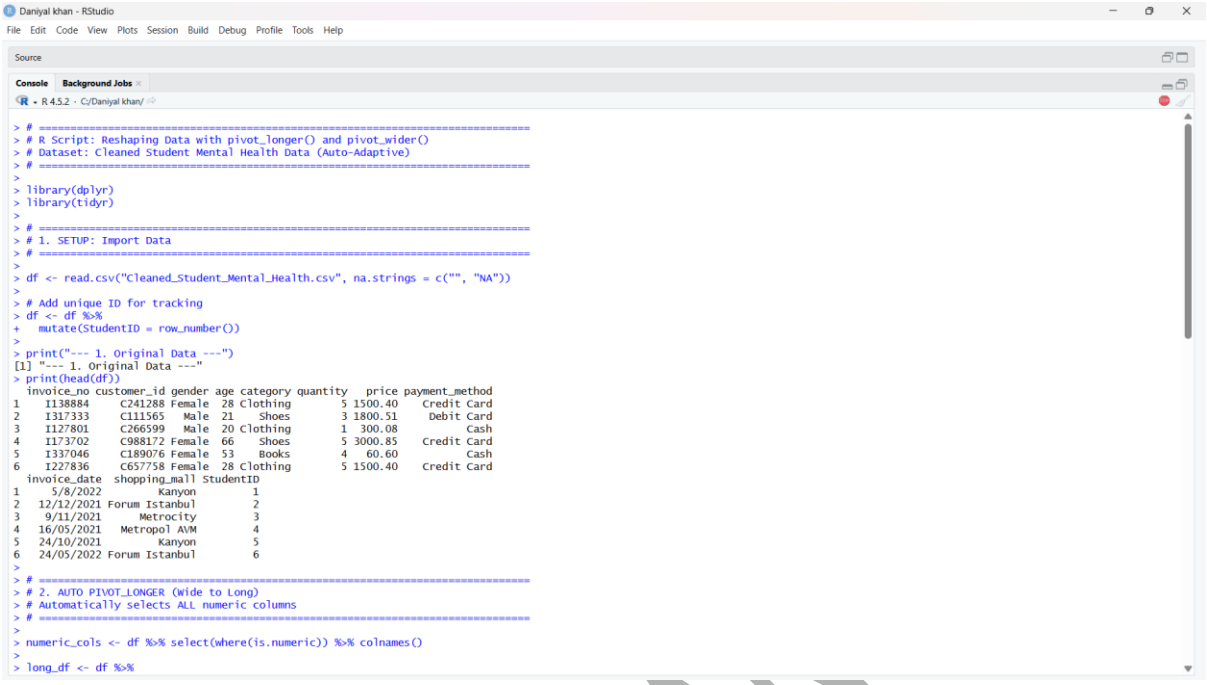
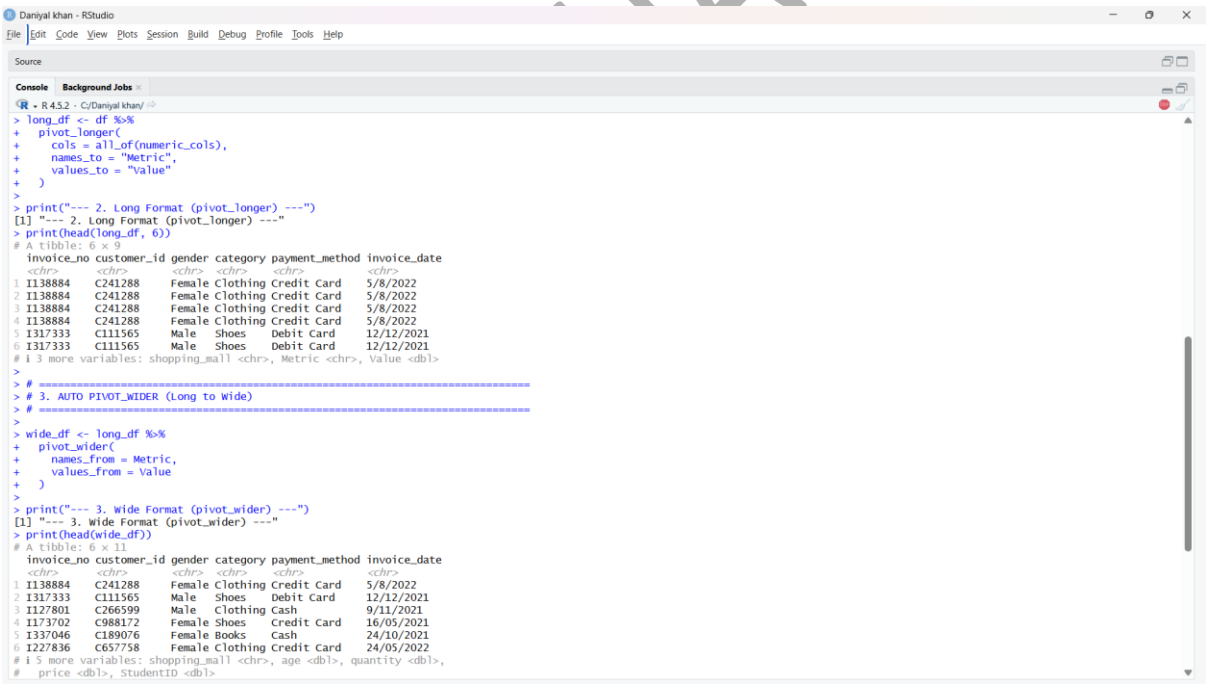
SHETH LUJ COLLLEGE

R-PROGRAMMING

11:-

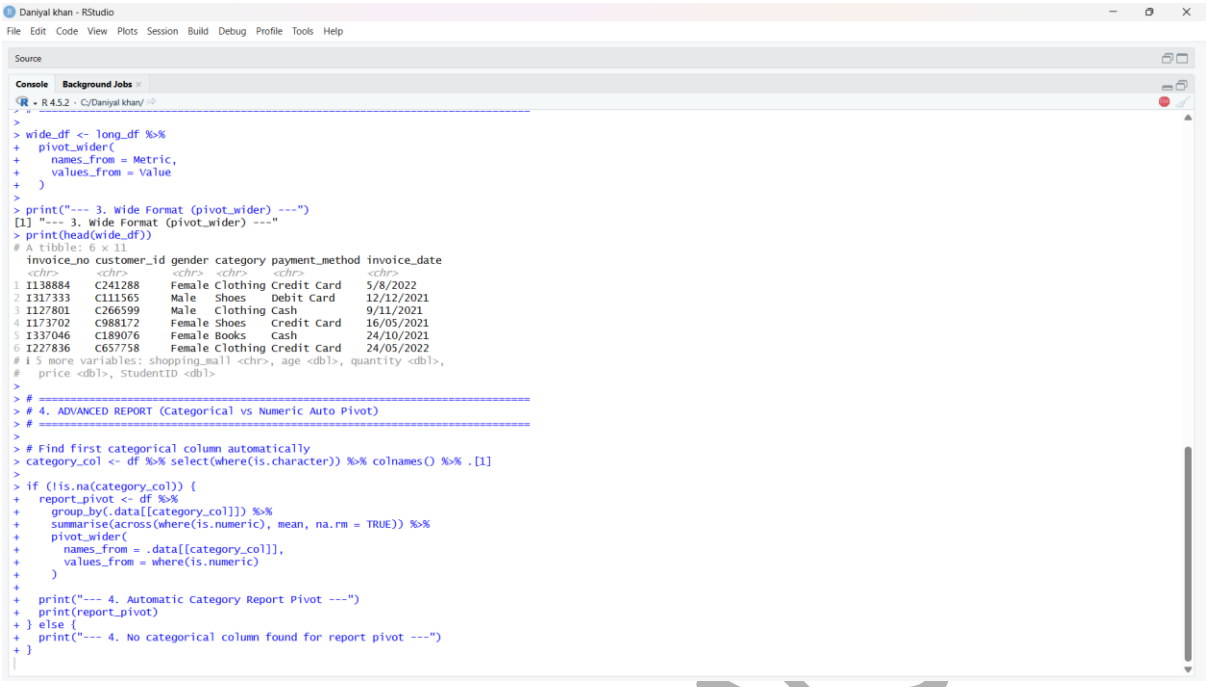




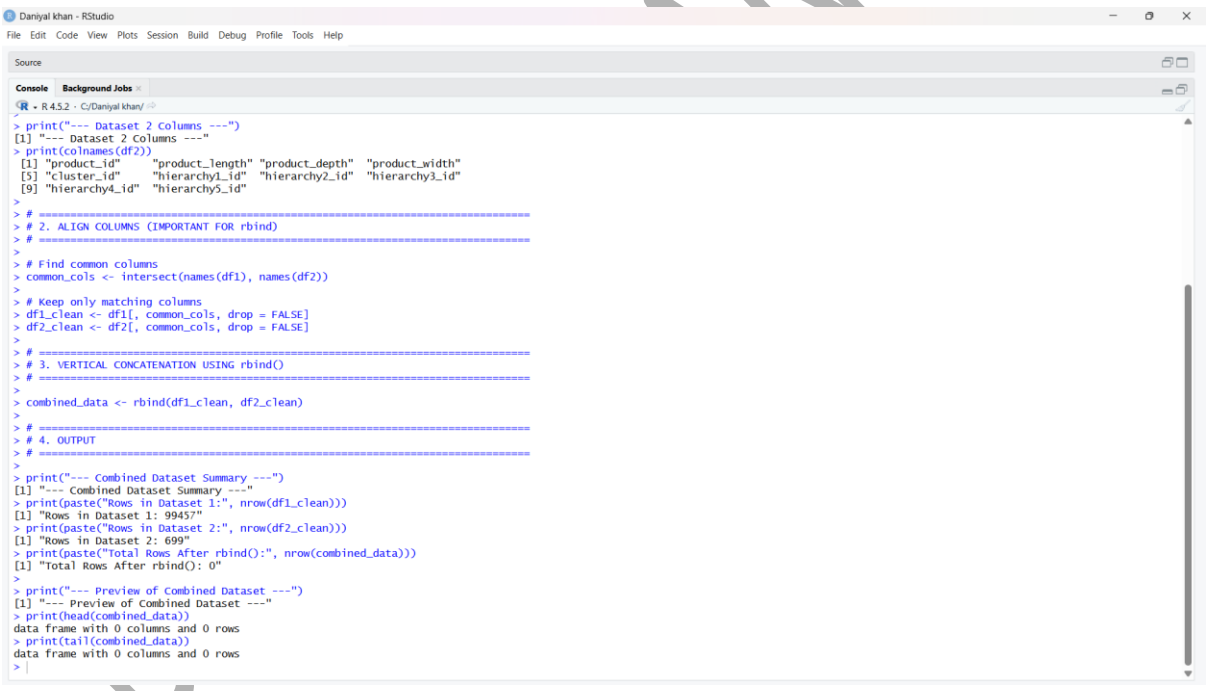
DANIYAL KHAN
S088

SHETH LUJ COLLLEGE

R-PROGRAMMING



12:-

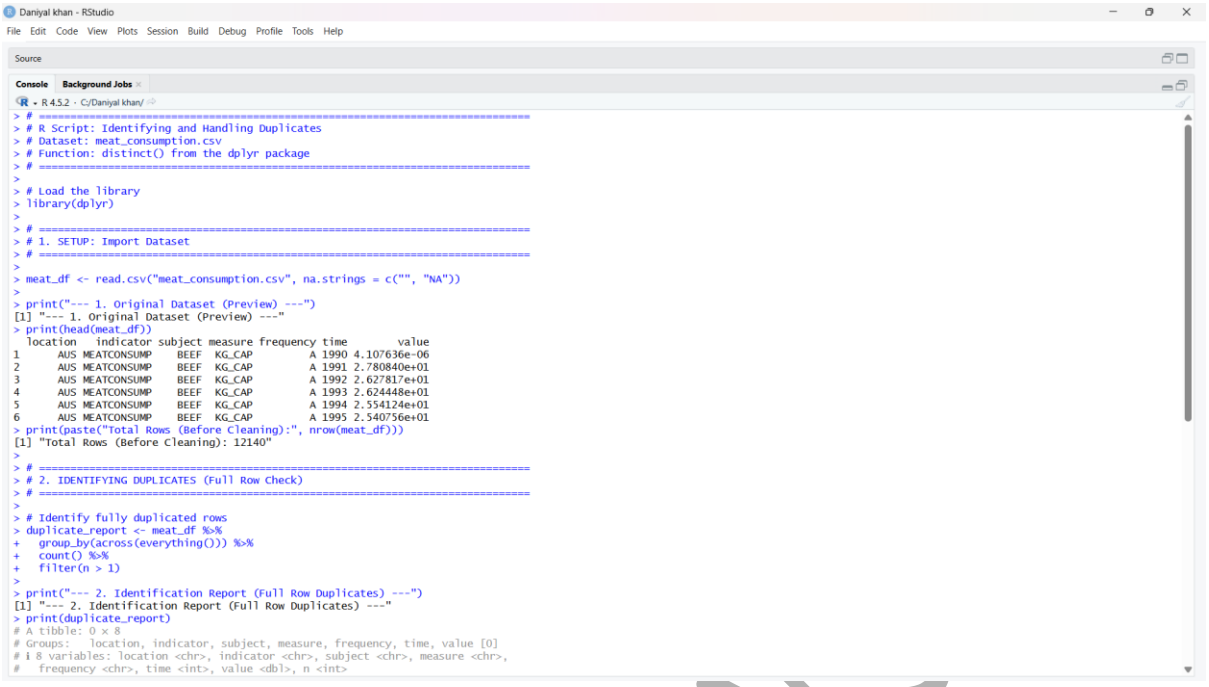


13:-

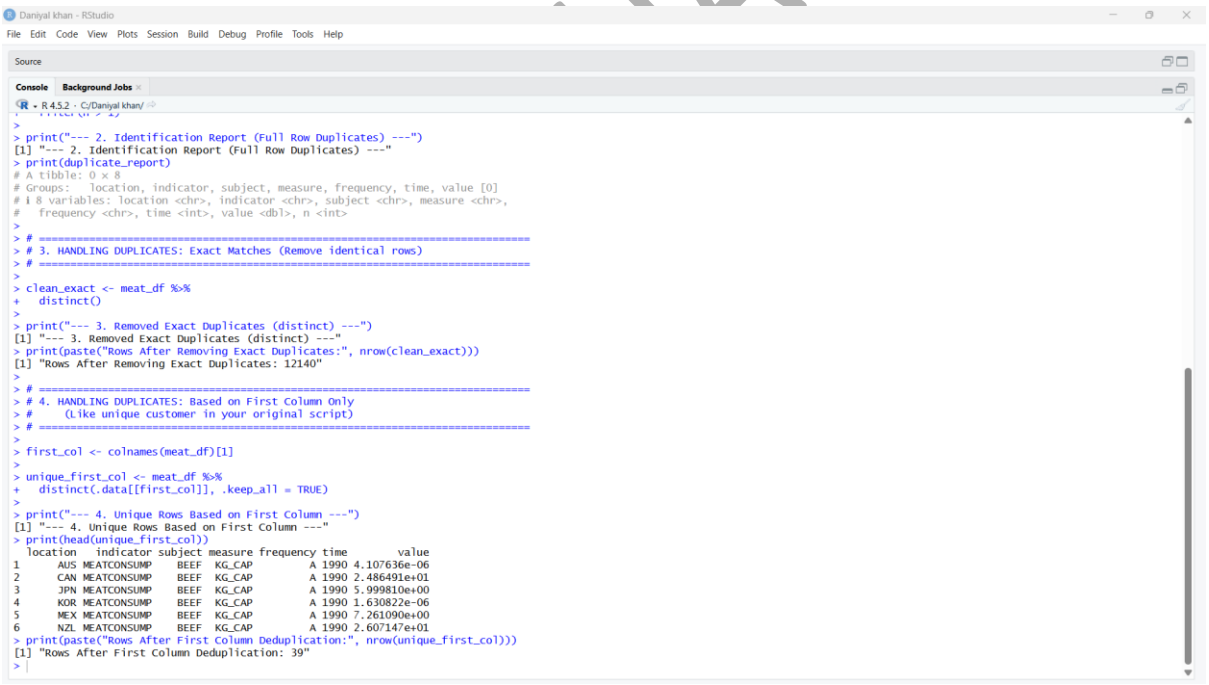
DANIYAL KHAN
S088

SHETH LUJ COLLLEGE

R-PROGRAMMING



```
> # ================================================================================
> # R Script: Identifying and Handling Duplicates
> # Dataset: meat_consumption.csv
> # Function: distinct() from the dplyr package
> # ================================================================================
>
> # Load the library
> library(dplyr)
>
> # ================================================================================
> # 1. SETUP: Import Dataset
> # ================================================================================
>
> meat_df <- read.csv("meat_consumption.csv", na.strings = c("", "NA"))
>
> print("--- 1. Original Dataset (Preview) ---")
[1] "--- 1. Original Dataset (Preview) ---"
> print(head(meat_df))
  location   indicator subject measure frequency time        value
1      AUS MEATCONSUMP    BEEF  KG_CAP         A 1990 4.107636e-06
2      AUS MEATCONSUMP    BEEF  KG_CAP         A 1991 2.780840e+01
3      AUS MEATCONSUMP    BEEF  KG_CAP         A 1992 2.627817e+01
4      AUS MEATCONSUMP    BEEF  KG_CAP         A 1993 2.624448e+01
5      AUS MEATCONSUMP    BEEF  KG_CAP         A 1994 2.554124e+01
6      AUS MEATCONSUMP    BEEF  KG_CAP         A 1995 2.540756e+01
> print(paste("Total Rows (Before Cleaning):", nrow(meat_df)))
[1] "Total Rows (Before Cleaning): 12140"
>
> # ================================================================================
> # 2. IDENTIFYING DUPLICATES (Full Row Check)
> # ================================================================================
>
> # Identify fully duplicated rows
> duplicate_report <- meat_df %>%
+   group_by(across(everything())) %>%
+   count() %>%
+   filter(n > 1)
>
> print("--- 2. Identification Report (Full Row Duplicates) ---")
[1] "--- 2. Identification Report (Full Row Duplicates) ---"
> print(duplicate_report)
