Car data Analysis Data Cleaning and transformation

Project by Mayur Borgude

In [66]:
```
import pandas as pd
import numpy as np
from matplotlib import pyplot as plt
```

In [50]:
```
# reading CSV as dataframes
df = pd.read_csv(r"https://raw.githubusercontent.com/Mayborg121/Car_DataAnalysis_Dashboard/main/Car%20Dashboard
df
```

Out[50]:

	manufacturer_name	model_name	transmission	color	odometer_value	year_produced	engine_fuel	engine_has_gas	engine_type	en
0	Ford	Fusion	mechanical	blue	245000	2006	gasoline	False	gasoline	
1	Dodge	Caravan	automatic	silver	265542	2002	gasoline	False	gasoline	
2	Ford	Galaxy	mechanical	blue	168000	2009	diesel	False	diesel	
3	Mazda	6	mechanical	other	225522	2008	gasoline	False	gasoline	
4	Audi	80	mechanical	black	370000	1991	gasoline	False	gasoline	
...	...	...	...	...	...	...	...	...	...	
30819	Mazda	Millenia	automatic	green	360493	1999	gasoline	False	gasoline	
30820	Audi	A3	automatic	grey	117000	2009	gasoline	False	gasoline	
30821	Mazda	626	mechanical	black	333000	1997	gasoline	False	gasoline	
30822	Audi	A6	automatic	violet	530000	1995	gasoline	False	gasoline	
30823	Honda	Odyssey	automatic	white	15000	2018	gasoline	False	gasoline	

30824 rows × 26 columns

- listing the cloumns to analyze the data

In [51]:
```
attributes_list = list(df)
print(attributes_list)
```

```
['manufacturer_name', 'model_name', 'transmission', 'color', 'odometer_value', 'year_produced', 'engine_fuel',
'engine_has_gas', 'engine_type', 'engine_capacity', 'body_type', 'has_warranty', 'state', 'drivetrain', 'featur
e_0', 'feature_1', 'feature_2', 'feature_3', 'feature_4', 'feature_5', 'feature_6', 'feature_7', 'feature_8', '
feature_9', 'duration_listed', 'price_usd']
```

In [52]:
```
print(attributes_list.index("feature_0"))
```

```
14
```

In [53]:
```
print(attributes_list.index("feature_9"))
```

```
23
```

- removing multiple columns at time.

In [54]:
```
df = df.drop("engine_type", axis = 1)
df
```

Out[54]:

	manufacturer_name	model_name	transmission	color	odometer_value	year_produced	engine_fuel	engine_has_gas	engine_capacity
0	Ford	Fusion	mechanical	blue	245000	2006	gasoline	False	1.6
1	Dodge	Caravan	automatic	silver	265542	2002	gasoline	False	3.3
2	Ford	Galaxy	mechanical	blue	168000	2009	diesel	False	1.8
3	Mazda	6	mechanical	other	225522	2008	gasoline	False	1.8
4	Audi	80	mechanical	black	370000	1991	gasoline	False	1.8
...	...	...	...	...	...	...	...	...	...
30819	Mazda	Millenia	automatic	green	360493	1999	gasoline	False	2.5
30820	Audi	A3	automatic	grey	117000	2009	gasoline	False	1.4
30821	Mazda	626	mechanical	black	333000	1997	gasoline	False	2.0
30822	Audi	A6	automatic	violet	530000	1995	gasoline	False	2.6
30823	Honda	Odyssey	automatic	white	15000	2018	gasoline	False	3.5

30824 rows × 25 columns

- Checking for NuLL values in dataset.

In [55]:
```
df.isnull().sum()
```

Out[55]:
```
manufacturer_name    0
model_name           0
transmission         0
color                0
odometer_value       0
year_produced        0
engine_fuel          0
engine_has_gas       0
engine_capacity      8
body_type            0
has_warranty         0
state                0
drivetrain           0
feature_0            0
feature_1            0
feature_2            0
feature_3            0
feature_4            0
feature_5            0
feature_6            0
feature_7            0
feature_8            0
feature_9            0
duration_listed      0
price_usd            0
dtype: int64
```

In [ ]:
```

```

In [57]:
```
manufacturer_list = df['manufacturer_name'].value_counts()
print(manufacturer_list,len(manufacturer_list))
```

```
manufacturer_name
Volkswagen       3406
Opel             2193
BMW              2091
Ford             2080
Renault          1999
Audi             1972
Mercedes-Benz    1807
Peugeot          1529
Citroen          1236
Nissan           1084
Mazda            1081
Toyota            994
Hyundai           908
Skoda             906
Kia               715
Mitsubishi        711
Fiat              656
Honda             639
Volvo             573
ВАЗ               386
Chevrolet         344
Chrysler          317
Dodge             247
Subaru            238
Seat              223
Rover             193
Suzuki            185
Daewoo            165
Alfa Romeo        162
ГАЗ               157
Lexus             155
Land Rover        142
Infiniti          135
LADA              125
Iveco             110
Jeep               82
Saab               81
Lancia             76
SsangYong          59
Geely              59
УАЗ                54
Acura              50
Mini               48
Chery              48
Porsche            45
Jaguar             45
Dacia              40
Москвич            40
Lifan              38
ЗАЗ                37
Buick              35
Cadillac           34
Pontiac            33
Great Wall         28
Lincoln            28
Name: count, dtype: int64 55
```

In [45]:

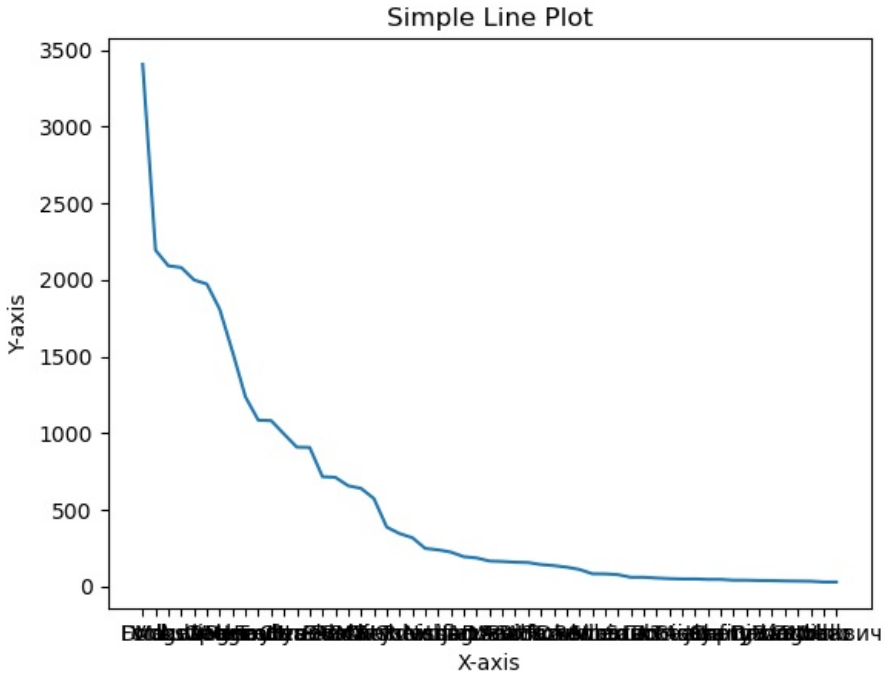
```
# variable to hold the count
cnt = 0

# list to hold visited values
visited = []

# loop for counting the unique
# values in height
for i in range(0, len(df['manufacturer_name'])):

    if df['manufacturer_name'][i] not in visited:

        visited.append(df['manufacturer_name'][i])

        cnt += 1

print("unique values :",visited, len(visited))
```

```
unique values : ['Ford', 'Dodge', 'Mazda', 'Audi', 'Volkswagen', 'Opel', 'Volvo', 'Peugeot', 'Renault', 'Honda'
, 'Toyota', 'Mercedes-Benz', 'Citroen', 'Hyundai', 'ВАЗ', 'Skoda', 'BMW', 'Kia', 'Fiat', 'Chrysler', 'Mitsubish
i', 'Rover', 'Chevrolet', 'Nissan', 'Lifan', 'LADA', 'Jaguar', 'УАЗ', 'Seat', 'Buick', 'Land Rover', 'Porsche',
'Suzuki', 'Alfa Romeo', 'Daewoo', 'Mini', 'Subaru', 'Lexus', 'Saab', 'ГАЗ', 'Lancia', 'Pontiac', 'Geely', 'Acur
a', 'Jeep', 'Chery', 'Infiniti', 'SsangYong', 'Dacia', 'ЗАЗ', 'Great Wall', 'Lincoln', 'Cadillac', 'Iveco', 'Mo
сквич'] 55
```

In [58]:

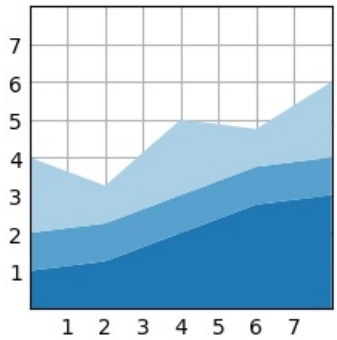
```
x = visited  # X-axis points
y = df['manufacturer_name'].value_counts()              # Y-axis points

plt.plot(x, y)             # Plot the chart
plt.xlabel("X-axis")       # Add X-axis label
plt.ylabel("Y-axis")       # Add Y-axis label
```

```
plt.title("Simple Line Plot")  # Add title
plt.show()                     # Display the plot
```



In [ ]:

In [64]:
```
plt.style.use('_mpl-gallery')

# make data
x = np.arange(0, 10, 2)
ay = [1, 1.25, 2, 2.75, 3]
by = [1, 1, 1, 1, 1]
cy = [2, 1, 2, 1, 2]
y = np.vstack([ay, by, cy])

# plot
fig, ax = plt.subplots()

ax.stackplot(x, y)

ax.set(xlim=(0, 8), xticks=np.arange(1, 8),
       ylim=(0, 8), yticks=np.arange(1, 8))

plt.show()
```



- Exporting the clean data into .CSV file

In [59]:
```
export_csv_data = df.to_csv('cars_data.csv', index = True)
print('\nCSV String:\n', export_csv_data)
```

```
CSV String:
 None
```