

# Go 学习路径 -- 相关基础

作者: [james](#)

原文链接: <https://ld246.com/article/1516880227626>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

现在开始接触Go一段时间了，基本路径就是看基础学习材料，开始写项目，有问题找解决问题的方。这里记录一下学习过程。

## go相关文章

- [Golang适合高并发场景的原因分析](#)
- [go build 不同系统下的可执行文件](#)

Golang 支持在一个平台下生成另一个平台可执行程序的交叉编译功能。

1、Mac下编译Linux, Windows平台的64位可执行程序:

```
$ CGO_ENABLED=0 GOOS=linux GOARCH=amd64 go build test.go
```

```
$ CGO_ENABLED=0 GOOS=windows GOARCH=amd64 go build test.go
```

2、Linux下编译Mac, Windows平台的64位可执行程序:

```
$ CGO_ENABLED=0 GOOS=darwin GOARCH=amd64 go build test.go
```

```
$ CGO_ENABLED=0 GOOS=windows GOARCH=amd64 go build test.go
```

3、Windows下编译Mac, Linux平台的64位可执行程序:

```
$ SET CGO_ENABLED=0 SET GOOS=darwin SET GOARCH=amd64 go build test.go
```

```
$ SET CGO_ENABLED=0 SET GOOS=linux SET GOARCH=amd64 go build test.go
```

## go基础学习材料

- [GO学习基础材料](#)
- [《Go 编程基础》](#)
- <https://github.com/Unknwon>
- [Go: break label与goto label的区别](#)

## GO学习路径—系统包

### 1. flag包

- 参考官网: <https://godoc.org/flag>
- Package flag implements command-line flag parsing.

After all flags are defined, call `flag.Parse()` to parse the command line into the defined flags.

### 2. sync包

- 参考: [锁和 sync 包](#)
- 参考: [协程 \(goroutine\) 与通道 \(channel\)](#)
- A WaitGroup waits for a collection of goroutines to finish. The main goroutine calls `Add` to set the number of goroutines to wait for. Then each of the goroutines runs and calls `Done` when finished. At the same time, `Wait` can be used to block until all goroutines have finished.

# GO学习路径—第三方开发包

## 1. glog日志包

- Leveled execution logs for Go.
- 官网: <https://code.google.com/p/google-glog/>
- The repository contains an open source version of the log package used inside Google. The master copy of the source lives inside Google, not here.

## 2. goconfig配置文件工具包

- 参考官网: <https://github.com/miguel-branco/goconfig>
- 安装: `go get "github.com/msbranco/goconfig"`
- 但是该配置项目已经不更新且不建议使用了, 建议使用: <https://github.com/robfig/config>
- 安装: `go get github.com/robfig/config`