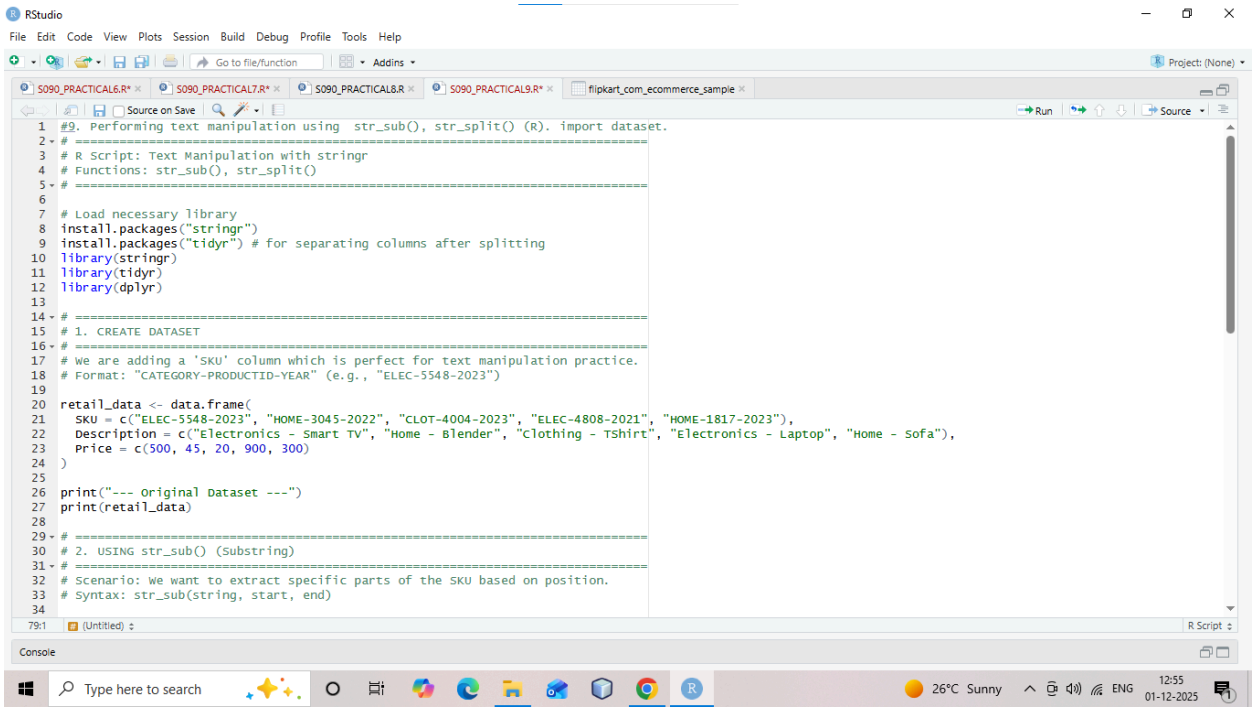
SHETH L.U.J. AND SIR M.V. COLLEGE

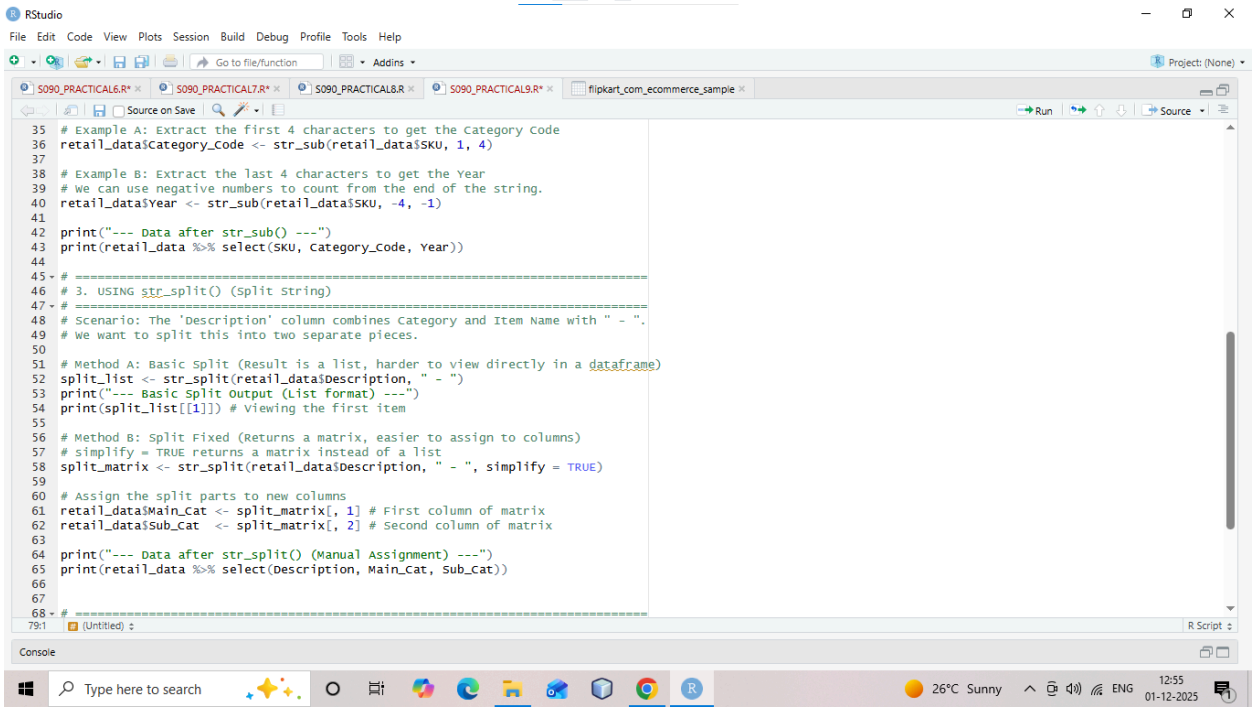
PRACTICAL NO 9

AIM: Performing text manipulation using str_sub(), str_split() (R). import dataset.

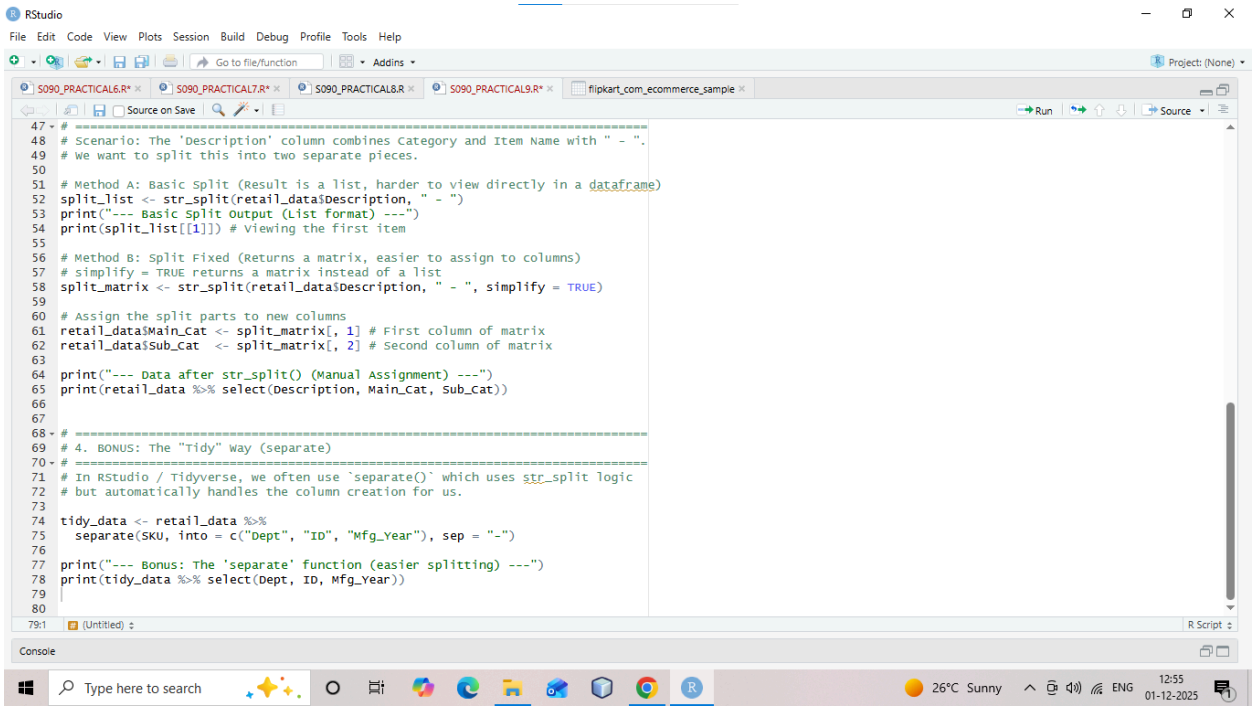
CODE:



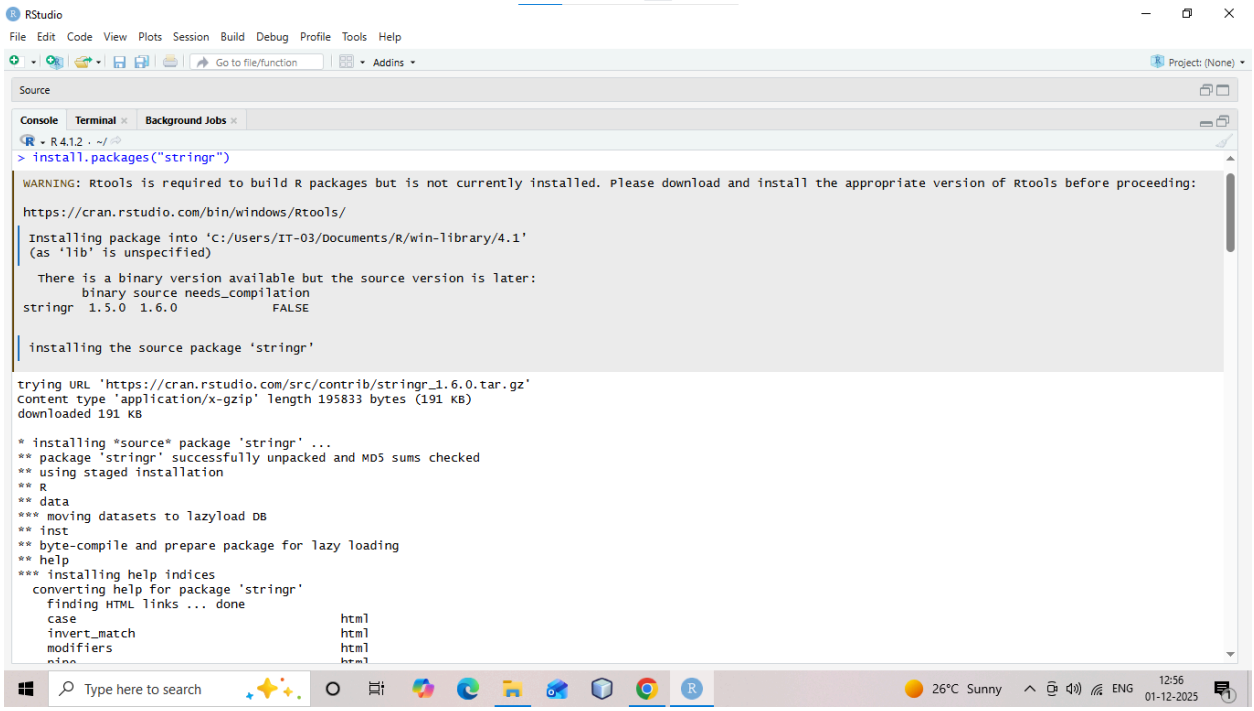
```r
#9. Performing text manipulation using  str_sub(), str_split() (R). import dataset.
# ================================================================================
# R Script: Text Manipulation with stringr
# Functions: str_sub(), str_split()
# ================================================================================

# Load necessary library
install.packages("stringr")
install.packages("tidyr") # for separating columns after splitting
library(stringr)
library(tidyr)
library(dplyr)

# ================================================================================
# 1. CREATE DATASET
# ================================================================================
# We are adding a 'SKU' column which is perfect for text manipulation practice.
# Format: "CATEGORY-PRODUCTID-YEAR" (e.g., "ELEC-5548-2023")

retail_data <- data.frame(
  SKU = c("ELEC-5548-2023", "HOME-3045-2022", "CLOT-4004-2023", "ELEC-4808-2021", "HOME-1817-2023"),
  Description = c("Electronics - Smart TV", "Home - Blender", "Clothing - TShirt", "Electronics - Laptop", "Home - Sofa"),
  Price = c(500, 45, 20, 900, 300)
)

print("--- Original Dataset ---")
print(retail_data)

# ================================================================================
# 2. USING str_sub() (Substring)
# ================================================================================
# Scenario: We want to extract specific parts of the SKU based on position.
# Syntax: str_sub(string, start, end)
```

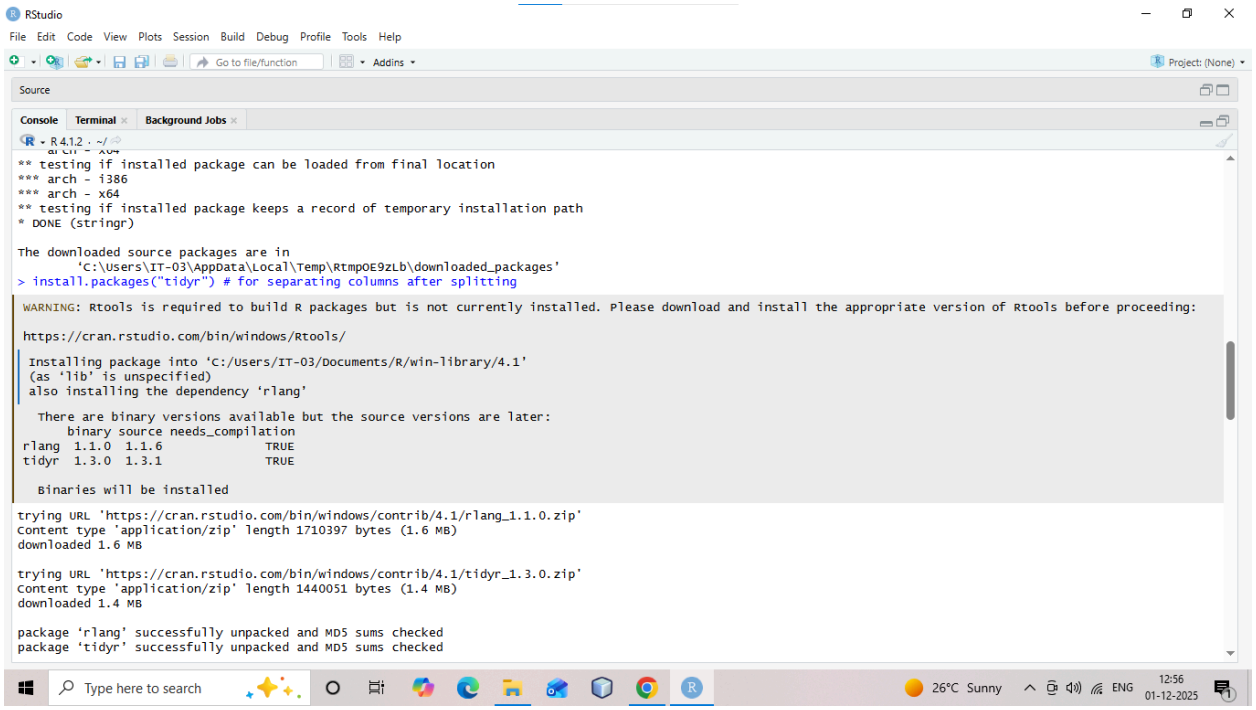


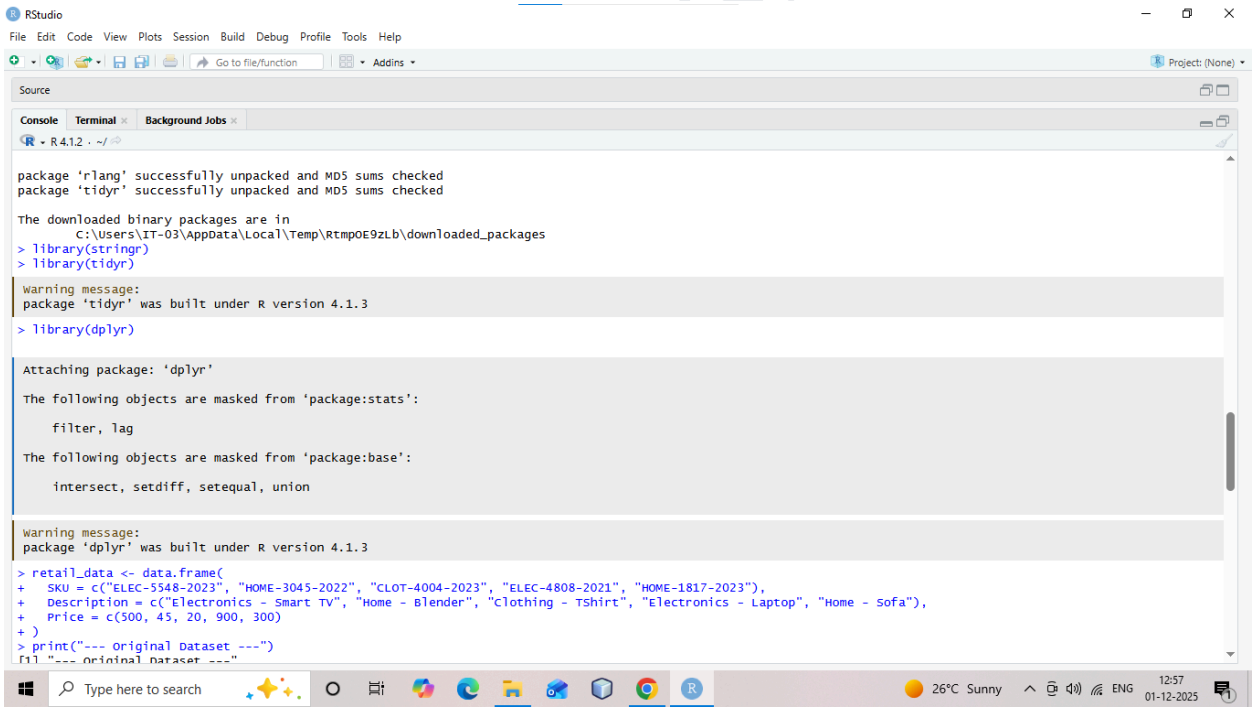
```r
# Example A: Extract the first 4 characters to get the Category Code
retail_data$Category_Code <- str_sub(retail_data$SKU, 1, 4)

# Example B: Extract the last 4 characters to get the Year
# We can use negative numbers to count from the end of the string.
retail_data$Year <- str_sub(retail_data$SKU, -4, -1)

print("--- Data after str_sub() ---")
print(retail_data %>% select(SKU, Category_Code, Year))

# ================================================================================
# 3. USING str_split() (Split String)
# ================================================================================
# Scenario: The 'Description' column combines Category and Item Name with " - ".
# We want to split this into two separate pieces.

# Method A: Basic Split (Result is a list, harder to view directly in a dataframe)
split_list <- str_split(retail_data$Description, " - ")
print("--- Basic Split Output (List format) ---")
print(split_list[[1]]) # Viewing the first item

# Method B: Split Fixed (Returns a matrix, easier to assign to columns)
# simplify = TRUE returns a matrix instead of a list
split_matrix <- str_split(retail_data$Description, " - ", simplify = TRUE)

# Assign the split parts to new columns
retail_data$Main_Cat <- split_matrix[, 1] # First column of matrix
retail_data$Sub_Cat  <- split_matrix[, 2] # Second column of matrix

print("--- Data after str_split() (Manual Assignment) ---")
print(retail_data %>% select(Description, Main_Cat, Sub_Cat))


# ================================================================================
```


RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
S090_PRACTICAL6.R*
S090_PRACTICAL7.R*
S090_PRACTICAL8.R
S090_PRACTICAL9.R*
flipkart_com_ecommerce_sample
# ==========
# Scenario: The 'Description' column combines Category and Item Name with " - ".
# We want to split this into two separate pieces.
# Method A: Basic Split (Result is a list, harder to view directly in a dataframe)
split_list <- str_split(retail_data$Description, " - ")
print("--- Basic Split Output (List format) ---")
print(split_list[[1]]) # Viewing the first item
# Method B: Split Fixed (Returns a matrix, easier to assign to columns)
# simplify = TRUE returns a matrix instead of a list
split_matrix <- str_split(retail_data$Description, " - ", simplify = TRUE)
# Assign the split parts to new columns
retail_data$Main_Cat <- split_matrix[, 1] # First column of matrix
retail_data$Sub_Cat <- split_matrix[, 2] # Second column of matrix
print("--- Data after str_split() (Manual Assignment) ---")
print(retail_data %>% select(Description, Main_Cat, Sub_Cat))
# ==========
# 4. BONUS: The "Tidy" Way (separate)
# ==========
# In RStudio / Tidyverse, we often use `separate()` which uses str_split logic
# but automatically handles the column creation for us.
tidy_data <- retail_data %>%
separate(SKU, into = c("Dept", "ID", "Mfg_Year"), sep = "-")
print("--- Bonus: The 'separate' function (easier splitting) ---")
print(tidy_data %>% select(Dept, ID, Mfg_Year))
Console
Type here to search
26°C Sunny
ENG
12:55
01-12-2025

OUTPUT:


RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Source
Console
Terminal
Background Jobs
R 4.1.2 · ~/
> install.packages("stringr")
WARNING: Rtools is required to build R packages but is not currently installed. Please download and install the appropriate version of Rtools before proceeding:
https://cran.rstudio.com/bin/windows/Rtools/
Installing package into 'C:/Users/IT-03/Documents/R/win-library/4.1'
(as 'lib' is unspecified)
There is a binary version available but the source version is later:
binary source needs_compilation
stringr 1.5.0 1.6.0 FALSE
installing the source package 'stringr'
trying URL 'https://cran.rstudio.com/src/contrib/stringr_1.6.0.tar.gz'
Content type 'application/x-gzip' length 195833 bytes (191 KB)
downloaded 191 KB
* installing *source* package 'stringr' ...
** package 'stringr' successfully unpacked and MD5 sums checked
** using staged installation
** R
** data
*** moving datasets to lazyload DB
** inst
** byte-compile and prepare package for lazy loading
** help
*** installing help indices
converting help for package 'stringr'
finding HTML links ... done
case html
invert_match html
modifiers html
Type here to search
26°C Sunny
ENG
12:56
01-12-2025


RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Source
Console Terminal Background Jobs
** testing if installed package can be loaded from final location
*** arch - i386
*** arch - x64
** testing if installed package keeps a record of temporary installation path
* DONE (stringr)
The downloaded source packages are in
'C:\Users\IT-03\AppData\Local\Temp\RtmpOE9zLb\downloaded_packages'
> install.packages("tidyr") # for separating columns after splitting
WARNING: Rtools is required to build R packages but is not currently installed. Please download and install the appropriate version of Rtools before proceeding:
https://cran.rstudio.com/bin/windows/Rtools/
Installing package into 'C:/Users/IT-03/Documents/R/win-library/4.1'
(as 'lib' is unspecified)
also installing the dependency 'rlang'
There are binary versions available but the source versions are later:
binary source needs_compilation
rlang 1.1.0 1.1.6 TRUE
tidyr 1.3.0 1.3.1 TRUE
Binaries will be installed
trying URL 'https://cran.rstudio.com/bin/windows/contrib/4.1/rlang_1.1.0.zip'
Content type 'application/zip' length 1710397 bytes (1.6 MB)
downloaded 1.6 MB
trying URL 'https://cran.rstudio.com/bin/windows/contrib/4.1/tidyr_1.3.0.zip'
Content type 'application/zip' length 1440051 bytes (1.4 MB)
downloaded 1.4 MB
package 'rlang' successfully unpacked and MD5 sums checked
package 'tidyr' successfully unpacked and MD5 sums checked
Type here to search
26°C Sunny
ENG
12:56
01-12-2025


RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Source
Console Terminal Background Jobs
package 'rlang' successfully unpacked and MD5 sums checked
package 'tidyr' successfully unpacked and MD5 sums checked
The downloaded binary packages are in
C:\Users\IT-03\AppData\Local\Temp\RtmpOE9zLb\downloaded_packages
> library(stringr)
> library(tidyr)
Warning message:
package 'tidyr' was built under R version 4.1.3
> 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
Warning message:
package 'dplyr' was built under R version 4.1.3
> retail_data <- data.frame(
+ SKU = c("ELEC-5548-2023", "HOME-3045-2022", "CLOT-4004-2023", "ELEC-4808-2021", "HOME-1817-2023"),
+ Description = c("Electronics - Smart TV", "Home - Blender", "Clothing - TShirt", "Electronics - Laptop", "Home - Sofa"),
+ Price = c(500, 45, 20, 900, 300)
+)
> print("--- Original Dataset ---")
[1] "--- Original Dataset ---"
Type here to search
26°C Sunny
ENG
12:57
01-12-2025