Data importing and tidy data II

Quantitative Methods for International Politics

IPOL 3270 • Fall 2026

September 22, 2026



Plan for today

Downloading, making, and reading data

Tidy data

Pivoting data

Downloading, making, and reading data

What does data look like?

Rectangles!

- CSV files (`.csv`)
- Excel files (`.xlsx`)
- Google Sheets
- Other programs like Stata (`.dta`)

CSV files

data > qatarcars.csv > data

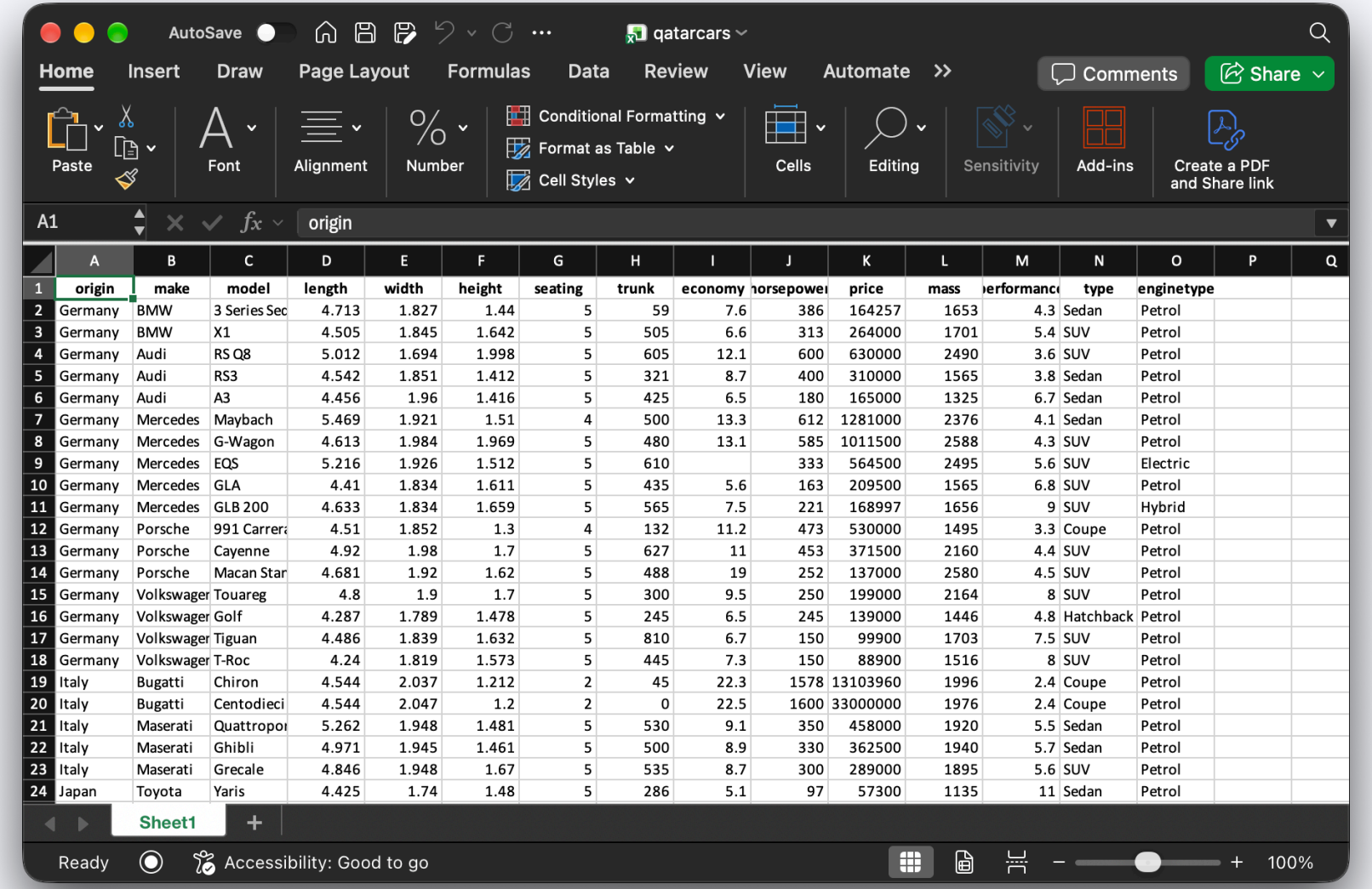
```
origin,make,model,length,width,height,seating,trunk,economy,horsepower,price,mass,performance,type,
Germany,BMW,3 Series Sedan,4.713,1.827,1.44,5,59,7.6,386,164257,1653,4.3,Sedan,Petrol
Germany,BMW,X1,4.505,1.845,1.642,5,505,6.6,313,264000,1701,5.4,SUV,Petrol
Germany,Audi,RS Q8,5.012,1.694,1.998,5,605,12.1,600,630000,2490,3.6,SUV,Petrol
Germany,Audi,RS3,4.542,1.851,1.412,5,321,8.7,400,310000,1565,3.8,Sedan,Petrol
Germany,Audi,A3,4.456,1.96,1.416,5,425,6.5,180,165000,1325,6.7,Sedan,Petrol
Germany,Mercedes,Maybach,5.469,1.921,1.51,4,500,13.3,612,1281000,2376,4.1,Sedan,Petrol
Germany,Mercedes,G-Wagon,4.613,1.984,1.969,5,480,13.1,585,1011500,2588,4.3,SUV,Petrol
Germany,Mercedes,EQS,5.216,1.926,1.512,5,610,,333,564500,2495,5.6,SUV,Electric
Germany,Mercedes,GLA,4.41,1.834,1.611,5,435,5.6,163,209500,1565,6.8,SUV,Petrol
Germany,Mercedes,GLB 200,4.633,1.834,1.659,5,565,7.5,221,168997,1656,9,SUV,Hybrid
Germany,Porsche,991 Carrera S,4.51,1.852,1.3,4,132,11.2,473,530000,1495,3.3,Coupe,Petrol
Germany,Porsche,Cayenne,4.92,1.98,1.7,5,627,11,453,371500,2160,4.4,SUV,Petrol
Germany,Porsche,Macan Standard,4.681,1.92,1.62,5,488,19,252,137000,2580,4.5,SUV,Petrol
Germany,Volkswagen,Touareg,4.8,1.9,1.7,5,300,9.5,250,199000,2164,8,SUV,Petrol
Germany,Volkswagen,Golf,4.287,1.789,1.478,5,245,6.5,245,139000,1446,4.8,Hatchback,Petrol
Germany,Volkswagen,Tiguan,4.486,1.839,1.632,5,810,6.7,150,99900,1703,7.5,SUV,Petrol
Germany,Volkswagen,T-Roc,4.24,1.819,1.573,5,445,7.3,150,88900,1516,8,SUV,Petrol
Italy,Bugatti,Chiron,4.544,2.037,1.212,2,45,22.3,1578,13103960,1996,2.4,Coupe,Petrol
Italy,Bugatti,Centodieci,4.544,2.047,1.2,2,0,22.5,1600,33000000,1976,2.4,Coupe,Petrol
Italy,Maserati,Quattroporte,5.262,1.948,1.481,5,530,9.1,350,458000,1920,5.5,Sedan,Petrol
Italy,Maserati,Ghibli,4.971,1.945,1.461,5,500,8.9,330,362500,1940,5.7,Sedan,Petrol
Italy,Maserati,Grecale,4.846,1.948,1.67,5,535,8.7,300,289000,1895,5.6,SUV,Petrol
Japan,Toyota,Yaris,4.425,1.74,1.48,5,286,5.1,97,57300,1135,11,Sedan,Petrol
Japan,Toyota,Raize,4.03,1.71,1.635,5,369,5,87,57700,1680,12,SUV,Petrol
Japan,Toyota,RAV4 Adventure,4.61,1.865,1.69,5,1085,7.3,203,130900,1583,8.2,SUV,Petrol
Japan,Toyota,Rav4 Hybrid,4.6,1.855,1.685,5,1085,4.1,219,117700,1674,7.5,SUV,Hybrid
Japan,Toyota,Fortuner,4.795,1.855,1.835,7,348,12.4,164,124000,2620,12,SUV,Petrol
```

CSV files

```
library(tidyverse)

my_data <- read_csv("WHATEVER.csv")
```

Excel files



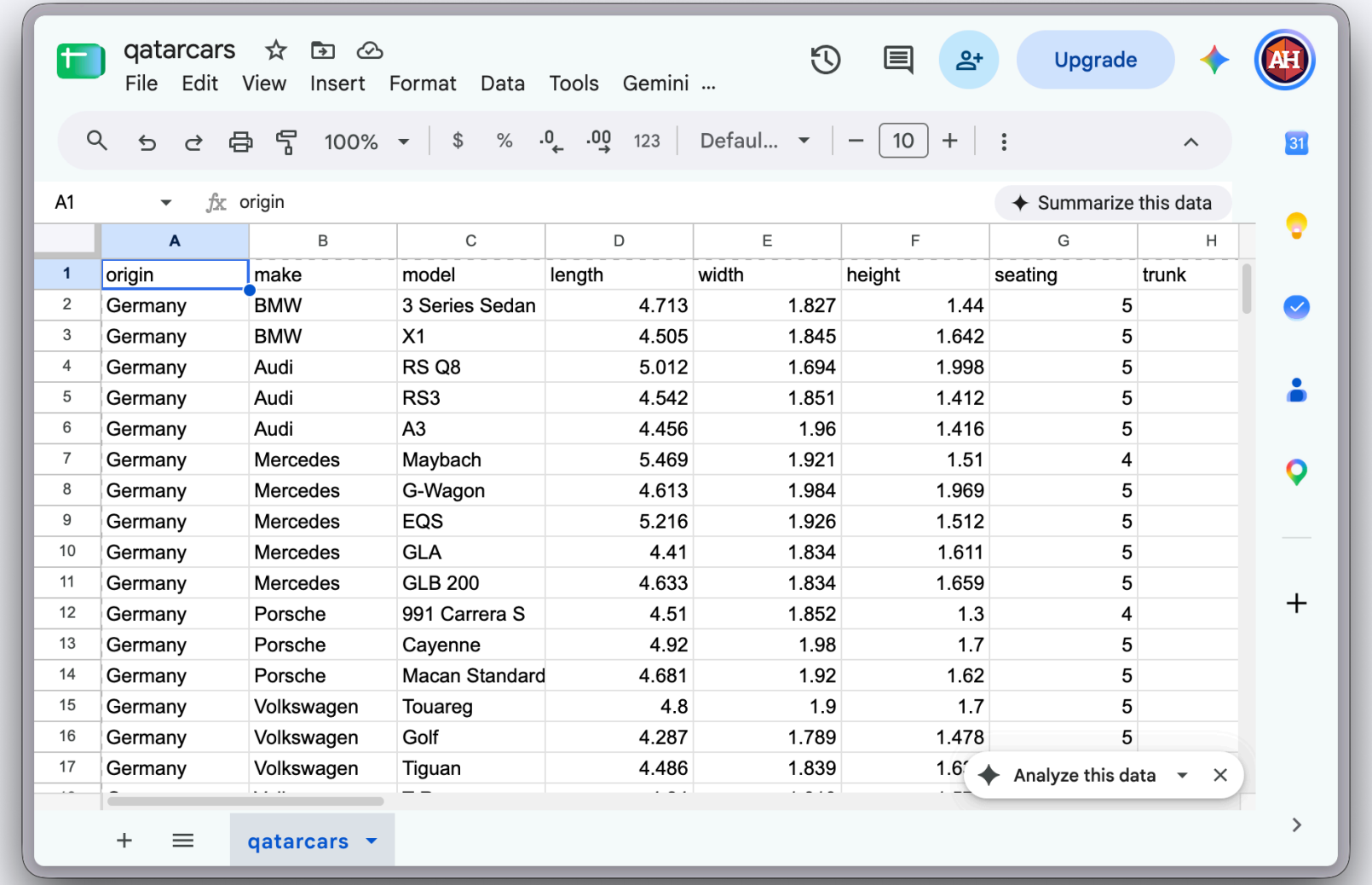
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	origin	make	model	length	width	height	seating	trunk	economy	horsepowe	price	mass	erformanc	type	enginetype		
2	Germany	BMW	3 Series Sec	4.713	1.827	1.44	5	59	7.6	386	164257	1653	4.3	Sedan	Petrol		
3	Germany	BMW	X1	4.505	1.845	1.642	5	505	6.6	313	264000	1701	5.4	SUV	Petrol		
4	Germany	Audi	RS Q8	5.012	1.694	1.998	5	605	12.1	600	630000	2490	3.6	SUV	Petrol		
5	Germany	Audi	RS3	4.542	1.851	1.412	5	321	8.7	400	310000	1565	3.8	Sedan	Petrol		
6	Germany	Audi	A3	4.456	1.96	1.416	5	425	6.5	180	165000	1325	6.7	Sedan	Petrol		
7	Germany	Mercedes	Maybach	5.469	1.921	1.51	4	500	13.3	612	1281000	2376	4.1	Sedan	Petrol		
8	Germany	Mercedes	G-Wagon	4.613	1.984	1.969	5	480	13.1	585	1011500	2588	4.3	SUV	Petrol		
9	Germany	Mercedes	EQS	5.216	1.926	1.512	5	610		333	564500	2495	5.6	SUV	Electric		
10	Germany	Mercedes	GLA	4.41	1.834	1.611	5	435	5.6	163	209500	1565	6.8	SUV	Petrol		
11	Germany	Mercedes	GLB 200	4.633	1.834	1.659	5	565	7.5	221	168997	1656	9	SUV	Hybrid		
12	Germany	Porsche	991 Carrera	4.51	1.852	1.3	4	132	11.2	473	530000	1495	3.3	Coupe	Petrol		
13	Germany	Porsche	Cayenne	4.92	1.98	1.7	5	627	11	453	371500	2160	4.4	SUV	Petrol		
14	Germany	Porsche	Macan Star	4.681	1.92	1.62	5	488	19	252	137000	2580	4.5	SUV	Petrol		
15	Germany	Volkswager	Touareg	4.8	1.9	1.7	5	300	9.5	250	199000	2164	8	SUV	Petrol		
16	Germany	Volkswager	Golf	4.287	1.789	1.478	5	245	6.5	245	139000	1446	4.8	Hatchback	Petrol		
17	Germany	Volkswager	Tiguan	4.486	1.839	1.632	5	810	6.7	150	99900	1703	7.5	SUV	Petrol		
18	Germany	Volkswager	T-Roc	4.24	1.819	1.573	5	445	7.3	150	88900	1516	8	SUV	Petrol		
19	Italy	Bugatti	Chiron	4.544	2.037	1.212	2	45	22.3	1578	13103960	1996	2.4	Coupe	Petrol		
20	Italy	Bugatti	Centodieci	4.544	2.047	1.2	2	0	22.5	1600	33000000	1976	2.4	Coupe	Petrol		
21	Italy	Maserati	Quattropor	5.262	1.948	1.481	5	530	9.1	350	458000	1920	5.5	Sedan	Petrol		
22	Italy	Maserati	Ghibli	4.971	1.945	1.461	5	500	8.9	330	362500	1940	5.7	Sedan	Petrol		
23	Italy	Maserati	Grecale	4.846	1.948	1.67	5	535	8.7	300	289000	1895	5.6	SUV	Petrol		
24	Japan	Toyota	Yaris	4.425	1.74	1.48	5	286	5.1	97	57300	1135	11	Sedan	Petrol		

