1)Write R command to create an empty data frame.

```
> emptydf<-data.frame()
> print(emptydf)
data frame with 0 columns and 0 rows
```

2)Write R command to create a data frame from four given vectors.

```
> ename<-c('akshay','jignesh','yash')
> eage<-c(21,20,19)
> epost<-c('manager','developer','manager')
> gender<-c('male','male','male')
> edataFrame<-
data.frame(ename=ename,eage=eage,epost=epost,gender=gender)
> print(edataFrame)
    ename eage     epost gender
1  akshay   21   manager   male
2 jignesh   20 developer   male
3    yash   19   manager   male
```

3)Write R command to get the structure of a given data frame.

```
> ename<-c('akshay','jignesh','yash')
> eage<-c(21,20,19)
> epost<-c('manager','developer','manager')
> gender<-c('male','male','male')
> edataFrame<-
data.frame(ename=ename,eage=eage,epost=epost,gender=gender)
> print(edataFrame)
    ename eage     epost gender
1  akshay   21   manager   male
2 jignesh   20 developer   male
3    yash   19   manager   male
> print(str(edataFrame))
'data.frame':   3 obs. of  4 variables:
 $ ename : Factor w/ 3 levels "akshay","jignesh",..: 1 2 3
 $ eage  : num  21 20 19
 $ epost : Factor w/ 2 levels "developer","manager": 2 1 2
 $ gender: Factor w/ 1 level "male": 1 1 1
NULL
```

4)Write R command to get the statistical summary and nature of the data of a given data frame.

```
> ename<-c('akshay','jignesh','yash')
> eage<-c(21,20,19)
> epost<-c('manager','developer','manager')
> gender<-c('male','male','male')
```

```
> edataFrame<-
data.frame(ename=ename,eage=eage,epost=epost,gender=gender)
> print(edataFrame)
    ename eage     epost gender
1  akshay   21   manager   male
2 jignesh   20 developer   male
3    yash   19   manager   male
> print(summary(edataFrame))
    ename        eage            epost    gender
 akshay :1   Min.   :19.0   developer:1   male:3
 jignesh:1   1st Qu.:19.5   manager  :2
 yash   :1   Median :20.0
            Mean   :20.0
            3rd Qu.:20.5
            Max.   :21.0
> class(edataFrame)
[1] "data.frame"
```

5)Write R command to extract specific column from a data frame using column name.

```
> ename<-c('akshay','jignesh','yash')
> eage<-c(21,20,19)
> epost<-c('manager','developer','manager')
> gender<-c('male','male','male')
> edataFrame<-
data.frame(ename=ename,eage=eage,epost=epost,gender=gender)
> print(edataFrame)
    ename eage     epost gender
1  akshay   21   manager   male
2 jignesh   20 developer   male
3    yash   19   manager   male
> print(edataFrame['epost'])
     epost
1   manager
2 developer
3   manager
```

6)Write R command to extract first two rows from a given data frame.

```
> ename<-c('akshay','jignesh','yash')
> eage<-c(21,20,19)
> epost<-c('manager','developer','manager')
> gender<-c('male','male','male')
```

```
> edataFrame<-
data.frame(ename=ename,eage=eage,epost=epost,gender=gender)
> print(edataFrame)
    ename eage     epost gender
1  akshay   21   manager   male
2 jignesh   20 developer   male
3    yash   19   manager   male
> print(edataFrame[1:2,])
    ename eage     epost gender
1  akshay   21   manager   male
2 jignesh   20 developer   male
```

7)Write R command to extract 3 rd and 5 th rows with 1 st and 3 rd columns from a given data frame.

```
> ename<-c('akshay','jignesh','yash')
> eage<-c(21,20,19)
> epost<-c('manager','developer','manager')
> gender<-c('male','male','male')
> edataFrame<-
data.frame(ename=ename,eage=eage,epost=epost,gender=gender)
> print(edataFrame)
    ename eage     epost gender
1  akshay   21   manager   male
2 jignesh   20 developer   male
3    yash   19   manager   male
> print(edataFrame[c(3,5),c(1,3)])
   ename   epost
3   yash manager
NA  <NA>    <NA>
```

8)Write R command to add a new column in a given data frame.

```
> edataFrame$experience<-c(1,2,4)
> print(edataFrame)
    ename eage     epost gender experience
1  akshay   21   manager   male          1
2 jignesh   20 developer   male          2
3    yash   19   manager   male          4
```

