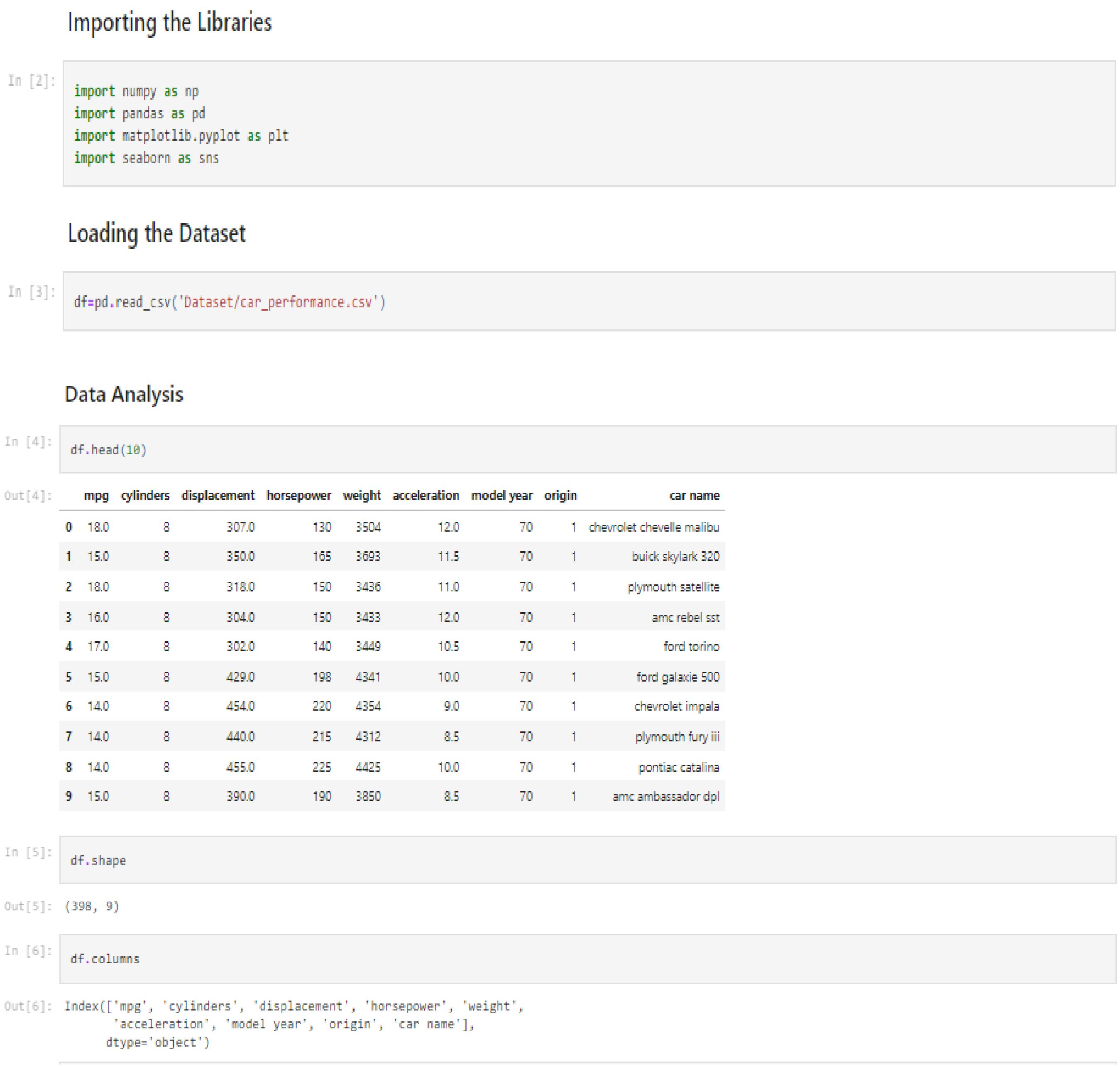
Data Preprocessing

Date	17 November 2022
Team ID	PNT2022TMID45545
Project name	MACHINE LEARNING BASED VEHICLE PERFORMANCE ANALAYZER



Importing the Libraries

In [2]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

Loading the Dataset

In [3]:

```
df=pd.read_csv('Dataset/car_performance.csv')
```

Data Analysis

In [4]:

```
df.head(10)
```

Out[4]:

	mpg	cylinders	displacement	horsepower	weight	acceleration	model year	origin	car name
0	18.0	8	307.0	130	3504	12.0	70	1	chevrolet chevelle malibu
1	15.0	8	350.0	165	3693	11.5	70	1	buick skylark 320
2	18.0	8	318.0	150	3436	11.0	70	1	plymouth satellite
3	16.0	8	304.0	150	3433	12.0	70	1	amc rebel sst
4	17.0	8	302.0	140	3449	10.5	70	1	ford torino
5	15.0	8	429.0	198	4341	10.0	70	1	ford galaxie 500
6	14.0	8	454.0	220	4354	9.0	70	1	chevrolet impala
7	14.0	8	440.0	215	4312	8.5	70	1	plymouth fury iii
8	14.0	8	455.0	225	4425	10.0	70	1	pontiac catalina
9	15.0	8	390.0	190	3850	8.5	70	1	amc ambassador dpl

In [5]:

```
df.shape
```

Out[5]:

```
(398, 9)
```

In [6]:

```
df.columns
```

Out[6]:

```
Index(['mpg', 'cylinders', 'displacement', 'horsepower', 'weight',
       'acceleration', 'model year', 'origin', 'car name'],
      dtype='object')
```

In [7]:

```
df.info()
```

```
RangeIndex: 398 entries, 0 to 397
Data columns (total 9 columns):
 #   Column        Non-Null Count  Dtype  
---  ------        --------------  -----  
 0   mpg           398 non-null    float64
 1   cylinders     398 non-null    int64  
 2   displacement  398 non-null    float64
 3   horsepower    398 non-null    int64  
 4   weight        398 non-null    int64  
 5   acceleration  398 non-null    float64
 6   model year    398 non-null    int64  
 7   origin        398 non-null    int64  
 8   car name      398 non-null    object 
dtypes: float64(3), int64(5), object(1)
memory usage: 28.1+ KB
```

In [11]:

```
df.nunique()
```

Out[11]:

```
mpg             129
cylinders         5
displacement     82
horsepower       93
weight          351
acceleration     95
model year       13
origin            3
car name        305
dtype: int64
```

In [13]:

```
df.origin.unique()
```

Out[13]:

```
array([1, 3, 2])
```

Handiling the Missing Values

In [15]:

```
df.isna().sum()
```

Out[15]:

```
mpg             0
cylinders       0
displacement    0
horsepower      0
weight          0
acceleration    0
model year      0
origin          0
car name        0
dtype: int64
```

In [16]:

```
# There is no Null Value in the data set
```

Lable encoding

In [17]:

```
# There is no Categorial value other than the car name (car name is not used for the performance predecting so we can drop the car name column), so we
```

Droping the car name column

In [18]:

```
df=df.iloc[:,:-1]
```

In [19]:

```
df.head()
```

Out[19]:

	mpg	cylinders	displacement	horsepower	weight	acceleration	model year	origin
0	18.0	8	307.0	130	3504	12.0	70	1
1	15.0	8	350.0	165	3693	11.5	70	1
2	18.0	8	318.0	150	3436	11.0	70	1
3	16.0	8	304.0	150	3433	12.0	70	1
4	17.0	8	302.0	140	3449	10.5	70	1

Splitting the dataset into dependent and independent Variable

In [20]:

```
x=df.iloc[:,1:]
```

In [21]:

```
y=df.iloc[:,0]
```

In [23]:

```
x.head()
```

Out[23]:

	cylinders	displacement	horsepower	weight	acceleration	model year	origin
0	8	307.0	130	3504	12.0	70	1
1	8	350.0	165	3693	11.5	70	1
2	8	318.0	150	3436	11.0	70	1
3	8	304.0	150	3433	12.0	70	1
4	8	302.0	140	3449	10.5	70	1

In [24]:

```
y.head()
```

Out[24]:

```
0    18.0
1    15.0
2    18.0
3    16.0
4    17.0
Name: mpg, dtype: float64
```

Splitting the dataset into train and test

In [25]:

```
from sklearn.model_selection import train_test_split
x_train,x_test, y_train, y_test = train_test_split(x,y,test_size=0.2)
```

In [26]:

```
x_train.shape,x_test.shape,y_train.shape,y_test.shape
```

Out[26]:

```
((318, 7), (80, 7), (318,), (80,))
```

Normalizing the values

In [28]:

```
from sklearn.preprocessing import StandardScaler
sd = StandardScaler()
x_train=sd.fit_transform(x_train)
x_test=sd.fit_transform(x_test)
```

In [30]:

```
x_train
```

Out[30]:

```
array([[ 0.32894571, -0.34956192,  0.47636441, ..., -0.74142165,
        -0.81838932,  1.77992292],
       [ 0.32894571,  0.07155568, -0.49772381, ...,  0.95037804,
         0.832231  , -0.71904171],
       [-0.85302871, -0.50269559, -0.36609027, ..., -0.02150689,
        -1.36859609, -0.71904171],
       ...,
       [ 0.32894571,  0.55009841,  0.02881036, ..., -0.38146427,
         0.00692084, -0.71904171],
       [-0.85302871, -0.98123832, -0.7609909 , ..., -0.38146427,
        -0.54328593, -0.71904171],
       [-0.85302871, -0.90467148, -0.94527786, ...,  1.05836525,
         0.28202423,  1.77992292]])
```

In [ ]: