

Service Computing

Principle, Technology and Architecture for building effitive, elastic and solid services on cloud

安装 go 语言开发环境

- [1、安装 VSCode 编辑器](#)
- [2、安装 golang](#)
 - [3.1 安装](#)
 - [3.2 设置环境变量](#)
 - [3.3 创建 hello world!](#)
- [4、安装必要的工具和插件](#)
 - [4.1 安装 Git 客户端](#)
 - [4.2 安装 go 的一些工具](#)
- [5、安装与运行 go tour](#)
- [6、实验报告与作业要求](#)

既然选择后台开发，自然建议你在 Linux 环境下安装 go 语言开发环境。这里仅是 centos 7 安装的部分内容。

1、安装 VSCode 编辑器



如果你是 vim 或 emacs 用户，可以忽略本段内容。

如果你曾经是 Notepad++ 或 Sublime text 或 Atom 的用户，你不得不考虑改用微软 VSCode 做轻量级的编程。它采用 JavaScript 技术，兼容几乎所有流行的操作系统，特别是对中文支持堪称完美！它不仅是跨平台多语言软件开发工具，而且是 Linux 平台写 [Github Flavored Markdown](#) 的神器。[官方介绍](#)：

Visual Studio Code 是一个轻量级但功能强大的源代码编辑器，可在 Windows，macOS 和 Linux 桌面上运行。它内置了对 JavaScript，TypeScript 和 Node.js 的支持，并为其他语言（如 C++，C#，Java，Python，PHP，Go）和运行时（如 .NET 和 Unity）提供了丰富的扩展生态系统。

linux 下安装：

- [Running VS Code on Linux](#)

2、安装 golang

Golang [官方网站](#) 提供了不同平台的安装。可是 ……

golang [中国项目组](#) 提供了近可能好的中文服务。如果你有兴趣，发现问题可联系它们，使得中文服务变得更加完善。

3.1 安装

中文安装指南位置：<https://go-zh.org/doc/install>。然而 …… 链接的二进制发行文件呢？

建议使用系统包管理工具安装，即使不是最新版本，也不影响正常使用。以 CentOS 7 为例：

```
$ sudo yum install golang
```

安装到哪个目录了呢？

```
$ rpm -ql golang |more
```

测试安装：

```
$ go version
```

3.2 设置环境变量

go 对编译、包管理、测试、部署、运行提供全程支持，了解**环境配置**非常重要！

[go 语言工作空间](#)

1、创建工作空间

```
$ mkdir $HOME/gowork
```

2、配置的环境变量，对于 centos 在 ~/.profile 文件中添加：

```
export GOPATH=$HOME/gowork
export PATH=$PATH:$GOPATH/bin
```

然后执行这些配置

```
$ source $HOME/.profile
```

3、检查配置

```
$ go env
...
GOPATH = ...
...
GOROOT = ...
...
```

3.3 创建 hello world!

请退出当前用户，然后重新登陆！！

创建源代码目录：

```
$ mkdir $GOPATH/src/github.com/github-user/hello -p
```

使用 vs code 创建 hello.go

```
package main

import "fmt"

func main() {
    fmt.Printf("hello, world\n")
}
```

在终端运行!

```
$ go run hello.go
hello, world
```

4、安装必要的工具和插件

4.1 安装 Git 客户端

go 语言的插件主要在 Github 上，安装 git 客户端是首要工作。

```
$ sudo yum install git
```

4.2 安装 go 的一些工具

进入 vscode，它提示要安装一些工作，但 … 悲剧发生了 failed to install.

仔细检查，发现 <https://golang.org/x/tools/>，emmm 原来 golang.org 连不上!

1、下载源代码到本地

```
# 创建文件夹
mkdir $GOPATH/src/golang.org/x/
# 下载源码
go get -d github.com/golang/tools
# copy
cp $GOPATH/src/github.com/golang/tools $GOPATH/src/golang.org/x/ -rf
```

2、安装工具包

```
$ go install golang.org/x/tools/go/buildutil
```

退出 vscode，再进入，按提示安装!



查看 go 当前工作空间的目录结构，应该和官方文档 [如何使用Go编程](#) 的工作空间一致

细节参考：[获取Golang.org上的Golang Packages](#)

3、安装运行 hello world

```
$ go install github.com/github-user/hello
$ hello
```

5、安装与运行 go tour

细节参见：[《Go 语言之旅》](#)

```
$ go get github.com/Go-zh/tour/gotour
$ gotour
```

6、实验报告与作业要求

实验报告：

仔细阅读 官方文档 [如何使用Go编程](#)，并按文档写第一个包，做第一次测试。

请写在 git 仓库 Readme.md 中。

可选博客：

问题：以编写 hello.go 为例，我们需要使用 Git 在 github.com/github-user/hello 目录下创建 git 本地仓库并绑定 github 对应的远程仓库。常见的操作包括拉取、提交、同步等。尽管 vscode 能解决常用操作，一旦出现错误 git 命令就是救命稻草。

注意：请使用 https 协议操作远程仓库，不要使用 git 协议。请在使用网络资源时务必注意！

作业：用博客帮助他人是一种美德，是锻炼自己学习能力和表达能力的机会。请写 git 使用经验，帮助小白入门。课程用额外加分的方式作为回报。

Service Computing maintained by [pmlpml](#)

本站总访问量次，本站访客数人次，本文总阅读量次

Published with [GitHub Pages](#)