In [1]:
```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score
import matplotlib.pyplot as plt
```

In [2]:
```
df = pd.read_csv(r"C:\Users\warul\Downloads\archive (2).zip")
```

In [3]:
```
print(df.info())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 301 entries, 0 to 300
Data columns (total 9 columns):
 #   Column         Non-Null Count  Dtype  
---  ------         --------------  -----  
 0   Car_Name       301 non-null    object 
 1   Year           301 non-null    int64  
 2   Selling_Price  301 non-null    float64
 3   Present_Price  301 non-null    float64
 4   Driven_kms     301 non-null    int64  
 5   Fuel_Type      301 non-null    object 
 6   Selling_type   301 non-null    object 
 7   Transmission   301 non-null    object 
 8   Owner          301 non-null    int64  
dtypes: float64(2), int64(3), object(4)
memory usage: 21.3+ KB
None
```

In [4]:
```
features = df[['Year', 'Present_Price', 'Driven_kms', 'Fuel_Type', 'Selling_type', 'Tr
```

In [5]:
```
features = pd.get_dummies(features, columns=['Fuel_Type', 'Selling_type', 'Transmissio
```

In [6]:
```
target = df['Selling_Price']
```

In [7]:
```
X_train, X_test, y_train, y_test = train_test_split(features, target, test_size=0.2, r
```

In [8]:
```
model = LinearRegression()
```

In [9]:
```
model.fit(X_train, y_train)
```

Out[9]:
```
▾ LinearRegression

LinearRegression()
```

In [10]:
```
y_pred = model.predict(X_test)
```

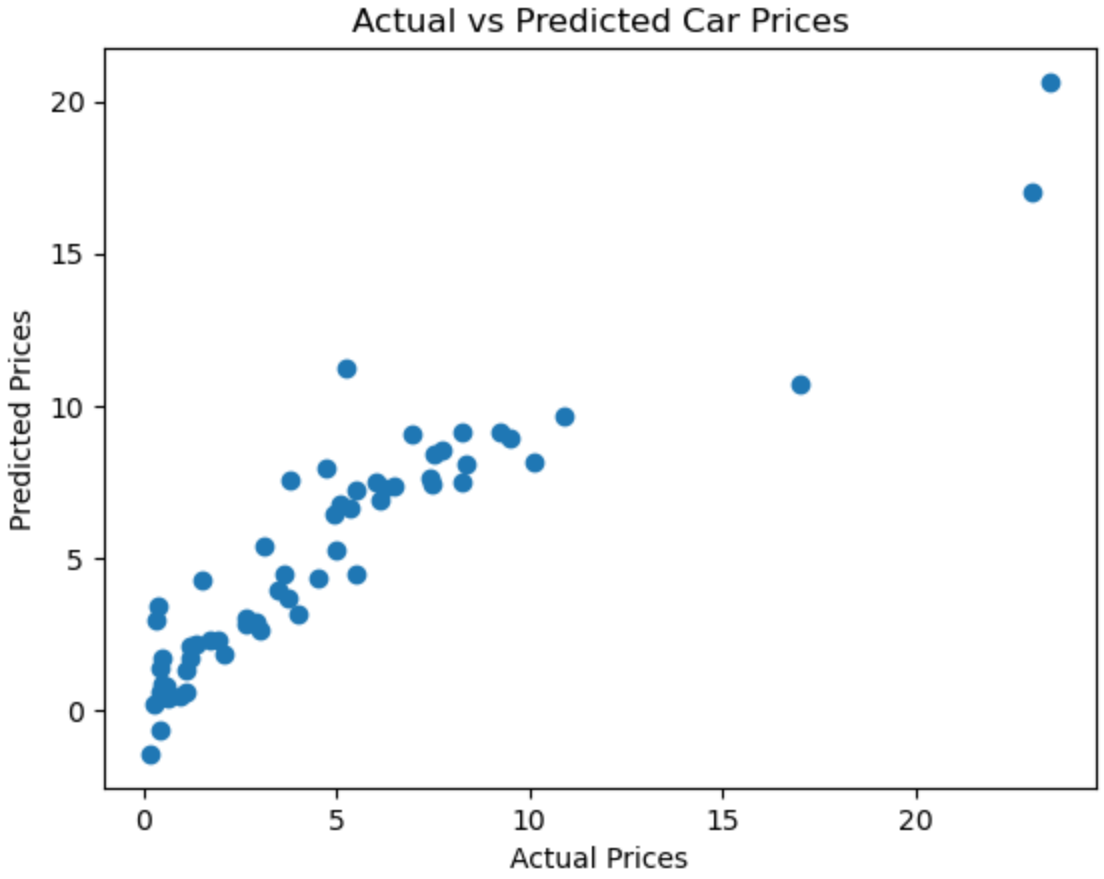
In [11]:
```
mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)
```

In [12]:
```
print(f'Mean Squared Error: {mse}')
print(f'R-squared Score: {r2}')

# Visualize predicted vs actual prices
plt.scatter(y_test, y_pred)
```

```
plt.xlabel('Actual Prices')
plt.ylabel('Predicted Prices')
plt.title('Actual vs Predicted Car Prices')
plt.show()
```

```
Mean Squared Error: 3.481349830520667
R-squared Score: 0.8488707839189312
```



In [ ]: