

Programming and collaborations: roles and tools when using R



Isabella Gollini

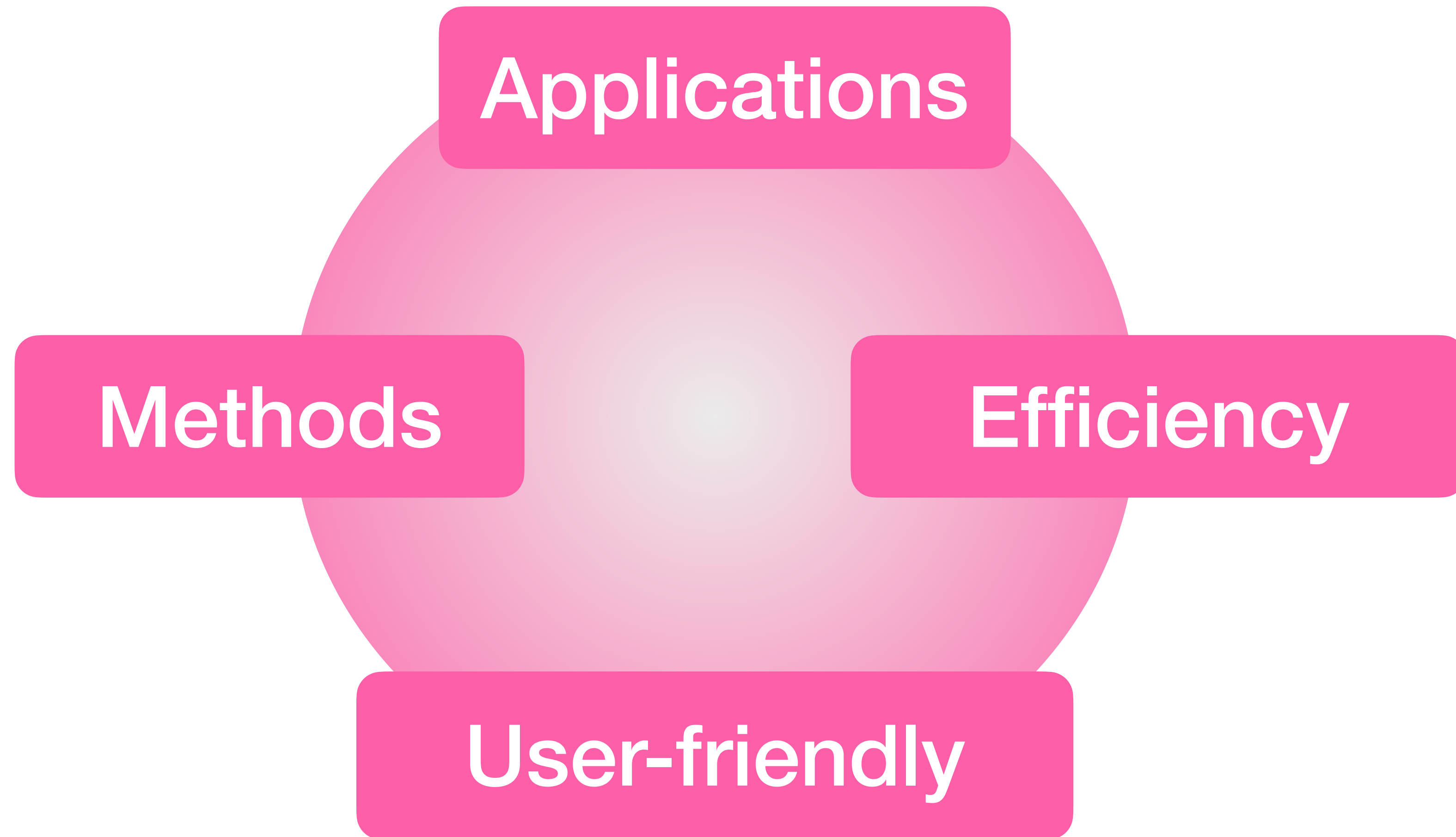
[@IsabellaGollini](https://twitter.com/IsabellaGollini)

isabella.gollini@ucd.ie

uRosConf

www.urosconf.org

Programming & Collaborating



How to combine them?



Use different perspectives!

Different perspectives

Developer

Teacher

Learner

Collaborator

Clear programming style

- Your programming style should be accessible to allow users to understand it and potential contribute by extending improving your code

The tidyverse style guide

Table of contents

[Welcome](#)

Analyses

1 **Files**

2 [Syntax](#)

3 [Functions](#)

4 [Pipes](#)

5 [ggplot2](#)

Packages

6 [Files](#)

7 [Documentation](#)

1 Files

1.1 Names

File names should be meaningful and end in `.R`. Avoid using special characters in file names - stick with numbers, letters, `-`, and `_`.

```
# Good
fit_models.R
utility_functions.R

# Bad
fit models.R
foo.r
stuff.r
```

style.tidyverse.org

Package dependences

- Be careful with package dependences! They may stop to maintain it, or make major changes!
- Always make a note of which functions you are calling from that package (if not too many!)
- Call external functions with `package::function()`
- Check if you can remove them! For example move from the `magrittr` pipe operator `%>%` to the native pipe `|>`

Commenting


- Your code is under continual development and can be a good starting point for the implementation of more advanced methodologies.
- You may need to share your code. Keep it full of comments.
- If you are using R, write at the top of each functions comments that can be converted with `roxygen2` into function documentation!

roxygen2.r-lib.org/



You write specially formatted comments in .R

```
#' Add a Column to a Data Frame
#'  
#' Allows you to specify the position. Will replace existing variable  
#' with the same name if present.  
#'  
#' @param x A data frame  
#' @param name Name of variable to create. If a variable of that name  
#' already exists it will be replaced  
#' @param value Values to insert.  
#' @param where Position to insert. Use 1 to insert on LHS, or -1 to insert on  
#' RHS.  
#' @examples  
#' df <- data.frame(x = 1:5)  
#' add_col(df, "y", runif(5))  
#' add_col(df, "y", runif(5), where = 1)  
#'  
#' add_col(df, "x", 5:1)
```

The Roxymart logo is located in the top right corner. It consists of a blue hexagon containing a white circle with a blue 'R' inside. Below the hexagon, the word 'roxyge' is written in a light blue, lowercase font.

