

```
In [ ]: '''  
Data Visualization III  
  
Download the Iris flower dataset or any other dataset into a DataFrame  
(https://archive.ics.uci.edu/ml/datasets/Iris ). Scan the dataset and  
  
1. List down the features and their types (e.g., numeric, nominal) and  
2. Create a histogram for each feature in the dataset to illustrate the  
3. Create a boxplot for each feature in the dataset.  
4. Compare distributions and identify outliers.  
'''
```

```
In [12]: import pandas as pd  
  
import seaborn as sns  
import matplotlib.pyplot as plt  
import numpy as np
```

```
In [3]: df = pd.read_csv('IRIS.csv')
```

```
In [4]: df
```

Out[4]:

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	Iris-setosa
1	4.9	3.0	1.4	0.2	Iris-setosa
2	4.7	3.2	1.3	0.2	Iris-setosa
3	4.6	3.1	1.5	0.2	Iris-setosa
4	5.0	3.6	1.4	0.2	Iris-setosa
...
145	6.7	3.0	5.2	2.3	Iris-virginica
146	6.3	2.5	5.0	1.9	Iris-virginica
147	6.5	3.0	5.2	2.0	Iris-virginica
148	6.2	3.4	5.4	2.3	Iris-virginica
149	5.9	3.0	5.1	1.8	Iris-virginica

150 rows × 5 columns

```
In [5]: df.describe()
```

```
Out[5]:
```

	sepal_length	sepal_width	petal_length	petal_width
count	150.000000	150.000000	150.000000	150.000000
mean	5.843333	3.054000	3.758667	1.198667
std	0.828066	0.433594	1.764420	0.763161
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 [7]: df.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   species         150 non-null   object  
dtypes: float64(4), object(1)
memory usage: 6.0+ KB
```

```
In [9]: df.dtypes
```

```
Out[9]: sepal_length    float64
sepal_width    float64
petal_length    float64
petal_width    float64
species         object
dtype: object
```

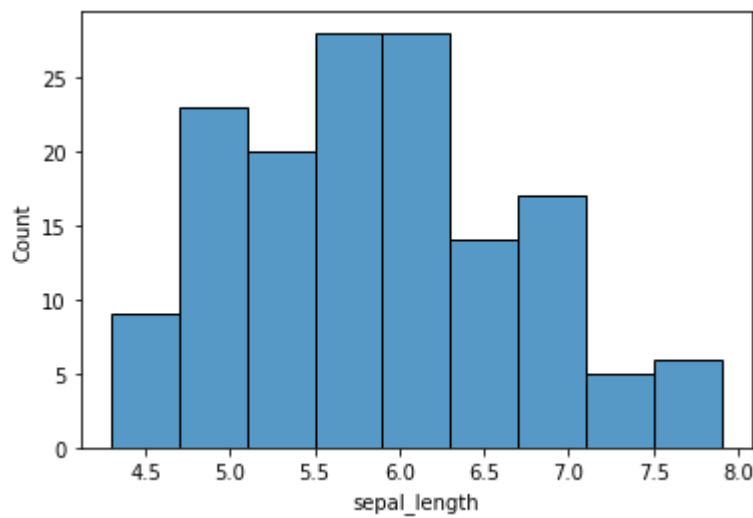
```
In [13]: np.unique(df['species'])
```

```
Out[13]: array(['Iris-setosa', 'Iris-versicolor', 'Iris-virginica'], dtype=object)
```

```
In [15]: # Create a histogram for each feature in the dataset to illustrate the distribution of each feature
# fig, axes = plt.subplots(2, 2, figsize=(16,8)) # 4 graphs plotted
```

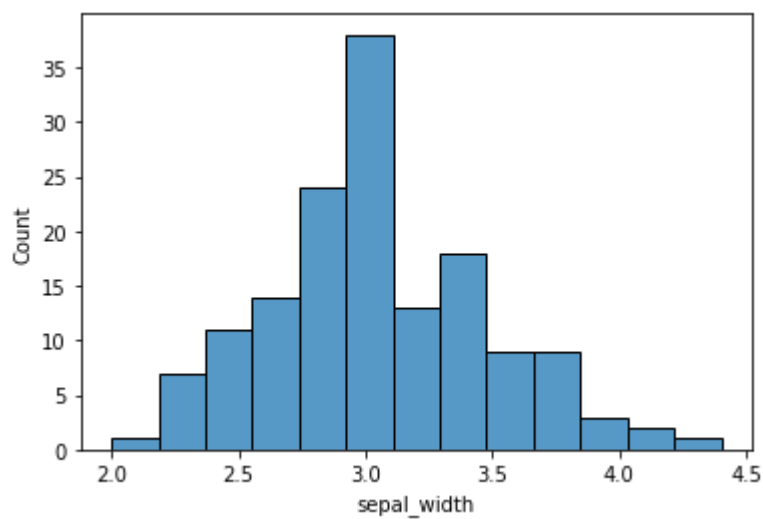
```
In [17]: sns.histplot(df['sepal_length'])
```

```
Out[17]: <AxesSubplot: xlabel='sepal_length', ylabel='Count'>
```



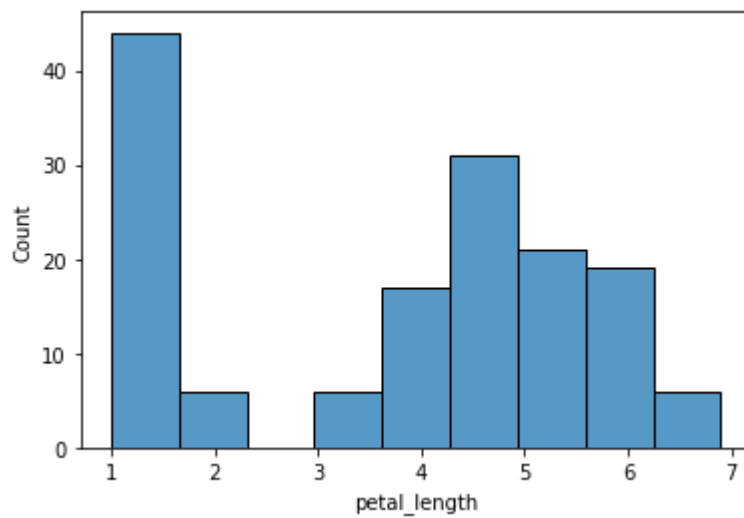
```
In [18]: sns.histplot(df['sepal_width'])
```

```
Out[18]: <AxesSubplot: xlabel='sepal_width', ylabel='Count'>
```



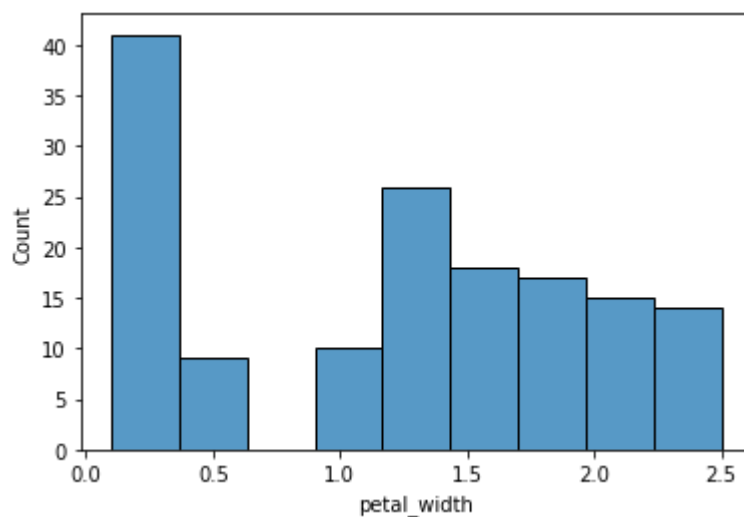
```
In [19]: sns.histplot(df['petal_length'])
```

```
Out[19]: <AxesSubplot: xlabel='petal_length', ylabel='Count'>
```



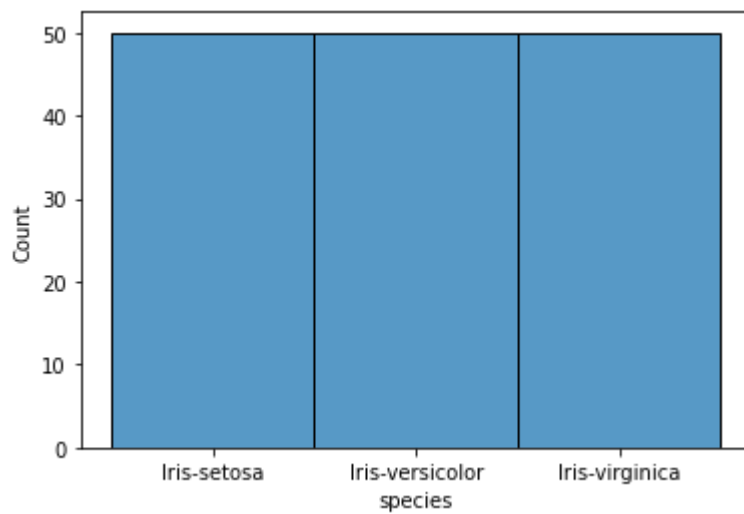
```
In [20]: sns.histplot(df['petal_width'])
```

```
Out[20]: <AxesSubplot: xlabel='petal_width', ylabel='Count'>
```



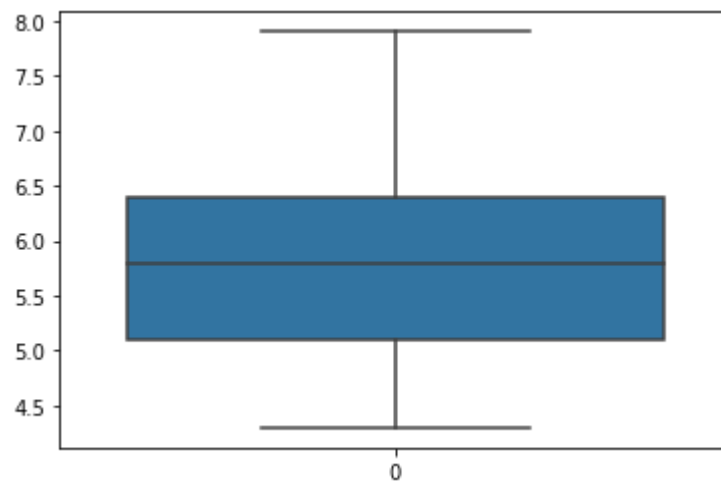
```
In [21]: sns.histplot(df['species'])
```

```
Out[21]: <AxesSubplot: xlabel='species', ylabel='Count'>
```



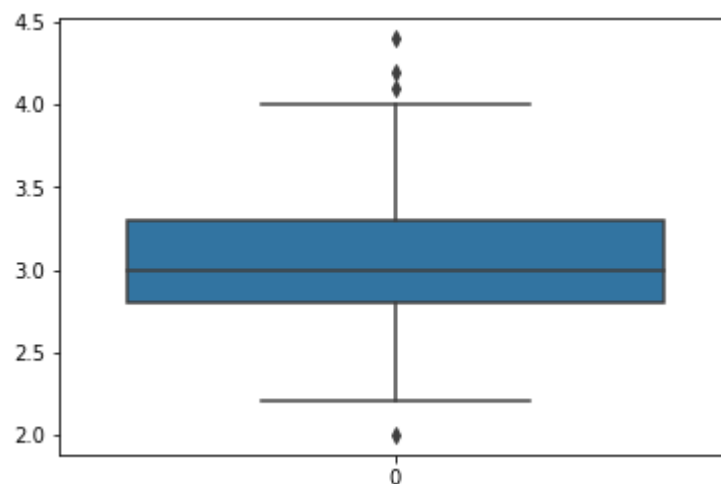
```
In [26]: sns.boxplot(df['sepal_length'])
```

```
Out[26]: <AxesSubplot: >
```



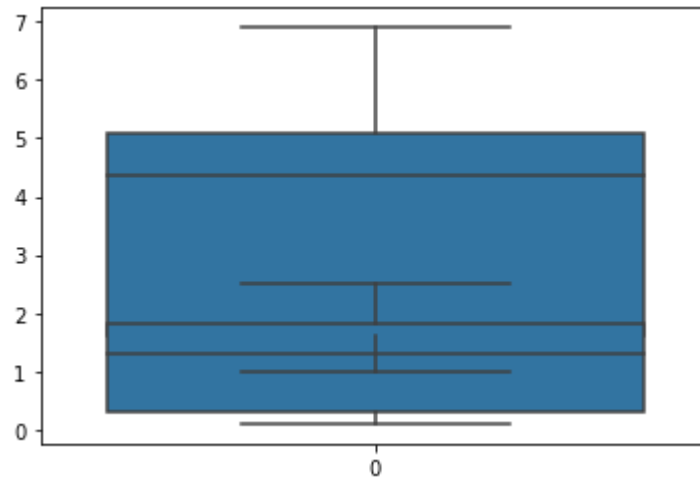
```
In [27]: sns.boxplot(df['sepal_width'])
```

```
Out[27]: <AxesSubplot: >
```



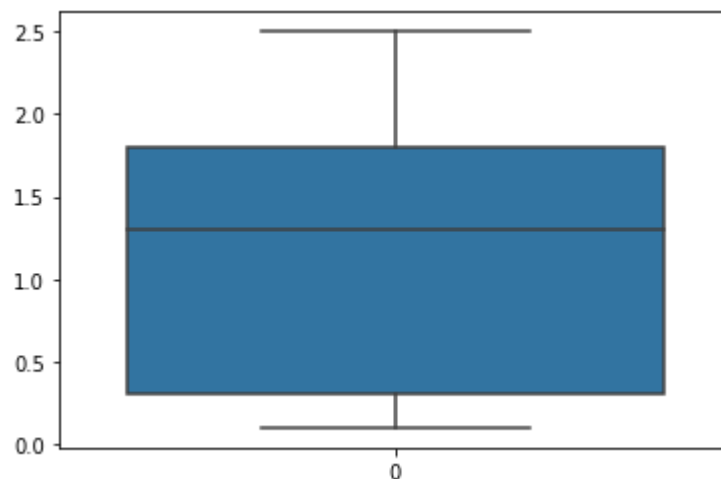
```
In [24]: sns.boxplot(df['petal_length'])
```

```
Out[24]: <AxesSubplot: >
```



```
In [25]: sns.boxplot(df['petal_width'])
```

```
Out[25]: <AxesSubplot: >
```



```
In [29]: # sns.boxplot(df['species']) -> not valid due to datatype
```

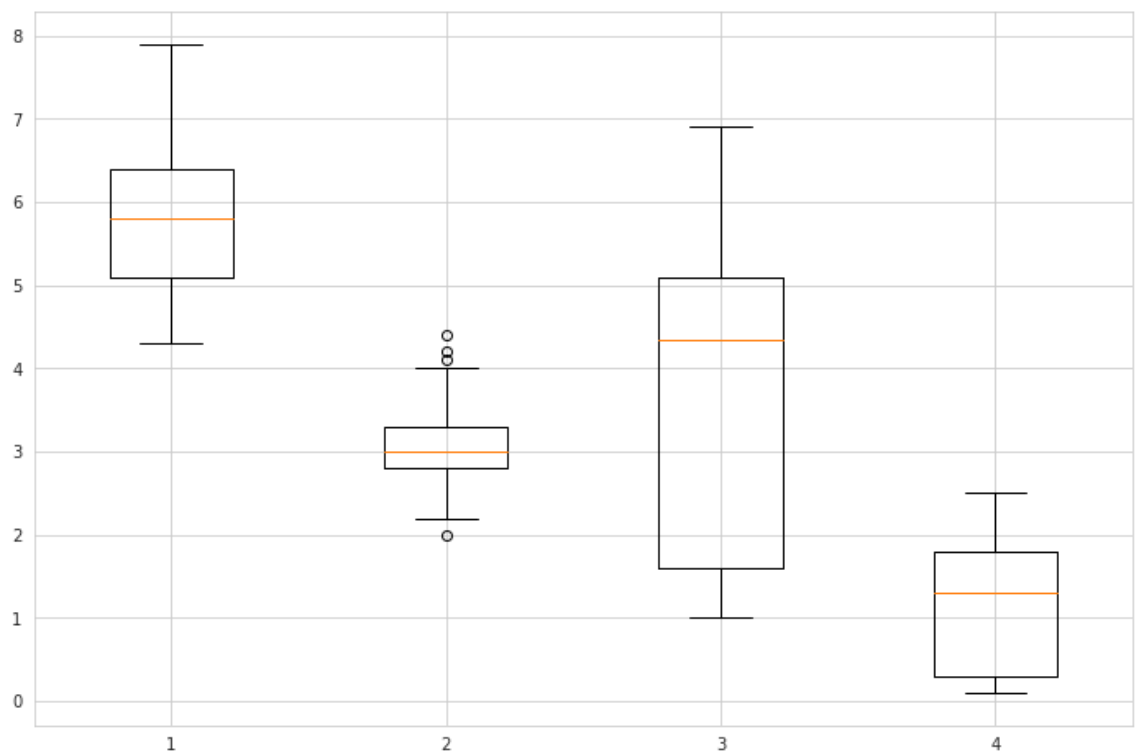
In [37]: *# comparing distributions and identifying outliers*

```
data_to_plot = [df['sepal_length'],df['sepal_width'],df['petal_length']

# sns.set_style("whitegrid")
# Creating a figure instance
fig = plt.figure(1, figsize=(12,8)) # 1 is the unique identifier of the figure
# 12,8 is the width and height in inches

# Creating an axes instance
ax = fig.add_subplot(111) # 111 means 1x1 grid, 1st subplot

# Creating the boxplot
bp = ax.boxplot(data_to_plot);
```



In []: