

BESPOKE MULTIPURPOSE CLASS

BMC

v0.2.2*

Pretentiousness Given Form

tecosaur
of GitHub

13th December 2019

Preface

Like with most things I didn't start out with the intent to end up this way. Initially I had a slowly growing template that I used for most documents; every so often I'd discover a package that did something I liked, or a setting that I preferred to be non-default. *Every* time that happened I'd want to go through the current documents I was working on and apply the latest revelations. Then when revisiting old documents I'd want to get them 'up to scratch'. There would always be the odd document I forgot about, or line missed, and so I quickly became tired of this process.

After realising that if I made a class and shoved it in my `texmf` directory that I'd be able to as many improvements as I like and they'd all be applied when I recompiled, *as well* as make initial configuration greatly simplified – I couldn't see a reason not to do it.

This class is very much written with my personal taste, and specific use case in mind. While I try to keep things general, it is very much built around my particular perspective. As such it is reasonable to think that to the community as a whole the self-importance in the name is a tad exaggerated or undeserved. Considering that is also designed to not just convey information but also designed to visually impress, the tagline 'Pretentiousness given form' seems somewhat appropriate.

I'm pleased to say that I consider this a project a success (in those respects). As I have largely drawn upon snippets of LaTeX floating around online I though the least I could do is give others that same opportunity. As such here you have an overview of my personal class designed to work for all of the documents I produce. In other words a *bespoke, multipurpose class* – or BMC for short.

Enjoy!

tecosaur

Contents

4	Chapter 1 What This Does
15	Chapter 2 Boring Info
26	Chapter 3 Reasoning

1

What This Does



Summary

- 1.1 Typography, 5
- 1.2 Boxes, 7
- 1.3 Colour, 8
- 1.4 Mathematics, 9
- 1.5 Code, 13
- 1.6 Chemistry, 14
- 1.7 Links and Metadata, 14

1.1 Typography

1.1.1 Typefaces

This package loads three typefaces.

1. IBM Plex Serif
2. IBM Plex Sans
3. IBM Plex Mono

I wanted a selection where serif, sans, and mono all mix well. Ideally with a few weight variants. Additionally I wanted a typographic style that meshed well with the large class of documents I indented to use this for. IBM Plex seems like a good fit (For more info see [3.2 Typefaces](#)). For all three of these a linespread of 1.15 is used.

Typeface	Bold	Semibold	Medium	Text	Regular	Light	Extra L	Thin
<code>\selectfont</code>	b	sb	mb	tx	m	l	el	t
Plex Serif	Words	Words	Words	Words	Words	Words	Words	Words
<i>Plex Serif</i>	<i>Words</i>	<i>Words</i>	<i>Words</i>	<i>Words</i>	<i>Words</i>	<i>Words</i>	<i>Words</i>	<i>Words</i>
Plex Sans	Words	Words	Words	Words	Words	Words	Words	Words
<i>Plex Sans</i>	<i>Words</i>	<i>Words</i>	<i>Words</i>	<i>Words</i>	<i>Words</i>	<i>Words</i>	<i>Words</i>	<i>Words</i>
Condensed	Words	Words	Words	Words	Words	Words	Words	Words
<i>Condensed</i>	<i>Words</i>	<i>Words</i>	<i>Words</i>	<i>Words</i>	<i>Words</i>	<i>Words</i>	<i>Words</i>	<i>Words</i>
Plex Mono	Words	Words	Words	Words	Words	Words	Words	Words
<i>Plex Mono</i>	<i>Words</i>	<i>Words</i>	<i>Words</i>	<i>Words</i>	<i>Words</i>	<i>Words</i>	<i>Words</i>	<i>Words</i>

Table 1.1: IBM Plex; Font Styles and Weights

Some Typeface Considerations

1

While this is the default, you can still load another font as usual in the preamble, e.g. `\usepackage{lmodern}` to switch to Latin Modern. Bear in mind that varying font weights are using throughout this class, so a font without the `sb`, `tx`, etc. weights will report warnings along the lines of `Font shape T1/FONT_HERE/STYLE/n' undefined`.

1.1.2 Roman Numerals

While biblatex does provide handy roman numeral command, it's nice to have them available regardless. Hence this class provides them if they aren't already available. To get upper case roman numerals use `\RN{1978}` to produce MCMLXXVIII, and `\Rn{1978}` to produce mcmlxxviii.

```
304 \providecommand*\RN}[1]{\expandafter\@slowromancap\romannumeral #1@}
305 \providecommand*\Rn}[1]{\romannumeral#1\relax}
```

1.1.3 Faux Small Caps

Some fonts (such as IBM Plex) are not kind enough to provide small caps. Simply using downscaled capitals is a barbaric and decidedly inferior solution. So `\fauxsc{}{}` is defined which, while not as nice as *true* small caps, is a darn sight better than just reducing the font size. `\fauxsc{}{}` is *automatically* used when `\textsc{}{}` is called if the current font does not have small caps.

```
\textsc using \fauxsc:  SMALL CAPS
Barbaric Solution:      SMALL CAPS
```

Usage Warning: Small Caps

1

If using this in macros or the like, you may get errors such as “Improper alphabetic constant”, “Missing = inserted for \ifnum”, and “Missing number, treated as zero”.

Here you will likely need to use `\expandafter\textsc\expandafter` instead.

1.1.4 Penalties

The class sets new penalties.

```
209 \@beginparpenalty=10000 % don't like it when a paragraph title is on a
    ↳ different page to the start of the content
210 \hyphenpenalty=500 % not a huge fan of hyphens, but they are worthwhile
211 \lefthyphenmin=2
212 \righthyphenmin=3
```

1.1.5 Captions

Caption labels are made to be upright sans-serif in the ‘text’ style, while captions are italic in the style of the body. When captions flow beyond a single line, ragged right alignment is used.

```
464 \setkomafont{caption}{\itshape\color{text}}
465 \setkomafont{captionlabel}{\fontfamily{\headingsFont}\fontseries{tx}}
    ↳ \selectfont\upshape\color{text}}
466 \captionsetup{justification=raggedright,singlelinecheck=true}
```

1.1.6 Terms and Acronyms

With the `gloss` option, this class loads and configures the `glossaries` package. Three things are changed

1. A new command `\acr{TEXT}` is added. This
 - a) Selects the next higher font weight
 - b) Scales the text vertically by a factor of 0.84, and horizontally by 0.91
 - c) Increases `tracking` by 7M/100*
2. Acronyms are typeset in the style of `\acr`, e.g. **Bespoke Multipurpose Class (BMC)**
3. The glossary style is configured to a variant of `long3col`
4. A new command `\newdefinedacronym{label}{short}{long}{description}` is added

1.2 Boxes

A collection of boxed environments are defined using `tcolorbox`, there have been two so far: **Information 1** and **Warning 1**. You can use these environments like so:

```
1 \begin{example}
2   This is an example. The next one is \autoref{eg:box-demo}.
3 \end{example}
```

Example 1

This is an example. The next one is **Example 2**.

These environments also accept three optional arguments.

1. The subtitle. This appears after the bold title (**Example N**).
2. The title, if this is set the counter value is shown in the right corner, as seen in the example below.
3. The label suffix, defaulting to the counter value. The full label is `<prefix>:<suffix>`.

```
1 \begin{example}[sub-title][title][box-demo]
2   This is another example, following up from \autoref{eg:1}.
3 \end{example}
```

title: sub-title

2

This is another example, following up from **Example 1**.

*See ¶ 2.1.6 of Bringhurst, 2004

A decent variety of environments are already defined (see [Table 1.2](#) and [Table 1.3](#)).

















Icon	Environment Name	Title	Label Prefix
 	example	Example	eg:
 	critical	Critical	crit:
 	question	Question	qu:
 	info	Information	info:
 	check	Check	check:
 	warning	Warning	warn:
 	tip	Tip	tip:
 	(maths environments)		

Table 1.2: Boxed Environments

Maths Environment	Label Prefix
proof	prf:
theorem	thr:
lemma	lem:
corollary	cor:
definition	def:
axiom	axi:
proposition	prop:

Table 1.3: Boxed Mathematics Environments

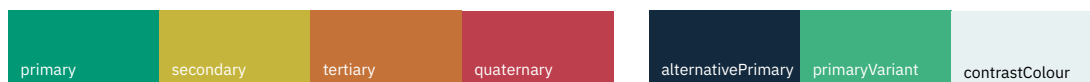
In addition to those provided, you can also define your own using the same helper function:

```
1 \newBmcBox{<env name>}{<colour>}{<title name>}[<reference prefix>][<icon>]
```

1.3 Colour

1.3.1 Theme Colours

This class makes use of the following defined colours.



Modifying these colours in the preamble affects the entire document.

1.3.2 Colour Palette

While `xcolor` does already come with some ‘nice’ shades, nice colour themes may be found at <https://flatuicolors.com>. These shades do not use the pretentious names listed, we just call them what they are (e.g. nephritis → green). Instead of overwriting the pre-existing colour, these colours have been differentiated by capitalisation, i.e. ‘Green’ instead of ‘green’. The default colour palette is shown below, however you can easily switch to another — see [2.4.2 Palettes](#) for more information.



It’s important to note that the palette only affects these named colours, and that the primary, secondary, etc. colours are not affected.

1.3.3 Functional Colours

This package has a few special colours that describe a particular aspect of a document, such as `href` and `inlinemath`. For more information see [2.4.1 Colours](#).

1.4 Mathematics

This class makes a few additions, and one or two modifications to Mathematics.

