DS1_C5_S1_Challenge

In [1]:

```
#Import the required liberary
import pandas as pd
import warnings
warnings.filterwarnings('ignore')
import matplotlib.pyplot as plt
import statistics as st
```

In [6]:

```
car = pd.read_excel(r'E:\Aishwarya official\Aishwarya Data Scince\course 5\DS1_C4_S5_Car_Da
car
```

Out[6]:

	Sl. No.	Make	Model	Variant	Displacement	Cylinders	Valves_Per_Cylinder	Drivetrai
0	0	Tata	Nano Genx	Xt	624.0	2.0	2.0	RW (Rea Whe Drive
1	1	Tata	Nano Genx	Xe	624.0	2.0	2.0	RW (Rea Whe Drive
2	2	Tata	Nano Genx	Emax Xm	624.0	2.0	2.0	RW (Rea Whe Drive
3	3	Tata	Nano Genx	Xta	624.0	2.0	2.0	RW (Rea Whe Drive
4	4	Tata	Nano Genx	Xm	624.0	2.0	2.0	RW (Rea Whe Drive
...	...	...	...	...	...	...	...	.
1271	1271	Honda	City	Vx Mt Diesel	1498.0	4.0	4.0	FW (Fro Whe Drive
1272	1272	Honda	City	Zx Mt Diesel	1498.0	4.0	4.0	FW (Fro Whe Drive
1273	1273	Honda	City	Zx Cvt Petrol	1497.0	4.0	4.0	FW (Fro Whe Drive
1274	1274	Honda	City	V Cvt Petrol	1497.0	4.0	4.0	FW (Fro Whe Drive
1275	1275	Mitsubishi	Montero	3.2 At	3200.0	4.0	4.0	AWD (A Whe Drive

1276 rows × 139 columns

Task 1

In [9]:

```
for item in car.columns:
    print(item," ",car[item].isna().sum())
```

```
Cylinder_Configuration   13
Engine_Location   43
Fuel_System   8
Fuel_Tank_Capacity_litre   69
Fuel_Type   0
Height_mm   1
Length_mm   0
Width_mm   12
Body_Type   6
Doors   4
City_Mileage_km_litre   555
Highway_Mileage_km_litre   800
ARAI_Certified_Mileage   114
ARAI_Certified_Mileage_for_CNG   1249
Kerb_Weight   365
Gears   105
Ground_Clearance   289
Front_Brakes   25
Rear_Brakes   25
Front_Suspension   59
```

In [8]:

```
car.isnull().sum()
```

Out[8]:

```
Sl. No.               0
Make                 75
Model                 0
Variant               0
Displacement         12
                   ...
USB_Ports          1247
Heads-Up_Display   1225
Welcome_Lights     1207
Battery            1263
Electric_Range     1259
Length: 139, dtype: int64
```

In [23]:

```
cr = car[['Make', 'Displacement', 'Fuel_Tank_Capacity_litre', 'City_Mileage_km_litre','Highw
cr.dropna(inplace=True)
cr
```

Out[23]:

	Make	Displacement	Fuel_Tank_Capacity_litre	City_Mileage_km_litre	Highway_Mileage
6	Datsun	799.0	28.0	21.38	
7	Datsun	799.0	28.0	21.38	
8	Datsun	799.0	28.0	21.38	
9	Datsun	799.0	28.0	21.38	
24	Suzuki	1196.0	40.0	12.00	
...	...	...	...	...	
1271	Honda	1498.0	40.0	22.60	
1272	Honda	1498.0	40.0	22.60	
1273	Honda	1497.0	40.0	18.00	
1274	Honda	1497.0	40.0	14.30	
1275	Mitsubishi	3200.0	88.0	8.25	

338 rows × 6 columns

In [38]:

```
cr=cr.groupby(['Make'])['Displacement','Fuel_Tank_Capacity_litre','City_Mileage_km_litre','
cr
```

Out[38]:

Make	Displacement	Fuel_Tank_Capacity_litre	City_Mileage_km_litre	Highway_Mileage_km
Audi	18098.0	375.0	41.30	
Bentley	22741.0	360.0	15.50	
Bmw	28919.0	788.0	171.32	2
Datsun	3196.0	112.0	85.52	
Dc	2000.0	60.0	8.00	
Fiat	11516.0	405.0	130.00	1
Force	10384.0	252.0	56.00	
Ford	7594.0	240.0	26.30	
Honda	28946.0	796.0	309.50	3
Hyundai	32460.0	1101.0	415.20	4
Icml	21934.0	550.0	94.60	1
Isuzu	5998.0	152.0	27.60	
Jaguar	23974.0	690.0	64.81	1
Lamborghini	6498.0	90.0	3.60	
Mahindra	76908.0	2225.0	548.05	6
Maserati	12369.0	231.0	13.80	
Mitsubishi	15585.0	438.0	40.75	
Nissan	10364.0	341.0	112.90	1
Porsche	11988.0	270.0	21.30	
Premier	8061.0	276.0	82.00	1
Renault	13260.0	450.0	126.20	1
Skoda	50630.0	1782.0	418.60	5
Suzuki	47594.0	1693.0	825.68	9
Tata	41682.0	1279.0	510.48	5
Toyota	68947.0	2129.0	629.01	7
Volkswagen	12881.0	450.0	171.00	1
Volvo	5549.0	177.0	38.25	

In [41]:

```
mean =[]
mode =[]
median =[]

for col in cr:
    mean.append(st.mean(cr[col]))
    mode.append(st.mode(cr[col]))
    median.append(st.median(cr[col]))

row_head = ['mean', 'mode', 'median']
col_name = ['Displacement','Fuel_Tank_Capacity_litr','City_Mileage_km_litre','Highway_Milea

# create dataframe of mean , median ,mode
d_data = pd.DataFrame ([mean, mode, median],  columns = col_name)
d_data

# insert column
d_data.insert(0,"Measures", row_head)
d_data
```

Out[41]:

	Measures	Displacement	Fuel_Tank_Capacity_litr	City_Mileage_km_litre	Highway_Mileage_km
0	mean	22225.037037	656.0	184.713704	219.90
1	mode	18098.000000	450.0	41.300000	54.60
2	median	13260.000000	405.0	85.520000	122.32

In [44]:

```
mean= []
SD =[]
CV=[]

# iterate  each column
for col in cr:
    col_mean= cr[col].mean()   #creating mean of each column
    mean.append(col_mean)             #storing the calculated mean in mean named folder
    col_std= cr[col].std()     #calculating standard deviation of each column
    SD.append(col_std)                #storing the calculated SDin SD name folder
    CV.append(col_std/col_mean*100)

row_head = ['mean', 'SD', 'CV']
col_name = ['Displacement','Fuel_Tank_Capacity_litr','City_Mileage_km_litre','Highway_Milea

# create dataframe of mean , median ,mode
d_data1 = pd.DataFrame ([mean, SD, CV],  columns = col_name)
d_data1


# insert column
d_data1.insert(0,"Measures", row_head)
d_data1
```

Out[44]:

	Measures	Displacement	Fuel_Tank_Capacity_litr	City_Mileage_km_litre	Highway_Mileage_km
0	mean	22225.037037	656.000000	184.713704	219.90
1	SD	19612.814181	630.529753	223.802065	255.17
2	CV	88.246486	96.117340	121.161593	116.03

In [21]:

```
su = cr[cr.Make == 'Suzuki']
to = cr[cr.Make == 'Mahindra']
ma = cr[cr.Make == 'Toyota']
```

In [55]:

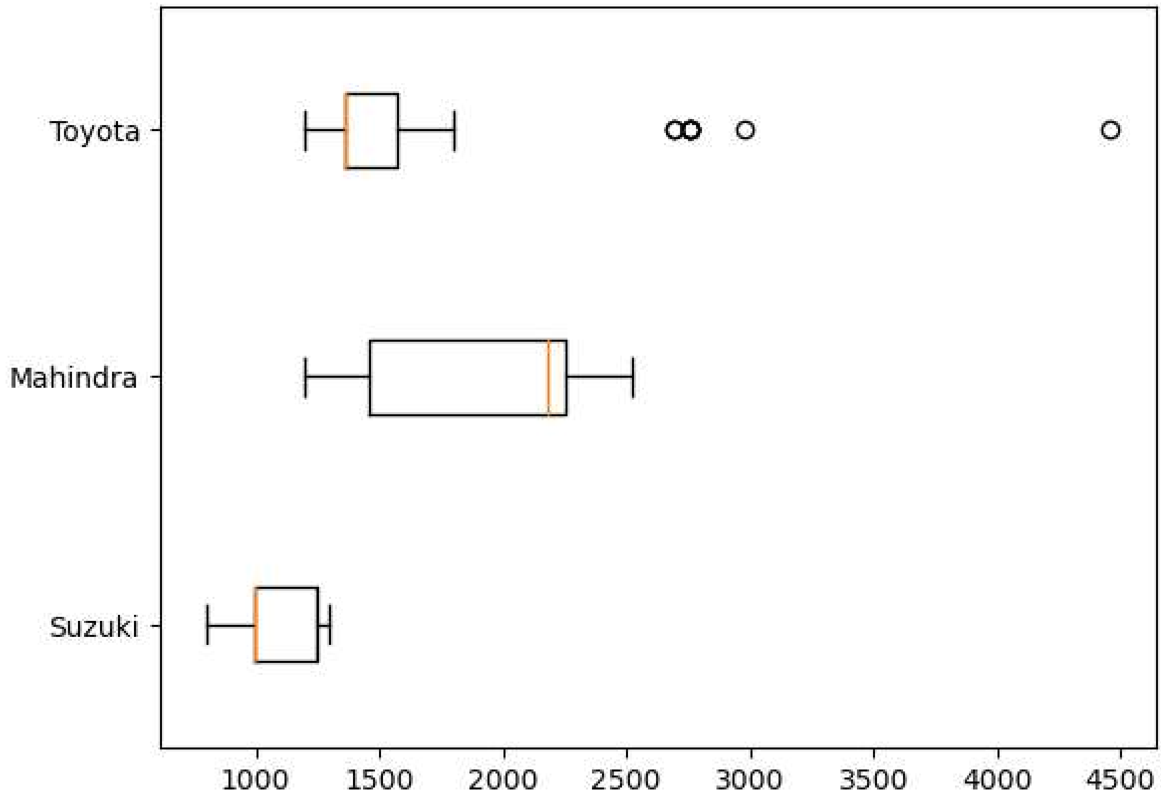
```
Suzuki_data = su['Displacement'].tolist()
Mahindra_data = to['Displacement'].tolist()
Toyota_data = ma['Displacement'].tolist()
```

In [56]:

```
plt.boxplot([Suzuki_data, Mahindra_data, Toyota_data], vert=0)
plt.yticks([1,2,3],['Suzuki', 'Mahindra', 'Toyota'])
```

Out[56]:

```
([<matplotlib.axis.YTick at 0x1ccd97de3d0>,
  <matplotlib.axis.YTick at 0x1ccd97c1d30>,
  <matplotlib.axis.YTick at 0x1ccd97f3880>],
 [Text(0, 1, 'Suzuki'), Text(0, 2, 'Mahindra'), Text(0, 3, 'Toyota')])
```



Task 2

In [54]:

```
import seaborn as sns
```

In [71]:

```
make = car.groupby(['Make'])['Displacement','Fuel_Tank_Capacity_litre', 'City_Mileage_km_li
```

In [72]:

```
corr = make[['Displacement','Fuel_Tank_Capacity_litre','City_Mileage_km_litre','Cylinders',
corr
```

Out[72]:

