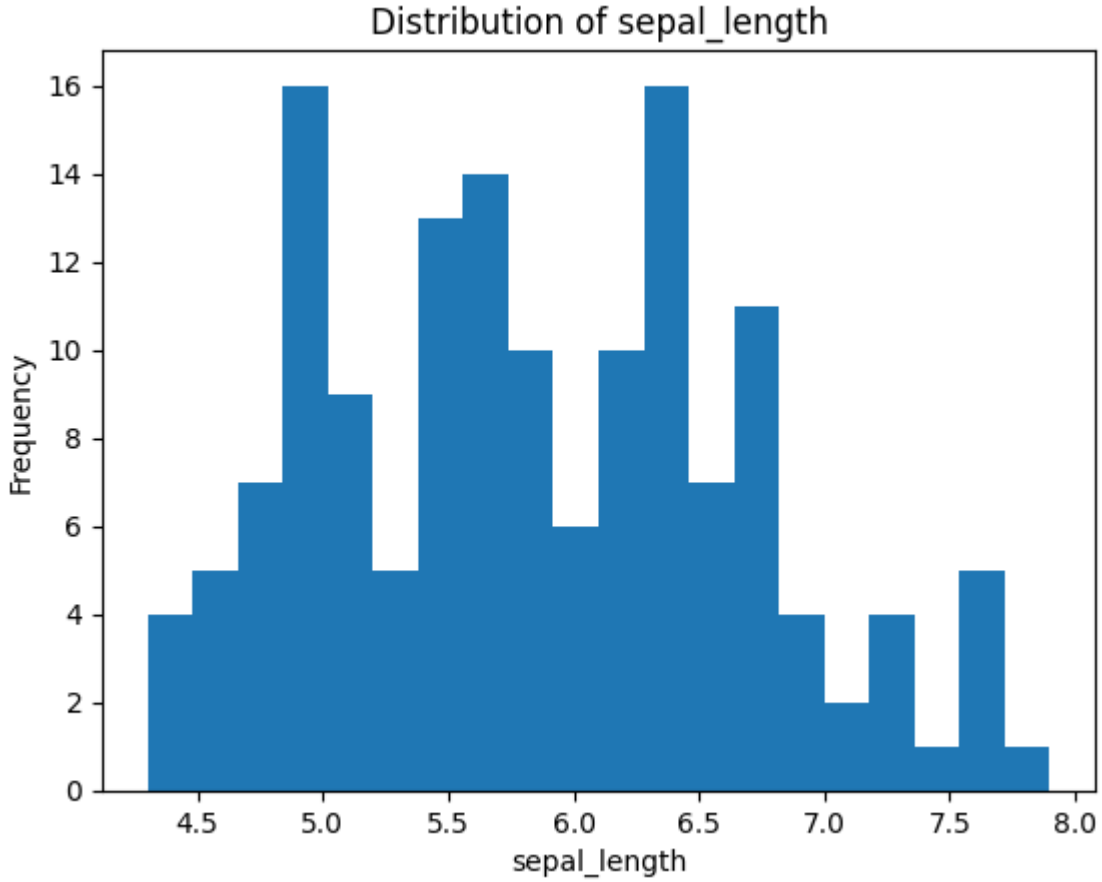
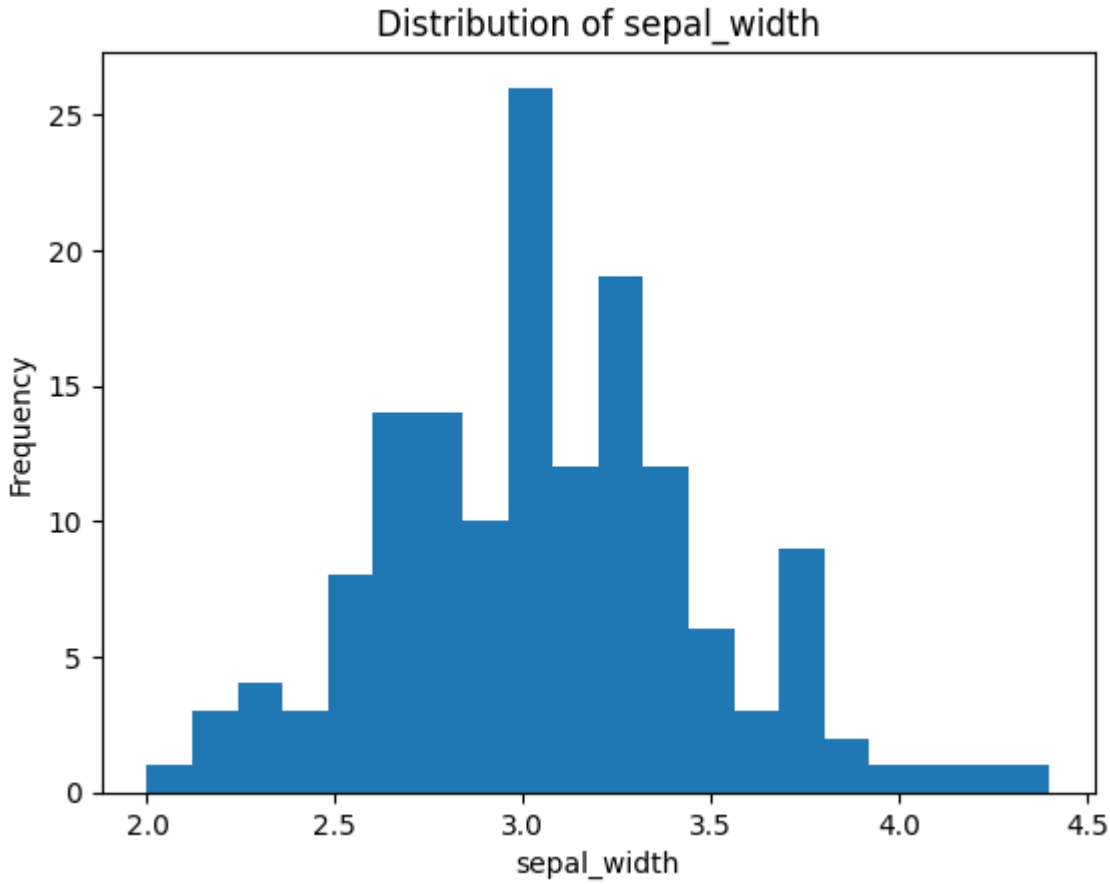
In [ ]:
```
# Title: Data Visualization-III

import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns


# Read the data from the file
df = pd.read_csv('data.csv')
```

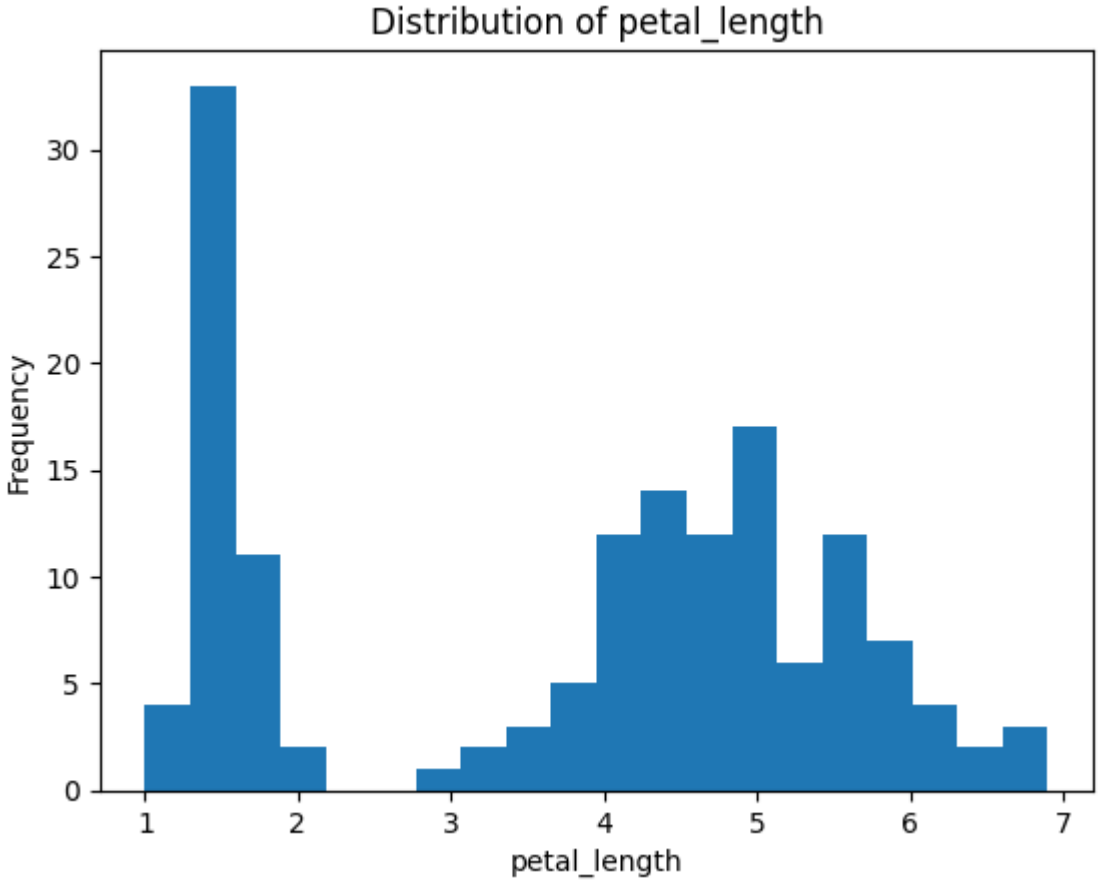
In [ ]:
```
# 1. List down the features and their types (e.g., numeric, nominal) available i
# Feature names and types
print("Features:")
