

Latex与Git

23020007030 关嘉琪

2024 年 9 月 13 日

1 实验目的

本次课程主要讲授了版本控制（Git）以及Latex文档编辑，通过对两者的学习来加强对两大便捷工具的使用

2 介绍

2.1 两大工具的优点

Git是一个开源的分布式版本控制系统，可以有效、高速地处理从很小到非常大的项目版本管理。也是Linus Torvalds为了帮助管理Linux内核开发而开发的一个开放源码的版本控制软件。

LaTeX原名TeX，是一种基于TeX的排版系统，用于生成专业排版的高质量文档。它提供了对复杂的数学公式、ID标签和脚注等的强大支持，被广泛应用于学术界、出版业和其他需要精确排版的领域。

通过 Git 和 LaTeX，可以自动化文档的构建和部署过程，确保文档的一致性和准确性。

3 实验内容

3.1 Latex学习例子10个

1. `\chapter{}` 章节题目

`\section{}` 标题

`\subsection{}` 小部分

`\subsubsection{}`更小的部分 从上往下层级依次细化

2. `\verb`命令里面——里面可以放入想表示的指令，这样它就会以文本的方式输出——

3. `begin[]`和`end[]`可以构成环境，在里面可以编写内容。

`begin[itemize]`和`end[itemize]`构成无序列表，`begin[enumerate]`和`end[enumerate]`构成有序列表

下面是例子：

1. 有序列表 //有序列表，指有序号

- 无序列表 //无序列表，指无序号

4.\includegraphics[width=\textwidth]{}该命令可以用来引入图片



图 1: 这是第二个例子的图片。

5.\newline的功能是换行，可以使用 \newline 命令来实现换行。这个命令会将当前位置设置为新的一行

6.\usepackage{}可以用来引入宏包或者设置字体编码下面是几个例子

```
\usepackage[utf8]{inputenc} % 设置输入编码
\usepackage[T1]{fontenc}    % 设置字体编码
\usepackage{graphicx}       % 插入图片
\usepackage{amsmath}        % 数学公式
\usepackage{amssymb}        % 数学符号
\usepackage{amssymb}        % 数学符号
\usepackage{hyperref}       % 超链接
```

7.创建表格的命令\hline

Column 1	Column 2	Column 3
Left	Center	Right

8. \footnote{}可以添加脚注

This is a text with a footnote¹.

9. \textbf{}是加粗，\textit{}是倾斜，\underline{}是加下划线

This is **textbf**, this is *textit*, and this is underlined.

10.\title是加题目My Simple Document

\author是加作者Jane Doe

\date是加日期—2024 年 9 月 13 日

¹This is the footnote.

3.2 Git学习例子10个

- 1.初始化新仓库: `git init`

```
$ git init
Initialized empty Git repository in D:/Latex论文/.git/
```

图 2: 初始化新仓库

- 2.添加文件: `git add .`

```
$ git add .
warning: in the working copy of 'latex.aux', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'latex.log', LF will be replaced by CRLF the next time Git touches it
```

图 3: 添加文件

- 3.将你的 LaTeX 源文件添加到 Git 仓库: `git commit -m "Initial commit of LaTeX project"`

```
$ git commit -m "Initial commit of LaTeX project"
[master (root-commit) 5d2753f] Initial commit of LaTeX project
6 files changed, 587 insertions(+)
create mode 100644 latex.aux
create mode 100644 latex.log
create mode 100644 latex.pdf
create mode 100644 latex.synctex.gz
create mode 100644 latex.tex
create mode 100644 "\\345\\233\\276\\347\\211\\207.png"
```

图 4: 将Latex源文件添加到Git仓库

- 4.查看工作目录和暂存区的状态:`git status`

```
$ git status
On branch main
nothing to commit, working tree clean
```

图 5: 查看工作目录和暂存区状态

- 5.在处理大型文档或尝试新功能时, 可以创建分支来隔离开发工作。 `git checkout -b feature-branch`

```
$ git checkout -b feature-branch
Switched to a new branch 'feature-branch'
```

图 6: 创建分支

- 6.列出所有分支: `git branch`

```
$ git branch
* main
master
```

图 7: 初始化新仓库

- 7.展示历来提交版本: `git log`

```
$ git log
commit fe42975c9ce1ef4485905d16ca00627b66365e60 (HEAD -> main)
Author: z <724810122@qq.com>
Date: Thu Sep 5 16:00:34 2024 +0800

    first commit

commit 3a51538cb04b7da0319d76de284c07c33b22425b
Author: z <724810122@qq.com>
Date: Thu Sep 5 15:46:12 2024 +0800

    Descriptive commit message

commit 5d2753fbc6c5981f8a7b6bbb34beca5550326701 (master)
Author: z <724810122@qq.com>
Date: Thu Sep 5 14:42:52 2024 +0800

    Initial commit of LaTeX project
```

图 8: 展示历来提交版本

- 8.打开任意版本: `git show hash(哈希值)`