Excel files

```
library(readxl)

my_data <- read_excel("WHATEVER.xlsx", sheet = "Sheet1")
```

Google Sheets



	A	B	C	D	E	F	G	H
1	origin	make	model	length	width	height	seating	trunk
2	Germany	BMW	3 Series Sedan	4.713	1.827	1.44	5	
3	Germany	BMW	X1	4.505	1.845	1.642	5	
4	Germany	Audi	RS Q8	5.012	1.694	1.998	5	
5	Germany	Audi	RS3	4.542	1.851	1.412	5	
6	Germany	Audi	A3	4.456	1.96	1.416	5	
7	Germany	Mercedes	Maybach	5.469	1.921	1.51	4	
8	Germany	Mercedes	G-Wagon	4.613	1.984	1.969	5	
9	Germany	Mercedes	EQS	5.216	1.926	1.512	5	
10	Germany	Mercedes	GLA	4.41	1.834	1.611	5	
11	Germany	Mercedes	GLB 200	4.633	1.834	1.659	5	
12	Germany	Porsche	991 Carrera S	4.51	1.852	1.3	4	
13	Germany	Porsche	Cayenne	4.92	1.98	1.7	5	
14	Germany	Porsche	Macan Standard	4.681	1.92	1.62	5	
15	Germany	Volkswagen	Touareg	4.8	1.9	1.7	5	
16	Germany	Volkswagen	Golf	4.287	1.789	1.478	5	
17	Germany	Volkswagen	Tiguan	4.486	1.839	1.6		

Google Sheets

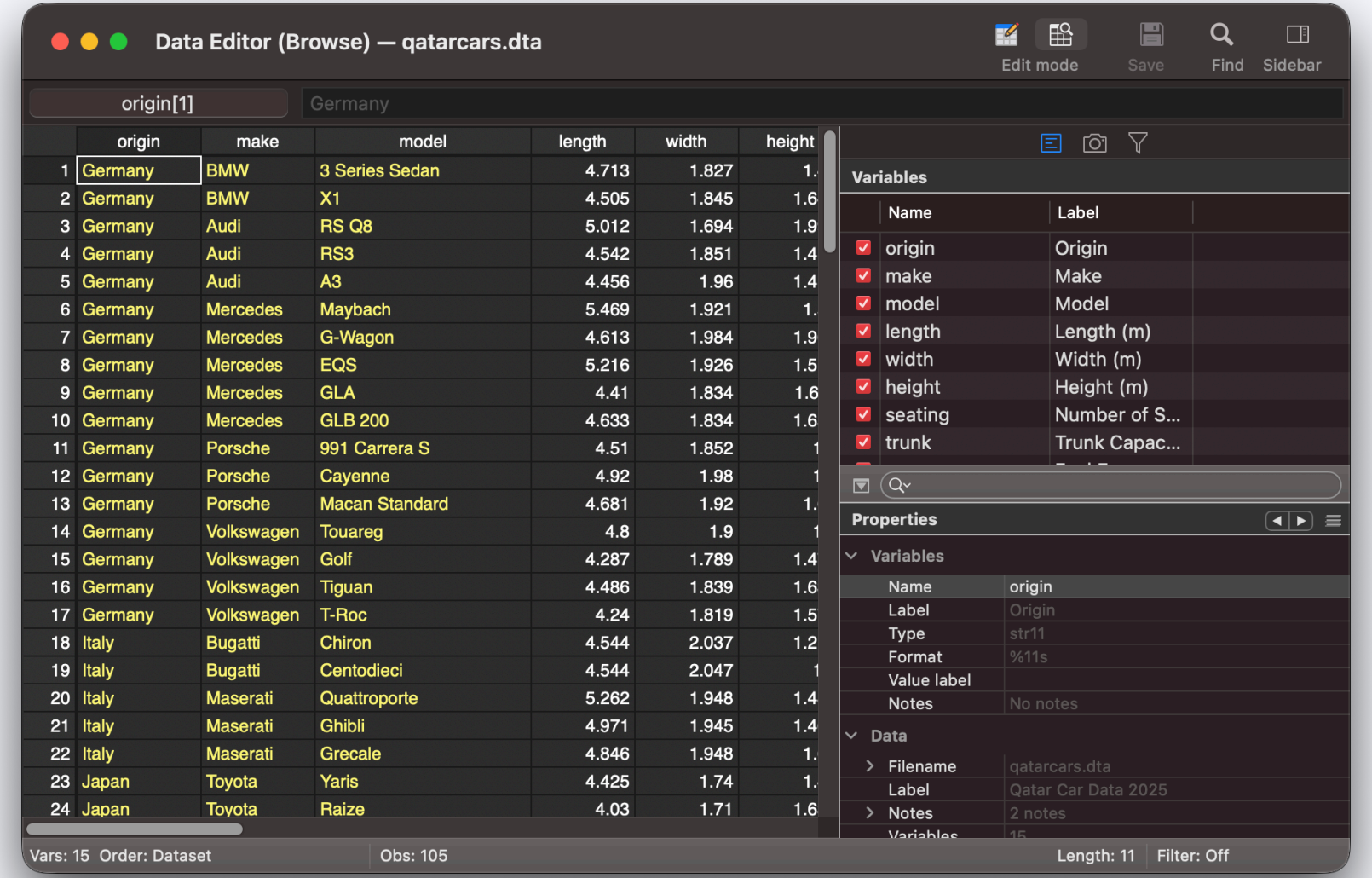
```
library(googlesheets4)

gs4_deauth()

# https://docs.google.com/spreadsheets/d/196A3XNQFSlyRvBdA
my_data <- read_sheet(
  "196A3XNQFSlyRvBdAu5DRPyI9woyMCwWK0lGNzo4Wcg8"
)
```

