* **Perform the following EDA on cars.csv file.**

**Find the datatype of all columns.**

**Find the number of missing values in each column.**

**Change the datatype of columns according to the data.**

**Fill missing values with the mean.**

**Describe the data in the given dataset.**

**Remove the outliers from the data.**

**Convert the categorical columns using encoder.**

1. **Perform the following EDA on cars.csv file.**

* import numpy as np
* import pandas as pd
* import warnings
* warnings.filterwarnings("ignore")
* df = pd.read\_csv("cars.csv")
* df

1. **Find the datatype of all columns.**

* df.dtypes

1. **Find the number of missing values in each column**.

* df.info()
* df["normalized-losses"].value\_counts()
* df["normalized-losses"].replace("?",np.nan, inplace = True)
* df
* df.info()

**4) Change the datatype of columns according to the data.**

* df["normalized-losses"] = df["normalized-losses"].astype(float)
* df.info()

**5) Fill missing values with the mean.**

* df\_mean = df["normalized-losses"].mean()
* df\_mean
* df["normalized-losses"]=df["normalized-losses"].fillna(df\_mean)
* df

**Doing 3rd step for horsepower column and then dropping rows which contain nan(only 2 nan) and then doing 4th step.**

* df["horsepower"].value\_counts()
* df["horsepower"].replace("?",np.nan, inplace = True)
* df["horsepower"].value\_counts()
* df
* df =df.dropna()
* df.isnull().sum()
* df["horsepower"] = df["horsepower"].astype(float)
* df.info()
* df.head()

**6) Describe the data in the given dataset.**

* df.describe()

**7) Remove the outliers from the data.**

* import seaborn as sns
* from matplotlib import pyplot as plt
* plt.figure(figsize = (10,10))
* sns.boxplot(y = df["price"], palette = "dark")
* plt.ylabel("Price", fontsize = 20, color = "green", weight = "bold")
* plt.grid()
* plt.show()
* plt.figure(figsize = (10,10))
* sns.boxplot(data = df, x = "make", y = "price", palette = "dark")
* plt.xlabel("Make", fontsize = 20, color = "green", weight = "bold")
* plt.ylabel("Price", fontsize = 20, color = "green", weight = "bold")
* plt.title("Make vs Price", fontsize = 20, color = "orange", weight = "bold")
* plt.xticks(rotation = 90)
* plt.grid()
* plt.show()
* df[(df["make"] == "dodge") & (df["price"] >= 12000)]
* df.drop(29, inplace = True)
* df[(df["make"] == "honda") & (df["price"] >= 12000)]
* df.drop(41, inplace = True)
* df[(df["make"] == "isuzu") & (df["price"] >= 23000)]
* df.drop(45, inplace = True)
* df[(df["make"] == "mitsubishi") & (df["price"] >= 14000)]
* df.drop(index = [83,84], inplace = True)
* df[(df["make"] == "plymouth") & (df["price"] >= 12000)]
* df.drop(124, inplace =True)
* df[(df["make"] == "toyota") & (df["price"] >= 15000)]
* df.drop(index = [172,178,179,180,181], inplace = True)

**8) Convert the categorical columns using encoder.**

* df\_cat = df.select\_dtypes("object")
* df\_cat
* df\_num = df.select\_dtypes(["int","float"])
* df\_num
* import numpy as np
* from sklearn.preprocessing import LabelEncoder
* le = LabelEncoder()
* for i in df\_cat:
* df\_cat[i] = le.fit\_transform(df\_cat[i])
* df\_cat
* new\_num = pd.concat([df\_cat,df\_num],axis = 1) # Concatenation happens column vise
* new\_num