

SHETH L.U.J. AND SIR M.V. COLLEGE

SUBJECT: DATA ANALYSIS WITH R

PRACTICAL NO. 11

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

OUTPUT:

The image shows two side-by-side screenshots of the RStudio interface. Both screenshots display the same R code in the 'Console' tab and show the results of the code execution.

Session 1 (Top):

```

RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Console Terminal Background Jobs
R 4.5.2 - ~/ ...
# =====#
# R Script: Reshaping data with pivot_longer() and pivot_wider()
# =====#
# Dataset: student_exam_scores.csv
# =====#
# 1. IMPORT DATA
# =====#
df <- read.csv("D:/S095 Aashka/student_exam_scores.csv")
head(df)
student_id hours_studied sleep_hours attendance_percent previous_scores exam_score
1 S001 8.0 8.8 72.1 45 30.2
2 S002 1.3 8.6 60.7 55 25.0
3 S003 4.0 8.2 73.7 86 35.8
4 S004 3.5 4.8 95.1 66 34.0
5 S005 9.1 6.4 89.8 71 40.3
6 S006 8.4 5.1 58.5 75 35.7
> print("--- 1. original data ---")
[1] "--- 1. original Data ---"
> print(head(df))
student_id hours_studied sleep_hours attendance_percent previous_scores exam_score
1 S001 8.0 8.8 72.1 45 30.2
2 S002 1.3 8.6 60.7 55 25.0
3 S003 4.0 8.2 73.7 86 35.8
4 S004 3.5 4.8 95.1 66 34.0
5 S005 9.1 6.4 89.8 71 40.3
6 S006 8.4 5.1 58.5 75 35.7
> 
> # 2. PIVOT_LONGER (wide - Long)
> # Purpose: Combine numeric variables into long format
> # =====#
> long_df <- df %>%
+ pivot_longer(
+   cols = c(hours_studied, sleep_hours, attendance_percent,
+             previous_scores, exam_score),
+   names_to = "Metric",
+   values_to = "Value"
+ )
> print("--- 2. Long Format (pivot_longer) ---")
[1] "--- 2. Long Format (pivot_longer) ---"
> print(head(long_df, 10))
# A tibble: 10 × 3
  student_id Metric    value
  <chr>      <chr>    <dbl>
1 S001       hours_studied 8.0
2 S002       hours_studied 1.3
3 S003       attendance_percent 72.1
4 S001       previous_scores 45.0
5 S001       exam_score     30.2
6 S002       hours_studied 8.6
7 S002       attendance_percent 60.7
8 S002       previous_scores 55.0
9 S002       exam_score     25.0
10 S002      exam_score     25.0
> 
> # 3. PIVOT_WIDER (Long - wide)
> # Purpose: Convert long_df BACK to original format
> # =====#
> wide_df <- long_df %>%
+ pivot_wider(
+   names_from = Metric,
+   values_from = Value
+ )
> print("--- 3. wide Format (pivot_wider) ---")
[1] "--- 3. wide Format (pivot_wider) ---"
> print(head(wide_df))

```

Session 2 (Bottom):

```

RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Console Terminal Background Jobs
R 4.5.2 - ~/ ...
# =====#
# 2. PIVOT_LONGER (wide - Long)
# =====#
# Purpose: Combine numeric variables into long format
# =====#
long_df <- df %>%
pivot_longer(
  cols = c(hours_studied, sleep_hours, attendance_percent,
            previous_scores, exam_score),
  names_to = "Metric",
  values_to = "Value"
)
> print("--- 2. Long Format (pivot_longer) ---")
[1] "--- 2. Long Format (pivot_longer) ---"
> print(head(long_df, 10))
# A tibble: 10 × 3
  student_id Metric    value
  <chr>      <chr>    <dbl>
1 S001       hours_studied 8.0
2 S002       hours_studied 1.3
3 S003       attendance_percent 72.1
4 S001       previous_scores 45.0
5 S001       exam_score     30.2
6 S002       hours_studied 8.6
7 S002       attendance_percent 60.7
8 S002       previous_scores 55.0
9 S002       exam_score     25.0
10 S002      exam_score     25.0
> 
> # 3. PIVOT_WIDER (Long - wide)
> # Purpose: Convert long_df BACK to original format
> # =====#
> wide_df <- long_df %>%
pivot_wider(
  names_from = Metric,
  values_from = Value
)
> print("--- 3. wide Format (pivot_wider) ---")
[1] "--- 3. wide Format (pivot_wider) ---"
> print(head(wide_df))

```

The screenshots show the RStudio interface with the 'Console' tab active, displaying the R code and its output. The 'Environment' and 'Plots' tabs are also visible in the top right. The bottom of each screenshot shows the Windows taskbar with various application icons.

SHETH L.U.J. AND SIR M.V. COLLEGE

SUBJECT: DATA ANALYSIS WITH R

RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Console Terminal Background Jobs

```
> # =====
> # 3. PIVOT_WIDER (long -> wide)
> # Purpose: Convert long_df BACK to original format
> #
> # wide_df <- long_df %>%
> #   pivot_wider(
> #     names_from = metric,
> #     values_from = value
> #   )
>
> print("---- 3. wide Format (pivot_wider) ----")
[1] "---- 3. wide Format (pivot_wider) ----"
> print(head(wide_df))
# # tibble: 6 x 6
student_id hours_studied sleep_hours attendance_percent previous_scores exam_score
<chr>    <dbl>      <dbl>        <dbl>        <dbl>        <dbl>
1 S001       8         8.8        72.1        45        30.2
2 S002      1.3        8.6        60.7        55        25
3 S003       4         8.2        73.7        86        35.8
4 S004      3.5        4.8        95.1        66        34
5 S005      9.1        6.4        89.8        71        40.3
6 S006      8.4        5.1        58.5        75        35.7
>
> # =====
> # 4. ADVANCED EXAMPLE:
> # Pivot where rows = student_id and columns = attendance_percent levels
> # (useful for reporting or heatmaps)
> #
> attendance_pivot <- df %>%
+   mutate(attendance_group = cut(attendance_percent,
+     breaks = c(0, 50, 75, 100),
+     labels = c("Low", "Medium", "High")) %>%
+   select(student_id, attendance_group, exam_score) %>%
+   pivot_wider(
+     names_from = attendance_group,
+     values_from = exam_score
+   )
>
> print("---- 4. Attendance Group Pivot ----")
[1] "---- 4. Attendance Group Pivot ----"
> print(head(attendance_pivot))
# # tibble: 6 x 3
student_id Medium High
<chr>    <dbl> <dbl>
1 S001       30.2 NA
2 S002       25    NA
3 S003       35.8 NA
4 S004        NA   34
5 S005        NA   40.3
6 S006       35.7 NA
```

23°C Mostly sunny

Search

Files Plots Packages Help Viewer Presentation

User Library

- askpass CRAN 1.2.1
- backports CRAN 1.5.0
- base64enc CRAN 0.1-3
- bit CRAN 4.6.0
- bit64 CRAN 4.6.0-1
- blob CRAN 1.24
- broom CRAN 1.0.10
- BSDA CRAN 1.2.2
- bslib CRAN 0.90
- cachem CRAN 1.1.0
- callr CRAN 3.7.6
- cellranger CRAN 1.1.0
- cli CRAN 3.6.5
- clipr CRAN 0.8.0
- conflicted CRAN 1.2.0
- cpp11 CRAN 0.52
- crayon CRAN 1.5.3
- curl CRAN 7.0.0
- data.table CRAN 1.17.8

Project: (None)

