In [21]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn import metrics
from sklearn.svm import SVC
```

In [2]:
```
data = pd.read_csv(r"C:\Users\91830\Desktop\DUK\AIML\svm\iris.csv")
```

In [3]:
```
data.head(5)
```

Out[3]:

	sepal.length	sepal.width	petal.length	petal.width	species
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 [4]:
```
data.dtypes
```

Out[4]:
```
sepal.length    float64
sepal.width     float64
petal.length    float64
petal.width     float64
species          object
dtype: object
```

In [5]:
```
data['target'] = data['species'].map({'Setosa': 0, 'Versicolor': 1, 'Virginica': 2})
```

In [6]:
```
data.head(5)
```

Out[6]:

	sepal.length	sepal.width	petal.length	petal.width	species	target
0	5.1	3.5	1.4	0.2	Setosa	0
1	4.9	3.0	1.4	0.2	Setosa	0
2	4.7	3.2	1.3	0.2	Setosa	0
3	4.6	3.1	1.5	0.2	Setosa	0
4	5.0	3.6	1.4	0.2	Setosa	0

In [7]:
```
data[data.target==1].head()
```

Out[7]:

	sepal.length	sepal.width	petal.length	petal.width	species	target
50	7.0	3.2	4.7	1.4	Versicolor	1
51	6.4	3.2	4.5	1.5	Versicolor	1
52	6.9	3.1	4.9	1.5	Versicolor	1
53	5.5	2.3	4.0	1.3	Versicolor	1
54	6.5	2.8	4.6	1.5	Versicolor	1

In [8]:
```
data[data.target==2].head()
```

Out[8]:

	sepal.length	sepal.width	petal.length	petal.width	species	target
100	6.3	3.3	6.0	2.5	Virginica	2
101	5.8	2.7	5.1	1.9	Virginica	2
102	7.1	3.0	5.9	2.1	Virginica	2
103	6.3	2.9	5.6	1.8	Virginica	2
104	6.5	3.0	5.8	2.2	Virginica	2

In [9]:
```
data.groupby('species').size()
```

Out[9]:
```
species
Setosa        50
Versicolor    50
Virginica     50
dtype: int64
```

In [10]:
```
train, test = train_test_split(data, test_size = 0.4, stratify = data['species'], random_state = 42
```

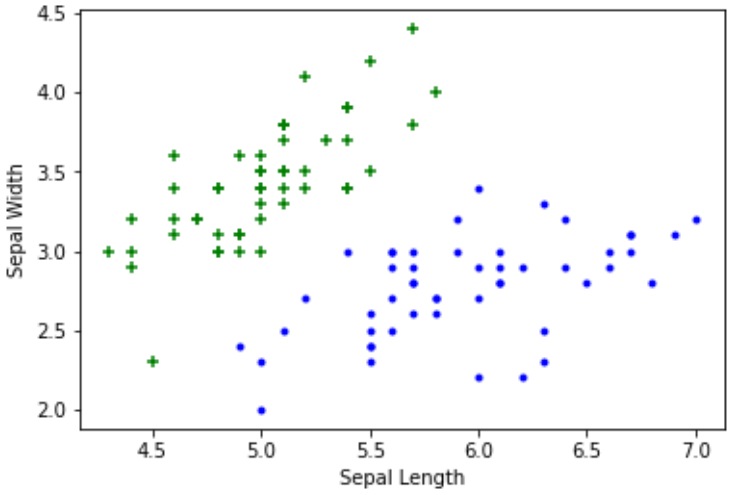
In [11]:
```
train.groupby('species').size()
```

Out[11]:
```
species
Setosa        30
Versicolor    30
Virginica     30
dtype: int64
```

In [13]:
```
df0 = data[:50]
df1 = data[50:100]
df2 = data[100:]
```

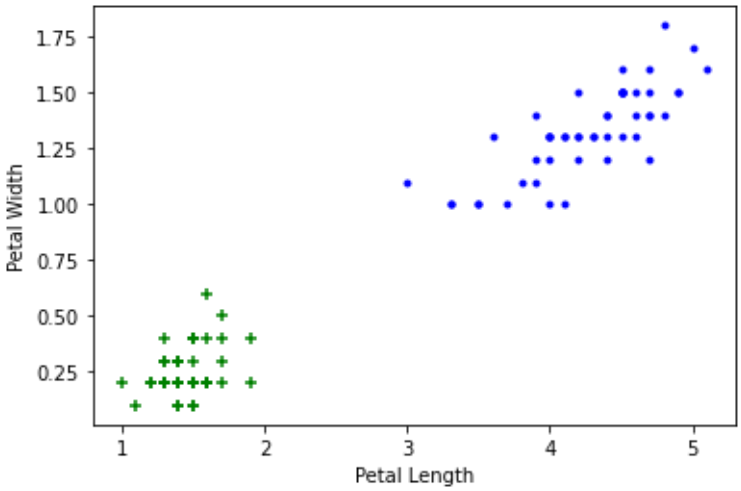
In [14]:
```
plt.xlabel('Sepal Length')
plt.ylabel('Sepal Width')
plt.scatter(df0['sepal.length'], df0['sepal.width'],color="green",marker='+')
plt.scatter(df1['sepal.length'], df1['sepal.width'],color="blue",marker='.')
```

Out[14]:
```
<matplotlib.collections.PathCollection at 0x202007ef7b8>
```



In [15]:
```
plt.xlabel('Petal Length')
plt.ylabel('Petal Width')
plt.scatter(df0['petal.length'], df0['petal.width'],color="green",marker='+')
plt.scatter(df1['petal.length'], df1['petal.width'],color="blue",marker='.')
```

Out[15]:
```
<matplotlib.collections.PathCollection at 0x202009ba7b8>
```



In [16]:
```
X = data.drop(['target','species'], axis='columns')
y = data.target
```

In [17]:
```
#Model Development
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2)
```

In [18]:
```
len(X_train)
```

Out[18]: 120

In [19]:
```
len(X_test)
```

Out[19]: 30

In [32]:
```
# SVC with linear kernel
linear_svc = SVC(kernel='linear').fit(X_train, y_train)
prediction=linear_svc.predict(X_test)
print('The accuracy of the linear SVC is',"{:.3f}".format(metrics.accuracy_score(prediction,y_test)
```

```
The accuracy of the linear SVC is 96.667 %
```