Git常用命令

创建

克隆一个已有仓库

$ git clone ssh://user@domain.com/repo.git

创建一个新的仓库

$ git init

提交

把修改添加到暂存区(修改、新增)

$ git add **.**

把修改添加到暂存区(修改、删除)

$ git add -u **.**

把修改添加到暂存区(修改、删除、新增)

$ git add -A **.**

把暂存区的内容进行1次提交

$ git commit –m “一些说明”

日志和状态

显示当前工作区中发生变化的文件

$ git status

被跟踪的文件的内容变化（所有）

$ git diff

查看某文件的内容变化

$ git diff HEAD -- <file>

显示从最近到最远的提交日志

$ git log

在一行内显示简略版本信息

$ git log --pretty=oneline

以简略方式查看带分支合并图的日志

$ git log --graph --pretty=oneline --abbrev-commit

版本管理

回退到上一个版本

$ git reset --hard HEAD^

回退到指定版本

$ git reset --hard 1094a

撤销（已add到）暂存区的修改/删除，使修改/删除仅在工作区

$ git reset HEAD <file>/.

恢复误删文件（未add）

$ git checkout -- <file>

删除文件（先手动删除）

$ git add/rm file

$ git commit -m "remove file"

过去的命令和及对应的版本号

$ git reflog

远程仓库

添加远程库关联

$ git remote add origin 远程仓库地址

推送分支到远程库（首次加 –u关联）

$ git push -u origin master

获取远程库的dev分支

$ git checkout –b dev origin/dev

分支管理

列出所有分支

$ git branch

创建名为dev的分支

$ git branch dev

删除名为dev的分支

$ git branch -d dev

强制删除未合并的分支

$ git branch -D feature-vulcan

切换到名为dev的分支

$ git switch dev

创建并切换到名为dev的分支

$ git switch -c dev

合并dev分支到当前分支（快进模式）

$ git merge dev

合并dev分支到当前分支（普通模式）

$ git merge --no-ff -m "提交说明" dev

!!!避免未提交就切换分支，工作区内容的修改将出现在不同的分支。

BUG分支

当前分支（未提交）保存现场

$ git stash

保存的现场列表

$ git stash list

恢复指定的现场（不删除）

$ git stash apply stash@{0}

恢复现场（删除）

$ git stash pop

删除现场(已恢复)

$ git stash drop

复制其他分支的版本到当前分支

$ git cherry-pick <缩略版本号>

多人协作

查看远程库信息

$ git remote

显示抓取和推送的origin的地址

$ git remote -v

把指定分支上的所有提交推送至远程库

$ git push origin dev

设置本地dev分支和远程dev分支链接

$ git branch --set-upstream-to=origin/dev dev

把远程库中与本地的当前分支对应的分支抓取下来，尝试自动合并

$ git pull

标签

打标签（当前分支最新提交）

$ git tag v1.0

查看标签

$ git tag

为历史版本打标签

$ git tag v0.9 f52c633

查看标签信息

$ git show v1.0

删除标签

$ git tag -d v1.0

把标签名推送到远程

$ git push origin v1.0

把所有标签名推送到远程

$ git push origin --tags

把删除操作推送至远程

$ git push origin :refs/tags/v0.9

杂项

配置用户名和email

$ git config --global user.name "Your Name"

$ git config --global user.email "email@example.com"

更新Git

git update-git-for-windows