

# easyformat

*the* manual for the 2017/06/23 version (v1.5.0)

Evert Provoost

## Abstract

easyformat is a package that allows the use of `_` to begin/end *italics* or **boldface** and `^` to begin/end SMALLCAPS. As an alternative to the standard L<sup>A</sup>T<sub>E</sub>X `\textit{italic}`, `\textbf{bold}` and `\textsc{smallcaps}`.

## Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Usage, syntax and examples</b>	<b>2</b>
<b>3</b>	<b>Troubleshooting</b>	<b>3</b>
<b>4</b>	<b>Macros</b>	<b>3</b>
4.1	Special characters . . . . .	3
4.2	easyformat-syntax . . . . .	3
<b>5</b>	<b>Technical details</b>	<b>4</b>
5.1	Why <code>_</code> and not <code>*</code> ? . . . . .	4
5.2	Does this break mathmode? . . . . .	4
<b>6</b>	<b>Changelog</b>	<b>4</b>
<b>7</b>	<b>Contact</b>	<b>5</b>

## 1 Introduction

Usually when you want to make something *italic* or **bold** in L<sup>A</sup>T<sub>E</sub>X you insert `\textit{italic}` or `\textbf{bold}` respectively.

However when writing a book or another text where this formatting is common, this quickly becomes annoying to type.

### Quick disclaimer:

I'm not saying that the L<sup>A</sup>T<sub>E</sub>X way is bad. I just want to explain that it's not ideal under all circumstances.

Simple markuplanguages (eg. Markdown) have a more elegant solution. If you, for example, wanted to write:

**Stop!** You *have* to wait!

In Markdown, you'd type:

**\*\*Stop!\*\*** You **\*have\*** to wait!

As you can see it's a matter of *wanting* italic or bold formatting, adding \*'s where needed and done; no (long) macros to type.

With the help of `easyformat` we can do something similar, here we get the same result with:

`--Stop!--` You `_have_` to wait!

(Why underscores? See: 5.1)

And since it is good practice to add abbreviations using SMALLCAPS, `easyformat` gives you `^smallcaps^`.

So:

`^nasa^` and `^esa^` are probably the best known space agencies.

Results in:

NASA and ESA are probably the best known space agencies.

## 2 Usage, syntax and examples

Like usual you import the package with: `\usepackage{easyformat}`

`easyformat` makes `_` an active character, however it was already 'special' before so you still have to type `\_` if you want to insert an underscore.

The same can be said of `^`, however since this one is difficult to add in ordinary L<sup>A</sup>T<sub>E</sub>X, the `easyformat` package gives you `\cir` to get `^`.

As already said, `easyformat` uses a Markdown-like syntax, so:

```
_italic,_ __bold,__ ___bolditalic___ and ^smallcaps!^
```

Gives:

*italic*, **bold**, ***bolditalic*** and SMALLCAPS!

However we can also do more complex stuff, eg.:

```
_Lorem __ipsum_ totalem__ ^da^ __givea _zin_ doram__  
_zet_tim, liefkan.
```

Becomes:

*Lorem ipsum totalem* DA **givea zin** *doram zettim*, liefkan.

Those with a good eye might have noticed that `easyformat` adds italics correction when needed. Just like `\textit{zet}tim` would have.

## 3 Troubleshooting

`easyformat` sometimes conflicts with the loading of other packages (eg. `babel`, `fontspec` and `microtype`). This is fixed by importing `easyformat` after those packages.

“For some or other reason I get some weird formatting...”

It could be a bug—in which case I’d love to hear from you (see 7) —however, you probably forgot to close your formatting somewhere and because of how `easyformat` works you can get weird results.

## 4 Macros

### 4.1 Special characters

---

`\cir` `\cir`

Insert a circumflex (^) character.

### 4.2 `easyformat-syntax`

---

`\enableeasyformat`  
`\disableeasyformat`

`\enable/disableeasyformat`  
Enables respectively disables the `easyformat-syntax`.

## 5 Technical details

### 5.1 Why `_` and not `*`?

To make `*` work we would have to make it an active character, however this would break things like `\section*{Art}`. The `_` is already protected by  $\TeX$  because of its meaning in mathmode. However, it has little to no use in textmode. This is why we can use `_` for this purpose and not `*`.

### 5.2 Does this break mathmode?

No it doesn't. `_` and `^` keep behaving like before in mathmode, their meaning only changes in textmode. (We first check whether we are in mathmode or not and then behave accordingly.)

## 6 Changelog

We only include important changes from v1.0.0 onwards, since earlier versions barely worked.

### 2017/06/23 v1.5.0

Fixed italics-correction in front of text.

### 2017/06/03 v1.4.0

Improved code readability. We now fully use `expl3`, which eliminates most of the possible future issues with  $\LaTeX$ 3. Fixed a kerning issue with `\cir` in mathmode. Removed: `\setundact`, `\setundsub`, `\setciract`, `\setcirsup`, `\nrfamily`, `\nrshape`, `\nrseries`, `\setfffamily`, `\setffshape` and `\setffseries` as these do not add any value to the package.

### 2017/05/28 v1.3.0

Removed forced re-enabling of the `easyformat`-syntax at the start of the document. Rewritten to use `expl3`, this makes the code shorter and a future move to  $\LaTeX$ 3 easier. Fixed an error that could occur with `\cir`. `easyformat` now also behaves correctly when a fontpackage is loaded.

### 2017/04/17 v1.2.0

Improved handling of mixed styles. Added `^smallcaps` for `SMALLCAPS`.

### **2017/04/09 v1.1.0**

Made usage of syntax in `\section{}`, and similar macros, possible. Changed the behaviour so it works more like `\emph{}` (this also simplifies the code).

### **2017/04/07 v1.0.0**

First stable version. It made ***bold-italics*** possible.

## **7 Contact**

If you want to make suggestions or have any questions whose answer could be included in a future version of this document, you can email to this address: <mailto:evert.provoost@gmail.com>

Made with ♡ in Berlaar, Belgium.