

系统开发工具基础实验报告

实验内容：实验一

姓名：张家宜 学号：2024020013045

日期：2025 年 9 月 9 日

目录

1	练习内容	2
1.1	Latex 文档编辑	2
1.1.1	<code>\documentclass</code>	2
1.1.2	<code>\begin & \end</code>	2
1.1.3	<code>\date</code>	2
1.1.4	<code>\title</code>	2
1.1.5	章节	2
1.1.6	创建标签	3
1.1.7	生成目录	3
1.1.8	中文字体支持	3
1.1.9	字体效果	3
1.1.10	彩色字体	4
1.1.11	字体大小	4
1.1.12	段落缩进	4
1.1.13	列表	4
1.1.14	注释和空格	5
1.1.15	特殊字符	5
1.1.16	表格	5
1.1.17	图表	7
1.1.18	公式	8
1.2	版本控制 (Git)	8
1.2.1	基础操作	8
1.2.2	查看历史与差异	9
1.2.3	分支与合并	9
1.2.4	远端操作	9
1.2.5	撤销与修改	10
1.2.6	Git 的课后练习	10
2	解题感悟	12

1 练习内容

1.1 Latex 文档编辑

1.1.1 `\documentclass`

```
1 \documentclass[a4paper, 12pt]{article}
2
3 \begin{document}
4   A sentence of text.
5 \end{document}
```

`\documentclass` 命令必须出现在每个 LaTeX 文档的开头。花括号内的文本指定了文档的类型。

1.1.2 `\begin & \end`

`\begin{document}` 和 `\end{document}` 可以将文本内容包裹起来

1.1.3 `\date`

`\date{}` 可以在 title 里面显示时间

使用 `\today` 可以显示当前的日期

2025 年 9 月 9 日

1.1.4 `\title`

```
1 \title{My First Document}
2 \author{My Name}
3 \date{\today}
4 \maketitle
```

`\maketitle` 命令可以给文档创建标题。

1.1.5 章节

```
1 \section{Introduction}
2 This is the introduction.
3
4 \section{Methods}
5
```

```

6 \subsection{Stage 1}
7 The first part of the methods.
8
9 \subsection{Stage 2}
10 The second part of the methods.
11
12 \section{Results}
13 Here are my results.

```

1.1.6 创建标签

我在上一章里面添加了`\label{test}`，这里就可以使用`\red`和`\pageref`引用
Here are my results. Referring to section 1.1.5 on page 2

1.1.7 生成目录

```

1 \pagenumbering{roman}
2 \tableofcontents
3 \newpage
4 \pagenumbering{arabic}

```

使用`\newpage`会另起一页

1.1.8 中文字体支持

只需要在文档的前导命令部分添加：`\usepackage[UTF8]{ctex}`
在 overleaf 里面还需要把文档的 Compiler 设置成 XeLaTeX

1.1.9 字体效果

```

1 \textit{words in italics} \textsl{words slanted} \textsc{words in
   smallcaps} \textbf{words
2 in bold} \texttt{words in teletype} \textsf{sans serif words} \textrm
   {roman
3 words} \underline{underlined words}

```

效果如下：

words in italics *words slanted* WORDS IN SMALLCAPS **words in bold** words in
teletype sans serif words roman words underlined words

1.1.10 彩色字体

`{\color{colname}text}`

其中 colname 是颜色的名字，text 是彩色文本内容。

fire

PS: 需要在 `\begin{document}` 前输入 `\usepackage{color}`

1.1.11 字体大小

```

1 normal size words {\tiny tiny words} {\scriptsize scriptsize words}
2 {\footnotesize footnotesize words} {\small small words} {\large large
   words}
3 {\Large Large words} {\LARGE LARGE words} {\huge huge words}

```

效果如下:

normal size words tiny words scriptsize words footnotesize words small words large words
Large words LARGE words huge words

1.1.12 段落缩进

如果想要段落顶格，在要顶格的段落前加 `\noindent` 命令即可。如果希望全局所有段落都顶格，在文档的某一位置使用 `\setlength{\parindent}{0pt}` 命令，之后的所有段落都会顶格。

1.1.13 列表

生成一个有序列表套无序列表

```

1 \begin{enumerate}
2   \item First thing
3   \item Second thing
4     \begin{itemize}
5       \item A sub-thing
6       \item Another sub-thing
7       \item[+] 加号
8       \item[*] 星号
9     \end{itemize}
10   \item Third thing
11 \end{enumerate}

```

1. First thing

2. Second thing

- A sub-thing
- Another sub-thing

+ 加号

* 星号

3. Third thing

`\item[-]` 会使用一个杠作为标志，你甚至可以使用一个单词，比如 `\item[One]`

1.1.14 注释和空格

使用`%` 创建一个单行注释，在这个字符之后的该行上的内容都会被忽略，直到下一行开始。

多个连续空格在 LaTeX 中被视为一个空格。多个连续空行被视为一个空行。空行的主要功能是开始一个新的段落。通常来说，LaTeX 忽略空行和其他空白字符，两个反斜杠 (`\\`) 可以被用来换行。

