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

```
# Load the Titanic dataset
titanic = sns.load_dataset("titanic")

# Display first 5 rows
titanic.head()
```

	survived	pclass	sex	age	sibsp	parch	fare	embarked	class	who	adult_male	deck	embark_town	alive	alone
0	0	3	male	22.0	1	0	7.2500	S	Third	man	True	NaN	Southampton	no	False
1	1	1	female	38.0	1	0	71.2833	C	First	woman	False	C	Cherbourg	yes	False
2	1	3	female	26.0	0	0	7.9250	S	Third	woman	False	NaN	Southampton	yes	True
3	1	1	female	35.0	1	0	53.1000	S	First	woman	False	C	Southampton	yes	False
4	0	3	male	35.0	0	0	8.0500	S	Third	man	True	NaN	Southampton	no	True

Next steps: Generate code with `titanic` View recommended plots New interactive sheet

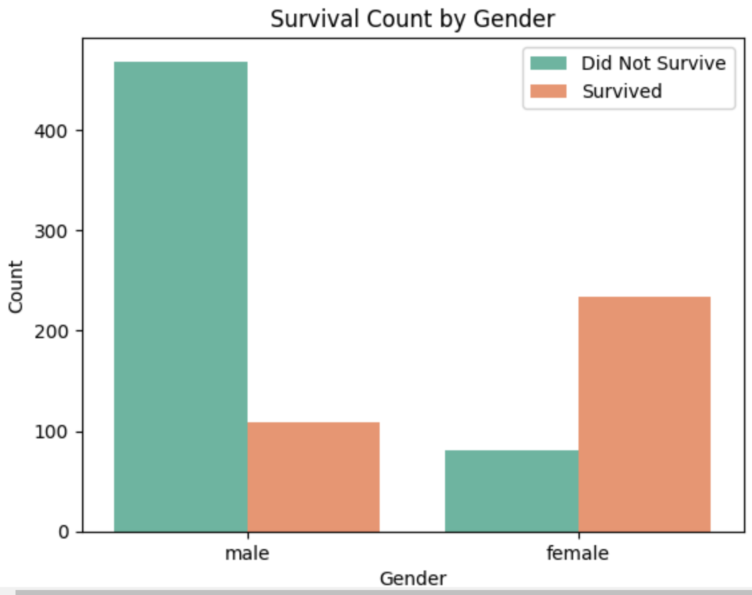
```
# Check basic details
titanic.info()

# Check for missing values
titanic.isnull().sum()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 15 columns):
 #   Column       Non-Null Count  Dtype   
---  ------       --------------  -----   
 0   survived     891 non-null    int64   
 1   pclass       891 non-null    int64   
 2   sex          891 non-null    object  
 3   age          714 non-null    float64 
 4   sibsp        891 non-null    int64   
 5   parch        891 non-null    int64   
 6   fare         891 non-null    float64 
 7   embarked     889 non-null    object  
 8   class        891 non-null    category
 9   who          891 non-null    object  
 10  adult_male   891 non-null    bool    
 11  deck         203 non-null    category
 12  embark_town  889 non-null    object  
 13  alive        891 non-null    object  
 14  alone        891 non-null    bool    
dtypes: bool(2), category(2), float64(2), int64(4), object(5)
memory usage: 80.7+ KB
```

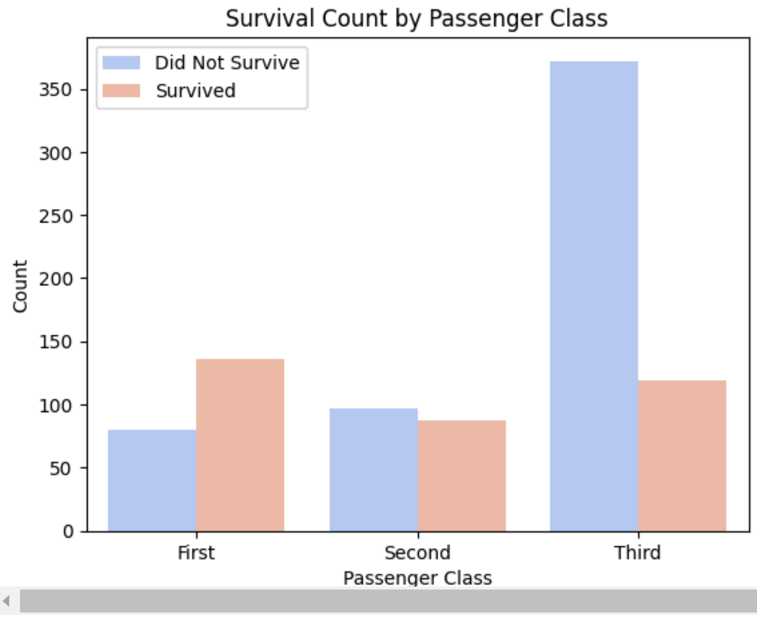
	0
survived	0
pclass	0
sex	0
age	177
sibsp	0
parch	0
fare	0
embarked	2
class	0
who	0
adult_male	0
deck	688
embark_town	2
alive	0
alone	0

dtype: int64

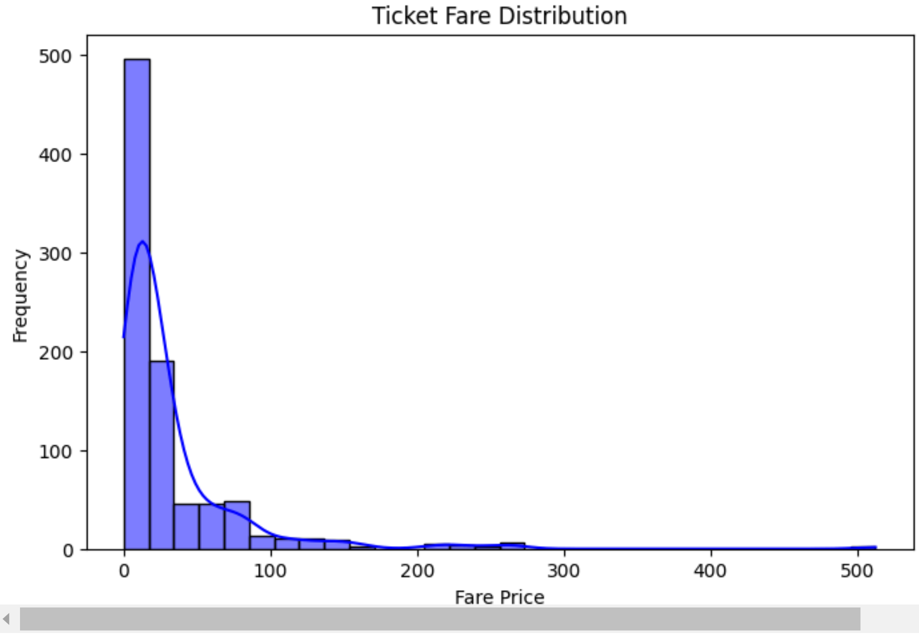
```
# Create a count plot for survival based on gender
sns.countplot(data=titanic, x="sex", hue="survived", palette="Set2")

# Add title
plt.title("Survival Count by Gender")
plt.xlabel("Gender")
plt.ylabel("Count")
plt.legend(["Did Not Survive", "Survived"])

# Show plot
plt.show()
```



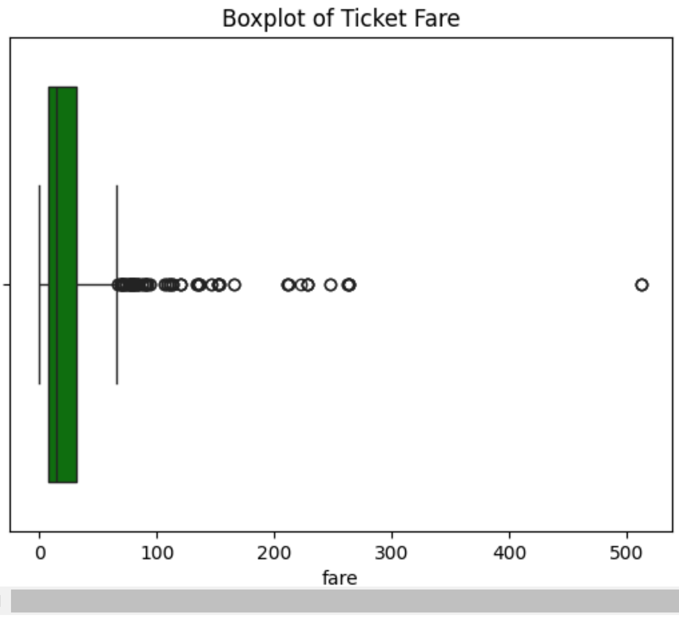
```
# Create a count plot for passenger class
sns.countplot(data=titanic, x="class", hue="survived", palette="coolwarm")

# Add title
plt.title("Survival Count by Passenger Class")
plt.xlabel("Passenger Class")
plt.ylabel("Count")
plt.legend(["Did Not Survive", "Survived"])

# Show plot
plt.show()
```



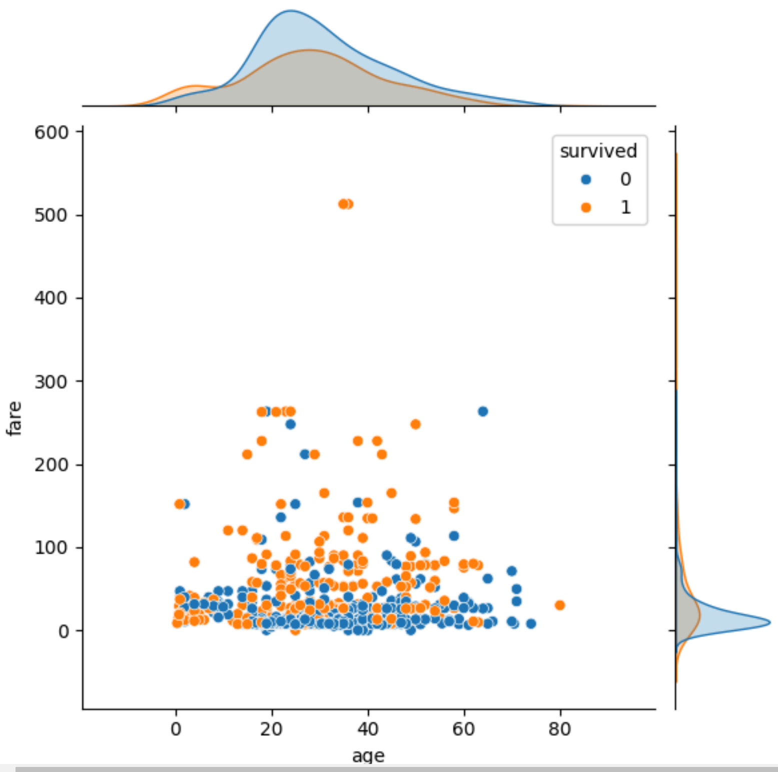
```
# Set figure size
plt.figure(figsize=(8, 5))

# Create histogram using Seaborn
sns.histplot(titanic["fare"], bins=30, kde=True, color="blue")

# Add title and labels
plt.title("Ticket Fare Distribution")
plt.xlabel("Fare Price")
plt.ylabel("Frequency")

# Show plot
plt.show()
```



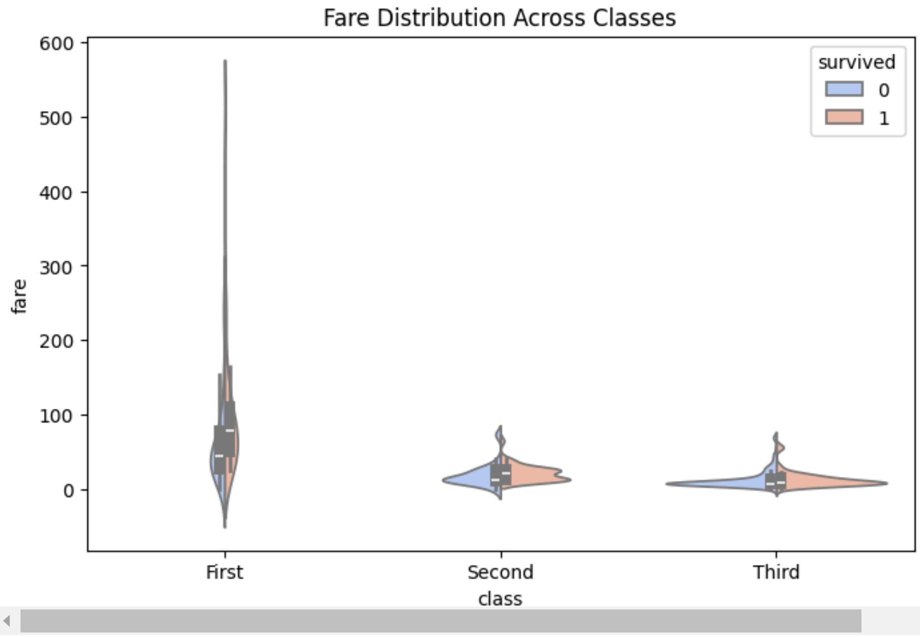
```
# Create a boxplot for fare prices
sns.boxplot(data=titanic, x="fare", color="green")

# Add title
plt.title("Boxplot of Ticket Fare")

# Show plot
plt.show()
```



```
# Create a joint plot
sns.jointplot(data=titanic, x="age", y="fare", kind="scatter", hue="survived")

# Show plot
plt.show()
```



```
# Create a violin plot
plt.figure(figsize=(8, 5))
sns.violinplot(data=titanic, x="class", y="fare", hue="survived", split=True, palette="coolwarm")

# Add title
plt.title("Fare Distribution Across Classes")

# Show plot
plt.show()
```


Fare Distribution Across Classes

```
# Create a swarm plot
plt.figure(figsize=(8, 5))
sns.swarmplot(data=titanic, x="class", y="fare", hue="survived", palette="Set2")

# Add title
plt.title("Swarm Plot of Fare Distribution by Class")

# Show plot
plt.show()
```

```
/usr/local/lib/python3.11/dist-packages/seaborn/categorical.py:3399: UserWarning: 12.5% of the points cannot be placed; you may want
  warnings.warn(msg, UserWarning)
/usr/local/lib/python3.11/dist-packages/seaborn/categorical.py:3399: UserWarning: 48.9% of the points cannot be placed; you may want
  warnings.warn(msg, UserWarning)
/usr/local/lib/python3.11/dist-packages/seaborn/categorical.py:3399: UserWarning: 74.3% of the points cannot be placed; you may want
  warnings.warn(msg, UserWarning)
/usr/local/lib/python3.11/dist-packages/seaborn/categorical.py:3399: UserWarning: 23.6% of the points cannot be placed; you may want
  warnings.warn(msg, UserWarning)
/usr/local/lib/python3.11/dist-packages/seaborn/categorical.py:3399: UserWarning: 58.2% of the points cannot be placed; you may want
  warnings.warn(msg, UserWarning)
/usr/local/lib/python3.11/dist-packages/seaborn/categorical.py:3399: UserWarning: 79.4% of the points cannot be placed; you may want
  warnings.warn(msg, UserWarning)
```