Aim:- Study and write a python program for calculating the central tendencies and measuring the dispersion of data. Use the house price prediction dataset and statistic library functions

Objectives:

To learn important functions of numpy library.

To learn central tendency measures and their implementation in python.

To learn measure of dispersion and its implementation in python

Course Outcomes: CO1

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

In [2]:
```
data = [15,28,99,86,88,110,103,87,94,78,77,85,46,35,68]
```

In [3]:
```
print(st.mean(data))
```

```
73.26666666666667
```

In [4]:
```
print(st.mode(data))
```

```
15
```

In [5]:
```
print(st.median(data))
```

```
85
```

In [6]:
```
print(np.sort(data))
```

```
[ 15  28  35  46  68  77  78  85  86  87  88  94  99 103 110]
```

In [8]:
```
print(st.stdev(data))
```

```
28.966647487527215
```

In [10]:
```
print(st.variance(data))
```

```
839.0666666666667
```

In [11]:
```
print(np.max(data))
```

```
110
```

In [12]:
```
print(np.min(data))
```

```
15
```

In [13]:
```
print(np.sum(data))
```

```
1099
```

In [15]:
```
print(st.harmonic_mean(data))
```

```
53.76452451942067
```

In [16]:
```
print(np.percentile(data,50))
```

```
85.0
```

In [17]:
```
print(st.median_low(data))
```

```
85
```

In [18]:
```
print(st.median_high(data))
```

```
85
```

In [20]:
```
from scipy.stats import skew
print(skew(data))
```

```
-0.7775097482749155
```

In [22]:
```
import scipy
scipy.stats.kurtosis(data)
```

Out[22]:
```
-0.6327948421225051
```

In [27]:
```
print(st.median_grouped(data,2))
```

```
85.0
```

In [28]:
```
print(st.median_grouped(data,2))
```

```
85.0
```

In [29]:
```
print(st.pvariance(data))
```

```
783.1288888888889
```

In [30]:
```
print(st.pstdev(data))
```

```
27.984440121054572
```

In [31]:
```
df = pd.read_csv("Exp-1 House Price.csv")
```

In [32]:
```
df.head(5)
```

Out[32]:

	Area	BHK	Bathroom	Furnishing	Locality	Parking	Price	Status	Transaction	Type	Per_Sqft
0	800.0	3	2.0	Semi-Furnished	Rohini Sector 25	1.0	6500000	Ready_to_move	New_Property	Builder_Floor	NaN
1	750.0	2	2.0	Semi-Furnished	J R Designers Floors, Rohini Sector 24	1.0	5000000	Ready_to_move	New_Property	Apartment	6667.0
2	950.0	2	2.0	Furnished	Citizen Apartment, Rohini Sector 13	1.0	15500000	Ready_to_move	Resale	Apartment	6667.0
3	600.0	2	2.0	Semi-Furnished	Rohini Sector 24	1.0	4200000	Ready_to_move	Resale	Builder_Floor	6667.0
4	650.0	2	2.0	Semi-Furnished	Rohini Sector 24 carpet area 650 sqft status R...	1.0	6200000	Ready_to_move	New_Property	Builder_Floor	6667.0

In [33]:
```
df.count()
```

Out[33]:
```
Area           1259
BHK            1259
Bathroom       1257
Furnishing     1254
Locality       1259
Parking        1226
Price          1259
Status         1259
Transaction    1259
Type           1254
Per_Sqft       1018
dtype: int64
```

In [34]:
```
df.sum()
```

```
C:\Users\admin\AppData\Local\Temp\ipykernel_14236\1703867807.py:1: FutureWarning: The default value of numeric_only in DataFram
e.sum is deprecated. In a future version, it will default to False. In addition, specifying 'numeric_only=None' is deprecated. S
elect only valid columns or specify the value of numeric_only to silence this warning.
  df.sum()
```

Out[34]:
```
Area                                                1846263.9797
BHK                                                         3521
Bathroom                                                  3213.0
Locality       Rohini Sector 25J R Designers Floors, Rohini S...
Parking                                                   2373.0
Price                                                 26825140000
Status         Ready_to_moveReady_to_moveReady_to_moveReady_t...
Transaction    New_PropertyNew_PropertyResaleResaleNew_Proper...
Per_Sqft                                              15972559.0
dtype: object
```

In [35]:
```
df.max()
```

```
C:\Users\admin\AppData\Local\Temp\ipykernel_14236\1151452817.py:1: FutureWarning: The default value of numeric_only in DataFram
e.max is deprecated. In a future version, it will default to False. In addition, specifying 'numeric_only=None' is deprecated. S
elect only valid columns or specify the value of numeric_only to silence this warning.
  df.max()
```

Out[35]:
```
Area                                                  24300.0
BHK                                                        10
Bathroom                                                 7.0
Locality       mind. The space is airy, is well located and c...
Parking                                                  114.0
Price                                                240000000
Status                                           Ready_to_move
Transaction                                             Resale
Per_Sqft                                             183333.0
dtype: object
```

In [36]:
```
df.min()
```

```
C:\Users\admin\AppData\Local\Temp\ipykernel_14236\3962516015.py:1: FutureWarning: The default value of numeric_only in DataFram
e.min is deprecated. In a future version, it will default to False. In addition, specifying 'numeric_only=None' is deprecated. S
elect only valid columns or specify the value of numeric_only to silence this warning.
  df.min()
```

Out[36]:
```
Area                                        28.0
BHK                                            1
Bathroom                                     1.0
Locality       APL Builder Floor, Greater Kailash 1
Parking                                      1.0
Price                                    1000000
Status                                Almost_ready
Transaction                        New_Property
Per_Sqft                                  1259.0
dtype: object
```

In [37]:
```
df.std()
```

```
C:\Users\admin\AppData\Local\Temp\ipykernel_14236\3390915376.py:1: FutureWarning: The default value of numeric_only in DataFram
e.std is deprecated. In a future version, it will default to False. In addition, specifying 'numeric_only=None' is deprecated. S
elect only valid columns or specify the value of numeric_only to silence this warning.
  df.std()
```

Out[37]:
```
Area        1.568055e+03
BHK         9.544249e-01
Bathroom    1.042220e+00
Parking     6.279212e+00
Price       2.560115e+07
Per_Sqft    2.113474e+04
dtype: float64
```

In [38]:
```
df.describe()
```

Out[38]:

	Area	BHK	Bathroom	Parking	Price	Per_Sqft
count	1259.000000	1259.000000	1257.000000	1226.000000	1.259000e+03	1018.000000
mean	1466.452724	2.796664	2.556086	1.935563	2.130670e+07	15690.136542
std	1568.055040	0.954425	1.042220	6.279212	2.560115e+07	21134.738568
min	28.000000	1.000000	1.000000	1.000000	1.000000e+06	1259.000000
25%	800.000000	2.000000	2.000000	1.000000	5.700000e+06	6364.000000
50%	1200.000000	3.000000	2.000000	1.000000	1.420000e+07	11291.500000
75%	1700.000000	3.000000	3.000000	2.000000	2.550000e+07	18000.000000
max	24300.000000	10.000000	7.000000	114.000000	2.400000e+08	183333.000000

In [39]:
```
df.corr()
```

```
C:\Users\admin\AppData\Local\Temp\ipykernel_14236\1134722465.py:1: FutureWarning: The default value of numeric_only in DataFram
e.corr is deprecated. In a future version, it will default to False. Select only valid columns or specify the value of numeric_o
nly to silence this warning.
  df.corr()
```

Out[39]:

	Area	BHK	Bathroom	Parking	Price	Per_Sqft
Area	1.000000	0.449438	0.535104	-0.009297	0.580836	0.162832
BHK	0.449438	1.000000	0.773267	-0.070707	0.571523	0.181540
Bathroom	0.535104	0.773267	1.000000	-0.032796	0.728108	0.219169
Parking	-0.009297	-0.070707	-0.032796	1.000000	-0.000448	0.001607
Price	0.580836	0.571523	0.728108	-0.000448	1.000000	0.322859
Per_Sqft	0.162832	0.181540	0.219169	0.001607	0.322859	1.000000

In [40]:
```
df.skew()
```

```
C:\Users\admin\AppData\Local\Temp\ipykernel_14236\1665899112.py:1: FutureWarning: The default value of numeric_only in DataFram
e.skew is deprecated. In a future version, it will default to False. In addition, specifying 'numeric_only=None' is deprecated.
Select only valid columns or specify the value of numeric_only to silence this warning.
  df.skew()
```

Out[40]:
```
Area         8.075156
BHK          0.530797
Bathroom     0.833207
Parking     15.123066
Price        3.498529
Per_Sqft     5.336103
dtype: float64
```

In [ ]:
```

```