Publication reference
DOI: xxx/xx

Demo Stylish Article

John Doe [1,2], His Friend, Athird One[3*]

Abstract
This document is only a demo explaining how to use the Stylish Article template.

Keywords
template, demo

[1]Dummy Organization, Somewhere, City, Country, 9999,
[2]Demo Journal,
[3]Third Affiliation,
***Corresponding Author**: at@x.com, https://www.x.com

Table of contents

1. Introduction

This template allows writing stylish articles in Markdown with Quarto[1]. It directly produces well-formatted PDF articles for self-archiving or in other formats, for example HTML.

2. Markdown

Markdown is a very simple language for producing various types of documents: HTML, PDF, and Word among others. Its documentation is available on the Quarto website[2].

This document was made RStudio, but other tools are available[3]: Quarto processes the Markdown code, passes it to Pandoc for transformation into LaTeX based on the Stylish Article template provided in this Quarto extension; finally LaTeX compiles it into PDF.

[1]https://quarto.org/
[2]https://quarto.org/docs/authoring/markdown-basics.html
[3]https://quarto.org/docs/get-started/

2.1 Motivation

Markdown is very easy to learn.

Markdown allows integrating your R, Python and more code for a *reproducible* result.

Markdown allows to produce, without rewriting the text, a document in different formats: HTML, LaTeX or Word for example.

3. Code

The main features of Markdown are summarized here. The full documentation is online[4].

3.1 R and other code

R code is included in code chunks:

```
head(cars)
```

```
  speed dist
1     4    2
2     4   10
3     7    4
4     7   22
5     8   16
6     9   10
```

Similar chunks support other languages such as Python. This template focuses on R but can be used with any other code language supported by Quarto.

3.2 Figures

Figures can be created by R code (Figure 1). A label is associated with each figure: its name is given in the `#| label:` option of the chunk that produces it. It must start with `fig-`. Cross-references are made with the command `@label`, e.g.: `@fig-pressure`.

[4]https://quarto.org/docs/authoring/markdown-basics.html/

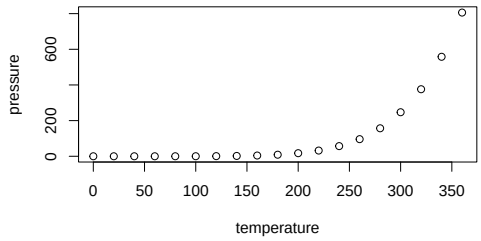

Figure 1. Figure *title* with maths : π and cross-reference to Table 1

In PDF, a figure can use the full width of the page by adding the following options to the header of the code chunk that generates it: 'fig.env="figure*"'.

Existing figures are integrated into a piece of code by the `include_graphics()` function, see Figure 2.

Figure 2. Logo

Systematically place these files in the `images` folder for the automation of GitHub pages.

3.3 Tables

The horizontal `-` and vertical separators `|` allow you to draw a table according to Markdown syntax, but this is not the best method in R.

Tables can also be produced by R code. The content of the table is in a dataframe. The `kable()` function in the *knitr* package prepares the table for display and passes the result to the `kable_styling` function in the *kableExtra* package for final formatting.

```
names(iris) <- c("Sepal length", "Width", "Petal length",
                 "Width", "Species")
knitr::kable(
  head(iris), longtable = TRUE, booktabs = TRUE
) %>%
  kableExtra::kable_styling(bootstrap_options = "striped")
```

The caption is specified by the `#| tbl-cap:` chunk option. At the moment, it is restricted to simple text without format, math or cross references (Table 1), in contrast with figure captions (see Figure 1).

Table 1. Table created by R, with \pi. Not correctly formated as of Quarto v. 1.2.

Sepal length	Width	Petal length	Width	Species
5.1	3.5	1.4	0.2	setosa
4.9	3.0	1.4	0.2	setosa
4.7	3.2	1.3	0.2	setosa
4.6	3.1	1.5	0.2	setosa
5.0	3.6	1.4	0.2	setosa
5.4	3.9	1.7	0.4	setosa

Always use the `booktabs = TRUE` argument so that the separator lines are optimal in LaTeX. The `bootstrap_options = "striped"` option provides more readable tables in HTML.

In LaTeX, `longtable = TRUE` selects the 'longtable' package to format tables. Such tables are placed in the text. Due to limits of the 'longtable' package which does not support multi-column layouts, the Stylish Article template uses a workaround that does not allow really long tables to be split into several columns or pages. So, `longtable = TRUE` only means that the table will be one-column wide and located in the text if possible. If not, it will float.

To use the full width of the page, like Table 2, `longtable` is set to `FALSE` and `table.envir = "table"` is added in the aguments of `kable()`.

This table contains mathematics: the `escape = FALSE` argument is necessary in `kable()`.

Finally, the `full_width = TRUE` argument in `kable_styling()` adjusts the width of the table to the available width. It must be set for correct formatting of two-column tables in LaTeX.

Note that tables can't be shown on the first page of the PDF output of the Stylish Article template: it would conflict with the table of contents.

