TITANIC DATA ANALYSIS

Code and Output:

[9]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from tkinter import Tk, filedialog

# Open file picker dialog
Tk().withdraw()  # Hide the root window
file_path = filedialog.askopenfilename(title="Select Titanic CSV File")

# Read the CSV file
data = pd.read_csv(file_path)

# Preview the data
print("File loaded successfully!\n")
print(data.head())
```

```
File loaded successfully!

   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
```

[4]:
```
print(data.describe())
```

```
       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
```

[14]:
```
df = pd.read_csv(r"C:\Users\priya\Downloads\archive (2)\Titanic-Dataset.csv")
```

[15]:
```
# Show a quick preview
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
```

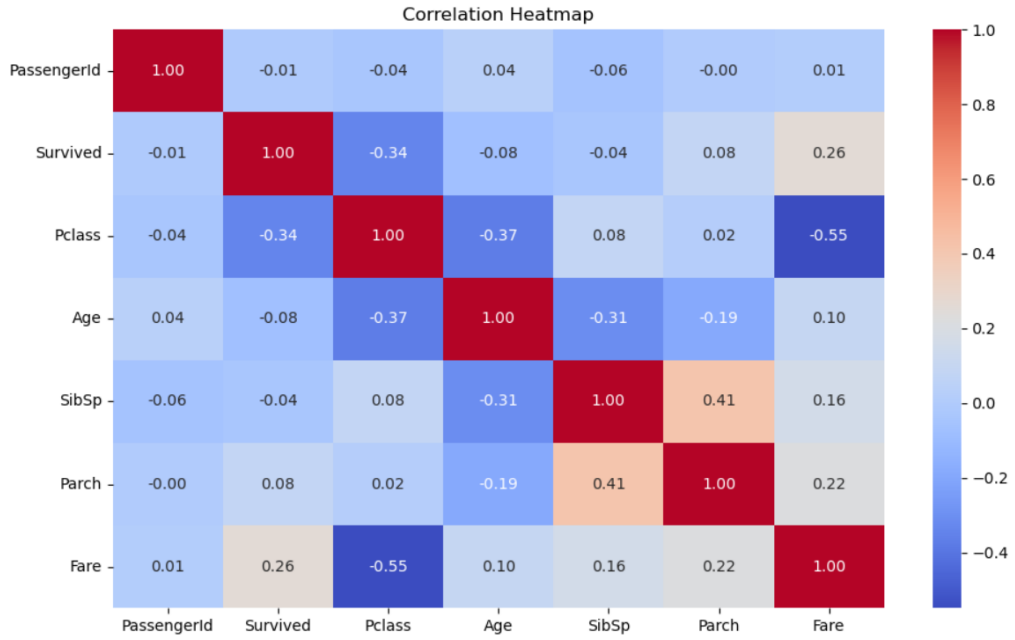
[16]:
```
# Step 3: Basic Info
print("\n=== Dataset Info ===")
print(df.info())
print("\n=== Summary Statistics ===")
print(df.describe(include='all'))
print("\n=== Missing Values ===")
print(df.isnull().sum())
```

```
=== Dataset 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

=== Summary Statistics ===
        PassengerId    Survived      Pclass                     Name   Sex  \
count    891.000000  891.000000  891.000000                      891   891
unique          NaN         NaN         NaN                      891     2
top             NaN         NaN         NaN  Braund, Mr. Owen Harris  male
freq            NaN         NaN         NaN                        1   577
mean     446.000000    0.383838    2.308642                      NaN   NaN
std      257.353842    0.486592    0.836071                      NaN   NaN
min        1.000000    0.000000    1.000000                      NaN   NaN
25%      223.500000    0.000000    2.000000                      NaN   NaN
50%      446.000000    0.000000    3.000000                      NaN   NaN
75%      668.500000    1.000000    3.000000                      NaN   NaN
max      891.000000    1.000000    3.000000                      NaN   NaN
```

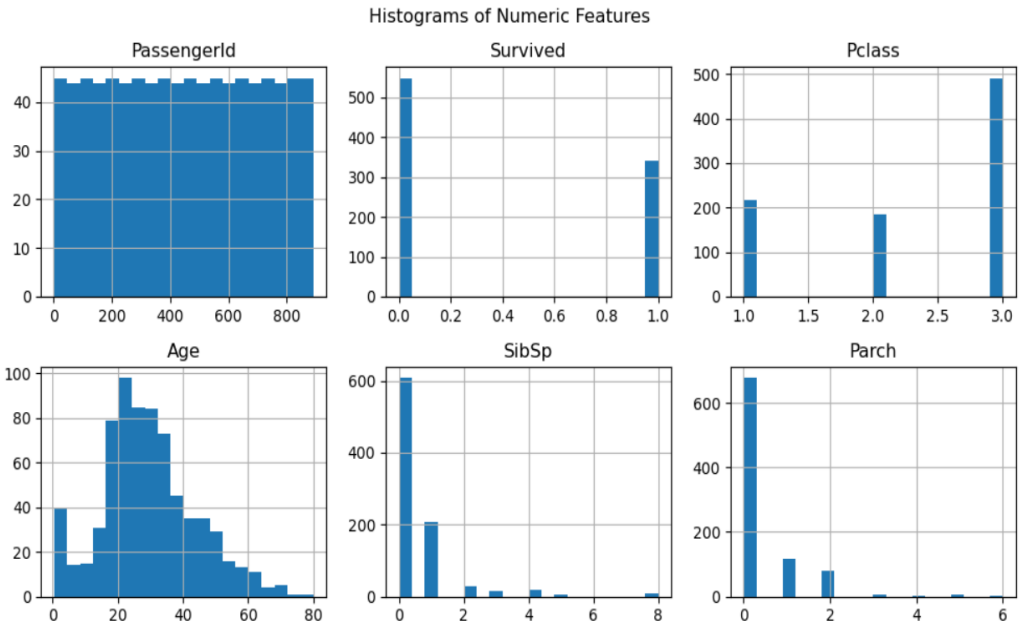
[17]:
```
# Step 4: Value Counts
print("\n=== Value Counts ===")
for col in df.select_dtypes(include='object').columns:
    print(f"\n{col}:\n{df[col].value_counts()}")
```

```
=== Value Counts ===

Name:
Name
Braund, Mr. Owen Harris                     1
Boulos, Mr. Hanna                           1
Frolicher-Stehli, Mr. Maxmillian            1
Gilinski, Mr. Eliezer                       1
Murdlin, Mr. Joseph                         1
                                           ..
Kelly, Miss. Anna Katherine "Annie Kate"    1
McCoy, Mr. Bernard                          1
Johnson, Mr. William Cahoone Jr             1
Keane, Miss. Nora A                         1
Dooley, Mr. Patrick                         1
Name: count, Length: 891, dtype: int64

Sex:
Sex
male      577
female    314
Name: count, dtype: int64

Ticket:
Ticket
347082      7
CA. 2343    7
1601        7
3101295     6
CA 2144     6
           ..
9234        1
19988       1
2693        1
PC 17612    1
370376      1
Name: count, Length: 681, dtype: int64
```

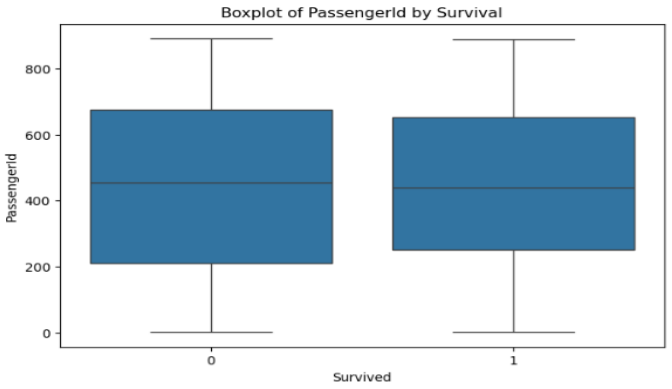
[18]:
```
# Continue analysis
plt.figure(figsize=(10, 6))
sns.heatmap(df.corr(numeric_only=True), annot=True, cmap='coolwarm', fmt=".2f")
plt.title("Correlation Heatmap")
plt.tight_layout()
plt.show()
```

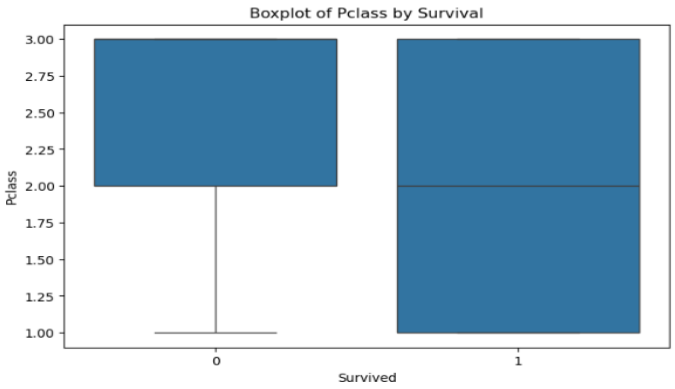


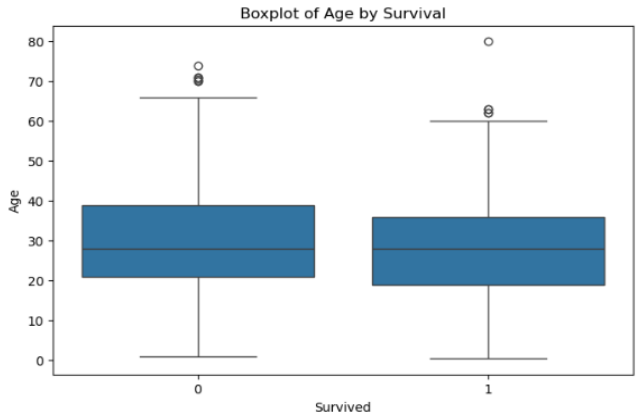
[19]:
```
# Step 6: Histograms
df.select_dtypes(include=['int64', 'float64']).hist(bins=20, figsize=(10, 8))
plt.suptitle("Histograms of Numeric Features")
plt.tight_layout()
plt.show()
```

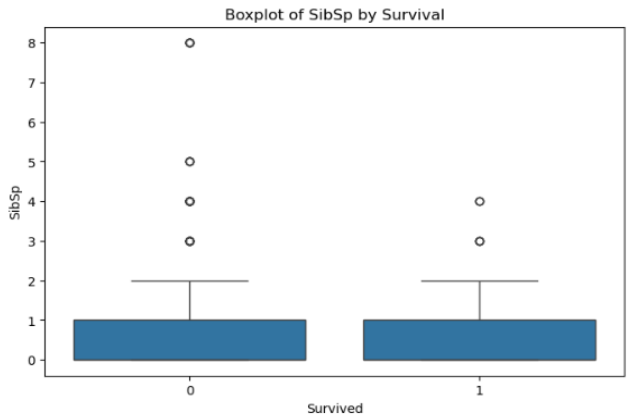


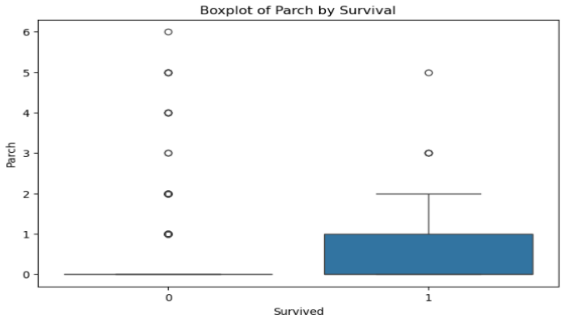
[20]:
```
# Step 7: Boxplots for key comparisons
numeric = df.select_dtypes(include=['int64', 'float64']).columns.tolist()
if 'Survived' in df.columns:
    for col in numeric:
        if col != 'Survived':
            plt.figure(figsize=(8, 5))
            sns.boxplot(x='Survived', y=col, data=df)
            plt.title(f"Boxplot of {col} by Survival")
            plt.show()
```

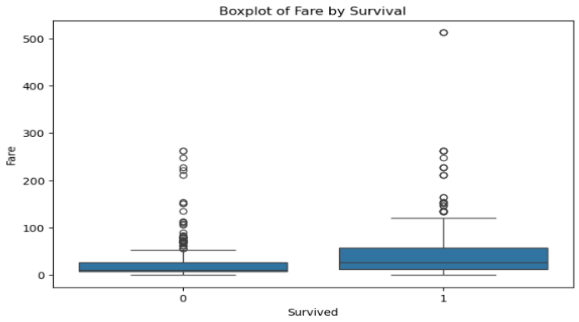




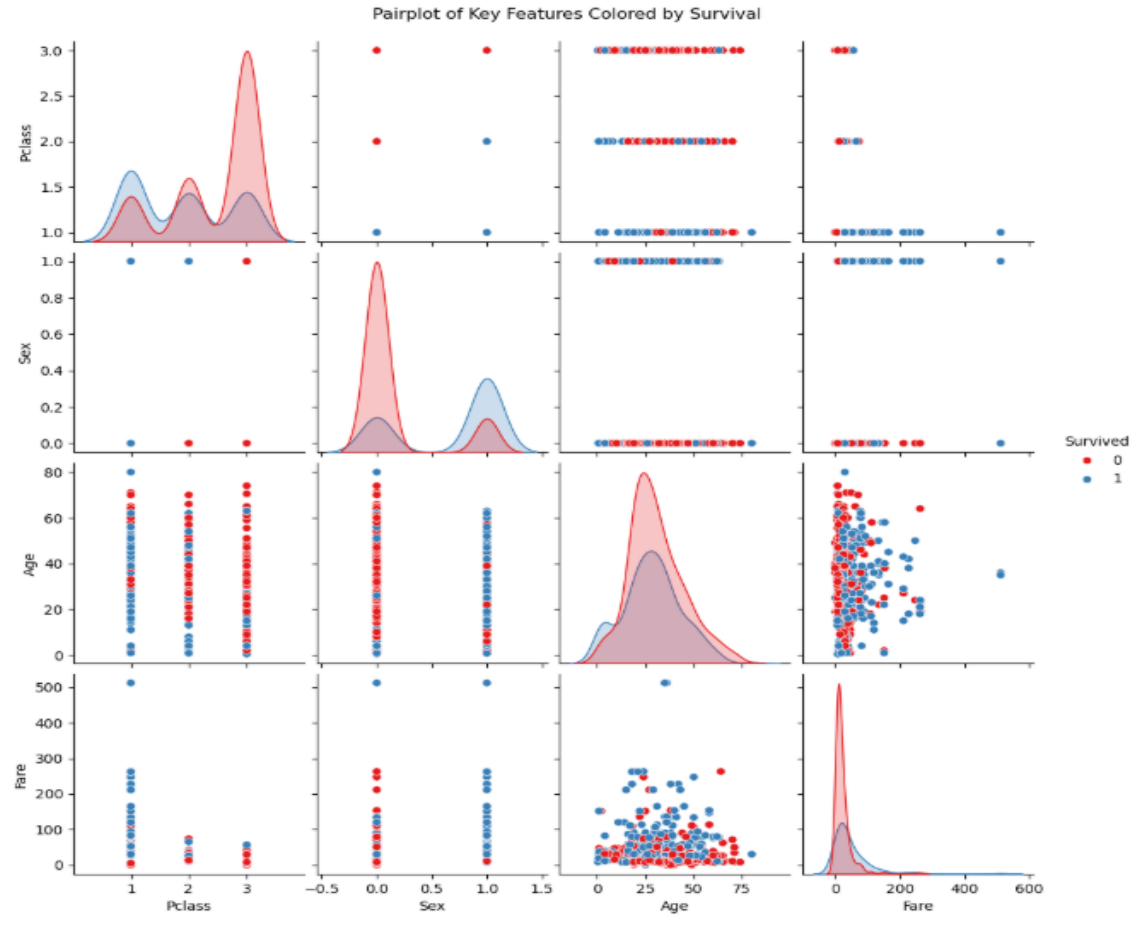








[21]:
```
# Step 8: Pairplot (if appropriate columns exist)
pairplot_cols = ['Survived', 'Pclass', 'Sex', 'Age', 'Fare']
existing_cols = [col for col in pairplot_cols if col in df.columns]
if set(['Survived', 'Age', 'Fare']).issubset(existing_cols):
    df_pair = df[existing_cols].dropna()
    if 'Sex' in df_pair.columns:
        df_pair['Sex'] = df_pair['Sex'].map({'male': 0, 'female': 1})
    sns.pairplot(df_pair, hue='Survived', palette='Set1')
    plt.suptitle("Pairplot of Key Features Colored by Survival", y=1.02)
    plt.show()
```



Summary:

1. Data Loading and Initial Exploration

- The dataset is read using pandas from a CSV file.
- Initial preview with .head() shows columns such as:
    - PassengerId, Survived, Pclass, Name, Sex, Age, SibSp, Parch, Ticket, Fare, Cabin, Embarked.

---

2. Basic Information and Statistics

- df.info() provides data types and non-null counts.
- df.describe(include='all') gives detailed stats for both numeric and categorical variables.
- **Missing Values Identified**:
    - Age: 177 missing
    - Cabin: ~687 missing
    - Embarked: 2 missing

---

3. Categorical Feature Distribution

- Used .value_counts() to summarize:
    - Sex: majority are male.
    - Embarked: most passengers boarded from 'S' (Southampton).
    - Pclass: more passengers in 3rd class.

---

4. Correlation Analysis

- A **correlation heatmap** was created to visualize relationships between numeric variables.
    - Strongest correlations:
        - Fare and Pclass (negatively correlated)
