

<http://www.markphelps.me/2017/08/20/optional-types-with-go-generate.html>

<https://www.golangtc.com/p/564429e3b09ecc72c300004c> gonet2基于grpc写的一个游戏服务器框架

<http://gonet2.github.io/> gonet2文档

<http://blog.csdn.net/q26335804/article/category/5726691> gonet2服务器架构分析

<https://http2.github.io/> 延伸阅读http2

<http://www.cnblogs.com/sevenyuan/p/6522431.html> leaf go游戏开发框架

<http://www.oschina.net/p/leafgo> leaf游戏服务器

<https://github.com/liangdas/mqantserver> mqant游戏服务器

<https://github.com/davyxu/cellnet> cellnet游戏服务器

<https://github.com/viphxin/xingo> <http://studygolang.com/p/xingo> xingo高性能网络库,游戏服务器

帧同步服务器: <https://github.com/viphxin/fighting>

mmo demo: <https://git.oschina.net/viphxin/xingo_demo>

xingo\_cluster demo: <https://github.com/viphxin/xingo_cluster>

<http://www.jianshu.com/p/c322edca985f> Tao - Go语言实现的TCP网络编程框架

<https://blog.minirplus.com/3539/> debian 安装vmtools

<https://blog.golang.org/context> <http://blog.csdn.net/sryan/article/details/51969129> net/context 笔记

<http://blog.csdn.net/dianyueneo/article/details/19158503> 微信收费事件背后被广泛忽略的技术细节——长连接心跳周期

http://www.jianshu.com/p/8175f51e662c 微信的智能心跳方案

<http://blog.csdn.net/Alsmile/article/details/48290223> GOROOT、GOPATH、GOBIN、project目录

<https://studygolang.com/articles/4331> golang time包

<http://stdlib.daoapp.io/> **Golang标准库文档(版本1.3.3)**

<https://studygolang.com/pkgdoc>

<http://www.flysnow.org/> 飞雪无情博客

<http://mirrors.flysnow.org/> 飞雪无情镜像

<http://blog.csdn.net/chszs/article/details/50076641> vscode golang开发环境

http://www.flysnow.org/2017/08/26/go-1-9-type-alias.html Go语言 | Go 1.9 新特性 Type Alias详解

<http://blog.jobbole.com/92290/> 你知道的RPC原理

<http://blog.jobbole.com/88984/> 一个故事讲清楚NIO

<http://blog.csdn.net/mindfloating/article/details/53872286> RPC使用上的一些注意点

<http://www.jianshu.com/p/f4f789846fde> RPC心得包含RPC消息体设计

<http://gad.qq.com/article/detail/7156090> RPC 本质问题

<http://tech.sina.com.cn/i/2016-05-10/doc-ifxryhhh1869879.shtml> 微博轻量级RPC框架Motan正式开源：支撑千亿调用

<https://git.oschina.net/airgzn/QQChatBot> QQ聊天机器人

<https://github.com/pandolia/qqbot> QQ聊天机器人 smartQQ 协议

<http://blog.csdn.net/mengxianhua/article/details/44201535> 远程调用服务(RPC)和消息(Message Queue)对比及其适用/不适用场合

<http://www.jianshu.com/p/8cc077f6dbb9> GRPC产生动机和设计原则

<http://geek.csdn.net/news/detail/128812> RPC框架选美

<http://www.imooc.com/article/18750> 从motan看RPC框架的设计

<http://blog.csdn.net/yeasy/article/details/52190007> probuffer和grpc你需要的知识

<https://segmentfault.com/a/1190000008672912> grpc发现均衡负载

<https://grpc.io/docs/> grpc官网

<https://github.com/google/protobuf> protobuf官网

<http://studygolang.com/articles/3192> grpc简单例子及资源

<http://blog.csdn.net/u011518120/article/details/54604615> protobuf3语言指南

<https://www.gitbook.com/book/smallnest/go-rpc/details> go rpc开发指南

<https://smallnest.gitbooks.io/go-rpc/content/> GO RPCX开发指南

<https://github.com/smallnest/rpcx> rpcx

<https://www.gitbook.com/book/smallnest/go-rpc/details> rpcx中的文档- RPC开发指南

<https://github.com/micro/go-micro> 类似 rpcx的 go rpc框架

https://github.com/coreos/etcd ETCD 是go语言编写的是用于共享配置和服务发现的分布式，一致性的KV存储系统。该项目目前最新稳定版本为2.3.0. 具体信息请参考[项目首页]和[Github]。ETCD是CoreOS公司发起的一个开源项目，授权协议为Apache

<https://github.com/coreos> CoreOS是一个基于Linux 内核的轻量级操作系统，为了计算机集群的基础设施建设而生，专注于自动化，轻松部署，安全，可靠，规模化。作为一个操作系统，CoreOS 提供了在应用容器内部署应用所需要的基础功能环境以及一系列用于服务发现和配置共享的内建工具。

<http://dockone.io/article/1026> 深入浅出CoreOS（一）：概述

<http://www.blogjava.net/yongboy/archive/2013/08/26/403325.html> coreos初体验

<https://github.com/google/omaha> CoreOS团队也使用了Chrome的更新引擎Omaha。Omaha使用的是一个基于XML客户端-服务器协议

<http://www.oschina.net/news/83719/tencent-tars-opensource> 腾讯开源微服务架构 Tars，高性能 RPC 开发框架

<https://github.com/Tencent/Tars>

<https://yq.aliyun.com/articles/11035> etcd ETCD系列之一：简介

<http://blog.csdn.net/shukebai/article/details/53464887> etcd介绍和简单使用

<https://yq.aliyun.com/articles/29897?spm=5176.100239.blogcont11035.14.x6DOtP> etcd系列之二：部署集群

<https://yq.aliyun.com/articles/86382?spm=5176.100239.blogcont29897.15.1c5kx5> etcd：从应用场景到实现原理的全方位解读

<http://blog.csdn.net/zouxinfox/article/details/5865249> google著名开源项目总结

gopkg.in/urfave/cli.v2

Package cli provides a minimal framework for creating and organizing command line Go applications.

<http://gopkg.in/urfave/cli.v2>

<https://www.maxmind.com/zh/about-maxmind> IP智能与在线欺诈检测工具供应商

<https://github.com/gonet2/geoip> 封装maxmind