In [2]:

```
!pip install numpy
```

```
Requirement already satisfied: numpy in c:\users\rahul\appdata\local\programs\pyt
hon\python312\lib\site-packages (1.26.3)
```

In [3]:

```
!pip install pandas
```

```
Requirement already satisfied: pandas in c:\users\rahul\appdata\local\programs\py
thon\python312\lib\site-packages (2.2.0)
Requirement already satisfied: numpy<2,>=1.26.0 in c:\users\rahul\appdata\local\p
rograms\python\python312\lib\site-packages (from pandas) (1.26.3)
Requirement already satisfied: python-dateutil>=2.8.2 in c:\users\rahul\appdata\l
ocal\programs\python\python312\lib\site-packages (from pandas) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in c:\users\rahul\appdata\local\progr
ams\python\python312\lib\site-packages (from pandas) (2023.3.post1)
Requirement already satisfied: tzdata>=2022.7 in c:\users\rahul\appdata\local\pro
grams\python\python312\lib\site-packages (from pandas) (2023.4)
Requirement already satisfied: six>=1.5 in c:\users\rahul\appdata\local\programs
\python\python312\lib\site-packages (from python-dateutil>=2.8.2->pandas) (1.16.
0)
```

In [4]:

```
!pip install scikit-learn
```

```
Requirement already satisfied: scikit-learn in c:\users\rahul\appdata\local\progr
ams\python\python312\lib\site-packages (1.5.1)
Requirement already satisfied: numpy>=1.19.5 in c:\users\rahul\appdata\local\prog
rams\python\python312\lib\site-packages (from scikit-learn) (1.26.3)
Requirement already satisfied: scipy>=1.6.0 in c:\users\rahul\appdata\local\progr
ams\python\python312\lib\site-packages (from scikit-learn) (1.13.0)
Requirement already satisfied: joblib>=1.2.0 in c:\users\rahul\appdata\local\prog
rams\python\python312\lib\site-packages (from scikit-learn) (1.4.2)
Requirement already satisfied: threadpoolctl>=3.1.0 in c:\users\rahul\appdata\loc
al\programs\python\python312\lib\site-packages (from scikit-learn) (3.5.0)
```

In [5]:

```
!pip install streamlit
```

```
Requirement already satisfied: streamlit in c:\users\rahul\appdata\local\programs
\python\python312\lib\site-packages (1.38.0)
Requirement already satisfied: altair<6,>=4.0 in c:\users\rahul\appdata\local\pro
grams\python\python312\lib\site-packages (from streamlit) (5.4.1)
Requirement already satisfied: blinker<2,>=1.0.0 in c:\users\rahul\appdata\local
\programs\python\python312\lib\site-packages (from streamlit) (1.8.2)
Requirement already satisfied: cachetools<6,>=4.0 in c:\users\rahul\appdata\local
\programs\python\python312\lib\site-packages (from streamlit) (5.5.0)
Requirement already satisfied: click<9,>=7.0 in c:\users\rahul\appdata\local\prog
rams\python\python312\lib\site-packages (from streamlit) (8.1.7)
Requirement already satisfied: numpy<3,>=1.20 in c:\users\rahul\appdata\local\pro
grams\python\python312\lib\site-packages (from streamlit) (1.26.3)
Requirement already satisfied: packaging<25,>=20 in c:\users\rahul\appdata\local
\programs\python\python312\lib\site-packages (from streamlit) (23.2)
Requirement already satisfied: pandas<3,>=1.3.0 in c:\users\rahul\appdata\local\p
rograms\python\python312\lib\site-packages (from streamlit) (2.2.0)
Requirement already satisfied: pillow<11,>=7.1.0 in c:\users\rahul\appdata\local
\programs\python\python312\lib\site-packages (from streamlit) (10.2.0)
Requirement already satisfied: protobuf<6,>=3.20 in c:\users\rahul\appdata\local
\programs\python\python312\lib\site-packages (from streamlit) (5.28.2)
Requirement already satisfied: pyarrow>=7.0 in c:\users\rahul\appdata\local\progr
ams\python\python312\lib\site-packages (from streamlit) (17.0.0)
Requirement already satisfied: requests<3,>=2.27 in c:\users\rahul\appdata\local
\programs\python\python312\lib\site-packages (from streamlit) (2.31.0)
Requirement already satisfied: rich<14,>=10.14.0 in c:\users\rahul\appdata\local
\programs\python\python312\lib\site-packages (from streamlit) (13.8.1)
Requirement already satisfied: tenacity<9,>=8.1.0 in c:\users\rahul\appdata\local
\programs\python\python312\lib\site-packages (from streamlit) (8.5.0)
Requirement already satisfied: toml<2,>=0.10.1 in c:\users\rahul\appdata\local\pr
ograms\python\python312\lib\site-packages (from streamlit) (0.10.2)
Requirement already satisfied: typing-extensions<5,>=4.3.0 in c:\users\rahul\appd
ata\local\programs\python\python312\lib\site-packages (from streamlit) (4.12.2)
Requirement already satisfied: gitpython!=3.1.19,<4,>=3.0.7 in c:\users\rahul\app
data\local\programs\python\python312\lib\site-packages (from streamlit) (3.1.43)
Requirement already satisfied: pydeck<1,>=0.8.0b4 in c:\users\rahul\appdata\local
\programs\python\python312\lib\site-packages (from streamlit) (0.9.1)
Requirement already satisfied: tornado<7,>=6.0.3 in c:\users\rahul\appdata\local
\programs\python\python312\lib\site-packages (from streamlit) (6.4)
Requirement already satisfied: watchdog<5,>=2.1.5 in c:\users\rahul\appdata\local
\programs\python\python312\lib\site-packages (from streamlit) (4.0.2)
Requirement already satisfied: jinja2 in c:\users\rahul\appdata\local\programs\py
thon\python312\lib\site-packages (from altair<6,>=4.0->streamlit) (3.1.2)
Requirement already satisfied: jsonschema>=3.0 in c:\users\rahul\appdata\local\pr
ograms\python\python312\lib\site-packages (from altair<6,>=4.0->streamlit) (4.20.
0)
Requirement already satisfied: narwhals>=1.5.2 in c:\users\rahul\appdata\local\pr
ograms\python\python312\lib\site-packages (from altair<6,>=4.0->streamlit) (1.8.
2)
Requirement already satisfied: colorama in c:\users\rahul\appdata\local\programs
\python\python312\lib\site-packages (from click<9,>=7.0->streamlit) (0.4.6)
Requirement already satisfied: gitdb<5,>=4.0.1 in c:\users\rahul\appdata\local\pr
ograms\python\python312\lib\site-packages (from gitpython!=3.1.19,<4,>=3.0.7->str
eamlit) (4.0.11)
Requirement already satisfied: python-dateutil>=2.8.2 in c:\users\rahul\appdata\l
ocal\programs\python\python312\lib\site-packages (from pandas<3,>=1.3.0->streamli
t) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in c:\users\rahul\appdata\local\progr
ams\python\python312\lib\site-packages (from pandas<3,>=1.3.0->streamlit) (2023.
3.post1)
Requirement already satisfied: tzdata>=2022.7 in c:\users\rahul\appdata\local\pro
```

```
grams\python\python312\lib\site-packages (from pandas<3,>=1.3.0->streamlit) (202
3.4)
Requirement already satisfied: charset-normalizer<4,>=2 in c:\users\rahul\appdata
\local\programs\python\python312\lib\site-packages (from requests<3,>=2.27->strea
mlit) (3.3.2)
Requirement already satisfied: idna<4,>=2.5 in c:\users\rahul\appdata\local\progr
ams\python\python312\lib\site-packages (from requests<3,>=2.27->streamlit) (3.6)
Requirement already satisfied: urllib3<3,>=1.21.1 in c:\users\rahul\appdata\local
\programs\python\python312\lib\site-packages (from requests<3,>=2.27->streamlit)
(2.1.0)
Requirement already satisfied: certifi>=2017.4.17 in c:\users\rahul\appdata\local
\programs\python\python312\lib\site-packages (from requests<3,>=2.27->streamlit)
(2023.11.17)
Requirement already satisfied: markdown-it-py>=2.2.0 in c:\users\rahul\appdata\lo
cal\programs\python\python312\lib\site-packages (from rich<14,>=10.14.0->streamli
t) (3.0.0)
Requirement already satisfied: pygments<3.0.0,>=2.13.0 in c:\users\rahul\appdata
\local\programs\python\python312\lib\site-packages (from rich<14,>=10.14.0->strea
mlit) (2.17.2)
Requirement already satisfied: smmap<6,>=3.0.1 in c:\users\rahul\appdata\local\pr
ograms\python\python312\lib\site-packages (from gitdb<5,>=4.0.1->gitpython!=3.1.1
9,<4,>=3.0.7->streamlit) (5.0.1)
Requirement already satisfied: MarkupSafe>=2.0 in c:\users\rahul\appdata\local\pr
ograms\python\python312\lib\site-packages (from jinja2->altair<6,>=4.0->streamli
t) (2.1.3)
Requirement already satisfied: attrs>=22.2.0 in c:\users\rahul\appdata\local\prog
rams\python\python312\lib\site-packages (from jsonschema>=3.0->altair<6,>=4.0->st
reamlit) (23.1.0)
Requirement already satisfied: jsonschema-specifications>=2023.03.6 in c:\users\r
ahul\appdata\local\programs\python\python312\lib\site-packages (from jsonschema>=
3.0->altair<6,>=4.0->streamlit) (2023.12.1)
Requirement already satisfied: referencing>=0.28.4 in c:\users\rahul\appdata\loca
l\programs\python\python312\lib\site-packages (from jsonschema>=3.0->altair<6,>=
4.0->streamlit) (0.32.0)
Requirement already satisfied: rpds-py>=0.7.1 in c:\users\rahul\appdata\local\pro
grams\python\python312\lib\site-packages (from jsonschema>=3.0->altair<6,>=4.0->s
treamlit) (0.16.2)
Requirement already satisfied: mdurl~=0.1 in c:\users\rahul\appdata\local\program
s\python\python312\lib\site-packages (from markdown-it-py>=2.2.0->rich<14,>=10.1
4.0->streamlit) (0.1.2)
Requirement already satisfied: six>=1.5 in c:\users\rahul\appdata\local\programs
\python\python312\lib\site-packages (from python-dateutil>=2.8.2->pandas<3,>=1.3.
0->streamlit) (1.16.0)
```

