

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

df = pd.read_csv('cars.csv')
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 100 entries, 0 to 99
Data columns (total 13 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Car_ID                100 non-null   int64
1   Brand                 100 non-null   object
2   Model                 100 non-null   object
3   Year                  100 non-null   int64
4   Kilometers_Driven     100 non-null   int64
5   Fuel_Type             100 non-null   object
6   Transmission          100 non-null   object
7   Owner_Type            100 non-null   object
8   Mileage               100 non-null   int64
9   Engine                100 non-null   int64
10  Power                 100 non-null   int64
11  Seats                 100 non-null   int64
12  Price                 100 non-null   int64
dtypes: int64(8), object(5)
memory usage: 10.3+ KB
```

```
In [2]: df.head()
```

Out[2]:

Kilometers_Driven	Fuel_Type	Transmission	Owner_Type	Mileage	Engine	Power	Seats	Price
50000	Petrol	Manual	First	15	1498	108	5	80000
40000	Petrol	Automatic	Second	17	1597	140	5	100000
20000	Petrol	Automatic	First	10	4951	395	4	250000
30000	Diesel	Manual	Third	23	1248	74	5	60000
60000	Diesel	Automatic	Second	18	1999	194	5	85000

data preprocessing

```
In [3]: df.isnull().any()
```

```
Out[3]: Car_ID           False
Brand             False
Model             False
Year             False
Kilometers_Driven False
Fuel_Type         False
Transmission      False
Owner_Type        False
Mileage           False
Engine            False
Power             False
Seats            False
Price            False
dtype: bool
```

```
In [4]: df.duplicated().any()
```

```
Out[4]: False
```

What are the different car brands in the dataset?

```
In [5]: unique_brands = df['Brand'].unique()
unique_brands
```

```
Out[5]: array(['Toyota', 'Honda', 'Ford', 'Maruti', 'Hyundai', 'Tata', 'Mahindra',
               'Volkswagen', 'Audi', 'BMW', 'Mercedes'], dtype=object)
```

```
In [ ]: SELECT DISTINCT Brand FROM dataset_table;
```

How many cars are there in the dataset?

```
In [6]: total_cars = len(df)
total_cars
```

```
Out[6]: 100
```

```
In [ ]: SELECT COUNT(*) AS total_cars FROM dataset_table;
```

What is the average price of cars in the dataset?

```
In [7]: average_price = df['Price'].mean()
average_price
```

```
Out[7]: 1574000.0
```

```
In [ ]: SELECT AVG(Price) AS average_price FROM dataset_table;
```

Which car has the highest mileage?

```
In [8]: max_mileage_car = df.loc[df['Mileage'].idxmax()]
max_mileage_car[['Brand', 'Model']]
```

```
Out[8]: Brand      Honda
Model      City
Name: 15, dtype: object
```

```
In [ ]: SELECT Brand, Model FROM dataset_table ORDER BY Mileage DESC LIMIT 1;
```

How many cars have automatic transmission (Transmission = 'Automatic')?

```
In [10]: automatic_cars_count = df[df['Transmission'] == 'Automatic']  
automatic_cars_count
```

Out[10]:

	Car_ID	Brand	Model	Year	Kilometers_Driven	Fuel_Type	Transmission	Owner
1	2	Honda	Civic	2019	40000	Petrol	Automatic	S
2	3	Ford	Mustang	2017	20000	Petrol	Automatic	
4	5	Hyundai	Sonata	2016	60000	Diesel	Automatic	S
6	7	Mahindra	Scorpio	2018	45000	Diesel	Automatic	S
7	8	Volkswagen	Polo	2020	25000	Petrol	Automatic	
8	9	Audi	A4	2017	30000	Diesel	Automatic	
9	10	BMW	X1	2019	20000	Diesel	Automatic	S
10	11	Mercedes	C-Class	2018	28000	Petrol	Automatic	
11	12	Ford	Endeavour	2017	35000	Diesel	Automatic	S
13	14	Tata	Harrier	2018	40000	Diesel	Automatic	
16	17	Volkswagen	Tiguan	2018	32000	Diesel	Automatic	
17	18	Audi	Q3	2016	38000	Petrol	Automatic	S
18	19	BMW	5 Series	2019	24000	Diesel	Automatic	
19	20	Mercedes	GLC	2017	26000	Petrol	Automatic	S
22	23	Hyundai	Verna	2019	26000	Diesel	Automatic	S
25	26	Volkswagen	Passat	2017	32000	Diesel	Automatic	S
26	27	Audi	A6	2018	28000	Petrol	Automatic	
27	28	BMW	X3	2019	22000	Diesel	Automatic	S
28	29	Mercedes	E-Class	2017	30000	Diesel	Automatic	
29	30	Toyota	Fortuner	2018	38000	Diesel	Automatic	S
31	32	Hyundai	Elantra	2017	32000	Diesel	Automatic	S
35	36	Volkswagen	Ameo	2020	15000	Petrol	Automatic	
36	37	Audi	A3	2017	38000	Petrol	Automatic	S
37	38	BMW	7 Series	2019	22000	Diesel	Automatic	
38	39	Mercedes	GLE	2018	26000	Petrol	Automatic	S
43	44	Maruti	S-Cross	2020	15000	Petrol	Automatic	S
45	46	Volkswagen	T-Roc	2019	22000	Petrol	Automatic	S
46	47	Audi	Q7	2017	30000	Diesel	Automatic	
47	48	BMW	X5	2018	28000	Petrol	Automatic	S
48	49	Mercedes	GLA	2019	24000	Diesel	Automatic	
49	50	Toyota	Camry	2016	38000	Petrol	Automatic	S
50	51	Ford	Mustang	2019	22000	Petrol	Automatic	
55	56	Audi	A5	2018	28000	Diesel	Automatic	
56	57	BMW	3 Series	2020	15000	Petrol	Automatic	S
57	58	Mercedes	C-Class	2019	22000	Diesel	Automatic	
60	61	Hyundai	Verna	2019	24000	Petrol	Automatic	S
63	64	Volkswagen	Passat	2017	32000	Diesel	Automatic	S
64	65	Audi	A6	2018	28000	Petrol	Automatic	

	Car_ID	Brand	Model	Year	Kilometers_Driven	Fuel_Type	Transmission	Owner
65	66	BMW	X3	2019	22000	Diesel	Automatic	S
66	67	Mercedes	E-Class	2017	30000	Diesel	Automatic	
67	68	Toyota	Fortuner	2018	38000	Diesel	Automatic	S
69	70	Hyundai	Elantra	2017	32000	Diesel	Automatic	S
73	74	Volkswagen	Ameo	2020	15000	Petrol	Automatic	
74	75	Audi	A3	2017	38000	Petrol	Automatic	S
75	76	BMW	7 Series	2019	22000	Diesel	Automatic	
76	77	Mercedes	GLE	2018	26000	Petrol	Automatic	S
81	82	Maruti	S-Cross	2020	15000	Petrol	Automatic	S
83	84	Volkswagen	T-Roc	2019	22000	Petrol	Automatic	S
84	85	Audi	Q7	2017	30000	Diesel	Automatic	
85	86	BMW	X5	2018	28000	Petrol	Automatic	S
86	87	Mercedes	GLA	2019	24000	Diesel	Automatic	
87	88	Toyota	Camry	2016	38000	Petrol	Automatic	S
88	89	Ford	Mustang	2019	22000	Petrol	Automatic	
93	94	Audi	A5	2018	28000	Diesel	Automatic	
94	95	BMW	3 Series	2020	15000	Petrol	Automatic	S
95	96	Mercedes	C-Class	2019	22000	Diesel	Automatic	
98	99	Hyundai	Verna	2019	24000	Petrol	Automatic	S

```
In [15]: automatic_cars_count = df[df['Transmission'] == 'Automatic'].shape
automatic_cars_count
```

```
Out[15]: (57, 13)
```

```
In [16]: automatic_cars_count = df[df['Transmission'] == 'Automatic'].shape[0]
automatic_cars_count
```

```
Out[16]: 57
```

```
In [ ]: SELECT COUNT(*) FROM dataset_table WHERE Transmission = 'Automatic';
```

What is the average kilometers driven per year for each brand?

```
In [17]: avg_kms_per_year = df.groupby('Brand')['Kilometers_Driven'].mean()  
avg_kms_per_year
```

```
Out[17]: Brand  
Audi          31600.000000  
BMW           21800.000000  
Ford          26727.272727  
Honda         32333.333333  
Hyundai       30545.454545  
Mahindra      23400.000000  
Maruti        21000.000000  
Mercedes      25800.000000  
Tata          30272.727273  
Toyota        36400.000000  
Volkswagen    25900.000000  
Name: Kilometers_Driven, dtype: float64
```

```
In [ ]: SELECT Brand, AVG(Kilometers_Driven) AS avg_kms_per_year FROM dataset_table
```

Which fuel type is most common among the cars?

```
In [19]: common_fuel_type = df['Fuel_Type'].mode()[0]  
common_fuel_type
```

```
Out[19]: 'Petrol'
```

```
In [ ]: SELECT Fuel_Type, COUNT(*) AS count FROM dataset_table GROUP BY Fuel_Type OR
```

```
In [ ]:
```

What is the maximum power output of a car in the dataset?

```
In [20]: max_power = df['Power'].max()  
max_power
```

```
Out[20]: 396
```

```
In [ ]: SELECT MAX(Power) AS max_power FROM dataset_table;
```

How many cars were manufactured each year?

```
In [21]: cars_per_year = df['Year'].value_counts()
cars_per_year
```

```
Out[21]: Year
2018      28
2019      28
2017      21
2020      17
2016       4
2021       2
Name: count, dtype: int64
```

```
In [ ]: SELECT Year, COUNT(*) AS car_count FROM dataset_table GROUP BY Year;
```

What is the average number of seats for each brand?

```
In [24]: avg_seats_per_brand = df.groupby('Brand')['Seats'].mean().astype(int)
avg_seats_per_brand
```

```
Out[24]: Brand
Audi      5
BMW       5
Ford      4
Honda     5
Hyundai   5
Mahindra  5
Maruti    5
Mercedes  5
Tata      5
Toyota    6
Volkswagen 5
Name: Seats, dtype: int32
```

```
In [ ]: SELECT Brand, AVG(Seats) AS avg_seats FROM dataset_table GROUP BY Brand;
```

Which car has the highest engine capacity?



```
In [25]: max_engine_car = df.loc[df['Engine'].idxmax()]
print(max_engine_car[['Brand', 'Model']])
```

```
Brand      Ford
Model      Mustang
Name: 2, dtype: object
```

```
In [ ]: SELECT Brand, Model FROM dataset_table ORDER BY Engine DESC LIMIT 1;
```


What is the most common owner type among the cars?

```
In [26]: common_owner_type = df['Owner_Type'].mode()[0]
print(common_owner_type)
```

First

```
In [ ]: SELECT Owner_Type, COUNT(*) AS count FROM dataset_table GROUP BY Owner_Type
```



What is the range of prices for cars in the dataset?

```
In [27]: price_range = (df['Price'].min(), df['Price'].max())
price_range
```

Out[27]: (450000, 4000000)

```
In [ ]: SELECT MIN(Price) AS min_price, MAX(Price) AS max_price FROM dataset_table;
```

How many unique car models are there in the dataset?

```
In [28]: unique_models_count = df['Model'].nunique()
unique_models_count
```

Out[28]: 58

```
In [ ]: SELECT COUNT(DISTINCT Model) AS unique_models_count FROM dataset_table;
```

What is the average mileage for each fuel type?

```
In [29]: avg_mileage_per_fuel_type = df.groupby('Fuel_Type')['Mileage'].mean()
avg_mileage_per_fuel_type
```

Out[29]: Fuel_Type
Diesel 17.416667
Petrol 17.019231
Name: Mileage, dtype: float64

In []: `SELECT Fuel_Type, AVG(Mileage) AS avg_mileage FROM dataset_table GROUP BY Fu`



In []: