

# Typst 简明使用教程

卓能文

2023-11-29

## 目 录

1 Typst 简介 .....	1
2 Typst 安装 .....	1
3 Typst 使用 .....	1
3.1 创建文件 .....	1
3.2 章节设置 .....	1
3.3 显示图片 .....	2
3.3.1 设置宽度: .....	2
3.3.2 居中显示: .....	2
3.3.3 设置标题: .....	3
3.3.4 多图并列 .....	3
3.3.5 多图并列 (带标题) .....	4
3.3.6 多图并列含间距 (带标题) .....	4
3.3.7 多图并列 (带子标题) .....	5
3.3.8 多图并列 (带子标题、子图无编号) .....	5
3.4 显示表格 .....	6
3.5 显示公式 .....	6
3.6 显示代码 .....	7
3.6.1 添加标题 .....	7
3.6.2 居左显示 .....	7
3.6.3 显示代码文件 .....	8

## 图形列表

图 3.1: 玫瑰 .....	3
图 3.2: 多图并列 (带标题) .....	4
图 3.3: 多图并列含间距 (带标题) .....	5
图 3.4: 多图并列 (带子标题) .....	5
图 3.5: 玫瑰 1 .....	5
图 3.6: 玫瑰 2 .....	5
图 3.7: 多图并列 (带子标题、子图无编号) .....	6

## 表格列表

表 3.1: 示例表格 .....	6
-------------------	---

## 代码列表

代码 3.1: 计算斐波纳契 .....	7
代码 3.2: 计算斐波纳契 .....	8
代码 3.3: 计算斐波纳契 .....	8

## 1 Typst 简介

Typst 是撰写任何长篇文本（如论文、文章、科学论文、书籍、报告和家庭作业）的优秀工具。此外，Typst 非常适合于编写任何包含数学符号的文档，例如在数学、物理和工程领域的论文。最后，由于其强大的风格化和自动化功能，它是任何一组具有共同风格的文件的绝佳选择，例如丛书。Typst 文档风格和 md 文档类似，所以很容易上手，同时内置了强大的脚本功能及较多的排版原语，因此，能比较轻松完成优质文档的撰写及排版工作。

## 2 Typst 安装

Typst 的本地安装非常简单，直接从 <https://github.com/typst/typst/releases> 下载适合自己操作系统的版本，解压到适当的地方即完成安装。另外，也可以在 <https://typst.app> 上注册账号，在线编辑 typst 文档，并下载生成的 PDF 文档。

编辑器建议采用 visual studio code，并安装 Typst LSP 和 Typst Preview 插件。

## 3 Typst 使用

### 3.1 创建文件

新建文本文档，以 .typ 为后缀。建议克隆 <https://github.com/soarowl/typst.git> 到本地，并将其中的 article.typ 复制到文档所在的目录，并适当进行修改。然后在文档头部添加如下内容：

```
1 #import "article.typ":*
2
3 #show: article.with(title: "Typst 简明使用教程", authors: ("卓能文",))
```

### 3.2 章节设置

格式有点类似 markdown，比较简单：

```
1 = 第一章
2 内容
3
4 == 第一节
5 内容
6
7 == 第二节
8 内容
9
10 == 第三节
11 内容
12
13 = 第二章
14 == 第一节
15
```

16  
17 == 第二节  
18 内容  
19  
20 = 第三章

### 3.3 显示图片

建议将图片保存在一个特定的目录，如 `images`、`img` 之类的地方。

```
1 #image("images/rose.jpg")
```

typst



#### 3.3.1 设置宽度:

```
1 #image("images/rose.jpg", width: 50%)
```

typst



#### 3.3.2 居中显示:

```
1 #align(center,image("images/rose.jpg", width: 50%))
```

typst



### 3.3.3 设置标题:

```
1 #figure(  
2   caption: [玫瑰],  
3   image("images/rose.jpg", width: 50%)  
4 )
```

typst



图 3.1: 玫瑰

注：放入 `#figure` 命令中的图片同时会在图形列表中出现。

### 3.3.4 多图并列

```
1 #grid(  
2   columns: (1fr, 1fr),  
3   image("images/rose.jpg"),  
4   image("images/rose.jpg"),  
5 )
```

typst



### 3.3.5 多图并列（带标题）

```
1 #figure(typst  
2   caption: [多图并列（带标题）],  
3   grid(  
4     columns: (1fr, 1fr),  
5     image("images/rose.jpg"),  
6     image("images/rose.jpg"),  
7   )  
8 )
```



图 3.2: 多图并列（带标题）

### 3.3.6 多图并列含间距（带标题）

```
1 #figure(typst  
2   caption: [多图并列含间距（带标题）],  
3   grid(  
4     columns: (1fr, 1fr),  
5     gutter: 10pt,  
6     image("images/rose.jpg"),  
7     image("images/rose.jpg"),  
8   )  
9 )
```





