5. Tables with flextable

Publication-ready tables for Word documents

Dr. Paul Schmidt

The default R output for tables looks unprofessional in Word documents — it appears as console output with monospace font. In this chapter, you will learn how to create professional, formatted tables using the flextable package that integrate seamlessly into Word documents.

Why flextable?

There are several R packages for tables (kable, gt, huxtable, flextable), but for Word output, **flextable** is the best choice:

- Native Word format (no detour via HTML)
- Full control over formatting
- Actively maintained and well documented
- Part of the “Officeverse” ecosystem

> ℹ Note
>
> The **gt** package is excellent for HTML output, but its Word support is more limited. For Word documents, I recommend flextable.

Basics

Creating a simple table

The simplest way to create a flextable:

```
adelie %>%
  head(5) %>%
  select(island, bill_length_mm, bill_depth_mm, body_mass_g) %>%
  flextable()
```

island	bill_length_mm	bill_depth_mm	body_mass_g
Torgersen	39.1	18.7	3,750
Torgersen	39.5	17.4	3,800
Torgersen	40.3	18.0	3,250
Torgersen	36.7	19.3	3,450
Torgersen	39.3	20.6	3,650

This already looks much better than `print(df)`! But the column widths are not optimal yet.

Automatic column widths

With `autofit()`, flextable automatically adjusts column widths:

```
adelie %>%
  head(5) %>%
  select(island, bill_length_mm, bill_depth_mm, body_mass_g) %>%
  flextable() %>%
  autofit()
```

island	bill_length_mm	bill_depth_mm	body_mass_g
Torgersen	39.1	18.7	3,750
Torgersen	39.5	17.4	3,800
Torgersen	40.3	18.0	3,250
Torgersen	36.7	19.3	3,450
Torgersen	39.3	20.6	3,650

Tip

`autofit()` should generally be placed at the end of the flextable pipeline, after all other formatting has been applied.

Renaming and formatting columns

Changing column headers

The automatic column names from the dataframe are often not ideal for a report:

```
adelie %>%
  head(5) %>%
  select(island, bill_length_mm, bill_depth_mm, body_mass_g) %>%
  flextable() %>%
  set_header_labels(
    island = "Island",
    bill_length_mm = "Bill Length (mm)",
    bill_depth_mm = "Bill Depth (mm)",
    body_mass_g = "Body Mass (g)"
  ) %>%
  autofit()
```

Island	Bill Length (mm)	Bill Depth (mm)	Body Mass (g)
Torgersen	39.1	18.7	3,750
Torgersen	39.5	17.4	3,800
Torgersen	40.3	18.0	3,250
Torgersen	36.7	19.3	3,450
Torgersen	39.3	20.6	3,650

Formatting numbers

For scientific tables, you often need a specific number of decimal places:

```
adelie %>%
  head(5) %>%
  select(island, bill_length_mm, bill_depth_mm, body_mass_g) %>%
  flextable() %>%
  set_header_labels(
    island = "Island",
    bill_length_mm = "Bill Length (mm)",
    bill_depth_mm = "Bill Depth (mm)",
    body_mass_g = "Body Mass (g)"
  ) %>%
  colformat_double(j = c("bill_length_mm", "bill_depth_mm"), digits = 1) %>%
  colformat_double(j = "body_mass_g", digits = 0) %>%
  autofit()
```

Island	Bill Length (mm)	Bill Depth (mm)	Body Mass (g)
Torgersen	39.1	18.7	3,750
Torgersen	39.5	17.4	3,800
Torgersen	40.3	18.0	3,250
Torgersen	36.7	19.3	3,450
Torgersen	39.3	20.6	3,650

Formatting and styling

Font and size

```
adelie %>%
  head(5) %>%
  select(island, bill_length_mm, body_mass_g) %>%
  flextable() %>%
  font(fontname = "Arial", part = "all") %>%
  fontsize(size = 10, part = "body") %>%
  fontsize(size = 11, part = "header") %>%
  autofit()
```

island	bill_length_mm	body_mass_g
Torgersen	39.1	3,750
Torgersen	39.5	3,800
Torgersen	40.3	3,250
Torgersen	36.7	3,450
Torgersen	39.3	3,650

Alignment

```
adelie %>%
  head(5) %>%
  select(island, bill_length_mm, body_mass_g) %>%
  flextable() %>%
  align(j = 1, align = "left", part = "all") %>%
  align(j = 2:3, align = "center", part = "all") %>%
  autofit()
```

island	bill_length_mm	body_mass_g
Torgersen	39.1	3,750
Torgersen	39.5	3,800
Torgersen	40.3	3,250
Torgersen	36.7	3,450
Torgersen	39.3	3,650

Borders

```
adelie %>%
  head(5) %>%
  select(island, bill_length_mm, body_mass_g) %>%
  flextable() %>%
  border_remove() %>%
  hline_top(border = fp_border(width = 2), part = "header") %>%
  hline_bottom(border = fp_border(width = 1), part = "header") %>%
  hline_bottom(border = fp_border(width = 2), part = "body") %>%
  autofit()
```

island	bill_length_mm	body_mass_g
Torgersen	39.1	3,750
Torgersen	39.5	3,800
Torgersen	40.3	3,250
Torgersen	36.7	3,450
Torgersen	39.3	3,650

Bold headers

```
adelie %>%
  head(5) %>%
  select(island, bill_length_mm, body_mass_g) %>%
  flextable() %>%
  bold(part = "header") %>%
  autofit()
```

island	**bill_length_mm**	**body_mass_g**
Torgersen	39.1	3,750
Torgersen	39.5	3,800
Torgersen	40.3	3,250
Torgersen	36.7	3,450
Torgersen	39.3	3,650

Creating a summary table

For our penguin report, we create a descriptive statistics table:

```
summary_table <- adelie %>%
  summarise(
    n = n(),
    `Bill Length (mm)` = mean(bill_length_mm),
    `SD` = sd(bill_length_mm),
    `Bill Depth (mm)` = mean(bill_depth_mm),
    `SD ` = sd(bill_depth_mm),
    `Body Mass (g)` = mean(body_mass_g),
    `SD  ` = sd(body_mass_g)
  )

summary_table %>%
  flextable() %>%
  colformat_double(digits = 1) %>%
  colformat_double(j = "n", digits = 0) %>%
  set_header_labels(n = "N") %>%
  bold(part = "header") %>%
  autofit()
```

N	Bill Length (mm)	SD	Bill Depth (mm)	SD	Body Mass (g)	SD
146	38.8	2.7	18.3	1.2	3,706.2	458.6

Grouped tables

A table with statistics per island:

```
adelie %>%
  group_by(island) %>%
  summarise(
    N = n(),
    `Bill Length` = mean(bill_length_mm),
    `Body Mass` = mean(body_mass_g),
    .groups = "drop"
  ) %>%
  flextable() %>%
  set_header_labels(island = "Island") %>%
  colformat_double(j = c("Bill Length"), digits = 1) %>%
  colformat_double(j = c("Body Mass"), digits = 0) %>%
  bold(part = "header") %>%
  hline_top(border = fp_border(width = 2), part = "header") %>%
  hline_bottom(border = fp_border(width = 1), part = "header") %>%
  hline_bottom(border = fp_border(width = 2), part = "body") %>%
  autofit()
```

Island	N	Bill Length	Body Mass
Biscoe	44	39.0	3,710
Dream	55	38.5	3,701
Torgersen	47	39.0	3,709

Table captions in Quarto

To add a table caption, use the chunk option `tbl-cap`:

~~~
```{r}
#| label: tbl-summary
#| tbl-cap: "Descriptive statistics of Adelie penguins"

summary_table %>%
  flextable() %>%
  autofit()
```
~~~

The label must start with `tbl-` for Quarto to recognize it as a table and enable cross-references (see Chapter 7).

Usage in Word documents

For correct display in Word documents, the chunk option `output: asis` is often no longer needed (current flextable versions detect the format automatically). If the table does not appear correctly, you can add it:

~~~~
```{r}
#| label: tbl-example
#| output: asis

my_table %>%
  flextable() %>%
  autofit()
```
~~~~

Complete example

Here is a complete, publication-ready table:

```r
adelie %>%
  group_by(island, sex) %>%
  summarise(
    N = n(),
    `Bill Length (mm)` = mean(bill_length_mm),
    `Body Mass (g)` = mean(body_mass_g),
    .groups = "drop"
  ) %>%
  flextable() %>%
  set_header_labels(
    island = "Island",
    sex = "Sex"
  ) %>%
  colformat_double(j = "Bill Length (mm)", digits = 1) %>%
  colformat_double(j = "Body Mass (g)", digits = 0) %>%
  font(fontname = "Arial", part = "all") %>%
  fontsize(size = 10, part = "all") %>%
  bold(part = "header") %>%
  align(align = "center", part = "header") %>%
  align(j = 1:2, align = "left", part = "body") %>%
  align(j = 3:5, align = "right", part = "body") %>%
  border_remove() %>%
  hline_top(border = fp_border(width = 1.5), part = "header") %>%
  hline_bottom(border = fp_border(width = 0.75), part = "header") %>%
  hline_bottom(border = fp_border(width = 1.5), part = "body") %>%
  autofit()
```

Island	Sex	N	Bill Length (mm)	Body Mass (g)
Biscoe	female	22	37.4	3,369
Biscoe	male	22	40.6	4,050
Dream	female	27	36.9	3,344
Dream	male	28	40.1	4,046
Torgersen	female	24	37.6	3,396
Torgersen	male	23	40.6	4,035

Exercise: Create a summary table

1. Create a table with the number of penguins per island and sex
2. Add a column with the average weight
3. Format the table professionally (font, borders, alignment)
4. Add a table caption with `tbl-cap`

Further resources

- flextable book — Comprehensive documentation
- flextable gallery — Examples and inspiration
- Officeverse — The ecosystem around flextable

What is next

Now we can create professional tables. In Chapter 6, we will learn how to optimally integrate ggplot2 graphics into Quarto documents — with the right size, resolution, and captions.

Bibliography