Problem set 3

Put your name here

Table of contents

3. Read in the separate data files. Make sure you have the `tidyverse` package loaded.

```r
# load packages
library(tidyverse)
```

```
-- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
v dplyr     1.1.4     v readr     2.1.5
v forcats   1.0.0     v stringr   1.5.1
v ggplot2   3.5.1     v tibble    3.2.1
v lubridate 1.9.3     v tidyr     1.3.1
v purrr     1.0.2
-- Conflicts ------------------------------------------ tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to becom
```

```r
# Load the separate datasets
fellowship <- read_csv("../data/The_Fellowship_Of_The_Ring.csv")
```

```
Rows: 3 Columns: 4
-- Column specification --------------------------------------------------------
Delimiter: ","
chr (2): Film, Species
dbl (2): Female, Male

i Use `spec()` to retrieve the full column specification for this data.
i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```r
tt <- read_csv("../data/The_Two_Towers.csv")
```

```
Rows: 3 Columns: 4
-- Column specification --------------------------------------------------------
Delimiter: ","
chr (2): Film, Species
dbl (2): Female, Male

i Use `spec()` to retrieve the full column specification for this data.
i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```r
rotk <- read_csv("../data/The_Return_Of_The_King.csv")
```

```
Rows: 3 Columns: 4
-- Column specification --------------------------------------------------------
Delimiter: ","
chr (2): Film, Species
dbl (2): Female, Male

i Use `spec()` to retrieve the full column specification for this data.
i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

4. Use the `bind_rows` function to merge the three data sets into a single data set. We haven't seen this function yet, look it up. Call the new merged data frame `lotr` (for "lord of the rings").

```r
# bind_rows() stacks data frames on top of each other
lotr <- bind_rows(fellowship, tt, rotk)
```

5. We later want to plot gender differences. Have a look at the data. Why is it not yet in a tidy format? Explain. Then use `pivot_longer` to reshape the data frame by adding two new variables, `Gender` and `Words`, to the data frame.

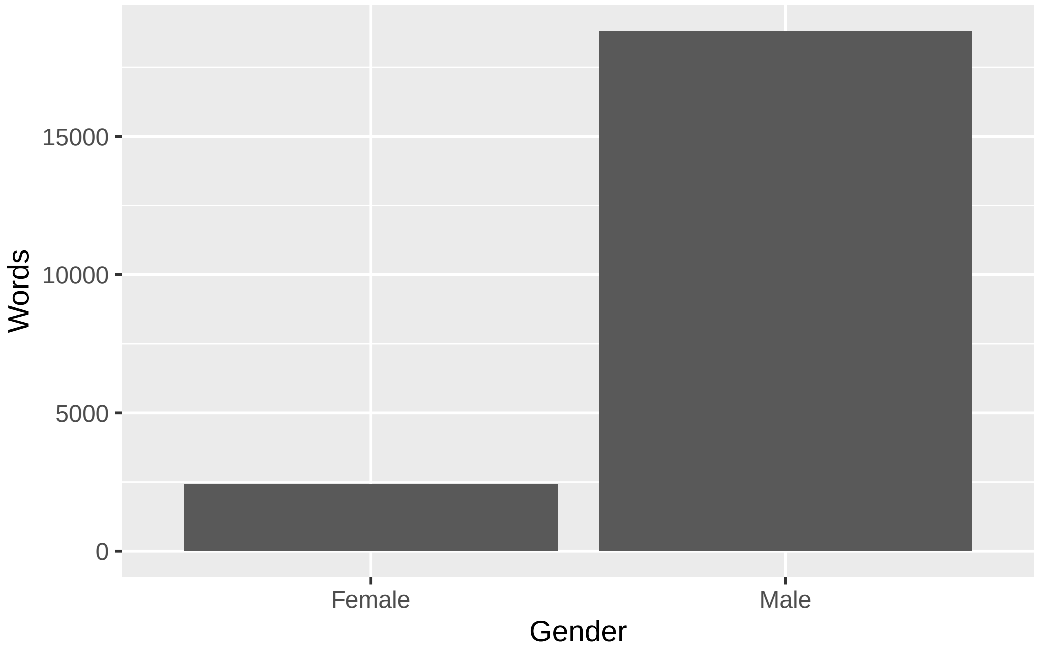
```r
# Make this wide data tidy
lotr <- lotr |>
  # This is the new way to make data long
  pivot_longer(cols = c(Female, Male),
               names_to = "Gender", values_to = "Words")
```

6. Does a certain gender dominate a movie? (Hint: Make a new summary data frame for which you group by `Gender` and then count sum the words.)

```
summary_data <- lotr |>
  group_by(Gender) |>
  summarise(Words = sum(Words))
```

7. Graph your summarized data. (Hint: use `geom_col` and the `Words` and `Gender` variables.)

```
ggplot(summary_data,
       aes(x = Gender, y = Words)) +
  geom_col()
```



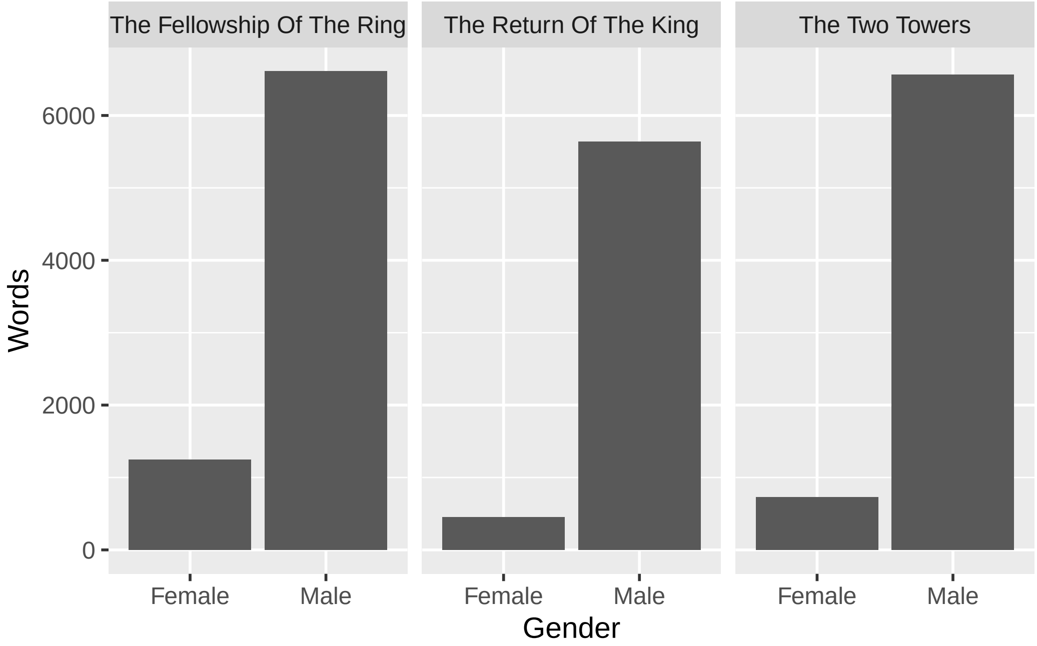
8. You've just plotted the averages across films. (Hint: Make a new summary data frame for which you group by both `Gender` and `Film` and then count sum the words.)

```
summary_data <- lotr |>
  group_by(Gender, Film) |>
  summarise(Words = sum(Words))
```

```
`summarise()` has grouped output by 'Gender'. You can override using the
`.groups` argument.
```

9. Try to make a new plot in which you differentiate between the different films (Hint: use faceting by `Gender` or `Film`).

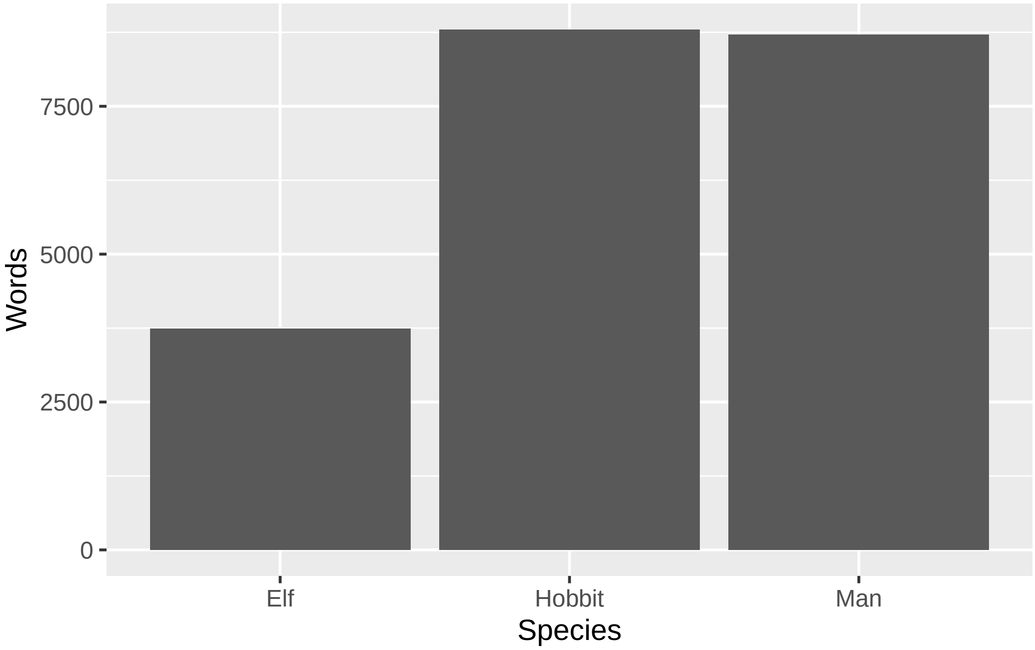
```
ggplot(summary_data,
       aes(x = Gender, y = Words)) +
  geom_col() +
  facet_wrap(vars(Film))
```



10. How about species? Does the dominant species differ on average (don't differentiate between the three movies here)? (Hint: Proceed just as for `Gender` in the beginning: make a new summary data frame for which you group by `Species` and then count sum the words.)

```
summary_data <- lotr |>
  group_by(Species) |>
  summarise(Words = sum(Words))
```

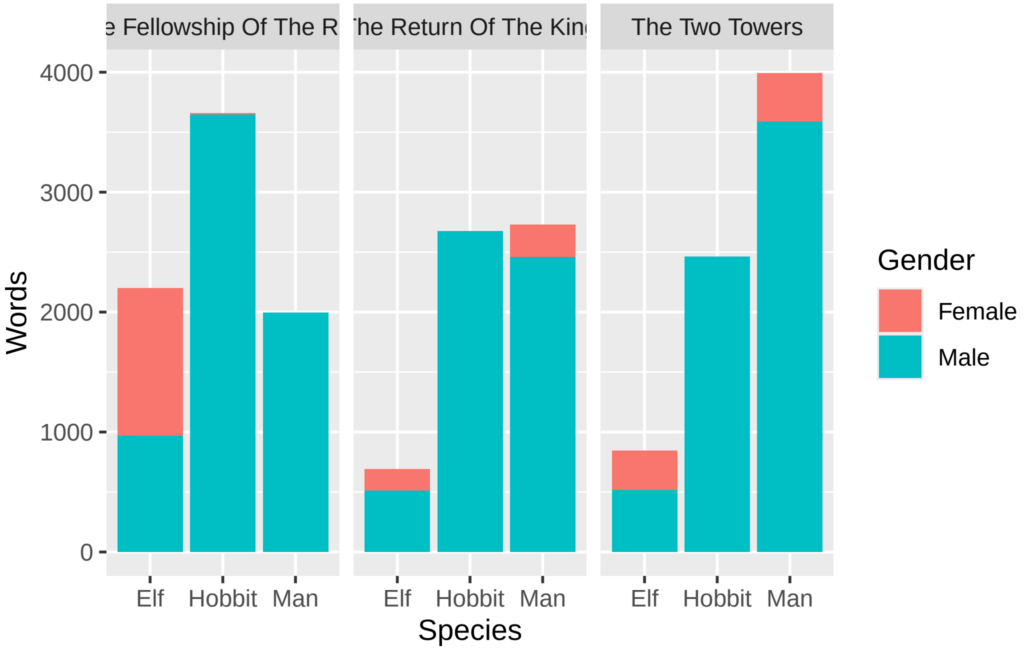
```
ggplot(summary_data,
       aes(x = Species, y = Words)) +
  geom_col()
```



11. Create a plot that visualizes the number of words spoken by species, gender, and film simultaneously. Use the complete tidy `lotr` data frame. You don't need to create a new summarized dataset (with `group_by(Species, Gender, Film)`) because the original data already has a row for each of those (you could make a summarized dataset, but it would be identical to the full version).

You need to show `Species`, `Gender`, and `Film` at the same time, but you only have two possible aesthetics (`x` and `fill`), so you'll also need to facet by the third. Play around with different combinations (e.g. try `x = Species`, then `x = Film`) until you find one that tells the clearest story. For fun, add a `labs()` layer to add a title and subtitle and caption.

```
ggplot(lotr,
       aes(x = Species, y = Words, fill = Gender)) +
  geom_col() +
  facet_wrap(vars(Film))
```


e Fellowship Of The R
'he Return Of The Kin
The Two Towers
Words
4000
3000
2000
1000
0
Elf
Hobbit
Man
Elf
Hobbit
Man
Elf
Hobbit
Man
Species
Gender
Female
Male