

terminalcode Package —Official Documentation

LoveElysia1314

dr.zqr@outlook.com

November 2, 2025 —Version 0.9.0

Abstract

The **terminalcode** package provides authentic-looking terminal-style code boxes for LaTeX documents. It features UTF-8 box-drawing characters, comprehensive ANSI 16-color support with dynamic dark and light themes, and seamless integration with external tools for capturing real terminal sessions. It is designed for technical documentation, tutorials, and any document requiring high-fidelity display of command-line interactions.

Contents

1 Introduction

1.1 Key Features

- **Authentic Terminal Styling:** Renders code blocks with a genuine terminal appearance, including UTF-8 box-drawing frames and monospaced fonts.
- **Dynamic Themes:** Switch between **dark** (default) and **light** themes at any point in the document.
- **ANSI Color Support:** Full support for the 16 standard ANSI colors, allowing for faithful reproduction of terminal output.
- **File Inclusion:** Display code and logs directly from external files using the `\terminput` command.
- **Syntax Highlighting:** Leverages the `listings` package for syntax highlighting across dozens of programming languages.
- **Tool Integration:** Works seamlessly with tools like `cmdlog2tex` to embed real, captured terminal sessions.

1.2 Requirements

- **Compiler:** XeLaTeX or LuaLaTeX (pdfLaTeX is **not** supported due to its limitations with Unicode and modern fonts).
- **File Encoding:** Your `.tex` source file must be saved with UTF-8 encoding.

2 Quick Start

2.1 Minimal Working Example

To get started, save the following code as `minimal-example.tex` and compile it with XeLaTeX or LuaLaTeX.

```
\documentclass{article}
\usepackage{ctex}
\usepackage[theme=dark]{terminalcode}

\begin{document}

\begin{termcode}[bash]{My First Terminal}
$ echo "Hello, terminalcode!"
Hello, terminalcode!
```

\end{termcode}

\end{document}

2.2 Showcase: Dark & Light Themes

The package excels at displaying real terminal sessions. Here is the same colored output rendered in both themes to demonstrate the effect.

2.2.1 Dark Theme (Default)

Dark: Terminal Session with Colors

```
1 (base) max@qmobile:~/Downloads/colorful_demo$ tree
2 .
3 |— archive.zip
4 |— directory_folder
5 |— executable_file
6 |— image.jpg
7 |— normal_file.txt
8 |— symlink_file -> normal_file.txt
9
10 1 directory, 5 files
```

2.2.2 Light Theme

Light: Terminal Session with Colors

```
1 (base) max@qmobile:~/Downloads/colorful_demo$ tree
2 .
3 |— archive.zip
4 |— directory_folder
5 |— executable_file
6 |— image.jpg
7 |— normal_file.txt
8 |— symlink_file -> normal_file.txt
9
10 1 directory, 5 files
```

3 Installation

3.1 Local Installation (Recommended)

For a single project, simply place `terminalcode.sty` in the same directory as your main `.tex` file. LaTeX will find it automatically.

3.2 System-wide Installation

For a permanent installation, move `terminalcode.sty` to your TeX distribution's search path and refresh the file database.

3.2.1 TeX Live (All Platforms)

```
mkdir -p $(kpsewhich -var-value TEXMFHOME)/tex/latex/terminalcode
cp terminalcode.sty $(kpsewhich -var-value TEXMFHOME)/tex/latex/terminalcode/
texhash
```

3.2.2 MiKTeX (Windows)

```
copy terminalcode.sty "C:\Program Files\MiKTeX\tex\latex\terminalcode\"
initexmf --update-fndb
```

4 Core Usage

4.1 The `termcode` Environment

Use the `termcode` environment to display inline code snippets.

4.1.1 Syntax

```
\begin{termcode}[language]{title}
... your code or output here ...
\end{termcode}
```

- **language** (optional, default: `text`): The programming language for syntax highlighting.
- **title** (required): The title displayed in the terminal window's header.

4.2 `\terminput` Command

Use `\terminput` to include content from an external file. This is the best way to manage long scripts or terminal logs.

4.2.1 Syntax

```
\terminput[language]{title}{path/to/file}
```

4.3 Multi-Language Support

Syntax highlighting is supported for many languages.

Bash Script

```
1 #!/bin/bash
2 for file in *.log; do
3     echo "Processing $file"
4     grep -c "ERROR" "$file"
5 done
```

Python Script

```
1 def greet(name):
2     """A simple greeting function."""
3     print(f"Hello, {name}!")
4
5 greet("World")
```

C Program

```
1 #include <stdio.h>
2
3 int main() {
4     printf("Hello from C!\n");
5     return 0;
6 }
```

5 Styling and Customization

5.1 Switching Themes

You can set the theme globally when loading the package or switch it dynamically within the document.

5.1.1 At Package Load Time

```
% Use light theme throughout the document
\usepackage[theme=light]{terminalcode}
```

5.1.2 At Runtime

Use `\terminalcodetheme{<theme>}` or its alias `\tctheme{<theme>}`.

```
\terminalcodetheme{light} % Switch to light theme
\begin{termcode}[text]{Light Theme Example}
This box uses the light theme.
\end{termcode}
```

```
\tctheme{dark} % Switch back to dark theme
\begin{termcode}[text]{Dark Theme Example}
And this one is dark again.
\end{termcode}
```

5.2 Package Options

Option	Values	Default
theme	dark, light	dark
monofont	Font name	DejaVu Sans Mono

Example: `\usepackage[theme=light, monofont=Courier New]{terminalcode}`

6 Advanced Features

6.1 Inputting the Escape Characters « and »

The symbols « (U+00AB, left-pointing double angle quotation mark) and » (U+00BB, right-pointing double angle quotation mark) are used to delimit LaTeX code within a `termcode` block. If you find them difficult to type, here are platform-specific methods:

6.1.1 Windows

- **Alt Code (requires Num Lock):**
 - «: Hold **Alt**, type **0171**, release **Alt**
 - »: Hold **Alt**, type **0187**, release **Alt**
- **Character Map:** Press **Win+R**, type **charmap**, search for “U+00AB” or “guillemet”, copy and paste.
- **Microsoft Pinyin (部分版本):** Try typing “guillemet”, “«”, “»”, or “双尖括号”, or add as a custom phrase in settings.
- **Simplest Method:** Copy from this document or search “« »” online and paste.

6.1.2 macOS

- **Default Shortcut (US Keyboard):**
 - «: Option + \
 - »: Option + Shift + \
- **Note:** Other keyboard layouts may differ. Check “System Settings → Keyboard → Input Methods → Show Virtual Keyboard”.

6.2 Manual ANSI Color Application

Within a code block, you can use «...» to escape to LaTeX and apply colors manually with `\ansicolor` or its alias `\ac`.

```
\begin{termcode}[text]{Manual ANSI Colors}
Normal text «\ac{31}»This is red«\ansireset» and back to normal.
Bright colors: «\ac{92}»This is bright green«\ansireset».
\end{termcode}
```

The `\ansicolor` command accepts a single color code argument (e.g., 31 for red, 92 for bright green). `\ansireset` reverts the formatting to the theme’s default.

6.2.1 Linux (e.g., Ubuntu)

- **Compose Key Method (if enabled):**
 - «: Compose + < + <
 - »: Compose + > + >
- **Unicode Input (X11):** Hold Ctrl + Shift + U, then:
 - Type “00ab” → release → get «
 - Type “00bb” → release → get »

Recommended: The simplest method across all platforms is to copy these symbols from this document or search “« »” online.

7 Integration with `cmdlog2tex`

For truly authentic terminal logs, we recommend the companion tool `cmdlog2tex`, which converts captured terminal sessions into a `.tex` file compatible with this package.

7.1 Workflow

1. **Capture:** Record a terminal session using tools like `script` (Linux/macOS) or by saving commands to a file.
2. **Convert:** Use `cmdlog2tex` to process the log file, preserving all ANSI colors.
3. **Include:** Import the resulting file into your document with `\terminput`.

7.2 Example

1. On Linux, record a session

```
script my_session.log
$ ls --color=auto
$ python --version
$ exit
```

2. Convert the log to a TeX file

```
cmdlog2tex my_session.log > my_session.tex
```

3. Include it in your document

```
\terminput[text]{My Recorded Session}{my_session.tex}
```

For detailed instructions, visit: <https://github.com/LoveElysia1314/cmdlog2tex>

8 Frequently Asked Questions

Q: Why does compilation fail with an error about pdfLaTeX?

A: This package relies on features only available in modern TeX engines. You **must** compile your document with XeLaTeX or LuaLaTeX.

Q: Why are the box-drawing characters or colors not showing correctly?

A: This is typically caused by one of two issues:

- Your document is not being compiled with XeLaTeX/LuaLaTeX.
- Your `.tex` file is not saved with UTF-8 encoding.

Q: How do I use a custom font?

A: Use the `monofont` package option: `\usepackage[monofont=Inconsolata]{terminalcode}`. Ensure the font is installed on your system. If not found, the package will fall back to a default monospaced font.

9 License and Contributing

9.1 License

This package is released under the **MIT License**. See the **LICENSE** file for complete details.

9.2 Contributing

Contributions, bug reports, and feature requests are welcome! Please visit the GitHub repository to participate.

- **Issues:** <https://github.com/LoveElysia1314/terminalcode-sty/issues>
- **Repository:** <https://github.com/LoveElysia1314/terminalcode-sty>