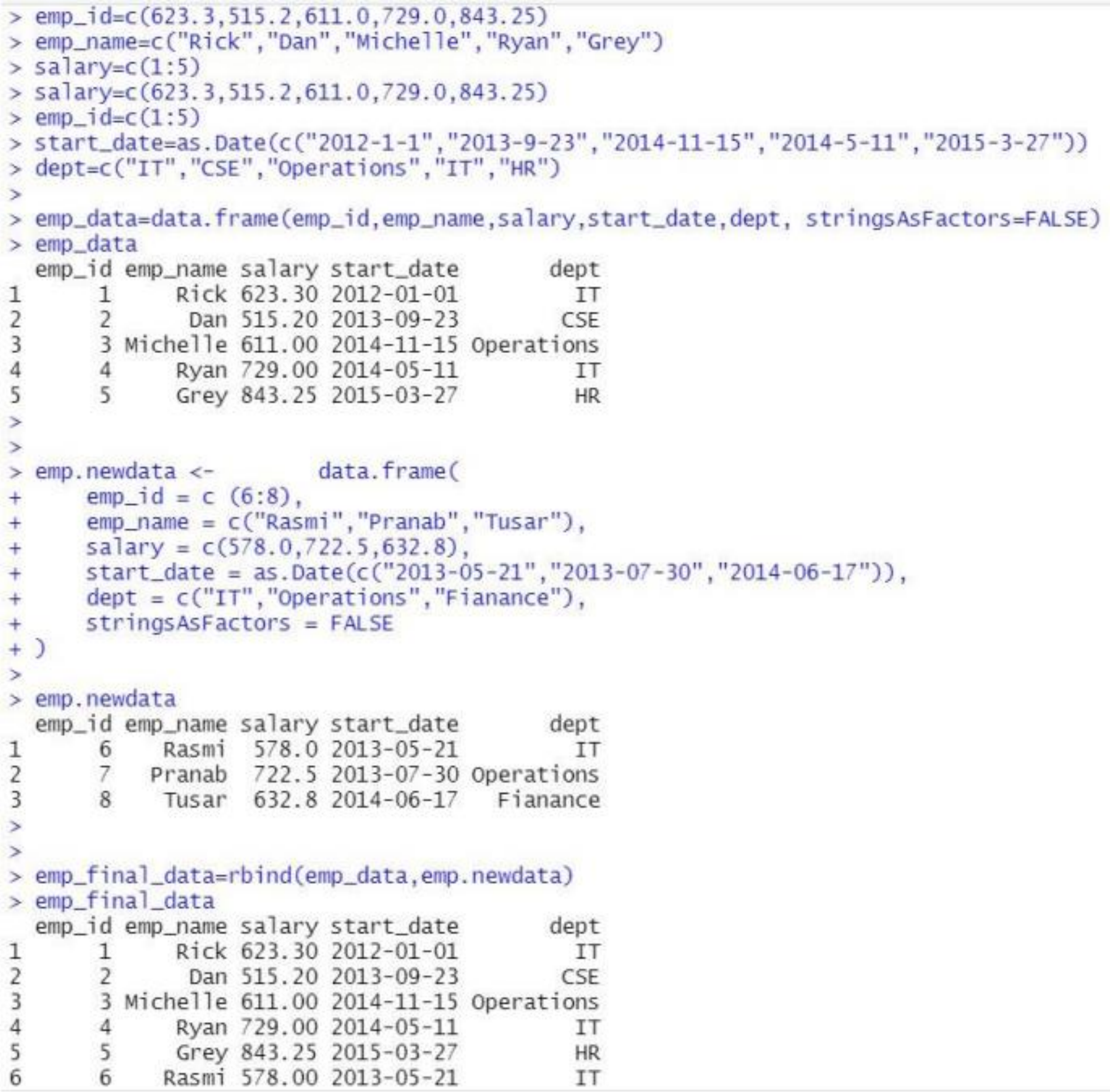
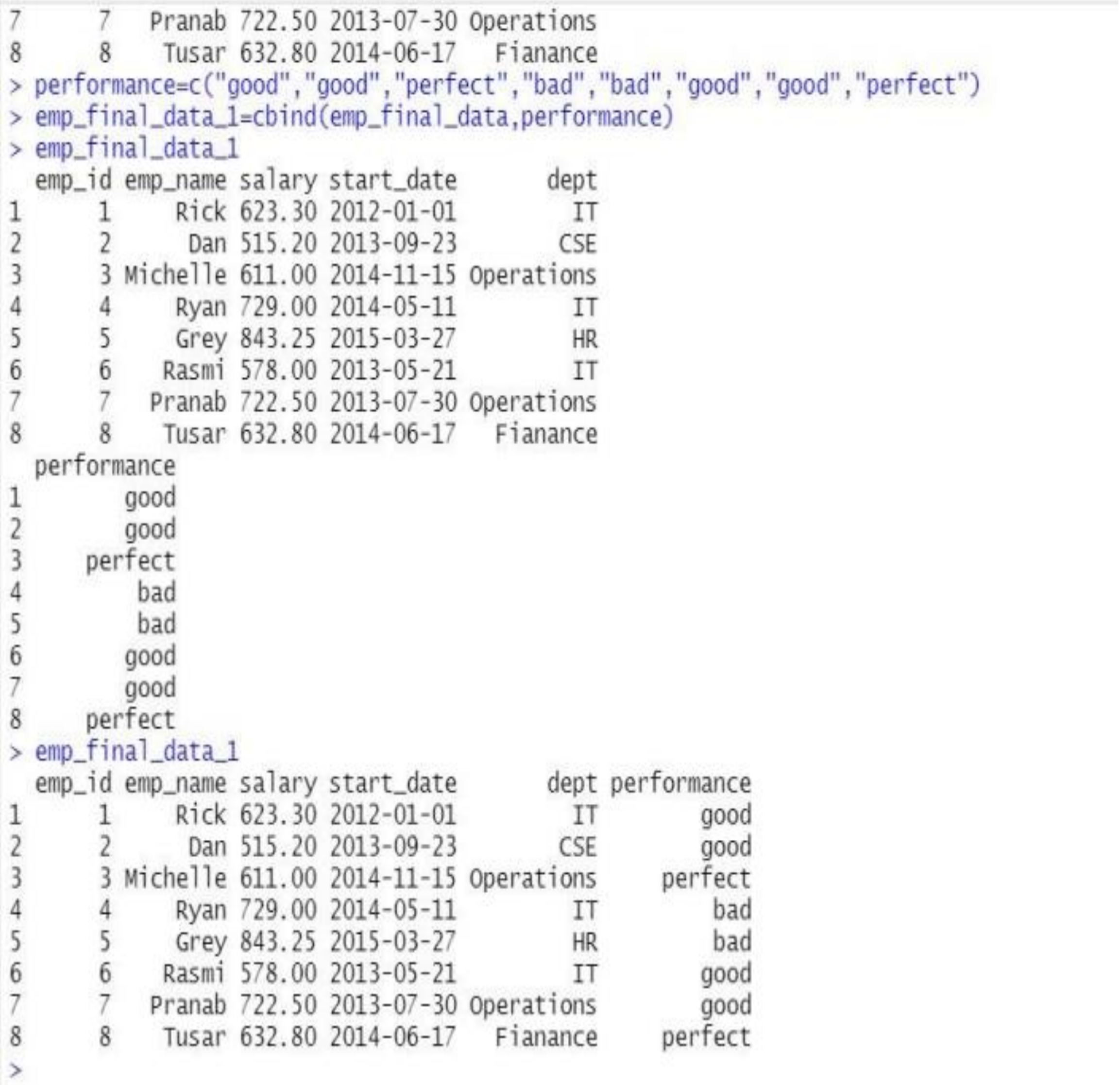
FDA

LAB-4

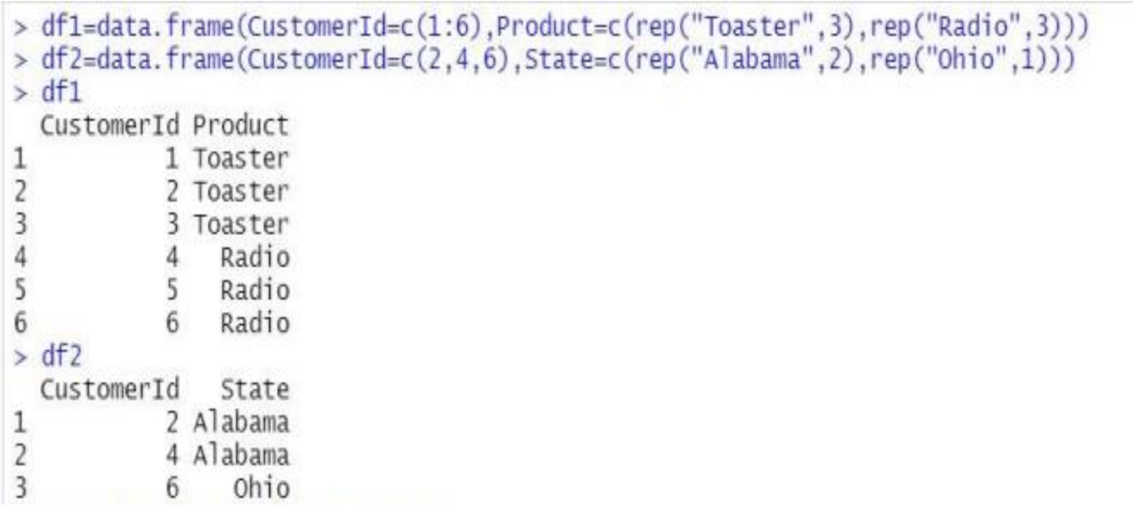
Practice the exercises explained in the class room (Follow the lecture slides) first and solve the exercises given below later:



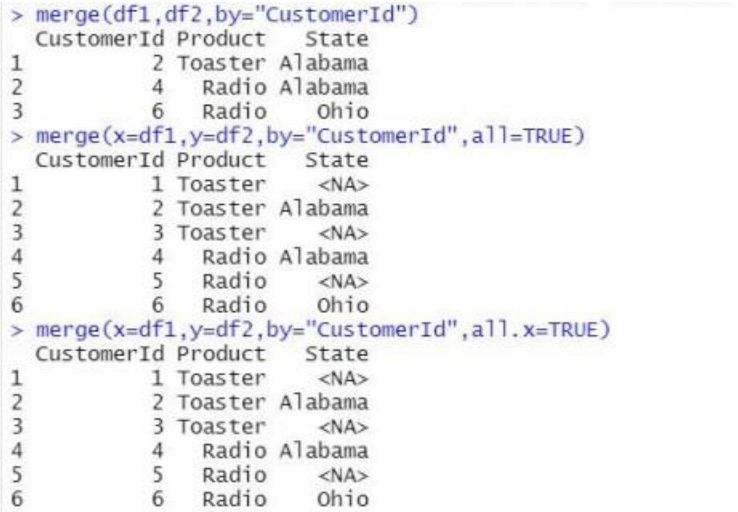
```
> emp_id=c(623.3,515.2,611.0,729.0,843.25)
> emp_name=c("Rick","Dan","Michelle","Ryan","Grey")
> salary=c(1:5)
> salary=c(623.3,515.2,611.0,729.0,843.25)
> emp_id=c(1:5)
> start_date=as.Date(c("2012-1-1","2013-9-23","2014-11-15","2014-5-11","2015-3-27"))
> dept=c("IT","CSE","Operations","IT","HR")
>
> emp_data=data.frame(emp_id,emp_name,salary,start_date,dept, stringsAsFactors=FALSE)
> emp_data
  emp_id emp_name salary start_date       dept
1      1     Rick 623.30 2012-01-01         IT
2      2      Dan 515.20 2013-09-23        CSE
3      3 Michelle 611.00 2014-11-15 Operations
4      4     Ryan 729.00 2014-05-11         IT
5      5     Grey 843.25 2015-03-27         HR
>
>
> emp.newdata <-        data.frame(
+     emp_id = c (6:8),
+     emp_name = c("Rasmi","Pranab","Tusar"),
+     salary = c(578.0,722.5,632.8),
+     start_date = as.Date(c("2013-05-21","2013-07-30","2014-06-17")),
+     dept = c("IT","Operations","Fianance"),
+     stringsAsFactors = FALSE
+ )
>
> emp.newdata
  emp_id emp_name salary start_date       dept
1      6    Rasmi  578.0 2013-05-21         IT
2      7   Pranab  722.5 2013-07-30 Operations
3      8    Tusar  632.8 2014-06-17   Fianance
>
>
> emp_final_data=rbind(emp_data,emp.newdata)
> emp_final_data
  emp_id emp_name salary start_date       dept
1      1     Rick 623.30 2012-01-01         IT
2      2      Dan 515.20 2013-09-23        CSE
3      3 Michelle 611.00 2014-11-15 Operations
4      4     Ryan 729.00 2014-05-11         IT
5      5     Grey 843.25 2015-03-27         HR
6      6    Rasmi 578.00 2013-05-21         IT
```



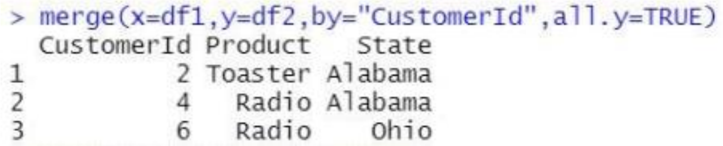
```
7      7   Pranab 722.50 2013-07-30 Operations
8      8    Tusar 632.80 2014-06-17   Fianance
> performance=c("good","good","perfect","bad","bad","good","good","perfect")
> emp_final_data_1=cbind(emp_final_data,performance)
> emp_final_data_1
  emp_id emp_name salary start_date       dept
1      1     Rick 623.30 2012-01-01         IT
2      2      Dan 515.20 2013-09-23        CSE
3      3 Michelle 611.00 2014-11-15 Operations
4      4     Ryan 729.00 2014-05-11         IT
5      5     Grey 843.25 2015-03-27         HR
6      6    Rasmi 578.00 2013-05-21         IT
7      7   Pranab 722.50 2013-07-30 Operations
8      8    Tusar 632.80 2014-06-17   Fianance
  performance
1        good
2        good
3     perfect
4         bad
5         bad
6        good
7        good
8     perfect
> emp_final_data_1
  emp_id emp_name salary start_date       dept performance
1      1     Rick 623.30 2012-01-01         IT        good
2      2      Dan 515.20 2013-09-23        CSE        good
3      3 Michelle 611.00 2014-11-15 Operations     perfect
4      4     Ryan 729.00 2014-05-11         IT         bad
5      5     Grey 843.25 2015-03-27         HR         bad
6      6    Rasmi 578.00 2013-05-21         IT        good
7      7   Pranab 722.50 2013-07-30 Operations        good
8      8    Tusar 632.80 2014-06-17   Fianance     perfect
>
```



