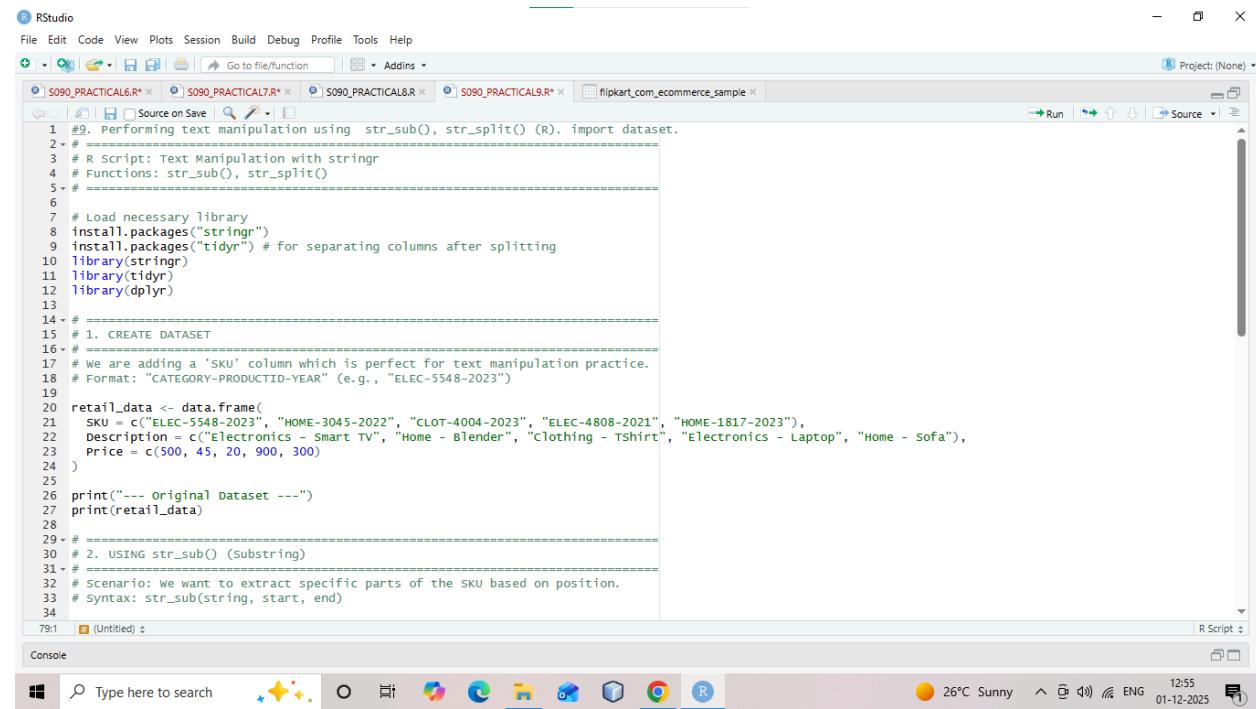


SHETH L.U.J. AND SIR M.V. COLLEGE

PRACTICAL NO 9

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

CODE:



RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Project: (None)

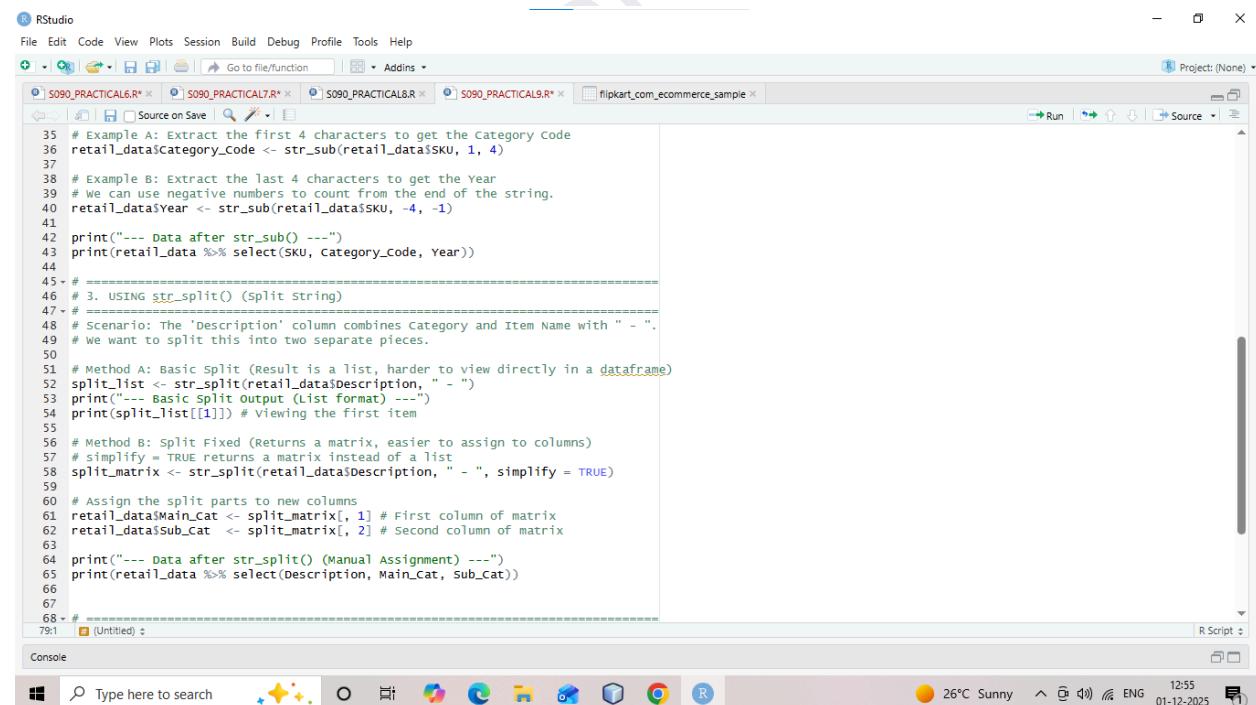
S090_PRACITCAL6.R* S090_PRACITCAL7.R* S090_PRACITCAL8.R* S090_PRACITCAL9.R* flipkart_com_ecommerce_sample

```
1 # 9. Performing text manipulation using str_sub(), str_split() (R). import dataset.
2 # -----
3 # R Script: Text Manipulation with stringr
4 # Functions: str_sub(), str_split()
5 # -----
6
7 # Load necessary library
8 install.packages("stringr")
9 install.packages("tidyverse") # for separating columns after splitting
10 library(stringr)
11 library(tidyverse)
12 library(dplyr)
13
14 # -----
15 # 1. CREATE DATASET
16 # -----
17 # we are adding a 'SKU' column which is perfect for text manipulation practice.
18 # Format: "CATEGORY-PRODUCTID-YEAR" (e.g., "ELEC-5548-2023")
19
20 retail_data <- data.frame(
21   SKU = c("ELEC-5548-2023", "HOME-3045-2022", "CLOT-4004-2023", "ELEC-4808-2021", "HOME-1817-2023"),
22   Description = c("Electronics - Smart TV", "Home - Blender", "Clothing - Tshirt", "Electronics - Laptop", "Home - Sofa"),
23   Price = c(500, 45, 20, 900, 300)
24 )
25
26 print("--- Original Dataset ---")
27 print(retail_data)
28
29 # -----
30 # 2. USING str_sub() (Substring)
31 # -----
32 # Scenario: We want to extract specific parts of the SKU based on position.
33 # Syntax: str_sub(string, start, end)
34
```

79:1 (Untitled) R Script

Console

Type here to search 26°C Sunny 12:55 01-12-2025



RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Project: (None)

S090_PRACITCAL6.R* S090_PRACITCAL7.R* S090_PRACITCAL8.R* S090_PRACITCAL9.R* flipkart_com_ecommerce_sample

```
35 # Example A: Extract the first 4 characters to get the category code
36 retail_data$category_code <- str_sub(retail_data$SKU, 1, 4)
37
38 # Example B: Extract the last 4 characters to get the Year
39 # We can use negative numbers to count from the end of the string.
40 retail_data$Year <- str_sub(retail_data$SKU, -4, -1)
41
42 print("--- Data after str_sub() ---")
43 print(retail_data %>% select(SKU, category_code, Year))
44
45 # -----
46 # 3. USING str_split() (Split string)
47 # -----
48 # Scenario: The 'Description' column combines Category and Item Name with " - ".
49 # We want to split this into two separate pieces.
50
51 # Method A: Basic Split (Result is a list, harder to view directly in a dataframe)
52 split_list <- str_split(retail_data$Description, " - ")
53 print(" --- Basic Split Output (List format) --- ")
54 print(split_list[[1]]) # Viewing the first item
55
56 # Method B: Split Fixed (Returns a matrix, easier to assign to columns)
57 # simplify = TRUE returns a matrix instead of a list
58 split_matrix <- str_split(retail_data$Description, " - ", simplify = TRUE)
59
60 # Assign the split parts to new columns
61 retail_data>Main_Cat <- split_matrix[, 1] # First column of matrix
62 retail_data>Sub_Cat <- split_matrix[, 2] # Second column of matrix
63
64 print(" --- Data after str_split() (Manual Assignment) --- ")
65 print(retail_data %>% select>Description, Main_Cat, Sub_Cat)
66
67
68 # -----
```

79:1 (Untitled) R Script

Console

Type here to search 26°C Sunny 12:55 01-12-2025

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

The screenshot shows the RStudio interface with several tabs open at the top: S090_PRACICAL6.R*, S090_PRACICAL7.R*, S090_PRACICAL8.R*, S090_PRACICAL9.R*, and flipkart_com_ecommerce_sample. The main editor area contains the following R code:

```
47 # 
48 # Scenario: The 'Description' column combines Category and Item Name with " - ".
49 # we want to split this into two separate pieces.
50
51 # Method A: Basic split (Result is a list, harder to view directly in a dataframe)
52 split_list <- str_split(retail_data$description, " - ")
53 print("--- basic split output (List format) ---")
54 print(split_list[[1]]) # viewing the first item
55
56 # Method B: Split Fixed (Returns a matrix, easier to assign to columns)
57 # simplify = TRUE returns a matrix instead of a list
58 split_matrix <- str_split(retail_data$description, " - ", simplify = TRUE)
59
60 # Assign the split parts to new columns
61 retail_data$Main_Cat <- split_matrix[, 1] # First column of matrix
62 retail_data$Sub_Cat <- split_matrix[, 2] # Second column of matrix
63
64 print(" --- data after str_split() (Manual Assignment) ---")
65 print(retail_data %>% select(Description, Main_Cat, Sub_Cat))
66
67
68 # =====
69 # 4. BONUS: The "Tidy" way (Separate)
70 # =====
71 # In RStudio / Tidyverse, we often use `separate()` which uses str_split logic
72 # but automatically handles the column creation for us.
73
74 tidy_data <- retail_data %>%
75   separate(SKU, into = c("Dept", "ID", "Mfg_Year"), sep = "-")
76
77 print(" --- Bonus: The 'separate' function (easier splitting) ---")
78 print(tidy_data %>% select(Dept, ID, Mfg_Year))
79
80
81
```

OUTPUT:

The screenshot shows the RStudio interface with the 'Console' tab selected. The output window displays the following text:

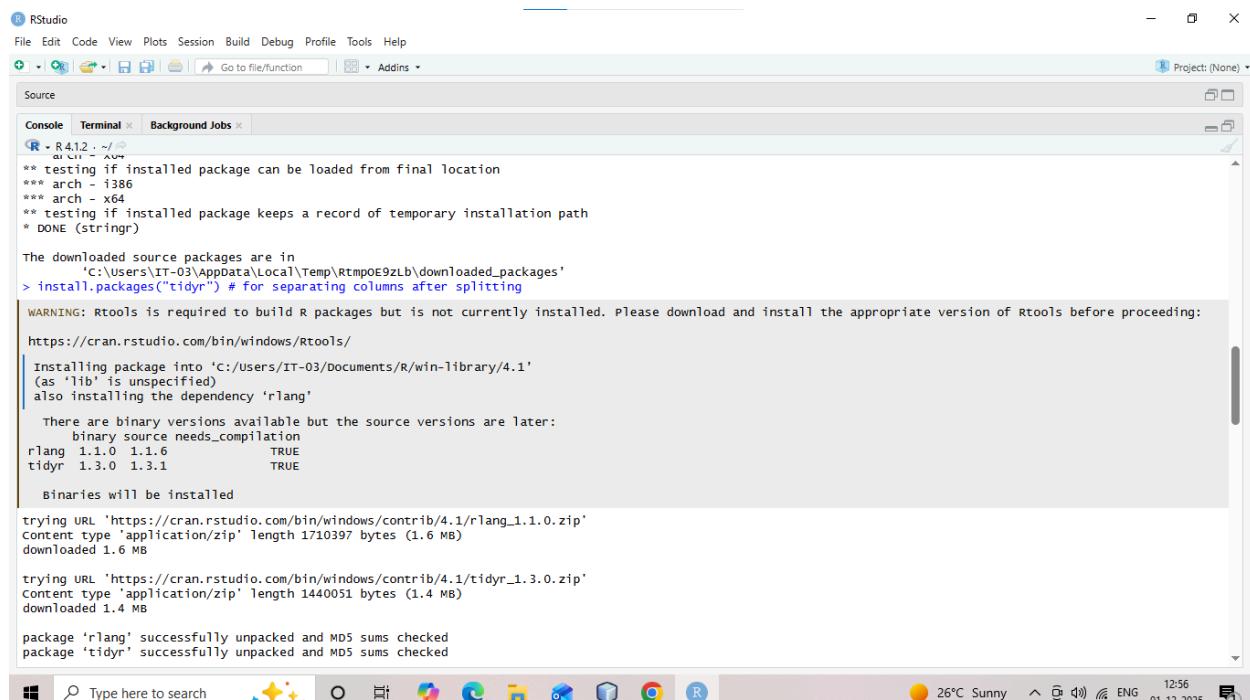
```
R> install.packages("stringr")
WARNING: Rtools is required to build R packages but is not currently installed. Please download and install the appropriate version of Rtools before proceeding:
https://cran.rstudio.com/bin/windows/Rtools/
Installing package into 'c:/users/IT-03/Documents/R/win-library/4.1'
(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
      nins                               html
```

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



RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Source
Console Terminal Background Jobs
R - R 4.1.2 - ~/
** 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 (string)

The downloaded source packages are in
'C:/Users/IT-03/AppData/Local/Temp/RtmpoE9zLb/downloaded_packages'
> install.packages("tidyverse") # 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'

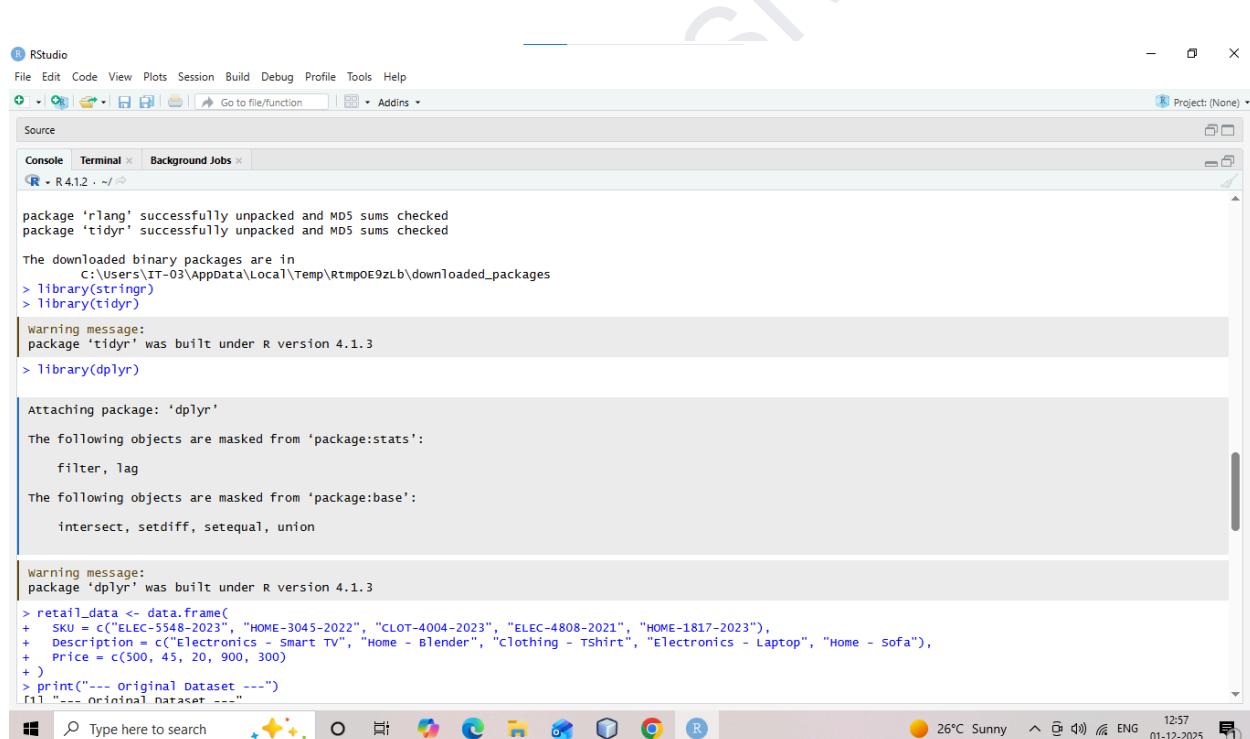
There are binary versions available but the source versions are later:
binary source needs_compilation
rlang 1.1.0 1.1.6 TRUE
tidyverse 1.3.0 1.3.1 TRUE

Binaries will be installed

trying URL 'https://cran.rstudio.com/bin/windows/contrib/4.1/rbase_4.1.0.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/tidyverse_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 'tidyverse' successfully unpacked and MD5 sums checked



RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Source
Console Terminal Background Jobs
R - R 4.1.2 - ~/

package 'rlang' successfully unpacked and MD5 sums checked
package 'tidyverse' 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(tidyverse)

Warning message:
package 'tidyverse' 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 ---"

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

The screenshot shows the RStudio interface with the following details:

- File Menu:** File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, Help.
- Project:** Project: (None)
- Source Tab:** Contains R code demonstrating various data manipulation techniques using base R functions like str_split and separate.
- Console Tab:** Shows the output of the R code, including data frames and summary statistics.
- Terminal Tab:** Not visible in the screenshot.
- Background Jobs Tab:** Not visible in the screenshot.
- Taskbar:** Includes icons for file operations (New, Open, Save, Print), RStudio Help, and Addins.
- System Tray:** Shows system status including battery level (26°C), weather (Sunny), signal strength, and date/time (01-12-2025).

The R code in the Source tab is as follows:

```
R> # Method A: Basic split (result is a list, harder to view directly in a dataframe)
R> split_list <- str_split(retail_data$Description, " - ")
R> print("--- Basic split output (List format) ---")
[1] "--- Basic split output (List format) ---"
R> print(split_list[[1]]) # Viewing the first item
[1] "Electronics" "Smart TV"
R> # Method B: split Fixed (Returns a matrix, easier to assign to columns)
R> # simplify = TRUE returns a matrix instead of a list
R> split_matrix <- str_split(retail_data$Description, " - ", simplify = TRUE)
R> # Assign the split parts to new columns
R> retail_data$Main_Cat <- split_matrix[, 1] # First column of matrix
R> retail_data$Sub_Cat <- split_matrix[, 2] # Second column of matrix
R> print("--- Data after str_split() (Manual Assignment) ---")
[1] "--- Data after str_split() (Manual Assignment) ---"
R> print(retail_data %>% select(Description, Main_cat, Sub_cat))
#> #> #> #> Description Main_Cat Sub_Cat
#> #> #> #> 1 Electronics - Smart TV Electronics Smart TV
#> #> #> #> 2 Home - Blender Home Blender
#> #> #> #> 3 Clothing - tshirt Clothing tshirt
#> #> #> #> 4 Electronics - Laptop Electronics Laptop
#> #> #> #> 5 Home - Sofa Home Sofa
#> tidy_data <- retail_data %>%
#>   separate(SKU, into = c("Dept", "ID", "Mfg_Year"), sep = "-")
R> print("--- Bonus: The 'separate' function (easier splitting) ---")
[1] "--- Bonus: The 'separate' function (easier splitting) ---"
R> print(tidy_data %>% select(Dept, ID, Mfg_Year))
#> #> #> Dept ID Mfg_Year
#> #> #> 1 ELEC 5548 2023
#> #> #> 2 HOME 3045 2022
#> #> #> 3 CLOT 4004 2023
#> #> #> 4 ELEC 4808 2021
```