链滴

Docker 部署 go 项目

作者：GumKey

原文链接：https://ld246.com/article/1565244079032

来源网站：链滴

最近用go写了一个web应用，但用docker部署时候，对于如何加入依赖的第三方包，产生了困惑，边感谢大D的解惑，下面开始 docker部署go开发的web应用

一、go mod

1、什么是go mod

在桌面新建一个文件夹HelloWorld，在HelloWorld里新建index.go

```
// /Users/gumkk/Desktop/HelloWorld/index.go 文件
package main
import (
    "github.com/jinzhu/configor"
    "fmt"
)
func main() {
    fmt.Println("使用外部包测试", configor.Config{})
}
```

运行结果：异常

```
HelloWorld — -bash — 80×24
gumkkdeMacBook-Pro:HelloWorld gumkk$ go run index.go
index.go:3:5: cannot find package "github.com/jinzhu/configor" in any of:
        /usr/local/go/src/github.com/jinzhu/configor (from $GOROOT)
        /usr/local/gopath/src/github.com/jinzhu/configor (from $GOPATH)
```

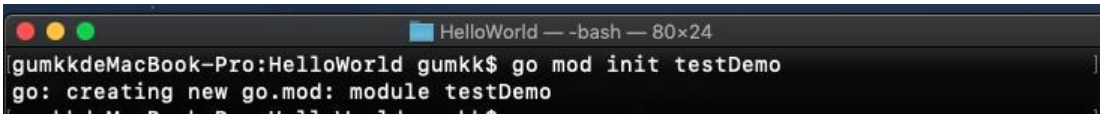
- go的项目需要引入第三库时，需要将项目放到$GOPATH/src目录下，否则报错/font>
- 如果想在你磁盘的任意位置运行你的go程序，你需要go mod，一个包依赖理工具

原文链接：Docker 部署 go 项目

2、go mod初尝试

$ go mod init testDemo

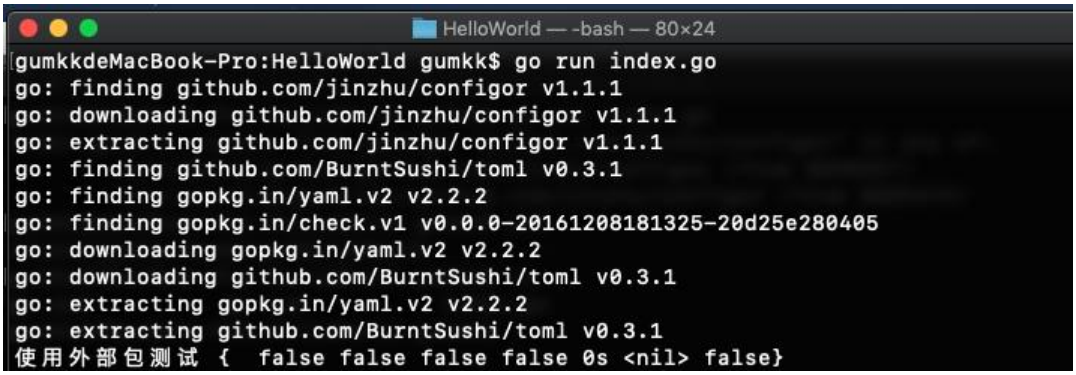
运行结果：



- 出现go: creating new go.mod: module testDemo表示初始化成功
- go mod init 后面定义你的项目名，当前目录会产生一个go.mod的文件</fon
>

$ go run index.go

`运行结果：成功



- 再次运行index.go文件，成功，当前目录会产生一个go.sum的文件

3、go mod 管理

可以用环境变量 GO111MODULE 开启或关闭模块支持，它有三个可选值：off、on、auto，默认值是 auto。

- GO111MODULE=off 无模块支持，go 会从 GOPATH 和 vendor 文件夹寻找包。
- GO111MODULE=on 模块支持，go 会忽略 GOPATH 和 vendor 文件夹，只根据 go.mod 下载赖。
- GO111MODULE=auto 在 GOPATH/src 外面且根目录有 go.mod 文件时，开启模块支持。

二、go build

基于上面使用go mod管理依赖包的项目HelloWorld，进行如下几种方编译，来熟悉go build命令，更多详情请看go build命令

1、go build 无参数编译

$ go build
$ ls
go.mod go.sum index.go testDemo

编译结果：产生testDemo文件， 默认为当前目录的项目名（go mod init testDemo），建议项目名你根目录文件名一致

2、go build+文件列表

```
$ go build index.go
$ ls
go.mod  go.sum  index  index.go
```

编译结果：产生index文件，和编译的文件名一致，编译多个文件时，默认以第一个文件名命名

3、go build+包

```
$ go build testDemo
$ ls
go.mod   go.sum   index.go   testDemo
```

编译结果：产生testDemo文件，当前包名为：testDemo（系统GOPATH目录下的项目，包名为：项根目录文件名）

4、go build 编译时的附加参数

附加参数	备注
-o	指定编译后生成的文件名
-v	编译时显示包名
-p n	开启并发编译，默认情况下该值为 CPU 逻核数
-a	强制重新构建
-n	打印编译时会用到的所有命令，但不真正执行
-x	打印编译时会用到的所有命令
-race	开启竞态检测

三、构建docker镜像

1、构建流程

- 项目开发使用go mod 管理依赖
- Dockerfile中配置docker环境变量：GO111MODULE=on，GOPROXY= https://goproxy.cn 代理）
- 使用go build 自动下载依赖的第三库

2、Dockerfile模版

```
FROM golang:alpine
```

```
WORKDIR /opt/LoopGraph/
COPY . /opt/LoopGraph/
ENV GO111MODULE=on
ENV GOPROXY=https://goproxy.cn
RUN apk add --no-cache gcc musl-dev git && go build -I -v

EXPOSE 8070
ENTRYPOINT ["./LoopGraph"]
```