

# The darkmode package\*

Ruben Deisenroth  
rdeisenroth@gmail.com

September 1, 2022

## Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Why . . . . .	1
<b>2</b>	<b>Usage</b>	<b>2</b>
2.1	Basic usage . . . . .	2
2.2	Adopting the Darkmode . . . . .	2
<b>3</b>	<b>Detailed documentation</b>	<b>2</b>
3.1	Package options . . . . .	2
3.2	User Commands . . . . .	3
3.3	Commands for developers . . . . .	3
<b>4</b>	<b>implementation</b>	<b>4</b>
	<b>Index</b>	<b>6</b>

## 1 Introduction

The `darkmode` Package provides an API for template and package developers to create dynamic color schemes for light- and darkmodes. For those unaware: We refer to *dark mode* when a document has a dark background with a light font and to light mode if it has a dark font with a light background.

### 1.1 Why

Now you might think: Why would anyone even need to use a dark mode for PDF files? While for the common task of printing a dark mode has a lot of disadvantages and would take a lot of ink, these days it becomes more and more common to view PDF files directly on a Screen. And Screens emit a lot of Blue light, which - especially in the evening - can be really annoying to stare at when working on a large text document. Also modern displays can get so bright that a stark black and white contrast is too strong for most lighting scenarios.

---

\*This document corresponds to `darkmode` v1.0.1, dated 2022-09-01.

## 2 Usage

### 2.1 Basic usage

The basic Idea behind this package is that the dark mode is enabled in the preamble once, either by passing the package option `enable` or via the command `\enabledarkmode`, and applies to the rest of the document.

### 2.2 Adopting the Darkmode

When defining custom colors and commands, we can check if the dark mode is active or not to dynamically adjust the Colors.

Example:

```
\DeclareDocumentCommand{\bluecode}{m}{%
  \texttt{\textcolor{\IfDarkModeTF{cyan}{blue}}{#1}}%
}
```

Note that in most scenarios it is possible to use the `\IfDarkModeTF` command inline without any sideeffects. If cases where this doesn't work, an alternative is to wrap the entire definition of a macro in the switch:

```
\IfDarkModeTF{%
  \definecolor{accentcolor}{RGB}{0, 157, 129}%
}{%
  \definecolor{accentcolor}{RGB}{0, 122, 93}%
}
```

Another usefull trick is the usage of the colors `\thepagecolor` from the `pagecolor`-Package and the color `fgcolor` which is defined to the default font color. You can mix any color with these two colors to easily derive fitting colors:

```
% Fancy Altering Table Row Colors
\rowcolors{2}{\thepagecolor}{fgcolor!10!\thepagecolor}
```

Note that `yellow!10!\thepagecolor` is not the same as `yellow!10` because the second one would just default to `yellow!10!white`.

## 3 Detailed documentation

### 3.1 Package options

The following options can be passed to the package:

- `enable` - boolean input, enables or disables the dark mode for the entire document
- `defaulthook` - boolean input, whether to enable or disable the default hook (`\@darkmode@on@enable` or `\darkmode@on@disable`)
- `nodefualthook` - meta key for disabeling the default hook

## 3.2 User Commands

---

<code>\enabledarkmode</code>	<code>\enabledarkmode</code> enables the dark Mode at the current position, while <code>\disabledarkmode</code>
<code>\disabledarkmode</code>	disables the dark mode at the current position.

---

**Note:** that this is only intended to be used in the preamble for an entire document mainly with a CI/CD. Trying to toggle the dark mode between different pages may lead to unexpected behavior and is not recommended. For regular use to pass the *enabled* option to the package.

---

<code>\IfDarkModeT</code> *	<code>\IfDarkModeT</code> $\{\langle true\ code\rangle\}$
<code>\IfDarkModeF</code> *	<code>\IfDarkModeF</code> $\{\langle false\ code\rangle\}$
<code>\IfDarkModeTF</code> *	<code>\IfDarkModeTF</code> $\{\langle true\ code\rangle\}$ $\{\langle false\ code\rangle\}$

---

Checks whether the dark mode is enabled or not. If so,  $\langle true\ code\rangle$  is executed, otherwise  $\langle false\ code\rangle$  is executed. These macros are fully expandable.

## 3.3 Commands for developers

This functionality is mostly aimed at package developers and template creators, for the end user it should not be necessary to manually fixup the Switching with compatible Packages and Classes.

---

<code>\@darkmode@set@enabled</code>	Functions for template developers that want to implement dark mode toggling with
<code>\@darkmode@set@disabled</code>	$\LaTeX 2_{\epsilon}$ .

---

New: 2022-08-30

---

## 4 implementation

`\g_@@_dark_mode_bool` First we define the package Options. The usage of `initial` ensures that the boolean `\g_@@_dark_mode_bool` always exists.

```
1 \RequirePackage{xcolor,pagecolor}
2 \keys_define:nn {darkmode} {
3   enable .bool_gset:N = \g_@@_dark_mode_bool,
4   enable .initial:n = false,
5   enable .default:n = true,
6   defaulthook .bool_gset:N = \g_@@_dark_mode_hook_bool,
7   defaulthook .initial:n = true,
8   defaulthook .default:n = true,
9   nodefaulthook .meta:n = {defaulthook=false},
10 }
```

Now we process the package options.

```
11 \ProcessKeysOptions{darkmode}
```

*(End definition for `\g_@@_dark_mode_bool`. This variable is documented on page ??.)*

`\_darkmode_if_dark_mode_p:` We need a global Conditional to decide whether the dark mode is currently active or not

```
\_darkmode_if_dark_mode:TF 12 \prg_new_conditional:Nnn \_darkmode_if_dark_mode: {T,F,TF} {
13   \bool_if:NTF \g_@@_dark_mode_bool
14   {\prg_return_true:}
15   {\prg_return_false:}
16 }
```

*(End definition for `\_darkmode_if_dark_mode:TF`.)*

```
\IfDarkModeT
\IfDarkModeF 17 \cs_set_eq:NN\IfDarkModeT \_darkmode_if_dark_mode:T
\IfDarkModeTF 18 \cs_set_eq:NN\IfDarkModeF \_darkmode_if_dark_mode:F
19 \cs_set_eq:NN\IfDarkModeTF \_darkmode_if_dark_mode:TF
```

*(End definition for `\IfDarkModeT`, `\IfDarkModeF`, and `\IfDarkModeTF`. These functions are documented on page 3.)*

```
\enabledarkmode
\@darkmode@set@enabled 20 \definecolor{darkmode@anthrazitgrau}{HTML}{293133}
\@darkmode@on@enable 21 \colorlet{fgcolor}{.}
22 \DeclareDocumentCommand{\@darkmode@set@enabled}{-}{
23   \bool_set_true:c {g_@@_dark_mode_bool}
24 }
25 \DeclareDocumentCommand{\@darkmode@on@enable}{-}{
26   \pagecolor{darkmode@anthrazitgrau}
27   \color{white}
28   \selectcolormodel{RGB}
29   \colorlet{fgcolor}{.}
30 }
31 \DeclareDocumentCommand{\enabledarkmode}{-}{
32   \@darkmode@set@enabled
33   \bool_if:cT {g_@@_dark_mode_hook_bool} {\@darkmode@on@enable}
34 }
```

*(End definition for `\enabledarkmode`, `\@darkmode@set@enabled`, and `\@darkmode@on@enable`. These functions are documented on page 3.)*

```

\disabledarkmode
\@darkmode@set@disabled 35 \DeclareDocumentCommand{\@darkmode@set@disabled}{}{
\@darkmode@on@disabled 36 \bool_set_false:c {g_@@_dark_mode_bool}
37 }
38 \DeclareDocumentCommand{\@darkmode@on@disable}{}{
39 \pagecolor{white}
40 \color{black}
41 \colorlet{fgcolor}{.}
42 }
43 \DeclareDocumentCommand{\disabledarkmode}{}{
44 \@darkmode@set@disabled
45 \bool_if:cT {g_@@_dark_mode_hook_bool} {\@darkmode@on@disable}
46 }

```

*(End definition for \disabledarkmode, \@darkmode@set@disabled, and \@darkmode@on@disabled. These functions are documented on page 3.)*

```

\__darkmode_update_from_bool
47 \DeclareDocumentCommand{\__darkmode_update_from_bool}{}{
48 \IfDarkModeTF{\enabledarkmode}{\disabledarkmode}
49 }
50 \__darkmode_update_from_bool

```

*(End definition for \\_\_darkmode\_update\_from\_bool.)*

# Index

The italic numbers denote the pages where the corresponding entry is described, numbers underlined point to the definition, all others indicate the places where it is used.

	<b>Symbols</b>	<code>\IfDarkModeTF</code> . . . . . 3, <u>17</u> , 48
<code>@@</code> commands:		<b>K</b>
<code>\g_@@_dark_mode_bool</code> . . . . . 4, <u>1</u> , 13		keys commands:
<code>\g_@@_dark_mode_hook_bool</code> . . . . . 6		<code>\keys_define:mn</code> . . . . . 2
	<b>B</b>	<b>P</b>
bool commands:		<code>\pagecolor</code> . . . . . 26, 39
<code>\bool_if:NTF</code> . . . . . 13, 33, 45		prg commands:
<code>\bool_set_false:N</code> . . . . . 36		<code>\prg_new_conditional:Nnn</code> . . . . . 12
<code>\bool_set_true:N</code> . . . . . 23		<code>\prg_return_false:</code> . . . . . 15
	<b>C</b>	<code>\prg_return_true:</code> . . . . . 14
<code>\color</code> . . . . . 27, 40		<code>\ProcessKeysOptions</code> . . . . . 11
<code>\colorlet</code> . . . . . 21, 29, 41		<b>R</b>
cs commands:		<code>\RequirePackage</code> . . . . . 1
<code>\cs_set_eq:NN</code> . . . . . 17, 18, 19		<b>S</b>
	<b>D</b>	<code>\selectcolormodel</code> . . . . . 28
darkmode internal commands:		<b>T</b>
<code>\__darkmode_if_dark_mode:TF</code> . . . . .		<b>T<sub>E</sub>X</b> and <b>L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub></b> commands:
. . . . . <u>12</u> , 17, 18, 19		<code>\@darkmode@on@disable</code> . . . . . 38, 45
<code>\__darkmode_if_dark_mode_p:</code> . . . . . <u>12</u>		<code>\@darkmode@on@disabled</code> . . . . . <u>35</u>
<code>\__darkmode_update_from_bool</code> . . . . . <u>47</u>		<code>\@darkmode@on@enable</code> . . . . . 2, <u>20</u>
<code>\DeclareDocumentCommand</code> . . . . .		<code>\@darkmode@set@disabled</code> . . . . . 3, <u>35</u>
. . . . . 22, 25, 31, 35, 38, 43, 47		<code>\@darkmode@set@enabled</code> . . . . . 3, <u>20</u>
<code>\definecolor</code> . . . . . 20		<code>\darkmode@on@disable</code> . . . . . 2
<code>\disableddarkmode</code> . . . . . 3, <u>35</u> , 48		<code>\disableddarkmode</code> . . . . . 3
	<b>E</b>	<code>\enableddarkmode</code> . . . . . 2, 3
<code>\enableddarkmode</code> . . . . . 3, <u>20</u> , 48		<code>\IfDarkModeF</code> . . . . . 3
	<b>I</b>	<code>\IfDarkModeT</code> . . . . . 3
<code>\IfDarkModeF</code> . . . . . 3, <u>17</u>		<code>\IfDarkModeTF</code> . . . . . 2, 3
<code>\IfDarkModeT</code> . . . . . 3, <u>17</u>		<code>\thepagecolor</code> . . . . . 2