The `tcolorbox` package

Manual for version 2.51 (2013/09/16)

Thomas F. Sturm[1]

Abstract

`tcolorbox` provides an environment for colored and framed text boxes with a heading line. Optionally, such a box can be split in an upper and a lower part. The package `tcolorbox` can be used for the setting of LaTeX examples where one part of the box displays the source code and the other part shows the output. Another common use case is the setting of theorems. The package supports saving and reuse of source code and text parts.

Contents

---

[1]Prof. Dr. Dr. Thomas F. Sturm, Institut für Mathematik und Informatik, Universität der Bundeswehr München, D-85577 Neubiberg, Germany; email: thomas.sturm@unibw.de

1 Introduction

The package originates from the first edition of my book « *LATEX – Einführung in das Textsatzsystem* » [16] in about 2006. For the LATEX examples and tutorials given there, I wanted to have accentuated and colored boxes to display source code and compiled text in combination. Since, in my opinion, this type of boxes is also quite useful to highlight definitions and theorems, I applied them for my lecture notes in mathematics [13–15] as well. With this package, you are invited to apply these boxes for similar projects.

Starting with version 2.00, all internal calculations are not longer performed by the package `calc` [18] but by ε-TEX [1] expressions since ε-TEX is assumed to be used by everybody (hopefully). The breaking news for version 2.00 is the support for breakable boxes. This new feature allows new applications of the package without affecting the core package too much if you do not need boxes to break automatically. With version 2.20, the often requested 'side by side' mode for listings has been added.

Since the first public release in 2011, I received a lot of feedback from all over the world. I want to thank all who wrote me for supporting this package by sending bug reports and ideas for new or better features.
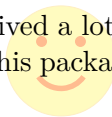
1.1 Installation

Typically, `tcolorbox` will be installed as part of a major LATEX distribution and there is nothing special to do for a user.

If you intend to make a local installation *by hand*, see the `README` file of the `tcolorbox` package for some hints. The short story is: you have to install not only `tcolorbox.sty`, but also all `*.code.tex` files in the local `texmf` tree.

1.2 Loading the Package

The base package `tcolorbox` loads the packages `pgf` [17], `verbatim` [12], `etoolbox` [6], and `environ` [10]. `tcolorbox` itself is loaded in the usual manner in the preamble:

```
\usepackage{tcolorbox}
```

The package takes option keys in the key-value syntax. Alternatively, you may use these keys later in the preamble with `\tcbuselibrary`→P.5 (see there). For example, the key to typeset listings is:

```
\usepackage[listings]{tcolorbox}
```

1.3 Libraries

The base package `tcolorbox` is extendable by program libraries. This is done by using option keys while loading the package or inside the preamble by applying the following macro with the same set of keys.

\tcbuselibrary{⟨*key list*⟩**}**

Loads the libraries given by the ⟨*key list*⟩.

```
\tcbuselibrary{listings,theorems}
```

The following keys are used inside **\tcbuselibrary** respectively **\usepackage** without the key tree path `/tcb/library/`.

/tcb/library/skins (LIB skins)

Loads the package `tikz` [17] and provides additional styles (skins) for the appearance of the colored boxes; see Section 6 from page 71.

/tcb/library/listings (LIB listings)

Loads the package `listings` [5] and provides additional macros for typesetting listings which are described in Section 7 from page 130.

/tcb/library/listingsutf8 (LIB listingsutf8)

Loads the packages `listings` [5] and `listingsutf8` [8] for UTF-8 support. This is a variant of the library LIB listings and is described in Section 7 from page 130.

/tcb/library/minted (LIB minted)

Loads the package `minted` [11] to typeset listings with the `Pygments` [9] tool, also see Section 7 on page 130.

/tcb/library/theorems (LIB theorems)

Provides additional macros for typesetting theorems which are described in Section 8 from page 155.

/tcb/library/breakable (LIB breakable)

Provides support for automatic box breaking from one page to another; see Section 9 from page 169.

/tcb/library/fitting (LIB fitting)

Provides support for font size adaption of the box content to the box dimensions; see Section 10 from page 181.

/tcb/library/hooks (LIB hooks)

Extends several option keys to 'hookable' keys; see Section 11 from page 190.

/tcb/library/documentation (LIB documentation)

Provides additional macros for typesetting LaTeX documentations which are described in Section 12 from page 198.

/tcb/library/many (style, no value)

Loads the libraries ⬛ skins , ⬛ breakable , ⬛ hooks , ⬛ theorems , and ⬛ fitting . Use this shortcut, if you want to use all features of `tcolorbox` with exception of typesetting listings and using the specialized ⬛ documentation library.

/tcb/library/most (style, no value)

Loads all libraries except ⬛ minted and ⬛ documentation . Use this shortcut, if you want to use all features of `tcolorbox` with exception of using the `minted` package and using the specialized ⬛ documentation library.

/tcb/library/all (style, no value)

Loads all libraries. Use this shortcut only, if you indend to use the ⬛ documentation library.

6
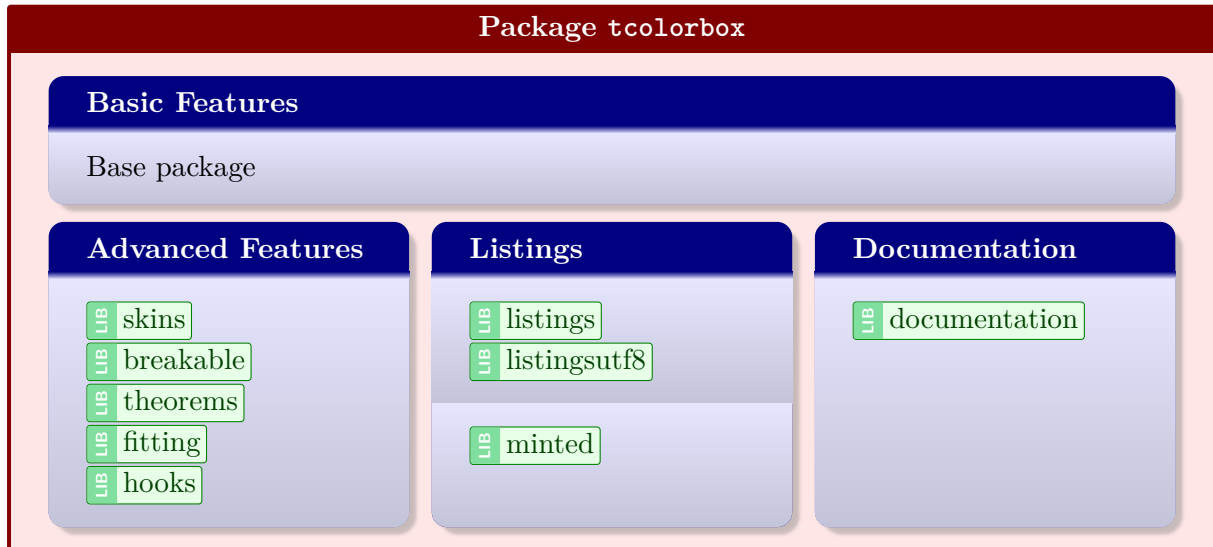
2 Macros for Box Creation

\begin{**tcolorbox**}[⟨*options*⟩]
 ⟨*environment content*⟩
\end{**tcolorbox**}

This is the main environment to create an accentuated colored text box with rounded corners and, optionally, two parts. The appearance of this box is controlled by numerous options. In the most simple case the source code

```
\begin{tcolorbox}
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

creates the following compiled text box:

> This is a **tcolorbox**.

The text content of the box can be divided in an upper and a lower part by the command \tcblower. Visually, both parts are separated by a line. For example:

```
\begin{tcolorbox}
This is another \textbf{tcolorbox}.
\tcblower
Here, you see the lower part of the box.
\end{tcolorbox}
```

This code gives the following box:

> This is another **tcolorbox**.
> _____
> Here, you see the lower part of the box.

The ⟨*options*⟩ control the appearance and several functions of the boxes, see section 3 for the complete list. A quick example is given here:

```
\begin{tcolorbox}[colback=red!5!white,colframe=red!75!black,title=My nice heading]
This is another \textbf{tcolorbox}.
\tcblower
Here, you see the lower part of the box.
\end{tcolorbox}
```

> **My nice heading**
>
> This is another **tcolorbox**.
> _____
> Here, you see the lower part of the box.

\tcblower

Used inside tcolorbox to separate the upper box part from the optional lower box part. The upper and the lower part are treated as separate functional units. If you only want to draw a line, see \tcbline^{→ P. 102}.

\tcbset{⟨*options*⟩}

Sets options for every following `tcolorbox`$^{→P.7}$ inside the current TEX group. By default, this does not apply to nested boxes, see Section 3.12 from page 56.

For example, the colors of the boxes may be defined for the whole document by this:

```
\tcbset{colback=red!5!white,colframe=red!75!black}
```

\tcbox[⟨*options*⟩]{⟨*box content*⟩}

Creates a colored box which is fitted to the width of the given ⟨*box content*⟩. In principal, most ⟨*options*⟩ for a `tcolorbox`$^{→P.7}$ can be used for **\tcbox** with some restrictions. A **\tcbox** cannot have a lower part and cannot be broken.

```
\tcbset{colframe=blue!50!black,colback=white,colupper=red!50!black,
        fonttitle=\bfseries,nobeforeafter,center title}

Text \tcbox[tcbox raise base]{Hello World}\hfill
%
\tcbox[left=0mm,right=0mm,top=0mm,bottom=0mm,boxsep=0mm,
  toptitle=0.5mm,bottomtitle=0.5mm,title=My table]{%
  \arrayrulecolor{blue!50!black}\renewcommand{\arraystretch}{1.2}%
  \begin{tabular}{r|c|l}
  One   & Two    & Three \\\hline\hline
  Men   & Mice   & Lions \\\hline
  Upper & Middle & Lower
  \end{tabular}}\hfill
%
\tcbox[colback=blue!85!black,
  left=0mm,right=0mm,top=0mm,bottom=0mm,boxsep=1mm,arc=0mm,boxrule=0.5pt,
  title=My picture]{%
  \includegraphics[width=5cm]{Basilica_5.png}}
```

\newtcolorbox[⟨*init options*⟩]{⟨*name*⟩}[⟨*number*⟩][⟨*default*⟩]{⟨*options*⟩}
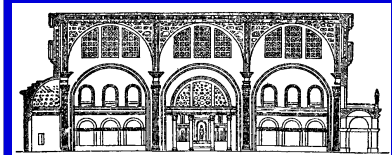
Creates a new environment ⟨*name*⟩ based on tcolorbox[→P.7]. Basically, **\newtcolorbox** operates like **\newenvironment**. This means, the new environment ⟨*name*⟩ optionally takes ⟨*number*⟩ arguments, where ⟨*default*⟩ is the default value for the optional first argument. The ⟨*options*⟩ are given to the underlying tcolorbox. Note that **/tcb/savedelimiter**[→P.14] is set to the given ⟨*name*⟩ automatically. The ⟨*init options*⟩ allow to set up automatic numbering, see Section 4 from page 65.

```
\newtcolorbox{mybox}{colback=red!5!white,
  colframe=red!75!black}

\begin{mybox}
This is my own box.
\end{mybox}
```

This is my own box.

```
\newtcolorbox{mybox}[1]{colback=red!5!white,
  colframe=red!75!black,fonttitle=\bfseries,
  title=#1}

\begin{mybox}{Hello there}
This is my own box with a mandatory title.
\end{mybox}
```

Hello there

This is my own box with a mandatory title.

```
\newtcolorbox{mybox}[2][]{colback=red!5!white,
  colframe=red!75!black,fonttitle=\bfseries,
  title=#2,#1}

\begin{mybox}[colback=yellow]{Hello there}
This is my own box with a mandatory title
and options.
\end{mybox}
```

Hello there

This is my own box with a mandatory title and options.

Definition in the preamble:

```
\newtcolorbox[auto counter,number within=section]{pabox}[2][]{%
  colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries,
  title=Examp.~\thetcbcounter: #2,#1}
```

```
\begin{pabox}[colback=yellow]{Hello there}
This is my own box with a mandatory
numbered title and options.
\end{pabox}
```

Examp. 2.1: Hello there

This is my own box with a mandatory numbered title and options.

\renewtcolorbox[⟨*init options*⟩]{⟨*name*⟩}[⟨*number*⟩][⟨*default*⟩]{⟨*options*⟩}

Operates like **\newtcolorbox**, but based on **\renewenvironment** instead of **\newenvironment**. An existing environment is redefined.

\newtcbox[⟨*init options*⟩]{\\⟨*name*⟩}[⟨*number*⟩][⟨*default*⟩]{⟨*options*⟩}

Creates a new macro \⟨*name*⟩ based on **\tcbox**^{→P.8}. Basically, **\newtcbox** operates like **\newcommand**. The new macro \⟨*name*⟩ optionally takes ⟨*number*⟩+1 arguments, where ⟨*default*⟩ is the default value for the optional first argument. The ⟨*options*⟩ are given to the underlying **tcbox**. The ⟨*init options*⟩ allow to set up automatic numbering, see Section 4 from page 65.

```
\newtcbox{\mybox}{colback=red!5!white,
  colframe=red!75!black}

\mybox{This is my own box.}
```
This is my own box.

```
\newtcbox{\mybox}[1]{colback=red!5!white,
  colframe=red!75!black,fonttitle=\bfseries,
  title=#1}

\mybox{Hello there}{This is my own box.}
```
Hello there

This is my own box.

```
\newtcbox{\mybox}[2][]{colback=red!5!white,
  colframe=red!75!black,fonttitle=\bfseries,
  title=#2,#1}

\mybox[colback=yellow]{Hello there}%
  {This is my own box.}
```
Hello there

This is my own box.

Definition in the preamble:

```
% counter from previous example
\newtcbox[use counter from=pabox]{\pbbox}[2][]{%
  colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries,
  title=(\thetcbcounter) #2,#1}
```

```
\pbbox[colback=yellow]{Hello there}%
  {This is my own box.}
```
(2.2) Hello there

This is my own box.

```
\newtcbox{\mybox}[1][red]{nobeforeafter,tcbox raise base,
  arc=0pt,outer arc=0pt,colback=#1!10!white,colframe=#1!50!black,
  boxsep=0pt,left=1pt,right=1pt,top=2pt,bottom=2pt,
  boxrule=0pt,bottomrule=1pt,toprule=1pt}
\newtcbox{\xmybox}[1][red]{nobeforeafter,tcbox raise base,
  arc=7pt,colback=#1!10!white,colframe=#1!50!black,
  before upper={\rule[-3pt]{0pt}{10pt}},boxrule=1pt,
  boxsep=0pt,left=6pt,right=6pt,top=2pt,bottom=2pt}

The \mybox[green]{quick} brown \mybox{fox} \mybox[blue]{jumps} over the
\mybox[green]{lazy} \mybox{dog}.\par
The \xmybox[green]{quick} brown \xmybox{fox} \xmybox[blue]{jumps} over the
\xmybox[green]{lazy} \xmybox{dog}.
```

The quick brown fox jumps over the lazy dog.
The quick brown fox jumps over the lazy dog.

\renewtcbox[⟨*init options*⟩]{\\⟨*name*⟩}[⟨*number*⟩][⟨*default*⟩]{⟨*options*⟩}

Operates like **\newtcbox**, but based on **\renewcommand** instead of **\newcommand**. An existing macro is redefined.

3 Option Keys

For the ⟨*options*⟩ in `tcolorbox`^{→P. 7} respectively `\tcbset`^{→P. 8} the following `pgf` keys can be applied. The key tree path `/tcb/` is not to be used inside these macros. It is easy to add your own style keys using the syntax for `pgf` keys, see [16, 17] or the examples starting from page 142.

3.1 Title

/tcb/title=⟨*text*⟩   (no default, initially empty)

Creates a heading line with ⟨*text*⟩ as content.

```
\begin{tcolorbox}[title=My heading line]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

> **My heading line**
>
> This is a **tcolorbox**.

/tcb/notitle   (no value, initially set)

Removes the title line if set before.

/tcb/adjusted title=⟨*text*⟩   (style, no default, initially unset)

Creates a heading line with ⟨*text*⟩ as content. The minimal height of this line is adjusted to fit the text given by `/tcb/adjust text`. This option makes sense for single line headings if boxes are set side by side with equal height. Note that it is very easy to trick this adjustment.

```
\tcbset{colback=White,arc=0mm,width=(\linewidth-4pt)/4,
equal height group=AT,before=,after=\hfill,fonttitle=\bfseries}

The following titles are not adjusted:\\
\foreach \n in {xxx,ggg,AAA,\"Agypten}
{\begin{tcolorbox}[title=\n,colframe=red!75!black]
  Some content.\end{tcolorbox}}
Now, we try again with adjusted titles:\\
\foreach \n in {xxx,ggg,AAA,\"Agypten}
{\begin{tcolorbox}[adjusted title=\n,colframe=blue!75!black]
  Some content.\end{tcolorbox}}
```

The following titles are not adjusted:

xxx	**ggg**	**AAA**	**Ägypten**
Some content.	Some content.	Some content.	Some content.

Now, we try again with adjusted titles:

xxx	**ggg**	**AAA**	**Ägypten**
Some content.	Some content.	Some content.	Some content.

/tcb/adjust text=⟨*text*⟩   (no default, initially Äpgjy)

This sets the reference text for `/tcb/adjusted title`. If your texts never exceed 'Äpgjy' in depth and height you don't need to care about this option.

3.2 Lower Part

/tcb/lowerbox=⟨*mode*⟩   (no default, initially `visible`)

Controls the treatment of the lower part of the box. Feasible values for ⟨*mode*⟩ are:

- `visible`: usual type setting of the lower part,
- `invisible`: empty space instead of the lower part contents,
- `ignored`: the lower part is not used (here).

The last two values are usually applied in connection with `savelowerto`.

```
\begin{tcolorbox}[lowerbox=invisible,colback=white]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part (but invisible).
\end{tcolorbox}

\begin{tcolorbox}[lowerbox=ignored,colback=white]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part (but ignored).
\end{tcolorbox}
```

> This is a **tcolorbox**.
> -
>
>

> This is a **tcolorbox**.

/tcb/savelowerto=⟨*file name*⟩   (no default, initially empty)

Saves the content of the lower part in a file for an optional later usage.

```
\begin{tcolorbox}[lowerbox=invisible,savelowerto=\jobname_bspsave.tex,colback=white]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part which may be quite complex:
$\displaystyle f(x)=\frac{1+x^2}{1-x^2}$.
\end{tcolorbox}

Now, we load the saved text:\\
\input{\jobname_bspsave.tex}
```

> This is a **tcolorbox**.
> -
>
>
>

Now, we load the saved text:

This is the lower part which may be quite complex: $f(x) = \dfrac{1 + x^2}{1 - x^2}$.

/tcb/lower separated=⟨*boolean value*⟩ (default `true`, initially `true`)

If set to `true`, the lower part is visually separated from the upper part. It depends from the chosen skin how the visualization of the separation is done.

```
% \tcbuselibrary{skins}
\tcbset{colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries,nobeforeafter,
  width=(\linewidth-4mm)/2,fontlower=\itshape}

\begin{tcolorbox}[title=Lower separated]
This is the upper part.
\tcblower
This is the lower part.
\end{tcolorbox}\hfill
\begin{tcolorbox}[title=Lower not separated,lower separated=false]
This is the upper part.
\tcblower
This is the lower part.
\end{tcolorbox}
\par\bigskip\par
\begin{tcolorbox}[sidebyside,title=Lower separated]
This is the upper part.
\tcblower
This is the lower part.
\end{tcolorbox}\hfill
\begin{tcolorbox}[sidebyside,title=Lower not separated,lower separated=false]
This is the upper part.
\tcblower
This is the lower part.
\end{tcolorbox}
\par\bigskip\par
\begin{tcolorbox}[beamer,title=Lower separated]
This is the upper part.
\tcblower
This is the lower part.
\end{tcolorbox}\hfill
\begin{tcolorbox}[beamer,title=Lower not separated,lower separated=false]
This is the upper part.
\tcblower
This is the lower part.
\end{tcolorbox}
```

Lower separated	**Lower not separated**
This is the upper part.	This is the upper part.
This is the lower part.	*This is the lower part.*

Lower separated	**Lower not separated**
This is the upper part. / *This is the lower part.*	This is the upper part. / *This is the lower part.*

Lower separated	**Lower not separated**
This is the upper part.	This is the upper part.
This is the lower part.	*This is the lower part.*

/tcb/savedelimiter=⟨*name*⟩   (no default, initially `tcolorbox`)

Used in connection with new environment definitions which extend `tcolorbox` and use or allow the option `savelowerto`. To catch the end of the new box environment ⟨*name*⟩ has to be the name of this environment. Additionally, the environment definition has to use `\tcolorbox` instead of `\begin{tcolorbox}` and `\endtcolorbox` instead of `\end{tcolorbox}`.

```
\newenvironment{mybox}[1]{%
  \tcolorbox[savedelimiter=mybox,
            savelowerto=\jobname_bspsave2.tex,lowerbox=ignored,
            colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries,
            title=#1]}%
  {\endtcolorbox}

\begin{mybox}{My Example}
Upper part.
\tcblower
Saved lower part!
\end{mybox}

Now, the saved part is used:
\begin{tcolorbox}[colback=green!5]
\input{\jobname_bspsave2.tex}
\end{tcolorbox}
```

> **My Example**
>
> Upper part.

Now, the saved part is used:

> Saved lower part!

The `savedelimiter` is used implicitly with `\newtcolorbox`[→P. 9] which allows a more convenient usage:

```
\newtcolorbox{mybox}[1]{%
            savelowerto=\jobname_bspsave2.tex,lowerbox=ignored,
            colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries,
            title=#1}%

\begin{mybox}{My Example}
Upper part.
\tcblower
Saved lower part!
\end{mybox}

Now, the saved part is used:
\begin{tcolorbox}[colback=green!5]
\input{\jobname_bspsave2.tex}
\end{tcolorbox}
```

> **My Example**
>
> Upper part.

Now, the saved part is used:

> Saved lower part!

3.3 Colors and Fonts

/tcb/colback=⟨*color*⟩ (no default, initially `black!5!white`)

Sets the background ⟨*color*⟩ of the box.

```
\begin{tcolorbox}[colback=white]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

> This is a **tcolorbox**.

/tcb/colframe=⟨*color*⟩ (no default, initially `black!75!white`)

Sets the frame ⟨*color*⟩ of the box.

```
\begin{tcolorbox}[colframe=red!50!yellow]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

> This is a **tcolorbox**.

/tcb/colupper=⟨*color*⟩ (no default, initially `black`)

Sets the text ⟨*color*⟩ of the upper part.

```
\begin{tcolorbox}[colupper=yellow,
  colback=blue!50,colframe=blue]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

> This is a **tcolorbox**.
> This is the lower part.

/tcb/collower=⟨*color*⟩ (no default, initially `black`)

Sets the text ⟨*color*⟩ of the lower part.

```
\begin{tcolorbox}[collower=yellow,
  colback=blue!50,colframe=blue]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

> This is a **tcolorbox**.
> This is the lower part.

/tcb/coltext=⟨*color*⟩ (style, no default, initially `black`)

Sets the text ⟨*color*⟩ of the box. This is an abbreviation for setting `colupper` and `collower` to the same value.

```
\begin{tcolorbox}[coltext=yellow,
  colback=blue!50,colframe=blue]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

> This is a **tcolorbox**.
> This is the lower part.

/tcb/coltitle=⟨*color*⟩ (no default, initially `white`)

Sets the title text ⟨*color*⟩ of the box.

```
\begin{tcolorbox}[coltitle=blue!50!black,
  colframe=blue!25,title=Test]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

> Test
> This is a **tcolorbox**.

/tcb/fontupper=⟨*text*⟩ (no default, initially empty)

Sets ⟨*text*⟩ before the content of the upper part (e. g. font settings).

```
\begin{tcolorbox}[fontupper=Hello!~\sffamily]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

Hello! This is a **tcolorbox**.

/tcb/fontlower=⟨*text*⟩ (no default, initially empty)

Sets ⟨*text*⟩ before the content of the lower part (e. g. font settings).

```
\begin{tcolorbox}[fontlower=\sffamily\bfseries]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

This is a **tcolorbox**.

This is the lower part.

/tcb/fonttitle=⟨*text*⟩ (no default, initially empty)

Sets ⟨*text*⟩ before the content of the title text (e. g. font settings).

```
\begin{tcolorbox}[fonttitle=\sffamily\bfseries\large,title=Hello]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

Hello

This is a **tcolorbox**.

More color options are provided by using skins documented in Section 6 from page 71.

3.4 Geometry

/tcb/width=⟨*length*⟩ (no default, initially \linewidth)
Sets the total width of the colored box to ⟨*length*⟩. See also /tcb/height^{→P.27}.

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[width=\linewidth/2]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

> This is a **tcolorbox**.

/tcb/toprule=⟨*length*⟩ (no default, initially 0.5mm)
Sets the line width of the top rule to ⟨*length*⟩.

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[toprule=3mm]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

> This is a **tcolorbox**.

/tcb/bottomrule=⟨*length*⟩ (no default, initially 0.5mm)
Sets the line width of the bottom rule to ⟨*length*⟩.

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[bottomrule=3mm]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

> This is a **tcolorbox**.

/tcb/leftrule=⟨*length*⟩ (no default, initially 0.5mm)
Sets the line width of the left rule to ⟨*length*⟩.

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[leftrule=3mm]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

> This is a **tcolorbox**.

17

/tcb/rightrule=⟨*length*⟩   (no default, initially `0.5mm`)

Sets the line width of the right rule to ⟨*length*⟩.

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[rightrule=3mm]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

> This is a **tcolorbox**.

/tcb/titlerule=⟨*length*⟩   (no default, initially `0.5mm`)

Sets the line width of the rule below the title to ⟨*length*⟩.

```
\tcbset{enhanced,colback=red!5!white,colframe=red!75!black,
  colbacktitle=red!90!black}

\begin{tcolorbox}[titlerule=3mm,title=This is the title]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

> This is the title
>
> This is a **tcolorbox**.

/tcb/boxrule=⟨*length*⟩   (style, no default, initially `0.5mm`)

Sets all rules of the frame to ⟨*length*⟩, i. e. /tcb/toprule^{→P. 17}, /tcb/bottomrule^{→P. 17}, /tcb/leftrule^{→P. 17}, /tcb/rightrule, and /tcb/titlerule.

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[boxrule=3mm]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

> This is a **tcolorbox**.

More options for drawing a /tcb/borderline^{→P. 91} are provided by using skins documented in Section 6 from page 71.

/tcb/arc=⟨*length*⟩ (no default, initially 1mm)

Sets the inner radius of the four frame arcs to ⟨*length*⟩.

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[arc=0mm]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
\begin{tcolorbox}[arc=4mm]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a **tcolorbox**.

This is a **tcolorbox**.

/tcb/outer arc=⟨*length*⟩ (no default, initially unset)

Sets the outer radius of the four frame arcs to ⟨*length*⟩.

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[arc=4mm,outer arc=1mm]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a **tcolorbox**.

/tcb/auto outer arc (no value, initially set)

Sets the outer radius of the four frame arcs automatically in dependency of the inner radius given by /tcb/arc.

/tcb/boxsep=⟨*length*⟩ (no default, initially 1mm)

Sets a common padding of ⟨*length*⟩ between the text content and the frame of the box. This value is added to the key values of left, right, top, bottom, and middle at the appropriate places.

```
\tcbset{colback=red!5!white,colframe=red!75!black,width=(\linewidth-4mm)/2,
  before=,after=\hfill}

\begin{tcolorbox}[boxsep=5mm]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
\begin{tcolorbox}[boxsep=5mm,draft]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

frame: w=195.33255pt, h=48.97505pt

This is a **tcolorbox**.

upper: w=141.2724pt, h=6.296pt

interior: w=192.4873pt, h=46.1298pt

/tcb/left=⟨*length*⟩ (style, no default, initially 4mm)

Sets the left space between all text parts and frame (additional to `boxsep`). This is an abbreviation for setting `lefttitle`, `leftupper`, and `leftlower` to the same value.

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[left=0mm]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a **tcolorbox**.

/tcb/lefttitle=⟨*length*⟩ (no default, initially 4mm)

Sets the left space between title text and frame (additional to `boxsep`).

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[lefttitle=3cm,title=My Title]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

My Title

This is a **tcolorbox**.

/tcb/leftupper=⟨*length*⟩ (no default, initially 4mm)

Sets the left space between upper text and frame (additional to `boxsep`).

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[leftupper=3cm,title=My Title]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

My Title

This is a **tcolorbox**.

/tcb/leftlower=⟨*length*⟩ (no default, initially 4mm)

Sets the left space between lower text and frame (additional to `boxsep`).

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[leftlower=3cm]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

This is a **tcolorbox**.

This is the lower part.

/tcb/right=⟨*length*⟩   (style, no default, initially 4mm)

Sets the right space between all text parts and frame (additional to `boxsep`). This is an abbreviation for setting `righttitle`, `rightupper`, and `rightlower` to the same value.

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[width=5cm,right=2cm]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a **tcolorbox**.

/tcb/righttitle=⟨*length*⟩   (no default, initially 4mm)

Sets the right space between title text and frame (additional to `boxsep`).

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[width=5cm,righttitle=2cm,title=My very long title text]
This is a \textbf{tcolorbox} with standard upper box dimensions.
\end{tcolorbox}
```

My very long title text

This is a **tcolorbox** with standard upper box dimensions.

/tcb/rightupper=⟨*length*⟩   (no default, initially 4mm)

Sets the right space between upper text and frame (additional to `boxsep`).

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[width=5cm,rightupper=2cm,title=My very long title text]
This is a \textbf{tcolorbox} with compressed upper box dimensions.
\end{tcolorbox}
```

My very long title text

This is a **tcolorbox** with compressed upper box dimensions.

/tcb/rightlower=⟨*length*⟩   (no default, initially 4mm)
Sets the right space between lower text and frame (additional to `boxsep`).

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[width=5cm,rightlower=2cm]
This is a \textbf{tcolorbox} with standard upper box dimensions.
\tcblower
This is the lower part with large space at right.
\end{tcolorbox}
```

This is a **tcolorbox** with standard upper box dimensions.

This is the lower part with large space at right.

/tcb/top=⟨*length*⟩   (no default, initially 2mm)
Sets the top space between text and frame (additional to `boxsep`).

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[top=0mm]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

This is a **tcolorbox**.

This is the lower part.

/tcb/toptitle=⟨*length*⟩   (no default, initially 0mm)
Sets the top space between title and frame (additional to `boxsep`).

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[toptitle=3mm,title=My title]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

My title

This is a **tcolorbox**.

/tcb/bottom=⟨*length*⟩ (no default, initially 2mm)
    Sets the bottom space between text and frame (additional to `boxsep`).

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[bottom=0mm]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

> This is a **tcolorbox**.
>
> This is the lower part.

/tcb/bottomtitle=⟨*length*⟩ (no default, initially 0mm)
    Sets the bottom space between title and frame (additional to `boxsep`).

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[bottomtitle=3mm,title=My title]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

> My title
>
> This is a **tcolorbox**.

/tcb/middle=⟨*length*⟩ (no default, initially 2mm)
    Sets the space between upper and lower text to the separation line (additional to `boxsep`).

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[middle=0mm,boxsep=0mm]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

> This is a **tcolorbox**.
> This is the lower part.

`/tcb/oversize`=⟨*length*⟩ (style, default `0pt`)

Sets the text width of the upper part to the current line width plus an optional ⟨*length*⟩. This is achieved by changing the keys `/tcb/width`^{→P. 17} `/tcb/enlarge left by`^{→P. 52}, and `/tcb/enlarge right by`^{→P. 52} appropriately. The resulting box is overlapping into the left and right margin of the page. Note that this style option has to be given *after* all other geometry keys!

```
\tcbset{colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries}

\textit{Normal text for comparison:}\\
\lipsum[2]

\begin{tcolorbox}[oversize,title=Oversized box]
\lipsum[2]
\end{tcolorbox}

\begin{tcolorbox}[title=Normal box]
\lipsum[2]
\end{tcolorbox}
```

Normal text for comparison:
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.

Oversized box

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.

Normal box

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.

/tcb/toggle left and right=⟨*toggle preset*⟩ (default `evenpage`, initially `none`)

According to the ⟨*toggle preset*⟩, the left and the right settings of the `tcolorbox` are switched or not. Feasible values are:

- `none`: no switching.
- `forced`: the values of the left and right rules and spaces are switched.
- `evenpage`: if the page is an even page, the values of the left and right rules and spaces are switched. It is recommended to use this setting in conjunction with `/tcb/check odd page`^{→ P.63}.

```
% \usepackage{changepage} for 'check odd page'
% \usepackage{lipsum}
% \usetikzlibrary{patterns}
% \tcbuselibrary{skins,breakable,hooks}
\begin{tcolorbox}[skin=enhancedmiddle,breakable,
  check odd page,toggle left and right,
  boxrule=0mm,top=0mm,bottom=0mm,left=1mm,right=1mm,
  rightrule=1cm,colupper=blue!25!black,
  watermark graphics=lichtspiel.jpg,watermark opacity=0.5,
  watermark overzoom=1.00,watermark opacity=0.25,
  frame style={pattern=crosshatch dots light steel blue},
  overlay app={%
    \ifoddpage\coordinate (X) at ([xshift=-5mm]frame.east);
          \else\coordinate (X) at ([xshift=5mm]frame.west);\fi
    \fill[shading=ball,ball color=blue!50!white,opacity=0.5] (X) circle (4mm);}]
\lipsum[1-6]
\end{tcolorbox}
```

This example switches a `1cm` thick rule from the left to the right side depending on the page number. Thereby, the rule is always on the outer side of the double-sided paper. Additionally, a ball is drawn on the outer side with help of an overlay.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

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, consectetuer 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 consectetuer.

Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

3.5 Height Control

In a typical usage scenario, the height of a `tcolorbox` is computed automatically to fit the content. Nevertheless, the height can be set to a fixed value or to fit commonly for several boxes, e. g. if boxes are set side by side.

The height control keys are only applicable to unbreakable boxes. If a box is set to be `/tcb/breakable`[→P. 171], the height is always computed according to the *natural height.*

**/tcb/natural height** (no value, initially set)
Sets the total height of the colored box to its natural height depending on the box content.

/tcb/height=⟨*length*⟩ (no default)
Sets the total height of the colored box to ⟨*length*⟩ independent of the box content. ⟨*length*⟩ is the minimum height of the box, if `/tcb/height plus` is larger than zero.

```
\tcbset{width=(\linewidth-2mm)/3,before=,after=\hfill,
colframe=blue!75!black,colback=white}

\begin{tcolorbox}[height=1cm,valign=center]
  This box has a height of 1cm.
\end{tcolorbox}
\begin{tcolorbox}[height=2cm,valign=center]
  This box has a height of 2cm.
\end{tcolorbox}
\begin{tcolorbox}[height=3cm,split=0.5,valign=center,valign lower=center]
  This box has a height of 3cm.
  \tcblower
  Lower part.
\end{tcolorbox}
```

This box has a height of 1cm.

This box has a height of 2cm.

This box has a height of 3cm.

Lower part.
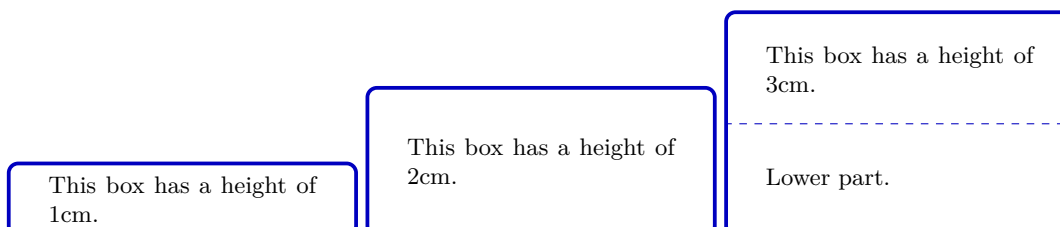
/tcb/height plus=⟨*length*⟩ (no default, initially `0pt`)
The box may extend a given fixed `/tcb/height` up to the given ⟨*length*⟩.

```
\tcbset{colback=red!5!white,colframe=red!75!black,left=1mm,top=1mm,bottom=1mm,
  right=1mm,boxsep=0mm,width=3cm,nobeforeafter}

\begin{tcolorbox}[height=1cm]
This is a tcolorbox.
\end{tcolorbox}
\begin{tcolorbox}[height=1cm,height plus=1cm]
This is a tcolorbox.
\end{tcolorbox}
\begin{tcolorbox}[height=1cm,height plus=1cm]
This is a tcolorbox. This is a tcolorbox. This is a tcolorbox.
\end{tcolorbox}
```

This is a tcolorbox.

This is a tcolorbox.

This is a tcolorbox.
This is a tcolorbox.
This is a tcolorbox.
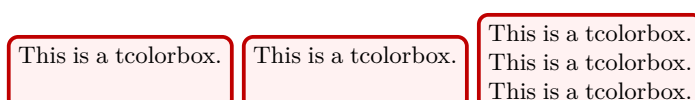
/tcb/height from=⟨*min*⟩ to ⟨*max*⟩   (style, no default)

Sets the box height to a dimension between ⟨*min*⟩ and ⟨*max*⟩.

```
% \usepackage{lipsum}
\newtcolorbox{mybox}{colback=red!5!white,colframe=red!75!black,left=1mm,top=1mm,
  bottom=1mm,right=1mm,boxsep=0mm,width=4.5cm,nobeforeafter,
  height from=2cm to 8cm}

\begin{mybox}
This is a tcolorbox.
\end{mybox}
\begin{mybox}
This is a tcolorbox. This is a tcolorbox. This is a tcolorbox.
\end{mybox}
\begin{mybox}
\lipsum[2]
\end{mybox}
```

This is a tcolorbox.

This is a tcolorbox. This is a tcolorbox. This is a tcolorbox.

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.

/tcb/space=⟨*fraction*⟩ (no default, initially 0)

If the height of a `tcolorbox` is not the natural height, the space difference between the forced and the natural size is distributed between the upper and the lower part of the box. This space could also be negative. ⟨*fraction*⟩ with a value between 0 and 1 is the amount of space which is added to the upper part, the rest is added to the lower part. If there is no lower part, then all of the space is added to the upper part always.

```
\tcbset{width=(\linewidth-2mm)/3,before=,after=\hfill,
colframe=blue!75!black,colback=white,height=3cm}

\foreach \f in {0.2,0.4,0.7}
{\begin{tcolorbox}[space=\f]
    This is the upper part.
    \tcblower
    This is the lower part.
\end{tcolorbox}}
```

This is the upper part.

This is the lower part.

This is the upper part.

This is the lower part.

This is the upper part.

This is the lower part.
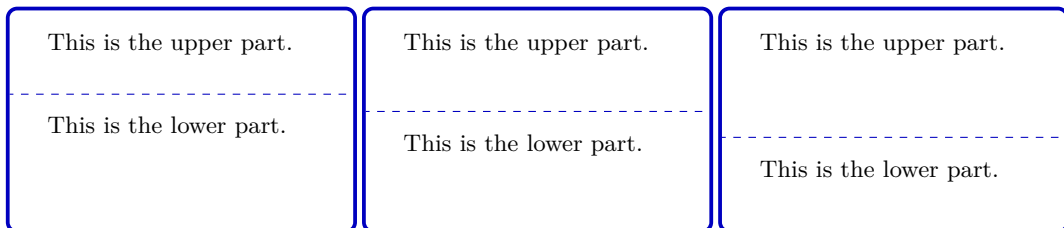
/tcb/space to upper (style)

This is an abbreviation for `space=1`, i.e. all extra space is added to the upper part.

**/tcb/space to lower** (style, initially set)

This is an abbreviation for `space=0`, i.e. all extra space is added to the lower part (if there is any).

/tcb/space to both (style)

This is an abbreviation for `space=0.5`, i.e. the extra space equally distributed between the upper and the lower part.

```
\tcbset{width=(\linewidth-2mm)/3,before=,after=\hfill,
colframe=blue!75!black,colback=white,height=3cm}

\foreach \myspace in {space to upper,space to both,space to lower}
{\begin{tcolorbox}[\myspace]
    This is the upper part.
    \tcblower
    This is the lower part.
\end{tcolorbox}}
```
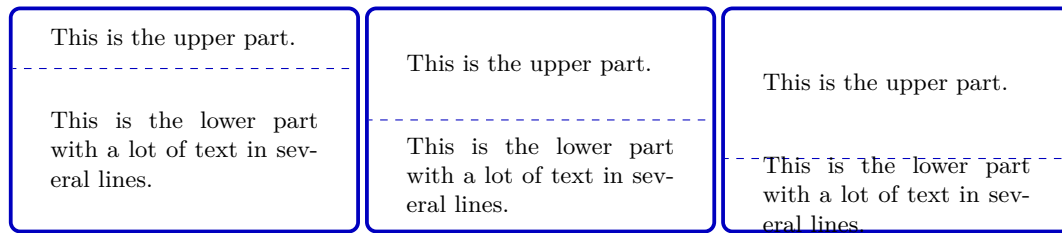
This is the upper part.

This is the lower part.

This is the upper part.

This is the lower part.

This is the upper part.

This is the lower part.
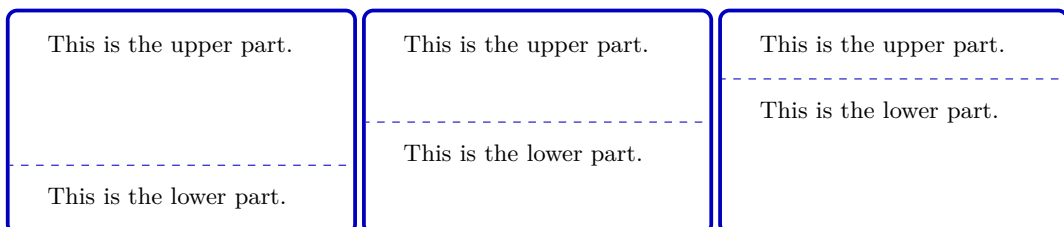
/tcb/split=⟨*fraction*⟩   (no default)

If the height of a `tcolorbox` is not the natural height, the ⟨*fraction*⟩ with a value between 0 and 1 determines the positioning of the segmentation between the upper and the lower part. Here, 0 stands for top and 1 for bottom. Note that the box is split regardless of the actual dimensions of the text parts!

```
\tcbset{width=(\linewidth-2mm)/3,before=,after=\hfill,height=3cm,
colback=white,colframe=blue!75!black,valign=center,valign lower=center}

\foreach \f in {0.1,0.5,0.8}
{\begin{tcolorbox}[split=\f]
This is the upper part.
\tcblower
This is the lower part with a lot of text in several lines.
\end{tcolorbox}}
```
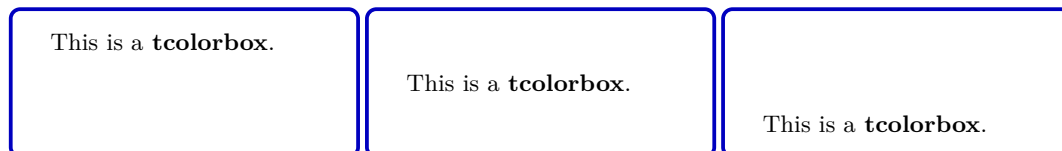
This is the upper part.

This is the lower part with a lot of text in several lines.

This is the upper part.

This is the lower part with a lot of text in several lines.

This is the upper part.

This is the lower part with a lot of text in several lines.

/tcb/valign=⟨*alignment*⟩   (no default, initially `top`)

If the height of a `tcolorbox` is not the natural height, `valign` determines the vertical ⟨*alignment*⟩ of the upper part. Feasible values are `top`, `center`, and `bottom`. For a box with natural height, these values are meaningless.

```
\tcbset{width=(\linewidth-2mm)/3,before=,after=\hfill,
colframe=blue!75!black,colback=white,height=2cm}

\foreach \myalign in {top,center,bottom}
{\begin{tcolorbox}[valign=\myalign]
  This is a \textbf{tcolorbox}.
\end{tcolorbox}}
```

This is a **tcolorbox**.

This is a **tcolorbox**.

This is a **tcolorbox**.

/tcb/valign lower=⟨*alignment*⟩   (no default, initially `top`)

This key has the same meaning for the lower part as `valign` for the upper part, i. e. it determines the vertical ⟨*alignment*⟩ of the lower part with feasible values `top`, `center`, and `bottom`.

/tcb/equal height group=⟨*id*⟩   (no default)

Boxes which are members of an `equal height group` will all get the same height, i.e. the maximum of all their natural heights. The ⟨*id*⟩ serves to distinguish between different height groups. This ⟨*id*⟩ should contain only characters which are feasible for TeX macro names, typically alphabetic characters but no numerals and spaces. Note that you have to compile twice to see changes and that height groups are global definitions.

```
\tcbset{width=(\linewidth-2mm)/3,before=,after=\hfill,arc=0mm,
colframe=blue!75!black,colback=white,fonttitle=\bfseries}

\begin{tcolorbox}[equal height group=A,adjusted title={One}]
  My smallest box.
\end{tcolorbox}%
\begin{tcolorbox}[equal height group=A,adjusted title={Two}]
  This box is also small.
  \tcblower
  But with a lower part.
\end{tcolorbox}%
\begin{tcolorbox}[equal height group=A,adjusted title={Three}]
  This box contains a lot of text just to fill the space
  with word flowing and flowing and flowing until the box
  is filled with all of it.
\end{tcolorbox}\linebreak
%
\tcbset{width=(\linewidth-1mm)/2,before=,after=\hfill,arc=0mm,
colframe=red!75!black,colback=white}
%
\begin{tcolorbox}[equal height group=B]
  Now, we use another equal height group.
\end{tcolorbox}%
\begin{tcolorbox}[equal height group=B]
  \begin{equation*}
    \int\limits_{0}^{1} x^2 = \frac13.
  \end{equation*}
\end{tcolorbox}
```

One	**Two**	**Three**
My smallest box.	This box is also small. ----- But with a lower part.	This box contains a lot of text just to fill the space with word flowing and flowing and flowing until the box is filled with all of it.

Now, we use another equal height group.	$$\int\limits_0^1 x^2 = \frac{1}{3}.$$

31

/tcb/minimum for equal height group=⟨*id*⟩:⟨*length*⟩   (no default, initially unset)
   Plants a ⟨*length*⟩ into the equal height group with the given ⟨*id*⟩. This ensures that the
   height will not drop below ⟨*length*⟩. Note that you cannot reduce a computed height value
   by using this key with a small value. The difference to applying `/tcb/height`[→P.27] directly
   is that the boxes are never too small for their content.

```
\tcbset{colframe=blue!75!black,colback=white,arc=0mm,
  before=,after=\hfill,fonttitle=\bfseries,left=2mm,right=2mm,
  width=3.5cm,
  equal height group=C,
  minimum for equal height group=C:3.5cm}

\begin{tcolorbox}
  My first box. All boxes will get 3.5cm times 3.5cm
  if the content height is not too large.
\end{tcolorbox}%
\begin{tcolorbox}
  My second box.
  \tcblower
  This is the lower part.
\end{tcolorbox}%
\begin{tcblisting}{}
\textbf{Mixed}
with a listing.
\end{tcblisting}
\begin{tcolorbox}[title={Fourth box}]
  My final box.
\end{tcolorbox}%
```

My first box. All boxes will get 3.5cm times 3.5cm if the content height is not too large.	My second box. ---------------- This is the lower part.	`\textbf{Mixed}` `with a listing.` ---------------- **Mixed** with a listing.	**Fourth box** My final box.

3.6 Box Content Additions

The following options introduce some arbitrary ⟨*code*⟩ to the content of a `tcolorbox`. These additions can be given at the begin or at the end of the title, the upper part, or the lower part.

/tcb/before title=⟨*code*⟩   (no default, initially unset)
The given ⟨*code*⟩ is placed *after* the color and font settings and *before* the content of the title.

```
\tcbset{before title={\textcolor{yellow}{\large Important:}~},
  colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[title=My title]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

> **Important: My title**
>
> This is a **tcolorbox**.

/tcb/after title=⟨*code*⟩   (no default, initially unset)
The given ⟨*code*⟩ is placed *after* the content of the title.

```
\tcbset{after title={\hfill\colorbox{Navy}{approved}},
  colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[title=My title]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

> **My title**   approved
>
> This is a **tcolorbox**.

/tcb/before upper=⟨*code*⟩   (no default, initially unset)
The given ⟨*code*⟩ is placed *after* the color and font settings and *before* the content of the upper part.

```
\tcbset{before upper={\textit{The story:}\par},
  colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[title=My title]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

> **My title**
>
> *The story:*
> This is a **tcolorbox**.

/tcb/after upper=⟨*code*⟩ (no default, initially unset)

The given ⟨*code*⟩ is placed *after* the content of the upper part.

```
\tcbset{after upper={\par\hfill\textit{Read more next week}},
  colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[title=My title]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

> **My title**
>
> This is a **tcolorbox**.
> *Read more next week*

/tcb/before lower=⟨*code*⟩ (no default, initially unset)

The given ⟨*code*⟩ is placed *after* the color and font settings and *before* the content of the lower part.

```
\tcbset{before lower=\textit{Behold:~},colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

> This is a **tcolorbox**.
> -
> *Behold:* This is the lower part.

/tcb/after lower=⟨*code*⟩ (no default, initially unset)

The given ⟨*code*⟩ is placed *after* the content of the lower part.

```
\tcbset{after lower=\ \textit{This is the end.},
  colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

> This is a **tcolorbox**.
> -
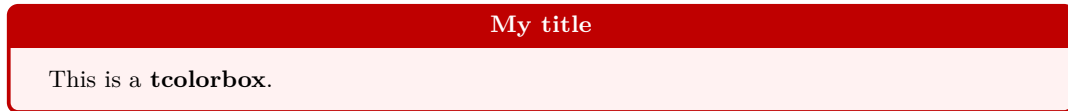> This is the lower part. *This is the end.*

/tcb/center title (style, no value, initially unset)
This style sets /tcb/before title→P. 33 to \centering.

```
\tcbset{center title,colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[title=My title]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```
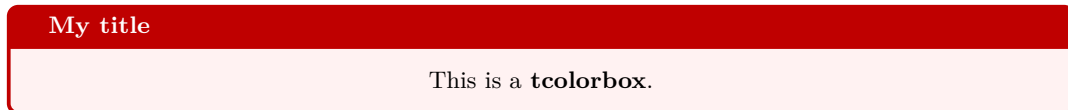
> **My title**
>
> This is a **tcolorbox**.

/tcb/center upper (style, no value, initially unset)
This style sets /tcb/before upper→P. 33 to \centering.

```
\tcbset{center upper,colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[title=My title]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```
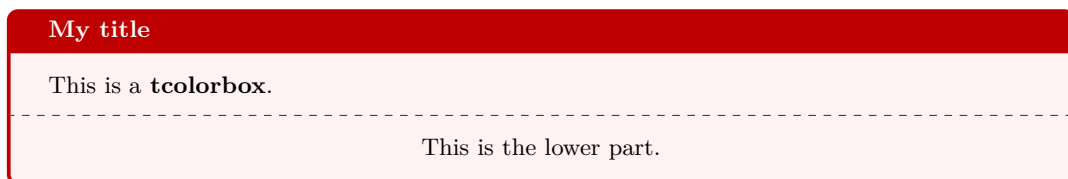
> **My title**
>
> This is a **tcolorbox**.

/tcb/center lower (style, no value, initially unset)
This style sets /tcb/before lower→P. 34 to \centering.

```
\tcbset{center lower,colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[title=My title]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

> **My title**
>
> This is a **tcolorbox**.
>
> -
>
> This is the lower part.

/tcb/flushleft title (style, no value, initially unset)

This style sets /tcb/before title$^{\rightarrow P.33}$ to \raggedright.

```
\tcbset{flushleft title,colback=red!5!white,
  colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[title=My title which is
  quite long but always flushed left]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

> **My title which is quite long but always flushed left**
>
> This is a **tcolorbox**.

/tcb/flushleft upper (style, no value, initially unset)

This style sets /tcb/before upper$^{\rightarrow P.33}$ to \raggedright.

```
\tcbset{flushleft upper,colback=red!5!white,
  colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[title=My title]
This is a \textbf{tcolorbox}. Here, the
content is flushed \textbf{left}.
You see the effect on the \textbf{right} hand \textbf{side}.
\end{tcolorbox}
```

> **My title**
>
> This is a **tcolorbox**. Here, the content is flushed left. You see the effect on the right hand side.

/tcb/flushleft lower (style, no value, initially unset)

This style sets /tcb/before lower$^{\rightarrow P.34}$ to \raggedright.

```
\tcbset{flushleft lower,colback=red!5!white,
  colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[title=My title]
This is a \textbf{tcolorbox}. The \textbf{upper} part
is not flushed \textbf{left} or \textbf{right}. This could be
unpleasant for small boxes.
\tcblower
This is the \textbf{lower} part. Here, the
content is flushed \textbf{left}.
You see the effect on the \textbf{right} hand \textbf{side}.
\end{tcolorbox}
```

> **My title**
>
> This is a **tcolorbox**. The upper part is not flushed left or right. This could be unpleasant for small boxes.
> - - - - - - - - - - - - - - - -
> This is the lower part. Here, the content is flushed left. You see the effect on the right hand side.

/tcb/flushright title (style, no value, initially unset)

This style sets /tcb/before title→P.33 to \raggedleft.

```
\tcbset{flushright title,colback=red!5!white,
  colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[title=My title]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

> **My title**
>
> This is a **tcolorbox**.

/tcb/flushright upper (style, no value, initially unset)

This style sets /tcb/before upper→P.33 to \raggedleft.

```
\tcbset{flushright upper,colback=red!5!white,
  colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[title=My title]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

> **My title**
>
> This is a **tcolorbox**.

/tcb/flushright lower (style, no value, initially unset)

This style sets /tcb/before lower→P.34 to \raggedleft.

```
\tcbset{flushright lower,colback=red!5!white,
  colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[title=My title]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

> **My title**
>
> This is a **tcolorbox**.
>
> This is the lower part.

/tcb/tabularx=⟨*preamble*⟩ (style)

This style sets **/tcb/before upper**[→P.33] and **/tcb/after upper**[→P.34] and several geometry keys to support a `tabularx` with the given ⟨*preamble*⟩. The packages `tabularx` [2], `array`, and `colortbl` have to be loaded separately.

```
% \usepackage{array,tabularx}
% \usepackage{colortbl} - or - \usepackage[table]{xcolor}
\newcolumntype{Y}{>{\raggedleft\arraybackslash}X}% see tabularx
\tcbset{enhanced,fonttitle=\bfseries\large,fontupper=\normalsize\sffamily,
  colback=yellow!10!white,colframe=red!50!black,colbacktitle=Salmon!30!white,
  coltitle=black,center title}

\begin{tcolorbox}[tabularx={X||Y|Y|Y|Y||Y},title=My table]
Group & One     & Two     & Three    & Four     & Sum\\\hline\hline
Red   & 1000.00 & 2000.00 &  3000.00 &  4000.00 & 10000.00\\\hline
Green & 2000.00 & 3000.00 &  4000.00 &  5000.00 & 14000.00\\\hline
Blue  & 3000.00 & 4000.00 &  5000.00 &  6000.00 & 18000.00\\\hline\hline
Sum   & 6000.00 & 9000.00 & 12000.00 & 15000.00 & 42000.00
\end{tcolorbox}
```

My table					
Group	One	Two	Three	Four	Sum
Red	1000.00	2000.00	3000.00	4000.00	10000.00
Green	2000.00	3000.00	4000.00	5000.00	14000.00
Blue	3000.00	4000.00	5000.00	6000.00	18000.00
Sum	6000.00	9000.00	12000.00	15000.00	42000.00

/tcb/tabularx*={⟨*code*⟩}{⟨*preamble*⟩} (style)

This is a variant of **/tcb/tabularx** which adds some ⟨*code*⟩ before the table starts.

```
% \usepackage{array,tabularx}
% \usepackage{colortbl} - or - \usepackage[table]{xcolor}
\tcbset{enhanced,fonttitle=\bfseries\large,fontupper=\normalsize\sffamily,
  colback=yellow!10!white,colframe=red!50!black,colbacktitle=Salmon!30!white,
  coltitle=black,center title}

\begin{tcolorbox}[tabularx*={\arrayrulewidth0.5mm}{X|X|X},title=My table]
One     & Two     & Three \\\hline\hline
1000.00 & 2000.00 &  3000.00\\\hline
2000.00 & 3000.00 &  4000.00
\end{tcolorbox}
```

My table		
One	Two	Three
1000.00	2000.00	3000.00
2000.00	3000.00	4000.00

This style adds a centered `tikzpicture` environment to the begin and to the end of the upper part.

```latex
% \usepackage{tikz}

\begin{tcolorbox}[tikz upper,fonttitle=\bfseries,colback=white,colframe=black,
                  title=TikZ drawing]
  \path[fill=yellow,draw=yellow!75!red] (0,0) circle (1cm);
  \fill[red] (45:5mm) circle (1mm);
  \fill[red] (135:5mm) circle (1mm);
  \draw[line width=1mm,red] (215:5mm) arc (215:325:5mm);
\end{tcolorbox}
```

TikZ drawing

/tcb/tikz lower=⟨*options*⟩ (style)

This style adds a centered `tikzpicture` environment to the begin and to the end of the lower part.

```latex
% \usepackage{tikz}
% \usetikzlibrary{skins,listings}
\tcbset{tikz lower,listing side text,fonttitle=\bfseries,
  bicolor,colback=LightBlue!50!white,colbacklower=white,colframe=black,
  righthand width=3cm}

\begin{tcblisting}{title=TikZ drawing}
\path[fill=yellow,draw=yellow!75!red]
    (0,0) circle (1cm);
\fill[red] (45:5mm) circle (1mm);
\fill[red] (135:5mm) circle (1mm);
\draw[line width=1mm,red]
    (215:5mm) arc (215:325:5mm);
\end{tcblisting}
```

TikZ drawing

```latex
\path[fill=yellow,draw=yellow!75!red]
    (0,0) circle (1cm);
\fill[red] (45:5mm) circle (1mm);
\fill[red] (135:5mm) circle (1mm);
\draw[line width=1mm,red]
    (215:5mm) arc (215:325:5mm);
```

/tcb/code=⟨*code*⟩   (no default, initially unset)

The given ⟨*code*⟩ is executed immediately. This option is useful to place some arbitrary code into an option list.

```
\tcbset{colback=red!5!white,colframe=red!75!black,
  code={Useless at this spot but functional.},
  fonttitle=\bfseries}

\begin{tcolorbox}[code={\newcommand{\mycommand}{\textit{working}}},
  title=My \mycommand\ title]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

Useless at this spot but functional.

My *working* title

This is a **tcolorbox**.

3.7 Overlays

With an overlay, arbitrary ⟨*graphical code*⟩ can be added to a `tcolorbox`. This code is executed *after* the frame and interior are drawn and *before* the text content is drawn. Therefore, you can decorate the `tcolorbox` with your own extensions. Common special cases are *watermarks* which are implemented using overlays. See Subsection 6.4 from page 80 if you want to add *watermarks*.

> If you use the core package only, the ⟨*graphical code*⟩ has to be `pgf` code and there is not much assistance for positioning. Therefore, the usage of the `/tcb/enhanced`→P.99 mode from the library skins is recommended which allows `tikz` code and gives access to `/tcb/geometry nodes`→P.73 for positioning.

`/tcb/overlay`=⟨*graphical code*⟩   (no default, initially unset)

Adds ⟨*graphical code*⟩ to the box drawing process. This ⟨*graphical code*⟩ is drawn *after* the frame and interior and *before* the text content.

```
% \tcbuselibrary{skins} % preamble
\tcbset{frogbox/.style={enhanced,colback=green!10,colframe=green!65!black,
  enlarge top by=5.5mm,
  overlay={\foreach \x in {2cm,3.5cm} {
    \begin{scope}[shift={([xshift=\x]frame.north west)}]
      \path[draw=green!65!black,fill=green!10,line width=1mm] (0,0) arc (0:180:5mm);
      \path[fill=black] (-0.2,0) arc (0:180:1mm);
    \end{scope}}}]}}

\begin{tcolorbox}[frogbox,title=My title]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

My title

This is a **tcolorbox**.

```
% \usetikzlibrary{patterns} % preamble
% \tcbuselibrary{skins}     % preamble
\tcbset{ribbonbox/.style={enhanced,colback=red!5!white,colframe=red!75!black,
  fonttitle=\bfseries,
  overlay={\path[fill=blue!75!white,draw=blue,double=white!85!blue,
    preaction={opacity=0.6,fill=blue!75!white},
    line width=0.1mm,double distance=0.2mm,
    pattern=fivepointed stars,pattern color=white!75!blue]
    ([xshift=-0.2mm,yshift=-1.02cm]frame.north east)
    -- ++(-1,1) -- ++(-0.5,0) -- ++(1.5,-1.5) -- cycle;}}}

\begin{tcolorbox}[ribbonbox,title=My title]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

My title

This is a **tcolorbox**.

- -

This is the lower part.

`/tcb/no overlay` (style, no default, initially set)
Removes the overlay if set before.

`/tcb/overlay broken`=⟨*graphical code*⟩ (no default, initially unset)
If the box is set to be `/tcb/breakable`[→P. 171] and *is* broken actually, then the ⟨*graphical code*⟩ is added to the box drawing process. `/tcb/overlay`[→P. 41] overwrites this key.

`/tcb/overlay unbroken`=⟨*graphical code*⟩ (no default, initially unset)
If the box is set to be `/tcb/breakable`[→P. 171] but *is not* broken actually or if the box is set to be `/tcb/unbreakable`[→P. 172], then the ⟨*graphical code*⟩ is added to the box drawing process. `/tcb/overlay`[→P. 41] overwrites this key.

`/tcb/overlay first`=⟨*graphical code*⟩ (no default, initially unset)
If the box is set to be `/tcb/breakable`[→P. 171] and *is* broken actually, then the ⟨*graphical code*⟩ is added to the box drawing process for the *first* part of the break sequence. `/tcb/overlay`[→P. 41] overwrites this key.

`/tcb/overlay middle`=⟨*graphical code*⟩ (no default, initially unset)
If the box is set to be `/tcb/breakable`[→P. 171] and *is* broken actually, then the ⟨*graphical code*⟩ is added to the box drawing process for the *middle* parts (if any) of the break sequence. `/tcb/overlay`[→P. 41] overwrites this key.

`/tcb/overlay last`=⟨*graphical code*⟩ (no default, initially unset)
If the box is set to be `/tcb/breakable`[→P. 171] and *is* broken actually, then the ⟨*graphical code*⟩ is added to the box drawing process for the *last* part of the break sequence. `/tcb/overlay`[→P. 41] overwrites this key.

`/tcb/overlay unbroken and first`=⟨*graphical code*⟩ (no default, initially unset)
This is an optimized abbreviation for setting `/tcb/overlay unbroken` and `/tcb/overlay first` together. `/tcb/overlay`[→P. 41] overwrites this key.

`/tcb/overlay middle and last`=⟨*graphical code*⟩ (no default, initially unset)
This is an optimized abbreviation for setting `/tcb/overlay middle` and `/tcb/overlay last` together. `/tcb/overlay`[→P. 41] overwrites this key.

This example demonstrates the application of break sequence specific overlay options. Here, we define an environment `myexample` based on `tcolorbox` where the visible drawing is done totally by overlay keys.
Here, the first application of `myexample` produces an unbroken `tcolorbox`. The frame is drawn by the code given with `/tcb/overlay unbroken`.
The second application of `myexample` is broken into several parts which are drawn by the codes given with `/tcb/overlay first`, `/tcb/overlay middle`, and `/tcb/overlay last`.

```
% Preamble:
%\usepackage{tikz,lipsum,calc}
%\tcbuselibrary{skins,breakable}
%\newcounter{example}
%\newlength{\examlen}
\colorlet{colexam}{red!75!black}

\newtcolorbox[use counter=example]{myexample}{%
  phantom={\settowidth{\global\examlen}{\Large\bfseries Example \thetcbcounter}},%
  title={Example \thetcbcounter},
  coltitle=colexam,fonttitle=\Large\bfseries,
  enhanced,breakable,before=\par\medskip,parbox=false,
  frame hidden,interior hidden,segmentation hidden,
```

```
      boxsep=0pt,left=0pt,right=3mm,toptitle=2mm,pad at break=0mm,
      overlay unbroken={\draw[colexam,line width=1pt] (frame.north west)
        --([xshift=-0.5pt]frame.north east)--([xshift=-0.5pt]frame.south east)
        --(frame.south west);
        \draw[colexam,line width=2pt] ([yshift=0.5pt]frame.north west)
        -- +(\examlen,0pt);},
      overlay first={\draw[colexam,line width=1pt] (frame.north west)
        --([xshift=-0.5pt]frame.north east)--([xshift=-0.5pt]frame.south east);
        \draw[red!75!black,line width=2pt] ([yshift=0.5pt]frame.north west)
        -- +(\examlen,0pt);},
      overlay middle={\draw[colexam,line width=1pt] ([xshift=-0.5pt]frame.north east)
        --([xshift=-0.5pt]frame.south east); },
      overlay last={\draw[colexam,line width=1pt] ([xshift=-0.5pt]frame.north east)
        --([xshift=-0.5pt]frame.south east)--(frame.south west);}%
   }

   \begin{myexample}
   \lipsum[1]
   \end{myexample}

   \begin{myexample}
   \lipsum[2-11]
   \end{myexample}

   \lipsum[12]% following text
```

Example 1

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

Example 2

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.

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, consectetuer 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 consectetuer.

Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

Sed commodo posuere pede. Mauris ut est. Ut quis purus. Sed ac odio. Sed vehicula hendrerit sem. Duis non odio. Morbi ut dui. Sed accumsan risus eget odio. In hac habitasse platea dictumst. Pellentesque non elit. Fusce sed justo eu urna porta tincidunt. Mauris felis odio, sollicitudin sed, volutpat a, ornare ac, erat. Morbi quis dolor. Donec pellentesque, erat ac sagittis semper, nunc dui lobortis purus, quis congue purus metus ultricies tellus. Proin et quam. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent sapien turpis, fermentum vel, eleifend faucibus, vehicula eu, lacus.

Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Donec odio elit, dictum in, hendrerit sit amet, egestas sed, leo. Praesent feugiat sapien aliquet odio. Integer vitae justo. Aliquam vestibulum fringilla lorem. Sed neque lectus, consectetuer at, consectetuer sed, eleifend ac, lectus. Nulla facilisi. Pellentesque eget lectus. Proin eu metus. Sed porttitor. In hac habitasse platea dictumst. Suspendisse eu lectus. Ut mi mi, lacinia sit amet, placerat et, mollis vitae, dui. Sed ante tellus, tristique ut, iaculis eu, malesuada ac, dui. Mauris nibh leo, facilisis non, adipiscing quis, ultrices a, dui.

Morbi luctus, wisi viverra faucibus pretium, nibh est placerat odio, nec commodo wisi enim eget quam. Quisque libero justo, consectetuer a, feugiat vitae, porttitor eu, libero. Suspendisse sed mauris vitae elit sollicitudin malesuada. Maecenas ultricies eros sit amet ante. Ut venenatis velit. Maecenas sed mi eget dui varius euismod. Phasellus aliquet volutpat odio. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Pellentesque sit amet pede ac sem eleifend consectetuer. Nullam elementum, urna vel imperdiet sodales, elit ipsum pharetra ligula, ac pretium ante justo a nulla. Curabitur tristique arcu eu metus. Vestibulum lectus. Proin mauris. Proin eu nunc eu urna hendrerit faucibus. Aliquam auctor, pede consequat laoreet varius, eros tellus scelerisque quam, pellentesque hendrerit ipsum dolor sed augue. Nulla nec lacus.

Suspendisse vitae elit. Aliquam arcu neque, ornare in, ullamcorper quis, commodo eu, libero. Fusce sagittis erat at erat tristique mollis. Maecenas sapien libero, molestie et, lobortis in, sodales eget, dui. Morbi ultrices rutrum lorem. Nam elementum ullamcorper leo. Morbi dui. Aliquam sagittis. Nunc placerat. Pellentesque tristique sodales est. Maecenas imperdiet lacinia velit. Cras non urna. Morbi eros pede, suscipit ac, varius vel, egestas non, eros.

Praesent malesuada, diam id pretium elementum, eros sem dictum tortor, vel consectetuer odio sem sed wisi.

Sed feugiat. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Ut pellentesque augue sed urna. Vestibulum diam eros, fringilla et, consectetuer eu, nonummy id, sapien. Nullam at lectus. In sagittis ultrices mauris. Curabitur malesuada erat sit amet massa. Fusce blandit. Aliquam erat volutpat. Aliquam euismod. Aenean vel lectus. Nunc imperdiet justo nec dolor.

Etiam euismod. Fusce facilisis lacinia dui. Suspendisse potenti. In mi erat, cursus id, nonummy sed, ullamcorper eget, sapien. Praesent pretium, magna in eleifend egestas, pede pede pretium lorem, quis consectetuer tortor sapien facilisis magna. Mauris quis magna varius nulla scelerisque imperdiet. Aliquam non quam. Aliquam porttitor quam a lacus. Praesent vel arcu ut tortor cursus volutpat. In vitae pede quis diam bibendum placerat. Fusce elementum convallis neque. Sed dolor orci, scelerisque ac, dapibus nec, ultricies ut, mi. Duis nec dui quis leo sagittis commodo.

```
% \tcbuselibrary{skins}
% \newcounter{example}
\newtcolorbox[use counter=example]{FancyTitle}[3][]{%
  enhanced,colback=blue!10!white,colframe=orange,top=4mm,
  enlarge top by=\baselineskip/2+1mm,
  enlarge top at break by=0mm,pad at break=2mm,
  fontupper=\normalsize,label={#3},
  overlay unbroken and first={%
    \node[rectangle,rounded corners,draw=black,fill=blue!20!white,
      inner sep=1mm,anchor=west,font=\small]
      at ([xshift=4.5mm]frame.north west)
        {\strut\textbf{Example \thetcbcounter: #2}};},
  #1}%

\begin{FancyTitle}{My fancy title}{fancy:title}
  \lipsum[1]
\end{FancyTitle}
```

Example 3: My fancy title

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

3.8   Floating Objects

/tcb/floatplacement=⟨*values*⟩   (no default, initially `htb`)

Sets ⟨*values*⟩ as default values for the usage of `float`. Feasible are the usual parameters for floating objects.

```
\tcbset{enhanced,colback=red!5!white,colframe=red!75!black,
  watermark color=red!15!white}

\begin{tcolorbox}[floatplacement=t,float,
                 title=Floating box from |floatplacement|,
                 watermark text={I am floating}]
  This floating box is placed at the top of a page.
\end{tcolorbox}
```

/tcb/float=⟨*values*⟩   (default from `floatplacement`)

Turns the box to a floating object where ⟨*values*⟩ are the usual parameters for such floating objects. If they are not used, the placement uses the default values given by `floatplacement`.

```
\begin{tcolorbox}[float, title=Floating box from |float|,
    enhanced,watermark text={I'm also floating}]
  This box floats to a feasible place automatically. You do not have to
  use a numbering for this floating object.
\end{tcolorbox}
```

Floating box from `float`

This box floats to a feasible place automatically. You do not have to use a numbering for this floating object.

/tcb/nofloat   (style, initially set)

Turns the floating behavior off.

3.9 Side by Side

/tcb/sidebyside=⟨*boolean value*⟩   (default `true`, initially `false`)

Normally, the upper part and the lower part of the box have their positions as their names suggest. If `sidebyside` is set to `true`, the upper part is drawn *left-handed* and the lower part is drawn *right-handed*. Both parts are drawn together with the geometry settings of the upper part but the space is divided horizontally according to the following options. Colors, fonts, and box content additions are used individually. The resulting box is unbreakable.

```
\tcbset{colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[title=My title,sidebyside]
This is the upper (\textit{left}-handed) part.
\tcblower
This is the lower (\textit{right}-handed) part.
\end{tcolorbox}
```

> **My title**
>
> This is the upper (*left-handed*) part.   This is the lower (*right-handed*) part.

/tcb/sidebyside align=⟨*alignment value*⟩   (no default, initially `center`)

Sets the vertical alignment for the left-handed and right-handed part. Feasible values are `center`, `top`, and `bottom`.

```
\tcbset{colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries,nobeforeafter,
  left=2mm,right=2mm,sidebyside,sidebyside gap=6mm,width=(\linewidth-2mm)/3}

\begin{tcolorbox}[adjusted title=center,sidebyside align=center]
This is a text which is too long for one line.
\tcblower
This is a short text.
\end{tcolorbox}\hfill
\begin{tcolorbox}[adjusted title=top,sidebyside align=top]
This is a text which is too long for one line.
\tcblower
This is a short text.
\end{tcolorbox}\hfill
\begin{tcolorbox}[adjusted title=bottom,sidebyside align=bottom]
This is a text which is too long for one line.
\tcblower
This is a short text.
\end{tcolorbox}
```
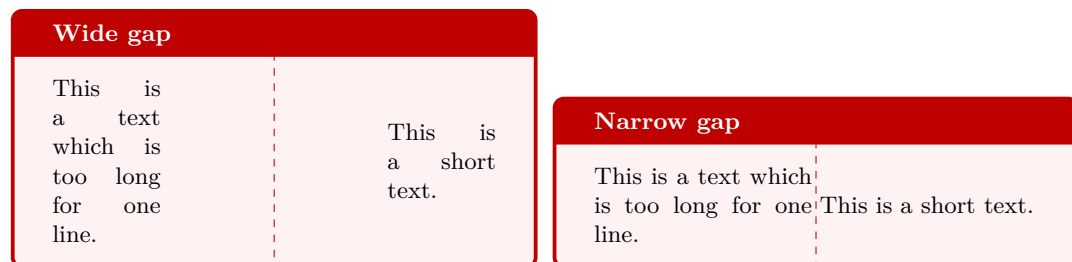
> **center**
>
> This is a text which is too long for one line.   This is a short text.

> **top**
>
> This is a text which is too long for one line.   This is a short text.

> **bottom**
>
> This is a text which is too long for one line.   This is a short text.

/tcb/sidebyside gap=⟨*length*⟩   (no default, initially 10mm)
    Sets the horizontal distance between the left-handed and right-handed part to ⟨*length*⟩.

```
\tcbset{colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries,nobeforeafter,
  sidebyside,width=(\linewidth-2mm)/2}

\begin{tcolorbox}[adjusted title=Wide gap,sidebyside gap=30mm]
This is a text which is too long for one line.
\tcblower
This is a short text.
\end{tcolorbox}\hfill
\begin{tcolorbox}[adjusted title=Narrow gap,sidebyside gap=1mm]
This is a text which is too long for one line.
\tcblower
This is a short text.
\end{tcolorbox}
```

Wide gap

This is	
------------	
a text	This is
which is	a short
too long	text.
for one	
line.	

Narrow gap

This is a text which
is too long for one · This is a short text.
line.

/tcb/lefthand width=⟨*length*⟩   (no default, initially unset)
    Sets the width of the left-handed part to the given ⟨*length*⟩.

```
\tcbset{colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[title=My title,sidebyside,lefthand width=3cm]
This is the upper (\textit{left}-handed) part.
\tcblower
This is the lower (\textit{right}-handed) part.
\end{tcolorbox}
```
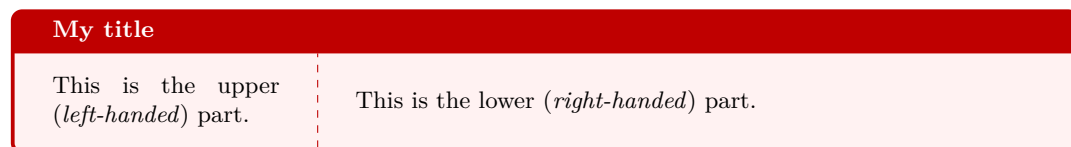
My title

This is the upper
(*left-handed*) part.   This is the lower (*right-handed*) part.
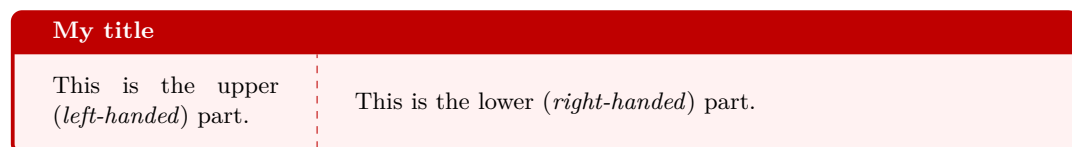
/tcb/righthand width=⟨*length*⟩   (no default, initially unset)
    Sets the width of the right-handed part to the given ⟨*length*⟩.

```
\tcbset{colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[title=My title,sidebyside,righthand width=3cm]
This is the upper (\textit{left}-handed) part.
\tcblower
This is the lower (\textit{right}-handed) part.
\end{tcolorbox}
```

My title

This is the upper (*left-handed*) part.   This is the lower
                                                   (*right-handed*) part.

/tcb/lefthand ratio=⟨*fraction*⟩ (no default, initially 0.5)

Sets the width of the left-handed part to the given ⟨*fraction*⟩ of the available space. ⟨*fraction*⟩ is a value between 0 and 1.

```
\tcbset{colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[title=My title,sidebyside,lefthand ratio=0.25]
This is the upper (\textit{left}-handed) part.
\tcblower
This is the lower (\textit{right}-handed) part.
\end{tcolorbox}
```

> **My title**
>
> This is the upper (*left-handed*) part. | This is the lower (*right-handed*) part.

/tcb/righthand ratio=⟨*fraction*⟩ (no default, initially 0.5)

Sets the width of the right-handed part to the given ⟨*fraction*⟩ of the available space. ⟨*fraction*⟩ is a value between 0 and 1.

```
\tcbset{colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[title=My title,sidebyside,righthand ratio=0.25]
This is the upper (\textit{left}-handed) part.
\tcblower
This is the lower (\textit{right}-handed) part.
\end{tcolorbox}
```

> **My title**
>
> This is the upper (*left-handed*) part. | This is the lower (*right-handed*) part.

3.10 Embedding into the Surroundings

Typically, but not necessarily, a `tcolorbox` is put inside a separate paragraph and has some vertical space before and after it. This behavior is controlled by the keys `before` and `after`.

/tcb/before=⟨*code*⟩   (no default, initially see /tcb/autoparskip)
    Sets the ⟨*code*⟩ which is executed before the colored box. It is not used for floating boxes.

/tcb/after=⟨*code*⟩   (no default, initially see /tcb/autoparskip)
    Sets the ⟨*code*⟩ which is executed after the colored box. It is not used for floating boxes.

/tcb/parskip   (style, no value)
    Sets the keys `before` and `after` to values which are recommended, if the package `parskip` *is* used and there is no better idea for `before` and `after`.

```
\tcbset{parskip/.style={before={\par\pagebreak[0]\noindent},
                        after={\par}}}
```

/tcb/noparskip   (style, no value)
    Sets the keys `before` and `after` to values which are recommended, if the package `parskip` is *not* used and there is no better idea for `before` and `after`.

```
\tcbset{noparskip/.style={before={\par\smallskip\pagebreak[0]\noindent},
                          after={\par\smallskip}}}
```

/tcb/autoparskip   (style, no value, initially set)
    Tries to detect the usage of the package `parskip` and sets the keys `before` and `after` accordingly. Actually, the following is done:
- If the length of `\parskip` is greater than `0pt` at the begin of the document, /tcb/parskip is executed. Here, the usage of package `parskip` is *assumed*.
- Otherwise, if the length of `\parskip` is not greater than `0pt` at the begin of the document, /tcb/noparskip is executed. Here, the absence of package `parskip` is *assumed*.

    `autoparskip` is the default for the package `tcolorbox`, if `before` or `after` are not changed otherwise.

/tcb/nobeforeafter   (style, no value)
    Abbreviation for clearing the keys `before` and `after`. The colored box is not put into a paragraph and there is no space before or after the box.

```
\tcbset{myone/.style={colback=LightGreen,colframe=DarkGreen,
  equal height group=nobefaf,width=\linewidth/4,nobeforeafter}}
\begin{tcolorbox}[myone,title=Box 1]Box 1\end{tcolorbox}%
\begin{tcolorbox}[myone,title=Box 2]Box 2\end{tcolorbox}%
\begin{tcolorbox}[myone,title=Box 3]Box 3\end{tcolorbox}%
\begin{tcolorbox}[myone,title=Box 4]Box 4\end{tcolorbox}
```

Box 1	Box 2	Box 3	Box 4
Box 1	Box 2	Box 3	Box 4

/tcb/baseline=⟨*length*⟩   (no default, initially `0pt`)
    Used to set the `\pgfsetbaseline` value of the resulting `tcolorbox`.

3.11 Bounding Box

Normally, every `tcolorbox` has a bounding box which fits exactly to the dimensions of the outer frame. Therefore, LaTeX reserves exactly the space needed for the box. This behavior can be changed by enlarging (or shrinking) the bounding box. If the bounding box is enlarged, the `tcolorbox` will get some clearance around it. If the bounding box is shrunk, i.e. enlarged with negative values, the `tcolorbox` will overlap to other parts of the page. For example, the `tcolorbox` could be stretched into the page margin.

/tcb/enlarge top by=⟨*length*⟩   (no default, initially `0mm`)
> Enlarges the bounding box distance to the top of the box by ⟨*length*⟩.

```
\tcbset{colframe=blue!75!black,colback=white}

\begin{tcolorbox}[enlarge top by=-5mm]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
\begin{tcolorbox}[enlarge top by=5mm]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

> This is a **tcolorbox**.

> This is a **tcolorbox**.

/tcb/enlarge bottom by=⟨*length*⟩   (no default, initially `0mm`)
> Enlarges the bounding box distance to the bottom of the box by ⟨*length*⟩.

```
\tcbset{colframe=blue!75!black,colback=white}

\begin{tcolorbox}[enlarge bottom by=5mm]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
\begin{tcolorbox}[enlarge bottom by=-5mm]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

> This is a **tcolorbox**.

> This is a **tcolorbox**.

/tcb/enlarge left by=⟨*length*⟩   (no default, initially `0mm`)

Enlarges the bounding box distance to the left side of the box by ⟨*length*⟩.

```
\tcbset{colframe=blue!75!black,colback=white}

\begin{tcolorbox}[enlarge left by=2cm,width=\linewidth-2cm]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
\begin{tcolorbox}[enlarge left by=-2cm,width=\linewidth+2cm]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

> This is a **tcolorbox**.

This is a **tcolorbox**.

/tcb/enlarge right by=⟨*length*⟩   (no default, initially `0mm`)

Enlarges the bounding box distance to the right side of the box by ⟨*length*⟩.

```
\tcbset{colframe=blue!75!black,colback=white}

\begin{tcolorbox}[enlarge right by=-2cm,width=\linewidth+2cm]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
\begin{tcolorbox}[enlarge right by=2cm,width=\linewidth-2cm]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

This is a **tcolorbox**.

This is a **tcolorbox**.

> **Floating box from `toggle enlargement`**
>
> This page is an odd page. Therefore, the left and right enlargements are not toggled (with some luck; otherwise use `forced`). This box stretches to the right margin on odd pages and to the left margin on even pages. The current document is one-sided – this feature makes sense for two-sided documents only.

`/tcb/toggle enlargement`=⟨*toggle preset*⟩ (default `evenpage`, initially `none`)

According to the ⟨*toggle preset*⟩, the left and the right enlargements of the bounding box are switched or not. Feasible values are:

- `none`: no switching.
- `forced`: the values of the left and right enlargement are switched.
- `evenpage`: if the page is an even page, the values of the left and right enlargement are switched. It is recommended to use this setting in conjunction with `/tcb/check odd page`[→P. 63].

```
% \usepackage{changepage} for 'check odd page'
\tcbset{colframe=blue!75!black,colback=white,
enlarge left by=-20mm,enlarge right by=5mm,width=\linewidth+15mm}

\begin{tcolorbox}[toggle enlargement=none]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
\begin{tcolorbox}[toggle enlargement=forced]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
\begin{tcolorbox}[toggle enlargement=evenpage,check odd page]
  This page is an \ifthenelse{\isodd{\thepage}}{odd}{even} page.
  Therefore, the left and right enlargements
  \ifthenelse{\isodd{\thepage}}{are not}{are} toggled (with some luck).
\end{tcolorbox}
```

> This is a **tcolorbox**.

> This is a **tcolorbox**.

> This page is an odd page. Therefore, the left and right enlargements are not toggled (with some luck).

```
\begin{tcolorbox}[colframe=red!60!black,colback=red!15!white,
  fonttitle=\bfseries,title=Floating box from \texttt{toggle enlargement},
  width=\textwidth+20mm,enlarge right by=-20mm,
  toggle enlargement=evenpage,float=t]
  This page is an \ifthenelse{\isodd{\thepage}}{odd}{even} page.
  Therefore, the left and right enlargements
  \ifthenelse{\isodd{\thepage}}{are not}{are} toggled (with some luck; otherwise
  use |forced|). This box stretches to the right margin on odd pages and to the left
  margin on even pages. The current document is one-sided -- this feature makes
  sense for two-sided documents only.
\end{tcolorbox}
```

> The following keys should not be used with breakable boxes or boxes with a lower part.

/tcb/shrink tight   (style, no value, initially unset)

The total colored box is shrunk to the dimensions of the upper part. There should be no lower part and no title. This style sets the `/tcb/boxsep`[→P.19] to `0pt` and other geometry keys to fitting values. This option is likely to be used with the following extrusion keys.

```
\tcbset{colframe=blue!75!black,colback=white,arc=0mm,boxrule=0.4pt,
       nobeforeafter,tcbox raise base,shrink tight}

\begin{tcolorbox}
This is a \textbf{tcolorbox}.
\end{tcolorbox}

Lorem \tcbox{ipsum} dolor sit amet, consectetuer adipiscing elit.
```

This is a **tcolorbox**.
Lorem ipsum dolor sit amet, consectetuer adipiscing elit.

/tcb/extrude left by=⟨*length*⟩   (style, no default, initially unset)

The (upper part of the) colored box is extruded by the given ⟨*length*⟩ to the left side. The inner width and the bounding box is kept unchanged and the operation is additiv!

```
\tcbset{enhanced,colframe=red,colback=yellow!25!white,
  frame style={opacity=0.25},interior style={opacity=0.5},
  nobeforeafter,tcbox raise base,shrink tight,extrude by=2mm}

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit,
vestibulum ut, placerat ac, adipiscing vitae, felis.
\tcbox[extrude left by=1cm]{Curabitur} dictum gravida mauris.
Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna.
```

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

/tcb/extrude right by=⟨*length*⟩   (style, no default, initially unset)

The (upper part of the) colored box is extruded by the given ⟨*length*⟩ to the right side. The inner width and the bounding box is kept unchanged and the operation is additiv!

```
\tcbset{enhanced,colframe=red,colback=yellow!25!white,
  frame style={opacity=0.25},interior style={opacity=0.5},
  nobeforeafter,tcbox raise base,shrink tight,extrude by=2mm}

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit,
vestibulum ut, placerat ac, adipiscing vitae, felis.
\tcbox[extrude right by=1cm]{Curabitur} dictum gravida mauris.
Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna.
```

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

/tcb/extrude top by=⟨*length*⟩   (style, no default, initially unset)

The (upper part of the) colored box is extruded by the given ⟨*length*⟩ to the top side. The inner width and the bounding box is kept unchanged and the operation is additiv!

```
\tcbset{enhanced,colframe=red,colback=yellow!25!white,
  frame style={opacity=0.25},interior style={opacity=0.5},
  nobeforeafter,tcbox raise base,shrink tight,extrude by=2mm}

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit,
vestibulum ut, placerat ac, adipiscing vitae, felis.
\tcbox[extrude top by=1cm]{Curabitur} dictum gravida mauris.
Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna.
```

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

/tcb/extrude bottom by=⟨*length*⟩   (style, no default, initially unset)

The (upper part of the) colored box is extruded by the given ⟨*length*⟩ to the bottom side. The inner width and the bounding box is kept unchanged and the operation is additiv!

```
\tcbset{enhanced,colframe=red,colback=yellow!25!white,
  frame style={opacity=0.25},interior style={opacity=0.5},
  nobeforeafter,tcbox raise base,shrink tight,extrude by=2mm}

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit,
vestibulum ut, placerat ac, adipiscing vitae, felis.
\tcbox[extrude bottom by=1cm]{Curabitur} dictum gravida mauris.
Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna.
```

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

/tcb/extrude by=⟨*length*⟩   (style, no default, initially unset)

The (upper part of the) colored box is extruded by the given ⟨*length*⟩ to all sides. The inner width and the bounding box is kept unchanged and the operation is additiv!

```
\tcbset{enhanced,colframe=red,colback=yellow!25!white,
  frame style={opacity=0.25},interior style={opacity=0.5},
  nobeforeafter,tcbox raise base,shrink tight,extrude by=2mm}

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit,
vestibulum ut, placerat ac, adipiscing vitae, felis. \tcbox{Curabitur} dictum
gravida mauris. \tcbox[colframe=Green,interior style={opacity=0.0}]{Nam}
arcu libero, nonummy eget, consectetuer id, \tcbox{vulputate} a, magna. Donec
vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus
et malesuada fames ac turpis egestas. \tcbox{Mauris ut leo.}
```

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

3.12 Layered Boxes and Every Box Settings

A `tcolorbox` may contain another `tcolorbox` and so on. The package takes track of the nesting level using a counter `tcblayer`. Counter values may be used for doing some fancy things, but you should never change the counter value yourself.

The package takes special care for the first four layers or nesting levels, called managed layers. Here, footnote texts are administrated to find their intended place and specific layer dependent options may be set by changing `/tcb/every box on layer n`[→P.57]. If needed, the number of managed layers can be increased by setting `\tcbsetmanagedlayer`[→P.57] to a higher value than 4.

The following styles have a considerable influence on how layered boxes are processed. Note especially that nested boxes are getting a `/tcb/reset`[→P.64] by default. You can change this, but be prepared for suprises if you do.

If the defaults are *not changed*, a `tcolorbox` gets its options in the following order. Following options overwrite preceding options.

1. On package load, all options are set to default values.

2. Every `\tcbset`[→P.8] command adds or changes options for the following boxes inside the current TeX group.

3. While entering a `tcolorbox`, a `/tcb/every box on layer n`[→P.57] or `/tcb/every box on higher layers`[→P.57] option list is applied. With default settings this means:

   - For layer 1 (lowest layer), the `/tcb/every box` option list is applied. Not overwritten options given by a preceding `\tcbset`[→P.8] survive.

   - For layer 2 and above (nested boxes), a `/tcb/reset`[→P.64] followed by `/tcb/every box` option list is applied. Every resettable options given by a preceding `\tcbset`[→P.8] and by the sourrounding box(es) are reset.

4. The ⟨*options*⟩ given to the `tcolorbox` are applied. Or, if the box was generated by `\newtcolorbox`[→P.9] or friends, the ⟨*options*⟩ given there are applied.

