In [262]:
```
import pandas as pd
```

In [263]:
```
# 2 main datatypes
series= pd.Series(['bmw','toyota','honda'])
```

In [264]:
```
series
```

Out[264]:
```
0       bmw
1    toyota
2     honda
dtype: object
```

In [265]:
```
#series = 1-dimensional
```

In [266]:
```
colours=pd.Series(['red','blue','white'])
```

In [267]:
```
colours
```

Out[267]:
```
0      red
1     blue
2    white
dtype: object
```

In [268]:
```
#dataFrame= 2-dimensional
car_data = pd.DataFrame({'car':series,'color':colours})
```

In [269]:
```
car_data
```

Out[269]:

	car	color
0	bmw	red
1	toyota	blue
2	honda	white

In [270]:
```
#import data
heart=pd.read_csv('heart.csv')
```

In [271]:

```
heart
```

Out[271]:

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
0	63	1	3	145	233	1	0	150	0	2.3	0	0	1	1
1	37	1	2	130	250	0	1	187	0	3.5	0	0	2	1
2	41	0	1	130	204	0	0	172	0	1.4	2	0	2	1
3	56	1	1	120	236	0	1	178	0	0.8	2	0	2	1
4	57	0	0	120	354	0	1	163	1	0.6	2	0	2	1
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
298	57	0	0	140	241	0	1	123	1	0.2	1	0	3	0
299	45	1	3	110	264	0	1	132	0	1.2	1	0	3	0
300	68	1	0	144	193	1	1	141	0	3.4	1	2	3	0
301	57	1	0	130	131	0	1	115	1	1.2	1	1	3	0
302	57	0	1	130	236	0	0	174	0	0.0	1	1	2	0

303 rows × 14 columns

In [272]:

```
#exporting data fram
heart.to_csv('exported-car-sales.csv',index=False)
```

In [273]:

```
export_car = pd.read_csv('exported-car-sales.csv')
```

In [274]:

```
export_car
```

Out[274]:

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
0	63	1	3	145	233	1	0	150	0	2.3	0	0	1	1
1	37	1	2	130	250	0	1	187	0	3.5	0	0	2	1
2	41	0	1	130	204	0	0	172	0	1.4	2	0	2	1
3	56	1	1	120	236	0	1	178	0	0.8	2	0	2	1
4	57	0	0	120	354	0	1	163	1	0.6	2	0	2	1
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
298	57	0	0	140	241	0	1	123	1	0.2	1	0	3	0
299	45	1	3	110	264	0	1	132	0	1.2	1	0	3	0
300	68	1	0	144	193	1	1	141	0	3.4	1	2	3	0
301	57	1	0	130	131	0	1	115	1	1.2	1	1	3	0
302	57	0	1	130	236	0	0	174	0	0.0	1	1	2	0

303 rows × 14 columns

In [275]:

```
car_sales=pd.read_csv('Car_sales.csv')
car_sales
```

Out[275]:

	Manufacturer	Model	Sales_in_thousands	__year_resale_value	Vehicle_type	Price_in_thousan
0	Acura	Integra	854.000	16.360	Passenger	21
1	Acura	TL	39.384	19.875	Passenger	28
2	Acura	CL	14.114	18.225	Passenger	N
3	Acura	RL	8.588	29.725	Passenger	42
4	Audi	A4	20.397	22.255	Passenger	23
...	...	...	...	...	...	
152	Volvo	V40	3.545	NaN	Passenger	24
153	Volvo	S70	15.245	NaN	Passenger	27
154	Volvo	V70	17.531	NaN	Passenger	28
155	Volvo	C70	3.493	NaN	Passenger	45
156	Volvo	S80	18.969	NaN	Passenger	36

157 rows × 16 columns

describe data

In [276]:

```
## attribut //check data type
car_sales.dtypes
```

Out[276]:

```
Manufacturer            object
Model                   object
Sales_in_thousands     float64
__year_resale_value    float64
Vehicle_type            object
Price_in_thousands     float64
Engine_size            float64
Horsepower             float64
Wheelbase              float64
Width                  float64
Length                 float64
Curb_weight            float64
Fuel_capacity          float64
Fuel_efficiency        float64
Latest_Launch           object
Power_perf_factor      float64
dtype: object
```

In [277]:
```
car_columns=car_sales.columns
car_columns
```

Out[277]:
```
Index(['Manufacturer', 'Model', 'Sales_in_thousands', '__year_resale_value',
       'Vehicle_type', 'Price_in_thousands', 'Engine_size', 'Horsepower',
       'Wheelbase', 'Width', 'Length', 'Curb_weight', 'Fuel_capacity',
       'Fuel_efficiency', 'Latest_Launch', 'Power_perf_factor'],
      dtype='object')
```

In [ ]:

In [278]:
```
# check index of file
car_sales.index
```

Out[278]:
```
RangeIndex(start=0, stop=157, step=1)
```

In [279]:
```
car_sales
```

0	Acura	Integra	854.000	16.360	Passenger
1	Acura	TL	39.384	19.875	Passenger
2	Acura	CL	14.114	18.225	Passenger
3	Acura	RL	8.588	29.725	Passenger
4	Audi	A4	20.397	22.255	Passenger
...	...	...	...	...	...
152	Volvo	V40	3.545	NaN	Passenger
153	Volvo	S70	15.245	NaN	Passenger
154	Volvo	V70	17.531	NaN	Passenger
155	Volvo	C70	3.493	NaN	Passenger
156	Volvo	S80	18.969	NaN	Passenger

157 rows × 16 columns

In [280]:

```
# Function
car_sales.describe()
```

Out[280]:

	Sales_in_thousands	__year_resale_value	Price_in_thousands	Engine_size	Horsepower	Wh
count	157.000000	122.000000	155.000000	156.000000	156.000000	156
mean	63.926726	21.326475	27.390755	3.060897	185.948718	107
std	103.138664	37.702759	14.351653	1.044653	56.700321	7
min	0.110000	5.160000	9.235000	1.000000	55.000000	92
25%	14.351000	11.268750	18.017500	2.300000	149.500000	103
50%	31.148000	14.195000	22.799000	3.000000	177.500000	107
75%	71.186000	19.886250	31.947500	3.575000	215.000000	112
max	854.000000	415.000000	85.500000	8.000000	450.000000	138

In [281]:

```
car_sales.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 157 entries, 0 to 156
Data columns (total 16 columns):
 #   Column               Non-Null Count  Dtype  
---  ------               --------------  -----  
 0   Manufacturer         157 non-null    object 
 1   Model                157 non-null    object 
 2   Sales_in_thousands   157 non-null    float64
 3   __year_resale_value  122 non-null    float64
 4   Vehicle_type         157 non-null    object 
 5   Price_in_thousands   155 non-null    float64
 6   Engine_size          156 non-null    float64
 7   Horsepower           156 non-null    float64
 8   Wheelbase            156 non-null    float64
 9   Width                156 non-null    float64
 10  Length               156 non-null    float64
 11  Curb_weight          155 non-null    float64
 12  Fuel_capacity        156 non-null    float64
 13  Fuel_efficiency      154 non-null    float64
 14  Latest_Launch        157 non-null    object 
 15  Power_perf_factor    155 non-null    float64
dtypes: float64(12), object(4)
memory usage: 19.8+ KB
```

In [282]:
```
car_sales.mean()
```

Out[282]:
```
Sales_in_thousands       63.926726
__year_resale_value      21.326475
Price_in_thousands       27.390755
Engine_size               3.060897
Horsepower              185.948718
Wheelbase               107.487179
Width                    71.150000
Length                  187.343590
Curb_weight               3.378026
Fuel_capacity            17.951923
Fuel_efficiency          23.844156
Power_perf_factor        77.043591
dtype: float64
```

In [283]:
```
car_prices = pd.Series([200,400,600])
```

In [284]:
```
car_sales.sum()
```

Out[284]:
```
Manufacturer           AcuraAcuraAcuraAcuraAudiAudiAudiBMWBMWBMWBuick...
Model                  IntegraTLCLRLA4A6A8323i328i528iCenturyRegalPar...
Sales_in_thousands                                               10036.5
__year_resale_value                                              2601.83
Vehicle_type           PassengerPassengerPassengerPassengerPassengerP...
Price_in_thousands                                               4245.57
Engine_size                                                        477.5
Horsepower                                                         29008
Wheelbase                                                          16768
Width                                                            11099.4
Length                                                           29225.6
Curb_weight                                                      523.594
Fuel_capacity                                                     2800.5
Fuel_efficiency                                                     3672
Latest_Launch          02-02-201206-03-201101-04-201203-10-201110-08-...
Power_perf_factor                                                11941.8
dtype: object
```

In [285]:
```
car_sales['Price_in_thousands'].dtype == 'float64' or 'int64'
```

Out[285]:
```
True
```

In [286]:
```
len(car_sales)
```

Out[286]:
```
157
```

In [287]:
```
car_data.color.dtypes
```

Out[287]:
```
dtype('O')
```

viewing and selection data

In [288]:
```
car_sales.head()
```

Out[288]:

	Manufacturer	Model	Sales_in_thousands	__year_resale_value	Vehicle_type	Price_in_thousands
0	Acura	Integra	854.000	16.360	Passenger	21.50
1	Acura	TL	39.384	19.875	Passenger	28.40
2	Acura	CL	14.114	18.225	Passenger	NaN
3	Acura	RL	8.588	29.725	Passenger	42.00
4	Audi	A4	20.397	22.255	Passenger	23.99

In [289]:
```
car_sales.tail()
```

Out[289]:

	Manufacturer	Model	Sales_in_thousands	__year_resale_value	Vehicle_type	Price_in_thousan
152	Volvo	V40	3.545	NaN	Passenger	24
153	Volvo	S70	15.245	NaN	Passenger	27
154	Volvo	V70	17.531	NaN	Passenger	28
155	Volvo	C70	3.493	NaN	Passenger	45
156	Volvo	S80	18.969	NaN	Passenger	36

In [290]:
```
# .loc and .iloc
animal =pd.Series(['cat','dog','bird','panda','snake'],index=[0,3,5,7,3])
```

In [291]:
```
animal
```

Out[291]:
```
0      cat
3      dog
5     bird
7    panda
3    snake
dtype: object
```

In [292]:
```
animal[3]
```

Out[292]:
```
3      dog
3    snake
dtype: object
```

In [293]:
```
# .loc // prefer to index
animal.loc[3]
```

Out[293]:
```
3      dog
3    snake
dtype: object
```

In [294]:
```
#.iloc // prefer to position
animal.iloc[3]
```

Out[294]:
```
'panda'
```

In [295]:
```
animal.iloc[:3]
```

Out[295]:
```
0     cat
3     dog
5    bird
dtype: object
```

In [296]:
```
car_sales['Manufacturer']
```

Out[296]:
```
0      Acura
1      Acura
2      Acura
3      Acura
4       Audi
       ...
152    Volvo
153    Volvo
154    Volvo
155    Volvo
156    Volvo
Name: Manufacturer, Length: 157, dtype: object
```

In [297]:
```
#both are same only syntax change
car_sales['Model']  # use for  space in name
car_sales.Model
```

Out[297]:
```
0      Integra
1           TL
2           CL
3           RL
4           A4
        ...
152        V40
153        S70
154        V70
155        C70
156        S80
Name: Model, Length: 157, dtype: object
```

In [298]:

```
car_sales.Model
```

Out[298]:

```
0      Integra
1           TL
2           CL
3           RL
4           A4
        ...   
152        V40
153        S70
154        V70
155        C70
156        S80
Name: Model, Length: 157, dtype: object
```

In [299]:

```
car_sales[car_sales["Horsepower"] == 'BMW']
```

Out[299]:

Manufacturer	Model	Sales_in_thousands	__year_resale_value	Vehicle_type	Price_in_thousands