Demo

$$\frac{1}{2\pi i} \int_{\gamma} f(x^{\mathbf{N} \in \mathbb{C}^{N \times 10}}) = \sum_{k=1}^m n(\gamma; a_k) \text{Res}(f; a_k).$$

$$\iiint_Q f(w, x, y, z) dw dx dy dz \leq \oint_{\partial Q} f' \left(\max \left\{ \frac{\|w\|}{|w^2 + x^2|}, \frac{\|z\|}{|y^2 + z^2|}, \frac{\|w \oplus z\|}{|x \oplus y|} \right\} \right).$$

1.4.1 Modifications

Less/greater than or equal The less than or equal, and greater than or equal symbols are changed as such:

$$\begin{array}{ccc} \leq & \text{old} & \geq \\ \leqslant & \text{new} & \geqslant \end{array}$$

Inline math colour After interspersing maths and text a fair bit I've begun to think there's some merit to the Beamer 'make all maths a different colour' approach. So I've redefined the LaTeX inline maths command such that `\(a^x + bx + c\)` now becomes $ax^2 + bx + c$. Avoiding this is easy, just change the colour of `inlinemath` in the preamble like so `\colorlet{inlinemath}{text}` and you won't notice this exists. For once-offs I've defined a starred variant `\(* a^x + bx + c\)` which produces the normal $ax^2 + bx + c$.

```
296 \renewrobustcmd{\({}\@ifstar\@inlinemath\@inlinemath}
297 \DeclareRobustCommand{\@inlinemath}{\relax\ifmmode\@badmath\else$\fi}
298 \DeclareRobustCommand{\@inlinemath}{\relax\ifmmode\@badmath\else$\fi}
    \color{inlinemath}
```

Matrix environment The default for matrices (using `\begin{bmatrix}` or similar) is left aligned values, with no option to change this. This class adds an optional parameter to change the alignment, (`\begin{bmatrix}[r]`), and defaults to right aligned.

Old

$$\begin{bmatrix} 3 & -2 \\ -1 & 7 \end{bmatrix}$$

New

$$\begin{bmatrix} 3 & -2 \\ -1 & 7 \end{bmatrix}$$

1.4.2 Additions

Delimiters

<code>\abs{x}</code>	→	$ x $
<code>\norm{x}</code>	→	$\ x\ $
<code>\ceil{x}</code>	→	$\lceil x \rceil$
<code>\floor{x}</code>	→	$\lfloor x \rfloor$
<code>\round{x}</code>	→	$\lceil x \rceil$

[†]Can also be used outside of maths mode.

Sets [†]	<code>\RR</code>	→	\mathbb{R}	<code>\RR[n]</code>	→	\mathbb{R}^n
	<code>\NN</code>	→	\mathbb{N}	<code>\NN[n]</code>	→	\mathbb{N}^n
	<code>\ZZ</code>	→	\mathbb{Z}	<code>\ZZ[n]</code>	→	\mathbb{Z}^n
	<code>\QQ</code>	→	\mathbb{Q}	<code>\QQ[n]</code>	→	\mathbb{Q}^n
	<code>\CC</code>	→	\mathbb{C}	<code>\CC[n]</code>	→	\mathbb{C}^n
	<code>\PP</code>	→	\mathbb{P}	<code>\PP[n]</code>	→	\mathbb{P}^n
	<code>\HH</code>	→	\mathbb{H}	<code>\HH[n]</code>	→	\mathbb{H}^n
Differentials [†]	<code>\dd</code>	→	d			
	<code>\dv{t}{y}</code>	→	$\frac{d}{dt}$			
	<code>\dv{x}{y}</code>	→	$\frac{dx}{dy}$			
	<code>\pdv{f}{x}</code>	→	$\frac{\partial f}{\partial x}$			
Stats Operators	<code>\Var</code>	→	Var			
	<code>\Cov</code>	→	Cov			
	<code>\E</code>	→	E			
Others	<code>\qed</code> [†]	→	■			
	<code>\qedhere</code> [†]	→	□			
	<code>\Lap</code>	→	\mathcal{L}			