9)Write R command to add new row(s) to an existing data frame.

```
> newRow<-
data.frame(ename='deepak',eage=19,epost='hr_manager',gender='male',expe
rience=7)
> edataFrame<-rbind(edataFrame,newRow)
```

```
> print(edataFrame)
    ename eage      epost gender experience
1  akshay    21    manager   male          1
2 jignesh    20  developer   male          2
3    yash    19    manager   male          4
4  deepak    19 hr_manager   male          7
```

10)Write R command to drop column(s) by name from a given data frame.

```
> edataFrame[,!(names(edataFrame) %in% 'gender')]
    ename eage      epost experience
1  akshay    21    manager          1
2 jignesh    20  developer          2
3    yash    19    manager          4
4  deepak    19 hr_manager          7
```

11)Write R command to drop row(s) by number from a given data frame.

```
> data = data.frame(
+     name = c('akshay', 'hardik', 'yash', 'jigo', 'nishant', 'darsh',
'nikhil','akhil','bhavin', 'kevin'),
+     age = c(21, 19, 16, 16, 19, 20, 14, 13, 8, 19),
+     percentage = c(81, 83, 82, 83, 82, 83, 81, 81, 82, 81),
+     gender = c('male', 'male', 'male', 'male', 'male', 'male',
'male', 'male', 'male', 'male')
+ )
> print("data frame is")
     [1] "data frame is"
> print(data)

     name age percentage gender
1   akshay  21         81   male
2   hardik  19         83   male
3     yash  16         82   male
4     jigo  16         83   male
5  nishant  19         82   male
6    darsh  20         83   male
7   nikhil  14         81   male
8    akhil  13         81   male
9   bhavin   8         82   male
10   kevin  19         81   male

> data <-data[-c(2, 4, 6),]
> print(data)
     name age percentage gender
1   akshay  21         81   male
3     yash  16         82   male
```

```
5  nishant  19         82   male
7   nikhil  14         81   male
8    akhil  13         81   male
9   bhavin   8         82   male
10   kevin  19         81   male
```

12)Write R command to sort a given data frame by multiple column(s).

```
> data = data.frame(
+     name = c('akshay', 'hardik', 'yash', 'jigo', 'nishant', 'darsh',
'nikhil','akhil','bhavin', 'kevin'),
+     age = c(21, 19, 16, 16, 19, 20, 14, 13, 8, 19),
+     percentage = c(81, 83, 82, 83, 82, 83, 81, 81, 82, 81),
+     gender = c('male', 'male', 'male', 'male', 'male', 'male',
'male', 'male', 'male', 'male')
+ )
> print("data frame is")
     [1] "data frame is"
> print(data)

     name age percentage gender
1   akshay  21         81   male
2   hardik  19         83   male
3     yash  16         82   male
4     jigo  16         83   male
5  nishant  19         82   male
6    darsh  20         83   male
7   nikhil  14         81   male
8    akhil  13         81   male
9   bhavin   8         82   male
10   kevin  19         81   male

> data = data[with(data, order(percentage,name)), ]
> print(data)
     name age percentage gender
8    akhil  13         81   male
1   akshay  21         81   male
10   kevin  19         81   male
7   nikhil  14         81   male
9   bhavin   8         82   male
5  nishant  19         82   male
3     yash  16         82   male
6    darsh  20         83   male
2   hardik  19         83   male
4     jigo  16         83   male
```

13)Write R command to create inner, outer, left, right join(merge) from given two data frames.

```
> dataframe1 = data.frame(numid = c(112, 141, 110, 111))
> dataframe2 = data.frame(numid = c(113, 115, 111, 112))
> print("Left outer Join:")
[1] "Left outer Join:"
> result = merge(dataframe1, dataframe2, by = "numid", all.x = TRUE)
> print(result)
  numid
1   110
2   111
3   112
4   141
> print("Right outer Join:")
[1] "Right outer Join:"
> result = merge(dataframe1,dataframe2, by = "numid", all.y = TRUE)
> print(result)
  numid
1   111
2   112
3   113
4   115
> print("Outer Join:")
[1] "Outer Join:"
> result = merge(dataframe1,dataframe2, by = "numid", all = TRUE)
> print(result)
  numid
1   110
2   111
3   112
4   113
5   115
6   141
> print("inner Join:")
[1] "inner Join:"
> result = merge(dataframe1, dataframe2, by = "numid")
> print(result)
  numid
1   111
2   112
```

