In [1]:

```
import pandas as pd
```

In [16]:

```
car_sales = pd.read_csv("data/car-sales.csv")
car_sales
```

Out[16]:

	Make	Colour	Odometer (KM)	Doors	Price
0	Toyota	White	150043	4	$4,000.00
1	Honda	Red	87899	4	$5,000.00
2	Toyota	Blue	32549	3	$7,000.00
3	BMW	Black	11179	5	$22,000.00
4	Nissan	White	213095	4	$3,500.00
5	Toyota	Green	99213	4	$4,500.00
6	Honda	Blue	45698	4	$7,500.00
7	Honda	Blue	54738	4	$7,000.00
8	Toyota	White	60000	4	$6,250.00
9	Nissan	White	31600	4	$9,700.00

In [4]:

```
# To get the data type of each column (String are considered as objects)
car_sales.dtypes
```

Out[4]:

```
Make              object
Colour            object
Odometer (KM)      int64
Doors              int64
Price             object
dtype: object
```

In [5]:

```
# To show the column names
car_columns = car_sales.columns
car_columns
```

Out[5]:

```
Index(['Make', 'Colour', 'Odometer (KM)', 'Doors', 'Price'], dtype='object')
```

In [6]:

```
# To get the range of index values (For rows)
car_sales.index
```

Out[6]:

```
RangeIndex(start=0, stop=10, step=1)
```

In [15]:

```
# To get count, mean, std, min , . . . of numeric columns
# Price is a string field. So It wont display
car_sales.describe()
```

Out[15]:

	Odometer (KM)	Doors

	Odometer (KM)	Doors
count	10.000000	10.000000
mean	78601.400000	4.000000
std	61983.471735	0.471405
min	11179.000000	3.000000
25%	35836.250000	4.000000
50%	57369.000000	4.000000
75%	96384.500000	4.000000
max	213095.000000	5.000000

In [17]:

```
# To display the details of the DataFrame (Ex: column names, data types, number of elem
car_sales.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10 entries, 0 to 9
Data columns (total 5 columns):
 #   Column         Non-Null Count  Dtype 
---  ------         --------------  ----- 
 0   Make           10 non-null     object
 1   Colour         10 non-null     object
 2   Odometer (KM)  10 non-null     int64 
 3   Doors          10 non-null     int64 
 4   Price          10 non-null     object
dtypes: int64(2), object(3)
memory usage: 528.0+ bytes
```

In [21]:

```
# To display sum of each column (string columns will be concatenate)
car_sales.sum()
```

Out[21]:

```
Make             ToyotaHondaToyotaBMWNissanToyotaHondaHondaToyo...
Colour               WhiteRedBlueBlackWhiteGreenBlueBlueWhiteWhite
Odometer (KM)                                               786014
Doors                                                           40
Price            $4,000.00$5,000.00$7,000.00$22,000.00$3,500.00...
dtype: object
```

In [25]:

```
# Display sum of Doors colums only
car_sales["Doors"].mean()
```

Out[25]:

```
4.0
```

In [24]:

```
# Each column in DataFrame are Series.
# To get mean of a Series (Same as above)
car_prices = pd.Series([1000, 2000, 3000])
car_prices.sum()
```

Out[24]:

```
6000
```

In [26]:

```
# To get the number of rows
len(car_sales)
```

Out[26]:

```
10
```

In [ ]: