

## Day 1 – R Programming

```
> #install.packages('caret')
> num = 10
> num
[1] 10
> library('caret')
> x = 10.2
> y <- 10
> z = "Hello"
> x
[1] 10.2
> y
[1] 10
> z
[1] "Hello"
> as.integer(x)
[1] 10
> a = 1 + 10i
> a
[1] 1+10i
> sqrt(144)
[1] 12
> a = 5; b = 15
> out = a > b
> out
[1] FALSE
> age <- c(21, 25, 28, 30, 20, 26)
> age
[1] 21 25 28 30 20 26
> id = c(1:10) #range values from 1-10
> id
[1] 1 2 3 4 5 6 7 8 9 10
> seq(1, 20)
[1] 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
> seq(2, 20, 2) #range values from 2 to 20 with offset 2
[1] 2 4 6 8 10 12 14 16 18 20
> loan_default <- c(TRUE, FALSE, FALSE, TRUE, TRUE)
> loan_default
[1] TRUE FALSE FALSE TRUE TRUE
> place_names <- c("China", "India", "Denmark", "UK", "Finland")
> place_names
[1] "China" "India" "Denmark" "UK" "Finland"
> class(loan_default)
[1] "logical"
```

```

> class(age)
[1] "numeric"
> class(z)
[1] "character"
> num_as_str <- c("10", "30", "40", "50")
> class(num_as_str)
[1] "character"
> numbers <- as.integer(num_as_str)
> class(numbers)
[1] "integer"
> mean(numbers)
[1] 32.5
> max(age)
[1] 30
> min(numbers)
[1] 10
> median(age)
[1] 25.5
> range(numbers)
[1] 10 50
> var(age)
[1] 15.2
> sort(age)
[1] 20 21 25 26 28 30
> sort(age, decreasing = TRUE)
[1] 30 28 26 25 21 20
> random_ele <- c(15, 2.5, TRUE, "Hello")
> random_ele
[1] "15"  "2.5" "TRUE" "Hello"
> class(random_ele)
[1] "character"
> mat <- c(1:16)
> mat <- matrix(mat, ncol=4)
> mat
      [,1] [,2] [,3] [,4]
[1,]  1   5   9  13
[2,]  2   6  10  14
[3,]  3   7  11  15
[4,]  4   8  12  16
> mat1 <- c(1:16)
> mat1 <- matrix(mat1, ncol = 4, byrow = T)
> mat1
      [,1] [,2] [,3] [,4]
[1,]  1   2   3   4

```

```

[2,] 5 6 7 8
[3,] 9 10 11 12
[4,] 13 14 15 16
> matrix(c(56, 72, 25, 14, 87, 99), ncol = 3, byrow = T)
      [,1] [,2] [,3]
[1,] 56 72 25
[2,] 14 87 99
> mat1[2,]
[1] 5 6 7 8
> mat1[2,2]
[1] 6
> mat1[,4]
[1] 4 8 12 16
> matr = matrix(c(5:16), nrow = 3, byrow = TRUE)
> column.names <- c("COL1", "COL2", "COL3")
> row.names <- c("ROW1", "ROW2", "ROW3")
> column.names <- c("COL1", "COL2", "COL3", "COL4")
> result <- matrix(c(5:16), nrow = 3, byrow = TRUE, dimnames = list(row.names, column.names))
> result
      COL1 COL2 COL3 COL4
ROW1  5  6  7  8
ROW2  9 10 11 12
ROW3 13 14 15 16
> employee = list(1, c("John", "Rose"), c(12000, 15000))
> employee
[[1]]
[1] 1

[[2]]
[1] "John" "Rose"

[[3]]
[1] 12000 15000

> employee[[1]]
[1] 1
> employee[[2]]
[1] "John" "Rose"
> employee[[3]]
[1] 12000 15000
> employee = list(EmpID=1, EmpName=c("John", "Rose"), basic_pay=c(12000, 15000))
> employee
$EmpID
[1] 1

```

```
$EmpName  
[1] "John" "Rose"
```

```
$basic_pay  
[1] 12000 15000
```

```
> employee$EmpName  
[1] "John" "Rose"  
> list_of_expenses <- list(100, 150, 350, 50)  
> class(list_of_expenses)  
[1] "list"  
> expenses <- unlist(list_of_expenses)  
> class(expenses)  
[1] "numeric"  
> length(expenses)  
[1] 4  
> days_from_purchase <- c(10, 15, 20, 25)  
> days_from_purchase  
[1] 10 15 20 25  
> ctf <- as.factor(days_from_purchase)  
> typeof(ctf)  
[1] "integer"  
> class(ctf)  
[1] "factor"  
> age <- c(21, 42, 28, 31, 19)  
> names <- c("John", "Sachin", "Rahul", "Ravi", "Sameer")  
> salary <- c(12000, 20000, 25000, 16000, 28000)  
> ownhouse <- c(TRUE, FALSE, TRUE, TRUE, FALSE)  
> mydf <- data.frame(names, age, salary, ownhouse)  
> mydf  
  names age salary ownhouse  
1 John  21 12000    TRUE  
2 Sachin 42 20000   FALSE  
3 Rahul  28 25000    TRUE  
4 Ravi   31 16000    TRUE  
5 Sameer 19 28000   FALSE  
> stock_price <- c(110.55, 102.50, 145.90, 130.70, 160.45, 112.80)  
> stock_mat <- matrix(stock_price, ncol = 2, byrow = T)  
> stock_df = data.frame(stock_mat)  
> stock_df  
  X1  X2  
1 110.55 102.5  
2 145.90 130.7
```

```

3 160.45 112.8
> colnames(stock_df) <- c("Open Price", "Close Price")
> letters[1:10]
[1] "a" "b" "c" "d" "e" "f" "g" "h" "i" "j"
> letters[1:26]
[1] "a" "b" "c" "d" "e" "f" "g" "h" "i" "j" "k" "l" "m" "n" "o" "p" "q" "r" "s" "t" "u" "v" "w" "x" "y" "z"
> rownames(stock_df) <- letters[1:3]
> stock_df
  Open Price Close Price
a    110.55    102.5
b    145.90    130.7
c    160.45    112.8
> stock_df$`Close Price`
[1] 102.5 130.7 112.8

```

## Day 2 – R Programming

```

> X <- matrix(c(50, 70, 40, 90, 60, 80, 50, 90, 100, 50, 30, 70), nrow = 3)
> X
      [,1] [,2] [,3] [,4]
[1,]  50  90  50  50
[2,]  70  60  90  30
[3,]  40  80 100  70
> rowSums(X)
[1] 240 250 290
> colSums(X)
[1] 160 230 240 150
> X <- rbind(X, apply(X, 2, mean)) #Add a row and apply mean function columnwise - 2, for rowwise its
1
> X
      [,1] [,2] [,3] [,4]
[1,] 50.00000 90.00000  50  50
[2,] 70.00000 60.00000  90  30
[3,] 40.00000 80.00000 100  70
[4,] 53.33333 76.66667  80  50
> X <- cbind(X, apply(X, 1, var)) #Add a column and apply variance function rowwise - 1
> X
      [,1] [,2] [,3] [,4] [,5]
[1,] 50.00000 90.00000  50  50 400.0000
[2,] 70.00000 60.00000  90  30 625.0000
[3,] 40.00000 80.00000 100  70 625.0000
[4,] 53.33333 76.66667  80  50 240.7407
> X <- matrix(c(50, 70, 40, 90, 60, 80, 50, 90, 100, 50, 30, 70), nrow = 3)
> X <- cbind(X, apply(X, 1, sd)) #Add a column and apply standard deviation function rowwise - 1
> X

```

```

      [,1] [,2] [,3] [,4] [,5]
[1,] 50 90 50 50 20
[2,] 70 60 90 30 25
[3,] 40 80 100 70 25
> X <- rbind(X, apply(X, 2, max)) #Add a row and apply maximum function columnwise - 2, for rowwise
its 1
> X
      [,1] [,2] [,3] [,4] [,5]
[1,] 50 90 50 50 20
[2,] 70 60 90 30 25
[3,] 40 80 100 70 25
[4,] 70 90 100 70 25
> stock_df[[1]] #1st column
[1] 110.55 145.90 160.45
> stock_df[[2]] #2nd column
[1] 102.5 130.7 112.8
> stock_df
  Open Price Close Price BuyOrSell
a   110.55    102.5    Sell
b   145.90    130.7    Sell
c   160.45    112.8    Sell
> stock_df[1:2, 2]
[1] 102.5 130.7
> stock_df[1:3, 1:2]
  Open Price Close Price
a   110.55    102.5
b   145.90    130.7
c   160.45    112.8
> stock_df[, 1:2]
  Open Price Close Price
a   110.55    102.5
b   145.90    130.7
c   160.45    112.8
> stock_df[c(1, 3), 1:2]
  Open Price Close Price
a   110.55    102.5
c   160.45    112.8
> stock_df[-1, 1]
[1] 145.90 160.45
> stock_df[-c(1, 3), 1:2]
  Open Price Close Price
b   145.9    130.7
> v_sub <- stock_df[1:3, 2]
> v_sub

```

```

[1] 102.5 130.7 112.8
> df_subsetdata <- stock_df[1:3, 2, drop=F]
> df_subsetdata
  Close Price
a    102.5
b    130.7
c    112.8
> class(v_sub)
[1] "numeric"
> class(df_subsetdata)
[1] "data.frame"
> setwd("C:/zubeda/PGA02_Zubu/R Programming") #Set current working directory
> housing_df <- read.csv("Housing.csv")
> housing_df
  price area bedrooms bathrooms stories mainroad guestroom basement
1 13300000 7420    4      2      3   yes    no    no
2 12250000 8960    4      4      4   yes    no    no
3 12250000 9960    3      2      2   yes    no   yes
4 12215000 7500    4      2      2   yes    no   yes
5 11410000 7420    4      1      2   yes   yes   yes
6 10850000 7500    3      3      1   yes    no   yes
7 10150000 8580    4      3      4   yes    no    no
8 10150000 16200   5      3      2   yes    no    no
9 9870000 8100    4      1      2   yes   yes   yes
10 9800000 5750    3      2      4   yes   yes    no
11 9800000 13200   3      1      2   yes    no   yes
12 9681000 6000    4      3      2   yes   yes   yes
13 9310000 6550    4      2      2   yes    no    no
14 9240000 3500    4      2      2   yes    no    no
15 9240000 7800    3      2      2   yes    no    no
16 9100000 6000    4      1      2   yes    no   yes
17 9100000 6600    4      2      2   yes   yes   yes
18 8960000 8500    3      2      4   yes    no    no
19 8890000 4600    3      2      2   yes   yes    no
20 8855000 6420    3      2      2   yes    no    no
21 8750000 4320    3      1      2   yes    no   yes
22 8680000 7155    3      2      1   yes   yes   yes
23 8645000 8050    3      1      1   yes   yes   yes
24 8645000 4560    3      2      2   yes   yes   yes
25 8575000 8800    3      2      2   yes    no    no
26 8540000 6540    4      2      2   yes   yes   yes
27 8463000 6000    3      2      4   yes   yes   yes
28 8400000 8875    3      1      1   yes    no    no
29 8400000 7950    5      2      2   yes    no   yes

```

30	8400000	5500	4	2	2	yes	no	yes
31	8400000	7475	3	2	4	yes	no	no
32	8400000	7000	3	1	4	yes	no	no
33	8295000	4880	4	2	2	yes	no	no
34	8190000	5960	3	3	2	yes	yes	yes
35	8120000	6840	5	1	2	yes	yes	yes
36	8080940	7000	3	2	4	yes	no	no
37	8043000	7482	3	2	3	yes	no	no
38	7980000	9000	4	2	4	yes	no	no
39	7962500	6000	3	1	4	yes	yes	no
40	7910000	6000	4	2	4	yes	no	no
41	7875000	6550	3	1	2	yes	no	yes
42	7840000	6360	3	2	4	yes	no	no
43	7700000	6480	3	2	4	yes	no	no
44	7700000	6000	4	2	4	yes	no	no
45	7560000	6000	4	2	4	yes	no	no
46	7560000	6000	3	2	3	yes	no	no
47	7525000	6000	3	2	4	yes	no	no
48	7490000	6600	3	1	4	yes	no	no
49	7455000	4300	3	2	2	yes	no	yes
50	7420000	7440	3	2	1	yes	yes	yes
51	7420000	7440	3	2	4	yes	no	no
52	7420000	6325	3	1	4	yes	no	no
53	7350000	6000	4	2	4	yes	yes	no
54	7350000	5150	3	2	4	yes	no	no
55	7350000	6000	3	2	2	yes	yes	no
56	7350000	6000	3	1	2	yes	no	no
57	7343000	11440	4	1	2	yes	no	yes
58	7245000	9000	4	2	4	yes	yes	no
59	7210000	7680	4	2	4	yes	yes	no
60	7210000	6000	3	2	4	yes	yes	no
61	7140000	6000	3	2	2	yes	yes	no
62	7070000	8880	2	1	1	yes	no	no
63	7070000	6240	4	2	2	yes	no	no
64	7035000	6360	4	2	3	yes	no	no
65	7000000	11175	3	1	1	yes	no	yes
66	6930000	8880	3	2	2	yes	no	yes
67	6930000	13200	2	1	1	yes	no	yes
68	6895000	7700	3	2	1	yes	no	no
69	6860000	6000	3	1	1	yes	no	no
70	6790000	12090	4	2	2	yes	no	no
71	6790000	4000	3	2	2	yes	no	yes
72	6755000	6000	4	2	4	yes	no	no
73	6720000	5020	3	1	4	yes	no	no



