

includeRnw - manual

v0.1.0

(build 11)

Andreas Strauman

May 1, 2018

This package is for including .Rnw-files in normal .tex-files. It invokes the command R from your command line.

If you found any bugs or want new functionality, to contribute, view the commented source, get latest version of this package or get in touch with me, you can do all of that at
<https://github.com/Strauman/includeRnw/>. If you have questions of functionality, kindly direct them to the community
<http://tex.stackexchange.com>. The author is active on this site regularly.

Contents

1	Quick start	2
2	Reference	2
2.1	Reference	2
2.2	Command Options	2
2.3	Package Options	3
2.4	User macros	4
2.5	List of all macros	5
3	Changelog	5

1 Quick start

```
\documentclass{article}
\usepackage{includeRnw}
\begin{document}
\includeRnw{path/to/my.Rnw}
\end{document}
```

The Rnw-file:

```
<<pChunk, echo=F, cache=F >>=
  read_chunk('r/.r')
@
```



Path to **r**-files are relative to the path of the **.Rnw**-file

2 Reference

2.1 Reference

\includeRnw[*irw-options*]{*path/to/file.Rnw*}

Uv0.1.0
2018/05/01

Compiles .Rnw-files using R. Assumes that R can be called from the command line. It is optional to add the extension .Rnw

2.2 Command Options

Options to use with **\includeRnw**

h

Use the **h** to prevent **\includeRnw** from actually knitting the file, but only include the knitted **.tex** file

f

f forces **\includeRnw** to actually knitting the file no matter, what (as long as it exists).

2.3 Package Options

\usepackage[[(*options*)] where the *options* are:

halt

If you do \usepackage[halt]{includeRnw}, then \includeRnw{my/file.Rnw} will *not* run knitr on it. However, if the knitted file exists, it will include this. You can override this with the f-option in the \includeRnw: \includeRnw[f]{my/file.Rnw}

new

This option would make the \includeRnw only run knitr on the file if the knitted file does not exist.

noknithead

This option will prevent \includeRnw from building and including the knitr-preamble.

texenv

Using the **texenv** option disables the default Rnw syntax and allows for a new one. See example below

The two following examples do the exact same thing:

This example use standard Rnw:

```
\documentclass[a4paper]{article}
\usepackage{includeRnw}
\begin{document}
<<pChunk, echo=F, cache=F >>=
  read_chunk('r/myRScript.r')
@
\end{document}
```

The following chunk uses the **texenv** option:

```
\documentclass[a4paper]{article}
\usepackage[texenv]{includeRnw}
\begin{document}
\begin{Rnw}{myChunk, echo=F, cache=F}
  read_chunk('r/myRScript.r')
\end{Rnw}
\end{document}
```

2.4 User macros

The default user macro settables are

```
\rnwInputDirectory{.}
\rnwKnittedSuffix{knitted}
\rnwKnitlogFile{\rnw@dir@output/knitlog.log}
\rnwKnitheadName{\rnw@dir@output/knithead}
```

Where `\rnw@dir@output` is the value of the set `\rnwOutputDirectory`

\rnwInputDirectory{<directory>}

This command sets what directory `\includeRnw` will be using. If you e.g. have all the Rnw-files inside a directory called `myRnws` then `\rnwInputDirectory{myRnws}` would fix this.

\rnwKnittedSuffix{<text>}

Set the default suffix of the filename of the knitted output

\rnwKnitlogFile{<filename>}

Sets the filename of the log output by knitr.

\rnwKnitheadName{<filename>}

Sets the filename of the preamble generated by knitr.

2.5 List of all macros

```
\includeRnw→ P. 2
\rnwInputDirectory
\rnwKnitheadName
\rnwKnitlogFile
\rnwKnittedSuffix
```

3 Changelog

v0.0.1 2018/04/20

- Created the package

v0.0.2 2018/04/28

- Fixed wrong error message on no shell-escape
- Added macros for setting custom values
- Added `noknithead`^{→ P. 3}

v0.1.0 2018/05/01

- Added `texenv`^{→ P. 3}
- Fixed bug where `new`^{→ P. 3} and `halt`^{→ P. 3} package options didn't work
- Fixed bug where `??`^{→ P. ??}, `h`^{→ P. 2} and `f`^{→ P. 2} command options didn't work