# **[Git学习笔记(SourceTree克隆、提交、推送、拉取等)](https://www.cnblogs.com/gamedaybyday/p/6373312.html)**

学习一下sourcetree使用git

目录

一 克隆Clone

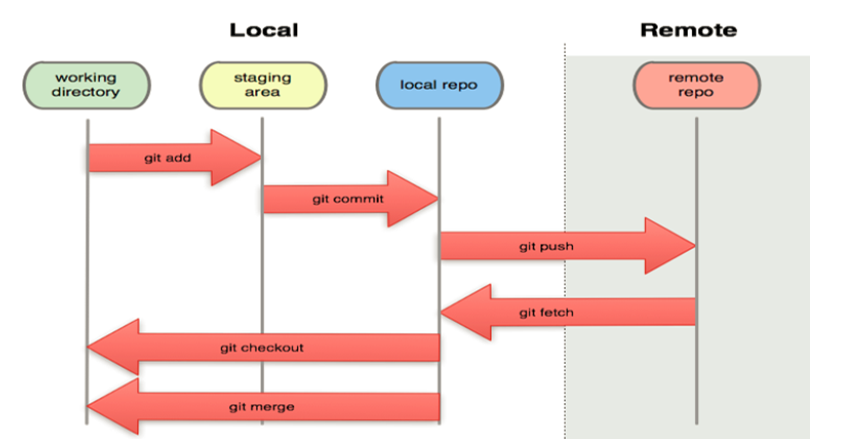
二 提交Commit和推送Push

三 拉取pull和获取fetch

四 版本回退reset

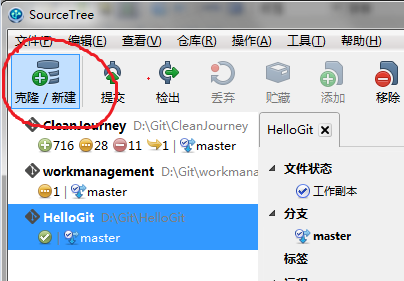
五 检出checkout

六 标签Tag

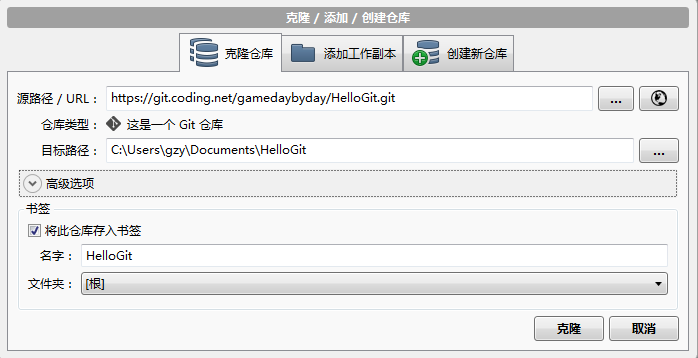
****

**一 从远程库克隆Clone**

**Clone就是将远程库的代码拷贝到本地。**



填写远程和本地项目路径，点击“克隆“。这样就会将服务器上项目代码克隆到本地了。



git -c diff.mnemonicprefix=false -c core.quotepath=false clone --recursive https://git.coding.net/gamedaybyday/HelloGit.git D:\Git\HelloGit

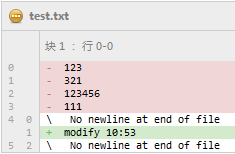
Cloning into 'D:\Git\HelloGit'...