Import Dataset 171 MB

Environment History Connections Tutorial

tidy_df 4362 obs. of 13 variables

wide_df 200 obs. of 6 variables

values

avg_price	5016.97063037249
difference	10
group1	num [1:10] 26 25 10 34 30 23 28 29 25 ...
group2	num [1:10] 22 24 19 31 27 21 26 28 24 ...
product	0
quotient	Inf

Files Plots Packages Help Viewer Presentation

Install Update

Package Source Version

23°C Mostly sunny

23°C Mostly sunny

10:42 08-12-2025

RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Console Terminal Background Jobs

```
> # =====
> # 3. wide Format (pivot_wider) ----
[1] "---- 3. wide Format (pivot_wider) ----"
> print(head(wide_df))
# # tibble: 6 x 6
student_id hours_studied sleep_hours attendance_percent previous_scores exam_score
<chr>    <dbl>      <dbl>        <dbl>        <dbl>        <dbl>
1 S001       8         8.8        72.1        45        30.2
2 S002      1.3        8.6        60.7        55        25
3 S003       4         8.2        73.7        86        35.8
4 S004      3.5        4.8        95.1        66        34
5 S005      9.1        6.4        89.8        71        40.3
6 S006      8.4        5.1        58.5        75        35.7
>
> # =====
> # 4. ADVANCED EXAMPLE:
> # Pivot where rows = student_id and columns = attendance_percent levels
> # (useful for reporting or heatmaps)
> #
> attendance_pivot <- df %>%
+   mutate(attendance_group = cut(attendance_percent,
+     breaks = c(0, 50, 75, 100),
+     labels = c("Low", "Medium", "High")) %>%
+   select(student_id, attendance_group, exam_score) %>%
+   pivot_wider(
+     names_from = attendance_group,
+     values_from = exam_score
+   )
>
> print("---- 4. Attendance Group Pivot ----")
[1] "---- 4. Attendance Group Pivot ----"
> print(head(attendance_pivot))
# # tibble: 6 x 3
student_id Medium High
<chr>    <dbl> <dbl>
1 S001       30.2 NA
2 S002       25    NA
3 S003       35.8 NA
4 S004        NA   34
5 S005        NA   40.3
6 S006       35.7 NA
```

23°C Mostly sunny

Search

Files Plots Packages Help Viewer Presentation

User Library

- askpass CRAN 1.2.1
- backports CRAN 1.5.0
- base64enc CRAN 0.1-3
- bit CRAN 4.6.0
- bit64 CRAN 4.6.0-1
- blob CRAN 1.24
- broom CRAN 1.0.10
- BSDA CRAN 1.2.2
- bslib CRAN 0.90
- cachem CRAN 1.1.0
- callr CRAN 3.7.6
- cellranger CRAN 1.1.0
- cli CRAN 3.6.5
- clipr CRAN 0.8.0
- conflicted CRAN 1.2.0
- cpp11 CRAN 0.52
- crayon CRAN 1.5.3
- curl CRAN 7.0.0
- data.table CRAN 1.17.8

Project: (None)

Import Dataset 171 MB

Environment History Connections Tutorial

tidy_df 4362 obs. of 13 variables

wide_df 200 obs. of 6 variables

values

avg_price	5016.97063037249
difference	10
group1	num [1:10] 26 25 10 34 30 23 28 29 25 ...
group2	num [1:10] 22 24 19 31 27 21 26 28 24 ...
product	0
quotient	Inf

Files Plots Packages Help Viewer Presentation

Install Update

Package Source Version

23°C Mostly sunny

23°C Mostly sunny

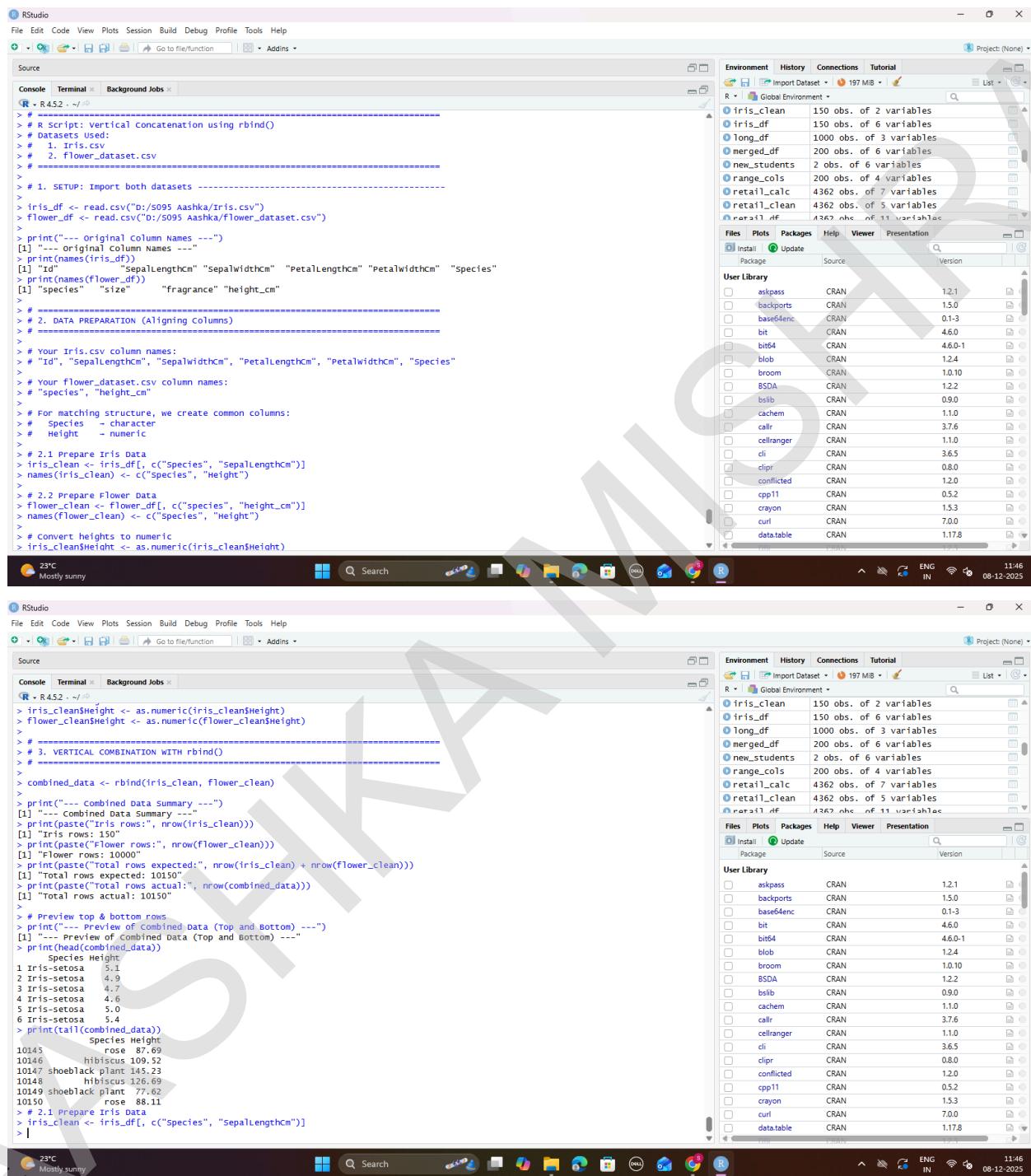
10:43 08-12-2025

NAME: AASHKA MISHRA
ROLL NO: S095

SHETH L.U.J. AND SIR M.V. COLLEGE
SUBJECT: DATA ANALYSIS WITH R

PRACTICAL NO. 12

AIM: Combining datasets vertically (concatenation) using rbind() (R).



The screenshot shows two instances of the RStudio IDE running on a Windows operating system. Both instances have identical environments and package libraries.

Environment:

- R version 4.5.2
- Global Environment contains:
 - iris_clean (150 obs. of 2 variables)
 - iris_df (150 obs. of 6 variables)
 - long_df (1000 obs. of 3 variables)
 - merged_df (200 obs. of 6 variables)
 - new_students (2 obs. of 6 variables)
 - range_cols (200 obs. of 4 variables)
 - retail_calc (4362 obs. of 7 variables)
 - retail_clean (4362 obs. of 5 variables)
 - retail_df (4362 obs. of 11 variables)

User Library:

- CRAN packages:
 - askpass (1.2.1)
 - backports (1.5.0)
 - base64enc (0.1-3)
 - bit (4.6.0)
 - bit64 (4.6.0-1)
 - blob (1.2.4)
 - broom (0.10.10)
 - BSDA (1.2.2)
 - bslib (0.9.0)
 - cachem (1.1.0)
 - callr (3.7.6)
 - cellranger (1.1.0)
 - cli (3.6.5)
 - clipr (0.8.0)
 - conflicted (1.2.0)
 - cpp11 (0.52)
 - crayon (1.5.3)
 - curl (7.0.0)
 - data.table (1.17.8)

Code in Console:

```

> # R Script: Vertical concatenation using rbind()
> # Datasets used:
> # 1. Iris.csv
> # 2. flower_dataset.csv
> #
> #
> # 1. SETUP: Import both datasets -----
>
> iris_df <- read.csv("D:/S095 Aashka/Iris.csv")
> flower_df <- read.csv("D:/S095 Aashka/flower_dataset.csv")
>
> print("--- original column names ---")
[1] " " "original column names"
> print(names(iris_df))
[1] "Id" "SepalLengthcm" "SepalWidthcm" "PetalLengthcm" "PetalWidthcm" "Species"
> print(names(flower_df))
[1] "species" "size" "Fragrance" "height_cm"
>
> # =====
> # 2. DATA PREPARATION (Aligning columns)
> #
>
> # Your Iris.csv column names:
> # "Id", "SepalLengthcm", "SepalWidthcm", "PetalLengthcm", "PetalWidthcm", "Species"
>
> # Your flower_dataset.csv column names:
> # "species", "height_cm"
>
> # For matching structure, we create common columns:
> # Species - character
> # Height - numeric
>
> # 2.1 Prepare Iris data
> iris_clean <- iris_df[, c("Species", "SepalLengthcm")]
> names(iris_clean) <- c("Species", "Height")
>
> # 2.2 Prepare Flower data
> flower_clean <- flower_df[, c("species", "height_cm")]
> names(flower_clean) <- c("Species", "Height")
>
> # Convert heights to numeric
> iris_clean$Height <- as.numeric(iris_clean$Height)

```

Below the first instance of RStudio, the taskbar shows the following icons: Weather (23°C, Mostly sunny), Start button, Search bar, File Explorer, Task View, File Manager, Control Panel, Internet Explorer, Google Chrome, Edge, File History, and RStudio icon. The system tray shows the date (08-12-2025) and time (11:46). Language settings are ENG IN.

Below the second instance of RStudio, the taskbar shows the same icons, including the RStudio icon, and the system tray shows the date (08-12-2025) and time (11:46). Language settings are ENG IN.

**SHETH L.U.J. AND SIR M.V. COLLEGE
SUBJECT: DATA ANALYSIS WITH R**

PRACTICAL NO. 13

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

OUTPUT:

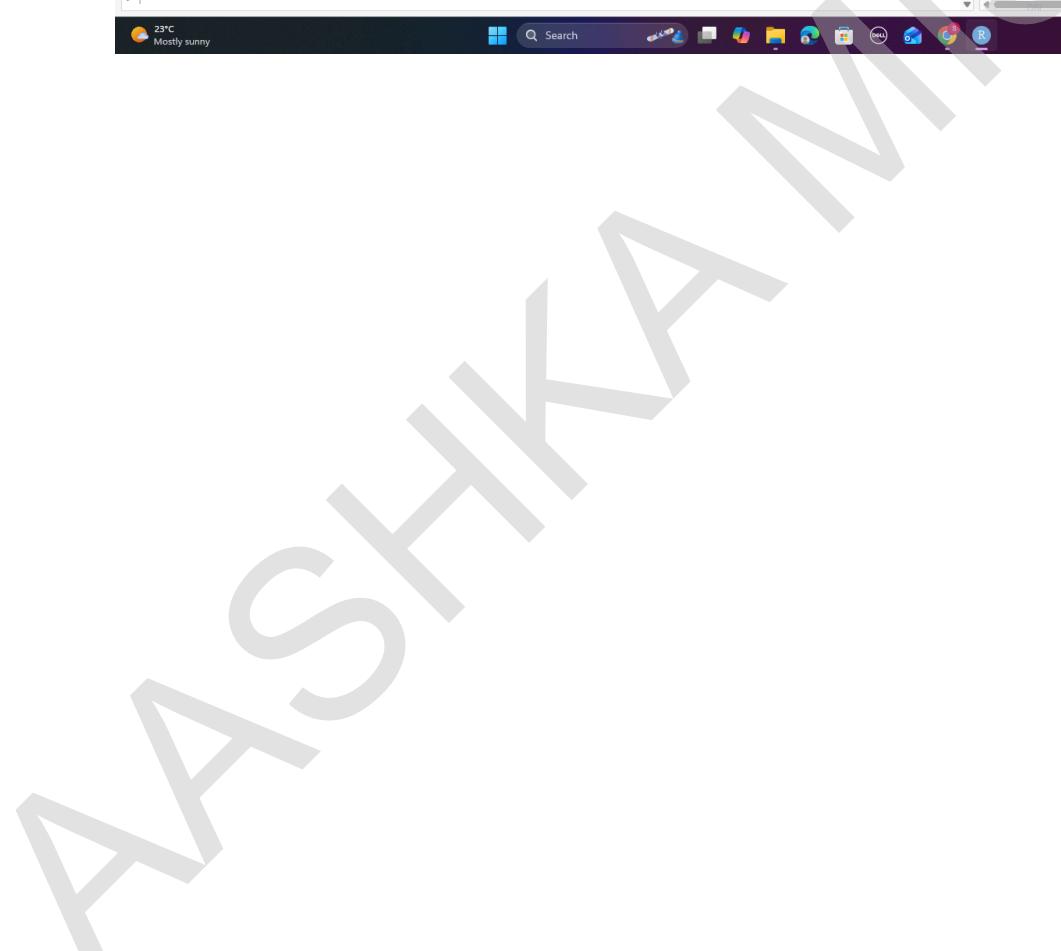
```
RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Source
Console Terminal Background Jobs
R 4.5.2 - ~
> # R Script: Identifying and Handling Duplicates using distinct()
> # dataset: student_exam_scores.csv
> #
> library(dplyr)
> #
> # 1. IMPORT YOUR DATA
> # <- read.csv("D:/s095 Aashka/student_exam_scores.csv")
> print("---- 1. Original dataset (Full Data) ----")
[1] "---- 1. Original dataset (Full Data) ----"
> print(head(df))
student_id hours_studied sleep_hours attendance_percent previous_scores exam_score
1 S001 8.0 8.8 72.1 45 30.2
2 S002 1.3 8.6 60.7 55 25.0
3 S003 4.0 8.2 73.7 86 35.8
4 S004 3.5 4.8 95.1 66 34.0
5 S005 9.1 6.4 89.8 71 40.3
6 S006 8.4 5.1 58.5 75 35.7
> print(paste("Total rows:", nrow(df)))
[1] "Total rows: 200"
>
> # 2. IDENTIFYING DUPLICATES (Exact duplicates of ENTIRE ROW)
> #
> duplicates_report <- df %>%
+   group_by(across(everything())) %>% # group by all columns
+   count() %>%
+   filter(n > 1) # keep only duplicates
> print("---- 2. Duplicate Rows Found (Exact Matches) ----")
[1] "---- 2. Duplicate Rows Found (Exact Matches) ----"
> print(duplicates_report)
# A tibble: 0 x 7
# Groups: student_id, hours_studied, sleep_hours, attendance_percent, previous_scores, exam_score [0]
# i 7 variables: student_id <chr>, hours_studied <dbl>, sleep_hours <dbl>, attendance_percent <dbl>, previous_scores <int>,
#   exam_score <dbl>, n <int>
>
> # 3. REMOVING EXACT DUPLICATES
Environment History Connections Tutorial
Project: (None)
split_list Large list (4362 elements, 802.7 KB)
split_matrix chr [1:4362, 1:2] "NA" "NA" "NA" ...
starts_with_ 200 obs. of 2 variables
student_clean 200 obs. of 2 variables
student_df 200 obs. of 6 variables
student_exam_ 200 obs. of 6 variables
tidy_df 4362 obs. of 13 variables
unique_score_ 200 obs. of 6 variables
unique_studen 200 obs. of 6 variables
Files Plots Packages Help Viewer Presentation
Install Update
User Library
askpass CRAN 1.2.1
backports CRAN 1.5.0
base64enc CRAN 0.1-3
bit CRAN 4.6.0
bit64 CRAN 4.6.0-1
blob CRAN 1.2.4
broom CRAN 1.0.10
BSDA CRAN 1.2.2
bslib CRAN 0.9.0
cachem CRAN 1.1.0
callr CRAN 3.7.6
cellranger CRAN 1.1.0
cli CRAN 3.6.5
clipr CRAN 0.8.0
conflicted CRAN 1.2.0
cpp11 CRAN 0.5.2
crayon CRAN 1.5.3
curl CRAN 7.0.0
data.table CRAN 1.17.8
Search ENG IN 11:08 08-12-2025
```

