

Git & GitHub 实验楼课程学习

笔记本： computer

创建时间： 2019/4/10 21:03

更新时间： 2019/4/13 21:31

作者： 1462486173@qq.com

URL： <https://www.shiyanlou.com/courses/running>

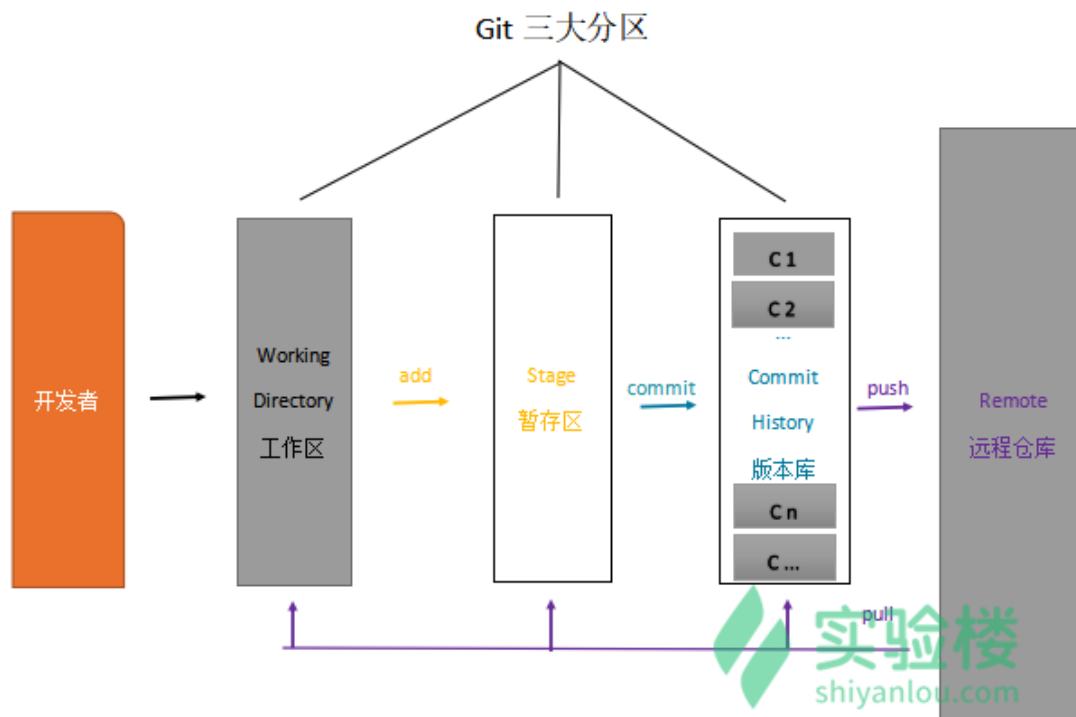
1、Git与GitHub简介

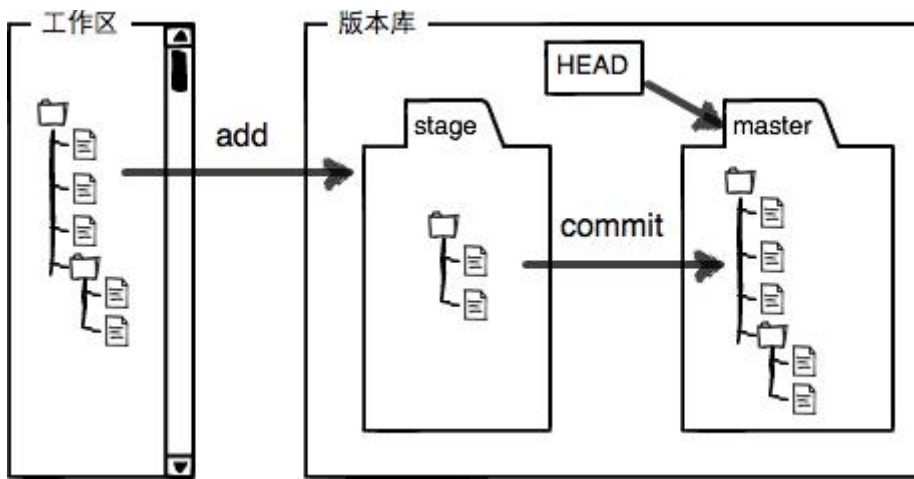
1. 在github 上创建仓库
2. 克隆github 上的仓库到本地： `git clone + [仓库地址]`

查看本地仓库所关联远程仓库： `git remote -v`

2、Git基本操作

1. Git三大区域：工作区、暂存区、版本区





2. 一次完整的修改、提交、推送操作

`git status` : 查看仓库状态

`git add` : 将文件添加到暂存区 (`git add .` 将全部文件添加)

`git reset -- +[文件名]` : 将暂存区修改文件撤销 (不加文件名则表示全部撤销)

`git diff` : 查看工作区被跟踪文件的修改详情 (只有在版本区存在的文件才是被跟踪文件)

3. 查看提交历史

查看提交历史常用命令 :

`git log [分支名]` : 查看某分支的提交历史, 不写分支名查看当前所在分支

`git log --oneline` : 一行显示提交历史

`git log -n` : 其中 `n` 是数字, 查看最近 `n` 个提交

`git log --author [贡献者名字]` 查看指定贡献者的提交记录

`git log --graph` 图示法显示提交历史

4. 配置个人信息git

接下来需要对 Git 进行一些本地配置 :

`user.email` : 写入你自己注册 GitHub 账号的邮箱

`user.name` : 你自己的 GitHub 账号名字

5. 提交暂存区修改

`git commit -m '备注内容'` : 生成新的提交

`git log` : 查看提交日志

`git branch -avv` : 查看全部分支信息

`git push` : 将本地新增的提交推送到GitHub远程仓库中 (命令后无需添加任何东西)

6. 版本回退

首先执行 `git reset --soft HEAD^` 撤销最近的一次提交, 将修改还原到暂存区。--soft 表示软退回, 对应的还有 --hard 硬退回, 后面会讲到, `HEAD^` 表示撤销一次提交, `HEAD^^` 表示撤销两次提交, 撤销 `n` 次可以简写为 `HEAD~n`。软退回一个提交后执行 `git branch -avv` 命令查看分支信息

7. commit 时间线分叉

`git push -f` : 表示强制推送

3、Git分支操作

1.git branch [分支名] : 创建本地分支

`git checkout -b [新分支名]` : 创建新的本地分支并自动切换到该分支

2.git push [主机名] [本地分支名] : [远程分支名] : 将本地分支推送到远程仓库中 (通常冒号前后名相同。冒号后不存在时, 使用该命令可自动创建该远程分支)

- 3.git branch -u [主机名/远程分支名] [本地分支名]：本地分支跟踪远程分支
- 4.git push [主机名]：[远程分支名]：删除远程分支（git push [主机名] --delete [远程分支名] 作用相同）
- 5.git branch -D [分支名]：删除本地分支
- 6.git branch -m [原分支名] [新分支名]：给本地分支改名（修改当前分支名，则可省略原分支名不写）
- 6.删除远程分支可以再任意本地分支进行。删除当前分支需要切换到master分支进行

4、Git标签tag

- 1.tag标签：issue 是仓库拥有者在 GitHub 上手动创建的，仓库被 Fork 时 issue 不会跟随。Tags 通常在本地使用 git 命令创建后推送到 GitHub 上，与 issue 相同的一点，它也只存在于项目仓库内，Fork 或提 PR 都不会带上它
 - 2.git tag [标签名] -m [备注信息] [提交版本号]：创建标签（其中 -m [备注信息] 可以省略不写，但建议不要省略。[提交版本号] 可以省略，如果是给当前分支最新的提交创建标签的话。）
- git tag :显示仓库中的全部标签列表
- git show [标签名]：查看标签详情
- git add [标签名]：创建本地标签
- git tag -d [标签名]：删除本地标签
- git push origin [标签名]：推送标签到远程仓库，需要联网
- git push origin :refs/tags/[标签名]：删除远程仓库的标签，命令中的标签名其实也就是文件名：
-

5、提交本地文件到Github

- 1.直接在线提交：在仓库界面选择“upload files”，将文件拖拽即可
- 2.git上传：

- 在本地建立一个可git 管理的文件夹：cd 到该目录（一般设置该文件名为工程名，对应 github上的仓库名）
- git init：将该目录（文件夹）设置为git 可管理的（该文件夹即为主仓库master）
- git add .:将所有文件添加到仓库 git commit -m '提交注释'：提交文件
- 关联github仓库：在github上创建同名仓库，复制仓库地址

git remote add origin [仓库地址]：关联

- git push -u origin master；上传本地代码