

实验一实验报告

1. git安装 安装git, 从git官网下载后直接点击可以安装: git官网地址 从Github克隆课程的仓库: 课程的仓库地址, 运行git bash应用 (该应用包含在git安装包内), 在命令行输入下面的命令 (命令运行成功后, 课程仓库会默认存放在Windows的用户文件夹下)

```
git config --global --unset http.sslCAInfo
git config --global http.sslCAInfo "D:/Program Files/Git/mingw64/ssl/certs/ca-bundle.crt"
```

2. git基础

- 按照教材完成了练习, 首先配置了用户名和邮箱地址, 设置了每个项目的主分支为main
- 克隆自己在github创建的仓库

```
git clone https://github.com/xBoltgun/python_class.git
```

- 创建一个简易的python程序hello.py

```
print("hello git world!")
print("hello everyone.")
```

- 将文件加入仓库

```
git add.
```

- 上传

```
git commit -m "666"
```

3. git练习

访问learngitbranching.js.org, 完成Main部分的Introduction Sequence和Ramping Up两个小节的学习。练习了如何提交, 跳转分支和撤回提交

4. markdown基础

访问[Markdown cheat-sheet](#)来帮助我用markdown撰写实验报告

使用Markdown编辑器VScode编写本次实验的实验报告, 包括实验过程与结果、实验考查和实验总结, 并将其导出为 PDF格式 来提交。

实验考察

1. 什么是版本控制？使用Git作为版本控制软件有什么优点？
 - 版本控制（Revision control）是一种在开发的过程中用于管理我们对文件、目录或工程等内容的修改历史方便查看更改历史记录，备份以便恢复以前的版本的软件工程技术。
 - 没有中央服务器，每个人的电脑就是一个完整的版本库
 - 工作的时候不需要联网了，因为版本都在自己电脑上。
2. 如何使用Git撤销还没有Commit的修改？如何使用Git检出（Checkout）已经以前的Commit？（实际操作）

```
git merge --abort  
  
git checkout -b (id)
```

3. Git中的HEAD是什么？如何让HEAD处于detached HEAD状态？（实际操作）
 - git中HEAD是指向当前所在分支的指针

```
git checkout ^^
```

4. 什么是分支（Branch）？如何创建分支？如何切换分支？（实际操作）
 - 分支指的是从主线上分离出来进行另外的操作

```
git branch newImage  
git checkout newImage
```

5. 如何合并分支？git merge和git rebase的区别在哪里？（实际操作）
 - 使用git merge 或者 git rebase
 - git rebase会把当前分支的 commit 放到公共分支的最后面,所以叫变基。就好像从公共分支又重新拉出来这个分支一样。
 - git merge会把公共分支和你当前的commit 合并在一起，形成一个新的 commit 提交
6. 如何在Markdown格式的文本中使用标题、数字列表、无序列表和超链接？（实际操作）

使用标题：

1. 数字列表：
 - 无序列表

[超链接](#)