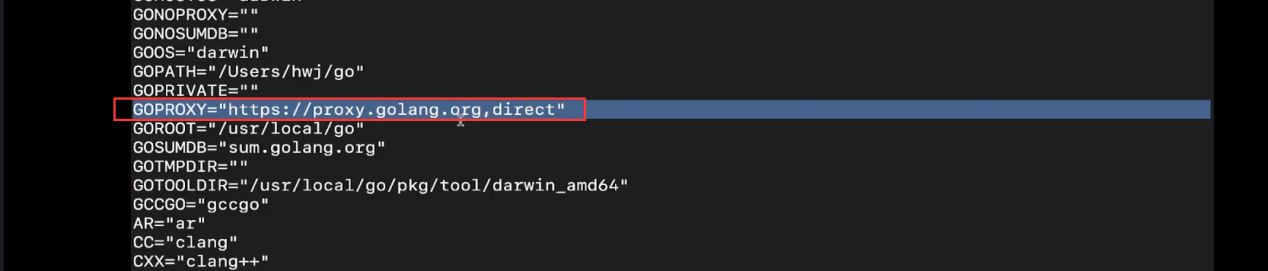
# go env

命令行输入：go env

查看当前go环境变量配置

# GOPROXY



GOPROXY就是拉取第三方依赖库的地址配置

对于国内用户来讲，默认地址连接十分缓慢，需要更换为国内镜像

打开网址：https://goproxy.cn/



执行命令：go env -w GOPROXY=https://goproxy.cn,direct

使用国内镜像还需要设置GO111MODULE参数

# GO111MODULE

## 什么是Go Modules

Go Modules 是 Go 语言的一种依赖管理方式

**没有包管理阶段**

* 一开始go发布的时候是没有包管理的
* go get命令会根据路径，把相应的模块获取并保存在$GOPATH/src
* 也没有版本的概念，master 就代表稳定的版本

**Go 1.13 阶段**

在 Go 1.13 下， GO111MODULE 的默认行为 (auto) 语义变了。

当存在 go.mod 文件时或处于 GOPATH 外， 其行为均会等同于 GO111MODULE=on。相当于 Go 1.13 下你可以将所有的代码仓库均不存储在 GOPATH 下。

当项目目录处于 GOPATH 内，且没有 go.mod 文件存在时其行为会等同于 GO111MODULE=off。

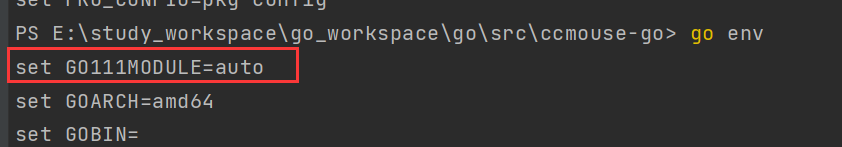
## Go Modules学习文档

https://studygolang.com/articles/31112?fr=sidebar

## 设置GO111MODULE

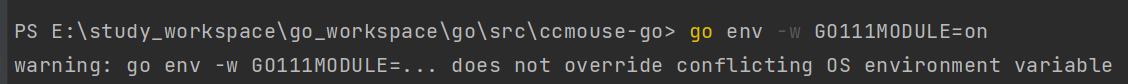
注意，国内镜像必须要GO111MODULE = on的状态才能使用

目前GO111MODULE 1.16.6版本是auto



执行：go env -w GO111MODULE=on

报错无法覆盖系统级的环境变量

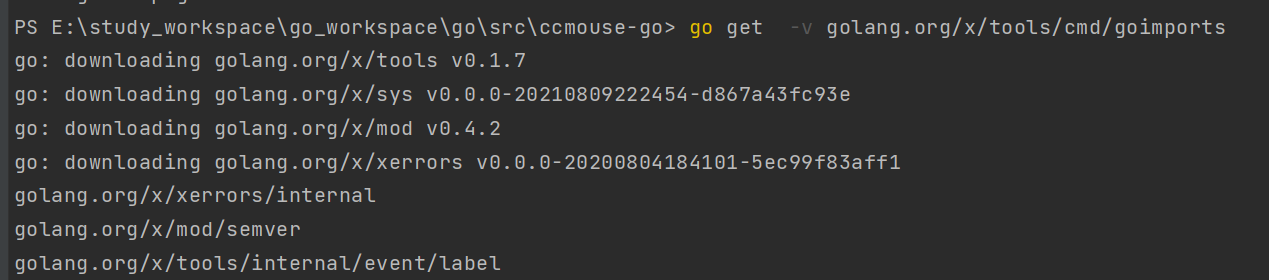


解决办法：别用编辑器的命令行，用系统自带的即可

# 测试国内镜像

## 测试命令

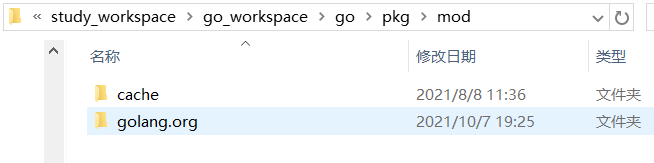
执行命令：go get -v golang.org/x/tools/cmd/goimports



## 依赖路径

Go Modules下载的依赖包在GOMODCACHE环境变量下，默认是$GOPATH/pkg/mod如：

GOMODCACHE=E:\study\_workspace\go\_workspace\go\pkg\mod



## go.mod文件

下载依赖后，会在项目根目录下生成一个go.mod文件，内容如下



此文件存放的是模块信息

module xxx为模块名