

In [1]:

```
import pandas as pd
```

In [3]:

```
Cars = pd.read_csv('Cars Dataset.csv')  
Cars
```

Out[3]:

	Make	Model	Type	Origin	DriveTrain	MSRP	Invoice	EngineSize	Cylinders	Horsepower
0	Acura	MDX	SUV	Asia	All	\$36,945	\$33,337	3.5	6.0	26
1	Acura	RSX Type S 2dr	Sedan	Asia	Front	\$23,820	\$21,761	2.0	4.0	20
2	Acura	TSX 4dr	Sedan	Asia	Front	\$26,990	\$24,647	2.4	4.0	20
3	Acura	TL 4dr	Sedan	Asia	Front	\$33,195	\$30,299	3.2	6.0	27
4	Acura	3.5 RL 4dr	Sedan	Asia	Front	\$43,755	\$39,014	3.5	6.0	22
...
423	Volvo	C70 LPT convertible 2dr	Sedan	Europe	Front	\$40,565	\$38,203	2.4	5.0	19
424	Volvo	C70 HPT convertible 2dr	Sedan	Europe	Front	\$42,565	\$40,083	2.3	5.0	24
425	Volvo	S80 T6 4dr	Sedan	Europe	Front	\$45,210	\$42,573	2.9	6.0	26
426	Volvo	V40	Wagon	Europe	Front	\$26,135	\$24,641	1.9	4.0	17
427	Volvo	XC70	Wagon	Europe	All	\$35,145	\$33,112	2.5	5.0	20

428 rows × 15 columns

In [7]:

```
Cars.head(5)
```

Out[7]:

	Make	Model	Type	Origin	DriveTrain	MSRP	Invoice	EngineSize	Cylinders	Horsepower	MPG
0	Acura	MDX	SUV	Asia	All	\$36,945	\$33,337	3.5	6.0	265	
1	Acura	RSX Type S 2dr	Sedan	Asia	Front	\$23,820	\$21,761	2.0	4.0	200	
2	Acura	TSX 4dr	Sedan	Asia	Front	\$26,990	\$24,647	2.4	4.0	200	
3	Acura	TL 4dr	Sedan	Asia	Front	\$33,195	\$30,299	3.2	6.0	270	
4	Acura	3.5 RL 4dr	Sedan	Asia	Front	\$43,755	\$39,014	3.5	6.0	225	

In [9]:

```
Cars.shape
```

```
Out[9]:  
(428, 15)
```

1. Instruction (For Data Cleaning) - Find all Null Values in the dataset. If there is any null value in any column, then fill it with the mean of that column.

```
In [28]:
```

```
Cars.isnull().sum()
```

```
Out[28]:
```

```
Make          0  
Model         0  
Type          0  
Origin        0  
DriveTrain    0  
MSRP          0  
Invoice       0  
EngineSize    0  
Cylinders     0  
Horsepower    0  
MPG_City      0  
MPG_Highway   0  
Weight        0  
Wheelbase     0  
Length        0  
dtype: int64
```

```
In [26]:
```

```
mean = Cars['Cylinders'].mean()  
Cars.fillna({'Cylinders': mean}, inplace=True)  
Cars
```

```
Out[26]:
```

	Make	Model	Type	Origin	DriveTrain	MSRP	Invoice	EngineSize	Cylinders	Horsepower
0	Acura	MDX	SUV	Asia	All	\$36,945	\$33,337	3.5	6.0	26
1	Acura	RSX Type S 2dr	Sedan	Asia	Front	\$23,820	\$21,761	2.0	4.0	20
2	Acura	TSX 4dr	Sedan	Asia	Front	\$26,990	\$24,647	2.4	4.0	20
3	Acura	TL 4dr	Sedan	Asia	Front	\$33,195	\$30,299	3.2	6.0	27
4	Acura	3.5 RL 4dr	Sedan	Asia	Front	\$43,755	\$39,014	3.5	6.0	22
...
423	Volvo	C70 LPT convertible 2dr	Sedan	Europe	Front	\$40,565	\$38,203	2.4	5.0	19
424	Volvo	C70 HPT convertible 2dr	Sedan	Europe	Front	\$42,565	\$40,083	2.3	5.0	24

	Make	Model	Type	Origin	DriveTrain	MSRP	Invoice	EngineSize	Cylinders	Horsepower
425	Volvo	S80 T6 4dr	Sedan	Europe	Front	\$45,210	\$42,573	2.9	6.0	26
426	Volvo	V40	Wagon	Europe	Front	\$26,135	\$24,641	1.9	4.0	17
427	Volvo	XC70	Wagon	Europe	All	\$35,145	\$33,112	2.5	5.0	20

428 rows × 15 columns

2. Question (Based on Value Counts)- Check what are the different types of Make are there in our dataset. And, what is the count (occurrence) of each Make in the data ?

In [31]:

```
Cars.head(1)
```

Out[31]:

	Make	Model	Type	Origin	DriveTrain	MSRP	Invoice	EngineSize	Cylinders	Horsepower	MPG_
0	Acura	MDX	SUV	Asia	All	\$36,945	\$33,337	3.5	6.0	265	

In [33]:

```
Cars['Make'].value_counts()
```

Out[33]:

```
Make
Toyota      28
Chevrolet   27
Mercedes-Benz 26
Ford        23
BMW         20
Audi        19
Honda       17
Nissan       17
Volkswagen  15
Chrysler    15
Dodge       13
Mitsubishi  13
Volvo       12
Jaguar      12
Hyundai     12
Subaru      11
Pontiac     11
Mazda       11
Lexus       11
Kia         11
Buick       9
Mercury     9
Lincoln     9
Saturn      8
Cadillac    8
Suzuki      8
```

```

Infiniti      8
GMC           8
Acura        7
Porsche       7
Saab          7
Land Rover    3
Oldsmobile    3
Jeep          3
Scion         2
Isuzu         2
MINI          2
Hummer        1
Name: count, dtype: int64

```

3. Instruction (Filtering) - Show all the records where Origin is Asia or Europe.

In [36]:

```
Cars.head(1)
```

Out[36]:

	Make	Model	Type	Origin	DriveTrain	MSRP	Invoice	EngineSize	Cylinders	Horsepower	MPG_
0	Acura	MDX	SUV	Asia	All	\$36,945	\$33,337	3.5	6.0	265	

In [42]:

```
Cars[Cars['Origin'].isin(['Asia', 'Europe'])]
```

Out[42]:

	Make	Model	Type	Origin	DriveTrain	MSRP	Invoice	EngineSize	Cylinders	Horsepower	
0	Acura	MDX	SUV	Asia	All	\$36,945	\$33,337	3.5	6.0	26	
1	Acura	RSX Type S 2dr	Sedan	Asia	Front	\$23,820	\$21,761	2.0	4.0	20	
2	Acura	TSX 4dr	Sedan	Asia	Front	\$26,990	\$24,647	2.4	4.0	20	
3	Acura	TL 4dr	Sedan	Asia	Front	\$33,195	\$30,299	3.2	6.0	27	
4	Acura	3.5 RL 4dr	Sedan	Asia	Front	\$43,755	\$39,014	3.5	6.0	22	
...
423	Volvo	C70 LPT convertible 2dr	Sedan	Europe	Front	\$40,565	\$38,203	2.4	5.0	19	
424	Volvo	C70 HPT convertible 2dr	Sedan	Europe	Front	\$42,565	\$40,083	2.3	5.0	24	
425	Volvo	S80 T6 4dr	Sedan	Europe	Front	\$45,210	\$42,573	2.9	6.0	26	
426	Volvo	V40	Wagon	Europe	Front	\$26,135	\$24,641	1.9	4.0	17	
427	Volvo	XC70	Wagon	Europe	All	\$35,145	\$33,112	2.5	5.0	20	

281 rows × 15 columns

4. Instruction (Removing unwanted records) - Remove all the records (rows) where Weight is above 4000.

In [45]:

```
Cars.head(1)
```

Out[45]:

	Make	Model	Type	Origin	DriveTrain	MSRP	Invoice	EngineSize	Cylinders	Horsepower	MPG_
0	Acura	MDX	SUV	Asia	All	\$36,945	\$33,337	3.5	6.0	265	

In [47]:

```
Cars[~(Cars['Weight'] > 4000)]
```

Out[47]:

	Make	Model	Type	Origin	DriveTrain	MSRP	Invoice	EngineSize	Cylinders	Horsepower	MPG_
1	Acura	RSX Type S 2dr	Sedan	Asia	Front	\$23,820	\$21,761	2.0	4.0	200	24
2	Acura	TSX 4dr	Sedan	Asia	Front	\$26,990	\$24,647	2.4	4.0	200	24
3	Acura	TL 4dr	Sedan	Asia	Front	\$33,195	\$30,299	3.2	6.0	240	24
4	Acura	3.5 RL 4dr	Sedan	Asia	Front	\$43,755	\$39,014	3.5	6.0	240	24
5	Acura	3.5 RL w/Navigation 4dr	Sedan	Asia	Front	\$46,100	\$41,100	3.5	6.0	240	24
...
423	Volvo	C70 LPT convertible 2dr	Sedan	Europe	Front	\$40,565	\$38,203	2.4	5.0	200	24
424	Volvo	C70 HPT convertible 2dr	Sedan	Europe	Front	\$42,565	\$40,083	2.3	5.0	200	24
425	Volvo	S80 T6 4dr	Sedan	Europe	Front	\$45,210	\$42,573	2.9	6.0	240	24
426	Volvo	V40	Wagon	Europe	Front	\$26,135	\$24,641	1.9	4.0	200	24
427	Volvo	XC70	Wagon	Europe	All	\$35,145	\$33,112	2.5	5.0	200	24

325 rows × 15 columns

5. Instruction (Applying function on a column) - Increase all the values of 'MPG_City' column by 3.

In [50]:

```
Cars.head(1)
```

Out[50]:

	Make	Model	Type	Origin	DriveTrain	MSRP	Invoice	EngineSize	Cylinders	Horsepower	MPG_
0	Acura	MDX	SUV	Asia	All	\$36,945	\$33,337	3.5	6.0	265	

In [70]:

```
Cars['MPG_City'] = Cars['MPG_City'].apply(lambda x : x+3)
Cars
```

Out[70]:

	Make	Model	Type	Origin	DriveTrain	MSRP	Invoice	EngineSize	Cylinders	Horsepower	
0	Acura	MDX	SUV	Asia	All	\$36,945	\$33,337	3.5	6.0	26	
1	Acura	RSX Type S 2dr	Sedan	Asia	Front	\$23,820	\$21,761	2.0	4.0	20	
2	Acura	TSX 4dr	Sedan	Asia	Front	\$26,990	\$24,647	2.4	4.0	20	
3	Acura	TL 4dr	Sedan	Asia	Front	\$33,195	\$30,299	3.2	6.0	27	
4	Acura	3.5 RL 4dr	Sedan	Asia	Front	\$43,755	\$39,014	3.5	6.0	22	
...
423	Volvo	C70 LPT convertible 2dr	Sedan	Europe	Front	\$40,565	\$38,203	2.4	5.0	19	
424	Volvo	C70 HPT convertible 2dr	Sedan	Europe	Front	\$42,565	\$40,083	2.3	5.0	24	
425	Volvo	S80 T6 4dr	Sedan	Europe	Front	\$45,210	\$42,573	2.9	6.0	26	
426	Volvo	V40	Wagon	Europe	Front	\$26,135	\$24,641	1.9	4.0	17	
427	Volvo	XC70	Wagon	Europe	All	\$35,145	\$33,112	2.5	5.0	20	

428 rows × 15 columns