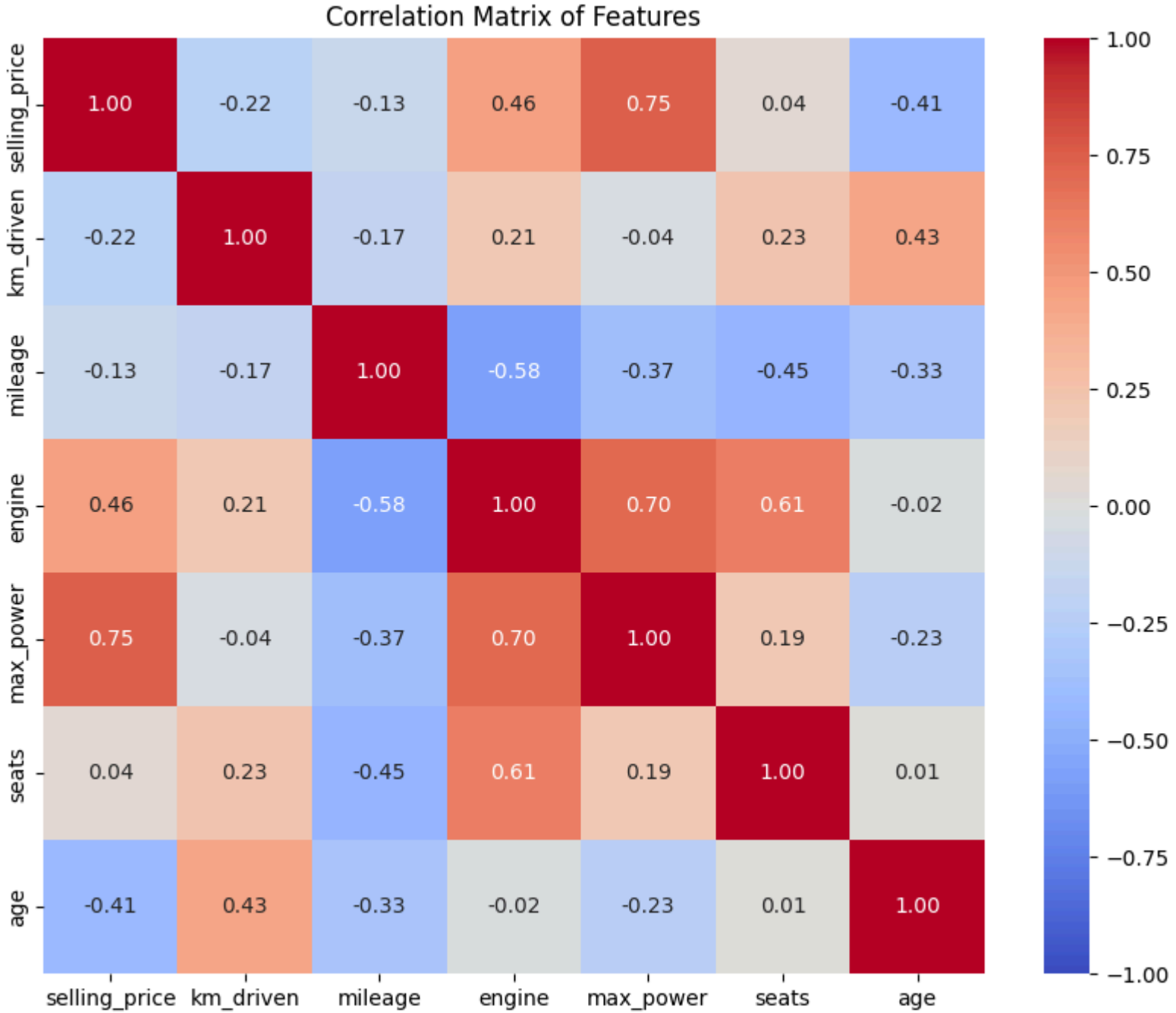
```
print("Name :", "$OUMYA")
print("Roll no. :", 2230324)
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score


file_path = 'Car details v3.csv'
car_data = pd.read_csv(file_path)


car_data['mileage'] = car_data['mileage'].str.extract('(\d+\.?\d*)').astype(float)
car_data['engine'] = car_data['engine'].str.extract('(\d+)').astype(float)
car_data['max_power'] = car_data['max_power'].str.extract('(\d+\.?\d*)').astype(float)


car_data = car_data.dropna(subset=['mileage', 'engine', 'max_power'])


current_year = 2025
car_data['age'] = current_year - car_data['year']
car_data = car_data.drop(columns=['name', 'year', 'torque'])


numeric_features = car_data.select_dtypes(include=np.number).columns.tolist()
correlation_matrix = car_data[numeric_features].corr()

plt.figure(figsize=(10, 8))
sns.heatmap(correlation_matrix, annot=True, fmt='.2f', cmap='coolwarm', vmin=-1, vmax=1)
plt.title('Correlation Matrix of Features')
plt.show()
```

```
top_corr_features = correlation_matrix['selling_price'].sort_values(ascending=False)[1:4]
print("Top 3 Features Correlated with Selling Price:")
print(top_corr_features)


plt.figure(figsize=(18, 5))
for i, feature in enumerate(top_corr_features.index, 1):
    plt.subplot(1, 3, i)
    sns.scatterplot(data=car_data, x=feature, y='selling_price', alpha=0.6)
    plt.title(f'Scatter Plot: Selling Price vs {feature}')
    plt.xlabel(feature)
    plt.ylabel('Selling Price')
plt.tight_layout()
plt.show()


car_data.hist(bins=20, figsize=(15, 10), edgecolor='black')
plt.suptitle('Histogram Plots of Features', fontsize=16)
plt.show()


X = car_data[['age', 'km_driven', 'mileage', 'engine', 'max_power', 'seats']]
y = car_data['selling_price']


X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)


model = LinearRegression()
model.fit(X_train, y_train)


y_pred = model.predict(X_test)


mse = mean_squared_error(y_test, y_pred)
rmse = np.sqrt(mse)
```

```
    r2 = r2_score(y_test, y_pred)

    print(f"Evaluation Metrics:\nMean Squared Error (MSE): {mse:.2f}")
    print(f"Root Mean Squared Error (RMSE): {rmse:.2f}")
    print(f"R^2 Score: {r2:.2f}")
```

Name : $OUMYA
Roll no. : 2230324



```
Top 3 Features Correlated with Selling Price:
max_power    0.749674
engine       0.455682
seats        0.041617
Name: selling_price, dtype: float64
```