Nombre: León Emiliano García Pérez

Matrícula: A00573074

Repeating the Tutorial with de Iris Dataset

In [76]:
```
# Define where you are running the code: colab or local
RunInColab          = False     # (False: no  | True: yes)

# If running in colab:
if RunInColab:
    # Mount your google drive in google colab
    from google.colab import drive
    drive.mount('/content/drive')

    # Find location
    #!pwd
    #!ls
    #!ls "/content/drive/My Drive/Colab Notebooks/MachineLearningWithPython/"

    # Define path del proyecto
    Ruta            = ""

else:
    # Define path del proyecto
    Ruta            = 'datasets'
```

In [77]:
```
# Import the packages that we will be using

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

# Dataset url

url = '/iris/iris.csv'

# Load the dataset

df = pd.read_csv(Ruta+url)
df.columns = ["PetalWidth", "PetalLength", "SepalWidth", "SepalLength", "Type"]
```

In [78]:
```
df.head()
```

Out[78]:

	PetalWidth	PetalLength	SepalWidth	SepalLength	Type
0	4.9	3.0	1.4	0.2	Iris-setosa
1	4.7	3.2	1.3	0.2	Iris-setosa
2	4.6	3.1	1.5	0.2	Iris-setosa
3	5.0	3.6	1.4	0.2	Iris-setosa
4	5.4	3.9	1.7	0.4	Iris-setosa

In [79]:
```
df.shape
```

Out[79]:
```
(149, 5)
```

In [80]:
```
df.describe()
```

Out[80]:

	PetalWidth	PetalLength	SepalWidth	SepalLength
count	149.000000	149.000000	149.000000	149.000000
mean	5.848322	3.054362	3.773826	1.206040
std	0.828594	0.435810	1.760543	0.760354
min	4.300000	2.000000	1.000000	0.100000
25%	5.100000	2.800000	1.600000	0.300000
50%	5.800000	3.000000	4.400000	1.300000
75%	6.400000	3.300000	5.100000	1.800000
max	7.900000	4.400000	6.900000	2.500000

In [81]:
```
vars = ["PetalWidth","PetalLength","SepalWidth","SepalLength"]
df2 = df[vars].dropna()
```

In [82]:
```
count = df['PetalWidth'].value_counts()
print(count)
count = df['PetalLength'].value_counts()
print(count)
count = df['SepalWidth'].value_counts()
print(count)
count = df['SepalLength'].value_counts()
print(count)
count = df['Type'].value_counts()
print(count)
```

```
5.0    10
6.3     9
6.7     8
5.1     8
5.7     8
5.8     7
5.5     7
6.4     7
5.6     6
5.4     6
6.1     6
6.0     6
4.9     6
6.5     5
4.8     5
6.2     4
7.7     4
6.9     4
5.2     4
4.6     4
4.4     3
5.9     3
7.2     3
6.8     3
4.7     2
6.6     2
4.3     1
7.0     1
4.5     1
7.1     1
7.6     1
7.3     1
5.3     1
7.4     1
7.9     1
Name: PetalWidth, dtype: int64
3.0    26
2.8    14
3.2    13
3.4    12
3.1    11
2.9    10
2.7     9
2.5     8
3.3     6
3.8     6
3.5     5
2.6     5
2.3     4
3.6     4
3.7     3
2.4     3
2.2     3
3.9     2
4.4     1
4.0     1
4.1     1
4.2     1
2.0     1
Name: PetalLength, dtype: int64
1.5    13
```

```
1.4    12
5.1     8
4.5     8
1.6     7
1.3     7
5.6     6
4.7     5
4.9     5
4.0     5
4.2     4
5.0     4
4.4     4
4.8     4
1.7     4
3.9     3
4.6     3
5.7     3
4.1     3
5.5     3
6.1     3
5.8     3
3.3     2
5.4     2
6.7     2
5.3     2
5.9     2
6.0     2
1.2     2
4.3     2
1.9     2
3.5     2
5.2     2
3.0     1
1.1     1
3.7     1
3.8     1
6.6     1
6.3     1
1.0     1
6.9     1
3.6     1
6.4     1
Name: SepalWidth, dtype: int64
0.2    28
1.3    13
1.8    12
1.5    12
1.4     8
2.3     8
1.0     7
0.4     7
0.3     7
2.1     6
2.0     6
0.1     5
1.2     5
1.9     5
1.6     4
2.5     3
2.2     3
2.4     3
```

```
1.1     3
1.7     2
0.6     1
0.5     1
Name: SepalLength, dtype: int64
Iris-versicolor    50
Iris-virginica     50
Iris-setosa        49
Name: Type, dtype: int64
```

In [83]:
```
proportion = df['PetalWidth'].value_counts(normalize=True)
print(proportion)
proportion = df['PetalLength'].value_counts(normalize=True)
print(proportion)
proportion = df['SepalWidth'].value_counts(normalize=True)
print(proportion)
proportion = df['SepalLength'].value_counts(normalize=True)
print(proportion)
proportion = df['Type'].value_counts(normalize=True)
print(proportion)
```

```
5.0    0.067114
6.3    0.060403
6.7    0.053691
5.1    0.053691
5.7    0.053691
5.8    0.046980
5.5    0.046980
6.4    0.046980
5.6    0.040268
5.4    0.040268
6.1    0.040268
6.0    0.040268
4.9    0.040268
6.5    0.033557
4.8    0.033557
6.2    0.026846
7.7    0.026846
6.9    0.026846
5.2    0.026846
4.6    0.026846
4.4    0.020134
5.9    0.020134
7.2    0.020134
6.8    0.020134
4.7    0.013423
6.6    0.013423
4.3    0.006711
7.0    0.006711
4.5    0.006711
7.1    0.006711
7.6    0.006711
7.3    0.006711
5.3    0.006711
7.4    0.006711
7.9    0.006711
Name: PetalWidth, dtype: float64
3.0    0.174497
2.8    0.093960
3.2    0.087248
3.4    0.080537
3.1    0.073826
2.9    0.067114
2.7    0.060403
2.5    0.053691
3.3    0.040268
3.8    0.040268
3.5    0.033557
2.6    0.033557
2.3    0.026846
3.6    0.026846
3.7    0.020134
2.4    0.020134
2.2    0.020134
3.9    0.013423
4.4    0.006711
4.0    0.006711
4.1    0.006711
4.2    0.006711
2.0    0.006711
Name: PetalLength, dtype: float64
1.5    0.087248
```

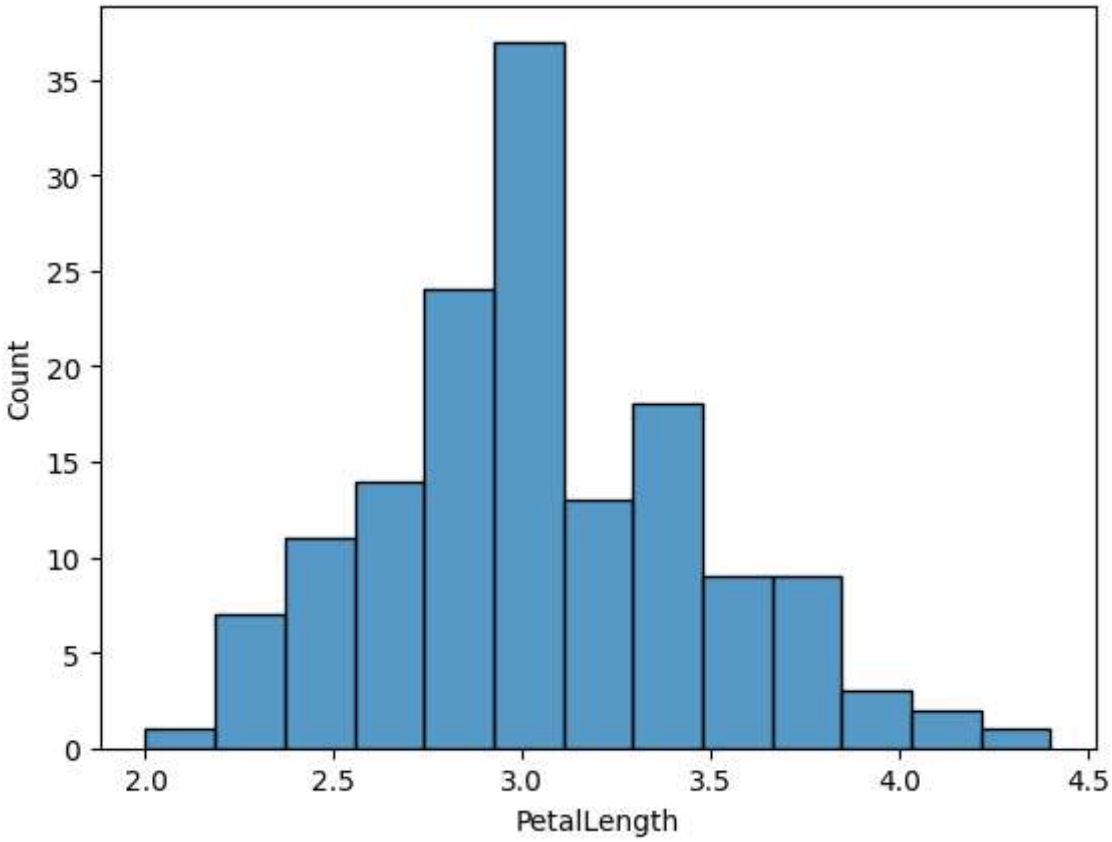
```
1.4    0.080537
5.1    0.053691
4.5    0.053691
1.6    0.046980
1.3    0.046980
5.6    0.040268
4.7    0.033557
4.9    0.033557
4.0    0.033557
4.2    0.026846
5.0    0.026846
4.4    0.026846
4.8    0.026846
1.7    0.026846
3.9    0.020134
4.6    0.020134
5.7    0.020134
4.1    0.020134
5.5    0.020134
6.1    0.020134
5.8    0.020134
3.3    0.013423
5.4    0.013423
6.7    0.013423
5.3    0.013423
5.9    0.013423
6.0    0.013423
1.2    0.013423
4.3    0.013423
1.9    0.013423
3.5    0.013423
5.2    0.013423
3.0    0.006711
1.1    0.006711
3.7    0.006711
3.8    0.006711
6.6    0.006711
6.3    0.006711
1.0    0.006711
6.9    0.006711
3.6    0.006711
6.4    0.006711
Name: SepalWidth, dtype: float64
0.2    0.187919
1.3    0.087248
1.8    0.080537
1.5    0.080537
1.4    0.053691
2.3    0.053691
1.0    0.046980
0.4    0.046980
0.3    0.046980
2.1    0.040268
2.0    0.040268
0.1    0.033557
1.2    0.033557
1.9    0.033557
1.6    0.026846
2.5    0.020134
2.2    0.020134
2.4    0.020134
```

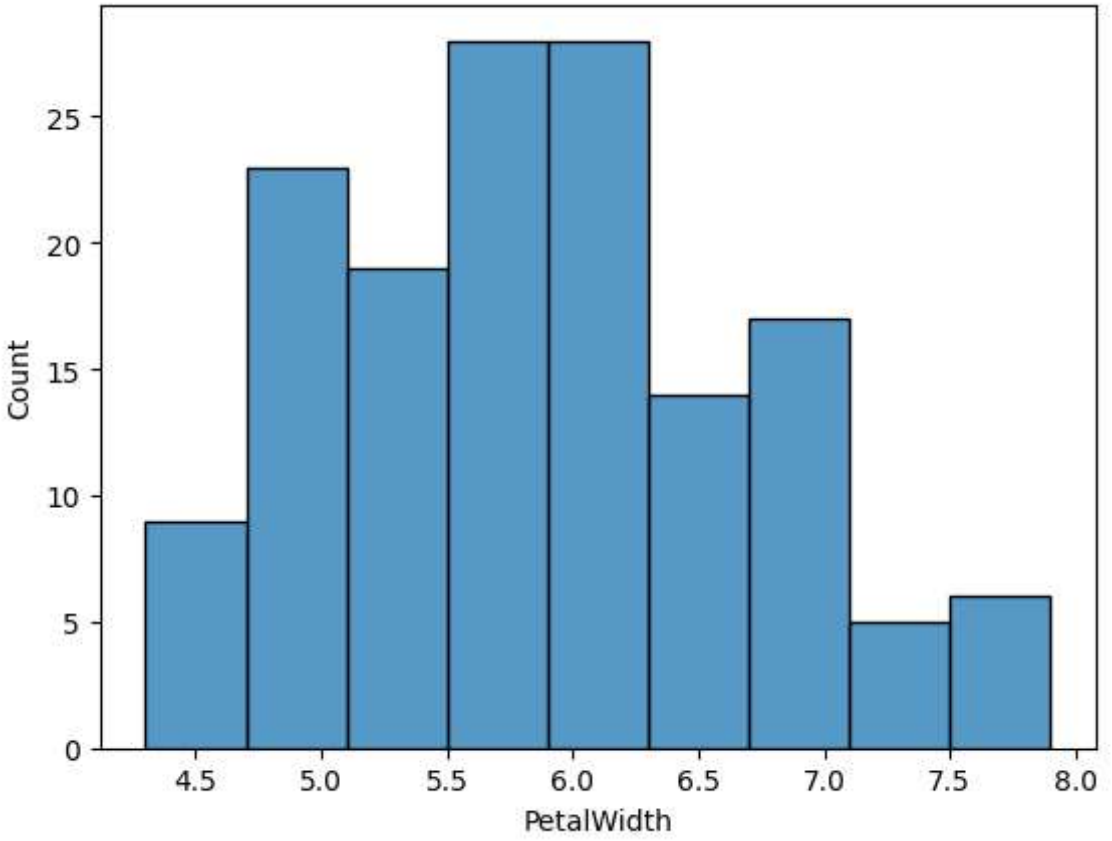
```
1.1    0.020134
1.7    0.013423
0.6    0.006711
0.5    0.006711
Name: SepalLength, dtype: float64
Iris-versicolor    0.335570
Iris-virginica     0.335570
Iris-setosa        0.328859
Name: Type, dtype: float64
```

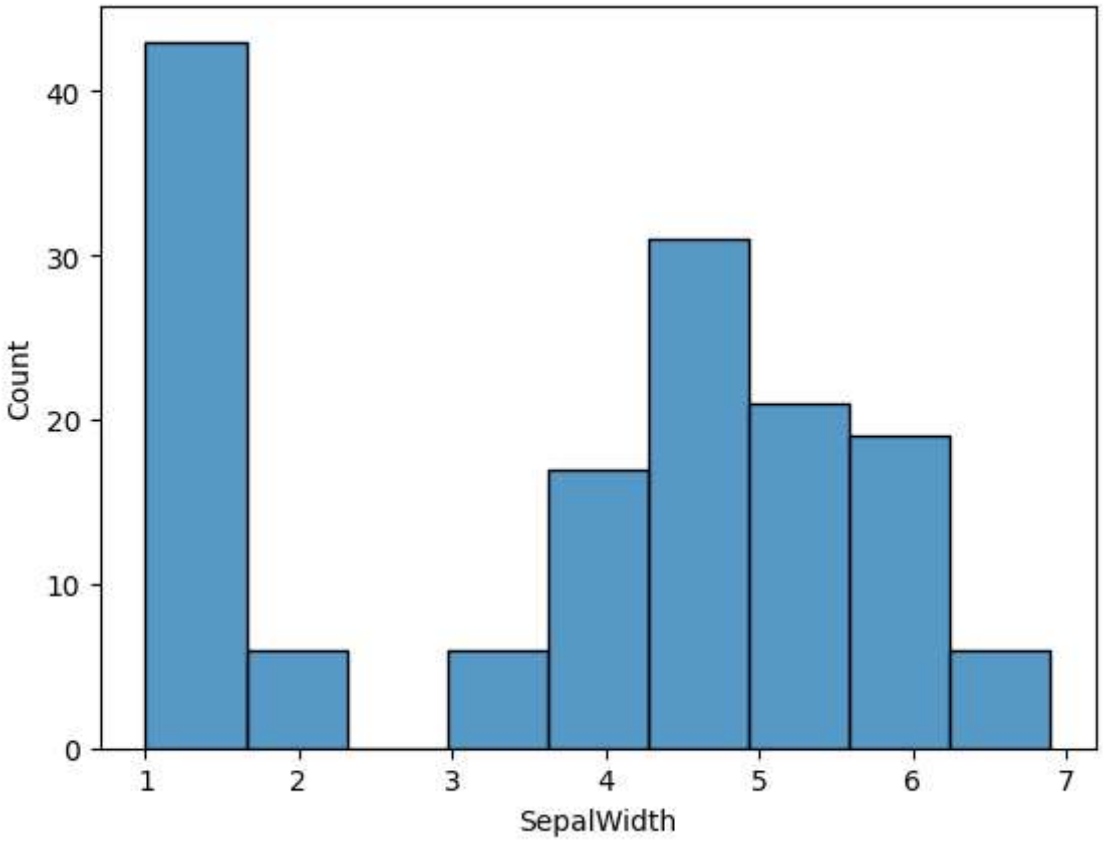
In [84]:
```
# Total number of observations

obs = df.count()
print("Number of observations: ", obs)

# Total number of null observations

obsNULL = df.isnull().sum()
print("Number of null observations: ", obsNULL)

# Total number of counts (excluding missing values)

counts = obs-obsNULL
print("Total number of counts: ", counts)
```

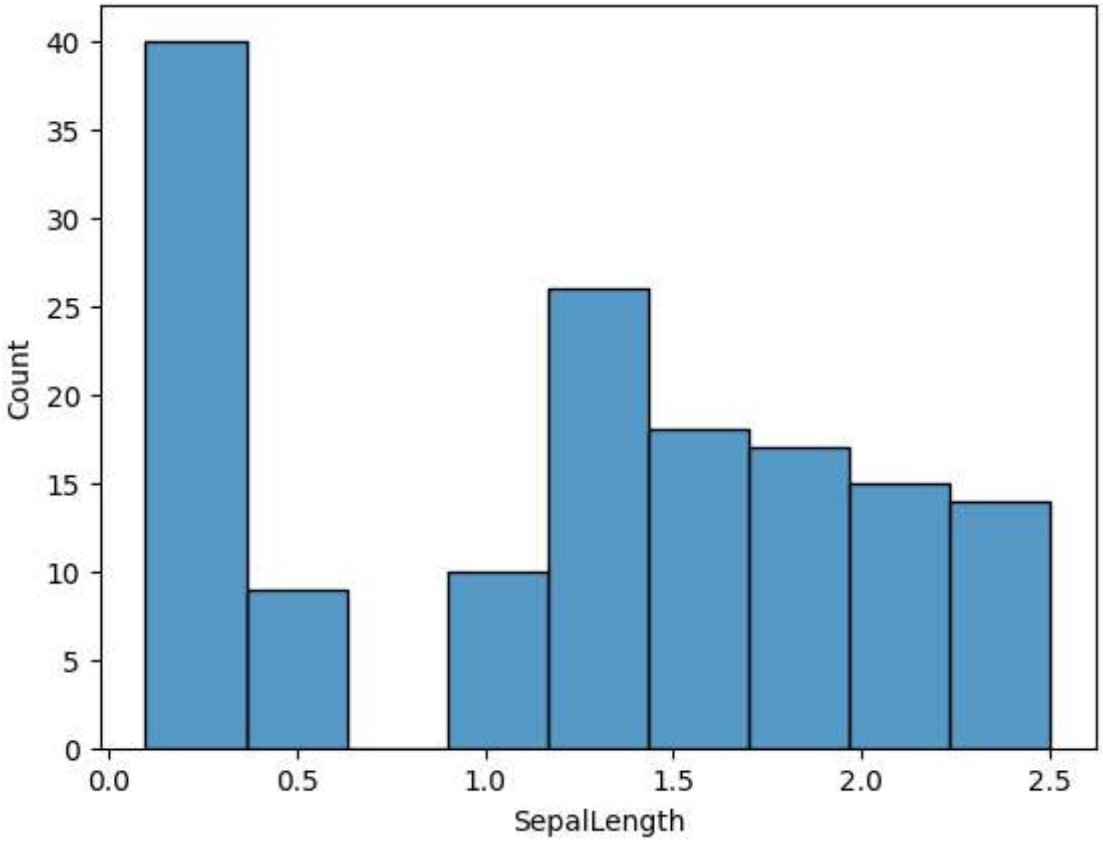
```
Number of observations:  PetalWidth     149
PetalLength    149
SepalWidth     149
SepalLength    149
Type           149
dtype: int64
Number of null observations:  PetalWidth     0
PetalLength    0
SepalWidth     0
SepalLength    0
Type           0
dtype: int64
Total number of counts:  PetalWidth     149
PetalLength    149
SepalWidth     149
SepalLength    149
Type           149
dtype: int64
```

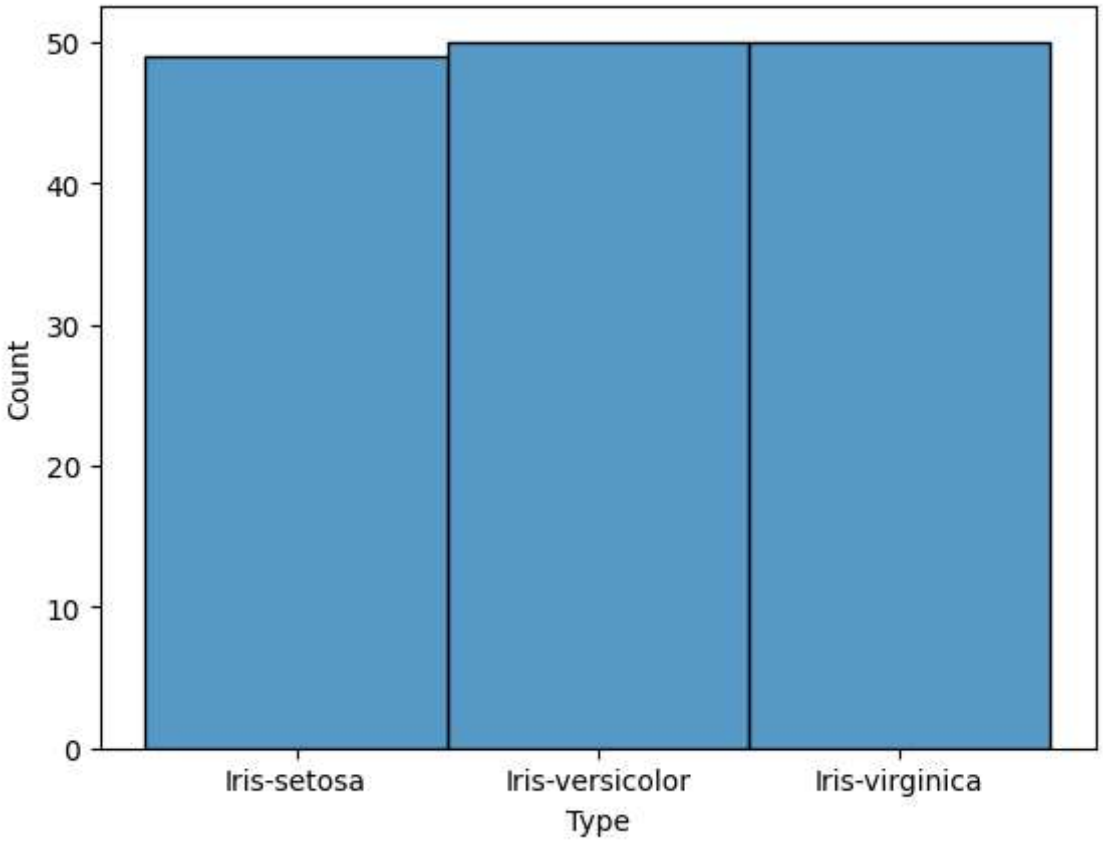
In [85]:
```
sns.histplot(data=df, x="PetalLength")
plt.show()

sns.histplot(data=df, x="PetalWidth")
plt.show()

sns.histplot(data=df, x="SepalWidth")
plt.show()

sns.histplot(data=df, x="SepalLength")
plt.show()

sns.histplot(data=df, x="Type")
plt.show()
```


Count
PetalLength
0
5
10
15
20
25
30
35
2.0
2.5
3.0
3.5
4.0
4.5

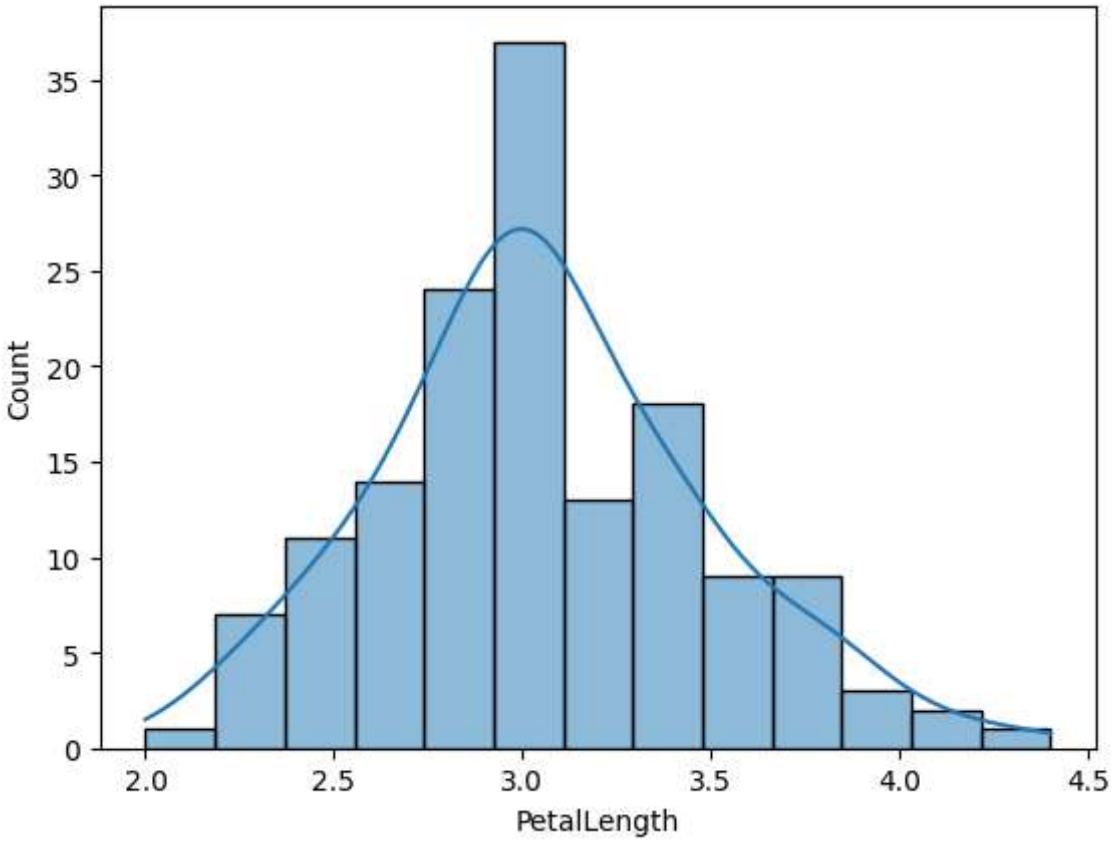

Count
PetalWidth
0
5
10
15
20
25
4.5
5.0
5.5
6.0
6.5
7.0
7.5
8.0

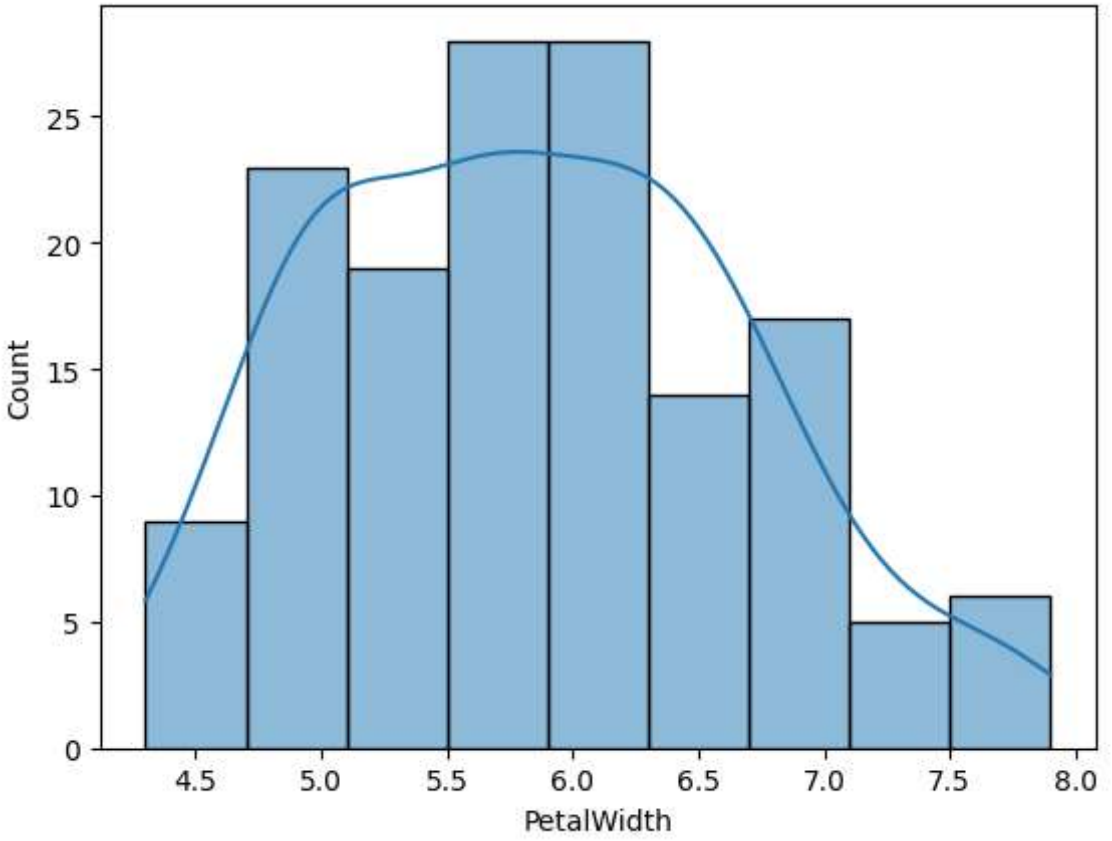

Count
SepalWidth
0
10
20
30
40
1
2
3
4
5
6
7

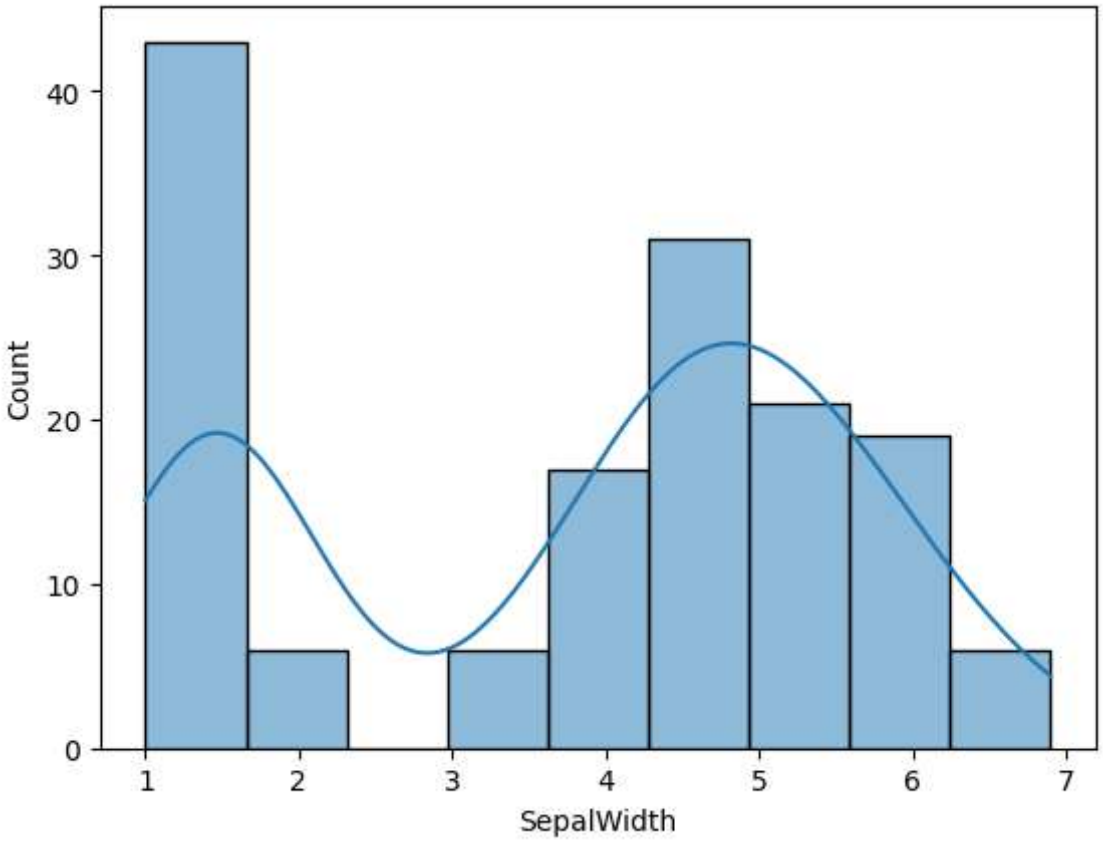

Count
SepalLength
0
5
10
15
20
25
30
35
40
0.0
0.5
1.0
1.5
2.0
2.5

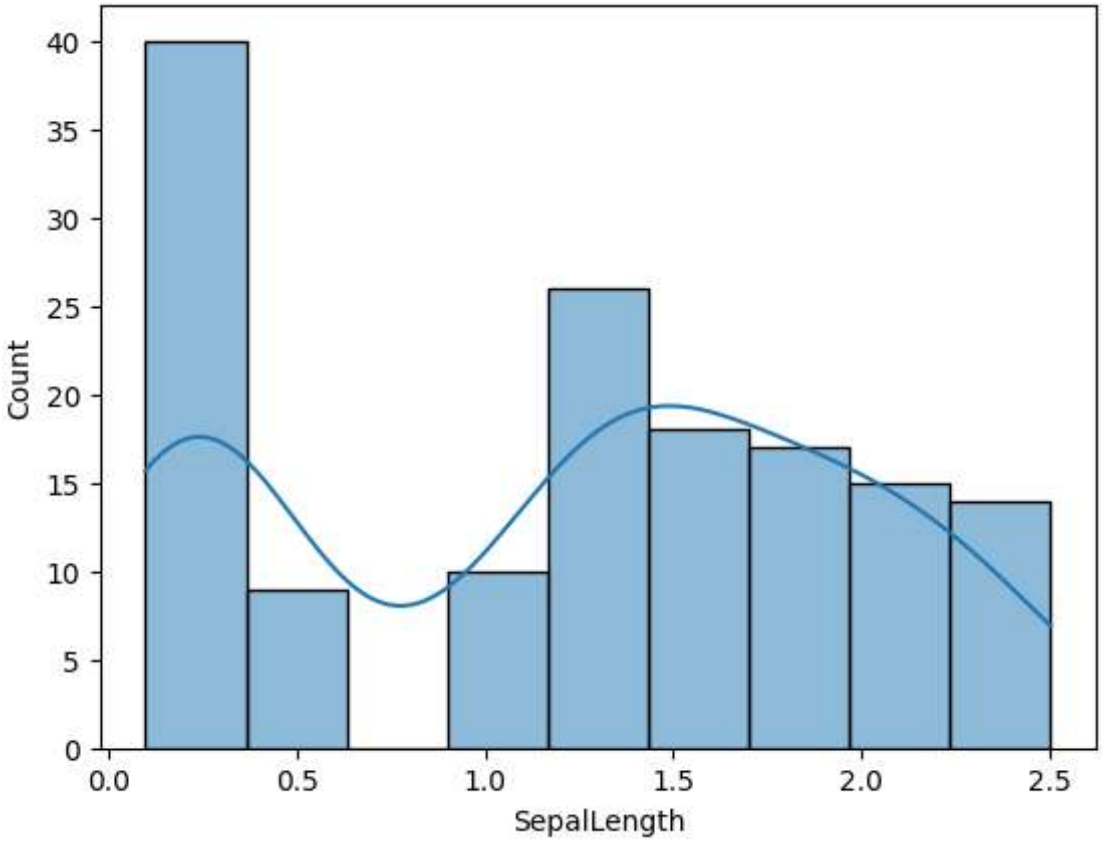


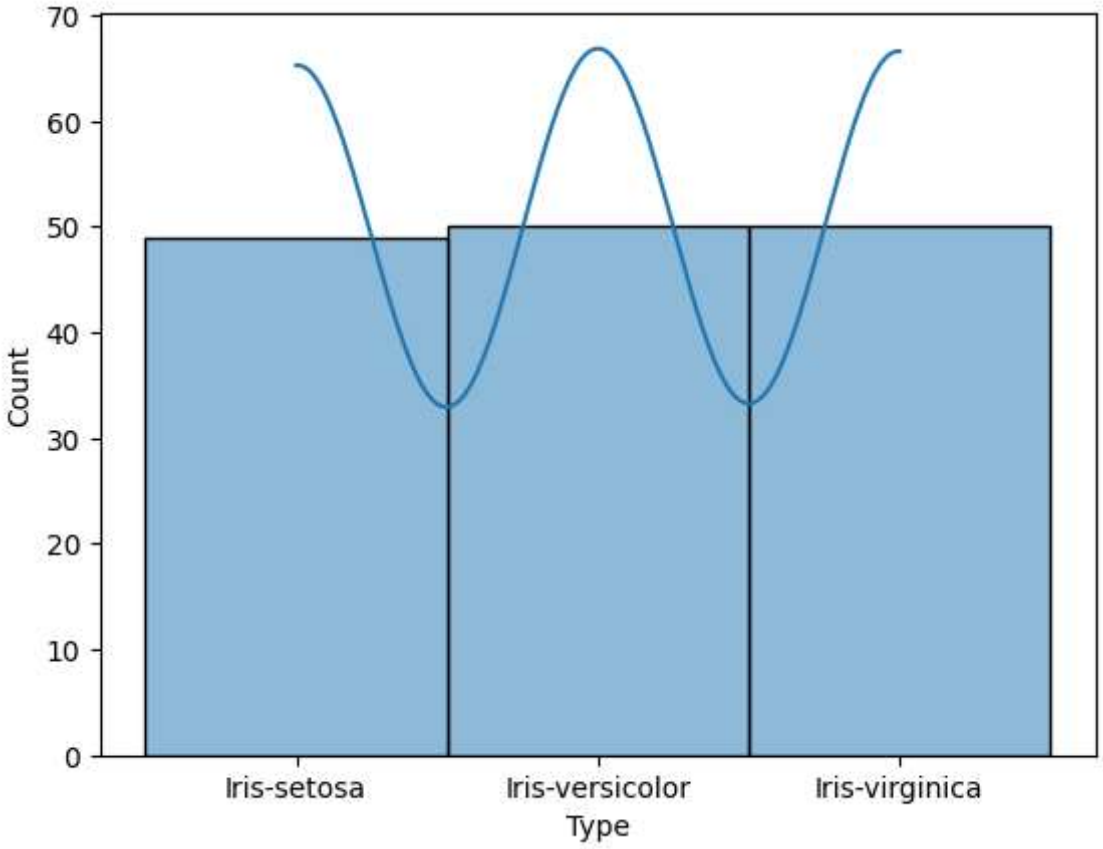
In [86]:
```
sns.histplot(data=df, x="PetalLength", kde=True)
plt.show()

sns.histplot(data=df, x="PetalWidth", kde=True)
plt.show()

sns.histplot(data=df, x="SepalWidth", kde=True)
plt.show()

sns.histplot(data=df, x="SepalLength", kde=True)
plt.show()

sns.histplot(data=df, x="Type", kde=True)
plt.show()
```


Count
PetalLength
0
5
10
15
20
25
30
35
2.0
2.5
3.0
3.5
4.0
4.5

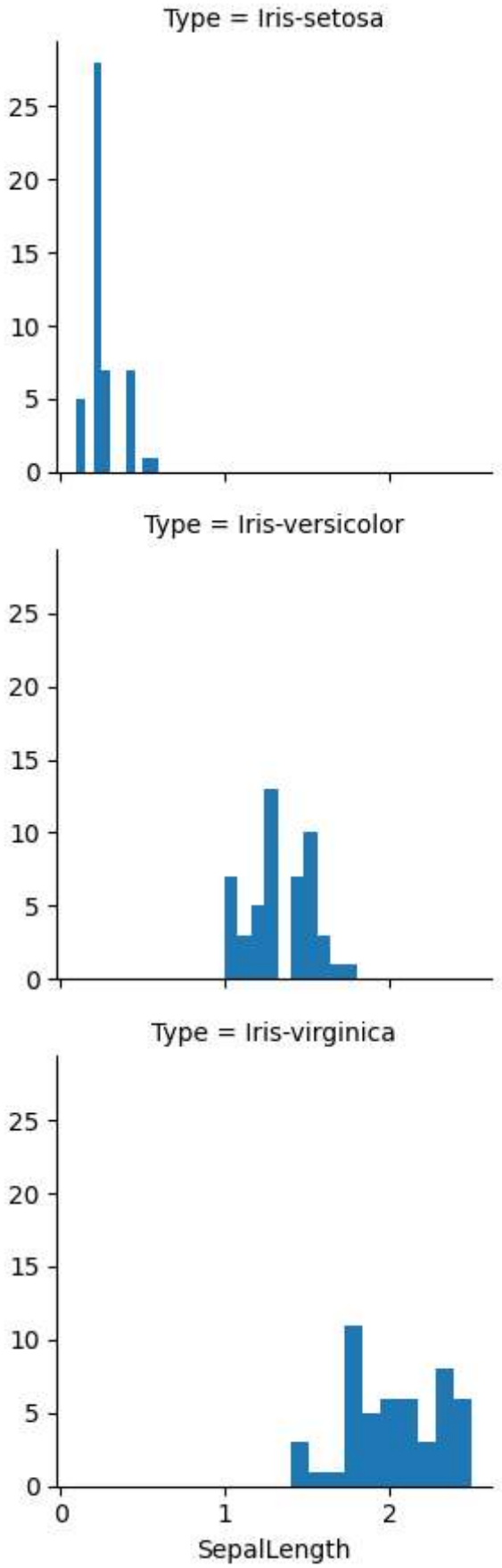

Count
PetalWidth
0
5
10
15
20
25
4.5
5.0
5.5
6.0
6.5
7.0
7.5
8.0


Count
SepalWidth
0
10
20
30
40
1
2
3
4
5
6
7

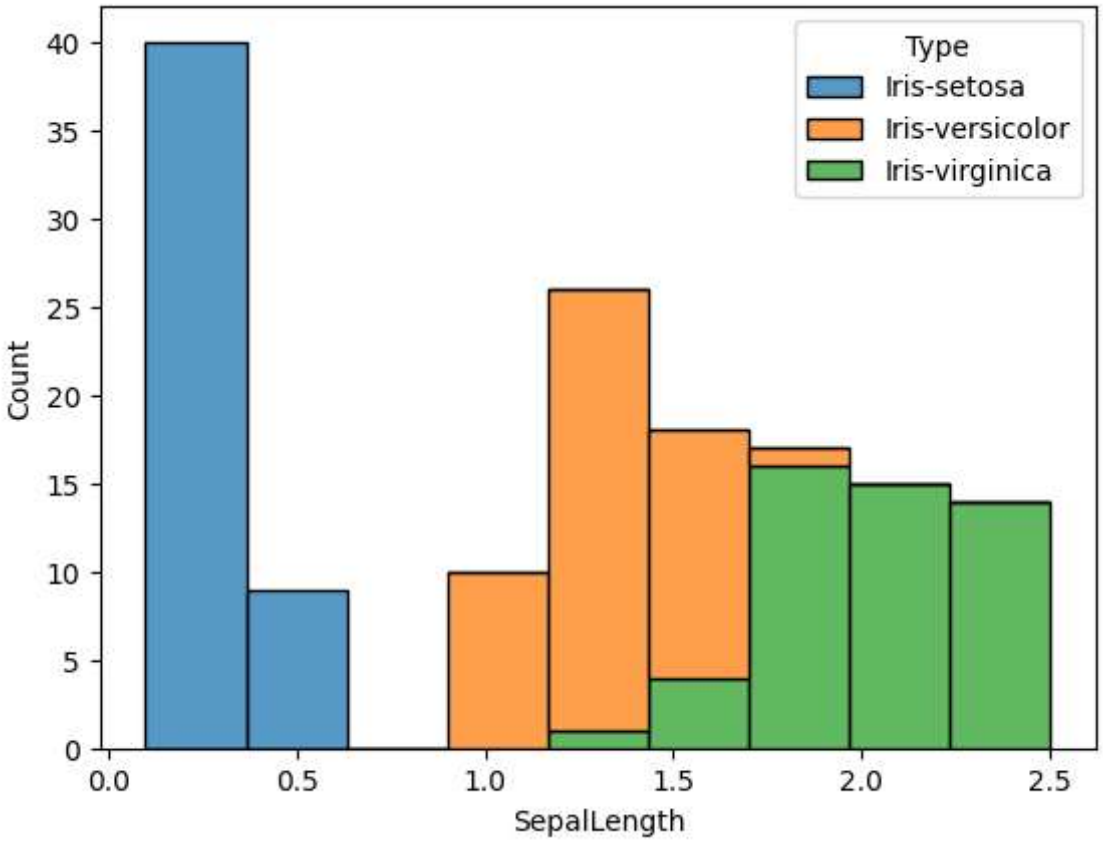

Count
SepalLength
0
5
10
15
20
25
30
35
40
0.0
0.5
1.0
1.5
2.0
2.5



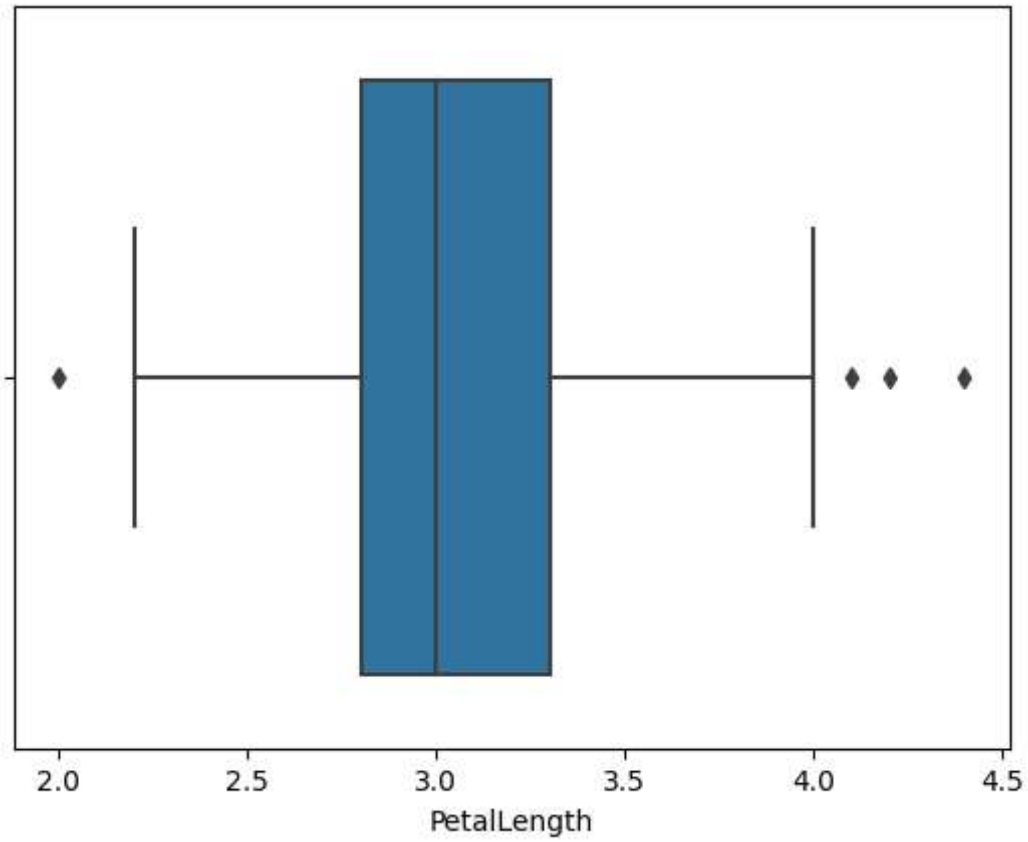
In [87]:
```
h = sns.FacetGrid(df, row = "Type")
h = h.map(plt.hist, "SepalLength")
plt.show()
```

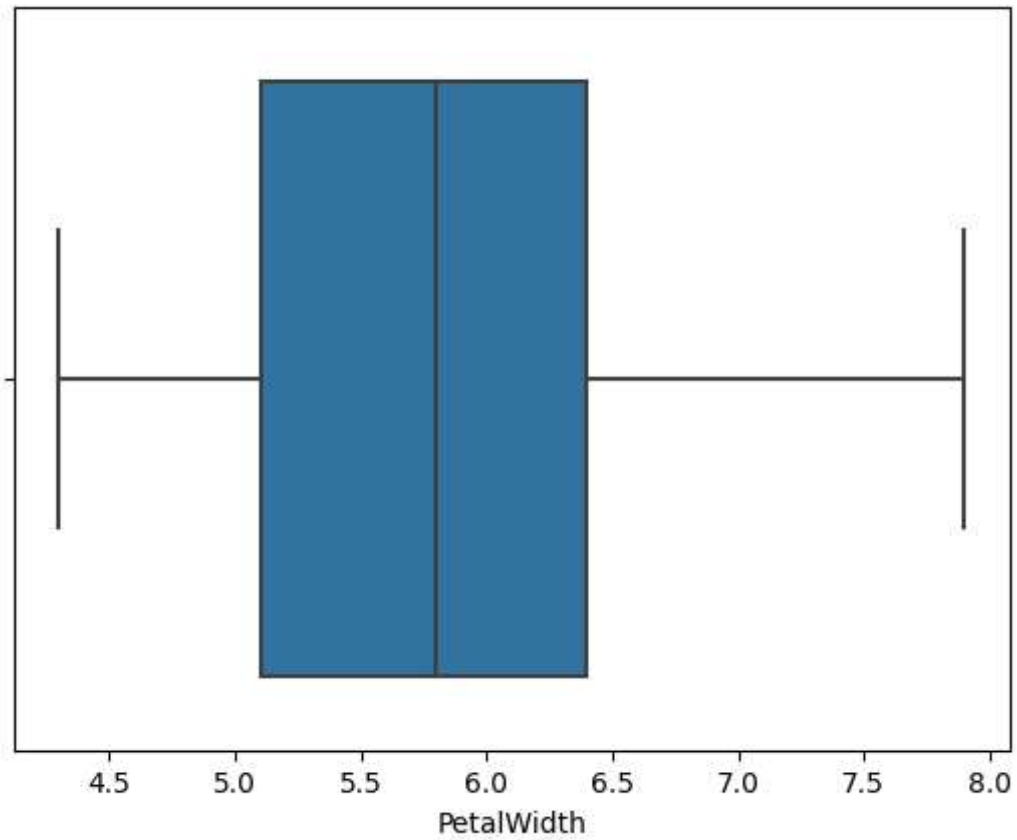


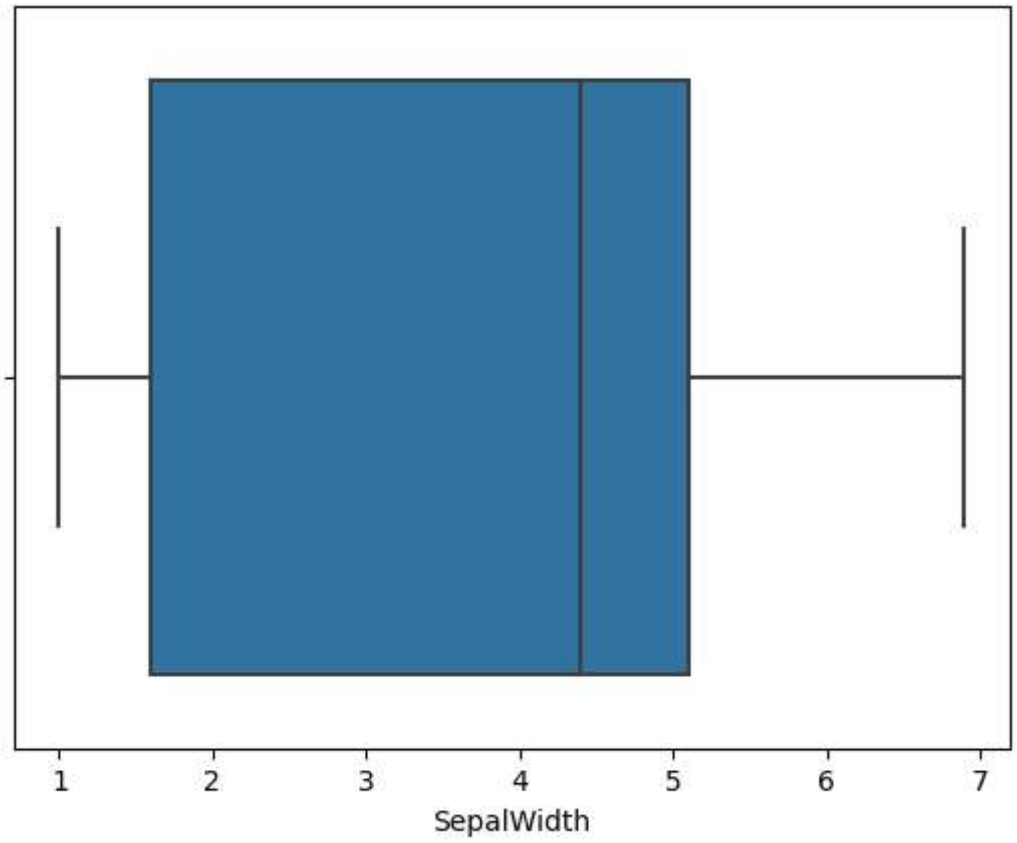
In [88]:
```
sns.histplot(data=df, x="SepalLength", hue="Type", multiple="stack")
plt.show()
```

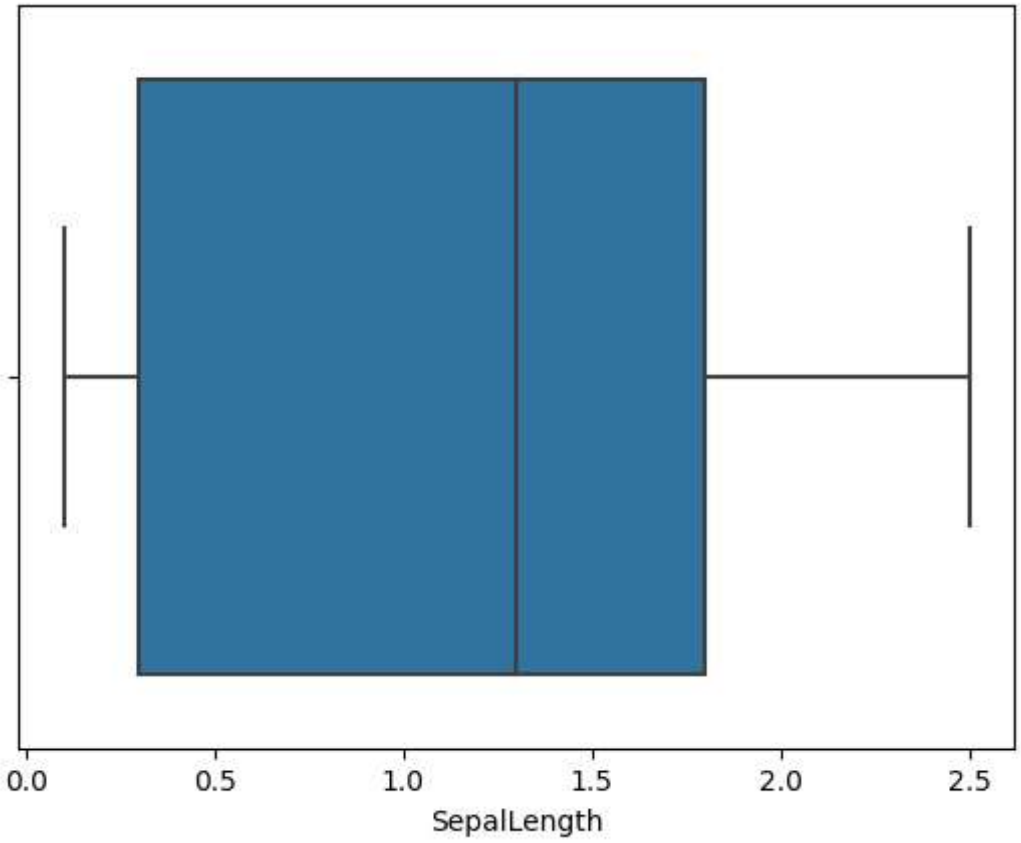


In [89]:
```
sns.boxplot(x=df["PetalLength"])
plt.show()

sns.boxplot(x=df["PetalWidth"])
plt.show()

sns.boxplot(x=df["SepalWidth"])
plt.show()

sns.boxplot(x=df["SepalLength"])
plt.show()
```


2.0
2.5
3.0
3.5
4.0
4.5
PetalLength

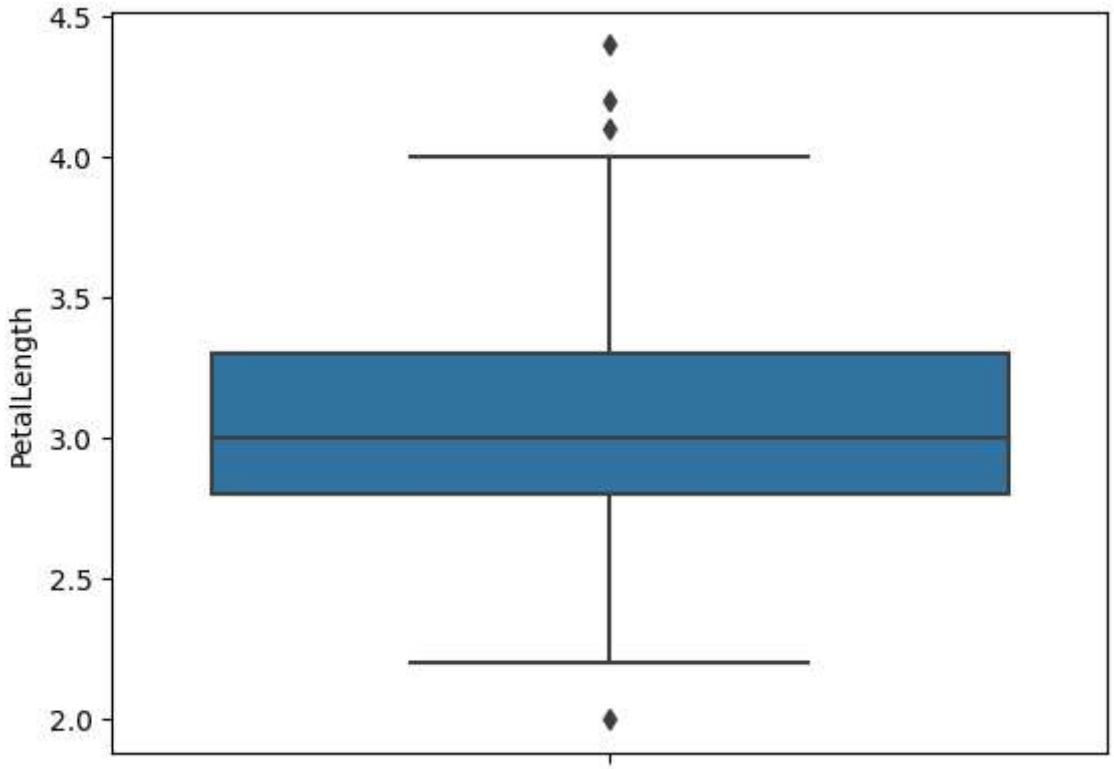

4.5
5.0
5.5
6.0
6.5
7.0
7.5
8.0
PetalWidth

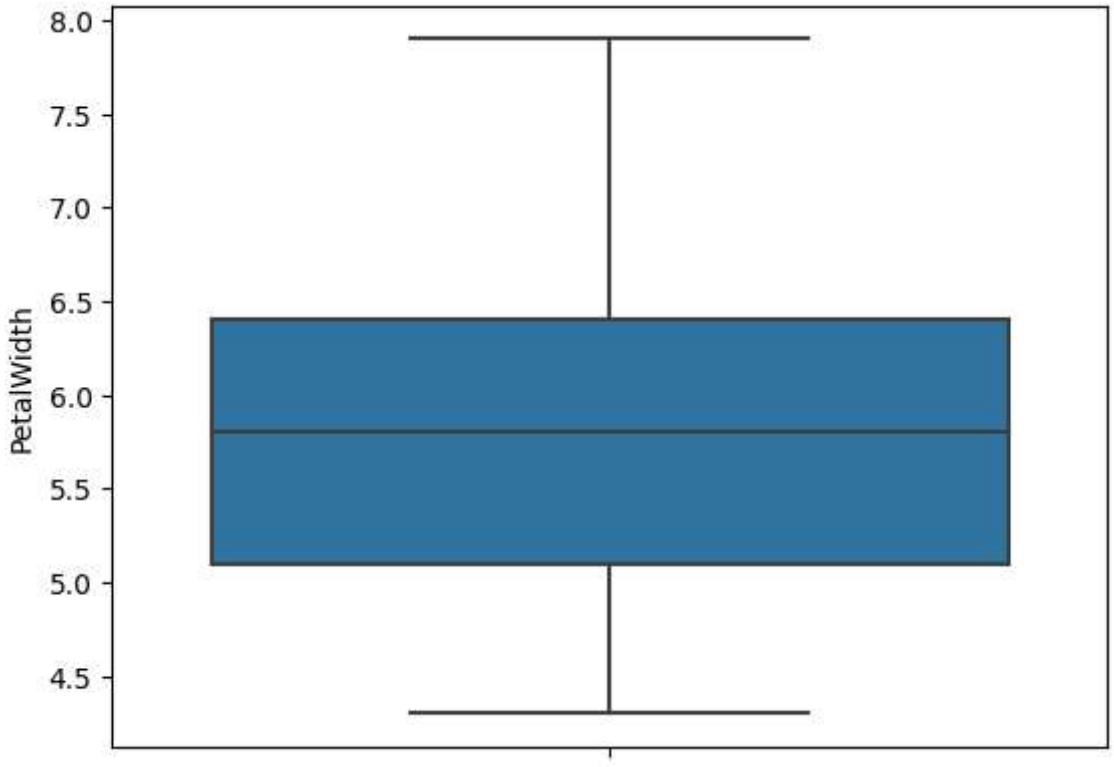


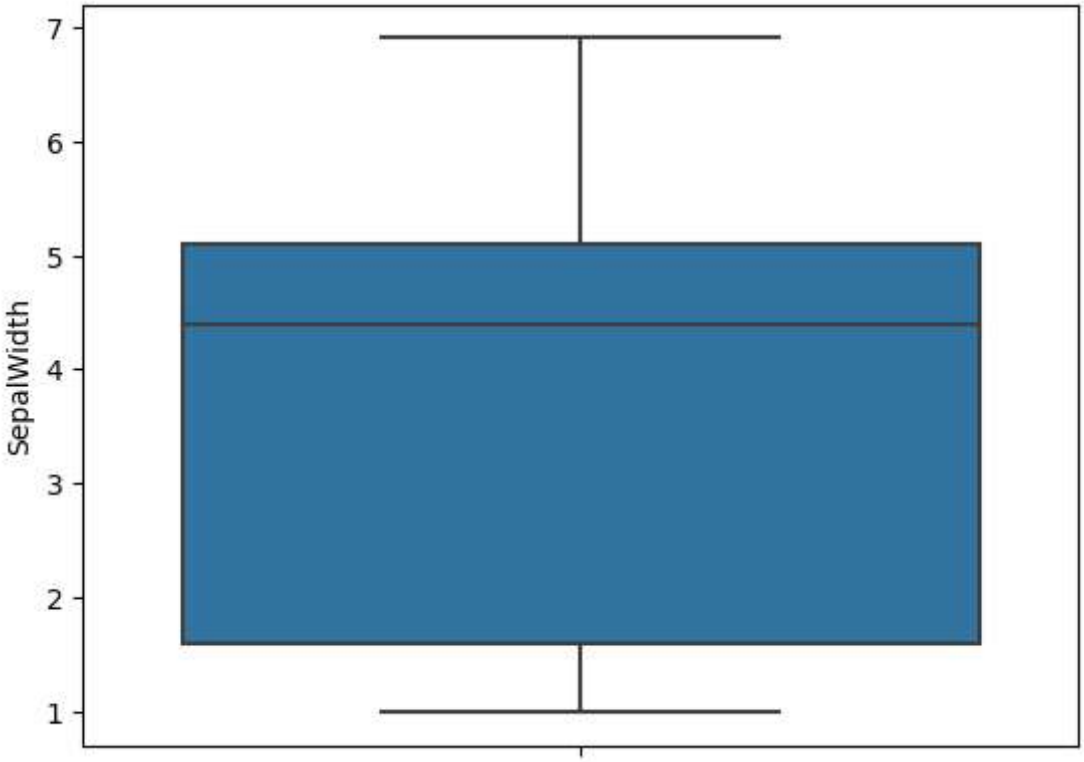


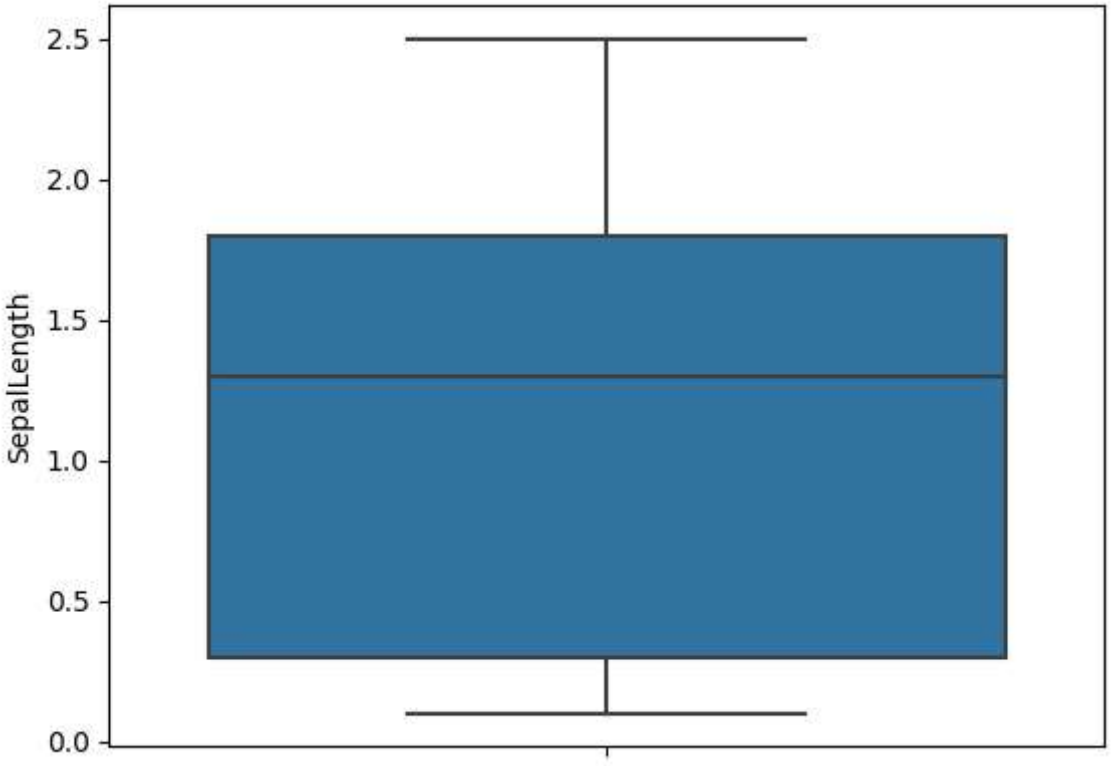
In [90]:
```
sns.boxplot(data=df, y='PetalLength')
plt.show()

sns.boxplot(data=df, y='PetalWidth')
plt.show()

sns.boxplot(data=df, y='SepalWidth')
plt.show()
```

```
sns.boxplot(data=df, y='SepalLength')
plt.show()
```

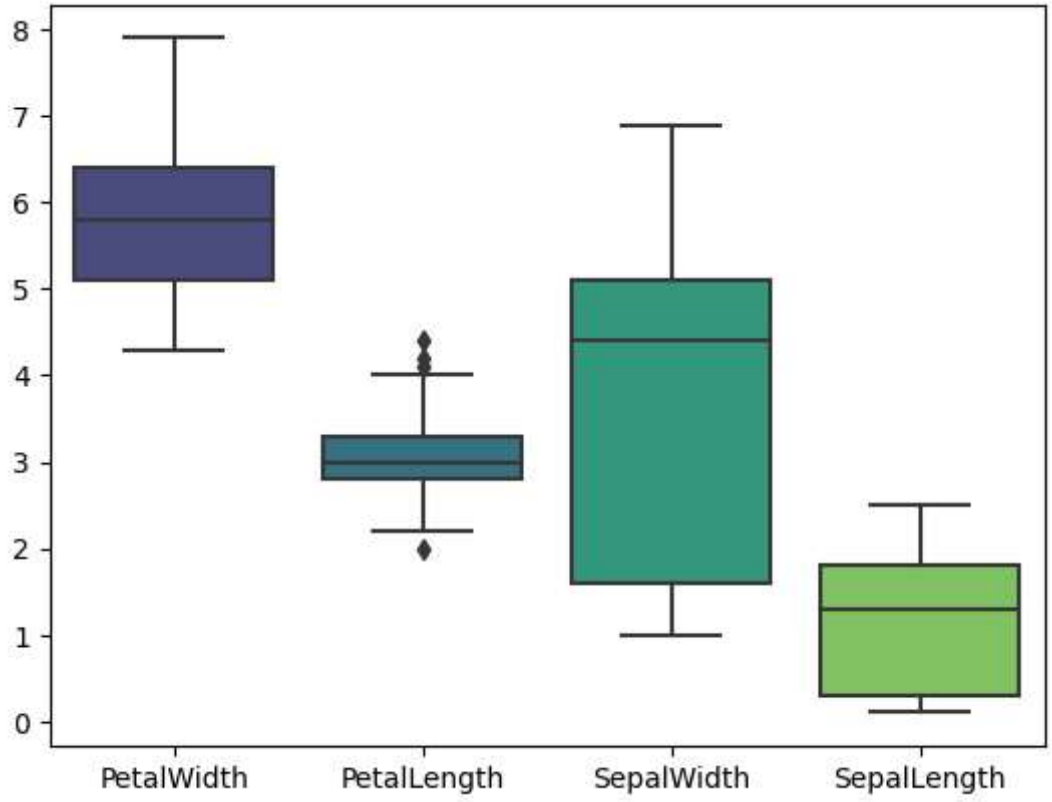




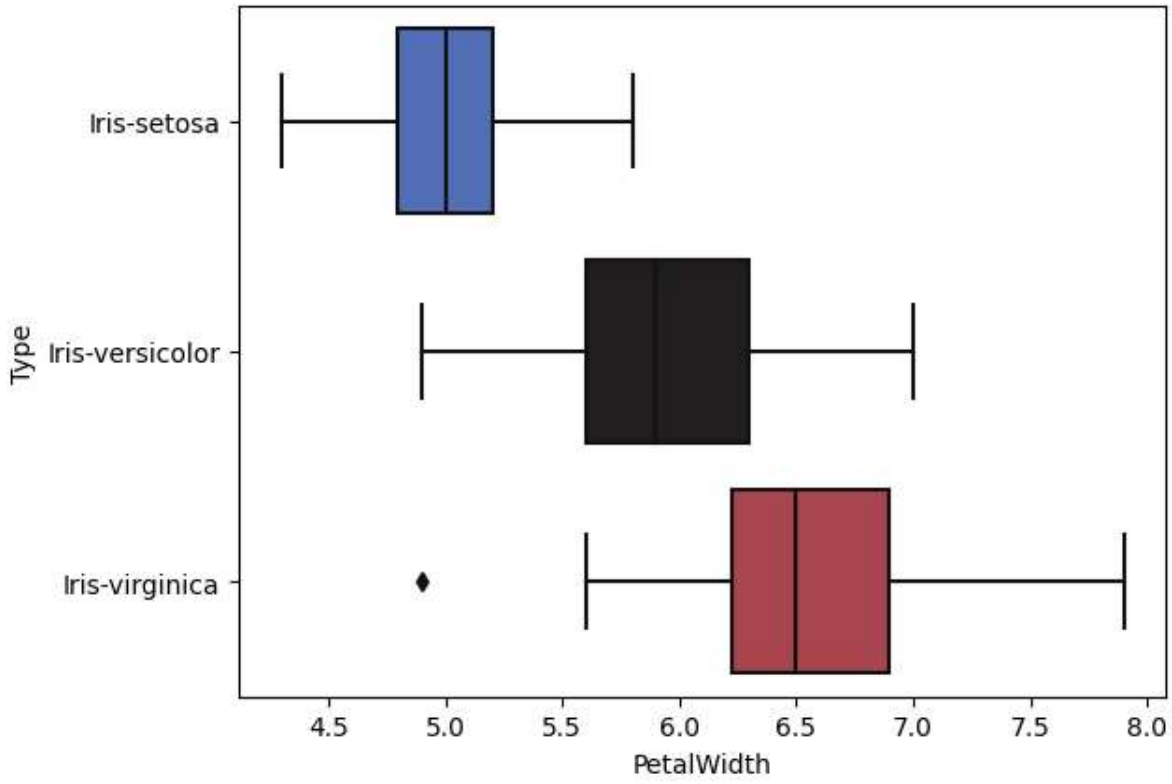




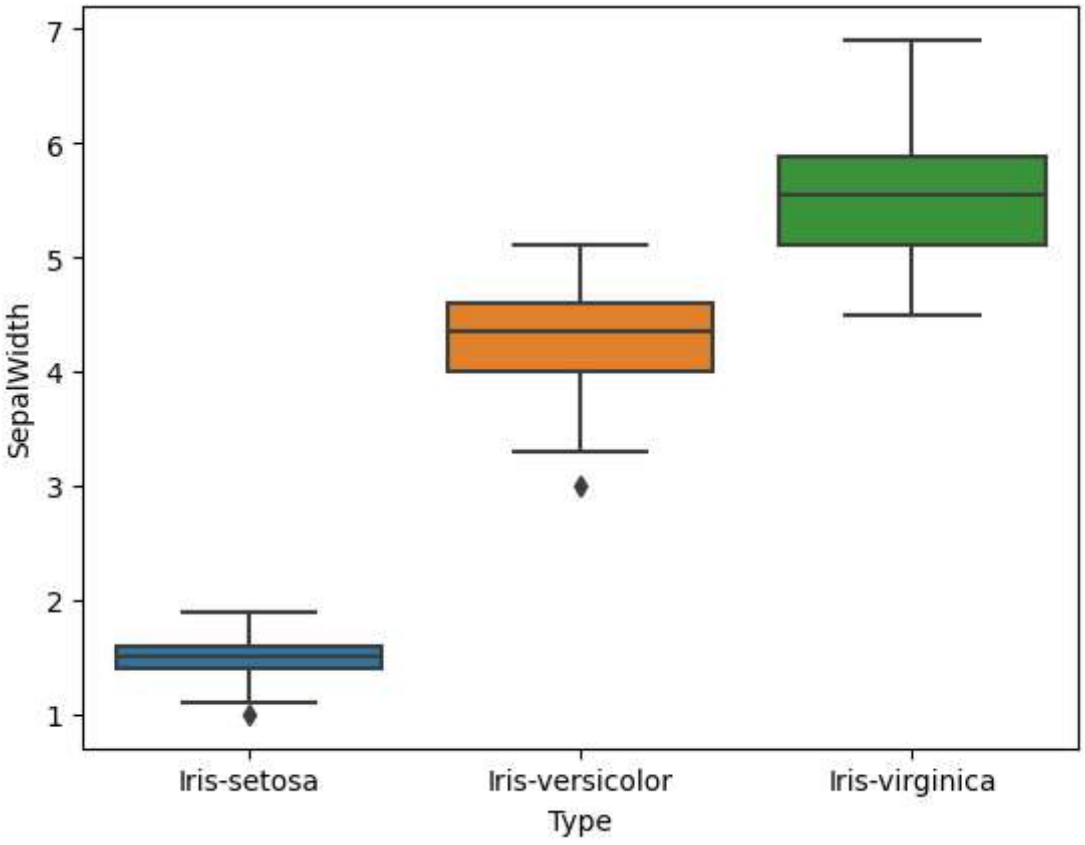
In [91]:
```
x = df.loc[:, ["PetalWidth", "PetalLength", "SepalWidth", "SepalLength"]]
xbp = sns.boxplot(data=x, orient="v", palette="viridis")
plt.show()
```



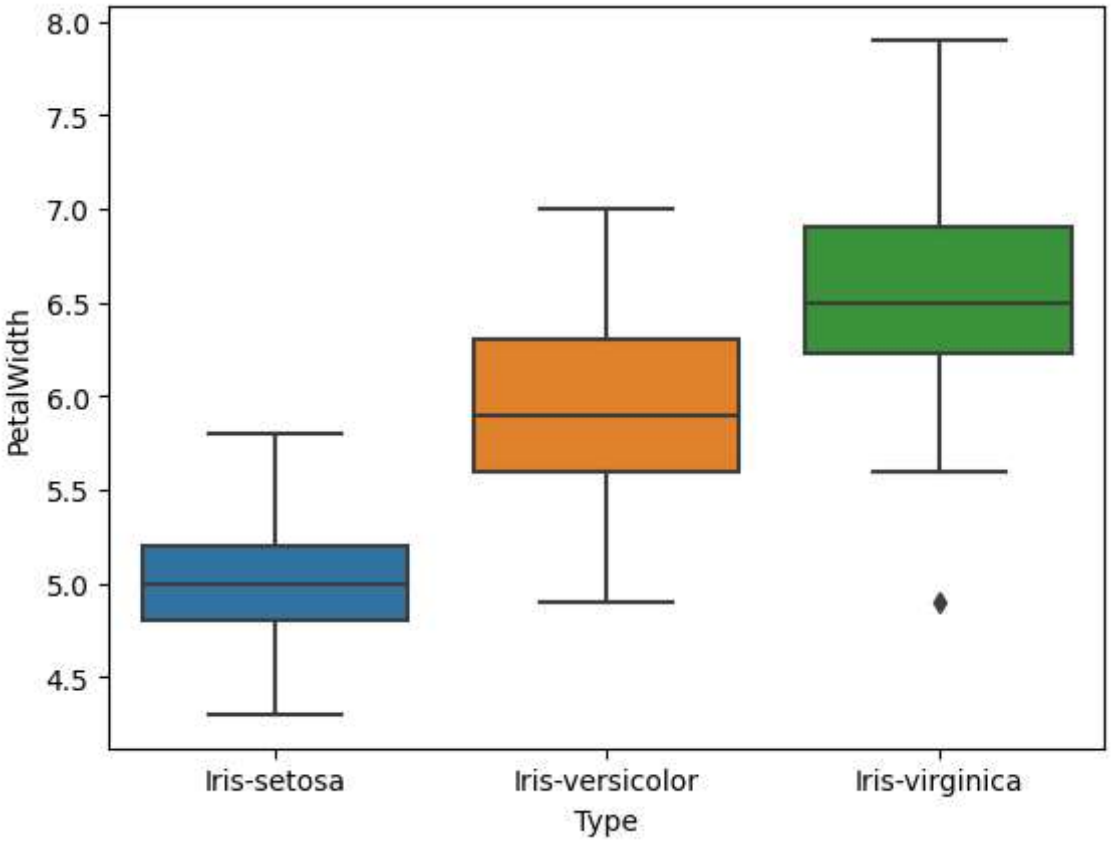
In [92]:
```
sns.boxplot(x = df["PetalWidth"], y = df["Type"], palette = "icefire")
plt.show()
```

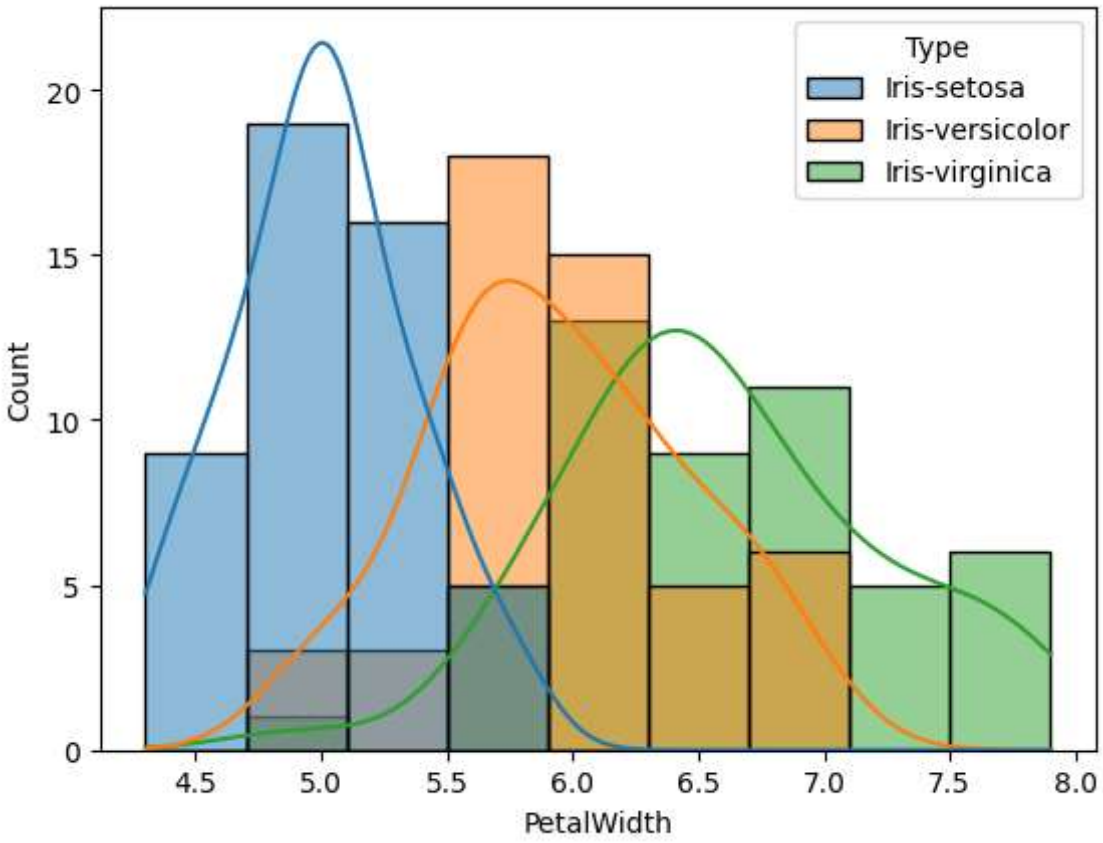


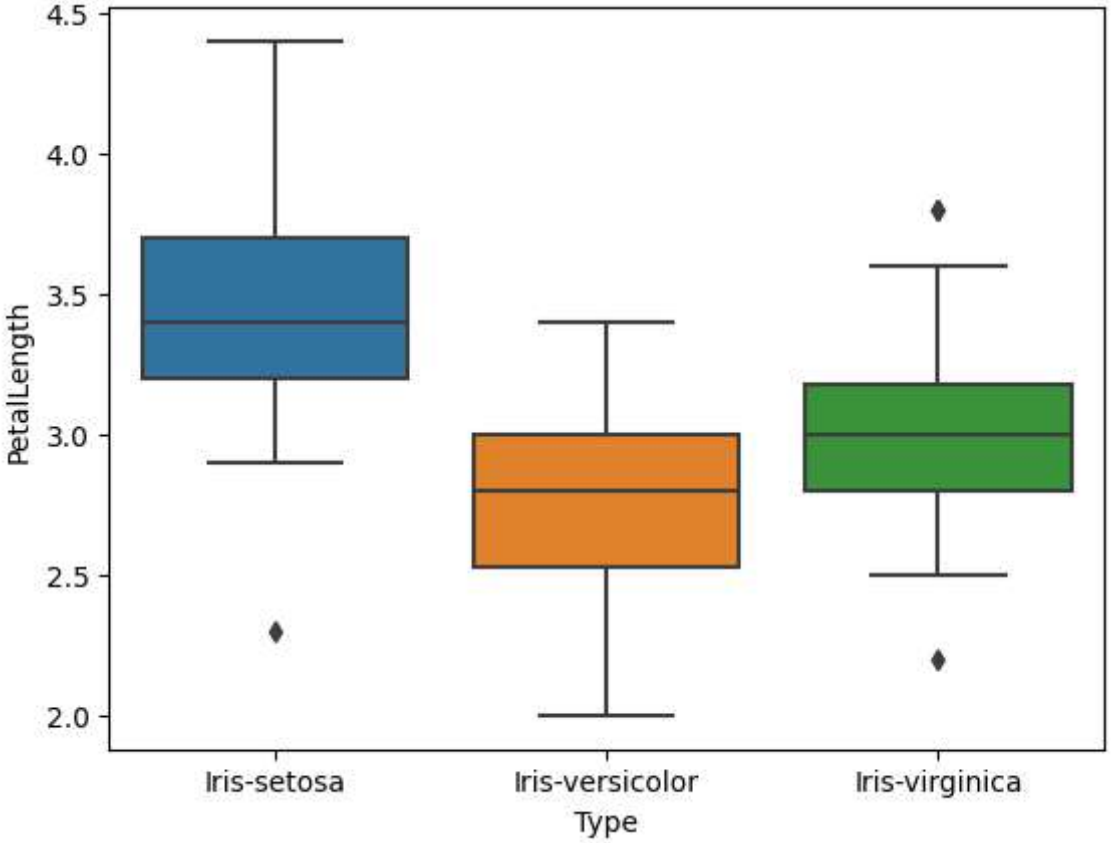
In [93]:
```
sns.boxplot(data=df, x='Type', y='SepalWidth')
plt.show()
```

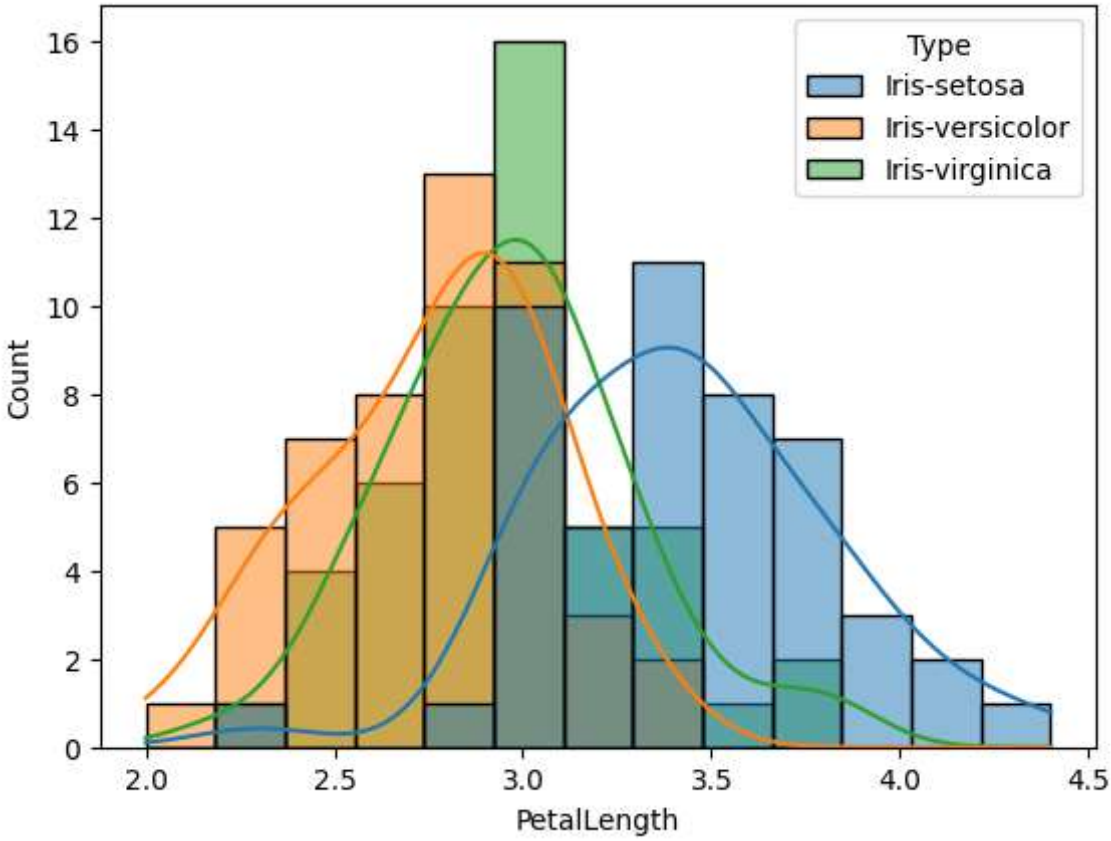


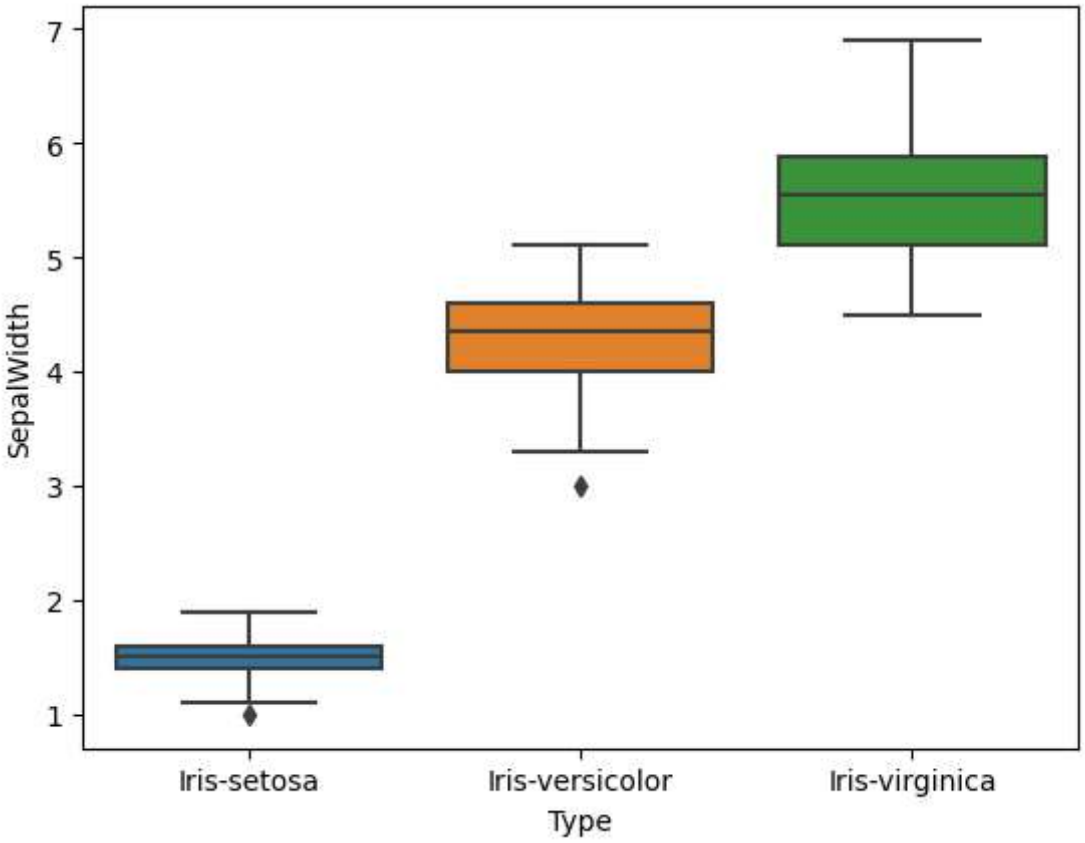
In [94]:
```
sns.boxplot(x="Type", y="PetalWidth", data=df)
plt.show()
sns.histplot(x="PetalWidth", hue="Type", data=df, kde=True)
plt.show()

sns.boxplot(x="Type", y="PetalLength", data=df)
plt.show()
sns.histplot(x="PetalLength", hue="Type", data=df, kde=True)
plt.show()

sns.boxplot(x="Type", y="SepalWidth", data=df)
plt.show()
sns.histplot(x="SepalWidth", hue="Type", data=df, kde=True)
plt.show()

sns.boxplot(x="Type", y="SepalLength", data=df)
plt.show()
sns.histplot(x="SepalLength", hue="Type", data=df, kde=True)
plt.show()
```

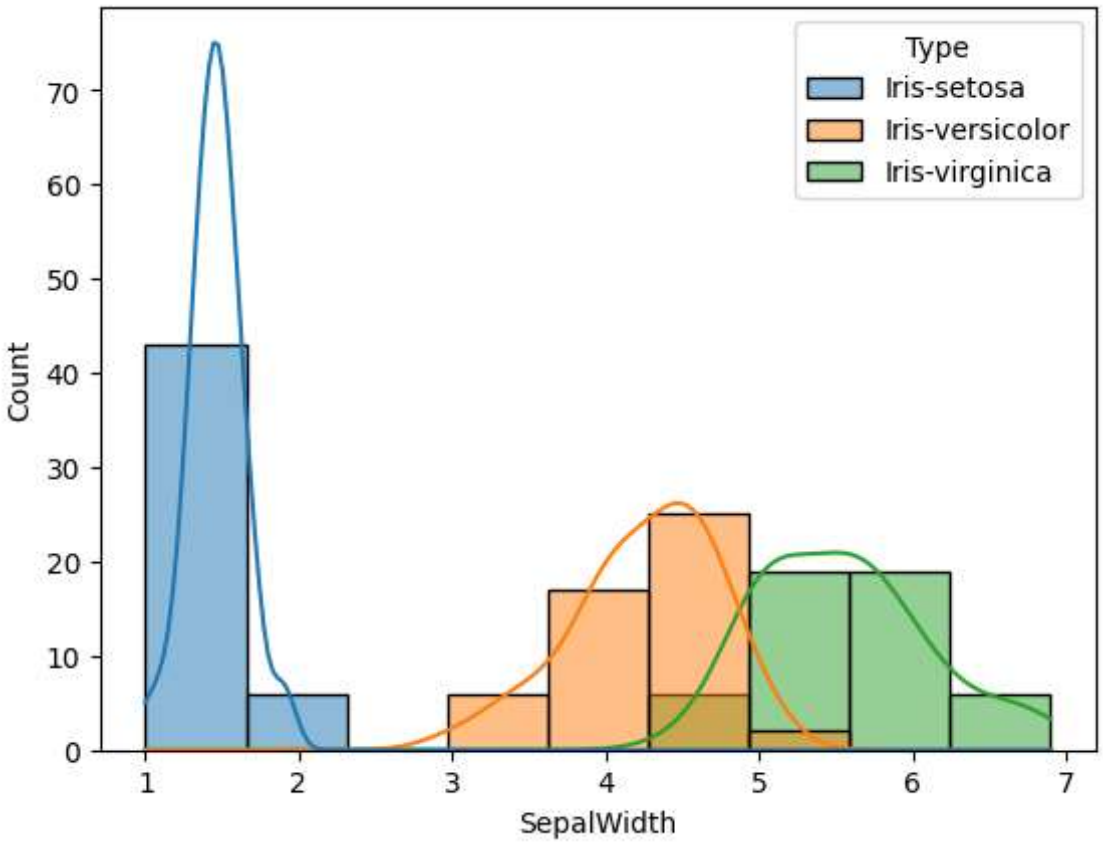

8.0
7.5
7.0
6.5
6.0
5.5
5.0
4.5
PetalWidth
Iris-setosa
Iris-versicolor
Iris-virginica
Type

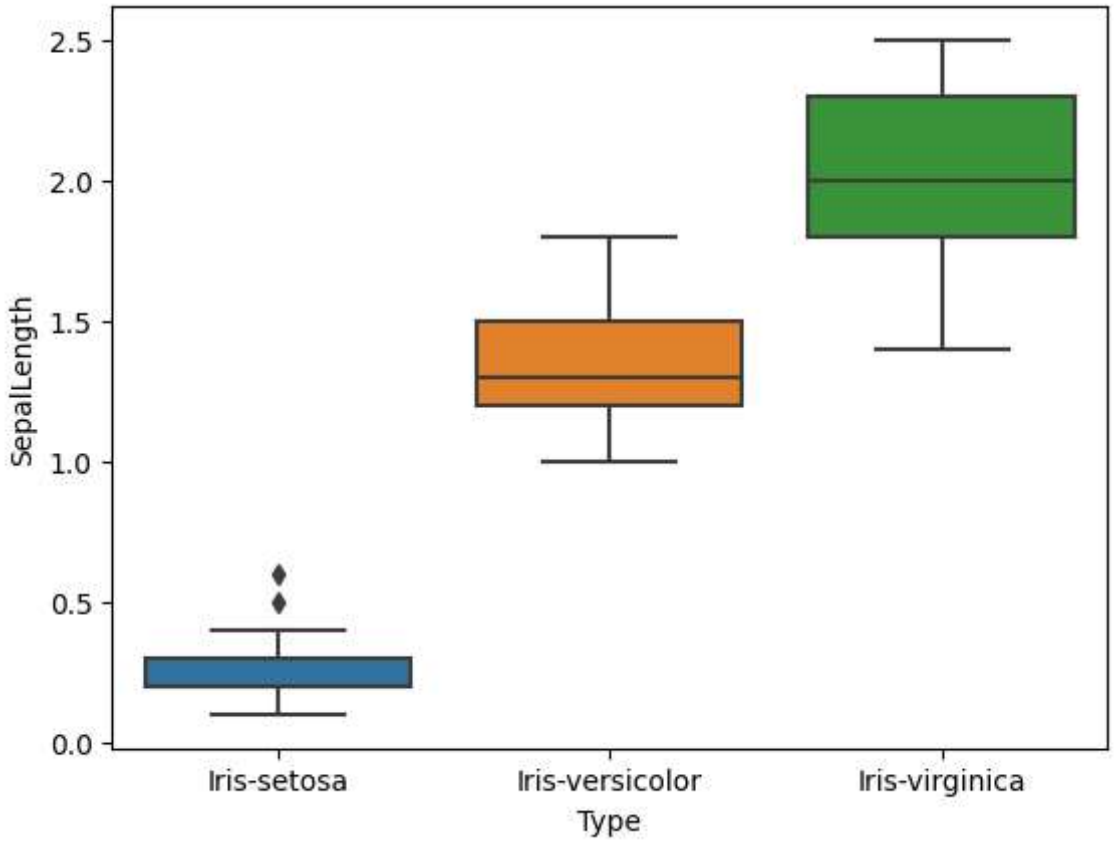

Type
Iris-setosa
Iris-versicolor
Iris-virginica
20
15
10
5
0
Count
4.5
5.0
5.5
6.0
6.5
7.0
7.5
8.0
PetalWidth

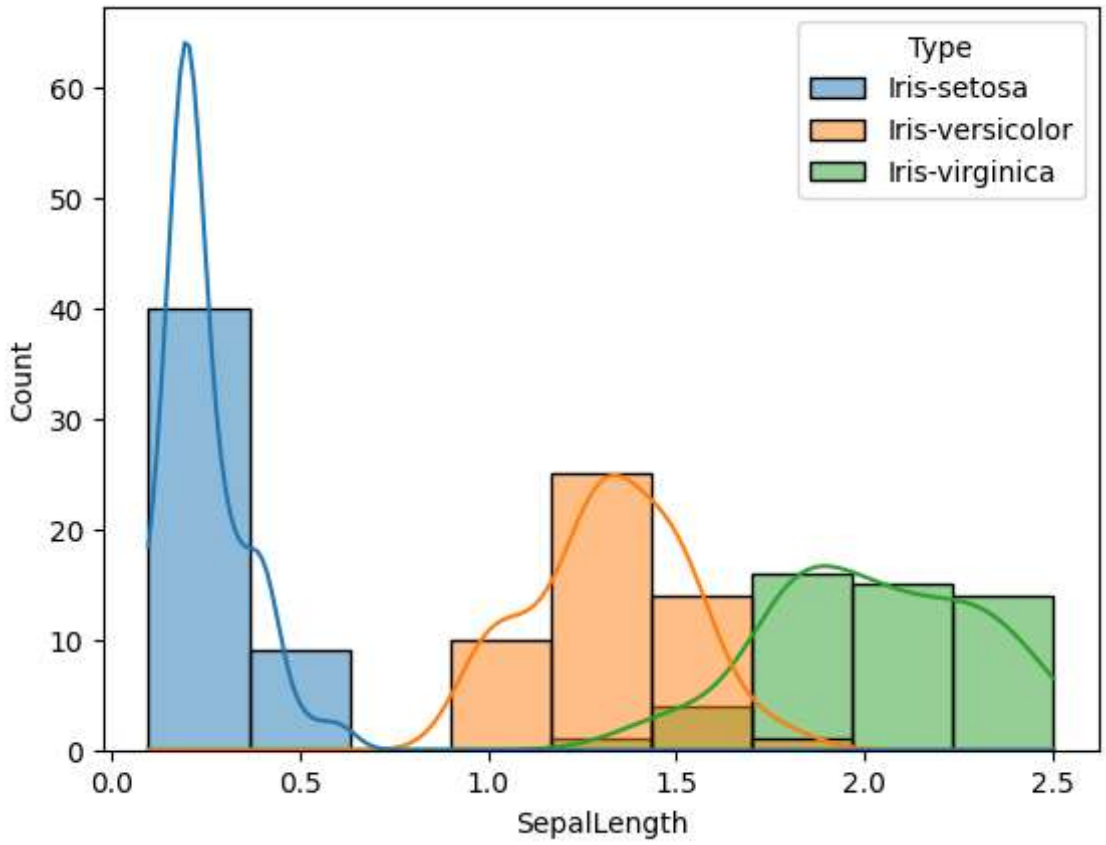

4.5
4.0
3.5
3.0
2.5
2.0
PetalLength
Iris-setosa
Iris-versicolor
Iris-virginica
Type


Type
Iris-setosa
Iris-versicolor
Iris-virginica
16
14
12
10
8
6
4
2
0
Count
2.0
2.5
3.0
3.5
4.0
4.5
PetalLength

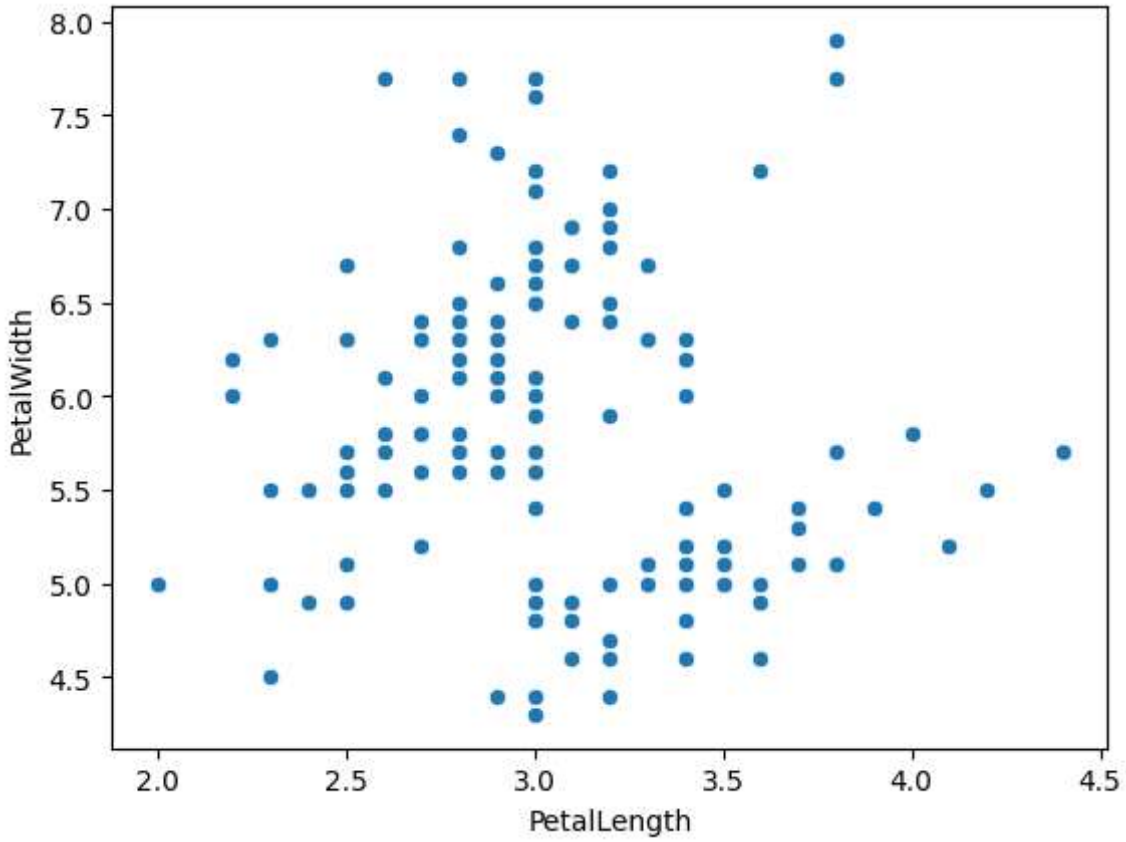

SepalWidth
Iris-setosa
Iris-versicolor
Iris-virginica
Type


Type
Iris-setosa
Iris-versicolor
Iris-virginica
Count
SepalWidth

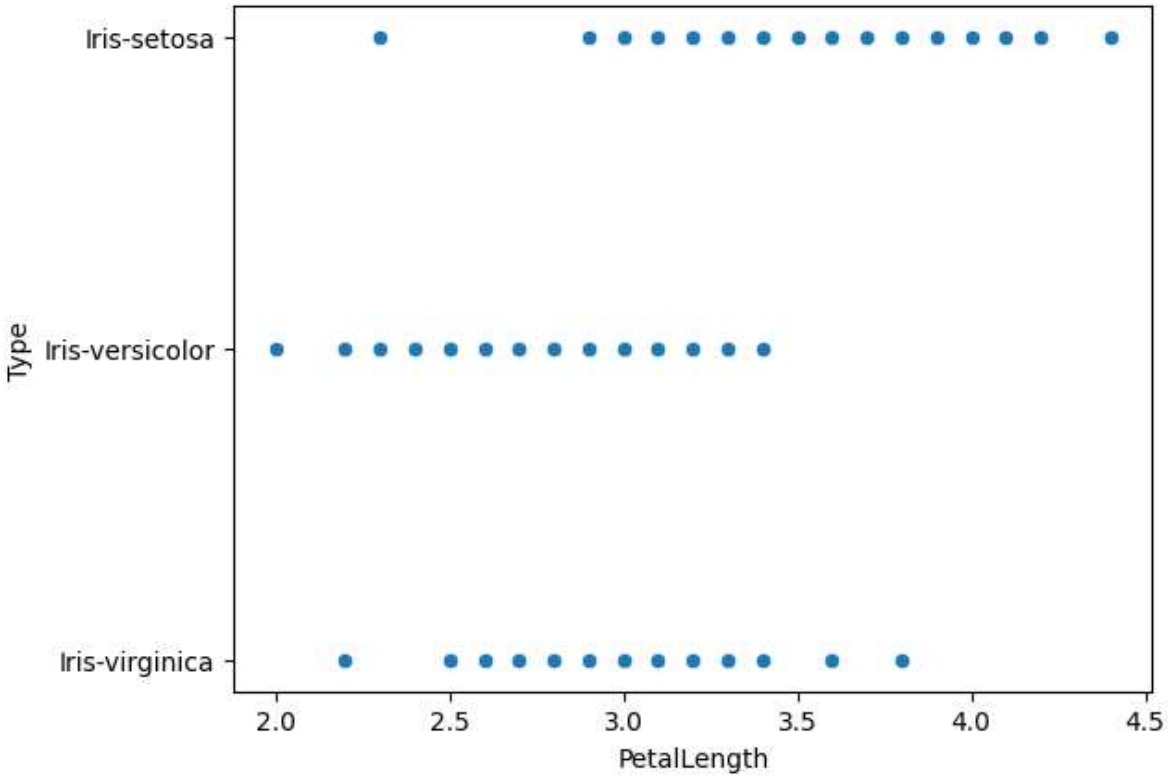




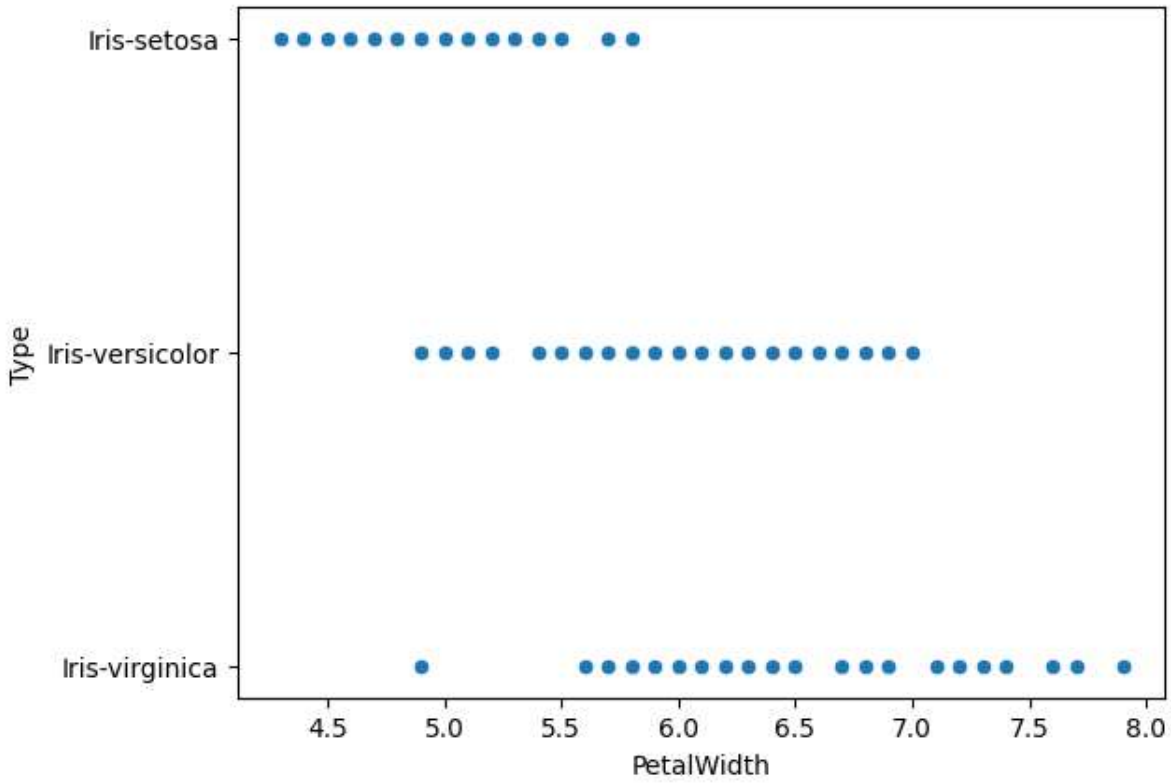
In [95]:
```
# scatter plot between two variables
sns.scatterplot(data = df, y = "PetalWidth", x = "PetalLength")
plt.show()
```



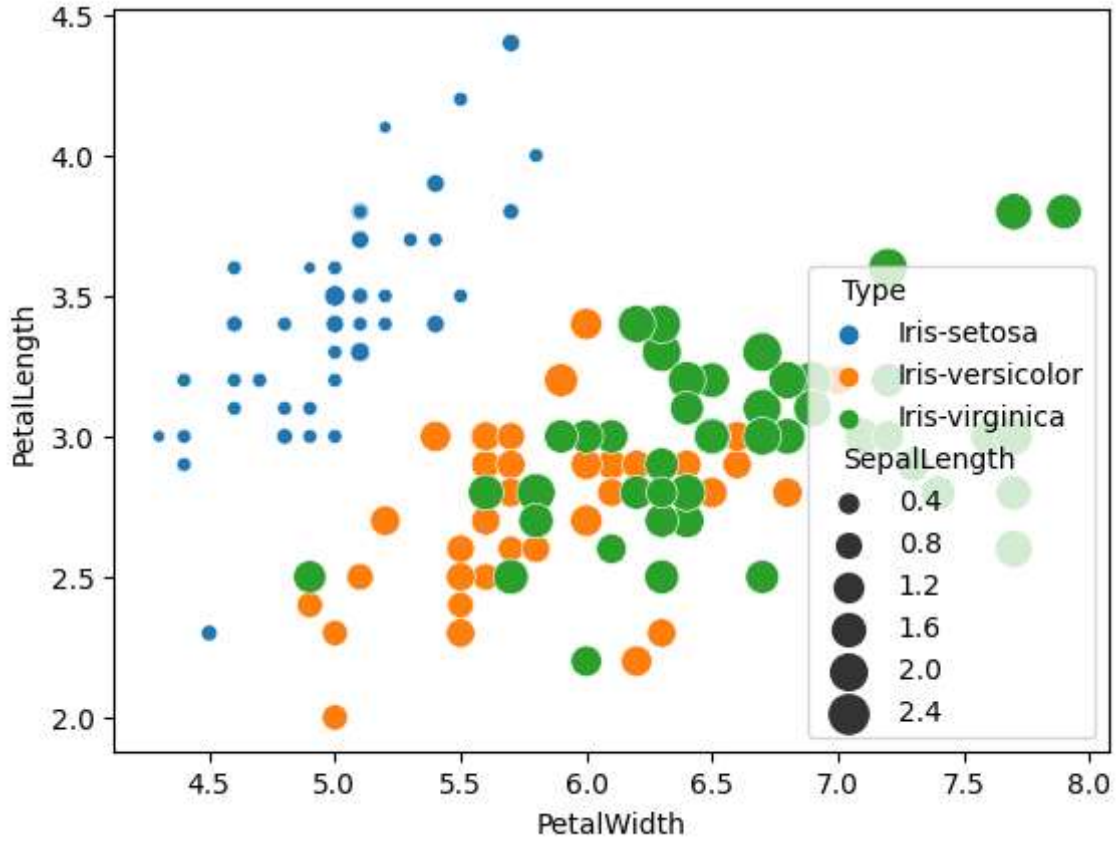
In [96]:
```
# scatter plot between two variables (one categorical)

sns.scatterplot(data = df, y = "Type", x = "PetalLength")
plt.show()
```



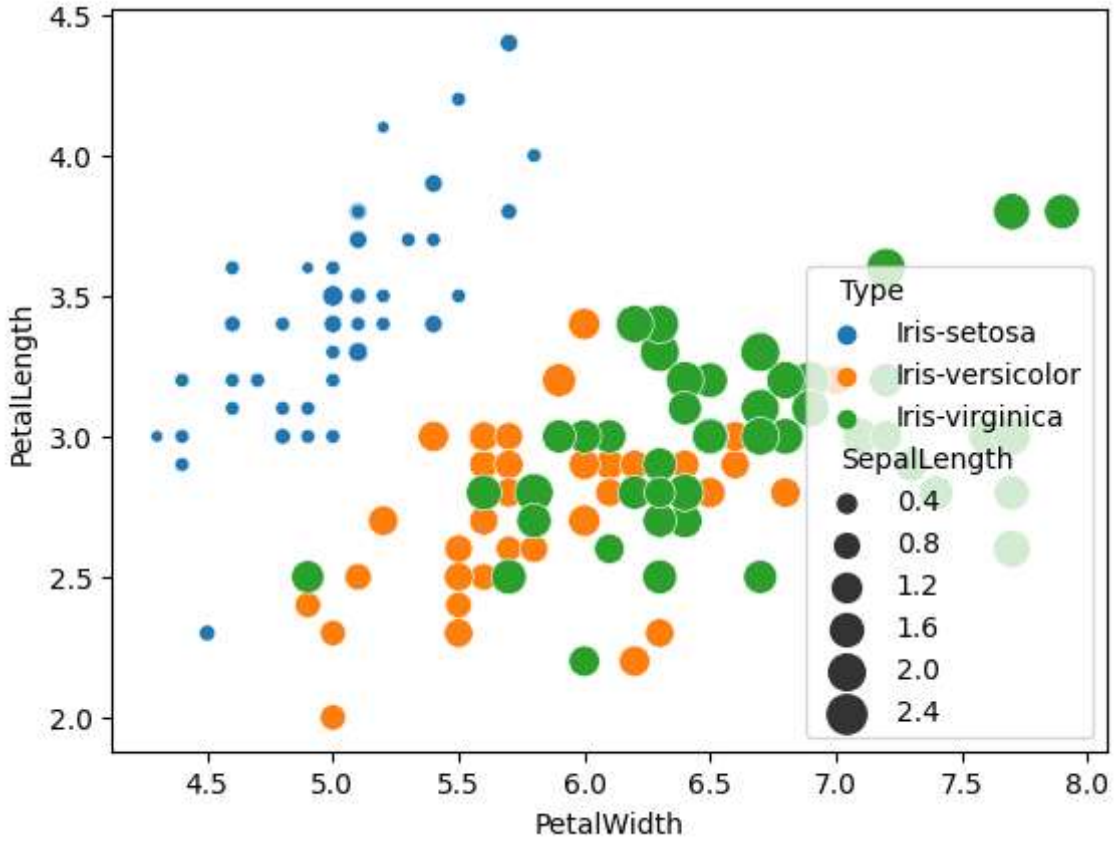
In [97]:
```
# scatter plot between two variables (one categorical)

sns.scatterplot(data = df, y = "Type", x = "PetalWidth")
plt.show()
```



In [98]:
```
# scatter plot between two variables grouped according to a categorical variable

sns.scatterplot(x='PetalWidth', y='PetalLength', hue='Type', size='SepalLength', si
plt.show()
```



In [99]:
```
# scatter plot between two variables grouped according to a categorical variable ar

sns.scatterplot(x='PetalWidth', y='PetalLength', hue='Type', size='SepalLength', si
plt.show()
```



Activity: work with the iris dataset

1. Plot the histograms for each of the four quantitative variables
2. Plot the histograms for each of the quantitative variables
3. Plot the boxplots for each of the quantitative variables
4. Plot the boxplots of the petal width grouped by type of flower
5. Plot the boxplots of the setal length grouped by type of flower
6. Provide a description (explaination from your observations) of each of the quantitative variables

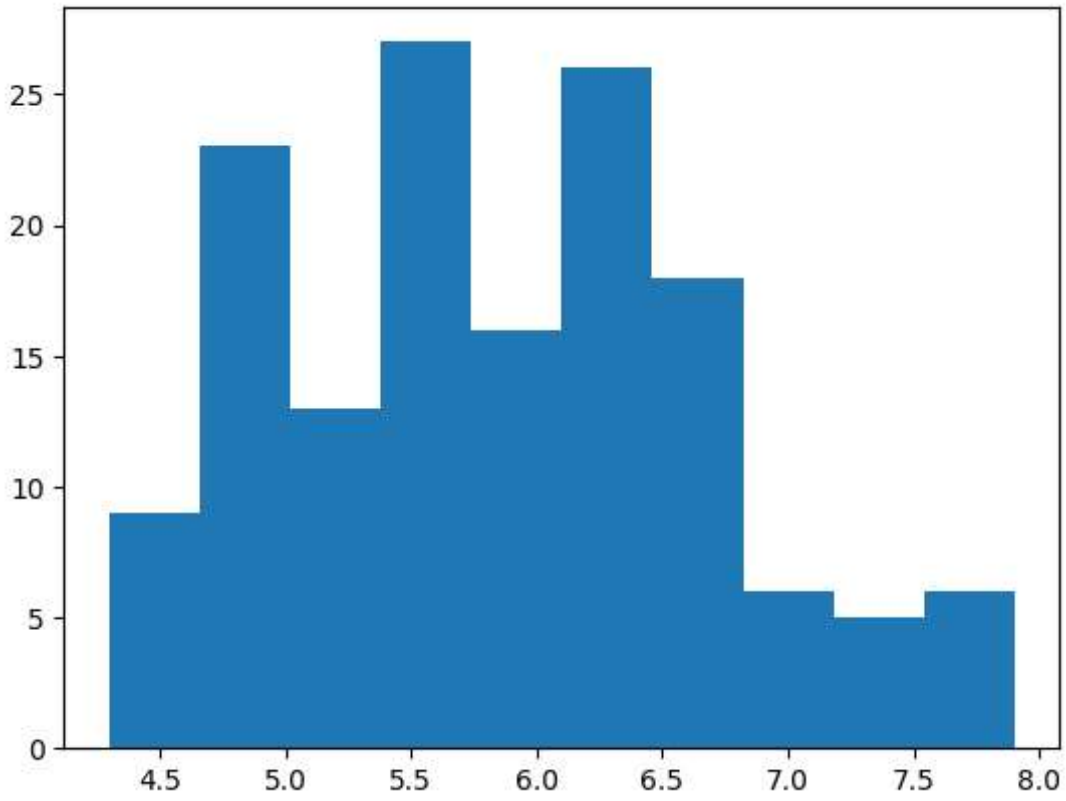
In [100…
```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

Ruta            = 'datasets'
```

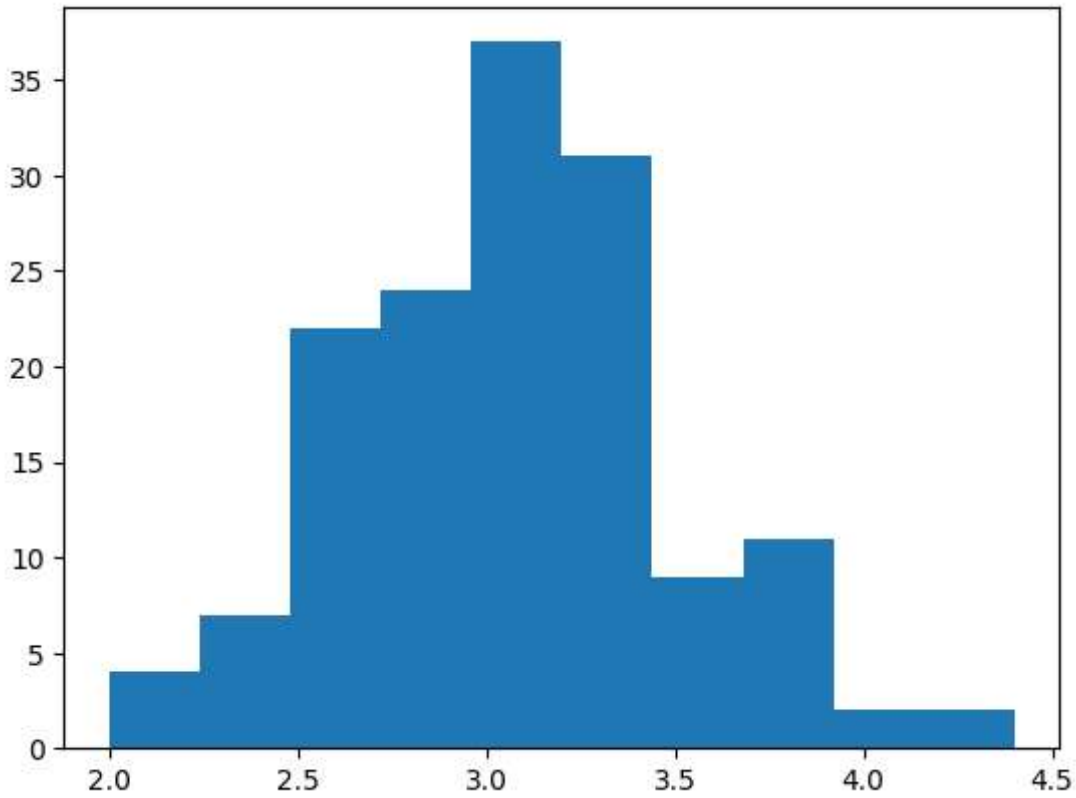
In [101…
```
newUrl = '/iris/iris.csv'

newDatos = pd.read_csv(Ruta+newUrl)
```

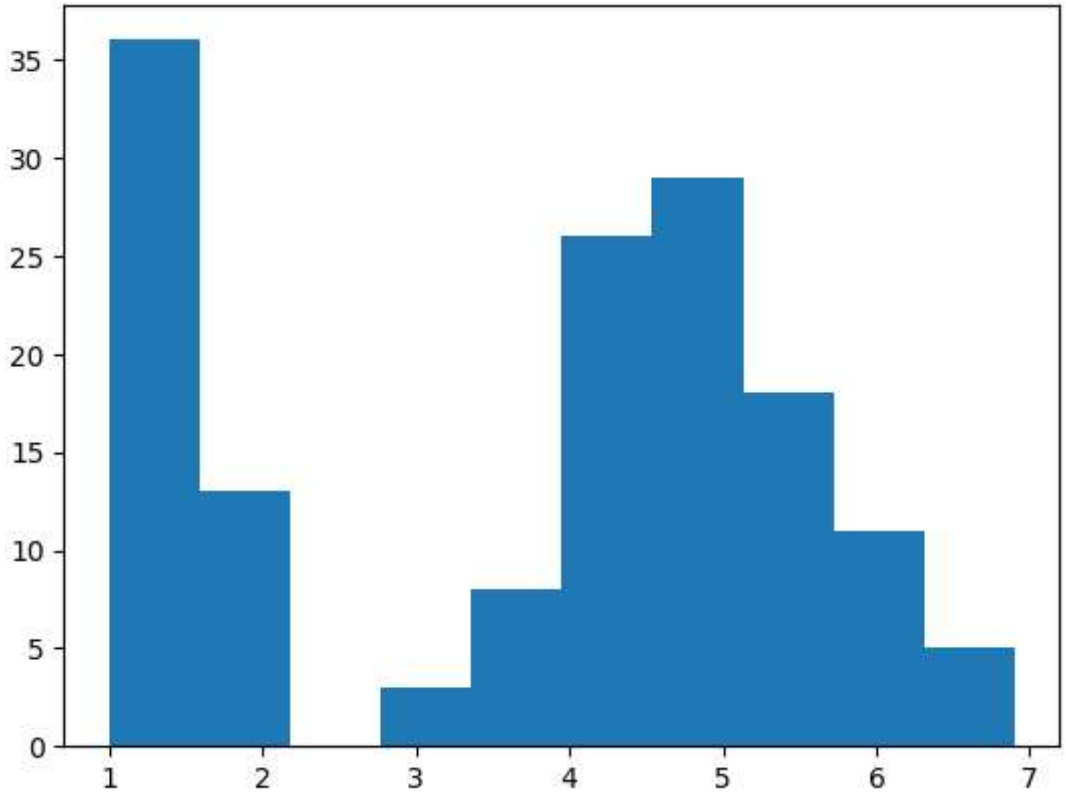
In [102…
```
newDatos.columns = ["PetalWidth", "PetalLength", "SepalWidth", "SepalLength", "Type
```

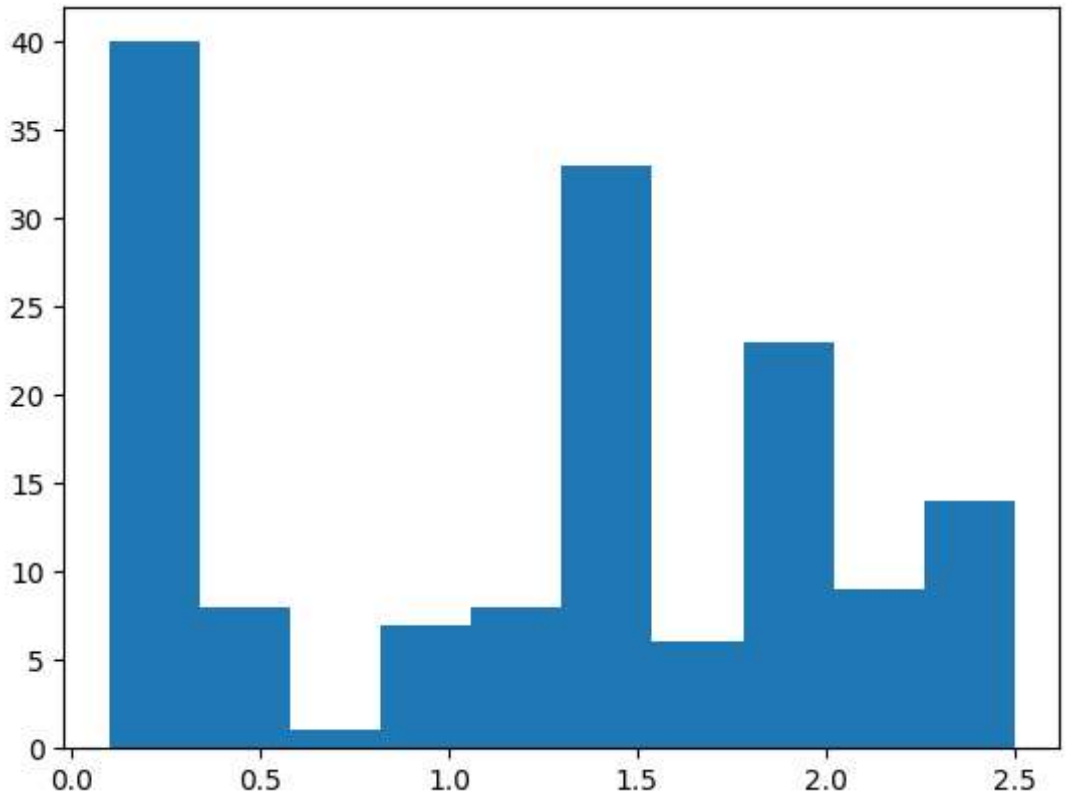
In [103…
```
#Plot the histograms for each of the four quantitative variables

plt.hist(newDatos["PetalWidth"])
plt.show()

plt.hist(newDatos["PetalLength"])
plt.show()
```

```
plt.hist(newDatos["SepalWidth"])
plt.show()

plt.hist(newDatos["SepalLength"])
plt.show()
```



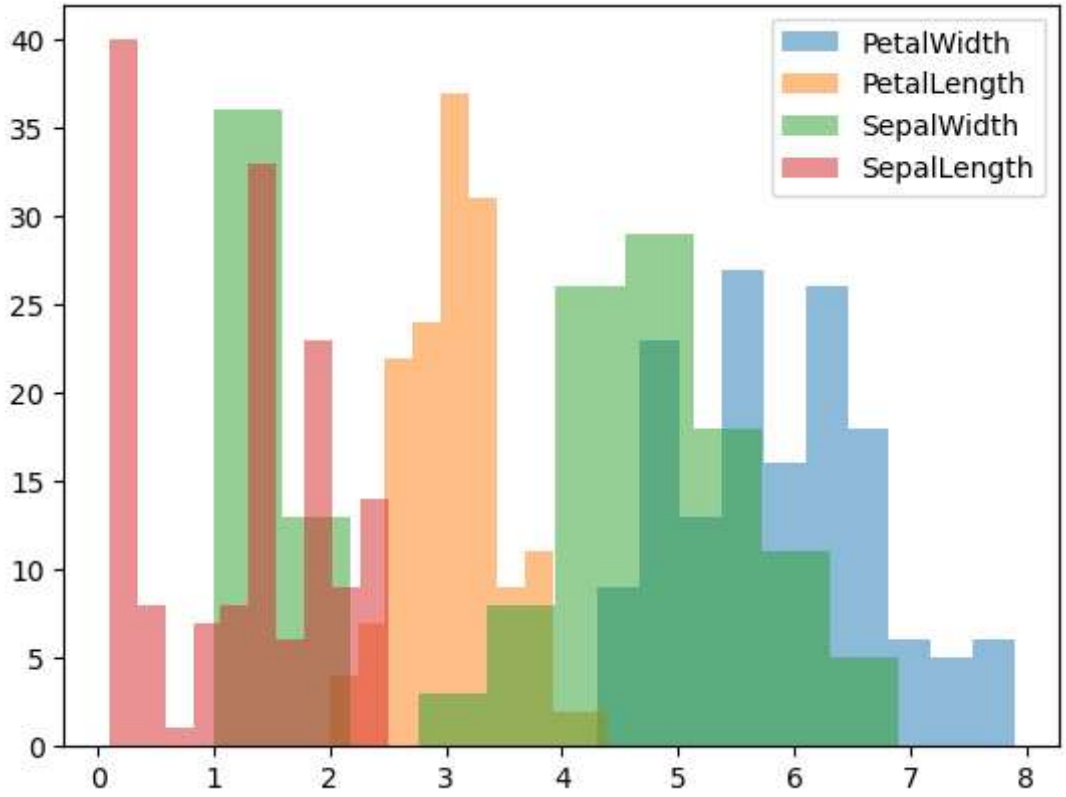






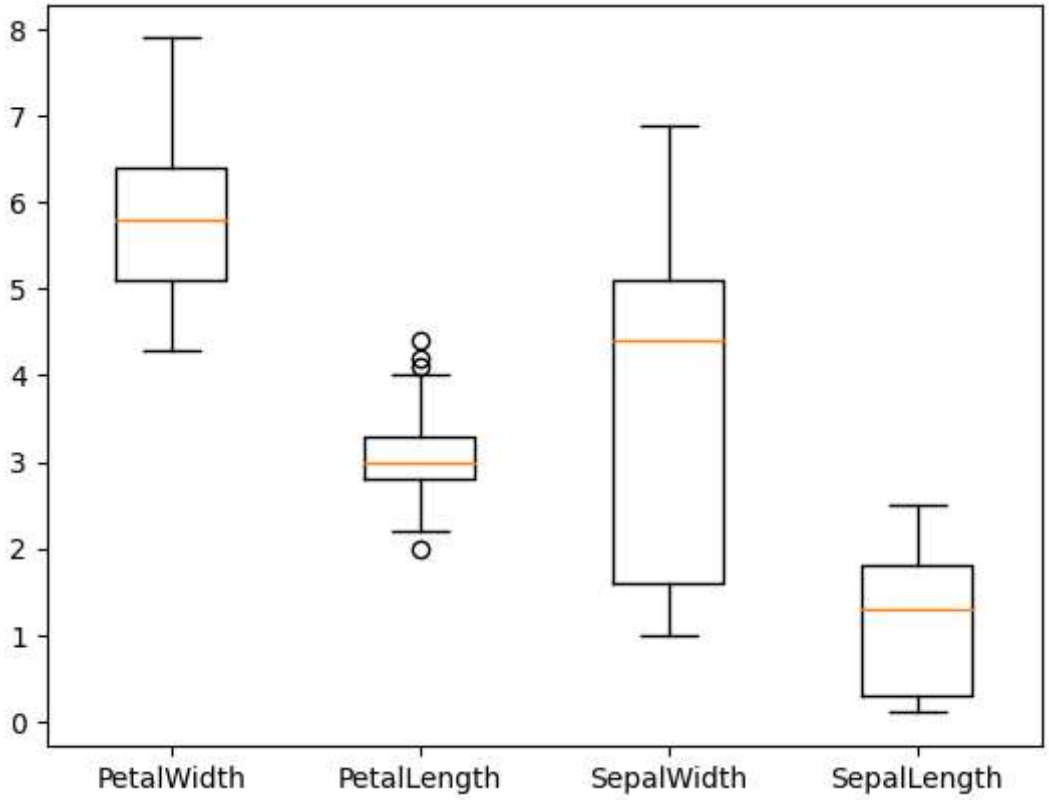
In [104…

```
# Plot the histograms for each of the quantitative variables

plt.hist(newDatos["PetalWidth"], alpha=0.5, label="PetalWidth")
plt.hist(newDatos["PetalLength"], alpha=0.5, label="PetalLength")
plt.hist(newDatos["SepalWidth"], alpha=0.5, label="SepalWidth")
plt.hist(newDatos["SepalLength"], alpha=0.5, label="SepalLength")
plt.legend()
plt.show()
```



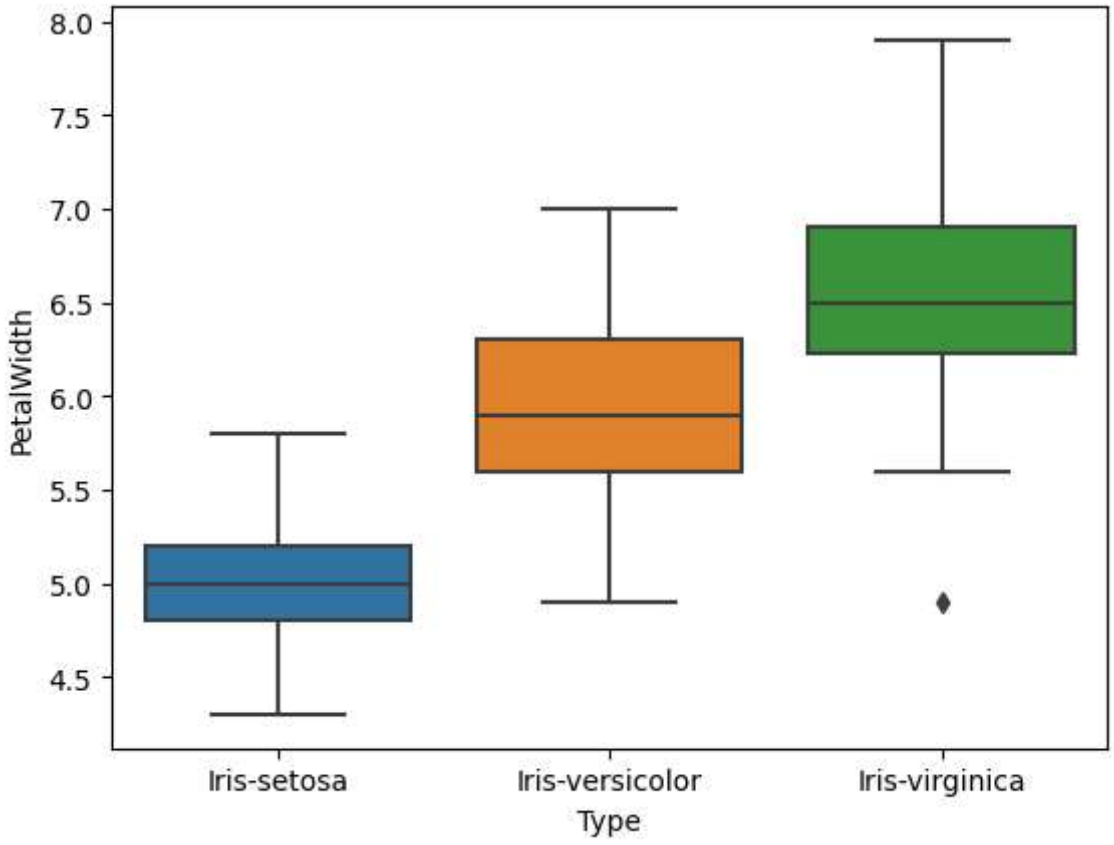
In [105…

```
#Plot the boxplots for each of the quantitative variables

plt.boxplot([newDatos["PetalWidth"], newDatos["PetalLength"], newDatos["SepalWidth"
plt.xticks([1, 2, 3, 4], ["PetalWidth", "PetalLength", "SepalWidth", "SepalLength"]
plt.show()
```



