Go + VsCode 环境配置详解

1. 下载并安装

2. 配置GOPATH

3. VsCode 配置

4. F5 调试运行

1. 下载并安装

Git https://gitforwindows.org/

Go https://golang.google.cn/dl/

VsCode https://golang.google.cn/dl/

2. 配置GOPATH

自定义创建GOPATH目录，例如：D:\SoftWare\Go\workplace，并在该目录下创建src、pkg、bin三个文件夹在这里插入图片描述

环境变量GOPATH

在这里插入图片描述

打开控制台(win+r,输入cmd，按回车)，输入 go env 查看go配置信息

在这里插入图片描述

3. VsCode 配置

安装Go扩展

![在这里插入图片描述](https://img-blog.csdnimg.cn/20210614231945516.png?x-oss-process=image/watermark,type\_ZmFuZ3poZW5naGVpdGk,shadow\_10,text\_aHR0cHM6Ly9ibG9nLmNzZG4ubmV0L3FxXzQzMjM1ODY3,size\_16,color\_FFFFFF,t\_

修改settings.json文件

在这里插入图片描述

在这里插入图片描述

粘贴以下代码

{

"workbench.colorTheme": "Default Light+",

"go.alternateTools": {

"editor.wordWrap": "on",

// 如果useLanguageServer设为true，那么在编写代码时引入本地没有的package时，会自动下载安装

// 就是有时候会非常卡，保存go的编码文件时偶尔会卡死。这点你们自己取舍吧

"go.useLanguageServer": false,

"editor.minimap.renderCharacters": false,

"editor.minimap.enabled": false,

"terminal.external.osxExec": "iTerm.app",

"go.docsTool": "gogetdoc",

"go.testFlags": ["-v","-count=1"],

"go.buildTags": "",

"go.buildFlags": [],

"go.lintFlags": [],

"go.vetFlags": [],

"go.coverOnSave": false,

"go.useCodeSnippetsOnFunctionSuggest": false,

"go.formatTool": "goreturns",

"go.gocodeAutoBuild": false,

"go.goroot": "填写GOROOT路径",

"go.gopath": "填写GOPATH路径",

"go.autocompleteUnimportedPackages": true,

"go.formatOnSave": true,

"window.zoomLevel": 0,

"debug.console.fontSize": 16,

"debug.console.lineHeight": 30,

}

}

创建Go文件

在这里插入图片描述

安装 Go 依赖

修改 Go 国内镜像

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

安装依赖

自动安装

Windows平台按下Ctrl+Shift+P

输入框中输入>go:install，下面会自动搜索相关命令，我们选择 Go:Install/Update Tools 这个命令在这里插入图片描述

选中所有并回车在这里插入图片描述

弹出在这里插入图片描述

等待即可

手动安装

到 GOPATH 目录下创建 golang.org/x 目录

然后在 golang.org/x 路径下启动 CMD 或者 Git Bash

执行命令

git clone http://github.com/golang/tools

git clone http://github.com/golang/lint

git clone http://github.com/golang/net

git clone http://github.com/golang/text

git clone http://github.com/golang/crypto

git clone http://github.com/golang/sys

进入GOPATH目录，执行以下命令：

go get -u -v github.com/nsf/gocode

go get -u -v github.com/sqs/goreturns

go get -u -v github.com/golang/lint/golint

go get -u -v github.com/newhook/go-symbols

go install golang.org/x/tools/cmd/guru

go install golang.org/x/tools/cmd/gorename

go get -u -v github.com/rogpeppe/godef

go get -u -v github.com/lukehoban/go-outline

go get -u -v github.com/derekparker/delve/cmd/dlv

4. F5 调试运行

lanuch.json 文件配置

{

"version": "0.2.0",

"configurations": [

{

"name": "GoLaunch",

"type": "go",

"request": "launch",

"mode": "debug",

"remotePath": "",

"port": 2345,

"host": "127.0.0.1",

"program": "${fileDirname}//main.go", // 将 main.go 换成你的文件名

"env": {

"GOPATH": "D:/SoftWare/Go/workplace/", // 你的GOPATH

"GOROOT": "D:/SoftWare/Go/" // GOROOT，GO的安装目录

},

"args": [],

//"showLog": true

}

]

}