Tables walkthrough (Winter2025)

Note: One important difference between R Markdown documents and Quarto documents is that in Quarto chunk options are typically included in special comments at the top of code chunks rather than within the line that begins the chunk. For all potions, check out the Quarto documentation.

Tables: Basic "Console Style"

We can render tables in R documents in several different forms. To start, you have your basic "console style." This is more or less what you'd see in the console you just ran the table (or called the object) in an R script or directly in the console. See Table 1.

Note: For cross-reference of tables, you need to include a label in the chunk options that starts with `tbl`. Then you can cite in-text using the syntax "@" + "tbl" + "label name". Adding a label option will enable the numbering of tables, too.

Tables: knitr and kable()

The `knitr` package contains the `kable()` function. Passing a dataframe into this function will produce a formatted table that's already pretty nice looking without needing additional modification. You don't need to do anything more than pipe the df to `kable()` to get this formatted table, but we can add optional arguments like `caption`.

```
mydata %>%
  group_by(cyl) %>%
  summarize(
    mean.mpg = mean(mpg),
    sd.mpg = sd(mpg),
    n.models = n()
  ) %>%
  knitr::kable(caption = "mpg stats by number of cylinders")
```

Table 1

```
mydata %>%
  group_by(cyl) %>%
  summarize(
    mean.mpg = mean(mpg),
    sd.mpg = sd(mpg),
    n.models = n()
  )
```

```
# A tibble: 3 x 4
    cyl mean.mpg sd.mpg n.models
  <dbl>    <dbl>  <dbl>    <int>
1     4     26.7   4.51       11
2     6     19.7   1.45        7
3     8     15.1   2.56       14
```

Table 2: mpg stats by number of cylinders

cyl	mean.mpg	sd.mpg	n.models
4	26.66364	4.509828	11
6	19.74286	1.453567	7
8	15.10000	2.560048	14

When you run the code chunk in R Studio, it is clearly different from the console version, but it's not exactly beautiful. However, that output that appears beneath your code chunk isn't the same as what you'll see when you actually render it. Some wrapper functions rely on LaTeX formatting that can't render in R Studio. When you run the chunk in your R notebook, you'll just see a blank box. The APA formatted table will render in your Word or PDF document. Note that not all wrapper functions will have this "blank box problem." Other wrappers will not only not show up in R Studio, but will only render in either Word *or* PDF format.

Back to `kable`, there is nearly unlimited customization you can employ, especially if you use the `kableExtra` package (which is exactly what it sounds like). Here are a handful of modifications I think you're most likely to want to know about:

1. Add a caption
2. Change column names
3. Specify column alignment
4. Format columns

5. Style table size and position
6. Make row- and column-specific tweaks
7. Group rows, columns, and cells
8. Add table (foot)notes

Many of the examples below are taken or adapted from the R Markdown Cookbook.

These tables modify column names and alignment identically but with slightly different syntax and add captions to describe each:

```
kable(iris, # iris is a built-in sample data in package knitr
      col.names = c("Sepal Length", "Sepal Width",  "Petal Length", "Petal Width",  "Species"
      align = c("l", "r", "l", "r", "c"),
      caption = "Change each column name and text alignments, each with a list of strings.")
```

Table 3: Change each column name and text alignments, each with a list of strings.

Sepal Length	Sepal Width	Petal Length	Petal Width	Species
5.1	3.5	1.4	0.2	setosa
4.9	3.0	1.4	0.2	setosa
4.7	3.2	1.3	0.2	setosa
4.6	3.1	1.5	0.2	setosa
5.0	3.6	1.4	0.2	setosa
5.4	3.9	1.7	0.4	setosa
4.6	3.4	1.4	0.3	setosa
5.0	3.4	1.5	0.2	setosa
4.4	2.9	1.4	0.2	setosa
4.9	3.1	1.5	0.1	setosa
5.4	3.7	1.5	0.2	setosa
4.8	3.4	1.6	0.2	setosa
4.8	3.0	1.4	0.1	setosa
4.3	3.0	1.1	0.1	setosa
5.8	4.0	1.2	0.2	setosa
5.7	4.4	1.5	0.4	setosa
5.4	3.9	1.3	0.4	setosa
5.1	3.5	1.4	0.3	setosa
5.7	3.8	1.7	0.3	setosa
5.1	3.8	1.5	0.3	setosa
5.4	3.4	1.7	0.2	setosa
5.1	3.7	1.5	0.4	setosa
4.6	3.6	1.0	0.2	setosa
5.1	3.3	1.7	0.5	setosa

Table 3: Change each column name and text alignments, each with a list of strings.

Sepal Length	Sepal Width	Petal Length	Petal Width	Species
4.8	3.4	1.9	0.2	setosa
5.0	3.0	1.6	0.2	setosa
5.0	3.4	1.6	0.4	setosa
5.2	3.5	1.5	0.2	setosa
5.2	3.4	1.4	0.2	setosa
4.7	3.2	1.6	0.2	setosa
4.8	3.1	1.6	0.2	setosa
5.4	3.4	1.5	0.4	setosa
5.2	4.1	1.5	0.1	setosa
5.5	4.2	1.4	0.2	setosa
4.9	3.1	1.5	0.2	setosa
5.0	3.2	1.2	0.2	setosa
5.5	3.5	1.3	0.2	setosa
4.9	3.6	1.4	0.1	setosa
4.4	3.0	1.3	0.2	setosa
5.1	3.4	1.5	0.2	setosa
5.0	3.5	1.3	0.3	setosa
4.5	2.3	1.3	0.3	setosa
4.4	3.2	1.3	0.2	setosa
5.0	3.5	1.6	0.6	setosa
5.1	3.8	1.9	0.4	setosa
4.8	3.0	1.4	0.3	setosa
5.1	3.8	1.6	0.2	setosa
4.6	3.2	1.4	0.2	setosa
5.3	3.7	1.5	0.2	setosa
5.0	3.3	1.4	0.2	setosa
7.0	3.2	4.7	1.4	versicolor
6.4	3.2	4.5	1.5	versicolor
6.9	3.1	4.9	1.5	versicolor
5.5	2.3	4.0	1.3	versicolor
6.5	2.8	4.6	1.5	versicolor
5.7	2.8	4.5	1.3	versicolor
6.3	3.3	4.7	1.6	versicolor
4.9	2.4	3.3	1.0	versicolor
6.6	2.9	4.6	1.3	versicolor
5.2	2.7	3.9	1.4	versicolor
5.0	2.0	3.5	1.0	versicolor
5.9	3.0	4.2	1.5	versicolor
6.0	2.2	4.0	1.0	versicolor

Table 3: Change each column name and text alignments, each with a list of strings.

Sepal Length	Sepal Width	Petal Length	Petal Width	Species
6.1	2.9	4.7	1.4	versicolor
5.6	2.9	3.6	1.3	versicolor
6.7	3.1	4.4	1.4	versicolor
5.6	3.0	4.5	1.5	versicolor
5.8	2.7	4.1	1.0	versicolor
6.2	2.2	4.5	1.5	versicolor
5.6	2.5	3.9	1.1	versicolor
5.9	3.2	4.8	1.8	versicolor
6.1	2.8	4.0	1.3	versicolor
6.3	2.5	4.9	1.5	versicolor
6.1	2.8	4.7	1.2	versicolor
6.4	2.9	4.3	1.3	versicolor
6.6	3.0	4.4	1.4	versicolor
6.8	2.8	4.8	1.4	versicolor
6.7	3.0	5.0	1.7	versicolor
6.0	2.9	4.5	1.5	versicolor
5.7	2.6	3.5	1.0	versicolor
5.5	2.4	3.8	1.1	versicolor
5.5	2.4	3.7	1.0	versicolor
5.8	2.7	3.9	1.2	versicolor
6.0	2.7	5.1	1.6	versicolor
5.4	3.0	4.5	1.5	versicolor
6.0	3.4	4.5	1.6	versicolor
6.7	3.1	4.7	1.5	versicolor
6.3	2.3	4.4	1.3	versicolor
5.6	3.0	4.1	1.3	versicolor
5.5	2.5	4.0	1.3	versicolor
5.5	2.6	4.4	1.2	versicolor
6.1	3.0	4.6	1.4	versicolor
5.8	2.6	4.0	1.2	versicolor
5.0	2.3	3.3	1.0	versicolor
5.6	2.7	4.2	1.3	versicolor
5.7	3.0	4.2	1.2	versicolor
5.7	2.9	4.2	1.3	versicolor
6.2	2.9	4.3	1.3	versicolor
5.1	2.5	3.0	1.1	versicolor
5.7	2.8	4.1	1.3	versicolor
6.3	3.3	6.0	2.5	virginica
5.8	2.7	5.1	1.9	virginica

Table 3: Change each column name and text alignments, each with a list of strings.

Sepal Length	Sepal Width	Petal Length	Petal Width	Species
7.1	3.0	5.9	2.1	virginica
6.3	2.9	5.6	1.8	virginica
6.5	3.0	5.8	2.2	virginica
7.6	3.0	6.6	2.1	virginica
4.9	2.5	4.5	1.7	virginica
7.3	2.9	6.3	1.8	virginica
6.7	2.5	5.8	1.8	virginica
7.2	3.6	6.1	2.5	virginica
6.5	3.2	5.1	2.0	virginica
6.4	2.7	5.3	1.9	virginica
6.8	3.0	5.5	2.1	virginica
5.7	2.5	5.0	2.0	virginica
5.8	2.8	5.1	2.4	virginica
6.4	3.2	5.3	2.3	virginica
6.5	3.0	5.5	1.8	virginica
7.7	3.8	6.7	2.2	virginica
7.7	2.6	6.9	2.3	virginica
6.0	2.2	5.0	1.5	virginica
6.9	3.2	5.7	2.3	virginica
5.6	2.8	4.9	2.0	virginica
7.7	2.8	6.7	2.0	virginica
6.3	2.7	4.9	1.8	virginica
6.7	3.3	5.7	2.1	virginica
7.2	3.2	6.0	1.8	virginica
6.2	2.8	4.8	1.8	virginica
6.1	3.0	4.9	1.8	virginica
6.4	2.8	5.6	2.1	virginica
7.2	3.0	5.8	1.6	virginica
7.4	2.8	6.1	1.9	virginica
7.9	3.8	6.4	2.0	virginica
6.4	2.8	5.6	2.2	virginica
6.3	2.8	5.1	1.5	virginica
6.1	2.6	5.6	1.4	virginica
7.7	3.0	6.1	2.3	virginica
6.3	3.4	5.6	2.4	virginica
6.4	3.1	5.5	1.8	virginica
6.0	3.0	4.8	1.8	virginica
6.9	3.1	5.4	2.1	virginica
6.7	3.1	5.6	2.4	virginica

Table 3: Change each column name and text alignments, each with a list of strings.

Sepal Length	Sepal Width	Petal Length	Petal Width	Species
6.9	3.1	5.1	2.3	virginica
5.8	2.7	5.1	1.9	virginica
6.8	3.2	5.9	2.3	virginica
6.7	3.3	5.7	2.5	virginica
6.7	3.0	5.2	2.3	virginica
6.3	2.5	5.0	1.9	virginica
6.5	3.0	5.2	2.0	virginica
6.2	3.4	5.4	2.3	virginica
5.9	3.0	5.1	1.8	virginica

```
kable(iris,
      col.names = gsub("[.]", " ", names(iris)),
      align = "lrlrc",
      caption = "Use 'gsub' function to replace periods with spaces and change text alignment
```

Table 4: Use 'gsub' function to replace periods with spaces and change text alignments with a single 'shortcut' string.

Sepal Length	Sepal Width	Petal Length	Petal Width	Species
5.1	3.5	1.4	0.2	setosa
4.9	3.0	1.4	0.2	setosa
4.7	3.2	1.3	0.2	setosa
4.6	3.1	1.5	0.2	setosa
5.0	3.6	1.4	0.2	setosa
5.4	3.9	1.7	0.4	setosa
4.6	3.4	1.4	0.3	setosa
5.0	3.4	1.5	0.2	setosa
4.4	2.9	1.4	0.2	setosa
4.9	3.1	1.5	0.1	setosa
5.4	3.7	1.5	0.2	setosa
4.8	3.4	1.6	0.2	setosa
4.8	3.0	1.4	0.1	setosa
4.3	3.0	1.1	0.1	setosa
5.8	4.0	1.2	0.2	setosa
5.7	4.4	1.5	0.4	setosa
5.4	3.9	1.3	0.4	setosa
5.1	3.5	1.4	0.3	setosa

Table 4: Use 'gsub' function to replace periods with spaces and change text alignments with a single 'shortcut' string.

Sepal Length	Sepal Width	Petal Length	Petal Width	Species
5.7	3.8	1.7	0.3	setosa
5.1	3.8	1.5	0.3	setosa
5.4	3.4	1.7	0.2	setosa
5.1	3.7	1.5	0.4	setosa
4.6	3.6	1.0	0.2	setosa
5.1	3.3	1.7	0.5	setosa
4.8	3.4	1.9	0.2	setosa
5.0	3.0	1.6	0.2	setosa
5.0	3.4	1.6	0.4	setosa
5.2	3.5	1.5	0.2	setosa
5.2	3.4	1.4	0.2	setosa
4.7	3.2	1.6	0.2	setosa
4.8	3.1	1.6	0.2	setosa
5.4	3.4	1.5	0.4	setosa
5.2	4.1	1.5	0.1	setosa
5.5	4.2	1.4	0.2	setosa
4.9	3.1	1.5	0.2	setosa
5.0	3.2	1.2	0.2	setosa
5.5	3.5	1.3	0.2	setosa
4.9	3.6	1.4	0.1	setosa
4.4	3.0	1.3	0.2	setosa
5.1	3.4	1.5	0.2	setosa
5.0	3.5	1.3	0.3	setosa
4.5	2.3	1.3	0.3	setosa
4.4	3.2	1.3	0.2	setosa
5.0	3.5	1.6	0.6	setosa
5.1	3.8	1.9	0.4	setosa
4.8	3.0	1.4	0.3	setosa
5.1	3.8	1.6	0.2	setosa
4.6	3.2	1.4	0.2	setosa
5.3	3.7	1.5	0.2	setosa
5.0	3.3	1.4	0.2	setosa
7.0	3.2	4.7	1.4	versicolor
6.4	3.2	4.5	1.5	versicolor
6.9	3.1	4.9	1.5	versicolor
5.5	2.3	4.0	1.3	versicolor
6.5	2.8	4.6	1.5	versicolor
5.7	2.8	4.5	1.3	versicolor

Table 4: Use 'gsub' function to replace periods with spaces and change text alignments with a single 'shortcut' string.

Sepal Length	Sepal Width	Petal Length	Petal Width	Species
6.3	3.3	4.7	1.6	versicolor
4.9	2.4	3.3	1.0	versicolor
6.6	2.9	4.6	1.3	versicolor
5.2	2.7	3.9	1.4	versicolor
5.0	2.0	3.5	1.0	versicolor
5.9	3.0	4.2	1.5	versicolor
6.0	2.2	4.0	1.0	versicolor
6.1	2.9	4.7	1.4	versicolor
5.6	2.9	3.6	1.3	versicolor
6.7	3.1	4.4	1.4	versicolor
5.6	3.0	4.5	1.5	versicolor
5.8	2.7	4.1	1.0	versicolor
6.2	2.2	4.5	1.5	versicolor
5.6	2.5	3.9	1.1	versicolor
5.9	3.2	4.8	1.8	versicolor
6.1	2.8	4.0	1.3	versicolor
6.3	2.5	4.9	1.5	versicolor
6.1	2.8	4.7	1.2	versicolor
6.4	2.9	4.3	1.3	versicolor
6.6	3.0	4.4	1.4	versicolor
6.8	2.8	4.8	1.4	versicolor
6.7	3.0	5.0	1.7	versicolor
6.0	2.9	4.5	1.5	versicolor
5.7	2.6	3.5	1.0	versicolor
5.5	2.4	3.8	1.1	versicolor
5.5	2.4	3.7	1.0	versicolor
5.8	2.7	3.9	1.2	versicolor
6.0	2.7	5.1	1.6	versicolor
5.4	3.0	4.5	1.5	versicolor
6.0	3.4	4.5	1.6	versicolor
6.7	3.1	4.7	1.5	versicolor
6.3	2.3	4.4	1.3	versicolor
5.6	3.0	4.1	1.3	versicolor
5.5	2.5	4.0	1.3	versicolor
5.5	2.6	4.4	1.2	versicolor
6.1	3.0	4.6	1.4	versicolor
5.8	2.6	4.0	1.2	versicolor
5.0	2.3	3.3	1.0	versicolor

Table 4: Use 'gsub' function to replace periods with spaces and change text alignments with a single 'shortcut' string.

Sepal Length	Sepal Width	Petal Length	Petal Width	Species
5.6	2.7	4.2	1.3	versicolor
5.7	3.0	4.2	1.2	versicolor
5.7	2.9	4.2	1.3	versicolor
6.2	2.9	4.3	1.3	versicolor
5.1	2.5	3.0	1.1	versicolor
5.7	2.8	4.1	1.3	versicolor
6.3	3.3	6.0	2.5	virginica
5.8	2.7	5.1	1.9	virginica
7.1	3.0	5.9	2.1	virginica
6.3	2.9	5.6	1.8	virginica
6.5	3.0	5.8	2.2	virginica
7.6	3.0	6.6	2.1	virginica
4.9	2.5	4.5	1.7	virginica
7.3	2.9	6.3	1.8	virginica
6.7	2.5	5.8	1.8	virginica
7.2	3.6	6.1	2.5	virginica
6.5	3.2	5.1	2.0	virginica
6.4	2.7	5.3	1.9	virginica
6.8	3.0	5.5	2.1	virginica
5.7	2.5	5.0	2.0	virginica
5.8	2.8	5.1	2.4	virginica
6.4	3.2	5.3	2.3	virginica
6.5	3.0	5.5	1.8	virginica
7.7	3.8	6.7	2.2	virginica
7.7	2.6	6.9	2.3	virginica
6.0	2.2	5.0	1.5	virginica
6.9	3.2	5.7	2.3	virginica
5.6	2.8	4.9	2.0	virginica
7.7	2.8	6.7	2.0	virginica
6.3	2.7	4.9	1.8	virginica
6.7	3.3	5.7	2.1	virginica
7.2	3.2	6.0	1.8	virginica
6.2	2.8	4.8	1.8	virginica
6.1	3.0	4.9	1.8	virginica
6.4	2.8	5.6	2.1	virginica
7.2	3.0	5.8	1.6	virginica
7.4	2.8	6.1	1.9	virginica
7.9	3.8	6.4	2.0	virginica

Table 4: Use 'gsub' function to replace periods with spaces and change text alignments with a single 'shortcut' string.

Sepal Length	Sepal Width	Petal Length	Petal Width	Species
6.4	2.8	5.6	2.2	virginica
6.3	2.8	5.1	1.5	virginica
6.1	2.6	5.6	1.4	virginica
7.7	3.0	6.1	2.3	virginica
6.3	3.4	5.6	2.4	virginica
6.4	3.1	5.5	1.8	virginica
6.0	3.0	4.8	1.8	virginica
6.9	3.1	5.4	2.1	virginica
6.7	3.1	5.6	2.4	virginica
6.9	3.1	5.1	2.3	virginica
5.8	2.7	5.1	1.9	virginica
6.8	3.2	5.9	2.3	virginica
6.7	3.3	5.7	2.5	virginica
6.7	3.0	5.2	2.3	virginica
6.3	2.5	5.0	1.9	virginica
6.5	3.0	5.2	2.0	virginica
6.2	3.4	5.4	2.3	virginica
5.9	3.0	5.1	1.8	virginica

These tables specify format of numeric columns:

```
d <- cbind(
  X1 = runif(3),
  X2 = 10^c(3, 5, 7),
  X3 = rnorm(3, 0, 1000))

kable(d, digits = 4,
      caption = "All numeric data in all columns display at most 4 decimal places")
```

Table 5: All numeric data in all columns display at most 4 decimal places

X1	X2	X3
0.9148	1e+03	955.9356
0.9371	1e+05	47.8847
0.2861	1e+07	-1104.5994

```
kable(d, digits = c(5, 0, 2),
      caption = "Round columns to 5, 0, and 2 digits (respectively).")
```

Table 6: Round columns to 5, 0, and 2 digits (respectively).

X1	X2	X3
0.91481	1e+03	955.94
0.93708	1e+05	47.88
0.28614	1e+07	-1104.60

```
kable(d, digits = 3,
      format.args = list(big.mark = ",", # Use US notation of including a comma (vs period) 
                         scientific = FALSE),
      caption = "Round all data to max 3 decimal places and do not use scientific notaiton."
```

Table 7: Round all data to max 3 decimal places and do not use scientific notaiton.

X1	X2	X3
0.915	1,000	955.936
0.937	100,000	47.885
0.286	10,000,000	-1,104.599

For the second half of the list (5-8), we need the `kableExtra` package. Most things that this package can do will work in both Word and PDF outputs, but Word does not support LaTeX formatting. As a result, the results are pretty iffy. Sticking with PDF output format is recommended. If you need to render to a Word doc, refer to the package documentation for suggestions on how to do so.

This package includes the `kbl()` function, which is *identical* to `kable()`. If you have `kableExtra` loaded you can use either, but the advantage of using `kbl()` is that it will throw an obvious error if you don't have the package loaded, which you might not notice otherwise.

To use styling functions in the `kableExtra` package, you'll create a kable with `kable()` or `kbl()` and pipe (`%>%`) it into the style function. Examples in this section come from the `kableExtra` package documentation.

```
library(kableExtra)
```

Table 8

[!h]
Table 9: Table will appear exactly where you call it.

	mpg	cyl	disp	hp	drat	wt
Mazda RX4	21.0	6	160	110	3.90	2.620
Mazda RX4 Wag	21.0	6	160	110	3.90	2.875
Datsun 710	22.8	4	108	93	3.85	2.320
Hornet 4 Drive	21.4	6	258	110	3.08	3.215
Hornet Sportabout	18.7	8	360	175	3.15	3.440

```
Attaching package: 'kableExtra'


The following object is masked from 'package:dplyr':

    group_rows
```

```
dt <- mtcars[1:5, 1:6]
```

By default, knitr will place tables wherever it thinks the best place in your document for that table will be. This is often great, but sometimes it's a very bad guesser. It will also guess the size/scale your table should be. You can control these things yourself.

In Table 8, the `hold_position` style option "pins" the table where you call it. Most examples from here on will include this argument.

```
kbl(dt,
    caption = "Table will appear exactly where you call it."
    ) %>%
  kable_styling(latex_options = "hold_position")
```

In Table 10, it shows the use of the `scale_down` option (with `hold_position`) to scale a wide table down to fit on the page. This can be used instead of or in addition to printing the table in landscape page format. See the `kableExtra` docs for how to do that.

```
kbl(cbind(dt, dt, dt), # create a very wide table
    caption = "When you have a wide table that will normally go out of the page, and you want
  kable_styling(latex_options = c("striped", "scale_down", "hold_position"))
```

Table 10

[!h]
Table 11: When you have a wide table that will normally go out of the page, and you want to scale down the table to fit the page, you can use the scale down option.

	mpg	cyl	disp	hp	drat	wt	mpg	cyl	disp	hp	drat	wt	mpg	cyl	disp	hp	drat	wt
Mazda RX4	21.0	6	160	110	3.90	2.620	21.0	6	160	110	3.90	2.620	21.0	6	160	110	3.90	2.620
Mazda RX4 Wag	21.0	6	160	110	3.90	2.875	21.0	6	160	110	3.90	2.875	21.0	6	160	110	3.90	2.875
Datsun 710	22.8	4	108	93	3.85	2.320	22.8	4	108	93	3.85	2.320	22.8	4	108	93	3.85	2.320
Hornet 4 Drive	21.4	6	258	110	3.08	3.215	21.4	6	258	110	3.08	3.215	21.4	6	258	110	3.08	3.215
Hornet Sportabout	18.7	8	360	175	3.15	3.440	18.7	8	360	175	3.15	3.440	18.7	8	360	175	3.15	3.440

Table 12

[!h]
Table 13: Use the scale up option to stretch to full page width.

	mpg	cyl	disp	hp	drat	wt
Mazda RX4	21.0	6	160	110	3.90	2.620
Mazda RX4 Wag	21.0	6	160	110	3.90	2.875
Datsun 710	22.8	4	108	93	3.85	2.320
Hornet 4 Drive	21.4	6	258	110	3.08	3.215
Hornet Sportabout	18.7	8	360	175	3.15	3.440

You can also scale *up* a table to make a small/narrow table full width with either the `scale_up` option (see Table 12) or `full_width` argument (see Table 14).

```
kbl(cbind(dt),
    caption = "Use the scale up option to stretch to full page width.") %>%
  kable_styling(latex_options = c("striped", "scale_up", "hold_position"))
```

```
kbl(dt,
    caption = "Stretch a small table to fit the page with the full width option.") %>%
  kable_styling(full_width = T, latex_options = c("hold_position"))
```

When you have very long tables that would extend over multiple pages, use the `longtable` argument, usually in combination with the `repeat_header` option (Table 16).

