Simple Operations

a) Enter the data {2,5,3,7,1,9,6}, directly and store it in a variable x.

Input:

```
> x<-c(2,5,3,7,1,9,6)
> x
```

Output:

```
[1] 2 5 3 7 1 9 6
```

b) Find the number of elements in x, i.e. in the data list.

Input:

```
> length(x)
```

Output:

```
[1] 7
```

c) Find the last element of x.

Input:

```
> x[length(x)]
```

Output:

```
[1] 6
```

d) Find the minimum element of x.

Input:

```
> min(x)
```

Output:

```
[1] 1
```

e) Find the maximum element of x.

Input:

```
> max(x)
```

Output:

```
[1] 9
```

1. Enter the data {1, 2, ... ,19,20} in a variable x.

a) Find the 3rd element in the data list.

Input:

```
> x<-c(1:20)
> x[3]
```

Output:

```
[1] 3
```

b) Find 3rd to 5th element in the data list.

Input:

```
> x[3:5]
```

Output:

```
[1] 3 4 5
```

c) Find 2nd, 5th, 6th, and 12th element in the list.

Input:

```
> x[c(2,5,6,12)]
```

Output:

```
[1]  2  5  6 12
```

d) Print the data as {20, 19, ..., 2, 1} without again entering the data.

Input:

```
> x[length(x):1]
```

Output:

```
[1] 20 19 18 17 16 15 14 13 12 11 10  9  8  7  6  5  4  3  2  1
```

2.

a) Create a data list (4, 4, 4, 4, 3, 3, 3, 5, 5, 5) using ‘rep’ function.

Input:

```
> x<-c(rep(4,4),rep(3,3),rep(5,3))
> x
```

Output:

```
[1] 4 4 4 4 3 3 3 5 5 5
```

b) Create a list (4, 6, 3, 4, 6, 3, ..., 4, 6, 3) where there 10 occurrences of 4, 6, and 3 in the given order.

Input:

```
> x<-c(rep(c(4,6,3),10))
> x
```

Output:

```
[1] 4 6 3 4 6 3 4 6 3 4 6 3 4 6 3 4 6 3 4 6 3 4 6 3 4 6 3 4 6 3
```

c) Create a list (3, 1, 5, 3, 2, 3, 4, 5, 7,7, 7, 7, 7,7, 6, 5, 4, 3, 2, 1, 34, 21, 54) using one line command.

Input:

```
> x<-c(3,1,5,3,2,3,4,5,c(rep(7,6)),c(6:1),34,21,54)
> x
```

Output:

```
[1]  3  1  5  3  2  3  4  5  7  7  7  7  7  7  6  5  4  3  2  1 34 21 54
```

d) First create a list (2, 1, 3, 4). Then append this list at the end with another list (5, 7, 12, 6, -8). Check whether the number of elements in the augmented list is 11.

Input:

```
> x<-c(2,1,3,4)
> x<-c(x,c(5,7,12,6,-8))
> length(x) == 11
```

Output:

```
[1] FALSE
```

3.

a) Print all numbers starting with 3 and ending with 7 with an increment of 0:5. Store these numbers in x..

Input:

```
> x <- seq(3, 7, 0.5)
> x
```

Output:

```
[1] 3.0 3.5 4.0 4.5 5.0 5.5 6.0 6.5 7.0
```

b) Print all even numbers between 2 and 14 (both inclusive)

Input:

```
> seq(2, 14, 2)
```

Output:

```
[1]  2  4  6  8 10 12 14
```

c) Type 2*x and see what you get. Each element of x is multiplied by 2.

Input:

```
> 2*x
```

Output:

```
[1]  6  7  8  9 10 11 12 13 14
```

4. Few simple statistical measures:

a) Enter data as 1,2, ... ,10.

Input:

```
> x <- 1 : 10
> x
```

Output:

```
[1]  1  2  3  4  5  6  7  8  9 10
```

b) Find sum of the numbers.

Input:

```
> sum(x)
```

Output:

```
[1] 55
```

c) Find mean, median.

Input:

```
> c(mean(x), median(x))
```

Output:

```
[1] 5.5 5.5
```

d) Find sum of squares of these values.

Input:

```
> sum(x*x)
```

Output:

```
[1] 385
```

e) Find the value of $1\frac{1}{n}\sum_{i=1}^{n}|x_i - \bar{x}|$, This is known as mean deviation about mean $(MD_{\bar{x}})$.

Input:

```
> md<-mean(abs(x-mean(x)))
> md
```

Output:

```
[1] 2.5
```

f) Check whether $(MD_{\bar{x}})$ is less than or equal to standard deviation.

Input:

```
md<=sd(x)
```

Output:

```
[1] TRUE
```

5. Few simple statistical measures:

a) Enter data as 1,2, ... ,10.

<u>Input:</u>

```
> x <- 1 : 10
> x
```

<u>Output:</u>

```
[1]  1  2  3  4  5  6  7  8  9 10
```

b) Find sum of the numbers.

<u>Input:</u>

```
> sum(x)
```

<u>Output:</u>

```
[1] 55
```

c) Find mean, median.

<u>Input:</u>

```
> c(mean(x), median(x))
```

<u>Output:</u>

```
[1] 5.5 5.5
```

d) Find sum of squares of these values.

<u>Input:</u>

```
> sum(x*x)
```

<u>Output:</u>

```
[1] 385
```

e) Find the value of $\frac{1}{n}\sum_{i=1}^{n}|x_i - \bar{x}|$, This is known as mean deviation about mean $(MD_{\bar{x}})$.

Input:

```
> md<-mean(abs(x-mean(x)))
> md
```

Output:

```
[1] 2.5
```

f) Check whether $(MD_{\bar{x}})$ is less than or equal to standard deviation.

Input:

```
md<=sd(x)
```

Output:

```
[1] TRUE
```

6. Reading a data file and working with it:

b) Read the file first and store it in a variable.

Input:

```
> x<- read.csv("house_data_1.csv",header=T)
> x
```

Output:

```
   Price FloorArea Rooms Age CentralHeating
1  52.00      1225     3 6.2             no
2  54.75      1230     3 7.5             no
3  57.50      1200     3 4.2             no
4  57.50      1000     2 8.8             no
5  59.75      1420     4 1.9            yes
6  62.50      1450     3 5.2             no
7  64.75      1380     4 6.6            yes
8  67.25      1510     4 2.3             no
9  67.50      1400     5 6.1             no
10 69.75      1550     6 9.2             no
```

```
11 70.00    1720    6 4.3    yes
12 75.50    1700    5 4.3     no
13 77.50    1660    6 1.0    yes
14 78.00    1800    7 7.0    yes
15 81.25    1830    6 3.6    yes
16 82.50    1790    6 1.7    yes
17 86.25    2010    6 1.2    yes
18 87.50    2000    6 0.0    yes
19 88.00    2100    8 2.3    yes
20 92.00    2240    7 0.7    yes
```

b) How many rows are there in this table? How many columns are there?.

Input:

```
> c(nrow(x), ncol(x))
```

Output:

```
[1] 20  5
```

c) How to find the number of rows and number of columns by a single command?.

Input:

```
> dim(x)
```

Output:

```
[1] 20  5
```

d) What are the variables in the data file?

Input:

```
> names(x)
```

Output:

```
[1] "Price"        "FloorArea"     "Rooms"        "Age"
"CentralHeating"
```

e) If the file is very large, naturally we cannot simply type `a', because it will cover the entire screen and we won't be able to understand anything. So how to see the top or bottom few lines in this file?.

Input:

```
> head(x)
```

Output:

```
   Price FloorArea Rooms Age CentralHeating
1 52.00     1225     3 6.2           no
2 54.75     1230     3 7.5           no
3 57.50     1200     3 4.2           no
4 57.50     1000     2 8.8           no
5 59.75     1420     4 1.9          yes
6 62.50     1450     3 5.2           no
```

Input:

```
> tail(x)
```

Output:

```
    Price FloorArea Rooms Age CentralHeating
15 81.25     1830     6 3.6          yes
16 82.50     1790     6 1.7          yes
17 86.25     2010     6 1.2          yes
18 87.50     2000     6 0.0          yes
19 88.00     2100     8 2.3          yes
20 92.00     2240     7 0.7          yes
```

f) If the number of columns is too large, we may face the same problem again. So how to see the first 5 rows and the first 3 columns?

Input:

```
x[1 : 5, 1 : 3]
```

Output:

	Price	FloorArea	Rooms
1	52.00	1225	3
2	54.75	1230	3
3	57.50	1200	3
4	57.50	1000	2
5	59.75	1420	4

g) How to get 1st, 3rd, 6th, and 10th row and 2nd, 4th, and 5th column?

Input:

```
x[c(1, 3, 6, 10), c(2, 4, 5)]
```

Output:

	FloorArea	Age	CentralHeating
1	1225	6.2	no
3	1200	4.2	no
6	1450	5.2	no
10	1550	9.2	no

h) How to get values in a specific row or a column?

Input:

> x[2,]

Output:

```
Price FloorArea Rooms Age CentralHeating
2 54.75      1230     3 7.5           no
```

Input:

> x[,3]

Output:

```
[1] 3 3 3 2 4 3 4 4 5 6 6 5 6 7 6 6 6 6 8 7
```

7.

a) Print all numbers starting with 3 and ending with 7 with an increment of 0:5. Store these numbers in x..

Input:

```
> x <- seq(3, 7, 0.5)
> x
```

Output:

```
[1] 3.0 3.5 4.0 4.5 5.0 5.5 6.0 6.5 7.0
```

b) Print all even numbers between 2 and 14 (both inclusive)

Input:

> seq(2, 14, 2)

Output:

```
[1]  2  4  6  8 10 12 14
```

c) Type 2*x and see what you get. Each element of x is multiplied by 2.

Input:

```
> 2*x
```

Output:

```
[1]  6  7  8  9 10 11 12 13 14
```

8. Calculate simple statistical measures using the values in the data file.

a) Find means, medians, and standard deviations of Price, Floor Area, Rooms, and Age.

Input:

```
> c(mean(x[,1]),median(x[,1]),sd(x[,1]))
```

Output:

```
[1] 71.58750 69.87500 12.21094
```

Input:

```
> c(mean(x[,2]),median(x[,2]),sd(x[,2]))
```

Output:

```
[1] 1610.7500 1605.0000  331.9649
```

Input:

```
> c(mean(x[,3]),median(x[,3]),sd(x[,3]))
```

Output:

```
[1] 5.00000 5.50000 1.65434
```

Input:

```
> c(mean(x[,4]),median(x[,4]),sd(x[,4]))
```

Output:

```
[1] 4.205000 4.250000 2.786523
```

b) How many houses have central heating and how many don't have?

<u>Input:</u>

```
> sum(x$CentralHeating=="yes")
```

<u>Output:</u>

```
[1] 11
```

<u>Input:</u>

```
> sum(x$CentralHeating=="no")
```

<u>Output:</u>

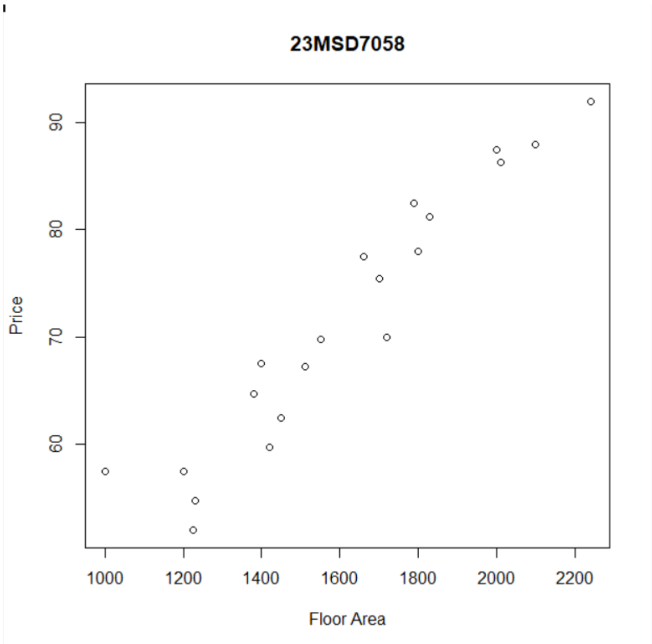
```
[1] 9
```

c) Plot Floor vs. Price, Price vs. Age, and Price vs. rooms, in separate graphs.

