CAR PRICE PREDICTION

การทำนายราคารถยนต์ในอนาคต

01076032

ELEMENTARY DIFFERENTIAL EQUATIONS

AND LINEAR ALGEBRA

MEMBER

Capybaras & กอดคอลงเรือ

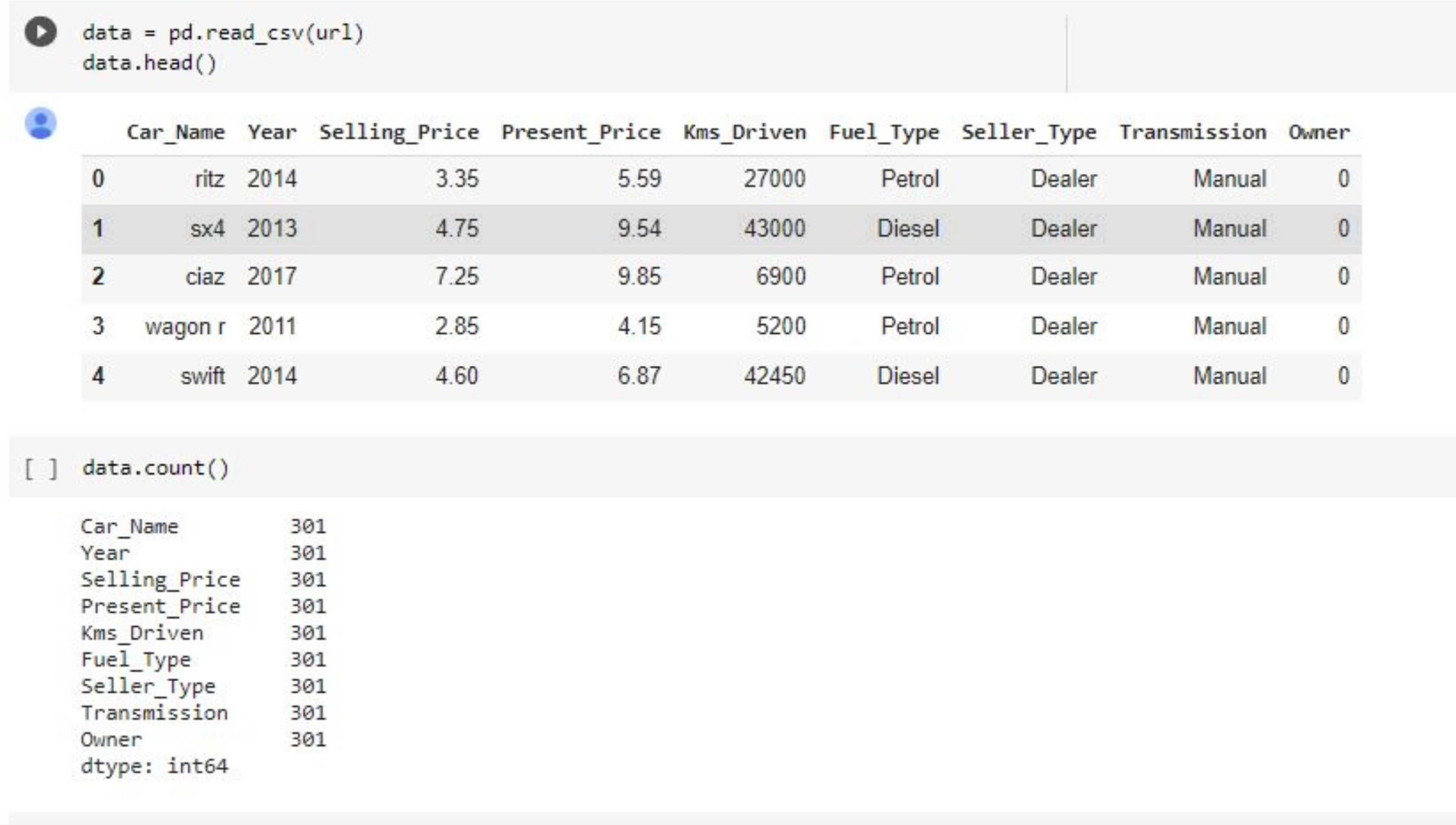
63010022	นายกฤต	รุ่งโรจน์กิจกุล	(ฟิลด์)
63010052	นายก้องเกียรติ	ชุนงาม	(ก้อง)
63010042	นายกฤษติภูมิ	เทียนงาม	(เอิร์ท)
63010332	นายณัฐพล	จำปานนท์	(โตโต้)

ขั้นตอนการทำงาน

1

นำข้อมูลเข้าสู่ระบบ

DATA



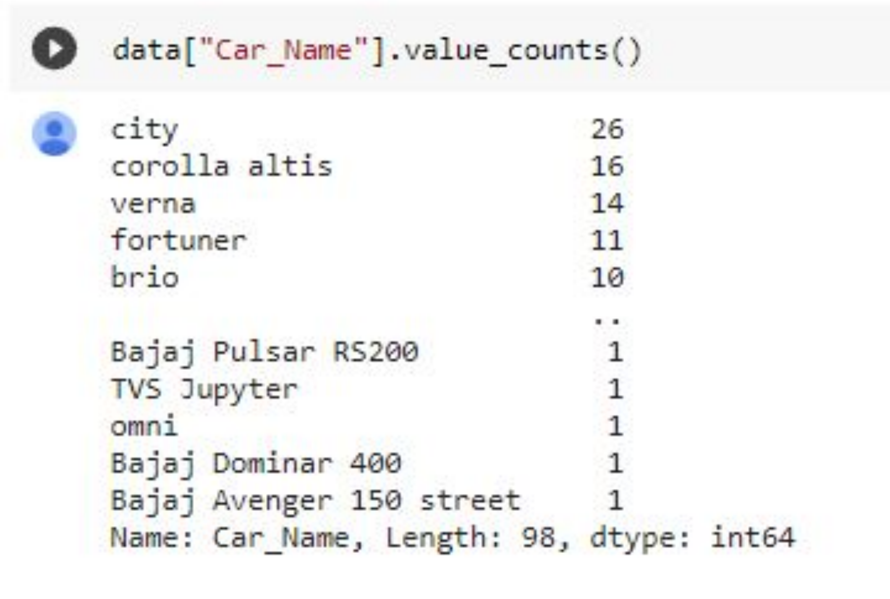
```
data = pd.read_csv(url)
data.head()
```

	Car_Name	Year	Selling_Price	Present_Price	Kms_Driven	Fuel_Type	Seller_Type	Transmission	Owner
0	ritz	2014	3.35	5.59	27000	Petrol	Dealer	Manual	0
1	sx4	2013	4.75	9.54	43000	Diesel	Dealer	Manual	0
2	ciaz	2017	7.25	9.85	6900	Petrol	Dealer	Manual	0
3	wagon r	2011	2.85	4.15	5200	Petrol	Dealer	Manual	0
4	swift	2014	4.60	6.87	42450	Diesel	Dealer	Manual	0

```
[ ] data.count()
```

```
Car_Name         301
Year             301
Selling_Price    301
Present_Price    301
Kms_Driven       301
Fuel_Type        301
Seller_Type      301
Transmission     301
Owner            301
dtype: int64
```

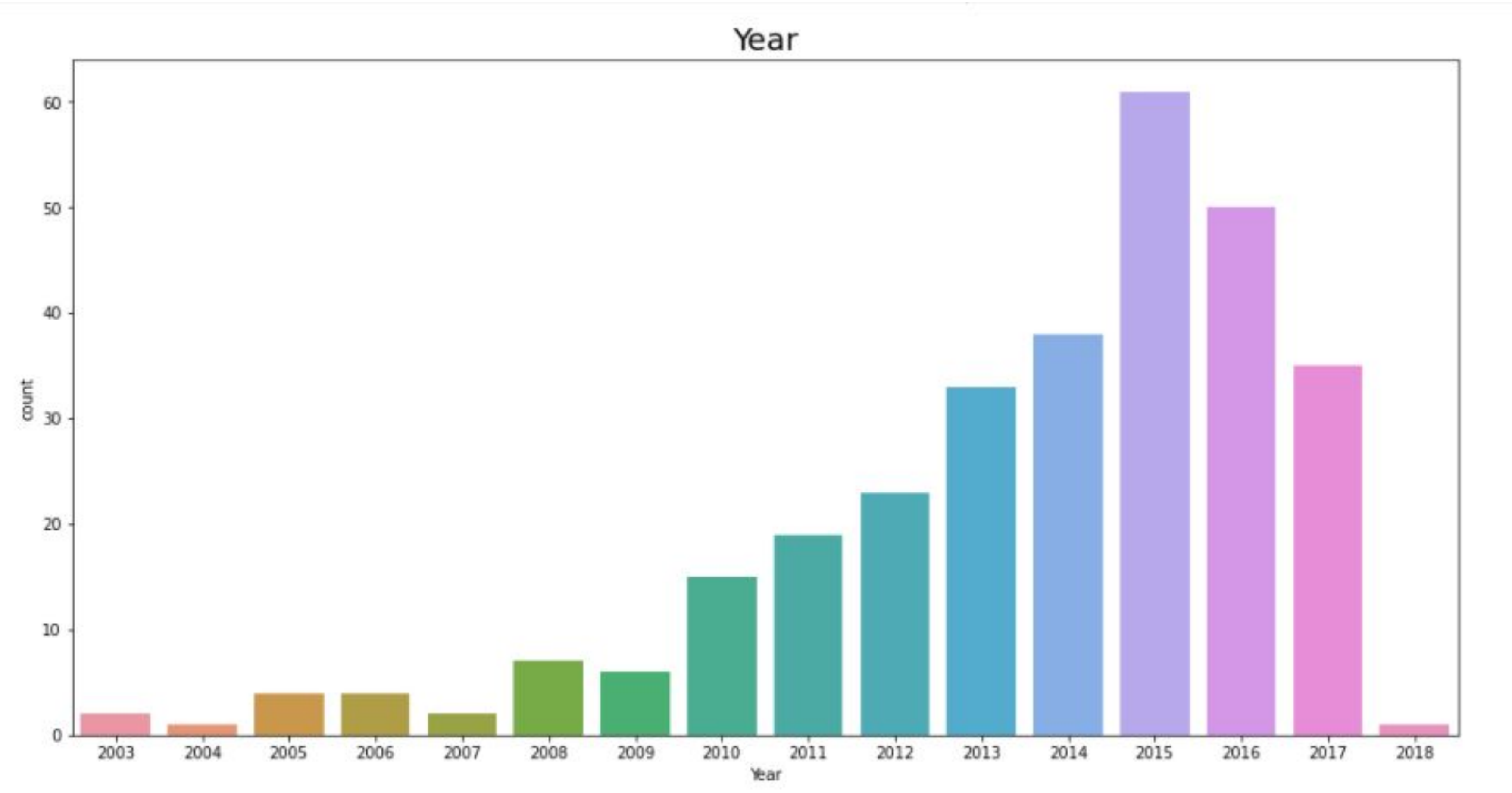
NAME



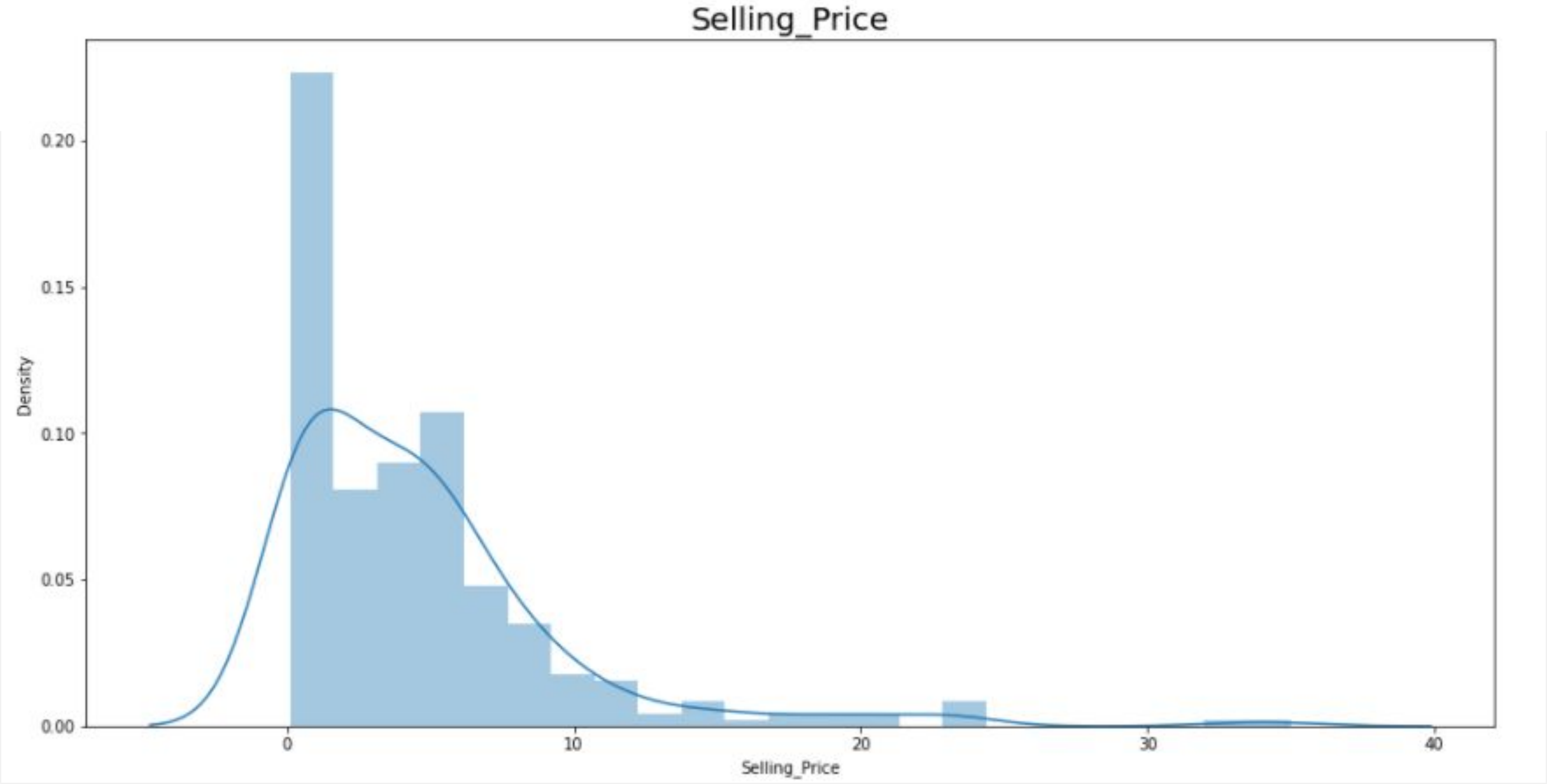
```
data["Car_Name"].value_counts()
```

```
city                        26
corolla altis               16
verna                       14
fortuner                    11
brio                        10
                            ..
Bajaj Pulsar RS200           1
TVS Jupyter                  1
omni                         1
Bajaj Dominar 400            1
Bajaj Avenger 150 street     1
Name: Car_Name, Length: 98, dtype: int64
```

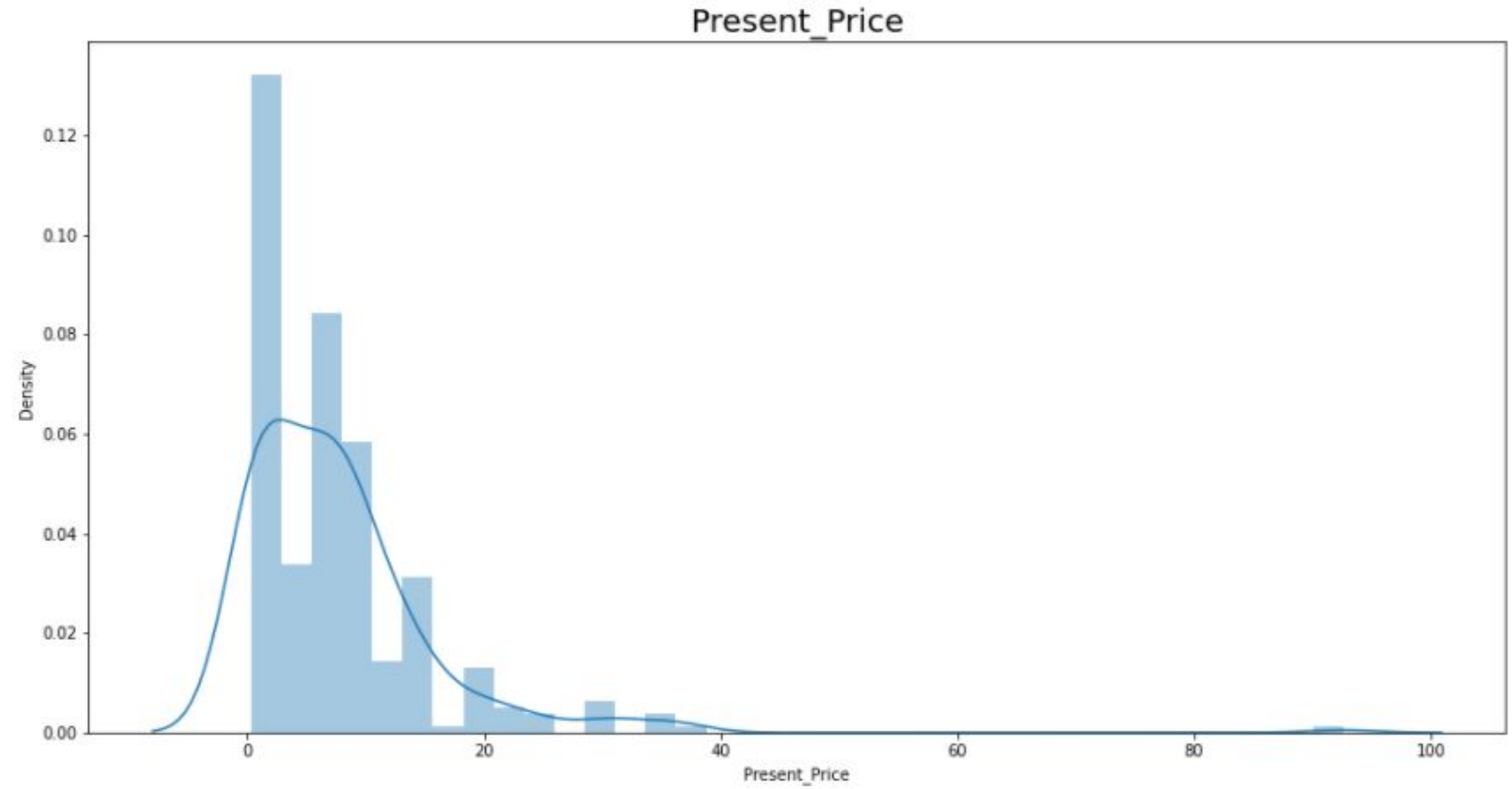
YEAR


Year
count
0
10
20
30
40
50
60
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
Year

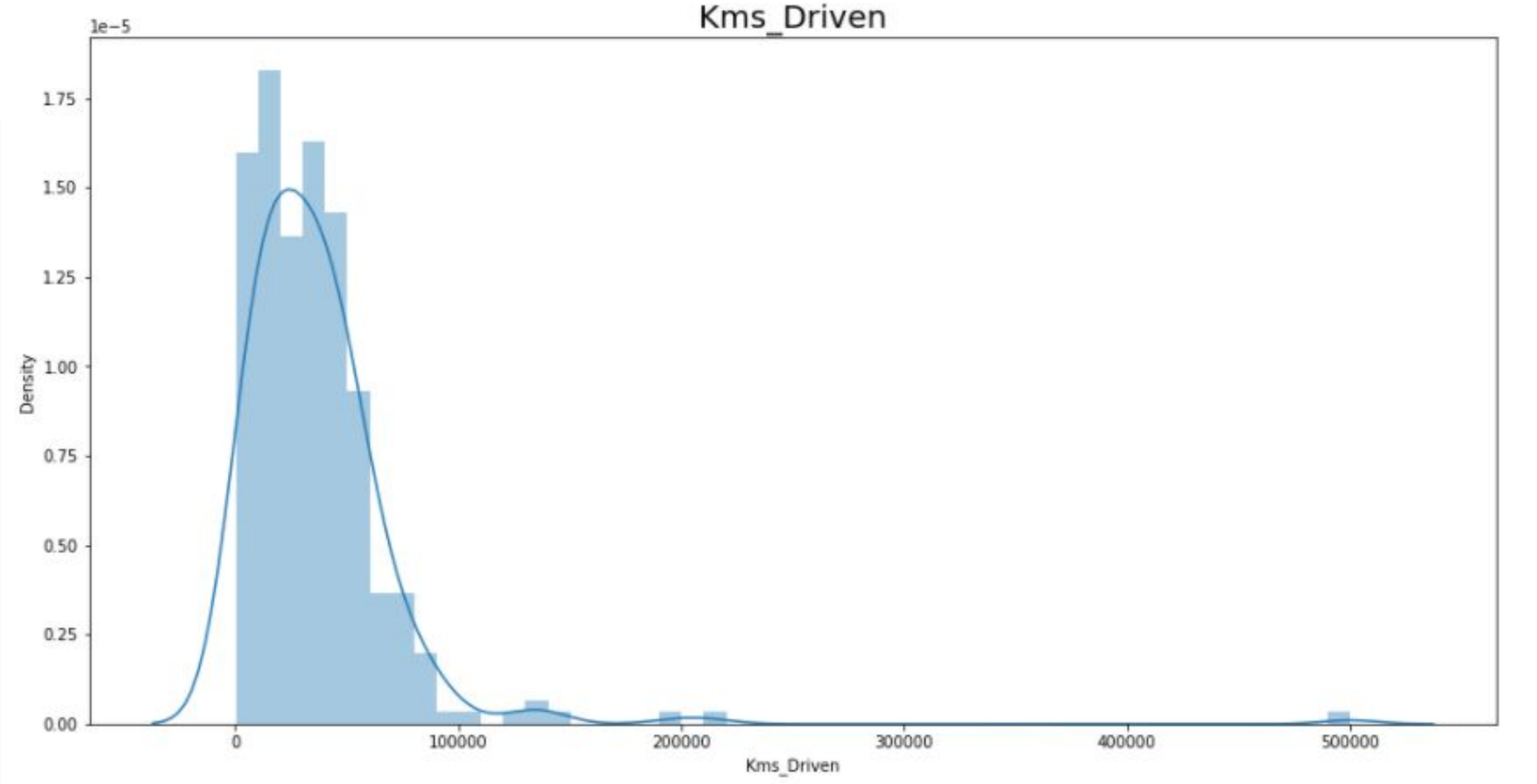
SELLING PRICE



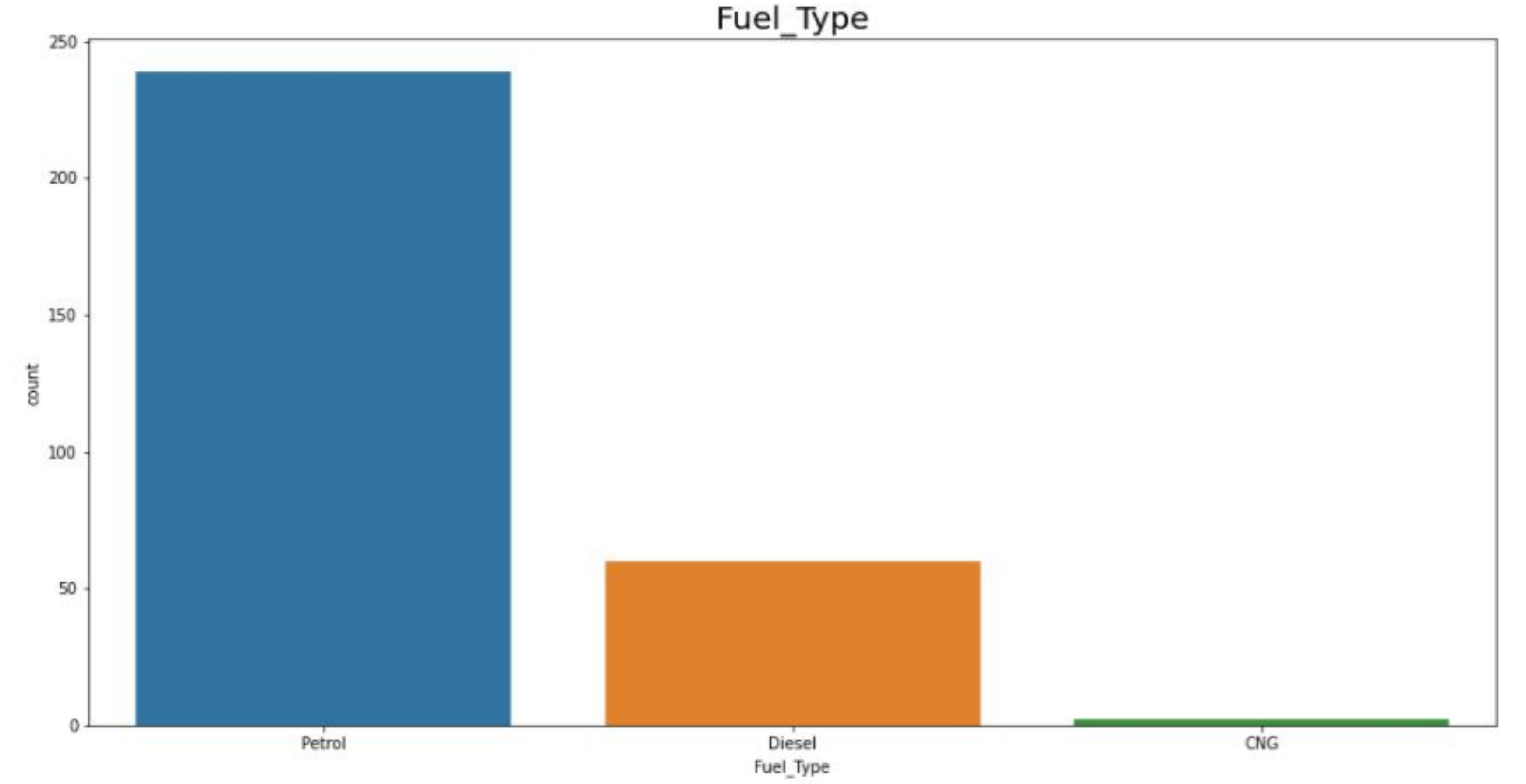
PRESENT PRICE



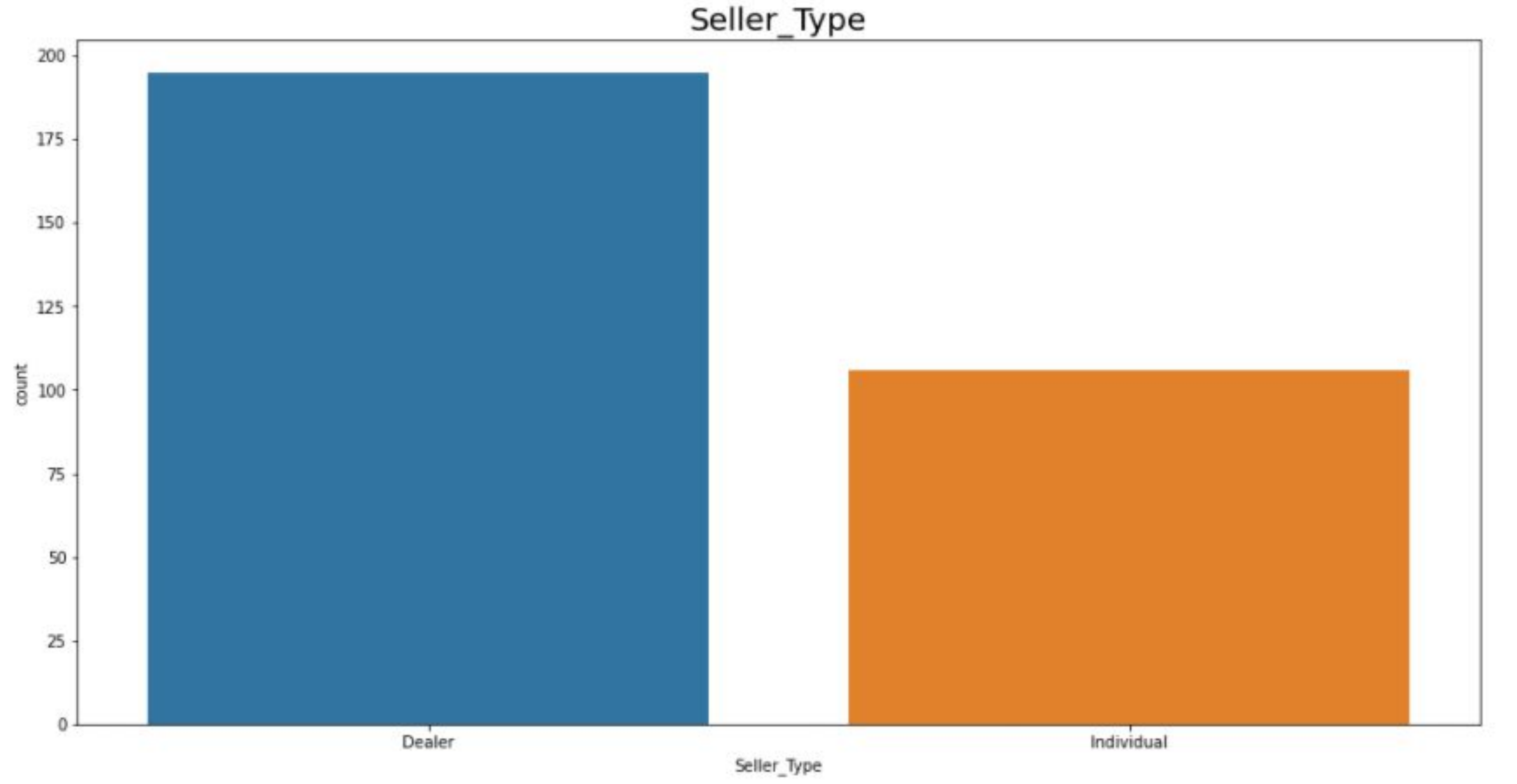
KMS DRIVEN


Kms_Driven
1e−5
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
Density
0
100000
200000
300000
400000
500000
Kms_Driven

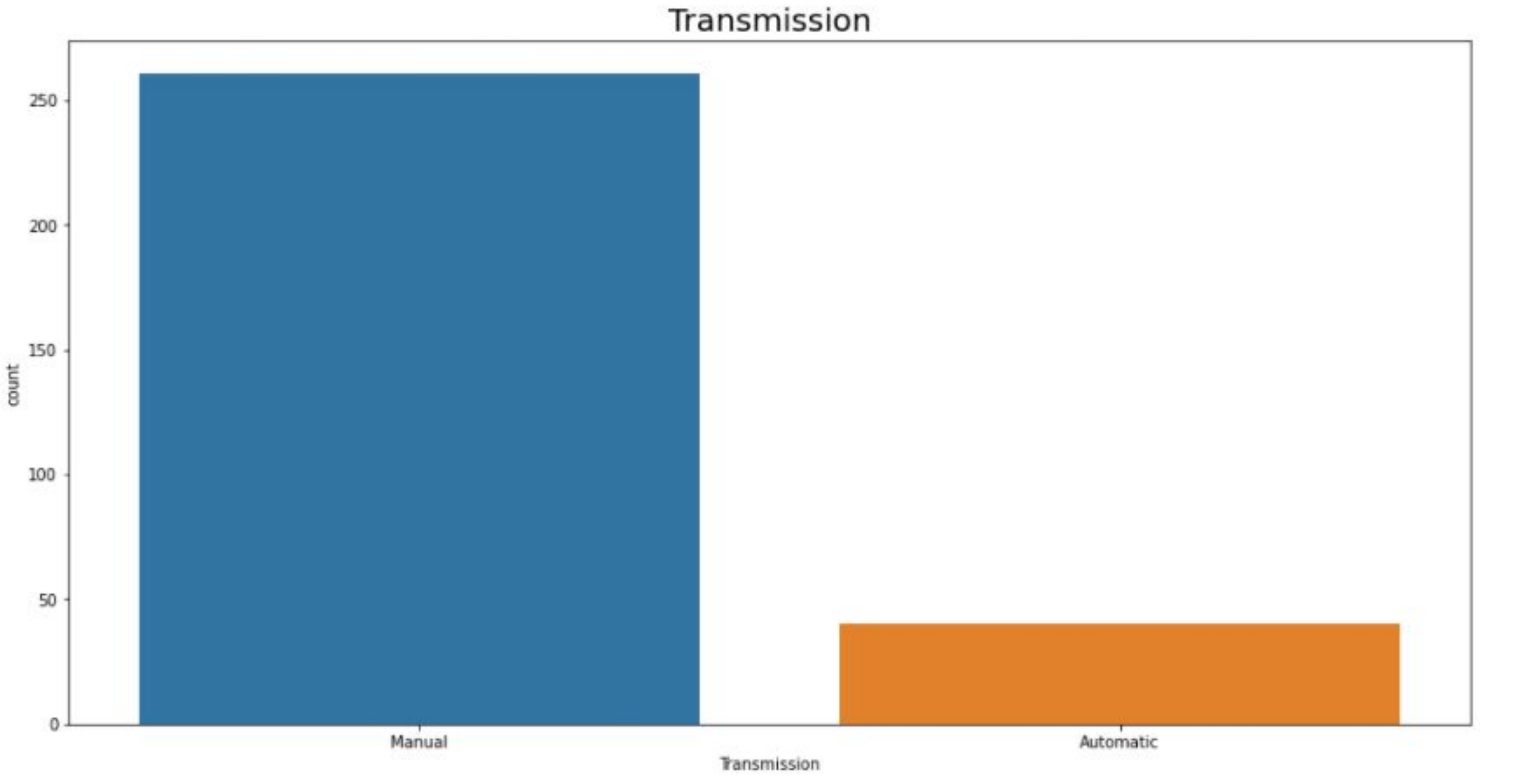
FUEL TYPE


Fuel_Type
250
200
150
100
50
0
count
Petrol
Diesel
CNG
Fuel_Type

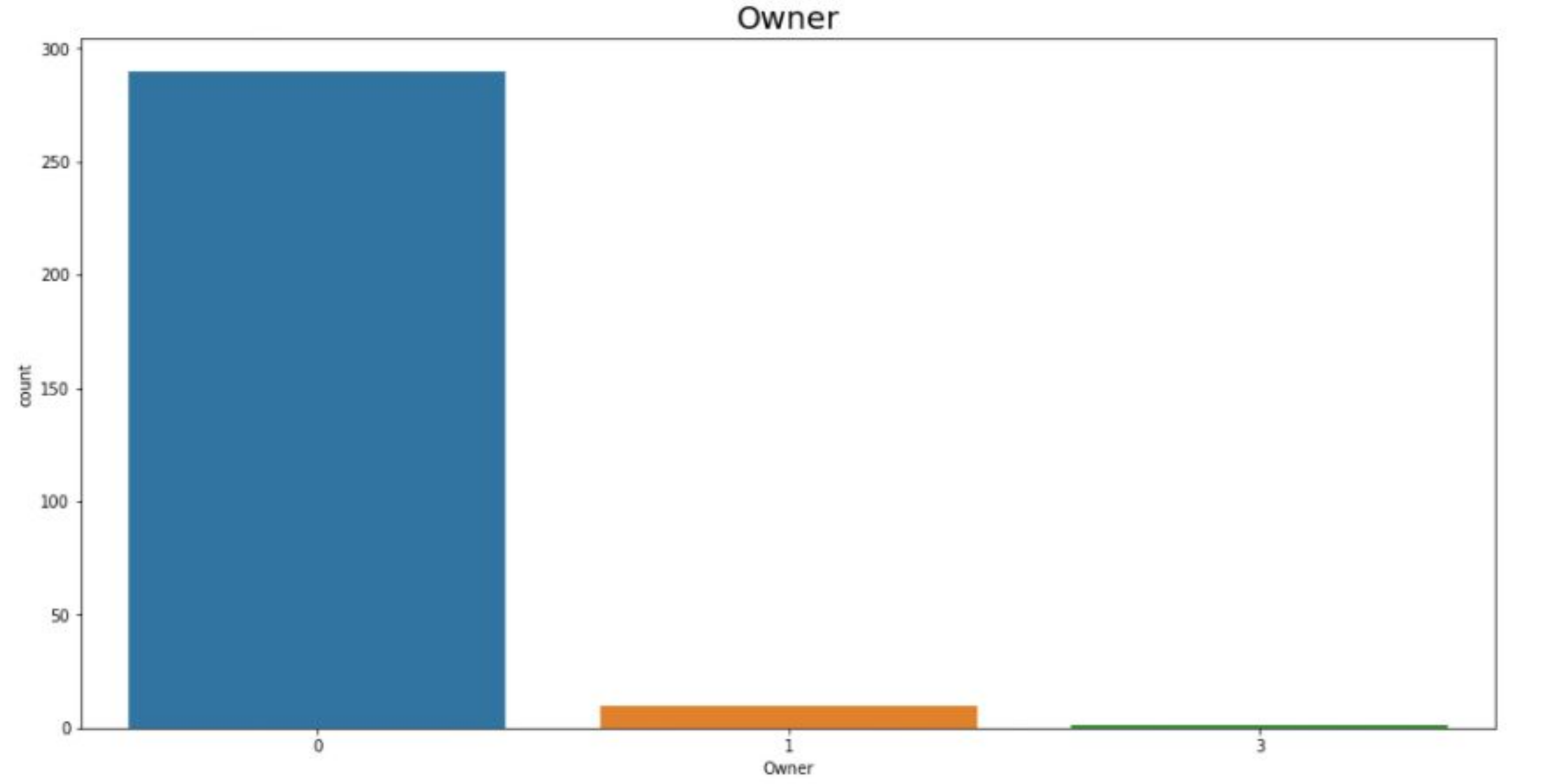
SELLER TYPE


Seller_Type
200
175
150
125
100
75
50
25
0
count
Dealer
Individual
Seller_Type

TRANSMISSION



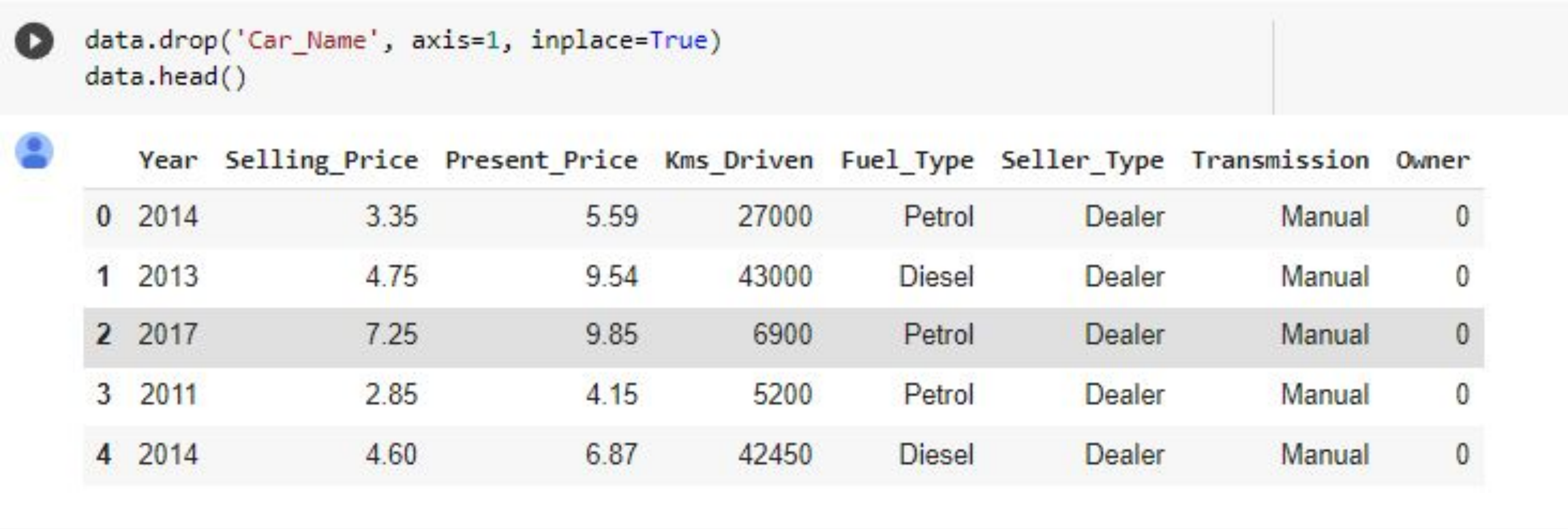
OWNER



2

ปรับแต่งข้อมูล

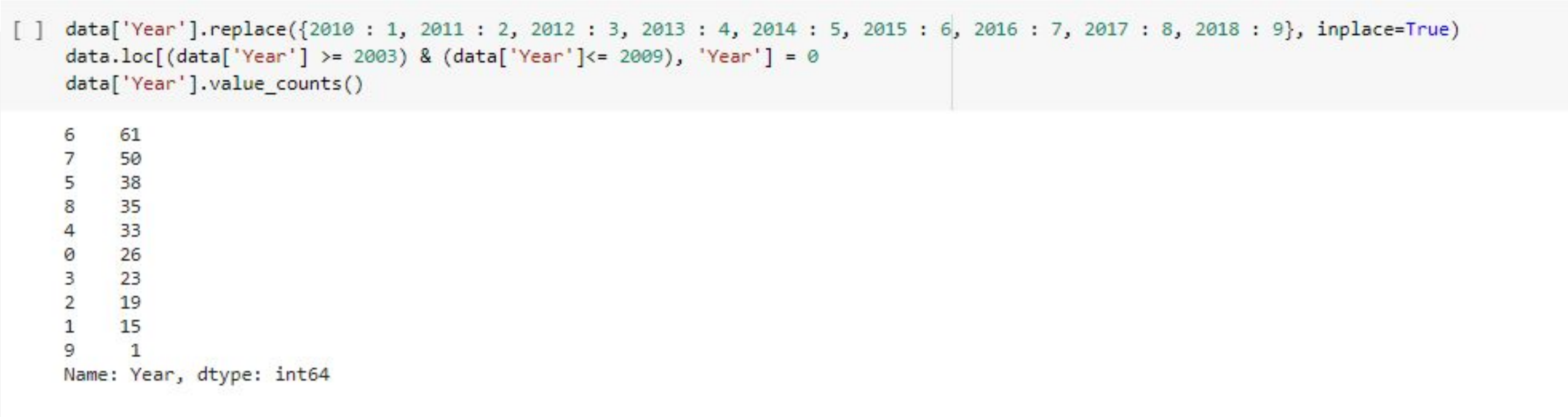
NAME



```
data.drop('Car_Name', axis=1, inplace=True)
data.head()
```

	Year	Selling_Price	Present_Price	Kms_Driven	Fuel_Type	Seller_Type	Transmission	Owner
0	2014	3.35	5.59	27000	Petrol	Dealer	Manual	0
1	2013	4.75	9.54	43000	Diesel	Dealer	Manual	0
2	2017	7.25	9.85	6900	Petrol	Dealer	Manual	0
3	2011	2.85	4.15	5200	Petrol	Dealer	Manual	0
4	2014	4.60	6.87	42450	Diesel	Dealer	Manual	0

