In [1]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
```

In [3]:
```
cars=pd.read_csv("Q7.csv")
cars
```

Out[3]:

	Unnamed: 0	Points	Score	Weigh
0	Mazda RX4	3.90	2.620	16.46
1	Mazda RX4 Wag	3.90	2.875	17.02
2	Datsun 710	3.85	2.320	18.61
3	Hornet 4 Drive	3.08	3.215	19.44
4	Hornet Sportabout	3.15	3.440	17.02
5	Valiant	2.76	3.460	20.22
6	Duster 360	3.21	3.570	15.84
7	Merc 240D	3.69	3.190	20.00
8	Merc 230	3.92	3.150	22.90
9	Merc 280	3.92	3.440	18.30
10	Merc 280C	3.92	3.440	18.90
11	Merc 450SE	3.07	4.070	17.40
12	Merc 450SL	3.07	3.730	17.60
13	Merc 450SLC	3.07	3.780	18.00
14	Cadillac Fleetwood	2.93	5.250	17.98
15	Lincoln Continental	3.00	5.424	17.82
16	Chrysler Imperial	3.23	5.345	17.42
17	Fiat 128	4.08	2.200	19.47
18	Honda Civic	4.93	1.615	18.52
19	Toyota Corolla	4.22	1.835	19.90
20	Toyota Corona	3.70	2.465	20.01
21	Dodge Challenger	2.76	3.520	16.87
22	AMC Javelin	3.15	3.435	17.30
23	Camaro Z28	3.73	3.840	15.41
24	Pontiac Firebird	3.08	3.845	17.05
25	Fiat X1-9	4.08	1.935	18.90
26	Porsche 914-2	4.43	2.140	16.70
27	Lotus Europa	3.77	1.513	16.90
28	Ford Pantera L	4.22	3.170	14.50
29	Ferrari Dino	3.62	2.770	15.50
30	Maserati Bora	3.54	3.570	14.60
31	Volvo 142E	4.11	2.780	18.60

In [4]:
```
# mean
cars.mean()
```

Out[4]:
```
Points     3.596563
Score      3.217250
Weigh     17.848750
dtype: float64
```

In [10]:
```
# median
cars.median()
```

Out[10]:
```
Points     3.695
Score      3.325
Weigh     17.710
dtype: float64
```

In [11]:
```
# Varience
cars.var()
```

Out[11]:
```
Points    0.285881
Score     0.957379
Weigh     3.193166
dtype: float64
```

In [12]:
```
# Standar Deviation
cars.std()
```

Out[12]:
```
Points    0.534679
Score     0.978457
Weigh     1.786943
dtype: float64
```

In [8]:
```
# mode
cars.mode()
```

Out[8]:

	Unnamed: 0	Points	Score	Weigh
0	AMC Javelin	3.07	3.44	17.02
1	Cadillac Fleetwood	3.92	NaN	18.90
2	Camaro Z28	NaN	NaN	NaN
3	Chrysler Imperial	NaN	NaN	NaN
4	Datsun 710	NaN	NaN	NaN
5	Dodge Challenger	NaN	NaN	NaN
6	Duster 360	NaN	NaN	NaN
7	Ferrari Dino	NaN	NaN	NaN
8	Fiat 128	NaN	NaN	NaN
9	Fiat X1-9	NaN	NaN	NaN
10	Ford Pantera L	NaN	NaN	NaN
11	Honda Civic	NaN	NaN	NaN
12	Hornet 4 Drive	NaN	NaN	NaN
13	Hornet Sportabout	NaN	NaN	NaN
14	Lincoln Continental	NaN	NaN	NaN
15	Lotus Europa	NaN	NaN	NaN
16	Maserati Bora	NaN	NaN	NaN
17	Mazda RX4	NaN	NaN	NaN
18	Mazda RX4 Wag	NaN	NaN	NaN
19	Merc 230	NaN	NaN	NaN
20	Merc 240D	NaN	NaN	NaN
21	Merc 280	NaN	NaN	NaN
22	Merc 280C	NaN	NaN	NaN
23	Merc 450SE	NaN	NaN	NaN
24	Merc 450SL	NaN	NaN	NaN
25	Merc 450SLC	NaN	NaN	NaN
26	Pontiac Firebird	NaN	NaN	NaN
27	Porsche 914-2	NaN	NaN	NaN
28	Toyota Corolla	NaN	NaN	NaN
29	Toyota Corona	NaN	NaN	NaN
30	Valiant	NaN	NaN	NaN
31	Volvo 142E	NaN	NaN	NaN

In [13]:
```
# Range
```

In [14]:
```
cars.describe()
```

Out[14]:

	Points	Score	Weigh
count	32.000000	32.000000	32.000000
mean	3.596563	3.217250	17.848750
std	0.534679	0.978457	1.786943
min	2.760000	1.513000	14.500000
25%	3.080000	2.581250	16.892500
50%	3.695000	3.325000	17.710000
75%	3.920000	3.610000	18.900000
max	4.930000	5.424000	22.900000

In [15]:
```
Points_Range=cars.Points.max()-cars.Points.min()
Points_Range
```

Out[15]:
```
2.17
```

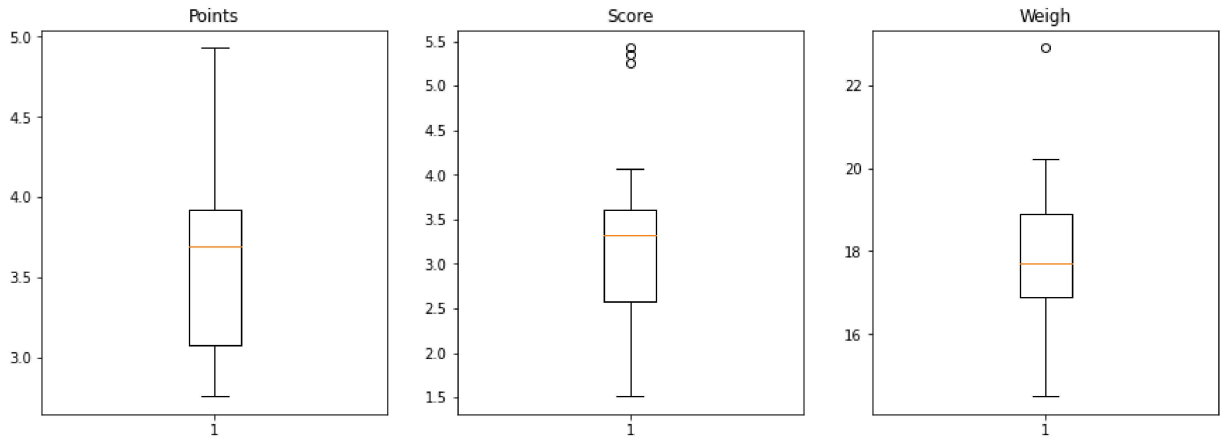
In [16]:
```
Score_Range=cars.Score.max()-cars.Score.min()
Score_Range
```

Out[16]:
```
3.9110000000000005
```

In [17]:
```
Weigh_Range=cars.Weigh.max()-cars.Weigh.min()
Weigh_Range
```

Out[17]:
```
8.399999999999999
```

In [18]:
```
f,ax=plt.subplots(figsize=(15,5))
plt.subplot(1,3,1)
plt.boxplot(cars.Points)
plt.title('Points')
plt.subplot(1,3,2)
plt.boxplot(cars.Score)
plt.title('Score')
plt.subplot(1,3,3)
plt.boxplot(cars.Weigh)
plt.title('Weigh')
plt.show()
```



In [ ]: