```
import pandas as pd
file_path = "/content/CAR DETAILS FROM CAR DEKHO.csv"
data = pd.read_csv(file_path)
print(data.head())
```

```
                      name  year  selling_price  km_driven    fuel  \
0             Maruti 800 AC  2007          60000      70000  Petrol
1  Maruti Wagon R LXI Minor  2007         135000      50000  Petrol
2      Hyundai Verna 1.6 SX  2012         600000     100000  Diesel
3    Datsun RediGO T Option  2017         250000      46000  Petrol
4     Honda Amaze VX i-DTEC  2014         450000     141000  Diesel

  seller_type transmission         owner
0  Individual       Manual   First Owner
1  Individual       Manual   First Owner
2  Individual       Manual   First Owner
3  Individual       Manual   First Owner
4  Individual       Manual  Second Owner
```

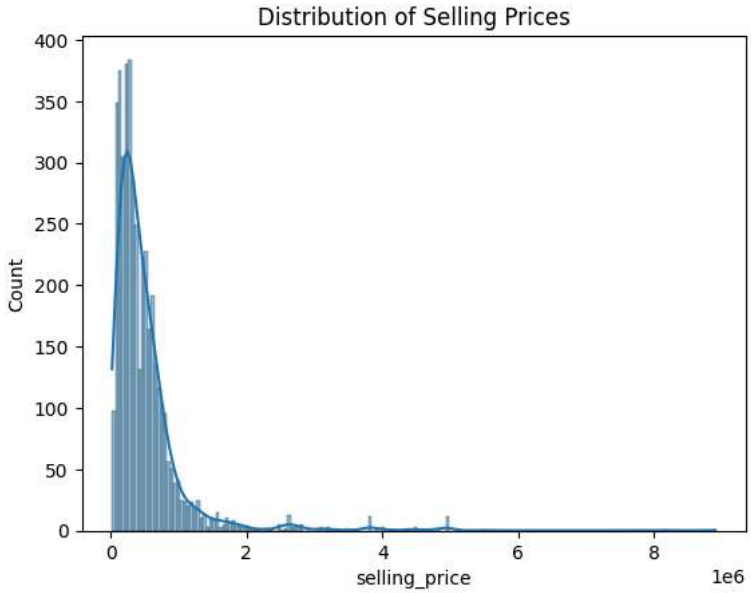
```
import pandas as pd
import numpy as np
file_path = "/content/CAR DETAILS FROM CAR DEKHO.csv"
data = pd.read_csv(file_path)
print(data.info())
data = data.dropna()
data['year'] = data['year'].astype(int)
data['km_driven'] = data['km_driven'].astype(str).str.replace(",", "").astype(int)
data['fuel'] = data['fuel'].astype("category").cat.codes
data['seller_type'] = data['seller_type'].astype("category").cat.codes
data['transmission'] = data['transmission'].astype("category").cat.codes
data['owner'] = data['owner'].astype("category").cat.codes
data = data.drop(columns=['name'])
print(data.head())
```

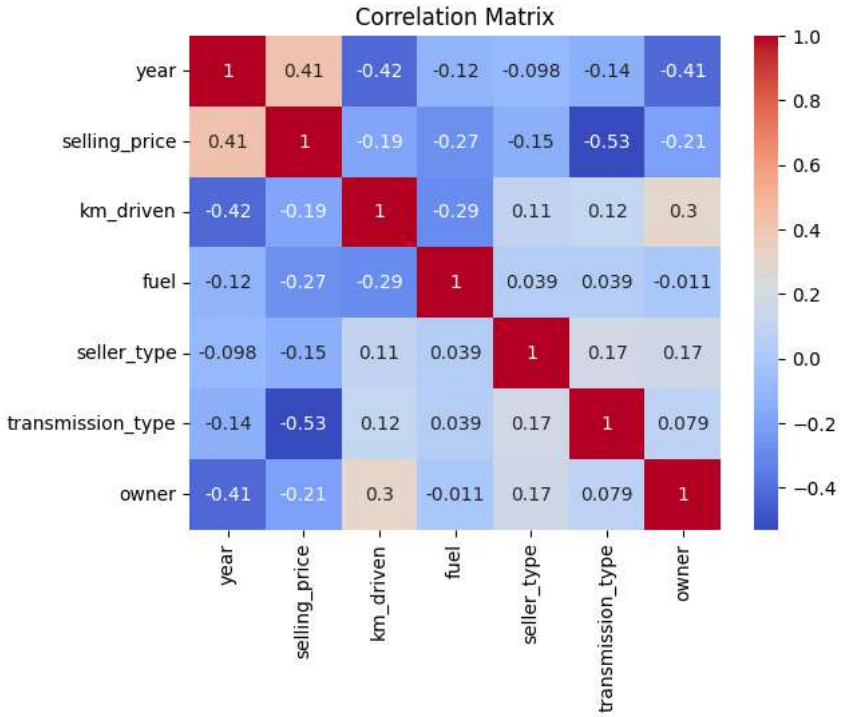
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4340 entries, 0 to 4339
Data columns (total 8 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   name           4340 non-null   object
 1   year           4340 non-null   int64
 2   selling_price  4340 non-null   int64
 3   km_driven      4340 non-null   int64
 4   fuel           4340 non-null   object
 5   seller_type    4340 non-null   object
 6   transmission   4340 non-null   object
 7   owner          4340 non-null   object
dtypes: int64(3), object(5)
memory usage: 271.4+ KB
None
   year  selling_price  km_driven  fuel  seller_type  transmission  owner
0  2007          60000      70000     4            1             1      0
1  2007         135000      50000     4            1             1      0
2  2012         600000     100000     1            1             1      0
3  2017         250000      46000     4            1             1      0
4  2014         450000     141000     1            1             1      2
```

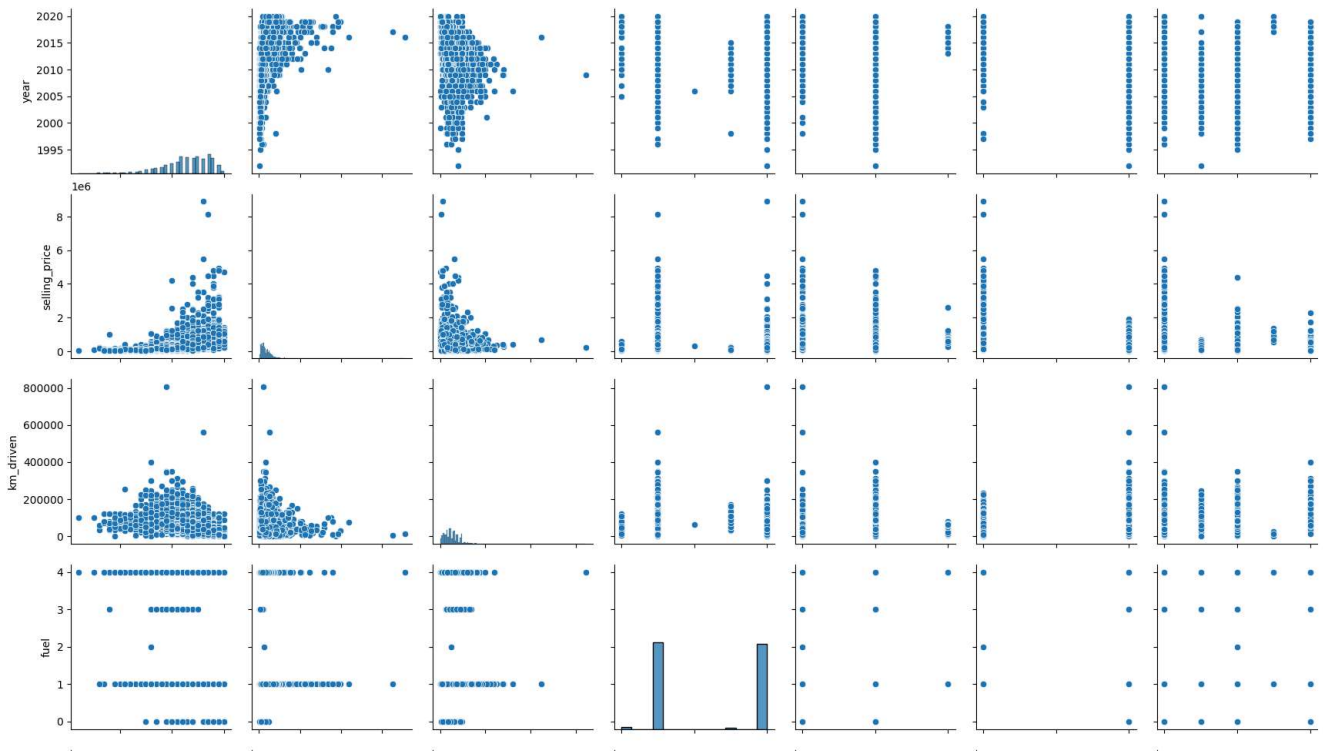
```
import pandas as pd
import numpy as np
file_path = "/content/CAR DETAILS FROM CAR DEKHO.csv"
data = pd.read_csv(file_path)
print("Dataset Info:")
print(data.info())
print("\nFirst 5 Rows of the Dataset:")
print(data.head())
data['year'] = data['year'].astype(int)
data['km_driven'] = data['km_driven'].astype(str).str.replace(',', '').astype
(int)
data = data.rename(columns={"transmission": "transmission_type"})
data['fuel'] = data['fuel'].astype('category').cat.codes
data['seller_type'] = data['seller_type'].astype('category').cat.codes
data['transmission_type'] = data['transmission_type'].astype('category').cat.
codes
data['owner'] = data['owner'].astype('category').cat.codes
data = data.drop(columns=['name'])
print("\nCleaned Dataset Info:")
print(data.info())
print("\nDescriptive Statistics of Cleaned Data:")
print(data.describe())
```

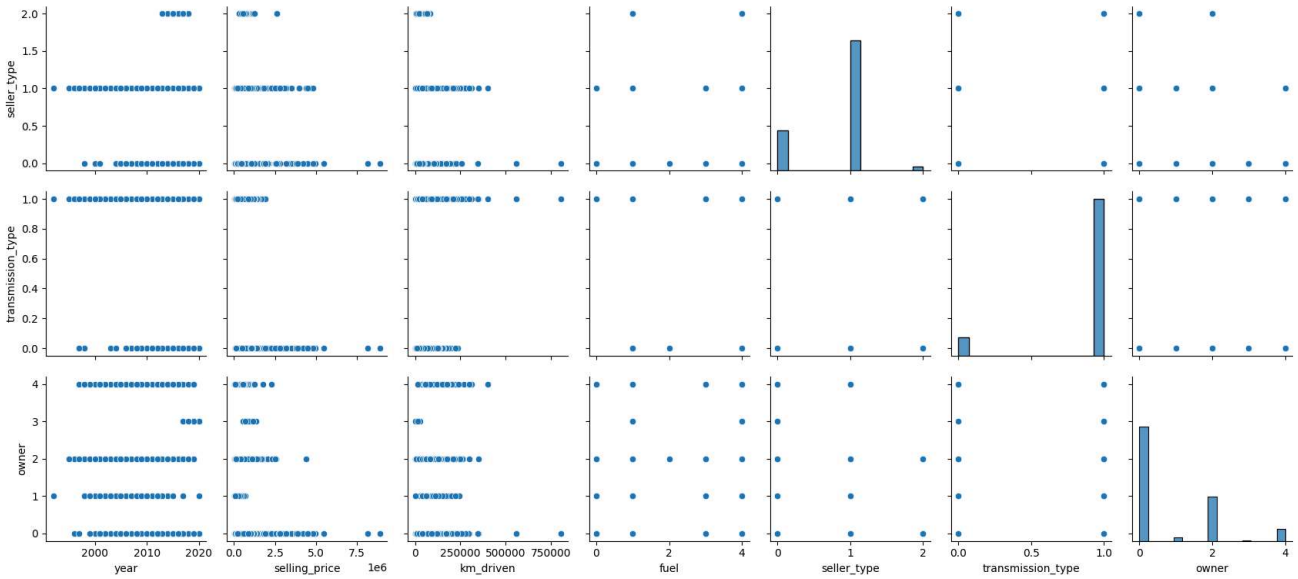
```
Dataset Info:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4340 entries, 0 to 4339
Data columns (total 8 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   name           4340 non-null   object
 1   year           4340 non-null   int64
 2   selling_price  4340 non-null   int64
 3   km_driven      4340 non-null   int64
 4   fuel           4340 non-null   object
 5   seller_type    4340 non-null   object
 6   transmission   4340 non-null   object
 7   owner          4340 non-null   object
dtypes: int64(3), object(5)
memory usage: 271.4+ KB
None

First 5 Rows of the Dataset:
                         name  year  selling_price  km_driven    fuel  \
0             Maruti 800 AC  2007          60000      70000  Petrol
1  Maruti Wagon R LXI Minor  2007         135000      50000  Petrol
2      Hyundai Verna 1.6 SX  2012         600000     100000  Diesel
3    Datsun RediGO T Option  2017         250000      46000  Petrol
4     Honda Amaze VX i-DTEC  2014         450000     141000  Diesel

  seller_type transmission         owner
0  Individual       Manual   First Owner
1  Individual       Manual   First Owner
2  Individual       Manual   First Owner
3  Individual       Manual   First Owner
4  Individual       Manual  Second Owner

Cleaned Dataset Info:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4340 entries, 0 to 4339
Data columns (total 7 columns):
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   year               4340 non-null   int64
 1   selling_price      4340 non-null   int64
 2   km_driven          4340 non-null   int64
 3   fuel               4340 non-null   int8
 4   seller_type        4340 non-null   int8
 5   transmission_type  4340 non-null   int8
 6   owner              4340 non-null   int8
dtypes: int64(3), int8(4)
memory usage: 118.8 KB
None

Descriptive Statistics of Cleaned Data:
              year  selling_price      km_driven         fuel  seller_type  \
count  4340.000000   4.340000e+03    4340.000000  4340.000000  4340.000000
mean   2013.090783   5.041273e+05   66215.777419     2.469124     0.794470
std       4.215344   5.785487e+05   46644.102194     1.508435     0.458629
min    1992.000000   2.000000e+04       1.000000     0.000000     0.000000
25%    2011.000000   2.087498e+05   35000.000000     1.000000     1.000000
50%    2014.000000   3.500000e+05   60000.000000     1.000000     1.000000
```

```
import seaborn as sns
import matplotlib.pyplot as plt
sns.histplot(data['selling_price'], kde=True)
plt.title("Distribution of Selling Prices")
plt.show()
correlation_matrix = data.corr()
sns.heatmap(correlation_matrix, annot=True, cmap="coolwarm")
plt.title("Correlation Matrix")
plt.show()
sns.pairplot(data)
plt.show()
```


Distribution of Selling Prices
Count
selling_price
1e6


Correlation Matrix
year
selling_price
km_driven
fuel
seller_type
transmission_type
owner




seller_type
transmission_type
owner
year
selling_price
1e6
km_driven
fuel
seller_type
transmission_type
owner

```
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.ensemble import RandomForestRegressor
from sklearn.tree import DecisionTreeRegressor
from sklearn.metrics import r2_score, mean_squared_error, mean_absolute_error
X = data.drop(columns=['selling_price'])
y = data['selling_price']
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
models = {
    "Linear Regression": LinearRegression(),
    "Decision Tree": DecisionTreeRegressor(),
    "Random Forest": RandomForestRegressor()
}
for name, model in models.items():
    model.fit(X_train, y_train)
    y_pred = model.predict(X_test)
    print(f"{name}:")
    print(f"  R2 Score: {r2_score(y_test, y_pred)}")
    print(f"  MSE: {mean_squared_error(y_test, y_pred)}")
    print(f"  MAE: {mean_absolute_error(y_test, y_pred)}")
    print("-" * 30)
```

```
Linear Regression:
  R2 Score: 0.3939008529340017
  MSE: 184963450105.73505
  MAE: 222188.94261516692
------------------------------
Decision Tree:
  R2 Score: 0.3614475228025418
  MSE: 194867242146.34915
  MAE: 181041.7785619942
------------------------------
Random Forest:
  R2 Score: 0.4984431680396646
  MSE: 153060241896.9983
  MAE: 169105.84811620106
------------------------------
```

```
from sklearn.model_selection import GridSearchCV
param_grid = {
    'n_estimators': [50, 100, 200],
    'max_depth': [10, 20, None]
    }
grid_search = GridSearchCV(RandomForestRegressor(), param_grid, cv=3, scoring='r2')
grid_search.fit(X_train, y_train)
print("Best Parameters:", grid_search.best_params_)
print("Best R2 Score:", grid_search.best_score_)
```

```
Best Parameters: {'max_depth': 10, 'n_estimators': 50}
Best R2 Score: 0.6301356713508779
```

