Exp # 6 (Merge Data Frames)

1.Create the dataframes to merge:

buildings <- data.frame(location=c(1, 2, 3), name=c("building1", "building2", "building3"))

data <- data.frame(survey=c(1,1,1,2,2,2), location=c(1,2,3,2,3,1), efficiency=c(51,64,70,71,80,58))

The dataframes, buildings and data have a common key variable called, "location". Use the merge() function to merge the two dataframes by "location", into a new dataframe, "buildingStats".

Answer: buildings <- data.frame(location=c(1, 2, 3),name=c("building1", "building2","building3"))

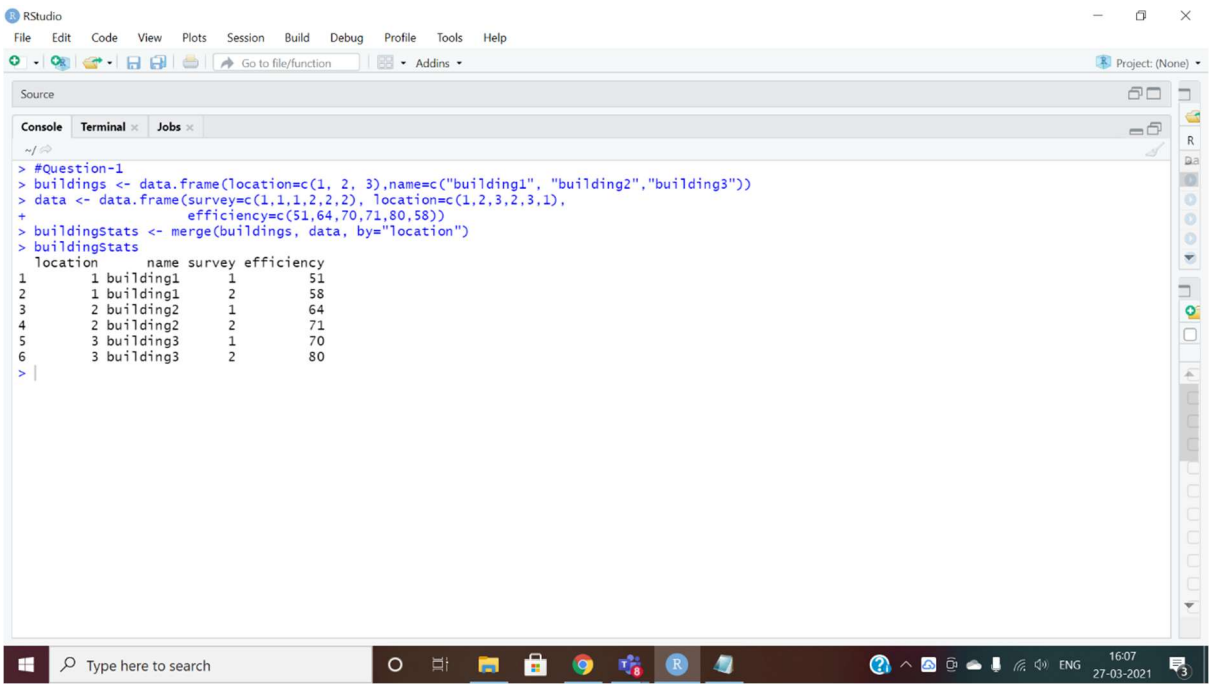
data <- data.frame(survey=c(1,1,1,2,2,2), location=c(1,2,3,2,3,1),

efficiency=c(51,64,70,71,80,58))

buildingStats <- merge(buildings, data, by="location")

buildingStats

OUTPUT



2 . Give the dataframes different key variable names:

buildings <- data.frame(location=c(1, 2, 3), name=c("building1", "building2", "building3"))

data <- data.frame(survey=c(1,1,1,2,2,2), LocationID=c(1,2,3,2,3,1), efficiency=c(51,64,70,71,80,58))

The dataframes, buildings and data now have corresponding variables called, location, and LocationID. Use the merge() function to merge the columns of the two dataframes by the corresponding variables.

Answer: buildings <- data.frame(location=c(1, 2, 3), name=c("building1", "building2", "building3"))

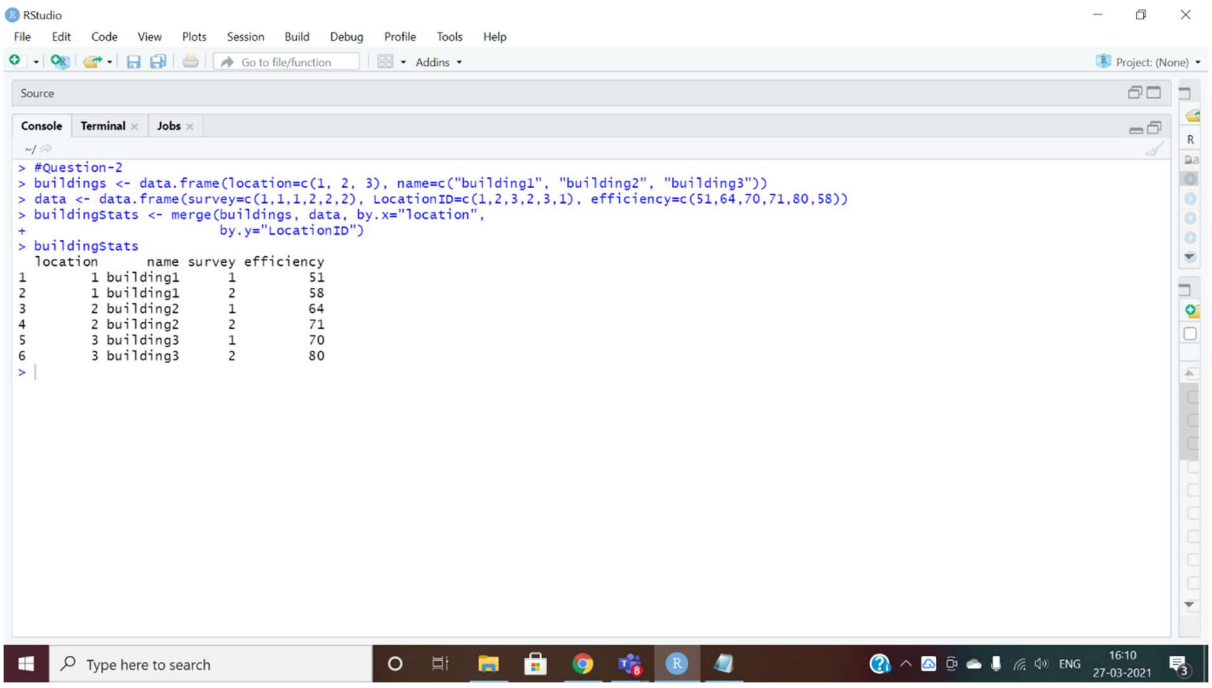
data <- data.frame(survey=c(1,1,1,2,2,2), LocationID=c(1,2,3,2,3,1), efficiency=c(51,64,70,71,80,58))

buildingStats <- merge(buildings, data, by.x="location",

by.y="LocationID")

buildingStats

OUTPUT



3 .

Inner Join:

The R merge() function automatically joins the frames by common variable names. In that case, demonstrate how you would perform the merge in Question 1 without specifying the key variable.

Answer: buildings <- data.frame(location=c(1, 2, 3), name=c("building1", "building2", "building3"))

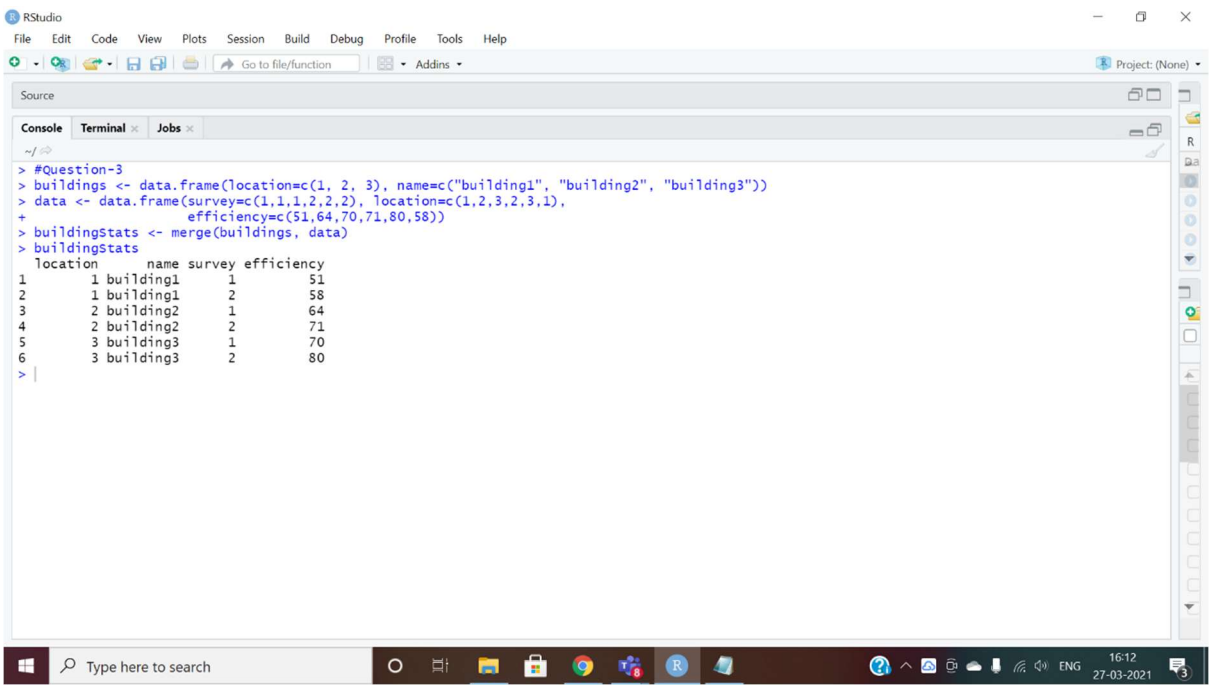
data <- data.frame(survey=c(1,1,1,2,2,2), location=c(1,2,3,2,3,1),

efficiency=c(51,64,70,71,80,58))

buildingStats <- merge(buildings, data)

buildingStats

OUTPUT



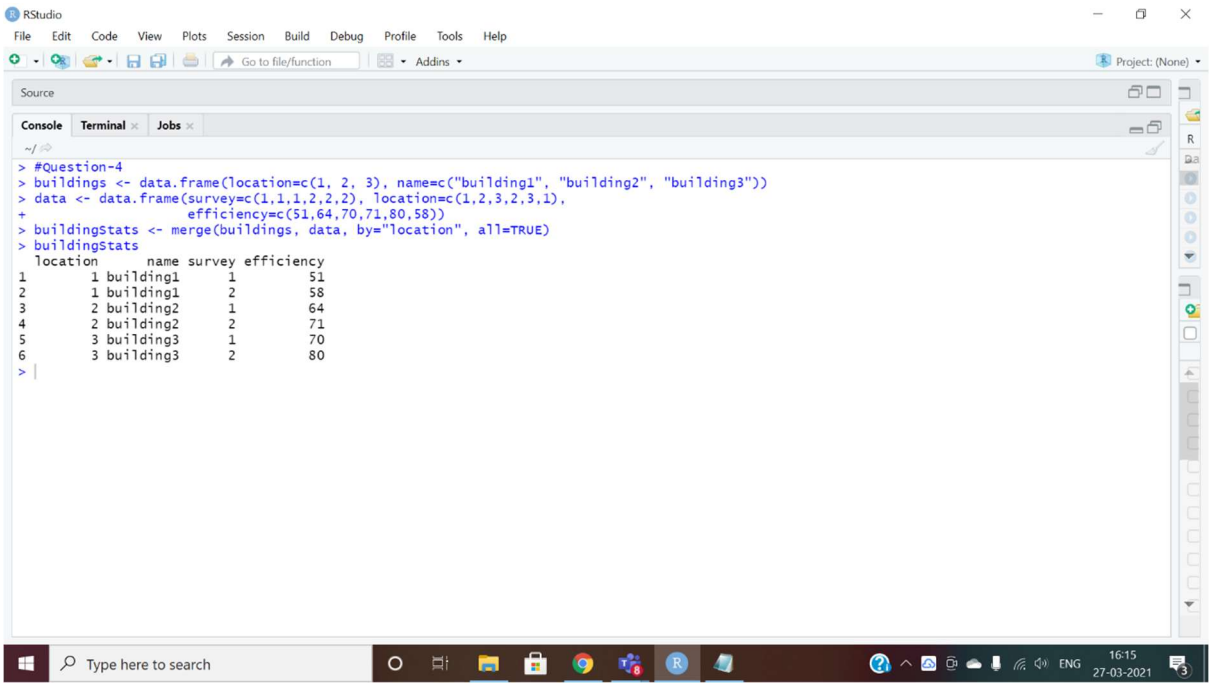
4 .

Outer Join:Merge the two dataframes from Question 1. Use the "all=" parameter in the merge() function to return all records from both tables. Also, merge with the key variable, "location".

Answer: buildings <- data.frame(location=c(1, 2, 3), name=c("building1", "building2", "building3"))

```
data <- data.frame(survey=c(1,1,1,2,2,2), location=c(1,2,3,2,3,1),
                   efficiency=c(51,64,70,71,80,58))
buildingStats <- merge(buildings, data, by="location", all=TRUE)
buildingStats
```

OUTPUT



5 .

Left Join:

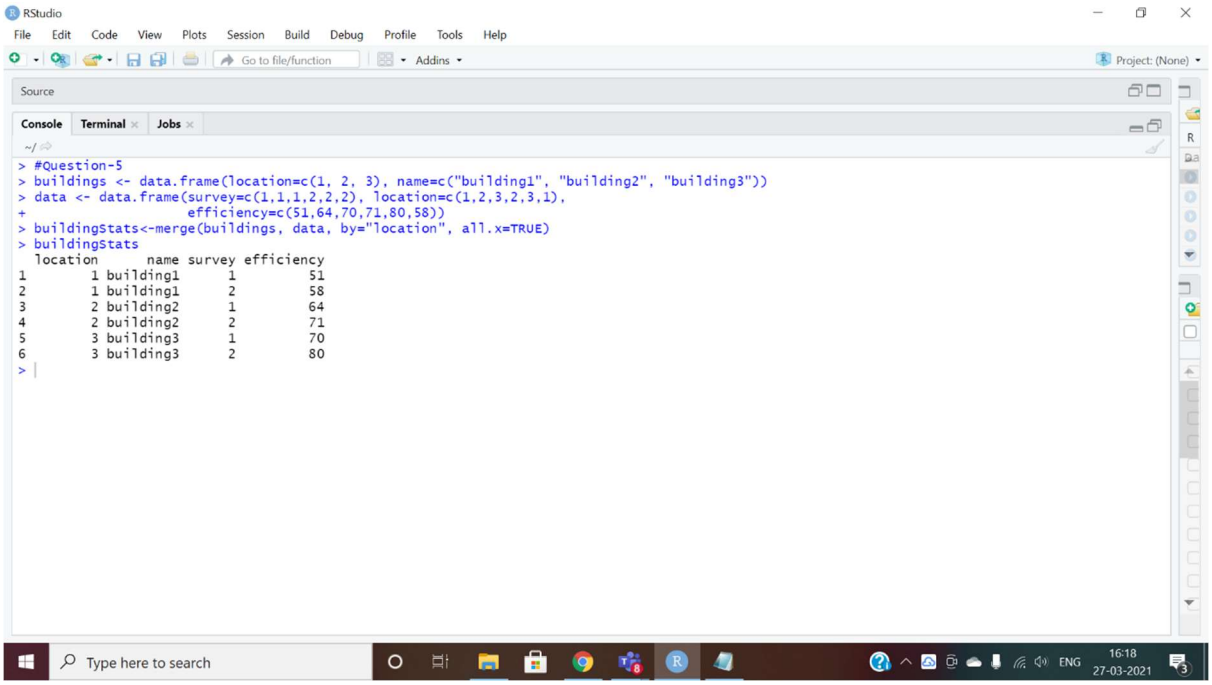
Merge the two dataframes from Question 1, and return all rows from the left table.

Specify the matching key from Question 1.

Answer: buildings <- data.frame(location=c(1, 2, 3), name=c("building1", "building2", "building3"))

```
data <- data.frame(survey=c(1,1,1,2,2,2), location=c(1,2,3,2,3,1),
                   efficiency=c(51,64,70,71,80,58))
buildingStats<-merge(buildings, data, by="location", all.x=TRUE)
buildingStats
```

OUTPUT



6 .

Right Join:

Merge the two dataframes from Question 1, and return all rows from the right table.

Use the matching key from Question 1 to return matching rows from the left table.

Answer: buildings <- data.frame(location=c(1, 2, 3), name=c("building1", "building2", "building3"))

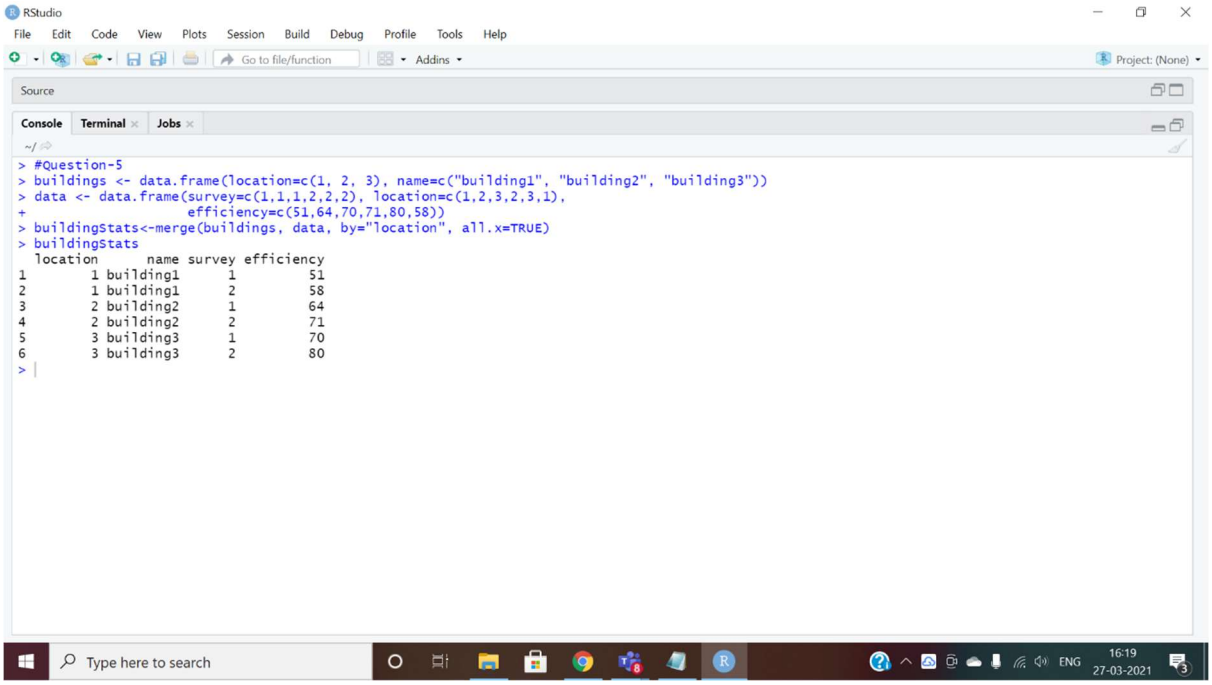
data <- data.frame(survey=c(1,1,1,2,2,2), location=c(1,2,3,2,3,1),

efficiency=c(51,64,70,71,80,58))

buildingStats <- merge(buildings, data, by="location", all.y=TRUE)

buildingStats

OUTPUT



7 .

Cross Join:

Merge the two dataframes from Question 1, into a "Cross Join" with each row of "buildings" matched to each row of "data". What new column names are created in "buildingStats"?

Answer: buildings <- data.frame(location=c(1, 2, 3), name=c("building1", "building2", "building3"))

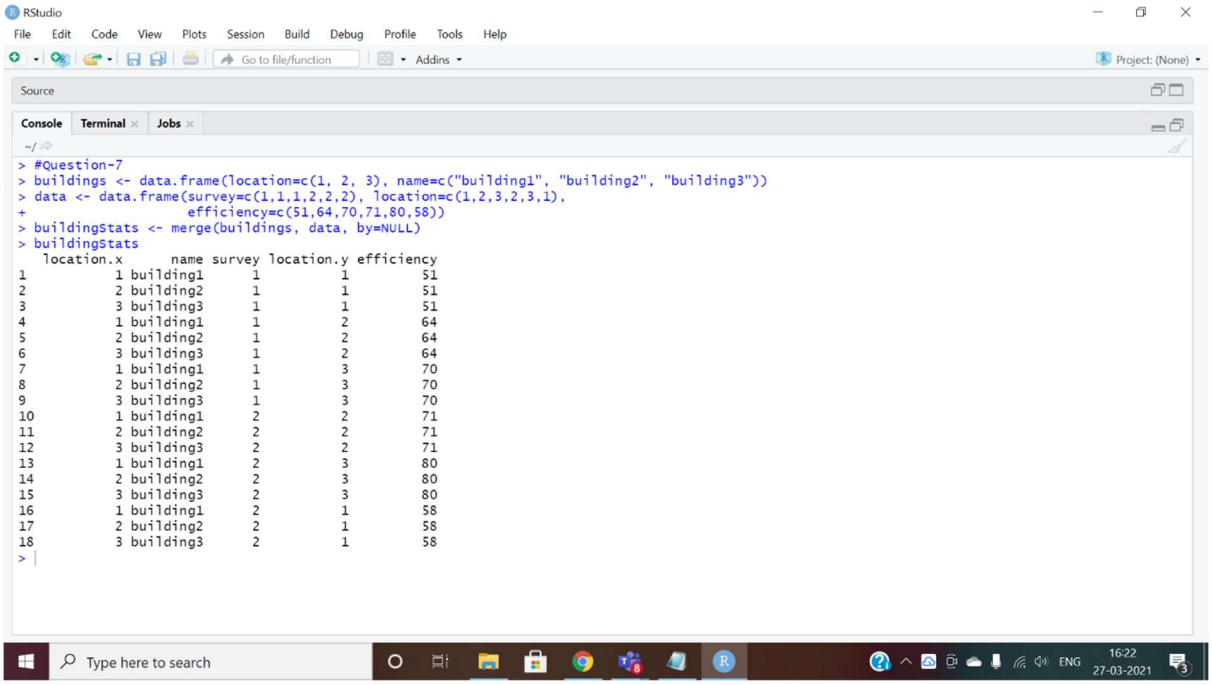
data <- data.frame(survey=c(1,1,1,2,2,2), location=c(1,2,3,2,3,1),

efficiency=c(51,64,70,71,80,58))

buildingStats <- merge(buildings, data, by=NULL)

buildingStats

OUTPUT



8.

Merging Dataframe rows:

To join two data frames (datasets) vertically, use the rbind function. The two data frames must have the same variables, but they do not have to be in the same order.

Merge the rows of the following two dataframes:

buildings <- data.frame(location=c(1, 2, 3), name=c("building1", "building2", "building3"))

buildings2 <- data.frame(location=c(5, 4, 6), name=c("building5", "building4", "building6"))

Also, specify a new dataframe, "allBuidings".

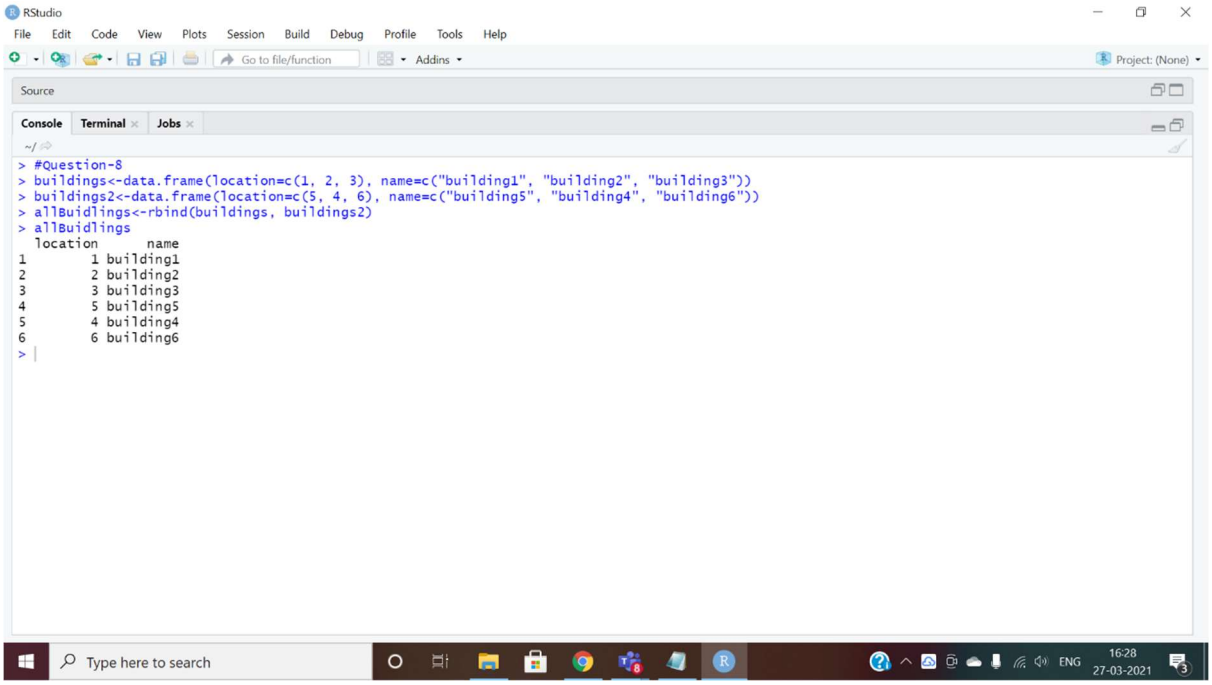
Answer: buildings <- data.frame(location=c(1, 2, 3), name=c("building1", "building2", "building3"))

buildings2 <- data.frame(location=c(5, 4, 6), name=c("building5", "building4", "building6"))

allBuidlings <- rbind(buildings, buildings2)

allBuidlings

OUTPUT



9.

Apply different join operations on the tables given below. Write the expected outputs.