Table 14

[!h]
Table 15: Stretch a small table to fit the page with the full width option.

	mpg	cyl	disp	hp	drat	wt
Mazda RX4	21.0	6	160	110	3.90	2.620
Mazda RX4 Wag	21.0	6	160	110	3.90	2.875
Datsun 710	22.8	4	108	93	3.85	2.320
Hornet 4 Drive	21.4	6	258	110	3.08	3.215
Hornet Sportabout	18.7	8	360	175	3.15	3.440

```
long_dt <- rbind(mtcars, mtcars) # create a very long table
kbl(long_dt,
    longtable = T,
    caption = "Use the 'longtable' argument in the kbl() to intelligently manage very long ta
 # add_header_above(c(" ", "Group 1" = 5, "Group 2" = 6)) %>%
  kable_styling(latex_options = c("repeat_header", "hold_position"))
```

Table 16: Use the 'longtable' argument in the kbl() to intelligently manage very long tables. Use the repeat header LaTeX option

	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
Mazda RX4	21.0	6	160.0	110	3.90	2.620	16.46	0	1	4	4
Mazda RX4 Wag	21.0	6	160.0	110	3.90	2.875	17.02	0	1	4	4
Datsun 710	22.8	4	108.0	93	3.85	2.320	18.61	1	1	4	1
Hornet 4 Drive	21.4	6	258.0	110	3.08	3.215	19.44	1	0	3	1
Hornet Sportabout	18.7	8	360.0	175	3.15	3.440	17.02	0	0	3	2
Valiant	18.1	6	225.0	105	2.76	3.460	20.22	1	0	3	1
Duster 360	14.3	8	360.0	245	3.21	3.570	15.84	0	0	3	4
Merc 240D	24.4	4	146.7	62	3.69	3.190	20.00	1	0	4	2
Merc 230	22.8	4	140.8	95	3.92	3.150	22.90	1	0	4	2
Merc 280	19.2	6	167.6	123	3.92	3.440	18.30	1	0	4	4
Merc 280C	17.8	6	167.6	123	3.92	3.440	18.90	1	0	4	4
Merc 450SE	16.4	8	275.8	180	3.07	4.070	17.40	0	0	3	3
Merc 450SL	17.3	8	275.8	180	3.07	3.730	17.60	0	0	3	3

Table 16: Use the 'longtable' argument in the kbl() to intelligently manage very long tables. Use the repeat header LaTeX option *(continued)*

	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
Merc 450SLC	15.2	8	275.8	180	3.07	3.780	18.00	0	0	3	3
Cadillac Fleetwood	10.4	8	472.0	205	2.93	5.250	17.98	0	0	3	4
Lincoln Continental	10.4	8	460.0	215	3.00	5.424	17.82	0	0	3	4
Chrysler Imperial	14.7	8	440.0	230	3.23	5.345	17.42	0	0	3	4
Fiat 128	32.4	4	78.7	66	4.08	2.200	19.47	1	1	4	1
Honda Civic	30.4	4	75.7	52	4.93	1.615	18.52	1	1	4	2
Toyota Corolla	33.9	4	71.1	65	4.22	1.835	19.90	1	1	4	1
Toyota Corona	21.5	4	120.1	97	3.70	2.465	20.01	1	0	3	1
Dodge Challenger	15.5	8	318.0	150	2.76	3.520	16.87	0	0	3	2
AMC Javelin	15.2	8	304.0	150	3.15	3.435	17.30	0	0	3	2
Camaro Z28	13.3	8	350.0	245	3.73	3.840	15.41	0	0	3	4
Pontiac Firebird	19.2	8	400.0	175	3.08	3.845	17.05	0	0	3	2
Fiat X1-9	27.3	4	79.0	66	4.08	1.935	18.90	1	1	4	1
Porsche 914-2	26.0	4	120.3	91	4.43	2.140	16.70	0	1	5	2
Lotus Europa	30.4	4	95.1	113	3.77	1.513	16.90	1	1	5	2
Ford Pantera L	15.8	8	351.0	264	4.22	3.170	14.50	0	1	5	4
Ferrari Dino	19.7	6	145.0	175	3.62	2.770	15.50	0	1	5	6
Maserati Bora	15.0	8	301.0	335	3.54	3.570	14.60	0	1	5	8
Volvo 142E	21.4	4	121.0	109	4.11	2.780	18.60	1	1	4	2
Mazda RX41	21.0	6	160.0	110	3.90	2.620	16.46	0	1	4	4
Mazda RX4 Wag1	21.0	6	160.0	110	3.90	2.875	17.02	0	1	4	4
Datsun 7101	22.8	4	108.0	93	3.85	2.320	18.61	1	1	4	1
Hornet 4 Drive1	21.4	6	258.0	110	3.08	3.215	19.44	1	0	3	1
Hornet Sportabout1	18.7	8	360.0	175	3.15	3.440	17.02	0	0	3	2
Valiant1	18.1	6	225.0	105	2.76	3.460	20.22	1	0	3	1
Duster 3601	14.3	8	360.0	245	3.21	3.570	15.84	0	0	3	4
Merc 240D1	24.4	4	146.7	62	3.69	3.190	20.00	1	0	4	2
Merc 2301	22.8	4	140.8	95	3.92	3.150	22.90	1	0	4	2
Merc 2801	19.2	6	167.6	123	3.92	3.440	18.30	1	0	4	4
Merc 280C1	17.8	6	167.6	123	3.92	3.440	18.90	1	0	4	4
Merc 450SE1	16.4	8	275.8	180	3.07	4.070	17.40	0	0	3	3
Merc 450SL1	17.3	8	275.8	180	3.07	3.730	17.60	0	0	3	3
Merc 450SLC1	15.2	8	275.8	180	3.07	3.780	18.00	0	0	3	3
Cadillac Fleetwood1	10.4	8	472.0	205	2.93	5.250	17.98	0	0	3	4
Lincoln Continental1	10.4	8	460.0	215	3.00	5.424	17.82	0	0	3	4
Chrysler Imperial1	14.7	8	440.0	230	3.23	5.345	17.42	0	0	3	4
Fiat 1281	32.4	4	78.7	66	4.08	2.200	19.47	1	1	4	1

Table 17

[!h]

Table 18: Align your table (not the text, the whole table), 'left' or 'center' (the default) with the position argument.

	mpg	cyl	disp	hp	drat	wt
Mazda RX4	21.0	6	160	110	3.90	2.620
Mazda RX4 Wag	21.0	6	160	110	3.90	2.875
Datsun 710	22.8	4	108	93	3.85	2.320
Hornet 4 Drive	21.4	6	258	110	3.08	3.215
Hornet Sportabout	18.7	8	360	175	3.15	3.440

Table 16: Use the 'longtable' argument in the kbl() to intelligently manage very long tables. Use the repeat header LaTeX option *(continued)*

	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
Honda Civic1	30.4	4	75.7	52	4.93	1.615	18.52	1	1	4	2
Toyota Corolla1	33.9	4	71.1	65	4.22	1.835	19.90	1	1	4	1
Toyota Corona1	21.5	4	120.1	97	3.70	2.465	20.01	1	0	3	1
Dodge Challenger1	15.5	8	318.0	150	2.76	3.520	16.87	0	0	3	2
AMC Javelin1	15.2	8	304.0	150	3.15	3.435	17.30	0	0	3	2
Camaro Z281	13.3	8	350.0	245	3.73	3.840	15.41	0	0	3	4
Pontiac Firebird1	19.2	8	400.0	175	3.08	3.845	17.05	0	0	3	2
Fiat X1-91	27.3	4	79.0	66	4.08	1.935	18.90	1	1	4	1
Porsche 914-21	26.0	4	120.3	91	4.43	2.140	16.70	0	1	5	2
Lotus Europa1	30.4	4	95.1	113	3.77	1.513	16.90	1	1	5	2
Ford Pantera L1	15.8	8	351.0	264	4.22	3.170	14.50	0	1	5	4
Ferrari Dino1	19.7	6	145.0	175	3.62	2.770	15.50	0	1	5	6
Maserati Bora1	15.0	8	301.0	335	3.54	3.570	14.60	0	1	5	8
Volvo 142E1	21.4	4	121.0	109	4.11	2.780	18.60	1	1	4	2

By default, tables are aligned center on the page, but they can also be aligned left or right. Note that alignment does not apply to full width tables. Table 17 uses the `position = "left" argument` to align left. Oddly, aligning right in the same way doesn't work as expected. In theory you can "float right" rather than position right, but this can produce errors. It's unlikely you'll need to right align a table though.

```
kbl(dt,
    caption = "Align your table (not the text, the whole table), 'left' or 'center' (the defa
  kable_styling(position = "left", latex_options = c("hold_position"))
```

Table 19

Table 20: This table uses column specifications to make many visual modifications to individual rows. The final line shows how the same function can modify just one cell in a column in a similar way.

	mpg	cyl	disp	hp	drat	wt	qsec	vs
Mazda RX4	21.0	6	160.0	110	3.90	2.620	16.46	0
Mazda RX4 Wag	21.0	6	160.0	110	3.90	2.875	17.02	0
Datsun 710	22.8	4	108.0	93	3.85	2.320	18.61	1
Hornet 4 Drive	21.4	6	258.0	110	3.08	3.215	19.44	1
Hornet Sportabout	18.7	8	360.0	175	3.15	3.440	17.02	0
Valiant	18.1	6	225.0	105	2.76	3.460	20.22	1
Duster 360	14.3	8	360.0	245	3.21	3.570	15.84	0
Merc 240D	24.4	4	146.7	62	3.69	3.190	20.00	~~1~~

You can make style adjustments to the whole table (e.g., font fact, background color, bold/italic) or to specific rows and columns. Table 19 includes three `column_spec` functions to modify properties in columns 2, 6, and 9.

```
mtcars[1:8, 1:8] %>%
kbl(booktabs = T,
    caption = "This table uses column specifications to make many visual modifications to ind
  kable_paper(full_width = F) %>%
  column_spec(2, color = spec_color(mtcars$mpg[1:8]),
              link = "https://haozhu233.github.io/kableExtra") %>%
  column_spec(6, color = "white",
              background = spec_color(mtcars$drat[1:8], end = 0.7),
              popover = paste("am:", mtcars$am[1:8])) %>%
#  The c(rep(XX,7), XX) bit here is basically saying
#  in this column (9) treat the first 7 columns one way and then the rest (the last)
#  this other way. e.g., strikeout should be FALSE for the 1st seven rows but TRUE
#  for the rest, color should be BLACK for 1-7 but RED for the rest
  column_spec(9, strikeout = c(rep(F, 7), T), bold = c(rep(F, 7), T),
              color = c(rep("black", 7), "red"))
```

Table 21 includes two `row_spec` functions to modify properties in rows 1 and rows 3-5.

```
kbl(dt, booktabs = T,
    caption = "You can specify styling for rows in the same way.") %>%
  kable_styling("striped", full_width = F) %>%
```

Table 21

Table 22: You can specify styling for rows in the same way.

	mpg	cyl	disp	hp	drat	wt
~~Mazda RX4~~	~~21.0~~	~~6~~	~~160~~	~~110~~	~~3.90~~	**~~2.620~~**
Mazda RX4 Wag	21.0	6	160	110	3.90	**2.875**
Datsun 710	**22.8**	**4**	**108**	**93**	**3.85**	**2.320**
Hornet 4 Drive	**21.4**	**6**	**258**	**110**	**3.08**	**3.215**
Hornet Sportabout	**18.7**	**8**	**360**	**175**	**3.15**	**3.440**

Table 23

[!h]
Table 24: Create grouped columns by adding a header.

	Group 1		Group 2		Group 3	
	mpg	cyl	disp	hp	drat	wt
Mazda RX4	21.0	6	160	110	3.90	2.620
Mazda RX4 Wag	21.0	6	160	110	3.90	2.875
Datsun 710	22.8	4	108	93	3.85	2.320
Hornet 4 Drive	21.4	6	258	110	3.08	3.215
Hornet Sportabout	18.7	8	360	175	3.15	3.440

```
  column_spec(7, border_left = T, bold = T) %>%
  row_spec(1, strikeout = T) %>%
  row_spec(3:5, bold = T, color = "white", background = "black")
```

You can "group" rows and columns, basically equivalent to merging cells in a spreadsheet. This is commonly needed for multi-level headers or organizing non-rectangular data (like a lot of model output). Table 23 uses the `add_header_above()` function to do so.

```
kbl(dt,
    caption = "Create grouped columns by adding a header.") %>%
  kable_styling(latex_options = c("striped", "hold_position")) %>%
  # The syntax within the list is "name of group header = [number of columns to be grouped u
  add_header_above(c(" " = 1, "Group 1" = 2, "Group 2" = 2, "Group 3" = 2))
```

You can group more than once and format these header rows individually. Use the total number of rows to specify how many columns to group, not the number of rows in the grouped row below it (Table 25).

Table 25

[!h]
Table 26: Group multiple times and format these header rows individually.

	Group 6					
	Group 4				Group 5	
	Group 1		Group 2		Group 3	
	mpg	cyl	disp	hp	drat	wt
Mazda RX4	21.0	6	160	110	3.90	2.620
Mazda RX4 Wag	21.0	6	160	110	3.90	2.875
Datsun 710	22.8	4	108	93	3.85	2.320
Hornet 4 Drive	21.4	6	258	110	3.08	3.215
Hornet Sportabout	18.7	8	360	175	3.15	3.440

```
kbl(dt,
    caption = "Group multiple times and format these header rows individually.") %>%
  kable_styling(latex_options = c("striped", "hold_position")) %>%
  # If you don't specify how many columns to group for that group name, it assumes =1
  add_header_above(c(" ", "Group 1" = 2, "Group 2" = 2, "Group 3" = 2)) %>%
  add_header_above(c(" ", "Group 4" = 4, "Group 5" = 2)) %>%
  add_header_above(c(" ", "Group 6" = 6), bold = T, italic = T)
```

Grouping rows is a little different. Table 27 uses the `pack_rows()` function to effectively add "header rows" throughout the table.

```
kbl(mtcars[1:10, 1:6],
    caption = "Group rows using packing.") %>%
  kable_styling(latex_options = c("striped", "hold_position")) %>%
  # syntax here is ("group label", [int of first row to include in group][int of last row to
  pack_rows("Group 1", 4, 7) %>%
  pack_rows("Group 2", 8, 10)
collapse_rows_dt <- data.frame(C1 = c(rep("a", 10), rep("b", 5)),
                               C2 = c(rep("c", 7), rep("d", 3), rep("c", 2), rep("d", 3)),
                               C3 = 1:15,
                               C4 = sample(c(0,1), 15, replace = TRUE))
```

Alternatively, use `collapse_rows()` in Table 29 for more of a "merge cells" effect.

```
collapse_rows_dt <- data.frame(C1 = c(rep("a", 10), rep("b", 5)),
                               C2 = c(rep("c", 7), rep("d", 3), rep("c", 2), rep("d", 3)),
```

Table 27

[!h]
Table 28: Group rows using packing.

	mpg	cyl	disp	hp	drat	wt
Mazda RX4	21.0	6	160.0	110	3.90	2.620
Mazda RX4 Wag	21.0	6	160.0	110	3.90	2.875
Datsun 710	22.8	4	108.0	93	3.85	2.320
Group 1						
Hornet 4 Drive	21.4	6	258.0	110	3.08	3.215
Hornet Sportabout	18.7	8	360.0	175	3.15	3.440
Valiant	18.1	6	225.0	105	2.76	3.460
Duster 360	14.3	8	360.0	245	3.21	3.570
Group 2						
Merc 240D	24.4	4	146.7	62	3.69	3.190
Merc 230	22.8	4	140.8	95	3.92	3.150
Merc 280	19.2	6	167.6	123	3.92	3.440

```
                              C3 = 1:15,
                              C4 = sample(c(0,1), 15, replace = TRUE))

# You will only see the "collapse" happen when you knit
# not when you run the chunk
kbl(collapse_rows_dt, booktabs = T, align = "c",
    caption = "Collapse rows for more of a 'merge cells' effect.") %>%
  column_spec(1, bold=T) %>%
  collapse_rows(columns = 1:2,
                latex_hline = "major",
                row_group_label_position = "first")
```

Finally, you can add notes or footnotes below your table with the `footnote()` function. Table 31 includes the 'general' argument for an unlabeled note and 'number', 'alphabet', and 'symbol' for footnotes with ordered labels.

```
kbl(dt, align = "c",
    caption = "Add unordered (general) and ordered footnotes.") %>%
  kable_styling(full_width = F) %>%
  footnote(general = "Here is a general comments of the table. ",
           number = c("Footnote 1; ", "Footnote 2; "),
           alphabet = c("Footnote A; ", "Footnote B; "),
```

Table 29

Table 30: Collapse rows for more of a 'merge cells' effect.

C1	C2	C3	C4
a	c	1	0
		2	1
		3	0
		4	0
		5	0
		6	0
		7	1
	d	8	1
		9	1
		10	1
b	c	11	0
		12	1
	d	13	0
		14	1
		15	1

```
            symbol = c("Footnote Symbol 1; ", "Footnote Symbol 2")
)
```

There is **so much more** you can do with kables! I am personally a huge fan of kables and the kableExtra package because it works beautifully with papaja, but you can also explore other packages for making beautiful tables:

- flextable
- gt table
- xtable
- stargazer
    - This one is designed to simplify tables for regression output

Table: flextable

```
library(flextable)
```

Table 31

Table 32: Add unordered (general) and ordered footnotes.

	mpg	cyl	disp	hp	drat	wt
Mazda RX4	21.0	6	160	110	3.90	2.620
Mazda RX4 Wag	21.0	6	160	110	3.90	2.875
Datsun 710	22.8	4	108	93	3.85	2.320
Hornet 4 Drive	21.4	6	258	110	3.08	3.215
Hornet Sportabout	18.7	8	360	175	3.15	3.440

Note:

Here is a general comments of the table.

[1] Footnote 1;

[2] Footnote 2;

[a] Footnote A;

[b] Footnote B;

[*] Footnote Symbol 1;

[†] Footnote Symbol 2

```
Attaching package: 'flextable'

The following objects are masked from 'package:kableExtra':

    as_image, footnote

The following object is masked from 'package:purrr':

    compose
```

```
tibble(Letters = c("A", "B", "C"),
       Numbers = 1:3) %>%
  flextable() %>%
  theme_apa()
```

Table 33: My Table

Letters	Numbers
A	1
B	2
C	3

Table: papaja and apa_table()

Alternatively, **in .rmd files**, we can use "wrapper" functions that add in a collection of tweaks for us, like the `apa_table()` function in `papaja`, which will knit the table in APA formatting.

Table: stargazer

```
# build up two regression models
mod_1 <- lm(Sepal.Length ~ Sepal.Width, data = iris)

mod_2 <- lm(Sepal.Length ~ Sepal.Width + Species, data = iris)

# export a table with the results of these two regression models
stargazer(mod_1, mod_2,type = "text")
```

```
===============================================================
                              Dependent variable:
                    ---------------------------------------
                                    Sepal.Length
                              (1)                   (2)
---------------------------------------------------------------
Sepal.Width                -0.223                0.804***
                           (0.155)               (0.106)

Speciesversicolor                                1.459***
                                                 (0.112)

Speciesvirginica                                 1.947***
                                                 (0.100)
```

```
Constant                   6.526***              2.251***
                           (0.479)               (0.370)

---------------------------------------------------------------
Observations                150                   150
R2                         0.014                 0.726
Adjusted R2                0.007                 0.720
Residual Std. Error  0.825 (df = 148)       0.438 (df = 146)
F Statistic         2.074 (df = 1; 148) 128.888*** (df = 3; 146)
===============================================================
Note:                                *p<0.1; **p<0.05; ***p<0.01
```

```r
# you can customize the table by renaming variables, adding title, omitting some statistic, a
stargazer(mod_1, mod_2,
          # add title to the table
          title = "Regression model results",
          # rename dependent variable
          dep.var.labels = "The length of sepal",
          # rename independent variable
          covariate.labels = c("The width of sepal", "Versicolor", "Virginica"),
          # "n" refers to the number of observations, "f" refers to the F stastic
          omit.stat = c("n","f"),
          # add confidence interval and set the level to 0.95
          ci = TRUE, ci.level = 0.95,
          # keep only two digits
          digits = 2,
          #display in plain text (rather than LaTeX)
          type = "text")
```

```
Regression model results
================================================
                         Dependent variable:
                     ---------------------------
                         The length of sepal
                           (1)             (2)
------------------------------------------------
The width of sepal        -0.22          0.80***
                      (-0.53, 0.08)   (0.60, 1.01)

Versicolor                                1.46***
```

```
                                      (1.24, 1.68)

Virginica                               1.95***
                                      (1.75, 2.14)

Constant                 6.53***        2.25***
                       (5.59, 7.46)    (1.53, 2.98)

-------------------------------------------------
R2                        0.01            0.73
Adjusted R2               0.01            0.72
Residual Std. Error 0.83 (df = 148) 0.44 (df = 146)
=================================================
Note:                    *p<0.1; **p<0.05; ***p<0.01
```

```r
stargazer(mod_1, mod_2,
          # add title to the table
          title = "Regression model results",
          # View documentation to see journal styles available
          #style = "asr",
          # rename dependent variable
          dep.var.labels = "The length of sepal",
          # rename independent variable
          covariate.labels = c("The width of sepal", "Versicolor", "Virginica"),
          # "n" refers to the number of observations, "f" refers to the F stastic
          omit.stat = c("n","f"),
          # add confidence interval and set the level to 0.95
          ci = TRUE, ci.level = 0.95,
          # keep only two digits
          digits = 2)
```

% Table created by stargazer v.5.2.3 by Marek Hlavac, Social Policy Institute. E-mail: marek.hlavac at gmail.com % Date and time: Fri, Feb 14, 2025 - 2:21:35 PM

```r
# save the output in html format
stargazer(mod_1, type = "html", out = "models.htm")
```

```
<table style="text-align:center"><tr><td colspan="2" style="border-bottom: 1px solid black">
<tr><td></td><td colspan="1" style="border-bottom: 1px solid black"></td></tr>
<tr><td style="text-align:left"></td><td>Sepal.Length</td></tr>
```

Table 34: Regression model results

	Dependent variable:	
	The length of sepal	
	(1)	(2)
The width of sepal	−0.22	0.80***
	(−0.53, 0.08)	(0.60, 1.01)
Versicolor		1.46***
		(1.24, 1.68)
Virginica		1.95***
		(1.75, 2.14)
Constant	6.53***	2.25***
	(5.59, 7.46)	(1.53, 2.98)
R^2	0.01	0.73
Adjusted R^2	0.01	0.72
Residual Std. Error	0.83 (df = 148)	0.44 (df = 146)
Note:		*p<0.1; **p<0.05; ***p<0.01

```
<tr><td colspan="2" style="border-bottom: 1px solid black"></td></tr><tr><td style="text-ali
<tr><td style="text-align:left"></td><td>(0.155)</td></tr>
<tr><td style="text-align:left"></td><td></td></tr>
<tr><td style="text-align:left">Constant</td><td>6.526<sup>***</sup></td></tr>
<tr><td style="text-align:left"></td><td>(0.479)</td></tr>
<tr><td style="text-align:left"></td><td></td></tr>
<tr><td colspan="2" style="border-bottom: 1px solid black"></td></tr><tr><td style="text-ali
<tr><td style="text-align:left">R<sup>2</sup></td><td>0.014</td></tr>
<tr><td style="text-align:left">Adjusted R<sup>2</sup></td><td>0.007</td></tr>
<tr><td style="text-align:left">Residual Std. Error</td><td>0.825 (df = 148)</td></tr>
<tr><td style="text-align:left">F Statistic</td><td>2.074 (df = 1; 148)</td></tr>
<tr><td colspan="2" style="border-bottom: 1px solid black"></td></tr><tr><td style="text-ali
</table>
```