In [93]:
```python
# Importing Libraries
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
```

In [95]:
```python
# Importing the Iris Dataset
iris = pd.read_csv(r'C:\Users\jasik\OneDrive\Documents\Datasets\Data\iris.csv')
```

In [97]:
```python
# Looking the dataset, its shape, datatype and descriptive statistics
```

In [99]:
```python
iris.head()
```

Out[99]:

	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

In [101…
```python
iris.shape
```

Out[101… (150, 5)

In [103…
```python
iris.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 150 entries, 0 to 149
Data columns (total 5 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   sepal.length  150 non-null    float64
 1   sepal.width   150 non-null    float64
 2   petal.length  150 non-null    float64
 3   petal.width   150 non-null    float64
 4   variety       150 non-null    object
dtypes: float64(4), object(1)
memory usage: 6.0+ KB
```

In [18]:
```python
iris.describe()
```

Out[18]:

	sepal.length	sepal.width	petal.length	petal.width
count	150.000000	150.000000	150.000000	150.000000
mean	5.843333	3.057333	3.758000	1.199333
std	0.828066	0.435866	1.765298	0.762238
min	4.300000	2.000000	1.000000	0.100000
25%	5.100000	2.800000	1.600000	0.300000
50%	5.800000	3.000000	4.350000	1.300000
75%	6.400000	3.300000	5.100000	1.800000
max	7.900000	4.400000	6.900000	2.500000

In [105…
```python
# Renaming the Columns
```

In [107…
```python
df = iris.rename( columns = {'sepal.length': 'SepalLength',
                       'sepal.width': 'SepalWidth',
                       'petal.length': 'PetalLength',
                       'petal.width': 'PetalWidth'})
```

In [109…
```python
# Data Cleaning - (looking at the null values and the duplicates)
```

In [111…
```python
df.isnull().sum()
```

Out[111…
```
SepalLength    0
SepalWidth     0
PetalLength    0
PetalWidth     0
variety        0
dtype: int64
```

In [113…
```python
df.duplicated().sum()
```

Out[113… 1

In [115…
```python
df.drop_duplicates(subset = 'variety')
```

Out[115…

	SepalLength	SepalWidth	PetalLength	PetalWidth	variety
0	5.1	3.5	1.4	0.2	Setosa
50	7.0	3.2	4.7	1.4	Versicolor
100	6.3	3.3	6.0	2.5	Virginica

In [117…
```python
# Counting the distinct variety
```

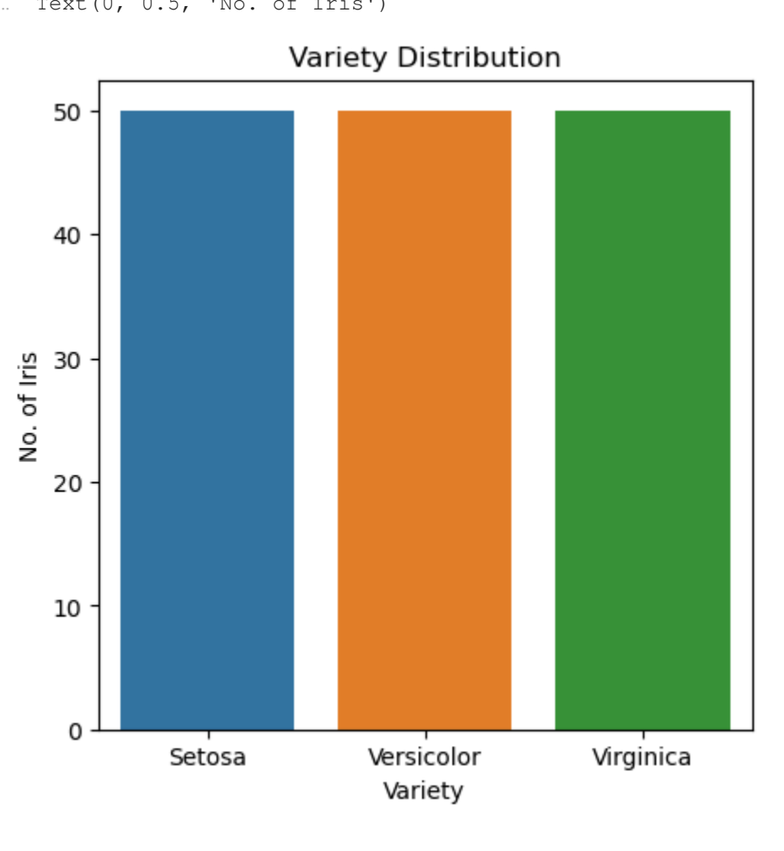
In [119…
```python
df.value_counts('variety')
```

Out[119…
```
variety
Setosa        50
Versicolor    50
Virginica     50
Name: count, dtype: int64
```

In [121…
```python
# Graphical Representation
```

In [123…
```python
plt.figure(figsize = (5,5))
sns.countplot(data = df, x = 'variety', hue = 'variety')
plt.title('Variety Distribution')
plt.xlabel('Variety')
plt.ylabel('No. of Iris')
```

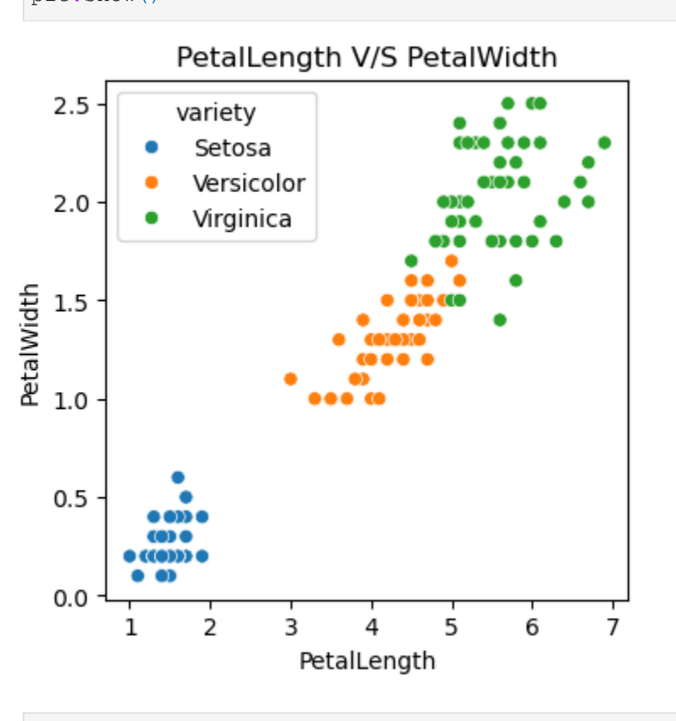
Out[123… Text(0, 0.5, 'No. of Iris')



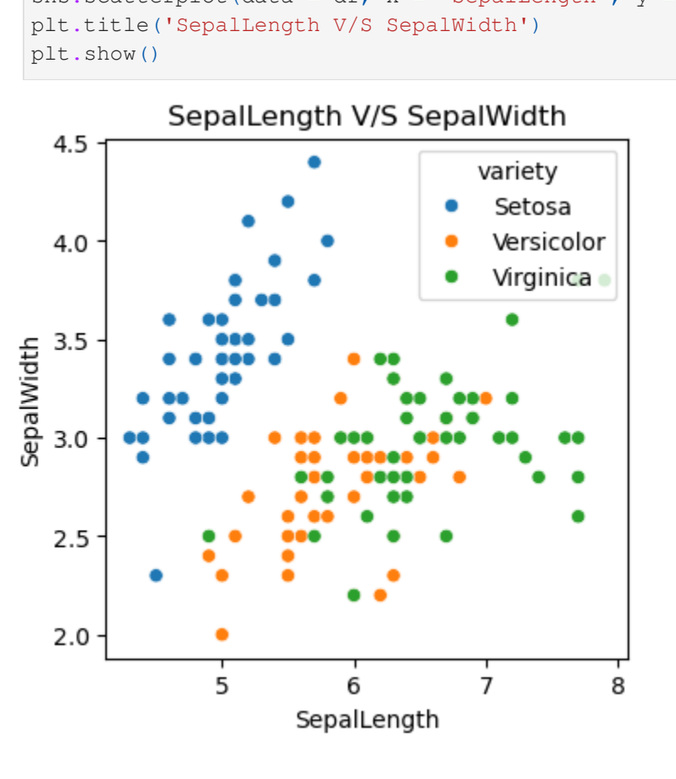
We can see that there is equal numbers of each variety in our dataset.

In [126…
```python
# Relationship Between the PetalLength, PetalWidth, SepalLenghth and the SepalLength
```

In [128…
```python
plt.figure(figsize=(4,4))
sns.scatterplot(data = df, x = 'PetalLength', y = 'PetalWidth', hue = 'variety')
plt.title('PetalLength V/S PetalWidth')
plt.show()
```



In [129…
```python
plt.figure(figsize=(4,4))
sns.scatterplot(data = df, x = 'SepalLength', y = 'SepalWidth', hue = 'variety')
plt.title('SepalLength V/S SepalWidth')
plt.show()
```



Observations

Setosa has smaller petal lengths and widths while Virginica has the largest of petal lengths and widths.

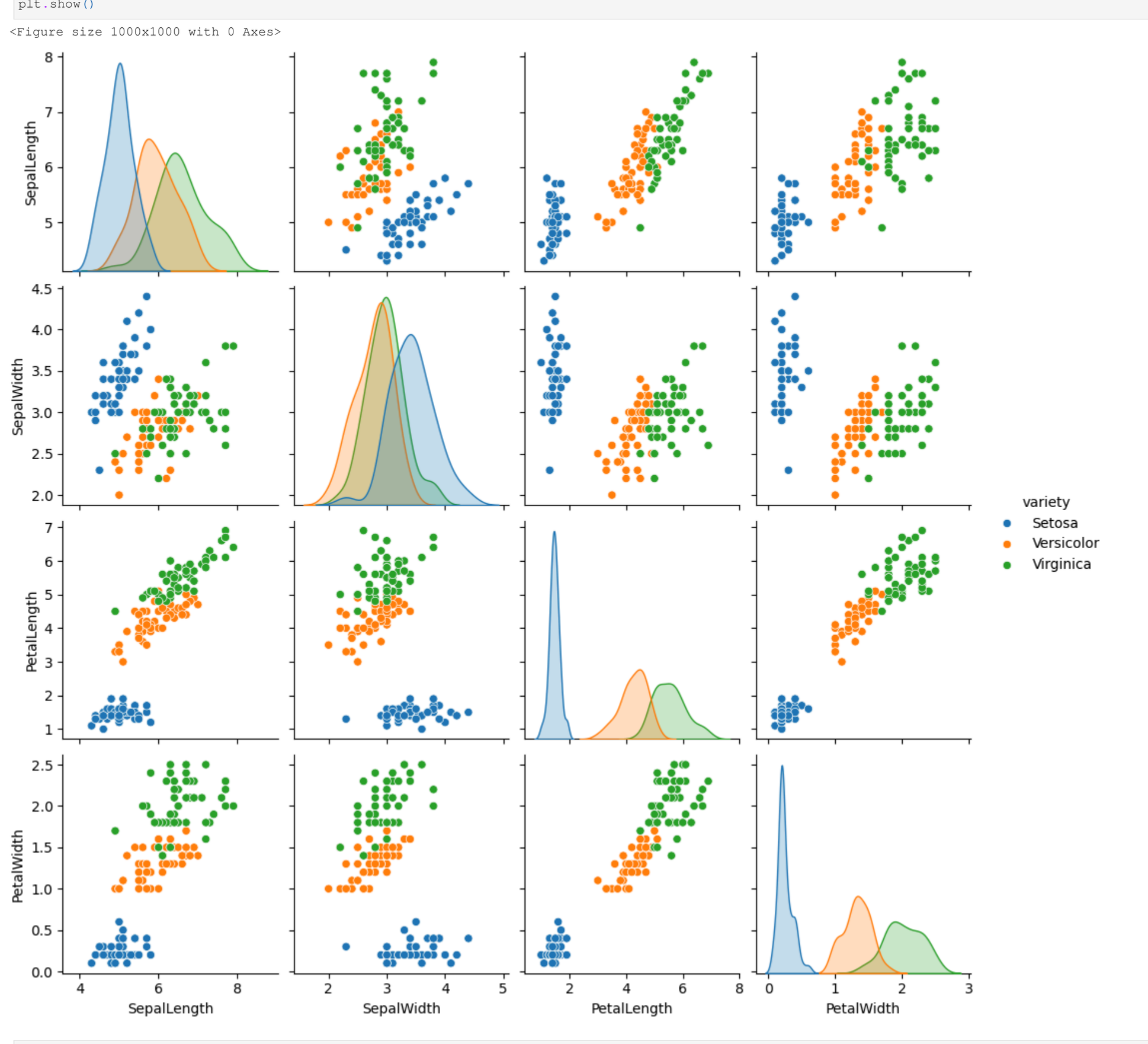
Also Versicolor lies between the two species in terms of petal length and width.

Setosa has smaller sepal lengths and larger sepal widths while Virginica has larger sepal lengths but smaller sepal widths.

Also Versicolor lies between the two species in terms of sepal length and width.s.s.

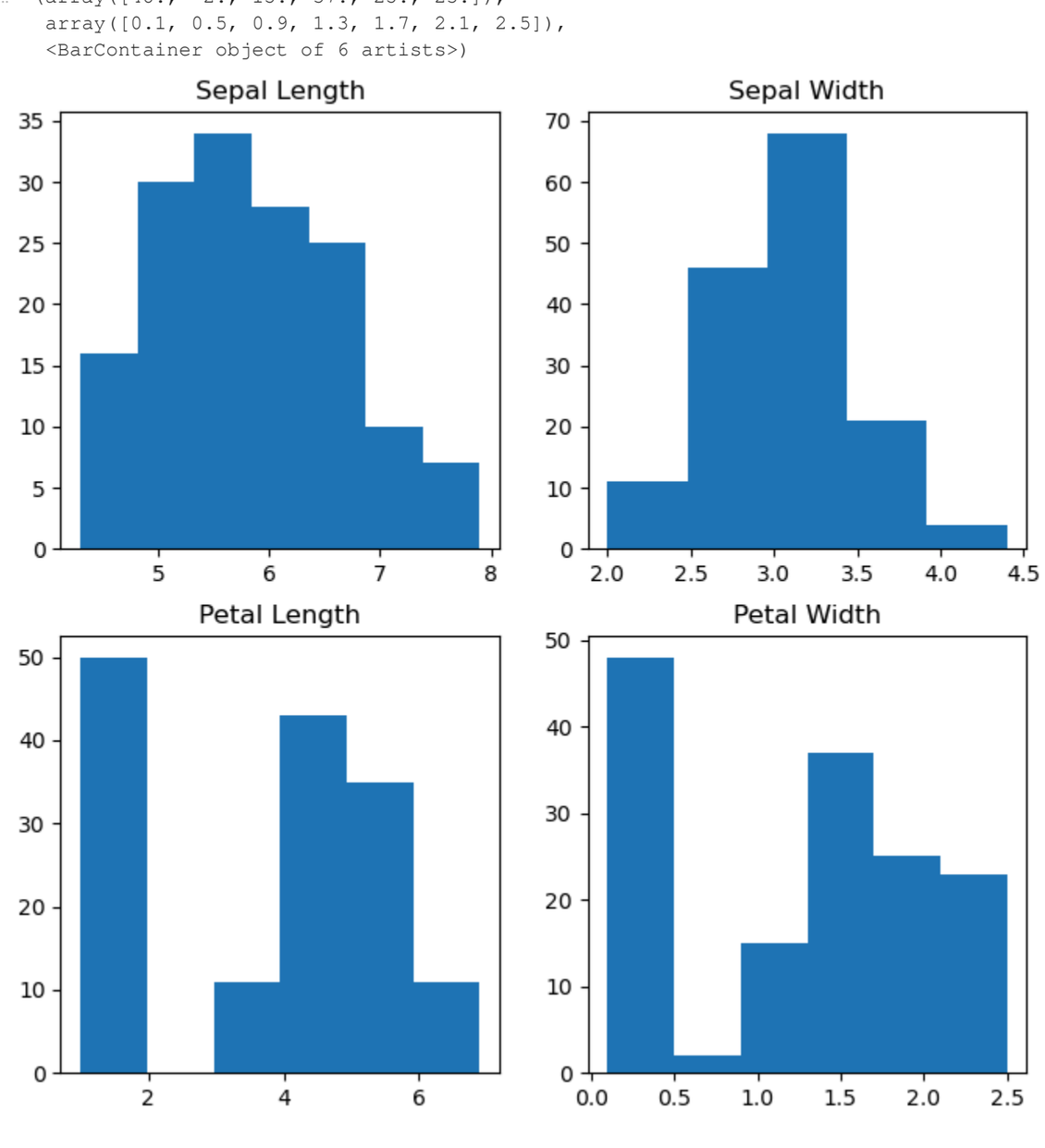
In [139…
```python
plt.figure(figsize=(10,10))
sns.pairplot(df, hue = 'variety')
plt.show()
```

```
<Figure size 1000x1000 with 0 Axes>
```



In [141…
```python
fig, axes = plt.subplots(2, 2, figsize=(8,8))

axes[0,0].set_title("Sepal Length")
axes[0,0].hist(df['SepalLength'], bins=7)

axes[0,1].set_title("Sepal Width")
axes[0,1].hist(df['SepalWidth'], bins=5)

axes[1,0].set_title("Petal Length")
axes[1,0].hist(df['PetalLength'], bins=6)

axes[1,1].set_title("Petal Width")
axes[1,1].hist(df['PetalWidth'], bins=6)
```

Out[141…
```
(array([48.,  2., 15., 37., 25., 23.]),
 array([0.1, 0.5, 0.9, 1.3, 1.7, 2.1, 2.5]),
 <BarContainer object of 6 artists>)
```



The highest frequency of the sepal length is between 30 and 35 which is between 5.5 and 6

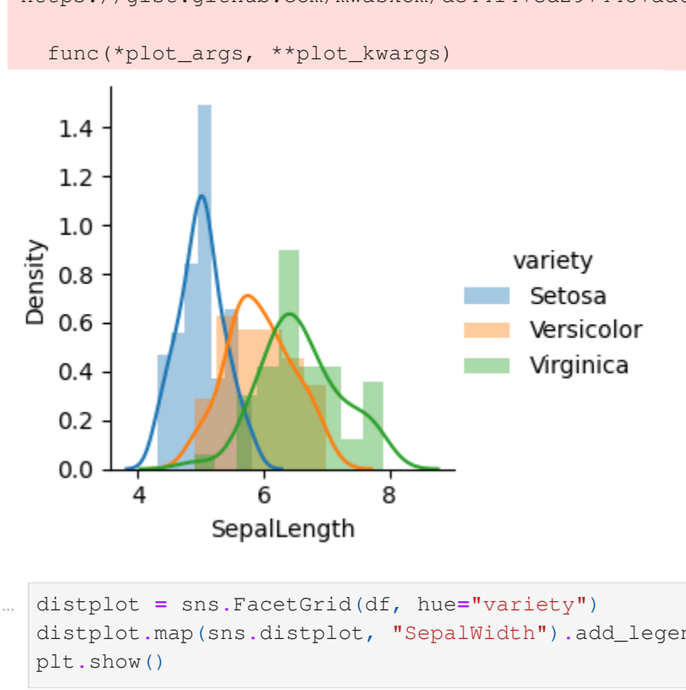
The highest frequency of the sepal Width is around 70 which is between 3.0 and 3.5

The highest frequency of the petal length is around 50 which is between 1 and 2

The highest frequency of the petal width is between 40 and 50 which is between 0.0 and 0.5

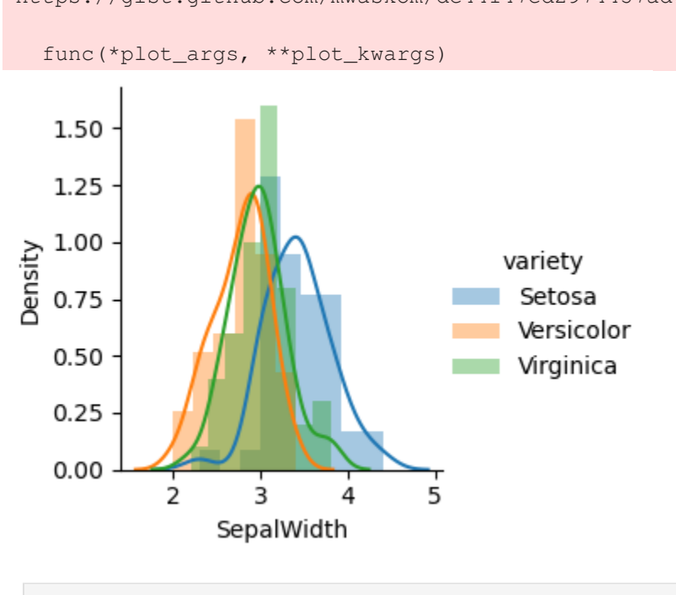
In [144…
```python
distplot = sns.FacetGrid(df, hue="variety")
distplot.map(sns.distplot, "SepalLength").add_legend()
plt.show()
```

```
C:\Users\jasik\anaconda3\Lib\site-packages\seaborn\axisgrid.py:854: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  func(*plot_args, **plot_kwargs)
C:\Users\jasik\anaconda3\Lib\site-packages\seaborn\axisgrid.py:854: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  func(*plot_args, **plot_kwargs)
C:\Users\jasik\anaconda3\Lib\site-packages\seaborn\axisgrid.py:854: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  func(*plot_args, **plot_kwargs)
```



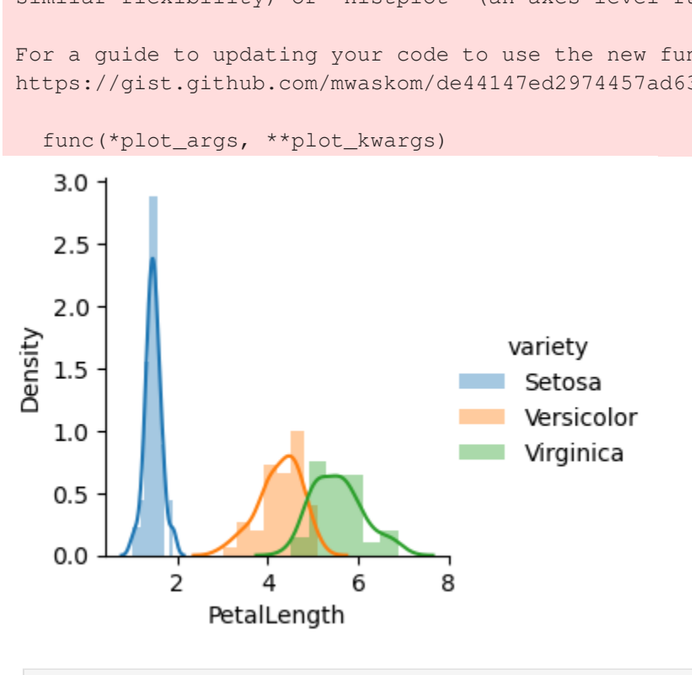
In [146…
```python
distplot = sns.FacetGrid(df, hue="variety")
distplot.map(sns.distplot, "SepalWidth").add_legend()
plt.show()
```

```
C:\Users\jasik\anaconda3\Lib\site-packages\seaborn\axisgrid.py:854: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  func(*plot_args, **plot_kwargs)
C:\Users\jasik\anaconda3\Lib\site-packages\seaborn\axisgrid.py:854: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  func(*plot_args, **plot_kwargs)
C:\Users\jasik\anaconda3\Lib\site-packages\seaborn\axisgrid.py:854: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  func(*plot_args, **plot_kwargs)
```



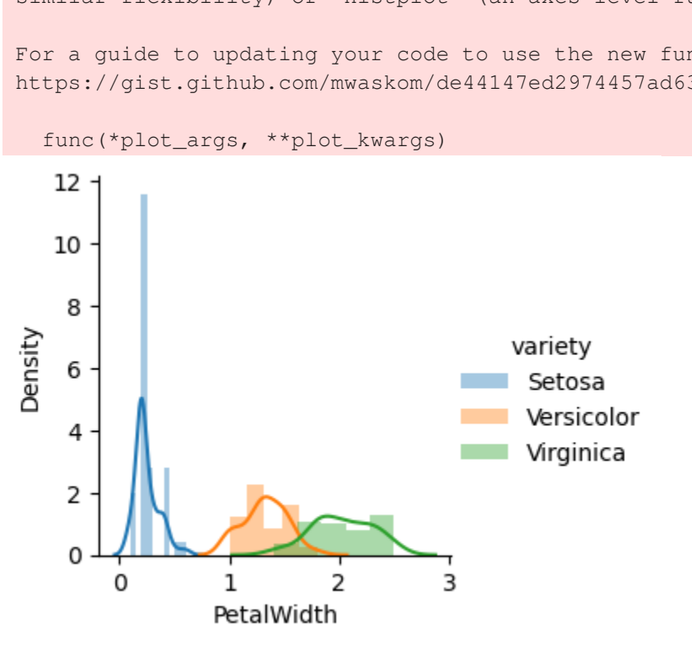
In [148…
```python
distplot = sns.FacetGrid(df, hue="variety")
distplot.map(sns.distplot, "PetalLength").add_legend()
plt.show()
```

```
C:\Users\jasik\anaconda3\Lib\site-packages\seaborn\axisgrid.py:854: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  func(*plot_args, **plot_kwargs)
C:\Users\jasik\anaconda3\Lib\site-packages\seaborn\axisgrid.py:854: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  func(*plot_args, **plot_kwargs)
C:\Users\jasik\anaconda3\Lib\site-packages\seaborn\axisgrid.py:854: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  func(*plot_args, **plot_kwargs)
```



In [150…
```python
distplot = sns.FacetGrid(df, hue="variety")
distplot.map(sns.distplot, "PetalWidth").add_legend()
plt.show()
```

```
C:\Users\jasik\anaconda3\Lib\site-packages\seaborn\axisgrid.py:854: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  func(*plot_args, **plot_kwargs)
C:\Users\jasik\anaconda3\Lib\site-packages\seaborn\axisgrid.py:854: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  func(*plot_args, **plot_kwargs)
C:\Users\jasik\anaconda3\Lib\site-packages\seaborn\axisgrid.py:854: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  func(*plot_args, **plot_kwargs)
```



Through the above graphs, we can see that there is a huge amount of overlapping in case of SepalLength and SepalWidth.

Similarly in case of PetalLength and PetalWidth, the overlapping reduces.

In [ ]:
```python
# Looking at the Corelation among the various variable.
```

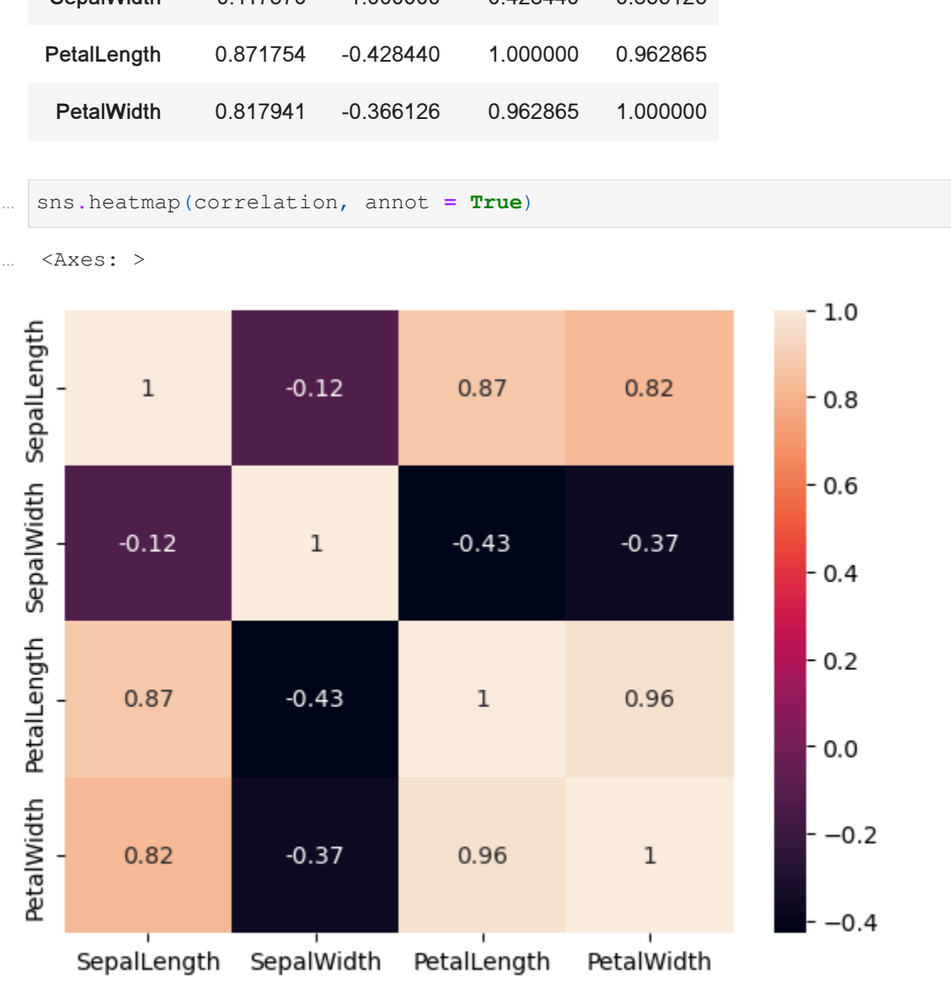
In [158…
```python
correlation = df.select_dtypes(include=['number']).corr()
correlation
```

Out[158…

	SepalLength	SepalWidth	PetalLength	PetalWidth
SepalLength	1.000000	-0.117570	0.871754	0.817941
SepalWidth	-0.117570	1.000000	-0.428440	-0.366126
PetalLength	0.871754	-0.428440	1.000000	0.962865
PetalWidth	0.817941	-0.366126	0.962865	1.000000

In [160…
```python
sns.heatmap(correlation, annot = True)
```

Out[160… <Axes: >



Obseravtion

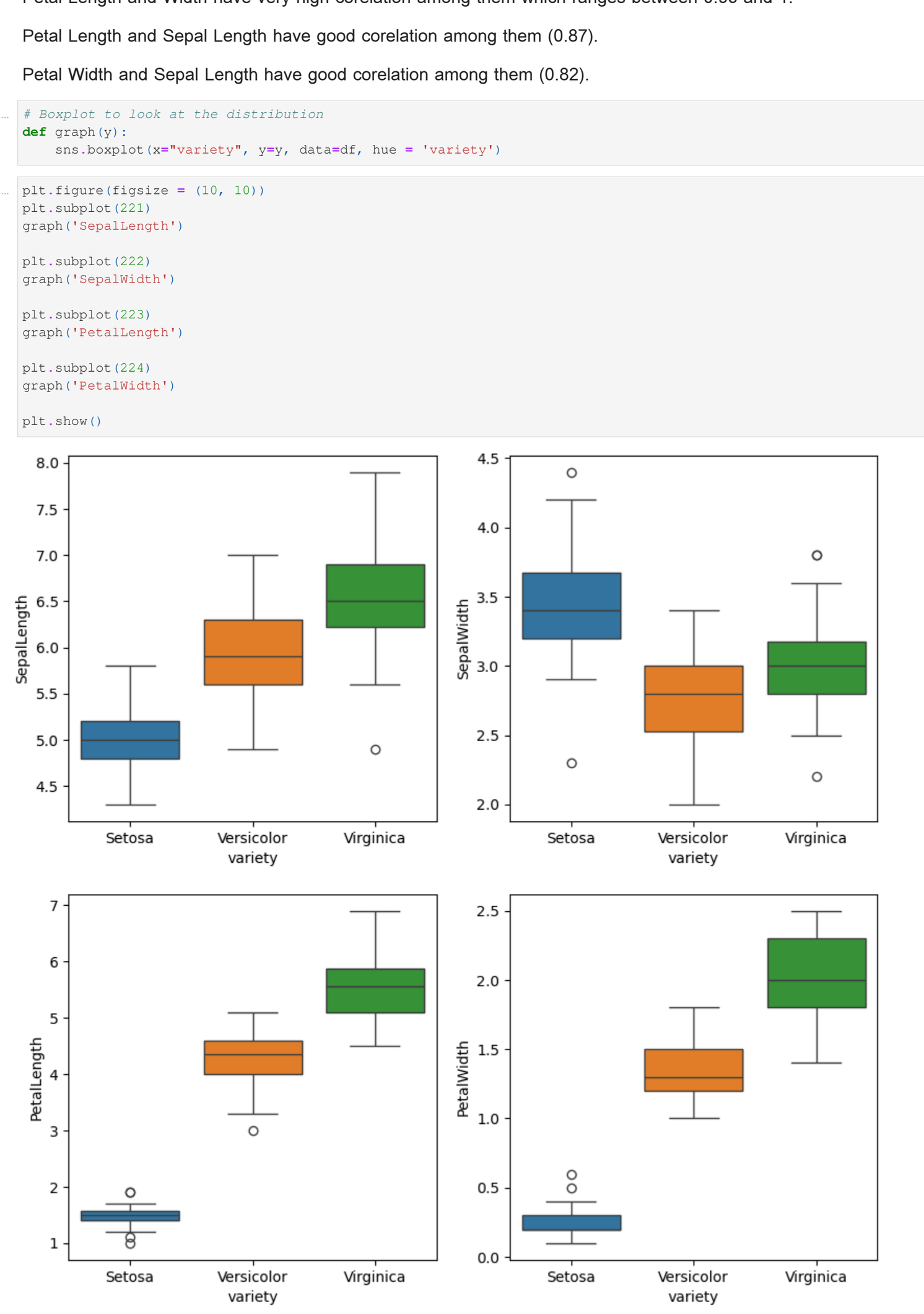
Petal Length and Width have very high corelation among them which ranges between 0.96 and 1.

Petal Length and Sepal Length have good corelation among them (0.87).

Petal Width and Sepal Length have good corelation among them (0.82).

In [172…
```python
# Boxplot to look at the distribution
def graph(y):
    sns.boxplot(x="variety", y=y, data=df, hue = 'variety')
```

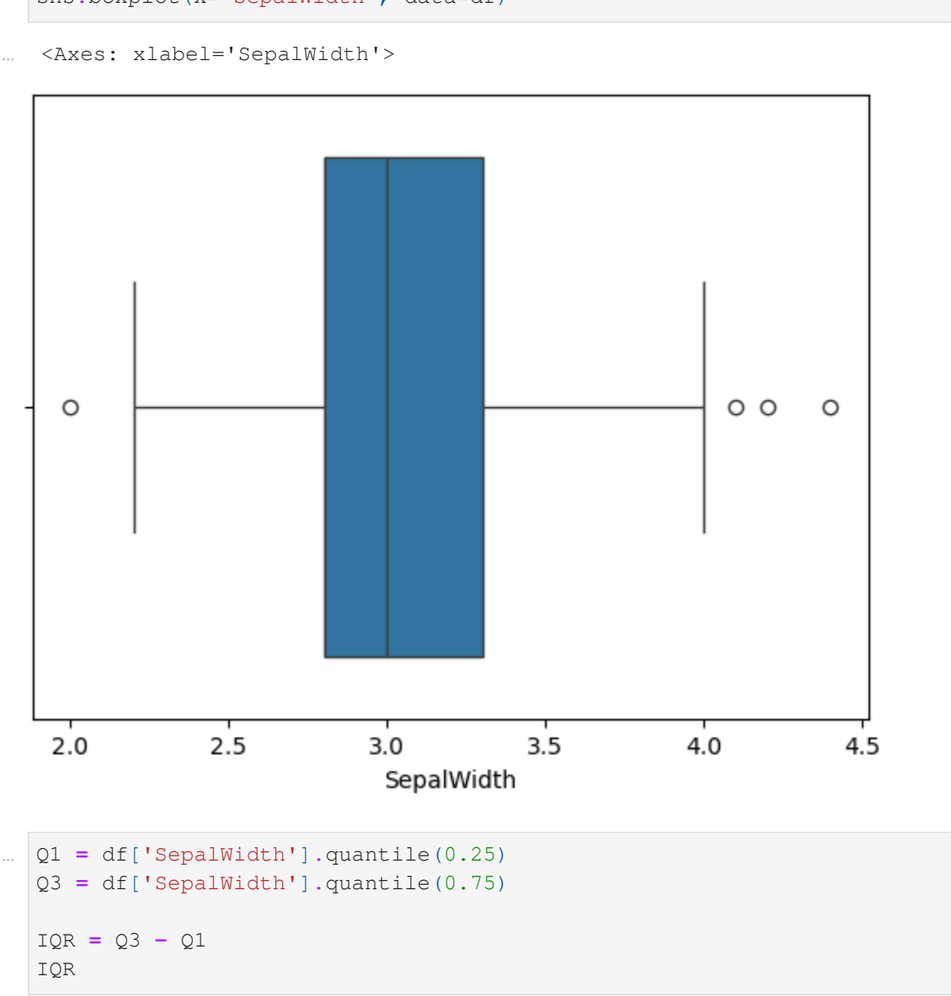
In [174…
```python
plt.figure(figsize = (10, 10))
plt.subplot(221)
graph('SepalLength')

plt.subplot(222)
graph('SepalWidth')

plt.subplot(223)
graph('PetalLength')

plt.subplot(224)
graph('PetalWidth')

plt.show()
```



Species Setosa has the smallest features and less distributed with some outliers whereas Virginica has the highest features.

In [179…
```python
# Looking for Outliers anf handling them

sns.boxplot(x='SepalWidth', data=df)
```

Out[179… <Axes: xlabel='SepalWidth'>



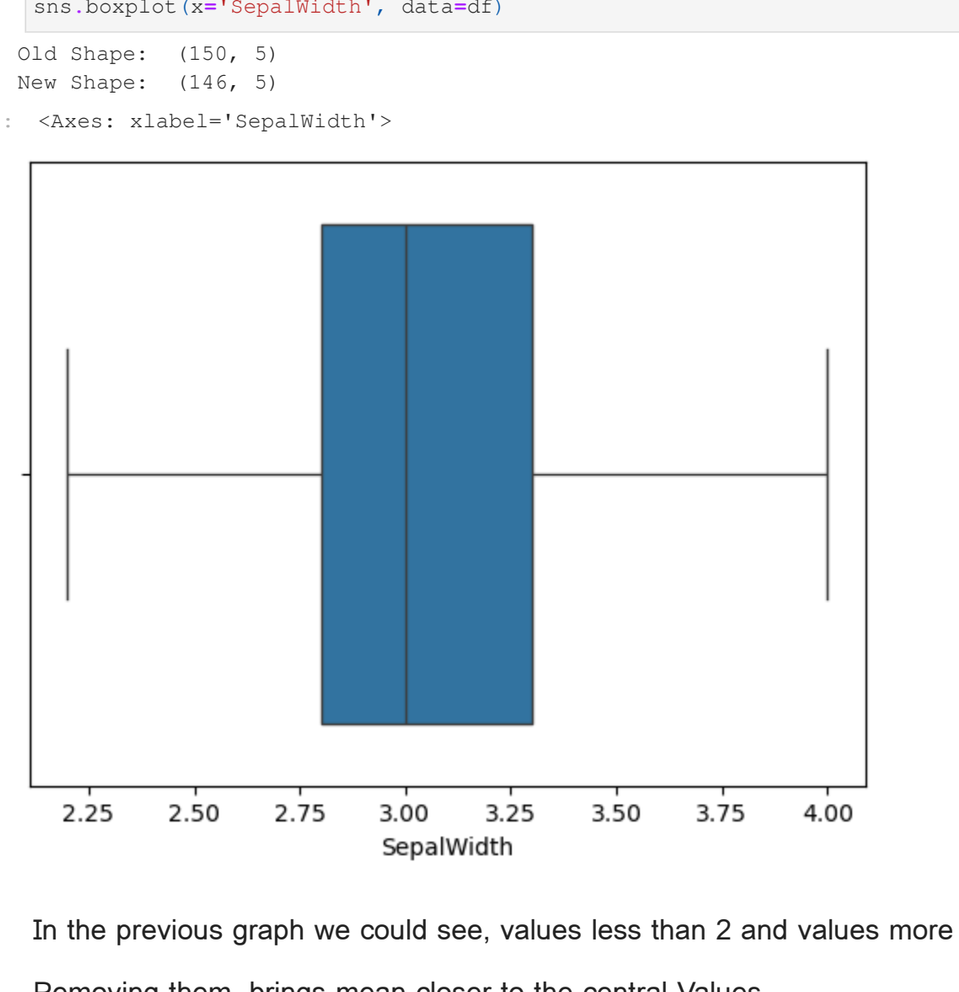
In [181…
```python
Q1 = df['SepalWidth'].quantile(0.25)
Q3 = df['SepalWidth'].quantile(0.75)

IQR = Q3 - Q1
IQR
```

Out[181… 0.5

In [91]:
```python
print("Old Shape: ", df.shape)
upper = np.where(df['SepalWidth'] >= (Q3+1.5*IQR))
lower = np.where(df['SepalWidth'] <= (Q1-1.5*IQR))

# Removing the Outliers
df.drop(upper[0], inplace = True)
df.drop(lower[0], inplace = True)

print("New Shape: ", df.shape)

sns.boxplot(x='SepalWidth', data=df)
```

```
Old Shape:  (150, 5)
New Shape:  (146, 5)
```

Out[91]: <Axes: xlabel='SepalWidth'>



In the previous graph we could see, values less than 2 and values more than 4 are acting as Outliers.

Removing them, brings mean closer to the central Values.

Project By : JASIKA GUPTA

LinkedIn : https://www.linkedin.com/in/jasika-gupta17/

GitHub : https://github.com/abhaysinh/Surveys-and-Research-Papers-On-Text-Summarization-with-Deep-Learning

Thank You!