

Git学习

李建霖 四月八日

Git基本理论

- Git的四个工作区域
 - 工作目录 (Working Directory) — 平时存放代码的地方
 - 暂存区 (Stage/Index) — 用于临时存放改动, 本质上是个文件
 - 资源区 (Repository /Git Directory) — 安全存放数据的位置, 有提交到所有版本的数据, 其中HEAD指向最新放入的仓库的版本
 - 远程git仓库 (Remote Directory) — 托管代码的服务器
- Git工作流程
 - 1、在工作目录中添加、修改文件
 - 2、将需要进行版本管理的文件放入暂存区域
 - 3、将暂存区域的文件提交到git仓库
- git项目搭建
 - 创建本地仓库: git init
 - 克隆远程仓库: git clone http://39.101.133.168:8807/yanjiaqing/gold_java.git

Git文件操作

- 文件的四种状态
 - Untracked**: 未跟踪, 此文件在文件夹中, 但并没有加到git仓库, 不参与版本控制, 通过git add状态变为**Staged**
 - Unmodify**: 文件已入库, 未修改, 两种去处
 - 1、被修改后变为**Modified**
 - 2、使用git rm移出版本库, 变为**Untracked**文件
 - Modified**: 文件已修改, 仅仅是修改, 并没有进行其他的操作, 两种去处
 - 1、通过git add进入暂存**Staged**状态
 - 2、使用git checkout (从库中取出文件, 覆盖当前修改) 丢弃修改, 返回到**unmodify**状态
 - Staged**: 暂存状态。执行git commi则将修改同步到库中, 这是库中的文件与本地文件又变为一致, 文件为**Unmodify**状态, 执行git reset HEAD filename 取消暂存, 文件状态变为**Modified**
- 查看文件状态
 - 查看指定文件的状态: git status [filename]
 - 查看所有文件状态: git status
 - 添加所有文件到暂存区 — git add .
 - 提交暂存区中的内容到本地仓库 — git commit -m "消息内容"
- 忽略文件
 - 不想把有些文件纳入版本控制中, 如数据库文件, 临时文件, 设计文件等
 - gitignore文件规则
 - #为注释, 会被忽略
 - *.txt:忽略所有以.txt结尾的文件
 - ! lib.txt lib.txt文件除外
 - /temp/: 仅忽略项目根目录下的TODO文件, 不包含其他目录temp
 - build/ :忽略build/目录下的所有文件
 - doc/*.txt: 会忽略doc目录下的txt文件, 但子目录中的不会忽略
- IDEA上使用Git
 - 新建文件, 绑定Git
 - 修改文件, 使用IDEA操作git
 - 添加到暂存区
 - commit提交
 - push到远程仓库

Git分支

- 分支中的常用命令
 - git branch : 列出所有本地分支
 - git branch -r:列出所有远程分支
 - git branch [b-name]:新建一个分支, 但依然停留在当前分支
 - git branch -b [b-name]: 新建一个分支并切换至新分支
 - git merge [branch]: 和并指定分支到当前分支
 - git branch -d [b-name]: 删除分支
 - git push origin --delete [b-name]
 - git branch -dr [remote/branch] } 删除远程分支

版本控制

- 什么是版本控制
 - 版本迭代
 - 实现协同开发
 - 追踪记录文件的历史记录
 - 并行开发, 提高开发效率
 - 统计工作量
 - 减轻开发人员的负担, 节省时间, 降低人为错误
- 版本控制的作用
- 版本控制的工具
 - Git、SVN、CVS、VSS等
- 版本控制分类
 - 所有版本都放在本地的电脑上 — 本地版本控制
 - 所有版本数据都保存在服务器中 — 集中式版本控制 SVN
 - 所有版本信息同步到每一个用户, 可以在本地看到历史记录, 不会因为服务器损坏或网络问题造成不能工作的问题 — 分布式版本控制 Git
- Git与SVN的区别
 - Git是分布式版本控制, 每个电脑都是一个完整的库, 工作时不需要联网。可以之间看到更新了哪些文件, 是目前最先进的分布式版本控制
 - SVN是集中式版本控制, 需要联网才能工作

准备工作

- 基本的Linux命令 (Git Base中使用)
 - 1、cd: 切换目录
 - 2、cd ..回退到上一个目录
 - 3、pwd: 显示当前所在的目录路径
 - 4、ls (ll) : 列出当前目录中的所有文件, ll列出的内容更为详细
 - 5、touch: 新建一个文件
 - 6、rm: 删除一个文件
 - 7、rm -r: 删除一个文件夹 (目录)
 - 8、mkdir: 新建一个目录 (文件夹)
 - 9、mv: 移动文件mv A B 把A文件移动到B目录中
 - 10、reset: 清屏, 重新初始化终端
 - 11、clear: 清屏
 - 12、history: 查看命令历史
 - 13、help: 帮助
 - 14、exit: 推出
 - 15、#: 注解
- Git环境变量
 - 配置用户名: git config --global user.email XXXX
 - 配置邮箱: git config --global user.email xxxx