

Making handout

to support your talk

Maïeul Rouquette Version 1.6.0
maieul <at> maieul <dot> net 21/04/2017

Contents

1	Introduction	1	5	Recopying the sectioning commands in the handout	3
1.1	Goal	1	6	Numbering the examples	3
1.2	Credits	1	7	Temporarily disabling the handout generation	4
1.3	French tutorial	2	8	Advanced use	4
2	Basis	2	8.1	Conditional code	4
3	Code to be run at the beginning / at the end of the handout	2	8.2	Insert code in the handout	4
4	Putting all your examples in a single folder	3	9	Change history	5

1 Introduction

1.1 Goal

In some fields of scholarship, a beamer does not offer good support when giving a talk. For example, in classical philology, the main sources are text, and it will be better to distribute a handout to the audience with extracts of the texts about which we will talk.

The package supports preparation of such handouts when writing the talk.

1.2 Credits

This package was created for Maïeul Rouquette's PHD¹ in 2014. It is licenced under the *LaTeX Project Public Licence*.²

All issues can be submitted, in French or English, on the GitHub issues page.³

¹<http://apocryphes.hypothese.org>.

²<http://latex-project.org/lppl/lppl-1-3c.html>.

³<https://github.com/maieul/handout/issues>.

1.3 French tutorial

As the idea for the package behavior came from French \LaTeX users⁴, and as the package's author speaks French as a native language, a French tutorial is available in <http://geekographie.maieul.net/136>.

All files in the examples' folder are in French, but the effect of the commands they use should still be clear. They have to be run with $X_{\text{e}}\LaTeX$.

2 Basis

The package can be loaded very quickly with the standard command `\usepackage`

```
\usepackage{handout}
```

+The idea of this package is to prepare a handout during the writing of a paper. When you want to add something to your handout, just write it in an external file, and call this external file with the command `\handout`:

```
Your text  
\handout{folder/example}  
Your text
```

The PDF output will contain two parts:

1. Your paper.
2. The handout.

You have just to split your pdf in two parts to obtain your handout for your audience. See example 1.

3 Code to be run at the beginning / at the end of the handout

You can execute any code at the beginning of the handout by defining a command `\beforehandout`. For example, to suppress the extra margins of the quotation environment:

```
\newcommand{\beforehandout}{%  
  \renewenvironment{quotation}{}{}}%  
}
```

See example 2.

Note that some actions are always performed before the handout:

⁴<http://fr.comp.text.tex.narkive.com/pXMop2kE/fabrication-d-un-exemplier>.

- Start a new page.
- Reset page, footnote, table and figure counters.
- Disable `\index` and `\label` command.

You can also execute any code at the end of the handout by defining a command `\afterhandout`. For example, if you want to add additional remarks on your handout, but not in your main text.

```
\newcommand{\afterhandout}{%
  Additional remarks.
}
```

4 Putting all your examples in a single folder

In most cases, all your external files will be in a single folder. So you can fix this folder with the package's option `dir`:

```
\usepackage[dir=folder]{handout}
...
Your text
\handout{example}
Your text
```

See example 3.

5 Recopying the sectioning commands in the handout

If you want your handout be organized with the same sectioning commands as your main paper, you can use the option `sectioning` when loading the package. See example 4.

6 Numbering the examples

The package option `numbering` automatically numbers the examples. Each example's number is by default printed in the margin, in a frame. See example 5.

To customize number's presentation, you can redefine three commands:

1. `\thehandoutnumber` to redefine the way the number is displayed. Standard is:

```
\renewcommand{\thehandoutnumber}{\fbox{\arabic{handoutnumber}}}
```

See [L^AT_EX](#) documentation on counter's appearance.

2. `\handoutnumber` to redefine the position of the number. Standard is:

```
\newcommand{\handoutnumber}[1]{\marginpar{#1}}
```

3. `\handoutnumberintxt`, to redefine the position of the number in the main text. Standard is:

```
\newcommand{\handoutnumberintxt}[1]{\handoutnumber{#1}}
```

7 Temporarily disabling the handout generation

The package option `disabled` allows you to disable the handout's generation.

You can also disable it for only some parts of the text: use `\disablehandout` to disable the handout's generation from the current point and `\enablehandout` to enable it from the current point.

You can also use `printing=false`, which makes \LaTeX number the examples without producing the handout.

8 Advanced use

8.1 Conditional code

The argument of a `\onlyhandout` command will be printed only in the handout. Conversely, the argument of a command `\nothandout` will be printed only in the main text. See example 6.

8.2 Insert code in the handout

8.2.1 `\forhandout` and `\forhandout*`

The argument of the `\forhandout` command will be added to the handout, even if used outside any included file.

All the commands inside will be expanded, except the one which starts argument..

The `\forhandout` command is quite complex to manage, but can be useful if you want to insert bibliographic references automatically into your handout. If you use *biblatex* to manage your bibliography, you can define a `\citehandout` command with this code:

```
\newcommand{\citehandout}{%
  \AtNextCitekey{%
    \forhandout{%
      \beforehandoutref%
    }%
  }%
  \forhandout{%
    \cite[\strfield{postnote}]{\strfield{entrykey}}%
  }%
}
```

```

\forhandout{%
  \afterhandoutref%
}%
}%
}
\newcommand{\beforehandoutref}{\par\noindent\hspace{-2\parindent}}
\newcommand{\afterhandoutref}{\par\vskip0.25\baselineskip}

```

You can call the `\citehandout` command before citation commands when you want the reference to be copied to the handout. See example 7 and the *biblatex* handbook.

Note that the citation tracker is automatically reset at the beginning of the handout.

The starred version of `\citehandout` command will not expand any macro inside the argument. For example, if you need to write instructions for readers:

```

\forhandout{Read the three following texts, looking for:
\begin{itemize}
^^I\item Places.
^^I\item Topics.
^^I\item Characters.
\end{itemize}
}

```

8.2.2 `\AtEveryHandout`

You can also add define a `\AtEveryHandout` command to be executed in the handout, before each content added to the handout.

9 Change history

1.6.0 2017-04-21

Add starred version of `\forhandout` macro 8.2.2 .

1.5.0 2016-11-15

Reset automatically table and figure counters before typesetting handout.

1.4.0 2016-10-24

Add possibility to add additional content before every element of an handout.

1.3.0 2016-10-13

Add possibility to add additional content at the end of the handout.

1.2.1 2015-01-06

Optimize performance with sectioning option.

1.2.0 2015-01-06

Add new option printing. 7

Optimize performance.

1.1.0 2014-12-15

Add `\enablehandout` and `\disablehandout`. 7

1.0.1 2014-05-23

Also disable indexing in the handout when using multiple index.

1.0.0 2014-03-20

First public release.