<u>Input:</u>

```
> plot(x$FloorArea,x$Price,main="23MSD7058",xlab="Floor Area",ylab="Price")
```

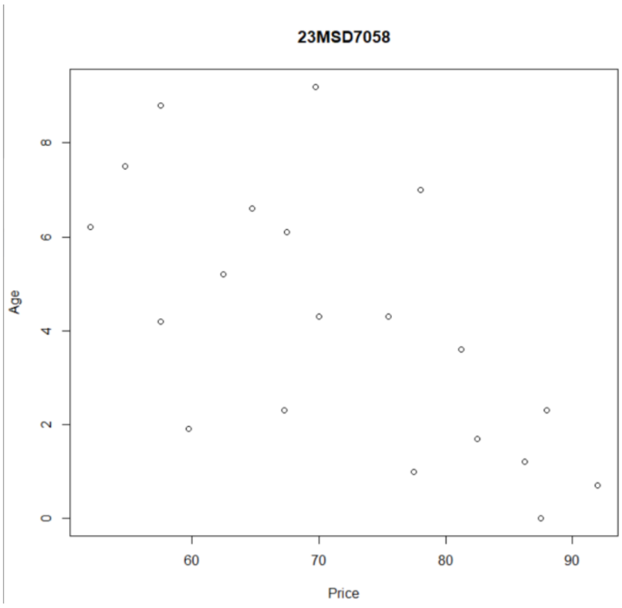
<u>Output:</u>



Input:

> plot(x$Price,x$Age,main="23MSD7058",xlab="Price",
ylab="Age")

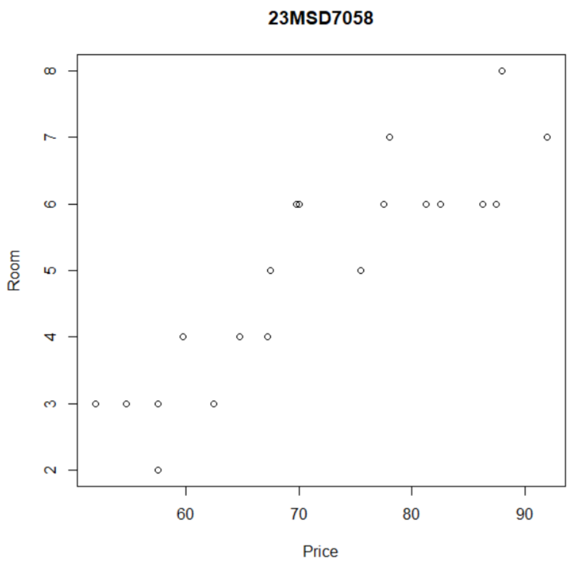
Output:



Input:

```
> plot(x$Price,x$Rooms,main="23MSD7058",xlab="Floor Area",ylab="Price")
```

Output:

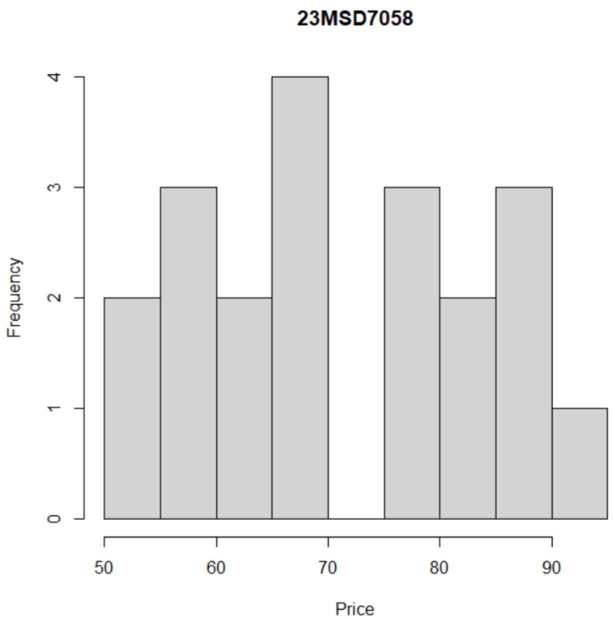


d) Draw histograms of Prices, FloorArea, and Age..

Input:

```
> hist(x$Price,main="23MSD7058",xlab="Price")
```

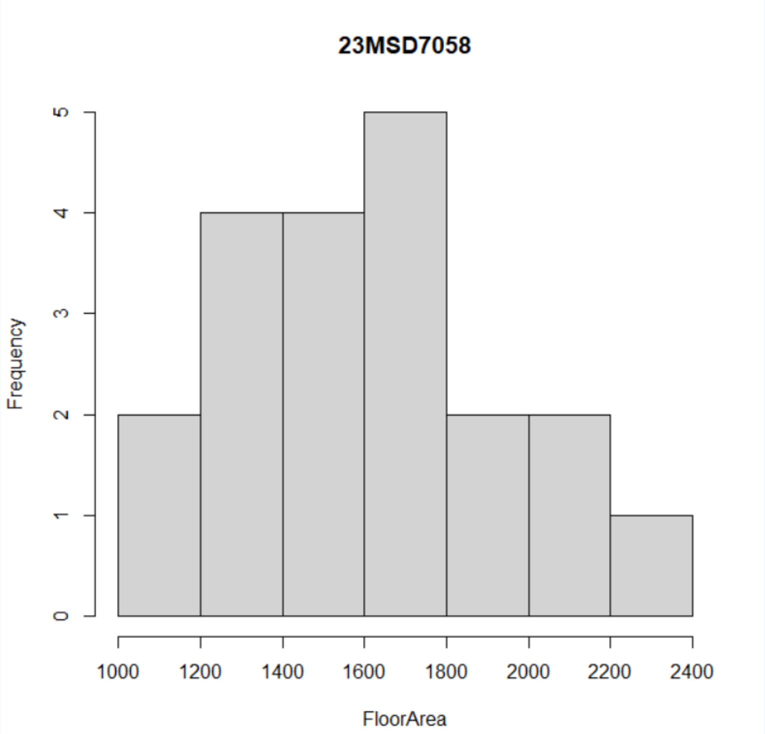
Output:



