

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

```
df = pd.read_csv('/content/Titanic-Dataset.csv')
```

```
df.head()
```

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	1	Cummings, Mrs. John Bradley (Florence Briggs Th...)	female	38.0	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
Futrelle, Mrs. Jacques Heath												

Next steps:

[Generate code with df](#)
[View recommended plots](#)
[New interactive sheet](#)

```
df.tail()
```

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
886	887	0	2	Montvila, Rev. Juozas	male	27.0	0	0	211536	13.00	NaN	S
887	888	1	1	Graham, Miss. Margaret Edith	female	19.0	0	0	112053	30.00	B42	S
888	889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	NaN	1	2	W./C. 6607	23.45	NaN	S
889	890	1	1	Behr, Mr. Karl Howell	male	26.0	0	0	111369	30.00	C148	C

```
print(df.info())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 12 columns):
#   Column      Non-Null Count  Dtype
---  -
0   PassengerId  891 non-null    int64
1   Survived     891 non-null    int64
2   Pclass       891 non-null    int64
3   Name         891 non-null    object
4   Sex          891 non-null    object
5   Age          714 non-null    float64
6   SibSp        891 non-null    int64
7   Parch        891 non-null    int64
8   Ticket       891 non-null    object
9   Fare         891 non-null    float64
10  Cabin        204 non-null    object
11  Embarked     889 non-null    object
dtypes: float64(2), int64(5), object(5)
memory usage: 83.7+ KB
None
```

```
# Fill missing Age with median
df['Age'].fillna(df['Age'].median(), inplace=True)
```

Show hidden output

```
# Drop 'Cabin' column (too many missing values)
df.drop(columns=['Cabin'], inplace=True)
```

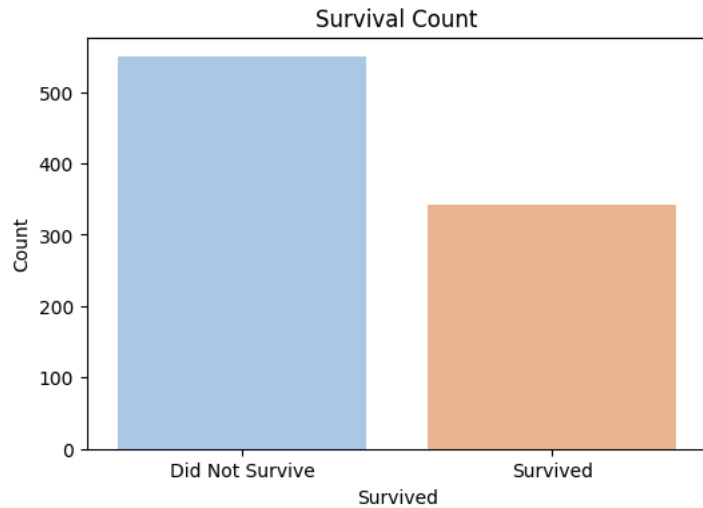
```
# Fill missing Embarked with mode
df['Embarked'].fillna(df['Embarked'].mode()[0], inplace=True)
```

Show hidden output

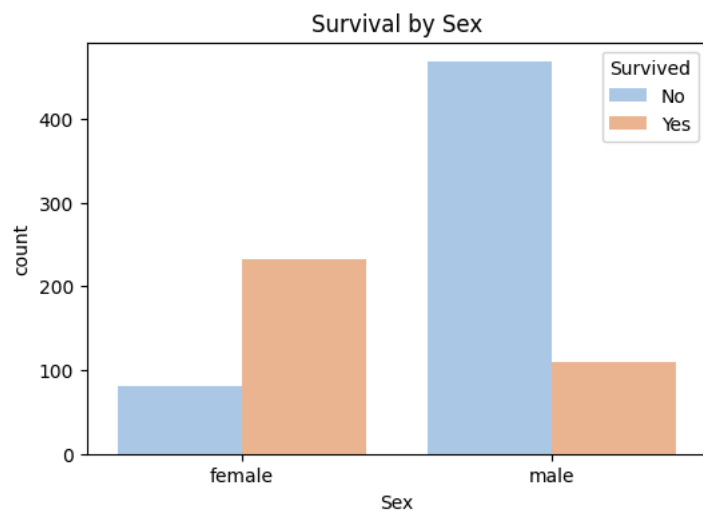
```
# Convert to category
df['Pclass'] = df['Pclass'].astype('category')
df['Sex'] = df['Sex'].astype('category')
df['Embarked'] = df['Embarked'].astype('category')
```

