In [9]:
```python
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

# Load the Iris dataset
iris = sns.load_dataset('iris')
```

In [10]:
```python
# -------------------------------
# 1. List features and their types
# -------------------------------
print("1. Features and Their Types:\n")
print(iris.dtypes)
print("\nFeature Summary:")
for col in iris.columns:
    dtype = iris[col].dtype
    if dtype == 'float64':
        print(f"- {col}: Numeric")
    else:
        print(f"- {col}: Nominal")
```

```
1. Features and Their Types:

sepal_length    float64
sepal_width     float64
petal_length    float64
petal_width     float64
species          object
dtype: object

Feature Summary:
- sepal_length: Numeric
- sepal_width: Numeric
- petal_length: Numeric
- petal_width: Numeric
- species: Nominal
```
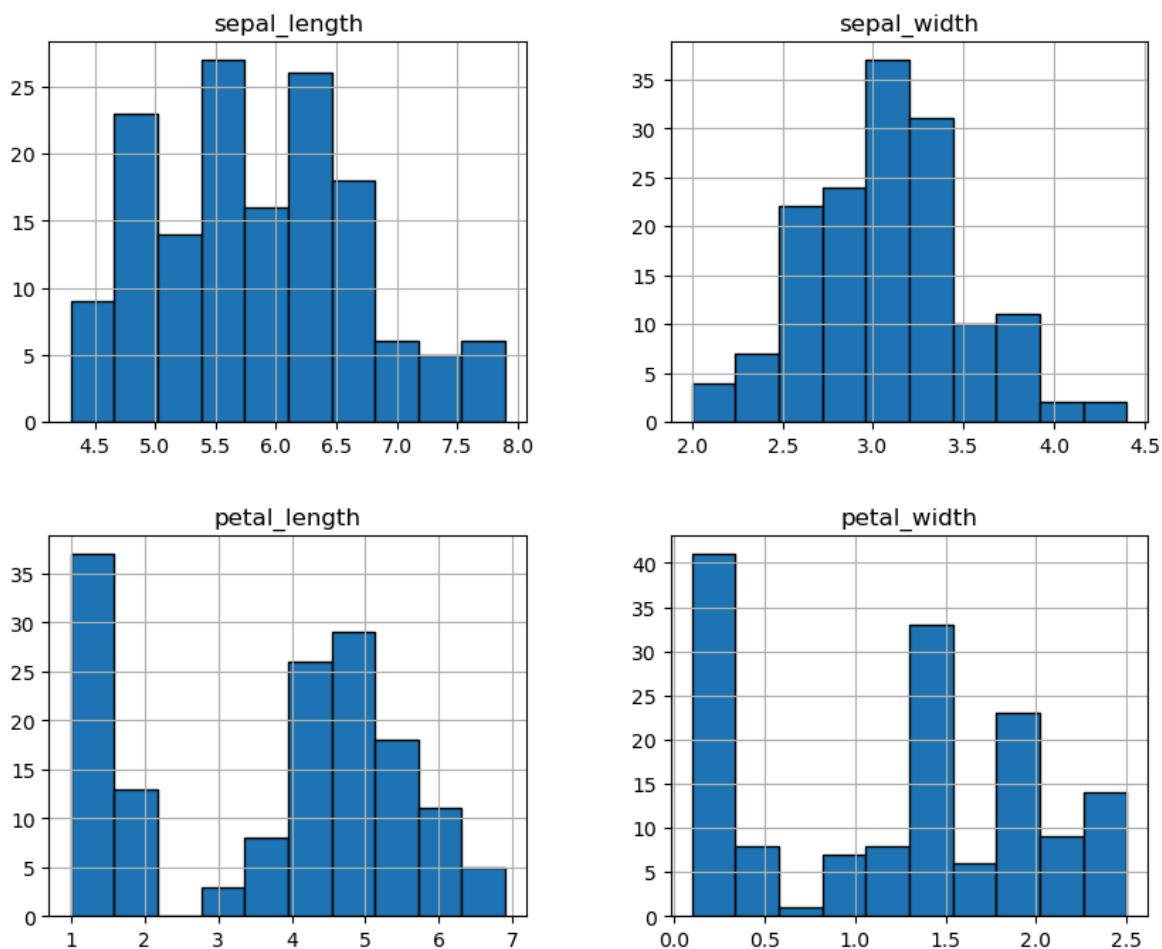
In [11]:
```python
# -------------------------------
# 2. Histograms for each feature
# -------------------------------
print("\n2. Plotting Histograms...")
iris.hist(figsize=(10, 8), edgecolor='black')
plt.suptitle("Histograms for Each Numeric Feature")
plt.show()
```

```
2. Plotting Histograms...
```

Histograms for Each Numeric Feature


```
In [12]:   # --------------------------------
           # 3. Boxplots for each feature
           # --------------------------------
           print("3. Plotting Boxplots...")
           plt.figure(figsize=(12, 8))
           for i, column in enumerate(iris.columns[:-1], 1):
               plt.subplot(2, 2, i)
               sns.boxplot(data=iris, x='species', y=column)
               plt.title(f'Boxplot of {column}')
           plt.tight_layout()
           plt.show()
```

3. Plotting Boxplots...

Boxplot of sepal_length

Boxplot of sepal_width

Boxplot of petal_length

Boxplot of petal_width
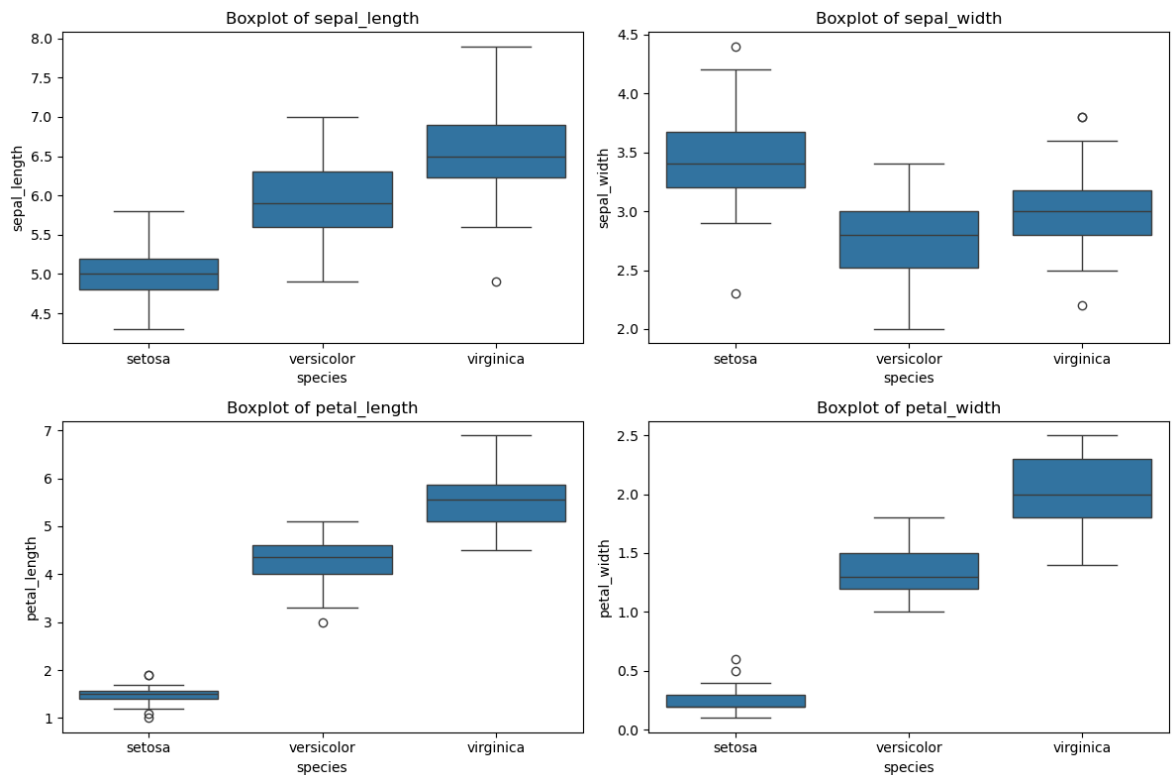
```
In [13]:    # -------------------------------
            # 4. Compare distributions & outliers
            # -------------------------------
            print("4. Distribution Comparison & Outliers:")
            for column in iris.columns[:-1]:
                desc = iris[column].describe()
                print(f"\nFeature: {column}")
                print(desc)

                q1 = iris[column].quantile(0.25)
                q3 = iris[column].quantile(0.75)
                iqr = q3 - q1
                lower_bound = q1 - 1.5 * iqr
                upper_bound = q3 + 1.5 * iqr

                outliers = iris[(iris[column] < lower_bound) | (iris[column] > upper_bound)]
                print(f"Potential outliers in {column}: {len(outliers)}")
                if not outliers.empty:
                    print(outliers[['species', column]])
                else:
                    print("No outliers found.")
```

4. Distribution Comparison & Outliers:

Feature: sepal_length
count   150.000000
mean   5.843333
std   0.828066
min   4.300000
25%   5.100000
50%   5.800000
75%   6.400000
max   7.900000
Name: sepal_length, dtype: float64
Potential outliers in sepal_length: 0
No outliers found.

Feature: sepal_width
count   150.000000
mean   3.057333
std   0.435866
min   2.000000
25%   2.800000
50%   3.000000
75%   3.300000
max   4.400000
Name: sepal_width, dtype: float64
Potential outliers in sepal_width: 4
        species sepal_width
15   setosa   4.4
32   setosa   4.1
33   setosa   4.2
60   versicolor   2.0

Feature: petal_length
count   150.000000
mean   3.758000
std   1.765298
min   1.000000
25%   1.600000
50%   4.350000
75%   5.100000
max   6.900000
Name: petal_length, dtype: float64
Potential outliers in petal_length: 0
No outliers found.

Feature: petal_width
count   150.000000
mean   1.199333
std   0.762238
min   0.100000
25%   0.300000
50%   1.300000
75%   1.800000
max   2.500000
Name: petal_width, dtype: float64
Potential outliers in petal_width: 0
No outliers found.

In [ ]:

In [ ]: