

一、配置

1、git config --global core.editor "'D:/InstallSoftWare/Sublime Text 3/sublime\_text.exe' -n -w"

2、git config --global push.default upstream

3、git config --global merge.conflictstyle diff3

4、git config --global user.name "Your Name"

5、git config --global user.email [you@example.com](mailto:you@example.com)

git config --system core.longpaths true 解决clone时超出了 Windows 的路径长度限制的错误

二、基本语法

git init （git 初始化）

git clone （拷贝远程仓库）

git add （git 添加）

git reset [文件] (将add的文件从暂存区删除，工作目录仍然存在)

git commit -m "Commit message" （提交git add的结果到repository）

git commit -a -m "Commit message" (提交repository）

git log -n 1

git status （git 状态）

staging area （暂存区）--- git add 的结果

working directory （工作目录）

repository （仓库） --- git commit 的结果

git diff ID1 ID2

git checkout ID2

git log --graph --oneline master 分支名 直观地查看提交历史记录

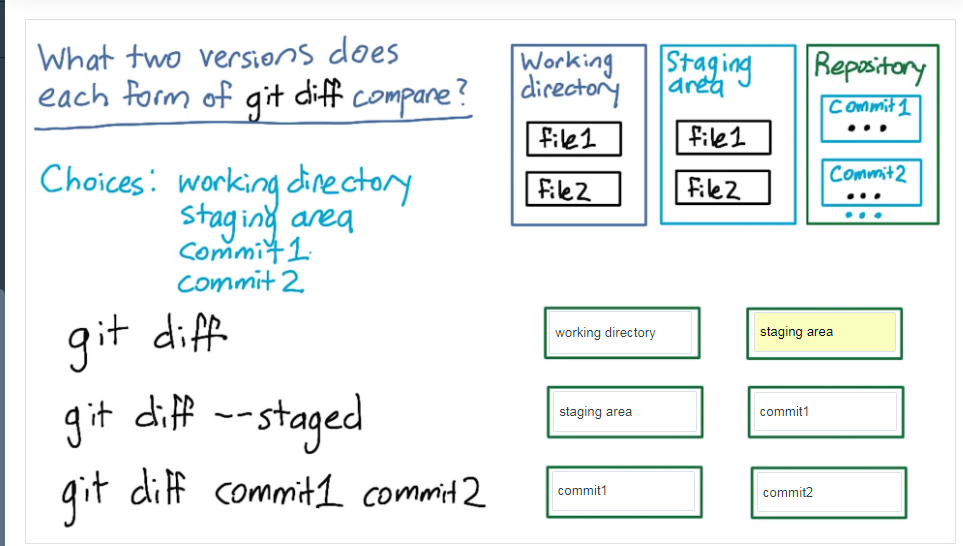
三、深入理解

1、git diff 查看工作目录，暂存区，仓库文件的对比

git diff （默认，比较工作目录，暂存区 文件的不同）

git diff --staged（比较暂存区,仓库最新（HEAD）commit 文件的不同）

git diff commit1 commit2（比较仓库不同commit的文件的不同）



2、分支

git branch 查看所有分支

git branch fix\_bug\_1 创建分支

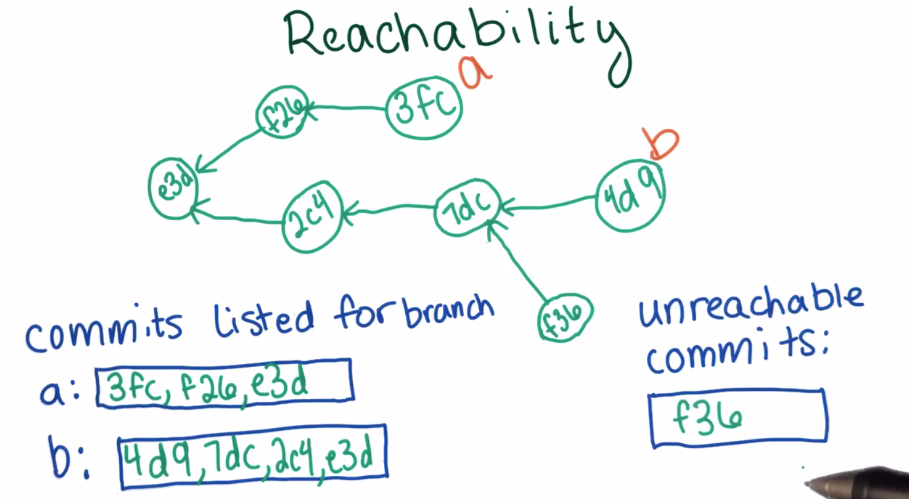
git checkout -b fix\_bug\_2 创建并检出分支fix\_bug\_2

git checkout fix\_bug\_1 切换分支

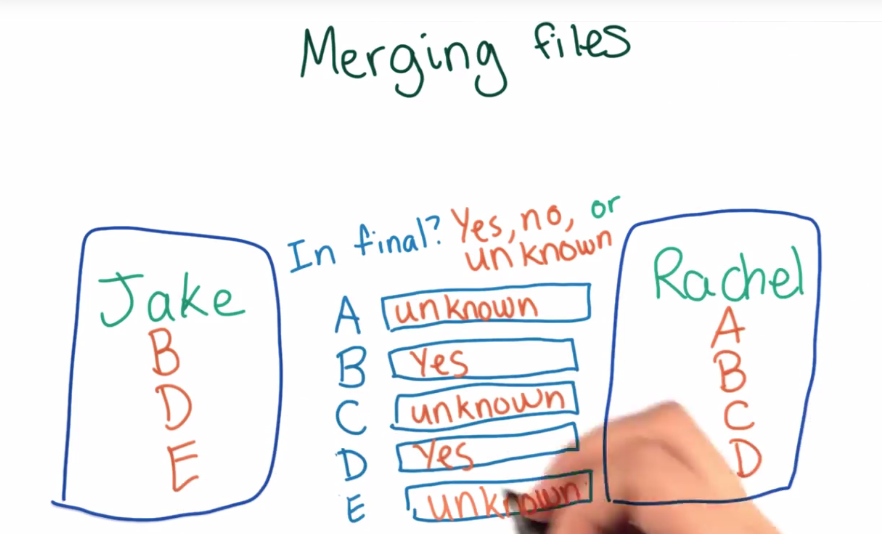
3、可访问性 reachability

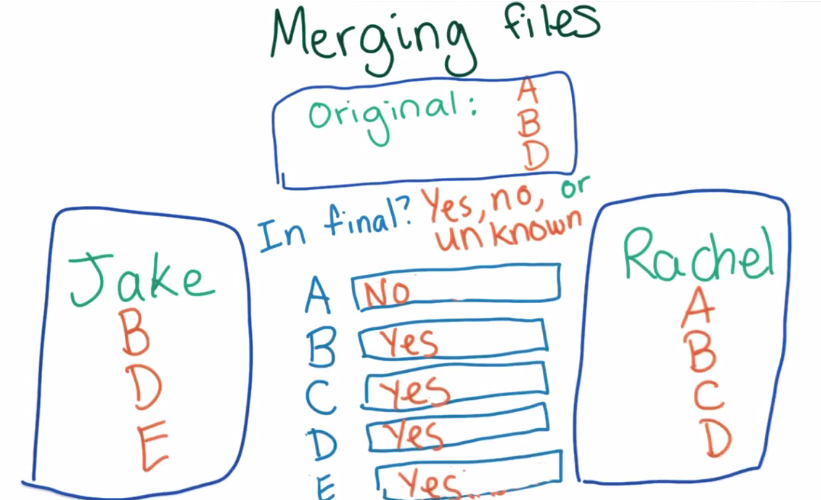
每个commit都存在一个执行其父commit的指针，commit不关心分支，只有分支会存储分支的位置

