

FINDING MEAN USING PROGRAM:

```
def mean(t):  
    return(float(sum(t))) / len(t)  
num = [10,20,30,40,50]  
print("Mean = ",mean(num))  
  
Mean = 30.0
```

FINDING MEAN USING NUMPY LIBRARY

```
import numpy as np  
np.mean([1,4,3,2,6,4,4,3,2,6])  
  
3.5
```

FINDING MEAN USING STATISTICS LIBRARY

```
import statistics as st  
num=[1,2,3,4,5,6,7,8,9]  
print(st.mean(num))  
  
5
```

MEDIAN

FINDING MEDIAN USING PROGRAM

```
# n=5  
# print(n//2)  
num=[10,20,30,40,50]  
n=len(num)  
num.sort()  
  
if(n%2 == 0):  
    med1 = num[n//2]  
    med2 = num[n//2] - 1  
    median = med1 + med2  
else:  
    median = num[n//2]  
  
print("Median is "+ str(median))  
  
Median is 30
```

FINDING MEDIAN USING NUMPY LIBRARY

```
import numpy as np  
num=[10,20,30,40,50]  
medi=np.median(num)  
print(medi)  
  
30.0
```

FINDING MEDIAN USING STATISTICS LIBRARY

```
import statistics as st  
  
st.median([10,20,30,40,50])  
  
30
```

MODE

FINDING MODE USING PROGRAM

```

from collections import Counter

# list of elements to calculate mode
n_num = [99,86,87,88,111,86,103,87,94,78,77,85,86]
n = len(n_num)

data = Counter(n_num)
get_mode = dict(data)
mode = [k for k, v in get_mode.items() if v == max(list(data.values()))]

if len(mode) == n:
    get_mode = "No mode found"
else:
    get_mode = "Mode is / are: " + ', '.join(map(str, mode))

print(get_mode)

    Mode is / are: 86

```

FINDING MODE USING SCIPY LIBRARY

#Finding mode using Scipy library:

```

from scipy import stats

speed= [99,86,87,88,111,86,103,87,94,78,77,85,86]
x = stats.mode(speed)

speed2=[4,1,2,2,3,5,4]
y = stats.mode(speed2)

print("Single Mode: ")
print(x)

print("\nMultiple Mode: ")
print(y)

Single Mode:
ModeResult(mode=array([86]), count=array([3]))

Multiple Mode:
ModeResult(mode=array([2]), count=array([2]))
<ipython-input-14-04ac3c3b1f50>:6: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `n
x = stats.mode(speed)
<ipython-input-14-04ac3c3b1f50>:9: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `n
y = stats.mode(speed2)

```

FINDING MODE USING STATISTICS LIBRARY

```

import statistics

print("Multimode : ",statistics.multimode([4,1,2,2,3,5,4]))
print("Mode      : ",statistics.mode([4,1,2,2,3,5,4]))

Multimode :  [4, 2]
Mode      :  4

```

FINDING VARIANCE USING NUMPY LIBRARY

```

import numpy as np

#Taking a list of elements:
list = [2,4,4,4,5,5,7,9]

#Calculating variance using var():
print(np.var(list))

4.0

```

FINDING VARIANCE USING STATISTICS LIBRARY

```
#Python program to calculate the variance of a list using SSTATISTICS:
```

```
dataset=[21,19,11,21,19,46,29]  
print(st.variance(dataset))
```

```
124.23809523809524
```

✓ 0s completed at 6:08 PM