for feature in df.columns:
    if df[feature].dtype == 'int64' or df[feature].dtype == 'float64':
        print(f"Type of {feature}: Numeric")
    else:
        print(f"Type of {feature}: Nominal")
    if df[feature].dtype == 'object':
        print(f"  Unique values: {df[feature].unique()}")
```

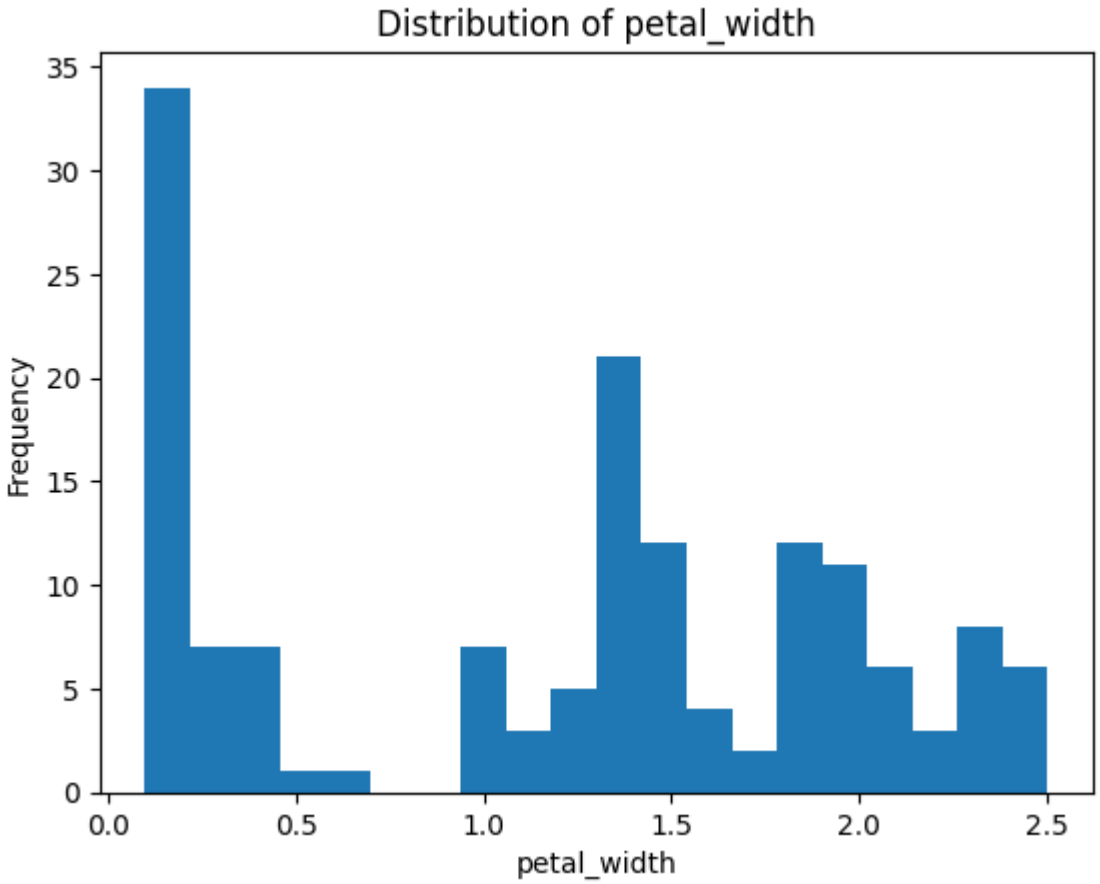
```
Features:
Type of sepal_length: Numeric
Type of sepal_width: Numeric
Type of petal_length: Numeric
Type of petal_width: Numeric
Type of class: Nominal
  Unique values: ['Iris-setosa' 'Iris-versicolor' 'Iris-virginica']
```

In [ ]:
```
# 2. Create a histogram for each feature in the dataset to illustrate the featur

for feature in df.columns:
    if df[feature].dtype == 'int64' or df[feature].dtype == 'float64':
        plt.hist(df[feature], bins=20)
        plt.title(f"Distribution of {feature}")
        plt.xlabel(feature)
        plt.ylabel("Frequency")
        plt.show()
```


Distribution of sepal_length
Frequency
sepal_length
0
2
4
6
8
10
12
14
16
4.5
5.0
5.5
6.0
6.5
7.0
7.5
8.0

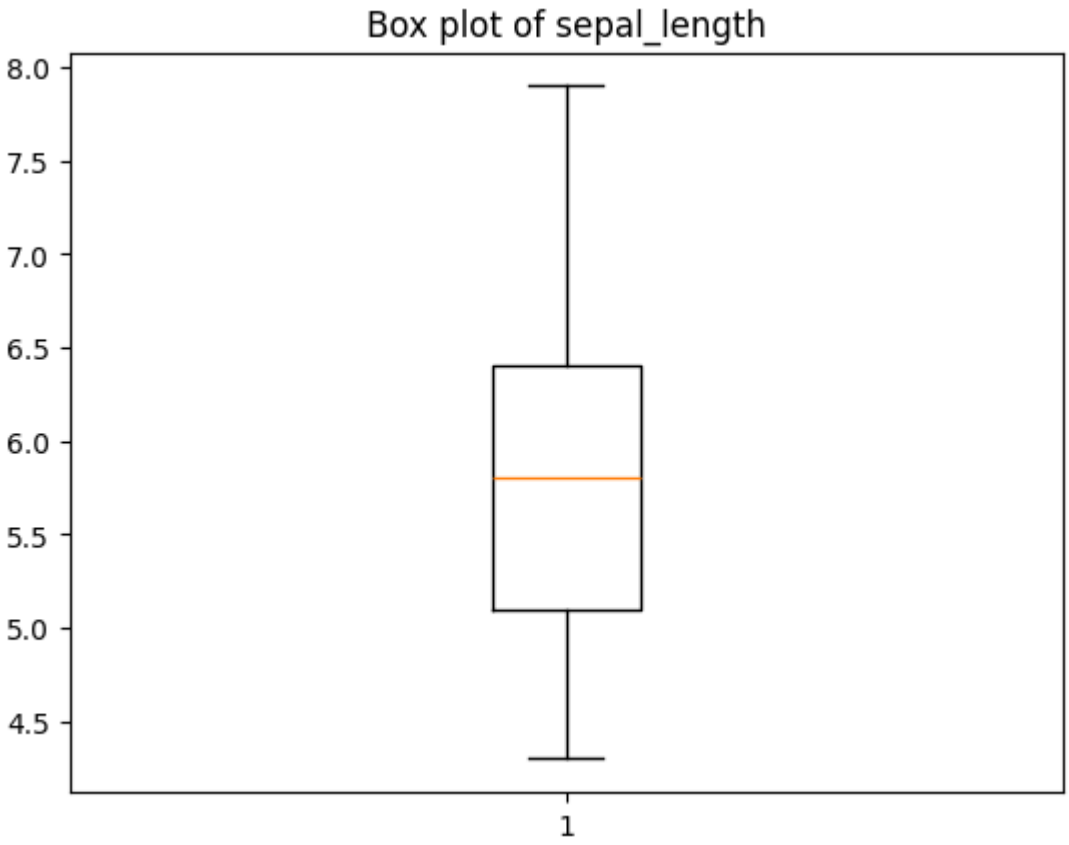

Distribution of sepal_width
Frequency
sepal_width
0
5
10
15
20
25
2.0
2.5
3.0
3.5
4.0
4.5

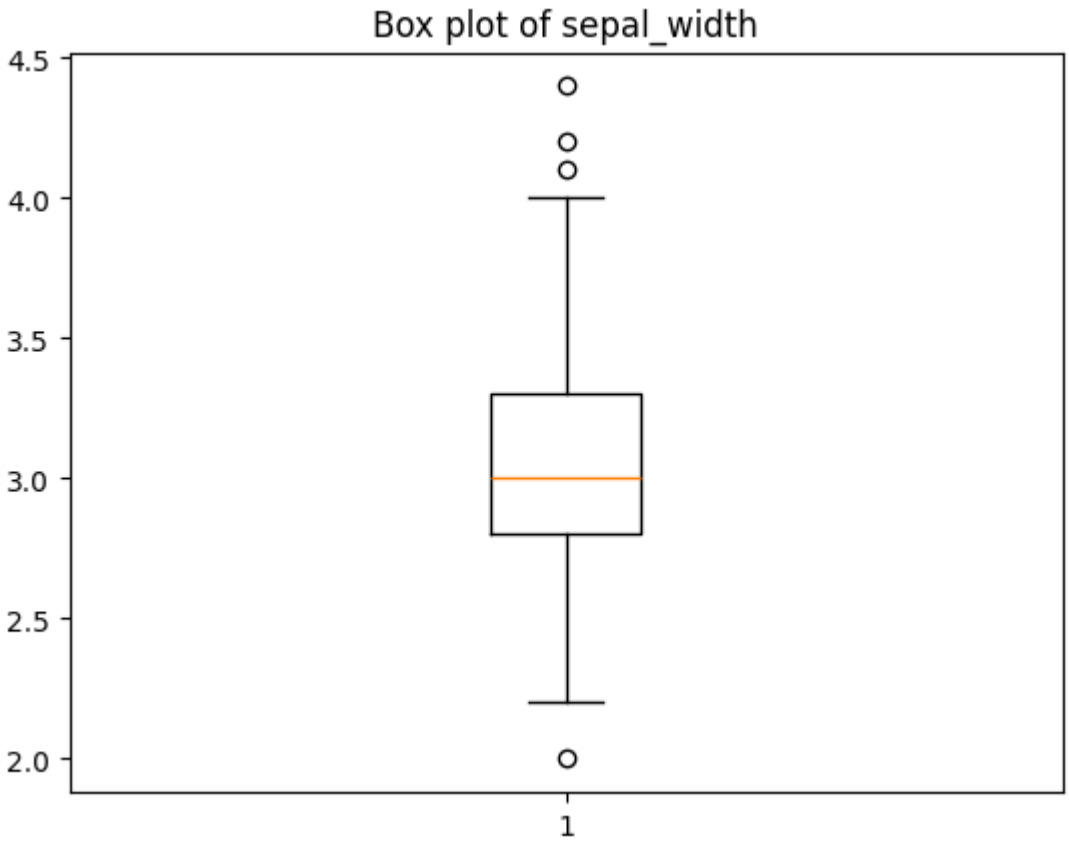


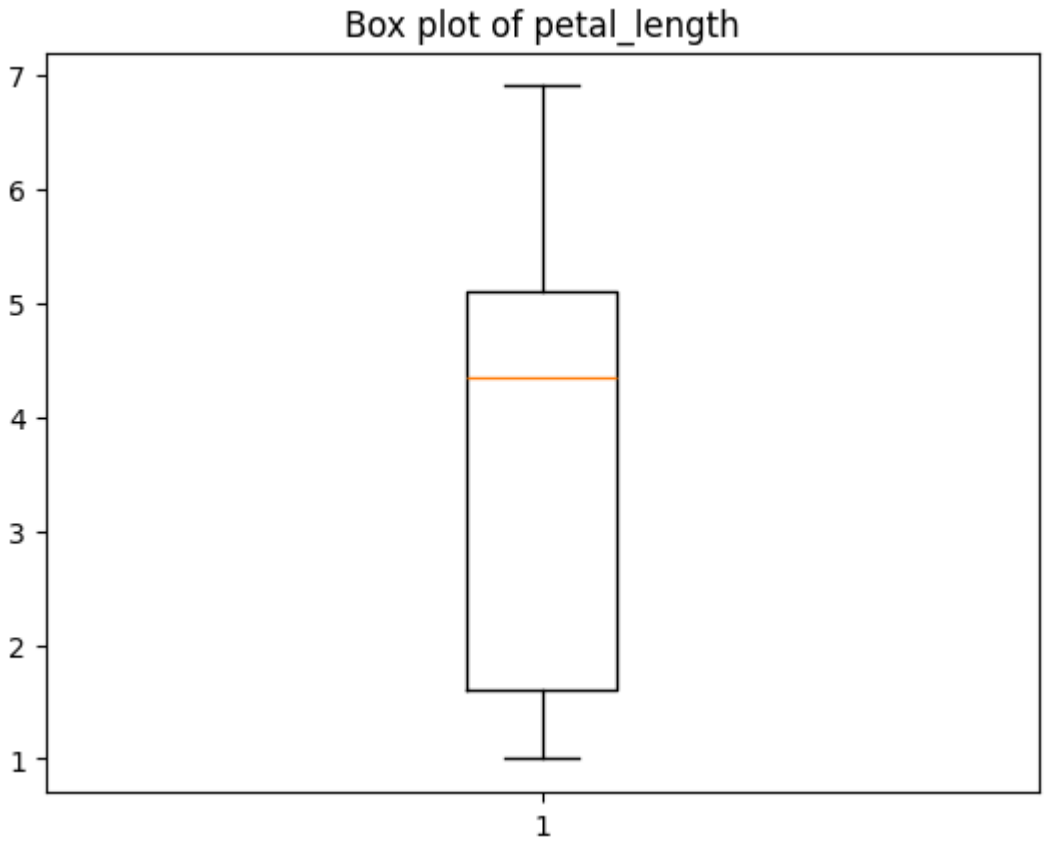


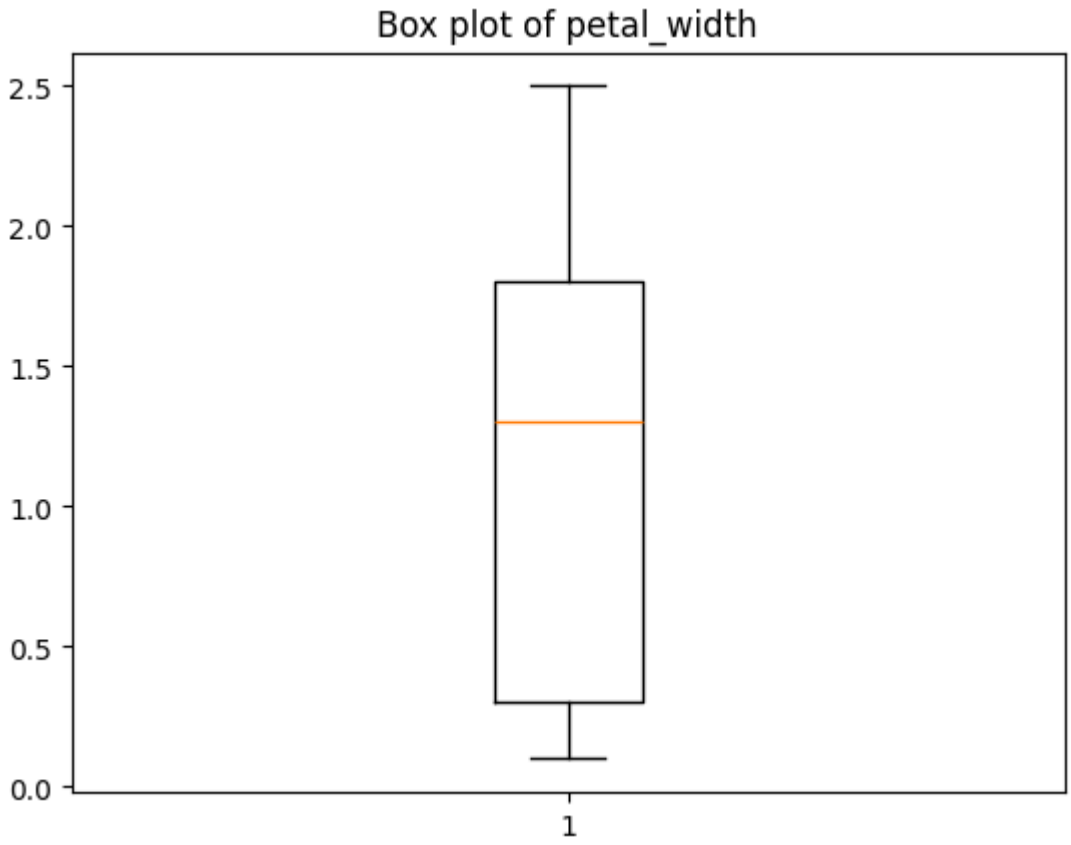
In [ ]:
```python
# 3. Create a box plot for each feature in the dataset.

for feature in df.columns:
    if df[feature].dtype == 'int64' or df[feature].dtype == 'float64':
        plt.boxplot(df[feature])
```

```
        plt.title(f"Box plot of {feature}")
        plt.show()
```

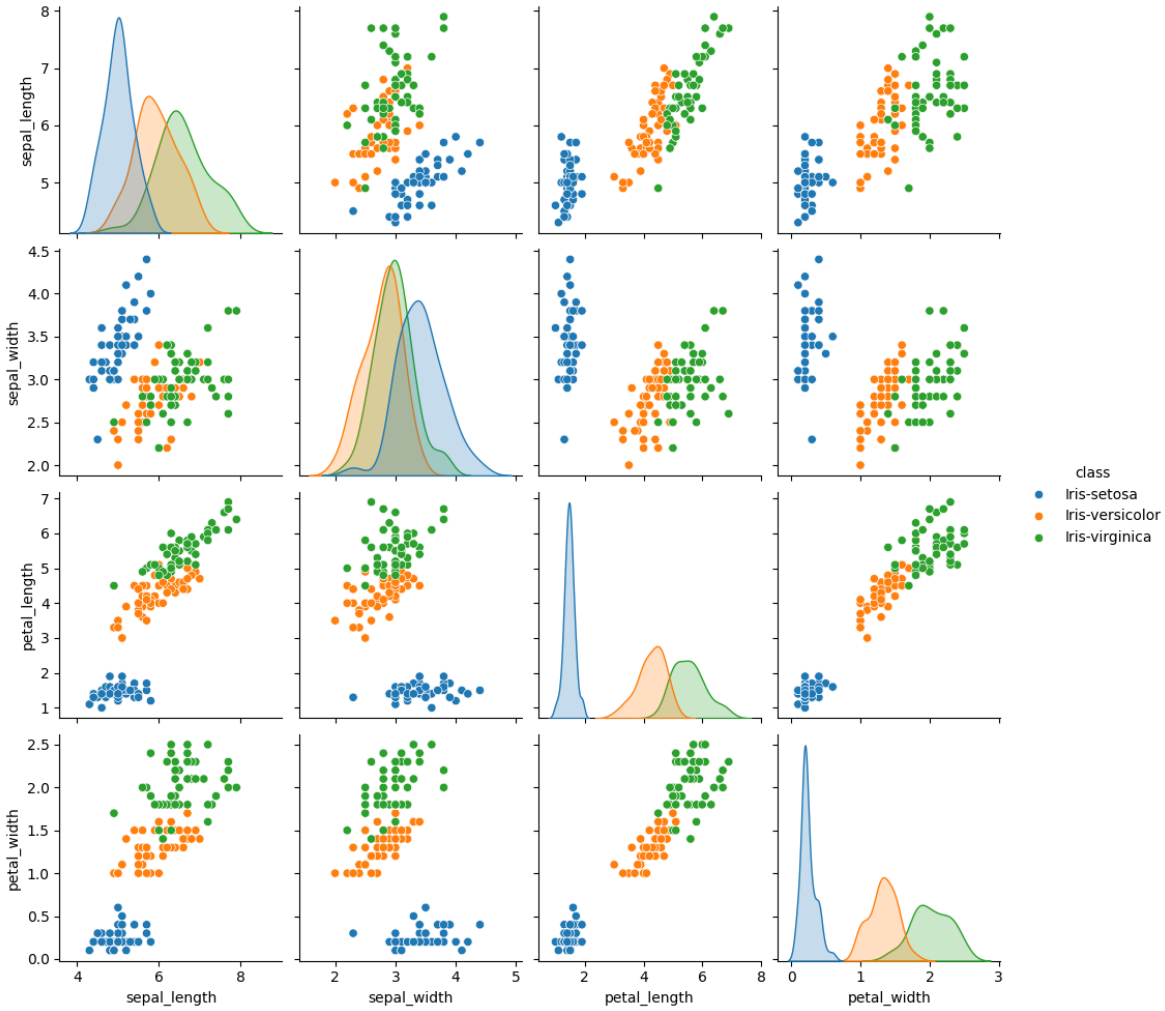








In [ ]:
```
# 4. Compare distributions and identify outliers.

sns.pairplot(df, hue='class')
plt.show()
```



In [ ]:
```
# Find Outliers numerically
for feature in df.columns:
    if df[feature].dtype == 'int64' or df[feature].dtype == 'float64':
        Q1 = df[feature].quantile(0.25)
        Q3 = df[feature].quantile(0.75)
        IQR = Q3 - Q1
        print(f"Outliers for {feature}: {df[(df[feature] < (Q1 - 1.5 * IQR)) | (
```

```
Outliers for sepal_length: Empty DataFrame
Columns: [sepal_length, sepal_width, petal_length, petal_width, class]
Index: []
Outliers for sepal_width:     sepal_length  sepal_width  petal_length  petal_widt
h            class
15            5.7          4.4           1.5           0.4      Iris-setosa
32            5.2          4.1           1.5           0.1      Iris-setosa
33            5.5          4.2           1.4           0.2      Iris-setosa
60            5.0          2.0           3.5           1.0  Iris-versicolor
Outliers for petal_length: Empty DataFrame
Columns: [sepal_length, sepal_width, petal_length, petal_width, class]
Index: []
Outliers for petal_width: Empty DataFrame
Columns: [sepal_length, sepal_width, petal_length, petal_width, class]
Index: []
```