Homework 3

Mariia Nikitash
Prepare your answers as a **single PDF file**.
Group work: You may work in groups of 1-3. Include all group member names in the PDF file. You may work with students in both sections (375-01, -02). Only one person in the group should submit to Canvas.
Due: check on Canvas.

1. Load the `nycflights13` library (will have to install the `nycflights13` package first) which contains flight arrival and departure data in a table called `flights`. Apply the tidyverse's data wrangling verbs to answer these questions. For each question, **give only the code (as one data pipeline with multiple "verbs" one after the other) beginning with `flights %>%` ….**

1. List data only for flights that departed on March 12, 2013.
   flights %>% filter(year == 2013, month == 3, day == 12) %>% View()
2. List data only for flights that were delayed (**both** arrival and departure) by more than 2 hours.
   flights %>% filter(arr_time > 2, dep_time > 2) %>% View()
3. List data only for flights that were delayed (**either** arrival or departure) by more than 2 hours.
   flights %>% filter(arr_time > 2 | dep_time > 2) %>% View()
4. List data only for flights that were operated by United, American, or Delta.
   flights %>% filter(carrier %in% c("UA", "DL", "AA"))
5. Sort data in order of fastest flights (`air_time`).
   flights %>% arrange(air_time) %>% View()
6. Sort data in order of longest duration flights (`air_time`).
   flights %>% arrange(-air_time) %>% View()
7. Show only the origin and destination of flights sorted by longest flights.
   flights %>% arrange(air_time) %>% select(origin, dest) %>% View()
8. Add a new variable that indicates the total delay (both departure and arrival delay).
   flights %>% mutate(total_delay = (dep_delay+arr_delay)) %>% View()
9. Show only the origin and destination of flights sorted by descending order of total delay.
   flights %>% mutate(total_delay = (dep_delay+arr_delay)) %>% arrange(-total_delay) %>% select(origin, dest) %>% View()
10. Show the average total delay for all flights.
    flights %>% mutate(total_delay = (dep_delay+arr_delay)) %>% summarize(mean(total_delay, na.rm=TRUE)) %>% View()
    Or
    flights %>% mutate(total_delay = (dep_delay+arr_delay)) %>% filter(!is.na(total_delay)) %>% summarize(mean(total_delay)) %>% View()
11. Show the average total delay for every departure city.

flights %>% mutate(total_delay = (dep_delay+arr_delay)) %>% filter(!is.na(total_delay)) %>% group_by(dest) %>% summarize(mean(total_delay)) %>% View()

12. Show the average total delay for every departure-arrival city pair.
    flights %>% mutate(total_delay = (dep_delay+arr_delay)) %>% filter(!is.na(total_delay)) %>% group_by(dest, origin) %>% summarize(mean(total_delay)) %>% View()

2. The following questions use the data collected from the anonymous survey given at the beginning of the course and from previous semesters. The dataset can be downloaded from the Datasets module on Canvas. Load the survey data into a variable called "survey". Use the function **`read_csv()`** that is part of the tidyverse library (instead of the built-in function `read.csv()`)

```
# install tidyverse
library(tidyverse)
survey <- read_csv("surveydataSpring2024.csv")
```

Apply the tidyverse's data wrangling verbs to answer these questions. For each question, **give only the code (as one data pipeline with multiple "verbs" one after the other) beginning with `survey %>%` ….**

a. Show only rows where Statistics skill is 10
   survey %>% select(Statistics) %>% filter(Statistics == 10) %>% View()
b. Show only rows where both Statistics and Math skills are 10
   survey %>% select(Statistics, Math) %>% filter(Statistics == 10, Math == 10) %>% View()
c. Show only the Semester, Communication, and Visualization columns where both Statistics and Math skills are 10
   survey %>% filter(Statistics == 10, Math == 10) %>% select(Semester, Communication, Visualization) %>% View()
d. Show only the Semester, Communication, and Visualization columns in decreasing order of Statistics and then Math skills
   survey %>% arrange(Statistics, Math) %>% select(Semester, Communication, Visualization) %>% View()
e. Show only rows where Statistics skill is missing
   survey %>% filter(is.na(Statistics)) %>% View()
f. Show for every student, only their Year, Semester, Math skill, Statistics skill, and the maximum of their Math and Statistics skills (Hint: mutate using **`pmax()`**)
   survey %>% select(Year, Semester, Math, Statistics) %>% mutate(pmax(Math, Statistics)) %>% View()
g. Show the median value of Computer Science skills
   survey %>% filter(!is.na(ComputerScience)) %>% summarise(median(ComputerScience)) %>% View()
h. Show the median value of Computer Science skills in Spring 2024

```
survey %>% filter(Semester== "Spring", Year== 2024) %>%
filter(!is.na(ComputerScience)) %>% summarise(median(ComputerScience)) %>%
View()
```

i. Show the median values of (each of) Computer Science, Math, and Statistics skills in every year

```
survey %>% group_by(Year) %>% summarise(median(ComputerScience,
na.rm=TRUE), median(Math, na.rm=TRUE), median(Statistics, na.rm=TRUE)) %>%
View()
```

j. Show for every year, the median values of (each of) Computer Science, Math, and Statistics skills of students who have taken CPSC483 (TakenCPSC483 is Yes)

```
survey %>% group_by(Year) %>% filter(TakenCPSC483 == "Yes") %>%
summarise(median(ComputerScience), median(Math), median(Statistics)) %>% View()
```

k. Show the median values of (each of) Computer Science, Math, and Statistics skills for groups of students who have taken and not taken CPSC483 (Hint: group_by)

```
survey %>% filter(!is.na(TakenCPSC483), !is.na(ComputerScience)) %>%
group_by(TakenCPSC483) %>% summarise(median(ComputerScience), median(Math,
na.rm=TRUE), median(Statistics, na.rm=TRUE)) %>% View()
```

l. Show the number of students who took the survey in every semester and year

```
survey %>% group_by(Year, Semester) %>% summarise(n()) %>% View()
```