

Pandas Lab -3 Exercise (Kaggle Automobile Dataset)

We will be using the [automobiles Dataset \(https://www.kaggle.com/nisargpatel/automobiles/data\)](https://www.kaggle.com/nisargpatel/automobiles/data) from Kaggle.

Answer each question asked below wrt the automobiles dataset. Load pandas as pd and upload the Automobile.csv file as auto

In [6]:

In [7]:

Check the head of the DataFrame.

In [4]:

Out[4]:

	symboling	normalized_losses	make	fuel_type	aspiration	number_of_doors	body_style	drive_wheels	engine_location	wheel_base	...	engine_size	fuel...
0	3	168	alfa-romero	gas	std	two	convertible	rwd	front	88.6	...	130	
1	3	168	alfa-romero	gas	std	two	convertible	rwd	front	88.6	...	130	
2	1	168	alfa-romero	gas	std	two	hatchback	rwd	front	94.5	...	152	
3	2	164	audi	gas	std	four	sedan	fwd	front	99.8	...	109	
4	2	164	audi	gas	std	four	sedan	4wd	front	99.4	...	136	

5 rows × 26 columns

\*\* How many rows and columns are there? \*\*

In [13]:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 201 entries, 0 to 200
Data columns (total 26 columns):
#   Column                Non-Null Count  Dtype
---  -
0   symboling              201 non-null   int64
1   normalized_losses      201 non-null   int64
2   make                   201 non-null   object
3   fuel_type              201 non-null   object
4   aspiration              201 non-null   object
5   number_of_doors        201 non-null   object
6   body_style             201 non-null   object
7   drive_wheels           201 non-null   object
8   engine_location        201 non-null   object
9   wheel_base             201 non-null   float64
10  length                 201 non-null   float64
11  width                  201 non-null   float64
12  height                 201 non-null   float64
13  curb_weight            201 non-null   int64
14  engine_type            201 non-null   object
15  number_of_cylinders     201 non-null   object
16  engine_size            201 non-null   int64
17  fuel_system            201 non-null   object
18  bore                   201 non-null   float64
19  stroke                 201 non-null   float64
20  compression_ratio      201 non-null   float64
21  horsepower             201 non-null   int64
22  peak_rpm               201 non-null   int64
23  city_mpg               201 non-null   int64
24  highway_mpg            201 non-null   int64
25  price                  201 non-null   int64
dtypes: float64(7), int64(9), object(10)
memory usage: 41.0+ KB
```

\*\* What is the average Price of all cars in the dataset? \*\*

In [14]:

Out[14]: 13207.129353233831

\*\* Which is the cheapest make and costliest make of car in the lot? \*\*

```
In [15]:
```

Out[15]:

	symboling	normalized_losses	make	fuel_type	aspiration	number_of_doors	body_style	drive_wheels	engine_location	wheel_base	...	engine_size	f
71	1	140	mercedes-benz	gas	std	two	hardtop	rwd	front	112.0	...	304	

1 rows × 26 columns

```
In [16]:
```

Out[16]:

	symboling	normalized_losses	make	fuel_type	aspiration	number_of_doors	body_style	drive_wheels	engine_location	wheel_base	...	engine_size	fue
134	2	83	subaru	gas	std	two	hatchback	fwd	front	93.7	...	97	

1 rows × 26 columns

\*\* How many cars have horsepower greater than 100? \*\*

```
In [17]:
```

Out[17]:

symboling	90
normalized_losses	90
make	90
fuel_type	90
aspiration	90
number_of_doors	90
body_style	90
drive_wheels	90
engine_location	90
wheel_base	90
length	90
width	90
height	90
curb_weight	90
engine_type	90
number_of_cylinders	90
engine_size	90
fuel_system	90
bore	90
stroke	90
compression_ratio	90
horsepower	90
peak_rpm	90
city_mpg	90
highway_mpg	90
price	90
dtype: int64	

\*\* How many hatchback cars are in the dataset ? \*\*

In [18]:

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 68 entries, 2 to 186
Data columns (total 26 columns):
#   Column                Non-Null Count  Dtype
---  -
0   symboling              68 non-null    int64
1   normalized_losses      68 non-null    int64
2   make                  68 non-null    object
3   fuel_type              68 non-null    object
4   aspiration              68 non-null    object
5   number_of_doors        68 non-null    object
6   body_style             68 non-null    object
7   drive_wheels           68 non-null    object
8   engine_location        68 non-null    object
9   wheel_base             68 non-null    float64
10  length                 68 non-null    float64
11  width                  68 non-null    float64
12  height                 68 non-null    float64
13  curb_weight            68 non-null    int64
14  engine_type            68 non-null    object
15  number_of_cylinders    68 non-null    object
16  engine_size            68 non-null    int64
17  fuel_system            68 non-null    object
18  bore                   68 non-null    float64
19  stroke                 68 non-null    float64
20  compression_ratio      68 non-null    float64
21  horsepower             68 non-null    int64
22  peak_rpm               68 non-null    int64
23  city_mpg               68 non-null    int64
24  highway_mpg            68 non-null    int64
25  price                  68 non-null    int64
dtypes: float64(7), int64(9), object(10)
memory usage: 14.3+ KB
```

\*\* What are the 3 most commonly found cars in the dataset? \*\*

In [19]:

```
Out[19]: toyota    32
         nissan    18
         mazda     17
         Name: make, dtype: int64
```

\*\* Someone purchased a car for 7099, what is the make of the car? \*\*

In [20]:

```
Out[20]: 87    nissan
         Name: make, dtype: object
```

\*\*\* Which cars are priced greater than 40000? \*\*

In [31]:

```
Out[31]:
```

	symboling	normalized_losses	make	fuel_type	aspiration	number_of_doors	body_style	drive_wheels	engine_location	wheel_base	...	engine_size	f
15	0	149	bmw	gas	std	two	sedan	rwd	front	103.5	...	209	
70	0	140	mercedes-benz	gas	std	four	sedan	rwd	front	120.9	...	308	
71	1	140	mercedes-benz	gas	std	two	hardtop	rwd	front	112.0	...	304	

3 rows × 26 columns

\*\* Which are the cars that are both a sedan and priced less than 7000? \*\*

In [21]:

Out[21]:

	symboling	normalized_losses	make	fuel_type	aspiration	number_of_doors	body_style	drive_wheels	engine_location	wheel_base	...	engine_size
19	0	81	chevrolet	gas	std	four	sedan	fwd	front	94.5	...	90
24	1	148	dodge	gas	std	four	sedan	fwd	front	93.7	...	90
42	0	110	isuzu	gas	std	four	sedan	rwd	front	94.3	...	111
50	1	113	mazda	gas	std	four	sedan	fwd	front	93.1	...	91
82	1	125	mitsubishi	gas	std	four	sedan	fwd	front	96.3	...	122
86	1	128	nissan	gas	std	two	sedan	fwd	front	94.5	...	97
88	1	128	nissan	gas	std	two	sedan	fwd	front	94.5	...	97
89	1	122	nissan	gas	std	four	sedan	fwd	front	94.5	...	97
118	1	154	plymouth	gas	std	four	sedan	fwd	front	93.7	...	90
152	0	91	toyota	gas	std	four	sedan	fwd	front	95.7	...	98

10 rows × 26 columns



The END