# A tibble: 0 × 8
# Groups:   location, indicator, subject, measure, frequency, time, value [0]
# ℹ 8 variables: location <chr>, indicator <chr>, subject <chr>, measure <chr>,
#   frequency <chr>, time <int>, value <dbl>, n <int>
```



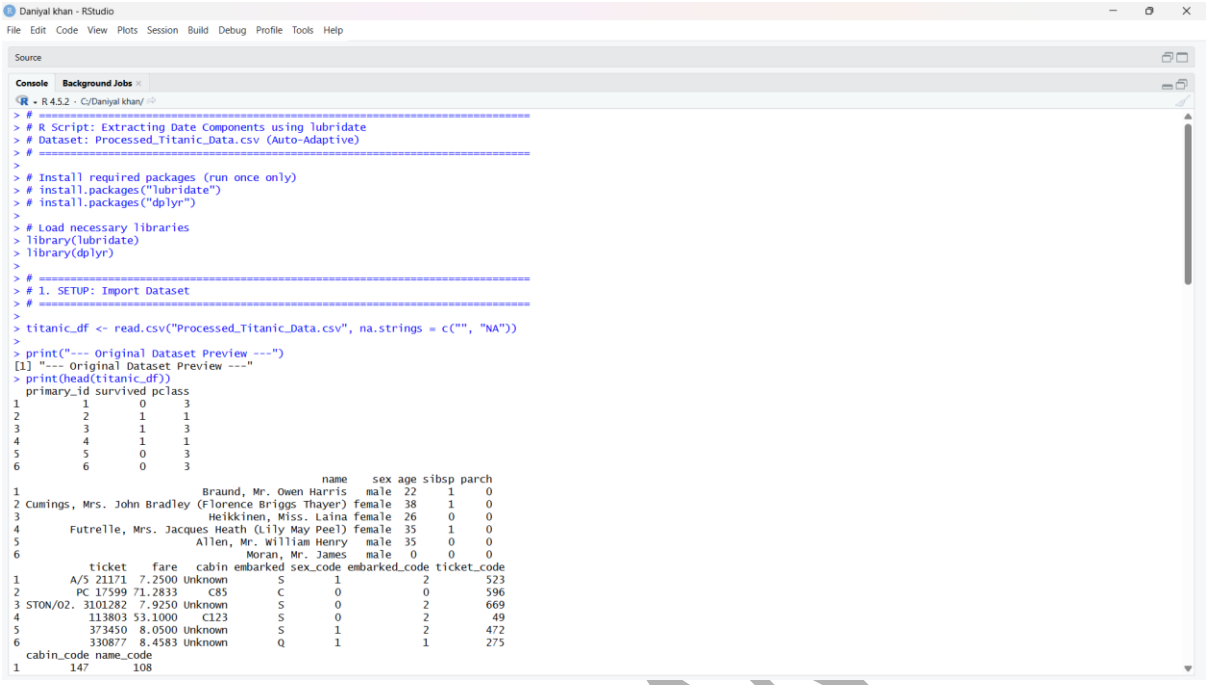
```
> print("--- 2. Identification Report (Full Row Duplicates) ---")
[1] "--- 2. Identification Report (Full Row Duplicates) ---"
> print(duplicate_report)
# A tibble: 0 × 8
# Groups:   location, indicator, subject, measure, frequency, time, value [0]
# ℹ 8 variables: location <chr>, indicator <chr>, subject <chr>, measure <chr>,
#   frequency <chr>, time <int>, value <dbl>, n <int>
>
> # ================================================================================
> # 3. HANDLING DUPLICATES: Exact Matches (Remove identical rows)
> # ================================================================================
>
> clean_exact <- meat_df %>%
+   distinct()
>
> print("--- 3. Removed Exact Duplicates (distinct) ---")
[1] "--- 3. Removed Exact Duplicates (distinct) ---"
> print(paste("Rows After Removing Exact Duplicates:", nrow(clean_exact)))
[1] "Rows After Removing Exact Duplicates: 12140"
>
> # ================================================================================
> # 4. HANDLING DUPLICATES: Based on First Column Only
> #     (Like unique customer in your original script)
> # ================================================================================
>
> first_col <- colnames(meat_df)[1]
>
> unique_first_col <- meat_df %>%
+   distinct(.data[[first_col]], .keep_all = TRUE)
>
> print("--- 4. Unique Rows Based on First Column ---")
[1] "--- 4. Unique Rows Based on First Column ---"
> print(head(unique_first_col))
  location   indicator subject measure frequency time        value
1      AUS MEATCONSUMP    BEEF  KG_CAP         A 1990 4.107636e-06
2      CAN MEATCONSUMP    BEEF  KG_CAP         A 1990 2.486491e+01
3      JPN MEATCONSUMP    BEEF  KG_CAP         A 1990 5.999810e+00
4      KOR MEATCONSUMP    BEEF  KG_CAP         A 1990 1.630822e-06
5      MEX MEATCONSUMP    BEEF  KG_CAP         A 1990 7.261090e+00
6      NZL MEATCONSUMP    BEEF  KG_CAP         A 1990 2.607147e+01
> print(paste("Rows After First Column Deduplication:", nrow(unique_first_col)))
[1] "Rows After First Column Deduplication: 39"
>
```

