

## MVLU COLLEGE

### PRACTICAL NO :- 07

AIM :- Selecting and dropping variables using select() in R. import dataset.

#### CODE :-

```
library(dplyr)
```

```
marks <- read.csv("C:/Users/itlab/Downloads/S100/Cleaned_Data_Science_Student_Marks.csv")
```

```
print("--- Original Dataset (First 3 rows) ---")
```

```
print(head(marks, 3))
```

```
# 2. SELECTING VARIABLES (Keeping Columns)
```

```
#method 1
```

```
# Scenario: We want only student_id, python_marks, and english_marks
```

```
selected_cols <- marks %>%
```

```
  select(student_id, python_marks, english_marks)
```

```
print("--- Selected Specific Columns ---")
```

```
print(head(selected_cols, 3))
```

```
#methods 2
```

```
# Scenario: Select columns from sql_marks to python_marks
```

```
range_cols <- marks %>%
```

```
  select(sql_marks:python_marks)
```

```
print("--- Selected Range of Columns (sql_marks to python_marks) ---")
```

```
print(head(range_cols, 3))
```

```
#method 3
```

```
# Scenario: Select all columns starting with "p" (python_marks, power_bi_marks)
```

```
starts_with_p <- marks %>%
```

```
  select(starts_with("p"))
```

```
print("--- Selected columns starting with 'p' ---")
```

```
print(head(starts_with_p, 3))
```

```
# 3. DROPPING VARIABLES (Removing Columns)
```

```
#Method A
```

```
# Scenario: Remove the 'location' column
```

```
dropped_one <- marks %>%
```

```
  select(-location)
```

```
print("--- Dataset with 'location' dropped ---")
```

```
print(names(dropped_one))
```

```
#Method B
```

# MVLU COLLEGE

```
# Scenario: Remove 'excel_marks' and 'power_bi_marks'  
dropped_multiple <- marks %>%  
  select(-excel_marks, -power_bi_marks)  
print("--- Dataset with 'excel_marks' and 'power_bi_marks' dropped ---")  
print(names(dropped_multiple))
```

## #Method C

```
# Scenario: Remove all columns from 'sql_marks' to 'python_marks'  
dropped_range <- marks %>%  
  select(-(sql_marks:python_marks))  
print("--- Dataset with columns 'sql_marks' to 'python_marks' dropped ---")  
print(names(dropped_range))
```

The screenshot shows the RStudio interface with the following sections:

- Console:** Displays the R code and its output. The output shows the removal of specific columns from the dataset.
- Environment:** Shows the global environment with various objects listed, such as `dropped\_one`, `dropped\_range`, `final\_dataset`, etc.
- File Browser:** Shows the project structure with files like `Home.R` and other R scripts.

# MVLU COLLEGE

The screenshot shows the RStudio interface. The top menu bar includes File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, Help, and Addins. The title bar says "RStudio". The main area has tabs for Console, Terminal, and Background Jobs. The Console tab displays R code and its output. The code involves selecting columns from various datasets (sql\_marks, python\_marks, excel\_marks) and dropping variables. The output shows data frames for each method and scenario. The Environment pane on the right lists objects in the Global Environment, such as dropped\_one, dropped\_range, final\_dataset, etc., with their respective sizes. The bottom taskbar shows system icons like battery level (35°C), search, file explorer, and browser.

```
R > 4.52 - ~/r
2      5      89      82
3      6      99      77
> #method 2
> # Scenario: select columns from sql_marks to python_marks
> range_cols <- marks %>
+   select(sql_marks$python_marks)
> print(head(range_cols, 3))
  sql_marks excel_marks python_marks
1     95      99      87
2     99      95      89
3     72      70      99
>
> #method 3
> # Scenario: Select all columns starting with "p" (python_marks, power_bi_marks)
> starts_with_p <- marks %>
+   select(starts_with("p"))
> print(head(starts_with_p, 3))
  python_marks power_bi_marks
1       87          82
2       89          86
3       99          79
> # 3. DROPPING VARIABLES (Removing Columns)
> #Method A
> # Scenario: Remove the 'location' column
> dropped_one <- marks %>
+   select(-location)
> print(names(dropped_one))
[1] "student_id"    "age"        "sql_marks"    "excel_marks"  "python_marks" "power_bi_marks"
[7] "english_marks"
> #Method B
> # Scenario: Remove 'excel_marks' and 'power_bi_marks'
> dropped_multiple <- marks %>
+   select(-excel_marks, -power.bi_marks)
> print(names(dropped_multiple))
[1] "student_id"    "location"    "age"        "sql_marks"    "python_marks" "english_marks"
> #Method C
> # Scenario: Remove all columns from 'sql_marks' to 'python_marks'
> dropped_range <- marks %>
+   select(-(sql_marks:python_marks))
> print(names(dropped_range))
[1] "student_id"    "location"    "age"        "power.bi_marks" "english_marks"
> |
```