EXP 1:

A PYTHON PROGRAM TO IMPLEMENT UNIVARIATE, BIVARIATE AND MULTIVARIATE REGRESSION

Aim:
To implement a Python program using univariate, bivariate, and multivariate regression features for a given iris dataset.

PROGRAM:

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np
df = pd.read_csv('../input/iris-dataset/iris.csv') df.head(150)
df.shape (150,5)

#univariate for sepal width
df.loc[df['variety']=='Setosa'] df_Setosa=df.loc[df['variety']=='Setosa']
df_Virginica=df.loc[df['variety']=='Virginica'] df_Versicolor=df.loc[df['variety']=='Versicolor']
plt.scatter(df_Setosa['sepal.width'],np.zeros_like(df_Setosa['sepal.width']))
plt.scatter(df_Virginica['sepal.width'],np.zeros_like(df_Virginica['sepal.width']))
plt.scatter(df_Versicolor['sepal.width'],np.zeros_like(df_Versicolor['sepal.width']))
plt.xlabel('sepal.width')
plt.show()

#univariate for sepal length
df.loc[df['variety']=='Setosa']
df_Setosa=df.loc[df['variety']=='Setosa']
df_Virginica=df.loc[df['variety']=='Virginica']
df_Versicolor=df.loc[df['variety']=='Versicolor']
plt.scatter(df_Setosa['sepal.length'],np.zeros_like(df_Setosa['sepal.length']))
plt.scatter(df_Virginica['sepal.length'],np.zeros_like(df_Virginica['sepal.length']))
plt.scatter(df_Versicolor['sepal.length'],np.zeros_like(df_Versicolor['sepal.length']))
plt.xlabel('sepal.length')
plt.show()

#univariate for petal width
df.loc[df['variety']=='Setosa']
df_Setosa=df.loc[df['variety']=='Setosa']
df_Virginica=df.loc[df['variety']=='Virginica']
df_Versicolor=df.loc[df['variety']=='Versicolor']
```

```
plt.scatter(df_Setosa['petal.width'],np.zeros_like(df_Setosa['petal.width']))
plt.scatter(df_Virginica['petal.width'],np.zeros_like(df_Virginica['petal.width']))
plt.scatter(df_Versicolor['petal.width'],np.zeros_like(df_Versicolor['petal.width']))
plt.xlabel('petal.width')
plt.show()
```

#univariate for petal length

```
df.loc[df['variety']=='Setosa'] df_Setosa=df.loc[df['variety']=='Setosa']
df_Virginica=df.loc[df['variety']=='Virginica'] df_Versicolor=df.loc[df['variety']=='Versicolor']
plt.scatter(df_Setosa['petal.length'],np.zeros_like(df_Setosa['petal.length']))
plt.scatter(df_Virginica['petal.length'],np.zeros_like(df_Virginica['petal.length']))
plt.scatter(df_Versicolor['petal.length'],np.zeros_like(df_Versicolor['petal.length']))
plt.xlabel('petal.length')
plt.show()
```

#bivariate sepal.width vs petal.width

```
sns.FacetGrid(df,hue='variety',height=5).map(plt.scatter,"sepal.width","petal.width").add_legend();
plt.show()
```

#bivariate sepal.length vs petal.length

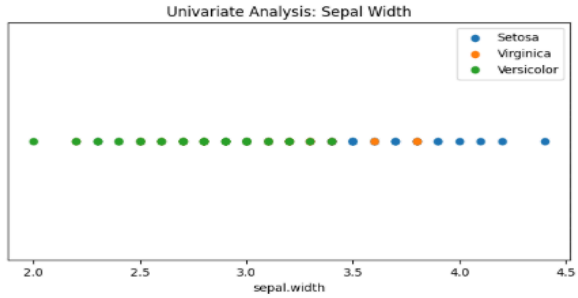
```
sns.FacetGrid(df,hue='variety',height=5).map(plt.scatter,"sepal.length","petal.length").add_legend();
plt.show()
```

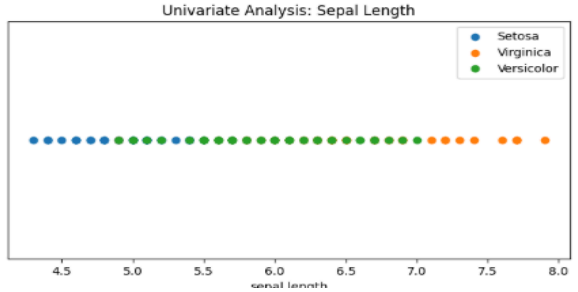
#multivariate all the features

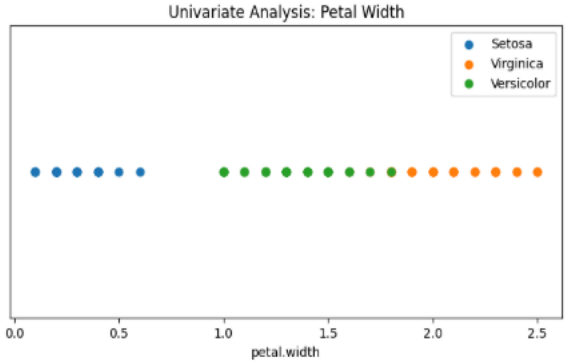
```
sns.pairplot(df,hue="variety",size=2)
```

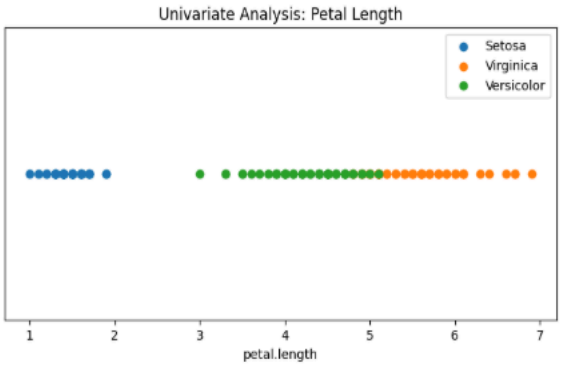
OUTPUT:

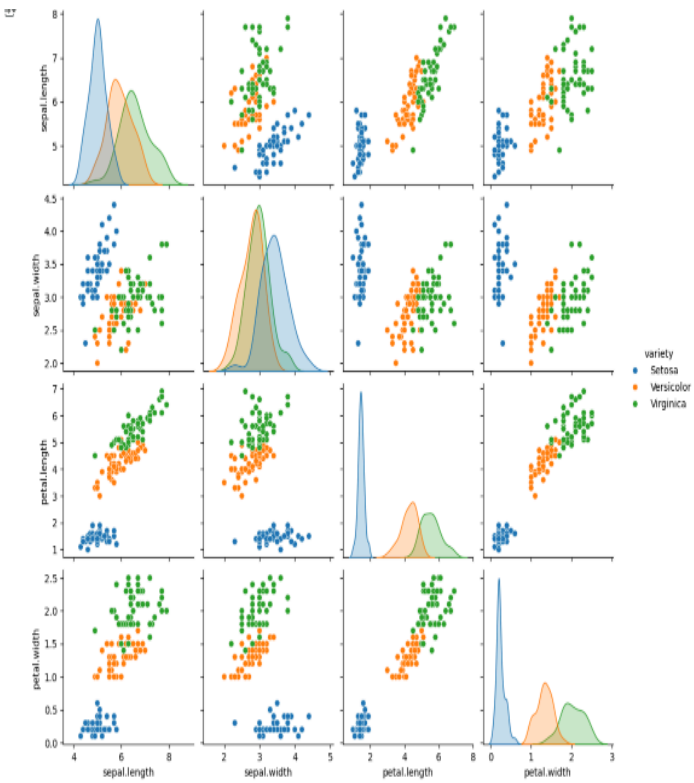
```
   sepal.length  sepal.width  petal.length  petal.width variety
0           5.1          3.5           1.4          0.2  Setosa
1           4.9          3.0           1.4          0.2  Setosa
2           4.7          3.2           1.3          0.2  Setosa
3           4.6          3.1           1.5          0.2  Setosa
4           5.0          3.6           1.4          0.2  Setosa
```











RESULT:

Thus, the Python program to implement univariate, bivariate, and multivariate regression features for the given dataset is analyzed, and the features are plotted using a scatter plot.