1.SVN是集中式版本控制工具,版本库集中放在中央服务器;Git是分布式版本控制系统,没有中央服务器,每个电脑都是完整的版本库,只需要把各自的修改推送给对方.

2.Git是分布式版本控制,所以需要配置用户名和邮箱作为标识

3.操作

创建版本库,Git管理目录下的所有文件,包括文件的修改,删除,能够追踪历史,文件还原,

操作;进入目录,Git init 变为管理的仓库

把文件添加到版本库,只能跟踪问问文件的改动,无法跟踪视屏,图片二进制文件的改动,

Git add 文件

Git commit -m " 提交的注释"提交

Git status 查看 当前是否含有未提交的文件

修改文件后,再次执行, 查看到已经发生改变

git diff 文件 : 查看修改的部分

之后提交修改操作和直接提交是一样的.

Git log 查看提交的记录

Git log --pretty=online 一行查看提交信息.

回退版本

Git reset --hard HEAD^ 上个版本

Git reset --hard HEAD^^ 上两个版本

GIT reset --hard HEAD~100 上100个版本

回复之前的最新的版本:

Git reset --hard 版本号

查看所有的版本

Git reflog

4.工作区 .git除外的文件

版本库 :.git隐藏文件,包括stage暂存区,Git自动创建的第一个分支master,以及指向master的一个指针.

add命令: 把文件添加到缓存区,

commit 命令 : 将暂存区的所有内容提交到当前分支上.

5.git撤销修改和删除文件操作

撤销修改 : git cheackout -- 文件 如果没有--,变成创建分支的命令

只提交到暂存区,不存在add, checkout 撤销暂存区的,

有一部分未commit,有一部分未add,只撤销未add的

删除:原理同上,

6远程仓库,

先注册GitHub账号要设置 我的账户 1412796229@qq.com liutao159159

在Git上创建完仓库后,会有相应的命令提示,创建新的仓库或者将本地仓库提交到Github上

Git push -u origin master 将本地仓库提交到github,实际上是吧当前分支master推送到远程,

-u 第一次推送master分支时,Git不但会吧本地的master分支内容推送到远程新的master分支,还会把本地master分支和远程的master分支关联起来.

以后直接使用Git push origin master 直接提交修改的文件

下载仓库:

7.创建切换分支

Git checkout - b dev

git branch 查看当前分支

-b 表示创建并切换

git checkout dev 切换分支

git merge 分支名称 合并分支

如果出现冲突 直接再次编辑 再次提交

8.推送到远程仓库时,首先要拉去,要解决无法pull的错误 后面加参数大概是允许什么操作

Git pull origin master --allow- \*\*-\*\*

然后在推送文件 git push origin master

直接remote add 远程仓库时,必须初始化本地仓库

clone仓库时,不需要init ,会直接生成

9.分支管理策略,通常合并分支时,会使用fast forward 模式,在这种模式下,删除分支,会丢掉分支信息,使用带参数来禁用 fast-forward模式,

git stash 将当前的工作现场隐藏起来,

git stash list 查看所有

git stash drop 删除一条

git stash pop 回复文件并且删除

git stash apply 回复 stash

10.多人协作

git remote 查看远程库的信息

git remove -v 查看远程库的详细信息.

11.eclipse 使用Git 同上,