Measure of center tendency

In [1]:
```
age = [12,23,45,23,43,54,34,23,12,34,54,56]
age
```

Out[1]:
```
[12, 23, 45, 23, 43, 54, 34, 23, 12, 34, 54, 56]
```

In [8]:
```
import numpy as np
```

In [9]:
```
print(np.mean(age))
```

```
34.416666666666664
```

In [10]:
```
print(np.median(age))
```

```
34.0
```

In [11]:
```
print(np.mode(age))  # mode is not f() of numpy
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
Cell In[11], line 1
----> 1 print(np.mode(age))

File C:\ProgramData\anaconda3\Lib\site-packages\numpy\__init__.py:320, in __g
etattr__(attr)
    317     from .testing import Tester
    318     return Tester
--> 320 raise AttributeError("module {!r} has no attribute "
    321                        "{!r}".format(__name__, attr))

AttributeError: module 'numpy' has no attribute 'mode'
```

In [13]:
```
import statistics as st
```

In [15]:
```
print(st.mean(age))
```

```
34.416666666666664
```

In [16]:
```
print(st.median(age))
```

```
34.0
```

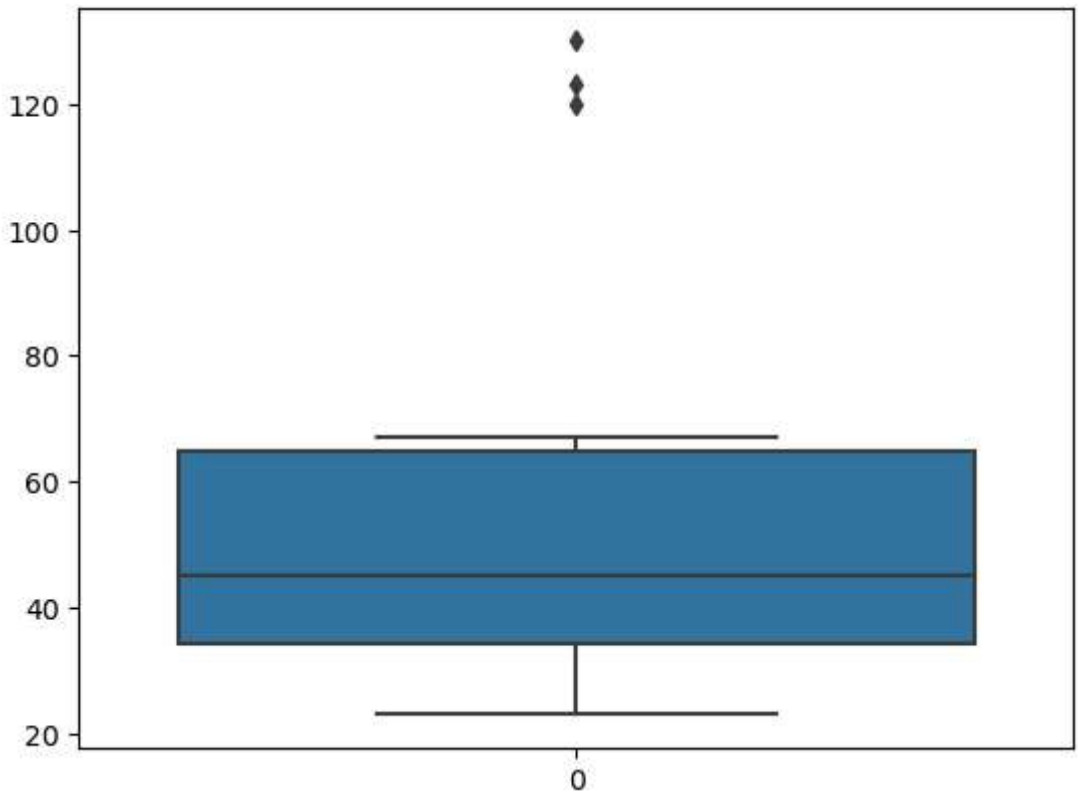
In [17]:
```
print(st.mode(age))
```

```
23
```

In [21]:
```
1 age = [23,34,45,34,45,67,45,34,56,43,23,35,65,44,65,120,130,123]
```

In [22]:
```
1 import seaborn as sns
2 sns.boxplot(age)
```

Out[22]: <Axes: >



5 number summary (min , q1, median, q3, max)

In [24]:
```
1
```

```
q1 =  34.25
q2 =  65.0
```

In [31]:
```
q1,q3 = np.percentile(age,[25,75])
print("q1 = ",q1)
print("q3 = ",q3)

iqr = q3 - q1

lower_fence = q1 - 1.5*(iqr)
higher_fence =  q3 + 1.5*(iqr)
md = np.median(age)


print("Lower Fence = ",lower_fence)
print("median is = ",md)
print("Lower Fence = ",higher_fence)
```

```
q1 =  34.25
q3 =  65.0
Lower Fence =  -11.875
median is =  45.0
Lower Fence =  111.125
```

In [38]:
```
data = [23,34,45,34,45,67,45,34,56,43,23,35,65,44,65,120,130,123]
data
```

Out[38]:
```
[23, 34, 45, 34, 45, 67, 45, 34, 56, 43, 23, 35, 65, 44, 65, 120, 130, 123]
```

In [52]:
```
# find varience using python


def varience(data):
    n = len(data)
    mean = sum(data)/n
    dev = [(x-mean)**2 for x in data]
    var = sum(dev)/n
    print("varience of data : ",var)

varience(data)
```

```
varience of data :  1058.9783950617286
```

In [60]:
```
print(np.std(data))
print(np.sqrt(var))
```

```
32.5419482370329
32.5419482370329
```

In [59]:
```
import statistics
print("the Standered deviation by statistics   : ",statistics.variance(dat

```

```
the Standered deviation by statistics   :  1121.2712418300653
```