Super Heroes

Name	Alignment	Gender	Publisher
Magneto	bad	male	Marvel
Storm	good	female	Marvel
Mystique	bad	female	Marvel
Batman	good	male	DC
Joker	bad	male	DC
Catwoman	bad	female	DC
Hellboy	good	male	Dark Horse Comics

Publishers

publisher	yr_founded
DC	1934
Marvel	1939
Image	1992

Answer:

1)Inner Join

inner join:suppressPackageStartupMessages(library(dplyr))

library(readr)

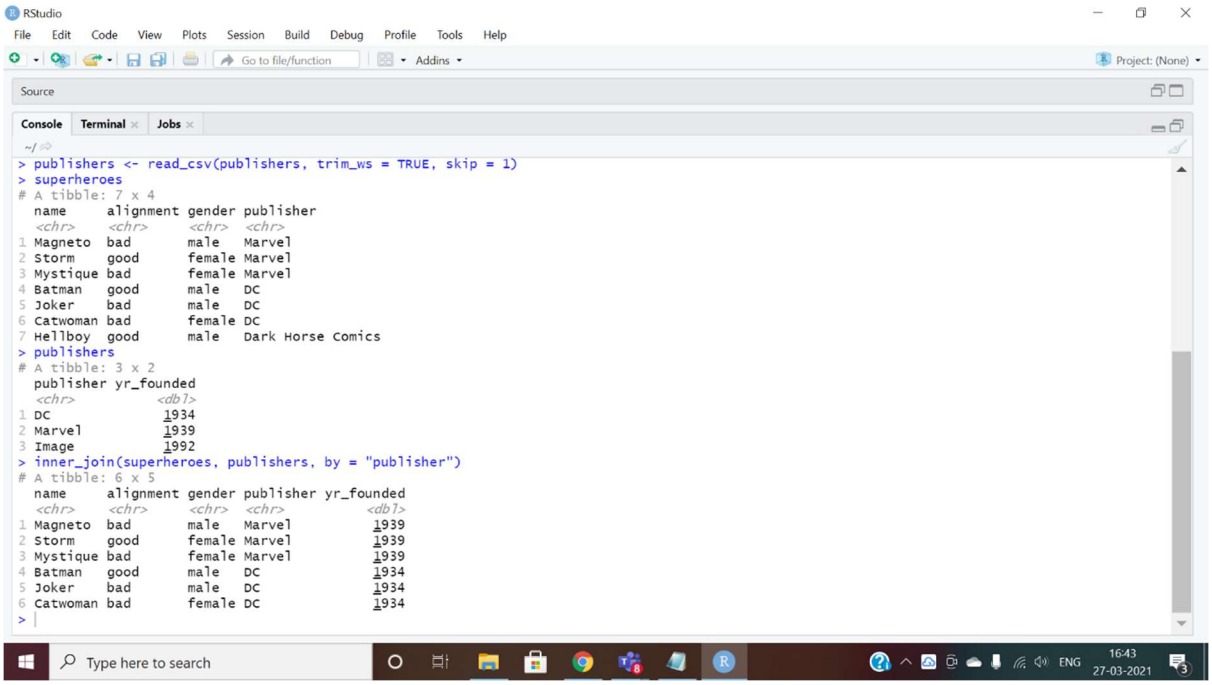
```
superheroes <- "
    name, alignment, gender,         publisher
 Magneto,       bad,   male,           Marvel
   Storm,      good, female,           Marvel
Mystique,       bad, female,           Marvel
  Batman,      good,   male,              DC
   Joker,       bad,   male,              DC
Catwoman,       bad, female,              DC
 Hellboy,      good,   male, Dark Horse Comics
"
superheroes <- read_csv(superheroes, trim_ws = TRUE, skip = 1)

publishers <- "
  publisher, yr_founded
         DC,       1934
     Marvel,       1939
      Image,       1992
"
publishers <- read_csv(publishers, trim_ws = TRUE, skip = 1)
```

superheroes

publishers

inner_join(superheroes, publishers, by = "publisher")

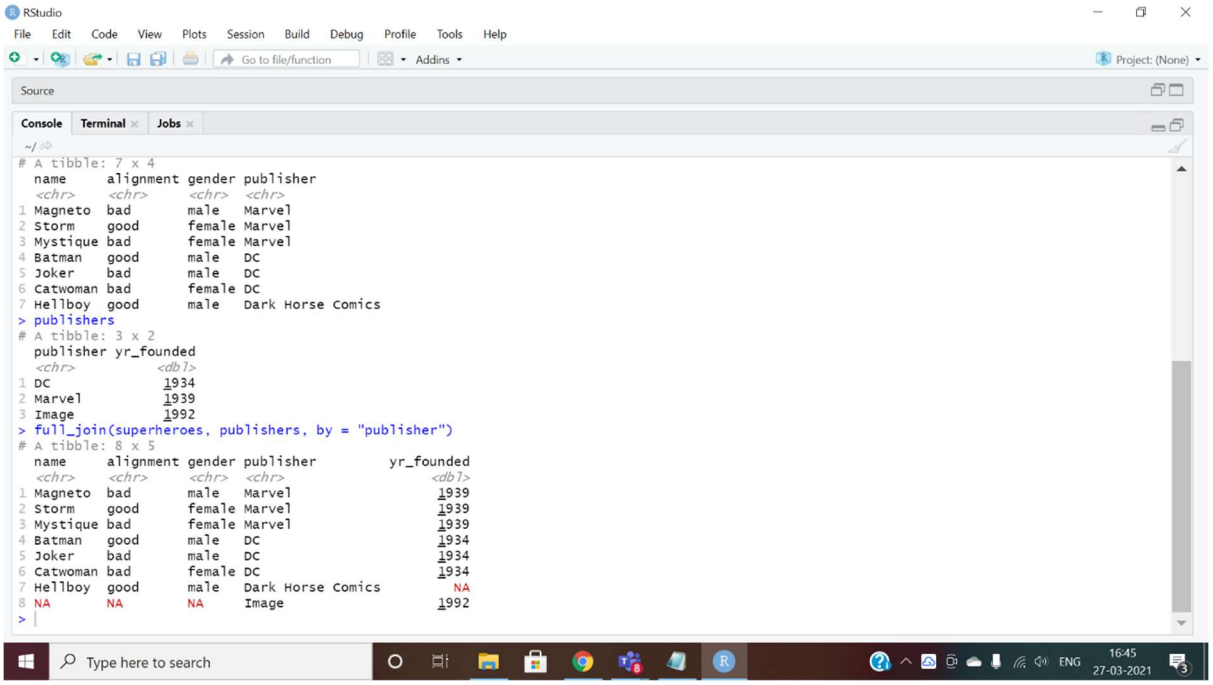
OUTPUT



2)Full Join

full_join(superheroes, publishers, by = "publisher")

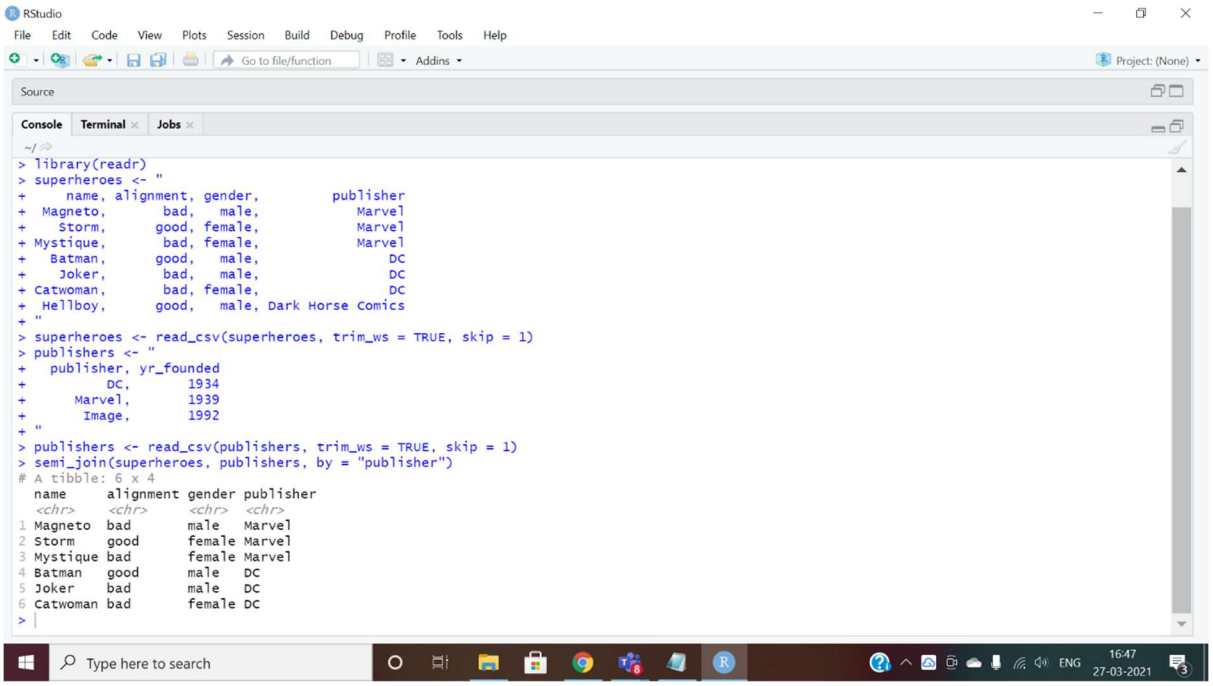
OUTPUT



3)**Semi join:**

semi_join(superheroes, publishers, by = "publisher")

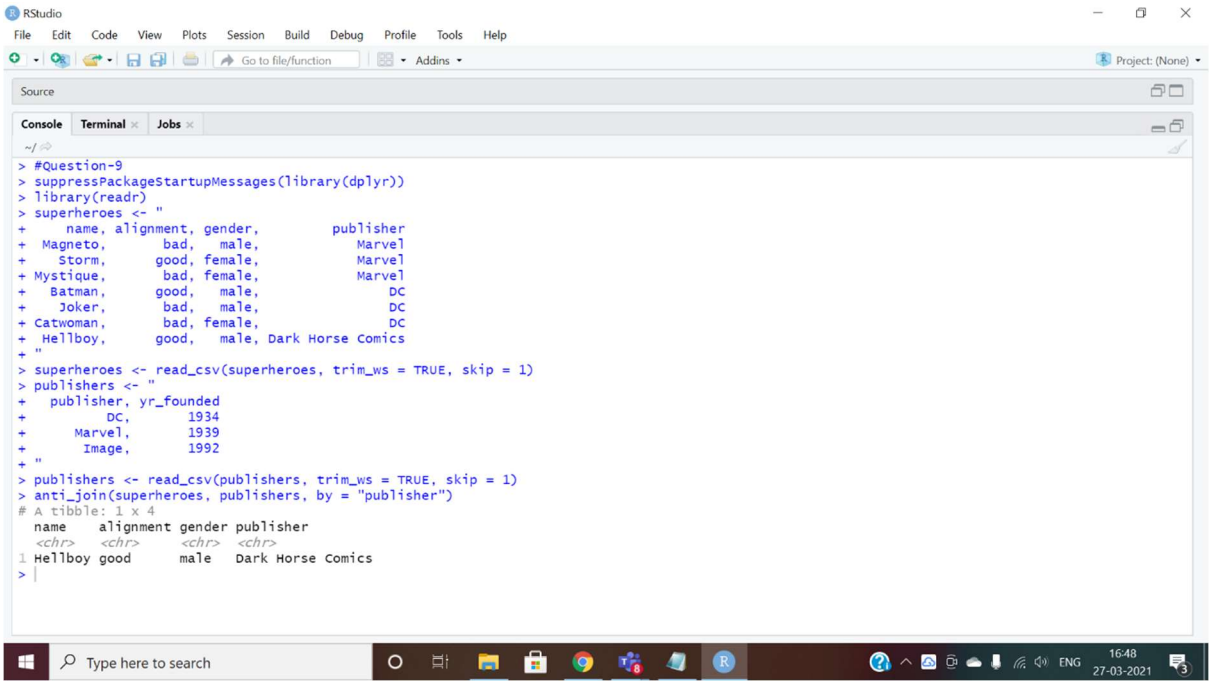
OUTPUT



4)Anti Join:

anti_join(superheroes, publishers, by = "publisher")

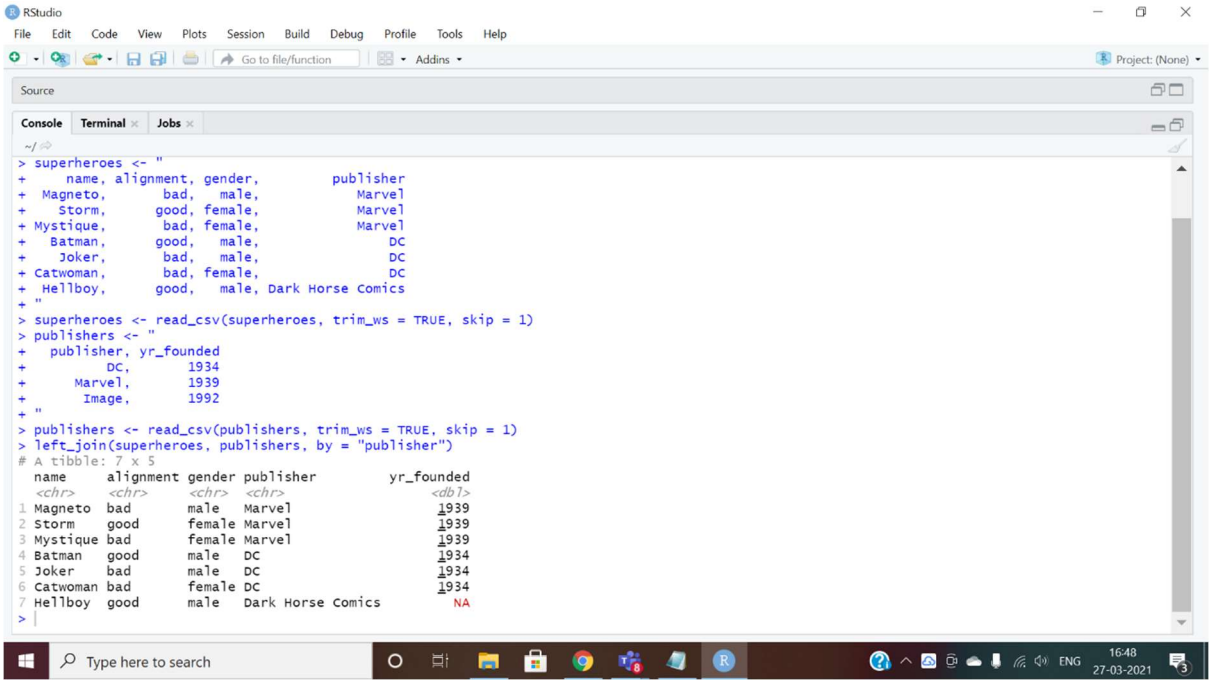
OUTPUT



5)Left Join:

left_join(superheroes, publishers, by = "publisher")

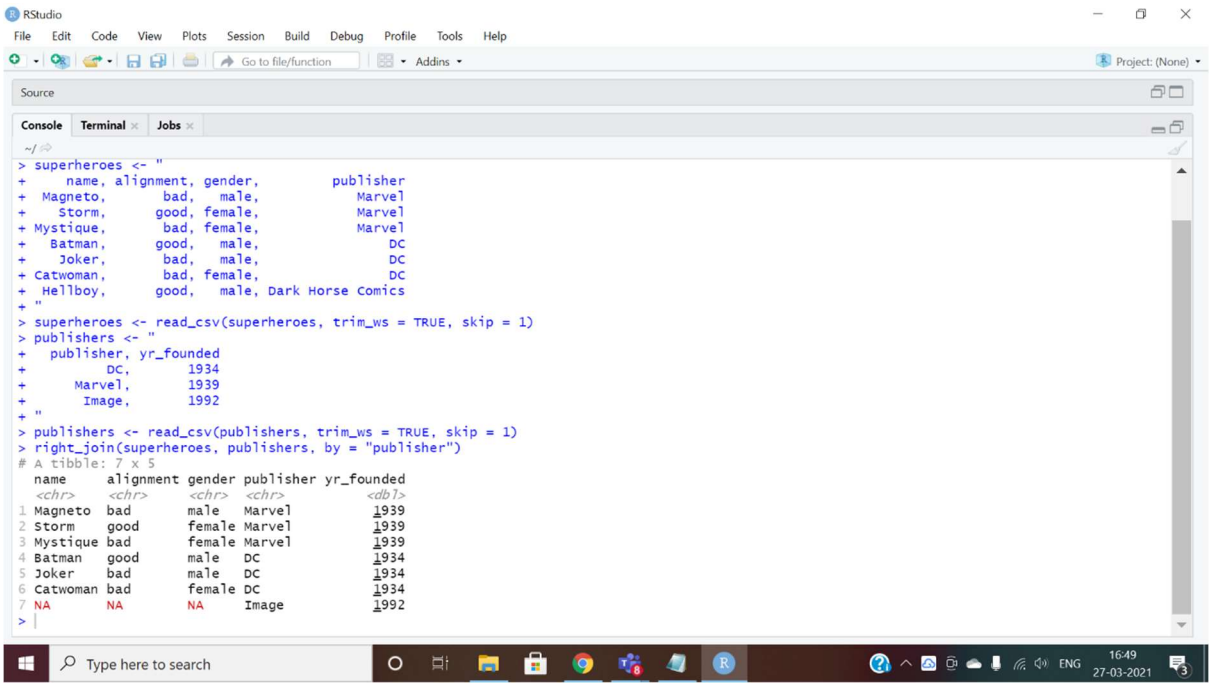
OUTPUT



6)Right Join:

right_join(superheroes, publishers, by = "publisher")

OUTPUT



10.

Use the merge() function to merge the mentioned dataframes.

authors <- data.frame(

surname = c("Tukey", "Venables", "Tierney", "Ripley", "McNeil"),

nationality = c("US", "Australia", "US", "UK", "Australia"),

retired = c("yes", rep("no", 4)))

books <- data.frame(

name = c("Tukey", "Venables", "Tierney", "Ripley", "Ripley", "McNeil"),

title = c("Exploratory Data Analysis",

"Modern Applied Statistics ...",

"LISP-STAT",

"Spatial Statistics", "Stochastic Simulation",

"Interactive Data Analysis"),

other.author = c(NA, "Ripley", NA, NA, NA, NA))

Answer: authors <- data.frame(

surname = c("Tukey", "Venables", "Tierney", "Ripley", "McNeil"),

nationality = c("US", "Australia", "US", "UK", "Australia"),

retired = c("yes", rep("no", 4)))

books <- data.frame(

name = c("Tukey", "Venables", "Tierney", "Ripley", "Ripley", "McNeil"),

title = c("Exploratory Data Analysis",

"Modern Applied Statistics ...",

"LISP-STAT",

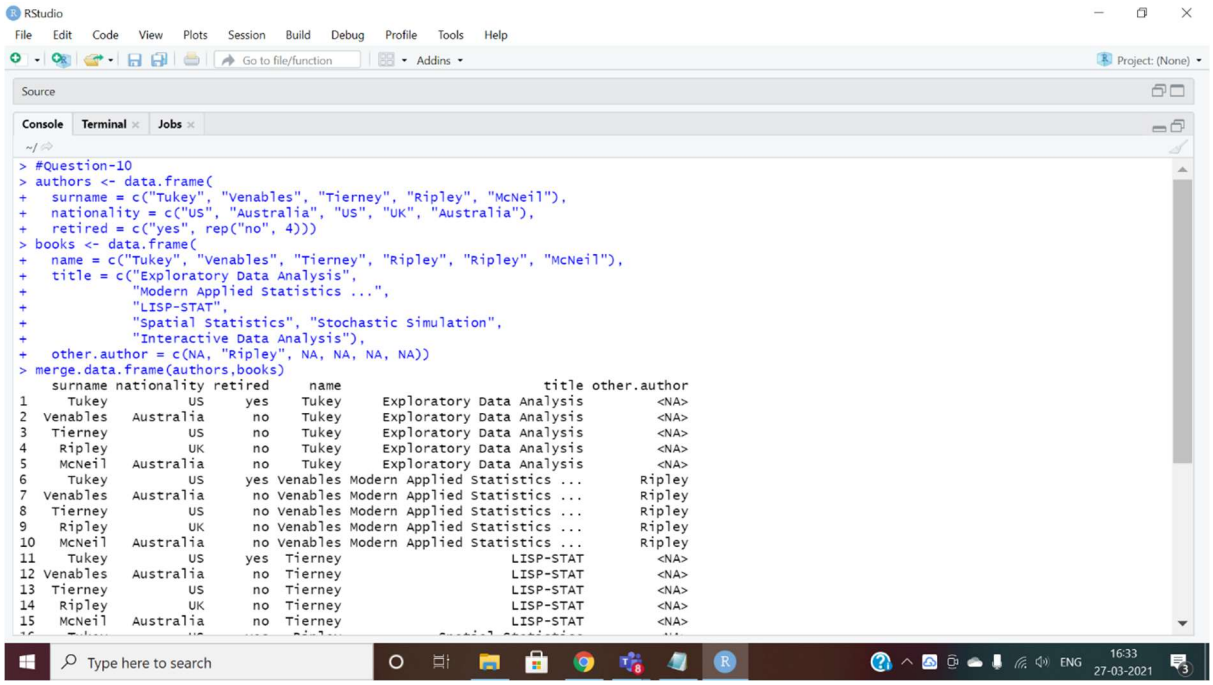
"Spatial Statistics", "Stochastic Simulation",

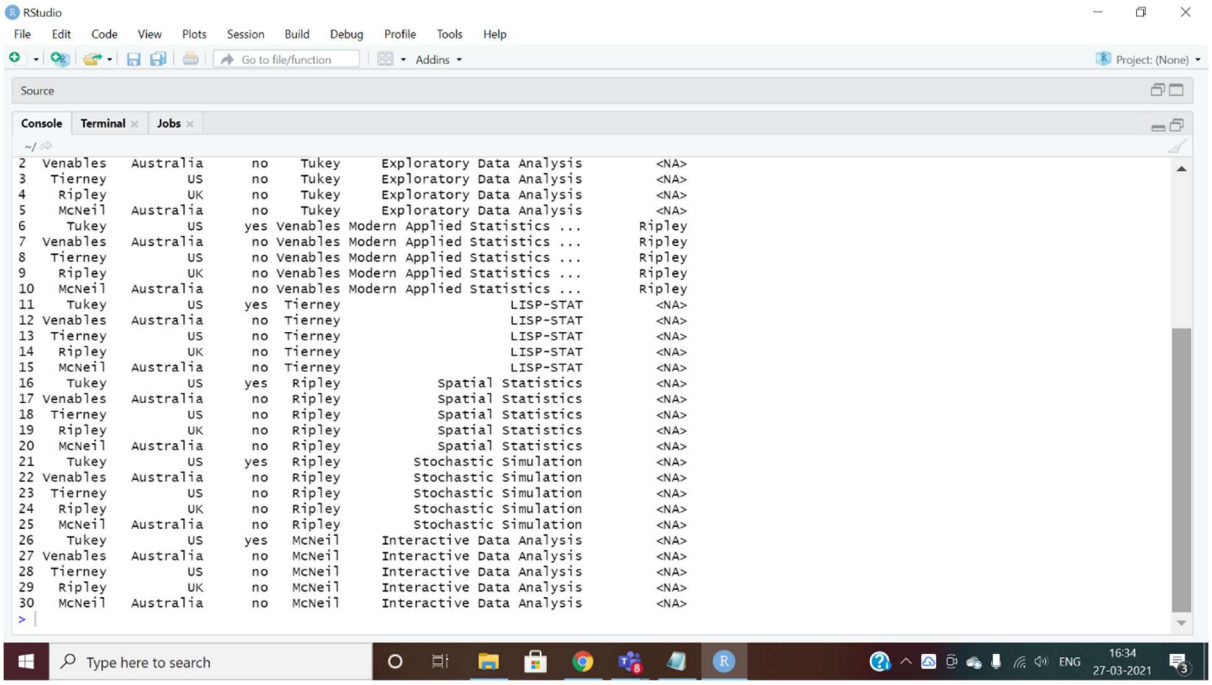
"Interactive Data Analysis"),

other.author = c(NA, "Ripley", NA, NA, NA, NA))

merge.data.frame(authors,books)

OUTPUT




RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
Project: (None)
Source
Console
Terminal
Jobs
~/
2 Venables Australia no Tukey Exploratory Data Analysis <NA>
3 Tierney US no Tukey Exploratory Data Analysis <NA>
4 Ripley UK no Tukey Exploratory Data Analysis <NA>
5 McNeil Australia no Tukey Exploratory Data Analysis <NA>
6 Tukey US yes Venables Modern Applied Statistics ... Ripley
7 Venables Australia no Venables Modern Applied Statistics ... Ripley
8 Tierney US no Venables Modern Applied Statistics ... Ripley
9 Ripley UK no Venables Modern Applied Statistics ... Ripley
10 McNeil Australia no Venables Modern Applied Statistics ... Ripley
11 Tukey US yes Tierney LISP-STAT <NA>
12 Venables Australia no Tierney LISP-STAT <NA>
13 Tierney US no Tierney LISP-STAT <NA>
14 Ripley UK no Tierney LISP-STAT <NA>
15 McNeil Australia no Tierney LISP-STAT <NA>
16 Tukey US yes Ripley Spatial Statistics <NA>
17 Venables Australia no Ripley Spatial Statistics <NA>
18 Tierney US no Ripley Spatial Statistics <NA>
19 Ripley UK no Ripley Spatial Statistics <NA>
20 McNeil Australia no Ripley Spatial Statistics <NA>
21 Tukey US yes Ripley Stochastic Simulation <NA>
22 Venables Australia no Ripley Stochastic Simulation <NA>
23 Tierney US no Ripley Stochastic Simulation <NA>
24 Ripley UK no Ripley Stochastic Simulation <NA>
25 McNeil Australia no Ripley Stochastic Simulation <NA>
26 Tukey US yes McNeil Interactive Data Analysis <NA>
27 Venables Australia no McNeil Interactive Data Analysis <NA>
28 Tierney US no McNeil Interactive Data Analysis <NA>
29 Ripley UK no McNeil Interactive Data Analysis <NA>
30 McNeil Australia no McNeil Interactive Data Analysis <NA>
>
Type here to search
ENG
16:34
27-03-2021