Welcome R-Ladies

Worldwide organization that promotes **gender diversity** in the **R** community via meetups and mentorship in a **friendly** and **safe** environment

5:30-5:45	Virtual Drinks & Networking
5:45-6:45	R tips & tricks!
6:45-7:00	Virtual Drinks & Networking

OK if we take pictures (or rather, screenshots)? *Otherwise, let us know!*

Slides adapted from R-Ladies Lausanne, thank you!

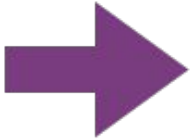
Code of Conduct

R-Ladies have created a code of conduct that **everybody** participating must agree to.

Excerpt:

R-Ladies is dedicated to providing a harassment-free experience for everyone. We do not tolerate harassment of participants in any form.

Read the full version here:
`https://github.com/rladies/starter-kit/wiki/Code-of-Conduct`



This ensures that the environment of the meetups stay **safe** and **friendly**

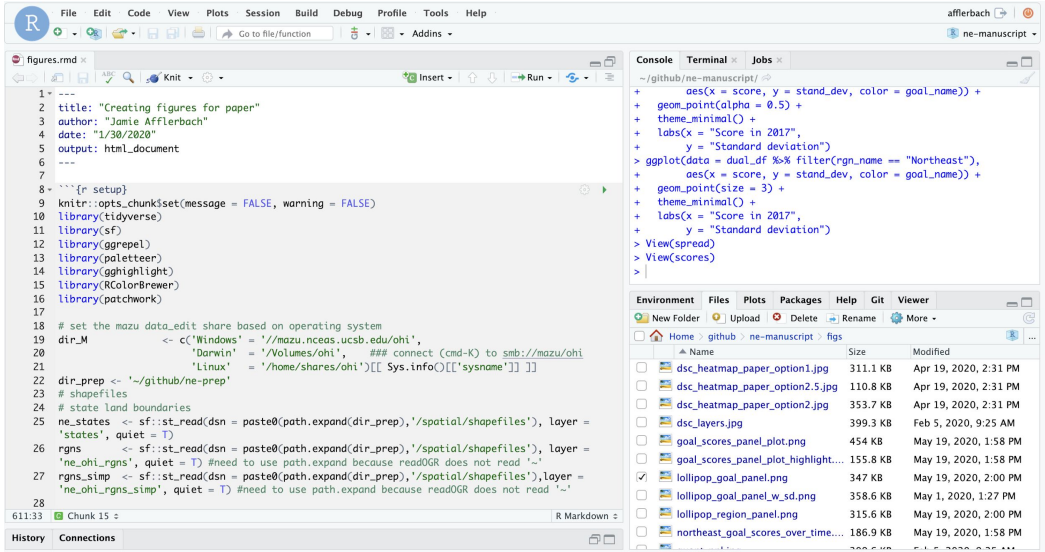
Part 1: R layouts & keyboard shortcuts

Part 2: packages & external resources

Part 3: Workflows

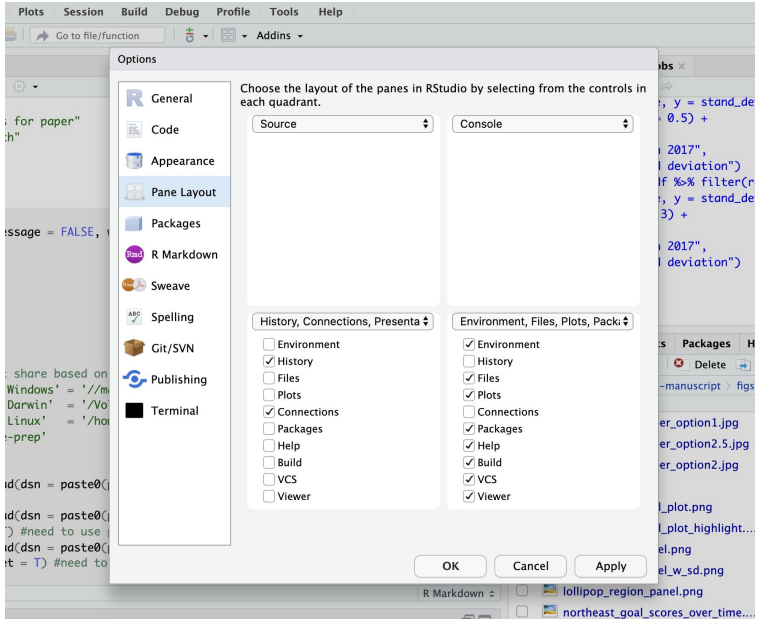
- Danielle Ferraro
- Jamie Montgomery
- Jasmine
- Ana Miller-ter Kuile
- An Bui
- Allison Horst
- Erin Winslow
- Juliette Verstaen

Jamie Montgomery - reorganizing RStudio panes

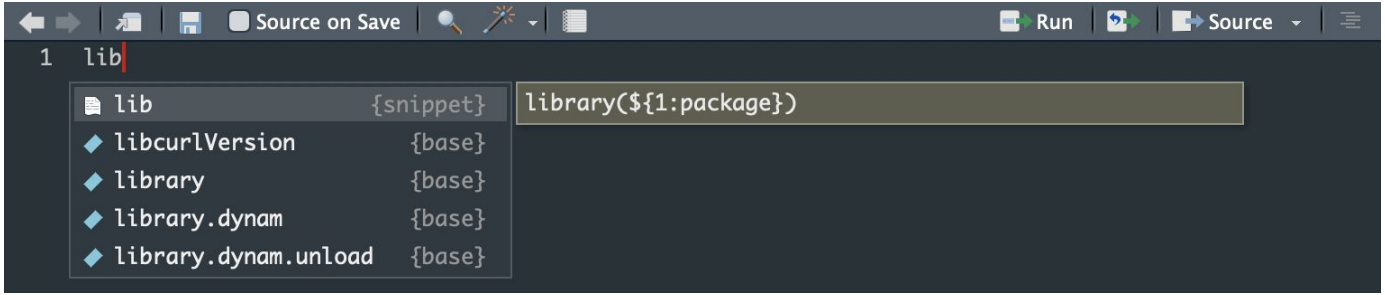


Why is it useful? Remove rarely used panes to increase visible space for scripting!

Customize by navigating to Tools > Global Options > Pane Layout (left sidebar)

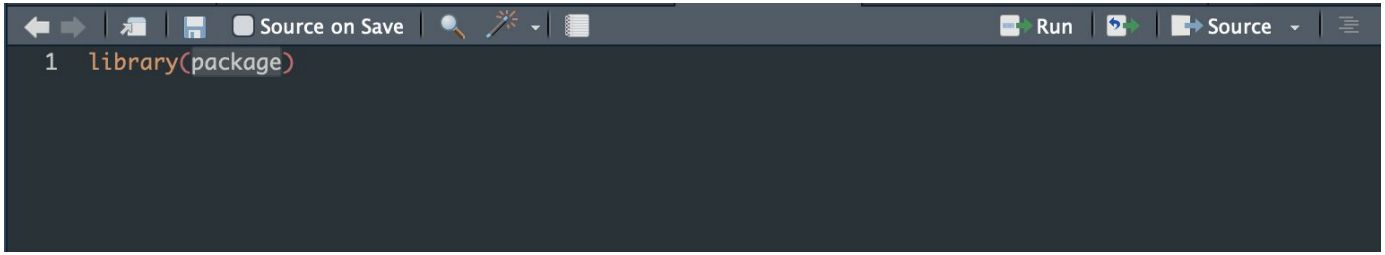


Danielle Ferraro - snippets make typing common code faster



You can see all default code snippets and add yours by clicking on Tools > Global Options > Code (left sidebar) > Edit Snippets

E.g. Type the first 3 characters '`lib`' then press Tab or Enter



Why is it useful? The '`library()`' auto fills AND highlights the space for you to automatically begin typing your package

Jasmine - RStudio “multi-line” cursor

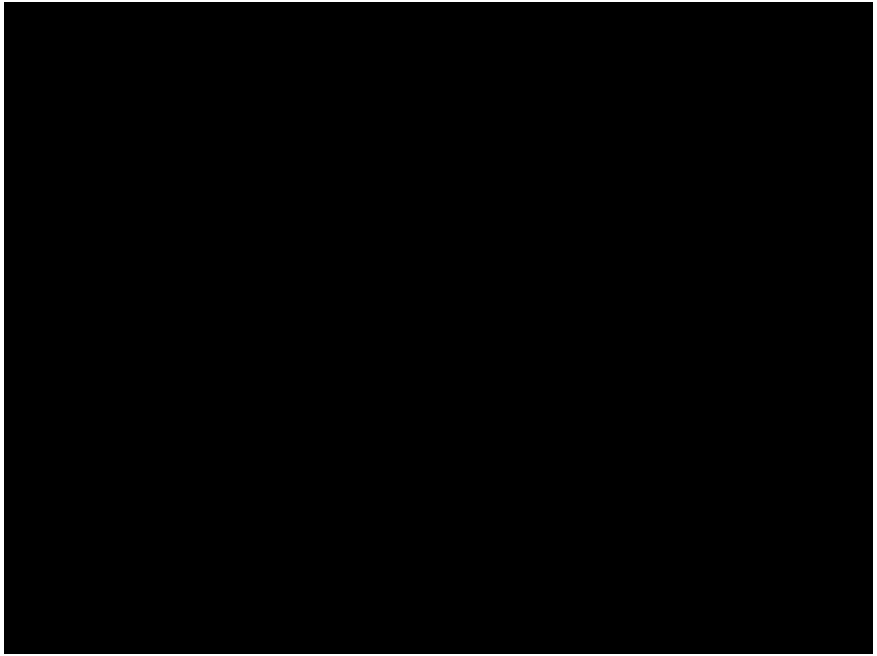
Why is it useful? Edit large chunks of similar code easily!

on mac: ctrl + option/alt + up/down arrow key or drag mouse

From Eiko Fried @EikoFried

Ana Miller-ter Kuile - quickly making assignment arrows

Why is it useful? It can be tedious to type out `<-` each time you need to assign an object



Option + - (on mac)

alt + - (on PC)

An Bui - shortcut for the {magrittr} pipe

The pipe operator in {magrittr} lets you *pipe* functions together to streamline your data processing!

Why is it useful? It's tedious to type out %>%, but the keyboard shortcut is

cmd+shift+M (on mac) or

ctrl+shift+M (on PC)

```
# broke:



# woke:
# cmd+shift+M (Mac) or ctrl+shift+M (PC)
```

An Bui - make your own operator

What's the opposite of the `%in%` operator?

Let's say you want to add a column to this data frame based on the "species" column.

```
> df
# A tibble: 4 x 2
  species count
  <chr>   <dbl>
1 crow       10
2 robin       7
3 owl         6
4 turkey      8
```

You have a vector you want to use to fill in a new column...

```
# a vector of characters
assholes <- c("crow", "turkey")
```

You make your new operator...

```
# the opposite of the %in% operator!
'%!in%' <- function(x,y)!('%in%'(x,y))
```

And use it!

```
# pipe that bad boy in
df_mutate <- df %>%
  mutate(quality = case_when(
    species %in% assholes ~ "bad",
    species %!in% assholes ~ "good"
  ))
```

The output!!

```
> df_mutate
# A tibble: 4 x 3
  species count quality
  <chr>   <dbl> <chr>
1 crow       10 bad
2 robin       7 good
3 owl         6 good
4 turkey      8 bad
```

s/o to Stack Overflow

```
'%!in%' <- function(x,y)!('%in%'(x,y))

c(1,3,11)%!in%1:10
[1] FALSE FALSE  TRUE
```

share edit follow

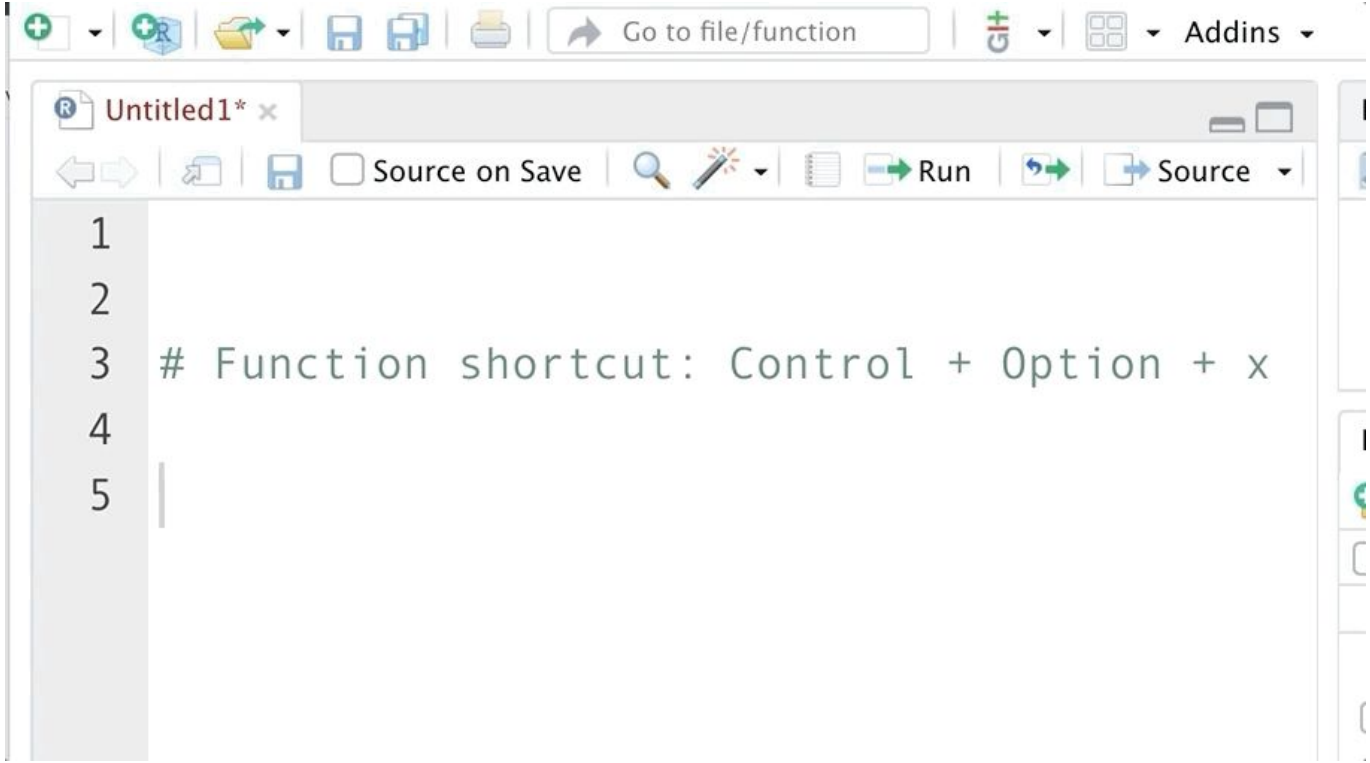
edited Sep 27 '18 at 22:04

JulienD

5,531 ● 6 ● 36 ● 65

Allison Horst - function notation shortcut

Why is it useful?
Way more efficient and you don't have to remember function notation



Control + Option + x

Erin Winslow - commenting code out shortcut

Why is it useful?
If you want to comment multiple lines out at a time, it's much faster!

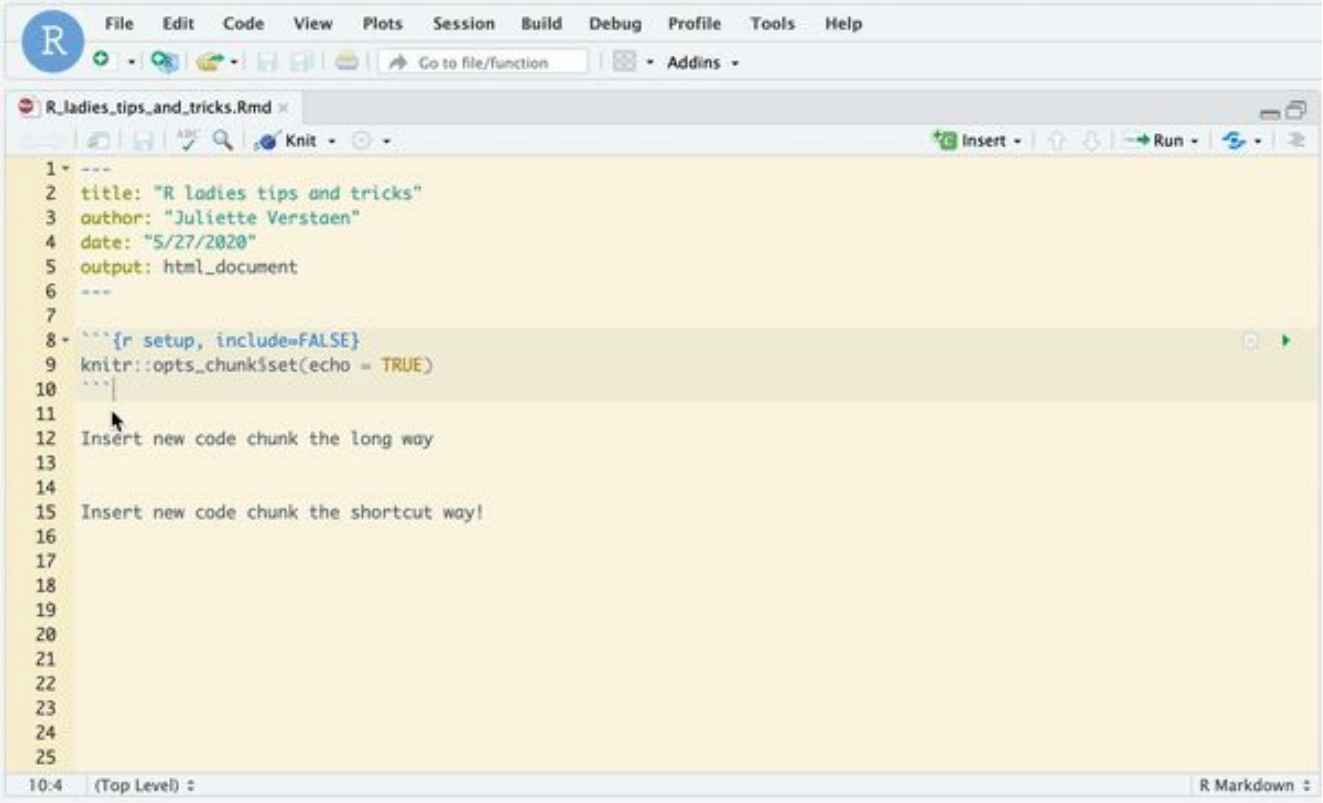
Command + Shift + c

```{r eelgrass_score_plot}

ggplot(eelgrass, aes(x = year, y = score, color = rgn_name)) +
  geom_line() +
  theme_bw() +
  geom_text_repel(data = subset(eelgrass, year == max(year)),
                  aes(label = rgn_name),
                  size = 3,
                  hjust = 0,
                  direction = "y",
                  nudge_x = 2,
                  segment.color = NA) +
  labs(x = "Year",
       y = "Score",
       title = "Eelgrass") +
  theme(legend.position = "none")

```

Juliette Verstaen - insert code chunk shortcut

Why is it useful?
Faster, and don't have to take hands off keyboard

Command + Option + i



Part 1: R layouts & keyboard shortcuts

Part 2: packages & external resources

Part 4: Workflows

- Ana Guerra
- Megsie Siple
- Francis Joyce
- Tracey
- Juliette Verstaen
- Ronnie Bailey-Steinitz

Ana Guerra - specify a function from a particular package

```
> tidyverse_conflicts()
── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
```

Sometimes R gets confused and it's good to let it know what you want - use the double colon '::' to do that!

Layout:

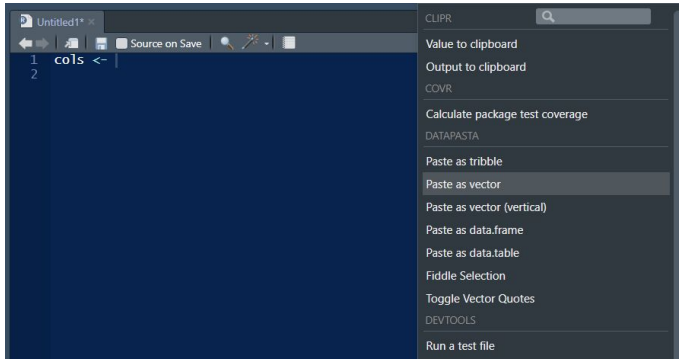
package::function

Megsie Siple - {datapasta} and RStudio addin

1. Find a color palette online

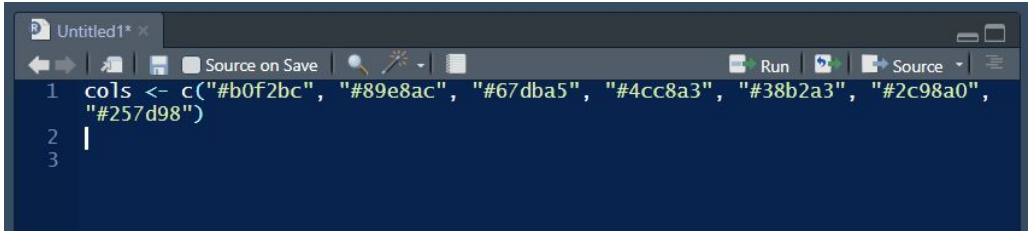
Use when pasting data from some other place (e.g. email, online table, text file) to get data formatting properly to analyze!

2. Paste that palette as a vector using {datapasta}



Eliminates needing to manually add quotations etc. when formatting as a vector--similar to `dput()` but for objects outside the R environment

3. The hex codes are a vector!

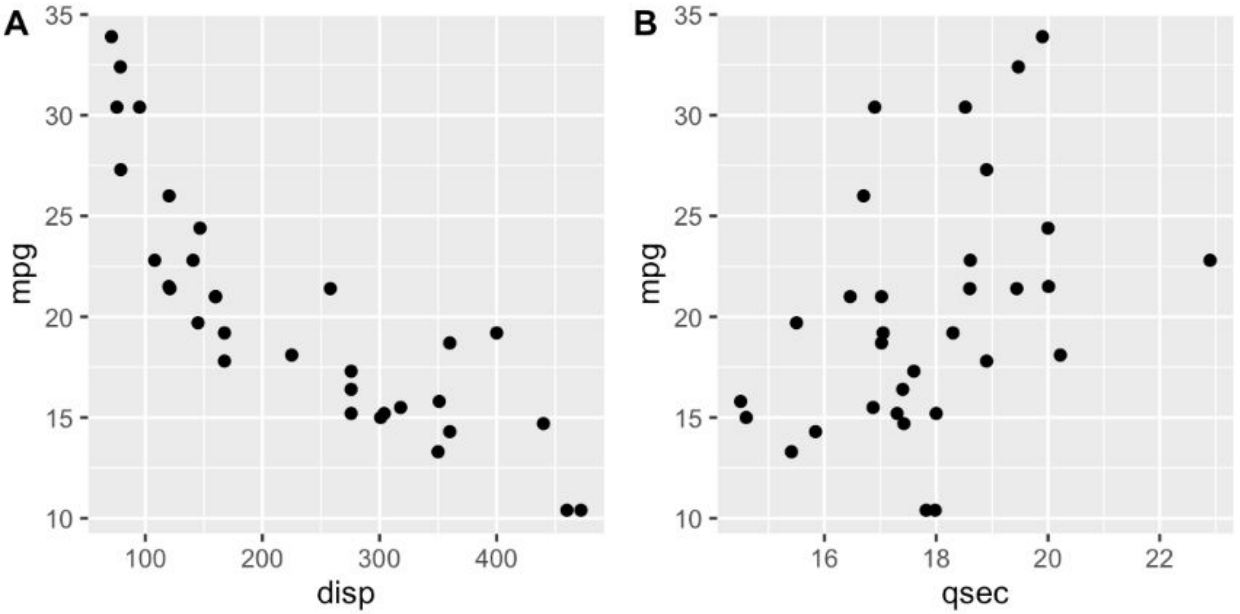


Francis Joyce - {cowplot}

```
plot_grid(p1, p2, labels = c('A', 'B'), label_size = 12)
```

Why is this useful?

Makes creating publication quality figures easier.



From Claus Wilke's tutorial:
https://wilkelab.org/cowplot/articles/plot_grid.html

Tracey- use {rebus} to “build regular expressions in a human readable way”

```
  report_txt
  <chr>
1 37YO f fell trail running - broken wrist
2 33YRF fracture suspected, slipped hiking
3 31 YO M concussed - hit head against wall bc reg expressions
4 1.5 YOF is sore - did a lot of squats
```

```
> reports4
# A tibble: 4 x 5
  report_txt                                                    gender age_unit age   unit
  <chr>                                                         <chr>  <chr>    <chr> <chr>
1 37YO f fell trail running - broken wrist                      f      37YO     37    YO
2 33YRF fracture suspected, slipped hiking                      F      33YR     33    YR
3 31 YO M concussed - hit head against wall bc reg expressions  M      31 YO    31    YO
4 1.5 YOF is sore - did a lot of squats                         F      1.5 YO   1.5   YO
```

```r
## pattern to match sex
gender <- optional(SPC) %R% char_class("MFmf")

reports2 <- reports %>%
  mutate(gender = str_trim(str_extract(report_txt, pattern = gender)))

## pattern to match age
age <- DGT %R% optional(DGT) %R% optional(DOT) %R% optional(DGT)

## age unit
unit <- optional(SPC) %R% or("YO", "YR", "MO")

reports3 <- reports2 %>%
  mutate(age_unit = str_extract(report_txt, pattern = age %R% optional(SPC) %R% unit))

reports4 <- reports3 %>%
  mutate(age = str_trim(str_extract(age_unit, pattern = age)),
         unit = str_trim(str_extract(age_unit, pattern = unit)))
```

- Create readable patterns
- Anchors (START, END)
- %R% “and then…”

Useful links in the notes!

Juliette Verstaen - dplyr::setdiff()

	name	house
1	Cho Chang	Ravenclaw
2	Cedric Diggory	Hufflepuff
3	Luna Lovegood	Ravenclaw
4	Remus Lupin	Gryffindor
5	Draco Malfoy	Slytherin
6	Harry Potter	Gryffindor

Data Frame 1 = house_df

	character	bio
1	Cho Chang	Ravenclaw stude...
2	Cedric Diggory	Participated in t...
3	Hermione Granger	One of Harry's b...
4	Bellatrix Lestrange	Death Eater who...
5	Luna Lovegood	Ravenclaw stude...
6	Remus Lupin	Friend of James ...
7	Draco Malfoy	Slytherin studen...
8	Harry Potter	The boy who liv...
9	Nymphadora To...	Married Remus ...

Data Frame 2 = bio_df

Why is it useful?

1. Helps explore data frames you might want to join
2. Don't need to do any data wrangling beforehand!

What people are listed in **data frame 2** that are not in **data frame 1**?

```{r r-ladies tips & tricks}

setdiff(house_df$name, bio_df$character)

```

```
character(0)
```

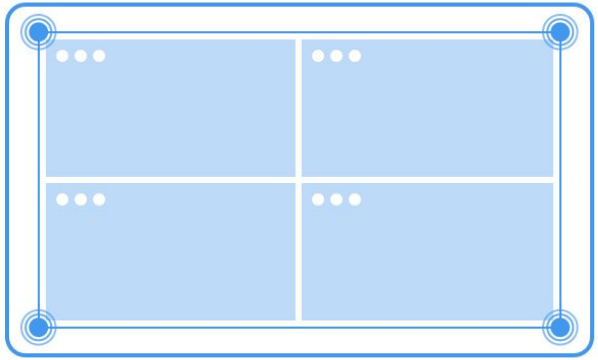
What people are listed in **data frame 1** that are not in **data frame 2**?

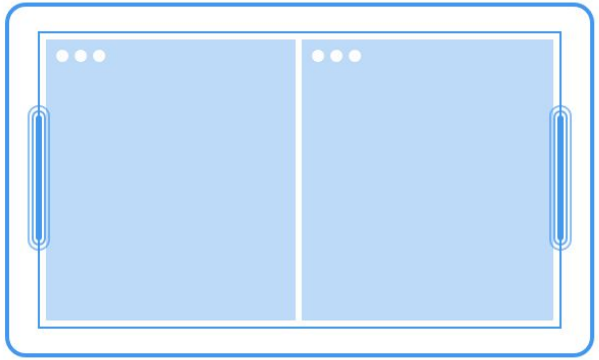
```{r r-ladies tips & tricks}

setdiff(bio_df$character, house_df$name)

```

```
[1] "Hermione Granger"    "Bellatrix Lestrange" "Nymphadora Tonks"
```

Ronnie Bailey-Steinitz - use the Magnet app for organizing windows on your one or dual screens

Keyboard shortcuts that automatically shrink windows to certain sizes and pins them to a location (e.g., half screen L, and half screen R)





Why is it useful? Makes for easier organization of your one or multiple monitors when you have 17 million windows open

Download for mac here: https://magnet.crowdcafe.com/

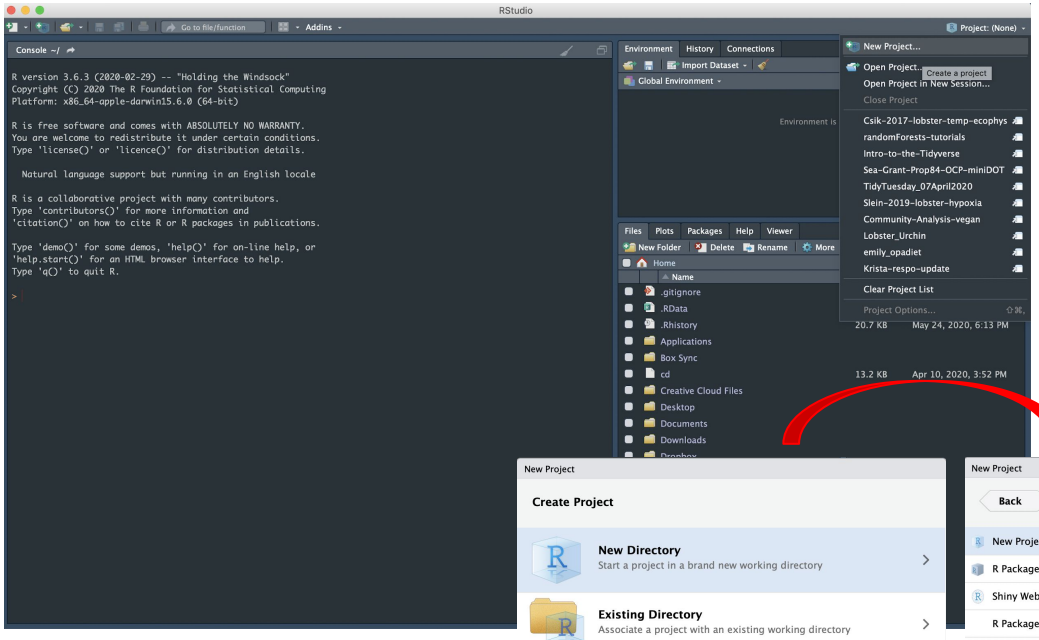
Part 1: R layouts & keyboard shortcuts

- Rocío
- Joe Curtis
- Sam Csik

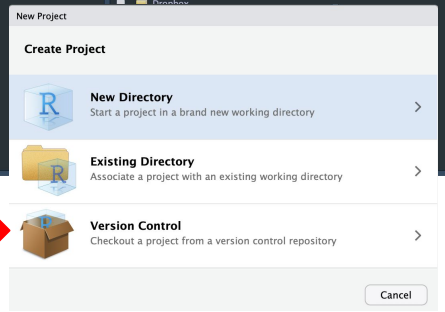
Part 2: packages

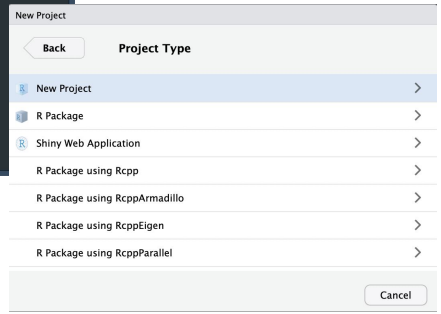
Part 3: Workflows

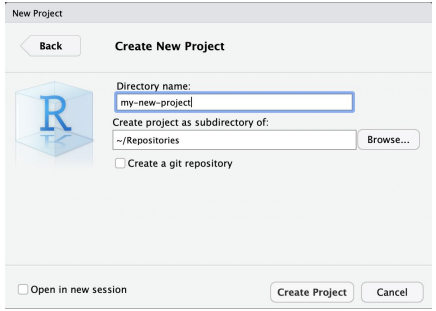
Rocío - project oriented workflow (1 of 2)



Click here to start a new project, or switch between projects



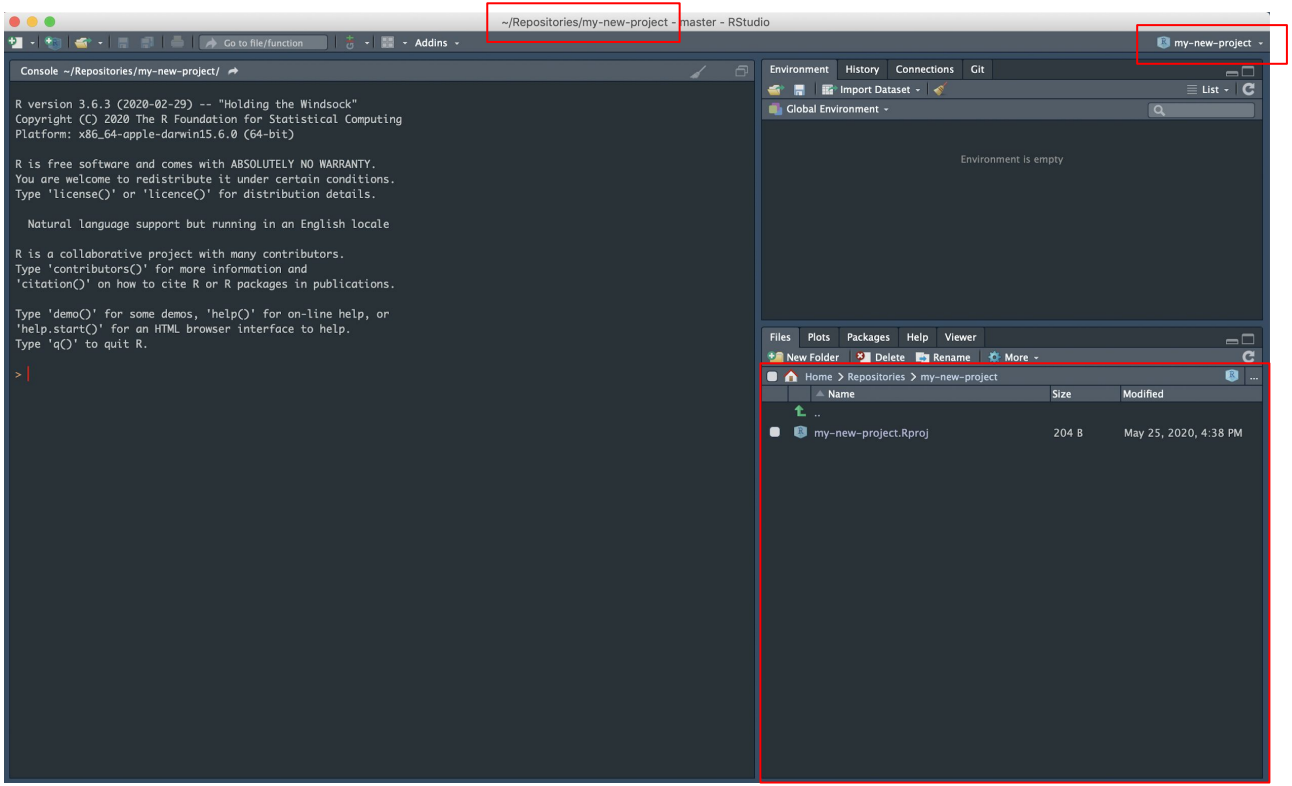




Check out the option to use version control (e.g. GitHub) for your project too!

Read more about project-oriented workflows here: https://www.tidyverse.org/blog/2017/12/workflow-vs-script

Rocío - project oriented workflow (2 of 2)



Why is it useful?

- more structure
- no more `setwd()`
- better for you and the people you share it with
- the list goes on and on!

Read more about project-oriented workflows here: https://www.tidyverse.org/blog/2017/12/workflow-vs-script

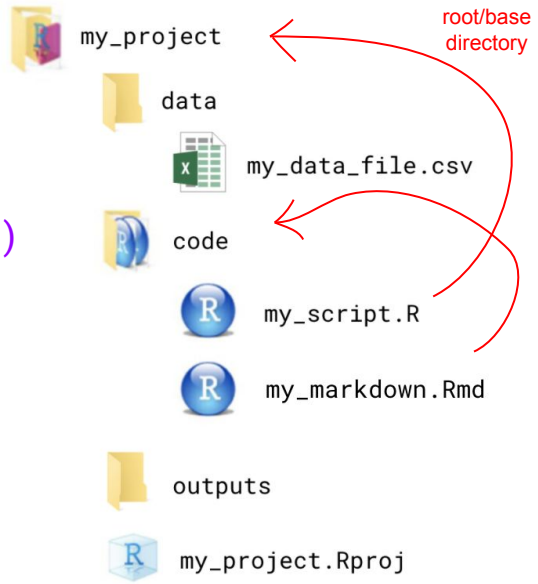
Joe Curtis - use an organized file structure + {here} to reliably locate your project's files relative to your root directory

{lubridate} & {here} both have a function called `here()` so use Ana Guerra's tip to avoid package clashes!

Combine `here::here()` with `read_csv()` to build a file path to the top level of your project file

```
my_data <- read_csv(here::here("data" "my_data_file.csv"))
```

Part of the reason file paths can be so difficult:



Why is it useful?

- Eliminates fragile "hard-wiring" of file paths (e.g. `setwd()`)
- Easily specify file paths in your .R and .Rmd files regardless of where they live

Read more about {here}...here: http://jenrichmond.rbind.io/post/how-to-use-the-here-package/ & https://github.com/jennybc/here_here

Sam Csik - separate script just for packages

libraries.R

Source on Save | Run | Source

```r
############################
# libraries
############################

library(bbmle) # maximum likelihood estiamtion used for FR analyses
library(boot) # bootstrapping
library(broom) # used alonside nls.multstart for tidying up model outputs
library(car) # model predictions
library(chron) # working with dates
library(cowplot) # organizing fig panels & printing plots
library(effsize) # calculate effect sizes
library(frair) # fitting FR
library(ggridges) # ridgeline plots for displaying LTER temps
library(gridExtra) # making knitted tables pretty
library(here) # because we love reproducibility
library(kableExtra) # knitting tables
library(lubridate) # tidyverse friendly dates
library(minpack.lm) # nonlinear least squares for fitting FR
library(naniar) # dealing with missing data in LTER time series
library(nlme) # nonlinear mixed effects models for consumption data
library(nls.multstart) # fitting FR
library(nlstools) # used alongside nls.multstart for estimating CI of non-linear regression
library(nls2) # nonlinear least squares with brute force for fitting FR
library(patchwork) # arranging plots
library(tidyverse) # the true hero
```

Create a separate script called `libraries.R` with all your packages

```r
##############################
# import libraries
##############################

source("libraries.R")
```

Why is it useful? Import packages into any other related scripts with one short line of code!

```
####**Packages Required:**

    - tidyverse
    - here
    - gridExtra
    - effsize
    - car
    - kableExtra
    - boot
```

But it's also good practice practice to keep track of dependencies for each script :)

Notes from the chat

- “notin”:
  - Gage Clawson: alternative to %!in% is `%notin%` <- Negate(`%in%`)
  - Allison Horst: Another option for case_when to capture anything that’s not included in previous conditions is to use a final else statement of ’T ~ whatever’
- {here}
  - Tannis Thorlakson: Best here package summary: https://malco.io/2018/11/05/why-should-i-use-the-here-package-when-i-m-already-using-projects/
  - Allison/Danielle Ferraro/Megsie Siple: here::here(“data”, “my_data.csv”) is better than here::here(“data/my_data.csv”)
- Colors
  - Gage Clawson: If you really love a certain color on your screen (this is for macs, not sure for PC), there is a app called “Digital Color Meter” which you can find in your finder. It will show you the rgb values on whatever you hover over with your mouse
- Updating Mac OS
  - Gage Clawson: This was super easy to fix the git issue when updating to catalina : https://michaelsoolee.com/git-not-working-macos-catalina-xcrun/

That’s All!

- We will be posting this presentation on the R-Ladies GitHub and sharing the link
- If you have anything else you’d like to add let us know !

Thank you to everyone who shared, we definitely learned some new tricks!