Practical 2

Jumping Rivers

First we must load the **tidyverse**

```
library("tidyverse")
```

Question 1 - Cocktails!

We've put together a small list containing the ingredients of some classic cocktails

```
data(cocktails, package = "jrTidyverse2")
```

a) How many cocktails are in the list?

b) Create a tibble called `drinks`, where one column contains the name of the cocktail, and the other column contains the vector of ingredients

c) Create a new column that contains the number of ingredients in each column using `mutate()` and **purrr**

d) We're off out! Tonight we're particularly thirsty for a cocktail with rum in it. Filter `drinks` such that it only has cocktails containing rum

Question 2 - Beer !

So, we're at the pub with 8 mates and it's your round. In total you've been tasked with ordering 4 ales, 3 ipas, 1 stout plus an ale for yourself! We can load a data set of all of the ale, ipa and stouts that the pub sells from the course package

We're going to randomly select each persons drink using **purrr**. If people had asked for an even number of ales, ipas and stouts we could have done this without **purrr** like so

a) Nest the data according to the drink `Type` and save it as `pub`.

```
## Warning: All elements of `...` must be named.
## Did you want `data = c(URL, ABV, Color)`?
```

b) Create a column called `n` that contains the total number of each drink `Type` you need to order

c) Create a new column called order that contains the randomly sampled drinks you are going to order. You should be using `map2()` to parallel map over the columns `data` and `n`. You should also be using `sample_n()` to perform the sampling.

d) To see the drinks, select only the `Type` and `order` column, then `unnest()`

```
## Warning: `cols` is now required.
## Please use `cols = c(order)`
```

Question 3 - Happiness

You may remember the happiness data we used for practical 1 was recorded over 3 years; 2015, 2016 and 2017. For this question I've turned the happiness list in 3 tibbles, with each one representing the year. Running the following code will copy each file into your current working directory as a `.csv` file

```
library("jrTidyverse2")
get_happiness()
## Files have been copied successfully!
##           Check your current working directory.
```

a) Using a combination of **purrr** and the `unnest()` function from **tidyr**, read in and combine the 3 data sets. Don't delete the column containing the file name!

b) The data within the csv files doesn't contain the year. Fortunately the file name does! Use the column containing the filename to create a column called `Year`. Have a look at the `str_remove()` or `parse_number()` functions from **stringr** and **readr** respectively

c) Pick 3 countries and plot their happiness rank over time.

d) Every country in the data set has requested that they have their data sent to them individually. Make use of **purrr**'s parallel mapping functions and `nest()` to write the data to `.csv` files