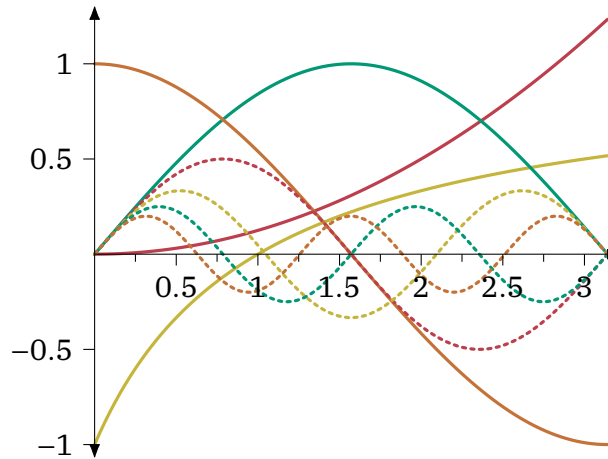
1.4.3 Plotting

Plot are nice, hence some effort has been put into making them look nice.

```

1 \begin{tikzpicture}
2   \begin{axis}[domain=0:pi]
3     \addplot+{x^2/8};
4     \addplot+{(x-1)/(x+1)};
5     \addplot+{sin(deg(x))};
6     \addplot+{cos(deg(x))};
7     \pgfplotsinvokeforeach{2,...,5}{
8       \addplot+{1/#1*sin(#1*deg(x))};}
9   \end{axis}
10 \end{tikzpicture}

```



Asymptopes A notable addition made is some nice asymptote-related pgf keys that have been added. There are three relevant keys

`v asym` locations(s) of vertical asymptotes

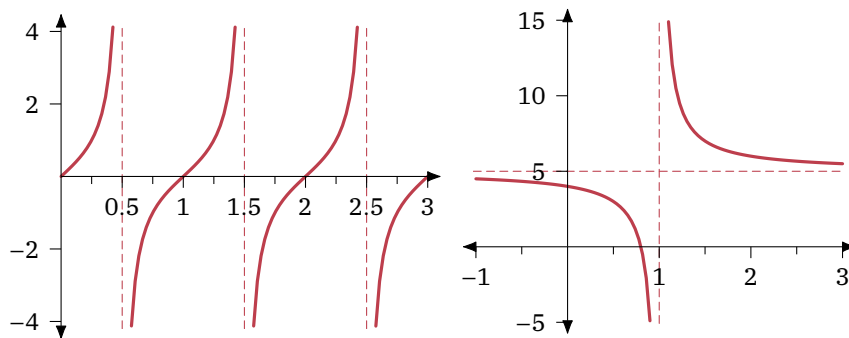
`h asym` locations(s) of horizontal asymptotes

`asym gap` distance around vertical asymptote which is cleared

```

1 \begin{tikzpicture}
2   \begin{axis}[domain=0:3,small]
3     \addplot+[asym gap=0.05,v asym={0.5,1.5,2.5}]{tan(deg(pi*x))};
4   \end{axis}
5 \end{tikzpicture}
6 \begin{tikzpicture}
7   \begin{axis}[domain=-1:3,small]
8     \addplot+[asym gap=0.1,v asym=1,h asym=5]{1/(x-1)+5};
9   \end{axis}
10 \end{tikzpicture}

```



1.5 Code

This package spends a few lines tweaking the minted and tcolorbox config to get code blocks to look rather nice.

For example:

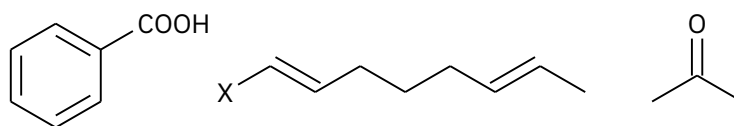
```

1 \section{Code}
2
3 This package spends a few lines tweaking
4 the minted and tcolorbox config to get code
5 blocks to look rather nice.
6
7 For example:
8 \begin{minted}[escapeinside=,highlightlines={8,17}]{tex}
9   \section{Code}
10
11   This package spends a few lines tweaking
12   the minted and tcolorbox config to get code
13   blocks to look rather nice.
14
15   For example:
16   ...
17 \end{minted}

```

1.6 Chemistry

When the `chem` option is used, `mhchem` is loaded with the configuration, however `chemfig` undergoes a few modifications to make the results look cleaner.



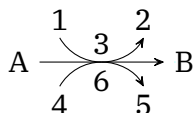
```

310 \setchemfig{
311   chemfig style={line width=0.06642 em}, % 'Line Width'
312   angle increment=30,
313   double bond sep=0.35700 em, % 'Bond Spacing'
314   atom sep=1.78500 em, % 'Fixed Length'
315   bond offset=0.18265 em % 'Margin Width'
316 }
317 \renewcommand*\printatom[1]{\small\ensuremath{\mathsf{#1}}}
```

We also grabbed some nice code for a custom arrow type `x` from a stackexchange answer so now we can do this:

```

1 \schemestart
2 A
3 \arrow{-X>[1][2][3][4][5][6]}
4 B
5 \schemestop
```



1.7 Links and Metadata

Both the `hyperref` and `hyperxmp` packages are used. The widely used `hyperref` package of course provides hyperlinks. This is ~~also~~ used to add a few extra links; specifically every page number is a link to the TOC, and the text of every header links to the relevant chapter page. This allows you to jump all over the document in just a few clicks.

The `hyperxmp` package is rather handy for setting a few fields of pdf metadata. Using the `\title`, `\author`, and `\subtitle` attributes it sets the relevant metadata fields.

2

Boring Info



Summary

- 2.1 Implementation, 16
- 2.2 Class Options, 16
- 2.3 Packages Used, 18
- 2.4 Configuration, 22

2.1 Implementation

While originally one giant class file, each major component has now been split off into a `.sty` file that can also operate as a standalone package. Currently, the following packages exist:

- `bmc-boxes`
- `bmc-color`
- `bmc-fonts`
- `bmc-maths`
- `bmx-ref`
- `bmc-sectioning`

2.2 Class Options

This class builds off `scrartcl`, any other options than those listed here will just be passed through.

2.2.1 Main Styling

`dark` Switches to a dark version of the style

`solid` Uses style with solid title page, and wide stripes on chapter pages, with solid colour bar at top of pages

`stripe` Uses plain background on title page, and thin stripes on chapter pages

`article` Use `scrartcl` class instead of `scrrept`

`notes` Move the margins to make room for notes

2.2.2 Fonts

Body Text Use `[body=<variant>]`, where variant is one of `serif`, `sans`, or `mono`. The default is `serif`.

Math Use `[math=<variant>]`, where variant is one of `serif`, `sans`, or `mono`. The default is `serif`.

2.2.3 Headings

These options set the style of the following components

- `\chapter` through to `\subparagraph`
- The page head, and page number
- Caption labels

Use `[headings=<variant>]`, where variant is one of `serif`, `sans`, or `mono`.
The default is `sans`.

2.2.4 Package Related

`chem` Load and configure `mhchem` and `chemfig` packages

`code` Load and configure minted package

`plot` Load and configure `pgfplots` package

`math` Load and configure some mathematical packages, and set font to match main text font (also see `math-serif` etc.)

2.3 Packages Used

2.3.1 Overview

Category	Packages				
General	<code>etoolbox</code>	<code>xpatch</code>	<code>Silence</code>	<code>ifdraft</code>	<code>geometry</code>
	<code>titlesec</code>	<code>titletoc</code>	<code>framed</code>	<code>textpos</code>	<code>calc</code>
	<code>xcolor</code>	<code>tikz</code>	<code>hyperref</code>	<code>hyperxmp</code>	<code>scrpage</code>
Text	<code>microtype</code>	<code>setspace</code>	<code>plex-serif</code>	<code>plex-sans</code>	<code>plex-mono</code>
	<code>multicol</code>				
Table	<code>booktabs</code>	<code>tabularx</code>	<code>longtable</code>		
Graphics	<code>graphicx</code>	<code>grffile</code>	<code>subcaption</code>	<code>caption</code>	
infoBulle	<code>infoBulle</code>	<code>marginInfoBulle</code>	<code>fontawesome5</code>		
Chemistry	<code>mhchem</code>	<code>chemfig</code>			
Code	<code>minted</code>	<code>tcolorbox</code>			
Maths	<code>amsmath</code>	<code>amssymb</code>	<code>mathdesign</code>	<code>xfrac</code>	<code>cancel</code>
	<code>mathtools</code>	<code>mathastext</code>	<code>pgfplots</code>		

Table 2.1: All Packages Used by This Class

2.3.2 General Packages

`etoolbox` Provides LaTeX frontends to some of the new primitives provided by e-TeX as well as some rather useful some generic tools – namely,

- Robust definitions
- Command Patching
- Command Protection
- Arithmetic counters and lengths
- Document Hooks
- Environment Hooks

`xpatch` Extends the command patching provided by `etoolbox`

`Silence` Allows me to ignore expected warnings.

`ifdraft` To make it easy to change things up a bit more than usual for draft mode.

`scrlayer-scrpage` To allow for those lovely headers and footers.

`geometry` Loaded with options,

```
452 a4paper, ignoreheadfoot, left=\leftmargin, right=\rightmargin, top=2cm,
    ↪ bottom=3.5cm, headsep=1cm
```

`titlesec` Allows for customisation of `\chapter` etc. Was originally used for all section commands, but now all except for `\chapter` have been transitioned to KOMA-script.

`titletoc` Allows significant tweaking to how the table of contents looks.

`framed` Facilitate the definition of new environments that take multi-line material, wrap it with some non-breakable formatting (some kind of box or decoration) and allow page breaks in the material

`textpos` Facilitates placement of boxes at absolute positions on the LaTeX page. Loaded with options `absolute, overlay`

`hyperref` Used to produce all sorts of hyperlinks in a document. Loaded with option `pdfa`

`hyperxmp` Improves metadata setting with `hyperref`.

`calc` Adds infix expressions to perform arithmetic on the arguments of the LaTeX commands `\setcounter`, `\addtocounter`, `\setlength`, and `\addtolength`

`xcolor` Provides all sorts of colour use and mixing capabilities.

`tikz` It's tikz. You can't draw anything without it.

2.3.3 Text

`microtype` Always good to have. It simply makes text look better, specifically it applies the following,

- Character protrusion
- Font expansion
- Adjustment of interword spacing and kerning

- Letterspacing

Configured with,

```
67 activate={true,nocompatibility}, final, tracking=true, kerning=true,
   ↪ spacing=true, factor=2000
```

plex-serif

plex-sans

plex-mono

`setspace` Provides an easy way to set line spacing with commands such as `\doublespacing` and `\setstretch{1.25}`.

`multicol` Split text into multiple columns (up to 10).

2.3.4 Table-related

`booktabs` Contributes different width `\hline` variants.

`tabularx` Adds the `tabularx` environment which has its width explicitly set, `x` column type which automatically determines its width based on its contents.

`longtable` Provides a good way of allowing tables to spread over multiple pages.

2.3.5 Graphics and Figures

`graphicx` Makes loading images (`includegraphics`) work well.

`grffile` This fixes the fix allowed filenames of `graphicx`.

`caption` Provides many ways to customise the captions in floating environments like `figure` and `table`, and cooperates with many other packages. Facilities include rotating captions, sideways captions, continued captions (for tables or figures that come in several parts). Loaded with option `hycap=true`

`subcaption` Allows for typesetting of sub-figures and sub-tables.

2.3.6 infoBulle

`fontawesome5` Fontawesome 5, need I say any more?

`infoBulle`

`marginInfoBulle`

2.3.7 Chemistry

`mhchem` Useful for simple inline chemistry.

`chemfig` Useful for chemical diagrams.

2.3.8 Code

`minted` Configured as follows,

```

476 \setminted{
477     frame=none,
478     % framesep=2mm,
479     baselinestretch=1.2,
480     fontsize=\footnotesize,
481     highlightcolor=page!95!text!80!primary,
482     linenos,
483     breakanywhere=true,
484     breakautoindent=true,
485     breaklines=true,
486     tabsize=4,
487     xleftmargin=3.5em,
488     autogobble=true,
489     obeytabs=true,
490     python3=true,
491     % texcomments=true,
492     framesep=2mm,
493     breakbefore=\\\.+,
494     breakafter=\\,
495 }
```

`tcolorbox` Used for prettifying the `minted` environment. Loaded with option `many`

2.3.9 Maths Related

These packages are loaded by the `math` option (or one of its derivatives).

`amsmath`, `amssymb` Extends the maths commands and symbols in latex.

`mathdesign` To use the Utopia font for maths symbols.

`xfrac` Allows split level fractions a/b better than `{a}^{b}` can produce.

`cancel` Allows for easy cancelling within maths like so $-x$ and x^0 . Loaded with option `makeroom`

`mathtools` Provides a variety of enhancements to make maths *even* better.

- Extensible symbols, such as brackets, arrows, harpoons, etc.;
- Various symbols such as `\coloneqq` ($:=$);
- Easy creation of new tag forms;
- Showing equation numbers only for referenced equations;
- Extensible arrows, harpoons and hookarrows;
- Starred versions of the amsmath matrix environments for specifying the column alignment;
- More building blocks: multilined, cases-like environments, new gathered environments;
- Maths versions of `\makebox`, `\llap`, `\rlap` etc.;
- Cramped maths styles; and more...

`mathastext` Uses relevant plex font for maths letters. Uses options `basic`, `italic`, `symbolgreek`.

`pgfplots` Loaded by the `plot` option.

2.4 Configuration

2.4.1 Colours

Name	Default (Light)	Default (Dark)
<code>text</code>	<code>#000000</code>	<code>#FCFCFC</code>
<code>page</code>	<code>#FFFFFF</code>	<code>#222222</code>
<code>href</code>	<code>tertiary</code>	<code>secondary</code>
<code>primaryVariant</code>	<code>primary!75!Cream</code> <code>>twheel,-3,360</code>	<code>''</code>
<code>inlinemath</code>	<code>secondary!50!text</code>	<code>tertiary!50!text</code>
<code>infoBulle</code>		
<code>infoBulleBackground</code>	<code>page!90!text</code>	<code>''</code>
<code>infoBulleText</code>	<code>text</code>	<code>''</code>
<code>marginInfoBulleBackground</code>	<code>page</code>	<code>''</code>
<code>marginInfoBulleText</code>	<code>text</code>	<code>''</code>
<code>criticalColor</code>	<code>Red</code>	<code>''</code>
<code>questionColor</code>	<code>Purple</code>	<code>''</code>
<code>informationColor</code>	<code>Green</code>	<code>''</code>
<code>checkColor</code>	<code>Blue</code>	<code>''</code>
<code>warningColor</code>	<code>Orange</code>	<code>''</code>
<code>tipColor</code>	<code>Purple</code>	<code>''</code>
<code>exampleColor</code>	<code>Blue</code>	<code>''</code>
<code>mathematicalColor</code>	<code>Orange</code>	<code>''</code>
<code>codeColor</code>	<code>Grey</code>	<code>''</code>

2.4.2 Palettes

The module `bmc-color` has a few inbuilt pallets sourced from <https://flatuicolors.com>. Using the relevant country code (see website) you can select a palette two ways:

1. Passing the palette as an option when loading, `[palette=<palette code>]`
2. Setting the palette via `\usepalette{<palette code>}`

The second method can be employed anywhere in the document.

Creating Your Own Palette

Palettes can be created with the command `\newpalette{<palette code>}{<colour definitions>}`. The ‘colour definitions’ simply consist of any `xcolor` commands that define the colour. You may wonder why not just define your own macro for your colours – this method provides two minor features that may prove useful depending on the situation.

- ‘Main’ colours (e.g. Red, Green...) not provided are populated from a default (de)
- ‘Derived’ colours (e.g. LightRed, DarkGreen) are mixed from main colours if not given

It is a good idea to provide colours from the following list, however you can provide as many or as few colours as you like.

Black, White, Cream, Grey, Red, Yellow, Blue, Green, Orange, Purple, Cyan.

Note: All except for Black, White, and Cream have the accompanying Light and Dark variants.

Default Palettes

de
German (default)

Grey Red Blue Green Yellow Orange Purple Cyan

fui
FlatUI

Grey Red Blue Green Yellow Orange Purple Cyan

au
Australian

Grey Red Blue Green Yellow Orange Purple Cyan

us
American

Grey Red Blue Green Yellow Orange Purple Cyan

uk
British

Grey Red Blue Green Yellow Orange Purple Cyan

ca
Canadian

Grey Red Blue Green Yellow Orange Purple Cyan

cn
Chinese

Grey Red Blue Green Yellow Orange Purple Cyan

nl
Dutch

Grey Red Blue Green Yellow Orange Purple Cyan

fr
French

Grey Red Blue Green Yellow Orange Purple Cyan

in
Indian

Grey Red Blue Green Yellow Orange Purple Cyan

ru
Russian

Grey Red Blue Green Yellow Orange Purple Cyan

es
Spanish

Grey Red Blue Green Yellow Orange Purple Cyan

se
Swedish

Grey Red Blue Green Yellow Orange Purple Cyan

tr
Turkish

Grey Red Blue Green Yellow Orange Purple Cyan

sol
Solarized

Grey Red Blue Green Yellow Orange Purple Cyan

mon
Monokai

Grey Red Blue Green Yellow Orange Purple Cyan

3

Reasoning



Summary

- 3.1 Layout, 27
- 3.2 Typefaces, 27
- 3.3 Colour, 28

3.1 Layout

3.2 Typefaces

3.2.1 Studies

Unfortunately large type studies on UI/UX and typography seem to be exclusive to large companies that care about this (e.g. microsoft, google, etc.) and don't seem to publish their results. So, I just have a quick look at the little that is easy to find.

It has long been known that there is a relationship between the reader's judged aesthetic pleasingness of text, and the legibility (Tinker and Paterson, 1942). Regardless of the underlying mechanisms, this provides strong justification for putting in the effort to make a document have good content, *and* good aesthetics.

A study by Kaspar, Wehlitz, von Knobelsdorff, Wulf, and von Saldern, 2015 presented subjects with research papers in serif and sans-serif fonts. They found that sans-serif fonts increase reading speed. While this was not a focus of the study, it is apparently in line with previous research. A 2005 study by Gasser, Boeke, Haffernan, and Tan, 2005 found that sans-serif fonts produced 9% improvement in recall ($N = 149, p = 0.05$).

While Kaspar et al., 2015 found that while sans-serif fonts are read faster, in every category tested, sans-serif fonts performed better. Specially serifs were found to improve:

- Comprehension
- Interest in paper and appeal of abstract
- Perceived quality and importance

More generally, Larson, Hazlett, Chaparro, and Picard, 2007 identified two measures that successfully indicate aesthetic difference. Performance of creative cognitive tasks after reading a document, and subconscious activation of the facial muscle responsible for frowning. For both measures a method that had $p = 0.04$ was yielded. It is rather interesting that they found that improved typography reduced frowning and improved creative cognitive task performance.

Semantic associations between the typeface chosen and the content also seem to be important. Kolenda, 2019b appears to give a good overview of this.

3.2.2 The Elements of Typographic Style

While, as demonstrated previously, there are *some* studies about typefaces, they are few and far between, done on a small scale, and rarely reproduced. Hence, one of the best resources for typographic guidances is eminent publications on the matter. *The elements of typographic style* is one of the foremost books, widely held in high esteem. Reading

through it there a several useful suggestions which I have tried to implement here.

Conclusions from ¶ 3.2.2 With some acronyms each letter is pronounced individually (such as the ISO or GNU). To achieve this effect typographically spaced small caps are very useful. They are also useful in a variety of other situations, for example:

WWII formally ended at 12:00 AM on 8 May nineteen forty-five AD.
 WWII formally ended at 12:00 AM on 8 May nineteen forty-five AD.

The general rule given is — *‘For abbreviations and acronyms in the midst of normal text, use spaced small caps’*.

Conclusions from ¶ 4.4.1 With regard to tables

1. Avoid shrinking font size
2. Minimise ‘furniture’ (rules, boxes, dots, etc.)
3. Rules / tint blocks, when absolutely necessary, should run in the reading direction
4. Rules along the (vertical) outside the table are useless
5. Ensure sufficient white space is maintained

Conclusions from ¶ 4.5.1 Short texts, such as research papers generally don’t benefit from much padding at the start and end. The same does *not* apply to books. As such, BMC implements two `\maketitle`s. The default full page one seen on this document, and a shorter one used when the `article` option is passed to the class.

Conclusions from ¶ 5.1.3 Use the ampersand (&) in heads and titles.

Conclusions from ¶ 5.1.4 Line-end hyphens best lie in the right margin. This is achieved by use of the `factor=2000` option with the microtype package.

3.3 Colour

It seems that emotional colour associations are not universal (Laurence, 1991), and it seems like much of the clamour about colour psychology which is presented authoritatively, really has little factual basis, as covered by O’Connor, 2011. However I mainly care about the western european perspective, which I think is the most heavily studied, so we should be alright.

Looking online I saw a lot of self-professed “colour-gurus”, the first respectable-looking page which seemed to provide a nice summary (i.e. wasn’t just red vs. green etc.)

is Kolenda, 2019a — you may recognise the author from the previous section on fonts. His page acknowledges complexity in the situation, and cites a bunch of articles, which I consider to be a good sign. The conclusions were summarised in a table, a copy of which you can find below.

Table 3.1: Colour Associations (from Nick Kolenda)

		Hue											S	L
Arousal	Relaxed				✓	✓	✓				✓	✓	✓	
	Exited	✓	✓	✓						✓		✓	✓	
Processing	Systematic				✓	✓	✓				✓	✓	✓	
	Heuristic	✓	✓	✓						✓		✓	✓	
Selling	Auction	✓	✓	✓						✓		✓	✓	
	Negotiation				✓	✓	✓				✓	✓	✓	
Traits	Competence					✓				✓			✓	
	Excitement	✓	✓	✓								✓	✓	
	Ruggedness				✓				✓			✓	✓	
	Sincerity			✓				✓			✓	✓	✓	
	Sophistication						✓				✓	✓		
Goal	Attention	✓	✓	✓								✓	✓	
	Action	✓	✓	✓								✓	✓	
	Liking				✓	✓	✓					✓		
Type	Utilitarian					✓	✓			✓				
	Hedonic	✓		✓			✓	✓						
Message	Prevention	✓												
	Gain					✓								
Gender	Male				✓	✓								
	Female	✓		✓		✓	✓	✓				✓	✓	

While Table 3.1 is nice, it isn't terribly useful for providing a holistic view of colour in design.

Bibliography

- Bringhurst, R. (2004). *The elements of typographic style*. Hartley and Marks, Publishers.
- Gasser, M., Boeke, J., Haffernan, M., & Tan, R. (2005). The influence of font type on information recall. *North American Journal of Psychology*, 7(2), 181–188.
- Kaspar, K., Wehlitz, T., von Knobelsdorff, S., Wulf, T., & von Saldern, M. A. O. (2015). A matter of font type: The effect of serifs on the evaluation of scientific abstracts. *International Journal of Psychology*, 50(5), 372–378.
- Kolenda, N. (2019a, October). Color psychology: An enormous guide. Retrieved from <https://www.nickkolenda.com/color-psychology>
- Kolenda, N. (2019b, July). How to choose the right font (according to science). Retrieved from <https://www.nickkolenda.com/font-psychology/>
- Larson, K., Hazlett, R. L., Chaparro, B. S., & Picard, R. W. (2007). Measuring the aesthetics of reading. In *People and computers xx—engage* (pp. 41–56). Springer.
- Laurence, J. (1991, January 1). Cross-cultural colour comparisons: Global marketers beware! 8(3). doi:10.1108/02651339110137279
- O'Connor, Z. (2011, April 11). Colour psychology and colour therapy: Caveat emptor. *Color Research and Application*, 36(3), 229, 234. Retrieved from <http://search.proquest.com/docview/889394375/>
- Tinker, M. A., & Paterson, D. G. (1942, February). Reader preferences and typography. *Journal of Applied Psychology*, 26(1), 38, 40.