

SETH L.U.J and SIR M.V COLLEGE

PRAC 11 12 13 14 15

AIM:- 11 Reshaping data using pivot_longer()/pivot_wider() (R).

The screenshot displays the RStudio interface with the following components:

- Source Editor:** Contains R code for loading data and reshaping it. The code includes comments and function calls like `pivot_longer()` and `pivot_wider()`.
- Environment:** Shows the objects created in the environment, including `category_arrow`, `df`, `long_df`, `student_marks`, and `wide_df`.
- Console:** Displays the output of the R code, showing the original wide data and the resulting long data format.
- Files:** Lists the files in the project, including `student_marks.csv`.

```
# R Script: Reshaping data using pivot_longer()/pivot_wider() (R)

# 1. Setup: Read student marks data and add student ID
df <- read.csv("C:/Users/mvlu/downloads/student_marks.csv", na.strings = c("", "NA")) %>%
  mutate(student_id = row_number()) %>%
  select(student_id, number_courses, time_study, marks)

# Print the original wide data
print("1. Original wide data ---")
print(head(df))

# 2. PIVOT_LONGER: Stack study metrics and marks into long format
long_df <- df %>%
  pivot_longer(
    cols = c(number_courses, time_study, marks),
    names_to = "metric",
    values_to = "value"
  )

# Print the long format (pivot_longer)
print("2. Long format (pivot_longer) ---")
```

The console output shows the first six rows of the data:

student_id	number_courses	time_study	marks
1	3	8.508	18.382
2	3	0.098	7.782
3	3	3.133	11.811
4	4	7.909	11.018
5	5	7.411	55.299
6	6	3.211	17.822

SETH L.U.J and SIR M.V COLLEGE

PRAC 11 12 13 14 15

The top screenshot shows an RStudio session where a data frame is being manipulated. The console output shows the result of a `pivot_wider` operation, which has created a wide data frame with columns for `number_courses`, `time_study`, and `marks`. The data is as follows:

studentID	number_courses	time_study	marks
1	3	4.51	19.2
2	4	0.098	7.73
3	4	3.13	13.8
4	6	7.81	15.1
5	6	3.21	17.8

The bottom screenshot shows the same RStudio session, but with a `pivot_longer` operation. The console output shows the result of a `pivot_longer` operation, which has created a long data frame with columns for `number_courses`, `time_study`, and `marks`. The data is as follows:

studentID	number_courses	time_study	marks
1	3	4.51	19.2
2	4	0.098	7.73
3	4	3.13	13.8
4	6	7.81	15.1
5	6	3.21	17.8

SETH L.U.J and SIR M.V COLLEGE

PRAC 11 12 13 14 15

```

19 # Print out the long format (pivot_longer)
20 print("---- 2. Long Format (pivot_longer) ----")
21 print(head(long_df, 5)) # show first 5 rows (3 metrics per student)
22
23 # 3. PIVOT WIDER: Spread data into wide format
24 wide_df <- long_df %>%
25   pivot_wider(
26     names_from = Metric,
27     values_from = Value
28   )
29
30 # 4. ADVANCED EXAMPLE: Pivot by number_courses to see Marks distribution
31 category_pivot <- df %>%
32   select(StudentID, number_courses, Marks) %>%
33   pivot_wider(
34     names_from = number_courses,
35     values_from = Marks
36   )
37
38 # Print out the wide format (pivot_wider)
39 print("---- 4. Course Pivot (Spreading number_courses) ----")
40 print(head(category_pivot))
41 # Print out the wide format (pivot_wider)
42 # Print out the wide format (pivot_wider)
43 studentID 1 2 3 4 5 6
44 Marks      10.2 NA NA NA NA NA
45           7.73 NA NA NA NA NA
46           13.8 NA NA NA NA NA
47           NA NA 53.0 NA NA NA
48           NA NA NA 55.3 NA NA
49           NA NA NA 17.8 NA NA

```

12 Combining datasets vertically (concatenation) using rbind() (R).

```

6 bats_clean <- data.frame(
7   Name = bats_df$species,
8   Value = ifelse(bats_df$numeric,
9     NA, NA)
10 )
11 student_clean <- data.frame(
12   Name = student_df$number_courses,
13   Value = as.numeric(student_df$marks)
14 )
15
16 # Print out the combined data (Top and Bottom)
17 # Print out the combined data (Top and Bottom)
18 # Print out the combined data (Top and Bottom)
19 # Print out the combined data (Top and Bottom)
20 # Print out the combined data (Top and Bottom)
21 # Print out the combined data (Top and Bottom)
22 # Print out the combined data (Top and Bottom)
23 # Print out the combined data (Top and Bottom)
24 # Print out the combined data (Top and Bottom)
25 # Print out the combined data (Top and Bottom)
26 # Print out the combined data (Top and Bottom)
27 # Print out the combined data (Top and Bottom)
28 # Print out the combined data (Top and Bottom)
29 # Print out the combined data (Top and Bottom)
30 # Print out the combined data (Top and Bottom)
31 # Print out the combined data (Top and Bottom)
32 # Print out the combined data (Top and Bottom)
33 # Print out the combined data (Top and Bottom)
34 # Print out the combined data (Top and Bottom)
35 # Print out the combined data (Top and Bottom)
36 # Print out the combined data (Top and Bottom)
37 # Print out the combined data (Top and Bottom)
38 # Print out the combined data (Top and Bottom)
39 # Print out the combined data (Top and Bottom)
40 # Print out the combined data (Top and Bottom)
41 # Print out the combined data (Top and Bottom)
42 # Print out the combined data (Top and Bottom)
43 # Print out the combined data (Top and Bottom)
44 # Print out the combined data (Top and Bottom)
45 # Print out the combined data (Top and Bottom)
46 # Print out the combined data (Top and Bottom)
47 # Print out the combined data (Top and Bottom)
48 # Print out the combined data (Top and Bottom)
49 # Print out the combined data (Top and Bottom)
50 # Print out the combined data (Top and Bottom)
51 # Print out the combined data (Top and Bottom)
52 # Print out the combined data (Top and Bottom)
53 # Print out the combined data (Top and Bottom)
54 # Print out the combined data (Top and Bottom)
55 # Print out the combined data (Top and Bottom)
56 # Print out the combined data (Top and Bottom)
57 # Print out the combined data (Top and Bottom)
58 # Print out the combined data (Top and Bottom)
59 # Print out the combined data (Top and Bottom)
60 # Print out the combined data (Top and Bottom)
61 # Print out the combined data (Top and Bottom)
62 # Print out the combined data (Top and Bottom)
63 # Print out the combined data (Top and Bottom)
64 # Print out the combined data (Top and Bottom)
65 # Print out the combined data (Top and Bottom)
66 # Print out the combined data (Top and Bottom)
67 # Print out the combined data (Top and Bottom)
68 # Print out the combined data (Top and Bottom)
69 # Print out the combined data (Top and Bottom)
70 # Print out the combined data (Top and Bottom)
71 # Print out the combined data (Top and Bottom)
72 # Print out the combined data (Top and Bottom)
73 # Print out the combined data (Top and Bottom)
74 # Print out the combined data (Top and Bottom)
75 # Print out the combined data (Top and Bottom)
76 # Print out the combined data (Top and Bottom)
77 # Print out the combined data (Top and Bottom)
78 # Print out the combined data (Top and Bottom)
79 # Print out the combined data (Top and Bottom)
80 # Print out the combined data (Top and Bottom)
81 # Print out the combined data (Top and Bottom)
82 # Print out the combined data (Top and Bottom)
83 # Print out the combined data (Top and Bottom)
84 # Print out the combined data (Top and Bottom)
85 # Print out the combined data (Top and Bottom)
86 # Print out the combined data (Top and Bottom)
87 # Print out the combined data (Top and Bottom)
88 # Print out the combined data (Top and Bottom)
89 # Print out the combined data (Top and Bottom)
90 # Print out the combined data (Top and Bottom)
91 # Print out the combined data (Top and Bottom)
92 # Print out the combined data (Top and Bottom)
93 # Print out the combined data (Top and Bottom)
94 # Print out the combined data (Top and Bottom)
95 # Print out the combined data (Top and Bottom)
96 # Print out the combined data (Top and Bottom)
97 # Print out the combined data (Top and Bottom)
98 # Print out the combined data (Top and Bottom)
99 # Print out the combined data (Top and Bottom)
100 # Print out the combined data (Top and Bottom)

```

AIM:- 13 Identifying and handling duplicates using distinct() (R).

OUTPUT:-

SETH L.U.J and SIR M.V COLLEGE

PRAC 11 12 13 14 15

```

RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

> bats_df <- read.csv("C:/Users/mvlsi/Downloads/bats_information.csv")
> bats_clean <- data.frame(
+   Name = bats_df$Species,
+   Value = as.numeric(sub("-", "", bats_df$Average, ifespan))
+ )

Warning message:
In data.frame(Name = bats_df$Species, Value = as.numeric(sub("-", "", :
NA: introduced by coercion

> print("--- Original Data ---")
[1] "--- Original Data ---"
> print(bats_clean)

      Name Value
1  Vangochiroptera (Microbat); Example: Big Brown Bat    NA
2  Vangochiroptera (Microbat); Example: Little Brown Bat    NA
3  Vangochiroptera (Microbat); Example: Brandt's Bat    NA
4  Vincterchiroptera (Megabat); Example: Horseshoe Bat    NA
5  Vincterchiroptera (Megabat); Example: Vampire Bat    NA
6  Vincterchiroptera (Megabat); Example: Fringed Myotis    NA
7  Vincterchiroptera (Megabat); Example: Horseshoe Bat    NA
8  Vincterchiroptera (Megabat); Example: Brandt's Bat    NA
9  Vangochiroptera (Microbat); Example: Big Brown Bat    NA
10 Vangochiroptera (Microbat); Example: Horseshoe Bat    NA
11 Vangochiroptera (Microbat); Example: Big Brown Bat    NA
12 Vincterchiroptera (Megabat); Example: Little Brown Bat    NA
13 Vincterchiroptera (Megabat); Example: Brandt's Bat    NA
14 Vincterchiroptera (Megabat); Example: Little Brown Bat    NA
15 Vangochiroptera (Microbat); Example: Little Brown Bat    NA
16 Vangochiroptera (Microbat); Example: Bumblebee Bat    NA
17 Vangochiroptera (Microbat); Example: Horseshoe Bat    NA
18 Vangochiroptera (Microbat); Example: Horseshoe Bat    NA
19 Vincterchiroptera (Megabat); Example: Fringed Myotis    NA
20 Vincterchiroptera (Megabat); Example: Big Brown Bat    NA
21 Vangochiroptera (Microbat); Example: Fringed Myotis    NA
22 Vangochiroptera (Microbat); Example: Big Brown Bat    NA

```

```

RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

> bats_clean %>%
+   group_by(Name, Value) %>%
+   count() %>%
+   filter(n > 1)

> print("--- Duplicate Rows ---")
[1] "--- Duplicate Rows ---"
> print(duplicates_report)
# A tibble: 18 x 3
# Groups:   Name, Value [18]
      Name                                     Value     n
  <chr>                                     <dbl> <int>
1 Vangochiroptera (Microbat); Example: Big Brown Bat    NA    1672
2 Vangochiroptera (Microbat); Example: Brandt's Bat    NA    1687
3 Vangochiroptera (Microbat); Example: Bumblebee Bat    NA    1490
4 Vangochiroptera (Microbat); Example: Egyptian Fruit Bat    NA    1709
5 Vangochiroptera (Microbat); Example: Fringed Myotis    NA    1749
6 Vangochiroptera (Microbat); Example: Horseshoe Bat    NA    1636
7 Vangochiroptera (Microbat); Example: Little Brown Bat    NA    1644
8 Vangochiroptera (Microbat); Example: Spectral Bat    NA    1492
9 Vangochiroptera (Microbat); Example: Vampire Bat    NA    1496
10 Vincterchiroptera (Megabat); Example: Big Brown Bat    NA    1693
11 Vincterchiroptera (Megabat); Example: Brandt's Bat    NA    1723
12 Vincterchiroptera (Megabat); Example: Bumblebee Bat    NA    1617
13 Vincterchiroptera (Megabat); Example: Egyptian Fruit Bat    NA    1657
14 Vincterchiroptera (Megabat); Example: Fringed Myotis    NA    1732
15 Vincterchiroptera (Megabat); Example: Horseshoe Bat    NA    1653

```


The screenshot shows the RStudio interface with the following content:

Console:

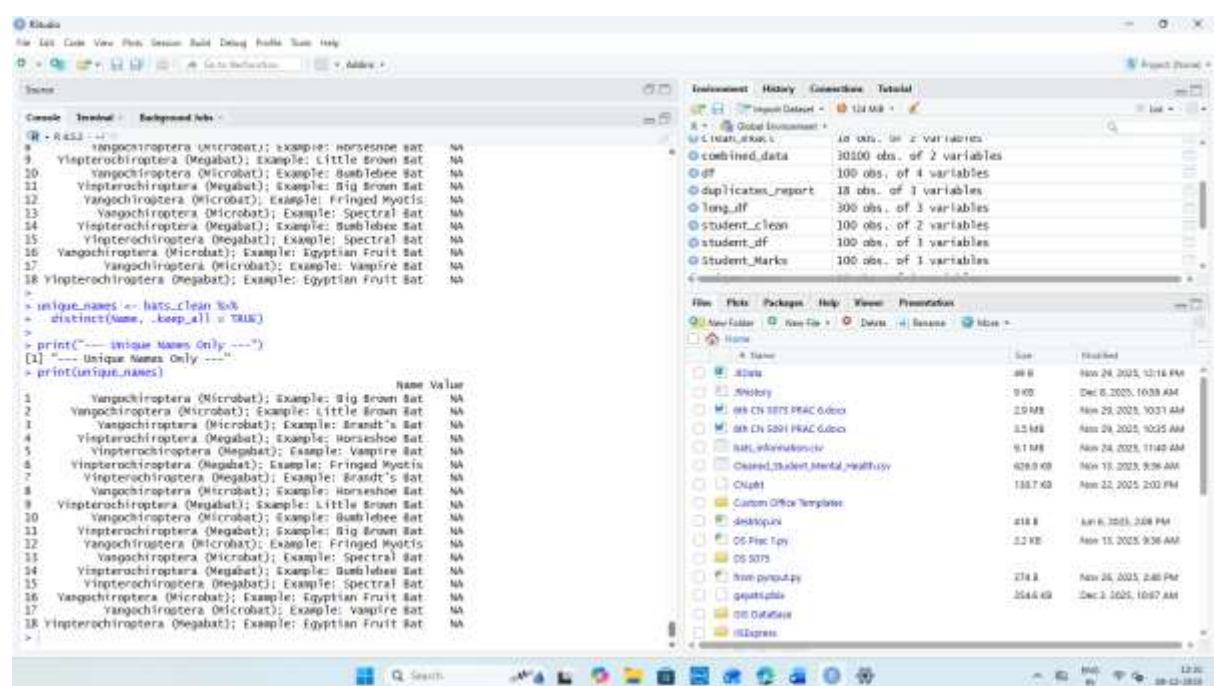
```
R - R65.1 -- 64-bit
# Source: Name, Value [18]
#>
#>      Name      Value      n
#>      <chr>    <dbl>    <dbl>
1 Yangchiroptera (Microbat); Example: Big Brown Bat NA 1672
2 Yangchiroptera (Microbat); Example: Brandt's Bat NA 1687
3 Yangchiroptera (Microbat); Example: Bumblebee Bat NA 1690
4 Yangchiroptera (Microbat); Example: Egyptian Fruit Bat NA 1709
5 Yangchiroptera (Microbat); Example: Fringed Myotis NA 1749
6 Yangchiroptera (Microbat); Example: Horseshoe Bat NA 1636
7 Yangchiroptera (Microbat); Example: Little Brown Bat NA 1644
8 Yangchiroptera (Microbat); Example: Spectral Bat NA 1592
9 Yangchiroptera (Microbat); Example: Vampire Bat NA 1696
10 Yinpterochiroptera (Megabat); Example: Big Brown Bat NA 1693
11 Yinpterochiroptera (Megabat); Example: Brandt's Bat NA 1723
12 Yinpterochiroptera (Megabat); Example: Bumblebee Bat NA 1617
13 Yinpterochiroptera (Megabat); Example: Egyptian Fruit Bat NA 1657
14 Yinpterochiroptera (Megabat); Example: Fringed Myotis NA 1732
15 Yinpterochiroptera (Megabat); Example: Horseshoe Bat NA 1653
16 Yinpterochiroptera (Megabat); Example: Little Brown Bat NA 1660
17 Yinpterochiroptera (Megabat); Example: Spectral Bat NA 1577
18 Yinpterochiroptera (Megabat); Example: Vampire Bat NA 1613

#> clean_exact <- bats_clean %>%
#>   distinct()
#>
#> print("--- Exact Duplicates Removed ---")
[1] --- Exact Duplicates Removed ---
#> print(clean_exact)

#>
#>      Name      Value      n
#>      <chr>    <dbl>    <dbl>
1 Yangchiroptera (Microbat); Example: Big Brown Bat NA
2 Yangchiroptera (Microbat); Example: Little Brown Bat NA
3 Yangchiroptera (Microbat); Example: Brandt's Bat NA
4 Yinpterochiroptera (Megabat); Example: Horseshoe Bat NA
5 Yinpterochiroptera (Megabat); Example: Vampire Bat NA
6 Yinpterochiroptera (Megabat); Example: Fringed Myotis NA
7 Yinpterochiroptera (Megabat); Example: Brandt's Bat NA
8 Yangchiroptera (Microbat); Example: Horseshoe Bat NA
9 Yinpterochiroptera (Megabat); Example: Little Brown Bat NA
```

Environment:

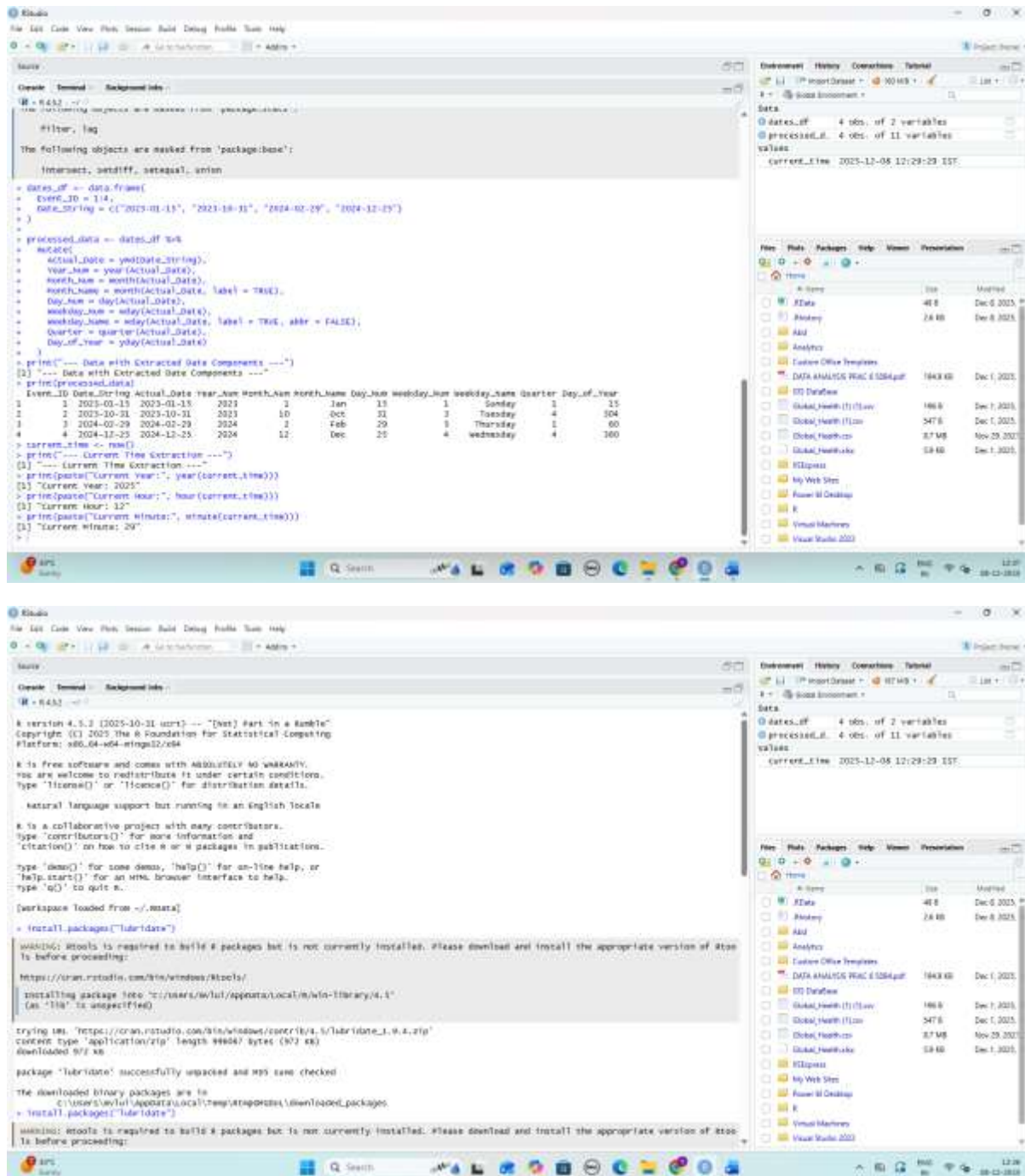
Object	Class	Size
combined_data	data.frame	30100 obs. of 2 variables
df	data.frame	100 obs. of 4 variables
duplicates_report	data.frame	18 obs. of 1 variables
long_df	data.frame	300 obs. of 3 variables
student_clean	data.frame	100 obs. of 3 variables
student_df	data.frame	100 obs. of 3 variables
student_marks	data.frame	100 obs. of 1 variables



OUTPUT:-

SETH L.U.J and SIR M.V COLLEGE

PRAC 11 12 13 14 15



15 Generating basic summaries using str() or summary() (R).

OUTPUT:-

SETH L.U.J and SIR M.V COLLEGE

PRAC 11 12 13 14 15

```

RStudio

File Edit View Plots Session Build Debug Profile Tools Help

Source
#E - R432 -
# library(dplyr)

# checking package 'dplyr'
# the following objects are loaded from 'package:stats':
#   rnorm, tag
# the following objects are loaded from 'package:base':
#   intersect, is.numeric, isequal, union

# library(dplyr)
# library(tibble)
# retail_df <- data.frame()
# #> tibble
# #> tibble
# category <- c("electronics", "home", "electronics", "clothing", "home", "clothing")
# price <- c(500.50, 45.00, 800.00, 90, 300.00, 15.00)
# in_stock <- c(TRUE, TRUE, FALSE, TRUE, TRUE, TRUE)
# rating <- c(4.5, 4.3, 4.8, 4.5, 4.2, 4.2)
# #> tibble
# #> print("Data loaded")
# [1] "Data loaded"
# #> print("Summary of df")
# [1] "Summary of df"
# #> str(retail_df)
# 'data.frame': 6 obs. of 5 variables:
# $ id : int 1 2 3 4 5 6
# $ category : chr "electronics" "home" "electronics" "clothing"
# $ price : num 500.5 45 800 90 300 15
# $ in_stock : lgl TRUE TRUE FALSE TRUE FALSE TRUE
# $ rating : num 4.5 3.8 4.8 4.5 4.2
# #> print("Output of summary() before factor conversion")
# [1] "Output of summary() before factor conversion"
# #> summary(retail_df)
#      id      category      price      in_stock      rating
# 1 id:1.00  electronics  500.50  TRUE  4.5
# 2 id:2.00  home        45.00   FALSE  3.8
# 3 id:3.00  electronics  800.00  TRUE  4.8
# 4 id:4.00  clothing     90.00   TRUE  4.5
# 5 id:5.00  home        300.00  TRUE  4.2
# 6 id:6.00  clothing     15.00  TRUE  4.2
# #> print("Output of summary() after factor conversion")
# [1] "Output of summary() after factor conversion"
# #> summary(retail_df)
#      id      category      price      in_stock      rating
# 1 id:1.00  electronics  500.50  TRUE  4.5
# 2 id:2.00  home        45.00   FALSE  3.8
# 3 id:3.00  electronics  800.00  TRUE  4.8
# 4 id:4.00  clothing     90.00   TRUE  4.5
# 5 id:5.00  home        300.00  TRUE  4.2
# 6 id:6.00  clothing     15.00  TRUE  4.2
# #> print("Average rating")
# [1] "Average rating: 4.13"
# #> print(paste("Highest Price:", max(price)))
# [1] "Highest Price: 800"

```

```

RStudio

File Edit View Plots Session Build Debug Profile Tools Help

Source
#E - R432 -
# #> print("Data loaded")
# [1] "Data loaded"
# #> print("Summary of df")
# [1] "Summary of df"
# #> str(retail_df)
# 'data.frame': 6 obs. of 5 variables:
# $ id : int 1 2 3 4 5 6
# $ category : chr "electronics" "home" "electronics" "clothing"
# $ price : num 500.5 45 800 90 300 15
# $ in_stock : lgl TRUE TRUE FALSE TRUE FALSE TRUE
# $ rating : num 4.5 3.8 4.8 4.5 4.2
# #> print("Output of summary() before factor conversion")
# [1] "Output of summary() before factor conversion"
# #> summary(retail_df)
#      id      category      price      in_stock      rating
# 1 id:1.00  electronics  500.50  TRUE  4.5
# 2 id:2.00  home        45.00   FALSE  3.8
# 3 id:3.00  electronics  800.00  TRUE  4.8
# 4 id:4.00  clothing     90.00   TRUE  4.5
# 5 id:5.00  home        300.00  TRUE  4.2
# 6 id:6.00  clothing     15.00  TRUE  4.2
# #> print("Average rating")
# [1] "Average rating: 4.13"
# #> print(paste("Highest Price:", max(price)))
# [1] "Highest Price: 800"
# #> print("Average rating")
# [1] "Average rating: 4.13"
# #> print(paste("Highest Price:", max(price)))
# [1] "Highest Price: 800"

```