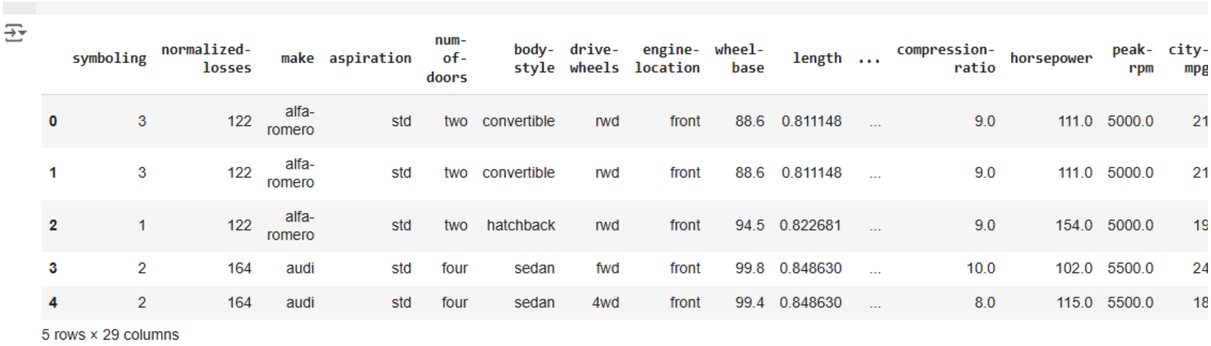
Implementation and Result:

```
import pandas as pd
import numpy as np
df = pd.read_csv('/content/automobile.csv')
df.head()
```



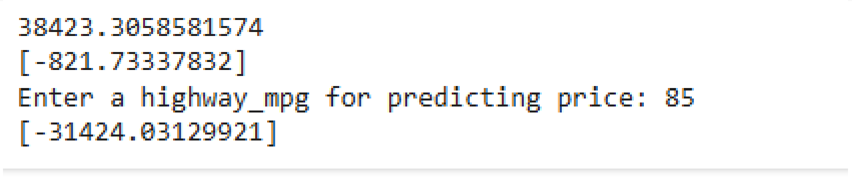
	symboling	normalized-losses	make	aspiration	num-of-doors	body-style	drive-wheels	engine-location	wheel-base	length	...	compression-ratio	horsepower	peak-rpm	city-mpg
0	3	122	alfa-romero	std	two	convertible	rwd	front	88.6	0.811148	...	9.0	111.0	5000.0	21
1	3	122	alfa-romero	std	two	convertible	rwd	front	88.6	0.811148	...	9.0	111.0	5000.0	21
2	1	122	alfa-romero	std	two	hatchback	rwd	front	94.5	0.822681	...	9.0	154.0	5000.0	19
3	2	164	audi	std	four	sedan	fwd	front	99.8	0.848630	...	10.0	102.0	5500.0	24
4	2	164	audi	std	four	sedan	4wd	front	99.4	0.848630	...	8.0	115.0	5500.0	18

5 rows × 29 columns

LinearRegression Linear regression in pandas for data analysis involves using the pandas library for data manipulation and statsmodels or scikit-learn for performing the regression analysis. First, you load and clean the data with pandas, ensuring no missing or erroneous values. This process helps in identifying trends, forecasting outcomes, and guiding data-driven decisions. Linear regression is a fundamental statistical method used to model the relationship between a dependent variable and one or more independent variables by fitting a linear equation to observed data. The goal of linear regression is to find the best-fitting line, typically by minimizing the sum of squared differences between the observed values and the values predicted by the model. This method assumes a linear relationship, where changes in the independent variables result in proportional changes in the dependent variable. In data analysis, linear regression helps in identifying trends, making predictions, and assessing the strength and nature of relationships between variables. It is widely used for forecasting, risk assessment, and determining variable importance in various fields such as economics, finance, and healthcare.

```
from sklearn.linear_model import LinearRegression
df_copy = df.copy()
lm = LinearRegression()
x = df[['highway-mpg']]
```

```
y = df['Price']
lm.fit(x,y)
lm_for_copy = LinearRegression()
x1 = df_copy[['highway-mpg']]
y1 = df_copy['Price']
lm_for_copy.fit(x1,y1)
lm_for_copy.predict(x)
intercept = lm_for_copy.intercept_
slope = lm_for_copy.coef_
print(intercept)
print(slope)
desired_mpg = int(input('Enter a highway_mpg for predicting price: '))
y = slope * desired_mpg + intercept
print(y)
```



```
38423.3058581574
[-821.73337832]
Enter a highway_mpg for predicting price: 85
[-31424.03129921]
```

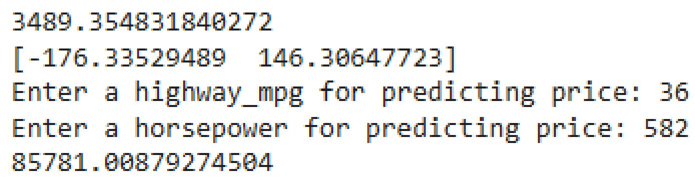
Multiple Regression :- Multiple regression in pandas involves extending linear regression to model the relationship between a dependent variable and two or more independent variables. This technique is useful when the target variable is influenced by multiple factors, and the goal is to understand the combined impact of these predictors. In Python, the pandas library is used for data manipulation, while libraries like statsmodels or scikit-learn are employed to perform the regression analysis. The model fits a linear equation with multiple predictors, estimating the coefficients for each independent variable. After fitting the model, performance metrics such as R-squared, adjusted R-squared, and p-values are analyzed to assess the model's explanatory power and the significance of each predictor. Multiple regression is widely used in data analysis for complex prediction tasks, understanding variable interactions, and identifying key factors influencing the outcome.

```
multi_reg_model = LinearRegression()
x = df[['highway-mpg','horsepower']]
y = df['Price']
multi_reg_model.fit(x,y)
multi_reg_model.predict(x)
intercept = multi_reg_model.intercept_
slope = multi_reg_model.coef_
print(intercept)
print(slope)
desired_mpg = int(input('Enter a highway_mpg for predicting price: '))
```

```
horsepower = int(input('Enter a horsepower for predicting price: '))

y = (slope[0] * desired_mpg +intercept) + (slope[1] * horsepower + intercept)

print(y)
```

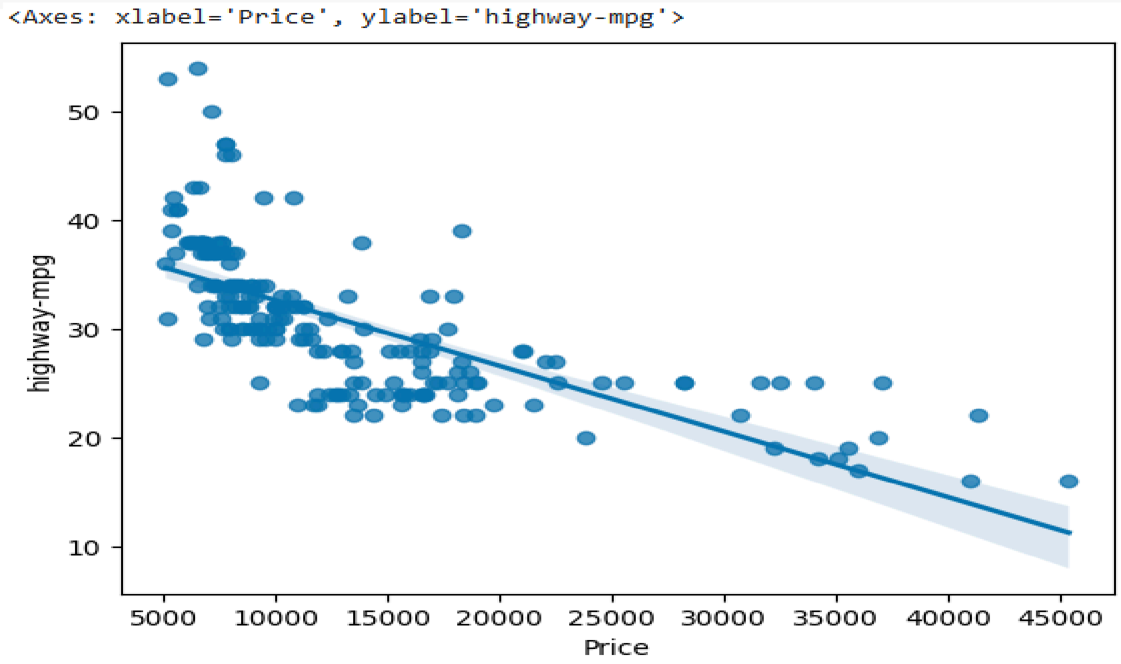


```
3489.354831840272
[-176.33529489  146.30647723]
Enter a highway_mpg for predicting price: 36
Enter a horsepower for predicting price: 582
85781.00879274504
```

Regression Plot :- A regression plot in pandas, often visualized using seaborn or matplotlib, is a powerful tool for visualizing the relationship between two continuous variables, typically for linear regression. This plot shows the data points along with a fitted regression line, which helps to understand how changes in the independent variable influence the dependent variable. Using pandas to prepare the data, you can apply seaborn.regplot() or matplotlib.pyplot.plot() to create this visualization, where the regression line is typically drawn using the least squares method. The plot not only provides a clear visual representation of the data trend but also displays statistical elements like confidence intervals and the line's slope. Regression plots are essential in data analysis for quick insights into the strength and nature of the relationship between variables, making them a fundamental tool for exploratory data analysis and model interpretation.

```
import seaborn as sns

sns.regplot(x="Price",y="highway-mpg",data=df)
```



```
<Axes: xlabel='Price', ylabel='highway-mpg'>
```

Colab Link :- https://colab.research.google.com/drive/1slixpLS8G4s2wIbD1jL66jIiZ9qIHFX5