**Data Analysis Project-1**

**Dataset overview:-**

* The dataset contains 371,528 rows of information, each representing a car listed for sale on ebay online marketplace.
* **dateCrawled** - When this ad was first crawled. All field-values are taken from this date.
* **name** - Name of the car.
* **seller** - Whether the seller is private or a dealer.
* **offerType** - The type of listing
* **price** - The price on the ad to sell the car.
* **abtest** - Whether the listing is included in an A/B test.
* **vehicleType** - The vehicle Type.
* **yearOfRegistration** - The year in which which year the car was first registered.
* **gearbox** - The transmission type.
* **powerPS** - The power of the car in PS.
* **model** - The car model name.
* **kilometer** - How many kilometers the car has driven.
* **monthOfRegistration** - The month in which which year the car was first registered.
* **fuelType** - What type of fuel the car uses.
* **brand** - The brand of the car.
* **notRepairedDamage** - If the car has a damage which is not yet repaired.
* **dateCreated** - The date on which the eBay listing was created.
* **nrOfPictures** - The number of pictures in the ad.
* **postalCode** - The postal code for the location of the vehicle.
* **lastSeenOnline** - When the crawler saw this ad last online.

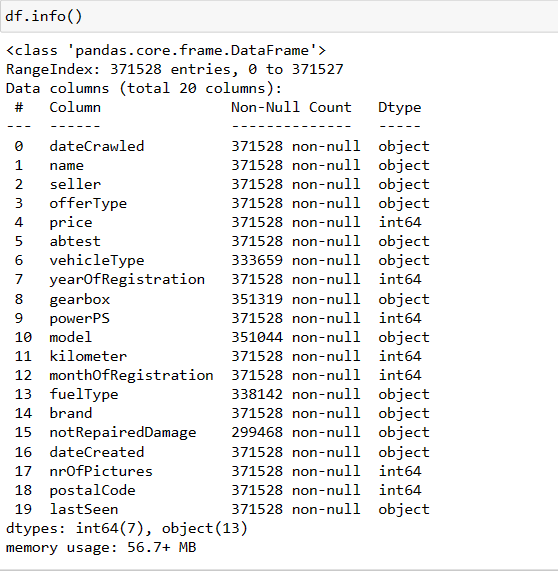
**Dataset Link** :-

<https://drive.google.com/file/d/1NFpNI-AiSLmta139rYaWrNy8xZ0VrbM9/view?usp=sharing>

**Analysis – 1**

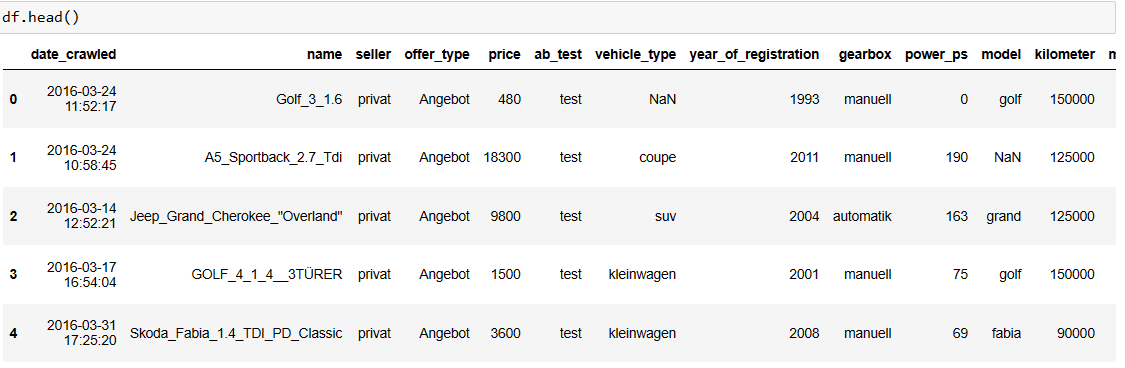
**1. Perform general Data analysis**

* Importing all the necessary libraries:
* **import pandas as pd**
* **import seaborn as sns**
* **import numpy as np**
* **import matplotlib.pyplot as plt**
* **Loading the dataset :**
* data=pd.read\_csv(r"C:\Users\Afrin\Downloads\autos.csv",encoding='Latin-1')
* **Brief Observation :-**

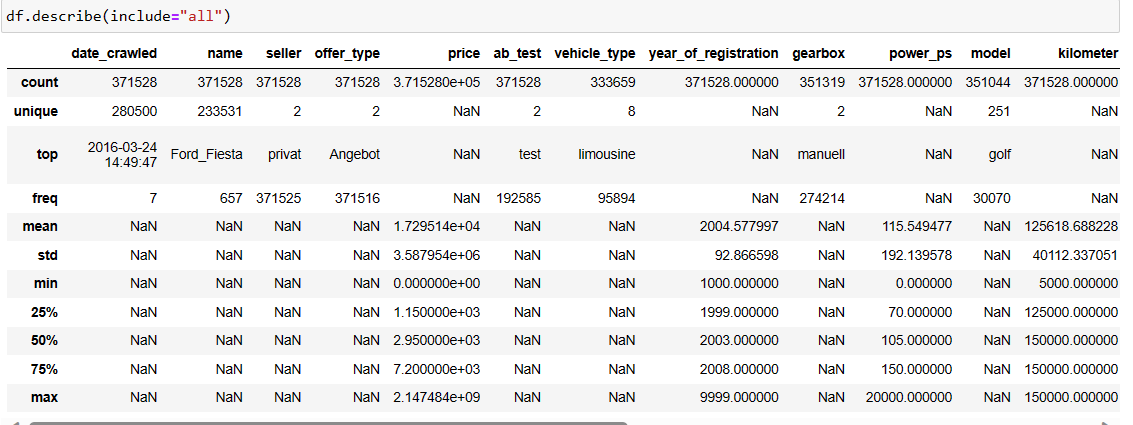
****

* The dataset consists of 20 columns
  + 13 columns contain data of object type, 5 columns are int.64 type.
  + camelcase in column names is needed to be changed to snakecase,which means we can just replace spaces with underscores.





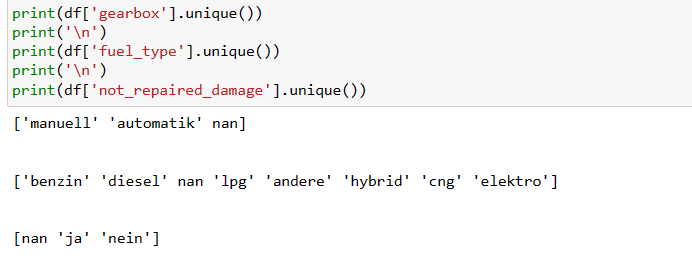
* In the cells above I have made the following modifications:
* I changed the style of writing the column names from camelcase to snakecase.
* This change will make column names easier to read, and I won't have to remember which letter is capitalized.
* I reworded some of the column names in order for them to be more descriptive and obvious.



* From above we can see that the num\_of\_pictures columns consists of zero values
* Hence we can drop that column as that will not help us in analysis.



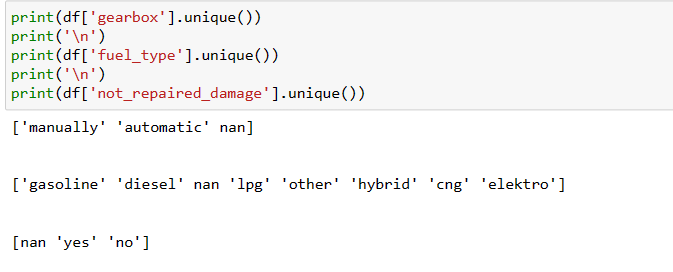
* Here we have dropped the num\_of\_pictures column.
* We can also notice that here some columns contain german words, which is normal since this dataset is from a german website.
* Let's fix that here by taking a look in the distinct values in each one of those columns so we know what words we'll need to translate.



* Now that we know all the german words in those columns, we'll create a dictionary for each of the columns containing the german word as key and the corresponding english word as value.
* Then, we'll use the pandas.Series.map to replace the german words for the english words in the whole dataset.

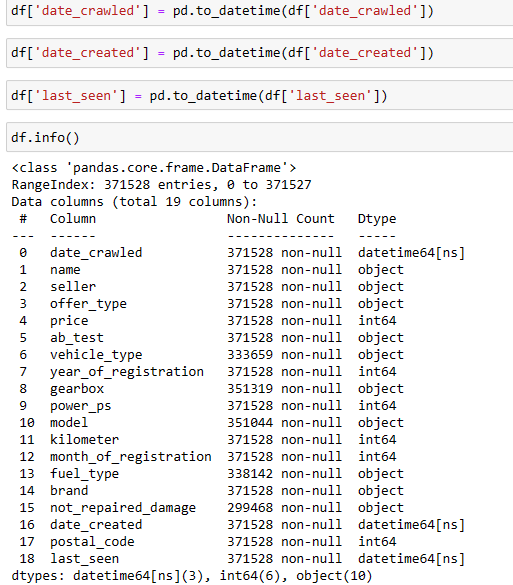


* Let’s check the results



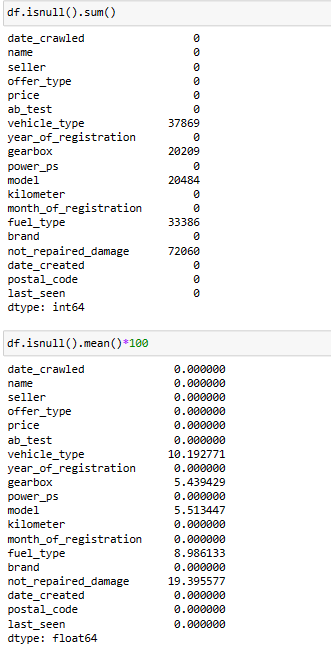
* Here we have successfully translated all the German letters into English letters

Now,lets change the datatypes of datetime cloumns which are in object type



* Here we have converted all the date time columns from object dtype to datetime dtype

**Treating the NULL values :**



* + here we can see that the 5 columns has missing values
  + and these 5 columns are categorical columns
  + None of these 5 columns has more than 20% of missing values

I have used the following code to replace all the null values with mode

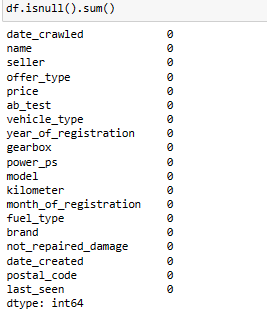
**df['vehicle\_type'].fillna(df['vehicle\_type'].mode()[0],inplace=True)**

**df=df.fillna({'gearbox':df['gearbox'].mode()[0],**

**'model':df['model'].mode()[0],**

**'fuel\_type':df['fuel\_type'].mode()[0],**

**'not\_repaired\_damage':df['not\_repaired\_damage'].mode()[0]})**

****

* After replacing the NULL values we can see that there are Zero NULL values present in the dataset.

**Can you tell me the Distribution of Vehicles based on Year of Registration with the help of a plot**

* using matplotlib and seaborn to create a distribution plot of the distribution of vehicles by year of registration.
* Using the code:

**df['year\_of\_registration'].where(((df['year\_of\_registration'] < 2023) & (df['year\_of\_registration'] > 1900)),inplace = True)**

**plt.subplots(figsize=(9,7))**

**sns.distplot(df['year\_of\_registration'])**

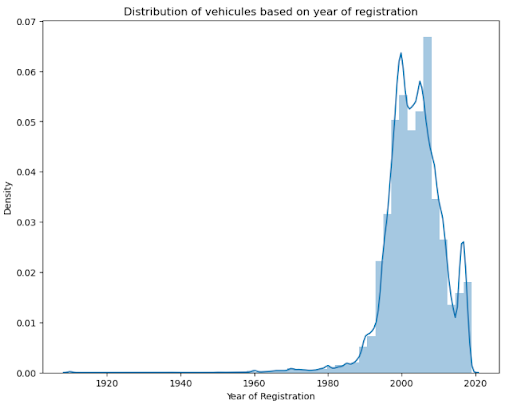
**plt.title('Distribution of vehicules based on year of registration')**

**plt.ylabel('Density')**

**plt.xlabel('Year of Registration')**

**plt.show()**

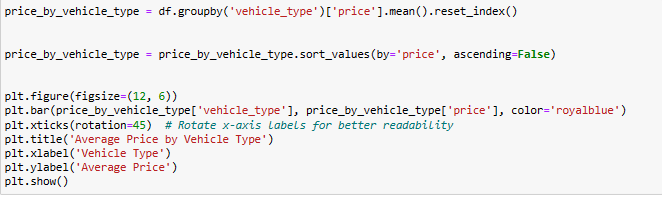
* The first line of code filters the data frame to only include vehicles that were registered between 1900 and 2023. This is done using the **where()** function, which takes a condition and a value as arguments. The condition in this case is **(df['year\_of\_registration'] < 2023) & (df['year\_of\_registration'] > 1900)**, which means that the vehicle must have been registered before 2023 and after 1900. The value argument **is inplace=True**, which means that the filtered data frame is assigned to the original data frame.
* **Plt.subplots()** is used to generate the new figure of size 9 inches by 7 inches.Then**, sns.distplot()** is used to generate the density plot with the help of dataset column
* Finally , we have given the y-axis label as “Density” using plt.ylabel() function
* And x-azis label as “Year of Registration using plt.xticks() function.



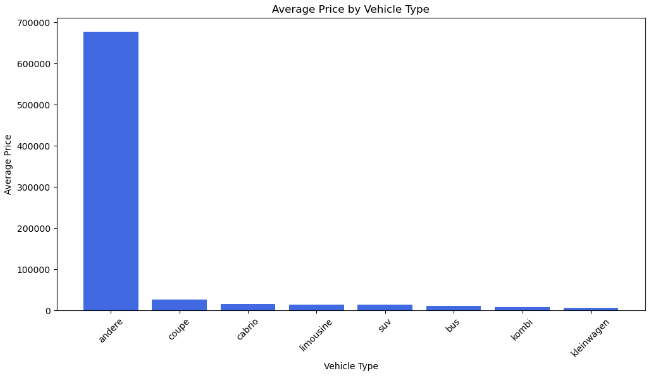
* The graph shows that the number of vehicles registered has increased steadily over time.
* There is a slight dip in the density of vehicles registered in the 1940s, However, the density of vehicles registered has increased rapidly since the 1960s.
* The cumulative density line shows that by 2020, over 90% of vehicles had been registered in the previous 30 years.
* Overall, the visualization shows that the number of vehicles registered has increased steadily over time, and that the fleet of vehicles on the road is relatively young.

# Create a plot based on the Variation of the price range by the vehicle type

* I am using seaborn's barplot function to create a bar plot.
* Using the code :



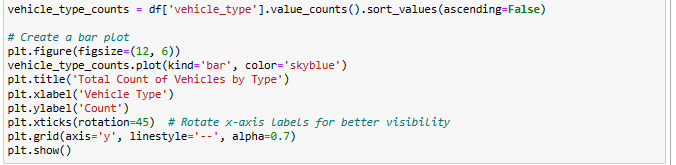
* using the **groupby()** function we calculates the mean price for each group.
* The **reset\_index()** function is then used to reset the index of the data frame, so that the vehicle type is now a column in the data frame.
* using the **sort\_values()** function, with the ascending we have sort the data frame by price in descending order.
* using the **subplots()** function, with the figsize argument set to (12, 6).creates a new figure with a size of 12 inches by 6 inches.
* using the **plt.bar()** function we have created a bar chart.
* and finally we have set the x-axis label to "Vehicle Type" using **plt.xlabel()** and y-axis label to "Average Price" using **plt.ylabel().**



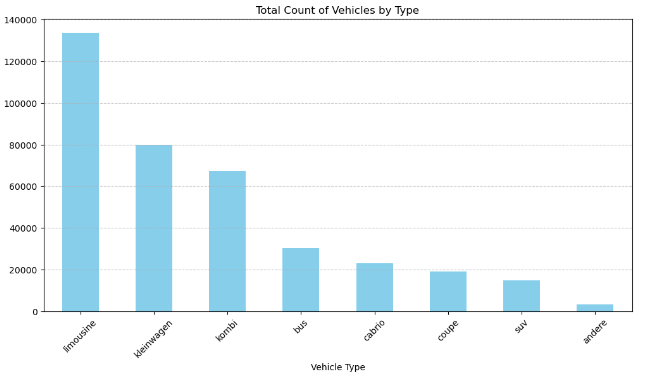
* The above bar graph shows the Average price of vehicles by vehicle types
* Here we can see that the Andere type vehicle has largest price where as the Kleinwagen has the every low price

# Find out Total count of vehicles by type available on ebay for sale.As well as create a visualization for the client

* I am using seaborn's barplot function to create a bar plot.
* Using the code :



* using the **sort\_values()** function, with the ascending we have sort the data frame by price in descending order.
* using the **subplots()** function, with the figsize argument set to (12, 6).creates a new figure with a size of 12 inches by 6 inches.
* using the **plt.bar()** function we have created a bar chart.
* and finally we have set the x-axis label to "Vehicle Type" using **plt.xlabel()** and y-axis label to "Count" using **plt.ylabel().**

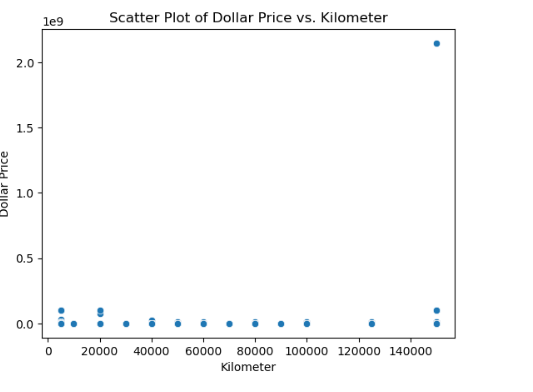


* From the above bar plot e can see that the vehicle type “Limousine” has the highest count as compared to the other vehicle types.
* Here the vehicle type “Andere” has the lowest count .

# Is there any relationship between dollar\_price and kilometer? (Explain with appropriate analysis)

# 

* Pearson's correlation coefficient of -0.00 indicates that there is no linear correlation between the two variables, price and kilometer, in the data set. This could be due to a number of factors, such as a small sample size, outliers, or a non-linear relationship between the two variables.



* Here we have created the Scatter plot to show the correlation between the price and kilometre.