

for data analysis

```
In [1]: import pandas as pd
import numpy as np
```

for data visualization

```
In [3]: import seaborn as sns
import matplotlib.pyplot as plt
```

```
In [35]: df= pd.read_csv("/Users/Hp/AppData/Roaming/Microsoft/Windows/Start Menu/Programs/Anaconda3 (64-bit)/Data_Visualization_with_Python_s2-main/vw.csv")
display(df.shape)
print('number of rows=',df.shape[0])
print('number of columns=',df.shape[1])
display(df)
print(df)
```

(15157, 8)  
number of rows= 15157  
number of columns= 8

	model	year	price	transmission	mileage	fuelType	mpg	engineSize
0	T-Roc	2019	25000	Automatic	13904	Diesel	49.6	2.0
1	T-Roc	2019	26883	Automatic	4562	Diesel	49.6	2.0
2	T-Roc	2019	20000	Manual	7414	Diesel	50.4	2.0
3	T-Roc	2019	33492	Automatic	4825	Petrol	32.5	2.0
4	T-Roc	2019	22900	Semi-Auto	6500	Petrol	39.8	1.5
...	...	...	...	...	...	...	...	...
15152	Eos	2012	5990	Manual	74000	Diesel	58.9	2.0
15153	Fox	2008	1799	Manual	88102	Petrol	46.3	1.2
15154	Fox	2009	1590	Manual	70000	Petrol	42.0	1.4
15155	Fox	2006	1250	Manual	82704	Petrol	46.3	1.2
15156	Fox	2007	2295	Manual	74000	Petrol	46.3	1.2

15157 rows × 8 columns

	model	year	price	transmission	mileage	fuelType	mpg	engineSize
0	T-Roc	2019	25000	Automatic	13904	Diesel	49.6	2.0
1	T-Roc	2019	26883	Automatic	4562	Diesel	49.6	2.0
2	T-Roc	2019	20000	Manual	7414	Diesel	50.4	2.0
3	T-Roc	2019	33492	Automatic	4825	Petrol	32.5	2.0
4	T-Roc	2019	22900	Semi-Auto	6500	Petrol	39.8	1.5
...	...	...	...	...	...	...	...	...
15152	Eos	2012	5990	Manual	74000	Diesel	58.9	2.0
15153	Fox	2008	1799	Manual	88102	Petrol	46.3	1.2
15154	Fox	2009	1590	Manual	70000	Petrol	42.0	1.4
15155	Fox	2006	1250	Manual	82704	Petrol	46.3	1.2
15156	Fox	2007	2295	Manual	74000	Petrol	46.3	1.2

[15157 rows x 8 columns]

```
In [6]: display(df.head())
```

	model	year	price	transmission	mileage	fuelType	mpg	engineSize
0	T-Roc	2019	25000	Automatic	13904	Diesel	49.6	2.0
1	T-Roc	2019	26883	Automatic	4562	Diesel	49.6	2.0
2	T-Roc	2019	20000	Manual	7414	Diesel	50.4	2.0
3	T-Roc	2019	33492	Automatic	4825	Petrol	32.5	2.0
4	T-Roc	2019	22900	Semi-Auto	6500	Petrol	39.8	1.5

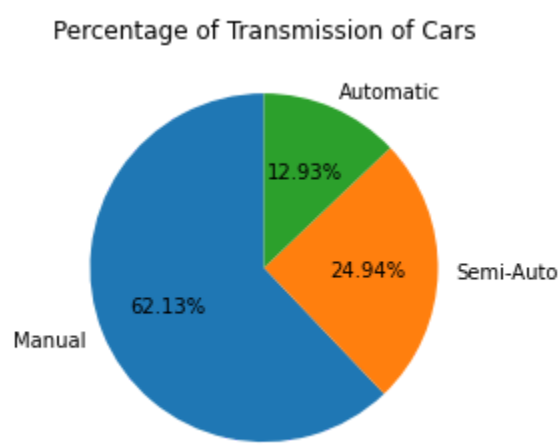
value counts()

```
In [8]: display(df["transmission"].value_counts())
```

Manual 9417  
Semi-Auto 3780  
Automatic 1960  
Name: transmission, dtype: int64

Pie-Chart

```
In [26]: import matplotlib.pyplot as plt
import seaborn as sns
df["transmission"].value_counts().plot(kind="pie",autopct='%1.2f%%',startangle=90)
plt.title("Percentage of Transmission of Cars")
plt.ylabel("")
plt.show()
```



Creating DataFrame

```
In [22]: dfame=pd.DataFrame(df["transmission"].value_counts())
display(dfame)
```

transmission	
Manual	9417
Semi-Auto	3780
Automatic	1960

```
In [24]: dfame=dfame.reset_index()
dfame=dfame.rename(columns={"index":"transmission type",
                             "transmission":"number of cars"})
display(dfame)
```

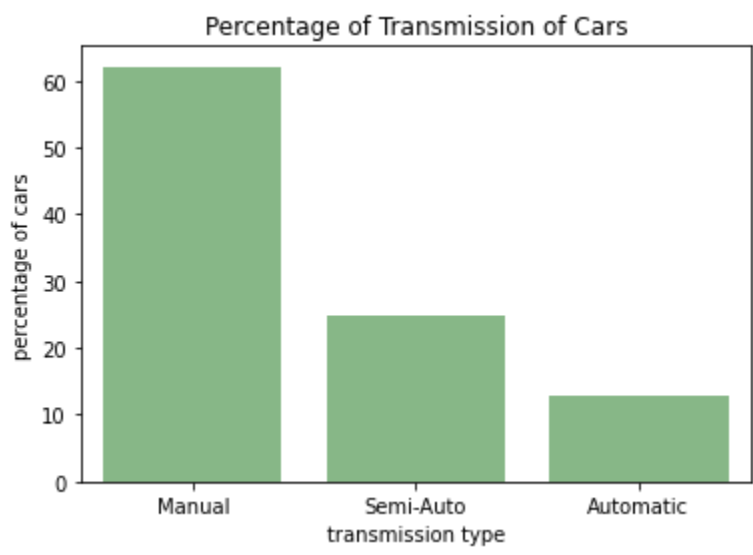
	transmission type	number of cars
0	Manual	9417
1	Semi-Auto	3780
2	Automatic	1960

```
In [41]: dfame["percentage of cars"]=(dfame["number of cars"]/df.shape[0])*100
display(dfame.round(2))
```

	transmission type	number of cars	percentage of cars
0	Manual	9417	62.13
1	Semi-Auto	3780	24.94
2	Automatic	1960	12.93

Bar-plot

```
In [42]: sns.barplot(x="transmission type",y="percentage of cars",data=dfame,alpha=0.50,color="green")
plt.title("Percentage of Transmission of Cars")
plt.show()
```



Comparison

The pie-chart gives a clear indication that manual type holds greater percentage alone than the the other two types combined. If needed, percentages can also be shown as annotations in the bar-chart but the pie-chart shows this comparison even without annotations. On the other hand, x and y-labels in the bar-graph presents a more detailed picture than the pie-chart as labels arev not used in pie-charts.

So, for a more detailed picture, bars can be helpful while pie-charts are applicable to make visualizations in an easier,faster way

```
In [ ]:
```