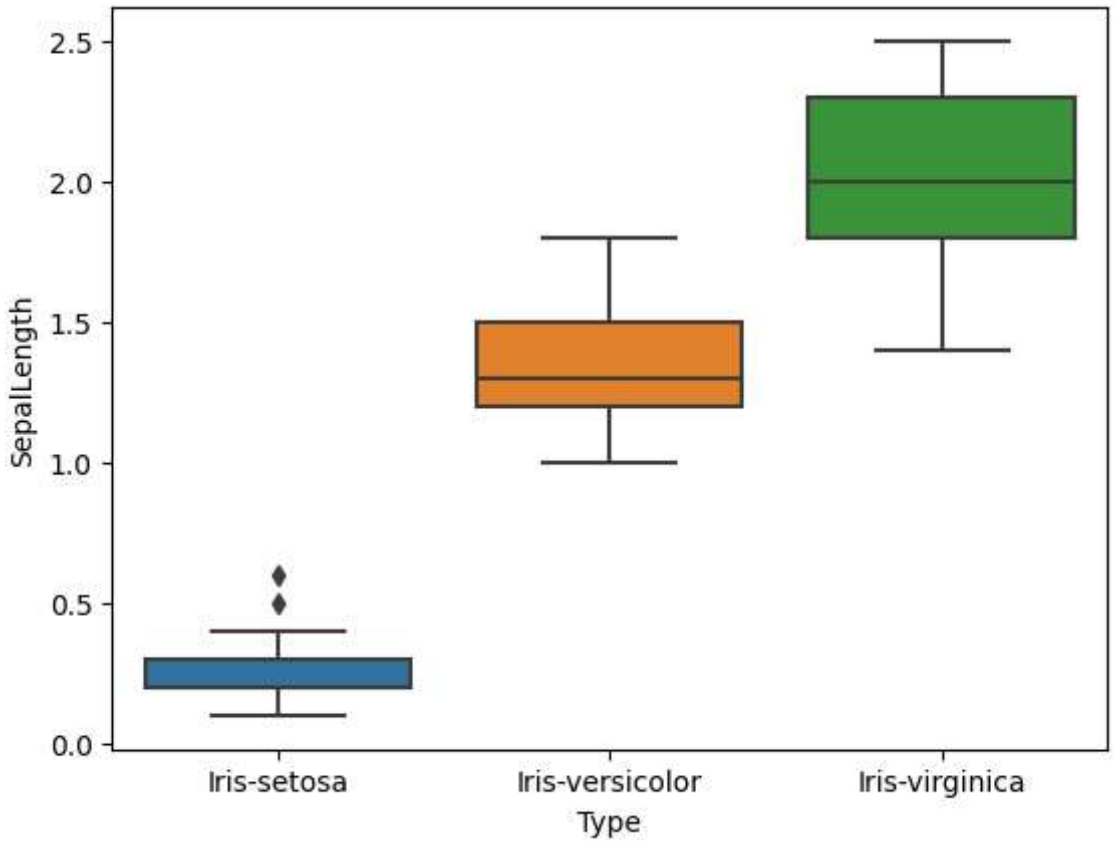
In [106…

```
#Plot the boxplots of the petal width grouped by type of flower

sns.boxplot(data=newDatos, x="Type", y="PetalWidth")
plt.show()
```



In [107…

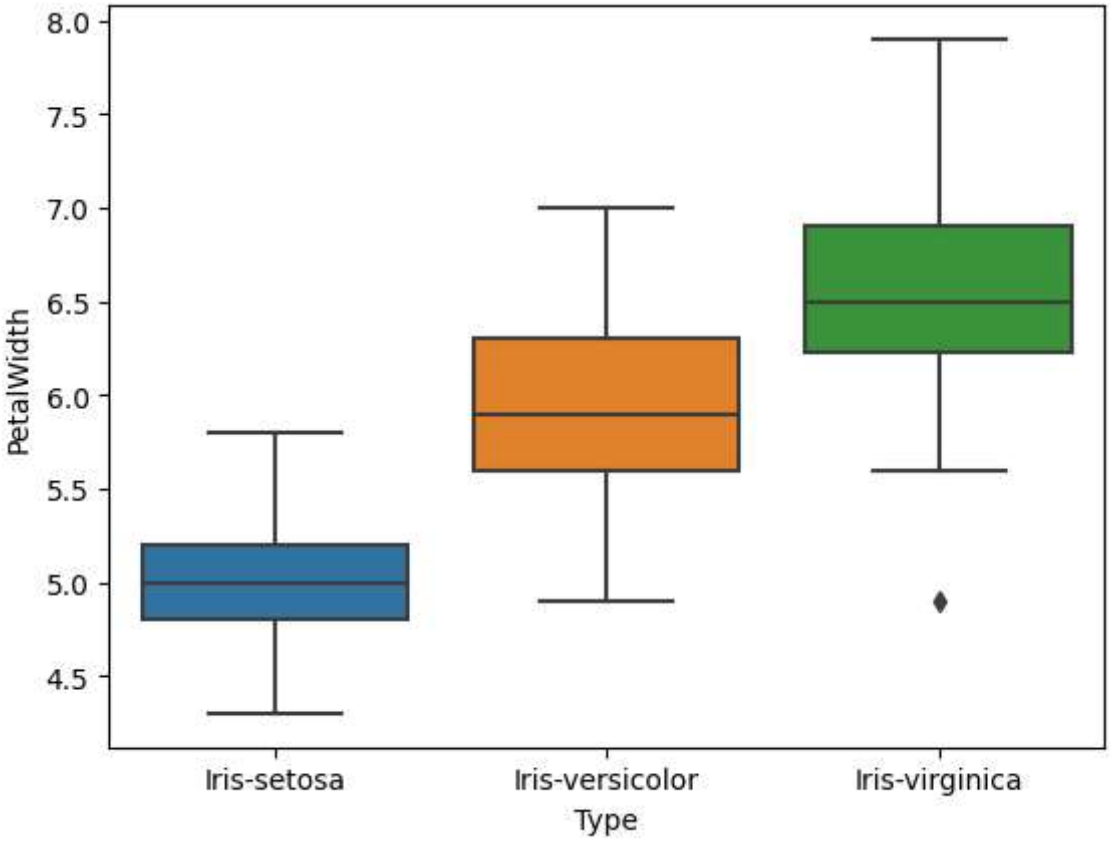
```
#Plot the boxplots of the sepal length grouped by type of flower

sns.boxplot(data=newDatos, x="Type", y="SepalLength")
plt.show()
```

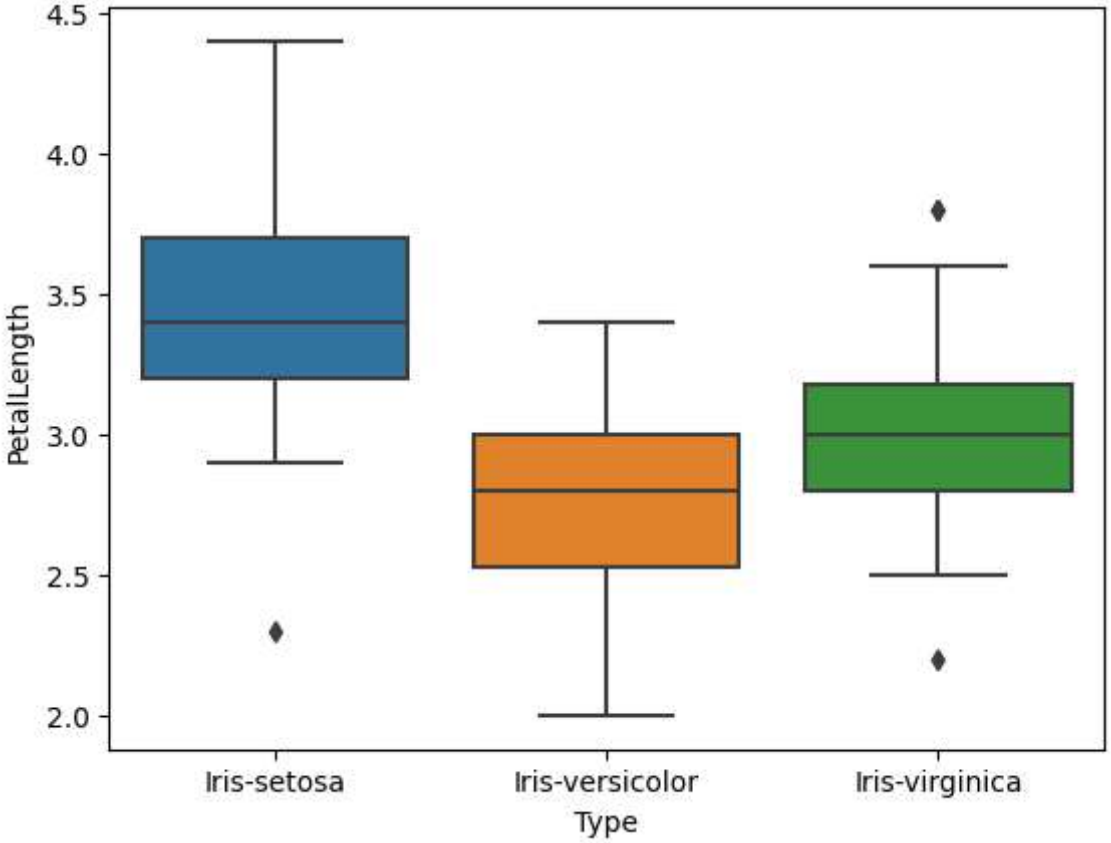


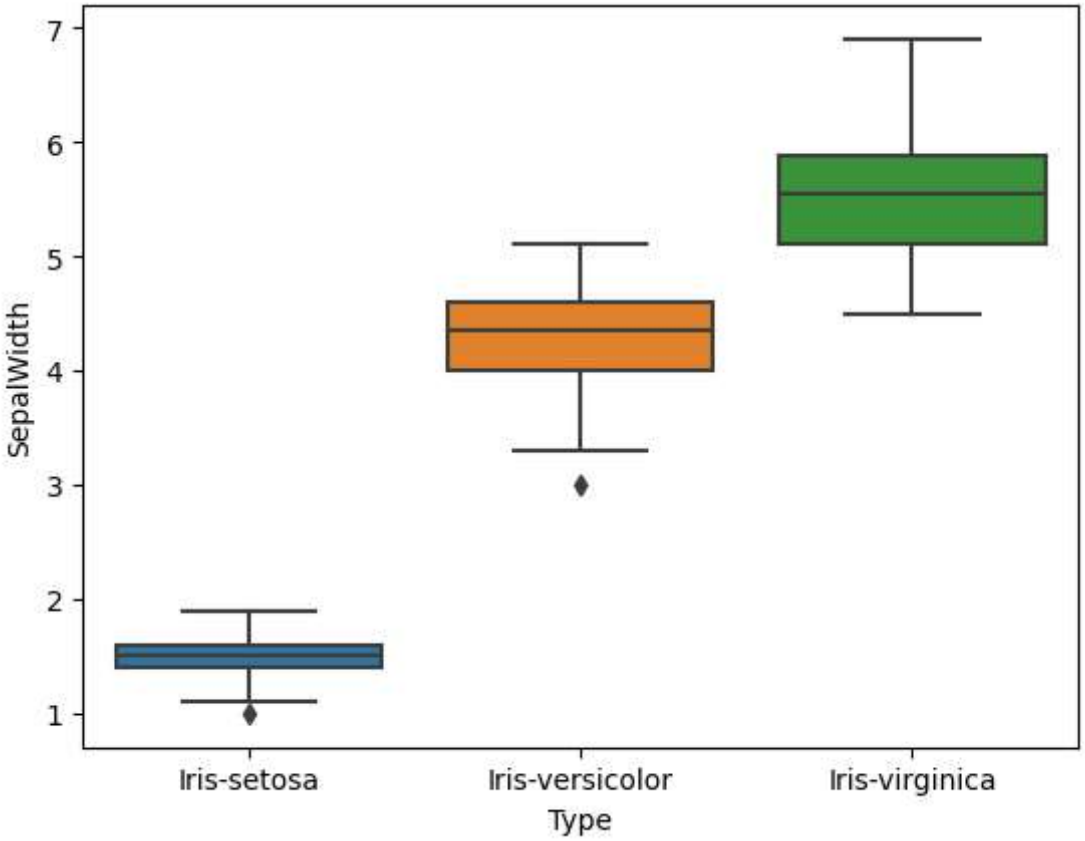
In [108…

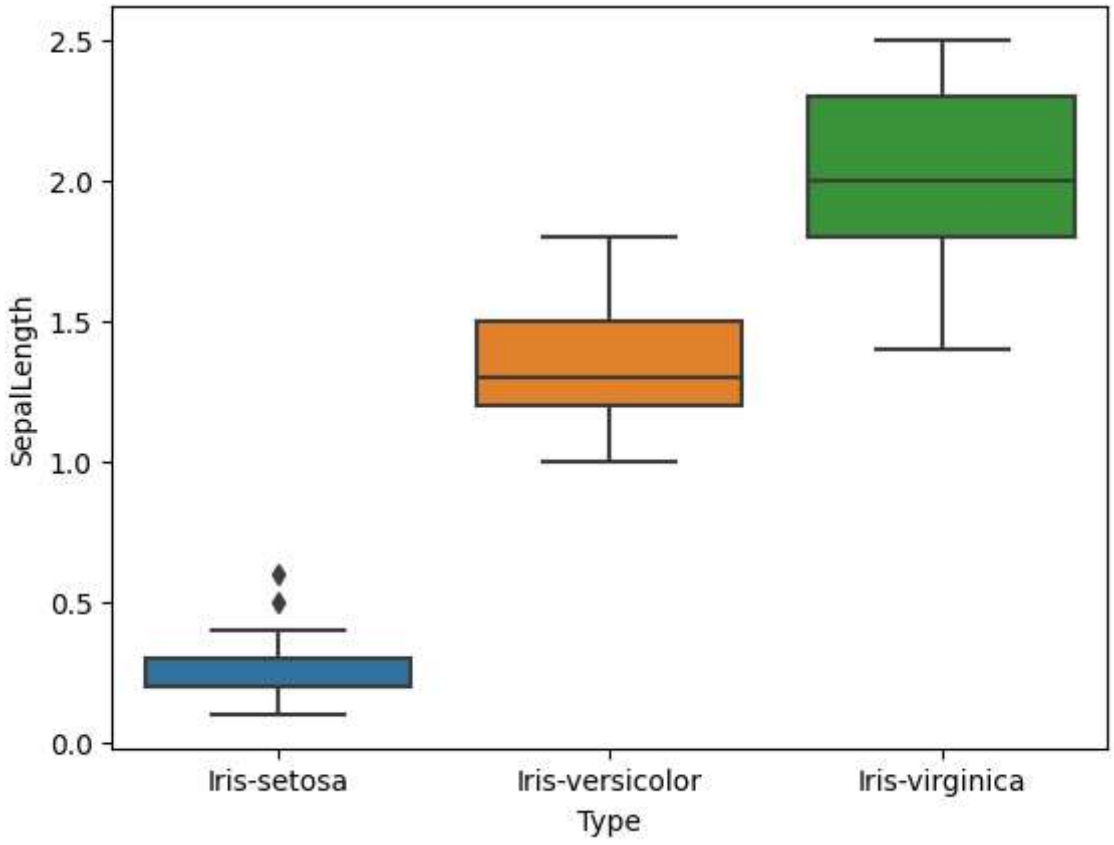
```
#All the boxplots grouped by type of flower

sns.boxplot(data=newDatos, x="Type", y="PetalWidth")
```

```
plt.show()

sns.boxplot(data=newDatos, x="Type", y="PetalLength")
plt.show()

sns.boxplot(data=newDatos, x="Type", y="SepalWidth")
plt.show()

sns.boxplot(data=newDatos, x="Type", y="SepalLength")
plt.show()
```




4.5
4.0
3.5
3.0
2.5
2.0
PetalLength
Iris-setosa
Iris-versicolor
Iris-virginica
Type


7
6
5
4
3
2
1
SepalWidth
Iris-setosa
Iris-versicolor
Iris-virginica
Type



In [109…

```
#Provide a description (explaination from your observations) of each of the quantit

"""
PetalWidth: Se puede asumir en función del análisis del ancho del petal que,
mientras más grande sea tiende a ser una Iris virgínica, y que
mientras más pequeña sea tiende a ser una Iris setosa.

PetalLength: Se puede asumir en función del análsis del largo del petal que,
mientras más grande sea tiende a ser una Iris setosa, y que
mientras más pequeña sea tiende a ser una Iris versicolor.

SepalWidth:Se puede asumir en función del análisis del ancho del sepal que,
mientras más grande sea tiende a ser una Iris virgínica, y que
mientras más pequeña sea tiende a ser una Iris setosa.

SepalLength: Se puede asumir en función del análisis del largo del sepal que,
mientras más grande sea tiende a ser una Iris virgínica, y que
mientras más pequeña sea tiende a ser una Iris setosa.

"""

"""
PetalWidth: Based on the analysis of petal width,
it can be assumed that the larger it is, the more likely it is an Iris virginica,
while the smaller it is, the more likely it is an Iris setosa.

PetalLength: Based on the analysis of petal length,
it can be assumed that the larger it is, the more likely it is an Iris setosa,
while the smaller it is, the more likely it is an Iris versicolor.

SepalWidth: Based on the analysis of sepal width,
it can be assumed that the larger it is, the more likely it is an Iris virginica,
while the smaller it is, the more likely it is an Iris setosa.
```

```
SepalLength: Based on the analysis of sepal length,
it can be assumed that the larger it is, the more likely it is an Iris virginica,
while the smaller it is, the more likely it is an Iris setosa.

"""
```

Out[109]:
```
'\nPetalWidth: Based on the analysis of petal width, \nit can be assumed that the
larger it is, the more likely it is an Iris virginica, \nwhile the smaller it is,
the more likely it is an Iris setosa.\n\nPetalLength: Based on the analysis of pet
al length, \nit can be assumed that the larger it is, the more likely it is an Iri
s setosa, \nwhile the smaller it is, the more likely it is an Iris versicolor.\n\n
SepalWidth: Based on the analysis of sepal width, \nit can be assumed that the lar
ger it is, the more likely it is an Iris virginica, \nwhile the smaller it is, the
more likely it is an Iris setosa.\n\nSepalLength: Based on the analysis of sepal l
ength, \nit can be assumed that the larger it is, the more likely it is an Iris vi
rginica, \nwhile the smaller it is, the more likely it is an Iris setosa.\n\n'
```

In [ ]: