

Git版本管理工具

git简介

- **简介**

Git是一款免费、开源的分布式版本控制系统

- **特点**

Git是一个开源的分布式版本控制系统，可以有效，高速的处理从很小到非常大的项目版本管理

git基础配置

1.配置用户名(提交时会引用)

```
git config --global user.name "你的用户名"
```

2.配置邮箱

```
git config --global user.email "你的邮箱"
```

3, 编码配置

避免git gui中的中文乱码

```
git config --global gui.encoding utf-8
```

避免 git status显示的中文文件名乱码

```
git config --global core.quotePath off
```

4, 其他

```
git config --global core.ignorecase false
```

git ssh key pair配置

1, 在git bash命令行窗口中输入:

```
ssh-keygen -t rsa -C "你的邮箱"
```

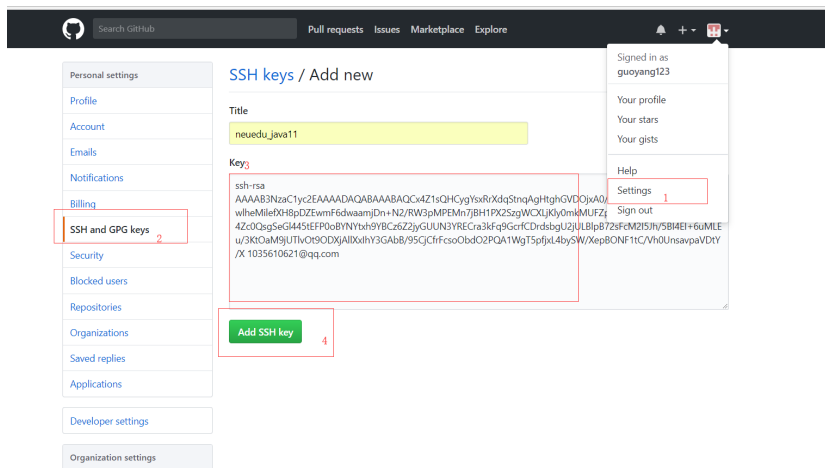
2,然后一路回车, 不要输入任何密码之类, 生成ssh key pair

3,在用户目录下生成.ssh文件夹, 找到公钥和私钥

```
id_rsa id_rsa.pub
```

4,将公钥的内容复制

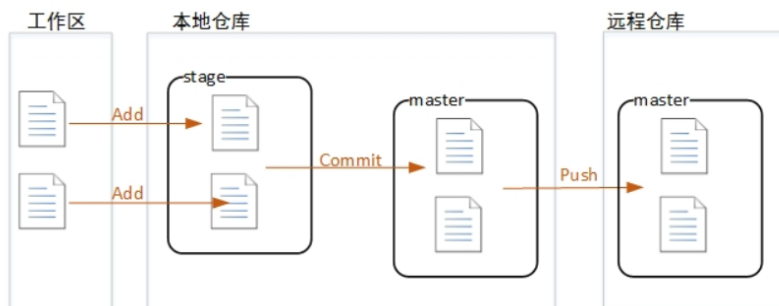
5, 进入github网站, 将公钥添加进去



git验证

执行git --version，出现版本信息，安装成功。

git工作原理



git常用命令

git init 创建本地仓库

git add 添加到暂存区

git commit -m "描述" 提交到本地仓库

git status 检查工作区文件状态

git log 查看提交committed

git reset --hard committid 版本回退

git branch 查看分支

git checkout -b dev 创建并切换到dev分支

切换分支: git checkout 分支名

拉取: git pull

提交: git push -u origin master

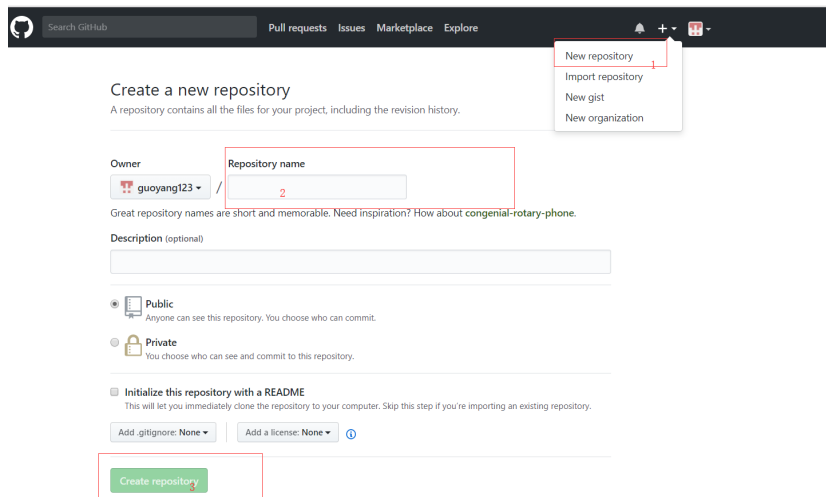
分支合并: git merge branchname

git push origin test:master // 提交本地test分支作为远程的master分支 //好像只写这一句，远程的github就会自动创建一个test分支

git push origin test:test // 提交本地test分支作为远程的test分支

github上创建一个项目

本地仓库与远程仓库



关联

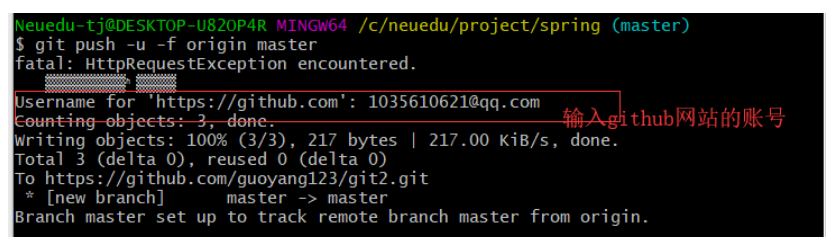
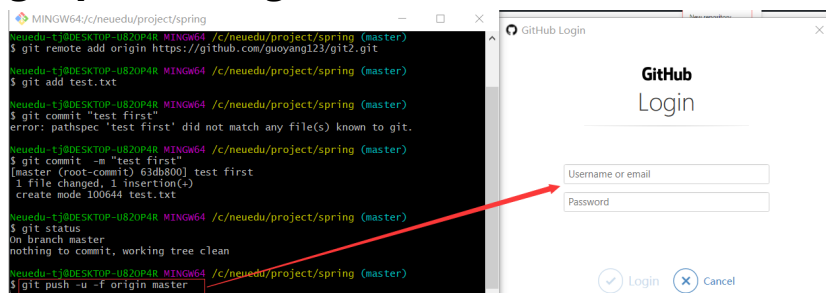
git remote add origin "远程仓库地址"

第一次向远程仓库推送

git push -u -f origin master

以后提交到远程

git push origin master



企业项目开发模式

项目采用：

- 分支开发，主干发布
 - 创建分支：git checkout -b v1.0 origin/master
 - 将分支推送到远程 git push origin HEAD -u
 - 检查远程，发现多了v1.0分支

项目提交到github

.gitignore文件 :告诉Git哪些文件不需要添加到版本管理中

忽略规则：

<https://www.cnblogs.com/kevingrace/p/5690241.html>

此为注释 - 将被 Git 忽略

*.a # 忽略所有 .a 结尾的文件

!lib.a # 但 lib.a 除外

/TODO # 仅仅忽略项目根目录下的 TODO 文件，不包括 subdir/TODO

build/ # 忽略 build/ 目录下的所有文件

doc/*.txt # 会忽略 doc/notes.txt 但不包括 doc/server/arch.txt

git add . //提交所有

码云：

<https://www.liaoxuefeng.com/wiki/0013739516305929606dd18361248578c67b8067c8c017b000/00150154460073692d151e784>

问题集锦

1,

error: failed to push some refs to 'https://github.com/guoyang123/guoyang123.github.io.git'

出现错误的主要原因是github中的README.md文件不在本地代码目录中

可以通过如下命令进行代码合并【注：pull=fetch+merge】

git pull --rebase origin master