Input:

> hist(x$FloorArea,main="23MSD7058",xlab="FloorArea")

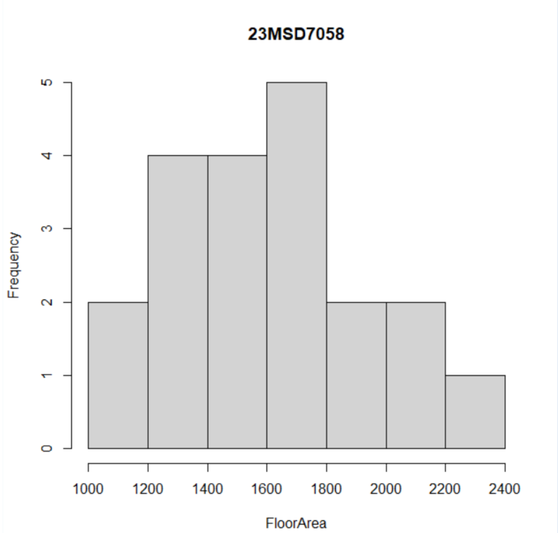
Output:



Input:

> hist(x$Age,main="23MSD7058",xlab="Age")

Output:

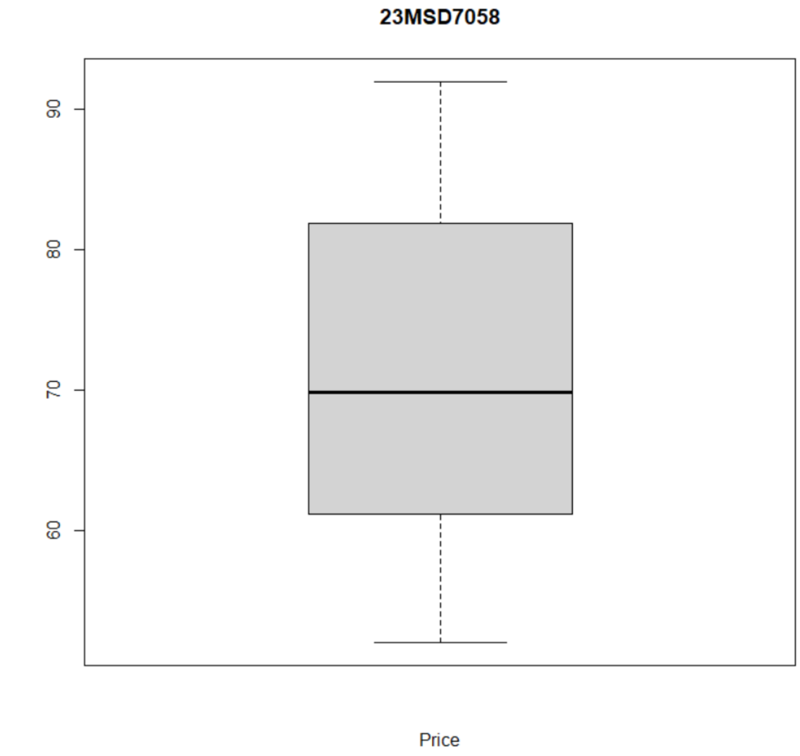


e) Draw box plots of Price, FloorArea, and Age.

Input:

```
> boxplot(x$Price,main="23MSD7058",xlab="Price")
```

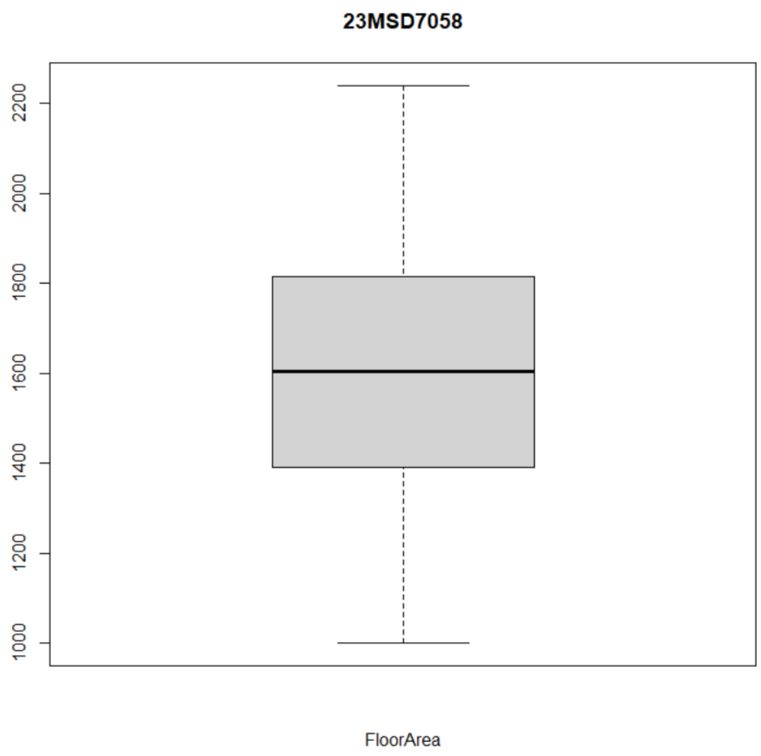
Output:



Input:

```
> boxplot(x$FloorArea,main="23MSD7058",xlab="
FloorArea")
```

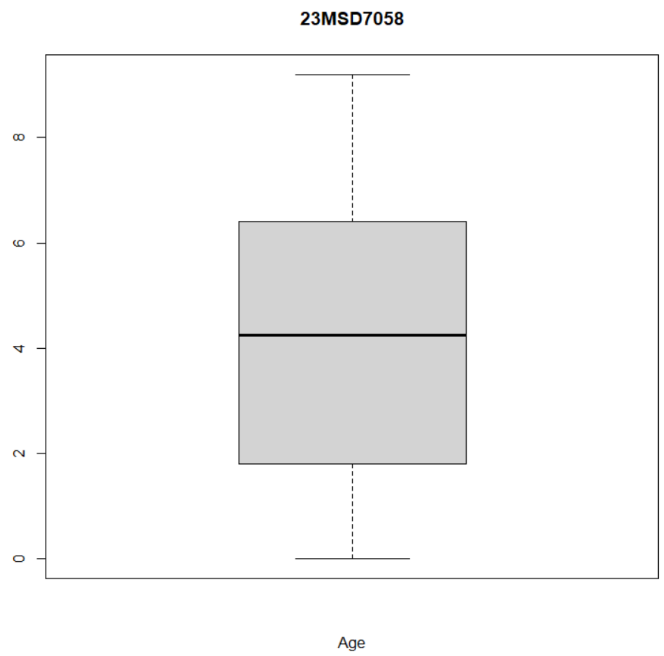
Output:



<u>Input:</u>

> boxplot(x$Age,main="23MSD7058",xlab="Age")

<u>Output:</u>

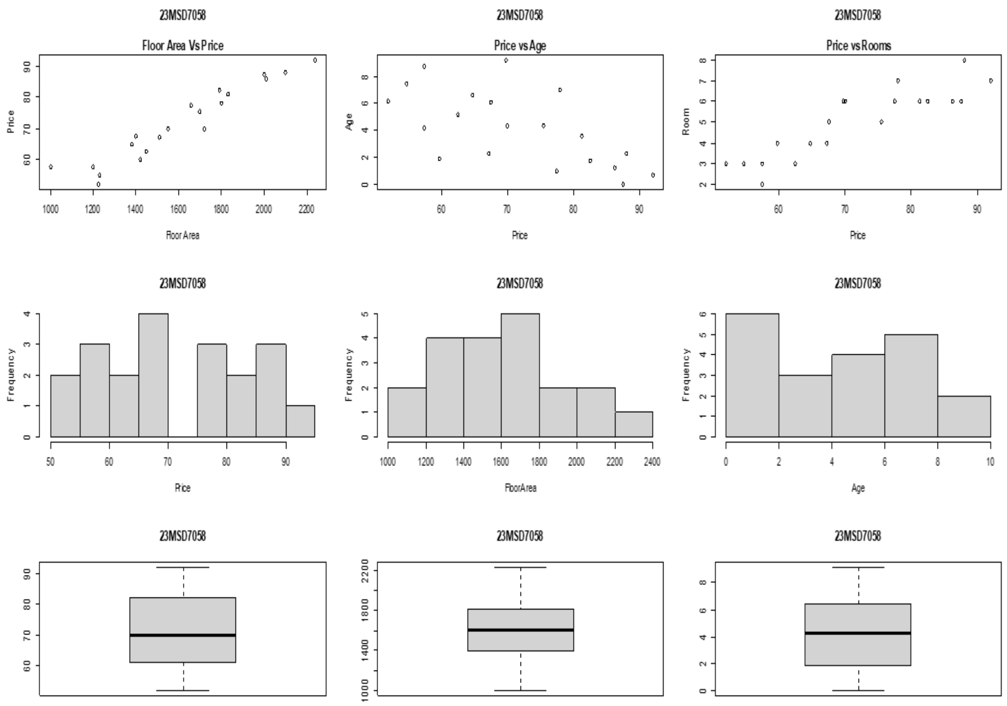


f) Draw all the graphs in (c), (d), and (e) on the same graph paper..

Input:

```
> par(mfrow =c(3,3))
>plot(x$FloorArea,x$Price,main="23MSD7058\n\nFloor Area Vs Price",xlab="Floor Area",ylab="Price")
> plot(x$Price,x$Age,main="23MSD7058\n\nPrice vs Age",xlab="Price",ylab="Age")
> plot(x$Price,x$Rooms,main="23MSD7058\n\nPrice vs Rooms",xlab="Price",ylab="Room")
> hist(x$Price,main="23MSD7058",xlab="Price")
>hist(x$FloorArea,main="23MSD7058",xlab="FloorArea")
> hist(x$Age,main="23MSD7058",xlab="Age")
> boxplot(x$Price,main="23MSD7058",xlab="Price")
>boxplot(x$FloorArea,main="23MSD7058",xlab="FloorArea")
>boxplot(x$Age,main="23MSD7058",xlab="Age")
```

Output:



9. Five terminals on an on-line computer system are attached to a communication line to the central computer system. The probability that any terminal is ready to transmit is 0.95. Let X denote the number of ready terminals.

a) Find the probability of the 3rd terminal.

Input:

```
> dbinom(3,5,0.95)
```

Output:

```
[1] 0.02143438
```

b) Also find the individual prob. of all terminals.

Input:

```
> dbinom(0:5,5,0.95)
```

Output:

```
[1] 0.0000003125 0.0000296875 0.0011281250
0.0214343750 0.2036265625
[6] 0.7737809375
```

10. A fair coin is tossed 10 times ; success and failure are "heads" and "tails" respectively, each with probability, 0.5 . Let X be the number of heads (successes) obtained.

c) Find the prob. of all cases and round off up to 4 decimal places.

Input:

```
> round(dbinom(0:10,10,0.5),4)
```

Output:

```
[1] 0.0010 0.0098 0.0439 0.1172 0.2051 0.2461 0.2051
0.1172 0.0439 0.0098
```

[11] 0.0010

b) Find the cumulative probability of all.

Input:

> round(pbinom(0:10,10,0.5),4)

Output:

[1] 0.0010 0.0107 0.0547 0.1719 0.3770 0.6230 0.8281 0.9453 0.9893 0.9990

[11] 1.0000

11. It is known that 20% of integrated circuit chips on a production line are defective. To maintain and monitor the quality of the chips, a sample of twenty chips is selected at regular intervals for inspection. Let X denote the number of defectives found in the sample.

a) Find the probability of all cases.

Input:

> dbinom(0:20,20,0.2)

Output:

[1] 1.152922e-02 5.764608e-02 1.369094e-01 2.053641e-01 2.181994e-01

[6] 1.745595e-01 1.090997e-01 5.454985e-02 2.216088e-02 7.386959e-03

[11] 2.031414e-03 4.616849e-04 8.656592e-05 1.331783e-05 1.664729e-06

[16] 1.664729e-07 1.300570e-08 7.650410e-10 3.187671e-11 8.388608e-13

[21] 1.048576e-14

b) Find the cumulative probability of all..

Input:

```
> pbinom(0:20,20,0.2)
```

Output:

```
[1] 0.01152922 0.06917529 0.20608472 0.41144886
0.62964826 0.80420779

[7] 0.91330749 0.96785734 0.99001821 0.99740517
0.99943659 0.99989827

[13] 0.99998484 0.99999815 0.99999982 0.99999999
1.00000000 1.00000000

[19] 1.00000000 1.00000000 1.00000000
```

12. It is known that 1% of bits transmitted through a digital transmission are received in error. One hundred bits are transmitted each day. Let X denote the number of bits found in error each day.

a) Find the probability of all cases.

Input:

```
> round(dbinom(0:100,100,0.01),4)
```

Output:

```
[1] 0.3660 0.3697 0.1849 0.0610 0.0149 0.0029 0.0005
0.0001 0.0000 0.0000

[11] 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
0.0000 0.0000 0.0000

[21] 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
0.0000 0.0000 0.0000

[31] 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
0.0000 0.0000 0.0000

[41] 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
0.0000 0.0000 0.0000
```

[51] 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000

[61] 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000

[71] 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000

[81] 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000

[91] 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000

[101] 0.0000

b) Find the cumulative probability of all.

Input:

```
> round(pbinom(0:100,100,0.01),4)
```

Output:

[1] 0.3660 0.7358 0.9206 0.9816 0.9966 0.9995 0.9999 1.0000 1.0000 1.0000

[11] 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000

[21] 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000

[31] 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000

[41] 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000

[51] 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000

[61] 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000

[71] 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000

[81] 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000

[91] 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000

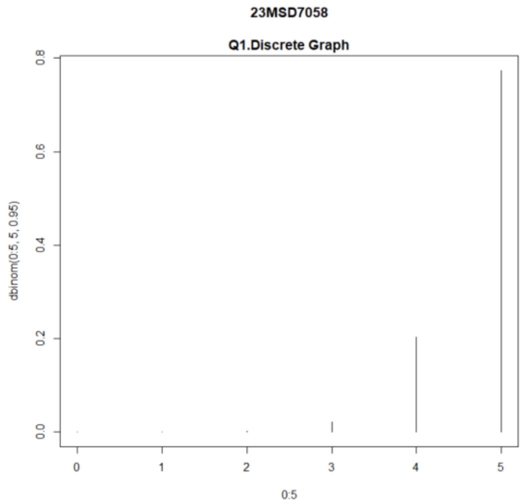
[101] 1.0000

13. Plot all the graphs bar diagram and step function

Input:

```
> plot(0:5,dbinom(0:5,5,0.95),type='h')
> title('23MSD7058\n\nQ1.Discrete Graph')
```

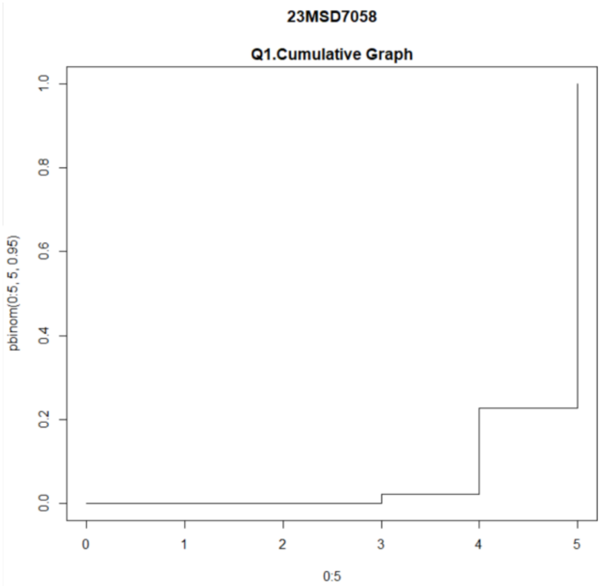
Output:



Input:

```
> plot(0:5,pbinom(0:5,5,0.95),type='s')
> title('23MSD7058\n\nQ1.Cumulative Graph')
```

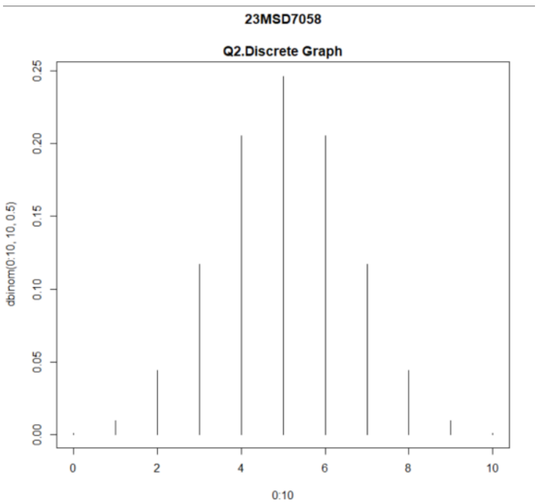
Output:



Input:

```
> plot(0:10,dbinom(0:10,10,0.5),type='h')
> title('23MSD7058\n\nQ2.Discrete Graph')
```

