In [146…
```
# Data frame selected from Kaggle.
# https://www.kaggle.com/datasets/uciml/iris
```

In [147…
```
# Step 1:
# Loading all the necessary libraries to process the data
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.cluster import KMeans
from sklearn.preprocessing import StandardScaler
from sklearn.preprocessing import LabelEncoder
```

In [148…
```
# Step 2:
# Load and explore data
df = pd.read_csv('iris.csv')

# Paso 2: Getting information about data set
print("Information obtained from the data set:")
print(df.info())
```

```
Information obtained from the data set:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 150 entries, 0 to 149
Data columns (total 6 columns):
 #   Column         Non-Null Count  Dtype  
---  ------         --------------  -----  
 0   Id             150 non-null    int64  
 1   SepalLengthCm  150 non-null    float64
 2   SepalWidthCm   150 non-null    float64
 3   PetalLengthCm  150 non-null    float64
 4   PetalWidthCm   150 non-null    float64
 5   Species        150 non-null    object 
dtypes: float64(4), int64(1), object(1)
memory usage: 7.2+ KB
None
```

In [149…
```
# Step 3:
# As part of exploring part, we show some record from the data set.
print("\nFirst records from data set:")
print(df.head())
```

```
First records from data set:
   Id  SepalLengthCm  SepalWidthCm  PetalLengthCm  PetalWidthCm      Species
0   1            5.1           3.5            1.4           0.2  Iris-setosa
1   2            4.9           3.0            1.4           0.2  Iris-setosa
2   3            4.7           3.2            1.3           0.2  Iris-setosa
3   4            4.6           3.1            1.5           0.2  Iris-setosa
4   5            5.0           3.6            1.4           0.2  Iris-setosa
```

In [150…
```
# Step 4:
# Getting descriptive statistics
print("\nDescriptive statistics:")
print(df.describe())
```

```
Descriptive statistics:
                Id  SepalLengthCm  SepalWidthCm  PetalLengthCm  PetalWidthCm
count   150.000000     150.000000    150.000000     150.000000    150.000000
mean     75.500000       5.843333      3.054000       3.758667      1.198667
std      43.445368       0.828066      0.433594       1.764420      0.763161
min       1.000000       4.300000      2.000000       1.000000      0.100000
25%      38.250000       5.100000      2.800000       1.600000      0.300000
50%      75.500000       5.800000      3.000000       4.350000      1.300000
75%     112.750000       6.400000      3.300000       5.100000      1.800000
max     150.000000       7.900000      4.400000       6.900000      2.500000
```

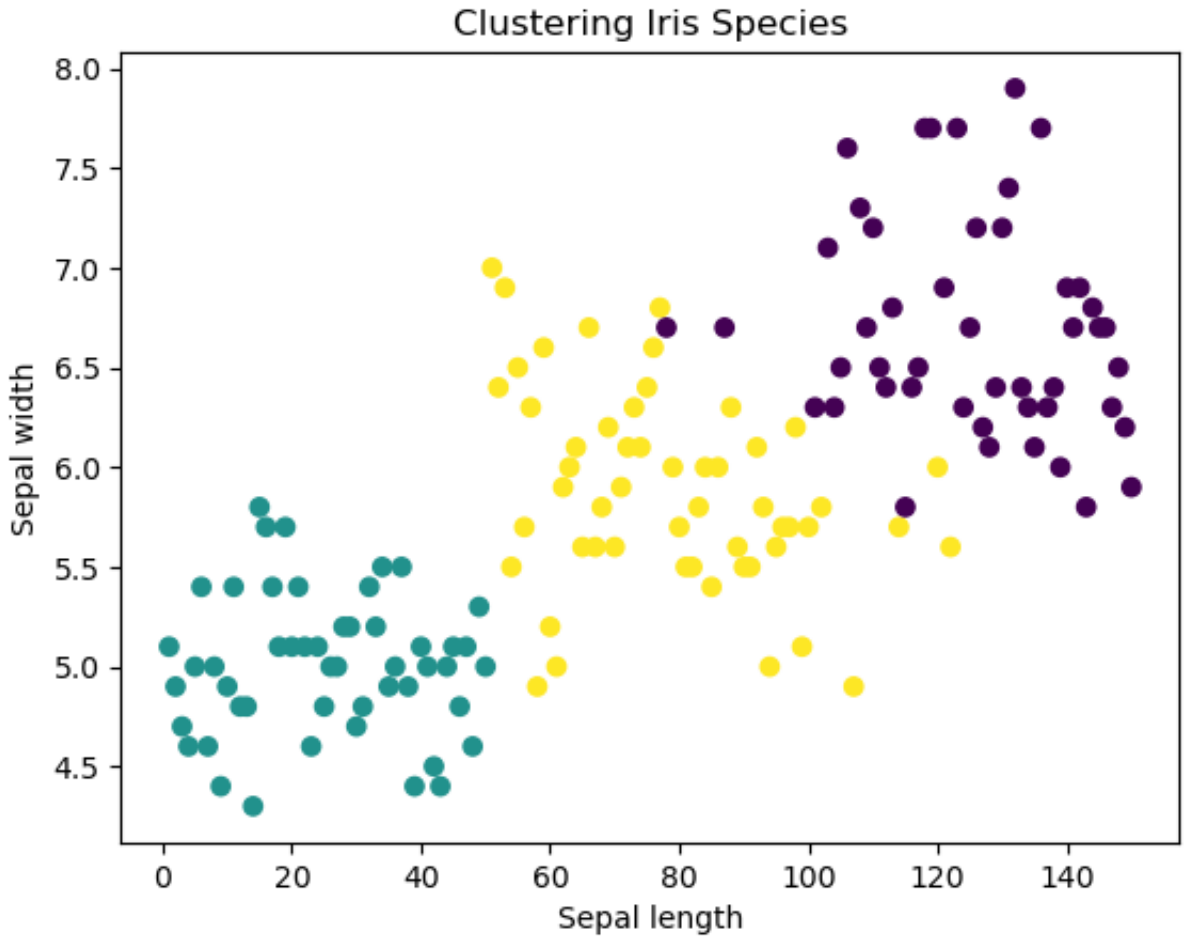
In [151…
```
# Step 5:
# Extract features from the dataset and store them in an array
X = df.iloc[:, :-1].values  # [:] Select all rows. [-1] Select all columns e
```

In [152…
```
# Step 6:
# Data preprocessing
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)
```

In [153…
```
# Step 7:
# Selecting a clustering algorithm
kmeans = KMeans(n_clusters=3, n_init=10, random_state=42)
```

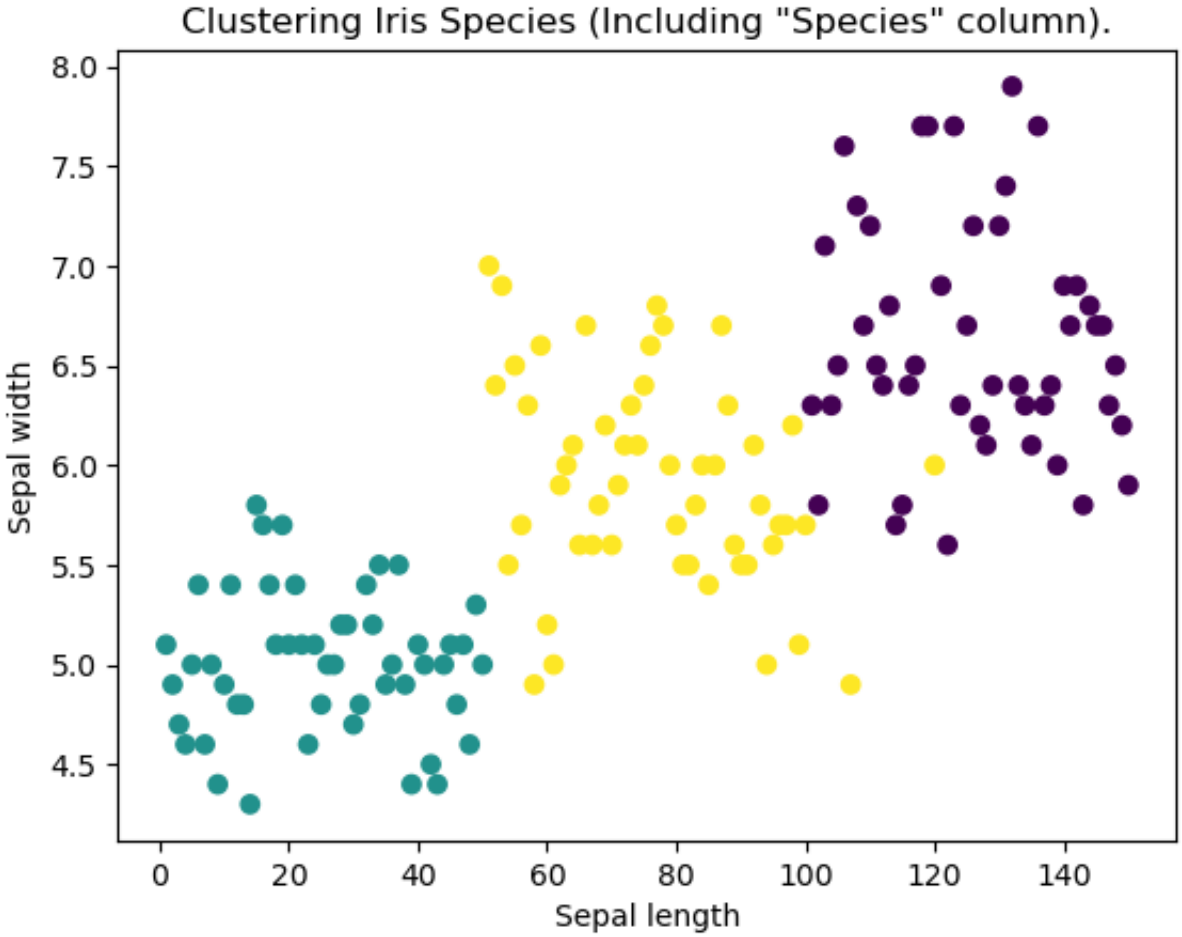
In [154…
```
# Step 8:
# Apply the clustering algorithm selected
labels = kmeans.fit_predict(X_scaled)
```

In [155…
```
# Step 9:
# Visualizing the results
plt.scatter(X[:, 0], X[:, 1], c=labels, cmap='viridis')
plt.xlabel('Sepal length')
plt.ylabel('Sepal width')
plt.title('Clustering Iris Species')
plt.show()
```



In [156…

```
# Step 11:
# What would happen if we entered a labeled data to the selected columns. Wo

# Create an instance of "LabelEncoder". LabelEncoder is used to assign a num
label_encoder = LabelEncoder()

# Convert species labels to numeric values
df['Species'] = label_encoder.fit_transform(df['Species'])

X = df.iloc[:, :].values

scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)

kmeans = KMeans(n_clusters=3, n_init=10, random_state=42)

labels = kmeans.fit_predict(X_scaled)

plt.scatter(X[:, 0], X[:, 1], c=labels, cmap='viridis')
plt.xlabel('Sepal length')
plt.ylabel('Sepal width')
plt.title('Clustering Iris Species (Including "Species" column).')
plt.show()
```


Clustering Iris Species (Including "Species" column).
Sepal width
Sepal length
8.0
7.5
7.0
6.5
6.0
5.5
5.0
4.5
0
20
40
60
80
100
120
140