The screenshot shows an RStudio interface with a large watermark of the letter 'A' in the center. The left pane displays a script named 'R - R 4.5.2 - ~/'. The script includes comments for steps 3 and 4, which involve removing exact duplicates and duplicates based on specific columns respectively. It also prints the total number of unique students and their details. The right pane shows the 'Environment' tab with various objects listed, and the bottom right corner shows system status icons.

```
> # =====  
> # 3. REMOVING EXACT DUPLICATES  
> # =====  
>  
> df_no_dups <- df %>%  
+ distinct() # removes only rows fully identical  
>  
> print(" --- 3. Dataset After Removing Exact Duplicates ---")  
[1] " --- 3. Dataset After Removing Exact Duplicates ---"  
> print(head(df_no_dups))  
student_id hours_studied sleep_hours attendance_percent previous_scores exam_score  
1 S001 8.0 8.8 72.1 45 30.2  
2 S002 1.3 8.6 60.7 55 25.0  
3 S003 4.0 8.2 73.7 86 35.8  
4 S004 3.5 4.8 95.1 66 34.0  
5 S005 9.1 6.4 89.8 71 40.3  
6 S006 8.4 5.1 58.5 75 35.7  
> print(paste("Rows after removing duplicates:", nrow(df_no_dups)))  
[1] "Rows after removing duplicates: 200"  
>  
> # =====  
> # 4. REMOVING DUPLICATES BASED ON SPECIFIC COLUMNS  
> # =====  
>  
> # Example: Keep FIRST occurrence of each student_id  
> # If student_id repeats, only the FIRST record is kept.  
>  
> unique_students <- df %>%  
+ distinct(student_id, .keep_all = TRUE)  
>  
> print(" --- 4. unique Students (based on student_id) ---")  
[1] " --- 4. unique Students (based on student_id) ---"  
> print(head(unique_students))  
student_id hours_studied sleep_hours attendance_percent previous_scores exam_score  
1 S001 8.0 8.8 72.1 45 30.2  
2 S002 1.3 8.6 60.7 55 25.0  
3 S003 4.0 8.2 73.7 86 35.8  
4 S004 3.5 4.8 95.1 66 34.0  
5 S005 9.1 6.4 89.8 71 40.3  
6 S006 8.4 5.1 58.5 75 35.7  
> print(paste("Total unique students:", nrow(unique_students)))  
[1] "Total unique students: 200"  
>  
> # Another example: unique combination of (student_id, exam_score)
```

NAME: AASHKA MISHRA
ROLL NO: S095

SHETH L.U.J. AND SIR M.V. COLLEGE
SUBJECT: DATA ANALYSIS WITH R



RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Console Terminal Background Jobs
R > R 4.5.2 - ~
5 S005 9.1 6.4 89.8 71 40.3
6 S006 8.4 5.1 58.5 75 35.7
> print(paste("Rows after removing duplicates:", nrow(df_no_dups)))
[1] "Rows after removing duplicates: 200"
>
> # =====
> # 4. REMOVING DUPLICATES BASED ON SPECIFIC COLUMNS
> # =====
> # Example: Keep FIRST occurrence of each student_id
> # If student_id repeats, only the FIRST record is kept.
>
> unique_students <- df %>%
+ distinct(student_id, .keep_all = TRUE)
>
> print("--- 4. Unique Students (based on student_id) ---")
[1] "--- 4. Unique Students (based on student_id) ---"
> print(head(unique_students))
student_id hours_studied sleep_hours attendance_percent previous_scores exam_score
1 S001 8.0 8.8 72.1 45 30.2
2 S002 1.3 8.6 60.7 55 25.0
3 S003 4.0 8.2 73.7 86 35.8
4 S004 3.5 4.8 95.1 66 34.0
5 S005 9.1 6.4 89.8 71 40.3
6 S006 8.4 5.1 58.5 75 35.7
> print(paste("Total unique students: ", nrow(unique_students)))
[1] "Total unique students: 200"
>
> # Another Example: Unique combination of (student_id, exam_score)
> unique_score_entries <- df %>%
+ distinct(student_id, exam_score, .keep_all = TRUE)
>
> print("--- 5. unique (student_id + exam_score) combinations ---")
[1] "--- 5. unique (student_id + exam_score) combinations ---"
> print(head(unique_score_entries))
student_id hours_studied sleep_hours attendance_percent previous_scores exam_score
1 S001 8.0 8.8 72.1 45 30.2
2 S002 1.3 8.6 60.7 55 25.0
3 S003 4.0 8.2 73.7 86 35.8
4 S004 3.5 4.8 95.1 66 34.0
5 S005 9.1 6.4 89.8 71 40.3
6 S006 8.4 5.1 58.5 75 35.7

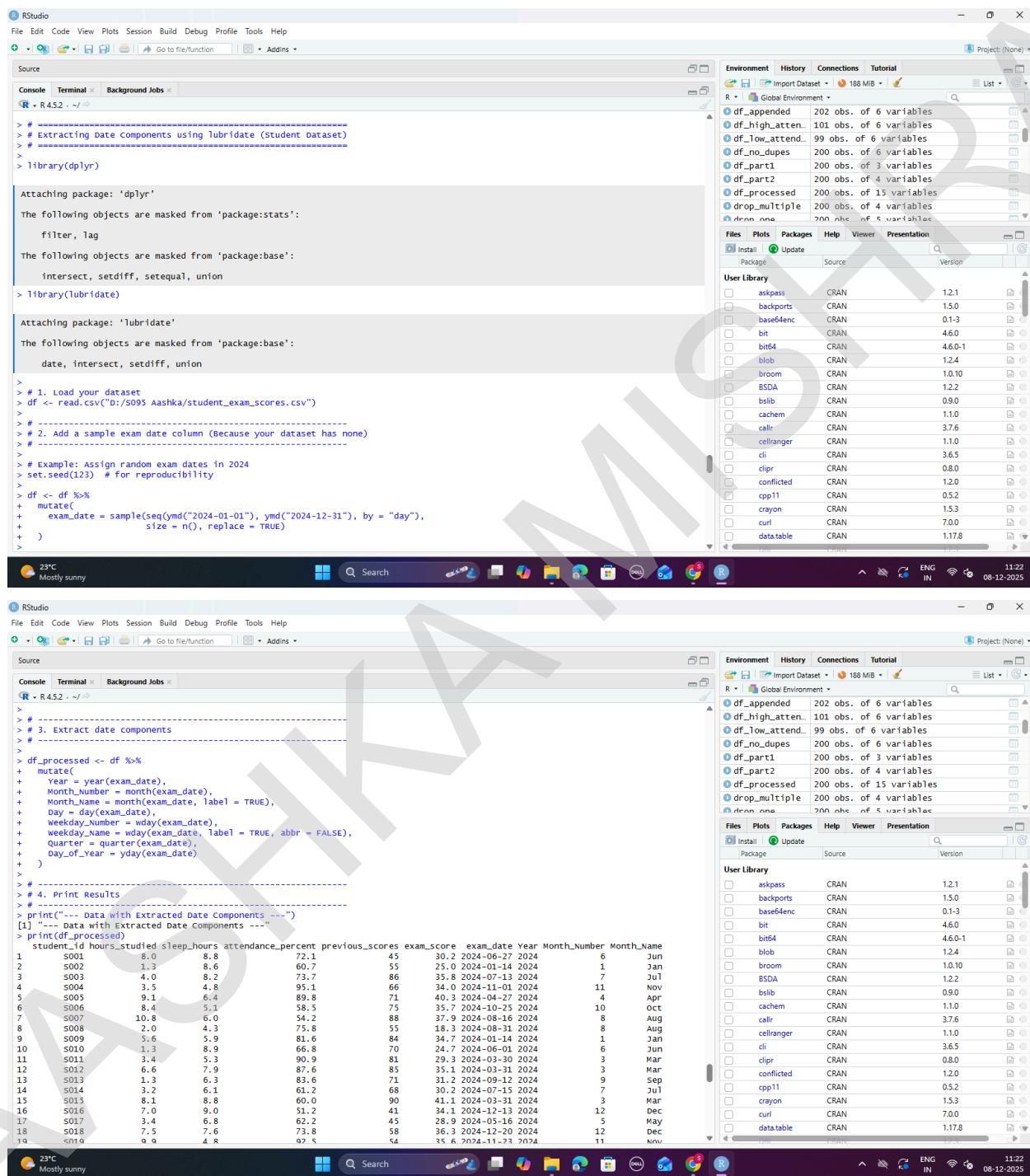
23°C Mostly sunny Search Apps Dell Google Chrome R ENG IN 08-12-2025 11:08

SHETH L.U.J. AND SIR M.V. COLLEGE
SUBJECT: DATA ANALYSIS WITH R

PRACTICAL NO. 14

AIM: Extracting date components using lubridate:: functions (R).

OUTPUT:



The screenshot shows two instances of RStudio running on a Windows desktop. Both instances have identical environments and packages loaded.

Environment View (Top Right):

- R Global Environment: df_appended (202 obs. of 6 variables), df_high_latten_ (101 obs. of 6 variables), df_low_attend_ (99 obs. of 6 variables), df_no_dups (200 obs. of 6 variables), df_part1 (200 obs. of 3 variables), df_part2 (200 obs. of 4 variables), df_processed (200 obs. of 15 variables), drop_multiple (200 obs. of 4 variables), drop_one (200 obs. of 5 variables).
- User Library: askpass (CRAN, 1.2.1), backports (CRAN, 1.5.0), base64enc (CRAN, 0.1-3), bit (CRAN, 4.6.0), bit64 (CRAN, 4.6.0-1), blob (CRAN, 1.2.4), broom (CRAN, 1.0.10), BSDA (CRAN, 1.2.2), bslib (CRAN, 0.9.0), cachem (CRAN, 1.1.0), callr (CRAN, 3.7.6), cellranger (CRAN, 1.1.0), cli (CRAN, 3.6.5), clipr (CRAN, 0.8.0), conflicted (CRAN, 1.2.0), cpp11 (CRAN, 0.52), crayon (CRAN, 1.5.3), curl (CRAN, 7.0.0), data.table (CRAN, 1.17.8).

Console View (Bottom Left):

```

# =====#
# Extracting date components using lubridate (student dataset)
# =====#
library(dplyr)

Attaching package: 'dplyr'

The following objects are masked from 'package:stats':
  filter, lag

The following objects are masked from 'package:base':
  intersect, setdiff, setequal, union

library(lubridate)

Attaching package: 'lubridate'

The following objects are masked from 'package:base':
  date, intersect, setdiff, union

# 1. Load your dataset
df <- read.csv("D:/S095 Aashka/student_exam_scores.csv")

# 2. Add a sample exam_date column (Because your dataset has none)
# Example: Assign random exam dates in 2024
set.seed(123) # for reproducibility

df <- df %>%
  mutate(
    exam_date = sample(seq(ymd("2024-01-01"), ymd("2024-12-31"), by = "day"),
                       size = n(), replace = TRUE)
  )
# 3. Extract date components
# =====#
# 4. Print Results
# =====#
print("--- Data with Extracted Date Components ---")
[1] "--- Data with Extracted Date Components ---"
print(df_processed)
  
```

Data View (Bottom Right):

student_id	hours_studied	sleep_hours	attendance_percent	previous_scores	exam_score	exam_date	Year	Month_Number	Month_Name
1	8.0	8.8	72.1	45	30.2	2024-06-27	2024	6	Jun
2	1.5	8.6	60.7	55	25.0	2024-01-12	2024	1	Jan
3	4.0	8.2	73.7	86	35.8	2024-07-13	2024	7	Jul
4	3.5	4.6	95.1	66	34.0	2024-11-01	2024	11	Nov
5	9.1	6.4	89.8	71	40.3	2024-04-27	2024	4	Apr
6	8.4	5.1	58.5	75	35.7	2024-10-25	2024	10	Oct
7	10.8	6.0	54.2	88	37.9	2024-08-16	2024	8	Aug
8	5.0	4.3	75.8	55	18.3	2024-08-31	2024	8	Aug
9	5.6	5.9	81.6	84	34.7	2024-01-14	2024	1	Jan
10	1.3	8.9	66.8	70	24.7	2024-06-01	2024	6	Jun
11	3.4	5.3	90.9	81	29.3	2024-03-30	2024	3	Mar
12	6.1	7.8	87.6	85	35.2	2024-09-01	2024	9	Sep
13	1.3	6.3	83.6	71	31.2	2024-09-12	2024	9	Sep
14	3.2	6.1	61.2	68	30.2	2024-07-15	2024	7	Jul
15	8.1	8.8	60.0	90	41.1	2024-03-31	2024	3	Mar
16	7.0	9.0	51.2	41	34.1	2024-12-13	2024	12	Dec
17	3.4	6.8	62.2	45	28.9	2024-05-16	2024	5	May
18	7.5	7.6	73.8	58	36.3	2024-12-20	2024	12	Dec
19	9.9	4.8	97.5	54	35.6	2024-11-23	2024	11	Nov

NAME: AASHKA MISHRA
ROLL NO: S095

SHETH L.U.J. AND SIR M.V. COLLEGE

SUBJECT: DATA ANALYSIS WITH R

RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Console Terminal Background Jobs

```
R > R 4.5.2 ~/ ~
[1] --- Data with Extracted Date Components ---
> print(df_processed)
```

student_id	hours_studied	sleep_hours	attendance_percent	previous_scores	exam_score	exam_date	Year	Month_Number	Month_Name
S002	8.0	8.5	72.1	45	30.2	2024-06-14	2024	6	Jun
S002	1.3	8.6	60.7	55	35.0	2024-06-14	2024	1	Jan
S003	4.0	8.2	73.7	86	35.8	2024-07-13	2024	7	Jul
S004	3.5	4.8	95.1	66	34.0	2024-11-01	2024	11	Nov
S005	9.1	6.4	89.8	71	40.3	2024-04-27	2024	4	Apr
S006	8.4	5.1	58.5	75	35.7	2024-10-25	2024	10	Oct
S007	10.8	6.0	54.2	88	37.9	2024-08-16	2024	8	Aug
S008	2.0	4.3	75.8	55	18.3	2024-03-31	2024	8	Aug
S009	5.6	5.9	81.6	84	34.7	2024-06-10	2024	1	Jan
S010	1.3	6.0	66.8	70	24.1	2024-06-10	2024	6	Jun
S011	3.4	5.3	90.9	81	29.3	2024-03-30	2024	3	Mar
S012	6.6	7.9	87.6	85	35.1	2024-03-31	2024	3	Mar
S013	1.3	6.3	83.6	71	31.2	2024-09-12	2024	9	Sep
S014	3.2	6.1	61.2	68	30.2	2024-07-15	2024	7	Jul
S015	8.1	8.8	60.0	90	41.1	2024-03-31	2024	3	Mar
S016	7.0	9.0	51.2	41	34.1	2024-12-13	2024	12	Dec
S017	3.4	6.8	62.2	45	28.9	2024-05-16	2024	5	May
S018	7.5	7.6	73.8	58	36.1	2024-06-20	2024	12	Dec
S019	9.9	4.5	92.5	54	35.6	2024-12-23	2024	11	Nov
S020	1.1	5.5	53.6	65	17.1	2024-02-26	2024	1	Jan
S021	9.9	8.8	70.7	84	46.0	2024-01-07	2024	1	Jan
S022	8.7	6.9	81.5	55	36.1	2024-05-16	2024	5	May
S023	4.7	6.7	59.7	59	29.4	2024-09-10	2024	9	Sep
S024	2.7	7.7	84.8	82	35.9	2024-07-29	2024	7	Jul
S025	11.5	4.3	74.7	77	39.2	2024-03-18	2024	3	Mar
S026	4.7	6.9	62.7	63	30.0	2024-03-21	2024	3	Mar
S027	2.0	6.5	82.8	70	29.0	2024-03-12	2024	2	Feb
S028	2.1	6.5	50.3	75	26.5	2024-12-24	2024	12	Dec
S029	10.3	4.8	87.5	73	37.2	2024-11-27	2024	11	Nov
S030	7.6	8.8	88.5	62	36.2	2024-05-22	2024	5	May
S031	9.9	4.4	55.3	67	34.5	2024-02-01	2024	2	Feb
S032	9.0	4.9	71.3	87	41.6	2024-04-18	2024	4	Apr
S033	6.9	7.0	58.8	75	38.1	2024-09-19	2024	9	Sep
S034	11.7	7.4	97.9	61	42.7	2024-11-25	2024	11	Nov
S035	5.2	5.2	75.9	62	32.0	2024-02-23	2024	1	Jan
S036	7.1	4.6	52.5	52	32.0	2024-11-24	2024	11	Nov
S037	10.1	6.5	62.5	59	44.7	2024-05-14	2024	5	May
S038	7.8	5.2	92.4	57	38.3	2024-11-04	2024	11	Nov
S039	10.5	7.0	72.8	59	39.8	2024-08-11	2024	8	Aug
S040	7.4	7.4	88.1	56	35.0	2024-06-14	2024	6	Jun

23°C Mostly sunny

Search

File Edit Code View Plots Session Build Debug Profile Tools Help

Console Terminal Background Jobs

```
R > R 4.5.2 ~/ ~
[1] --- Data with Extracted Date Components ---
> print(df_processed)
```

student_id	hours_studied	sleep_hours	attendance_percent	previous_scores	exam_score	exam_date	Year	Month_Number	Month_Name
S040	7.4	7.1	90.1	56	35.0	2024-06-14	2024	6	Jun
S041	8.8	6.1	83.4	54	34.2	2024-08-04	2024	8	Aug
S042	1.5	6.9	99.4	47	23.1	2024-10-16	2024	10	Oct
S043	3.5	6.6	79.8	86	37.2	2024-03-09	2024	3	Mar
S044	4.2	8.7	97.5	52	30.3	2024-03-12	2024	3	Mar
S045	1.9	5.0	94.6	60	26.1	2024-01-10	2024	3	Mar
S046	3.6	7.1	60.6	47	31.8	2024-02-13	2024	3	Mar
S047	2.1	5.2	86.0	87	31.6	2024-02-20	2024	5	May
S048	4.1	6.0	75.2	74	31.1	2024-07-28	2024	7	Jul
S049	8.0	7.4	91.5	88	41.2	2024-12-18	2024	12	Dec
S050	5.0	5.5	77.4	84	30.6	2024-12-12	2024	12	Dec
S051	5.1	5.6	94.9	51	28.8	2024-06-01	2024	6	Jun
S052	3.3	7.8	87.2	52	30.7	2024-10-20	2024	10	Oct
S053	3.9	4.4	63.0	87	46.1	2024-01-06	2024	10	Oct
S054	11.3	6.3	70.0	70	34.4	2024-07-10	2024	3	Mar
S055	8.1	9.0	95.4	57	37.0	2024-11-21	2024	11	Nov
S056	7.7	9.0	81.9	57	37.0	2024-11-21	2024	11	Nov
S057	2.9	4.4	88.3	86	35.2	2024-08-10	2024	8	Aug
S058	9.0	5.1	76.1	77	38.2	2024-01-16	2024	1	Jan
S059	2.8	5.3	81.3	88	33.0	2024-04-25	2024	4	Apr
S060	5.2	8.7	63.7	73	29.2	2024-04-03	2024	4	Apr
S061	11.9	8.4	53.9	78	48.6	2024-09-18	2024	9	Sep
S062	8.0	8.4	64.3	58	36.1	2024-08-22	2024	8	Aug
S063	7.1	5.8	63.6	46	27.1	2024-02-20	2024	3	Mar
S064	8.5	8.4	66.0	93	35.1	2024-12-07	2024	12	Dec
S065	10.3	8.2	77.0	52	39.5	2024-02-08	2024	2	Feb
S066	9.5	7.5	56.9	58	36.7	2024-06-07	2024	6	Jun

Day Weekday_Number Weekday_Name Quarter Day_of_Year

1	27	5	Thursday	2	179
2	14	1	Sunday	1	14
3	13	7	Saturday	3	195
4	1	6	Friday	4	306
5	27	7	Saturday	2	118
6	25	6	Friday	4	209
7	16	6	Friday	3	229
8	31	7	Saturday	3	244
9	14	1	Sunday	1	14
10	1	7	Saturday	2	153
11	30	7	Saturday	1	90
12	31	1	Sunday	1	91
13	12	5	Thursday	3	256
14	15	2	Monday	3	197

23°C Mostly sunny

Search

File Edit Code View Plots Session Build Debug Profile Tools Help

Console Terminal Background Jobs

```
R > R 4.5.2 ~/ ~
[1] --- Data with Extracted Date Components ---
> print(df_processed)
```

student_id	hours_studied	sleep_hours	attendance_percent	previous_scores	exam_score	exam_date	Year	Month_Number	Month_Name
S040	7.4	7.1	90.1	56	35.0	2024-06-14	2024	6	Jun
S041	8.8	6.1	83.4	54	34.2	2024-08-04	2024	8	Aug
S042	1.5	6.9	99.4	47	23.1	2024-10-16	2024	10	Oct
S043	3.5	6.6	79.8	86	37.2	2024-03-09	2024	3	Mar
S044	4.2	8.7	97.5	52	30.3	2024-03-12	2024	3	Mar
S045	1.9	5.0	94.6	60	26.1	2024-01-10	2024	3	Mar
S046	3.6	7.1	60.6	47	31.8	2024-02-13	2024	3	Mar
S047	2.1	5.2	86.0	87	31.6	2024-02-20	2024	5	May
S048	4.1	6.0	75.2	74	31.1	2024-07-28	2024	7	Jul
S049	8.0	7.4	91.5	88	41.2	2024-12-18	2024	12	Dec
S050	5.0	5.5	77.4	84	30.6	2024-12-12	2024	12	Dec
S051	5.1	5.6	94.9	51	28.8	2024-06-01	2024	6	Jun
S052	3.3	7.8	87.2	52	30.7	2024-10-20	2024	10	Oct
S053	3.9	4.4	63.0	87	46.1	2024-01-06	2024	10	Oct
S054	11.3	6.3	70.0	70	34.4	2024-07-10	2024	3	Mar
S055	8.1	9.0	95.4	57	37.0	2024-11-21	2024	11	Nov
S056	7.7	9.0	81.9	57	37.0	2024-11-21	2024	11	Nov
S057	2.9	4.4	88.3	86	35.2	2024-08-10	2024	8	Aug
S058	9.0	5.1	76.1	77	38.2	2024-01-16	2024	1	Jan
S059	2.8	5.3	81.3	88	33.0	2024-04-25	2024	4	Apr
S060	5.2	8.7	63.7	73	29.2	2024-04-03	2024	4	Apr
S061	11.9	8.4	53.9	78	48.6	2024-09-18	2024	9	Sep
S062	8.0	8.4	64.3	58	36.1	2024-08-22	2024	8	Aug
S063	7.1	5.8	63.6	46	27.1	2024-02-20	2024	3	Mar
S064	8.5	8.4	66.0	93	35.1	2024-12-07	2024	12	Dec
S065	10.3	8.2	77.0	52	39.5	2024-02-08	2024	2	Feb
S066	9.5	7.5	56.9	58	36.7	2024-06-07	2024	6	Jun

Day Weekday_Number Weekday_Name Quarter Day_of_Year

1	27	5	Thursday	2	179
2	14	1	Sunday	1	14
3	13	7	Saturday	3	195
4	1	6	Friday	4	306
5	27	7	Saturday	2	118
6	25	6	Friday	4	209
7	16	6	Friday	3	229
8	31	7	Saturday	3	244
9	14	1	Sunday	1	14
10	1	7	Saturday	2	153
11	30	7	Saturday	1	90
12	31	1	Sunday	1	91
13	12	5	Thursday	3	256
14	15	2	Monday	3	197

NAME: AASHKA MISHRA
ROLL NO: S095

SHETH L.U.J. AND SIR M.V. COLLEGE

SUBJECT: DATA ANALYSIS WITH R

```

SHETH L.U.J. AND SIR M.V. COLLEGE
SUBJECT: DATA ANALYSIS WITH R

R - 4.5.2 - ~
Source
Console Terminal Background Jobs
[R - 4.5.2 - ~]
14      2   Monday    3    197
15     31    Sunday    1     91
16     13    Friday    4    348
17     16   Thursday    2    137
18     20    Friday    4    355
19     23    Saturday   4    328
20     26    Friday    1    26
21     7    Sunday    1     7
22     16   Thursday   2    137
23     10   Tuesday   3    254
24     29    Monday    3    211
25     18    Monday    1    278
26     21   Thursday   1    81
27     12    Monday    1    43
28     24    Tuesday   4    359
29     27   Wednesday  4    332
30     22   Wednesday  2    143
31     1    Thursday   1    32
32     18   Thursday   2    109
33     19   Thursday   3    263
34     25    Monday    4    330
35     23    Tuesday   1    23
36     4     Monday   4    309
37     14    Tuesday   2    135
38     4     Monday   4    309
39     11    Sunday    3    224
40     14    Friday    2    166
41     4     Sunday    3    217
42     16   Wednesday  4    290
43     9     Saturday   1    69
44     12    Tuesday   1    72
45     16   Saturday   1    76
46     3     Sunday    1    63
47     20    Monday    2    141
48     28    Sunday    3    210
49     18   Wednesday  4    353
50     12   Thursday   4    347
51     1     Saturday   2    153
52     20    Sunday    4    294
53     3     Thursday   4    277
54     10    Saturday   1    41
55     30    Saturday   1    90
56     11    Monday    4    316
57     10    Saturday   3    223
58     16    Tuesday   1    16
59     25    Thursday   2    116
60     3     Wednesday  2    94
61     18   Wednesday  3    262
62     22   Thursday   3    235
63     26    Tuesday   1     86
64     7     Saturday   4    342
65     8     Thursday   1     39
66     7     Friday    2    159
[ reached 'max' / getoption("max.print") -- omitted 134 rows ]

```

Environment History Connections Tutorial

R - Global Environment

- df_appended 202 obs. of 6 variables
- df_high_atten_ 101 obs. of 6 variables
- df_low_attend_ 99 obs. of 6 variables
- df_no_dupes 200 obs. of 6 variables
- df_part1 200 obs. of 3 variables
- df_part2 200 obs. of 4 variables
- df_processed 200 obs. of 15 variables
- drop_multiple 200 obs. of 4 variables
- drop_na 200 obs. of 5 variables

Files Plots Packages Help Viewer Presentation

User Library

Package	Source	Version
askpass	CRAN	1.2.1
backports	CRAN	1.5.0
base64enc	CRAN	0.1-3
bit	CRAN	4.6.0
bit64	CRAN	4.6.0-1
blob	CRAN	1.2.4
broom	CRAN	1.0.10
BSDA	CRAN	1.2.2
bslib	CRAN	0.9.0
cachem	CRAN	1.1.0
callr	CRAN	3.7.6
cellranger	CRAN	1.1.0
cli	CRAN	3.6.5
clipr	CRAN	0.8.0
conflicted	CRAN	1.2.0
cpp11	CRAN	0.52
crayon	CRAN	1.5.3
curl	CRAN	7.0.0
data.table	CRAN	1.17.8

23°C Mostly sunny Search ENG IN 11:23 08-12-2025

Environment History Connections Tutorial

R - Global Environment

- df_appended 202 obs. of 6 variables
- df_high_atten_ 101 obs. of 6 variables
- df_low_attend_ 99 obs. of 6 variables
- df_no_dupes 200 obs. of 6 variables
- df_part1 200 obs. of 3 variables
- df_part2 200 obs. of 4 variables
- df_processed 200 obs. of 15 variables
- drop_multiple 200 obs. of 4 variables
- drop_na 200 obs. of 5 variables

Files Plots Packages Help Viewer Presentation

User Library

Package	Source	Version
askpass	CRAN	1.2.1
backports	CRAN	1.5.0
base64enc	CRAN	0.1-3
bit	CRAN	4.6.0
bit64	CRAN	4.6.0-1
blob	CRAN	1.2.4
broom	CRAN	1.0.10
BSDA	CRAN	1.2.2
bslib	CRAN	0.9.0
cachem	CRAN	1.1.0
callr	CRAN	3.7.6
cellranger	CRAN	1.1.0
cli	CRAN	3.6.5
clipr	CRAN	0.8.0
conflicted	CRAN	1.2.0
cpp11	CRAN	0.52
crayon	CRAN	1.5.3
curl	CRAN	7.0.0
data.table	CRAN	1.17.8

23°C Mostly sunny Search ENG IN 11:23 08-12-2025

SHETH L.U.J. AND SIR M.V. COLLEGE

SUBJECT: DATA ANALYSIS WITH R

PRACTICAL NO. 15

AIM: Generating basic summaries using str() or summary() (R).

OUTPUT:

```

# R Script: Generating basic summaries using str() and summary()
# Dataset: student_exam_scores.csv
# -----
# 1. SETUP: Load Dataset
df <- read.csv("D:/S095 Aashka/student_exam_scores.csv")
# print("Data Loaded ---")
# --- Data Loaded ---
# Preview first rows
student_id hours_studied sleep_hours attendance_percent previous_scores exam_score
1 S001 8.4 7.8 73.7 45 30.2
2 S002 1.3 8.6 60.7 55 25.0
3 S003 4.0 8.2 73.7 86 35.8
4 S004 3.5 4.8 95.1 66 34.0
5 S005 9.1 6.4 89.8 71 40.3
6 S006 8.4 5.1 58.5 75 35.7
# -----
# 2. USING str() - Structure of Dataset
# -----
# print("----- OUTPUT OF str() -----")
# [1] "----- OUTPUT OF str() -----"
str(df)
'data.frame': 200 obs. of 6 variables:
$ student_id : chr "S001" "S003" "S004" ...
$ hours_studied : num 8 1.3 4 3.5 9.1 8.4 10.8 2 5.6 1.3 ...
$ sleep_hours : num 8.8 8.6 8.2 4.8 6.4 5.1 6.4 3 5.9 8.9 ...
$ attendance_percent: num 72.1 60.7 73.7 95.1 89.8 58.5 54.2 75.8 81.6 66.8 ...
$ previous_scores : int 45 55 88 66 71 75 88 55 84 70 ...
$ exam_score : num 30.2 25 35.8 34 40.3 35.7 37.9 18.3 34.7 24.7 ...
# -----
# 3. USING summary() - Summary statistics
# -----
# print("----- OUTPUT OF summary() -----")
# [1] "----- OUTPUT OF summary() -----"
summary(df)
student_id    hours_studied   sleep_hours   attendance_percent previous_scores   exam_score
Length:200      Min. : 1.000   Min. :4.000   Min. :50.30       Min. :40.00      Min. :17.10
Class :character  1st Qu.: 3.500   1st Qu.:14.000   1st Qu.:50.30       1st Qu.:40.00      1st Qu.:29.50
Median : 6.150   Median :6.700   Median :6.200   Median :67.5        Median :67.5        Median :34.05
Mean   : 6.325   Mean   :6.622   Mean   :74.83        Mean   :66.8        Mean   :33.95
3rd Qu.: 9.000   3rd Qu.:8.025   3rd Qu.:87.42       3rd Qu.:80.0        3rd Qu.:38.75
Max.   :12.000   Max.   :9.000   Max.   :100.00       Max.   :95.0        Max.   :51.30
# -----
# 4. IMPROVING summary() BY CONVERTING ID TO FACTOR (optional)
# Reason: student_id is categorical, not numeric
# -----
df$student_id <- as.factor(df$student_id)
# print("----- OUTPUT OF summary() After Converting student_id to Factor -----")
# [1] "----- OUTPUT OF summary() After Converting student_id to Factor -----"
summary(df)
student_id    hours_studied   sleep_hours   attendance_percent previous_scores   exam_score
S001 : 1  Min. : 1.000   Min. :4.000   Min. :50.30       Min. :40.00      Min. :17.10
S002 : 1  1st Qu.: 3.500   1st Qu.:14.000   1st Qu.:50.30       1st Qu.:40.00      1st Qu.:29.50
S003 : 1  Median : 6.150   Median :6.700   Median :67.5        Median :67.5        Median :34.05
S004 : 1  Mean   : 6.325   Mean   :6.622   Mean   :74.83        Mean   :66.8        Mean   :33.95
S005 : 1  3rd Qu.: 9.000   3rd Qu.:8.025   3rd Qu.:87.42       3rd Qu.:80.0        3rd Qu.:38.75
S006 : 1  Max.   :12.000   Max.   :9.000   Max.   :100.00       Max.   :95.0        Max.   :51.30
(Other):194
# -----
# 5. Accessing Specific Summaries
# mean(df$exam_score)
avg_exam_score <- mean(df$exam_score)
max_study_hours <- max(df$hours_studied)
# print(paste("Average Exam Score:", avg_exam_score))
# [1] "Average Exam Score: 33.95"
# print(paste("Maximum Hours Studied:", max_study_hours))
# [1] "Maximum Hours Studied: 12"

```

NAME: AASHKA MISHRA
ROLL NO: S095

SHETH L.U.J. AND SIR M.V. COLLEGE

SUBJECT: DATA ANALYSIS WITH R

RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Console Terminal Background Jobs
R + R4.52 - ~
> # 3. USING summary() - Summary Statistics
> print("---- OUTPUT OF summary() ----")
[1] "---- OUTPUT OF summary() ----"
> summary(df)
student_id hours_studied sleep_hours attendance_percent previous_scores exam_score
Length:200 Min. : 1.000 Min. :4.000 Min. : 50.30 Min. :40.0 Min. :17.10
Class :character 1st Qu.: 3.500 1st Qu.:5.300 1st Qu.: 62.20 1st Qu.:54.0 1st Qu.:29.50
Mode :character Median : 6.150 Median :6.700 Median : 75.25 Median :67.5 Median :34.05
Mean : 6.325 Mean :6.622 Mean : 74.83 Mean :66.8 Mean :33.95
3rd Qu.: 9.000 3rd Qu.:8.025 3rd Qu.: 87.42 3rd Qu.:80.0 3rd Qu.:38.75
Max. :12.000 Max. :9.000 Max. :100.00 Max. :95.0 Max. :51.30
>
> # 4. IMPROVING summary() BY CONVERTING ID TO FACTOR (optional)
> # Reason: student_id is categorical, not numeric
> # ======
> df\$student_id <- as.factor(df\$student_id)
>
> print("---- OUTPUT OF summary() After Converting student_id to Factor ----")
[1] "---- OUTPUT OF summary() After converting student_id to Factor ----"
> summary(df)
student_id hours_studied sleep_hours attendance_percent previous_scores exam_score
S001 : 1 Min. : 1.000 Min. :4.000 Min. : 50.30 Min. :40.0 Min. :17.10
S002 : 1 1st Qu.: 3.500 1st Qu.:5.300 1st Qu.: 62.20 1st Qu.:54.0 1st Qu.:29.50
S003 : 1 Median : 6.150 Median :6.700 Median : 75.25 Median :67.5 Median :34.05
S004 : 1 Mean : 6.325 Mean :6.622 Mean : 74.83 Mean :66.8 Mean :33.95
S005 : 1 3rd Qu.: 9.000 3rd Qu.:8.025 3rd Qu.: 87.42 3rd Qu.:80.0 3rd Qu.:38.75
S006 : 1 Max. :12.000 Max. :9.000 Max. :100.00 Max. :95.0 Max. :51.30
>
> # ======
> # 5. Accessing Specific Summaries
> # ======
> avg_exam_score <- mean(df\$exam_score)
> max_study_hours <- max(df\$hours_studied)
>
> print(paste("Average Exam Score: ", avg_exam_score))
[1] "Average Exam Score: 33.95"
> print(paste("Maximum Hours Studied: ", max_study_hours))
[1] "Maximum Hours Studied: 12"
> |

23°C
Mostly sunny