

系统基础开发工具

实验一

姓 名: 陈 佳 玲

学 号: 23100021002

专 业: 23 级环境工程

2025 年 9 月 4 日

目录

1 实验目的	2
2 实验内容	2
2.1 Git	2
2.1.1 创建本地库并初始化	2
2.1.2 创建新文件	2
2.1.3 检查当前文件的状态	3
2.1.4 将文件添加到暂存区	3
2.1.5 提交更新到本地仓库	3
2.1.6 查看提交历史	4
2.1.7 创建分支	4
2.1.8 切换分支	4
2.1.9 合并分支	4
2.1.10 删除文件	5
2.2 LaTeX	6
2.2.1 文档的结构	6
2.2.2 文章的标题、作者、日期	6
2.2.3 文章结构划分	6
2.2.4 文字样式	6
2.2.5 文本内部换行	6
2.2.6 数学公式	6
2.2.7 创建有序表	6
2.2.8 注释添加	7
2.2.9 特殊符号输入	7
2.2.10 目录生成	7
3 心得体会	7
4 实验代码查看链接	7

1 实验目的

1. 学习版本控制 (Git) 通过课堂学习和课下探索练习, 能够掌握基本的 Git 指令, 并且能够对 git 产生新的认识。
2. 学习 LaTeX 文档编辑学习 LaTeX 文档编辑器, 练习相关的代码编写, 掌握 LaTeX 的排版方法, 能够学以致用, 用 LaTeX 来生成自己的实验报告。

2 实验内容

2.1 Git

2.1.1 创建本地库并初始化

- (1) 创建本地库

```
71883@MINGW64 ~  
$ mkdir chen
```

图 1: 创建仓库

- (2) 进入目录并初始化仓库

```
71883@MINGW64 ~ (master)  
$ cd chen  
  
71883@MINGW64 ~/chen (master)  
$ git init  
Initialized empty Git repository in C:/Users/71883/chen/.git/
```

图 2: 初始化仓库

2.1.2 创建新文件

查看工作区和暂存区的状态, 输出结果为状态报告。

```
71883@MINGW64 ~/chen (master)  
$ echo 111111 > 1_file.txt  
  
71883@MINGW64 ~/chen (master)  
$ git status  
On branch master  
Untracked files:  
  (use "git add <file>..." to include in what will be committed)  
    1_file.txt  
    newrepo/  
  
nothing added to commit but untracked files present (use "git add" to track)  
71883@MINGW64 ~/chen (master)
```

图 3: 创建新文件并添加内容

2.1.3 检查当前文件的状态

查看工作区和暂存区的状态, 输出结果为状态报告。

```
71883@MINGW64 ~/chen (master)
$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    1_file.txt
    newrepo/

nothing added to commit but untracked files present (use "git add" to track)
71883@MINGW64 ~/chen (master)
```

图 4: 查看文件状态

2.1.4 将文件添加到暂存区

(1) 将工作区的文件更改添加到暂存区, 准备下次提交

```
71883@MINGW64 ~/chen (master)
$ git add .
warning: in the working copy of '1_file.txt', LF will be replaced by CRLF the next time Git touches it
```

图 5: 添加文件到暂存区

(2) 再次检查状态

```
71883@MINGW64 ~/chen (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   1_file.txt
```

图 6: 验证暂存区添加结果

2.1.5 提交更新到本地仓库

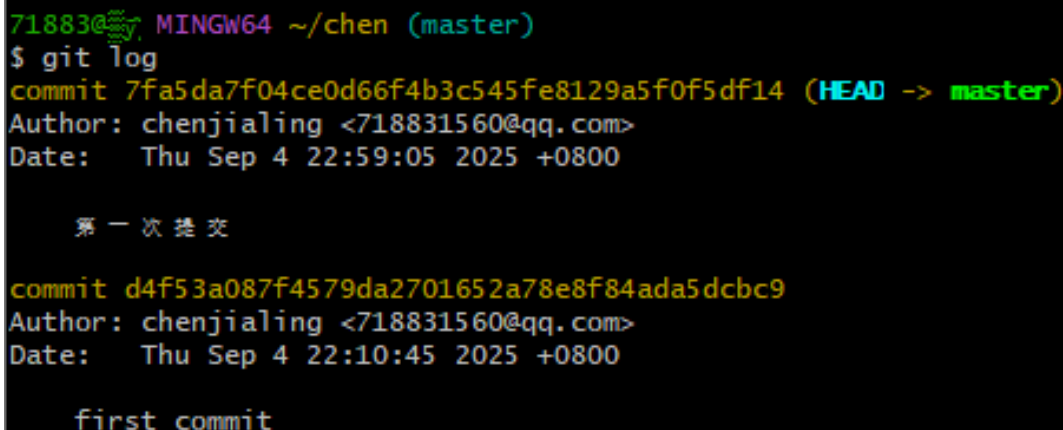
将暂存区的文件提交到版本历史中:

```
71883@MINGW64 ~/chen (master)
$ git commit -m "第一次提交"
[master 7fa5da7] 第一次提交
1 file changed, 1 insertion(+)
create mode 100644 1_file.txt
```

图 7: 提交更改

2.1.6 查看提交历史

验证提交已成功创建, 显示提交历史, 包括作者、日期和提交消息



```
71883@MINGW64 ~/chen (master)
$ git log
commit 7fa5da7f04ce0d66f4b3c545fe8129a5f0f5df14 (HEAD -> master)
Author: chenjialing <718831560@qq.com>
Date: Thu Sep 4 22:59:05 2025 +0800

    第一次提交

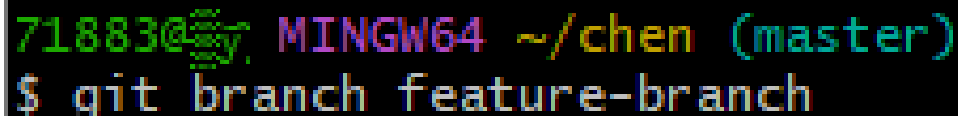
commit d4f53a087f4579da2701652a78e8f84ada5dcbc9
Author: chenjialing <718831560@qq.com>
Date: Thu Sep 4 22:10:45 2025 +0800

    first commit
```

图 8: 验证提交

2.1.7 创建分支

使用指令“git branch feature-branch”创建分支, 分支是指向特定提交的轻量级指针; 创建分支不会复制文件, 只是创建一个新指针; 使用分支可以在不影响主线的前提下开发新功能。

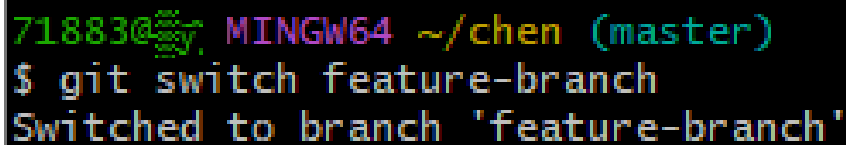


```
71883@MINGW64 ~/chen (master)
$ git branch feature-branch
```

图 9: 创建分支

2.1.8 切换分支

使用指令“git switch feature-branch”切换分支, 更新工作目录中的文件



```
71883@MINGW64 ~/chen (master)
$ git switch feature-branch
Switched to branch 'feature-branch'
```

图 10: 切换分支

2.1.9 合并分支

使用 git branch 指令查看分支之后, 使用指令 git merge feature-branch 合并分支

```
71883@SY MINGW64 ~/chen (feature-branch)
$ git branch
* feature-branch
  master

71883@SY MINGW64 ~/chen (feature-branch)
$ git switch master
Switched to branch 'master'

71883@SY MINGW64 ~/chen (master)
$ git merge feature-branch
Already up to date.
```

图 11: 合并分支

2.1.10 删除文件

使用 `git rm newrepo.txt` 指令删除文件，并使用 `git status` 查看删除结果。

```
71883@SY MINGW64 ~/chen (master)
$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    1_file.txt
    newrepo/

nothing added to commit but untracked files present (use "git add" to track)

71883@SY MINGW64 ~/chen (master)
$ git add .
warning: in the working copy of '1_file.txt', LF will be replaced by CRLF the next time Git touches it
error: 'newrepo/' does not have a commit checked out
fatal: adding files failed

71883@SY MINGW64 ~/chen (master)
$ rm newrepo
rm: cannot remove 'newrepo': Is a directory

71883@SY MINGW64 ~/chen (master)
$ rm -rf newrepo

71883@SY MINGW64 ~/chen (master)
$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    1_file.txt

nothing added to commit but untracked files present (use "git add" to track)
```

图 12: 删除文件

2.2 LaTeX

2.2.1 文档的结构

(1) 文档的类型定义: `\documentclass{}` 是 LaTeX 文档的第一个命令。它控制了文档的页边距、字体大小等。文档的类型有 `article` (文章) `report` (报告), `book` (书籍), `letter` (信函), `beamer` (幻灯片)

(2) 标记文档正文内容的开始和结束: `\begin{document}` 和 `\end{document}`

2.2.2 文章的标题、作者、日期

(1) 标题 `\title{系统基础开发工具}`

(2) 作者 `\author{陈佳玲}`

(3) 日期 `\date{\today}`: 能够生成当天的日期

(5) 标题栏 `\maketitle`

2.2.3 文章结构划分

章节标题 `\section{}`, `\subsection{}`, `\subsubsection{}`

2.2.4 文字样式

(1) 粗体 `\textbf{关键点}` 这是**关键点** (2) 斜体 `\textit{特别注意}` 需要特别注意

2.2.5 文本内部换行

在段落内部进行手动换行使用 `\\` 或 `\newline`

2.2.6 数学公式

在文本行中插入简洁的数学公式或符号, 使用 `$...$`, 例如 `$y=2x+1$`。若要凸显数学公式则使用 `[...]`, 例如 `[x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}]`,

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

., 很多数学公式的实现还需要引入 `amsmath` 这个宏包, 实现方法是在导言区添加 `\usepackage{amsmath}`

2.2.7 创建有序表

使用 `enumerate` 生成带自动编号的列表, 用于需要表示顺序的步骤或条目。如

`\begin{enumerate}`

`\item 第一步`

`\item 第二步`

`\end{enumerate}` | 生成结果

1. 第一步

2. 第二步

2.2.8 注释添加

使用%在代码中添加不会被编译和输出的注释,如输出这是一个普通句子。% 这是关于上一句的注释,结果为这是一个普通句子。

2.2.9 特殊符号输入

使用反斜杠\ 转义输出许多特殊字符,如#等保留为命令符号。

2.2.10 目录生成

在 LaTeX 中为了自动生成目录,我们需要在正文区添加\tableofcontents,通常,为了让目录和后面的正文内容不在同一页,我们可以加上\newpage来实现另起一页

3 心得体会

通过学习 Git,我掌握了版本控制的核心概念与基础操作。实践中的问题排查深化了我对 workflow 原理的理解,这项技能不仅提升了我的代码管理效率,更培养了我严谨协作的习惯。LaTeX 的学习让我体验到“内容与格式分离”的专业排版。从文档结构构建到数学公式、表格的精准处理,我深刻感受到其自动化排版与交叉引用的强大优势。将它应用于实验报告撰写后,我彻底解放了格式调整的负担,能更专注于内容本身,使得文档格式更加美观标准。

4 实验代码查看链接

本次报告相关练习、报告和代码均可以在[查看](#)