Git 常用命令

git init here -- 创建本地仓库(repository)，将会在文件夹下创建一个 .git 文件夹，.git 文件夹里存储了所有的版本信息、标记等内容

git remote add origin git@github.com:winter1991/helloworld.git

-- 把本地仓库和远程仓库关联起来。如果不执行这个命令的话，每次 push 的时候都需要指定远程服务器的地址

git add -- 从本地仓库增删，结果将会保存到本机的缓存里面

git rm

git commit -m "注释" -- 提交，把本机缓存中的内容提交到本机的 HEAD 里面

git push origin master -- 把本地的 commit(提交) push 到远程服务器上， origin 也就是之前 git remote add origin 那个命令里面的 origin，origin 替代了服务器仓库地址：git push git@github.com:winter1991/helloworld.git master

git pull origin master -- 从远程服务器 pull 新的改动

git status -- 查看状态

git add -A -- 提交全部修改

1. git 配置：

git config --global user.name "xxx" -- 配置用户名，上传本地 repository 到服务器上的时候，在 Github 上会显示这里配置的上传者信息

git config --global user.email "xxx" -- 配置邮箱

git config --list 查看配置列表

配置 sshkey ： 上传代码时使用这个 sshkey 来确认是否有上传权限

1. 创建本地 ssh ： ssh-keygen -t rsa -C "Github 的注册邮箱"

2. 在 Github 中添加这个 sshkey ：

复制 C:\Documents and Settings\Administrator\.ssh\id\_rsa.pub 文件中的内容；

登录 Github --> Account Setting --> SSH-KEY --> Add SSH-KEY --> 粘贴id\_rsa.pub中的内容；

3. 验证： ssh -T git@github.com

出现 Hi xxx! You've successfully authenticated, but GitHub does not provide shell access. 说明配置成功，可以连接上 Github

2. 建立仓库 repository ：

git init here -- 创建本地仓库

git remote add origin git@github.com:用户名/仓库名.git

-- 把本地仓库和远程仓库关联起来， 如果不执行这个命令的话，每次 push 的时候都需要指定远程服务器的地址

3. 从远程仓库中下载新的改动：

git pull origin master

4. 提交本地修改到远程仓库中：

git add

git add -A -- 将改动添加到本地仓库中

git rm xxx -- 从本地仓库中删除指定文件

git rm -r xxx -- 从本地仓库中删除指定文件夹

git commit -m "注释" -- 把本机缓存中的内容提交到本机的 HEAD 里面

git push origin master -- 把本地的 commit push 到远程仓库中

5. 使用 .gitignore 文件忽略指定的内容：

1. 在本地仓库根目录创建 .gitignore 文件。Win7 下不能直接创建，可以创建 ".gitignore." 文件，后面的标点自动被忽略；

2. 过滤文件和文件夹： [Tt]emp/ 过滤 Temp\temp 文件夹； \*.suo 过滤 .suo 文件；

3. 不过滤文件和文件夹： !\*.c