EDA

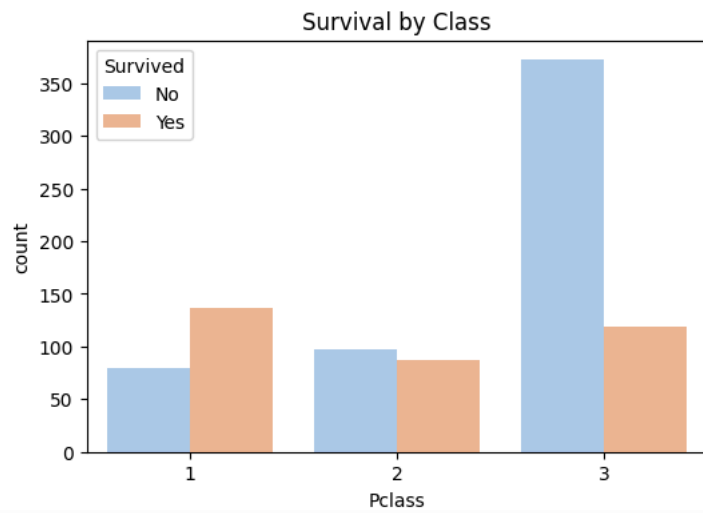
```
# 1. Survival Count (with hue for style)
plt.figure(figsize=(6, 4))
sns.countplot(x='Survived', hue='Survived', data=df, palette='pastel', legend=False)
plt.title("Survival Count")
plt.xticks([0, 1], ['Did Not Survive', 'Survived'])
plt.ylabel("Count")
plt.show()
```



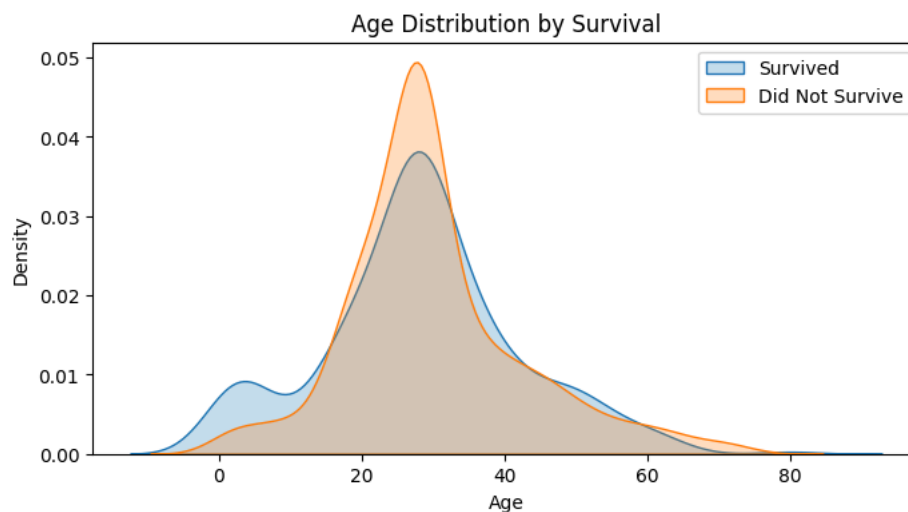
```
# 2. Survival by Sex
plt.figure(figsize=(6, 4))
sns.countplot(x='Sex', hue='Survived', data=df, palette='pastel')
plt.title("Survival by Sex")
plt.legend(title='Survived', labels=['No', 'Yes'])
plt.show()
```



```
# 3. Survival by Passenger Class
plt.figure(figsize=(6, 4))
sns.countplot(x='Pclass', hue='Survived', data=df, palette='pastel')
plt.title("Survival by Class")
plt.legend(title='Survived', labels=['No', 'Yes'])
plt.show()
```



```
# 4. Age Distribution by Survival
plt.figure(figsize=(8, 4))
sns.kdeplot(df[df['Survived'] == 1]['Age'], label='Survived', fill=True)
sns.kdeplot(df[df['Survived'] == 0]['Age'], label='Did Not Survive', fill=True)
plt.title("Age Distribution by Survival")
plt.legend()
plt.show()
```

