

Manual of Typst

Duolei WANG

wangdl2020@mail.sustech.edu.cn

为 Typst 语言写一个极其简要的文档，能够帮助快速入门这一门语言。

Contents

| | |
|---------------------|----------|
| 1 快速入门 | 3 |
| 1.1 编程语言内核 | 3 |

1 快速入门

1.1 编程语言内核

首先需要声明的是，Typst 更像是一门简单的编程语言，据编者了解，大概是 Rust 语言。要完全理解 Typst 更像是初窥一个解释器的门径。下面进行一些重点介绍。

1. 如何理解文件排版：

它通过特殊的语句来控制我们在 .pdf 文件里的文本的输出格式。总的来说，Typst 编译器实现了 .typ \rightarrow .pdf 的映射。

Typst 编译器对 .typ 格式的文本进行统一的处理，使用 # 区分文本内容，和编程语言内容。所有未经修饰的直接书写的部分，在 .typ 文件里的语句都会被统一的处理为文本。如果使用 # 标记，将会调用编译器内部的编程语言相关部分，如果不使用，所有的输入都会默认是文本内容。

Typst 支持 $\$$ 、 $=$ 等类似 markdown 的标记，其实他们也只是一些函数的语法糖，指定了这个输入内部的文本会以数学公式、标题等格式呈现。

于是，使用 typst 书写格式化的文本的思路就很简单，核心的使用思路是对某个文本内容设置某个格式，可以分为以下步骤：1. 设置文本格式 \rightarrow 2. 输入文本内容。

比如：

- 1. 书写标题：设置为标题格式 \rightarrow 标题内容
- 2. 书写文本：设置为文本格式 \rightarrow 文本内容

局部设置文本的格式，可以使用以下语法进行书写

`#text(格式参数)[文本内容]`

可以注意到，这种方式就像是编程语言中的函数传参。后面方括号内的内容是 Typst 设定的 `content` 类型的参数，它的内部可以是任何格式，以作为一个整体的参数。

或者，可以使用 `#set` 参数去更改全局变量设置，这种方式会进行覆盖式的设置，对以后书写的文本都生效。

`#set 参数名(参数内容)`

细心的读者可以感觉到，即便提供了两种不同范围的设置方式，仍然具有使用上的不方便：比如我想让一级标题统一采用某某格式的字体，但是按照上述 `#set` 方式进行的时候，就只能反复设置同一段内容。

于是，typst 里的 `#show` 参数就起到了用处。

`#show` 参数使用：`#show \rightarrow selector` 的格式进行统一处理，右侧 `selector` 意为某个筛选器（以一定格式筛选某个文本块）：

```
#show heading(level: 2): it => block[
```

参数设置

```
]
```

此处，指定了被筛选的内容 `heading`，筛选条件是 `level = 2`，后面 `it => block` 的语法是代之这个被筛选到的内容将会被统一执行 `block` 中的某些参数设置。如果只有单条语句，直接使用

```
#show selector: 参数设置
```

即可。

值得注意的是，`typst` 的 `selector` 不仅仅可以筛选 `heading` 等函数关键字，还可以使用正则表达式、直接的文本搜索、标签、`Everything` 等：

- `#show regex(正则表达式内容)`
- `#show "某个单词"`，注意这种搜索甚至可以完成文本替换，只需要在 `:` 后面直接写入单词即可。
- `#show <Label>: set...` 来筛选特定标签
- `#show rest` 来表示全部内容 `s`