In [1]:
```
import pandas as pd
```

In [2]:
```
Data = pd.read_csv(r"Cars Dataset")
```

In [3]:
```
Data
```

Out[3]:

	Make	Model	Type	Origin	DriveTrain	MSRP	Invoice	EngineSize	Cylinders	Horsepower	MPG_City	MPG_Highway	Weight
0	Acura	MDX	SUV	Asia	All	$36,945	$33,337	3.5	6.0	265	17	23	4451
1	Acura	RSX Type S 2dr	Sedan	Asia	Front	$23,820	$21,761	2.0	4.0	200	24	31	2778
2	Acura	TSX 4dr	Sedan	Asia	Front	$26,990	$24,647	2.4	4.0	200	22	29	3230
3	Acura	TL 4dr	Sedan	Asia	Front	$33,195	$30,299	3.2	6.0	270	20	28	3575
4	Acura	3.5 RL 4dr	Sedan	Asia	Front	$43,755	$39,014	3.5	6.0	225	18	24	3880
...	...	...	...	...	...	...	...	...	...	...	...	...	...
423	Volvo	C70 LPT convertible 2dr	Sedan	Europe	Front	$40,565	$38,203	2.4	5.0	197	21	28	3450
424	Volvo	C70 HPT convertible 2dr	Sedan	Europe	Front	$42,565	$40,083	2.3	5.0	242	20	26	3450
425	Volvo	S80 T6 4dr	Sedan	Europe	Front	$45,210	$42,573	2.9	6.0	268	19	26	3653
426	Volvo	V40	Wagon	Europe	Front	$26,135	$24,641	1.9	4.0	170	22	29	2822
427	Volvo	XC70	Wagon	Europe	All	$35,145	$33,112	2.5	5.0	208	20	27	3823

428 rows × 15 columns

In [4]:
```
Data.shape
```

Out[4]:
```
(428, 15)
```

For Data Cleaning:

A) Find all Null value in the dataset if there is any null value in any column then fill it with the mean of that column?

In [6]:
```
Data.isnull().sum()
```

Out[6]:
```
Make          0
Model         0
Type          0
Origin        0
DriveTrain    0
MSRP          0
Invoice       0
EngineSize    0
Cylinders     2
Horsepower    0
MPG_City      0
MPG_Highway   0
Weight        0
Wheelbase     0
Length        0
dtype: int64
```

In [9]:
```
Data["Cylinders"].fillna(Data["Cylinders"].mean(),inplace= True)
```

In [10]:
```
Data.isnull().sum()
```

Out[10]:
```
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
```

2) Check what are the different types of Make are there in our dataset and what is the count (occurence) of each make in the data?

In [11]:
```
Data.head(2)
```

Out[11]:

	Make	Model	Type	Origin	DriveTrain	MSRP	Invoice	EngineSize	Cylinders	Horsepower	MPG_City	MPG_Highway	Weight	Wheelba
0	Acura	MDX	SUV	Asia	All	$36,945	$33,337	3.5	6.0	265	17	23	4451	
1	Acura	RSX Type S 2dr	Sedan	Asia	Front	$23,820	$21,761	2.0	4.0	200	24	31	2778	

In [15]:
```
Data["Make"].value_counts()
```

Out[15]:
```
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: Make, dtype: int64
```

Filtering:show all the records where Origin is Asia or Europe?

In [17]:
```
Data.head(3)
```

Out[17]:

	Make	Model	Type	Origin	DriveTrain	MSRP	Invoice	EngineSize	Cylinders	Horsepower	MPG_City	MPG_Highway	Weight	Wheelba
0	Acura	MDX	SUV	Asia	All	$36,945	$33,337	3.5	6.0	265	17	23	4451	
1	Acura	RSX Type S 2dr	Sedan	Asia	Front	$23,820	$21,761	2.0	4.0	200	24	31	2778	
2	Acura	TSX 4dr	Sedan	Asia	Front	$26,990	$24,647	2.4	4.0	200	22	29	3230	

In [20]:
```
Data[Data["Origin"].isin(['Asia','Europe'])]
```

Out[20]:

	Make	Model	Type	Origin	DriveTrain	MSRP	Invoice	EngineSize	Cylinders	Horsepower	MPG_City	MPG_Highway	Weight
0	Acura	MDX	SUV	Asia	All	$36,945	$33,337	3.5	6.0	265	17	23	4451
1	Acura	RSX Type S 2dr	Sedan	Asia	Front	$23,820	$21,761	2.0	4.0	200	24	31	2778
2	Acura	TSX 4dr	Sedan	Asia	Front	$26,990	$24,647	2.4	4.0	200	22	29	3230
3	Acura	TL 4dr	Sedan	Asia	Front	$33,195	$30,299	3.2	6.0	270	20	28	3575
4	Acura	3.5 RL 4dr	Sedan	Asia	Front	$43,755	$39,014	3.5	6.0	225	18	24	3880
...	...	...	...	...	...	...	...	...	...	...	...	...	...
423	Volvo	C70 LPT convertible 2dr	Sedan	Europe	Front	$40,565	$38,203	2.4	5.0	197	21	28	3450
424	Volvo	C70 HPT convertible 2dr	Sedan	Europe	Front	$42,565	$40,083	2.3	5.0	242	20	26	3450
425	Volvo	S80 T6 4dr	Sedan	Europe	Front	$45,210	$42,573	2.9	6.0	268	19	26	3653
426	Volvo	V40	Wagon	Europe	Front	$26,135	$24,641	1.9	4.0	170	22	29	2822
427	Volvo	XC70	Wagon	Europe	All	$35,145	$33,112	2.5	5.0	208	20	27	3823

281 rows × 15 columns

4) Removing unwanted records:

A) Remove all the records(rows) where weight is above 4000?