图 3.3: 多图并列含间距（带标题）

### 3.3.7 多图并列（带子标题）

```
1 #figure(
2   caption: [多图并列（带子标题）],
3   grid(
4     columns: (1fr, 1fr),
5     gutter: 10pt,
6     figure(
7       caption: [玫瑰 1],
8       image("images/rose.jpg")
9     ),
10    figure(
11      caption: [玫瑰 2],
12      image("images/rose.jpg")
13    ),
14  )
15 )
```



图 3.5: 玫瑰 1



图 3.6: 玫瑰 2

图 3.4: 多图并列（带子标题）

### 3.3.8 多图并列（带子标题、子图无编号）

```
1 #figure(
2   caption: [多图并列（带子标题、子图无编号）],
3   grid(
4     columns: (1fr, 1fr),
5     gutter: 10pt,
6     [#image("images/rose.jpg")玫瑰 1],
7     [#image("images/rose.jpg")玫瑰 2],
8   )
9 )
```

```
)
9 )
```



玫瑰 1



玫瑰 2

图 3.7: 多图并列 (带子标题、子图无编号)

### 3.4 显示表格

```
1 #figure(
2   caption: [示例表格],
3   kind: table,
4   supplement: "表",
5   ``tbl
6     Rx   Nx
7     Rx   Nx.
8   _
9   software|version
10  _
11     AFL|2.39b
12     Mutt|1.8.0
13     Ruby|1.8.7.374
14 TeX Live|2015
15 _
16 ``
17 )
```

表 3.1: 示例表格

software	version
AFL	2.39b
Mutt	1.8.0
Ruby	1.8.7.374
TeX Live	2015

注: 由于目前 Typst 中有 bug, 显示表格时, 必须加上 `kind` 和 `supplement` 字段。

更多用法请参考 <https://github.com/maxcrees/tbl.typ>

### 3.5 显示公式

```
1 勾股定理可用公式:  $a^2 + b^2 = c^2$  表示。
```

勾股定理可用公式:  $a^2 + b^2 = c^2$  表示。

```
1 $ \sum_{k=1}^n k = (n(n+1)) / 2 $
```



$$\sum_{k=1}^n k = \frac{n(n+1)}{2} \quad (1)$$

Typst 默认只能显示一级公式，不能按章节重新计数，可采用第三方包 `i-figured` 实现，本模板已经内置。格式请参考 `latex` 相关文档。

### 3.6 显示代码

代码可以很容易添加，格式和 `markdown` 一样。

```
1  ```py3typst
2  def fibonacci(n):
3      if n <= 1:
4          return n
5      else:
6          return fibonacci(n - 1) + fibonacci(n - 2)
7  ```
```

```
1  def fibonacci(n):python
2      if n <= 1:
3          return n
4      else:
5          return fibonacci(n - 1) + fibonacci(n - 2)
```

#### 3.6.1 添加标题

```
1  #figure(typst
2      caption: [计算斐波纳契],
3      ```py3
4      def fibonacci(n):
5          if n <= 1:
6              return n
7          else:
8              return fibonacci(n - 1) + fibonacci(n - 2)
9      ```
10 )
```

代码 3.1: 计算斐波纳契

```
1      def fibonacci(n):python
2          if n <= 1:
3              return n
4          else:
5              return fibonacci(n - 1) + fibonacci(n - 2)
```

#### 3.6.2 居左显示

因为 `figure` 命令会导致代码居中显示，添加 `align(start)` 命令让代码居左：

```
1  #figure(typst
2      caption: [计算斐波纳契],
3      align(start)[
4          ```py3
5          def fibonacci(n):
6              if n <= 1:
7                  return n
8          ```
```

```
9         return fibonacci(n - 1) + fibonacci(n - 2)
10     """
11 ]
12 )
```

代码 3.2: 计算斐波纳契

```
1 def fibonacci(n):
2     if n <= 1:
3         return n
4     else:
5         return fibonacci(n - 1) + fibonacci(n - 2)
```

python

### 3.6.3 显示代码文件

在 Typst 文档中添加太多代码，导致可读性降低，也不便于后续采用相应的工具进行编辑、更新、管理与维护，建议将代码组织在一个文件夹中。

```
1 #figure(
2     caption: [计算斐波纳契],
3     align(start, raw(read("src/fibonacci.py"), lang: "py3", block: true))
4 )
```

typst

代码 3.3: 计算斐波纳契

```
1 def fibonacci(n):
2     if n <= 1:
3         return n
4     else:
5         return fibonacci(n - 1) + fibonacci(n - 2)
6
```

python