# R Markdown : : **CHEAT SHEET**

What is R Markdown?

.Rmd files · An R Markdown (.Rmd) file is a record of your research. It contains the code that a scientist needs to reproduce your work along with the narration that a reader needs to understand your work.

Reproducible Research · At the click of a button, or the type of a command, you can rerun the code in an R Markdown file to reproduce your work and export the results as a finished report.

Dynamic Documents · You can choose to export the finished report in a variety of formats, including html, pdf, MS Word, or RTF documents; html or pdf based slides, Notebooks, and more.

Workflow
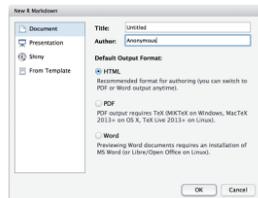
❶ **Open a new .Rmd file** at File ▶ New File ▶ R Markdown. Use the wizard that opens to pre-populate the file with a template

❷ **Write document** by editing template

❸ **Knit document to create report**; use knit button or **render()** to knit

❹ **Preview Output** in IDE window

❺ **Publish** (optional) to web server

❻ **Examine build log** in R Markdown console

❼ **Use output file** that is saved along side .Rmd

---

report.Rmd

```
1  ---
2  title: "R Markdown"
3  author: "RStudio"
4  output:
5    html_document:
6      toc: TRUE
7  ---
8
9  ```{r setup, include=FALSE}
10 knitr::opts_chunk$set(echo = TRUE)
11 ```
12
13 ## R Markdown
14
15 This is an R Markdown document.
16 Markdown is a simple formatting
17 syntax for authoring HTML, PDF,
18 and MS Word documents.
19
20 ```{r cars}
21 summary(cars)
22 ```
23
24 For more details on using R Markdown
25 see <http://rmarkdown.rstudio.com>.
```

labels: ① ② set preview location · insert code chunk · go to code chunk · run code chunk(s) · publish · Title · setup · R Markdown cars · show outline · run all previous chunks · modify chunk options · run current chunk

Console:
```
~/Desktop/R-Markdown-Cheatsheet/
> library(rmarkdown)
> render("report.Rmd", output_file = "report.html")
```

File path to output document
~/Desktop/R-Markdown-Cheatsheet/report.html

report.html · Open in Browser · Find ❺ · Publish

Find in document · synch publish button to accounts at **rpubs.com**, **shinyapps.io**, RStudio Connect · Reload document

R Markdown
RStudio
• R Markdown

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents.

```
summary(cars)
```

```
##     speed           dist
##  Min.   : 4.0   Min.   :  2.00
##  1st Qu.:12.0   1st Qu.: 26.00
##  Median :15.0   Median : 36.00
##  Mean   :15.4   Mean   : 42.98
##  3rd Qu.:19.0   3rd Qu.: 56.00
##  Max.   :25.0   Max.   :120.00
```

For more details on using R Markdown see http://rmarkdown.rstudio.com.

Files · Plots · Packages · Help · Viewer
New Folder · Delete · Rename · More
Home · Desktop · R-Markdown-Cheatsheet

report.Rmd	398 B	Feb 26, 2016, 3:36 PM
report.html ❼	581.3 KB	Feb 26, 2016, 3:36 PM

.rmd Structure

YAML Header
Optional section of render (e.g. pandoc) options written as key:value pairs (YAML).

At start of file

Between lines of - - -

Text
Narration formatted with markdown, mixed with:

Code Chunks
Chunks of embedded code. Each chunk:

Begins with ```` ```{r} ````

ends with ```` ``` ````

R Markdown will run the code and append the results to the doc. It will use the location of the .Rmd file as the **working directory**

Parameters

Parameterize your documents to reuse with different inputs (e.g., data, values, etc.)

1. **Add parameters** · Create and set parameters in the header as sub-values of params

2. **Call parameters** · Call parameter values in code as params$<name>

3. **Set parameters** · Set values wth Knit with parameters or the params argument of render():

render("doc.Rmd", params = list(n = 1, d = as.Date("2015-01-01"))

```
---
params:
  n: 100
  d: !r Sys.Date()
---
```

Today's date is `` `r params$d` ``

Knit with Parameters...
- Knit to HTML
- Knit to PDF
- Knit to Word
- Knit with Parameters...

Interactive Documents

Turn your report into an interactive Shiny document in 4 steps

1. Add runtime: shiny to the YAML header.
2. Call Shiny input functions to embed input objects.
3. Call Shiny render functions to embed reactive output.
4. Render with rmarkdown::run or click Run Document in RStudio IDE

```
---
output: html_document
runtime: shiny
---

```{r, echo = FALSE}
numericInput("n",
  "How many cars?", 5)

renderTable({
  head(cars, input$n)
})```
```

How many cars? 5

	speed	dist
1	4.00	2.00
2	4.00	10.00
3	7.00	4.00
4	7.00	22.00
5	8.00	16.00

Embed a complete app into your document with shiny::**shinyAppDir()**

NOTE: *Your report will rendered as a Shiny app, which means you must choose an html output format, like* **html_document**, *and serve it with an active R Session.*

render

Use rmarkdown::**render()** to render/knit at cmd line. Important args:

input - file to render **output_format** -	**output_options** - List of render options (as in YAML)	**output_file** **output_dir**	**params** - list of params to use	**envir** - environment to evaluate code chunks in	**encoding** - of input file

Embed code with knitr syntax

INLINE CODE
Insert with `` `r <code>` ``. Results appear as text without code.

Built with `` `r getRversion()` `` ➤ Built with 3.2.3

CODE CHUNKS
One or more lines surrounded with ```` ```{r} ```` and ```` ``` ````. Place chunk options within curly braces, after **r**. Insert with

```
```{r echo=TRUE}
getRversion()
```
```

```
getRversion()
```
```
[1] '3.2.3'
```

### GLOBAL OPTIONS
Set with knitr::**opts_chunk$set()**, e.g.

```
```{r include=FALSE}
knitr::opts_chunk$set(echo = TRUE)
```
```

## IMPORTANT CHUNK OPTIONS

**cache** - cache results for future knits (default = FALSE)

**cache.path** - directory to save cached results in (default = "cache/")

**child** - file(s) to knit and then include (default = NULL)

**collapse** - collapse all output into single block (default = FALSE)

**comment** - prefix for each line of results (default = '##')

**dependson** - chunk dependencies for caching (default = NULL)

**echo** - Display code in output document (default = TRUE)

**engine** - code language used in chunk (default = 'R')

**error** - Display error messages in doc (TRUE) or stop render when errors occur (FALSE) (default = FALSE)

**eval** - Run code in chunk (default = TRUE)

**fig.align** - 'left', 'right', or 'center' (default = 'default')

**fig.cap** - figure caption as character string (default = NULL)

**fig.height, fig.width** - Dimensions of plots in inches

**highlight** - highlight source code (default = TRUE)

**include** - Include chunk in doc after running (default = TRUE)

**message** - display code messages in document (default = TRUE)

**results** (default = 'markup')
'asis' - passthrough results
'hide' - do not display results
'hold' - put all results below all code

**tidy** - tidy code for display (default = FALSE)

**warning** - display code warnings in document (default = TRUE)

Options not listed above: R.options, aniopts, autodep, background, cache.comments, cache.lazy, cache.rebuild, cache.vars, dev, dev.args, dpi, engine.opts, engine.path, fig.asp, fig.env, fig.ext, fig.keep, fig.lp, fig.path, fig.pos, fig.process, fig.retina, fig.scap, fig.show, fig.showtext, fig.subcap, interval, out.extra, out.height, out.width, prompt, purl, ref.label, render, size, split, tidy.opts

# Pandoc's Markdown

Write with syntax on the left to create effect on right (after render)

Plain text
End a line with two spaces
to start a new paragraph.
*italics* and **bold**
`verbatim code`
sub/superscript^2^~2~
~~strikethrough~~
escaped: \* \_ \\
endash: --, emdash: ---
equation: $A = \pi*r^{2}$
equation block:

$$E = mc^{2}$$

> block quote

# Header1 {#anchor}

## Header 2 {#css_id}

### Header 3 {.css_class}

#### Header 4

##### Header 5

###### Header 6

<!--Text comment-->

\textbf{Tex ignored in HTML}
<em>HTML ignored in pdfs</em>

<http://www.rstudio.com>
[link](www.rstudio.com)
Jump to [Header 1](#anchor)
image:

![Caption](smallorb.png)

* unordered list
    + sub-item 1
    + sub-item 2
        - sub-sub-item 1

* item 2

    Continued (indent 4 spaces)

1. ordered list
2. item 2
    i) sub-item 1
        A.  sub-sub-item 1

(@) A list whose numbering

continues after

(@) an interruption

Term 1

:   Definition 1

| Right | Left | Default | Center |
|------:|:-----|---------|:------:|
| 12 | 12 | 12 | 12 |
| 123 | 123 | 123 | 123 |
| 1 | 1 | 1 | 1 |

- slide bullet 1
- slide bullet 2

(>- to have bullets appear on click)

horizontal rule/slide break:

***

A footnote [^1]

[^1]: Here is the footnote.

---

**Rendered output (right column):**

Plain text
End a line with two spaces
to start a new paragraph.
*italics* and **bold**
`verbatim code`
sub/superscript²₂
strikethrough
escaped: * _ \
endash: –, emdash: —
equation: $A = \pi*r^2$
equation block:

$$E = mc^2$$

block quote

# Header1

## Header 2

### Header 3

#### Header 4

##### Header 5

###### Header 6

HTML ignored in pdfs

http://www.rstudio.com
link
Jump to Header 1
image:

Caption

* unordered list
    ◦ sub-item 1
    ◦ sub-item 2
        ▪ sub-sub-item 1
* item 2

    Continued (indent 4 spaces)

1. ordered list
2. item 2
    i. sub-item 1
        A. sub-sub-item 1

1. A list whose numbering

continues after

2. an interruption

Term 1

Definition 1

| Right | Left | Default | Center |
|-------|------|---------|--------|
| 12 | 12 | 12 | 12 |
| 123 | 123 | 123 | 123 |
| 1 | 1 | 1 | 1 |

* slide bullet 1
* slide bullet 2

(>- to have bullets appear on click)

horizontal rule/slide break:

A footnote [1]

1. Here is the footnote. ↵

# Set render options with YAML

When you render, R Markdown

1. runs the R code, embeds results and text into .md file with knitr
2. then converts the .md file into the finished format with pandoc

Rmd → knitr → md → pandoc → [documents]

Set a document's default output format in the YAML header:

```
---
output: html_document
---
# Body
```

output value	creates
html_document	html
pdf_document	pdf (requires Tex)
word_document	Microsoft Word (.docx)
odt_document	OpenDocument Text
rtf_document	Rich Text Format
md_document	Markdown
github_document	Github compatible markdown
ioslides_presentation	ioslides HTML slides
slidy_presentation	slidy HTML slides
beamer_presentation	Beamer pdf slides (requires Tex)

Customize output with sub-options (listed to the right):

Indent 2 spaces **Indent 4 spaces**

```
---
output: html_document:
    code_folding: hide
    toc_float: TRUE
---
# Body
```

html tabsets
Use tablet css class to place sub-headers into tabs

```
# Tabset {.tabset .tabset-fade .tabset-pills}
## Tab 1
text 1
## Tab 2
text 2
### End tabset
```

Tabset

Tab 1 | Tab 2

text 1
End tabset

sub-option	description	html	pdf	word	odt	rtf	md	github	ioslides	slidy	beamer
citation_package	The LaTeX package to process citations, natbib, biblatex or none		X				X				X
code_folding	Let readers to toggle the display of R code, "none", "hide", or "show"	X									
colortheme	Beamer color theme to use										X
css	CSS file to use to style document	X							X	X	
dev	Graphics device to use for figure output (e.g. "png")	X	X				X	X	X	X	X
duration	Add a countdown timer (in minutes) to footer of slides										X
fig_caption	Should figures be rendered with captions?	X	X	X	X				X		X
fig_height, fig_width	Default figure height and width (in inches) for document	X	X	X	X	X	X	X	X	X	X
highlight	Syntax highlighting: "tango", "pygments", "kate","zenburn", "textmate"	X	X	X							X
includes	File of content to place in document (in_header, before_body, after_body)	X	X						X	X	X
incremental	Should bullets appear one at a time (on presenter mouse clicks)?								X	X	X
keep_md	Save a copy of .md file that contains knitr output	X		X	X	X					X
keep_tex	Save a copy of .tex file that contains knitr output		X								X
latex_engine	Engine to render latex, "pdflatex", "xelatex", or "lualatex"		X								X
lib_dir	Directory of dependency files to use (Bootstrap, MathJax, etc.)	X							X	X	
mathjax	Set to local or a URL to use a local/URL version of MathJax to render equations	X							X	X	
md_extensions	Markdown extensions to add to default definition or R Markdown	X	X	X	X	X	X	X	X	X	X
number_sections	Add section numbering to headers	X	X								
pandoc_args	Additional arguments to pass to Pandoc	X	X	X	X	X	X	X	X	X	X
preserve_yaml	Preserve YAML front matter in final document?						X				
reference_docx	docx file whose styles should be copied when producing docx output			X							
self_contained	Embed dependencies into the doc	X							X	X	
slide_level	The lowest heading level that defines individual slides										X
smaller	Use the smaller font size in the presentation?								X		
smart	Convert straight quotes to curly, dashes to em-dashes, … to ellipses, etc.	X							X	X	
template	Pandoc template to use when rendering file quarterly_report.html).	X	X		X					X	X
theme	Bootswatch or Beamer theme to use for page	X									X
toc	Add a table of contents at start of document	X	X	X			X	X	X		X
toc_depth	The lowest level of headings to add to table of contents	X	X	X			X	X	X		
toc_float	Float the table of contents to the left of the main content	X									

Create a Reusable Template

1. **Create a new package** with a inst/rmarkdown/templates directory

2. In the directory, **Place a folder** that contains:
template.yaml (see below)
skeleton.Rmd (contents of the template)
any supporting files

3. **Install the package**

4. **Access template** in wizard at File ▶ New File ▶ R Markdown

template.yaml

```
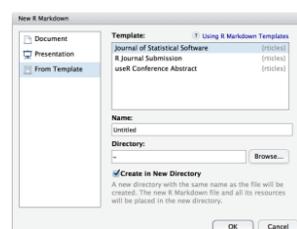---
name: My Template
---
```

Table Suggestions

Several functions format R data into tables

Table with kable	
eruptions	waiting
3.600	79
1.800	54
3.333	74
2.283	62

	eruptions	waiting
1	3.60	79.00
2	1.80	54.00
3	3.33	74.00
4	2.28	62.00

Table with xtable

Table with stargazer		
	eruptions	waiting
1	3.600	79
2	1.800	54
3	3.333	74
4	2.283	62

data <- faithful[1:4,]

```{r results = 'asis'}
knitr::kable(data, caption = "Table with kable")
```

```{r results = "asis"}
print(xtable::xtable(data, caption = "Table with xtable"),
    type = "html", html.table.attributes = "border=0"))
```

```{r results = "asis"}
stargazer::stargazer(data, type = "html", title = "Table
    with stargazer")
```

Learn more in the **stargazer**, **xtable**, and **knitr** packages.

Citations and Bibliographies

Create citations with .bib, .bibtex, .copac, .enl, .json, .medline, .mods, .ris, .wos, and .xml files

1. **Set bibliography file** and CSL 1.0 Style file (optional) in the YAML header
2. **Use citation keys in text**

```
---
bibliography: refs.bib
csl: style.csl
---
```

Smith cited [@smith04].
Smith cited without author [-@smith04].
@smith04 cited in line.

3. **Render.** Bibliography will be added to end of document

Smith cited (Joe Smith 2004).
Smith cited without author (2004).
Joe Smith (2004) cited in line.