Tidy Data

Tidy data

A data set is **tidy** iff:

1. Each **variable** is in its own **column**
2. Each **case** is in its own **row**
3. Each **value** is in its own **cell**

Adapted from 'Master the tidyverse' CC by RStudio

`table1` is tidy

country <chr>	year <int>	cases <int>	population <int>
Afghanistan	1999	745	19987071
Afghanistan	2000	2666	20595360
Brazil	1999	37737	172006362
Brazil	2000	80488	174504898
China	1999	212258	1272915272
China	2000	213766	1280428583

6 rows

Adapted from 'Master the tidyverse' CC by RStudio

`table1` is tidy

country <chr>	year <int>	cases <int>	population <int>	rate <dbl>
Afghanistan	1999	745	19987071	0.0000372741
Afghanistan	2000	2666	20595360	0.0001294466
Brazil	1999	37737	172006362	0.0002193930
Brazil	2000	80488	174504898	0.0004612363
China	1999	212258	1272915272	0.0001667495
China	2000	213766	1280428583	0.0001669488

6 rows

```
table1 %>%
  mutate(rate = cases/population)
```

Adapted from "Master the tidyverse" CC by RStudio

`table2` isn't tidy

contains two variables

country <chr>	year <int>	type <chr>	count <int>
Afghanistan	1999	cases	745
Afghanistan	1999	population	19987071
Afghanistan	2000	cases	2666
Afghanistan	2000	population	20595360
Brazil	1999	cases	37737
Brazil	1999	population	172006362
Brazil	2000	cases	80488
Brazil	2000	population	174504898
China	1999	cases	212258
China	1999	population	1272915272

1-10 of 12 rows

Previou

It's hard to manipulate

Adapted from 'Master the tidyverse' CC by RStudio

Your Turn 1

Is `bp_systolic` tidy?

subject_id <dbl>	time_1 <dbl>	time_2 <dbl>	time_3 <dbl>
1	120	118	121
2	125	131	NA
3	141	NA	NA

3 rows

Adapted from 'Master the tidyverse' CC by RStudio

Your Turn 1

Is `bp_systolic` tidy?

subject_id <dbl>	time_1 <dbl>	time_2 <dbl>	time_3 <dbl>
1	120	118	121
2	125	131	NA
3	141	NA	NA

3 rows

Variables:

- subject
- time
- systolic blood pressure

Adapted from 'Master the tidyverse' CC by RStudio

`bp_systolic2` is tidy

subject_id <dbl>	time <dbl>	systolic <dbl>
1	1	120
1	2	118
1	3	121
2	1	125
2	2	131
3	1	141

6 rows

Adapted from 'Master the tidyverse' CC by RStudio

Your Turn 2

05-Tidy-data.Rmd

Using `bp_systolic2` with `group_by()`, and `summarise()`

- Find the average systolic blood pressure for each subject
- Find the last time each subject was measured

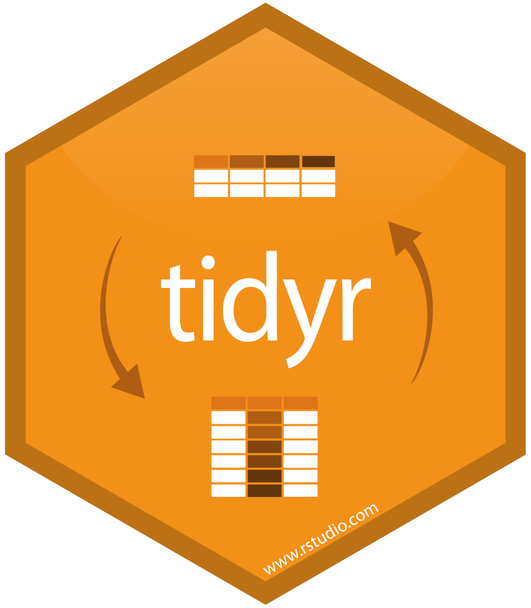
Adapted from 'Master the tidyverse' CC by RStudio

```
bp_systolic2 %>%
  group_by(subject_id) %>%
  summarise(avg_sys = mean(systolic),
    last_measurement = max(time))
```

subject_id <dbl>	avg_sys <dbl>	last_measurement <dbl>
1	119.6667	3
2	128.0000	2
3	141.0000	1

3 rows

tidyr

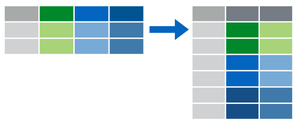


A tidyverse package that reshapes the layout of tabular data.

