

LateX 入门

2025 年 3 月 5 日

1 LateX 是以环境为主的

在 VSCode 中, Ctrl + Alt + B 是编译成 PDF 的快捷键.

1.1 居中显示

使用 center 来创建居中环境, 注意, 反斜杠符号是特殊符号, 在 LateX 里是告诉编译器: 我要开始一条命令了, 所以直接输入反斜杠是不能显示的, 要转义, 转义的方法是输入三个斜杠

文字加粗, 文字斜体, 文字下划线, 代码用等宽字体: This is some code

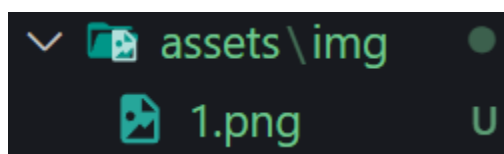


图 1: 示例图片标题

要注意一下怎么对路径进行引用。像这个项目里, 结构如下:

tex 文件是你编辑的文件, 要找到你的图片, 向上返回一层后, 到 report, 再向上返回一层, 到 project 的根目录, 然后进入 assets, 再进入 img, 然后找到了 1.png, 所以要用../assets/img/1.png

这里是引用图片的例子: 如图 1 所示。

1.2 子标题

写点东西, 再写点东西, 再写点。

1.2.1 子子标题

2 表格

如表格 1 所示

A	B	C	D
1	2	3	
4	5	6	
7	8	9	

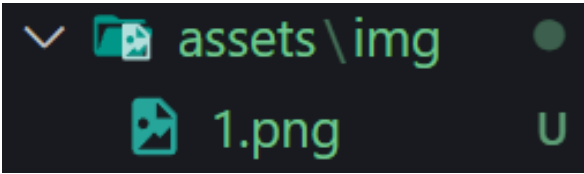
表 1: 示例表格

如表格 2 所示.

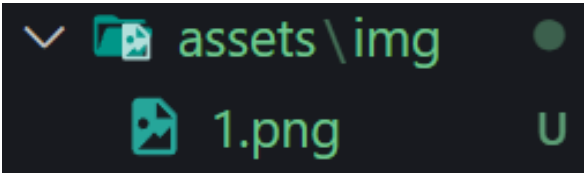
A	B	C	D
1	2	3	
4	5	6	
7	8	9	

表 2: 示例表格 2

2.1 子图包



(a) 子图 1



(b) 子图 2

图 2: 示例子图

6666

7777

2.2 分页

8888

2.3 分栏

3 基础介绍

3.1 分栏优势

分栏排版常见于学术期刊和会议论文集，具有以下优点：

- 提高页面利用率
- 增强可读性
- 方便图表与文字对照
- 适应不同阅读场景

1. 提高页面 $x^2 + y^2 = z^2$ 利用率

2. 增强可读性

3. 方便图表与文字对照

4. 适应不同阅读场景

3.2 数学公式示例

洛伦兹方程：

$$\begin{cases} \frac{dx}{dt} = \sigma(y - x) \\ \frac{dy}{dt} = x(\rho - z) - y \\ \frac{dz}{dt} = xy - \beta z \end{cases} \quad (1)$$

麦克斯韦方程组：

$$\begin{aligned} \nabla \cdot \mathbf{E} &= \frac{\rho}{\varepsilon_0} \\ \nabla \cdot \mathbf{B} &= 0 \\ \nabla \times \mathbf{E} &= -\frac{\partial \mathbf{B}}{\partial t} \\ \nabla \times \mathbf{B} &= \mu_0 \left(\mathbf{J} + \varepsilon_0 \frac{\partial \mathbf{E}}{\partial t} \right) \end{aligned}$$

4 详细内容

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec

vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

4.1 图片示例

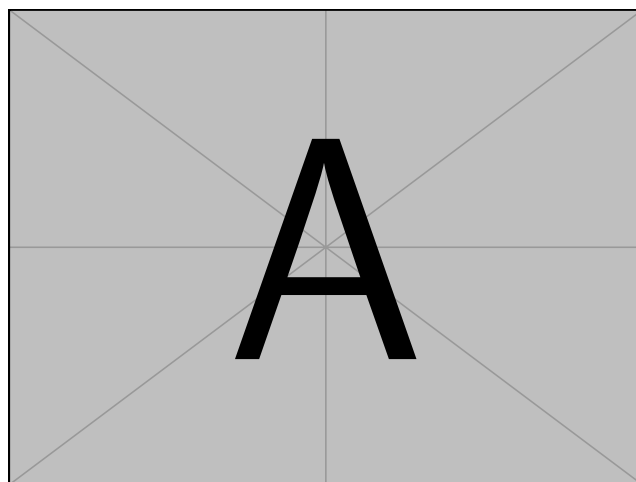


图 3: 第一栏中的示例图片

5 第二栏内容

5.1 定理环境

```
\begin{theorem}
```

任意大于2的偶数可以表示为两个素数之和

```
\end{theorem}
```

5.2 代码示例

```
def fibonacci(n):  
    a, b = 0, 1  
    for _ in range(n):  
        yield a  
        a, b = b, a + b
```

5.3 颜色文本

红色文本、蓝色文本和绿色文本示例

5.4 引用文献

以下是文献引用示例：

1. 文献 [key1]
2. 文献 [key2]
3. 文献 [key3]

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam.

Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetur adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetur.

6 代码块