

# codebox:代码盒子排版宏包

## codebox:programming source code box

耿楠

nangeng@nwafu.edu.cn

2022/01/04 v1.0.3<sup>‡</sup>

### 简介

codebox 是一个基于 tcolorbox 用 L<sup>A</sup>T<sub>E</sub>X3 开发的 L<sup>A</sup>T<sub>E</sub>X 宏包,它提供用于排版源代码盒子的环境和命令。其中,codebox 和 codeview 环境用于根据环境内容排版代码,\codefile 和 \cvfile 命令用于根据文件内容排版代码。同时, codebox 还提供了各环境和命令的星号版本,如 codebox\* 和 codeview\*,\codefile\* 和 \cvfile\*,用于为代码盒子添加底线注释。

## 目录

第 1 节 简介	1	3.8 代码行距 . . . . .	5
第 2 节 用户接口	2	3.9 行号间距 . . . . .	5
2.1 codebox 和 codebox* 环境 . .	2	3.10 引用标签 . . . . .	5
2.2 \codefile 和 \codefile* 命令	2	第 4 节 计数器	6
2.3 codeview 和 codeview* 环境	3	第 5 节 排版样例	7
2.4 \cvfile 和 \cvfile* 命令 . .	4	5.1 Java 代码 . . . . .	7
第 3 节 选项说明	4	5.2 Python 代码 . . . . .	7
3.1 代码引擎 . . . . .	4	5.3 listings 引擎 . . . . .	8
3.2 代码语言 . . . . .	4	第 6 节 代码实现	8
3.3 代码名称 . . . . .	4	版本历史	16
3.4 代码样式 . . . . .	5	代码索引	16
3.5 代码字号 . . . . .	5		
3.6 底线注释内容 . . . . .	5		
3.7 底线注释格式 . . . . .	5		

## 第 1 节 简介

codebox 是一个基于 tcolorbox 用 L<sup>A</sup>T<sub>E</sub>X3 开发的 L<sup>A</sup>T<sub>E</sub>X 宏包,它可以根据用户设置使用 minted 或 listings 宏包实现代码盒子排版。

该宏包提供了 codebox 和 codeview 两个环境和 \codefile 和 \cvfile 两个命令以排版代码盒子。

同时,codebox 宏包还提供环境和命令的星号版本,用于为代码盒子添加底线注释。

<sup>\*</sup><https://github.com/registor/codebox>

<sup>‡</sup>[https://gitee.com/nwafu\\_nan/codebox](https://gitee.com/nwafu_nan/codebox)

## 第2节 用户接口

### 2.1 codebox 和 codebox\* 环境

---

```
codebox      \begin{codebox}[(外观选项)]{盒子标题}
codebox*     .....
             \end{codebox}
             \begin{codebox*}[(外观选项)]{盒子标题}
             .....
             \end{codebox*}
```

---

New: 2021-12-25  
Updated: 2021-12-25

---

使用环境内容排版代码盒子。可以通过 `{盒子标题}` 为代码盒子设置标题。

`[(外观选项)]` 中通过 `key-value` 方式设置代码盒子外观。

星号环境 `codebox*` 用于为代码盒子添加底线注释, 此时, 需要在 `[(外观选项)]` 中用 `<comments>= <底线注释>` 选项提供**底线注释**内容。

代码盒子的外观也可以通过 `\codeset` 命令的逗号分隔 `key-value` 列表进行设置。

```
1 \centering
2 \begin{codebox}{代码标题}
3   \include <stdio.h>
4   \include <stdlib.h>
5
6   int main(void)
7   {
8       printf("Hello World!\n");
9
10      return 0;
11  }
12 \end{codebox}
```

#### 代码盒子标题

```
1 \include <stdio.h>
2 \include <stdlib.h>
3
4 int main(void)
5 {
6     printf("Hello World!\n");
7
8     return 0;
9 }
```

### 2.2 \codefile 和 \codefile\* 命令

---

```
\codefile    \codefile [(外观选项)] {盒子标题} {代码文件}
\codefile*   \codefile* [(外观选项)] {盒子标题} {代码文件}
```

---

New: 2021-12-25  
Updated: 2021-12-25

---

用于排版代码盒子, 源代码来自于 `{代码文件}`, 可以通过 `{盒子标题}` 为代码盒子设置标题。

`[(外观选项)]` 中通过 `key-value` 方式设置代码盒子外观。

星号命令 `codefile*` 用于为代码盒子添加底线注释, 此时, 需要在 `[(外观选项)]` 中用 `<comments>= <底线注释>` 选项提供**底线注释**内容。

代码盒子的外观也可以通过 `\codeset` 命令的逗号分隔 `key-value` 列表进行设置。

```

1 \centering
2 \codefile{代码盒子标题}{test.c}

```

### 代码盒子标题

```

1 #include <stdio.h>
2 #include <stdlib.h>
3
4 int main(void)
5 {
6     printf("Hello World!\n");
7
8     return 0;
9 }

```

## 2.3 codeview 和 codeview\* 环境

```

codeview \begin{codeview}[<外观选项>]{<盒子标题>}
codeview* .....
\end{codeview}
\begin{codeview*}[<外观选项>]{<盒子标题>}
.....
\end{codeview*}

```

New: 2021-12-26  
Updated: 2021-12-26

使用环境内容排版带编号代码盒子。可以通过 {<盒子标题>} 为代码盒子设置标题。

[<外观选项>] 中通过 key-value 方式设置代码盒子外观。

星号环境 codeview\* 用于为代码盒子添加底线注释, 此时, 需要在 [<外观选项>] 中用 <comments>=<底线注释>选项提供**底线注释**内容。

代码盒子的外观也可以通过\codeset命令的逗号分隔 key-value 列表进行设置。

```

1 \centering
2 \begin{codeview}{代码标题}
3     #include <stdio.h>
4     #include <stdlib.h>
5
6     int main(void)
7     {
8         printf("Hello World!\n");
9
10        return 0;
11    }
12 \end{codeview}

```

### 代码 1 代码标题

```

1 #include <stdio.h>
2 #include <stdlib.h>
3
4 int main(void)
5 {
6     printf("Hello World!\n");
7
8     return 0;
9 }

```

## 2.4 \cvfile和\cvfile\*命令

\cvfile  
\cvfile\*

New: 2021-12-26  
Updated: 2021-12-26

\cvfile [⟨外观选项⟩] {⟨盒子标题⟩} {⟨代码文件⟩}  
\cvfile\* [⟨外观选项⟩] {⟨盒子标题⟩} {⟨代码文件⟩}

用于排版带编号代码盒子, 源代码来自于 {⟨代码文件⟩}, 可以通过 {⟨盒子标题⟩} 为代码盒子设置标题。

[⟨外观选项⟩] 中通过 key-value 方式设置代码盒子外观。

星号命令 `cvfile*` 用于为代码盒子添加底线注释, 此时, 需要在 [⟨外观选项⟩] 中用 `<comments>=⟨底线注释>` 选项提供底线注释内容。

代码盒子的外观也可以通过 `codeset` 命令的逗号分隔 key-value 列表进行设置。

```
1 \centering
2 \cvfile*[comments=这是一个简单的C语言代码]{代码标题}{test.c}
```



## 第3节 选项说明

`codebox` 宏包提供了一系列选项, 以设置代码盒子的外观样式。载入 `codebox` 宏包后, 以下选项均可通过用户接口命令 `\codeset` 进行设置。当然, 这些选项也可以通过各个环境或命令的 [⟨外观选项⟩] 进行设置。

### 3.1 代码引擎

minted

New: 2021-12-26  
Updated: 2021-12-26

minted = (true|false)

初始值 = true

minted 用于设置代码排版引擎, 如为 **true** 则使用 `minted` 宏包排版代码盒子中的内容, 如为 **false** 则使用 `listings` 宏包排版代码盒子中的内容。默认值为 **true**。

### 3.2 代码语言

lang

New: 2021-12-26  
Updated: 2021-12-26

lang = {⟨代码语言⟩}

初始值 = c

lang 用于设置代码语言, 默认值为 **c** 语言。

### 3.3 代码名称

pretitle

New: 2021-12-26  
Updated: 2021-12-26

pretitle = {⟨代码名称⟩}

初始值 = Code

pretitle 用于代码编号前的代码名称前缀, 默认值为 **Code**。

### 3.4 代码样式

<b>codestyle</b>	<code>codestyle = {&lt;代码样式&gt;}</code>	初始值 = <code>codeblocks</code>
New: 2021-12-26 Updated: 2021-12-26	<code>codestyle</code> 用于设置代码盒子中代码的样式, 仅对 <b>minted</b> 引擎有效, 默认值为 <code>codeblocks</code> 。	

### 3.5 代码字号

<b>codesize</b>	<code>codesize = {&lt;字号命令&gt;}</code>	初始值 = <code>\small</code>
New: 2021-12-26 Updated: 2021-12-26	<code>codesize</code> 用于设置代码盒子中代码的字号, 仅对 <b>minted</b> 引擎有效, 默认值为 <code>\small</code> 。	

### 3.6 底线注释内容

<b>comments</b>	<code>comments = {&lt;注释内容&gt;}</code>	初始值 = 无
New: 2021-12-26 Updated: 2021-12-26	<code>commentf</code> 用于设置代码盒子底线注释的内容, 默认值为无。	

### 3.7 底线注释格式

<b>commentf</b>	<code>commentf = {&lt;格式命令组&gt;}</code>	初始值 = <code>\small\sffamily</code>
New: 2021-12-26 Updated: 2021-12-26	<code>commentf</code> 用于设置代码盒子底线注释的格式, 默认值为 <code>\small\sffamily</code> 。	

### 3.8 代码行距

<b>codestretch</b>	<code>codestretch = {&lt;浮点数&gt;}</code>	初始值 = <code>1.0</code>
New: 2021-12-26 Updated: 2021-12-26	<code>codestretch</code> 用于设置代码盒子内容 (代码) 的行距, 仅对 <b>minted</b> 引擎有效, 取浮点数。默认值为 <code>1.0</code> 。	

### 3.9 行号间距

<b>linenumsep</b>	<code>linenumsep = {&lt;浮点数&gt;}</code>	初始值 = <code>3.0</code>
New: 2021-12-26 Updated: 2021-12-26	<code>linenumsep</code> 用于设置代码盒子行号与代码的间距, 仅对 <b>minted</b> 引擎有效, 取浮点数, 单位是 mm。默认值为 <code>3.0</code> 。	

