Tables with the *gt package and gtExtras*

Aditya Narayan Rai
Letícia Figueiredo Collado
Lino Hans Julian Zurmüehl

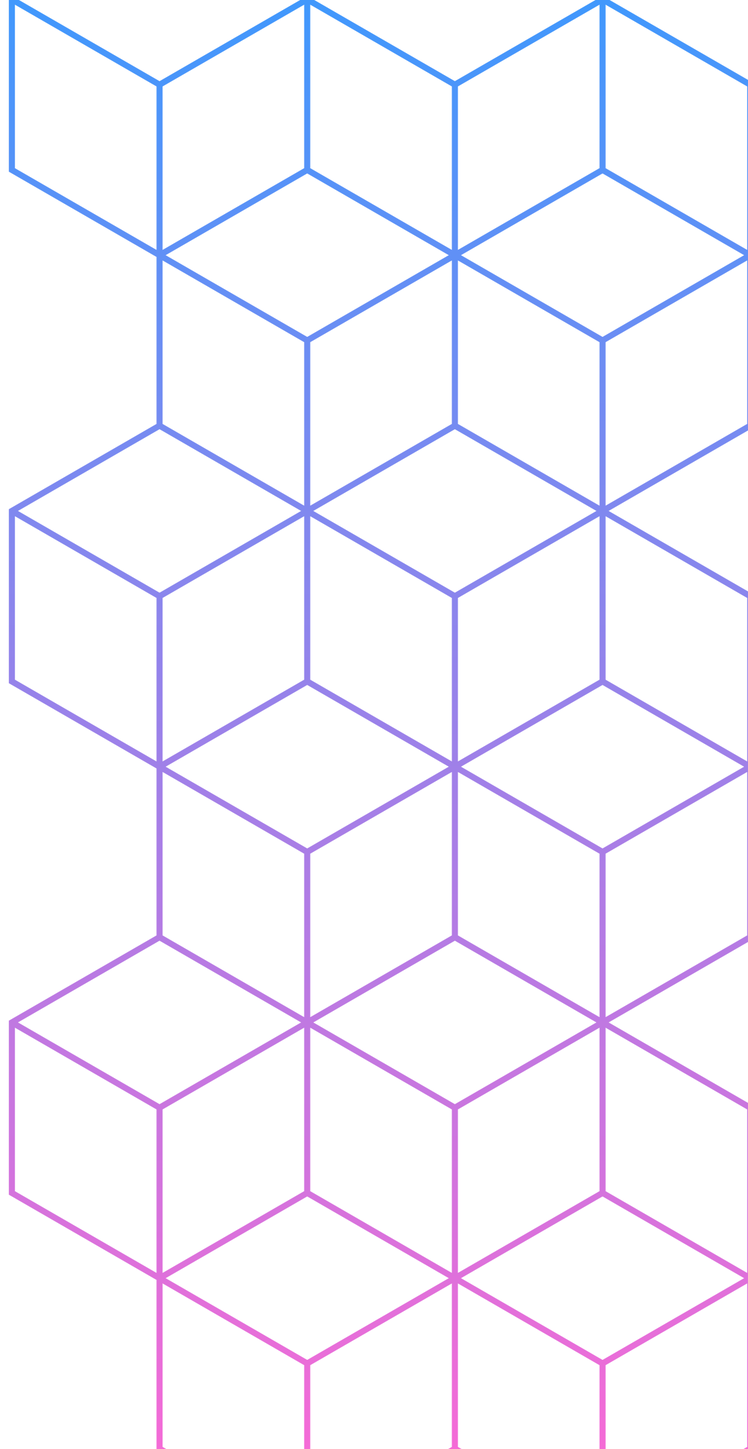
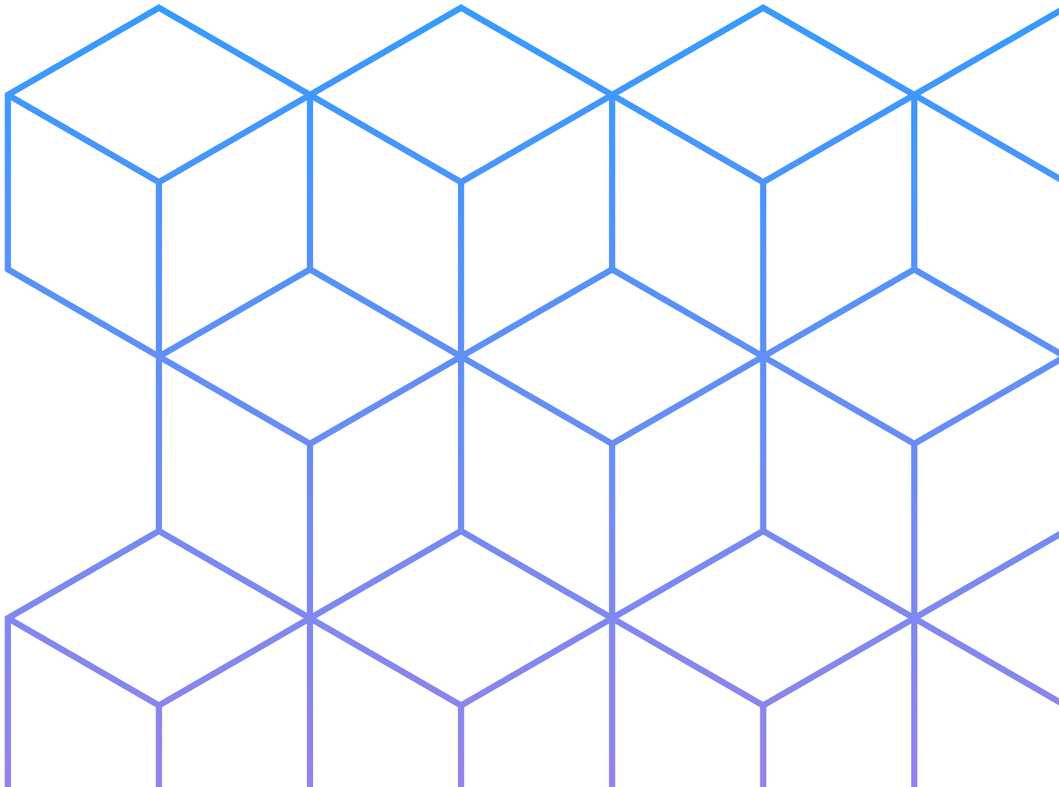


