Data Visualization III Download the Iris flower dataset or any other dataset into a DataFrame. (e.g., https://archive.ics.uci.edu/ml/datasets/Iris). Scan the dataset and give the inference as:

1. List down the features and their types (e.g., numeric, nominal) available in the dataset.
2. Create a histogram for each feature in the dataset to illustrate the feature distributions.
3. Create a boxplot for each feature in the dataset.
4. Compare distributions and identify outliers.

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

```
df = pd.read_csv('Iris.csv')
```

```
df.head()
```

	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

```
df.describe()
```

Saved successfully!

	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

```
df.info()
```

```
<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
```

```
df.columns
```

```
Index(['Id', 'SepalLengthCm', 'SepalWidthCm', 'PetalLengthCm', 'PetalWidthCm',
       'Species'],
      dtype='object')
```

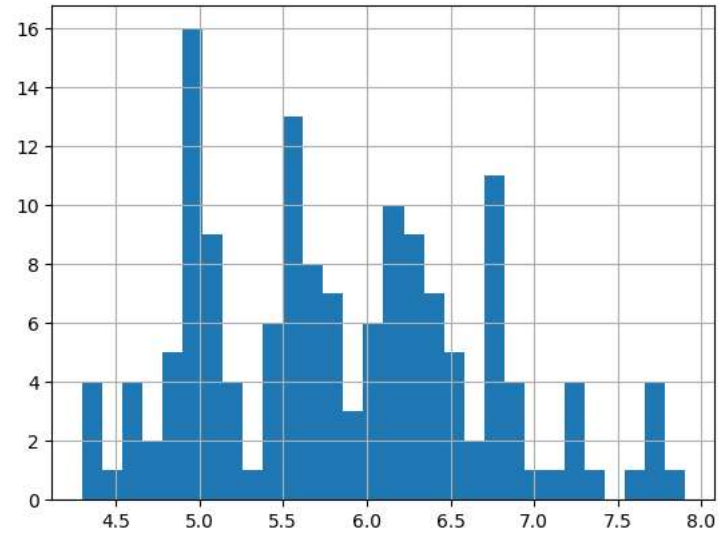
```
df['SepalLengthCm'].max()
```

```
df['SepalLengthCm'].min()
```

```
4.3
```

```
df['SepalLengthCm'].hist(bins = 30)
```

```
<Axes: >
```




```
df['PetalLengthCm'].max()
```

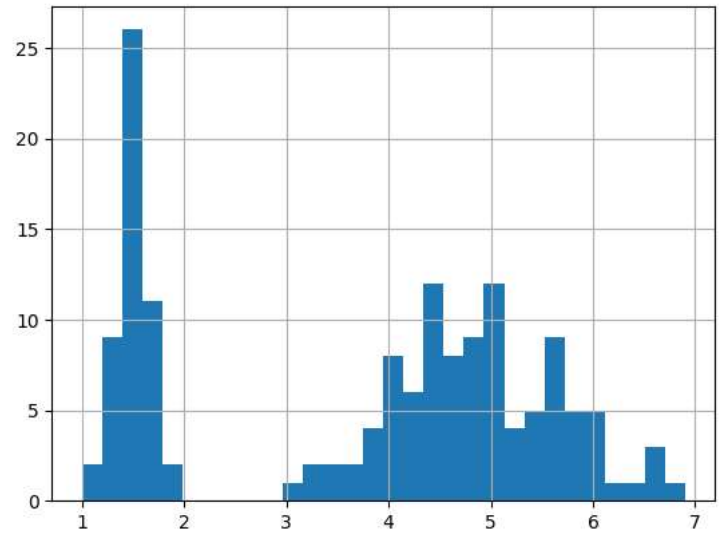
Saved successfully!

```
df['PetalLengthCm'].min()
```

```
1.0
```

```
df['PetalLengthCm'].hist(bins = 30)
```

```
<Axes: >
```




```
df['PetalWidthCm'].max()
```

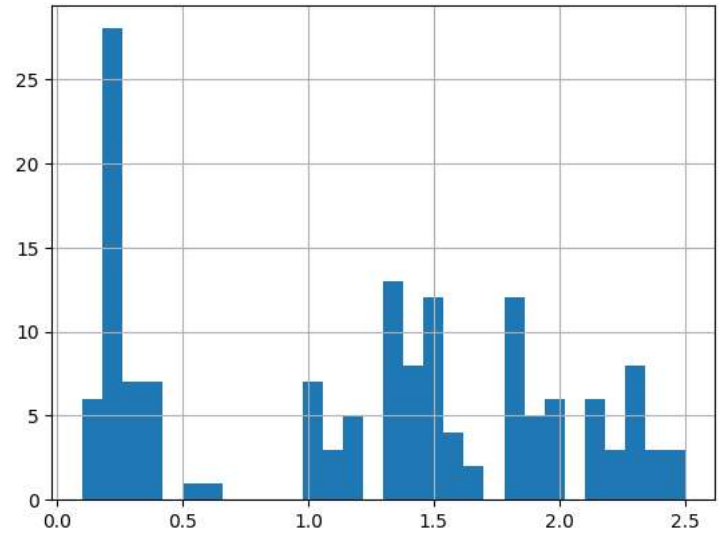
```
2.5
```

```
df['PetalWidthCm'].min()
```

```
0.1
```

```
df['PetalWidthCm'].hist(bins = 30)
```

```
<Axes: >
```




```
df['SepalWidthCm'].max()
```

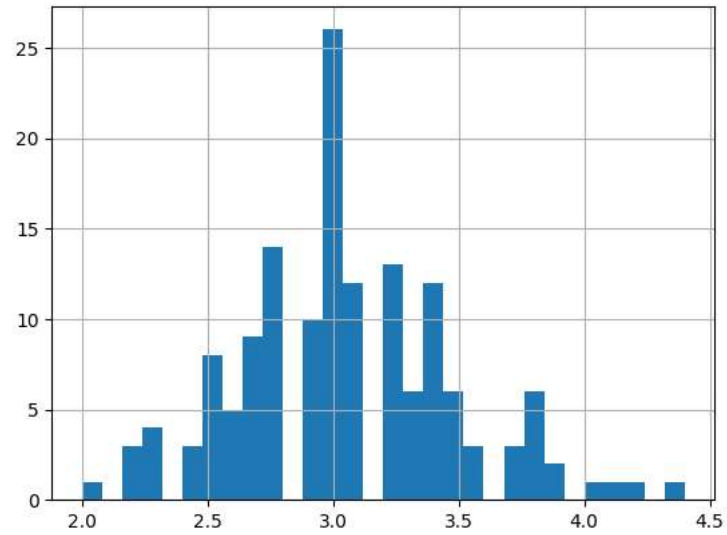
```
4.4
```

```
df['SepalWidthCm'].min()
```

```
2.0
```



```
<Axes: >
```

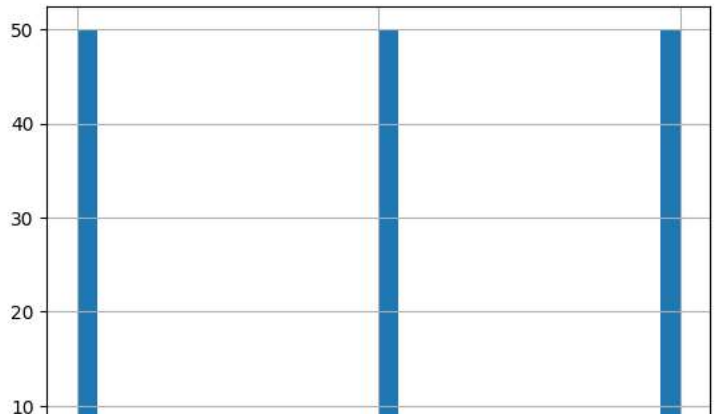



```
df['Species'].value_counts()
```

```
Iris-setosa        50
Iris-versicolor    50
Iris-virginica     50
Name: Species, dtype: int64
```

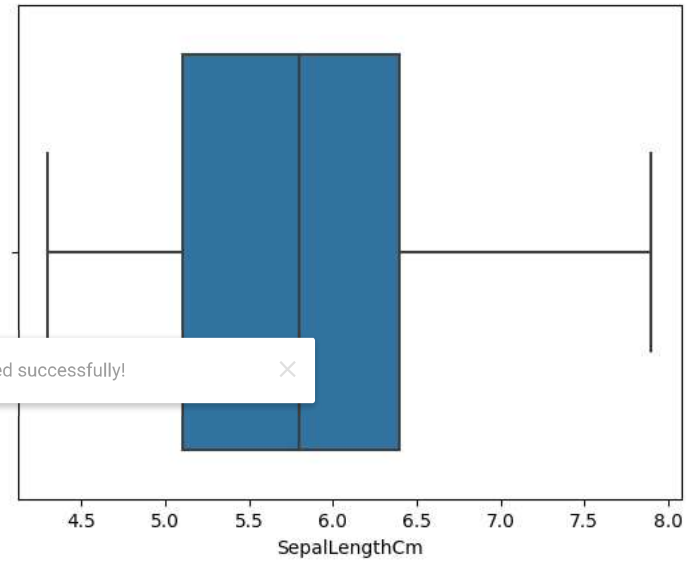
```
df['Species'].hist(bins = 30)
```

```
<Axes: >
```



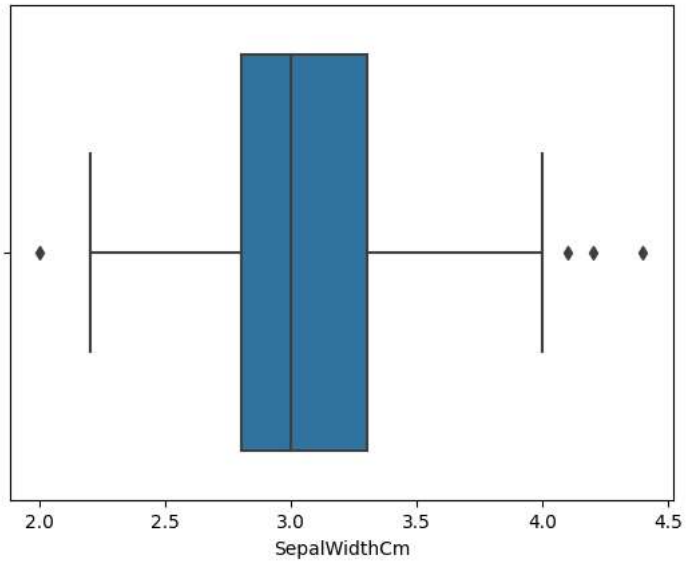

```
sns.boxplot(x = 'SepalLengthCm' ,data = df)
```

```
<Axes: xlabel='SepalLengthCm'>
```



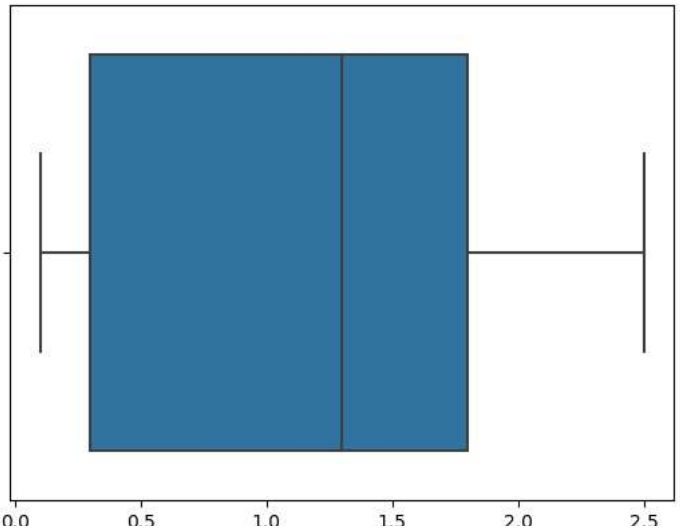

```
sns.boxplot(x = 'SepalWidthCm' , data = df)
```

```
<Axes: xlabel='SepalWidthCm'>
```



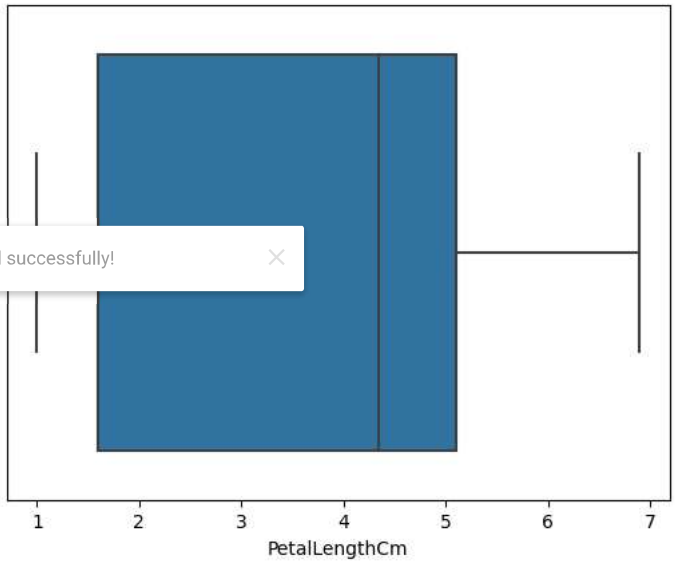

```
sns.boxplot(x = 'PetalWidthCm' , data = df)
```

```
<Axes: xlabel='PetalWidthCm'>
```




```
sns.boxplot(x = 'PetalLengthCm', data = df)
```

```
<Axes: xlabel='PetalLengthCm'>
```



Saved successfully!