

→ 需要虚拟机

# Git Talk 1



# Git 是什么

- Git 是目前世界上最先进的分布式版本控制系统
  - 来自廖雪峰的官方网站教程
  - HCC 目前的所有项目，包括幻灯片都在用 git 管理



# Git 设计理念

- 项目目录

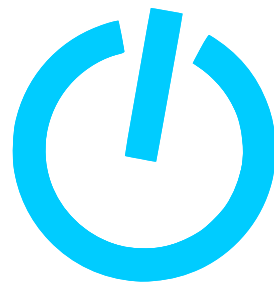
- .git 不可直接读取
  - 暂存区 临时存储
  - 版本库 永久存储
- 其他目录和文件
  - 工作区 开发使用

- shiyiquan

- .git
  - 暂存区
  - 版本库
    - 98d22f3
    - 118532c
- shiyiquan
- README.md

# 使用 Fedora

- 我们将会用 Fedora 虚拟机使用下面的功能
- 步骤
  - 打开桌面右下角的 Oracle VM VirtualBox
  - 点击“启动”按钮
  - 中途需要输入两次密码，都是
    - shiyiquan



# 基础操作

- 创建版本库

- mkdir `text`
- cd `test`
- git init

- 开始工作

- cat > `jdk1`

- 提交代码

- git add `jdk1`
  - # 工作区 → 暂存区
- git add -A
  - # 添加全部文件
- git commit
  - # 第一次提交需设置用户
  - # 暂存区 → 版本库
  - # 和 vi 操作一样

# 状态查看

- `git status`

- # 检查状态

- `git diff`

- # 对比工作区和暂存区

- `git log`

- # 查看版本库日志

- 尝试

- `cat >> test`

- `git add -A`

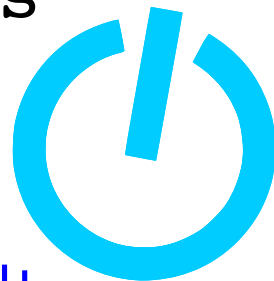
- `cat >> test`

- `git status`

- `git diff`

- `git log`

- # 按 `q` 退出



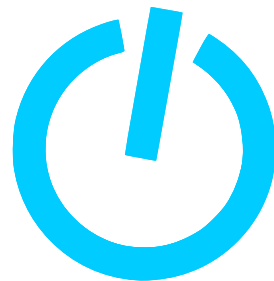
# 高级选项

- `git log`
  - `--oneline`
    - # 摘要模式
  - `--graph`
    - # 生成代码关系图
- `git commit`
  - `-m` '说明'
  - # 不进入 vi
- `--graph`
  - 在版本复杂后很有用
- `--oneline`
  - 左侧是版本号，输入四位以上就可以



# 回溯

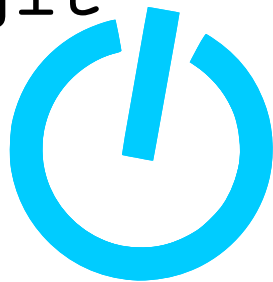
- 命令
  - `git reset --hard`  
版本号
- 特殊用法
  - `HEAD^`
    - # 前一个版本
  - `HEAD~10`
    - # 前 10 个版本
- 回溯后将会无法看到更靠后的版本
- 查看操作记录
  - `git reflog`
  - # 可以看到以后的版本并跳转





# 连接远程版本库

- 通过连接提供 git 服务的服务器同步代码
  - `git remote add origin 链接`
- 链接
  - `https://git.oschina.net/ 用户 / 项目 .git`
  - `git@git.oschina.net: 用户 / 项目 .git`
  - # 服务商一般会提供链接
  - 甚至可以是一个本地的 git 目录
  - # "origin" 是名称，用于快速访问链接



# 推送和接收

- 推送

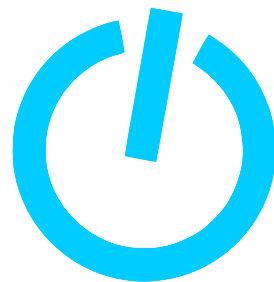
- `git push origin master`
  - # 将当前所在分支推送到 origin 的 master 分支

- 接收

- `git pull origin master`
  - # 将 origin 的 master 分支拉到当前分支

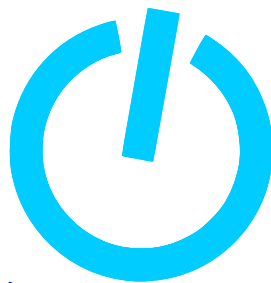
- 分支

- "master" 是默认分支名称



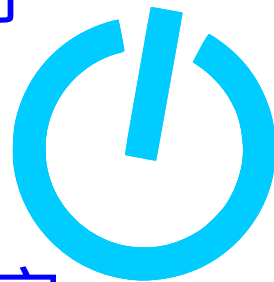
# 分支

- 分支可以代表不同阶段的代码，可以互相合并
- 创建
  - `git branch dev` # 从当前分支创建 dev 分支
- 跳转
  - `git checkout dev` # 切换到 dev
- 合并
  - 从网络拉取
  - `git merge master` # master → 当前



# 分支练习

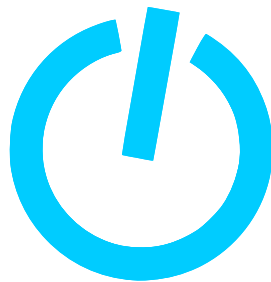
- 从 master 开始执行
  - git branch dev
  - echo master version >> jkl
  - git commit -am 'master' # 相当于 add 后提交
  - git checkout dev # jkl 变了
  - echo dev version >> jkl
  - git commit -am 'dev'
  - git merge master # 出现冲突



# 冲突

- 冲突后 git 会提示冲突的文件，只要按照标记手动合并然后重新提交即可

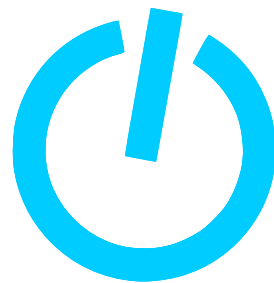
```
- <<<<<<< HEAD          # 当前分支
- dev version
- =====                # 分割线
- master version
- >>>>>>> master        # 合并分支
```



# 其它功能

→ 这个不讲

- `git tag` # 标记代码版本
- `git stash` # 隐藏版本
- `git config` # 自定义 git
- `.gitignore` # 忽略文件
- `git grep` # 全文搜索



# 感谢参加此次活动

