

```

data = """
SL,SW,PL,PW,Classification
5.1,3.5,1.4,0.2,Iris-setosa
4.9,3,1.4,0.2,Iris-setosa
4.7,3.2,1.3,0.2,Iris-setosa
4.6,3.1,1.5,0.2,Iris-setosa
5,3.6,1.4,0.2,Iris-setosa
5.4,3.9,1.7,0.4,Iris-setosa
4.6,3.4,1.4,0.3,Iris-setosa
5,3.4,1.5,0.2,Iris-setosa
4.4,2.9,1.4,0.2,Iris-setosa
4.9,3.1,1.5,0.1,Iris-setosa
...
"""

import pandas as pd
from io import StringIO
df = pd.read_csv(StringIO(data))
print("Columns in the dataset:")
print(df.columns)

```

```

Columns in the dataset:
Index(['SL', 'SW', 'PL', 'PW', 'Classification'], dtype='object')

```

```
#Calculate mean of each data set
```

```

mean_values = df.mean(numeric_only=True)
print("\nMean of each column:")
print(mean_values)

```

```

Mean of each column:
SW    3.31
PL     1.45
PW     0.22
dtype: float64

```

```
#Check for the null values present in the dataset.
```

```

print("\nChecking for null values:")
print(df.isnull().sum())

```

```

Checking for null values:
SL          0
SW          1
PL          1
PW          1
Classification  1
dtype: int64

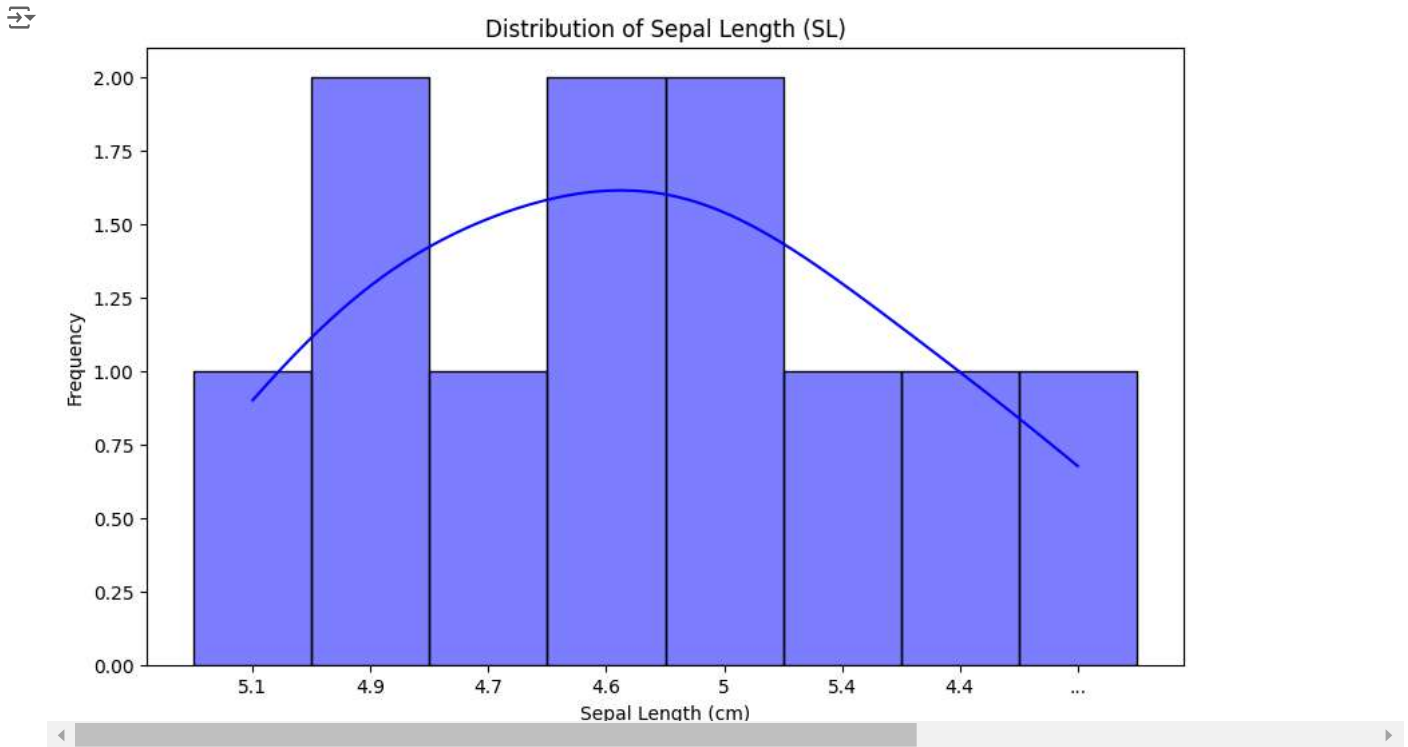
```

```
#Perform meaningful visualizations using the dataset. Bring at least 3 meaningful visualizations.
```

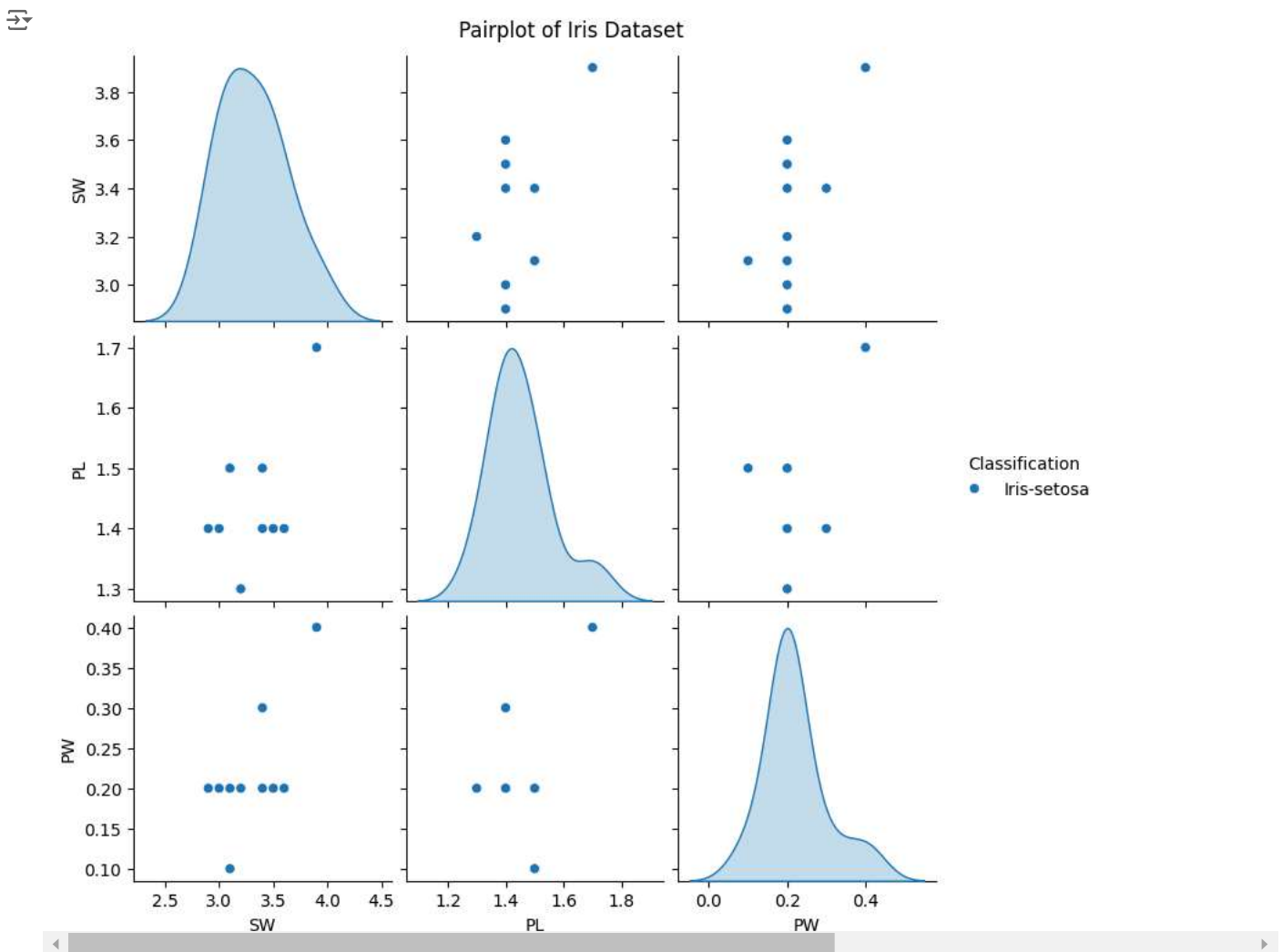
```

import matplotlib.pyplot as plt
import seaborn as sns
plt.figure(figsize=(10, 6))
sns.histplot(df['SL'], kde=True, color='blue', bins=20)
plt.title('Distribution of Sepal Length (SL)')
plt.xlabel('Sepal Length (cm)')
plt.ylabel('Frequency')
plt.show()


```



```
sns.pairplot(df, hue='Classification')
plt.suptitle('Pairplot of Iris Dataset', y=1.02)
plt.show()
```



```
plt.figure(figsize=(10, 6))
sns.boxplot(x='Classification', y='SL', data=df, palette='Set1')
plt.title('Boxplot of Sepal Length across Species')
plt.xlabel('Species')
plt.ylabel('Sepal Length (cm)')
plt.show()
```

 <ipython-input-12-329ebffa6f59>:2: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `x` variable to `hue` and set `legend` to `True` to add a legend to the plot.

```
sns.boxplot(x='Classification', y='SL', data=df, palette='Set1')
```

