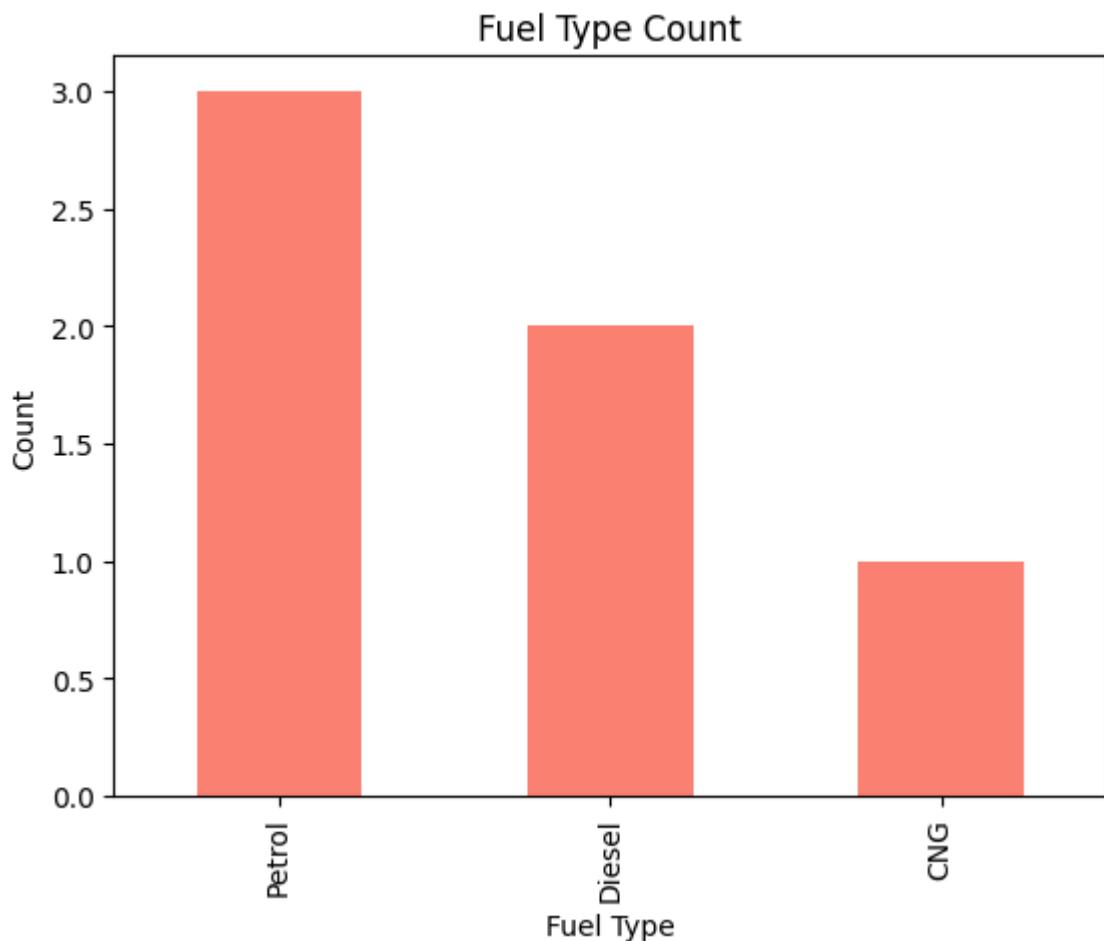


```
import pandas as pd  
import matplotlib.pyplot as plt  
  
# Sample DataFrame (replace this with your actual data loading step)  
data = pd.DataFrame({  
    'Fuel_Type': ['Petrol', 'Diesel', 'Petrol', 'CNG', 'Diesel', 'Petrol']  
})  
  
data['Fuel_Type'].value_counts().plot(kind='bar', color='salmon')  
plt.title('Fuel Type Count')  
plt.xlabel('Fuel Type')  
plt.ylabel('Count') # You might want to replace 'Mileage' with 'Count' here  
plt.show()
```



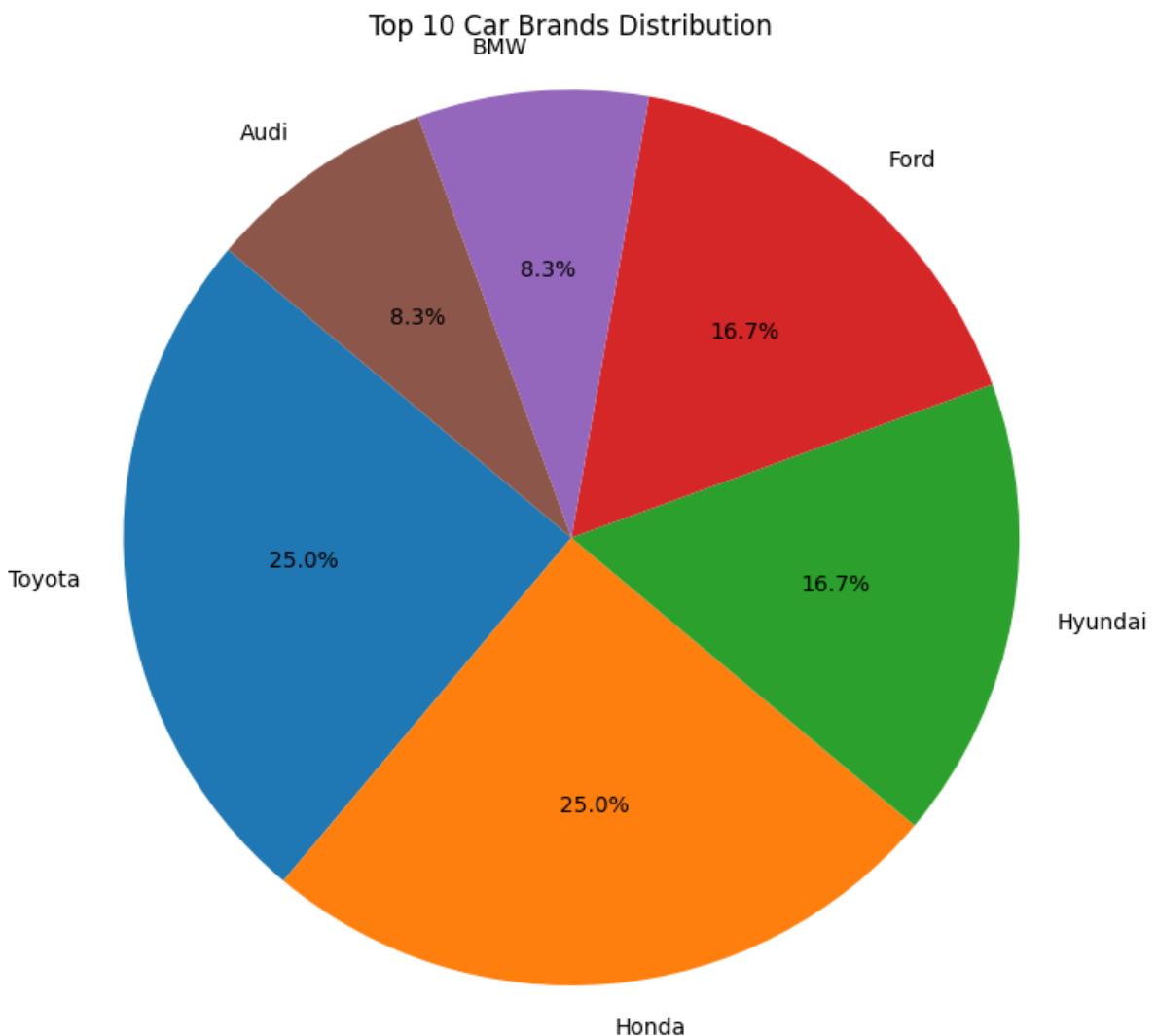
```
import pandas as pd
import matplotlib.pyplot as plt

# Example DataFrame with car brands (replace with your actual dataset)
data = pd.DataFrame({
    'Car_Make': ['Toyota', 'Hyundai', 'Honda', 'Toyota', 'Ford', 'Honda', 'Hyundai', 'Toyota', 'BMW',
    'Ford', 'Audi', 'Honda']
})

# Strip any unwanted whitespace from column names (just in case)
data.columns = data.columns.str.strip()

# Modify 'Car_Make' to your actual column name, if different
brand_counts = data['Car_Make'].value_counts().head(10)

plt.figure(figsize=(8, 8))
plt.pie(brand_counts, labels=brand_counts.index, autopct='%.1f%%', startangle=140)
plt.title('Top 10 Car Brands Distribution')
plt.axis('equal')
plt.show()
```



```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

df = pd.read_csv('used_cars_data.csv')

df_encoded = df.copy()
df_encoded['Fuel_Type_Encoded'] = df['Fuel_Type'].astype('category').cat.codes
```

```

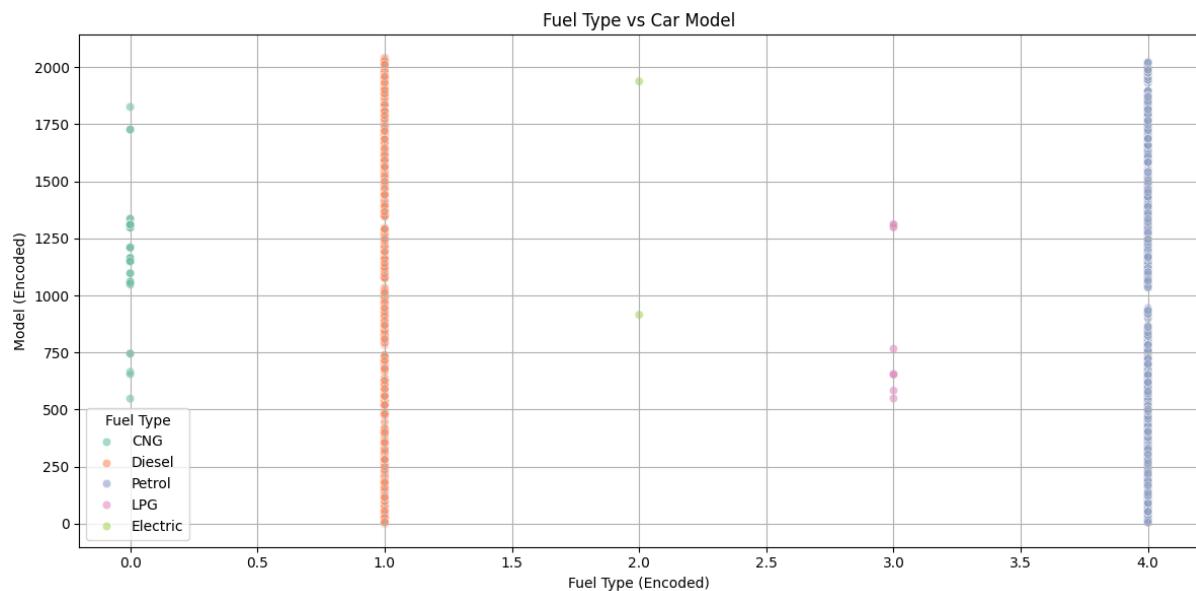
df_encoded['Name_Encoded'] = df['Name'].astype('category').cat.codes

plt.figure(figsize=(12, 6))

sns.scatterplot(data=df_encoded, x='Fuel_Type_Encoded', y='Name_Encoded', hue='Fuel_Type',
palette='Set2', alpha=0.6)

plt.title('Fuel Type vs Car Model')
plt.xlabel('Fuel Type (Encoded)')
plt.ylabel('Model (Encoded)')
plt.legend(title='Fuel Type')
plt.grid(True)
plt.tight_layout()
plt.show()

```



```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from datetime import datetime

# Load the dataset
df = pd.read_csv('used_cars_data.csv') # 🗂 Replace with your actual filename

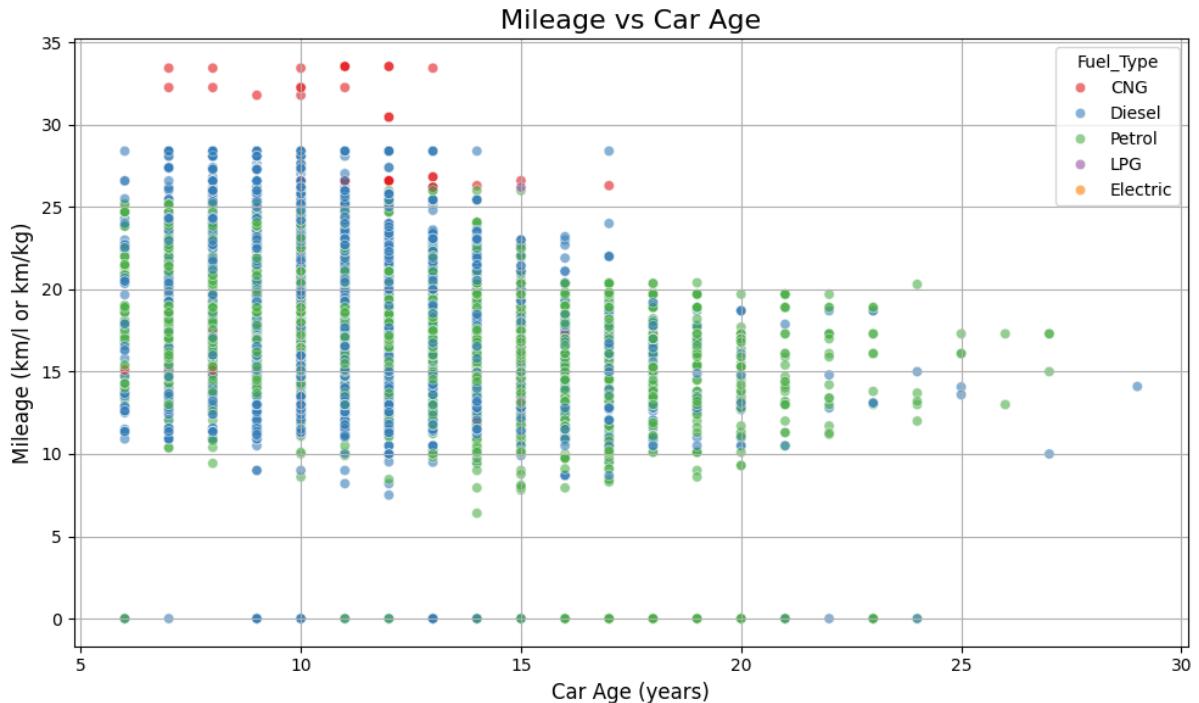
# Clean Mileage column: strip ' kmpl' or ' km/kg' and convert to float
df['Mileage'] = df['Mileage'].str.extract(r'(\d+\.\d+)').astype(float)

# Calculate Car Age based on Year
current_year = datetime.now().year
df['Car_Age'] = current_year - df['Year']

# Optional: check if columns exist and look fine
print(df[['Year', 'Mileage', 'Car_Age', 'Fuel_Type']].head())

# Plotting: Mileage vs Car Age, colored by Fuel Type
plt.figure(figsize=(10, 6))
sns.scatterplot(data=df, x='Car_Age', y='Mileage', hue='Fuel_Type', palette='Set1', alpha=0.6)

plt.title('Mileage vs Car Age', fontsize=16)
plt.xlabel('Car Age (years)', fontsize=12)
plt.ylabel('Mileage (km/l or km/kg)', fontsize=12)
plt.grid(True)
plt.tight_layout()
plt.show()
```



```

import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from datetime import datetime

# Load dataset
df = pd.read_csv('used_cars_data.csv')

# Clean column names
df.columns = df.columns.str.strip()

# Use correct model column name (adjust if your dataset has a different one)
model_column = 'Name' # Change this if your actual column is different

# Clean Mileage column (if exists)
if 'Mileage' in df.columns:
    df['Mileage'] = df['Mileage'].astype(str).str.extract(r'(\d+\.\d+)').astype(float)

```

```
# Calculate Car Age
if 'Year' in df.columns:
    df['Car_Age'] = datetime.now().year - df['Year']
else:
    raise ValueError("Column 'Year' not found in dataset.")

# Filter Top 10 Car Models
top_models = df[model_column].value_counts().nlargest(10).index
df_top = df[df[model_column].isin(top_models)]

# Plot Box Plot: Model vs Car Age
plt.figure(figsize=(12, 6))
sns.boxplot(data=df_top, x=model_column, y='Car_Age', palette='Set2')
plt.xticks(rotation=45)
plt.title('Car Age Distribution by Top 10 Models')
plt.xlabel('Car Model')
plt.ylabel('Car Age (years)')
plt.tight_layout()
plt.show()
```

