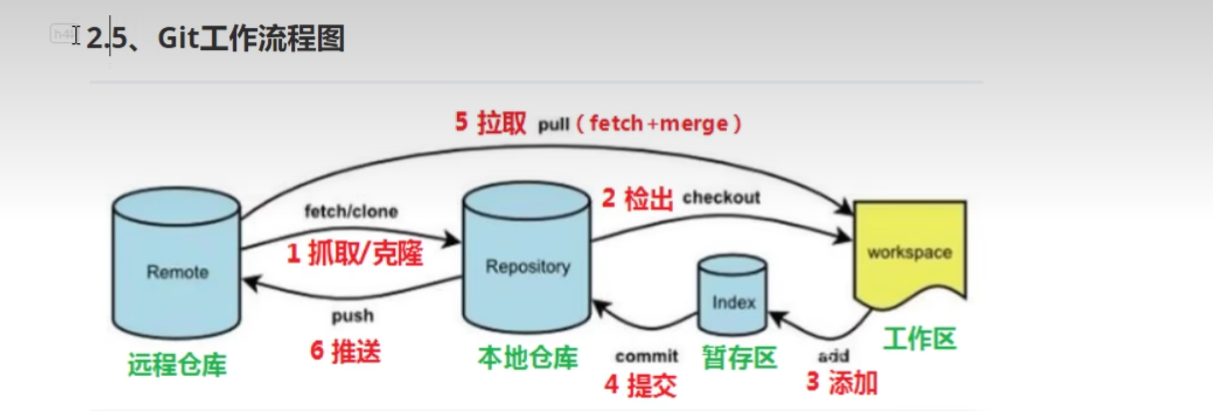
git

概念：

Git是一个开源的分布式版本控制系统，可以有效且高效的处理很小到非常大的项目版本管理



Git的常用指令：

git init 在指定文件夹中创建Git本地仓库

git add .|文件名 将文件夹中的所有文件都添加到暂存区中，.是通配符匹配所有文件

git commit -m “注释信息” 将暂存区中的文件全部交到本地仓库中管理

git status 查看修改的状态（暂存区、工作区）

git log 查看文件提交等日志，按q退出日志的查看

git reflog 查看被删除的日志

git reset --hard commitID （commitID可以在log日志中查看）回退到之前的版本

分支：

什么是分支，Git中的分支是指将文件同时分为两份相同的路线，每一个人修改一条分支上的文件，分支之间互不影响，多用于修严重bug以免影响代码的正常使用。

git checkout 分支名 切换分支

git checkout -b 分支名 切换分支，如果分支不存在则创建一个新分支

git branch 查看现有的本地分支

git branch 分支名 创建一条新的本地分支

git merge 分支名称 将分支合并为一条，一般将分支合并到master分支上

git branch -d 分支名 删除指定分支，不能删除当前所在分支

git branch -D 分支名 强制删除指定分支

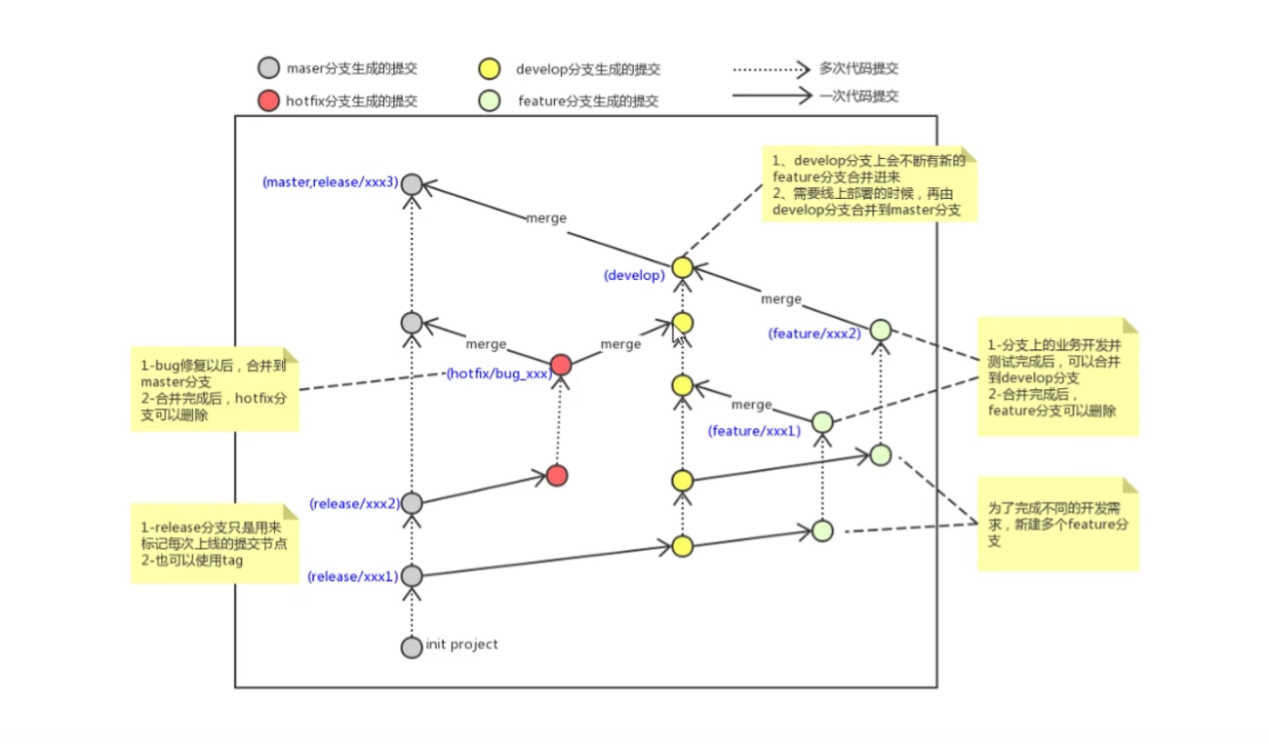
合并分支的流程：首先需要切换到想合并成的分支上，再将其他分支合并到该分支上

什么时候分支会删除失败需要强制删除，当分支作了修改但没有进行合并，删除修改的分支就会失败，防止误操作，这时就可以使用-D强制删除分支

解决分支合并冲突：

当两个分支修改的文件内容一样时，就会产生冲突，需要手动解决冲突，在合并后的文件上手动解决

开发的合并流程以及分支图



连接远程仓库：

1. 配置SSH公钥
2. 先初始化本地仓库，然后与创建的远程仓库进行对接

git remote add 自定义的仓库名(一般叫origin) 远程仓库的SSH地址

查看远程仓库：git remote

将代码推到远程仓库：

首次推送可以先将对应的分支与远程仓库的分支进行配对，使用的语法：

git push --set-upstream origin (本地分支):(远程分支)分支匹配后之后的推送代码只需要git push即可，会自动推送到对应的分支

将远程仓库的代码拉取到本地：

1. git fetch [自定义的仓库名origin] [要抓取的远程分支]将指定分支抓取到本地仓库但不会合并的本地的分支，需要手动用merge进行合并。如果没有指定分支名称，就自动抓取和本地分支匹配的分支
2. git pull [自定义的仓库名origin] [要抓取的远程分支] 将指定分支抓取到本地仓库会自动匹配对应的分支进行合并，相当于fetch+merge。如果没有指定分支名称，就自动抓取和本地分支匹配的分支

克隆仓库clone：

当已经有一个仓库时，中途有另一个开发者加入项目的开发，就首先要从远程仓库将所有的代码拉取到他自己本地的仓库，这种行为就叫clone。一般每一个人只会进行一次克隆操作，后面的代码更新只需要将对应分支的代码拉取更新到本地即可，即使用fetch和pull。