NAME – ANSH GOEL

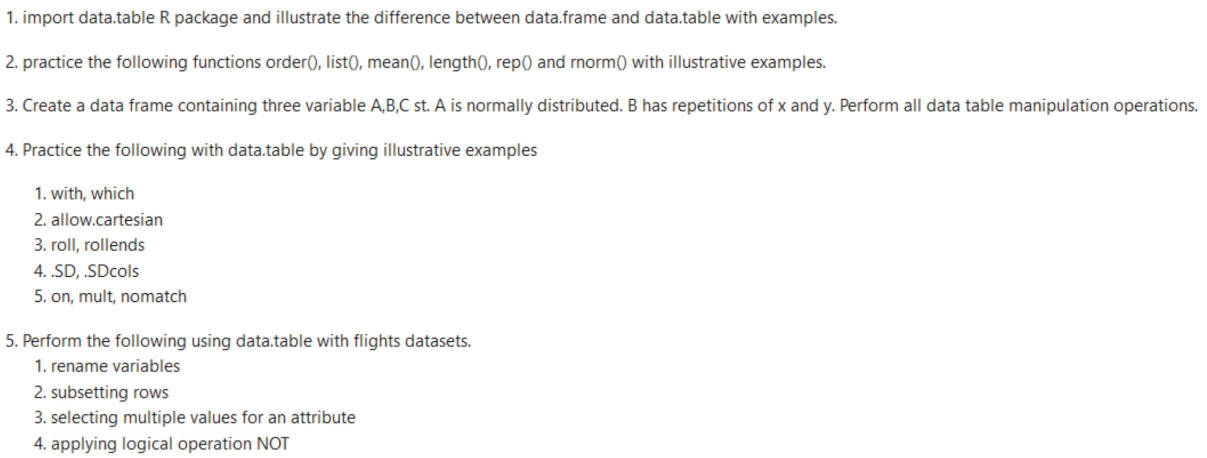
REG NO. – 20BCE1798

COURSE NAME – FOUNDATION OF DATA ANALYTICS (FDA)

COURSE CODE: 3505

DATE – 21st JULY, 2022

LAB 4



1. import data.table R package and illustrate the difference between data.frame and data.table with examples.

2. practice the following functions order(), list(), mean(), length(), rep() and rnorm() with illustrative examples.

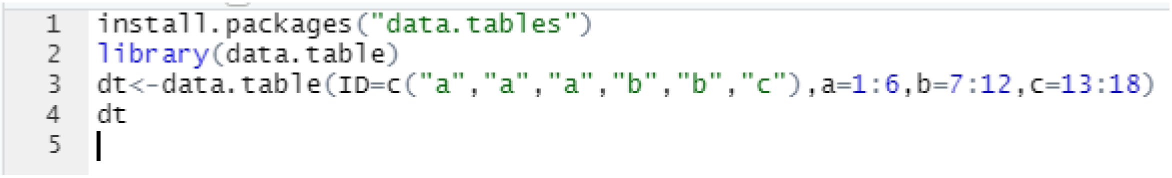
3. Create a data frame containing three variable A,B,C st. A is normally distributed. B has repetitions of x and y. Perform all data table manipulation operations.

4. Practice the following with data.table by giving illustrative examples

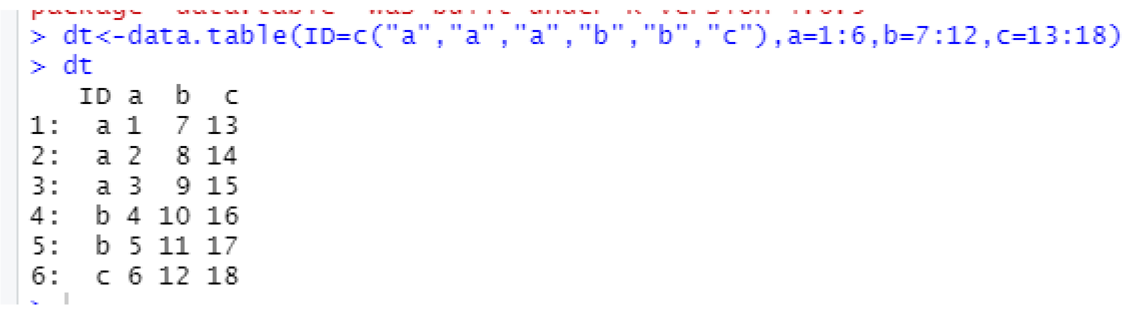
   1. with, which
   2. allow.cartesian
   3. roll, rollends
   4. .SD, .SDcols
   5. on, mult, nomatch

5. Perform the following using data.table with flights datasets.
   1. rename variables
   2. subsetting rows
   3. selecting multiple values for an attribute
   4. applying logical operation NOT

Q1)



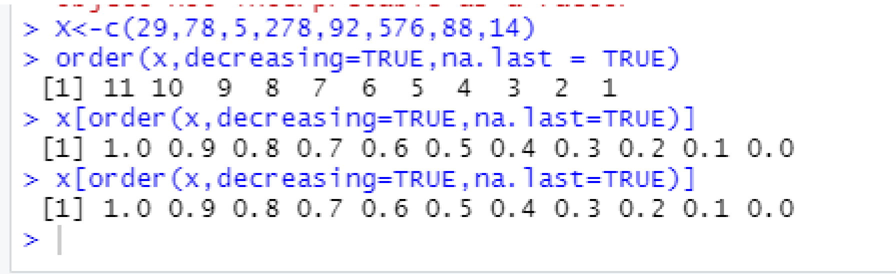
```
install.packages("data.tables")
library(data.table)
dt<-data.table(ID=c("a","a","a","b","b","c"),a=1:6,b=7:12,c=13:18)
dt

```



```
package 'data.table' was built under R version [illegible]
> dt<-data.table(ID=c("a","a","a","b","b","c"),a=1:6,b=7:12,c=13:18)
> dt
   ID a  b  c
1:  a 1  7 13
2:  a 2  8 14
3:  a 3  9 15
4:  b 4 10 16
5:  b 5 11 17
6:  c 6 12 18
```

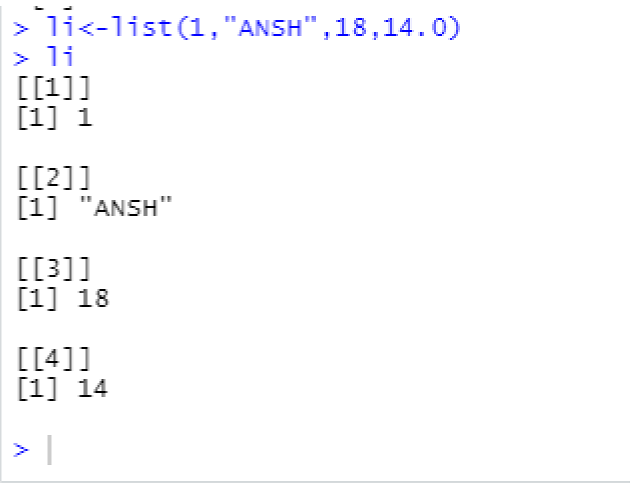
Q2)

a) Order()- returns the index which will sort the array in the mentioned order



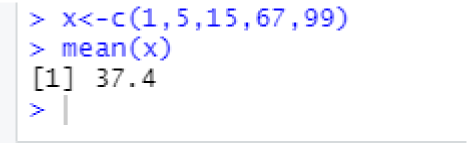
```
> x<-c(29,78,5,278,92,576,88,14)
> order(x,decreasing=TRUE,na.last = TRUE)
 [1] 11 10  9  8  7  6  5  4  3  2  1
> x[order(x,decreasing=TRUE,na.last=TRUE)]
 [1] 1.0 0.9 0.8 0.7 0.6 0.5 0.4 0.3 0.2 0.1 0.0
> x[order(x,decreasing=TRUE,na.last=TRUE)]
 [1] 1.0 0.9 0.8 0.7 0.6 0.5 0.4 0.3 0.2 0.1 0.0
> 
```

b) List()- to create a collection of different data types



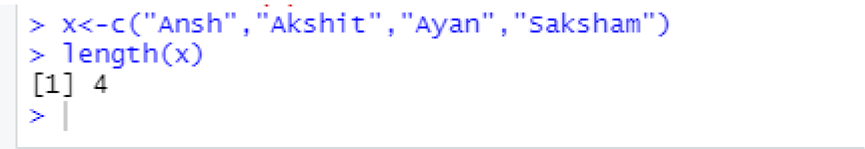
```
> li<-list(1,"ANSH",18,14.0)
> li
[[1]]
[1] 1

[[2]]
[1] "ANSH"

[[3]]
[1] 18

[[4]]
[1] 14

> 
```

c) Mean ()- to find the mean of the elements in the specified vector



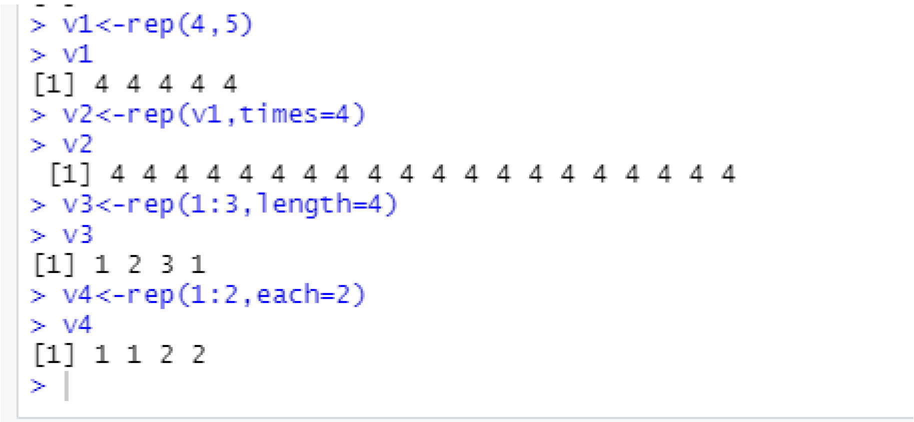
```
> x<-c(1,5,15,67,99)
> mean(x)
[1] 37.4
> 
```

d) Length()- to find the length- the number of elements in the specified vector



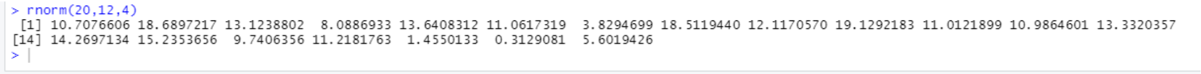
```
> x<-c("Ansh","Akshit","Ayan","Saksham")
> length(x)
[1] 4
> 
```

e) Rep()- to replicate elements of a vector



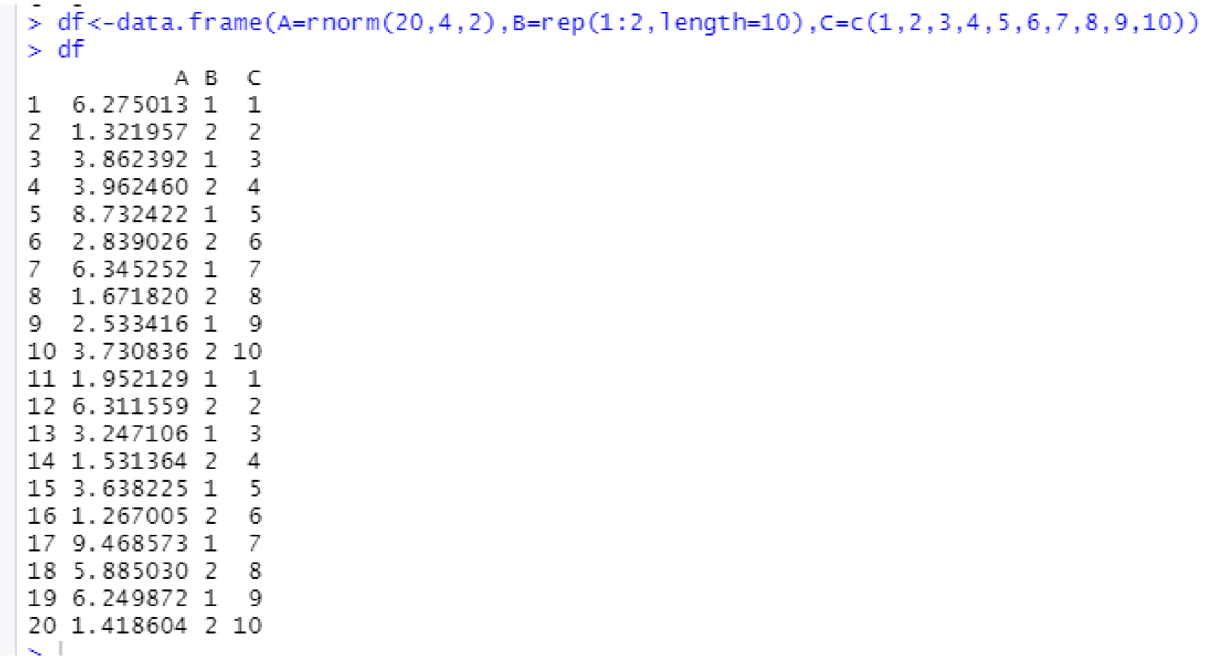
```
> v1<-rep(4,5)
> v1
[1] 4 4 4 4 4
> v2<-rep(v1,times=4)
> v2
 [1] 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4
> v3<-rep(1:3,length=4)
> v3
[1] 1 2 3 1
> v4<-rep(1:2,each=2)
> v4
[1] 1 1 2 2
> |
```