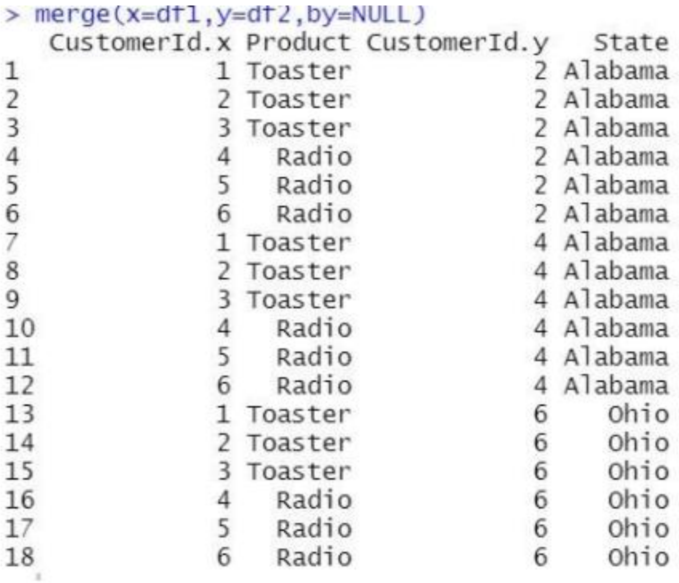
```
> df1=data.frame(CustomerId=c(1:6),Product=c(rep("Toaster",3),rep("Radio",3)))
> df2=data.frame(CustomerId=c(2,4,6),State=c(rep("Alabama",2),rep("Ohio",1)))
> df1
  CustomerId Product
1          1 Toaster
2          2 Toaster
3          3 Toaster
4          4   Radio
5          5   Radio
6          6   Radio
> df2
  CustomerId   State
1          2 Alabama
2          4 Alabama
3          6    Ohio
```



```
> merge(df1,df2,by="CustomerId")
  CustomerId Product   State
1          2 Toaster Alabama
2          4   Radio Alabama
3          6   Radio    Ohio
> merge(x=df1,y=df2,by="CustomerId",all=TRUE)
  CustomerId Product   State
1          1 Toaster    <NA>
2          2 Toaster Alabama
3          3 Toaster    <NA>
4          4   Radio Alabama
5          5   Radio    <NA>
6          6   Radio    Ohio
> merge(x=df1,y=df2,by="CustomerId",all.x=TRUE)
  CustomerId Product   State
1          1 Toaster    <NA>
2          2 Toaster Alabama
3          3 Toaster    <NA>
4          4   Radio Alabama
5          5   Radio    <NA>
6          6   Radio    Ohio
```



```
> merge(x=df1,y=df2,by="CustomerId",all.y=TRUE)
  CustomerId Product   State
1          2 Toaster Alabama
2          4   Radio Alabama
3          6   Radio    Ohio
```



```
> merge(x=df1,y=df2,by=NULL)
   CustomerId.x Product CustomerId.y   State
1             1 Toaster            2 Alabama
2             2 Toaster            2 Alabama
3             3 Toaster            2 Alabama
4             4   Radio            2 Alabama
5             5   Radio            2 Alabama
6             6   Radio            2 Alabama
7             1 Toaster            4 Alabama
8             2 Toaster            4 Alabama
9             3 Toaster            4 Alabama
10            4   Radio            4 Alabama
11            5   Radio            4 Alabama
12            6   Radio            4 Alabama
13            1 Toaster            6    Ohio
14            2 Toaster            6    Ohio
15            3 Toaster            6    Ohio
16            4   Radio            6    Ohio
17            5   Radio            6    Ohio
18            6   Radio            6    Ohio
```

1)

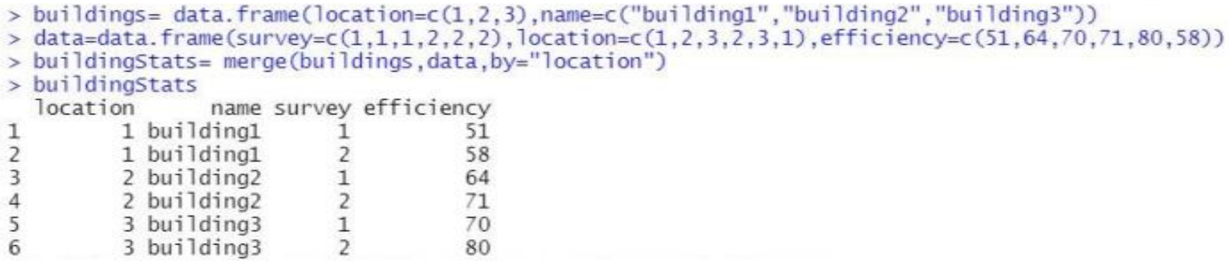
Create the dataframes to merge:

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”.



```
> 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
  location      name survey efficiency
1        1 building1      1         51
2        1 building1      2         58
3        2 building2      1         64
4        2 building2      2         71
5        3 building3      1         70
6        3 building3      2         80
```

2)

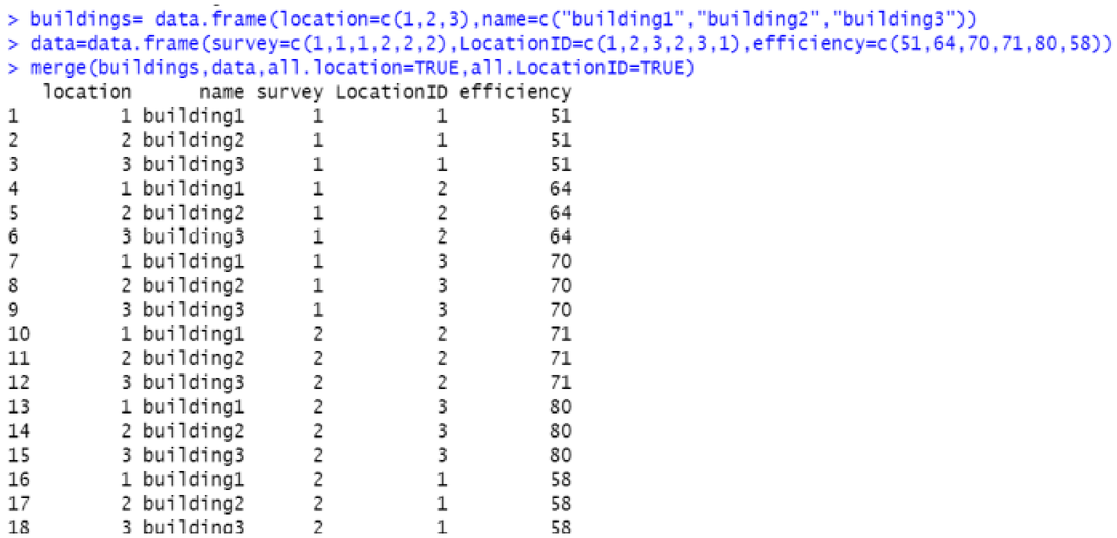
Give the dataframes different key variable names:

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.

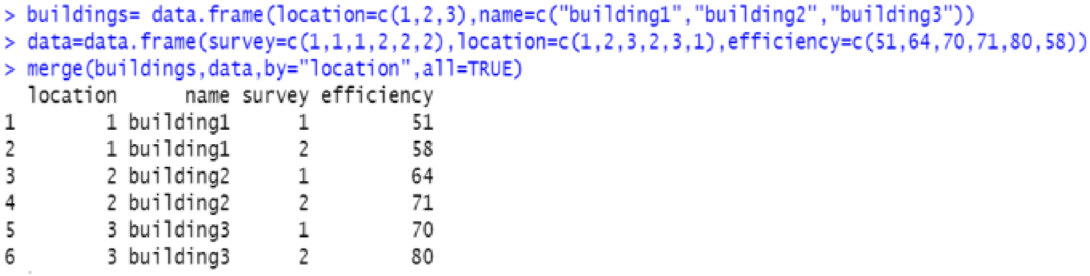


```
> buildings= data.frame(location=c(1,2,3),name=c("building1","building2","building3"))
> data=data.frame(survey=c(1,1,1,2,2,2),LocationID=c(1,2,3,2,3,1),efficiency=c(51,64,70,71,80,58))
> merge(buildings,data,all.location=TRUE,all.LocationID=TRUE)
   location      name survey LocationID efficiency
1         1 building1      1          1         51
2         2 building2      1          1         51
3         3 building3      1          1         51
4         1 building1      1          2         64
5         2 building2      1          2         64
6         3 building3      1          2         64
7         1 building1      1          3         70
8         2 building2      1          3         70
9         3 building3      1          3         70
10        1 building1      2          2         71
11        2 building2      2          2         71
12        3 building3      2          2         71
13        1 building1      2          3         80
14        2 building2      2          3         80
15        3 building3      2          3         80
16        1 building1      2          1         58
17        2 building2      2          1         58
18        3 building3      2          1         58
```

4)

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 Exercise 1 without specifying the key variable.



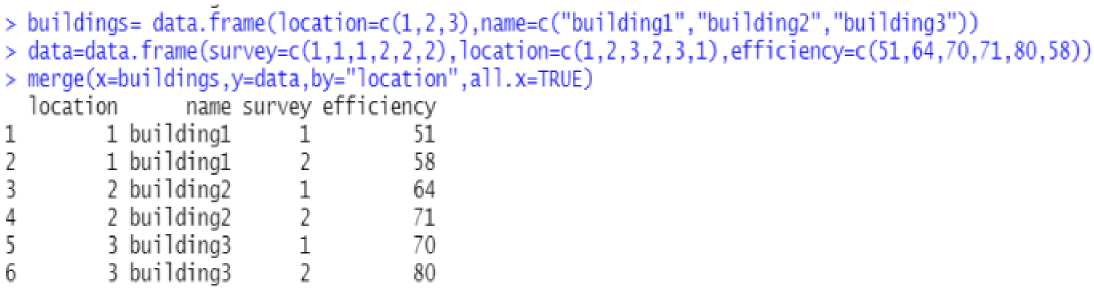
```
> buildings= data.frame(location=c(1,2,3),name=c("building1","building2","building3"))
> data=data.frame(survey=c(1,1,1,2,2,2),location=c(1,2,3,2,3,1),efficiency=c(51,64,70,71,80,58))
> merge(buildings,data,by="location",all=TRUE)
  location      name survey efficiency
1        1 building1      1         51
2        1 building1      2         58
3        2 building2      1         64
4        2 building2      2         71
5        3 building3      1         70
6        3 building3      2         80
```

5)

Left Join:

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

Specify the matching key from Exercise 1.



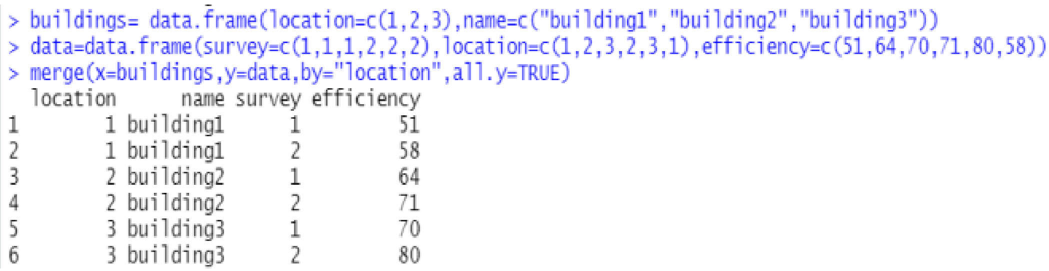
```
> buildings= data.frame(location=c(1,2,3),name=c("building1","building2","building3"))
> data=data.frame(survey=c(1,1,1,2,2,2),location=c(1,2,3,2,3,1),efficiency=c(51,64,70,71,80,58))
> merge(x=buildings,y=data,by="location",all.x=TRUE)
  location      name survey efficiency
1        1 building1      1         51
2        1 building1      2         58
3        2 building2      1         64
4        2 building2      2         71
5        3 building3      1         70
6        3 building3      2         80
```

6)

Right Join:

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

Use the matching key from Exercise 1 to return matching rows from the left table

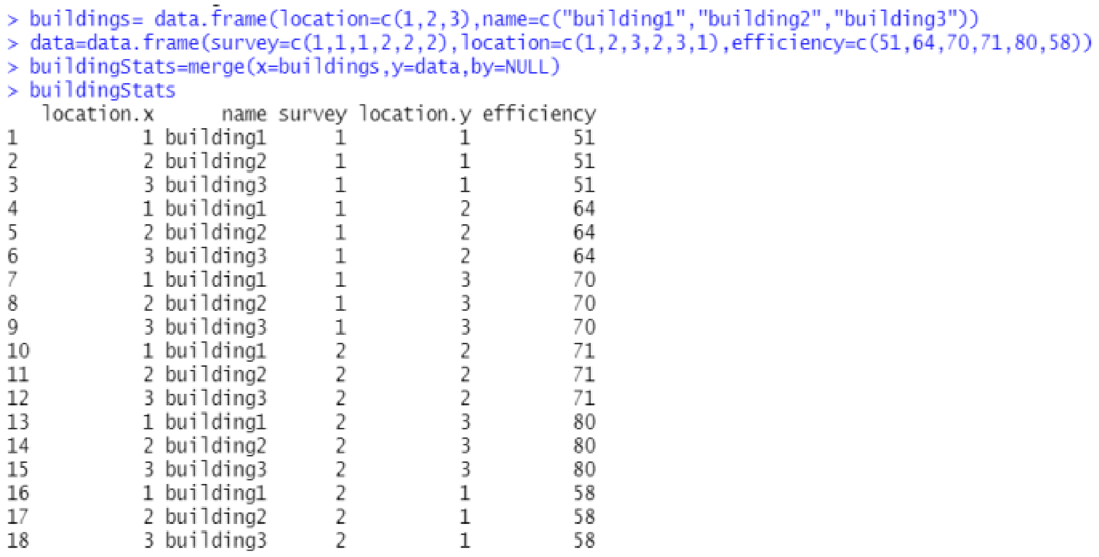


```
> buildings= data.frame(location=c(1,2,3),name=c("building1","building2","building3"))
> data=data.frame(survey=c(1,1,1,2,2,2),location=c(1,2,3,2,3,1),efficiency=c(51,64,70,71,80,58))
> merge(x=buildings,y=data,by="location",all.y=TRUE)
  location      name survey efficiency
1        1 building1      1         51
2        1 building1      2         58
3        2 building2      1         64
4        2 building2      2         71
5        3 building3      1         70
6        3 building3      2         80
```

7)

Cross Join:

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

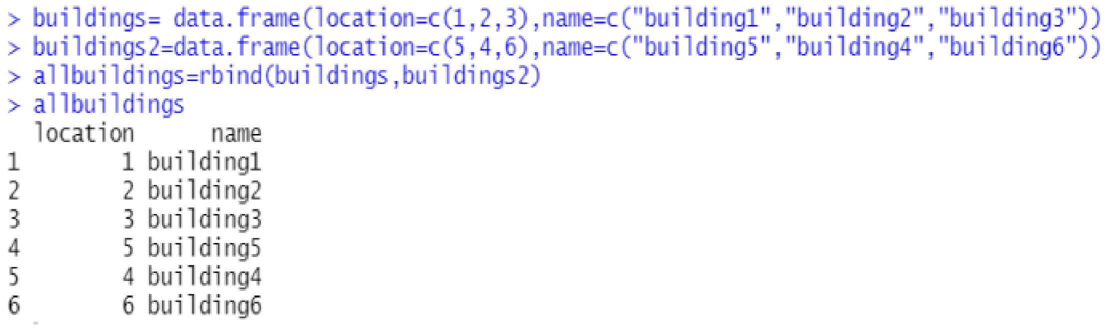


```
> buildings= data.frame(location=c(1,2,3),name=c("building1","building2","building3"))
> data=data.frame(survey=c(1,1,1,2,2,2),location=c(1,2,3,2,3,1),efficiency=c(51,64,70,71,80,58))
> buildingStats=merge(x=buildings,y=data,by=NULL)
> buildingStats
   location.x      name survey location.y efficiency
1           1 building1      1          1         51
2           2 building2      1          1         51
3           3 building3      1          1         51
4           1 building1      1          2         64
5           2 building2      1          2         64
6           3 building3      1          2         64
7           1 building1      1          3         70
8           2 building2      1          3         70
9           3 building3      1          3         70
10          1 building1      2          2         71
11          2 building2      2          2         71
12          3 building3      2          2         71
13          1 building1      2          3         80
14          2 building2      2          3         80
15          3 building3      2          3         80
16          1 building1      2          1         58
17          2 building2      2          1         58
18          3 building3      2          1         58
```

8)

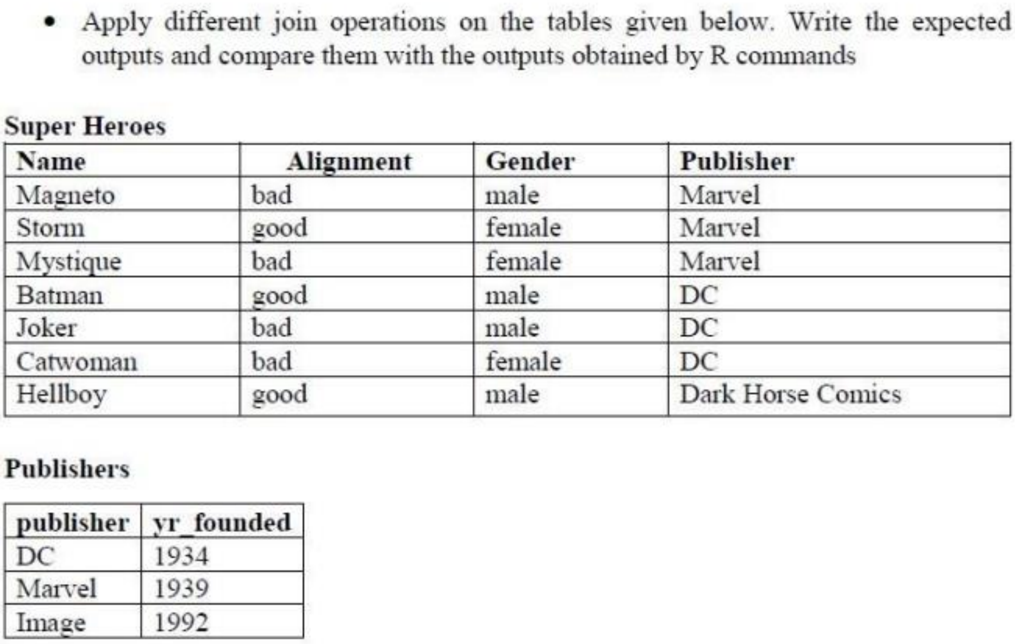
Merging Dataframe rows:

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”.



```
> 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"))
> allbuildings=rbind(buildings,buildings2)
> allbuildings
  location      name
1        1 building1
2        2 building2
3        3 building3
4        5 building5
5        4 building4
6        6 building6
```

QUESTIONS



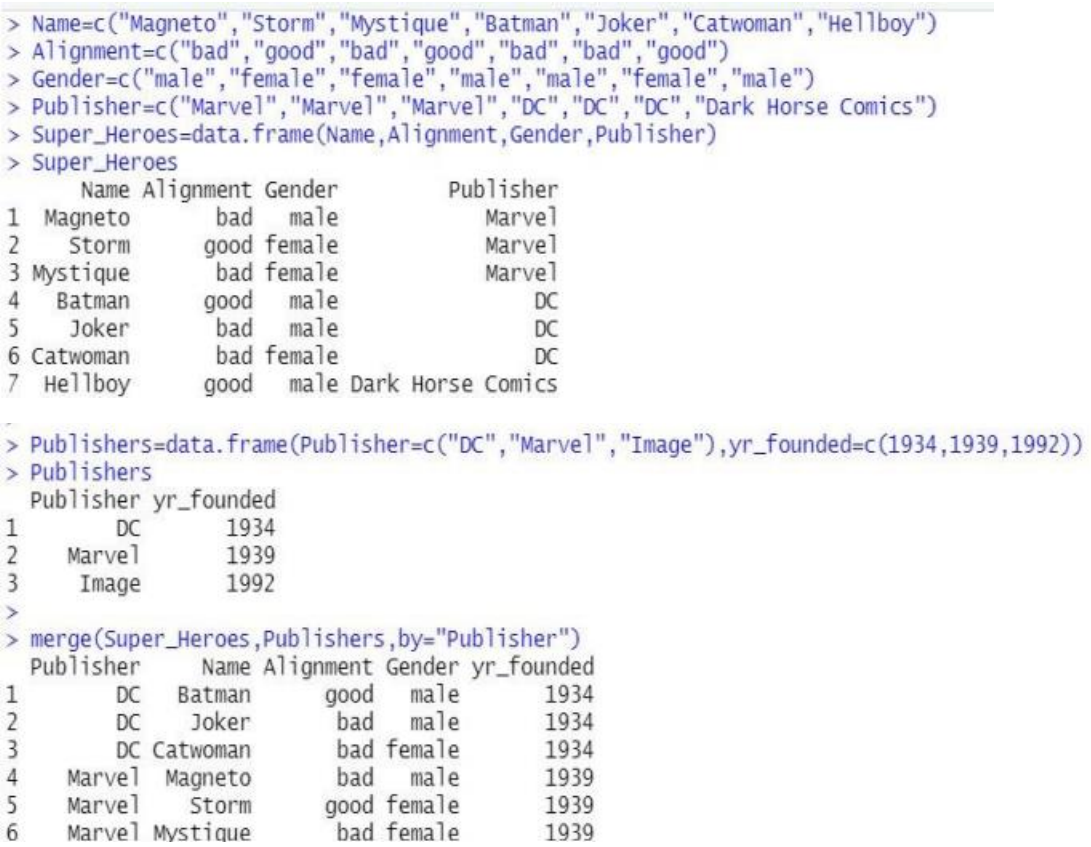
- Apply different join operations on the tables given below. Write the expected outputs and compare them with the outputs obtained by R commands

Super Heroes

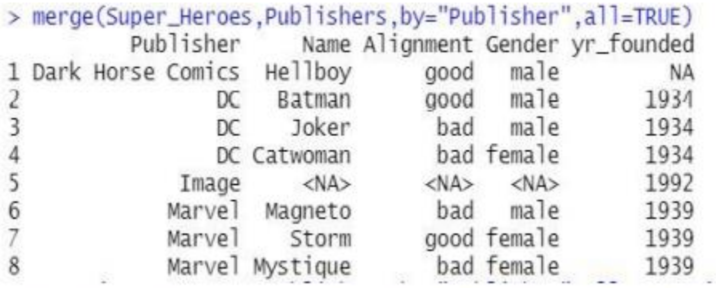
Name	Alignment	Gender	Publisher
Magneto	bad	male	Marvel
Storm	good	female	Marvel
Mystique	bad	female	Marvel
Batman	good	male	DC
Joker	bad	male	DC
Catwoman	bad	female	DC
Hellboy	good	male	Dark Horse Comics

Publishers

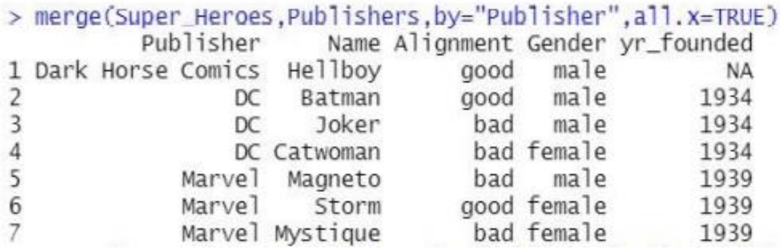
publisher	yr_founded
DC	1934
Marvel	1939
Image	1992



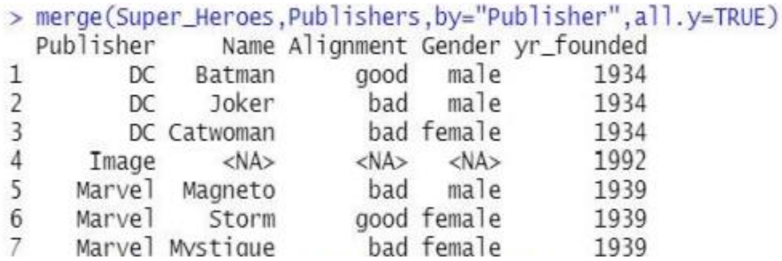
```
> Name=c("Magneto","Storm","Mystique","Batman","Joker","Catwoman","Hellboy")
> Alignment=c("bad","good","bad","good","bad","bad","good")
> Gender=c("male","female","female","male","male","female","male")
> Publisher=c("Marvel","Marvel","Marvel","DC","DC","DC","Dark Horse Comics")
> Super_Heroes=data.frame(Name,Alignment,Gender,Publisher)
> Super_Heroes
      Name Alignment Gender         Publisher
1  Magneto       bad   male            Marvel
2    Storm      good female            Marvel
3 Mystique       bad female            Marvel
4   Batman      good   male                DC
5    Joker       bad   male                DC
6 Catwoman       bad female                DC
7  Hellboy      good   male Dark Horse Comics

> Publishers=data.frame(Publisher=c("DC","Marvel","Image"),yr_founded=c(1934,1939,1992))
> Publishers
  Publisher yr_founded
1        DC       1934
2    Marvel       1939
3     Image       1992
>
> merge(Super_Heroes,Publishers,by="Publisher")
  Publisher     Name Alignment Gender yr_founded
1        DC   Batman      good   male       1934
2        DC    Joker       bad   male       1934
3        DC Catwoman       bad female       1934
4    Marvel  Magneto       bad   male       1939
5    Marvel    Storm      good female       1939
6    Marvel Mystique       bad female       1939
```



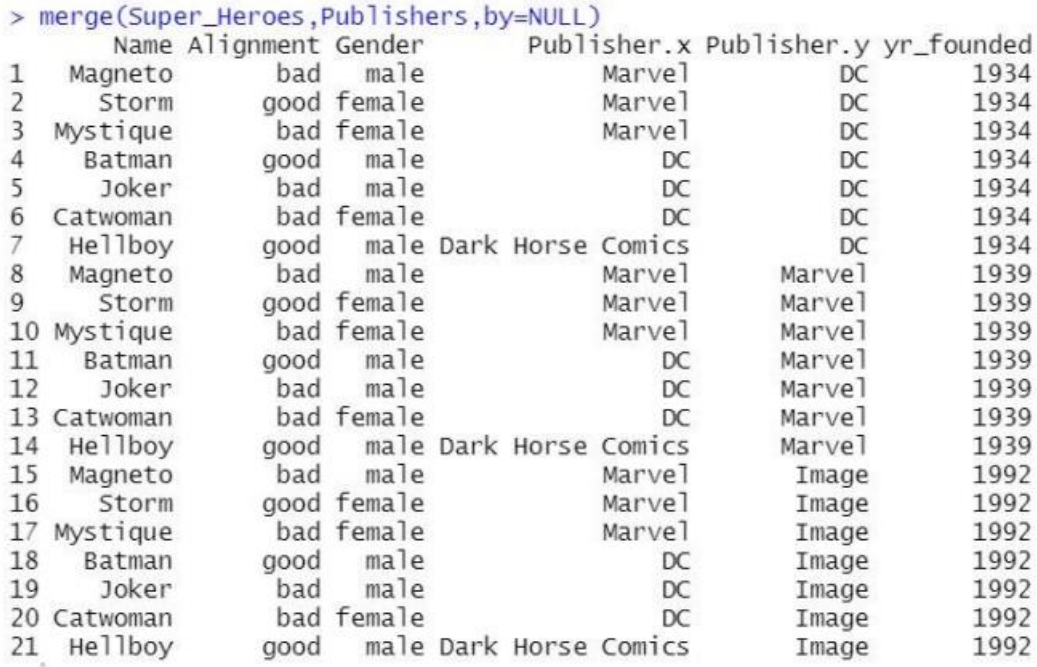
```
> merge(Super_Heroes,Publishers,by="Publisher",all=TRUE)
          Publisher     Name Alignment Gender yr_founded
1 Dark Horse Comics  Hellboy      good   male         NA
2                DC   Batman      good   male       1934
3                DC    Joker       bad   male       1934
4                DC Catwoman       bad female       1934
5             Image     <NA>      <NA>   <NA>       1992
6            Marvel  Magneto       bad   male       1939
7            Marvel    Storm      good female       1939
8            Marvel Mystique       bad female       1939
```



```
> merge(Super_Heroes,Publishers,by="Publisher",all.x=TRUE)
          Publisher     Name Alignment Gender yr_founded
1 Dark Horse Comics  Hellboy      good   male         NA
2                DC   Batman      good   male       1934
3                DC    Joker       bad   male       1934
4                DC Catwoman       bad female       1934
5            Marvel  Magneto       bad   male       1939
6            Marvel    Storm      good female       1939
7            Marvel Mystique       bad female       1939
```



```
> merge(Super_Heroes,Publishers,by="Publisher",all.y=TRUE)
  Publisher     Name Alignment Gender yr_founded
1        DC   Batman      good   male       1934
2        DC    Joker       bad   male       1934
3        DC Catwoman       bad female       1934
4     Image     <NA>      <NA>   <NA>       1992
5    Marvel  Magneto       bad   male       1939
6    Marvel    Storm      good female       1939
7    Marvel Mystique       bad female       1939
```

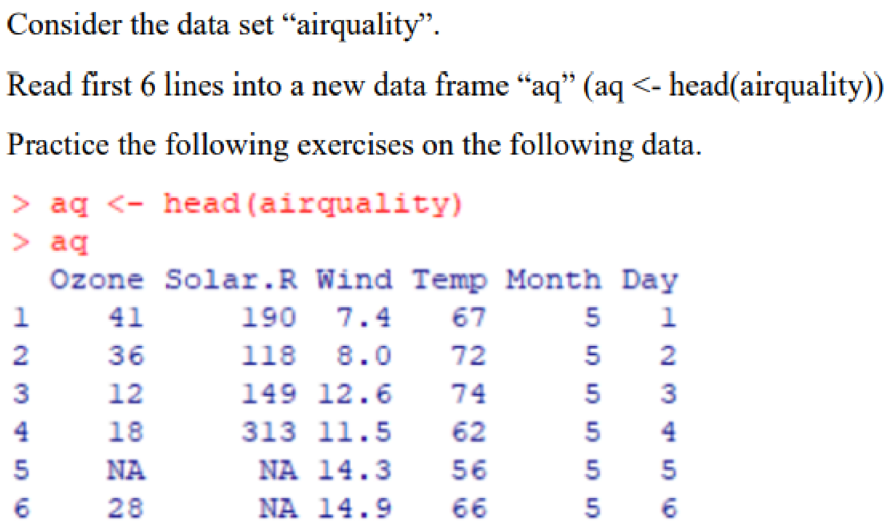


```
> merge(Super_Heroes,Publishers,by=NULL)
       Name Alignment Gender       Publisher.x Publisher.y yr_founded
1   Magneto       bad   male            Marvel          DC       1934
2     Storm      good female            Marvel          DC       1934
3  Mystique       bad female            Marvel          DC       1934
4    Batman      good   male                DC          DC       1934
5     Joker       bad   male                DC          DC       1934
6  Catwoman       bad female                DC          DC       1934
7   Hellboy      good   male Dark Horse Comics          DC       1934
8   Magneto       bad   male            Marvel      Marvel       1939
9     Storm      good female            Marvel      Marvel       1939
10 Mystique       bad female            Marvel      Marvel       1939
11   Batman      good   male                DC      Marvel       1939
12    Joker       bad   male                DC      Marvel       1939
13 Catwoman       bad female                DC      Marvel       1939
14  Hellboy      good   male Dark Horse Comics      Marvel       1939
15  Magneto       bad   male            Marvel       Image       1992
16    Storm      good female            Marvel       Image       1992
17 Mystique       bad female            Marvel       Image       1992
18   Batman      good   male                DC       Image       1992
19    Joker       bad   male                DC       Image       1992
20 Catwoman       bad female                DC       Image       1992
21  Hellboy      good   male Dark Horse Comics       Image       1992
```

Data Import and Export:

EXPORT DATA

-- EXERCISES --



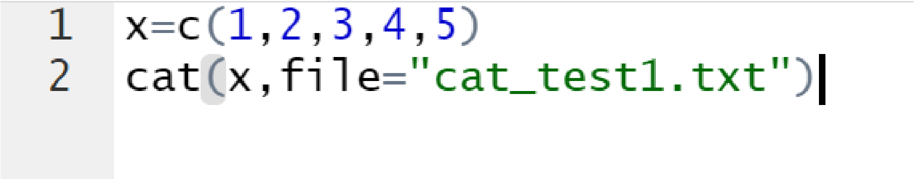
Consider the data set "airquality".

Read first 6 lines into a new data frame "aq" (aq <- head(airquality))

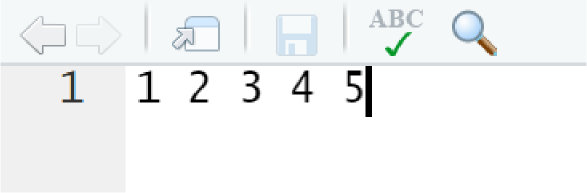
Practice the following exercises on the following data.

```
> aq <- head(airquality)
> aq
  Ozone Solar.R Wind Temp Month Day
1    41     190  7.4   67     5   1
2    36     118  8.0   72     5   2
3    12     149 12.6   74     5   3
4    18     313 11.5   62     5   4
5    NA      NA 14.3   56     5   5
6    28      NA 14.9   66     5   6
```

Write a command to export (store/save) data into the file cat_test1.txt (Use only two arguments). After creating file check the output.

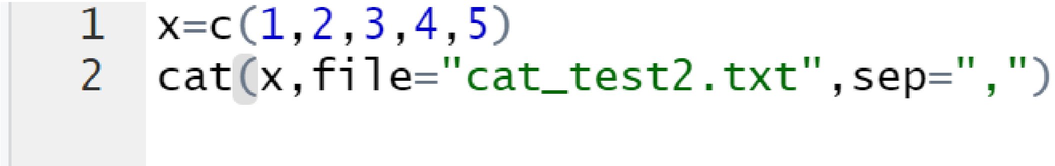


```
x=c(1,2,3,4,5)
cat(x,file="cat_test1.txt")
```

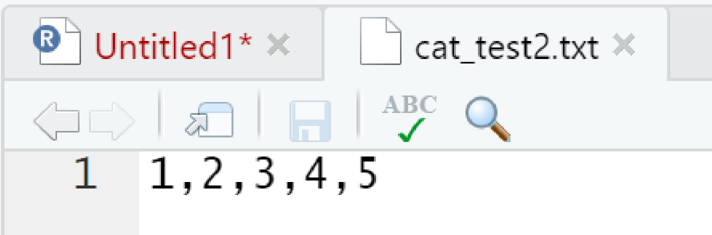


```
1 2 3 4 5
```

Write a command to export data into the file cat_test2.txt. Use separator as comma

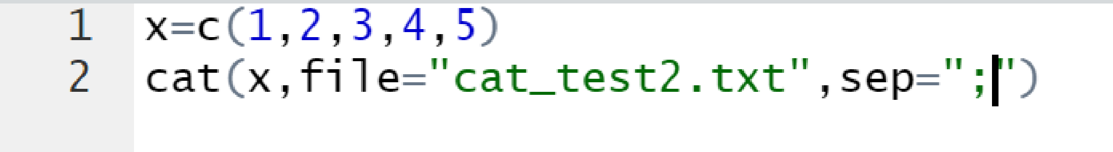


```
x=c(1,2,3,4,5)
cat(x,file="cat_test2.txt",sep=",")
```

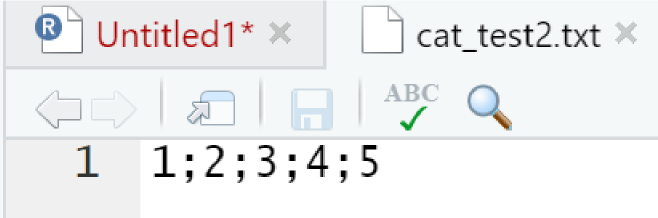


```
1,2,3,4,5
```

Write a command to export data into the file cat_test3.txt. Use separator as semi colon



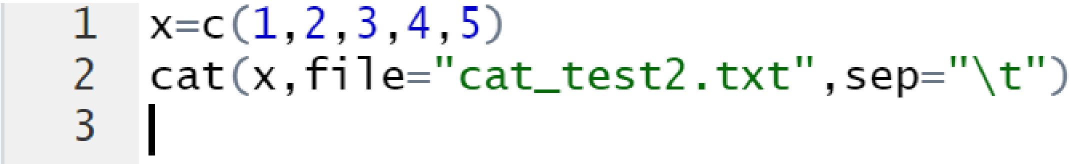
```
x=c(1,2,3,4,5)
cat(x,file="cat_test2.txt",sep=";")
```



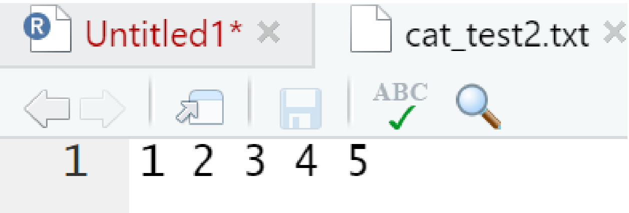
```
1;2;3;4;5
```

Write a command to export data into the file cat_test3.txt. Use separator as tab

(use \t to insert tab)



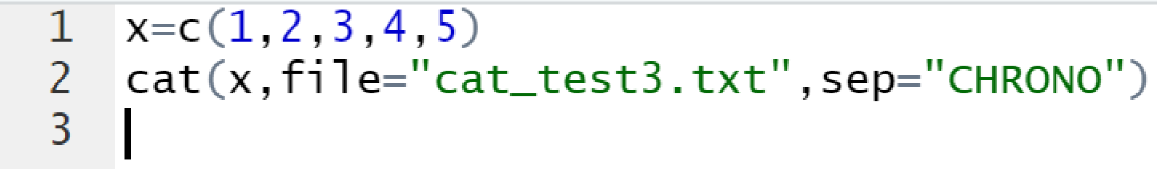
```
x=c(1,2,3,4,5)
cat(x,file="cat_test2.txt",sep="\t")

```



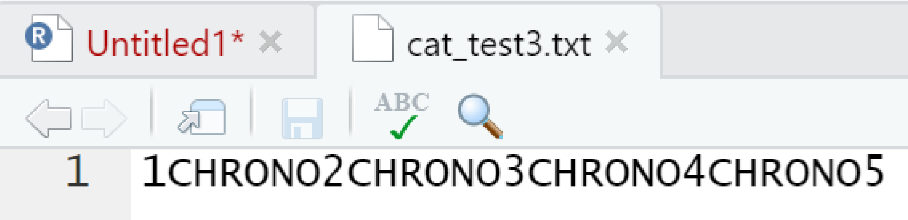
```
1	2	3	4	5
```

Can a separator be any string that you want to insert as delim?? Experiment it.

Ans: Yes



```
x=c(1,2,3,4,5)
cat(x,file="cat_test3.txt",sep="CHRONO")

```



```
1CHRONO2CHRONO3CHRONO4CHRONO5
```

*) What are Delimiters? What is the need of delimiters?

A delimiter is a sequence of one or more characters used to separate text strings. The most popular delimiters are commas (,), semicolon (;), braces ({}), pipes (|) and slashes (/). Quickly and easily convert column data to any separator character with our online delimiter tool.

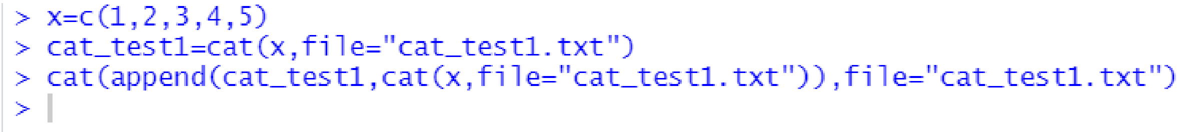
*) What is the difference between over writing and appending?

Overwriting: If you replace the class name with your own name then you must define your own methods for each generic that could be called with it unless the default method of that generic is ok.

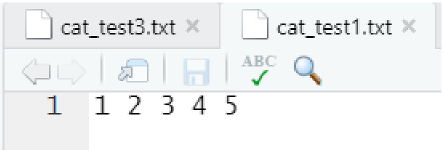
Appending: Normally you don't want to place your class after the existing class. If you did

then it would only get called if there were no data.frame method.

*) Write a command to append the same data into the file cat_test1.txt. After creating file check the output.

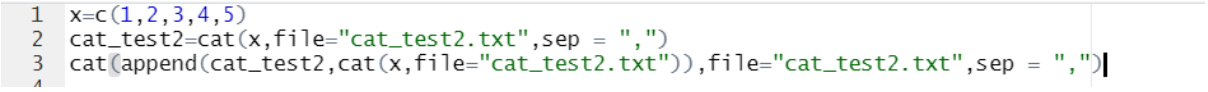


```
> x=c(1,2,3,4,5)
> cat_test1=cat(x,file="cat_test1.txt")
> cat(append(cat_test1,cat(x,file="cat_test1.txt")),file="cat_test1.txt")
> 
```

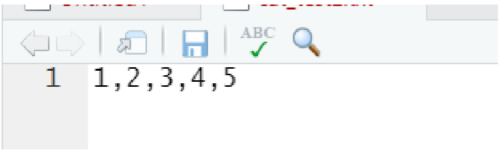


```
1 2 3 4 5
```

*)Write a command to append the same data into the file cat_test2.txt. Use separator as comma



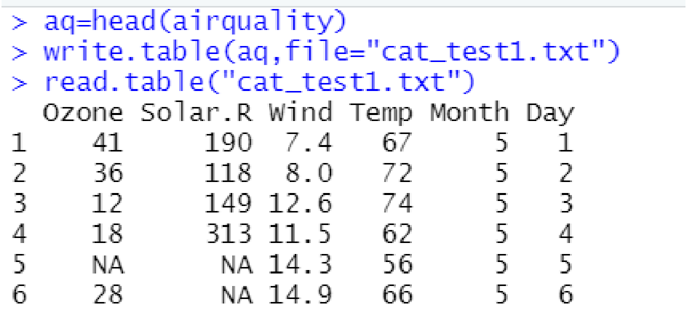
```
x=c(1,2,3,4,5)
cat_test2=cat(x,file="cat_test2.txt",sep = ",")
cat(append(cat_test2,cat(x,file="cat_test2.txt")),file="cat_test2.txt",sep = ",")
```



```
1,2,3,4,5
```

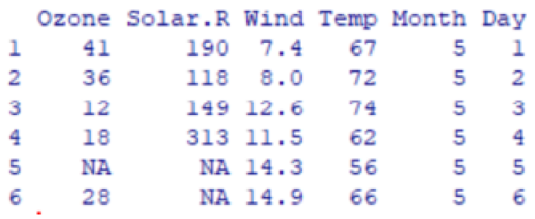
Using read.table command:

Write a command to read the data from cat_test1.txt. Display the output.

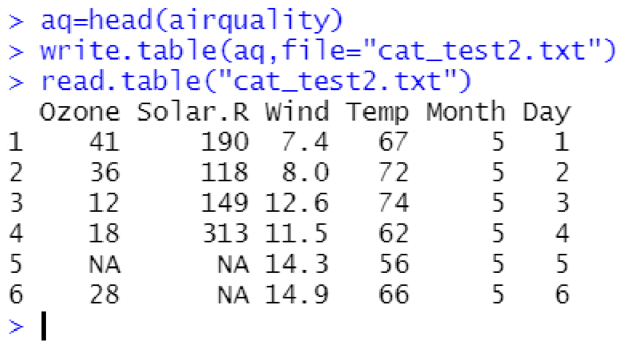


```
> aq=head(airquality)
> write.table(aq,file="cat_test1.txt")
> read.table("cat_test1.txt")
  Ozone Solar.R Wind Temp Month Day
1    41     190  7.4   67     5   1
2    36     118  8.0   72     5   2
3    12     149 12.6   74     5   3
4    18     313 11.5   62     5   4
5    NA      NA 14.3   56     5   5
6    28      NA 14.9   66     5   6
```

Write a command to read the data from cat_test2.txt. Display the output. Modify your command to get the output as given below

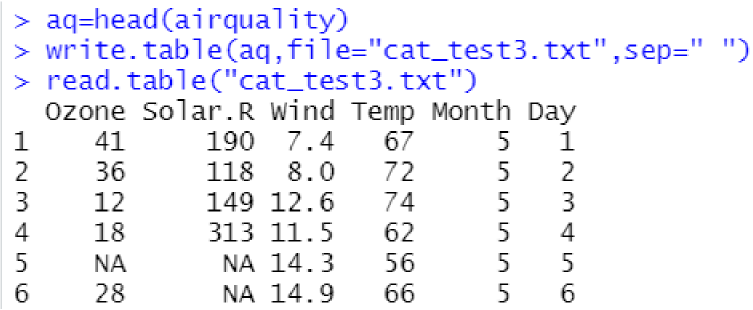


```
  Ozone Solar.R Wind Temp Month Day
1    41     190  7.4   67     5   1
2    36     118  8.0   72     5   2
3    12     149 12.6   74     5   3
4    18     313 11.5   62     5   4
5    NA      NA 14.3   56     5   5
6    28      NA 14.9   66     5   6
```



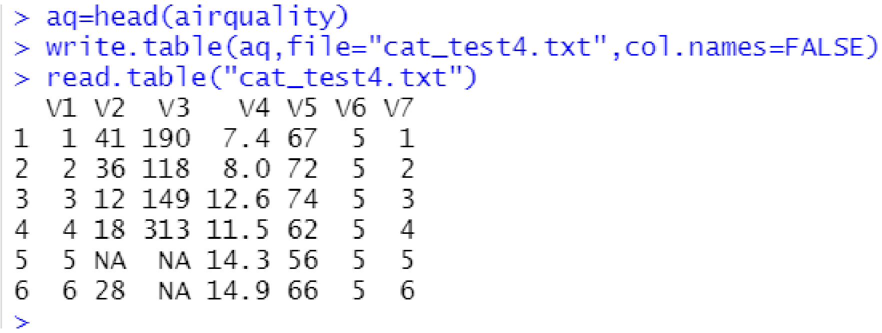
```
> aq=head(airquality)
> write.table(aq,file="cat_test2.txt")
> read.table("cat_test2.txt")
  Ozone Solar.R Wind Temp Month Day
1    41     190  7.4   67     5   1
2    36     118  8.0   72     5   2
3    12     149 12.6   74     5   3
4    18     313 11.5   62     5   4
5    NA      NA 14.3   56     5   5
6    28      NA 14.9   66     5   6
> 
```

Write a command to read the data from remaining files and display the output as in the above figure.



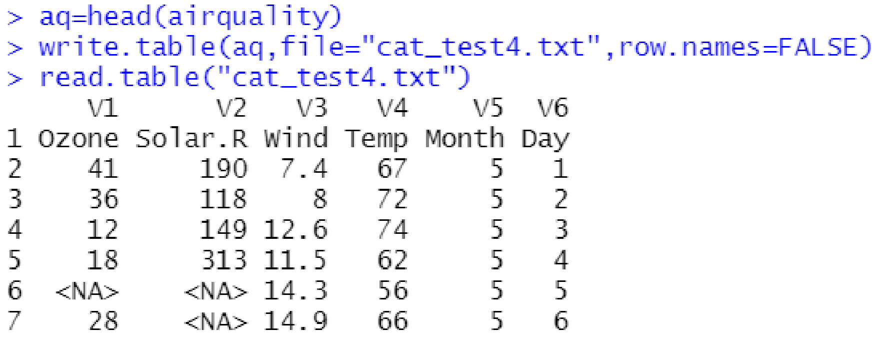
```
> aq=head(airquality)
> write.table(aq,file="cat_test3.txt",sep=" ")
> read.table("cat_test3.txt")
  Ozone Solar.R Wind Temp Month Day
1    41     190  7.4   67     5   1
2    36     118  8.0   72     5   2
3    12     149 12.6   74     5   3
4    18     313 11.5   62     5   4
5    NA      NA 14.3   56     5   5
6    28      NA 14.9   66     5   6
```

Write a command to read the data without column names



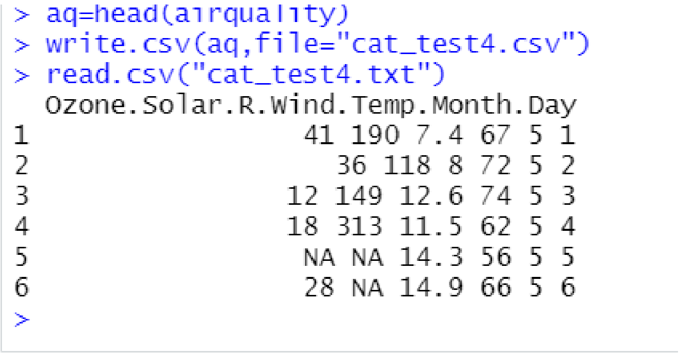
```
> aq=head(airquality)
> write.table(aq,file="cat_test4.txt",col.names=FALSE)
> read.table("cat_test4.txt")
  V1 V2  V3   V4 V5 V6 V7
1  1 41 190  7.4 67  5  1
2  2 36 118  8.0 72  5  2
3  3 12 149 12.6 74  5  3
4  4 18 313 11.5 62  5  4
5  5 NA  NA 14.3 56  5  5
6  6 28  NA 14.9 66  5  6
>
```

Write a command to read the data without row names



```
> aq=head(airquality)
> write.table(aq,file="cat_test4.txt",row.names=FALSE)
> read.table("cat_test4.txt")
     V1      V2   V3   V4    V5  V6
1 Ozone Solar.R Wind Temp Month Day
2    41     190  7.4   67     5   1
3    36     118    8   72     5   2
4    12     149 12.6   74     5   3
5    18     313 11.5   62     5   4
6  <NA>    <NA> 14.3   56     5   5
7    28    <NA> 14.9   66     5   6
```

Using read.csv command:



```
> aq=head(airquality)
> write.csv(aq,file="cat_test4.csv")
> read.csv("cat_test4.txt")
  Ozone.Solar.R.Wind.Temp.Month.Day
1                 41 190 7.4 67 5 1
2                   36 118 8 72 5 2
3                12 149 12.6 74 5 3
4                18 313 11.5 62 5 4
5                  NA NA 14.3 56 5 5
6                  28 NA 14.9 66 5 6
>
```