Statistics

data=[1,2,3,4,5]

mean( )
- It will perform the operation on list (float and integer) . Mean will take the sum of values and divide with the number of elements present in the list
- In other words it will take the average

harmonic_mean( )
- To find the harmonic mean of a set of n numbers, add the reciprocals of the numbers in the set, divide the sum by n, then take the reciprocal of the result.

```
>>> import statistics as s
>>> s.mean([1,2,3,4,5])
3
>>> s.mean([4,3,5,1,2])
3
>>> s.harmonic_mean([1,2,3,4,5])
2.18978102189781
```

median( )
- Taking the elements in sorted order and taking the middle element .
- If we are odd elements we can easily take middle number but if we have even number of elements it will take two middle numbers and add them . By adding them we get median. It will take the average

median_high( )
- If the number of list values is odd, it returns the exact middle value which is higher .

median_low( )
- It will return the lower middle value of the list

```
>>> import statistics as s
>>>
>>> s.median_low([1,2,3,4,5,6])
3
>>> s.median_high([1,2,3,4,5,6])
4
>>> s.median_high([1,2,30,50,51,52])
50
>>> s.median_low([1,2,30,50,51,52])
30
>>> s.median([1,2,30,50,51,52])
```

mode( )
• From the list of elements it will find out most occurring numbers .

```
>>> import statistics as s
>>> s.mode([1,1,1,2,2,3])
1
>>> s.mode([1,1,1,2,2,2,3])
1
>>> s.mode([4,4,4,1,1,1,2,2,2,3])
4
```

pvariance ( )
• It will find out the mean than it will take the whole square

pstdev( )
• Root of the pvariance is pstdev(standard division)

stdev( ) and variance ( )
It will not take the whole list . It will take some sample of data

```
>>> import statistics as s
>>> s.mode([1,1,1,2,2,3])
1
>>> s.mode([1,1,1,2,2,2,3])
1
>>> s.mode([4,4,4,1,1,1,2,2,2,3])
4
>>> s.pvariance([1,2,3,4,5])
2
>>> s.pstdev([1,2,3,4,5])
1.4142135623730951
>>> s.variance([1,2,3,4,5])
2.5
>>> s.stdev([1,2,3,4,5])
1.5811388300841898
```