

Meet the Toolkit

K Arnold, based on IntroDS.org

Q&A

Will we use databases / SQL? Yes, in the later part of the class.

Will everything be on Moodle? Moodle Calendar will have all dates, and direct links to anything outside of Moodle that you'll need.

Do you remember your take-away point from Monday?

Logistics

- "Check off" all Moodle activities under "Introduction"
 - Prep 1
 - Lec 1.1
 - Discussion 1
 - Quiz 1
 - Lab 1.2
 - Lec 1.3
- Start Prep 2 for Wednesday (no class Monday)
- Homework 1 posted soon
- Piazza: keep it up!

So far...

- Monday: Overall objectives: projects, topics, dispositions
- Wednesday:
 - Hands on practice with R, RStudio, Git, GitHub
 - First look at summarizing data in R
- Today:
 - Review Wednesday's activity
 - Overview of the toolkit we're using

Questions so far?

Reproducible data analysis

Reproducibility checklist

What does it mean for a data analysis to be "reproducible"?

Reproducibility checklist

What does it mean for a data analysis to be "reproducible"?

Near-term goals:

- Can you re-make all tables and figures easily?
- Does the code actually do what you think it does?
- In addition to what was done, is it clear *why* it was done?

Long-term goals:

- Can the code be used for other data?
- Can you extend the code to do other things?

Toolkit

- Scriptability → R
- Literate programming (code, narrative, output in one place) → R
Markdown
- Version control → Git / GitHub

Tour: R and RStudio

The image shows the RStudio interface with several annotations:

- data viewer**: Points to the 'penguins' data table in the top-left pane.
- environment**: Points to the 'Values' section in the top-right pane, showing 'x' with value '2'.
- arithmetic**: Points to the console output of the first command: `> 2 + 2`.
- load package**: Points to the console command: `> library(palmerpenguins)`.
- view data**: Points to the console command: `> View(penguins)`.
- object assignment**: Points to the console command: `> x <- 2`.
- access variable**: Points to the console command: `> penguins$flipper_length_mm`.
- use function**: Points to the console command: `> mean(penguins$flipper_length_mm)`.
- get help**: Points to the console command: `> ?mean`.

The console shows the following commands and output:

```
> 2 + 2
[1] 4
> x <- 2
> x * 3
[1] 6
> library(palmerpenguins)
> View(penguins)
> penguins$flipper_length_mm
[1] 181 186 195 NA 193 190 181 195 193 190 186 180 182 191
[337] 206 189 195 207 202 193 210 198
> mean(penguins$flipper_length_mm)
[1] NA
> ?mean
> mean(penguins$flipper_length_mm, na.rm = TRUE)
[1] 200.9152
```

The right pane shows the documentation for the `mean` function, including its description, usage, arguments, and examples.

A short list (for now) of R essentials

- Functions are (most often) verbs, followed by what they will be applied to in parentheses:

```
do_this(to_this)  
do_that(to_this, to_that, with_those)
```

A short list (for now) of R essentials

- Functions are (most often) verbs, followed by what they will be applied to in parentheses:

```
do_this(to_this)  
do_that(to_this, to_that, with_those)
```

- Packages are loaded with the `library` function:

```
library(package_name)
```

R essentials (continued)

- Columns (variables) in data frames are accessed with \$:

```
dataframe$var_name
```

R essentials (continued)

- Columns (variables) in data frames are accessed with \$:

```
dataframe$var_name
```

- Object documentation can be accessed with ?

```
?mean
```

tidyverse



tidyverse.org

- The **tidyverse** is an opinionated collection of R packages designed for data science
- All packages share an underlying philosophy and a common grammar

rmarkdown

rmarkdown.rstudio.com

- write code and prose in reproducible computational documents



R Markdown

R Markdown

- Fully reproducible reports -- each time you knit the analysis is ran from the beginning
- Simple markdown syntax for text
- Code goes in chunks, defined by three backticks, narrative goes outside of chunks

Tour: R Markdown

The image shows the RStudio interface with an R Markdown document titled "Bechdel.Rmd" open in the editor. The document is divided into three main sections: a YAML header, a text paragraph, and a code chunk. Handwritten annotations highlight these sections: "knit" points to the Knit button in the toolbar; "link" points to a URL in the text; "code chunk" points to the R code block. The rendered HTML output is shown on the right, displaying the title "Bechdel", the author "Mine Çetinkaya-Rundel", and the text paragraph. The code chunk is rendered as a preformatted block of R code.

knit

link

code chunk

yaml

```
1 ---
2 title: "Bechdel"
3 author: "Mine Çetinkaya-Rundel"
4 output:
5   html_document:
6     fig_height: 4
7     fig_width: 9
8 ---
9
10 In this mini analysis we work with the data used
11 in the FiveThirtyEight story titled ["The
12 Dollar-And-Cents Case Against Hollywood's
13 Exclusion of Women"](https://fivethirtyeight.com/f
14 eatures/the-dollar-and-cents-case-against-hollywo
15 ds-exclusion-of-women/). Your task is to fill in
16 the blanks denoted by `___`.
17
18 ## Data and packages
19
20 We start with loading the packages we'll use.
21
22 ```{r load-packages, message=FALSE}
23 library(fivethirtyeight)
24 library(tidyverse)
25 ```
```

Bechdel

Mine Çetinkaya-Rundel

In this mini analysis we work with the data used in the [FiveThirtyEight](https://fivethirtyeight.com/features/the-dollar-and-cents-case-against-hollywoods-exclusion-of-women/) story titled "[The Dollar-And-Cents Case Against Hollywood's Exclusion of Women](https://fivethirtyeight.com/features/the-dollar-and-cents-case-against-hollywoods-exclusion-of-women/)". Your task is to fill in the blanks denoted by `___`.

Data and packages

We start with loading the packages we'll use.

```
library(fivethirtyeight)
library(tidyverse)
```

The dataset contains information on 1794 movies released between 1970 and 2013. However we'll focus our analysis on movies released between 1990 and 2013.

```
bechdel90_13 <- bechdel %>%
  filter(between(year, 1990, 2013))
```

There are `___` such movies.

The financial variables we'll focus on are the following:

- `budget_2013` : Budget in 2013 inflation adjusted dollars
- `domgross_2013` : Domestic gross (US) in 2013 inflation adjusted dollars
- `intgross_2013` : Total International (i.e., worldwide) gross in 2013 inflation adjusted dollars

Environments

The environment of your R Markdown document is separate from the Console!

Remember this, and expect it to bite you a few times as you're learning to work with R Markdown!

Environments

First, run the following in the console

```
x <- 2  
x * 3
```

All looks good, eh?

Environments

First, run the following in the console

```
x <- 2  
x * 3
```

All looks good, eh?

Then, add the following in an R chunk in your R Markdown document

```
x * 3
```

What happens? Why the error?

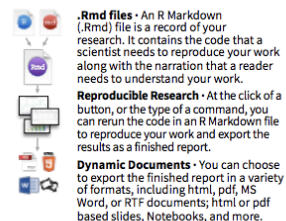
R Markdown help

R Markdown Cheat Sheet

Help -> Cheatsheets

R Markdown :: CHEAT SHEET

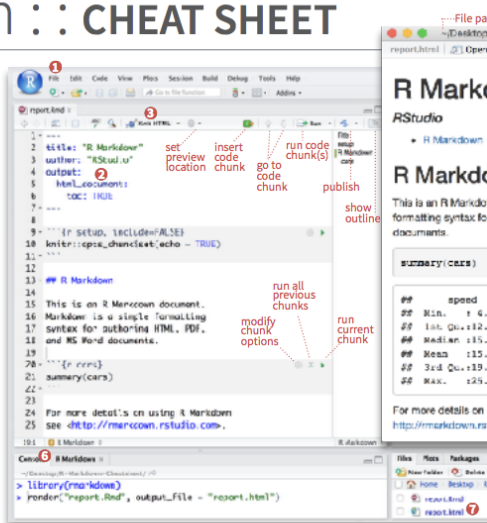
What is R Markdown?



Workflow



- 1 **Open a new .Rmd file** at File ► New File ► R Markdown. Use the wizard that opens to pre-populate the file with a template
- 2 **Write document** by editing template
- 3 **Knit document to create report**; use knit button or render() to knit
- 4 **Preview Output** in IDE window
- 5 **Publish** (optional) to web server
- 6 **Examine build log** in R Markdown console
- 7 **Use output file** that is saved along side .Rmd






render

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

input - file to render	output_options - List of render options (as in YAML)	output_file	params - list of params to use
output_format		output_dir	

Embed code with knitr syntax

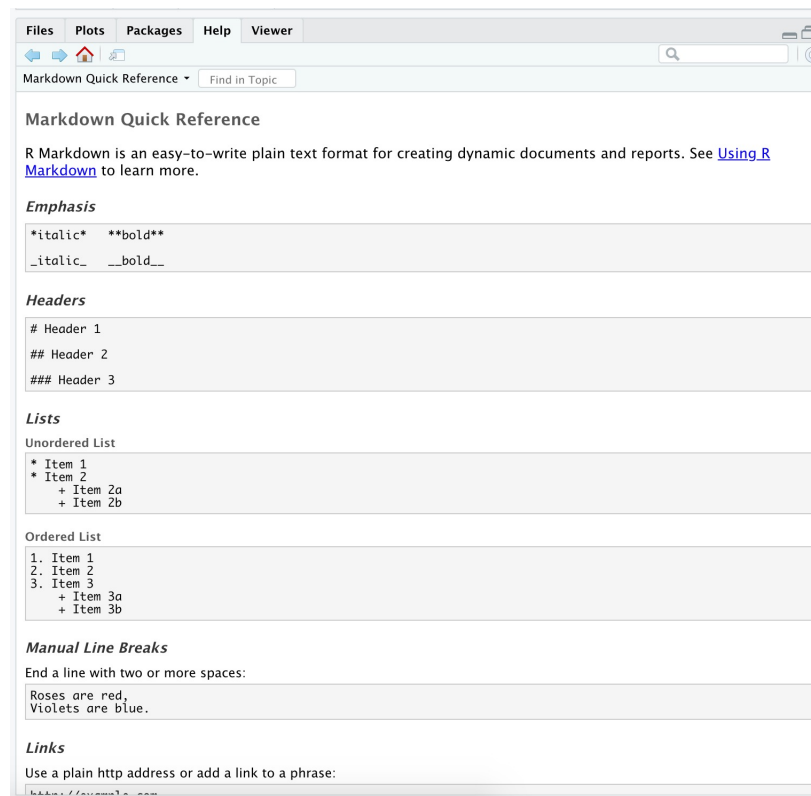
INLINE CODE
Insert with ``r<code>``. Results appear as text without code.
Built with ``rgetRversion()`` Built with 3.2.3

CODE CHUNKS
One or more lines surrounded with ````{r}` and `````. Place chunk options within curly braces, after `r`. Insert with  ````{r echo=TRUE}`
`getVersion()`  

```
GLOBAL OP1
Set with knitr::opts_chunk$set(
  `r` = FALSE,
  knitr::opts_chunk$set(
    `r` = FALSE,
    ...
  )
)
```

Markdown Quick Reference

Help -> Markdown Quick Reference



How will we use R Markdown?

- Every assignment / report / project / etc. is an R Markdown document
- You'll always have a template R Markdown document to start with
- The amount of scaffolding in the template will decrease over the semester

Getting help in R

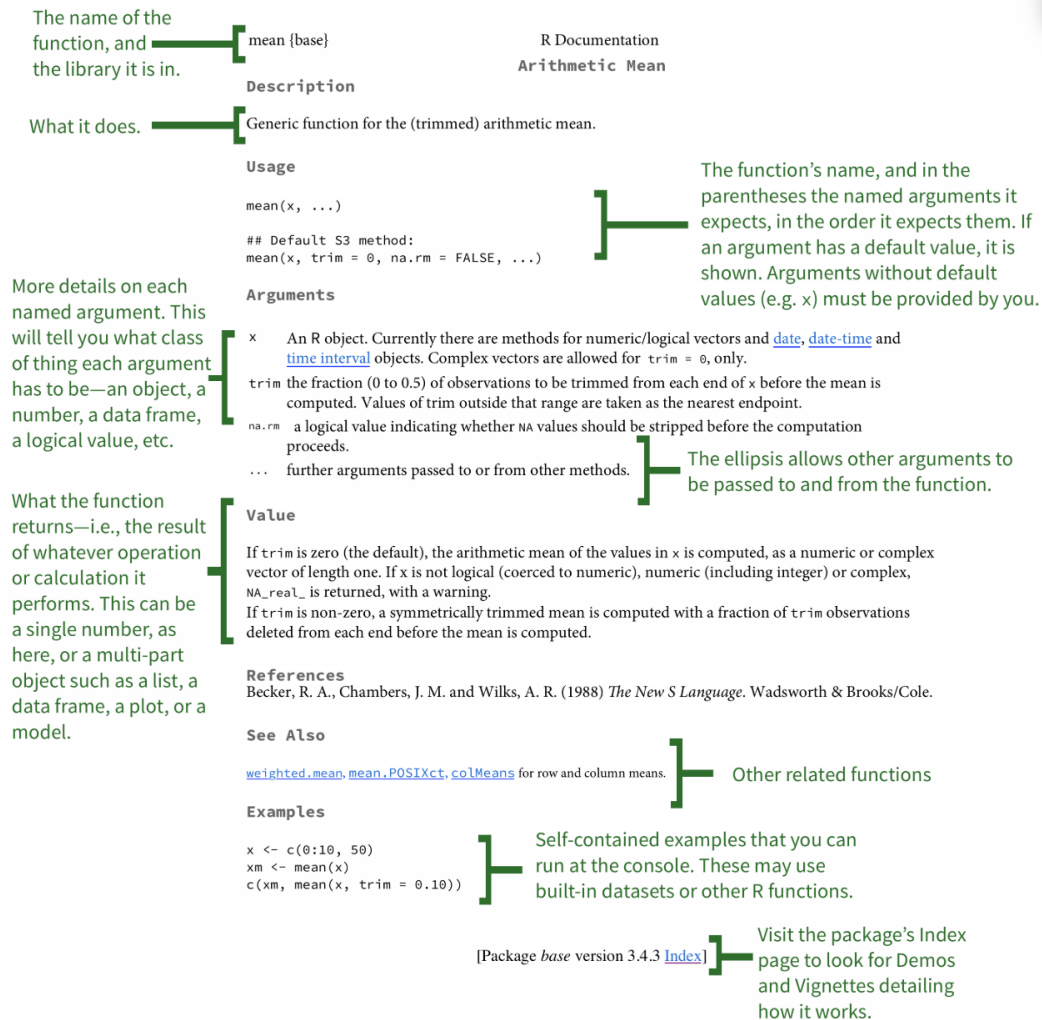


Figure A.1: The structure of an R help page.

Source: <http://socviz.co/appendix.html#a-little-more-about-r>

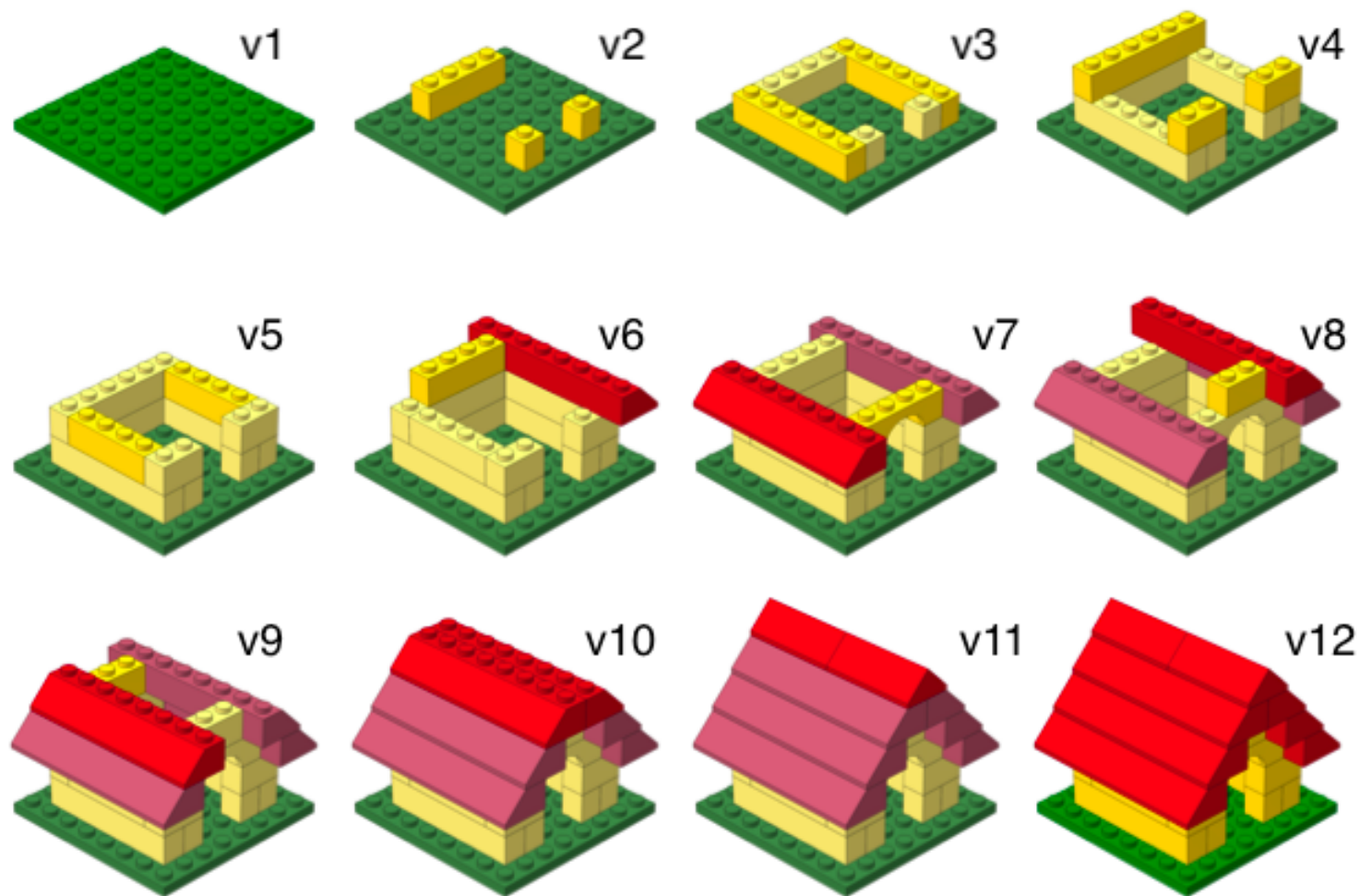
Version Control

Git and GitHub



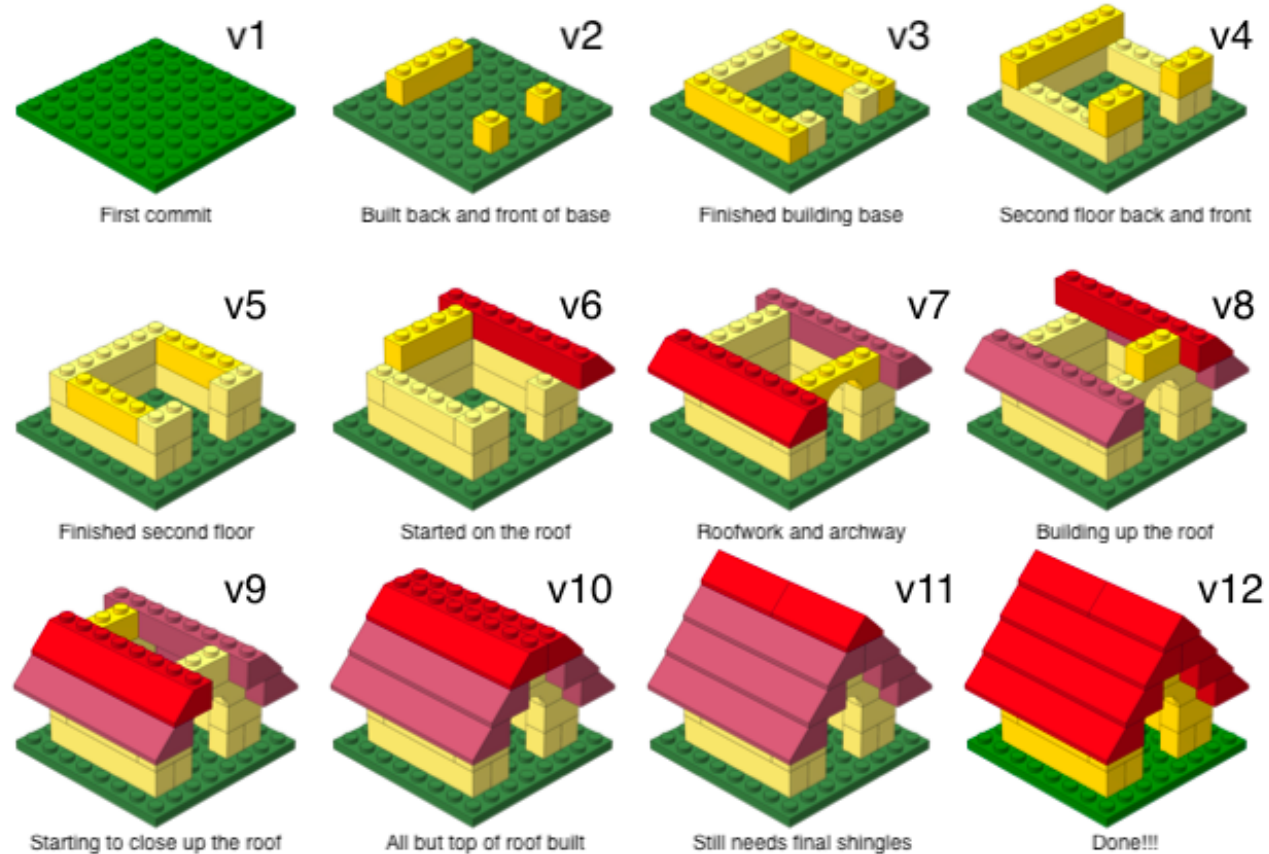
- Git is a version control system -- like “Track Changes” features from Microsoft Word, on steroids
- It's not the only version control system, but it's a very popular one
- GitHub is the home for your Git-based projects on the internet
- We will use GitHub as a platform for web hosting and collaboration

Versioning

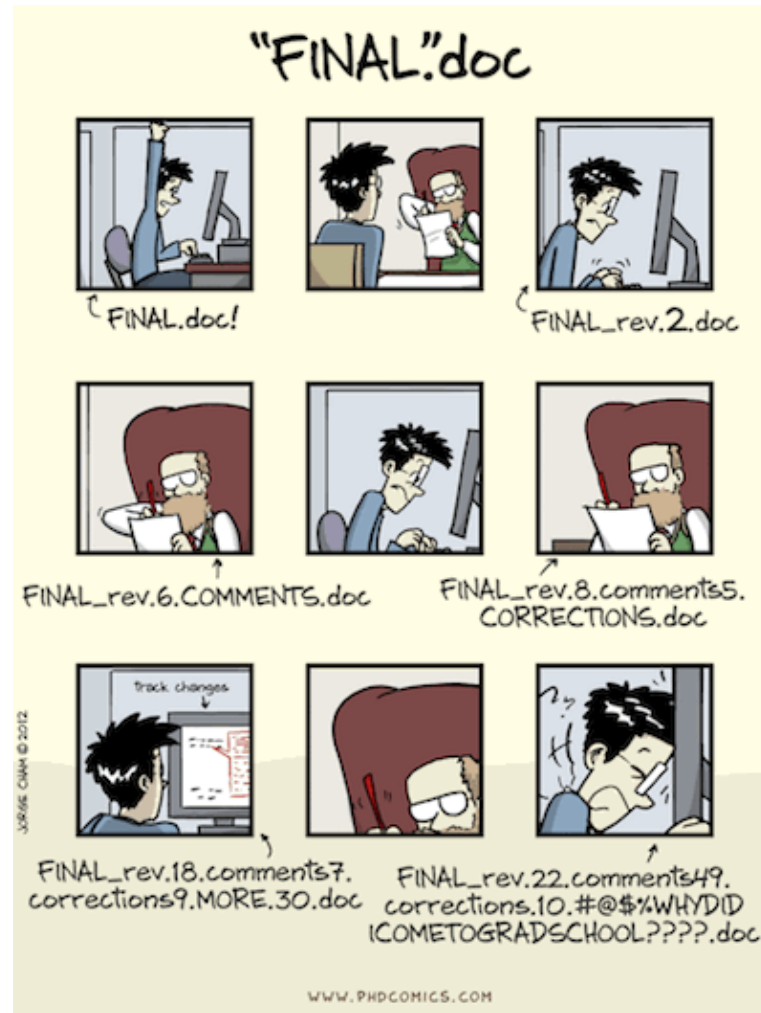


Versioning

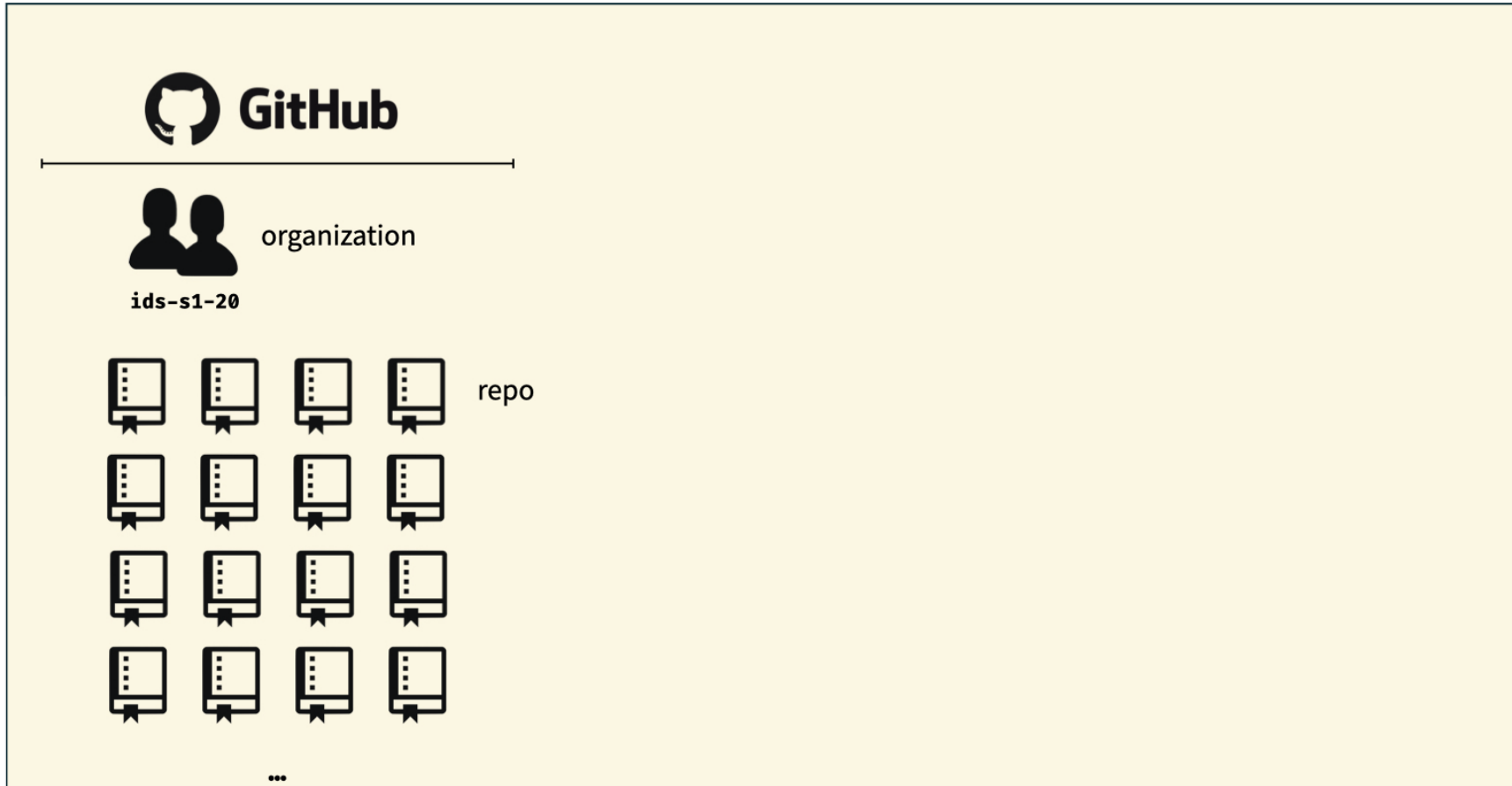
with human readable messages



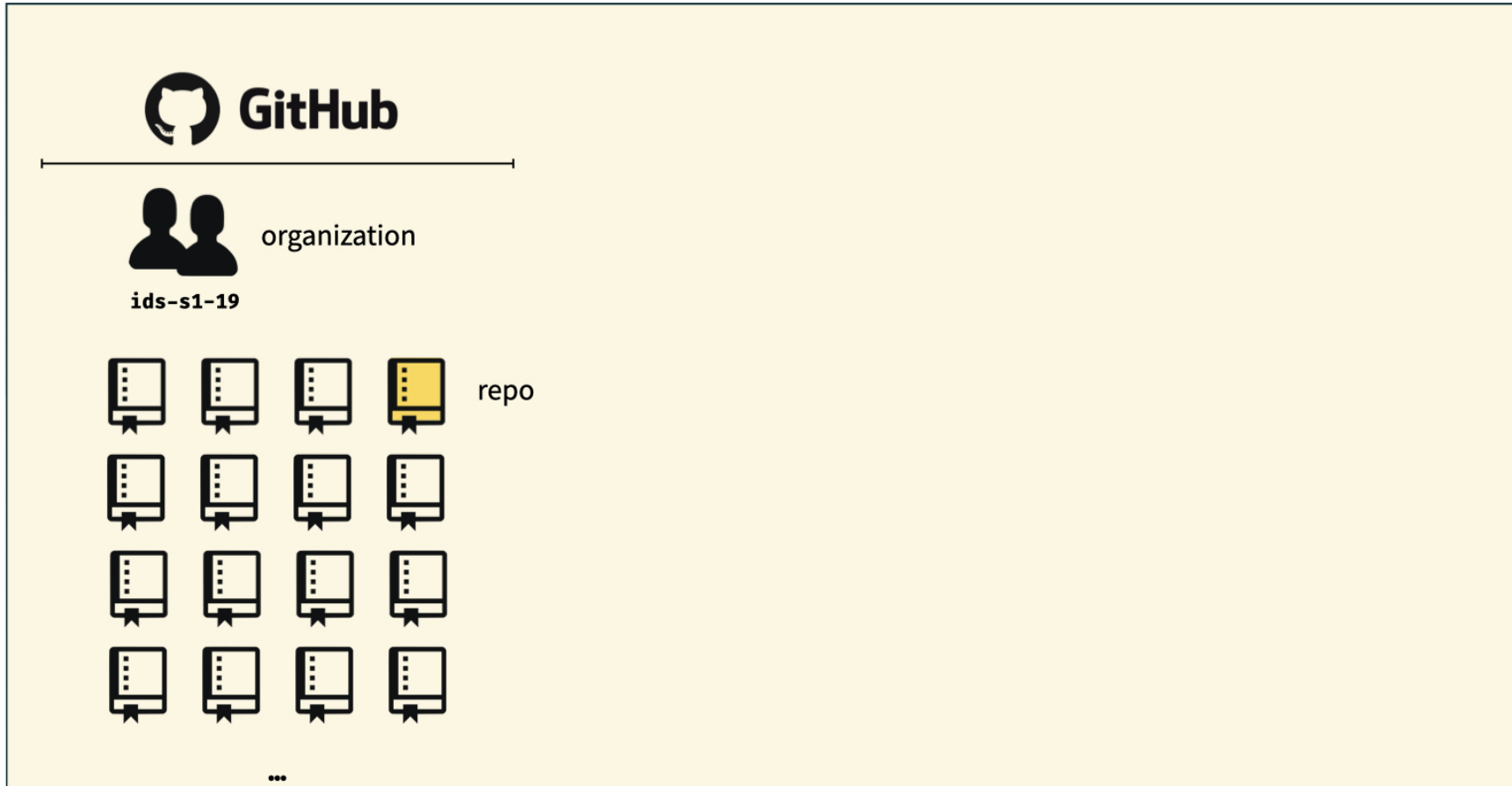
Why do we need version control?



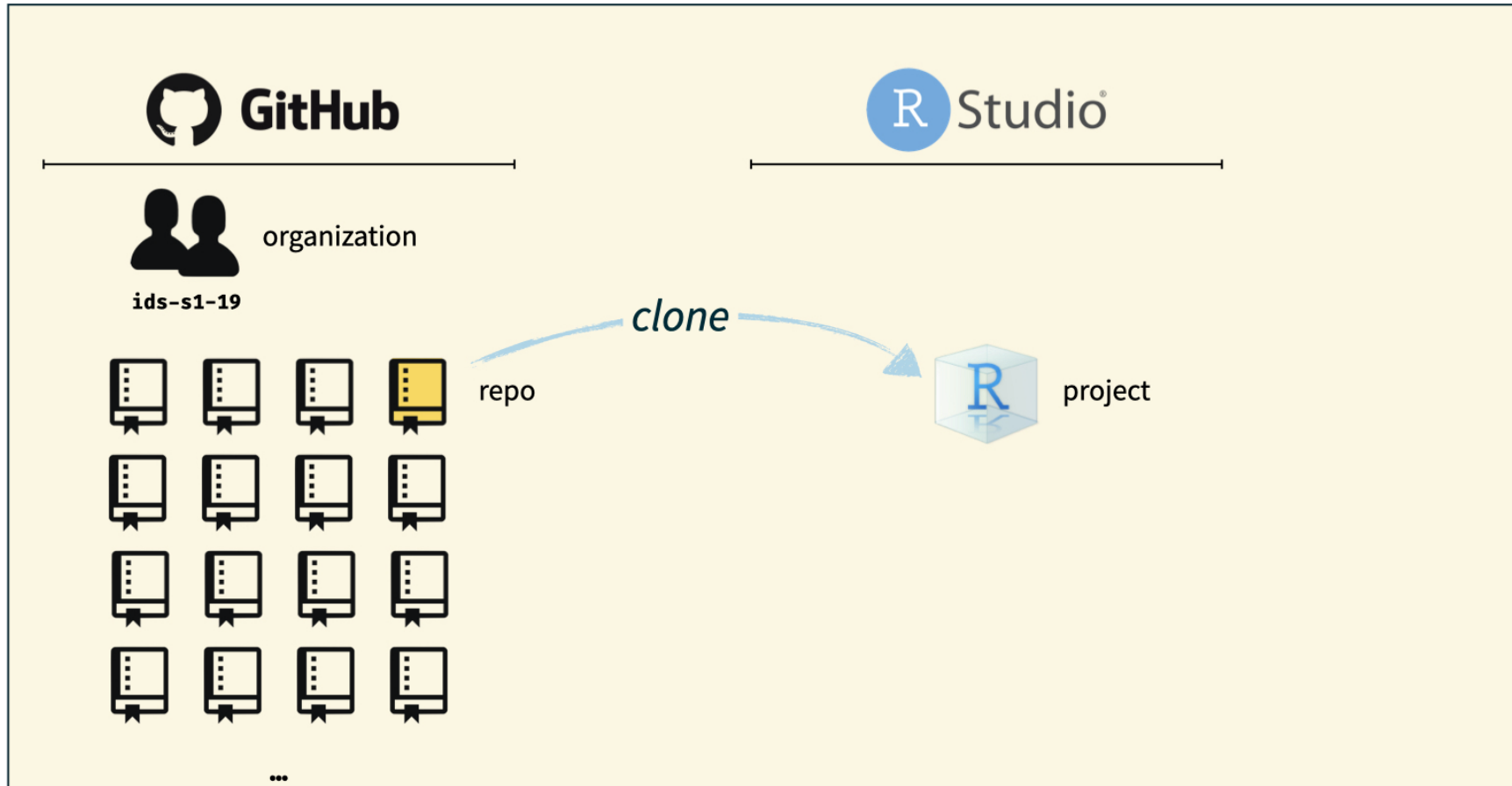
How will we use Git and GitHub?



How will we use Git and GitHub?



How will we use Git and GitHub?



How will we use Git and GitHub?