5. If the box was generated by `\newtcolorbox`[→P.9] or friends, some automated options are applied.

/tcb/every box (style)

By default, this style is empty.

```
% default setting:
\tcbset{every box/.style={}}
```

It may be changed by redefining this style.

```
% setting all boxes to be enhanced:
\tcbset{every box/.style={enhanced}}
```

/tcb/every box on layer n (style)

Here, n has to be replaced by a number ranging from 1 to the highest managed layer number (4 by default).

```
% default settings:
\tcbset{
  every box on layer 1/.style={every box},
  every box on layer 2/.style={reset,every box},
  every box on layer 3/.style={reset,every box},
  every box on layer 4/.style={reset,every box},
  }
```

/tcb/every box on higher layers (style)

Higher layers are layers above the highest managed layer number (4 by default).

```
\tcbset{every box on higher layers/.style={reset,every box}}
```

\tcbsetmanagedlayer{⟨*number*⟩}

Replaces the highest managed layer number by ⟨*number*⟩ where 4 is the default. This macro can only be used inside the preamble. Using a ⟨*number*⟩ lower than 4 typically makes no sense, but is not forbidden.

```
% \usepackage{lipsum}
% \tcbuselibrary{skins,breakable}
\tcbset{enhanced,breakable,colframe=red!75!black,fonttitle=\bfseries,
  colback=red!5!white,
  every box/.style={enhanced,watermark text=\thetcblayer,
    before=\par\smallskip,after=\par\smallskip},
  every box on layer 2/.style={reset,every box,colback=yellow!10!white,
    drop fuzzy shadow}}
\begin{tcolorbox}[title=Layer 1 Box]
Here comes a footnote\footnote{Footnote from layer 1 box}.
\lipsum[2]
  \begin{tcolorbox}[title=Layer 2 Box]
  abc\footnote{The footnote of abc}
  \end{tcolorbox}
  \begin{tcolorbox}[title=Another Box,ams equation]
    \tcbhighmath{\sum\limits_{n=1}^{\infty} \frac{1}{n}} = \infty.
  \end{tcolorbox}
Some text\footnote{Footnote from some text}.
  \begin{tcolorbox}[title=Yet Another Box]
    \tcboxfit[height=2cm]{\lipsum[1]}
    My text.
    \begin{tcolorbox}
      Another lipsum text\footnote{A lipsum text}. \lipsum[3]
      \begin{tcolorbox}[title=Layer 4,colframe=blue,colback=white]
        Layer 4\footnote{Layer 4 footnote}
      \end{tcolorbox}
      The End\footnote{Last footnote}.
    \end{tcolorbox}
  \end{tcolorbox}
\end{tcolorbox}
```

> **Layer 1 Box**
>
> Here comes a footnote[a]. 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.

Layer 2 Box

abc[a]

---
[a]The footnote of abc

Another Box

$$\sum_{n=1}^{\infty} \frac{1}{n} = \infty. \tag{1}$$

Some text[b].

Yet Another Box

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

My text.

Another lipsum text[a]. 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.

Layer 4

Layer 4[a]

---
[a]Layer 4 footnote

The End[b].

---
[a]A lipsum text
[b]Last footnote

---
[a]Footnote from layer 1 box
[b]Footnote from some text

3.13 Capture Mode

/tcb/capture=⟨*mode*⟩   (no default, initially `minipage`)

The capture ⟨*mode*⟩ defines how the box content is processed.

Feasible values for ⟨*mode*⟩ are:

- `minipage`:
  This is the default ⟨*mode*⟩ for `tcolorbox`[→P. 7]. The content may have an upper and a lower part. Optionally, the box can be `/tcb/breakable`[→P. 171]. The box content is put into a minipage or into something similar to a minipage.
- `hbox`:
  This is the default ⟨*mode*⟩ for `\tcbox`[→P. 8]. The content cannot have a lower part and cannot be broken. The colored box is sized according to the dimensions of the content. A shortcut to set this mode is `/tcb/hbox`.
- `fitbox`: (needs the ⌷ LIB ⌷ fitting library)
  This is the default ⟨*mode*⟩ for `\tcboxfit`[→P. 181]. The content cannot have a lower part and cannot be broken. The content is sized according to the dimensions of the colored box. A shortcut to set this mode is `/tcb/fit`[→P. 183].

```
\tcbset{colframe=blue!75!black,colback=white}

\begin{tcolorbox}[capture=minipage]
This is a tcolorbox.
\end{tcolorbox}

\begin{tcolorbox}[capture=hbox]
This is a tcolorbox.
\end{tcolorbox}

\begin{tcolorbox}[capture=fitbox,height=9mm]% needs the 'fitting' library
This is a tcolorbox.
\end{tcolorbox}
```

> This is a tcolorbox.

> This is a tcolorbox.
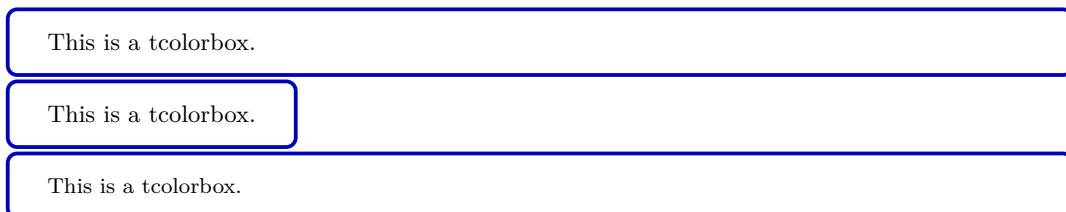
> This is a tcolorbox.

/tcb/hbox   (style, no default)

Shortcut for `capture=hbox`.

```
\tcbset{colframe=blue!75!black,colback=white}

\begin{tcolorbox}[hbox]
This is a tcolorbox.
\end{tcolorbox}
```

> This is a tcolorbox.
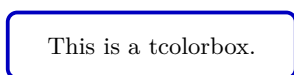
/tcb/minipage   (style, no default)

Shortcut for `capture=minipage`.

3.14 Text Characteristics

/tcb/parbox=⟨*boolean value*⟩ (default `true`, initially `true`)

The text inside a `tcolorbox` is formatted using a LATEX `minipage` if the box is unbreakable. If breakable, the box tries a mimicry of a `minipage`. In a `minipage` or `parbox`, paragraphs are formatted slightly different as the main text. If the key value is set to `false`, the normal main text behavior is restored. In some situations, this has some unwanted side effects. It is recommended to use this experimental setting only where you really want to have this feature.

```
% \usepackage{lipsum}  % preamble
\tcbset{width=(\linewidth-2mm)/2,nobeforeafter,arc=1mm,
  colframe=blue!75!black,colback=white,fonttitle=\bfseries,fontupper=\small,
  left=2mm,right=2mm,top=1mm,bottom=1mm,equal height group=parbox}

\begin{tcolorbox}[parbox,adjusted title={parbox=true (normal)}]
  \lipsum[1-2]
\end{tcolorbox}\hfill%
\begin{tcolorbox}[parbox=false,adjusted title={parbox=false}]
  \lipsum[1-2]
\end{tcolorbox}%
```

parbox=true (normal)	parbox=false
Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.	Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

3.15 Files

/tcb/tempfile=⟨*file name*⟩   (no default, initially `\jobname.tcbtemp`)

Sets ⟨*file name*⟩ as name for the temporary file which is used inside `tcbwritetemp`^{→P.70} and `\tcbusetemp`^{→P.70} implicitely.

3.16 \tcbox Specials

The following options are applicable for `\tcbox`^{→P.8} and `\tcboxmath`^{→P.156} only.

/tcb/tcbox raise=⟨*length*⟩   (no default, initially `0pt`)

Raises the `\tcbox`^{→P.8} by the given ⟨*length*⟩.

```
\tcbset{colframe=blue!50!black,colback=white,colupper=red!50!black,
        fonttitle=\bfseries,before=,after=,center title}

Test\dotfill
\tcbox[tcbox raise base]{Hello World 1}\dotfill
\tcbox{Hello World 2}\dotfill
\tcbox[tcbox raise=5mm]{Hello World 3}
```

Test . . . . . .   Hello World 1   . . . . . .   Hello World 2   . . . . . .   Hello World 3

/tcb/tcbox raise base   (style, no value, initially unset)

Raises the `\tcbox`^{→P.8} such that the base of its content matches the base of the environmental line; see example above.

3.17 Skins

There are additional option keys which change the appearance of a `tcolorbox`. If only the core package is used, there is only one *skin* and these keys are meaningless. The library **LIB** skins adds more skins. The appropriate option keys for skins of the core package are therefore described in section 6.1 from page 71.

3.18 Counters, Labels and References

/tcb/phantom=⟨*code*⟩ (no default, initially unset)

The ⟨*code*⟩ is put in a box at the upper left corner of the `tcolorbox`. If the `tcolorbox` is breakable, the ⟨*code*⟩ is executed for the first box of the break sequence only. If there already was some phantom code given, the new ⟨*code*⟩ is appended.

The ⟨*code*⟩ is intended to be used for counter stepping, labelling, and related operations which do not produce visible text.

- The ⟨*code*⟩ is executed before the title and box content, i. e. counter values are ensured to be increased before usage.
- Labels are ensured to reference the correct page number.
- The ⟨*code*⟩ is executed only once even during fitting operations for title and box content.
- In combination with the `hyperref` package, the hyper anchor is set to the upper left corner of the `tcolorbox`, i. e. links inside the pdf document will jump to the box pleasantly.
- Since the ⟨*code*⟩ is executed inside a TEX group, only global operations can survive this group.

Examples for the `phantom` usage are given in Section 7.8 from page 149, e. g. Example 7.1 on page 150.

/tcb/nophantom (no value, initially set)

Removes the phantom code if set before.

/tcb/label=⟨*marker*⟩ (no default, initially unset)

The ⟨*marker*⟩ is set as label text for a reference with the `\ref` macro. Typically, this option is used for numbered boxes, see Subsection 4.1 from page 65, e. g. `/tcb/new/auto counter`→P. 65.

/tcb/label type=⟨*type*⟩ (no default, initially unset)

This option key can be used only in conjunction with the `cleveref` package [4] which has to be loaded separately. ⟨*type*⟩ has to be a cross-reference type *known* to `cleveref` like `theorem`, `algorithm`, `result`, etc. References made with `cleveref` will use this type. Note that using `label type` will result in compilation errors, if `cleveref` is not loaded. For an example, see Theorem 8.3.5 on page 166.

/tcb/no label type (no value, initially set)

Removes a `/tcb/label type`, if set before.

/tcb/step=⟨*counter*⟩ (no default, initially unset)

Shortcut for `phantom={\refstepcounter{#1}}`. The given ⟨*counter*⟩ is increased and ready for labelling. This option is not needed when using the convenient automated numbering introduced with version 2.40, see Subsection 4.1 from page 65.

/tcb/step and label={⟨*counter*⟩}{⟨*marker*⟩} (no default, initially unset)

Shortcut for using `/tcb/step` and `/tcb/label`. This option is not needed when using the convenient automated numbering introduced with version 2.40, see Subsection 4.1 from page 65.

/tcb/list entry=⟨*text*⟩ (no default, initially unset)

If the «list of tcolorbox(es)» feature described in Subsection 4.2 from page 69 is used, this key describes the ⟨*text*⟩ for an entry into the generated list, e. g.

```
list entry={\numberline{\thetcbcounter}My beautiful Example}
```

See Section 7.8 from page 149 for a complete example.

/tcb/add to list={⟨*list*⟩}{⟨*type*⟩} (no default, initially unset)

If the «list of tcolorbox(es)» feature described in Subsection 4.2 from page 69 is used, list entries are generated automatically. With this key, you can enforce an entry to the given ⟨*list*⟩ with the given ⟨*type*⟩. This issues:
\addcontentsline{⟨*list*⟩}{⟨*type*⟩}{⟨*entry text*⟩}

/tcb/check odd page=⟨*boolean value*⟩ (default `true`, initially `false`)

If set to `true`, the even/odd page testing from the package `changepage` is applied. The /tcb/toggle enlargement^{→P.53} check and the /tcb/toggle left and right^{→P.25} check will use the `\ifoddpage` macro from this package. This options is independent from /tcb/phantom^{→P.62}. Note that you have to include the package `changepage` by hand[2]:

```
\usepackage{changepage}
\strictpagecheck
```

The macro `\ifoddpage` can be used inside overlay or watermark code to test if the box is on an odd page. This will work also for boxes in a break sequence. Note that you cannot use the test inside the normal box content.

```
% \usepackage{changepage}
\tcbset{colframe=blue!75!black,colback=white}

\begin{tcolorbox}[enhanced,check odd page,
  watermark text={\ifoddpage Odd\else Even\fi\ page!}]
\lipsum[1]
\end{tcolorbox}
```

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

---

[2]If `changepage` is not included, a compilation error will arise.

3.19 Externalization

If the *externalization* library of the `tikz` package is used and `/tcb/graphical environment`[→P.72] is set to `tikzpicture`, a `tcolorbox` could trigger the externalization process which will arise a compilation error.

To avoid this, there are two possible strategies:

- Ensure, that `\tikzexternaldisable` is set before a `tcolorbox` is used. If you typically use the pattern `\tikzexternalenable` *some picture* `\tikzexternaldisable`, there is nothing to care about.

- If *externalization* is enabled globally, use `/tcb/shield externalize` to shield any `tcolorbox`. The preamble code could look like this:

```
\usetikzlibrary{external}
\tikzexternalize
\tcbset{shield externalize}
```

`/tcb/shield externalize`=⟨*boolean value*⟩   (default `true`, initially `false`)
   If set to `true`, the drawing part of the `tcolorbox` is not being externalized which is a good thing at the current state of art. Nevertheless, if the `tcolorbox` contains a `tikzpicture`, this picture is still externalized. Pictures drawn with help of `/tcb/tikz upper`[→P.39] or alike are *not* externalized.

If a `tcolorbox` is used inside a node of an encircling `tikzpicture` which is externalized, do *not* use `\tikzexternaldisable` in front of the `tcolorbox`. `/tcb/shield externalize` is deactivated automatically inside a `tikzpicture`.

3.20 Miscellaneous

`/tcb/reset`   (no value, initially set)
   Sets (nearly) all `tcolorbox` settings (including loaded libraries) back to their default values. `/tcb/savedelimiter`[→P.14] and `/tcb/capture`[→P.59] keep their values. This option is useful for boxes in boxes where the inner box should not inherit the settings of the outer box. Note that for boxes inside boxes the `reset` is done automatically, if the standard settings of the package are used (v2.40 and above), see Section 3.12 from page 56.

4 Initialization Option Keys

The *initialization* options are only applicable for the generation of new environments and commands based on `tcolorbox` and friends. Particularly, they can be used for

- `\newtcolorbox`^{→P.9},

- `\newtcbox`^{→P.10},

- `\newtcblisting`^{→P.134},

- `\newtcbinputlisting`^{→P.135},

- `\newtcbtheorem`^{→P.155}, and

- `\newtcboxfit`^{→P.182}.

> Typically, these options may generate counters and alike. It is **strongly** recommended to use initialization options inside the preamble only. Otherwise, you may get trouble when using LaTeX's `\include` features.

4.1 Numbered Boxes

Counters assigned using the initialization options are administrated automatically. Especially, they are increased for each new box. Independent from the real counter name, the counter value can be referenced by **\thetcbcounter**, e. g. inside the title of the box. The real counter name is stored inside **\tcbcounter**.

/tcb/new/auto counter (no value, initially unset)
Creates a new counter automatically. With **/tcb/new/number format**^{→P.67} and **/tcb/new/number within**^{→P.67}, the appearance and behavior of the counter can be changed. The counter value is referenced by **\thetcbcounter**.

```
Definition in the preamble:
\newtcolorbox[auto counter,number within=section]{pabox}[2][]{%
  colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries,
  title=Examp.~\thetcbcounter: #2,#1}
```

```
\begin{pabox}[label={myautocounter}]{Title with number}
This box is automatically numbered with \ref{myautocounter} on page
\pageref{myautocounter}. Inside the box, the \thetcbcounter\ can
also be referenced by |\thetcbcounter|.
The real counter name is \texttt{\tcbcounter}.
\end{pabox}
```

> **Examp. 4.1: Title with number**
>
> This box is automatically numbered with 4.1 on page 65. Inside the box, the 4.1 can also be referenced by \thetcbcounter. The real counter name is `tcb@cnt@pabox`.

/tcb/new/use counter from=⟨*tcolorbox*⟩ (no default, initially unset)

Here, a counter from another ⟨*tcolorbox*⟩ is reused. Note that the setting for /tcb/new/number format[→P. 67] and /tcb/new/number within[→P. 67] are inherited and cannot be changed. The counter value is referenced by **\thetcbcounter**.

```
\newtcolorbox[use counter from=pabox]{mybox}[2][]{%
colback=blue!5!white,colframe=blue!75!black,fonttitle=\bfseries,
title=Some Box \thetcbcounter: #2,#1}

\begin{mybox}[label={myusecounterfrom}]{Title with continued number}
This box is automatically numbered with \ref{myusecounterfrom} on page
\pageref{myusecounterfrom}. Inside the box, the \thetcbcounter\ can
also be referenced by |\thetcbcounter|.
The real counter name is \texttt{\tcbcounter}.
\end{mybox}
```

> **Some Box 4.2: Title with continued number**
>
> This box is automatically numbered with 4.2 on page 66. Inside the box, the 4.2 can also be referenced by \thetcbcounter. The real counter name is tcb@cnt@pabox.

/tcb/new/use counter=⟨*counter*⟩ (no default, initially unset)

Here, an ordinary existing LaTeX counter is used for numbering. With /tcb/new/number format[→P. 67] and /tcb/new/number within[→P. 67], the appearance and behavior of the counter can be changed. The counter value is referenced by **\thetcbcounter**.

```
% \newcounter{myexample}%  preamble
\newtcolorbox[use counter=myexample,number format=\Alph]{mybox}[2][]{%
colback=green!5!white,colframe=green!55!black,fonttitle=\bfseries,
title=Some Box \thetcbcounter: #2,#1}

\begin{mybox}[label={myusecounter}]{Title with \LaTeX\ number}
This box is automatically numbered with \ref{myusecounter} on page
\pageref{myusecounter}. Inside the box, the \thetcbcounter\ can
also be referenced by |\thetcbcounter|.
The real counter name is \texttt{\tcbcounter}.
\end{mybox}
```

> **Some Box A: Title with LaTeX number**
>
> This box is automatically numbered with A on page 66. Inside the box, the A can also be referenced by \thetcbcounter. The real counter name is myexample.

/tcb/new/no counter (no value, initially set)

The created boxes are not numbered. This is the default. The option may be used to overrule a previous option.

/tcb/new/number within=⟨*counter*⟩   (no default, initially unset)
    The automatic counter is set to zero, if ⟨*counter*⟩ is increased. Additionally, during output,
    the value of ⟨*counter*⟩ is prepended to the value of the automatic counter.
    To prepend the automatic counter with the chapter number and to reset it with every new
    chapter, use:

    number within=chapter

    See /tcb/new/use counter^{→P.66} for a complete example.

/tcb/new/number format=⟨*format macro*⟩   (no default, initially `\arabic`)
    Declares the format of the automatic counter. The ⟨*format macro*⟩ can be any valid LaTeX
    number formatting macro like `\arabic`, `\roman`, etc.
    To display the counter value in large roman numbers, use:

    number format=\Roman

    See /tcb/new/auto counter^{→P.65} for a complete example.

/tcb/new/number freestyle=⟨*code*⟩   (no default, initially unset)
    Allows advanced control over the complete number format. This option overrules the format
    given by /tcb/new/number within and /tcb/new/number format. Nevertheless, you can
    combine it with /tcb/new/number within to get the desired reset property.
    The ⟨*code*⟩ is some formatting code which should contain `\tcbcounter` to reference the
    automated counter. Since this ⟨*code*⟩ is expanded, you have to secure each macro with
    `\noexpand` with *exception* of `\tcbcounter`.

Definition in the preamble:

```
\newtcolorbox[auto counter,number within=section,
  number freestyle={(Q/\noexpand\thesection/\noexpand\Alph{\tcbcounter})},
  ]{phbox}[2][]{%
  colback=yellow!15!white,colframe=blue!75!black,fonttitle=\bfseries,
  title=Question~\thetcbcounter: #2,#1}
```

```
\begin{phbox}[label={myfreestyle}]{Title with freestyle number}
This box is automatically numbered with \ref{myfreestyle} on page
\pageref{myfreestyle}. Inside the box, the \thetcbcounter\ can
also be referenced by |\thetcbcounter|.
The real counter name is \texttt{\tcbcounter}.
\end{phbox}
```

Question (Q/4/A): Title with freestyle number

This box is automatically numbered with (Q/4/A) on page 67. Inside the box, the (Q/4/A)
can also be referenced by `\thetcbcounter`. The real counter name is `tcb@cnt@phbox`.

/tcb/new/crefname={⟨*singular*⟩}{⟨*plural*⟩}   (no default, initially unset)

This option key can be used only in conjunction with the `cleveref` package [4] which has to be loaded separately. It creates a cross-reference type for the new `tcolorbox`'es, where the lowercase ⟨*singular*⟩ and ⟨*plural*⟩ forms of the cross-reference are given. See `/tcb/label type`^{→P. 62} and [4] for more information.

/tcb/new/Crefname={⟨*singular*⟩}{⟨*plural*⟩}   (no default, initially unset)

This option key can be used only in conjunction with the `cleveref` package [4] which has to be loaded separately. It creates a cross-reference type for the new `tcolorbox`'es, where the uppercase ⟨*singular*⟩ and ⟨*plural*⟩ forms of the cross-reference are given. See `/tcb/label type`^{→P. 62} and [4] for more information.

4.2 Lists of `tcolorboxes`

For figures and tables, LaTeX provides the `\listoffigures` and `\listoftables` commands to create lists of these numbered entities. Also, a `tcolorbox` can be part of such a kind of list.

1. Assign a list ⟨*name*⟩ by the *initialization* option `/tcb/new/list inside`.

2. Optionally, a new ⟨*type*⟩ for list entries may be assigned by the *initialization* option `/tcb/new/list type`.

3. List entries a generated automatically within each new `tcolorbox` using the above initialization.

    • If `/tcb/list entry`$^{\to P.63}$ is set, the entry is generated with it.

    • Otherwise, if `/tcb/title`$^{\to P.11}$ is set, the entry is generated with it.

    • Otherwise, the entry is generated with the current number and the environment name.

4. The generated list is displayed by `\tcblistof`.

/tcb/new/list inside=⟨*name*⟩   (no default, initially unset)
Assigns a list or contents file to the generated `tcolorbox`es. Entries to this list are saved to a file which gets the ⟨*name*⟩ as file name extension. The list is references by this name in `\tcblistof`. For example:

    list inside=exam

See Section 7.8 from page 149 for a complete example.

/tcb/new/list type=⟨*type*⟩   (no default, initially `tcolorbox`)
Optionally, some ⟨*type*⟩ can be assigned to the list entries. For a new ⟨*type*⟩, a macro `\l@`⟨*type*⟩ has to exist which controls the format of the list entry. The default type is defined by

    \newcommand*\l@tcolorbox{\@dottedtocline{1}{1.5em}{2.3em}}

This is identical to the `\l@section` setting of LaTeX. `\l@tcolorbox` can be redefined or a new ⟨*type*⟩ can be assigned.

\tcblistof[⟨*macro*⟩]{⟨*name*⟩}{⟨*title text*⟩}
Displays the generated list of `tcolorbox`es with the given ⟨*name*⟩. The heading is generated by ⟨*macro*⟩{⟨*title text*⟩} where `\section` is the default setting for ⟨*macro*⟩.
To display the list inside a subsection, use for example:

    \tcblistof[\subsection]{exam}{List of Exercises}

The result of the example is found as Subsection 7.9 on page 152.
The core of the list is generated by `\@starttoc{`⟨*name*⟩`}` which can be wrapped into an own macro.

5 Saving and Loading of Verbatim Texts

The following macros are slightly modified versions of the original macros from the known packages `moreverb` and `verbatim`. They are used implicitly inside of a `tcolorbox` environment, but they can be used outside also.

\begin{**tcbverbatimwrite**}{⟨*file name*⟩}
  ⟨*environment content*⟩
\end{**tcbverbatimwrite**}

> Saves the ⟨*environment content*⟩ to a file named by ⟨*file name*⟩. TEX macros inside the environment are not expanded.

```
\begin{tcbverbatimwrite}{\jobname_verbexp.tex}
  This text is saved \textit{as is}.
\end{tcbverbatimwrite}

Now, we are using the file:\par
\input{\jobname_verbexp.tex}
```

Now, we are using the file:
This text is saved *as is*.

This environment may be used inside an own environment. Note, that inside the environment definition \tcbverbatimwrite has to be used instead of \begin{tcbverbatimwrite} and \endtcbverbatimwrite instead of \end{tcbverbatimwrite}.

```
\newenvironment{myverbatim}{%
  \begingroup\tcbverbatimwrite{\jobname_myverb.tex}}%
  {\endtcbverbatimwrite\endgroup}

\begin{myverbatim}
  This is the text which is saved by my own environment.
\end{myverbatim}

Now, we are using the file:\par
\input{\jobname_myverb.tex}
```

Now, we are using the file:
This is the text which is saved by my own environment.

\begin{**tcbwritetemp**}
  ⟨*environment content*⟩
\end{**tcbwritetemp**}

> Has the same function as **tcbverbatimwrite**, but uses the key value of `tempfile` for the file name.

```
\begin{tcbwritetemp}
  This text is saved \textit{as is}.
\end{tcbwritetemp}

Now, we are using the file:\par
\tcbusetemp
```

Now, we are using the file:
This text is saved *as is*.

\tcbusetemp

> Loads the current temporary file which was saved by **tcbwritetemp**.

6 Library ⸤LIB⸥ skins

The library is loaded by a package option or inside the preamble by:

```
\tcbuselibrary{skins}
```

This also loads the package tikz [17]. Typically but not necessarily, the following skins use tikz instead of pgf.

6.1 Technical Overview and Core Package Option Keys

From a technical point of view, a *skin* is a style definition for the appearance of a tcolorbox. The core package provides some additional option keys for skins but only a single skin called standard^{→P.98}. The ⸤LIB⸥ skins library adds several more skins. To change a skin, only one option from the core package has to be set.

/tcb/skin=⟨*name*⟩   (style, no default, initially standard)

Sets the current skin to ⟨*name*⟩. This is a style definition which sets all the following keys, i. e. for many use cases there is nothing more to do.

```
\tcbset{colback=Salmon!50!white,colframe=FireBrick!75!black,
  width=(\linewidth-8mm)/2,before=,after=\hfill,equal height group=ske}

\begin{tcolorbox}[adjusted title=My title]
  This is my content.
\end{tcolorbox}
\begin{tcolorbox}[skin=beamer,beamer,adjusted title=My title]
  This is my content.
\end{tcolorbox}
```

My title	**My title**
This is my content.	This is my content.
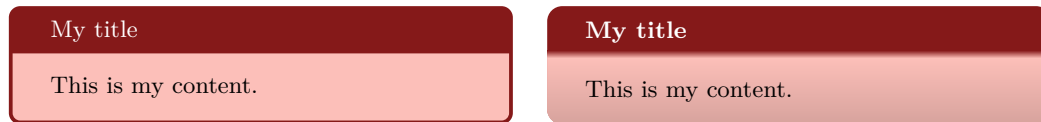
On first read, you may skip the rest of this subsection and proceed to Subsection 6.2 on page 74. All following keys in this subsection are automatically set by the selected skin and you may never have to temper with them. Nevertheless, they can be used after a skin was selected to modify this skin.

/tcb/skin first=⟨*name*⟩   (style, no default, initially standard)

If the box is set to be /tcb/breakable^{→P.171} and *is* broken actually, then the skin for the *first* part of the break sequence is set to ⟨*name*⟩, see Subsection 9.5 on page 175. Typically, this key is set by a /tcb/skin.

/tcb/skin middle=⟨*name*⟩   (style, no default, initially standard)

If the box is set to be /tcb/breakable^{→P.171} and *is* broken actually, then the skin for the *middle* parts (if any) of the break sequence is set to ⟨*name*⟩, see Subsection 9.5 on page 175. Typically, this key is set by a /tcb/skin.

/tcb/skin last=⟨*name*⟩   (style, no default, initially standard)

If the box is set to be /tcb/breakable^{→P.171} and *is* broken actually, then the skin for the *last* part of the break sequence is set to ⟨*name*⟩, see Subsection 9.5 on page 175. Typically, this key is set by a /tcb/skin.

71

/tcb/graphical environment=⟨*name*⟩   (no default, initially `pgfpicture`)

Sets the graphical environment for the `tcolorbox` to ⟨*name*⟩. Feasible values are `pgfpicture` and `tikzpicture` or environments which inherit from one of these two. This key is set by a /tcb/skin^{→P.71} and may seldom be used directly.

The skin of a `tcolorbox` is drawn by up to four *engines*. Afterwards, the text content is drawn which is not part of a skin. The four steps are:

1. The *frame* of the box.

2. The *interior* of the box. The interior of a box with title is drawn differently from a box without title.

3. The *segmentation* (line) of the box, if there is a lower part.

4. The *title area* of the box, if there is a title.

Every engine for the up to four steps can be set to one of the following types:

1. `standard`: the original code from the core package.

2. `path`: a `tikz` path which can be controlled by options.

3. `pathfirst`: a `tikz` path which can be controlled by options.

4. `pathmiddle`: a `tikz` path which can be controlled by options.

5. `pathlast`: a `tikz` path which can be controlled by options.

6. `freelance`: arbitrary user code.

/tcb/frame engine=⟨*name*⟩   (no default, initially `standard`)

Sets the *frame* drawing engine for a box to ⟨*name*⟩. Typically, this key is set by a /tcb/skin^{→P.71}. Feasible values for ⟨*name*⟩ are:
- `standard`: the original code from the core package,
- `path`: a `tikz` path which is controlled by /tcb/frame style^{→P.74},
- `pathfirst`: a `tikz` path which is controlled by /tcb/frame style^{→P.74},
- `pathmiddle`: a `tikz` path which is controlled by /tcb/frame style^{→P.74},
- `pathlast`: a `tikz` path which is controlled by /tcb/frame style^{→P.74},
- `freelance`: arbitrary user code which is given by /tcb/frame code^{→P.77}.

/tcb/interior titled engine=⟨*name*⟩   (no default, initially `standard`)

Sets the *interior* drawing engine for a titled box to ⟨*name*⟩. Typically, this key is set by a /tcb/skin^{→P.71}. Feasible values for ⟨*name*⟩ are:
- `standard`: the original code from the core package,
- `path`: a `tikz` path which is controlled by /tcb/interior style^{→P.74},
- `pathfirst`: a `tikz` path which is controlled by /tcb/interior style^{→P.74},
- `pathmiddle`: a `tikz` path which is controlled by /tcb/interior style^{→P.74},
- `pathlast`: a `tikz` path which is controlled by /tcb/interior style^{→P.74},
- `freelance`: arbitrary user code which is given by /tcb/interior titled code^{→P.77}.

/tcb/interior engine=⟨*name*⟩   (no default, initially `standard`)

Sets the *interior* drawing engine for an untitled box to ⟨*name*⟩. Typically, this key is set by a /tcb/skin^{→P.71}. Feasible values for ⟨*name*⟩ are:
- `standard`: the original code from the core package,
- `path`: a `tikz` path which is controlled by /tcb/interior style^{→P.74},
- `pathfirst`: a `tikz` path which is controlled by /tcb/interior style^{→P.74},
- `pathmiddle`: a `tikz` path which is controlled by /tcb/interior style^{→P.74},
- `pathlast`: a `tikz` path which is controlled by /tcb/interior style^{→P.74},
- `freelance`: arbitrary user code which is given by /tcb/interior code^{→P.78}.

/tcb/segmentation engine=⟨*name*⟩ (no default, initially `standard`)

Sets the *segmentation* (line) drawing engine for a box to ⟨*name*⟩. Typically, this key is set by a `/tcb/skin`[→P.71]. Feasible values for ⟨*name*⟩ are:

- `standard`: the original code from the core package,
- `path`: a `tikz` path which is controlled by `/tcb/segmentation style`[→P.75],
- `freelance`: arbitrary user code which is given by `/tcb/segmentation code`[→P.78].

/tcb/title engine=⟨*name*⟩ (no default, initially `standard`)

Sets the *title area* drawing engine for a titled box to ⟨*name*⟩. Typically, this key is set by a `/tcb/skin`[→P.71]. Feasible values for ⟨*name*⟩ are:

- `standard`: the original code from the core package,
- `path`: a `tikz` path which is controlled by `/tcb/title style`[→P.76],
- `pathfirst`: a `tikz` path which is controlled by `/tcb/title style`[→P.76],
- `pathmiddle`: a `tikz` path which is controlled by `/tcb/title style`[→P.76],
- `pathlast`: a `tikz` path which is controlled by `/tcb/title style`[→P.76],
- `freelance`: arbitrary user code which is given by `/tcb/title code`[→P.79].

/tcb/geometry nodes=⟨*boolean value*⟩ (default `true`, initially `false`)

If set to `true`, up to four `tikz` nodes are defined for a `tcolorbox` which are named `frame`, `interior`, `segmentation`, and `title`. These nodes describe the boundaries of the equally named parts of a `tcolorbox`. They are used by all engines of type `path` and they may be used by engines of type `freelance`. Typically, this key is set by a `/tcb/skin`[→P.71].

```
\tcbset{colback=Salmon!50!white,colframe=FireBrick!75!black,
  width=(\linewidth-8mm)/2,before=,after=\hfill,equal height group=geon}

\begin{tcolorbox}[adjusted title=The title]
  The upper part.\tcblower The lower part.
\end{tcolorbox}
\begin{tcolorbox}[adjusted title=The title,skin=freelance,
  frame code={\path[draw=red,fill=red!25]
      (frame.south west) rectangle (frame.north east);},
  interior titled code={\path[draw=blue,fill=blue!25]
      (interior.south west) rectangle (interior.north east);},
  segmentation code={\path[draw=green,fill=green!25]
      (segmentation.south west) rectangle (segmentation.north east);},
  title code={\path[draw=black,fill=brown!75!black]
      (title.south west) rectangle (title.north east);}]
  The upper part.\tcblower The lower part.
\end{tcolorbox}
```
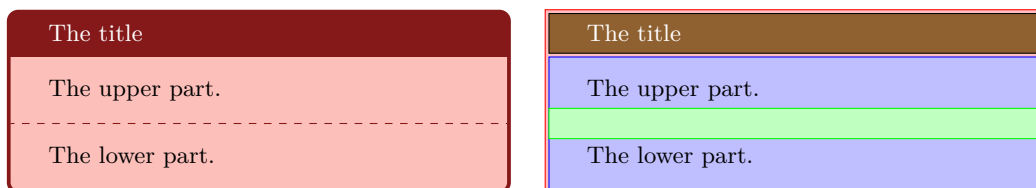
6.2 Style Option Keys

The following style options are applicable for all skins which use engines of type `path`, `pathfirst`, `pathmiddle`, or `pathlast`. Especially, the skin enhanced[→P. 99] supports *all* of them and standard[→P. 98] *none*.

/tcb/frame style=⟨tikz *keys*⟩   (style, no default)

   The ⟨tikz *keys*⟩ are used inside the `tikz` path command for drawing the *frame* of the box. This option is available if the /tcb/frame engine[→P. 72] is set to `path`, `pathfirst`, `pathmiddle`, or `pathlast`. It is *not* available for `standard` and it *may* be applicable for `freelance`.

```
\tcbset{colback=red!5!white,fonttitle=\bfseries}

\begin{tcolorbox}[enhanced,title=My title,
  frame style={left color=red!75!black,
               right color=blue!75!black}]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

> **My title**
>
> This is a **tcolorbox**.
> -
> This is the lower part.
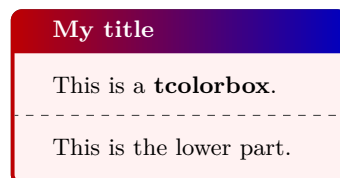
**/tcb/frame hidden**   (style, no value)

   This is a shortcut for `frame style={draw=none,fill=none}`. Depending on the skin, this option switches off the drawing of the frame.

```
\tcbset{colback=red!5!white,colframe=red!75!black,
  fonttitle=\bfseries,coltitle=black}

\begin{tcolorbox}[enhanced,title=My title,
  frame hidden]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

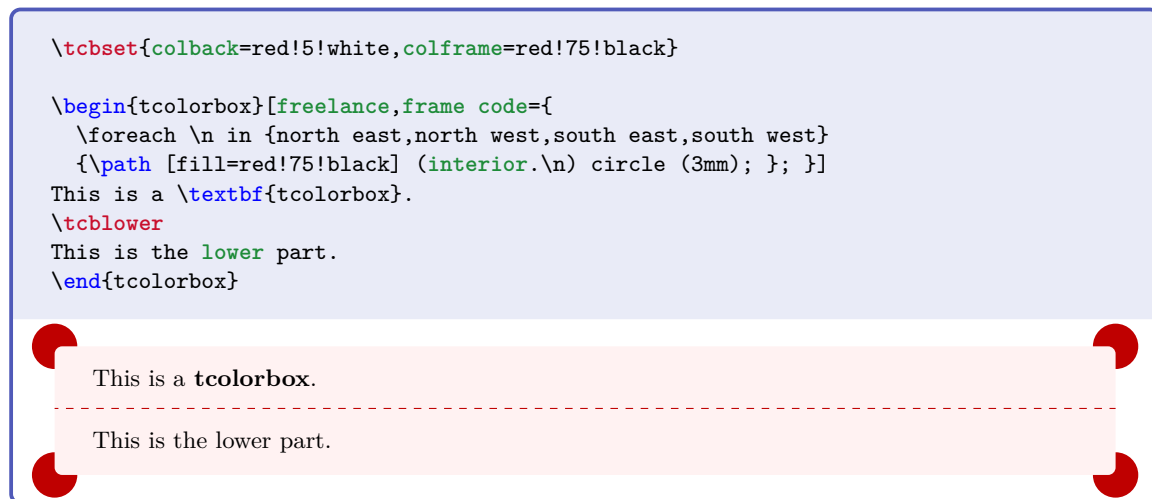> **My title**
>
> This is a **tcolorbox**.
> -
> This is the lower part.

/tcb/interior style=⟨tikz *keys*⟩   (style, no default)

   The ⟨tikz *keys*⟩ are used inside the `tikz` path command for drawing the *interior* of the box. They are used for the titled and for the untitled version as well.
   This option is available if the /tcb/interior titled engine[→P. 72] or /tcb/interior engine[→P. 72] is set to `path`, `pathfirst`, `pathmiddle`, or `pathlast`. It is *not* available for `standard` and it *may* be applicable for `freelance`.

```
\tcbset{colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[enhanced,title=My title,
  interior style={left color=red!20!white,
                  right color=yellow!50!white}]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

> **My title**
>
> This is a **tcolorbox**.
> -
> This is the lower part.

/tcb/interior hidden (style, no value)

This is a shortcut for `interior style={draw=none,fill=none}`. Depending on the skin, this option switches off the drawing of the interior.

```
\tcbset{frame style={top color=red!20!white,
  bottom color=red!20!white!75!black},
  fonttitle=\bfseries,coltitle=black}

\begin{tcolorbox}[enhanced,title=My title,
  interior hidden]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

> **My title**
>
> This is a **tcolorbox**.
> -
> This is the lower part.

/tcb/segmentation style=⟨tikz *keys*⟩ (style, no default)

The ⟨`tikz` *keys*⟩ are used inside the `tikz` path command for drawing the *segmentation* line of the box.

This option is available if the [/tcb/segmentation engine](→ P. 73) is set to `path`. It is *not* available for `standard` and it *may* be applicable for `freelance`.

```
\tcbset{colback=red!5!white,colframe=red!75!black,
  fonttitle=\bfseries}

\begin{tcolorbox}[enhanced,title=My title,
  segmentation style={double=white,draw=blue,
                double distance=1pt,solid}]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

> **My title**
>
> This is a **tcolorbox**.
> ═══════════════════
> This is the lower part.

/tcb/segmentation hidden (style, no value)

This is a shortcut for `segmentation style={draw=none,fill=none}`. Depending on the skin, this option switches off the drawing of the segmentation line. See also [/tcb/lower separated](→ P. 13) which has the same effect for most skins.

```
\tcbset{colback=red!5!white,colframe=red!75!black,
  fonttitle=\bfseries}

\begin{tcolorbox}[title=My title,
  enhanced,segmentation hidden]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

> **My title**
>
> This is a **tcolorbox**.
>
> This is the lower part.

/tcb/title style=⟨tikz *keys*⟩ (style, no default)

The ⟨tikz *keys*⟩ are used inside the tikz path command for drawing the *title area* of the box.

This option is available if the /tcb/title engine^{→P.73} is set to `path`, `pathfirst`, `pathmiddle`, or `pathlast`. It is *not* available for `standard` and it *may* be applicable for `freelance`.

```
\tcbset{colback=red!5!white,colframe=red!75!black,
  coltitle=blue!50!black,fonttitle=\bfseries}

\begin{tcolorbox}[enhanced,title=My title,
  title style={left color=blue!15!yellow,
              right color=red!85!black}]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

> **My title**
>
> This is a **tcolorbox**.
>
> This is the lower part.
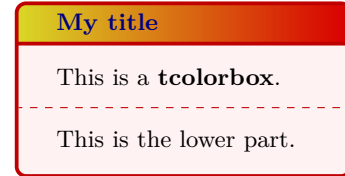
/tcb/colbacktitle=⟨*color*⟩ (style, no default, initially unset)

Abbreviation for setting /tcb/title style to `fill=`⟨*color*⟩. This fills the background of the title area with the given ⟨*color*⟩.

This option is available if the /tcb/title engine^{→P.73} is set to `path`, `pathfirst`, `pathmiddle`, or `pathlast`. It is *not* available for `standard` and it *may* be applicable for `freelance`.

```
\tcbset{colback=red!5!white,colframe=red!75!black,
  coltitle=blue!50!black,fonttitle=\bfseries}

\begin{tcolorbox}[enhanced,title=My title,
  colbacktitle=yellow!50!white]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

> **My title**
>
> This is a **tcolorbox**.
>
> This is the lower part.
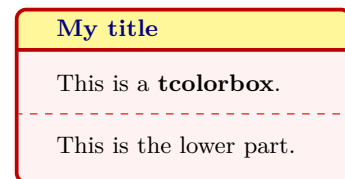
/tcb/title hidden (style, no value)

This is a shortcut for `title style={draw=none,fill=none}`. Depending on the skin, this option switches off the drawing of the title background. This is the default behavior for `standard`^{→P.98} and `enhanced`^{→P.99}.

```
\tcbset{colback=red!5!white,colframe=red!75!black,
  fonttitle=\bfseries}

\begin{tcolorbox}[title=My title,
  enhanced,title hidden]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

> **My title**
>
> This is a **tcolorbox**.
>
> This is the lower part.
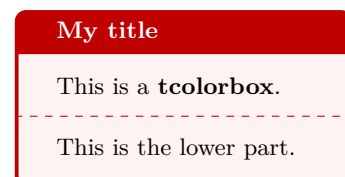
6.3 Code Option Keys

The following code options are applicable for all skins which use engines of type `freelance`. Especially, the skin `freelance`→P.108 supports *all* of them, `standard`→P.98 and `enhanced`→P.99 *none* of them.

/tcb/frame code=⟨*graphical code*⟩   (code, default from `standard`)

The given ⟨*graphical code*⟩ is used for drawing the *frame* of the box.

This option is available only if the `/tcb/frame engine`→P.72 is set to `freelance`.

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[freelance,frame code={
  \foreach \n in {north east,north west,south east,south west}
  {\path [fill=red!75!black] (interior.\n) circle (3mm); }; }]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

> This is a **tcolorbox**.
> -
> This is the lower part.

/tcb/interior titled code=⟨*graphical code*⟩   (code, default from `standard`)

The given ⟨*graphical code*⟩ is used for drawing the *interior* of the box, if the box comes with a title.

This option is available only if the `/tcb/interior titled engine`→P.72 is set to `freelance`.
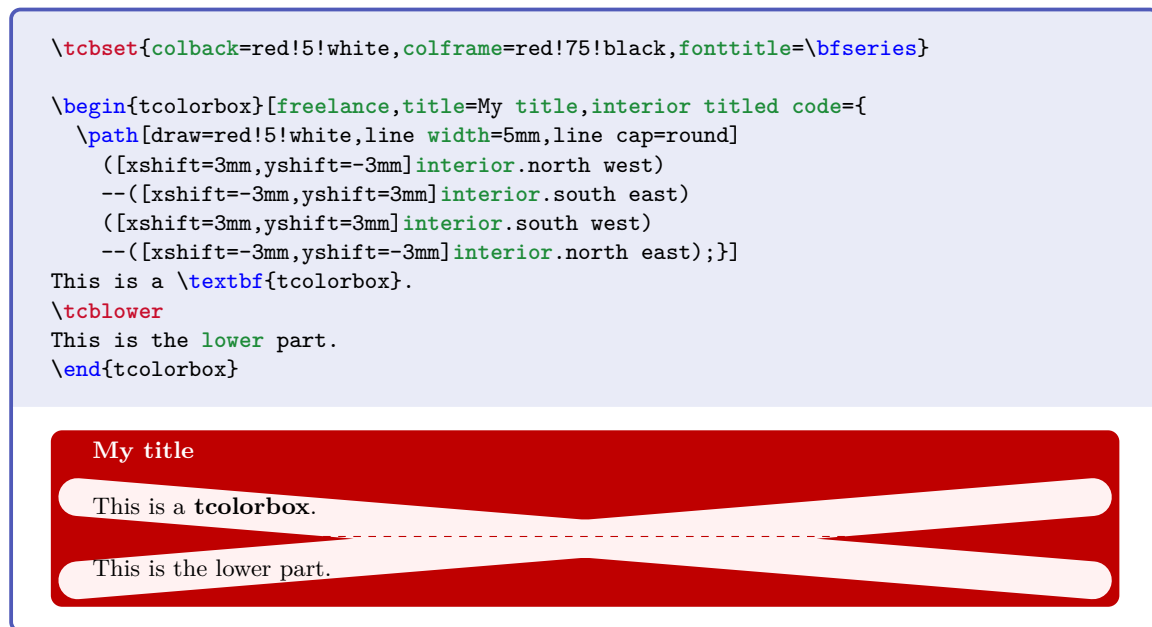
```
\tcbset{colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[freelance,title=My title,interior titled code={
  \path[draw=red!5!white,line width=5mm,line cap=round]
    ([xshift=3mm,yshift=-3mm]interior.north west)
    --([xshift=-3mm,yshift=3mm]interior.south east)
    ([xshift=3mm,yshift=3mm]interior.south west)
    --([xshift=-3mm,yshift=-3mm]interior.north east);}]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

> **My title**
>
> This is a **tcolorbox**.
> -
> This is the lower part.

