Web scraping and string cleaning

This script is heavily inspired by https://www.gastonsanchez.com/r4strings/cleaning.html

Data

```r
library(tidyverse) # Package `stringr` most important for strings
```

```
## -- 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 become error
```

```r
library(rvest) # For web scraping
```

```
##
## Attaching package: 'rvest'
##
## The following object is masked from 'package:readr':
##
##     guess_encoding
```

```r
wiki_jump <- 'https://en.wikipedia.org/wiki/Men%27s_long_jump_world_record_progression'

long_jump <- read_html(wiki_jump)
tbl <- long_jump |>
  html_element("table") |>   # use html_elements() for all tables
  html_table()
tbl |> head()
```

```
## # A tibble: 6 x 5
##   Mark                      Wind  Athlete                Place                Date
##   <chr>                     <chr> <chr>                  <chr>                <chr>
## 1 7.61 m (24 ft 11+1/2 in) ""    Peter O'Connor (IRE)   Dublin, Ireland      5 Au~
## 2 7.69 m (25 ft 2+3/4 in)  ""    Edward Gourdin (USA)   Cambridge, United ~ 23 J~
## 3 7.76 m (25 ft 5+1/2 in)  ""    Robert LeGendre (USA) Paris, France        7 Ju~
## 4 7.89 m (25 ft 10+1/2 in) ""    DeHart Hubbard (USA)   Chicago, United St~ 13 J~
## 5 7.90 m (25 ft 11 in)     ""    Edward Hamm (USA)      Cambridge, United ~ 7 Ju~
## 6 7.93 m (26 ft 0 in)      "0.0" Sylvio Cator (HAI)     Paris, France        9 Se~
```

Finding the mark in meters

```r
marks <- tbl |> pull(Mark)
m1 <- marks[1]
m1
```

```
## [1] "7.61 m (24 ft 11+1/2 in)"
```

Using substring

```
str_sub(m1, 1, 4)
```

```
## [1] "7.61"
```

```
str_sub(m1, 1, 4) |> as.numeric()
```

```
## [1] 7.61
```

Using string detection/extraction (regular expression)

```
str_detect(m1, pattern = "[0-9]\\.[0-9][0-9]")
```

```
## [1] TRUE
```

```
str_extract(m1, pattern = "[0-9]\\.[0-9][0-9]")
```

```
## [1] "7.61"
```

Applying the method to the entire table

```
tbl2 <- tbl |>
  mutate(meters_sub = str_sub(Mark, 1, 4) |> as.numeric(),
         meters_ext = str_extract(Mark, pattern = "[0-9]\\.[0-9][0-9]"))
tbl2 |> select(starts_with("m"))
```

```
## # A tibble: 19 x 3
##    Mark                             meters_sub meters_ext
##    <chr>                                 <dbl> <chr>
##  1 7.61 m (24 ft 11+1/2 in)              7.61 7.61
##  2 7.69 m (25 ft 2+3/4 in)               7.69 7.69
##  3 7.76 m (25 ft 5+1/2 in)               7.76 7.76
##  4 7.89 m (25 ft 10+1/2 in)              7.89 7.89
##  5 7.90 m (25 ft 11 in)                  7.9  7.90
##  6 7.93 m (26 ft 0 in)                   7.93 7.93
##  7 7.98 m (26 ft 2 in)                   7.98 7.98
##  8 8.13 m (26 ft 8 in)                   8.13 8.13
##  9 8.21 m (26 ft 11 in)                  8.21 8.21
## 10 8.24 m (27 ft 1/4 in)                 8.24 8.24
## 11 8.28 m (27 ft 1+3/4 in)               8.28 8.28
## 12 8.31 m (27 ft 3 in) A                 8.31 8.31
## 13 8.33 m (27 ft 3+3/4 in)[2]            8.33 8.33
## 14 8.31 m (27 ft 3 in)                   8.31 8.31
## 15 8.34 m (27 ft 4+1/4 in)               8.34 8.34
## 16 8.35 m (27 ft 4+1/2 in)[5]            8.35 8.35
## 17 8.35 m (27 ft 4+1/2 in) A             8.35 8.35
## 18 8.90 m (29 ft 2+1/4 in) A             8.9  8.90
## 19 8.95 m (29 ft 4+1/4 in)               8.95 8.95
```

Making a new variable based on cases:

```
tbl3 <- tbl2 |>
  mutate(length_class  = case_when(
    meters_sub < 8 ~ "short",
    meters_sub > 8.5 ~ "long",
    TRUE ~ "Medium"
  ))
tbl3 |> select(meters_sub, length_class)
```

```
## # A tibble: 19 x 2
##    meters_sub length_class
##         <dbl> <chr>
##  1       7.61 short
##  2       7.69 short
##  3       7.76 short
##  4       7.89 short
##  5       7.9  short
##  6       7.93 short
##  7       7.98 short
##  8       8.13 Medium
##  9       8.21 Medium
## 10       8.24 Medium
## 11       8.28 Medium
## 12       8.31 Medium
## 13       8.33 Medium
## 14       8.31 Medium
## 15       8.34 Medium
## 16       8.35 Medium
## 17       8.35 Medium
## 18       8.9  long
## 19       8.95 long
```

Same, but from string rather than numeric

```
tbl4 <- tbl |>
  mutate(country  = case_when(
    str_detect(Athlete, "USA") ~ "US",
    str_detect(Athlete, "URS") ~ "USSR",
    TRUE ~ "Other"
  ))
tbl4 |> select(Athlete, country)
```

```
## # A tibble: 19 x 2
##    Athlete                   country
##    <chr>                     <chr>
##  1 Peter O'Connor (IRE)      Other
##  2 Edward Gourdin (USA)      US
##  3 Robert LeGendre (USA)     US
##  4 DeHart Hubbard (USA)      US
##  5 Edward Hamm (USA)         US
##  6 Sylvio Cator (HAI)        Other
##  7 Chuhei Nambu (JPN)        Other
##  8 Jesse Owens (USA)         US
##  9 Ralph Boston (USA)        US
## 10 Ralph Boston (USA)        US
## 11 Ralph Boston (USA)        US
## 12 Igor Ter-Ovanesyan (URS) USSR
## 13 Phil Shinnick (USA)       US
## 14 Ralph Boston (USA)        US
## 15 Ralph Boston (USA)        US
## 16 Ralph Boston (USA)        US
## 17 Igor Ter-Ovanesyan (URS) USSR
## 18 Bob Beamon (USA)          US
## 19 Mike Powell (USA)         US
```

Exercise:

Make a data.frame with the total, land and water area of each country in the world in **square miles** based on: https://en.wikipedia.org/wiki/List_of_countries_and_dependencies_by_area

Hints:

- Use `html_elements()` (notice the `s` at the end) to extract all tables.
- Use `str_extract()` and be aware that `(` and `)` need to be escaped by `\\` in search pattern.
- Convert the numbers to numeric values in R (you may need to remove `,` first).