Table of contents



What is the gt package?

A simple way to create nice looking tables for a more visually appealing display of data in articles, presentations, webpages, books, etc.

	day	open	close	volume	difference	change
[1] 1987-10-14	Wed	$314.52	$305.23	207.40M	−$9.29	−2.95%
1987-10-15	Thu	$305.21	$298.08	263.20M	−$7.13	−2.34%
1987-10-16	Fri	$298.08	$282.70	338.50M	−$15.38	−5.16%
1987-10-19	Mon	$282.70	$224.84	604.30M	−$57.86	[2] −20.47%
1987-10-20	Tue	$225.06	$236.83	608.10M	$11.77	5.23%
1987-10-21	Wed	$236.83	$258.38	449.60M	$21.55	9.10%
1987-10-22	Thu	$258.24	$248.25	392.20M	−$9.99	−3.87%
1987-10-23	Fri	$248.29	$248.22	245.60M	−$0.07	−0.03%

(1) Commerce report on trade deficit.

(2) Black Monday market crash, representing the greatest one-day percentage decline in U.S. stock market history.

What is the gt package?

A simple way to create nice looking tables for a more visually appealing display of data in articles, presentations, webpages, books, etc.

	day	open	close	volume	difference	change
1987-10-14	Wed	$314.52	$305.23	207.40M	−$9.29	−2.95%
1987-10-15	Thu	$305.21	$298.08	263.20M	−$7.13	−2.34%
1987-10-16	Fri	$298.08	$282.70	339.60M	−$15.38	−5.16%
1987-10-19	Mon	$282.70	$			
1987-10-20	Tue	$225.06	$			
1987-10-21	Wed	$236.83	$			
1987-10-22	Thu	$258.24	$			
1987-10-23	Fri	$248.29	$			

(1) Commerce report on trade deficit.

(2) Black Monday market crash, representing the greatest one-day percentage decline in U.S. stock market history.

Populations of the Benelux Countries							
	Year						
	1960	1970	1980	1990	2000	2010	2020
Belgium	9,153,489	9,655,549	9,859,242	9,967,379	10,251,250	10,895,586	11,538,604
Netherlands	11,486,631	13,038,526	14,149,800	14,951,510	15,925,513	16,615,394	17,441,500
Luxembourg	313,970	339,171	364,150	381,850	436,300	506,953	630,419

What is the gt package?

A simple way to create nice looking tables for a more visually appealing display of data in articles, presentations, webpages, books, etc.

	day	open	close	volume	differe
[1] 1987-10-14	Wed	$314.52	$305.23	207.40M	−$9
1987-10-15	Thu	$305.21	$298.08	263.20M	−$7
1987-10-16	Fri	$298.08	$282.70	338.50M	−$15
1987-10-19	Mon	$282.70	$224.84	604.30M	−$57
1987-10-20	Tue	$225.06	$236.83	608.10M	$11
1987-10-21	Wed	$236.83	$258.38	449.60M	$21
1987-10-22	Thu	$258.24	$248.25	392.20M	−$9
1987-10-23	Fri	$248.29	$248.22	245.60M	−$0

(1) Commerce report on trade deficit.

(2) Black Monday market crash, representing the greatest one-day percentage de
market history.

	Belgium	9,153,48
	Netherlands	11,486,63
	Luxembourg	313,97

Largest Five in towny

Changes in vital numbers from 2016 to 2021.[a]

	Area, km^2	Density, ppl/km^2	Population
TORONTO	631.1	4,328 → 4,428	2.73M → 2.79M
OTTAWA	2,788.2	335 → 365	934K → 1.02M
MISSISSAUGA	292.7	2,465 → 2,453	722K → 718K
BRAMPTON	265.9	2,233 → 2,469	594K → 656K
HAMILTON	1,118.3	480 → 509	537K → 569K

[a] Data was used from their respective census-year publications.

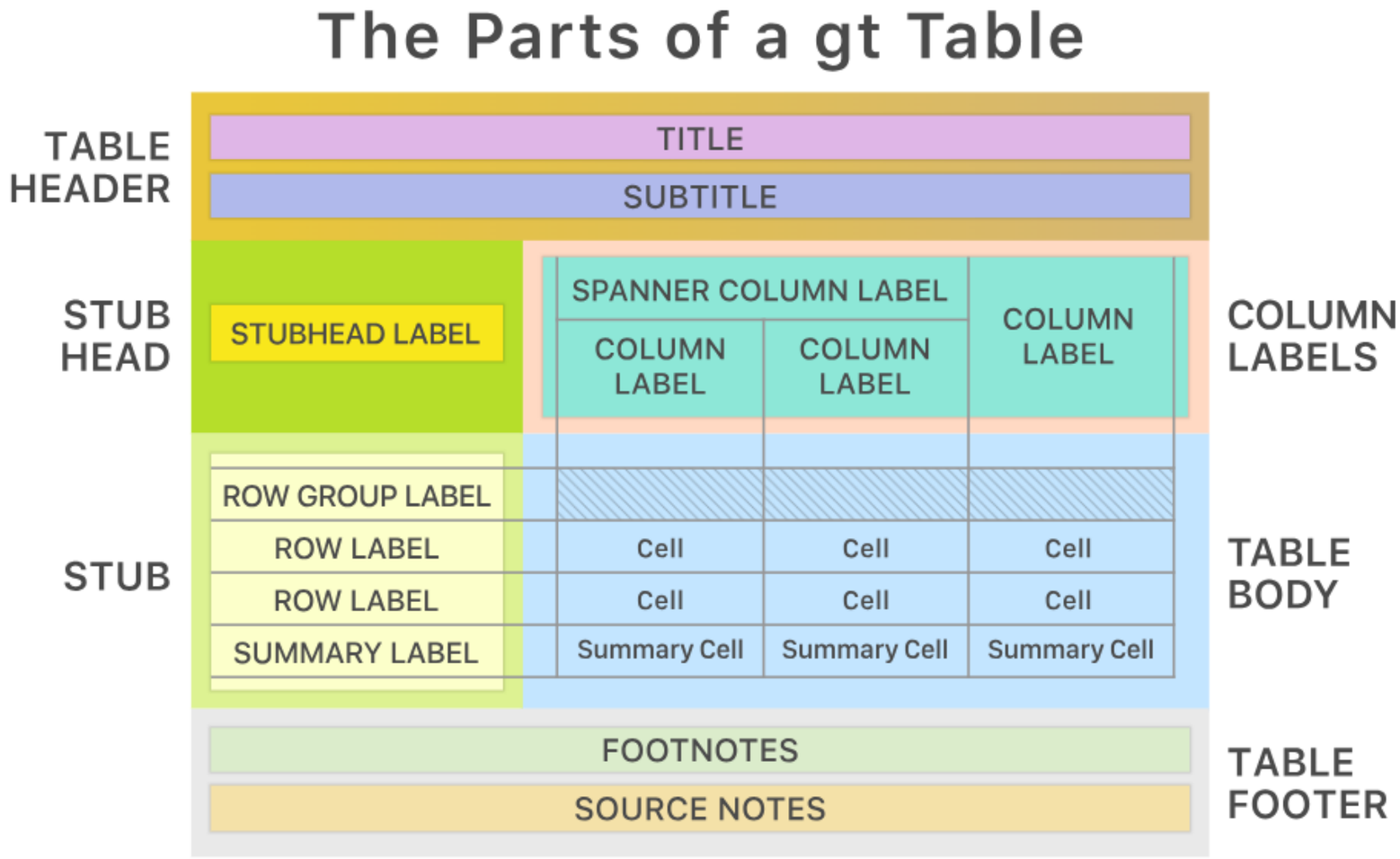
All figures are compiled in the `towny` dataset (in the **gt** package).

What is the gt package?

A simple way to create nice looking tables for a more visually appealing display of data in articles, presentations, webpages, books, etc.

Population Range		Countries
1B–2B	*2*	
300M–500M	*1*	
200M–300M	*4*	
100M–200M	*7*	
50M–100M	*15*	
30M–50M	*20*	
20M–30M	*11*	
10M–20M	*31*	
5M–10M	*32*	
3M–5M	*12*	
2M–3M	*14*	
1M–2M	*10*	

Basic elements



Getting started

The **gt package** provides ten different datasets: **countrypops**, **sza**, **gtcars**, **sp500**, **pizzaplace**, **exibble**, **towny**, **metro**, **rx_adsl**, and **rx_addv**

➜ We can find more information on the built in datasets in the project's website.

Getting started

Let’s take a look at one of them!
The **metro** dataset contains information on all 308 Paris Metro stations (feb/2023).
This is what it looks like:

lines	connect_rer	connect_tramway	connect_transilien	connect_other	passengers	latitude	longitude	location
1	*NA*	*NA*	*NA*	*NA*	2079212	48.87528	2.290000	Paris 16th, Paris 17th
1, 5, 8	*NA*	*NA*	*NA*	*NA*	8069243	48.85308	2.369077	Paris 4th, Paris 11th, Paris 12th
1	*NA*	*NA*	*NA*	*NA*	2106827	48.84528	2.428333	Saint-Mandé, Vincennes
1, 13	*NA*	*NA*	*NA*	*NA*	1909005	48.86750	2.313500	Paris 8th
1, 2, 6	A	*NA*	*NA*	*NA*	4291663	48.87389	2.295000	Paris 8th, Paris 16th, Paris 17th
1	*NA*	*NA*	*NA*	*NA*	3617738	48.84444	2.440000	Paris 12th, Vincennes
1, 4, 7, 11, 14	A, B, D	*NA*	*NA*	*NA*	8350794	48.85835	2.347324	Paris 1st, Paris 4th
1, 8, 12	*NA*	*NA*	*NA*	*NA*	3401219	48.86541	2.321110	Paris 1st
1	*NA*	*NA*	*NA*	*NA*	4708183	48.88833	2.250000	Courbevoie, Puteaux
1, 9	*NA*	*NA*	*NA*	*NA*	6173830	48.86898	2.309890	Paris 8th
1, 14	A, D	*NA*	R	TGV, TGV Lyria, Renfe-SNCF, OUIGO, Frecciarossa	28640475	48.84472	2.373889	Paris 12th
1	*NA*	*NA*	*NA*	*NA*	3842260	48.87194	2.300556	Paris 8th
1, 11	*NA*	*NA*	*NA*	*NA*	7251729	48.85749	2.351525	Paris 4th
1	A	T2	L, U	*NA*	9256802	48.89185	2.238539	Puteaux
1	*NA*	*NA*	*NA*	*NA*	3954920	48.88083	2.272222	Neuilly-sur-Seine
1	*NA*	*NA*	*NA*	*NA*	1869612	48.86108	2.340283	Paris 1st
1, 2, 6, 9	A	*NA*	*NA*	*NA*	6050797	48.84833	2.395833	Paris 11th, Paris 12th
1, 7	*NA*	*NA*	*NA*	*NA*	4822599	48.86264	2.336578	Paris 1st

Getting started

The *passengers* variable includes the total number of entries in each station for the year of 2021. What are the 10 busiest stations in Paris?

```
library(tidyverse)
library(gt)

busiest_stations <- metro %>%
  select (name, lines, passengers, location) %>%
  arrange(desc(passengers)) %>%
  slice(1:10)

gt(busiest_stations)
```

name	lines	passengers	location
Gare du Nord	4, 5	34503097	Paris 10th
Saint-Lazare	3, 12, 13, 14	33128384	Paris 8th, Paris 9th
Gare de Lyon	1, 14	28640475	Paris 12th
Montparnasse—Bienvenüe	4, 6, 12, 13	20407224	Paris 6th, Paris 14th, Paris 15th
Gare de l'Est	4, 5, 7	15538471	Paris 10th
Bibliothèque François Mitterrand	14	11104474	Paris 13th
République	3, 5, 8, 9, 11	11079708	Paris 3rd, Paris 10th, Paris 11th
Les Halles	4	10623876	Paris 1st
La Défense	1	9256802	Puteaux
Châtelet	1, 4, 7, 11, 14	8350794	Paris 1st, Paris 4th

Keeps getting better

No matter how simple the information, we can still work on the layout, and even on how the data itself is presented.

```
busyst_table <- gt(busyst) %>%
  cols_label(name = "Station name",
             lines = "Lines",
             passengers = "Passengers",
             location = "Location") %>%
  tab_header(title = "Busiest Metro stations in Paris",
             subtitle = "Data from the year 2021")
```

Busiest Metro stations in Paris

Data from the year 2021

Station name	Lines	Passengers	Location
Gare du Nord	4, 5	34503097	Paris 10th
Saint-Lazare	3, 12, 13, 14	33128384	Paris 8th, Paris 9th
Gare de Lyon	1, 14	28640475	Paris 12th
Montparnasse—Bienvenüe	4, 6, 12, 13	20407224	Paris 6th, Paris 14th, Paris 15th
Gare de l'Est	4, 5, 7	15538471	Paris 10th
Bibliothèque François Mitterrand	14	11104474	Paris 13th
République	3, 5, 8, 9, 11	11079708	Paris 3rd, Paris 10th, Paris 11th
Les Halles	4	10623876	Paris 1st
La Défense	1	9256802	Puteaux
Châtelet	1, 4, 7, 11, 14	8350794	Paris 1st, Paris 4th

Keeps getting better

And then we can make other modifications.

```
busyst_table %>%
  tab_options(table.width = px(780),
              data_row.padding = px(6),
              heading.align = 'left',
              column_labels.background.color = 'lightgrey',
              heading.title.font.size = px(24),
              heading.subtitle.font.size = px(14),
              table_body.hlines.width = px(0)) %>%

  cols_width(name ~ px(170),
             location ~ px(170),
             everything() ~ px(90)) %>%

  cols_align(align = "left",
             columns = everything()) %>%

  tab_style(style = cell_text(color = "hotpink",
                              weight = 'bold'),
            locations = cells_title(groups = 'title')) %>%

  tab_style(style = cell_text(style = 'italic'),
            locations = cells_title(groups = 'subtitle')) %>%

  tab_style(style = cell_text(weight = 'bold'),
            locations = cells_column_labels()) %>%

  fmt_number(columns = passengers,
             decimals = 0,
             use_seps = TRUE,
             sep_mark = ",")
```

Overall layout:

- Table width
- Space between rows
- Font sizes
- Lines between rows

Columns width and alignment

Title

Subtitle

Column names

Number separators

Keeps getting better

```
busyst_table %>%
  tab_options(table.width = px(780),
              data_row.padding = px(6),
              heading.align = 'left',
              column_labels.background.color = 'lightgrey',
              heading.title.font.size = px(24),
              heading.subtitle.font.size = px(14),
              table_body.hlines.width = px(0)) %>%

  cols_width(name ~ px(170),
             location ~ px(170),
             everything() ~ px(90)) %>%

  cols_align(align = "left",
             columns = everything()) %>%

  tab_style(style = cell_text(color = "hotpink",
                              weight = 'bold'),
            locations = cells_title(groups = 'title')) %>%

  tab_style(style = cell_text(style = 'italic'),
            locations = cells_title(groups = 'subtitle')) %>%

  tab_style(style = cell_text(weight = 'bold'),
            locations = cells_column_labels()) %>%

  fmt_number(columns = passengers,
             decimals = 0,
             use_seps = TRUE,
             sep_mark = ",")
```

What does it look like now?

Keeps getting better

Busiest Metro stations in Paris

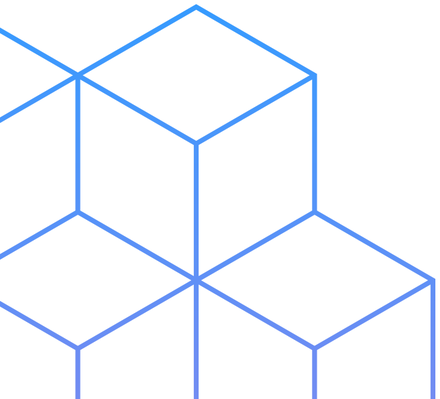
Data from the year 2021

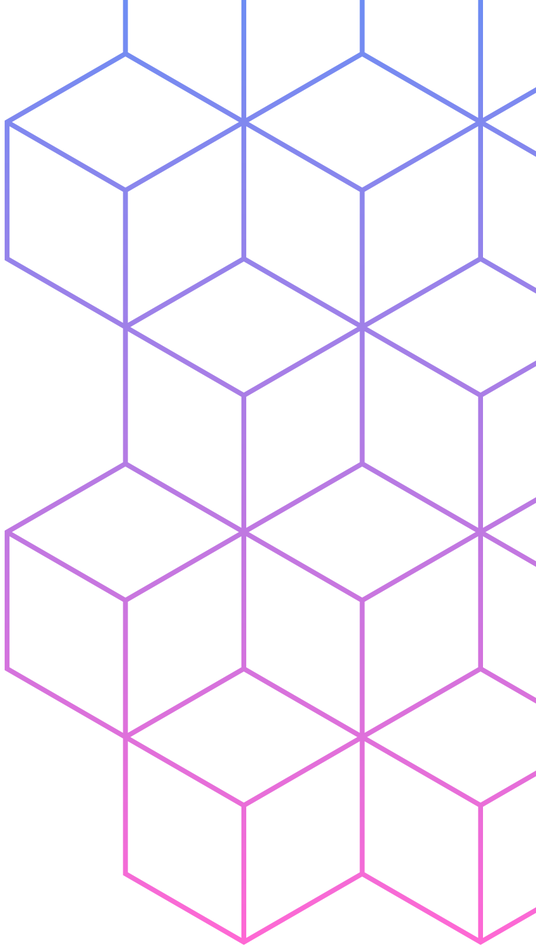
Station name	Lines	Passengers	Location
Gare du Nord	4, 5	34,503,097	Paris 10th
Saint-Lazare	3, 12, 13, 14	33,128,384	Paris 8th, Paris 9th
Gare de Lyon	1, 14	28,640,475	Paris 12th
Montparnasse—Bienvenüe	4, 6, 12, 13	20,407,224	Paris 6th, Paris 14th, Paris 15th
Gare de l'Est	4, 5, 7	15,538,471	Paris 10th
Bibliothèque François Mitterrand	14	11,104,474	Paris 13th
République	3, 5, 8, 9, 11	11,079,708	Paris 3rd, Paris 10th, Paris 11th
Les Halles	4	10,623,876	Paris 1st
La Défense	1	9,256,802	Puteaux
Châtelet	1, 4, 7, 11, 14	8,350,794	Paris 1st, Paris 4th

Keeps getting better

And we can always make more and more modifications, like

- Changing the font
- Adding country flags
- Adding currency symbols
- Adjusting the color palette
- Adding ggplots, e.g.: a scatter plot, a bar chart, etc
- Adding images from local repository or from the web
- Creating summary lines by group of rows, with a mean, sum, standard deviation, etc
- Adding or removing borders and background color for lines, columns and specific cells





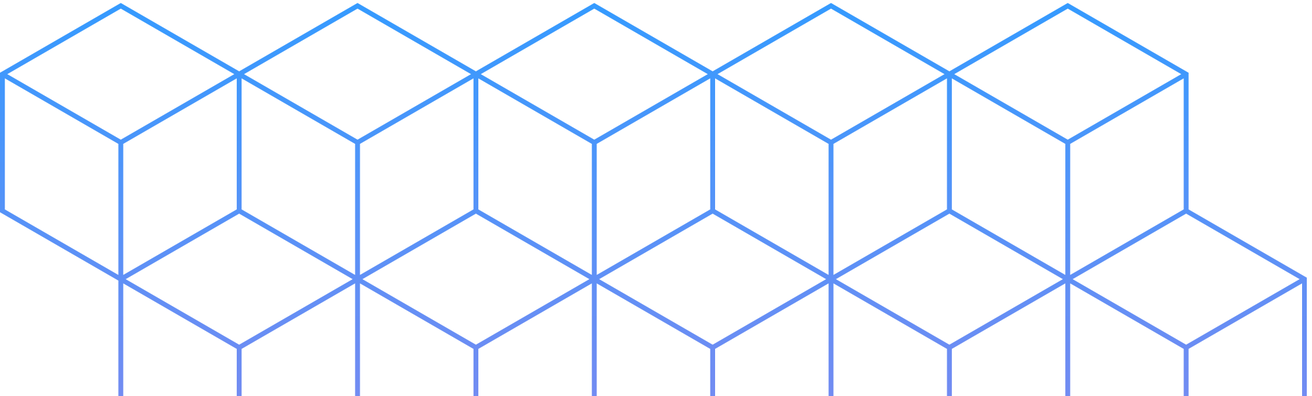
And more!

What about gtExtras?

gtExtras provides some features that are, for now, not available in the original **gt package**, such as:

- Modifying image characteristics
- Better formating numbers
- Further improving the options when adding plots
- Adding hyperlinks
- Extracting row index for existing gt tables
- Saving tables as *.png*
- *Adding badges based on values*

Let’s give it a try!



What about gtExtras?

Busiest Metro stations in Paris

Data from the year 2021

Station name	N. of lines	Passengers	Location
Gare du Nord	2	34,503,097	Paris 10th
Saint-Lazare	4	33,128,384	Paris 8th, Paris 9th
Gare de Lyon	2	28,640,475	Paris 12th
Montparnasse—Bienvenüe	4	20,407,224	Paris 6th, Paris 14th, Paris 15th
Gare de l'Est	3	15,538,471	Paris 10th
Bibliothèque François Mitterrand	1	11,104,474	Paris 13th
République	5	11,079,708	Paris 3rd, Paris 10th, Paris 11th
Les Halles	1	10,623,876	Paris 1st
La Défense	1	9,256,802	Puteaux
Châtelet	5	8,350,794	Paris 1st, Paris 4th

Changes:

- Sum number of lines in each station
- Highlight the stations with higher number of lines
- Indicate station outside Paris arrondissements

What about gtExtras?

```
busyst <- busyst %>%
  mutate(outside = ifelse(!str_detect(location, "Paris"),
                          "arrow-left", NA),
         lines2 = str_count(lines, ",")+1) %>%
  select(name, lines2, passengers, location, outside)
```

dplyr:

- Locate stations without Paris in their names
- Count number of lines in each station
- Select relevant columns

```
busyst_table %>%
  tab_options(...

              %>%

gt_badge(lines2,
         palette = c("5" = "hotpink",
                     "4" = "hotpink",
                     "3" = "pink",
                     "2" = "pink",
                     "1" = "white"),
         alpha = 0.9) %>%

gt_fa_column(outside)
```

gt & **gtExtras**

- Add badge on stations with the highest number of lines
- Indicate station outside Paris

What else?

gtExtras also provides us with templates of themes related to famous layouts, like The New York Times, ESPN and Excel, among others.

```
gt(nyt) %>%
  cols_label(name = "Station name",
             lines2 = "N. of lines",
             passengers = "Passengers",
             location = "Location") %>%
  tab_header(title = "Busiest Metro stations in Paris",
             subtitle = "Data from the year 2021,
             styled like the NY Times") %>%
  gt_theme_nytimes()
```

Busiest Metro stations in Paris

Data from the year 2021, styled like the NY Times

STATION NAME	N. OF LINES	PASSENGERS	LOCATION
Gare du Nord	2	34503097	Paris 10th
Saint-Lazare	4	33128384	Paris 8th, Paris 9th
Gare de Lyon	2	28640475	Paris 12th
Montparnasse—Bienvenüe	4	20407224	Paris 6th, Paris 14th, Paris 15th
Gare de l'Est	3	15538471	Paris 10th
Bibliothèque François Mitterrand	1	11104474	Paris 13th
République	5	11079708	Paris 3rd, Paris 10th, Paris 11th
Les Halles	1	10623876	Paris 1st
La Défense	1	9256802	Puteaux
Châtelet	5	8350794	Paris 1st, Paris 4th

Learn more!

The **gt project** has a website of their own, with:

- A full tutorial and documentation for every function available
- Case studies with datasets provided by the package
- News on version releases

You can also check out the project's GitHub page or their Twitter account for more information and development news.

The Discussions tab on the project's GitHub and the **gt package** Discord Server are great places to check out people's work, share your own or ask questions if you're feeling lost and need some help!

There's loads of other online tutorials using both **gt** and **gtExtras**, that people were kind enough to produce and share, like here, here and here.

Learn more!

The **gtExtras** also has a page of their own, with:

- Documentation for all the functions available
- A tutorial for plotting with gtExtras

The theme templates provided by the **gtExtras** package can be found here, and an example can be found here.

Last, but not least, other packages also work with **gt** or add features to it, such as:

- gtsummary, for statistical summaries of data frames and regression models
- pointblank, for data validation
- tfrmt, for dealing with metadata
- gto, to allow for outputs accepted in the Microsoft Office universe

Thank you!

Let us know if you have
any questions =]