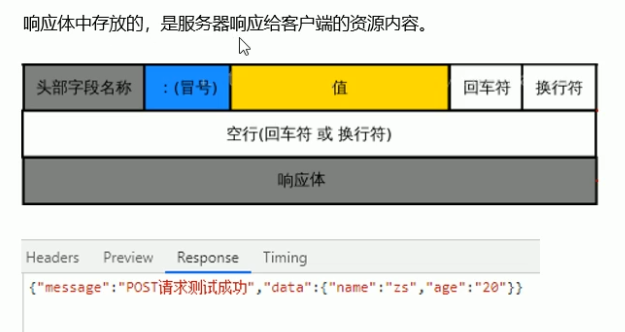
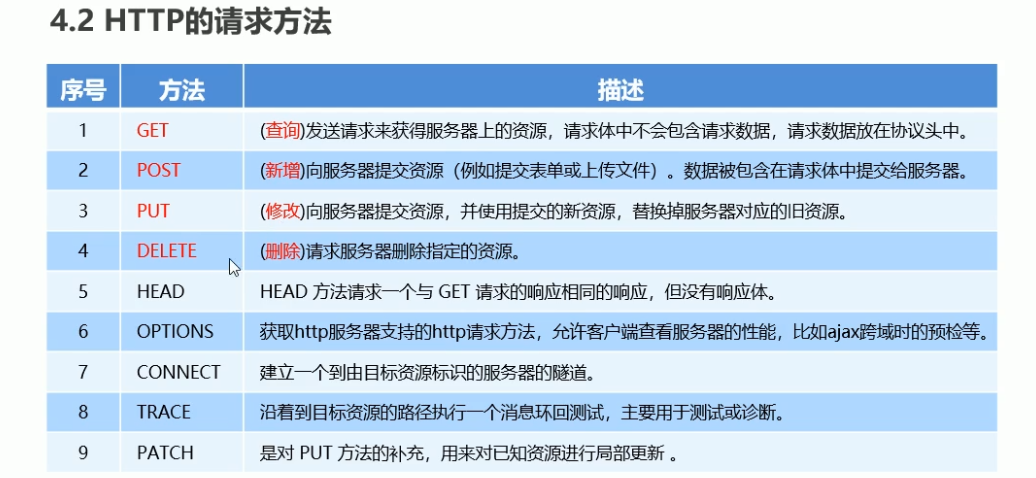
HTTP响应消息



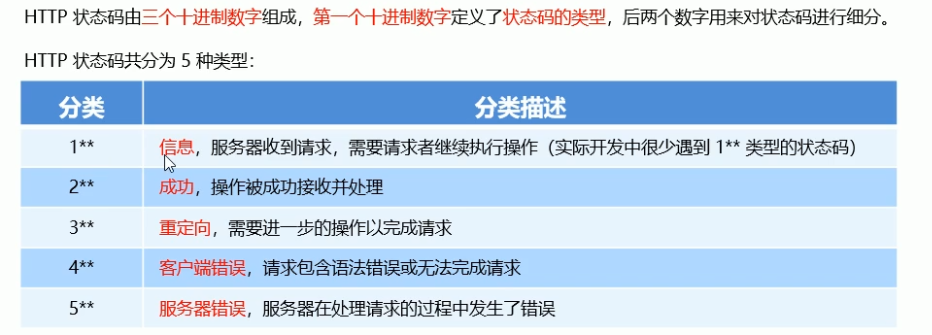


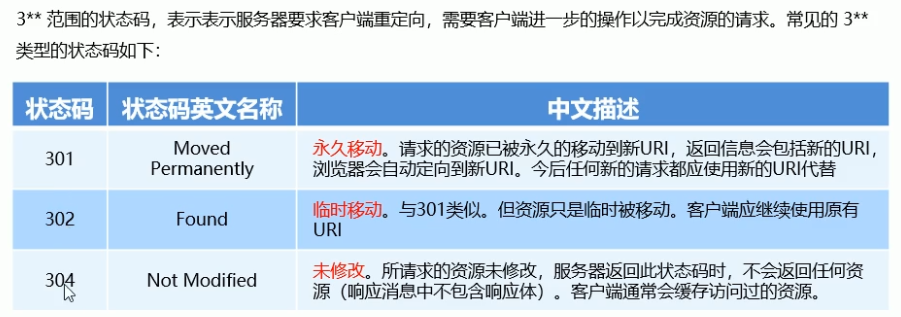
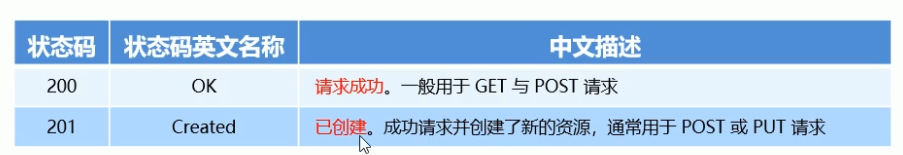


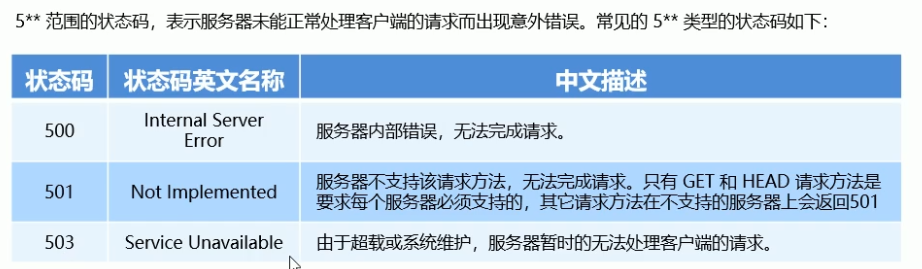
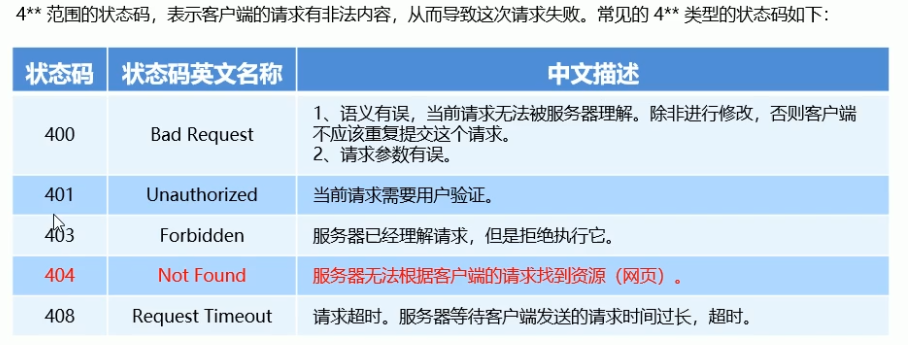
响应体在response中



http请求方法

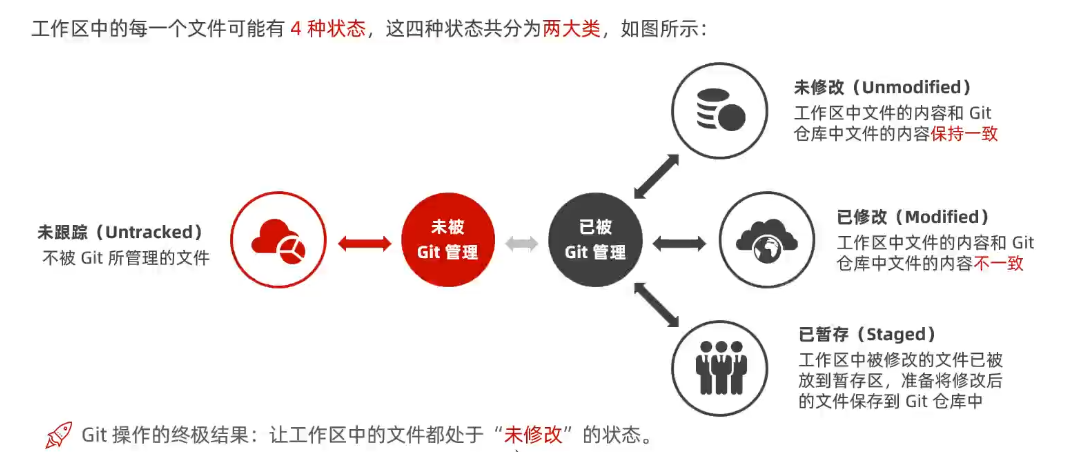






Git 操作：

git init ：初始化，产生隐藏目录



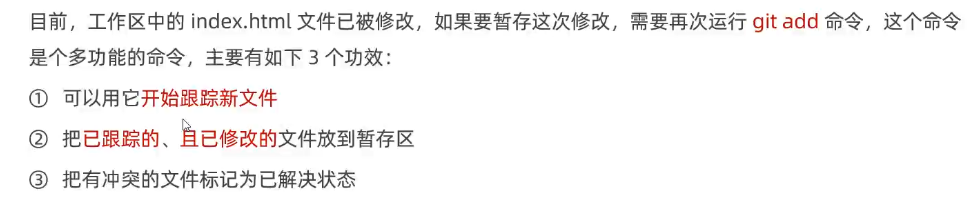
git status：查看文件状态

git status -s :简洁状态报告

？？表示未跟踪 A 表示已暂存 M表示已修改但未放入暂存区 M表示已修改且已放入暂存区 D表示下次commit时，将从仓库中移除这个文件

git add 文件名：加到暂存区

git commit [-m “备注”] 将暂存区中的文件提交到仓库



git checkout -- 文件名：撤销修改，其实就是用仓库中的文件覆盖指定的文件

git add . ：将当前目录所有文件加入暂存区

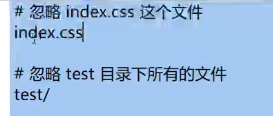
git reset HEAD 文件名：将文件从暂存区移除

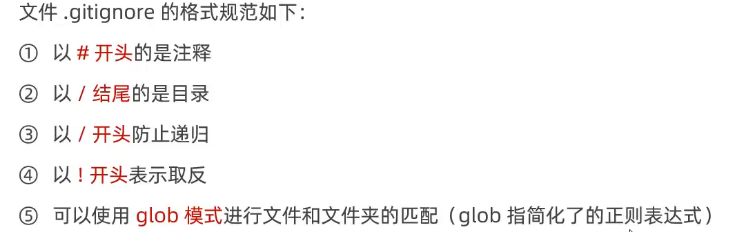
git commit -a：将所有已跟踪的文件直接加入仓库，跳过加入暂存区的步骤

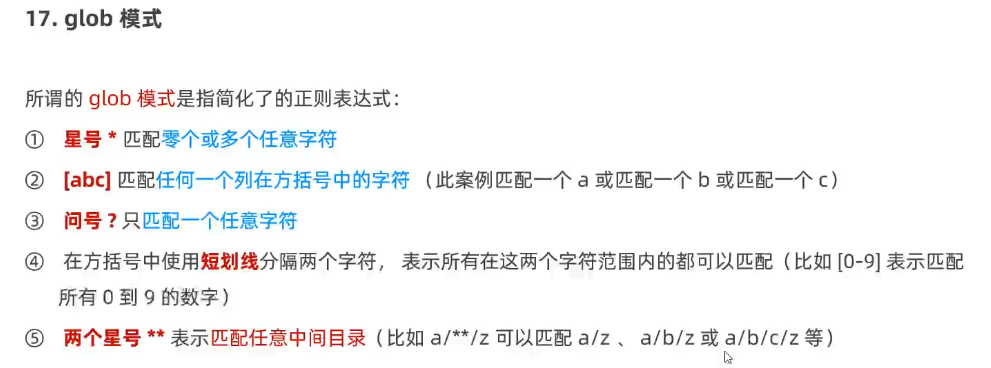
移除文件：从仓库和工作区同时移除；只从仓库移除，而将工作区中的文件状态修改为？？



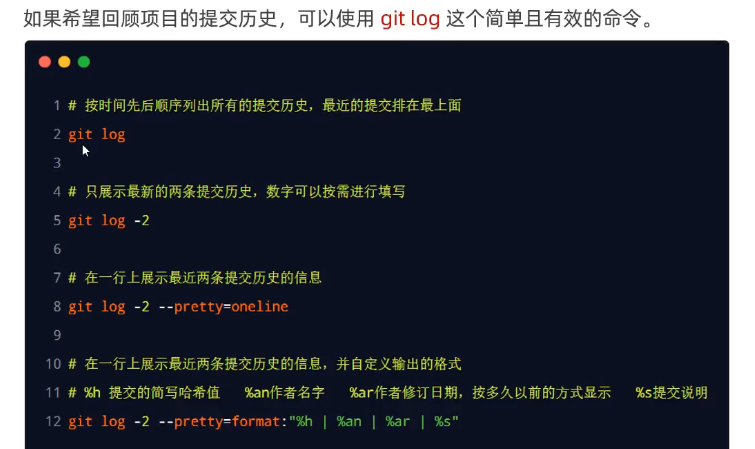
移除文件只有在下次再commit时才会生效，因为他修改的是文件的状态。

git忽略文件，.gitignore，如：





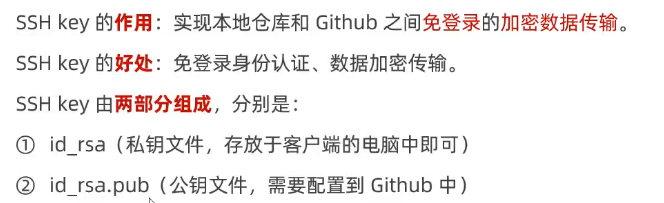
查看提交历史



回退到指定版本，要注意在旧版本中必须使用reflog才能查看完整的提交历史



git push：第二次及以上更新仓库





ssh-keygen -t rsa -b 4096 -C “[2379849682@qq.com](mailto:2379849682@qq.com)”

配置SSH key



检查SSH是否配置成功



将远程仓库克隆到本地



git分支

查看分支：git branch，\*代表当前所处的分支

创建分支：git branch 分支名称，不会切换分支，即创建新分支后，不会跳到新分支上去

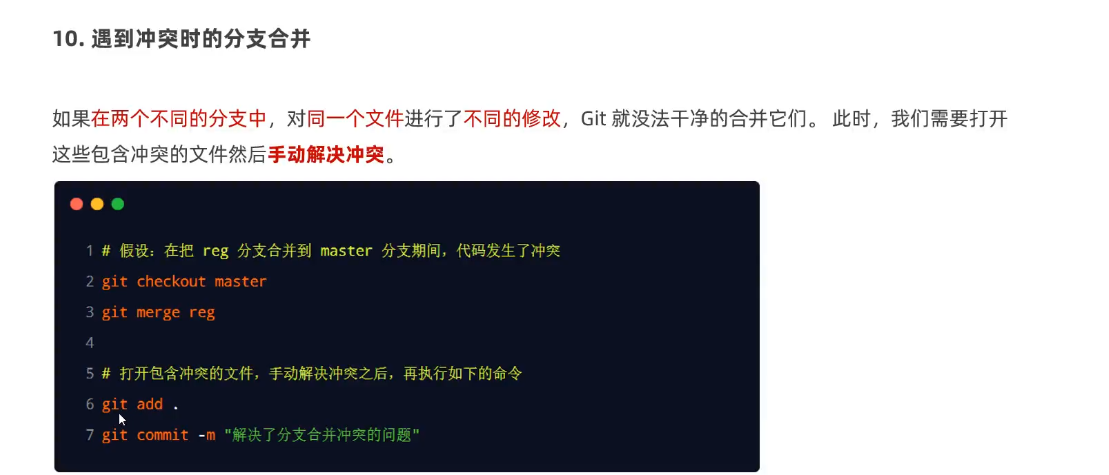
切换分支：git checkout 分支名称

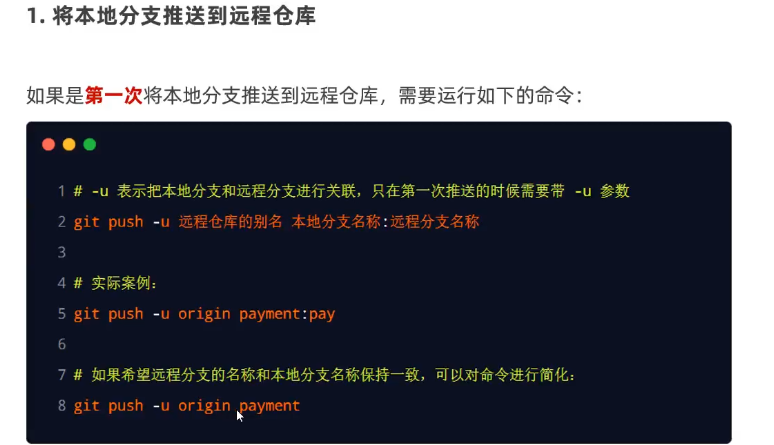
创建并立即切换分支：git checkout -b 分支名称

合并分支：git merge 分支名称，即将指定分支合并到当前所处的分支

合并分支前，先在子分支将代码commit ，再合并到主分支

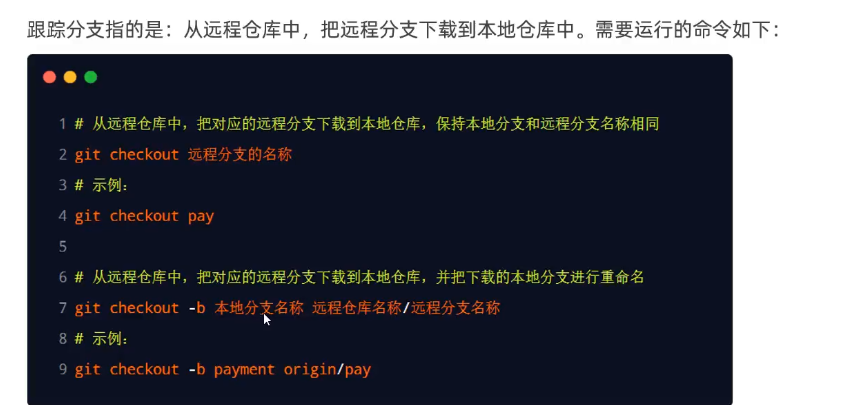
删除指定分支：git branch -d 分支名称

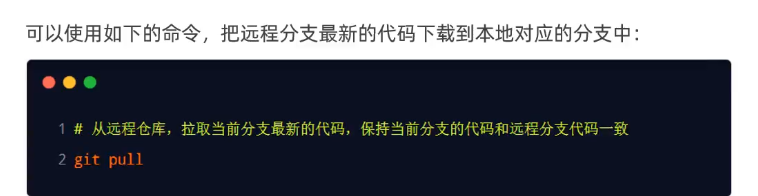






git remote show 仓库名称







iframe标签配合 a标签的使用

隐藏域：

隐藏的、看不见的input框

layui快速为表单赋值：layui.form.val（‘表单lay-filter属性值’，data）

注意快速填充的时候，input框的name属性和data对象里面的属性名要相同

JQ：serialize（）函数，快速获取表单内容，属性名就是输入框name属性的值。

layui.form.render( )方法，让layui重新渲染一下表单结构，避免表单不更新。

layui分页切换时，获取当前页的页号等属性：



使用到了jump这个回调函数，当发生分页切换时，被触发（直接从jump函数开始执行）。obj.curr就是当前页号

触发jump回调函数的第二种方式：调用laypage.render（）方法后，也会无条件执行一次jump回调函数，这个是从laypage.render（）方法开始执行，而不是直接从jump函数开始执行。即是否是first的区别

为了解决上个问题，所以才有first参数 ，即注释中首次不执行的代码

获取某个class元素的个数：$(‘.name’).length

FormData 对象：new FormData（参数是一个DOM元素）

向FormData 对象中追加一个键值对：fd.append（‘key’，value）