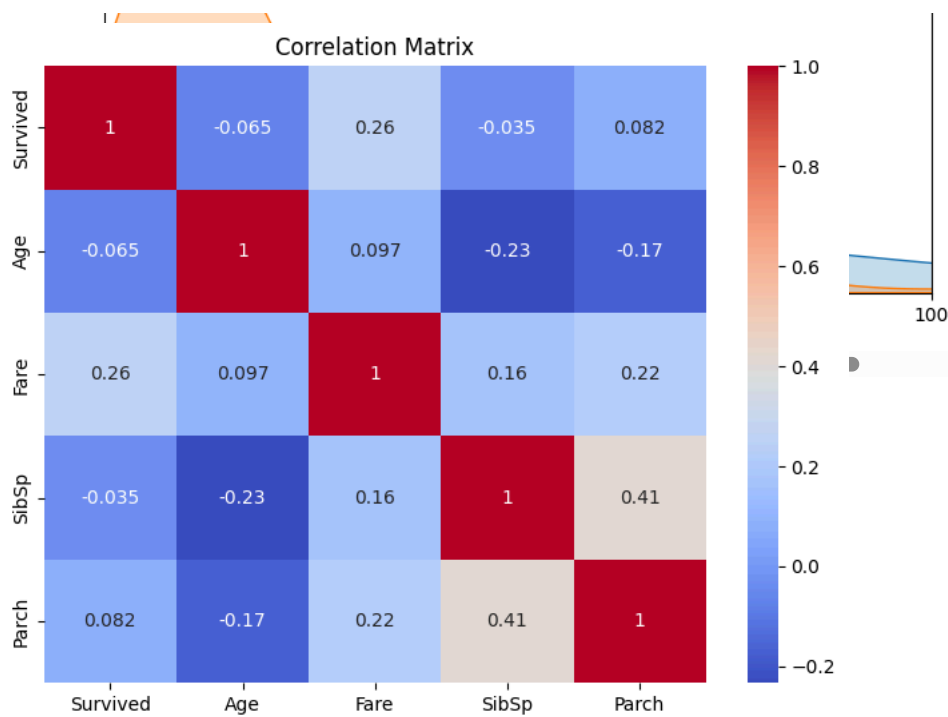


```
# 5. Fare Distribution by Survival
plt.figure(figsize=(8, 4))
sns.kdeplot(df[df['Survived'] == 1]['Fare'], label='Survived', fill=True)
sns.kdeplot(df[df['Survived'] == 0]['Fare'], label='Did Not Survive', fill=True)
plt.title("Fare Distribution by Survival")
plt.xlim(0, 100) # limit outliers
plt.legend()
plt.show()
```



Fare Distribution by Survival

```
# 6. Correlation Heatmap
plt.figure(figsize=(8, 6))
sns.heatmap(df[['Survived', 'Age', 'Fare', 'SibSp', 'Parch']].corr(), annot=True, cmap='coolwarm')
plt.title("Correlation Matrix")
plt.show()
```



Start coding or [generate](#) with AI.

Start coding or [generate](#) with AI.

Start coding or [generate](#) with AI.