Question 2:

Intent:

We would like you to learn the basics of python and data science to load a dataset, read it and perform some operations to find multiple mathematical metrics such as average, maximum, minimum and such.

Here is a dataset for autos.
https://drive.google.com/file/d/1QP21K5tiJAjt5NA7W2FxSe9Wam9-tIcQ/view?usp=sharing

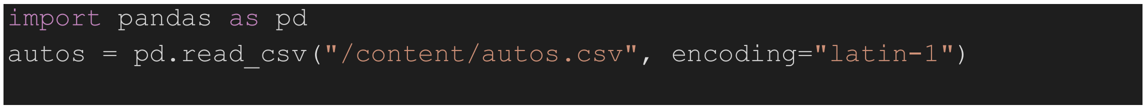
Flow:

1. Download this dataset.
2. Write basic python script to load csv and read it as dataframe
3. Use the dataframe to perform following:
    a. Find Average price of autos (using **price** column of dataset)
    b. Print the list of different possible types of **VehicleType** found in dataset
    c. Calculate and print lowest **yearOfRegistration** and highest **yearOfRegistration**
    d. Find and print standard deviation of column **kilometer**
    e. Draw a bar graph to represent count of different type of column **brand**
    f. Find out which **VehicleType** is sold minimum and maximum
    g. Create a pie chart to represent different types of **gearbox** count

---

https://colab.research.google.com/drive/1X4kPx3xOqisC106fq3NRcC3wPBeqfMBl?usp=sharing

(The link to google collab where I executed all the queries)



```
import pandas as pd
autos = pd.read_csv("/content/autos.csv", encoding="latin-1")
```

Python script to load the csv and read it as a data frame
Data frame’s name being ‘autos’
Here initially I **import** a module named pandas [**import pandas as pd]**
PANDAS is an open source python library used for data manipulation and analysis
It has many reusable functions necessary for working with huge amount of structured data
This library is built on top of NumPy library

Few features of panda library include:

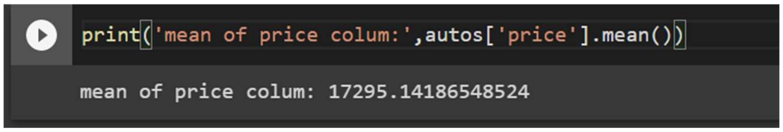
- Data manipulation
- Data structures
- Missing data handling
- I/O : it supports reading and writing data in various formats like CSV, Excel etc.

The 2nd line of code, a CSV file is being read and the data frame named ‘autos’ is being created!

```
autos
```

on typing the data frame’s name the data set will be created.

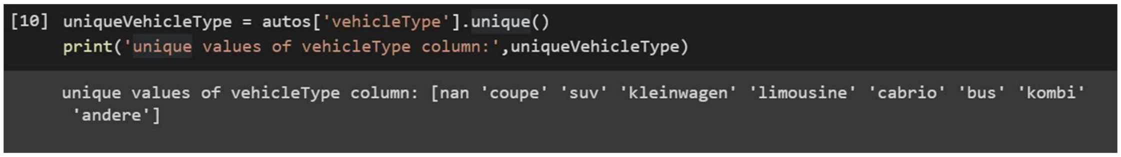
a) **Find Average price of autos (using price column of dataset)**



```
print('mean of price colum:',autos['price'].mean())

mean of price colum: 17295.14186548524
```

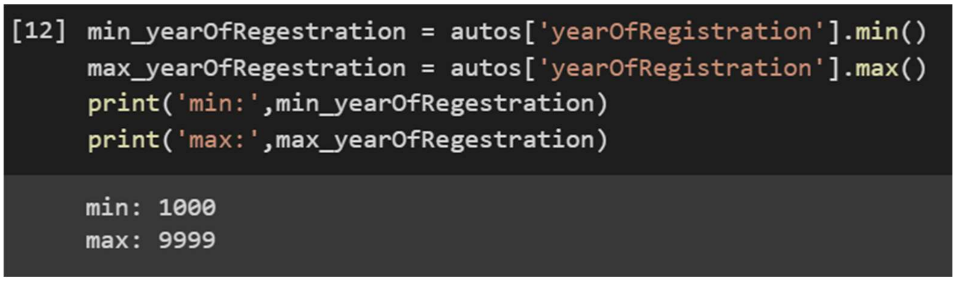
b) Print the list of different possible types of VehicleType found in dataset



```
[10] uniqueVehicleType = autos['vehicleType'].unique()
     print('unique values of vehicleType column:',uniqueVehicleType)

     unique values of vehicleType column: [nan 'coupe' 'suv' 'kleinwagen' 'limousine' 'cabrio' 'bus' 'kombi'
      'andere']
```

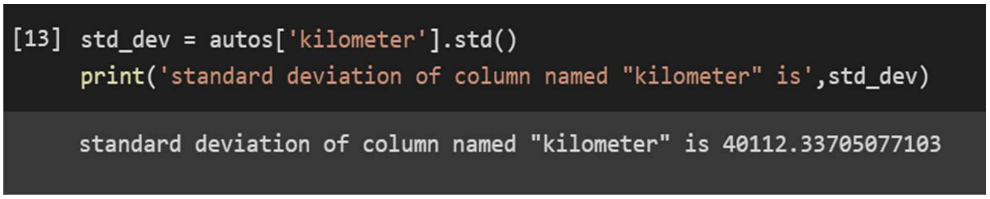
c) Calculate and print lowest yearOfRegistration and highest yearOfRegistration



```
[12] min_yearOfRegestration = autos['yearOfRegistration'].min()
     max_yearOfRegestration = autos['yearOfRegistration'].max()
     print('min:',min_yearOfRegestration)
     print('max:',max_yearOfRegestration)

     min: 1000
     max: 9999
```

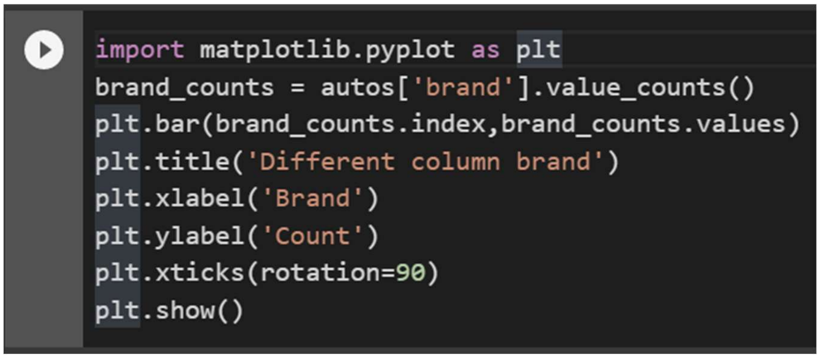
d) Find and print standard deviation of column kilometer



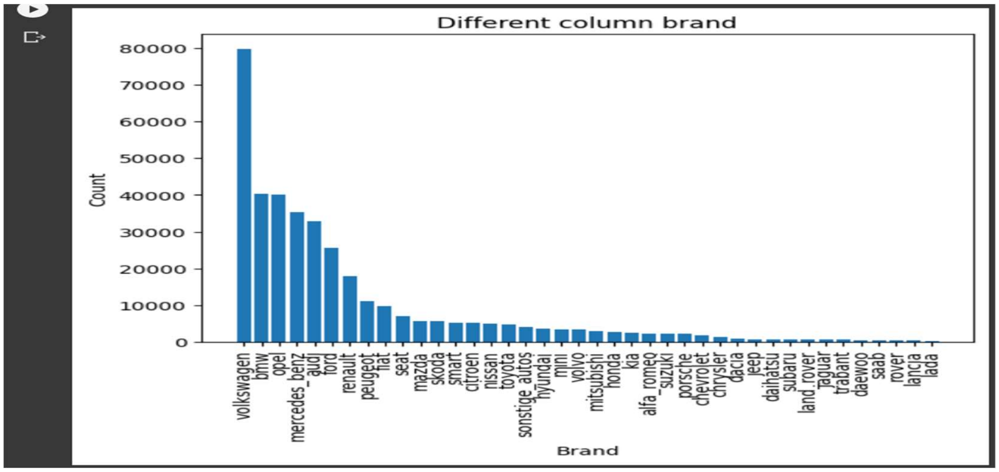
```
[13] std_dev = autos['kilometer'].std()
     print('standard deviation of column named "kilometer" is',std_dev)

     standard deviation of column named "kilometer" is 40112.33705077103
```

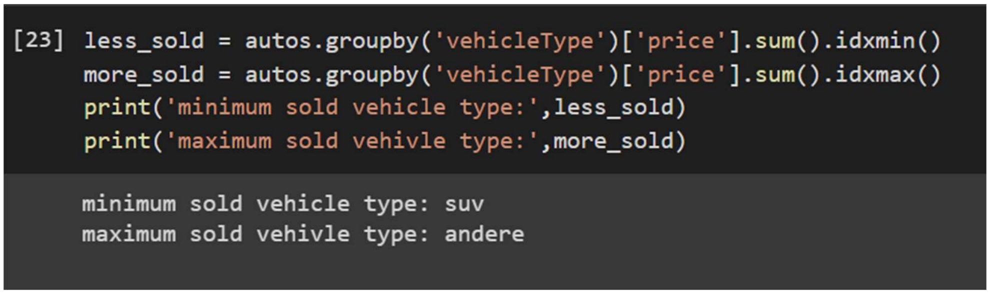
e) **Draw a bar graph to represent count of different type of column brand**



```
import matplotlib.pyplot as plt
brand_counts = autos['brand'].value_counts()
plt.bar(brand_counts.index,brand_counts.values)
plt.title('Different column brand')
plt.xlabel('Brand')
plt.ylabel('Count')
plt.xticks(rotation=90)
plt.show()
```



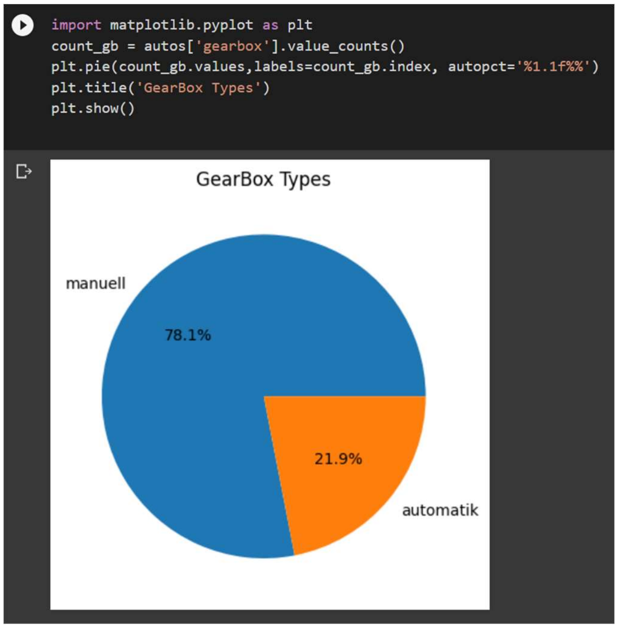
f) **Find out which VehicleType is sold minimum and maximum**



```
[23] less_sold = autos.groupby('vehicleType')['price'].sum().idxmin()
     more_sold = autos.groupby('vehicleType')['price'].sum().idxmax()
     print('minimum sold vehicle type:',less_sold)
     print('maximum sold vehivle type:',more_sold)
```

```
minimum sold vehicle type: suv
maximum sold vehivle type: andere
```

g) Create a pie chart to represent different types of gearbox count



```
import matplotlib.pyplot as plt
count_gb = autos['gearbox'].value_counts()
plt.pie(count_gb.values,labels=count_gb.index, autopct='%1.1f%%')
plt.title('GearBox Types')
plt.show()
```

Advantages of pandas library

- Easy to use
- Comprehensive data structure
- Flexible data handling
- Powerful data visualization
- Good integration with other libraries

Disadvantages of pandas

- Large memory usage
- Slow memory usage
- Steep learning curve
- Limited functionality
- Inefficient for some operations

if you need to perform basic mathematical operations on small datasets, you can use Python's built-in data structures such as lists, tuples, and dictionaries.

If you need to perform complex operations on large datasets or manipulate tabular data, pandas may be a better choice as it provides a comprehensive set of data structures and functions specifically designed for these tasks.