You write specially formatted comments in .R



Roxygen comment

```
#' Roxygen comment
#'
```

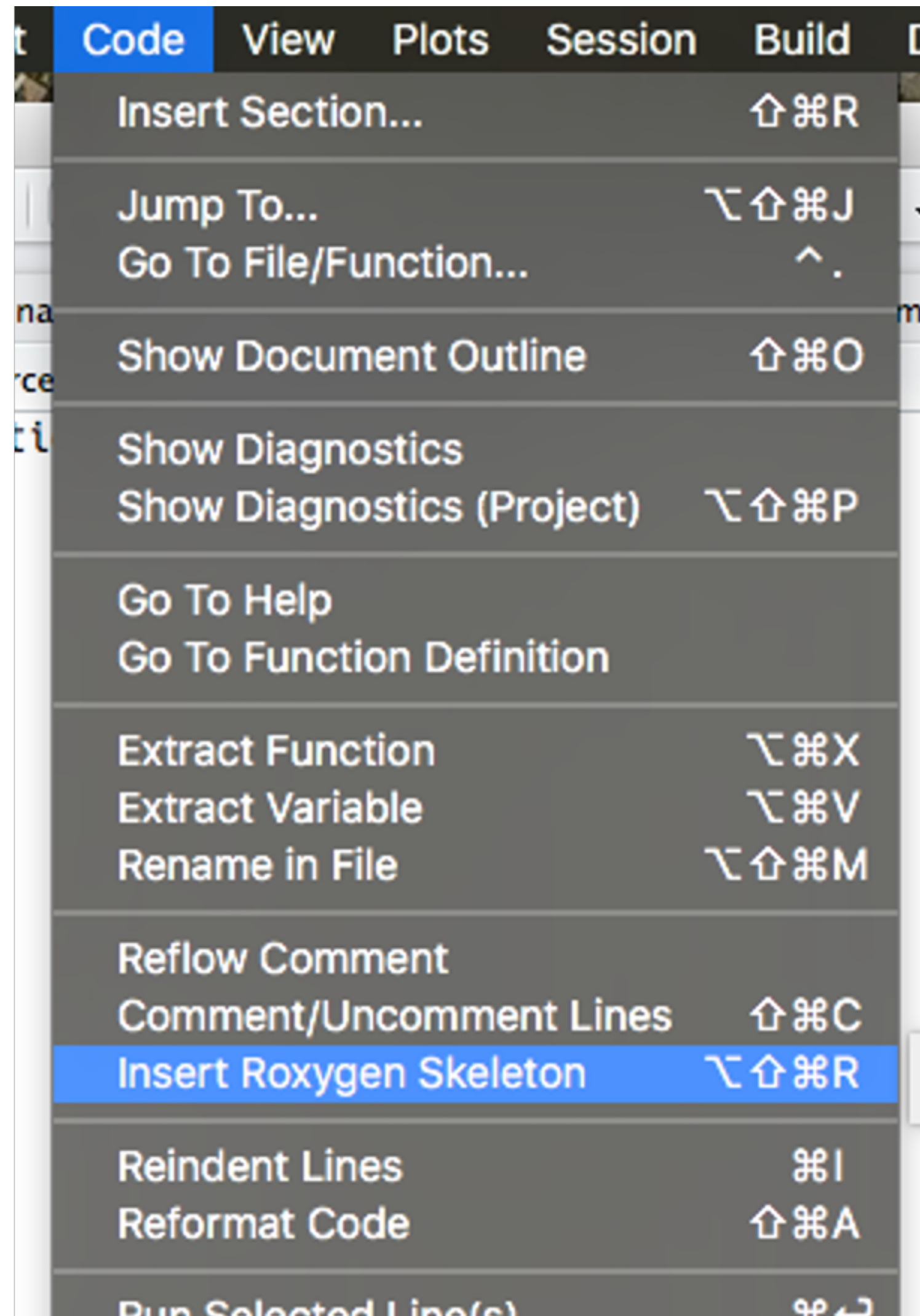
Allows you to specify the position. Will replace existing variable with the same name if present.

```
#'
```

Roxygen tag

```
#' @param create. If a variable of that name
#' already exists it will be replaced
#' @param value Values to insert.
#' @param where Position to insert. Use 1 to insert on LHS, or -1 to insert on
#' RHS.
#' @examples
#' df <- data.frame(x = 1:5)
#' add_col(df, "y", runif(5))
#' add_col(df, "y", runif(5), where = 1)
#'
```

RStudio helps you remember

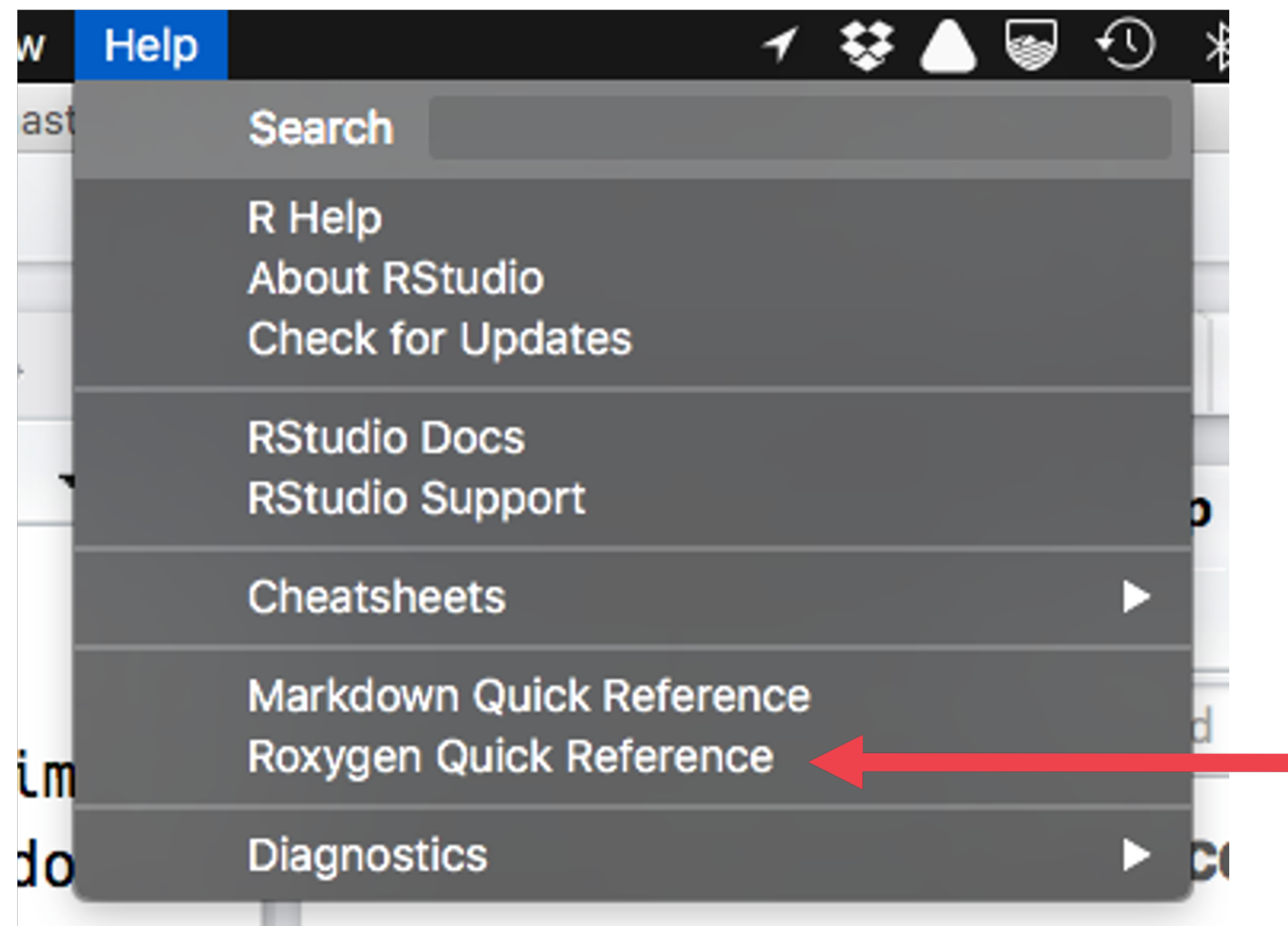
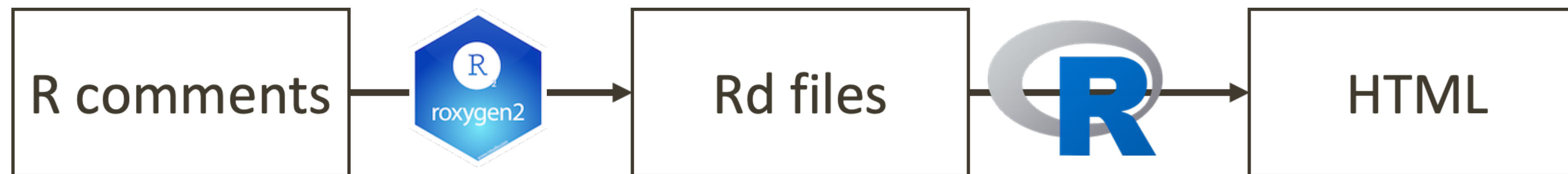


```
#' Title
#'\n
#' @param x
#' @param y
#' @param z
#'\n
#' @return
#' @export
#'\n
#' @examples
fun <- function(x, y, z) {

}
```


R packages

- `roxygen2` works for documenting R packages.



roxygen2.r-lib.org/

roxygen2 translates to .Rd

```
% Generated by roxygen2: do not edit by hand
% Please edit documentation in R/add_col.R
\name{add_col}
\alias{add_col}
\title{Add a Column to a Data Frame}
\usage{
add_col(x, name, value, where = -1)
}
\arguments{
\item{x}{A data frame}

\item{name}{Name of variable to create. If a variable of that name
already exists it will be replaced}

\item{value}{Values to insert.}

\item{where}{Position to insert. Use 1 to insert on LHS, or -1 to
insert on
RHS.}
}
\description{
Allows you to specify the position. Will replace existing variable
with the same name if present.
}
```

**Don't modify the .Rd
file if you used
roxygen2**

Add a Column to a Data Frame

Description

Allows you to specify the position. Will replace existing variable with the same name if present.

Usage

```
add_col(x, name, value, where = -1)
```

Arguments

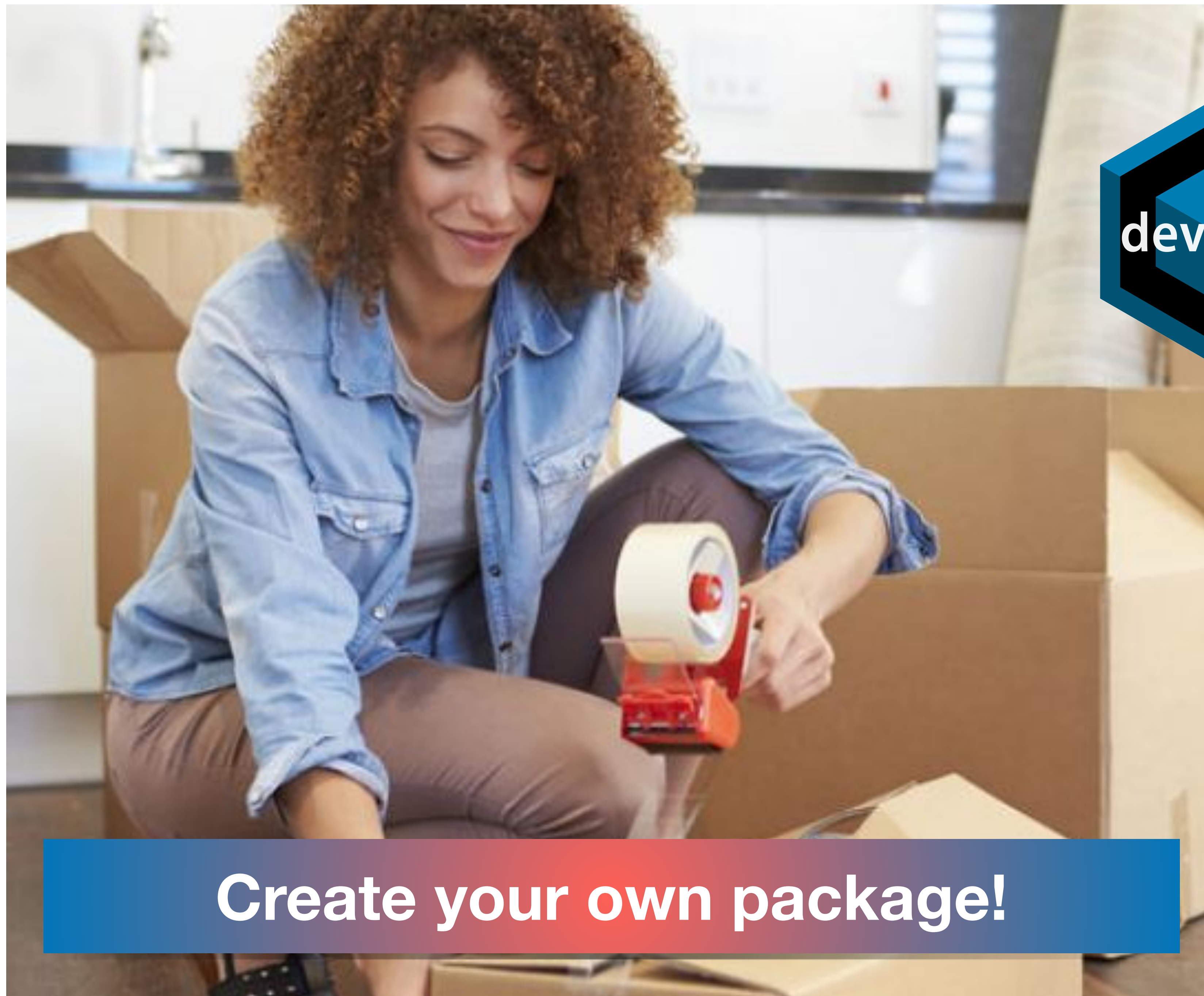
- x** A data frame
- name** Name of variable to create. If a variable of that name already exists it will be replaced
- value** Values to insert.
- where** Position to insert. Use 1 to insert on LHS, or -1 to insert on RHS.

