1. **git在windows上的安装过程**

1.安装windows版本的git可执行程序

2.建议安装TortoiseGit（海龟git），方便用界面形提交、管理、查看、比较版本等操作。

3.安装完成后在git Bash上输入以下命令配置：

git config --global user.name "XXXXyour name"

git config --global user.email "XXXXyour email"

**二、创建SSH及SSH配置（若已配置请忽略）**

1.首先查看用户目录是否有.ssh目录：有则直接下命令cat ~/.ssh/id\_rsa.pub，复制其内容，查看3步骤；没有请看2步骤。

2.使用ssh-keygen -t rsa -C "XXXyour email" -b 4096产生key，然后再输入命令cat ~/.ssh/id\_rsa.pub，复制输出内容

3.登录GitHub，到个人图标上点击并选择“Settings”，再选择“SSH Keys”图标，在右边的title中粘贴1,2中的key内容，点击“Add Key”按钮添加key

**三、git常规步骤和命令**

1.拉取远程的git源码：git clone 路径

如下载拓扑逻辑源码：git clone git@git.biggersoft.com:TPFW\_TEAM/UTTOPOD.git

2.对源码修改管理：

（1）创建分支（可以直接在master上进行，但好的习惯是添加分支，待修改好后提交合并到master）

git branch 分支名称

（2）切换分支：git checkout 分支名称

（3）a.添加本地仓库修改的文件或目录到暂存区：git add 文件或者目录

b.删除本地仓库文件：git rm 文件名

删除本地仓库文件夹: git rm -r 文件夹

（4）修改提交仓库：git commit -m "XXX修改说明"

（5）修改合并master：

a.就在master修改，进入（6）步骤

b.在其他分支修改：先切到master分支，然后合并到master。

git checkout master

git merge 其他分支名

（6）推送到远程仓库：git push origin 分支名称 (如:git push origin master)

（7）删除分支：git branch -d 分支名称

（8）查看本地分支：git branch

3.撤销修改

（1）尚未提交到暂存区，丢弃本地修改：git checkout 文件

（2）添加到暂存区，但未提交：git reset head 文件 此时暂存区的修改被撤销掉，重放回工作区，可在修改添加到暂存区或者工作区用（1）方法丢弃修改

4.版本回退步骤

（1）查看提交的版本记录：git log,可以添加参数 --pretty=oneline让每个版本记录只输出一行:git log --pretty=oneline

（2）确定回退版本，当前是HEAD，上一个版本为HEAD^,上上个版本为HEAD^^，再网上为HEAD^n。git log输出内容中commit之后的一串输出也表示版本号。

（3）退版本命令：git reset --hard 版本 （注意记录最新版本号）

eg：git reset --hard HEAD^

git reset --hard 2999616711a5567b977bfc15fe3ec26f46be0b1d

（4）回退版本后想回到新版本：

git reset --hard 原始最新版本号

5.创建本地仓库并提交到远程仓库步骤:

（1）创建本地仓库：git init（注意：在需要的总目录使用该命令）

（2）git add .

（3）git commit -m "注解"

（4）在远程仓库建立仓库路径，如果文件大选ssh建立，文件小勇http

（5）连接远程仓库：

Http：git remote add origin http://XX开头的远程仓库路径

SSH： git remote set-url origin git@git.XXXX开头的远程仓库路径

（6）把本地仓库推送到远程仓库：git push -u origin 分支名称

说明：git push (-u) origin 分支名称 有无(-u)的区别是：

远程仓库是空内容，加上使用-u

远程仓库有内容之后，去掉-u

6.关于远程分支拉取与建立：

（1）查看远程分支： git branch –r

（2）拉取远程分支：

a：git checkout –b 本地分支名 origin/远程分支名

该命令会先在本地建立分支拉取远程分支，并自动切换到该本地分支

b：git fetch origin 远程分支名:本地分支名

该命令会在本地建立分支名，并拉取远程分支，但不会自动切换到本地分支，需要手动checkout到本地分支

（3）远程分支建立：

a：先建立一个本地分支： git branch 本地分支名

b：git push origin 本地分支名:远程分支名

（4）远程分支的删除：

a：git push origin :远程分支名 推送一个空分支到远程分支，相当于删除远程分支

b：git push origin -–delete 远程分支名 删除指定的远程分支

（5） git删除远程分支的文件

a．仅仅删除远程分支文件，不删除本地的文件；

删除远程文件filename

git rm --cached filename

git commit -m "delete remote file filename "

git push -u origin master(此处是当前分支的名字)

删除远程文件夹directoryname

git rm -r --cached directoryname

git commit -m "delete remote directory directoryname "

git push -u origin master(此处是当前分支的名字)

b. 删除本地文件与远程分支文件

删除远程文件filename

git rm filename

git commit -m "delete file filename "

git push -u origin master(此处是当前分支的名字)

删除远程文件夹directoryname

git rm -r directoryname

git commit -m "delete directory directoryname "

git push -u origin master(此处是当前分支的名字)

7.其他常用命令说明：

git status 查看本地仓库源码文件状态，可显示有修改记录和未控制文件记录

git diff 查看所有本地修改文件差异

git difftool 调用图形化差异比较工具

git pull 拉取远程版本库的提交，即用此命令更新源码到最新版本。默认的远程分支名和本地分支名是一致的。

git pull origin 远程分支名：本地分支名 拉去远程仓库的指定分支文件到本地仓库的指定分支。

8. 修改本地分支的名字

git branch -m oldname newname

9. 查看暂存区的文件 git ls-files

查看最新提交的说明git show –stat

查看所有提交说明 git log

10. 发现前一次的提交有错误

在工作区修改错误->git add filename->git commit –amend -m “description” 上一次的错误提交就会被这一次的覆盖。

发现写错的不是上一次的提交而是倒数第二个

第一步：把当前commit rebase到指定的commit上

[git rebase -i HEAD^^ --等同于 git rebase -i HEAD~2]

[^ 的用法： ^ 表示把commit往回偏移，偏移量就是 ^ 的个数。]

[~ 的用法： ~ 后面跟一个数，表示往回偏移，偏移量就是 ~ 后面跟着的数。]

第二步：执行完上一步之后，会进入一个编辑页面，点 i 进入编辑状态，修改pick为edit，然后退出

第三步：修改完要修改的文件之后，提交

git add laugh.txt

git commit --amend

git rebase –continue

想丢弃刚才的提交 git reset --hard HEAD^

git reset --soft HEAD^

git reset HEAD^

–hard：重置位置的同时，清空工作目录的所有改动；

–soft：重置位置的同时，保留工作目录和暂存区的内容，并把重置 HEAD 的位置所导致的新的文件差异放进暂存区。

–mixed（默认）：重置位置的同时，保留工作目录的内容，并清空暂存区。