Instructions

- Clear the environment
- Open a new R Script where you will do the exercise and later save in the project directory.
- Add the purpose of the file and the author
- Here are the main activities
  1) Load the `rio`, `lubridate` and `tidyverse` package
  2) Load the data using the `import`
  3) Select variables
  4) Rename variables with spaces
  5) Filter the values of interest
  6) Mutate variables
  7) Mutate with if_else

Part 1 (*You* :-) will do this together)

1.1

- Load the `rio`, `lubridate` and `tidyverse` package
- Use the `hospital_df` and `location_df` object names; Load the `line_hospital_data.csv` and the `line_hospitals_locations.xlsx` data

```
# General format
hospital_df <- import("Data/line_hospital_data.csv")
```

a) What is the structure of the data
b) How many observations are in the data?
c) What is the class of 'case_id

1.2 For this part we use the hospital_df data

- Create a subset called `sub1` of lab measurements, case_id and age data
- How many columns do you have now? add that as a comment in your code
- Hint: use `select` function

1.3 Rename variables with spaces

- Rename all the variables with a space and replace space with _
- Rename `date_of_outcome` to `date_outcome`

```
rename(.data, ...)
```

1.4 Filter data

- Create a data frame of all the participants who died: Hint use the `outcome` variable
- glimpse the data of the participant that died , how many rows and column, add that as a comment in your code
- create a data set of males over 25 years
-

```
## example of female data set
female <- filter(hosp_data,gender=="f" )
```

- Use `filter` and `select` to show case_id , hosp_date and date_onset of patients who recovered.
- How many rows and columns, add that as a comment in your code

- Create a data of participants who had `cough` and `chills` , then select only case_id,gender and age. Hint: try and use the pipe operator
- Filter participants that had `cough` AND `chills` OR `aches` OR their ct_blood IS GREATER than 20

1.5 mutate

- Create a BMI variable where BMI= weight_kg/height_meters * height_meters
  - Hint: you have to create height in meters variables
- Create a variable short is `YES` if ht_cm < 80 otherwise `NO`
- Hint: use IF_ELSE
- Create age group of 10 year gaps, i.e `0-10,10-20,20-30,30-40,40-50,Over_50`
- HINT use CASE_WHEN

```
## Hint of CASE WHEN
hosp_data <- hosp_data %>%
  mutate(paed =ifelse(age>14,"Adult","Paed"),
                      age_group=case_when(
                        age<14 ~"Paed",
                        age >13 & age< 19 ~"Gen Z",
                        age>18 & age<30 ~ "Non Gen Z",
                        .default = "other"))
```

1.6 Merging

- Left join the `hospital_df` and `location_df` using case_id as the ID call the data `merge1`
- Inner join the `hospital_df` and `location_df` using case_id as the ID call the data `merge2`
- Full join the `hospital_df` and `location_df` using case_id as the ID call the data `merge3`
- Look at the glimpse of the 3 data sets above, what is the difference
- Export all the 3 files as `.xlsx` into the data folder]]

```
hosp_data <- import("Data/line_hospital.xlsx")
hospital_location <- import('data/line_hospital_sub.csv')
hosp_left_joined <- left_join(
  x = hosp_data,
  y = hospital_location,
  by = "case_id"
)
```

1.8 Create a pipe

Create a pipe chain which selects the `male` gender data

1. renames the variables with space
2. creates BMI
3. creates age group of 10
4. age of greater than 18

1.7 Extras

Run the following and create the year and month from the hospital date variable

```
hosp_data <- hosp_data %>%
  mutate(date_onset=as_date(date_onset))
```

Hint to create year and month

```
mutate(hosp_year = year(date_onset) ,
       hosp_month = month(date_onset))
```

Use the hosp_data$year to check the length and report