1.1.15 特殊字符

在他们前面添加反斜杠进行转义 `\# \$ \% \^{} \& _ \{ \} \sim`

`# $ % ^ & _ { } ~`

反斜杠使用`\textbackslash` 命令代替。

1.1.16 表格

Apples	Green
Strawberries	Red
Orange	Orange

Apples	Green
Strawberries	Red
Oranges	Orange

8	here's
86	stuff
2008	now

尝试画出表格

Item	Quantity	Price(\$)
Nail	500	0.34
Wooden boards	100	4.00
Bricks	240	11.50

City	Year		
	2006	2007	2008
London	21972	29713	29741
Berlin	14102	28741	29104
Paris	2919	291	2914

1.1.17 图表



图 1: Here is my image

1.1.18 公式

$$1 + 2 = 3$$

$$1 + 2 = 3 \tag{1}$$

$$a = b + c \tag{2}$$

$$= y - z \tag{3}$$

$$e^2$$

$$n_1$$

$$\frac{2}{3}$$

$$\sqrt{199}$$

$$\sum_{x=1}^5 y^z$$

$$\int_a^b f(x)$$

$$\delta, \Delta$$

1.2 版本控制 (Git)

1.2.1 基础操作

- `git help <command>`
查看某个 git 命令的帮助信息
- `git init`
初始化一个新的仓库（生成 `.git` 目录）
- `git status`
查看当前仓库状态（修改/暂存情况）
- `git add <file>`
将文件加入暂存区
- `git commit`
提交暂存区内容（注意撰写良好的提交信息）

1.2.2 查看历史与差异

- `git log`
查看提交历史
- `git log --all --graph --decorate`
图形化显示提交与分支
- `git diff <file>`
比较工作区与暂存区差异
- `git diff <rev> <file>`
比较某文件在两个版本间的差异
- `git checkout <rev>`
切换到某个提交或分支

1.2.3 分支与合并

- `git branch`
查看分支列表
- `git branch <name>`
新建分支
- `git checkout -b <name>`
新建并切换分支
- `git merge <rev>`
合并指定提交/分支
- `git mergetool`
使用工具解决冲突
- `git rebase`
变基操作（重新应用提交）

1.2.4 远端操作

- `git remote`
查看远端仓库

- `git remote add <name> <url>`
添加远端仓库
- `git push <remote> <local>:<remote>`
推送本地分支到远端
- `git branch --set-upstream-to=<remote>/<branch>`
建立本地与远端分支的关联
- `git fetch`
获取远端更新（不合并）
- `git pull`
获取并合并远端更新（= fetch + merge）
- `git clone <url>`
克隆远端仓库

1.2.5 撤销与修改

- `git commit --amend`
修改最近一次提交
- `git reset`

命令	效果
<code>git reset --soft <commit></code>	撤销提交，但保留暂存区和工作区的改动
<code>git reset --mixed <commit></code>	清空暂存区，保留工作区改动（默认模式）
<code>git reset --hard <commit></code>	提交、暂存区和工作区全部重置（不可恢复）

表 1: git reset 的三种模式对比

- `git checkout -- <file>`
丢弃工作区修改

1.2.6 Git 的课后练习

2.1 `git log --all --graph --decorate`

2.2 `git log -1 README.md`

2.3 git blame __config.yml | grep collections

3

```
1 git filter-branch --force \  
2   --index-filter 'git rm --cached --ignore-unmatch ./my_password  
3   ' \  
4   --prune-empty \  
   --tag-name-filter cat -- --all
```

filter-branch 用来改写仓库的历史

`--index-filter 'git rm --cached --ignore-unmatch ./my_password'`

`index-filter` 会在历史的每一次提交里执行你指定的命令。

`--prune-empty` 有些提交可能唯一的改动就是那个密码文件。删掉后就变成“空提交”了, 这个选项会自动删除这些空提交。

`--tag-name-filter cat` 保证 tag 还能继续指向正确的 commit

`--all` 表示要作用于所有分支和所有 tag

4 stash 就像一个临时抽屉, 把改动存起来, 工作区会恢复到干净状态

2 解题感悟

L^AT_EX 是一款非常好用的排版工具。许多在传统编辑器中需要反复手动调整的格式与字体，在 L^AT_EX 中常常只需改动几行代码即可完成；而且一旦抽象成模板，后续便能复用，事半功倍。对于在其他编辑器（如 Word）中较为麻烦的公式、表格等内容，只要掌握了基本方法，在 L^AT_EX 里就能高效而规范地录入。

在项目开发中，Git 十分实用：它能有效进行版本控制，避免出现重大问题时无从挽回，同时也便利了调试与团队协作。正如 [1] 所说：“尽管 Git 的接口有些丑陋，但是它的底层设计和思想却是非常优雅的。”

参考文献

- [1] Missing Semester 中文版, <https://missing-semester-cn.github.io/>
- [2] OI-Wiki \LaTeX 工具, <https://oi-wiki.org/tools/latex/>

GitHub 链接

<https://github.com/Misasasasaka/report/tree/main/P1>