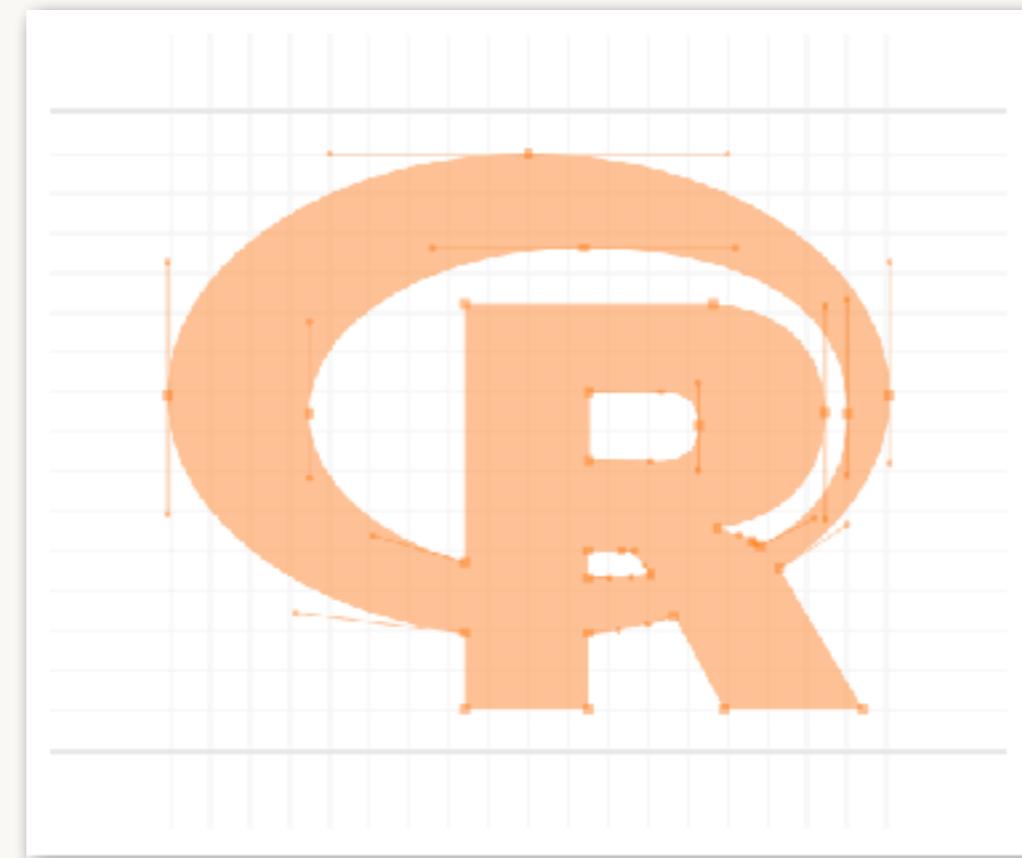


# INTRODUCTION TO DATA ANALYSIS WITH R

MAY 2018 – HANDS-ON DATA WORKSHOP



# Welcome!

**Download all datasets, by clicking Clone > Zip**

`https://github.com/epfl-exts/Intro-To-R-Workshop`

**Install more packages in R, by running**

`install.packages(c("tidyverse", "shiny",  
"xml2", "rvest", "openxlsx", "janitor"))`

# *Menu du jour*

**R, AN ECOSYSTEM FOR DATA**  
and for meeting great people

# Menu du jour

**09:00-10:30** Workshop

**10:30-11:00** Coffee Break

**11:00-12:30** Workshop

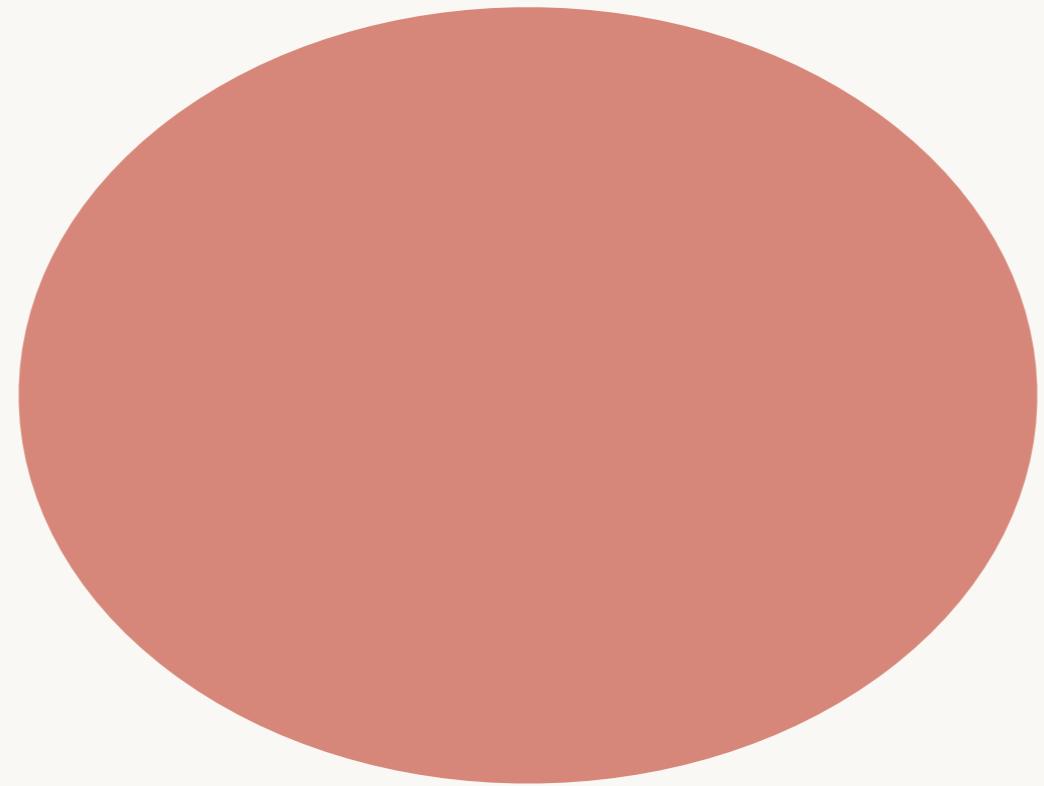
**12:30-13:30** Lunch Break

**13:30-15:00** Workshop

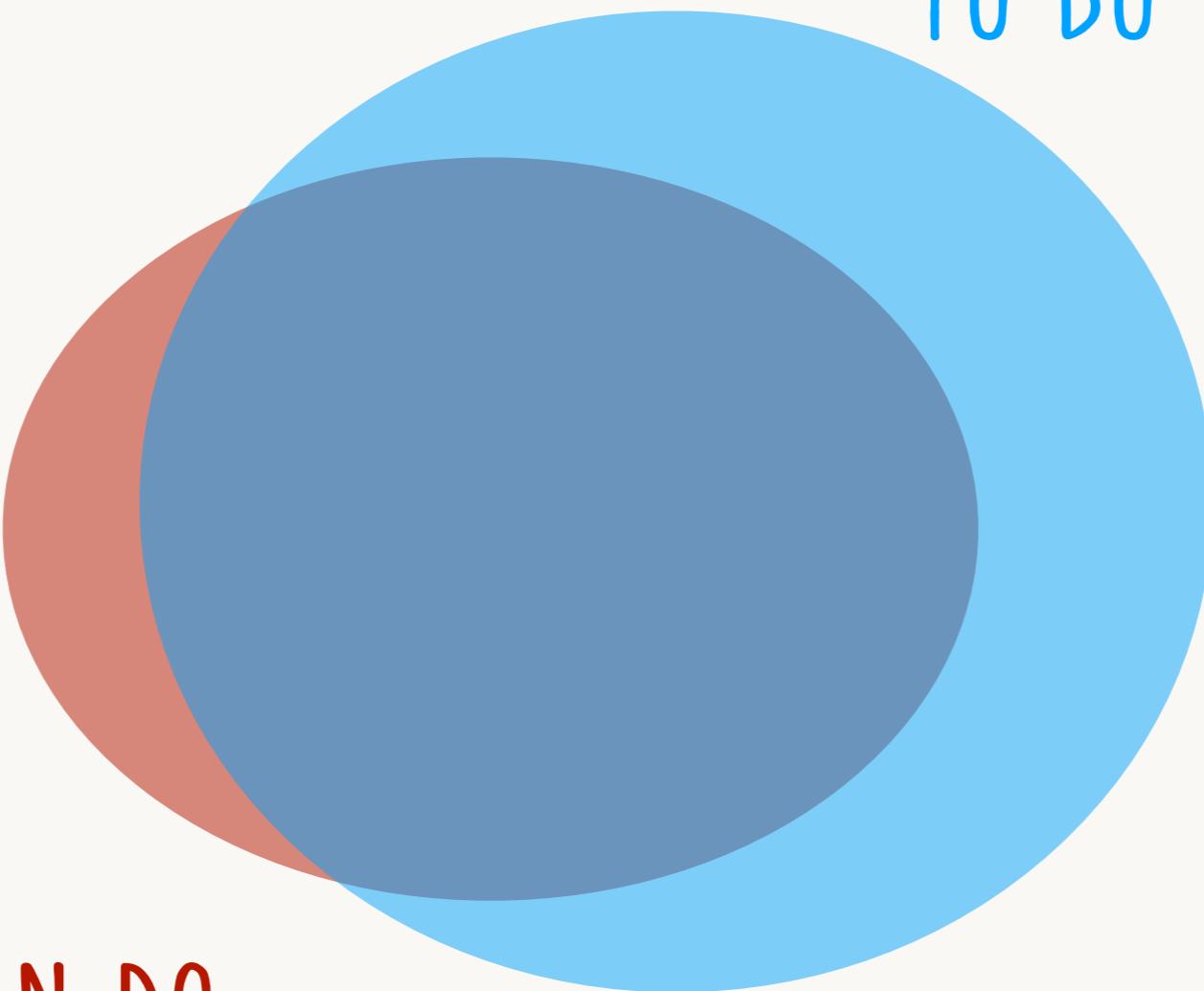
**15:00-15:30** Coffee Break

**15:30-17:00** Workshop

**17:00** End



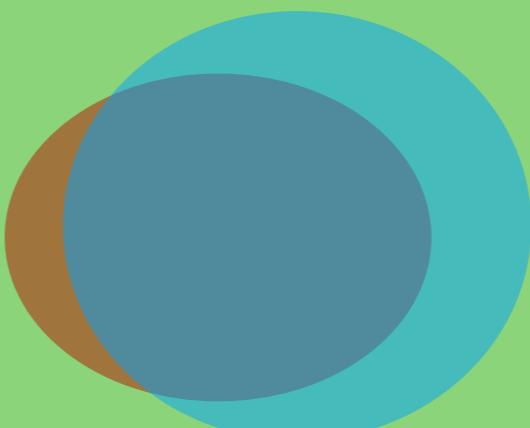
WHAT YOU CAN DO  
WITH EXCEL



WHAT YOU'LL LEARN  
TO DO WITH R TODAY

WHAT YOU CAN DO  
WITH EXCEL

# WHAT CAN BE DONE IN R





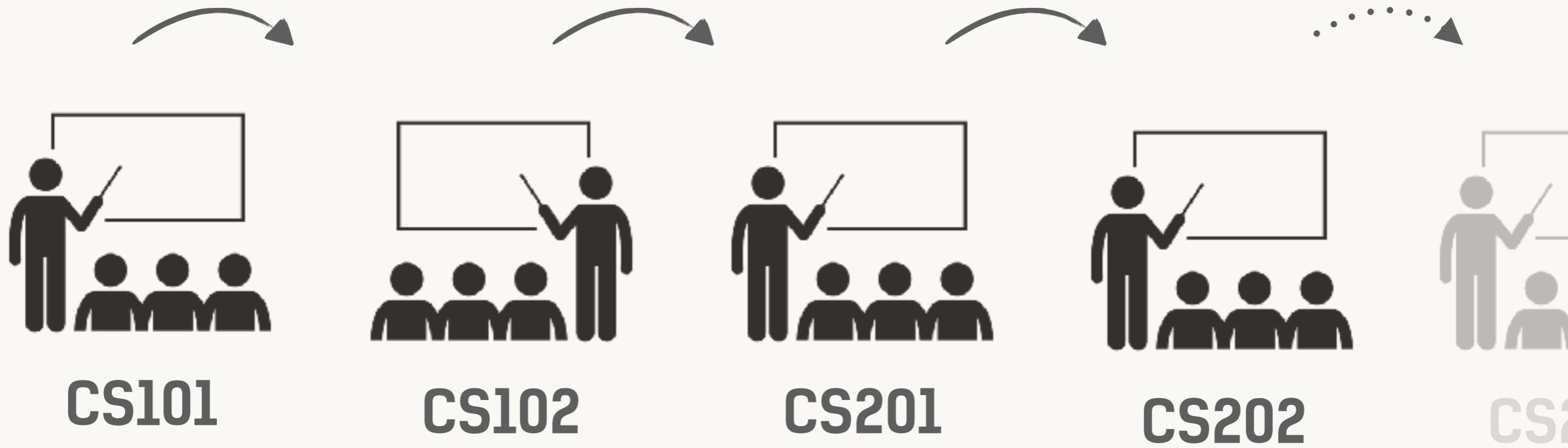
8 hours later...



8 hours later...

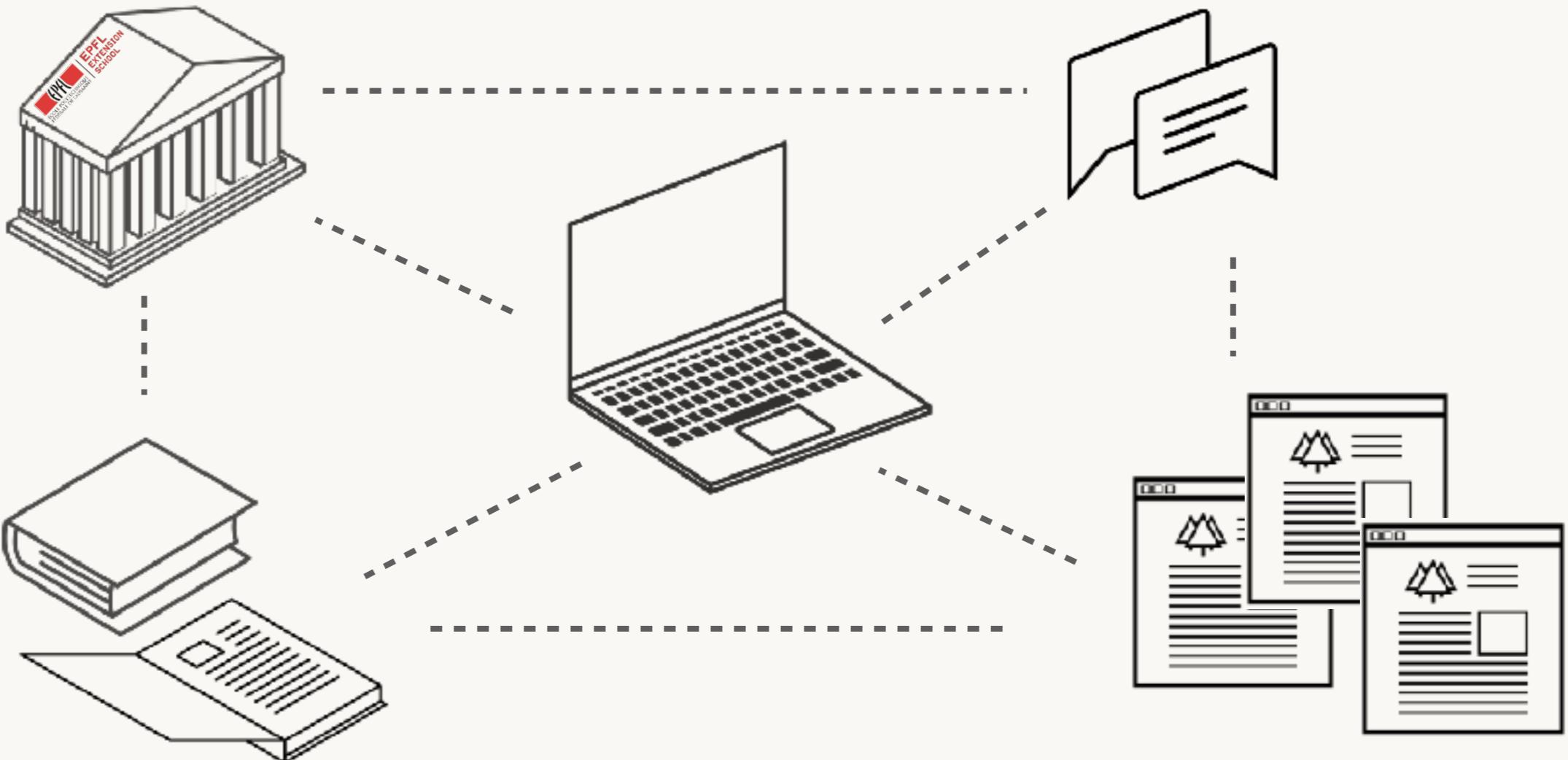
**What it's like to learn  
to code...**

# Learning to code (15 years ago)...



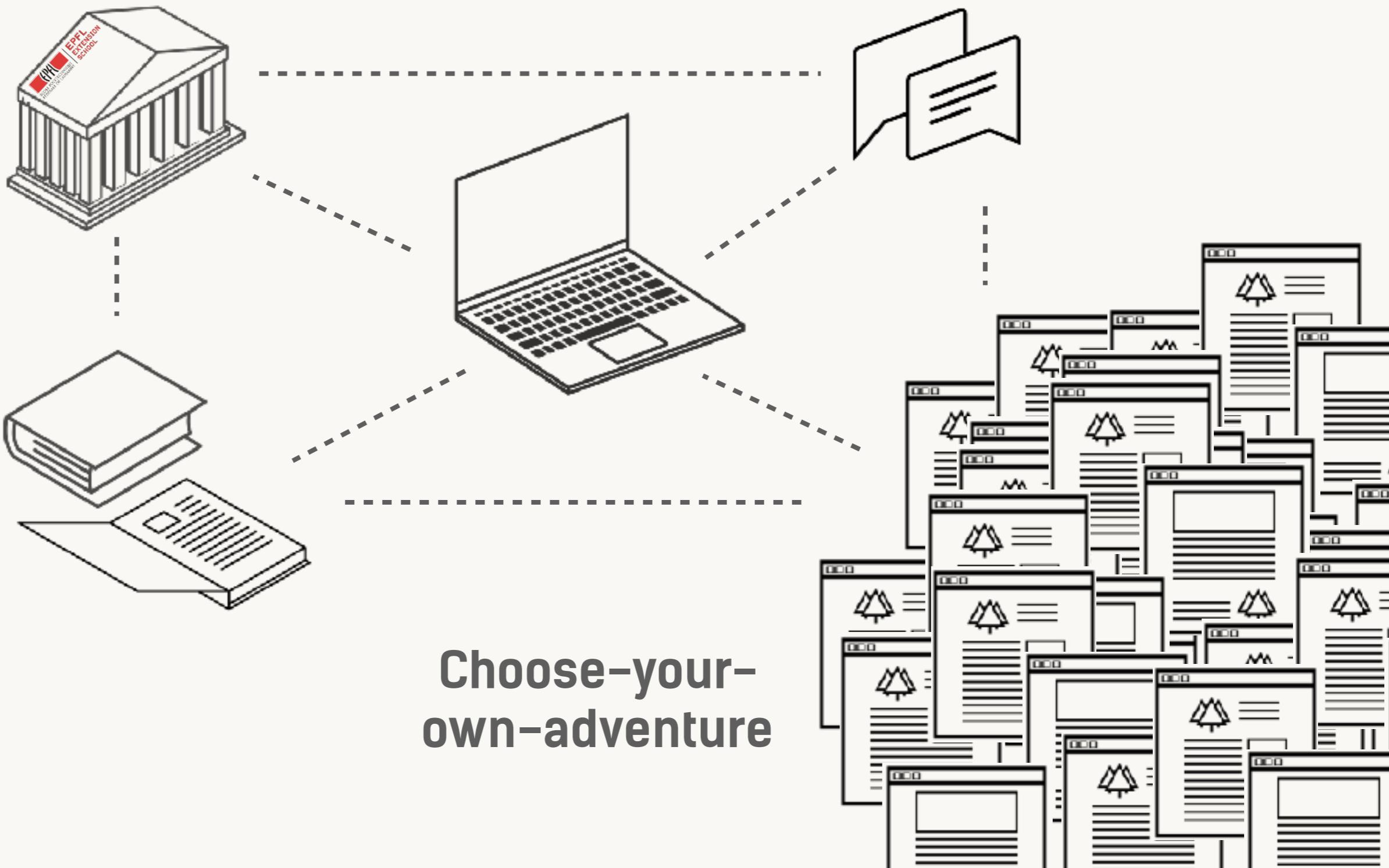
A “straight” path

# Learning to code (today)...

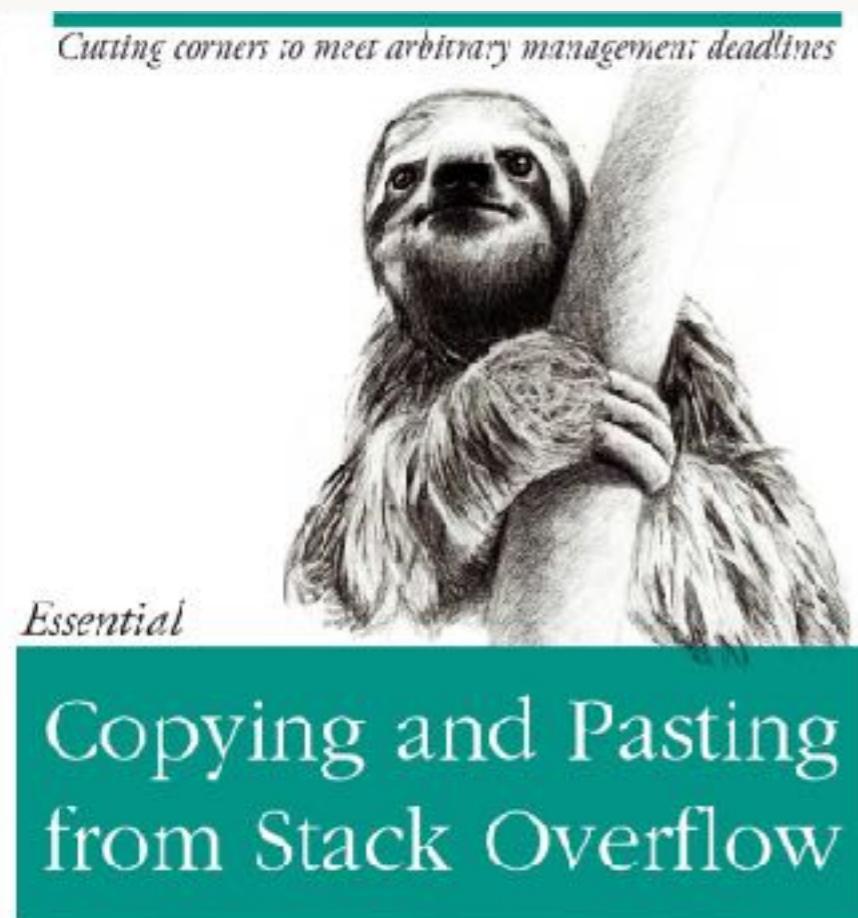


**Choose-your-  
own-adventure**

# Learning to code (today)...



# Learning from questions on stackoverflow.com is currently a big part of learning to code.



Cutting corners to meet arbitrary management deadlines

http://www.stackoverflow.com

Questions Developer Jobs Tags Users [r] how to extract substring 820 2 14 25 1 422 Ask Question

Search results found containing how to extract substring tagged with r

[r] how to extract substring search

69 results relevance newest votes active

R is a free, open-source programming language and software environment for statistical computing, bioinformatics, visualization and general computing. Provide minimal, reproducible, representative example(s) with your questions. Use `re`() for regex and specify all non-base packages with library ...

Learn more... Improve tag info Top users Synonyms (2) r jobs

Q: How to extract substring of input? How to extract substrings from this input: input = c("Inspecting sequence ID NM\_000775", "Inspecting sequence ID NM\_001014925", "Inspecting sequence ID NM\_00100484") to get output= c("NM\_000775", "NM\_001014975", "NM\_00100484") ... asked May 19 '15 by Paul J. 2 votes 5 answers 2 regres 2

Q: How to extract substring in R [closed] Response times will be capture in ServiceNow which, save in respect of manifest error, will be conclusive proof of the time period taken. I want to **extract** 2 weeks after the word "away". How this can be achieved in R? ... I have a string like the below: e.g. "Supplier will initially respond to High Priority incidents. Supplier will subsequently update EY every 60 minutes or at an interval EY specifies. Reporting and ... asked Jun 5 '17 by Anum -2 votes 2 answers 2 r string

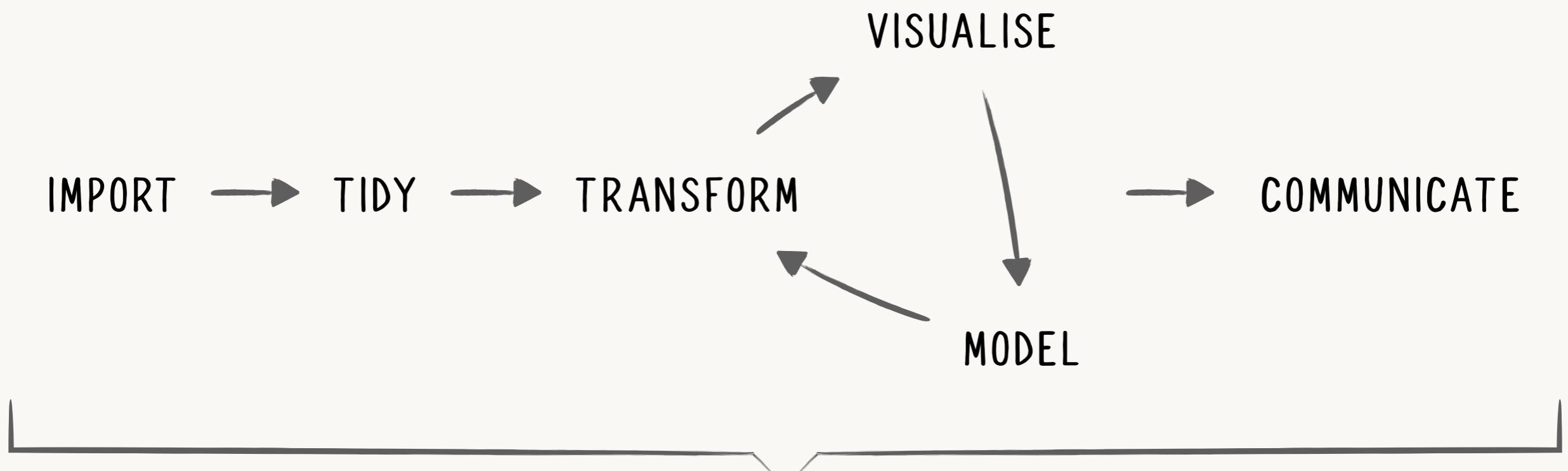
Hot Network Questions

- Which is the more standard package, getopt or optparse (with an 's')?
- Unidentified Plant that started blooming
- How can I protect my website against bitflipping?
- Running a simulation on pure Ubuntu vs on Ubuntu in Windows (WSL)
- How can I avoid infinity symbols in a PGFPlotsstable with calculated columns?
- Reaction of Grignard reagent with primary amines
- Is there a single word for both "zoom" and "chart"?
- How to get work done whilst feeling ongoing shame?
- If the 3-Body force problem hasn't been solved, how do rocket scientists plan orbits of spacecraft?
- In the definition of IS DS income?
- Probability of a specific event when rolling a die five times
- Cheese Cornered Puzzle
- Why do I need to place 'not' in the same line as

O'REILLY®

The Practical Developer  
@ThePracticalDev

# Why learn R

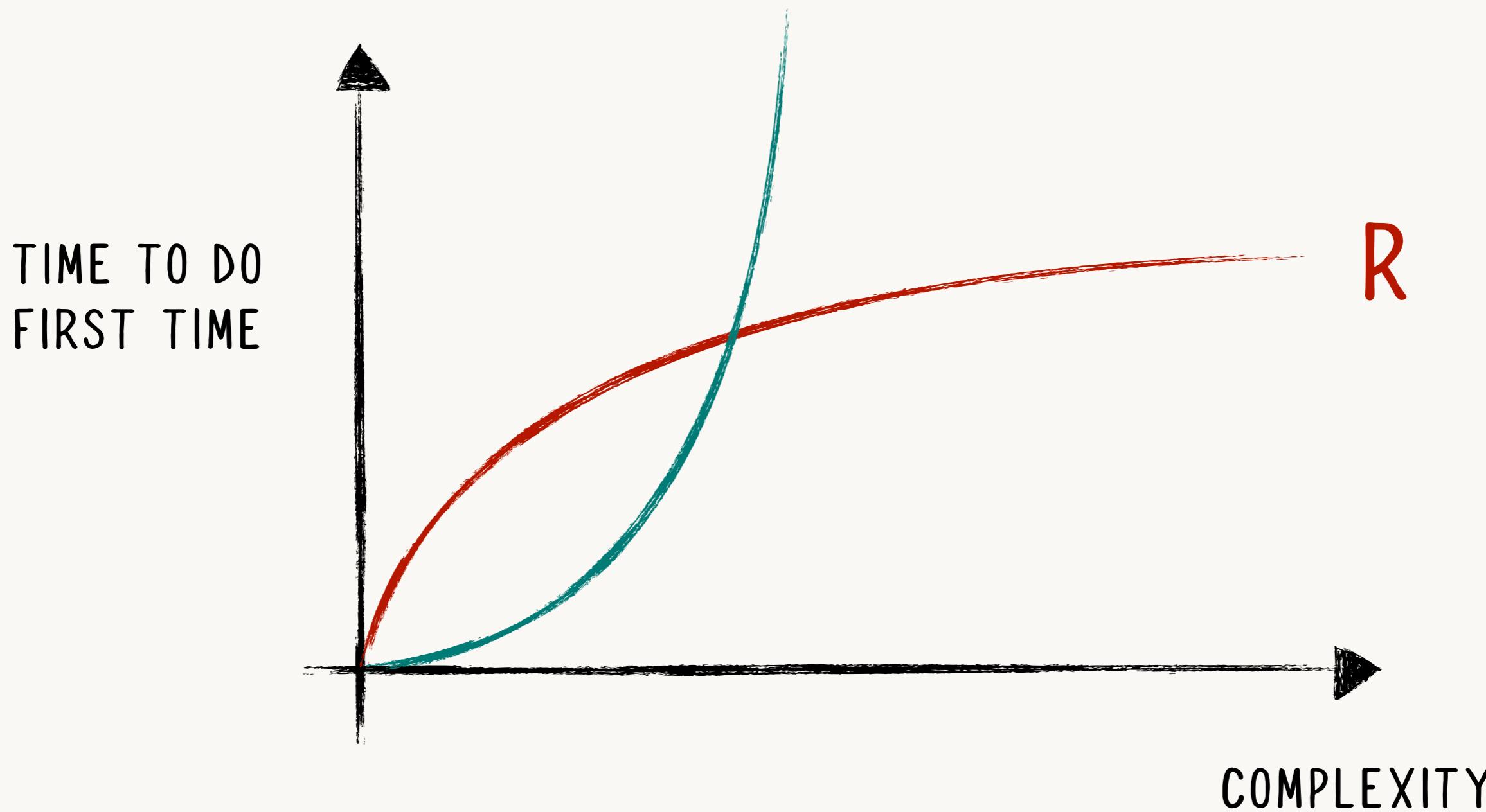


# **Complexity management**

Complex/long analysis are way  
easier to write than in Excel

# First time you do an analysis

EXCEL

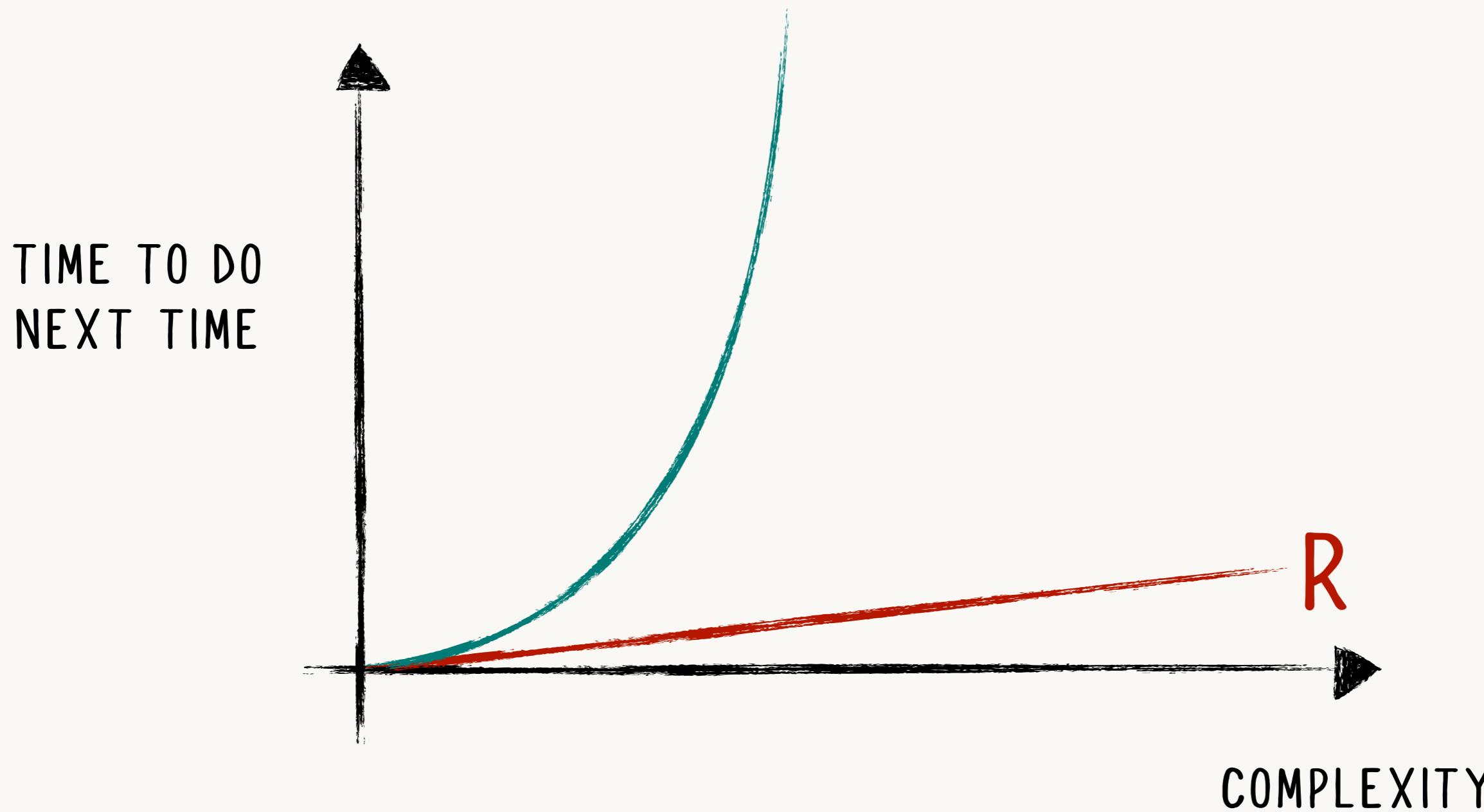


# Automation

Analysis can be shared, modified and  
repeated in an instant.

# Time when you redo an analysis

EXCEL



# Diversity of input/output

Write once, use everywhere.

R script can start from any data sources

and produce:

→ pdf,

→ word

→ excel

→ websites

→ dashboard

→ powerpoint...

# Functions and variables

## The building blocks of R

# How to talk to R

DO THIS ON THIS  
THEN THIS  
THEN THAT  
THEN THIS  
THEN THIS

THEN THIS  
THEN THIS  
THEN THAT  
THEN THIS  
THEN THIS  
THEN THAT

AND SO ON...

# R only speaks functions

```
function_name( argument(s) )
```

R only speaks functions  
WITH ONE SINGLE ARGUMENT

EXCEL            **LOWER("MY TEXT")**

R            **str\_to\_lower("MY TEXT")**

“my text”

R only speaks functions  
WITH MULTIPLE ARGUMENTS

EXCEL

**LEFT("MY TEXT", 2)**

R

**str\_sub("MY TEXT", end=2)**

**"MY"**

R only speaks functions  
AND THEY CAN BE NESTED

`LEFT(LOWER("MY TEXT"), 2)`

`str_sub(str_to_lower("MY TEXT"), end=2)`

“my”

# Nested function are challenging

(DO THAT LAST(DO THIS THEN(DO THIS FIRST)))



1

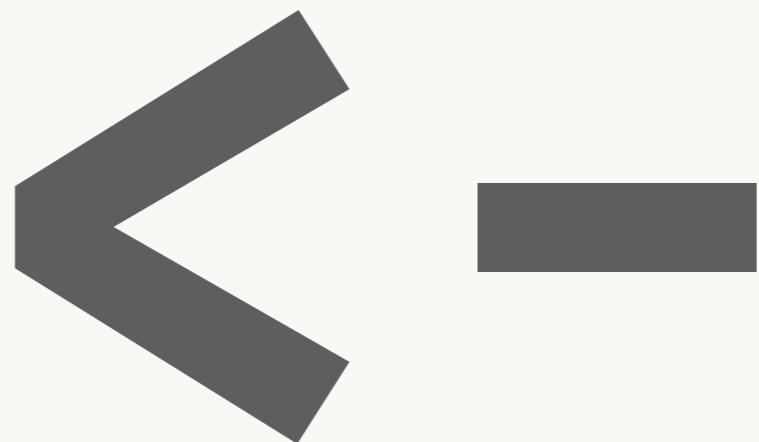


2

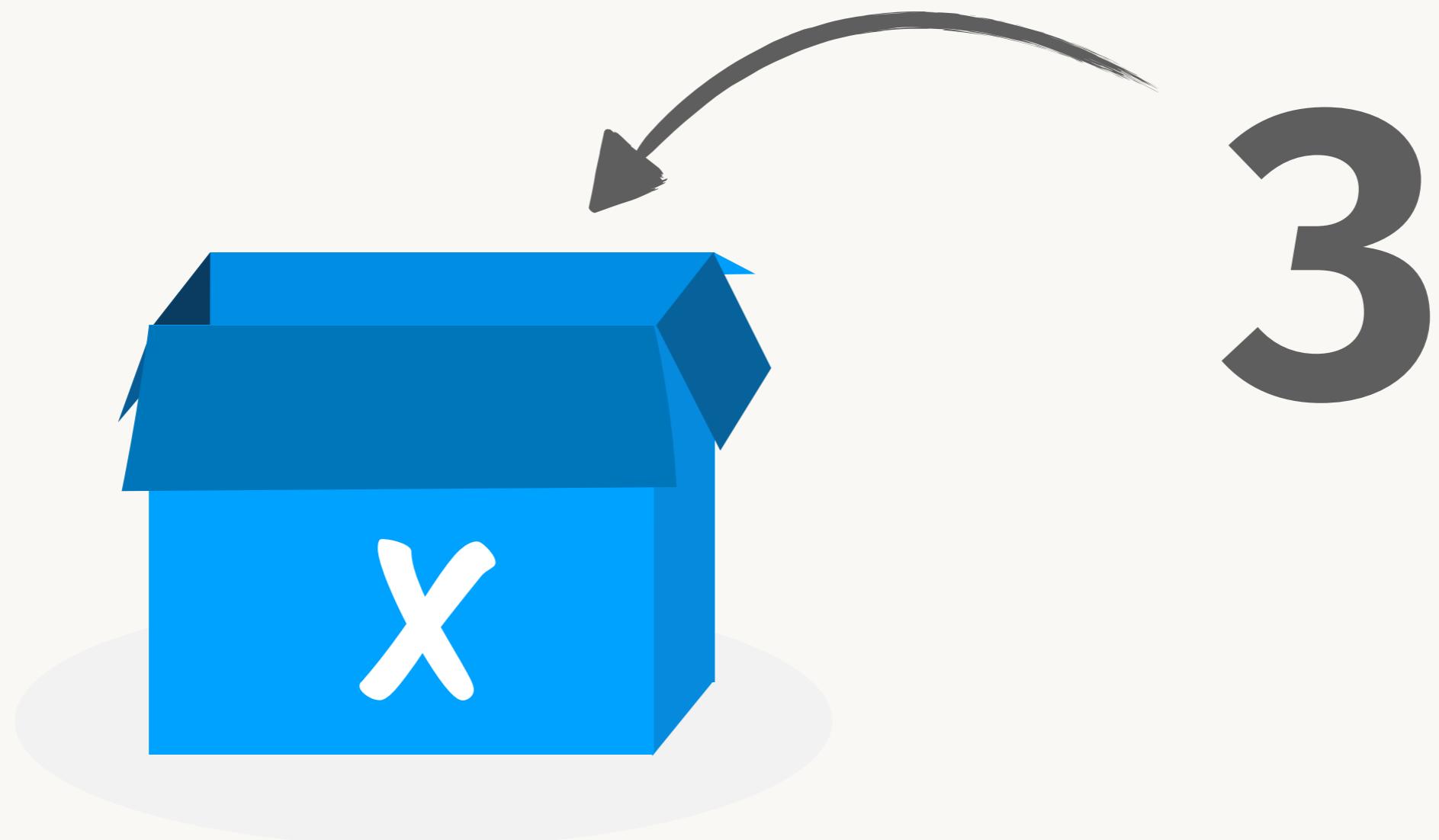


3

# The assign arrow(s)



# The assign arrow(s)



# The assign arrow(s)

X

VARIABLE NAME

<

-

3

VARIABLE VALUE

# Dataframe

## A “spreadsheet” in R

# The dataframe ( or “tibble” )

location	country	time	budget
PRT	Portugal	2006	1064372
MUS	Mauritius	2000	NA
HUN	Hungary	2012	1357253
EST	Estonia	2002	33372
URY	Uruguay	1998	24602
PRT	Portugal	2000	378849
ISL	Iceland	1999	79556
LUX	Luxembourg	2011	316044
MEX	Mexico	2015	2450078
IRL	Ireland	2013	1825961

# The dataframe ( or “tibble” )

```
# A tibble: 100 x 4
  location country   time budget
  <chr>     <chr>   <int>   <dbl>
1 PRT       Portugal 2006    1064372.
2 MUS       Mauritius 2000      NA
3 HUN       Hungary  2012    1357253.
4 EST       Estonia  2002    33372.
5 URY       Uruguay  1998    24602.
6 PRT       Portugal 2000    378849.
7 ISL       Iceland  1999    79556.
8 LUX       Luxembourg 2011   316044.
9 MEX       Mexico   2015    2450078.
10 IRL      Ireland  2013   1825961.
# ... with 90 more rows
```

# **How to modify them?**

# select

**Only the columns you need, in the order you want**

df

	C1	C2	C3
1			
2			
3			
4			



	C1	C3
1		
2		
3		
4		

# select

**Only the columns you need, in the order you want**

df

	C1	C2	C3
1			
2			
3			
4			



	C1	C3
1		
2		
3		
4		

`select(df, C1, C2)`

# filter

Only the rows you need

df

	C1	C2	C3
1	DOG		
2	CAT		
3	BIRD		
4	DOG		



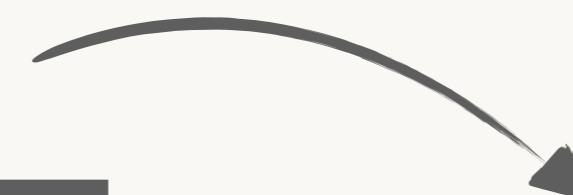
	C1	C2	C3
1	DOG		
4	DOG		

# filter

Only the rows you need

df

	C1	C2	C3
1	DOG		
2	CAT		
3	BIRD		
4	DOG		



	C1	C2	C3
1	DOG		
4	DOG		

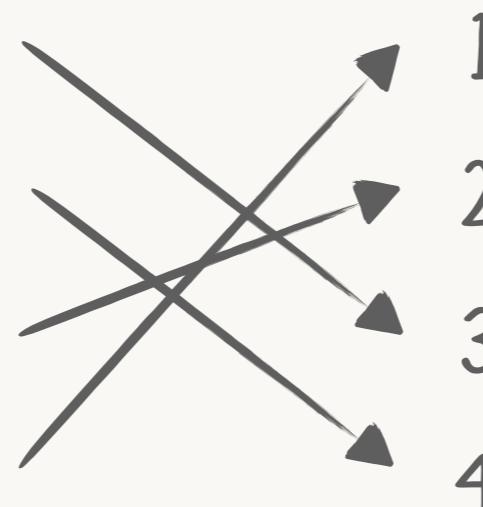
`filter(df, C1 == "DOG")`

# arrange

Order the rows according to values

df

	C1	C2	C3
1	JANE		21
2	JO		41
3	B0B		12
4	LIZ		3



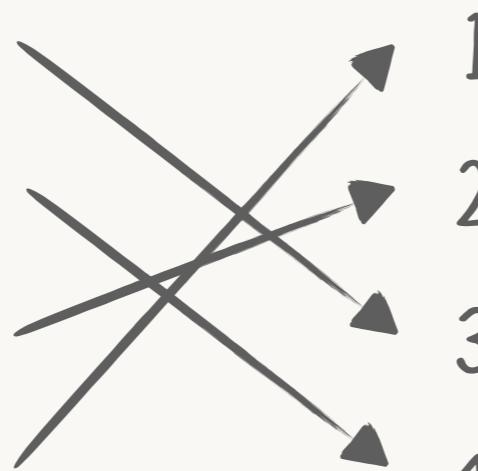
	C1	C2	C3
1	LIZ		3
2	B0B		12
3	JANE		21
4	JO		41

# arrange

Order the rows according to values

df

	C1	C2	C3
1	JANE		21
2	JO		41
3	B0B		12
4	LIZ		3



	C1	C2	C3
1	LIZ		3
2	B0B		12
3	JANE		21
4	JO		41

arrange(df, C3)

# mutate

## Create new columns

df

	C1	C2
1	10	
2	20	
3	30	
4	40	



	C1	C2	C3
1	10		20
2	20		40
3	30		60
4	40		80

# mutate

## Create new columns

df

	C1	C2
1	10	
2	20	
3	30	
4	40	



	C1	C2	C3
1	10		20
2	20		40
3	30		60
4	40		80



```
mutate(df, C3 = C1*2)
```



# Dplyr 101

## Exercise

**<http://bit.do/exts-dplyr-1>**

# The pipe sign

% > %

Let's escape the nested functions *NIGHTMARE!*

**% > %**



%>%

Without %>%

f(x)

h( g( f(x) ) )

With %>%

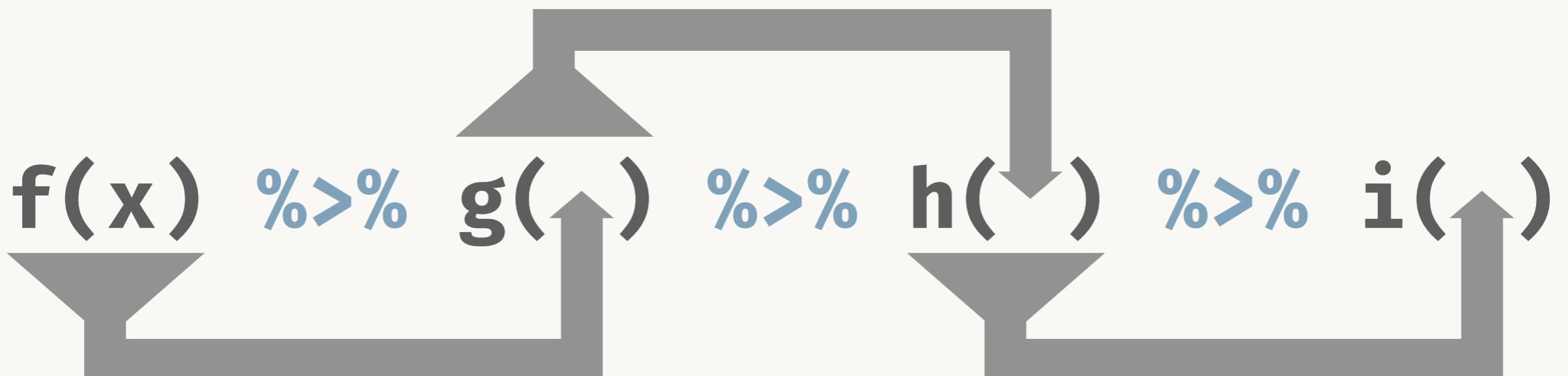
x %>% f

f(x) %>% g %>% h

% > %

f(x) %>% g( ) %>% h( ) %>% i( )

% > %



# select Only the columns you need

df

	C1	C2	C3
1			
2			
3			
4			



	C1	C3
1		
2		
3		
4		

`select(df, C1, C2)`

*OR*

`df %>% select(C1, C2)`

# filter Only the rows you need

df

	C1	C2	C3
1	DOG		
2	CAT		
3	BIRD		
4	DOG		

1      1  
2      4

	C1	C2	C3
1	DOG		
4	DOG		

`filter(df, C1 == "DOG")`

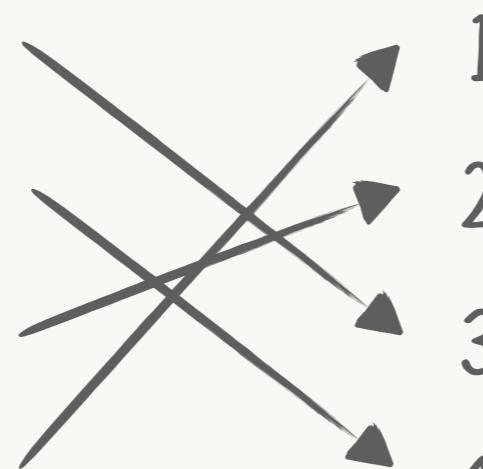
*OR*

`df %>% filter(C1 == "DOG")`

# arrange Order the rows according to values

df

	C1	C2	C3
1	JANE		21
2	JO		41
3	BOB		12
4	LIZ		3



	C1	C2	C3
1	LIZ		3
2	BOB		12
3	JANE		21
4	JO		41

arrange(df, C3)

OR

df %>% arrange(C3)

# mutate Create new columns

df

	C1	C2	C3
1	10		20
2	20		40
3	30		60
4	40		80

```
mutate(df, C3 = C1*2)
```

OR

```
df %>% mutate(C3 = C1*2)
```

# Dplyr 101

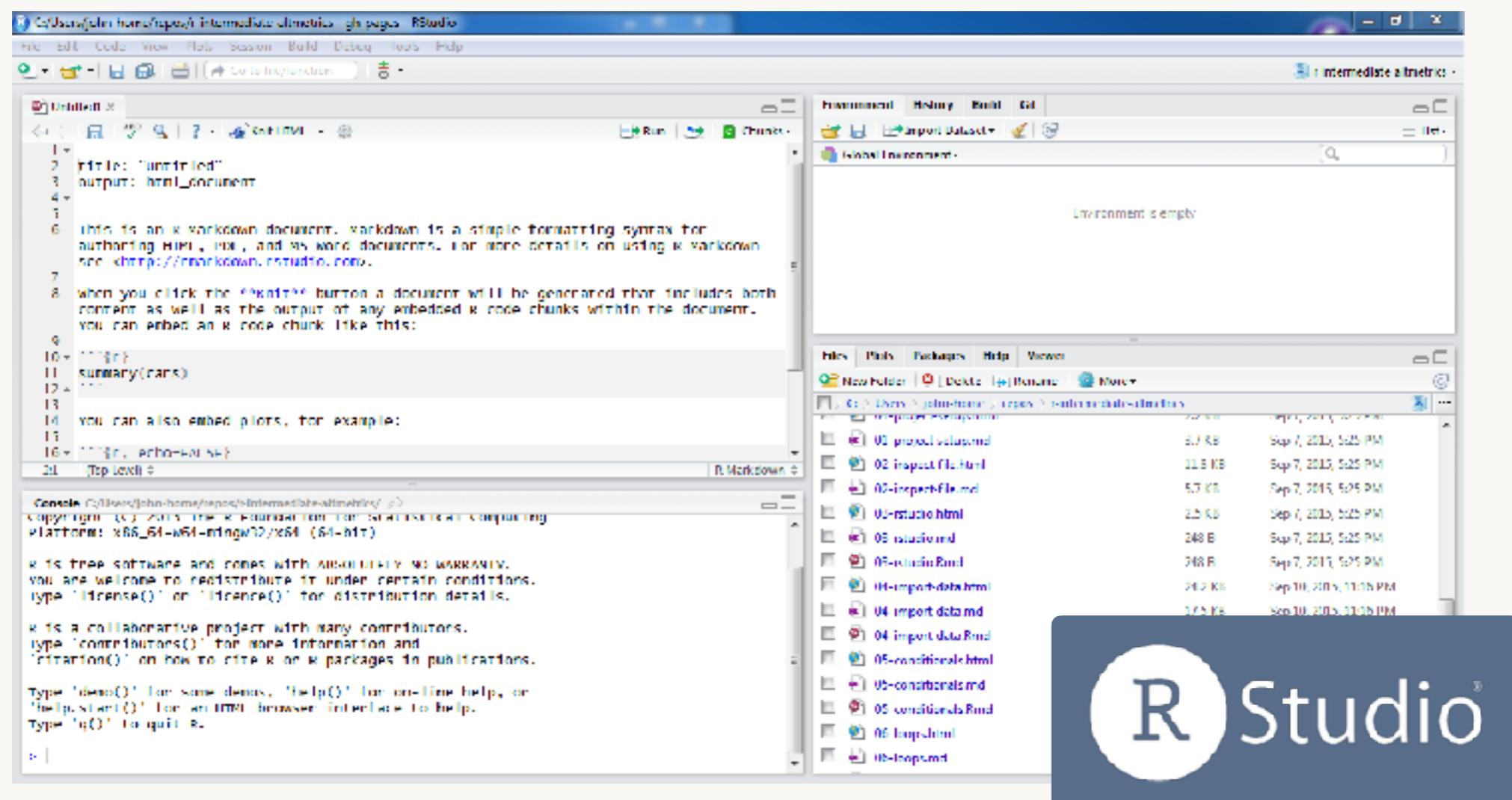
## Exercise

**<http://bit.do/exts-dplyr-1>**

# RStudio Walkthrough

## Our R editor

# RStudio



## WHERE WE TYPE R CODE

The screenshot shows the RStudio interface. On the left, a code editor window displays a Markdown file named 'Untitled.Rmd'. The code includes R code chunks and a summary of R Markdown features. On the right, a 'Console' window shows the R environment and a list of files in the current directory.

```
1: <!-- This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.
```

```
2: title: "Untitled"
3: output: html_document
```

```
4:
```

```
5:
```

```
6: this is an R Markdown document. markdown is a simple formatting syntax for
authoring HTML, PDF, and MS Word documents. For more details on using R Markdown
see http://rmarkdown.rstudio.com.
```

```
7:
```

```
8: when you click the KnitHTML button a document will be generated that includes both
content as well as the output of any embedded R code chunks within the document.
you can embed an R code chunk like this:
```

```
9:
```

```
10: ````{r}
11: summary(cars)
12: ````
```

```
13:
```

```
14: you can also embed plots, for example:
15:
16: ````{r, echo=FALSE}
```

```
17: mtcars %>%
```

```
18:   ggplot(aes(mpg, wt)) +
19:     geom_point()
```

```
20:
```

```
21: ````
```

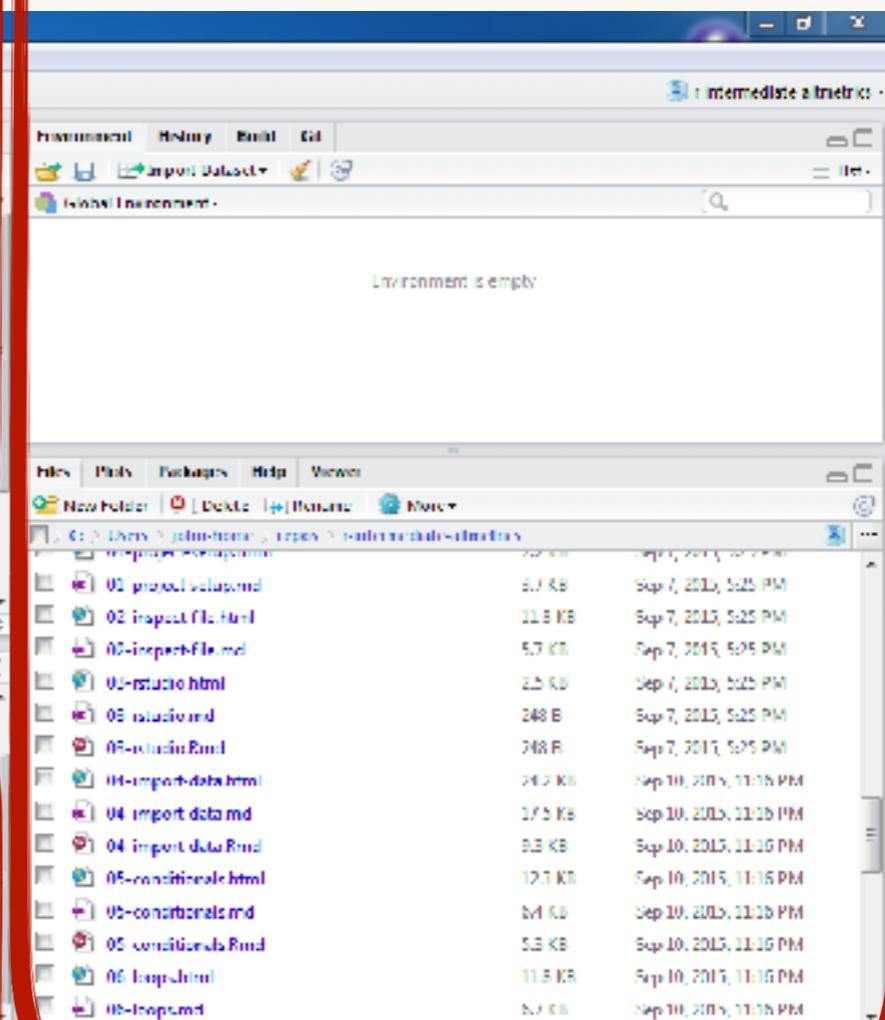
```
22:
```

```
23: >
```

```
24: > #> [1] "This is free software and comes with ABSOLUTELY NO WARRANTY."
25: > #> You are welcome to redistribute it under certain conditions.
26: > #> type 'license()' or 'licence()' for distribution details.
27: >
28: > #> R is a collaborative project with many contributors.
29: > #> type 'contributors()' for more information and
30: > #> 'citation()' on how to cite R or R packages in publications.
31: >
32: > Type 'demo()' for some demos, 'help()' for on-line help, or
33: > 'help.start()' for an HTML browser interface to help.
34: > Type 'q()' to quit R.
```

```
35: >
```

## HELPERS

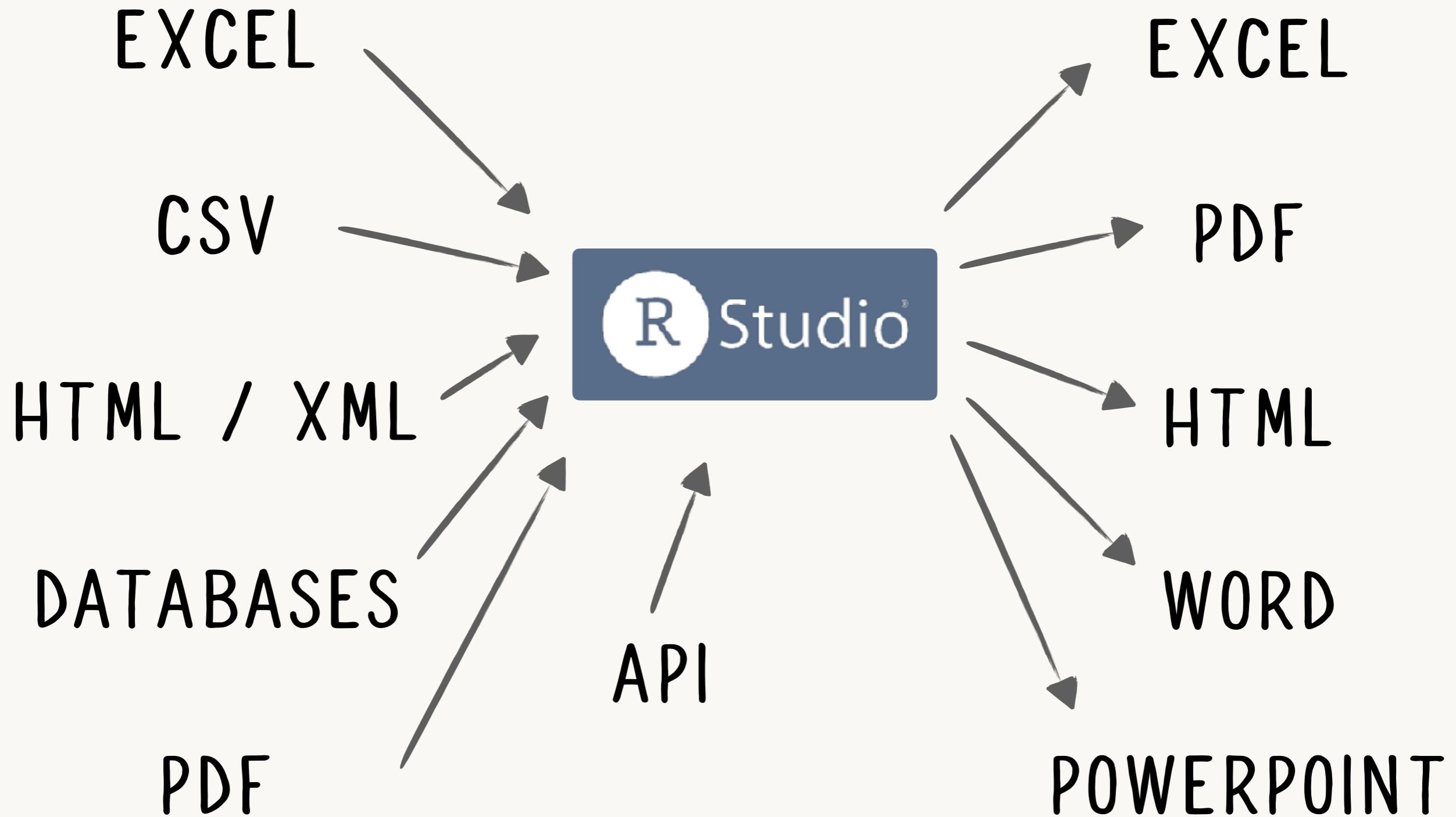


## WHERE WE EXECUTE R CODE

# Inputs and Outputs

How to get things in? What to get out?

# Inputs



# Outputs

THERE IS AN ~~APP~~ FOR THAT  
PACKAGE



CSV



EXCEL



HTML

# Inputs

✓ EXCEL

✓ CSV

HTML / XML

DATABASES

PDF



# Outputs

EXCEL

PDF

HTML

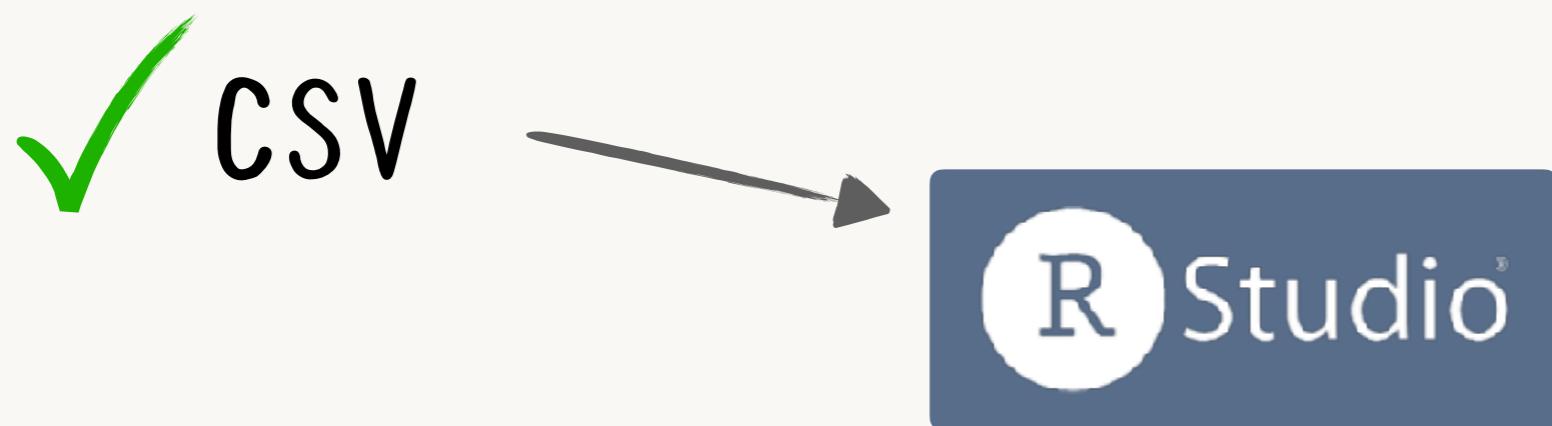
WORD

POWERPOINT

API

# Inputs

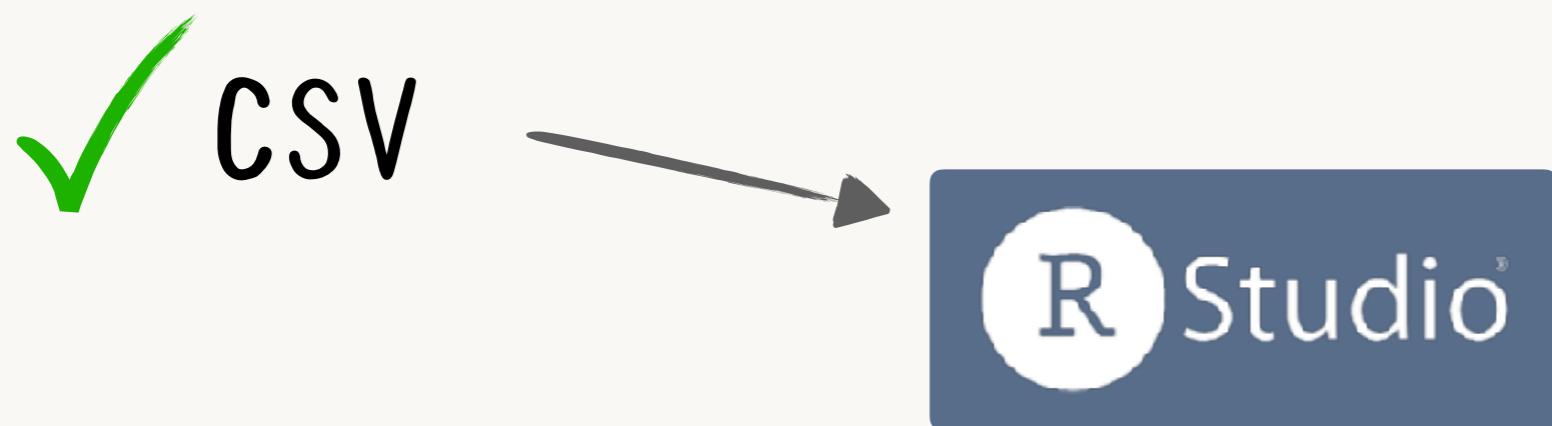
Import CSV file in R



```
cancer_data <- readr::read_csv(  
  "data/breast-cancer.csv")
```

# Inputs

Import CSV file in R



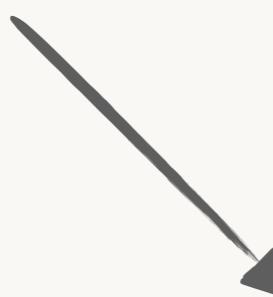
```
cancer_data <- readr::read_csv(  
  "data/breast-cancer.csv")
```

FILE LOCATION

# Inputs

✓ EXCEL

Import Excel file in R



```
oil_data_raw <- readxl::read_xls(  
  "oil-price.xls", sheet = "Data 1", skip = 2)
```

# Inputs

✓ EXCEL

Import Excel file in R



```
oil_data_raw <- readxl::read_xls(  
  "oil-price.xls", sheet = "Data 1", skip = 2)
```

FILE LOCATION

DATA LOCATION IN FILE

# Inputs

## Scrape webpage in R

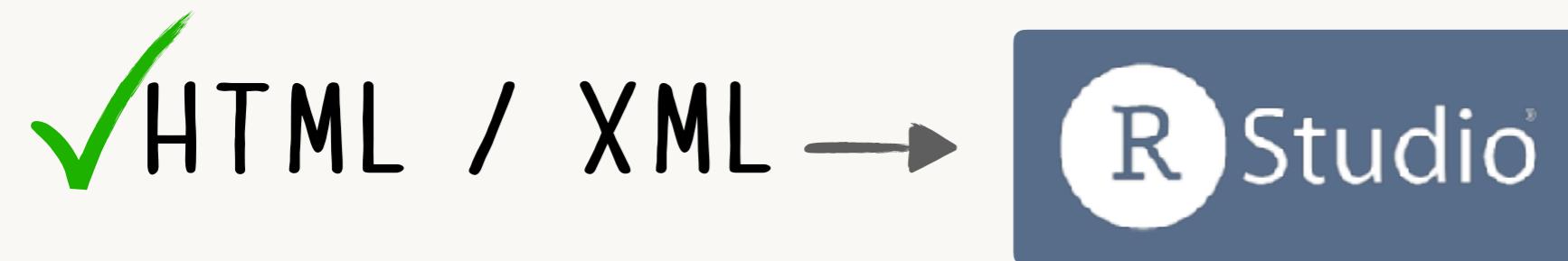
✓ HTML / XML →



```
url <- "https://en.wikipedia.org/wiki/Cantons_of_Switzerland"  
  
density <- url %>%  
  
xml2::read_html() %>%  
  
rvest::html_node(  
  
  xpath='//table[contains(@class,"wikitable")]') %>%  
  
rvest::html_table()
```

# Inputs

## Scrape webpage in R

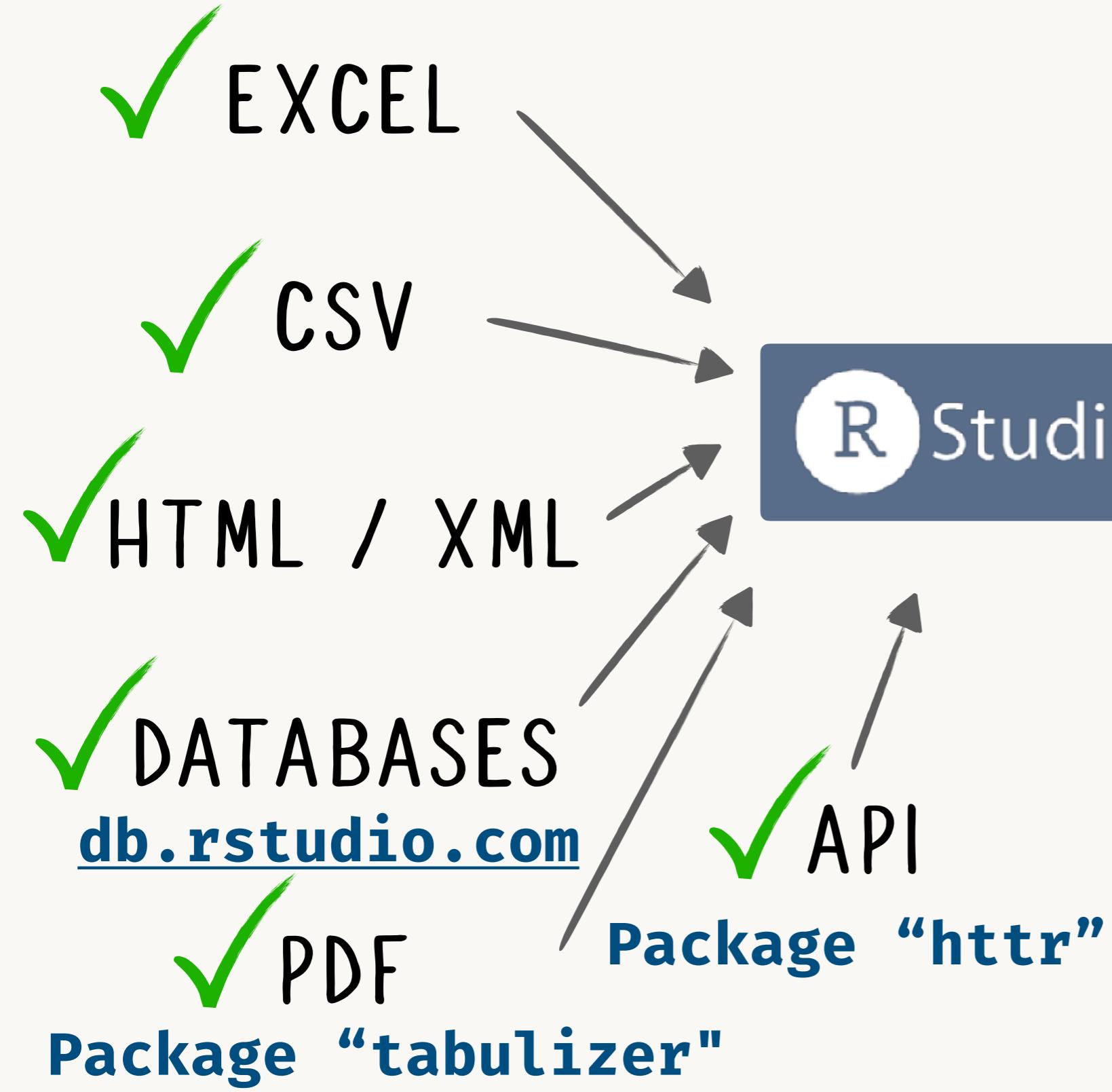


```
url <- "https://en.wikipedia.org/wiki/Cantons_of_Switzerland"  
density <- url %>%  
xml2::read_html() %>%  
rvest::html_node(  
  xpath='//table[contains(@class,"wikitable")]') %>%  
rvest::html_table()
```

The code is annotated with descriptions:

- density <- url %>%  
xml2::read\_html() %>%** } IMPORT THE HTML FROM THE URL
- rvest::html\_node(** FILTER TO KEEP ONLY THE HTML TABLE
- xpath='//table[contains(@class,"wikitable")]'** ) %>%
- rvest::html\_table()** CONVERT HTML TABLE TO DATAFRAME

# Inputs



# Outputs

# Save to excel

# Outputs



```
openxlsx::write.xlsx(
```

```
  cancer_data, "cancer_excel.xlsx", asTable = TRUE)
```

# Save to excel

# Outputs

EXCEL ✓



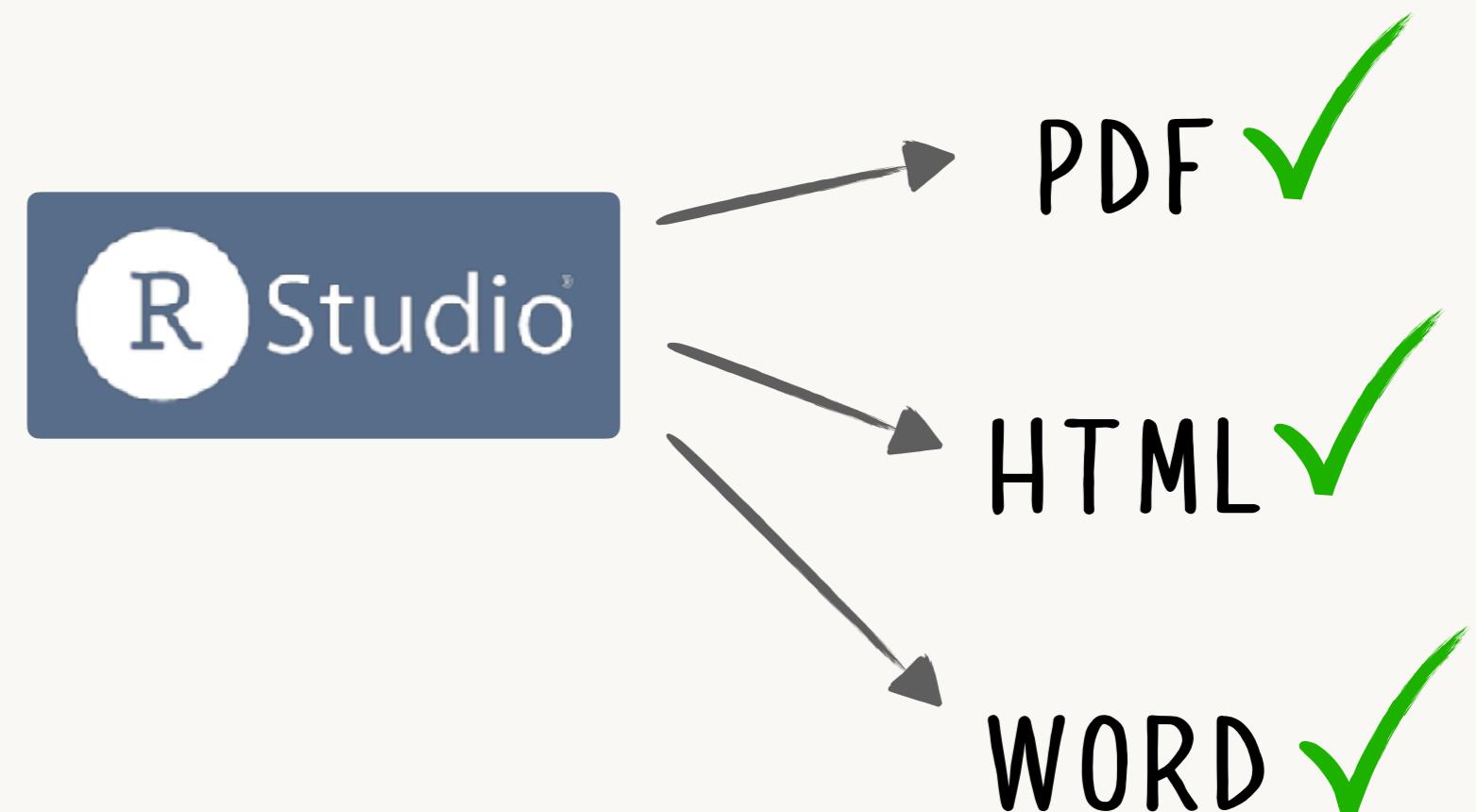
```
openxlsx::write.xlsx( FILE NAME TO CREATE  
cancer_data, "cancer_excel.xlsx", asTable = TRUE)
```

A DATAFRAME

A BIT OF FORMATTING  
FOR MORE SEE HUXTABLE PACKAGE

# How to create reports

## Outputs



Inputs

Outputs

# Rmarkdown

# demo and exercise

PDF

POWERPOINT

CSV

HTML

DATABASE

XMLE

PDF

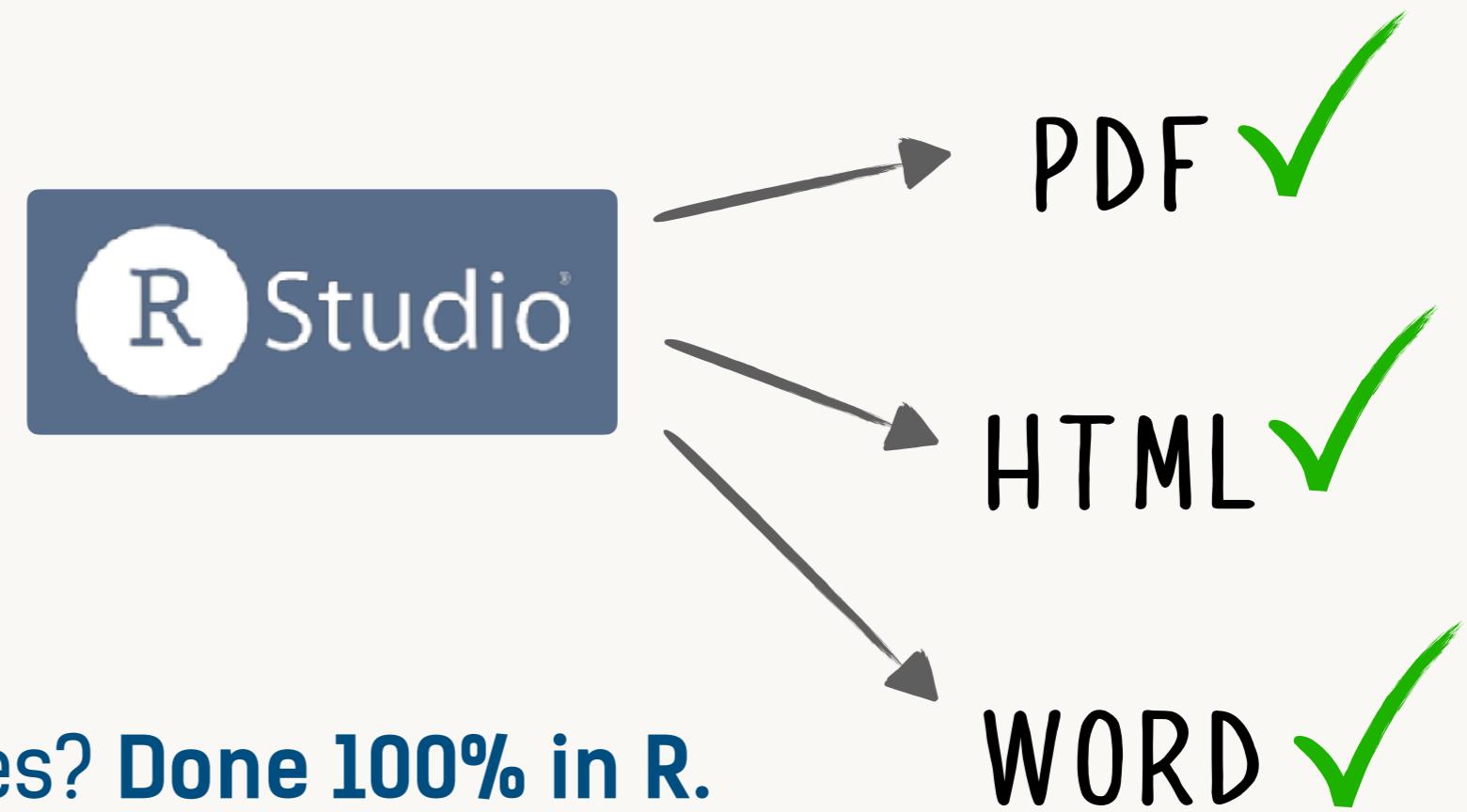
HTML

WORD

POWERPOINT

# How to create reports

# Outputs



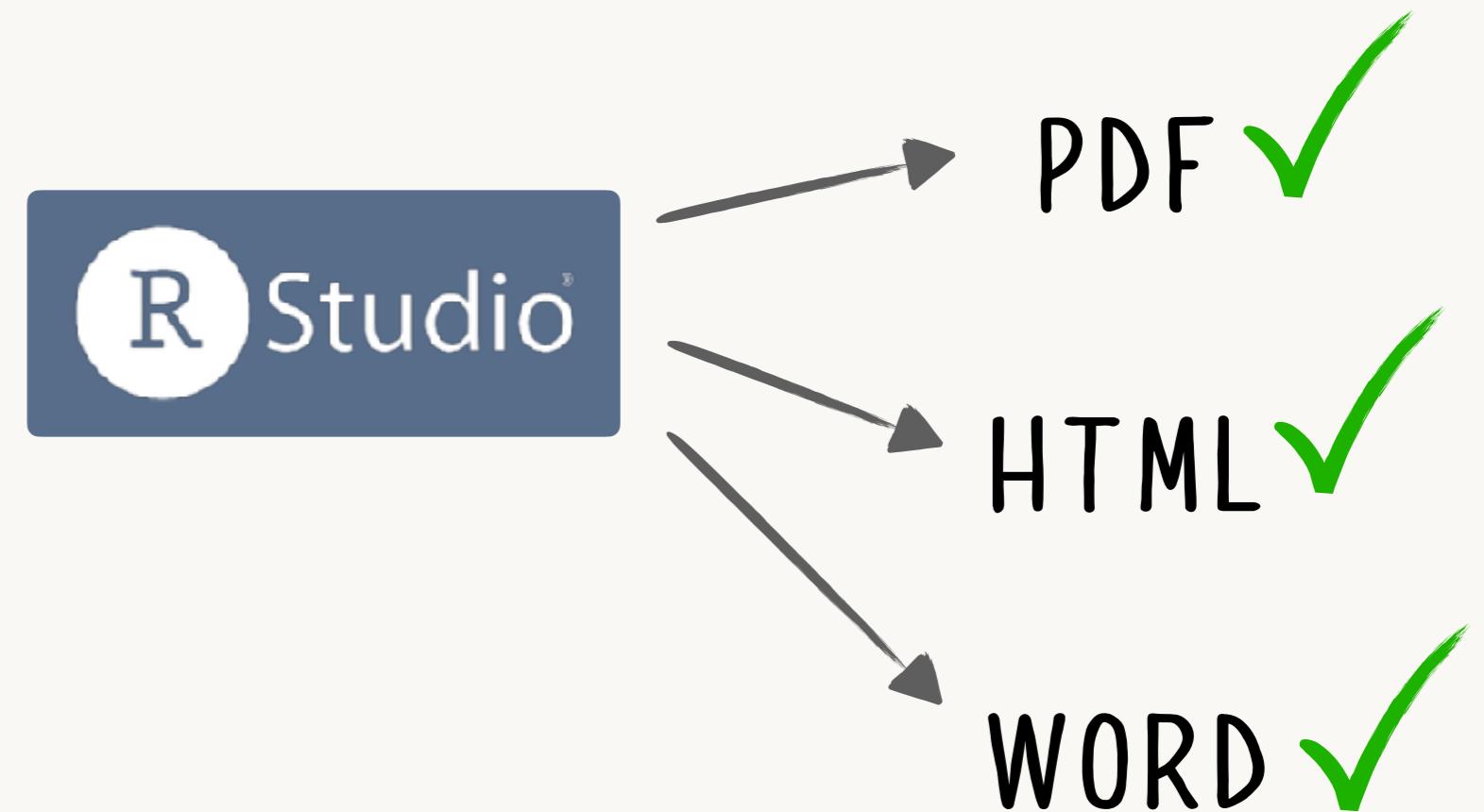
Today's exercises websites? Done 100% in R.

My website ([xvrdm.github.io](https://xvrdm.github.io))? Done 100% in R.

This slide deck (<http://bit.do/ggslides>)? Done 100% in R.

# How to create reports

## Outputs



More examples and code at

<https://rmarkdown.rstudio.com/gallery.html>

# Inputs

- ✓ EXCEL
- ✓ CSV
- ✓ HTML / XML
- ✓ DATABASES
- ✓ PDF
- ✓ API



# Outputs

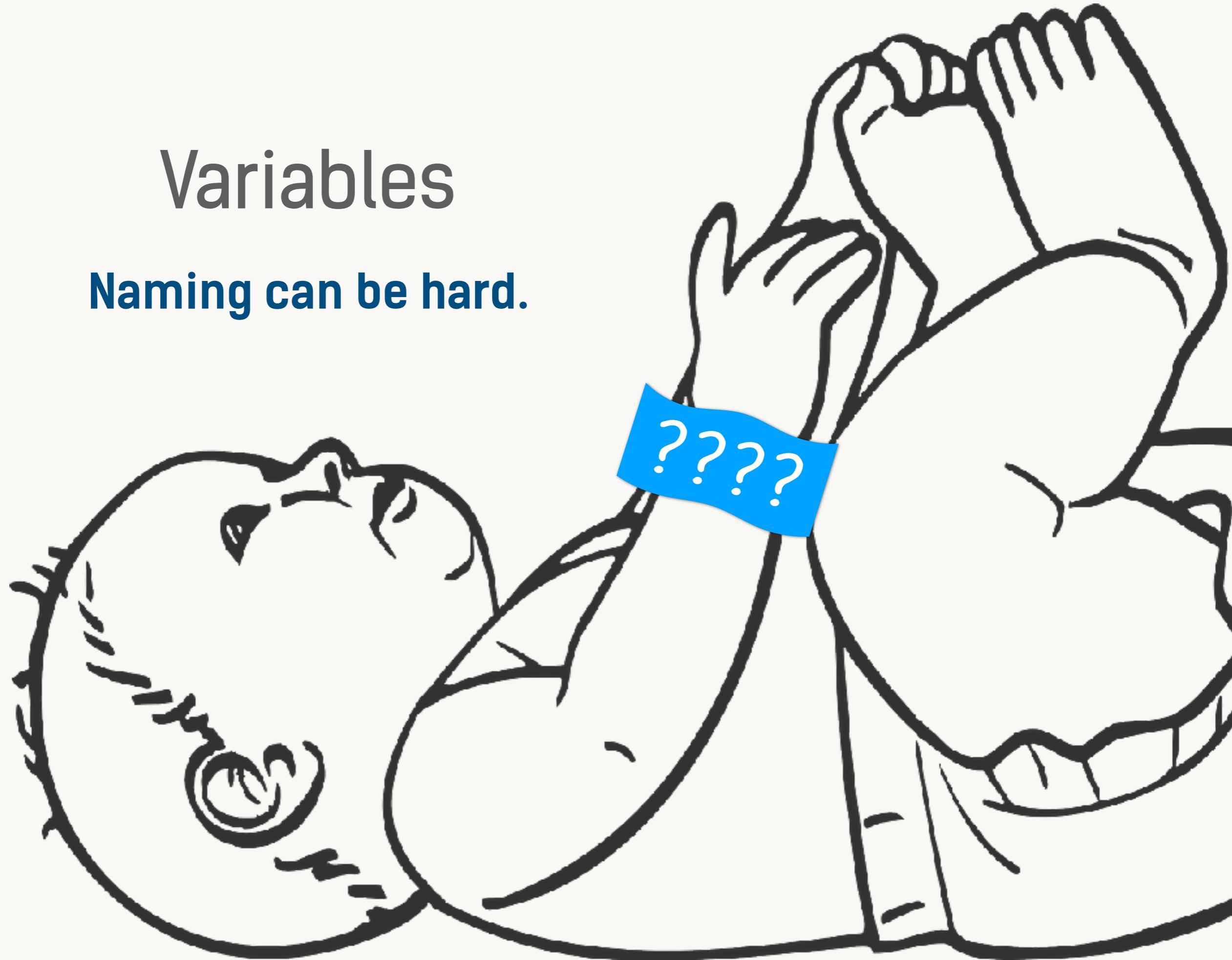
- EXCEL ✓
  - PDF ✓
  - HTML ✓
  - WORD ✓
  - POWERPOINT ✓
- Package "officer"

# A recap on variables

Sometimes R gives too much freedom

# Variables

Naming can be hard.



# Variables

Avoid “-” and “ ” in your variable names.

**my\_variable** ✓ YES

**my-variable** ✗ NO

**my variable** ✗ NO

# Variables

Rather than being forbidden, “-” and “ ” are allowed in variable name if you surround with the confusing ` sign

`my_variable` ✓ YES

``my-variable`` ~ MEH

``my variable`` ~ MEH

# Advanced dplyr functions

# group\_by

Make your dataframes “group-aware”



	CLIENT	C
1	B0B	
2	B0B	
3	ALICE	
4	ALICE	
5	JEAN	

	CLIENT	C	GROUP
1	B0B		B0B
2	B0B		
3	ALICE		ALICE
4	ALICE		
5	JEAN		G. JEAN

**group\_by(CLIENT)**

# group\_by Make your dataframes “group-aware”

df

The diagram illustrates the transformation of a DataFrame 'df' into a grouped DataFrame. On the left, 'df' is shown as a 5x2 grid with columns 'CLIENT' and 'C2'. Rows are indexed from 1 to 5, with values 'B0B', 'B0B', 'ALICE', 'ALICE', and 'JEAN' respectively. An arrow points from 'df' to the grouped DataFrame on the right. The grouped DataFrame has the same structure but includes a 'GROUP' column on the far right. The first two rows ('B0B') are grouped together under 'GROUP B0B'. The next two rows ('ALICE') are grouped together under 'GROUP ALICE'. The fifth row ('JEAN') is listed separately under 'G. JEAN'. Red boxes highlight the 'CLIENT' and 'C2' columns for each group, and red lines separate the groups.

	CLIENT	C2				
1	B0B		1	B0B		GROUP B0B
2	B0B		2	B0B		
3	ALICE		3	ALICE		GROUP ALICE
4	ALICE		4	ALICE		
5	JEAN		5	JEAN		G. JEAN

group\_by(df, CLIENT)

OR

df %>% group\_by(CLIENT)

# summarise

## Summarise dataframe (by groups or not)

	CLIENT	C
1	B0B	1
2	B0B	1
3	ALICE	1
4	ALICE	1
5	JEAN	1



GROUP

B0B

GROUP

ALICE

G. JEAN

CLIENT	SUM(C)
B0B	2
ALICE	2
JEAN	1

`summarise(sum(c))`

# summarise Summarise dataframe

df

	CLIENT	C
1	B0B	1
2	B0B	1
3	ALICE	1
4	ALICE	1
5	JEAN	1

GROUP  
B0B  
GROUP  
ALICE  
G. JEAN



CLIENT	SUM(C)
B0B	2
ALICE	2
JEAN	1

`summarise(df, sum(C))`

*OR*

`df %>% summarise(sum(C))`

# dplyr joins

X	Y	Z	
1			+
2			=
3			

X	A
1	
2	
3	

X	Y	Z	A
1			
2			
3			

A column in common



left-join



right-join

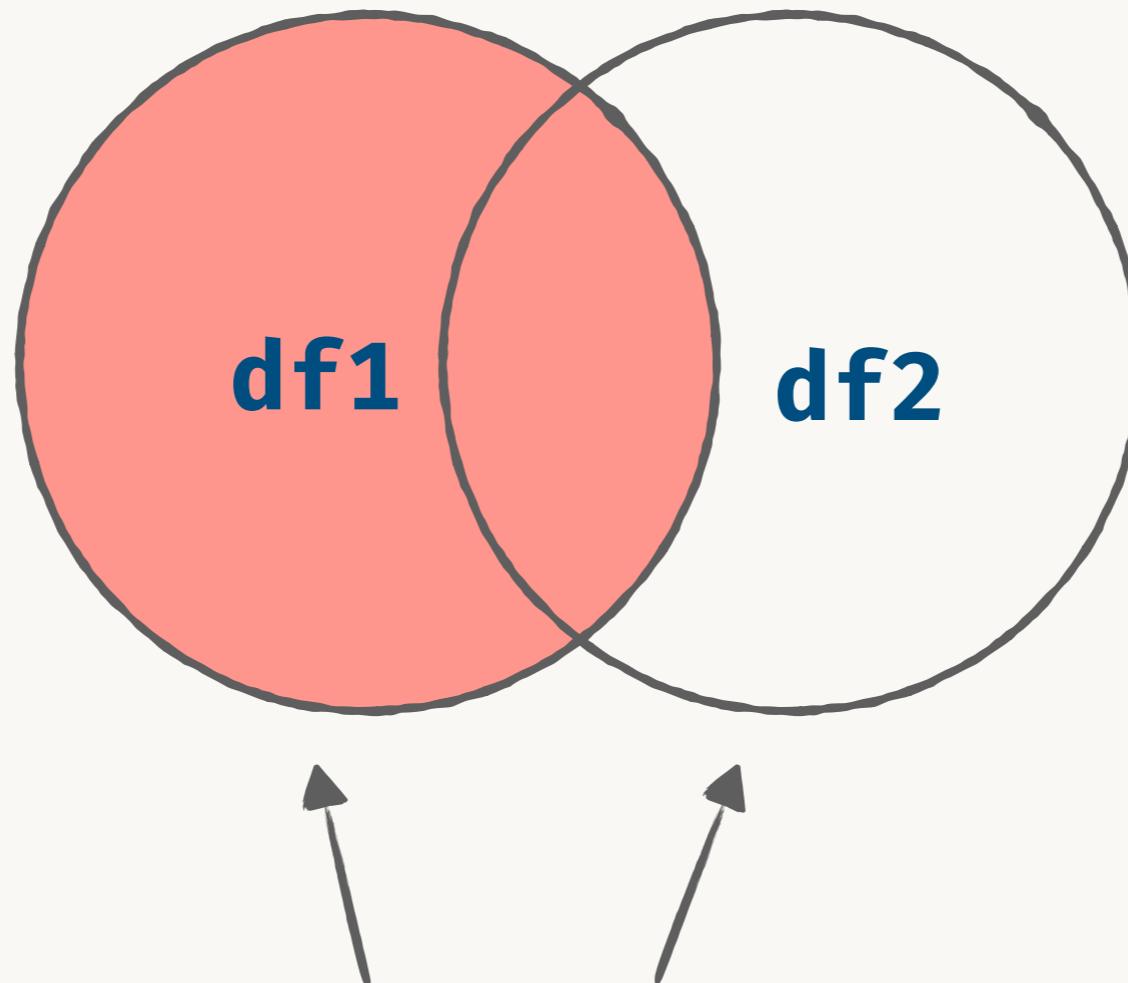


full-join

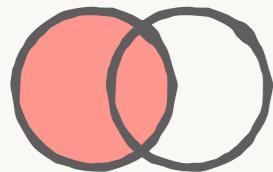


inner-join

# left-join



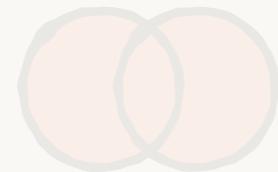
```
left_join( df1, df2, by=c("key"))
```



left-join



right-join



full-join



inner-join

# left-join

X	Y	Z	
1			
2			
3			
4			

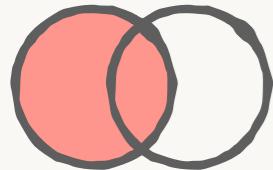
+

X	A
1	
2	
3	
5	

=

X	Y	Z	A
1			
2			
3			
4			NA

A column nearly in common



left-join



right-join



full-join



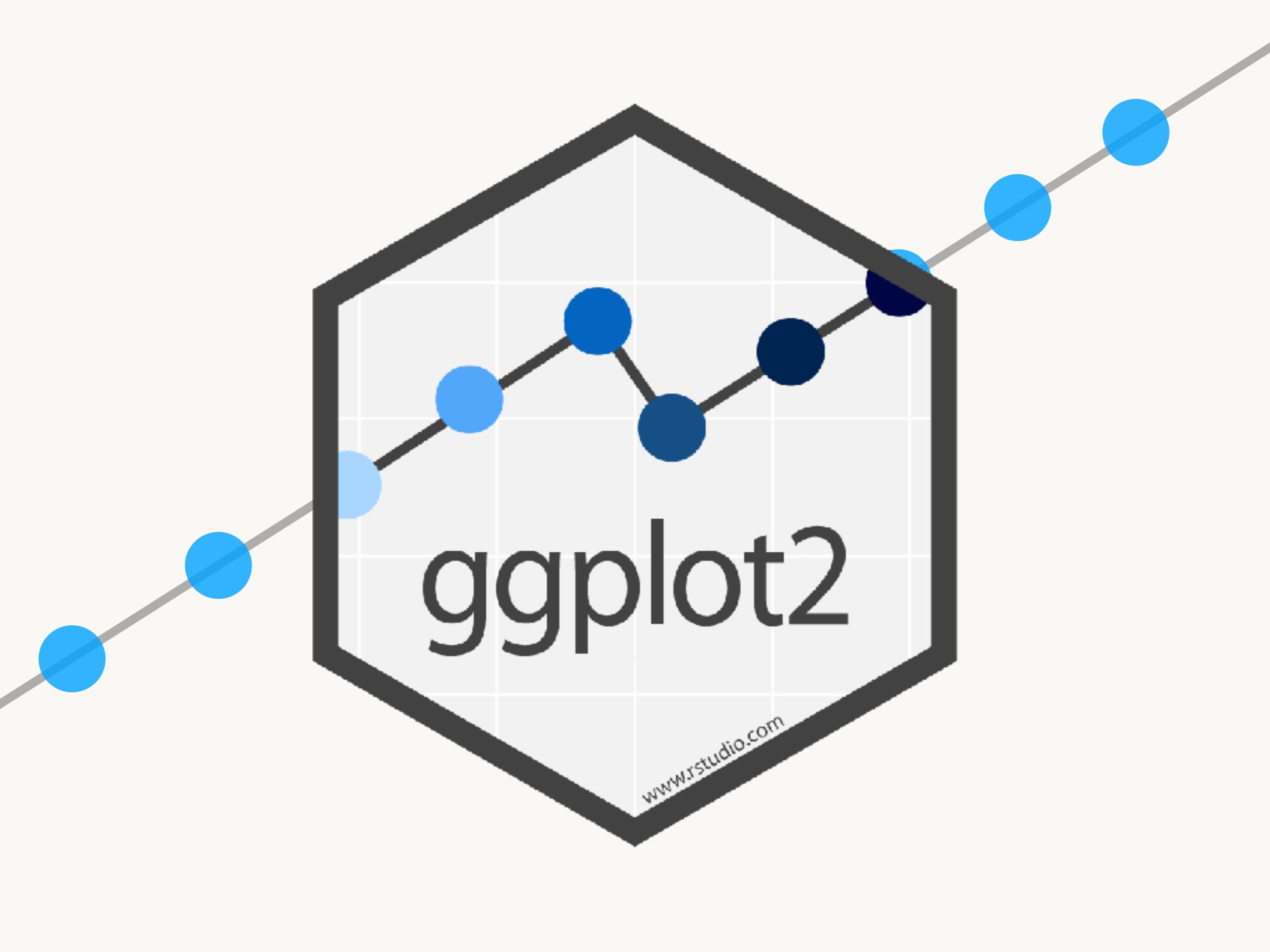
inner-join

# dplyr 201 exercise

<http://bit.do/exts-dplyr-2>

# How to do charts in R?

## An intro to the ggplot2 library



The image features the ggplot2 logo, which is a dark gray hexagon containing the word "ggplot2" in a large, bold, black sans-serif font. The hexagon is set against a light gray grid background. Overlaid on the logo is a network graph consisting of several blue circular nodes connected by black lines. Some nodes are a lighter shade of blue, while others are a darker shade, suggesting a hierarchy or different states. The network is positioned such that it partially obscures the text "ggplot2".

www.rstudio.com



`ggplot(...)`



WHAT DATA WILL BE  
BOUND TO THE CHART

+

`geom_*`



WHAT CHART TYPE(S)  
WE WANT (LINES, BARS...)

+

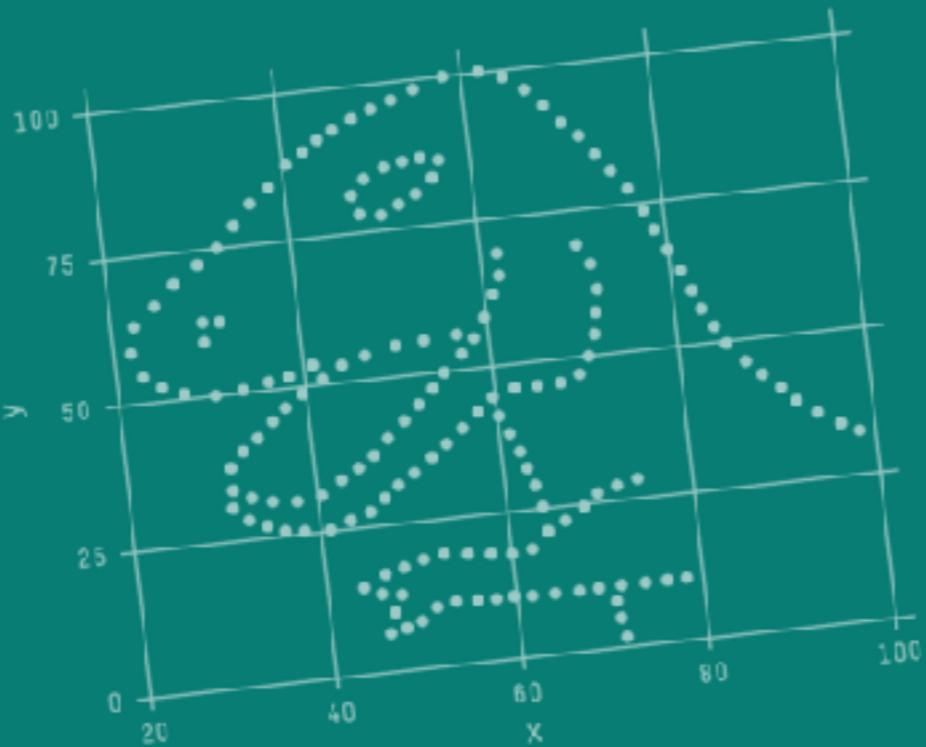
`theme()`



HOW TO CUSTOMIZE  
THE LOOK OF THE CHART

# THE HITCHHIKER'S GUIDE TO GGPLOT2

Don't panic and create beautiful plots with R



Jodie Burchell  
Mauricio Vargas

GREAT BOOK.  
PAY AS MUCH AS YOU  
WANT FOR THE  
“BOOK ONLY” VERSION.

<http://bit.do/ggplot-book>

THE HITCHHIKER'S  
GUIDE TO GGPlot2

Don't panic and create beautiful plots with R

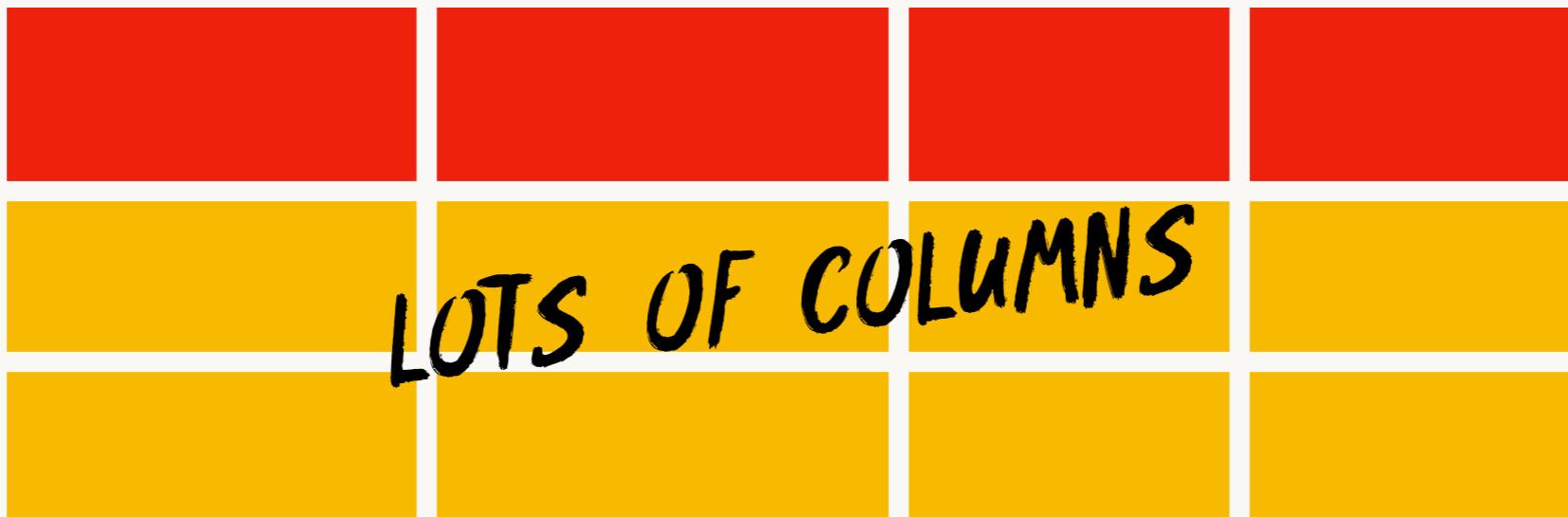
# ggplot2 demo and exercise

<http://bit.do/ggplot-book>

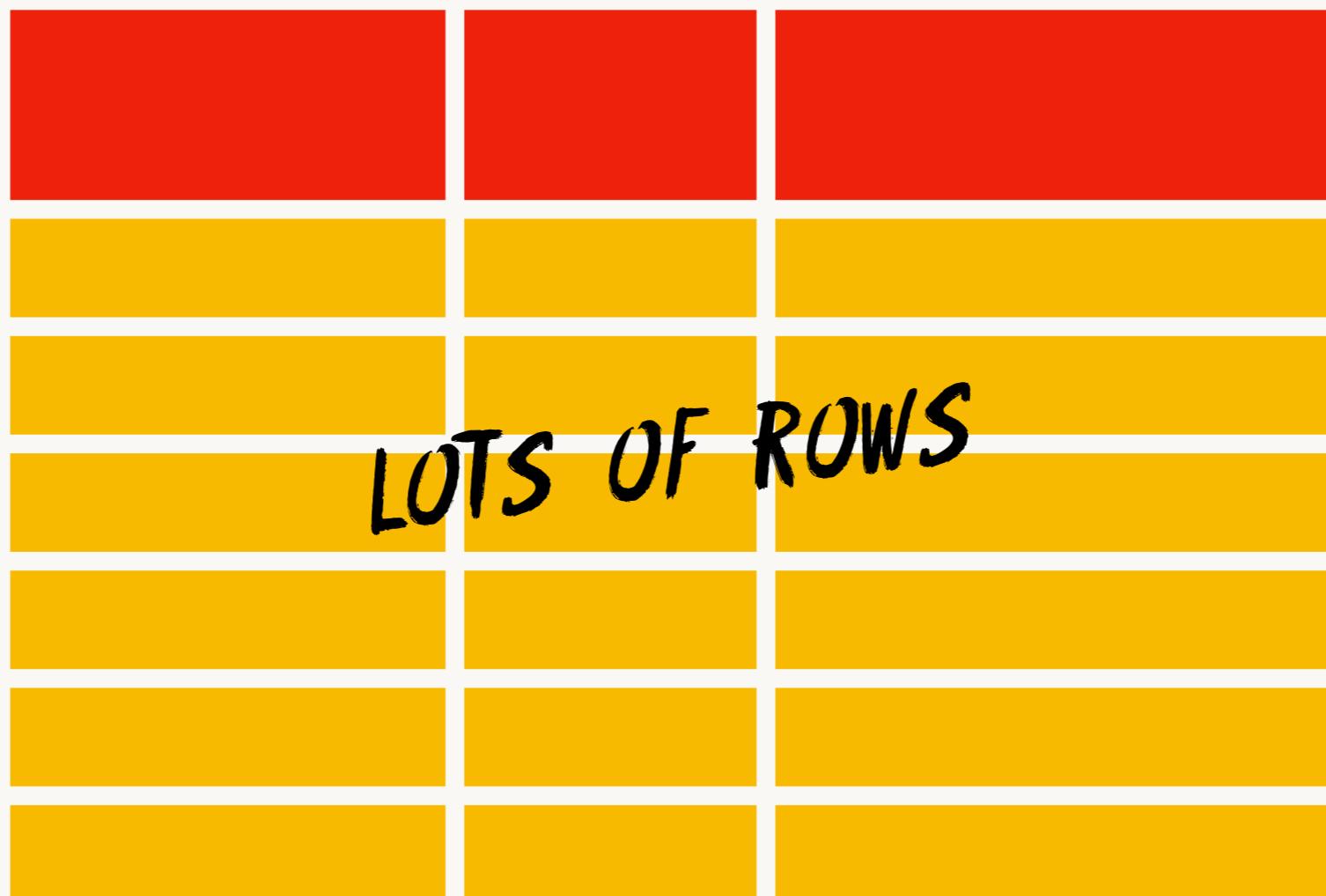
# How to reshape in R

## Pivot-table style "column to rows"

## HORIZONTAL DATA



## VERTICAL DATA



## HORIZONTAL DATA

NAME	STREET	CITY	POSTCODE
JULIA SCHMITZ	CASA POSRCLAS 64	ECUBLENS	1224
DAMIEN DOUFFET	VIA ALTISIO 108	SPIRINGEN	6464

## VERTICAL DATA

NAME	ADDRESS	VALUE
JULIA SCHMITZ	STREET	CASA POSRCLAS 64
JULIA SCHMITZ	CITY	ECUBLENS
JULIA SCHMITZ	POSTCODE	1224
DAMIEN DOUFFET	STREET	VIA ALTISIO 108
DAMIEN DOUFFET	CITY	SPIRINGEN
DAMIEN DOUFFET	POSTCODE	6464

COUNTRY	BUDGET 1900	BUDGET 1901	BUDGET 1902
FRANCE			
GERMANY			

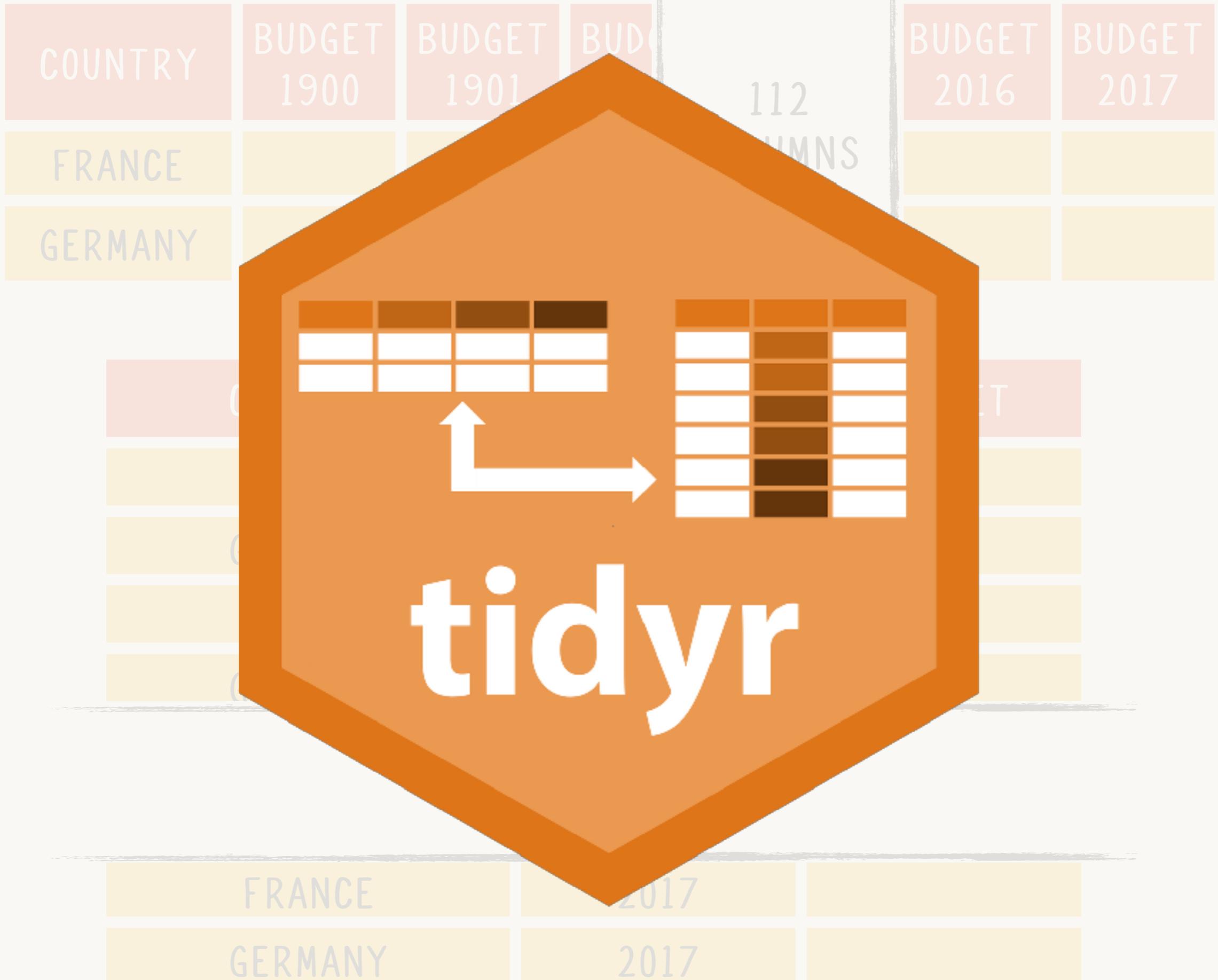
112  
COLUMNS

BUDGET 2016	BUDGET 2017

COUNTRY	YEAR	BUDGET
FRANCE	1900	
GERMANY	1900	
FRANCE	1901	
GERMANY	1901	

226 ROWS

FRANCE	2017	
GERMANY	2017	



## HORIZONTAL DATA

NAME	STREET	CITY	POSTCODE
JULIA SCHMITZ	CASA POSRCLAS 64	ECUBLENS	1224
DAMIEN DOUFFET	VIA ALTISIO 108	SPIRINGEN	6464

`spread()`

`gather()`

## VERTICAL DATA

NAME	ADDRESS	VALUE
JULIA SCHMITZ	STREET	CASA POSRCLAS 64
JULIA SCHMITZ	CITY	ECUBLENS
JULIA SCHMITZ	POSTCODE	1224
DAMIEN DOUFFET	STREET	VIA ALTISIO 108
DAMIEN DOUFFET	CITY	SPIRINGEN
DAMIEN DOUFFET	POSTCODE	6464

NAME	STREET	CITY	POSTCODE
JULIA SCHMITZ	CASA POSRCLAS 64	ECUBLENS	1224
DAMIEN DOUFFET	VIA ALTISIO 108	SPIRINGEN	6464

**gather(key=“ADDRESS”, value=“VALUE”,  
STREET, CITY, POSTCODE)**

NAME	ADDRESS	VALUE
JULIA SCHMITZ	STREET	CASA POSRCLAS 64
JULIA SCHMITZ	CITY	ECUBLENS
JULIA SCHMITZ	POSTCODE	1224
DAMIEN DOUFFET	STREET	VIA ALTISIO 108
DAMIEN DOUFFET	CITY	SPIRINGEN
DAMIEN DOUFFET	POSTCODE	6464



NAME	STREET	CITY	POSTCODE
JULIA SCHMITZ	CASA POSRCLAS 64	ECUBLENS	1224
DAMIEN DOUFFET	VIA ALTISIO 108	SPIRINGEN	6464

`spread(key=“ADDRESS”, value=“VALUE”)`

NAME	ADDRESS	VALUE
JULIA SCHMITZ	STREET	CASA POSRCLAS 64
JULIA SCHMITZ	CITY	ECUBLENS
JULIA SCHMITZ	POSTCODE	1224
DAMIEN DOUFFET	STREET	VIA ALTISIO 108
DAMIEN DOUFFET	CITY	SPIRINGEN
DAMIEN DOUFFET	POSTCODE	6464

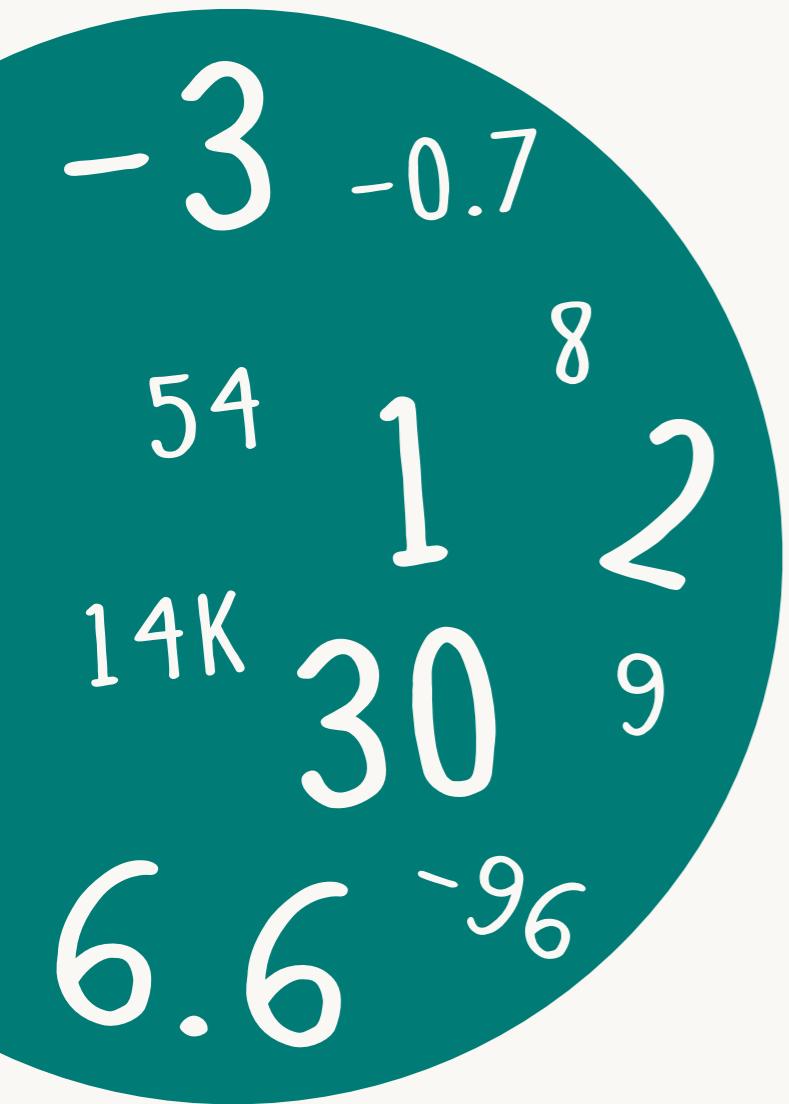
# **tidyverse**

**<http://bit.do/exts-tidyr>**

# Useful packages

For transforming strings  
and do “math” with dates

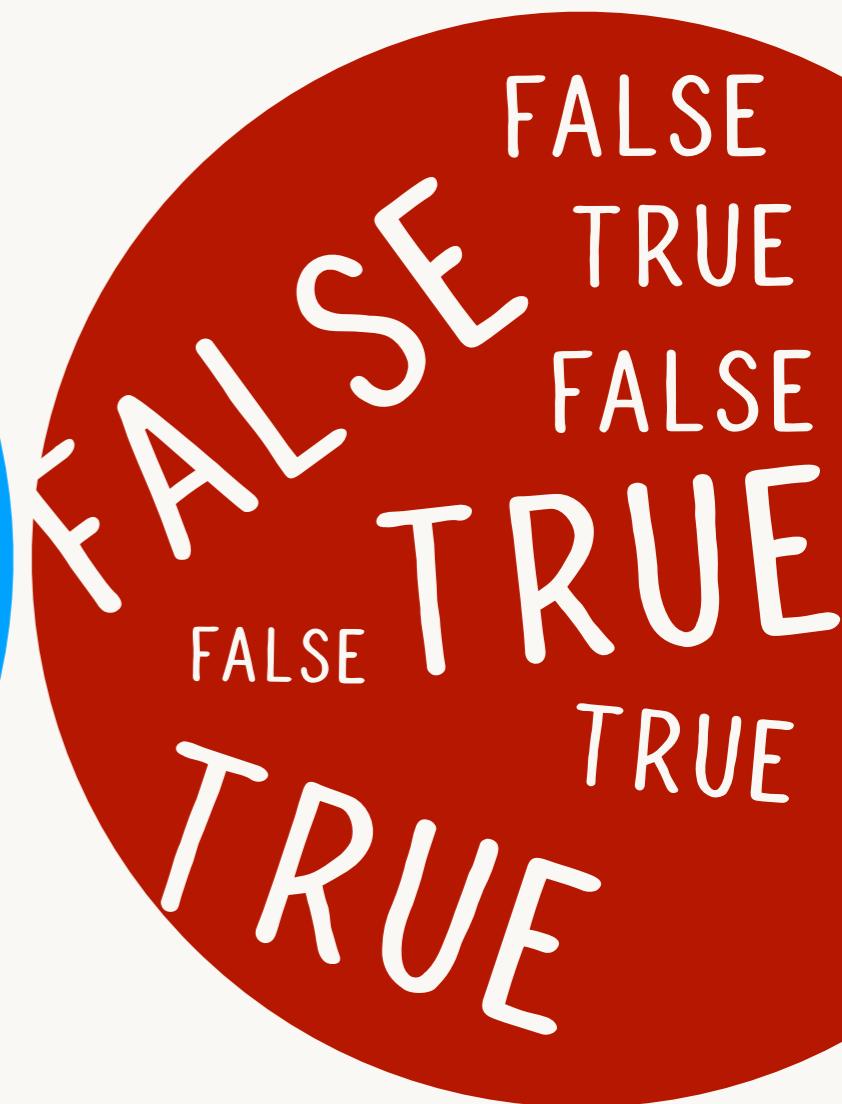
# R data classes



Numerics



Characters



Logicals

# R data classes

stringr



www.rstudio.com



Numerics

Character

Logicals



```
str_to_lower(str_sub("MY TEXT", end=2))
```

=

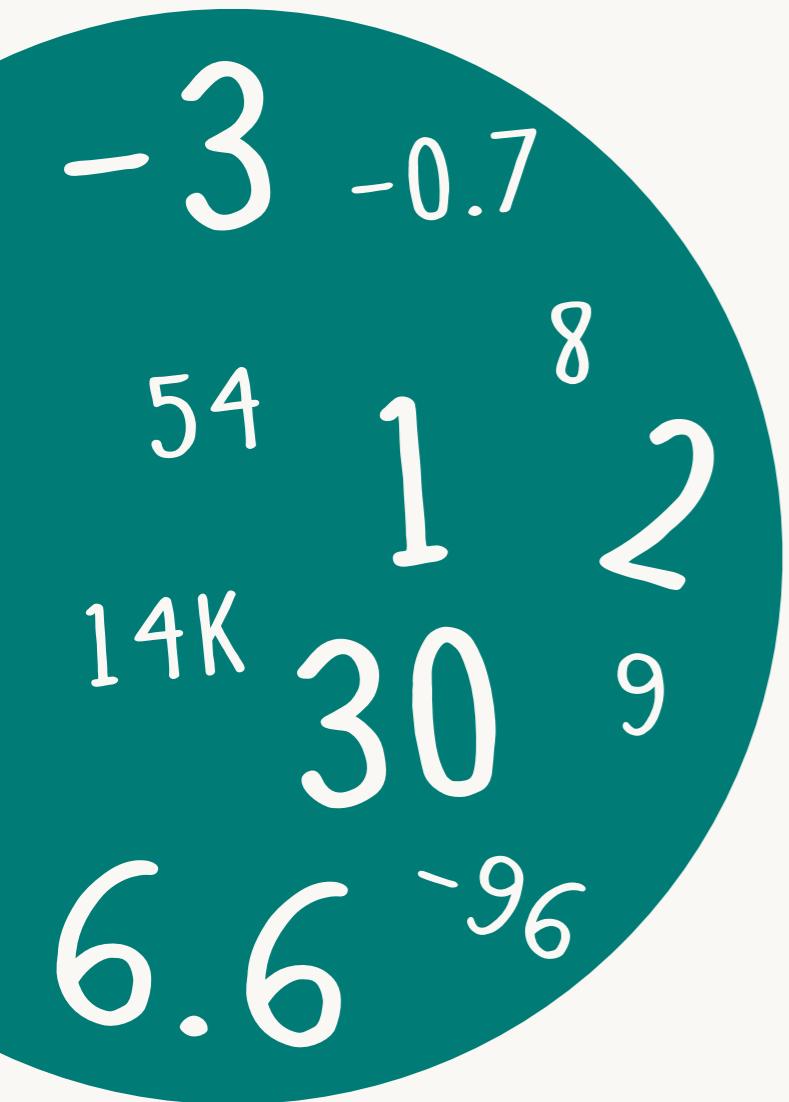
```
str_to_lower("MY TEXT") %>%  
  str_sub(end=2)
```

“my”

# String Exercise

<http://bit.do/exts-stringr>

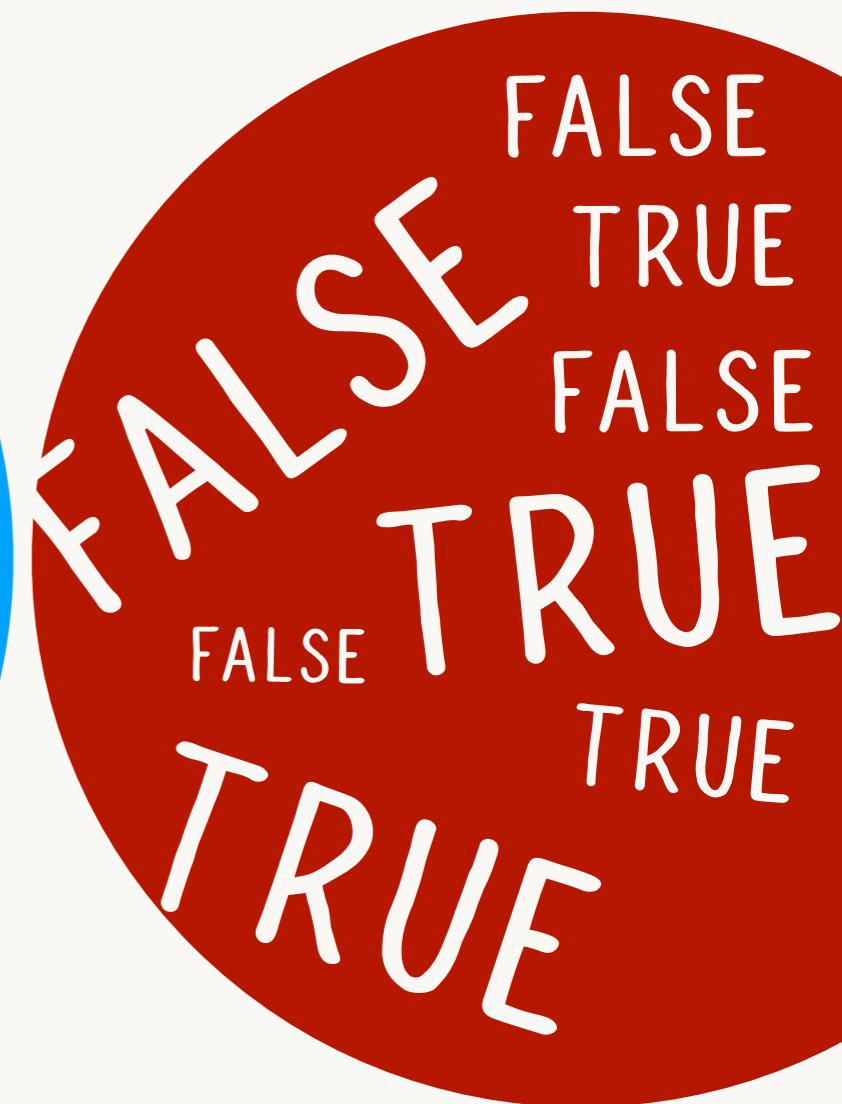
# R data classes



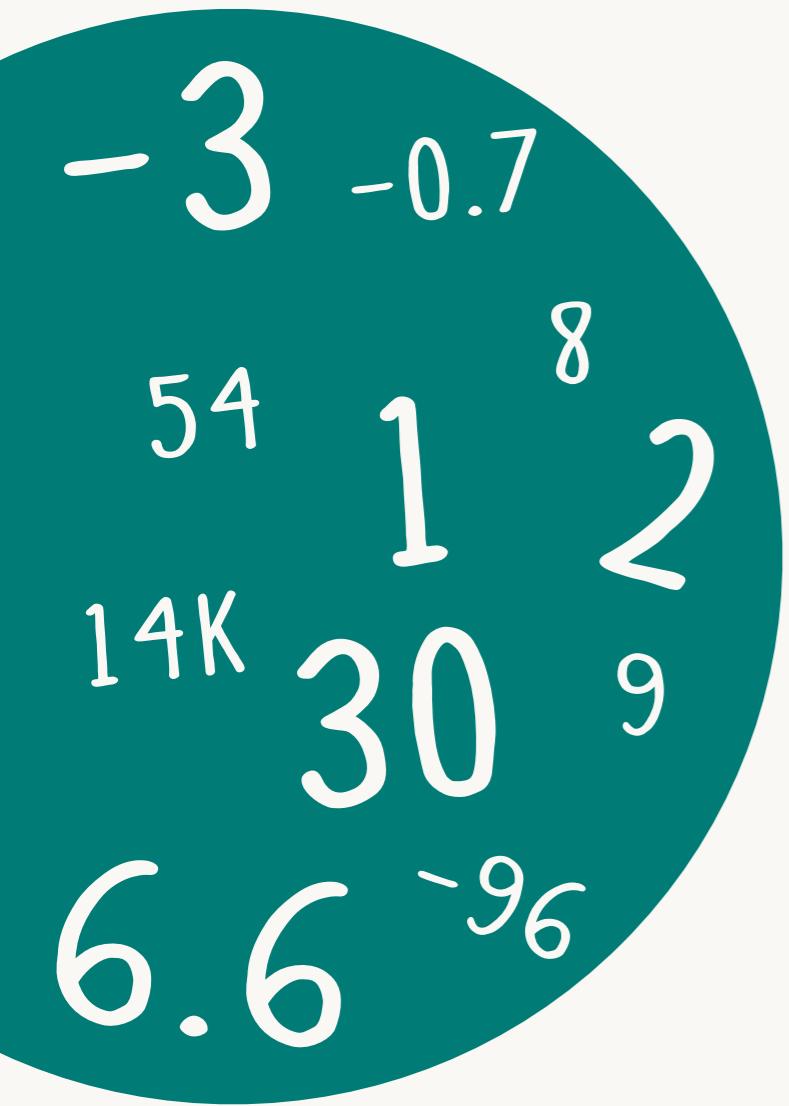
Numerics



Characters



Logicals



## Numerics

A green circle containing the following text: "23-08-13", "2018-01-12", "160601", "> MARCH", "APRIL", "THU", and "9".

## Characters

A blue circle containing the following text: "HELLO", "“SKUD”", "TOTAL", "“TEST”", "“CLIENT”", and "“CUSTOMER”".

## Dates

A red circle containing the following text: "FALSE", "TRUE", "FALSE", "TRUE", "TRUE", "TRUE", and "FALSE".

## Logicals



Numerics



Logicals

# Dates Exercise

<http://bit.do/exts-dates>

All feedback is  
super helpful

<http://bit.do/HODsurvey>

# EPFL Extension School

## Courses & Programs in Data & Data Science 2018

---

Applied Data Science: Machine Learning

Next cohort - 15 June

---

"50 Things You Need to Know about Data"

Starts - 18 September  
Exclusive enrollment for workshop participants:  
<http://bit.do/data50>

---

Thinking and Creating with Code

Starts - December 2018  
Enrollment opens September 2018

---

Deep Learning

Enrollment opens end 2018

Enroll today at [exts.epfl.ch](http://exts.epfl.ch)



More questions? [xavier.adam@epfl.ch](mailto:xavier.adam@epfl.ch) or twitter [@xvrdm](https://twitter.com/xvrdm)  
...and thanks a lot for coming!