昆明城市学院

实 训 报 告

**2022—2023学年** **第**1**学期**

**学 院：数工学院 专业班级：**

**姓 名： 学 号：**

**课程名称：Git版本控制 任课教师： 卫明**

|  |  |
| --- | --- |
| **总 成 绩** |  |
| **教师签名** |  |

填写说明

1、学生应于每次实验课结束后，根据指导教师的要求，认真完成实验报告，并按时提交。

2、实验项目中若涉及药品或试剂，应在“实验设备、材料”一栏注明药品或试剂的名称、浓度、纯度级别和用量。

3、实验记录应按照实际实验操作填写。

4、实验项目结束后，应对实验过程和结果进行认真地分析和总结。

5、学生应妥善保管好实验报告。

实训报告

|  |  |  |  |
| --- | --- | --- | --- |
| **实验名称** | Git练习一 | **实验日期** | 2022年10月14日 |
| **课程名称** | Git版本控制 | **实验项目** | Git练习一 |
| **一、实验目的**  1. 掌握Git配置账户基本操作。  2.掌握Git本地仓库创建、git搭建服务器并配置ssh(免密钥登录)。  **二、实验原理**  **1.git配置账户**  (1)在桌面右键选择Git Bash Here输入命令(对应你 的github的账户名称和邮箱)  (2)git config --global user.name "ZhangXiaoxiang"  (3)git config --global user.email "1090510535@qq.com"  (4)此时在用户目录下已经生成配置文件,如我的是 C:\Users\10905\.gitconfig  **2.** **git搭建服务器并配置ssh(免密钥登录)**   1. 搭建git服务器（远程仓库） ：统一的托管网站（https://github.com/） 2. 由于每次提交(本地到服务器(github))都会需要输入密码,所以最好设置免密钥登录， 3. 可以配置ssh ,以免每次提交代码都要输入密码.一劳永逸. 4. 现在本地生成ssh： 5. ssh-keygen -t rsa -C 1090510535@qq.com 一直回车(直到出现一个图形表示成功,此次本地已经生成ssh) 6. 比如我的路径下C:\Users\10905\.ssh有2个文件(id\_rsa.pub表示公共密码,这个文件用文本编辑器打开,并复制密码) 7. 忒别注意赋值粘贴后需要看一下是否有回车符,不然待会测试连通性会报错,没有权限啊什么的. 8. 在远程服务器github官网(托管平台)设置ssh: 9. 图片展示   IMG_256   1. 测试连通性(固定命令):ssh -T git@github.com 2. Hi ZhangXiaoixiang! You've successfully authenticated, but GitHub does not provide shell access. 3. (有You've successfully authenticated就行了,不管but后面的了)   **3.在远程建立git项目仓库(git新建一个仓库)**   1. https:*//github.com/ZhangXiaoixiang/testGit.git* 2. 本地新建项目 3. 如(C:\Users\10905\Desktop\localdemo有一个Git的java项目) 4. 在桌面右键选择Git Bash Here 5. git init(在该目录下会生成.git的隐藏文件) 6. 指明本地项目需要关联的远程仓库(如果是第二次会包已将存在了,所以第一次需要指定) 7. git remote add origin https:*//github.com/ZhangXiaoixiang/testGit.git* 8. git add . *//文件-暂存区(注意add后有空格)* 9. git commit -m "注释内容:这是第一次提交" *//暂存区-本地分支（默认master）* 10. git push -u origin master(国外网站有点慢,然后执行完毕到GitHub官网看一下是否提交成功   **4.第一次下载项目(本地存代码的目录)（远程-本地）**   1. 同理(本地需要获取项目的文件夹目录里 右键-git bash),输入克隆(下载)命令 2. git clone https://github.com/ZhangXiaoixiang/testGit.git 3. 项目下载完毕后就可以对项目进行修改了,比如修改完成后 4. 提交(本地-远程) 5. git init(最好还是初始化一下) 6. git add . 7. git commit -m "提交到分支,修改了代码了" 8. git push origin master(然后到GitHub官网看一下是否提交成功) 9. 如果提示需要账户名和密码(第一次会需要输入提示,以后就不会了) 10. 更新(远程-本地) 11. git pull(远程代码就更新了,相当于再次刷新,如果服务器端修改了代码,直接拉取一下,和服务器保持一致的意思)   **注：参考**1.https://blog.csdn.net/wozniakzhang/article/details/86491809?utm\_term=windows%E4%B8%8B%E4%BD%BF%E7%94%A8git%E5%91%BD%E4%BB%A4&utm\_medium=distribute.pc\_aggpage\_search\_result.none-task-blog-2~all~sobaiduweb~default-0-86491809-null-null&spm=3001.4430  2. https://blog.csdn.net/quiet\_girl/article/details/78359202?utm\_term=windows%E4%B8%8B%E4%BD%BF%E7%94%A8git%E5%91%BD%E4%BB%A4&utm\_medium=distribute.pc\_aggpage\_search\_result.none-task-blog-2~all~sobaiduweb~default-1-78359202-null-null&spm=3001.4430  **三、实验设备、材料**  1.Windows系统计算机一台  2.git环境  **四、实验过程** | | | |
|  | | | |