**Github笔记**

# 基础

## 简述

Github.com

借助git托管项目

## 基本概念

### 仓库（Repository）

用来存放项目代码，每个项目对应一个仓库，多个开源项目则有多个

### 收藏（Star）

类似视频收藏，收藏项目，方便下次查看

### 复制克隆项目（fork）

#### 1.复制

Ghaya/test仓库

forked之后

Kosame/test仓库

Forked from Ghaya/test仓库

#### 2.发起请求（pull Request）

在Kosame在仓库中新建一个update.php后，可以

发起请求（pull request）请求Ghaya进行更新合并

#### 3.关注（Watch）

关注项目，当项目更新可以接收到通知

#### 4.事务卡片（Issue）

发现代码bug，讨论使用

### 翻墙软件 Shadowsocks

### 删除

#### git 删除

git rm <fileName>

#### 手动删除并同步

git add –A

git add --all . 这里 -A 是 --all 的一个简写。

### 忽略

创建.gitignore文件

在里面编写规则，并且add commit 该文件（.gitignore）

规则如下

# 此为注释 – 将被 Git 忽略

# 忽略所有 .a 结尾的文件

# \*.a

# 但 lib.a 除外

# !lib.a

# 仅仅忽略项目根目录下的 TODO 文件，不包括 subdir/TODO

# /TODO

# 忽略 build/ 目录下的所有文件

# build/

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

# doc/\*.txt

# 忽略 doc/ 目录下所有扩展名为 txt 的文件

# doc/\*\*/\*.txt

查看忽略规则

### 版本回退

git log 查看历史提交记录的commit id

git reset –hard [commit id] 回退到commit id 的版本

# 远程仓库管理

## # 配置用户名

git config --global user.name "GhayaStory" //（ "username"是自己的账户名，）

## # 配置邮箱

git config --global user.email "578532261@qq.com" //("username@email.com"注册账号时用的邮箱)

git config --global --list 命令查看配置是否OK

## 查看ssh

ls -la ~/.ssh

cat ~/.ssh/id\_rsa.pub

## 生成ssh 继续刚才的操作，在命令框中输入以下命令，

ssh-keygen -t rsa

ssh-keygen -t rsa -C "youremail@example.com"

ssh-keygen -t rsa -b 4096 -C "你的邮箱地址"

### 添加ssh到ssh-agent中

eval "$(ssh-agent -s)"

## 测试ssh

测试一下配置是否成功，在Git Bush命令框（就是刚才配置账号和邮箱的命令框）中继续输入以下命令，回车

ssh -T git@github.com

## 提交文件到本地仓库

当前目录创建版本库

右键 git bash here

git init 初始化当前目录git仓库

git add file1.txt file2.txt 添加文件到仓库

git commit –m “remarks” 提交到仓库

git log --pretty=oneline 查看提交记录

git reset --hard HEAD^ 回退到上一个版本

git reset --hard 27e1056 退回到指定版本 任意版本

car file.txt 查看文件内容

git reflog 查看命令记录

-----------------------

git remote -v 查看远程仓库地址

## 克隆一个远程仓库

git clone git@github.com:GhayaStory/testgitread.git

git clone git@github.com:michaelliao/gitskills.git

## 关联本地仓库到远程仓库

git remote add origin [git@github.com:GhayaStory/testgit.git](mailto:git@github.com:GhayaStory/testgit.git)

git remote add origin git@github.com:GhayaStory/LearnProject.git

git remote add origin git@github.com:GhayaStory/Note.git

## 推送当前仓库

git push -u origin master

git push origin master

与远程仓库的分支建立关联，关联后可以直接git push

git push --set-upstream origin master

## 更新远程仓库

git pull --rebase origin master

git pull

### 强制同步

git fetch –all

git reset –hard origin/master

git pull

**git强制覆盖本地命令**： git fetch --all && git reset --hard origin/master && git pull

## 分支管理

### 分支建立关联

设置默认分支

git branch --set-upstream-to=origin/master master

创建仓库命令

下表列出了 git 创建仓库的命令：

命令 说明

git init 初始化仓库

git clone 拷贝一份远程仓库，也就是下载一个项目。

提交与修改

Git 的工作就是创建和保存你的项目的快照及与之后的快照进行对比。

下表列出了有关创建与提交你的项目的快照的命令：

命令 说明

git add 添加文件到暂存区

git status 查看仓库当前的状态，显示有变更的文件。

git diff 比较文件的不同，即暂存区和工作区的差异。

git commit 提交暂存区到本地仓库。

git reset 回退版本。

git rm 将文件从暂存区和工作区中删除。

git mv 移动或重命名工作区文件。

提交日志

命令 说明

git log 查看历史提交记录

git blame <file> 以列表形式查看指定文件的历史修改记录

远程操作

命令 说明

git remote 远程仓库操作

git fetch 从远程获取代码库

git pull 下载远程代码并合并

git push 上传远程代码并合并