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

In [2]:
```
import numpy as np
```

In [3]:
```
cd
```

```
C:\Users\CVR
```

In [7]:
```
am=pd.read_csv(r"C:\Users\CVR\Downloads\Automobile.csv")
```

In [8]:
```
am
```

Out[8]:

	symboling	normalized_losses	make	fuel_type	aspiration	number_of_doors	body_style	d
0	3	168	alfa-romero	gas	std	two	convertible	
1	3	168	alfa-romero	gas	std	two	convertible	
2	1	168	alfa-romero	gas	std	two	hatchback	
3	2	164	audi	gas	std	four	sedan	
4	2	164	audi	gas	std	four	sedan	
...	...	...	...	...	...	...	...	
196	-1	95	volvo	gas	std	four	sedan	
197	-1	95	volvo	gas	turbo	four	sedan	
198	-1	95	volvo	gas	std	four	sedan	
199	-1	95	volvo	diesel	turbo	four	sedan	
200	-1	95	volvo	gas	turbo	four	sedan	

201 rows × 26 columns

In [9]:
```
type(am)
```

Out[9]:
```
pandas.core.frame.DataFrame
```

In [10]:
```
am.head()
```

Out[10]:

	symboling	normalized_losses	make	fuel_type	aspiration	number_of_doors	body_style	driv
0	3	168	alfa-romero	gas	std	two	convertible	
1	3	168	alfa-romero	gas	std	two	convertible	
2	1	168	alfa-romero	gas	std	two	hatchback	
3	2	164	audi	gas	std	four	sedan	
4	2	164	audi	gas	std	four	sedan	

5 rows × 26 columns

In [12]:

```
am.tail()
```

Out[12]:

	symboling	normalized_losses	make	fuel_type	aspiration	number_of_doors	body_style	dri
196	-1	95	volvo	gas	std	four	sedan	
197	-1	95	volvo	gas	turbo	four	sedan	
198	-1	95	volvo	gas	std	four	sedan	
199	-1	95	volvo	diesel	turbo	four	sedan	
200	-1	95	volvo	gas	turbo	four	sedan	

5 rows × 26 columns

In [13]:

```
am.info()
```

```
<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
```

In [14]:

```
am.describe
```

Out[14]:

```
<bound method NDFrame.describe of      symboling  normalized_losses         make f
uel_type aspiration  \
0            3                168  alfa-romero       gas        std
1            3                168  alfa-romero       gas        std
2            1                168  alfa-romero       gas        std
3            2                164         audi       gas        std
4            2                164         audi       gas        std
..         ...                ...          ...       ...        ...
196         -1                 95        volvo       gas        std
197         -1                 95        volvo       gas      turbo
198         -1                 95        volvo       gas        std
199         -1                 95        volvo    diesel      turbo
200         -1                 95        volvo       gas      turbo

    number_of_doors   body_style drive_wheels engine_location  wheel_base  \
0               two  convertible          rwd           front        88.6
1               two  convertible          rwd           front        88.6
2               two    hatchback          rwd           front        94.5
3              four        sedan          fwd           front        99.8
4              four        sedan          4wd           front        99.4
..              ...          ...          ...             ...         ...
196            four        sedan          rwd           front       109.1
197            four        sedan          rwd           front       109.1
198            four        sedan          rwd           front       109.1
199            four        sedan          rwd           front       109.1
200            four        sedan          rwd           front       109.1

     ...  engine_size  fuel_system  bore  stroke compression_ratio horsepower  \
0    ...          130         mpfi  3.47    2.68               9.0        111
1    ...          130         mpfi  3.47    2.68               9.0        111
2    ...          152         mpfi  2.68    3.47               9.0        154
3    ...          109         mpfi  3.19    3.40              10.0        102
4    ...          136         mpfi  3.19    3.40               8.0        115
..   ...          ...          ...   ...     ...               ...        ...
196  ...          141         mpfi  3.78    3.15               9.5        114
197  ...          141         mpfi  3.78    3.15               8.7        160
198  ...          173         mpfi  3.58    2.87               8.8        134
199  ...          145          idi  3.01    3.40              23.0        106
200  ...          141         mpfi  3.78    3.15               9.5        114

     peak_rpm city_mpg  highway_mpg  price
0        5000       21           27  13495
1        5000       21           27  16500
2        5000       19           26  16500
3        5500       24           30  13950
4        5500       18           22  17450
..        ...      ...          ...    ...
196      5400       23           28  16845
197      5300       19           25  19045
198      5500       18           23  21485
199      4800       26           27  22470
200      5400       19           25  22625

[201 rows x 26 columns]>
```

