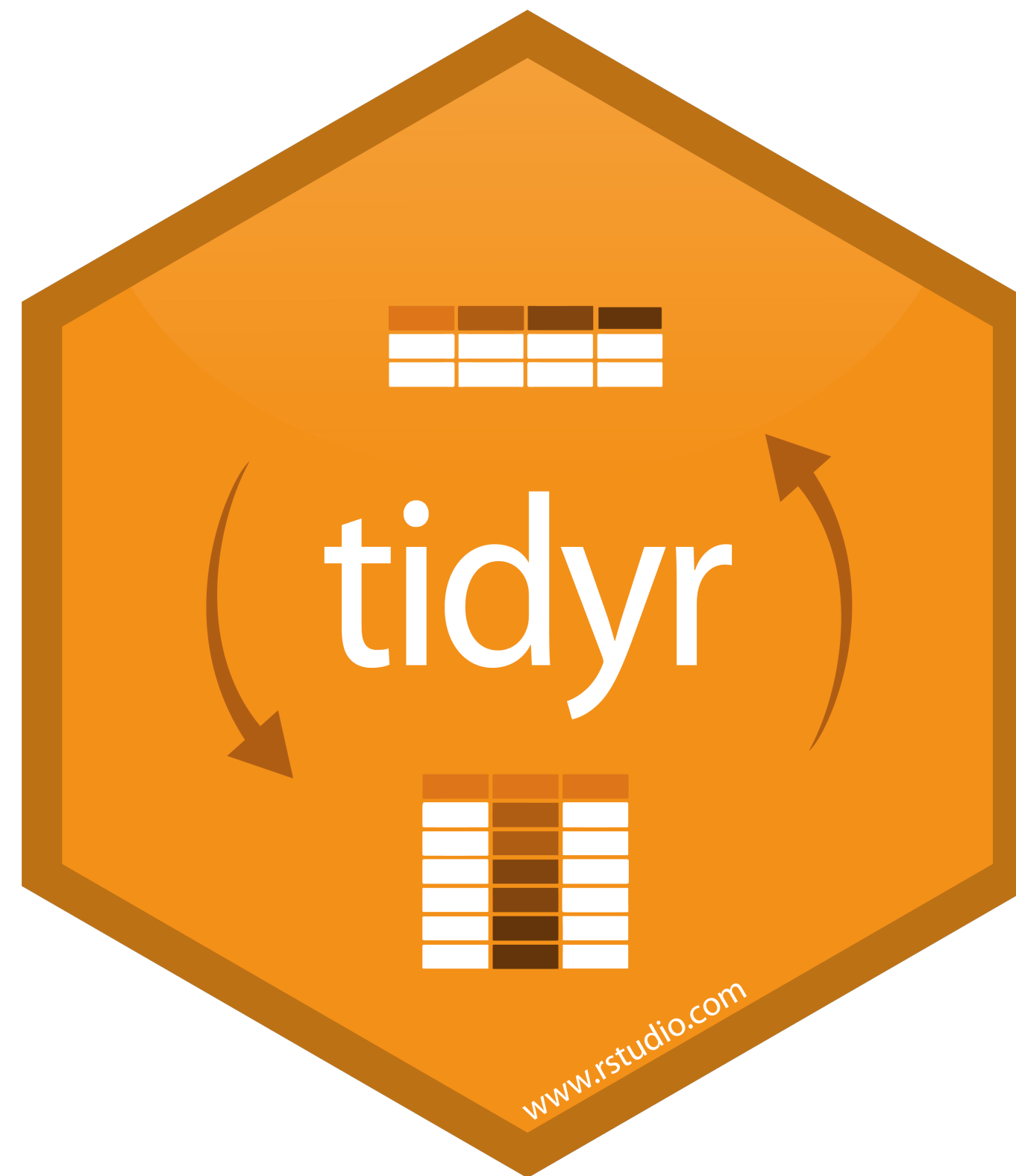


# Tidy Data with



Navigate up to the **03-Tidy** folder.

Open on **03-Tidy-Exercises**.

"Data are not just numbers,  
they are numbers with a context."

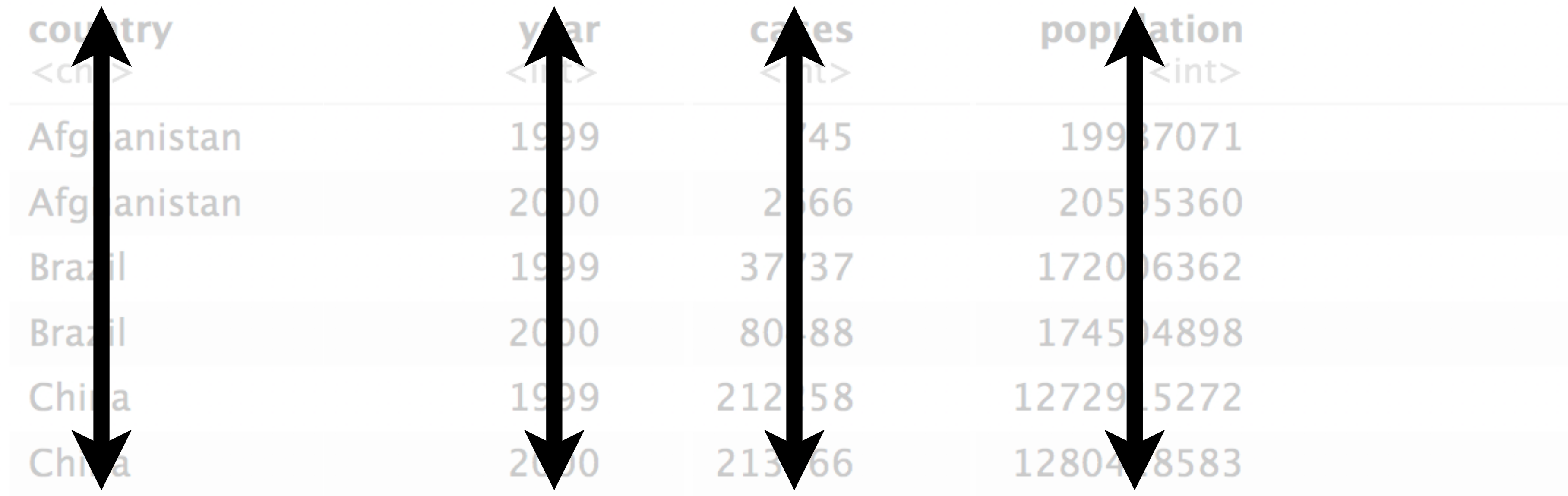
- George Cobb and David Moore (1997)



# Quiz

What are the variables in this data set?

table1



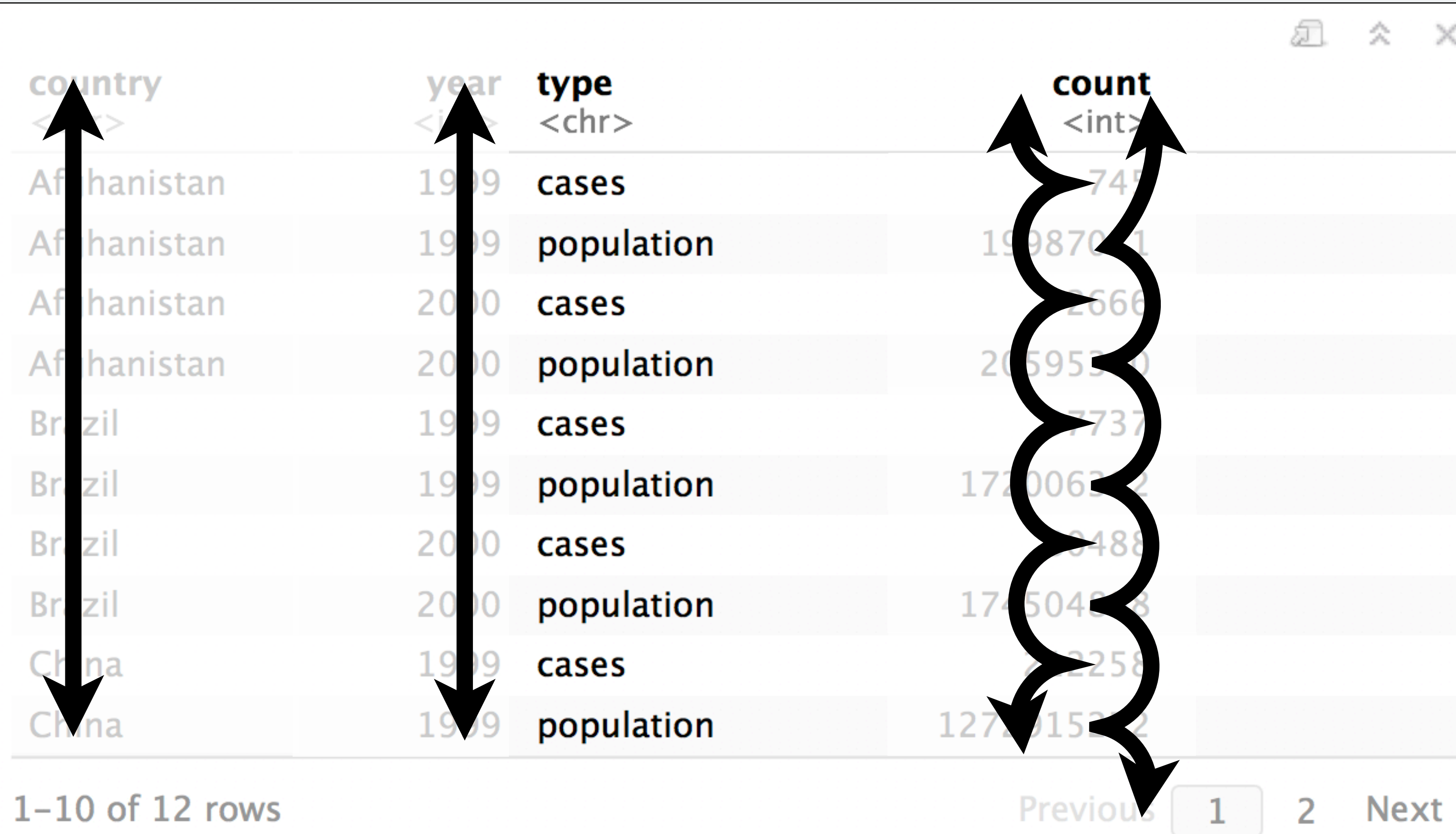
country <chr>	year <int>	cases <int>	population <int>
Afghanistan	1999	745	19987071
Afghanistan	2000	266	20595360
Brazil	1999	37737	172006362
Brazil	2000	80488	174504898
China	1999	212158	1272915272
China	2000	213766	128048583