Examples

```
df <- data.frame(x = 1:5)
add_col(df, "y", runif(5))
add_col(df, "y", runif(5), where = 1)

add_col(df, "x", 5:1)
```

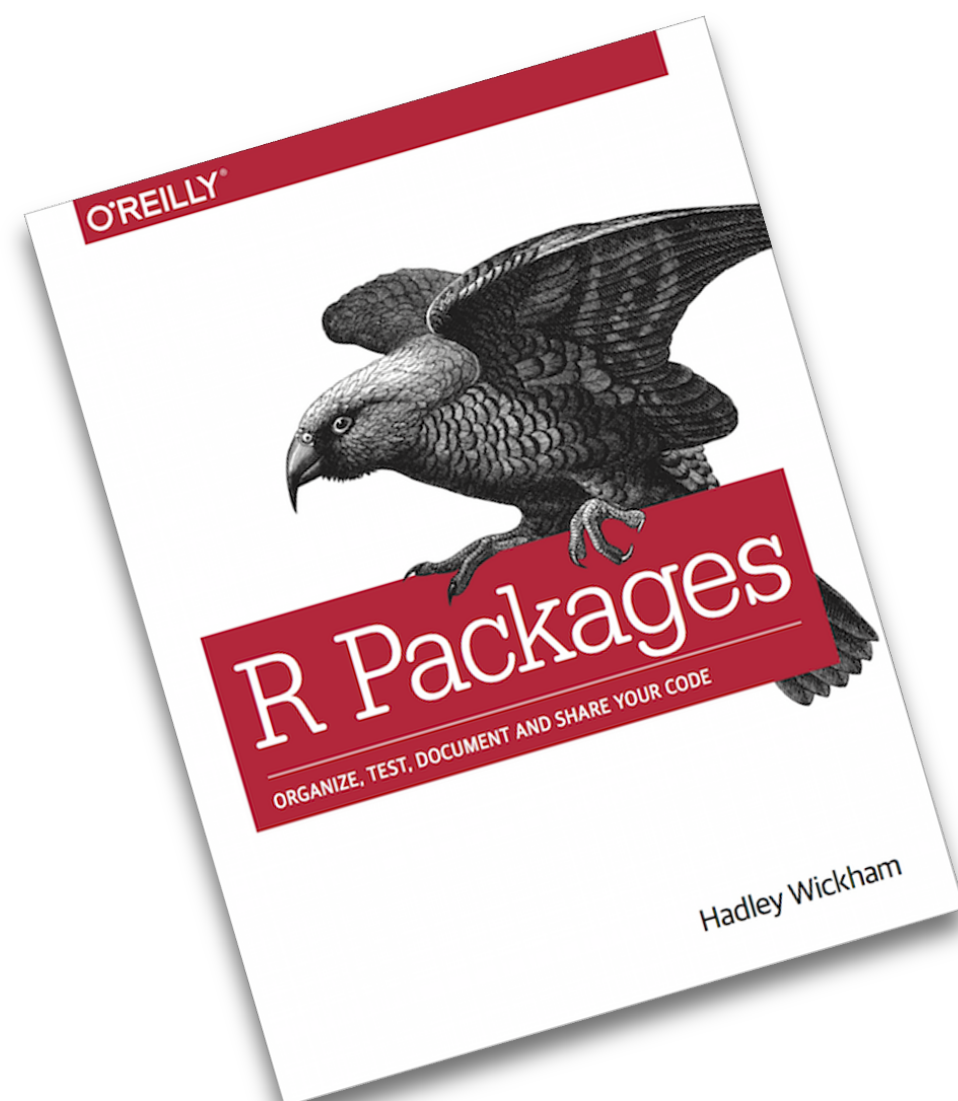
**R translates to
.html for viewing**



Create your own package!

Write your R package

- Start thinking about writing an R package as soon as you write your first function for your project, or even before.
- `usethis::create_package("mypackage")` will setup your package
- Check if the package name is available: `available::available("mypackage")`
- Easy to share, great documentation, an excuse to keep your folders tidy!



r-pkgs.org/



Build a website for your package

- `pkgdown::build_site()` will create your website

Home page with “home” icon:

- automatically generated from one of the following four files: `index.Rmd`; `README.Rmd`; `index.md`; `README.md`

Reference:

- The function `reference` generates one page for each `.Rd` file in `man/`, by default generate an overall index, which is an alphabetically ordered list of functions.

If the files are available in the package:

Articles:

- automatically build the `.Rmd` vignettes

News:

- if `NEWS.md` is present

Get Started

- if you have an article with the same name as the package.

A link to your your github repo (if listed in the `DESCRIPTION` url field).



pkgdown.r-lib.org/

Git & GitHub/GitLab

- Efficient and reliable code, well maintained and available on software development platforms e.g.,



GitLab



GitHub

Let's Git started

- For Version Control
- To share your work
- To collaborate



happygitwithr.com

Reproducibility & Dynamic report

- File format easy to use for all the collaborators, such as



rmarkdown.rstudio.com



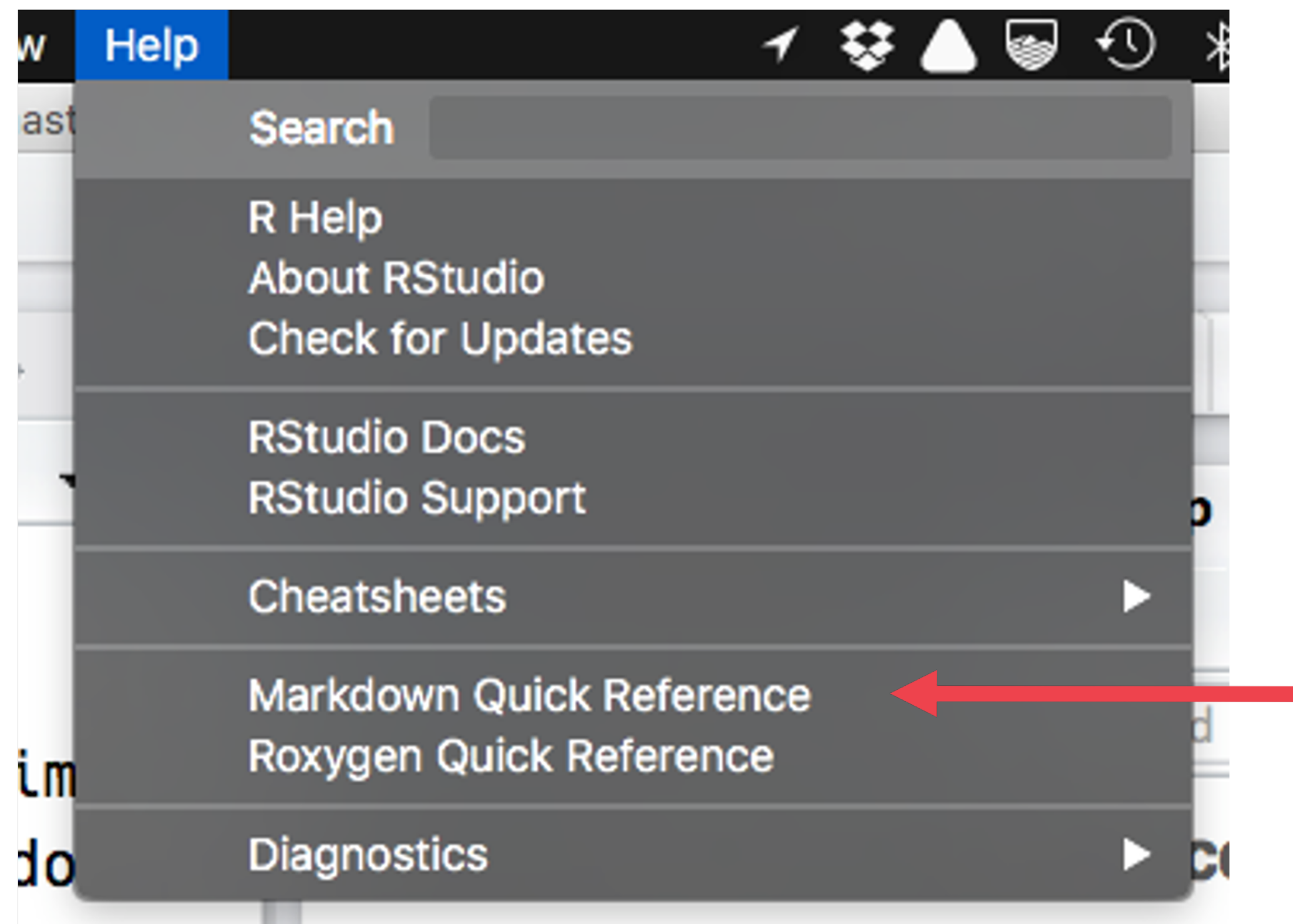
quarto.org

Where you intersperse code chunks (R, Python, Julia, C++, SQL) with markdown text

- Options can be controlled on a document or chunk level whether to show code and/or output.

Reproducibility & Dynamic report

- Incorporate code inline by using starting with ``r`` and ending with ```, or code chunks by starting with ``{r}`` and ending with ```



Reproducibility & Dynamic report

- Use R Sweave if you like to add R code on LaTeX.
 - Just change the file extension from .tex to .Rnw
 - Incorporate code inline by using `\Sexpr{ }`, or code chunks by starting with `<<>>=` and ending with `@`
 - `vignette("Sweave", package = "utils")`

yihui.org/knitr/





```
if (identical(method, "glm.fit"))  
  control <- do.call("glm.control", control)  
mt <- attr(mf, "terms")  
Y <- model.response(mf, "any")  
if (length(dim(Y)) == 1L) {  
  nm <- rownames(Y)  
  dim(Y) <- NULL  
  if (!is.null(nm))  
    names(Y) <- nm  
}  
X <- if (!is.empty.model(mt))  
  model.matrix(mt, mf, contrasts)  
else matrix(, NROW(Y), 0L)  
weights <- as.vector(model.weights(mf))  
if (!is.null(weights) && !is.numeric(weights))  
  stop("'weights' must be a numeric vector")  
if (!is.null(weights) && any(weights < 0))  
  stop("negative weights not allowed")  
offset <- as.vector(model.offset(mf))  
if (!is.null(offset)) {  
  if (length(offset) != NROW(Y))  
    stop(gettextf("number of offsets is %d should equal  
length(offset), NROW(Y)), domain = NA")  
}
```

Not all collaborators love coding

- Choose alternative to show you code:
 - Just show the output if working on a document
 - Create a nice package with user friendly functions that minimise the amount of code needed
 - Create nice interfaces

shiny.rstudio.com/



Teach (& Learn)





Be part of your communities

- Engagement & interaction with other practitioners is crucial



Recap

- Write neat code full of comments
- Use  **GitLab**  **GitHub**
- Write reproducible and dynamic reports
- Your code must be accessible and easy to use
- Share your work and listen to all feedback



THANK YOU!

