In [ ]:

```
from sklearn.datasets import load_iris
import pandas as pd
import matplotlib.pyplot as plt
```

In [ ]:

```
data = load_iris()

#data
```

In [ ]:

```
## to find how many features are there in dataset

data.shape
```

In [ ]:

```
# Extract 'data' attr from data object, column names from attr 'feature_names'
df = pd.DataFrame(data=data['data'], columns=data['feature_names'])

# adding a col of target to the dataframe
df['target'] = data['target']
```

In [ ]:

```
df.head()
```

Out[ ]:

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

In [ ]:

```
y = df['target']

# axis=1 (or axis='columns') is vertical axis - col
# axis=0 (or axis='row') is horizontal axis - row
x = df.drop(['target'],axis=1)
```

In [ ]:

```
x.dtypes
```

Out[ ]:

```
sepal length (cm)    float64
sepal width (cm)     float64
petal length (cm)    float64
petal width (cm)     float64
dtype: object
```

In [ ]:

```
data.describe()

## Std Dev- by how much the values of a column differ from the mean value for that column
```

In [ ]:

```
## average squared deviation of each column value from its mean
data.var()
```

In [ ]:

```
# Perfectly balanced data set
y.value_counts()
```

Out[ ]:

```
2    50
1    50
0    50
Name: target, dtype: int64
```

In [ ]:

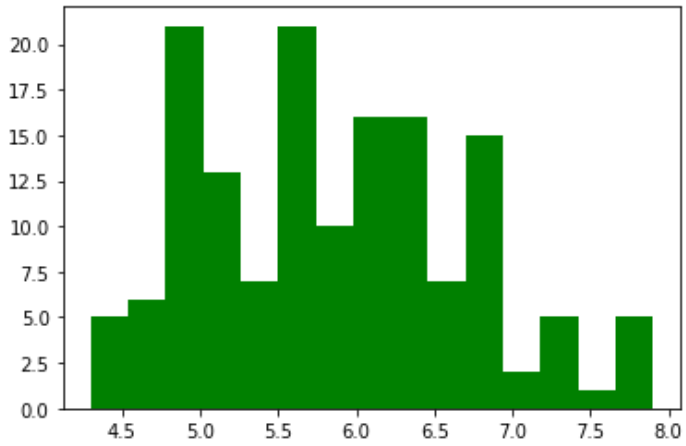
```
x.describe()
```

Out[ ]:

	sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)
count	150.000000	150.000000	150.000000	150.000000
mean	5.843333	3.057333	3.758000	1.199333
std	0.828066	0.435866	1.765298	0.762238
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

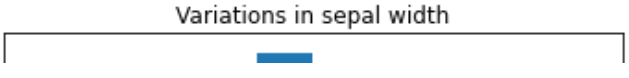
In [ ]:

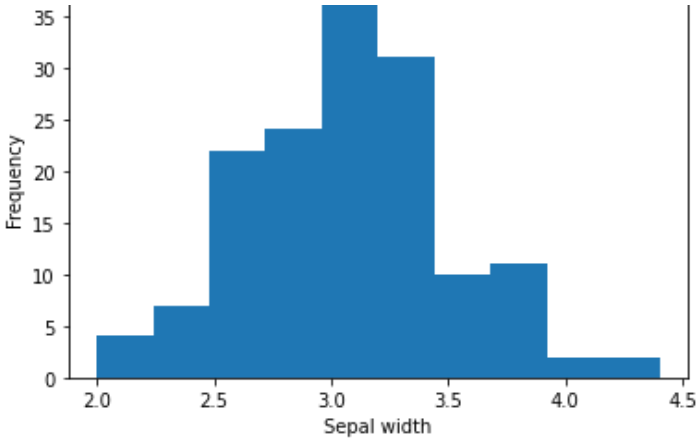
```
# ## Histograms groups the column values of dataset into bins
plt.hist(x['sepal length (cm)'],bins=15,color='green')
plt.show()
```



In [ ]:

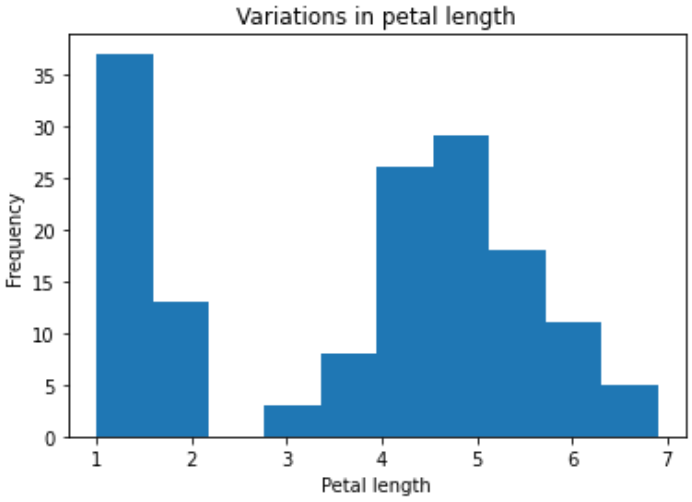
```
plt.hist(x['sepal width (cm)'])
plt.title('Variations in sepal width')
plt.xlabel('Sepal width')
plt.ylabel('Frequency')
plt.show()
plt.show()
```





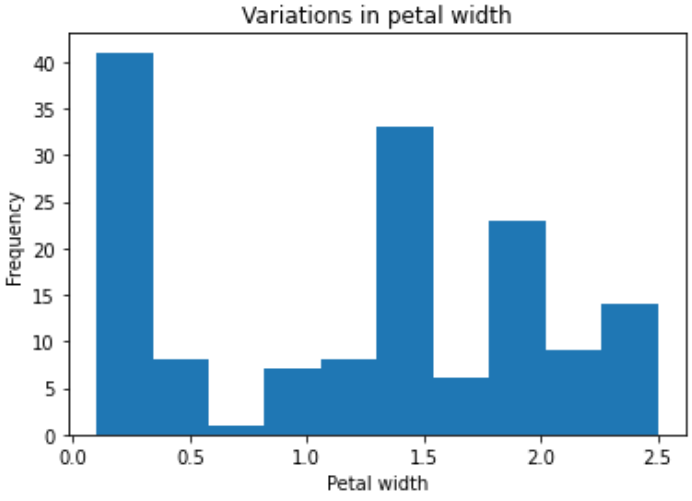
In [ ]:

```
plt.hist(x['petal length (cm)'])
plt.title('Variations in petal length')
plt.xlabel('Petal length')
plt.ylabel('Frequency')
plt.show()
```



In [ ]:

```
plt.hist(x['petal width (cm)'])
plt.title('Variations in petal width')
plt.xlabel('Petal width')
plt.ylabel('Frequency')
plt.show()
```

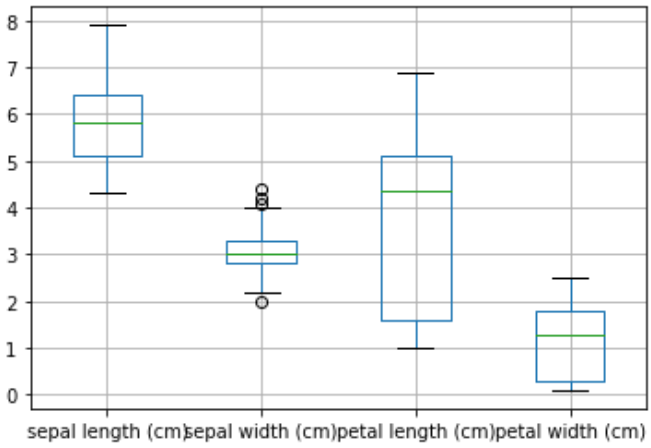


In [ ]:

```
## A boxplot is a standardized way of displaying the dataset based on a five-number summa
ry:
## the minimum, the maximum, the sample median, and the first quartile (25th percentile)
```

```
and
## third quartile (75th percentile).

x.boxplot()
plt.show()
```



In [ ]:

```
x.boxplot(['sepal width (cm)'])
```

Out[ ]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f9cf30da650>
```