+ Code + Text
Copy
to   Connect ▾   ⌄
Drive

```python
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
```

```python
df=pd.read_csv('/content/car_age_p
df
```

|	Year	Price
0	2018	465000
1	2019	755000
2	2019	700000
3	2018	465000
4	2018	465000
...	...	...
107	2016	375000
108	2014	300000
109	2015	425000
110	2016	420000
111	2015	425000

112 rows × 2 columns

```python
df.info()
```

```
<class 'pandas.core.frame.DataFram
RangeIndex: 112 entries, 0 to 111
Data columns (total 2 columns):
 #   Column  Non-Null Count   Dtype
---  ------  --------------   -----
 0   Year    112 non-null     int64
 1   Price   112 non-null     int64
dtypes: int64(2)
memory usage: 1.9 KB
```

```python
import seaborn as sns
plt.figure(figsize=(8, 5))
sns.scatterplot(x=df["Year"], y=df
plt.xlabel("Year of Manufacturing"
plt.ylabel("Price (INR)")
plt.title("Year vs. Price of Hyund
plt.grid(True)
plt.show()
```


Year vs. Price of

```python
X = df[["Year"]]
y = df["Price"]
```
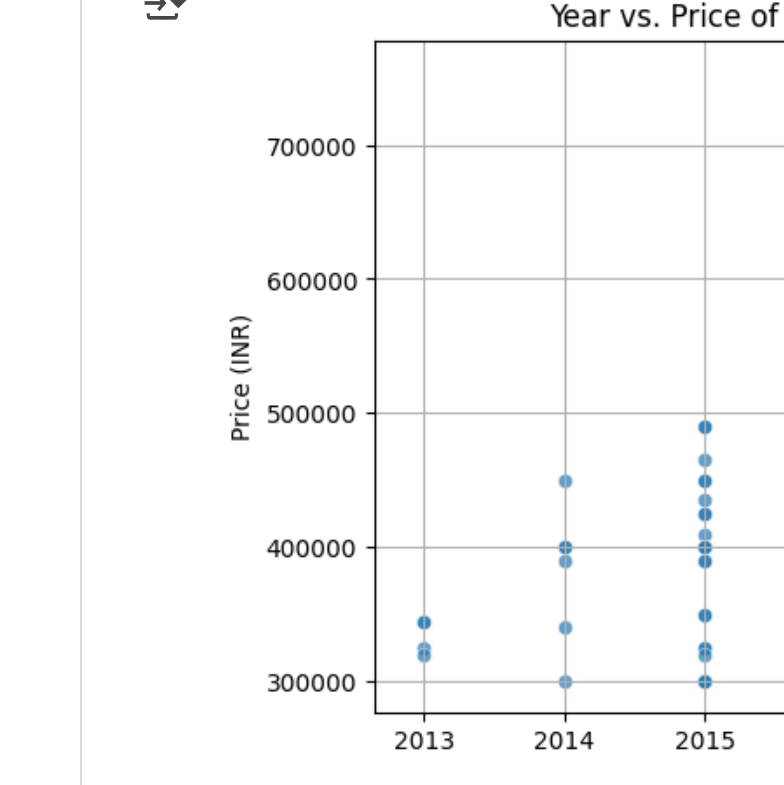
```python
from sklearn.model_selection impor
from sklearn.linear_model import L
from sklearn.metrics import mean_sq
X_train, X_test, y_train, y_test =
```

```python
linear_model = LinearRegression()
lasso_model = Lasso(alpha=1000)
```

```python
linear_model.fit(X_train, y_train)
lasso_model.fit(X_train, y_train)
```

```
▼   Lasso          ● ●
Lasso(alpha=1000)
```

```python
y_pred_linear = linear_model.predi
y_pred_lasso = lasso_model.predict
```

```python
mse_linear = mean_squared_error(y_
r2_linear = r2_score(y_test, y_pre

mse_lasso = mean_squared_error(y_t
r2_lasso = r2_score(y_test, y_pred
```

```python
year_2022 = pd.DataFrame([[2022]],
price_2022_linear = linear_model.p
price_2022_lasso = lasso_model.pre
print("Model Performance:")
print(f"Linear Regression - MSE: {
print(f"Lasso Regression  - MSE: {
```

```
Model Performance:
Linear Regression - MSE: 432690625
Lasso Regression  - MSE: 428101211
```

```python
plt.figure(figsize=(8, 5))
sns.scatterplot(x=df["Year"], y=df
plt.plot(X, linear_model.predict(X
plt.plot(X, lasso_model.predict(X)
plt.xlabel("Year of Manufacturing"
plt.ylabel("Price (INR)")
plt.title("Comparison of Linear an
plt.legend()
plt.grid(True)
plt.show()
```