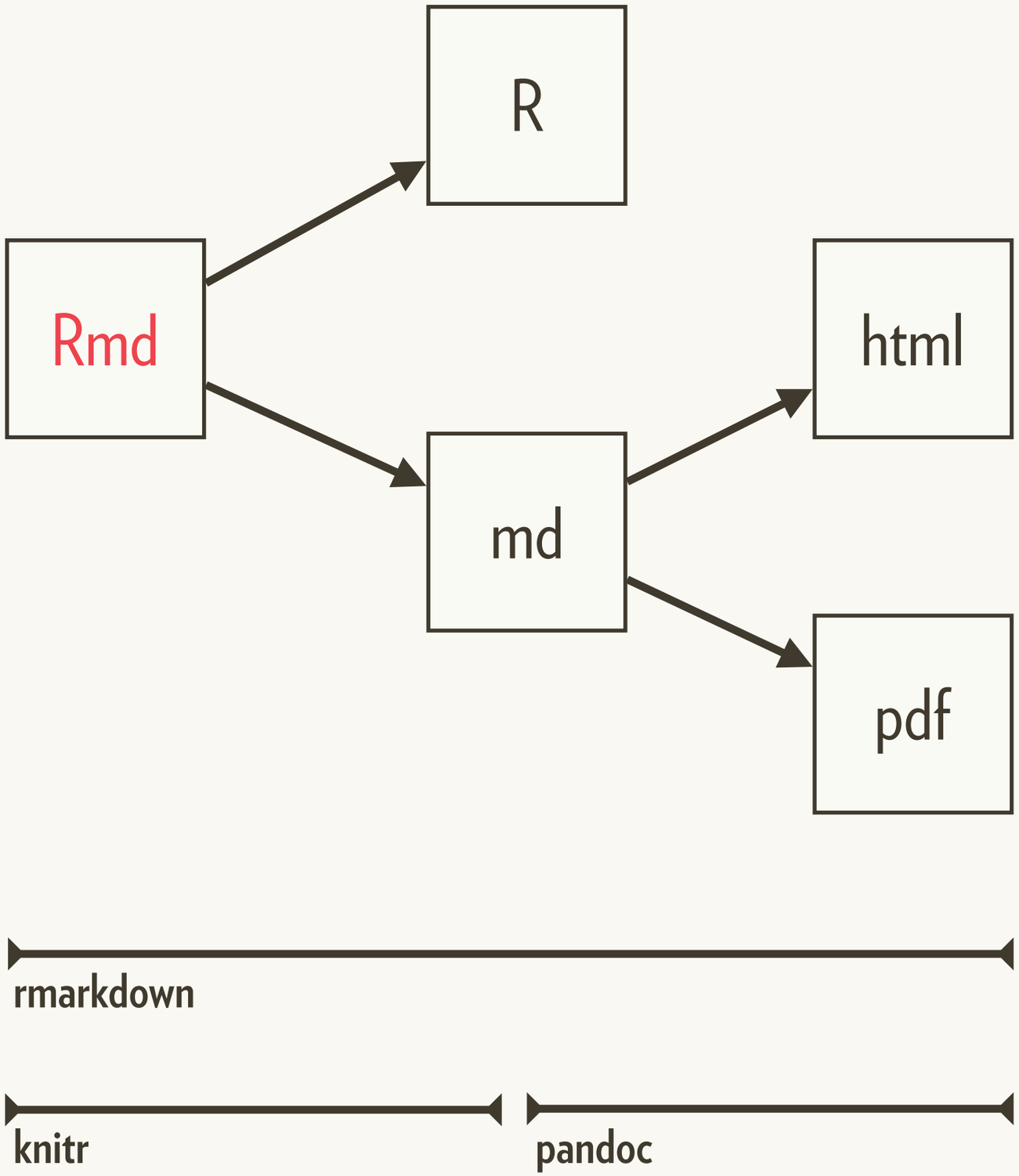
vignettes/

January 2017

Hadley Wickham
@hadleywickham
Chief Scientist, **RStudio**

