```
import pandas as pd
df=pd.read_csv("/content/Data_set.csv")
```

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 100 entries, 0 to 99
Data columns (total 9 columns):
 #   Column                    Non-Null Count  Dtype  
---  ------                    --------------  -----  
 0   show_name                 96 non-null     object 
 1   country                   100 non-null    object 
 2   num_episodes              100 non-null    int64  
 3   aired_on                  99 non-null     object 
 4   original_network          99 non-null     object 
 5   rating                    96 non-null     float64
 6   current_overall_rank      97 non-null     float64
 7   lifetime_popularity_rank  100 non-null    int64  
 8   watchers                  97 non-null     float64
dtypes: float64(3), int64(2), object(4)
memory usage: 7.2+ KB
```

```
df.isnull()
```

	show_name	country	num_episodes	aired_on	original_network	rating	current_overall_rank	lifetime_popularity_rank	watchers
0	True	False	False	False	False	False	False	False	False
1	True	False	False	False	False	False	False	False	False
2	False	False	False	False	False	False	False	False	False
3	False	False	False	False	False	False	False	False	False
4	False	False	False	False	False	False	False	False	False
...	...	...	...	...	...	...	...	...	...
95	False	False	False	False	False	False	False	False	False
96	False	False	False	False	False	False	False	False	False
97	False	False	False	False	False	False	False	False	True
98	False	False	False	False	False	False	False	False	False
99	False	False	False	False	False	False	False	False	False

100 rows × 9 columns

```
df.isnull().sum()
```

	0
show_name	4
country	0
num_episodes	0
aired_on	1
original_network	1
rating	4
current_overall_rank	3
lifetime_popularity_rank	0
watchers	3

dtype: int64

```
df.fillna(0)
```

	show_name	country	num_episodes	aired_on	original_network	rating	current_overall_rank	lifetime_popularity_rank	watch
0	0	South Korea	16	Friday, Saturday	tvN	8.9	33.0	1	1117(
1	0	South Korea	16	Friday, Saturday	jTBC	8.7	89.0	2	1009
2	Descendants of the Sun	South Korea	16	Wednesday, Thursday	KBS2	8.7	77.0	3	963
3	Boys Over Flowers	South Korea	25	Monday, Tuesday	KBS2	7.7	2249.0	4	942
4	W	South Korea	16	Wednesday, Thursday	MBC	8.5	201.0	5	921
...	...	...	...	...	...	...	...	...	
95	Shut Up: Flower Boy Band	South Korea	16	Monday, Tuesday	tvN	8.1	806.0	99	346(
96	Blood	South	20	Monday,	KBS2	7.4	3271.0	100	346(

```
df.fillna(method='ffill')
```

```
<ipython-input-8-5c0beae7dc1e>:1: FutureWarning: DataFrame.fillna with 'method' is deprecated and will raise in a future version. Us
  df.fillna(method='ffill')
```

	show_name	country	num_episodes	aired_on	original_network	rating	current_overall_rank	lifetime_popularity_rank	watch
0	NaN	South Korea	16	Friday, Saturday	tvN	8.9	33.0	1	1117(
1	NaN	South Korea	16	Friday, Saturday	jTBC	8.7	89.0	2	1009
2	Descendants of the Sun	South Korea	16	Wednesday, Thursday	KBS2	8.7	77.0	3	963
3	Boys Over Flowers	South Korea	25	Monday, Tuesday	KBS2	7.7	2249.0	4	942
4	W	South Korea	16	Wednesday, Thursday	MBC	8.5	201.0	5	921
...	...	...	...	...	...	...	...	...	
95	Shut Up: Flower Boy Band	South Korea	16	Monday, Tuesday	tvN	8.1	806.0	99	346(
96	Blood	South	20	Monday,	KBS2	7.4	3271.0	100	346(

```
df.fillna(method='bfill')
```

```
<ipython-input-9-b823574c06e2>:1: FutureWarning: DataFrame.fillna with 'method' is deprecated and will raise in a future version. Us
  df.fillna(method='bfill')
```

	show_name	country	num_episodes	aired_on	original_network	rating	current_overall_rank	lifetime_popularity_rank	watch
0	Descendants of the Sun	South Korea	16	Friday, Saturday	tvN	8.9	33.0	1	1117(
1	Descendants of the Sun	South Korea	16	Friday, Saturday	jTBC	8.7	89.0	2	1009
2	Descendants of the Sun	South Korea	16	Wednesday, Thursday	KBS2	8.7	77.0	3	963
3	Boys Over Flowers	South Korea	25	Monday, Tuesday	KBS2	7.7	2249.0	4	942
4	W	South Korea	16	Wednesday, Thursday	MBC	8.5	201.0	5	921
...	...	...	...	...	...	...	...	...	
95	Shut Up: Flower Boy Band	South Korea	16	Monday, Tuesday	tvN	8.1	806.0	99	346(
96	Blood	South	20	Monday,	KBS2	7.4	3271.0	100	346(

```
df['watchers'].fillna(value=df['watchers'].mean())
```

	watchers
0	111706.000000
1	100950.000000
2	96318.000000
3	94228.000000
4	92121.000000
...	...
95	34668.000000
96	34666.000000
97	52994.907216
98	34615.000000
99	34523.000000

100 rows × 1 columns

dtype: float64

```
df.dropna(axis=0)
```

	show_name	country	num_episodes	aired_on	original_network	rating	current_overall_rank	lifetime_popularity_rank	watch
2	Descendants of the Sun	South Korea	16	Wednesday, Thursday	KBS2	8.7	77.0	3	963
3	Boys Over Flowers	South Korea	25	Monday, Tuesday	KBS2	7.7	2249.0	4	942
4	W	South Korea	16	Wednesday, Thursday	MBC	8.5	201.0	5	921
5	You Who Came from the Stars	South Korea	21	Wednesday, Thursday	SBS	8.6	112.0	6	913
6	Weightlifting Fairy Kim Bok Joo	South Korea	16	Wednesday, Thursday	MBC	8.8	40.0	7	913
...	...	...	...	...	...	...	...	...	
94	Flower of Evil	South Korea	16	Wednesday, Thursday	tvN	9.1	4.0	98	349

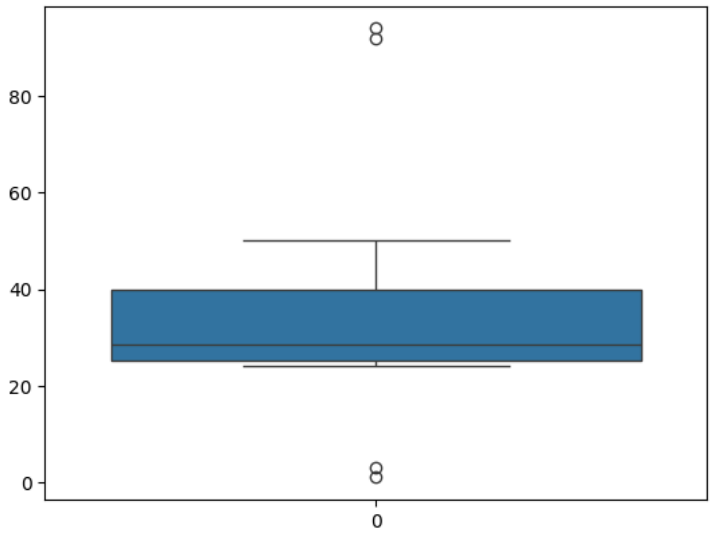
outlier Detection

```
import pandas as pd
import seaborn as sns
age=[1,3,28,27,25,92,30,39,40,50,26,24,29,94]
af=pd.DataFrame(age)
af
```

	0
0	1
1	3
2	28
3	27
4	25
5	92
6	30
7	39
8	40
9	50
10	26
11	24
12	29
13	94

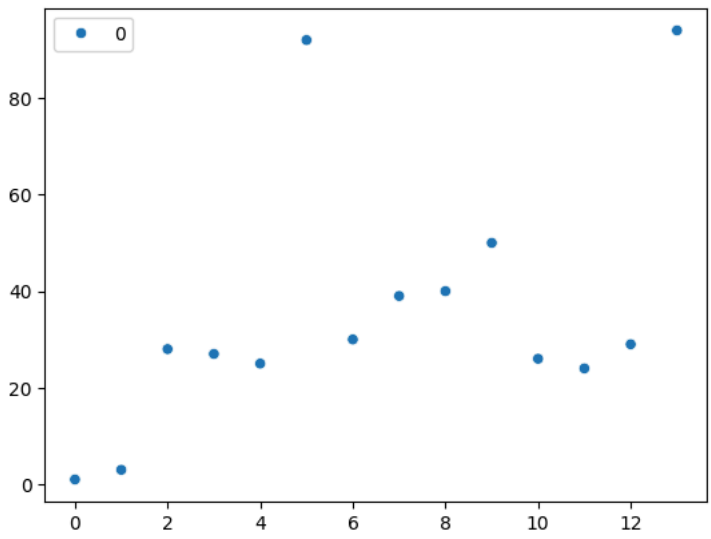
```
sns.boxplot(data=af)
```

```
<Axes: >
```




```
sns.scatterplot(data=af)
```

```
<Axes: >
```




```
q1=af.quantile(0.25)
q3=af.quantile(0.75)
```

```
iqr=q3-q1
iqr
```

	0
0	14.5

dtype: float64

```
low=q1-1.5*iqr
low
```

	0
0	3.5

dtype: float64

```
hight=q3+1.5*iqr
hight
```

	0
0	61.5

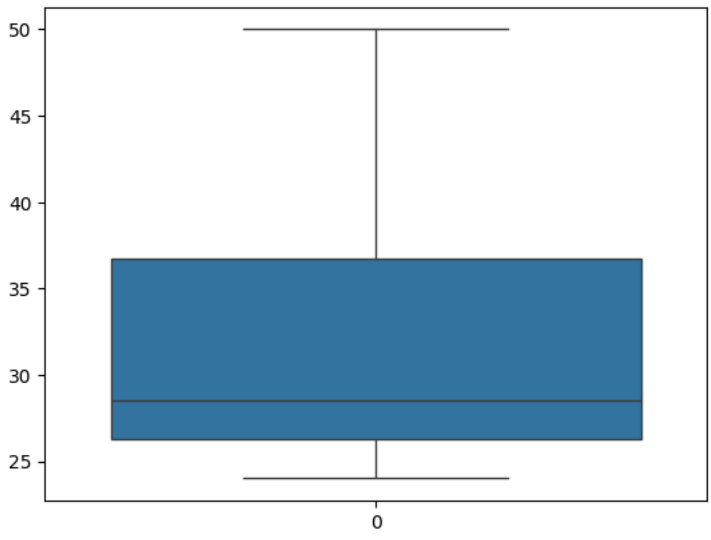
dtype: float64

```
af=af[((af>=low)&(af<=high))]
af.dropna()
```

	0
2	28.0
3	27.0
4	25.0
6	30.0
7	39.0
8	40.0
9	50.0
10	26.0
11	24.0
12	29.0

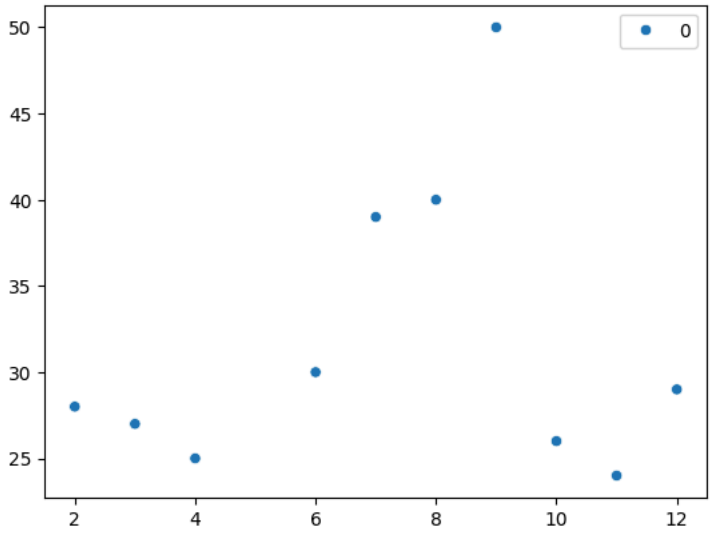
```
sns.boxplot(data=af)
```

```
<Axes: >
```




```
sns.scatterplot(data=af)
```

<Axes: >



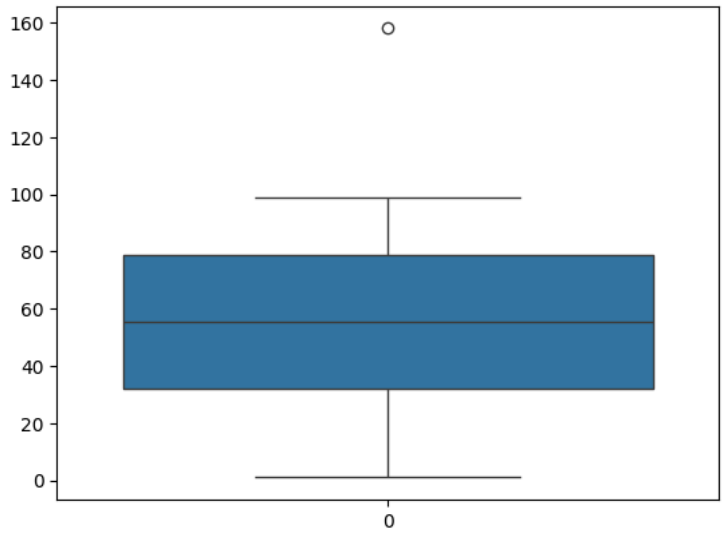

```
from scipy import stats
import numpy as np


data = [1, 12, 15, 18, 21, 24, 27, 30, 33, 36, 39, 42, 45, 48, 51, 54, 57, 60, 63, 66, 69, 72, 75, 78, 81, 84, 87, 90, 93, 96, 99, 158]
df = pd.DataFrame(data, columns=['Values'])
df
```

	Values
0	1
1	12
2	15
3	18
4	21
5	24
6	27
7	30
8	33
9	36
10	39
11	42
12	45
13	48
14	51
15	54
16	57
17	60
18	63
19	66
20	69
21	72
22	75
23	78
24	81
25	84
26	87
27	90
28	93
29	96
30	99
31	158

```
sns.boxplot(data=df)
```

```
<Axes: >
```



```
z_scores = stats.zscore(df['Values'])
threshold = 3
outliers_indices = np.where(np.abs(z_scores) > threshold)
outliers_df = df.iloc[outliers_indices]
print(outliers_df)
```

```
    Values
31     158
```

```
df_no_outliers = df[np.abs(z_scores) <= threshold]
df_no_outliers
```

	Values
0	1
1	12
2	15
3	18
4	21
5	24
6	27
7	30
8	33
9	36
10	39
11	42
12	45
13	48
14	51
15	54
16	57
17	60
18	63
19	66