Project 2 : Titanic Survival Prediction

In [1]:
```
import pandas as pd
```

In [2]:
```
titanic_data=pd.read_csv("C:/Users/Ayush/Desktop/Afame Tech/DA Project Details/Titanic-Dataset.csv")
titanic_data
```

Out[2]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S
...	...	...	...	...	...	...	...	...	...	...	...	...
886	887	0	2	Montvila, Rev. Juozas	male	27.0	0	0	211536	13.0000	NaN	S
887	888	1	1	Graham, Miss. Margaret Edith	female	19.0	0	0	112053	30.0000	B42	S
888	889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	NaN	1	2	W./C. 6607	23.4500	NaN	S
889	890	1	1	Behr, Mr. Karl Howell	male	26.0	0	0	111369	30.0000	C148	C
890	891	0	3	Dooley, Mr. Patrick	male	32.0	0	0	370376	7.7500	NaN	Q

891 rows × 12 columns

In [3]:
```
titanic_data.shape
```

Out[3]:
```
(891, 12)
```

In [4]:
```
titanic_data.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
```

In [5]:
```
titanic_data.describe()
```

Out[5]:

	PassengerId	Survived	Pclass	Age	SibSp	Parch	Fare
count	891.000000	891.000000	891.000000	714.000000	891.000000	891.000000	891.000000
mean	446.000000	0.383838	2.308642	29.699118	0.523008	0.381594	32.204208
std	257.353842	0.486592	0.836071	14.526497	1.102743	0.806057	49.693429
min	1.000000	0.000000	1.000000	0.420000	0.000000	0.000000	0.000000
25%	223.500000	0.000000	2.000000	20.125000	0.000000	0.000000	7.910400
50%	446.000000	0.000000	3.000000	28.000000	0.000000	0.000000	14.454200
75%	668.500000	1.000000	3.000000	38.000000	1.000000	0.000000	31.000000
max	891.000000	1.000000	3.000000	80.000000	8.000000	6.000000	512.329200

In [6]:
```
titanic_data.columns
```

Out[6]:
```
Index(['PassengerId', 'Survived', 'Pclass', 'Name', 'Sex', 'Age', 'SibSp',
       'Parch', 'Ticket', 'Fare', 'Cabin', 'Embarked'],
      dtype='object')
```

In [7]:
```
titanic_data.isnull().any() #three columns has null values and those are : Age,Cabin,Embarked
```

Out[7]:
```
PassengerId    False
Survived       False
Pclass         False
Name           False
Sex            False
Age             True
SibSp          False
Parch          False
Ticket         False
Fare           False
Cabin           True
Embarked        True
dtype: bool
```

Handling missing values in Age Column

In [8]:
```
titanic_data['Age'].isna().sum() #177 values
```

Out[8]:
```
177
```

In [9]:
```
titanic_data['Age']
```

Out[9]:
```
0      22.0
1      38.0
2      26.0
3      35.0
4      35.0
       ...
886    27.0
887    19.0
888     NaN
889    26.0
890    32.0
Name: Age, Length: 891, dtype: float64
```

Using Predictive imputation for handling the missing values in the Age column

In [10]:
```
titanic_data['Age'].unique() #Age should not be in float because should int shows definite number and age is defined in definite number always
```

Out[10]:
```
array([22.  , 38.  , 26.  , 35.  ,   nan, 54.  ,  2.  , 27.  , 14.  ,
        4.  , 58.  , 20.  , 39.  , 55.  , 31.  , 34.  , 15.  , 28.  ,
        8.  , 19.  , 40.  , 66.  , 42.  , 21.  , 18.  ,  3.  ,  7.  ,
       49.  , 29.  , 65.  , 28.5 ,  5.  , 11.  , 45.  , 17.  , 32.  ,
       16.  , 25.  ,  0.83, 30.  , 33.  , 23.  , 24.  , 46.  , 59.  ,
       71.  , 37.  , 47.  , 14.5 , 70.5 , 32.5 , 12.  ,  9.  , 36.5 ,
       51.  , 55.5 , 40.5 , 44.  ,  1.  , 61.  , 56.  , 50.  , 36.  ,
       45.5 , 20.5 , 62.  , 41.  , 52.  , 63.  , 23.5 ,  0.92, 43.  ,
       60.  , 10.  , 64.  , 13.  , 48.  ,  0.75, 53.  , 57.  , 80.  ,
       70.  , 24.5 ,  6.  ,  0.67, 30.5 ,  0.42, 34.5 , 74.  ])
```

In [11]:
```
from sklearn.ensemble import RandomForestRegressor
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error

# Check data types of all columns
print(titanic_data.dtypes)

# Prepare the data: Drop rows with missing age values and convert categorical variables into numerical representations if necessary
# For simplicity, let's drop 'Cabin' and 'Embarked' columns for now
titanic_data_temp = titanic_data.drop(['Cabin', 'Embarked'], axis=1)

# Check for non-numeric values
print(titanic_data_temp.select_dtypes(include=['object']).columns)

# It seems like 'Sex' column contains non-numeric values. Let's encode it.
titanic_data_temp = pd.get_dummies(titanic_data_temp, columns=['Sex'])

# Split the data into features (X) and target variable (y)
X = titanic_data_temp.drop(['Age', 'Name', 'Ticket'], axis=1)
y = titanic_data_temp['Age']

# Split the dataset into training and test sets
X_train, X_test = X[~y.isnull()], X[y.isnull()]
y_train = y[~y.isnull()]

# Train the model
model = RandomForestRegressor(random_state=42)
model.fit(X_train, y_train)

# Predict missing ages
predicted_ages = model.predict(X_test)

# Impute missing values
titanic_data.loc[titanic_data['Age'].isnull(), 'Age'] = predicted_ages
```

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

In [12]:
```
titanic_data['Age'].isna().sum() #clearing missing values
```

Out[12]:
```
0
```

In [13]:
```
titanic_data['Age'] = titanic_data['Age'].astype(int)
```

In [14]:
```
titanic_data['Age']
```

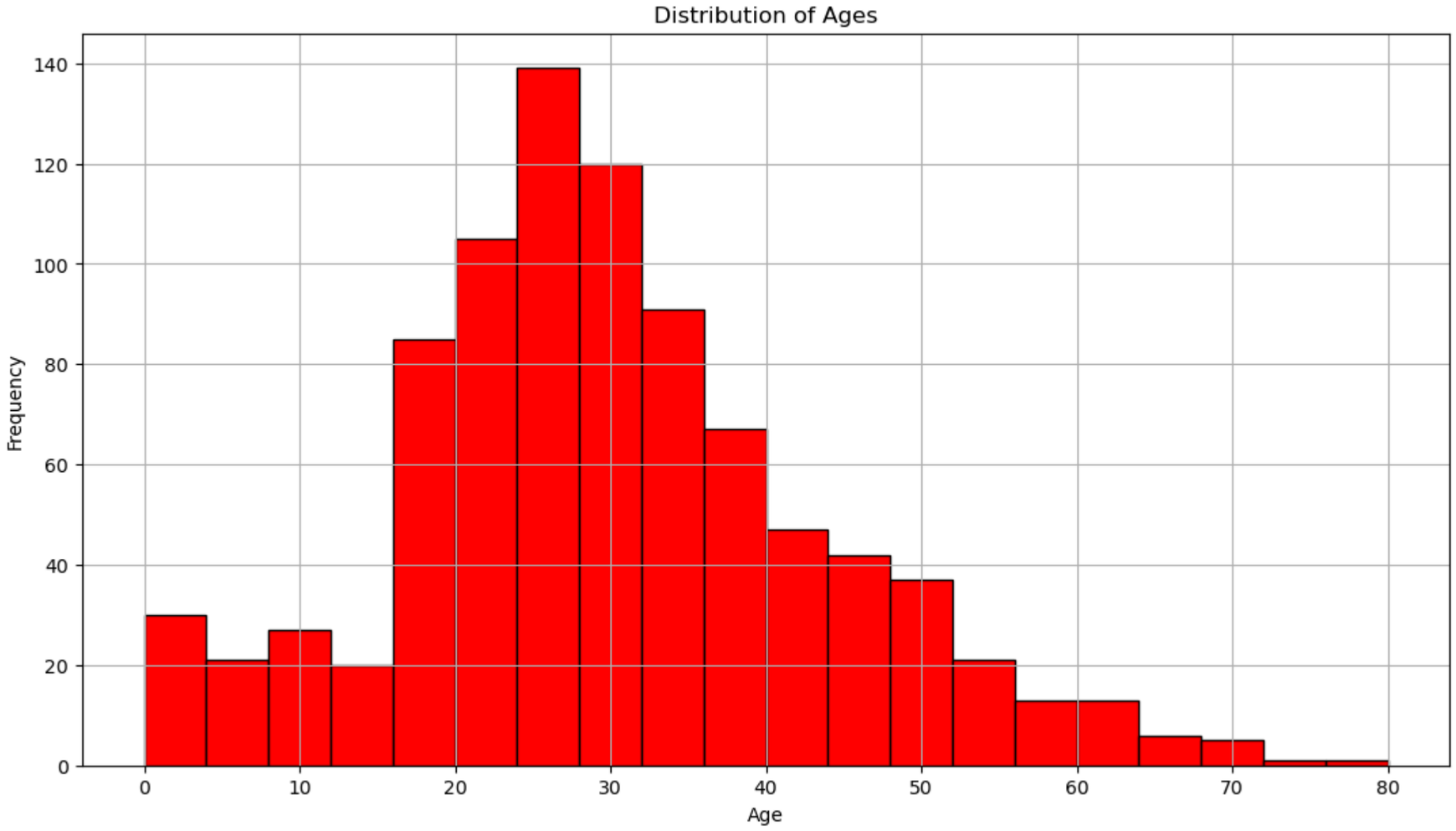
Out[14]:
```
0      22
1      38
2      26
3      35
4      35
       ..
886    27
887    19
888    23
889    26
890    32
Name: Age, Length: 891, dtype: int32
```

In [15]:
```
titanic_data.sample(7)
```

Out[15]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
554	555	1	3	Ohman, Miss. Velin	female	22	0	0	347085	7.7750	NaN	S
99	100	0	2	Kantor, Mr. Sinai	male	34	1	0	244367	26.0000	NaN	S
481	482	0	2	Frost, Mr. Anthony Wood "Archie"	male	39	0	0	239854	0.0000	NaN	S
279	280	1	3	Abbott, Mrs. Stanton (Rosa Hunt)	female	35	1	1	C.A. 2673	20.2500	NaN	S
659	660	0	1	Newell, Mr. Arthur Webster	male	58	0	2	35273	113.2750	D48	C
396	397	0	3	Olsson, Miss. Elina	female	31	0	0	350407	7.8542	NaN	S
151	152	1	1	Pears, Mrs. Thomas (Edith Wearne)	female	22	1	0	113776	66.6000	C2	S

In [16]:
```
import matplotlib.pyplot as plt

# Create histogram
plt.figure(figsize=(13, 7))
plt.hist(titanic_data['Age'], bins=20, color='red', edgecolor='black')
plt.title('Distribution of Ages')
plt.xlabel('Age')
plt.ylabel('Frequency')
plt.grid(True)
plt.show()
```



In [17]:
```
titanic_data['Age'].value_counts()
```

Out[17]:
```
Age
24    45
30    41
25    38
28    35
22    32
      ..
53     2
57     2
66     1
80     1
74     1
Name: count, Length: 71, dtype: int64
```

Handling missing values in Cabin Column

In [18]:
```
titanic_data['Cabin'].isna().sum() #687 values
```

Out[18]:
```
687
```

In [19]:
```
titanic_data['Cabin']
```

Out[19]:
```
0       NaN
1       C85
2       NaN
3      C123
4       NaN
       ...
886     NaN
887     B42
888     NaN
889    C148
890     NaN
Name: Cabin, Length: 891, dtype: object
```

In [20]:
```
titanic_data['Cabin'].isna().sum()
```

Out[20]:
```
687
```

In [21]:
```
titanic_data.sample(3)
```

Out[21]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
338	339	1	3	Dahl, Mr. Karl Edwart	male	45	0	0	7598	8.050	NaN	S
68	69	1	3	Andersson, Miss. Erna Alexandra	female	17	4	2	3101281	7.925	NaN	S
526	527	1	2	Ridsdale, Miss. Lucy	female	50	0	0	W./C. 14258	10.500	NaN	S

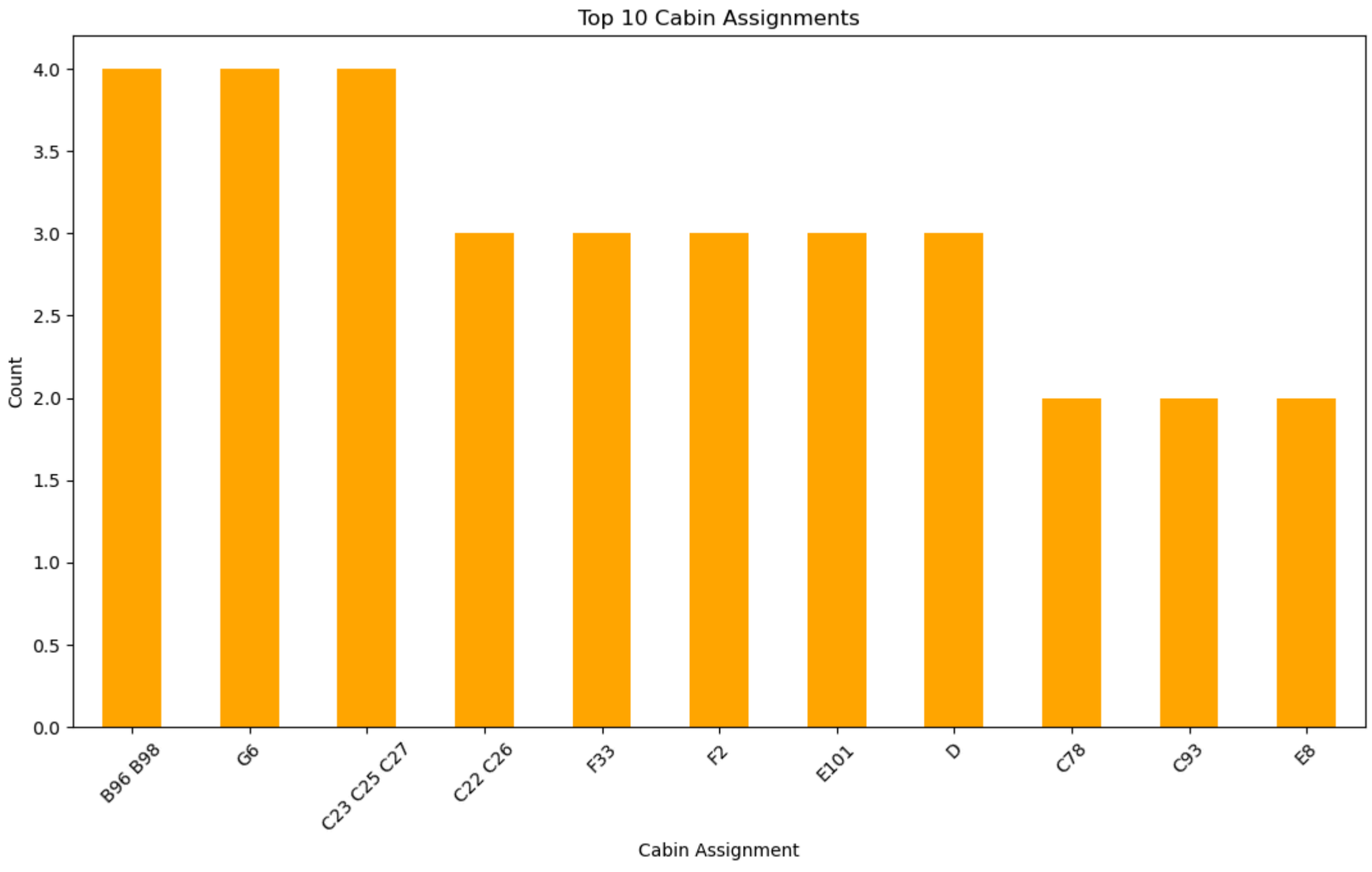
In [22]:
```
titanic_data['Cabin'].unique()
```

Out[22]:
```
array([nan, 'C85', 'C123', 'E46', 'G6', 'C103', 'D56', 'A6',
       'C23 C25 C27', 'B78', 'D33', 'B30', 'C52', 'B28', 'C83', 'F33',
       'F G73', 'E31', 'A5', 'D10 D12', 'D26', 'C110', 'B58 B60', 'E101',
       'F E69', 'D47', 'B86', 'F2', 'C2', 'E33', 'B19', 'A7', 'C49', 'F4',
       'A32', 'B4', 'B80', 'A31', 'D36', 'D15', 'C93', 'C78', 'D35',
       'C87', 'B77', 'E67', 'B94', 'C125', 'C99', 'C118', 'D7', 'A19',
       'B49', 'D', 'C22 C26', 'C106', 'C65', 'E36', 'C54',
       'B57 B59 B63 B66', 'C7', 'E34', 'C32', 'B18', 'C124', 'C91', 'E40',
       'T', 'C128', 'D37', 'B35', 'E50', 'C82', 'B96 B98', 'E10', 'E44',
       'A34', 'C104', 'C111', 'C92', 'E38', 'D21', 'E12', 'E63', 'A14',
       'B37', 'C30', 'D20', 'B79', 'E25', 'D46', 'B73', 'C95', 'B38',
       'B39', 'B22', 'C86', 'C70', 'A16', 'C101', 'C68', 'A10', 'E68',
       'B41', 'A20', 'D19', 'D50', 'D9', 'A23', 'B50', 'A26', 'D48',
       'E58', 'C126', 'B71', 'B51 B53 B55', 'D49', 'B5', 'B20', 'F G63',
       'C62 C64', 'E24', 'C90', 'C45', 'E8', 'B101', 'D45', 'C46', 'D30',
       'E121', 'D11', 'E77', 'F38', 'B3', 'D6', 'B82 B84', 'D17', 'A36',
       'B102', 'B69', 'E49', 'C47', 'D28', 'E17', 'A24', 'C50', 'B42',
       'C148'], dtype=object)
```

In [23]:
```
titanic_data['Cabin']
```

Out[23]:
```
0       NaN
1       C85
2       NaN
3      C123
4       NaN
       ...
886     NaN
887     B42
888     NaN
889    C148
890     NaN
Name: Cabin, Length: 891, dtype: object
```

In [76]:
```
import matplotlib.pyplot as plt

# Count the number of occurrences of each cabin assignment
cabin_value_count = titanic_data['Cabin'].value_counts().head(11)

# Plotting
plt.figure(figsize=(13, 7))
cabin_value_count.plot(kind='bar',color='orange')
plt.xlabel('Cabin Assignment')
plt.ylabel('Count')
plt.title('Top 10 Cabin Assignments')
plt.xticks(rotation=45)
plt.show()
```



Handling Missing values in Embarked

In [77]:
```
titanic_data['Embarked'].isna().sum() # 2values
```

Out[77]:
```
0
```

In [78]:
```
titanic_data['Embarked'].unique()
```

Out[78]:
```
array(['S', 'C', 'Q'], dtype=object)
```

In [79]:
```
# Filling missing values in 'Embarked' column with the mode
mode_embarked = titanic_data['Embarked'].mode()[0]
titanic_data['Embarked'].fillna(mode_embarked, inplace=True)
```

In [80]:
```
titanic_data['Embarked'].isna().sum() #clearing the missing values
```

Out[80]:
```
0
```

Checking the Remaining columns in the dataset

PassengerId Column

In [81]:
```
titanic_data['PassengerId']
```

Out[81]:
```
0        1
1        2
2        3
3        4
4        5
      ...
886    887
887    888
888    889
889    890
890    891
Name: PassengerId, Length: 891, dtype: int64
```

In [82]:
```
titanic_data.shape
```

Out[82]:
```
(891, 12)
```

In [83]:
```
titanic_data['PassengerId'].nunique() #we have exact unique row numbers for each column
```

Out[83]:
```
891
```

Survived Column

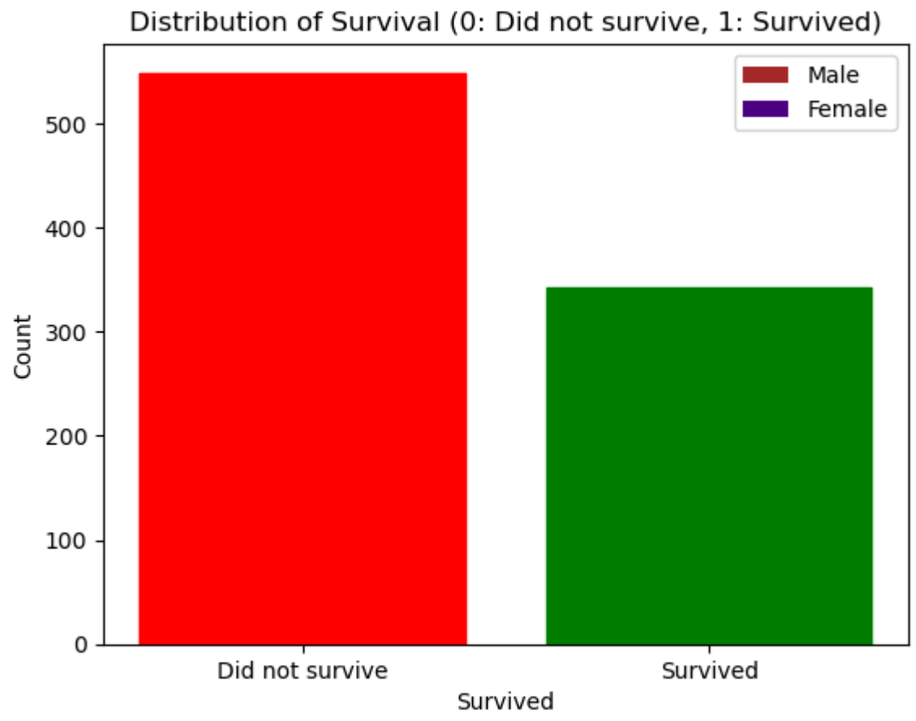
In [84]:
```
titanic_data['Survived']
```

Out[84]:
```
0      0
1      1
2      1
3      1
4      0
      ..
886    0
887    1
888    0
889    1
890    0
Name: Survived, Length: 891, dtype: int64
```

In [85]:
```
titanic_data['Survived'].unique()
```

Out[85]:
```
array([0, 1], dtype=int64)
```

In [86]:
```
from matplotlib.patches import Patch
survived_distribution = titanic_data['Survived'].value_counts()

# Plotting the distribution
bars= plt.bar(survived_distribution.index, survived_distribution.values)
bars[0].set_color('red')
bars[1].set_color('green')
plt.xlabel('Survived')
plt.ylabel('Count')
plt.title('Distribution of Survival (0: Did not survive, 1: Survived)')
plt.legend(handles=handles)
plt.xticks([0, 1], ['Did not survive', 'Survived'])
plt.show()
```



Pclass Column

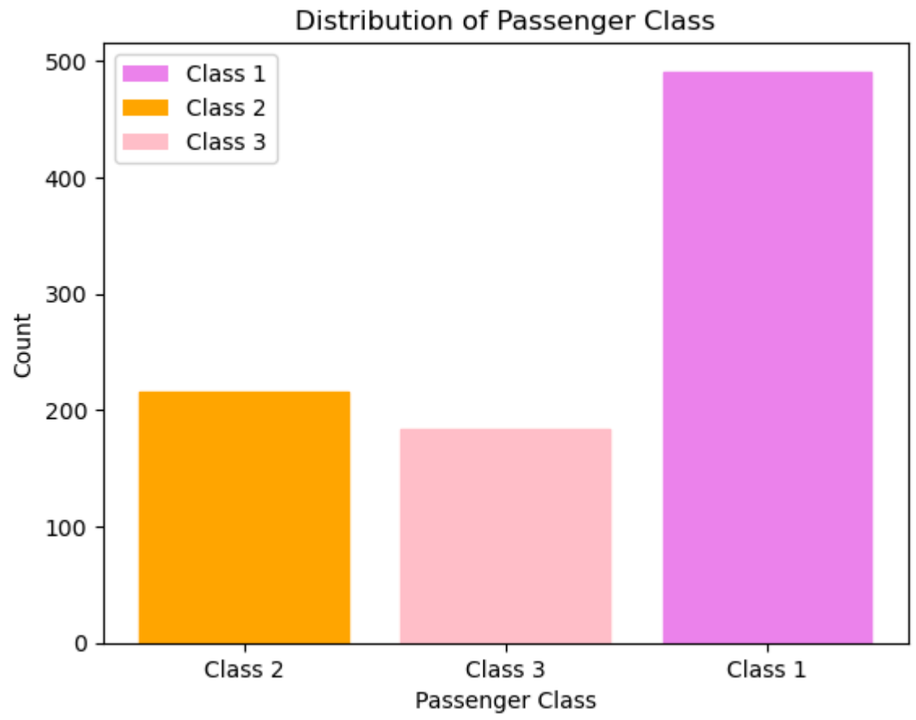
In [87]:
```
titanic_data['Pclass']
```

Out[87]:
```
0      3
1      1
2      3
3      1
4      3
      ..
886    2
887    1
888    3
889    1
890    3
Name: Pclass, Length: 891, dtype: int64
```

In [88]:
```
titanic_data['Pclass'].unique()
```

Out[88]:
```
array([3, 1, 2], dtype=int64)
```

In [89]:
```
from matplotlib.patches import Patch
# Assuming titanic_data is your DataFrame containing the 'Pclass' column
pclass_distribution = titanic_data['Pclass'].value_counts()

# Plotting the distribution
bars=plt.bar(pclass_distribution.index, pclass_distribution.values)
bars[0].set_color('Violet')
bars[1].set_color('Orange')
bars[2].set_color('Pink')
plt.xlabel('Passenger Class')
plt.ylabel('Count')
handles = [
    Patch(facecolor="Violet", label="Class 1"),
    Patch(facecolor="Orange", label="Class 2"),
    Patch(facecolor="Pink", label="Class 3")
]

plt.legend(handles=handles)
plt.title('Distribution of Passenger Class')
plt.xticks(pclass_distribution.index, ['Class 1', 'Class 2', 'Class 3'])
plt.show()
```



Name Column

In [90]:
```
titanic_data['Name']
```

Out[90]:
```
0                              Braund, Mr. Owen Harris
1    Cumings, Mrs. John Bradley (Florence Briggs Th...
2                               Heikkinen, Miss. Laina
3         Futrelle, Mrs. Jacques Heath (Lily May Peel)
4                             Allen, Mr. William Henry
                           ...
886                              Montvila, Rev. Juozas
887                       Graham, Miss. Margaret Edith
888           Johnston, Miss. Catherine Helen "Carrie"
889                              Behr, Mr. Karl Howell
890                                Dooley, Mr. Patrick
Name: Name, Length: 891, dtype: object
```

In [91]:
```
titanic_data['Name'].nunique()
```

Out[91]:
```
891
```

In [92]:
```
titanic_data['Name'].unique()
```

Out[92]:
```
array(['Braund, Mr. Owen Harris',
       'Cumings, Mrs. John Bradley (Florence Briggs Thayer)',
       'Heikkinen, Miss. Laina',
       'Futrelle, Mrs. Jacques Heath (Lily May Peel)',
       'Allen, Mr. William Henry', 'Moran, Mr. James',
       'McCarthy, Mr. Timothy J', 'Palsson, Master. Gosta Leonard',
       'Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg)',
       'Nasser, Mrs. Nicholas (Adele Achem)',
       'Sandstrom, Miss. Marguerite Rut', 'Bonnell, Miss. Elizabeth',
       'Saundercock, Mr. William Henry', 'Andersson, Mr. Anders Johan',
       'Vestrom, Miss. Hulda Amanda Adolfina',
       'Hewlett, Mrs. (Mary D Kingcome) ', 'Rice, Master. Eugene',
       'Williams, Mr. Charles Eugene',
       'Vander Planke, Mrs. Julius (Emelia Maria Vandemoortele)',
       'Masselmani, Mrs. Fatima', 'Fynney, Mr. Joseph J',
       'Beesley, Mr. Lawrence', 'McGowan, Miss. Anna "Annie"',
       'Sloper, Mr. William Thompson', 'Palsson, Miss. Torborg Danira',
       'Asplund, Mrs. Carl Oscar (Selma Augusta Emilia Johansson)',
       'Emir, Mr. Farred Chehab', 'Fortune, Mr. Charles Alexander',
       'O\'Dwyer, Miss. Ellen "Nellie"', 'Todoroff, Mr. Lalio',
       'Uruchurtu, Don. Manuel E',
       'Spencer, Mrs. William Augustus (Marie Eugenie)',
       'Glynn, Miss. Mary Agatha', 'Wheadon, Mr. Edward H',
       'Meyer, Mr. Edgar Joseph', 'Holverson, Mr. Alexander Oskar',
       'Mamee, Mr. Hanna', 'Cann, Mr. Ernest Charles',
       'Vander Planke, Miss. Augusta Maria',
       'Nicola-Yarred, Miss. Jamila',
       'Ahlin, Mrs. Johan (Johanna Persdotter Larsson)',
       'Turpin, Mrs. William John Robert (Dorothy Ann Wonnacott)',
       'Kraeff, Mr. Theodor', 'Laroche, Miss. Simonne Marie Anne Andree',
       'Devaney, Miss. Margaret Delia', 'Rogers, Mr. William John',
       'Lennon, Mr. Denis', "O'Driscoll, Miss. Bridget",
       'Samaan, Mr. Youssef',
       'Arnold-Franchi, Mrs. Josef (Josefine Franchi)',
       'Panula, Master. Juha Niilo', 'Nosworthy, Mr. Richard Cater',
       'Harper, Mrs. Henry Sleeper (Myna Haxtun)',
       'Faunthorpe, Mrs. Lizzie (Elizabeth Anne Wilkinson)',
       'Ostby, Mr. Engelhart Cornelius', 'Woolner, Mr. Hugh',
       'Rugg, Miss. Emily', 'Novel, Mr. Mansouer',
       'West, Miss. Constance Mirium',
       'Goodwin, Master. William Frederick', 'Sirayanian, Mr. Orsen',
       'Icard, Miss. Amelie', 'Harris, Mr. Henry Birkhardt',
       'Skoog, Master. Harald', 'Stewart, Mr. Albert A',
       'Moubarek, Master. Gerios', 'Nye, Mrs. (Elizabeth Ramell)',
       'Crease, Mr. Ernest James', 'Andersson, Miss. Erna Alexandra',
       'Kink, Mr. Vincenz', 'Jenkin, Mr. Stephen Curnow',
       'Goodwin, Miss. Lillian Amy', 'Hood, Mr. Ambrose Jr',
       'Chronopoulos, Mr. Apostolos', 'Bing, Mr. Lee',
       'Moen, Mr. Sigurd Hansen', 'Staneff, Mr. Ivan',
       'Moutal, Mr. Rahamin Haim', 'Caldwell, Master. Alden Gates',
