In [9]:
```
import pandas as pd
data = pd.read_csv('c.csv')
data
```

Out[9]:

	Rating	Comment
0	5.0	Good
1	4.0	Awesome
2	3.0	Good
3	NaN	NaN
4	5.0	Tasty
5	4.0	Amazing
6	NaN	NaN
7	3.0	Best coffee
8	2.0	Need to improve
9	4.0	Superb

In [10]:
```
print(data.info())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10 entries, 0 to 9
Data columns (total 2 columns):
 #   Column   Non-Null Count  Dtype  
---  ------   --------------  -----  
 0   Rating   8 non-null      float64
 1   Comment  8 non-null      object 
dtypes: float64(1), object(1)
memory usage: 292.0+ bytes
None
```

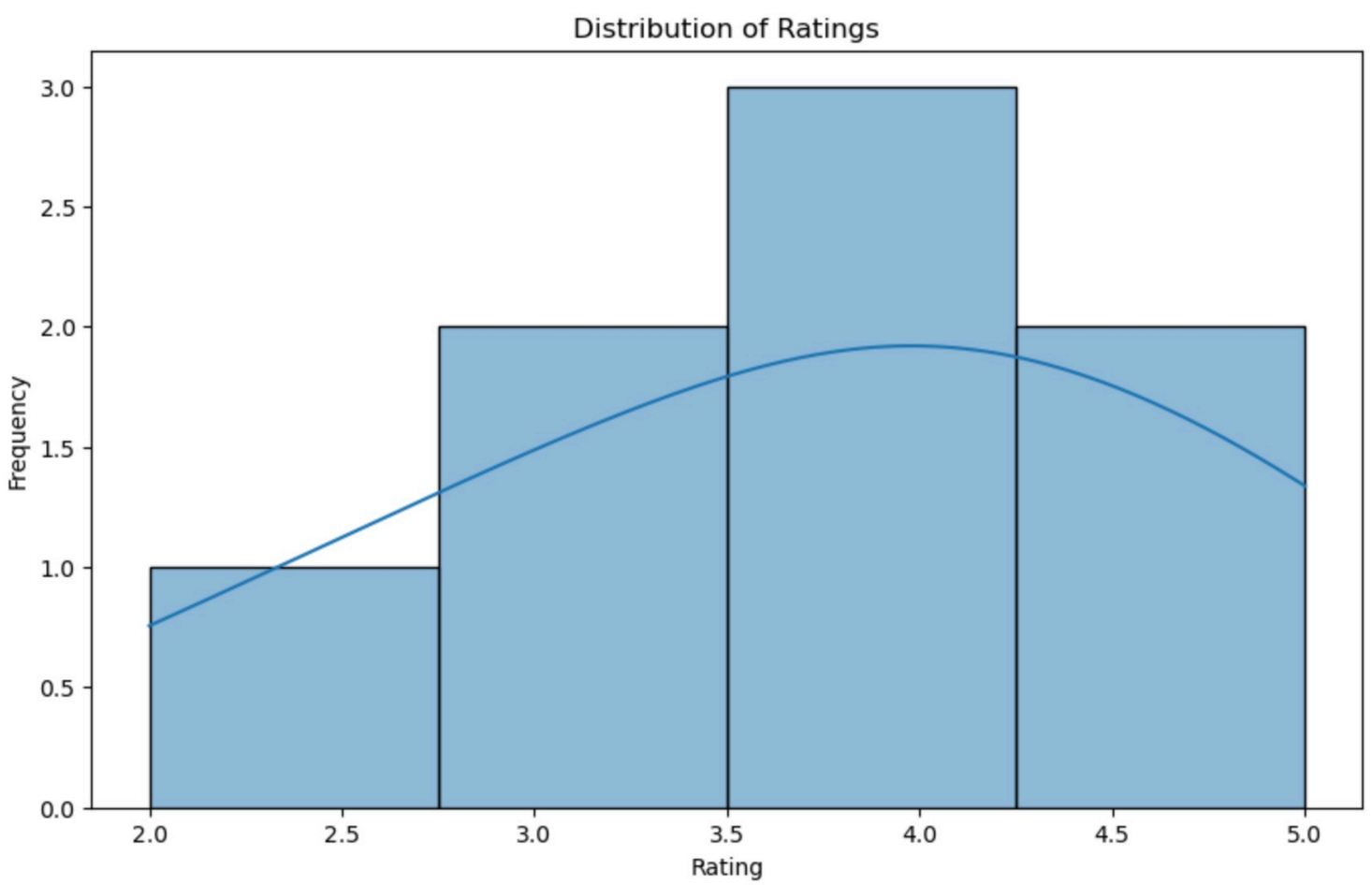
In [11]:
```
print(data.isnull().sum())
```

