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

# Settings for plots
sns.set_style('whitegrid')
plt.rcParams['figure.figsize'] = (8,5)

# Load dataset
df = pd.read_csv("tested.csv")  # change filename if different
df.head()
```

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Far
0	892	0	3	Kelly, Mr. James	male	34.5	0	0	330911	7.829
1	893	1	3	Wilkes, Mrs. James (Ellen Needs)	female	47.0	1	0	363272	7.000
2	894	0	2	Myles, Mr. Thomas Francis	male	62.0	0	0	240276	9.687
3	895	0	3	Wirz, Mr. Albert	male	27.0	0	0	315154	8.662
4	896	1	3	Hirvonen, Mrs. Alexander	female	22.0	1	1	3101298	12.287

Next steps: Generate code with df | View recommended plots | New interactive sheet

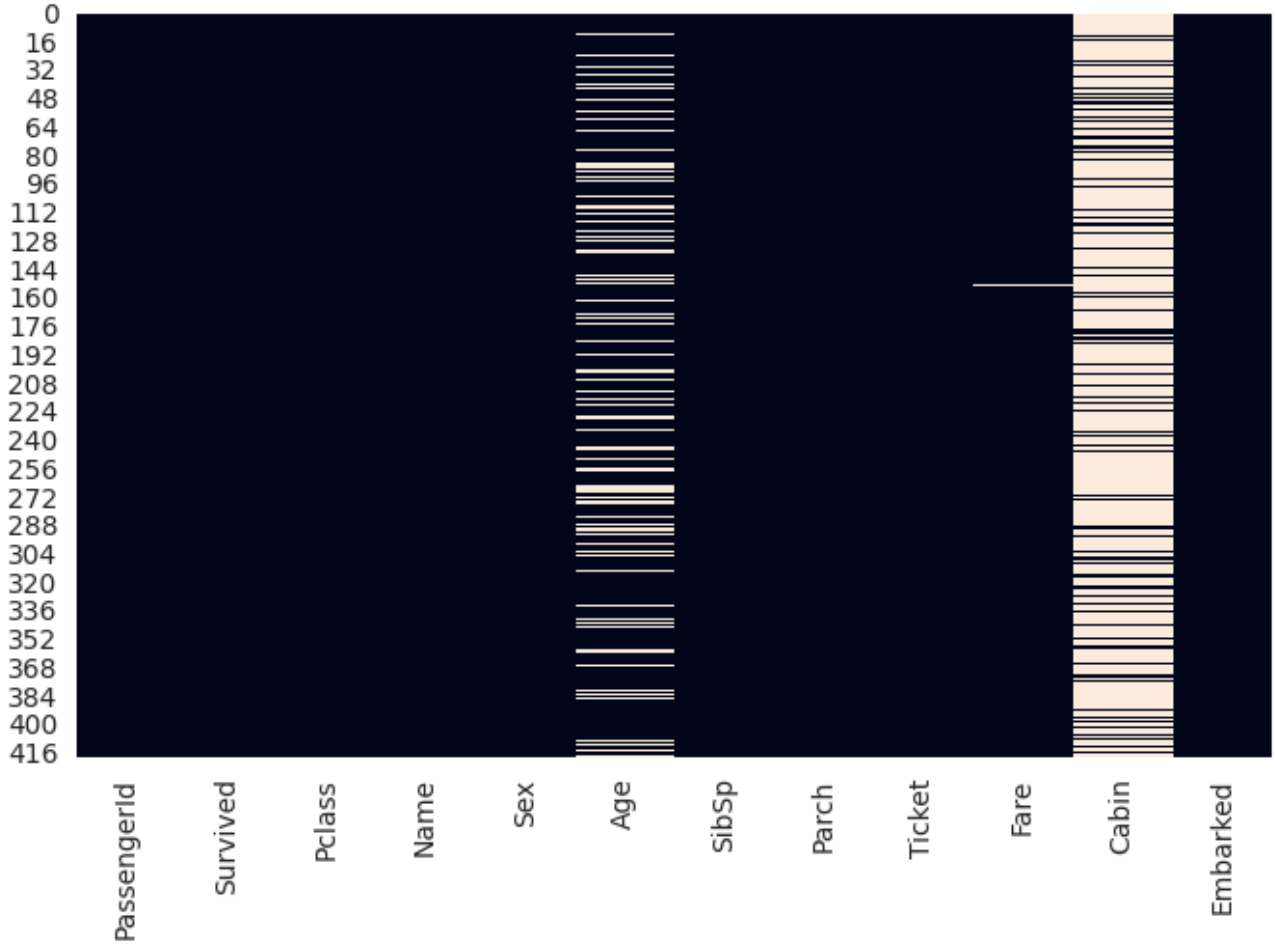
```
df.shape
df.info()
df.describe()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 418 entries, 0 to 417
Data columns (total 12 columns):
 #   Column       Non-Null Count  Dtype  
---  ------       --------------  -----  
 0   PassengerId  418 non-null    int64  
 1   Survived     418 non-null    int64  
 2   Pclass       418 non-null    int64  
 3   Name         418 non-null    object 
 4   Sex          418 non-null    object 
 5   Age          332 non-null    float64
 6   SibSp        418 non-null    int64  
 7   Parch        418 non-null    int64  
 8   Ticket       418 non-null    object 
 9   Fare         417 non-null    float64
 10  Cabin        91 non-null     object 
 11  Embarked     418 non-null    object 
dtypes: float64(2), int64(5), object(5)
memory usage: 39.3+ KB
```

	PassengerId	Survived	Pclass	Age	SibSp	Parch	Fa
count	418.000000	418.000000	418.000000	332.000000	418.000000	418.000000	417.000(
mean	1100.500000	0.363636	2.265550	30.272590	0.447368	0.392344	35.627
std	120.810458	0.481622	0.841838	14.181209	0.896760	0.981429	55.907
min	892.000000	0.000000	1.000000	0.170000	0.000000	0.000000	0.000(
25%	996.250000	0.000000	1.000000	21.000000	0.000000	0.000000	7.895
50%	1100.500000	0.000000	3.000000	27.000000	0.000000	0.000000	14.454
75%	1204.750000	1.000000	3.000000	39.000000	1.000000	0.000000	31.500(
max	1309.000000	1.000000	3.000000	76.000000	8.000000	9.000000	512.329

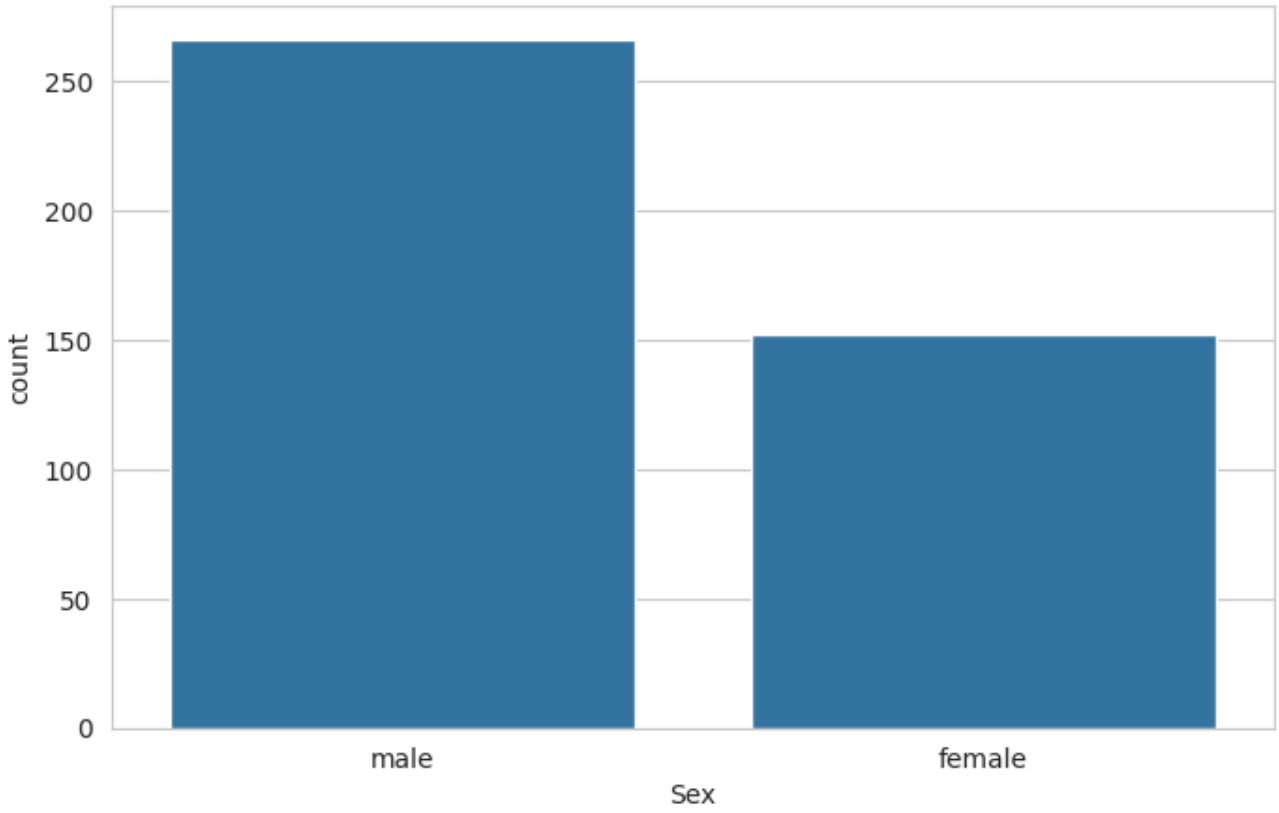
```
#missing values
df.isnull().sum()
sns.heatmap(df.isnull(), cbar=False)
```

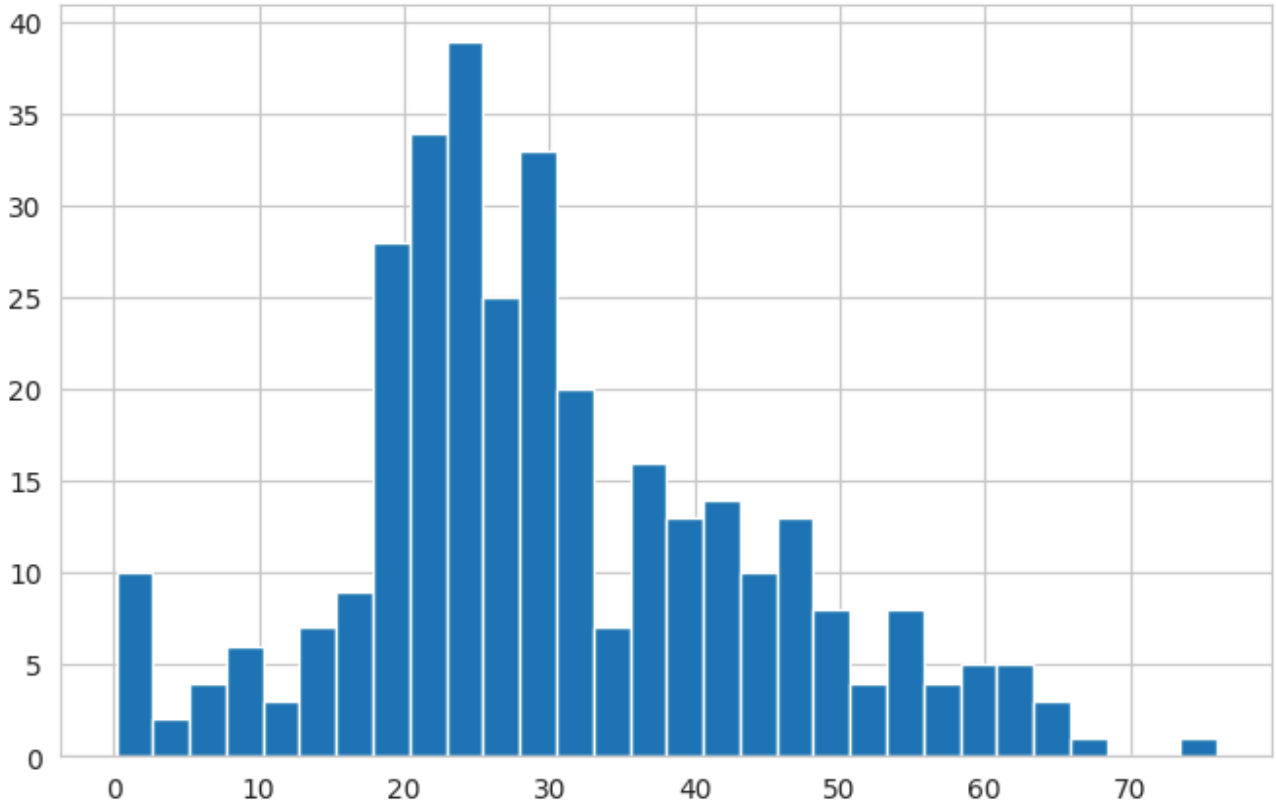
```
<Axes: >
```



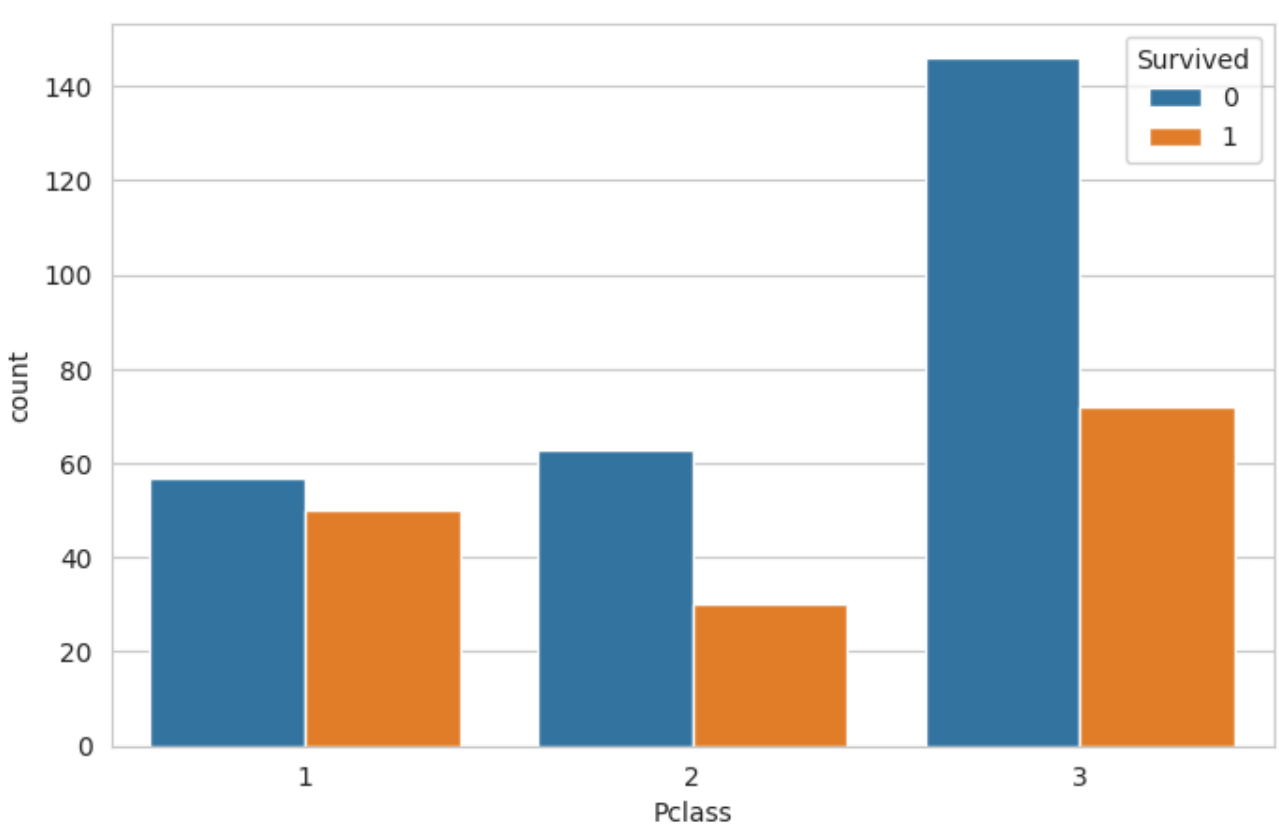

```
#univariate analysis
sns.countplot(x='Sex', data=df)
plt.show()

df['Age'].hist(bins=30)
plt.show()
```






```
#bivariate analysis
sns.countplot(x='Pclass', hue='Survived', data=df)
plt.show()
```




```
#correlation
corr = df.corr(numeric_only=True)
sns.heatmap(corr, annot=True, cmap='coolwarm')
plt.show()
```