```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np

df = pd.read_csv('/content/Iris.csv')

df
```

	5.1	3.5	1.4	0.2	Iris-setosa
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
...	...	...	...	...	...
144	6.7	3.0	5.2	2.3	Iris-virginica
145	6.3	2.5	5.0	1.9	Iris-virginica
146	6.5	3.0	5.2	2.0	Iris-virginica
147	6.2	3.4	5.4	2.3	Iris-virginica
148	5.9	3.0	5.1	1.8	Iris-virginica

149 rows × 5 columns

Next steps: Generate code with df | View recommended plots | New interactive sheet

Display basic info about the dataset

```
df.describe()
```

	5.1	3.5	1.4	0.2
count	149.000000	149.000000	149.000000	149.000000
mean	5.848322	3.051007	3.774497	1.205369
std	0.828594	0.433499	1.759651	0.761292
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

```
df.dtypes
```

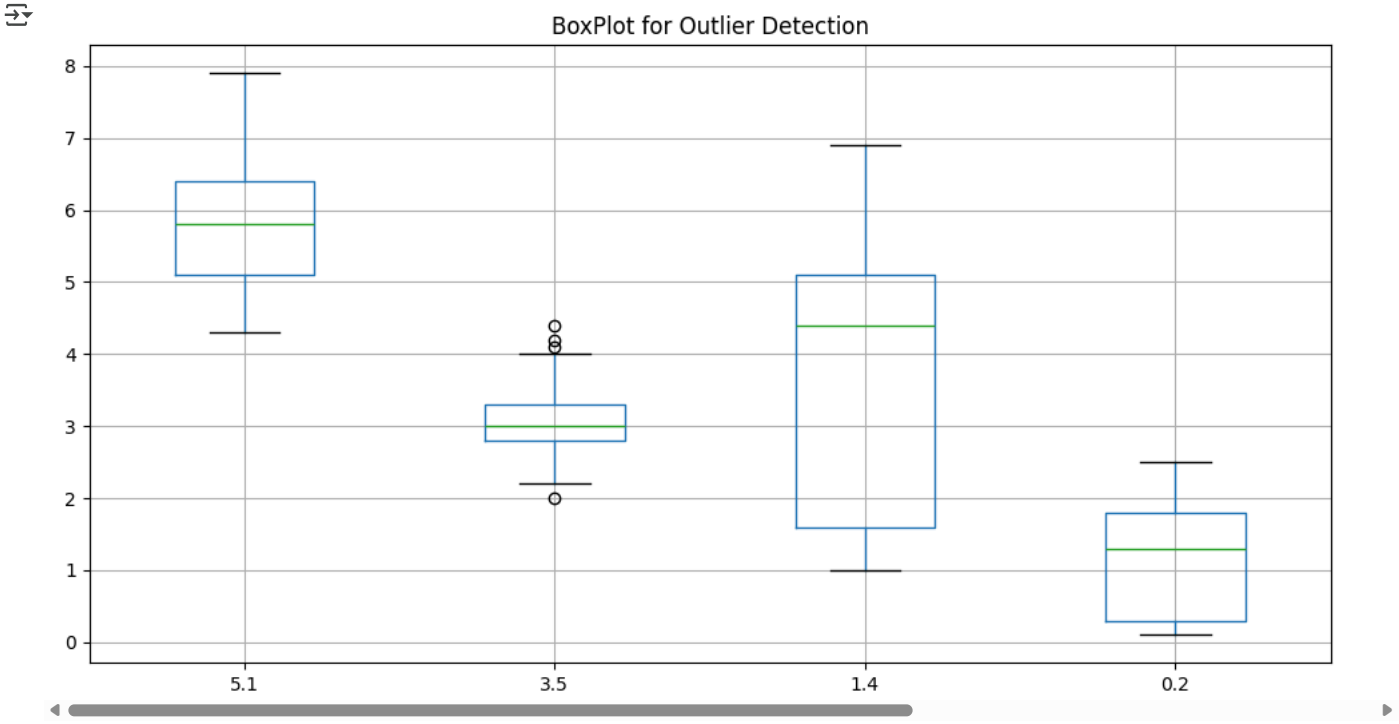
	0
5.1	float64
3.5	float64
1.4	float64
0.2	float64
Iris-setosa	object

dtype: object

2. Detect Outliers Using BoxPlot

```
# Select numerical columns for outlier detection
numeric_columns = df.select_dtypes(include=['float64', 'int64']).columns
```

```
# Plot BoxPlot for each numerical column
plt.figure(figsize=(12,6))
df[numeric_columns].boxplot()
plt.title("BoxPlot for Outlier Detection")
plt.show()
```



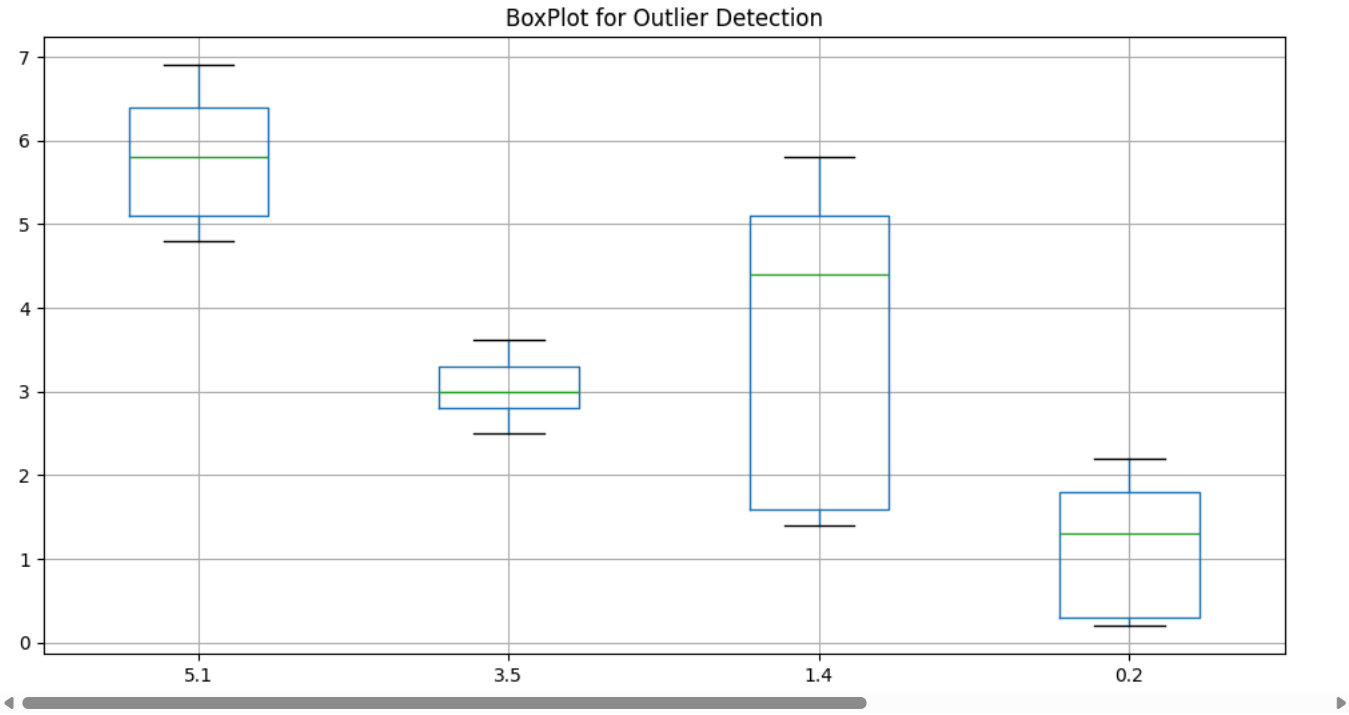
3. Handle Outliers Using Quantile-Based Flooring and Capping

```
# Define outlier capping function
def cap_outliers(df, column):
    Q1 = df[column].quantile(0.10)  # 10th percentile
    Q3 = df[column].quantile(0.90)  # 90th percentile
    df[column] = np.where(df[column] < Q1, Q1, df[column])  # Flooring
    df[column] = np.where(df[column] > Q3, Q3, df[column])  # Capping

# Apply outlier handling to numerical columns
for col in numeric_columns:
    cap_outliers(df, col)

print("\nUpdated Dataset After Handling Outliers:\n", df.describe())
```

```
Updated Dataset After Handling Outliers:
              5.1         3.5         1.4         0.2
count  149.000000  149.000000  149.000000  149.000000
mean     5.821477    3.040268    3.744295    1.193960
std      0.710497    0.348495    1.682999    0.732658
min      4.800000    2.500000    1.400000    0.200000
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      6.900000    3.620000    5.800000    2.200000
```

```
print("Data Set after removing outliers:")
plt.figure(figsize=(12,6))
df[numeric_columns].boxplot()
plt.title("BoxPlot for Outlier Detection")
plt.show()
```

```
Data Set after removing outliers:
```



Start coding or generate with AI.