Homework 1

Problem 1

Moments in statistics are quantitative measures that describe the specific characteristics of a probability distribution. It helps to understand the data set's shape, spread, and central tendency. For example:

- The first *raw moment* is the mean of a distribution. For a random variable, this is the *expected value*. It can be positive, negative, or zero.
- The second *central moment* is the variance. The variance is never negative.

The name 'moment' comes from the mathematical equivalence to the moment in classical mechanics.

The general formula for the k -th raw moment of a random sample of n observations $X_1, X_2, \ldots, X_n$ (or a numerical vector) is

$$\mu'_k = \frac{1}{n}\sum_{i=1}^{n} X_i^k$$

In particular, the first raw moment is the sample mean $\mu'_1 = \bar{X}$, the second raw moment is the mean of the squared sample values $\mu'_2 = \bar{X^2}$, etc.

1. Implement a function factory `sample_raw_moment_factory()` that takes a positive integer argument `k` and returns a function that takes as an argument a numeric vector and calculates the `k`-th sample raw moment. Provide it with an optional Boolean argument `na.rm` (similar to the one for the `mean()` function) that omits `NA` values from the calculation if any are present.
2. Use the function factory to create the function `raw_moment_2()` that takes as an argument a numerical vector and returns its second raw sample moment.
3. Check the documentation of the function `purrr::map_if()` . Use it to create a function operator `calculate_numeric_column_stats()` that takes as an argument a function and returns a function that takes as an argument a dataframe and then applies the function to the numerical columns of a the dataframe, and returns a list of the values.
4. Use the function operator `calculate_numeric_column_stats()` and the function `raw_moment_2()` to create a function `calculate_raw_moment_2()` that takes as an argument a dataframe and returns a list of the second raw sample moments of its numeric columns.
5. Use the function `calculate_raw_moment_2()` on the dataframe `mtcars`.

Problem 2

Implement a `Shape` S4 class that:

1. Has the slots:
   - `name` - a character type object, to represent the name of the shape

- `side_lengths` - a numerical vector of positive numbers to represent the lengths of the sides of the shape

2. A prototype list of default values for the slots;
3. A validator that checks if the `name` slot is not the empty string and that all elements of `side_lengths` are positive numbers;
4. A helper constructor `Shape()` that also casts the `name` argument to character before creating a new object;
5. Implements a `show` generic and method for the class with a suitable representation of the object to be printed out;
6. Implements getter and setter methods for the slots (e.g. `name()` and `name<-()`). The setters should also use the validator in order to make sure that only correct values can be assigned to the slots;
7. Implement the generic `is_rhombus` and a method for the `Shape` class that checks using the values in the `side_lengths` slot whether the object is a rhombus (e.g. whether the shape consists of 4 sides and they're all of equal length). The method should return `TRUE` if the shape is a rhombus and `FALSE` otherwise.