Task 5: Exploratory Data Analysis (EDA)

🎯 Objective:

Extract insights using visual and statistical exploration.

🛠️ Tools:

Python (Pandas, Matplotlib, Seaborn)

📦 Dataset Used:

Titanic Dataset from Kaggle

✅ Deliverables:

- Jupyter Notebook with Code and Visuals
- PDF Report of Findings

🔎 Hints / Mini Guide:

- Use `.describe()`, `.info()`, `.value_counts()` to understand structure
- Visualize relationships using `sns.pairplot()`, `sns.heatmap()`
- Plot with `histograms`, `boxplots`, `scatterplots`
- Write meaningful observations for each plot
- Conclude with key findings

1. Dataset Overview

- Shape of dataset: Rows × Columns
- Summary statistics using `.describe()`
- Data types & missing values using `.info()` and `.isnull().sum()`

---

2. Univariate Analysis

- **Survived** (Target variable) - Count plot
- **Pclass** - Distribution
- **Sex** - Count of males vs females
- **Age** - Histogram & boxplot
- **Fare** - Histogram

---

3. Bivariate Analysis

- **Survival vs Sex** – Bar plot
- **Survival vs Pclass** – Stacked bar
- **Age vs Survival** – Boxplot
- **Fare vs Survival** – KDE plot or histogram
- **Heatmap** – Correlation matrix

---

4. Key Insights

- Gender played a key role in survival.
- Higher-class passengers had better survival chances.
- Younger individuals and those who paid higher fares also had higher survival rates.

---

5. Conclusion

- The analysis reveals significant patterns in survival based on demographic and economic factors.
- Recommendations for model-building based on top features: Sex, Pclass, Age, Fare.

6. Code :

```
# 1. Importing Libraries
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# Optional: Style settings
sns.set(style="whitegrid")
plt.rcParams["figure.figsize"] = (8, 5)

# 2. Load Dataset
df = pd.read_csv('/content/Titanic-Dataset.csv')

# 3. Initial Exploration
print(df.head())
```

```
   PassengerId  Survived  Pclass  \
0            1         0       3
1            2         1       1
2            3         1       3
3            4         1       1
4            5         0       3

                                                Name     Sex   Age  SibSp  \
0                            Braund, Mr. Owen Harris    male  22.0      1
1  Cumings, Mrs. John Bradley (Florence Briggs Th...  female  38.0      1
2                             Heikkinen, Miss. Laina  female  26.0      0
3       Futrelle, Mrs. Jacques Heath (Lily May Peel)  female  35.0      1
4                           Allen, Mr. William Henry    male  35.0      0

   Parch            Ticket     Fare Cabin Embarked
0      0         A/5 21171   7.2500   NaN        S
1      0          PC 17599  71.2833   C85        C
2      0  STON/O2. 3101282   7.9250   NaN        S
3      0            113803  53.1000  C123        S
4      0            373450   8.0500   NaN        S
```

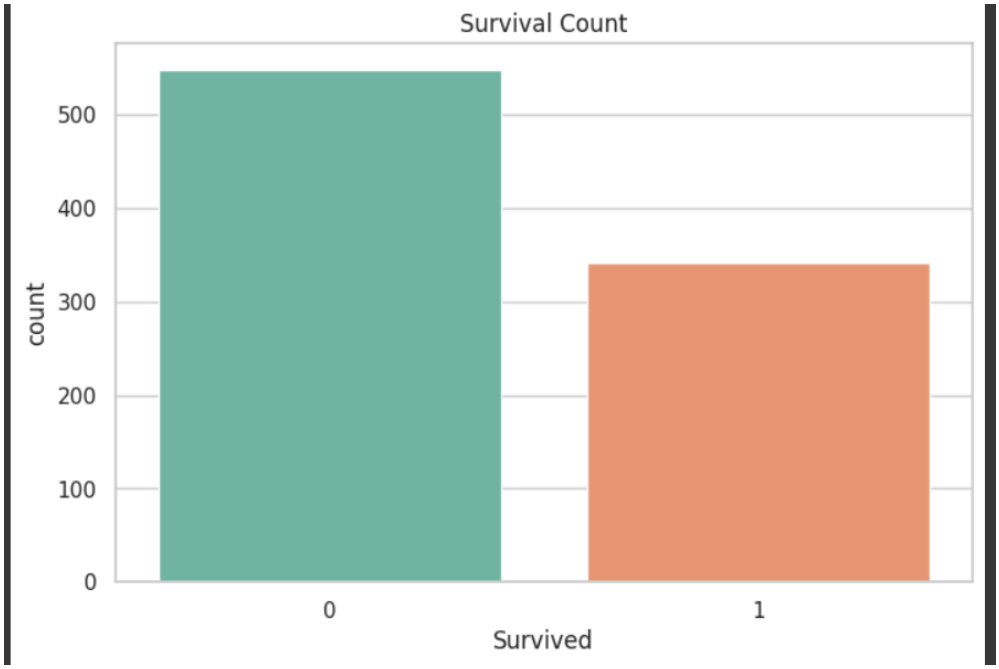
```
print(df.info())
```

```
<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
```

```
print(df.describe())
print(df.isnull().sum())
```

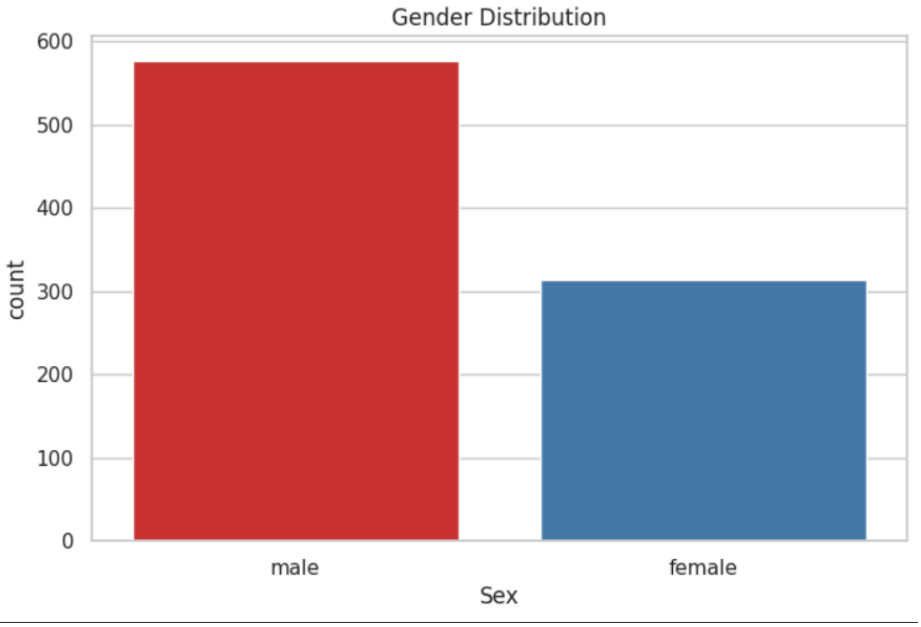
```
       PassengerId    Survived      Pclass         Age       SibSp  \
count   891.000000  891.000000  891.000000  714.000000  891.000000
mean    446.000000    0.383838    2.308642   29.699118    0.523008
std     257.353842    0.486592    0.836071   14.526497    1.102743
min       1.000000    0.000000    1.000000    0.420000    0.000000
25%     223.500000    0.000000    2.000000   20.125000    0.000000
50%     446.000000    0.000000    3.000000   28.000000    0.000000
75%     668.500000    1.000000    3.000000   38.000000    1.000000
max     891.000000    1.000000    3.000000   80.000000    8.000000

             Parch        Fare
count  891.000000  891.000000
mean     0.381594   32.204208
std      0.806057   49.693429
min      0.000000    0.000000
25%      0.000000    7.910400
50%      0.000000   14.454200
75%      0.000000   31.000000
max      6.000000  512.329200
PassengerId      0
Survived         0
Pclass           0
Name             0
Sex              0
Age            177
SibSp            0
Parch            0
Ticket           0
Fare             0
```

```
sns.countplot(x='Survived', data=df, palette='Set2')
plt.title('Survival Count')
plt.show()
```

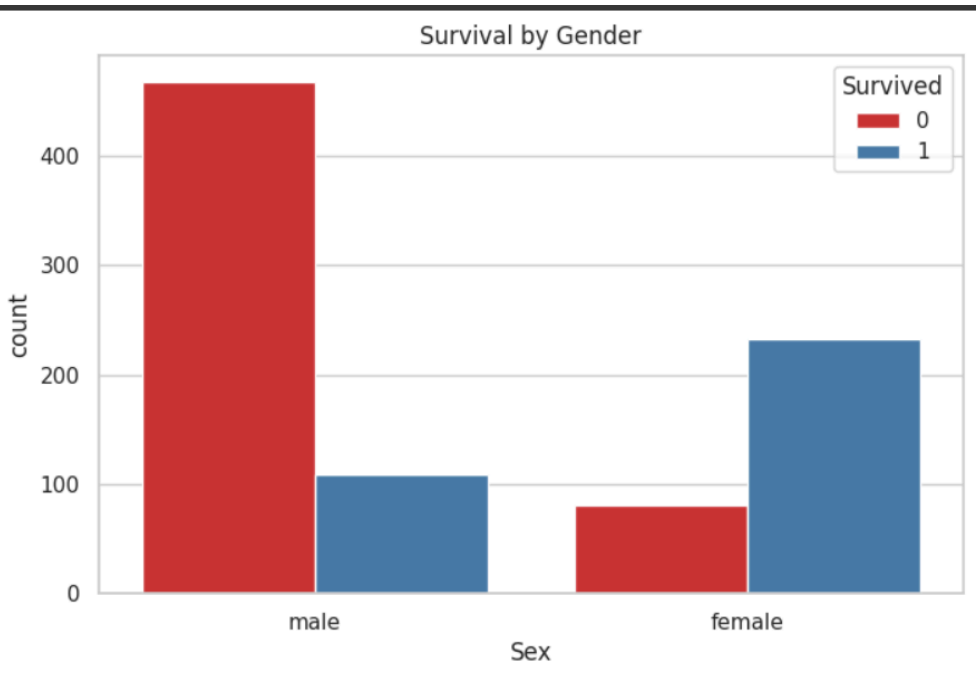



```
sns.countplot(x='Sex', data=df, palette='Set1')
plt.title('Gender Distribution')
plt.show()
```

```
sns.countplot(x='Sex', data=df, palette='Set1')
```

