关联远程仓库

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1.git config --global user.name 用户名 \*本地配置用户名\*

2.git config --global user.email 邮箱 \*本地配置邮箱\*

3.ssh-keygen -t rsa -C 邮箱 \*生成ssh key\*

4.clip < ~/.ssh/id\_rsa.pub \*复制ssh key\* ->到github官网粘贴ssh key关联

5.ssh -T git@github.com \*测试连接是否成功\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

上传文件

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1.git init \*初始化仓库\*

2.git add . \*上传此文件\*

3.git commit -m "更新的信息" \*来提交\*

4.git push -u origin master \*把本地库的所有内容推送到远程库上\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

创建和切换分支

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1.gitk \*查看项目源码在各个时间点上的代码提交情况（主要查看SHA1 ID）\*

2.git branch 分支名 SHA1ID \*创建分支\*

3.git branch \*查看分支\*

4.git branch -a \*查看所有分支\*

5.git checkout 分支名 \*切换分支\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

一、 Git 常用命令速查

git branch 查看本地所有分支

git status 查看当前状态?

git commit 提交?

git branch -a 查看所有的分支

git branch -r 查看远程所有分支

git commit -am "init" 提交并且加注释?

git remote add origin git@192.168.1.119:ndshow

git push origin master 将文件给推到服务器上?

git remote show origin 显示远程库origin里的资源?

git push origin master:develop

git push origin master:hb-dev 将本地库与服务器上的库进行关联?

git checkout --track origin/dev 切换到远程dev分支

git branch -D master develop 删除本地库develop

git checkout -b dev 建立一个新的本地分支dev

git merge origin/dev 将分支dev与当前分支进行合并

git checkout dev 切换到本地dev分支

git remote show 查看远程库

git add .

git rm 文件名(包括路径) 从git中删除指定文件

git clone git://github.com/schacon/grit.git 从服务器上将代码给拉下来

git config --list 看所有用户

git ls-files 看已经被提交的

git rm [file name] 删除一个文件

git commit -a 提交当前repos的所有的改变

git add [file name] 添加一个文件到git index

git commit -v 当你用－v参数的时候可以看commit的差异

git commit -m "This is the message describing the commit" 添加commit信息

git commit -a -a是代表add，把所有的change加到git index里然后再commit

git commit -a -v 一般提交命令

git log 看你commit的日志

git diff 查看尚未暂存的更新

git rm a.a 移除文件(从暂存区和工作区中删除)

git rm --cached a.a 移除文件(只从暂存区中删除)

git commit -m "remove" 移除文件(从Git中删除)

git rm -f a.a 强行移除修改后文件(从暂存区和工作区中删除)

git diff --cached 或 $ git diff --staged 查看尚未提交的更新

git stash push 将文件给push到一个临时空间中

git stash pop 将文件从临时空间pop下来

---------------------------------------------------------

git remote add origin git@github.com:username/Hello-World.git

git push origin master 将本地项目给提交到服务器中

-----------------------------------------------------------

git pull 本地与服务器端同步

-----------------------------------------------------------------

git push (远程仓库名) (分支名) 将本地分支推送到服务器上去。

git push origin serverfix:awesomebranch

------------------------------------------------------------------

git fetch 相当于是从远程获取最新版本到本地，不会自动merge

git commit -a -m "log\_message" (-a是提交所有改动，-m是加入log信息) 本地修改同步至服务器端 ：

git branch branch\_0.1 master 从主分支master创建branch\_0.1分支

git branch -m branch\_0.1 branch\_1.0 将branch\_0.1重命名为branch\_1.0

git checkout branch\_1.0/master 切换到branch\_1.0/master分支

du -hs

git branch 删除远程branch

git push origin :branch\_remote\_name

git branch -r -d branch\_remote\_name

-----------------------------------------------------------

初始化版本库，并提交到远程服务器端

mkdir WebApp

cd WebApp

git init 本地初始化

touch README

git add README 添加文件

git commit -m 'first commit'

git remote add origin git@github.com:daixu/WebApp.git