Experiment 3

Question 2

```
[3]: library(tidyverse)
```

```
Attaching core tidyverse packages
tidyverse 2.0.0
dplyr      1.1.4       readr      2.1.5
forcats    1.0.0       stringr    1.5.1
ggplot2    3.5.1       tibble     3.2.1
lubridate  1.9.4       tidyr      1.3.1
purrr      1.0.4
Conflicts

tidyverse_conflicts()
 dplyr::filter() masks stats::filter()
 dplyr::lag()    masks stats::lag()
 Use the conflicted package
(<http://conflicted.r-lib.org/>) to force all conflicts to
become errors
```

```
[4]: setwd("/home/asus/content/Notes/Semester 4/FDN Lab/Experiments/Experiment␣
      ↪3")
```

```
[5]: df_mean <- data.frame(
       ID = c(1, 2, 3, 4, 5, 6, 7, 8, 9, 10),
       Name = c("Alice", "Bob", NA, "David", "Emma", "Frank", NA, "Hannah",␣
      ↪"Ian", "Jack"),
       Age = c(25, NA, 30, 29, NA, 35, 40, NA, 50, 27),
       Salary = c(50000, 60000, 55000, NA, 70000, 75000, 80000, 65000, NA,␣
      ↪72000),
       Score = c(80, 90, NA, 85, 88, 92, NA, 77, 95, Inf)
     )
```

Identify missing data (is.na(df), sum(is.na(df))).

```
[6]: # i. Identify missing data
     print(is.na(df_mean))   # Identify missing values
     print(sum(is.na(df_mean)))   # Count total missing values
```

```
        ID  Name   Age Salary Score
[1,] FALSE FALSE FALSE  FALSE FALSE
[2,] FALSE FALSE  TRUE  FALSE FALSE
[3,] FALSE  TRUE FALSE  FALSE  TRUE
[4,] FALSE FALSE FALSE   TRUE FALSE
```

```
 [5,] FALSE FALSE  TRUE  FALSE FALSE
 [6,] FALSE FALSE FALSE  FALSE FALSE
 [7,] FALSE  TRUE FALSE  FALSE  TRUE
 [8,] FALSE FALSE  TRUE  FALSE FALSE
 [9,] FALSE FALSE FALSE   TRUE FALSE
[10,] FALSE FALSE FALSE  FALSE FALSE
[1] 9
```

Remove missing rows (na.omit(df))

```
[7]: df_mean_no_na <- na.omit(df_mean)
     print(df_mean_no_na)
```

```
    ID  Name Age Salary Score
1    1 Alice  25  50000    80
6    6 Frank  35  75000    92
10  10  Jack  27  72000   Inf
```

Replace NA with zero (df[is.na(df)] <- 0).

```
[8]: df_mean_zero <- df_mean
     df_mean_zero[is.na(df_mean_zero)] <- 0
     print(df_mean_zero)
```

```
    ID   Name Age Salary Score
1    1  Alice  25  50000    80
2    2    Bob   0  60000    90
3    3      0  30  55000     0
4    4  David  29      0    85
5    5   Emma   0  70000    88
6    6  Frank  35  75000    92
7    7      0  40  80000     0
8    8 Hannah   0  65000    77
9    9    Ian  50      0    95
10  10   Jack  27  72000   Inf
```

Replace NA with column mean (df$Age[is.na(df$Age)] <- mean(df$Age, na.rm=TRUE)).

```
[9]: df_mean_mean <- df_mean

     df_mean$Age[is.na(df_mean$Age)] <- mean(df_mean$Age, na.rm = TRUE)
     df_mean$Salary[is.na(df_mean$Salary)] <- mean(df_mean$Salary, na.rm =␣
      ↪TRUE)
     df_mean$Score[is.na(df_mean$Score)] <- mean(df_mean$Score, na.rm = TRUE)

     print(df_mean_mean)
```

```
      ID   Name Age Salary Score
1    1  Alice  25  50000    80
2    2    Bob  NA  60000    90
3    3   <NA>  30  55000    NA
4    4  David  29     NA    85
5    5   Emma  NA  70000    88
6    6  Frank  35  75000    92
7    7   <NA>  40  80000    NA
8    8 Hannah  NA  65000    77
9    9    Ian  50     NA    95
10  10   Jack  27  72000   Inf
```

Remove Inf and NaN (df$Score[is.infinite(df$Score) | is.nan(df$Score)] <- NA)

```
[10]: df_mean_clean <- df_mean
      df_mean_clean$Score[is.infinite(df_mean_clean$Score) | is.
       ↪nan(df_mean_clean$Score)] <- NA
      print(df_mean_clean)
```

```
      ID   Name      Age Salary Score
1    1  Alice 25.00000  50000    80
2    2    Bob 33.71429  60000    90
3    3   <NA> 30.00000  55000    NA
4    4  David 29.00000  65875    85
5    5   Emma 33.71429  70000    88
6    6  Frank 35.00000  75000    92
7    7   <NA> 40.00000  80000    NA
8    8 Hannah 33.71429  65000    77
9    9    Ian 50.00000  65875    95
10  10   Jack 27.00000  72000    NA
```

Use tidyverse's replace_na() for selective column handling.

```
[11]: df_mean_tidy <- df_mean %>%
        mutate(
          Age = replace_na(Age, mean(Age, na.rm = TRUE)),
          Salary = replace_na(Salary, median(Salary, na.rm = TRUE))
        )
      print(df_mean_tidy)
```

```
     ID   Name      Age Salary Score
1   1  Alice 25.00000  50000    80
2   2    Bob 33.71429  60000    90
3   3   <NA> 30.00000  55000   Inf
4   4  David 29.00000  65875    85
5   5   Emma 33.71429  70000    88
6   6  Frank 35.00000  75000    92
```

```
7    7    <NA> 40.00000  80000     Inf
8    8 Hannah 33.71429  65000      77
9    9     Ian 50.00000  65875      95
10  10    Jack 27.00000  72000     Inf
```

Drop columns with excessive missing data (df <- df[, colSums(is.na(df)) < nrow(df) * 0.5])

```
[12]: df_mean_filtered <- df_mean[, colSums(is.na(df_mean)) < (nrow(df_mean) *
      →0.5)]
      print(df_mean_filtered)
```

```
      ID   Name      Age Salary Score
1     1  Alice 25.00000  50000    80
2     2    Bob 33.71429  60000    90
3     3   <NA> 30.00000  55000   Inf
4     4  David 29.00000  65875    85
5     5   Emma 33.71429  70000    88
6     6  Frank 35.00000  75000    92
7     7   <NA> 40.00000  80000   Inf
8     8 Hannah 33.71429  65000    77
9     9    Ian 50.00000  65875    95
10   10   Jack 27.00000  72000   Inf
```

Fill missing categorical values with the mode.

```
[13]: # viii. Fill missing categorical values with mode
      fill_mode <- function(x) {
        if (is.character(x)) {
          mode_value <- names(sort(table(x), decreasing = TRUE))[1]
          x[is.na(x)] <- mode_value
        }
        return(x)
      }
      df_mean_mode <- df_mean
      df_mean_mode$Name <- fill_mode(df_mean_mode$Name)
      print(df_mean_mode)
```

```
      ID   Name      Age Salary Score
1     1  Alice 25.00000  50000    80
2     2    Bob 33.71429  60000    90
3     3  Alice 30.00000  55000   Inf
4     4  David 29.00000  65875    85
5     5   Emma 33.71429  70000    88
6     6  Frank 35.00000  75000    92
7     7  Alice 40.00000  80000   Inf
8     8 Hannah 33.71429  65000    77
9     9    Ian 50.00000  65875    95
```

```
10 10    Jack 27.00000   72000    Inf
```