Git

什么是Git？

Git是作为我们源代码管理、版本控制和团队开发时经常用的工具。

Git是如何实现源代码管理的？

1. 保存开发过程，它可以详细的保存我们每一次提交的内容以及修改内容。
2. 实验新特性而不破坏现有数据，产品需求在更新，我们需要开发新特性，为了不破环现有数据，我们可以使用git中的分制管理。
3. 多人合作开发，我们每个开发人员都可以从远程仓库下载最新版本，然后进行修改，提交，最终实现一个完整效果，高效的管理我们的项目。
4. Git 的安装步骤：
5. 下载Git安装包，官方下载地址：<http://msysgit.github.io/>
6. 安装，详细安装过程：<http://jingyan.baidu.com/article/a3a3f811d4cd308da2eb8ad1.html>
7. 安装完成之后，在开始菜单找到“Git Bash”，蹦出一个类似命令行窗口的东西，说明Git安装成功。
8. Git安装完成之后进行如下配置：

在Git Bash 命令行窗口中输入命令： $ git config –global user.name “wangChenYang”

$ git config –global user.email “email@xx.com”

1. 创建版本库：

例如： （1）在桌面创建一个文件夹： myProject;

(2) $ cd /c/Users/ly/Desktop/myProject;

(3) $ git init; 将该文件夹变为 git 可管理的创库。

1. 简单运用git命令（添加文件到git仓库）

4-1. $ git status 查看git状态，看有没有要提交的文件。

4-2 .$ git add filesName 将文件添加到git 仓库。

4-3. $ git commit -m “描述信息” 将文件提交到 git 仓库。

4-4. $ git diff 查看修改的内容。

4-5. $ git log 输出日志 （git log –oneline ,显示日志简略信息）

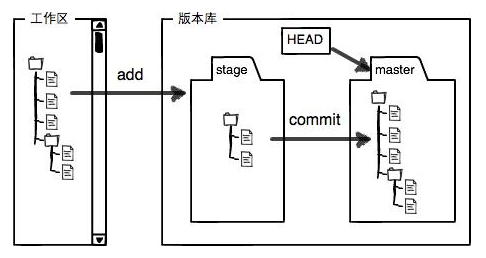
1. 理解工作区、暂存区和持久区。

stage:暂存区

master： 指向当前提交， HEAD指向当前分支。

git add ，该命令是将工作区的内容提交到暂存区。

git commit, 该命令是将暂存区的内容提交到分支，即持久区。



1. 撤销修改

6-1. $ git checkout – file 撤销工作区最后一次修改的代码

6-2. $ git reset HEAD file 撤销暂存区的修改，回退到工作区

6-3. $ git reset –hard 版本号 回退到仓库中的指定版本

1. 分支管理

7-1. 查看分支，当前分支前会加\*号 $ git branch

7-2. 创建分支 $ git branch masterName

7-3. 切换分支 $ git checkout masterName

(7-2、7-3 合并 $ git checkout -b masterName)

7-4. 合并指定分支 到 当前分支 $ git merge masterName

7-5. 删除分支 $ git branch -d masterName

1. 远程仓库

8-1. 在GitHub 网站上注册一个账号，获取免费的git远程仓库。

8-2. 登陆GitHub注册的账号。

SSH是计算机之间的加密通信协议。

假设A 是中央服务器，B为客户端；当B想要访问中央服务器上的资源（加密资源），那么中央服务器必须能够识别客户端。

中央服务器如何识别客户端？

第一步：客户端首先生成公钥和私钥这两个秘钥对。（8-3）

第二步：中央服务器需要添加客户端生成的公钥（8-4）

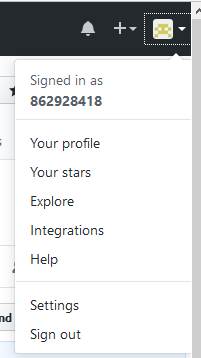
客户端如何访问中央服务器中的资源？

通过 8-6 中的命令，从远程仓库克隆一份到本地。当然也可以将本地内容推送到远程仓库，

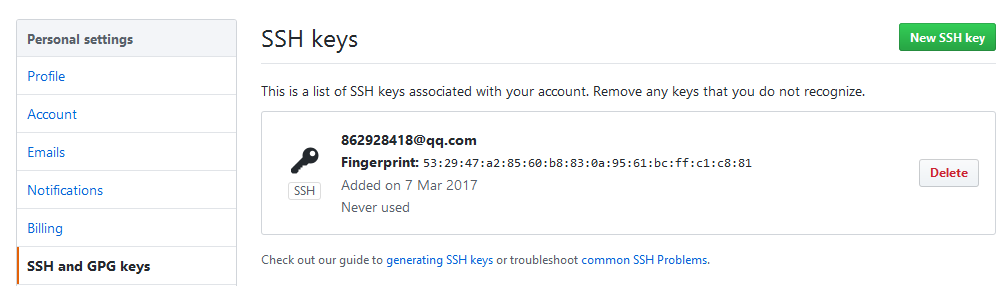
因为本地Git仓库和GitHub远程仓库之间的传输是通过SSH加密的，所以需要设置以下内容：

8-3. 创建SSH Key，$ ssh-keygen -t rsa -C [email@xxx.com](mailto:email@xxx.com), 一路回车，然后到用户主目录中查看SSH Key的秘钥对是否生成，即id\_rsa 和 id\_rsa.pub这两个文件

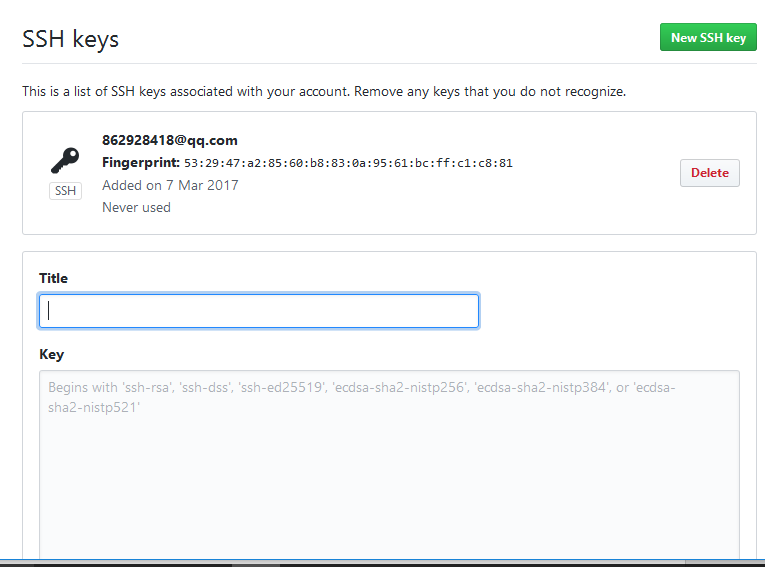
8-4. 打开用户目录中的settings



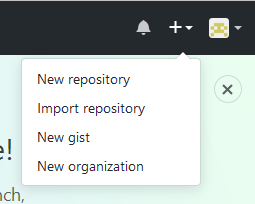
然后点击 **SSH and GPG keys**，接着点击 **New SSH key** 添加SSH key。



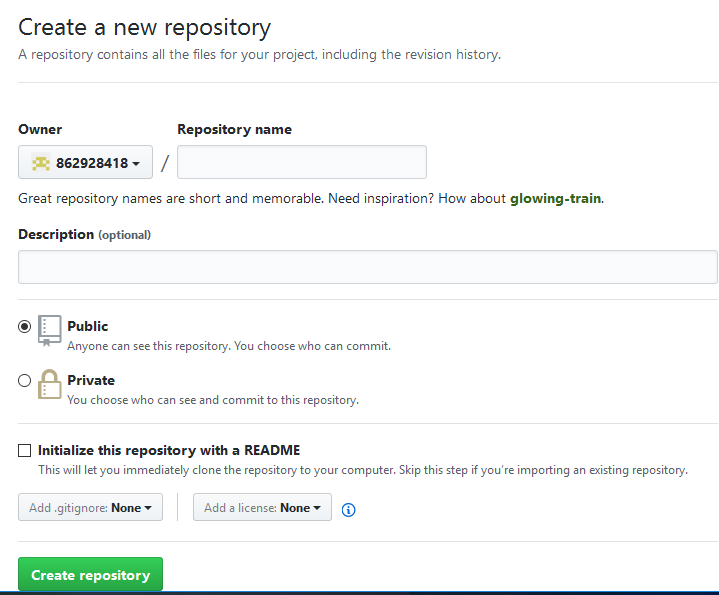
接着填写任意的 title（可以不填写，默认为你的邮箱），在key的文本框中粘贴 id\_rsa.pub 文件的内容。



8-5.添加远程仓库，点击 New repository



填写远程仓库的名称，描述性信息勾选Public选项。



8-6.克隆一个本地库: $ git clone [git@github.com:githubAccountName/xxx.git](mailto:git@github.com:githubAccountName/xxx.git) (githubAccountName: 指的是你在github上的帐户名； xxx.git : 指的是远程仓库的名字)

8-7. 将本地内容推送到远程仓库： git push origin master

8-9.在本地仓库下，运行命令：$ git remote add origin [git@github.com:yourGitHubName/directory.git](mailto:git@github.com:yourGitHubName/directory.git). 关联本地仓库与远程仓库。

8-10. $ git push -u origin master, 将本地仓库的所有内容推送到远程库