DANIYAL KHAN
S088

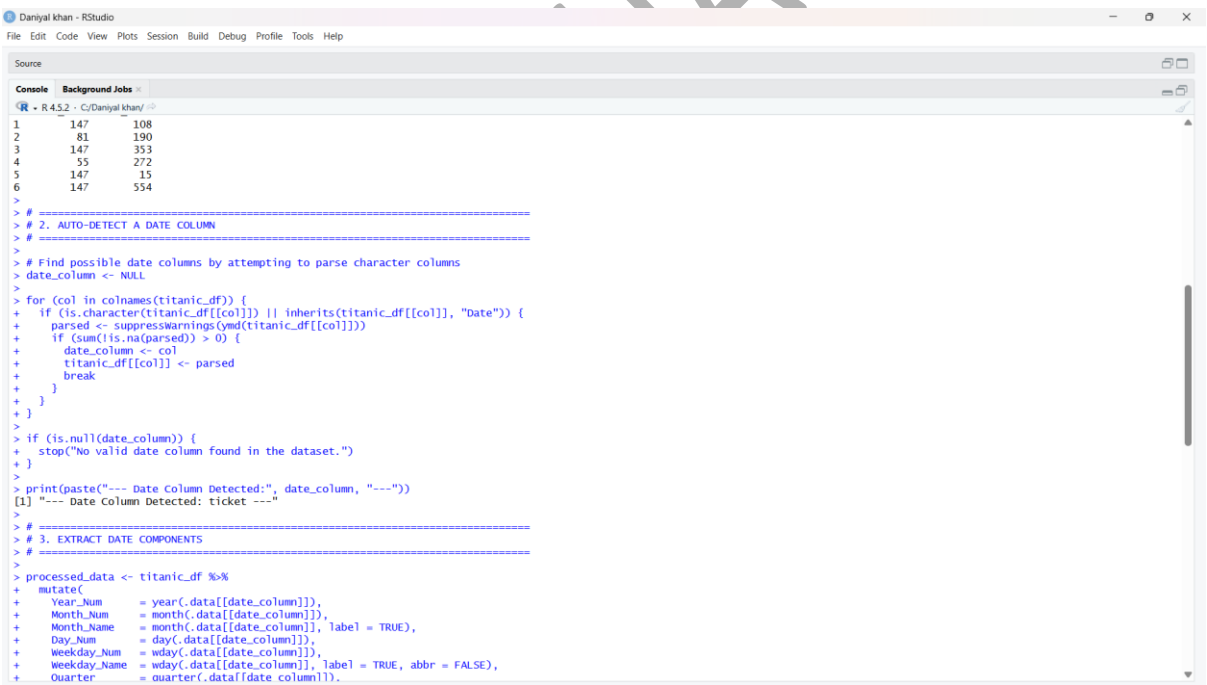
SHETH LUJ COLLLEGE

R-PROGRAMMING

14:-



```
Daniyal khan - RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Source
Console  Background Jobs
R 4.5.2 · C:/Daniyal khan/
> # ================================================================================
> # R Script: Extracting Date Components using lubridate
> # Dataset: Processed_Titanic_Data.csv (Auto-Adaptive)
> # ================================================================================
>
> # Install required packages (run once only)
> # install.packages("lubridate")
> # install.packages("dplyr")
>
> # Load necessary libraries
> library(lubridate)
> library(dplyr)
>
> # ================================================================================
> # 1. SETUP: Import Dataset
> # ================================================================================
>
> titanic_df <- read.csv("Processed_Titanic_Data.csv", na.strings = c("", "NA"))
>
> print("--- Original Dataset Preview ---")
[1] "--- Original Dataset Preview ---"
> print(head(titanic_df))
  primary_id survived pclass
1          1        0      3
2          2        1      1
3          3        1      3
4          4        1      1
5          5        0      3
6          6        0      3
                                                name    sex age sibsp parch
1                            Braund, Mr. Owen Harris   male  22     1     0
2 Cumings, Mrs. John Bradley (Florence Briggs Thayer) female  38     1     0
3                             Heikkinen, Miss. Laina female  26     0     0
4       Futrelle, Mrs. Jacques Heath (Lily May Peel) female  35     1     0
5                           Allen, Mr. William Henry   male  35     0     0
6                                   Moran, Mr. James   male   0     0     0
            ticket     fare   cabin embarked sex_code embarked_code ticket_code
1        A/5 21171   7.2500 Unknown        S        1             2         523
2         PC 17599 71.2833     C85        C        0             0         596
3 STON/O2. 3101282   7.9250 Unknown        S        0             2         669
4           113803 53.1000    C123        S        0             2          49
5           373450   8.0500 Unknown        S        1             2         472
6           330877   8.4583 Unknown        Q        1             1         275
  cabin_code name_code
1        147       108
```

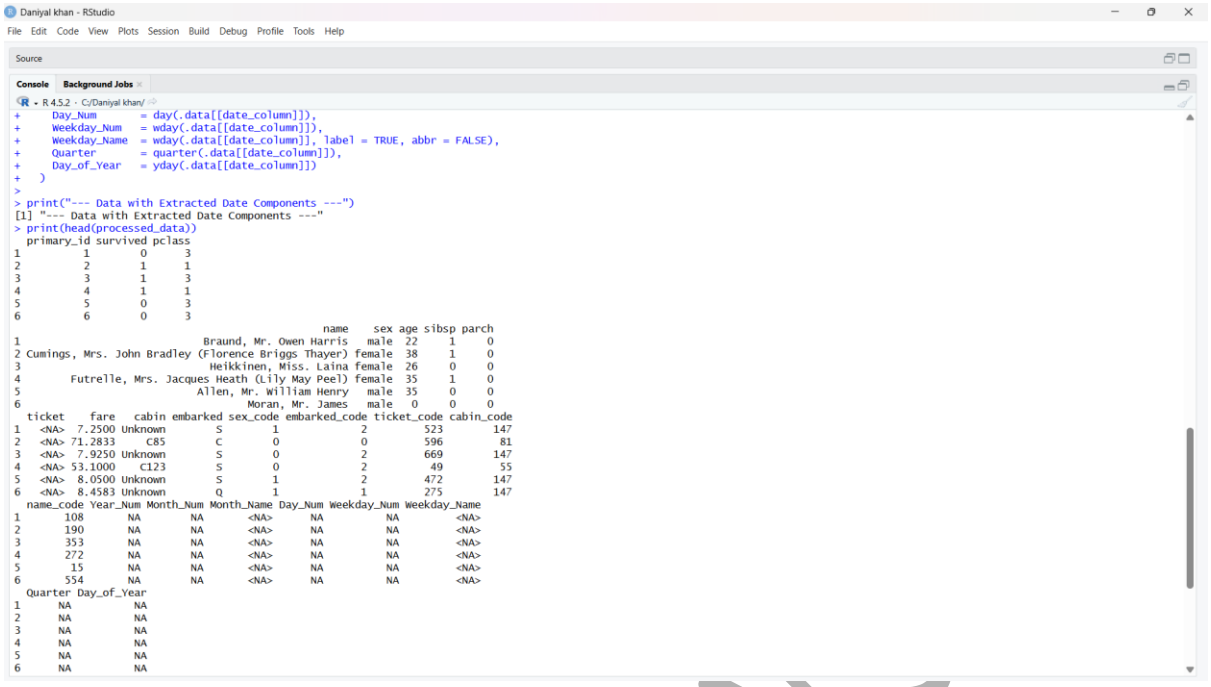


```
Daniyal khan - RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Source
Console  Background Jobs
R 4.5.2 · C:/Daniyal khan/
1        147       108
2         81       190
3        147       353
4         55       272
5        147        15
6        147       554
>
> # ================================================================================
> # 2. AUTO-DETECT A DATE COLUMN
> # ================================================================================
>
> # Find possible date columns by attempting to parse character columns
> date_column <- NULL
>
> for (col in colnames(titanic_df)) {
+   if (is.character(titanic_df[[col]]) || inherits(titanic_df[[col]], "Date")) {
+     parsed <- suppressWarnings(ymd(titanic_df[[col]]))
+     if (sum(!is.na(parsed)) > 0) {
+       date_column <- col
+       titanic_df[[col]] <- parsed
+       break
+     }
+   }
+ }
>
> if (is.null(date_column)) {
+   stop("No valid date column found in the dataset.")
+ }
>
> print(paste("--- Date Column Detected:", date_column, "---"))
[1] "--- Date Column Detected: ticket ---"
>
> # ================================================================================
> # 3. EXTRACT DATE COMPONENTS
> # ================================================================================
>
> processed_data <- titanic_df %>%
+   mutate(
+     Year_Num      = year(.data[[date_column]]),
+     Month_Num     = month(.data[[date_column]]),
+     Month_Name    = month(.data[[date_column]], label = TRUE),
+     Day_Num       = day(.data[[date_column]]),
+     Weekday_Num   = wday(.data[[date_column]]),
+     Weekday_Name  = wday(.data[[date_column]], label = TRUE, abbr = FALSE),
+     Quarter       = quarter(.data[[date_column]]),
```

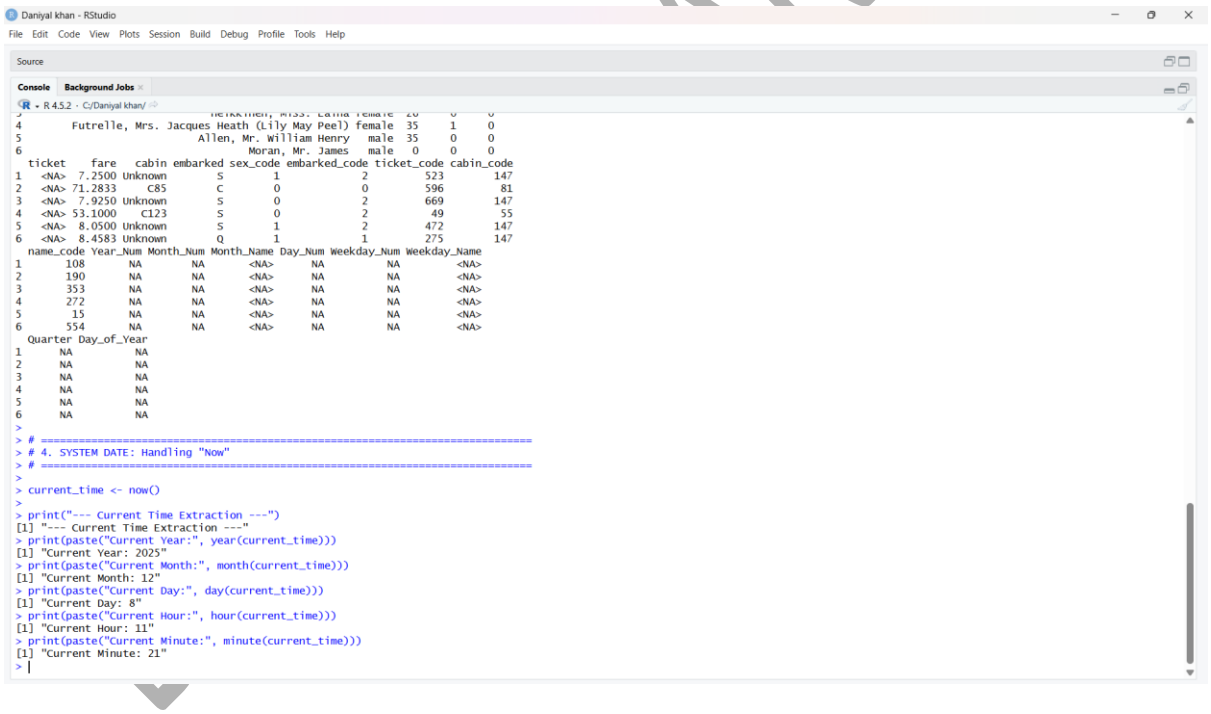
DANIYAL KHAN
S088

SHETH LUJ COLLLEGE

R-PROGRAMMING



```
Daniyal khan - RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Source
Console  Background Jobs
R 4.5.2 · C:/Daniyal khan/
+      Day_Num        = day(.data[[date_column]]),
+      Weekday_Num    = wday(.data[[date_column]]),
+      Weekday_Name   = wday(.data[[date_column]], label = TRUE, abbr = FALSE),
+      Quarter        = quarter(.data[[date_column]]),
+      Day_of_Year    = yday(.data[[date_column]])
+    )
>
> print("--- Data with Extracted Date Components ---")
[1] "--- Data with Extracted Date Components ---"
> print(head(processed_data))
  primary_id survived pclass
1          1        0      3
2          2        1      1
3          3        1      3
4          4        1      1
5          5        0      3
6          6        0      3
                                                 name    sex age sibsp parch
1                             Braund, Mr. Owen Harris   male  22     1     0
2 Cumings, Mrs. John Bradley (Florence Briggs Thayer) female  38     1     0
3                              Heikkinen, Miss. Laina female  26     0     0
4        Futrelle, Mrs. Jacques Heath (Lily May Peel) female  35     1     0
5                            Allen, Mr. William Henry   male  35     0     0
6                                    Moran, Mr. James   male   0     0     0
  ticket    fare   cabin embarked sex_code embarked_code ticket_code cabin_code
1   <NA>  7.2500 Unknown        S        1             2         523        147
2   <NA> 71.2833     C85        C        0             0         596         81
3   <NA>  7.9250 Unknown        S        0             2         669        147
4   <NA> 53.1000    C123        S        0             2          49         55
5   <NA>  8.0500 Unknown        S        1             2         472        147
6   <NA>  8.4583 Unknown        Q        1             1         275        147
  name_code Year_Num Month_Num Month_Name Day_Num Weekday_Num Weekday_Name
1       108       NA        NA       <NA>      NA          NA         <NA>
2       190       NA        NA       <NA>      NA          NA         <NA>
3       353       NA        NA       <NA>      NA          NA         <NA>
4       272       NA        NA       <NA>      NA          NA         <NA>
5        15       NA        NA       <NA>      NA          NA         <NA>
6       554       NA        NA       <NA>      NA          NA         <NA>
  Quarter Day_of_Year
1      NA          NA
2      NA          NA
3      NA          NA
4      NA          NA
5      NA          NA
6      NA          NA
```



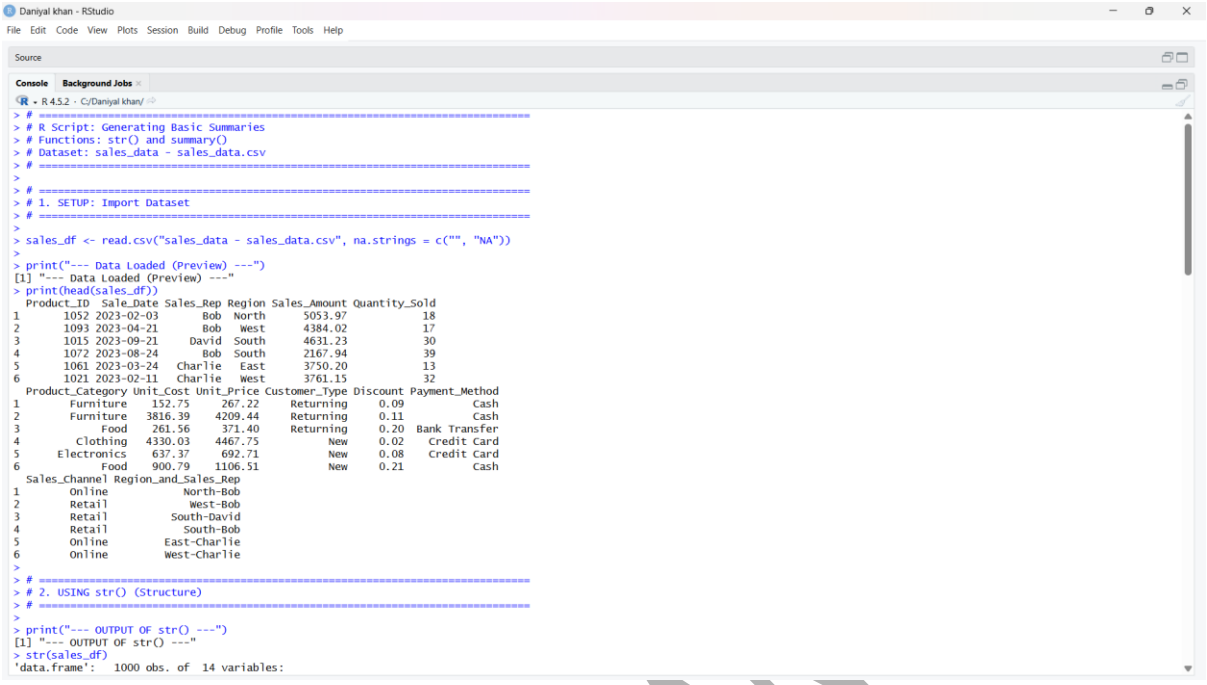
```
Daniyal khan - RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Source
Console  Background Jobs
R 4.5.2 · C:/Daniyal khan/
4        Futrelle, Mrs. Jacques Heath (Lily May Peel) female  35     1     0
5                            Allen, Mr. William Henry   male  35     0     0
6                                    Moran, Mr. James   male   0     0     0
  ticket    fare   cabin embarked sex_code embarked_code ticket_code cabin_code
1   <NA>  7.2500 Unknown        S        1             2         523        147
2   <NA> 71.2833     C85        C        0             0         596         81
3   <NA>  7.9250 Unknown        S        0             2         669        147
4   <NA> 53.1000    C123        S        0             2          49         55
5   <NA>  8.0500 Unknown        S        1             2         472        147
6   <NA>  8.4583 Unknown        Q        1             1         275        147
  name_code Year_Num Month_Num Month_Name Day_Num Weekday_Num Weekday_Name
1       108       NA        NA       <NA>      NA          NA         <NA>
2       190       NA        NA       <NA>      NA          NA         <NA>
3       353       NA        NA       <NA>      NA          NA         <NA>
4       272       NA        NA       <NA>      NA          NA         <NA>
5        15       NA        NA       <NA>      NA          NA         <NA>
6       554       NA        NA       <NA>      NA          NA         <NA>
  Quarter Day_of_Year
1      NA          NA
2      NA          NA
3      NA          NA
4      NA          NA
5      NA          NA
6      NA          NA
>
> # ================================================================
> # 4. SYSTEM DATE: Handling "Now"
> # ================================================================
>
> current_time <- now()
>
> print("--- Current Time Extraction ---")
[1] "--- Current Time Extraction ---"
> print(paste("Current Year:", year(current_time)))
[1] "Current Year: 2025"
> print(paste("Current Month:", month(current_time)))
[1] "Current Month: 12"
> print(paste("Current Day:", day(current_time)))
[1] "Current Day: 8"
> print(paste("Current Hour:", hour(current_time)))
[1] "Current Hour: 11"
> print(paste("Current Minute:", minute(current_time)))
[1] "Current Minute: 21"
>
```

DANIYAL KHAN
S088

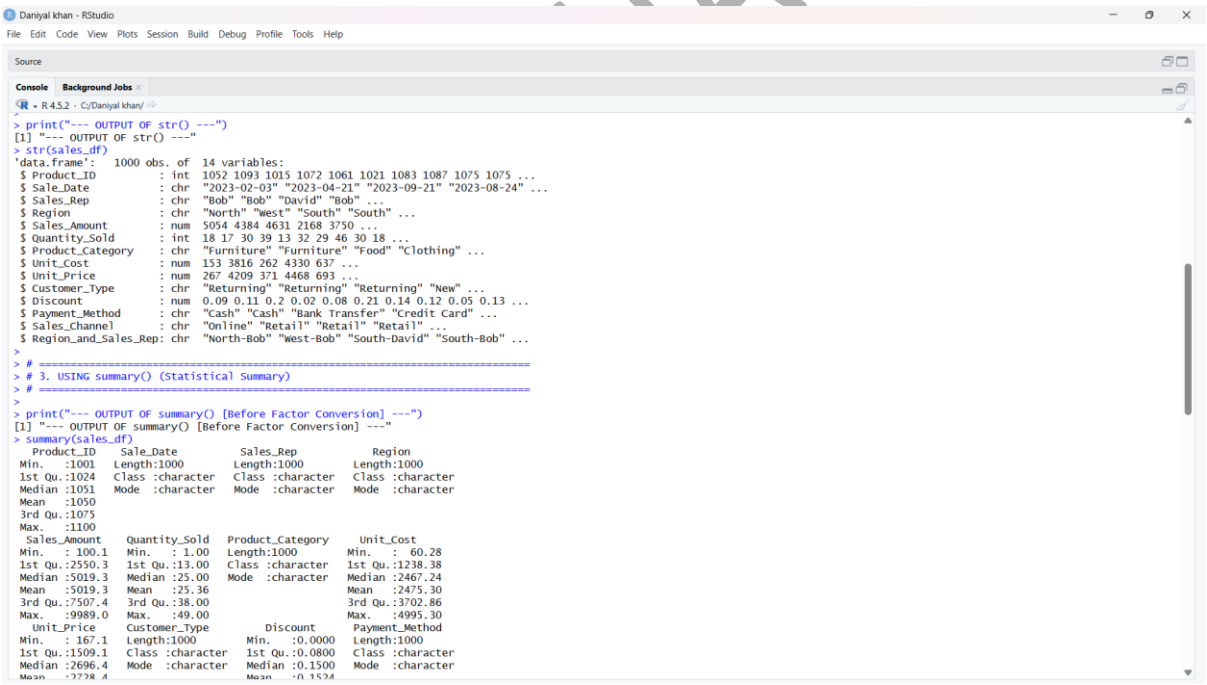
SHETH LUJ COLLLEGE

R-PROGRAMMING

15:-



```
> # ================================================================================
> # R Script: Generating Basic Summaries
> # Functions: str() and summary()
> # Dataset: sales_data - sales_data.csv
> # ================================================================================
>
> # ================================================================================
> # 1. SETUP: Import Dataset
> # ================================================================================
>
> sales_df <- read.csv("sales_data - sales_data.csv", na.strings = c("", "NA"))
>
> print("--- Data Loaded (Preview) ---")
[1] "--- Data Loaded (Preview) ---"
> print(head(sales_df))
  Product_ID  Sale_Date Sales_Rep Region Sales_Amount Quantity_Sold
1       1052 2023-02-03       Bob  North      5053.97            18
2       1093 2023-04-21       Bob   West      4384.02            17
3       1015 2023-09-21     David  South      4631.23            30
4       1072 2023-08-24       Bob  South      2167.94            39
5       1061 2023-03-24   Charlie   East      3750.20            13
6       1021 2023-02-11   Charlie   West      3761.15            32
  Product_Category Unit_Cost Unit_Price Customer_Type Discount Payment_Method
1        Furniture    152.75     267.22     Returning     0.09           Cash
2        Furniture   3816.39    4209.44     Returning     0.11           Cash
3             Food    261.56     371.40     Returning     0.20  Bank Transfer
4         Clothing   4330.03    4467.75           New     0.02    Credit Card
5      Electronics    637.37     692.71           New     0.08    Credit Card
6             Food    900.79    1106.51           New     0.21           Cash
  Sales_Channel Region_and_Sales_Rep
1        Online           North-Bob
2        Retail            West-Bob
3        Retail          South-David
4        Retail            South-Bob
5        Online         East-Charlie
6        Online         West-Charlie
>
> # ================================================================================
> # 2. USING str() (Structure)
> # ================================================================================
>
> print("--- OUTPUT OF str() ---")
[1] "--- OUTPUT OF str() ---"
> str(sales_df)
'data.frame':   1000 obs. of  14 variables:
```

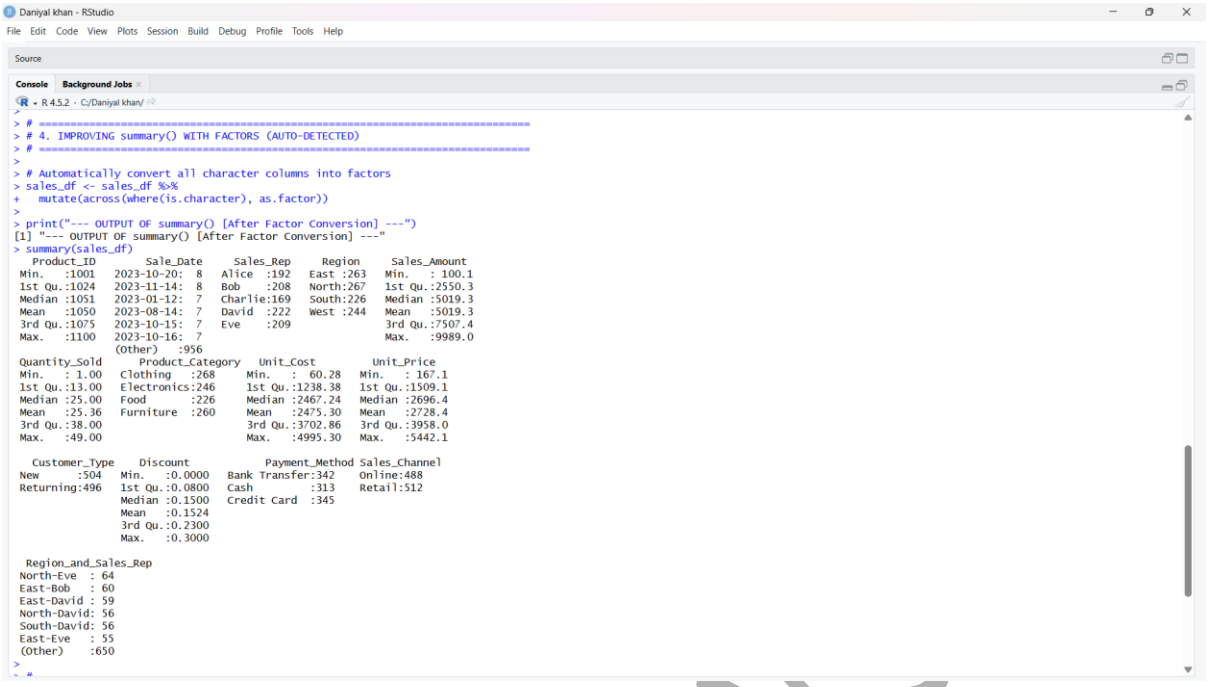


```
> print("--- OUTPUT OF str() ---")
[1] "--- OUTPUT OF str() ---"
> str(sales_df)
'data.frame':   1000 obs. of  14 variables:
 $ Product_ID          : int  1052 1093 1015 1072 1061 1021 1083 1087 1075 1075 ...
 $ Sale_Date           : chr  "2023-02-03" "2023-04-21" "2023-09-21" "2023-08-24" ...
 $ Sales_Rep           : chr  "Bob" "Bob" "David" "Bob" ...
 $ Region              : chr  "North" "West" "South" "South" ...
 $ Sales_Amount        : num  5054 4384 4631 2168 3750 ...
 $ Quantity_Sold       : int  18 17 30 39 13 32 29 46 30 18 ...
 $ Product_Category    : chr  "Furniture" "Furniture" "Food" "Clothing" ...
 $ Unit_Cost           : num  153 3816 262 4330 637 ...
 $ Unit_Price          : num  267 4209 371 4468 693 ...
 $ Customer_Type       : chr  "Returning" "Returning" "Returning" "New" ...
 $ Discount            : num  0.09 0.11 0.2 0.02 0.08 0.21 0.14 0.12 0.05 0.13 ...
 $ Payment_Method      : chr  "Cash" "Cash" "Bank Transfer" "Credit Card" ...
 $ Sales_Channel       : chr  "Online" "Retail" "Retail" "Retail" ...
 $ Region_and_Sales_Rep: chr  "North-Bob" "West-Bob" "South-David" "South-Bob" ...
>
> # ================================================================================
> # 3. USING summary() (Statistical Summary)
> # ================================================================================
>
> print("--- OUTPUT OF summary() [Before Factor Conversion] ---")
[1] "--- OUTPUT OF summary() [Before Factor Conversion] ---"
> summary(sales_df)
   Product_ID    Sale_Date          Sales_Rep           Region
 Min.   :1001   Length:1000        Length:1000        Length:1000
 1st Qu.:1024   Class :character   Class :character   Class :character
 Median :1051   Mode  :character   Mode  :character   Mode  :character
 Mean   :1050
 3rd Qu.:1075
 Max.   :1100
  Sales_Amount    Quantity_Sold   Product_Category     Unit_Cost
 Min.   : 100.1   Min.   : 1.00   Length:1000        Min.   :  60.28
 1st Qu.:2550.3   1st Qu.:13.00   Class :character   1st Qu.:1238.38
 Median :5019.3   Median :25.00   Mode  :character   Median :2467.24
 Mean   :5019.3   Mean   :25.36                      Mean   :2475.30
 3rd Qu.:7507.4   3rd Qu.:38.00                      3rd Qu.:3702.86
 Max.   :9989.0   Max.   :49.00                      Max.   :4995.30
   Unit_Price     Customer_Type         Discount      Payment_Method
 Min.   : 167.1   Length:1000        Min.   :0.0000   Length:1000
 1st Qu.:1509.1   Class :character   1st Qu.:0.0800   Class :character
 Median :2696.4   Mode  :character   Median :0.1500   Mode  :character
 Mean   :2728.4                      Mean   :0.1524
```

DANIYAL KHAN
S088

SHETH LUJ COLLLEGE

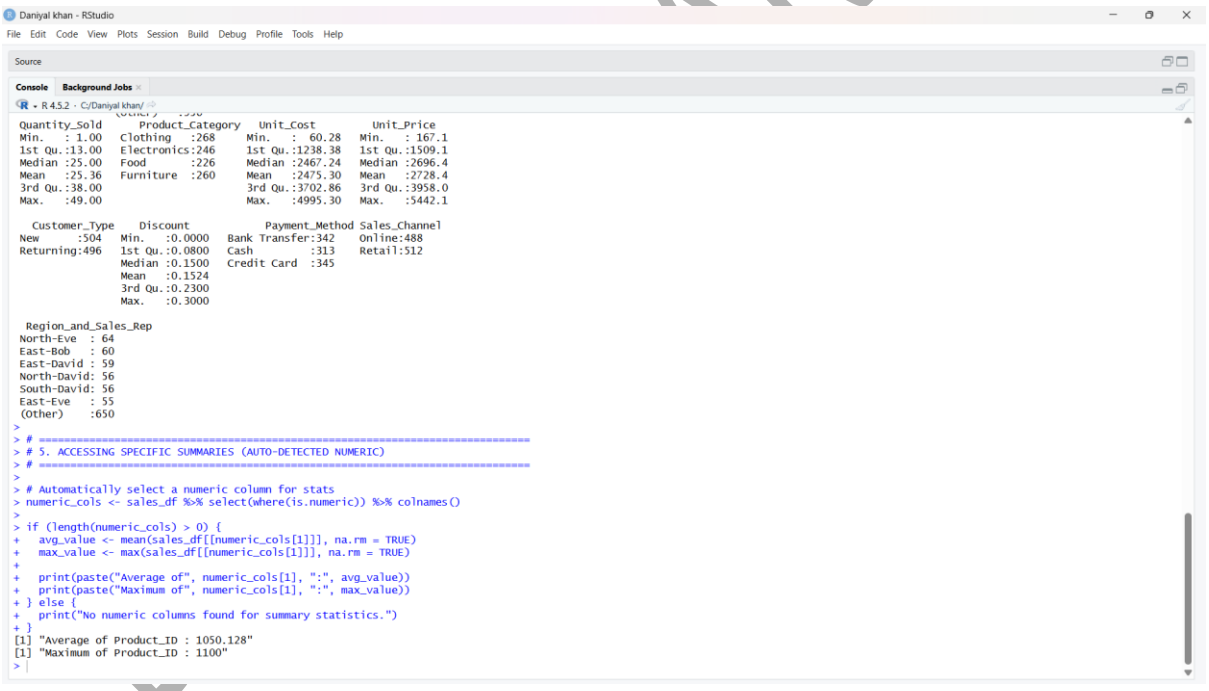
R-PROGRAMMING



```
> # ================================================================================
> # 4. IMPROVING summary() WITH FACTORS (AUTO-DETECTED)
> # ================================================================================
>
> # Automatically convert all character columns into factors
> sales_df <- sales_df %>%
+   mutate(across(where(is.character), as.factor))
>
> print("--- OUTPUT OF summary() [After Factor Conversion] ---")
[1] "--- OUTPUT OF summary() [After Factor Conversion] ---"
> summary(sales_df)
  Product_ID          Sale_Date     Sales_Rep     Region     Sales_Amount
Min.   :1001   2023-10-20:  8   Alice  :192   East :263   Min.   : 100.1
1st Qu.:1024   2023-11-14:  8   Bob    :208   North:267   1st Qu.:2550.3
Median :1051   2023-01-12:  7   Charlie:169   South:226   Median :5019.3
Mean   :1050   2023-08-14:  7   David  :222   West :244   Mean   :5019.3
3rd Qu.:1075   2023-10-15:  7   Eve    :209               3rd Qu.:7507.4
Max.   :1100   2023-10-16:  7                             Max.   :9989.0
               (Other)   :956
Quantity_Sold      Product_Category   Unit_Cost         Unit_Price
Min.   : 1.00   Clothing   :268     Min.   :  60.28   Min.   : 167.1
1st Qu.:13.00   Electronics:246     1st Qu.:1238.38   1st Qu.:1509.1
Median :25.00   Food       :226     Median :2467.24   Median :2696.4
Mean   :25.36   Furniture  :260     Mean   :2475.30   Mean   :2728.4
3rd Qu.:38.00                       3rd Qu.:3702.86   3rd Qu.:3958.0
Max.   :49.00                       Max.   :4995.30   Max.   :5442.1

  Customer_Type    Discount           Payment_Method Sales_Channel
New      :504   Min.   :0.0000   Bank Transfer:342   Online:488
Returning:496   1st Qu.:0.0800   Cash         :313   Retail:512
                Median :0.1500   Credit Card  :345
                Mean   :0.1524
                3rd Qu.:0.2300
                Max.   :0.3000

 Region_and_Sales_Rep
North-Eve  : 64
East-Bob   : 60
East-David : 59
North-David: 56
South-David: 56
East-Eve   : 55
(Other)    :650
>
> #
```



```
Quantity_Sold      Product_Category   Unit_Cost         Unit_Price
Min.   : 1.00   Clothing   :268     Min.   :  60.28   Min.   : 167.1
1st Qu.:13.00   Electronics:246     1st Qu.:1238.38   1st Qu.:1509.1
Median :25.00   Food       :226     Median :2467.24   Median :2696.4
Mean   :25.36   Furniture  :260     Mean   :2475.30   Mean   :2728.4
3rd Qu.:38.00                       3rd Qu.:3702.86   3rd Qu.:3958.0
Max.   :49.00                       Max.   :4995.30   Max.   :5442.1

  Customer_Type    Discount           Payment_Method Sales_Channel
New      :504   Min.   :0.0000   Bank Transfer:342   Online:488
Returning:496   1st Qu.:0.0800   Cash         :313   Retail:512
                Median :0.1500   Credit Card  :345
                Mean   :0.1524
                3rd Qu.:0.2300
                Max.   :0.3000

 Region_and_Sales_Rep
North-Eve  : 64
East-Bob   : 60
East-David : 59
North-David: 56
South-David: 56
East-Eve   : 55
(Other)    :650
>
> # ================================================================================
> # 5. ACCESSING SPECIFIC SUMMARIES (AUTO-DETECTED NUMERIC)
> # ================================================================================
>
> # Automatically select a numeric column for stats
> numeric_cols <- sales_df %>% select(where(is.numeric)) %>% colnames()
>
> if (length(numeric_cols) > 0) {
+   avg_value <- mean(sales_df[[numeric_cols[1]]], na.rm = TRUE)
+   max_value <- max(sales_df[[numeric_cols[1]]], na.rm = TRUE)
+
+   print(paste("Average of", numeric_cols[1], ":", avg_value))
+   print(paste("Maximum of", numeric_cols[1], ":", max_value))
+ } else {
+   print("No numeric columns found for summary statistics.")
+ }
[1] "Average of Product_ID : 1050.128"
[1] "Maximum of Product_ID : 1100"
>
```

DANIYAL KHAN
S088

SHETH LUJ COLLLEGE

R-PROGRAMMING

DANIYAL KHAN
S088