6 rows

# Quiz

What are the variables in this data set?

table2



The image shows a screenshot of a data table with four columns: country, year, type, and count. The table contains 10 rows of data. Hand-drawn arrows indicate the variables in the data set: a straight arrow for 'country', a straight arrow for 'year', and a wavy arrow for 'count'.


country <chr>	year <int>	type <chr>	count <int>
Afghanistan	1999	cases	745
Afghanistan	1999	population	1998701
Afghanistan	2000	cases	2666
Afghanistan	2000	population	2059530
Brazil	1999	cases	7737
Brazil	1999	population	17200632
Brazil	2000	cases	9488
Brazil	2000	population	17450488
China	1999	cases	2258
China	1999	population	12720152

1-10 of 12 rows

Previous 1 2 Next



table3



	<b>country</b> <chr>	<b>year</b> <int>	<b>rate</b> <chr>
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

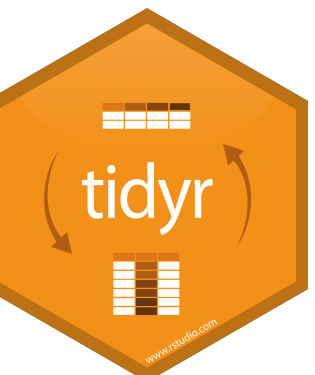


table4a

table4b

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

3 rows

	<b>country</b> <chr>	<b>1999</b> <int>	<b>2000</b> <int>
1	Afghanistan	19987071	20595360
2	Brazil	172006362	174504898
3	China	1272915272	1280428583

3 rows

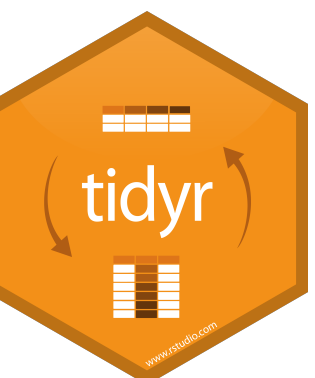
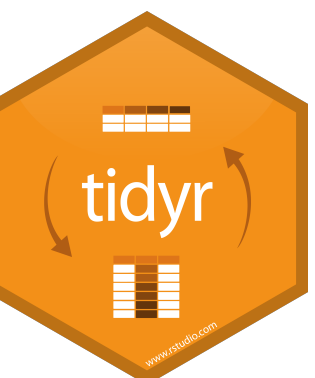


table5



	<b>country</b> <chr>	<b>century</b> <chr>	<b>year</b> <chr>	<b>rate</b> <chr>
1	Afghanistan	19	99	745/19987071
2	Afghanistan	20	00	2666/20595360
3	Brazil	19	99	37737/172006362
4	Brazil	20	00	80488/174504898
5	China	19	99	212258/1272915272
6	China	20	00	213766/1280428583

6 rows





"Data comes in many formats, but R prefers just one: tidy data."

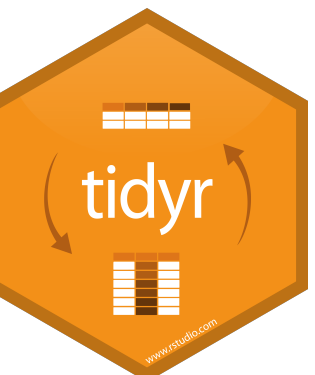
- Garrett Grolemund


# Tidy data

country	year	cases	pop
Afghanistan	1999	745	10137321
Afghanistan	2000	666	20125120
Afghanistan	2001	787	22795522
Afghanistan	2002	1153	24795522
Afghanistan	2003	2223	27137321
Afghanistan	2004	3760	2842363

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**

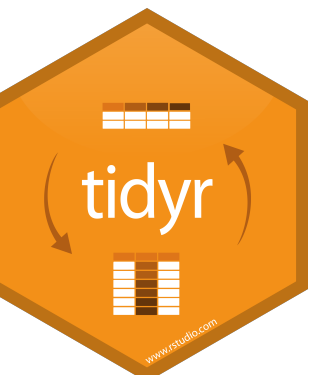




<b>country</b> <chr>	<b>year</b> <int>	<b>cases</b> <int>	<b>population</b> <int>
Afghanistan	1999	745	19987071
Afghanistan	2000	2666	20595360
Brazil	1999	37737	172006362
Brazil	2000	80488	174504898
China	1999	212258	1272915272
China	2000	213766	1280428583

6 rows

```
table1$country
table1$year
table1$cases
table1$population
```





📄 ⬆ ✕

country <chr>	year <int>	type <chr>	count <int>
Afghanistan	1999	cases	745
Afghanistan	1999	pop	19987071
Afghanistan	2000	cases	2666
Afghanistan	2000	pop	195360
Brazil	1999	cases	737
Brazil	1999	pop	162
Brazil	2000	cases	88
Brazil	2000	pop	108
China	1999	cases	58
China	1999	pop	172

1–10 of 12 rows

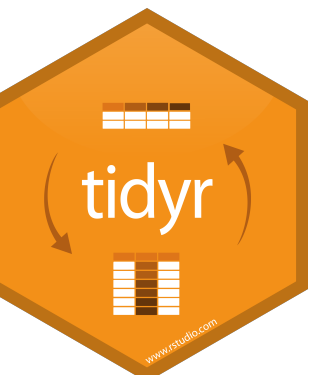
Previous 1 2 Next

```
table2$country
table2$year
table2$count[c(1,3,5,7,9,11)]
table2$count[c(2,4,6,8,10,12)]
```

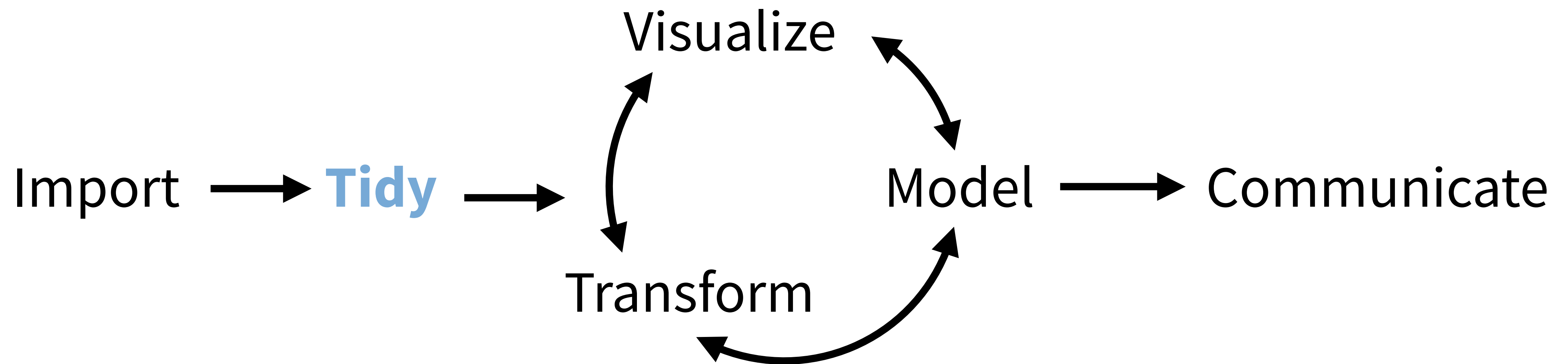
<b>country</b> <chr>	<b>year</b> <int>	<b>cases</b> <int>	<b>population</b> <int>	<b>rate</b> <dbl>
Afghanistan	1999	745	19987071	0.0000372741
Afghanistan	2000	2666	20595360	0.0001294466
Brazil	1999	37737	172006362	0.0002193930
Brazil	2000	80488	174504898	0.0004612363
China	1999	212258	1272915272	0.0001667495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

6 rows

```
table1$cases / table1$population -> table1$rate
```



# (Applied) Data Science



Program





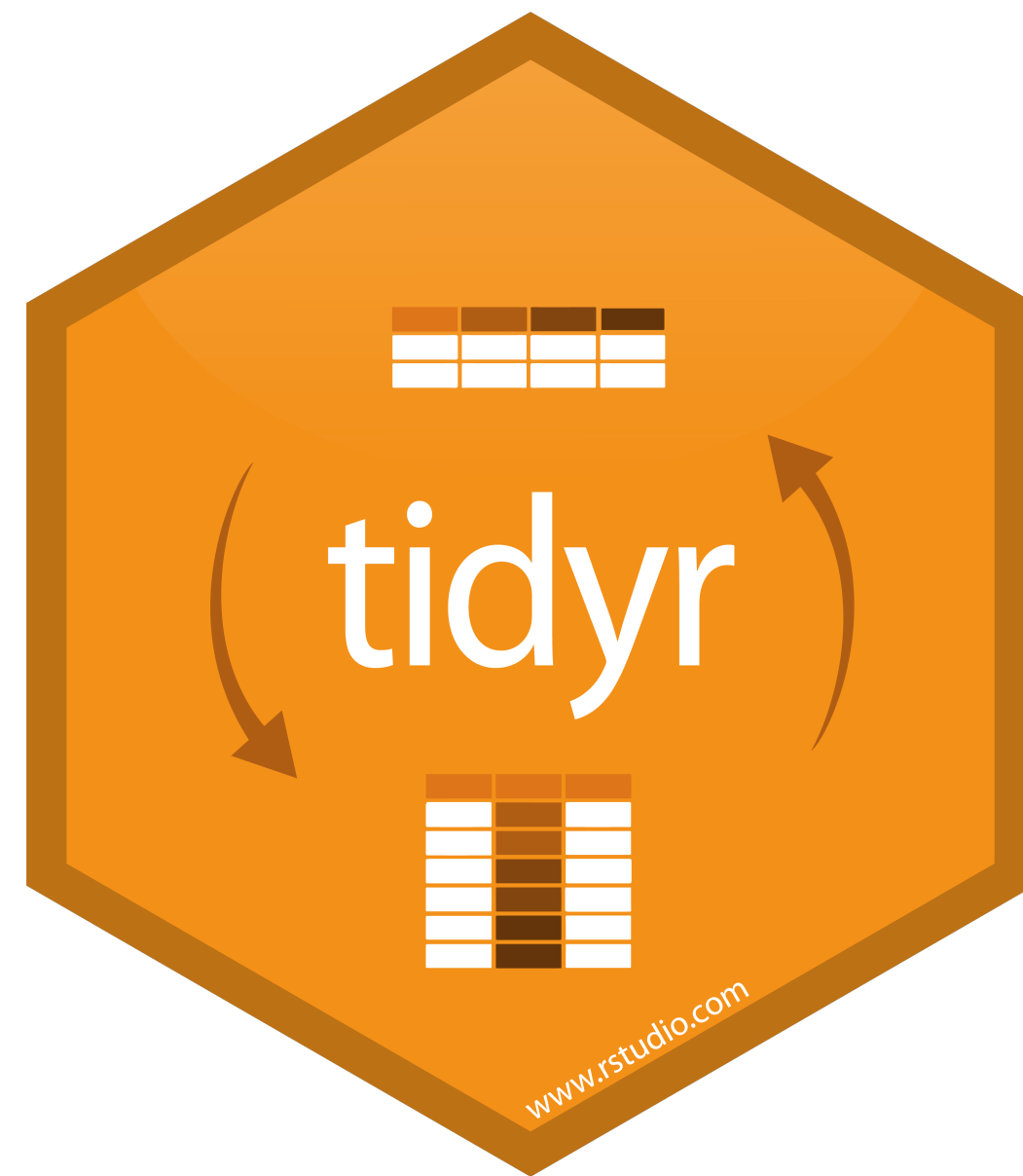
"Tidy data sets are all alike; but every messy data set is messy in its own way."

- Hadley Wickham

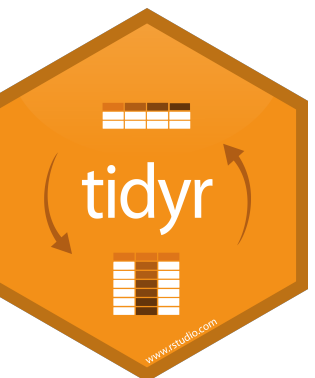
# tidyr



# tidyr



A package that reshapes the layout of tabular data.





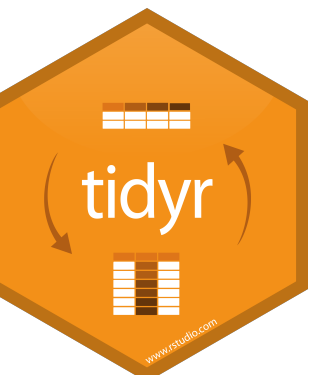
# gather()



# Toy data

```
03-Tidy-Data.Rmd x
1 ---
2 title: "Tidy Data"
3 output: html_notebook
4 ---
5
6 ```{r setup}
7 library(tidyverse)
8 library(babynames)
9
10 # Toy data
11 cases <- tribble(
12   ~Country, ~"2011", ~"2012", ~"2013",
13   "FR",      7000,    6900,    7000,
14   "DE",      5800,    6000,    6200,
15   "US",     15000,   14000,    13000,
16 )
17
18 pollution <- tribble(
19   ~city, ~size, ~amount,
20   "New York", "large", 23,
21   "New York", "small", 14,
22   "London", "large", 22,
23   "London", "small", 16,
24   "Beijing", "large", 121,
25   "Beijing", "small", 121,
26 )
27
28 x <- tribble(
29   ~x1, ~x2,
30   "A", 1,
31   "B", NA,
32   "C", NA,
33   "D", 3,
34   "E", NA,
35 )
```

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



# Quiz

What are the variables in cases?

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



# Quiz

What are the variables in cases?

Country	2011	2012	2013
FR	7000	6900	7000
DE	6800	6000	6200
US	15000	14000	13000

- Country
- Year
- Count

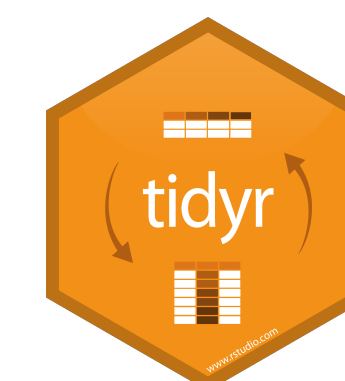
# Your Turn 1

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

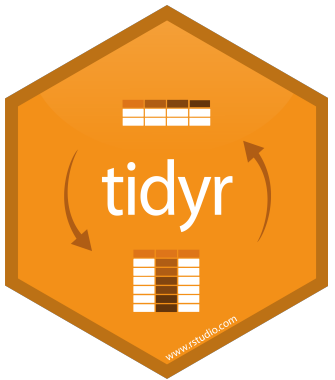
04:00

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



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

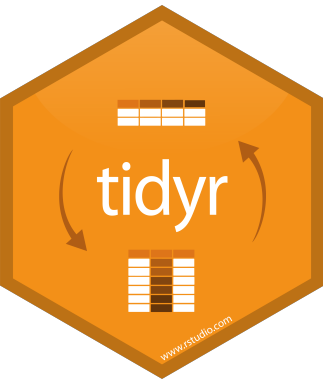
Country	Year	n
---------	------	---





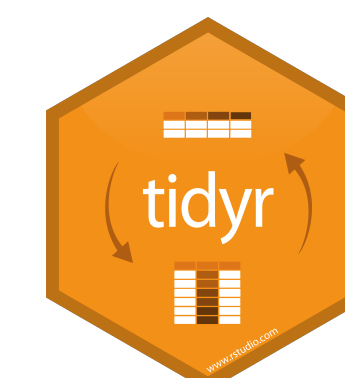
Country	2011	2012	2013
FR	7000	6900	7000
DE	5800	6000	6200
US	15000	14000	13000

Country	Year	n
FR	2011	7000



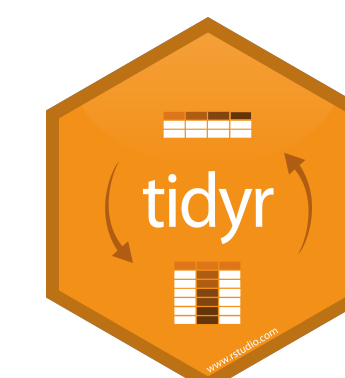
Country	2011	2012	2013
FR	7000	6900	7000
DE	5800	6000	6200
US	15000	14000	13000

Country	Year	n
FR	2011	7000
DE	2011	5800



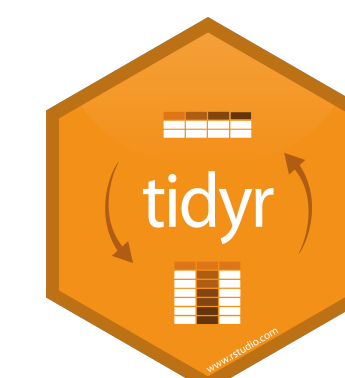
Country	2011	2012	2013
FR	7000	6900	7000
DE	5800	6000	6200
US	15000	14000	13000

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



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

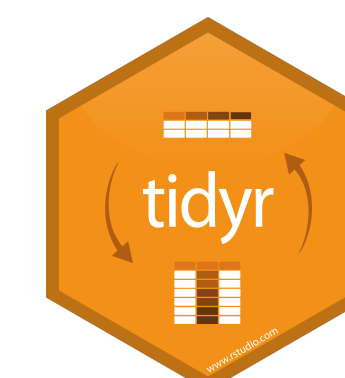
Country	Year	n
FR	2011	7000
DE	2011	5800
US	2011	15000
FR	2012	6900





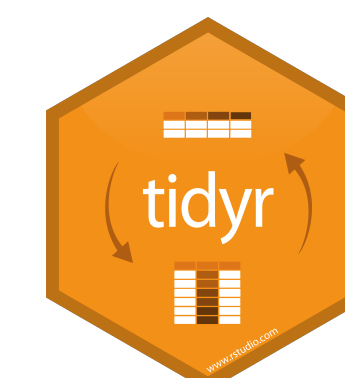
Country	2011	2012	2013
FR	7000	6900	7000
DE	5800	6000	6200
US	15000	14000	13000

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



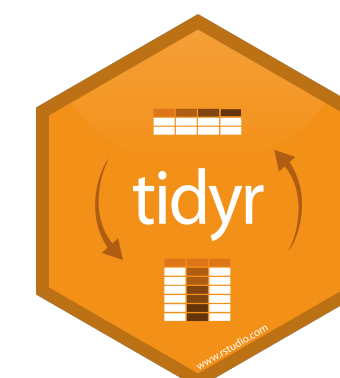
Country	2011	2012	2013
FR	7000	6900	7000
DE	5800	6000	6200
US	15000	14000	13000

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



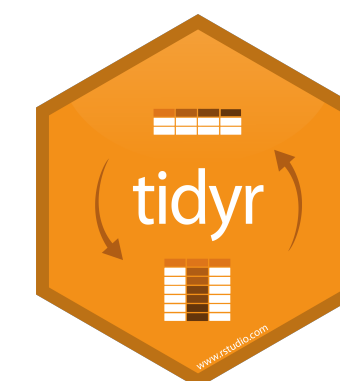
Country	2011	2012	2013
FR	7000	6900	7000
DE	5800	6000	6200
US	15000	14000	13000

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



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

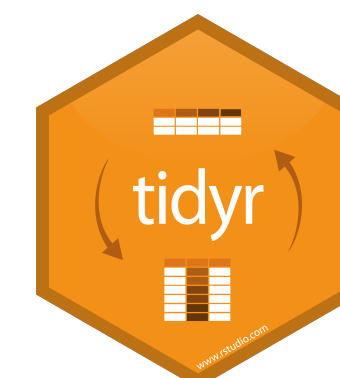
Country	Year	n
FR	2011	7000
DE	2011	5800
US	2011	15000
FR	2012	6900
DE	2012	6000
US	2012	14000
FR	2013	7000
DE	2013	6200





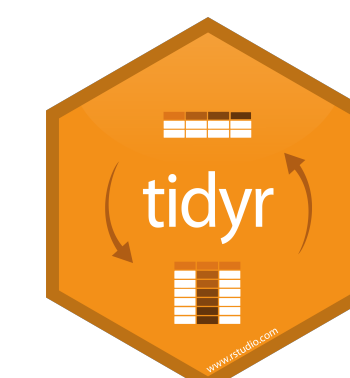
Country	2011	2012	2013
FR	7000	6900	7000
DE	5800	6000	6200
US	15000	14000	13000

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



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

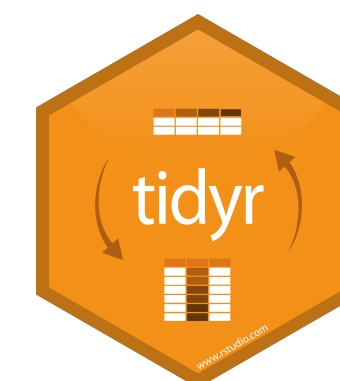
Country	Year	Revenue
FR	2011	7000
DE	2011	5800
US	2011	15000
FR	2012	6900
DE	2012	6000
US	2012	14000
FR	2013	7000
DE	2013	6200
US	2013	13000



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

gather()

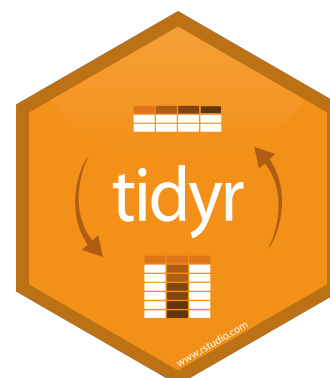
Country	Year	n
FR	2011	7000
DE	2011	5800
US	2011	15000
FR	2012	6900
DE	2012	6000
US	2012	14000
FR	2013	7000
DE	2013	6200
US	2013	13000



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

12

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

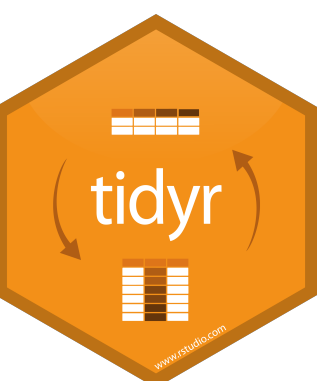




**key** (former column names)

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

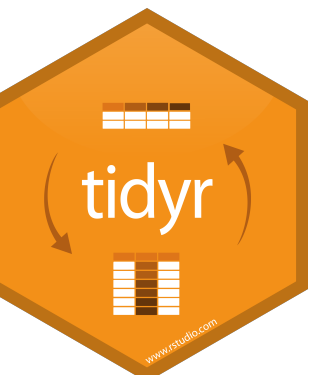
Country	Year	n
FR	2011	7000
DE	2011	5800
US	2011	15000
FR	2012	6900
DE	2012	6000
US	2012	14000
FR	2013	7000
DE	2013	6200
US	2013	13000



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



# 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)

# gather()

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

numeric  
indexes

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



# gather()

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

names

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

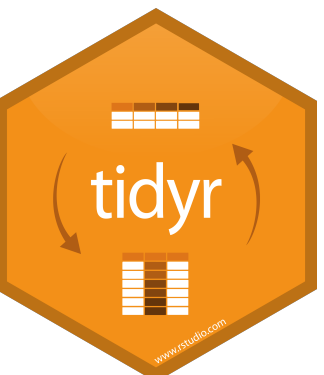
# gather()

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

Everything  
except...

**Not Country Not Country Not Country**

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



# Your Turn 2

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


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

3 rows

03:00

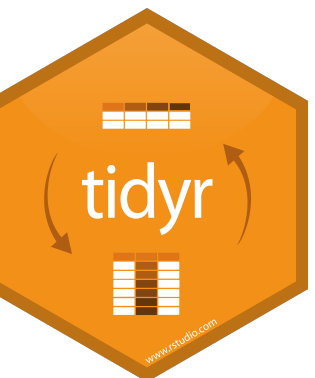


```
table4a %>%  
  gather(key = "year", value = "n", 2:3)
```




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

6 rows

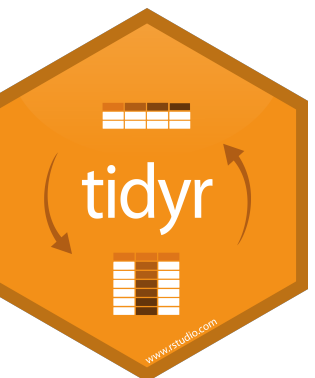


```
table4a %>%  
  gather(key = "year", value = "n", 2:3, convert = TRUE)
```



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

6 rows





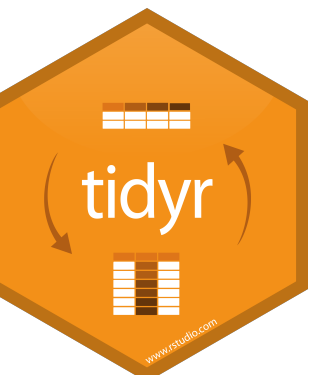
# spread()



# Toy data

```
03-Tidy-Data.Rmd x
1 ---
2 title: "Tidy Data"
3 output: html_notebook
4 ---
5
6 ```{r setup}
7 library(tidyverse)
8 library(babynames)
9
10 # Toy data
11 cases <- tribble(
12   ~Country, ~"2011", ~
13     "FR", 7000,
14     "DE", 5800,
15     "US", 15000,
16 )
17
18 pollution <- tribble(
19   ~city, ~size, ~
20     "New York", "large",
21     "New York", "small",
22     "London", "large",
23     "London", "small",
24     "Beijing", "large",
25     "Beijing", "small",
26 )
27
28 x <- tribble(
29   ~x1, ~x2,
30     "A", 1,
31     "B", NA,
32     "C", NA,
33     "D", 3,
34     "E", NA
35 )
```

```
pollution <- tribble(
  ~city, ~size, ~amount,
  "New York", "large", 23,
  "New York", "small", 14,
  "London", "large", 22,
  "London", "small", 16,
  "Beijing", "large", 121,
  "Beijing", "small", 56
)
```



# Quiz

What are the variables in pollution?

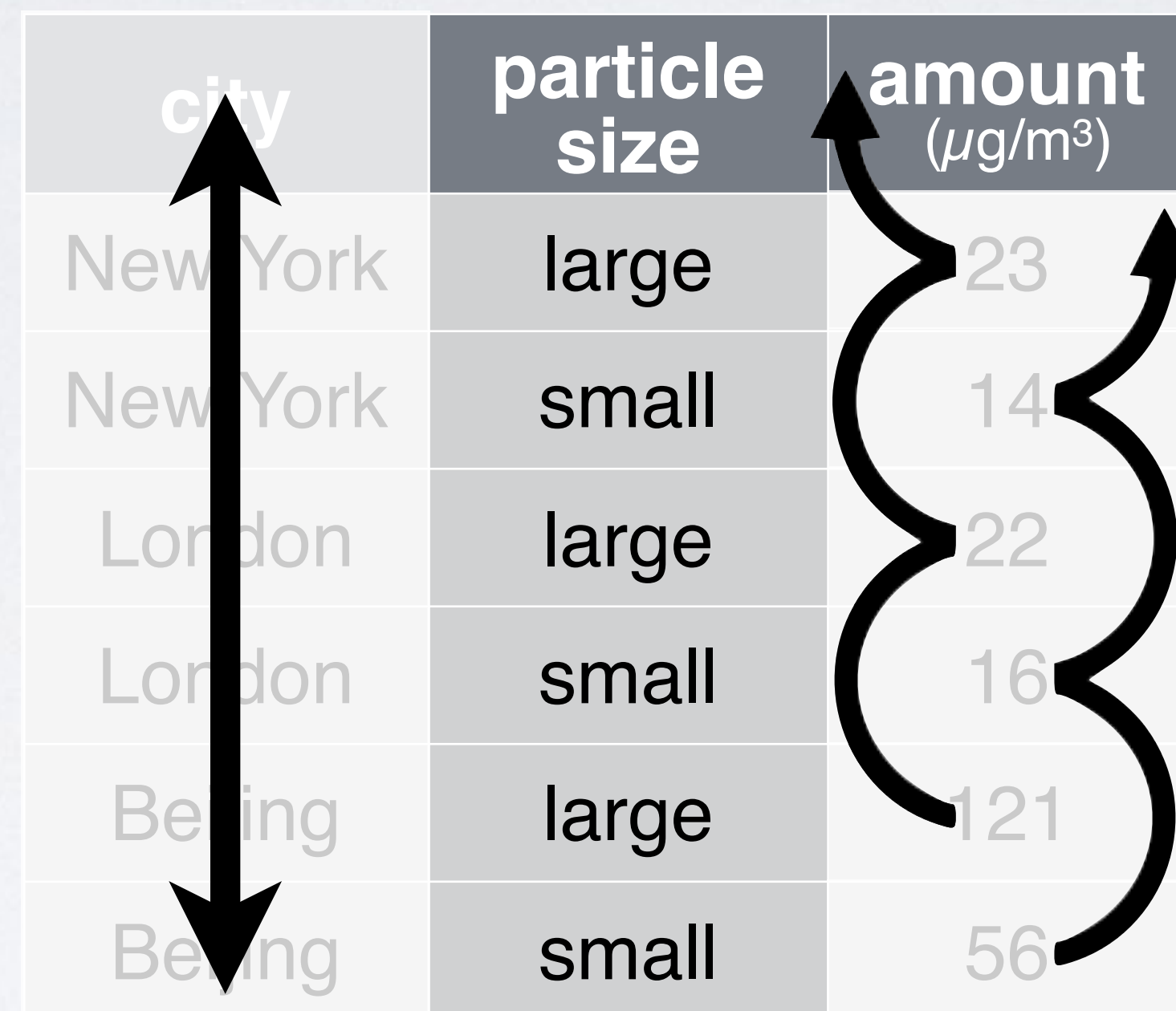
city	particle size	amount ( $\mu\text{g}/\text{m}^3$ )
New York	large	23
New York	small	14
London	large	22
London	small	16
Beijing	large	121
Beijing	small	56



# Quiz

What are the variables in pollution?

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



The diagram illustrates the relationships between the variables in the table. A long vertical double-headed arrow spans the 'city' column, indicating that 'city' is a variable. A curved double-headed arrow connects the 'particle size' and 'amount' columns, indicating that 'particle size' is a variable. Another curved double-headed arrow connects the two rows for each city in the 'amount' column, indicating that 'amount' is a variable.

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

# Your Turn 3

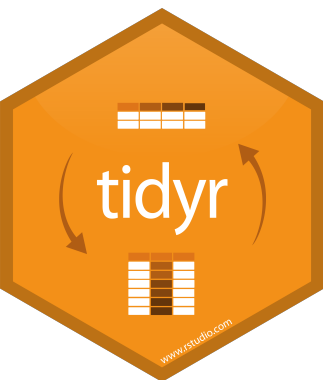
On a sheet of paper, draw how this data set would look if it had the same values grouped into three columns: *city*, *large*, *small*

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

03:00

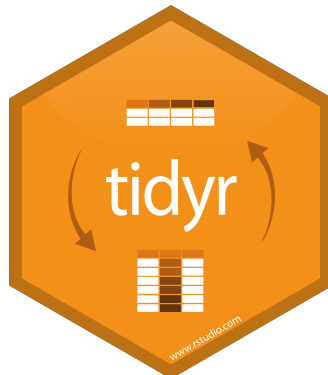


city	size	amount
New York	large	23
New York	small	14
London	large	22
London	small	16
Beijing	large	121
Beijing	small	56



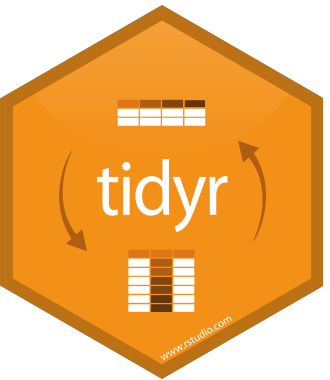
city	size	amount
New York	large	23
New York	small	14
London	large	22
London	small	16
Beijing	large	121
Beijing	small	56

city	large	small
------	-------	-------



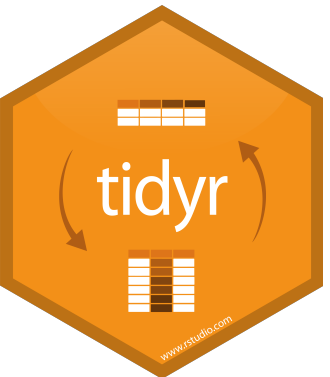
city	size	amount
New York	large	23
New York	small	14
London	large	22
London	small	16
Beijing	large	121
Beijing	small	56

city	large	small
New York	23	



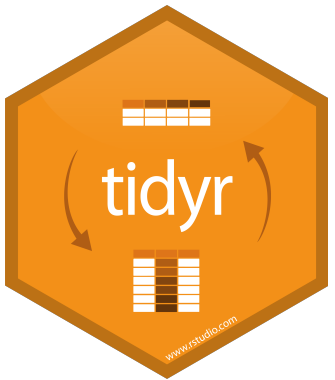
city	size	amount
New York	large	23
New York	small	14
London	large	22
London	small	16
Beijing	large	121
Beijing	small	56

city	large	small
New York	23	14



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

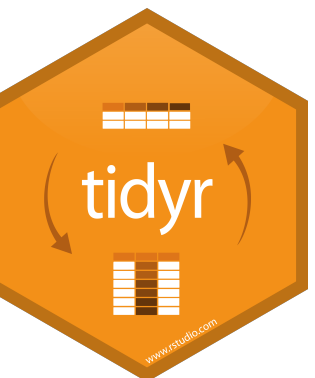
city	large	small
New York	23	14
London	22	





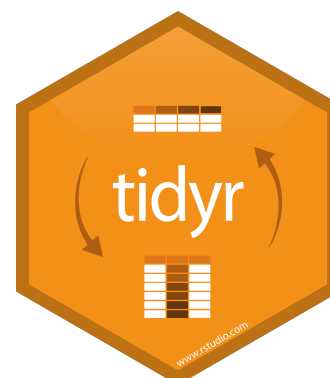
city	size	amount
New York	large	23
New York	small	14
London	large	22
London	small	16
Beijing	large	121
Beijing	small	56

city	large	small
New York	23	14
London	22	16



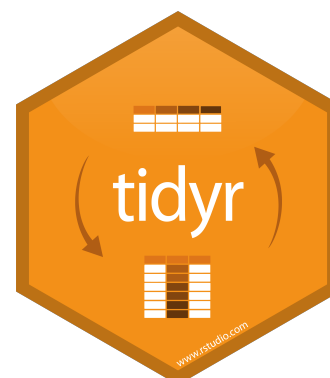
city	size	amount
New York	large	23
New York	small	14
London	large	22
London	small	16
Beijing	large	121
Beijing	small	56

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



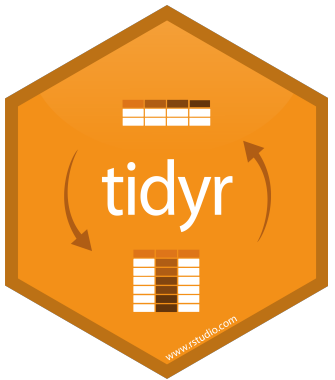
city	size	amount
New York	large	23
New York	small	14
London	large	22
London	small	16
Beijing	large	121
Beijing	small	56

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



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



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

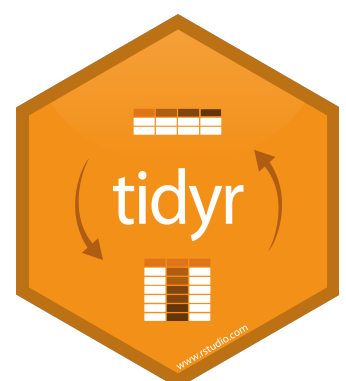


1

2

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

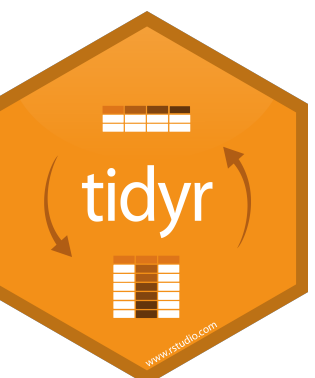
city	large	small
New York	23	14
London	22	16
Beijing	121	56



**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

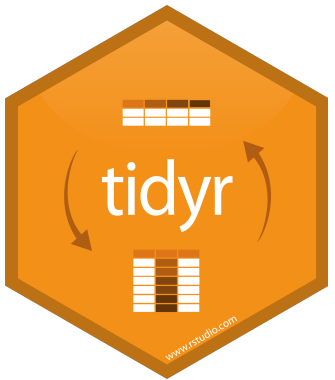
city	large	small
New York	23	14
London	22	16
Beijing	121	56



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



# spread()

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

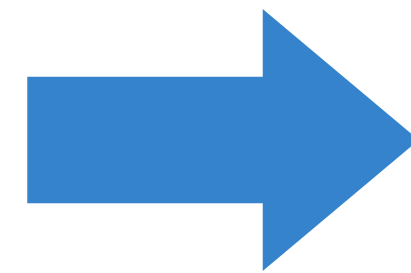
**data frame to  
reshape**

**column to use for keys**  
(becomes new  
column names)

**column to use for values**  
(becomes new  
column cells)

```
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

# Your Turn 4


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

03:00

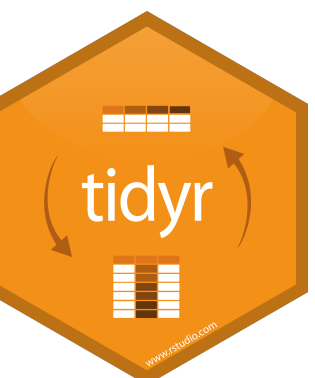


```
table2 %>%  
  spread(key = type, value = count)
```

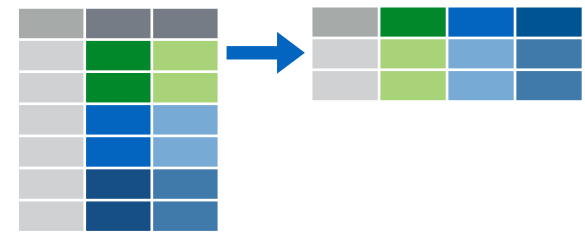


	<b>country</b> <chr>	<b>year</b> <int>	<b>cases</b> <int>	<b>population</b> <int>
1	Afghanistan	1999	745	19987071
2	Afghanistan	2000	2666	20595360
3	Brazil	1999	37737	172006362
4	Brazil	2000	80488	174504898
5	China	1999	212258	1272915272
6	China	2000	213766	1280428583

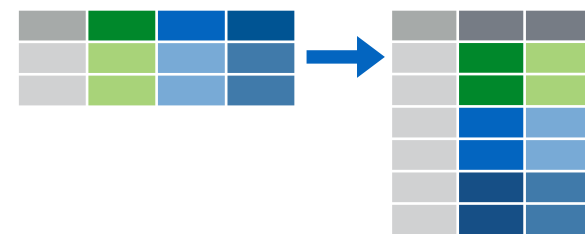
6 rows



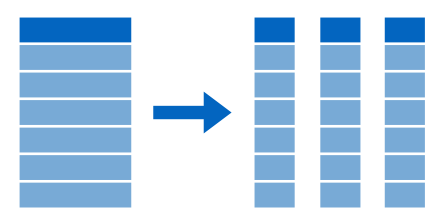
# Recap



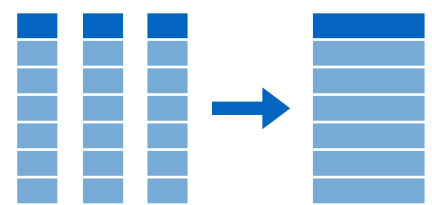
Move values into column names with **spread()**



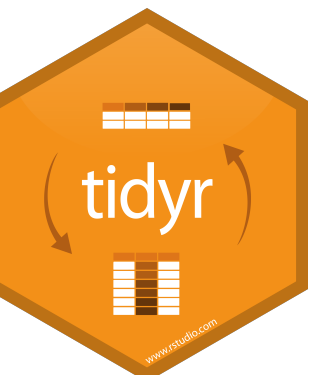
Move column names into values with **gather()**



Split a column with **separate()** or **separate\_rows()**



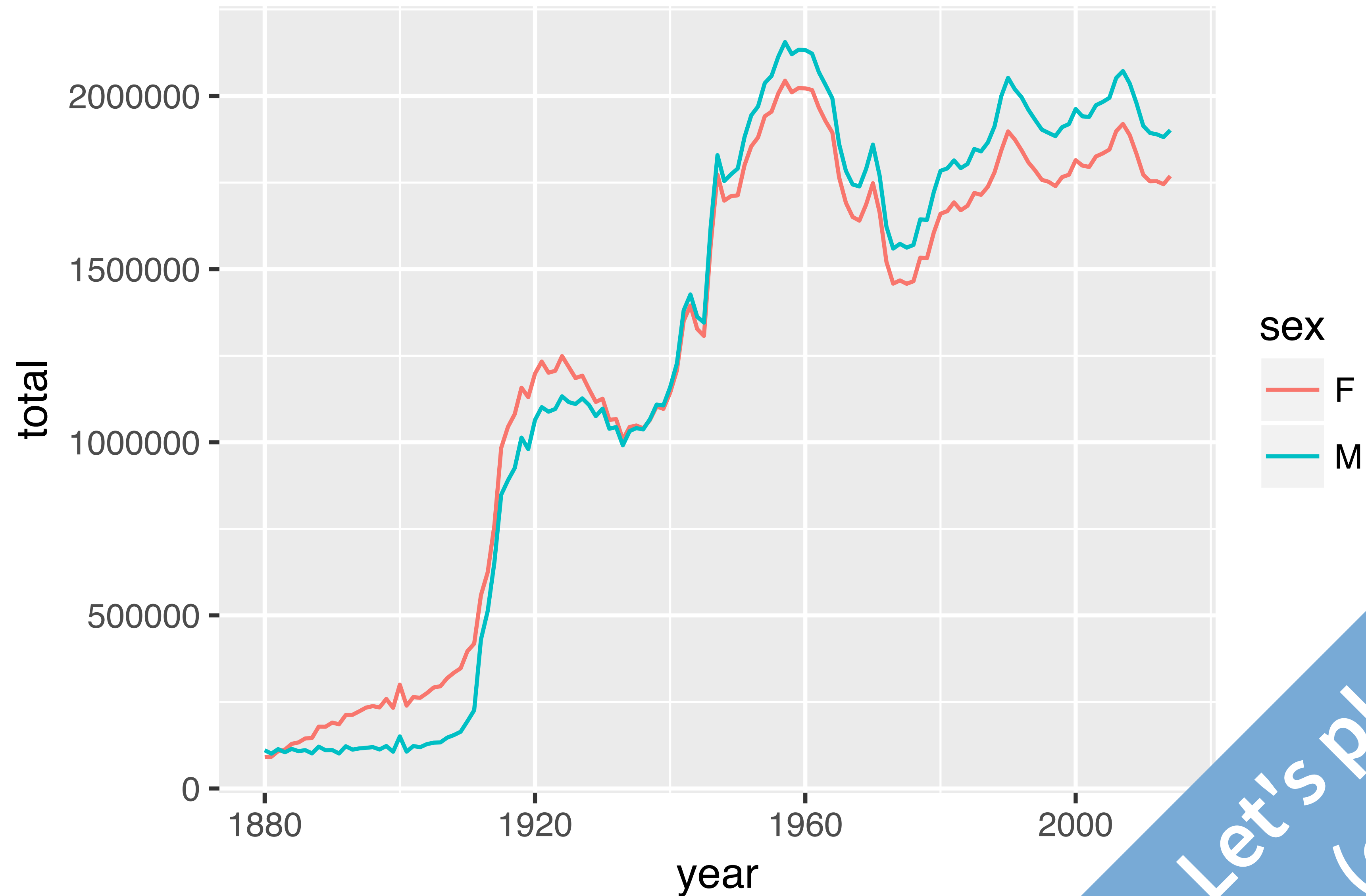
Unite columns with **unite()**



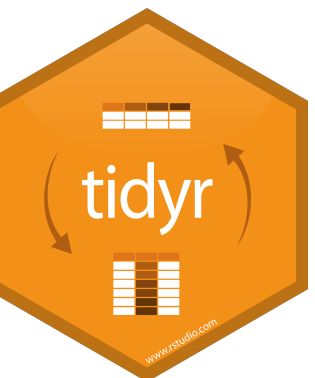
# Reshaping Final Exam



# Number of children by year and gender



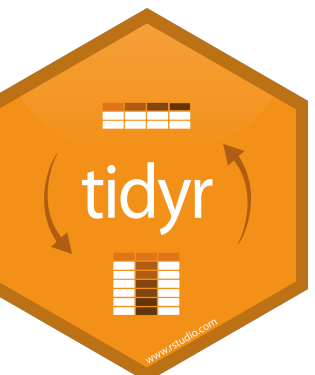
Let's plot % male  
(or female)



# Can we calculate the yearly percent of boys (or girls)?

```
babynames
```

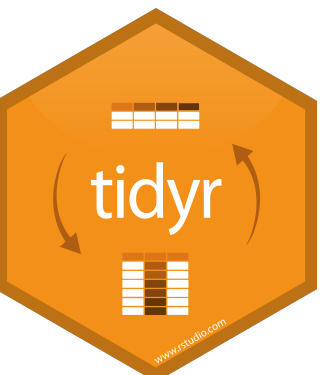
	year	sex	name	n	prop
	<dbl>	<chr>	<chr>	<int>	<dbl>
1	1880	F	Mary	7065	0.0724
2	1880	F	Anna	2604	0.0267
3	1880	F	Emma	2003	0.0205
4	1880	F	Elizabeth	1939	0.0199
5	1880	F	Minnie	1746	0.0179
6	1880	F	Margaret	1578	0.0162



# Can we calculate the yearly percent of boys (or girls)?

```
babynames %>%  
  group_by(year, sex) %>%  
  summarise(n = sum(n))
```

	year	sex	n
	<dbl>	<chr>	<int>
1	1880	F	90993
2	1880	M	110491
3	1881	F	91954
4	1881	M	100745
5	1882	F	107850
6	1882	M	113688





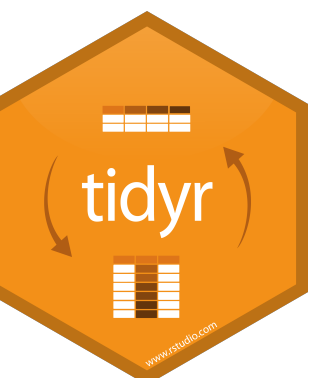
# Can we calculate the yearly percent of boys (or girls)?

```
babynames %>%  
  group_by(year, sex) %>%  
  summarise(n = sum(n))
```

	year	sex	n
	<dbl>	<chr>	<int>
1	1880	F	90993
2	1880	M	110491
3	1881	F	91954
4	1881	M	100745
5	1882	F	107850
6	1882	M	113688

$$\% \text{ male} = \frac{\text{male}}{\text{male} + \text{female}} \times 100$$

**Now  
what?**

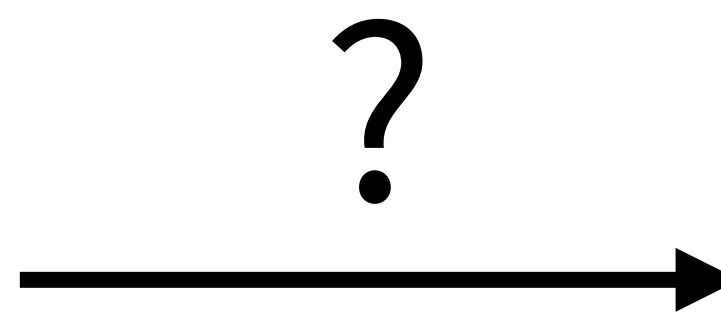


# Can we calculate the yearly percent of boys (or girls)?

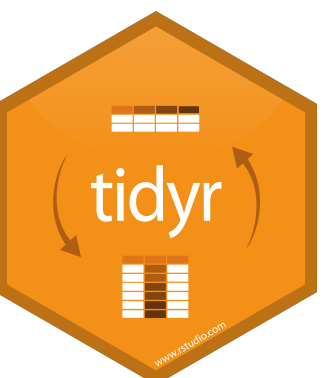
```
better_layout %>%
```

```
  mutate(percent_male = M / (M + F) * 100)
```

	year	sex	n
	<dbl>	<chr>	<int>
1	1880	F	90993
2	1880	M	110491
3	1881	F	91954
4	1881	M	100745
5	1882	F	107850
6	1882	M	113688



	year	F	M
	<dbl>	<int>	<int>
1	1880	90993	110491
2	1881	91954	100745
3	1882	107850	113688
4	1883	112321	104629
5	1884	129022	114445
6	1885	133055	107800



# Your Turn 4

05:00

Extend this code to reshape the data. Calculate the percent of male (or female) children by year. Then plot the percent over time.

```
babynames %>%
```

```
  group_by(year, sex) %>%
```

```
  summarise(n = sum(n))
```

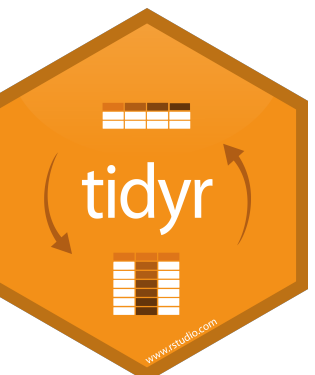
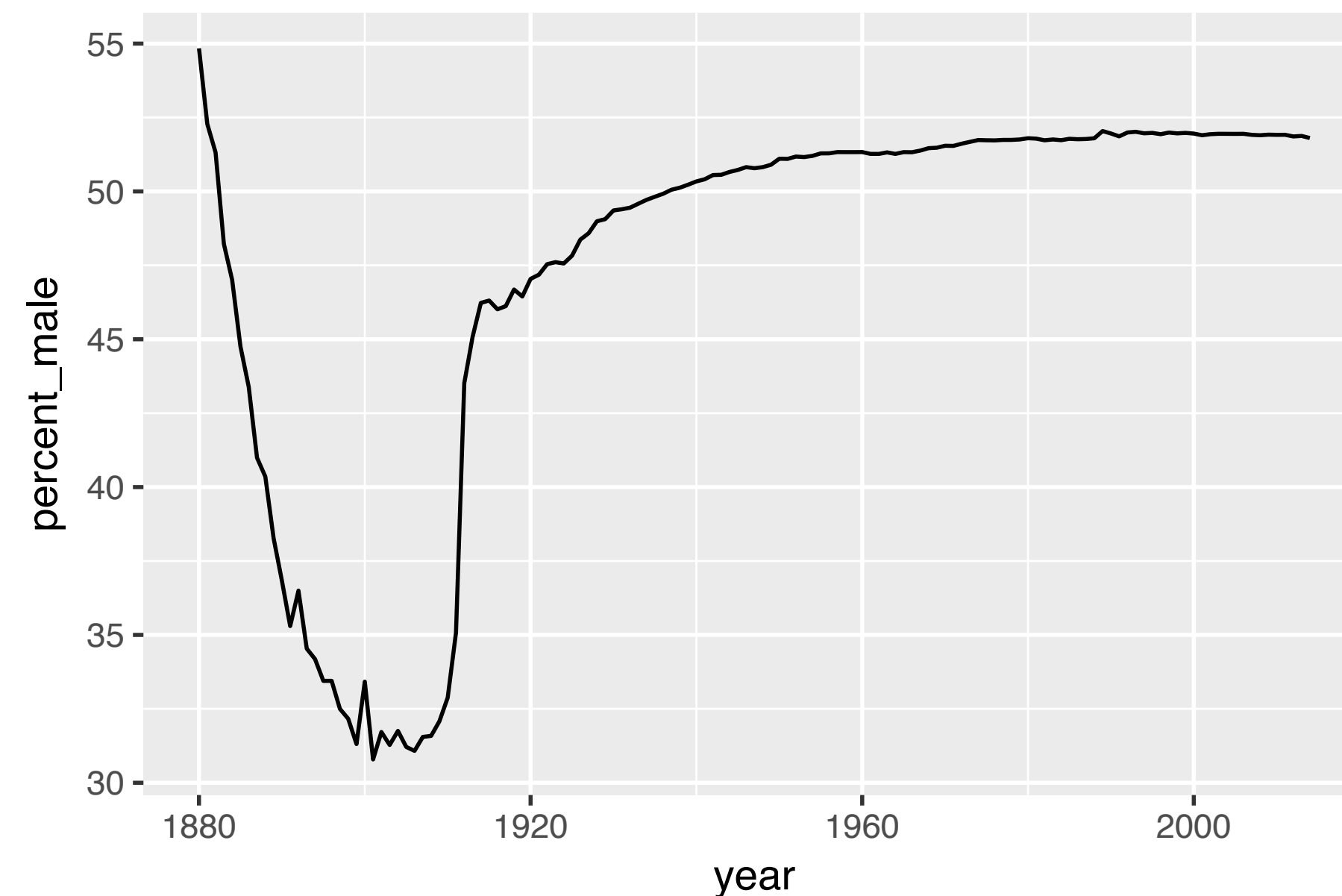
	year	sex	n
	<dbl>	<chr>	<int>
1	1880	F	90993
2	1880	M	110491
3	1881	F	91954

?



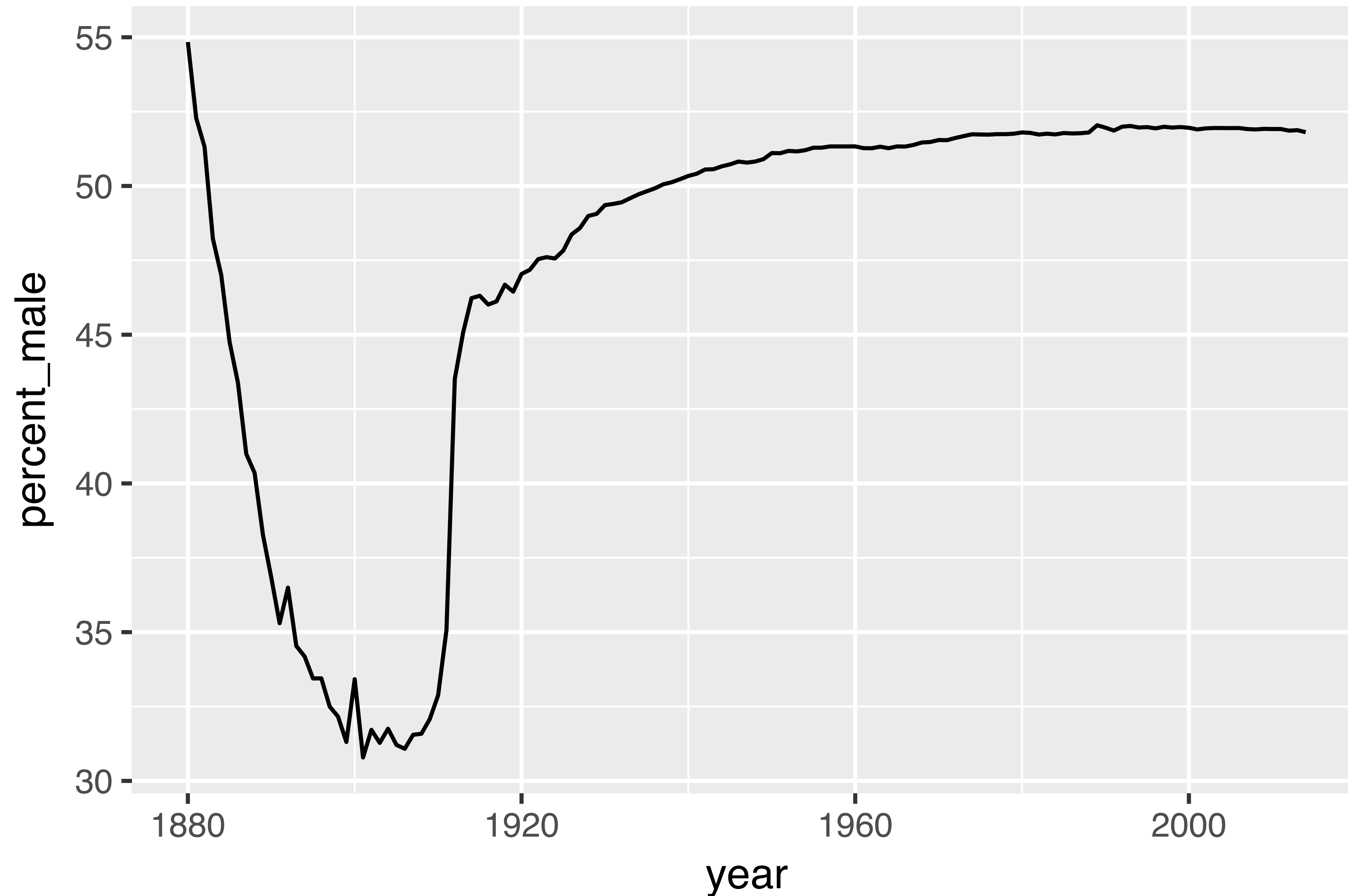
	year	F	M
	<dbl>	<int>	<int>
1	1880	90993	110491
2	1881	91954	100745
3	1882	107850	113688

```
babynames %>%  
  group_by(year, sex) %>%  
  summarise(n = sum(n)) %>%  
  spread(sex, n) %>%  
  mutate(percent_male = M / (M + F) * 100) %>%  
  ggplot(aes(year, percent_male)) + geom_line()
```





# Percent of children that are male by year

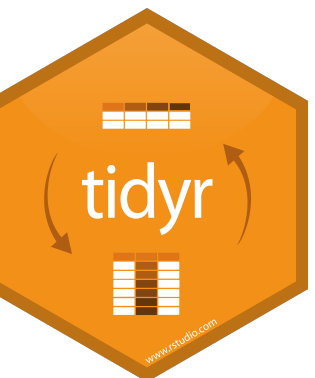




# General advice

Describe what you want to do in an **equation**. Each **variable** in the equation should correspond to a column in your data:

- "color by sex"  
**color** = **sex**
- "calculate the proportion of males"  
**prop** male = number of **males** / number of **females** + number of **males**



# Tidy Data with

