Car Seat Sales

In [15]:
```
# Import package pandas
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
```

In [16]:
```
# Introduction

# Import dataset
df = pd.read_csv("Carseats.csv")
df.head(10)
```

Out[16]:

	Unnamed: 0	Sales	CompPrice	Income	Advertising	Population	Price	ShelveLoc	Age	Educ
0	1	9.50	138	73	11	276	120	Bad	42	
1	2	11.22	111	48	16	260	83	Good	65	
2	3	10.06	113	35	10	269	80	Medium	59	
3	4	7.40	117	100	4	466	97	Medium	55	
4	5	4.15	141	64	3	340	128	Bad	38	
5	6	10.81	124	113	13	501	72	Bad	78	
6	7	6.63	115	105	0	45	108	Medium	71	
7	8	11.85	136	81	15	425	120	Good	67	
8	9	6.54	132	110	0	108	124	Medium	76	
9	10	4.69	132	113	0	131	124	Medium	76	

In [ ]:
```
The data set 'Carseats' was obtained from https://vincentarelbundock.github.io
/Rdatasets/datasets.html.
    The dataset contains 400 observations across 11 variables.
    The data set focuses on child car seat sales among 400 different stores us
ing simulated data.
```

In [13]:
```
# EDA

# mean of sales
np.mean(df.Sales)
```

Out[13]:
```
7.496325000000001
```

In [14]:
```
# standard deviation of sales
np.std(df.Sales)
```

Out[14]:
```
2.820582695893705
```

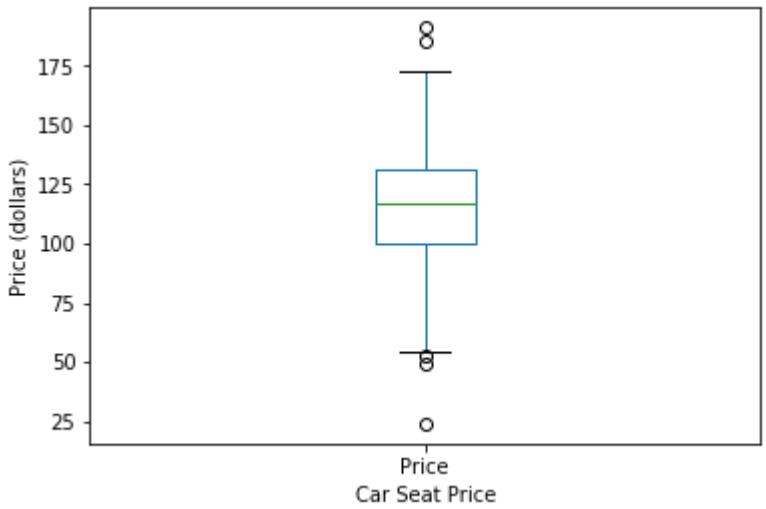
In [ ]:
```
The mean of the car seat unit sales for each location was calculated to be 7.4
96 (thousands of dollars) with a standard deviation of 2.821 (thousands of dol
lars).
```

In [19]:
```
# Visualization

# Numeric variable - box plot
df['Price'].plot(kind = "box")
plt.xlabel('Car Seat Price')
plt.ylabel("Price (dollars)")


The box plot below demonstrates an average unit price around 115 dollars.
There are a few outliers above 175 dollars and around 25 dollars on both ends
of the spectrum.
```

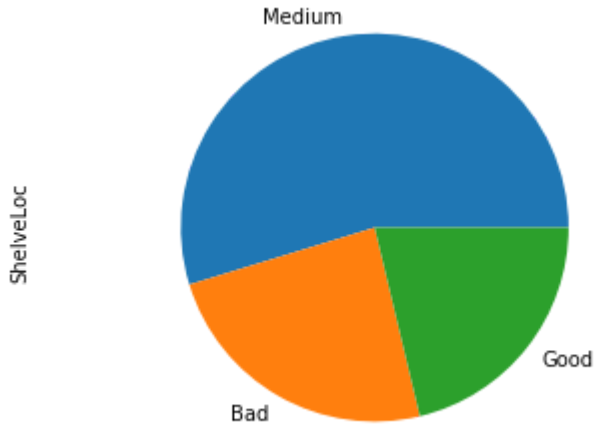
Out[19]:
```
Text(0,0.5,'Price (dollars)')
```



In [20]:
```
# Categorical variable - pie chart
df['ShelveLoc'].value_counts() \
.plot(kind = "pie") \
.axis('equal')

The pie chart below illustrates the ratio of the different qualities of shelvi
ng location of the car seats.
As depicted, the "Medium" quality is most popular among the different location
s.
The other two categories appear to be roughly even in comparison.
```

Out[20]:
```
(-1.1107450160226942,
 1.1005116894570615,
 -1.1338269874777946,
 1.1123784856777597)
```



In [ ]: