

Learn Git

创建版本库

- mkdir learnGit — cd learnGit — pwd
- 初始化Git仓库 — git init
- 添加文件到Git仓库
 - git add <file>
 - git commit -m <message>

时光机穿梭

工作区状态

- git status
- git diff — 查看工作区和版本库最新版本的差别

历史记录

- git log
 - git log --pretty=oneline
 - commit id
- git reflog

版本回溯

- HEAD 最新版本
 - 上个版本 — HEAD^
 - 上上个版本 — HEAD^^
 - 上100个版本 — HEAD~100
- git reset
 - git reset --hard HEAD^
 - git reset --hard <commit id>
 - git cat <file>

工作区和暂存区

- 工作区 Working Directory — 目录 learnGit
- ! add
- 版本库 Repository — .git
 - 暂存区 stage/index
 - ! commit
 - master — 由HEAD指向

管理修改

- 第一次修改 — git add — 第二次修改 — git commit — 结果：只提交第一次修改
- 第一次修改 — git add — 第二次修改 — git add — git commit — 如果不用git add, 就不会加入commit

撤销修改

- 工作区：丢弃工作区修改 — git checkout --file
- 暂存区：修改丢弃
 - git reset HEAD <file>
 - 丢弃工作区修改
- 已提交到版本库 — 版本追回 (未推送到远程库)

删除文件

- 确认删除 — rm <file>
- 删除文件 — git checkout --<file>

远程仓库 GitHub

创建SSH key

- ssh-keygen -t rsa -C "youremail@example.com"
- github settings

添加远程库

- git remote add origin git@server-name:path/repo-name.git
- git push -u origin master
- git push origin master

从远程库克隆

- 使用ssh: git clone git@github.com:path/name.git
- 使用https: https://github.com/path/name.git

分支管理

创建与合并分支

- 查看分支: git branch
- 创建分支: git branch <name>
- 切换分支: git checkout <name>
- 创建+切换分支: git checkout -b <name>
- 合并某分支到当前分支: git merge <name>
- 删除分支: git branch -d <name>

解决冲突

- 无法自动合并分支，先解决冲突再提交合并。
- 手动编辑Git合并失败的文件
- git log --graph 分支合并图

分支管理策略

- master仅仅用来发布新版本，在dev上干活
- no-ff 普通模式合并 能看到历史
- fast forward合并看不出曾经做过合并

bug分支

- 创建bug分支修复bug
- git stash --> 修复bug --> git stash pop

Feature分支

- 开发新Feature最好创建新分支
- 强行删除未合并分支: git branch -D <name>

多人协作

- 直到远程库信息 git remote -v
- 本地新建分支不推送则不可见
- 本地推送分支
 - git push origin branch-name
 - 若失败 --> git pull --> 处理冲突
- 本地创建与远程对应分支 — git checkout -b branch-name origin/branch-name
- 建立本地分支和远程分支关联 — git branch --set-upstream branch-name origin/branch-name

rebase

标签管理

创建标签

- git tag <tagname> 默认HEAD也可以指定commit id
- git tag -a <tagname> -m "指定标签信息"
- git tag 查看所有标签

操作标签

- git push origin <tagname>
- git push origin --tags
- git tag -d <tagname>
- git tag origin refs/tags/<tagname>

自定义Git

忽略特殊文件

配置别名

搭建Git服务器