Intro to the Tidyverse

Thomas Mock

RStudio, Inc.

updated: 2020-09-08

Today's Agenda

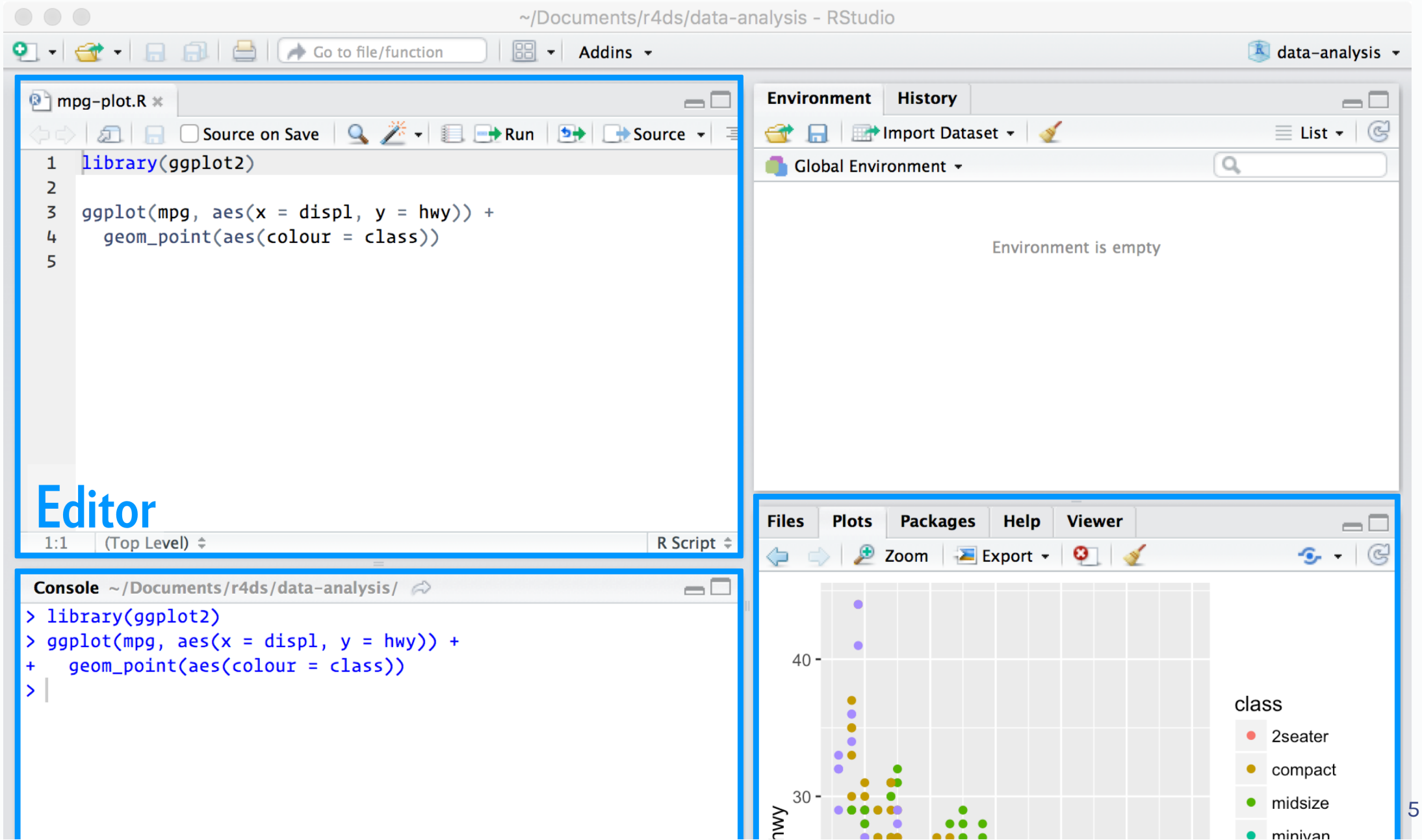
- RStudio
  - Writing code
  - File manipulation
  - Package control
- R coding basics
  - Math
  - Assignment
  - Functions
  - Load and install packages
- `Tidyverse`
  - Read data in with `readr`
  - Tidy data with`tidyr`
  - Transform data with `dplyr`
  - Plot data with `ggplot2`

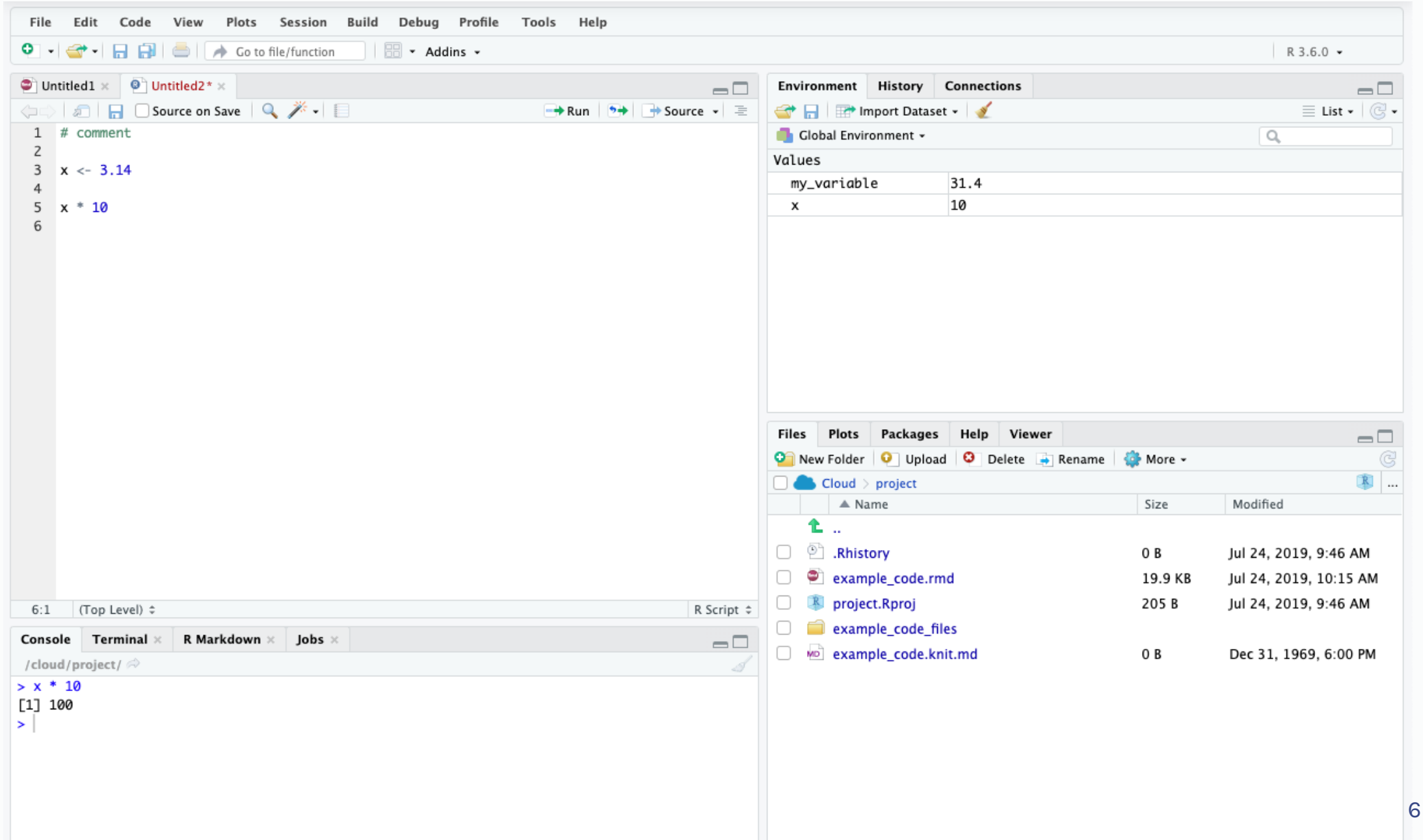
RStudio

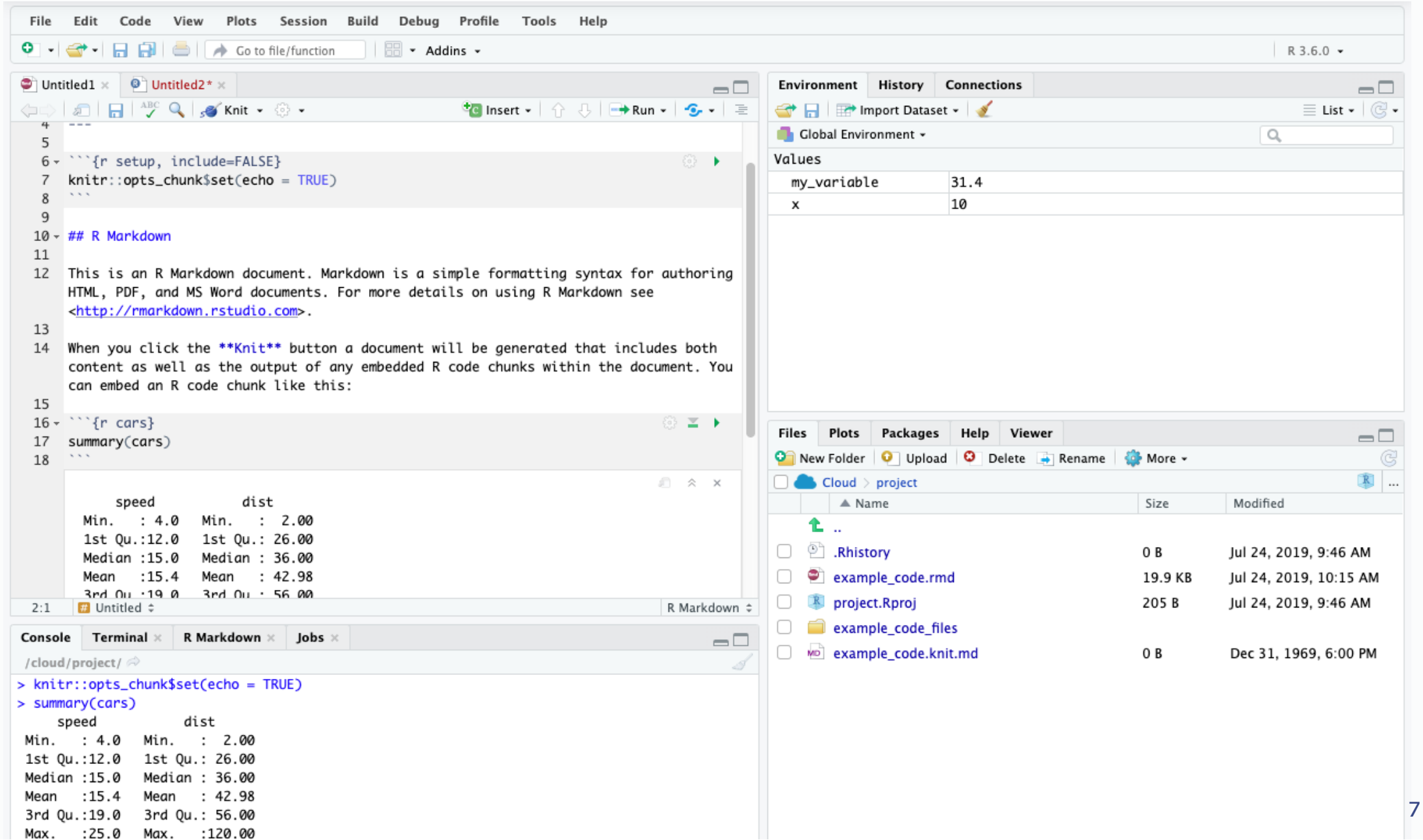
RStudio is an IDE (integrated development environment)

- A place to write
    - Console
    - R Scripts
    - R Markdown
    - Code Completion
    - Debugging
- A place to open *things*
    - File and path exploration
    - Open plots, data, .R/.Rmd file
- A place for projects
    - Self-contained structure
    - Consistent/easy pathing
    - Keep relevant files/code together with output

RStudio Basics


~/Documents/r4ds/data-analysis - RStudio
Go to file/function
Addins
data-analysis
mpg-plot.R
Source on Save
Run
Source
1 library(ggplot2)
2
3 ggplot(mpg, aes(x = displ, y = hwy)) +
4 geom_point(aes(colour = class))
5
Editor
1:1
(Top Level)
R Script
Environment
History
Import Dataset
List
Global Environment
Environment is empty
Files
Plots
Packages
Help
Viewer
Zoom
Export
Console ~/Documents/r4ds/data-analysis/
> library(ggplot2)
> ggplot(mpg, aes(x = displ, y = hwy)) +
+ geom_point(aes(colour = class))
>
40
30
hwy
class
2seater
compact
midsize
minivan


File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
R 3.6.0
Untitled1
Untitled2*
Source on Save
Run
Source
1 # comment
2
3 x <- 3.14
4
5 x * 10
6
6:1
(Top Level)
R Script
Console
Terminal
R Markdown
Jobs
/cloud/project/
> x * 10
[1] 100
>
Environment
History
Connections
Import Dataset
List
Global Environment
Values
my_variable 31.4
x 10
Files
Plots
Packages
Help
Viewer
New Folder
Upload
Delete
Rename
More
Cloud
project
Name
Size
Modified
..
.Rhistory 0 B Jul 24, 2019, 9:46 AM
example_code.rmd 19.9 KB Jul 24, 2019, 10:15 AM
project.Rproj 205 B Jul 24, 2019, 9:46 AM
example_code_files
example_code.knit.md 0 B Dec 31, 1969, 6:00 PM

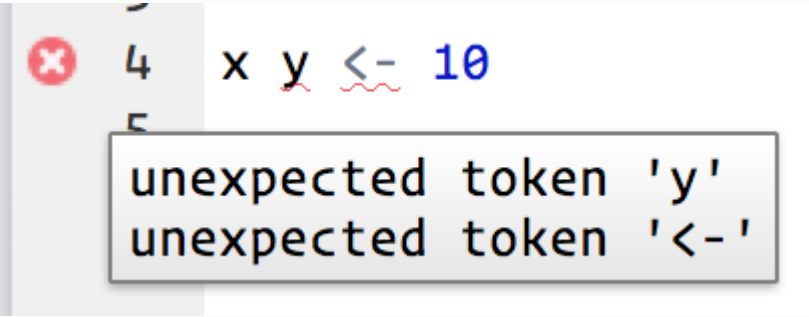

File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
R 3.6.0
Untitled1
Untitled2*
Knit
Insert
Run

```{r setup, include=FALSE}
knitr::opts_chunk$set(echo = TRUE)
```
R Markdown
This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.
When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:
```{r cars}
summary(cars)
```
speed dist
Min. : 4.0 Min. : 2.00
1st Qu.:12.0 1st Qu.: 26.00
Median :15.0 Median : 36.00
Mean :15.4 Mean : 42.98
3rd Qu.:19.0 3rd Qu.: 56.00
2:1
Untitled
R Markdown
Console
Terminal
R Markdown
Jobs
/cloud/project/
> knitr::opts_chunk$set(echo = TRUE)
> summary(cars)
speed dist
Min. : 4.0 Min. : 2.00
1st Qu.:12.0 1st Qu.: 26.00
Median :15.0 Median : 36.00
Mean :15.4 Mean : 42.98
3rd Qu.:19.0 3rd Qu.: 56.00
Max. :25.0 Max. :120.00
Environment
History
Connections
Import Dataset
List
Global Environment
Values
my_variable 31.4
x 10
Files
Plots
Packages
Help
Viewer
New Folder
Upload
Delete
Rename
More
Cloud
project
Name
Size
Modified
..
.Rhistory 0 B Jul 24, 2019, 9:46 AM
example_code.rmd 19.9 KB Jul 24, 2019, 10:15 AM
project.Rproj 205 B Jul 24, 2019, 9:46 AM
example_code_files
example_code.knit.md 0 B Dec 31, 1969, 6:00 PM

RStudio Diagnostics

The script editor highlights syntax errors

Hover over the cross to see the problem



RStudio also warns about potential problems

Everyone makes mistakes!

Errors are *ok*, it happens to everyone!

```
my_variable <- x * 3.14

my_variab1e
```

```
## Warning: # Error: object 'my_variab1e' not found
```

```
my_variable
```

```
## [1] 31.4
```

- R is essentially the most aggressive spell-checker of all time!
- Focus on what the error says and if you don't understand it, Googling can often help!

R Code Basics

R Code Basics

Assignment

```
x <- 3 * 4
x
```

```
## [1] 12
```

```
y <- 5
y * x
```

```
## [1] 60
```

```
tmp_df <- data.frame(
  col_1 = c(1, 2, 3),
  col_2 = c("a", "b", "c")
  )

tmp_df
```

```
##   col_1 col_2
## 1     1     a
## 2     2     b
## 3     3     c
```

Functions

A function is essentially shorthand to call specific code

```
function_name(arg1, arg2, arg3)
```

```
seq(from, to, by,
    length.out, along.with)
```

```
seq(from = 10, to = 100, by = 10)
```

```
##  [1]  10  20  30  40  50  60  70  80  90 100
```

```
seq(10, 100, 10)
```

```
##  [1]  10  20  30  40  50  60  70  80  90 100
```

```
result_out <- seq(10, 100, length.out = 5)

result_out
```

```
## [1]  10.0  32.5  55.0  77.5 100.0
```

The %>% == and then

Rather than multiple assignment or nesting functions

```
did_something <- do_something(data)

did_another_thing <- do_another_thing(did_something)

final_thing <- do_last_thing(did_another_thing)
```

```
final_thing <- do_last_thing(
  do_another_thing(
    do_something(
      data
      )
    )
  )
```

```
final_thing <- data %>%
      do_something() %>%
      do_another_thing() %>%
      do_last_thing()
```

The Pipe %>%

```
data %>%
  do_something(.) %>%
  do_another_thing(.) %>%
  do_last_thing(.)
```

`do_something(data)` is equivalent to:

- `data %>% do_something(data = .)`
- `data %>% do_something(.)`
- `data %>% do_something()`

```r
data_in <- seq(10, 100, by = 10)

result_out <- mean(data_in)

result_out
```

```
## [1] 55
```

```r
mean(seq(10,100, by = 10))
```

```
## [1] 55
```

```r
seq(10, 100, by = 10) %>%
  mean()
```

```
## [1] 55
```

```r
mean_output <- seq(10, 100, by = 10) %>%
  mean()

mean_output
```

```
## [1] 55
```

About the penguins

```
penguins <- palmerpenguins::penguins

penguins %>%
  glimpse()
```

```
## Rows: 344
## Columns: 8
## $ species           <fct> Adelie, Adelie, Adelie, Adelie, Adelie, Adelie, Ade…
## $ island            <fct> Torgersen, Torgersen, Torgersen, Torgersen, Torgers…
## $ bill_length_mm    <dbl> 39.1, 39.5, 40.3, NA, 36.7, 39.3, 38.9, 39.2, 34.1,…
## $ bill_depth_mm     <dbl> 18.7, 17.4, 18.0, NA, 19.3, 20.6, 17.8, 19.6, 18.1,…
## $ flipper_length_mm <int> 181, 186, 195, NA, 193, 190, 181, 195, 193, 190, 18…
## $ body_mass_g       <int> 3750, 3800, 3250, NA, 3450, 3650, 3625, 4675, 3475,…
## $ sex               <fct> male, female, female, NA, female, male, female, mal…
## $ year              <int> 2007, 2007, 2007, 2007, 2007, 2007, 2007, 2007, 200…
```

More Complex Example

```
penguins %>%
  filter(species == "Adelie" & !is.na(sex)) %>%
  group_by(sex, island) %>%
  summarize(mean = mean(body_mass_g, na.rm = TRUE))
```

```
## `summarise()` regrouping output by 'sex' (override with `.groups` argument)

## # A tibble: 6 x 3
## # Groups:   sex [2]
##   sex    island     mean
##   <fct>  <fct>     <dbl>
## 1 female Biscoe    3369.
## 2 female Dream     3344.
## 3 female Torgersen 3396.
## 4 male   Biscoe    4050
## 5 male   Dream     4046.
## 6 male   Torgersen 4035.
```


tidyverse

Tidyverse

An opinionated collection of R packages for data science.

All packages share an underlying design philosophy, grammar, and data structures

- Core packages - `readr`, `tidyr`, `dplyr`, `ggplot2`

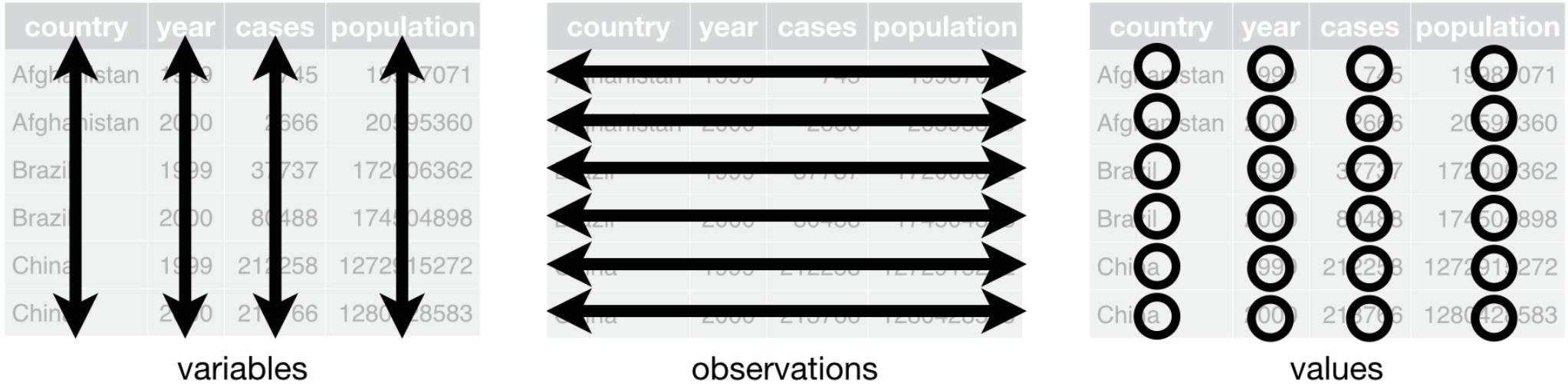
Tidyverse

`Tidyverse` is an R package, as such you need to do two things to be able to use it

- `install.packages("tidyverse")`
    - This downloads and installs the `tidyverse`
- `library(tidyverse)`
    - This loads and gives you access to the `tidyverse` package

`tidyverse` Core Principles

- Built around `data` - usually as a `data.frame` or `tibble`
- Built around `tidy` data
  - Each `variable` in its own `column`
  - Each `observation` or `case` in its own `row`
  - Each type of observational units forms a table



Untidy data

```
untidy_df
```

```
## # A tibble: 5 x 7
##   age_group male_2016 female_2016 male_2017 female_2017 male_2018 female_2018
##   <chr>         <dbl>       <dbl>     <dbl>       <dbl>     <dbl>       <dbl>
## 1 < 18          22000       20000     22000       20000     22000       20000
## 2 18-30         36000       35000     36000       35000     36000       35000
## 3 31-50         50000       40000     50000       40000     50000       40000
## 4 51-60         62000       60000     62000       60000     62000       60000
## 5 > 60          75000       72000     75000       72000     75000       72000
```

Tidy data

```
tidy_df
```

```
## # A tibble: 30 x 4
##    age_group gender year  income
##    <chr>     <chr>  <chr>  <dbl>
##  1 < 18      male   2016   22000
##  2 18-30     male   2016   36000
##  3 31-50     male   2016   50000
##  4 51-60     male   2016   62000
##  5 > 60      male   2016   75000
##  6 < 18      female 2016   20000
##  7 18-30     female 2016   35000
##  8 31-50     female 2016   40000
##  9 51-60     female 2016   60000
## 10 > 60      female 2016   72000
## # … with 20 more rows
```

Tidy the data

```
untidy_df %>%
  pivot_longer(cols = male_2016:female_2018,
               names_to = "gender_year",
               values_to = "income") %>%
  separate(gender_year, into = c("gender", "year"))
```

```
## # A tibble: 30 x 4
##    age_group gender year  income
##    <chr>     <chr>  <chr>  <dbl>
##  1 < 18      male   2016   22000
##  2 < 18      female 2016   20000
##  3 < 18      male   2017   22000
##  4 < 18      female 2017   20000
##  5 < 18      male   2018   22000
##  6 < 18      female 2018   20000
##  7 18-30     male   2016   36000
##  8 18-30     female 2016   35000
##  9 18-30     male   2017   36000
## 10 18-30     female 2017   35000
## # … with 20 more rows
```

Tidy the data

```
untidy_df %>%
  pivot_longer(
    cols = male_2016:female_2018,
    names_to = c("gender", "year"),
    names_pattern = "(.*)_(.*)",
    values_to = "income"
  )
```

```
## # A tibble: 30 x 4
##    age_group gender year  income
##    <chr>     <chr>  <chr>  <dbl>
##  1 < 18      male   2016   22000
##  2 < 18      female 2016   20000
##  3 < 18      male   2017   22000
##  4 < 18      female 2017   20000
##  5 < 18      male   2018   22000
##  6 < 18      female 2018   20000
##  7 18-30     male   2016   36000
##  8 18-30     female 2016   35000
##  9 18-30     male   2017   36000
## 10 18-30     female 2017   35000
## # … with 20 more rows
```

Read data in

Read in data with `readr`, `haven`, `readxl`

`readr`

- `read_csv()`, `read_tsv()`, `read_delim()`

`haven`

- `read_sas()`, `read_spss()`, `read_stata()`, `read_dta()`

`readxl`

- `read_xls()`, `read_xlsx()`, `read_excel()`

dplyr

6 Main verbs

- filter()
- arrange()
- select()
- mutate()
- group_by()
- summarise()

Simple use

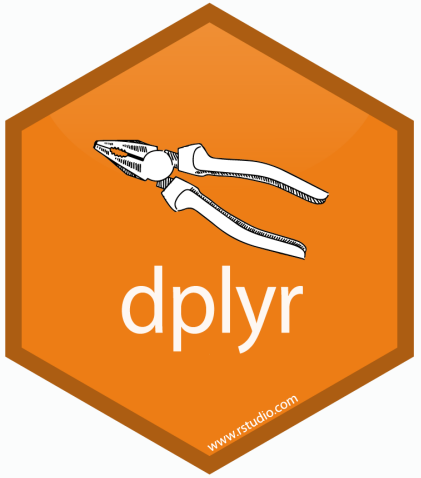
- pull()
- n()/count()
- glimpse()

Advanced iterations

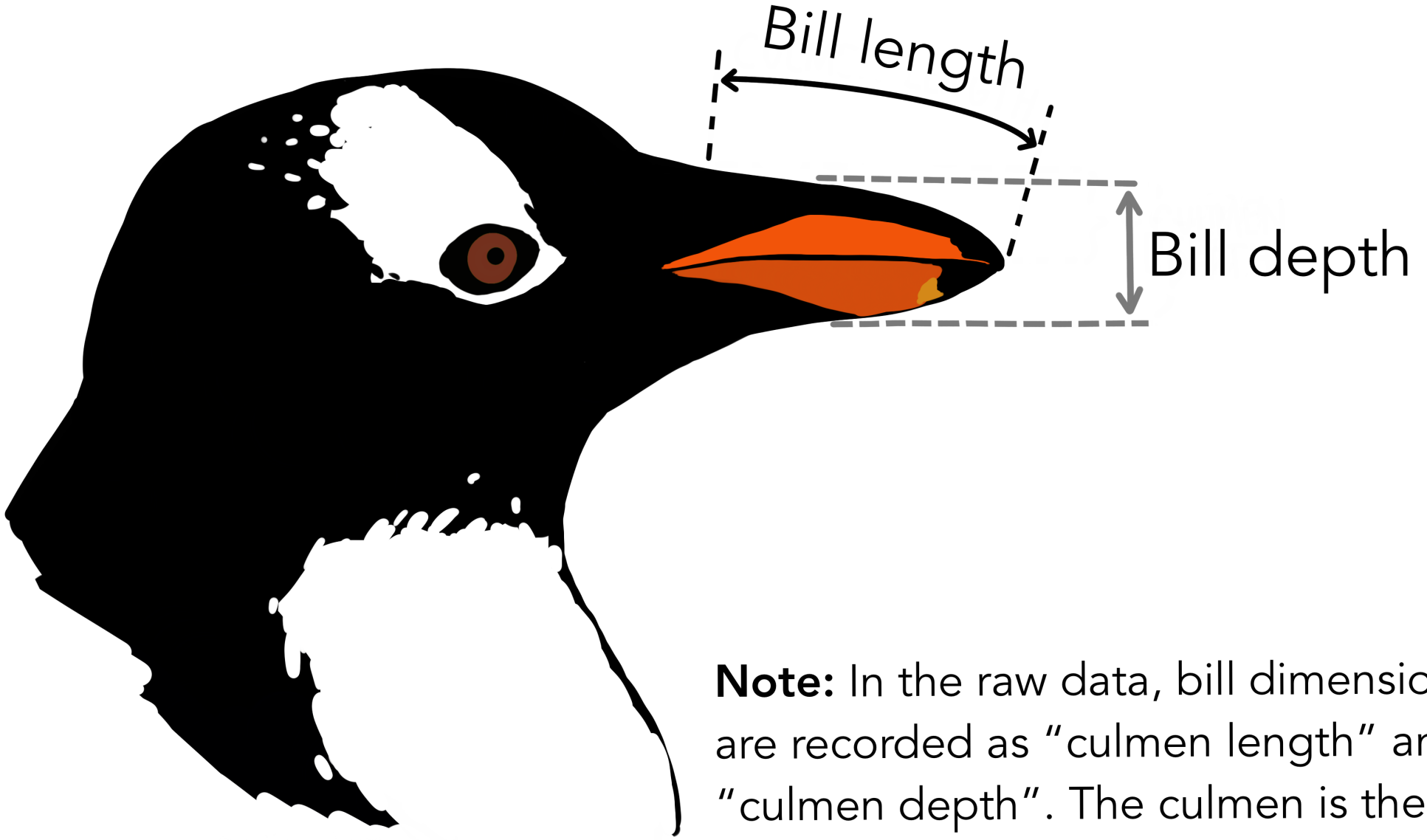
- across()
- rowwise()

More info

- dplyr.tidyverse.org
- R for Data Science



Meet the penguins



Note: In the raw data, bill dimensions are recorded as "culmen length" and "culmen depth". The culmen is the dorsal ridge atop the bill.

penguins dataset

`data.frame` vs `tibble`

```
penguin_df <- palmerpenguins::penguins %>% as.data.frame()
class(penguin_df)
```

```
## [1] "data.frame"
```

```
penguins <- as_tibble(penguin_df)

class(penguins)
```

```
## [1] "tbl_df"     "tbl"        "data.frame"
```

penguins dataset

`data.frame` vs `tibble`

```
penguin_df
```

```
##       species    island bill_length_mm bill_depth_mm flipper_length_mm
## 1      Adelie Torgersen           39.1          18.7               181
## 2      Adelie Torgersen           39.5          17.4               186
## 3      Adelie Torgersen           40.3          18.0               195
## 4      Adelie Torgersen             NA            NA                NA
## 5      Adelie Torgersen           36.7          19.3               193
## 6      Adelie Torgersen           39.3          20.6               190
## 7      Adelie Torgersen           38.9          17.8               181
## 8      Adelie Torgersen           39.2          19.6               195
## 9      Adelie Torgersen           34.1          18.1               193
## 10     Adelie Torgersen           42.0          20.2               190
## 11     Adelie Torgersen           37.8          17.1               186
## 12     Adelie Torgersen           37.8          17.3               180
## 13     Adelie Torgersen           41.1          17.6               182
## 14     Adelie Torgersen           38.6          21.2               191
## 15     Adelie Torgersen           34.6          21.1               198
## 16     Adelie Torgersen           36.6          17.8               185
## 17     Adelie Torgersen           38.7          19.0               195
```

penguins dataset

```
penguins <- as_tibble(penguin_df)
```

```
penguins
```

```
## # A tibble: 344 x 8
##    species island bill_length_mm bill_depth_mm flipper_length_… body_mass_g
##    <fct>   <fct>           <dbl>         <dbl>             <int>       <int>
##  1 Adelie  Torge…           39.1          18.7               181        3750
##  2 Adelie  Torge…           39.5          17.4               186        3800
##  3 Adelie  Torge…           40.3          18                 195        3250
##  4 Adelie  Torge…           NA            NA                  NA          NA
##  5 Adelie  Torge…           36.7          19.3               193        3450
##  6 Adelie  Torge…           39.3          20.6               190        3650
##  7 Adelie  Torge…           38.9          17.8               181        3625
##  8 Adelie  Torge…           39.2          19.6               195        4675
##  9 Adelie  Torge…           34.1          18.1               193        3475
## 10 Adelie  Torge…           42            20.2               190        4250
## # … with 334 more rows, and 2 more variables: sex <fct>, year <int>
```

penguins dataset

```
head(penguins, 5)
```

```
## # A tibble: 5 x 8
##   species island bill_length_mm bill_depth_mm flipper_length_… body_mass_g sex
##   <fct>   <fct>           <dbl>         <dbl>             <int>       <int> <fct>
## 1 Adelie  Torge…           39.1          18.7               181        3750 male
## 2 Adelie  Torge…           39.5          17.4               186        3800 fema…
## 3 Adelie  Torge…           40.3          18                 195        3250 fema…
## 4 Adelie  Torge…           NA            NA                  NA          NA <NA>
## 5 Adelie  Torge…           36.7          19.3               193        3450 fema…
## # … with 1 more variable: year <int>
```

```
tail(penguins, 5)
```

```
## # A tibble: 5 x 8
##   species island bill_length_mm bill_depth_mm flipper_length_… body_mass_g sex
##   <fct>   <fct>           <dbl>         <dbl>             <int>       <int> <fct>
## 1 Chinst… Dream           55.8          19.8               207        4000 male
## 2 Chinst… Dream           43.5          18.1               202        3400 fema…
## 3 Chinst… Dream           49.6          18.2               193        3775 male
## 4 Chinst… Dream           50.8          19                 210        4100 male
## 5 Chinst… Dream           50.2          18.7               198        3775 fema…
```

dplyr::slice()

```
slice(penguins, 1:3)
```

```
## # A tibble: 3 x 8
##   species island bill_length_mm bill_depth_mm flipper_length_… body_mass_g sex
##   <fct>   <fct>           <dbl>         <dbl>             <int>       <int> <fct>
## 1 Adelie  Torge…           39.1          18.7               181        3750 male
## 2 Adelie  Torge…           39.5          17.4               186        3800 fema…
## 3 Adelie  Torge…           40.3          18                 195        3250 fema…
## # … with 1 more variable: year <int>
```

```
slice(penguins, 1, 3, 5)
```

```
## # A tibble: 3 x 8
##   species island bill_length_mm bill_depth_mm flipper_length_… body_mass_g sex
##   <fct>   <fct>           <dbl>         <dbl>             <int>       <int> <fct>
## 1 Adelie  Torge…           39.1          18.7               181        3750 male
## 2 Adelie  Torge…           40.3          18                 195        3250 fema…
## 3 Adelie  Torge…           36.7          19.3               193        3450 fema…
## # … with 1 more variable: year <int>
```

dplyr::slice_min() & dplyr::slice_max()

```
# bottom 3 beak lengths
slice_min(penguins, order_by = bill_length_mm, n = 3)
```

```
## # A tibble: 3 x 8
##   species island bill_length_mm bill_depth_mm flipper_length_… body_mass_g sex
##   <fct>   <fct>           <dbl>         <dbl>             <int>       <int> <fct>
## 1 Adelie  Dream            32.1          15.5               188        3050 fema…
## 2 Adelie  Dream            33.1          16.1               178        2900 fema…
## 3 Adelie  Torge…           33.5          19                 190        3600 fema…
## # … with 1 more variable: year <int>
```

```
# top 3 beak lengths
slice_max(penguins, order_by = bill_length_mm, n = 3)
```

```
## # A tibble: 3 x 8
##   species island bill_length_mm bill_depth_mm flipper_length_… body_mass_g sex
##   <fct>   <fct>           <dbl>         <dbl>             <int>       <int> <fct>
## 1 Gentoo  Biscoe           59.6          17                 230        6050 male
## 2 Chinst… Dream            58            17.8               181        3700 fema…
## 3 Gentoo  Biscoe           55.9          17                 228        5600 male
## # … with 1 more variable: year <int>
```

dplyr::slice_sample()

```
slice_sample(penguins, n = 10) # random selection
```

```
## # A tibble: 10 x 8
##    species island bill_length_mm bill_depth_mm flipper_length_… body_mass_g
##    <fct>   <fct>           <dbl>         <dbl>             <int>       <int>
##  1 Adelie  Torge…           38.6          21.2               191        3800
##  2 Adelie  Biscoe           39.7          17.7               193        3200
##  3 Chinst… Dream            50.9          19.1               196        3550
##  4 Adelie  Dream            41.5          18.5               201        4000
##  5 Adelie  Biscoe           43.2          19                 197        4775
##  6 Adelie  Torge…           38.7          19                 195        3450
##  7 Adelie  Dream            40.3          18.5               196        4350
##  8 Gentoo  Biscoe           52.2          17.1               228        5400
##  9 Adelie  Dream            39            18.7               185        3650
## 10 Chinst… Dream            53.5          19.9               205        4500
## # … with 2 more variables: sex <fct>, year <int>
```

tibble::glimpse()

```
glimpse(penguins)
```

```
## Rows: 344
## Columns: 8
## $ species           <fct> Adelie, Adelie, Adelie, Adelie, Adelie, Adelie, Ade…
## $ island            <fct> Torgersen, Torgersen, Torgersen, Torgersen, Torgers…
## $ bill_length_mm    <dbl> 39.1, 39.5, 40.3, NA, 36.7, 39.3, 38.9, 39.2, 34.1,…
## $ bill_depth_mm     <dbl> 18.7, 17.4, 18.0, NA, 19.3, 20.6, 17.8, 19.6, 18.1,…
## $ flipper_length_mm <int> 181, 186, 195, NA, 193, 190, 181, 195, 193, 190, 18…
## $ body_mass_g       <int> 3750, 3800, 3250, NA, 3450, 3650, 3625, 4675, 3475,…
## $ sex               <fct> male, female, female, NA, female, male, female, mal…
## $ year              <int> 2007, 2007, 2007, 2007, 2007, 2007, 2007, 2007, 200…
```

Quick Pause for `Logic`

Logical operators in R. `?base::Logic` - for additional details

Operator	Description	TRUE	FALSE
`<`	Less than	`3 < 5`	`100 < 1`
`<=`	Less than or equal to	`2 <= 2`	`4 <= 2`
`>`	Greater than	`5 > 3`	`1 > 100`
`>=`	Greater than or equal to	`25 >= 25.1`	`12 >= 100`
`==`	Exactly equal to	`"cat" == "cat"`	`"cat" == "dog"`
`!=`	NOT equal to	`5 != 3`	`as.character(5) != "5"`
`x %in% y`	Returns TRUE for x that are present in y	`3 %in% c(1, 2, 3)`	`3 %in% c(4:9)`
`!(x %in% y)`	Returns TRUE for NOT present in y	`!(3 %in% c(4:9))`	`!("cat" %in% c("dog", "cat", "rat"))`
`x | y`	x OR y	`5 == 3 | 3 != 2`	`"cat" == "dog" | 3 != 3`
`x & y`	x AND y	`3 == 3 & "dog" == "dog"`	`5 == 3 & 3 != 2`

dplyr::filter()