        - SibSp and Parch (moderate positive correlation)
        - Survived and Pclass, Fare, and Sex (after encoding)

---

5. Univariate Analysis

- **Histograms** for all numeric columns were plotted.
    - Skewness observed in Fare.
    - Age shows a bell-shaped curve, slightly skewed right.

◈ 6. Bivariate Analysis

- **Boxplots of numeric variables vs. Survived** were generated:
    - **Age vs. Survived**: younger passengers (especially children) had higher survival rates.
    - **Fare vs. Survived**: higher fares correlated with survival.
    - **Pclass vs. Survived** (via correlation or boxplot): 1st class passengers had the highest survival rate.

---

7. Pairplot Analysis

- A subset of variables (Survived, Pclass, Sex, Age, Fare) was selected for a seaborn.pairplot.
- **Sex column** was encoded: male = 0, female = 1.
- This helped to visually explore pairwise relationships:

Clear separability between survivors and non-survivors along features like Fare, Pclass, and Sex.

Insights Drawn from the Analysis

- **Gender is a strong indicator of survival**: females more likely to survive.
- **Passenger class matters**: higher class = higher chance of survival.
- **Young age favored survival**, especially children.
- **Fare** is positively related to survival (likely tied to class).
- **Embarkation point** had a mild effect, with passengers from Cherbourg having a slightly higher survival rate.