```r
# DTA250
# Spring 2024

# Iterations ----

# Load the tidyverse package
library(tidyverse)

# Set seed for reproducibility
set.seed(123)

# TODO
# Run the following code to create the data frame
df <- tibble(
  a = rnorm(10),
  b = rnorm(10),
  c = rnorm(10),
  d = rnorm(10)
)

#TODO
# Use the summarize() function to calculate the number of rows and the median
# for each column in the data frame.
# use n() to calculate the number of rows and median() to calculate the median

df |>
  summarize(
    n = n(),
    median_a = median(a),
    median_b = median(b),
    median_c = median(c),
    median_d = median(d)
  )


# You can also use the across() function to apply the same function to multiple

#TODO
# Use the across() function to calculate the median for each column in the data
# frame.
# You need to use everything() function to specify all the columns
# You still need to use n() to calculate the number of rows

df |>
  summarize(
    n = n(),
    across(everything(), median)
  )


# TODO
# Run the following code to create a new version of the data frame
df <- tibble(
  grp = sample(2, 10, replace=TRUE),
  a = rnorm(10),
  b = rnorm(10),
  c = rnorm(10),
  d = rnorm(10)
)

# Notice there is a new column called "grp" in the data frame
# This column has two unique values (1 and 2)
# We don't need to find the median for this column.
# We will use the group_by() function to group the data by the "grp" column
# Then we will use the summarize() function to calculate the median for each
# column in the data frame

#TODO
# Use the group_by() function to group the data by the "grp" column
# Then use the summarize() function to calculate the median for each column in
# the data frame

df |>
  group_by(grp) |>
  summarize(
    n = n(),
    across(everything(), median)
  )

# TODO
# Run the following code to create a new version of the data frame
df <- tibble(
  grp = sample(2, 10, replace=TRUE),
  a = rnorm(10),
  b = rnorm(10),
  c = rnorm(10),
  d = rnorm(10),
  e = c("a", "b", "c", "d", "e", "f", "g", "h", "i", "j"),
  f = c("j", "k", "l", "m", "n", "o", "p", "q", "r", "s"),
  g = c(T, F, T, F, T, F, T, F, T, F),
  h = c(F, T, F, T, F, T, F, T, F, T)
)

# You can use the where() function to select columns based on their data type
# For example, you can use the where() function to select only the numeric
# columns in the data frame

#TODO
# Use the where() function to select only the numeric columns in the data frame
# Then use the group_by() function to group the data by the "grp" column
# Then use the summarize() function to calculate the median for each numerical
# column in the data frame

df |>
  select(where(is.numeric)) |>
  group_by(grp) |>
  summarize(
    n = n(),
    across(everything(), median)
  )

# TODO
# Use the where() function to select only the character columns in the data
# frame

df |>
  select(where(is.character))

# TODO
# Use the where() function to select only the logical columns in the data frame

df |>
  select(where(is.logical))
```

```
# TODO
# Run the following code to create this function

rnorm_na <- function(n, n_na, mean = 0, sd = 1) {
  sample(c(rnorm(n - n_na, mean = mean, sd = sd), rep(NA, n_na)))
}

# The above function will create random numbers with missing values in them.

# TODO
# Run the following code to create a new data frame

df_miss <- tibble(
  a = rnorm_na(5, 1),
  b = rnorm_na(5, 1),
  c = rnorm_na(5, 2),
  d = rnorm(5)
)

# TODO
# Try and calculate the mean of each column in the data frame above

df_miss |>
  summarize(
    n = n(),
    across(everything(), mean)
  )

# Notice the NA results.

# We will use an advanced technique to calculate the mean of each column in the
# data frame above.
# We will use the na.rm argument in the mean() function to remove the NA values
# You have done this before. However, we will use the across() function to
# apply the mean() function to all the columns in the data frame
# We will use the function() function to specify the mean() function and the
# na.rm argument

#TODO
# Use the across() function to calculate the mean for each column in the data
# frame above. Use the function() function to specify the mean() function and
```

```r
# the na.rm argument

df_miss |>
  summarize(
    n = n(),
    across(everything(), function(x) mean(x, na.rm = TRUE))
  )

# One last thing, you can apply multiple function to the columns in the data
# frame using the list() function

# TODO
# Use the list() function to calculate the mean and median for each column in
# the data frame above

df_miss |>
  summarize(
    across(a:d, list(
      mean = \(x) mean(x, na.rm = TRUE),
      median = \(x) median(x, na.rm = TRUE)
    )),
    n = n()
  )

# OR
df_miss |>
  summarize(
    across(a:d, list(
      mean = function(x) mean(x, na.rm = TRUE),
      median = function(x) median(x, na.rm = TRUE)
    )),
    n = n()
  )
```