f) Rnorm()- to create the normal distribution of random variates. It takes the number of variables, required mean and standard deviation as parameters



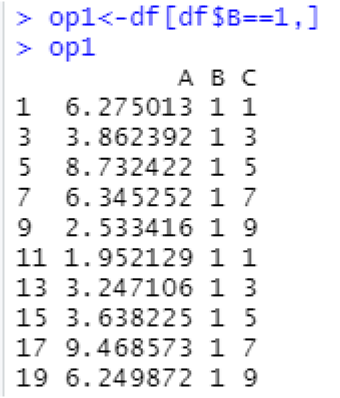
```
> rnorm(20,12,4)
 [1] 10.7076606 18.6897217 13.1238802  8.0886933 13.6408312 11.0617319  3.8294699 18.5119440 12.1170570 19.1292183 11.0121899 10.9864601 13.3320357
[14] 14.2697134 15.2353656  9.7406356 11.2181763  1.4550133  0.3129081  5.6019426
> |
```

Q3)



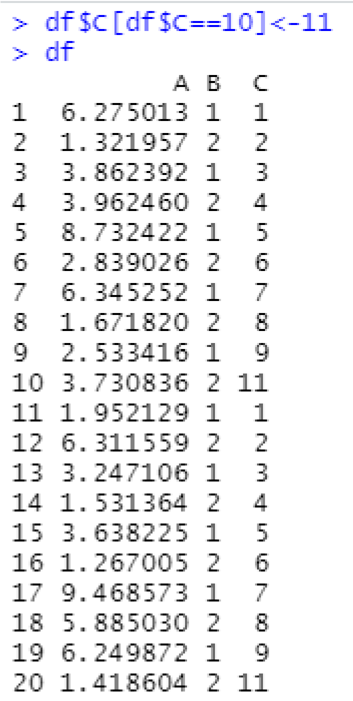
```
> df<-data.frame(A=rnorm(20,4,2),B=rep(1:2,length=10),C=c(1,2,3,4,5,6,7,8,9,10))
> df
          A B  C
1  6.275013 1  1
2  1.321957 2  2
3  3.862392 1  3
4  3.962460 2  4
5  8.732422 1  5
6  2.839026 2  6
7  6.345252 1  7
8  1.671820 2  8
9  2.533416 1  9
10 3.730836 2 10
11 1.952129 1  1
12 6.311559 2  2
13 3.247106 1  3
14 1.531364 2  4
15 3.638225 1  5
16 1.267005 2  6
17 9.468573 1  7
18 5.885030 2  8
19 6.249872 1  9
20 1.418604 2 10
> |
```

i)Sub setting



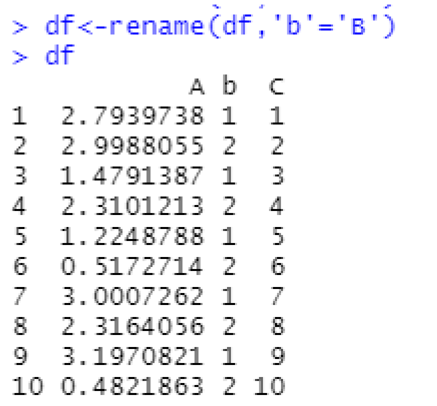
```
> op1<-df[df$B==1,]
> op1
          A B C
1  6.275013 1 1
3  3.862392 1 3
5  8.732422 1 5
7  6.345252 1 7
9  2.533416 1 9
11 1.952129 1 1
13 3.247106 1 3
15 3.638225 1 5
17 9.468573 1 7
19 6.249872 1 9
```

ii)Replacing a value



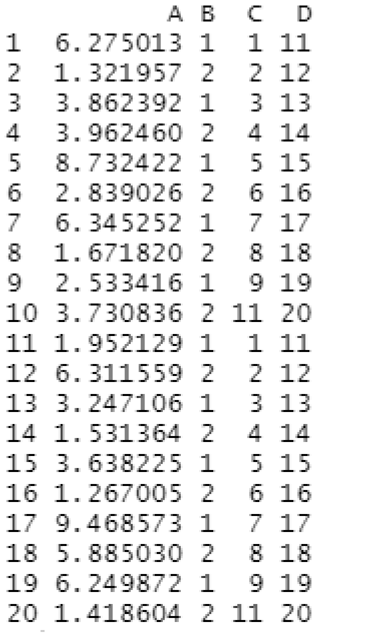
```
> df$C[df$C==10]<-11
> df
          A B  C
1  6.275013 1  1
2  1.321957 2  2
3  3.862392 1  3
4  3.962460 2  4
5  8.732422 1  5
6  2.839026 2  6
7  6.345252 1  7
8  1.671820 2  8
9  2.533416 1  9
10 3.730836 2 11
11 1.952129 1  1
12 6.311559 2  2
13 3.247106 1  3
14 1.531364 2  4
15 3.638225 1  5
16 1.267005 2  6
17 9.468573 1  7
18 5.885030 2  8
19 6.249872 1  9
20 1.418604 2 11
```

iii)Renaming a variable



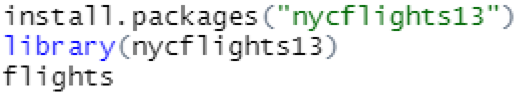
```
> df<-rename(df,'b'='B')
> df
           A b  C
1  2.7939738 1  1
2  2.9988055 2  2
3  1.4791387 1  3
4  2.3101213 2  4
5  1.2248788 1  5
6  0.5172714 2  6
7  3.0007262 1  7
8  2.3164056 2  8
9  3.1970821 1  9
10 0.4821863 2 10
```

iv)Adding a column

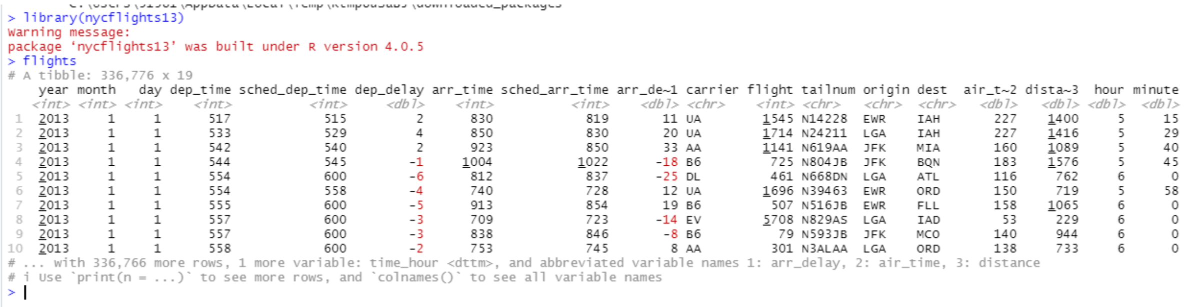


```
          A B  C  D
1  6.275013 1  1 11
2  1.321957 2  2 12
3  3.862392 1  3 13
4  3.962460 2  4 14
5  8.732422 1  5 15
6  2.839026 2  6 16
7  6.345252 1  7 17
8  1.671820 2  8 18
9  2.533416 1  9 19
10 3.730836 2 11 20
11 1.952129 1  1 11
12 6.311559 2  2 12
13 3.247106 1  3 13
14 1.531364 2  4 14
15 3.638225 1  5 15
16 1.267005 2  6 16
17 9.468573 1  7 17
18 5.885030 2  8 18
19 6.249872 1  9 19
20 1.418604 2 11 20
```

Q5)



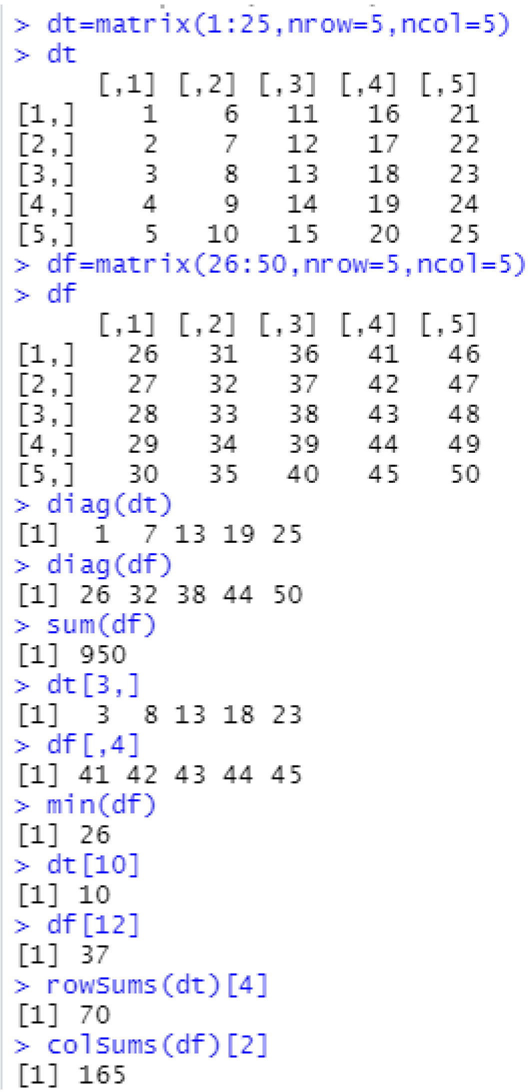
```
install.packages("nycflights13")
library(nycflights13)
flights
```



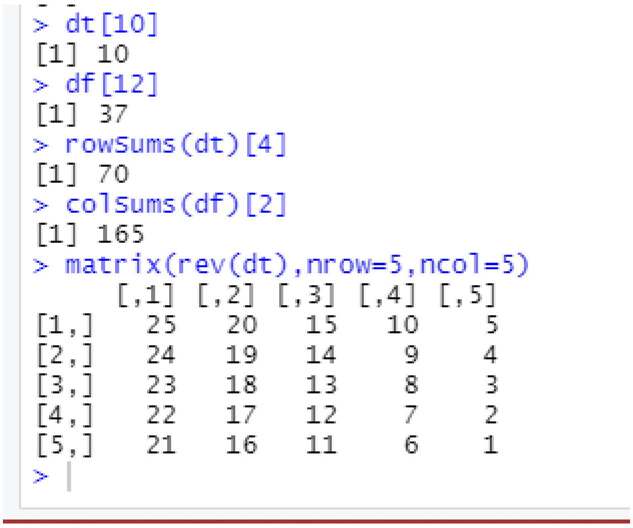
```
> library(nycflights13)
Warning message:
package ‘nycflights13’ was built under R version 4.0.5
> flights
# A tibble: 336,776 x 19
    year month   day dep_time sched_dep_time dep_delay arr_time sched_arr_time arr_de~1 carrier flight tailnum origin dest  air_t~2 dista~3  hour minute
   <int> <int> <int>    <int>          <int>     <dbl>    <int>          <int>    <dbl> <chr>    <int> <chr>   <chr>  <chr>   <dbl>   <dbl> <dbl>  <dbl>
 1  2013     1     1      517            515         2      830            819       11 UA        1545 N14228  EWR    IAH       227    1400     5     15
 2  2013     1     1      533            529         4      850            830       20 UA        1714 N24211  LGA    IAH       227    1416     5     29
 3  2013     1     1      542            540         2      923            850       33 AA        1141 N619AA  JFK    MIA       160    1089     5     40
 4  2013     1     1      544            545        -1     1004           1022      -18 B6         725 N804JB  JFK    BQN       183    1576     5     45
 5  2013     1     1      554            600        -6      812            837      -25 DL         461 N668DN  LGA    ATL       116     762     6      0
 6  2013     1     1      554            558        -4      740            728       12 UA        1696 N39463  EWR    ORD       150     719     5     58
 7  2013     1     1      555            600        -5      913            854       19 B6         507 N516JB  EWR    FLL       158    1065     6      0
 8  2013     1     1      557            600        -3      709            723      -14 EV        5708 N829AS  LGA    IAD        53     229     6      0
 9  2013     1     1      557            600        -3      838            846       -8 B6          79 N593JB  JFK    MCO       140     944     6      0
10  2013     1     1      558            600        -2      753            745        8 AA         301 N3ALAA  LGA    ORD       138     733     6      0
# … with 336,766 more rows, 1 more variable: time_hour <dttm>, and abbreviated variable names 1: arr_delay, 2: air_time, 3: distance
# i Use `print(n = ...)` to see more rows, and `colnames()` to see all variable names
> 
```

Q6) . Create a 2 matrix of dimension 5x5

```
dt=matrix(1:25,nrow=5,ncol=5)
dt
df=matrix(26:50,nrow=5,ncol=5)
df

#(i) Find the diagonal element of matrix 1 and matrix 2.
diag(dt)
diag(df)

#(ii) Find the sum of all the values in matrix 2
sum(df)

#(iii) Display 3rd row in matrix 1 and 4th column in matrix 2
dt[3,]
df[,4]

#(iv) Find the smallest element in matrix 2
min(df)

#(v) Display the 10th element in matrix 1 and 12th element in matrix 2
dt[10]
df[12]

#(vi) Find sum of all values of 4th row in matrix 1 and 2nd column in matrix 2
rowSums(dt)[4]
colSums(df)[2]

#(vii) Display the reverse of all the elements in matrix 1
matrix(rev(dt),nrow=5,ncol=5)
```



```
> dt=matrix(1:25,nrow=5,ncol=5)
> dt
     [,1] [,2] [,3] [,4] [,5]
[1,]    1    6   11   16   21
[2,]    2    7   12   17   22
[3,]    3    8   13   18   23
[4,]    4    9   14   19   24
[5,]    5   10   15   20   25
> df=matrix(26:50,nrow=5,ncol=5)
> df
     [,1] [,2] [,3] [,4] [,5]
[1,]   26   31   36   41   46
[2,]   27   32   37   42   47
[3,]   28   33   38   43   48
[4,]   29   34   39   44   49
[5,]   30   35   40   45   50
> diag(dt)
[1]  1  7 13 19 25
> diag(df)
[1] 26 32 38 44 50
> sum(df)
[1] 950
> dt[3,]
[1]  3  8 13 18 23
> df[,4]
[1] 41 42 43 44 45
> min(df)
[1] 26
> dt[10]
[1] 10
> df[12]
[1] 37
> rowSums(dt)[4]
[1] 70
> colSums(df)[2]
[1] 165
```



```
> dt[10]
[1] 10
> df[12]
[1] 37
> rowSums(dt)[4]
[1] 70
> colSums(df)[2]
[1] 165
```

```
> matrix(rev(dt),nrow=5,ncol=5)
     [,1] [,2] [,3] [,4] [,5]
[1,]   25   20   15   10    5
[2,]   24   19   14    9    4
[3,]   23   18   13    8    3
[4,]   22   17   12    7    2
[5,]   21   16   11    6    1
> 
```

Q4)

```
> DT = data.table(x=rep(c("b","a","c"),each=3), y=c(1,3,6), v=1:9)
> head(DT)
   x y v
1: b 1 1
2: b 3 2
3: b 6 3
4: a 1 4
5: a 3 5
6: a 6 6
> X = data.table(x=c("c","b"), v=8:7, foo=c(4,2))
> head(X)
   x v foo
1: c 8   4
2: b 7   2
> data <- data.frame(x1 = c(5, 3, 1),x2 = c(4, 3, 1))
> data
  x1 x2
1  5  4
2  3  3
3  1  1
> with(data,x1+x2)
[1] 9 6 2
> which(mtcars$disp == 160)
[1] 1 2
> DT[.("a", 1:5), on=c("x", "y"), roll=-Inf]
   x y v
1: a 1 4
2: a 2 5
3: a 3 5
4: a 4 6
5: a 5 6
> DT[, .SD[1]]
   x y v
1: b 1 1
> DT[, .SD, .SDcols=x:y]
   x y
1: b 1
2: b 3
3: b 6
4: a 1
5: a 3
6: a 6
7: c 1
8: c 3
9: c 6
> DT[X, on="x"]
   x y v i.v foo
1: c 1 7   8   4
```

```
2: c 3 8   8   4
3: c 6 9   8   4
4: b 1 1   7   2
5: b 3 2   7   2
6: b 6 3   7   2
> DT[X, on="x", mult="last"]
   x y v i.v foo
1: c 6 9   8   4
2: b 6 3   7   2
> DT[X, on="x", nomatch=NULL]
   x y v i.v foo
1: c 1 7   8   4
2: c 3 8   8   4
3: c 6 9   8   4
4: b 1 1   7   2
5: b 3 2   7   2
6: b 6 3   7   2
```