Raghav V. Rathi

C34

2113208

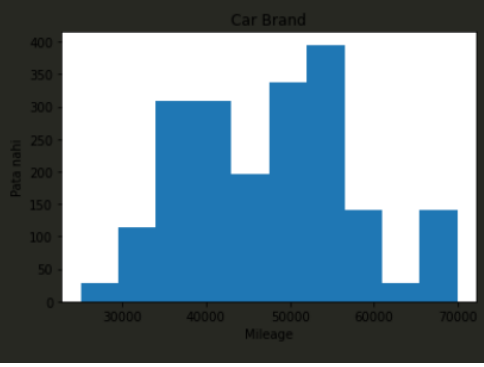
Practical 1

```
import pandas as pd
df = pd.read_csv("Car Data.csv") #I am working in Windows environment
#Reading the dataset in a dataframe using Pandas
print(df.head(3))
```

```
[3]
...
   Car ID   Brand  Model  Year   Color  Mileage  Price     Location
0       1  Toyota  Camry  2018   White    45000  18000  Los Angeles
1       2   Honda  Civic  2019    Blue    35000  16000     New York
2       3    Ford  Focus  2017  Silver    55000  14000      Chicago
```

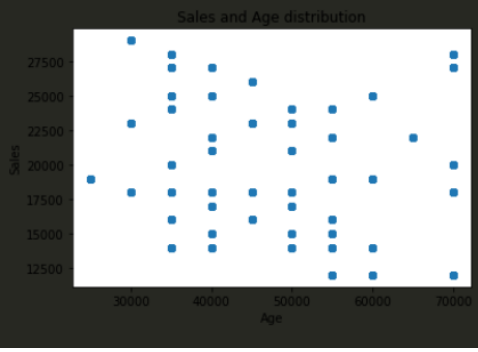
```
import pandas as pd
import matplotlib.pyplot as plt
df=pd.read_csv("Car Data.csv")
fig=plt.figure()
ax = fig.add_subplot(1,1,1)
ax.hist(df['Mileage'],bins = 10)
plt.title('Car Brand')
plt.xlabel('Mileage')
plt.ylabel('Pata nahi')
plt.show()
```

```
[12]
...
```



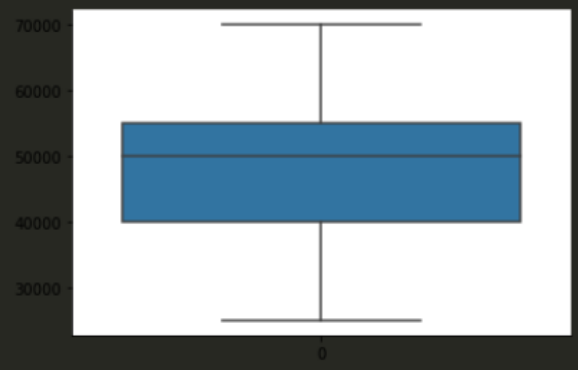
```
fig=plt.figure()

ax = fig.add_subplot(1,1,1)
ax.scatter(df['Mileage'],df['Price'])
plt.title('Mileage and Price distribution')
plt.xlabel('Mileage')
plt.ylabel('Price')
plt.show()
```

[13]




```
import seaborn as sns
sns.boxplot(df['Mileage'])
sns.despine()
```

[14]



```
df=pd.read_csv("Car Data.csv")
print(df)
test= df.groupby(['Brand','Price'])
test.size()
```

```
      Car ID      Brand      Model  Year   Color  Mileage  Price       Location
0          1     Toyota      Camry  2018   White    45000  18000    Los Angeles
1          2      Honda      Civic  2019    Blue    35000  16000       New York
2          3       Ford      Focus  2017  Silver    55000  14000        Chicago
3          4  Chevrolet      Cruze  2016     Red    60000  12000          Miami
4          5    Hyundai    Elantra  2018   Black    40000  15000  San Francisco
...      ...        ...        ...   ...     ...      ...    ...            ...
1995    1996    Hyundai   Palisade  2019  Silver    65000  22000  San Francisco
1996    1997     Toyota     Sienna  2018     Red    55000  16000         Dallas
1997    1998      Honda        Fit  2018    Gray    50000  14000        Atlanta
1998    1999       Ford     Fusion  2017   White    55000  19000        Phoenix
1999    2000  Chevrolet     Malibu  2016    Blue    30000  23000        Houston

[2000 rows x 8 columns]
```

```
Brand      Price
Chevrolet  12000    31
           14000    34
           15000    28
           16000    22
           17000    24
                    ..
Toyota     25000    23
           26000    10
           27000    24
           28000    12
           29000    13
Length: 85, dtype: int64
```

```
test= df.groupby(['Price'])
test.describe()
```

	Car ID								Year					Mileage						
	count	mean	std	min	25%	50%	75%	max	count	mean	...	75%	max	count	mean	std	min	25%	50%	75%	max
Price																					
12000	169.0	1003.550296	579.066787	4.0	501.00	998.0	1495.00	1992.0	169.0	2017.562130	...	2019.0	2020.0	169.0	58343.195266	5529.194526	55000.0	55000.0	55000.0	60000.0	70000.0
14000	170.0	1001.158824	582.460343	3.0	501.75	1000.5	1499.25	1998.0	170.0	2017.582353	...	2019.0	2020.0	170.0	50029.411765	9553.602419	35000.0	40000.0	52500.0	60000.0	60000.0
15000	141.0	999.397163	580.030638	5.0	502.00	999.0	1496.00	1993.0	141.0	2017.510638	...	2019.0	2020.0	141.0	48936.170213	4957.264173	40000.0	50000.0	50000.0	50000.0	55000.0
16000	114.0	1000.728070	586.638063	2.0	500.75	999.5	1498.25	1997.0	114.0	2017.359649	...	2019.0	2020.0	114.0	45000.000000	7164.316690	35000.0	37500.0	45000.0	52500.0	55000.0
17000	140.0	1001.500000	575.758536	29.0	518.25	1007.5	1496.75	1986.0	140.0	2017.557143	...	2019.0	2020.0	140.0	44000.000000	4916.570133	40000.0	40000.0	40000.0	50000.0	50000.0
18000	198.0	996.939394	581.168337	1.0	499.50	998.0	1496.50	1995.0	198.0	2017.530303	...	2019.0	2020.0	198.0	44924.242424	11969.475483	30000.0	35000.0	45000.0	50000.0	70000.0
19000	142.0	1003.683099	584.055768	6.0	504.25	1002.5	1500.75	1999.0	142.0	2017.042254	...	2018.0	2020.0	142.0	49859.154930	12784.959886	25000.0	55000.0	55000.0	55000.0	60000.0
20000	84.0	998.500000	577.242312	15.0	504.25	993.5	1482.75	1972.0	84.0	2017.619048	...	2019.0	2020.0	84.0	46666.666667	16598.253162	35000.0	35000.0	35000.0	70000.0	70000.0
21000	112.0	996.500000	576.469747	13.0	504.75	996.5	1488.25	1980.0	112.0	2017.214286	...	2019.0	2020.0	112.0	45000.000000	5022.472023	40000.0	40000.0	45000.0	50000.0	50000.0
22000	141.0	1003.787234	580.020416	8.0	505.00	1002.0	1499.00	1996.0	141.0	2017.354610	...	2019.0	2020.0	141.0	54078.014184	8053.456154	40000.0	55000.0	55000.0	55000.0	65000.0
23000	113.0	1002.407080	581.634959	12.0	509.00	1006.0	1503.00	2000.0	113.0	2017.601770	...	2019.0	2020.0	113.0	43628.318584	8298.229900	30000.0	30000.0	45000.0	50000.0	50000.0
24000	84.0	1001.166667	577.254779	17.0	506.25	995.5	1484.75	1974.0	84.0	2017.630952	...	2019.0	2020.0	84.0	46666.666667	8549.407551	35000.0	35000.0	50000.0	55000.0	55000.0
25000	112.0	993.500000	576.380474	16.0	506.25	996.5	1486.75	1977.0	112.0	2017.758929	...	2019.0	2020.0	112.0	42500.000000	10354.091327	35000.0	35000.0	37500.0	45000.0	60000.0
26000	56.0	1002.500000	579.062157	24.0	513.25	1002.5	1491.75	1981.0	56.0	2017.250000	...	2018.0	2020.0	56.0	45000.000000	0.000000	45000.0	45000.0	45000.0	45000.0	45000.0
27000	112.0	1001.500000	576.438678	21.0	511.50	1002.0	1492.50	1983.0	112.0	2017.232143	...	2019.0	2020.0	112.0	45000.000000	14642.896381	35000.0	35000.0	37500.0	47500.0	70000.0
28000	56.0	998.500000	579.098202	19.0	508.75	998.5	1488.25	1978.0	56.0	2017.178571	...	2019.0	2020.0	56.0	52500.000000	17658.374269	35000.0	35000.0	52500.0	70000.0	70000.0
29000	56.0	1003.500000	579.062157	25.0	514.25	1003.5	1492.75	1982.0	56.0	2017.553571	...	2019.0	2020.0	56.0	30000.000000	0.000000	30000.0	30000.0	30000.0	30000.0	30000.0

17 rows × 24 columns