NAME – SAARA ANAND

REG NO – 21BCE8156

SLOT – L55+L56

FDA LAB ASSIGNMENT 7

Practice commands on dpylr package.

Create your own data frame and apply the following functions:

Create Dataframe:

```
# Using data.frame
my_df <- data.frame(
  Name = c("Alice", "Tina", "Charlie", "Dolly", "Emma", "Polly", "Sam" ),
  Age = c(25, 30, 22, 28, 35, 45, 23),
  Salary = c(45000, 52000, 40000, 60000, 70000, 65000, 340000)
)
my_df
```

```
> # Using data.frame
> my_df <- data.frame(
+   Name = c("Alice", "Tina", "Charlie", "Dolly", "Emma", "Polly", "Sam" ),
+   Age = c(25, 30, 22, 28, 35, 45, 23),
+   Salary = c(45000, 52000, 40000, 60000, 70000, 65000, 340000)
+ )
> my_df
     Name Age Salary
1   Alice  25  45000
2    Tina  30  52000
3 Charlie  22  40000
4   Dolly  28  60000
5    Emma  35  70000
6   Polly  45  65000
7     Sam  23 340000
>
```

Select Function

```
library(dplyr)

# Using data.frame

my_df <- data.frame(

  Name = c("Alice", "Tina", "Charlie", "Dolly", "Emma", "Polly", "Sam" ),

  Age = c(25, 30, 22, 28, 35, 45, 23),

  Salary = c(45000, 52000, 40000, 60000, 70000, 65000, 340000)

)

my_df

selected_data <- select(my_df, Name, Age)

selected_data
```

```
> selected_data <- select(my_df, Name, Age)
> selected_data
     Name Age
1   Alice  25
2    Tina  30
3 Charlie  22
4   Dolly  28
5    Emma  35
6   Polly  45
7     Sam  23
> |
```

```
> selected_data <- select(my_df, Name, Salary)
> selected_data
     Name Salary
1   Alice  45000
2    Tina  52000
3 Charlie  40000
4   Dolly  60000
5    Emma  70000
6   Polly  65000
7     Sam 340000
> |
```

Removing duplicates using distinct() function

```
# Using data.frame
my_df <- data.frame(
  Name = c("Alice", "Tina", "Charlie", "Dolly", "Emma", "Polly", "Sam" ),
  Age = c(25, 30, 22, 28, 35, 45, 23),
  Salary = c(45000, 52000, 40000, 60000, 70000, 65000, 340000)
)
my_df
```

distinct(my_df)

```
> distinct(my_df)
     Name Age Salary
1   Alice  25  45000
2    Tina  30  52000
3 Charlie  22  40000
4   Dolly  28  60000
5    Emma  35  70000
6   Polly  45  65000
7     Sam  23 340000
>
```

Rename Function

library(dplyr)

Using data.frame

my_df <- data.frame(

Name = c("Alice", "Tina", "Charlie", "Dolly", "Emma", "Polly", "Sam"),

Age = c(25, 30, 22, 28, 35, 45, 23),

Salary = c(45000, 52000, 40000, 60000, 70000, 65000, 340000)

)

my_df

renamed_data <- rename(my_df, Full_Name = Name, Years = Age)

renamed_data

```
> renamed_data <- rename(my_df, Full_Name = Name, Years = Age)
> renamed_data
  Full_Name Years Salary
1     Alice    25  45000
2      Tina    30  52000
3   Charlie    22  40000
4     Dolly    28  60000
5      Emma    35  70000
6     Polly    45  65000
7       Sam    23 340000
>
```

```
> renamed_data <- rename(my_df, Full_Name = Name, Amount=Salary)
> renamed_data
  Full_Name Age Amount
1     Alice  25  45000
2      Tina  30  52000
3   Charlie  22  40000
4     Dolly  28  60000
5      Emma  35  70000
6     Polly  45  65000
7       Sam  23 340000
>
```

Filter Function

```
# Using data.frame

my_df <- data.frame(

  Name = c("Alice", "Tina", "Charlie", "Dolly", "Emma", "Polly", "Sam" ),

  Age = c(25, 30, 22, 28, 35, 45, 23),

  Salary = c(45000, 52000, 40000, 60000, 70000, 65000, 340000)

)

my_df

mydf = filter(my_df, Name == "Alice")
```

mydf

```
> mydf = filter(my_df, Name == "Alice")
> mydf
   Name Age Salary
1 Alice  25  45000
>
```

```
> mydf = filter(my_df, Age == "30")
> mydf
  Name Age Salary
1 Tina  30  52000
>
```

grepl function

Using data.frame

my_df <- data.frame(

Name = c("Alice", "Tina", "Charlie", "Dolly", "Emma", "Polly", "Sam"),

Age = c(25, 30, 22, 28, 35, 45, 23),

Salary = c(45000, 52000, 40000, 60000, 70000, 65000, 340000)

)

mydf = filter(my_df, grepl("A", Name))

mydf

```
> mydf = filter(my_df, grepl("A", Name))
> mydf
   Name Age Salary
1 Alice  25  45000
>
```

```
> mydf = filter(my_df, grepl("ol", Name))
> mydf
   Name Age Salary
1 Dolly  28  60000
2 Polly  45  65000
>
```

Summarise Function

```
library(dplyr)

# Using data.frame

my_df <- data.frame(

  Name = c("Alice", "Tina", "Charlie", "Dolly", "Emma", "Polly", "Sam" ),

  Age = c(25, 30, 22, 28, 35, 45, 23),

  Salary = c(45000, 52000, 40000, 60000, 70000, 65000, 340000)

)

summary_data <- summarise(my_df, Avg_Age = mean(Age), Total_Salary = sum(Salary))

summary_data
```

```
> summary_data <- summarise(my_df, Avg_Age = mean(Age), Total_Salary = sum(Salary))
> summary_data
   Avg_Age Total_Salary
1 29.71429       672000
>
```

```
> summary_data <- summarise(my_df, Avg_Age = mean(Age), Avg_Salary = mean(Salary))
> summary_data
   Avg_Age Avg_Salary
1 29.71429   52285.71
>
```

Arrange Function

```
library(dplyr)

# Using data.frame

my_df <- data.frame(

  Name = c("Alice", "Tina", "Charlie", "Dolly", "Emma", "Polly", "Sam" ),

  Age = c(25, 30, 22, 28, 35, 45, 23),

  Salary = c(45000, 52000, 40000, 60000, 70000, 65000, 34000)

)

my_df

sorted_data <- arrange(my_df, Age)

sorted_data
```

```
> sorted_data <- arrange(my_df, Age)
> sorted_data
     Name Age Salary
1 Charlie  22  40000
2     Sam  23  34000
3   Alice  25  45000
4   Dolly  28  60000
5    Tina  30  52000
6    Emma  35  70000
7   Polly  45  65000
>
```

```
> sorted_data <- arrange(my_df, Salary)
> sorted_data
     Name Age Salary
1     Sam  23  34000
2 Charlie  22  40000
3   Alice  25  45000
4    Tina  30  52000
5   Dolly  28  60000
6   Polly  45  65000
7    Emma  35  70000
>
```

Pipe Operator

```
# Using data.frame

my_df <- data.frame(

  Name = c("Alice", "Tina", "Charlie", "Dolly", "Emma", "Polly", "Sam" ),

  Age = c(25, 30, 22, 28, 35, 45, 23),

  Salary = c(45000, 52000, 40000, 60000, 70000, 65000, 340000)

)

dt = my_df %>% select(Name, Age) %>% sample_n(5)

dt
```

```
> dt = my_df %>% select(Name, Age) %>% sample_n(5)
> dt
     Name Age
1   Dolly  28
2   Polly  45
3     Sam  23
4    Emma  35
5 Charlie  22
>
```

```
> dt = my_df %>% select(Age) %>% sample_n(5)
> dt
  Age
1  30
2  28
3  35
4  23
5  45
> |
```

Group by Function

```
library(dplyr)

# Using data.frame

my_df <- data.frame(

  Name = c("Alice", "Tina", "Charlie", "Dolly", "Emma", "Polly", "Sam" ),

  Age = c(25, 30, 22, 28, 35, 45, 23),

  Salary = c(45000, 52000, 40000, 60000, 70000, 65000, 34000)

)

my_df

grouped_data <- group_by(my_df, Age_Group = ifelse(Age >= 25, "25+", "Under 25"))

grouped_data
```

```
> grouped_data <- group_by(my_df, Age_Group = ifelse(Age >= 25, "25+", "Under 25"))
> grouped_data
# A tibble: 7 x 4
# Groups:   Age_Group [2]
  Name      Age Salary Age_Group
  <chr>   <dbl>  <dbl> <chr>
1 Alice      25  45000 25+
2 Tina       30  52000 25+
3 Charlie    22  40000 Under 25
4 Dolly      28  60000 25+
5 Emma       35  70000 25+
6 Polly      45  65000 25+
7 Sam        23  34000 Under 25
>
```

```
> grouped_data <- group_by(my_df, Age_Group = ifelse(Age >= 30, "30+", "Under 30"))
> grouped_data
# A tibble: 7 x 4
# Groups:   Age_Group [2]
  Name      Age Salary Age_Group
  <chr>   <dbl>  <dbl> <chr>
1 Alice      25  45000 Under 30
2 Tina       30  52000 30+
3 Charlie    22  40000 Under 30
4 Dolly      28  60000 Under 30
5 Emma       35  70000 30+
6 Polly      45  65000 30+
7 Sam        23  34000 Under 30
>
```

Do Function

```
library(dplyr)

# Using data.frame

my_df <- data.frame(

  Name = c("Alice", "Tina", "Charlie", "Dolly", "Emma", "Polly", "Sam" ),

  Age = c(25, 30, 22, 28, 35, 45, 23),

  Salary = c(45000, 52000, 40000, 60000, 70000, 65000, 34000)

)
```

```
my_df
custom_summary <- function(x) {
  return(list(mean_age = mean(x$Age), total_salary = sum(x$Salary)))
}
grouped_data <- group_by(my_df, Age_Group = ifelse(Age >= 25, "25+", "Under 25"))
custom_summary_data <- do(grouped_data, custom_summary(.))
```

Slice Function

```
library(dplyr)
# Using data.frame
my_df <- data.frame(
  Name = c("Alice", "Tina", "Charlie", "Dolly", "Emma", "Polly", "Sam" ),
  Age = c(25, 30, 22, 28, 35, 45, 23),
  Salary = c(45000, 52000, 40000, 60000, 70000, 65000, 34000)
)
my_df
sliced_data <- slice(my_df, 1:2) # Extracts the first two rows
sliced_data
```

```
> sliced_data <- slice(my_df, 1:2) # Extracts the first two rows
> sliced_data
   Name Age Salary
1 Alice  25  45000
2  Tina  30  52000
>
```

```
> sliced_data <- slice(my_df, 3:5) # Extracts the first two rows
> sliced_data
     Name Age Salary
1 Charlie  22  40000
2   Dolly  28  60000
3    Emma  35  70000
>
```

Mutate function

```
library(dplyr)

# Using data.frame

my_df <- data.frame(

  Name = c("Alice", "Tina", "Charlie", "Dolly", "Emma", "Polly", "Sam" ),

  Age = c(25, 30, 22, 28, 35, 45, 23),

  Salary = c(45000, 52000, 40000, 60000, 70000, 65000, 34000)

)

my_df

mutated_data <- mutate(my_df, Age_In_10_Years = Age + 10)

mutated_data
```

```
> mutated_data <- mutate(my_df, Age_In_10_Years = Age + 10)
> mutated_data
     Name Age Salary Age_In_10_Years
1   Alice  25  45000               35
2    Tina  30  52000               40
3 Charlie  22  40000               32
4   Dolly  28  60000               38
5    Emma  35  70000               45
6   Polly  45  65000               55
7     Sam  23  34000               33
>
```

```
> mutated_data <- mutate(my_df, Age_In_10_Years = Age + 5)
> mutated_data
     Name Age Salary Age_In_10_Years
1   Alice  25  45000               30
2    Tina  30  52000               35
3 Charlie  22  40000               27
4   Dolly  28  60000               33
5    Emma  35  70000               40
6   Polly  45  65000               50
7     Sam  23  34000               28
>
```

Rank Function

```
library(dplyr)

my_df <- data.frame(

  Name = c("Alice", "Tina", "Charlie", "Dolly", "Emma", "Polly", "Sam" ),

  Age = c(25, 30, 22, 28, 35, 45, 23),

  Salary = c(45000, 52000, 40000, 60000, 70000, 65000, 340000)

)

my_df
```

```
df <- my_df %>%mutate(Age_Rank = rank(Age))
df
```

```
> df <- my_df %>%mutate(Age_Rank = rank(Age))
> df
      Name Age Salary Age_Rank
1    Alice  25  45000        3
2     Tina  30  52000        5
3  Charlie  22  40000        1
4    Dolly  28  60000        4
5     Emma  35  70000        6
6    Polly  45  65000        7
7      Sam  23 340000        2
>
```

Join Functions

```
# Create STUDENT DATA data frame
student_data <- data.frame(
  student_id = c(1, 2, 3, 4, 5),
  name = c("John", "Alice", "Bob", "Emily", "Michael"),
  age = c(18, 20, NA, 19, 21),
  grade = c("A", "B", "C", "B", "A")
)
# Create ORGANIZATION DATA data frame
organization_data <- data.frame(
  organization_id = c(1, 2, 3, 4, 5),
  name = c("Company A", "Company B", "Company C",
"Company D", "Company E"),
```

```
  industry = c("Tech", "Finance", "Healthcare", "Tech",
"Education"),
  revenue = c(1000000, 500000, NA, 2000000, 800000)
)

merged_data <- merge(student_data, organization_data, by
= "name", all = TRUE)

merged_data
```

	name	student_id	age	grade	organization_id	industry	revenue
1	Alice	2	20	B	NA	<NA>	NA
2	Bob	3	NA	C	NA	<NA>	NA
3	Company A	NA	NA	<NA>	1	Tech	1e+06
4	Company B	NA	NA	<NA>	2	Finance	5e+05
5	Company C	NA	NA	<NA>	3	Healthcare	NA
6	Company D	NA	NA	<NA>	4	Tech	2e+06
7	Company E	NA	NA	<NA>	5	Education	8e+05
8	Emily	4	19	B	NA	<NA>	NA
9	John	1	18	A	NA	<NA>	NA
10	Michael	5	21	A	NA	<NA>	NA

```
> |

# Perform Left Join

left_join <- merge(student_data, organization_data, by =
"name", all.x = TRUE)

# Perform Right Join

right_join <- merge(student_data, organization_data, by =
"name", all.y = TRUE)

# Perform Outer Join

outer_join <- merge(student_data, organization_data, by =
"name", all = TRUE)
```

left_join

right_join

outer_join

```
> left_join
     name student_id age grade organization_id industry revenue
1   Alice          2  20     B               NA     <NA>      NA
2     Bob          3  NA     C               NA     <NA>      NA
3   Emily          4  19     B               NA     <NA>      NA
4    John          1  18     A               NA     <NA>      NA
5 Michael          5  21     A               NA     <NA>      NA
> right_join
       name student_id age grade organization_id   industry revenue
1 Company A         NA  NA  <NA>               1       Tech   1e+06
2 Company B         NA  NA  <NA>               2    Finance   5e+05
3 Company C         NA  NA  <NA>               3 Healthcare      NA
4 Company D         NA  NA  <NA>               4       Tech   2e+06
5 Company E         NA  NA  <NA>               5  Education   8e+05
> outer_join
        name student_id age grade organization_id   industry revenue
1      Alice          2  20     B               NA       <NA>      NA
2        Bob          3  NA     C               NA       <NA>      NA
3  Company A         NA  NA  <NA>               1       Tech   1e+06
4  Company B         NA  NA  <NA>               2    Finance   5e+05
5  Company C         NA  NA  <NA>               3 Healthcare      NA
6  Company D         NA  NA  <NA>               4       Tech   2e+06
7  Company E         NA  NA  <NA>               5  Education   8e+05
8      Emily          4  19     B               NA       <NA>      NA
9       John          1  18     A               NA       <NA>      NA
10   Michael          5  21     A               NA       <NA>      NA
>
```

If else Function

Create STUDENT DATA data frame

df <- data.frame(

  student_id = c(1, 2, 3, 4, 5),

```
  name = c("John", "Alice", "Bob", "Emily", "Michael"),
  age = c(18, 20, 22, 19, 21),
  grade = c("A", "B", "C", "B", "A")
)
dfs <- ifelse(df$age >= 18, "Pass", "Fail")
print(dfs)
```

```
> dfs <- ifelse(df$age >= 18, "Pass", "Fail")
> print(dfs)
[1] "Pass" "Pass" "Pass" "Pass" "Pass"
>
```

```
> dfs <- ifelse(df$age >= 20, "Pass", "Fail")
> print(dfs)
[1] "Fail" "Pass" "Pass" "Fail" "Pass"
>
```

bind_rows() Function

```
# Create STUDENT DATA data frame
df1 <- data.frame(
  student_id = c(1, 2, 3, 4, 5),
  name = c("John", "Alice", "Bob", "Emily", "Michael"),
  age = c(18, 20, NA, 19, 21),
  grade = c("A", "B", "C", "B", "A")
)
# Create ORGANIZATION DATA data frame
```

```
df2 <- data.frame(
  organization_id = c(1, 2, 3, 4, 5),
  name = c("Company A", "Company B", "Company C", "Company D", "Company E"),
  industry = c("Tech", "Finance", "Healthcare", "Tech", "Education"),
  revenue = c(1000000, 500000, NA, 2000000, 800000)
)
combined_data <- bind_rows(df1, df2)
combined_data
```

```
> combined_data <- bind_rows(df1, df2)
> combined_data
   student_id      name age grade organization_id   industry revenue
1           1      John  18     A               NA       <NA>      NA
2           2     Alice  20     B               NA       <NA>      NA
3           3       Bob  NA     C               NA       <NA>      NA
4           4     Emily  19     B               NA       <NA>      NA
5           5   Michael  21     A               NA       <NA>      NA
6          NA Company A  NA  <NA>                1       Tech   1e+06
7          NA Company B  NA  <NA>                2    Finance   5e+05
8          NA Company C  NA  <NA>                3 Healthcare      NA
9          NA Company D  NA  <NA>                4       Tech   2e+06
10         NA Company E  NA  <NA>                5  Education   8e+05
>
```

bind_cols() Function

```
# Create STUDENT DATA data frame
df1 <- data.frame(
  student_id = c(1, 2, 3, 4, 5),
  name = c("John", "Alice", "Bob", "Emily", "Michael"),
```

```
  age = c(18, 20, NA, 19, 21),
  grade = c("A", "B", "C", "B", "A")
)
# Create ORGANIZATION DATA data frame
df2 <- data.frame(
  organization_id = c(1, 2, 3, 4, 5),
  name = c("Company A", "Company B", "Company C", "Company D", "Company E"),
  industry = c("Tech", "Finance", "Healthcare", "Tech", "Education"),
  revenue = c(1000000, 500000, NA, 2000000, 800000)
)
combined_data <- bind_cols(df1, df2)
combined_data
```

```
> combined_data <- bind_cols(df1, df2)
New names:
• `name` -> `name...2`
• `name` -> `name...6`
> combined_data
  student_id name...2 age grade organization_id  name...6   industry revenue
1          1     John  18     A               1 Company A       Tech   1e+06
2          2    Alice  20     B               2 Company B    Finance   5e+05
3          3      Bob  NA     C               3 Company C Healthcare      NA
4          4    Emily  19     B               4 Company D       Tech   2e+06
5          5  Michael  21     A               5 Company E  Education   8e+05
>
```

----------X----------

Thank you!