In [300]:

```
car_sales[car_sales["Horsepower"] >= 170]
```

Out[300]:

	Manufacturer	Model	Sales_in_thousands	__year_resale_value	Vehicle_type	Price_in_thousar
1	Acura	TL	39.384	19.875	Passenger	28.4
2	Acura	CL	14.114	18.225	Passenger	N
3	Acura	RL	8.588	29.725	Passenger	42.0
5	Audi	A6	18.780	23.555	Passenger	33.9
6	Audi	A8	1.380	39.000	Passenger	62.0
...	...	...	...	...	...	
138	Toyota	Avalon	63.849	18.140	Passenger	25.5
141	Toyota	Sienna	65.119	NaN	Car	22.3
144	Toyota	Land Cruiser	9.835	34.080	Car	51.7
155	Volvo	C70	3.493	NaN	Passenger	45.5
156	Volvo	S80	18.969	NaN	Passenger	36.0

93 rows × 16 columns

In [301]:
```
#crosstab // for comparing column
pd.crosstab(car_sales['Model'],car_sales['Manufacturer'])
```

Out[301]:

Manufacturer	Acura	Audi	BMW	Buick	Cadillac	Chevrolet	Chrysler	Dodge	Ford	Honda	...
Model											
03-Sep	0	0	0	0	0	0	0	0	0	0	...
05-Sep	0	0	0	0	0	0	0	0	0	0	...
3000GT	0	0	0	0	0	0	0	0	0	0	...
300M	0	0	0	0	0	0	1	0	0	0	...
323i	0	0	1	0	0	0	0	0	0	0	...
...	...	...	...	...	...	...	...	...	...	...	...
Viper	0	0	0	0	0	0	0	1	0	0	...
Voyager	0	0	0	0	0	0	0	0	0	0	...
Windstar	0	0	0	0	0	0	0	0	1	0	...
Wrangler	0	0	0	0	0	0	0	0	0	0	...
Xterra	0	0	0	0	0	0	0	0	0	0	...

156 rows × 30 columns

In [302]:
```
# groupby
car_sales.groupby(['Manufacturer']).mean()
```

Mitsubishi	25.842143	14.262143	22.167429	2.800000	165.14
Nissan	57.090714	13.612000	22.360714	2.914286	169.00
Oldsmobile	29.831500	16.573750	25.622833	3.450000	190.00
Plymouth	16.000250	9.858333	22.642500	2.475000	166.75
Pontiac	61.755667	13.824000	22.949167	3.433333	185.00
Porsche	4.042667	56.475000	62.473333	3.166667	272.33
Saab	10.653000	NaN	29.610000	2.150000	177.50
Saturn	33.770000	10.193333	14.271000	2.020000	119.60
Subaru	40.067500	NaN	21.395000	2.500000	165.00
Toyota	82.245000	16.657500	21.982889	2.622222	160.66
Volkswagen	34.868667	14.345000	17.698333	1.966667	120.83
Volvo	12.623333	NaN	30.933333	2.300000	182.16

In [ ]:

```

```

In [ ]:

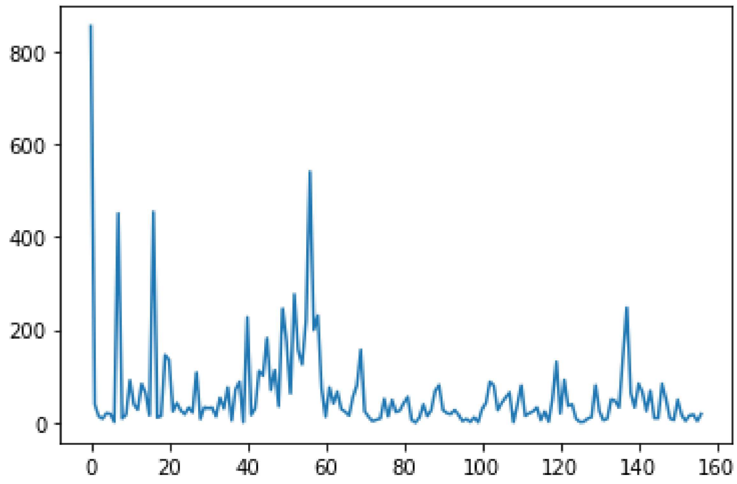
```

```

In [303]:

```
car_sales['Sales_in_thousands'].plot()
# not show plot so use
# %matplotlib inline
#car_sales['Sales_in_thousands'].plot()
```

Out[303]:

```
<AxesSubplot:>
```

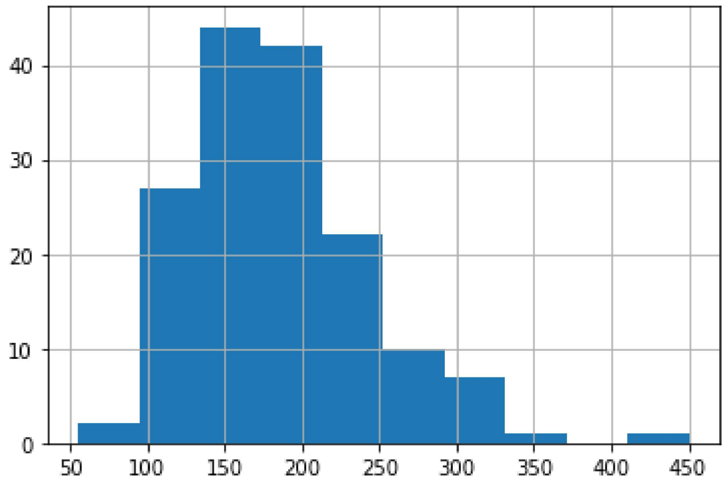


In [304]:

```
car_sales['Horsepower'].hist()
```

Out[304]:

```
<AxesSubplot:>
```

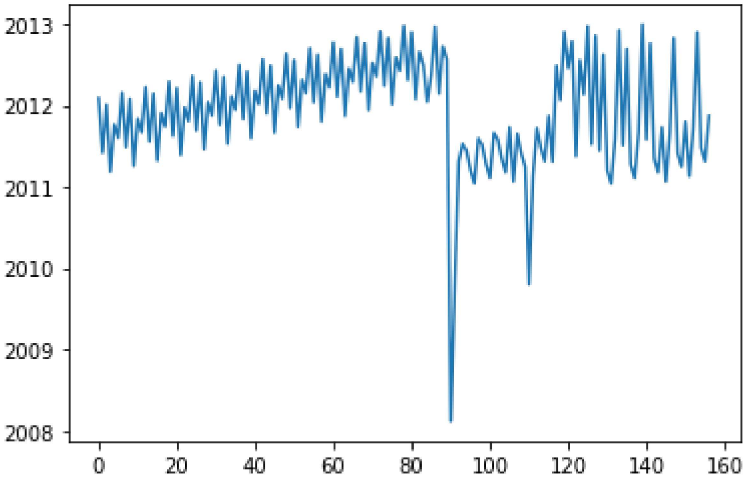


In [305]:

```
# convert object data to int for ploting
carConvert= pd.to_datetime(car_sales['Latest_Launch'])
```

In [306]:
```
carConvert
carConvert.plot()
```

Out[306]:
```
<AxesSubplot:>
```



Manipulating Data

In [307]:
```
car_sales
car_sales['Manufacturer'].str.lower()
```

Out[307]:
```
0      acura
1      acura
2      acura
3      acura
4       audi
       ...
152    volvo
153    volvo
154    volvo
155    volvo
156    volvo
Name: Manufacturer, Length: 157, dtype: object
```

In [308]:
```
car_sales['Manufacturer'] = car_sales['Manufacturer'].str.lower()
car_sales
```

Out[308]:

	Manufacturer	Model	Sales_in_thousands	__year_resale_value	Vehicle_type	Price_in_thousan
0	acura	Integra	854.000	16.360	Passenger	21
1	acura	TL	39.384	19.875	Passenger	28
2	acura	CL	14.114	18.225	Passenger	N
3	acura	RL	8.588	29.725	Passenger	42
4	audi	A4	20.397	22.255	Passenger	23
...	...	...	...	...	...	
152	volvo	V40	3.545	NaN	Passenger	24
153	volvo	S70	15.245	NaN	Passenger	27
154	volvo	V70	17.531	NaN	Passenger	28
155	volvo	C70	3.493	NaN	Passenger	45
156	volvo	S80	18.969	NaN	Passenger	36

157 rows × 16 columns

In [309]:
```
car_missing=pd.read_csv('Car_sales_Missing.csv')
```

In [310]:
```
car_missing
```

Out[310]:

	Manufacturer	Model	Sales_in_thousands	__year_resale_value	Vehicle_type	Price_in_thousan
0	Acura	Integra	NaN	16.360	Passenger	21
1	Acura	TL	39.384	19.875	Passenger	28
2	Acura	CL	14.114	18.225	Passenger	N
3	Acura	RL	8.588	29.725	Passenger	42
4	Audi	A4	20.397	22.255	Passenger	23
...	...	...	...	...	...	
152	Volvo	V40	3.545	NaN	Passenger	24
153	Volvo	S70	15.245	NaN	Passenger	27
154	Volvo	V70	17.531	NaN	Passenger	28
155	Volvo	C70	3.493	NaN	Passenger	45
156	Volvo	S80	18.969	NaN	Passenger	36

157 rows × 16 columns

In [311]:

```
# fill missing value(NAN)
car_missing['Sales_in_thousands'].fillna(car_missing['Sales_in_thousands'].mean()
```

In [312]:

```
car_missing
```

Out[312]:

	Manufacturer	Model	Sales_in_thousands	__year_resale_value	Vehicle_type	Price_in_thousan
0	Acura	Integra	55.977279	16.360	Passenger	21
1	Acura	TL	39.384000	19.875	Passenger	28
2	Acura	CL	14.114000	18.225	Passenger	N
3	Acura	RL	8.588000	29.725	Passenger	42
4	Audi	A4	20.397000	22.255	Passenger	23
...	...	...	...	...	...	
152	Volvo	V40	3.545000	NaN	Passenger	24
153	Volvo	S70	15.245000	NaN	Passenger	27
154	Volvo	V70	17.531000	NaN	Passenger	28
155	Volvo	C70	3.493000	NaN	Passenger	45
156	Volvo	S80	18.969000	NaN	Passenger	36

157 rows × 16 columns

In [313]:

```
#drop NAN value
car_missing.dropna()
```

Out[313]:

	Manufacturer	Model	Sales_in_thousands	__year_resale_value	Vehicle_type	Price_in_thousan
0	Acura	Integra	55.977279	16.360	Passenger	21
1	Acura	TL	39.384000	19.875	Passenger	28
3	Acura	RL	8.588000	29.725	Passenger	42
4	Audi	A4	20.397000	22.255	Passenger	23
5	Audi	A6	18.780000	23.555	Passenger	33
...	...	...	...	...	...	
145	Volkswagen	Golf	9.761000	11.425	Passenger	14
146	Volkswagen	Jetta	83.721000	13.240	Passenger	16
147	Volkswagen	Passat	51.102000	16.725	Passenger	21
148	Volkswagen	Cabrio	9.569000	16.575	Passenger	19
149	Volkswagen	GTI	5.596000	13.760	Passenger	17

118 rows × 16 columns

In [314]:

```
# Column  from series
seats_column = pd.Series([5,5,5,5])
#new Coloumn called seats
car_sales['Seats'] = seats_column
car_sales
```

Out[314]:

	Manufacturer	Model	Sales_in_thousands	__year_resale_value	Vehicle_type	Price_in_thousan
0	acura	Integra	854.000	16.360	Passenger	21
1	acura	TL	39.384	19.875	Passenger	28
2	acura	CL	14.114	18.225	Passenger	N
3	acura	RL	8.588	29.725	Passenger	42
4	audi	A4	20.397	22.255	Passenger	23
...	...	...	...	...	...	
152	volvo	V40	3.545	NaN	Passenger	24
153	volvo	S70	15.245	NaN	Passenger	27
154	volvo	V70	17.531	NaN	Passenger	28
155	volvo	C70	3.493	NaN	Passenger	45
156	volvo	S80	18.969	NaN	Passenger	36

157 rows × 17 columns

In [315]:

```
car_sales['Seats'].fillna(5,inplace=True)
```

In [316]:

```
car_sales
```

Out[316]:

	Manufacturer	Model	Sales_in_thousands	__year_resale_value	Vehicle_type	Price_in_thousan
0	acura	Integra	854.000	16.360	Passenger	21
1	acura	TL	39.384	19.875	Passenger	28
2	acura	CL	14.114	18.225	Passenger	N
3	acura	RL	8.588	29.725	Passenger	42
4	audi	A4	20.397	22.255	Passenger	23
...	...	...	...	...	...	
152	volvo	V40	3.545	NaN	Passenger	24
153	volvo	S70	15.245	NaN	Passenger	27
154	volvo	V70	17.531	NaN	Passenger	28
155	volvo	C70	3.493	NaN	Passenger	45
156	volvo	S80	18.969	NaN	Passenger	36

157 rows × 17 columns

In [317]:

```
car_sales['Fuel_efficiency(L)'] =car_sales['Fuel_efficiency']/2 * 100
car_sales
```

Out[317]:

	Manufacturer	Model	Sales_in_thousands	__year_resale_value	Vehicle_type	Price_in_thousan
0	acura	Integra	854.000	16.360	Passenger	21
1	acura	TL	39.384	19.875	Passenger	28
2	acura	CL	14.114	18.225	Passenger	N
3	acura	RL	8.588	29.725	Passenger	42
4	audi	A4	20.397	22.255	Passenger	23
...	...	...	...	...	...	
152	volvo	V40	3.545	NaN	Passenger	24
153	volvo	S70	15.245	NaN	Passenger	27
154	volvo	V70	17.531	NaN	Passenger	28
155	volvo	C70	3.493	NaN	Passenger	45
156	volvo	S80	18.969	NaN	Passenger	36

157 rows × 18 columns

In [318]:

```
# creating a column from a single value
car_sales['Number of wheels'] = 4
car_sales
```

Out[318]:

	Manufacturer	Model	Sales_in_thousands	__year_resale_value	Vehicle_type	Price_in_thousan
0	acura	Integra	854.000	16.360	Passenger	21
1	acura	TL	39.384	19.875	Passenger	28
2	acura	CL	14.114	18.225	Passenger	N
3	acura	RL	8.588	29.725	Passenger	42
4	audi	A4	20.397	22.255	Passenger	23
...	...	...	...	...	...	
152	volvo	V40	3.545	NaN	Passenger	24
153	volvo	S70	15.245	NaN	Passenger	27
154	volvo	V70	17.531	NaN	Passenger	28
155	volvo	C70	3.493	NaN	Passenger	45
156	volvo	S80	18.969	NaN	Passenger	36

157 rows × 19 columns

In [319]:

```
# drop a column
car_sales = car_sales.drop("Fuel_efficiency",axis=1)
car_sales
```

Out[319]:

	Manufacturer	Model	Sales_in_thousands	__year_resale_value	Vehicle_type	Price_in_thousan
0	acura	Integra	854.000	16.360	Passenger	21
1	acura	TL	39.384	19.875	Passenger	28
2	acura	CL	14.114	18.225	Passenger	N
3	acura	RL	8.588	29.725	Passenger	42
4	audi	A4	20.397	22.255	Passenger	23
...	...	...	...	...	...	
152	volvo	V40	3.545	NaN	Passenger	24
153	volvo	S70	15.245	NaN	Passenger	27
154	volvo	V70	17.531	NaN	Passenger	28
155	volvo	C70	3.493	NaN	Passenger	45
156	volvo	S80	18.969	NaN	Passenger	36

157 rows × 18 columns

In [ ]: