DATA202/STAT483 Assignment 4

Q1. (14 Marks)

Answer the following questions using the `motor_vehicle_reduced.csv` dataset. You will need the `ggplot` and `ggthemes` libraries. You can use either the `dplyr` , `tibble` , `tidyr` libraries **OR** the base R functions (you don't have to show both methods, just pick one).

Read the file into R. If you are going to use `dplyr` , also convert the data frame to a tibble.

a. Find the number of vehicles in registered in TLA Wellington *City* that are used as private passenger vehicles. [3 marks]
b. Find out how many white or silver vehicles have been imported from Japan and registered in New Zealand in 2000 or later. [3 marks]
c. Produce a table summarising all the different Volkswagen models in the dataset with non-zero gross vehicle mass. Give the model name, mean gross vehicle mass and the earliest and latest vehicle year for each model. [5 marks]
d. Produce a contingency table giving the number of vehicles for every combination of make and import status. Restrict the table to the 10 makes with the most new vehicles, and show all import statuses for those makes. [3 marks]

Q2. (17 Marks)

Answer the following questions using the `food_prices_yearmonth.csv` dataset.

Read the file into R.

a. Reduce the dataset to only items measured in kg, and check this by displaying a list of the first 10 unique item names in the reduced dataset. [2 marks]
b. Make a new data frame/tibble containing only the January values. [1 mark]
c. Create a table that gives the mean price of all the kg-valued items for each year in January, **display the table**, and keep that table as a new tibble/data frame. You don't need to convert the prices into dollar prices, just display them as numbers. [2 marks]
d. Use the table you just created to produce a time-series line plot of mean price by year, using the `ggplot2` package. Make sure to label the plot axes correctly. [3 marks]
e. Go back to the full kg-weighted items dataset for **all months**. Reduce it to just the 2018 data. [1 mark]
f. Select three of the kg-weighted items. Again using `ggplot2` , plot a bar chart showing prices for each month, and showing those three items side-by-side in different colours. Label the y-axis as dollar price.

   Make sure the months are in the correct order – you may need to set Month to a factor with the right order of levels. Also make sure the month labels are not displayed overlapping. [8 marks]

Q3. (9 Marks) Copy the following code into your RMarkdown answers document. It defines a function, and then runs that function.

Run the code – it should stop with an error. Use the `browser()` command or the R option `options(error=recover)` to debug the function and find the three mistakes in it. The mistakes may include code that is incorrect, code that needs to be removed or code that is missing.

If the ‘Stop’ button doesn’t close browser mode, go to the console and use the `Esc` key to escape from browser mode.

For each of the three mistakes, either change the code to fix the bug or write a comment in your RMarkdown explaining where you think the bug is. Each mistake is worth 2 marks for finding it and 1 mark for fixing it, and partial credit will be given for incomplete answers.

```
library(tidyr)
library(dplyr)
library(tibble)
vehicles <- as_tibble(read.csv("motor_vehicle_reduced.csv"))

summarise_vehicles <- function(region, type, max_axles, earliest_year = min(vehicles$VEHICLE_YEAR)) {
    if (!(region %in% unique(vehicles$TLA))) stop(paste(region, "is not in the list of TLAs (regions) in the dataset."))
    if (!(type %in% unique(vehicles$VEHICLE_TYPE))) stop(paste(type, "is not in the list of vehicle types in the dataset."))

    vehicles_sub <- filter(vehicles, TLA==region & VEHICLE_TYPE==type &
                               NUMBER_OF_AXLES <= max_axles &
                               VEHICLE_YEAR >= earliest_year &
                               NUMBER_OF_DOORS > 3) %>%
                    mutate(VEHICLE_DECADE = floor(VEHICLE_YEAR/10)*10)

    vehicles_sub <- filter(vehicles_sub, GROSS_VEHICLE_MASS < 0)

    vehicles_sub <- group_by(vehicles_sub, MAKE, VEHICLE_DECADE) %>%
                    arrange(VEHICLE_decade) %>%
                    select(BASIC_COLOUR, BODY_TYPE, MODEL, MAKE,
                           VEHICLE_DECADE, CC_RATING, GROSS_VEHICLE_MASS)

    result <- summarise(vehicles_sub, N=n(), Mean_CC_Rating=mean(CC_RATING))

    filter(result, N > 10)
}

summarise_vehicles("AUCKLAND","PASSENGER CAR/VAN",2,2000)
```

(Assignment total: 40 Marks)