Assignment -10 -Data Visualization 3

Kaustubh Shrikant Kabra

ERP Number :- 38

TE Comp 1

Download the Iris flower dataset or any other dataset into a DataFrame. (e.g., https://archive.ics.uci.edu/ml/datasets/Iris). Scan the dataset and give the inference as:

1. List down the features and their types (e.g., numeric, nominal) available in the dataset.
2. Create a histogram for each feature in the dataset to illustrate the feature distributions.
3. Create a box plot for each feature in the dataset.
4. Compare distributions and identify outliers.

In [19]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [20]:
```
data=pd.read_csv('iris flower.csv')
```

In [21]:
```
data.head()
```

Out[21]:

	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

In [22]:
```
data.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
```

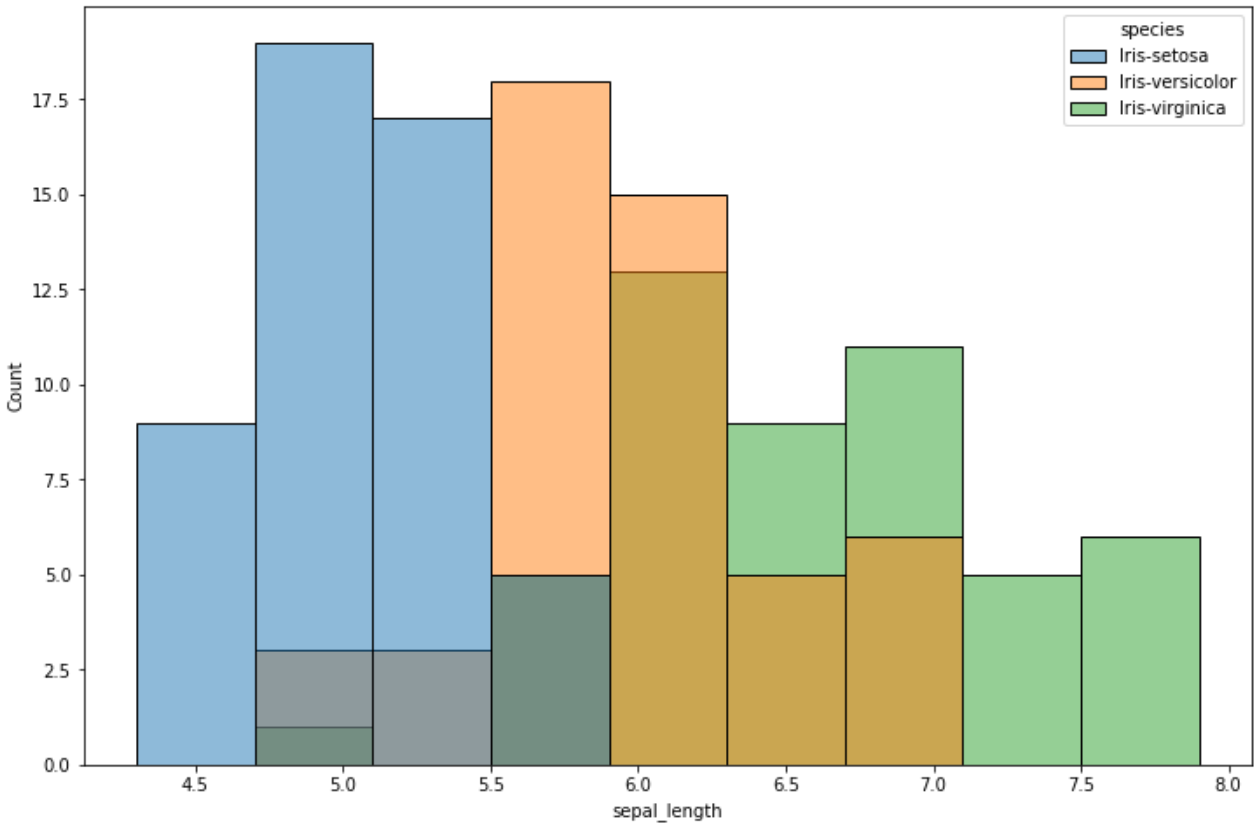
In [23]:
```
data.describe()
```

Out[23]:

	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

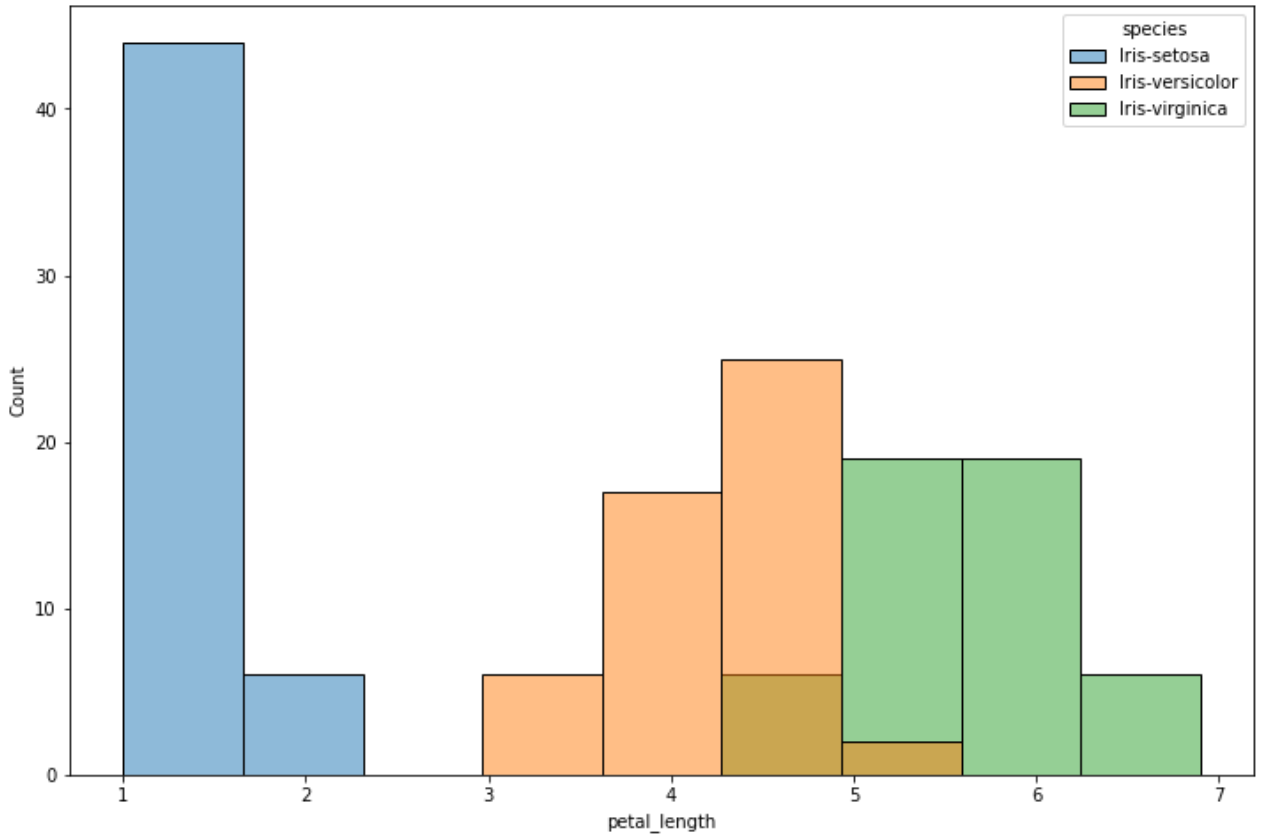
In [24]:
```
plt.figure(figsize=(12,8))
sns.histplot(x=data['sepal_length'], hue=data['species'])
plt.plot()
```

Out[24]:
```
[]
```



In [25]:
```
plt.figure(figsize=(12,8))
sns.histplot(x=data['petal_length'], hue=data['species'])
plt.plot()
```

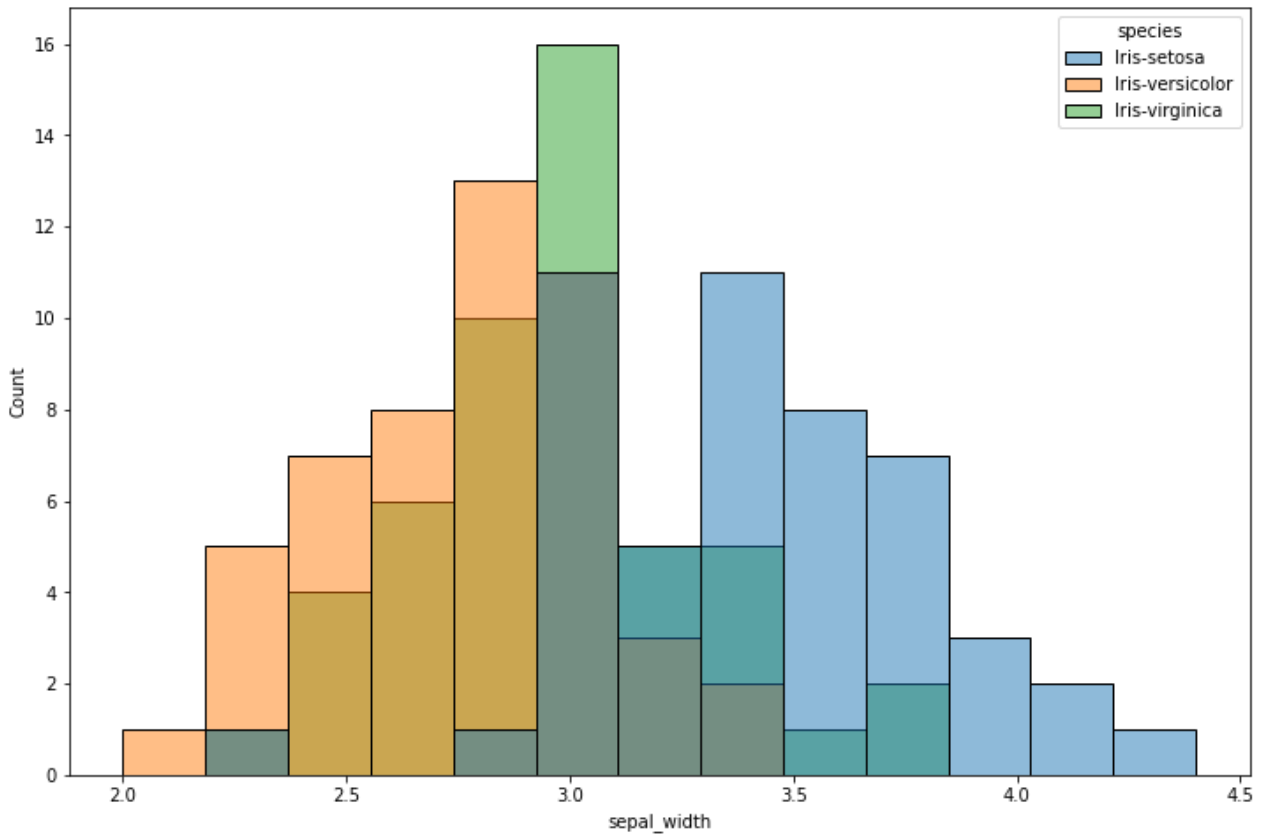
Out[25]:
```
[]
```



In [26]:

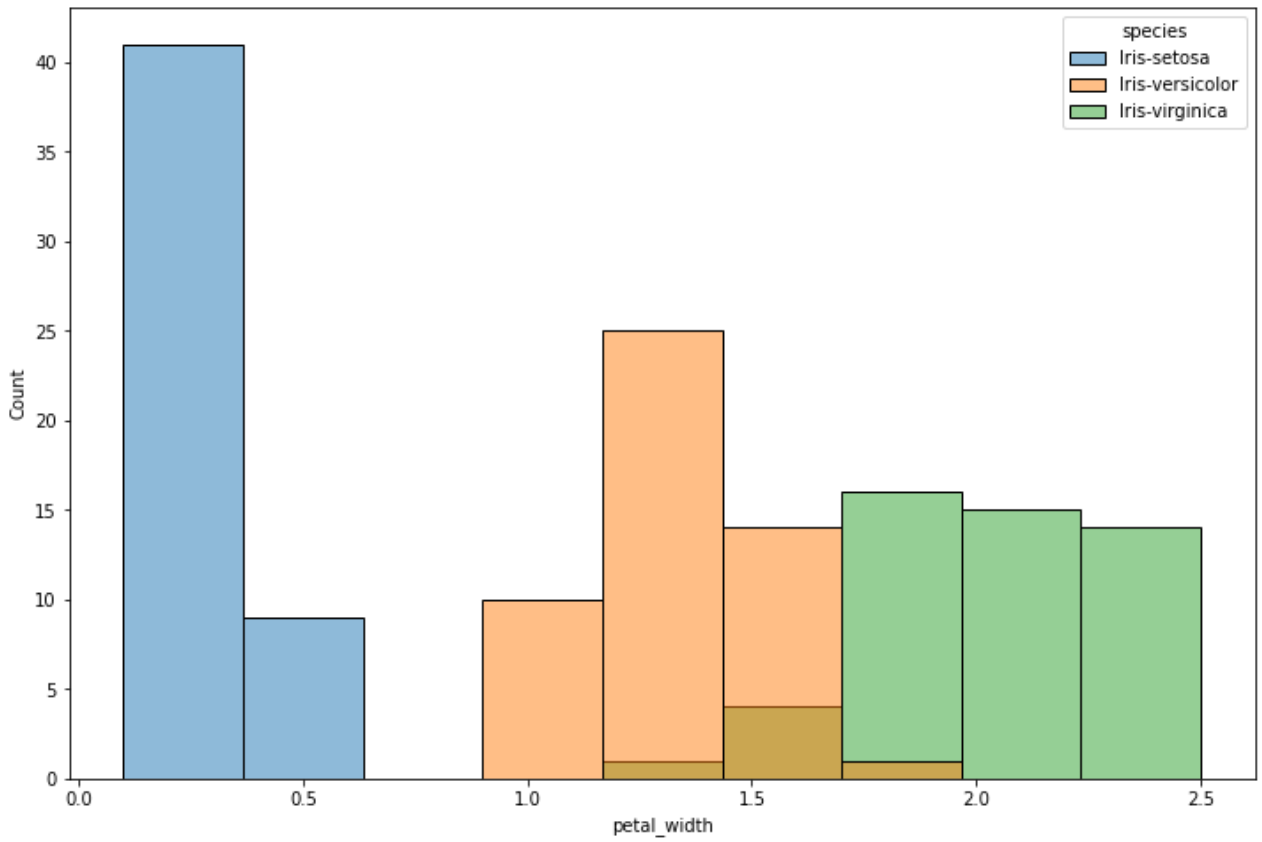
```
plt.figure(figsize=(12,8))
sns.histplot(x=data['sepal_width'], hue=data['species'])
plt.plot()
```

Out[26]: []



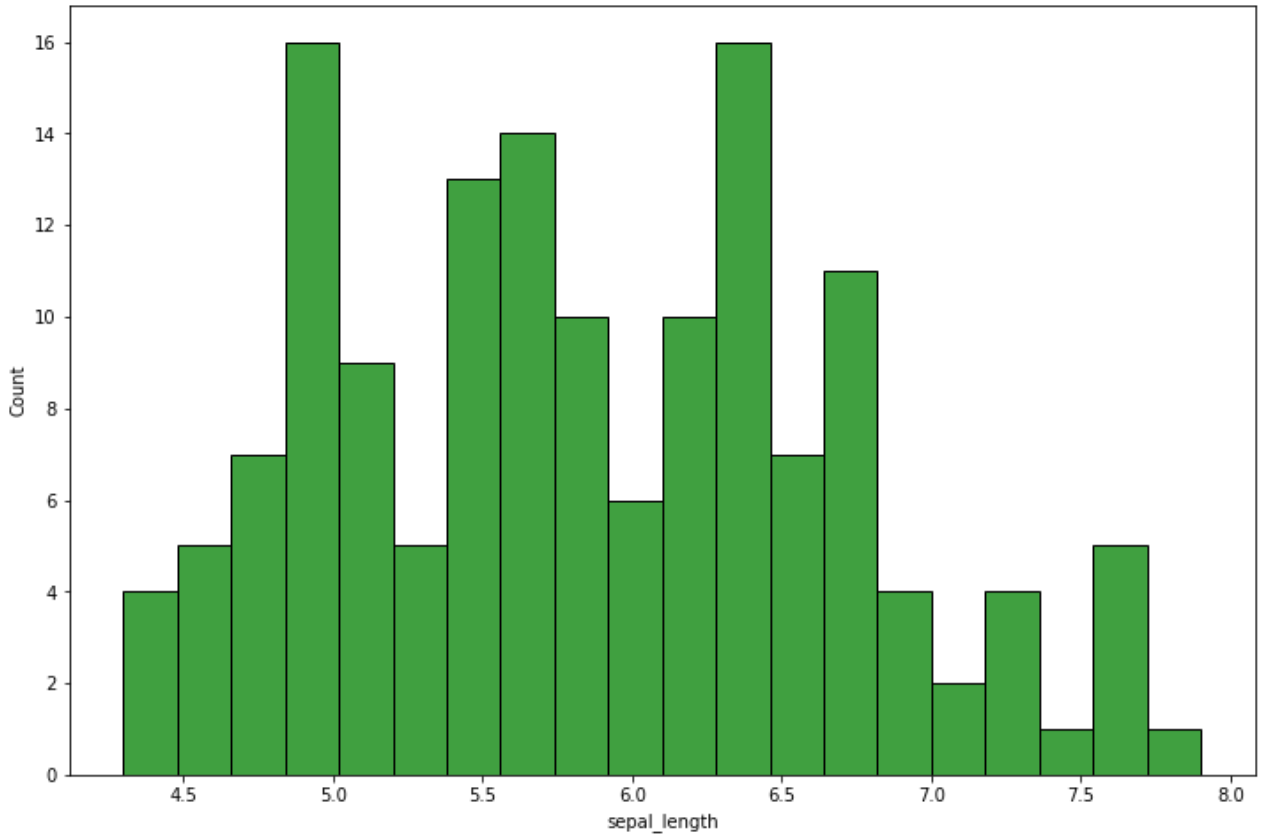
In [27]:
```
plt.figure(figsize=(12,8))
sns.histplot(x=data['petal_width'], hue=data['species'])
plt.plot()
```

Out[27]: []



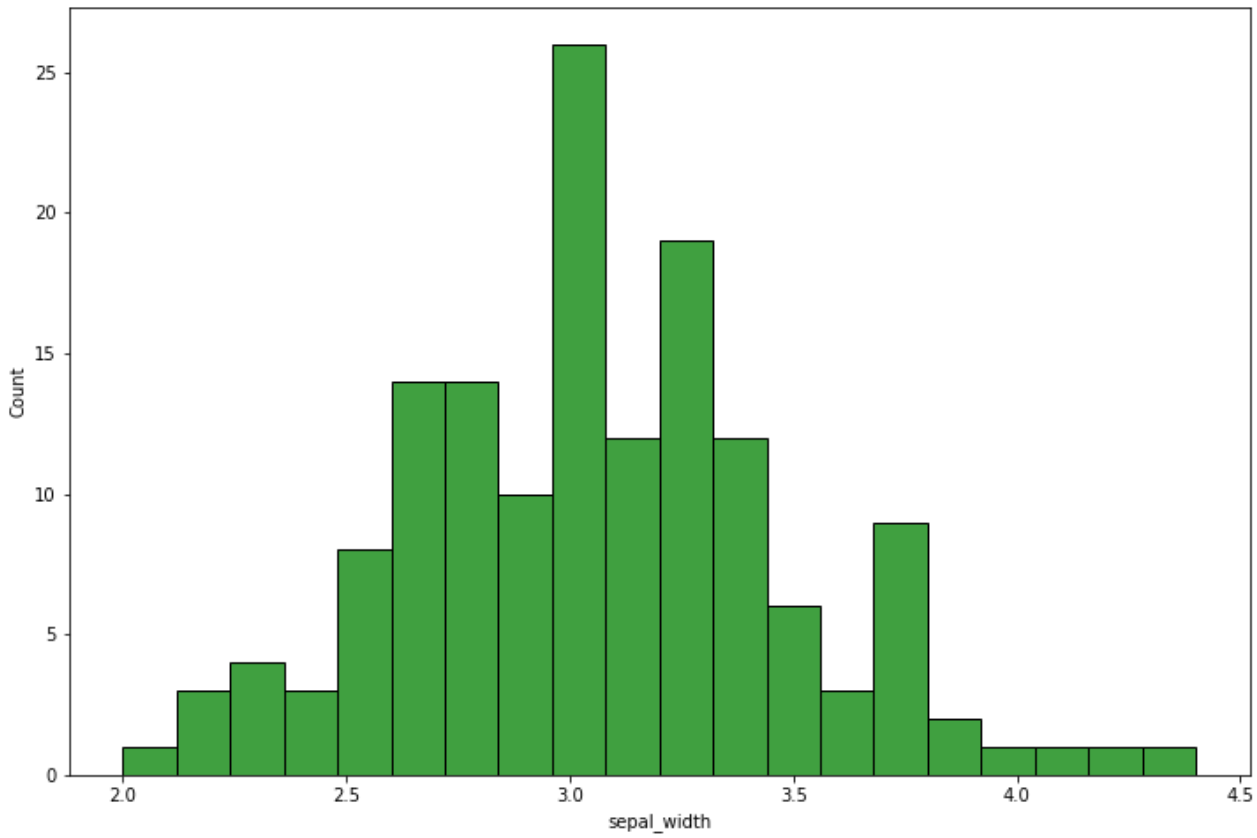
In [28]:
```
plt.figure(figsize=(12,8))
sns.histplot(x=data['sepal_length'],bins=20, color='green')
plt.plot()
```

Out[28]: []



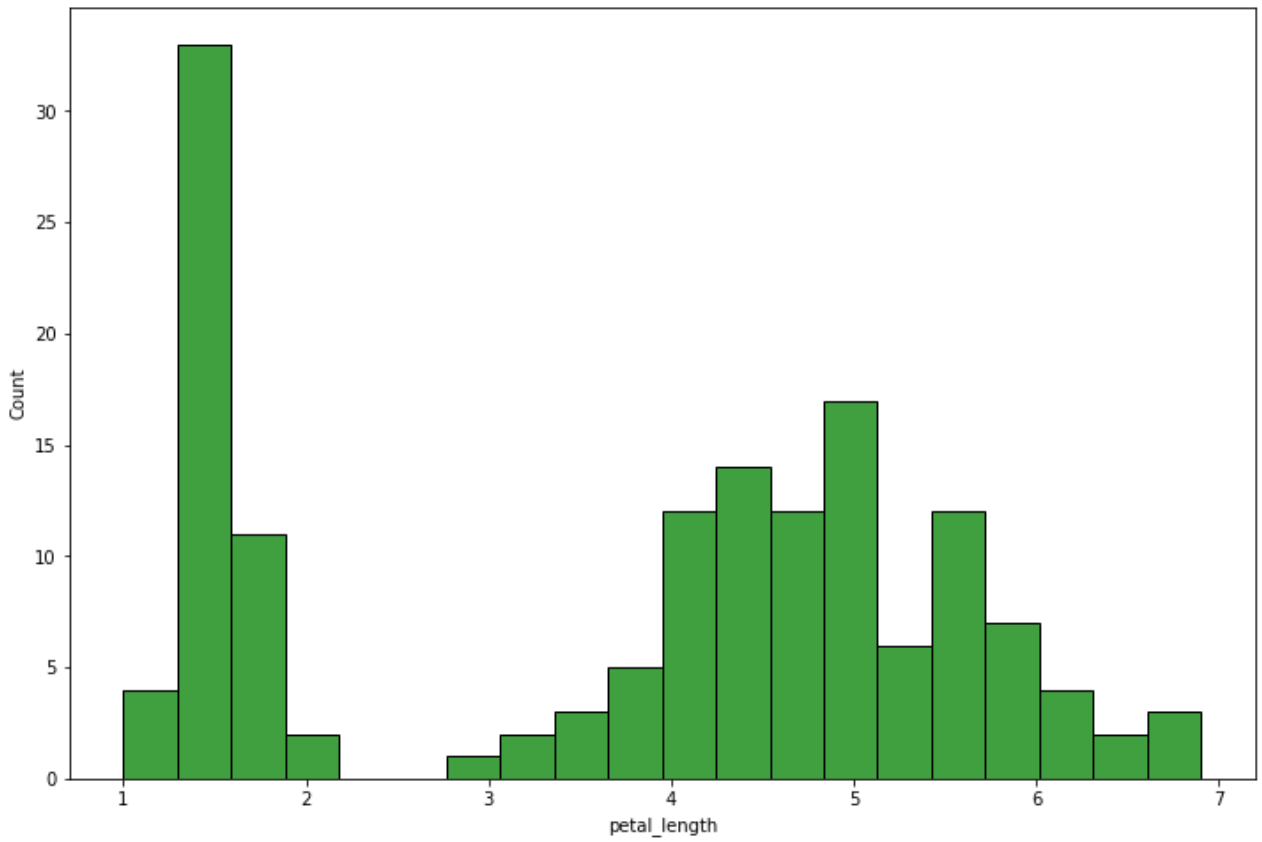
In [29]:
```
plt.figure(figsize=(12,8))
sns.histplot(x=data['sepal_width'],bins=20, color='green')
plt.plot()
```

Out[29]:
```
[]
```



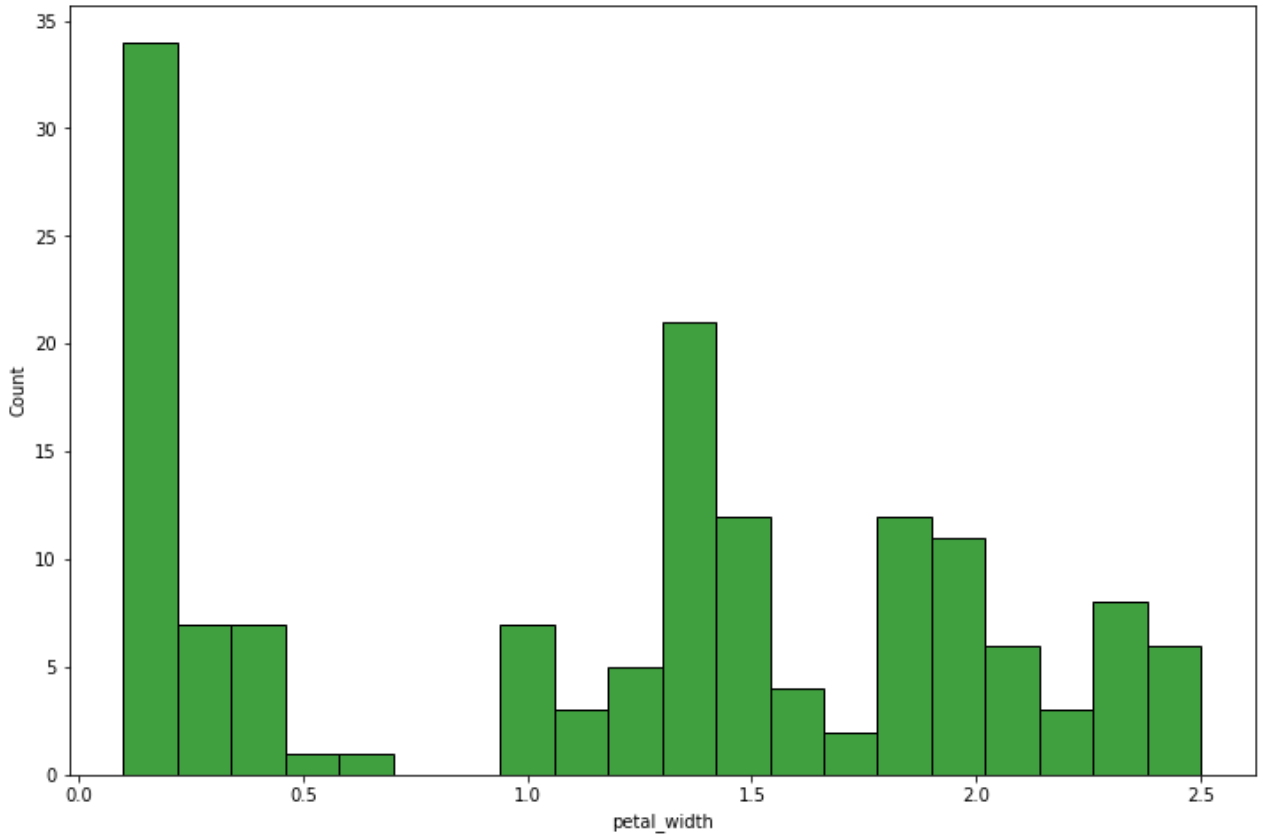
In [30]:
```
plt.figure(figsize=(12,8))
sns.histplot(x=data['petal_length'],bins=20, color='green')
plt.plot()
```

Out[30]: []



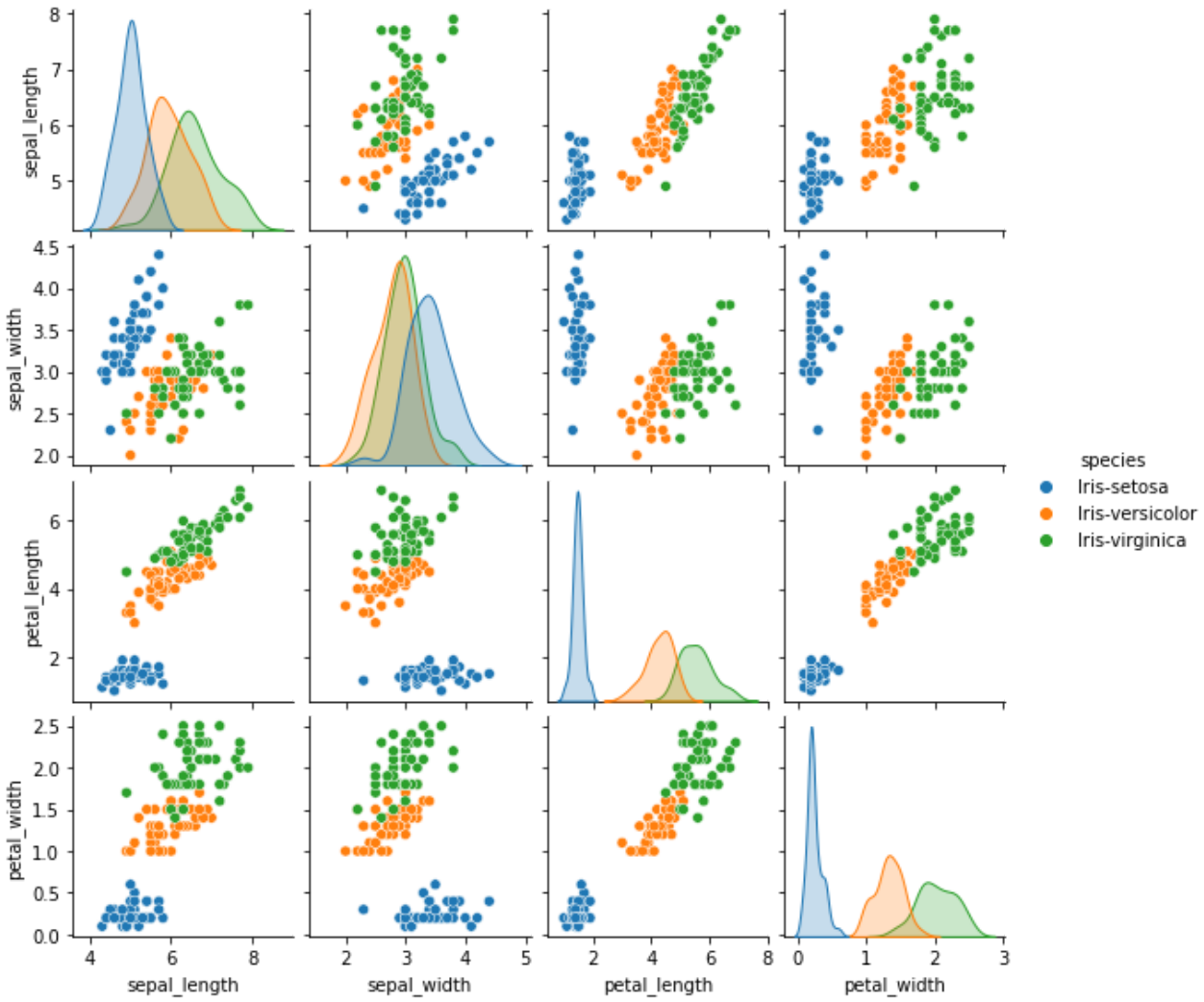
In [31]:
```
plt.figure(figsize=(12,8))
sns.histplot(x=data['petal_width'],bins=20, color='green')
plt.plot()
```

Out[31]: []



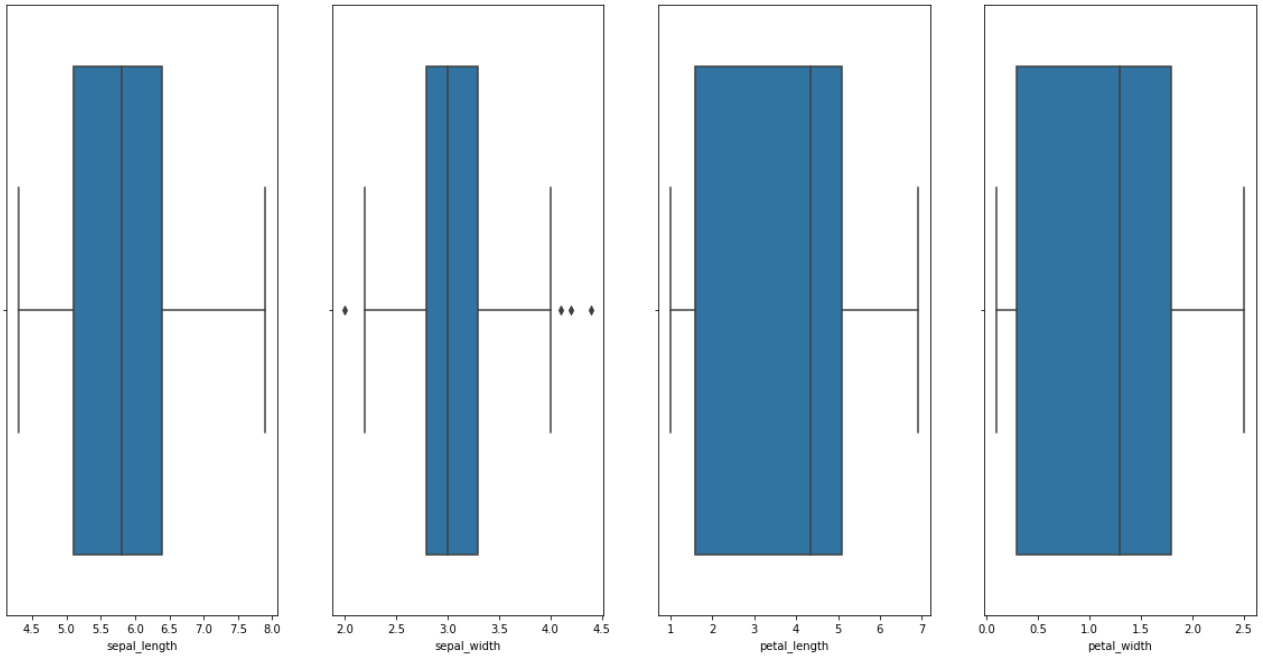
In [32]:
```
sns.pairplot(data, hue='species', height=2)
```

Out[32]:
```
<seaborn.axisgrid.PairGrid at 0x221433cf1c0>
```



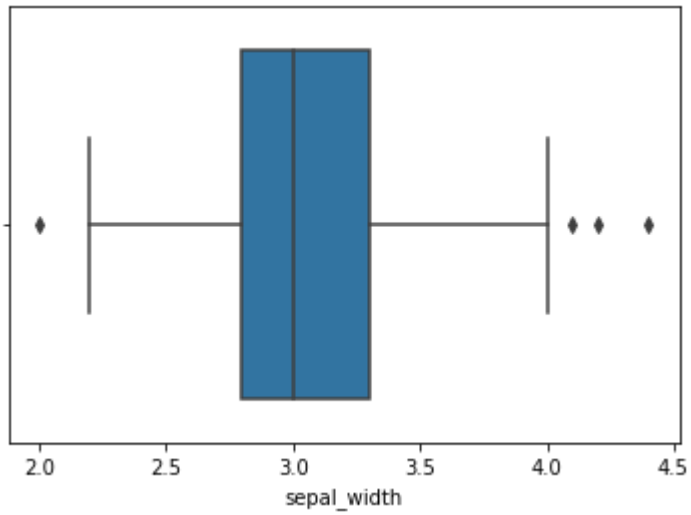
In [33]:
```
import warnings
```

In [34]:
```
warnings.filterwarnings('ignore')
features_=data.columns.values[:-1]
fig=plt.figure(figsize=(20,10))
for columns, feature in enumerate(features_):
    fig.add_subplot(1,4,columns+1)
    sns.boxplot(data[feature],data=data)
plt.show()
```



In [38]:
```
sns.boxplot(x='sepal_width', data=data)
```

Out[38]:
```
<AxesSubplot:xlabel='sepal_width'>
```



In the above graph, the values above 4 and below 2 are acting as outliers.

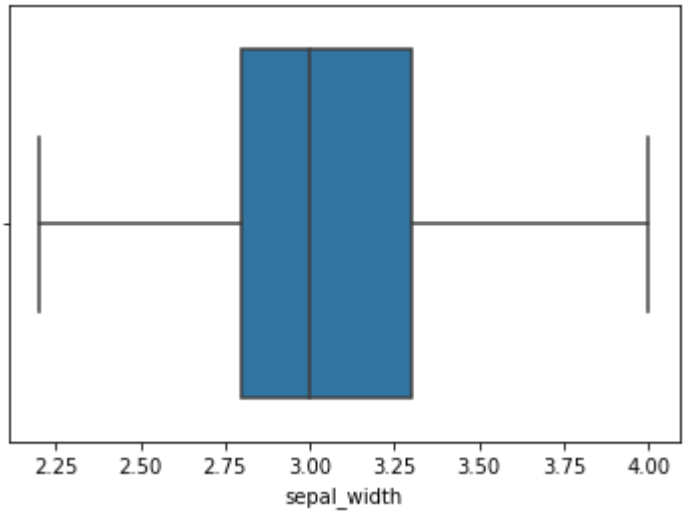
In [44]:
```
# Removing Outliers
# IQR
Q1 = np.percentile(data['sepal_width'], 25,interpolation = 'midpoint')

Q3 = np.percentile(data['sepal_width'], 75,interpolation = 'midpoint')
IQR = Q3 - Q1

print("Old Shape: ", data.shape)

# Upper bound
upper = np.where(data['sepal_width'] >= (Q3+1.5*IQR))

# Lower bound
lower = np.where(data['sepal_width'] <= (Q1-1.5*IQR))
```

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

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

sns.boxplot(x='sepal_width', data=data)
```

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

Out[44]:

```
<AxesSubplot:xlabel='sepal_width'>
```

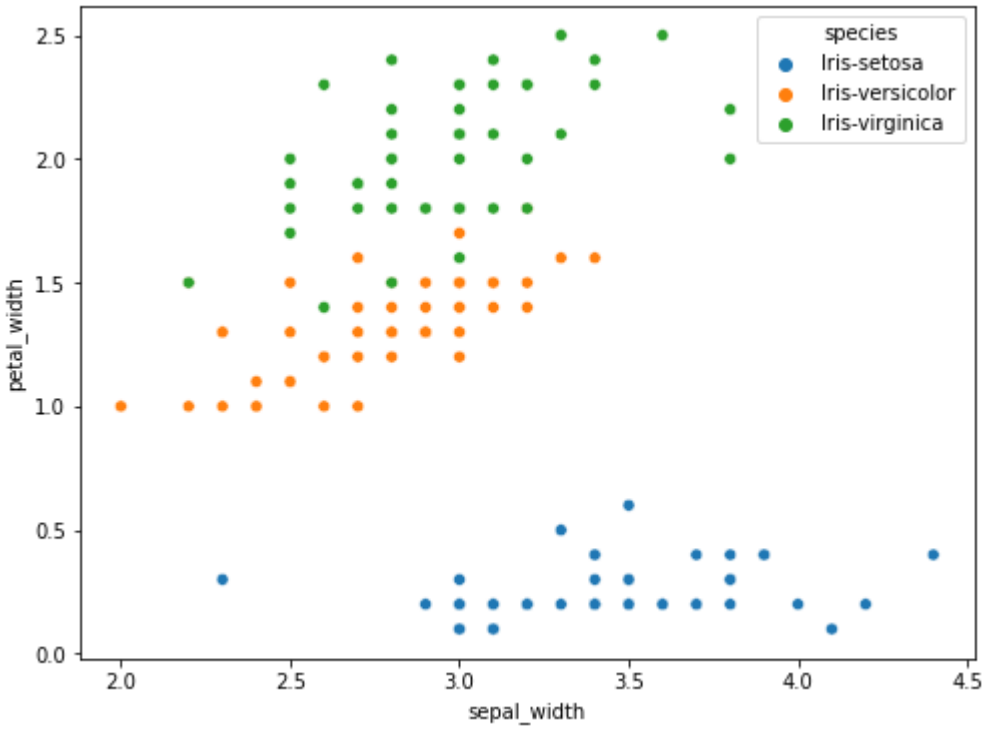


In [35]:

```
plt.figure(figsize=(8,6))
sns.scatterplot(data=data,x='sepal_width', y='petal_width', hue='species')
plt.plot()
```

Out[35]:

```
[]
```



In [36]:
```
plt.figure(figsize=(8,6))
sns.scatterplot(data=data,x='sepal_length', y='petal_length', hue='species')
plt.plot()
```

Out[36]: []