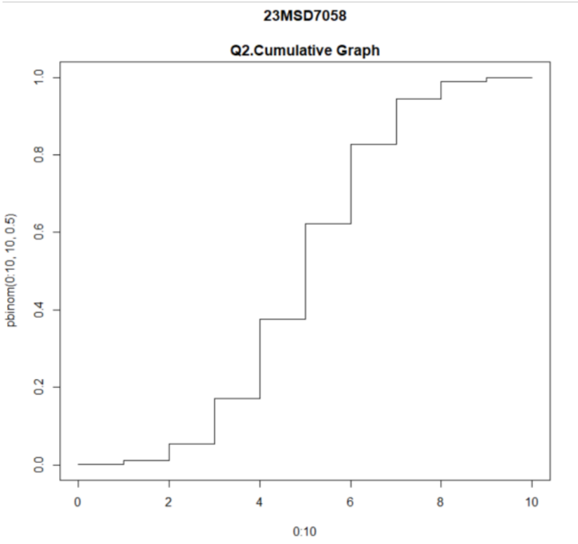
Output:



Input:

```
> plot(0:10,pbinom(0:10,10,0.5),type='s')
> title('23MSD7058\n\nQ2.Cumulative Graph')
```

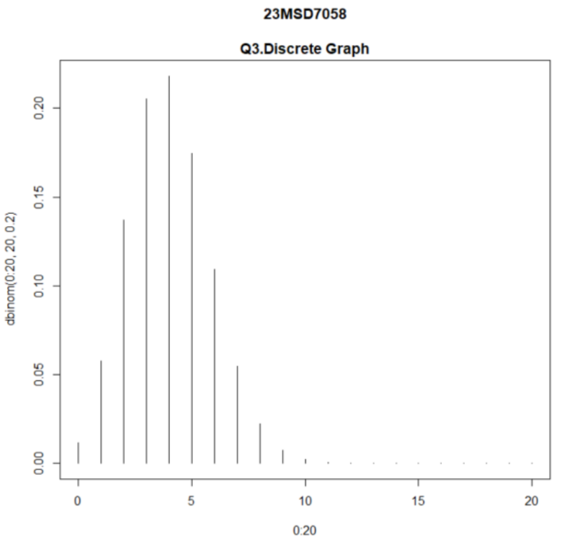
Output:



Input:

> plot(0:10,pbinom(0:10,10,0.5),type='s')
> title('23MSD7058\n\nQ2.Cumulative Graph')

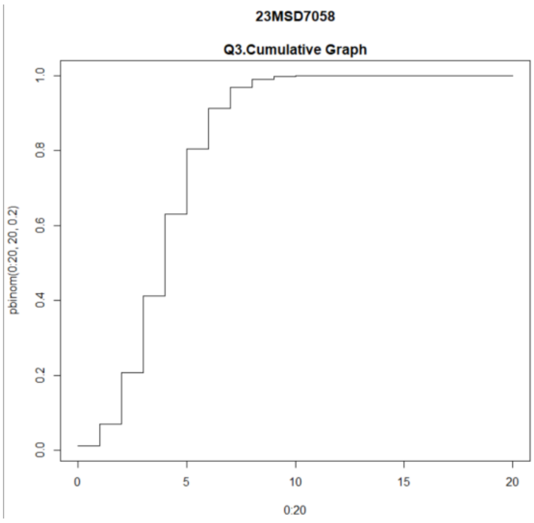
Output:



Input:

> plot(0:10,pbinom(0:10,10,0.5),type='s')
> title('23MSD7058\n\nQ2.Cumulative Graph')

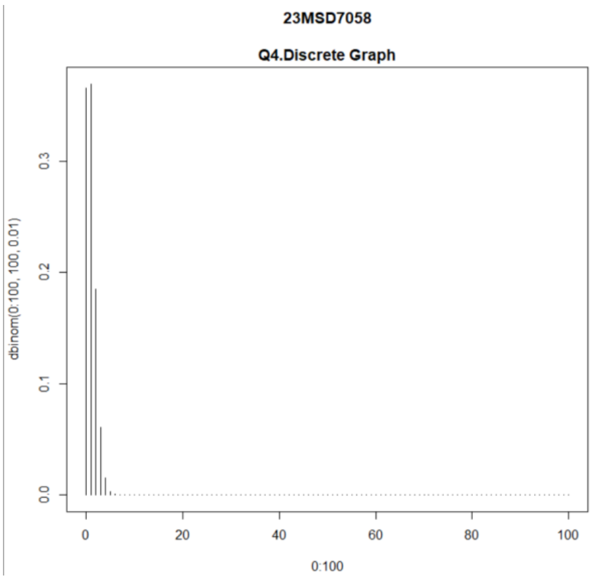
Output:



Input:

> plot(0:100,dbinom(0:100,100,0.01),type='h')
> title('23MSD7058\n\nQ4.Discrete Graph')

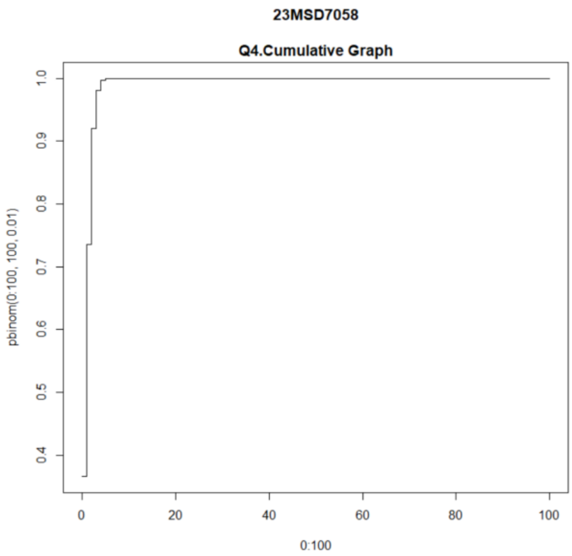
Output:



Input:

> plot(0:100,dbinom(0:100,100,0.01),type='h')
> title('23MSD7058\n\nQ4.Discrete Graph')

Output:



23MSD7058
Q4.Cumulative Graph
pbinom(0:100, 100, 0.01)
0.4
0.5
0.6
0.7
0.8
0.9
1.0
0
20
40
60
80
100
0:100

14.Suppose IQs are normally distributed with a mean of 100 and a standard deviation of 15..

a. What percentage of people have an IQ less than 125.

