

```
In [2]: ▶ import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from scipy import stats as st
import seaborn as sb
```

Question 1

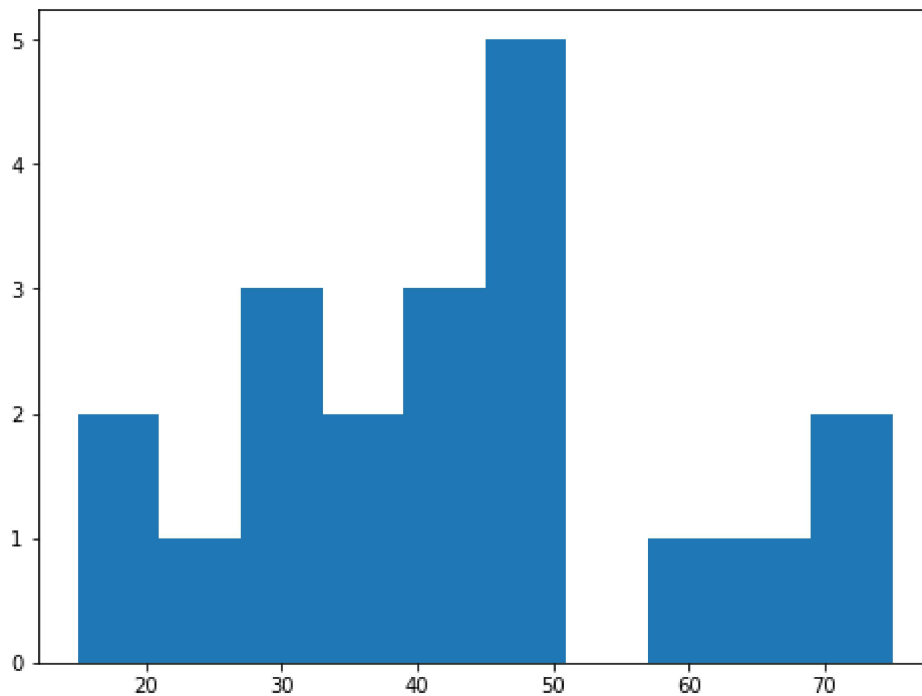
```
In [3]: ▶ nomreh = [15,20,25,28,30,32,35,38,40,45,48,49,42,43,45,46,60,65,69,75]
```

```
In [4]: ▶ len(nomreh)
```

Out[4]: 20

```
In [10]: ▶ fig = plt.figure(figsize=(8,6))
plt.hist(nomreh, bins=10)
```

Out[10]: (array([2., 1., 3., 2., 3., 5., 0., 1., 1., 2.]),
array([15., 21., 27., 33., 39., 45., 51., 57., 63., 69., 75.]),
<BarContainer object of 10 artists>)



```
In [11]: ▶ np.mean (nomreh)
```

Out[11]: 42.5

```
In [12]: ▶ st.skew (nomreh)
```

```
Out[12]: 0.34954164246512265
```

```
In [95]: ▶ np.median(nomreh)
```

```
Out[95]: 42.5
```

```
In [96]: ▶ st.mode(nomreh)
```

```
Out[96]: ModeResult(mode=array([45]), count=array([2]))
```

Question 2

```
In [13]: ▶ add = np.arange(0,100,5)  
add
```

```
Out[13]: array([ 0,  5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 75, 80,  
              85, 90, 95])
```

```
In [14]: ▶ n1 = np.array(nomreh)  
n1
```

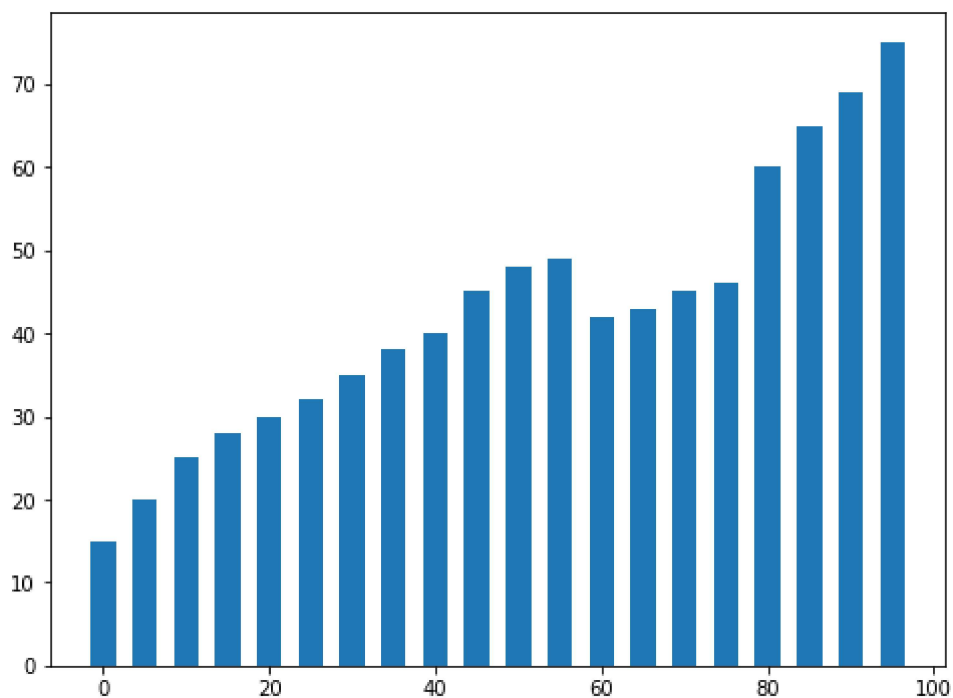
```
Out[14]: array([15, 20, 25, 28, 30, 32, 35, 38, 40, 45, 48, 49, 42, 43, 45, 46, 60,  
              65, 69, 75])
```

```
In [15]: ▶ add_1 = list(add)  
len(add_1)
```

```
Out[15]: 20
```