**二 提交Commit和推送Push**

**commit将工作空间修改提交到本地库。**

**push将本地库修改提交到远程库。**

新建一个test.txt来测试，任意改点什么。在文件状态处可查看，红色为删除，蓝色为增加部分。



 将修改后文件由未暂存文件，勾选到已暂存文件。

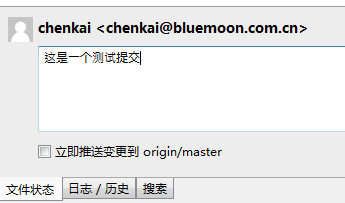


选择提交



 添加修改日志。

这里如果勾选“立即推送变更”则会同时执行commit和push。

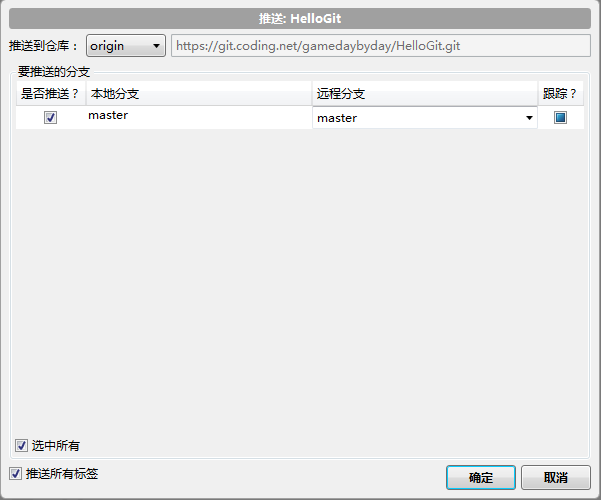


git -c diff.mnemonicprefix=false -c core.quotepath=false commit -q -F C:\Users\gzy\AppData\Local\Temp\ofkmvj0p.tft

 这时，修改的代码提交到了本地仓库。sourcetree会提示有修改代码尚未推送到远程库。



 选择推送。将本地仓库推送push到远程库。



[?](https://www.cnblogs.com/gamedaybyday/p/6373312.html)

|  |  |
| --- | --- |
| 1 | git -c diff.mnemonicprefix=false -c core.quotepath=false push -v --tags origin master:master |

**三 拉取pull和获取fetch**

**pull 从远程拉取最新版本 到本地  自动合并 merge**

**fetch   从远程获取最新版本 到本地   不会自动合并 merge**



虚线表示拉取到本地仓库

实现表示拉取到本地仓库，并合并到工作空间

git -c diff.mnemonicprefix=false -c core.quotepath=false fetch origin

From https://git.coding.net/gamedaybyday/HelloGit

9664a7c..42c8f2e master -> origin/master

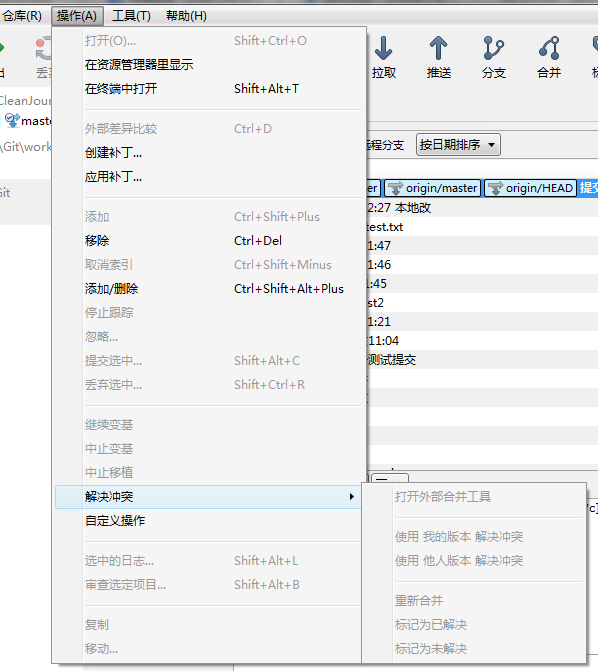
git -c diff.mnemonicprefix=false -c core.quotepath=false fetch origin

git -c diff.mnemonicprefix=false -c core.quotepath=false pull origin master

From https://git.coding.net/gamedaybyday/HelloGit

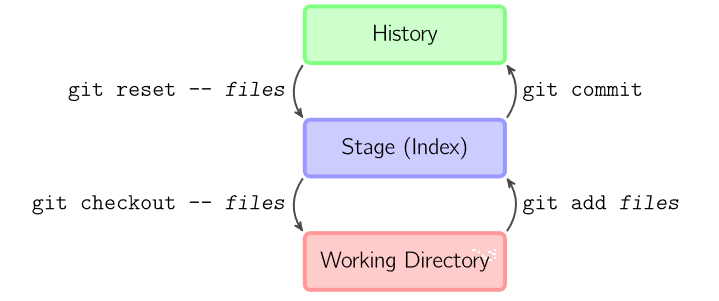
\* branch master -> FETCH\_HEAD

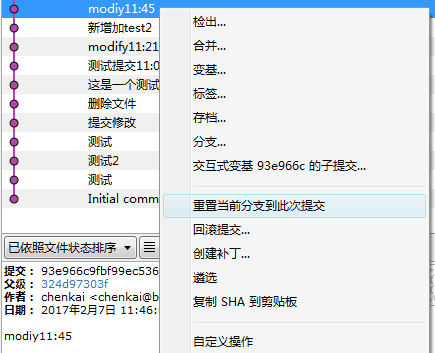
假如有冲突，使用解决冲突。

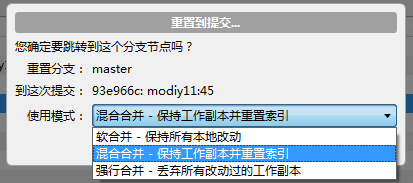


**四 版本回退reset**

**reset 将当前版本回退到选定的版本**

****





git -c diff.mnemonicprefix=false -c core.quotepath=false reset -q --mixed 657cf95e9925724ea54f81c2dc03f09ab025d571

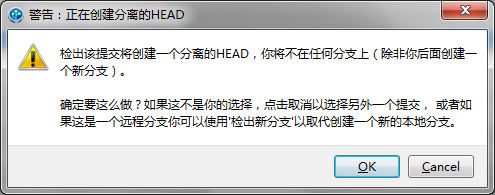
**五 检出checkout**

**checkout 创建分支，切换分支。**

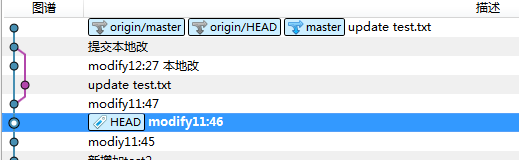
**checkout时，如果只有当前分支，则会新建一个分支。**

**如果是两个分支间checkout，则会切换分支。**

IMG_270



将项目从最新checkout到modify11:46，这时HEAD指向到modify11:46这个版本。

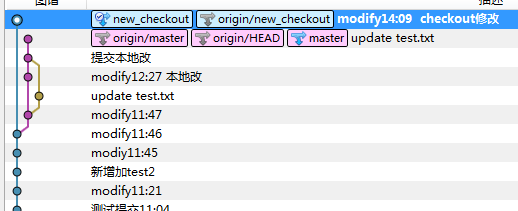


git -c diff.mnemonicprefix=false -c core.quotepath=false checkout 657cf95e9925724ea54f81c2dc03f09ab025d571

Note: checking out '657cf95e9925724ea54f81c2dc03f09ab025d571'.

当修改这个checkout的分支时。可以看到从modify11:46checkout后，修改的提交会忽略到modify11:47以后的提交。

相当于checkout命令 创建了新分支，并切换到了这个分支。



但是这个分支是不存在的。要新建一个分支。

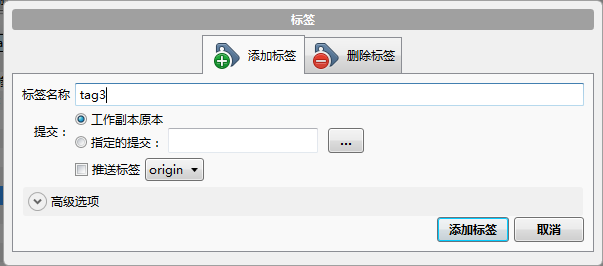




**六 标签Tag**

**tag 就是给这个版本打个标记，方便用于识别查看等操作。**







git -c diff.mnemonicprefix=false -c core.quotepath=false tag -a -m "" tag3

转载自 https://www.cnblogs.com/gamedaybyday/p/6373312.html