Git log是以指定的commit或者指定分支的的最新commit为起点，向前跟踪，直至commit没有父commit,因此存在不可达的commit



3、合并





如果删除某个分支并导致无法从现有的分支访问一些提交，则在 Git 的垃圾收集过程运行之前，仍可继续通过提交 ID 访问这些提交。除非你主动关闭此过程，否则它将不时自动运行。也可以通过命令 git gc 手动运行此过程。

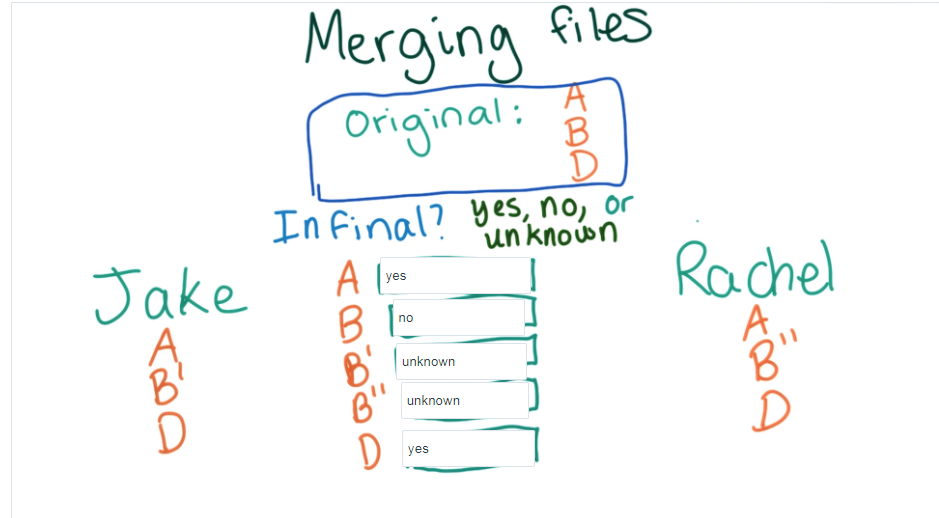
git merge 还将在合并的版本中包含当前检出的分支

* 如果检出了 branch1，并且运行 git merge branch2 branch3，则合并的版本会将 branch1 以及 branch2 和 branch3 组合起来. 如果想将 branch2 合并到 branch1 中，只需键入 git checkout branch1，然后键入 git merge branch2 即可。键入 git merge branch1 branch2 的唯一原因是，它能帮助你对要合并的分支更加心中有数。

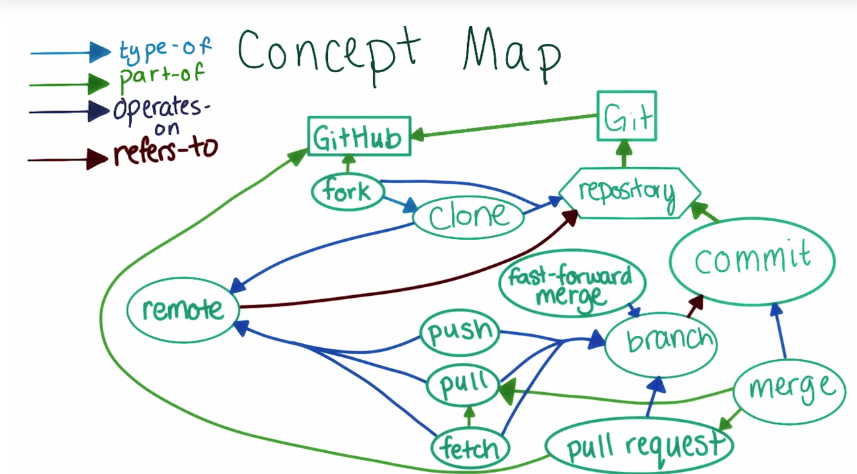
由于是合并两个分支，因此在命令行中按何顺序键入分支并不重要。关键是要记住，git merge 始终将所有指定的分支合并到当前检出的分支中，并为该分支新建一个提交。

1. 运行 git merge --abort，将文件恢复到你开始合并之前的状态
2. 合并冲突（Windows –[“回车”和“换行”（或者 CRLF 或 \r\n）]与 Unix （或者 LF 或 \）系统之间的换行符）
3. Caroline 的文件包含 LF，因为她的文件是在使用 LF 的 Mac OSX 上编辑的。如果某 Windows 用户要编辑 Caroline 的文件，Windows 文本编辑器可能会将所有 LF 均转换为 CRLF，以便该用户能够编辑文件。在该 Windows 用户将她的文件与 Caroline 的文件合并时，不同的 LF 和 CRLF 字符将会造成合并冲突。
4. 要修复此问题，Windows 用户应将 autocrlf 全局属性设置为 true：git config --global core.autocrlf true。
5. 分支比对命令 将提交与所在分支进行对比的命令是 git show commit\_id

3、合并冲突



git fetch origin 跟新远程跟踪分支



4、GITHUB

1）github 创建代码库 new repository

echo "# learning\_git" >> README.md

git init

git add README.md

git commit -m "first commit"

git remote add origin https://github.com/harveyhwliu/learning\_git.git

git push -u origin master

2)push

git remote add origin https://github.com/harveyhwliu/learning\_git.git

git push -u origin master

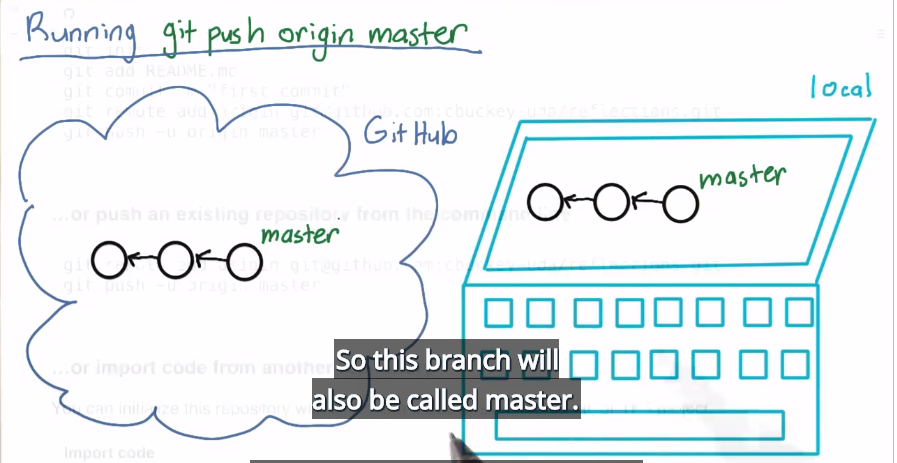
git remote 查看当前和创建远程代码库

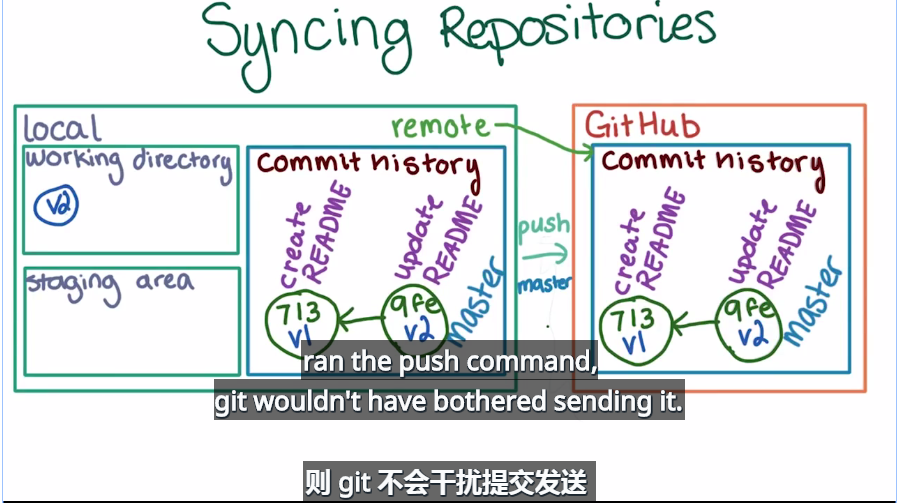
git remote add origin https://github.com/harveyhwliu/learning\_git.git 添加远程代码库 – 只有一个远程代码库时使用origin

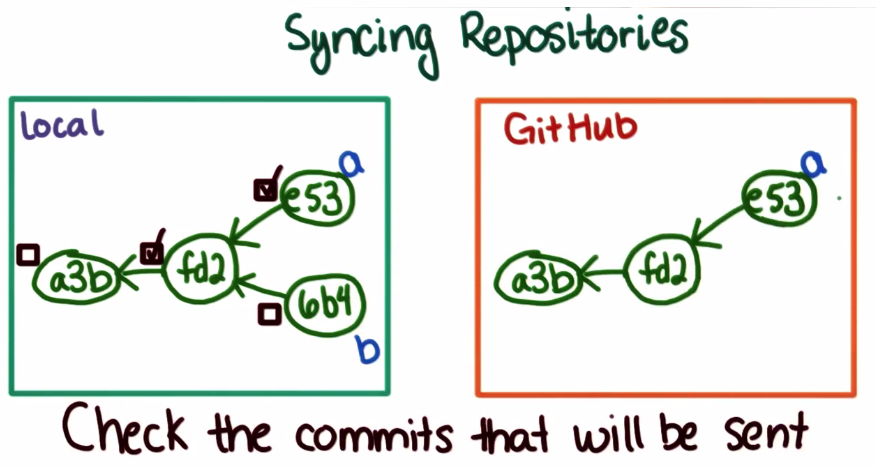
git push origin master 将本地master分支代码推送远程代码库origin

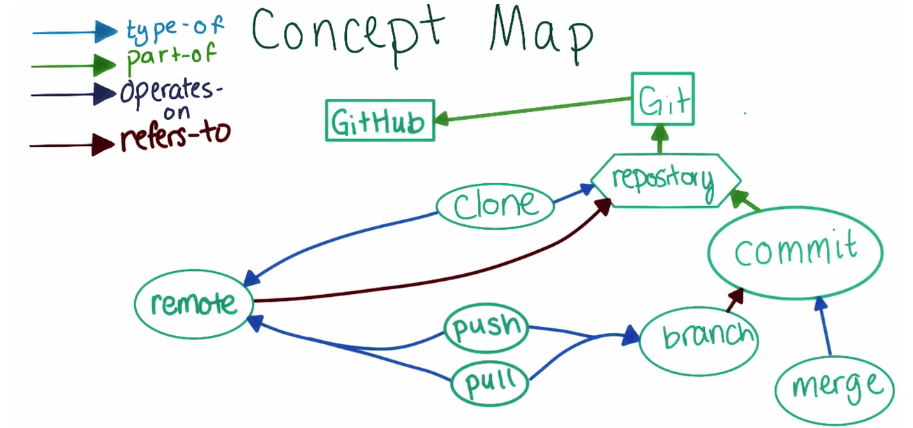
git pull origin master 将远程代码库origin内容拉取到本地master分支

git remote -v 查看远程代码库详细信息

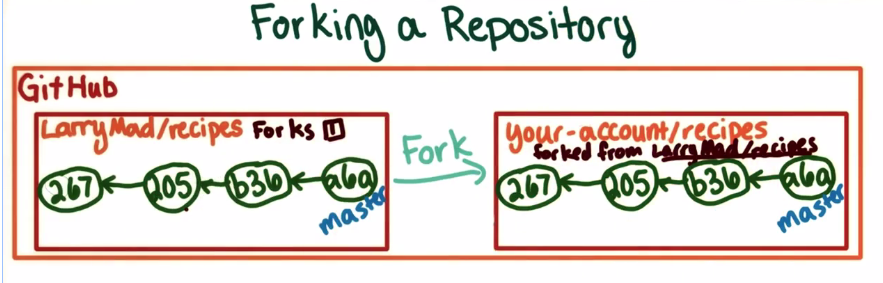








Github 复制别人的代码 fork，仅在github上使用



Clone 可以从远程仓库或者本地repository中进行，fork仅在github上进行