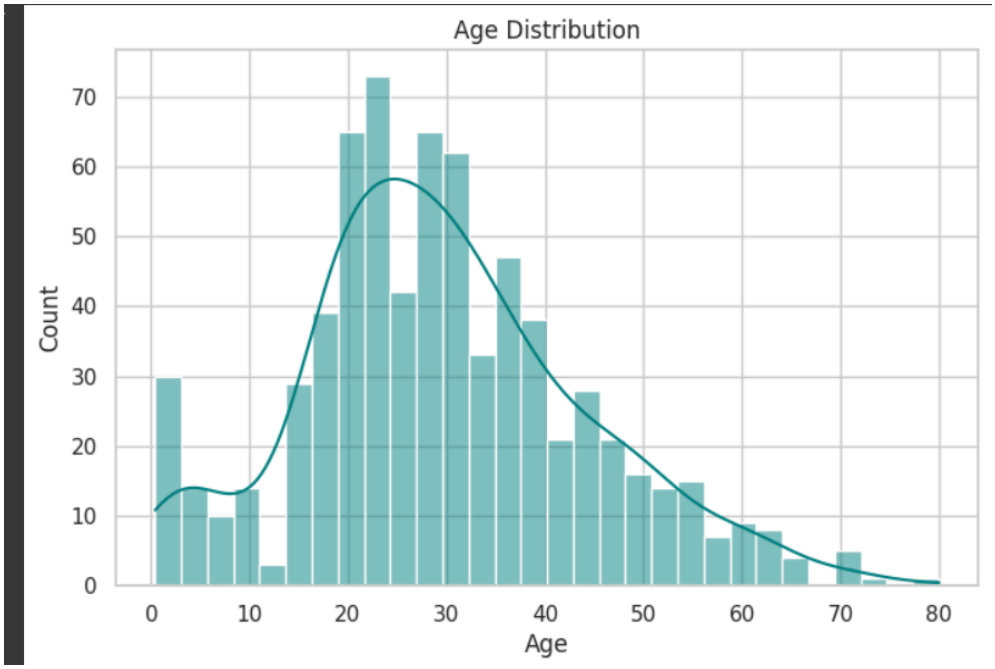



```
sns.countplot(x='Sex', hue='Survived', data=df, palette='Set1')
plt.title('Survival by Gender')
plt.show()
```

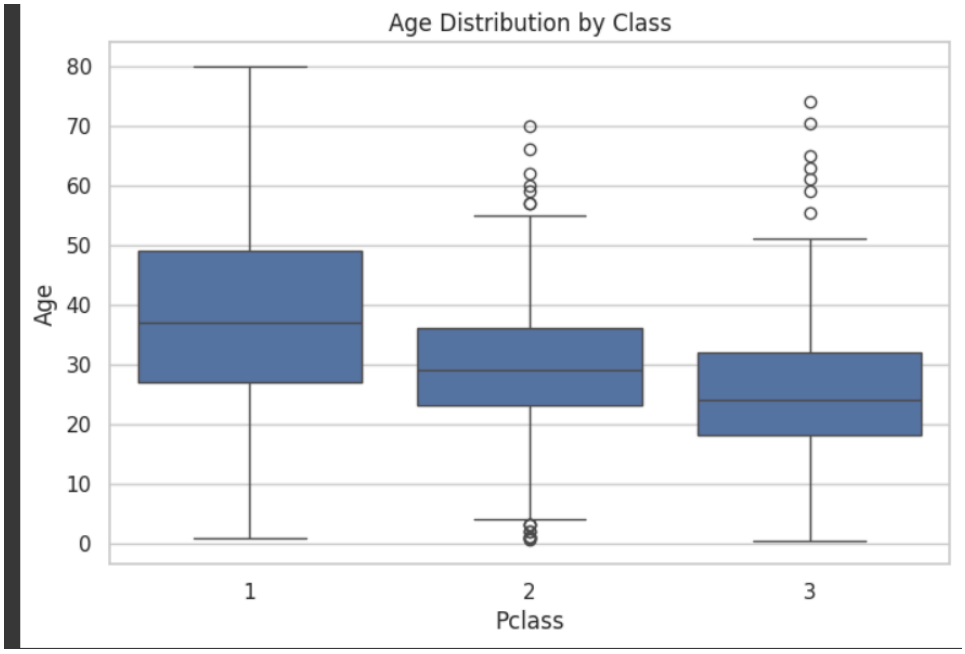



```
sns.histplot(df['Age'].dropna(), bins=30, kde=True, color='teal')
plt.title('Age Distribution')
```

```
plt.show()
```




```
sns.boxplot(x='Pclass', y='Age', data=df)
plt.title('Age Distribution by Class')
plt.show()
```




```
plt.figure(figsize=(10, 6))
```

```
sns.heatmap(df.corr(numeric_only=True), annot=True, cmap='coolwarm')
plt.title('Correlation Heatmap')
plt.show()
```