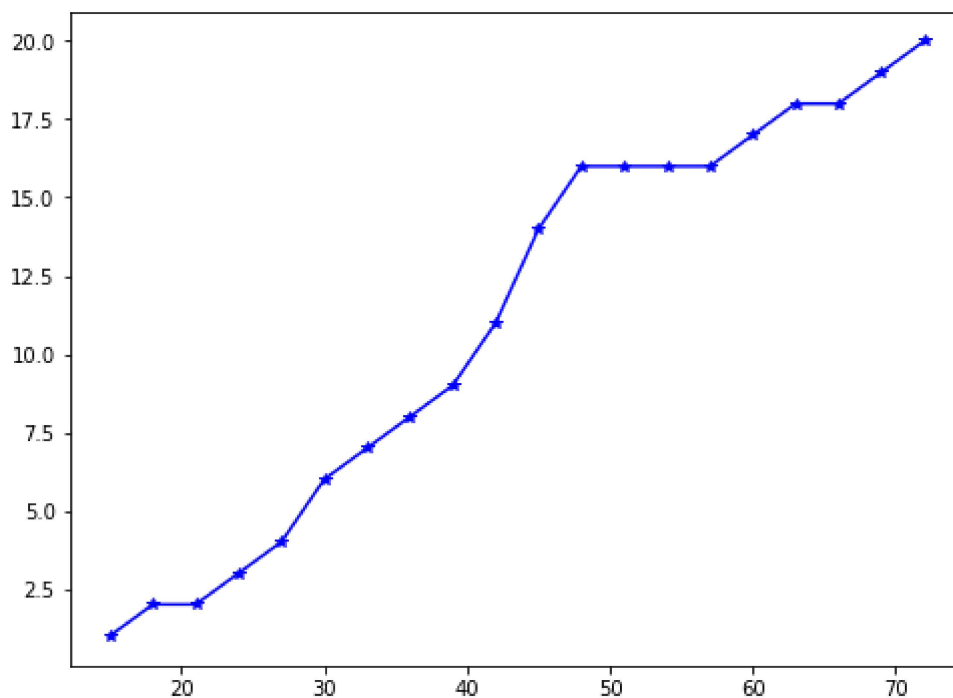
```
In [16]: fig = plt.figure(figsize=(8,6))  
plt.bar(add,nomreh,width=3)
```

Out[16]: <BarContainer object of 20 artists>



```
In [17]: value,base = np.histogram(nomreh,bins=20)  
cum = np.cumsum(value)  
fig = plt.figure(figsize=(8,6))  
plt.plot(base[:-1],cum,'b-*')
```

Out[17]: [<matplotlib.lines.Line2D at 0x222314ff6a0>]

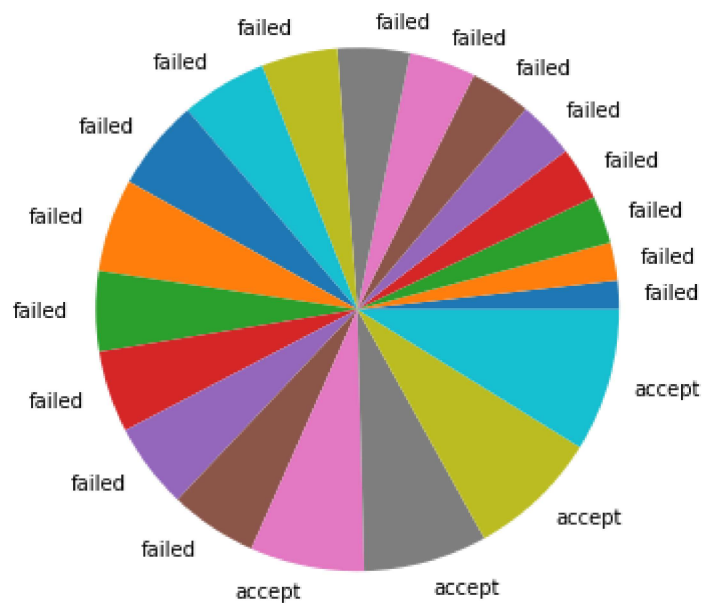


The score of 75 is among scattered data

Question 3

```
In [18]: ▶ vote = []
a = "accept"
b = "failed"
c = str(a)
d = str(b)
for i in nomreh:
    if i<60:
        vote.append(d)
    elif i>=60:
        vote.append(c)

fig = plt.figure(figsize=(8,6))
plt.pie(nomreh,labels = vote)
plt.show()
```

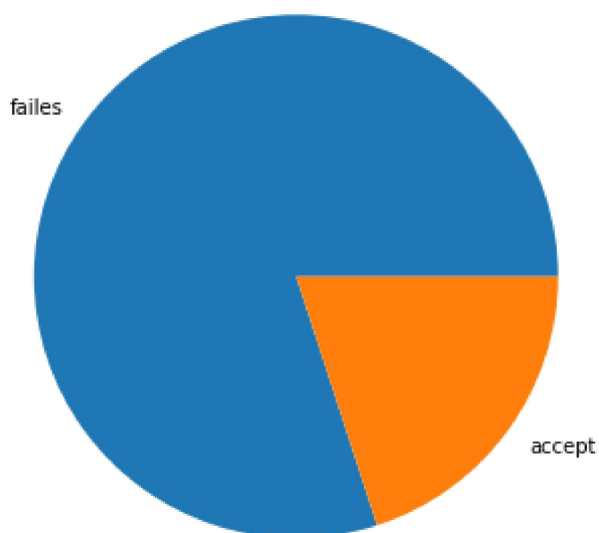


```
In [19]: ▶ print(vote)

['failed', 'failed', 'failed', 'failed', 'failed', 'failed', 'failed', 'f
ailed', 'failed', 'failed', 'failed', 'failed', 'failed', 'fail
ed', 'failed', 'accept', 'accept', 'accept', 'accept']
```

```
In [47]: ▶ vote_1 = ['failes', 'accept']  
add_2 = [80, 20]  
fig = plt.figure(figsize=(8, 6))  
plt.pie(add_2, labels = vote_1)
```

```
Out[47]: ([<matplotlib.patches.Wedge at 0x22232cde0d0>,  
          <matplotlib.patches.Wedge at 0x22232cde610>],  
          [Text(-0.8899187180267095, 0.6465637441936395, 'failes'),  
          Text(0.8899187482945414, -0.6465637025335375, 'accept')])
```

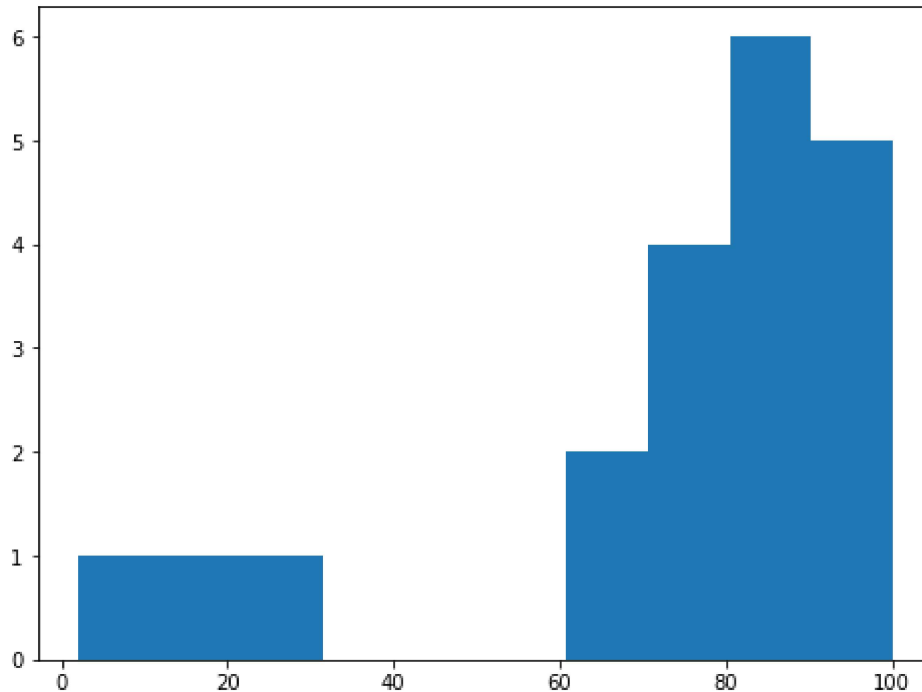


Question 4

```
In [148]: ▶ nomreh_2 = [100, 98, 97, 95, 93, 90, 88, 85, 83, 82, 81, 79, 77, 75, 72, 69, 68, 22, 17, 2]
```

```
In [149]: fig = plt.figure(figsize=(8,6))  
plt.hist(nomreh_2, bins=10)
```

```
Out[149]: (array([1., 1., 1., 0., 0., 0., 2., 4., 6., 5.]),  
array([ 2., 11.8, 21.6, 31.4, 41.2, 51., 60.8, 70.6, 80.4,  
90.2, 100. ]),  
<BarContainer object of 10 artists>)
```



```
In [150]: st.skew(nomreh_2)
```

```
Out[150]: -1.560231750404717
```

```
In [151]: np.mean (nomreh_2)
```

```
Out[151]: 73.65
```

```
In [152]: np.median(nomreh_2)
```

```
Out[152]: 81.5
```

```
In [153]: st.mode(nomreh_2)
```

```
Out[153]: ModeResult(mode=array([2]), count=array([1]))
```

```
In [155]: st.mode(nomreh_2)
```

```
Out[155]: ModeResult(mode=array([2]), count=array([1]))
```

Question 5

```
In [156]: ▶ normal = pd.read_excel('normal.xlsx')  
normal
```

Out[156]:

	nomreh
0	15
1	20
2	25
3	28
4	30
5	32
6	35
7	38
8	40
9	45
10	48
11	49
12	42
13	43
14	45
15	46
16	60
17	65
18	69
19	75

```
In [159]: ► cholegy = pd.read_excel('cholegy.xlsx')  
cholegy
```

Out[159]:

nomreh	
0	100
1	98
2	97
3	95
4	93
5	90
6	88
7	85
8	83
9	82
10	81
11	79
12	77
13	75
14	72
15	69
16	68
17	12
18	2
19	1

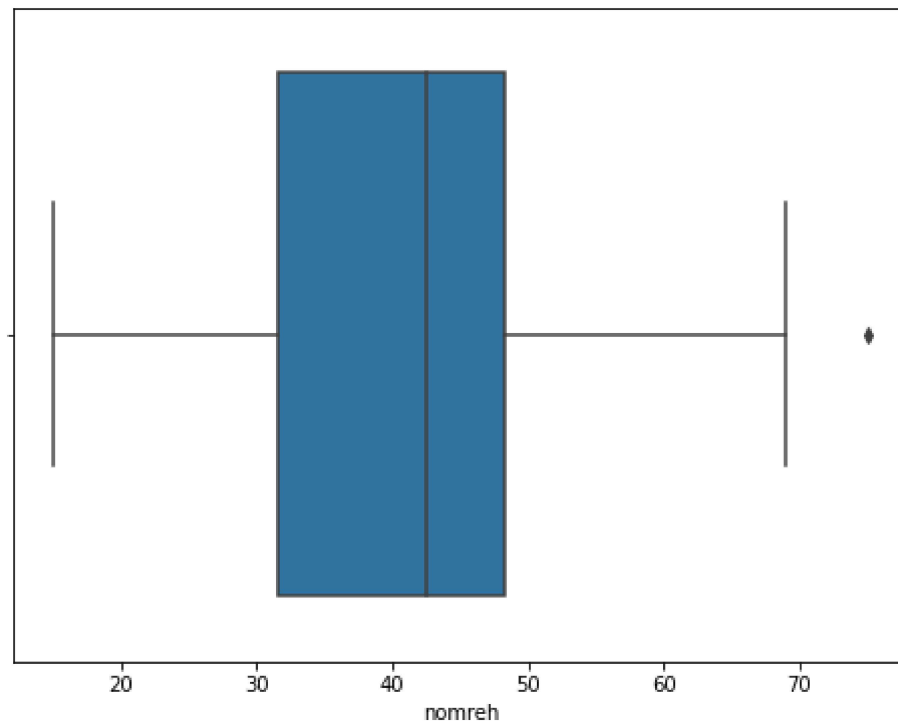
Question 6

```
In [157]: ► nomreh_normal = normal['nomreh']
```



```
In [163]: fig = plt.figure(figsize=(8,6))  
          sb.boxplot(x=nomreh_normal)
```

```
Out[163]: <AxesSubplot:xlabel='nomreh'>
```



```
In [160]: nomreh_cholegy = cholegy['nomreh']
```

```
In [162]: ▶ fig = plt.figure(figsize=(10,8))  
sb.boxplot(x=nomreh_cholegy)
```

```
Out[162]: <AxesSubplot:xlabel='nomreh'>
```