Returns rows where the logical argument is TRUE

```
# sex EQUAL to MALE
filter(penguins, species == "Adelie")
```

```
## # A tibble: 152 x 8
##    species island bill_length_mm bill_depth_mm flipper_length_… body_mass_g
##    <fct>   <fct>           <dbl>         <dbl>             <int>       <int>
##  1 Adelie  Torge…           39.1          18.7               181        3750
##  2 Adelie  Torge…           39.5          17.4               186        3800
##  3 Adelie  Torge…           40.3          18                 195        3250
##  4 Adelie  Torge…           NA            NA                  NA          NA
##  5 Adelie  Torge…           36.7          19.3               193        3450
##  6 Adelie  Torge…           39.3          20.6               190        3650
##  7 Adelie  Torge…           38.9          17.8               181        3625
##  8 Adelie  Torge…           39.2          19.6               195        4675
##  9 Adelie  Torge…           34.1          18.1               193        3475
## 10 Adelie  Torge…           42            20.2               190        4250
## # … with 142 more rows, and 2 more variables: sex <fct>, year <int>
```

dplyr::filter()

```
penguins %>%
  # species matching Adelie or Gentoo
  filter(species %in% c("Adelie", "Gentoo"))
```

```
## # A tibble: 276 x 8
##    species island bill_length_mm bill_depth_mm flipper_length_… body_mass_g
##    <fct>   <fct>           <dbl>         <dbl>             <int>       <int>
##  1 Adelie  Torge…           39.1          18.7               181        3750
##  2 Adelie  Torge…           39.5          17.4               186        3800
##  3 Adelie  Torge…           40.3          18                 195        3250
##  4 Adelie  Torge…           NA            NA                  NA          NA
##  5 Adelie  Torge…           36.7          19.3               193        3450
##  6 Adelie  Torge…           39.3          20.6               190        3650
##  7 Adelie  Torge…           38.9          17.8               181        3625
##  8 Adelie  Torge…           39.2          19.6               195        4675
##  9 Adelie  Torge…           34.1          18.1               193        3475
## 10 Adelie  Torge…           42            20.2               190        4250
## # … with 266 more rows, and 2 more variables: sex <fct>, year <int>
```

dplyr::filter()

```
penguins %>%
  # species EQUAL to Chinstrap and bill lenth greater than 53
  filter(species != "Chinstrap" & bill_length_mm >= 53)
```

```
## # A tibble: 5 x 8
##   species island bill_length_mm bill_depth_mm flipper_length_… body_mass_g sex
##   <fct>   <fct>           <dbl>         <dbl>             <int>       <int> <fct>
## 1 Gentoo  Biscoe           59.6          17                 230        6050 male
## 2 Gentoo  Biscoe           54.3          15.7               231        5650 male
## 3 Gentoo  Biscoe           55.9          17                 228        5600 male
## 4 Gentoo  Biscoe           53.4          15.8               219        5500 male
## 5 Gentoo  Biscoe           55.1          16                 230        5850 male
## # … with 1 more variable: year <int>
```

dplyr::arrange()

arrange defaults to smallest to largest

```
penguins %>%
  arrange(bill_length_mm)
```

```
## # A tibble: 344 x 8
##    species island bill_length_mm bill_depth_mm flipper_length_… body_mass_g
##    <fct>   <fct>           <dbl>         <dbl>             <int>       <int>
##  1 Adelie  Dream            32.1          15.5               188        3050
##  2 Adelie  Dream            33.1          16.1               178        2900
##  3 Adelie  Torge…           33.5          19                 190        3600
##  4 Adelie  Dream            34            17.1               185        3400
##  5 Adelie  Torge…           34.1          18.1               193        3475
##  6 Adelie  Torge…           34.4          18.4               184        3325
##  7 Adelie  Biscoe           34.5          18.1               187        2900
##  8 Adelie  Torge…           34.6          21.1               198        4400
##  9 Adelie  Torge…           34.6          17.2               189        3200
## 10 Adelie  Biscoe           35            17.9               190        3450
## # … with 334 more rows, and 2 more variables: sex <fct>, year <int>
```

dplyr::arrange()

`desc` means descending order, ie largest to smallest

```
penguins %>%
  arrange(desc(bill_length_mm))
```

```
## # A tibble: 344 x 8
##    species island bill_length_mm bill_depth_mm flipper_length_… body_mass_g
##    <fct>   <fct>           <dbl>         <dbl>             <int>       <int>
##  1 Gentoo  Biscoe           59.6          17                 230        6050
##  2 Chinst… Dream            58            17.8               181        3700
##  3 Gentoo  Biscoe           55.9          17                 228        5600
##  4 Chinst… Dream            55.8          19.8               207        4000
##  5 Gentoo  Biscoe           55.1          16                 230        5850
##  6 Gentoo  Biscoe           54.3          15.7               231        5650
##  7 Chinst… Dream            54.2          20.8               201        4300
##  8 Chinst… Dream            53.5          19.9               205        4500
##  9 Gentoo  Biscoe           53.4          15.8               219        5500
## 10 Chinst… Dream            52.8          20                 205        4550
## # … with 334 more rows, and 2 more variables: sex <fct>, year <int>
```

dplyr::arrange()

```
penguins %>%
  arrange(desc(flipper_length_mm), desc(bill_length_mm))
```

```
## # A tibble: 344 x 8
##    species island bill_length_mm bill_depth_mm flipper_length_… body_mass_g
##    <fct>   <fct>           <dbl>         <dbl>             <int>       <int>
##  1 Gentoo  Biscoe           54.3          15.7               231        5650
##  2 Gentoo  Biscoe           59.6          17                 230        6050
##  3 Gentoo  Biscoe           55.1          16                 230        5850
##  4 Gentoo  Biscoe           52.1          17                 230        5550
##  5 Gentoo  Biscoe           51.5          16.3               230        5500
##  6 Gentoo  Biscoe           50            16.3               230        5700
##  7 Gentoo  Biscoe           49.8          16.8               230        5700
##  8 Gentoo  Biscoe           48.6          16                 230        5800
##  9 Gentoo  Biscoe           49.8          15.9               229        5950
## 10 Gentoo  Biscoe           49.5          16.2               229        5800
## # … with 334 more rows, and 2 more variables: sex <fct>, year <int>
```

dplyr::select()

```
penguins %>%
  select(species, sex) %>%
  glimpse()
```

```
## Rows: 344
## Columns: 2
## $ species <fct> Adelie, Adelie, Adelie, Adelie, Adelie, Adelie, Adelie, Adeli…
## $ sex     <fct> male, female, female, NA, female, male, female, male, NA, NA,…
```

dplyr::select()

```
penguins %>%
  select(species, sex, island, body_mass_g) %>%
  glimpse()
```

```
## Rows: 344
## Columns: 4
## $ species     <fct> Adelie, Adelie, Adelie, Adelie, Adelie, Adelie, Adelie, A…
## $ sex         <fct> male, female, female, NA, female, male, female, male, NA,…
## $ island      <fct> Torgersen, Torgersen, Torgersen, Torgersen, Torgersen, To…
## $ body_mass_g <int> 3750, 3800, 3250, NA, 3450, 3650, 3625, 4675, 3475, 4250,…
```

```
penguins %>%
  select(species, sex, island, body_mass_g) %>%
  select(-island) %>%
  glimpse()
```

```
## Rows: 344
## Columns: 3
## $ species     <fct> Adelie, Adelie, Adelie, Adelie, Adelie, Adelie, Adelie, A…
## $ sex         <fct> male, female, female, NA, female, male, female, male, NA,…
## $ body_mass_g <int> 3750, 3800, 3250, NA, 3450, 3650, 3625, 4675, 3475, 4250,…
```

dplyr::select()

```
penguins %>%
  select(sex, everything()) %>%
  glimpse()
```

```
## Rows: 344
## Columns: 8
## $ sex               <fct> male, female, female, NA, female, male, female, mal…
## $ species           <fct> Adelie, Adelie, Adelie, Adelie, Adelie, Adelie, Ade…
## $ island            <fct> Torgersen, Torgersen, Torgersen, Torgersen, Torgers…
## $ bill_length_mm    <dbl> 39.1, 39.5, 40.3, NA, 36.7, 39.3, 38.9, 39.2, 34.1,…
## $ bill_depth_mm     <dbl> 18.7, 17.4, 18.0, NA, 19.3, 20.6, 17.8, 19.6, 18.1,…
## $ flipper_length_mm <int> 181, 186, 195, NA, 193, 190, 181, 195, 193, 190, 18…
## $ body_mass_g       <int> 3750, 3800, 3250, NA, 3450, 3650, 3625, 4675, 3475,…
## $ year              <int> 2007, 2007, 2007, 2007, 2007, 2007, 2007, 2007, 200…
```

dplyr::select()

```
penguins %>%
  select(starts_with("bill"), contains("flip")) %>%
  glimpse()
```

```
## Rows: 344
## Columns: 3
## $ bill_length_mm    <dbl> 39.1, 39.5, 40.3, NA, 36.7, 39.3, 38.9, 39.2, 34.1,…
## $ bill_depth_mm     <dbl> 18.7, 17.4, 18.0, NA, 19.3, 20.6, 17.8, 19.6, 18.1,…
## $ flipper_length_mm <int> 181, 186, 195, NA, 193, 190, 181, 195, 193, 190, 18…
```

dplyr::mutate()

```
penguins %>%
  select(species) %>%
  glimpse()
```

```
## Rows: 344
## Columns: 1
## $ species <fct> Adelie, Adelie, Adelie, Adelie, Adelie, Adelie, Adelie, Adeli…
```

```
penguins %>%
  mutate(species = factor(species,
                          levels = c("Adelie", "Chinstrap", "Gentoo"),
                          labels = c("AD", "CS", "GT"))) %>%
  select(species) %>%
  glimpse()
```

```
## Rows: 344
## Columns: 1
## $ species <fct> AD, AD, AD, AD, AD, AD, AD, AD, AD, AD, AD, AD, AD, AD, AD, A…
```

dplyr::mutate()

```
penguins %>%
  mutate(body_mass_kg = body_mass_g / 1000,
         body_mass = body_mass_kg * 1000) %>%
  select(body_mass_kg, body_mass_g, body_mass) %>%
  head(10)
```

```
## # A tibble: 10 x 3
##    body_mass_kg body_mass_g body_mass
##           <dbl>       <int>     <dbl>
##  1         3.75        3750      3750
##  2         3.8         3800      3800
##  3         3.25        3250      3250
##  4        NA             NA        NA
##  5         3.45        3450      3450
##  6         3.65        3650      3650
##  7         3.62        3625      3625
##  8         4.68        4675      4675
##  9         3.48        3475      3475
## 10         4.25        4250      4250
```

dplyr::group_by()

```
penguins %>%
  group_by(species)
```

```
## # A tibble: 344 x 8
## # Groups:   species [3]
##    species island bill_length_mm bill_depth_mm flipper_length_… body_mass_g
##    <fct>   <fct>           <dbl>         <dbl>             <int>       <int>
##  1 Adelie  Torge…           39.1          18.7               181        3750
##  2 Adelie  Torge…           39.5          17.4               186        3800
##  3 Adelie  Torge…           40.3          18                 195        3250
##  4 Adelie  Torge…           NA            NA                  NA          NA
##  5 Adelie  Torge…           36.7          19.3               193        3450
##  6 Adelie  Torge…           39.3          20.6               190        3650
##  7 Adelie  Torge…           38.9          17.8               181        3625
##  8 Adelie  Torge…           39.2          19.6               195        4675
##  9 Adelie  Torge…           34.1          18.1               193        3475
## 10 Adelie  Torge…           42            20.2               190        4250
## # … with 334 more rows, and 2 more variables: sex <fct>, year <int>
```

dplyr::group_by()

```
penguins %>%
  group_by(species) %>%
  slice(1)
```

```
## # A tibble: 3 x 8
## # Groups:   species [3]
##   species island bill_length_mm bill_depth_mm flipper_length_… body_mass_g sex
##   <fct>   <fct>           <dbl>         <dbl>             <int>       <int> <fct>
## 1 Adelie  Torge…            39.1          18.7               181        3750 male
## 2 Chinst… Dream             46.5          17.9               192        3500 fema…
## 3 Gentoo  Biscoe            46.1          13.2               211        4500 fema…
## # … with 1 more variable: year <int>
```

dplyr::group_by()

```
penguins %>%
  group_by(species) %>%
  slice_max(bill_length_mm, n = 1)
```

```
## # A tibble: 3 x 8
## # Groups:   species [3]
##   species island bill_length_mm bill_depth_mm flipper_length_… body_mass_g sex
##   <fct>   <fct>           <dbl>         <dbl>             <int>       <int> <fct>
## 1 Adelie  Torge…           46            21.5                194        4200 male
## 2 Chinst… Dream            58            17.8                181        3700 fema…
## 3 Gentoo  Biscoe           59.6          17                  230        6050 male
## # … with 1 more variable: year <int>
```

dplyr::group_by()

```
penguins %>%
  group_by(species) %>%
  arrange(desc(bill_length_mm)) %>%
  slice(1)
```

```
## # A tibble: 3 x 8
## # Groups:   species [3]
##   species island bill_length_mm bill_depth_mm flipper_length_… body_mass_g sex
##   <fct>   <fct>           <dbl>         <dbl>             <int>       <int> <fct>
## 1 Adelie  Torge…           46            21.5                194        4200 male
## 2 Chinst… Dream            58            17.8                181        3700 fema…
## 3 Gentoo  Biscoe           59.6          17                  230        6050 male
## # … with 1 more variable: year <int>
```

dplyr::group_by()

```
penguins %>%
  group_by(species, island) %>%
  count()
```

```
## # A tibble: 5 x 3
## # Groups:   species, island [5]
##   species   island        n
##   <fct>     <fct>     <int>
## 1 Adelie    Biscoe       44
## 2 Adelie    Dream        56
## 3 Adelie    Torgersen    52
## 4 Chinstrap Dream        68
## 5 Gentoo    Biscoe      124
```

dplyr::summarize()

```
penguins %>%
  summarize(mean = mean(body_mass_g))
```

```
## # A tibble: 1 x 1
##    mean
##   <dbl>
## 1    NA
```

```
penguins %>%
  summarize(mean = mean(body_mass_g, na.rm = TRUE))
```

```
## # A tibble: 1 x 1
##    mean
##   <dbl>
## 1 4202.
```

dplyr::summarize()

```
penguins %>%
  summarize(median(body_mass_g, na.rm = TRUE))
```

```
## # A tibble: 1 x 1
##   `median(body_mass_g, na.rm = TRUE)`
##                                 <dbl>
## 1                                4050
```

```
penguins %>%
  summarize(median_mass= median(body_mass_g, na.rm = TRUE))
```

```
## # A tibble: 1 x 1
##   median_mass
##         <dbl>
## 1        4050
```

dplyr::summarize()

```
penguins %>%
  group_by(species) %>%
  summarize(mean_mass = mean(body_mass_g, na.rm = TRUE),
            sd_mass = sd(body_mass_g, na.rm = TRUE),
            n = n())
```

```
## `summarise()` ungrouping output (override with `.groups` argument)

## # A tibble: 3 x 4
##   species   mean_mass sd_mass     n
##   <fct>         <dbl>   <dbl> <int>
## 1 Adelie        3701.    459.   152
## 2 Chinstrap     3733.    384.    68
## 3 Gentoo        5076.    504.   124
```

dplyr::mutate() + across()

```
penguins %>%
  mutate(across(c(species, island), as.character)) %>%
  select(species, island) %>%
  glimpse()
```

```
## Rows: 344
## Columns: 2
## $ species <chr> "Adelie", "Adelie", "Adelie", "Adelie", "Adelie", "Adelie", "…
## $ island  <chr> "Torgersen", "Torgersen", "Torgersen", "Torgersen", "Torgerse…
```

```
penguins %>%
  select(species, island) %>%
  glimpse()
```

```
## Rows: 344
## Columns: 2
## $ species <fct> Adelie, Adelie, Adelie, Adelie, Adelie, Adelie, Adelie, Adeli…
## $ island  <fct> Torgersen, Torgersen, Torgersen, Torgersen, Torgersen, Torger…
```

dplyr::summarize() + across

```
penguins %>%
  group_by(species) %>%
  summarize(
    across(c(body_mass_g, bill_length_mm), mean, na.rm = TRUE),
    n = n()
    )
```

```
## `summarise()` ungrouping output (override with `.groups` argument)

## # A tibble: 3 x 4
##   species   body_mass_g bill_length_mm     n
##   <fct>           <dbl>          <dbl> <int>
## 1 Adelie          3701.           38.8   152
## 2 Chinstrap       3733.           48.8    68
## 3 Gentoo          5076.           47.5   124
```

dplyr::summarize() + across

```
penguins %>%
  group_by(species) %>%
  summarize(
    across(c(body_mass_g, bill_length_mm, bill_depth_mm),
           list(
             mean = ~mean(.x, na.rm = TRUE),
             sd = ~sd(.x, na.rm = TRUE))
           ),
    n = n()
    )
```

```
## `summarise()` ungrouping output (override with `.groups` argument)

## # A tibble: 3 x 8
##   species body_mass_g_mean body_mass_g_sd bill_length_mm_… bill_length_mm_…
##   <fct>              <dbl>          <dbl>             <dbl>             <dbl>
## 1 Adelie             3701.           459.              38.8              2.66
## 2 Chinst…            3733.           384.              48.8              3.34
## 3 Gentoo             5076.           504.              47.5              3.08
## # … with 3 more variables: bill_depth_mm_mean <dbl>, bill_depth_mm_sd <dbl>,
## #   n <int>
```

tidyr

The goal of tidyr is to help you create tidy data. Tidy data is data where:

- Each variable is in a column.
- Each observation is a row.
- Each value is a cell.

Make Taller and Make Wider

- `pivot_longer()` - "lengthens" data, increasing the number of rows and decreasing the number of columns.
- `pivot_wider()` - "widens" data, increases the number of columns and decreasing the number of rows.

Separate and unite columns

- `separate()` - Separate one column into multiple columns.
- `unite()` - Unite multiple columns into one.

Tidy the data

```
untidy_df
```

```
## # A tibble: 5 x 7
##   age_group male_2016 female_2016 male_2017 female_2017 male_2018 female_2018
##   <chr>         <dbl>       <dbl>     <dbl>       <dbl>     <dbl>       <dbl>
## 1 < 18          22000       20000     22000       20000     22000       20000
## 2 18-30         36000       35000     36000       35000     36000       35000
## 3 31-50         50000       40000     50000       40000     50000       40000
## 4 51-60         62000       60000     62000       60000     62000       60000
## 5 > 60          75000       72000     75000       72000     75000       72000
```

Tidy the data

```
untidy_df %>%
  pivot_longer(cols = male_2016:female_2018,
               names_to = "gender_year",
               values_to = "income")
```

```
## # A tibble: 30 x 3
##    age_group gender_year income
##    <chr>     <chr>         <dbl>
##  1 < 18      male_2016     22000
##  2 < 18      female_2016   20000
##  3 < 18      male_2017     22000
##  4 < 18      female_2017   20000
##  5 < 18      male_2018     22000
##  6 < 18      female_2018   20000
##  7 18-30     male_2016     36000
##  8 18-30     female_2016   35000
##  9 18-30     male_2017     36000
## 10 18-30     female_2017   35000
## # … with 20 more rows
```

Tidy the data

```
untidy_df %>%
  pivot_longer(cols = male_2016:female_2018,
               names_to = "gender_year",
               values_to = "income")  %>%
  separate(gender_year, into = c("gender", "year"))
```

```
## # A tibble: 30 x 4
##    age_group gender year  income
##    <chr>     <chr>  <chr>  <dbl>
##  1 < 18      male   2016   22000
##  2 < 18      female 2016   20000
##  3 < 18      male   2017   22000
##  4 < 18      female 2017   20000
##  5 < 18      male   2018   22000
##  6 < 18      female 2018   20000
##  7 18-30     male   2016   36000
##  8 18-30     female 2016   35000
##  9 18-30     male   2017   36000
## 10 18-30     female 2017   35000
## # … with 20 more rows
```

Tidy the data

```
untidy_df %>%
  pivot_longer(
    cols = male_2016:female_2018,
    names_to = c("gender", "year"),
    names_pattern = "(.*)_(.*)",
    values_to = "income"
  )
```

```
## # A tibble: 30 x 4
##    age_group gender year  income
##    <chr>     <chr>  <chr>  <dbl>
##  1 < 18      male   2016   22000
##  2 < 18      female 2016   20000
##  3 < 18      male   2017   22000
##  4 < 18      female 2017   20000
##  5 < 18      male   2018   22000
##  6 < 18      female 2018   20000
##  7 18-30     male   2016   36000
##  8 18-30     female 2016   35000
##  9 18-30     male   2017   36000
## 10 18-30     female 2017   35000
## # … with 20 more rows
```

Untidy the data

```
tidy_df %>%
  unite("gender_year", c("gender", "year"), sep = "_")
```

```
## # A tibble: 30 x 3
##    age_group gender_year income
##    <chr>     <chr>        <dbl>
##  1 < 18      male_2016    22000
##  2 18-30     male_2016    36000
##  3 31-50     male_2016    50000
##  4 51-60     male_2016    62000
##  5 > 60      male_2016    75000
##  6 < 18      female_2016  20000
##  7 18-30     female_2016  35000
##  8 31-50     female_2016  40000
##  9 51-60     female_2016  60000
## 10 > 60      female_2016  72000
## # … with 20 more rows
```

Untidy the data

```
tidy_df %>%
  unite("gender_year", c("gender", "year"), sep = "_") %>%
  pivot_wider(names_from = gender_year, values_from = income)
```

```
## # A tibble: 5 x 7
##   age_group male_2016 female_2016 male_2017 female_2017 male_2018 female_2018
##   <chr>         <dbl>       <dbl>     <dbl>       <dbl>     <dbl>       <dbl>
## 1 < 18          22000       20000     22000       20000     22000       20000
## 2 18-30         36000       35000     36000       35000     36000       35000
## 3 31-50         50000       40000     50000       40000     50000       40000
## 4 51-60         62000       60000     62000       60000     62000       60000
## 5 > 60          75000       72000     75000       72000     75000       72000
```

Untidy the data

```
tidy_df %>%
  pivot_wider(names_from = c(gender, year), values_from = income)
```

```
## # A tibble: 5 x 7
##   age_group male_2016 female_2016 male_2017 female_2017 male_2018 female_2018
##   <chr>         <dbl>       <dbl>     <dbl>       <dbl>     <dbl>       <dbl>
## 1 < 18          22000       20000     22000       20000     22000       20000
## 2 18-30         36000       35000     36000       35000     36000       35000
## 3 31-50         50000       40000     50000       40000     50000       40000
## 4 51-60         62000       60000     62000       60000     62000       60000
## 5 > 60          75000       72000     75000       72000     75000       72000
```

ggplot2

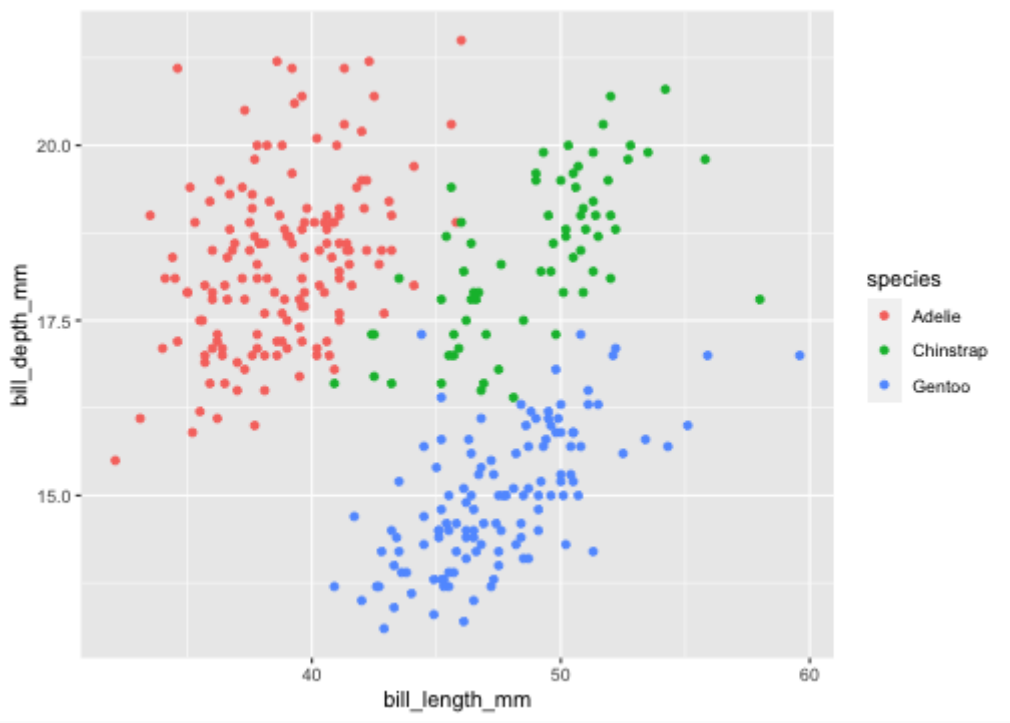
3 Core parts

- `ggplot` - builds base layer
- `geom_` is the shape
    - `geom_point()`
    - `geom_line()`
    - `geom_bar()`
    - `geom_boxplot()`
    - `geom_?()`

- `aes` is the mappings/relationships
    - Horizontal Dimensions (x)
    - Vertical Dimensions (y)
    - Color
    - Shape
    - Size
    - Transparency
    - Relationships

```
ggplot(data = <DATA>, mapping = aes(<MAPPINGS>)) +
  <GEOM_FUNCTION>()
```

```
ggplot(penguins, aes(x = bill_length_mm, y = bill_depth_mm, color = species)) +
  geom_point()
```



Supply the data, tell `ggplot2` the aesthetic mappings, and then add layers of `plots` via `geom_`

- `geom_point()` - Points
- `geom_dotplot()` - Dot plot
- `geom_hline()` - Horizontal reference line
- `geom_vline()` - Vertical reference line
- `geom_boxplot()` - A box and whisker plot
- `geom_density()` - Smoothed density estimates
- `geom_errorbarh()` - Horizontal error bars
- `geom_hex()` - Hexagonal heatmap of 2d bin counts
- `geom_jitter()` - Jittered points
- `geom_linerange()` - Vertical interval line
- `geom_pointrange()` - Vertical point line
- `geom_line()` - Connect observations line
- `geom_step()` - Connect observations via step lines
- `geom_polygon()` - Polygons
- `geom_segment()` - Line segment
- `geom_ribbon()` - Ribbon plot
- `geom_area()` - Area plot
- `geom_rug()` - Rug plots in the margins
- `geom_smooth()` - Smoothed conditional means
- `geom_label()` - Label points with text
- `geom_text()` - Add text
- `geom_violin()` - Violin plot
- `geom_sf()` - Visual sf objects
- `geom_map()` - Plot map
- `geom_qq_line()` - A quantile-quantile plot
- `geom_histogram()` - Histogram plot

ggplot2

Gapminder dataset

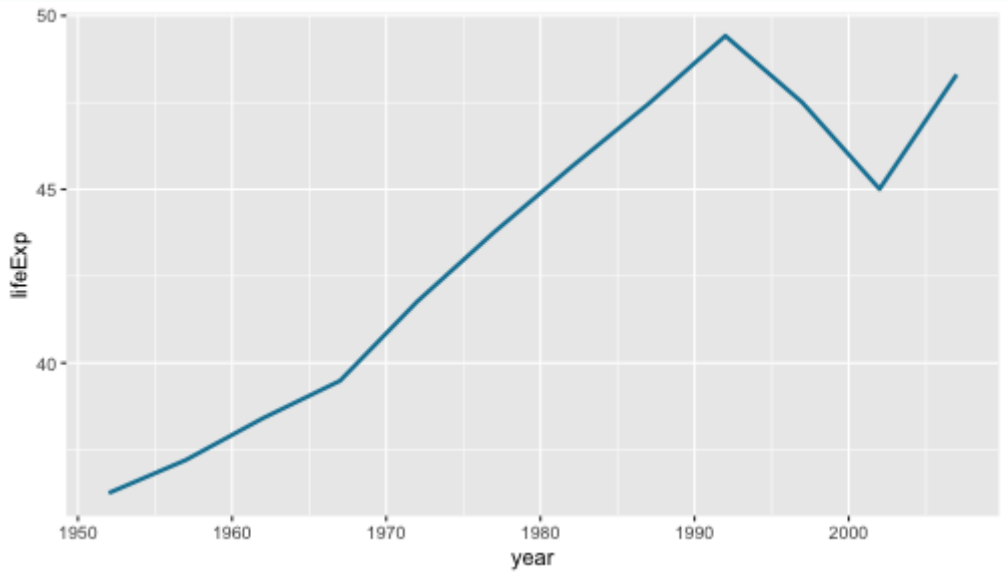
Excerpt from the Gapminder data. For each of 142 countries, the data provides values for life expectancy, GDP per capita, and population, every five years, from 1952 to 2007.

```
gapminder_df <- gapminder::gapminder

glimpse(gapminder_df)
```

```
## Rows: 1,704
## Columns: 6
## $ country   <fct> Afghanistan, Afghanistan, Afghanistan, Afghanistan, Afghani…
## $ continent <fct> Asia, Asia, Asia, Asia, Asia, Asia, Asia, Asia, Asia, Asia,…
## $ year      <int> 1952, 1957, 1962, 1967, 1972, 1977, 1982, 1987, 1992, 1997,…
## $ lifeExp   <dbl> 28.801, 30.332, 31.997, 34.020, 36.088, 38.438, 39.854, 40.…
## $ pop       <int> 8425333, 9240934, 10267083, 11537966, 13079460, 14880372, 1…
## $ gdpPercap <dbl> 779.4453, 820.8530, 853.1007, 836.1971, 739.9811, 786.1134,…
```

