



VIT[®]
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

LAB ASSIGNMENT – 2

Data Manipulation using R

Name: Tejas Rahul Rokade

Reg. No.: 20BDS0033

Lab Slot: L33 & L34

1. Create user defined functions to perform various arithmetic operations and call each functions using menu driven format (try functions with and without parameters, functions with default argument)
2. Familiarize basic statistical operation on a random vector of 100 elements
 - a. Mean
 - b. Median
 - c. Mode
 - d. Range
 - e. IQR
 - f. Standard deviation
 - g. Summary
 - h. Histogram
 - i. Table
3. Perform given operations on a data frame
 - a. Create a data frame
 - b. Access a component ([, [[, \$)
 - c. Structure of data frame
 - d. Add new column
 - e. Add new row
 - f. Delete column
 - g. Delete specific row
 - h. Order data frame (with, order, arrange)
4. Read Air quality dataset and handle the missing data using following technique
 - a. Drop Row
 - b. Drop Column
 - c. Imputation (Replace with unknown, mean or Group mean)

1. Create user defined functions to perform various arithmetic operations and call each functions using menu driven format (try functions with and without parameters, functions with default argument)

Code:

```
add = function(x, y) {
    return(x + y)
}
subtract = function(x, y) {
    return(x - y)
}
multiply = function(x, y) {
    return(x * y)
}
divide = function(x, y) {
    return(x / y)
}
print("Select operation.") print("1.Add") print("2.Subtract")
print("3.Multiply") print("4.Divide") choice =
as.integer(readline(prompt="Enter choice[1/2/3/4]: ")) num1 =
as.integer(readline(prompt="Enter first number: ")) num2 =
as.integer(readline(prompt="Enter second number: "))
operator = switch(choice,"+","-","*","/")
result = switch(choice, add(num1, num2), subtract(num1, num2),
multiply(num1, num2), divide(num1, num2))
print(paste(num1, operator, num2, "=", result))
```

Output:

```
> add = function(x, y) {
+   return(x + y)
+ }
> subtract = function(x, y) {
+   return(x - y)
+ }
> multiply = function(x, y) {
+   return(x * y)
+ }
> divide = function(x, y) {
+   return(x / y)
+ }
> print("Select operation.")
[1] "Select operation."
> print("1.Add")
[1] "1.Add"
> print("2.Subtract")
[1] "2.Subtract"
> print("3.Multiply")
[1] "3.Multiply"
> print("4.Divide")
[1] "4.Divide"
> choice = as.integer(readline(prompt="Enter choice[1/2/3/4]: "))
Enter choice[1/2/3/4]: 4
> num1 = as.integer(readline(prompt="Enter first number: "))
Enter first number: 3
> num2 = as.integer(readline(prompt="Enter second number: "))
Enter second number: 12
> operator = switch(choice,"+","-","*","/")
> result = switch(choice, add(num1, num2), subtract(num1, num2), multiply(num1, num2), divide(num1, num2))
> print(paste(num1, operator, num2, "=", result))
[1] "3 / 12 = 0.25"
```

2. Familiarize basic statistical operation on a random vector of 100 elements

- a.* Mean
- b.* Median
- c.* Mode
- d.* Range
- e.* IQR
- f.* Standard Deviation
- g.* Summary
- h.* Histogram
- i.* Table

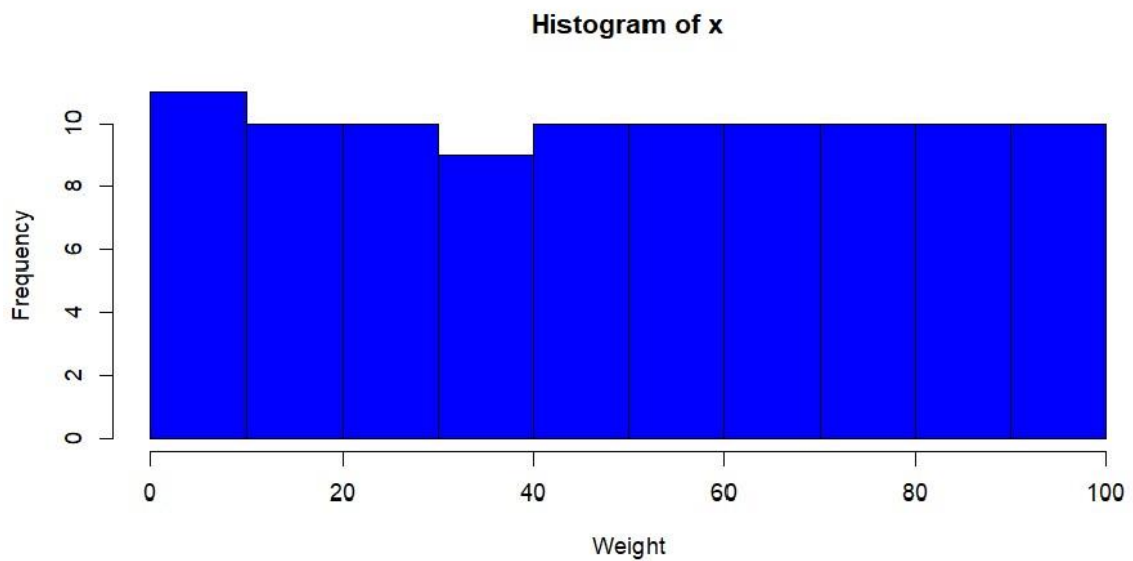
Code:

```
set.seed(123)
x = sample(0:100, size = 100)
x
mean(x) median(x)
getmode = function(v) {
  uniqx = unique(x)
  uniqx[which.max(tabulate(match(x, uniqx)))]
} result =
getmode(x)
print(result)
max(x)-min(x) #range
IQR(x) sd(x)
summary(x)
hist(x,xlab = "Weight",col = "blue",border = "black") table(x)
```

Output:

```
> set.seed(123)
> x = sample(0:100, size = 100)
> x
 [1] 30 78 50 13 66 41 49 42 97 24 89 68 56 8 71 25 6 95 87 35 77 93 75 14 31
84 82 40 22 26 59 52 99
[34] 69 86 37 33 28 4 7 11 12 17 32 88 96 91 65 20 80 73 46 85 100 15 29 5 79
61 21 83 38 47 16 64 3
[67] 70 62 54 19 98 57 92 45 72 2 43 94 60 63 48 39 90 67 51 36 23 27 58 9 1
55 10 74 44 81 18 76 0
[100] 53
> mean(x)
[1] 50.16
> median(x)
[1] 50.5
> getmode = function(v) {
+   uniqx = unique(x)
+   uniqx[which.max(tabulate(match(x, uniqx)))]
+ }
> result = getmode(x)
> print(result)
[1] 30
> max(x)-min(x) #range
[1] 100
> IQR(x)
[1] 50.5
> sd(x)
[1] 29.4034
> summary(x)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   0.00  24.75   50.50   50.16   75.25  100.00
```

```
> hist(x,xlab = "Weight",col = "blue",border = "black")
```



```
> table(x)
x
 0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25
1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
26 27 28 29 30 31 32 33 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52
1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78
1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100
1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
```

3. Perform given operations on a data frame

- a. Create a data frame
- b. Access a component ([, [[, \$)
- c. Structure of data frame
- d. Add new column
- e. Add new row
- f. Delete column
- g. Delete specific row
- h. Order data frame (with, order, arrange)

Code:

```
df = data.frame("animal"=c('koala', 'hedgehog', 'sloth',
'panda'),"country"=c('Australia', 'Italy', 'Peru', 'China'),"avg_sleep_hours"=c(21,
18, 17, 10)) print(df) df[["animal"]] df[["animal"]]
```

```
df$animal str(df)
```

```
cbind(df,color=c("black","grey","red","white"))
```

```

rbind(df,list("penguin","Antarctica",12,"white"))
) df$color=NULL
df new_df=df[-
c(2), ] new_df
order(df)
arrange(df)

```

Output:

```

> df = data.frame("animal"=c('koala', 'hedgehog', 'sloth', 'panda'),"country"=c('Australia', 'Italy', 'Peru', 'China'),"avg_sleep_hours"=c(21,
18, 17, 10))
> print(df)
  animal country avg_sleep_hours
1 koala  Australia             21
2 hedgehog    Italy             18
3 sloth      Peru              17
4 panda     China              10
> df[["animal"]]
  animal
1 koala
2 hedgehog
3 sloth
4 panda
> df[["animal"]]
[1] "koala" "hedgehog" "sloth" "panda"
> df$animal
[1] "koala" "hedgehog" "sloth" "panda"
> str(df)
'data.frame': 4 obs. of 3 variables:
 $ animal : chr "koala" "hedgehog" "sloth" "panda"
 $ country : chr "Australia" "Italy" "Peru" "China"
 $ avg_sleep_hours: num 21 18 17 10
> cbind(df,color=c("black","grey","red","white"))
  animal country avg_sleep_hours color
1 koala  Australia             21 black
2 hedgehog    Italy             18 grey
3 sloth      Peru              17 red
4 panda     China              10 white
> rbind(df,list("penguin","Antarctica",12,"white"))
  animal country avg_sleep_hours
1 koala  Australia             21
2 hedgehog    Italy             18
3 sloth      Peru              17
4 panda     China              10
5 penguin  Antarctica           12
> df$color=NULL
> df
  animal country avg_sleep_hours
1 koala  Australia             21
2 hedgehog    Italy             18
3 sloth      Peru              17
4 panda     China              10

```

```

> new_df=df[-c(2), ]
> new_df
  animal country avg_sleep_hours
1 koala  Australia             21
3 sloth      Peru              17
4 panda     China              10
> order(df)
[1] 12 11 10 9 5 8 2 6 1 4 7 3
> arrange(df)
  animal country avg_sleep_hours
1 koala  Australia             21
2 hedgehog    Italy             18
3 sloth      Peru              17
4 panda     China              10
> |

```

4. Read Air quality dataset and handle the missing data using following technique
 - a. Drop Row
 - b. Drop Column
 - c. Imputation (Replace with unknown, mean or Group mean)

Code:

```
df=datasets::airquality
df$y=na.omit(df$y)
colSums(is.na(df))==0
x=is.na(df) df[x]=0 df
```

Output:

```
> df=datasets::airquality
> df
  Ozone Solar.R Wind Temp Month Day
1     41    190   7.4   67     5     1
2     36    118   8.0   72     5     2
3     12    149  12.6   74     5     3
4     18    313  11.5   62     5     4
5     NA     NA  14.3   56     5     5
6     28     NA  14.9   66     5     6
7     23    299   8.6   65     5     7
8     19     99  13.8   59     5     8
9     18     19  20.1   61     5     9
10    NA    194   8.6   69     5    10
11     7     NA   6.9   74     5    11
12    16    256   9.7   69     5    12
13    11    290   9.2   66     5     1
14    14    274  10.9   68     5     2
15    18     65  13.2   58     5     3
16    14    334  11.5   64     5     4
17    34    307  12.0   66     5     5
18     6     78  18.4   97     5     6
19    30    322  11.5   66     5     7
20    11     44   9.7   62     5     8
21     1     32   9.7   59     5     9
22    11    320  16.6   73     5    10
23     4     35   9.7   61     5    11
24    32     92  12.0   61     5    12
25    NA    666  16.6   57     5     1
26    NA    266  14.9   58     5     2
27    NA    274  15.0   57     5     3
28    23    120  12.0   67     5     4
29    45    252  14.9   81     5     5
30   115    223   5.7   79     5     6
31    37    227   7.4   76     5     7
32    NA    286   8.6   78     5     8
33    NA    287   9.7   74     5     9
34    NA    242  16.1   67     5    10
35    NA    166   9.2   84     5    11
36    NA    220  18.6   85     5    12
37    NA    164  14.3   79     5     1
38    29   1127   9.7   89     5     2
39    NA    273   6.9   87     5     3
40    71    291  13.8   90     5     4
41    39    323  11.5   87     5     5
42    NA    159  10.9   93     5     6
43    NA    250   9.2   92     5     7
44    23    148   8.0   82     5     8
45    NA    322  13.8   80     5     9
46    NA    322  11.5   79     5    10
47    21    191  14.9   77     5    11
48    37    264  20.7   72     5    12
49    20     37   9.2   65     5     1
50    12    120  11.5   73     5     2
51    13    137  10.3   76     5     3
52    NA    150   6.3   77     5     4
53    NA     59   1.7   76     5     5
54    NA     91   4.6   76     5     6
55    NA    250   6.3   76     5     7
56    NA    135   8.0   79     5     8
57    NA    127   6.0   78     5     9
58    NA     47  10.3   73     5    10
59    NA     98  11.5   80     5    11
60    NA     31  14.9   77     5    12
61    NA    138   8.0   83     5     1
62    13     41   4.1   84     5     2
63    49    248   9.2   85     5     3
64    32    236   9.2   81     5     4
65    NA    101  10.9   84     5     5
66    64    175   4.6   83     5     6
67    40    314  10.9   83     5     7
68    77    276   5.1   88     5     8
69    97    267   6.3   92     5     9
70    97    272   5.7   92     5    10
```

```