In [6]:
```
import pickle
```

In [7]:
```
import numpy as np
```

In [8]:
```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
```

In [9]:
```
df=pd.read_csv("Cardetails.csv")
```

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

Out[10]:

	name	year	selling_price	km_driven	fuel	seller_type	transmission	owner	n
0	Maruti Swift Dzire VDI	2014	450000	145500	Diesel	Individual	Manual	First Owner	
1	Skoda Rapid 1.5 TDI Ambition	2014	370000	120000	Diesel	Individual	Manual	Second Owner	
2	Honda City 2017-2020 EXi	2006	158000	140000	Petrol	Individual	Manual	Third Owner	
3	Hyundai i20 Sportz Diesel	2010	225000	127000	Diesel	Individual	Manual	First Owner	
4	Maruti Swift VXI BSIII	2007	130000	120000	Petrol	Individual	Manual	First Owner	

In [11]:

```
df.drop(columns=["torque"],inplace=True)
```

In [12]:

```
df.head()
```

Out[12]:

	name	year	selling_price	km_driven	fuel	seller_type	transmission	owner	n
0	Maruti Swift Dzire VDI	2014	450000	145500	Diesel	Individual	Manual	First Owner	
1	Skoda Rapid 1.5 TDI Ambition	2014	370000	120000	Diesel	Individual	Manual	Second Owner	
2	Honda City 2017-2020 EXi	2006	158000	140000	Petrol	Individual	Manual	Third Owner	
3	Hyundai i20 Sportz Diesel	2010	225000	127000	Diesel	Individual	Manual	First Owner	
4	Maruti Swift VXI BSIII	2007	130000	120000	Petrol	Individual	Manual	First Owner	

In [13]:
```
df.shape
```

Out[13]:
```
(8128, 12)
```

Preprocessing

Null check

In [14]:
```
df.isnull().sum()
```

Out[14]:
```
name                0
year                0
selling_price       0
km_driven           0
fuel                0
seller_type         0
transmission        0
owner               0
mileage           221
engine            221
max_power         215
seats             221
dtype: int64
```

In [15]:
```
df.dropna(inplace=True)
```

In [16]:
```
df.shape
```

Out[16]:
```
(7907, 12)
```

Duplicate Check

In [17]:
```
df.duplicated().sum()
```

Out[17]:
```
1189
```

In [18]:
```
df.drop_duplicates(inplace=True)
```

In [19]:
```
df.shape
```

Out[19]:
```
(6718, 12)
```

In [20]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 6718 entries, 0 to 8125
Data columns (total 12 columns):
 #   Column         Non-Null Count  Dtype  
---  ------         --------------  -----  
 0   name           6718 non-null   object 
 1   year           6718 non-null   int64  
 2   selling_price  6718 non-null   int64  
 3   km_driven      6718 non-null   int64  
 4   fuel           6718 non-null   object 
 5   seller_type    6718 non-null   object 
 6   transmission   6718 non-null   object 
 7   owner          6718 non-null   object 
 8   mileage        6718 non-null   object 
 9   engine         6718 non-null   object 
 10  max_power      6718 non-null   object 
 11  seats          6718 non-null   float64
dtypes: float64(1), int64(3), object(8)
memory usage: 682.3+ KB
```

Data Analysis

In [21]:

```
for col in df.columns:
    print('unique values of' + col)
    print(df[col].unique())
    print('===================')
```

```
unique values ofname
['Maruti Swift Dzire VDI' 'Skoda Rapid 1.5 TDI Ambition'
 'Honda City 2017-2020 EXi' ... 'Tata Nexon 1.5 Revotorq XT'
 'Ford Freestyle Titanium Plus Diesel BSIV'
 'Toyota Innova 2.5 GX (Diesel) 8 Seater BS IV']
===================
unique values ofyear
[2014 2006 2010 2007 2017 2001 2011 2013 2005 2009 2016 2012 2002 2015
 2018 2019 2008 2020 1999 2000 2003 2004 1994 1998 1997 1995 1996]
===================
unique values ofselling_price
[  450000   370000   158000   225000   130000   440000    96000    45000
   350000   200000   500000    92000   280000   180000   400000   778000
   150000   680000   174000   950000   525000   600000   575000   275000
   300000   220000   254999   670000   730000   650000   330000   366000
  1149000   425000  2100000   925000   675000   819999   390000  1500000
   700000  1450000  1090000   850000  1650000  1750000  1590000  1689999
  1425000   265000   190000   630000   540000   448000   745000  1025000
   235000  1700000  1200000   610000  2500000   484999   315000   290000
   455000   351000   535000   175000   565000   120000   725000   185000
   615000   270000   625000   866000   375000   522000   451999   475000
   780000   595000  1140000   360000   105000   135000   690000  3975000
  5150000  3200000  4100000  4500000  6000000  3790000  5800000  1864999
  2700000   795000  3400000  2650000  5850000   975000   805000  2625000
   811000   550000   645000  2550000   599000   875000   894999   340000
  1525000  2150000  1900000  2900000 10000000   160000   515000    40000
   480000   487000   386000   100000   490000   345000    85000   215000
   320000   459999    65000   720000   800000   520000   580000   420000
  1250000   830000   335000   329000   775000   524000   245000   910000
   900000   250000  1100000   640000   409999   165000    80000  1000000
   110000   170000  1380000  1300000   530000   310000  1150000   380000
   404000   114999   465000   210000  4600000   509999    90000   409000
   395000    81000   555000   229999  1210000   325000    60000    89000
   434999  3000000   570000  1550000    64000   140000   503000  3800000
  2000000   740000   750000  5400000  5500000   779000  2475000   240000
  1085000    95000  1290000   219000   260000   840000   292000   385000
  1050000   735000   560000   585000   760000   430000   643000   285000
   660000   268000    88000   561000   755000   125000  1600000   801000
   317000   495000   236000  1800000   415000   217000   466000   605000
  1350000   337000   655000   196000   299000   634000  1225000   238000
   590000   195000   715000   168000   531000  1625000   145000   441000
    67500   321000   298000  1075000   681000   810000   573000   101000
  1019999   244000  1165000   241000   327000   620000   204999   576000
   695000   231000   261000   371000   421000   710000  1040000   199000
   305000  1745000   611000    75000    55000  1325000   885000   108000
   272000  1125000   388000   757000   626000   793000  3750000  1400000
  1925000   844999  2940000  1825000   890000   770000   467000   711000
   511000   166000   392000   833000    35000   880000  1030000   521000
   451000   472000   649000   429000  1576000   825000   470000  1560000
  1950000   786000   731000   191000   651000   445000    93150   403000
   144000    52000   302000   869999   155000   246000   405000    70000
  3250000  1143000   635000  1575000   318000   152000   248000  1465000
  1080000   678000  1460000   756000   665000  2950000   377000  1675000
   169000   685000   749000   899000   259000   416000   773000   889000
   118000   256000  1888000    99000  2280000    31504   295000   444000
   860000  2450000   198000  2825000  2125000  3350000   365000   919999
  2600000   839000  1594000   161000   423000   551000    50000   411000
  1530000   376000  1850000    86000   822000    30000   252000   131000
   128000   790000   604000  1175000   141000   151000   577000    83000
   946000   356000   955000   978999   348000   399000   362000   499000
   358000   278000   426000   211000  1270000   426999   960000  1031000
    72000   479000   764000  2199000  3600000   184000   119000   267000
   328000   112999    39000  2850000  2800000  3900000   221000   631000
   308000   940000   361000   355000  7200000   930000  1670000   980000
   486000   242000   654000   835000   408000    94500  1110000   438999
   911999   693000   449000  1390000   698000   539000   765000   562000
   571000   589000   227999  1151000  1295000  4090000   347000   493000
   545000  1475000   287000  1630000   501000   181000  2711000  1035000
   316000   813000  1190000   378000   156000   194000   754000    55599
  1511000    78000   250999   312000    66000   369000   149000   752000
   142000   111000   206000   567000   699000   632000   905000  1145000
    58000   179000  1515000   178000   222000   387000  1405000   282000
   458000   157000   497000  1227000   574000   121000  3100000    68000
  1920000   808000   512000   671000   799000  1365000  1490000    80500
   189000   892000    93000  1680000   667000   204000    42000   694000
   861999   476999  5200000  2375000  2750000  1385000   785000  1275000
  1748999   132000    33351  2064000   469000   906000   797000  6223000
  5923000  6523000   122000   891000   346000   532000   974000   483000
   138000   432000  1757000   412000   126000   736000   579000   187000
  4000000  1898999   482000   401000   737000   291000   999000   957000
   834000   297000   669000   616000   129000   115999   861000  3251000
   911000   148000   433000   488000    67000   516000   136000    29999
   124000  1938000   172000    59000  3500000   537000   849000   851000
   597000   741000   556000   549000   679000   969999  1237000    31000
   666000    54000   374000   455999   453000   258000   428000   262000
   389000   339000   277000   578000   684000   949000   227000  5830000
   721000   751000  2400000   990000  1282000  1147000   558000   722000
   563000  2300000  2200000   488999   276000   434000   372000  1520000
   202999   541000   254000   528000   102000   311000   977000  1251000
  1445000    98000    59259   153000  1051000   591000   319000   454000
  1160000  3300000   791000   802999  1789999  1044999   279000   944999
  1859000  2175000   430999   803999   934000   357000   965000   341000
    75527  1132000   352000   252999   313000   391000  2051000   212000
  2575000  1516000   359000   209000  1220000  1180000    45957  1265000
   382000   526000    90177    46000   307000   746000]
==================
unique values ofkm_driven
[ 145500  120000  140000  127000   45000  175000    5000   90000  169000
   68000  100000   40000   70000   53000   80000   50000   72000   35000
   28000   25000    2388   16200   10000   15000   42000   60000   76000
   28900   86300   23300   32600   10300   77000   99000   27800   49800
  151000   54700   64000   63000  127700   33900   59000  110000  147000
   30000  135000    9850   78000  170000   49000   32000   38000   44000
   12000   55500   61500  150000   37800  114000   48000   69000   13000
   76139   65000    1303   31800   20000   17000   21000   37000   29500
    7500   19000   41000   39000   22000   47000   72200   49900    9000
   11000   95000   18000   46100   16000    9654   24300   42163    8000
   71000    9500   36600   14000  120600   86000    7800   31377   75000
   93000  125000   13500  162500   92500  158000    2000  181000  193000
  122358   95200   33033   63063  207890    7976   16500   99361   33000
   80100  160000   26000   29000   58000   36000  185000  162000   67000
   52000   68089   58343   38817   56494   79328    5621   25538   69779
   56290    8500    7032  106000  240000  214000    1000  265000  134000
    2136  250000  130000   99500  119000   44665  123000   34000  146000
   85000  201850   46000  190000   43000   14200   19100   54000  101500
   55735    1500  181491   53319   40906   31711   43755   66693   59549
   94385   73730  221889  116104  150546  148120  142000  126000   24000
  101000  168000  165000   98500   98000   77800   51492  136000   91000
  167000  180000    6000   91500  132478   51000  105000  137500   88000
  156060  107000   57000  300000  200000   18945  144000  107825    1620
```

```
141000  108000   55000   37944   90400   96000  193623   63186  219000
 23000   94000  140500  143000  195000  360003    8079  114368   79990
 81632  155201  101504   90165   86017   85036  125531  206000   49700
 74000    6550   56000   41779   31000   13534    7000   28156   63309
155000   11500    3000  157000   27000   26300   89100  104000  210000
216000   84000  161000   72113  113000   96443   51500  104500   49102
 81000  227000  117000    8576   19723   21446   52289   52236   54470
 25471   33928   59736   54290    2118   61379   58544   74381   63982
 56429   19500   56832   30528   65285   20102   52380   74250   57247
178000   42323   42462   54723   42545   20375   43978   40200   27233
 28832  103000  142500  115000   35500   17500   92000    4500   83000
  6750   18500   50478    9200    2300    4295    5229    4701    6757
 28182   11533   11688   57728   23712   70670   43381   79011   58500
 43500   12700   49455   46815  334000  100875  118000  128000  144030
176000   89000   97000   12800   65100   15200   82000  116500  145000
 73000  113226  160500  132000  217000  139000    3500   62000 1500000
120005   16830   93468   16589   53201   25272   53619    6001  202000
 71717   84321  177000  205000   41800   88700  188000  225000    5500
  4875    2700   58511   26235   26766    1600   24522   13663   58945
375000  156975   27620  143200  138000  189000  156000   10800   67500
255000   71850  112000  159000   76131   51777   64788   46951   74537
 23500   68600   13887   66000   24400   27500   67840  380000  112880
 61915  184000    6200   16700    7967    9656   68609   33360   14317
 87237   62960   26634  163720   67998   20706  100581   67303   79139
 84784  130376   33500   22053   78432  270000    7200  117500   78500
115717    4000   26432   46702   46706   53523   56365   49563   56432
 52365   53512   46533   58632   65300   39395  230000   32500   64800
248000   66444  218463  153000   87185  137000    3564   48756   80322
173000  148000    6500  220000  108957   22200  260000   42108   79000
 17100   43001  100600   92385  116000   72500   82500   50600  121000
101903   85568    3100  186000   59300   89322   48300   23511   22512
 62900   52442   34156   34152   62841   38600   27654   48698   32995
 90150   85700   25339   41232   68240   92651   88754   23700   69500
 28050   22947    7672   36659   44500    9900  124316    6544   49600
198000  103994   35008   31100   42312   54891   18890   75010    4337
 14495   83585   48406   56239   33243   15858   87847   87000  112072
 52269   15780 2360457  108800   12500  577414  121941  152186   82246
 89580    2789   35278    2860   78562   36088  129000  500000  133000
 64500   24500   45500   71500   93500   60300   44391   56315   15151
114321   37333  330000  291000   21500  197000   40300   30030   29029
 72072   82082   80600   58609   33003  165500    5800   68700  102996
 96500   52412   85472   87452   66530   32331  172000   23456  136500
111000   42500   47200  246000    5200   10500   93331  109000    5400
102000   29700    7600   29340   80800   24265   56900   88200   40800
  8588  305000  212000  101200  179150   84487  264000   51146  298000
149000  291977   36800   52200  248200    7720   57882  242000   55380
 18816   33019   47747   45900   68697  147279   24700   43526   55885
 16034  156040   93415  122000   68519   22966    2350   59872   91182
 59500   75500  152500   69123   77524   40523   23600  290000   24177
121779   15381   74800  109322  475000   61000   80500   65755    2600
 15500   50700  440000  152000  200400    7300    9750  157138  285000
127991   13120   87500   62200   87540  231438  239451  154000  426000
182000  215000   77300   61260   17601    4773   57900    8600   78010
 83844   23999   91400  178500   66953  166000  131000  100750   48676
 79500  200185   49025   76460   47370   45217  222300   44600   28080
 28800   21900   28100    7673   96272   22500    7400  370000  150360
 42130  147500   56194   54188   54043   66657   35582   54327    7949
 59734   49185   56389   58245   36422   42535   40736   56246   55403
105358   49200   81500    3177   39414   30078   44588   49907   67082
112879   10200    1300  175802  192000   73840  320000   24857   26442
```

```
  50800   40142    9400   12584    4300   14548   44772   48500   28782
  28451   36500   44885  131111   58400   23400    3010  163000   77088
 136511   36710   28180   44077   55768  123278   26500   18484    2560
 103655   56975  187000  218000   73257   26320   38426  116700  176062
  77500   39500   55896   30400  164000  271000   75262   91863   47552
  14700   19700  145241   95500   53473   75958   31596   85710  129627
  55130   22522   51856   29434   34500    6825  112011   53534  248119
  24019   19600  376412  183000  108916   59865   68140   53190   70100
  47725   70195   77395   74321   80235   77150   61100   82300   82050
  29899   59235  201000   77215   48228   58559   50856   18300   59292
  21147   61173   91567   33400   92686   67600   49500  112048   30646
  43011  104300   73500  280000   35700   22700   21871   55425   37659
  37500   19800   45629   68850   45775   68203   97343   82507   59400
 125876   56800   65204    2286   15732   49523   36521   64481   49060
  37161   60175   49957   43235   50699       1   50074   30154   26263
  84925   46357   46737  110048  186388   20171   37151   58161   50171
  55161    9599   16151   28161   80868  194000  191000]
===================
unique values offuel
['Diesel' 'Petrol' 'LPG' 'CNG']
===================
unique values ofseller_type
['Individual' 'Dealer' 'Trustmark Dealer']
===================
unique values oftransmission
['Manual' 'Automatic']
===================
unique values ofowner
['First Owner' 'Second Owner' 'Third Owner' 'Fourth & Above Owner'
 'Test Drive Car']
===================
unique values ofmileage
['23.4 kmpl' '21.14 kmpl' '17.7 kmpl' '23.0 kmpl' '16.1 kmpl' '20.14 kmpl'
 '17.3 km/kg' '23.59 kmpl' '20.0 kmpl' '19.01 kmpl' '17.3 kmpl'
 '19.3 kmpl' '18.9 kmpl' '18.15 kmpl' '24.52 kmpl' '19.7 kmpl'
 '22.54 kmpl' '21.0 kmpl' '25.5 kmpl' '26.59 kmpl' '21.5 kmpl' '20.3 kmpl'
 '21.4 kmpl' '24.7 kmpl' '18.2 kmpl' '16.8 kmpl' '24.3 kmpl' '14.0 kmpl'
 '18.6 kmpl' '33.44 km/kg' '23.95 kmpl' '17.0 kmpl' '20.63 kmpl'
 '13.93 kmpl' '16.0 kmpl' '17.8 kmpl' '18.5 kmpl' '12.55 kmpl'
 '12.99 kmpl' '14.8 kmpl' '13.5 kmpl' '26.0 kmpl' '20.65 kmpl' '27.3 kmpl'
 '11.36 kmpl' '17.68 kmpl' '14.28 kmpl' '18.53 kmpl' '14.84 kmpl'
 '21.12 kmpl' '20.36 kmpl' '21.27 kmpl' '18.16 kmpl' '22.0 kmpl'
 '25.1 kmpl' '20.51 kmpl' '21.66 kmpl' '25.2 kmpl' '22.9 kmpl'
 '16.02 kmpl' '20.54 kmpl' '22.77 kmpl' '15.71 kmpl' '23.1 kmpl'
 '19.02 kmpl' '19.81 kmpl' '26.2 km/kg' '16.47 kmpl' '15.04 kmpl'
 '19.1 kmpl' '21.79 kmpl' '18.8 kmpl' '21.21 kmpl' '15.37 kmpl'
 '11.79 kmpl' '19.0 kmpl' '14.3 kmpl' '15.8 kmpl' '15.1 kmpl' '19.09 kmpl'
 '22.32 kmpl' '21.9 kmpl' '14.53 kmpl' '21.63 kmpl' '20.85 kmpl'
 '20.45 kmpl' '19.67 kmpl' '23.01 kmpl' '20.77 kmpl' '17.92 kmpl'
 '17.01 kmpl' '22.37 kmpl' '19.33 kmpl' '9.5 kmpl' '12.83 kmpl'
 '22.48 kmpl' '16.78 kmpl' '14.67 kmpl' '15.0 kmpl' '13.96 kmpl'
 '18.0 kmpl' '12.07 kmpl' '26.21 kmpl' '10.8 kmpl' '16.3 kmpl' '13.6 kmpl'
 '14.74 kmpl' '15.6 kmpl' '19.56 kmpl' '22.69 kmpl' '19.16 kmpl'
 '18.12 kmpl' '12.1 kmpl' '17.5 kmpl' '42.0 kmpl' '20.4 kmpl' '21.1 kmpl'
 '19.44 kmpl' '13.0 kmpl' '21.43 kmpl' '22.95 kmpl' '16.2 kmpl'
 '15.3 kmpl' '28.09 kmpl' '17.4 kmpl' '19.4 kmpl' '26.6 km/kg' '17.6 kmpl'
 '28.4 kmpl' '14.1 kmpl' '25.17 kmpl' '22.74 kmpl' '17.57 kmpl'
 '16.95 kmpl' '19.49 kmpl' '17.21 kmpl' '13.2 kmpl' '14.2 kmpl'
 '26.8 kmpl' '25.4 kmpl' '11.5 kmpl' '27.28 kmpl' '17.97 kmpl' '12.8 kmpl'
 '16.55 kmpl' '12.05 kmpl' '14.07 kmpl' '21.02 kmpl' '11.57 kmpl'
 '17.9 kmpl' '15.96 kmpl' '17.1 kmpl' '17.19 kmpl' '21.01 kmpl'
 '24.0 kmpl' '25.6 kmpl' '21.38 kmpl' '23.84 kmpl' '23.08 kmpl'
 '14.24 kmpl' '20.71 kmpl' '15.64 kmpl' '14.5 kmpl' '16.34 kmpl'
 '27.39 kmpl' '11.1 kmpl' '13.9 kmpl' '20.88 km/kg' '20.92 kmpl'
 '23.8 kmpl' '24.4 kmpl' '15.29 kmpl' '21.19 kmpl' '22.5 kmpl' '19.6 kmpl'
 '23.65 kmpl' '25.32 kmpl' '23.5 kmpl' '16.6 kmpl' '23.9 kmpl' '20.8 kmpl'
 '27.62 kmpl' '12.9 kmpl' '25.44 kmpl' '17.88 kmpl' '22.7 kmpl'
 '17.2 kmpl' '15.42 kmpl' '19.68 kmpl' '18.7 kmpl' '15.4 kmpl'
 '19.34 kmpl' '22.71 kmpl' '25.8 kmpl' '13.7 kmpl' '12.2 kmpl'
 '18.49 kmpl' '9.0 kmpl' '0.0 kmpl' '13.58 kmpl' '10.1 kmpl' '20.5 kmpl'
 '25.0 kmpl' '10.5 kmpl' '22.07 kmpl' '22.3 kmpl' '15.26 kmpl'
 '20.62 kmpl' '27.4 kmpl' '23.2 kmpl' '14.4 kmpl' '18.4 kmpl'
 '30.46 km/kg' '14.02 kmpl' '11.0 kmpl' '20.6 kmpl' '22.05 kmpl'
 '20.2 kmpl' '18.1 kmpl' '22.1 kmpl' '19.87 kmpl' '13.01 kmpl'
 '18.06 kmpl' '26.1 kmpl' '16.52 kmpl' '13.55 kmpl' '24.2 kmpl'
 '25.83 kmpl' '11.2 kmpl' '17.09 kmpl' '21.03 kmpl' '17.45 kmpl'
 '21.64 kmpl' '21.94 km/kg' '13.87 kmpl' '19.98 kmpl' '20.52 kmpl'
 '23.57 kmpl' '11.7 kmpl' '17.43 kmpl' '18.88 kmpl' '13.68 kmpl'
 '11.18 kmpl' '20.89 kmpl' '11.8 kmpl' '19.62 kmpl' '21.7 kmpl'
 '14.9 kmpl' '19.5 kmpl' '10.91 kmpl' '15.7 kmpl' '20.73 kmpl'
 '15.85 kmpl' '20.7 kmpl' '14.23 kmpl' '16.5 kmpl' '17.36 kmpl'
 '12.6 kmpl' '16.36 kmpl' '14.95 kmpl' '16.9 kmpl' '19.2 kmpl'
 '16.96 kmpl' '22.15 kmpl' '18.78 kmpl' '19.61 kmpl' '17.71 kmpl'
 '18.3 kmpl' '19.12 kmpl' '19.72 kmpl' '12.0 kmpl' '11.4 kmpl'
 '23.03 kmpl' '11.07 kmpl' '15.9 kmpl' '17.67 kmpl' '20.46 kmpl'
 '13.1 kmpl' '13.45 km/kg' '24.8 kmpl' '15.73 kmpl' '15.11 kmpl'
 '12.7 kmpl' '21.2 kmpl' '20.38 kmpl' '21.56 kmpl' '13.22 kmpl'
 '14.49 kmpl' '15.05 kmpl' '23.26 kmpl' '15.41 kmpl' '13.8 kmpl'
 '22.27 kmpl' '32.52 km/kg' '14.66 kmpl' '12.12 kmpl' '16.84 kmpl'
 '14.09 kmpl' '14.7 kmpl' '13.4 kmpl' '15.5 kmpl' '13.49 kmpl'
 '11.88 km/kg' '14.6 kmpl' '10.75 kmpl' '24.5 kmpl' '11.74 kmpl'
 '16.07 kmpl' '15.63 kmpl' '26.3 km/kg' '23.7 km/kg' '25.47 kmpl'
 '17.05 kmpl' '23.3 kmpl' '11.9 kmpl' '13.38 kmpl' '20.86 kmpl'
 '19.2 km/kg' '10.9 kmpl' '18.25 kmpl' '15.2 kmpl' '20.37 kmpl'
 '17.8 km/kg' '21.8 kmpl' '11.96 kmpl' '24.04 kmpl' '19.69 kmpl'
 '13.73 kmpl' '21.04 kmpl' '25.01 kmpl' '10.93 kmpl' '10.9 km/kg'
 '24.29 kmpl' '13.44 kmpl' '20.07 kmpl' '21.1 km/kg' '19.08 kmpl'
 '20.34 kmpl' '11.68 kmpl' '12.5 kmpl' '12.3 kmpl' '23.87 kmpl'
 '16.38 kmpl' '17.42 kmpl' '10.0 kmpl' '18.24 kmpl' '10.71 kmpl'
 '19.59 kmpl' '16.7 kmpl' '19.83 kmpl' '21.76 kmpl' '16.05 kmpl'
 '20.28 kmpl' '16.25 kmpl' '16.73 kmpl' '18.48 kmpl' '13.2 km/kg'
 '21.4 km/kg' '14.99 kmpl' '18.76 kmpl' '16.4 kmpl' '19.64 kmpl'
 '14.94 kmpl' '16.6 km/kg' '16.0 km/kg' '17.11 kmpl' '22.8 km/kg'
 '32.26 km/kg' '33.0 km/kg' '12.4 kmpl' '18.44 kmpl' '16.09 kmpl'
 '19.0 km/kg' '12.62 kmpl' '21.13 kmpl' '15.17 kmpl' '21.73 kmpl'
 '21.72 kmpl' '12.85 kmpl' '14.81 kmpl' '13.24 kmpl' '14.4 km/kg'
 '21.49 kmpl' '14.62 kmpl' '26.83 km/kg' '11.45 kmpl' '12.08 kmpl'
 '15.74 kmpl' '11.3 kmpl' '15.1 km/kg' '14.21 kmpl' '11.72 kmpl'
 '16.51 kmpl']
===================
unique values ofengine
['1248 CC' '1498 CC' '1497 CC' '1396 CC' '1298 CC' '1197 CC' '1061 CC'
 '796 CC' '1364 CC' '1399 CC' '1461 CC' '993 CC' '1198 CC' '1199 CC'
 '998 CC' '1591 CC' '2179 CC' '1368 CC' '2982 CC' '2494 CC' '2143 CC'
 '2477 CC' '1462 CC' '2755 CC' '1968 CC' '1798 CC' '1196 CC' '1373 CC'
 '1598 CC' '1998 CC' '1086 CC' '1194 CC' '1172 CC' '1405 CC' '1582 CC'
 '999 CC' '2487 CC' '1999 CC' '3604 CC' '2987 CC' '1995 CC' '1451 CC'
 '1969 CC' '2967 CC' '2497 CC' '1797 CC' '1991 CC' '2362 CC' '1493 CC'
 '1599 CC' '1341 CC' '1794 CC' '799 CC' '1193 CC' '2696 CC' '1495 CC'
 '1186 CC' '1047 CC' '2498 CC' '2956 CC' '2523 CC' '1120 CC' '624 CC'
 '1496 CC' '1984 CC' '2354 CC' '814 CC' '793 CC' '1799 CC' '936 CC'
 '1956 CC' '1997 CC' '1499 CC' '1948 CC' '2997 CC' '2489 CC' '2499 CC'
 '2609 CC' '2953 CC' '1150 CC' '1994 CC' '1388 CC' '1527 CC' '2199 CC'
 '995 CC' '2993 CC' '1586 CC' '1390 CC' '909 CC' '2393 CC' '3198 CC'
 '1339 CC' '2835 CC' '2092 CC' '1595 CC' '2496 CC' '1596 CC' '1597 CC'
 '2596 CC' '2148 CC' '1299 CC' '1590 CC' '2231 CC' '2694 CC' '2200 CC'
 '1795 CC' '1896 CC' '1796 CC' '1422 CC' '1489 CC' '2359 CC' '2197 CC'
 '2999 CC' '1781 CC' '2650 CC' '1343 CC' '2446 CC' '3498 CC' '2198 CC'
 '2776 CC' '1950 CC']
===================
unique values ofmax_power
['74 bhp' '103.52 bhp' '78 bhp' '90 bhp' '88.2 bhp' '81.86 bhp' '57.5 bhp'
 '37 bhp' '67.1 bhp' '68.1 bhp' '108.45 bhp' '60 bhp' '73.9 bhp' '67 bhp'
 '82 bhp' '88.5 bhp' '46.3 bhp' '88.73 bhp' '64.1 bhp' '98.6 bhp'
 '88.8 bhp' '83.81 bhp' '83.1 bhp' '47.3 bhp' '73.8 bhp' '34.2 bhp'
 '35 bhp' '81.83 bhp' '40.3 bhp' '121.3 bhp' '138.03 bhp' '160.77 bhp'
 '117.3 bhp' '116.3 bhp' '83.14 bhp' '67.05 bhp' '168.5 bhp' '100 bhp'
 '120.7 bhp' '98.63 bhp' '175.56 bhp' '103.25 bhp' '171.5 bhp' '100.6 bhp'
 '174.33 bhp' '187.74 bhp' '170 bhp' '78.9 bhp' '88.76 bhp' '86.8 bhp'
 '108.495 bhp' '108.62 bhp' '93.7 bhp' '103.6 bhp' '98.59 bhp' '189 bhp'
 '67.04 bhp' '68.05 bhp' '58.2 bhp' '82.85 bhp' '81.80 bhp' '73 bhp'
 '120 bhp' '94.68 bhp' '160 bhp' '65 bhp' '155 bhp' '69.01 bhp'
 '126.32 bhp' '138.1 bhp' '83.8 bhp' '126.2 bhp' '98.96 bhp' '62.1 bhp'
 '86.7 bhp' '188 bhp' '214.56 bhp' '177 bhp' '280 bhp' '148.31 bhp'
 '254.79 bhp' '190 bhp' '177.46 bhp' '204 bhp' '141 bhp' '117.6 bhp'
 '241.4 bhp' '282 bhp' '150 bhp' '147.5 bhp' '108.5 bhp' '103.5 bhp'
 '183 bhp' '181.04 bhp' '157.7 bhp' '164.7 bhp' '91.1 bhp' '400 bhp'
 '68 bhp' '75 bhp' '85.8 bhp' '87.2 bhp' '53 bhp' '118 bhp' '103.2 bhp'
 '83 bhp' '84 bhp' '58.16 bhp' '147.94 bhp' '74.02 bhp' '53.3 bhp'
 '80 bhp' '88.7 bhp' '97.7 bhp' '121.36 bhp' '162 bhp' '140 bhp' '94 bhp'
 '100.57 bhp' '82.9 bhp' '83.11 bhp' '70 bhp' '153.86 bhp' '121 bhp'
 '126.3 bhp' '73.97 bhp' '171 bhp' '69 bhp' '99.6 bhp' '102 bhp' '105 bhp'
 '63 bhp' '79.4 bhp' '97.9 bhp' '63.1 bhp' '66.1 bhp' '110 bhp'
 '174.5 bhp' '53.26 bhp' '73.75 bhp' '67.06 bhp' '64.08 bhp' '37.5 bhp'
 '189.3 bhp' '158.8 bhp' '61.7 bhp' '55.2 bhp' '71.01 bhp' '73.74 bhp'
 '147.9 bhp' '71 bhp' '77 bhp' '121.4 bhp' '113.4 bhp' '47 bhp' '130 bhp'
 '57.6 bhp' '138 bhp' '52.8 bhp' '53.64 bhp' '53.5 bhp' '76.8 bhp'
 '82.4 bhp' '113.42 bhp' '76 bhp' '84.8 bhp' '56.3 bhp' '218 bhp'
 '112 bhp' '92 bhp' '105.5 bhp' '169 bhp' '95 bhp' '72.4 bhp' '115 bhp'
 '152 bhp' '91.2 bhp' '156 bhp' '74.9 bhp' '62 bhp' '105.3 bhp'
 '73.94 bhp' '85.80 bhp' '85 bhp' '118.3 bhp' '72 bhp' '147.51 bhp'
 '58 bhp' '64 bhp' '126.24 bhp' '76.9 bhp' '194.3 bhp' '99.23 bhp'
 '89.84 bhp' '123.7 bhp' '118.35 bhp' '99 bhp' '241 bhp' '136 bhp'
 '261.4 bhp' '104.68 bhp' '37.48 bhp' '104 bhp' '88.50 bhp' '63.12 bhp'
 '91.7 bhp' '102.5 bhp' '177.6 bhp' '45 bhp' '123.37 bhp' '147.8 bhp'
 '184 bhp' '84.48 bhp' '68.07 bhp' '74.96 bhp' '167.6 bhp' '152.87 bhp'
 '112.2 bhp' '83.83 bhp' '197 bhp' '110.4 bhp' '104.55 bhp' '103 bhp'
 '103.3 bhp' '66 bhp' '108.6 bhp' '165 bhp' '163.7 bhp' '116.9 bhp'
 '94.93 bhp' '127 bhp' '198.5 bhp' '179.5 bhp' '120.69 bhp' '121.31 bhp'
 '138.08 bhp' '187.7 bhp' '80.8 bhp' '86.79 bhp' '93.87 bhp' '116.6 bhp'
 '143 bhp' '92.7 bhp' '88 bhp' '58.33 bhp' '78.8 bhp' '64.4 bhp' '125 bhp'
 '139.01 bhp' '254.8 bhp' '181 bhp' '258 bhp' '55.23 bhp' '270.9 bhp'
 '265 bhp' '157.75 bhp' '101 bhp' '186 bhp' '187.4 bhp' '224 bhp'
 '64.9 bhp' '148 bhp' '35.5 bhp' '89.75 bhp' '32.8 bhp' '91.72 bhp'
 '106 bhp' '98.97 bhp' '66.6 bhp' '86 bhp' '65.3 bhp' '98.82 bhp'
 '198.25 bhp' '38 bhp' '142 bhp' '132 bhp' '174.57 bhp' '178 bhp'
 '163.2 bhp' '203.2 bhp' '177.5 bhp' '175 bhp' '57 bhp' '80.84 bhp'
 '68.4 bhp' '167.67 bhp' '170.63 bhp' '52 bhp' '149.5 bhp' '48.21 bhp'
 ' bhp' '201.1 bhp' '100.5 bhp' '144 bhp' '194.4 bhp' '168.7 bhp'
 '104.5 bhp' '103.26 bhp' '116.4 bhp' '98.79 bhp' '80.9 bhp' '58.3 bhp'
 '272 bhp' '235 bhp' '167.62 bhp' '170.30 bhp' '139.46 bhp' '158 bhp'
 '110.5 bhp' '82.5 bhp' '141.1 bhp' '38.4 bhp' '197.2 bhp' '161 bhp'
 '194 bhp' '122.4 bhp' '134.10 bhp' '60.2 bhp' '134 bhp' '203 bhp'
 '135.1 bhp']
===================
unique values ofseats
[ 5.  4.  7.  8.  6.  9. 10. 14.  2.]
===================
```

In [22]:
```python
def get_brand_name(car_name):
    car_name= car_name.split(" ")[0]
    return car_name.strip()
```

In [23]:
```python
def clean_data(value):
    value= value.split(" ")[0]
    value= value.strip()
    if value == '':
        value=0
    return float(value)
```

In [24]:
```python
df['name']=df['name'].apply(get_brand_name)
```

In [25]:
```python
df['name'].unique()
```

Out[25]:
```
array(['Maruti', 'Skoda', 'Honda', 'Hyundai', 'Toyota', 'Ford', 'Renault',
       'Mahindra', 'Tata', 'Chevrolet', 'Datsun', 'Jeep', 'Mercedes-Benz',
       'Mitsubishi', 'Audi', 'Volkswagen', 'BMW', 'Nissan', 'Lexus',
       'Jaguar', 'Land', 'MG', 'Volvo', 'Daewoo', 'Kia', 'Fiat', 'Force',
       'Ambassador', 'Ashok', 'Isuzu', 'Opel'], dtype=object)
```

In [26]:
```python
df['mileage']=df['mileage'].apply(get_brand_name)
```

In [27]:
```python
df['max_power']=df['max_power'].apply(clean_data)
```

In [28]:
```python
df['engine']=df['engine'].apply(clean_data)
```

In [29]:
```python
for col in df.columns:
    print("uniwue values of" + col)
    print(df[col].unique())
    print("===============")
```

```
uniwue values ofname
['Maruti' 'Skoda' 'Honda' 'Hyundai' 'Toyota' 'Ford' 'Renault' 'Mahindra'
 'Tata' 'Chevrolet' 'Datsun' 'Jeep' 'Mercedes-Benz' 'Mitsubishi' 'Audi'
 'Volkswagen' 'BMW' 'Nissan' 'Lexus' 'Jaguar' 'Land' 'MG' 'Volvo' 'Daewoo'
 'Kia' 'Fiat' 'Force' 'Ambassador' 'Ashok' 'Isuzu' 'Opel']
===============
uniwue values ofyear
[2014 2006 2010 2007 2017 2001 2011 2013 2005 2009 2016 2012 2002 2015
 2018 2019 2008 2020 1999 2000 2003 2004 1994 1998 1997 1995 1996]
===============
uniwue values ofselling_price
[  450000   370000   158000   225000   130000   440000    96000    45000
   350000   200000   500000    92000   280000   180000   400000   778000
   150000   680000   174000   950000   525000   600000   575000   275000
   300000   220000   254999   670000   730000   650000   330000   366000
  1149000   425000  2100000   925000   675000   819999   390000  1500000
   700000  1450000  1090000   850000  1650000  1750000  1590000  1689999
  1425000   265000   190000   630000   540000   448000   745000  1025000
   235000  1700000  1200000   610000  2500000   484999   315000   290000
   455000   351000   535000   175000   565000   120000   725000   185000
   615000   270000   625000   866000   375000   522000   451999   475000
   780000   595000  1140000   360000   105000   135000   690000  3975000
  5150000  3200000  4100000  4500000  6000000  3790000  5800000  1864999
  2700000   795000  3400000  2650000  5850000   975000   805000  2625000
   811000   550000   645000  2550000   599000   875000   894999   340000
  1525000  2150000  1900000  2900000 10000000   160000   515000    40000
   480000   487000   386000   100000   490000   345000    85000   215000
   320000   459999    65000   720000   800000   520000   580000   420000
  1250000   830000   335000   329000   775000   524000   245000   910000
   900000   250000  1100000   640000   409999   165000    80000  1000000
   110000   170000  1380000  1300000   530000   310000  1150000   380000
   404000   114999   465000   210000  4600000   509999    90000   409000
   395000    81000   555000   229999  1210000   325000    60000    89000
   434999  3000000   570000  1550000    64000   140000   503000  3800000
  2000000   740000   750000  5400000  5500000   779000  2475000   240000
  1085000    95000  1290000   219000   260000   840000   292000   385000
  1050000   735000   560000   585000   760000   430000   643000   285000
   660000   268000    88000   561000   755000   125000  1600000   801000
   317000   495000   236000  1800000   415000   217000   466000   605000
  1350000   337000   655000   196000   299000   634000  1225000   238000
   590000   195000   715000   168000   531000  1625000   145000   441000
    67500   321000   298000  1075000   681000   810000   573000   101000
  1019999   244000  1165000   241000   327000   620000   204999   576000
   695000   231000   261000   371000   421000   710000  1040000   199000
   305000  1745000   611000    75000    55000  1325000   885000   108000
   272000  1125000   388000   757000   626000   793000  3750000  1400000
  1925000   844999  2940000  1825000   890000   770000   467000   711000
   511000   166000   392000   833000    35000   880000  1030000   521000
   451000   472000   649000   429000  1576000   825000   470000  1560000
  1950000   786000   731000   191000   651000   445000    93150   403000
   144000    52000   302000   869999   155000   246000   405000    70000
  3250000  1143000   635000  1575000   318000   152000   248000  1465000
  1080000   678000  1460000   756000   665000  2950000   377000  1675000
   169000   685000   749000   899000   259000   416000   773000   889000
   118000   256000  1888000    99000  2280000    31504   295000   444000
   860000  2450000   198000  2825000  2125000  3350000   365000   919999
  2600000   839000  1594000   161000   423000   551000    50000   411000
  1530000   376000  1850000    86000   822000    30000   252000   131000
   128000   790000   604000  1175000   141000   151000   577000    83000
   946000   356000   955000   978999   348000   399000   362000   499000
   358000   278000   426000   211000  1270000   426999   960000  1031000
    72000   479000   764000  2199000  3600000   184000   119000   267000
   328000   112999    39000  2850000  2800000  3900000   221000   631000
   308000   940000   361000   355000  7200000   930000  1670000   980000
   486000   242000   654000   835000   408000    94500  1110000   438999
   911999   693000   449000  1390000   698000   539000   765000   562000
   571000   589000   227999  1151000  1295000  4090000   347000   493000
   545000  1475000   287000  1630000   501000   181000  2711000  1035000
   316000   813000  1190000   378000   156000   194000   754000    55599
  1511000    78000   250999   312000    66000   369000   149000   752000
   142000   111000   206000   567000   699000   632000   905000  1145000
    58000   179000  1515000   178000   222000   387000  1405000   282000
   458000   157000   497000  1227000   574000   121000  3100000    68000
  1920000   808000   512000   671000   799000  1365000  1490000    80500
   189000   892000    93000  1680000   667000   204000    42000   694000
   861999   476999  5200000  2375000  2750000  1385000   785000  1275000
  1748999   132000    33351  2064000   469000   906000   797000  6223000
  5923000  6523000   122000   891000   346000   532000   974000   483000
   138000   432000  1757000   412000   126000   736000   579000   187000
  4000000  1898999   482000   401000   737000   291000   999000   957000
   834000   297000   669000   616000   129000   115999   861000  3251000
   911000   148000   433000   488000    67000   516000   136000    29999
   124000  1938000   172000    59000  3500000   537000   849000   851000
   597000   741000   556000   549000   679000   969999  1237000    31000
   666000    54000   374000   455999   453000   258000   428000   262000
   389000   339000   277000   578000   684000   949000   227000  5830000
   721000   751000  2400000   990000  1282000  1147000   558000   722000
   563000  2300000  2200000   488999   276000   434000   372000  1520000
   202999   541000   254000   528000   102000   311000   977000  1251000
  1445000    98000    59259   153000  1051000   591000   319000   454000
  1160000  3300000   791000   802999  1789999  1044999   279000   944999
  1859000  2175000   430999   803999   934000   357000   965000   341000
    75527  1132000   352000   252999   313000   391000  2051000   212000
  2575000  1516000   359000   209000  1220000  1180000    45957  1265000
   382000   526000    90177    46000   307000   746000]
===============
uniwue values ofkm_driven
[ 145500  120000  140000  127000   45000  175000    5000   90000  169000
   68000  100000   40000   70000   53000   80000   50000   72000   35000
   28000   25000    2388   16200   10000   15000   42000   60000   76000
   28900   86300   23300   32600   10300   77000   99000   27800   49800
  151000   54700   64000   63000  127700   33900   59000  110000  147000
   30000  135000    9850   78000  170000   49000   32000   38000   44000
   12000   55500   61500  150000   37800  114000   48000   69000   13000
   76139   65000    1303   31800   20000   17000   21000   37000   29500
    7500   19000   41000   39000   22000   47000   72200   49900    9000
   11000   95000   18000   46100   16000    9654   24300   42163    8000
   71000    9500   36600   14000  120600   86000    7800   31377   75000
   93000  125000   13500  162500   92500  158000    2000  181000  193000
  122358   95200   33033   63063  207890    7976   16500   99361   33000
   80100  160000   26000   29000   58000   36000  185000  162000   67000
   52000   68089   58343   38817   56494   79328    5621   25538   69779
   56290    8500    7032  106000  240000  214000    1000  265000  134000
    2136  250000  130000   99500  119000   44665  123000   34000  146000
   85000  201850   46000  190000   43000   14200   19100   54000  101500
   55735    1500  181491   53319   40906   31711   43755   66693   59549
   94385   73730  221889  116104  150546  148120  142000  126000   24000
  101000  168000  165000   98500   98000   77800   51492  136000   91000
  167000  180000    6000   91500  132478   51000  105000  137500   88000
  156060  107000   57000  300000  200000   18945  144000  107825    1620
```

```
 141000  108000   55000   37944   90400   96000  193623   63186  219000
  23000   94000  140500  143000  195000  360003    8079  114368   79990
  81632  155201  101504   90165   86017   85036  125531  206000   49700
  74000    6550   56000   41779   31000   13534    7000   28156   63309
 155000   11500    3000  157000   27000   26300   89100  104000  210000
 216000   84000  161000   72113  113000   96443   51500  104500   49102
  81000  227000  117000    8576   19723   21446   52289   52236   54470
  25471   33928   59736   54290    2118   61379   58544   74381   63982
  56429   19500   56832   30528   65285   20102   52380   74250   57247
 178000   42323   42462   54723   42545   20375   43978   40200   27233
  28832  103000  142500  115000   35500   17500   92000    4500   83000
   6750   18500   50478    9200    2300    4295    5229    4701    6757
  28182   11533   11688   57728   23712   70670   43381   79011   58500
  43500   12700   49455   46815  334000  100875  118000  128000  144030
 176000   89000   97000   12800   65100   15200   82000  116500  145000
  73000  113226  160500  132000  217000  139000    3500   62000 1500000
 120005   16830   93468   16589   53201   25272   53619    6001  202000
  71717   84321  177000  205000   41800   88700  188000  225000    5500
   4875    2700   58511   26235   26766    1600   24522   13663   58945
 375000  156975   27620  143200  138000  189000  156000   10800   67500
 255000   71850  112000  159000   76131   51777   64788   46951   74537
  23500   68600   13887   66000   24400   27500   67840  380000  112880
  61915  184000    6200   16700    7967    9656   68609   33360   14317
  87237   62960   26634  163720   67998   20706  100581   67303   79139
  84784  130376   33500   22053   78432  270000    7200  117500   78500
 115717    4000   26432   46702   46706   53523   56365   49563   56432
  52365   53512   46533   58632   65300   39395  230000   32500   64800
 248000   66444  218463  153000   87185  137000    3564   48756   80322
 173000  148000    6500  220000  108957   22200  260000   42108   79000
  17100   43001  100600   92385  116000   72500   82500   50600  121000
 101903   85568    3100  186000   59300   89322   48300   23511   22512
  62900   52442   34156   34152   62841   38600   27654   48698   32995
  90150   85700   25339   41232   68240   92651   88754   23700   69500
  28050   22947    7672   36659   44500    9900  124316    6544   49600
 198000  103994   35008   31100   42312   54891   18890   75010    4337
  14495   83585   48406   56239   33243   15858   87847   87000  112072
  52269   15780 2360457  108800   12500  577414  121941  152186   82246
  89580    2789   35278    2860   78562   36088  129000  500000  133000
  64500   24500   45500   71500   93500   60300   44391   56315   15151
 114321   37333  330000  291000   21500  197000   40300   30030   29029
  72072   82082   80600   58609   33003  165500    5800   68700  102996
  96500   52412   85472   87452   66530   32331  172000   23456  136500
 111000   42500   47200  246000    5200   10500   93331  109000    5400
 102000   29700    7600   29340   80800   24265   56900   88200   40800
   8588  305000  212000  101200  179150   84487  264000   51146  298000
 149000  291977   36800   52200  248200    7720   57882  242000   55380
  18816   33019   47747   45900   68697  147279   24700   43526   55885
  16034  156040   93415  122000   68519   22966    2350   59872   91182
  59500   75500  152500   69123   77524   40523   23600  290000   24177
 121779   15381   74800  109322  475000   61000   80500   65755    2600
  15500   50700  440000  152000  200400    7300    9750  157138  285000
 127991   13120   87500   62200   87540  231438  239451  154000  426000
 182000  215000   77300   61260   17601    4773   57900    8600   78010
  83844   23999   91400  178500   66953  166000  131000  100750   48676
  79500  200185   49025   76460   47370   45217  222300   44600   28080
  28800   21900   28100    7673   96272   22500    7400  370000  150360
  42130  147500   56194   54188   54043   66657   35582   54327    7949
  59734   49185   56389   58245   36422   42535   40736   56246   55403
 105358   49200   81500    3177   39414   30078   44588   49907   67082
 112879   10200    1300  175802  192000   73840  320000   24857   26442
```

```
   50800   40142    9400   12584    4300   14548   44772   48500   28782
   28451   36500   44885  131111   58400   23400    3010  163000   77088
  136511   36710   28180   44077   55768  123278   26500   18484    2560
  103655   56975  187000  218000   73257   26320   38426  116700  176062
   77500   39500   55896   30400  164000  271000   75262   91863   47552
   14700   19700  145241   95500   53473   75958   31596   85710  129627
   55130   22522   51856   29434   34500    6825  112011   53534  248119
   24019   19600  376412  183000  108916   59865   68140   53190   70100
   47725   70195   77395   74321   80235   77150   61100   82300   82050
   29899   59235  201000   77215   48228   58559   50856   18300   59292
   21147   61173   91567   33400   92686   67600   49500  112048   30646
   43011  104300   73500  280000   35700   22700   21871   55425   37659
   37500   19800   45629   68850   45775   68203   97343   82507   59400
  125876   56800   65204    2286   15732   49523   36521   64481   49060
   37161   60175   49957   43235   50699       1   50074   30154   26263
   84925   46357   46737  110048  186388   20171   37151   58161   50171
   55161    9599   16151   28161   80868  194000  191000]
===============
uniwue values offuel
['Diesel' 'Petrol' 'LPG' 'CNG']
===============
uniwue values ofseller_type
['Individual' 'Dealer' 'Trustmark Dealer']
===============
uniwue values oftransmission
['Manual' 'Automatic']
===============
uniwue values ofowner
['First Owner' 'Second Owner' 'Third Owner' 'Fourth & Above Owner'
 'Test Drive Car']
===============
uniwue values ofmileage
['23.4' '21.14' '17.7' '23.0' '16.1' '20.14' '17.3' '23.59' '20.0' '19.01'
 '19.3' '18.9' '18.15' '24.52' '19.7' '22.54' '21.0' '25.5' '26.59' '21.5'
 '20.3' '21.4' '24.7' '18.2' '16.8' '24.3' '14.0' '18.6' '33.44' '23.95'
 '17.0' '20.63' '13.93' '16.0' '17.8' '18.5' '12.55' '12.99' '14.8' '13.5'
 '26.0' '20.65' '27.3' '11.36' '17.68' '14.28' '18.53' '14.84' '21.12'
 '20.36' '21.27' '18.16' '22.0' '25.1' '20.51' '21.66' '25.2' '22.9'
 '16.02' '20.54' '22.77' '15.71' '23.1' '19.02' '19.81' '26.2' '16.47'
 '15.04' '19.1' '21.79' '18.8' '21.21' '15.37' '11.79' '19.0' '14.3'
 '15.8' '15.1' '19.09' '22.32' '21.9' '14.53' '21.63' '20.85' '20.45'
 '19.67' '23.01' '20.77' '17.92' '17.01' '22.37' '19.33' '9.5' '12.83'
 '22.48' '16.78' '14.67' '15.0' '13.96' '18.0' '12.07' '26.21' '10.8'
 '16.3' '13.6' '14.74' '15.6' '19.56' '22.69' '19.16' '18.12' '12.1'
 '17.5' '42.0' '20.4' '21.1' '19.44' '13.0' '21.43' '22.95' '16.2' '15.3'
 '28.09' '17.4' '19.4' '26.6' '17.6' '28.4' '14.1' '25.17' '22.74' '17.57'
 '16.95' '19.49' '17.21' '13.2' '14.2' '26.8' '25.4' '11.5' '27.28'
 '17.97' '12.8' '16.55' '12.05' '14.07' '21.02' '11.57' '17.9' '15.96'
 '17.1' '17.19' '21.01' '24.0' '25.6' '21.38' '23.84' '23.08' '14.24'
 '20.71' '15.64' '14.5' '16.34' '27.39' '11.1' '13.9' '20.88' '20.92'
 '23.8' '24.4' '15.29' '21.19' '22.5' '19.6' '23.65' '25.32' '23.5' '16.6'
 '23.9' '20.8' '27.62' '12.9' '25.44' '17.88' '22.7' '17.2' '15.42'
 '19.68' '18.7' '15.4' '19.34' '22.71' '25.8' '13.7' '12.2' '18.49' '9.0'
 '0.0' '13.58' '10.1' '20.5' '25.0' '10.5' '22.07' '22.3' '15.26' '20.62'
 '27.4' '23.2' '14.4' '18.4' '30.46' '14.02' '11.0' '20.6' '22.05' '20.2'
 '18.1' '22.1' '19.87' '13.01' '18.06' '26.1' '16.52' '13.55' '24.2'
 '25.83' '11.2' '17.09' '21.03' '17.45' '21.64' '21.94' '13.87' '19.98'
 '20.52' '23.57' '11.7' '17.43' '18.88' '13.68' '11.18' '20.89' '11.8'
 '19.62' '21.7' '14.9' '19.5' '10.91' '15.7' '20.73' '15.85' '20.7'
 '14.23' '16.5' '17.36' '12.6' '16.36' '14.95' '16.9' '19.2' '16.96'
 '22.15' '18.78' '19.61' '17.71' '18.3' '19.12' '19.72' '12.0' '11.4'
 '23.03' '11.07' '15.9' '17.67' '20.46' '13.1' '13.45' '24.8' '15.73'
 '15.11' '12.7' '21.2' '20.38' '21.56' '13.22' '14.49' '15.05' '23.26'
 '15.41' '13.8' '22.27' '32.52' '14.66' '12.12' '16.84' '14.09' '14.7'
 '13.4' '15.5' '13.49' '11.88' '14.6' '10.75' '24.5' '11.74' '16.07'
 '15.63' '26.3' '23.7' '25.47' '17.05' '23.3' '11.9' '13.38' '20.86'
 '10.9' '18.25' '15.2' '20.37' '21.8' '11.96' '24.04' '19.69' '13.73'
 '21.04' '25.01' '10.93' '24.29' '13.44' '20.07' '19.08' '20.34' '11.68'
 '12.5' '12.3' '23.87' '16.38' '17.42' '10.0' '18.24' '10.71' '19.59'
 '16.7' '19.83' '21.76' '16.05' '20.28' '16.25' '16.73' '18.48' '14.99'
 '18.76' '16.4' '19.64' '14.94' '17.11' '22.8' '32.26' '33.0' '12.4'
 '18.44' '16.09' '12.62' '21.13' '15.17' '21.73' '21.72' '12.85' '14.81'
 '13.24' '21.49' '14.62' '26.83' '11.45' '12.08' '15.74' '11.3' '14.21'
 '11.72' '16.51']
===============
uniwue values ofengine
[1248. 1498. 1497. 1396. 1298. 1197. 1061.  796. 1364. 1399. 1461.  993.
 1198. 1199.  998. 1591. 2179. 1368. 2982. 2494. 2143. 2477. 1462. 2755.
 1968. 1798. 1196. 1373. 1598. 1998. 1086. 1194. 1172. 1405. 1582.  999.
 2487. 1999. 3604. 2987. 1995. 1451. 1969. 2967. 2497. 1797. 1991. 2362.
 1493. 1599. 1341. 1794.  799. 1193. 2696. 1495. 1186. 1047. 2498. 2956.
 2523. 1120.  624. 1496. 1984. 2354.  814.  793. 1799.  936. 1956. 1997.
 1499. 1948. 2997. 2489. 2499. 2609. 2953. 1150. 1994. 1388. 1527. 2199.
  995. 2993. 1586. 1390.  909. 2393. 3198. 1339. 2835. 2092. 1595. 2496.
 1596. 1597. 2596. 2148. 1299. 1590. 2231. 2694. 2200. 1795. 1896. 1796.
 1422. 1489. 2359. 2197. 2999. 1781. 2650. 1343. 2446. 3498. 2198. 2776.
 1950.]
===============
uniwue values ofmax_power
[ 74.    103.52   78.     90.     88.2    81.86   57.5    37.     67.1
  68.1   108.45   60.     73.9    67.     82.     88.5    46.3    88.73
  64.1    98.6    88.8    83.81   83.1    47.3    73.8    34.2    35.
  81.83   40.3   121.3   138.03  160.77  117.3   116.3    83.14   67.05
 168.5   100.    120.7    98.63  175.56  103.25  171.5   100.6   174.33
 187.74  170.     78.9    88.76   86.8   108.495 108.62   93.7   103.6
  98.59  189.     67.04   68.05   58.2    82.85   81.8    73.    120.
  94.68  160.     65.    155.     69.01  126.32  138.1    83.8   126.2
  98.96   62.1    86.7   188.    214.56  177.    280.    148.31  254.79
 190.    177.46  204.    141.    117.6   241.4   282.    150.    147.5
 108.5   103.5   183.    181.04  157.7   164.7    91.1   400.     68.
  75.     85.8    87.2    53.    118.    103.2    83.     84.     58.16
 147.94   74.02   53.3    80.     88.7    97.7   121.36  162.    140.
  94.    100.57   82.9    83.11   70.    153.86  121.    126.3    73.97
 171.     69.     99.6   102.    105.     63.     79.4    97.9    63.1
  66.1   110.    174.5    53.26   73.75   67.06   64.08   37.5   189.3
 158.8    61.7    55.2    71.01   73.74  147.9    71.     77.    121.4
 113.4    47.    130.     57.6   138.     52.8    53.64   53.5    76.8
  82.4   113.42   76.     84.8    56.3   218.    112.     92.    105.5
 169.     95.     72.4   115.    152.     91.2   156.     74.9    62.
 105.3    73.94   85.    118.3    72.    147.51   58.     64.    126.24
  76.9   194.3    99.23   89.84  123.7   118.35   99.    241.    136.
 261.4   104.68   37.48  104.     63.12   91.7   102.5   177.6    45.
 123.37  147.8   184.     84.48   68.07   74.96  167.6   152.87  112.2
  83.83  197.    110.4   104.55  103.    103.3    66.    108.6   165.
 163.7   116.9    94.93  127.    198.5   179.5   120.69  121.31  138.08
 187.7    80.8    86.79   93.87  116.6   143.     92.7    88.     58.33
  78.8    64.4   125.    139.01  254.8   181.    258.     55.23  270.9
 265.   157.75  101.    186.    187.4   224.     64.9   148.     35.5
  89.75   32.8    91.72  106.     98.97   66.6    86.     65.3    98.82
 198.25   38.    142.    132.    174.57  178.    163.2   203.2   177.5
 175.     57.      80.84   68.4    167.67  170.63   52.     149.5    48.21
   0.    201.1    100.5   144.     194.4   168.7   104.5   103.26  116.4
  98.79   80.9     58.3   272.     235.    167.62  170.3   139.46  158.
 110.5    82.5    141.1    38.4    197.2   161.    194.    122.4   134.1
  60.2   134.     203.    135.1  ]
===============
uniwue values ofseats
[ 5.  4.  7.  8.  6.  9. 10. 14.  2.]
===============
```

In [30]:
```python
df['name'].nunique()
```

Out[30]:
```
31
```

In [31]:
```python
df['name'].replace(['Maruti', 'Skoda', 'Honda', 'Hyundai', 'Toyota', 'Ford', 'Re
                    'Mahindra', 'Tata', 'Chevrolet', 'Datsun', 'Jeep', 'Mercedes
                    'Mitsubishi', 'Audi', 'Volkswagen', 'BMW', 'Nissan', 'Lexus'
                    'Land', 'MG', 'Volvo', 'Daewoo', 'Kia', 'Fiat', 'Force', 'Am
                    'Ashok', 'Isuzu', 'Opel'],
                   [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 1
                    21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31],
                   inplace=True)
```

```
C:\Users\rahul\AppData\Local\Temp\ipykernel_11944\2045344005.py:1: FutureWarning:
A value is trying to be set on a copy of a DataFrame or Series through chained as
signment using an inplace method.
The behavior will change in pandas 3.0. This inplace method will never work becau
se the intermediate object on which we are setting values always behaves as a cop
y.

For example, when doing 'df[col].method(value, inplace=True)', try using 'df.meth
od({col: value}, inplace=True)' or df[col] = df[col].method(value) instead, to pe
rform the operation inplace on the original object.


  df['name'].replace(['Maruti', 'Skoda', 'Honda', 'Hyundai', 'Toyota', 'Ford', 'R
enault',
C:\Users\rahul\AppData\Local\Temp\ipykernel_11944\2045344005.py:1: FutureWarning:
Downcasting behavior in `replace` is deprecated and will be removed in a future v
ersion. To retain the old behavior, explicitly call `result.infer_objects(copy=Fa
lse)`. To opt-in to the future behavior, set `pd.set_option('future.no_silent_dow
ncasting', True)`
  df['name'].replace(['Maruti', 'Skoda', 'Honda', 'Hyundai', 'Toyota', 'Ford', 'R
enault',
```

In [32]:
```python
df['transmission'].unique()
```

Out[32]:
```
array(['Manual', 'Automatic'], dtype=object)
```

In [33]:
```python
df['transmission'].replace(['Manual', 'Automatic'],[1,2],inplace=True)
```

```
C:\Users\rahul\AppData\Local\Temp\ipykernel_11944\4195162967.py:1: FutureWarning:
A value is trying to be set on a copy of a DataFrame or Series through chained as
signment using an inplace method.
The behavior will change in pandas 3.0. This inplace method will never work becau
se the intermediate object on which we are setting values always behaves as a cop
y.

For example, when doing 'df[col].method(value, inplace=True)', try using 'df.meth
od({col: value}, inplace=True)' or df[col] = df[col].method(value) instead, to pe
rform the operation inplace on the original object.


  df['transmission'].replace(['Manual', 'Automatic'],[1,2],inplace=True)
C:\Users\rahul\AppData\Local\Temp\ipykernel_11944\4195162967.py:1: FutureWarning:
Downcasting behavior in `replace` is deprecated and will be removed in a future v
ersion. To retain the old behavior, explicitly call `result.infer_objects(copy=Fa
lse)`. To opt-in to the future behavior, set `pd.set_option('future.no_silent_dow
ncasting', True)`
  df['transmission'].replace(['Manual', 'Automatic'],[1,2],inplace=True)
```

In [34]:

```
df['seller_type'].unique()
```

Out[34]:

```
array(['Individual', 'Dealer', 'Trustmark Dealer'], dtype=object)
```

In [35]:

```
df['seller_type'].replace(['Individual', 'Dealer', 'Trustmark Dealer'],[1,2,3],i
```

```
C:\Users\rahul\AppData\Local\Temp\ipykernel_11944\1872772954.py:1: FutureWarning:
A value is trying to be set on a copy of a DataFrame or Series through chained as
signment using an inplace method.
The behavior will change in pandas 3.0. This inplace method will never work becau
se the intermediate object on which we are setting values always behaves as a cop
y.

For example, when doing 'df[col].method(value, inplace=True)', try using 'df.meth
od({col: value}, inplace=True)' or df[col] = df[col].method(value) instead, to pe
rform the operation inplace on the original object.


  df['seller_type'].replace(['Individual', 'Dealer', 'Trustmark Dealer'],[1,2,3],
inplace=True)
C:\Users\rahul\AppData\Local\Temp\ipykernel_11944\1872772954.py:1: FutureWarning:
Downcasting behavior in `replace` is deprecated and will be removed in a future v
ersion. To retain the old behavior, explicitly call `result.infer_objects(copy=Fa
lse)`. To opt-in to the future behavior, set `pd.set_option('future.no_silent_dow
ncasting', True)`
  df['seller_type'].replace(['Individual', 'Dealer', 'Trustmark Dealer'],[1,2,3],
inplace=True)
```

In [36]:

```
df['fuel'].unique()
```

Out[36]:

```
array(['Diesel', 'Petrol', 'LPG', 'CNG'], dtype=object)
```

In [37]:

```
df['fuel'].replace(['Diesel', 'Petrol', 'LPG', 'CNG'],[1,2,3,4],inplace=True)
```

```
C:\Users\rahul\AppData\Local\Temp\ipykernel_11944\140813953.py:1: FutureWarning:
A value is trying to be set on a copy of a DataFrame or Series through chained as
signment using an inplace method.
The behavior will change in pandas 3.0. This inplace method will never work becau
se the intermediate object on which we are setting values always behaves as a cop
y.

For example, when doing 'df[col].method(value, inplace=True)', try using 'df.meth
od({col: value}, inplace=True)' or df[col] = df[col].method(value) instead, to pe
rform the operation inplace on the original object.


  df['fuel'].replace(['Diesel', 'Petrol', 'LPG', 'CNG'],[1,2,3,4],inplace=True)
C:\Users\rahul\AppData\Local\Temp\ipykernel_11944\140813953.py:1: FutureWarning:
Downcasting behavior in `replace` is deprecated and will be removed in a future v
ersion. To retain the old behavior, explicitly call `result.infer_objects(copy=Fa
lse)`. To opt-in to the future behavior, set `pd.set_option('future.no_silent_dow
ncasting', True)`
  df['fuel'].replace(['Diesel', 'Petrol', 'LPG', 'CNG'],[1,2,3,4],inplace=True)
```

In [38]:
```
df.reset_index(inplace=True)
```

In [39]:
```
df
```

Out[39]:

	index	name	year	selling_price	km_driven	fuel	seller_type	transmission	ow
0	0	1	2014	450000	145500	1	1	1	F Ow
1	1	2	2014	370000	120000	1	1	1	Seco Ow
2	2	3	2006	158000	140000	2	1	1	Th Ow
3	3	4	2010	225000	127000	1	1	1	F Ow
4	4	1	2007	130000	120000	2	1	1	F Ow
...	...	...	...	...	...	...	...	...	
6713	8121	1	2013	260000	50000	2	1	1	Seco Ow
6714	8122	4	2014	475000	80000	1	1	1	Seco Ow
6715	8123	4	2013	320000	110000	2	1	1	F Ow
6716	8124	4	2007	135000	119000	1	1	1	Fou Ab Ow
6717	8125	1	2009	382000	120000	1	1	1	F Ow

6718 rows × 13 columns

In [40]:

```
df['owner'].unique()
```

Out[40]:

```
array(['First Owner', 'Second Owner', 'Third Owner',
       'Fourth & Above Owner', 'Test Drive Car'], dtype=object)
```

In [41]:

```
df['owner'].replace(['First Owner', 'Second Owner', 'Third Owner',
       'Fourth & Above Owner', 'Test Drive Car'],[1,2,3,4,5],inplace=True)
```

```
C:\Users\rahul\AppData\Local\Temp\ipykernel_11944\2564996163.py:1: FutureWarning:
A value is trying to be set on a copy of a DataFrame or Series through chained as
signment using an inplace method.
The behavior will change in pandas 3.0. This inplace method will never work becau
se the intermediate object on which we are setting values always behaves as a cop
y.

For example, when doing 'df[col].method(value, inplace=True)', try using 'df.meth
od({col: value}, inplace=True)' or df[col] = df[col].method(value) instead, to pe
rform the operation inplace on the original object.


  df['owner'].replace(['First Owner', 'Second Owner', 'Third Owner',
C:\Users\rahul\AppData\Local\Temp\ipykernel_11944\2564996163.py:1: FutureWarning:
Downcasting behavior in `replace` is deprecated and will be removed in a future v
ersion. To retain the old behavior, explicitly call `result.infer_objects(copy=Fa
lse)`. To opt-in to the future behavior, set `pd.set_option('future.no_silent_dow
ncasting', True)`
  df['owner'].replace(['First Owner', 'Second Owner', 'Third Owner',
```

In [42]:
```
df.drop(columns=['index'],inplace=True)
```

In [43]:
```
df
```

Out[43]:

	name	year	selling_price	km_driven	fuel	seller_type	transmission	owner	mil
0	1	2014	450000	145500	1	1	1	1	
1	2	2014	370000	120000	1	1	1	2	
2	3	2006	158000	140000	2	1	1	3	
3	4	2010	225000	127000	1	1	1	1	
4	1	2007	130000	120000	2	1	1	1	
...	...	...	...	...	...	...	...	...	
6713	1	2013	260000	50000	2	1	1	2	
6714	4	2014	475000	80000	1	1	1	2	
6715	4	2013	320000	110000	2	1	1	1	
6716	4	2007	135000	119000	1	1	1	4	
6717	1	2009	382000	120000	1	1	1	1	

6718 rows × 12 columns

In [44]:
```
input_data=df.drop(columns=['selling_price'])
output_data=df['selling_price']
```

In [45]:
```
x_train, x_test,y_train, y_test=train_test_split(input_data,output_data,test_siz
```

Model Creation

In [46]:
```
model = LinearRegression()
```

In [47]:
```
model.fit(x_train,y_train)
```

Out[47]:
```
▾ LinearRegression

LinearRegression()
```

In [48]:
```
predict=model.predict(x_test)
```

In [49]:
```
predict
```

Out[49]:
```
array([ 714429.51291266, 1154654.8955286 ,  641084.72096227, ...,
        230364.23622756,  492022.89751284,   92300.01128162])
```

In [50]:
```
x_train.head(1)
```

Out[50]:

	name	year	km_driven	fuel	seller_type	transmission	owner	mileage	engine
3359	1	2016	90000	2	1	1	1	18.5	1197.0

In [62]:
```
input_data_model = pd.DataFrame([[2,2022,9000,1,3,2,4   ,2.48,1995.0,187.74,5.0]
```

In [63]:
```
input_data_model
```

Out[63]:

	name	year	km_driven	fuel	seller_type	transmission	owner	mileage	engine	m
0	2	2022	9000	1	3	2	4	2.48	1995.0	

In [64]:
```
model.predict(input_data_model)
```

Out[64]:
```
array([2214971.69180985])
```

In [54]:
```
import pickle as pk
```

In [55]:
```
pk.dump(model,open('model.pkl','wb'))
```

In [ ]: