Git命令行

克隆Clone

git clone git@github.com:CoolInc/someniceapp.git

添加修改内容到暂存区Stage

提交所有变化

git add -A

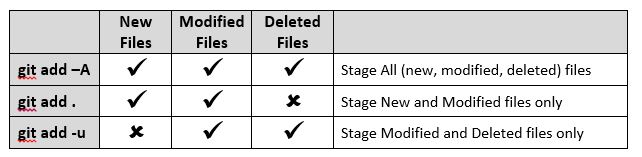
提交被修改(modified)和被删除(deleted)文件，不包括新文件(new)

git add -u

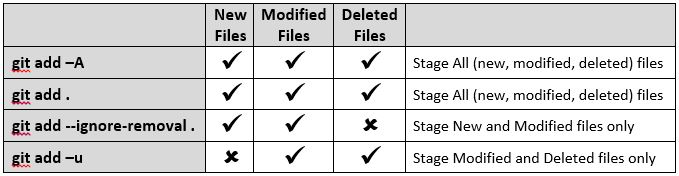
提交新文件(new)和被修改(modified)文件，不包括被删除(deleted)文件

git add .

Git 1.x



Git 2.x



提交Commit

git commit -m "I made some super cool function"

推送分支Push

git push <远程主机名> <本地分支名>:<远程分支名>

例：将本地master分支推送到远程master分支

git push origin master:master

推送本地创建的新分支，同时在远程创建上游跟踪分支

git push --set-upstream origin new-cool-function

创建分支Branch

git branch new-cool-function

切换分支Checkout

git checkout new-cool-function

提交到GitHub时输入用户名和密码

git config --local credential.helper ""

显示git的当前代码库的用户名

git config user.name

显示git的当前代码库的用户邮箱

git config user.email

修改git的用户名（与提交时的GitHub用户名不同，这个会显示在提交记录中）

git config user.name "Git User"

修改git的用户邮箱

git config user.email "gituser@mymail.com"

显示git的全局用户名

git config --global user.name

显示git的全局用户邮箱

git config --global user.email

修改git的全局用户名

git config --global user.name "Git User"

修改git的用户邮箱

git config --global user.email "gituser@mymail.com"

查看提交日志Log

git log --pretty=oneline

查看分支列表和当前分支

git branch

创建并切换到新分支

git checkout -b issue345

为某次提交创建新分支

git checkout -b new\_branch 27b06870957e44d5b02606af668c9a120df4b7e0

合并分支Merge（将issue345分支合并到master）

git checkout master

git merge issue345

打标签Tag

git tag -a <tagname> -m "message"

为过去的提交补打标签

git tag -a v0.1 27b06870957e44d5b02606af668c9a120df4b7e0

推送标签到远程代码库

git push origin --tags

查看所有标签

git tag

显示标签详细信息

git show v0.1

按标签切换分支（新建分支0.1，签出v0.1标签的内容，并切换到该分支）

git checkout -b 0.1 v0.1

按标签切换分支（签出v0.1标签的内容，但不新建分支）

git checkout v0.1

**注意：尽量不要存在名称相同的分支与标签，如果存在名称相同的分支和标签，在使用checkout命令时会无法分辨到底是要checkout分支还是标签。**

明示checkout标签

git checkout -b test refs/tags/v0.1

明示checkout分支

git checkout refs/heads/v0.1