**本地项目上传github**

1. 建立git仓库

cd 到不让你弟项目根目录,执行git命令,此命令会在当前目录下创建一个.git文件夹

git init

1. 将项目所有文件添加到仓库中

git add .

这个命令会把当前路径下的所有文件，添加到待上传的文件列表中。 如果想添加某个特定的文件，只需把.换成特定的文件名即可

1. 将add 的文件commit 到仓库

git commit –m “注释语句”

1. 去github上创建自己的Repository，点击NewRepository 点击Create repository，就会进入到类似下面的一个页面，拿到创建的仓库的https地址
2. 将本地的仓库关联到github上

git remote add origin <https://自己的仓库地址>

1. 上传代码到github远程仓库

git push –u origin master

执行完后，如果没有异常，等待执行完就上传成功了，中间可能会让你输入Username和Password，你只要输入github的账号和密码就行了

**拉取git项目到本地**

1.打开终端，cd到自己想要存放项目的文件夹

cd 文件夹

2.输入git clone url ，url为你要拉取的项目地址

git clone url

3.项目拉取成功

**如果改动后的项目要上传到github**

1.git add 你改动后的文件,如果想要全部上传 git add .

git add .

2.执行git commit -m “要添加的注释”

git commit -m "注释"

3.git push 上传，可能会让你输入github账号和密码按提示输入即可

git push

**修改上传的用户名和密码**

1. git config --global user.name "Your Name"
2. git config --global user.email you@example.com

全局的通过vim ~/.gitconfig来查看

1. git config user.name "Your Name"
2. git config user.email [you@example.com](mailto:you@example.com)

局部的通过当前路径下的 .git/config文件来查看

也可以修改提交的用户名和Email：

1. git commit --amend --author='Your Name <you@example.com>'

git commit已提交的Author信息可以通过git log查看

1. git log

git 常见命令

git add <filename>

git commit -m “描述”

git status

git diff <filename>

git clone <仓库地址>

分支:

查看分支: git branch

创建分支: git branch <分支name>

切换分支: git checkout <分支name>

创建+切换分支: git checkout -b <分支name>

合并某分支到当前分支: git merge <分支name>

删除分支: git branch -d <分支name>

删除github 上的分支: git push origin :dmf

**例如:**

github上已经有master分支 和dev分支

在本地

git checkout -b dev 新建并切换到本地dev分支

git pull origin dev 本地分支与远程分支相关联

在本地新建分支并推送到远程

git checkout -b test

git push origin test   这样远程仓库中也就创建了一个test分支

丢弃本地修改，当本地修改了一些文件或者新增文件，但是想删除

git checkout . // 本地所有修改的，没有提交的，都返回原来的状态

git stash // 把本地所有修改没有提交的暂存到stash里面，可用 git stash pop 恢复

git reset --hard HASH // 返回到某个节点，不保留修改

git reset --soft HASH // 返回到某个节点，保留修改

git clean -df // 返回到某个节点

git clean 参数

-n 显示 将要 删除的文件 和 目录

-f 删除文件

-df 删除文件 和 目录

也可以使用

git checkout . && git clean -xdf