

PREPROCESSING TRAINING DATA :CAR ECONOMY PRICE FILE

```
import numpy as np
import pandas as pd
df=pd.read_csv('/content/train-data (1).csv')
df
```

	Unnamed: 0	Name	Location	Year	Kilometers_Driven	Fuel_Type	Transmissio
0	0	Maruti Wagon R LXI CNG	Mumbai	2010	72000	CNG	Manu:
1	1	Hyundai Creta 1.6 CRDi SX Option	Pune	2015	41000	Diesel	Manu:
2	2	Honda Jazz V	Chennai	2011	46000	Petrol	Manu:
3	3	Maruti Ertiga VDI	Chennai	2012	87000	Diesel	Manu:
4	4	Audi A4 New 2.0 TDI Multitronic	Coimbatore	2013	40670	Diesel	Automati

```
#First 5 observation print
df.head()
```

Saved successfully!

	Unnamed: 0	Name	Location	Year	Kilometers_Driven	Fuel_Type	Transmission	Owner_Type	Mileage
0	0	Maruti Wagon R LXI CNG	Mumbai	2010	72000	CNG	Manual	First	26.6 km/kg
1	1	Hyundai Creta 1.6 CRDi SX Option	Pune	2015	41000	Diesel	Manual	First	19.6 km/kg

```
#Last 5 observation print
df.tail()
```

	Unnamed: 0	Name	Location	Year	Kilometers_Driven	Fuel_Type	Transmission	Owner_Type	Mileage
6014	6014	Maruti Swift VDI	Delhi	2014	27365	Diesel	Manual	First	2 km/kg
6015	6015	Hyundai Xcent 1.1 CRDi S	Jaipur	2015	100000	Diesel	Manual	First	2 km/kg
6016	6016	Mahindra Xuv600	Jaipur	2012	55000	Diesel	Manual	Second	1 km/kg

```
#columns heading print
df.columns
```

```
Index(['Unnamed: 0', 'Name', 'Location', 'Year', 'Kilometers_Driven',
       'Fuel_Type', 'Transmission', 'Owner_Type', 'Mileage', 'Engine', 'Power',
       'Seats', 'New_Price', 'Price'],
      dtype='object')
```

```
#Each column type
df.dtypes
```

```
Unnamed: 0      int64
Name           object
Location       object
Year          int64
Kilometers_Driven  int64
Fuel_Type      object
Transmission    object
Owner_Type     object
Mileage        object
Engine         object
Power         object
Seats         float64
New_Price     object
Price         float64
dtype: object
```

```
#Each name count
```

```
namecount=df['Name'].value_counts()
```

Saved successfully!

```
Mahindra XUV500 W8 2WD      49
Maruti Swift VDI            45
Honda City 1.5 S MT        34
Maruti Swift Dzire VDI     34
Maruti Swift VDI BSIV      31
..
Ford Fiesta Titanium 1.5 TDCi 1
Mahindra Scorpio S10 AT 4WD 1
Hyundai i20 1.2 Era        1
Toyota Camry W4 (AT)       1
Mahindra Xylo D4 BSIV      1
Name: Name, Length: 1878, dtype: int64
```

```
#Each location count
```

```
Locationcount=df['Location'].value_counts()
```

```
Locationcount
```

```
Mumbai      790
Hyderabad   742
Kochi       651
Coimbatore  636
Pune        622
Delhi       554
Kolkata     535
Chennai     494
Jaipur      413
Bangalore   358
Ahmedabad   224
Name: Location, dtype: int64
```

```
#Graphical representation Each Locationscount pie chart
```

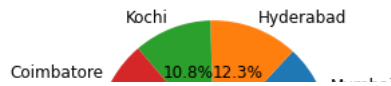
```
import matplotlib.pyplot as plt
```

```
Locationcount.plot(kind='pie', fontsize=12, autopct='%1.1f%%')
```

```
plt.title('locationcounts graph', color='green', fontsize=26)
```

```
Text(0.5, 1.0, 'locationcounts graph')
```

## locationcounts graph



```
#Each fueltype count
```

```
FuelTypecount=df['Fuel_Type'].value_counts()
```

```
FuelTypecount
```

```
Diesel      3205
Petrol      2746
CNG         56
LPG         10
Electric     2
Name: Fuel_Type, dtype: int64
```

```
#Graphical representation Each FuelTypecount horizontal bar graph
```

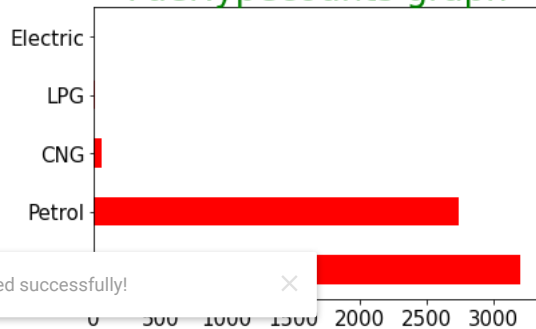
```
import matplotlib.pyplot as plt
```

```
FuelTypecount.plot(kind='barh',color='red',fontsize=15)
```

```
plt.title('FuelTypecounts graph',color='green',fontsize=26)
```

```
Text(0.5, 1.0, 'FuelTypecounts graph')
```

## FuelTypecounts graph



Saved successfully!

```
#Each Transmission count
```

```
Transmissioncount=df['Transmission'].value_counts()
```

```
Transmissioncount
```

```
Manual      4299
Automatic   1720
Name: Transmission, dtype: int64
```

```
#Graphical representation Each Transmission count Pie chart
```

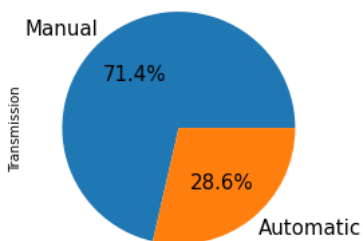
```
import matplotlib.pyplot as plt
```

```
Transmissioncount.plot(kind='pie',fontsize=15,autopct='%1.1f%%')
```

```
plt.title('Transmissioncount chart',color='red',fontsize=15)
```

```
Text(0.5, 1.0, 'Transmissioncount chart')
```

## Transmissioncount chart



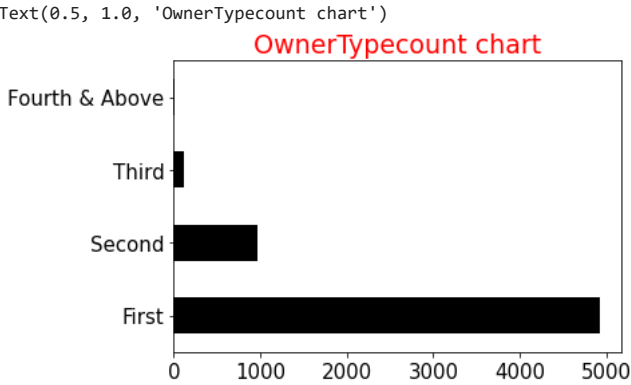
```
#Each ownertype count
```

```
OwnerTypecount=df['Owner_Type'].value_counts()
```

```
OwnerTypecount
```

```
First      4929
Second     968
Third      113
Fourth & Above 9
Name: Owner_Type, dtype: int64
```

```
#Graphical representation Each ownertype count horizontal bar graph
import matplotlib.pyplot as plt
OwnerTypecount.plot(kind='barh',color='black',fontsize=15)
plt.title('OwnerTypecount chart',color='red',fontsize=19)
```



```
#To find missing values
df.isna().sum()
```

```
Unnamed: 0      0
Name            0
Location        0
Year            0
Kilometers_Driven  0
Fuel_Type       0
Transmission     0
Owner_Type      0
Mileage         2
Engine          36
Power           36
Seats          42
```

Saved successfully! ✕

dtype: int64

Encoding using getdummies

```
df1=pd.get_dummies(df[['Location','Fuel_Type','Transmission','Owner_Type']],drop_first=True)
df1
```

	Location_Bangalore	Location_Chennai	Location_Coimbatore	Location_Delhi	Location_Hyderabad	Location_Mumbai
0	0	0	0	0	0	0
1	0	0	0	0	0	0
2	0	1	0	0	0	0
3	0	1	0	0	0	0
4	0	0	1	0	0	0
...	...	...	...	...	...	...
6014	0	0	0	1	0	0
6015	0	0	0	0	0	0
6016	0	0	0	0	0	0
6017	0	0	0	0	0	0
6018	0	0	0	0	0	1

6019 rows × 18 columns

#Concatination-combining old and new dataframe

```
#Concatination-combining old and new dataframe
dfe=pd.concat([df,df1],axis=1)
dfe
```

	Unnamed: 0	Name	Location	Year	Kilometers_Driven	Fuel_Type	Transmission	Owner_Type	Mileage
0	0	Maruti Wagon R LXI CNG	Mumbai	2010	72000	CNG	Manual	First	15.2 km/kg
1	1	Hyundai Creta 1.6 CRDi SX Option	Pune	2015	41000	Diesel	Manual	First	19.67 kmpl
2	2	Honda Jazz V	Chennai	2011	46000	Petrol	Manual	First	18.2 kmpl
3	3	Maruti Ertiga VDI	Chennai	2012	87000	Diesel	Manual	First	20.77 kmpl
4	4	Audi A4 New 2.0 TDI Multitronic	Coimbatore	2013	40670	Diesel	Automatic	Second	15.2 kmpl
...	...	...	...	...	...	...	...	...	...
6014	6014	Maruti Swift VDI	Delhi	2014	27365	Diesel	Manual	First	28.4 kmpl
6015	6015	Hyundai Xcent 1.1 CRDi S	Jaipur	2015	100000	Diesel	Manual	First	24.4 kmpl
6016	6016	Mahindra Xylo D4 BSIV	Jaipur	2012	55000	Diesel	Manual	Second	14.0 kmpl
6017	6017	Maruti Wagon R VXi	Kolkata	2013	46000	Petrol	Manual	First	18.9 kmpl
6018	6018	Chevrolet Tavera Neo LTZ	Hydrabad	2011	47000	Diesel	Manual	First	25.44 kmpl

Saved successfully! 6019 rows × 32 columns

```
#dropping unwanted columns
dfe.drop(['Unnamed: 0', 'Name', 'Location', 'Fuel_Type',
          'Transmission', 'Owner_Type', 'New_Price'],axis=1,inplace=True)
dfe
```

	Year	Kilometers_Driven	Mileage	Engine	Power	Seats	Price	Location_Bangalore	Location_Chennai
0	2010	72000	26.6 km/kg	998 CC	58.16 bhp	5.0	1.75	0	
1	2015	41000	19.67 kmpl	1582 CC	126.2 bhp	5.0	12.50	0	
2	2011	46000	18.2 kmpl	1199 CC	88.7 bhp	5.0	4.50	0	
3	2012	87000	20.77 kmpl	1248 CC	88.76 bhp	7.0	6.00	0	
4	2013	40670	15.2 kmpl	1968 CC	140.8 bhp	5.0	17.74	0	
...	...	...	...	...	...	...	...	...	...
6014	2014	27365	28.4 kmpl	1248 CC	74 bhp	5.0	4.75	0	
6015	2015	100000	24.4 kmpl	1120 CC	71 bhp	5.0	4.00	0	
6016	2012	55000	14.0 kmpl	2498 CC	112 bhp	8.0	2.90	0	
6017	2013	46000	18.9 kmpl	998 CC	67.1 bhp	5.0	2.65	0	
6018	2011	47000	25.44 kmpl	936 CC	57.6 bhp	5.0	2.50	0	

6019 rows × 25 columns

Remove string characters

```
dfe['Mileage']=dfe['Mileage'].str.replace('km/kg','')
dfe['Mileage']=dfe['Mileage'].str.replace('kmpl','')
dfe['Engine']=dfe['Engine'].str.replace('CC','')
dfe['Power']=dfe['Power'].str.replace('bhp','')
dfe
```

	Year	Kilometers_Driven	Mileage	Engine	Power	Seats	Price	Location_Bangalore	Location_Chenn
0	2010	72000	26.6	998	58.16	5.0	1.75	0	
1	2015	41000	19.67	1582	126.2	5.0	12.50	0	
2	2011	46000	18.2	1199	88.7	5.0	4.50	0	
3	2012	87000	20.77	1248	88.76	7.0	6.00	0	
4	2013	40670	15.2	1968	140.8	5.0	17.74	0	
...	...	...	...	...	...	...	...	...	...
6014	2014	27365	28.4	1248	74	5.0	4.75	0	
6015	2015	100000	24.4	1120	71	5.0	4.00	0	
6016	2012	55000	14.0	2498	112	8.0	2.90	0	
6017	2013	46000	18.9	998	67.1	5.0	2.65	0	
6018	2011	47000	25.44	936	57.6	5.0	2.50	0	

Saved successfully!

×

replace null with zero(0)

```
dfe['Mileage']=dfe['Mileage'].str.replace('null','0')
dfe['Engine']=dfe['Engine'].str.replace('null','0')
dfe['Power']=dfe['Power'].str.replace('null','0')
dfe
```

	Year	Kilometers_Driven	Mileage	Engine	Power	Seats	Price	Location_Bangalore	Location_Chenn
0	2010	72000	26.6	998	58.16	5.0	1.75	0	
1	2015	41000	19.67	1582	126.2	5.0	12.50	0	
2	2011	46000	18.2	1199	88.7	5.0	4.50	0	
3	2012	87000	20.77	1248	88.76	7.0	6.00	0	
4	2013	40670	15.2	1968	140.8	5.0	17.74	0	
...	...	...	...	...	...	...	...	...	...
6014	2014	27365	28.4	1248	74	5.0	4.75	0	
6015	2015	100000	24.4	1120	71	5.0	4.00	0	
6016	2012	55000	14.0	2498	112	8.0	2.90	0	
6017	2013	46000	18.9	998	67.1	5.0	2.65	0	
6018	2011	47000	25.44	936	57.6	5.0	2.50	0	

6019 rows × 25 columns

dfe.dtypes

Year	int64
Kilometers_Driven	int64
Mileage	object

```
Engine          object
Power           object
Seats           float64
Price           float64
Location_Bangalore  uint8
Location_Chennai   uint8
Location_Coimbatore uint8
Location_Delhi     uint8
Location_Hyderabad uint8
Location_Jaipur    uint8
Location_Kochi     uint8
Location_Kolkata   uint8
Location_Mumbai    uint8
Location_Pune      uint8
Fuel_Type_Diesel   uint8
Fuel_Type_Electric uint8
Fuel_Type_LPG      uint8
Fuel_Type_Petrol   uint8
Transmission_Manual uint8
Owner_Type_Fourth & Above uint8
Owner_Type_Second  uint8
Owner_Type_Third   uint8
dtype: object
```

converting object to float

```
dfe['Mileage']=dfe['Mileage'].astype(float)
dfe['Engine']=dfe['Engine'].astype(float)
dfe['Power']=dfe['Power'].astype(float)
```

dfe.dtypes

```
Year          int64
Kilometers_Driven int64
Mileage       float64
Engine        float64
Price         float64
Location_Bangalore  uint8
Location_Chennai   uint8
Location_Coimbatore uint8
Location_Delhi     uint8
Location_Hyderabad uint8
Location_Jaipur    uint8
Location_Kochi     uint8
Location_Kolkata   uint8
Location_Mumbai    uint8
Location_Pune      uint8
Fuel_Type_Diesel   uint8
Fuel_Type_Electric uint8
Fuel_Type_LPG      uint8
Fuel_Type_Petrol   uint8
Transmission_Manual uint8
Owner_Type_Fourth & Above uint8
Owner_Type_Second  uint8
Owner_Type_Third   uint8
dtype: object
```

Saved successfully!

