**学习日志**

姓名： 陈昭宇 学号：1812977 日期：6.15

**学习内容安排：**

上午：

Git简介、安装Git、创建版本库、版本回退、工作区与暂存区、管理修改、撤销修改、删除文件

下午：

远程仓库、分支管理、标签管理、使用GitHub、忽略特殊文件、配置别名、使用SourceTree

**学习反馈：**

已掌握知识：

Git的来源（Linux用两周用C语言开发【因为他不喜欢集中式】）

Git版本库的创建（使用git pwd选择当前的操作位置，用git init 变成版本库【选择磁盘用cd: d\和文件系统不一样】）

Git的回退（用git log 查看日志；git reset-hard head^【head是一个指针默认在master分支里，head^是上个版本；也可直接通过id reset】）

Git的修改（用git add添加git admit确认【有点像MYSQL ; add从工作区加入暂存区；admit才加入版本库】）

Git撤销（用git checkout --file）

Git删除（先在文件系统里删除，再git rm file【这样在误删时可以用git checkout-file恢复】）

Git远程库（在github中创建，然后git pull 下来，通过git push回去）

Git分支（一般操作时master分支发布稳定版本，dev接受所有人的分支的合并）

Git分支创建与合并（用git swith -c 分支名创建并进入【也可用checkout -b】；用git merge合并 git branch -d 删除【一般在分支上操作完再合并主分支然后删除自己的分支】）

Git特殊分支（

BUG分支用git stash停止手头工作；创建bug分支修改；用git stash apply回复

Feature分支就是开发一个新功能时开一个分支，方便加入和删除

）

Git标签管（用git tag name 添加标签【和commit同步】git tag -d 删除本地标签 git push origin tagname 推送标签）

未掌握知识：

知识都学习到了就是关于分支冲突的部分没有实践

学习心得：Git真的是一个非常优秀的版本控制器，可以清晰明确的显示每个参与者的分支；方便开发。但是知识的了解和学会使用可能还需要一段时间的熟悉，这次的报告我也会创建自己的分支，然后合并到master上，希望通过这样能够更加熟悉今天学习的git语句。