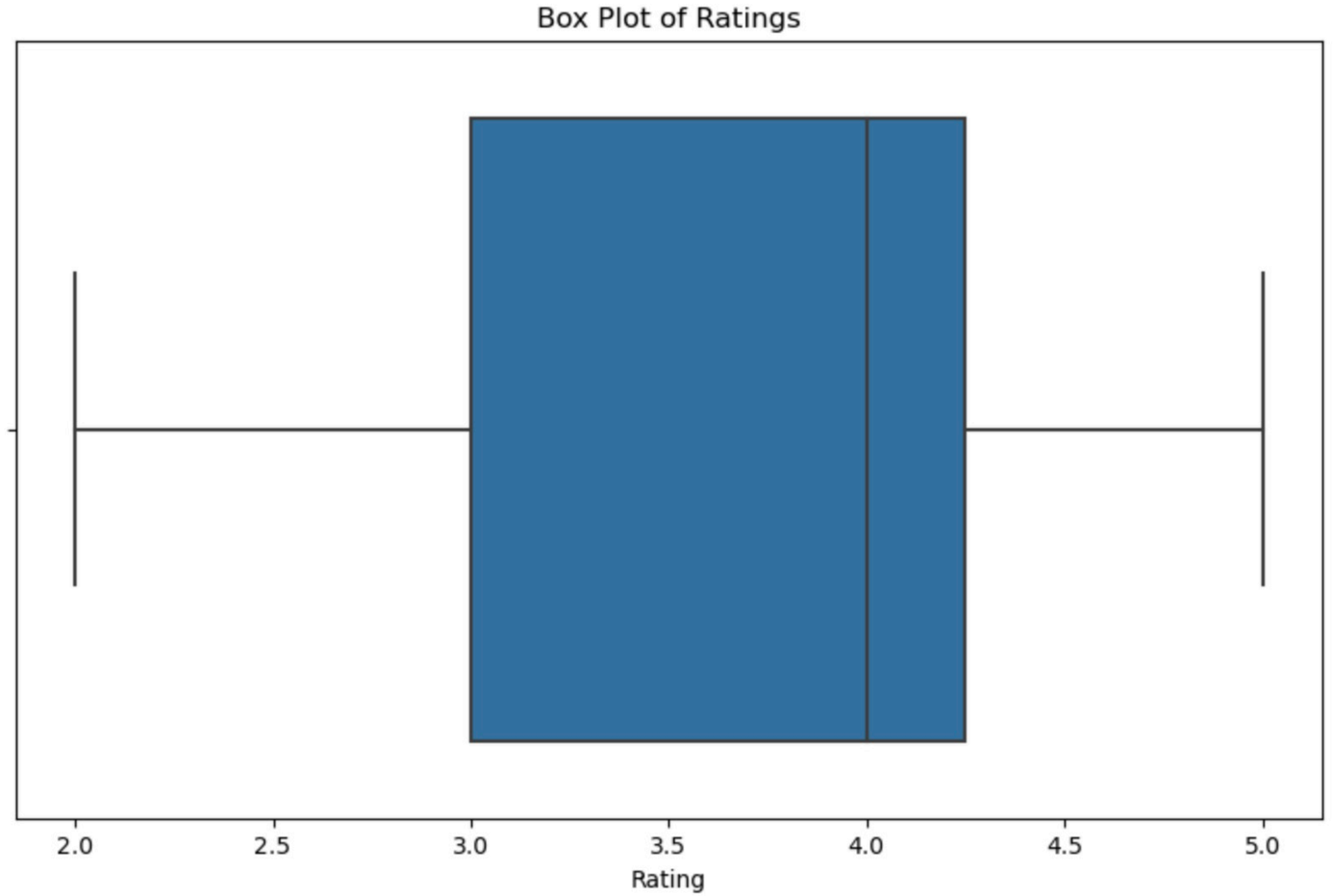
```
Rating     2
Comment    2
dtype: int64
```

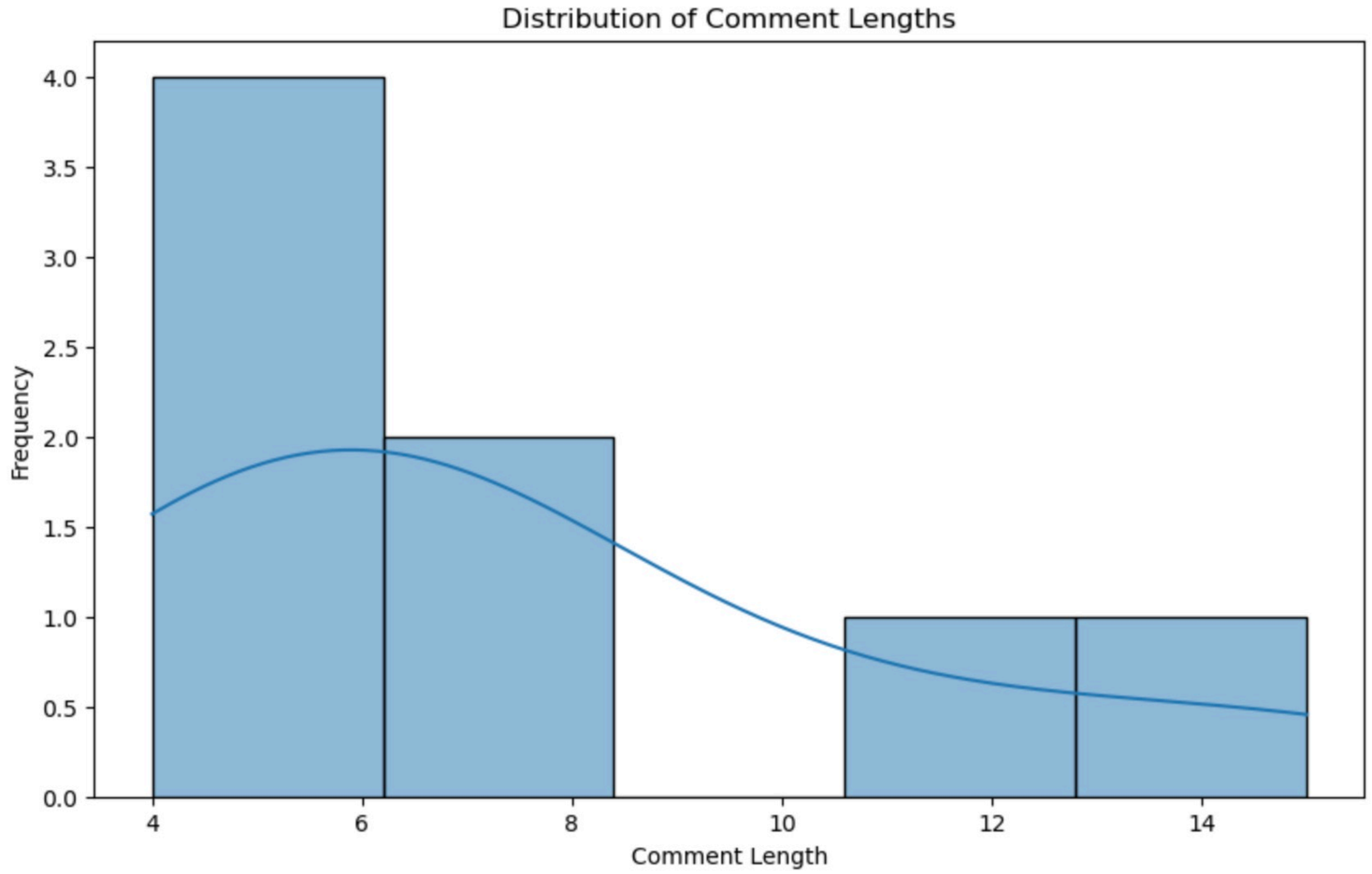
In [12]:
```
print(data.isnull().sum())
data = data.dropna()
```

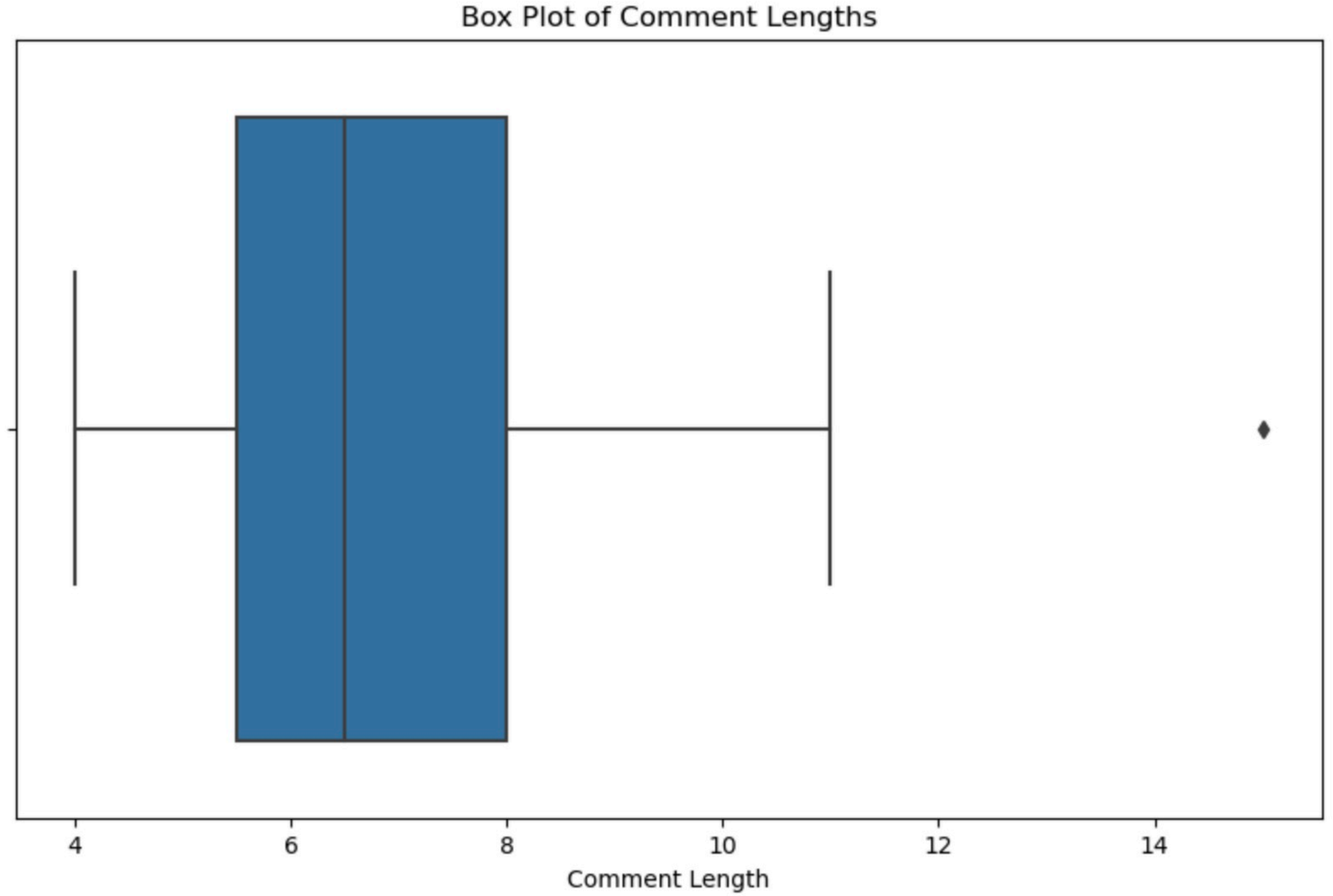
```
Rating     2
Comment    2
dtype: int64
```

In [17]:
```
import seaborn as sns
import matplotlib.pyplot as plt
plt.figure(figsize=(10, 6))
sns.histplot(data['Rating'], kde=True)
plt.title('Distribution of Ratings')
plt.xlabel('Rating')
plt.ylabel('Frequency')
plt.show()
plt.figure(figsize=(10, 6))
sns.boxplot(x=data['Rating'])
plt.title('Box Plot of Ratings')
plt.xlabel('Rating')
plt.show()
data['comment_length'] = data['Comment'].apply(len)
plt.figure(figsize=(10, 6))
sns.histplot(data['comment_length'], kde=True)
plt.title('Distribution of Comment Lengths')
plt.xlabel('Comment Length')
plt.ylabel('Frequency')
plt.show()
plt.figure(figsize=(10, 6))
sns.boxplot(x=data['comment_length'])
plt.title('Box Plot of Comment Lengths')
plt.xlabel('Comment Length')
plt.show()
```

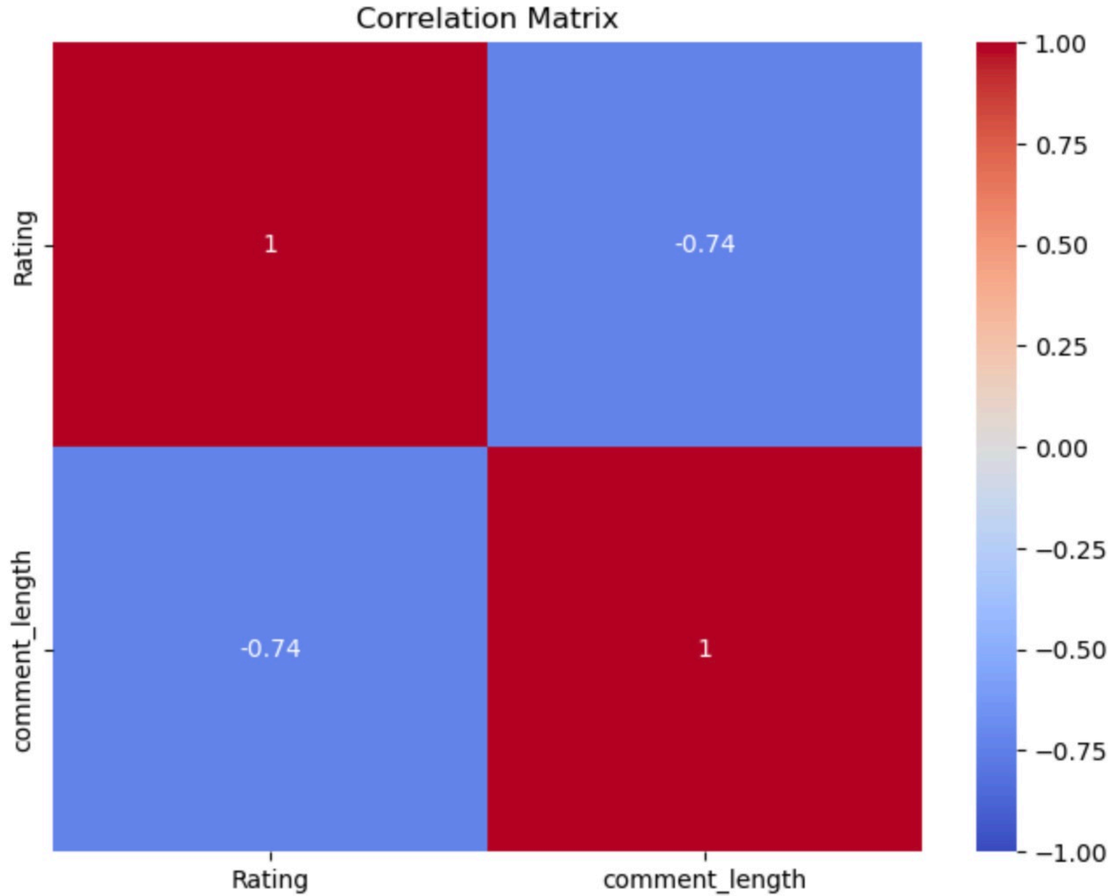

Distribution of Ratings
Frequency
Rating
0.0
0.5
1.0
1.5
2.0
2.5
3.0
2.0
2.5
3.0
3.5
4.0
4.5
5.0


Box Plot of Ratings
2.0
2.5
3.0
3.5
4.0
4.5
5.0
Rating


Distribution of Comment Lengths
Frequency
Comment Length
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4
6
8
10
12
14


Box Plot of Comment Lengths
4
6
8
10
12
14
Comment Length

In [18]:
```
correlation_matrix = data[['Rating', 'comment_length']].corr()
plt.figure(figsize=(8, 6))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', vmin=-1, vmax=1)
plt.title('Correlation Matrix')
plt.show()
```



In [19]:
```
plt.figure(figsize=(10, 6))
sns.scatterplot(x='Rating', y='comment_length', data=data)
plt.title('Scatter Plot of Rating vs. Comment Length')
plt.xlabel('Rating')
plt.ylabel('Comment Length')
plt.show()
```