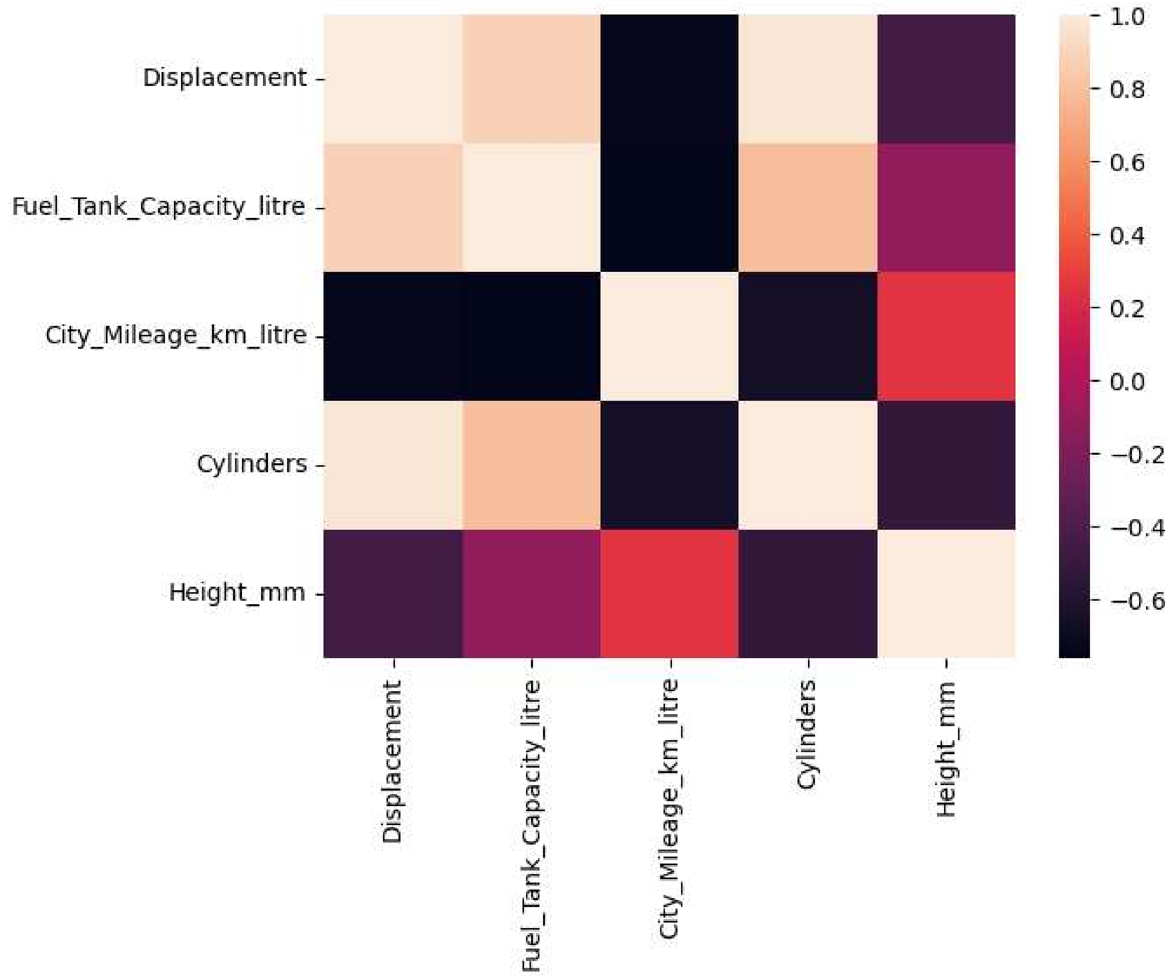
	Displacement	Fuel_Tank_Capacity_litre	City_Mileage_km_litre	Cylinder:
Displacement	1.000000	0.881811	-0.736779	0.97103:
Fuel_Tank_Capacity_litre	0.881811	1.000000	-0.758070	0.79227
City_Mileage_km_litre	-0.736779	-0.758070	1.000000	-0.66611
Cylinders	0.971032	0.792271	-0.666116	1.00000
Height_mm	-0.448780	-0.114598	0.252883	-0.52274

In [73]:

```
sns.heatmap(corr)
```

Out[73]:

<AxesSubplot:>



In [67]:

```
make['Displacement'].corr(make['City_Mileage_km_litre'])
```

Out[67]:

```
-0.7367789216015502
```

In [ ]:

```
make['Displacement'].corr(make[''])
```

In [68]:

```
make['Displacement'].corr(make['Cylinders'])
```

Out[68]:

```
0.9710315137022268
```

In [70]:

```
make['Displacement'].corr(make['Height_mm'])
```

Out[70]:

```
-0.4487804365986555
```

Conclusion : - From above heatmap and correalation coefficient its observable that number of cylinders have the highest correlation with mileage and followed by displacement