In [1]:
```
import pandas as pd
import numpy as np
import math
```

In [2]:
```
path="E:\\New folder\\IBM-313 Marks.xlsx"
table=pd.read_excel(path)
table
```

Out[2]:

	S.No.	MTE (25)	Mini Project (25)	Total (50)	ETE (50)	Total
0	1	5.00	20	25.00	12.0	37.00
1	2	11.05	20	31.05	26.0	57.05
2	3	8.10	20	28.10	14.0	42.10
3	4	6.00	10	16.00	13.0	29.00
4	5	11.35	20	31.35	17.0	48.35
...	...	...	...	...	...	...
74	75	12.05	10	22.05	20.0	42.05
75	76	12.25	10	22.25	28.0	50.25
76	77	1.75	10	11.75	NaN	0.00
77	78	3.00	10	13.00	NaN	0.00
78	79	5.80	10	15.80	12.0	27.80

79 rows × 6 columns

In [3]:
```
x=table['Total']
np.mean(x)
```

Out[3]:
```
46.90632911392405
```

In [4]:
```
np.median(x)
```

Out[4]:
```
45.0
```

In [5]:
```
import statistics
statistics.mode(x)
```

Out[5]:
```
48.35
```

In [6]:
```
a=np.array([1,2,3,4,5])
p=np.percentile(a,50)#return 50th percentile
print(p)
```

```
3.0
```

In [7]:
```
k=['Ram',65,2.5]
print(k)
```

```
['Ram', 65, 2.5]
```

In [8]:
```
for i in k:
    print(i)
```

```
Ram
65
2.5
```

In [9]:
```
a=np.array([1,2,3,4,5])
q1=np.percentile(a,25)
q1
```

Out[9]:
```
2.0
```

In [10]:
```
q2=np.percentile(a,50)
q2
```

Out[10]:
```
3.0
```

In [11]:
```
q3=np.percentile(a,75)
q3
```

Out[11]:
```
4.0
```

In [12]:
```
IQ=q3-q1 #Inter-Quartile range
IQ
```

Out[12]:
```
2.0
```

In [13]:
```
np.var(x)#population variance
```

Out[13]:
```
262.78147892965876
```

In [14]:
```
import statistics
statistics.pstdev(x)#population standard deviation
```

Out[14]:
```
16.210536046955966
```

In [15]:
```
statistics.stdev(x)#sample standard deviation
```

Out[15]:
```
16.31411880088133
```

In [16]:
```
np.std(x)#population standard deviation
```

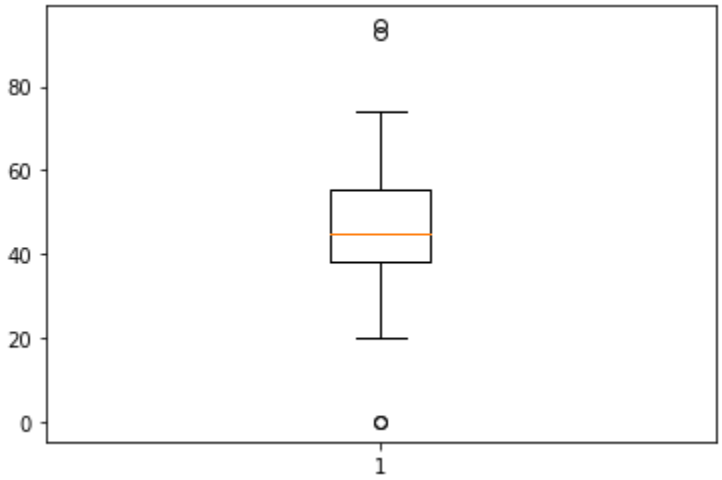
Out[16]:
```
16.210536046955966
```

In [17]:
```
from scipy.stats import skew
skew(x)#To find skewness of the data
```

Out[17]:
```
0.10226407464884266
```

In [18]:

```
pyplot.boxplot(x)
pyplot.show() #here circle represent that some data are out-layred means some data are be
```



Binomial Distribution

A Gallup survey found that 65% of all financial consumers were very satisfied with their primary financial institution. Suppose that 25 financial consumers are sampled and if the Gallup survey result still holds true today, what is the probability that exactly 19 are very satisfied with their primary financial institution?

In [19]:
```
import scipy
from scipy.stats import binom
```

In [20]:
```
print(binom.pmf(k=19,n=25,p=0.65))#pmf -> probability mass function,cuz here we want to k
#Jo mere x hai wo yaha k hai
```

```
0.090777998593228
```

According to the U.S. Census Bureau, approximately 6% of all workers in Jackson, Mississippi, are unemployed. In conducting a random telephone survey in Jackson, what is the probability of getting two or fewer unemployed workers in a sample of 20?

In [21]:
```
binom.cdf(2,20,0.06)#cdf->cummulative distribution function,cuz here we want P(0+1+2)or P
```

Out[21]:
```
0.8850275957378549
```

Poisson Distribution

In [22]:
```
from scipy.stats import poisson
```

In [23]:
```
poisson.pmf(3,2)# x=3, lamda=mean=variance=2
```

Out[23]:
```
0.18044704431548356
```

Bank customers arrive randomly on week days afternoon at an average of 3.2 customers every 4 minutes. What is the probability of exactly 5 customers in a 4 minute interval on a week day afternoon?

In [24]:
```
poisson.pmf(5,3.2) #P(X=5)
```

Out[24]:
```
0.11397938346351824
```

In [25]:
```
#Bank customers arrive randomly on week days afternoon at an average of 3.2 customers eve
#What is the probability of having more than 7 customers in a 4 minute interval on a week
```

In [26]:
```
prob=poisson.cdf(7,3.2) #P(X>7)=1-P(X=0)-P(X=1)-P(X=2)-P(X=3)-P(X=4)-P(X=5)-P(X=6)-P(X=7)
#CDF considers from (0-X)
prob
```

Out[26]:
```
0.9831701582510425
```

In [27]:
```
prob_x_more_than_7=1-prob
prob_x_more_than_7
```

Out[27]:
```
0.01682984174895752
```

Bank customers arrive randomly on week days afternoon at an average of 3.2 customers every 4 minutes. What is the probability of exactly 10 customers in 8 minute interval on a week day afternoon?

In [28]:
```
poisson.pmf(10,6.4) #P(X=10) and to match the unit=8mins will multiply 3.2*2/4*2=6.4/8 mi
```

Out[28]:
```
0.052790043854115495
```

Uniform Distribution

Suppose the amount of time it takes to assemble a plastic module ranges from 27 to 39 seconds and that assembly times are uniformly distributed. Describe the distribution. What is the probability that a given assembly will take between 30 and 35 seconds?

In [29]:
```
U=np.arange(27,40,1)
U
```

Out[29]:
```
array([27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39])
```

In [30]:
```
from scipy.stats import uniform
uniform.mean(loc=27,scale=12) #scale =39-27
```

Out[30]:
```
33.0
```

In [31]:
```
uniform.cdf(np.arange(30,36,1),loc=27,scale=12)
```

Out[31]:
```
array([0.25      , 0.33333333, 0.41666667, 0.5       , 0.58333333,
       0.66666667])
```

In [32]:
```
Prob=0.66666667-0.25
Prob
```

Out[32]:
```
0.41666667
```

According to the National Association of Insurance Commissioners, the average annual cost for automobile insurance in the United States in a recent year was 691. Suppose automobile

insurance costs are uniformly distributed in the United States with a range from 200 to 1,182. What is the standard deviation of this uniform distribution?

In [33]:
```
uniform.mean(loc=200,scale=982)
```

Out[33]:
```
691.0
```

In [34]:
```
print(math.sqrt(982**2/12)) #Use formula of sqrt(variance) in Uniform Distribution
```

```
283.4789821721062
```

In [35]:
```
uniform.std(loc=200,scale=982)
```

Out[35]:
```
283.4789821721062
```

Normal Distribution

In [36]:
```
from scipy.stats import norm
```

In [37]:
```
val=68 #x=68
mean=65.5
sd=2.5 #std
norm.cdf(val,mean,sd)
```

Out[37]:
```
0.8413447460685429
```

cdf(x>val)

In [38]:
```
print(1-norm.cdf(val,mean,sd))
```

```
0.15865525393145707
```

cdf(val1<x<val2)

In [39]:
```
norm.cdf(68,mean,sd)-norm.cdf(63,mean,sd)
```

Out[39]:
```
0.6826894921370859
```

What is the probability of obtaining a score greater than 700 on a GMAT test that has a mean of 494 and a standard deviation of 100? Assume GMAT scores are normally distributed.

P(x > 700 | μ = 494 and σ = 100) = ?

In [40]:
```
prob=norm.cdf(700,494,100)
prob
```

Out[40]:
```
0.9803007295906231
```

In [41]:
```
actual_prob=1-prob
actual_prob
```

Out[41]:
```
0.019699270409376912
```

What is the probability of obtaining a score greater than 550 or less?

In [42]:
```
norm.cdf(550,494,100)
```

Out[42]:
```
0.712260281150973
```

What is the probability of obtaining a score between 300 and 600 on GMAT Exam?

In [43]:
```
norm.cdf(600,494,100)-norm.cdf(300,494,100)
```

Out[43]:
```
0.8292378553956377
```

In [44]:
```
norm.ppf(0.95)# probablity function for finding a given area
```

Out[44]:
```
1.6448536269514722
```

In [45]:
```
norm.ppf(1-0.6772)#cuz we are moving letf side
```

Out[45]:
```
-0.45988328292440145
```

Hypergeomertic Distribution

Assume that 18 major computer manufacturing companies operate in the United States and that 12 are located in Silicon Valley of California. If three of these companies are randomly selected from the list, what is the probability that none of the selected companies is located in Silicon Valley of California?

In [46]:
```
from scipy.stats import hypergeom
pval=hypergeom.cdf(0,18,3,12) #sf is survavl function sf=1-cdf P(X=3)
#(x,N,n,A)
pval
```

Out[46]:
```
0.024509803921568606
```

In [47]:
```
actual=1-pval #P(X=0) mtlb humne prob nikala X=3 ke liye usme se 1-p(X) kiye to P(X=0) ke
actual
```

Out[47]:
```
0.9754901960784313
```

A western city has 18 police officers eligible for promotion. Eleven of the 18 are Hispanic. Suppose only five of the police officers are chosen for promotion and that one is Hispanic. If the officers chosen for promotion had been selected by chance alone, what is the probability that one or fewer of the five promoted officers would have been Hispanic?

In [48]:
```
pval=hypergeom.cdf(1,18,5,11)
pval
```

Out[48]:
```
0.04738562091503275
```

Exponential Distribution

A manufacturing firm has been involved in statistical quality control for several years. As part of

y selected and tested. From the records of these tests, it

has been established that a defective part occurs in a pattern that is Poisson distributed on the average of 1.38 defects every 20 minutes during production runs. Use this information to determine the probability that less than 15min will elapse between any two defects.

In [49]:
```
mean_of_expo=1/1.38# cuz lamda is given in Poissons form
mean_of_expo
```

Out[49]:
```
0.7246376811594204
```

In [50]:
```
from scipy.stats import expon
```

In [51]:
```
expon.cdf(0.75,0,(1/1.38)) #15/20 =0.75,loc=0
```

Out[51]:
```
0.6447736190750485
```

In [ ]:
```

```