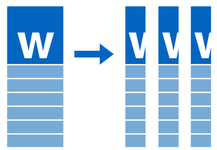
Adapted from 'Master the tidyverse' CC by RStudio

Reshaping verbs in tidyr

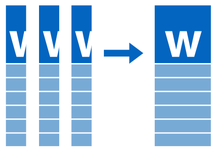
Move values into column names with **spread()**



Move column names into values with **gather()**



Split a column with **separate()** or **separate_rows()**



Unite columns with **unite()**

Adapted from 'Master the tidyverse' CC by RStudio

gather()

Toy data

01-Reshaping-Data.Rmd

```r
---
title: "Reshaping Data"
output: html_notebook
---

```{r setup}
library(tidyverse)

# Toy data
cases <- tribble(
  ~Country, ~"2011", ~"2012",
      "FR",    7000,    6900,
      "DE",    5800,    6000,
      "US",   15000,   14000,
)

pollution <- tribble(
       ~city,    ~size, ~amour
  "New York", "large",      2
  "New York", "small",      1,
    "London", "large",     22,
    "London", "small",     16,
   "Beijing", "large",    121,
   "Beijing", "small",    121
)


bp_systolic <- tribble(
  ~ subject_id,  ~ time_1, ~ time_2, ~ time_3,
             1,       120,      118,      121,
```

```r
cases <- tribble(
  ~Country, ~"2011", ~"2012", ~"2013",
      "FR",    7000,    6900,    7000,
      "DE",    5800,    6000,    6200,
      "US",   15000,   14000,   13000
)
```

Adapted from 'Master the tidyverse' CC by RStudio
```

Quiz

What are the variables in `cases`?

Country	2011	2012	2013
FR	7000	6900	7000
DE	5800	6000	6200
US	15000	14000	13000

Adapted from 'Master the tidyverse' CC by RStudio

Quiz

What are the variables in cases?

Country	[illegible]	[illegible]	[illegible]
FR	7000	[illegible]	7000
DE	[illegible]800	6000	620[illegible]
US	15000	[illegible]	13000

- Country
- Year
- Count

Adapted from 'Master the tidyverse' CC by RStudio

Your Turn 3

On a sheet of paper, draw how the cases data set would look if it had the same values grouped into three columns: *country*, *year*, *n*

Country	2011	2012	2013
FR	7000	6900	7000
DE	5800	6000	6200
US	15000	14000	13000

Adapted from 'Master the tidyverse' CC by RStudio

Country	2011	2012	2013
FR	7000	6900	7000
DE	5800	6000	6200
US	15000	14000	13000

Adapted from 'Master the tidyverse' CC by RStudio

Country	2011	2012	2013
FR	7000	6900	7000
DE	5800	6000	6200
US	15000	14000	13000

Country	Year	n

Adapted from 'Master the tidyverse' CC by RStudio

Country	2011	2012	2013
FR	7000	6900	7000
DE	5800	6000	6200
US	15000	14000	13000

Country	Year	n
FR	2011	7000

Adapted from 'Master the tidyverse' CC by RStudio

Country	2011	2012	2013
FR	7000	6900	7000
DE	5800	6000	6200
US	15000	14000	13000

Country	Year	n
FR	2011	7000
DE	2011	5800

Adapted from 'Master the tidyverse' CC by RStudio

Country	2011	2012	2013
FR	7000	6900	7000
DE	5800	6000	6200
US	15000	14000	13000

Country	Year	n
FR	2011	7000
DE	2011	5800
US	2011	15000

Adapted from 'Master the tidyverse' CC by RStudio

Country	2011	2012	2013
FR	7000	6900	7000
DE	5800	6000	6200
US	15000	14000	13000

Country	Year	n
FR	2011	7000
DE	2011	5800
US	2011	15000
FR	2012	6900

Adapted from 'Master the tidyverse' CC by RStudio

Country	2011	2012	2013
FR	7000	6900	7000
DE	5800	6000	6200
US	15000	14000	13000

Country	Year	n
FR	2011	7000
DE	2011	5800
US	2011	15000
FR	2012	6900
DE	2012	6000

Adapted from 'Master the tidyverse' CC by RStudio

Country	2011	2012	2013
FR	7000	6900	7000
DE	5800	6000	6200
US	15000	14000	13000

Country	Year	n
FR	2011	7000
DE	2011	5800
US	2011	15000
FR	2012	6900
DE	2012	6000
US	2012	14000

Adapted from 'Master the tidyverse' CC by RStudio

Country	2011	2012	2013
FR	7000	6900	7000
DE	5800	6000	6200
US	15000	14000	13000

Country	Year	n
FR	2011	7000
DE	2011	5800
US	2011	15000
FR	2012	6900
DE	2012	6000
US	2012	14000
FR	2013	7000

Adapted from 'Master the tidyverse' CC by RStudio

Country	2011	2012	2013
FR	7000	6900	7000
DE	5800	6000	6200
US	15000	14000	13000

Country	Year	n
FR	2011	7000
DE	2011	5800
US	2011	15000
FR	2012	6900
DE	2012	6000
US	2012	14000
FR	2013	7000
DE	2013	6200

Adapted from 'Master the tidyverse' CC by RStudio

Country	2011	2012	2013
FR	7000	6900	7000
DE	5800	6000	6200
US	15000	14000	13000

Country	Year	n
FR	2011	7000
DE	2011	5800
US	2011	15000
FR	2012	6900
DE	2012	6000
US	2012	14000
FR	2013	7000
DE	2013	6200
US	2013	13000

Adapted from 'Master the tidyverse' CC by RStudio

Country	2011	2012	2013
FR	7000	6900	7000
DE	5800	6000	6200
US	15000	14000	13000

Country	Year	
FR	2011	7000
DE	2011	5800
US	2011	15000
FR	2012	6900
DE	2012	6000
US	2012	14000
FR	2013	7000
DE	2013	6200
US	2013	13000

Adapted from 'Master the tidyverse' CC by RStudio

Country	2011	2012	2013
FR	7000	6900	7000
DE	5800	6000	6200
US	15000	14000	13000

gather()

Country	Year	n
FR	2011	7000
DE	2011	5800
US	2011	15000
FR	2012	6900
DE	2012	6000
US	2012	14000
FR	2013	7000
DE	2013	6200
US	2013	13000

Adapted from 'Master the tidyverse' CC by RStudio

Country	2011	2012	2013
FR	7000	6900	7000
DE	5800	6000	6200
US	15000	14000	13000

	1	2
Country	Year	n
FR	2011	7000
DE	2011	5800
US	2011	15000
FR	2012	6900
DE	2012	6000
US	2012	14000
FR	2013	7000
DE	2013	6200
US	2013	13000

Adapted from 'Master the tidyverse' CC by RStudio

Country	2011	2012	2013
FR	7000	6900	7000
DE	5800	6000	6200
US	15000	14000	13000

key (former column names)

Country	Year	n
FR	2011	7000
DE	2011	5800
US	2011	15000
FR	2012	6900
DE	2012	6000
US	2012	14000
FR	2013	7000
DE	2013	6200
US	2013	13000

Adapted from 'Master the tidyverse' CC by RStudio

key **value** (former cells)

Country	2011	2012	2013
FR	7000	6900	7000
DE	5800	6000	6200
US	15000	14000	13000

Country	Year	n
FR	2011	7000
DE	2011	5800
US	2011	15000
FR	2012	6900
DE	2012	6000
US	2012	14000
FR	2013	7000
DE	2013	6200
US	2013	13000

Adapted from 'Master the tidyverse' CC by RStudio

gather()

```
cases %>% gather(key = "year", value = "n", 2:4)
```

data frame to reshape

name of the new key column (a character string)

name of the new value column (a character string)

numeric indexes of columns to collapse (or names)

Adapted from 'Master the tidyverse' CC by RStudio

gather()

```
cases %>% gather("year","n", 2:4)
```

numeric indexes

Country	2 2011	3 2012	4 2013
<chr>	<dbl>	<dbl>	<dbl>
FR	7000	6900	7000
DE	5800	6000	6200
US	15000	14000	13000

Adapted from 'Master the tidyverse' CC by RStudio

gather()

```
cases %>% gather("year","n", "2011", "2012", "2013")
```

names

Country <chr>	2011 2011 <dbl>	2012 2012 <dbl>	2013 2013 <dbl>
FR	7000	6900	7000
DE	5800	6000	6200
US	15000	14000	13000

Adapted from 'Master the tidyverse' CC by RStudio

gather()

```
cases %>% gather("year","n", -Country)
```

Everything except...

Country <chr>	Not Country 2011 <dbl>	Not Country 2012 <dbl>	Not Country 2013 <dbl>
FR	7000	6900	7000
DE	5800	6000	6200
US	15000	14000	13000

Adapted from 'Master the tidyverse' CC by RStudio

Your Turn 4

Use **gather()** to reorganize **table4a** into three columns: *country*, *year*, and *cases*.

	country <chr>	1999 <int>	2000 <int>
1	Afghanistan	745	2666
2	Brazil	37737	80488
3	China	212258	213766

3 rows

Adapted from 'Master the tidyverse' CC by RStudio

```
table4a %>%
  gather(key = "year", value = "n", 2:3)
```

country <chr>	year <chr>	n <int>
Afghanistan	1999	745
Brazil	1999	37737
China	1999	212258
Afghanistan	2000	2666
Brazil	2000	80488
China	2000	213766

6 rows

Adapted from 'Master the tidyverse' CC by RStudio

```
table4a %>%
  gather(key = "year", value = "n", 2:3, convert = TRUE)
```

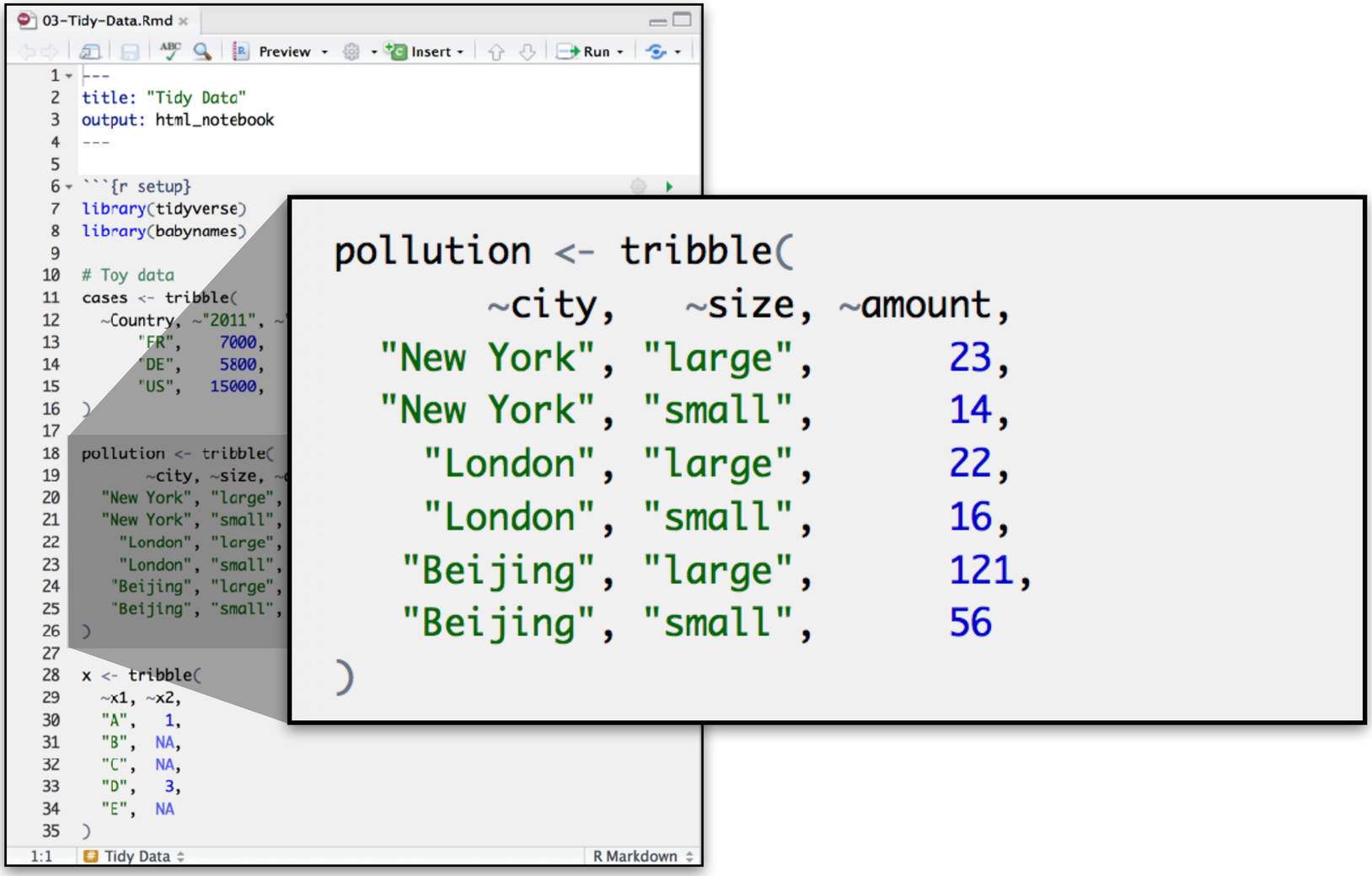
country <chr>	year <int>	n <int>
Afghanistan	1999	745
Brazil	1999	37737
China	1999	212258
Afghanistan	2000	2666
Brazil	2000	80488
China	2000	213766

6 rows

Adapted from 'Master the tidyverse' CC by RStudio

spread()

Toy data



Adapted from 'Master the tidyverse' CC by RStudio

Quiz

What are the variables in `pollution`?

city	particle size	amount ($\mu g/m^3$)
New York	large	23
New York	small	14
London	large	22
London	small	16
Beijing	large	121
Beijing	small	56

Adapted from 'Master the tidyverse' CC by RStudio

Quiz

What are the variables in pollution?

city	particle size	amount ($\mu g/m^3$)
New York	large	23
New York	small	14
London	large	22
London	small	16
Beijing	large	121
Beijing	small	56

- City
- Amount of large particulate
- Amount of small particulate

Adapted from 'Master the tidyverse' CC by RStudio

Your Turn 5

On a sheet of paper, draw how this data set would look if it had the same values grouped into three columns: *city*, *large*, *small*

city	size	amount
New York	large	23
New York	small	14
London	large	22
London	small	16
Beijing	large	121
Beijing	small	56

Adapted from 'Master the tidyverse' CC by RStudio

city	size	amount
New York	large	23
New York	small	14
London	large	22
London	small	16
Beijing	large	121
Beijing	small	56

Adapted from 'Master the tidyverse' CC by RStudio

city	size	amount
New York	large	23
New York	small	14
London	large	22
London	small	16
Beijing	large	121
Beijing	small	56

city	large	small

Adapted from 'Master the tidyverse' CC by RStudio

city	size	amount
New York	large	23
New York	small	14
London	large	22
London	small	16
Beijing	large	121
Beijing	small	56

city	large	small
New York	23	

Adapted from 'Master the tidyverse' CC by RStudio

city	size	amount
New York	large	23
New York	small	14
London	large	22
London	small	16
Beijing	large	121
Beijing	small	56

city	large	small
New York	23	14

Adapted from 'Master the tidyverse' CC by RStudio

city	size	amount
New York	large	23
New York	small	14
London	large	22
London	small	16
Beijing	large	121
Beijing	small	56

city	large	small
New York	23	14
London	22	

Adapted from 'Master the tidyverse' CC by RStudio

city	size	amount
New York	large	23
New York	small	14
London	large	22
London	small	16
Beijing	large	121
Beijing	small	56

city	large	small
New York	23	14
London	22	16

Adapted from 'Master the tidyverse' CC by RStudio

city	size	amount
New York	large	23
New York	small	14
London	large	22
London	small	16
Beijing	large	121
Beijing	small	56

city	large	small
New York	23	14
London	22	16
Beijing	121	

Adapted from 'Master the tidyverse' CC by RStudio

city	size	amount
New York	large	23
New York	small	14
London	large	22
London	small	16
Beijing	large	121
Beijing	small	56

city	large	small
New York	23	14
London	22	16
Beijing	121	56

Adapted from 'Master the tidyverse' CC by RStudio

city	size	amount
New York	large	23
New York	small	14
London	large	22
London	small	16
Beijing	large	121
Beijing	small	56

city	large	small
New York	23	14
London	22	16
Beijing	121	56

Adapted from 'Master the tidyverse' CC by RStudio

city	size	amount
New York	large	23
New York	small	14
London	large	22
London	small	16
Beijing	large	121
Beijing	small	56

spread()

city	large	small
New York	23	14
London	22	16
Beijing	121	56

Adapted from 'Master the tidyverse' CC by RStudio

	1	2
city	size	amount
New York	large	23
New York	small	14
London	large	22
London	small	16
Beijing	large	121
Beijing	small	56

city	large	small
New York	23	14
London	22	16
Beijing	121	56

Adapted from 'Master the tidyverse' CC by RStudio

key (new column names)

city	size	amount
New York	large	23
New York	small	14
London	large	22
London	small	16
Beijing	large	121
Beijing	small	56

city	large	small
New York	23	14
London	22	16
Beijing	121	56

Adapted from 'Master the tidyverse' CC by RStudio

key **value** (new cells)

city	size	amount
New York	large	23
New York	small	14
London	large	22
London	small	16
Beijing	large	121
Beijing	small	56

city	large	small
New York	23	14
London	22	16
Beijing	121	56

Adapted from 'Master the tidyverse' CC by RStudio

spread()

```
pollution %>% spread(key = size, value = amount)
```

data frame to reshape

column to use for keys (becomes new column names)

column to use for values (becomes new column cells)

Adapted from 'Master the tidyverse' CC by RStudio

```
pollution %>% spread(size, amount)
```

```
      city  size amount
1 New York large     23
2 New York small     14
3   London large     22
4   London small     16
5  Beijing large    121
6  Beijing small     56
```

```
      city large small
1  Beijing   121    56
2   London    22    16
3 New York    23    14
```

CC by RStudio

Your Turn 6

Use **spread()** to reorganize **table2** into four columns: *country*, *year*, *cases*, and *population*.

country <chr>	year <int>	type <chr>	count <int>
Afghanistan	1999	cases	745
Afghanistan	1999	population	19987071
Afghanistan	2000	cases	2666
Afghanistan	2000	population	20595360
Brazil	1999	cases	37737
Brazil	1999	population	172006362

Adapted from 'Master the tidyverse' CC by RStudio

```
table2 %>%
  spread(key = type, value = count)
```

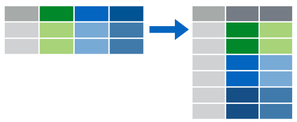
	country <chr>	year <int>	cases <int>	population <int>
1	Afghanistan	1999	745	19987071
2	Afghanistan	2000	2666	20595360
3	Brazil	1999	37737	172006362
4	Brazil	2000	80488	174504898
5	China	1999	212258	1272915272
6	China	2000	213766	1280428583

6 rows

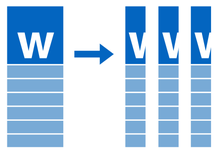
Adapted from 'Master the tidyverse' CC by RStudio

Reshaping verbs in tidyr

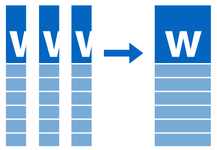
Move values into column names with **spread()**



Move column names into values with **gather()**



Split a column with **separate()** or **separate_rows()**



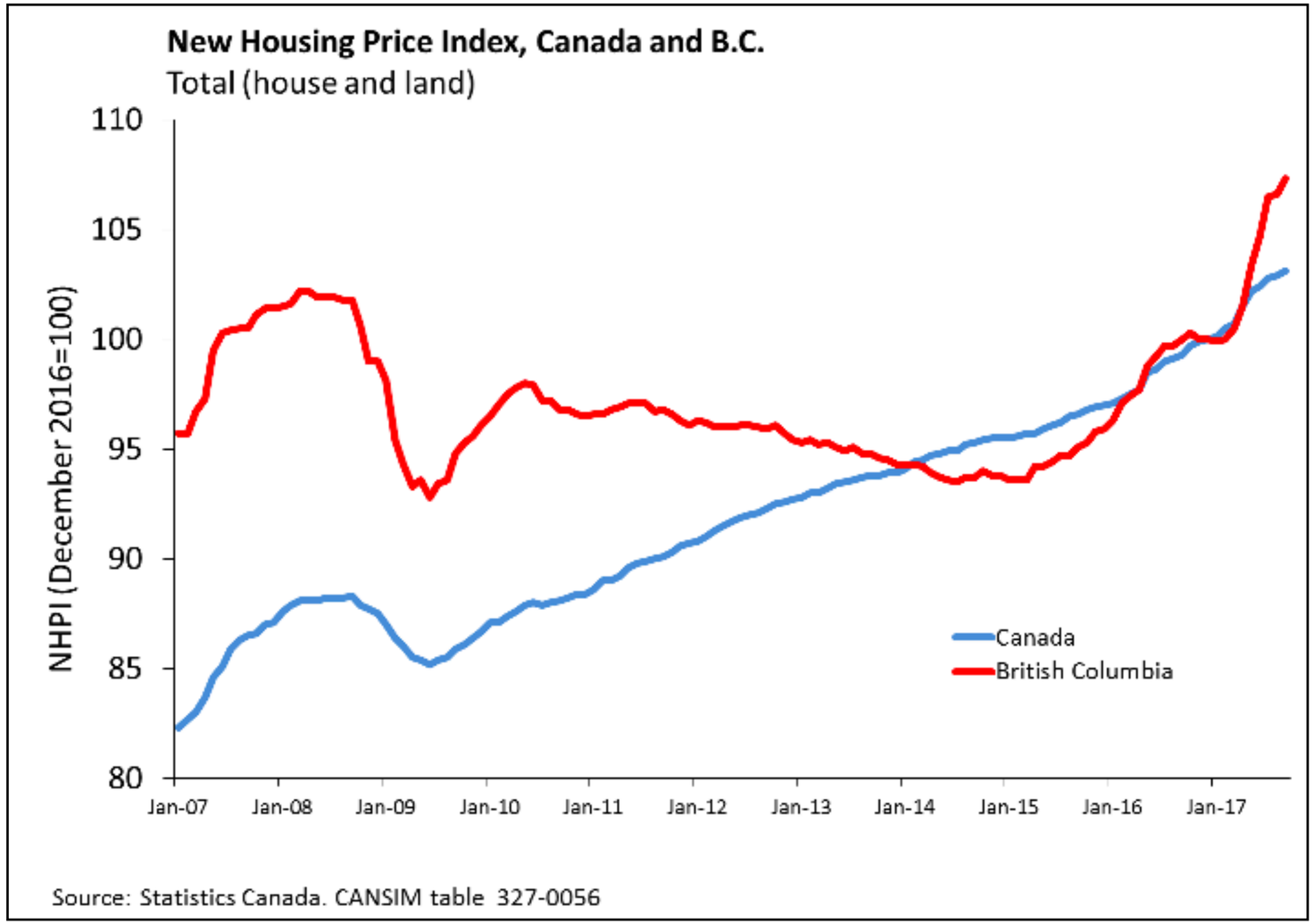
Unite columns with **unite()**

Adapted from 'Master the tidyverse' CC by RStudio

Project

New Housing Price Index

https://www2.gov.bc.ca/gov/content/data/statistics/infoline/infoline-2017/17-146-price-new-housing



CC BY Charlotte Wickham

Your Turn

Switch back to the project-housing.Rproj project.

1. Open 02-tidy.Rmd
2. Take a look at `housing_raw`
3. **Brainstorm:** What needs to happen to get it in a form to plot?

CC BY Charlotte Wickham

lubridate::parse_date_time()

Easy way to convert strings into dates

```
lubridate::parse_date_time(c("Jan 2016"), order = "my")
# [1] "2016-01-01 UTC"
```

CC BY Charlotte Wickham

stringr::str_detect()

Easy way to look for matches to a pattern

```
stringr::str_detect(c("apple", "pear", "pineapple"),
  pattern = "apple")
# [1]  TRUE FALSE  TRUE
```

CC BY Charlotte Wickham

CC BY Charlotte Wickham