Importing and Loading Data

In this section, all necessary libraries have been imported, and the Titanic dataset has been loaded into a pandas DataFrame called `titanic`. The libraries used include:

- **Pandas**: For data manipulation and analysis.
- **NumPy**: For numerical operations and handling arrays.
- **Matplotlib & Seaborn**: For data visualization.
- **SciPy**: For statistical functions.

Loading the dataset allows us to begin exploring the data and understanding its structure, which is crucial for subsequent cleaning and analysis steps.

```
# Importing all necessary libraries
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
from scipy import stats
from scipy.stats import chi2_contingency  # Importing Chi-Square test
function

# Setting Seaborn’s darkgrid style for contrast
sns.set(style="darkgrid")

# Setting Matplotlib style for clean visuals
plt.style.use('seaborn-v0_8-dark')

# Loading the Titanic dataset into a pandas DataFrame called 'titanic'
titanic = pd.read_csv('titanic/train.csv')

# Displaying the first 10 rows of the dataset to inspect
titanic.head(10)
```

```
   PassengerId  Survived  Pclass  \
0            1         0       3
1            2         1       1
2            3         1       3
3            4         1       1
4            5         0       3
5            6         0       3
6            7         0       1
7            8         0       3
8            9         1       3
9           10         1       2

                                                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
5                                   Moran, Mr. James    male   NaN
0
6                            McCarthy, Mr. Timothy J    male  54.0
0
7                     Palsson, Master. Gosta Leonard    male   2.0
3
8  Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg)  female  27.0
0
9                Nasser, Mrs. Nicholas (Adele Achem)  female  14.0
1

   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
5      0            330877   8.4583   NaN        Q
6      0             17463  51.8625   E46        S
7      1            349909  21.0750   NaN        S
8      2            347742  11.1333   NaN        S
9      0            237736  30.0708   NaN        C
```

Data Cleaning

Data cleaning is a crucial step in preparing the dataset for analysis. This process involves identifying and addressing missing values and other inconsistencies in the data. The following actions have been taken:

1. **Identifying Missing Values**:
   Initially, the missing values in the dataset have been identified using the `isnull().sum()` method. This step helps us determine which columns have missing data and informs the strategy for handling them.

2. **Filling Missing Values for 'Age'**:
   The **Age** column had missing values that could not be dropped due to the importance of age in survival analysis. We filled these missing values using the **median**, as it is less sensitive to outliers than the mean. This ensures the filled values represent the typical age distribution without skewing the data.

3. **Dropping the ‘Cabin’ Column**:
   The **Cabin** column was found to have a large proportion of missing values (more than **77%**). Since imputing these values would introduce too much noise, we dropped the column to maintain the integrity of the dataset.

4. **Filling Missing Values for ‘Embarked’**:
   Missing values in the **Embarked** column were filled with the **mode** (most frequent value). This choice helps maintain the completeness of the dataset without removing rows that may contain valuable information. Dropping rows with missing **Embarked** values could potentially reduce the dataset’s richness and lead to biased analysis.

5. **Dropping Duplicate Rows**:
   We checked for and dropped any **duplicate rows** to ensure each entry in the dataset represents a **unique individual**. Duplicates can lead to biased results in analysis, so it is critical to eliminate them before further steps.

6. **Creating an ‘Age Group’ Column**:
   To facilitate deeper analysis, a new column named **Age Group** has been created. This column categorizes passengers into three distinct groups:

   - **Child** (under **18 years**)
   - **Adult** (**18-59 years**)
   - **Senior** (**60 years and above**)

   This categorization helps analyze survival rates based on different age groups and provides additional insights.

These steps have been taken to clean the data thoroughly and ensure it is ready for further analysis. The approach balances maintaining as much data as possible while minimizing theintroduction of inaccuracies. introduction of inaccuracies.

```
# Checking for missing values
missing_values = titanic.isnull().sum()
print(missing_values)
```

```
PassengerId      0
Survived         0
Pclass           0
Name             0
Sex              0
Age            177
SibSp            0
Parch            0
Ticket           0
Fare             0
Cabin          687
```

```
Embarked       2
dtype: int64

# Filling missing 'Age' values with the median
titanic['Age'] = 
titanic['Age'].fillna(np.median(titanic['Age'].dropna()))

# Dropping the 'Cabin' column due to too many missing values
titanic = titanic.drop(columns=['Cabin'])

# Filling missing 'Embarked' values with the mode (most common value)
titanic['Embarked'] = 
titanic['Embarked'].fillna(titanic['Embarked'].mode()[0])

# Dropping duplicate rows if any exist
titanic = titanic.drop_duplicates()

# Defining age group categories
def categorize_age(age):
    """Categorizes age into Child, Adult, or Senior."""
    if age < 18:
        return 'Child'
    elif 18 <= age < 60:
        return 'Adult'
    else:
        return 'Senior'

# Creating a new column 'AgeGroup' in the dataset
titanic['AgeGroup'] = titanic['Age'].apply(categorize_age)
```

Exploratory Data Analysis (EDA)

The EDA phase involves visualizing the data to uncover underlying patterns and relationships. Key actions taken include:

1. **Displaying a Summary of the Cleaned Dataset**: A summary of the cleaned dataset has been displayed using the `describe()` method. This summary includes count, mean, standard deviation, minimum, and maximum values for the numerical columns, which provides insights into the dataset's structure and characteristics.

2. **Visualizing Gender Distribution**: A count plot has been created to show the gender distribution of passengers. This visualization helps to understand the gender demographics on board the Titanic. The results indicate that there were more male passengers than female passengers, suggesting that the journey was predominantly male.

3. **Visualizing Age Distribution**: A histogram has been generated to display the distribution of passenger ages. The histogram reveals a significant number of young passengers, including children, and shows that the majority of passengers were

adults. Understanding age distribution is crucial for analyzing how age may have influenced survival rates.

4. **Survival Rate by Age Group**: After categorizing ages into groups (Child, Adult, Senior), a bar plot has been created to visualize survival rates among these age groups. This visualization indicates that children had a higher survival rate compared to adults and seniors. This trend could suggest that children were prioritized during evacuation.

5. **Survival Rate by Gender and Class**: A grouped bar plot has been generated to compare survival rates across different genders and passenger classes (1st, 2nd, and 3rd class). The results show that:

    - Female passengers had a significantly higher survival rate compared to male passengers in all classes.

    - The survival rate was highest among females in 1st class, followed by females in 2nd and 3rd classes.

    - Males in 1st class had a higher survival rate than those in 2nd and 3rd classes, but overall, their survival rate was much lower than that of females.

    This visualization emphasizes the impact of gender and class on survival chances, reflecting social norms and practices during emergencies.

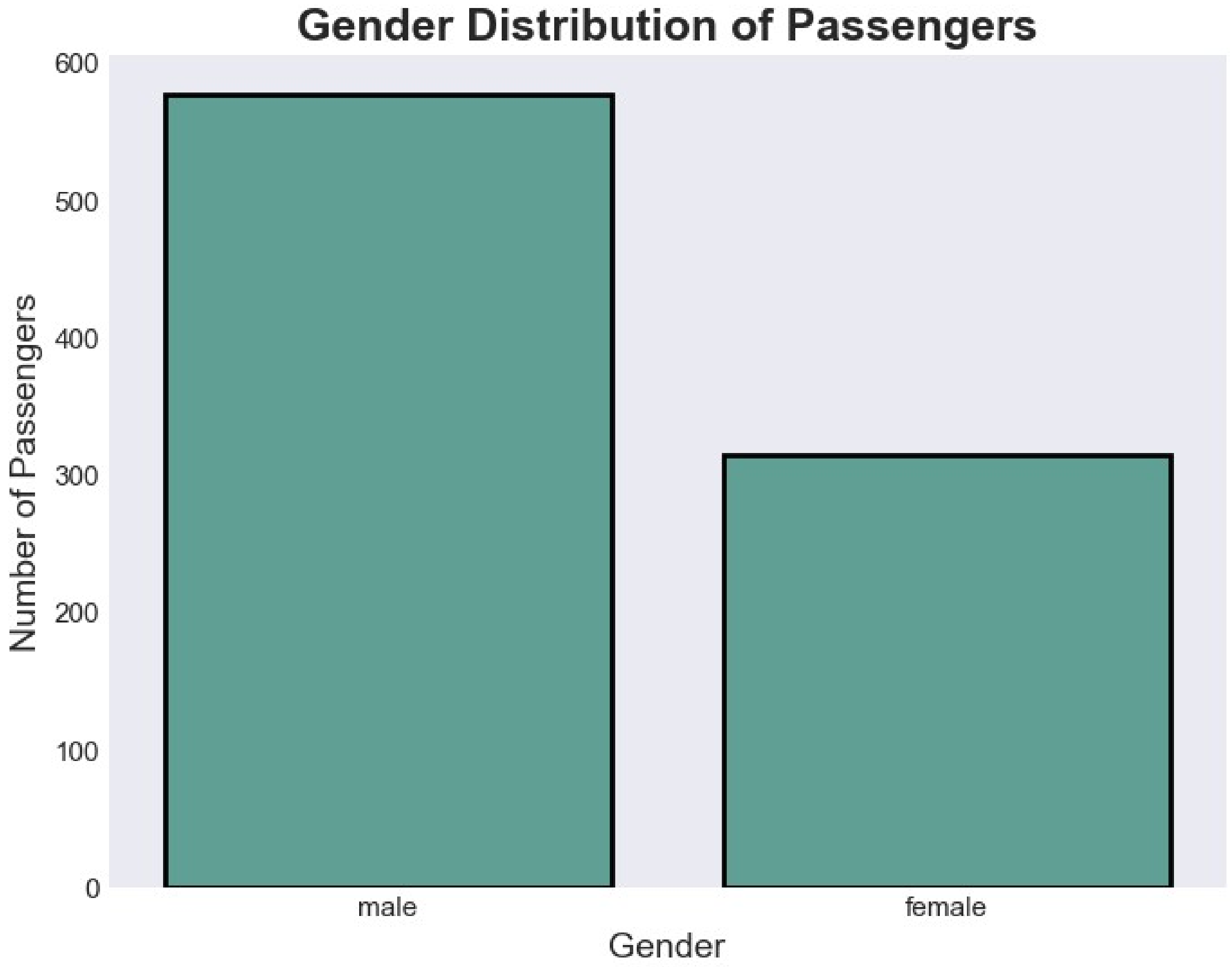
6. **Correlation Matrix**: The correlation matrix has been computed and visualized using a heatmap. This analysis reveals relationships between numeric features in the dataset. For example, there is a moderate positive correlation between the `Fare` and survival rates, indicating that those who paid higher fares had a better chance of survival. A strong correlation could indicate that one variable may predict another, which is essential for building predictive models.

These visualizations and statistical summaries provide insights into the dataset, allowing for informed hypotheses and further analysis. By examining survival rates across different demographic factors, we can better understand the complex dynamics at play during the Titanic disaster.

```
# Displaying a summary of the cleaned dataset
titanic.describe().round(2)
```

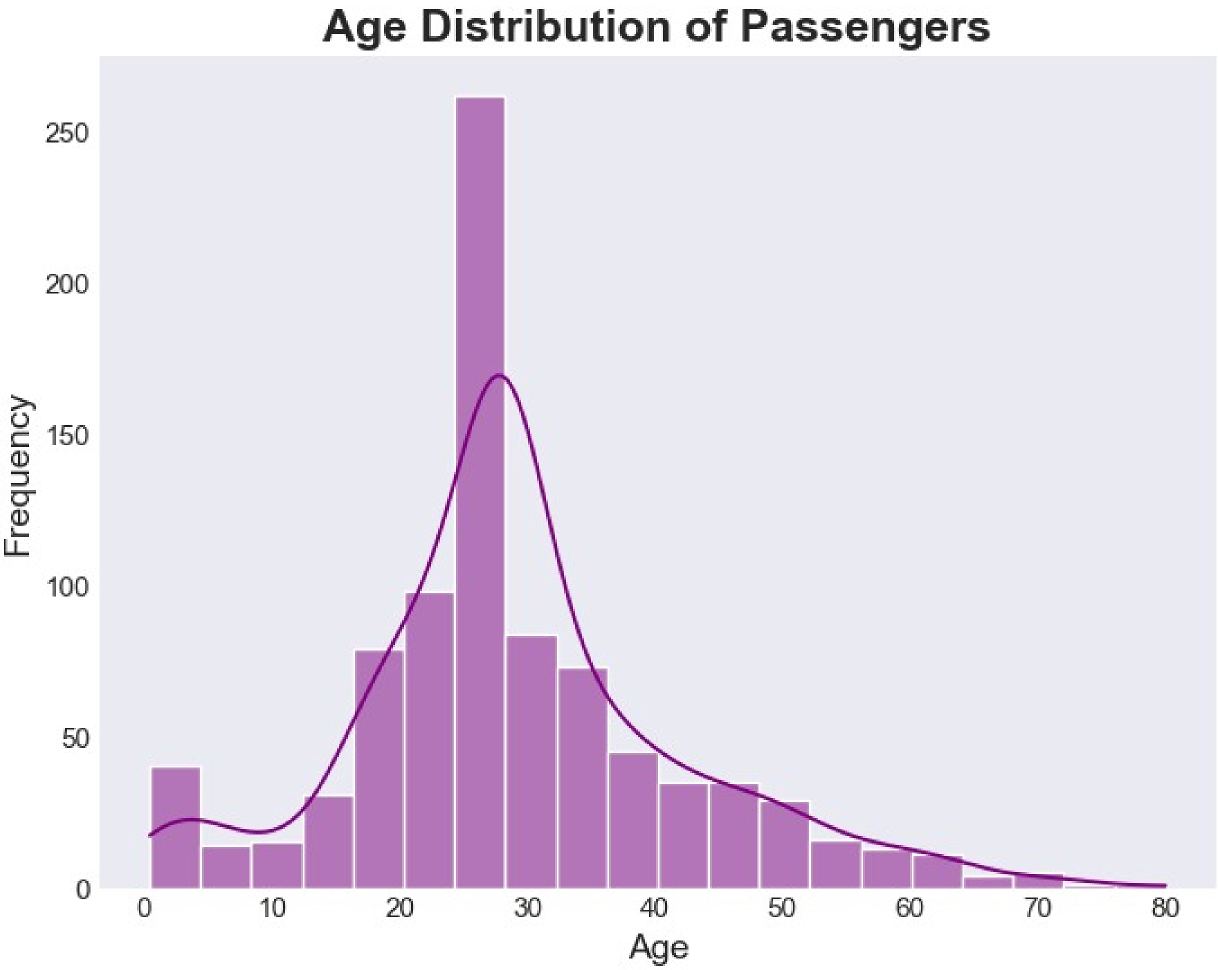
```
       PassengerId  Survived  Pclass     Age   SibSp   Parch    Fare
count       891.00    891.00  891.00  891.00  891.00  891.00  891.00
mean        446.00      0.38    2.31   29.36    0.52    0.38   32.20
std         257.35      0.49    0.84   13.02    1.10    0.81   49.69
min           1.00      0.00    1.00    0.42    0.00    0.00    0.00
25%         223.50      0.00    2.00   22.00    0.00    0.00    7.91
50%         446.00      0.00    3.00   28.00    0.00    0.00   14.45
75%         668.50      1.00    3.00   35.00    1.00    0.00   31.00
max         891.00      1.00    3.00   80.00    8.00    6.00  512.33
```

```python
# Setting figure size and dark blueish palette
plt.figure(figsize=(8, 6))
sns.set_palette("dark:#5A9_r")

# Creating a bar plot for gender distribution
sns.countplot(x='Sex', data=titanic, edgecolor='black', linewidth=2)

# Customizing title and labels with larger font sizes
plt.title('Gender Distribution of Passengers', fontsize=18,
weight='bold')
plt.xlabel('Gender', fontsize=14)
plt.ylabel('Number of Passengers', fontsize=14)

# Showing the plot
plt.show()
```

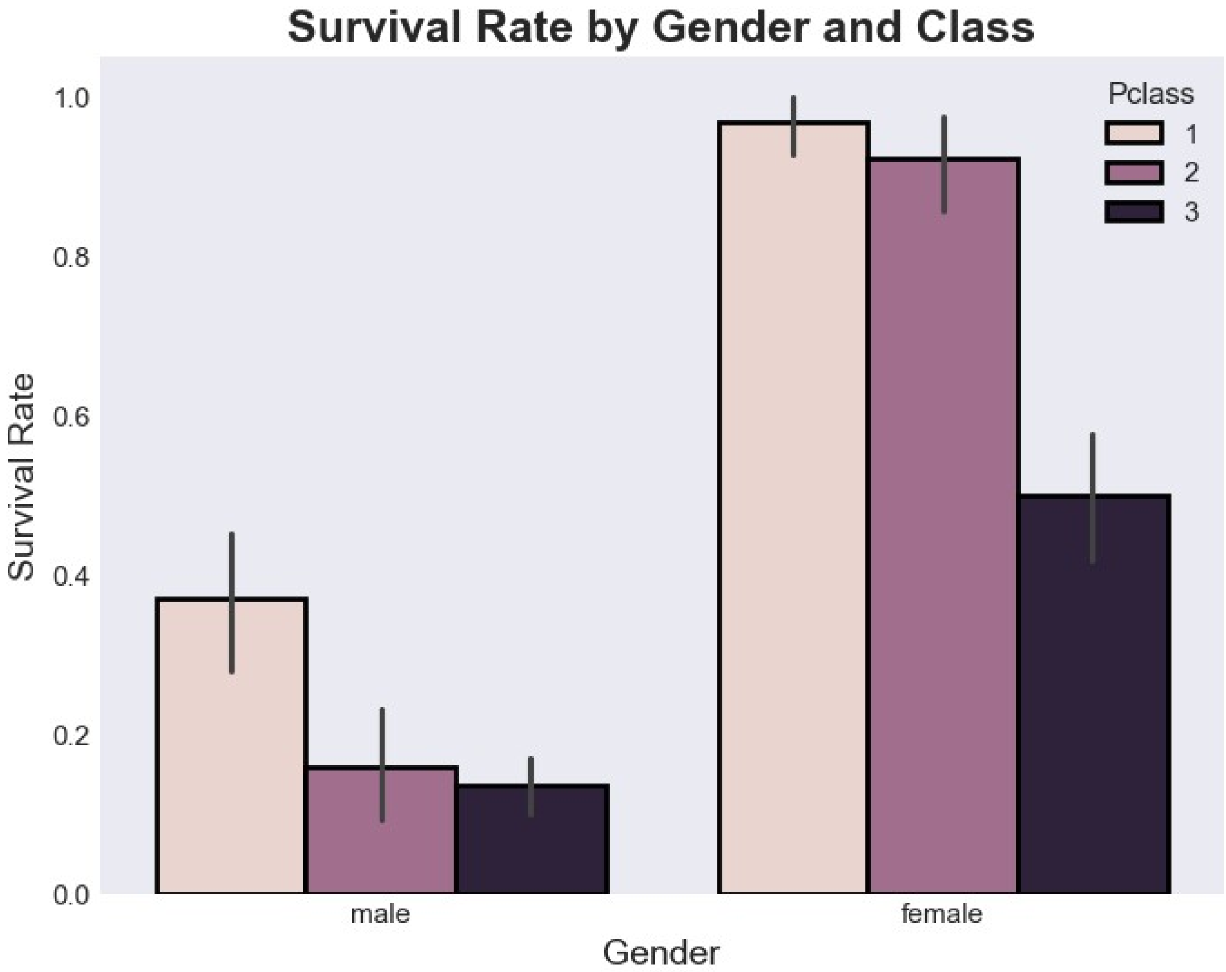


```python
# Setting figure size and violet color palette
plt.figure(figsize=(8, 6))
sns.set_palette("Purples_r")
```

```
# Creating a histogram for age distribution
sns.histplot(titanic['Age'], bins=20, kde=True, color='purple')

# Adding title and labels
plt.title('Age Distribution of Passengers', fontsize=18,
weight='bold')
plt.xlabel('Age', fontsize=14)
plt.ylabel('Frequency', fontsize=14)

# Showing the plot
plt.show()
```

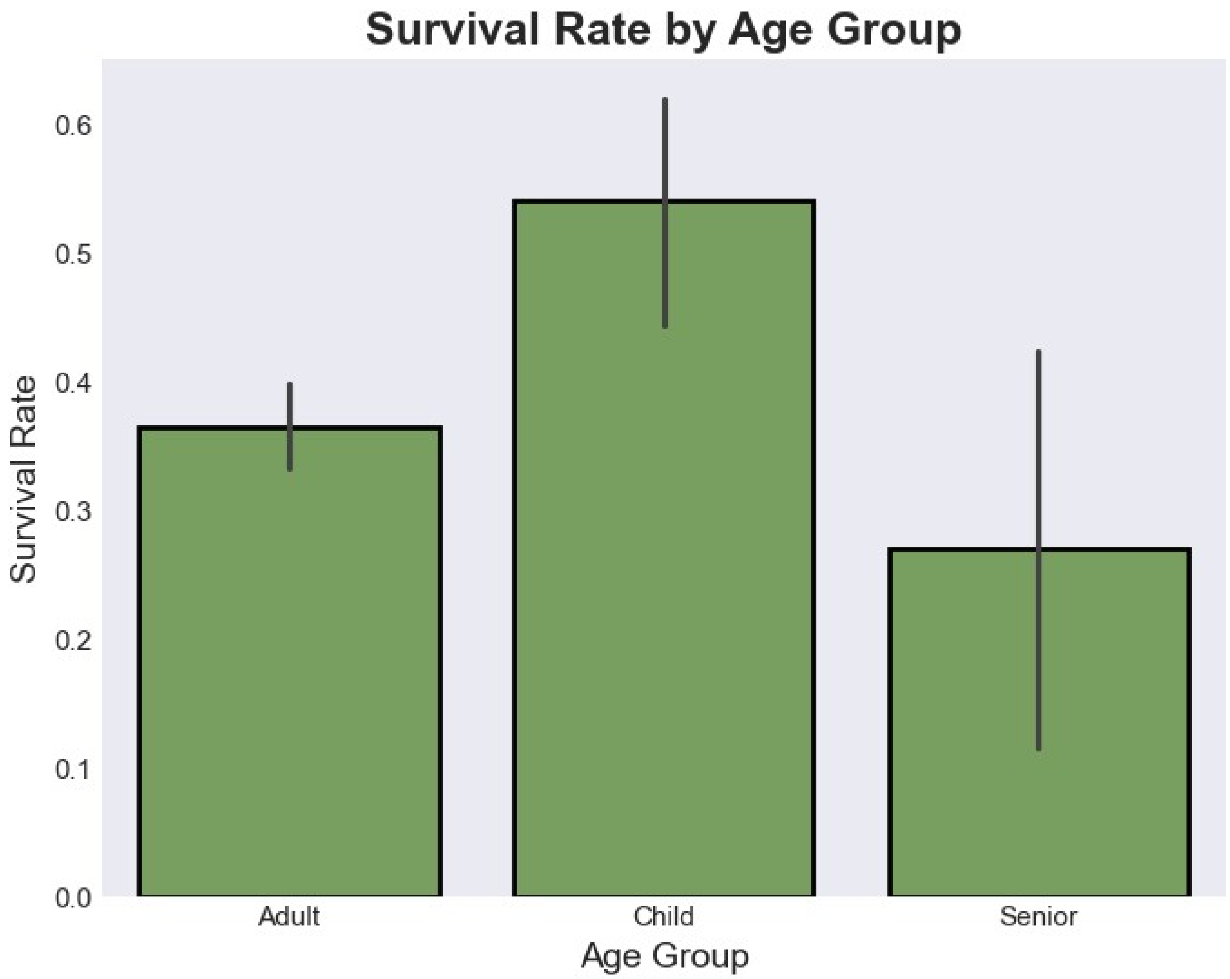


```
# Setting figure size and another dark palette
plt.figure(figsize=(8, 6))
sns.set_palette("dark:#6A5_r")

# Creating a bar plot to visualize survival rates by gender and class
sns.barplot(x='Sex', y='Survived', hue='Pclass', data=titanic,
edgecolor='black', linewidth=2)

# Adding title and labels
plt.title('Survival Rate by Gender and Class', fontsize=18,
weight='bold')
plt.xlabel('Gender', fontsize=14)
plt.ylabel('Survival Rate', fontsize=14)

# Showing the plot
plt.show()
```



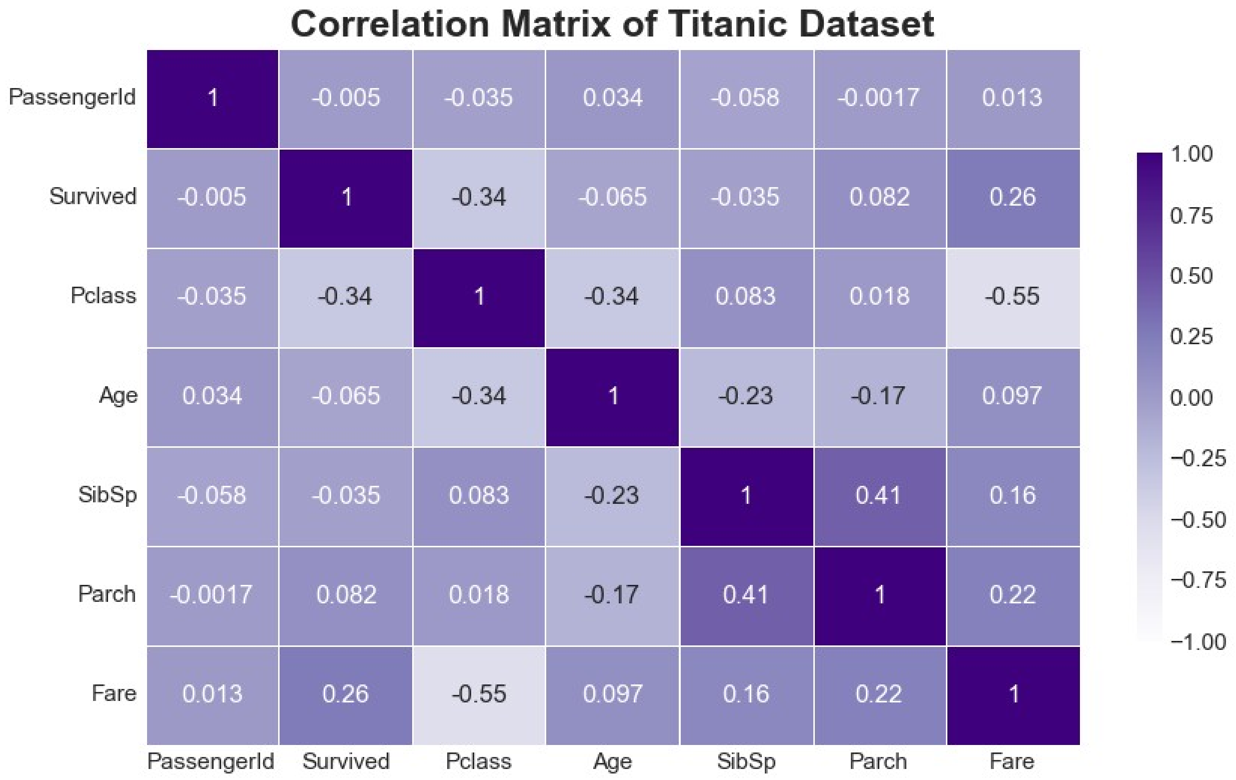
```
# Bar plot for survival rate by age group
plt.figure(figsize=(8, 6))
sns.set_palette("dark:#7A5_r")

# Creating a bar plot showing survival rate by age group
sns.barplot(x='AgeGroup', y='Survived', data=titanic,
edgecolor='black', linewidth=2)
```

```
# Customizing title and labels
plt.title('Survival Rate by Age Group', fontsize=18, weight='bold')
plt.xlabel('Age Group', fontsize=14)
plt.ylabel('Survival Rate', fontsize=14)

# Showing the plot
plt.show()
```



```
# Dropping non-numeric columns before calculating the correlation
matrix
titanic_numeric = titanic.drop(columns=['Name', 'Ticket', 'Embarked',
'Sex', 'AgeGroup'])

# Setting figure size and dark palette
plt.figure(figsize=(10, 6))
corr_matrix = titanic_numeric.corr()

# Creating a heatmap for the correlation matrix with dark violet-blue
colors
```

```
sns.heatmap(corr_matrix, annot=True, cmap='Purples', linewidths=0.5,
vmin=-1, vmax=1, cbar_kws={'shrink': 0.7})

# Adding title to the heatmap
plt.title('Correlation Matrix of Titanic Dataset', fontsize=18,
weight='bold')

# Showing the heatmap
plt.show()
```



	PassengerId	Survived	Pclass	Age	SibSp	Parch	Fare
PassengerId	1	-0.005	-0.035	0.034	-0.058	-0.0017	0.013
Survived	-0.005	1	-0.34	-0.065	-0.035	0.082	0.26
Pclass	-0.035	-0.34	1	-0.34	0.083	0.018	-0.55
Age	0.034	-0.065	-0.34	1	-0.23	-0.17	0.097
SibSp	-0.058	-0.035	0.083	-0.23	1	0.41	0.16
Parch	-0.0017	0.082	0.018	-0.17	0.41	1	0.22
Fare	0.013	0.26	-0.55	0.097	0.16	0.22	1

Basic Statistical Analysis

In this section, key statistics have been calculated to better understand the central tendencies and variations in the dataset. Additionally, a t-test and chi-square test have been performed to explore statistical relationships.

1. **Calculating Mean, Median, and Mode**:
   The mean, median, and mode of the `Fare` and `Age` columns have been calculated to summarize the central tendencies and dispersion in these numerical variables.

   – **Fare:**
     - Mean: $32.2
     - Median: $14.45
     - Mode: $8.05

   – **Age**:

    - Mean: 29.36 years
    - Median: 28 years
    - Mode: 28 years

2. **T-test for Survival Rate by Gender**:
   A t-test has been conducted to determine if there is a statistically significant difference in the survival rates between males and females. This test helps assess whether gender played a significant role in survival chances.

   – **Result**: The t-test reveals a p-value of `p = 2.6993e-11`, which indicates the observed differences in survival rates between genders are statistically significant.

3. **Chi-Square Test for Gender vs. Survival**:
   A chi-square test has been performed to examine the relationship between gender and survival. This test assesses whether survival rates vary significantly between male and female passengers.

   – **Chi-Square Result**: The chi-square test produced a chi-square value of `260.7170` with a p-value of `p = 1.1974e-58`. This result helps to determine the association between gender and survival is statistically significant.

These analyses provide a deeper understanding of the dataset's variables and relationships, helping to uncover important factors that influenced passenger survival on the Titanic.

```python
# Fare statistics
fare_mean = np.mean(titanic['Fare']).round(2)
fare_median = np.median(titanic['Fare']).round(2)
fare_mode = stats.mode(titanic['Fare'], nan_policy='omit').mode

# Age statistics
age_mean = np.mean(titanic['Age']).round(2)
age_median = np.median(titanic['Age']).round(2)
age_mode = stats.mode(titanic['Age'], nan_policy='omit').mode

# Printing the results
print(f"Fare - Mean: {fare_mean}, Median: {fare_median}, Mode:
{fare_mode if fare_mode.size > 0 else 'N/A'}")
print(f"Age - Mean: {age_mean}, Median: {age_median}, Mode: {age_mode
if age_mode.size > 0 else 'N/A'}")
```

```
Fare - Mean: 32.2, Median: 14.45, Mode: 8.05
Age - Mean: 29.36, Median: 28.0, Mode: 28.0
```

```python
# Separating fare data based on survival
fare_survived = titanic[titanic['Survived'] == 1]['Fare']
fare_not_survived = titanic[titanic['Survived'] == 0]['Fare']

# Performing a t-test for fare between survivors and non-survivors
t_stat_fare, p_val_fare = stats.ttest_ind(fare_survived,
fare_not_survived, equal_var=False)
```

```python
# Printing the t-test results in scientific notation
print(f"T-test for Fare: T-statistic: {t_stat_fare:.4e}, P-value:
{p_val_fare:.4e}")
```

```
T-test for Fare: T-statistic: 6.8391e+00, P-value: 2.6993e-11
```

```python
# Creating a contingency table for gender vs survival
contingency_table_gender = pd.crosstab(titanic['Sex'],
titanic['Survived'])
chi2_gender, p_gender, dof_gender, expected_gender =
chi2_contingency(contingency_table_gender)

# Printing Chi-Square test results for gender vs survival
print(f"Chi-Square Test for Gender vs Survival: Chi2 Statistic:
{chi2_gender:.4f}, P-value: {p_gender:.4e}")
```

```
Chi-Square Test for Gender vs Survival: Chi2 Statistic: 260.7170, P-
value: 1.1974e-58
```