Input:

```
> pnorm(125,mean=100,sd=15)
```

Output:

```
[1] 0.9522096
```

b. What percentage of people have an IQ greater than 110?

Input:

```
> 1-pnorm(110,mean=100,sd=15)
```

Output:

```
[1] 0.2524925
```

c. What percentage of people have an IQ between 110 and 125?.

Input:

```
> iq_125=pnorm(125,mean=100,sd=15)
> iq_110=pnorm(110,mean=100,sd=15)
> iq_125-iq_110
```

Output:

```
[1] 0.2047022
```

d. What IQ separates the lower 25% from the others?

Input:

```
> qnorm(0.25,100,15)
```

Output:

[1] 89.88265

e. What IQ separates the top 10% from the others?

Input:

```
> qnorm(0.90,100,15)
```

Output:

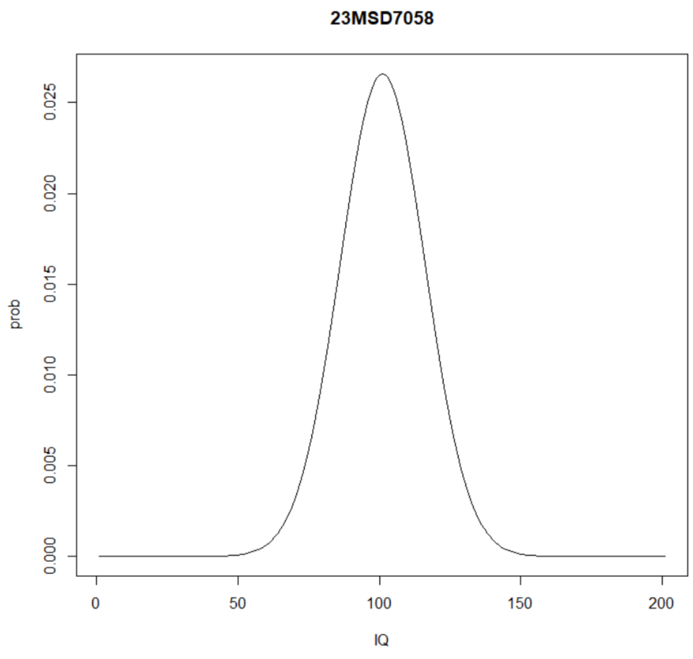
[1] 119.2233

f. Plot the above normal distribution.

Input:

```
> prob<-dnorm(seq(0,200),mean=100,sd=15)
> plot(prob,type='l',main='23MSD7058',xlab='IQ')
```

Output:



15.Suppose that the average number of accidents occurring weekly on a particular stretch of a highway equals 3. Calculate the probability that there is atleast one accident this week.

Input:

```
> 1-dpois(0,3)
```

Output:

```
[1] 0.9502129
```

16. Suppose the probability that an item produced by a certain machine will be defective is 0.1. Find the probability that a sample of 10 items will contain at most one defective item. Assume that the quality of successive items is independent.

Input:

```
> dpois(0,1)+dpois(1,1)
```

Output:

```
[1] 0.7357589
```

Input:

```
> ppois(1,1)
```

Output:

```
[1] 0.7357589
```

17. A large industrial firm allows a discount on any invoice that is paid within 30 days. Of all invoices, 10% receive the discount. In a company audit, 12 invoices are sampled at random. What is the probability that fewer than 4 of the sampled invoices receive the discount?

Input:

```
> sum(dpois(c(0,1,2,3),1.2))
```

Output:

[1] 0.966231

Input:

```
> ppois(3,1.2)
```

Output:

[1] 0.9662

18. Arsenic concentration in public drinking water supplies is a potential health risk. An article in the Arizona Republic (Sunday, May 27, 2001) reported drinking water arsenic concentrations in parts per billion (ppb) for 10 metropolitan Phoenix communities and 10 communities in rural Arizona. The data follow:

Metro Phoenix	Rural Arizona
Phoenix, 3	Rimrock, 48
Chandler, 7	Goodyear, 44
Gilbert, 25	New River, 40
Glendale, 10	Apachie Junction, 38
Mesa, 15	Buckeye, 33
Paradise Valley,6	Nogales, 21
Peoria, 12	Black Canyon City, 20
Scottsdale, 25	Sedona, 12
Tempe, 15	Payson, 1
Sun City, 7	Casa Grande, 18

We wish to determine if there is any difference in mean arsenic concentrations between metropolitan Phoenix communities and communities in rural Arizona.

Input:

```
> x1 = c(3,7,25,10,15,6,12,25,15,7)
> x2 = c(48,44,40,38,33,21,20,12,1,18)
> t.test(x1,x2,paired = FALSE,var.equal=FALSE,conf.level=0.95)
```

Output:

```
        Welch Two Sample t-test

data:  x1 and x2
```

t = -2.7669, df = 13.196, p-value = 0.01583

alternative hypothesis: true difference in means is not equal to 0

95 percent confidence interval:

-26.694067 -3.305933

sample estimates:

mean of x mean of y

12.5   27.5

19. Suppose we want to know whether the choice of sport is independent of gender. So,we asked one hundred men and one hundred women which sport they prefer to play among archery, boxing and cycling and summarizes the data obtained in the following two-way table.

Suppose we want to know whether the choice of sport is independent of gender. So, we asked one hundred men and one hundred women which sport they prefer to play among archery, boxing, and cycling and summarizes the data obtained in the following two-way table.

		Sport Preference			
		Archery	Boxing	Cycling	
Gender	Female	35	15	50	**100**
	Male	10	30	60	**100**
		45	**45**	**110**	**200**

Input:

```
> female<-c(35,15,50)
> male<-c(10,30,60)
> data1=data.frame(female,male)
> chisq.test(data1)
```

Output:

```
        Pearson's Chi-squared test

data:  data1
```

X-squared = 19.798, df = 2, p-value = 5.023e-05

12.5 27.5