(Though often easier to just download as CSV)

Other programs (Stata, SAS, SPSS)

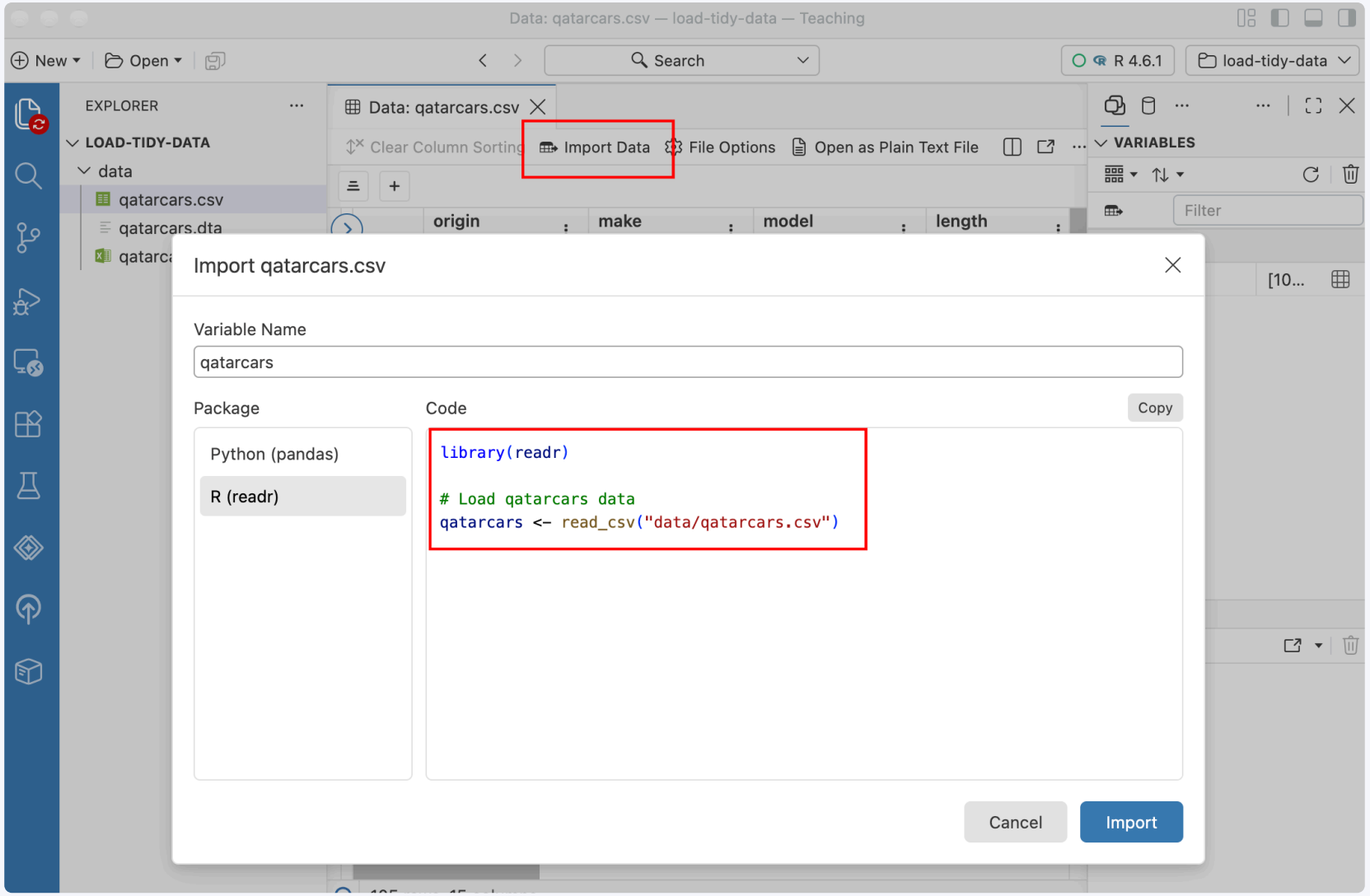


Other programs (Stata, SAS, SPSS)

```
library(haven)

my_data <- read_stata("WHATEVER.dta")
```

Let Positron help you!



Your/our turn #0

- Create a new empty Positron project!
- Download the `example-data.zip` folder from the class website, unzip it, and put it in your project

Your turn #1

1. Download a dataset from Qatar's Open Data Catalog: https://www.data.gov.qa/explore/
2. Place the downloaded file in your Positron project folder
3. Load it into R
4. For bonus fun, make a plot!

07:00

Tidy data

What is tidy data?

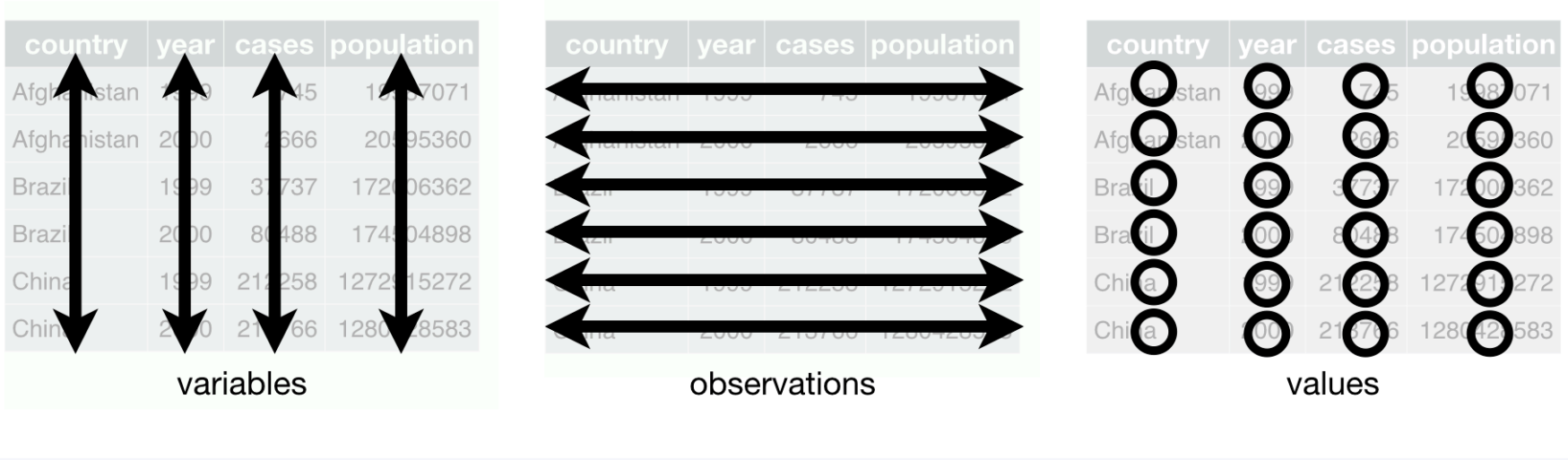
~~Clean, perfect data~~

Data shaped to make computers happy

Each variable is a column

Each observation is a row

Each cell is a single measurement



Identification vs. measurement

Identification variables uniquely identify each observation

country	year	lifeExp	gdpPercap	pop
Afghanistan	1952	28.801	779.4453	8425333
Afghanistan	1957	30.332	820.8530	9240934
Afghanistan	1962	31.997	853.1007	10267083
Afghanistan	1967	34.020	836.1971	11537966
Afghanistan	1972	36.088	739.9811	13079460
Afghanistan	1977	38.438	786.1134	14880372
Afghanistan	1982	39.854	978.0114	12881816
Afghanistan	1987	40.822	852.3959	13867957
Afghanistan	1992	41.674	649.3414	16317921
Afghanistan	1997	41.763	635.3414	22227415

Identification:

`country`, `year`

Measurement:

`lifeExp`, `gdpPercap`, `pop`

Untidy data for humans

student	exercise1	exercise2	quiz1
Ulyaa	42	88	95
Renee	91	96	82
Siddeeqa	21	37	89
Valentin	38	95	56

assessment	Ulyaa	Renee	Siddeeqa	Valentin
exercise1	42	91	21	38
exercise2	88	96	37	95
quiz1	95	82	89	56

Untidy data for humans

student	exercise1	exercise2	quiz1
Ulyaa	42	88	95
Renee	91	96	82
Siddeeqa	21	37	89
Valentin	38	95	56

- What are the variables here?
- Which are identification variables?
- Which are measurement variables?

Untidy data for humans

student	exercise1	exercise2	quiz1
Ulyaa	42	88	95
Renee	91	96	82
Siddeeqa	21	37	89
Valentin	38	95	56

- What are the variables here?
  - **Student**, **Assessment**, **Grade**
- Which are identification variables?
  - **Student**, **Assessment**
- Which are measurement variables?
  - **Grade**

Tidy data for computers

student	assessment	grade
Ulyaa	exercise1	42
Ulyaa	exercise2	88
Ulyaa	quiz1	95
Renee	exercise1	91
Renee	exercise2	96
Renee	quiz1	82
Siddeeqa	exercise1	21
Siddeeqa	exercise2	37
Siddeeqa	quiz1	89
Valentin	exercise1	38
Valentin	exercise2	95
Valentin	quiz1	56

- What are the variables here?
  - **Student**, **Assessment**, **Grade**
- Which are identification variables?
  - **Student**, **Assessment**
- Which are measurement variables?
  - **Grade**

Same data!

Wide data

student	exercise1	exercise2	quiz1
Ulyaa	42	88	95
Renee	91	96	82
Siddeeqa	21	37	89
Valentin	38	95	56

Long data

student	assessment	grade
Ulyaa	exercise1	42
Ulyaa	exercise2	88
Ulyaa	quiz1	95
Renee	exercise1	91
Renee	exercise2	96
Renee	quiz1	82
Siddeeqa	exercise1	21
Siddeeqa	exercise2	37
Siddeeqa	quiz1	89
Valentin	exercise1	38
Valentin	exercise2	95
Valentin	quiz1	56

Untidy data isn't bad!

Who had the biggest improvement in exercise scores?

Who scored the lowest on quiz 1?

student	exercise1	exercise2	quiz1
Ulyaa	42	88	95
Renee	91	96	82
Siddeeqa	21	37	89
Valentin	38	95	56

student	assessment	grade
Ulyaa	exercise1	42
Ulyaa	exercise2	88
Ulyaa	quiz1	95
Renee	exercise1	91
Renee	exercise2	96
Renee	quiz1	82
Siddeeqa	exercise1	21
Siddeeqa	exercise2	37
Siddeeqa	quiz1	89
Valentin	exercise1	38
Valentin	exercise2	95
Valentin	quiz1	56

Tidy data easier to manipulate

```
classroom
```

```
# A tibble: 4 × 4
  student  exercise1 exercise2 quiz1
  <chr>        <dbl>     <dbl> <dbl>
1 Ulyaa           42        88    95
2 Renee           91        96    82
3 Siddeeqa        21        37    89
4 Valentin        38        95    56
```

```
classroom_long
```

```
# A tibble: 12 × 3
   student  assessment grade
   <chr>    <chr>      <dbl>
 1 Ulyaa    exercise1     42
 2 Ulyaa    exercise2     88
 3 Ulyaa    quiz1         95
 4 Renee    exercise1     91
 5 Renee    exercise2     96
 6 Renee    quiz1         82
 7 Siddeeqa exercise1     21
 8 Siddeeqa exercise2     37
 9 Siddeeqa quiz1         89
10 Valentin exercise1     38
11 Valentin exercise2     95
12 Valentin quiz1         56
```

```
classroom_long |>
  group_by(student) |>
  summarize(
    avg_grade = mean(grade),
    med_grade = median(grade),
    total = sum(grade)
  ) |>
  mutate(pct = total / 300)
```

```
# A tibble: 4 × 5
  student  avg_grade med_grade total   pct
  <chr>        <dbl>     <dbl> <dbl> <dbl>
1 Renee         89.7        91   269 0.897
2 Siddeeqa      49          37   147 0.49
3 Ulyaa         75          88   225 0.75
4 Valentin      63          56   189 0.63
```

Tidy data easier to plot

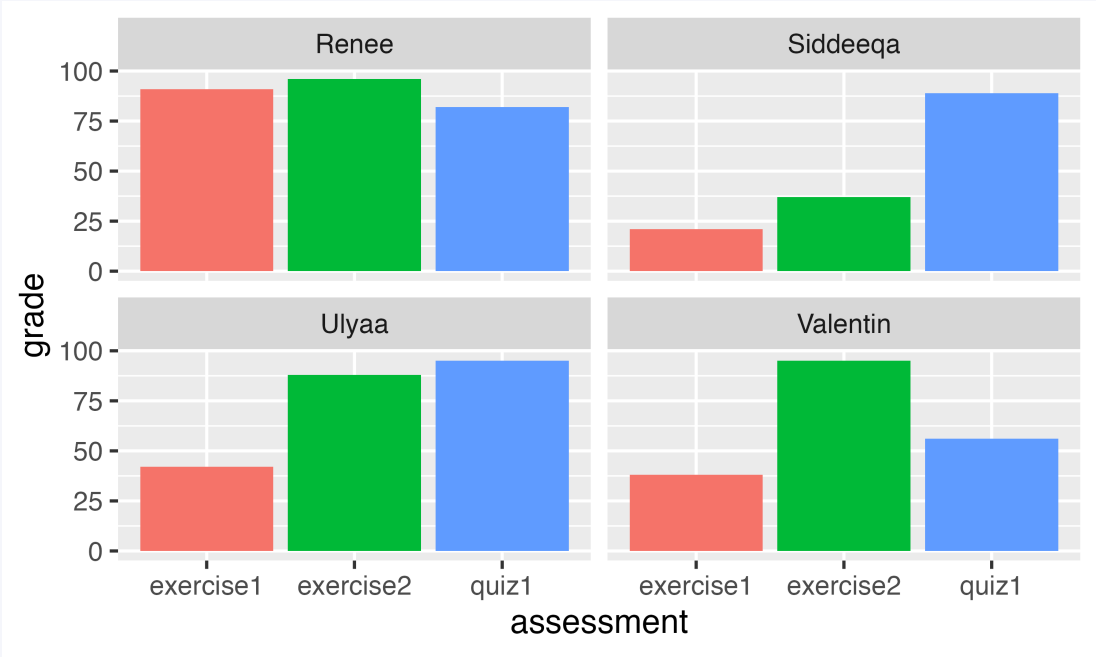
```
classroom
```

```
# A tibble: 4 × 4
  student  exercise1 exercise2 quiz1
  <chr>        <dbl>     <dbl> <dbl>
1 Ulyaa           42        88    95
2 Renee           91        96    82
3 Siddeeqa        21        37    89
4 Valentin        38        95    56
```

```
classroom_long
```

```
# A tibble: 12 × 3
   student  assessment grade
   <chr>    <chr>      <dbl>
 1 Ulyaa    exercise1     42
 2 Ulyaa    exercise2     88
 3 Ulyaa    quiz1         95
 4 Renee    exercise1     91
 5 Renee    exercise2     96
 6 Renee    quiz1         82
 7 Siddeeqa exercise1     21
 8 Siddeeqa exercise2     37
 9 Siddeeqa quiz1         89
10 Valentin exercise1     38
11 Valentin exercise2     95
12 Valentin quiz1         56
```

```
classroom_long |>
  ggplot(aes(
    x = assessment, y = grade, fill = assessment
  )) +
  geom_col() +
  guides(fill = "none") +
  facet_wrap(vars(student))
```



Common types of untidy data

- Column headers are values, not variable names
- Variables and values are not in rows or columns
- Multiple variables are stored in one column

Headers are values

religion	<$10k	$10-20k	$20-30k	$30-40k	$40-50k	$50-75k
Agnostic	27	34	60	81	76	137
Atheist	12	27	37	52	35	70
Buddhist	27	21	30	34	33	58
Catholic	418	617	732	670	638	1116
Don’t know/refused	15	14	15	11	10	35
Evangelical Prot	575	869	1064	982	881	1486
Hindu	1	9	7	9	11	34
Historically Black Prot	228	244	236	238	197	223
Jehovah's Witness	20	27	24	24	21	30
Jewish	19	19	25	25	30	95
Mainline Prot	289	495	619	655	651	1107
Mormon	29	40	48	51	56	112
Muslim	6	7	9	10	9	23
Orthodox	13	17	23	32	32	47

Headers are values

religion	income	count
Agnostic	<$10k	27
Agnostic	$10-20k	34
Agnostic	$20-30k	60
Agnostic	$30-40k	81
Agnostic	$40-50k	76
Agnostic	$50-75k	137
Atheist	<$10k	12
Atheist	$10-20k	27
Atheist	$20-30k	37
Atheist	$30-40k	52
Atheist	$40-50k	35
Atheist	$50-75k	70
Buddhist	<$10k	27
Buddhist	$10-20k	21

Variables not in table

student	exercise1	exercise2	quiz1
Ulyaa	*42*	88	**95**
Renee	**91**	**96**	82
Siddeeqa	*21*	*37*	89
Valentin	*38*	**95**	56

Key
Junior
Senior
Surprisingly high
Surprisingly low

Variables not in table

student	assessment	grade	class	surprise
Ulyaa	exercise1	42	Junior	Low surprise
Ulyaa	exercise2	88	Junior	
Ulyaa	quiz1	95	Junior	High surprise
Renee	exercise1	91	Senior	High surprise
Renee	exercise2	96	Senior	High surprise
Renee	quiz1	82	Senior	
Siddeeqa	exercise1	21	Junior	Low surprise
Siddeeqa	exercise2	37	Junior	Low surprise
Siddeeqa	quiz1	89	Junior	
Valentin	exercise1	38	Junior	Low surprise
Valentin	exercise2	95	Junior	High surprise
Valentin	quiz1	56	Junior	

Multiple variables in column

iso2	year	m014	m1524	m2534	m3544	f014	f1524	f2534	f3544
AD	1996	0	0	0	4	0	1	1	0
AD	1997	0	0	1	2	0	1	2	3
AD	1998	0	0	0	1	NA	NA	NA	NA
AD	1999	0	0	0	1	0	0	0	1
AD	2000	0	0	1	0	NA	NA	NA	NA
AD	2001	0	NA	NA	2	NA	NA	NA	NA
AD	2002	0	0	0	1	0	1	0	0
AD	2003	0	0	0	1	0	1	1	1
AD	2004	0	0	0	1	0	0	1	0
AD	2005	0	0	1	1	0	1	1	1
AD	2006	0	1	1	2	0	0	1	0
AD	2007	NA	NA	NA	NA	NA	NA	1	NA

Multiple variables in column

iso2	year	sex	age	n
AD	1996	m	014	0
AD	1996	m	1524	0
AD	1996	m	2534	0
AD	1996	m	3544	4
AD	1996	f	014	0
AD	1996	f	1524	1
AD	1996	f	2534	1
AD	1996	f	3544	0
AD	1997	m	014	0
AD	1997	m	1524	0
AD	1997	m	2534	1
AD	1997	m	3544	2

Your turn #2

country	score_vault	score_floor
United States	46.866	45.999
Russia	45.733	42.032
China	44.332	42.066

02:00

Your turn #2

country	event	score
United States	vault	46.866
United States	floor	45.999
Russia	vault	45.733
Russia	floor	42.032
China	vault	44.332
China	floor	42.066

Your turn #3

country	1957	1967	1977	1987	1997	2007
Afghanistan	30.332	34.020	38.438	40.822	41.763	43.828
Albania	59.280	66.220	68.930	72.000	72.950	76.423
Algeria	45.685	51.407	58.014	65.799	69.152	72.301
Angola	31.999	35.985	39.483	39.906	40.963	42.731
Argentina	64.399	65.634	68.481	70.774	73.275	75.320
Australia	70.330	71.100	73.490	76.320	78.830	81.235
Austria	67.480	70.140	72.170	74.940	77.510	79.829
Bahrain	53.832	59.923	65.593	70.750	73.925	75.635

03:00

Your turn #3

country	year	lifeExp
Afghanistan	1957	30.332
Afghanistan	1967	34.020
Afghanistan	1977	38.438
Afghanistan	1987	40.822
Afghanistan	1997	41.763
Afghanistan	2007	43.828
Albania	1957	59.280
Albania	1967	66.220
Albania	1977	68.930

Your turn #4

country	vault_2012	floor_2012	vault_2016	floor_2016
United States	48.132	45.366	46.866	45.999
Russia	46.366	41.599	45.733	42.032
China	44.266	40.833	44.332	42.066

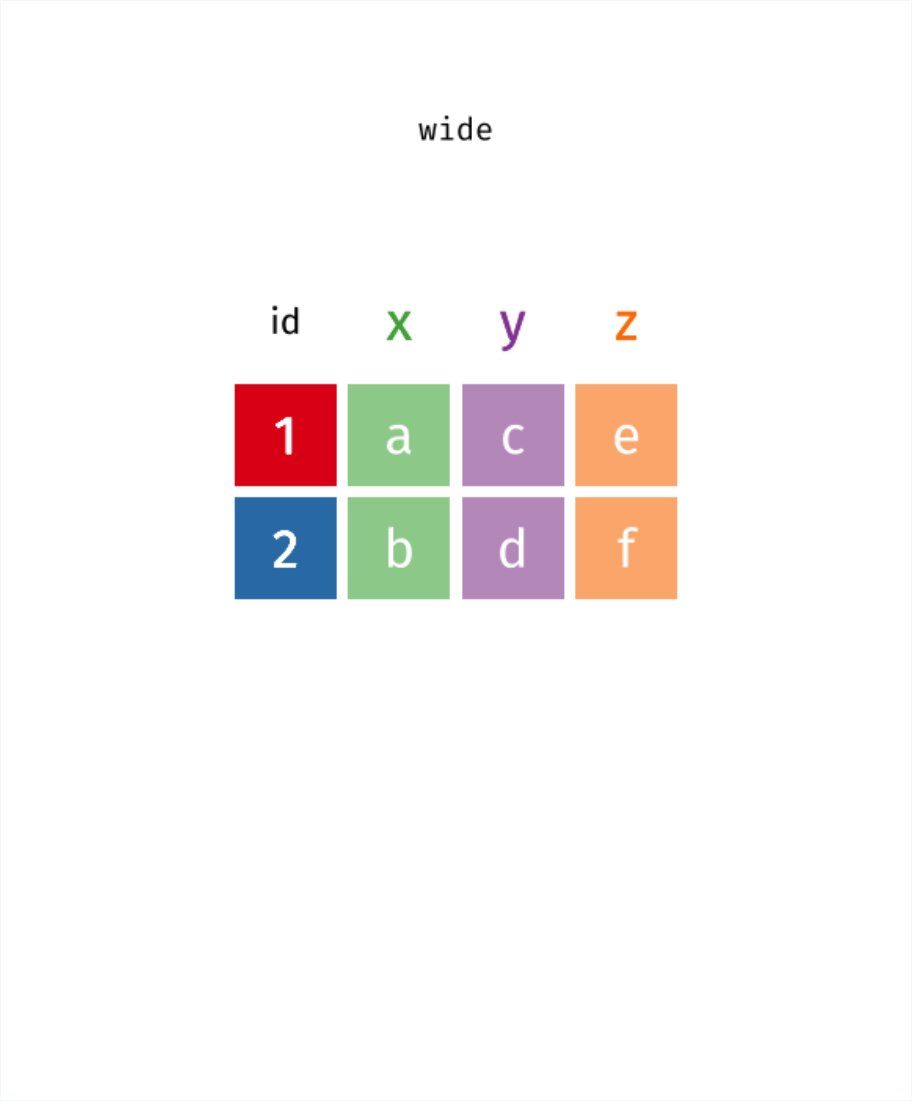
04:00

Your turn #4

country	event	year	score
United States	vault	2012	48.132
United States	floor	2012	45.366
United States	vault	2016	46.866
United States	floor	2016	45.999
Russia	vault	2012	46.366
Russia	floor	2012	41.599
Russia	vault	2016	45.733
Russia	floor	2016	42.032
China	vault	2012	44.266
China	floor	2012	40.833
China	vault	2016	44.332
China	floor	2016	42.066

Pivoting

Pivoting longer




```
wide_data |>
  pivot_longer(
    # Columns to move
    cols = BLAH,

    # Name of new
    # identification variable
    names_to = "key",

    # Name of new
    # measurement variable
    values_to = "value"
  )
```

Pivoting longer

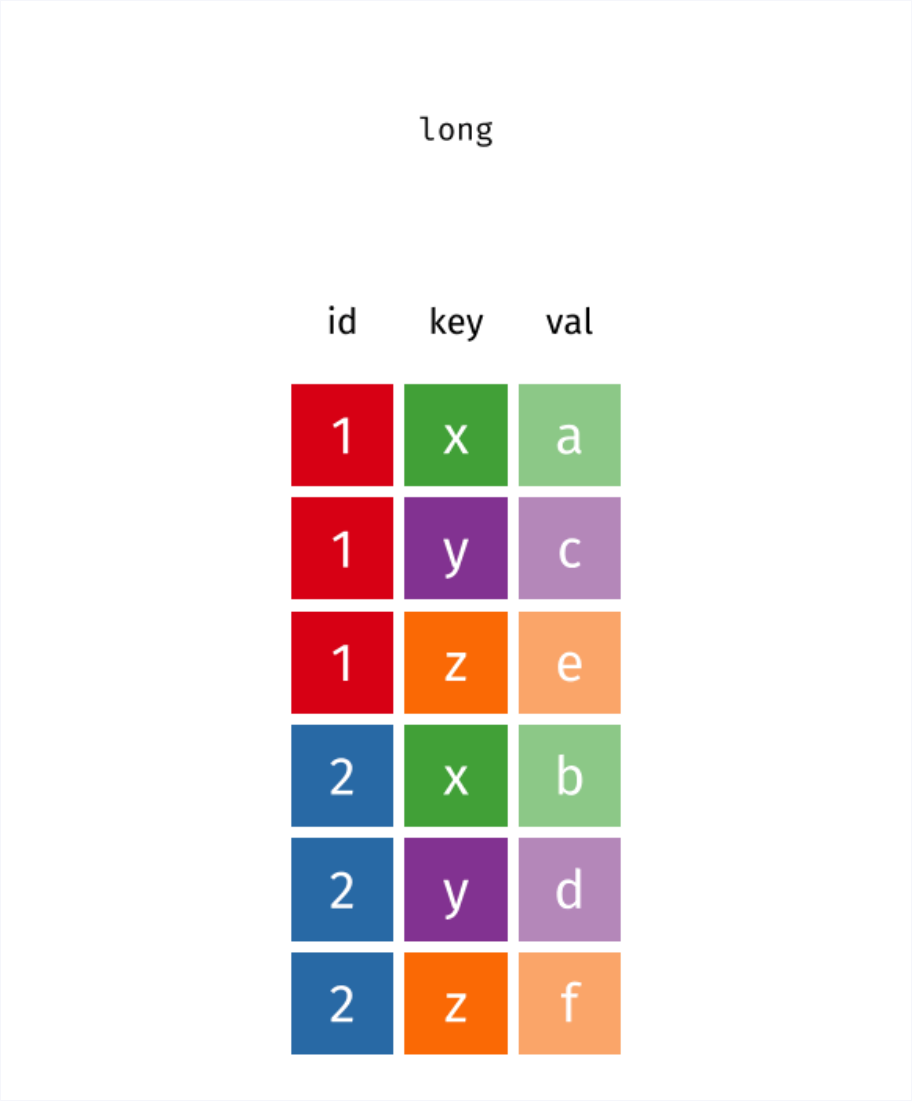
```
classroom
```

```
# A tibble: 4 × 4
  student  exercise1 exercise2 quiz1
  <chr>        <dbl>     <dbl> <dbl>
1 Ulyaa           42        88    95
2 Renee           91        96    82
3 Siddeeqa        21        37    89
4 Valentin        38        95    56
```

```
classroom |>
  pivot_longer(
    cols = 2:4,  # or cols = -student
    names_to = "assessment",
    values_to = "grade"
  )
```

```
# A tibble: 12 × 3
   student  assessment grade
   <chr>    <chr>      <dbl>
 1 Ulyaa    exercise1     42
 2 Ulyaa    exercise2     88
 3 Ulyaa    quiz1         95
 4 Renee    exercise1     91
 5 Renee    exercise2     96
 6 Renee    quiz1         82
 7 Siddeeqa exercise1     21
 8 Siddeeqa exercise2     37
 9 Siddeeqa quiz1         89
10 Valentin exercise1     38
11 Valentin exercise2     95
12 Valentin quiz1         56
```

Pivoting wider




```
long_data |>
  pivot_wider(
    # Names for the
    # spread out columns
    names_from = "key",

    # Values for the
    # spread out columns
    values_from = "value"
  )
```

Pivoting wider

```
classroom_long
```

```
# A tibble: 12 × 3
   student  assessment grade
   <chr>    <chr>      <dbl>
 1 Ulyaa    exercise1     42
 2 Ulyaa    exercise2     88
 3 Ulyaa    quiz1         95
 4 Renee    exercise1     91
 5 Renee    exercise2     96
 6 Renee    quiz1         82
 7 Siddeeqa exercise1     21
 8 Siddeeqa exercise2     37
 9 Siddeeqa quiz1         89
10 Valentin exercise1     38
11 Valentin exercise2     95
12 Valentin quiz1         56
```

```
classroom_long |>
  pivot_wider(
    names_from = "assessment",
    values_from = "grade"
  )
```

```
# A tibble: 4 × 4
  student  exercise1 exercise2 quiz1
  <chr>        <dbl>     <dbl> <dbl>
1 Ulyaa           42        88    95
2 Renee           91        96    82
3 Siddeeqa        21        37    89
4 Valentin        38        95    56
```

Your/our turn #5

Pivoting!