



MUSE-BOOK BASED ON RPG TEMPLATE

Created by Camuse 2019 年 6 月 19 日

目 录

第一卷 唯一一卷

第一章 基本功能介绍	2
1.1 基本构成	2
1.2 无序列表的使用	2
1.3 彩色盒子的使用	2
1.4 在顶部和底部添加优美的图片	4
1.5 透明的图片	5
1.6 为图片选择你想要的透明度	6
1.7 定理等环境	6

第一卷

唯一一卷



第一章

基本功能介绍

1.1 基本构成

本模板有卷、章、节、小节、次小节构成。目录只显示到小节。

1.2 无序列表的使用

✂ 原模板的配色很好看

✂ 原模板的图片很好看

✂ 原模板的盒子也不错

1.3 彩色盒子的使用

简单的彩色盒子

这是一个简单的盒子，中括号里面放上盒子的名称。

颜色可以改变的

盒子虽然很简答，但是可以改变颜色，花括号里面是典型的 `rgb` 确定颜色的方法。不写的话就和上一个盒子一样用默认的颜色，填的话就变成你要的颜色。

凸显标题的盒子

这个盒子是另一个样式，能够凸显标题。

个盒子是设计来引用的，引用的放在盒子里面就更加突出了。

有背景图片的盒子

这是一个带背景图片的盒子。

这是盒子的设计就是从 `tcolotbox` 的说明文档上直接抄来的，挺好看的。



1.4 在顶部和底部添加优美的图片

使用了之后会把页眉页脚这些的位置改变哦，方括号里面只能填 `t` 或者 `b`。`t` 自然是 `top`，`b` 就是 `bottom`。实际上加的图片是浮动体，可能出现的位置不是那么理想。你喜欢的话就用吧。





1.5 透明的图片

如果觉得不想改变整个版面的边距，实际上可以加上 * 号，这样加入的图片位置就在当前页面了，不会乱跑了。这样做的话，图片是带透明度的。担心透明度不满意？自己的图片透明度不合适怎么办？

1.6 为图片选择你想要的透明度

实际上这个指令可以加第三个参数，如同示例的，0.7 是图片的不透明度。你可以选择 0-1 之间的任意值。1 就是显示原图的样子，0 就是完全透明了。

1.7 定理等环境

定理、定义、命题、例子这四个环境已经定义好了。可以直接使用。

关于引用方面。我使用 `clevercref` 宏包来完成，因为它比较智能，很好用。所以一下的交叉引用都是用它。如果你还是喜欢普通的引用的话也还是可以用。

例子 1.7.1: 例子盒子的简单使用

这是一个例子。在盒子内部自己引用的话可以用 例子 1.7.1 里面具体的代码方法。但是这个引用相当鸡肋。出了盒子就不能引用了。当然要有 label 才好了。如果你没写 label 的话，就像我这个例子里面一样，那么我自动给它绑定了一个，可以用例子 1.7.1。但是如果你在前面添加章节的话，自动编的 label 就全部改变了，所以还是老老实实的自己加 label 吧。不要不添加，除非你认为不需要引用。

例子 1.7.2: 记得被它们加上 label 哦

加上 label 之后就可以放心使用了。建议加上前缀 `exa:`，不加也可以，但是这样容易区分，免得自己后面都把 label 混了。引用一下自己的 例子 1.7.2。这样子就不怕 label 乱了。

定理 1.7.1: 一个定理

这里面根本不是一个真正的定理，这只是一个例子啊。建议 label 加上前缀 `thm:`。

定理 1.7.1 是一个很有用的定理啊，你看它告诉你怎么使用定理环境了。

定义 1.7.1: 喜欢耍流氓的人就是流氓

我们定义喜欢耍流氓的人为流氓。

我认为定义 1.7.1 的定义不妥当，耍流氓也可以是把流氓耍了，这个把流氓耍了的人不应该叫做流氓。

命题 1.7.1: 该命题是一个假命题

我是一个假的命题。

命题 1.7.1 是一个假命题。我该信吗？

Shell 1.7.1: 一个命令行

```
Camuse $> git clone https://github.com/CamuseCao/RPG-LaTeX-Template
Camuse $> cd RPG-Latex-Template
Camuse $> latexmk -xelatex
```

Shell 1.7.1向我们展示了这个项目的基本使用。

Listing 1.7.1: 结果与代码

$$\alpha + \beta = \omega$$

(1.1)

```
1 \begin{equation}
2   \alpha + \beta = \omega
3 \end{equation}
```

Listing 1.7.2: 代码与结果

```
1 \begin{equation}
2   \alpha + \beta = \omega
3 \end{equation}
```

$$\alpha + \beta = \omega$$

(1.2)

Listing 1.7.3: 上下版本的

```
1 \begin{equation}
2   \alpha + \beta = \omega
3 \end{equation}
```

$$\alpha + \beta = \omega$$

(1.3)

Listing 1.7.4: 只有代码

```
1 \begin{equation}
2   \alpha + \beta = \omega
3 \end{equation}
```