```
pip install streamlit
```

```
Requirement already satisfied: streamlit in /usr/local/lib/python3.10/dist-packages (1.40.2)
Requirement already satisfied: altair<6,>=4.0 in /usr/local/lib/python3.10/dist-packages (from streamlit) (4.2.2)
Requirement already satisfied: blinker<2,>=1.0.0 in /usr/local/lib/python3.10/dist-packages (from streamlit) (1.9.0)
Requirement already satisfied: cachetools<6,>=4.0 in /usr/local/lib/python3.10/dist-packages (from streamlit) (5.5.0)
Requirement already satisfied: click<9,>=7.0 in /usr/local/lib/python3.10/dist-packages (from streamlit) (8.1.7)
Requirement already satisfied: numpy<3,>=1.23 in /usr/local/lib/python3.10/dist-packages (from streamlit) (1.26.4)
Requirement already satisfied: packaging<25,>=20 in /usr/local/lib/python3.10/dist-packages (from streamlit) (24.2)
Requirement already satisfied: pandas<3,>=1.4.0 in /usr/local/lib/python3.10/dist-packages (from streamlit) (2.2.2)
Requirement already satisfied: pillow<12,>=7.1.0 in /usr/local/lib/python3.10/dist-packages (from streamlit) (11.0.0)
Requirement already satisfied: protobuf<6,>=3.20 in /usr/local/lib/python3.10/dist-packages (from streamlit) (4.25.5)
Requirement already satisfied: pyarrow>=7.0 in /usr/local/lib/python3.10/dist-packages (from streamlit) (17.0.0)
Requirement already satisfied: requests<3,>=2.27 in /usr/local/lib/python3.10/dist-packages (from streamlit) (2.32.3)
Requirement already satisfied: rich<14,>=10.14.0 in /usr/local/lib/python3.10/dist-packages (from streamlit) (13.9.4)
Requirement already satisfied: tenacity<10,>=8.1.0 in /usr/local/lib/python3.10/dist-packages (from streamlit) (9.0.0)
Requirement already satisfied: toml<2,>=0.10.1 in /usr/local/lib/python3.10/dist-packages (from streamlit) (0.10.2)
Requirement already satisfied: typing-extensions<5,>=4.3.0 in /usr/local/lib/python3.10/dist-packages (from streamlit) (4.12.2)
Requirement already satisfied: watchdog<7,>=2.1.5 in /usr/local/lib/python3.10/dist-packages (from streamlit) (6.0.0)
Requirement already satisfied: gitpython!=3.1.19,<4,>=3.0.7 in /usr/local/lib/python3.10/dist-packages (from streamlit) (3.1.43)
Requirement already satisfied: pydeck<1,>=0.8.0b4 in /usr/local/lib/python3.10/dist-packages (from streamlit) (0.9.1)
Requirement already satisfied: tornado<7,>=6.0.3 in /usr/local/lib/python3.10/dist-packages (from streamlit) (6.3.3)
Requirement already satisfied: entrypoints in /usr/local/lib/python3.10/dist-packages (from altair<6,>=4.0->streamlit) (0.4)
Requirement already satisfied: jinja2 in /usr/local/lib/python3.10/dist-packages (from altair<6,>=4.0->streamlit) (3.1.4)
Requirement already satisfied: jsonschema>=3.0 in /usr/local/lib/python3.10/dist-packages (from altair<6,>=4.0->streamlit) (4.23.0)
Requirement already satisfied: toolz in /usr/local/lib/python3.10/dist-packages (from altair<6,>=4.0->streamlit) (0.12.1)
Requirement already satisfied: gitdb<5,>=4.0.1 in /usr/local/lib/python3.10/dist-packages (from gitpython!=3.1.19,<4,>=3.0.7->strea
Requirement already satisfied: python-dateutil>=2.8.2 in /usr/local/lib/python3.10/dist-packages (from pandas<3,>=1.4.0->streamlit)
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.10/dist-packages (from pandas<3,>=1.4.0->streamlit) (2024.2)
Requirement already satisfied: tzdata>=2022.7 in /usr/local/lib/python3.10/dist-packages (from pandas<3,>=1.4.0->streamlit) (2024.2
```

```
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.10/dist-packages (from requests<3,>=2.27->streaml
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-packages (from requests<3,>=2.27->streamlit) (3.10)
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.10/dist-packages (from requests<3,>=2.27->streamlit) (2
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.10/dist-packages (from requests<3,>=2.27->streamlit) (2
Requirement already satisfied: markdown-it-py>=2.2.0 in /usr/local/lib/python3.10/dist-packages (from rich<14,>=10.14.0->streamlit)
Requirement already satisfied: pygments<3.0.0,>=2.13.0 in /usr/local/lib/python3.10/dist-packages (from rich<14,>=10.14.0->streamlit
Requirement already satisfied: smmap<6,>=3.0.1 in /usr/local/lib/python3.10/dist-packages (from gitdb<5,>=4.0.1->gitpython!=3.1.19,
Requirement already satisfied: MarkupSafe>=2.0 in /usr/local/lib/python3.10/dist-packages (from jinja2->altair<6,>=4.0->streamlit)
Requirement already satisfied: attrs>=22.2.0 in /usr/local/lib/python3.10/dist-packages (from jsonschema>=3.0->altair<6,>=4.0->strea
Requirement already satisfied: jsonschema-specifications>=2023.03.6 in /usr/local/lib/python3.10/dist-packages (from jsonschema>=3.0
Requirement already satisfied: referencing>=0.28.4 in /usr/local/lib/python3.10/dist-packages (from jsonschema>=3.0->altair<6,>=4.0
Requirement already satisfied: rpds-py>=0.7.1 in /usr/local/lib/python3.10/dist-packages (from jsonschema>=3.0->altair<6,>=4.0->stre
Requirement already satisfied: mdurl~=0.1 in /usr/local/lib/python3.10/dist-packages (from markdown-it-py>=2.2.0->rich<14,>=10.14.0
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.10/dist-packages (from python-dateutil>=2.8.2->pandas<3,>=1.4.0->s
```

```
import joblib
joblib.dump(model, "car_price_model.pkl")
```

```
['car_price_model.pkl']
```

```
import streamlit as st
import numpy as np
import joblib
model = joblib.load("car_price_model.pkl")
st.title("Car Price Prediction App")
st.write("Predict the price of a used car based on various features.")
year = st.number_input("Year of Purchase", min_value=2000, max_value=2023, step=1)
km_driven = st.number_input("Kilometers Driven", min_value=0)
fuel = st.selectbox("Fuel Type", ["Petrol", "Diesel", "CNG", "LPG", "Electric"])
seller_type = st.selectbox("Seller Type", ["Individual", "Dealer", "Trustmark Dealer"])
transmission = st.selectbox("Transmission Type", ["Manual", "Automatic"])
owner = st.selectbox("Owner Type", ["First Owner", "Second Owner", "Third Owner", "Fourth & Above", "Test Drive Car"])
fuel_mapping = {"Petrol": 0, "Diesel": 1, "CNG": 2, "LPG": 3, "Electric": 4}
seller_type_mapping = {"Individual": 0, "Dealer": 1, "Trustmark Dealer": 2}
transmission_mapping = {"Manual": 0, "Automatic": 1}
owner_mapping = {
    "First Owner": 0, "Second Owner": 1, "Third Owner": 2,
    "Fourth & Above": 3, "Test Drive Car": 4
}
features = np.array([
    year, km_driven, fuel_mapping[fuel], seller_type_mapping[seller_type],
    transmission_mapping[transmission], owner_mapping[owner]
]).reshape(1, -1)
if st.button("Predict Price"):
    try:
        price = model.predict(features)
        st.success(f"Predicted Price: ₹ {price[0]:,.2f}")
    except Exception as e:
        st.error(f"An error occurred: {e}")
```

```
2024-12-01 13:45:41.881 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare mode.
2024-12-01 13:45:41.882 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare mode.
2024-12-01 13:45:41.885 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare mode.
2024-12-01 13:45:41.886 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare mode.
2024-12-01 13:45:41.888 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare mode.
2024-12-01 13:45:41.890 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare mode.
2024-12-01 13:45:41.891 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare mode.
2024-12-01 13:45:41.893 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare mode.
2024-12-01 13:45:41.893 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare mode.
2024-12-01 13:45:41.894 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare mode.
2024-12-01 13:45:41.895 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare mode.
2024-12-01 13:45:41.896 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare mode.
2024-12-01 13:45:41.897 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare mode.
2024-12-01 13:45:41.902 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare mode.
2024-12-01 13:45:41.905 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare mode.
2024-12-01 13:45:41.905 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare mode.
2024-12-01 13:45:41.906 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare mode.
2024-12-01 13:45:41.907 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare mode.
2024-12-01 13:45:41.908 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare mode.
2024-12-01 13:45:41.909 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare mode.
2024-12-01 13:45:41.911 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare mode.
2024-12-01 13:45:41.912 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare mode.
2024-12-01 13:45:41.913 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare mode.
```