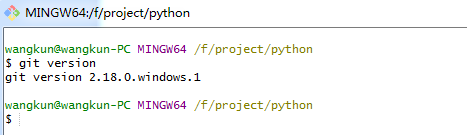
git 的使用

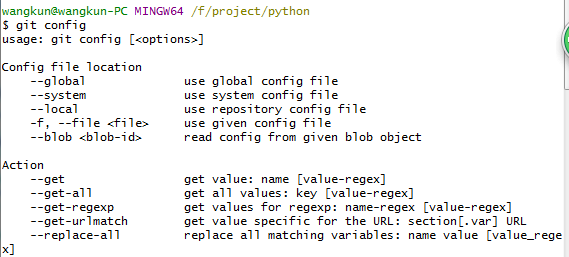
1. 查看git 版本信息

命令：git version

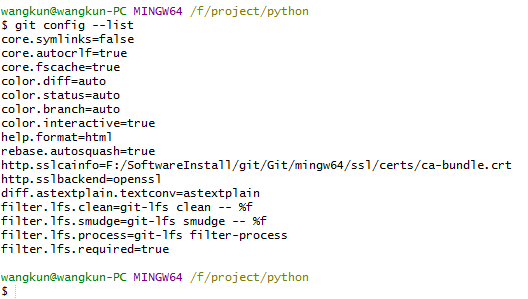


1. 配置git

命令git congig：可查看config用法



命令git config --list: 查看配置信息



命令git config --global user.name XXX:配置用户名

命令git config --global user.email XXX:配置邮箱

命令git config --list:

1. 版本初始化

命令：git init ，会在响应目录下生成一个.git 文件夹记录工程信息

1. git 的3层结构

Working directory ------- 工作区

Staging index ------- 暂存区

git directory(Repository) ------- 版本库

1. git 中文件的4中状态

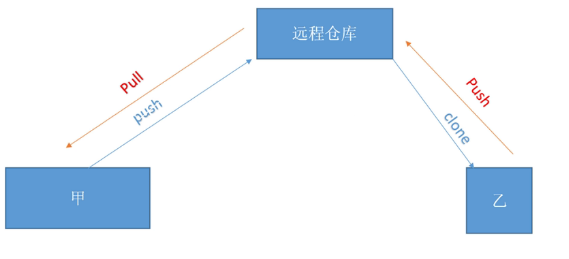
Untracked ------- 未被追踪

Modified ------- 文件被修改但还没有被增加到暂存区

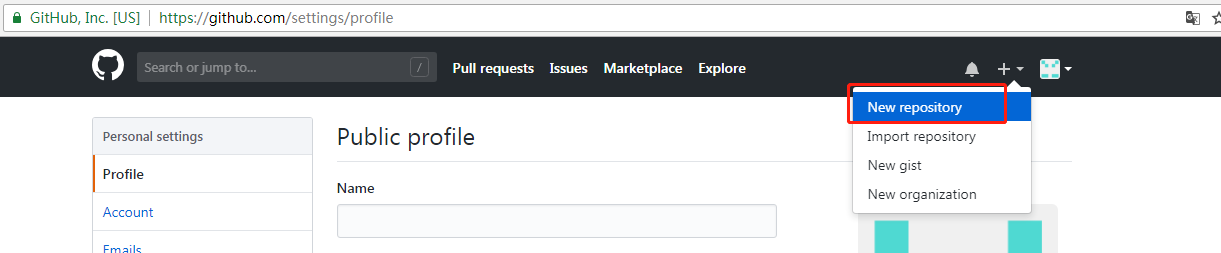
Staged ------- 添加到了暂存区，但还没提交

Committed ------- 已提交到本地仓库中

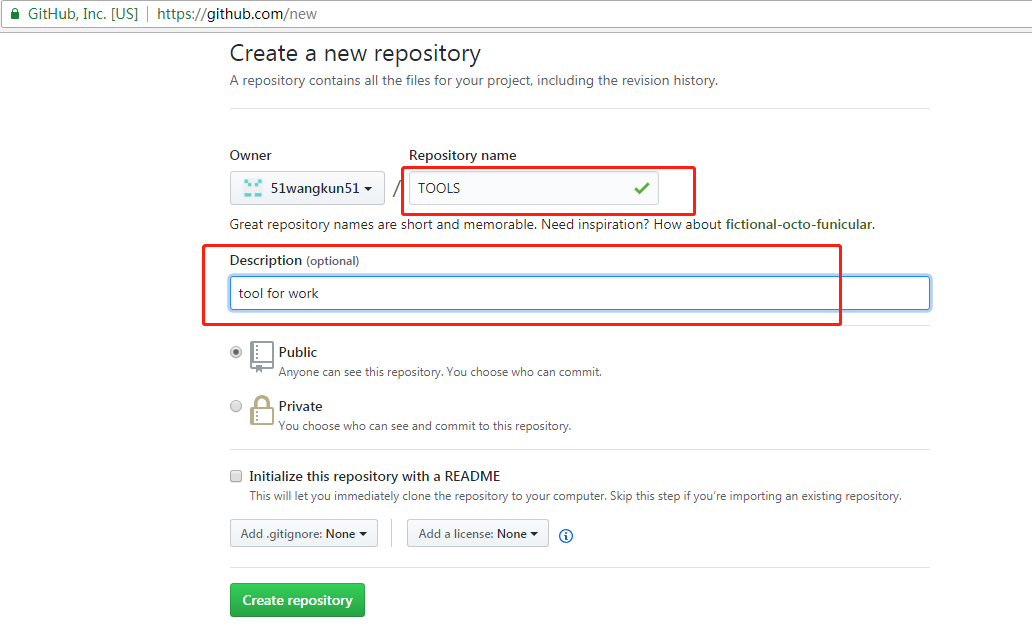
1. 远程仓库



①新建远程仓库



增加工程名和描述



1. git 常用命令

git status 查看目录下所有文件的状态

git add . 将本目录下的所有文件编程跟踪状态（增加单个文件用git add XXX）

git commit -m “XXX” 提交并写备注信息

git commit -am “XXX” 等同于先后执行以上命令

git rm --cached XXX(filename) 当文件被add后，删除暂存区，但不删除工作区

git rm --f XXX(filename) 当文件被add后，删除暂存区和工作区

git mv XXX 修改文件名

git log --oneline 单行显示所有记录

git commit --amend 撤销上一次的提交并将暂存区的文件重新提交（或者改变上次提交的备注）

git branch XXX 建立新的分支

git checkout XXX(branch name) 切换到新的分支

git checkout -b XXX以上两个命令的合并

git checkout -- XXX(filename) 拉取暂存区的文件并替换工作区的文件

git reset HEAD XXX(filename) 拉取最后一次提交的这个文件到暂存区，改操作不影响工作区

git branch 查看本地所有分支

git reset YYY(version number) XXX(filename) 将YYY版本号提交操作中的XXX文件拉回到暂存区

git branch -d XXX(branch name) 删除分支

git merge dev 合并dev分支到当前分支

git diff 默认比较工作区和暂存区的差异

git diff --staged 比较暂存区和版本库的差异

git diff 版本号 版本号

git diff 分支名 分支名

git stash将工作区的修改暂存起来（删除工作区的修改）

git stash list 查看暂存号

git stash apply 暂存号

git stash drop 暂存号

