**一、基础配置：**

**2.设定用户与Email：**

git config --global user.name "你的名字"

git config --global user.email "你的邮箱"

**2.指定你的编辑器，你的Diff工具：**

git config --global core.editor vim

git config --global merge.tool vimdiff

**二、新建GIT库：**

**1.初始化空库：**

git init

**2.从远程复制GIT库**

git clone http远程地址 本地文件夹名

**三、操作GIT库:**

**1. 把文件添加到仓库**

git add 文件名.扩展名

git rm删掉

**2.** **把文件提交到仓库**

git commit -m "提交的信息"

git commit -m后面输入的是本次提交的说明

git commit命令执行成功后会告诉你更改内容

git commit –a提交所有改动。不用add

**3.查看log**

git log //显示全部

git log -1 //只显示一条log

git log --pretty=oneline //单行显示

**4.显示工作目录树的状态,查看改动**

Git status

**5.保存并修改提交**

git diff

**5.撤销修改**

git checkout -- readme.txt意思就是，把readme.txt文件在工作区的修改全部撤销，这里有两种情况：

一种是readme.txt自修改后还没有被放到暂存区，现在，撤销修改就回到和版本库一模一样的状态；

一种是readme.txt已经添加到暂存区后，又作了修改，现在，撤销修改就回到添加到暂存区后的状态。

总之，就是让这个文件回到最近一次git commit或git add时的状态。

git reset HEAD file可以把暂存区的修改撤销掉（unstage），重新放回工作区

**四、版本回退**

1.返回上一个commit

git reset –-hard HEAD^

2.撤消回退

git reset –hare 版本号（版本号要记住或者通过下面的git reflog查询）

3.命令记录

git reflog用来记录你的每一次命令

**四、操作分支:**

**1.新建分支**

git branch 新分支 父分支

**2.切换分支**

git checkout 分支

**3.查看信息**

git status

**4.删除分支**

git branch –d 分支名

**5.合并到主分支**

git checkout master

git rebase 分支名(希望变基到哪条分支的末梢，就使用哪条分支名称做参数)



**五、处理发布**

**1.添加标签**

git tag 标签的名称 希望打标签的点

**2.查看标签**

git tag

**七、远程库**

**1、关联远程库**

git remote add origin ssh地址(要先设置ssh key)

git remote add 远程库名 https地址

**2、推送到远程库**

git push -u 上面定义的远程库名 要推送的分支

由于远程库是空的，我们第一次推送master分支时，加上了-u参数，Git不但会把本地的master分支内容推送的远程新的master分支，还会把本地的master分支和远程的master分支关联起来，在以后的推送或者拉取时就可以简化命令。

只要本地作了提交，就可以通过命令：

git push origin master

**3.从远程复制GIT库**

git clone http远程地址 本地文件夹名

**工作区（Working Directory）**

就是你在电脑里能看到的目录

**版本库（Repository）**

工作区有一个隐藏目录.git，这个不算工作区，而是Git的版本库。

Git的版本库里存了很多东西，其中最重要的就是称为stage（或者叫index）的暂存区，还有Git为我们自动创建的第一个分支master，以及指向master的一个指针叫HEAD。



前面讲了我们把文件往Git版本库里添加的时候，是分两步执行的：

第一步是用git add把文件添加进去，实际上就是把文件修改添加到暂存区；

第二步是用git commit提交更改，实际上就是把暂存区的所有内容提交到当前分支。

因为我们创建Git版本库时，Git自动为我们创建了唯一一个master分支，所以，现在，git commit就是往master分支上提交更改。

初始化为仅包含空README.MD文件的空项目：

git init

git add README.md

git commit -m "first commit"

git remote add origin git@code.csdn.net:alwcelcs/gittest.git

git push -u origin master

用已经存在的项目初始化项目：

git remote add origin git@code.csdn.net:alwcelcs/gittest.git

git push -u origin master