14)Write R command to replace NA values with 3 in a given data frame.

```
> data = data.frame(
+     name = c('akshay', 'hardik', 'yash', 'jigo', 'nishant', 'darsh',
'nikhil','akhil','bhavin', 'kevin'),
+     age = c(21, 19, 16, 16, 19, 20, 14, 13, 8, 19),
+     percentage = c(81, NA, 82,NA,NA, 83,NA, 81,NA, 81),
+     gender = c('male', 'male', 'male', 'male', 'male', 'male',
'male', 'male', 'male', 'male')
+ )
> print("data frame is:")
[1] "data frame is:"
> print(data)
     name age percentage gender
1   akshay  21         81   male
2   hardik  19         NA   male
3     yash  16         82   male
4     jigo  16         NA   male
5  nishant  19         NA   male
6    darsh  20         83   male
7   nikhil  14         NA   male
8    akhil  13         81   male
9   bhavin   8         NA   male
10   kevin  19         81   male
> data[is.na(data)] = 3
> print("remove NA and replace 3")
[1] "remove NA and replace 3"
> print(data)
     name age percentage gender
1   akshay  21         81   male
2   hardik  19          3   male
3     yash  16         82   male
4     jigo  16          3   male
5  nishant  19          3   male
6    darsh  20         83   male
7   nikhil  14          3   male
8    akhil  13         81   male
9   bhavin   8          3   male
10   kevin  19         81   male
```

15)Write R command to change a column name of a given data frame.

```
> data = data.frame(
+     name = c('akshay', 'hardik', 'yash', 'jigo', 'nishant', 'darsh',
'nikhil','akhil','bhavin', 'kevin'),
```

```
+     age = c(21, 19, 16, 16, 19, 20, 14, 13, 8, 19),
+     percentage = c(81, NA, 82,NA,NA, 83,NA, 81,NA, 81),
+     gender = c('male', 'male', 'male', 'male', 'male', 'male',
'male', 'male', 'male', 'male')
+ )
> print("data frame is:")
[1] "data frame is:"
> print(data)
     name age percentage gender
1   akshay  21         81   male
2   hardik  19         NA   male
3     yash  16         82   male
4     jigo  16         NA   male
5  nishant  19         NA   male
6    darsh  20         83   male
7   nikhil  14         NA   male
8    akhil  13         81   male
9   bhavin   8         NA   male
10   kevin  19         81   male
> print("Change column name name to friend name in dataframe:")
[1] "Change column name name to friend name in dataframe:"
> colnames(data)[which(names(data) == "name")] = "friend.name"
> print(data)
   friend.name age percentage gender
1       akshay  21         81   male
2       hardik  19          3   male
3         yash  16         82   male
4         jigo  16          3   male
5      nishant  19          3   male
6        darsh  20         83   male
7       nikhil  14          3   male
8        akhil  13         81   male
9       bhavin   8          3   male
10       kevin  19         81   male
```

16)Write R command to change more than one column name of a given data frame.

```
> data = data.frame(
+     name = c('akshay', 'hardik', 'yash', 'jigo', 'nishant', 'darsh',
'nikhil','akhil','bhavin', 'kevin'),
+     age = c(21, 19, 16, 16, 19, 20, 14, 13, 8, 19),
+     percentage = c(81, NA, 82,NA,NA, 83,NA, 81,NA, 81),
```

```
+     gender = c('male', 'male', 'male', 'male', 'male', 'male',
'male', 'male', 'male', 'male')
+ )
> print("data frame is:")
[1] "data frame is:"
> print(data)
     name age percentage gender
1   akshay  21         81   male
2   hardik  19         NA   male
3     yash  16         82   male
4     jigo  16         NA   male
5  nishant  19         NA   male
6    darsh  20         83   male
7   nikhil  14         NA   male
8    akhil  13         81   male
9   bhavin   8         NA   male
10   kevin  19         81   male
> print("Change column name friend.name to name and percentage to avg
dataframe:")
[1] "Change column name friend.name to name and percentage to avg
dataframe:"
> colnames(data)[which(names(data) == "friend.name")] = "name"
> colnames(data)[which(names(data) == "percentage")] = "avg"
> print("dataframe is:")
[1] "dataframe is:"
> print(data)
     name age avg gender
1   akshay  21  81   male
2   hardik  19   3   male
3     yash  16  82   male
4     jigo  16   3   male
5  nishant  19   3   male
6    darsh  20  83   male
7   nikhil  14   3   male
8    akhil  13  81   male
9   bhavin   8   3   male
10   kevin  19  81   male
```

17)Write R command to select some random rows from a given data frame.

```
> dataframe1 = data.frame(
+     name = c('yash', 'jigo', 'nikhil', 'nishant', 'akshay'),
+     age = c(21, 91, 16, 22, 19),
+     class = c("mca","mca","mca","mca","mca"),
```

```
+     qualify = c('yes', 'no', 'yes', 'yes', 'no')
+ )
> print("Original dataframe:")
[1] "Original dataframe:"
> print(dataframe1)
    name age class qualify
1   yash  21   mca     yes
2   jigo  91   mca      no
3 nikhil  16   mca     yes
4 nishant  22   mca     yes
5 akshay  19   mca      no
> print("Select three random rows of the said dataframe:")
[1] "Select three random rows of the said dataframe:"
> print(dataframe1[sample(nrow(dataframe1), 3),])
    name age class qualify
3 nikhil  16   mca     yes
2   jigo  91   mca      no
4 nishant  22   mca     yes
```

18)Write R command to reorder an given data frame by column name.

```
> data = data.frame(
+     name = c('akshay', 'hardik', 'yash', 'jigo', 'nishant', 'darsh',
'nikhil','akhil','bhavin', 'kevin'),
+     age = c(21, 19, 16, 16, 19, 20, 14, 13, 8, 19),
+     percentage = c(81, NA, 82,NA,NA, 83,NA, 81,NA, 81),
+     gender = c('male', 'male', 'male', 'male', 'male', 'male',
'male', 'male', 'male', 'male')
+ )
> print("data frame is:")
[1] "data frame is:"
> print(data)
     name age percentage gender
1   akshay  21         81   male
2   hardik  19         NA   male
3     yash  16         82   male
4     jigo  16         NA   male
5  nishant  19         NA   male
6    darsh  20         83   male
7   nikhil  14         NA   male
8    akhil  13         81   male
9   bhavin   8         NA   male
10   kevin  19         81   male
```

```
> print("reorder by column name:")
[1] "reorder by column name:"
> data = data[c("name", "gender","age", "avg")]
> print(data)
     name gender age avg
1   akshay   male  21  81
2   hardik   male  19   3
3     yash   male  16  82
4     jigo   male  16   3
5  nishant   male  19   3
6    darsh   male  20  83
7   nikhil   male  14   3
8    akhil   male  13  81
9   bhavin   male   8   3
10   kevin   male  19  81
```

22)Write R command to save the information of a data frame in a file and display the information of the file.

```
> data = data.frame(
+          name = c('akshay', 'hardik', 'yash', 'jigo', 'nishant',
'darsh', 'nikhil','akhil','bhavin', 'kevin'),
+          age = c(21, 19, 16, 16, 19, 20, 14, 13, 8, 19),
+          percentage = c(81, NA, 82,NA,NA, 83,NA, 81,NA, 81),
+          gender = c('male', 'male', 'male', 'male', 'male', 'male',
'male', 'male', 'male', 'male')
+      )
> print("data frame is:")
[1] "data frame is:"
> print(data)
     name age percentage gender
1   akshay  21         81   male
2   hardik  19         NA   male
3     yash  16         82   male
4     jigo  16         NA   male
5  nishant  19         NA   male
6    darsh  20         83   male
7   nikhil  14         NA   male
8    akhil  13         81   male
9   bhavin   8         NA   male
10   kevin  19         81   male
> save(data,file="data.rda")
> load("data.rda")
> file.info("data.rda")
```

```
        size isdir mode               mtime               ctime
data.rda  325 FALSE  664 2019-09-10 16:08:23 2019-09-10 16:08:23
                     atime  uid  gid  uname grname
data.rda 2019-09-10 16:08:36 1000 1000 akshay akshay
```

23)Write R command to count the number of NA values in a data frame column.

```
> data = data.frame(
+          name = c('akshay', 'hardik', 'yash', 'jigo', 'nishant',
'darsh', 'nikhil','akhil','bhavin', 'kevin'),
+          age = c(21, 19, 16, 16, 19, 20, 14, 13, 8, 19),
+          percentage = c(81, NA, 82,NA,NA, 83,NA, 81,NA, 81),
+          gender = c('male', 'male', 'male', 'male', 'male', 'male',
'male', 'male', 'male', 'male')
+      )
> print("dataframe is:")
[1] "dataframe is:"
> print("dataframe is:")
[1] "dataframe is:"
> print(data)
      name age percentage gender
1   akshay  21         81   male
2   hardik  19         NA   male
3     yash  16         82   male
4     jigo  16         NA   male
5  nishant  19         NA   male
6    darsh  20         83   male
7   nikhil  14         NA   male
8    akhil  13         81   male
9   bhavin   8         NA   male
10   kevin  19         81   male
> print("The number of NA values in attempts column:")
[1] "The number of NA values in attempts column:"
> print(sum(is.na(data$percentage)))
[1] 5
```

24)Write R command to create a data frame using two given vectors and display the duplicated elements and unique rows of the said data frame.

```
> a = c(110,20,1120,102,410,150,201,310)
> b = c(110,320,110,201,20,150,30,3120)
> print("original data frame is:")
```

```
[1] "original data frame is:"
> ab = data.frame(a,b)
> print(ab)
     a    b
1  110  110
2   20  320
3 1120  110
4  102  201
5  410   20
6  150  150
7  201   30
8  310 3120
> print("Duplicate elements of data frame:")
[1] "Duplicate elements of data frame:"
> print(duplicated(ab))
[1] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
> print("Unique rows of data frame:")
[1] "Unique rows of data frame:"
> print(unique(ab))
     a    b
1  110  110
2   20  320
3 1120  110
4  102  201
5  410   20
6  150  150
7  201   30
8  310 3120
```

25)Subset the vector, “mtcars[,1]“, for values greater than “15.0“.

```
> subset(mtcars[,1], mtcars[,1] > 15.0)
 [1] 21.0 21.0 22.8 21.4 18.7 18.1 24.4 22.8 19.2 17.8 16.4 17.3 15.2
32.4 30.4
[16] 33.9 21.5 15.5 15.2 19.2 27.3 26.0 30.4 15.8 19.7 21.4
```

26)Subset the dataframe, “mtcars” for rows with “mpg” greater than , or equal to,21 miles per gallon.

```
> subset(mtcars, mpg >= 21)
               mpg cyl  disp  hp drat    wt  qsec vs am gear carb
Mazda RX4      21.0   6 160.0 110 3.90 2.620 16.46  0  1    4    4
Mazda RX4 Wag  21.0   6 160.0 110 3.90 2.875 17.02  0  1    4    4
```

```
Datsun 710     22.8   4 108.0  93 3.85 2.320 18.61  1  1    4    1
Hornet 4 Drive 21.4   6 258.0 110 3.08 3.215 19.44  1  0    3    1
Merc 240D      24.4   4 146.7  62 3.69 3.190 20.00  1  0    4    2
Merc 230       22.8   4 140.8  95 3.92 3.150 22.90  1  0    4    2
Fiat 128       32.4   4  78.7  66 4.08 2.200 19.47  1  1    4    1
Honda Civic    30.4   4  75.7  52 4.93 1.615 18.52  1  1    4    2
Toyota Corolla 33.9   4  71.1  65 4.22 1.835 19.90  1  1    4    1
Toyota Corona  21.5   4 120.1  97 3.70 2.465 20.01  1  0    3    1
Fiat X1-9      27.3   4  79.0  66 4.08 1.935 18.90  1  1    4    1
Porsche 914-2  26.0   4 120.3  91 4.43 2.140 16.70  0  1    5    2
Lotus Europa   30.4   4  95.1 113 3.77 1.513 16.90  1  1    5    2
Volvo 142E     21.4   4 121.0 109 4.11 2.780 18.60  1  1    4    2
```

27)Subset “mtcars” for rows wih “cyl” less than “6“, and “gear” exactly equal to “4“

```
> subset(mtcars, cyl < 6 & gear == 4)
               mpg cyl  disp  hp drat    wt  qsec vs am gear carb
Datsun 710     22.8   4 108.0  93 3.85 2.320 18.61  1  1    4    1
Merc 240D      24.4   4 146.7  62 3.69 3.190 20.00  1  0    4    2
Merc 230       22.8   4 140.8  95 3.92 3.150 22.90  1  0    4    2
Fiat 128       32.4   4  78.7  66 4.08 2.200 19.47  1  1    4    1
Honda Civic    30.4   4  75.7  52 4.93 1.615 18.52  1  1    4    2
Toyota Corolla 33.9   4  71.1  65 4.22 1.835 19.90  1  1    4    1
Fiat X1-9      27.3   4  79.0  66 4.08 1.935 18.90  1  1    4    1
Volvo 142E     21.4   4 121.0 109 4.11 2.780 18.60  1  1    4    2
```

28)Subset “mtcars” for rows greater than, or equal to, 21 miles per gallon. Also,select only the columns, “mpg” through “hp“.

```
> subset(mtcars, mtcars$mpg >= 21, select = mpg:hp)
               mpg cyl  disp  hp
Mazda RX4      21.0   6 160.0 110
Mazda RX4 Wag  21.0   6 160.0 110
Datsun 710     22.8   4 108.0  93
Hornet 4 Drive 21.4   6 258.0 110
Merc 240D      24.4   4 146.7  62
Merc 230       22.8   4 140.8  95
Fiat 128       32.4   4  78.7  66
Honda Civic    30.4   4  75.7  52
Toyota Corolla 33.9   4  71.1  65
Toyota Corona  21.5   4 120.1  97
Fiat X1-9      27.3   4  79.0  66
```

```
Porsche 914-2  26.0   4 120.3  91
Lotus Europa   30.4   4  95.1 113
Volvo 142E     21.4   4 121.0 109
```

29)Subset “airquality” for “Ozone” greater than “28“, or “Temp” greater than “70“ Return the first five rows.

```
> head(subset(airquality, Ozone > 28 | Temp > 70))
   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
11     7      NA  6.9   74     5  11
17    34     307 12.0   66     5  17
19    30     322 11.5   68     5  19
```

30)Subset “airquality” for “Ozone” greater than “28“, and “Temp” greater than “70“.Select the columns, “Ozone” and “Temp“. Return the first five rows.

```
> head(subset(airquality, Ozone > 28 & Temp > 70, select=c(Ozone,
Temp)))
   Ozone Temp
2     36   72
29    45   81
30   115   79
31    37   76
38    29   82
40    71   90
```

31)Subset the “CO2” dataframe for “Treatment” values of “chilled“, and “uptake” values greater that “15“. Remove the category, “conc“. Return the first 10 rows.

```
> head(subset(CO2, Treatment=="chilled" & uptake > 15, select=-conc),
10)
   Plant   Type Treatment uptake
23   Qc1 Quebec   chilled   24.1
24   Qc1 Quebec   chilled   30.3
25   Qc1 Quebec   chilled   34.6
26   Qc1 Quebec   chilled   32.5
27   Qc1 Quebec   chilled   35.4
28   Qc1 Quebec   chilled   38.7
30   Qc2 Quebec   chilled   27.3
31   Qc2 Quebec   chilled   35.0
32   Qc2 Quebec   chilled   38.8
```

```
33   Qc2 Quebec   chilled   38.6
```

32)Subset the “airquality” dataframe for rows without “Ozone” values of “NA“.

```
> subset(airquality, !is.na(Ozone))
    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
6      28      NA 14.9   66     5   6
7      23     299  8.6   65     5   7
8      19      99 13.8   59     5   8
9       8      19 20.1   61     5   9
11      7      NA  6.9   74     5  11
12     16     256  9.7   69     5  12
13     11     290  9.2   66     5  13
14     14     274 10.9   68     5  14
15     18      65 13.2   58     5  15
16     14     334 11.5   64     5  16
17     34     307 12.0   66     5  17
18      6      78 18.4   57     5  18
19     30     322 11.5   68     5  19
20     11      44  9.7   62     5  20
21      1       8  9.7   59     5  21
22     11     320 16.6   73     5  22
23      4      25  9.7   61     5  23
24     32      92 12.0   61     5  24
28     23      13 12.0   67     5  28
29     45     252 14.9   81     5  29
30    115     223  5.7   79     5  30
31     37     279  7.4   76     5  31
38     29     127  9.7   82     6   7
40     71     291 13.8   90     6   9
41     39     323 11.5   87     6  10
44     23     148  8.0   82     6  13
47     21     191 14.9   77     6  16
48     37     284 20.7   72     6  17
49     20      37  9.2   65     6  18
50     12     120 11.5   73     6  19
51     13     137 10.3   76     6  20
62    135     269  4.1   84     7   1
63     49     248  9.2   85     7   2
```

64	32	236	9.2	81	7	3
66	64	175	4.6	83	7	5
67	40	314	10.9	83	7	6
68	77	276	5.1	88	7	7
69	97	267	6.3	92	7	8
70	97	272	5.7	92	7	9
71	85	175	7.4	89	7	10
73	10	264	14.3	73	7	12
74	27	175	14.9	81	7	13
76	7	48	14.3	80	7	15
77	48	260	6.9	81	7	16
78	35	274	10.3	82	7	17
79	61	285	6.3	84	7	18
80	79	187	5.1	87	7	19
81	63	220	11.5	85	7	20
82	16	7	6.9	74	7	21
85	80	294	8.6	86	7	24
86	108	223	8.0	85	7	25
87	20	81	8.6	82	7	26
88	52	82	12.0	86	7	27
89	82	213	7.4	88	7	28
90	50	275	7.4	86	7	29
91	64	253	7.4	83	7	30
92	59	254	9.2	81	7	31
93	39	83	6.9	81	8	1
94	9	24	13.8	81	8	2
95	16	77	7.4	82	8	3
96	78	NA	6.9	86	8	4
97	35	NA	7.4	85	8	5
98	66	NA	4.6	87	8	6
99	122	255	4.0	89	8	7
100	89	229	10.3	90	8	8
101	110	207	8.0	90	8	9
104	44	192	11.5	86	8	12
105	28	273	11.5	82	8	13
106	65	157	9.7	80	8	14
108	22	71	10.3	77	8	16
109	59	51	6.3	79	8	17
110	23	115	7.4	76	8	18
111	31	244	10.9	78	8	19
112	44	190	10.3	78	8	20
113	21	259	15.5	77	8	21
114	9	36	14.3	72	8	22

```
116    45     212  9.7   79     8  24
117   168     238  3.4   81     8  25
118    73     215  8.0   86     8  26
120    76     203  9.7   97     8  28
121   118     225  2.3   94     8  29
122    84     237  6.3   96     8  30
123    85     188  6.3   94     8  31
124    96     167  6.9   91     9   1
125    78     197  5.1   92     9   2
126    73     183  2.8   93     9   3
127    91     189  4.6   93     9   4
128    47      95  7.4   87     9   5
129    32      92 15.5   84     9   6
130    20     252 10.9   80     9   7
131    23     220 10.3   78     9   8
132    21     230 10.9   75     9   9
133    24     259  9.7   73     9  10
134    44     236 14.9   81     9  11
135    21     259 15.5   76     9  12
136    28     238  6.3   77     9  13
137     9      24 10.9   71     9  14
138    13     112 11.5   71     9  15
139    46     237  6.9   78     9  16
140    18     224 13.8   67     9  17
141    13      27 10.3   76     9  18
142    24     238 10.3   68     9  19
143    16     201  8.0   82     9  20
144    13     238 12.6   64     9  21
145    23      14  9.2   71     9  22
146    36     139 10.3   81     9  23
147     7      49 10.3   69     9  24
148    14      20 16.6   63     9  25
149    30     193  6.9   70     9  26
151    14     191 14.3   75     9  28
152    18     131  8.0   76     9  29
153    20     223 11.5   68     9  30
```

33)Subset “airquality” for “Ozone” greater than “100“. Select the columns “Ozone“,“Temp“, “Month” and “Day” only.

```
> subset(airquality, Ozone > 100, select = c(Ozone, Temp, Month, Day))
   Ozone Temp Month Day
30    115   79     5  30
62    135   84     7   1
```

```
86    108   85     7  25
99    122   89     8   7
101   110   90     8   9
117   168   81     8  25
121   118   94     8  29
```

34)Subset “LifeCycleSavings” for “sr” greater than “8“, and less than “10“. Remove columns “pop75” through “dpi“.

```
> subset(LifeCycleSavings, sr > 8 & sr < 10, select = -pop75:-dpi)
           sr pop15  ddpi
Canada   8.79 31.72  2.43
India    9.00 41.31  1.54
Venezuela 9.22 46.40  0.53
Uruguay  9.24 28.13  1.88
Libya    8.89 43.69 16.71
```