3.4 Lists

Lists are indicated by `*`, `+` and `-` (three hierarchical levels) or numbers `1.`, `i.` and `A.` (numbered lists). Indentation of lists indicates their level: `*`, `+` and `-` may be replaced by `-` at all levels, but four spaces are needed to nest a list into another.

Table 2. Large table with maths, on two columns

Treatment	Timber	Thinning	Fuelwood	%AGB lost
Control				0
T1	DBH $\geq$ 50 cm, commercial species, $\approx$ 10 trees/ha			[12% – 33%]
T2	DBH $\geq$ 50 cm, commercial species, $\approx$ 10 trees/ha	DBH $\geq$ 40 cm, non-valuable species, $\approx$ 30 trees/ha		[33% – 56%]
T3	DBH $\geq$ 50 cm, commercial species, $\approx$ 10 trees/ha	DBH $\geq$ 50 cm, non-valuable species, $\approx$ 15 trees/ha	40 cm $\leq$ DBH $\leq$ 50 cm, non-valuable species, $\approx$ 15 trees/ha	[35% – 56%]

- First element of a list
  - sub-list
- Second element
- Continuation of the list

Leave an empty line before and after the list, but not between its items.

3.5 Math

Equations in LaTeX format can be inserted inline, like $A = \pi r^2$ or isolated like

$$e^{i\pi} = -1.$$

They can be numbered, see Equation 1, after adding them a label:

$$A = \pi r^2. \tag{1}$$

3.6 Cross-references

Figures and tables have a label declared in their code chunk option `tbl.cap` or `fig.cap`, starting with `fig-` or `tbl-`.

For equations, the label is added manually by the code `{#eq-xxx}` after the end of the equation.

Sections can be tagged by ending their title with `{#sec-yyy}`.

In all cases, the call to the reference is made by `@`.

3.7 Bibliography

Bibliographic references included in the `.bib` file declared in the document header can be called by `[@CitationKey]`, in parentheses (Xie et al., 2018), or without square brackets, in the text, as Xie (2016).

The bibliography is processed by Pandoc when producing Word or HTML documents. The bibliographic style can be specified, by adding the line

```
csl: file_name.csl
```

in the document header and copying the *.csl* style file to the project folder. More than a thousand styles are available[5] and their URL can be used instead of copying the file, e.g.:

```
csl: https://www.zotero.org/styles/xxx
```

For PDF documents, the bibliography is managed by natbib. The style is declared in the header:

```
biblio-style: chicago
```

It can be changed as long as the appropriate `.bst` file (by default: `chicago.bst`) is included in the project.

3.8 Forcing line breaks

Hyphenation is handled automatically in LaTeX. If a word is not hyphenated correctly, add its hyphenation in the preamble of the file with the command `hyphenation` (words are separated by spaces, hyphenation locations are represented by dashes).

If LaTeX can't find a solution for the line break, for example because some code is too long a non-breaking block, add the LaTeX command `\break` to the line break location. Do not leave a space before the command. The HTML document ignores LaTeX commands.

3.9 Languages

Languages are declared in the document header.

The main language of the document (`lang`) changes the name of some elements, such as the table of contents. The change of language in the document

[5] https://github.com/citation-style-language/styles

(one of `otherlangs`) is managed in LaTeX (but not fully in HTML).

For a single word, to ensure correct hyphenation in LaTeX, use the following command, ignored un HTML:

```
\foreignlanguage{italian}{ciao}
```

For a paragraph, to also ensure correct quotes and punctuation spacing, use

```
::: {lang=fr}
"Bonjour" en français!
:::
```

to obtain:

« Bonjour » en français!

The current language has an effect only in LaTeX output: a space is added before double punctuation in French, the size of spaces is larger at the beginning of sentences in English, etc.

Language codes are used in the header, such as `en-US` but language names are necessary in LaTeX. Name matches are listed in table 3 of the polyglossia package documentation[6]. Note that this template uses the 'babel' package rather than 'polyglossia'.

3.10 Text color

This template includes the 'color-text' filter for Quarto. It makes colored text possible using the `[content]{color=<name>}` syntax. This is a red example. Defined colors are blue, green and red. To add more colors, they must be defined in the document header in `format:`, `stylisharticle-pdf:`, `header-includes:` . This is the code for 'grey':

```
    \definecolor{grey}{RGB}{191, 191, 191}
```

They must also be included in a `.css` file declared in `format:`, `stylisharticle-html:`

```
    css: colors.css
```

In this css, colors are declared by

```
.color-grey {color: rgb(191, 191, 191);}
```

References

Xie, Y. (2016). *bookdown: Authoring Books and Technical Documents with R Markdown*. Boca Raton, Florida: Chapman and Hall/CRC.

Xie, Y., J. Allaire, and G. Grolemund (2018). *R Markdown: The Definitive Guide*. Boca Raton, Florida: Chapman and Hall/CRC.

[6] http://mirrors.ctan.org/macros/unicodetex/latex/polyglossia/polyglossia.pdf