×

Engine,Power,Mileage >>>null==>>>0

```
dfe.loc[dfe.Mileage==0, 'Mileage']=np.NaN
dfe.loc[dfe.Engine==0, 'Engine']=np.NaN
dfe.loc[dfe.Power==0, 'Power']=np.NaN
```

#To find missing values  
dfe.isna().sum()

```
Year          0
Kilometers_Driven 0
Mileage       70
Engine        36
Power        143
Seats         42
Price         0
Location_Bangalore 0
Location_Chennai   0
Location_Coimbatore 0
Location_Delhi     0
Location_Hyderabad 0
Location_Jaipur    0
Location_Kochi     0
```

```
Location_Kolkata      0
Location_Mumbai       0
Location_Pune         0
Fuel_Type_Diesel      0
Fuel_Type_Electric    0
Fuel_Type_LPG         0
Fuel_Type_Petrol      0
Transmission_Manual   0
Owner_Type_Fourth & Above 0
Owner_Type_Second     0
Owner_Type_Third      0
dtype: int64
```

Missing values filling

```
dfe['Mileage']=dfe['Mileage'].fillna(dfe['Mileage'].mean())
dfe['Engine']=dfe['Engine'].fillna(dfe['Engine'].mean())
dfe['Power']=dfe['Power'].fillna(dfe['Power'].mean())
dfe['Seats']=dfe['Seats'].fillna(dfe['Seats'].mode()[0])
```

```
dfe.isna().sum()
```

```
Year      0
Kilometers_Driven  0
Mileage    0
Engine     0
Power      0
Seats      0
Price      0
Location_Bangalore  0
Location_Chennai  0
Location_Coimbatore  0
Location_Delhi  0
Location_Hyderabad  0
Location_Jaipur  0
Location_Kochi  0
```

Saved successfully!

×

```
Location_Pune      0
Fuel_Type_Diesel   0
Fuel_Type_Electric 0
Fuel_Type_LPG      0
Fuel_Type_Petrol   0
Transmission_Manual 0
Owner_Type_Fourth & Above 0
Owner_Type_Second  0
Owner_Type_Third   0
dtype: int64
```

```
#Seperate x
x=dfe.drop(['Price'],axis=1)
x
```

	Year	Kilometers_Driven	Mileage	Engine	Power	Seats	Location_Bangalore	Location_Chennai	Location_Kochi
0	2010	72000	26.60	998.0	58.16	5.0	0	0	0
1	2015	41000	19.67	1582.0	126.20	5.0	0	0	0
2	2011	46000	18.20	1199.0	88.70	5.0	0	1	0
3	2012	87000	20.77	1248.0	88.76	7.0	0	1	0
4	2013	40670	15.20	1968.0	140.80	5.0	0	0	0
...	...	...	...	...	...	...	...	...	...
6014	2014	27365	28.40	1248.0	74.00	5.0	0	0	0
6015	2015	100000	24.40	1120.0	71.00	5.0	0	0	0
6016	2012	55000	14.00	2498.0	112.00	8.0	0	0	0
6017	2013	46000	18.90	998.0	67.10	5.0	0	0	0
6018	2011	47000	25.44	936.0	57.60	5.0	0	0	0

6019 rows × 24 columns



```
#Seperate y
y=dfe['Price']
y

0      1.75
1     12.50
2      4.50
3      6.00
4     17.74
...
6014    4.75
6015    4.00
6016    2.90
6017    2.65
6018    2.50
Name: Price, Length: 6019, dtype: float64
```

PREPROCESSING TESTING DATA :CAR ECONOMY PRICE FILE

```
import numpy as np
import pandas as pd
df2=pd.read_csv('/content/test-data.csv')
df2
```

	Unnamed: 0	Name	Location	Year	Kilometers_Driven	Fuel_Type	Transmission	Owner_Type	Mi
0	0	Maruti Alto K10 LXI CNG	Delhi	2014	40929	CNG	Manual	First	
1	1	Maruti Alto 800 2016-2019 LXI	Coimbatore	2013	54493	Petrol	Manual	Second	
2	2	Hyundai i20 Nios Sport 2.4 MT	Mumbai	2017	34000	Diesel	Manual	First	
3	3	Toyota Etios Liva GD	Hyderabad	2012	139000	Diesel	Manual	First	
4	4	Hyundai i20 Magna	Mumbai	2014	29000	Petrol	Manual	First	
...	...	...	...	...	...	...	...	...	
1229	1229	Volkswagen Vento Diesel Trendline	Hyderabad	2011	89411	Diesel	Manual	First	
1230	1230	Volkswagen Polo GT TSI	Mumbai	2015	59000	Petrol	Automatic	First	

Saved successfully!

×

```
#First 5 observation print
df2.head()
```

	Unnamed: 0	Name	Location	Year	Kilometers_Driven	Fuel_Type	Transmission	Owner_Type	Mileage
0	0	Maruti Alto K10 LXI CNG	Delhi	2014	40929	CNG	Manual	First	32.26 km/kg
1	1	Maruti Alto 800 2016-2019 LXI	Coimbatore	2013	54493	Petrol	Manual	Second	24.7 kmpl

```
#Last 5 observation print
df2.tail()
```

	Unnamed: 0	Name	Location	Year	Kilometers_Driven	Fuel_Type	Transmission	Owner_Type	Mileage
1229	1229	Volkswagen Vento Diesel Trendline	Hyderabad	2011	89411	Diesel	Manual	First	
1230	1230	Volkswagen Polo GT TSI	Mumbai	2015	59000	Petrol	Automatic	First	

```
#Column Heading print
df2.columns
```

Index(['Unnamed: 0', 'Name', 'Location', 'Year', 'Kilometers\_Driven', 'Fuel\_Type', 'Transmission', 'Owner\_Type', 'Mileage', 'Engine', 'Power', 'Seats', 'New\_Price'], dtype='object', name='columns')

Saved successfully!

X

```
#Each column type
df2.dtypes
```

Unnamed: 0 int64  
Name object  
Location object  
Year int64  
Kilometers\_Driven int64  
Fuel\_Type object  
Transmission object  
Owner\_Type object  
Mileage object  
Engine object  
Power object  
Seats float64  
New\_Price object  
dtype: object

```
#To find missing values
df2.isna().sum()
```

Unnamed: 0 0  
Name 0  
Location 0  
Year 0  
Kilometers\_Driven 0  
Fuel\_Type 0  
Transmission 0  
Owner\_Type 0  
Mileage 0  
Engine 10  
Power 10  
Seats 11  
New\_Price 1052  
dtype: int64

```
#Each name count
namecount1=df2['Name'].value_counts()
namecount1
```

Maruti Alto LXi 9  
Honda City 1.5 V MT 8  
Maruti Swift Dzire VDI 8

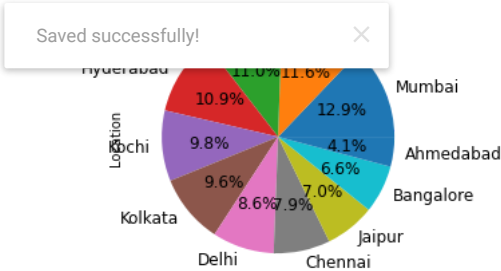
```
Volkswagen Polo 1.2 MPI Highline      8
Hyundai i10 Magna                      7
..
Hyundai Santro GLS I - Euro II        1
Honda City i DTEC VX Option BL        1
Land Rover Discovery 4 SDV6 SE         1
Hyundai Verna CRDi 1.6 SX Option       1
Mercedes-Benz E-Class 2009-2013 E 220 CDI Avantgarde  1
Name: Name, Length: 769, dtype: int64
```

```
#Each Location count
Locationcount1=df2['Location'].value_counts()
Locationcount1
```

```
Mumbai      159
Pune        143
Coimbatore   136
Hyderabad    134
Kochi        121
Kolkata      119
Delhi        106
Chennai      97
Jaipur       86
Bangalore    82
Ahmedabad    51
Name: Location, dtype: int64
```

```
#Graphical representation Each location count pie chart
import matplotlib.pyplot as plt
Locationcount1.plot(kind='pie',fontsize=12,autopct='%1.1f%%')
plt.title('Locationcount1 Chart',color='red',fontsize=18)
```

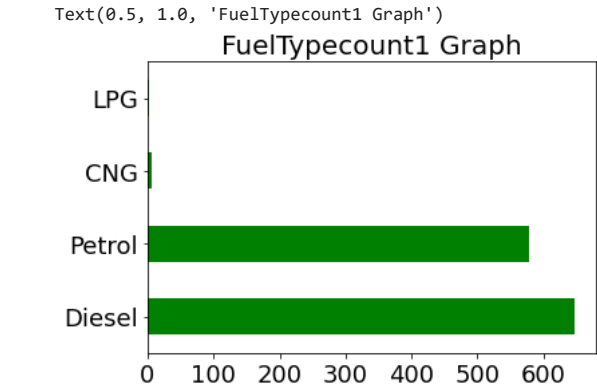
```
Text(0.5, 1.0, 'Locationcount1 Chart')
Locationcount1 Chart
```



```
#Each Fuel_Type count
FuelTypecount1=df2['Fuel_Type'].value_counts()
FuelTypecount1
```

```
Diesel      647
Petrol       579
CNG           6
LPG           2
Name: Fuel_Type, dtype: int64
```

```
#Graphical representation Each fueltype count horizontal bar graph
import matplotlib.pyplot as plt
FuelTypecount1.plot(kind='barh',color='green',fontsize=18)
plt.title('FuelTypecount1 Graph',color='black',fontsize=20)
```



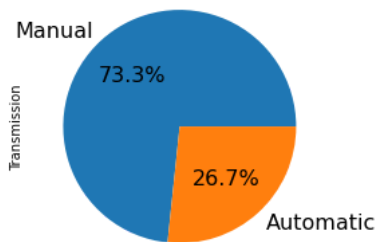
```
#Each Transmission count
Transmissioncount1=df2['Transmission'].value_counts()
Transmissioncount1
```

```
Manual      905
Automatic   329
Name: Transmission, dtype: int64
```

```
#Graphical representation Each transmission count pie chart
import matplotlib.pyplot as plt
Transmissioncount1.plot(kind='pie',fontsize=16,autopct='%1.1f%%')
plt.title('Transmissioncount1 Chart',color='brown',fontsize=19)
```

```
Text(0.5, 1.0, 'Transmissioncount1 Chart')
```

**Transmissioncount1 Chart**



```
#EACH Owner_Type count
OwnerTypecount1=df2['Owner_Type'].value_counts()
OwnerTypecount1
```

```
First      1023
```

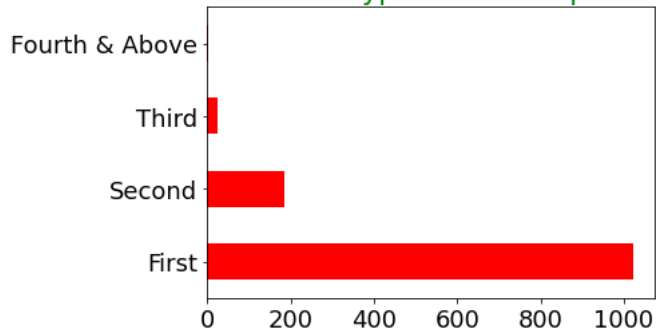
Saved successfully!

```
Name: Owner_Type, dtype: int64
```

```
#Graphical representation ownertype count horizontal bar graph
import matplotlib.pyplot as plt
OwnerTypecount1.plot(kind='barh',color='red',fontsize=18)
plt.title('OwnerTypecount1 Graph',color='green',fontsize=21)
```

```
Text(0.5, 1.0, 'OwnerTypecount1 Graph')
```

**OwnerTypecount1 Graph**



### Encoding using getdummies

```
df3=pd.get_dummies(df2[['Location','Fuel_Type','Transmission','Owner_Type']],drop_first=True)
df3
```

	Location_Bangalore	Location_Chennai	Location_Coimbatore	Location_Delhi	Location_Hyderabad	Location_Kolkata
0	0	0	0	1	0	0
1	0	0	1	0	0	0
2	0	0	0	0	0	0
3	0	0	0	0	0	1
4	0	0	0	0	0	0
...	...	...	...	...	...	...
1229	0	0	0	0	0	1

Concatination

1231	0	0	0	0	0	0
------	---	---	---	---	---	---

```
dfe2=pd.concat([df2,df3],axis=1)
dfe2
```

	Unnamed: 0	Name	Location	Year	Kilometers_Driven	Fuel_Type	Transmission	Owner_Type	Mileage
0	0	Maruti Alto K10 LXI CNG	Delhi	2014	40929	CNG	Manual	First	
1	1	Maruti Alto 800 2016-2019 LXI	Coimbatore	2013	54493	Petrol	Manual	Second	
2	2	Toyota Innova Crysta Touring Sport 2.4 MT	Mumbai	2017	34000	Diesel	Manual	First	
3	3	Hyundai Ecos Live GD	Hyderabad	2012	139000	Diesel	Manual	First	
4	4	Hyundai i20 Magna	Mumbai	2014	29000	Petrol	Manual	First	
...	...	...	...	...	...	...	...	...	
1229	1229	Volkswagen Vento Diesel Trendline	Hyderabad	2011	89411	Diesel	Manual	First	
1230	1230	Volkswagen Polo GT TSI	Mumbai	2015	59000	Petrol	Automatic	First	
1231	1231	Nissan Micra Diesel XV	Kolkata	2012	28000	Diesel	Manual	First	
1232	1232	Volkswagen Polo GT TSI	Pune	2013	52262	Petrol	Automatic	Third	
1233	1233	Mercedes-Benz E-Class 2009-2013 E 220 CDI Avantgarde	Kochi	2014	72443	Diesel	Automatic	First	

1234 rows × 30 columns

```
dfe2.isna().sum()
```

Unnamed: 0	0
Name	0
Location	0
Year	0
Kilometers_Driven	0
Fuel_Type	0
Transmission	0
Owner_Type	0
Mileage	0
Engine	10

```
Power          10
Seats          11
New_Price      1052
Location_Bangalore  0
Location_Chennai  0
Location_Coimbatore  0
Location_Delhi  0
Location_Hyderabad  0
Location_Jaipur  0
Location_Kochi  0
Location_Kolkata  0
Location_Mumbai  0
Location_Pune  0
Fuel_Type_Diesel  0
Fuel_Type_LPG  0
Fuel_Type_Petrol  0
Transmission_Manual  0
Owner_Type_Fourth & Above  0
Owner_Type_Second  0
Owner_Type_Third  0
dtype: int64
```

```
#Dropping unwanted columns
dfe2.drop(['Unnamed: 0', 'Name', 'Location', 'Fuel_Type',
          'Transmission', 'Owner_Type', 'New_Price'],axis=1,inplace=True)
dfe2
```

	Year	Kilometers_Driven	Mileage	Engine	Power	Seats	Location_Bangalore	Location_Chennai	Loc
0	2014	40929	32.26 km/kg	998 CC	58.2 bhp	4.0	0	0	
1	2013	54493	24.7 kmpl	796 CC	47.3 bhp	5.0	0	0	
2	2017	34000	13.68 kmpl	2393 CC	147.8 bhp	7.0	0	0	
			23.59 kmpl	1364 CC	null bhp	5.0	0	0	
4	2014	29000	18.5 kmpl	1197 CC	82.85 bhp	5.0	0	0	
...	...	...	...	...	...	...	...	...	
1229	2011	89411	20.54 kmpl	1598 CC	103.6 bhp	5.0	0	0	
1230	2015	59000	17.21 kmpl	1197 CC	103.6 bhp	5.0	0	0	
1231	2012	28000	23.08 kmpl	1461 CC	63.1 bhp	5.0	0	0	
1232	2013	52262	17.2 kmpl	1197 CC	103.6 bhp	5.0	0	0	
1233	2014	72443	10.0 kmpl	2148 CC	170 bhp	5.0	0	0	

1234 rows × 23 columns

Replace string values

```
dfe2['Mileage']=dfe2['Mileage'].str.replace('km/kg','')
dfe2['Mileage']=dfe2['Mileage'].str.replace('kmpl','')
dfe2['Engine']=dfe2['Engine'].str.replace('CC','')
dfe2['Power']=dfe2['Power'].str.replace('bhp','')
dfe2
```

	Year	Kilometers_Driven	Mileage	Engine	Power	Seats	Location_Bangalore	Location_Chennai	Loc
0	2014	40929	32.26	998	58.2	4.0	0	0	
1	2013	54493	24.7	796	47.3	5.0	0	0	
2	2017	34000	13.68	2393	147.8	7.0	0	0	
3	2012	139000	23.59	1364	null	5.0	0	0	
4	2014	29000	18.5	1197	82.85	5.0	0	0	
...	...	...	...	...	...	...	...	...	...
1229	2011	89411	20.54	1598	103.6	5.0	0	0	
1230	2015	59000	17.21	1197	103.6	5.0	0	0	
1231	2012	28000	23.08	1461	63.1	5.0	0	0	

Repace null with zero(0)

1233	2014	72443	10.0	2148	170	5.0	0	0	
------	------	-------	------	------	-----	-----	---	---	--

```
dfe2['Mileage']=dfe2['Mileage'].str.replace('null','0')
dfe2['Engine']=dfe2['Engine'].str.replace('null','0')
dfe2['Power']=dfe2['Power'].str.replace('null','0')
dfe2
```

	Year	Kilometers_Driven	Mileage	Engine	Power	Seats	Location_Bangalore	Location_Chennai	Loc
0	2014	40929	32.26	998	58.2	4.0	0	0	
1	2013	54493	24.7	796	47.3	5.0	0	0	
2	2017	34000	13.68	2393	147.8	7.0	0	0	
3	2012	139000	23.59	1364	0	5.0	0	0	
			18.5	1197	82.85	5.0	0	0	
			...	...	...	...	...	...	...
1229	2011	89411	20.54	1598	103.6	5.0	0	0	
1230	2015	59000	17.21	1197	103.6	5.0	0	0	
1231	2012	28000	23.08	1461	63.1	5.0	0	0	
1232	2013	52262	17.2	1197	103.6	5.0	0	0	
1233	2014	72443	10.0	2148	170	5.0	0	0	

Saved successfully!

×

1234 rows × 23 columns

convert object type to float type

```
dfe2['Mileage']=dfe2['Mileage'].astype(float)
dfe2['Engine']=dfe2['Engine'].astype(float)
dfe2['Power']=dfe2['Power'].astype(float)
```

dfe2.dtypes

Year	int64
Kilometers_Driven	int64
Mileage	float64
Engine	float64
Power	float64
Seats	float64
Location_Bangalore	uint8
Location_Chennai	uint8
Location_Coimbatore	uint8
Location_Delhi	uint8
Location_Hyderabad	uint8
Location_Jaipur	uint8
Location_Kochi	uint8
Location_Kolkata	uint8
Location_Mumbai	uint8
Location_Pune	uint8
Fuel_Type_Diesel	uint8
Fuel_Type_LPG	uint8
Fuel_Type_Petrol	uint8
Transmission_Manual	uint8

```

Owner_Type_Fourth & Above    uint8
Owner_Type_Second           uint8
Owner_Type_Third            uint8
dtype: object

```

### Engine,Power,Mileage >>>null==>>>0

```

dfe2.loc[dfe2.Engine==0, 'Engine']=np.NaN
dfe2.loc[dfe2.Power==0, 'Power']=np.NaN
dfe2.loc[dfe2.Mileage==0, 'Mileage']=np.NaN

```

```
dfe2.isna().sum()
```

```

Year                0
Kilometers_Driven   0
Mileage             13
Engine             10
Power              32
Seats              11
Location_Bangalore  0
Location_Chennai    0
Location_Coimbatore  0
Location_Delhi       0
Location_Hyderabad   0
Location_Jaipur       0
Location_Kochi        0
Location_Kolkata      0
Location_Mumbai       0
Location_Pune         0
Fuel_Type_Diesel      0
Fuel_Type_LPG         0
Fuel_Type_Petrol      0
Transmission_Manual   0
Owner_Type_Fourth & Above  0
Owner_Type_Second     0
Owner_Type_Third      0

```

Saved successfully!

### Filling Missing values

```

dfe2['Engine']=dfe2['Engine'].fillna(dfe2['Engine'].mean())
dfe2['Mileage']=dfe2['Mileage'].fillna(dfe2['Mileage'].mean())
dfe2['Power']=dfe2['Power'].fillna(dfe2['Power'].mean())
dfe2['Seats']=dfe2['Seats'].fillna(dfe2['Seats'].mode()[0])

```

```
dfe2.isna().sum()
```

```

Year                0
Kilometers_Driven   0
Mileage             0
Engine             0
Power              0
Seats              0
Location_Bangalore  0
Location_Chennai    0
Location_Coimbatore  0
Location_Delhi       0
Location_Hyderabad   0
Location_Jaipur       0
Location_Kochi        0
Location_Kolkata      0
Location_Mumbai       0
Location_Pune         0
Fuel_Type_Diesel      0
Fuel_Type_LPG         0
Fuel_Type_Petrol      0
Transmission_Manual   0
Owner_Type_Fourth & Above  0
Owner_Type_Second     0
Owner_Type_Third      0
dtype: int64

```

```

#seperate x_test
x_test=dfe2
x_test

```



	Year	Kilometers_Driven	Mileage	Engine	Power	Seats	Location_Bangalore	Location_Chennai
0	2014	40929	32.26	998.0	58.20000	4.0	0	0
1	2013	54493	24.70	796.0	47.30000	5.0	0	0
2	2017	34000	13.68	2393.0	147.80000	7.0	0	0
3	2012	139000	23.59	1364.0	110.38042	5.0	0	0
4	2014	29000	18.50	1197.0	82.85000	5.0	0	0
...	...	...	...	...	...	...	...	...
1229	2011	89411	20.54	1598.0	103.60000	5.0	0	0
1230	2015	59000	17.21	1197.0	103.60000	5.0	0	0
1231	2012	28000	23.08	1461.0	63.10000	5.0	0	0
1232	2013	52262	17.20	1197.0	103.60000	5.0	0	0
1233	2014	72443	10.00	2148.0	170.00000	5.0	0	0

1234 rows × 23 columns

```
#First 5 observation Testing data
x_test.head()
```

	Year	Kilometers_Driven	Mileage	Engine	Power	Seats	Location_Bangalore	Location_Chennai	Location_Other
0	2014	40929	32.26	998.0	58.20000	4.0	0	0	0
1	2013	54493	24.70	796.0	47.30000	5.0	0	0	0
2	2017	34000	13.68	2393.0	147.80000	7.0	0	0	0
3	2012	139000	23.59	1364.0	110.38042	5.0	0	0	0
4	2014	29000	18.50	1197.0	82.85000	5.0	0	0	0

Saved successfully!

✕

```
#Drop Fuel type electric column to make training testing same no of columns
x=x.drop(['Fuel_Type_Electric'],axis=1)
x
```

	Year	Kilometers_Driven	Mileage	Engine	Power	Seats	Location_Bangalore	Location_Chennai	Location_Other
0	2010	72000	26.60	998.0	58.16	5.0	0	0	0
1	2015	41000	19.67	1582.0	126.20	5.0	0	0	0
2	2011	46000	18.20	1199.0	88.70	5.0	0	1	0
3	2012	87000	20.77	1248.0	88.76	7.0	0	1	0
4	2013	40670	15.20	1968.0	140.80	5.0	0	0	0
...	...	...	...	...	...	...	...	...	...
6014	2014	27365	28.40	1248.0	74.00	5.0	0	0	0
6015	2015	100000	24.40	1120.0	71.00	5.0	0	0	0
6016	2012	55000	14.00	2498.0	112.00	8.0	0	0	0
6017	2013	46000	18.90	998.0	67.10	5.0	0	0	0
6018	2011	47000	25.44	936.0	57.60	5.0	0	0	0

6019 rows × 23 columns

Model creation

```
from sklearn.linear_model import LinearRegression
model=LinearRegression()
model.fit(x,y)
```

```
y_pred=model.predict(x_test)
y_pred
array([ 2.87588492, -1.29344912, 16.1069494 , ...,  0.1378514 ,
        9.27293255, 21.48043251])
```



Saved successfully!