In [15]:

```
am.shape
```

Out[15]:

```
(201, 26)
```

In [16]:

```
am.columns
```

```
Out[16]: Index(['symboling', 'normalized_losses', 'make', 'fuel_type', 'aspiration',
               'number_of_doors', 'body_style', 'drive_wheels', 'engine_location',
               'wheel_base', 'length', 'width', 'height', 'curb_weight', 'engine_type',
               'number_of_cylinders', 'engine_size', 'fuel_system', 'bore', 'stroke',
               'compression_ratio', 'horsepower', 'peak_rpm', 'city_mpg',
               'highway_mpg', 'price'],
              dtype='object')
```

```
In [17]: am.loc[2:3]
```

Out[17]:

	symboling	normalized_losses	make	fuel_type	aspiration	number_of_doors	body_style	driv
2	1	168	alfa-romero	gas	std	two	hatchback	
3	2	164	audi	gas	std	four	sedan	

2 rows × 26 columns

```
In [18]: am.loc[2]
```

```
Out[18]: symboling                        1
         normalized_losses              168
         make                   alfa-romero
         fuel_type                      gas
         aspiration                     std
         number_of_doors                two
         body_style               hatchback
         drive_wheels                   rwd
         engine_location              front
         wheel_base                    94.5
         length                       171.2
         width                         65.5
         height                        52.4
         curb_weight                   2823
         engine_type                   ohcv
         number_of_cylinders            six
         engine_size                    152
         fuel_system                   mpfi
         bore                          2.68
         stroke                        3.47
         compression_ratio              9.0
         horsepower                     154
         peak_rpm                      5000
         city_mpg                        19
         highway_mpg                     26
         price                        16500
         Name: 2, dtype: object
```

```
In [26]: am.loc[:,"fuel_type"]
```

Out[26]:

```
0         gas
1         gas
2         gas
3         gas
4         gas
        ...  
196       gas
197       gas
198       gas
199    diesel
200       gas
Name: fuel_type, Length: 201, dtype: object
```

In [19]:

```
am.iloc[:3:2]
```

Out[19]:

	symboling	normalized_losses	make	fuel_type	aspiration	number_of_doors	body_style	driv
0	3	168	alfa-romero	gas	std	two	convertible	
2	1	168	alfa-romero	gas	std	two	hatchback	

2 rows × 26 columns

In [24]:

```
am.iloc[:,:3]
```

Out[24]:

	symboling	normalized_losses	make
0	3	168	alfa-romero
1	3	168	alfa-romero
2	1	168	alfa-romero
3	2	164	audi
4	2	164	audi
...	...	...	...
196	-1	95	volvo
197	-1	95	volvo
198	-1	95	volvo
199	-1	95	volvo
200	-1	95	volvo

201 rows × 3 columns

In [20]:

```
am.isnull()
```

```
Out[20]:
        symboling  normalized_losses   make  fuel_type  aspiration  number_of_doors  body_style  dri
    0       False              False  False      False       False            False       False
    1       False              False  False      False       False            False       False
    2       False              False  False      False       False            False       False
    3       False              False  False      False       False            False       False
    4       False              False  False      False       False            False       False
  ...         ...                ...    ...        ...         ...              ...         ...
  196       False              False  False      False       False            False       False
  197       False              False  False      False       False            False       False
  198       False              False  False      False       False            False       False
  199       False              False  False      False       False            False       False
  200       False              False  False      False       False            False       False

201 rows × 26 columns
```

```
In [51]: am.sample(10)
```

```
Out[51]:
        symboling  normalized_losses  manufacturer  fuel_type  aspiration  number_of_doors  body_st
   57           0                115         mazda        gas         std             four      sec
  182           2                 94    volkswagen        gas         std             four      sec
   10           0                192           bmw        gas         std             four      sec
   41           1                107         honda        gas         std              two      sec
  164           2                134        toyota        gas         std              two     hard
   51           1                113         mazda        gas         std             four      sec
  137           0                102        subaru        gas         std             four      sec
   75           2                161    mitsubishi        gas         std              two   hatchb
  155           0                 91        toyota     diesel         std             four   hatchb
   23           1                148         dodge        gas         std             four   hatchb

10 rows × 26 columns
```

```
In [21]: am.isnull().sum()
```

Out[21]:

```
symboling              0
normalized_losses      0
make                   0
fuel_type              0
aspiration             0
number_of_doors        0
body_style             0
drive_wheels           0
engine_location        0
wheel_base             0
length                 0
width                  0
height                 0
curb_weight            0
engine_type            0
number_of_cylinders    0
engine_size            0
fuel_system            0
bore                   0
stroke                 0
compression_ratio      0
horsepower             0
peak_rpm               0
city_mpg               0
highway_mpg            0
price                  0
dtype: int64
```

In [29]:

```
am["length"].unique()
```

Out[29]:

```
array([168.8, 171.2, 176.6, 177.3, 192.7, 176.8, 189. , 193.8, 197. ,
       141.1, 155.9, 158.8, 157.3, 174.6, 173.2, 144.6, 150. , 163.4,
       157.1, 167.5, 175.4, 169.1, 170.7, 172.6, 199.6, 191.7, 159.1,
       166.8, 169. , 177.8, 175. , 190.9, 187.5, 202.6, 180.3, 208.1,
       199.2, 178.4, 173. , 172.4, 165.3, 170.2, 165.6, 162.4, 173.4,
       181.7, 184.6, 178.5, 186.7, 198.9, 167.3, 168.9, 181.5, 186.6,
       156.9, 157.9, 172. , 173.5, 173.6, 158.7, 169.7, 166.3, 168.7,
       176.2, 175.6, 183.5, 187.8, 171.7, 159.3, 165.7, 180.2, 183.1,
       188.8])
```

In [30]:

```
am["fuel_type"].unique()
```

Out[30]:

```
array(['gas', 'diesel'], dtype=object)
```

In [35]:

```
dups=am[am.duplicated()]
```

In [36]:

```
dups
```

Out[36]:

	symboling	normalized_losses	make	fuel_type	aspiration	number_of_doors	body_style	drive_v

0 rows × 26 columns

In [37]:

```
am=am.drop_duplicates()
```

In [40]:

```
am.rename(columns={"make":"manufacturer"})
```

Out[40]:

	symboling	normalized_losses	manufacturer	fuel_type	aspiration	number_of_doors	body_st
0	3	168	alfa-romero	gas	std	two	converti
1	3	168	alfa-romero	gas	std	two	converti
2	1	168	alfa-romero	gas	std	two	hatchb
3	2	164	audi	gas	std	four	sec
4	2	164	audi	gas	std	four	sec
...	...	...	...	...	...	...	
196	-1	95	volvo	gas	std	four	sec
197	-1	95	volvo	gas	turbo	four	sec
198	-1	95	volvo	gas	std	four	sec
199	-1	95	volvo	diesel	turbo	four	sec
200	-1	95	volvo	gas	turbo	four	sec

201 rows × 26 columns

In [42]:

```
am.rename(columns={"make":"manufacturer"},inplace=True)
```

In [43]:

```
am
```

Out[43]:

	symboling	normalized_losses	manufacturer	fuel_type	aspiration	number_of_doors	body_st
0	3	168	alfa-romero	gas	std	two	converti
1	3	168	alfa-romero	gas	std	two	converti
2	1	168	alfa-romero	gas	std	two	hatchb
3	2	164	audi	gas	std	four	sec
4	2	164	audi	gas	std	four	sec
...	...	...	...	...	...	...	
196	-1	95	volvo	gas	std	four	sec
197	-1	95	volvo	gas	turbo	four	sec
198	-1	95	volvo	gas	std	four	sec
199	-1	95	volvo	diesel	turbo	four	sec
200	-1	95	volvo	gas	turbo	four	sec

201 rows × 26 columns

In [44]:

```
am["price"].mean()
```

Out[44]:

```
13207.129353233831
```

In [45]:

```
am["price"].median()
```

Out[45]:

```
10295.0
```

In [46]:

```
am["price"].mode()
```

Out[46]:

```
0      5572
1      6229
2      6692
3      7295
4      7609
5      7775
6      7898
7      7957
8      8495
9      8845
10     8921
11     9279
12    13499
13    16500
14    18150
Name: price, dtype: int64
```

In [50]:

```
c=(am["price"]==5572).sum()
c
```

Out[50]:

```
2
```

In [47]:

```
am["price"].std()
```

Out[47]:

```
7947.066341939271
```

In [48]:

```
am["price"].var()
```

Out[48]:

```
63155863.443184026
```

In [52]:

```
am["price"].isnull()
```

Out[52]:

```
0      False
1      False
2      False
3      False
4      False
       ...  
196    False
197    False
198    False
199    False
200    False
Name: price, Length: 201, dtype: bool
```

In [55]:

```
(am["price"].isnull()).sum()
```

Out[55]:

```
0
```

In [57]:

```
am.fillna(13207.129353233831)

#to replace the null values with a specific value
#here we are replacing nan with mean value but if u provide inplace=True the origi
```

Out[57]:

	symboling	normalized_losses	manufacturer	fuel_type	aspiration	number_of_doors	body_st
0	3	168	alfa-romero	gas	std	two	converti
1	3	168	alfa-romero	gas	std	two	converti
2	1	168	alfa-romero	gas	std	two	hatchb
3	2	164	audi	gas	std	four	sec
4	2	164	audi	gas	std	four	sec
...	...	...	...	...	...	...	
196	-1	95	volvo	gas	std	four	sec
197	-1	95	volvo	gas	turbo	four	sec
198	-1	95	volvo	gas	std	four	sec
199	-1	95	volvo	diesel	turbo	four	sec
200	-1	95	volvo	gas	turbo	four	sec

201 rows × 26 columns

```
In [ ]:
```

```
In [ ]:
```

```
In [ ]:
```

```
In [ ]:
```