Prodigy Infotech Internship by chaitanya gadekar

Task 2 : Exploratory Data Analysis - Titanic Dataset

Problem Statement

About the Dataset

The sinking of Titanic is one of the most notorious shipwrecks in the history. In 1912, during her voyage, the Titanic sank after colliding with an iceberg, killing 1502 out of 2224 passengers and crew.

Objective of the task:

1. Understand the Dataset & cleanup (if required).
1. Perform Exploratory Data Analysis.

link to the Dataset : https://www.kaggle.com/datasets/yasserh/titanic-dataset

Data Preparation

```python
# Importing Libraries
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

import warnings
warnings.filterwarnings('ignore')
```

```python
# Loading the Dataset
titanic = pd.read_csv('Titanic-Dataset.csv')
titanic
```

```
     PassengerId  Survived  Pclass  \
0              1         0       3
1              2         1       1
2              3         1       3
3              4         1       1
4              5         0       3
..           ...       ...     ...
886          887         0       2
887          888         1       1
888          889         0       3
889          890         1       1
890          891         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
..                                                 ...     ...   ...    ...
886                              Montvila, Rev. Juozas    male  27.0      0
887                       Graham, Miss. Margaret Edith  female  19.0      0
888           Johnston, Miss. Catherine Helen "Carrie"  female   NaN      1
889                              Behr, Mr. Karl Howell    male  26.0      0
890                                Dooley, Mr. Patrick    male  32.0      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
..     ...               ...      ...   ...      ...
886      0            211536  13.0000   NaN        S
887      0            112053  30.0000   B42        S
888      2        W./C. 6607  23.4500   NaN        S
889      0            111369  30.0000  C148        C
890      0            370376   7.7500   NaN        Q

[891 rows x 12 columns]
```

```python
# Showing first 5 rows
titanic.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
```

```
# Showing last 5 rows
titanic.tail()
```

```
     PassengerId  Survived  Pclass                                      Name
\
886          887         0       2                     Montvila, Rev. Juozas
887          888         1       1              Graham, Miss. Margaret Edith
888          889         0       3  Johnston, Miss. Catherine Helen "Carrie"
889          890         1       1                     Behr, Mr. Karl Howell
890          891         0       3                       Dooley, Mr. Patrick

        Sex   Age  SibSp  Parch      Ticket   Fare Cabin Embarked
886    male  27.0      0      0      211536  13.00   NaN        S
887  female  19.0      0      0      112053  30.00   B42        S
888  female   NaN      1      2  W./C. 6607  23.45   NaN        S
889    male  26.0      0      0      111369  30.00  C148        C
890    male  32.0      0      0      370376   7.75   NaN        Q
```

Basic Understanding of the Dataset

```
# Showing no. of rows and columns of dataset
titanic.shape
```

```
(891, 12)
```

This dataset contains 891 rows and 12 columns.

```
# checking for columns
titanic.columns
```

```
Index(['PassengerId', 'Survived', 'Pclass', 'Name', 'Sex', 'Age', 'SibSp',
       'Parch', 'Ticket', 'Fare', 'Cabin', 'Embarked'],
      dtype='object')
```

Information about Columns

1. PassengerId: unique id number to each passenger
1. Survived: passenger survive(1) or died(0)
2. Pclass: passenger class
3. Name: name
4. Sex: gender of passenger
5. Age: age of passenger
6. SibSp: number of siblings/spouses
7. Parch: number of parents/children
8. Ticket: ticket number
9. Fare: amount of money spent on ticket
10. Cabin: cabin category
11. Embarked: port where passenger embarked (C = Cherbourg, Q = Queenstown, S = Southampton)

```
# Checking for data types
titanic.dtypes
```

```
PassengerId      int64
Survived         int64
Pclass           int64
Name            object
Sex             object
Age            float64
SibSp            int64
Parch            int64
Ticket          object
Fare           float64
Cabin           object
Embarked        object
dtype: object
```

```
# Showing inforamation about the dataset
titanic.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
```

From the above infomation we can see that

1. There are 891 rows and 12 columns.
1. There are total of 5 columns having categorical data.
2. Rest of the columns having int and float data types.

Data Preprocessing and Data Cleaning

Handling Duplicated Values

```
# checking for duplicated values
titanic.duplicated().sum()
```

```
0
```

No duplicates values found

Null Values Treatment

```
# checking for null values
nv = titanic.isna().sum().sort_values(ascending=False)
nv = nv[nv>0]
nv
```

```
Cabin       687
Age         177
Embarked      2
dtype: int64
```

```
# Cheecking what percentage column contain missing values
titanic.isnull().sum().sort_values(ascending=False)*100/len(titanic)
```

```
Cabin          77.104377
Age            19.865320
Embarked        0.224467
PassengerId     0.000000
Survived        0.000000
Pclass          0.000000
Name            0.000000
Sex             0.000000
```

```
SibSp           0.000000
Parch           0.000000
Ticket          0.000000
Fare            0.000000
dtype: float64
```

```
# Since Cabin Column has more than 75 % null values .So , we will drop this column
titanic.drop(columns = 'Cabin', axis = 1, inplace = True)
titanic.columns
```

```
Index(['PassengerId', 'Survived', 'Pclass', 'Name', 'Sex', 'Age', 'SibSp',
       'Parch', 'Ticket', 'Fare', 'Embarked'],
      dtype='object')
```

```
# Filling Null Values in Age column with mean values of age column
titanic['Age'].fillna(titanic['Age'].mean(),inplace=True)

# filling null values in Embarked Column with mode values of embarked column
titanic['Embarked'].fillna(titanic['Embarked'].mode()[0],inplace=True)
```

```
# checking for null values
titanic.isna().sum()
```

```
PassengerId    0
Survived       0
Pclass         0
Name           0
Sex            0
Age            0
SibSp          0
Parch          0
Ticket         0
Fare           0
Embarked       0
dtype: int64
```

Checking for unique values

```
# Finding no. of unique values in each column of dataset
titanic[['PassengerId', 'Survived', 'Pclass', 'Name', 'Sex', 'Age', 'SibSp',
       'Parch', 'Ticket', 'Fare', 'Embarked']].nunique().sort_values()
```

```
Survived         2
Sex              2
Pclass           3
Embarked         3
SibSp            7
Parch            7
Age             89
Fare           248
Ticket         681
```

```
PassengerId    891
Name           891
dtype: int64
```

Showing unique values of different columns

```
titanic['Survived'].unique()
```

```
array([0, 1], dtype=int64)
```

```
titanic['Sex'].unique()
```

```
array(['male', 'female'], dtype=object)
```

```
titanic['Pclass'].unique()
```

```
array([3, 1, 2], dtype=int64)
```

```
titanic['SibSp'].unique()
```

```
array([1, 0, 3, 4, 2, 5, 8], dtype=int64)
```

```
titanic['Parch'].unique()
```

```
array([0, 1, 2, 5, 3, 4, 6], dtype=int64)
```

```
titanic['Embarked'].unique()
```

```
array(['S', 'C', 'Q'], dtype=object)
```

Dropping Some Unnecessary Columns

There are 3 columns i.e.. 'PassengerId' , 'Name' , 'Ticket' are unnecessary columns which have no use in data modelling . So, we will drop these 3 columns

```
titanic.drop(columns=['PassengerId','Name','Ticket'],axis=1,inplace=True)
titanic.columns
```

```
Index(['Survived', 'Pclass', 'Sex', 'Age', 'SibSp', 'Parch', 'Fare',
       'Embarked'],
      dtype='object')
```

Descriptive Statistical Analysis

```
# descriptive statistical analysis of dataset
titanic.describe()
```

```
         Survived      Pclass         Age       SibSp       Parch        Fare
count  891.000000  891.000000  891.000000  891.000000  891.000000  891.000000
mean     0.383838    2.308642   29.699118    0.523008    0.381594   32.204208
std      0.486592    0.836071   13.002015    1.102743    0.806057   49.693429
min      0.000000    1.000000    0.420000    0.000000    0.000000    0.000000
```

```
25%      0.000000    2.000000   22.000000    0.000000    0.000000    7.910400
50%      0.000000    3.000000   29.699118    0.000000    0.000000   14.454200
75%      1.000000    3.000000   35.000000    1.000000    0.000000   31.000000
max      1.000000    3.000000   80.000000    8.000000    6.000000  512.329200
```

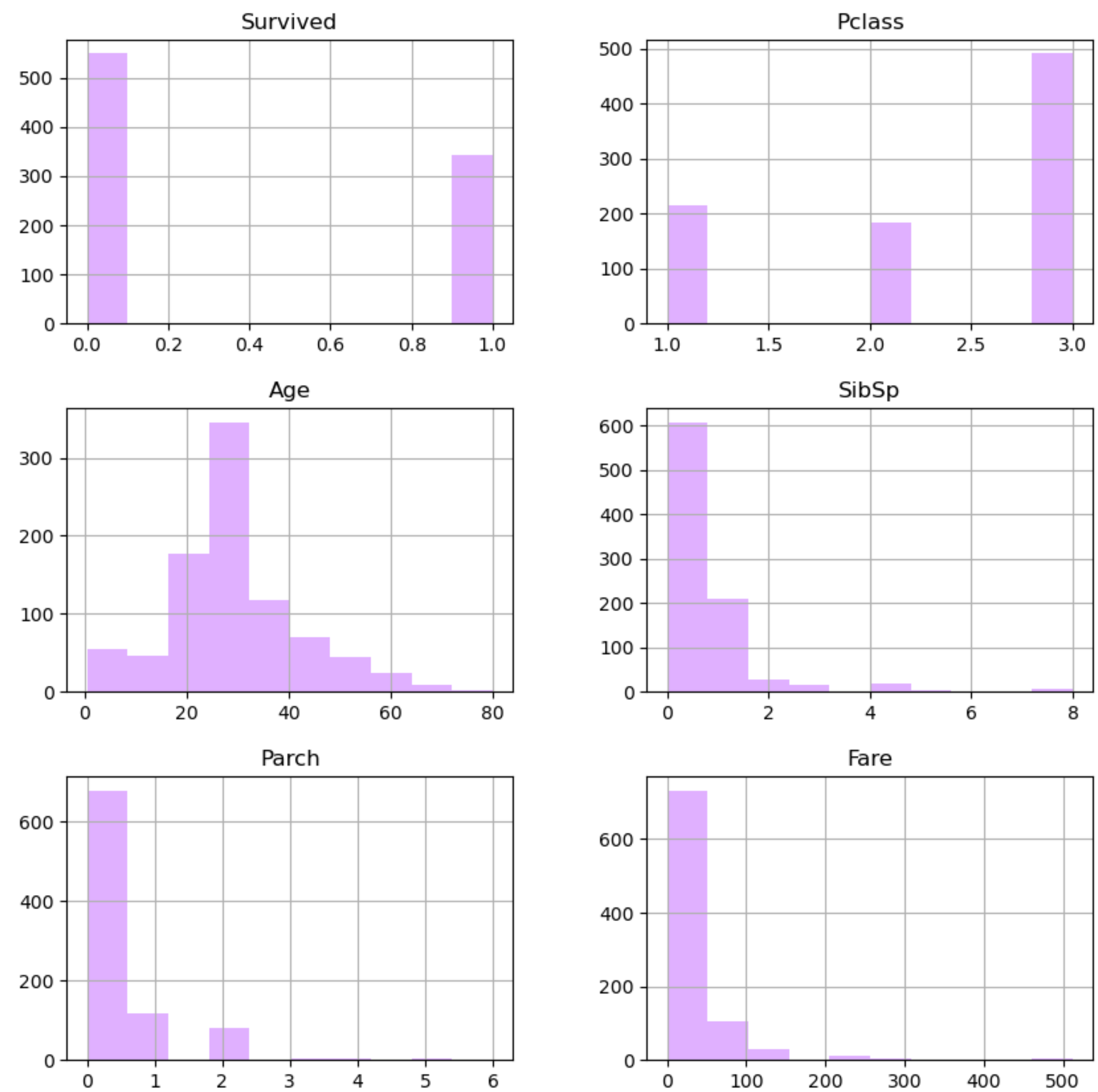
```
# Statistical Analysis about categorical columns
titanic.describe(include='O')
```

```
         Sex Embarked
count    891      891
unique     2        3
top     male        S
freq     577      646
```

Data Visualization

Plotting HistPlot

```
# Plotting Histplot for Dataset
titanic.hist(figsize=(10,10),color='#E0B0FF')
plt.show()
```



Insights:

1. Survived Column shows a binomial distribution, i.e. 1 for survived and 0 for died.
1. Pclass Column shows a trinomial distribution, i.e. 1 for first class and 2 for second class and 3 for third class.
2. Age distribution mostly lies between 20-40 age group.
3. Fare normally lies between 0 to 100.

Plotting CountPlot for categorical columns

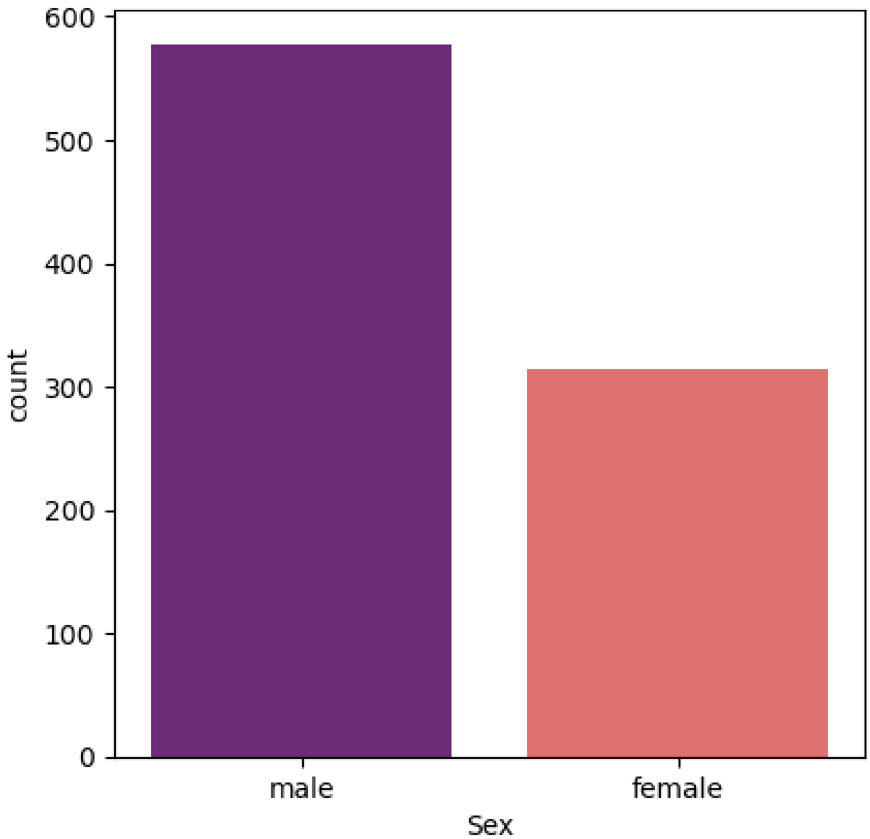
```
cat_cols = titanic.select_dtypes(include='object').columns
cat_cols
```

```
Index(['Sex', 'Embarked'], dtype='object')
```

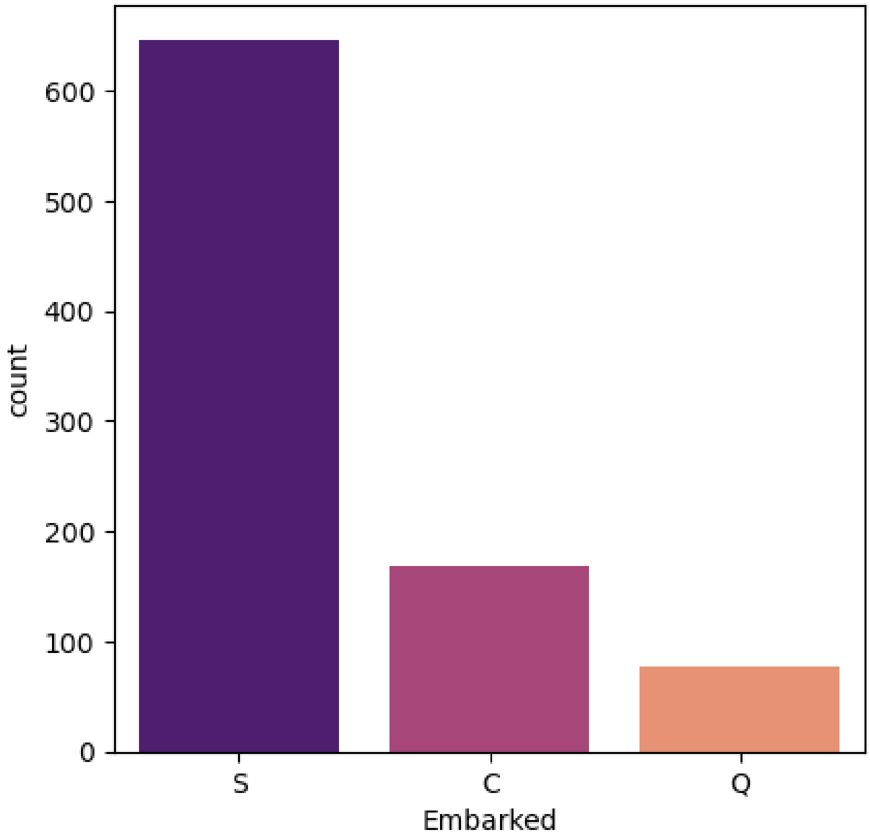
```
for i in cat_cols:
    print(titanic[i].value_counts())
```

```
    plt.figure(figsize=(5,5))
    sns.countplot(x=titanic[i],palette='magma')
    plt.xlabel(i)
    plt.show()
```

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




```
S    646
C    168
Q     77
Name: Embarked, dtype: int64
```



Insights:

1. The 1st plot clearly shows Male population is more than female population.
1. The 2nd plot clearly shows that most of the people choose Southampton as their port of embarkation.

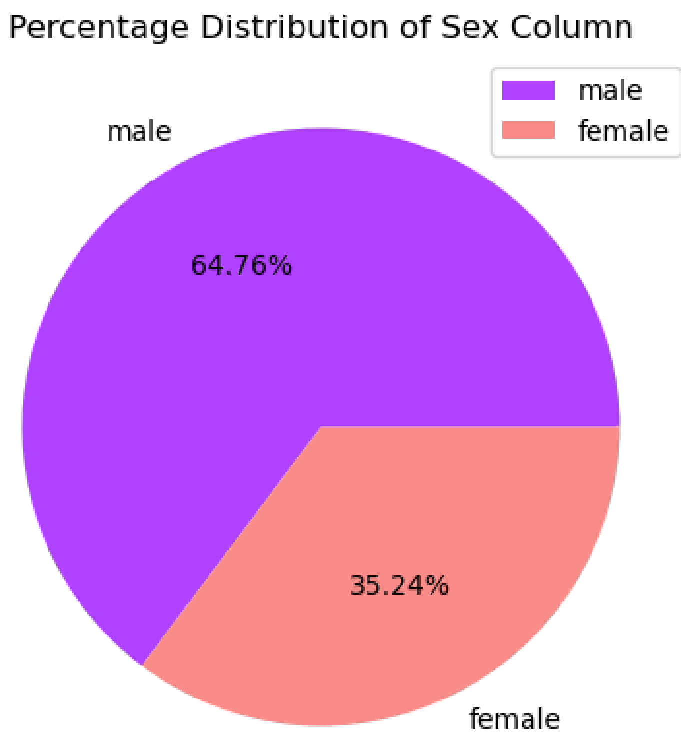
Sex Column

```
d1 = titanic['Sex'].value_counts()
d1
```

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

Percentage Distribution of Sex

```
# Plotting Percantage Distribution of Sex Column
plt.figure(figsize=(5,5))
plt.pie(d1.values,labels=d1.index,autopct='%.2f%%',colors=['#B041FF','#F98B88
'])
plt.title('Percentage Distribution of Sex Column')
plt.legend()
plt.show()
```

