MINI PROJECT

Exploratory Data Analysis (EDA)

A project report on exploring the dataset

on

Car sales

Submitted by

NIVETHA SREE M

Computer Science and Engineeering

Adhiyamaan College Of Engineering, Hosur.

Date: 12/12/2022

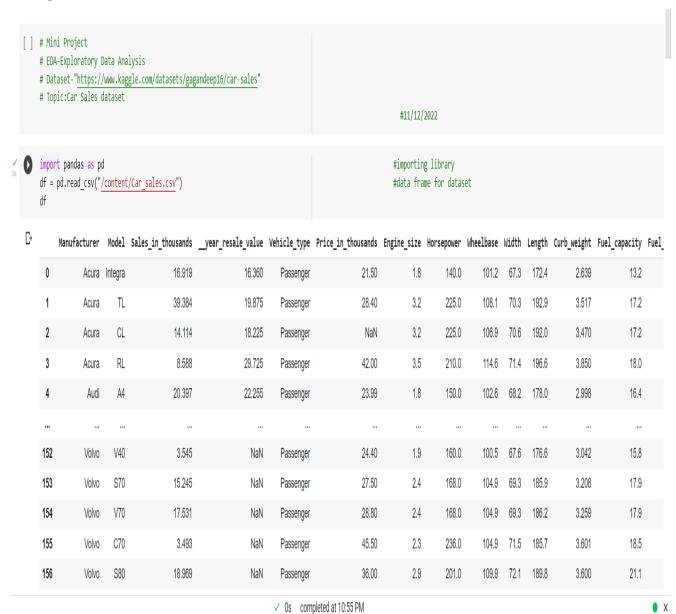
Objective:

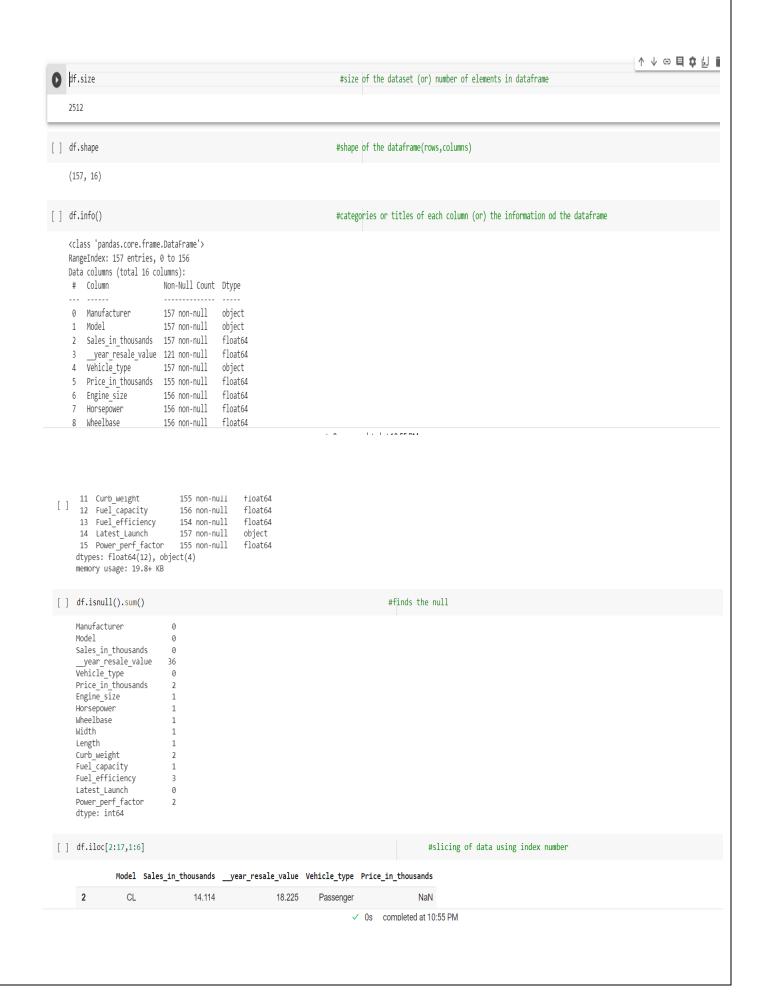
To Explore the dataset (EDA).

Dataset: Car Sales.

Link for dataset: https://www.kaggle.com/datasets/gagandeep16/car-sales

REPORT:





[] df.iloc[2:17,1:6]

#slicing of data using index number

	Model	Sales_in_thousands	year_resale_value	Vehicle_type	Price_in_thousands
2	CL	14.114	18.225	Passenger	NaN
3	RL	8.588	29.725	Passenger	42.000
4	A4	20.397	22.255	Passenger	23.990
5	A6	18.780	23.555	Passenger	33.950
6	A8	1.380	39.000	Passenger	62.000
7	323i	19.747	NaN	Passenger	26.990
8	328i	9.231	28.675	Passenger	33.400
9	528i	17.527	36.125	Passenger	38.900
10	Century	91.561	12.475	Passenger	21.975
11	Regal	39.350	13.740	Passenger	25.300
12	Park Avenue	27.851	20.190	Passenger	31.965
13	LeSabre	83.257	13.360	Passenger	27.885
14	DeVille	63.729	22.525	Passenger	39.895
15	Seville	15.943	27.100	Passenger	44.475
16	Eldorado	6.536	25.725	Passenger	39.665

[] df.nunique

 $\mbox{\tt\#unique}$ names present in the dataframe

✓ 0s completed at 10:55 PM

▶ df.nunique

#unique names present in the dataframe

_								
₽	≺bound	d method Dat	aFrame.nunique of	Manufactur	er Model Sa	ales_in_thousands	year_resale_value	\
	0	Acura	Integra	16.919	16.	360		
	1	Acura	TL	39.384	19.8	875		
	2	Acura	CL	14.114	18.3	225		
	3	Acura	RL	8.588	29.	725		
	4	Audi	A4	20.397	22.2	255		
	152	Volvo	V40	3.545	1	NaN		
	153	Volvo	S70	15.245	1	NaN		
	154	Volvo	V70	17.531	1	NaN		
	155	Volvo	C70	3.493	1	NaN		
	156	Volvo	S80	18.969	1	NaN		
	Ve	hicle type	Price in thousan	ds Engine size	Horsepower I	wheelbase \		
	0	Passenger	21.	50 1.8	140.0	101.2		
	1	Passenger	28.	40 3.2	225.0	108.1		
	2	Passenger	N	aN 3.2	225.0	106.9		
	3	Passenger	42.	00 3.5	210.0	114.6		
	4	Passenger	23.	99 1.8	150.0	102.6		
	152	Passenger	24.	40 1.9	160.0	100.5		
	153	Passenger	27.	50 2.4	168.0	104.9		
	154	Passenger	28.	80 2.4	168.0	104.9		
	155	Passenger	45.	50 2.3	236.0	104.9		
	156	Passenger	36.	00 2.9	201.0	109.9		
	V	Vidth Lengt	th Curb weight F	uel capacity F	uel efficiency	Latest Launch \		
	0	67.3 172.	4 2.639	13.2	28.0	2/2/2012		
	1	70.3 192.	9 3.517	17.2	25.0	6/3/2011		
	2	70.6 192.	0 3.470	17.2	26.0	1/4/2012		
	3	71.4 196.	6 3.850	18.0	22.0	3/10/2011		
	4	68.2 178.	0 2.998	16.4	27.0	10/8/2011		

✓ 0s completed at 10:55 PM

```
[ ] df.groupby("Manufacturer",sort = False).size()
                                                                                                   #sorting in Alphabetical order
      Manufacturer
      Acura
Audi
      BMW
      Cadillac
      Chevrolet
                    9
7
     Chrysler
Dodge
                   11
                   11
      Honda
      Hyundai
      Infiniti
      Jaguar
      Jeep
      Lexus
      Lincoln
      Mitsubishi
      Mercury
Mercedes-B
                    6
9
     Nissan
Oldsmobile
      Plymouth
      Pontiac
      Porsche
Saab
      Saturn
      Subaru
      Toyota
      Volkswagen
                    6
      Volvo
                    6
] df.Model.value_counts()
                                                                          #count of specified(Model) from dataframe
   Neon
                2
   Integra
   Cutlass
               1
   Sentra
               1
   Altima
               1
   Windstar
   Expedition 1
   Ranger
                1
   F-Series
   S80
   Name: Model, Length: 156, dtype: int64
] df.Fuel_capacity.value_counts()
   18.5 14
   17.0 9
   20.0
          8
   19.0
           8
           7
   16.0
   13.2
           6
   15.9
   15.0
           5
   14.5
           5
   17.5
           5
   21.1
           4
   17.2
           4
   16.4
           4
   18.0
           4
   12.1
           3
   16.6
                                                                     ✓ 0s completed at 10:55 PM
```

```
#importing library
[] import numpy as np
    Lowest_fuel_capacity = np.sum((df['Fuel_capacity'] >= 10)&(df['Fuel_capacity'] < 15))
                                                                                                      #divide a column into categories
    Low_fuel_capacity = np.sum((df['Fuel_capacity'] >= 15)&(df['Fuel_capacity'] < 20))
                                                                                                      #on the basis of fuel capacity
    Average_fuel_capacity = np.sum((df['Fuel_capacity'] >= 20)&(df['Fuel_capacity'] \leq 25))
    High_fuel_capacity = np.sum((df['Fuel_capacity'] >= 25)&(df['Fuel_capacity'] < 30))</pre>
    print(Lowest_fuel_capacity,Low_fuel_capacity,Average_fuel_capacity,High_fuel_capacity)
    27 93 23 9
[ ] np.min(df.Fuel_capacity)
                                                                #the least fuel holding capacity
    10.3
[] np.max(df.Fuel_capacity)
                                                                #the highest fuel holding capacity
    32.0
[ ] #CONCLUSION OF THE EDA
     #data frame for dataset
     #size of the dataset (or) number of elements in dataframe
     #shape of the dataframe(rows,columns)
     #categories or titles of each column (or) the information od the dataframe
     #finds the null
     #slicing of data using index number
     #unique names present in the dataframe
     #sorting in Alphabetical order
     #count of specified(Model) from dataframe
     #divide a column into categories on the basis of fuel capacity
     #the least fuel holding capacity
     #the highest fuel holding capacity
```

/ On completed at 10:EE DM

LINK TO DRIVE: https://tinyurl.com/rinexprojnivetha