Vignettes



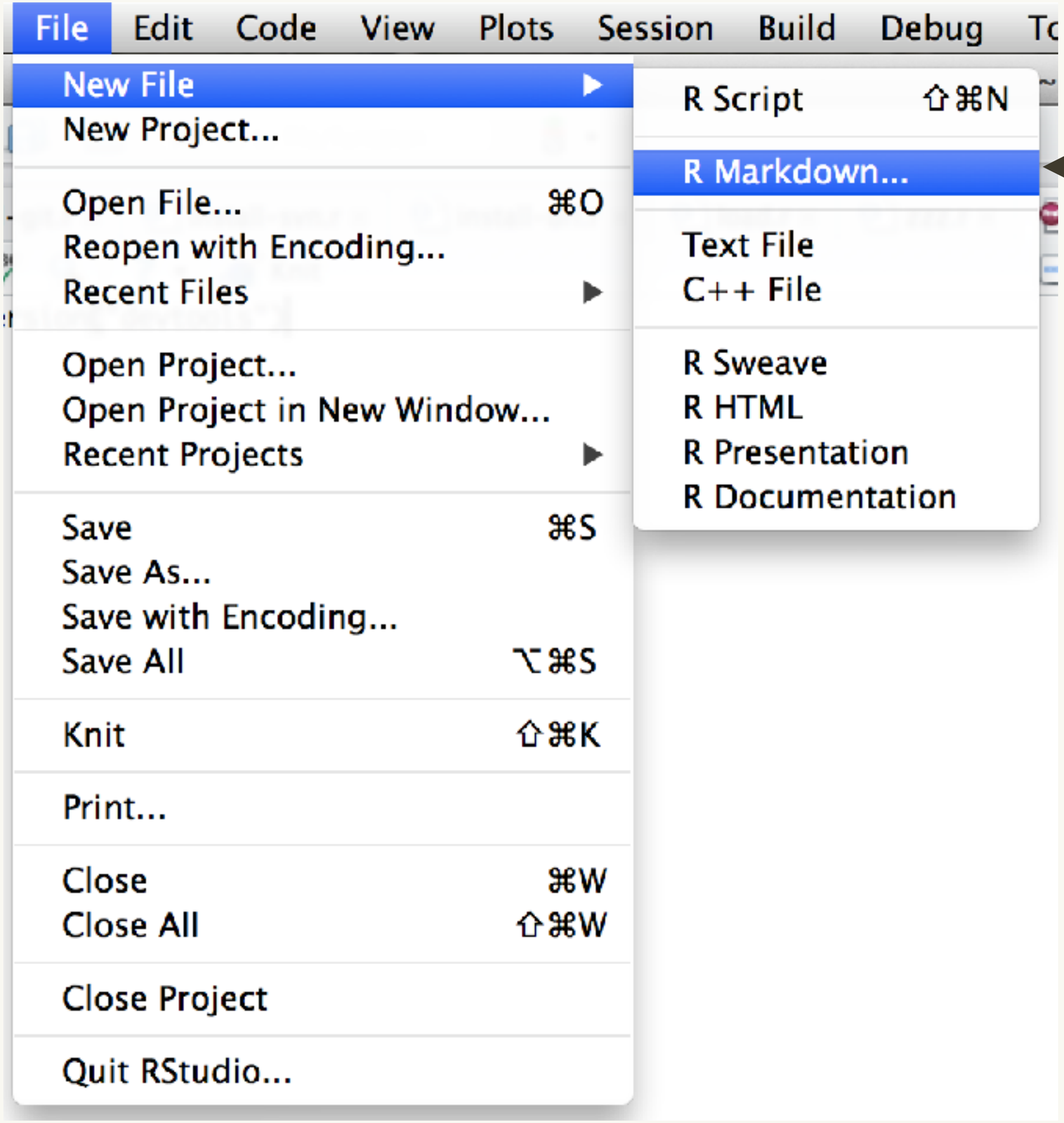
Lets you combine prose and code to explain your how you package works.

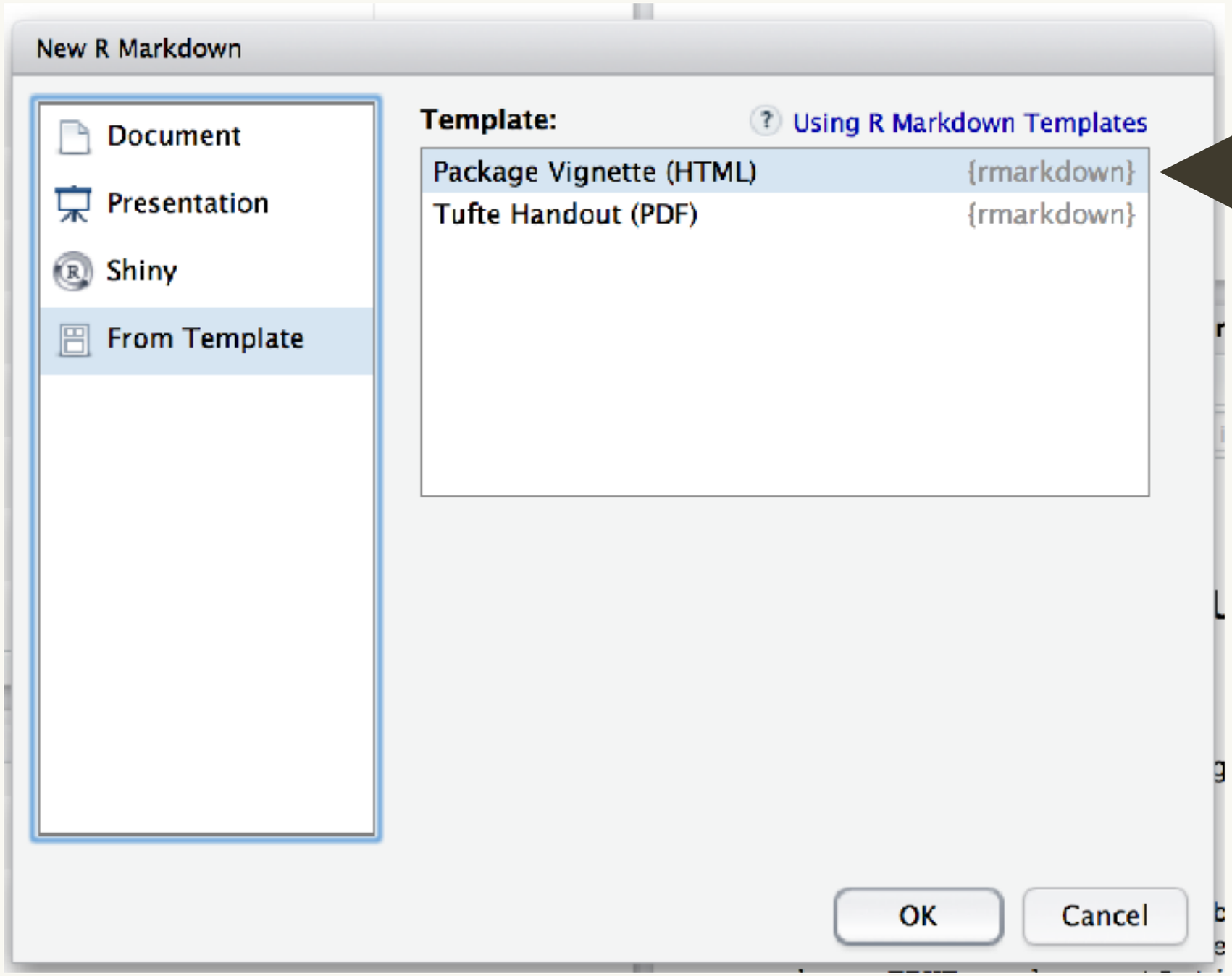
The hard part is the writing, not the technology!

http://r-pkgs.had.co.nz/vignettes.html

Easiest way to get started is with use_vignette()

```
devtools::use_vignette("name")

# Adds to DESCRIPTION
Suggests: knitr
VignetteBuilder: knitr

# Creates vignettes/
# Drafts vignettes/name.Rmd
```


File
Edit
Code
View
Plots
Session
Build
Debug
New File
New Project...
Open File... ⌘O
Reopen with Encoding...
Recent Files
Open Project...
Open Project in New Window...
Recent Projects
Save ⌘S
Save As...
Save with Encoding...
Save All ⌥⌘S
Knit ⇧⌘K
Print...
Close ⌘W
Close All ⇧⌘W
Close Project
Quit RStudio...
R Script ⇧⌘N
R Markdown...
Text File
C++ File
R Sweave
R HTML
R Presentation
R Documentation


New R Markdown
Document
Presentation
Shiny
From Template
Template:
Using R Markdown Templates
Package Vignette (HTML) {rmarkdown}
Tufte Handout (PDF) {rmarkdown}
OK
Cancel

Vignette = Rmarkdown + special metadata

```
---
title: "Vignette Title"
author: "Vignette author"
date: "`r Sys.Date()`"
output: rmarkdown::html_vignette
vignette: >
  %\VignetteIndexEntry{Vignette Title}
  %\VignetteEngine{knitr::rmarkdown}
  \usepackage[utf8]{inputenc}
---

Vignettes are long form documentation commonly included in packages.
Because they are part of the distribution of the package, they need to
be as compact as possible. The `html_vignette` output type provides a
custom style sheet (and tweaks some options) to ensure that the
resulting html is as small as possible. The `html_vignette` format:


...
```

Special output format for vignettes

Special metadata needed by R

Markdown

“Markdown is a text-to-HTML conversion tool for web writers. Markdown allows you to write using an easy-to-read, easy-to-write plain text format, then convert it to structurally valid XHTML (or HTML).”

“The overriding design goal for Markdown’s formatting syntax is to make it as readable as possible. The idea is that a Markdown-formatted document should be publishable as-is, as plain text, without looking like it’s been marked up with tags or formatting instructions.”

—*John Gruber*

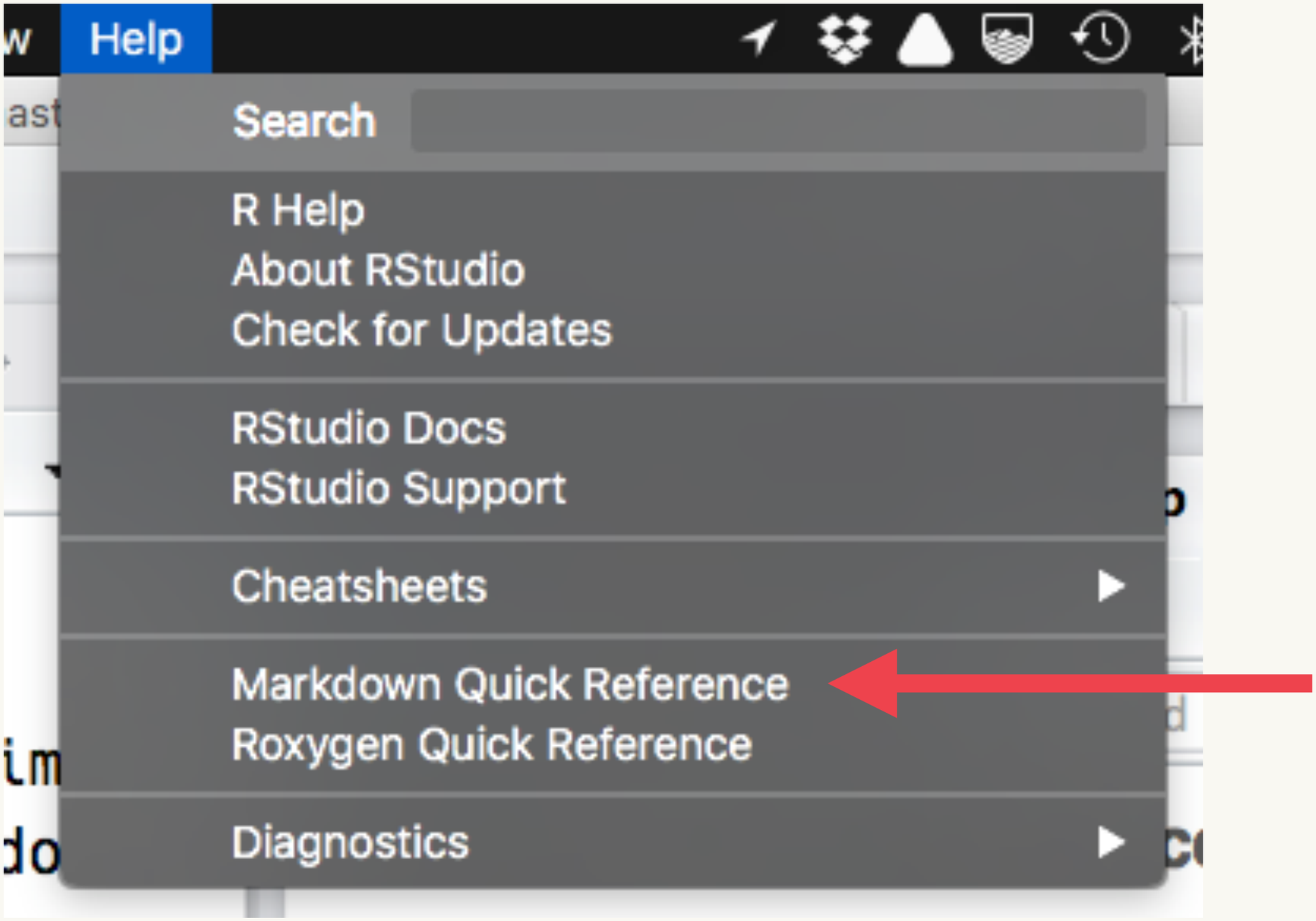
````
# This is a top level heading

This is some text. Make text _italic_ with single  underscores (or stars). Make it **bold** with double stars (or underscores). Here is a [link to a markdown guide](http://bit.ly/19fAexE).

* This is a list
* This is another item

```R
# Some R code
f <- function() x + 1
```

## This is a secondary heading

You can also do `inline code`, numbered lists and quotes and more.
````

Your turn

Read http://r-pkgs.had.co.nz/vignettes.html

How do you create a numbered list?

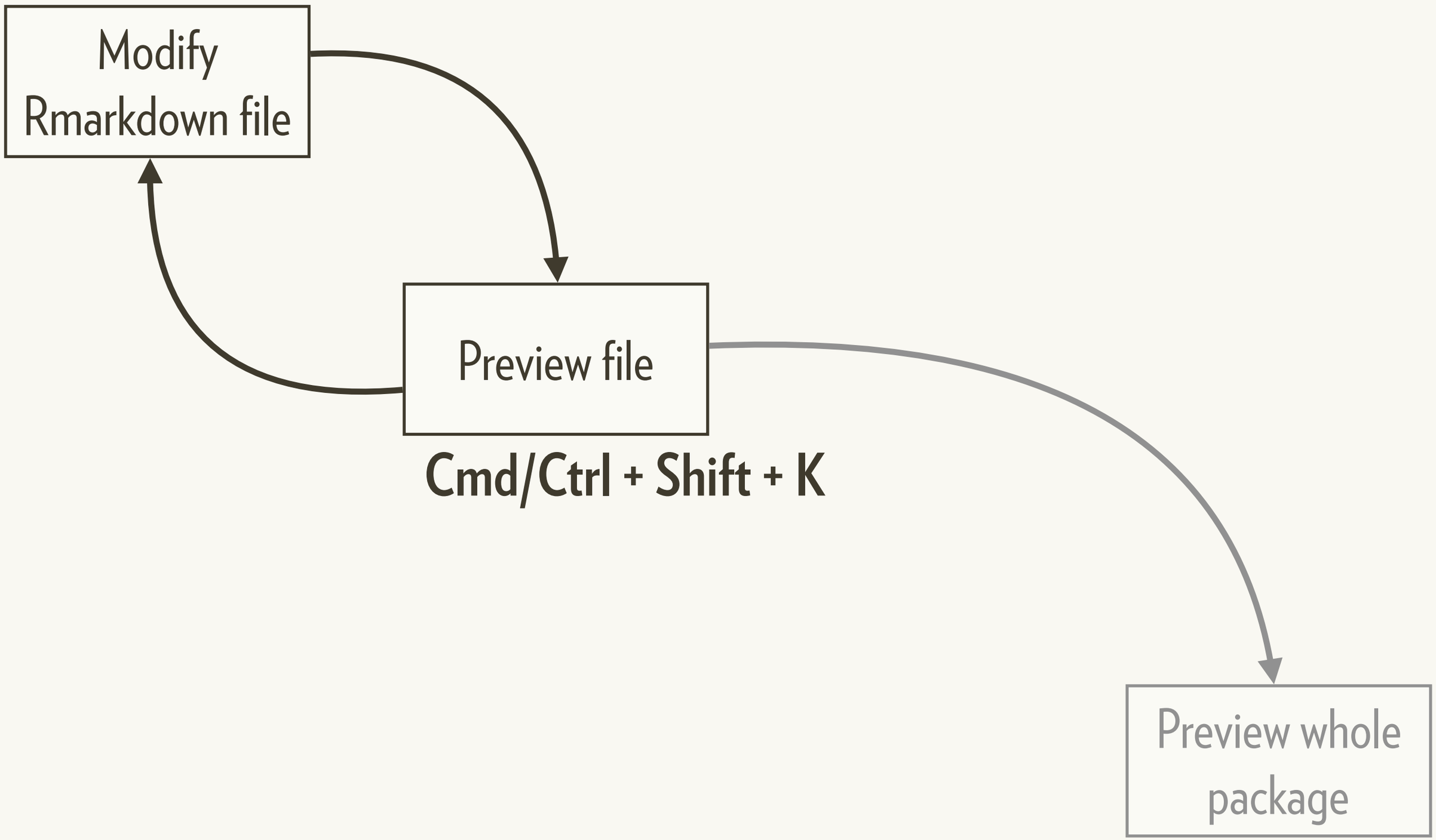
How do you add a horizontal rule?


Help
Search
R Help
About RStudio
Check for Updates
RStudio Docs
RStudio Support
Cheatsheets
Markdown Quick Reference
Roxygen Quick Reference
Diagnostics

Knitr

````
The main difference with R markdown compared to markdown is that
instead of putting R code in blocks like:

```r
# Some R code
f <- function() x + 1
```

you put it in blocks like

```{r}
# Some R code
f <- function() x + 1
```

and the code is run and the output included in a generated .md file.

More documentation at http://yihui.name/knitr/
````

Documentation workflow




```
devtools::install(build_vignettes = TRUE)
browseVignettes()
```

Change working directory/project to:

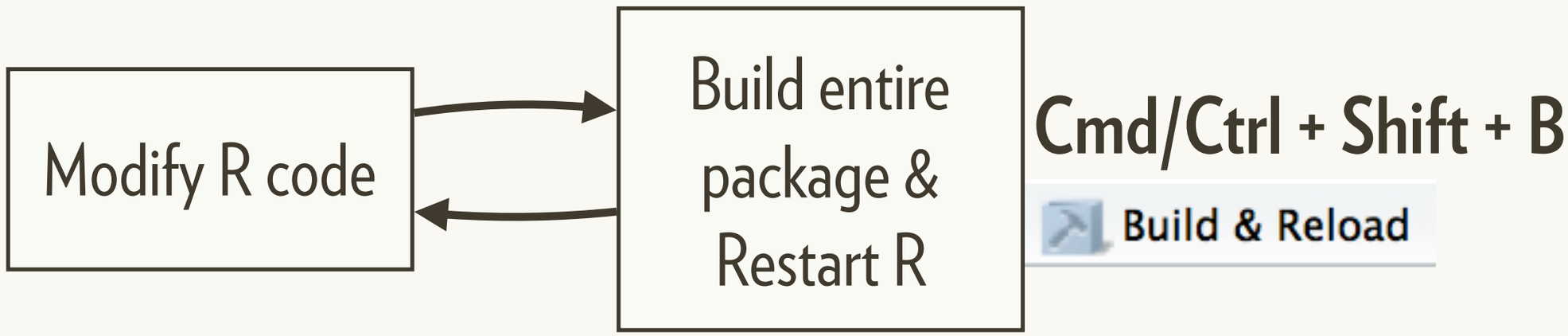
[document-me]

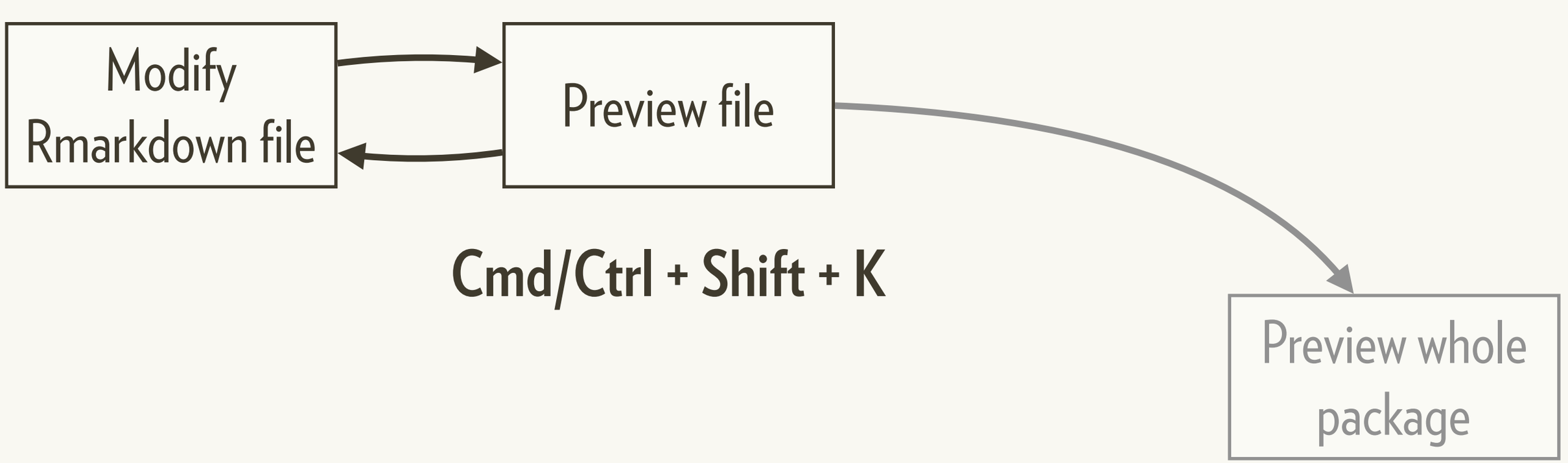
Your turn

Open `vignettes/clt.Rmd`. Knit it.

Why does it fail? Fix the error.

Vignette workflow





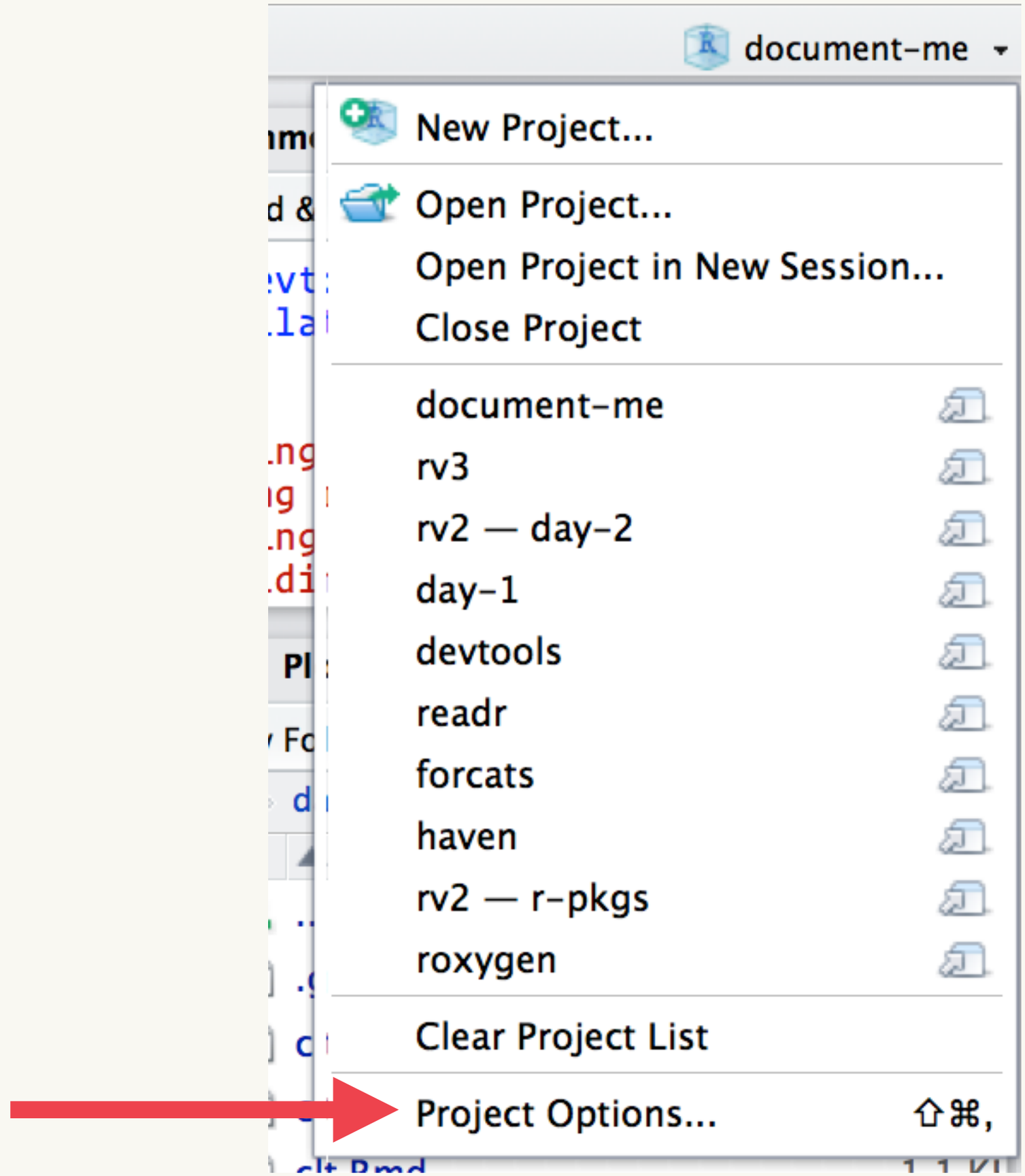

```
devtools::install(build_vignettes = TRUE)
browseVignettes()
```

```{r, include = FALSE}
# include = FALSE suppresses the output of this block
library(rv2)
```

Other useful options:

* message = FALSE: suppresses messages
* error = TRUE: if you want to show an error
* eval = FALSE: show the code, don't run it

See more at http://r-pkgs.had.co.nz/vignettes.html#knitr
Complete list at http://yihui.name/knitr/options


document-me
New Project...
Open Project...
Open Project in New Session...
Close Project
document-me
rv3
rv2 — day-2
day-1
devtools
readr
forcats
haven
rv2 — r-pkgs
roxygen
Clear Project List
Project Options...
⇧⌘,

Project Options

General

Code Editing

Sweave

Build Tools

Git/SVN

Packrat

Project build tools: Package

Package directory:

(Project Root) Browse...

Use devtools package functions if available

Generate documentation with Roxygen Configure...

Build and Reload — R CMD INSTALL additional options:

--no-multiarch --with-keep.source

Check Package — R CMD check additional options:

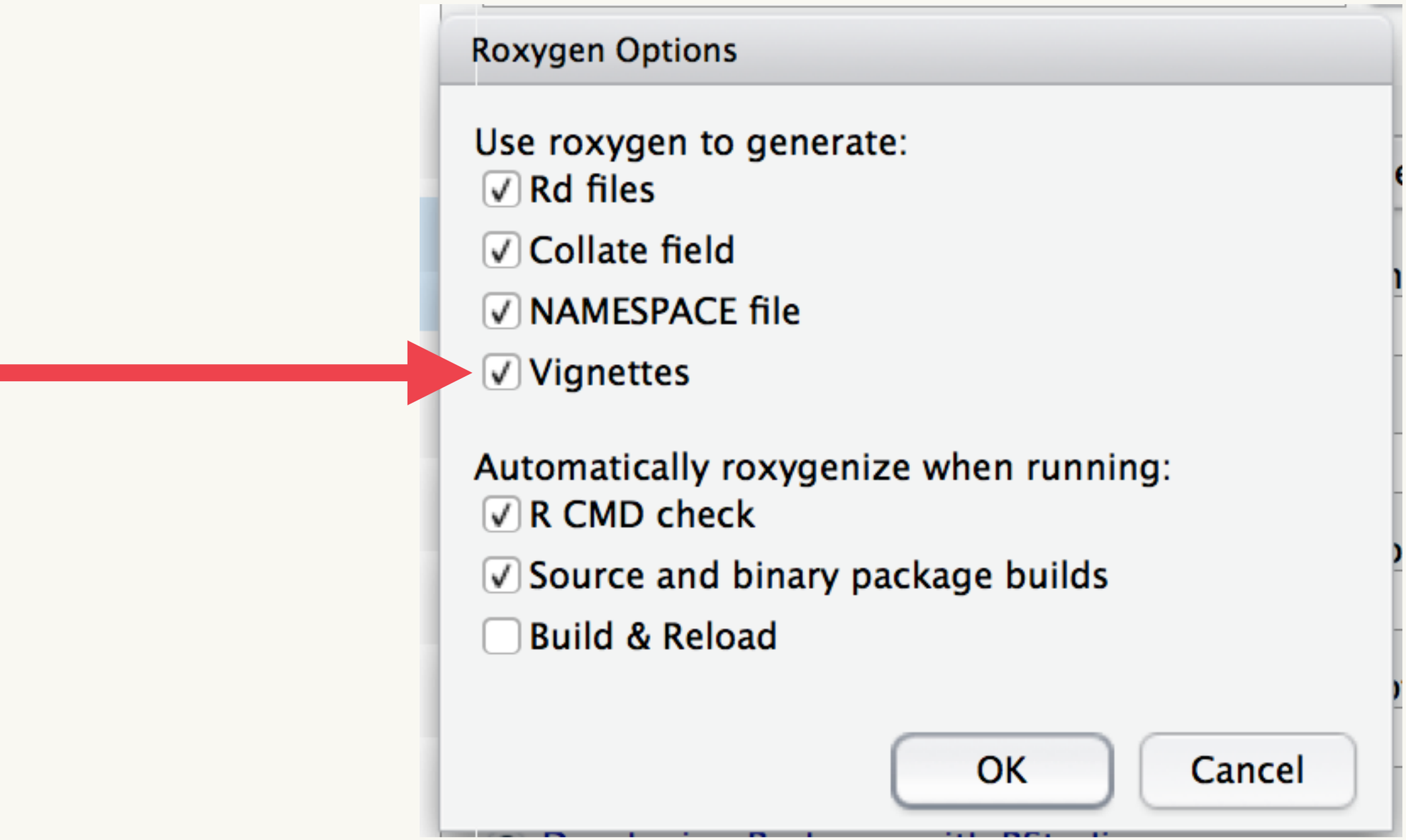
Build Source Package — R CMD build additional options:

Build Binary Package — R CMD INSTALL additional options:

Developing Packages with RStudio

OK Cancel

Then vignettes are built by document()



This work is licensed under the
Creative Commons Attribution-Noncommercial 3.0
United States License.

To view a copy of this license, visit
http://creativecommons.org/licenses/by-nc/3.0/us/