y = na.omit(df)
y
  Ozone Solar.Wind Temp Month Day
1 41.6 11.1 11.1 1 1
2 33.4 11.1 11.1 1 2
3 41.6 11.1 11.1 1 3
4 44.1 11.1 11.1 1 4
5 41.6 11.1 11.1 1 5
6 44.1 11.1 11.1 1 6
7 41.6 11.1 11.1 1 7
8 44.1 11.1 11.1 1 8
9 41.6 11.1 11.1 1 9
10 44.1 11.1 11.1 1 10
11 41.6 11.1 11.1 1 11
12 44.1 11.1 11.1 1 12
13 41.6 11.1 11.1 2 1
14 44.1 11.1 11.1 2 2
15 41.6 11.1 11.1 2 3
16 44.1 11.1 11.1 2 4
17 41.6 11.1 11.1 2 5
18 44.1 11.1 11.1 2 6
19 41.6 11.1 11.1 2 7
20 44.1 11.1 11.1 2 8
21 41.6 11.1 11.1 2 9
22 44.1 11.1 11.1 2 10
23 41.6 11.1 11.1 2 11
24 44.1 11.1 11.1 2 12
25 41.6 11.1 11.1 3 1
26 44.1 11.1 11.1 3 2
27 41.6 11.1 11.1 3 3
28 44.1 11.1 11.1 3 4
29 41.6 11.1 11.1 3 5
30 44.1 11.1 11.1 3 6
31 41.6 11.1 11.1 3 7
32 44.1 11.1 11.1 3 8
33 41.6 11.1 11.1 3 9
34 44.1 11.1 11.1 3 10
35 41.6 11.1 11.1 3 11
36 44.1 11.1 11.1 3 12
37 41.6 11.1 11.1 4 1
38 44.1 11.1 11.1 4 2
39 41.6 11.1 11.1 4 3
40 44.1 11.1 11.1 4 4
41 41.6 11.1 11.1 4 5
42 44.1 11.1 11.1 4 6
43 41.6 11.1 11.1 4 7
44 44.1 11.1 11.1 4 8
45 41.6 11.1 11.1 4 9
46 44.1 11.1 11.1 4 10
47 41.6 11.1 11.1 4 11
48 44.1 11.1 11.1 4 12
49 41.6 11.1 11.1 5 1
50 44.1 11.1 11.1 5 2
51 41.6 11.1 11.1 5 3
52 44.1 11.1 11.1 5 4
53 41.6 11.1 11.1 5 5
54 44.1 11.1 11.1 5 6
55 41.6 11.1 11.1 5 7
56 44.1 11.1 11.1 5 8
57 41.6 11.1 11.1 5 9
58 44.1 11.1 11.1 5 10
59 41.6 11.1 11.1 5 11
60 44.1 11.1 11.1 5 12
61 41.6 11.1 11.1 6 1
62 44.1 11.1 11.1 6 2
63 41.6 11.1 11.1 6 3
64 44.1 11.1 11.1 6 4
65 41.6 11.1 11.1 6 5
66 44.1 11.1 11.1 6 6
67 41.6 11.1 11.1 6 7
68 44.1 11.1 11.1 6 8
69 41.6 11.1 11.1 6 9
70 44.1 11.1 11.1 6 10
71 41.6 11.1 11.1 6 11
72 44.1 11.1 11.1 6 12
73 41.6 11.1 11.1 7 1
74 44.1 11.1 11.1 7 2
75 41.6 11.1 11.1 7 3
76 44.1 11.1 11.1 7 4
77 41.6 11.1 11.1 7 5
78 44.1 11.1 11.1 7 6
79 41.6 11.1 11.1 7 7
80 44.1 11.1 11.1 7 8
81 41.6 11.1 11.1 7 9
82 44.1 11.1 11.1 7 10
83 41.6 11.1 11.1 7 11
84 44.1 11.1 11.1 7 12
85 41.6 11.1 11.1 8 1
86 44.1 11.1 11.1 8 2
87 41.6 11.1 11.1 8 3
88 44.1 11.1 11.1 8 4
89 41.6 11.1 11.1 8 5
90 44.1 11.1 11.1 8 6
91 41.6 11.1 11.1 8 7
92 44.1 11.1 11.1 8 8
93 41.6 11.1 11.1 8 9
94 44.1 11.1 11.1 8 10
95 41.6 11.1 11.1 8 11
96 44.1 11.1 11.1 8 12
97 41.6 11.1 11.1 9 1
98 44.1 11.1 11.1 9 2
99 41.6 11.1 11.1 9 3
100 44.1 11.1 11.1 9 4
101 41.6 11.1 11.1 9 5
102 44.1 11.1 11.1 9 6
103 41.6 11.1 11.1 9 7
104 44.1 11.1 11.1 9 8
105 41.6 11.1 11.1 9 9
106 44.1 11.1 11.1 9 10
107 41.6 11.1 11.1 9 11
108 44.1 11.1 11.1 9 12
109 41.6 11.1 11.1 10 1
110 44.1 11.1 11.1 10 2
111 41.6 11.1 11.1 10 3
112 44.1 11.1 11.1 10 4
113 41.6 11.1 11.1 10 5
114 44.1 11.1 11.1 10 6
115 41.6 11.1 11.1 10 7
116 44.1 11.1 11.1 10 8
117 41.6 11.1 11.1 10 9
118 44.1 11.1 11.1 10 10
119 41.6 11.1 11.1 10 11
120 44.1 11.1 11.1 10 12
121 41.6 11.1 11.1 11 1
122 44.1 11.1 11.1 11 2
123 41.6 11.1 11.1 11 3
124 44.1 11.1 11.1 11 4
125 41.6 11.1 11.1 11 5
126 44.1 11.1 11.1 11 6
127 41.6 11.1 11.1 11 7
128 44.1 11.1 11.1 11 8
129 41.6 11.1 11.1 11 9
130 44.1 11.1 11.1 11 10
131 41.6 11.1 11.1 11 11
132 44.1 11.1 11.1 11 12
133 41.6 11.1 11.1 12 1
134 44.1 11.1 11.1 12 2
135 41.6 11.1 11.1 12 3
136 44.1 11.1 11.1 12 4
137 41.6 11.1 11.1 12 5
138 44.1 11.1 11.1 12 6
139 41.6 11.1 11.1 12 7
140 44.1 11.1 11.1 12 8
141 41.6 11.1 11.1 12 9
142 44.1 11.1 11.1 12 10
143 41.6 11.1 11.1 12 11
144 44.1 11.1 11.1 12 12
145 41.6 11.1 11.1 1 1
146 44.1 11.1 11.1 1 2
147 41.6 11.1 11.1 1 3
148 44.1 11.1 11.1 1 4
149 41.6 11.1 11.1 1 5
150 44.1 11.1 11.1 1 6
151 41.6 11.1 11.1 1 7
152 44.1 11.1 11.1 1 8
153 41.6 11.1 11.1 1 9
154 44.1 11.1 11.1 1 10
155 41.6 11.1 11.1 1 11
156 44.1 11.1 11.1 1 12
157 41.6 11.1 11.1 2 1
158 44.1 11.1 11.1 2 2
159 41.6 11.1 11.1 2 3
160 44.1 11.1 11.1 2 4
161 41.6 11.1 11.1 2 5
162 44.1 11.1 11.1 2 6
163 41.6 11.1 11.1 2 7
164 44.1 11.1 11.1 2 8
165 41.6 11.1 11.1 2 9
166 44.1 11.1 11.1 2 10
167 41.6 11.1 11.1 2 11
168 44.1 11.1 11.1 2 12
169 41.6 11.1 11.1 3 1
170 44.1 11.1 11.1 3 2
171 41.6 11.1 11.1 3 3
172 44.1 11.1 11.1 3 4
173 41.6 11.1 11.1 3 5
174 44.1 11.1 11.1 3 6
175 41.6 11.1 11.1 3 7
176 44.1 11.1 11.1 3 8
177 41.6 11.1 11.1 3 9
178 44.1 11.1 11.1 3 10
179 41.6 11.1 11.1 3 11
180 44.1 11.1 11.1 3 12
181 41.6 11.1 11.1 4 1
182 44.1 11.1 11.1 4 2
183 41.6 11.1 11.1 4 3
184 44.1 11.1 11.1 4 4
185 41.6 11.1 11.1 4 5
186 44.1 11.1 11.1 4 6
187 41.6 11.1 11.1 4 7
188 44.1 11.1 11.1 4 8
189 41.6 11.1 11.1 4 9
190 44.1 11.1 11.1 4 10
191 41.6 11.1 11.1 4 11
192 44.1 11.1 11.1 4 12
193 41.6 11.1 11.1 5 1
194 44.1 11.1 11.1 5 2
195 41.6 11.1 11.1 5 3
196 44.1 11.1 11.1 5 4
197 41.6 11.1 11.1 5 5
198 44.1 11.1 11.1 5 6
199 41.6 11.1 11.1 5 7
200 44.1 11.1 11.1 5 8
201 41.6 11.1 11.1 5 9
202 44.1 11.1 11.1 5 10
203 41.6 11.1 11.1 5 11
204 44.1 11.1 11.1 5 12
205 41.6 11.1 11.1 6 1
206 44.1 11.1 11.1 6 2
207 41.6 11.1 11.1 6 3
208 44.1 11.1 11.1 6 4
209 41.6 11.1 11.1 6 5
210 44.1 11.1 11.1 6 6
211 41.6 11.1 11.1 6 7
212 44.1 11.1 11.1 6 8
213 41.6 11.1 11.1 6 9
214 44.1 11.1 11.1 6 10
215 41.6 11.1 11.1 6 11
216 44.1 11.1 11.1 6 12
217 41.6 11.1 11.1 7 1
218 44.1 11.1 11.1 7 2
219 41.6 11.1 11.1 7 3
220 44.1 11.1 11.1 7 4
221 41.6 11.1 11.1 7 5
222 44.1 11.1 11.1 7 6
223 41.6 11.1 11.1 7 7
224 44.1 11.1 11.1 7 8
225 41.6 11.1 11.1 7 9
226 44.1 11.1 11.1 7 10
227 41.6 11.1 11.1 7 11
228 44.1 11.1 11.1 7 12
229 41.6 11.1 11.1 8 1
230 44.1 11.1 11.1 8 2
231 41.6 11.1 11.1 8 3
232 44.1 11.1 11.1 8 4
233 41.6 11.1 11.1 8 5
234 44.1 11.1 11.1 8 6
235 41.6 11.1 11.1 8 7
236 44.1 11.1 11.1 8 8
237 41.6 11.1 11.1 8 9
238 44.1 11.1 11.1 8 10
239 41.6 11.1 11.1 8 11
240 44.1 11.1 11.1 8 12
241 41.6 11.1 11.1 9 1
242 44.1 11.1 11.1 9 2
243 41.6 11.1 11.1 9 3
244 44.1 11.1 11.1 9 4
245 41.6 11.1 11.1 9 5
246 44.1 11.1 11.1 9 6
247 41.6 11.1 11.1 9 7
248 44.1 11.1 11.1 9 8
249 41.6 11.1 11.1 9 9
250 44.1 11.1 11.1 9 10
251 41.6 11.1 11.1 9 11
252 44.1 11.1 11.1 9 12
253 41.6 11.1 11.1 10 1
254 44.1 11.1 11.1 10 2
255 41.6 11.1 11.1 10 3
256 44.1 11.1 11.1 10 4
257 41.6 11.1 11.1 10 5
258 44.1 11.1 11.1 10 6
259 41.6 11.1 11.1 10 7
260 44.1 11.1 11.1 10 8
261 41.6 11.1 11.1 10 9
262 44.1 11.1 11.1 10 10
263 41.6 11.1 11.1 10 11
264 44.1 11.1 11.1 10 12
265 41.6 11.1 11.1 11 1
266 44.1 11.1 11.1 11 2
267 41.6 11.1 11.1 11 3
268 44.1 11.1 11.1 11 4