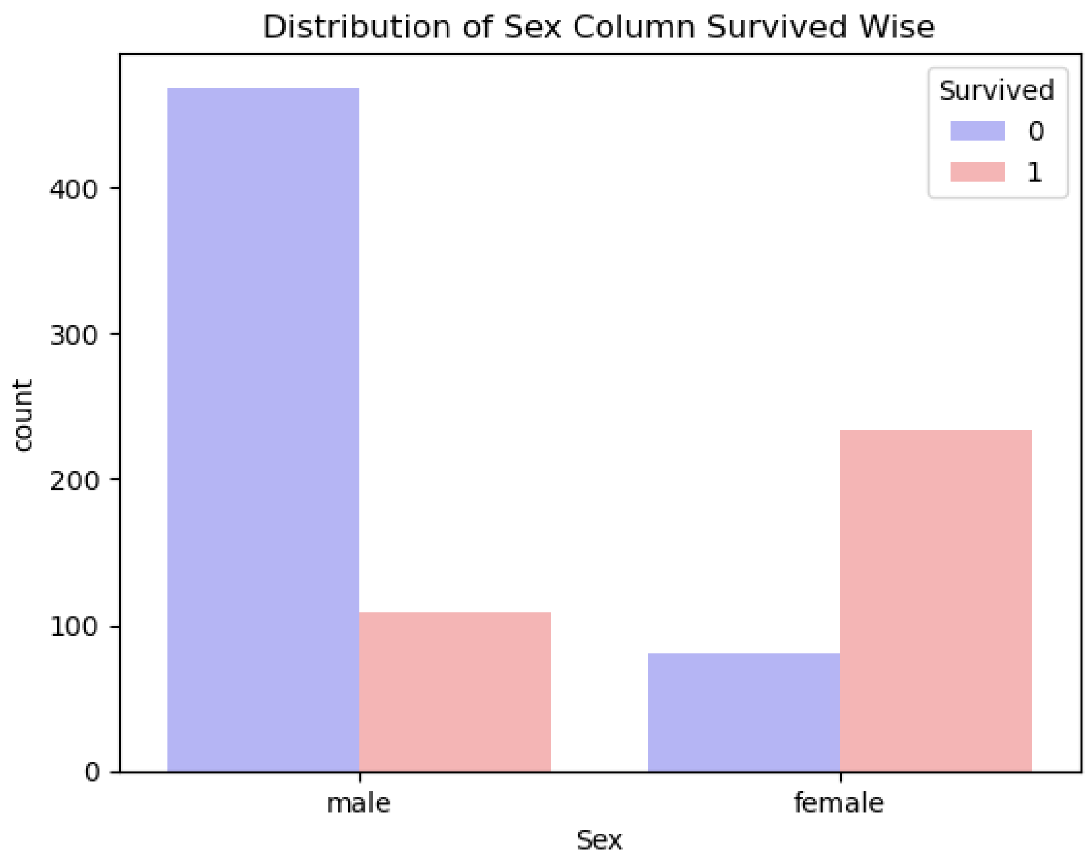


Insight:

The proportion of male population is more than female population

Distribuion of Sex Column by Survived

```
# Showing Distribution of Sex Column Survived Wise
sns.countplot(x=titanic['Sex'],hue=titanic['Survived'],palette = 'bwr') # In
Sex (0 represents died and 1 represents survived)
plt.title('Distribution of Sex Column Survived Wise')
plt.show()
```

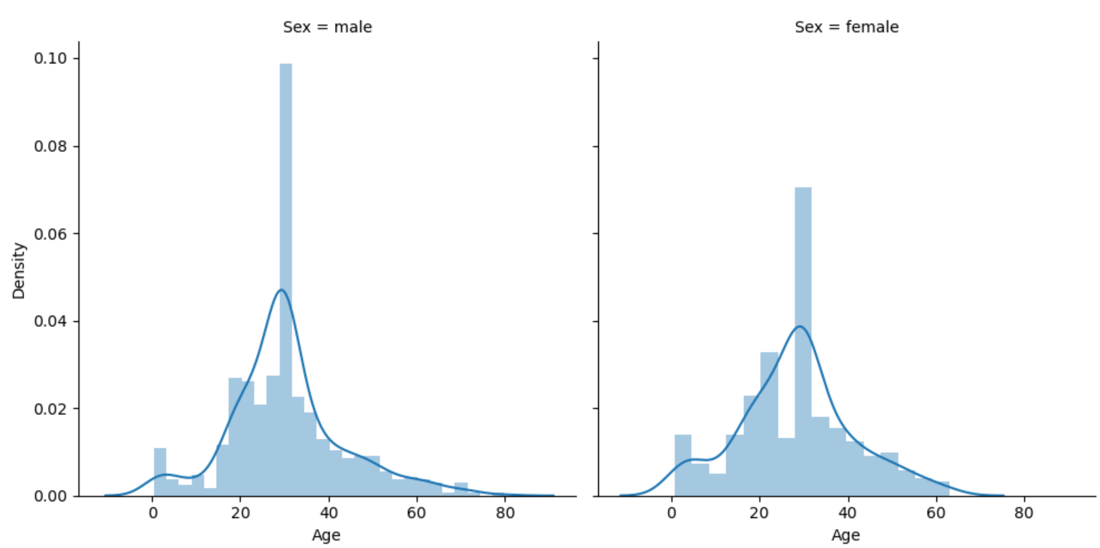


Insight:

This Plot clearly shows Male died more than females and females survived more than male.

Showing Distribution of Sex of Passengers Age wise

```
d2 = sns.FacetGrid(titanic, col="Sex",height=5)
d2 = (d2.map(sns.distplot, "Age").add_legend())
```

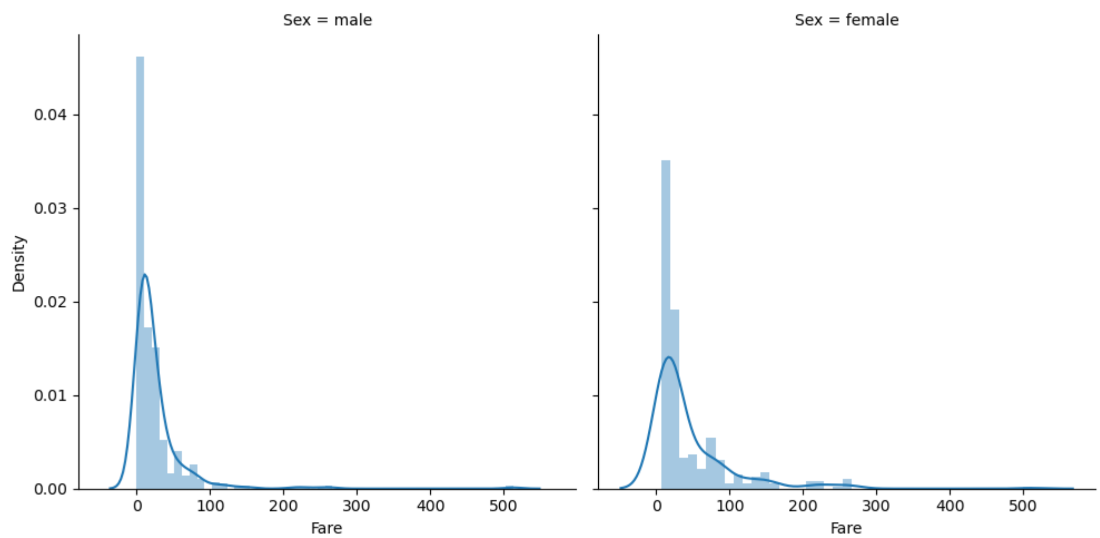


Insights:

we can see proportion of both males and females are generally lies between 20-40 age group

Showing Distribution of Sex of passengers fare wise

```
d3 = sns.FacetGrid(titanic, col="Sex",height=5)
d3 = (d3.map(sns.distplot, "Fare").add_legend())
```



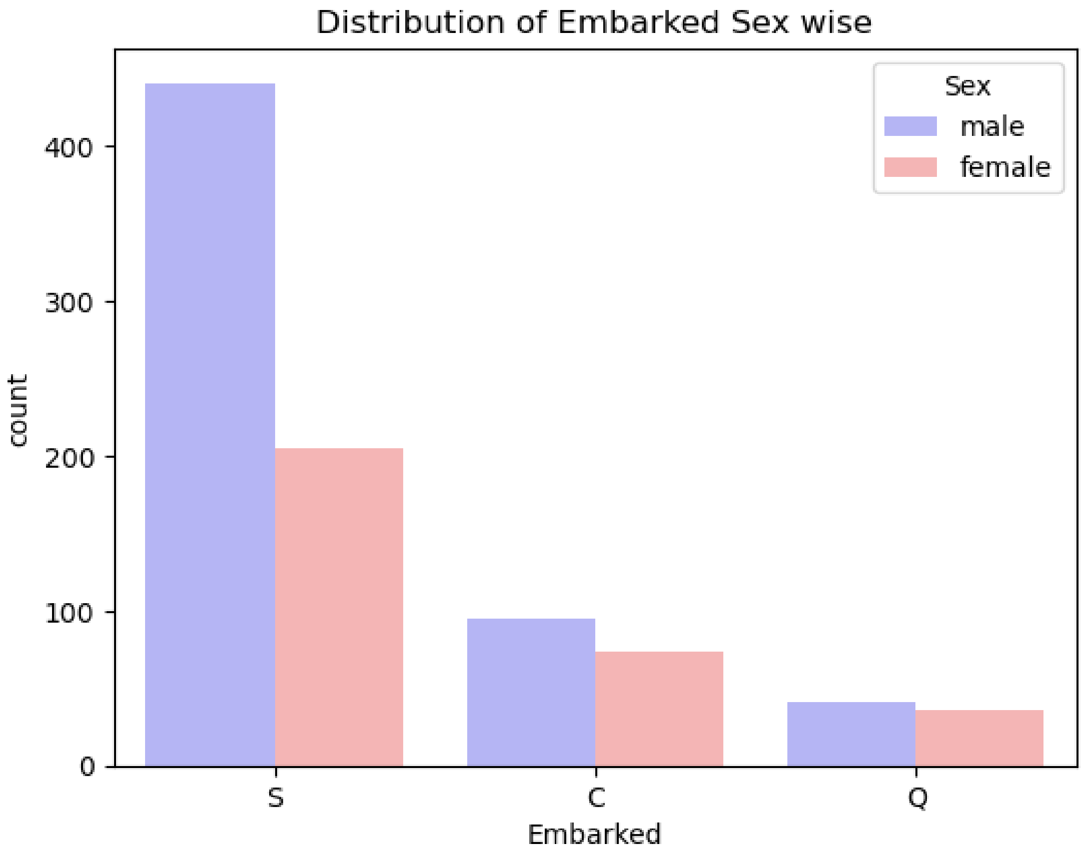
Insights:

From the Plot we can see that for both the genders , the price of the ticket are generally lies between 0 to 100. But, the density of the ticket is more for males.

Embarked Column

Showing Distribution of Embarked Sex wise

```
# Showing Distribution of Embarked Sex wise
sns.countplot(x=titanic['Embarked'],hue=titanic['Sex'],palette='bwr')
plt.title('Distribution of Embarked Sex wise')
plt.show()
```

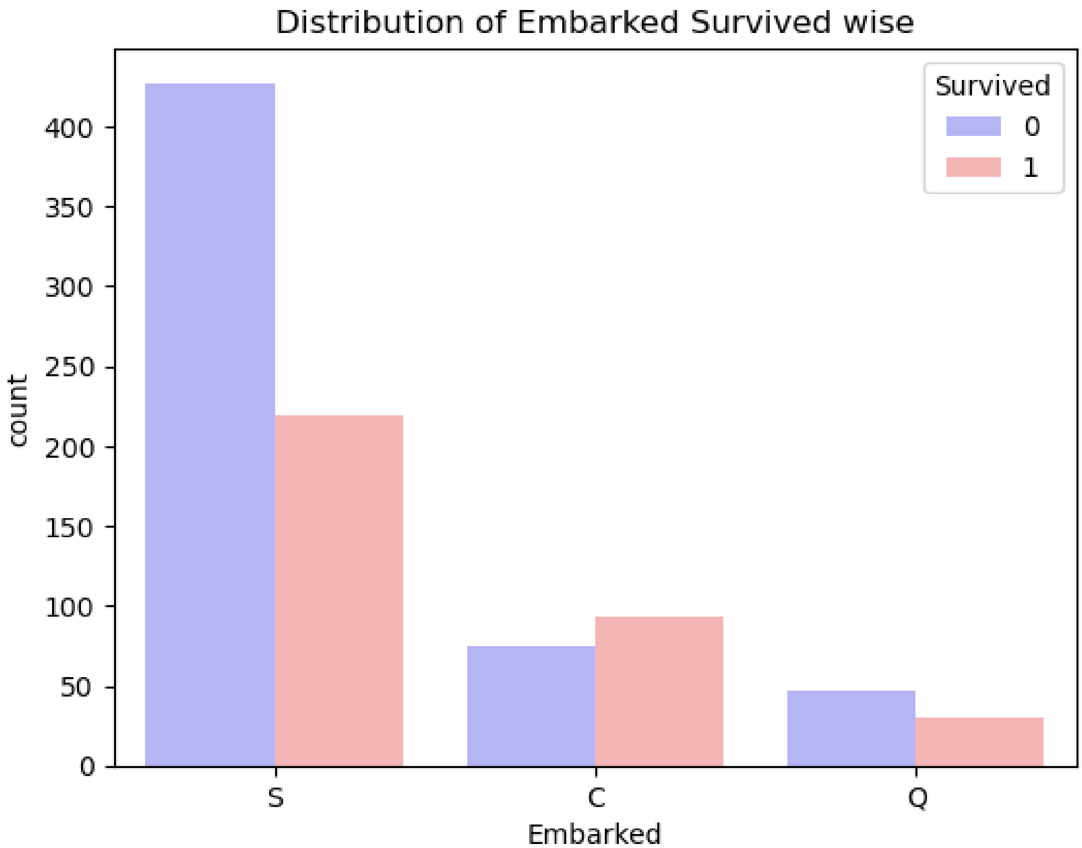


Insights:

We can Clearly see both kind of peoples either males or females mostly choose Southampton as their port of embarkation.

Showing Distribution of Embarked Survived wise

```
# Showing Distribution of Embarked Survived wise
sns.countplot(x=titanic['Embarked'],hue=titanic['Survived'],palette='bwr')
plt.title('Distribution of Embarked Survived wise')
plt.show()
```



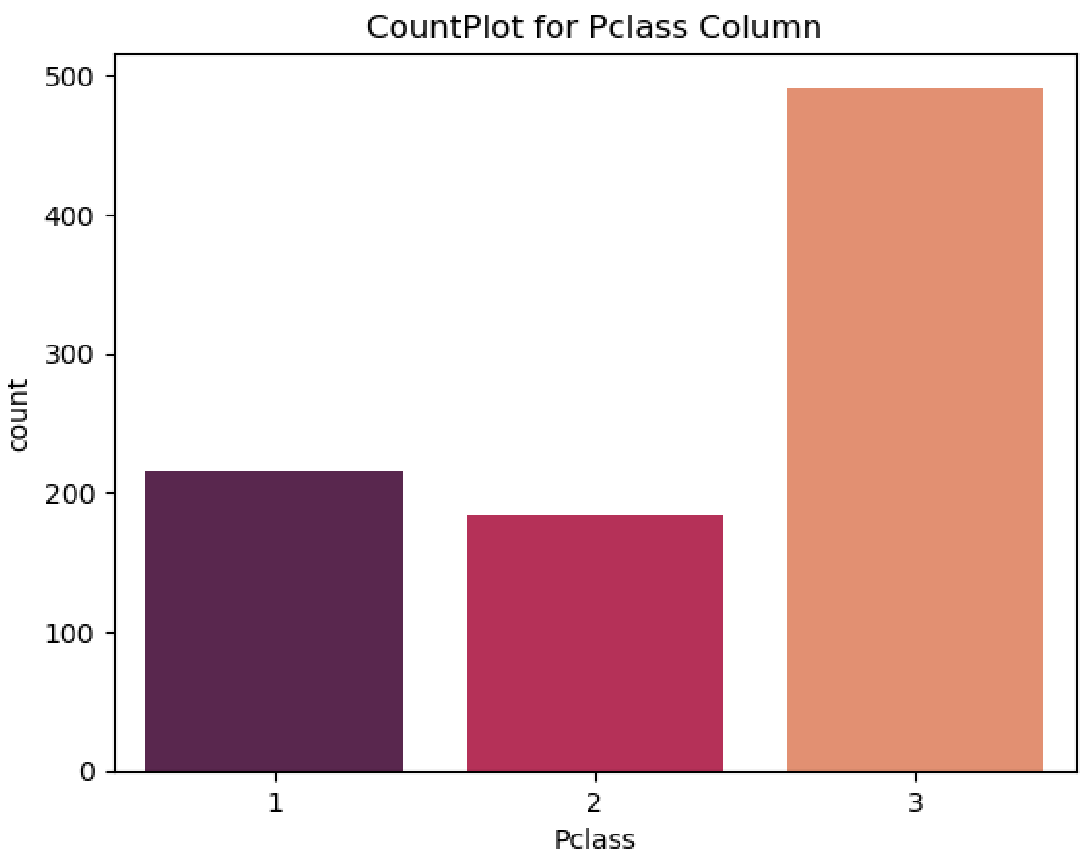
Insights:

1. The people who choose Southampton Embarked, death ratio is more than alive.
1. The people who choose Cherbourg Embarked, alive ratio is more than died.
2. The people who choose Queenstown Embarked, death ratio is more than alive.

Pclass Column

CountPlot for Pclass Column

```
# Plotting CountPlot for Pclass Column
sns.countplot(x=titanic['Pclass'],palette='rocket')
plt.title('CountPlot for Pclass Column')
plt.show()
```

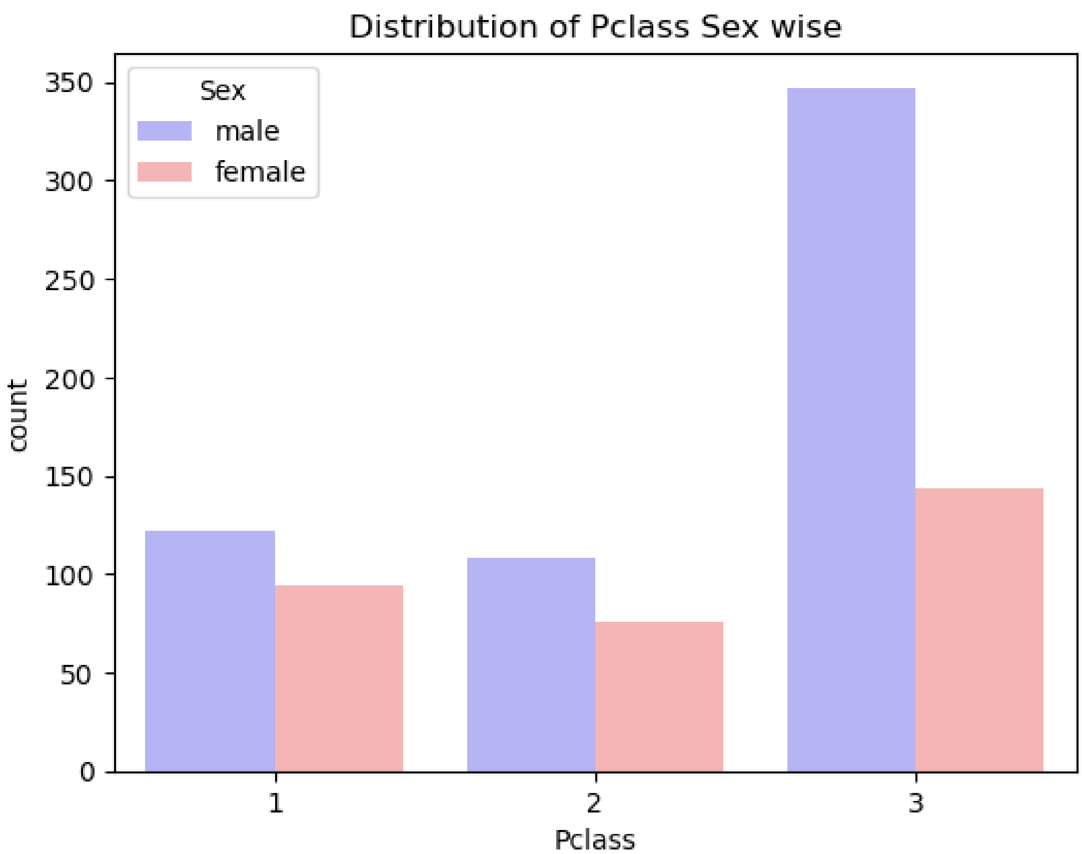


Insights:

From ther plot we can clearly observe that most of the people choose third class

Showing Distribution of Pclass Sex wise

```
# Showing Distribution of Pclass Sex wise
sns.countplot(x=titanic['Pclass'],hue=titanic['Sex'],palette='bwr')
plt.title('Distribution of Pclass Sex wise')
plt.show()
```



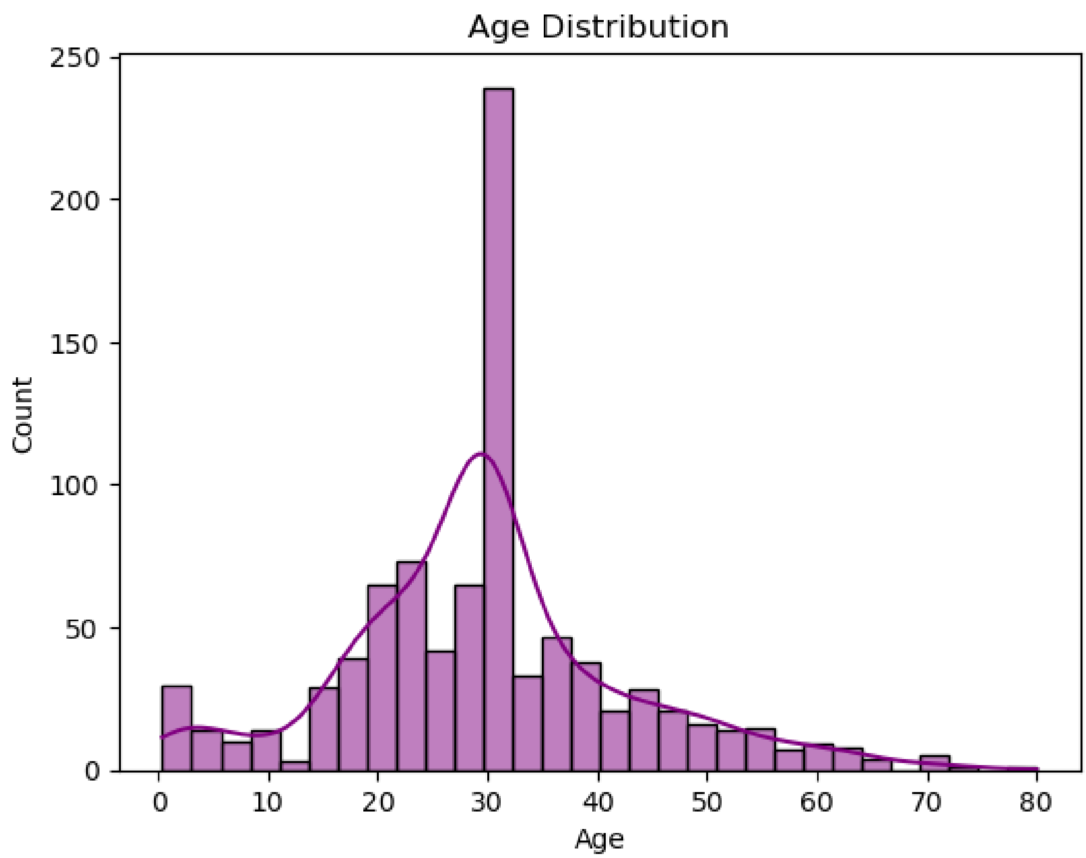
Insights:

As we draw the conclusion from the above plot that most of the people choose third class burt the the proportion of male is a way higher than females.

Age Column

Age Distribution

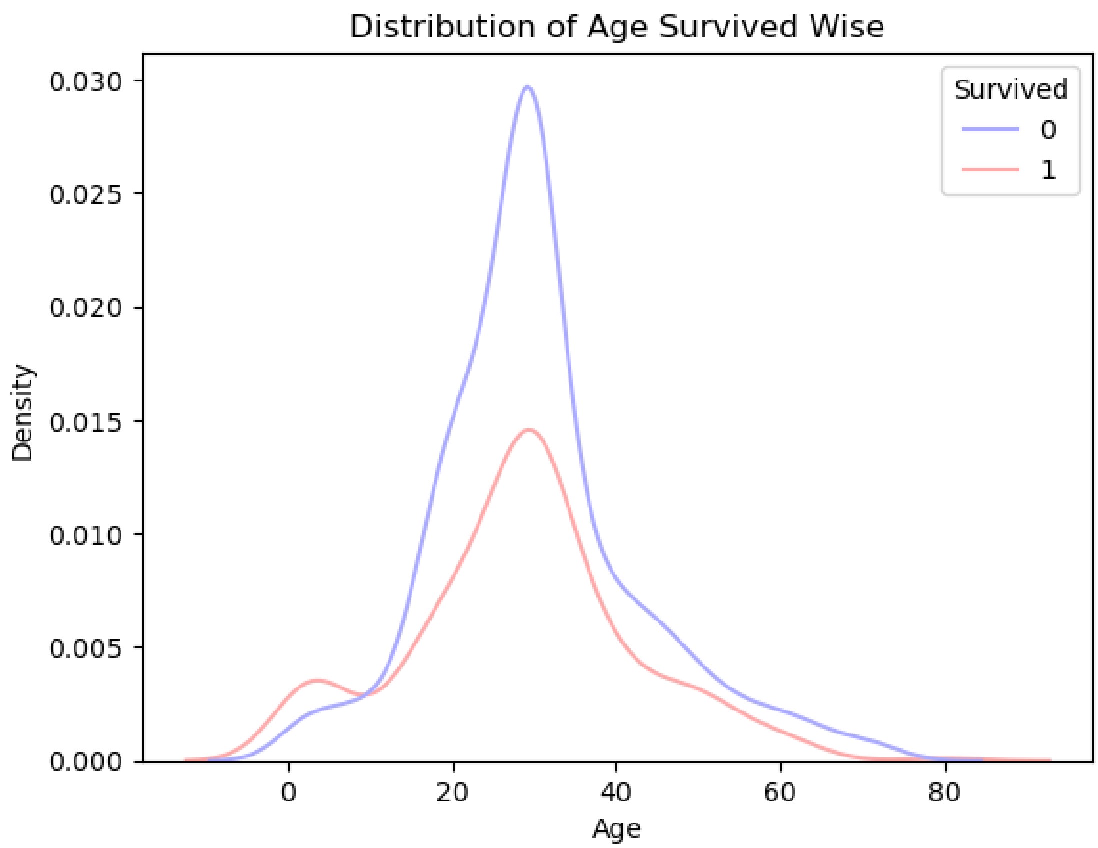
```
# Age Distribution
sns.histplot(x=titanic['Age'],kde=True,color='purple')
plt.title('Age Distribution')
plt.show()
```



Insights:

From this plot it came to know that most of the people lie between 20-40 age group.

```
# Showing Distribution of Age Survived Wise
sns.kdeplot(x=titanic['Age'],hue=titanic['Survived'],palette='bwr')
plt.title('Distribution of Age Survived Wise')
plt.show()
```



Insights:

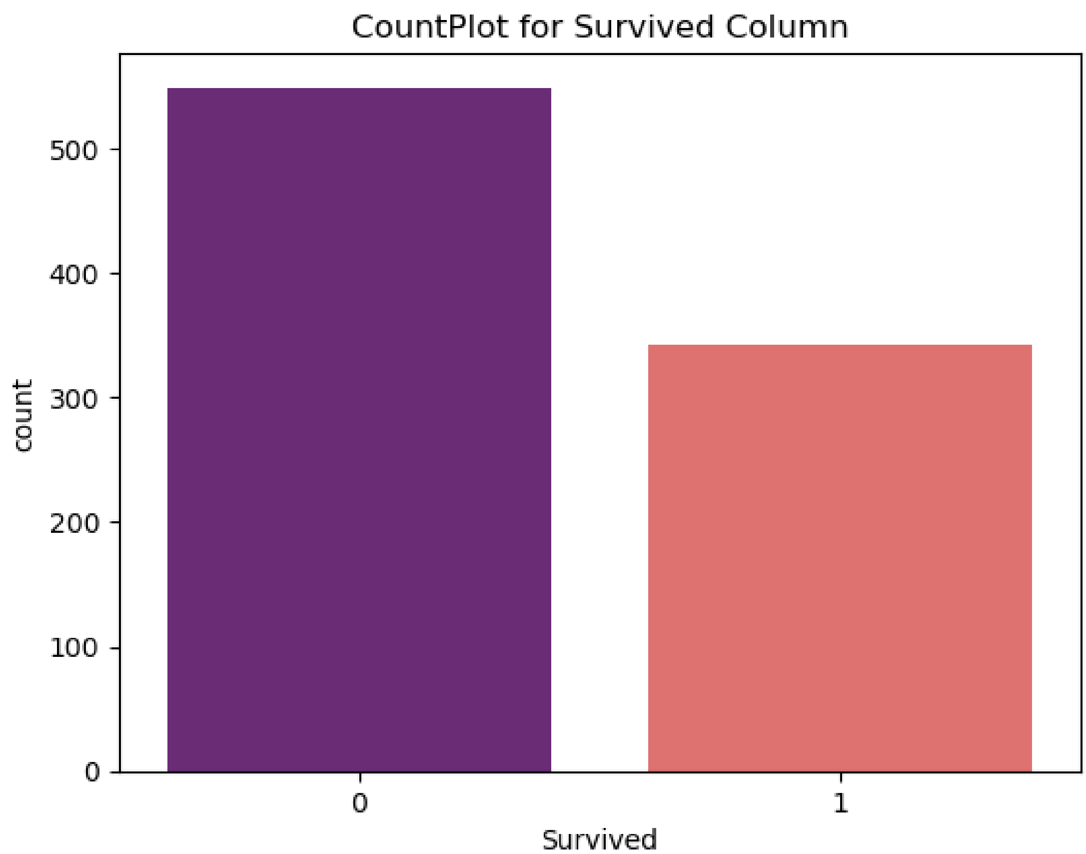
This Plot showing most people of age group of 20-40 are died