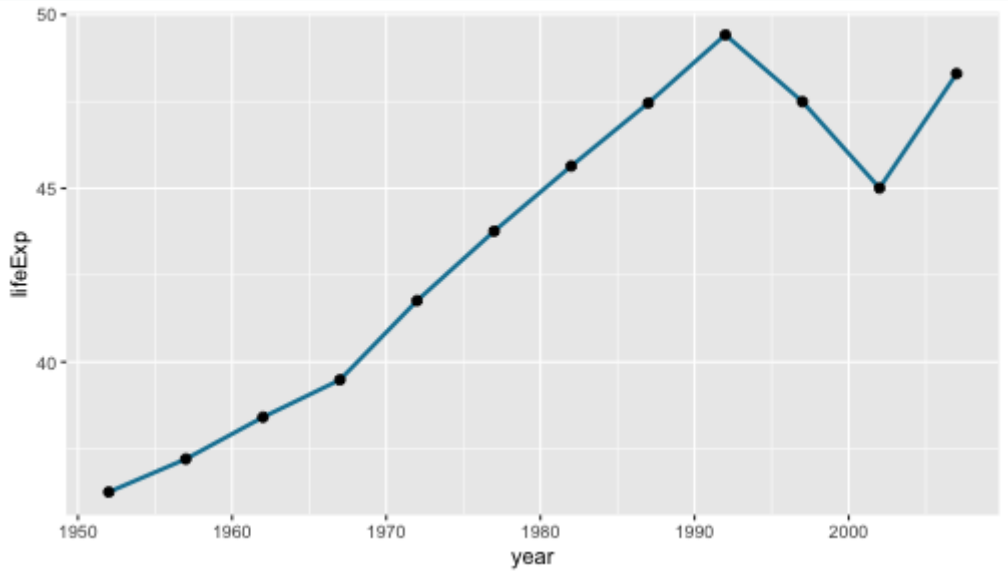
ggplot2

```
gapminder_df %>%
  filter(country == "Malawi") %>%
  ggplot(aes(x = year, y = lifeExp)) +
  geom_line(colour = "#1380A1", size = 1)
```



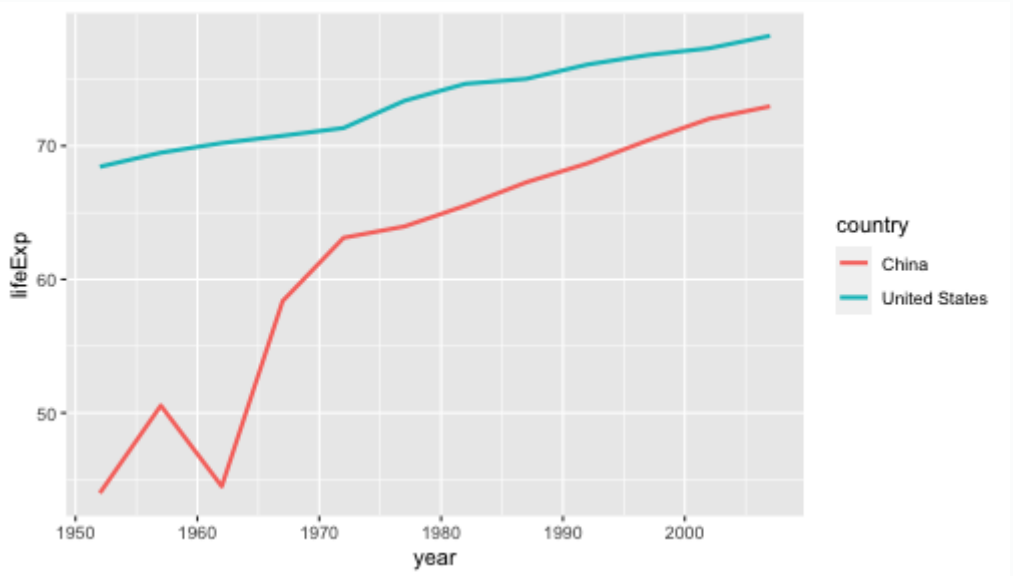
ggplot2

```
gapminder_df %>%
  filter(country == "Malawi") %>%
  ggplot(aes(x = year, y = lifeExp)) +
  geom_line(colour = "#1380A1", size = 1) +
  geom_point(size = 2)
```



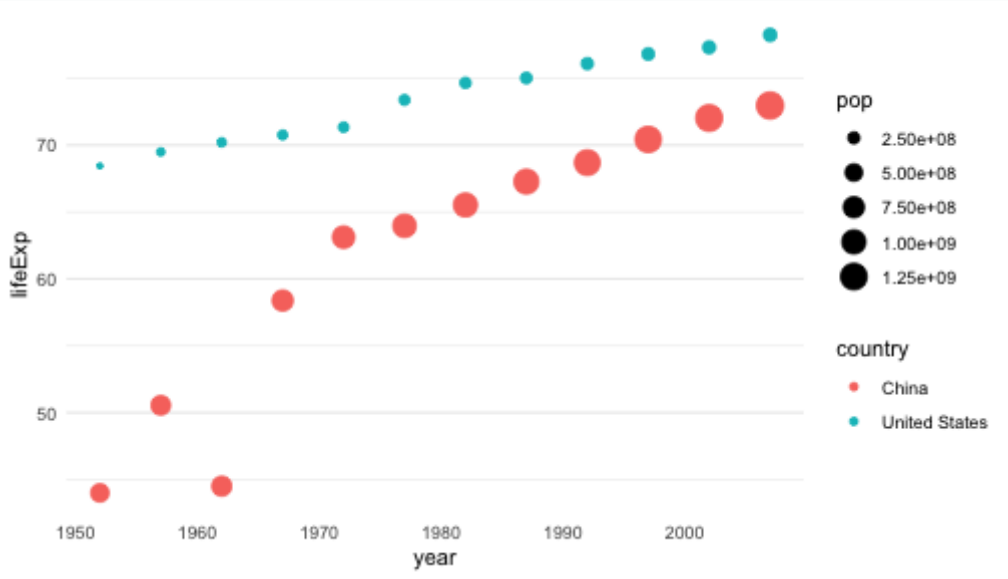
ggplot2

```
gapminder_df %>%
  filter(country == "China" | country == "United States") %>%
  ggplot(aes(x = year, y = lifeExp, colour = country)) +
  geom_line(size = 1)
```



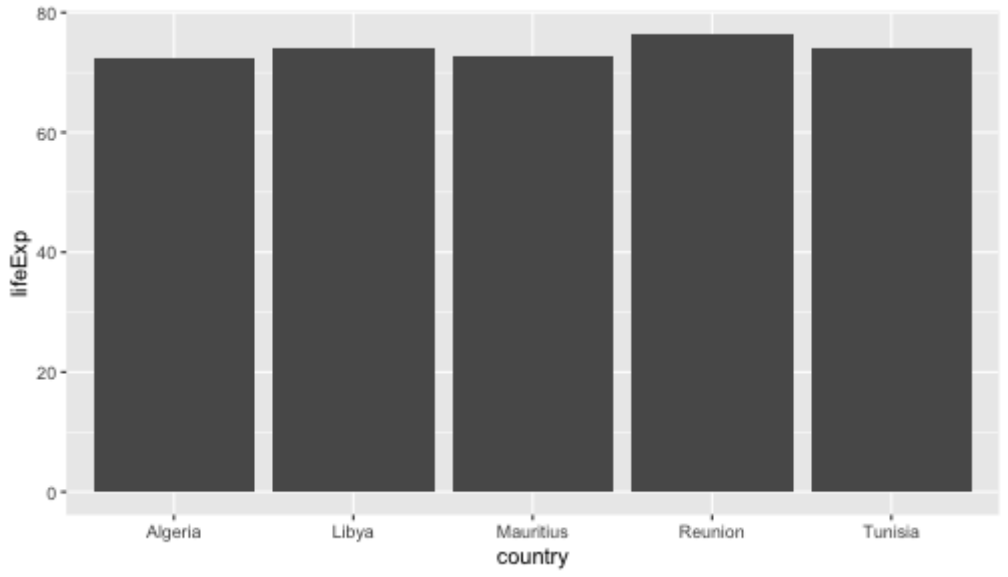
ggplot2

```
gapminder_df %>%
  filter(country == "China" | country == "United States") %>%
  ggplot(aes(x = year, y = lifeExp, colour = country,
             size = pop)) +
  geom_point() +
  theme_minimal() +
  theme(panel.grid.major.x = element_blank(),
        panel.grid.minor.x = element_blank())
```



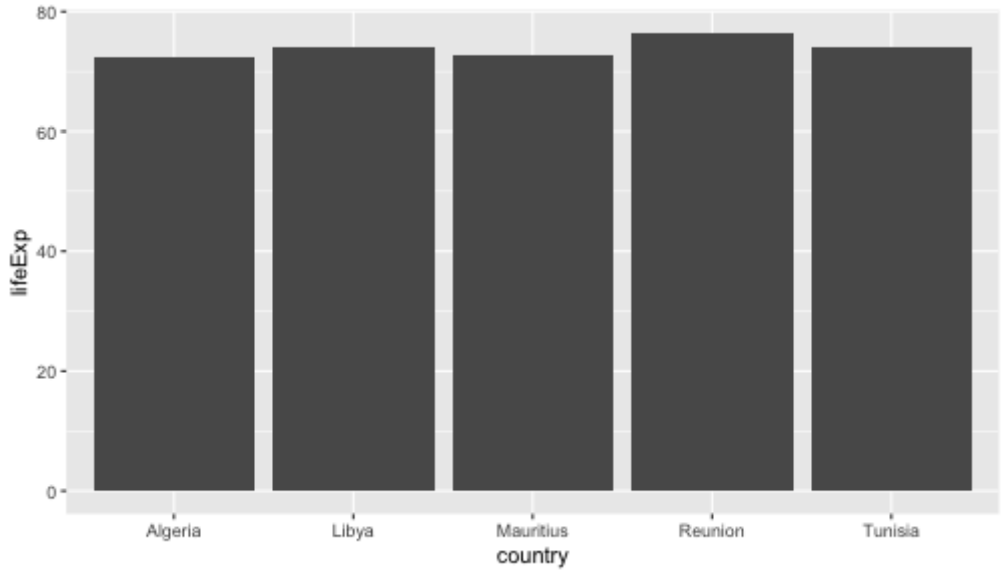
ggplot2

```
bar_df <- gapminder_df %>%
  filter(year == 2007 & continent == "Africa") %>%
  arrange(desc(lifeExp)) %>%
  head(5)

(bars <- ggplot(bar_df, aes(x = country, y = lifeExp)) +
  geom_bar(stat = "identity"))
```



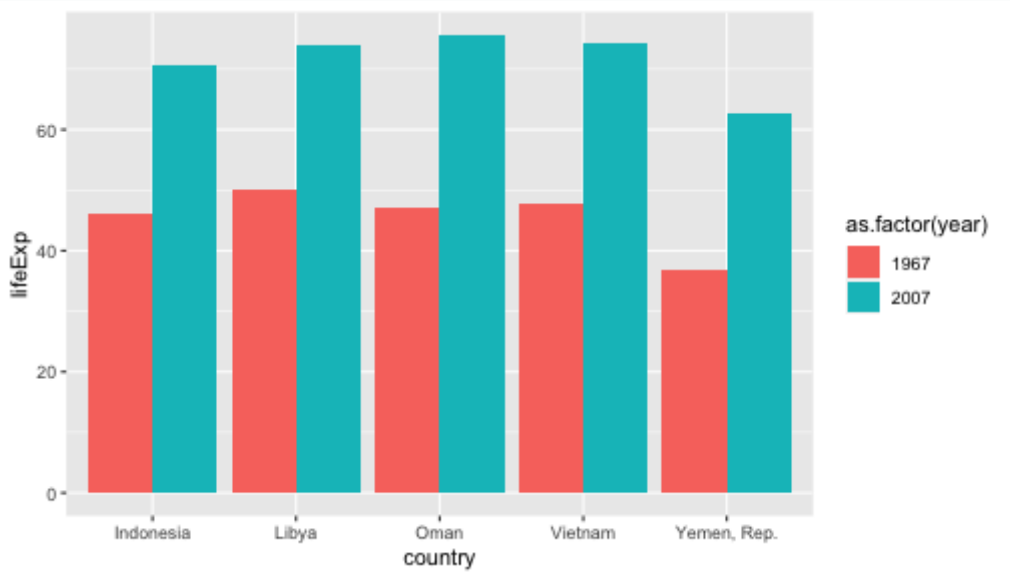
ggplot2

```
bar_df <- gapminder_df %>%
  filter(year == 2007 & continent == "Africa") %>%
  arrange(desc(lifeExp)) %>%
  head(5)

(bars <- ggplot(bar_df, aes(x = country, y = lifeExp)) +
  geom_col())
```



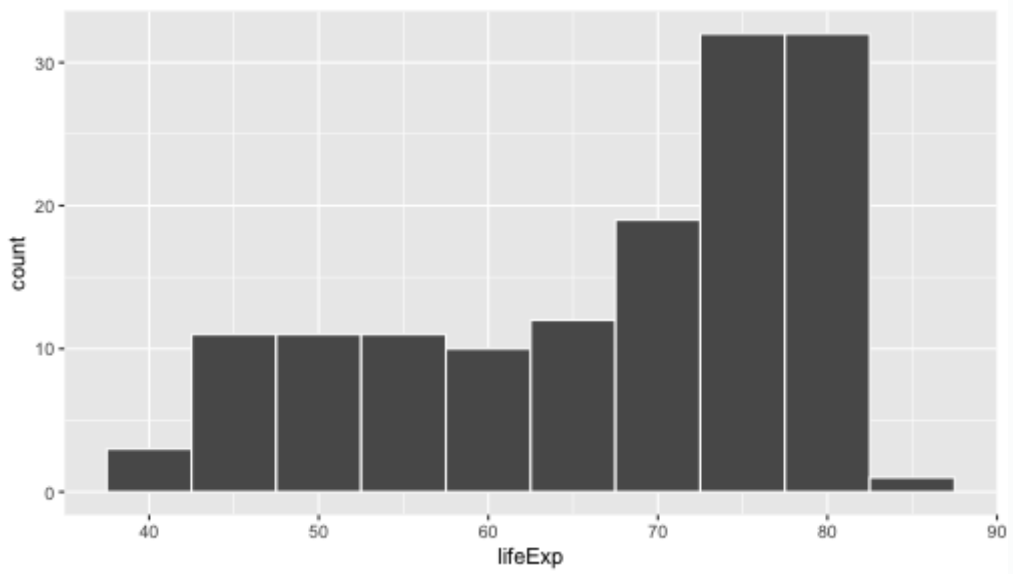
ggplot2

```
grouped_bars <- ggplot(grouped_bar_df,
                       aes(x = country,
                           y = lifeExp,
                           fill = as.factor(year))) +
  geom_bar(stat="identity", position="dodge")

grouped_bars
```



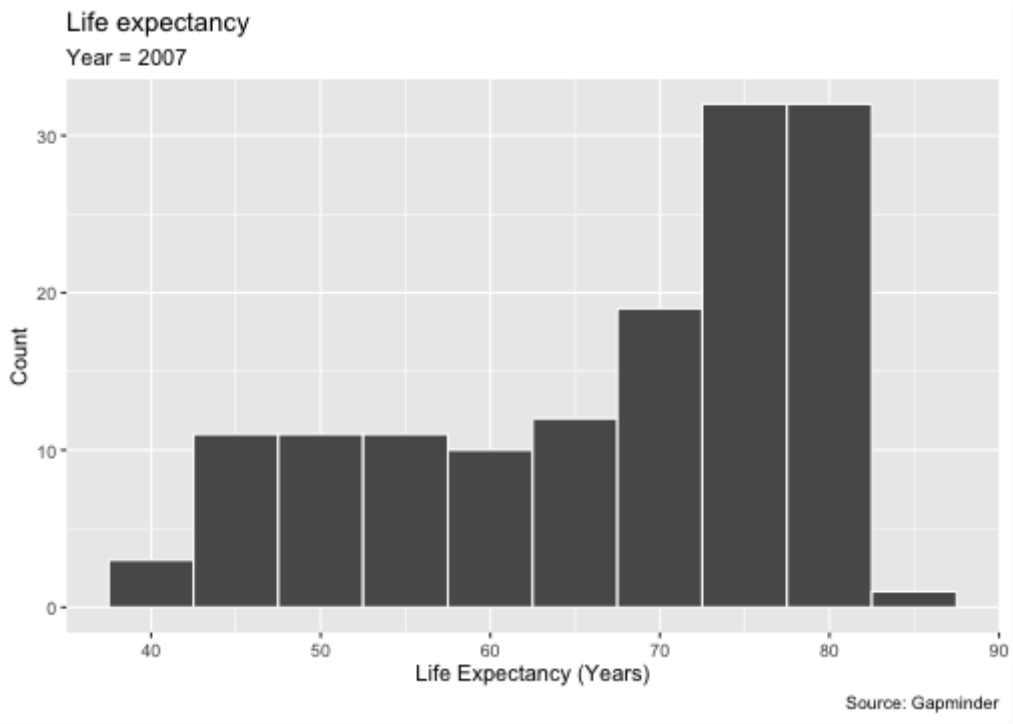
ggplot2

```
hist_plot <- gapminder_df %>%
  filter(year == 2007) %>%
  ggplot(aes(x = lifeExp)) +
  geom_histogram(binwidth = 5, color = "white")

hist_plot
```



ggplot2

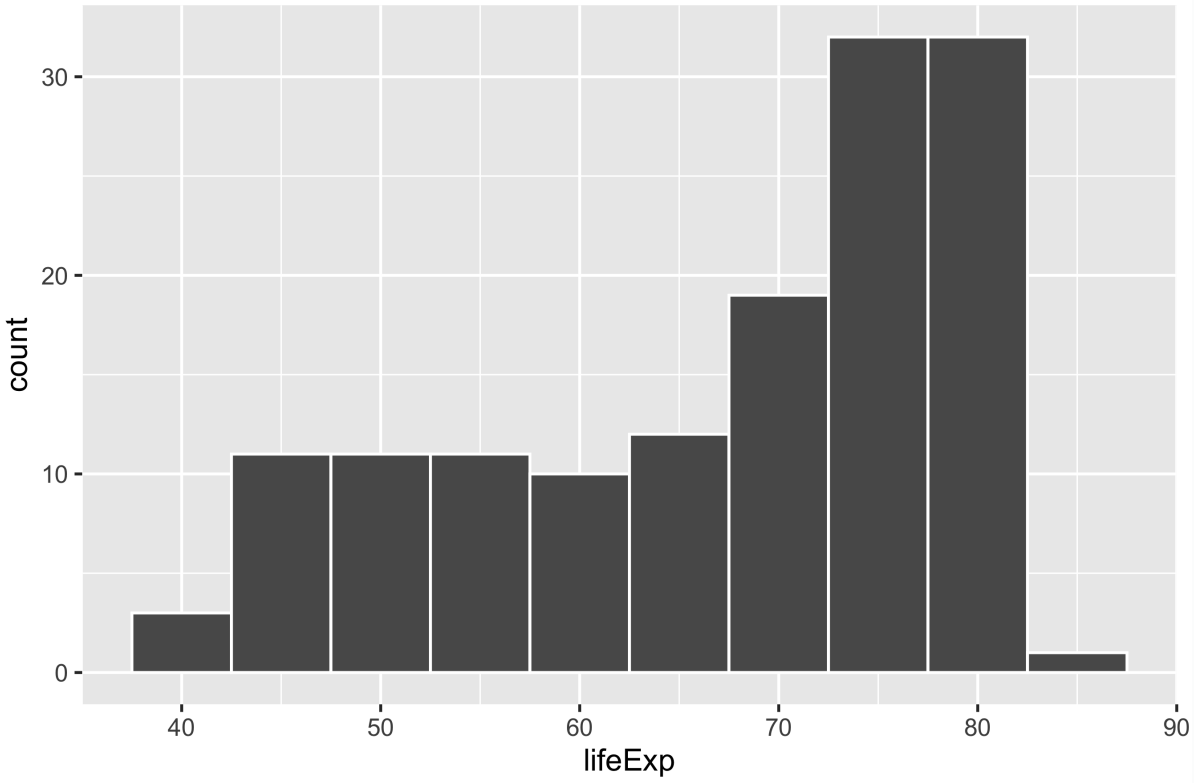
```
hist_plot +
  labs(x = "Life Expectancy (Years)",
       y = "Count",
       title = "Life expectancy",
       subtitle = "Year = 2007",
       caption = "Source: Gapminder")
```



ggplot2

```
ggsave("life_exp_2007.png", hist_plot,
       height = 4, width = 6, units = "in", dpi = 450)
```

```
knitr::include_graphics("life_exp_2007.png")
```



ggplot2

Unlimited customization via theme!

```
theme(line, rect, text, title, aspect.ratio, axis.title, axis.title.x,
  axis.title.x.top, axis.title.x.bottom, axis.title.y, axis.title.y.left,
  axis.title.y.right, axis.text, axis.text.x, axis.text.x.top,
  axis.text.x.bottom, axis.text.y, axis.text.y.left, axis.text.y.right,
  axis.ticks, axis.ticks.x, axis.ticks.x.top, axis.ticks.x.bottom,
  axis.ticks.y, axis.ticks.y.left, axis.ticks.y.right, axis.ticks.length,
  axis.ticks.length.x, axis.ticks.length.x.top, axis.ticks.length.x.bottom,
  axis.ticks.length.y, axis.ticks.length.y.left, axis.ticks.length.y.right,
  axis.line, axis.line.x, axis.line.x.top, axis.line.x.bottom, axis.line.y,
  axis.line.y.left, axis.line.y.right, legend.background, legend.margin,
  legend.spacing, legend.spacing.x, legend.spacing.y, legend.key,
  legend.key.size, legend.key.height, legend.key.width, legend.text,
  legend.text.align, legend.title, legend.title.align, legend.position,
  legend.direction, legend.justification, legend.box, legend.box.just,
  legend.box.margin, legend.box.background, legend.box.spacing,
  panel.background, panel.border, panel.spacing, panel.spacing.x,
  panel.spacing.y, panel.grid, panel.grid.major, panel.grid.minor,
  panel.grid.major.x, panel.grid.major.y, panel.grid.minor.x,
  panel.grid.minor.y, panel.ontop, plot.background, plot.title,
  plot.subtitle, plot.caption, plot.tag, plot.tag.position, plot.margin,
  strip.background, strip.background.x, strip.background.y,
  strip.placement, strip.text, strip.text.x, strip.text.y,
```

ggplot2

Saved themes apply multiple theme elements all at once

- Saves writing
- Reproducibility

Built in themes

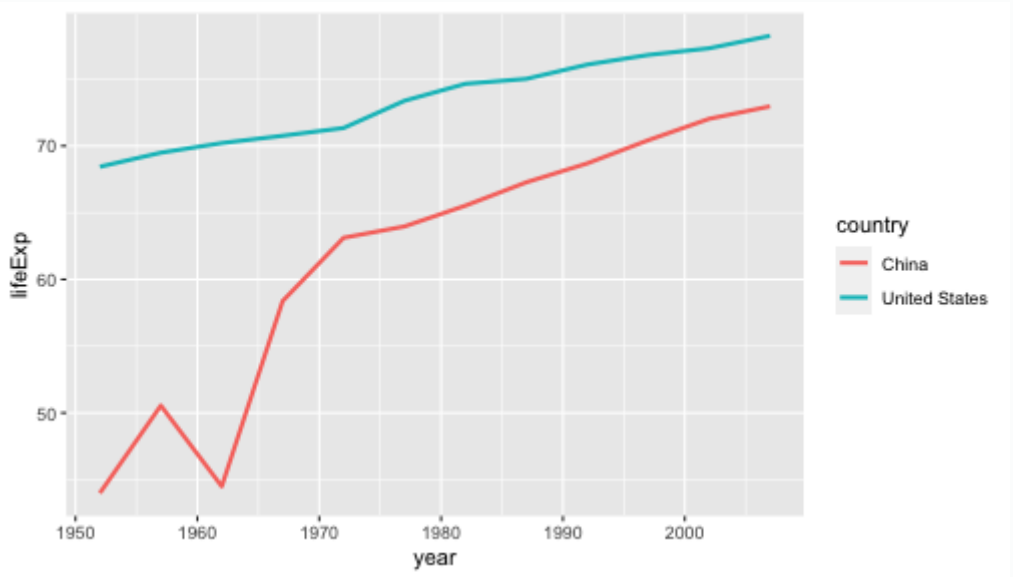
- `theme_grey()` # default
- `theme_bw()`
- `theme_minimal()`
- `theme_classic()`

New packages

- `bbplot`
  - `bbc_style()`
- `urbnthemes()`
  - `theme_urbn_print()`
- `ggthemes`
  - `theme_few()`
  - `theme_excel()`
  - `theme_economist()`
- Build your own package/theme!

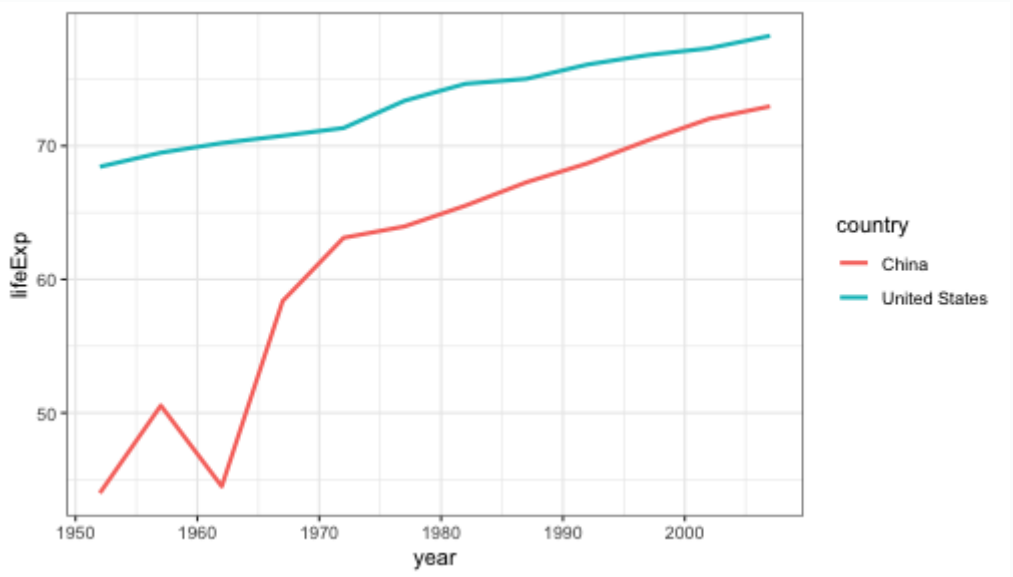
ggplot2

```
gapminder_df %>%
  filter(country == "China" | country == "United States") %>%
  ggplot(aes(x = year, y = lifeExp, colour = country)) +
  geom_line(size = 1) +
  theme_grey()
```



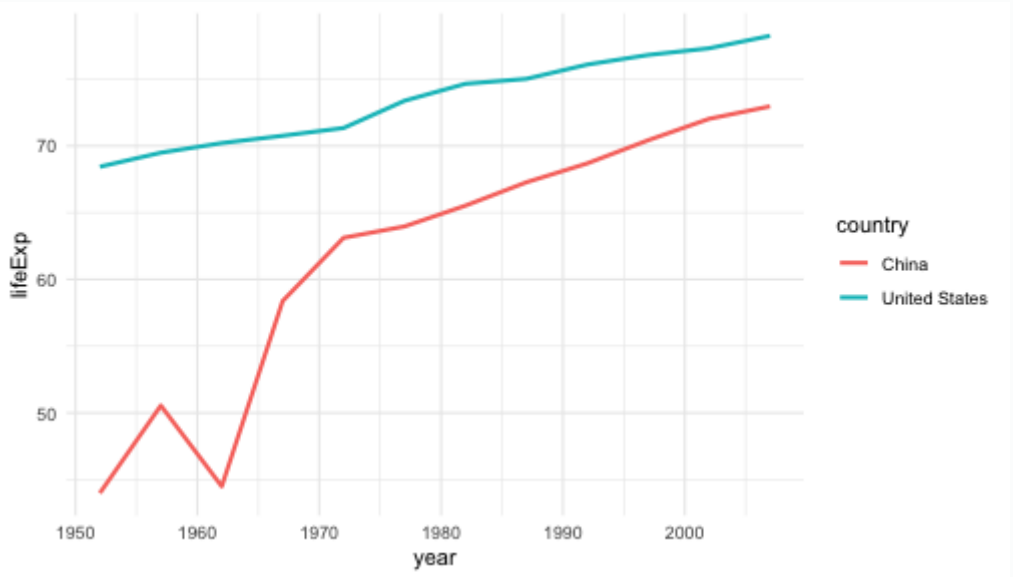
ggplot2

```
gapminder_df %>%
  filter(country == "China" | country == "United States") %>%
  ggplot(aes(x = year, y = lifeExp, colour = country)) +
  geom_line(size = 1) +
  theme_bw()
```



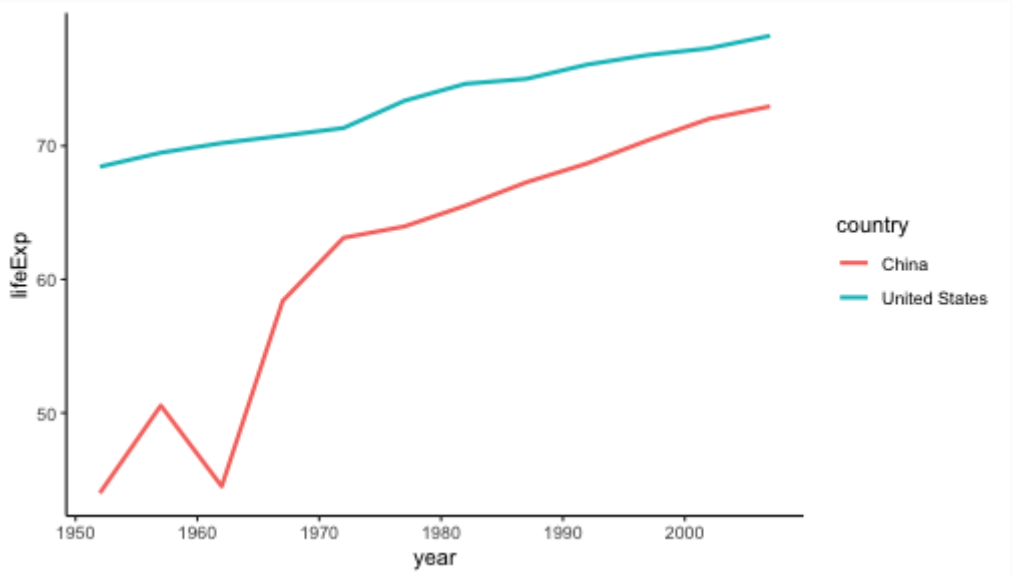
ggplot2

```
gapminder_df %>%
  filter(country == "China" | country == "United States") %>%
  ggplot(aes(x = year, y = lifeExp, colour = country)) +
  geom_line(size = 1) +
  theme_minimal()
```



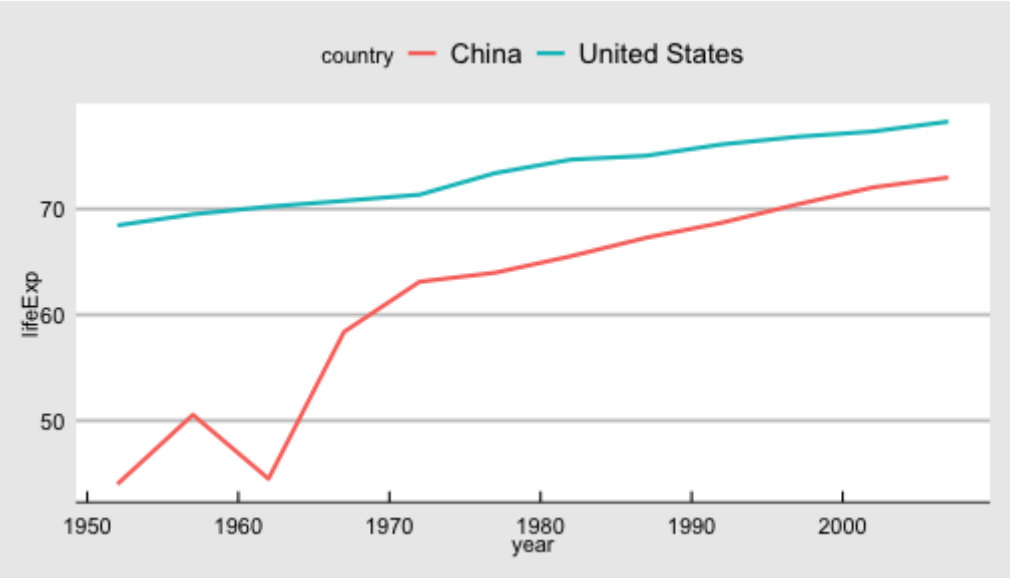
ggplot2

```
gapminder_df %>%
  filter(country == "China" | country == "United States") %>%
  ggplot(aes(x = year, y = lifeExp, colour = country)) +
  geom_line(size = 1) +
  theme_classic()
```



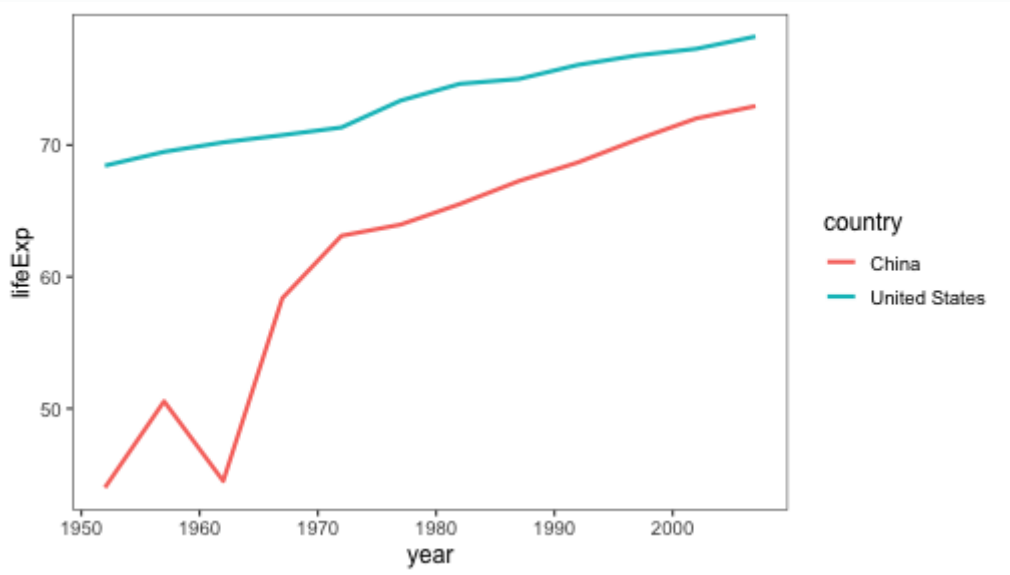
ggplot2

```
gapminder_df %>%
  filter(country == "China" | country == "United States") %>%
  ggplot(aes(x = year, y = lifeExp, colour = country)) +
  geom_line(size = 1) +
  ggthemes::theme_economist_white()
```



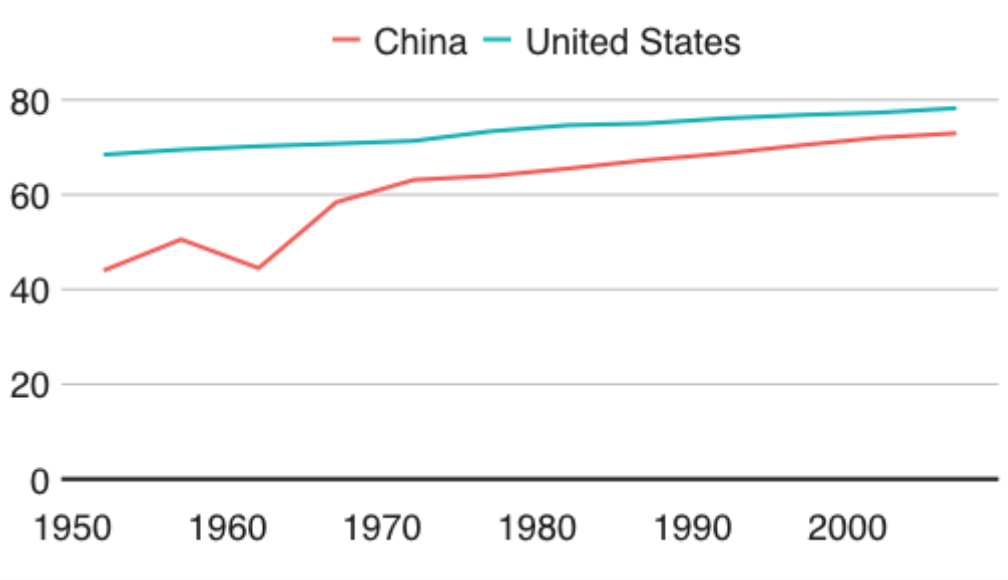
ggplot2

```
gapminder_df %>%
  filter(country == "China" | country == "United States") %>%
  ggplot(aes(x = year, y = lifeExp, colour = country)) +
  geom_line(size = 1) +
  ggthemes::theme_few()
```



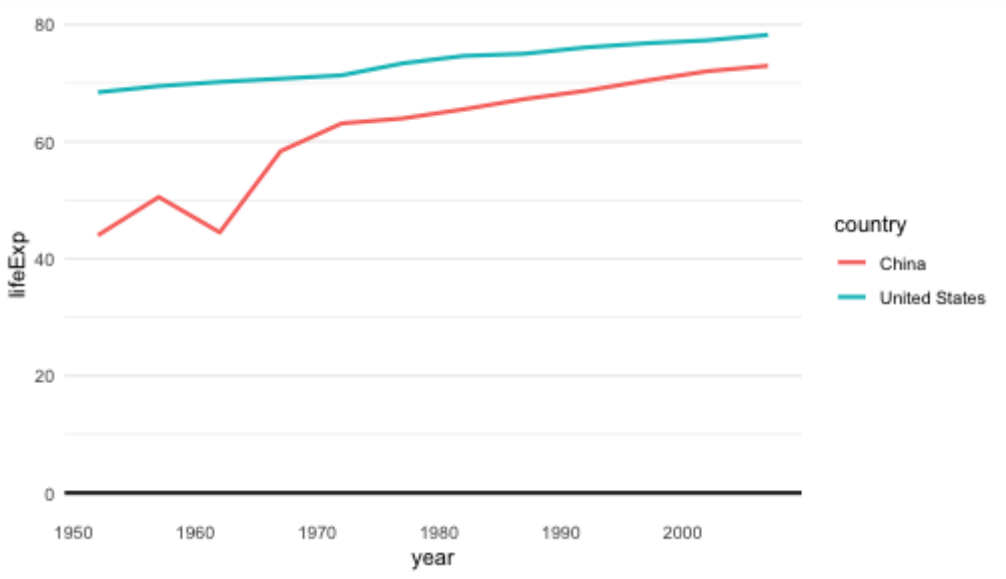
ggplot2

```
gapminder_df %>%
  filter(country == "China" | country == "United States") %>%
  ggplot(aes(x = year, y = lifeExp, colour = country)) +
  geom_line(size = 1) +
  geom_hline(yintercept = 0, size = 1, colour="#333333") +
  bbplot::bbc_style()
```



ggplot2

```
gapminder_df %>%
  filter(country == "China" | country == "United States") %>%
  ggplot(aes(x = year, y = lifeExp, colour = country)) +
  geom_line(size = 1) +
  geom_hline(yintercept = 0, size = 1, colour="#333333") +
  theme_minimal() +
  theme(panel.grid.major.x = element_blank(),
        panel.grid.minor.x = element_blank())
```



Whirlwind of information!

`tidyverse` focuses on `tidy` data

- `Tidyverse`
  - Read data in with `readr`
  - Tidy data with `tidyr`
  - Transform data with `dplyr`
  - Plot data with `ggplot2`
- Next Steps
  - R for Data Science book (free!)
  - RStudio Cloud Primers (free!)