Step 1: Import the Required Libraries and Load Dataset into PandaFrame

[53]:
```
import pandas as pd
import numpy as np
df=pd.read_csv(r"C:\Users\UNKNOWN_CODER\DSDBA\Assign1\data.csv")
```

[54]:
```
print(df)
```

```
           Car       Model  Volume  Weight  CO2
0       Toyoty        Aygo    1000     790   99
1   Mitsubishi  Space Star    1200    1160   95
2        Skoda      Citigo    1000     929   95
3         Fiat         500     900     865   90
4         Mini      Cooper    1500    1140  105
5           VW         Up!    1000     929  105
6        Skoda       Fabia    1400    1109   90
7    Mercedes     A-Class    1500    1365   92
8         Ford      Fiesta    1500    1112   98
9         Audi          A1    1600    1150   99
10     Hyundai         I20    1100     980   99
11      Suzuki       Swift    1300     990  101
12        Ford      Fiesta    1000    1112   99
13       Honda       Civic    1600    1252   94
14      Hundai         I30    1600    1326   97
15        Opel       Astra    1600    1330   97
16         BMW           1    1600    1365   99
17       Mazda           3    2200    1280  104
18       Skoda       Rapid    1600    1119  104
19        Ford       Focus    2000    1328  105
20        Ford      Mondeo    1600    1584   94
21        Opel    Insignia    2000    1428   99
22   Mercedes     C-Class    2100    1365   99
23       Skoda     Octavia    1600    1415   99
24       Volvo         S60    2000    1415   99
25   Mercedes         CLA    1500    1465  102
26        Audi          A4    2000    1490  104
27        Audi          A6    2000    1725  114
28       Volvo         V70    1600    1523  109
29         BMW           5    2000    1705  114
30   Mercedes     E-Class    2100    1605  115
31       Volvo        XC70    2000    1746  117
32        Ford       B-Max    1600    1235  104
33         BMW         216    1600    1390  108
34        Opel      Zafira    1600    1405  109
35   Mercedes         SLK    2500    1395  120
```

Step 2: Data Preprocessing : check for missing values in the data using pandas isnull(), describe() function to get some initial statistics.

[55]:
```
## Check for Missing Values
missing_values = df.isnull().sum()
print(missing_values)
```

```
Car       0
Model     0
Volume    0
Weight    0
CO2       0
dtype: int64
```

[56]:
```
# Display the first few rows of the dataset
display(df.head())
```

	Car	Model	Volume	Weight	CO2
0	Toyoty	Aygo	1000	790	99
1	Mitsubishi	Space Star	1200	1160	95
2	Skoda	Citigo	1000	929	95
3	Fiat	500	900	865	90
4	Mini	Cooper	1500	1140	105

[57]:
```
## Statistical Summary
print("\nStatistical Summary:")
print(df.describe())
```

```
Statistical Summary:
            Volume       Weight         CO2
count    36.000000    36.000000   36.000000
mean   1611.111111  1292.277778  102.027778
std     388.975047   242.123889    7.454571
min     900.000000   790.000000   90.000000
25%    1475.000000  1117.250000   97.750000
50%    1600.000000  1329.000000   99.000000
75%    2000.000000  1418.250000  105.000000
max    2500.000000  1746.000000  120.000000
```

[58]:
```
## Dataset Dimensions
dimensions = df.shape
print(f"\nDataset Dimensions: {dimensions}")
```

```
Dataset Dimensions: (36, 5)
```

[59]:
```
print(df.info())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 36 entries, 0 to 35
Data columns (total 5 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   Car     36 non-null     object
 1   Model   36 non-null     object
 2   Volume  36 non-null     int64
 3   Weight  36 non-null     int64
 4   CO2     36 non-null     int64
dtypes: int64(3), object(2)
memory usage: 1.5+ KB
None
```

Step 3: Data Formatting and Normalization : Summarize the types of variables by checking the data types of the variables in the data set. If variables are not in the correct data type, apply proper type conversions.

[60]:
```
# Display data types
print(df.dtypes)

print("\nConverting 'Car' and 'Model' to Categorical Variables...")
df['Car'] = df['Car'].astype('category')
df['Model'] = df['Model'].astype('category')
print(df.dtypes)
```

```
Car       object
Model     object
Volume     int64
Weight     int64
CO2        int64
dtype: object


Converting 'Car' and 'Model' to Categorical Variables...
Car       category
Model     category
Volume       int64
Weight       int64
CO2          int64
dtype: object
```

Step 4: Converting Categorical Variables to Quantitative Variables

[61]:
```
# One-hot encoding for categorical variables
print("\nApplying One-Hot Encoding to Categorical Variables...")
df_encoded = pd.get_dummies(df, columns=['Car', 'Model'], drop_first=True)
print(df_encoded.head())
```

```
Applying One-Hot Encoding to Categorical Variables...
   Volume  Weight  CO2  Car_BMW  Car_Fiat  Car_Ford  Car_Honda  Car_Hundai  \
0    1000     790   99    False     False     False      False       False
1    1200    1160   95    False     False     False      False       False
2    1000     929   95    False     False     False      False       False
3     900     865   90    False      True     False      False       False
4    1500    1140  105    False     False     False      False       False

   Car_Hyundai  Car_Mazda  ...  Model_Octavia  Model_Rapid  Model_S60  \
0        False      False  ...          False        False      False
1        False      False  ...          False        False      False
2        False      False  ...          False        False      False
3        False      False  ...          False        False      False
4        False      False  ...          False        False      False

   Model_SLK  Model_Space Star  Model_Swift  Model_Up!  Model_V70  Model_XC70  \
0      False             False        False      False      False       False
1      False              True        False      False      False       False
2      False             False        False      False      False       False
3      False             False        False      False      False       False
4      False             False        False      False      False       False

   Model_Zafira
0         False
1         False
2         False
3         False
4         False

[5 rows x 53 columns]
```