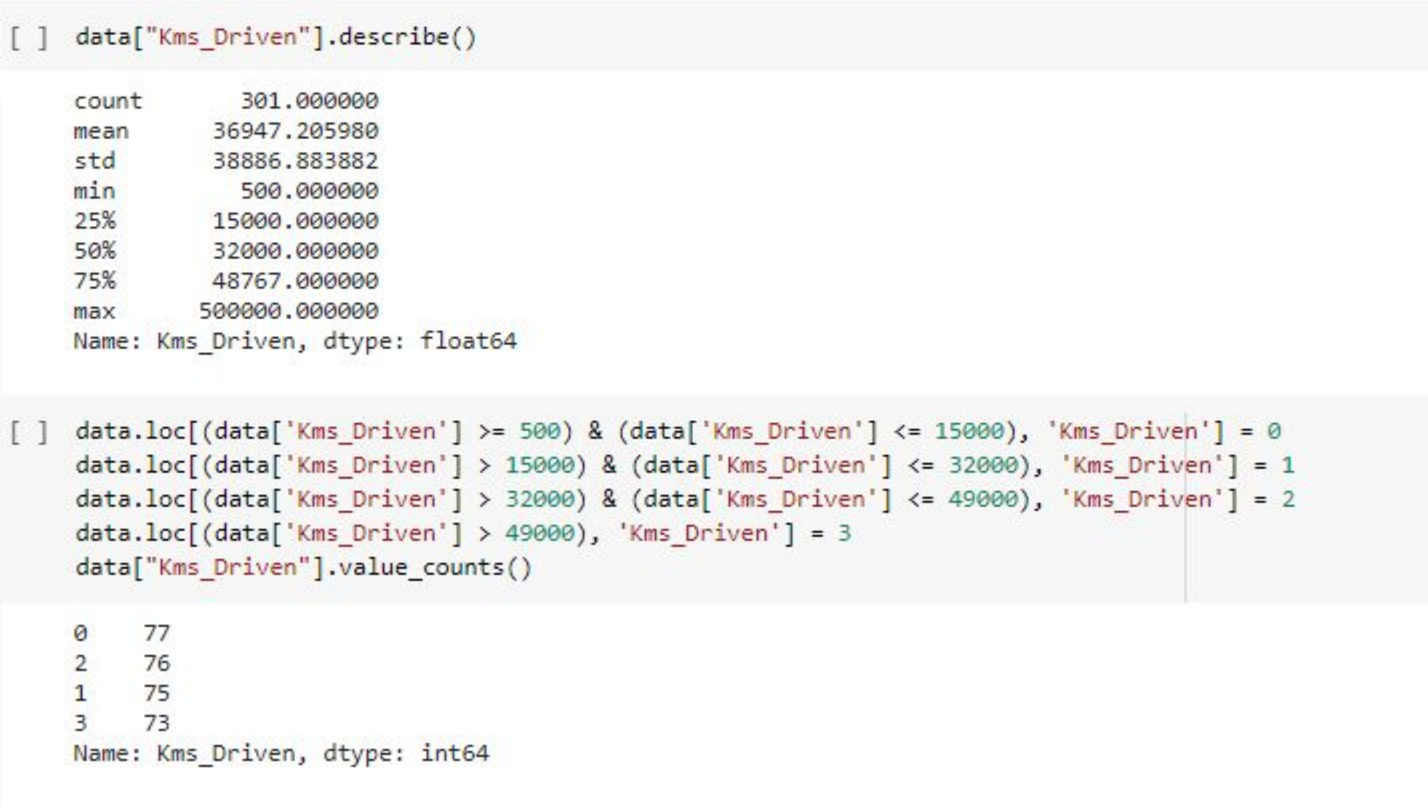
YEAR



```
[ ]  data['Year'].replace({2010 : 1, 2011 : 2, 2012 : 3, 2013 : 4, 2014 : 5, 2015 : 6, 2016 : 7, 2017 : 8, 2018 : 9}, inplace=True)
     data.loc[(data['Year'] >= 2003) & (data['Year']<= 2009), 'Year'] = 0
     data['Year'].value_counts()
```

```
6    61
7    50
5    38
8    35
4    33
0    26
3    23
2    19
1    15
9     1
Name: Year, dtype: int64
```

KMS DRIVEN



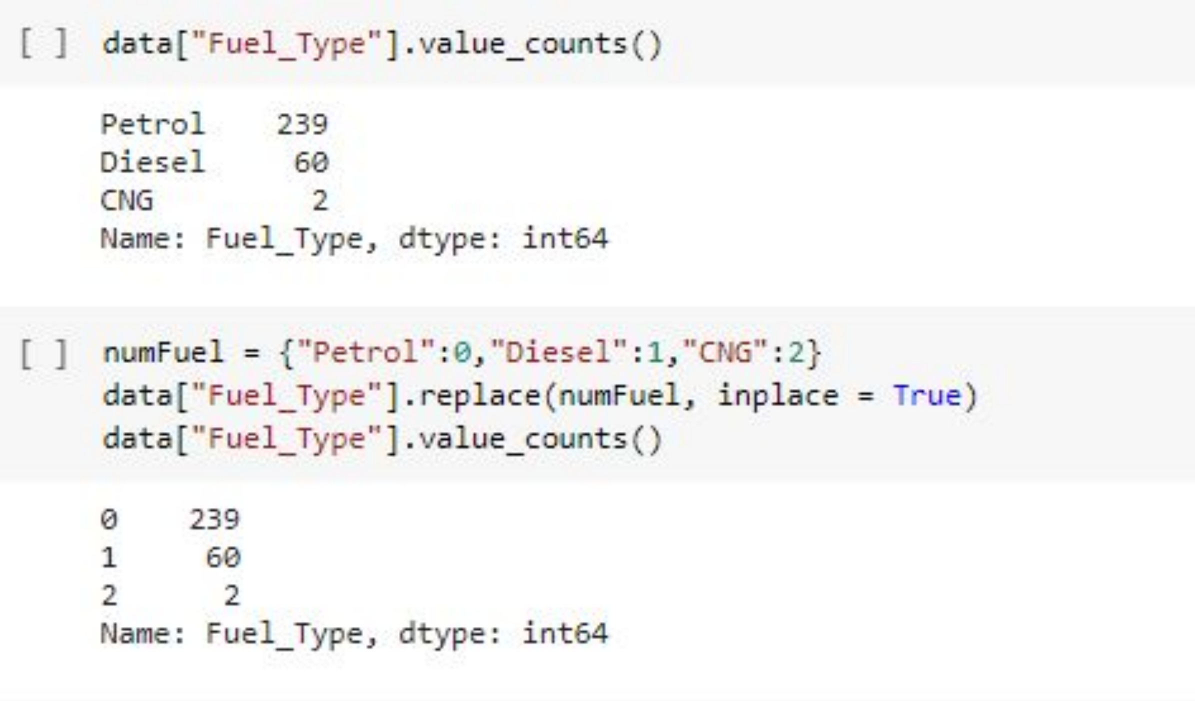
```
[ ]  data["Kms_Driven"].describe()
```

```
count       301.000000
mean      36947.205980
std       38886.883882
min         500.000000
25%       15000.000000
50%       32000.000000
75%       48767.000000
max      500000.000000
Name: Kms_Driven, dtype: float64
```

```
[ ]  data.loc[(data['Kms_Driven'] >= 500) & (data['Kms_Driven'] <= 15000), 'Kms_Driven'] = 0
     data.loc[(data['Kms_Driven'] > 15000) & (data['Kms_Driven'] <= 32000), 'Kms_Driven'] = 1
     data.loc[(data['Kms_Driven'] > 32000) & (data['Kms_Driven'] <= 49000), 'Kms_Driven'] = 2
     data.loc[(data['Kms_Driven'] > 49000), 'Kms_Driven'] = 3
     data["Kms_Driven"].value_counts()
```

```
0    77
2    76
1    75
3    73
Name: Kms_Driven, dtype: int64
```

FUEL TYPE



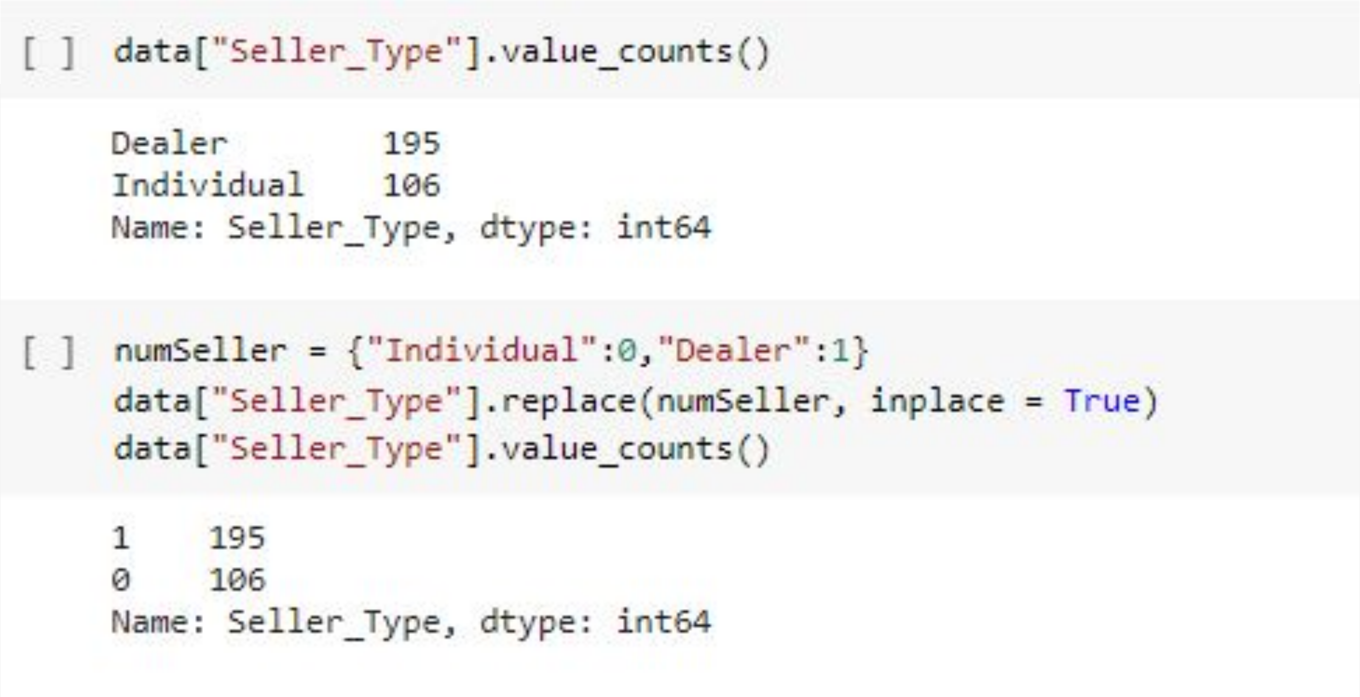
```
[ ]  data["Fuel_Type"].value_counts()
```

```
Petrol    239
Diesel     60
CNG         2
Name: Fuel_Type, dtype: int64
```

```
[ ]  numFuel = {"Petrol":0,"Diesel":1,"CNG":2}
     data["Fuel_Type"].replace(numFuel, inplace = True)
     data["Fuel_Type"].value_counts()
```

```
0    239
1     60
2      2
Name: Fuel_Type, dtype: int64
```

SELLER TYPE



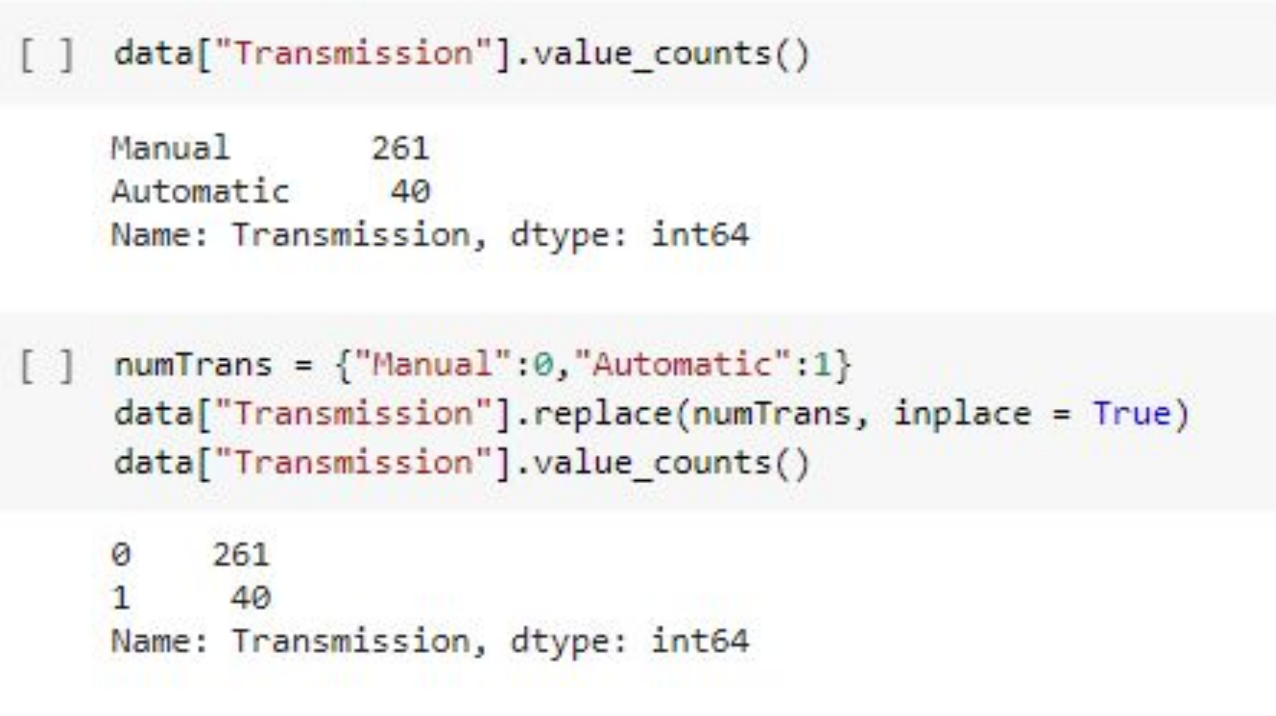
```
[ ]  data["Seller_Type"].value_counts()
```

```
Dealer        195
Individual    106
Name: Seller_Type, dtype: int64
```

```
[ ]  numSeller = {"Individual":0,"Dealer":1}
     data["Seller_Type"].replace(numSeller, inplace = True)
     data["Seller_Type"].value_counts()
```

```
1    195
0    106
Name: Seller_Type, dtype: int64
```

TRANSMISSION



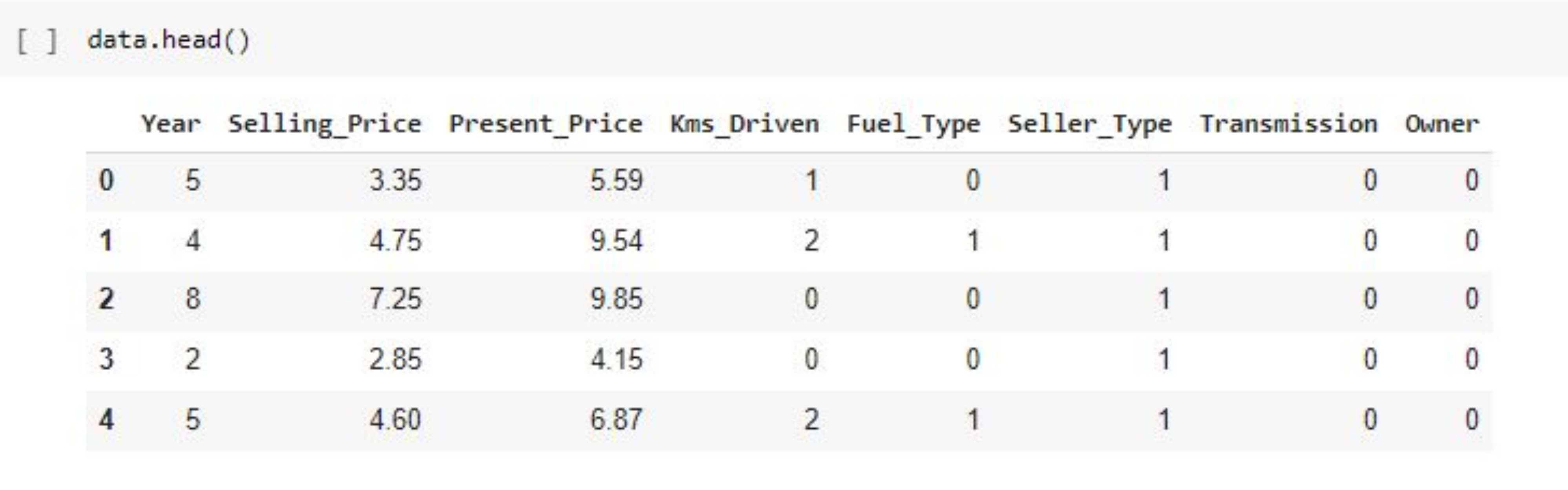
```
[ ]  data["Transmission"].value_counts()
```

```
Manual       261
Automatic     40
Name: Transmission, dtype: int64
```

```
[ ]  numTrans = {"Manual":0,"Automatic":1}
     data["Transmission"].replace(numTrans, inplace = True)
     data["Transmission"].value_counts()
```

```
0    261
1     40
Name: Transmission, dtype: int64
```

DATA



```
[ ] data.head()
```

	Year	Selling_Price	Present_Price	Kms_Driven	Fuel_Type	Seller_Type	Transmission	Owner
0	5	3.35	5.59	1	0	1	0	0
1	4	4.75	9.54	2	1	1	0	0
2	8	7.25	9.85	0	0	1	0	0
3	2	2.85	4.15	0	0	1	0	0
4	5	4.60	6.87	2	1	1	0	0

3

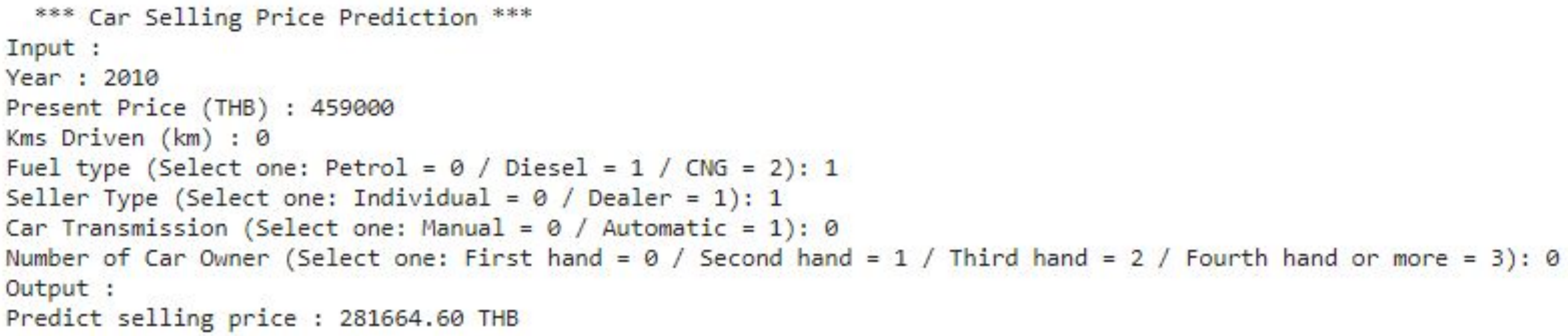
วิเคราะห์ข้อมูล

Correlations

Linear regression

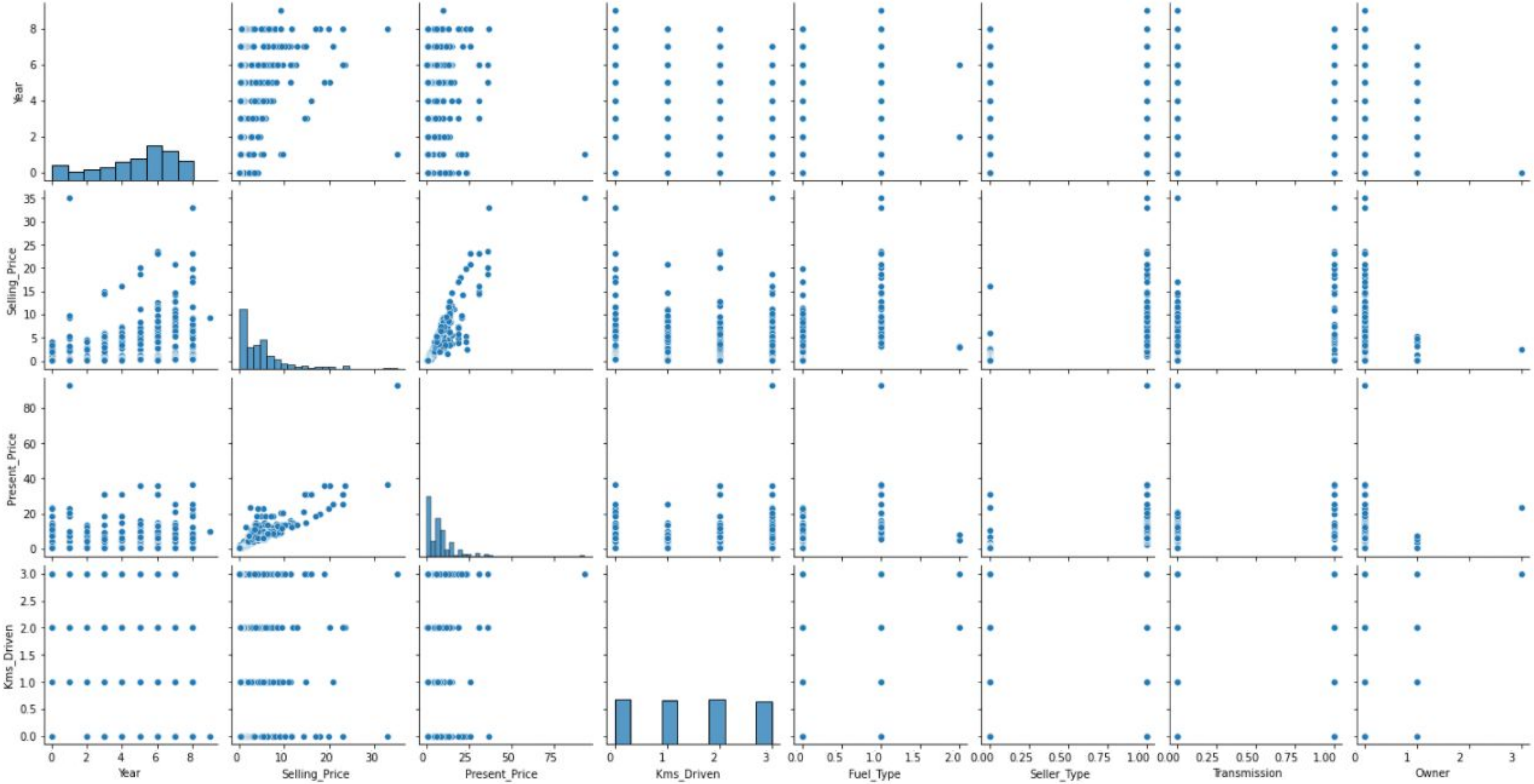
4

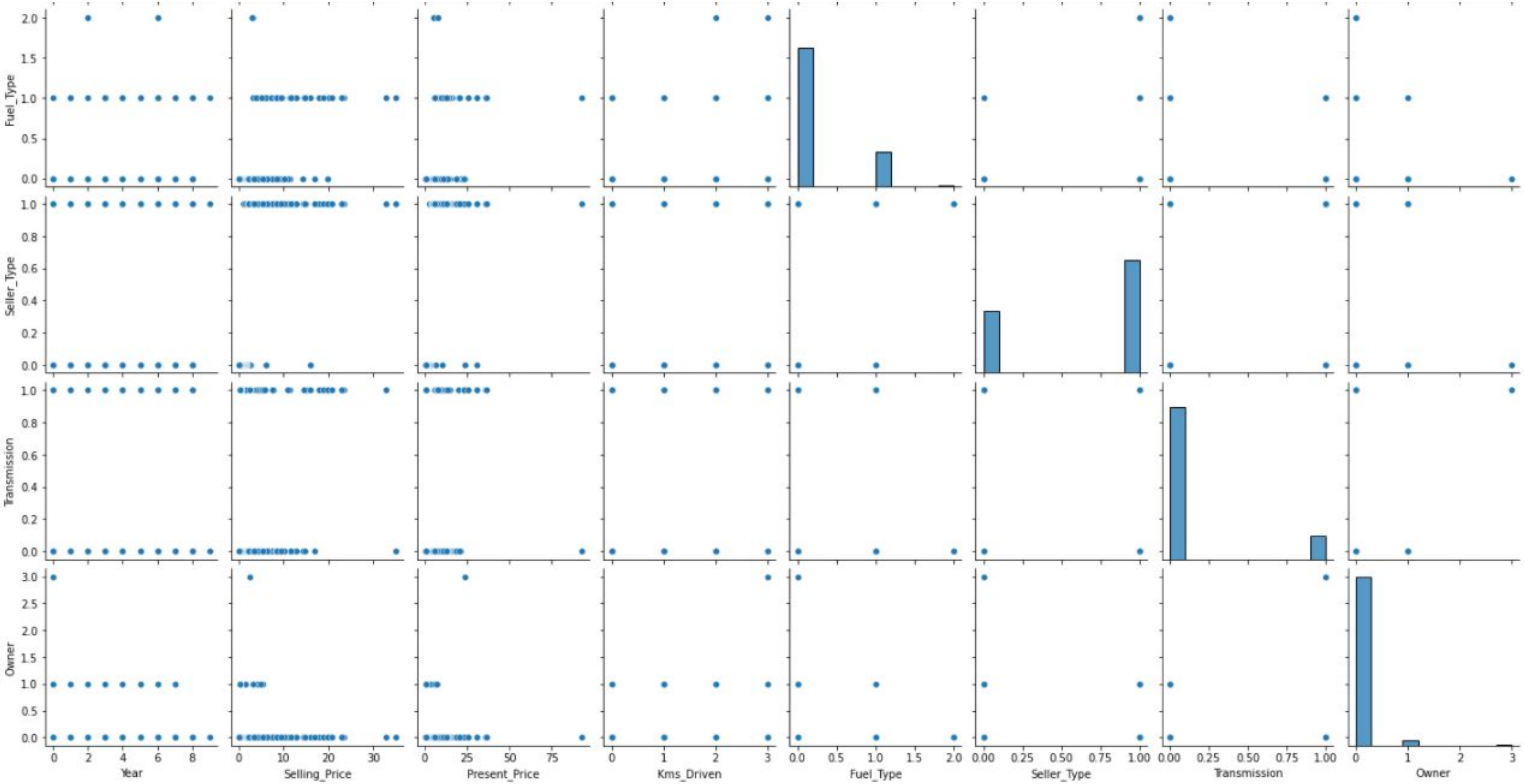
MODEL

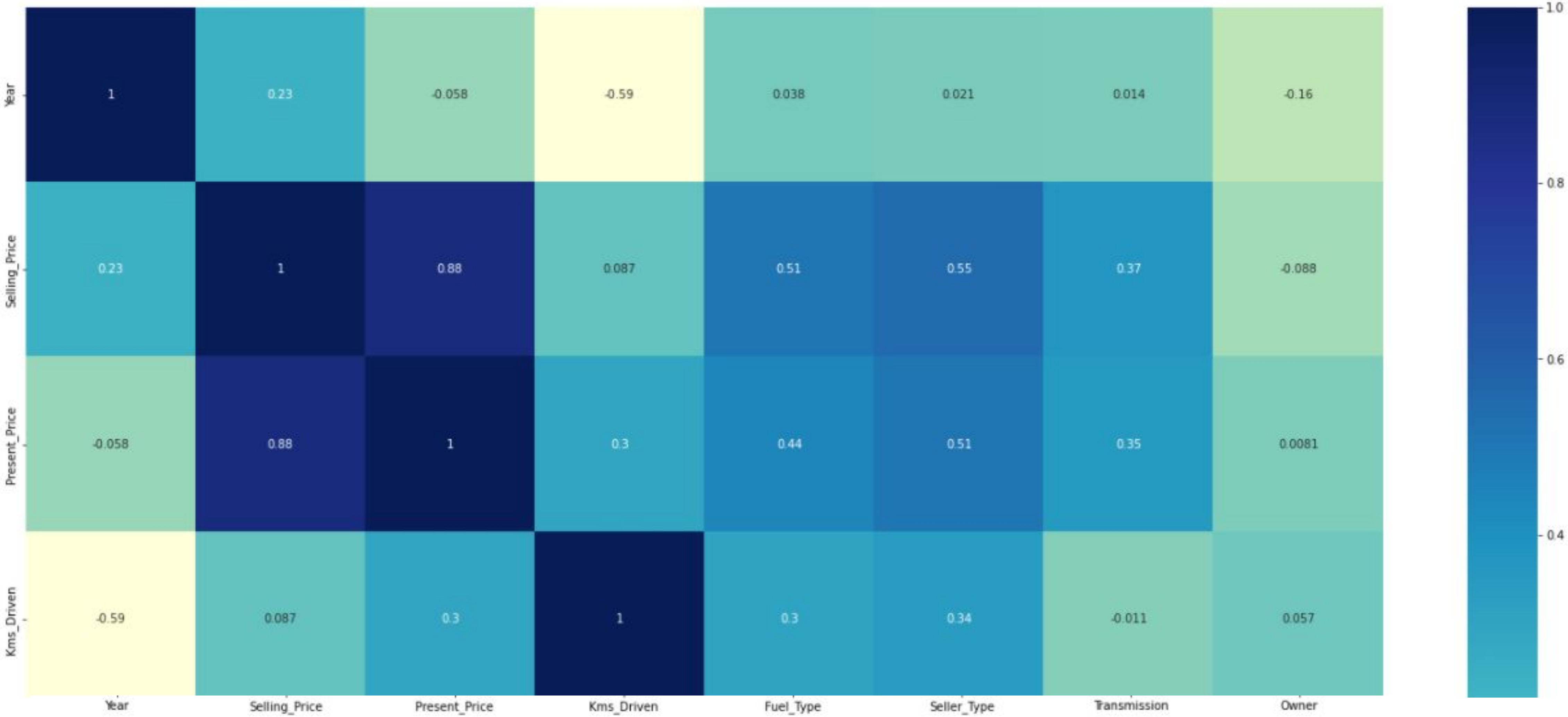


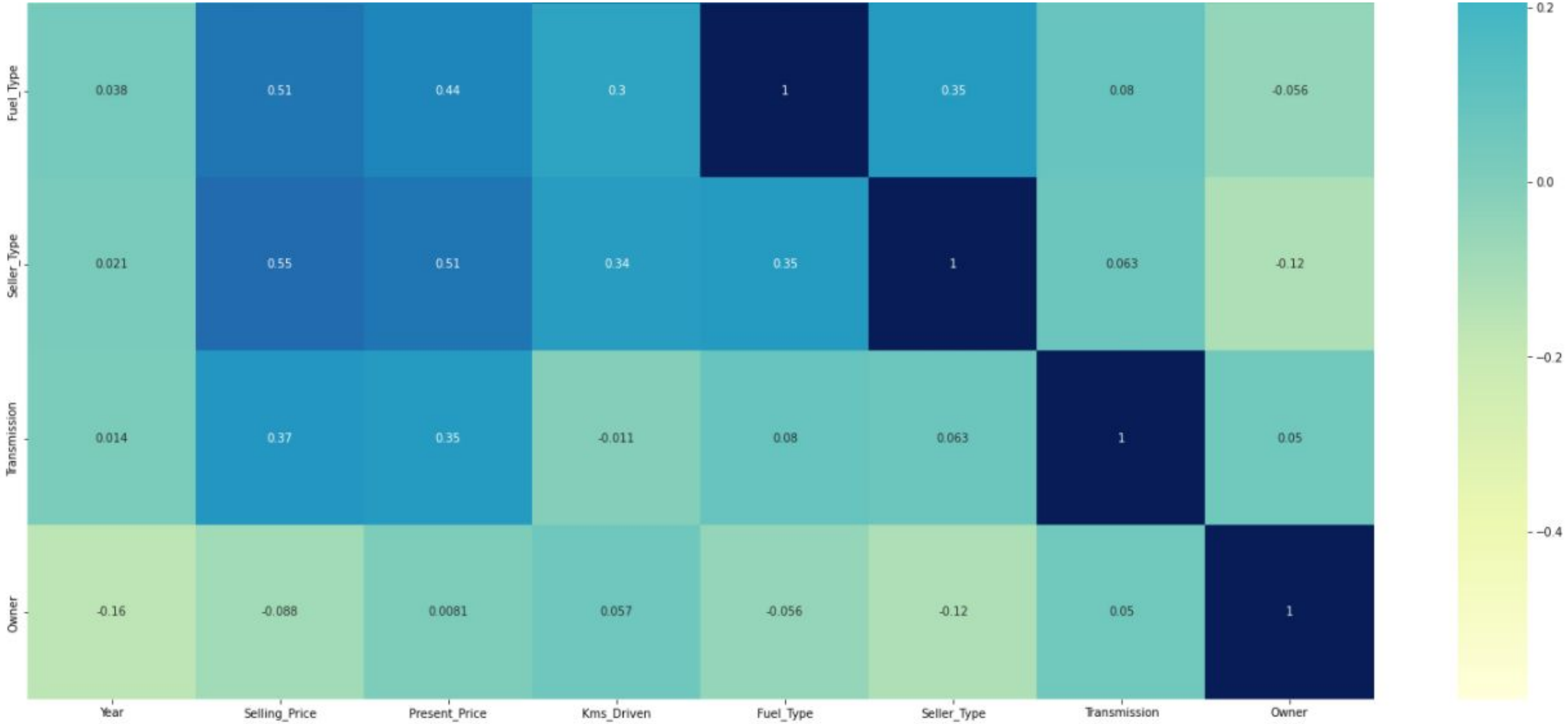
```
  *** Car Selling Price Prediction ***
Input :
Year : 2010
Present Price (THB) : 459000
Kms Driven (km) : 0
Fuel type (Select one: Petrol = 0 / Diesel = 1 / CNG = 2): 1
Seller Type (Select one: Individual = 0 / Dealer = 1): 1
Car Transmission (Select one: Manual = 0 / Automatic = 1): 0
Number of Car Owner (Select one: First hand = 0 / Second hand = 1 / Third hand = 2 / Fourth hand or more = 3): 0
Output :
Predict selling price : 281664.60 THB
```

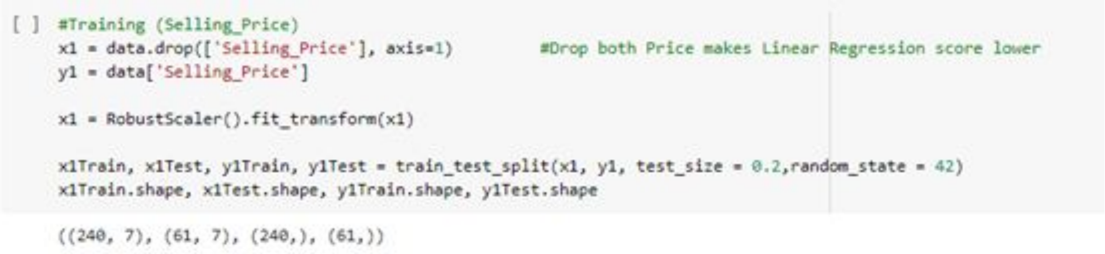
การประมวลผล


Year
8
6
4
2
0
Selling_Price
35
30
25
20
15
10
5
0
Present_Price
80
60
40
20
0
Kms_Driven
3.0
2.5
2.0
1.5
1.0
0.5
0.0
0
2
4
6
8
Year
0
10
20
30
Selling_Price
0
25
50
75
Present_Price
0
1
2
3
Kms_Driven
0.0
0.5
1.0
1.5
2.0
Fuel_Type
0.00
0.25
0.50
0.75
1.00
Seller_Type
0.00
0.25
0.50
0.75
1.00
Transmission
0
1
2
3
Owner


Fuel_Type
2.0
1.5
1.0
0.5
0.0
Seller_Type
1.0
0.8
0.6
0.4
0.2
0.0
Transmission
1.0
0.8
0.6
0.4
0.2
0.0
Owner
3.0
2.5
2.0
1.5
1.0
0.5
0.0
0 2 4 6 8
Year
0 10 20 30
Selling_Price
0 25 50 75
Present_Price
0 1 2 3
Kms_Driven
0.0 0.5 1.0 1.5 2.0
Fuel_Type
0.00 0.25 0.50 0.75 1.00
Seller_Type
0.00 0.25 0.50 0.75 1.00
Transmission
0 1 2 3
Owner

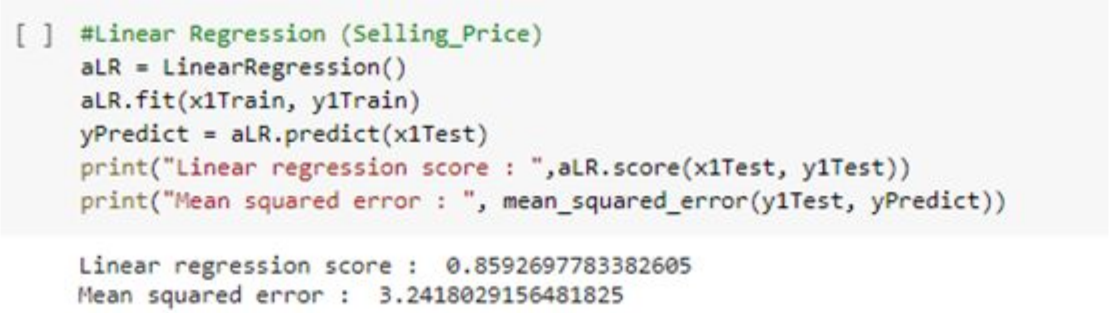

Year
Selling_Price
Present_Price
Kms_Driven
1
0.23
-0.058
-0.59
0.038
0.021
0.014
-0.16
0.23
1
0.88
0.087
0.51
0.55
0.37
-0.088
-0.058
0.88
1
0.3
0.44
0.51
0.35
0.0081
-0.59
0.087
0.3
1
0.3
0.34
-0.011
0.057
Year
Selling_Price
Present_Price
Kms_Driven
Fuel_Type
Seller_Type
Transmission
Owner
1.0
0.8
0.6
0.4


Fuel_Type
0.038
0.51
0.44
0.3
1
0.35
0.08
-0.056
Seller_Type
0.021
0.55
0.51
0.34
0.35
1
0.063
-0.12
Transmission
0.014
0.37
0.35
-0.011
0.08
0.063
1
0.05
Owner
-0.16
-0.088
0.0081
0.057
-0.056
-0.12
0.05
1
Year
Selling_Price
Present_Price
Kms_Driven
Fuel_Type
Seller_Type
Transmission
Owner
0.2
0.0
−0.2
−0.4



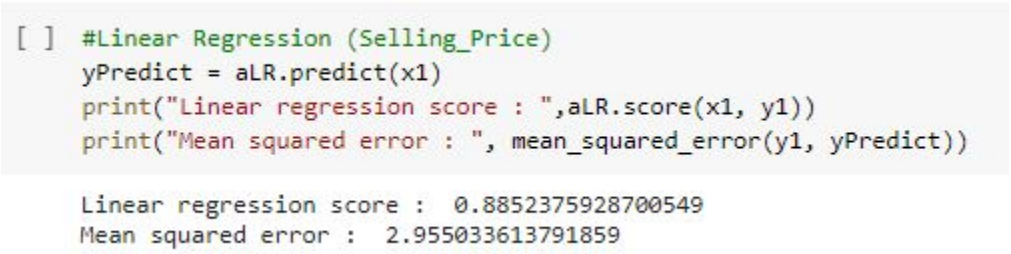
```
[ ]  #Training (Selling_Price)
     x1 = data.drop(['Selling_Price'], axis=1)        #Drop both Price makes Linear Regression score lower
     y1 = data['Selling_Price']

     x1 = RobustScaler().fit_transform(x1)

     x1Train, x1Test, y1Train, y1Test = train_test_split(x1, y1, test_size = 0.2,random_state = 42)
     x1Train.shape, x1Test.shape, y1Train.shape, y1Test.shape

     ((240, 7), (61, 7), (240,), (61,))
```

นำชุดข้อมูลที่ผ่านการสุ่มมาทำการ Train โมเดล



```
[ ]  #Linear Regression (Selling_Price)
     aLR = LinearRegression()
     aLR.fit(x1Train, y1Train)
     yPredict = aLR.predict(x1Test)
     print("Linear regression score : ",aLR.score(x1Test, y1Test))
     print("Mean squared error : ", mean_squared_error(y1Test, yPredict))

     Linear regression score :  0.8592697783382605
     Mean squared error :  3.2418029156481825
```

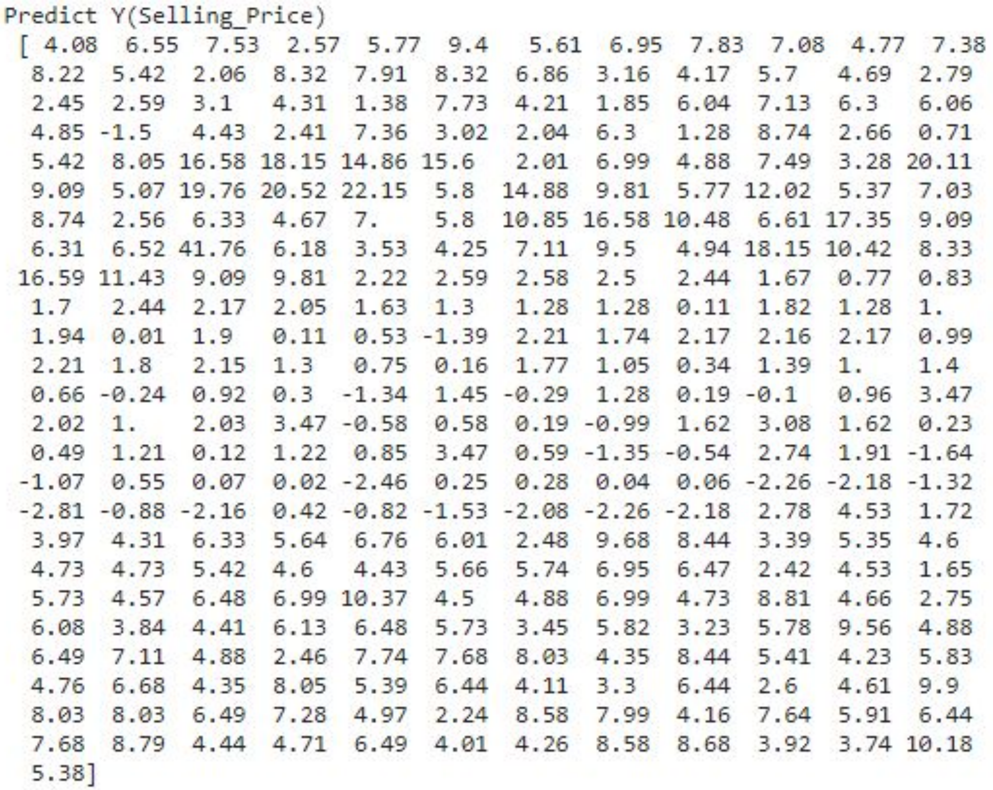
นำโมเดลที่ผ่านการ Train มาทำนายชุดข้อมูลที่เหลือ



```
#Linear Regression (Selling_Price)
yPredict = aLR.predict(x1)
print("Linear regression score : ",aLR.score(x1, y1))
print("Mean squared error : ", mean_squared_error(y1, yPredict))
```

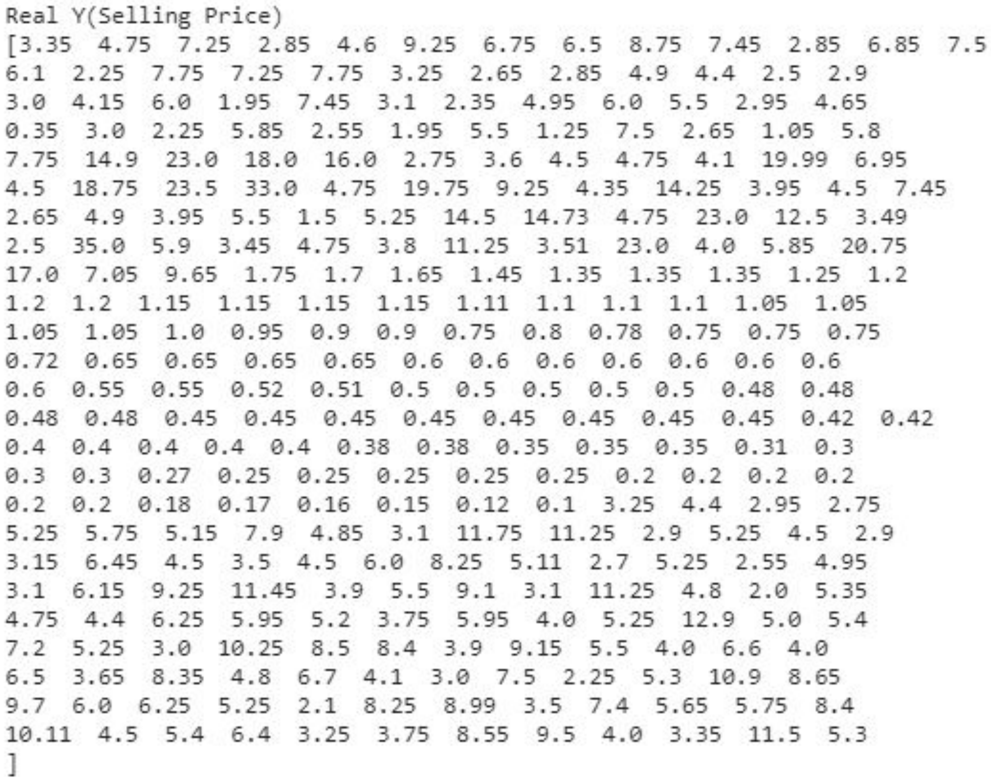
```
Linear regression score :  0.8852375928700549
Mean squared error :  2.955033613791859
```

นำชุดข้อมูลทั้งหมดมาทำนายราคารถยนต์



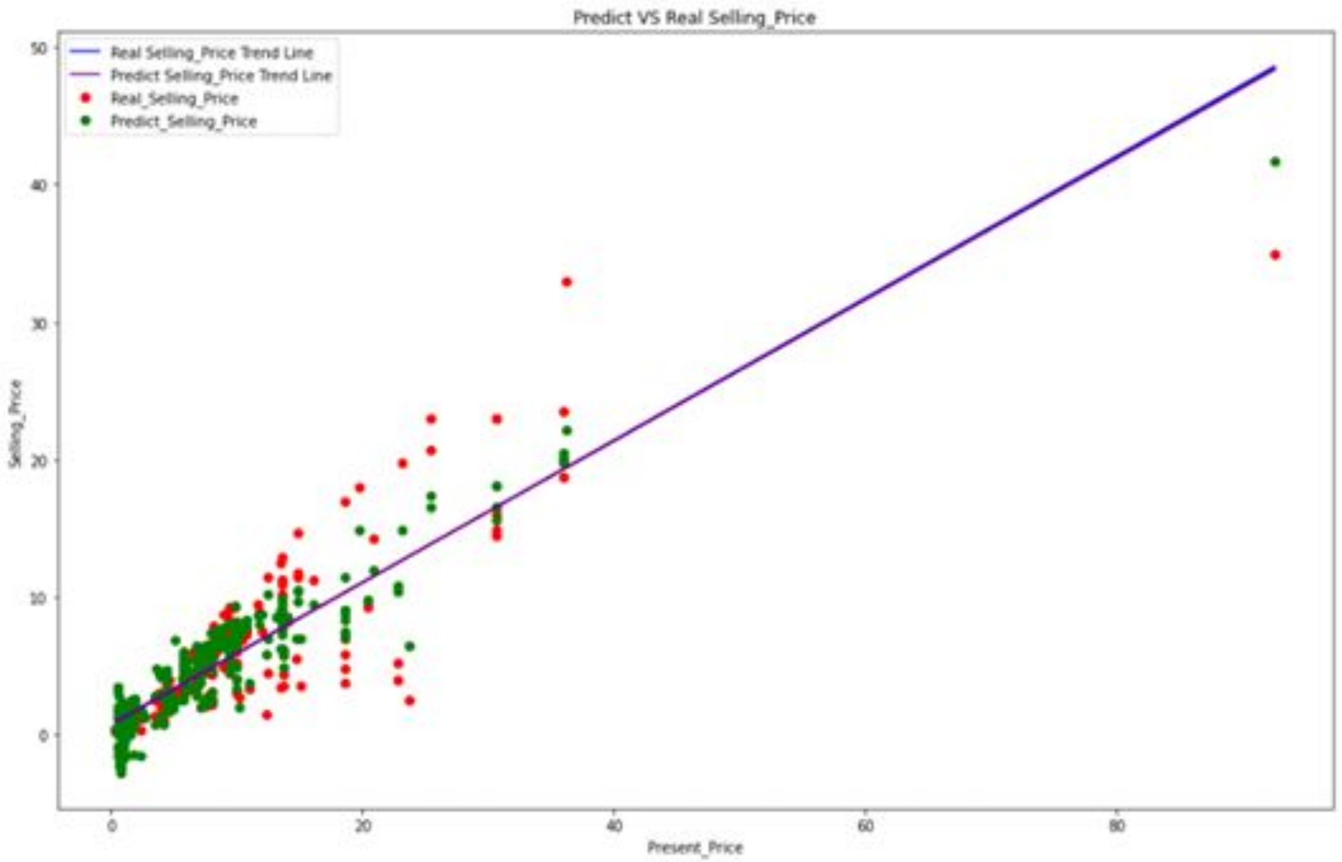
```
Predict Y(Selling_Price)
 [ 4.08  6.55  7.53  2.57  5.77  9.4   5.61  6.95  7.83  7.08  4.77  7.38
  8.22  5.42  2.06  8.32  7.91  8.32  6.86  3.16  4.17  5.7   4.69  2.79
  2.45  2.59  3.1   4.31  1.38  7.73  4.21  1.85  6.04  7.13  6.3   6.06
  4.85 -1.5   4.43  2.41  7.36  3.02  2.04  6.3   1.28  8.74  2.66  0.71
  5.42  8.05 16.58 18.15 14.86 15.6   2.01  6.99  4.88  7.49  3.28 20.11
  9.09  5.07 19.76 20.52 22.15  5.8  14.88  9.81  5.77 12.02  5.37  7.03
  8.74  2.56  6.33  4.67  7.    5.8  10.85 16.58 10.48  6.61 17.35  9.09
  6.31  6.52 41.76  6.18  3.53  4.25  7.11  9.5   4.94 18.15 10.42  8.33
 16.59 11.43  9.09  9.81  2.22  2.59  2.58  2.5   2.44  1.67  0.77  0.83
  1.7   2.44  2.17  2.05  1.63  1.3   1.28  1.28  0.11  1.82  1.28  1.
  1.94  0.01  1.9   0.11  0.53 -1.39  2.21  1.74  2.17  2.16  2.17  0.99
  2.21  1.8   2.15  1.3   0.75  0.16  1.77  1.05  0.34  1.39  1.    1.4
  0.66 -0.24  0.92  0.3  -1.34  1.45 -0.29  1.28  0.19 -0.1   0.96  3.47
  2.02  1.    2.03  3.47 -0.58  0.58  0.19 -0.99  1.62  3.08  1.62  0.23
  0.49  1.21  0.12  1.22  0.85  3.47  0.59 -1.35 -0.54  2.74  1.91 -1.64
 -1.07  0.55  0.07  0.02 -2.46  0.25  0.28  0.04  0.06 -2.26 -2.18 -1.32
 -2.81 -0.88 -2.16  0.42 -0.82 -1.53 -2.08 -2.26 -2.18  2.78  4.53  1.72
  3.97  4.31  6.33  5.64  6.76  6.01  2.48  9.68  8.44  3.39  5.35  4.6
  4.73  4.73  5.42  4.6   4.43  5.66  5.74  6.95  6.47  2.42  4.53  1.65
  5.73  4.57  6.48  6.99 10.37  4.5   4.88  6.99  4.73  8.81  4.66  2.75
  6.08  3.84  4.41  6.13  6.48  5.73  3.45  5.82  3.23  5.78  9.56  4.88
  6.49  7.11  4.88  2.46  7.74  7.68  8.03  4.35  8.44  5.41  4.23  5.83
  4.76  6.68  4.35  8.05  5.39  6.44  4.11  3.3   6.44  2.6   4.61  9.9
  8.03  8.03  6.49  7.28  4.97  2.24  8.58  7.99  4.16  7.64  5.91  6.44
  7.68  8.79  4.44  4.71  6.49  4.01  4.26  8.58  8.68  3.92  3.74 10.18
  5.38]
```

ค่าที่ได้จากการทำนาย

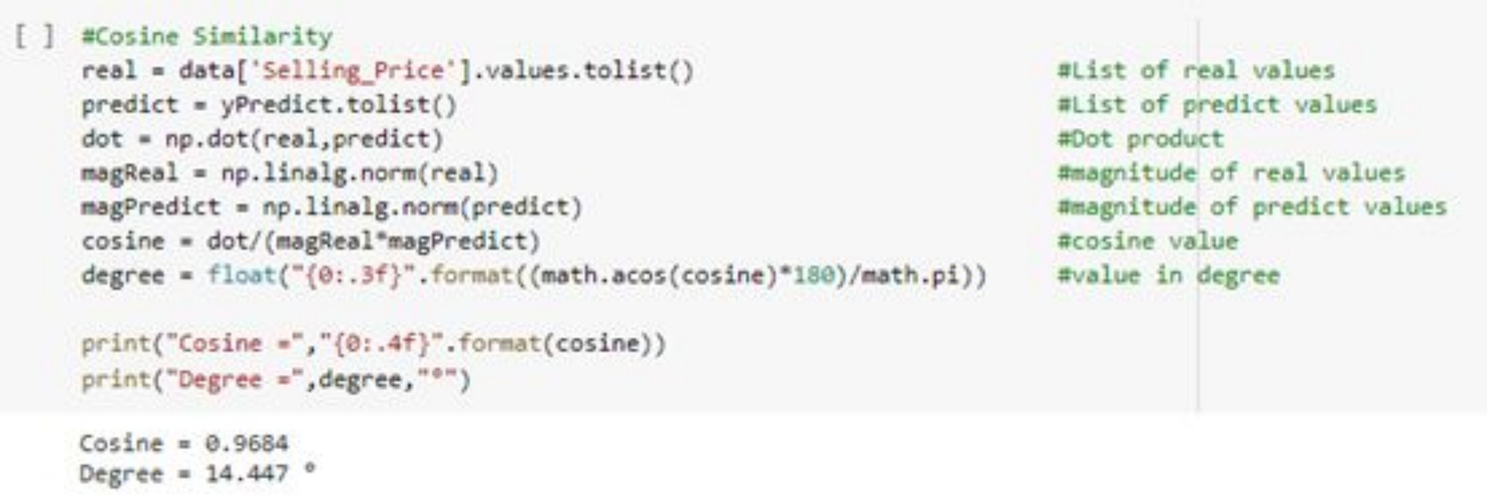


```
Real Y(Selling Price)
[3.35  4.75  7.25  2.85  4.6  9.25  6.75  6.5  8.75  7.45  2.85  6.85  7.5
6.1  2.25  7.75  7.25  7.75  3.25  2.65  2.85  4.9  4.4  2.5  2.9
3.0  4.15  6.0  1.95  7.45  3.1  2.35  4.95  6.0  5.5  2.95  4.65
0.35  3.0  2.25  5.85  2.55  1.95  5.5  1.25  7.5  2.65  1.05  5.8
7.75  14.9  23.0  18.0  16.0  2.75  3.6  4.5  4.75  4.1  19.99  6.95
4.5  18.75  23.5  33.0  4.75  19.75  9.25  4.35  14.25  3.95  4.5  7.45
2.65  4.9  3.95  5.5  1.5  5.25  14.5  14.73  4.75  23.0  12.5  3.49
2.5  35.0  5.9  3.45  4.75  3.8  11.25  3.51  23.0  4.0  5.85  20.75
17.0  7.05  9.65  1.75  1.7  1.65  1.45  1.35  1.35  1.35  1.25  1.2
1.2  1.2  1.15  1.15  1.15  1.15  1.11  1.1  1.1  1.1  1.05  1.05
1.05  1.05  1.0  0.95  0.9  0.9  0.75  0.8  0.78  0.75  0.75  0.75
0.72  0.65  0.65  0.65  0.65  0.6  0.6  0.6  0.6  0.6  0.6  0.6
0.6  0.55  0.55  0.52  0.51  0.5  0.5  0.5  0.5  0.5  0.48  0.48
0.48  0.48  0.45  0.45  0.45  0.45  0.45  0.45  0.45  0.45  0.42  0.42
0.4  0.4  0.4  0.4  0.4  0.38  0.38  0.35  0.35  0.35  0.31  0.3
0.3  0.3  0.27  0.25  0.25  0.25  0.25  0.25  0.2  0.2  0.2  0.2
0.2  0.2  0.18  0.17  0.16  0.15  0.12  0.1  3.25  4.4  2.95  2.75
5.25  5.75  5.15  7.9  4.85  3.1  11.75  11.25  2.9  5.25  4.5  2.9
3.15  6.45  4.5  3.5  4.5  6.0  8.25  5.11  2.7  5.25  2.55  4.95
3.1  6.15  9.25  11.45  3.9  5.5  9.1  3.1  11.25  4.8  2.0  5.35
4.75  4.4  6.25  5.95  5.2  3.75  5.95  4.0  5.25  12.9  5.0  5.4
7.2  5.25  3.0  10.25  8.5  8.4  3.9  9.15  5.5  4.0  6.6  4.0
6.5  3.65  8.35  4.8  6.7  4.1  3.0  7.5  2.25  5.3  10.9  8.65
9.7  6.0  6.25  5.25  2.1  8.25  8.99  3.5  7.4  5.65  5.75  8.4
10.11  4.5  5.4  6.4  3.25  3.75  8.55  9.5  4.0  3.35  11.5  5.3
]
```

ค่าจริง



กราฟเปรียบเทียบค่าที่ได้จาก Model การทำนาย และ ค่าจริง

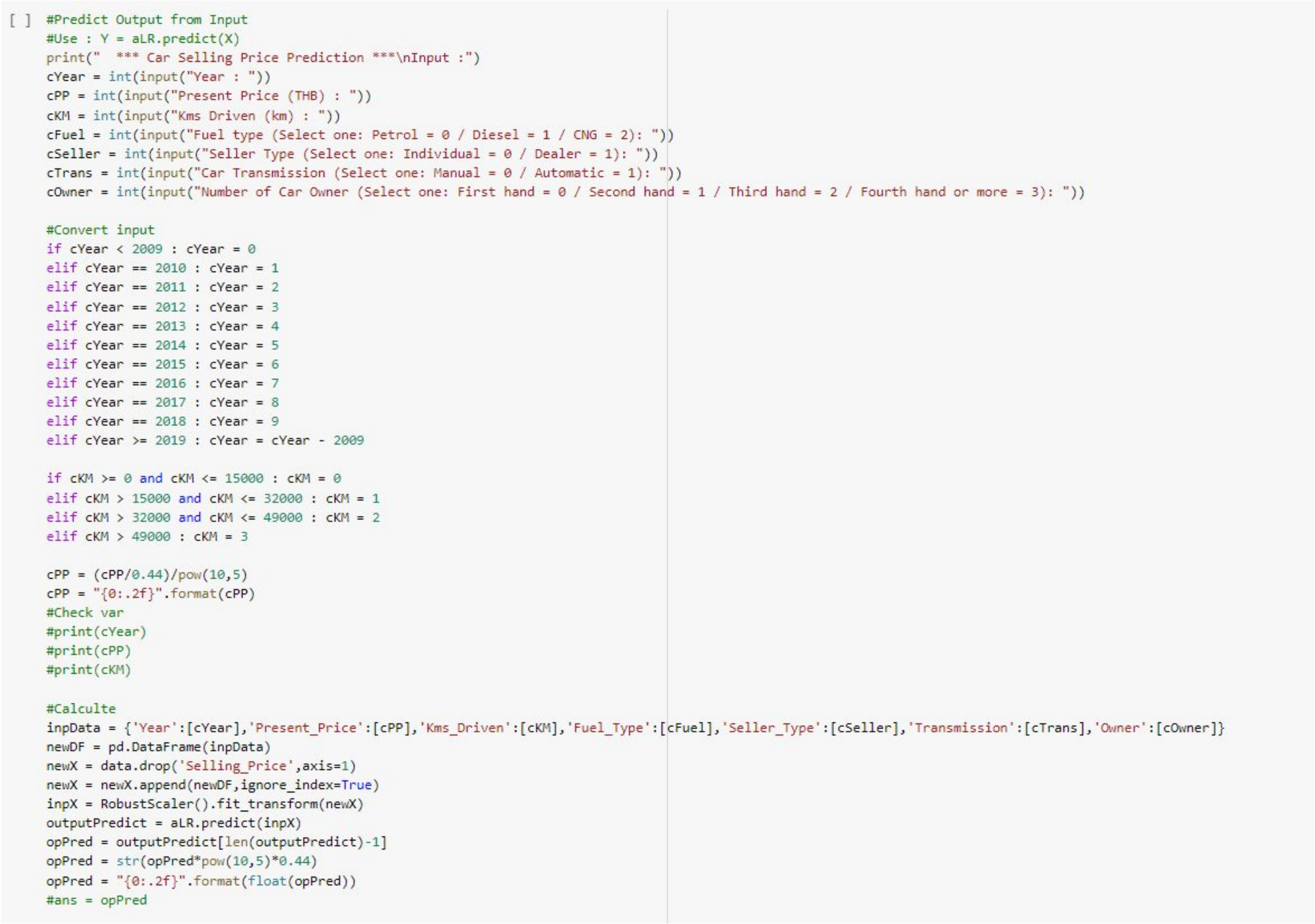


```
#Cosine Similarity
real = data['Selling_Price'].values.tolist()                           #List of real values
predict = yPredict.tolist()                                            #List of predict values
dot = np.dot(real,predict)                                             #Dot product
magReal = np.linalg.norm(real)                                         #magnitude of real values
magPredict = np.linalg.norm(predict)                                   #magnitude of predict values
cosine = dot/(magReal*magPredict)                                      #cosine value
degree = float("{0:.3f}".format((math.acos(cosine)*180)/math.pi))     #value in degree

print("Cosine =","{0:.4f}".format(cosine))
print("Degree =",degree,"°")
```

```
Cosine = 0.9684
Degree = 14.447 °
```

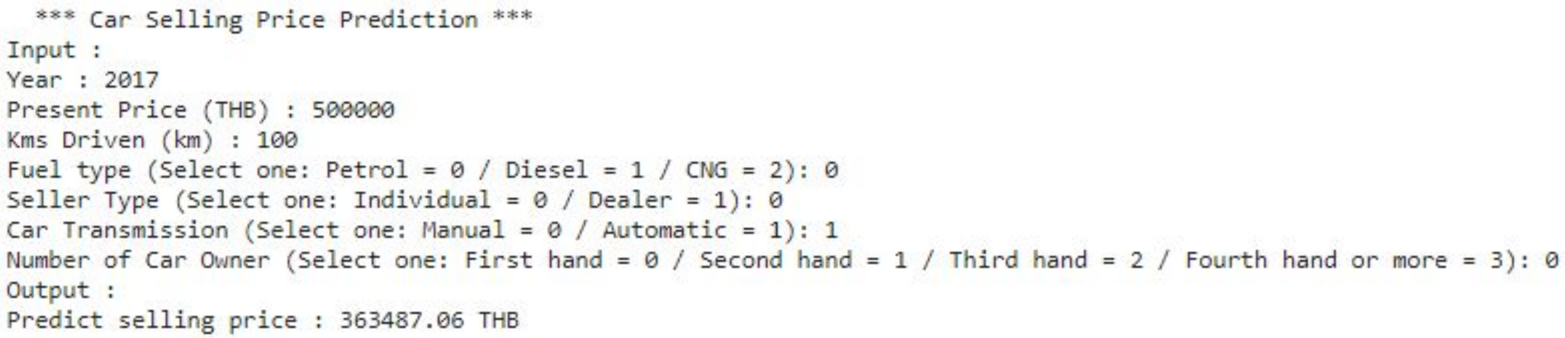
หาค่าของ Cosine Similarity เพื่อวัดประสิทธิภาพของ Model การทำนาย

ผลการทดสอบ

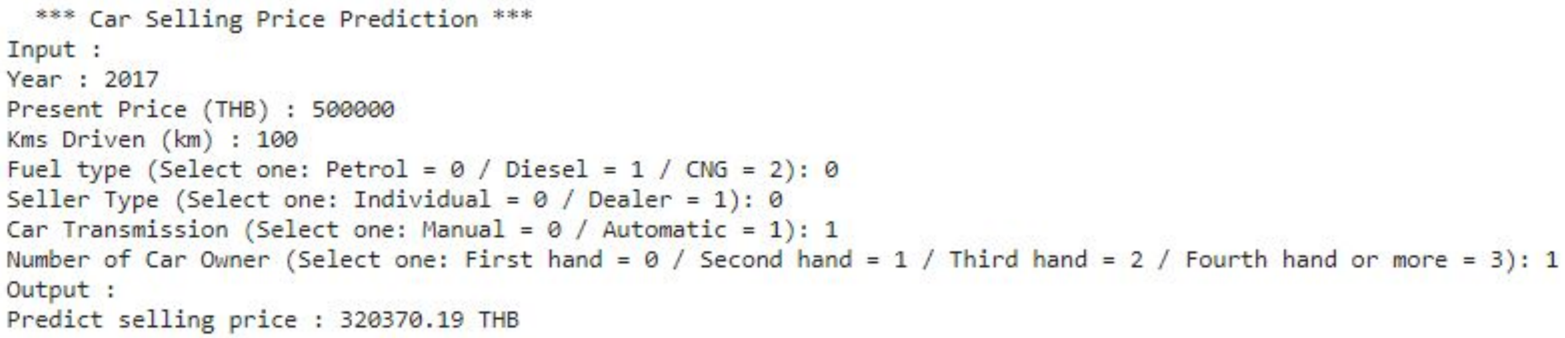


```
[ ] #Predict Output from Input
    #Use : Y = aLR.predict(X)
    print("  *** Car Selling Price Prediction ***\nInput :")
    cYear = int(input("Year : "))
    cPP = int(input("Present Price (THB) : "))
    cKM = int(input("Kms Driven (km) : "))
    cFuel = int(input("Fuel type (Select one: Petrol = 0 / Diesel = 1 / CNG = 2): "))
    cSeller = int(input("Seller Type (Select one: Individual = 0 / Dealer = 1): "))
    cTrans = int(input("Car Transmission (Select one: Manual = 0 / Automatic = 1): "))
    cOwner = int(input("Number of Car Owner (Select one: First hand = 0 / Second hand = 1 / Third hand = 2 / Fourth hand or more = 3): "))

    #Convert input
    if cYear < 2009 : cYear = 0
    elif cYear == 2010 : cYear = 1
    elif cYear == 2011 : cYear = 2
    elif cYear == 2012 : cYear = 3
    elif cYear == 2013 : cYear = 4
    elif cYear == 2014 : cYear = 5
    elif cYear == 2015 : cYear = 6
    elif cYear == 2016 : cYear = 7
    elif cYear == 2017 : cYear = 8
    elif cYear == 2018 : cYear = 9
    elif cYear >= 2019 : cYear = cYear - 2009

    if cKM >= 0 and cKM <= 15000 : cKM = 0
    elif cKM > 15000 and cKM <= 32000 : cKM = 1
    elif cKM > 32000 and cKM <= 49000 : cKM = 2
    elif cKM > 49000 : cKM = 3

    cPP = (cPP/0.44)/pow(10,5)
    cPP = "{0:.2f}".format(cPP)
    #Check var
    #print(cYear)
    #print(cPP)
    #print(cKM)

    #Calculte
    inpData = {'Year':[cYear],'Present_Price':[cPP],'Kms_Driven':[cKM],'Fuel_Type':[cFuel],'Seller_Type':[cSeller],'Transmission':[cTrans],'Owner':[cOwner]}
    newDF = pd.DataFrame(inpData)
    newX = data.drop('Selling_Price',axis=1)
    newX = newX.append(newDF,ignore_index=True)
    inpX = RobustScaler().fit_transform(newX)
    outputPredict = aLR.predict(inpX)
    opPred = outputPredict[len(outputPredict)-1]
    opPred = str(opPred*pow(10,5)*0.44)
    opPred = "{0:.2f}".format(float(opPred))
    #ans = opPred
```

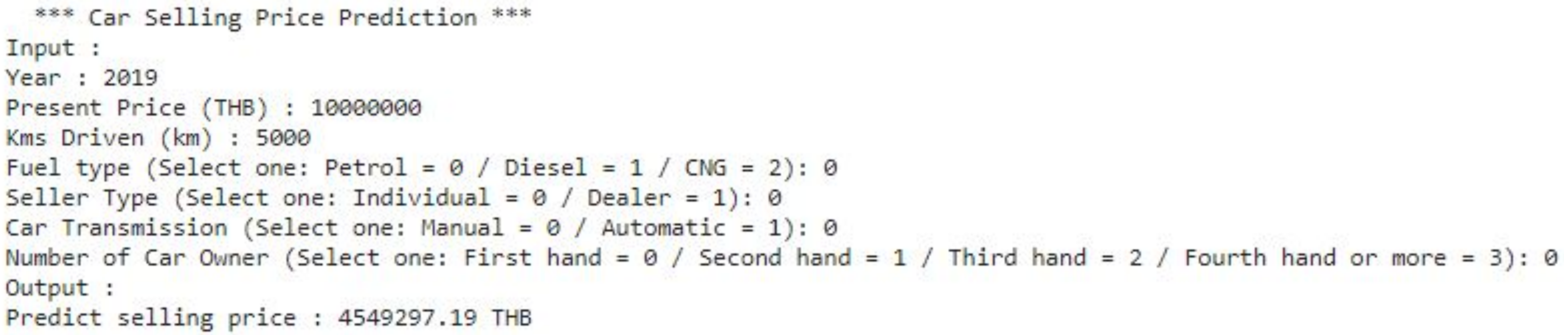
การรับข้อมูลจากผู้ใช้งานเข้ามาสู่ Model การทำนายและแสดงค่าการทำนาย



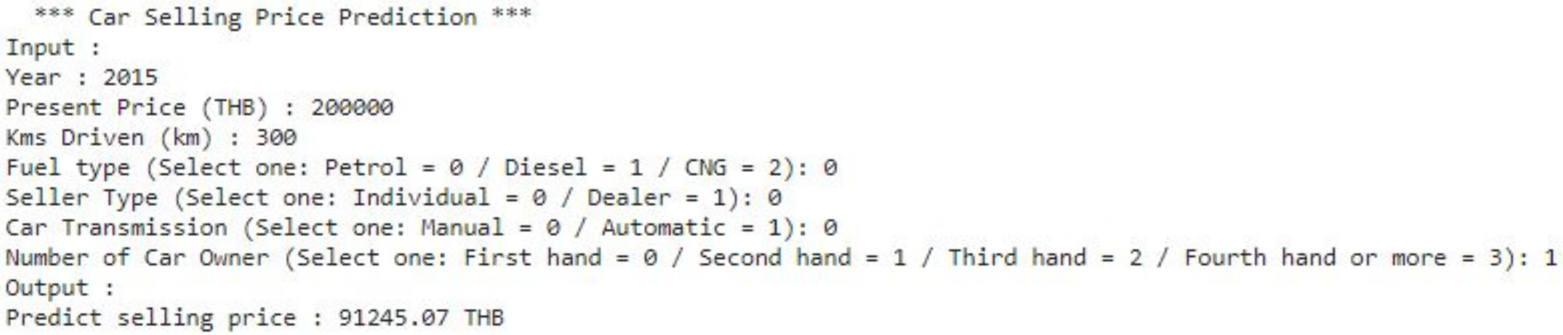
```
  *** Car Selling Price Prediction ***
Input :
Year : 2017
Present Price (THB) : 500000
Kms Driven (km) : 100
Fuel type (Select one: Petrol = 0 / Diesel = 1 / CNG = 2): 0
Seller Type (Select one: Individual = 0 / Dealer = 1): 0
Car Transmission (Select one: Manual = 0 / Automatic = 1): 1
Number of Car Owner (Select one: First hand = 0 / Second hand = 1 / Third hand = 2 / Fourth hand or more = 3): 0
Output :
Predict selling price : 363487.06 THB
```



```
  *** Car Selling Price Prediction ***
Input :
Year : 2017
Present Price (THB) : 500000
Kms Driven (km) : 100
Fuel type (Select one: Petrol = 0 / Diesel = 1 / CNG = 2): 0
Seller Type (Select one: Individual = 0 / Dealer = 1): 0
Car Transmission (Select one: Manual = 0 / Automatic = 1): 1
Number of Car Owner (Select one: First hand = 0 / Second hand = 1 / Third hand = 2 / Fourth hand or more = 3): 1
Output :
Predict selling price : 320370.19 THB
```



```
 *** Car Selling Price Prediction ***
Input :
Year : 2019
Present Price (THB) : 10000000
Kms Driven (km) : 5000
Fuel type (Select one: Petrol = 0 / Diesel = 1 / CNG = 2): 0
Seller Type (Select one: Individual = 0 / Dealer = 1): 0
Car Transmission (Select one: Manual = 0 / Automatic = 1): 0
Number of Car Owner (Select one: First hand = 0 / Second hand = 1 / Third hand = 2 / Fourth hand or more = 3): 0
Output :
Predict selling price : 4549297.19 THB
```



```
 *** Car Selling Price Prediction ***
Input :
Year : 2015
Present Price (THB) : 200000
Kms Driven (km) : 300
Fuel type (Select one: Petrol = 0 / Diesel = 1 / CNG = 2): 0
Seller Type (Select one: Individual = 0 / Dealer = 1): 0
Car Transmission (Select one: Manual = 0 / Automatic = 1): 0
Number of Car Owner (Select one: First hand = 0 / Second hand = 1 / Third hand = 2 / Fourth hand or more = 3): 1
Output :
Predict selling price : 91245.07 THB
```

THANKS!