In [22]:
```
Data.head(2)
```

Out[22]:

	Make	Model	Type	Origin	DriveTrain	MSRP	Invoice	EngineSize	Cylinders	Horsepower	MPG_City	MPG_Highway	Weight	Wheelba
0	Acura	MDX	SUV	Asia	All	$36,945	$33,337	3.5	6.0	265	17	23	4451	
1	Acura	RSX Type S 2dr	Sedan	Asia	Front	$23,820	$21,761	2.0	4.0	200	24	31	2778	

In [25]:
```
Data[~(Data['Weight']>4000)]
```

Out[25]:

	Make	Model	Type	Origin	DriveTrain	MSRP	Invoice	EngineSize	Cylinders	Horsepower	MPG_City	MPG_Highway	Weight
1	Acura	RSX Type S 2dr	Sedan	Asia	Front	$23,820	$21,761	2.0	4.0	200	24	31	2778
2	Acura	TSX 4dr	Sedan	Asia	Front	$26,990	$24,647	2.4	4.0	200	22	29	3230
3	Acura	TL 4dr	Sedan	Asia	Front	$33,195	$30,299	3.2	6.0	270	20	28	3575
4	Acura	3.5 RL 4dr	Sedan	Asia	Front	$43,755	$39,014	3.5	6.0	225	18	24	3880
5	Acura	3.5 RL w/Navigation 4dr	Sedan	Asia	Front	$46,100	$41,100	3.5	6.0	225	18	24	3893
...	...	...	...	...	...	...	...	...	...	...	...	...	...
423	Volvo	C70 LPT convertible 2dr	Sedan	Europe	Front	$40,565	$38,203	2.4	5.0	197	21	28	3450
424	Volvo	C70 HPT convertible 2dr	Sedan	Europe	Front	$42,565	$40,083	2.3	5.0	242	20	26	3450
425	Volvo	S80 T6 4dr	Sedan	Europe	Front	$45,210	$42,573	2.9	6.0	268	19	26	3653
426	Volvo	V40	Wagon	Europe	Front	$26,135	$24,641	1.9	4.0	170	22	29	2822
427	Volvo	XC70	Wagon	Europe	All	$35,145	$33,112	2.5	5.0	208	20	27	3823

325 rows × 15 columns

6) Applying function on a column:

A) Increase all the values of 'MPG_City' column by 3

In [26]:
```
Data.head(3)
```

Out[26]:

	Make	Model	Type	Origin	DriveTrain	MSRP	Invoice	EngineSize	Cylinders	Horsepower	MPG_City	MPG_Highway	Weight	Wheelba
0	Acura	MDX	SUV	Asia	All	$36,945	$33,337	3.5	6.0	265	17	23	4451	
1	Acura	RSX Type S 2dr	Sedan	Asia	Front	$23,820	$21,761	2.0	4.0	200	24	31	2778	
2	Acura	TSX 4dr	Sedan	Asia	Front	$26,990	$24,647	2.4	4.0	200	22	29	3230	

In [27]:
```
Data['MPG_City'] = Data['MPG_City'].apply(lambda x:x+3)
```

In [28]:
```
Data
```

Out[28]:

	Make	Model	Type	Origin	DriveTrain	MSRP	Invoice	EngineSize	Cylinders	Horsepower	MPG_City	MPG_Highway	Weight
0	Acura	MDX	SUV	Asia	All	$36,945	$33,337	3.5	6.0	265	20	23	4451
1	Acura	RSX Type S 2dr	Sedan	Asia	Front	$23,820	$21,761	2.0	4.0	200	27	31	2778
2	Acura	TSX 4dr	Sedan	Asia	Front	$26,990	$24,647	2.4	4.0	200	25	29	3230
3	Acura	TL 4dr	Sedan	Asia	Front	$33,195	$30,299	3.2	6.0	270	23	28	3575
4	Acura	3.5 RL 4dr	Sedan	Asia	Front	$43,755	$39,014	3.5	6.0	225	21	24	3880
...	...	...	...	...	...	...	...	...	...	...	...	...	...
423	Volvo	C70 LPT convertible 2dr	Sedan	Europe	Front	$40,565	$38,203	2.4	5.0	197	24	28	3450
424	Volvo	C70 HPT convertible 2dr	Sedan	Europe	Front	$42,565	$40,083	2.3	5.0	242	23	26	3450
425	Volvo	S80 T6 4dr	Sedan	Europe	Front	$45,210	$42,573	2.9	6.0	268	22	26	3653
426	Volvo	V40	Wagon	Europe	Front	$26,135	$24,641	1.9	4.0	170	25	29	2822
427	Volvo	XC70	Wagon	Europe	All	$35,145	$33,112	2.5	5.0	208	23	27	3823

428 rows × 15 columns

In [ ]:
```

```