Titanic

Examining socio-demographic factors like age, class, and family size to understand their impact on survival rates. Through EDA, this project aims to uncover patterns that reveal the underlying factors contributing to passenger survival. The goal is to provide anuanced view of how different attributes influenced outcomes, offering insights into the historical event's human aspect.

In [1]:
```
import pandas as pd
import seaborn as sns
import numpy as np
import matplotlib.pyplot as plt
from IPython.display import Image, display
%matplotlib inline
import warnings

warnings.filterwarnings('ignore')
```

```
C:\Users\Dell\anaconda3\lib\site-packages\pandas\core\computation\expressions.py:21: Use
rWarning: Pandas requires version '2.8.4' or newer of 'numexpr' (version '2.8.3' current
ly installed).
  from pandas.core.computation.check import NUMEXPR_INSTALLED
C:\Users\Dell\anaconda3\lib\site-packages\pandas\core\arrays\masked.py:60: UserWarning:
Pandas requires version '1.3.6' or newer of 'bottleneck' (version '1.3.5' currently inst
alled).
  from pandas.core import (
```

In [2]:
```
# Read the datasets into seaborn DataFrame objects
titanic = sns.load_dataset ('titanic')
```

In [3]:
```
#Exploring Data Using pandas method
titanic.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 15 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   survived     891 non-null    int64
 1   pclass       891 non-null    int64
 2   sex          891 non-null    object
 3   age          714 non-null    float64
 4   sibsp        891 non-null    int64
 5   parch        891 non-null    int64
 6   fare         891 non-null    float64
 7   embarked     889 non-null    object
 8   class        891 non-null    category
 9   who          891 non-null    object
 10  adult_male   891 non-null    bool
 11  deck         203 non-null    category
 12  embark_town  889 non-null    object
 13  alive        891 non-null    object
 14  alone        891 non-null    bool
dtypes: bool(2), category(2), float64(2), int64(4), object(5)
memory usage: 80.7+ KB
```

In [4]:
```
titanic[titanic['deck'].isnull()]
```

Out[4]:

	survived	pclass	sex	age	sibsp	parch	fare	embarked	class	who	adult_male	deck	embarl
0	0	3	male	22.0	1	0	7.2500	S	Third	man	True	NaN	Southa

2	1	3	female	26.0	0	0	7.9250	S	Third	woman	False	NaN	Southa
4	0	3	male	35.0	0	0	8.0500	S	Third	man	True	NaN	Southa
5	0	3	male	NaN	0	0	8.4583	Q	Third	man	True	NaN	Quee
7	0	3	male	2.0	3	1	21.0750	S	Third	child	False	NaN	Southa
...	...	...	...	...	...	...	...	...	...	...	...	...	
884	0	3	male	25.0	0	0	7.0500	S	Third	man	True	NaN	Southa
885	0	3	female	39.0	0	5	29.1250	Q	Third	woman	False	NaN	Quee
886	0	2	male	27.0	0	0	13.0000	S	Second	man	True	NaN	Southa
888	0	3	female	NaN	1	2	23.4500	S	Third	woman	False	NaN	Southa
890	0	3	male	32.0	0	0	7.7500	Q	Third	man	True	NaN	Quee

688 rows × 15 columns

In [5]:
```
titanic['sex'].isnull()
```

Out[5]:
```
0      False
1      False
2      False
3      False
4      False
       ...
886    False
887    False
888    False
889    False
890    False
Name: sex, Length: 891, dtype: bool
```

In [6]:
```
titanic['embark_town'].unique()
```

Out[6]:
```
array(['Southampton', 'Cherbourg', 'Queenstown', nan], dtype=object)
```

In [7]:
```
titanic[titanic['embark_town'].isnull()]
```

Out[7]:

	survived	pclass	sex	age	sibsp	parch	fare	embarked	class	who	adult_male	deck	embark_town
61	1	1	female	38.0	0	0	80.0	NaN	First	woman	False	B	NaN
829	1	1	female	62.0	0	0	80.0	NaN	First	woman	False	B	NaN

In [8]:
```
titanic['embark_town']
```

Out[8]:
```
0      Southampton
1        Cherbourg
2      Southampton
3      Southampton
4      Southampton
          ...
886    Southampton
887    Southampton
888    Southampton
889      Cherbourg
890     Queenstown
Name: embark_town, Length: 891, dtype: object
```

In [9]:
```
titanic.isnull().sum()
```

Out[9]:
```
survived         0
pclass           0
sex              0
age            177
sibsp            0
parch            0
fare             0
embarked         2
class            0
who              0
adult_male       0
deck           688
embark_town      2
alive            0
alone            0
dtype: int64
```

In [10]:
```
titanic.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 15 columns):
 #   Column       Non-Null Count  Dtype   
---  ------       --------------  -----   
 0   survived     891 non-null    int64   
 1   pclass       891 non-null    int64   
 2   sex          891 non-null    object  
 3   age          714 non-null    float64 
 4   sibsp        891 non-null    int64   
 5   parch        891 non-null    int64   
 6   fare         891 non-null    float64 
 7   embarked     889 non-null    object  
 8   class        891 non-null    category
 9   who          891 non-null    object  
 10  adult_male   891 non-null    bool    
 11  deck         203 non-null    category
 12  embark_town  889 non-null    object  
 13  alive        891 non-null    object  
 14  alone        891 non-null    bool    
dtypes: bool(2), category(2), float64(2), int64(4), object(5)
memory usage: 80.7+ KB
```

In [11]:
```
titanic.loc[:,('alone','alive')]
```

Out[11]:

	alone	alive
0	False	no
1	False	yes
2	True	yes
3	False	yes
4	True	no
...	...	...
886	True	no
887	True	yes
888	False	no
889	True	yes
890	True	no

891 rows × 2 columns

Handling with missing values

In [12]:
```
titanic.isnull().sum()
```

Out[12]:
```
survived         0
pclass           0
sex              0
age            177
sibsp            0
parch            0
fare             0
embarked         2
class            0
who              0
adult_male       0
deck           688
embark_town      2
alive            0
alone            0
dtype: int64
```

In [13]:
```
titanic.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 15 columns):
 #   Column       Non-Null Count  Dtype   
---  ------       --------------  -----   
 0   survived     891 non-null    int64   
 1   pclass       891 non-null    int64   
 2   sex          891 non-null    object  
 3   age          714 non-null    float64 
 4   sibsp        891 non-null    int64   
 5   parch        891 non-null    int64   
 6   fare         891 non-null    float64 
 7   embarked     889 non-null    object  
 8   class        891 non-null    category
 9   who          891 non-null    object  
 10  adult_male   891 non-null    bool    
 11  deck         203 non-null    category
 12  embark_town  889 non-null    object  
 13  alive        891 non-null    object  
 14  alone        891 non-null    bool    
dtypes: bool(2), category(2), float64(2), int64(4), object(5)
memory usage: 80.7+ KB
```

In [14]:
```
pd.DataFrame(titanic.dtypes)
```

Out[14]:

	0
survived	int64
pclass	int64
sex	object
age	float64
sibsp	int64
parch	int64
fare	float64

embarked	object
class	category
who	object
adult_male	bool
deck	category
embark_town	object
alive	object
alone	bool

In [15]:
```
titanic.dtypes
```

Out[15]:
```
survived          int64
pclass            int64
sex              object
age             float64
sibsp             int64
parch             int64
fare            float64
embarked         object
class          category
who              object
adult_male         bool
deck           category
embark_town      object
alive            object
alone              bool
dtype: object
```

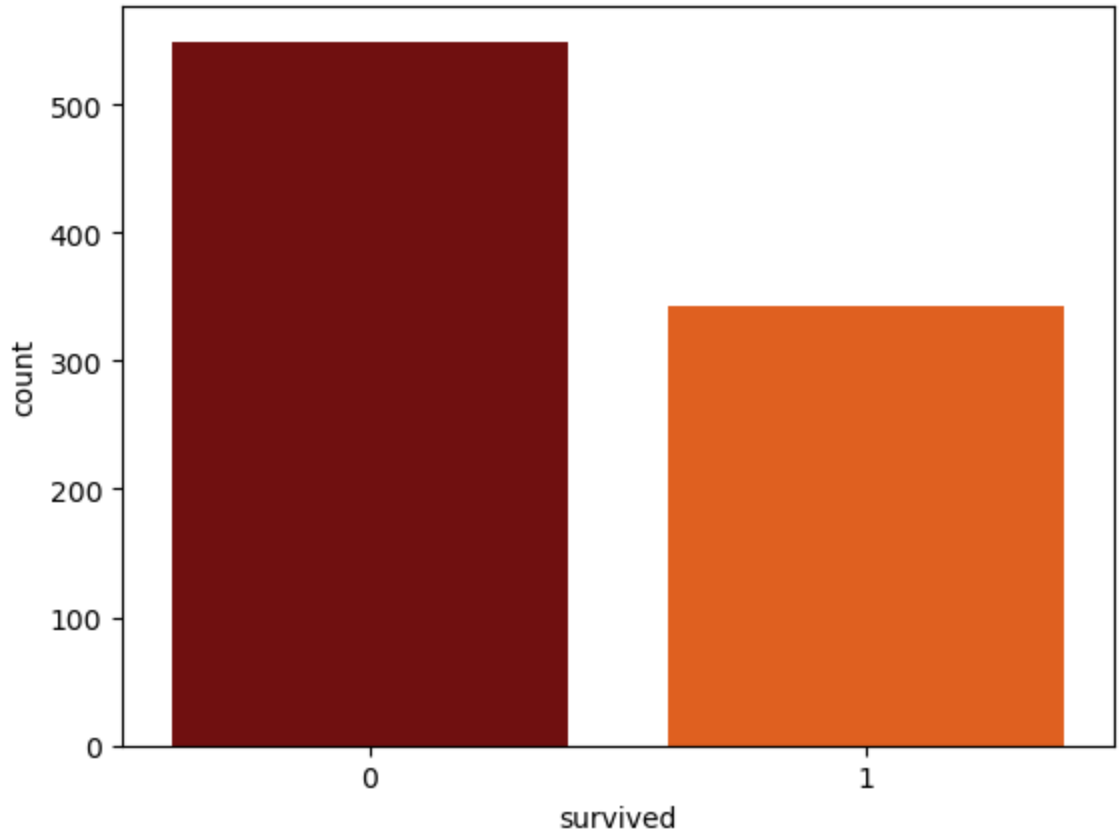
In [16]:
```
titanic.columns
```

Out[16]:
```
Index(['survived', 'pclass', 'sex', 'age', 'sibsp', 'parch', 'fare',
       'embarked', 'class', 'who', 'adult_male', 'deck', 'embark_town',
       'alive', 'alone'],
      dtype='object')
```

In [17]:
```
titanic.isnull().sum()
```

Out[17]:
```
survived         0
pclass           0
sex              0
age            177
sibsp            0
parch            0
fare             0
embarked         2
class            0
who              0
adult_male       0
deck           688
embark_town      2
alive            0
alone            0
dtype: int64
```

In [18]:
```
sns.countplot(x ='survived' , data=titanic ,palette = 'gist_heat')
plt.show()
```



In [19]:
```
titanic['sex'].value_counts()
```

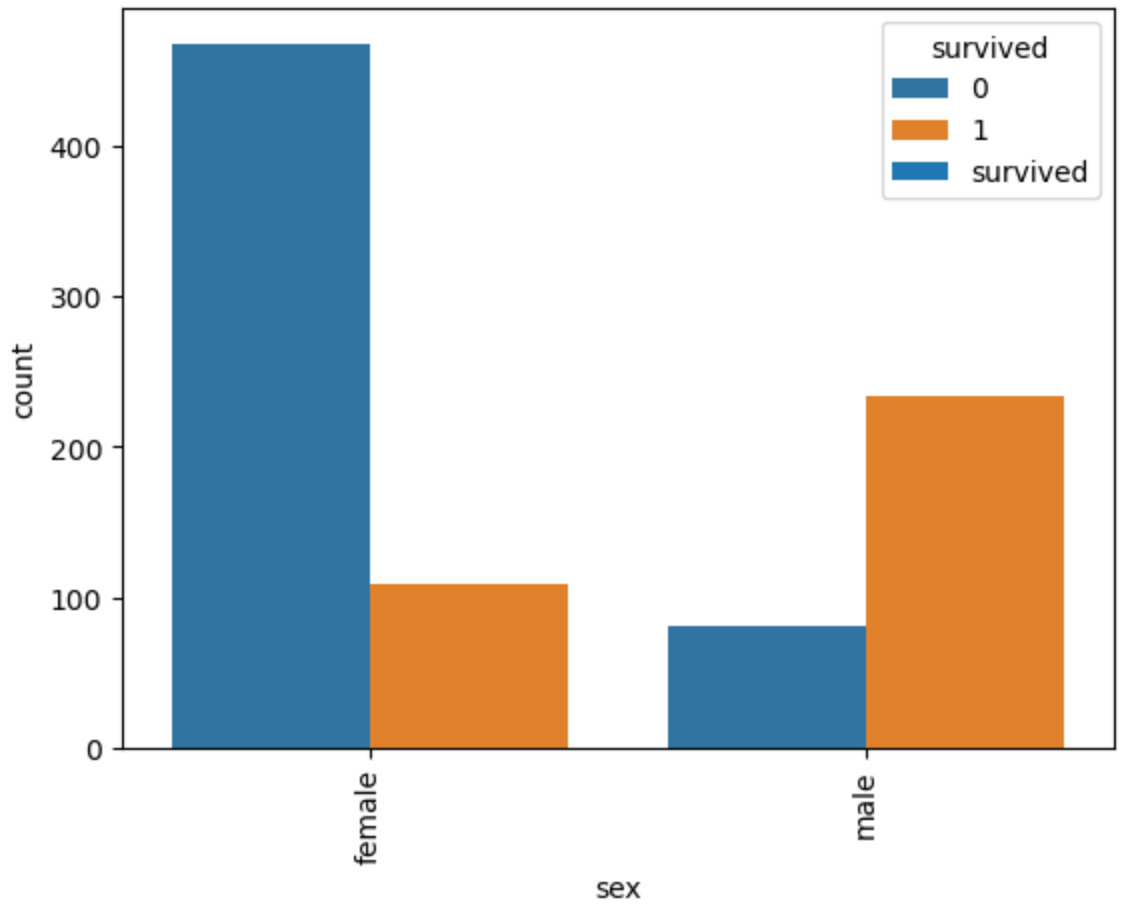
Out[19]:
```
sex
male      577
female    314
Name: count, dtype: int64
```

In [20]:
```
titanic.groupby(['sex','survived'])['survived'].count()
```

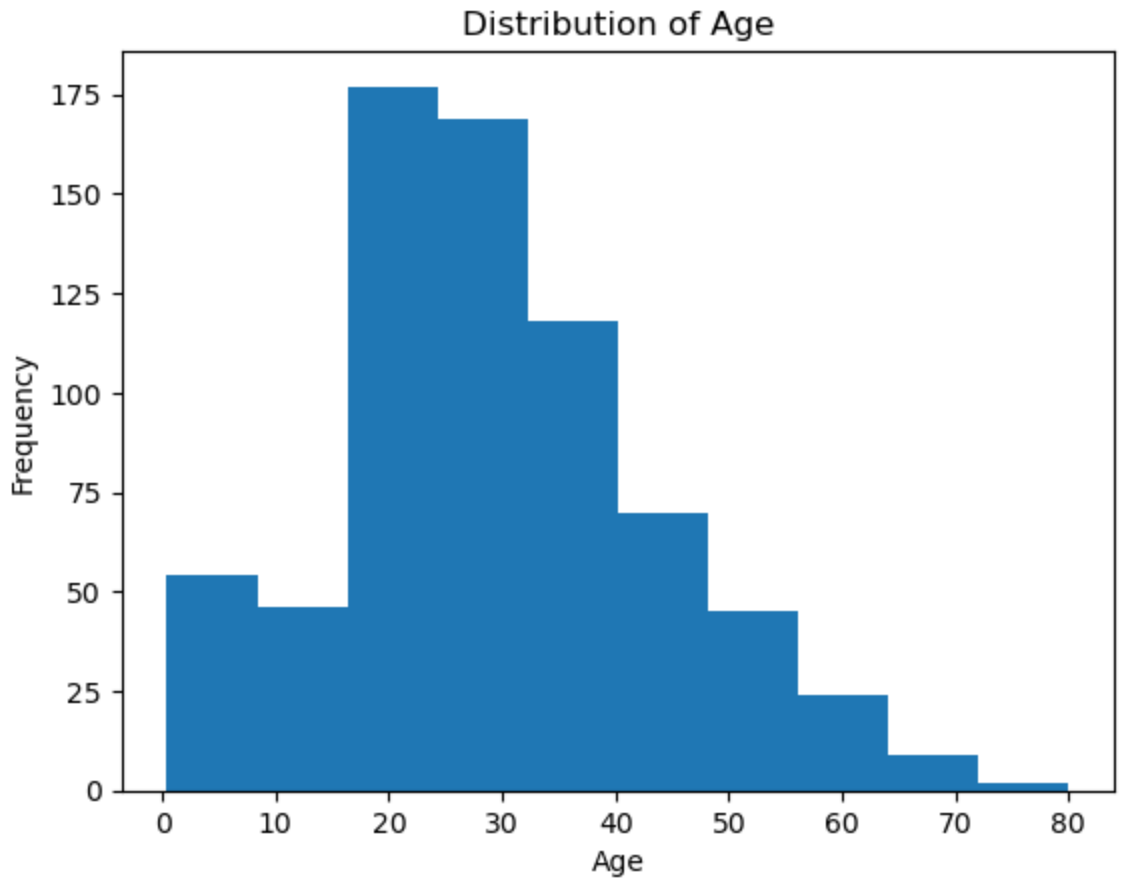
Out[20]:
```
sex     survived
female  0            81
        1           233
male    0           468
        1           109
Name: survived, dtype: int64
```

It is clear that `233` **female** survived out of `314`. And out of `577` **male** `109` survived. The survival ratio of female is much greater than that of male. It can be seen clearly in following graph

In [21]:
```
titanic[['sex','survived']].groupby(['sex']).mean().plot.bar()
sns.countplot(x='sex',hue='survived',data=titanic)
plt.show()
```



In [22]:
```
plt.hist(titanic['age'], bins=10)
plt.xlabel('Age')
plt.ylabel('Frequency')
plt.title('Distribution of Age')
plt.show()
```

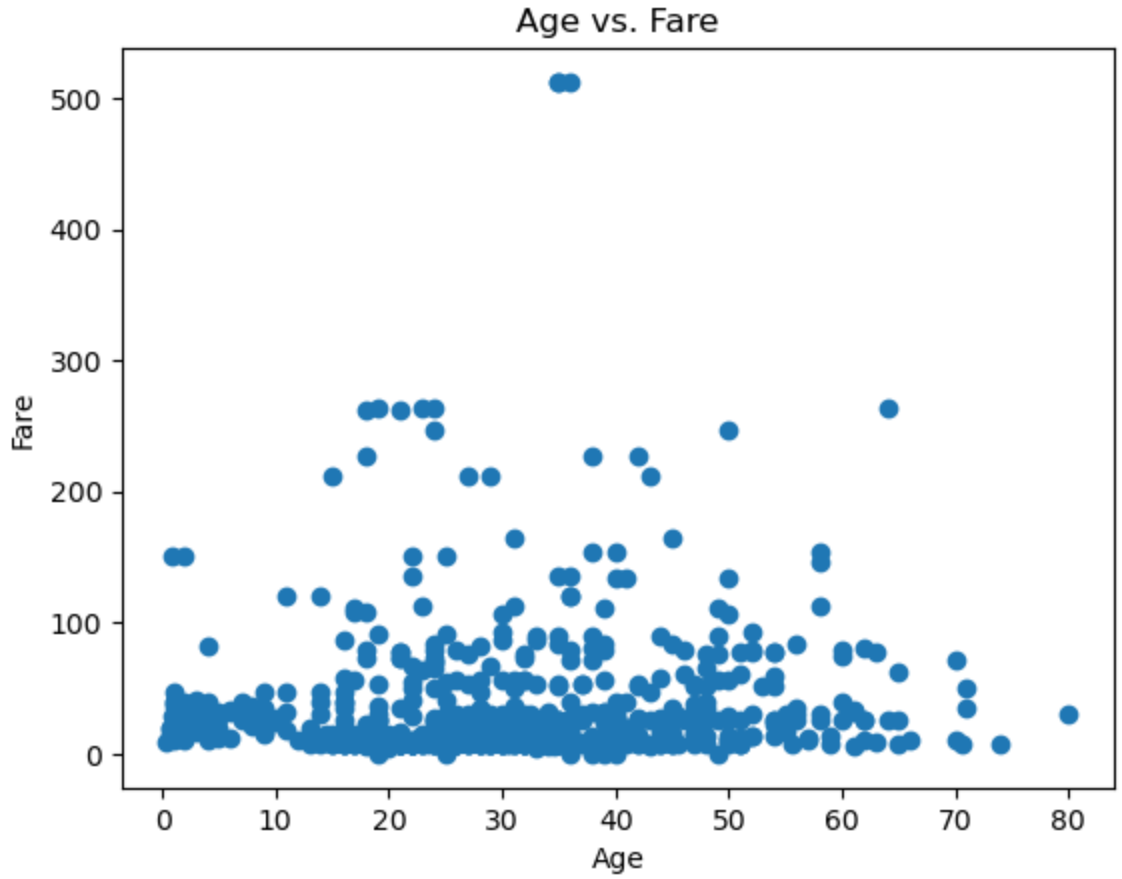


As shown in this graph the max number of passanger is between age range `18 to 28`.

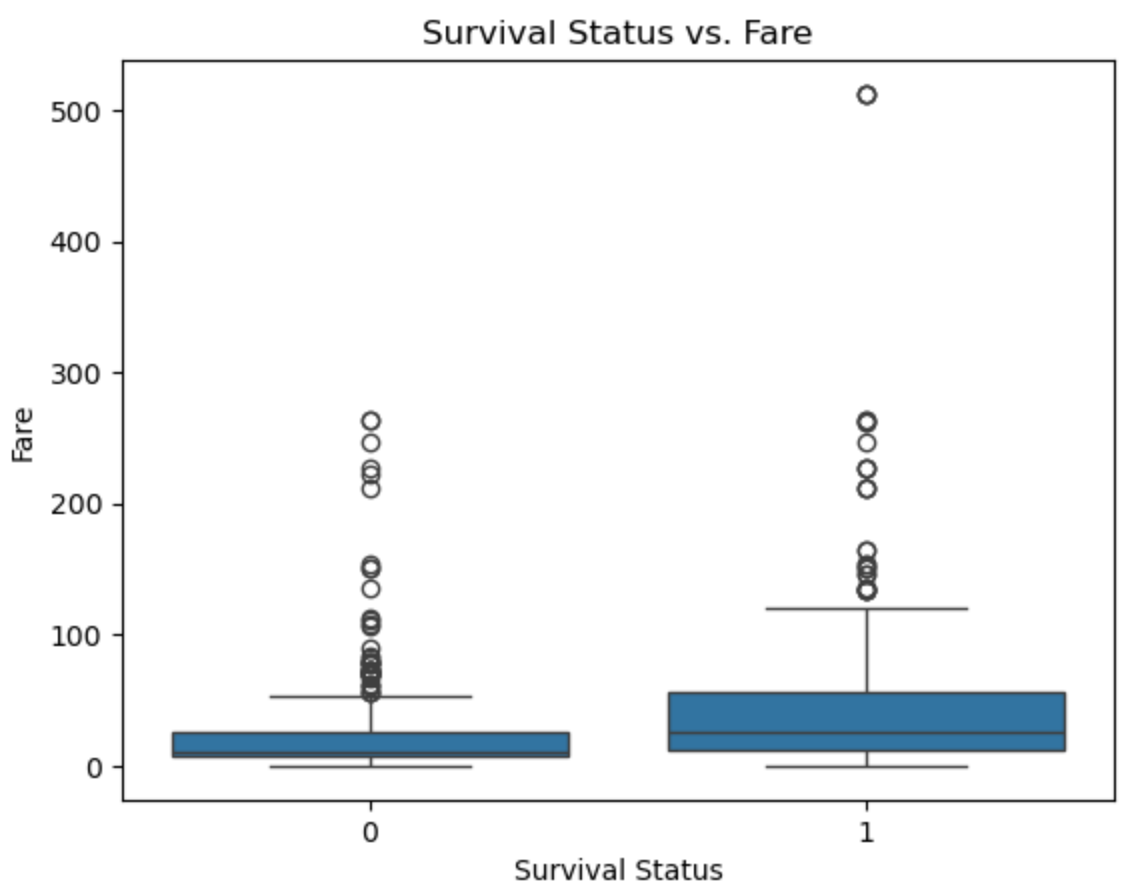
In [23]:
```
titanic.columns
```

Out[23]:
```
Index(['survived', 'pclass', 'sex', 'age', 'sibsp', 'parch', 'fare',
       'embarked', 'class', 'who', 'adult_male', 'deck', 'embark_town',
       'alive', 'alone'],
      dtype='object')
```

In [24]:
```
# Scatter plot
plt.scatter(titanic['age'], titanic['fare'])
plt.xlabel('Age')
plt.ylabel('Fare')
plt.title('Age vs. Fare')
plt.show()
```

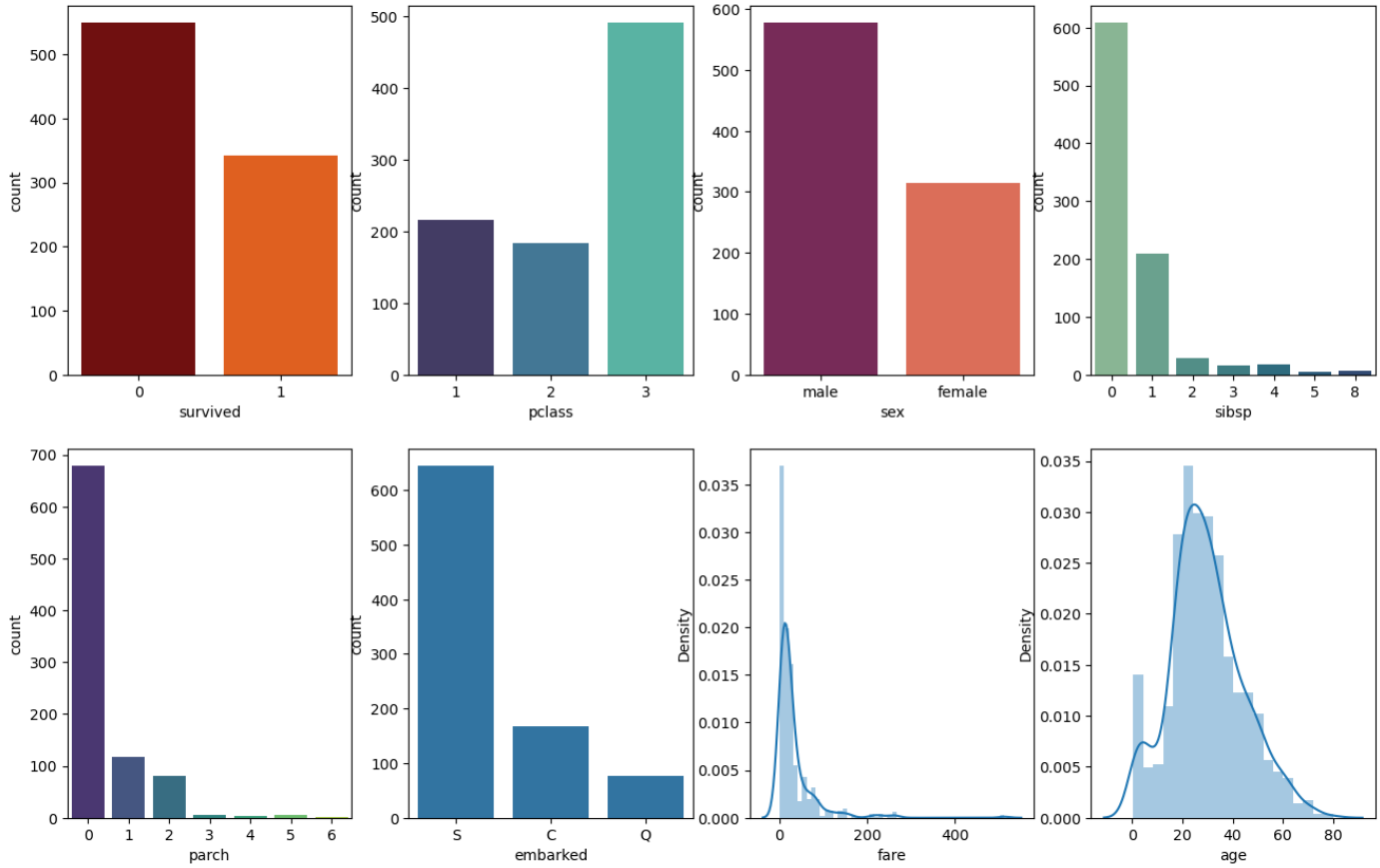


In [25]:
```
# Box plot
sns.boxplot(x=titanic['survived'], y=titanic['fare'])
plt.xlabel('Survival Status')
plt.ylabel('Fare')
plt.title('Survival Status vs. Fare')
plt.show()
```



In [58]:

```
fig, axes = plt.subplots(2, 4, figsize=(16, 10))
sns.countplot(x='survived',data=titanic,ax=axes[0,0],palette = 'gist_heat')
sns.countplot(x='pclass',data=titanic,ax=axes[0,1],palette="mako")
sns.countplot(x='sex',data=titanic,ax=axes[0,2],palette='rocket')
sns.countplot(x='sibsp',data=titanic,ax=axes[0,3],palette='crest')
sns.countplot(x='parch',data=titanic,ax=axes[1,0],palette='viridis')
sns.countplot(x='embarked',data=titanic,ax=axes[1,1])
sns.distplot(titanic['fare'], kde=True,ax=axes[1,2])
sns.distplot(titanic['age'].dropna(),kde=True,ax=axes[1,3])
plt.show()
```



- We can clearly see that male survial rates is around 20% where as female survial rate is about 75% which suggests that gender has a strong relationship with the survival rates.
- There is also a clear relationship between Pclass and the survival by referring to first plot below. Passengers on Pclass1 had a better survial rate of approx 60% whereas passengers on pclass3 had the worst survial rate of approx 22%
- There is also a marginal relationship between the fare and survial rate.
- I have quantified the above relationships further in the last statsical modelling section

Conclusion:

Based on the exploratory data analysis, we can summarize the key findings, insights, and potential areas for further investigation. This could include patterns, trends, outliers, or relationships observed during the analysis.