### 3.10 引用标签

<b>label</b>	<code>label = {&lt;引用标签名称&gt;}</code>	初始值 = 空
New: 2022-1-4 Updated: 2022-1-4	<code>label</code> 用于设置带编号代码盒子的引用标签, 该选项仅对 <code>codeview/codeview*</code> 环境或 <code>\cvfile/\cvfile*</code> 命令有效。默认值为空。	

## 第 4 节 计数器

---

**cvcounter**

---

New: 2021-12-28

Updated: 2021-12-28

---

`codebox` 宏包提供了一个计数器 `cvcounter`, 当使用带编号环境 `codeview/codeview*` 和命令 `\cvfile/\cvfile*` 排版代码时, 可以用于对代码盒子进行计数。默认情况下, 如果存在 `thechapter` 计数器, 则其父计数器设置为 `chapter` 否则将按全文统一计数。

可以使用类似 `\renewcommand{\thecvcounter}{\thechapter.\arabic{cvcounter}}` 的形式更改编号输出内容。

## 第5节 排版样例

`codebox` 宏包可用于需要语法高亮代码排版场合,以避免使用代码截图。代码盒子可以带/不带底线注释。

### 5.1 Java 代码

可以使用`\codeset`命令设置代码语言。

```
1 \centering
2 \codeset{lang=java}
3 \codefile{Java代码盒子}{hellojava.java}
```

#### Java 代码盒子

```
1 public class HelloWorld {
2     public static void main(String[] args){
3         System.out.println("Hello World!");
4     }
5 }
```

### 5.2 Python 代码

还可以使用环境或命令的[外观选项]设置代码语言。当然,也可以通过使用 `label` 选项为编号代码添加引用标签,并通过`\ref`命令引用该标签,如代码3所示。

```
1 \centering
2 \cvfile[lang=python,label=code-test]{Python代码示例}{hellopy.py}
```

#### 代码 3 Python 代码示例

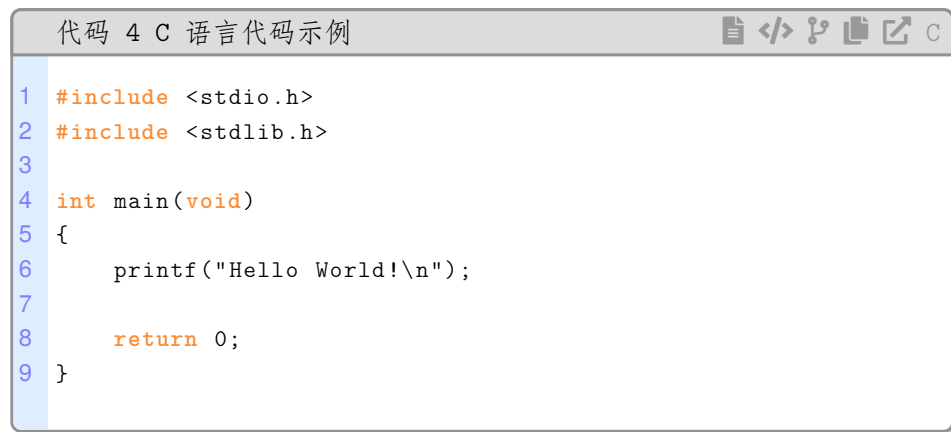


```
1 import tensorflow as tf
2 import numpy as np
3 import os
4 os.environ['TF_CPP_MIN_LOG_LEVEL'] = '2'
5
6 # Create 100 phony x, y data points in Numpy, y = x * 0.1 + 0.3
7 x_data = np.random.random(100).astype("float32")
8 y_data = x_data * 0.1 + 0.3
9
10 # Try to find values for W and b that compute y_data = W * x_data + b
11 W = tf.Variable(tf.random_uniform([1], -1.0, 1.0))
12 b = tf.Variable(tf.zeros([1]))
13 y = W * x_data + b
14
15 # Minimize the mean squared errors.
16 loss = tf.reduce_mean(tf.square(y - y_data))
17 optimizer = tf.train.GradientDescentOptimizer(0.5)
18 train = optimizer.minimize(loss)
19
20 # Before starting, initialize the variables. We will 'run' this first
21 init = tf.global_variables_initializer()
22
23 # Launch the graph.
24 sess = tf.Session()
25 sess.run(init)
26
27 # Fit the line.
28 for step in range(201):
29     sess.run(train)
30     if step % 20 == 0:
31         print(step, sess.run(W), sess.run(b))
32
```

### 5.3 listings 引擎

在选项中使用  $\langle minted \rangle = \langle false \rangle$  以使用 listings 宏包排版代码。

```
1 \centering
2 \cvfile[minted=false,lang=c]{C语言代码示例}{test.c}
```



## 第 6 节 代码实现

该宏包使用 L<sup>A</sup>T<sub>E</sub>X3 语法编写, 依赖 expl3 环境, 并需调用 l3packages、tcolorbox、minted listings、fontawesome5 等宏包。

按照 L<sup>A</sup>T<sub>E</sub>X3 语法, 代码中的空格、换行、回车与制表符会完全被忽略, 而下划线“\_”和冒号“:”则可作为一般字母使用。正常的空格可以使用 “~” 代替; 至于 ~ 原来所表示的“带子”, 则要用 L<sup>A</sup>T<sub>E</sub>X2<sub>ε</sub> 的原始命令 \nobreakspace 代替。

以下代码中有一些形如  $\langle *package \rangle$  的标记, 这是 DocStrip 中的 “guard”, 用来选择性地提取文件。“\*”和“/”分别表示该部分的开始和结束。不含“\*”和“/”的 guard 出现在行号右侧, 它们用来确定单独一行代码的归属。这些 guard 的颜色深浅不一, 用以明确嵌套关系。

另有若干形如  $\langle @@=codebox \rangle$  的 guard, 它们由 l3docstrip 定义, 用来标识名字空间 (模块)。

### 6.1 环境检测与准备

```
1 <*package>
2 <@@=codebox>
```

载入必要的 L<sup>A</sup>T<sub>E</sub>X3 宏包

```
3 \RequirePackage { xtemplate, l3keys2e, xparse }
```

### 6.2 载入其它宏包

载入 fontawesome5、tcolorbox、amssymb、xcolor 宏包

```
4 \RequirePackage {fontawesome5, tcolorbox, varwidth, xcolor, etoolbox}
```

解决 Expl3 中 tcolorbox 和 tikz 附加库无法加载的的补丁 (需要 etoolbox 宏包支持)

<https://tex.stackexchange.com/questions/57424/using-of-usetikzlibrary-in-an-exp>

```
5 \ExplSyntaxOff
6 \patchcmd
7 {\tcb@input@library@in}
8 {%
9   \input\tcbpkgprefix#1\relax%
10 }
11 {%
12   \@pushfilename
13   \input\tcbpkgprefix#1\relax%
14   \@popfilename
```



```

15 }
16 {}{}
17
18 \patchcmd
19 {\pgfutil@InputIfFileExists}
20 {\input #1}
21 {%
22   \@pushfilename
23   \xdef\@currname{#1}%
24   \input #1 %
25   \@popfilename
26 }
27 {}{}
28 \ExplSyntaxOn

```

载入 tcolorbox 和 tikz 附加库

```

29 \tcbuselibrary{skins, xparse, breakable}
30 \tcbuselibrary{minted,listings}
31 \usetikzlibrary{shapes.geometric}

```

### 6.3 定义颜色

```

32 \definecolor{cvgrayc}{RGB}{247,247,247}
33 \definecolor{cvgray}{RGB}{220,220,220}
34 \definecolor{cvgrayb}{RGB}{153,153,153}
35 \definecolor{cvblue}{RGB}{223,238,255}
36 \definecolor{chengse}{RGB}{250,140,53}

```

### 6.4 代码盒子样式设计

代码盒子 lang 样式

```

37 \tcbset{%
38   skin=enhanced,
39   lang/.style={%
40     breakable,%
41     drop~shadow,%
42     colframe=gray!75!black,%
43     left=4.5mm,%
44     enhanced,%
45     colframe=tcbcolback!60!black,%
46     colback=white,%
47     colbacktitle=tcbcolback!5!gray!10!white,%
48     fonttitle=\bfseries,%
49     coltitle=black,%
50     attach~boxed~title~to~top~center={%
51       yshift=-0.25mm-\tcbboxeditheight/2,%
52       yshifttext=2mm-\tcbboxeditheight/2%
53     },%
54     attach~boxed~title~to~top~left={%
55       xshift=1cm,%
56       yshift*=1mm-\tcbboxeditheight%
57     },%
58     varwidth~boxed~title*=-3cm,%
59     boxed~title~style={%
60       frame~code={%
61         \path[fill=tcbcolback!30!black] ([yshift=-1mm,xshift=-1mm]frame.north~west)%
62         arc[start~angle=0,end~angle=180,radius=1mm] ([yshift=-1mm,xshift=1mm]frame.north~east)%
63         arc[start~angle=180,end~angle=0,radius=1mm];%
64         \path[left~color=tcbcolback!60!black,right~color=tcbcolback!60!black,
65           middle~color=tcbcolback!80!black] ([xshift=-2mm]frame.north~west)%
66         --([xshift=2mm]frame.north~east)[rounded~corners=1mm]%
67         --([xshift=1mm,yshift=-1mm]frame.north~east)%
68         --(frame.south~east)%
69         --(frame.south~west)%
70         --([xshift=-1mm,yshift=-1mm]frame.north~west)[sharp~corners]%
71         --cycle;%

```

```

72     },%
73     interior~engine=empty%
74 },%
75 overlay={%
76     \begin{tcbclipinterior}
77         \fill[tcbcolback!80!black] (frame.south~west) rectangle
78             ([xshift=5mm]frame.north~west);
79     \end{tcbclipinterior}%
80 },
81 }%
82 }%

```

### 代码盒子 cv 样式

```

83 \tcbset{%
84     skin=enhanced,
85     cv/.style={
86         boxrule=0.4mm,
87         breakable,
88         top=0mm,
89         boxsep=1mm,
90         drop~shadow,
91         attach~boxed~title~to~top,
92         colframe=blue!75!black,
93         left=4.5mm,
94         right=0mm,
95         enhanced,
96         colframe=tcbcolback!60!black,
97         colback=white,
98         colbacktitle=cvgray,
99         fonttitle=\ttfamily,
100        coltitle=black,
101        underlay~boxed~title = {
102            \begin{tcbclipinterior}
103                \fill[cvblue] (frame.south~west) rectangle ([xshift=5.0mm,yshift=0mm]frame.north~west);
104            \end{tcbclipinterior}
105        }
106    }
107 }%

```

## 6.5 内部变量声明

定义变量。

```

\l__codebox_minted_bool
\l__codebox_comment_bool
\l__codebox_language_tl
\l__codebox_code_style_tl
\l__codebox_code_fontsize_tl
\l__codebox_code_name_tl
\l__codebox_comment_contents_tl
\l__codebox_comment_format_tl
\l__codebox_label_name_tl
\l__codebox_baseline_stretch_fp
\l__codebox_linenumber_sep_fp
108 \bool_new:N \l__codebox_minted_bool
109 \bool_new:N \l__codebox_comment_bool
110
111 \tl_new:N \l__codebox_language_tl
112 \tl_new:N \l__codebox_code_style_tl
113 \tl_new:N \l__codebox_code_fontsize_tl
114 \tl_new:N \l__codebox_code_name_tl
115 \tl_new:N \l__codebox_comment_contents_tl
116 \tl_new:N \l__codebox_comment_format_tl
117 \tl_new:N \l__codebox_label_name_tl
118 \fp_new:N \l__codebox_baseline_stretch_fp
119 \fp_new:N \l__codebox_linenumber_sep_fp

```

## 6.6 选项处理

定义 codebox 键值类。

```

120 \keys_define:nn { codebox }
121 {

```

**minted** 是否使用 minted 宏包排版代码,默认为 true。

```

122     minted .bool_set:N = \l__codebox_minted_bool,
123     minted .default:n = true,

```

```

124     minted .initial:n = true,

lang 设置排版代码语言,默认为c。
125     lang .tl_set:N = \l__codebox_language_tl,
126     lang .initial:n = c,

pretitle 代码名称前缀
127     pretitle .tl_set:N = \l__codebox_code_name_tl,
128     pretitle .initial:n = Code,

codestyle 代码样式
129     codestyle .tl_set:N = \l__codebox_code_style_tl,
130     codestyle .initial:n = default,

codesize 代码字号
131     codesize .tl_set:N = \l__codebox_code_fontsize_tl,
132     codesize .initial:n = \small,

comments 注释格式
133     comments .tl_set:N = \l__codebox_comment_contents_tl,
134     comments .initial:n = {},

commentf 注释格式
135     commentf .tl_set:N = \l__codebox_comment_format_tl,
136     commentf .initial:n = \small\sffamily,

codestretch 基线伸展系数
137     codestretch .code:n = { \fp_set:Nn \l__codebox_baseline_stretch_fp { #1 }
138                           },
139     codestretch .initial:n = 1.0,

linenumsep 行号与文本之间的距离(单位:mm)
140     linenumsep .fp_set:N = \l__codebox_linenumber_sep_fp,
141     linenumsep .initial:n = 3.00,

label 注释格式
142     label .tl_set:N = \l__codebox_label_name_tl,
143     label .initial:n = {},

处理未知选项。
144     unknown .code:n = { \__codebox_error:n { unknown-option } }
145 }
146 \msg_new:nnn { codebox } { unknown-option }
147 { package~ option~ "\l_keys_key_tl"~ is~ unknown. }

```

## 6.7 选项设置用户接口

`\codeset` 选项设置用户接口。

```

148 \NewDocumentCommand \codeset { m }
149 { \keys_set:nn { codebox } {#1} }

```

## 6.8 内部函数

`\_codebox_set_counter_parent:` 设置代码编号计数器的父计数器。

```

150 \cs_new:Npn \_codebox_set_counter_parent:
151 {
152   \ifdef{\thechapter}
153   {
154     \newcounter{cvcounter}[chapter]%
155     \renewcommand{\thecvcounter}{\thechapter.\arabic{cvcounter}}
156   }{
157     \newcounter{cvcounter}%[section]%
158   }
159 }
```

`\_codebox_minted_engine_tcbset:n #1` 设置代码盒子 `minted` 宏包参数。

```

160 \cs_new:Npn \_codebox_minted_engine_tcbset:n #1
161 {
162   \tcbset{
163     listing~engine=minted,%
164     minted~style=#1,
165     minted~options={%
166       autogobble,
167       breaklines,%
168       fontsize=\tl_use:N \l__codebox_code_fontsize_tl,%
169       baselinestretch=\fp_eval:n { \l__codebox_baseline_stretch_fp },
170       breaksymbolleft={},%
171       linenos,%
172       numbersep=\fp_eval:n { \l__codebox_linenumbersep_fp }~mm,
173     },%
174     minted~language=\tl_use:N \l__codebox_language_tl%
175   }
176 }
177 \cs_generate_variant:Nn \_codebox_minted_engine_tcbset:n {V}
```

`\_codebox_listings_engine_tcbset:` 设置代码盒子 `listings` 宏包参数。

```

178 \cs_new:Npn \_codebox_listings_engine_tcbset:
179 {
180   \lstset{% general command to set parameter(s)
181     basicstyle=\ttfamily\tl_use:N \l__codebox_code_fontsize_tl, % print whole listing small
182     keywordstyle=\color{chengse}\bfseries,%
183     identifierstyle=, % nothing happens
184     breaklines=true,
185     commentstyle=\itshape\color{gray},
186     stringstyle=\ttfamily, % typewriter type for strings
187     numbers=left,%
188     numberstyle=\tl_use:N \l__codebox_code_fontsize_tl\sffamily\color[rgb]{0.5,0.5,1.0},%
189   }%
190   \tcbset{
191     overlay={%
192       \begin{tcbclipinterior}
193         \fill[tcbcolback!80!black] (frame.south~west) rectangle
194                                   ([xshift=5mm]frame.north~west);
195       \end{tcbclipinterior}%
196     },%
197     listing~options={
198       language=\tl_use:N \l__codebox_language_tl,
199       numbersep=\fp_eval:n { \l__codebox_linenumbersep_fp }~mm,
200       %linebackgroundcolor={\ifodd\value{lstnumber}\color{cvgray}\fi}
201     }
202   }%
203 }
```

`\_codebox_code_engine_tcbset:` 设置代码盒子引擎宏包参数。

```

204 \cs_new:Npn \_codebox_code_engine_tcbset:
205 {
```

```

206 \bool_if:NTF \l__codebox_minted_bool
207 {
208   \__codebox_minted_engine_tcbset:V \l__codebox_code_style_tl
209 }{
210   \__codebox_listings_engine_tcbset:
211 }
212 }

```

设置代码盒子底线注释。

```

\__codebox_comment_tcbset:
213 \cs_new:Npn \__codebox_comment_tcbset:
214 {
215   \bool_if:NTF { \l__codebox_comment_bool }
216   {
217     \tcbset{%
218       listing~and~comment,%
219       colbacklower=tcbcolback!5!yellow!10!white,%
220       collower=tcbcolback!60!black,%
221       comment={\tl_use:N \l__codebox_comment_format_tl~
222                 \tl_use:N \l__codebox_comment_contents_tl},%
223     }
224   }{
225     \tcbset{ listing~only,}
226   }
227 }

```

## 6.9 不计数代码盒子用户接口

定义不带底线注释的代码盒子环境 `codebox`

```

228 \NewDocumentEnvironment{ codebox }{0}{ m }
229 {
230   \bool_set_false:N \l__codebox_comment_bool
231   \group_begin:
232   \IfNoValueF{#1}{ \keys_set:nn { codebox } { #1 } }
233   \__codebox_code_engine_tcbset:
234   \__codebox_comment_tcbset:
235   \tcbset{lang, title={#2} }%
236   \tcblisting{}
237 }{
238   \endtcblisting
239   \group_end:
240 }

```

定义带底线注释的代码盒子环境 `codebox*`

```

241 \NewDocumentEnvironment{ codebox* }{0}{ m }
242 {
243   \bool_set_false:N \l__codebox_comment_bool
244   \group_begin:
245   \IfNoValueF{#1}{ \keys_set:nn { codebox } { #1 } }
246   \__codebox_code_engine_tcbset:
247   \__codebox_comment_tcbset:
248   \tcbset{lang, title={#2} }%
249   \tcblisting{}
250 }{
251   \endtcblisting
252   \group_end:
253 }

```

定义带/不带底线注释的从文件读取内容的命令 `\codefile/\codefile*`

```

254 \NewDocumentCommand \codefile { s O{} m m }
255 {
256   \IfBooleanTF{#1}
257   {
258     \bool_set_true:N \l__codebox_comment_bool
259   }{
260     \bool_set_false:N \l__codebox_comment_bool

```

```

261     }
262
263     \group_begin:
264     \IfNoValueF{#2}{ \keys_set:nn { codebox } { #2 } }
265     \__codebox_code_engine_tcbset:
266     \__codebox_comment_tcbset:
267     \tcbset{
268         lang,
269         title={#3},
270         listing~file={#4},
271     }%
272     \tcbinputlisting{ }
273     \group_end:
274 }

```

## 6.10 计数代码盒子用户接口

设置代码盒子计算器

```
275 \__codebox_set_counter_parent:
```

定义不带底线注释的代码盒子环境 codeview

```

276 \DeclareTCBListing[use~counter=cvcounter]{codeviewaux}{m}
277 {
278     title={\tl_use:N \l__codebox_code_name_tl\nobreakspace\thecvcounter\nobreakspace~#1},
279 }
280 \NewDocumentEnvironment{ codeview }{0}{ m }
281 {
282     \bool_set_false:N \l__codebox_comment_bool
283     \group_begin:
284     \IfNoValueF{#1}{ \keys_set:nn { codebox } { #1 } }
285     \__codebox_code_engine_tcbset:
286     \__codebox_comment_tcbset:
287     \tl_if_empty:NF \l__codebox_label_name_tl
288     {
289         \tcbset{
290             label=\tl_use:N \l__codebox_label_name_tl,
291         }
292     }
293     \tcbset{
294         cv,
295         overlay~unbroken~and~first ={
296             \begin{tcbclipinterior}
297                 \node[inner~sep=0pt,anchor=north~east,yshift=-3pt,xshift=-5pt,text=cvgrayb]
298                 at (frame.north~east){
299                     \ttfamily\faFile* \faCode \faCodeBranch \faCopy \faExternalLink* \
300                     \MakeUppercase{\tl_use:N \l__codebox_language_tl}
301                 };
302             \end{tcbclipinterior}
303         }
304     }%
305     \codeviewaux{#2}
306 }{
307     \endcodeviewaux
308     \group_end:
309 }

```

定义带底线注释的代码盒子环境 codeview\*

```

310 \NewDocumentEnvironment{ codeview* }{0}{ m }
311 {
312     \bool_set_true:N \l__codebox_comment_bool
313     \group_begin:
314     \IfNoValueF{#1}{ \keys_set:nn { codebox } { #1 } }
315     \__codebox_code_engine_tcbset:
316     \__codebox_comment_tcbset:
317     \tl_if_empty:NF \l__codebox_label_name_tl
318     {

```

```

319     \tcbset{
320       label=\tl_use:N \l__codebox_label_name_tl,
321     }
322   }
323   \tcbset{
324     cv,
325     overlay~unbroken~and~first={
326       \begin{tcbclipinterior}
327         \node[inner~sep=0pt,anchor=north~east,yshift=-3pt,xshift=-5pt,text=cvgrayb]
328           at (frame.north~east){
329             \ttfamily\faFile* \faCode \faCodeBranch \faCopy \faExternalLink* \
330             \MakeUppercase{\tl_use:N \l__codebox_language_tl}
331           };
332       \end{tcbclipinterior}
333     }
334   }%
335   \codeviewaux{#2}
336 }{
337   \endcodeviewaux
338   \group_end:
339 }

```

定义带/不带底线注释的从文件读取内容的命令\cvfile/\cvfile\*

```

340 \DeclareTCBInputListing[use~counter=cvcounter]{\langfileaux}{m}
341 {
342   title={\tl_use:N \l__codebox_code_name_tl\nobreakspace\thecvcounter\nobreakspace~#1},
343 }
344 \NewDocumentCommand \cvfile { s O{} m m }
345 {
346   \IfBooleanTF{#1}
347   {
348     \bool_set_true:N \l__codebox_comment_bool
349   }{
350     \bool_set_false:N \l__codebox_comment_bool
351   }
352
353   \group_begin:
354   \IfNoValueF{#2}{ \keys_set:nn { codebox } { #2 } }
355   \__codebox_code_engine_tcbset:
356   \__codebox_comment_tcbset:
357   \tl_if_empty:NF \l__codebox_label_name_tl
358   {
359     \tcbset{
360       label=\tl_use:N \l__codebox_label_name_tl,
361     }
362   }
363   \tcbset{
364     cv,
365     listing~file={#4},
366     overlay~unbroken~and~first={
367       \begin{tcbclipinterior}
368         \node[inner~sep=0pt,anchor=north~east,yshift=-3pt,xshift=-5pt,text=cvgrayb]
369           at (frame.north~east){
370             \ttfamily\faFile* \faCode \faCodeBranch \faCopy \faExternalLink* \
371             \MakeUppercase{\tl_use:N \l__codebox_language_tl}
372           };
373       \end{tcbclipinterior}
374     }
375   }%
376   \langfileaux{#3}
377   \group_end:
378 }
379 </package>

```

## 版本历史

<b>v1.0.0</b>	(2021/12/25)	修改按章编号时 cvcounter 的编号方式	12
General: 开始编写模板	1	将 minted/listings 的参数设置独立为函数	12
<b>v1.0.1</b>	(2021/12/26 – 2021/12/27)	添加 cvcounter 使用说明	5
General: minted2.6 发布, 删除原补丁代码	9	<b>v1.0.3</b>	(2022/1/4)
修订代码盒子前多一个逗号的问题	13	General: 为带编号代码盒子添加引用标签选项	14
删除原 L <sup>A</sup> T <sub>E</sub> X3 版本检测代码	8	增加引用标签 label 选项	11
调整 minted style 选项为参数实现	12	添加 label 引用标签选项的使用说明	5
<b>v1.0.2</b>	(2022/1/2 – 2021/12/28)	\l__codebox_linenumbersep_fp: 增	
General: 为计数器两端添加\nobreakspace	14	加\_ @@_label_name_tl 变量	10

## 代码索引

意大利体的数字表示描述对应索引项的页码; 带下划线的数字表示定义对应索引项的代码行号; 罗马字体的数字表示使用对应索引项的代码行号。

\_	299, 329, 370	\_codebox_minted_engine_tcbset:n	160, 177, 208
		\_codebox_minted_engine_tcbset:n#1	160
		\_codebox_set_counter_parent:	150, 275
<b>A</b>		codebox*	2
\arabic	155	\codefile	2, 254
		\codefile*	2
<b>B</b>		\codeset	148
\begin	76, 102, 192, 296, 326, 367	codesize	5, 131
\bfseries	48, 182	codestretch	5, 137
bool commands:		codestyle	5, 129
\bool_if:NTF	206, 215	codeview	3
\bool_new:N	108, 109	codeview*	3
\bool_set_false:N	230, 243, 260, 282, 350	\codeviewaux	305, 335
\bool_set_true:N	258, 312, 348	\color	182, 185, 188, 200
<b>C</b>		commentf	5, 135
codebox	2	comments	5, 133
codebox internal commands:		CS commands:	
\l__codebox_baseline_stretch_fp	108, 137, 169	\cs_generate_variant:Nn	177
\_codebox_code_engine_tcbset:		\cs_new:Npn	150, 160, 178, 204, 213
	204, 233, 246, 265, 285, 315, 355	cvcounter	6
\l__codebox_code_fontsize_tl	108, 131, 168, 181, 188	\cvfile	4, 344
\l__codebox_code_name_tl	108, 127, 278, 342	\cvfile*	4
\l__codebox_code_style_tl	108, 129, 208		
\l__codebox_comment_bool		<b>D</b>	
	108, 215, 230, 243, 258, 260, 282, 312, 348, 350	\DeclareTCBInputListing	340
\l__codebox_comment_contents_tl	108, 133, 222	\DeclareTCBListing	276
\l__codebox_comment_format_tl	108, 135, 221	\definecolor	32, 33, 34, 35, 36
\_codebox_comment_tcbset:			
	213, 234, 247, 266, 286, 316, 356	<b>E</b>	
\_codebox_error:n	144	\end	79, 104, 195, 302, 332, 373
\l__codebox_label_name_tl		\endcodeviewaux	307, 337
	108, 142, 287, 290, 317, 320, 357, 360	\endtcblisting	238, 251
\l__codebox_language_tl	108, 125, 174, 198, 300, 330, 371	\ExplSyntaxOff	5
\l__codebox_linenumbersep_fp	108, 140, 172, 199	\ExplSyntaxOn	28
\_codebox_listings_engine_tcbset:	178, 210		
\l__codebox_minted_bool	108, 122, 206	<b>F</b>	
		\faCode	299, 329, 370



- `\faCodeBranch` ..... 299, 329, 370  
`\faCopy` ..... 299, 329, 370  
`\faExternalLink` ..... 299, 329, 370  
`\faFile` ..... 299, 329, 370  
`\fi` ..... 200  
`\fill` ..... 77, 103, 193  
**fp commands:**  
`\fp_eval:n` ..... 169, 172, 199  
`\fp_new:N` ..... 118, 119  
`\fp_set:Nn` ..... 137
- G**
- group commands:**  
`\group_begin:` ..... 231, 244, 263, 283, 313, 353  
`\group_end:` ..... 239, 252, 273, 308, 338, 377
- I**
- `\IfBooleanTF` ..... 256, 346  
`\ifdef` ..... 152  
`\IfNoValueF` ..... 232, 245, 264, 284, 314, 354  
`\ifodd` ..... 200  
`\input` ..... 9, 13, 20, 24  
`\itshape` ..... 185
- K**
- keys commands:**  
`\keys_define:nn` ..... 120  
`\l_keys_key_tl` ..... 147  
`\keys_set:nn` ..... 149, 232, 245, 264, 284, 314, 354
- L**
- `label` ..... 5, 142  
`lang` ..... 4, 125  
`\langfileaux` ..... 340, 376  
`linenumsep` ..... 5, 140  
`\lstset` ..... 180
- M**
- `\MakeUppercase` ..... 300, 330, 371  
`minted` ..... 4, 122  
**msg commands:**  
`\msg_new:nnn` ..... 146
- N**
- `\newcounter` ..... 154, 157  
`\NewDocumentCommand` ..... 148, 254, 344  
`\NewDocumentEnvironment` ..... 228, 241, 280, 310  
`\nobreakspace` ..... 278, 342  
`\node` ..... 297, 327, 368
- P**
- `\patchcmd` ..... 6, 18  
`\path` ..... 61, 64  
`pretitle` ..... 4, 127
- R**
- `\relax` ..... 9, 13
- `\renewcommand` ..... 155  
`\RequirePackage` ..... 3, 4
- S**
- `\sffamily` ..... 136, 188  
`\small` ..... 132, 136
- T**
- `\tcbininputlisting` ..... 272  
`\tcblisting` ..... 236, 249  
`\tcboxedtitleheight` ..... 51, 52, 56  
`\tcbpkgprefix` ..... 9, 13  
`\tcbset` ..... 37, 83, 162, 190, 217, 225, 235, 248, 267, 289, 293, 319, 323, 359, 363  
`\tcbuselibrary` ..... 29, 30  
**T<sub>E</sub>X and L<sup>A</sup>T<sub>E</sub>X<sub>2</sub> $\epsilon$  commands:**  
`\@currname` ..... 23  
`\@popfilename` ..... 14, 25  
`\@pushfilename` ..... 12, 22  
`\arabic{cvcounter}` ..... 6  
`\begin` ..... 2, 3  
`\codefile` ..... 1, 2, 13  
`\codefile*` ..... 1, 2, 13  
`\codeset` ..... 2–4, 7  
`\cvfile` ..... 1, 4–6, 15  
`\cvfile*` ..... 1, 4–6, 15  
`\end` ..... 2, 3  
`\nobreakspace` ..... 8  
`\pgfutil@InputIfFileExists` ..... 19  
`\ref` ..... 7  
`\renewcommand{\thecvcounter}{\thechapter.\arabic{cvcounter}}` ..... 6  
`\sffamily` ..... 5  
`\small` ..... 5  
`\tcb@input@library@in` ..... 7  
`\thechapter.\arabic{cvcounter}` ..... 6  
`\thecvcounter` ..... 6  
`\thechapter` ..... 152, 155  
`\thecvcounter` ..... 155, 278, 342  
**tl commands:**  
`\tl_if_empty:NTF` ..... 287, 317, 357  
`\tl_new:N` ..... 111, 112, 113, 114, 115, 116, 117  
`\tl_use:N` ..... 168, 174, 181, 188, 198, 221, 222, 278, 290, 300, 320, 330, 342, 360, 371  
`\ttfamily` ..... 99, 181, 186, 299, 329, 370
- U**
- `\usetikzlibrary` ..... 31
- V**
- `\value` ..... 200
- X**
- `\xdef` ..... 23