1.17. Exercises

1.17.1. Exercise 1: Familiarizing Yourself with Base Plotting

Type the following commands into a script and then execute them together:

```
plot(x = 1, y = 1, type = 'n',
     xlim = c(-2, 2), ylim = c(-2, 2))
points(x = -1, y = 1)
segments(x0 = -0.5, y0 = -1, x1 = 0.5, y1 = -1)
```

The first line opens up an empty plot with a point at the coordinates `x = 1` and `y = 1`. The `type = 'n'` argument means that this point is not actually displayed. `xlim` and `ylim` specify the plot margins.

The `points()` function plots a single point at the specified coordinates. The `segments()` function plots a line segment. The beginning of the line segment is given by the arguments `x0` and `y0`. The end points are given by `x1` and `y1`.

What is displayed in your plotting window is actually a one-eyed smiley. By adding additional `points()` and `segments()`, can you create a more elaborate smiley? This exercise will help to wrap your head around working with coordinate systems.

1.17.2. Exercise 2: Swirl

The interactive `swirl` package teaches you R inside R.

```
install.packages('swirl')
library(swirl)
swirl()
```

Complete the first four modules of the R programming course. If you have extra time, complete the first five courses of the Exploratory Data Analysis course. Throughout your R journey, you can come back to `swirl` at any time and complete more courses.

1.17.3. Exercise 3: Spot-the-Error #1

Type the following two lines of code into your R script, exactly as they are printed here on the page. Then execute them. This will result in two error messages.

```
x_values <- c(1, 2 3, 4, 5, 6, 7, 8, 9)
mean_x <- mean(X_values)
```

Each line contains one error. Can you find them and correct them?[11]

11 I thank Márton Sóskuthy for this exercise idea.

1.17.4. Exercise 4: Spot-the-Error #2

Why does the following command return an `NA` value?

```
x <- c(2, 3, 4, '4')
mean(x)
```

Can you use the function `as.numeric()` to solve this problem?

1.17.5. Exercise 5: Spot-the-Error: #3

The following line of code tries to extract the row from the `nettle` data frame that contains information on the country Yemen. Why does this return an error and can you fix this?

```
nettle[nettle$Country = 'Yemen',  ]
```

1.17.6. Exercise 6: Indexing Data Frames

Gillespie and Lovelace (2017: 4) say that "R is notorious for allowing users to solve problems in many ways". In this section, you will learn a bunch of different ways of extracting information from the same data frame. Some of these ways are redundant, but knowing multiple paths to the same goal gives you flexibility in how to approach data analysis problems. This exercise also teaches you how indexing statements can be used recursively, stacked on top of each other.

```
head(nettle)  # display first 6 rows
```

```
     Country Population Area  MGS Langs
1    Algeria       4.41 6.38 6.60    18
2     Angola       4.01 6.10 6.22    42
3  Australia       4.24 6.89 6.00   234
4 Bangladesh       5.07 5.16 7.40    37
5      Benin       3.69 5.05 7.14    52
6    Bolivia       3.88 6.04 6.92    38
```

Next, the following statements extract information from this data frame. You haven't been taught all of these ways of indexing yet. However, try to understand what the corresponding code achieves and I'm sure you'll be able to figure it out.

Importantly, think about what is being extracted *first*, only *then* type in the command to see whether the output matches your expectations.

```
nettle[2, 5]
```