The **noindentafter** package[*]

Michiel Helvensteijn

mhelvens+latex@gmail.com

November 30, 2014

Development of this package is organized at latex-noindentafter.googlecode.com.
I am happy to receive feedback there!

---

1 Introduction

Some people prefer certain LaTeX environments —such as `itemize` or `theorem`—
never to be followed by an indented paragraph. It is of course possible to manually
add the `\noindent` command to each occurrence, but it's easy to forget to do this.
Also, all those `\noindent` commands leave your TeX code just a little bit more
chaotic and more burdensome to read.

There are a number of solutions floating around, but many of them are not
particularly robust, or only work in specific cases.

The `noindentafter` package provides a few simple commands which can be
used to patch existing macros and environments. They work as expected in all
cases I've encountered.

2 Usage

`\NoIndentAfterThis`   This is the most primitive macro offered by the package. It forces a paragraph
break and suppresses indentation for whatever follows:

```
Lorem ipsum dolor sit amet, consectetur adipiscing
elit. \NoIndentAfterThis Pellentesque hendrerit volutpat
feugiat. Ut purus leo, pulvinar sit amet vehicula non,
pulvinar eu lacus. Curabitur id mollis ligula.
```

Lorem ipsum dolor sit amet, consectetur adipiscing elit.
Pellentesque hendrerit volutpat feugiat. Ut purus leo, pulvinar sit amet vehicula non, pulvinar eu lacus. Curabitur id mollis ligula.

---

[*]This document corresponds to noindentafter 0.2.2, dated 2014/11/30.

`\NoIndentAfterEnv` `{⟨environment⟩}`

This is probably the command most people will want to use. The ⟨environment⟩ argument should contain an environment name. After using this command, any paragraph following such an environment will remain unindented.

```
Lorem ipsum dolor sit amet, consectetur adipiscing elit.

\begin{itemize}
  \item First Item
  \item Second Item
\end{itemize}

Pellentesque hendrerit volutpat feugiat. Ut purus
leo, pulvinar sit amet vehicula non, pulvinar eu
lacus. Curabitur id mollis ligula.

\NoIndentAfterEnv{itemize}

Vestibulum id erat venenatis, facilisis enim non,
consectetur felis. Quisque iaculis eu arcu non pretium.

\begin{itemize}
  \item Third Item
  \item Fourth Item
\end{itemize}

Curabitur est elit, posuere pulvinar laoreet sed, varius
id mi. Nam lobortis elit nec mauris condimentum gravida.
```

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

- First Item

- Second Item

Pellentesque hendrerit volutpat feugiat. Ut purus leo, pulvinar sit amet vehicula non, pulvinar eu lacus. Curabitur id mollis ligula.
Vestibulum id erat venenatis, facilisis enim non, consectetur felis. Quisque iaculis eu arcu non pretium.

- Third Item

- Fourth Item

Curabitur est elit, posuere pulvinar laoreet sed, varius id mi. Nam lobortis elit nec mauris condimentum gravida.

You'll probably want to use these commands in the document preamble but, as you can see, you don't really have to.

Note that using the `\NoIndentAfterEnv` command on your own environment is not the same as adding `\NoIndentAfterThis` to the end of it:

```
\newenvironment{test}{\itshape}{\NoIndentAfterThis}

\begin{test}
    Lorem ipsum dolor sit amet, consectetur
    adipiscing elit.
\end{test}

Pellentesque hendrerit volutpat feugiat. Ut purus
leo, pulvinar sit amet vehicula non, pulvinar eu
lacus. Curabitur id mollis ligula.
```

Lorem ipsum dolor sit amet, consectetur adipiscing elit.
    Pellentesque hendrerit volutpat feugiat. Ut purus leo, pulvinar sit amet vehicula non, pulvinar eu lacus. Curabitur id mollis ligula.

`\NoIndentAfterEnv` works because it bypasses the group that surrounds every LATEX environment (and localizes declarations).

`\NoIndentAfterCmd` `{⟨command⟩}`

Finally, you may also patch command sequences using the `\NoIndentAfterCmd` command. There may not be many situations where this is useful, but for me, there is at least one:

```
Lorem ipsum dolor sit amet, consectetur adipiscing elit:

\[ 1 + 1 = 2 \]

Pellentesque hendrerit volutpat feugiat. Ut purus leo:

\NoIndentAfterCmd \]

\[ 2 + 2 = 4 \]

Vestibulum id erat venenatis, facilisis enim non,
consectetur felis.
```

Lorem ipsum dolor sit amet, consectetur adipiscing elit:

$$1 + 1 = 2$$

Pellentesque hendrerit volutpat feugiat. Ut purus leo:

$$2 + 2 = 4$$

Vestibulum id erat venenatis, facilisis enim non, consectetur felis.

Note that braces around the ⟨*command*⟩ argument are optional, as always when an argument in TEX consists of only a single token.

3   Acknowledgements

I had been looking for a solution like this for quite a while. I came up with a number of hacks that workedin some situations but not in others. Eventually I asked a question on http://tex.stackexchange.com:

- http://tex.stackexchange.com/questions/112404

I received two excellent answers – one from cgnieder and one from lockstep. This package was originally based on the technique proposed by cgnieder, as it is simpler and more generally applicable. Still, the answer by lockstep is definitely worth a read.

Most recently, a better approach was implemented by tudscr to fix the spacing above headers:

- https://github.com/mhelvens/latex-noindentafter/pull/1

4   Implementation

We now show and explain the entire implementation from `noindentafter.sty`.

4.1   Package Info

```
1 \NeedsTeXFormat{LaTeX2e}
2 \ProvidesPackage{noindentafter}[2014/11/30 0.2.2
3   prevent paragraph indentation after specific environments or macros]
```

4.2   Packages

```
4 \RequirePackage{etoolbox}
```

4.3   Patches

\end   The package `etoolbox` provides the command `\AfterEndEnvironment` which creates a hook executed at a very late point inside the `\end` command. However, this hook is still located before `\ignorespaces`, which is too early to properly suppress the indention after an environment. Therefore another hook is now added to `\end` using `\patchcmd`. This new hook puts new code at the very end.

```
5 \patchcmd\end{%
6   \if@ignore\@ignorefalse\ignorespaces\fi%
7 }{%
8   \if@ignore\@ignorefalse\ignorespaces\fi%
9   \csuse{@noindent@#1@hook}%
10 }{}{%
11   \PackageWarningNoLine{noindentafter}{%
12     Patching '\string\end' failed!\MessageBreak%
13     '\string\NoIndentAfter...' commands won't work%
14   }%
15 }
```

4.4 Macros

`\@NoIndentAfter` This command implements the main principle behind this package. It checks whether it is followed by a paragraph. If so, the command `\par` is temporarily changed using `\everypar`, so that the following paragraph is not indented. Immediately afterwards, default paragraph behavior is restored with `\@restorepar` (from the LaTeX base).

```
16 \newcommand*\@NoIndentAfter{%
17   \@ifnextchar\par{%
18     \def\par{%
19       \everypar{\setbox\z@\lastbox\everypar{}}%
20       \@restorepar%
21     }%
22   }{}%
23 }
```

`\NoIndentAfterThis` Enforce a paragraph break and suppress indentation for whatever follows.

```
24 \newrobustcmd*{\NoIndentAfterThis}{\@NoIndentAfter\par\par}
```

`\NoIndentAfterEnv` {⟨environment⟩}

Append `\@NoIndentAfter` to the output of ⟨environment⟩ by using the new environment hook.

```
25 \newrobustcmd{\NoIndentAfterEnv}[1]{%
26   \csdef{@noindent@#1@hook}{\@NoIndentAfter}%
27 }
```

`\NoIndentAfterCmd` {⟨command⟩}

Append `\NoIndentAfterThis` to the output of ⟨command⟩.

```
28 \newrobustcmd*{\NoIndentAfterCmd}[1]{%
29   \apptocmd{#1}{\NoIndentAfterThis}{}{%
30     \PackageWarning{noindentafter}{%
31       Patching '\string#1' failed!\MessageBreak%
32       '\string\NoIndentAfterCmd' won't work%
33     }%
34   }%
35 }
```

Change History

Index

Numbers written in italic refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in roman refer to the code lines where the entry is used.