Survived Column

Countplot for Survived

```
# Plotting CountPlot for Survived Column
print(titanic['Survived'].value_counts())
sns.countplot(x=titanic['Survived'],palette='magma')
plt.title('CountPlot for Survived Column')
plt.show()
```

```
0    549
1    342
Name: Survived, dtype: int64
```



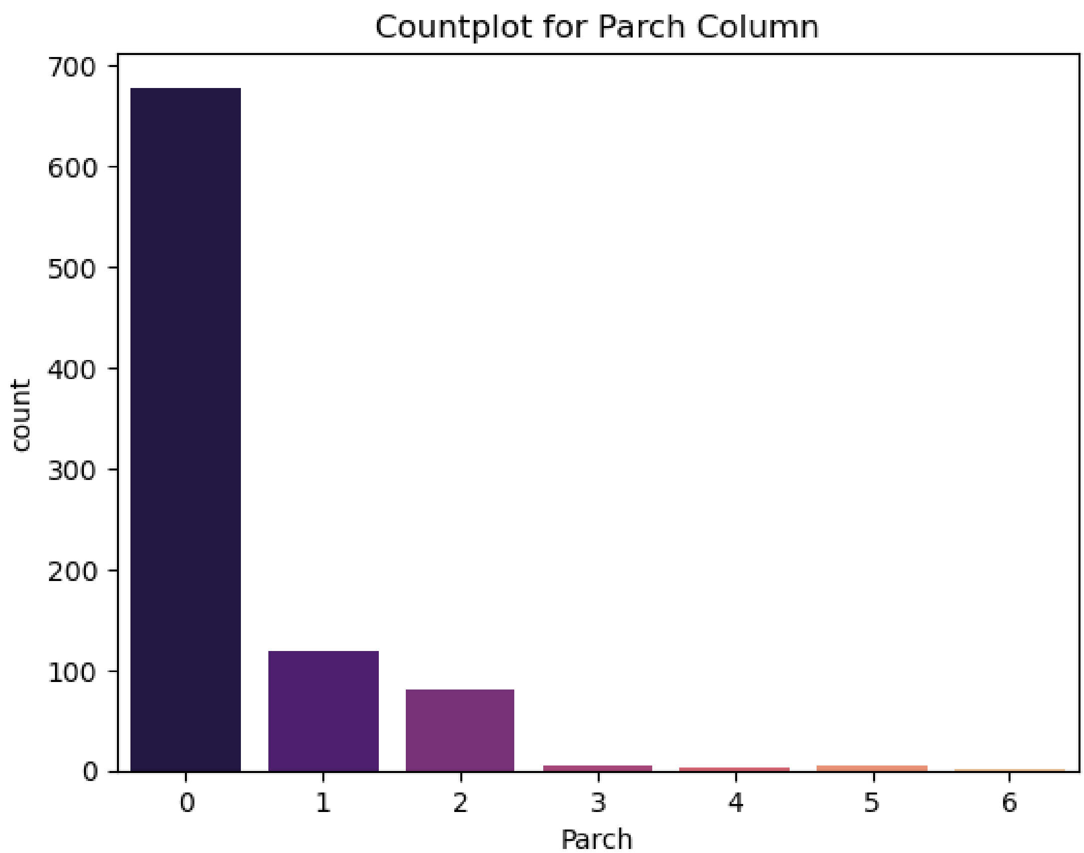
Insights:

This plot Clearly shows most people are died

Parch Column

CountPlot for Parch Column

```
# Countplot for Parch Column
sns.countplot(x=titanic['Parch'],palette='magma')
plt.title('Countplot for Parch Column')
plt.show()
```

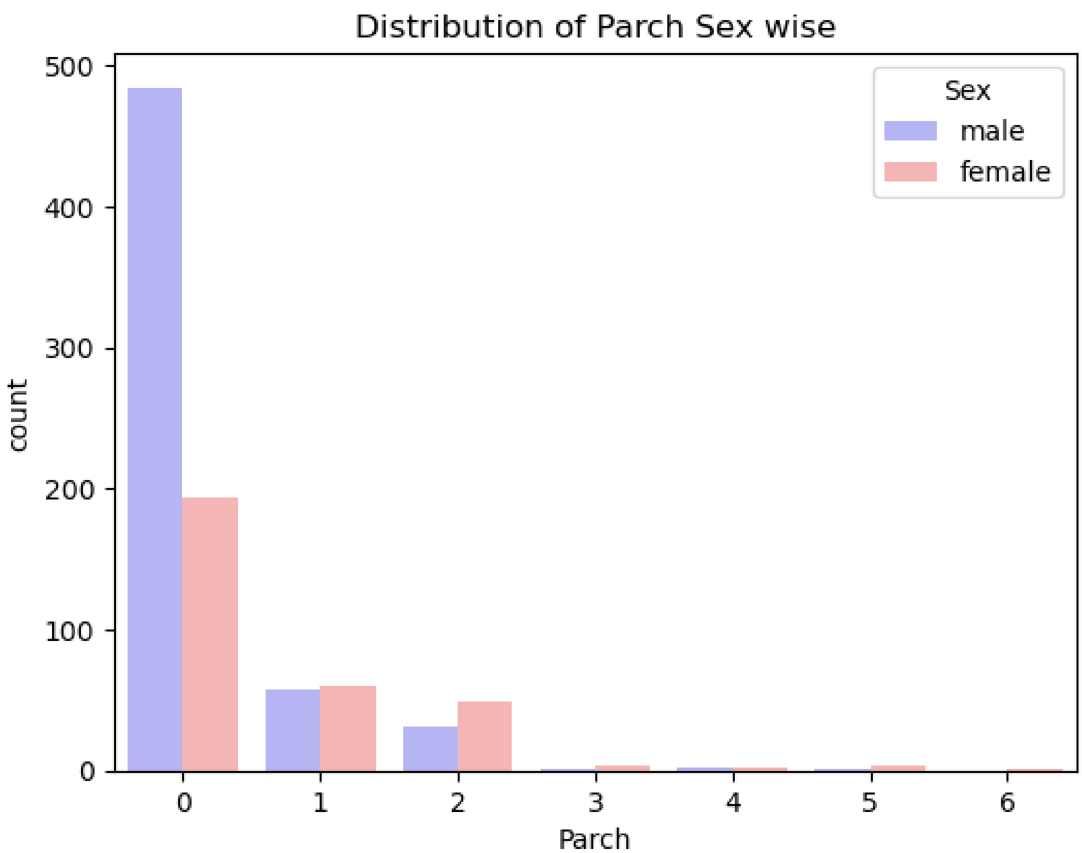


Insight:

1. Most of the passengers having 0 parents and childrens.
1. There are very few no. of passengers having 6 parents and childrens.

Showing Distribution of Parch Sex wise

```
# showing Distribution of Parch Sex wise
sns.countplot(x=titanic['Parch'],hue=titanic['Sex'],palette='bwr')
plt.title('Distribution of Parch Sex wise')
plt.show()
```

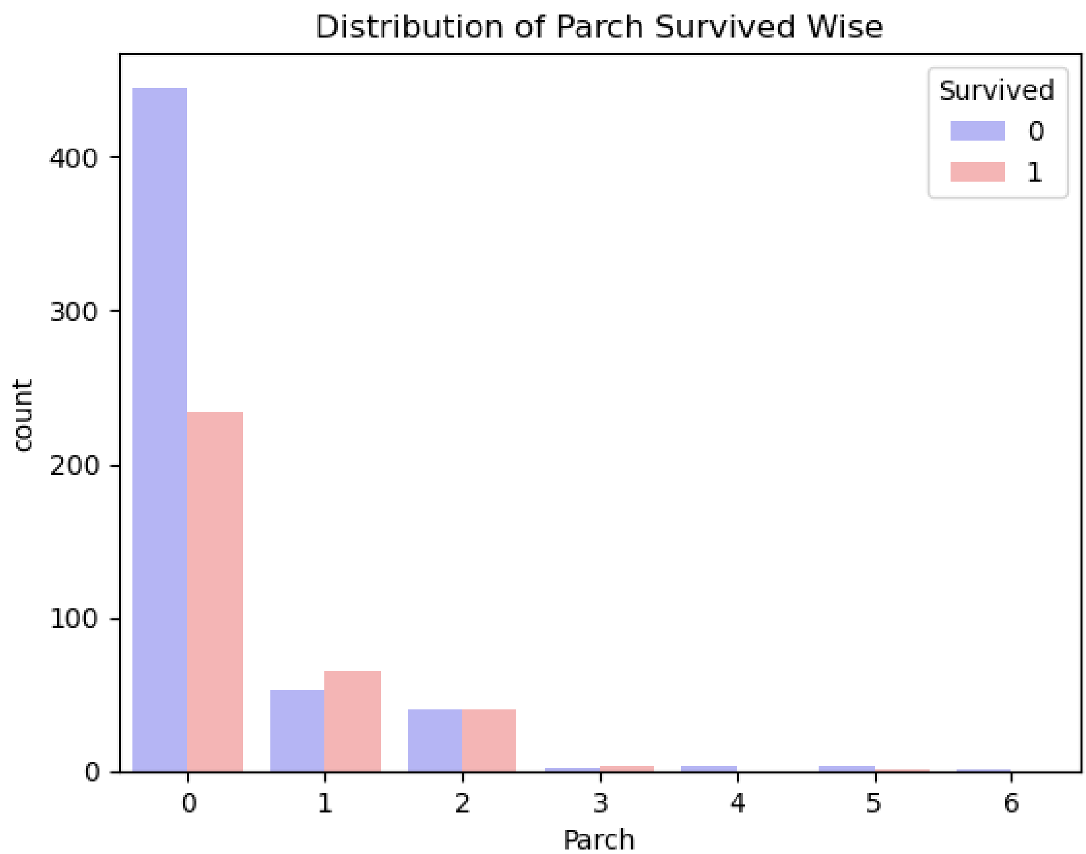


Insights:

1. The passengers are having 0 parents and childrens are mostly males
1. The passengers are having 1 parents and childrens are mostly females.
2. The passengers are having 2 parents and childrens are mostly females.
3. The passengers are having 3 parents and childrens are mostly females.
4. The passengers are having 4 parents and childrens are equivalent to each other..
5. The passengers are having 5 parents and childrens are mostly females.
6. The passengers are having 6 parents and childrens are only females.

Showing Distribution of Parch Survived Wise

```
# Showing Distribution of Parch Survived Wise
sns.countplot(x=titanic['Parch'],hue=titanic['Survived'],palette='bwr')
plt.title('Distribution of Parch Survived Wise')
plt.show()
```



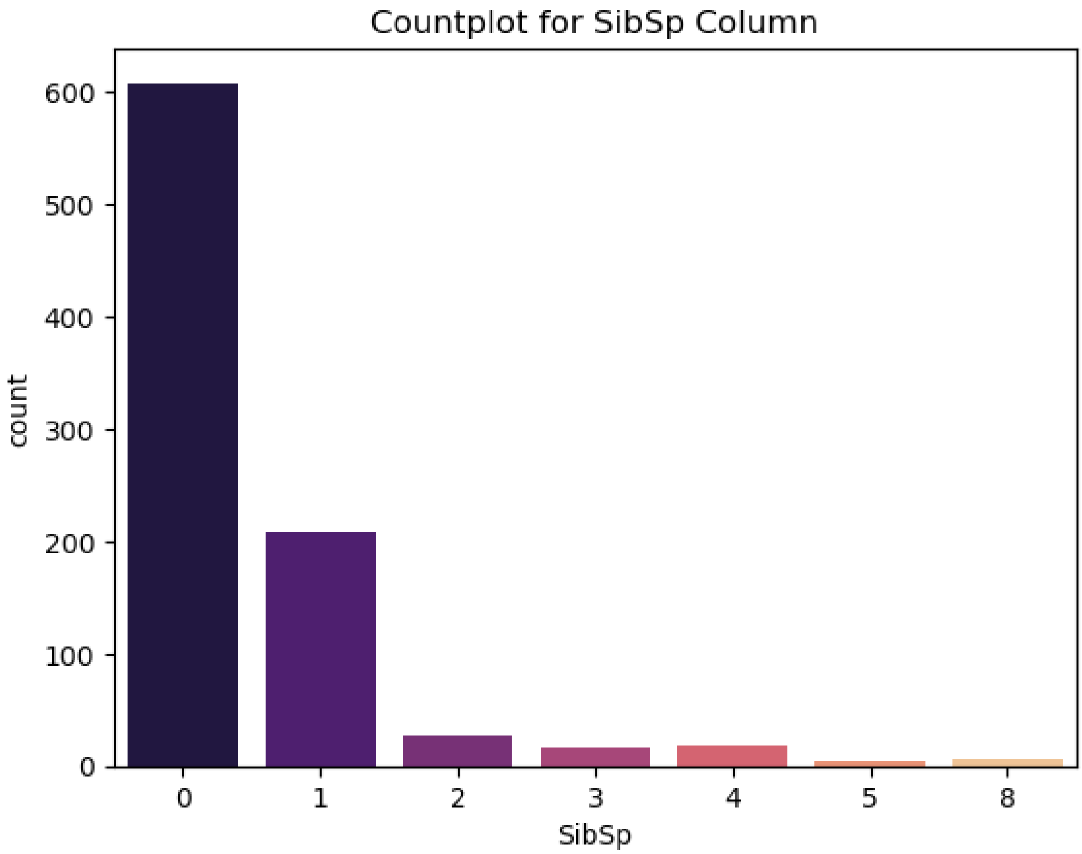
Insights:

1. The passengers having 0 Parents and childrens died more than survived.
1. The passengers having 1 Parents and childrens survived more than died.
2. The passengers having 2 Parents and childrens , died is equal to survived.
3. The passengers having 3 Parents and childrens , survived more than died.
4. The passengers having 4 Parents and childrens , mostly died.
5. The passengers having 5 Parents and childrens , died more than survived.
6. The passengers having 6 Parents and childrens , mostly died.

SibSp Column

Countplot for Sibsp Column

```
# countplot for SibSp Column
sns.countplot(x=titanic['SibSp'],palette='magma')
plt.title('Countplot for SibSp Column')
plt.show()
```

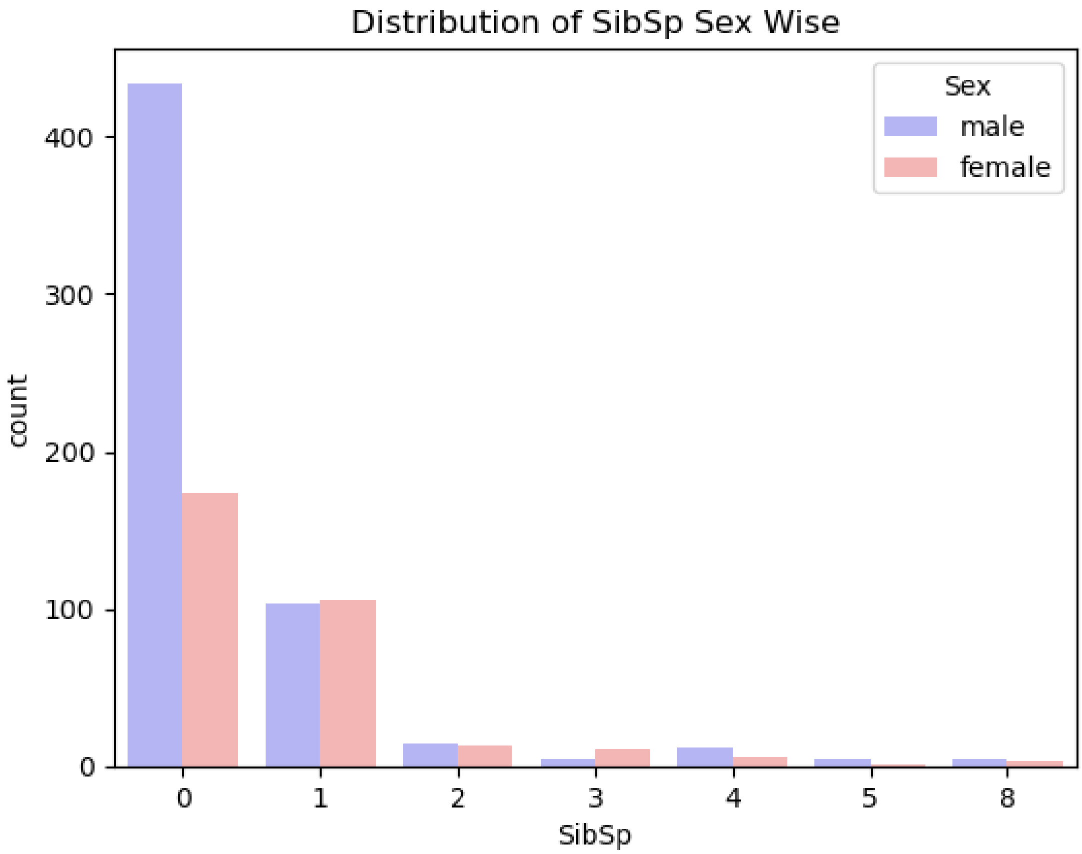


Insights:

1. Most of the passengers having 0 siblings and spouses.
1. There are very few no. of passengers having 5 siblings and spouses.

Showing Distribution of SibSp Sex wise

```
# Showing Distribution of SibSp Sex Wise
sns.countplot(x=titanic['SibSp'],hue=titanic['Sex'],palette='bwr')
plt.title('Distribution of SibSp Sex Wise')
plt.show()
```

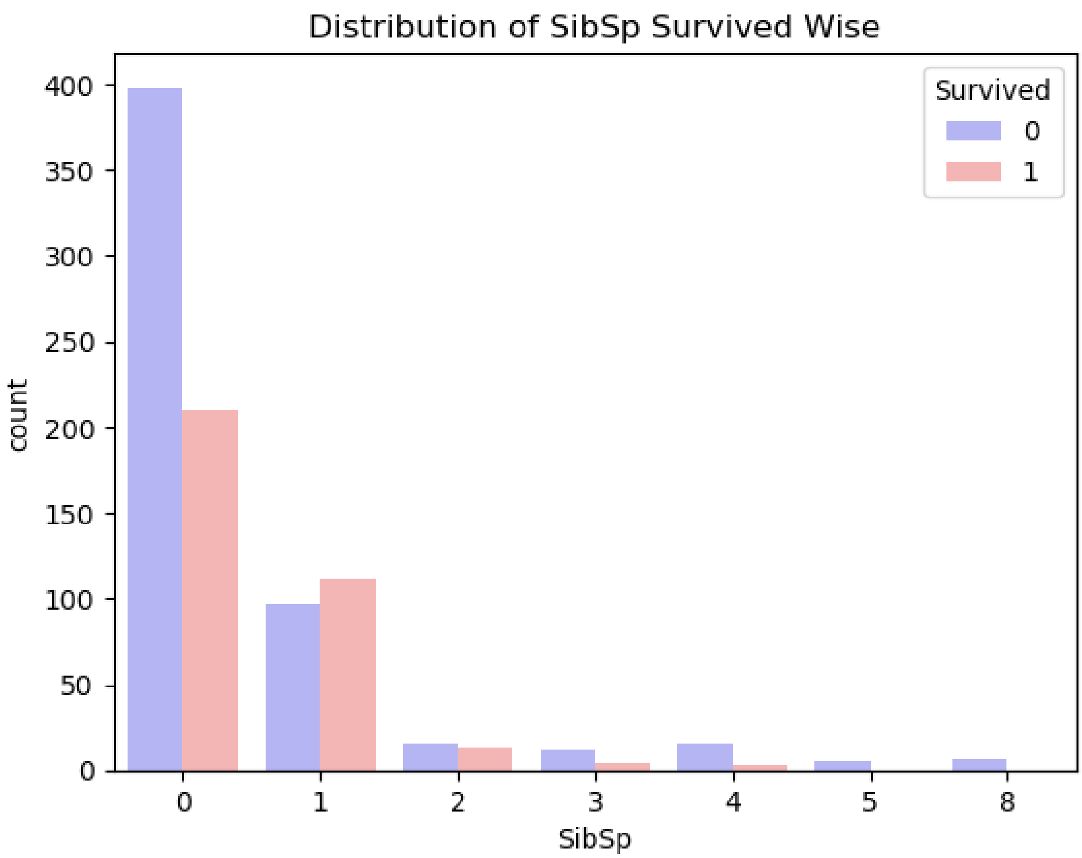


Insights:

1. The passengers having 0 siblings and spouses are mostly males.
1. The passengers having 1 siblings and spouses are mostly females.
2. The passengers having 2 siblings and spouses are mostly males.
3. The passengers having 3 siblings and spouses are mostly females.
4. The passengers having 4 siblings and spouses are mostly males.
5. The passengers having 5 siblings and spouses are mostly males.
6. The passengers having 6 siblings and spouses are mostly males.

Showing Distribution of SibSp Survived Wise

```
# Showing Distribution of SibSp Survived Wise
sns.countplot(x=titanic['SibSp'],hue=titanic['Survived'],palette='bwr')
plt.title('Distribution of SibSp Survived Wise')
plt.show()
```

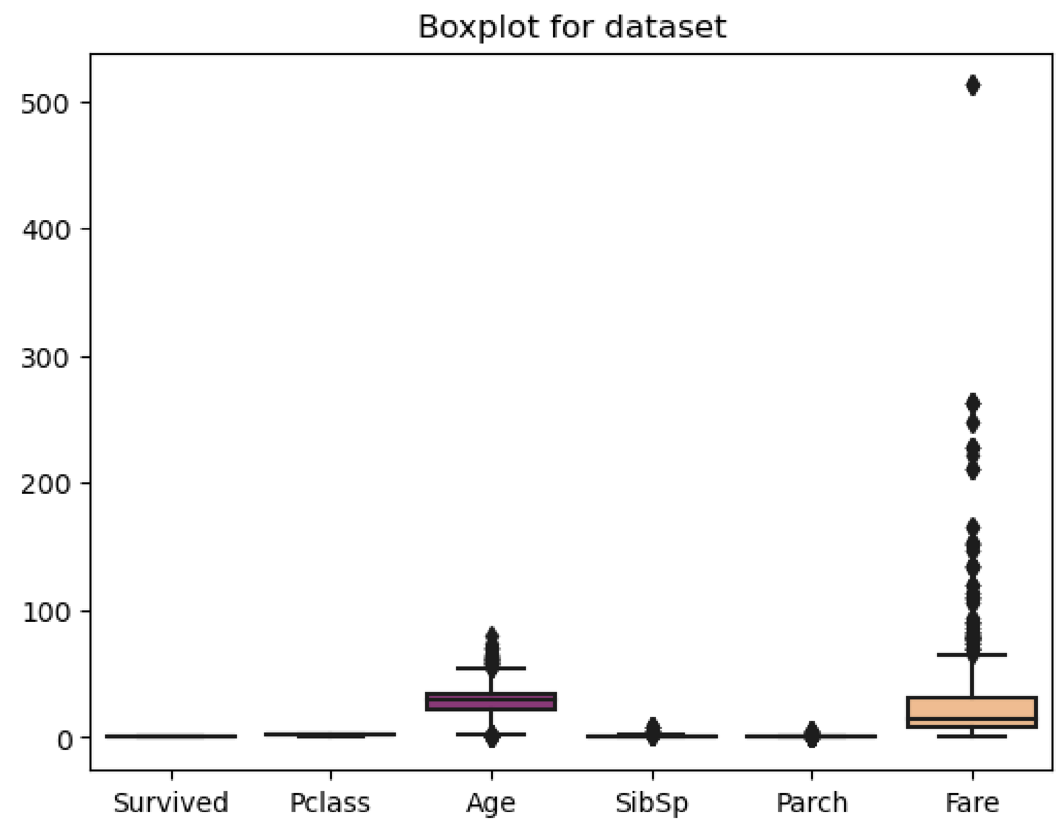


Insights:

1. The passengers having 0 siblings and spouses died more than survived.
1. The passengers having 1 siblings and spouses survived more than died.
2. The passengers having 2 siblings and spouses died more than survived.
3. The passengers having 3 siblings and spouses died more than survived.
4. The passengers having 4 siblings and spouses died more than survived.
5. The passengers having 5 siblings and spouses died only
6. The passengers having 6 siblings and spouses died only

Plotting BoxPlot and Checking for Outliers

```
# Plotting Boxplot for dataset
# Checking for outliers
sns.boxplot(titanic,palette='magma')
plt.title('Boxplot for dataset')
plt.show()
```



Insights:

This Plot showing Outliers in 2 columns i.e.. Age and Fare.But, Outliers in Age column not affecting dataset rather than fare one . So, we will removing Outliers present in Fare column.

```
# Removing Outliers

column = titanic[['Age','Fare']]
q1 = np.percentile(column, 25)
q3 = np.percentile(column, 75)
iqr = q3 - q1
lower_bound = q1 - 1.5 * iqr
upper_bound = q3 + 1.5 * iqr
titanic[['Age','Fare']] = column[(column > lower_bound) & (column <
upper_bound)]

sns.boxplot(titanic,palette='magma')
plt.title('Boxplot for dataset after removing outliers')
plt.show()
```

Plotting Correlation Plot

```
# showing Correlation
titanic.corr()
```

	Survived	Pclass	Age	SibSp	Parch	Fare
Survived	1.000000	-0.338481	-0.061145	-0.035322	0.081629	0.234422
Pclass	-0.338481	1.000000	-0.329012	0.083081	0.018443	-0.589776
Age	-0.061145	-0.329012	1.000000	-0.233936	-0.179291	0.071485
SibSp	-0.035322	0.083081	-0.233936	1.000000	0.414838	0.370388
Parch	0.081629	0.018443	-0.179291	0.414838	1.000000	0.336844
Fare	0.234422	-0.589776	0.071485	0.370388	0.336844	1.000000

```
# Showing Correlation Plot
sns.heatmap(titanic.corr(),annot=True,cmap='Pastel1')
plt.title('Correlation Plot')
plt.show()
```

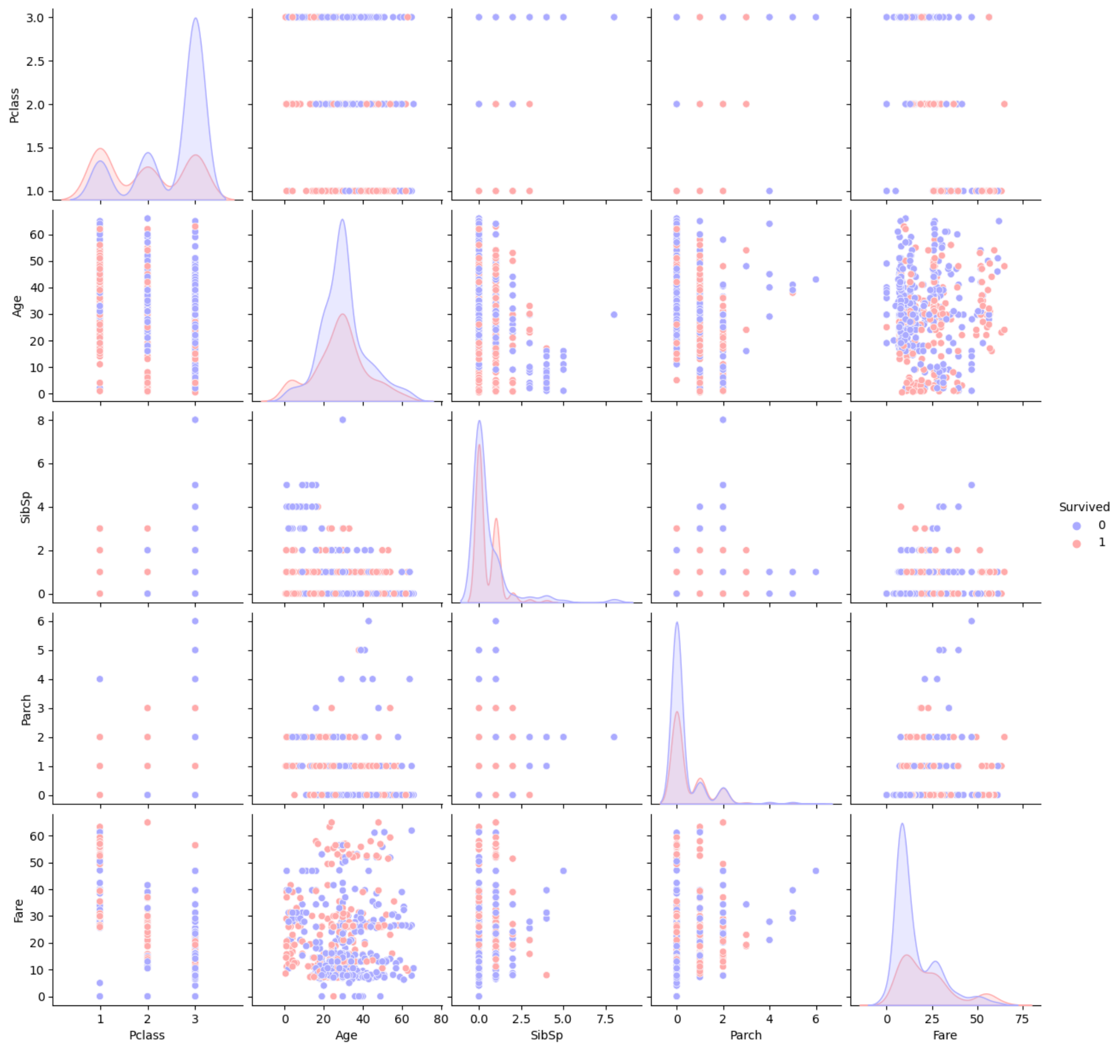
Insights:

This Plot is clearly showing

1. Strong Positive Correlation between SibSp and Parch
1. Strong Negative Correlation between Pclass and Fare

Plotting Pairplot

```
# Plotting pairplot
sns.pairplot(titanic,hue='Survived',palette='bwr')
plt.show()
```



Conclusion

The sinking of the Titanic is indeed a tragic and historically significant event. The dataset we have provided contains various features related to the passengers onboard the Titanic. It include features like PassengerId, Survived, Pclass, Name, Sex, Age, SibSp, Parch, Ticket, Fare, Cabin, and Embarked. By doing the analysis on these features, we are able to get the survival rate of passengers onboard the titanic, impact of Pclass and embarked on the passengers survival, Age and Fare wise ditribution of passengers, survival rate of different passengers gender wise, impact of having siblings and spouses, parents and children on the passengers survival and so on.

This dataset is a very good source for performing Exploratory Data Analysis.