

# 系统开发工具基础 (第一次实验报告)

程乾楠 24020007014

August 31, 2025

# 1. 关于 Latex 的使用

## 1.0 Latex 的界面的介绍

overleaf 可以提供网页端的 Latex 编辑器的使用，本实验就是在 overleaf 的帮助下完成的，欢迎提出指正。

## 1.1 Latex 中的转义

在使用 Latex 进行编辑时，`\documentclass{article}` 这一行代码是声明文档的类型，在“`\begin{document}`”和“`\end{document}`”之间的就是文档的内容。如果想要使文本内容出现一些代码的文本，就可以使用转义。  
→ 常见的转义方法有`\begin{verbatim}`，`end{verbatim}`。`\texttt{}`包裹命令，例如`\texttt{\textbackslash documentclass\{article\}}`。

## 1.2 Latex 中如何使用中文

要在声明文档类型后使用 `xeCJK` 这个包，使用的代码如下：

```
\usepackage{xeCJK}
```

而且要在 Menu 中改变编译器为 XeLatex，然后就可以使用中文了。

## 1.3 换行符的使用

在 Latex 中，换行操作可以使用：

```
\newline 或者\\
```

它们不会触发缩进，但是使用两段文字中间空一行的操作，那样就会判定为两段，会触发缩进。

## 1.4 换页符的使用

在 Latex 中，换页符可以使用：

`\newpage`，功能是从当前位置强制开始新的一页。而`\clearpage`，但是后者除了换页，还会把所有未排版的浮动体（figure/table）先输出。

## 1.5 居中对齐

在写左右居中对齐时，可以用：居中对齐：

将想要居中对齐的内容放到：

```
\begin{center} 和 \end{center} 中
```

## 1.6 居左居右对齐

同 1.5 一样，将 center 换为 flushright,flushleft 就可以了。

## 1.7 插入图片（固定位置）

在 Latex 固定位置中插入图片：

在固定位置插入图片，需要先新建文件夹存放图片，然后上传图片，需要加一

↪ 个 `\usepackage{graphicx}`

之后设置图片搜索路径：

比如 `\graphicspath{{./picture/}}`，`graphicspath` 可以指定图片文件夹。

然后写入以下代码：

`\includegraphics[width=0.6\textwidth]{图片的文件名}`

或者 `\includegraphics[width=n cm,height=m cm]{图片的文件名}`

`n` 和 `m` 为小于页面的宽度和长度的数字。

完成这些就可以了。

展示效果：



## 1.8 插入图片（浮动位置）

在 Latex 浮动位置中插入图片：

在以下代码的基础上：

```
\includegraphics[width=0.6\textwidth]{图片的文件名}
```

前后加上`\begin{figure}` 和`\end{figure}` 就可以了，其代表含义为图片出

→ 现的位置不一定在代码出现的位置上，而是在这个位置附近浮动

## 1.9 英文字体改变

使用 Hello,world 作为例子：

```
\textbf{Hello,world}\par
\underline{Hello, world}\par
\textit{Hello, world}\par
{\small Hello, world}\par
{\large Hello, world}\par
{\Large Hello, world}\par
{\LARGE Hello, world}\par
{\huge Hello, world}\par
{\Huge Hello, world}\par
```

**Hello,world**  
Hello, world  
*Hello, world*  
Hello, world  
Hello, world  
Hello, world  
Hello, world  
Hello, world  
Hello, world  
Hello, world

## 1.10 中文字体改变

**你好，世界**  
你好，世界  
*你好，世界*  
你好，世界  
你好，世界  
你好，世界  
你好，世界  
你好，世界  
你好，世界  
你好，世界

## 1.11 插入表格

使用：

| 列 1  | 列 2  | 列 3  |
|------|------|------|
| 内容 1 | 内容 2 | 内容 3 |
| 内容 4 | 内容 5 | 内容 6 |

Table 1: 简单表格示例

## 1.12 图片加标题

Figure 1: 鸣潮角色图片



## 1.12 图片引用

Figure 2: 鸣潮角色图片



如图 2 所示，这是一张示例图片。

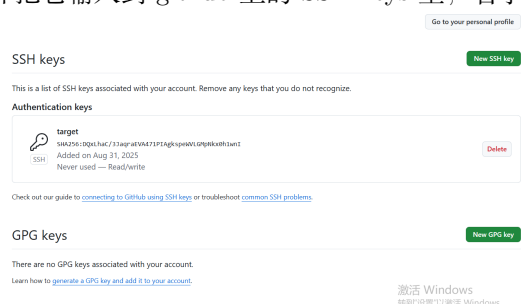
## 2.git 的使用

### 2.1 在 VS Code 里配置 Git 的用户名和邮箱，让它和 GitHub 对应上

git config --global user.name "你的 GitHub 用户名" git config --global user.email "你的 GitHub 邮箱" 使用这两行指令，可以用 VS Code 里配置 Git 的用户名和邮箱，user.name 建议和 GitHub 上的用户名一致。user.email 必须是 GitHub 账号绑定的邮箱，否则 GitHub 可能不会显示你是这次提交的作者。

### 2.2 使用 SSH key 的方式把本地 Git 和 GitHub 连接起来

通过 git bash 输入这行指令 ssh-keygen -t rsa，之后会显示储存位置，找到公钥，并把它输入到 github 上的 SSH keys 里，名字可以随便取。



这里是弄好的截图。

### 0.1 2.3 给 Git 设置代理

通过以下两行指令设置代理：git config --global http.proxy http://127.0.0.1: 代理的端口 git config --global https.proxy http://127.0.0.1: 代理的端口

### 0.2 2.4 将 GitHub 上的仓库克隆到本地文件夹中

使用 HTTPS 方式克隆：先打开想要克隆到的文件夹，在里面打开 git bash，然后输入：git clone https://github.com/yourname/myrepo.git

### 2.5 查看当前文件状态

通过 git status 可以完成。

```
MINGW64/c/Users/cheng/Desktop/agf
cheng@DESKTOP-GDU885Q MINGW64 ~/Desktop/agf (master)
$ git clone https://github.com/target107/git666.git
Cloning into 'git666'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 3 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (3/3), done.

cheng@DESKTOP-GDU885Q MINGW64 ~/Desktop/agf (master)
$ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    git666/

nothing added to commit but untracked files present (use "git add" to track)
cheng@DESKTOP-GDU885Q MINGW64 ~/Desktop/agf (master)
$ \
```

## 2.6 将文件上传到 github

先添加到暂存区，通过 `git add filename`. 后缀来进行添加。

```
MINGW64/c/Users/cheng/Desktop/agf
cheng@DESKTOP-GDU885Q MINGW64 ~/Desktop/agf (master)
$ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    git666/

nothing added to commit but untracked files present (use "git add" to track)
cheng@DESKTOP-GDU885Q MINGW64 ~/Desktop/agf (master)
$ git sdd first.txt
git: 'sdd' is not a git command. See 'git --help'.

The most similar command is
  add

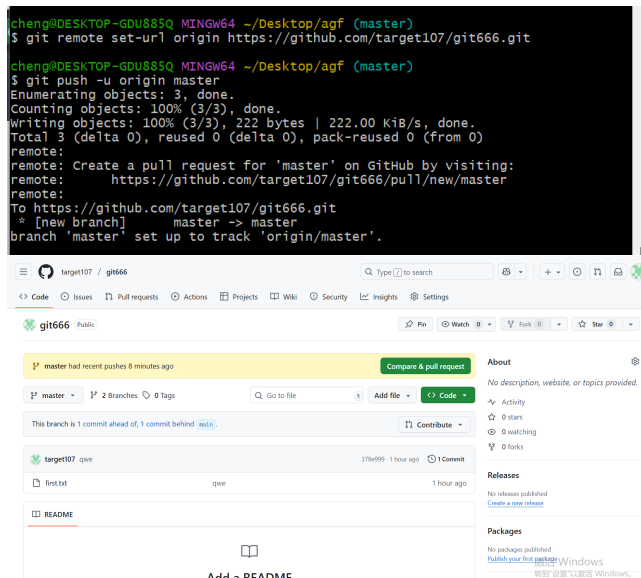
cheng@DESKTOP-GDU885Q MINGW64 ~/Desktop/agf (master)
$ git add first.txt
cheng@DESKTOP-GDU885Q MINGW64 ~/Desktop/agf (master)
$ |
```

再提交到本地仓库：`git commit -m "本地仓库名"`

然后通过 HTTPS 连上远程仓库：本人最开始把通过 SSH 连接远程仓库的地址格式弄错了，弄成这样了：

`git@https://github.com/target107/git666.git`

导致我后来又修改了 origin 指向的地址，`git remote set-url origin https://github.com/target107/git666.git` 之后才推送成功的：`git push -u origin master`



### 3. 实验后的反思

#### 3.1 实验中存在的问题

在使用转义时，没有考虑到 `verbatim` 不会换行的问题，使用 `ai` 提出的方法时，没有考虑到自动换行是加强内容，要使用 `fvextra` 包。在学习 `git` 时遇到的问题很多，什么操作也不会，也不知道是干什么的，后来的学习才认识到 `git` 的作用，就比如 `SSH` 连接远程仓库，最后发现密钥用不了，试验结束后再看看能不能用 `SSH` 连接到 `github` 仓库。

#### 3.2 实验后的反思

在学习时，一定要举一反三，学习时要灵活多变，边学边实践才是最正确的学习方法