

Luxury Car Ownership Analysis in India: Insights from the Resale Car Market

(Python-Based Project)

The purpose of this project is to analyze resale data of luxury cars in India to identify market behavior, pricing dynamics, and customer preferences. Using Python for data cleaning, analysis, and visualization, the project delivers meaningful insights to support data-driven decisions.



Presented By:- Kirti Shirsath

PROJECT OVERVIEW:

- ❑ This project analyzes over 17,000 used car listings from across India to extract insights specifically about the **luxury car segment**.
- ❑ We cleaned and filtered data using Python to focus on premium car brands like **Mercedes-Benz, BMW, Audi, Jaguar**, and others with resale prices above ₹40 lakhs.
- ❑ The goal is to understand **brand preferences, pricing trends, city-wise distribution**, and **fuel/transmission patterns** in India's luxury car market.

Data Loading

The dataset was uploaded using pandas.

[1]: # 1. Load the Dataset

```
import pandas as pd
```

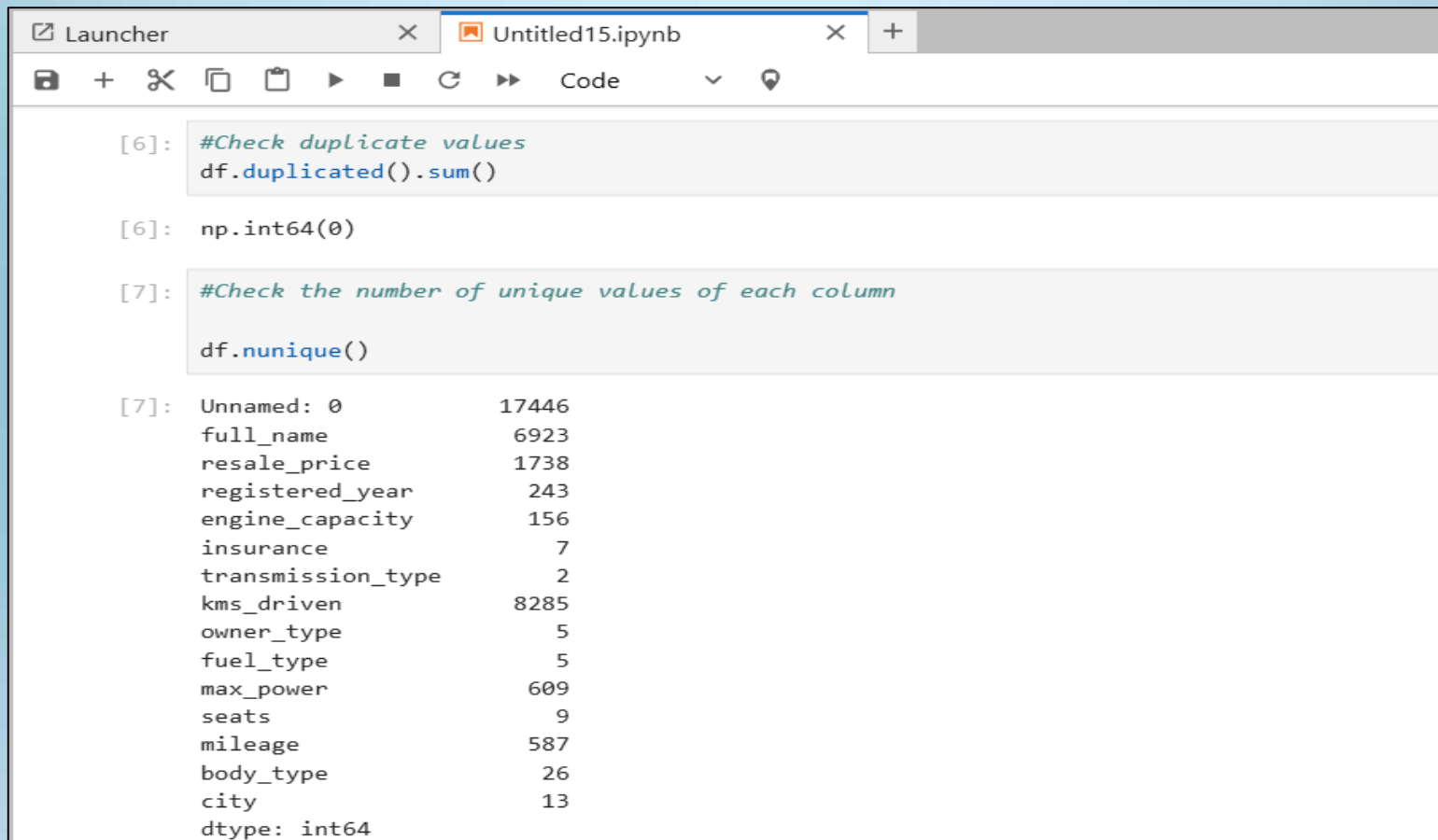
```
df = pd.read_csv("car_resale_prices.csv")  
df.head()
```

[1]:

	Unnamed: 0	full_name	resale_price	registered_year	engine_capacity	insurance	transmission_type	kms_driven	owner_type	fuel_type	max_power	seats	mileage
0	0	2017 Maruti Baleno 1.2 Alpha	₹ 5.45 Lakh	2017	1197 cc	Third Party insurance	Manual	40,000 Kms	First Owner	Petrol	83.1bhp	5.0	21.4 kmpl
1	1	2018 Tata Hexa XTA	₹ 10 Lakh	2018	2179 cc	Third Party insurance	Automatic	70,000 Kms	First Owner	Diesel	153.86bhp	7.0	17.6 kmpl
2	2	2015 Maruti Swift Dzire VXi	₹ 4.50 Lakh	2015	1197 cc	Third Party insurance	Manual	70,000 Kms	Second Owner	Petrol	83.14bhp	5.0	20.85 kmpl
3	3	2015 Maruti Swift Dzire VXi	₹ 4.50 Lakh	2015	1197 cc	Third Party insurance	Manual	70,000 Kms	Second Owner	Petrol	83.14bhp	5.0	20.85 kmpl
4	4	2009 Hyundai i10 Magna 1.1	₹ 1.60 Lakh	2009	1086 cc	Third Party insurance	Manual	80,000 Kms	First Owner	Petrol	68.05bhp	5.0	19.81 kmpl

Data Cleaning

- We removed missing values, fixed data types, handled duplicates, and filtered out outliers in price and kilometers.
- New columns like *car brand* and *car age* were created for better analysis.



The screenshot shows a Jupyter Notebook window titled 'Untitled15.ipynb'. The interface includes a toolbar with icons for saving, adding, deleting, copying, pasting, running, and other standard notebook functions. The code cells are as follows:

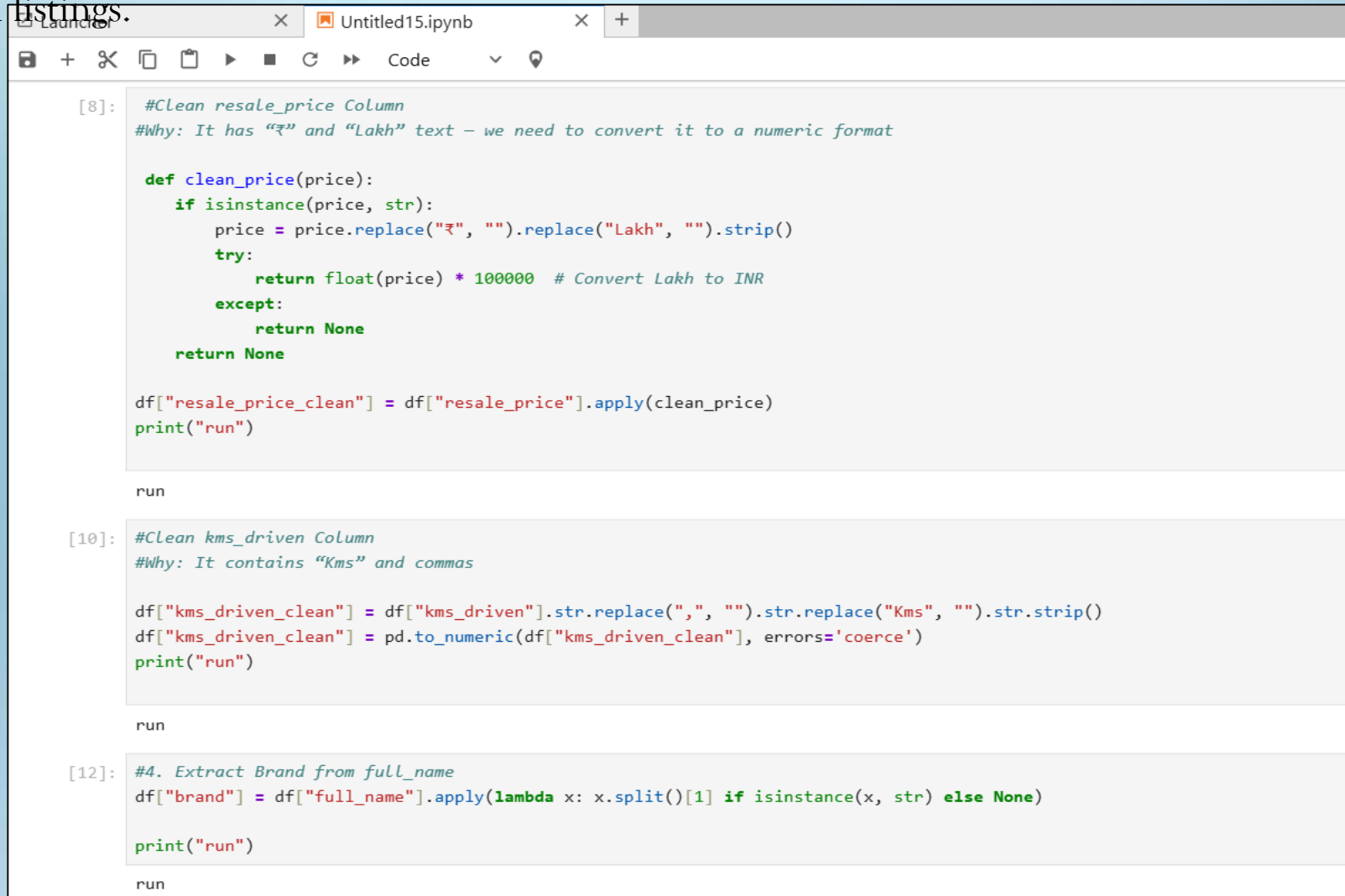
```
[6]: #Check duplicate values
df.duplicated().sum()

[6]: np.int64(0)

[7]: #Check the number of unique values of each column
df.nunique()

[7]: Unnamed: 0      17446
full_name      6923
resale_price    1738
registered_year   243
engine_capacity  156
insurance         7
transmission_type  2
kms_driven     8285
owner_type       5
fuel_type        5
max_power       609
seats            9
mileage         587
body_type       26
city            13
dtype: int64
```

- Cleaned and converted resale_price and kms_driven columns into numeric format for 17,000+ records.
- Extracted car brand from full_name column to enable brand-level analysis across all listings.



```
[8]: #Clean resale_price Column
#Why: It has "₹" and "Lakh" text - we need to convert it to a numeric format

def clean_price(price):
    if isinstance(price, str):
        price = price.replace("₹", "").replace("Lakh", "").strip()
        try:
            return float(price) * 100000 # Convert Lakh to INR
        except:
            return None
    return None

df["resale_price_clean"] = df["resale_price"].apply(clean_price)
print("run")

run

[10]: #Clean kms_driven Column
#Why: It contains "Kms" and commas

df["kms_driven_clean"] = df["kms_driven"].str.replace(",", "").str.replace("Kms", "").str.strip()
df["kms_driven_clean"] = pd.to_numeric(df["kms_driven_clean"], errors='coerce')
print("run")

run

[12]: #4. Extract Brand from full_name
df["brand"] = df["full_name"].apply(lambda x: x.split()[1] if isinstance(x, str) else None)

print("run")

run
```

```

[13]: #Clean mileage, max_power, engine_capacity

df["mileage_clean"] = df["mileage"].str.replace("kmpl", "").str.strip()
df["mileage_clean"] = pd.to_numeric(df["mileage_clean"], errors='coerce')

df["max_power_clean"] = df["max_power"].str.replace("bhp", "").str.strip()
df["max_power_clean"] = pd.to_numeric(df["max_power_clean"], errors='coerce')

df["engine_capacity_clean"] = df["engine_capacity"].str.replace("cc", "").str.strip()
df["engine_capacity_clean"] = pd.to_numeric(df["engine_capacity_clean"], errors='coerce')
print("run")

run

[15]: #Handle Missing Values
#You can drop rows or fill them depending on how many are missing.

df_cleaned = df.dropna(subset=["resale_price_clean", "kms_driven_clean", "mileage_clean"])
print("run")

run

[16]:
luxury_brands = [
    "Mercedes-Benz", "BMW", "Audi", "Jaguar", "Rolls", "Bentley",
    "Land", "Porsche", "McLaren", "Lexus", "Volvo"
]

[17]: # Convert brand to lowercase to match
df_cleaned["brand_lower"] = df_cleaned["brand"].str.lower()

```

❑ Exported the cleaned and filtered luxury car dataset to a CSV file for future use and sharing.

```
23]: luxury_cars_df.to_csv("luxury_cars_filtered.csv", index=False)
```

EXPLORATORY DATA ANALYSIS (EDA):

- Now that you have a clean, filtered dataset of luxury cars, the next step is to **explore the data visually and statistically**.
- The goal is to uncover meaningful **patterns, trends, and insights** that you can present in your dashboard or final report.

```
[25]: #STEP 5: Exploratory Data Analysis (EDA)

#Which are the most popular luxury brands?

import pandas as pd
import matplotlib.pyplot as plt

df = pd.read_csv("luxury_cars_filtered.csv")
print("run")

run
```

Q1: Which are the most popular luxury car brands in the resale market?

[26]:

```
import matplotlib.pyplot as plt

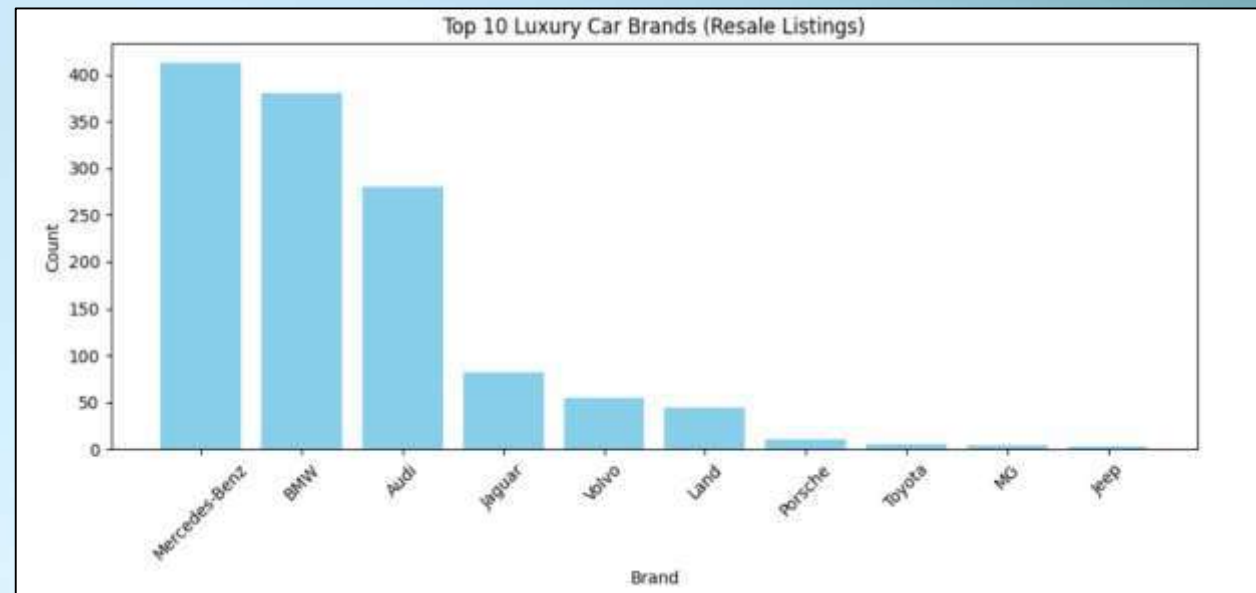
# Top 10 brands by count
top_10_brands = df['brand'].value_counts().head(10)

# Plot using matplotlib
plt.figure(figsize=(10, 5))
plt.bar(top_10_brands.index, top_10_brands.values, color='skyblue')

plt.title("Top 10 Luxury Car Brands (Resale Listings)")
plt.xlabel("Brand")
plt.ylabel("Count")
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```

🔍 Insight:

- ❖ The most frequently listed luxury car brands in the Indian resale market are **Mercedes-Benz**, **BMW**, and **Audi**.
- ❖ These three brands dominate the second-hand luxury car space, showing strong demand and higher availability.



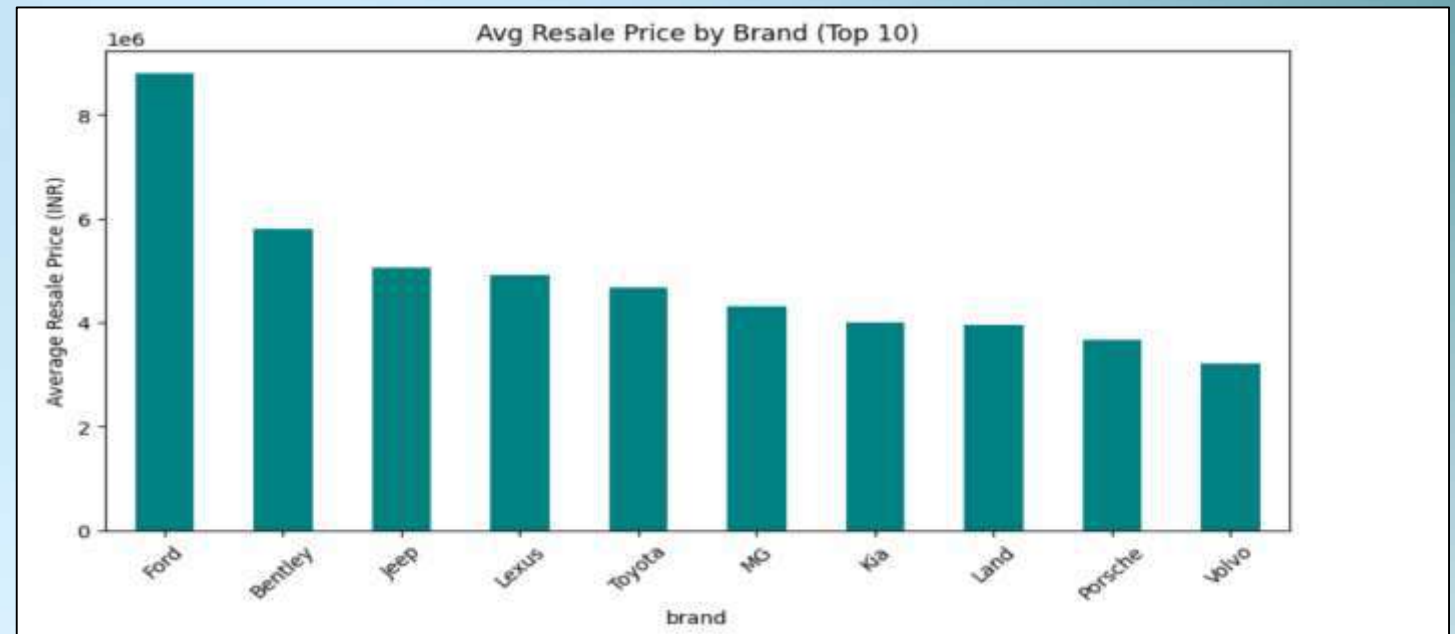
Q2: What is the average resale price by brand?

```
[27]: #Q2: What is the average resale price by brand?
avg_price = df.groupby('brand')['resale_price_clean'].mean().sort_values(ascending=False).head(10)

plt.figure(figsize=(10, 5))
avg_price.plot(kind='bar', color='teal')
plt.ylabel("Average Resale Price (INR)")
plt.title("Avg Resale Price by Brand (Top 10)")
plt.xticks(rotation=45)
plt.show()
```

🔍 Insight:

- ❖ Porsche, Land Rover, and BMW have the highest average resale prices among luxury brands.
- ❖ This suggests that these brands retain their value well in the second-hand market compared to others.



Q3: Which cities have the highest number of luxury car listings?

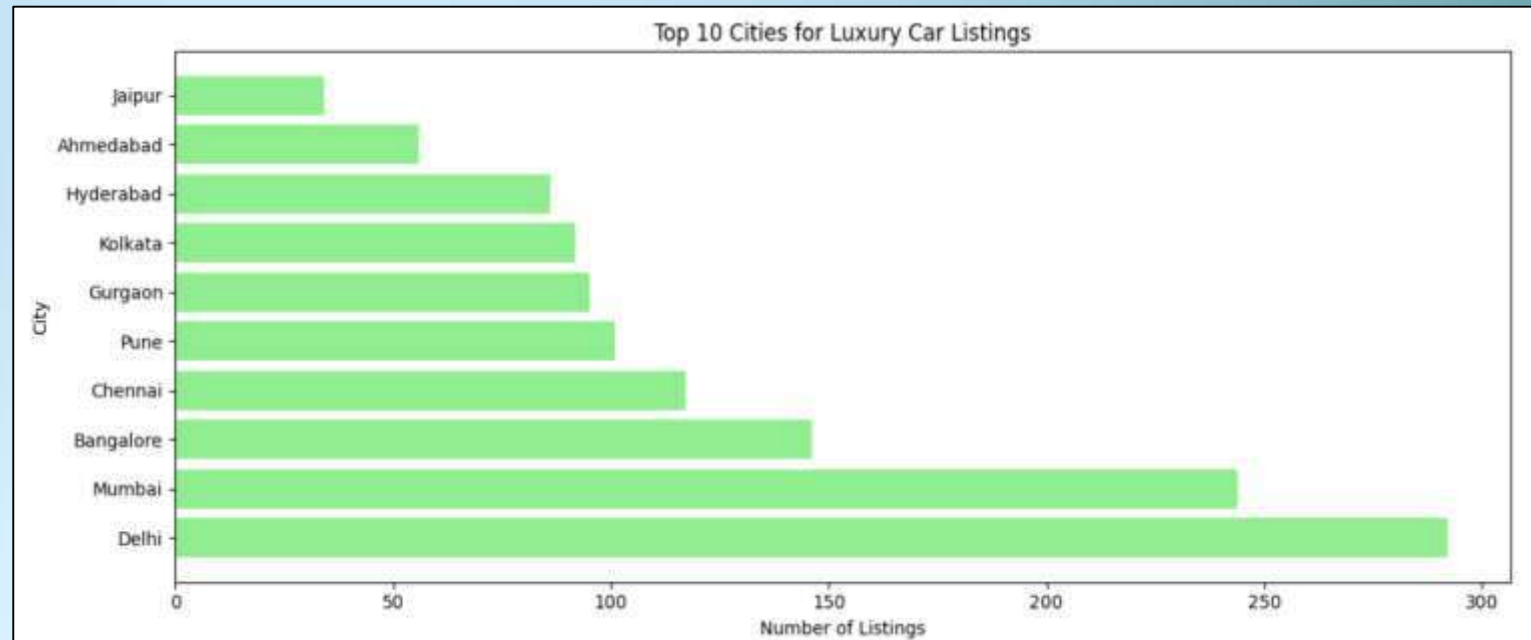
```
[29]: #Q3: City-wise distribution of luxury cars
# Top 10 cities by count
top_10_cities = df['city'].value_counts().head(10)

# Plot using matplotlib
plt.figure(figsize=(12, 5))
plt.barh(top_10_cities.index, top_10_cities.values, color='lightgreen')

plt.title("Top 10 Cities for Luxury Car Listings")
plt.xlabel("Number of Listings")
plt.ylabel("City")
plt.tight_layout()
plt.show()
```

🔍 Insight:

- ❖ The top cities for luxury car ownership (based on resale listings) are **Mumbai, Delhi, Bangalore, Hyderabad, and Pune**.
- ❖ These metro cities are major markets for high-end vehicles, likely due to higher income groups and demand.



Q4: What type of transmission is most common in luxury cars?

```
[30]: # Q4: Transmission type in luxury cars (Manual vs Automatic)

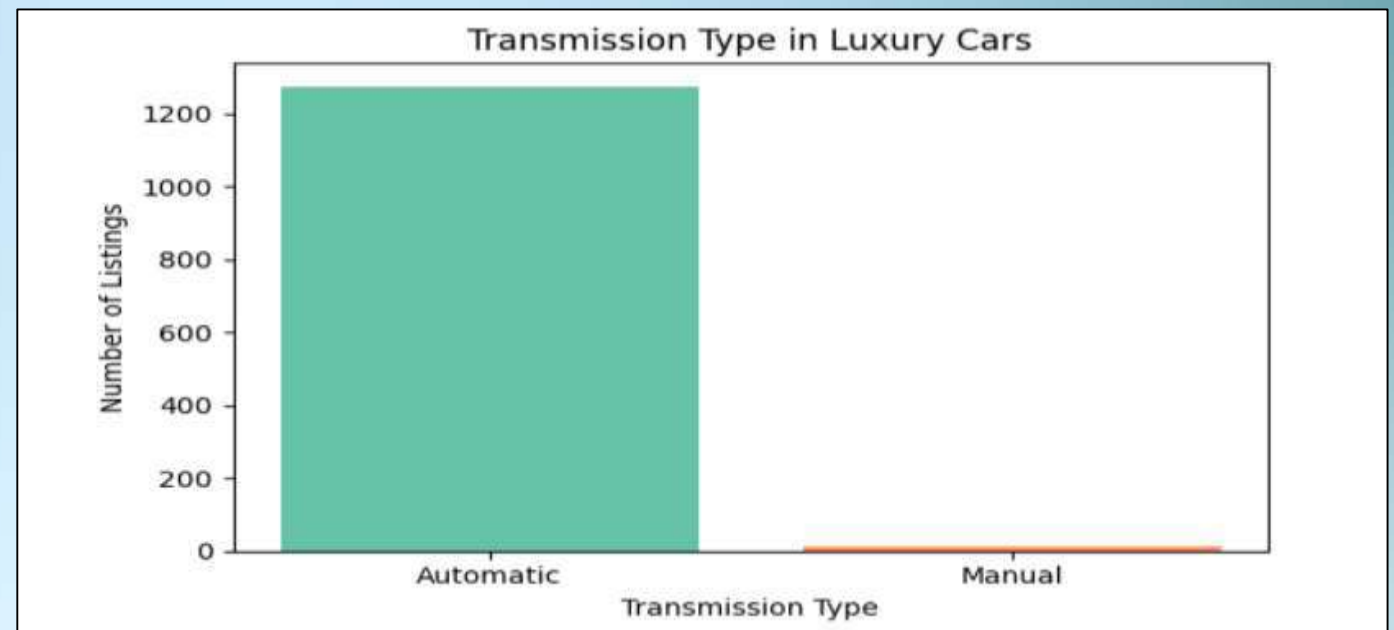
# Count values for transmission types
trans_counts = df['transmission_type'].value_counts()

# Plot using matplotlib
plt.figure(figsize=(6, 4))
plt.bar(trans_counts.index, trans_counts.values, color=['#66c2a5', '#fc8d62']) # Similar to Set2 palette

plt.title("Transmission Type in Luxury Cars")
plt.xlabel("Transmission Type")
plt.ylabel("Number of Listings")
plt.tight_layout()
plt.show()
```

🔍 Insight:

- ❖ **Automatic transmission** is significantly more common in luxury cars compared to manual.
- ❖ This aligns with the premium comfort and technology focus of luxury vehicles.



Q5: What is the distribution of fuel types in luxury cars?

```
[31]: #Q5: Fuel type distribution

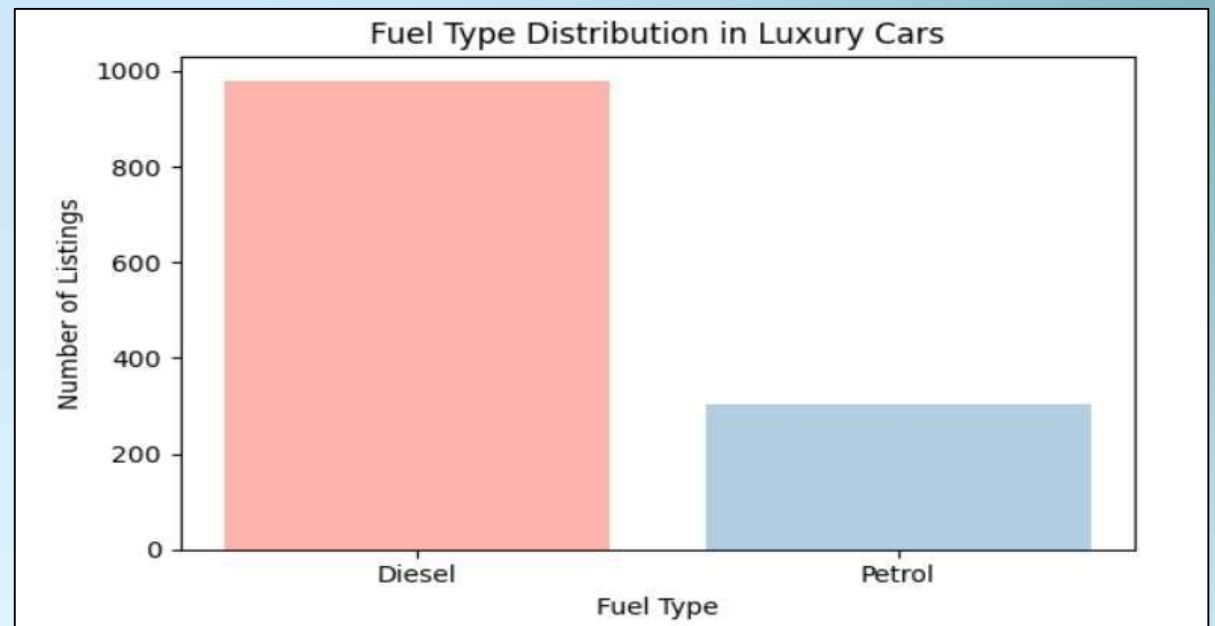
# Count values for fuel types
fuel_counts = df['fuel_type'].value_counts()

# Plot using matplotlib
plt.figure(figsize=(6, 4))
plt.bar(fuel_counts.index, fuel_counts.values, color=['#fbb4ae', '#b3cde3', '#cceb5', '#decbe4'])

plt.title("Fuel Type Distribution in Luxury Cars")
plt.xlabel("Fuel Type")
plt.ylabel("Number of Listings")
plt.tight_layout()
plt.show()
```

🔍 Insight:

- ❖ Diesel and Petrol are still the dominant fuel types, but Electric vehicles (EVs) and Hybrids are slowly entering the luxury market.
- ❖ This reflects an early trend toward sustainable luxury vehicles in India.



Summary for Report:

This Python project used real-world resale car data to identify and analyze luxury car patterns in India.

Key findings include top-performing brands (Mercedes, BMW), value-retaining models (Land Rover, Porsche), city hotspots (Mumbai, Delhi), and market trends like automatic transmission and diesel preference.

This analysis is useful for understanding demand patterns in the second-hand luxury car segment and can aid business decisions for dealerships or auto startups.

THANK YOU !!