

搭建前端环境

1.安装 JavaScript 运行时 -- Nod

(1) 配置 NPM

npm config set prefix "D:\Program Files\nodejs\node_global" npm config set cache "D:\Program Files\nodejs\node_cache"

(2) 配置国内镜像源

npm config set registry <https://registry.npm.taobao.org>

2.安装编辑器 -- Visual Studio Code

(1) 安装插件

注意：插件并非越多越好，有些插件已经过时了，有些插件对性能有严重影响。

3.安装分布式版本控制工具 -- Git

(1) 设置Git的user name和email:

打开git.exe，输入命令：

```
$ git config --global user.name ""
```

```
$ git config --global user.email ""
```

(2) 生成SSH密钥过程:

查看是否已经有了ssh密钥：cd ~/.ssh

如果没有密钥则不会有此文件夹，有则备份删除

生成密钥：

```
$ ssh-keygen -t rsa -C "xxxx@vchangyi.com"
```

按3个回车，密码为空。

Your identification has been saved in /home/tekkub/.ssh/id_rsa.

Your public key has been saved in /home/tekkub/.ssh/id_rsa.pub.

The key fingerprint is:

最后得到了两个文件：id_rsa和id_rsa.pub

登陆gitlab, Profile Settings -> SSH Keys -> ADD SSH KEYS ,找到本地的id_rsa.pub文件，复制出里面的内容，添加到 key 内。

(3) 创建版本库

```
git init  
ls -ha
```

(4) 常用命令

workspace (工作区)：在电脑里能看到的目录。staging area (暂存区/缓存区)：英文叫 stage 或 index。一般存放在 .git 目录下的 index 文件 (.git/index) 中，所以我们把暂存区有时也叫作索引 (index)。local repository (本地仓库)：工作区有一个隐藏目录 .git，这个不算工作区，而是 Git 的版本库。remote repository (远程仓库)

(5) 克隆远程仓库

```
git clone  
  
git clone git@github.com:fsliurujie/test.git --SSH协议  
git clone git://github.com/fsliurujie/test.git --GIT协议  
git clone https://github.com/fsliurujie/test.git --HTTPS协议
```

(6) 本地仓库和远程仓库关联 (同步)

```
$ git remote add origin git@server-name:path/repo-name.git  
$ git remote add origin git@github.com:{用户名}/{远程仓库名}.git
```

关联后，使用命令如下第一次推送master分支的所有内容；

```
$ git push -u origin master
```

此后，每次本地提交后，只要有必要，就可以使用如下命令推送最新修改；

```
$ git push origin master
```