269 41.6 11.1 11.1 11 5
270 44.1 11.1 11.1 11 6
271 41.6 11.1 11.1 11 7
272 44.1 11.1 11.1 11 8
273 41.6 11.1 11.1 11 9
274 44.1 11.1 11.1 11 10
275 41.6 11.1 11.1 11 11
276 44.1 11.1 11.1 11 12
277 41.6 11.1 11.1 12 1
278 44.1 11.1 11.1 12 2
279 41.6 11.1 11.1 12 3
280 44.1 11.1 11.1 12 4
281 41.6 11.1 11.1 12 5
282 44.1 11.1 11.1 12 6
283 41.6 11.1 11.1 12 7
284 44.1 11.1 11.1 12 8
285 41.6 11.1 11.1 12 9
286 44.1 11.1 11.1 12 10
287 41.6 11.1 11.1 12 11
288 44.1 11.1 11.1 12 12
289 41.6 11.1 11.1 1 1
290 44.1 11.1 11.1 1 2
291 41.6 11.1 11.1 1 3
292 44.1 11.1 11.1 1 4
293 41.6 11.1 11.1 1 5
294 44.1 11.1 11.1 1 6
295 41.6 11.1 11.1 1 7
296 44.1 11.1 11.1 1 8
297 41.6 11.1 11.1 1 9
298 44.1 11.1 11.1 1 10
299 41.6 11.1 11.1 1 11
300 44.1 11.1 11.1 1 12
301 41.6 11.1 11.1 2 1
302 44.1 11.1 11.1 2 2
303 41.6 11.1 11.1 2 3
304 44.1 11.1 11.1 2 4
305 41.6 11.1 11.1 2 5
306 44.1 11.1 11.1 2 6
307 41.6 11.1 11.1 2 7
308 44.1 11.1 11.1 2 8
309 41.6 11.1 11.1 2 9
310 44.1 11.1 11.1 2 10
311 41.6 11.1 11.1 2 11
312 44.1 11.1 11.1 2 12
313 41.6 11.1 11.1 3 1
314 44.1 11.1 11.1 3 2
315 41.6 11.1 11.1 3 3
316 44.1 11.1 11.1 3 4
317 41.6 11.1 11.1 3 5
318 44.1 11.1 11.1 3 6
319 41.6 11.1 11.1 3 7
320 44.1 11.1 11.1 3 8
321 41.6 11.1 11.1 3 9
322 44.1 11.1 11.1 3 10
323 41.6 11.1 11.1 3 11
324 44.1 11.1 11.1 3 12
325 41.6 11.1 11.1 4 1
326 44.1 11.1 11.1 4 2
327 41.6 11.1 11.1 4 3
328 44.1 11.1 11.1 4 4
329 41.6 11.1 11.1 4 5
330 44.1 11.1 11.1 4 6
331 41.6 11.1 11.1 4 7
332 44.1 11.1 11.1 4 8
333 41.6 11.1 11.1 4 9
334 44.1 11.1 11.1 4 10
335 41.6 11.1 11.1 4 11
336 44.1 11.1 11.1 4 12
337 41.6 11.1 11.1 5 1
338 44.1 11.1 11.1 5 2
339 41.6 11.1 11.1 5 3
340 44.1 11.1 11.1 5 4
341 41.6 11.1 11.1 5 5
342 44.1 11.1 11.1 5 6
343 41.6 11.1 11.1 5 7
344 44.1 11.1 11.1 5 8
345 41.6 11.1 11.1 5 9
346 44.1 11.1 11.1 5 10
347 41.6 11.1 11.1 5 11
348 44.1 11.1 11.1 5 12
349 41.6 11.1 11.1 6 1
350 44.1 11.1 11.1 6 2
351 41.6 11.1 11.1 6 3
352 44.1 11.1 11.1 6 4
353 41.6 11.1 11.1 6 5
354 44.1 11.1 11.1 6 6
355 41.6 11.1 11.1 6 7
356 44.1 11.1 11.1 6 8
357 41.6 11.1 11.1 6 9
358 44.1 11.1 11.1 6 10
359 41.6 11.1 11.1 6 11
360 44.1 11.1 11.1 6 12
361 41.6 11.1 11.1 7 1
362 44.1 11.1 11.1 7 2
363 41.6 11.1 11.1 7 3
364 44.1 11.1 11.1 7 4
365 41.6 11.1 11.1 7 5
366 44.1 11.1 11.1 7 6
367 41.6 11.1 11.1 7 7
368 44.1 11.1 11.1 7 8
369 41.6 11.1 11.1 7 9
370 44.1 11.1 11.1 7 10
371 41.6 11.1 11.1 7 11
372 44.1 11.1 11.1 7 12
373 41.6 11.1 11.1 8 1
374 44.1 11.1 11.1 8 2
375 41.6 11.1 11.1 8 3
376 44.1 11.1 11.1 8 4
377 41.6 11.1 11.1 8 5
378 44.1 11.1 11.1 8 6
379 41.6 11.1 11.1 8 7
380 44.1 11.1 11.1 8 8
381 41.6 11.1 11.1 8 9
382 44.1 11.1 11.1 8 10
383 41.6 11.1 11.1 8 11
384 44.1 11.1 11.1 8 12
385 41.6 11.1 11.1 9 1
386 44.1 11.1 11.1 9 2
387 41.6 11.1 11.1 9 3
388 44.1 11.1 11.1 9 4
389 41.6 11.1 11.1 9 5
390 44.1 11.1 11.1 9 6
391 41.6 11.1 11.1 9 7
392 44.1 11.1 11.1 9 8
393 41.6 11.1 11.1 9 9
394 44.1 11.1 11.1 9 10
395 41.6 11.1 11.1 9 11
396 44.1 11.1 11.1 9 12
397 41.6 11.1 11.1 10 1
398 44.1 11.1 11.1 10 2
399 41.6 11.1 11.1 10 3
400 44.1 11.1 11.1 10 4
401 41.6 11.1 11.1 10 5
402 44.1 11.1 11.1 10 6
403 41.6 11.1 11.1 10 7
404 44.1 11.1 11.1 10 8
405 41.6 11.1 11.1 10 9
406 44.1 11.1 11.1 10 10
407 41.6 11.1 11.1 10 11
408 44.1 11.1 11.1 10 12
409 41.6 11.1 11.1 11 1
410 44.1 11.1 11.1 11 2
411 41.6 11.1 11.1 11 3
412 44.1 11.1 11.1 11 4
413 41.6 11.1 11.1 11 5
414 44.1 11.1 11.1 11 6
415 41.6 11.1 11.1 11 7
416 44.1 11.1 11.1 11 8
417 41.6 11.1 11.1 11 9
418 44.1 11.1 11.1 11 10
419 41.6 11.1 11.1 11 11
420 44.1 11.1 11.1 11 12
421 41.6 11.1 11.1 12 1
422 44.1 11.1 11.1 12 2
423 41.6 11.1 11.1 12 3
424 44.1 11.1 11.1 12 4
425 41.6 11.1 11.1 12 5
426 44.1 11.1 11.1 12 6
427 41.6 11.1 11.1 12 7
428 44.1 11.1 11.1 12 8
429 41.6 11.1 11.1 12 9
430 44.1 11.1 11.1 12 10
431 41.6 11.1 11.1 12 11
432 44.1 11.1 11.1 12 12
433 41.6 11.1 11.1 1 1
434 44.1 11.1 11.1 1 2
435 41.6 11.1 11.1 1 3
436 44.1 11.1 11.1 1 4
437 41.6 11.1 11.1 1 5
438 44.1 11.1 11.1 1 6
439 41.6 11.1 11.1 1 7
440 44.1 11.1 11.1 1 8
441 41.6 11.1 11.1 1 9
442 44.1 11.1 11.1 1 10
443 41.6 11.1 11.1 1 11
444 44.1 11.1 11.1 1 12
445 41.6 11.1 11.1 2 1
446 44.1 11.1 11.1 2 2
447 41.6 11.1 11.1 2 3
448 44.1 11.1 11.1 2 4
449 41.6 11.1 11.1 2 5
450 44.1 11.1 11.1 2 6
451 41.6 11.1 11.1 2 7
452 44.1 11.1 11.1 2 8
453 41.6 11.1 11.1 2 9
454 44.1 11.1 11.1 2 10
455 41.6 11.1 11.1 2 11
456 44.1 11.1 11.1 2 12
457 41.6 11.1 11.1 3 1
458 44.1 11.1 11.1 3 2
459 41.6 11.1 11.1 3 3
460 44.1 11.1 11.1 3 4
461 41.6 11.1 11.1 3 5
462 44.1 11.1 11.1 3 6
463 41.6 11.1 11.1 3 7
464 44.1 11.1 11.1 3 8
465 41.6 11.1 11.1 3 9
466 44.1 11.1 11.1 3 10
467 41.6 11.1 11.1 3 11
468 44.1 11.1 11.1 3 12
469 41.6 11.1 11.1 4 1
470 44.1 11.1 11.1 4 2
471 41.6 11.1 11.1 4 3
472 44.1 11.1 11.1 4 4
473 41.6 11.1 11.1 4 5
474 44.1 11.1 11.1 4 6
475 41.6 11.1 11.1 4 7
476 44.1 11.1 11.1 4 8
477 41.6 11.1 11.1 4 9
478 44.1 11.1 11.1 4 10
479 41.6 11.1 11.1 4 11
480 44.1 11.1 11.1 4 12
481 41.6 11.1 11.1 5 1
482 44.1 11.1 11.1 5 2
483 41.6 11.1 11.1 5 3
484 44.1 11.1 11.1 5 4
485 41.6 11.1 11.1 5 5
486 44.1 11.1 11.1 5 6
487 41.6 11.1 11.1 5 7
488 44.1 11.1 11.1 5 8
489 41.6 11.1 11.1 5 9
490 44.1 11.1 11.1 5 10
491 41.6 11.1 11.1 5 11
492 44.1 11.1 11.1 5 12
493 41.6 11.1 11.1 6 1
494 44.1 11.1 11.1 6 2
495 41.6 11.1 11.1 6 3
496 44.1 11.1 11.1 6 4
497 41.6 11.1 11.1 6 5
498 44.1 11.1 11.1 6 6
499 41.6 11.1 11.1 6 7
500 44.1 11.1 11.1 6 8
501 41.6 11.1 11.1 6 9
502 44.1 11.1 11.1 6 10
503 41.6 11.1 11.1 6 11
504 44.1 11.1 11.1 6 12
505 41.6 11.1 11.1 7 1
506 44.1 11.1 11.1 7 2
507 41.6 11.1 11.1 7 3
508 44.1 11.1 11.1 7 4
509 41.6 11.1 11.1 7 5
510 44.1 11.1 11.1 7 6
511 41.6 11.1 11.1 7 7
512 44.1 11.1 11.1 7 8
513 41.6 11.1 11.1 7 9
514 44.1 11.1 11.1 7 10
515 41.6 11.1 11.1 7 11
516 44.1 11.1 11.1 7 12
517 41.6 11.1 11.1 8 1
518 44.1 11.1 11.1 8 2
519 41.6 11.1 11.1 8 3
520 44.1 11.1 11.1 8 4
521 41.6 11.1 11.1 8 5
522 44.1 11.1 11.1 8 6
523 41.6 11.1 11.1 8 7
524 44.1 11.1 11.1 8 8
525 41.6 11.1 11.1 8 9
526 44.1 11.1 11.1 8 10
527 41.6 11.1 11.1 8 11
528 44.1 11.1 11.1 8 12
529 41.6 11.1 11.1 9 1
530 44.1 11.1 11.1 9 2
531 41.6 11.1
```



```

> x=is.na(df)
> df[x]=0
> df

```

	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	11.3	62	5	4
5	0	0	14.3	56	5	5
6	28	0	14.9	66	5	6
7	23	239	8.6	65	5	7
8	19	99	12.8	59	5	8
9	8	19	20.1	61	5	9
10	0	194	8.6	69	5	10
11	7	0	8.9	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	72	5	22
23	4	25	9.7	61	5	23
24	32	92	12.0	61	5	24
25	0	66	16.6	57	5	25
26	0	266	14.9	58	5	26
27	0	0	8.0	57	5	27
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
32	0	286	8.6	78	6	1
33	0	287	9.7	74	6	2
34	0	242	16.1	67	6	3
35	0	186	9.2	84	6	4
36	0	220	8.6	85	6	5
37	0	264	14.3	79	6	6
38	29	127	9.7	82	6	7
39	0	273	6.8	87	6	8
40	71	291	13.8	90	6	9
41	39	323	11.5	87	6	10
42	0	259	10.9	93	6	11
43	0	250	9.2	92	6	12
44	23	148	8.0	82	6	13
45	0	322	13.8	80	6	14
46	0	322	11.5	79	6	15
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
52	0	150	6.3	77	6	21
53	0	59	1.7	76	6	22
54	0	91	4.6	76	6	23
55	0	250	6.3	76	6	24
56	0	135	8.0	75	6	25
57	0	127	8.0	78	6	26
58	0	47	10.3	73	6	27
59	0	98	11.5	80	6	28
60	0	31	14.9	77	6	29
61	0	138	8.0	83	6	30
62	135	269	4.1	84	7	1
63	49	248	9.2	85	7	2
64	32	236	9.2	81	7	3
65	0	101	10.9	84	7	4
66	64	175	4.6	83	7	5
67	40	314	10.9	83	7	6
68	77	276	5.1	88	7	7