/tcb/interior code=⟨*graphical code*⟩   (code, default from `standard`)

The given ⟨*graphical code*⟩ is used for drawing the *interior* of the box, if the box is without a title.

This option is available only if the `/tcb/interior engine`^{→P.72} is set to `freelance`.

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[freelance,interior code={
  \path[draw=red!5!white,line width=5mm,line cap=round]
    ([xshift=3mm,yshift=-3mm]interior.north west)
    --([xshift=-3mm,yshift=3mm]interior.south east)
    ([xshift=3mm,yshift=3mm]interior.south west)
    --([xshift=-3mm,yshift=-3mm]interior.north east);}]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```


/tcb/segmentation code=⟨*graphical code*⟩   (code, default from `standard`)
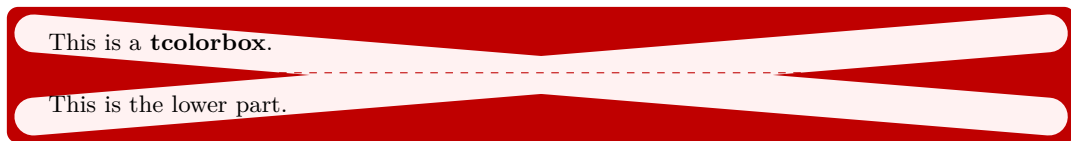
The given ⟨*graphical code*⟩ is used for drawing the *segmentation* area of the box.

This option is available only if the `/tcb/segmentation engine`^{→P.73} is set to `freelance`.

```
\tcbset{colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[freelance,title=My title,segmentation code={
  \path[top color=red!5!white,bottom color=red!5!white,middle color=blue]
  (segmentation.south west) rectangle (segmentation.north east);}]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```

/tcb/title code=⟨*graphical code*⟩   (code, default from `standard`)

The given ⟨*graphical code*⟩ is used for drawing the *title* area of the box.

This option is available only if the /tcb/title engine^{→P.73} is set to `freelance`.

```
\tcbset{colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries,
  coltitle=black}

\begin{tcolorbox}[freelance,title=My title,title code={
  \path[draw=yellow,solid,decorate,line width=2mm,
    decoration={coil,aspect=0,segment length=10.1mm}]
    ([xshift=1mm]title.west) -- ([xshift=-1mm]title.east);}]
This is a \textbf{tcolorbox}.
\tcblower
This is the lower part.
\end{tcolorbox}
```


79
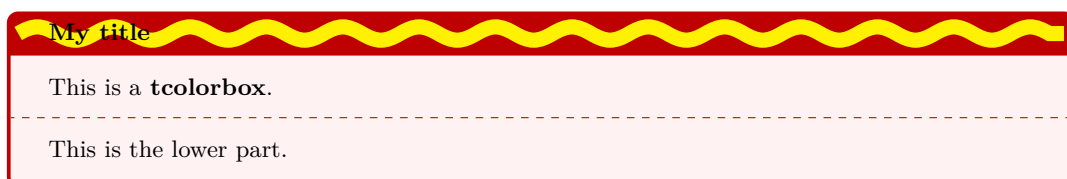
6.4 Watermark Option Keys

The following watermark options are applicable for all skins which use `tikzpicture` as
`/tcb/graphical environment`[→P.72]. Therefore, the skin `standard`[→P.98] does not support these
watermarks, but all other skins, e.g. `enhanced`[→P.99].

> The watermark options rely on the more general overlay options described in Section 3.7 from
> page 41. Therefore, *watermarks* and *overlays* cannot be used mixed. But a mixture is possible
> with the [LIB hooks] library, see Section 11.

`/tcb/watermark text`=⟨*text*⟩ (no default, initially unset)
Writes some ⟨*text*⟩ in the center of the interior region of a `tcolorbox`. This ⟨*text*⟩ is written
after the frame and interior are drawn and *before* the text content is drawn. It is zoomed
or stretched according the values of `/tcb/watermark zoom`[→P.83] or `/tcb/watermark`
`stretch`[→P.85].

```
\tcbset{colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[enhanced,title=My title,watermark text=My Watermark]
\lipsum[1]
\tcblower
\lipsum[2]
\end{tcolorbox}
```

My title

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, plac-
erat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy
eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habi-
tant 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.

`/tcb/watermark text on`=⟨*part*⟩ is ⟨*text*⟩ (no default, initially unset)
This option writes some ⟨*text*⟩ in the center of the interior region of a `tcolorbox` as described
for `/tcb/watermark text`. But this is done only for boxes named ⟨*part*⟩ of a break sequence,
see `/tcb/breakable`[→P.171].
Feasible values for ⟨*part*⟩ are:
  - `broken`: all broken box parts,
  - `unbroken`: unbroken boxes only,
  - `first`: first parts of a break sequence,
  - `middle`: middle parts of a break sequence,
  - `last`: last parts of a break sequence,
  - `unbroken and first`: unbroken boxes and first parts of a break sequence,
  - `middle and last`: middle and last parts of a break sequence.

/tcb/watermark graphics=⟨*file name*⟩   (no default, initially unset)

Draws an external picture referenced by ⟨*file name*⟩ in the center of the interior region of a `tcolorbox`. The picture is drawn *after* the frame and interior are drawn and *before* the text content is drawn. It is zoomed or stretched according the values of /tcb/watermark zoom^{→P.83} or /tcb/watermark stretch^{→P.85}.

```
\tcbset{colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[enhanced,title=My title,watermark graphics=Basilica_5.png,
  watermark opacity=0.15]
\lipsum[1-2]
\tcblower
This example uses a public domain picture from\\
\url{http://commons.wikimedia.org/wiki/File:Basilica_5.png}
\end{tcolorbox}
```


My title

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

This example uses a public domain picture from
http://commons.wikimedia.org/wiki/File:Basilica_5.png

/tcb/watermark graphics on=⟨*part*⟩ is ⟨*file name*⟩   (no default, initially unset)

This option draws a picture referenced by ⟨*file name*⟩ in the center of the interior region of a `tcolorbox` as described for /tcb/watermark graphics. But this is done only for boxes named ⟨*part*⟩ of a break sequence, see /tcb/breakable^{→P.171}.

Feasible values for ⟨*part*⟩ are:

- `broken`: all broken box parts,
- `unbroken`: unbroken boxes only,
- `first`: first parts of a break sequence,
- `middle`: middle parts of a break sequence,
- `last`: last parts of a break sequence,
- `unbroken and first`: unbroken boxes and first parts of a break sequence,
- `middle and last`: middle and last parts of a break sequence.

/tcb/watermark tikz=⟨*graphical code*⟩ (no default, initially unset)

Draws the given `tikz` ⟨*graphical code*⟩ in the center of the interior region of a `tcolorbox`.
The code is executed *after* the frame and interior are drawn and *before* the text content is drawn. The result is zoomed or stretched according the values of or .

```
\tcbset{colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[enhanced,title=My title,
  watermark tikz={\draw[line width=2mm] circle (1cm)
    node{\fontfamily{ptm}\fontseries{b}\fontsize{20mm}{20mm}\selectfont ?};}]
\lipsum[1]
\tcblower
\lipsum[2]
\end{tcolorbox}
```


/tcb/watermark tikz on=⟨*part*⟩ is ⟨*graphical code*⟩ (no default, initially unset)

This option draws the given `tikz` ⟨*graphical code*⟩ in the center of the interior region of a `tcolorbox` as described for `/tcb/watermark tikz`. But this is done only for boxes named ⟨*part*⟩ of a break sequence, see .

Feasible values for ⟨*part*⟩ are:

- `broken`: all broken box parts,
- `unbroken`: unbroken boxes only,
- `first`: first parts of a break sequence,
- `middle`: middle parts of a break sequence,
- `last`: last parts of a break sequence,
- `unbroken and first`: unbroken boxes and first parts of a break sequence,
- `middle and last`: middle and last parts of a break sequence.

/tcb/no watermark (style, no default, initially set)

Removes the watermark if set before. This is an alias for .

`/tcb/watermark opacity`=⟨*fraction*⟩ (no default, initially `1.00`)

Sets the opacity value ∈ [0, 1] for a watermark.

```
\tcbset{enhanced,colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries,
  watermark text=Watermark,nobeforeafter,width=(\linewidth-2mm)/2}

\begin{tcolorbox}[title=Opacity 1.00,watermark opacity=1.00]
\lipsum[2]
\end{tcolorbox}\hfill%
\begin{tcolorbox}[title=Opacity 0.50,watermark opacity=0.50]
\lipsum[2]
\end{tcolorbox}%
```

Opacity 1.00	**Opacity 0.50**
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.	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.

`/tcb/watermark zoom`=⟨*fraction*⟩ (no default, initially `0.75`)

Sets the zoom value for a watermark. The zoom respects the aspect ratio. The value 1.0 means to fill the whole box until the watermark touches the frame.

```
\tcbset{enhanced,colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries,
  watermark text=Watermark,nobeforeafter,width=(\linewidth-2mm)/2}

\begin{tcolorbox}[title=Zoom 1.0,watermark zoom=1.0]
\lipsum[2]
\end{tcolorbox}\hfill%
\begin{tcolorbox}[title=Zoom 0.5,watermark zoom=0.5]
\lipsum[2]
\end{tcolorbox}%
```

Zoom 1.0	**Zoom 0.5**
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.	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.

/tcb/watermark overzoom=⟨*fraction*⟩   (no default, initially unset)

Sets the overzoom value for a watermark. The overzoom respects the aspect ratio. The value 1.0 means to fill the whole box until the watermark touches all four sides of the frame.

```
\tcbset{enhanced,colback=white,colframe=blue!50!black,fonttitle=\bfseries,
  watermark opacity=0.5,
  watermark graphics=lichtspiel.jpg,nobeforeafter,width=(\linewidth-2mm)/2}

\begin{tcolorbox}[title=Zoom 1.0,watermark zoom=1.0]
\lipsum[1]
\end{tcolorbox}\hfill%
\begin{tcolorbox}[title=Overzoom 1.0,watermark overzoom=1.0]
\lipsum[1]
\end{tcolorbox}%
```

Zoom 1.0

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

Overzoom 1.0

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

If a /tcb/watermark overzoom value of 1.0 is used in connection with invisible top and bottom rules which still have a thickness greater than 0pt, the space of these invisible rules may not be covered by the watermark. For example, this situation may occur during the breaking of /tcb/enhanced^{→P. 99} boxes. To avoid this optical glitch, just set /tcb/pad at break^{→P. 173} to any desired value.

/tcb/watermark stretch=⟨*fraction*⟩ (no default, initially unset)

Sets the stretch value for a watermark. The stretch value is applied to width and height in relation to the box dimensions. It does not respect the aspect ratio. The value 1.0 means to fill the whole box.

```
\tcbset{enhanced,colback=white,colframe=blue!50!black,fonttitle=\bfseries,
  watermark graphics=lichtspiel.jpg,watermark opacity=0.5,
  nobeforeafter,width=(\linewidth-2mm)/2}

\begin{tcolorbox}[title=Stretch 1.00,watermark stretch=1.00]
\lipsum[2]
\end{tcolorbox}\hfill%
\begin{tcolorbox}[title=Stretch 0.50,watermark stretch=0.50]
\lipsum[2]
\end{tcolorbox}%
```

Stretch 1.00

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.

Stretch 0.50

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.

/tcb/watermark color=⟨*color*⟩ (no default, initially mixed background and frame color)

Sets the color for the watermark.

```
\tcbset{colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[enhanced,title=My title,watermark text=My Watermark,
  watermark color=yellow!50!red]
\lipsum[1]
\end{tcolorbox}
```
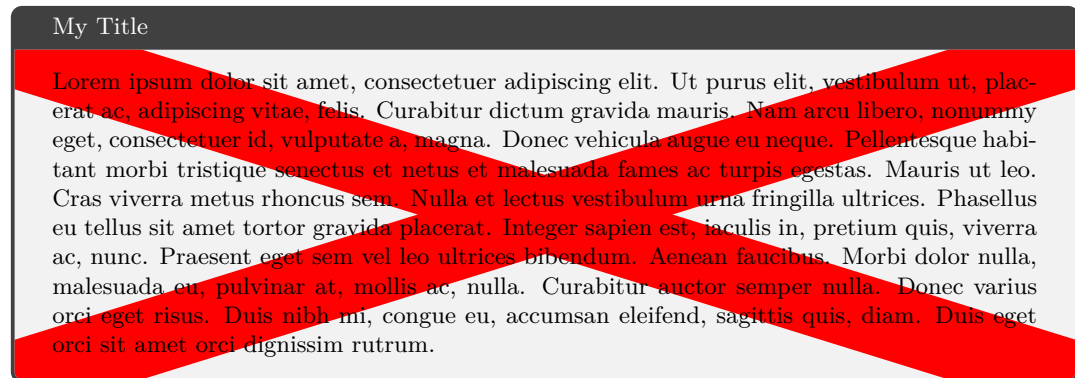
My title

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

/tcb/clip watermark=⟨*boolean value*⟩   (default `true`, initially `true`)

Sets the watermark to be clipped to the interior area.

```
\tcbset{enhanced,colback=white,colframe=blue!50!white,fonttitle=\bfseries,
  watermark opacity=0.5,watermark stretch=1.00,arc=3mm,
  watermark graphics=lichtspiel.jpg}

\begin{tcolorbox}[title=Clip (default),clip watermark]
\lipsum[1]
\end{tcolorbox}

\begin{tcolorbox}[title=No clip,clip watermark=false]
\lipsum[1]
\end{tcolorbox}%
```

Clip (default)

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

No clip

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

6.5 Clip Environments

The following clip environments are applicable for all skins which use engines of type `path`, `pathfirst`, `pathmiddle`, or `pathlast`. Especially, the skin enhanced[→P. 99] supports *all* of them and standard[→P. 98] *none.* The typical area of application is inside overlay code, see Section 3.7 from page 41.

\begin{**tcbclipframe**}
  ⟨*environment content*⟩
\end{**tcbclipframe**}
    Defines a `Tikz` scope which clips to the frame area path.

```
\makeatletter
\newtcolorbox{picturebox}[2][]{%
  enhanced,frame hidden,interior hidden,fonttitle=\bfseries,
  overlay={\begin{tcbclipframe}\node at (frame)
    {\includegraphics[width=\tcb@width,height=\tcb@height]{#2}};\end{tcbclipframe}%
    \begin{tcbclipinterior}\fill[white,opacity=0.75]
    (frame.south west) rectangle (frame.north east);\end{tcbclipinterior}},#1}
\makeatother

\begin{picturebox}[title=My Picture Box]{lichtspiel.jpg}
\lipsum[1]
\end{picturebox}
```

My Picture Box

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

\begin{**tcbclipinterior**}
 ⟨*environment content*⟩
\end{**tcbclipinterior**}
> Defines a `Tikz` scope which clips to the interior area path.

```
\begin{tcolorbox}[enhanced,title=My Title,
  overlay={\begin{tcbclipinterior}
    \draw[red,line width=1cm] (interior.north west)--(interior.south east);
    \draw[red,line width=1cm] (interior.south west)--(interior.north east);
  \end{tcbclipinterior}}]
\lipsum[1]
\end{tcolorbox}
```


\begin{**tcbcliptitle**}
 ⟨*environment content*⟩
\end{**tcbcliptitle**}
> Defines a `Tikz` scope which clips to the title area path.

```
\begin{tcolorbox}[enhanced,title=My Title,colframe=blue,colback=yellow!10!white,
  overlay={\begin{tcbcliptitle}\node at (title)
  {\includegraphics[width=\linewidth]{lichtspiel.jpg}};\end{tcbcliptitle}}]
\lipsum[1]
\end{tcolorbox}
```
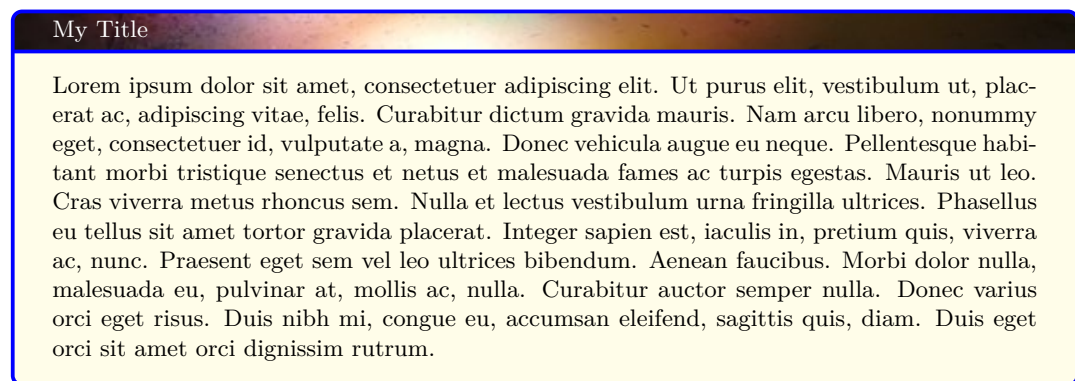
/tcb/clip title=⟨*boolean value*⟩ (default `true`, initially `false`)

Sets the title to be clipped to the title area.

```
\tcbset{enhanced,width=5cm,colframe=red!50!white,coltitle=black}

\begin{tcolorbox}[title=\mbox{This is a title which is unbreakable and far too long}]
This is a tcolorbox.
\end{tcolorbox}

\begin{tcolorbox}[title=\mbox{This is a title which is unbreakable and far too long},
  clip title]
This is a tcolorbox.
\end{tcolorbox}
```

This is a title which is unbreakable and far too long

This is a tcolorbox.

This is a title which is unbreak

This is a tcolorbox.

/tcb/clip upper=⟨*boolean value*⟩ (default `true`, initially `false`)

Sets the upper part to be clipped to the interior area.

```
\newcommand{\mygraphics}[2][]{%
  \tcbox[enhanced,boxsep=0pt,top=0pt,bottom=0pt,left=0pt,
    right=0pt,boxrule=0.4pt,drop fuzzy shadow,clip upper,
    colback=black!75!white,toptitle=2pt,bottomtitle=2pt,nobeforeafter,
    center title,fonttitle=\small\sffamily,title=\detokenize{#2}]
  {\includegraphics[width=\the\dimexpr(\linewidth-4mm)/2\relax]{#2}}}

\mygraphics{lichtspiel.jpg}\hfill
\mygraphics{Basilica_5.png}
```

lichtspiel.jpg

Basilica_5.png
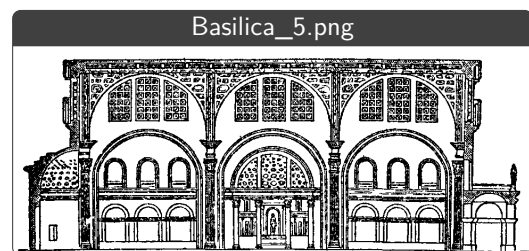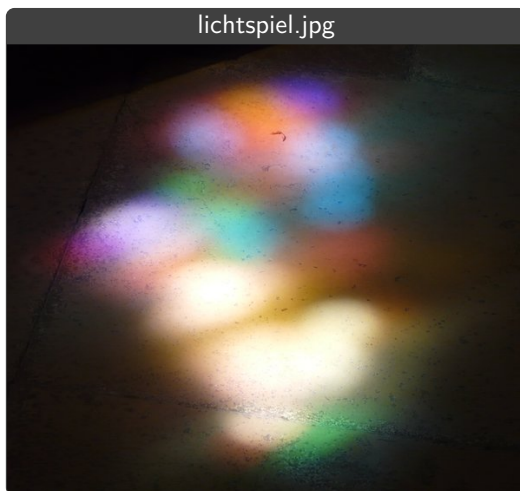
The example for `/tcb/clip upper`^{→P.89} sizes the box according to the dimensions of the picture. To do it the other way around, the watermark options provide an easy solution.

```
\newcommand{\mygraphics}[2][]{%
  \tcbox[enhanced,capture=minipage,boxsep=0pt,top=0pt,bottom=0pt,left=0pt,
    right=0pt,boxrule=0.4pt,drop fuzzy shadow,nobeforeafter,
    colback=black!75!white,toptitle=2pt,bottomtitle=2pt,
    center title,fonttitle=\small\sffamily,title=\detokenize{#2},
    width=(\linewidth-4mm)/2,height=6cm,colbacktitle={black},
    watermark zoom=1.0,watermark graphics={#2}]{}}

\mygraphics{lichtspiel.jpg}\hfill
\mygraphics{Basilica_5.png}
```
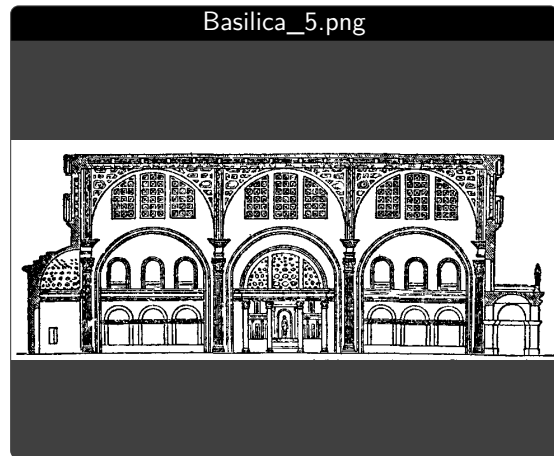

/tcb/clip lower=⟨*boolean value*⟩   (default `true`, initially `false`)

Sets the lower part to be clipped to the interior area.

```
\tcbset{enhanced,width=5cm,colframe=red!50!black,text and listing}

\begin{tcblisting}{}
Donau\-dampf\-schiff\-fahrts\-ka\-pi\-t\"ans\-m\"ut\-zen\-fran\-sen
\end{tcblisting}

\begin{tcblisting}{clip lower}
Donau\-dampf\-schiff\-fahrts\-ka\-pi\-t\"ans\-m\"ut\-zen\-fran\-sen
\end{tcblisting}
```

6.6 Border Line Option Keys

The following border line options are applicable for most skins which use `tikzpicture` as
`/tcb/graphical environment`→P. 72. Therefore, the skin `standard`→P. 98 does not support these
border lines, but most other skins, e.g. `enhanced`→P. 99.

The border lines are independent from the normal `tcolorbox` rules. They may be used with or
without the `/tcb/segmentation engine`→P. 73.

The border lines are stackable, i.e. several different border lines can be used on the same
`tcolorbox`. They are drawn *after* the box frame and box interior and *before* overlays or water-
marks.

> Technically, the normal `tcolorbox` rules result from a TikZ *filling* process. The border lines are
> created by a TikZ *drawing* process. This can be used to apply different effects.

/tcb/borderline={⟨*width*⟩}{⟨*offset*⟩}{⟨*options*⟩}   (no default, initially unset)
    Adds a new border line to the stack of border lines. This border line is drawn with the
    given ⟨*width*⟩ and gets a ⟨*offset*⟩ computed from the frame outline. A positive ⟨*offset*⟩ value
    moves the borderline inside the `tcolorbox` and a negative ⟨*offset*⟩ value moves it outside
    without changing the bounding box.
    The border line is drawn along a TikZ path with the given TikZ ⟨*options*⟩. Note that the
    TikZ `line width` option should not be used here.
    The border lines adapt to the rounded corners of the `tcolorbox`. An inside border line will
    switch to sharp corners if necessary, an outside border line will always be rounded if not
    set to `sharp corners`.

```
\begin{tcolorbox}[enhanced,title=Rounded corners,fonttitle=\bfseries,boxsep=5pt,
  arc=8pt,
  borderline={0.5pt}{0pt}{red},
  borderline={0.5pt}{5pt}{blue,dotted},
  borderline={0.5pt}{-5pt}{green,dashed} ]
This is a tcolorbox.
\end{tcolorbox}
\bigskip
\begin{tcolorbox}[enhanced,title=Sharp corners,fonttitle=\bfseries,boxsep=5pt,
  arc=0pt,outer arc=0pt,
  borderline={0.5pt}{0pt}{red},
  borderline={0.5pt}{5pt}{blue,dotted},
  borderline={0.5pt}{-5pt}{green,dashed,sharp corners} ]
This is a tcolorbox.
\end{tcolorbox}
```

Rounded corners

This is a tcolorbox.

Sharp corners

This is a tcolorbox.

```
% \usepackage{lipsum}
\begin{tcolorbox}[enhanced,arc=3mm,boxrule=1.5mm,boxsep=1.5mm,
  colback=yellow!20!white,
  colframe=blue,
  borderline={1mm}{1mm}{white},
  borderline={1mm}{2mm}{red} ]
  \lipsum[1]
\end{tcolorbox}
```

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

```
% \usepackage{lipsum}
\begin{tcolorbox}[enhanced,arc=3mm,boxrule=1.5mm,
  frame hidden,colback=blue!10!white,
  borderline={1mm}{0mm}{blue,dotted} ]
  \lipsum[2]
\end{tcolorbox}
```

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.

```
% \usepackage{lipsum}
\begin{tcolorbox}[enhanced,skin=enhancedmiddle,
  frame hidden,interior hidden,top=0mm,bottom=0mm,boxsep=0mm,
  borderline={0.75mm}{0mm}{red},
  borderline={0.75mm}{0.75mm}{red!50!yellow},
  borderline={0.75mm}{1.5mm}{yellow}, ]
  \lipsum[3]
\end{tcolorbox}
```

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.

```
% \usepackage{lipsum}
\newtcolorbox{mygreenbox}[2][]{%
  enhanced,width=\linewidth-6pt,
  enlarge top by=3pt,enlarge bottom by=3pt,
  enlarge left by=3pt,enlarge right by=3pt,
  title={#2},frame hidden,boxrule=0pt,top=1mm,bottom=1mm,
  colframe=green!30!black, colbacktitle=green!50!yellow,
  coltitle=black, colback=green!25!white,
  borderline={0.5pt}{-0.5pt}{green!75!blue},
  borderline={1pt}{-3pt}{green!50!blue},#1}

\begin{mygreenbox}{My title}
  \lipsum[4]
\end{mygreenbox}
```

My title

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer 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.

/tcb/no borderline   (no default, initially set)
    Removes all border lines if set before.

6.7 Shadow Option Keys

The following shadow options are applicable for most skins which use `tikzpicture` as
/tcb/graphical environment→P. 72. Therefore, the skin standard→P. 98 does not support these
shadows, but most other skins, e. g. enhanced→P. 99.

The shadows are stackable, i. e. several different shadows can be used on the same `tcolorbox`.
They are drawn *before* the box frame is drawn.

/tcb/shadow={⟨*xshift*⟩}{⟨*yshift*⟩}{⟨*offset*⟩}{⟨*options*⟩}   (no default)
Adds a new shadow to the stack of shadows. This shadow is follows the outline of the
`tcolorbox` but is shifted by ⟨*xshift*⟩ and ⟨*yshift*⟩. The ⟨*offset*⟩ value is a distance value
from the frame outline. A positive ⟨*offset*⟩ value shrinks the shadow and a negative ⟨*offset*⟩
value enlarges the shadow. The shadow is filled along a TikZ path with the given TikZ
⟨*options*⟩.
The shadows adapt to the rounded corners of the `tcolorbox`. An shrinked shadow will
switch to sharp corners if necessary, an enlarged shadow may become more rounded de-
pending on several factors.

Shadows are not considered for the bounding box computation by default. Large shadows
may be overlaped by the following content. But, the bounding box can be adapted if
necessary.

```
\tcbset{enhanced,colback=red!5!white,
  colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[title=My own shadow,
  shadow={2mm}{-1mm}{0mm}{black!50!white}]
This is a tcolorbox.
\end{tcolorbox}
\par\bigskip
\begin{tcolorbox}[title=Another shadow,
  shadow={-1mm}{-2mm}{0mm}{fill=blue,
    opacity=0.5}]
This is a tcolorbox.
\end{tcolorbox}
\par\bigskip
\begin{tcolorbox}[title=Double shadow,
  shadow={-1.5mm}{-1.5mm}{0mm}{fill=blue,
    opacity=0.25},
  shadow={1.5mm}{-1.5mm}{0mm}{fill=red,
    opacity=0.25}]
This is a tcolorbox.
\end{tcolorbox}
\par\bigskip
\begin{tcolorbox}[title=Far shadow,
  shadow={5.5mm}{-3.5mm}{2mm}{fill=black,
    opacity=0.25}]
This is a tcolorbox.
\end{tcolorbox}
\par\bigskip\bigskip
\begin{tcolorbox}[title=Halo shadow,
  shadow={0mm}{0mm}{-1.5mm}%
    {fill=yellow!75!red,opacity=0.5}]
This is a tcolorbox.
\end{tcolorbox}
```

My own shadow

This is a tcolorbox.

Another shadow

This is a tcolorbox.

Double shadow

This is a tcolorbox.

Far shadow

This is a tcolorbox.

Halo shadow

This is a tcolorbox.

/tcb/fuzzy shadow={⟨*xshift*⟩}{⟨*yshift*⟩}{⟨*offset*⟩}{⟨*step*⟩}{⟨*options*⟩}   (no default)

Adds a new fuzzy shadow to the stack of shadows. Actually, this option adds seversal shadows which appear like a shadow with a fuzzy border. This fuzzy shadow is follows the outline of the `tcolorbox` but is shifted by ⟨*xshift*⟩ and ⟨*yshift*⟩. The ⟨*offset*⟩ value is a distance value from the frame outline. A positive ⟨*offset*⟩ value shrinks the shadow and a negative ⟨*offset*⟩ value enlarges the shadow. The {⟨*step*⟩} value describes a shrink offset used for the combination of the partial shadows. The shadow is filled along a TikZ path with the given TikZ ⟨*options*⟩ but any `opacity` value will be ignored.

```
\tcbset{enhanced,colback=red!5!white,
  colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[title=My own shadow,
  fuzzy shadow={2mm}{-1mm}{0mm}{0.1mm}%
              {black!50!white}]
This is a tcolorbox.
\end{tcolorbox}
\par\bigskip
\begin{tcolorbox}[title=Another shadow,
  fuzzy shadow={-1mm}{-2mm}{0mm}{0.2mm}%
              {fill=blue}]
This is a tcolorbox.
\end{tcolorbox}
\par\bigskip
\begin{tcolorbox}[title=Double shadow,
  fuzzy shadow={-1.5mm}{-1.5mm}{0mm}{0.1mm}%
              {blue},
  fuzzy shadow={1.5mm}{-1.5mm}{0mm}{0.1mm}%
              {red}]
This is a tcolorbox.
\end{tcolorbox}
\par\bigskip
\begin{tcolorbox}[title=Far shadow,
  fuzzy shadow={5.5mm}{-3.5mm}{0mm}{0.3mm}%
              {black}]
This is a tcolorbox.
\end{tcolorbox}
\par\bigskip\bigskip
\begin{tcolorbox}[title=Glow shadow,
  fuzzy shadow={0mm}{0mm}{-1.5mm}{0.15mm}%
              {yellow!75!red}]
This is a tcolorbox.
\end{tcolorbox}
```

My own shadow

This is a tcolorbox.

Another shadow

This is a tcolorbox.

Double shadow

This is a tcolorbox.

Far shadow

This is a tcolorbox.

Glow shadow

This is a tcolorbox.

```
\newtcolorbox{mybox}[1][]{enhanced,
  fuzzy shadow={1.0mm}{-1.0mm}{0.12mm}{0mm}{blue!50!white},
  fuzzy shadow={-1.0mm}{-1.0mm}{0.12mm}{0mm}{red!50!white},
  fuzzy shadow={-1.0mm}{1.0mm}{0.12mm}{0mm}{green!50!white},
  fuzzy shadow={1.0mm}{1.0mm}{0.12mm}{0mm}{yellow!50!white},#1
}

\begin{mybox}[title=A multi shadow box]
This is a tcolorbox.
\end{mybox}
```

A multi shadow box

This is a tcolorbox.

/tcb/drop shadow=⟨*color*⟩   (style, default `black!50!white`)

Adds a new shadow with standard dimensions to the stack of shadows. Optionally, the ⟨*color*⟩ for the shadow can be changed.

```
\tcbset{enhanced,colback=red!5!white,
  colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[drop shadow]
This is a tcolorbox.
\end{tcolorbox}\par\bigskip
\begin{tcolorbox}[title=Another shadow,
  drop shadow=blue]
This is a tcolorbox.
\end{tcolorbox}
```

> This is a tcolorbox.

> **Another shadow**
> This is a tcolorbox.

/tcb/drop fuzzy shadow=⟨*color*⟩   (style, default `black!50!white`)

Adds a new fuzzy shadow with standard dimensions to the stack of shadows. Optionally, the ⟨*color*⟩ for the shadow can be changed.

```
\tcbset{enhanced,colback=red!5!white,
  colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[drop fuzzy shadow]
This is a tcolorbox.
\end{tcolorbox}\par\bigskip
\begin{tcolorbox}[title=Another shadow,
  drop fuzzy shadow=blue]
This is a tcolorbox.
\end{tcolorbox}
```

> This is a tcolorbox.

> **Another shadow**
> This is a tcolorbox.

/tcb/drop midday shadow=⟨*color*⟩   (style, default `black!50!white`)

Adds a new shadow with standard dimensions to the stack of shadows. Optionally, the ⟨*color*⟩ for the shadow can be changed.

```
\tcbset{enhanced,colback=red!5!white,
  colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[drop midday shadow]
This is a tcolorbox.
\end{tcolorbox}\par\bigskip
\begin{tcolorbox}[title=Another shadow,
  drop midday shadow=blue]
This is a tcolorbox.
\end{tcolorbox}
```

> This is a tcolorbox.

> **Another shadow**
> This is a tcolorbox.

/tcb/drop fuzzy midday shadow=⟨*color*⟩   (style, default `black!50!white`)

Adds a new fuzzy shadow with standard dimensions to the stack of shadows. Optionally, the ⟨*color*⟩ for the shadow can be changed.

```
\tcbset{enhanced,colback=red!5!white,
  colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[drop fuzzy midday shadow]
This is a tcolorbox.
\end{tcolorbox}\par\bigskip
\begin{tcolorbox}[title=Another shadow,
  drop fuzzy midday shadow=blue]
This is a tcolorbox.
\end{tcolorbox}
```

> This is a tcolorbox.

> **Another shadow**
> This is a tcolorbox.

/tcb/halo=⟨*size*⟩ `with` ⟨*color*⟩ (style, default `0.9mm with yellow`)

Adds a new halo shadow with the given ⟨*color*⟩ which overlaps the colorbox an all sides by ⟨*size*⟩.

```
\tcbset{enhanced,colback=red!5!white,
  colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[title=My own halo,halo]
This is a tcolorbox.
\end{tcolorbox}
\par\bigskip\bigskip
\begin{tcolorbox}[title=Another halo,
  halo=2mm with green]
This is a tcolorbox.
\end{tcolorbox}
```

> **My own halo**
>
> This is a tcolorbox.

> **Another halo**
>
> This is a tcolorbox.

/tcb/fuzzy halo=⟨*size*⟩ `with` ⟨*color*⟩ (style, default `0.9mm with yellow`)

Adds a new fuzzy halo shadow with the given ⟨*color*⟩ which overlaps the colorbox an all sides by ⟨*size*⟩ plus `0.48mm`.

```
\tcbset{enhanced,colback=red!5!white,
  colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[title=My own halo,fuzzy halo]
This is a tcolorbox.
\end{tcolorbox}
\par\bigskip\bigskip
\begin{tcolorbox}[title=Another halo,
  fuzzy halo=2mm with green]
This is a tcolorbox.
\end{tcolorbox}
```

> **My own halo**
>
> This is a tcolorbox.

> **Another halo**
>
> This is a tcolorbox.

```
\begin{tcolorbox}[blank,
  fuzzy halo=2mm with red!50!white,
  fuzzy halo=1mm with white,]
\lipsum[1]
\end{tcolorbox}
```

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

/tcb/no shadow (no default)

Removes all shadows if set before.

6.8 Skin 'standard'

Note that the option keys `/tcb/frame style`[→P.74], `/tcb/interior style`[→P.74], `/tcb/segmentation style`[→P.75], and `/tcb/title style`[→P.76] are not be applicable to the standard skin. Also, watermarks (see Subsection 6.4) are not usable with the standard skin.

/tcb/skin=`standard` (skin)

This is the standard skin from the core package. All drawing engines are set to type `standard`. The drawing is based on `pgf` commands and does not need the `tikz` package.

> **Environment and engines for the skin 'standard'**
>
> `/tcb/graphical environment`[→P.72]: `pgfpicture`
> `/tcb/frame engine`[→P.72]: `standard`
> `/tcb/interior titled engine`[→P.72]: `standard`
> `/tcb/interior engine`[→P.72]: `standard`
> `/tcb/segmentation engine`[→P.73]: `standard`
> `/tcb/title engine`[→P.73]: `standard`

/tcb/standard (style, no value)

This is an abbreviation for setting `skin=standard`.

```
\tcbset{standard,equal height group=standard,
  colback=LightGreen,colframe=DarkGreen,colbacklower=LimeGreen!75!LightGreen,
  width=(\linewidth-6mm)/4,nobeforeafter,
  left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm}
%
\begin{tcolorbox}
  This is my content.
\end{tcolorbox}\hfill
\begin{tcolorbox}
  This is my content.
  \tcblower
  More content.
\end{tcolorbox}\hfill
\begin{tcolorbox}[adjusted title=My title]
  This is my content.
\end{tcolorbox}\hfill
\begin{tcolorbox}[adjusted title=My title]
  This is my content.
  \tcblower
  More content.
\end{tcolorbox}
```

This is my content.	This is my content. ------ More content.	**My title** This is my content.	**My title** This is my content. ------ More content.

6.9 Skin Family 'enhanced'

If you like the standard appearance of a `tcolorbox` but you want to have some 'enhanced' features, the `enhanced` skin is what you are looking for.

/tcb/skin=enhanced   (skin)

This skin translates the drawing commands of the core package into `tikz` path commands. Therefore, it allows all `tikz` high level options for these paths and has more flexibility compared to the standard$^{→P.98}$ skin. You pay for this with some prolonged compilation time. The `tikz` path options can be given with the option keys /tcb/frame style$^{→P.74}$, /tcb/interior style$^{→P.74}$, /tcb/segmentation style$^{→P.75}$, and /tcb/title style$^{→P.76}$.

> **Environment and engines for the skin 'enhanced'**
>
> /tcb/graphical environment$^{→P.72}$:   `tikzpicture`
> /tcb/frame engine$^{→P.72}$:   `path`
> /tcb/interior titled engine$^{→P.72}$: `path`
> /tcb/interior engine$^{→P.72}$:   `path`
> /tcb/segmentation engine$^{→P.73}$:   `path`
> /tcb/title engine$^{→P.73}$:   `path`

/tcb/enhanced   (style, no value)

This is an abbreviation for setting `skin=enhanced`.

```
\tcbset{enhanced,equal height group=enhanced,
  colback=LightGreen,colframe=DarkGreen,colbacklower=LimeGreen!75!LightGreen,
  width=(\linewidth-6mm)/4,nobeforeafter,
  left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm}
%
\begin{tcolorbox}
  This is my content.
\end{tcolorbox}\hfill
\begin{tcolorbox}
  This is my content.
  \tcblower
  More content.
\end{tcolorbox}\hfill
\begin{tcolorbox}[adjusted title=My title]
  This is my content.
\end{tcolorbox}\hfill
\begin{tcolorbox}[adjusted title=My title]
  This is my content.
  \tcblower
  More content.
\end{tcolorbox}
```

```
% \usetikzlibrary{shadings}          % preamble
\tcbset{skin=enhanced,fonttitle=\bfseries,
  frame style={upper left=blue,upper right=red,lower left=yellow,lower right=green},
  interior style={white,opacity=0.5},
  segmentation style={black,solid,opacity=0.2,line width=1pt}}

\begin{tcolorbox}[title=Nice box in rainbow colors]
  With the 'enhanced' skin, it is quite easy to produce fancy looking effects.
  \tcblower
  Note that this is still a \texttt{tcolorbox}.
\end{tcolorbox}
```

Nice box in rainbow colors

With the 'enhanced' skin, it is quite easy to produce fancy looking effects.

Note that this is still a `tcolorbox`.

```
% \usetikzlibrary{decorations.pathmorphing} % preamble
\tcbset{skin=enhanced,fonttitle=\bfseries,boxrule=1mm,
  frame style={draw=FireBrick,fill=Salmon},drop fuzzy shadow,
  interior style={draw=FireBrick,top color=Salmon!10,bottom color=Salmon!20},
  segmentation style={draw=FireBrick,solid,decorate,
        decoration={coil,aspect=0,segment length=10.1mm}}}

\begin{tcblisting}{title=A listing box with shadow and some specials}
Of course, skins can be used for listings also.
\begin{equation}
  \int\limits_1^2 \frac{1}{x}~dx = \ln(2).
\end{equation}
\end{tcblisting}
```

A listing box with shadow and some specials

```
Of course, skins can be used for listings also.
\begin{equation}
  \int\limits_1^2 \frac{1}{x}~dx = \ln(2).
\end{equation}
```

Of course, skins can be used for listings also.

$$\int\limits_1^2 \frac{1}{x}\, dx = \ln(2). \tag{2}$$

/tcb/enhanced standard (style, no value)

For unbreakable boxes, this is identical to using /tcb/enhanced^{→P.99}. But, for breakable boxes, the *break sequence* is identical to the standard^{→P.98} skin, see Section 9.5 from page 175.

/tcb/blank (style, initially unset)

This style relies on the skin `enhanced`[→P.99]. All drawing operations are disabled and all margins are set to `0pt`.

```
\begin{tcolorbox}[blank,watermark text=A blank box]
\lipsum[1]
\end{tcolorbox}
```

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

```
% \tcbuselibrary{fitting}
\newtcboxfit{\mybox}[1]{blank,width=4cm,height=7cm,top=4pt,
  watermark text=#1}

\begin{tabular}{|c|c|c|}\hline
A & B & C\\\hline
\mybox{A}{\lipsum[1]} & \mybox{B}{\lipsum[2]} & \mybox{C}{\lipsum[3]}\\\hline
\end{tabular}
```

A	B	C
Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.	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.

\tcbline

Sometimes, a line is only a line. With `\tcblower`^{→P.7} you separate the box content into two functional units. `\tcbline` draws only a line which looks like the segmentation line between upper and lower part. Furthermore, you can use `\tcbline` more than just once. `\tcbline` always uses the `path` drawing engine. Therefore, the `/tcb/segmentation style`^{→P.75} can be applied.

```
\tcbset{enhanced,colframe=blue!50!black,colback=white}

\begin{tcolorbox}[colupper=red!50!black,collower=green!50!black]
\lipsum[1]
\tcbline
\lipsum[2]
\tcblower
\lipsum[3]
\tcbline
\lipsum[4]
\end{tcolorbox}
```

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

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, consectetuer 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.

/tcb/skin=**enhancedfirst** (skin)

This is a flavor of enhanced^{→P.99} which is used as a *first* part in a break sequence for enhanced^{→P.99}. Nevertheless, this skin can be applied independently.

Environment and engines for the skin 'enhancedfirst'

/tcb/graphical environment^{→P.72}: tikzpicture
/tcb/frame engine^{→P.72}:   pathfirst
/tcb/interior titled engine^{→P.72}: pathfirst
/tcb/interior engine^{→P.72}:   pathfirst
/tcb/segmentation engine^{→P.73}: path
/tcb/title engine^{→P.73}:   pathfirst

```
\tcbset{skin=enhancedfirst,equal height group=enhancedfirst,
  colback=LightGreen,colframe=DarkGreen,
  width=(\linewidth-6mm)/4,nobeforeafter,
  left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm}
%
\begin{tcolorbox}
  This is my content.
\end{tcolorbox}\hfill
\begin{tcolorbox}
  This is my content.
  \tcblower
  More content.
\end{tcolorbox}\hfill
\begin{tcolorbox}[adjusted title=My title]
  This is my content.
\end{tcolorbox}\hfill
\begin{tcolorbox}[adjusted title=My title]
  This is my content.
  \tcblower
  More content.
\end{tcolorbox}
```

This is my content.	This is my content.	My title	My title
	More content.	This is my content.	This is my content.
			More content.

/tcb/skin=enhancedmiddle (skin)

This is a flavor of enhanced[→P.99] which is used as a *middle* part in a break sequence for enhanced[→P.99]. Nevertheless, this skin can be applied independently.

> **Environment and engines for the skin 'enhancedmiddle'**
>
> /tcb/graphical environment[→P.72]: tikzpicture
> /tcb/frame engine[→P.72]: pathmiddle
> /tcb/interior titled engine[→P.72]: pathmiddle
> /tcb/interior engine[→P.72]: pathmiddle
> /tcb/segmentation engine[→P.73]: path
> /tcb/title engine[→P.73]: pathmiddle

```
\tcbset{skin=enhancedmiddle,equal height group=enhancedmiddle,
  colback=LightGreen,colframe=DarkGreen,
  width=(\linewidth-6mm)/4,nobeforeafter,
  left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm}
%
\begin{tcolorbox}
  This is my content.
\end{tcolorbox}\hfill
\begin{tcolorbox}
  This is my content.
  \tcblower
  More content.
\end{tcolorbox}\hfill
\begin{tcolorbox}[adjusted title=My title]
  This is my content.
\end{tcolorbox}\hfill
\begin{tcolorbox}[adjusted title=My title]
  This is my content.
  \tcblower
  More content.
\end{tcolorbox}
```

This is my content.	This is my content. ───────── More content.	**My title** This is my content.	**My title** This is my content. ───────── More content.

/tcb/marker (style, no value)

This styles relies on the skin enhancedmiddle. It is intended to be used as an optical marker like a highlighter pen.

```
\begin{tcolorbox}[marker]
\lipsum[2]
\end{tcolorbox}
```

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.

This examples demonstrates the creation of several *text marker* environments based on
`enhancedmiddle`^{→P.104}.

```
\tcbset{textmarker/.style={%
    skin=enhancedmiddle,breakable,parbox=false,
    boxrule=0mm,leftrule=5mm,rightrule=5mm,boxsep=0mm,arc=0mm,outer arc=0mm,
    left=3mm,right=3mm,top=1mm,bottom=1mm,enlarge left by=-8mm,enlarge right by=-8mm,
    toptitle=1mm,bottomtitle=1mm,width=\the\dimexpr\linewidth+1.6cm\relax}}

\newtcolorbox{yellow}{textmarker,colback=yellow!5!white,colframe=yellow}
\newtcolorbox{orange}{textmarker,colback=DarkOrange!5!white,
                      colframe=DarkOrange!75!yellow}
\newtcolorbox{red}{textmarker,colback=red!5!white,colframe=red}
\newtcolorbox{blue}{textmarker,colback=DeepSkyBlue!5!white,colframe=DeepSkyBlue}
\newtcolorbox{green}{textmarker,colback=Chartreuse!5!white,colframe=Chartreuse}

\begin{yellow}
  \lipsum[1]
\end{yellow}

\begin{orange}
  \lipsum[2]
\end{orange}

\begin{red}
  \lipsum[3]
\end{red}

\begin{green}
  \lipsum[4]
\end{green}

\begin{blue}
  \lipsum[5]
\end{blue}
\lipsum[6]% following text
```

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

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, consectetuer 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 consectetuer.

Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

/tcb/skin=**enhancedlast** (skin)

This is a flavor of enhanced[→P.99] which is used as a *last* part in a break sequence for enhanced[→P.99]. Nevertheless, this skin can be applied independently.

Environment and engines for the skin 'enhancedlast'

/tcb/graphical environment[→P.72]: tikzpicture
/tcb/frame engine[→P.72]: pathlast
/tcb/interior titled engine[→P.72]: pathlast
/tcb/interior engine[→P.72]: pathlast
/tcb/segmentation engine[→P.73]: path
/tcb/title engine[→P.73]: pathlast

```
\tcbset{skin=enhancedlast,equal height group=enhancedlast,
  colback=LightGreen,colframe=DarkGreen,
  width=(\linewidth-6mm)/4,nobeforeafter,
  left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm}
%
\begin{tcolorbox}
  This is my content.
\end{tcolorbox}\hfill
\begin{tcolorbox}
  This is my content.
  \tcblower
  More content.
\end{tcolorbox}\hfill
\begin{tcolorbox}[adjusted title=My title]
  This is my content.
\end{tcolorbox}\hfill
\begin{tcolorbox}[adjusted title=My title]
  This is my content.
  \tcblower
  More content.
\end{tcolorbox}
```

This is my content.	This is my content. More content.	My title This is my content.	My title This is my content. More content.

6.10 Skin Family 'freelance'

This skin family is intended for the experienced user who looks for special customizations of a `tcolorbox`. It gives full freedom for the appearance, but it requires some `tikz` coding. For small adaptions, you may want to use overlays instead, see Section 3.7 on page 41.

/tcb/skin=**freelance**   (skin)

This skin gives full freedom for the appearance of the `tcolorbox`. All drawing engines are set to type `freelance`; they use the `tikz` package and compute the `/tcb/geometry nodes`→P.73. This skin is useful for boxes which should differ much from the normal appearance. Note that this difference has to be programmed by the user. The drawing code can be given with the following option keys. As default value, the code from the `standard` skin is set.

> **Environment and engines for the skin 'freelance'**
>
> /tcb/graphical environment→P.72: tikzpicture
> /tcb/frame engine→P.72:   freelance
> /tcb/interior titled engine→P.72: freelance
> /tcb/interior engine→P.72:   freelance
> /tcb/segmentation engine→P.73:   freelance
> /tcb/title engine→P.73:   freelance

/tcb/freelance   (style, no value)

This is an abbreviation for setting `skin=freelance`.

```
\tcbset{freelance,equal height group=freelance,
  colback=LightGreen,colframe=DarkGreen,
  width=(\linewidth-6mm)/4,nobeforeafter,
  left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm}
%
\begin{tcolorbox}
  This is my content.
\end{tcolorbox}\hfill
\begin{tcolorbox}
  This is my content.
  \tcblower
  More content.
\end{tcolorbox}\hfill
\begin{tcolorbox}[adjusted title=My title]
  This is my content.
\end{tcolorbox}\hfill
\begin{tcolorbox}[adjusted title=My title]
  This is my content.
  \tcblower
  More content.
\end{tcolorbox}
```

This is my content.	This is my content. More content.	My title This is my content.	My title This is my content. More content.

```
    \tcbset{skin=freelance,boxrule=2mm,enlarge top by=2mm,enlarge bottom by=2mm,
      enlarge left by=3mm,enlarge right by=3mm,width=\linewidth-6mm,
    frame code={\path[top color=FireBrick,bottom color=FireBrick,middle color=FireBrick!50,
      draw=FireBrick!75!black,double=Gold,rounded corners=1mm]
      (frame.south west) -- ([xshift=-3mm]frame.west) -- (frame.north west)
      -- ([yshift=2mm]frame.north) -- (frame.north east) -- ([xshift=3mm]frame.east)
      -- (frame.south east) -- ([yshift=-2mm]frame.south) -- cycle;},
    interior titled code={\path[outer color=Gold,inner color=white,draw=Gold,
      double=FireBrick!75!black,rounded corners=5mm]
      (interior.south west) rectangle (interior.north east);},
    segmentation code={\path[draw=FireBrick,opacity=0.25] ([xshift=2cm]segmentation.west)
      -- (segmentation.north) -- ([xshift=-2cm]segmentation.east)
      -- (segmentation.south) -- cycle;}}

  \begin{tcolorbox}[title=My title]
    This is the upper part.
    \tcblower
    This is the lower part.
  \end{tcolorbox}
```

My title

This is the upper part.

This is the lower part.
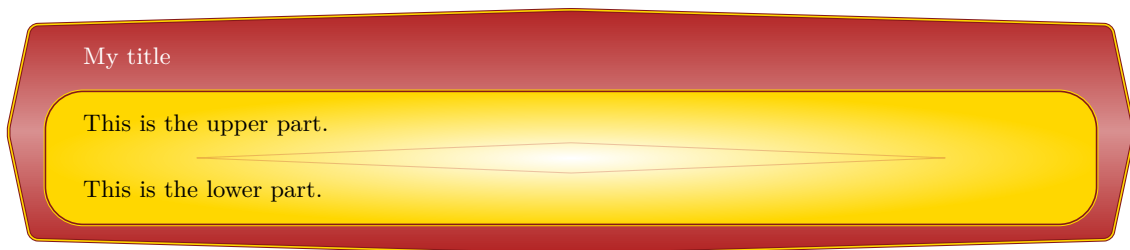
/tcb/skin=freelancefirst   (skin)
This skin equals `freelance`^{→P.108} with exception of the break sequence, see Section 9.5 on
page 175. It is used as first part of the break sequence of `freelance`^{→P.108}. `/tcb/extend`
`freelancefirst` can be used to customize this part.

/tcb/skin=freelancemiddle   (skin)
This skin equals `freelance`^{→P.108} with exception of the break sequence, see Section 9.5 on
page 175. It is used as middle part of the break sequence of `freelance`^{→P.108}. `/tcb/extend`
`freelancemiddle` can be used to customize this part.

/tcb/skin=freelancelast   (skin)
This skin equals `freelance`^{→P.108} with exception of the break sequence, see Section 9.5 on
page 175. It is used as last part of the break sequence of `freelance`^{→P.108}. `/tcb/extend`
`freelancelast` can be used to customize this part.

/tcb/extend freelance=⟨*graphical code*⟩   (no default, initially empty)
The ⟨*graphical code*⟩ is added to the skin definition of `freelance`^{→P.108}.

/tcb/extend freelancefirst=⟨*graphical code*⟩   (no default, initially empty)
The ⟨*graphical code*⟩ is added to the skin definition of `freelancefirst` which is used as
first part of the break sequence of `freelance`^{→P.108}. Refer to the following example for
usage.

/tcb/extend freelancemiddle=⟨*graphical code*⟩   (no default, initially empty)
The ⟨*graphical code*⟩ is added to the skin definition of `freelancemiddle` which is used as
middle part of the break sequence of `freelance`^{→P.108}. Refer to the following example for
usage.

/tcb/extend freelancelast=⟨*graphical code*⟩   (no default, initially empty)
The ⟨*graphical code*⟩ is added to the skin definition of `freelancelast` which is used as last
part of the break sequence of `freelance`^{→P.108}. Refer to the following example for usage.

This example demonstrates a breakable freelance box. Here, we define an environment `freebox`. The first application of `freebox` produces an unbroken `tcolorbox`. The box is drawn by the code given by /tcb/frame code^{→P.77} and /tcb/interior code^{→P.78}. The second application of `freebox` is broken into several parts which are drawn by the codes given by /tcb/extend freelancefirst, /tcb/extend freelancemiddle, and /tcb/extend freelancelast.

```
% Preamble:
%\usepackage{tikz,lipsum}
%\tcbuselibrary{skins,breakable}
\tikzset{colframe/.style={fill,top color=red!75!black,bottom color=red!75!black,
    middle color=red},
  colint/.style={fill=yellow!50!white}, coltria/.style={fill=red!15!white}}

\newtcolorbox{freebox}[1][]{freelance,breakable,leftrule=5mm,left=2mm,
  watermark color=red!50!yellow!75!white,
  watermark text on=unbroken is unbroken box,
  watermark text on=first is first part,
  watermark text on=middle is middle part,
  watermark text on=last is last part,
  % code for unbroken boxes:
  frame code={\path[colframe] (frame.south west)--(frame.north west)
    --([xshift=-5mm]frame.north east)--([yshift=-5mm]frame.north east)
    --([yshift=5mm]frame.south east)--([xshift=-5mm]frame.south east)--cycle; },
  interior code={\path[colint] (interior.south west)--(interior.north west)
    --([xshift=-4.8mm]interior.north east)--([yshift=-4.8mm]interior.north east)
    --([yshift=4.8mm]interior.south east)--([xshift=-4.8mm]interior.south east)
    --cycle; },
  % code for the first part of a break sequence:
  extend freelancefirst={%
    frame code={\path[colframe] (frame.south west)--(frame.north west)
      --([xshift=-5mm]frame.north east)--([yshift=-5mm]frame.north east)
      --(frame.south east)--cycle;
      \path[coltria] ([xshift=2.5mm,yshift=1mm]frame.south west) -- +(120:2mm)
      -- +(60:2mm)-- cycle; },
    interior code={\path[colint] (interior.south west|-frame.south)
      --(interior.north west)--([xshift=-4.8mm]interior.north east)
      --([yshift=-4.8mm]interior.north east)--(interior.south east|-frame.south)
      --cycle; },
  },%
  % code for the middle part of a break sequence:
  extend freelancemiddle={%
    frame code={\path[colframe] (frame.south west)--(frame.north west)
      --(frame.north east)--(frame.south east)--cycle;
      \path[coltria] ([xshift=2.5mm,yshift=-1mm]frame.north west) -- +(240:2mm)
        -- +(300:2mm) -- cycle;
      \path[coltria] ([xshift=2.5mm,yshift=1mm]frame.south west) -- +(120:2mm)
        -- +(60:2mm) -- cycle;
      },
    interior code={\path[colint] (interior.south west|-frame.south)
      --(interior.north west|-frame.north)--(interior.north east|-frame.north)
      --(interior.south east|-frame.south)--cycle; },
  },
  % code for the last part of a break sequence:
  extend freelancelast={%
    frame code={\path[colframe] (frame.south west)--(frame.north west)
      --(frame.north east)--([yshift=5mm]frame.south east)
      --([xshift=-5mm]frame.south east)--cycle;
      \path[coltria] ([xshift=2.5mm,yshift=-1mm]frame.north west) -- +(240:2mm)
      -- +(300:2mm) -- cycle;
      },
```

```
    interior code={\path[colint] (interior.south west)
      --(interior.north west|-frame.north)--(interior.north east|-frame.north)
      --([yshift=4.8mm]interior.south east)--([xshift=-4.8mm]interior.south east)
      --cycle; },
    },
  #1}

\begin{freebox}
\lipsum[1]
\end{freebox}

\begin{freebox}
\lipsum[1-12]
\end{freebox}
```

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.
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, consectetuer 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 consectetuer.

Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

Sed commodo posuere pede. Mauris ut est. Ut quis purus. Sed ac odio. Sed vehicula hendrerit sem. Duis non odio. Morbi ut dui. Sed accumsan risus eget odio. In hac habitasse platea dictumst. Pellentesque non elit. Fusce sed justo eu urna porta tincidunt. Mauris felis odio, sollicitudin sed, volutpat a, ornare ac, erat. Morbi quis dolor. Donec pellentesque, erat ac sagittis semper, nunc dui lobortis purus, quis congue purus metus ultricies tellus. Proin et quam. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent sapien turpis, fermentum vel, eleifend faucibus, vehicula eu, lacus.

Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Donec odio elit, dictum in, hendrerit sit amet, egestas sed, leo. Praesent feugiat sapien aliquet odio. Integer vitae justo. Aliquam vestibulum fringilla lorem. Sed neque lectus, consectetuer at, consectetuer sed, eleifend ac, lectus. Nulla facilisi. Pellentesque eget lectus. Proin eu metus. Sed porttitor. In hac habitasse platea dictumst. Suspendisse eu lectus. Ut mi mi, lacinia sit amet, placerat et, mollis vitae, dui. Sed ante tellus, tristique ut, iaculis eu, malesuada ac, dui. Mauris nibh leo, facilisis non, adipiscing quis, ultrices a, dui.

Morbi luctus, wisi viverra faucibus pretium, nibh est placerat odio, nec commodo wisi enim eget quam. Quisque libero justo, consectetuer a, feugiat vitae, porttitor eu, libero. Suspendisse sed mauris vitae elit sollicitudin malesuada. Maecenas ultricies eros sit amet ante. Ut venenatis velit. Maecenas sed mi eget dui varius euismod. Phasellus aliquet volutpat odio. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Pellentesque sit amet pede ac sem eleifend consectetuer. Nullam elementum, urna vel imperdiet sodales, elit ipsum pharetra ligula, ac pretium ante justo a nulla. Curabitur tristique arcu eu metus. Vestibulum lectus. Proin mauris. Proin eu nunc eu urna hendrerit faucibus. Aliquam auctor, pede consequat laoreet varius, eros tellus scelerisque quam, pellentesque hendrerit ipsum dolor sed augue. Nulla nec lacus. Suspendisse vitae elit. Aliquam arcu neque, ornare in, ullamcorper quis, commodo eu,

libero. Fusce sagittis erat at erat tristique mollis. Maecenas sapien libero, molestie et, lobortis in, sodales eget, dui. Morbi ultrices rutrum lorem. Nam elementum ullamcorper leo. Morbi dui. Aliquam sagittis. Nunc placerat. Pellentesque tristique sodales est. Maecenas imperdiet lacinia velit. Cras non urna. Morbi eros pede, suscipit ac, varius vel, egestas non, eros. Praesent malesuada, diam id pretium elementum, eros sem dictum tortor, vel consectetuer odio sem sed wisi.

Sed feugiat. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Ut pellentesque augue sed urna. Vestibulum diam eros, fringilla et, consectetuer eu, nonummy id, sapien. Nullam at lectus. In sagittis ultrices mauris. Curabitur malesuada erat sit amet massa. Fusce blandit. Aliquam erat volutpat. Aliquam euismod. Aenean vel lectus. Nunc imperdiet justo nec dolor.

Etiam euismod. Fusce facilisis lacinia dui. Suspendisse potenti. In mi erat, cursus id, nonummy sed, ullamcorper eget, sapien. Praesent pretium, magna in eleifend egestas, pede pede pretium lorem, quis consectetuer tortor sapien facilisis magna. Mauris quis magna varius nulla scelerisque imperdiet. Aliquam non quam. Aliquam porttitor quam a lacus. Praesent vel arcu ut tortor cursus volutpat. In vitae pede quis diam bibendum placerat. Fusce elementum convallis neque. Sed dolor orci, scelerisque ac, dapibus nec, ultricies ut, mi. Duis nec dui quis leo sagittis commodo.

6.11 Skin Family 'bicolor'

/tcb/skin=bicolor (skin)

This skin is quite similar to the standard^{→P.98} and enhanced^{→P.99} skin. But instead of a segmentation line, the optional lower part of the box is filled with a different color or drawn with a different style.

> **Environment and engines for the skin 'bicolor'**
>
> /tcb/graphical environment^{→P.72}: `tikzpicture`
> /tcb/frame engine^{→P.72}: `path`
> /tcb/interior titled engine^{→P.72}: `freelance`
> /tcb/interior engine^{→P.72}: `freelance`
> /tcb/segmentation engine^{→P.73}: `freelance`
> /tcb/title engine^{→P.73}: `path`

- The most basic usage of this skin is to set the background color of the lower part by /tcb/colbacklower^{→P.115} and all other options like for the standard^{→P.98} skin.

```
\begin{tcolorbox}[skin=bicolor,title=The title,
    colframe=FireBrick!75!black,colback=Salmon!50!white,colbacklower=Salmon]
  The upper part.
  \tcblower
  The lower part.
\end{tcolorbox}
```

> **The title**
>
> The upper part.
>
> The lower part.

- The more advanced usage of this skin is to apply the /tcb/frame style^{→P.74} and the /tcb/interior style^{→P.74} like for the enhanced^{→P.99} skin. Also, the /tcb/segmentation style^{→P.75} can be used, but it is applied to the whole lower part.

```
\begin{tcolorbox}[skin=bicolor,title=The title,
    frame style={top color=FireBrick,
                 bottom color=FireBrick!15!white,draw=black},
    interior style={left color=Salmon,right color=Salmon!50!white},
    segmentation style={right color=Salmon,left color=Salmon!50!white}]
  The upper part.
  \tcblower
  The lower part.
\end{tcolorbox}
```

> **The title**
>
> The upper part.
>
> The lower part.

/tcb/bicolor (style, no value)

This is an abbreviation for setting `skin=bicolor`.

```
\tcbset{bicolor,equal height group=bicolor,
  colback=LightGreen,colframe=DarkGreen,colbacklower=LimeGreen!75!LightGreen,
  width=(\linewidth-6mm)/4,nobeforeafter,
  left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm}
%
\begin{tcolorbox}
  This is my content.
\end{tcolorbox}\hfill
\begin{tcolorbox}
  This is my content.
  \tcblower
  More content.
\end{tcolorbox}\hfill
\begin{tcolorbox}[adjusted title=My title]
  This is my content.
\end{tcolorbox}\hfill
\begin{tcolorbox}[adjusted title=My title]
  This is my content.
  \tcblower
  More content.
\end{tcolorbox}
```

| This is my content. | This is my content.

More content. | My title

This is my content. | My title

This is my content.

More content. |

/tcb/colbacklower=⟨*color*⟩   (no default, initially `black!15!white`)

Sets the background ⟨*color*⟩ of the lower part. It depends on the skin, if this value is used.

```
\tcbset{gitexample/.style={listing and comment,comment={#1},
  skin=bicolor,boxrule=1mm,fonttitle=\bfseries,coltitle=black,
  frame style={draw=black,left color=Gold,right color=Goldenrod!50!Gold},
  colback=black,colbacklower=Goldenrod!75!Gold,
  colupper=white,collower=black,
  listing options={language={bash},aboveskip=0pt,belowskip=0pt,nolol,
  basicstyle=\ttfamily\bfseries,extendedchars=true}}}

\begin{tcblisting}{title={Snapshot of the staging area},
  gitexample={The option '-a' automatically stages all tracked and modified
              files before the commit.\par
              This can be combined with the message option '-m'
              as seen in the third line.}}
git commit
git commit -a
git commit -am 'changes to my example'
\end{tcblisting}
```

Snapshot of the staging area

```
git commit
git commit -a
git commit -am 'changes to my example'
```

The option '-a' automatically stages all tracked and modified files before the commit.
This can be combined with the message option '-m' as seen in the third line.

/tcb/skin=**bicolorfirst** (skin)

This is a flavor of `bicolor`[→P.114] which is used as a *first* part in a break sequence for `bicolor`[→P.114]. Nevertheless, this skin can be applied independently.

Environment and engines for the skin 'bicolorfirst'

/tcb/graphical environment[→P.72]:   tikzpicture
/tcb/frame engine[→P.72]:   pathfirst
/tcb/interior titled engine[→P.72]: freelance
/tcb/interior engine[→P.72]:   freelance
/tcb/segmentation engine[→P.73]:   freelance
/tcb/title engine[→P.73]:   pathfirst

```
\tcbset{skin=bicolorfirst,equal height group=bicolorfirst,
  colback=LightGreen,colframe=DarkGreen,colbacklower=LimeGreen!75!LightGreen,
  width=(\linewidth-6mm)/4,nobeforeafter,
  left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm}
%
\begin{tcolorbox}
  This is my content.
\end{tcolorbox}\hfill
\begin{tcolorbox}
  This is my content.
  \tcblower
  More content.
\end{tcolorbox}\hfill
\begin{tcolorbox}[adjusted title=My title]
  This is my content.
\end{tcolorbox}\hfill
\begin{tcolorbox}[adjusted title=My title]
  This is my content.
  \tcblower
  More content.
\end{tcolorbox}
```

This is my content.	This is my content. More content.	My title This is my content.	My title This is my content. More content.

/tcb/skin=**bicolormiddle** (skin)

This is a flavor of `bicolor`[→P.114] which is used as a *middle* part in a break sequence for `bicolor`[→P.114]. Nevertheless, this skin can be applied independently.

Environment and engines for the skin 'bicolormiddle'

/tcb/graphical environment[→P.72]:   tikzpicture
/tcb/frame engine[→P.72]:   pathmiddle
/tcb/interior titled engine[→P.72]:   freelance
/tcb/interior engine[→P.72]:   freelance
/tcb/segmentation engine[→P.73]:   freelance
/tcb/title engine[→P.73]:   pathmiddle

```
\tcbset{skin=bicolormiddle,equal height group=bicolormiddle,
  colback=LightGreen,colframe=DarkGreen,colbacklower=LimeGreen!75!LightGreen,
  width=(\linewidth-6mm)/4,nobeforeafter,
  left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm}
%
\begin{tcolorbox}
  This is my content.
\end{tcolorbox}\hfill
\begin{tcolorbox}
  This is my content.
  \tcblower
  More content.
\end{tcolorbox}\hfill
\begin{tcolorbox}[adjusted title=My title]
  This is my content.
\end{tcolorbox}\hfill
\begin{tcolorbox}[adjusted title=My title]
  This is my content.
  \tcblower
  More content.
\end{tcolorbox}
```

This is my content.	This is my content. More content.	My title This is my content.	My title This is my content. More content.

/tcb/skin=**bicolorlast** (skin)

This is a flavor of `bicolor`[→P.114] which is used as a *last* part in a break sequence for `bicolor`[→P.114]. Nevertheless, this skin can be applied independently.

Environment and engines for the skin 'bicolorlast'

/tcb/graphical environment[→P.72]:   `tikzpicture`
/tcb/frame engine[→P.72]:   `pathlast`
/tcb/interior titled engine[→P.72]: `freelance`
/tcb/interior engine[→P.72]:   `freelance`
/tcb/segmentation engine[→P.73]:   `freelance`
/tcb/title engine[→P.73]:   `pathlast`

```
\tcbset{skin=bicolorlast,equal height group=bicolorlast,
  colback=LightGreen,colframe=DarkGreen,colbacklower=LimeGreen!75!LightGreen,
  width=(\linewidth-6mm)/4,nobeforeafter,
  left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm}
%
\begin{tcolorbox}
  This is my content.
\end{tcolorbox}\hfill
\begin{tcolorbox}
  This is my content.
  \tcblower
  More content.
\end{tcolorbox}\hfill
\begin{tcolorbox}[adjusted title=My title]
  This is my content.
\end{tcolorbox}\hfill
\begin{tcolorbox}[adjusted title=My title]
  This is my content.
  \tcblower
  More content.
\end{tcolorbox}
```

| This is my content. | This is my content. More content. | My title This is my content. | My title This is my content. More content. |

6.12 Skin Family 'beamer'

/tcb/skin=beamer (skin)

This skin resembles boxes known from the beamer class and therefore is called 'beamer'. It uses the normal colors from the core package but shades them a little bit. To use this skin, the tikz library shadings has to be included in the preamble by:

```
\usetikzlibrary{shadings}
```

The appearance of the skin can be controlled by /tcb/frame style$^{\rightarrow\text{P.74}}$ and /tcb/interior style$^{\rightarrow\text{P.74}}$, if needed. Here, the *segmentation* cannot be controlled by a style.

> **Environment and engines for the skin 'beamer'**
>
> /tcb/graphical environment$^{\rightarrow\text{P.72}}$:   tikzpicture
> /tcb/frame engine$^{\rightarrow\text{P.72}}$:   path
> /tcb/interior titled engine$^{\rightarrow\text{P.72}}$: freelance
> /tcb/interior engine$^{\rightarrow\text{P.72}}$:   freelance
> /tcb/segmentation engine$^{\rightarrow\text{P.73}}$:   freelance
> /tcb/title engine$^{\rightarrow\text{P.73}}$:   path

/tcb/beamer (style, no value)

This is an abbreviation for setting skin=beamer.
It also changes the geometry and some style options.

```
\tcbset{beamer,equal height group=beamer,
  colback=LightGreen,colframe=DarkGreen,
  width=(\linewidth-6mm)/4,nobeforeafter,
  left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm}
%
\begin{tcolorbox}
  This is my content.
\end{tcolorbox}\hfill
\begin{tcolorbox}
  This is my content.
  \tcblower
  More content.
\end{tcolorbox}\hfill
\begin{tcolorbox}[adjusted title=My title]
  This is my content.
\end{tcolorbox}\hfill
\begin{tcolorbox}[adjusted title=My title]
  This is my content.
  \tcblower
  More content.
\end{tcolorbox}
```

This is my content.	This is my content. More content.	**My title** This is my content.	**My title** This is my content. More content.

```
\begin{tcolorbox}[beamer,colback=Salmon!50!white,colframe=FireBrick!75!black,
  adjusted title=A colored box with the 'beamer' skin]
This box looks like a box provided by the \texttt{beamer} class.
\end{tcolorbox}
```

A colored box with the 'beamer' skin

This box looks like a box provided by the beamer class.

```
\begin{tcolorbox}[beamer,colframe=blue,colback=black,
  watermark graphics=lichtspiel.jpg,
  coltext=white,watermark opacity=0.75,watermark stretch=1.0,
  title=Beamer Box with background picture]
\lipsum[1]
\end{tcolorbox}
```

Beamer Box with background picture

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

```
\newtcolorbox{myblock}[2][]{%
  beamer,breakable,colback=LightBlue,colframe=DarkBlue,#1,title=#2}%

\begin{myblock}{Beamerish \texttt{block}: \texttt{myblock}}
\lipsum[1]
\end{myblock}
```

Beamerish block: myblock

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

/tcb/skin=**beamerfirst** (skin)

This is a flavor of beamer[→P.119] which is used as a *first* part in a break sequence for beamer[→P.119]. Nevertheless, this skin can be applied independently.

Environment and engines for the skin 'beamerfirst'

/tcb/graphical environment[→P.72]:   tikzpicture
/tcb/frame engine[→P.72]:   pathfirst
/tcb/interior titled engine[→P.72]: freelance
/tcb/interior engine[→P.72]:   freelance
/tcb/segmentation engine[→P.73]:   freelance
/tcb/title engine[→P.73]:   pathfirst

```
\tcbset{beamer,skin=beamerfirst,equal height group=beamerfirst,
  colback=LightGreen,colframe=DarkGreen,
  width=(\linewidth-6mm)/4,nobeforeafter,
  left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm}
%
\begin{tcolorbox}
  This is my content.
\end{tcolorbox}\hfill
\begin{tcolorbox}
  This is my content.
  \tcblower
  More content.
\end{tcolorbox}\hfill
\begin{tcolorbox}[adjusted title=My title]
  This is my content.
\end{tcolorbox}\hfill
\begin{tcolorbox}[adjusted title=My title]
  This is my content.
  \tcblower
  More content.
\end{tcolorbox}
```

This is my content.	This is my content. More content.	**My title** This is my content.	**My title** This is my content. More content.

/tcb/skin=**beamermiddle** (skin)

This is a flavor of beamer[→P.119] which is used as a *middle* part in a break sequence for beamer[→P.119]. Nevertheless, this skin can be applied independently.

Environment and engines for the skin 'beamermiddle'

/tcb/graphical environment[→P.72]: tikzpicture
/tcb/frame engine[→P.72]:   pathmiddle
/tcb/interior titled engine[→P.72]: freelance
/tcb/interior engine[→P.72]:   freelance
/tcb/segmentation engine[→P.73]:   freelance
/tcb/title engine[→P.73]:   pathmiddle

```
\tcbset{beamer,skin=beamermiddle,equal height group=beamermiddle,
  colback=LightGreen,colframe=DarkGreen,
  width=(\linewidth-6mm)/4,nobeforeafter,
  left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm}
%
\begin{tcolorbox}
  This is my content.
\end{tcolorbox}\hfill
\begin{tcolorbox}
  This is my content.
  \tcblower
  More content.
\end{tcolorbox}\hfill
\begin{tcolorbox}[adjusted title=My title]
  This is my content.
\end{tcolorbox}\hfill
\begin{tcolorbox}[adjusted title=My title]
  This is my content.
  \tcblower
  More content.
\end{tcolorbox}
```

/tcb/skin=**beamerlast** (skin)

This is a flavor of beamer[→P. 119] which is used as a *last* part in a break sequence for beamer[→P. 119]. Nevertheless, this skin can be applied independently.

Environment and engines for the skin 'beamerlast'

/tcb/graphical environment[→P. 72]:   tikzpicture
/tcb/frame engine[→P. 72]:   pathlast
/tcb/interior titled engine[→P. 72]: freelance
/tcb/interior engine[→P. 72]:   freelance
/tcb/segmentation engine[→P. 73]:   freelance
/tcb/title engine[→P. 73]:   pathlast

```
\tcbset{beamer,skin=beamerlast,equal height group=beamerlast,
  colback=LightGreen,colframe=DarkGreen,
  width=(\linewidth-6mm)/4,nobeforeafter,
  left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm}
%
\begin{tcolorbox}
  This is my content.
\end{tcolorbox}\hfill
\begin{tcolorbox}
  This is my content.
  \tcblower
  More content.
\end{tcolorbox}\hfill
\begin{tcolorbox}[adjusted title=My title]
  This is my content.
\end{tcolorbox}\hfill
\begin{tcolorbox}[adjusted title=My title]
  This is my content.
  \tcblower
  More content.
\end{tcolorbox}
```

This is my content.	This is my content.	**My title**	**My title**
	More content.	This is my content.	This is my content.
			More content.

6.13 Skin Family 'widget'

/tcb/skin=widget (skin)

This skin uses the normal colors from the core package but shades them a little bit. To use this skin, the `tikz` library `shadings` has to be included in the preamble by:

```
\usetikzlibrary{shadings}
```

The appearance of the skin can be controlled by /tcb/frame style^{→P.74}, /tcb/interior style^{→P.74}, and /tcb/segmentation style^{→P.75}, if needed.

> **Environment and engines for the skin 'widget'**
>
> /tcb/graphical environment^{→P.72}: tikzpicture
> /tcb/frame engine^{→P.72}:   path
> /tcb/interior titled engine^{→P.72}: path
> /tcb/interior engine^{→P.72}:   path
> /tcb/segmentation engine^{→P.73}: freelance
> /tcb/title engine^{→P.73}:   freelance

/tcb/widget (style, no value)

This is an abbreviation for setting `skin=widget`.

It also changes the geometry and some style options.

```
\tcbset{widget,equal height group=widget,
  colback=LightGreen,colframe=DarkGreen,
  width=(\linewidth-6mm)/4,nobeforeafter,
  left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm}
%
\begin{tcolorbox}
  This is my content.
\end{tcolorbox}\hfill
\begin{tcolorbox}
  This is my content.
  \tcblower
  More content.
\end{tcolorbox}\hfill
\begin{tcolorbox}[adjusted title=My title]
  This is my content.
\end{tcolorbox}\hfill
\begin{tcolorbox}[adjusted title=My title]
  This is my content.
  \tcblower
  More content.
\end{tcolorbox}
```

This is my content.	This is my content. More content.	**My title** This is my content.	**My title** This is my content. More content.

```
\begin{tcolorbox}[widget,colback=Salmon!50!white,colframe=FireBrick!75!black,
  adjusted title=A colored box with the 'widget' skin]
This is my content.
\end{tcolorbox}
```

A colored box with the 'widget' skin

This is my content.

/tcb/skin=**widgetfirst** (skin)

This is a flavor of widget[→P. 124] which is used as a *first* part in a break sequence for widget[→P. 124]. Nevertheless, this skin can be applied independently.

Environment and engines for the skin 'widgetfirst'

/tcb/graphical environment[→P. 72]:	tikzpicture
/tcb/frame engine[→P. 72]:	pathfirst
/tcb/interior titled engine[→P. 72]:	pathfirst
/tcb/interior engine[→P. 72]:	pathfirst
/tcb/segmentation engine[→P. 73]:	freelance
/tcb/title engine[→P. 73]:	freelance

```
\tcbset{widget,skin=widgetfirst,equal height group=widgetfirst,
  colback=LightGreen,colframe=DarkGreen,
  width=(\linewidth-6mm)/4,nobeforeafter,
  left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm}
%
\begin{tcolorbox}
  This is my content.
\end{tcolorbox}\hfill
\begin{tcolorbox}
  This is my content.
  \tcblower
  More content.
\end{tcolorbox}\hfill
\begin{tcolorbox}[adjusted title=My title]
  This is my content.
\end{tcolorbox}\hfill
\begin{tcolorbox}[adjusted title=My title]
  This is my content.
  \tcblower
  More content.
\end{tcolorbox}
```

This is my content.	This is my content. More content.	**My title** This is my content.	**My title** This is my content. More content.

/tcb/skin=**widgetmiddle** (skin)

This is a flavor of `widget`[→P.124] which is used as a *middle* part in a break sequence for `widget`[→P.124]. Nevertheless, this skin can be applied independently.

Environment and engines for the skin 'widgetmiddle'

/tcb/graphical environment[→P.72]:	tikzpicture
/tcb/frame engine[→P.72]:	pathmiddle
/tcb/interior titled engine[→P.72]:	pathmiddle
/tcb/interior engine[→P.72]:	pathmiddle
/tcb/segmentation engine[→P.73]:	freelance
/tcb/title engine[→P.73]:	freelance

```
\tcbset{widget,skin=widgetmiddle,equal height group=widgetmiddle,
  colback=LightGreen,colframe=DarkGreen,
  width=(\linewidth-6mm)/4,nobeforeafter,
  left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm}
%
\begin{tcolorbox}
  This is my content.
\end{tcolorbox}\hfill
\begin{tcolorbox}
  This is my content.
  \tcblower
  More content.
\end{tcolorbox}\hfill
\begin{tcolorbox}[adjusted title=My title]
  This is my content.
\end{tcolorbox}\hfill
\begin{tcolorbox}[adjusted title=My title]
  This is my content.
  \tcblower
  More content.
\end{tcolorbox}
```

This is my content.	This is my content. More content.	**My title** This is my content.	**My title** This is my content. More content.

/tcb/skin=**widgetlast** (skin)

This is a flavor of `widget`[→P. 124] which is used as a *last* part in a break sequence for `widget`[→P. 124]. Nevertheless, this skin can be applied independently.

Environment and engines for the skin 'widgetlast'

/tcb/graphical environment[→P. 72]: `tikzpicture`
/tcb/frame engine[→P. 72]: `pathlast`
/tcb/interior titled engine[→P. 72]: `pathlast`
/tcb/interior engine[→P. 72]: `pathlast`
/tcb/segmentation engine[→P. 73]: `freelance`
/tcb/title engine[→P. 73]: `freelance`

```
\tcbset{widget,skin=widgetlast,equal height group=widgetlast,
  colback=LightGreen,colframe=DarkGreen,
  width=(\linewidth-6mm)/4,nobeforeafter,
  left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm}
%
\begin{tcolorbox}
  This is my content.
\end{tcolorbox}\hfill
\begin{tcolorbox}
  This is my content.
  \tcblower
  More content.
\end{tcolorbox}\hfill
\begin{tcolorbox}[adjusted title=My title]
  This is my content.
\end{tcolorbox}\hfill
\begin{tcolorbox}[adjusted title=My title]
  This is my content.
  \tcblower
  More content.
\end{tcolorbox}
```

This is my content.	This is my content. More content.	**My title** This is my content.	**My title** This is my content. More content.

6.14 Skin 'draft'

/tcb/skin=draft (skin)

This skin is intended to be used while drafting new geometric settings for a `tcolorbox`.

> **Environment and engines for the skin 'draft'**
>
> /tcb/graphical environment$^{\rightarrow\text{P.72}}$:   tikzpicture
> /tcb/frame engine$^{\rightarrow\text{P.72}}$:   freelance
> /tcb/interior titled engine$^{\rightarrow\text{P.72}}$:   freelance
> /tcb/interior engine$^{\rightarrow\text{P.72}}$:   freelance
> /tcb/segmentation engine$^{\rightarrow\text{P.73}}$:   path
> /tcb/title engine$^{\rightarrow\text{P.73}}$:   path

/tcb/draft (style, no value)

This is an abbreviation for setting `skin=draft`.

```
\tcbset{draft,equal height group=draft,
  colback=LightGreen,colframe=DarkGreen,
  width=(\linewidth-6mm)/4,nobeforeafter,
  left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm}
%
\begin{tcolorbox}
  This is my content.
\end{tcolorbox}\hfill
\begin{tcolorbox}
  This is my content.
  \tcblower
  More content.
\end{tcolorbox}\hfill
\begin{tcolorbox}[adjusted title=My title]
  This is my content.
\end{tcolorbox}\hfill
\begin{tcolorbox}[adjusted title=My title]
  This is my content.
  \tcblower
  More content.
\end{tcolorbox}
```

frame: w=101.71863pt, h=56.11296pt frame: w=101.71863pt, h=56.11296pt frame: w=101.71863pt, h=56.11296pt frame: w=101.71863pt, h=56.11296pt

This is my content.   upper: w=87.49234pt, h=7.95pt   title: w=87.49234pt, h=9.2425pt   title: w=87.49234pt, h=9.2425pt

upper: w=87.49234pt, h=41.8866   More content.   This is my content.   upper: w=87.49234pt, h=7.95pt

  lower: w=87.49234pt, h=22.5586   lower: w=87.49234pt, h=25.5310   lower: w=87.49234pt, h=22.5586

interior: w=98.87338pt, h=53.2267pt interior: w=98.87338pt, h=53.2267pt interior: w=98.87338pt, h=36.9489pt interior: w=98.87338pt, h=36.9489pt

128

```
\vspace*{3mm}
\begin{tcolorbox}[draft,title=A colored box with the 'draft' skin]
\lipsum[1-3]
\tcblower
\lipsum[4-6]
\end{tcolorbox}
```

frame: w=423.94617pt, h=506.20157pt

A colored box with the 'draft' skin   title: w=392.64822pt, h=6.2pt

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat
ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget,
consectetuer 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 accum-
san bibendum, erat ligula aliqu   upper: w=392.64822pt, h=249.95pt us 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.
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, consectetuer 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 d   lower: w=392.64822pt, h=205.95pt agittis 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 consectetuer.
Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean
faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut
lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque
vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec
luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in,
velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

interior: w=421.10092pt, h=490.04318pt
```

# 7 Libraries ▣ listings, ▣ listingsutf8, and ▣ minted

## 7.1 Loading the Libraries

In contrast to other `tcolorbox` libraries, the libraries ▣ listings, ▣ listingsutf8, and ▣ minted are concurrent in the sense that they all do the same thing, i. e. displaying listings with or without typesetting the listing in LATEX parallel. The difference is the underlying LATEX package which does the core job for displaying a listing. So, typically, you need just *one* of these libraries. If you do not have a clue, which one of them you should use, you should take ▣ listingsutf8.

> The order in which the libraries are included influences the default settings and the `/tcb/reset`→ P. 64 behavior. The settings of a later loaded library overwrite the settings of a previous loaded library. A library is never loaded twice.

### 7.1.1 Loading ▣ listings

This library uses the package `listings` [5] to typeset listings. It is loaded by a package option or inside the preamble by:

```
\tcbuselibrary{listings}
```

This also loads the package `listings` [5].

The `/tcb/listing engine`→ P. 139 is set to `listings` by the library. To reactivate this setting, if overwritten by other libraries, use

```
\tcbset{listing engine=listings}
```

7.1.2 Loading ▣ listingsutf8

To extend `listings` for UTF-8 encoded sources, you can use the support from the package `listingsutf8` [8] by loading the library variant ▣ listingsutf8.

```
\tcbuselibrary{listingsutf8}
\tcbset{listing utf8=latin1}% optional; 'latin1' is the default.
```

This also loads the library ▣ listings and the packages `listings` [5] and `listingsutf8` [8].

The `/tcb/listing engine`→ P. 139 is set to `listings` by the library. To reactivate this setting, if overwritten by other libraries, use

```
\tcbset{listing engine=listings}
```

7.1.3 Loading ᴸᴵᴮ minted

This library uses the package `minted` [11] to typeset listings. It is loaded by a package option or inside the preamble by:

```
\tcbuselibrary{minted}
```

This also loads the package `minted` [11].

The `minted` package uses the external tool `Pygments` [9] to apply syntax highlighting. It has to be installed and set up, before the library can be used, see [11] and [9]. The `tcolorbox` library ᴸᴵᴮ minted does not work, if the package `minted` [11] does not work.

The `/tcb/listing engine`^(→P.139) is set to `minted` by the library. To reactivate this setting, if overwritten by other libraries, use

```
\tcbset{listing engine=minted}
```

7.2 Common Macros of the Libraries

\begin{**tcblisting**}{⟨*options*⟩}
  ⟨*environment content*⟩
\end{**tcblisting**}

Creates a colored box based on a `tcolorbox`^(→P.7). Controlled by the given ⟨*options*⟩, the environment content is typeset normally and/or as a listing. Furthermore, the ⟨*options*⟩ control appearance and functions of the `tcolorbox`. By default, the listing is interpreted as a LATEX listing.

```
\begin{tcblisting}{colback=red!5!white,colframe=red!75!black}
This is a \LaTeX\ example which displays the text as source code
and in compiled form.
\end{tcblisting}
```

> This is a \LaTeX\ example which displays the text as source code
> and in compiled form.
> --
> This is a LATEX example which displays the text as source code and in compiled form.

```
% \usetikzlibrary{listings} /or/ \usetikzlibrary{listingsutf8}
\begin{tcblisting}{colback=yellow!5,colframe=yellow!50!black,listing only,
    title=This is source code in another language (XML), fonttitle=\bfseries,
  listing options={language=XML,columns=fullflexible,keywordstyle=\color{red}}}
<?xml version="1.0"?>
<project name="Package tcolorbox" default="documentation" basedir=".">
  <description>
    Apache Ant build file (http://ant.apache.org/)
  </description>
</project>
\end{tcblisting}
```

This is source code in another language (XML)

```
<?xml version="1.0"?>
<project name="Package␣tcolorbox" default="documentation" basedir=".">
  <description>
    Apache Ant build file (http://ant.apache.org/)
  </description>
</project>
```

```
% \usetikzlibrary{minted}
\begin{tcblisting}{colback=yellow!5,colframe=yellow!50!black,listing only,
    title=This is source code in another language (XML), fonttitle=\bfseries,
  listing engine=minted,minted language=xml}
<?xml version="1.0"?>
<project name="Package tcolorbox" default="documentation" basedir=".">
  <description>
    Apache Ant build file (http://ant.apache.org/)
  </description>
</project>
\end{tcblisting}
```

This is source code in another language (XML)

```
<?xml version="1.0"?>
<project name="Package tcolorbox" default="documentation" basedir=".">
  <description>
    Apache Ant build file (http://ant.apache.org/)
  </description>
</project>
```

```
% This box is as wide as needed (listing only !!)
% \usetikzlibrary{skins}
\begin{tcblisting}{colback=green!5!white,colframe=green!50!black,listing only,
  hbox,enhanced,drop fuzzy shadow,before=\begin{center},after=\end{center}}
\begin{tikzpicture}
\fill[red] (0,0) rectangle (1,1);
\end{tikzpicture}
\end{tcblisting}
```

```
\begin{tikzpicture}
\fill[red] (0,0) rectangle (1,1);
\end{tikzpicture}
```

\begin{**tcboutputlisting**}

⟨*environment content*⟩

\end{**tcboutputlisting**}

Saves the environment content to a file which is named by the key value of listing file. Later, this file can be loaded by \tcbinputlisting or \tcbuselistingtext or \tcbuselistinglisting.

```
\begin{tcboutputlisting}
This \textbf{text} is written to a standardized file for later usage.
\end{tcboutputlisting}
```

\tcbinputlisting{⟨*options*⟩}

Creates a colored boxed based on a tcolorbox. The text content is read from a file named by the key value of listing file. Apart from that, the function is equal to that of tcblisting[→P. 131].

```
\tcbinputlisting{colback=red!5!white,colframe=red!75!black,text only}
\tcbinputlisting{colback=green!5,colframe=green!75!black,listing only}
```

```
\begin{tikzpicture}
\fill[red] (0,0) rectangle (1,1);
\end{tikzpicture}
```

\tcbuselistingtext

Loads text from a file named by the key value of listing file.

```
\tcbuselistingtext
```

\tcbuselistinglisting

Typesets text as listing from a file named by the key value of listing file.

```
\tcbuselistinglisting

\begin{tikzpicture}
\fill[red] (0,0) rectangle (1,1);
\end{tikzpicture}
```

\tcbusetemplisting

Typesets text as listing from a temporary file which was written by tcbwritetemp[→P. 70].

\newtcblisting[⟨*init options*⟩]{⟨*name*⟩}[⟨*number*⟩][⟨*default*⟩]{⟨*options*⟩}

Creates a new environment ⟨*name*⟩ based on `tcblisting`^{→P.131}. Basically, \newtcblisting operates like \newenvironment. This means, the new environment ⟨*name*⟩ optionally takes ⟨*number*⟩ arguments, where ⟨*default*⟩ is the default value for the optional first argument. The ⟨*options*⟩ are given to the underlying `tcblisting`. Note that `/tcb/savedelimiter`^{→P.14} is set to the given ⟨*name*⟩ automatically. The ⟨*init options*⟩ allow to set up automatic numbering, see Section 4 from page 65.

```
\newtcblisting{mybox}{%
  colback=red!5!white,
  colframe=red!75!black}

\begin{mybox}
This is my \LaTeX\ box.
\end{mybox}
```

This is my \LaTeX\ box.
- - - - - - - - - - - - - - - - - -
This is my LaTeX box.

```
\newtcblisting{mybox}[1]{%
  colback=red!5!white,
  colframe=red!75!black,
  fonttitle=\bfseries,
  title=#1}

\begin{mybox}{Listing Box}
This is my \LaTeX\ box.
\end{mybox}
```

Listing Box

This is my \LaTeX\ box.
- - - - - - - - - - - - - - - - - -
This is my LaTeX box.

```
\newtcblisting{mybox}[2][]{%
  colback=red!5!white,
  colframe=red!75!black,
  fonttitle=\bfseries,
  title=#2,#1}

\begin{mybox}[listing only]
  {Listing Box}
This is my \LaTeX\ box.
\end{mybox}
\bigskip

\begin{mybox}[listing side text]
  {Listing Box}
This is my
\LaTeX\ box.
\end{mybox}
```

Listing Box

This is my \LaTeX\ box.

Listing Box

This is my   This   is   my
\LaTeX\ box.   LaTeX box.

Definition in the preamble:

```
\newtcblisting[auto counter]{mycbox}[1]{%
  colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries,
  title=Listing \thetcbcounter: #1}
```

```
\begin{mycbox}{Listing Box}
This is my \LaTeX\ box.
\end{mycbox}
```

Listing 1: Listing Box

This is my \LaTeX\ box.
- - - - - - - - - - - - - - - - - -
This is my LaTeX box.

\renewtcblisting[⟨*init options*⟩]{⟨*name*⟩}[⟨*number*⟩][⟨*default*⟩]{⟨*options*⟩}

Operates like \newtcblisting, but based on \renewenvironment instead of \newenvironment. An existing environment is redefined.

\newtcbinputlisting[⟨*init options*⟩]{\⟨*name*⟩}[⟨*number*⟩][⟨*default*⟩]{⟨*options*⟩}

Creates a new macro \⟨*name*⟩ based on \tcbinputlisting[→P. 133]. Basically, \newtcbinputlisting operates like \newcommand. The new macro \⟨*name*⟩ optionally takes ⟨*number*⟩+1 arguments, where ⟨*default*⟩ is the default value for the optional first argument. The ⟨*options*⟩ are given to the underlying tcbinputlisting. The ⟨*init options*⟩ allow to set up automatic numbering, see Section 4 from page 65.

```
\newtcbinputlisting[use counter from=mycbox]{\mylisting}[2][]{%
  listing file={#2},
  title=Listing (\thetcbcounter) of \texttt{#2},
  colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries,
  listing only,breakable,#1}

\mylisting[before upper=\textit{This is the included file content:}]
          {\jobname.tcbtemp}
```

Listing (2) of tcolorbox.tcbtemp

This is the included file content:
```
\newtcbinputlisting[use counter from=mycbox]{\mylisting}[2][]{%
  listing file={#2},
  title=Listing (\thetcbcounter) of \texttt{#2},
  colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries,
  listing only,breakable,#1}

\mylisting[before upper=\textit{This is the included file content:}]
          {\jobname.tcbtemp}
```

```
\newtcbinputlisting[use counter from=mycbox]{\mylisting}[2][]{%
  listing engine=minted,minted language=latex,minted style=colorful,
  listing file={#2},
  title=Listing (\thetcbcounter) of \texttt{#2},
  colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries,
  listing only,breakable,#1}

\mylisting[before upper=\textit{This is the included file content:}]
          {\jobname.tcbtemp}
```

Listing (3) of tcolorbox.tcbtemp

This is the included file content:
```
\newtcbinputlisting[use counter from=mycbox]{\mylisting}[2][]{%
  listing engine=minted,minted language=latex,minted style=colorful,
  listing file={#2},
  title=Listing (\thetcbcounter) of \texttt{#2},
  colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries,
  listing only,breakable,#1}

\mylisting[before upper=\textit{This is the included file content:}]
          {\jobname.tcbtemp}
```

\renewtcbinputlisting[⟨*init options*⟩]{\⟨*name*⟩}[⟨*number*⟩][⟨*default*⟩]{⟨*options*⟩}

Operates like \newtcbinputlisting, but based on \renewcommand instead of \newcommand. An existing macro is redefined.

7.3 Option Keys of the [LIB listings] Library

/tcb/listing options=⟨*key list*⟩   (no default, initially `style=tcblatex`)

Sets the options from the package `listings` [5] which are used during typesetting of the listing. For LaTeX listings, there is a predefined `listings` style named `tcblatex` which can be used.

```
\begin{tcblisting}{colback=red!5!white,colframe=red!25,left=6mm,
listing options={style=tcblatex,numbers=left,numberstyle=\tiny\color{red!75!black}}}
This is a \LaTeX\ example which displays the text as source code
and in compiled form. Additionally, we use line numbers here.
\end{tcblisting}
```

```
1  This is a \LaTeX\ example which displays the text as source code
2  and in compiled form. Additionally, we use line numbers here.
```

This is a LaTeX example which displays the text as source code and in compiled form. Additionally, we use line numbers here.

/tcb/no listing options   (no value, initially unset)

Abbreviation for `listing options={}`. This removes all options for the `listings` package. This includes the `tcblisting` standard style `tcblatex` and the encoding presets. Use this option, if you want to set the `listings` options outside of `tcblisting`, e.g. globally in the preamble.

```
\begin{tcblisting}{no listing options}
All \textit{listings} options removed.
\end{tcblisting}
```

```
All \textit{listings} options removed.
```

All *listings* options removed.

/tcb/listing style=⟨*style*⟩   (no default, initially `tcblatex`)

Abbreviation for `listing options={style=...}`. This key sets a ⟨*style*⟩ for the `listings` package, see [5]. For LaTeX, there is a predefined style named `tcblatex`.

```
\begin{tcblisting}{colback=red!5!white,colframe=red!75!black,
listing style=tcblatex}
Here, we use the predefined style.
\end{tcblisting}
```

```
Here, we use the predefined style.
```

Here, we use the predefined style.

/tcb/listing inputencoding=⟨*encoding*⟩   (no default, initially `\inputencodingname`)

Sets the input encoding value for the predefined listing style `tcblatex` and `tcbdocumentation` from the library LIB documentation. The initial value is derived from the package `inputenc` if used.

/tcb/listing remove caption=⟨*boolean value*⟩   (default `true`, initially `true`)

If set to `true`, some part of the caption building code of the `listings` package is silenced to prevent some unwanted interaction with the `hyperref` package resulting in additional vertical space. If set to `false`, the `listings` package code is kept unchanged. Note that listings outside `tcblisting`→P.131 and `\tcbinputlisting`→P.133 are always processed normally. Typically, a user is not expected to use this key at all.

See further options in Section 7.6 on page 139.

7.4 Option Keys of the LIB listingsutf8 Library

The LIB listingsutf8 library is an extension of the LIB listings library, so all options from Section 7.3 on the previous page are applicable.

/tcb/listing utf8=⟨*one-byte-encoding*⟩   (style, no default, initially `latin1`)

Abbreviation for using **/tcb/listing inputencoding** together with UTF-8 support from the package `listingsutf8` [8]. This option is available only for the library variant LIB listingsutf8. The ⟨*one-byte-encoding*⟩ is one of the applicable encodings from [8], e. g. `latin1`.

See further options in Section 7.6 on page 139.

7.5 Option Keys of the LIB minted Library

/tcb/minted language=⟨*programming language*⟩   (no default, initially `latex`)

Sets a ⟨*programming language*⟩ known to `Pygments` [9].

```
\begin{tcblisting}{listing engine=minted,minted style=trac,
  minted language=java,
  colback=red!5!white,colframe=red!75!black,listing only}
public class HelloWorld {
  // A 'Hello World' in Java
  public static void main(String[] args) {
    System.out.println("Hello World!");
  }
}
\end{tcblisting}
```

```
public class HelloWorld {
  // A 'Hello World' in Java
  public static void main(String[] args) {
    System.out.println("Hello World!");
  }
}
```

137

/tcb/minted options=⟨*key list*⟩   (no default, initially `tabsize=2,fontsize=\small`)
Sets the options from the package `minted` [11] which are used during typesetting of the listing.

```
% \tcbuselibrary{skins}
\newtcblisting{myjava}{listing engine=minted,minted style=colorful,
  minted language=java,minted options={fontsize=\small,linenos,numbersep=3mm},
  colback=blue!5!white,colframe=blue!75!black,listing only,
  left=5mm,enhanced,
  overlay={\begin{tcbclipinterior}\fill[red!20!blue!20!white] (frame.south west)
    rectangle ([xshift=5mm]frame.north west);\end{tcbclipinterior}}}

\begin{myjava}
public class HelloWorld {
  // A 'Hello World' in Java
  public static void main(String[] args) {
    System.out.println("Hello World!");
  }
}
\end{myjava}
```

```
1  public class HelloWorld {
2    // A 'Hello World' in Java
3    public static void main(String[] args) {
4      System.out.println("Hello World!");
5    }
6  }
```

/tcb/minted style=⟨*style*⟩   (no default, initially unset)
Sets a ⟨*style*⟩ known to `Pygments` [9]. This is independent from `/tcb/minted options`.
Note that styles are always applied globally; all following examples will be set in the given
⟨*style*⟩ until a new style is set. Also note that setting `\usemintedstyle{`⟨*style*⟩`}` only once
per document is more economic, if all styles in a document are the same. For examples of
different styles, see `/tcb/minted language`^{→P.137} and `/tcb/minted options`.

See further options in Section 7.6 on the next page.

7.6 Common Option Keys of all Libraries

For the ⟨*options*⟩ in `tcblisting`^{→P. 131} respectively `\tcbinputlisting`^{→P. 133} the following `pgf` keys can be applied. The key tree path `/tcb/` is not to be used inside these macros.

/tcb/listing engine=⟨*engine*⟩ (no default)

Sets the ⟨*engine*⟩ which typesets the listings. Feasible values are

- `listings`, if library **LIB** listings or **LIB** listingsutf8 is loaded.
- `minted`, if library **LIB** minted is loaded.

/tcb/listing file=⟨*file name*⟩ (no default, initially `\jobname.listing`)

Sets the ⟨*file name*⟩ of the file which is used to save listings.

/tcb/listing and text (no value, initially set)

Typesets the environment content as listing in the upper part and as compiled text in the lower part.

```
\begin{tcblisting}{colback=red!5!white,colframe=red!75!black,listing and text}
This is a \LaTeX\ example.
\end{tcblisting}
```

> This is a **\LaTeX** example.
> -
> This is a LᴬTEX example.

/tcb/text and listing (no value)

Typesets the environment content as compiled text in the upper part and as listing in the lower part.

```
\begin{tcblisting}{colback=red!5!white,colframe=red!75!black,text and listing}
This is a \LaTeX\ example.
\end{tcblisting}
```

> This is a LᴬTEX example.
> -
> This is a **\LaTeX** example.

/tcb/listing only (no value)

Typesets the environment content as listing.

```
\begin{tcblisting}{colback=red!5!white,colframe=red!75!black,listing only}
This is a \LaTeX\ example.
\end{tcblisting}
```

> This is a **\LaTeX** example.

/tcb/text only (no value)

Typesets the environment content as compiled text.

```
\begin{tcblisting}{colback=red!5!white,colframe=red!75!black,text only}
This is a \LaTeX\ example.
\end{tcblisting}
```

> This is a LATEX example.

/tcb/comment=⟨*text*⟩ (no default, initially empty)

Records a comment with ⟨*text*⟩ as content. The comment is displayed only in conjunction with `/tcb/listing and comment` and `/tcb/comment and listing`.

```
\begin{tcblisting}{comment={This comment is really only a comment},
  colback=red!5!white,colframe=red!75!black}
This is a \textbf{tcolorbox}.
\end{tcblisting}
```

> This is a \textbf{tcolorbox}.
> --
> This is a **tcolorbox**.

/tcb/listing and comment (no value)

Typesets the environment content as listing in the upper part and a given comment in the lower part.

```
\begin{tcblisting}{colback=red!5!white,colframe=red!75!black,listing and comment,
  comment={This is my comment. It may contain line breaks.\par
    It can even use the environment content
    \flqq\ignorespaces\tcbuselistingtext\unskip\frqq}}
This is a \LaTeX\ example.
\end{tcblisting}
```

> This is a \LaTeX\ example.
> --
> This is my comment. It may contain line breaks.
> It can even use the environment content «This is a LATEX example.»

/tcb/comment and listing (no value)

Typesets a given comment in the upper part and the environment content as listing in the lower part.

```
\begin{tcblisting}{colback=red!5!white,colframe=red!75!black,comment and listing,
comment={This is my comment.}}
This is a \LaTeX\ example.
\end{tcblisting}
```

> This is my comment.
> --
> This is a \LaTeX\ example.

/tcb/listing side text (no value)

Typesets the environment content side by side as listing in the left (upper) part and as compiled text in the right (lower) part.

```
\begin{tcblisting}{colback=red!5!white,colframe=red!75!black,listing side text}
This is a \LaTeX\ example.
\end{tcblisting}
```

This is a \LaTeX\ example.   This is a LaTeX example.

/tcb/text side listing (no value)

Typesets the environment content side by side as compiled text in the left (upper) part and as listing in the right (lower) part.

```
\begin{tcblisting}{colback=red!5!white,colframe=red!75!black,text side listing}
This is a \LaTeX\ example.
\end{tcblisting}
```

This is a LaTeX example.   This is a \LaTeX\ example.

/tcb/listing outside text (no value)

Typesets the environment content side by side as listing in a `tcolorbox` and as compiled text outside the box in the right part of the page. Nevertheless, the outside text is treated as *lower* part of the `tcolorbox` and can be formatted with all lower part options. The space partitioning is done with the side by side options from Section 3.9, see page 47.

```
\begin{tcblisting}{colback=red!5!white,colframe=red!75!black,listing outside text}
This is a \LaTeX\ example.
\end{tcblisting}
```

This is a \LaTeX\ example.   This is a LaTeX example.

/tcb/text outside listing (no value)

Typesets the environment content side by side as listing in a `tcolorbox` and as compiled text outside the box in the left part of the page. Nevertheless, the outside text is treated as *lower* part of the `tcolorbox` and can be formatted with all lower part options. The space partitioning is done with the side by side options from Section 3.9, see page 47.

```
\begin{tcblisting}{colback=red!5!white,colframe=red!75!black,text outside listing}
This is a \LaTeX\ example.
\end{tcblisting}
```

This is a LaTeX example.   This is a \LaTeX\ example.

7.7 Creation of LaTeX Tutorials

The following source code gives a guideline for the creation of LaTeX tutorials. In the next section, a framework for LaTeX exercises is described. All examples shall be numbered optionally.

Firstly, some additional `tcb` keys are defined for the appearance. For the examples, three environments `texexp`, `texexptitled`, and `texexptitledspec` are defined with automatic numbering.

- `texexp` is used for untitled examles,

- `texexptitled` is used for titled examles,

- `texexptitledspec` is used for titled examles with special treatment.

Definition in the preamble:
```
\tcbset{
  texexp/.style={colframe=red!50!yellow!50!black, colback=red!50!yellow!5!white,
      coltitle=red!50!yellow!3!white,
      fonttitle=\small\sffamily\bfseries, fontupper=\small, fontlower=\small},
  example/.style 2 args={texexp,
      title={Example \thetcbcounter: #1},label={#2}},
}

\newtcblisting{texexp}[1]{texexp,#1}
\newtcblisting[auto counter,number within=section]{texexptitled}[3][]{%
  example={#2}{#3},#1}
\newtcolorbox[use counter from=texexptitled]{texexptitledspec}[3][]{%
  example={#2}{#3},#1}
```

```
\begin{tcblisting}{texexp}
This is a \LaTeX\ example which displays the text as source code
and in compiled form.
\end{tcblisting}
```

> This is a **\LaTeX** example which displays the text as source code
> and in compiled form.
> -
> This is a LaTeX example which displays the text as source code and in compiled form.

```
\begin{texexptitled}{First example with a title line}{firstExample}
Here, we use Example \ref{firstExample} with a title line.
\end{texexptitled}
```

> **Example 7.1: First example with a title line**
>
> Here, we use Example **\ref**{firstExample} with a title line.
> -
> Here, we use Example 7.1 with a title line.

```
\begin{texexp}{}
This is a \LaTeX\ example which displays the text as source code
and in compiled form.
\end{texexp}
```

This is a **\LaTeX** example which displays the text as source code
and in compiled form.

This is a LaTeX example which displays the text as source code and in compiled form.

```
\begin{texexp}{text and listing}
This is a \LaTeX\ example which displays the text as source code
and in compiled form.
\end{texexp}
```

This is a LaTeX example which displays the text as source code and in compiled form.

This is a **\LaTeX** example which displays the text as source code
and in compiled form.

```
\begin{texexp}{listing only}
This is a \LaTeX\ example which displays the text as source code only.
\end{texexp}
```

This is a **\LaTeX** example which displays the text as source code only.

```
\begin{texexp}{text only}
This is a \LaTeX\ example which displays the text in compiled form only.
\end{texexp}
```

This is a LaTeX example which displays the text in compiled form only.

```
\begin{texexptitled}{An Example with a Heading}{heading1}
This is a \LaTeX\ example with a numbered heading line
which can be referred to.
\end{texexptitled}
Here, we see Example \ref{heading1}.
```

Example 7.2: An Example with a Heading

```
This is a \LaTeX\ example with a numbered heading line
which can be referred to.
```

This is a LaTeX example with a numbered heading line which can be referred to.

Here, we see Example 7.2.

```
\begin{texexptitled}[listing only]{Another Example with a Heading}{heading2}
The keys can be used in combination. Here, an example with a heading line
and source code only is given.
\end{texexptitled}
Here, we see Example \ref{heading2}.
```

Example 7.3: Another Example with a Heading

```
The keys can be used in combination. Here, an example with a heading line
and source code only is given.
```

Here, we see Example 7.3.

```
\begin{texexptitled}[float]{A floating Example with a Heading}{heading3}
This is another \LaTeX\ example with numbered heading line.
But now, the box is a floating object.
\end{texexptitled}
```

Example 7.4: A floating Example with a Heading

```
This is another \LaTeX\ example with numbered heading line.
But now, the box is a floating object.
```

This is another LaTeX example with numbered heading line. But now, the box is a floating object.

```
The floating box of the last example is seen as Example \ref{heading3}
on page \pageref{heading3}.
```

The floating box of the last example is seen as Example 7.4 on page 144.

```
\begin{texexptitledspec}{Special application}{texexpbox1}
\begin{lstlisting}[style=tcblatex]
Some \LaTeX\ source code.
\end{lstlisting}
\tcblower
For special cases, the environment |texexptitledspec| with style
|example| can be used directly. As one can see, the upper and the lower
part of the box can be used uncoupled also.
\end{texexptitledspec}
```

Example 7.5: Special application

```
Some \LaTeX\ source code.
```

For special cases, the environment `texexptitledspec` with style `example` can be used directly. As one can see, the upper and the lower part of the box can be used uncoupled also.

The following series of examples demonstrate the application of `tcolorbox`[→P.7] options for diversification.

```
\begin{texexptitled}{How to use options (1):\par The basic example}{options1}
\begin{tikzpicture}
\path[fill=yellow!50!white] (0,0) circle (11mm);
\path[fill=white] (0,0) circle (9mm);
\foreach \w/\c in {90/red,210/green,330/blue}
{\path[shading=ball,ball color=\c] (\w:1cm) circle (7mm);}
\end{tikzpicture}
\end{texexptitled}
```

Example 7.6: How to use options (1):
The basic example

```
\begin{tikzpicture}
\path[fill=yellow!50!white] (0,0) circle (11mm);
\path[fill=white] (0,0) circle (9mm);
\foreach \w/\c in {90/red,210/green,330/blue}
{\path[shading=ball,ball color=\c] (\w:1cm) circle (7mm);}
\end{tikzpicture}
```

```
\begin{texexptitled}[center lower,enhanced,segmentation hidden,middle=0mm]
  {How to use options (2):\par The text output is centered and the
    segmentation line has vanished.}{options2}
\begin{tikzpicture}
\path[fill=yellow!50!white] (0,0) circle (11mm);
\path[fill=white] (0,0) circle (9mm);
\foreach \w/\c in {90/red,210/green,330/blue}
{\path[shading=ball,ball color=\c] (\w:1cm) circle (7mm);}
\end{tikzpicture}
\end{texexptitled}
```

Example 7.7: How to use options (2):
The text output is centered and the segmentation line has vanished.

```
\begin{tikzpicture}
\path[fill=yellow!50!white] (0,0) circle (11mm);
\path[fill=white] (0,0) circle (9mm);
\foreach \w/\c in {90/red,210/green,330/blue}
{\path[shading=ball,ball color=\c] (\w:1cm) circle (7mm);}
\end{tikzpicture}
```



```
\begin{texexptitled}[tikz lower,bicolor,colbacklower=white]
  {How to use options (3):\par Here, the |tikzpicture| is totally hidden.
    The |bicolor| skin highlights the output.}{options3}
\path[fill=yellow!50!white] (0,0) circle (11mm);
\path[fill=white] (0,0) circle (9mm);
\foreach \w/\c in {90/red,210/green,330/blue}
{\path[shading=ball,ball color=\c] (\w:1cm) circle (7mm);}
\end{texexptitled}
```

Example 7.8: How to use options (3):
Here, the `tikzpicture` is totally hidden. The `bicolor` skin highlights the output.

```
\path[fill=yellow!50!white] (0,0) circle (11mm);
\path[fill=white] (0,0) circle (9mm);
\foreach \w/\c in {90/red,210/green,330/blue}
{\path[shading=ball,ball color=\c] (\w:1cm) circle (7mm);}
```

```
\begin{texexptitled}[center lower,listing side text,righthand width=3.5cm,
  bicolor,colbacklower=white]
  {How to use options (4):\par The |bicolor| skin also works with side
   by side mode}{options4}
\begin{tikzpicture}
\path[fill=yellow!50!white] (0,0) circle (11mm);
\path[fill=white] (0,0) circle (9mm);
\foreach \w/\c in {90/red,210/green,330/blue}
{\path[shading=ball,ball color=\c]
    (\w:1cm) circle (7mm);}
\end{tikzpicture}
\end{texexptitled}
```

Example 7.9: How to use options (4):
The `bicolor` skin also works with side by side mode

```
\begin{tikzpicture}
\path[fill=yellow!50!white] (0,0) circle (11mm);
\path[fill=white] (0,0) circle (9mm);
\foreach \w/\c in {90/red,210/green,330/blue}
{\path[shading=ball,ball color=\c]
    (\w:1cm) circle (7mm);}
\end{tikzpicture}
```

```
\begin{texexptitled}[center lower,listing outside text,righthand width=3.5cm]
  {How to use options (5):\par Putting our picture outside is just
   a matter of one word.}{options5}
\begin{tikzpicture}
\path[fill=yellow!50!white] (0,0) circle (11mm);
\path[fill=white] (0,0) circle (9mm);
\foreach \w/\c in {90/red,210/green,330/blue}
{\path[shading=ball,ball color=\c]
    (\w:1cm) circle (7mm);}
\end{tikzpicture}
\end{texexptitled}
```

Example 7.10: How to use options (5):
Putting our picture outside is just a matter of one word.

```
\begin{tikzpicture}
\path[fill=yellow!50!white] (0,0) circle (11mm);
\path[fill=white] (0,0) circle (9mm);
\foreach \w/\c in {90/red,210/green,330/blue}
{\path[shading=ball,ball color=\c]
    (\w:1cm) circle (7mm);}
\end{tikzpicture}
```

```
\begin{texexptitled}[beamer,center lower,text outside listing,lefthand width=3.5cm]
  {How to use options (6):\par Our style is easily transformed into
    a beamerish one.}{options6}
\begin{tikzpicture}
\path[fill=yellow!50!white] (0,0) circle (11mm);
\path[fill=white] (0,0) circle (9mm);
\foreach \w/\c in {90/red,210/green,330/blue}
{\path[shading=ball,ball color=\c]
    (\w:1cm) circle (7mm);}
\end{tikzpicture}
\end{texexptitled}
```

Example 7.11: How to use options (6):
Our style is easily transformed into a beamerish one.


```
\begin{tikzpicture}
\path[fill=yellow!50!white] (0,0) circle (11mm);
\path[fill=white] (0,0) circle (9mm);
\foreach \w/\c in {90/red,210/green,330/blue}
{\path[shading=ball,ball color=\c]
    (\w:1cm) circle (7mm);}
\end{tikzpicture}
```

148

7.8 Creation of LaTeX Exercises

In the following, a guideline is given for the creation of LaTeX exercises with solutions. These solutions are saved to disk for application at a place of choice. Therefore, all used exercises are logged to a file `\jobname.sol` for automatic processing. The solution contents themselves are saved to a subdirectory named `solutions`.

```
%\newcounter{texercise}  % preamble
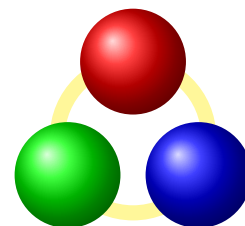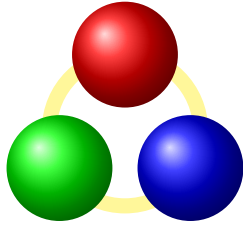
\newwrite\solout
\def\openoutsol{\immediate\openout\solout\jobname.sol}
\def\solfile#1{solutions/texercise#1.tex}
\def\writesol#1{\immediate\write\solout{\noexpand\processsol{\thetcbcounter}{#1}}}%
\def\closeoutsol{\immediate\closeout\solout}
\def\inputsol{\IfFileExists{\jobname.sol}{\input{\jobname.sol}}{}}
```

- Before the first exercise is given, `\openoutsol` has to be called to start logging.

- The solution is given as content of a `tcboutputlisting`→P.133 environment. Note, that you can use this content also inside the exercise with `\tcbuselistingtext`→P.133 in compiled form.

- After the last exercise is given (and before using the solutions), `\closeoutsol` has to be called to stop logging.

- The solutions are loaded by `\inputsol`.

Inside the exercise text, there may be text parts which are needed as LaTeX source code and as compiled text as well. These parts can be saved by `tcbwritetemp`→P.70 and used in compiled form by `\tcbusetemp`→P.70 or as source code by `\tcbusetemplisting`→P.133.

At first, we generate some a common style for the exercises and the solutions. Further, since exercises and solutions should be numbered, we force to use a label ⟨*marker*⟩. Automatically, the label `exe:`⟨*marker*⟩ is used to mark the exercise and the label `sol:`⟨*marker*⟩ is used to mark the solution.

```
\tcbset{texercisestyle/.style={arc=0.5mm, colframe=blue!25!yellow!90!white,
  colback=blue!25!yellow!5!white, coltitle=blue!25!yellow!40!black,
  fonttitle=\small\sffamily\bfseries, fontupper=\small, fontlower=\small}}
```

With these preparations, the kernel environment `texercise` for our exercises is created quickly:

```
Definition in the preamble:
\newtcolorbox[auto counter,number within=section,list inside=exam]%
  {texercise}[2][]{texercisestyle,
  listing file={\solfile\thetcbcounter},label={exe:#2},
  phantom={\writesol{#2}},
  title={Exercise \thetcbcounter\hfill\mdseries Solution on page \pageref{sol:#2}},
  list entry={\numberline{\thetcbcounter}Exercise with solution on
              page \pageref{sol:#2}},#1}
```

```
\newtcolorbox{texercise}[2][]{texercisestyle,
  listing file={\solfile\thetexercise},
  phantom={\refstepcounter{texercise}\label{exe:#2}\writesol{#2}},
  title={Exercise \arabic{texercise}%
    \hfill\mdseries Solution on page \pageref{sol:#2}},#1}
```

The following examples demonstrate the application.

```
\begin{texercise}{tabular_example}
\textit{Create the following table:}\par\smallskip%
\begin{tcboutputlisting}
\begin{tabular}{|p{3cm}|p{3cm}|p{3cm}|p{3cm}|}\hline
\multicolumn{4}{|c|}{\bfseries\itshape Das alte Italien}\\\hline
\multicolumn{2}{|c|}{\bfseries Antike} &
\multicolumn{2}{c|}{\bfseries Mittelalter}\\\hline
\multicolumn{1}{|c|}{\itshape Republik}&
\multicolumn{1}{c|}{\itshape Kaiserreich}&
\multicolumn{1}{c|}{\itshape Franken}&
\multicolumn{1}{c|}{\itshape Teilstaaten}\\\hline
In den Zeiten der r\"{o}mischen Republik standen dem Staat jeweils zwei
Konsuln vor, deren Machtbefugnisse identisch waren. &
Das r\"{o}mische Kaiserreich wurde von einem Alleinherrscher, dem Kaiser,
regiert.
& In der V\"{o}lkerwanderungszeit \"{u}bernahmen die Goten und sp\"{a}ter die
Franken die Vorherrschaft.
& Im sp\"{a}teren Mittelalter regierten F\"{u}rsten einen Fleckenteppich
von Einzelstaaten.\\\hline
\end{tabular}
\end{tcboutputlisting}
\tcbuselistingtext%
\end{texercise}
```

Exercise 7.1

Create the following table:

Das alte Italien			
Antike		**Mittelalter**	
Republik	*Kaiserreich*	*Franken*	*Teilstaaten*
In den Zeiten der römischen Republik standen dem Staat jeweils zwei Konsuln vor, deren Machtbefugnisse identisch waren.	Das römische Kaiserreich wurde von einem Alleinherrscher, dem Kaiser, regiert.	In der Völkerwanderungszeit übernahmen die Goten und später die Franken die Vorherrschaft.	Im späteren Mittelalter regierten Fürsten einen Fleckenteppich von Einzelstaaten.

```
\begin{texercise}{macro_oneparam}
\begin{tcboutputlisting}
\newcommand{\headingline}[1]{%
  \begin{center}\Large\bfseries #1\end{center}}
\end{tcboutputlisting}
\tcbuselistingtext%

Create a new macro \verb+\headingline+ which produces the
following output:\par\smallskip
\begin{tcbwritetemp}
\headingline{Very important heading}
\end{tcbwritetemp}
\tcbusetemplisting\tcbusetemp%
\end{texercise}
```

Exercise 7.2   Solution on page 153

Create a new macro \headingline which produces the following output:

\headingline{Very important heading}

Very important heading

```
\begin{texercise}{macro_twoparam}
\begin{tcboutputlisting}
\newcommand{\minitable}[2]{%
  \begin{center}\begin{tabular}{p{10cm}}\hline%
  \multicolumn{1}{c}{\bfseries#1}\\\hline%
  #2\\\hline%
  \end{tabular}\end{center}}
\end{tcboutputlisting}
\tcbuselistingtext%
Create a new macro \verb+\minitable+ which produces the
following output:\par\smallskip
\begin{tcbwritetemp}
\minitable{My heading}{In this tiny tabular, there is only a heading
  and some text below which has a width of ten centimeters.}
\end{tcbwritetemp}
\tcbusetemplisting\par\smallskip\tcbusetemp%
\end{texercise}
```

Exercise 7.3   Solution on page 153

Create a new macro \minitable which produces the following output:

\minitable{My heading}{In this tiny tabular, there is only a heading
  and some text below which has a width of ten centimeters.}

| **My heading** |
| In this tiny tabular, there is only a heading and some text below which has a width of ten centimeters. |

```
\begin{texercise}{macro_threeparam}
\begin{tcboutputlisting}
\newcommand{\synop}[3]{%
  \begin{tabular}{@{}p{(\linewidth-\tabcolsep*2-\arrayrulewidth)/2}|%
    p{(\linewidth-\tabcolsep*2-\arrayrulewidth)/2}@{}}\hline
  \multicolumn{2}{c}{\bfseries #1}\\\hline
  \multicolumn{1}{c|}{\itshape English}&
  \multicolumn{1}{c}{\itshape German}\\\hline
  #2 & #3
  \end{tabular}}
\end{tcboutputlisting}
\tcbuselistingtext%
Create a new macro \verb+\synop+ which typesets a synoptic text according
to the following example. Base your macro on a tabular which takes the
total line width.\par\smallskip
\begin{tcbwritetemp}
\synop{Neil Armstrong}%
{That's one small step for a man, one giant leap for mankind.}%
{Das ist ein kleiner Schritt f\"{u}r einen Mann,
  ein riesiger Sprung f\"{u}r die Menschheit.}
\end{tcbwritetemp}
\tcbusetemplisting\par\smallskip\tcbusetemp%
\end{texercise}
```

Exercise 7.4

Create a new macro `\synop` which typesets a synoptic text according to the following example. Base your macro on a tabular which takes the total line width.

```
\synop{Neil Armstrong}%
{That's one small step for a man, one giant leap for mankind.}%
{Das ist ein kleiner Schritt f\"{u}r einen Mann,
  ein riesiger Sprung f\"{u}r die Menschheit.}
```

Neil Armstrong	
English	*German*
That's one small step for a man, one giant leap for mankind.	Das ist ein kleiner Schritt für einen Mann, ein riesiger Sprung für die Menschheit.

Now, we give a list of all exercises with:

```
\tcblistof[\subsection]{exam}{List of Exercises%
  \label{listofexercises}}
```

7.9 List of Exercises

7.10 Solutions for the given LATEX Exercises

For all solutions, a macro `\processsol` was written to the file `\jobname.sol`. Now, we need a definition for this macro to use the solutions.

```
% \usepackage{hyperref} % for \phantomsection
\newcommand{\processsol}[2]{%
  \tcbinputlisting{texercisestyle,listing only,
    phantom={\phantomsection\label{sol:#2}},%
    title={Solution for Exercise \ref{exe:#2} on page \pageref{exe:#2}},
    listing file={\solfile{#1}}}}
```

The loading of all solutions is done by:

```
\inputsol
```

With this, we get:

Solution for Exercise 7.1 on page 150

```
\begin{tabular}{|p{3cm}|p{3cm}|p{3cm}|p{3cm}|}\hline
\multicolumn{4}{|c|}{\bfseries\itshape Das alte Italien}\\\hline
\multicolumn{2}{|c|}{\bfseries Antike} &
\multicolumn{2}{c|}{\bfseries Mittelalter}\\\hline
\multicolumn{1}{|c|}{\itshape Republik}&
\multicolumn{1}{c|}{\itshape Kaiserreich}&
\multicolumn{1}{c|}{\itshape Franken}&
\multicolumn{1}{c|}{\itshape Teilstaaten}\\\hline
In den Zeiten der r\"{o}mischen Republik standen dem Staat jeweils zwei
Konsuln vor, deren Machtbefugnisse identisch waren. &
Das r\"{o}mische Kaiserreich wurde von einem Alleinherrscher, dem Kaiser,
regiert.
& In der V\"{o}lkerwanderungszeit \"{u}bernahmen die Goten und sp\"{a}ter die
Franken die Vorherrschaft.
& Im sp\"{a}teren Mittelalter regierten F\"{u}rsten einen Fleckenteppich
von Einzelstaaten.\\\hline
\end{tabular}
```

Solution for Exercise 7.2 on page 151

```
\newcommand{\headingline}[1]{%
  \begin{center}\Large\bfseries #1\end{center}}
```

Solution for Exercise 7.3 on page 151

```
\newcommand{\minitable}[2]{%
  \begin{center}\begin{tabular}{p{10cm}}\hline%
  \multicolumn{1}{c}{\bfseries#1}\\\hline%
  #2\\\hline%
  \end{tabular}\end{center}}
```

```
\newcommand{\synop}[3]{%
  \begin{tabular}{@{}p{(\linewidth-\tabcolsep*2-\arrayrulewidth)/2}|%
    p{(\linewidth-\tabcolsep*2-\arrayrulewidth)/2}@{}}\hline
  \multicolumn{2}{c}{\bfseries #1}\\\hline
  \multicolumn{1}{c|}{\itshape English}&
  \multicolumn{1}{c}{\itshape German}\\\hline
  #2 & #3
  \end{tabular}}
```

8 Library ▣ **theorems**

The library is loaded by a package option or inside the preamble by:

```
\tcbuselibrary{theorems}
```

This also loads the package amsmath.

8.1 Macros of the Library

\newtcbtheorem[⟨*init options*⟩]{⟨*name*⟩}{⟨*display name*⟩}{⟨*options*⟩}{⟨*prefix*⟩}

Creates a new environment ⟨*name*⟩ based on **tcolorbox** to frame a (mathematical) theorem. The ⟨*display name*⟩ is used in the title line with a number, e. g. «Theorem 5.1». The ⟨*options*⟩ are given to the underlying **tcolorbox** to control the appearance. The ⟨*init options*⟩ allow to set up automatic numbering, see Section 4 from page 65.

The new environment ⟨*name*⟩ takes one optional and two mandatory parameters. The optional parameter supplements the options and should be used only in rare cases. The first mandatory parameter is the title text for the theorem and the second mandatory parameter is a ⟨*marker*⟩. The theorem is automatically labeled with ⟨*prefix*⟩:⟨*marker*⟩.

Definition in the preamble:
```
\newtcbtheorem[number within=section]{mytheo}{My Theorem}%
  {colback=green!5,colframe=green!35!black,fonttitle=\bfseries}{th}
```

```
\begin{mytheo}{This is my title}{theoexample}
  This is the text of the theorem. The counter is automatically assigned and,
  in this example, prefixed with the section number. This theorem is numbered with
  \ref{th:theoexample} and is given on page \pageref{th:theoexample}.
\end{mytheo}
```

> **My Theorem 8.1: This is my title**
>
> This is the text of the theorem. The counter is automatically assigned and, in this example, prefixed with the section number. This theorem is numbered with 8.1 and is given on page 155.

```
\begin{mytheo}[label=myownlabel]{This is my title}{}
  The label parameter can be left empty without \LaTeX\ error.
  Or you may use an own label to reference Theorem \ref{myownlabel}.
\end{mytheo}
```

> **My Theorem 8.2: This is my title**
>
> The label parameter can be left empty without LATEX error. Or you may use an own label to reference Theorem 8.2.

```
\begin{mytheo}{}{}
  The title can also be left empty without problem. Note that the ':'
  vanished magically.
\end{mytheo}
```

> **My Theorem 8.3**
>
> The title can also be left empty without problem. Note that the ':' vanished magically.

\renewtcbtheorem[⟨*init options*⟩]{⟨*name*⟩}{⟨*display name*⟩}{⟨*options*⟩}{⟨*prefix*⟩}

Operates like \newtcbtheorem^{→P.155}, but based on \renewenvironment instead of \newenvironment. An existing environment is redefined.

\tcbmaketheorem{⟨*name*⟩}{⟨*display name*⟩}{⟨*options*⟩}{⟨*counter*⟩}{⟨*prefix*⟩}

> \newtcbtheorem^{→P.155} supersedes this macro.

Creates a new environment ⟨*name*⟩ based on tcolorbox to frame a (mathematical) theorem. The ⟨*display name*⟩ is used in the title line with a number, e.g. «Theorem 5.1». The ⟨*options*⟩ are given to the underlying tcolorbox to control the appearance. The ⟨*counter*⟩ is used for automatic numbering. The new environment ⟨*name*⟩ takes one optional and two mandatory parameters. The optional parameter supplements the options and should be used only in rare cases. The first mandatory parameter is the title text for the theorem and the second mandatory parameter is a ⟨*marker*⟩. The theorem is automatically labeled with ⟨*prefix*⟩:⟨*marker*⟩.

\tcboxmath[⟨*options*⟩]{⟨*mathematical box content*⟩}

Creates a tcolorbox^{→P.7} which is fitted to the width of the given ⟨*mathematical box content*⟩. This box is intended to be applied as part of a larger formula and may be used as replacement for the \boxed macro of amsmath.

```
\begin{equation}
\tcbset{fonttitle=\scriptsize}
\tcboxmath[colback=LightBlue!25!white,colframe=blue]{ a^2 = 16 }
\quad \Rightarrow \quad
\tcboxmath[colback=Salmon!25!white,colframe=red,title=Implication]%
  { a = 4 ~\vee~ a=-4. }
\end{equation}
```

$$a^2 = 16 \quad \Rightarrow \quad \boxed{a = 4 \ \vee \ a = -4.} \tag{3}$$

\tcbhighmath[⟨*options*⟩]{⟨*mathematical box content*⟩}

This is a special case of the **\tcboxmath**[→ P. 156] macro which uses the style **/tcb/highlight math**[→ P. 159]. It is intended to provide context sensitive highlighting of formula parts. The color settings via **/tcb/highlight math style**[→ P. 159] may be different inside theorems or other colored areas and outside.

```
\tcbset{myformula/.style={colback=yellow!10!white,colframe=red!50!black,
  highlight math style={colback=LightBlue!50!white,colframe=Navy}}}

\begin{align}
  \tcbhighmath{\sum\limits_{n=1}^{\infty} \frac{1}{n}} &= \infty.\\
  \int x^2 ~\text{d}x                      &= \frac13 x^3 + c.
\end{align}

\begin{tcolorbox}[ams align,myformula]
  \tcbhighmath{\sum\limits_{n=1}^{\infty} \frac{1}{n}} &= \infty.\\
  \int x^2 ~\text{d}x                      &= \frac13 x^3 + c.
\end{tcolorbox}
```

$$\boxed{\sum_{n=1}^{\infty} \frac{1}{n}} = \infty. \tag{4}$$

$$\int x^2 ~\mathrm{d}x = \frac{1}{3}x^3 + c. \tag{5}$$

$$\boxed{\sum_{n=1}^{\infty} \frac{1}{n}} = \infty. \tag{6}$$

$$\int x^2 ~\mathrm{d}x = \frac{1}{3}x^3 + c. \tag{7}$$

8.2 Option Keys of the Library

/tcb/separator sign=⟨*sign*⟩ (no default, initial :)

The given ⟨*sign*⟩ is used inside the title text of a theorem as separater between display name combined with number and the specific title text. It is omitted, if there is no specific title text.

```
% \usepackage{amssymb}
\newtcbtheorem[use counter from=mytheo]{sometheorem}{Theorem}%
  {colback=white,colframe=red!50!black,fonttitle=\bfseries,
   separator sign={\ $\blacktriangleright$}}{theo}
\begin{sometheorem}{My example}{}
My theorem text.
\end{sometheorem}
```

> **Theorem 8.4 ▶ My example**
>
> My theorem text.

/tcb/separator sign colon (style, no value, initially set)

Sets /tcb/separator sign to the default colon : sign.

/tcb/separator sign dash (style, no value)

Sets /tcb/separator sign to an en-dash sign.

```
\newtcbtheorem[use counter from=mytheo]{sometheorem}{Theorem}%
  {colback=white,colframe=red!50!black,fonttitle=\bfseries,
   separator sign dash}{theo}
\begin{sometheorem}{My example}{}
My theorem text.
\end{sometheorem}
```

> **Theorem 8.5 – My example**
>
> My theorem text.

/tcb/theorem={⟨*display name*⟩}{⟨*counter*⟩}{⟨*title*⟩}{⟨*marker*⟩} (no default)

This key is internally used by \tcbmaketheorem[→P. 156], but can be used directly in a **tcolorbox** for a more flexible approach. The ⟨*display name*⟩ is used together with the increased ⟨*counter*⟩ value and the ⟨*title*⟩ for the title line of the box. Additionally, a \label with the given ⟨*marker*⟩ is created.

```
% \newcounter{texercise}%  preamble
\begin{tcolorbox}[colback=green!10,colframe=green!50!black,arc=4mm,
                  theorem={Test}{texercise}{Direct usage}{myMarker}]
Here, we see the test \ref{myMarker}.
\end{tcolorbox}
```

> Test 1: Direct usage
>
> Here, we see the test 1.

For a common appearance inside the document, the key **theorem** should not be used directly as in the example above, but as part of a new environment created by hand or using \tcbmaketheorem[→P. 156] or using its successor \newtcbtheorem[→P. 155].

`/tcb/highlight math` (style, no value)

Predefined style which is used for `\tcbhighmath`[→P.157]. It can be changed comfortable with `/tcb/highlight math style`.

`/tcb/highlight math style`=⟨*style definition*⟩ (style, no default)

Changes the definition for `/tcb/highlight math` to the given ⟨*style definition*⟩. See `\tcbhighmath`[→P.157] for an example.

`/tcb/math upper` (style, no value)

Sets the upper part to mathematical mode with font `\displaystyle`.

`/tcb/math lower` (style, no value)

Sets the lower part to mathematical mode with font `\displaystyle`.

`/tcb/math` (style, no value)

Sets the upper part *and* lower part to mathematical mode with font `\displaystyle`.

```
\begin{tcolorbox}[math,colback=yellow!10!white,colframe=red!50!black]
  \sum\limits_{n=1}^{\infty} \frac{1}{n} = \infty.
\end{tcolorbox}
```

$$\sum_{n=1}^{\infty} \frac{1}{n} = \infty.$$

The following styles are only tested to work with the original `amsmath` environments. If e.g. the `equation` environment is redefined as `gather`, then `/tcb/ams equation` should / could not be used. Obviously, you are encouraged to use `/tcb/ams gather`[→ P. 162] in this case.

/tcb/ams equation upper   (style, no value)
Adds an `amsmath` `equation` environment to the begin and to the end if the upper part.

/tcb/ams equation lower   (style, no value)
Adds an `amsmath` `equation` environment to the begin and to the end if the lower part.

/tcb/ams equation   (style, no value)
Adds an `amsmath` `equation` environment to the begin and to the end if the upper *and* lower part.

```
\begin{tcolorbox}[ams equation,colback=yellow!10!white,colframe=red!50!black]
  \sum\limits_{n=1}^{\infty} \frac{1}{n} = \infty.
\end{tcolorbox}
```

$$\sum_{n=1}^{\infty} \frac{1}{n} = \infty. \tag{8}$$

/tcb/ams equation* upper   (style, no value)
Adds an `amsmath` `equation*` environment to the begin and to the end if the upper part.

/tcb/ams equation* lower   (style, no value)
Adds an `amsmath` `equation*` environment to the begin and to the end if the lower part.

/tcb/ams equation*   (style, no value)
Adds an `amsmath` `equation*` environment to the begin and to the end if the upper *and* lower part.

```
\begin{tcolorbox}[ams equation*,colback=yellow!10!white,colframe=red!50!black]
  \sum\limits_{n=1}^{\infty} \frac{1}{n} = \infty.
\end{tcolorbox}
```

$$\sum_{n=1}^{\infty} \frac{1}{n} = \infty.$$

/tcb/ams align upper (style, no value)

Adds an `amsmath` `align` environment to the begin and to the end if the upper part.

/tcb/ams align lower (style, no value)

Adds an `amsmath` `align` environment to the begin and to the end if the lower part.

/tcb/ams align (style, no value)

Adds an `amsmath` `align` environment to the begin and to the end if the upper *and* lower part.

```
\begin{tcolorbox}[ams align,colback=yellow!10!white,colframe=red!50!black]
  \sum\limits_{n=1}^{\infty} \frac{1}{n} &= \infty.\\
  \int x^2 ~\text{d}x                    &= \frac13 x^3 + c.
\end{tcolorbox}
```

$$\sum_{n=1}^{\infty} \frac{1}{n} = \infty. \tag{9}$$

$$\int x^2 \ \mathrm{d}x = \frac{1}{3}x^3 + c. \tag{10}$$

/tcb/ams align* upper (style, no value)

Adds an `amsmath` `align*` environment to the begin and to the end if the upper part.

/tcb/ams align* lower (style, no value)

Adds an `amsmath` `align*` environment to the begin and to the end if the lower part.

/tcb/ams align* (style, no value)

Adds an `amsmath` `align*` environment to the begin and to the end if the upper *and* lower part.

```
\begin{tcolorbox}[ams align*,colback=yellow!10!white,colframe=red!50!black]
  \sum\limits_{n=1}^{\infty} \frac{1}{n} &= \infty.\\
  \int x^2 ~\text{d}x                    &= \frac13 x^3 + c.
\end{tcolorbox}
```

$$\sum_{n=1}^{\infty} \frac{1}{n} = \infty.$$

$$\int x^2 \ \mathrm{d}x = \frac{1}{3}x^3 + c.$$

/tcb/ams gather upper   (style, no value)

Adds an `amsmath` `gather` environment to the begin and to the end if the upper part.

/tcb/ams gather lower   (style, no value)

Adds an `amsmath` `gather` environment to the begin and to the end if the lower part.

/tcb/ams gather   (style, no value)

Adds an `amsmath` `gather` environment to the begin and to the end if the upper *and* lower part.

```
\begin{tcolorbox}[ams gather,colback=yellow!10!white,colframe=red!50!black]
  \sum\limits_{n=1}^{\infty} \frac{1}{n} = \infty.\\
  \int x^2 ~\text{d}x = \frac13 x^3 + c.
\end{tcolorbox}
```

$$\sum_{n=1}^{\infty} \frac{1}{n} = \infty. \tag{11}$$

$$\int x^2 \ \mathrm{d}x = \frac{1}{3}x^3 + c. \tag{12}$$

/tcb/ams gather* upper   (style, no value)

Adds an `amsmath` `gather*` environment to the begin and to the end if the upper part.

/tcb/ams gather* lower   (style, no value)

Adds an `amsmath` `gather*` environment to the begin and to the end if the lower part.

/tcb/ams gather*   (style, no value)

Adds an `amsmath` `gather*` environment to the begin and to the end if the upper *and* lower part.

```
\begin{tcolorbox}[ams gather*,colback=yellow!10!white,colframe=red!50!black]
  \sum\limits_{n=1}^{\infty} \frac{1}{n} = \infty.\\
  \int x^2 ~\text{d}x = \frac13 x^3 + c.
\end{tcolorbox}
```

$$\sum_{n=1}^{\infty} \frac{1}{n} = \infty.$$

$$\int x^2 \ \mathrm{d}x = \frac{1}{3}x^3 + c.$$

/tcb/ams nodisplayskip upper (style, no value)

Neutralizes the `\abovedisplayskip` of a following `align` or `gather` environment for the upper part. Note that the text content has to start with such a formula.

/tcb/ams nodisplayskip lower (style, no value)

Neutralizes the `\abovedisplayskip` of a following `align` or `gather` environment for the lower part. Note that the text content has to start with such a formula.

/tcb/ams nodisplayskip (style, no value)

Neutralizes the `\abovedisplayskip` of a following `align` or `gather` environment for the upper part *and* lower part. Note that the text content has to start with such a formula.

```
\begin{tcolorbox}[ams nodisplayskip,colback=yellow!10!white,colframe=red!50!black]
  \begin{gather}
  \sum\limits_{n=1}^{\infty} \frac{1}{n} = \infty.\\
  \int x^2 ~\text{d}x = \frac13 x^3 + c.
  \end{gather}
  And now for something completely different.
\end{tcolorbox}
```

$$\sum_{n=1}^{\infty} \frac{1}{n} = \infty. \tag{13}$$

$$\int x^2 \ \mathrm{d}x = \frac{1}{3}x^3 + c. \tag{14}$$

And now for something completely different.

New colored mathematical environments are easily created using `\newtcolorbox`^{→ P. 9}:

```
\newtcolorbox{mymath}{ams gather*,colback=yellow!10!white,colframe=red!50!black}

\begin{mymath}
  \sum\limits_{n=1}^{\infty} \frac{1}{n} = \infty.\\
  \int x^2 ~\text{d}x = \frac13 x^3 + c.
\end{mymath}
```

$$\sum_{n=1}^{\infty} \frac{1}{n} = \infty.$$

$$\int x^2 \ \mathrm{d}x = \frac{1}{3}x^3 + c.$$

All described options like `/tcb/ams gather upper`^{→ P. 162}, `/tcb/ams gather lower`^{→ P. 162}, `/tcb/ams gather`^{→ P. 162} are (partially) setting (overwritting) the keys `/tcb/before upper`^{→ P. 33}, `/tcb/after upper`^{→ P. 34}, `/tcb/before lower`^{→ P. 34}, `/tcb/after lower`^{→ P. 34}.

Therefore, e. g. `\tcbset{ams gather,before upper={\text{Pythagoras:}}}` produces an invalid result. For this case, you are invited to use
`\tcbset{ams gather,before upper app={\text{Pythagoras:}}}`,
see `/tcb/before upper app`^{→ P. 191}.

8.3 Examples for Definitions and Theorems

In the following, the application of `\tcbmaketheorem`[→P.156] to highlight mathematical definitions, theorems, or the like is demonstrated.

At first, additional `tcb` keys are created for the appearance of the colored boxes. It is assumed that theorems and corollaries should be identically colored. All following environments are numbered with a common counter, but this can be changed easily. Here, the counter output is supplemented by the subsection number.

Definition in the preamble:

```
\tcbset{
  defstyle/.style={fonttitle=\bfseries\upshape, fontupper=\slshape,
              arc=0mm, colback=blue!5!white,colframe=blue!75!black},
  theostyle/.style={fonttitle=\bfseries\upshape, fontupper=\slshape,
              colback=red!10!white,colframe=red!75!black},
}
\newtcbtheorem[number within=subsection]{Definition}{Definition}{defstyle}{def}
\newtcbtheorem[use counter from=Definition]{Theorem}{Theorem}{theostyle}{theo}
\newtcbtheorem[use counter from=Definition]{Corollary}{Corollary}{theostyle}{cor}
```

By `\newtcbtheorem`[→P.155], commonly numbered theorem environments are created now. `defstyle` and `theostyle` are used for the appearance.

Now, everything is prepared for the following examples.

```
The following theorem is numbered as Theorem \ref{theo:diffbarstetig} and
referenced with the marker \texttt{theo:diffbarstetig}.\bigskip

\begin{Theorem}{Differenzierbarkeit bedingt Stetigkeit, wobei diese Benennung
  zu Testzwecken ungew\"{o}hnlich lang ist}{diffbarstetig}%
  Eine Funktion $f:I\to\mathbb{R}$ ist in $x_0\in I$ stetig, wenn $f$ in
  $x_0$ differenzierbar ist.
\end{Theorem}
```

The following theorem is numbered as Theorem 8.3.1 and referenced with the marker `theo:diffbarstetig`.

> **Theorem 8.3.1: Differenzierbarkeit bedingt Stetigkeit, wobei diese Benennung zu Testzwecken ungewöhnlich lang ist**
>
> *Eine Funktion $f : I \to \mathbb{R}$ ist in $x_0 \in I$ stetig, wenn f in x_0 differenzierbar ist.*

```
The following definition is numbered as Definition \ref{def:diffbarkeit} and
referenced with the marker \texttt{def:diffbarkeit}.\bigskip

\begin{Definition}{Differenzierbarkeit}{diffbarkeit}
  Eine Funktion $f:~I\to\mathbb{R}$ auf einem Intervall $I$ hei\ss{}t in
  $x_0\in I$ differenzierbar oder linear approximierbar,
  wenn der Grenzwert
  \begin{equation*}
  \lim\limits_{x\to x_0}\frac{f(x)-f(x_0)}{x-x_0}=
  \lim\limits_{h\to 0}\frac{f(x_0+h)-f(x_0)}{h}
  \end{equation*}
  existiert. Bei Existenz hei\ss{}t dieser Grenzwert Ableitung
  oder Differentialquotient von $f$ in $x_0$ und man
  schreibt f\"{u}r ihn
  \begin{equation*}
  f'(x_0)\quad\text{oder}\quad\frac{df}{dx}(x_0).
  \end{equation*}
\end{Definition}
```

The following definition is numbered as Definition 8.3.2 and referenced with the marker `def:diffbarkeit`.

> **Definition 8.3.2: Differenzierbarkeit**
>
> *Eine Funktion $f : I \to \mathbb{R}$ auf einem Intervall I heißt in $x_0 \in I$ differenzierbar oder linear approximierbar, wenn der Grenzwert*
> $$\lim_{x\to x_0} \frac{f(x) - f(x_0)}{x - x_0} = \lim_{h\to 0} \frac{f(x_0 + h) - f(x_0)}{h}$$
> *existiert. Bei Existenz heißt dieser Grenzwert Ableitung oder Differentialquotient von f in x_0 und man schreibt für ihn*
> $$f'(x_0) \quad oder \quad \frac{df}{dx}(x_0).$$

```
The following corollary is numbered as Corollary \ref{cor:nullstellen} and
referenced with the marker \texttt{cor:nullstellen}.\bigskip

\begin{Corollary}{Nullstellenexistenz}{nullstellen}
  Ist $f:[a,b]\to\mathbb{R}$ stetig und haben $f(a)$ und $f(b)$ entgegengesetzte
  Vorzeichen, also $f(a)f(b)<0$, so besitzt $f$ eine Nullstelle $x_0\in]a,b[$,
  also $f(x_0)=0$.
\end{Corollary}
```

The following corollary is numbered as Corollary 8.3.3 and referenced with the marker `cor:nullstellen`.

> **Corollary 8.3.3: Nullstellenexistenz**
>
> *Ist $f : [a,b] \to \mathbb{R}$ stetig und haben $f(a)$ und $f(b)$ entgegengesetzte Vorzeichen, also $f(a)f(b) < 0$, so besitzt f eine Nullstelle $x_0 \in]a,b[$, also $f(x_0) = 0$.*

```
\begin{Theorem}[boxrule=2mm,toptitle=-1.5mm,bottomtitle=-1.5mm]{%
    Hinreichende Bedingung f\"{u}r Wendepunkte}{wendehinreichend}%
  $f$ sei eine auf einem Intervall $]a,b[$ dreimal stetig differenzierbare Funktion.
  Ist $f''(x_0)=0$ in $x_0\in]a,b[$ und $f'''(x_0)\ne 0$, so ist
  $(x_0,f(x_0))$ ein Wendepunkt von $f$.
\end{Theorem}
```

Theorem 8.3.4: Hinreichende Bedingung für Wendepunkte

f sei eine auf einem Intervall $]a,b[$ dreimal stetig differenzierbare Funktion. Ist $f''(x_0) = 0$ in $x_0 \in]a,b[$ und $f'''(x_0) \neq 0$, so ist $(x_0, f(x_0))$ ein Wendepunkt von f.

```
% \usepackage{varioref}
% \usepackage{cleveref}
% \tcbuselibrary{skins}
\newtcbtheorem[use counter from=Definition]{YetAnotherTheorem}{Theorem}%
  {label type=theorem,enhanced,frame hidden,
  boxrule=2mm,titlerule=0mm,toptitle=1mm,bottomtitle=1mm,
  fonttitle=\bfseries\large,fontupper=\normalsize,
  coltitle=green!35!black,colbacktitle=green!15!white,
  colback=green!50!yellow!15!white,borderline={1pt}{0pt}{green!25!blue},
  }{theo}

\begin{YetAnotherTheorem}{Mittelwertsatz f\"{u}r $n$ Variable}{meanvaluetheorem}%
  Es sei $n\in\mathbb{N}$, $D\subseteq\mathbb{R}^n$ eine offene Menge und
  $f\in C^{1}(D,\mathbb{R})$. Dann gibt es auf jeder Strecke
  $[x_0,x]\subset D$ einen Punkt $\xi\in[x_0,x]$, so dass gilt
  \begin{equation*}
  f(x)-f(x_0) = \operatorname{grad} f(\xi)^{\top}(x-x_0)
  \end{equation*}
\end{YetAnotherTheorem}

\medskip
Here, |cleveref| support is used to reference \Cref{theo:meanvaluetheorem}
on \Cpageref{theo:meanvaluetheorem}. This \namecref{theo:meanvaluetheorem}
can also be referenced by |\Vref| resulting in \Vref{theo:meanvaluetheorem}.
```

Theorem 8.3.5: Mittelwertsatz für n Variable

Es sei $n \in \mathbb{N}$, $D \subseteq \mathbb{R}^n$ eine offene Menge und $f \in C^1(D, \mathbb{R})$. Dann gibt es auf jeder Strecke $[x_0, x] \subset D$ einen Punkt $\xi \in [x_0, x]$, so dass gilt

$$f(x) - f(x_0) = \operatorname{grad} f(\xi)^{\top}(x - x_0)$$

Here, `cleveref` support is used to reference Theorem 8.3.5 on Page 166. This theorem can also be referenced by `\Vref` resulting in Theorem 8.3.5.

Note that `/tcb/label type`[→P. 62] was used in the example above to feed `cleveref` [4] with the needed name information.

> Here, using |\Vref| resulting in \Vref{theo:meanvaluetheorem} is more interesting\ldots

Here, using **\Vref** resulting in Theorem 8.3.5 on the previous page is more interesting. . .

```
% \tcbuselibrary{skins}
\newcbtheorem[use counter from=Definition]{YetAnotherTheorem}{Theorem}%
  {enhanced,arc=0mm,outer arc=0mm,
  boxrule=0mm,toprule=1mm,bottomrule=1mm,left=1mm,right=1mm,
  titlerule=0mm,toptitle=0mm,bottomtitle=1mm,top=0mm,
  colframe=red!50!black,colback=red!5!white,coltitle=red!50!black,
  title style={top color=yellow!50!white,bottom color=red!5!white,
    middle color=yellow!50!white},
  fonttitle=\bfseries\sffamily\normalsize,fontupper=\normalsize\itshape,
  }{theo}

\begin{YetAnotherTheorem}{Mittelwertsatz f\"{u}r $n$ Variable}{mittelwertsatz_n2}%
  Es sei $n\in\mathbb{N}$, $D\subseteq\mathbb{R}^n$ eine offene Menge und
  $f\in C^{1}(D,\mathbb{R})$. Dann gibt es auf jeder Strecke
  $[x_0,x]\subset D$ einen Punkt $\xi\in[x_0,x]$, so dass gilt
  \begin{equation*}
  f(x)-f(x_0) = \operatorname{grad} f(\xi)^{\top}(x-x_0)
  \end{equation*}
\end{YetAnotherTheorem}
```

Theorem 8.3.6: Mittelwertsatz für n Variable

Es sei $n \in \mathbb{N}$, $D \subseteq \mathbb{R}^n$ eine offene Menge und $f \in C^1(D, \mathbb{R})$. Dann gibt es auf jeder Strecke $[x_0, x] \subset D$ einen Punkt $\xi \in [x_0, x]$, so dass gilt

$$f(x) - f(x_0) = \operatorname{grad} f(\xi)^{\top}(x - x_0)$$

You need more attention for your theorems? Here, you are ...

```
% tcbuselibrary{skins}  % preamble
\begin{Theorem}[enhanced,
    fuzzy halo=3mm with yellow,
    fuzzy halo=2mm with red,
    fuzzy halo=1mm with yellow,
    watermark color=red!35!white,
    watermark text={Overacting\\Fundamental Theorem}]%
  {Fundamental Theorem of Theorems}{fundamental}%
  \lipsum[1-2]
\end{Theorem}
```

Theorem 8.3.7: Fundamental Theorem of Theorems

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

Using \newtcbtheorem is a convenient way to generate a new theorem type. On the other hand, it enforces to use a titled `tcolorbox`. If you prefer to have an embedded title, you may create a new theorem type with the more flexible \newtcolorbox macro:

```
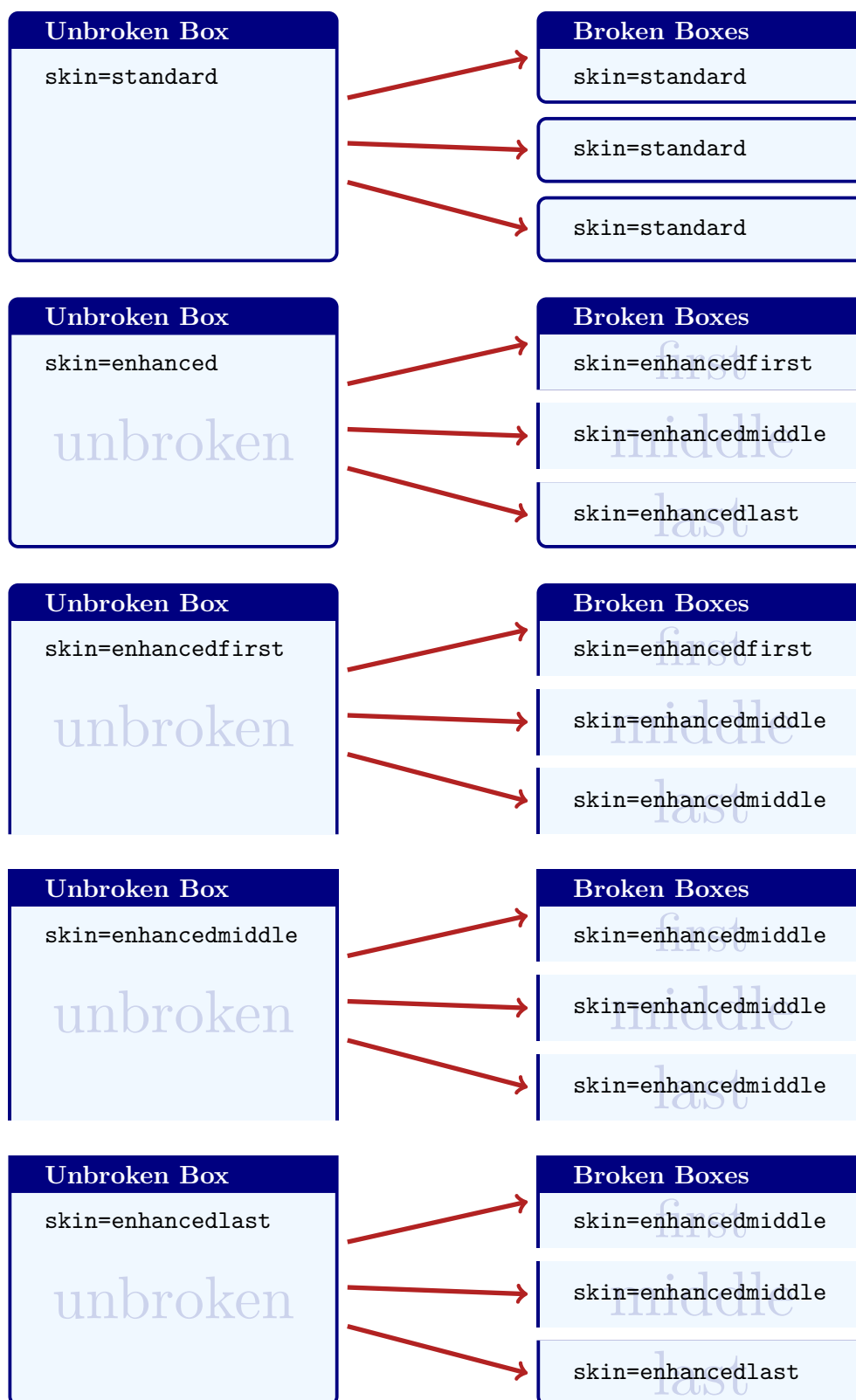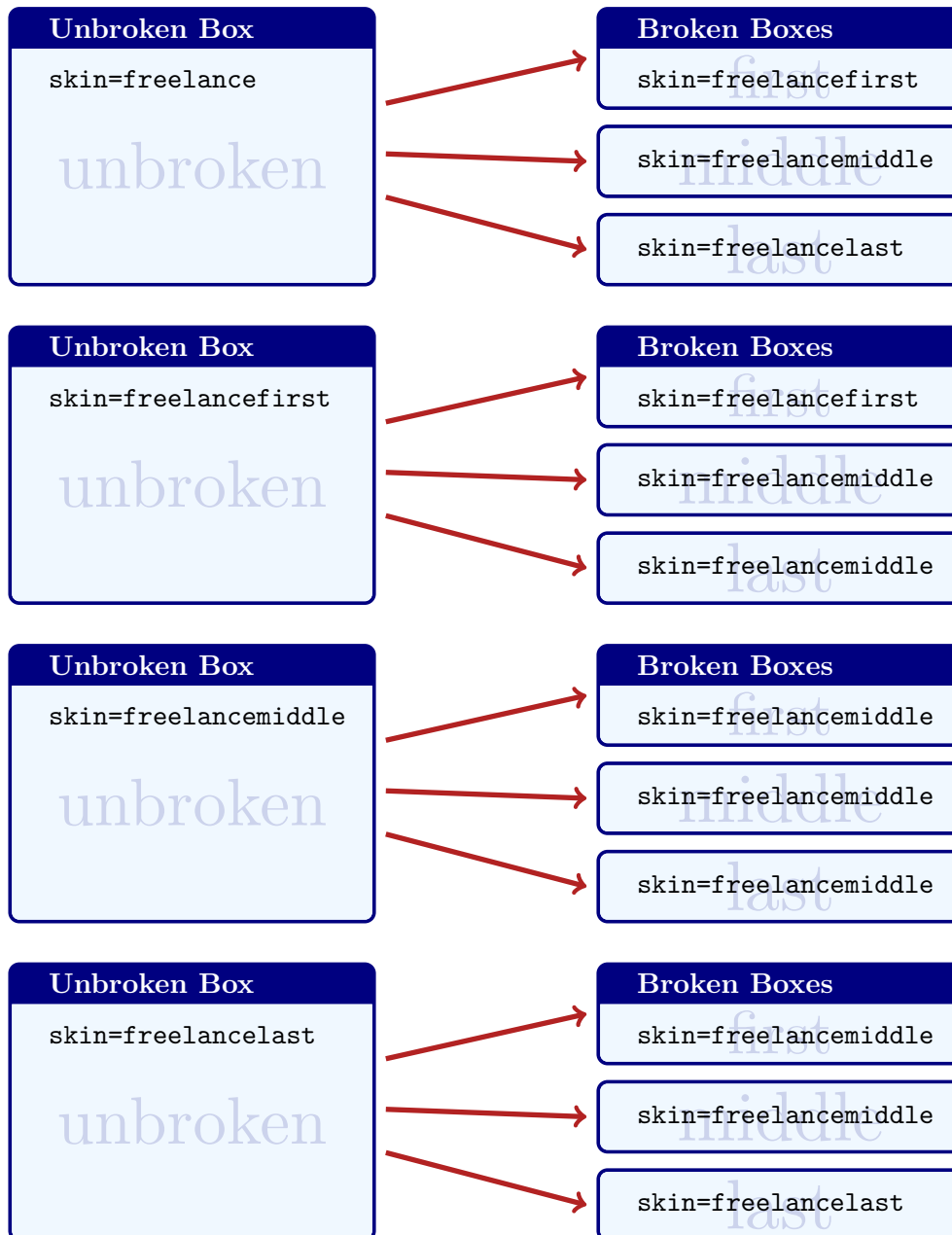% \tcbuselibrary{skins}
\newtcolorbox[use counter from=Definition]{YetAnotherTheorem}[3][%
  enhanced,colframe=blue!50!black,colback=yellow!20!white,
  drop fuzzy shadow=blue!50!black!50!white,boxrule=0.4pt,
  fontupper=\itshape]{%
  before upper={\textcolor{red!50!black}{\upshape\bfseries
    Theorem~\thetcbcounter\ (#2)}\quad},
  list entry={\numberline{\thetcbcounter}#2},%
  label={theo:#3},#1]

\begin{YetAnotherTheorem}{Mittelwertsatz f\"{u}r $n$ Variable}{mittelwertsatz_n3}%
  Es sei $n\in\mathbb{N}$, $D\subseteq\mathbb{R}^n$ eine offene Menge und
  $f\in C^{1}(D,\mathbb{R})$. Dann gibt es auf jeder Strecke
  $[x_0,x]\subset D$ einen Punkt $\xi\in[x_0,x]$, so dass gilt
  \begin{equation*}
  f(x)-f(x_0) = \operatorname{grad} f(\xi)^{\top}(x-x_0)
  \end{equation*}
\end{YetAnotherTheorem}
```

Theorem 8.3.8 (Mittelwertsatz für n Variable)   *Es sei $n \in \mathbb{N}$, $D \subseteq \mathbb{R}^n$ eine offene Menge und $f \in C^1(D, \mathbb{R})$. Dann gibt es auf jeder Strecke $[x_0, x] \subset D$ einen Punkt $\xi \in [x_0, x]$, so dass gilt*

$$f(x) - f(x_0) = \operatorname{grad} f(\xi)^{\top} (x - x_0)$$

9 Library [LIB] breakable

The library is loaded by a package option or inside the preamble by:

```
\tcbuselibrary{breakable}
```

9.1 Technical Overview

The library [LIB] breakable supports the automatic breaking of a `tcolorbox`. This feature is enabled by `/tcb/breakable`[→P. 171] and disabled by `/tcb/unbreakable`[→P. 172].

If a `tcolorbox` is set to be `/tcb/breakable`[→P. 171], then the following algorithm is executed:

1. The box content is read to a box register similar but not identical to the unbreakable case.

2. If the total box fits into the current page, it is shipped out visibly unbroken and the algorithm stops.

   Unbroken Box

   The box. *unbroken*

3. Otherwise, it is checked if at least `/tcb/lines before break`[→P. 172] of the upper box can be placed on the current page. If not, a page break is inserted and the algorithm goes back to Step 2.

4. Now, the *break sequence* starts. The upper box part or the lower box part is split such that it fits into the current page. The fitting part is named *first part* of the *break sequence* and shipped out.

   Broken Box

   The box. *first*

5. If the remaining content of the total box fits into the current page, the algorithm continues with Step 7, else with Step 6.

6. The upper box part or the lower box part is split such that it fits into the current page. The fitting part is named *middle part* of the *break sequence* and shipped out. Then, the algorithm goes back to Step 5.

   The box. *middle*

7. The remaining part is named *last part* of the *break sequence* and shipped out. The algorithm stops.

   The box. *last*

The algorithm takes care that the optional segmentation line never appears at the end of a box. The optional lower box part is also checked to have at least `/tcb/lines before break`[→P. 172].

In principal, all boxes of the *break sequence* share the same geometric parameters. The differences are:

- The given `/tcb/before`$^{\rightarrow P.\,50}$ and `/tcb/after`$^{\rightarrow P.\,50}$ values are used only before the *first* and after the *last* part of the *break sequence*.

- A special behavior between the parts of the *break sequence* can be given by `/tcb/toprule at break`$^{\rightarrow P.\,173}$, `/tcb/bottomrule at break`$^{\rightarrow P.\,173}$, `/tcb/enlarge top at break by`$^{\rightarrow P.\,174}$, and `/tcb/enlarge bottom at break by`$^{\rightarrow P.\,174}$.

- The `/tcb/skin`$^{\rightarrow P.\,71}$ decides *how* the *first*, *middle*, and *last* part look like. Actually, every part type has its own skin given by the options `/tcb/skin first`$^{\rightarrow P.\,71}$, `/tcb/skin middle`$^{\rightarrow P.\,71}$, and `/tcb/skin last`$^{\rightarrow P.\,71}$. Typically, these options are set automatically by the main skin, see Subsection 9.5 from page 175.

9.2   Limitations and Known Bugs

- The box content is a TeX `\vbox` register which has a restricted capacity. Therefore, you cannot place hundreds of pages inside a `tcolorbox`.

- You can nest an unbreakable `tcolorbox` inside another `tcolorbox`, even inside a breakable one. But you should not nest a breakable box inside a breakable box since this will give a mess. Inside a breakable box, the further breaking is disabled by default. If you really want to or have to use the keys `/tcb/breakable`$^{\rightarrow P.\,171}$ or `/tcb/unbreakable`$^{\rightarrow P.\,172}$ inside the content of an outer `tcolorbox`, you have to guard this with a TeX group.

- If your text content contains some text color changing commands, your color will not survive the break to the next box[3].

---

[3]The reason is that I am to dumb to catch the current color at the split point. If you know the trick, let me know.

9.3 Main Option Keys

/tcb/breakable (no value)

Allows the `tcolorbox` to be breakable. If the box is larger than the available space at the current page, the box is automatically broken and continued to the next next page. All sorts of `tcolorbox` can be made breakable. It depends on the skin how the breaking looks like. If you do not know better, use **/tcb/enhanced**^{→ P. 99} for breaking a box. The parts of the *break sequence* are numbered by the counter `tcbbreakpart`.

```
% \usepackage{lipsum}  % preamble
\tcbset{colback=red!5!white,colframe=red!75!black,
  watermark color=yellow!25!white,watermark text=\arabic{tcbbreakpart},
  fonttitle=\bfseries}

\begin{tcolorbox}[breakable,enhanced,title=My breakable box]
\lipsum[1-6]
\end{tcolorbox}
```

My breakable box

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

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, consectetuer 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 consectetuer.

Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

/tcb/unbreakable (no value, initially set)
Sets the `tcolorbox` to be unbreakable.

/tcb/title after break=⟨*text*⟩ (no default, initially empty)
The `/tcb/title`$^{\rightarrow\text{P. 11}}$ is used only for the *first* part of a *break sequence*. Use `title after break` to create a heading line with ⟨*text*⟩ as content for all following parts.

/tcb/notitle after break (no value, initially set)
Removes the title line or following parts in a *break sequence* if set before.

/tcb/adjusted title after break=⟨*text*⟩ (style, no default, initially unset)
Works like `/tcb/adjusted title`$^{\rightarrow\text{P. 11}}$ but applied to `/tcb/title after break`.

/tcb/lines before break=⟨*number*⟩ (no default, initially 2)
Assures that the given ⟨*number*⟩ of lines of the upper box part or the lower box part are placed before a break happens.

/tcb/enlargepage=⟨*length*⟩/⟨*length*⟩…/⟨*length*⟩ (no default, initially 0pt)
Inserts a `\enlargethispage{`⟨*length*⟩`}` to the pages of the break sequence, i.e. allows to enlarge (or shrink) partial boxes. The first ⟨*length*⟩ is applied to the first partial box, the second ⟨*length*⟩ is applied to the second partial box, and so on. The last ⟨*length*⟩ value is applied to all following partial boxes if any. Note that floating boxes will not be enlarged.

```
\begin{tcolorbox}[breakable,enlargepage=0mm/\baselineskip/2\baselineskip/0mm,...
```

The example code enlarged the second partial box by one line, the third partial box by two lines, and all following parts are not enlarged.

If an automated page break occures before the first partial box, the page enlargement is applied to the page before the first partial box *and* again to the page of the first partial box. Insert a manual break to prevent this.

In general, `enlargepage` should be used at the final stage of a document for fine-tuning only.

/tcb/shrink break goal=⟨*length*⟩ (no default, initially 0pt)
This is an emergency parameter if the break algorithm produces unpleasant breaks. It shrinks the goal height of the current box part by ⟨*length*⟩ which may result in smaller boxes. Never use negative values.

9.4 Option Keys for the Break Appearance

/tcb/toprule at break=⟨*length*⟩   (no default, initially 0.5mm)

Sets the line width of the top rule to ⟨*length*⟩ *if* the box is `/tcb/breakable`→P.171.
In this case, it is applied to *middle* and *last* parts in a break sequence. Note that
`/tcb/toprule`→P.17 overwrites this value if used afterwards.

/tcb/bottomrule at break=⟨*length*⟩   (no default, initially 0.5mm)

Sets the line width of the bottom rule to ⟨*length*⟩ *if* the box is `/tcb/breakable`→P.171.
In this case, it is applied to *first* and *middle* parts in a break sequence. Note that
`/tcb/bottomrule`→P.17 overwrites this value if used afterwards.

/tcb/topsep at break=⟨*length*⟩   (no default, initially 0mm)

Additional vertical space of ⟨*length*⟩ which is added at the top of *middle* and *last* parts in a
break sequence. In general, it is not advisable to change this value if these parts start with
a rule or a title.

/tcb/bottomsep at break=⟨*length*⟩   (no default, initially 0mm)

Additional vertical space of ⟨*length*⟩ which is added at the bottom of *first* and *middle* parts
in a break sequence. In general, it is not advisable to change this value if these parts end
with a rule.

/tcb/pad before break=⟨*length*⟩   (style, no default, initially 3.5mm)

Sets the total amount of vertical space after the text content and before the break point
to ⟨*length*⟩. This style sets `/tcb/toprule at break` to `0pt` and changes `/tcb/topsep at
break` as required. In general, it is not advisable to change this value if the *middle* and *last*
parts in a break sequence start with a rule or a title.

/tcb/pad after break=⟨*length*⟩   (style, no default, initially 3.5mm)

Sets the total amount of vertical space after the break point and before the text content to
⟨*length*⟩. This style sets `/tcb/bottomrule at break` to `0pt` and changes `/tcb/bottomsep
at break` as required. In general, it is not advisable to change this value if the *first* and
middle parts in a break sequence end with a rule.

/tcb/pad at break=⟨*length*⟩   (style, no default, initially 3.5mm)

Abbreviation for setting ⟨*length*⟩ to `/tcb/pad before break` and `/tcb/pad after break`.

```
% \usepackage{lipsum}  % preamble
\tcbset{colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[enhanced,breakable,pad at break=0mm,
  title={For this box, the pad space at the break point is set to 0mm}]
  \lipsum[1-2]
\end{tcolorbox}
```

For this box, the pad space at the break point is set to 0mm

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut,
placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero,
nonummy eget, consectetuer 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.

/tcb/enlarge top at break by=⟨*length*⟩ (no default, initially `0mm`)

Enlarges the bounding box distance to the top of the box by ⟨*length*⟩ *if* the box is `/tcb/breakable`^{→ P. 171}. In this case, it is applied to *middle* and *last* parts in a break sequence. `/tcb/enlarge top by`^{→ P. 51} overwrites this key.

/tcb/enlarge bottom at break by=⟨*length*⟩ (no default, initially `0mm`)

Enlarges the bounding box distance to the bottom of the box by ⟨*length*⟩ *if* the box is `/tcb/breakable`^{→ P. 171}. In this case, it is applied to *first* and *middle* parts in a break sequence. `/tcb/enlarge bottom by`^{→ P. 51} overwrites this key.

9.5 Break Sequence for the Skins

The following diagrams document the *break sequence* for different skins. Depending on the main skin of a `tcolorbox`, the actual skins of the *break sequence* parts are displayed.

Unbroken Box		Broken Boxes
skin=standard	→	skin=standard
	→	skin=standard
	→	skin=standard

Unbroken Box		Broken Boxes
skin=enhanced	→	skin=enhancedfirst
unbroken	→	skin=enhancedmiddle
	→	skin=enhancedlast

Unbroken Box		Broken Boxes
skin=enhancedfirst	→	skin=enhancedfirst
unbroken	→	skin=enhancedmiddle
	→	skin=enhancedmiddle

Unbroken Box		Broken Boxes
skin=enhancedmiddle	→	skin=enhancedmiddle
unbroken	→	skin=enhancedmiddle
	→	skin=enhancedmiddle

Unbroken Box		Broken Boxes
skin=enhancedlast	→	skin=enhancedmiddle
unbroken	→	skin=enhancedmiddle
	→	skin=enhancedlast

Unbroken Box	Broken Boxes
skin=freelance	skin=freelancefirst
unbroken	skin=freelancemiddle
	skin=freelancelast

Unbroken Box	Broken Boxes
skin=freelancefirst	skin=freelancefirst
unbroken	skin=freelancemiddle
	skin=freelancemiddle

Unbroken Box	Broken Boxes
skin=freelancemiddle	skin=freelancemiddle
unbroken	skin=freelancemiddle
	skin=freelancemiddle

Unbroken Box	Broken Boxes
skin=freelancelast	skin=freelancemiddle
unbroken	skin=freelancemiddle
	skin=freelancelast

Unbroken Box

`skin=bicolor`

unbroken

Broken Boxes

`skin=bicolorfirst` first

`skin=bicolormiddle` middle

`skin=bicolorlast` last

Unbroken Box

`skin=bicolorfirst`

unbroken

Broken Boxes

`skin=bicolorfirst` first

`skin=bicolormiddle` middle

`skin=bicolormiddle` last

Unbroken Box

`skin=bicolormiddle`

unbroken

Broken Boxes

`skin=bicolormiddle` first

`skin=bicolormiddle` middle

`skin=bicolormiddle` last

Unbroken Box

`skin=bicolorlast`

unbroken

Broken Boxes

`skin=bicolormiddle` first

`skin=bicolormiddle` middle

`skin=bicolorlast` last

Unbroken Box

skin=beamer

unbroken

Broken Boxes

skin=beamerfirst

skin=beamermiddle

skin=beamerlast

Unbroken Box

skin=beamerfirst

unbroken

Broken Boxes

skin=beamerfirst

skin=beamermiddle

skin=beamermiddle

Unbroken Box

skin=beamermiddle

unbroken

Broken Boxes

skin=beamermiddle

skin=beamermiddle

skin=beamermiddle

Unbroken Box

skin=beamerlast

unbroken

Broken Boxes

skin=beamermiddle

skin=beamermiddle

skin=beamerlast

Unbroken Box	Broken Boxes
skin=widget	skin=widgetfirst
unbroken	skin=widgetmiddle
	skin=widgetlast

Unbroken Box	Broken Boxes
skin=widgetfirst	skin=widgetfirst
unbroken	skin=widgetmiddle
	skin=widgetmiddle

Unbroken Box	Broken Boxes
skin=widgetmiddle	skin=widgetmiddle
unbroken	skin=widgetmiddle
	skin=widgetmiddle

Unbroken Box	Broken Boxes
skin=widgetlast	skin=widgetmiddle
unbroken	skin=widgetmiddle
	skin=widgetlast

frame: w=142.26378pt, h=108.12054pt
title: w=110.96584pt, h=7.65457pt
Unbroken Box
skin=draft
upper: w=110.96584pt, h=73.43599pt
interior: w=139.41853pt, h=90.50758pt

frame: w=142.26378pt, h=39.83368pt
title: w=110.96584pt, h=7.65457pt
Broken Boxes
upper: w=110.96584pt, h=5.14912pt
interior: w=139.41853pt, h=22.22072pt
frame: w=142.26378pt, h=28.45274pt
upper: w=110.96584pt, h=8.5359pt
interior: w=139.41853pt, h=25.6075pt
frame: w=142.26378pt, h=28.45274pt
skin=draft
upper: w=110.96584pt, h=8.5359pt
interior: w=139.41853pt, h=25.6075pt

9.6 Break by Hand (Faked Break)

Since the appearance of broken boxes is done by skins, it is quite easy to 'fake a break'. For this, you actually don't need the [LIB] breakable library at all.

```
\tcbset{enhanced,equal height group=fakedbreak,
  colback=LightGreen,colframe=DarkGreen,
  width=(\linewidth-6mm)/3,nobeforeafter,
  left=1mm,right=1mm,top=1mm,bottom=1mm,middle=1mm}
%
\begin{tcolorbox}[title=My broken box,skin=enhancedfirst]
This is a box which breaks from one column to another
\end{tcolorbox}\hfill
\begin{tcolorbox}[skin=enhancedmiddle]
column. I am sorry to say that this is a trick.
Nevertheless, you may use this trick for your
\end{tcolorbox}\hfill
\begin{tcolorbox}[skin=enhancedlast]
own purposes.
\end{tcolorbox}
```

My broken box

This is a box which breaks from one column to another

column. I am sorry to say that this is a trick. Nevertheless, you may use this trick for your

own purposes.

10 Library ⸎ fitting

The library is loaded by a package option or inside the preamble by:

```
\tcbuselibrary{fitting}
```

10.1 Macros of the Library

\tcboxfit[⟨*options*⟩]{⟨*box content*⟩}

Creates a colored box where the given ⟨*box content*⟩ is fitted to the width and height of the box. A `tcboxfit` has to have a fixed height. If no fixed height is given, a square box is constructed. In principal, most ⟨*options*⟩ for a `tcolorbox`[→ P. 7] can be used for **\tcboxfit** with some restrictions. A **\tcboxfit** cannot have a lower part and cannot be broken.

```
% \usepackage{lipsum}
\tcbset{colframe=blue!50!black,colback=red!10!white,
        boxsep=0pt,top=1mm,bottom=1mm,left=1mm,right=1mm,
        nobeforeafter,width=(\linewidth-2mm)/3}

\tcboxfit[height=8cm]{\lipsum[1]}\hfill
\tcboxfit[height=4cm]{\lipsum[1]}\hfill
\tcboxfit[height=2cm]{\lipsum[1]}

\medskip
\tcbset{width=(\linewidth-2mm)/2,colback=green!10!white,boxsep=1mm}
\tcboxfit[height=4cm]{\lipsum[2]}\hfill
\tcboxfit[height=4cm,title=With a title]{\lipsum[2]}
```

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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. 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.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

With a title

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.

\newtcboxfit[⟨*init options*⟩]{\\⟨*name*⟩}[⟨*number*⟩][⟨*default*⟩]{⟨*options*⟩}

Creates a new macro \\⟨*name*⟩ based on **\tcboxfit**^{→P.181}. Basically, **\newtcboxfit** operates like **\newcommand**. The new macro \\⟨*name*⟩ optionally takes ⟨*number*⟩+1 arguments, where ⟨*default*⟩ is the default value for the optional first argument. The ⟨*options*⟩ are given to the underlying **tcboxfit**. The ⟨*init options*⟩ allow to set up automatic numbering, see Section 4 from page 65.

```
\newtcboxfit{\mybox}{colback=red!5!white,
  colframe=red!75!black,width=4cm,
  height=1.5cm,center upper}

\mybox{This is my own box.}\par
\mybox{This is my own box with more text
       to be written.}
```

> This is my own box.

> This is my own box with more text to be written.

```
% \usepackage{lipsum}
\newtcboxfit{\mybox}[2]{colback=red!5!white,
  colframe=red!75!black,fonttitle=\bfseries,
  boxsep=1mm,left=0mm,right=0mm,top=0mm,
  bottom=0mm,center upper,valign=center,
  nobeforeafter,width=#1,height=#2}

\mybox{2.5cm}{1cm}{First box}%
\mybox{2.5cm}{1cm}{Second box with more text}\\
\mybox{5cm}{2cm}{Third box with text}\\
\mybox{5cm}{3cm}{\lipsum[1]}
```

> First box   Second box with more text

> Third box with text

> Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

```
% \usepackage{lipsum}
\newtcboxfit{\mybox}[2][]{colback=red!5!white,
  colframe=red!75!black,
  width=#2,height=#2/3*2,#1}

\mybox[colback=yellow]{5cm}%
  {\lipsum[2]}
```

> 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.

\renewtcboxfit[⟨*init options*⟩]{\\⟨*name*⟩}[⟨*number*⟩][⟨*default*⟩]{⟨*options*⟩}

Operates like **\newtcboxfit**, but based on **\renewcommand** instead of **\newcommand**. An existing macro is redefined.

\tcbfontsize{⟨*factor*⟩}

Selects a font size inside a tcolorbox which is scaled with the given ⟨*factor*⟩ relative to **\tcbfitdim**.

```
\tcbset{colback=red!5!white,colframe=red!75!black}
\begin{tcolorbox}[fit basedim=10pt]
  {\tcbfontsize{0.25} Very tiny,}\\
  {\tcbfontsize{0.5} Small,}\\
  {\tcbfontsize{1} Normal,}\\
  {\tcbfontsize{2} Large,}\\
  {\tcbfontsize{4} Huge.}
\end{tcolorbox}
```

> Very tiny,
> Small,
> Normal,
> Large,
> Huge.

10.2 Option Keys of the Library

The font size for the content of a box with fixed width and fixed height can be adjusted automatically. This is called the *fitbox capture mode*. Note that the fit control algorithm constructs a series of versions for the box and selects the 'best'. Therefore, the compilation time is quite longer than for a normal box. The algorithm will fail, if a different selected font size does not change the overall size of the box content. The \tcboxfit^{→P.181} macro uses this algorithm by default.

> ⚠ The fit control keys are only applicable to unbreakable boxes without a lower part. The box content should not change counters.

/tcb/fit (style, initially unset)
Sets the /tcb/capture^{→P.59} mode to `fitbox`, i.e. enables the font size adjustment algorithm. Thereby, a tcolorbox^{→P.7} acts like \tcboxfit^{→P.181} where the given ⟨*box content*⟩ is fitted to the width and height of the box. Therefore, the box has to have a fixed height. If no fixed height is given, a square box is constructed. The font dimension **\tcbfitdim** can also be used to adjust the margins of the box since a box with a tiny font may not need large margins. The number of constructed boxes is saved to the macro **\tcbfitsteps** for analysis.

```
% \usepackage{lipsum}
% \tcbuselibrary{skins}
\newtcolorbox{fitting}[2][]{fit,height=#2,boxsep=1pt,valign=center,
  top=0.4\tcbfitdim,bottom=0.4\tcbfitdim,
  left=0.75\tcbfitdim,right=0.75\tcbfitdim,
  enhanced,watermark text={\tcbfitsteps},
  colframe=blue!75!black,colback=white,#1}

\begin{fitting}{4cm}
\lipsum[1]
\end{fitting}

\begin{fitting}{2cm}
\lipsum[2]
\end{fitting}

\begin{fitting}{1cm}
\lipsum[3]
\end{fitting}
```

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

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.

/tcb/fit to=⟨*width*⟩ and ⟨*height*⟩ (style, initially unset)
Shortcut for using **/tcb/fit**^{→P.183} and setting the ⟨*width*⟩ and ⟨*height*⟩ values separately.

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[fit to=3cm and 2cm]
  This box content is fitted to the given
  dimensions.
\end{tcolorbox}
```

This box content is fitted to the given dimensions.

/tcb/fit to height=⟨*height*⟩ (style, initially unset)
Shortcut for using **/tcb/fit**^{→P.183} and setting the ⟨*height*⟩ value separately.

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[fit to height=2cm]
  This box content is fitted to the given
  height.
\end{tcolorbox}
```

This box content is fitted to the given height.

/tcb/fit basedim=⟨*length*⟩ (no default, initially 10pt)
Sets the starting font dimension for the font size adjustment algorithm to ⟨*length*⟩. The algorithm never enlarges this dimension.

```
\tcbset{colback=red!5!white,colframe=red!75!black}

\begin{tcolorbox}[fit to=4cm and 2cm]
  Too few words for the box.
\end{tcolorbox}

\begin{tcolorbox}[fit to=4cm and 2cm,
  fit basedim=50pt]
  Enough words for the box.
\end{tcolorbox}
```

Too few words for the box.

Enough words for the box.

/tcb/fit skip=⟨*real value*⟩ (no default, initially 1.2)
Sets the skip value of the selected font to ⟨*real value*⟩ times **\tcbfitdim**.

```
% \usepackage{lipsum}
\tcbset{colback=red!5!white,
  colframe=red!75!black,left=1mm,
  right=1mm,boxsep=0mm}

\begin{tcolorbox}[fit to=5cm and 4cm,
  fit skip=1.0  ]
  \lipsum[1]
\end{tcolorbox}
```

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

`/tcb/fit fontsize macros` (style, initially unset)

Redefines the standard LATEX font size macros \tiny, \scriptsize, \footnotesize, \small, \normalsize, \large, \Large, \LARGE, \huge, and \Huge, to set font sizes relative to the current **`\tcbfitdim`**. Note that the display skip values for mathematical formulas are respected by the redefined macros.

```
% \usepackage{lipsum}
\tcbset{colback=red!5!white,
  colframe=red!75!black,left=1mm,
  right=1mm,boxsep=0mm}

\begin{tcolorbox}[fit to height=4cm]
  {\Large\bfseries This text is
          not adapted:\par}
  \lipsum[2]
\end{tcolorbox}

\begin{tcolorbox}[fit to height=4cm,
  fit fontsize macros ]
  {\Large\bfseries This text is adapted:\par}
  \lipsum[2]
\end{tcolorbox}
```

This text is not adapted:
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.

This text is adapted:
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.

```
\tcbset{colback=red!5!white,
  colframe=red!75!black,left=1mm,
  right=1mm,boxsep=0mm}

\let\realHuge=\Huge

\begin{tcolorbox}[fit basedim=7pt,
  fontupper=\normalsize,
  fit fontsize macros]
The relative relative font size macros
are also usable without the
\textit{fit} algorithm.\par
{\Huge Adapted Huge} ---
{\realHuge Original Huge}
\end{tcolorbox}
```

The relative relative font size macros are also usable without the *fit* algorithm.

Adapted Huge — Original Huge

`/tcb/fit height plus`=⟨*dimension*⟩   (no default, initially `0pt`)

The box is allowed to enlarge the fixed height up to the given ⟨*dimension*⟩, before a font size fit is applied. An optional `/tcb/fit width plus` is tried after the height adaption.

```
% \usepackage{lipsum}
\tcbset{colback=red!5!white,colframe=red!75!black,left=1mm,top=1mm,bottom=1mm,
  right=1mm,boxsep=0mm,width=3cm,height=3cm,nobeforeafter}

\begin{tcolorbox}[fit]
This is a tcolorbox.
\end{tcolorbox}
\begin{tcolorbox}[fit,fit height plus=1cm]
This is a tcolorbox.
\end{tcolorbox}
\begin{tcolorbox}[fit]
\lipsum[2]
\end{tcolorbox}
\begin{tcolorbox}[fit,fit height plus=1cm]
\lipsum[2]
\end{tcolorbox}
```


`/tcb/fit width plus`=⟨*dimension*⟩   (no default, initially `0pt`)

The box is allowed to enlarge the fixed width up to the given ⟨*dimension*⟩, before a font size fit is applied. An optional `/tcb/fit height plus` is tried before the width adaption.

```
% \usepackage{lipsum}
\tcbset{colback=red!5!white,colframe=red!75!black,left=1mm,top=1mm,bottom=1mm,
  right=1mm,boxsep=0mm,width=3cm,height=3cm,nobeforeafter}

\begin{tcolorbox}[fit]
This is a tcolorbox.
\end{tcolorbox}
\begin{tcolorbox}[fit,fit width plus=1cm]
This is a tcolorbox.
\end{tcolorbox}
\begin{tcolorbox}[fit]
\lipsum[2]
\end{tcolorbox}
\begin{tcolorbox}[fit,fit width plus=1cm]
\lipsum[2]
\end{tcolorbox}
```


186

Typically, the optional title of a `tcolorbox` is not part of the fit operation. If a `/tcb/fit width plus` is applied, the title is also adapted to the new width. If counters are increased inside the title text, they may be increased more than one time. To avoid this, you are encouraged to use `/tcb/phantom`[→P.62] or `/tcb/step and label`[→P.62] to set counters or use automatic numbering, see Subsection 4.1 from page 65.

`/tcb/fit width from=`⟨*min*⟩ `to` ⟨*max*⟩   (style, no default)
    Sets the box width to ⟨*min*⟩ and allows the width to grow up to ⟨*max*⟩.

```
% \usepackage{lipsum}
\tcbset{colback=red!5!white,colframe=red!75!black,left=1mm,top=1mm,bottom=1mm,
  right=1mm,boxsep=0mm,height=4cm}

\begin{tcolorbox}[fit,width=\linewidth/2]
\lipsum[2]
\end{tcolorbox}\par
\begin{tcolorbox}[fit width from=\linewidth/2 to \linewidth]
\lipsum[2]
\end{tcolorbox}\par
```

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.

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.

/tcb/fit height from=⟨*min*⟩ **to** ⟨*max*⟩   (style, no default)

Sets the box height to ⟨*min*⟩ and allows the height to grow up to ⟨*max*⟩.

```
% \usepackage{lipsum}
\newtcolorbox{mybox}{colback=red!5!white,colframe=red!75!black,left=1mm,top=1mm,
  bottom=1mm,right=1mm,boxsep=0mm,width=4cm,nobeforeafter,
  fit height from=1cm to 8cm}

\begin{mybox}
This is a tcolorbox.
\end{mybox}
\begin{mybox}
This is a tcolorbox. This is a tcolorbox. This is a tcolorbox.
\end{mybox}
\begin{mybox}
\lipsum[2]
\end{mybox}
```

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.

This is a tcolorbox.

This is a tcolorbox. This is a tcolorbox. This is a tcolorbox.

The following options set control parameters for the fit algorithm. The algorithms is a bisection method which stops if certain stop conditions are fulfilled. The options should be seen as experimental and are likely to change in future versions, if necessary.

`/tcb/fit maxstep`=⟨*number*⟩   (no default, initially `20`)

   Sets the maximal step size for the font size adjustment algorithm. In normal situations, the algorithm stops before reaching the intial value of 20 steps. If the box content does not shrink, this value prevents an endless loop.

`/tcb/fit maxfontdiff`=⟨*dimension*⟩   (no default, initially `0.1pt`)

   The algorithm stops, if the font size is determined within a deviation of ⟨*dimension*⟩.

`/tcb/fit maxfontdiffgap`=⟨*dimension*⟩   (no default, initially `1pt`)

   The algorithm stops, if the number of lines is determined and the font size is determined within a deviation of ⟨*dimension*⟩.

`/tcb/fit maxwidthdiff`=⟨*dimension*⟩   (no default, initially `1pt`)

   The algorithm stops, if the (optionally) flexible box width is determined within a deviation of ⟨*dimension*⟩.

`/tcb/fit maxwidthdiffgap`=⟨*dimension*⟩   (no default, initially `10pt`)

   The algorithm stops, if the number of lines is determined and the (optionally) flexible box width is determined within a deviation of ⟨*dimension*⟩.

11 Library ⟦LIB⟧ hooks

The library is loaded by a package option or inside the preamble by:

```
\tcbuselibrary{hooks}
```

For the skin related options, the library ⟦LIB⟧ skins has to be loaded separately.

11.1 Concept of Hooks

A hook is a placeholder in some LaTeX code where additional code can be added. For example, the LaTeX macro `\AtBeginDocument` adds code to a hook which is placed at the begin of every document.

Several option keys of `tcolorbox` allow to provide some code which is added to specific places of a colored box. For example, `/tcb/before upper`$^{\to P.33}$ places code before the content of the upper part. A following usage of this key overwrites any prior settings.

The library ⟦LIB⟧ hooks extends `/tcb/before upper`$^{\to P.33}$ and several more existing keys to 'hookable' versions, e.g. `/tcb/before upper app`$^{\to P.191}$ and `/tcb/before upper pre`$^{\to P.191}$. The 'hookable' keys don't overwrite prior settings but either *app*end or *pre*pend the newly given code to the existing code.

The general naming convention (with some small exceptions) is:

- ⟨*option key*⟩ `app`: works like ⟨*option key*⟩ but *app*ends its code to the existing code.

- ⟨*option key*⟩ `pre`: works like ⟨*option key*⟩ but *pre*pends its code to the existing code.

If the original ⟨*option key*⟩ is used (again), all code will be overwritten. Therefore, the order of the option key usage is crucial.

```
% \usepackage{array,tabularx}
\newcolumntype{Y}{>{\raggedleft\arraybackslash}X}% see tabularx
\tcbset{enhanced,fonttitle=\bfseries\large,fontupper=\normalsize\sffamily,
  colback=yellow!10!white,colframe=red!50!black,colbacktitle=Salmon!30!white,
  coltitle=black,center title,
  tabularx={X||Y|Y|Y|Y||Y},% this sets 'before upper' and 'after upper'
  before upper app={Group & One & Two & Three & Four & Sum\\\hline\hline}  }

\begin{tcolorbox}[title=My table]
Red   & 1000.00 & 2000.00 &  3000.00 &  4000.00 & 10000.00\\\hline
Green & 2000.00 & 3000.00 &  4000.00 &  5000.00 & 14000.00\\\hline
Blue  & 3000.00 & 4000.00 &  5000.00 &  6000.00 & 18000.00\\\hline\hline
Sum   & 6000.00 & 9000.00 & 12000.00 & 15000.00 & 42000.00
\end{tcolorbox}
```

My table					
Group	One	Two	Three	Four	Sum
Red	1000.00	2000.00	3000.00	4000.00	10000.00
Green	2000.00	3000.00	4000.00	5000.00	14000.00
Blue	3000.00	4000.00	5000.00	6000.00	18000.00
Sum	6000.00	9000.00	12000.00	15000.00	42000.00

11.2 Box Content Additions

The following option keys extend the options given in Subsection 3.6 from page 33.

/tcb/before title app=⟨*code*⟩   (no default)
    Appends the given ⟨*code*⟩ to **/tcb/before title**^{→P. 33} *after* the color and font settings and *before* the content of the title.

/tcb/before title pre=⟨*code*⟩   (no default)
    Prepends the given ⟨*code*⟩ to **/tcb/before title**^{→P. 33} *after* the color and font settings and *before* the content of the title.

/tcb/after title app=⟨*code*⟩   (no default)
    Appends the given ⟨*code*⟩ to **/tcb/after title**^{→P. 33} *after* the content of the title.

/tcb/after title pre=⟨*code*⟩   (no default)
    Prepends the given ⟨*code*⟩ to **/tcb/after title**^{→P. 33} *after* the content of the title.

/tcb/before upper app=⟨*code*⟩   (no default)
    Appends the given ⟨*code*⟩ to **/tcb/before upper**^{→P. 33} *after* the color and font settings and *before* the content of the upper part.

/tcb/before upper pre=⟨*code*⟩   (no default)
    Prepends the given ⟨*code*⟩ to **/tcb/before upper**^{→P. 33} *after* the color and font settings and *before* the content of the upper part.

/tcb/after upper app=⟨*code*⟩   (no default)
    Appends the given ⟨*code*⟩ to **/tcb/after upper**^{→P. 34} *after* the content of the upper part.

/tcb/after upper pre=⟨*code*⟩   (no default)
    Prepends the given ⟨*code*⟩ to **/tcb/after upper**^{→P. 34} *after* the content of the upper part.

```
% \tcbuselibrary{theorems}
\tcbset{ams align,%  this sets 'before upper' and 'after upper'
  colback=yellow!10!white,colframe=red!50!black,
  before upper app={\frac{2}{\sqrt{2}}&=\sqrt{2}.\\},
  after upper pre={\\\sin\left(\frac{\pi}{2}\right)&=1.},
}

\begin{tcolorbox}
  \sum\limits_{n=1}^{\infty} \frac{1}{n} &= \infty.\\
  \int x^2 ~\text{d}x &= \frac13 x^3 + c.
\end{tcolorbox}
```

$$\frac{2}{\sqrt{2}} = \sqrt{2}. \tag{15}$$

$$\sum_{n=1}^{\infty} \frac{1}{n} = \infty. \tag{16}$$

$$\int x^2 \; \mathrm{d}x = \frac{1}{3}x^3 + c. \tag{17}$$

$$\sin\left(\frac{\pi}{2}\right) = 1. \tag{18}$$

/tcb/before lower app=⟨*code*⟩   (no default)
   Appends the given ⟨*code*⟩ to `/tcb/before lower`[→P. 34] *after* the color and font settings and
   before the content of the lower part.

/tcb/before lower pre=⟨*code*⟩   (no default)
   Prepends the given ⟨*code*⟩ to `/tcb/before lower`[→P. 34] *after* the color and font settings
   and *before* the content of the lower part.

/tcb/after lower app=⟨*code*⟩   (no default)
   Appends the given ⟨*code*⟩ to `/tcb/after lower`[→P. 34] *after* the content of the lower part.

/tcb/after lower pre=⟨*code*⟩   (no default)
   Prepends the given ⟨*code*⟩ to `/tcb/after lower`[→P. 34] *after* the content of the lower part.

11.3 Embedding into the Surroundings

The following option keys extend the options given in Subsection 3.10 from page 50.

> The 'hookable' versions are usable inside the document. In the preamble, they can only be used
> after explicit setting of `/tcb/before`[→P. 50] and `/tcb/after`[→P. 50] or by e. g. `/tcb/parskip`[→P. 50].

/tcb/before app=⟨*code*⟩   (no default)
   Appends the given ⟨*code*⟩ to `/tcb/before`[→P. 50] before the colored box.

/tcb/before pre=⟨*code*⟩   (no default)
   Prepends the given ⟨*code*⟩ to `/tcb/before`[→P. 50] before the colored box.

/tcb/after app=⟨*code*⟩   (no default)
   Appends the given ⟨*code*⟩ to `/tcb/after`[→P. 50] after the colored box.

/tcb/after pre=⟨*code*⟩   (no default)
   Prepends the given ⟨*code*⟩ to `/tcb/after`[→P. 50] after the colored box.

```
\tcbset{colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[title=My title,before app={The box follows:\\[4pt]},
  after app={This is the end.}]
This is a \textbf{tcolorbox}.
\end{tcolorbox}
```

The box follows:

> **My title**
>
> This is a **tcolorbox**.

This is the end.

11.4 Overlays

The following option keys extend the options given in Subsection 3.7 from page 41.

/tcb/overlay app=⟨*graphical code*⟩   (no default)
    Appends the given ⟨*graphical code*⟩ to .

```
% \usetikzlibrary{patterns} % preamble
% \tcbuselibrary{skins} % preamble
\tcbset{frogbox/.style={enhanced,colback=green!10,colframe=green!65!black,
  enlarge top by=5.5mm,
  overlay={\foreach \x in {2cm,3.5cm} {
    \begin{scope}[shift={([xshift=\x]frame.north west)}]
      \path[draw=green!65!black,fill=green!10,line width=1mm] (0,0) arc (0:180:5mm);
      \path[fill=black] (-0.2,0) arc (0:180:1mm);
    \end{scope}}}]}}
\tcbset{ribbon/.style={overlay app={%
  \path[fill=blue!75!white,draw=blue,double=white!85!blue,
    preaction={opacity=0.6,fill=blue!75!white},
    line width=0.1mm,double distance=0.2mm,
    pattern=fivepointed stars,pattern color=white!75!blue]
    ([xshift=-0.2mm,yshift=-1.02cm]frame.north east)
    -- ++(-1,1) -- ++(-0.5,0) -- ++(1.5,-1.5) -- cycle;}}}

\begin{tcolorbox}[frogbox,title=My title]
This is a \textbf{tcolorbox}.
\end{tcolorbox}

\begin{tcolorbox}[frogbox,ribbon,title=My title]
This is a \textbf{tcolorbox}.\par
Here, we apply a second overlay.
\end{tcolorbox}
```

My title

This is a **tcolorbox**.

My title

This is a **tcolorbox**.
Here, we apply a second overlay.

/tcb/overlay pre=⟨*graphical code*⟩   (no default)
    Prepends the given ⟨*graphical code*⟩ to .

/tcb/overlay unbroken app=⟨*graphical code*⟩   (no default)
    Appends the given ⟨*graphical code*⟩ to .

/tcb/overlay unbroken pre=⟨*graphical code*⟩   (no default)
    Prepends the given ⟨*graphical code*⟩ to .

`/tcb/overlay middle app`=⟨*graphical code*⟩ (no default)

Appends the given ⟨*graphical code*⟩ to `/tcb/overlay middle`→ P. 42.

`/tcb/overlay middle pre`=⟨*graphical code*⟩ (no default)

Prepends the given ⟨*graphical code*⟩ to `/tcb/overlay middle`→ P. 42.

`/tcb/overlay last app`=⟨*graphical code*⟩ (no default)

Appends the given ⟨*graphical code*⟩ to `/tcb/overlay last`→ P. 42.

`/tcb/overlay last pre`=⟨*graphical code*⟩ (no default)

Prepends the given ⟨*graphical code*⟩ to `/tcb/overlay last`→ P. 42.

`/tcb/overlay broken app`=⟨*graphical code*⟩ (no default)

Appends the given ⟨*graphical code*⟩ to `/tcb/overlay broken`→ P. 42.

`/tcb/overlay broken pre`=⟨*graphical code*⟩ (no default)

Prepends the given ⟨*graphical code*⟩ to `/tcb/overlay broken`→ P. 42.

`/tcb/overlay unbroken and first app`=⟨*graphical code*⟩ (no default)

Appends the given ⟨*graphical code*⟩ to `/tcb/overlay unbroken and first`→ P. 42.

`/tcb/overlay unbroken and first pre`=⟨*graphical code*⟩ (no default)

Prepends the given ⟨*graphical code*⟩ to `/tcb/overlay unbroken and first`→ P. 42.

`/tcb/overlay middle and last app`=⟨*graphical code*⟩ (no default)

Appends the given ⟨*graphical code*⟩ to `/tcb/overlay middle and last`→ P. 42.

`/tcb/overlay middle and last pre`=⟨*graphical code*⟩ (no default)

Prepends the given ⟨*graphical code*⟩ to `/tcb/overlay middle and last`→ P. 42.

11.5   Watermarks

The following option keys extend the options given in Subsection 6.4 from page 80.

> Watermarks are special overlays. The [LIB hooks] library allows the combination of several watermarks and overlays.

/tcb/watermark text app=⟨*text*⟩   (no default)

Appends a **/tcb/watermark text**→P. 80 to the colored box.

```
\tcbset{colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries}

\begin{tcolorbox}[enhanced,title=My title,watermark graphics=Basilica_5.png,
  watermark opacity=0.25,
  watermark text app=Basilica,watermark color=Navy
  ]
\lipsum[1-2]
\tcblower
This example uses a public domain picture from\\
\url{http://commons.wikimedia.org/wiki/File:Basilica_5.png}
\end{tcolorbox}
```

My title

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

- -

This example uses a public domain picture from
http://commons.wikimedia.org/wiki/File:Basilica_5.png

/tcb/watermark text pre=⟨*text*⟩   (no default)

Prepends a **/tcb/watermark text**→P. 80 to the colored box.

/tcb/watermark text app on=⟨*part*⟩ is ⟨*text*⟩   (no default)

Appends a **/tcb/watermark text on**→P. 80 the named ⟨*part*⟩ of a break sequence.

/tcb/watermark text pre on=⟨*part*⟩ is ⟨*text*⟩   (no default)

Prepends a **/tcb/watermark text on**→P. 80 the named ⟨*part*⟩ of a break sequence.

/tcb/watermark graphics app=⟨*file name*⟩ (no default)
Appends a [/tcb/watermark graphics](#)^{→P.81} referenced by ⟨*file name*⟩ to the colored box.

/tcb/watermark graphics pre=⟨*file name*⟩ (no default)
Prepends a [/tcb/watermark graphics](#)^{→P.81} referenced by ⟨*file name*⟩ to the colored box.

/tcb/watermark graphics app on=⟨*part*⟩ is ⟨*file name*⟩ (no default)
Appends a [/tcb/watermark graphics on](#)^{→P.81} the named ⟨*part*⟩ of a break sequence. The picture is referenced by ⟨*file name*⟩.

/tcb/watermark graphics pre on=⟨*part*⟩ is ⟨*file name*⟩ (no default)
Prepends a [/tcb/watermark graphics on](#)^{→P.81} the named ⟨*part*⟩ of a break sequence. The picture is referenced by ⟨*file name*⟩.

/tcb/watermark tikz app=⟨*graphical code*⟩ (no default)
Appends a [/tcb/watermark tikz](#)^{→P.82} with the given tikz ⟨*graphical code*⟩ to the colored box.

/tcb/watermark tikz pre=⟨*graphical code*⟩ (no default)
Prepends a [/tcb/watermark tikz](#)^{→P.82} with the given tikz ⟨*graphical code*⟩ to the colored box.

```
% \usepackage{tikz}
\tcbset{colback=red!5!white,colframe=red!75!black,fonttitle=\bfseries,
  watermark color=Navy,watermark opacity=0.25,
  smiley/.style={watermark tikz pre={%
    \path[fill=yellow,draw=yellow!75!red] (0,0) circle (1cm);
    \fill[red] (45:5mm) circle (1mm);
    \fill[red] (135:5mm) circle (1mm);
    \draw[line width=1mm,red] (215:5mm) arc (215:325:5mm);}}}

\begin{tcolorbox}[enhanced,title=My title, watermark text=Watermark,
  smiley]
\lipsum[1-2]
\end{tcolorbox}
```

My title

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

/tcb/watermark tikz app on=⟨*part*⟩ is ⟨*graphical code*⟩ (no default)
Appends a [/tcb/watermark tikz on](#)^{→P.82} the named ⟨*part*⟩ of a break sequence.

/tcb/watermark tikz pre on=⟨*part*⟩ is ⟨*graphical code*⟩ (no default)
Prepends a [/tcb/watermark tikz on](#)^{→P.82} the named ⟨*part*⟩ of a break sequence.

11.6 Skin Code

The following option keys extend the options given in Subsection 6.3 from page 77.

> The following code options are applicable only for skins which use engines of type `freelance`. If you intend to add only a few things to an existing skin, you should consider using overlays instead.

/tcb/frame code app=⟨*graphical code*⟩   (no default)
     Appends the given ⟨*graphical code*⟩ to /tcb/frame code^{→P.77}.

/tcb/frame code pre=⟨*graphical code*⟩   (no default)
     Prepends the given ⟨*graphical code*⟩ to /tcb/frame code^{→P.77}.

/tcb/interior titled code app=⟨*graphical code*⟩   (no default)
     Appends the given ⟨*graphical code*⟩ to /tcb/interior titled code^{→P.77}.

```
\begin{tcolorbox}[title=My title,freelance,colframe=Navy,
  frame code app={\draw[yellow,line width=1cm] (
    frame.south west)--(frame.north east);},
  interior titled code app={\draw[red,line width=1cm]
    (frame.north west)--(frame.south east);},
  ]
\lipsum[1]
\end{tcolorbox}
```

My title

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer 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.

/tcb/interior titled code pre=⟨*graphical code*⟩   (no default)
     Prepends the given ⟨*graphical code*⟩ to /tcb/interior titled code^{→P.77}.

/tcb/interior code app=⟨*graphical code*⟩   (no default)
     Appends the given ⟨*graphical code*⟩ to /tcb/interior code^{→P.78}.

/tcb/interior code pre=⟨*graphical code*⟩   (no default)
     Prepends the given ⟨*graphical code*⟩ to /tcb/interior code^{→P.78}.

/tcb/segmentation code app=⟨*graphical code*⟩   (no default)
     Appends the given ⟨*graphical code*⟩ to /tcb/segmentation code^{→P.78}.

/tcb/segmentation code pre=⟨*graphical code*⟩   (no default)
     Prepends the given ⟨*graphical code*⟩ to /tcb/segmentation code^{→P.78}.

/tcb/title code app=⟨*graphical code*⟩   (no default)
     Appends the given ⟨*graphical code*⟩ to /tcb/title code^{→P.79}.

/tcb/title code pre=⟨*graphical code*⟩   (no default)
     Prepends the given ⟨*graphical code*⟩ to /tcb/title code^{→P.79}.

12 Library LIB documentation

This library has the single purpose to support LaTeX package documentations like this one. Actually, the visual nature follows the approach from Till Tantau's `pgf` [17] documentation. Typically, this library is assumed to be used in conjunction with the class `ltxdoc` or alike.

The library is loaded by a package option or inside the preamble by:

```
\tcbuselibrary{documentation}
```

This also loads the library LIB listings, see section 7, and a bunch of packages, namely `doc`, `pifont`, `marvosym`, `hyperref`, `makeidx`, and `refcount`.

For UTF-8 support, load:

```
\tcbuselibrary{listingsutf8,documentation}
```

For `minted` [11] support, load:

```
\tcbuselibrary{documentation,minted}
\tcbset{listing engine=minted}
```

12.1 Macros of the Library

\begin{docCommand}[⟨*options*⟩]{⟨*name*⟩}{⟨*parameters*⟩}
  ⟨*command description*⟩
\end{docCommand}

> Documents a LaTeX macro with given ⟨*name*⟩ where ⟨*name*⟩ is written without backslash. The given ⟨*options*⟩ are set with `\tcbset`→P.8. This macro takes mandatory or optional ⟨*parameters*⟩. It is automatically indexed and can be referenced with `\refCom`→P.203{⟨*name*⟩}.

```
\begin{docCommand}{foomakedocSubKey}{\marg{name}\marg{key path}}
  Creates a new environment \meta{name} based on \refEnv{docKey} for the
  documentation of keys with the given \meta{key} path.
\end{docCommand}
```

> **\foomakedocSubKey**{⟨*name*⟩}{⟨*key path*⟩}
>
> > Creates a new environment ⟨*name*⟩ based on docKey→P.199 for the documentation of keys with the given ⟨*key path*⟩.

```
\begin{docCommand}[color definition=blue]{foomakedocSubKey*}%
    {\marg{name}\marg{key path}}
  Creates a new environment \meta{name} based on \refEnv{docKey} for the
  documentation of keys with the given \meta{key} path.
\end{docCommand}
```

> **\foomakedocSubKey***{⟨*name*⟩}{⟨*key path*⟩}
>
> > Creates a new environment ⟨*name*⟩ based on docKey→P.199 for the documentation of keys with the given ⟨*key path*⟩.

\begin{docEnvironment}[⟨*options*⟩]{⟨*name*⟩}{⟨*parameters*⟩}
 ⟨*environment description*⟩
\end{docEnvironment}

Documents a LaTeX environment with given ⟨*name*⟩. The given ⟨*options*⟩ are set with
\tcbset[→ P. 8]. This environment takes mandatory or optional ⟨*parameters*⟩. It is automatically indexed and can be referenced with \refEnv[→ P. 203]{⟨*name*⟩}.

```
\begin{docEnvironment}{foocolorbox}{\oarg{options}}
  This is the main environment to create an accentuated colored text box with
  rounded corners and, optionally, two parts.
\end{docEnvironment}
```

\begin{foocolorbox}[⟨*options*⟩]
 ⟨*environment content*⟩
\end{foocolorbox}
   This is the main environment to create an accentuated colored text box with rounded corners and, optionally, two parts.

```
\begin{docEnvironment}%
    [doclang/environment content=My content text]%
    {foocolorbox*}{\oarg{options}}
  This is the main environment to create an accentuated colored text box with
  rounded corners and, optionally, two parts.
\end{docEnvironment}
```

\begin{foocolorbox*}[⟨*options*⟩]
 ⟨*My content text*⟩
\end{foocolorbox*}
   This is the main environment to create an accentuated colored text box with rounded corners and, optionally, two parts.

\begin{docKey}[⟨*key path*⟩]{⟨*name*⟩}{⟨*parameters*⟩}{⟨*description*⟩}
 ⟨*key description*⟩
\end{docKey}

Documents a key with given ⟨*name*⟩ and an optional ⟨*key path*⟩. This key takes mandatory or optional ⟨*parameters*⟩ as value with a short ⟨*description*⟩. It is automatically indexed and can be referenced with \refKey[→ P. 204]{⟨*name*⟩}.

```
\begin{docKey}[foo]{footitle}{=\meta{text}}{no default, initially empty}
  Creates a heading line with \meta{text} as content.
\end{docKey}
```

/foo/footitle=⟨*text*⟩   (no default, initially empty)
   Creates a heading line with ⟨*text*⟩ as content.

\docAuxCommand{⟨*name*⟩}

Documents an auxiliary or minor LaTeX macro with given ⟨*name*⟩ where ⟨*name*⟩ is written without backslash. This macro is automatically indexed.

```
The macro \docAuxCommand{fooaux} holds some interesting data.
```

The macro **\fooaux** holds some interesting data.

\docColor{⟨*name*⟩}
   Documents a color with given ⟨*name*⟩. The color is automatically indexed.

   > The **color** \docColor{foocolor} is available.
   >
   > The color **foocolor** is available.

\cs{⟨*name*⟩}
   Macro from `ltxdoc` [3] to typeset a command word ⟨*name*⟩ where the backslash is prefixed. The library overwrites the original macro.

   > This is a \cs{foocommand}.
   >
   > This is a \foocommand.

\meta{⟨*text*⟩}
   Macro from `doc` [7] to typeset a meta ⟨*text*⟩.

   > This is a \meta{text}.
   >
   > This is a ⟨*text*⟩.

\marg{⟨*text*⟩}
   Macro from `ltxdoc` [3] to typeset a ⟨*text*⟩ with curly brackets as a mandatory argument. The library overwrites the original macro.

   > This is a mandatory \marg{argument}.
   >
   > This is a mandatory {⟨*argument*⟩}.

\oarg{⟨*text*⟩}
   Macro from `ltxdoc` [3] to typeset a ⟨*text*⟩ with square brackets as an optional argument. The library overwrites the original macro.

   > This is an optional \oarg{argument}.
   >
   > This is an optional [⟨*argument*⟩].

\brackets{⟨*text*⟩}
   Sets the given ⟨*text*⟩ with curly brackets.

   > Here we use \brackets{some **text**}.
   >
   > Here we use {some text}.

```
\begin{dispExample}
 ⟨environment content⟩
\end{dispExample}
```

Creates a colored box based on a `tcolorbox`[→P.7]. It displays the environment content as source code in the upper part and as compiled text in the lower part of the box. The appearance is controlled by `/tcb/documentation listing style`[→P.205] and the style `/tcb/docexample`[→P.205]. It may be changed by redefining this style.

```
\begin{dispExample}
This is a \LaTeX\ example.
\end{dispExample}
```

```
This is a \LaTeX\ example.
```

This is a LaTeX example.

```
\begin{dispExample*}{⟨options⟩}
 ⟨environment content⟩
\end{dispExample*}
```

The starred version of `dispExample` takes `tcolorbox`[→P.7] ⟨options⟩ as parameter. These ⟨options⟩ are executed after `/tcb/docexample`[→P.205].

```
\begin{dispExample*}{sidebyside}
This is a \LaTeX\ example.
\end{dispExample*}
```

```
This is a \LaTeX\ example.
```                          This is a LaTeX example.

\begin{dispListing}
 ⟨*environment content*⟩
\end{dispListing}
> Creates a colored box based on a `tcolorbox`[→P.7]. It displays the environment content as source code. The appearance is controlled by `/tcb/documentation listing style`[→P.205] and the style `/tcb/docexample`[→P.205]. It may be changed by redefining this style.

```
\begin{dispListing}
This is a \LaTeX\ example.
\end{dispListing}
```

```
    This is a \LaTeX\ example.
```

\begin{dispListing*}{⟨*options*⟩}
 ⟨*environment content*⟩
\end{dispListing*}
> The starred version of `dispListing` takes `tcolorbox`[→P.7] ⟨*options*⟩ as parameter. These ⟨*options*⟩ are executed after `/tcb/docexample`[→P.205].

```
\begin{dispListing*}{title=My listing}
This is a \LaTeX\ example.
\end{dispListing*}
```

```
    My listing

    This is a \LaTeX\ example.
```

\begin{absquote}
 ⟨*environment content*⟩
\end{absquote}
> Used to typeset an abstract as quoted and small text.

```
\begin{absquote}
|tcolorbox| provides an environment for colored and framed text boxes with a
heading line. Optionally, such a box can be split in an upper and a lower part.
\end{absquote}
```

> tcolorbox provides an environment for colored and framed text boxes with a heading line. Optionally, such a box can be split in an upper and a lower part.

\tcbmakedocSubKey{⟨*name*⟩}{⟨*key path*⟩}

Creates a new environment ⟨*name*⟩ based on docKey^{→P. 199} for the documentation of keys with the given ⟨*key path*⟩ as default. The new environment ⟨*name*⟩ takes the same parameters as docKey^{→P. 199} itself.

```
\tcbmakedocSubKey{docFooKey}{foo}

\begin{docFooKey}{foodummy}{=\meta{nothing}}{no default, initially empty}
Some key.
\end{docFooKey}
```

/foo/foodummy=⟨*nothing*⟩   (no default, initially empty)
    Some key.

\refCom{⟨*name*⟩}

References a documented LATEX macro with given ⟨*name*⟩ where ⟨*name*⟩ is written without backslash. The page reference is suppressed if it links to the same page.

```
We have created \refCom{foomakedocSubKey} as an example.
```

We have created \foomakedocSubKey^{→P. 198} as an example.

\refCom*{⟨*name*⟩}

References a documented LATEX macro with given ⟨*name*⟩ where ⟨*name*⟩ is written without backslash. There is no page reference.

```
We have created \refCom*{foomakedocSubKey} as an example.
```

We have created \foomakedocSubKey as an example.

\refEnv{⟨*name*⟩}

References a documented LATEX environment with given ⟨*name*⟩. The page reference is suppressed if it links to the same page.

```
We have created \refEnv{foocolorbox} as an example.
```

We have created foocolorbox^{→P. 199} as an example.

\refEnv*{⟨*name*⟩}

References a documented LATEX environment with given ⟨*name*⟩. There is no page reference.

```
We have created \refEnv*{foocolorbox} as an example.
```

We have created foocolorbox as an example.

\refKey{⟨*name*⟩}

References a documented key with given ⟨*name*⟩ where ⟨*name*⟩ is the full path name of the key. The page reference is suppressed if it links to the same page.

```
We have created \refKey{/foo/footitle} as an example.
```

We have created /foo/footitle^{→P. 199} as an example.

\refKey*{⟨*name*⟩}

References a documented key with given ⟨*name*⟩ where ⟨*name*⟩ is the full path name of the key. There is no page reference.

```
We have created \refKey*{/foo/footitle} as an example.
```

We have created /foo/footitle as an example.

\colDef{⟨*text*⟩}

Sets ⟨*text*⟩ with the definition color, see /tcb/color definition^{→P. 205}.

```
This is my \colDef{text}.
```

This is my text.

\colOpt{⟨*text*⟩}

Sets ⟨*text*⟩ with the option color, see /tcb/color option^{→P. 205}.

```
This is my \colOpt{text}.
```

This is my text.

12.2 Option Keys of the Library

/tcb/docexample (style, no value)

Sets the style for dispExample[→P.201] and dispListing[→P.202] with the colors `ExampleBack` and `ExampleFrame`. To change the appearance of the examples, this style can be redefined.

/tcb/documentation listing options=⟨*key list*⟩ (no default, initially `style=tcbdocumentation`)

Sets the options from the package `listings` [5]. They are used inside dispExample[→P.201] and dispListing[→P.202] to typeset the listings. Note that this is not identical to the key /tcb/listing options[→P.136] which is used for 'normal' listings.
Used for /tcb/listing engine[→P.139]=`listings` only.

/tcb/documentation listing style=⟨*listing style*⟩ (no default, initially `tcbdocumentation`)

Abbreviation for `documentation listing options={style=...}`. This key sets a ⟨*style*⟩ for the `listings` package, see [5]. Note that this is not identical to the key /tcb/listing style[→P.136] which is used for 'normal' listings.
Used for /tcb/listing engine[→P.139]=`listings` only.

/tcb/documentation minted style=⟨*key list*⟩ (no default, initially unset)

Sets a ⟨*style*⟩ known to `Pygments` [9] for the package `minted` [11], if used. Note that this is not identical to the key /tcb/minted style[→P.138] which is used for 'normal' listings.
Used for /tcb/listing engine[→P.139]=`minted` only.

/tcb/documentation minted options=⟨*minted style*⟩ (no default, initially `tabsize=2,fontsize=\small`)

Sets the options from the package `minted` [11] which are used during typesetting of the listing, if used. Note that this is not identical to the key /tcb/minted options[→P.138] which is used for 'normal' listings.
Used for /tcb/listing engine[→P.139]=`minted` only.

/tcb/color definition=⟨*color*⟩ (no default, initially `Definition`)

Sets the highlight color used by macro and key definitions.

/tcb/color option=⟨*color*⟩ (no default, initially `Option`)

Sets the color used for optional arguments.

/tcb/color hyperlink=⟨*color*⟩ (no default, initially `Hyperlink`)

Sets the color for all hyper-links, i. e. all internal and external links.

/tcb/before example=⟨*macros*⟩ (no default, initially `\par\smallskip`)

Sets the ⟨*macros*⟩ which are executed before dispExample[→P.201] and dispListing[→P.202] additional to /tcb/before[→P.50].

/tcb/after example=⟨*macros*⟩ (no default, initially empty)

Sets the ⟨*macros*⟩ which are executed after dispExample[→P.201] and dispListing[→P.202] additional to /tcb/after[→P.50].

/tcb/index format=⟨*format*⟩ (no default, initially `pgf`)

Determines the basic ⟨*format*⟩ of the generated index. Feasible values are:
- `pgf`: The index is formatted like in the `pgf` documentation.
- `doc`: The index assumed to be formatted by `doc`/`ltxdoc`. A usage of `makeindex` with `-s gind.ist` is assumed. The package `hypdoc` has to be loaded *before* `tcolorbox`.

/tcb/index actual=⟨*character*⟩ (no default, initially `@`)

Sets the character for 'actual' in automatic indexing.

/tcb/index quote=⟨*character*⟩ (no default, initially `"`)

Sets the character for 'quote' in automatic indexing.

/tcb/index level=⟨*character*⟩ (no default, initially !)
     Sets the character for 'level' in automatic indexing.

/tcb/index default settings (style, no value)
     Sets the makeindex default values for /tcb/index actual→P. 205, /tcb/index quote→P. 205, and /tcb/index level.

/tcb/index german settings (style, no value)
     Sets the makeindex values recommended for German language texts. This is identical to setting the following:

> `\tcbset{index actual={=},index quote={!},index level={>}}`

The following keys are provided for language specific settings. The English language is predefined.

/tcb/english language (style, no value)
     Sets all language specific settings to English.

/tcb/doclang/color=⟨*text*⟩ (no default, initially color)
     Text used in the index for colors.

/tcb/doclang/colors=⟨*text*⟩ (no default, initially Colors)
     Heading text in the index for colors.

/tcb/doclang/environment content=⟨*text*⟩ (no default, initially environment content)
     Text used in docEnvironment→P. 199.

/tcb/doclang/environment=⟨*text*⟩ (no default, initially environment)
     Text used in the index for environments.

/tcb/doclang/environments=⟨*text*⟩ (no default, initially Environments)
     Heading text in the index for environments.

/tcb/doclang/key=⟨*text*⟩ (no default, initially key)
     Text used in the index for keys.

/tcb/doclang/index=⟨*text*⟩ (no default, initially Index)
     Heading text for the index.

/tcb/doclang/pageshort=⟨*text*⟩ (no default, initially P.)
     Short text for page references.

12.3   Predefined Colors of the Library

The following colors are predefined. They are used as default colors in some library commands.

Option ■ , **Definition** ■ , **ExampleFrame** ■ , **ExampleBack** □ , **Hyperlink** ■ .

References

[1] Peter Breitenlohner. *The ε-TEX manual*. Version 2. Feb. 1998.
http://ftp.gwdg.de/pub/ctan/systems/e-tex/v2/doc/etex_man.pdf.

[2] David Carlisle. *The tabularx package*. Jan. 7, 1999.
hhttp://mirror.ctan.org/macros/latex/required/tools/tabularx.pdf.

[3] David Carlisle. *The file ltxdoc.dtx for use with LATEX 2ε*. Nov. 11, 2007.
http://mirror.ctan.org/macros/latex/base/ltxdoc.dtx.

[4] Toby Cubitt. *The cleveref Package*. Mar. 22, 2013.
http://mirror.ctan.org/macros/latex/contrib/cleveref/cleveref.pdf.

[5] Carsten Heinz and Brooks Moses. *The Listings Package*. Version 1.4. Feb. 2007.
http://mirror.ctan.org/macros/latex/contrib/listings/listings.pdf.

[6] Philipp Lehman. *The etoolbox Package*. Jan. 3, 2011.
http://mirror.ctan.org/macros/latex/contrib/etoolbox/etoolbox.pdf.

[7] Frank Mittelbach. *The doc and shortvrb Packages*. June 30, 2011.
http://mirror.ctan.org/macros/latex/base/doc.dtx.

[8] Heiko Oberdiek. *The listingsutf8 package*. Nov. 10, 2011.
http://mirror.ctan.org/macros/latex/contrib/oberdiek/listingsutf8.pdf.

[9] *Pygments. Python syntax highlighter*. 2013.
http://pygments.org/.

[10] Will Robertson. *A couple of things involving environments*. Apr. 1, 2013.
http://mirror.ctan.org/macros/latex/contrib/environ/environ.pdf.

[11] Konrad Rudolph. *The minted package. Highlighted source code in LATEX*. Sept. 17, 2011.
http://mirror.ctan.org/macros/latex/contrib/minted/minted.pdf.

[12] Rainer Schöpf, Bernd Raichle, and Chris Rowley. *A New Implementation of LATEX's verbatim and verbatim* Environments*. Mar. 12, 2001.
http://mirror.ctan.org/macros/latex/required/tools/verbatim.pdf.

[13] Thomas F. Sturm. *Mathematik 1 (B.Eng.)* Vorlesungs-Skriptum. Neubiberg: Universität der Bundeswehr München, 2007.
http://www.unibw.de/bw/sturm.

[14] Thomas F. Sturm. *Mathematik 2 (B.Eng.)* Vorlesungs-Skriptum. Neubiberg: Universität der Bundeswehr München, 2008.
http://www.unibw.de/bw/sturm.

[15] Thomas F. Sturm. *Höhere Mathematik: Fortgeschrittene mathematische Methoden (M.Eng.)* Vorlesungs-Skriptum. Neubiberg: Universität der Bundeswehr München, 2010.
http://www.unibw.de/bw/sturm.

[16] Thomas F. Sturm. *LATEX – Einführung in das Textsatzsystem*. 9th edition. RRZN-Handbücher. Hannover: Regionales Rechenzentrum für Niedersachsen (RRZN), May 2012.
http://www.rrzn.uni-hannover.de/buch.html?&titel=latex.

[17] Till Tantau. *The TikZ and PGF Packages. Manual for version 2.10*. Oct. 25, 2010.
http://mirror.ctan.org/graphics/pgf/base/doc/generic/pgf/pgfmanual.pdf.

[18] Kresten Krab Thorub, Frank Jensen, and Chris Rowley. *The calc package – Infix notation arithmetic in LATEX*. Aug. 22, 2007.
http://mirror.ctan.org/macros/latex/required/tools/.

Index