74	6685000	6600	2	2	4	yes	no	yes
75	6650000	4040	3	1	2	yes	no	yes
76	6650000	4260	4	2	2	yes	no	no

hotwaterheating airconditioning parking prefarea furnishingstatus

1	no	yes	2	yes	furnished
2	no	yes	3	no	furnished
3	no	no	2	yes	semi-furnished
4	no	yes	3	yes	furnished
5	no	yes	2	no	furnished
6	no	yes	2	yes	semi-furnished
7	no	yes	2	yes	semi-furnished
8	no	no	0	no	unfurnished
9	no	yes	2	yes	furnished
10	no	yes	1	yes	unfurnished
11	no	yes	2	yes	furnished
12	yes	no	2	no	semi-furnished
13	no	yes	1	yes	semi-furnished
14	yes	no	2	no	furnished
15	no	no	0	yes	semi-furnished
16	no	no	2	no	semi-furnished
17	no	yes	1	yes	unfurnished
18	no	yes	2	no	furnished
19	no	yes	2	no	furnished
20	no	yes	1	yes	semi-furnished
21	yes	no	2	no	semi-furnished
22	no	yes	2	no	unfurnished
23	no	yes	1	no	furnished
24	no	yes	1	no	furnished
25	no	yes	2	no	furnished
26	no	yes	2	yes	furnished
27	no	yes	0	yes	semi-furnished
28	no	no	1	no	semi-furnished
29	yes	no	2	no	unfurnished
30	no	yes	1	yes	semi-furnished
31	no	yes	2	no	unfurnished
32	no	yes	2	no	semi-furnished
33	no	yes	1	yes	furnished
34	no	no	1	no	unfurnished
35	no	yes	1	no	furnished
36	no	yes	2	no	furnished
37	yes	no	1	yes	furnished
38	no	yes	2	no	furnished
39	no	yes	2	no	unfurnished
40	no	yes	1	no	semi-furnished

41	no	yes	0	yes	furnished
42	no	yes	0	yes	furnished
43	no	yes	2	no	unfurnished
44	no	no	2	no	semi-furnished
45	no	yes	1	no	furnished
46	no	yes	0	no	semi-furnished
47	no	yes	1	no	furnished
48	no	yes	3	yes	furnished
49	no	no	1	no	unfurnished
50	no	yes	0	yes	semi-furnished
51	no	no	1	yes	unfurnished
52	no	yes	1	no	unfurnished
53	no	yes	1	no	furnished
54	no	yes	2	no	semi-furnished
55	no	yes	1	no	semi-furnished
56	no	yes	1	no	unfurnished
57	no	no	1	yes	semi-furnished
58	no	yes	1	yes	furnished
59	no	yes	1	no	semi-furnished
60	no	yes	1	no	furnished
61	no	no	1	no	semi-furnished
62	no	yes	1	no	semi-furnished
63	no	yes	1	no	furnished
64	no	yes	2	yes	furnished
65	no	yes	1	yes	furnished
66	no	yes	1	no	furnished
67	yes	no	1	no	furnished
68	no	no	2	no	unfurnished
69	no	yes	1	no	furnished
70	no	no	2	yes	furnished
71	no	yes	0	yes	semi-furnished
72	no	yes	0	no	unfurnished
73	no	yes	0	yes	unfurnished
74	no	no	0	yes	furnished
75	yes	no	1	no	furnished
76	yes	no	0	no	semi-furnished

[ reached 'max' / getOption("max.print") -- omitted 469 rows ]

```
> dim(housing_df) #no. of rows, no. of columns
```

```
[1] 545 13
```

```
> filter_df <- housing_df[housing_df$price > 10000000, ]
```

```
> filter_df
```

	price	area	bedrooms	bathrooms	stories	mainroad	guestroom	basement
1	13300000	7420	4	2	3	yes	no	no
2	12250000	8960	4	4	4	yes	no	no

3	12250000	9960	3	2	2	yes	no	yes
4	12215000	7500	4	2	2	yes	no	yes
5	11410000	7420	4	1	2	yes	yes	yes
6	10850000	7500	3	3	1	yes	no	yes
7	10150000	8580	4	3	4	yes	no	no
8	10150000	16200	5	3	2	yes	no	no

hotwaterheating airconditioning parking prefarea furnishingstatus

1	no	yes	2	yes	furnished
2	no	yes	3	no	furnished
3	no	no	2	yes	semi-furnished
4	no	yes	3	yes	furnished
5	no	yes	2	no	furnished
6	no	yes	2	yes	semi-furnished
7	no	yes	2	yes	semi-furnished
8	no	no	0	no	unfurnished

```
> filt_df <- housing_df[housing_df$area > 6000, ]
```

```
> filt_df
```

	price	area	bedrooms	bathrooms	stories	mainroad	guestroom	basement
1	13300000	7420	4	2	3	yes	no	no
2	12250000	8960	4	4	4	yes	no	no
3	12250000	9960	3	2	2	yes	no	yes
4	12215000	7500	4	2	2	yes	no	yes
5	11410000	7420	4	1	2	yes	yes	yes
6	10850000	7500	3	3	1	yes	no	yes
7	10150000	8580	4	3	4	yes	no	no
8	10150000	16200	5	3	2	yes	no	no
9	9870000	8100	4	1	2	yes	yes	yes
11	9800000	13200	3	1	2	yes	no	yes
13	9310000	6550	4	2	2	yes	no	no
15	9240000	7800	3	2	2	yes	no	no
17	9100000	6600	4	2	2	yes	yes	yes
18	8960000	8500	3	2	4	yes	no	no
20	8855000	6420	3	2	2	yes	no	no
22	8680000	7155	3	2	1	yes	yes	yes
23	8645000	8050	3	1	1	yes	yes	yes
25	8575000	8800	3	2	2	yes	no	no
26	8540000	6540	4	2	2	yes	yes	yes
28	8400000	8875	3	1	1	yes	no	no
29	8400000	7950	5	2	2	yes	no	yes
31	8400000	7475	3	2	4	yes	no	no
32	8400000	7000	3	1	4	yes	no	no
35	8120000	6840	5	1	2	yes	yes	yes
36	8080940	7000	3	2	4	yes	no	no
37	8043000	7482	3	2	3	yes	no	no

38	7980000	9000	4	2	4	yes	no	no
41	7875000	6550	3	1	2	yes	no	yes
42	7840000	6360	3	2	4	yes	no	no
43	7700000	6480	3	2	4	yes	no	no
48	7490000	6600	3	1	4	yes	no	no
50	7420000	7440	3	2	1	yes	yes	yes
51	7420000	7440	3	2	4	yes	no	no
52	7420000	6325	3	1	4	yes	no	no
57	7343000	11440	4	1	2	yes	no	yes
58	7245000	9000	4	2	4	yes	yes	no
59	7210000	7680	4	2	4	yes	yes	no
62	7070000	8880	2	1	1	yes	no	no
63	7070000	6240	4	2	2	yes	no	no
64	7035000	6360	4	2	3	yes	no	no
65	7000000	11175	3	1	1	yes	no	yes
66	6930000	8880	3	2	2	yes	no	yes
67	6930000	13200	2	1	1	yes	no	yes
68	6895000	7700	3	2	1	yes	no	no
70	6790000	12090	4	2	2	yes	no	no
74	6685000	6600	2	2	4	yes	no	yes
77	6650000	6420	3	2	3	yes	no	no
78	6650000	6500	3	2	3	yes	no	no
83	6615000	10500	3	2	1	yes	no	yes
86	6510000	8250	3	2	3	yes	no	no
87	6510000	6670	3	1	3	yes	no	yes
89	6475000	7410	3	1	1	yes	yes	yes
90	6440000	8580	5	3	2	yes	no	no
92	6419000	6750	2	1	1	yes	yes	yes
94	6300000	7200	3	2	1	yes	no	yes
97	6300000	9000	3	1	1	yes	no	yes
98	6300000	6400	3	1	1	yes	yes	yes
99	6293000	6600	3	2	3	yes	no	no
101	6230000	6600	3	2	1	yes	no	yes
104	6195000	6350	3	2	3	yes	yes	no
108	6125000	6420	3	1	3	yes	no	yes
110	6090000	6615	4	2	2	yes	yes	no
111	6090000	6600	3	1	1	yes	yes	yes
112	6090000	8372	3	1	3	yes	no	no
114	6083000	9620	3	1	1	yes	no	yes
115	6020000	6800	2	1	1	yes	yes	yes
116	6020000	8000	3	1	1	yes	yes	yes
117	6020000	6900	3	2	1	yes	yes	yes
119	5950000	6420	3	1	1	yes	no	yes
120	5950000	7020	3	1	1	yes	no	yes

121	5950000	6540	3	1	1	yes	yes	yes
122	5950000	7231	3	1	2	yes	yes	yes
123	5950000	6254	4	2	1	yes	no	yes
124	5950000	7320	4	2	2	yes	no	no
125	5950000	6525	3	2	4	yes	no	no
126	5943000	15600	3	1	1	yes	no	no

hotwaterheating airconditioning parking prefarea furnishingstatus

1	no	yes	2	yes	furnished
2	no	yes	3	no	furnished
3	no	no	2	yes	semi-furnished
4	no	yes	3	yes	furnished
5	no	yes	2	no	furnished
6	no	yes	2	yes	semi-furnished
7	no	yes	2	yes	semi-furnished
8	no	no	0	no	unfurnished
9	no	yes	2	yes	furnished
11	no	yes	2	yes	furnished
13	no	yes	1	yes	semi-furnished
15	no	no	0	yes	semi-furnished
17	no	yes	1	yes	unfurnished
18	no	yes	2	no	furnished
20	no	yes	1	yes	semi-furnished
22	no	yes	2	no	unfurnished
23	no	yes	1	no	furnished
25	no	yes	2	no	furnished
26	no	yes	2	yes	furnished
28	no	no	1	no	semi-furnished
29	yes	no	2	no	unfurnished
31	no	yes	2	no	unfurnished
32	no	yes	2	no	semi-furnished
35	no	yes	1	no	furnished
36	no	yes	2	no	furnished
37	yes	no	1	yes	furnished
38	no	yes	2	no	furnished
41	no	yes	0	yes	furnished
42	no	yes	0	yes	furnished
43	no	yes	2	no	unfurnished
48	no	yes	3	yes	furnished
50	no	yes	0	yes	semi-furnished
51	no	no	1	yes	unfurnished
52	no	yes	1	no	unfurnished
57	no	no	1	yes	semi-furnished
58	no	yes	1	yes	furnished
59	no	yes	1	no	semi-furnished

62	no	yes	1	no	semi-furnished
63	no	yes	1	no	furnished
64	no	yes	2	yes	furnished
65	no	yes	1	yes	furnished
66	no	yes	1	no	furnished
67	yes	no	1	no	furnished
68	no	no	2	no	unfurnished
70	no	no	2	yes	furnished
74	no	no	0	yes	furnished
77	no	yes	0	yes	furnished
78	no	yes	0	yes	furnished
83	no	yes	1	yes	furnished
86	no	yes	0	no	furnished
87	no	no	0	yes	unfurnished
89	no	yes	2	yes	unfurnished
90	no	no	2	no	furnished
92	no	no	2	yes	furnished
94	no	yes	3	no	semi-furnished
97	no	no	1	yes	furnished
98	no	yes	1	yes	semi-furnished
99	no	yes	0	yes	unfurnished
101	no	yes	0	yes	unfurnished
104	no	yes	0	no	furnished
108	no	no	0	yes	unfurnished
110	yes	no	1	no	semi-furnished
111	no	no	2	yes	semi-furnished
112	no	yes	2	no	unfurnished
114	no	no	2	yes	furnished
115	no	no	2	no	furnished
116	no	yes	2	yes	semi-furnished
117	no	no	0	yes	unfurnished
119	no	yes	0	yes	furnished
120	no	yes	2	yes	semi-furnished
121	no	no	2	yes	furnished
122	no	yes	0	yes	semi-furnished
123	no	no	1	yes	semi-furnished
124	no	no	0	no	furnished
125	no	no	1	no	furnished
126	no	yes	2	no	semi-furnished

[ reached 'max' / getOption("max.print") -- omitted 81 rows ]

```
> price <- 5
> if(price > 5) {
+   print("Sell the stock")
+ } else {
```

```

+ print("Buy the stock")
+ }
[1] "Buy the stock"
> source("Conditional.R")
[1] "Buy the stock"
> stock_df
  Open Price Close Price BuyOrSell
a   110.55    102.5    Sell
b   145.90    130.7    Sell
c   160.45    112.8    Sell
> stock_df$BuyOrSell <- ifelse(stock_df$`Close Price` < 80, "Buy", "Sell")
> stock_df
  Open Price Close Price BuyOrSell
a   110.55    102.5    Sell
b   145.90    130.7    Sell
c   160.45    112.8    Sell
> for (x in 1:10) { print(x ^ 2) } #i raised to 2
[1] 1
[1] 4
[1] 9
[1] 16
[1] 25
[1] 36
[1] 49
[1] 64
[1] 81
[1] 100
> mtcars #inbuilt dataset
      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
Hornet Sportabout   18.7   8 360.0 175 3.15 3.440 17.02 0 0   3   2
Valiant              18.1   6 225.0 105 2.76 3.460 20.22 1 0   3   1
Duster 360           14.3   8 360.0 245 3.21 3.570 15.84 0 0   3   4
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
Merc 280             19.2   6 167.6 123 3.92 3.440 18.30 1 0   4   4
Merc 280C            17.8   6 167.6 123 3.92 3.440 18.90 1 0   4   4
Merc 450SE           16.4   8 275.8 180 3.07 4.070 17.40 0 0   3   3
Merc 450SL           17.3   8 275.8 180 3.07 3.730 17.60 0 0   3   3
Merc 450SLC          15.2   8 275.8 180 3.07 3.780 18.00 0 0   3   3
Cadillac Fleetwood  10.4   8 472.0 205 2.93 5.250 17.98 0 0   3   4

```

```

Lincoln Continental 10.4  8 460.0 215 3.00 5.424 17.82 0 0  3  4
Chrysler Imperial  14.7  8 440.0 230 3.23 5.345 17.42 0 0  3  4
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
Dodge Challenger    15.5  8 318.0 150 2.76 3.520 16.87 0 0  3  2
AMC Javelin         15.2  8 304.0 150 3.15 3.435 17.30 0 0  3  2
Camaro Z28          13.3  8 350.0 245 3.73 3.840 15.41 0 0  3  4
Pontiac Firebird    19.2  8 400.0 175 3.08 3.845 17.05 0 0  3  2
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
Ford Pantera L      15.8  8 351.0 264 4.22 3.170 14.50 0 1  5  4
Ferrari Dino        19.7  6 145.0 175 3.62 2.770 15.50 0 1  5  6
Maserati Bora       15.0  8 301.0 335 3.54 3.570 14.60 0 1  5  8
Volvo 142E          21.4  4 121.0 109 4.11 2.780 18.60 1 1  4  2

```

```
> iris #inbuilt dataset
```

```

Sepal.Length Sepal.Width Petal.Length Petal.Width Species
1      5.1      3.5      1.4      0.2  setosa
2      4.9      3.0      1.4      0.2  setosa
3      4.7      3.2      1.3      0.2  setosa
4      4.6      3.1      1.5      0.2  setosa
5      5.0      3.6      1.4      0.2  setosa
6      5.4      3.9      1.7      0.4  setosa
7      4.6      3.4      1.4      0.3  setosa
8      5.0      3.4      1.5      0.2  setosa
9      4.4      2.9      1.4      0.2  setosa
10     4.9      3.1      1.5      0.1  setosa
11     5.4      3.7      1.5      0.2  setosa
12     4.8      3.4      1.6      0.2  setosa
13     4.8      3.0      1.4      0.1  setosa
14     4.3      3.0      1.1      0.1  setosa
15     5.8      4.0      1.2      0.2  setosa
16     5.7      4.4      1.5      0.4  setosa
17     5.4      3.9      1.3      0.4  setosa
18     5.1      3.5      1.4      0.3  setosa
19     5.7      3.8      1.7      0.3  setosa
20     5.1      3.8      1.5      0.3  setosa
21     5.4      3.4      1.7      0.2  setosa
22     5.1      3.7      1.5      0.4  setosa
23     4.6      3.6      1.0      0.2  setosa
24     5.1      3.3      1.7      0.5  setosa
25     4.8      3.4      1.9      0.2  setosa

```



26	5.0	3.0	1.6	0.2	setosa
27	5.0	3.4	1.6	0.4	setosa
28	5.2	3.5	1.5	0.2	setosa
29	5.2	3.4	1.4	0.2	setosa
30	4.7	3.2	1.6	0.2	setosa
31	4.8	3.1	1.6	0.2	setosa
32	5.4	3.4	1.5	0.4	setosa
33	5.2	4.1	1.5	0.1	setosa
34	5.5	4.2	1.4	0.2	setosa
35	4.9	3.1	1.5	0.2	setosa
36	5.0	3.2	1.2	0.2	setosa
37	5.5	3.5	1.3	0.2	setosa
38	4.9	3.6	1.4	0.1	setosa
39	4.4	3.0	1.3	0.2	setosa
40	5.1	3.4	1.5	0.2	setosa
41	5.0	3.5	1.3	0.3	setosa
42	4.5	2.3	1.3	0.3	setosa
43	4.4	3.2	1.3	0.2	setosa
44	5.0	3.5	1.6	0.6	setosa
45	5.1	3.8	1.9	0.4	setosa
46	4.8	3.0	1.4	0.3	setosa
47	5.1	3.8	1.6	0.2	setosa
48	4.6	3.2	1.4	0.2	setosa
49	5.3	3.7	1.5	0.2	setosa
50	5.0	3.3	1.4	0.2	setosa
51	7.0	3.2	4.7	1.4	versicolor
52	6.4	3.2	4.5	1.5	versicolor
53	6.9	3.1	4.9	1.5	versicolor
54	5.5	2.3	4.0	1.3	versicolor
55	6.5	2.8	4.6	1.5	versicolor
56	5.7	2.8	4.5	1.3	versicolor
57	6.3	3.3	4.7	1.6	versicolor
58	4.9	2.4	3.3	1.0	versicolor
59	6.6	2.9	4.6	1.3	versicolor
60	5.2	2.7	3.9	1.4	versicolor
61	5.0	2.0	3.5	1.0	versicolor
62	5.9	3.0	4.2	1.5	versicolor
63	6.0	2.2	4.0	1.0	versicolor
64	6.1	2.9	4.7	1.4	versicolor
65	5.6	2.9	3.6	1.3	versicolor
66	6.7	3.1	4.4	1.4	versicolor
67	5.6	3.0	4.5	1.5	versicolor
68	5.8	2.7	4.1	1.0	versicolor
69	6.2	2.2	4.5	1.5	versicolor

70	5.6	2.5	3.9	1.1 versicolor
71	5.9	3.2	4.8	1.8 versicolor
72	6.1	2.8	4.0	1.3 versicolor
73	6.3	2.5	4.9	1.5 versicolor
74	6.1	2.8	4.7	1.2 versicolor
75	6.4	2.9	4.3	1.3 versicolor
76	6.6	3.0	4.4	1.4 versicolor
77	6.8	2.8	4.8	1.4 versicolor
78	6.7	3.0	5.0	1.7 versicolor
79	6.0	2.9	4.5	1.5 versicolor
80	5.7	2.6	3.5	1.0 versicolor
81	5.5	2.4	3.8	1.1 versicolor
82	5.5	2.4	3.7	1.0 versicolor
83	5.8	2.7	3.9	1.2 versicolor
84	6.0	2.7	5.1	1.6 versicolor
85	5.4	3.0	4.5	1.5 versicolor
86	6.0	3.4	4.5	1.6 versicolor
87	6.7	3.1	4.7	1.5 versicolor
88	6.3	2.3	4.4	1.3 versicolor
89	5.6	3.0	4.1	1.3 versicolor
90	5.5	2.5	4.0	1.3 versicolor
91	5.5	2.6	4.4	1.2 versicolor
92	6.1	3.0	4.6	1.4 versicolor
93	5.8	2.6	4.0	1.2 versicolor
94	5.0	2.3	3.3	1.0 versicolor
95	5.6	2.7	4.2	1.3 versicolor
96	5.7	3.0	4.2	1.2 versicolor
97	5.7	2.9	4.2	1.3 versicolor
98	6.2	2.9	4.3	1.3 versicolor
99	5.1	2.5	3.0	1.1 versicolor
100	5.7	2.8	4.1	1.3 versicolor
101	6.3	3.3	6.0	2.5 virginica
102	5.8	2.7	5.1	1.9 virginica
103	7.1	3.0	5.9	2.1 virginica
104	6.3	2.9	5.6	1.8 virginica
105	6.5	3.0	5.8	2.2 virginica
106	7.6	3.0	6.6	2.1 virginica
107	4.9	2.5	4.5	1.7 virginica
108	7.3	2.9	6.3	1.8 virginica
109	6.7	2.5	5.8	1.8 virginica
110	7.2	3.6	6.1	2.5 virginica
111	6.5	3.2	5.1	2.0 virginica
112	6.4	2.7	5.3	1.9 virginica
113	6.8	3.0	5.5	2.1 virginica

114	5.7	2.5	5.0	2.0	virginica
115	5.8	2.8	5.1	2.4	virginica
116	6.4	3.2	5.3	2.3	virginica
117	6.5	3.0	5.5	1.8	virginica
118	7.7	3.8	6.7	2.2	virginica
119	7.7	2.6	6.9	2.3	virginica
120	6.0	2.2	5.0	1.5	virginica
121	6.9	3.2	5.7	2.3	virginica
122	5.6	2.8	4.9	2.0	virginica
123	7.7	2.8	6.7	2.0	virginica
124	6.3	2.7	4.9	1.8	virginica
125	6.7	3.3	5.7	2.1	virginica
126	7.2	3.2	6.0	1.8	virginica
127	6.2	2.8	4.8	1.8	virginica
128	6.1	3.0	4.9	1.8	virginica
129	6.4	2.8	5.6	2.1	virginica
130	7.2	3.0	5.8	1.6	virginica
131	7.4	2.8	6.1	1.9	virginica
132	7.9	3.8	6.4	2.0	virginica
133	6.4	2.8	5.6	2.2	virginica
134	6.3	2.8	5.1	1.5	virginica
135	6.1	2.6	5.6	1.4	virginica
136	7.7	3.0	6.1	2.3	virginica
137	6.3	3.4	5.6	2.4	virginica
138	6.4	3.1	5.5	1.8	virginica
139	6.0	3.0	4.8	1.8	virginica
140	6.9	3.1	5.4	2.1	virginica
141	6.7	3.1	5.6	2.4	virginica
142	6.9	3.1	5.1	2.3	virginica
143	5.8	2.7	5.1	1.9	virginica
144	6.8	3.2	5.9	2.3	virginica
145	6.7	3.3	5.7	2.5	virginica
146	6.7	3.0	5.2	2.3	virginica
147	6.3	2.5	5.0	1.9	virginica
148	6.5	3.0	5.2	2.0	virginica
149	6.2	3.4	5.4	2.3	virginica
150	5.9	3.0	5.1	1.8	virginica

```
> names(mtcars) #variable/column names
```

```
[1] "mpg" "cyl" "disp" "hp" "drat" "wt" "qsec" "vs" "am" "gear" "carb"
```

```
> for (c in names(mtcars)) { print(c) }
```

```
[1] "mpg"
```

```
[1] "cyl"
```

```
[1] "disp"
```

```
[1] "hp"
```

```

[1] "drat"
[1] "wt"
[1] "qsec"
[1] "vs"
[1] "am"
[1] "gear"
[1] "carb"
> price <- 12.99
> while (price < 15) {
+   price <- price + 1
+   print(price)
+ }
[1] 13.99
[1] 14.99
[1] 15.99
> check_price <- function(x) {
+   if(x > 110) {
+     print("Price beyond threshold")
+   } else {
+     print("Price within threshold")
+   }
+ }
> check_price(200)
[1] "Price beyond threshold"
> myvect <- c(10, 20, 30, NA, 60, 80)
> mean(myvect)
[1] NA
> sd(myvect)
[1] NA
> min(myvect)
[1] NA
> mean(myvect, na.rm = TRUE)
[1] 40
> stock_price <- c(10, 5, 20, 15, 12, 22)
> matrix_form <- matrix(stock_price, ncol = 2, byrow = TRUE)
> matrix_form
      [,1] [,2]
[1,]  10   5
[2,]  20  15
[3,]  12  22
> apply(matrix_form, 1, sum)
[1] 15 35 34
> apply(matrix_form, 2, sum)
[1] 42 42

```

```
> lapply(1:3, function(x) x ^ 2) #Returns list
```

```
[[1]]
```

```
[1] 1
```

```
[[2]]
```

```
[1] 4
```

```
[[3]]
```

```
[1] 9
```

```
> sapply(1:3, function(x) x ^ 2) #Returns vector
```

```
[1] 1 4 9
```

```
> l <- lapply(1:3, function(x) x ^ 2)
```

```
> class(l)
```

```
[1] "list"
```

```
> s <- sapply(1:3, function(x) x ^ 2)
```

```
> class(s)
```

```
[1] "numeric"
```

```
> #Initial Date: 1/1/1970
```

```
> purchase_on <- 365
```

```
> class(purchase_on) <- "Date" #Convert to Date & Adds 365 days to the default date
```

```
> purchase_on
```

```
[1] "1971-01-01"
```

```
> purchase_on <- -10
```

```
> class(purchase_on) <- "Date" #Convert to Date & Subtracts 10 days from the default date
```

```
> purchase_on
```

```
[1] "1969-12-22"
```

```
> purchase_date <- as.Date(365, origin=as.Date("2015-03-31")) #365 days added to origin date
```

```
> purchase_date
```

```
[1] "2016-03-30"
```

```
> sale_date <- as.Date(-10, origin=as.Date("2015-02-10")) #10 days subtracted from origin date
```

```
> sale_date
```

```
[1] "2015-01-31"
```

```
> format(sale_date, "%Y")
```

```
[1] "2015"
```

```
> format(sale_date, "%m")
```

```
[1] "01"
```

```
> format(sale_date, "%b")
```

```
[1] "Jan"
```

```
> format(sale_date, "%B")
```

```
[1] "January"
```

```
> Sys.Date()
```

```
[1] "2022-02-15"
```

```
> format(Sys.Date(), "%d/%m/%Y")
```

```

[1] "15/02/2022"
> as.Date("2021/02/04", format="%Y/%m/%d") #convert a format of date to date type
[1] "2021-02-04"
> as.Date(purchase_date) > as.Date(sale_date)
[1] TRUE
> as.Date(purchase_date) < as.Date(sale_date)
[1] FALSE
> first_date <- "2020-05-16"
> second_date <- "2020-12-24"
> as.Date(first_date) > as.Date(second_date)
[1] FALSE
> as.Date(first_date) < as.Date(second_date)
[1] TRUE
> dim(housing_df)
[1] 545 13
> str(housing_df)
'data.frame': 545 obs. of 13 variables:
 $ price      : int 13300000 12250000 12250000 12215000 11410000 10850000 10150000 10150000
9870000 9800000 ...
 $ area       : int 7420 8960 9960 7500 7420 7500 8580 16200 8100 5750 ...
 $ bedrooms   : int 4 4 3 4 4 3 4 5 4 3 ...
 $ bathrooms   : int 2 4 2 2 1 3 3 3 1 2 ...
 $ stories     : int 3 4 2 2 2 1 4 2 2 4 ...
 $ mainroad    : chr "yes" "yes" "yes" "yes" ...
 $ guestroom   : chr "no" "no" "no" "no" ...
 $ basement    : chr "no" "no" "yes" "yes" ...
 $ hotwaterheating : chr "no" "no" "no" "no" ...
 $ airconditioning : chr "yes" "yes" "no" "yes" ...
 $ parking     : int 2 3 2 3 2 2 2 0 2 1 ...
 $ prefarea    : chr "yes" "no" "yes" "yes" ...
 $ furnishingstatus: chr "furnished" "furnished" "semi-furnished" "furnished" ...
> summary(housing_df)
  price      area      bedrooms  bathrooms
Min.   :1750000 Min.   :1650  Min.   :1.000  Min.   :1.000
1st Qu.: 3430000 1st Qu.: 3600  1st Qu.:2.000  1st Qu.:1.000
Median : 4340000 Median : 4600  Median :3.000  Median :1.000
Mean   : 4766729 Mean   : 5151  Mean   :2.965  Mean   :1.286
3rd Qu.: 5740000 3rd Qu.: 6360  3rd Qu.:3.000  3rd Qu.:2.000
Max.   :13300000 Max.   :16200 Max.   :6.000  Max.   :4.000
 stories   mainroad   guestroom   basement
Min.   :1.000 Length:545   Length:545   Length:545
1st Qu.:1.000 Class:character Class:character Class:character
Median :2.000 Mode :character Mode :character Mode :character
Mean   :1.806

```

```

3rd Qu.:2.000
Max. :4.000
hotwaterheating airconditioning parking prefarea
Length:545 Length:545 Min. :0.0000 Length:545
Class :character Class :character 1st Qu.:0.0000 Class :character
Mode :character Mode :character Median :0.0000 Mode :character
      Mean :0.6936
      3rd Qu.:1.0000
      Max. :3.0000

furnishingstatus
Length:545
Class :character
Mode :character

```

### Day 3 – R Programming

```

> ages <- c(34, 45, 26, 32, 21)
> location <- c("Urban", "Rural", "Urban", "Rural", "Urban")
> tapply(ages, location, mean) #location wise age mean
Rural Urban
38.5 27.0
> #history() #get previous command
> setwd("C:/zubeda/PGA02_Zubu/R Programming") #Set current working directory
> housing_df = read.csv("Housing.csv")
> housing_df
      price area bedrooms bathrooms stories mainroad guestroom basement hotwaterheating
airconditioning parking
1 13300000 7420 4 2 3 yes no no no yes 2
2 12250000 8960 4 4 4 yes no no no yes 3
3 12250000 9960 3 2 2 yes no yes no no 2
4 12215000 7500 4 2 2 yes no yes no yes 3
5 11410000 7420 4 1 2 yes yes yes no yes 2
6 10850000 7500 3 3 1 yes no yes no yes 2
7 10150000 8580 4 3 4 yes no no no yes 2
8 10150000 16200 5 3 2 yes no no no no 0
9 9870000 8100 4 1 2 yes yes yes no yes 2
10 9800000 5750 3 2 4 yes yes no no yes 1
11 9800000 13200 3 1 2 yes no yes no yes 2
12 9681000 6000 4 3 2 yes yes yes yes no 2
13 9310000 6550 4 2 2 yes no no no yes 1
14 9240000 3500 4 2 2 yes no no yes no 2
15 9240000 7800 3 2 2 yes no no no no 0
16 9100000 6000 4 1 2 yes no yes no no 2
17 9100000 6600 4 2 2 yes yes yes no yes 1
18 8960000 8500 3 2 4 yes no no no yes 2

```

19	8890000	4600	3	2	2	yes	yes	no	no	yes	2
20	8855000	6420	3	2	2	yes	no	no	no	yes	1
21	8750000	4320	3	1	2	yes	no	yes	yes	no	2
22	8680000	7155	3	2	1	yes	yes	yes	no	yes	2
23	8645000	8050	3	1	1	yes	yes	yes	no	yes	1
24	8645000	4560	3	2	2	yes	yes	yes	no	yes	1
25	8575000	8800	3	2	2	yes	no	no	no	yes	2
26	8540000	6540	4	2	2	yes	yes	yes	no	yes	2
27	8463000	6000	3	2	4	yes	yes	yes	no	yes	0
28	8400000	8875	3	1	1	yes	no	no	no	no	1
29	8400000	7950	5	2	2	yes	no	yes	yes	no	2
30	8400000	5500	4	2	2	yes	no	yes	no	yes	1
31	8400000	7475	3	2	4	yes	no	no	no	yes	2
32	8400000	7000	3	1	4	yes	no	no	no	yes	2
33	8295000	4880	4	2	2	yes	no	no	no	yes	1
34	8190000	5960	3	3	2	yes	yes	yes	no	no	1
35	8120000	6840	5	1	2	yes	yes	yes	no	yes	1
36	8080940	7000	3	2	4	yes	no	no	no	yes	2
37	8043000	7482	3	2	3	yes	no	no	yes	no	1
38	7980000	9000	4	2	4	yes	no	no	no	yes	2
39	7962500	6000	3	1	4	yes	yes	no	no	yes	2
40	7910000	6000	4	2	4	yes	no	no	no	yes	1
41	7875000	6550	3	1	2	yes	no	yes	no	yes	0
42	7840000	6360	3	2	4	yes	no	no	no	yes	0
43	7700000	6480	3	2	4	yes	no	no	no	yes	2
44	7700000	6000	4	2	4	yes	no	no	no	no	2
45	7560000	6000	4	2	4	yes	no	no	no	yes	1
46	7560000	6000	3	2	3	yes	no	no	no	yes	0
47	7525000	6000	3	2	4	yes	no	no	no	yes	1
48	7490000	6600	3	1	4	yes	no	no	no	yes	3
49	7455000	4300	3	2	2	yes	no	yes	no	no	1
50	7420000	7440	3	2	1	yes	yes	yes	no	yes	0
51	7420000	7440	3	2	4	yes	no	no	no	no	1
52	7420000	6325	3	1	4	yes	no	no	no	yes	1
53	7350000	6000	4	2	4	yes	yes	no	no	yes	1
54	7350000	5150	3	2	4	yes	no	no	no	yes	2
55	7350000	6000	3	2	2	yes	yes	no	no	yes	1
56	7350000	6000	3	1	2	yes	no	no	no	yes	1
57	7343000	11440	4	1	2	yes	no	yes	no	no	1
58	7245000	9000	4	2	4	yes	yes	no	no	yes	1
59	7210000	7680	4	2	4	yes	yes	no	no	yes	1
60	7210000	6000	3	2	4	yes	yes	no	no	yes	1
61	7140000	6000	3	2	2	yes	yes	no	no	no	1
62	7070000	8880	2	1	1	yes	no	no	no	yes	1



63	7070000	6240	4	2	2	yes	no	no	no	yes	1
64	7035000	6360	4	2	3	yes	no	no	no	yes	2
65	7000000	11175	3	1	1	yes	no	yes	no	yes	1
66	6930000	8880	3	2	2	yes	no	yes	no	yes	1
67	6930000	13200	2	1	1	yes	no	yes	yes	no	1
68	6895000	7700	3	2	1	yes	no	no	no	no	2
69	6860000	6000	3	1	1	yes	no	no	no	yes	1
70	6790000	12090	4	2	2	yes	no	no	no	no	2
71	6790000	4000	3	2	2	yes	no	yes	no	yes	0
72	6755000	6000	4	2	4	yes	no	no	no	yes	0
73	6720000	5020	3	1	4	yes	no	no	no	yes	0
74	6685000	6600	2	2	4	yes	no	yes	no	no	0
75	6650000	4040	3	1	2	yes	no	yes	yes	no	1
76	6650000	4260	4	2	2	yes	no	no	yes	no	0

prefarea furnishingstatus

1	yes	furnished
2	no	furnished
3	yes	semi-furnished
4	yes	furnished
5	no	furnished
6	yes	semi-furnished
7	yes	semi-furnished
8	no	unfurnished
9	yes	furnished
10	yes	unfurnished
11	yes	furnished
12	no	semi-furnished
13	yes	semi-furnished
14	no	furnished
15	yes	semi-furnished
16	no	semi-furnished
17	yes	unfurnished
18	no	furnished
19	no	furnished
20	yes	semi-furnished
21	no	semi-furnished
22	no	unfurnished
23	no	furnished
24	no	furnished
25	no	furnished
26	yes	furnished
27	yes	semi-furnished
28	no	semi-furnished
29	no	unfurnished

30	yes	semi-furnished
31	no	unfurnished
32	no	semi-furnished
33	yes	furnished
34	no	unfurnished
35	no	furnished
36	no	furnished
37	yes	furnished
38	no	furnished
39	no	unfurnished
40	no	semi-furnished
41	yes	furnished
42	yes	furnished
43	no	unfurnished
44	no	semi-furnished
45	no	furnished
46	no	semi-furnished
47	no	furnished
48	yes	furnished
49	no	unfurnished
50	yes	semi-furnished
51	yes	unfurnished
52	no	unfurnished
53	no	furnished
54	no	semi-furnished
55	no	semi-furnished
56	no	unfurnished
57	yes	semi-furnished
58	yes	furnished
59	no	semi-furnished
60	no	furnished
61	no	semi-furnished
62	no	semi-furnished
63	no	furnished
64	yes	furnished
65	yes	furnished
66	no	furnished
67	no	furnished
68	no	unfurnished
69	no	furnished
70	yes	furnished
71	yes	semi-furnished
72	no	unfurnished
73	yes	unfurnished

```

74  yes    furnished
75  no     furnished
76  no     semi-furnished
[ reached 'max' / getOption("max.print") -- omitted 469 rows ]
> dev.off()      #clear plot window
null device
      1
> par(mfrow=c(2,1)) #subplots/partions of 2 rows, 1 col
> #Univariate Analysis
> hist(housing_df$area, col = "orange")
> boxplot(housing_df$area, col = "light blue")
> dev.off()
null device
      1
> boxplot(housing_df$area, horizontal = T, col = "light blue")
> dev.off()
null device
      1
> summary(mtcars)
      mpg      cyl      disp      hp      drat      wt      qsec
Min. :10.40 Min. :4.000 Min. : 71.1 Min. : 52.0 Min. :2.760 Min. :1.513 Min. :14.50
1st Qu.:15.43 1st Qu.:4.000 1st Qu.:120.8 1st Qu.: 96.5 1st Qu.:3.080 1st Qu.:2.581 1st Qu.:16.89
Median :19.20 Median :6.000 Median :196.3 Median :123.0 Median :3.695 Median :3.325
Median :17.71
Mean :20.09 Mean :6.188 Mean :230.7 Mean :146.7 Mean :3.597 Mean :3.217 Mean
:17.85
3rd Qu.:22.80 3rd Qu.:8.000 3rd Qu.:326.0 3rd Qu.:180.0 3rd Qu.:3.920 3rd Qu.:3.610 3rd
Qu.:18.90
Max. :33.90 Max. :8.000 Max. :472.0 Max. :335.0 Max. :4.930 Max. :5.424 Max. :22.90
      vs      am      gear      carb
Min. :0.0000 Min. :0.0000 Min. :3.000 Min. :1.000
1st Qu.:0.0000 1st Qu.:0.0000 1st Qu.:3.000 1st Qu.:2.000
Median :0.0000 Median :0.0000 Median :4.000 Median :2.000
Mean :0.4375 Mean :0.4062 Mean :3.688 Mean :2.812
3rd Qu.:1.0000 3rd Qu.:1.0000 3rd Qu.:4.000 3rd Qu.:4.000
Max. :1.0000 Max. :1.0000 Max. :5.000 Max. :8.000
> #Bivariate Analysis
> table(mtcars$vs, mtcars$gear) #Frequency table/Cross table

      3 4 5
0 12 2 4
1 3 10 1
> #row index - vs, col index - gear
> df_numeric_vars <- Filter(is.numeric, housing_df) #Filter(condition, df)

```

```

> names(df_numeric_vars)
[1] "price" "area" "bedrooms" "bathrooms" "stories" "parking"
> df_categorical_vars <- Filter(is.factor, housing_df)
> names(df_categorical_vars)
character(0)
> rownames(mtcars)
[1] "Mazda RX4" "Mazda RX4 Wag" "Datsun 710" "Hornet 4 Drive"
[5] "Hornet Sportabout" "Valiant" "Duster 360" "Merc 240D"
[9] "Merc 230" "Merc 280" "Merc 280C" "Merc 450SE"
[13] "Merc 450SL" "Merc 450SLC" "Cadillac Fleetwood" "Lincoln Continental"
[17] "Chrysler Imperial" "Fiat 128" "Honda Civic" "Toyota Corolla"
[21] "Toyota Corona" "Dodge Challenger" "AMC Javelin" "Camaro Z28"
[25] "Pontiac Firebird" "Fiat X1-9" "Porsche 914-2" "Lotus Europa"
[29] "Ford Pantera L" "Ferrari Dino" "Maserati Bora" "Volvo 142E"
> #?data/fn/keyword - get help documentation internally
> #??data/fn/keyword - get help documentation online
> ?mtcars
> ?iris
> counts <- table(mtcars$vs, mtcars$gear)
> #Side by Side barplot
> barplot(counts, main="Car Distribution by Gears and VS", xlab="Number of Gears", ylab="Frequency",
col=c("darkblue", "red"), legend=rownames(counts), beside=TRUE)
> dev.off()
null device
1
> #Stacked barplot
> barplot(counts, main="Car Distribution by Gears and VS", xlab="Number of Gears", ylab="Frequency",
col=c("darkblue", "red"), legend=rownames(counts), names.arg=c("3", "4", "5"))
> #names.arg - label appear at the bottom of each bar
> nas <- sapply(housing_df, function(X) sum(is.na(x))) #Missing value checking
> nas
      price      area bedrooms bathrooms stories  mainroad
      0         0         0         0         0         0
  guestroom  basement hotwaterheating airconditioning parking  prefarea
      0         0         0         0         0         0
furnishingstatus
      0
> missing_percent <- (nas * 100) / (nrow(housing_df))
> missing_percent
      price      area bedrooms bathrooms stories  mainroad
      0         0         0         0         0         0
  guestroom  basement hotwaterheating airconditioning parking  prefarea
      0         0         0         0         0         0
furnishingstatus

```

```

0
> colnames(mtcars)
[1] "mpg" "cyl" "disp" "hp" "drat" "wt" "qsec" "vs" "am" "gear" "carb"
> names(mtcars)
[1] "mpg" "cyl" "disp" "hp" "drat" "wt" "qsec" "vs" "am" "gear" "carb"
> dev.off()
null device
1
> library(dplyr)
> library(ggplot2)
> data.frame(missing_percent, variable=colnames(housing_df))%>% #redirection operator/pipe
operator for chaining commands with dependency, passing output of one to another
+ ggplot(aes(variable, missing_percent)) +
+ geom_bar(stat="identity") + #height of bars to represent values in the data
+ labs(x="Features", y="Percent of Missing values") +
+ theme(axis.text.x=element_text(angle=90, hjust=1))
> #aes(reorder(variable col, - or + the variable to be sorted)) sorts output in asc or desc order
> paste("Hello", "Everybody") #Concat elements seperated by spaces
[1] "Hello Everybody"
> paste("A", "1", sep="") #Concat elements with no spaces
[1] "A1"
> x <- c(32, 12, 30, 45)
> labels <- c("Mumbai", "Chennai", "Pune", "Banglore")
> pct <- round(x / sum(x) * 100)
> lbls <- paste(labels, pct)
> lbls <- paste(lbls, "%", sep="")
> pct
[1] 27 10 25 38
> lbls
[1] "Mumbai 27%" "Chennai 10%" "Pune 25%" "Banglore 38%"
> pie(x, labels=lbls, col=rainbow(length(lbls)), main="City Pie Chart") #rainbow(length) will generate 4
hexdecimal values
> legend("topright", c("Mumbai", "Chennai", "Pune", "Banglore"), cex=0.5, fill=rainbow(length(x)))
#cex=Controls zoom of the font
> legend("topright", c("Mumbai", "Chennai", "Pune", "Banglore"), cex=1, fill=rainbow(length(x)))
> #install.packages("Quandl")
> library("Quandl")

```

## Day 4 – R Programming

```

> dev.off()
null device
1
> setwd("C:/zubeda/PGA02_Zubu/R Programming")
> library("plyr")

```

```

> library("ggplot2")
> df_AP <- read.csv("ADANIPORTS.csv")
> edit(df_AP)

```

	Date	Symbol	Series	Prev.Close	Open	High	Low	Last	Close	VWAP	Volume	Turnover
1	2007-11-27	MUNDRAPORT	EQ	440.00	770.00	1050.00	770.00	959.00	962.90	984.72	27294366	2687719053785000
2	2007-11-28	MUNDRAPORT	EQ	962.90	984.00	990.00	874.00	885.00	893.90	941.38	4581338	431276530165000
3	2007-11-29	MUNDRAPORT	EQ	893.90	909.00	914.75	841.00	887.00	884.20	888.09	5124121	455065846265000
4	2007-11-30	MUNDRAPORT	EQ	884.20	890.00	958.00	890.00	929.00	921.55	929.17	4609762	428325662830000
5	2007-12-03	MUNDRAPORT	EQ	921.55	939.75	995.00	922.00	980.00	969.30	965.65	2977470	287519974300000
6	2007-12-04	MUNDRAPORT	EQ	969.30	985.00	1056.00	976.00	1049.00	1041.45	1015.39		4849250 492386736075000
7	2007-12-05	MUNDRAPORT	EQ	1041.45	1061.00	1099.50	1050.00	1084.00	1082.45	1082.79		2848209 308400973015000
8	2007-12-06	MUNDRAPORT	EQ	1082.45	1089.00	1109.70	1051.00	1090.10	1081.30	1087.03		1749516 190177114020000
9	2007-12-07	MUNDRAPORT	EQ	1081.30	1100.00	1134.00	1078.00	1100.00	1102.40	1106.57		2247904 248746530710000
10	2007-12-10	MUNDRAPORT	EQ	1102.40	1110.00	1110.00	1061.10	1073.55	1075.40	1080.38		1012350 109372679360000
11	2007-12-11	MUNDRAPORT	EQ	1075.40	1081.00	1089.00	1041.00	1046.00	1047.65	1067.80		810464 86541556460000
12	2007-12-12	MUNDRAPORT	EQ	1047.65	1032.00	1065.00	1016.00	1036.90	1036.80	1043.92		744799 77751369165000
13	2007-12-13	MUNDRAPORT	EQ	1036.80	1040.00	1150.00	1030.25	1131.15	1129.95	1109.09		3067687 340233907520000
14	2007-12-14	MUNDRAPORT	EQ	1129.95	1139.90	1140.00	1101.10	1107.00	1110.50	1119.55		1070737 119874627765000
15	2007-12-17	MUNDRAPORT	EQ	1110.50	1140.00	1168.00	1021.50	1052.00	1044.25	1102.42		1404955 154884767715000
16	2007-12-18	MUNDRAPORT	EQ	1044.25	1045.00	1109.90	1031.55	1085.00	1074.95	1077.84		1226984 132249513310000
17	2007-12-19	MUNDRAPORT	EQ	1074.95	1091.00	1116.00	1046.30	1078.00	1066.90	1082.93		845666 91579757645000
18	2007-12-20	MUNDRAPORT	EQ	1066.90	1083.50	1083.50	1051.00	1067.00	1060.20	1065.52		623288 66412706110000
19	2007-12-24	MUNDRAPORT	EQ	1060.20	1095.00	1192.00	1085.25	1160.00	1156.80	1160.77		2060892 239221361310000
20	2007-12-26	MUNDRAPORT	EQ	1156.80	1175.00	1214.00	1148.00	1212.00	1199.90	1183.30		1467031 173593856540000

21	2007-12-27	MUNDRAPORT	EQ	1199.90	1215.00	1240.00	1204.00	1209.00	1211.65	1222.58
977495 119506465945000										
22	2007-12-28	MUNDRAPORT	EQ	1211.65	1189.40	1274.00	1175.00	1270.00	1249.10	1221.31
1164138 142177280540000										
23	2007-12-31	MUNDRAPORT	EQ	1249.10	1263.35	1295.00	1261.00	1268.00	1268.80	1277.64
737249 94194213815000										
24	2008-01-01	MUNDRAPORT	EQ	1268.80	1279.00	1319.00	1263.70	1308.00	1296.85	1285.72
491348 63173462100000										
25	2008-01-02	MUNDRAPORT	EQ	1296.85	1310.25	1324.00	1270.00	1300.15	1307.45	1302.15
703815 91647340425000										
26	2008-01-03	MUNDRAPORT	EQ	1307.45	1305.00	1314.70	1261.15	1267.15	1275.80	1289.24
505058 65114250075000										
27	2008-01-04	MUNDRAPORT	EQ	1275.80	1278.80	1294.80	1233.00	1239.90	1240.35	1256.03
550795 69181674340000										
28	2008-01-07	MUNDRAPORT	EQ	1240.35	1240.00	1278.90	1215.00	1233.00	1227.25	1244.76
630963 78539769975000										
29	2008-01-08	MUNDRAPORT	EQ	1227.25	1240.00	1255.00	1185.00	1202.00	1204.80	1217.08
530499 64565951270000										
30	2008-01-09	MUNDRAPORT	EQ	1204.80	1200.00	1210.00	1151.00	1181.00	1180.25	1176.37
627507 73818313330000										
31	2008-01-10	MUNDRAPORT	EQ	1180.25	1185.00	1199.80	1110.00	1118.00	1121.55	1156.44
438806 50745246590000										
32	2008-01-11	MUNDRAPORT	EQ	1121.55	1128.00	1130.00	1063.00	1096.00	1085.85	1087.78
616938 67109272025000										
33	2008-01-14	MUNDRAPORT	EQ	1085.85	1082.40	1082.40	1031.10	1035.00	1035.15	1042.40
835916 87135710755000										
34	2008-01-15	MUNDRAPORT	EQ	1035.15	1045.60	1078.70	1036.05	1057.00	1049.55	1050.69
830493 87259337110000										
35	2008-01-16	MUNDRAPORT	EQ	1049.55	1046.00	1064.00	1000.00	1038.30	1030.40	1032.86
816188 84300609685000										
36	2008-01-17	MUNDRAPORT	EQ	1030.40	1050.00	1053.50	1011.00	1014.95	1020.90	1033.73
336003 34733490900000										
37	2008-01-18	MUNDRAPORT	EQ	1020.90	1010.00	1072.00	974.90	995.00	994.60	1022.57
676854 69213280915000										
38	2008-01-21	MUNDRAPORT	EQ	994.60	995.00	1005.00	795.70	853.00	825.05	880.77 788623
69459899855000										
39	2008-01-22	MUNDRAPORT	EQ	825.05	700.00	810.00	660.05	739.00	735.55	703.20 546161
38406113705000										
40	2008-01-23	MUNDRAPORT	EQ	735.55	760.00	881.90	760.00	862.20	857.00	818.67 535462
43836526980000										
41	2008-01-24	MUNDRAPORT	EQ	857.00	875.00	935.00	812.00	814.70	814.15	854.83 511017
43683319425000										
42	2008-01-25	MUNDRAPORT	EQ	814.15	820.00	883.00	820.00	866.00	865.70	858.33 404045
34680333860000										

43	2008-01-28	MUNDRAPORT	EQ	865.70	835.00	835.00	783.20	822.00	820.80	804.38	467052
37568552380000											
44	2008-01-29	MUNDRAPORT	EQ	820.80	840.00	860.00	820.05	840.00	840.75	841.27	220070
18513823345000											
45	2008-01-30	MUNDRAPORT	EQ	840.75	849.80	864.00	822.25	834.00	830.45	833.82	286190
23863110660000											
46	2008-01-31	MUNDRAPORT	EQ	830.45	831.00	849.90	812.55	836.60	837.65	833.58	194300
16196555895000											
47	2008-02-01	MUNDRAPORT	EQ	837.65	831.65	852.30	820.00	826.00	825.35	828.09	204391
16925451805000											
48	2008-02-04	MUNDRAPORT	EQ	825.35	847.90	872.40	840.00	859.00	856.10	858.77	280230
24065208695000											
49	2008-02-05	MUNDRAPORT	EQ	856.10	856.00	857.00	830.00	834.65	834.30	842.06	162093
13649192020000											
50	2008-02-06	MUNDRAPORT	EQ	834.30	803.00	824.90	780.00	809.00	807.50	810.50	193260
15663794125000											
51	2008-02-07	MUNDRAPORT	EQ	807.50	825.00	830.00	792.00	795.90	796.25	809.53	212932
17237575975000											
52	2008-02-08	MUNDRAPORT	EQ	796.25	810.00	830.00	765.15	786.00	784.05	781.48	285025
22274252000000											
53	2008-02-11	MUNDRAPORT	EQ	784.05	785.00	785.00	695.00	699.00	711.20	736.23	223955
16488264325000											
54	2008-02-12	MUNDRAPORT	EQ	711.20	725.00	734.95	655.60	689.00	681.30	681.38	303409
20673577510000											
55	2008-02-13	MUNDRAPORT	EQ	681.30	815.90	815.90	664.00	678.00	670.95	681.68	214900
14649214640000											
56	2008-02-14	MUNDRAPORT	EQ	670.95	680.00	714.00	680.00	710.00	709.80	704.71	269032
18959036175000											
57	2008-02-15	MUNDRAPORT	EQ	709.80	700.00	763.70	681.25	729.00	728.75	734.23	353049
25921872820000											
58	2008-02-18	MUNDRAPORT	EQ	728.75	735.00	775.00	735.00	772.00	771.60	762.33	342580
26115882900000											
59	2008-02-19	MUNDRAPORT	EQ	771.60	779.00	786.90	760.20	767.00	763.90	772.24	137412
10611555840000											
60	2008-02-20	MUNDRAPORT	EQ	763.90	750.00	760.00	720.00	740.00	732.10	730.61	197489
14428706935000											
61	2008-02-21	MUNDRAPORT	EQ	732.10	762.00	762.00	730.10	738.90	737.60	741.53	125558
9310465240000											
62	2008-02-22	MUNDRAPORT	EQ	737.60	723.00	737.00	715.00	724.50	724.00	726.52	81070
5889922195000											
63	2008-02-25	MUNDRAPORT	EQ	724.00	725.05	758.90	702.30	707.00	707.65	711.70	152803
10875065635000											
64	2008-02-26	MUNDRAPORT	EQ	707.65	725.00	744.00	713.00	735.00	735.80	733.73	251269
18436350425000											



65 2008-02-27 MUNDRAPORT EQ 735.80 749.70 783.40 741.00 744.00 746.40 762.47 305320  
23279802440000

66 2008-02-28 MUNDRAPORT EQ 746.40 740.00 754.90 725.05 740.00 737.75 738.91 112491  
8312092510000

Trades Deliverable.Volume X.Deliverble

1	NA	9859619	0.3612
2	NA	1453278	0.3172
3	NA	1069678	0.2088
4	NA	1260913	0.2735
5	NA	816123	0.2741
6	NA	1537667	0.3171
7	NA	904260	0.3175
8	NA	825691	0.4720
9	NA	697763	0.3104
10	NA	417514	0.4124
11	NA	415191	0.5123
12	NA	363848	0.4885
13	NA	1040076	0.3390
14	NA	525239	0.4905
15	NA	670298	0.4771
16	NA	449420	0.3663
17	NA	344171	0.4070
18	NA	276356	0.4434
19	NA	807879	0.3920
20	NA	469389	0.3200
21	NA	355431	0.3636
22	NA	503564	0.4326
23	NA	316377	0.4291
24	NA	172911	0.3519
25	NA	221397	0.3146
26	NA	217437	0.4305
27	NA	230237	0.4180
28	NA	239404	0.3794
29	NA	228866	0.4314
30	NA	259280	0.4132
31	NA	200150	0.4561
32	NA	312121	0.5059
33	NA	570824	0.6829
34	NA	504259	0.6072
35	NA	478517	0.5863
36	NA	145194	0.4321
37	NA	278615	0.4116
38	NA	474223	0.6013
39	NA	376194	0.6888

40	NA	283881	0.5302
41	NA	258346	0.5056
42	NA	178177	0.4410
43	NA	241365	0.5168
44	NA	74141	0.3369
45	NA	165926	0.5798
46	NA	103890	0.5347
47	NA	115715	0.5661
48	NA	128195	0.4575
49	NA	96153	0.5932
50	NA	110565	0.5721
51	NA	106275	0.4991
52	NA	154857	0.5433
53	NA	118002	0.5269
54	NA	187180	0.6169
55	NA	108761	0.5061
56	NA	148611	0.5524
57	NA	110621	0.3133
58	NA	154099	0.4498
59	NA	47543	0.3460
60	NA	89397	0.4527
61	NA	37956	0.3023
62	NA	31808	0.3924
63	NA	71403	0.4673
64	NA	53136	0.2115
65	NA	84490	0.2767
66	NA	36730	0.3265

```

5 2007-12-03 MUNDRAPOET   EQ   921.55 939.75 995.00 922 980 969.30 965.65 2977470
287519974300000   NA
6 2007-12-04 MUNDRAPOET   EQ   969.30 985.00 1056.00 976 1049 1041.45 1015.39 4849250
492386736075000   NA

```

Deliverable.Volume X.Deliverble

```

1      9859619    0.3612
2      1453278    0.3172
3      1069678    0.2088
4      1260913    0.2735
5       816123    0.2741
6      1537667    0.3171

```

```

> v <- c(8, 14, 26, 5, 43)
> plot(v, type="o") #Line plot with points
> plot(v, type="p") #Points plot
> plot(v, type="l") #Line plot without points
> plot(v, type="o", col="red", xlab="Month", ylab="Rainfall", main="Rainfall Chart")
> v <- c(12, 14, 28, 5, 44)
> t <- c(15, 8, 8, 10, 13)
> plot(v, type="o", col="blue", xlab="Month", ylab="Rainfall", main="Rainfall Chart")
> lines(t, type="o", col="red")
> df_aapl <- read.csv("AAPL.csv")
> head(df_aapl)

```

Date Open High Low Close Adj.Close Volume

```

1 2021-02-17 131.25 132.22 129.47 130.84 130.0669 97918500
2 2021-02-18 129.20 130.00 127.41 129.71 128.9436 96856700
3 2021-02-19 130.24 130.71 128.80 129.87 129.1027 87668800
4 2021-02-22 128.01 129.72 125.60 126.00 125.2555 103916400
5 2021-02-23 123.76 126.71 118.39 125.86 125.1164 158273000
6 2021-02-24 124.94 125.56 122.23 125.35 124.6094 111039900

```

```

> df_waltdisney <- read.csv("DIS.csv")
> head(df_waltdisney)

```

Date Open High Low Close Adj.Close Volume

```

1 2021-02-17 185.36 187.63 182.16 186.44 186.44 11391800
2 2021-02-18 184.79 186.40 182.84 183.00 183.00 12380900
3 2021-02-19 184.27 184.78 182.79 183.65 183.65 8834500
4 2021-02-22 181.74 194.02 181.53 191.76 191.76 18799600
5 2021-02-23 193.59 198.94 188.66 197.09 197.09 23191400
6 2021-02-24 197.58 200.60 195.33 197.51 197.51 16205900

```

```

> df_nike <- read.csv("NKE.csv")
> head(df_nike)

```

Date Open High Low Close Adj.Close Volume

```

1 2021-02-17 141.30 144.56 140.21 143.99 142.9153 6437100
2 2021-02-18 142.98 145.39 141.21 145.09 144.0071 4486800
3 2021-02-19 145.43 145.50 141.50 142.02 140.9601 7486000

```

```

4 2021-02-22 141.54 142.46 136.26 136.67 135.6500 8985900
5 2021-02-23 136.03 136.83 131.58 136.13 135.1140 10364100
6 2021-02-24 135.06 135.96 133.95 135.65 134.6376 6360900
> df_aapl <- cbind(df_aapl, Stock="")
> df_waltdisney <- cbind(df_waltdisney, Stock="")
> df_nike <- cbind(df_nike, Stock="")
> head(df_aapl)
  Date Open High Low Close Adj.Close Volume Stock
1 2021-02-17 131.25 132.22 129.47 130.84 130.0669 97918500
2 2021-02-18 129.20 130.00 127.41 129.71 128.9436 96856700
3 2021-02-19 130.24 130.71 128.80 129.87 129.1027 87668800
4 2021-02-22 128.01 129.72 125.60 126.00 125.2555 103916400
5 2021-02-23 123.76 126.71 118.39 125.86 125.1164 158273000
6 2021-02-24 124.94 125.56 122.23 125.35 124.6094 111039900
> head(df_waltdisney)
  Date Open High Low Close Adj.Close Volume Stock
1 2021-02-17 185.36 187.63 182.16 186.44 186.44 11391800
2 2021-02-18 184.79 186.40 182.84 183.00 183.00 12380900
3 2021-02-19 184.27 184.78 182.79 183.65 183.65 8834500
4 2021-02-22 181.74 194.02 181.53 191.76 191.76 18799600
5 2021-02-23 193.59 198.94 188.66 197.09 197.09 23191400
6 2021-02-24 197.58 200.60 195.33 197.51 197.51 16205900
> head(df_nike)
  Date Open High Low Close Adj.Close Volume Stock
1 2021-02-17 141.30 144.56 140.21 143.99 142.9153 6437100
2 2021-02-18 142.98 145.39 141.21 145.09 144.0071 4486800
3 2021-02-19 145.43 145.50 141.50 142.02 140.9601 7486000
4 2021-02-22 141.54 142.46 136.26 136.67 135.6500 8985900
5 2021-02-23 136.03 136.83 131.58 136.13 135.1140 10364100
6 2021-02-24 135.06 135.96 133.95 135.65 134.6376 6360900
> df_aapl$Stock <- paste(df_aapl$Stock, "Bertrandt", sep="")
> df_waltdisney$Stock <- paste(df_waltdisney$Stock, "Deutsche Bank", sep="")
> df_nike$Stock <- paste(df_nike$Stock, "Siemens", sep="")
> head(df_aapl)
  Date Open High Low Close Adj.Close Volume Stock
1 2021-02-17 131.25 132.22 129.47 130.84 130.0669 97918500 Bertrandt
2 2021-02-18 129.20 130.00 127.41 129.71 128.9436 96856700 Bertrandt
3 2021-02-19 130.24 130.71 128.80 129.87 129.1027 87668800 Bertrandt
4 2021-02-22 128.01 129.72 125.60 126.00 125.2555 103916400 Bertrandt
5 2021-02-23 123.76 126.71 118.39 125.86 125.1164 158273000 Bertrandt
6 2021-02-24 124.94 125.56 122.23 125.35 124.6094 111039900 Bertrandt
> head(df_waltdisney)
  Date Open High Low Close Adj.Close Volume Stock
1 2021-02-17 185.36 187.63 182.16 186.44 186.44 11391800 Deutsche Bank

```

```

2 2021-02-18 184.79 186.40 182.84 183.00 183.00 12380900 Deutsche Bank
3 2021-02-19 184.27 184.78 182.79 183.65 183.65 8834500 Deutsche Bank
4 2021-02-22 181.74 194.02 181.53 191.76 191.76 18799600 Deutsche Bank
5 2021-02-23 193.59 198.94 188.66 197.09 197.09 23191400 Deutsche Bank
6 2021-02-24 197.58 200.60 195.33 197.51 197.51 16205900 Deutsche Bank

```

```
> head(df_nike)
```

```

      Date Open  High  Low Close Adj.Close Volume Stock
1 2021-02-17 141.30 144.56 140.21 143.99 142.9153 6437100 Siemens
2 2021-02-18 142.98 145.39 141.21 145.09 144.0071 4486800 Siemens
3 2021-02-19 145.43 145.50 141.50 142.02 140.9601 7486000 Siemens
4 2021-02-22 141.54 142.46 136.26 136.67 135.6500 8985900 Siemens
5 2021-02-23 136.03 136.83 131.58 136.13 135.1140 10364100 Siemens
6 2021-02-24 135.06 135.96 133.95 135.65 134.6376 6360900 Siemens

```

```
> df_allStocks <- rbind(df_aapl, df_waltdisney, df_nike)
```

```
> df_allStocks
```

```

      Date Open  High  Low Close Adj.Close Volume Stock
1 2021-02-17 131.25 132.22 129.47 130.84 130.0669 97918500 Bertrandt
2 2021-02-18 129.20 130.00 127.41 129.71 128.9436 96856700 Bertrandt
3 2021-02-19 130.24 130.71 128.80 129.87 129.1027 87668800 Bertrandt
4 2021-02-22 128.01 129.72 125.60 126.00 125.2555 103916400 Bertrandt
5 2021-02-23 123.76 126.71 118.39 125.86 125.1164 158273000 Bertrandt
6 2021-02-24 124.94 125.56 122.23 125.35 124.6094 111039900 Bertrandt
7 2021-02-25 124.68 126.46 120.54 120.99 120.2751 148199500 Bertrandt
8 2021-02-26 122.59 124.85 121.20 121.26 120.5436 164560400 Bertrandt
9 2021-03-01 123.75 127.93 122.79 127.79 127.0350 116307900 Bertrandt
10 2021-03-02 128.41 128.72 125.01 125.12 124.3807 102260900 Bertrandt
11 2021-03-03 124.81 125.71 121.84 122.06 121.3388 112966300 Bertrandt
12 2021-03-04 121.75 123.60 118.62 120.13 119.4202 178155000 Bertrandt
13 2021-03-05 120.98 121.94 117.57 121.42 120.7026 153766600 Bertrandt
14 2021-03-08 120.93 121.00 116.21 116.36 115.6725 154376600 Bertrandt
15 2021-03-09 119.03 122.06 118.79 121.09 120.3745 129525800 Bertrandt
16 2021-03-10 121.69 122.17 119.45 119.98 119.2711 111943300 Bertrandt
17 2021-03-11 122.54 123.21 121.26 121.96 121.2394 103026500 Bertrandt
18 2021-03-12 120.40 121.17 119.16 121.03 120.3149 88105100 Bertrandt
19 2021-03-15 121.41 124.00 120.42 123.99 123.2574 92403800 Bertrandt
20 2021-03-16 125.70 127.22 124.72 125.57 124.8281 115227900 Bertrandt
21 2021-03-17 124.05 125.86 122.34 124.76 124.0229 111932600 Bertrandt
22 2021-03-18 122.88 123.18 120.32 120.53 119.8179 121229700 Bertrandt
23 2021-03-19 119.90 121.43 119.68 119.99 119.2811 185549500 Bertrandt
24 2021-03-22 120.33 123.87 120.26 123.39 122.6610 111912300 Bertrandt
25 2021-03-23 123.33 124.24 122.14 122.54 121.8160 95467100 Bertrandt
26 2021-03-24 122.82 122.90 120.07 120.09 119.3805 88530500 Bertrandt
27 2021-03-25 119.54 121.66 119.00 120.59 119.8775 98844700 Bertrandt
28 2021-03-26 120.35 121.48 118.92 121.21 120.4938 94071200 Bertrandt

```

29 2021-03-29 121.65 122.58 120.73 121.39 120.6728 80819200 Bertrandt  
30 2021-03-30 120.11 120.40 118.86 119.90 119.1916 85671900 Bertrandt  
31 2021-03-31 121.65 123.52 121.15 122.15 121.4283 118323800 Bertrandt  
32 2021-04-01 123.66 124.18 122.49 123.00 122.2733 75089100 Bertrandt  
33 2021-04-05 123.87 126.16 123.07 125.90 125.1561 88651200 Bertrandt  
34 2021-04-06 126.50 127.13 125.65 126.21 125.4643 80171300 Bertrandt  
35 2021-04-07 125.83 127.92 125.14 127.90 127.1443 83466700 Bertrandt  
36 2021-04-08 128.95 130.39 128.52 130.36 129.5898 88844600 Bertrandt  
37 2021-04-09 129.80 133.04 129.47 133.00 132.2142 106686700 Bertrandt  
38 2021-04-12 132.52 132.85 130.63 131.24 130.4646 91420000 Bertrandt  
39 2021-04-13 132.44 134.66 131.93 134.43 133.6357 91266500 Bertrandt  
40 2021-04-14 134.94 135.00 131.66 132.03 131.2499 87222800 Bertrandt  
41 2021-04-15 133.82 135.00 133.64 134.50 133.7053 89347100 Bertrandt  
42 2021-04-16 134.30 134.67 133.28 134.16 133.3673 84922400 Bertrandt  
43 2021-04-19 133.51 135.47 133.34 134.84 134.0433 94264200 Bertrandt  
44 2021-04-20 135.02 135.53 131.81 133.11 132.3235 94812300 Bertrandt  
45 2021-04-21 132.36 133.75 131.30 133.50 132.7112 68847100 Bertrandt  
46 2021-04-22 133.04 134.15 131.41 131.94 131.1605 84566500 Bertrandt  
47 2021-04-23 132.16 135.12 132.16 134.32 133.5264 78657500 Bertrandt  
48 2021-04-26 134.83 135.06 133.56 134.72 133.9240 66905100 Bertrandt  
49 2021-04-27 135.01 135.41 134.11 134.39 133.5960 66015800 Bertrandt  
50 2021-04-28 134.31 135.02 133.08 133.58 132.7907 107760100 Bertrandt  
51 2021-04-29 136.47 137.07 132.45 133.48 132.6913 151101000 Bertrandt  
52 2021-04-30 131.78 133.56 131.07 131.46 130.6833 109839500 Bertrandt  
53 2021-05-03 132.04 134.07 131.83 132.54 131.7569 75135100 Bertrandt  
54 2021-05-04 131.19 131.49 126.70 127.85 127.0946 137564700 Bertrandt  
55 2021-05-05 129.20 130.45 127.97 128.10 127.3431 84000900 Bertrandt  
56 2021-05-06 127.89 129.75 127.13 129.74 128.9735 78128300 Bertrandt  
57 2021-05-07 130.85 131.26 129.48 130.21 129.6606 78973300 Bertrandt  
58 2021-05-10 129.41 129.54 126.81 126.85 126.3147 88071200 Bertrandt  
59 2021-05-11 123.50 126.27 122.77 125.91 125.3787 126142800 Bertrandt  
60 2021-05-12 123.40 124.64 122.25 122.77 122.2519 112172300 Bertrandt  
61 2021-05-13 124.58 126.15 124.26 124.97 124.4426 105861300 Bertrandt  
62 2021-05-14 126.25 127.89 125.85 127.45 126.9122 81918000 Bertrandt  
63 2021-05-17 126.82 126.93 125.17 126.27 125.7372 74244600 Bertrandt  
64 2021-05-18 126.56 126.99 124.78 124.85 124.3232 63342900 Bertrandt  
65 2021-05-19 123.16 124.92 122.86 124.69 124.1638 92612000 Bertrandt  
66 2021-05-20 125.23 127.72 125.10 127.31 126.7728 76857100 Bertrandt  
67 2021-05-21 127.82 128.00 125.21 125.43 124.9007 79295400 Bertrandt  
68 2021-05-24 126.01 127.94 125.94 127.10 126.5637 63092900 Bertrandt  
69 2021-05-25 127.82 128.32 126.32 126.90 126.3645 72009500 Bertrandt  
70 2021-05-26 126.96 127.39 126.42 126.85 126.3147 56575900 Bertrandt  
71 2021-05-27 126.44 127.64 125.08 125.28 124.7513 94625600 Bertrandt  
72 2021-05-28 125.57 125.80 124.55 124.61 124.0842 71311100 Bertrandt

73 2021-06-01 125.08 125.35 123.94 124.28 123.7556 67637100 Bertrandt  
74 2021-06-02 124.28 125.24 124.05 125.06 124.5323 59278900 Bertrandt  
75 2021-06-03 124.68 124.85 123.13 123.54 123.0187 76229200 Bertrandt  
76 2021-06-04 124.07 126.16 123.85 125.89 125.3588 75169300 Bertrandt  
77 2021-06-07 126.17 126.32 124.83 125.90 125.3687 71057600 Bertrandt  
78 2021-06-08 126.60 128.46 126.21 126.74 126.2052 74403800 Bertrandt  
79 2021-06-09 127.21 127.75 126.52 127.13 126.5935 56877900 Bertrandt  
80 2021-06-10 127.02 128.19 125.94 126.11 125.5778 71186400 Bertrandt  
81 2021-06-11 126.53 127.44 126.10 127.35 126.8126 53522400 Bertrandt  
82 2021-06-14 127.82 130.54 127.07 130.48 129.9294 96906500 Bertrandt  
83 2021-06-15 129.94 130.60 129.39 129.64 129.0929 62746300 Bertrandt  
84 2021-06-16 130.37 130.89 128.46 130.15 129.6008 91815000 Bertrandt  
85 2021-06-17 129.80 132.55 129.65 131.79 131.2339 96721700 Bertrandt  
86 2021-06-18 130.71 131.51 130.24 130.46 129.9095 108953300 Bertrandt  
87 2021-06-21 130.30 132.41 129.21 132.30 131.7417 79663300 Bertrandt  
88 2021-06-22 132.13 134.08 131.62 133.98 133.4146 74783600 Bertrandt  
89 2021-06-23 133.77 134.32 133.23 133.70 133.1358 60214200 Bertrandt  
90 2021-06-24 134.45 134.64 132.93 133.41 132.8470 68711000 Bertrandt  
91 2021-06-25 133.46 133.89 132.81 133.11 132.5483 70783700 Bertrandt  
92 2021-06-28 133.41 135.25 133.35 134.78 134.2113 62111300 Bertrandt  
93 2021-06-29 134.80 136.49 134.35 136.33 135.7547 64556100 Bertrandt  
94 2021-06-30 136.17 137.41 135.87 136.96 136.3821 63261400 Bertrandt  
95 2021-07-01 136.60 137.33 135.76 137.27 136.6908 52485800 Bertrandt  
96 2021-07-02 137.90 140.00 137.75 139.96 139.3694 78852600 Bertrandt  
97 2021-07-06 140.07 143.15 140.07 142.02 141.4207 108181800 Bertrandt  
98 2021-07-07 143.54 144.89 142.66 144.57 143.9599 104911600 Bertrandt  
99 2021-07-08 141.58 144.06 140.67 143.24 142.6355 105575500 Bertrandt  
100 2021-07-09 142.75 145.65 142.65 145.11 144.4977 99890800 Bertrandt  
101 2021-07-12 146.21 146.32 144.00 144.50 143.8902 76299700 Bertrandt  
102 2021-07-13 144.03 147.46 143.63 145.64 145.0254 100827100 Bertrandt  
103 2021-07-14 148.10 149.57 147.68 149.15 148.5206 127050800 Bertrandt  
104 2021-07-15 149.24 150.00 147.09 148.48 147.8534 106820300 Bertrandt  
105 2021-07-16 148.46 149.76 145.88 146.39 145.7722 93251400 Bertrandt  
106 2021-07-19 143.75 144.07 141.67 142.45 141.8489 121434600 Bertrandt  
107 2021-07-20 143.46 147.10 142.96 146.15 145.5332 96350000 Bertrandt  
108 2021-07-21 145.53 146.13 144.63 145.40 144.7864 74993500 Bertrandt  
109 2021-07-22 145.94 148.20 145.81 146.80 146.1805 77338200 Bertrandt  
110 2021-07-23 147.55 148.72 146.92 148.56 147.9331 71447400 Bertrandt  
111 2021-07-26 148.27 149.83 147.70 148.99 148.3613 72434100 Bertrandt  
112 2021-07-27 149.12 149.21 145.55 146.77 146.1507 104818600 Bertrandt  
113 2021-07-28 144.81 146.97 142.54 144.98 144.3682 118931200 Bertrandt  
114 2021-07-29 144.69 146.55 144.58 145.64 145.0254 56699500 Bertrandt  
115 2021-07-30 144.38 146.33 144.11 145.86 145.2445 70382000 Bertrandt  
116 2021-08-02 146.36 146.95 145.25 145.52 144.9059 62880000 Bertrandt

```

117 2021-08-03 145.81 148.04 145.18 147.36 146.7382 64786600 Bertrandt
118 2021-08-04 147.27 147.79 146.28 146.95 146.3299 56368300 Bertrandt
119 2021-08-05 146.98 147.84 146.17 147.06 146.4394 46397700 Bertrandt
120 2021-08-06 146.35 147.11 145.63 146.14 145.7413 54067400 Bertrandt
121 2021-08-09 146.20 146.70 145.52 146.09 145.6915 48908700 Bertrandt
122 2021-08-10 146.44 147.71 145.30 145.60 145.2028 69023100 Bertrandt
123 2021-08-11 146.05 146.72 145.53 145.86 145.4621 48493500 Bertrandt
124 2021-08-12 146.19 149.05 145.84 148.89 148.4838 72282600 Bertrandt
125 2021-08-13 148.97 149.44 148.27 149.10 148.6933 59318800 Bertrandt
[ reached 'max' /getOption("max.print") -- omitted 637 rows ]
> df_allStocks$Date <- as.character(df_allStocks$Date)
> datesplit_list <- strsplit(df_allStocks$Date, "-")
> df_dates <- lapply(datesplit_list)
> colnames(df_dates) <- c("Year", "Month", "Day")
> df_allStocks <- cbind(df_allStocks, df_dates)
> names(df_allStocks)
[1] "Date" "Open" "High" "Low" "Close" "Adj.Close" "Volume" "Stock" "Year"
[10] "Month" "Day"
> head(df_allStocks)
      Date Open High Low Close Adj.Close Volume Stock Year Month Day
1 2021-02-17 131.25 132.22 129.47 130.84 130.0669 97918500 Bertrandt 2021 02 17
2 2021-02-18 129.20 130.00 127.41 129.71 128.9436 96856700 Bertrandt 2021 02 18
3 2021-02-19 130.24 130.71 128.80 129.87 129.1027 87668800 Bertrandt 2021 02 19
4 2021-02-22 128.01 129.72 125.60 126.00 125.2555 103916400 Bertrandt 2021 02 22
5 2021-02-23 123.76 126.71 118.39 125.86 125.1164 158273000 Bertrandt 2021 02 23
6 2021-02-24 124.94 125.56 122.23 125.35 124.6094 111039900 Bertrandt 2021 02 24
> g <- ggplot(data=df_aapl, aes(x=Date, y=Open, group=1)) # group 1st param
> g <- g + geom_line(linetype="dashed")
> g
> g <- ggplot(data=df_aapl, aes(x=Date, y=Open, group=1)) # group 1st param
> g <- g + geom_line(linetype="dashed", col="red")
> g
> g <- ggplot(data=df_aapl, aes(x=Date, y=Open, group=1)) # group 1st param
> g <- g + geom_line(linetype="solid", col="red", size=1.5)
> g <- g + labs(title="Apple Inc", subtitle="Open Prices", y="Open", x="Year", caption="Yearwise Apple
Stock")
> g
> options(scipen = 999)
> ggplot(data=df_allStocks, aes(x=Stock, y=Volume)) +
+ geom_bar(stat="identity") #if we want heights of the bars to represent values in the data, map a
value to y aes
> #scipen - avoid scientific notations by giving largest limit eg. 999
> ggplot(data=df_allStocks, aes(x=Stock, y=Volume)) +
+ geom_bar(stat="identity") + coord_flip() #coord_flip to create horizontal plot

```



```

> ggplot(data=df_allStocks, aes(x=Stock, y=Volume)) +
+   geom_bar(stat="identity", width=0.5) #change width of bars
> ggplot(data=df_allStocks, aes(x=Stock, y=Volume)) +
+   geom_bar(stat="identity", width=0.5, col="blue")
> ggplot(data=df_allStocks, aes(x=Stock, y=Volume, fill=Stock)) +
+   geom_bar(stat="identity", width=0.5)
> #fill=Stock - fill colors automatically as per the levels of the bar
> ggplot(df_nike, aes(x=Open)) + geom_histogram()
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
> ggplot(df_waltdisney, aes(x=Open)) + geom_histogram()
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
> ggplot(df_nike, aes(x=Volume)) + geom_histogram(fill="lightblue", color="darkblue")
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
> ggplot(df_nike, aes(x=Close)) + geom_histogram(fill="lightblue", color="darkblue")
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
> ggplot(df_nike, aes(x=Close)) + geom_histogram(fill="lightblue", color="darkblue", binwidth=3)
> ggplot(df_nike, aes(x=Open)) +
+   geom_histogram(aes(y=..density..), fill="white", colour="black") +
+   geom_density(alpha=.2, fill="Turquoise") #alpha controls the transparency
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
> ggplot(df_nike, aes(x=Open, col=Stock)) + geom_histogram(fill="light blue", binwidth=3)
> ggplot(df_allStocks, aes(x=Open, col=Stock)) + geom_histogram(fill="light blue", binwidth=3)
#Different outline color for different stock category
> ggplot(df_waltdisney, aes(x=Open, y=Close)) + geom_point()
> ggplot(df_nike, aes(x=Open, y=Close)) + geom_point(size=2, shape=23) + geom_smooth(method="lm")
`geom_smooth()` using formula 'y ~ x'
> #size - size of point, shape - shape of point (0-25), method="lm" - draw linear model (linear regression)
line
> ggplot(df_nike, aes(x=Open, y=Close)) +
+   geom_point(shape=18, color="dark grey") +
+   geom_smooth(method="lm", linetype="dashed", color="red")
> df_midwest = read.csv("http://goo.gl/G1K41K")
> dim(df_midwest)
[1] 437 28
> summary(df_midwest)
  PID      county      state      area      poptotal      popdensity
Min. : 561 Length:437      Length:437      Min. :0.00500 Min. : 1701 Min. : 85.05
1st Qu.: 670 Class:character Class:character 1st Qu.:0.02400 1st Qu.: 18840 1st Qu.: 622.41
Median :1221 Mode :character Mode :character Median :0.03000 Median : 35324 Median :
1156.21
Mean :1437                      Mean :0.03317 Mean : 96130 Mean : 3097.74
3rd Qu.:2059                      3rd Qu.:0.03800 3rd Qu.: 75651 3rd Qu.: 2330.00
Max. :3052                      Max. :0.11000 Max. :5105067 Max. :88018.40
  popwhite  popblack  popamerindian  popasian  popother  percwhite

```

```

Min. : 416 Min. : 0 Min. : 4.0 Min. : 0 Min. : 0 Min. :10.69
1st Qu.: 18630 1st Qu.: 29 1st Qu.: 44.0 1st Qu.: 35 1st Qu.: 20 1st Qu.:94.89
Median : 34471 Median : 201 Median : 94.0 Median : 102 Median : 66 Median :98.03
Mean : 81840 Mean : 11024 Mean : 343.1 Mean : 1310 Mean : 1613 Mean :95.56
3rd Qu.: 72968 3rd Qu.: 1291 3rd Qu.: 288.0 3rd Qu.: 401 3rd Qu.: 345 3rd Qu.:99.07
Max. :3204947 Max. :1317147 Max. :10289.0 Max. :188565 Max. :384119 Max. :99.82
percblack percamerindan percasian percother popadults perchsd
Min. : 0.0000 Min. : 0.05623 Min. :0.0000 Min. :0.00000 Min. : 1287 Min. :46.91
1st Qu.: 0.1157 1st Qu.: 0.15793 1st Qu.:0.1737 1st Qu.:0.09102 1st Qu.: 12271 1st Qu.:71.33
Median : 0.5390 Median : 0.21502 Median :0.2972 Median :0.17844 Median : 22188 Median
:74.25
Mean : 2.6763 Mean : 0.79894 Mean :0.4872 Mean :0.47906 Mean : 60973 Mean :73.97
3rd Qu.: 2.6014 3rd Qu.: 0.38362 3rd Qu.:0.5212 3rd Qu.:0.48050 3rd Qu.: 47541 3rd Qu.:77.20
Max. :40.2100 Max. :89.17738 Max. :5.0705 Max. :7.52427 Max. :3291995 Max. :88.90
percollege percprof poppovertyknown percpovertyknown percbelowpoverty
percchilbelowpovert
Min. : 7.336 Min. : 0.5203 Min. : 1696 Min. :80.90 Min. : 2.180 Min. : 1.919
1st Qu.:14.114 1st Qu.: 2.9980 1st Qu.: 18364 1st Qu.:96.89 1st Qu.: 9.199 1st Qu.:11.624
Median :16.798 Median : 3.8142 Median : 33788 Median :98.17 Median :11.822 Median :15.270
Mean :18.273 Mean : 4.4473 Mean : 93642 Mean :97.11 Mean :12.511 Mean :16.447
3rd Qu.:20.550 3rd Qu.: 4.9493 3rd Qu.: 72840 3rd Qu.:98.60 3rd Qu.:15.133 3rd Qu.:20.352
Max. :48.079 Max. :20.7913 Max. :5023523 Max. :99.86 Max. :48.691 Max. :64.308
percadultpoverty percelderlypoverty inmetro category
Min. : 1.938 Min. : 3.547 Min. :0.0000 Length:437
1st Qu.: 7.668 1st Qu.: 8.912 1st Qu.:0.0000 Class :character
Median :10.008 Median :10.869 Median :0.0000 Mode :character
Mean :10.919 Mean :11.389 Mean :0.3432
3rd Qu.:13.182 3rd Qu.:13.412 3rd Qu.:1.0000
Max. :43.312 Max. :31.162 Max. :1.0000
> ggplot(df_midwest, aes(x=area, y=poptotal)) +
+ geom_point(shape=18, color="dark grey") +
+ geom_smooth(method="lm", linetype="dashed", color="red")
`geom_smooth()` using formula 'y ~ x'
> ggplot(df_midwest, aes(x=area, y=poptotal)) + geom_point(shape=18, color="dark
grey")+geom_smooth(method="lm", linetype="dashed", color="red") + coord_cartesian(xlim=c(0,0.1),
ylim=c(0,600000))
`geom_smooth()` using formula 'y ~ x'
> seq(1, 20, 3)
[1] 1 4 7 10 13 16 19
> g <- ggplot(df_midwest, aes(x=area, y=poptotal)) +
+ geom_point(size=2) +
+ geom_smooth(method="lm", col="black") +
+ coord_cartesian(xlim=c(0,0.1), ylim=c(0,1000000)) +

```

```

+ labs(title="Area Vs Population", subtitle = "Using midwest dataset", y="Population", x="area", caption
= "Midwest Demographics")
> g + scale_x_continuous(breaks=seq(0, 0.10, 0.01))
`geom_smooth()` using formula 'y ~ x'
> g + scale_y_continuous(breaks=seq(0, 1000000, 50000))
`geom_smooth()` using formula 'y ~ x'
> g <- ggplot(df_midwest, aes(x=area, y=poptotal)) +
+ geom_point(aes(color=state), size=2) +
+ geom_smooth(method="lm", col="black") +
+ coord_cartesian(xlim=c(0,0.1), ylim=c(0,1000000)) +
+ labs(title="Area Vs Population", subtitle = "Using midwest dataset", y="Population", x="area", caption
= "Midwest Demographics")
> g + scale_x_continuous(breaks=seq(0, 0.10, 0.01))
`geom_smooth()` using formula 'y ~ x'
> g + scale_y_continuous(breaks=seq(0, 1000000, 50000))
> ggplot(df_allStocks, aes(x=Month, y=Close)) + geom_boxplot()
> ggplot(df_allStocks, aes(x=Month, y=Close)) + geom_boxplot() + coord_flip()
> ggplot(df_allStocks, aes(x=Month, y=Close, color=Month)) + geom_boxplot() + coord_flip()
> ggplot(df_midwest, aes(x=state, y=poptotal)) + geom_boxplot(outlier.color = "red", outlier.shape = 1,
outlier.size = 2)
> ggplot(df_allStocks, aes(x=Year, y=Close)) + geom_boxplot() + facet_grid(~ Stock)
> ggplot(df_allStocks, aes(x=Month, y=Close)) + geom_boxplot() + facet_grid(Stock ~ Year)
> ggplot(df_allStocks, aes(x=Open)) +
+ geom_histogram(color="black", fill="white") +
+ facet_grid(Stock ~ .)
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
> ggplot(df_allStocks, aes(x=Open, color=Stock)) +
+ geom_histogram(fill="white") +
+ facet_grid(Stock ~ .)
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
> ggplot(df_allStocks, aes(x=Close, color=Stock)) +
+ geom_histogram(fill="white") +
+ facet_grid(Stock ~ ., scales="free_y")

```