Git学习

Git init 创建仓库

Git add 添加内容

Git commit -m “备注” 提交添加的内容

Git log 查看提交的版本信息

Git log –pretty=oneline 单行输出版本信息

Git reset --hard HEAD^ 回退到上一个版本

Git reset --hard 123456 回退到指定版本号的版本

Git reflog git的操做（所有命令）记录

Git{

工作区->版本库[

暂存区，

分支（master分支，其他分支）]

}

Git add 只是将工作区的提交到暂存区，git commit是将暂存区的提交到分支上

Git status 查看工作区的状态

**在中，文件状态是一个非常重要的概念，不同的状态对应不同的操作。**

**在未进行操作之前，存在三种状态： ， （没有直接）及 。（了但是没有）**

**就是你必须把某些文件放到工作目录中，但又不能提交它们，比如保存了数据库密码的配置文件啦，等等。那么看着也烦怎么处理呢？好在考虑到了大家的感受，这个问题解决起来也很简单，在工作区的根目录下创建一个特殊的.文件，然后把要忽略的文件名填进去，就会自动忽略这些文件。**

Git撤销修改：{

一：撤销工作区的修改给git checkout -- 文件名称

二：撤销暂存区的修改git reset HEAD 文件名称

}

git删除：{

一：直接将工作区的删除了rm+文件名，

【1:确定删除（git rm 文件名）+（git commit -m “备注”）】

【2：不想删除了git checkout -- 文件名】

二：直接使用git rm 文件名，此时工作区暂存区都已删除

【1：确认删除git commit -m “备注”】

【2：不想删除git checkout HEAD -- 文件名】

}

生成ssh key：ssh-keygen -t rsa -C [yxj9721@163.com](mailto:yxj9721@163.com)，中间有三次询问，

第一次是询问ssh-key的存放路径，

第二次是询问是否需要密码登录，

第三次确认上次的密码（没有设置密码直接enter键）

将本地仓库和远程关联：git remote add origin(远程仓库别名) +远程地址

查看远程仓库名称：git remote

删除远程映射：Git remote rm 远程仓库名称

拉取远程仓库文件{

git pull origin master --allow-unrelated-histories

git pull origin master

}

向远程仓库推送 git push -u origin master

克隆远程仓库：git clone + 远程仓库地址

添加分支并切换：git checkout -b 分支名称

添加分支：git branch 分支名称

切换分支：git checkout 分支名称

Git merge 分支名称这是默认的Fast forward模式，删除分支后，会丢掉分支信息

删除分支：git branch -d 分支名称

强行删除分支：git branch -D 分支名

禁用此模式的命令是：git merge --no--ff -m “备注信息” + 分支名称

Git stash 储藏工作区的工作

Git stash list 查看储藏的任务

Git stash pop（恢复和删除）=git stash apply（恢复工作区） + git stash drop（删除stash）

并不是一定要把本地分支往远程推送，那么，哪些分支需要推送，哪些不需要呢？

* master分支是主分支，因此要时刻与远程同步；
* dev分支是开发分支，团队所有成员都需要在上面工作，所以也需要与远程同步

从远程克隆仓库到本地，默认只能看到master分支，

1. 如果要看其他分支，需要执行这个命令：

git checkout -b dev origin/dev

1. 本地分支和远程分支的链接：

git branch --set-upstream dev origin/dev

1. 向远程推送代码：git push origin dev（分支名称）
2. 拉取远程分支的代码：git pull