1. git status：看是否存在已经修改的版本
2. git diff：查看修改的内容
3. git add：添加到stage中，等待commit到branch中
4. git commit –m “：提交更改
5. git resert --hard HEAD^：表示返回上一版本，如果版本很多的话，就使用HEAD~，如果被清除就得根据Logq去找之前的Hash值，基本上前5位就可以，把HEAD^替换就行
6. 要回退到历史版本就要用git reflog来查看以前的版本
7. 必须经过git add 将修改提交到缓存区，commit才能进行修改
8. git checkout -- file可以丢弃工作区的修改
9. git reset HEAD <file>可以把暂存区的修改撤销掉（unstage），重新放回工作区，

10、git rm file，记得 git commit，如果只是误删除文件，可以通过git checkout filename，来回滚，前提是你已经提交到了版本库，但是只能恢复最新的一次提交，但是会丢失你提交之前的修改内容。

11、git checkout –b dev 创建并切换到分支dev，git checkout master切换到master分支，用git merge dev将分支合并，然后git branch –d dev删除分支



12、git log --graph --pretty=oneline --abbrev-commit看分支的合并情况

13、git merge --no-ff -m "merge with no-ff" dev 禁用fast forward

HEAD指向的版本就是当前版本，因此，Git允许我们在版本的历史之间穿梭，使用命令git reset --hard commit\_id。

穿梭前，用git log可以查看提交历史，以便确定要回退到哪个版本。

要重返未来，用git reflog查看命令历史，以便确定要回到未来的哪个版本。