       'Dowdell, Miss. Elizabeth', 'Waelens, Mr. Achille',
       'Sheerlinck, Mr. Jan Baptist', 'McDermott, Miss. Brigdet Delia',
       'Carrau, Mr. Francisco M', 'Ilett, Miss. Bertha',
       'Backstrom, Mrs. Karl Alfred (Maria Mathilda Gustafsson)',
       'Ford, Mr. William Neal', 'Slocovski, Mr. Selman Francis',
       'Fortune, Miss. Mabel Helen', 'Celotti, Mr. Francesco',
       'Christmann, Mr. Emil', 'Andreasson, Mr. Paul Edvin',
       'Chaffee, Mr. Herbert Fuller', 'Dean, Mr. Bertram Frank',
       'Coxon, Mr. Daniel', 'Shorney, Mr. Charles Joseph',
       'Goldschmidt, Mr. George B', 'Greenfield, Mr. William Bertram',
       'Doling, Mrs. John T (Ada Julia Bone)', 'Kantor, Mr. Sinai',
       'Petranec, Miss. Matilda', 'Petroff, Mr. Pastcho ("Pentcho")',
       'White, Mr. Richard Frasar', 'Johansson, Mr. Gustaf Joel',
       'Gustafsson, Mr. Anders Vilhelm', 'Mionoff, Mr. Stoytcho',
```

```
'Salkjelsvik, Miss. Anna Kristine', 'Moss, Mr. Albert Johan',
'Rekic, Mr. Tido', 'Moran, Miss. Bertha',
'Porter, Mr. Walter Chamberlain', 'Zabour, Miss. Hileni',
'Barton, Mr. David John', 'Jussila, Miss. Katriina',
'Attalah, Miss. Malake', 'Pekoniemi, Mr. Edvard',
'Connors, Mr. Patrick', 'Turpin, Mr. William John Robert',
'Baxter, Mr. Quigg Edmond', 'Andersson, Miss. Ellis Anna Maria',
'Hickman, Mr. Stanley George', 'Moore, Mr. Leonard Charles',
'Nasser, Mr. Nicholas', 'Webber, Miss. Susan',
'White, Mr. Percival Wayland', 'Nicola-Yarred, Master. Elias',
'McMahon, Mr. Martin', 'Madsen, Mr. Fridtjof Arne',
'Peter, Miss. Anna', 'Ekstrom, Mr. Johan', 'Drazenoic, Mr. Jozef',
'Coelho, Mr. Domingos Fernandeo',
'Robins, Mrs. Alexander A (Grace Charity Laury)',
'Weisz, Mrs. Leopold (Mathilde Francoise Pede)',
'Sobey, Mr. Samuel James Hayden', 'Richard, Mr. Emile',
'Newsom, Miss. Helen Monypeny', 'Futrelle, Mr. Jacques Heath',
'Osen, Mr. Olaf Elon', 'Giglio, Mr. Victor',
'Boulos, Mrs. Joseph (Sultana)', 'Nysten, Miss. Anna Sofia',
'Hakkarainen, Mrs. Pekka Pietari (Elin Matilda Dolck)',
'Burke, Mr. Jeremiah', 'Andrew, Mr. Edgardo Samuel',
'Nicholls, Mr. Joseph Charles',
'Andersson, Mr. August Edvard ("Wennerstrom")',
'Ford, Miss. Robina Maggie "Ruby"',
'Navratil, Mr. Michel ("Louis M Hoffman")',
'Byles, Rev. Thomas Roussel Davids', 'Bateman, Rev. Robert James',
'Pears, Mrs. Thomas (Edith Wearne)', 'Meo, Mr. Alfonzo',
'van Billiard, Mr. Austin Blyler', 'Olsen, Mr. Ole Martin',
'Williams, Mr. Charles Duane', 'Gilnagh, Miss. Katherine "Katie"',
'Corn, Mr. Harry', 'Smiljanic, Mr. Mile',
'Sage, Master. Thomas Henry', 'Cribb, Mr. John Hatfield',
'Watt, Mrs. James (Elizabeth "Bessie" Inglis Milne)',
'Bengtsson, Mr. John Viktor', 'Calic, Mr. Jovo',
'Panula, Master. Eino Viljami',
'Goldsmith, Master. Frank John William "Frankie"',
'Chibnall, Mrs. (Edith Martha Bowerman)',
'Skoog, Mrs. William (Anna Bernhardina Karlsson)',
'Baumann, Mr. John D', 'Ling, Mr. Lee',
'Van der hoef, Mr. Wyckoff', 'Rice, Master. Arthur',
'Johnson, Miss. Eleanor Ileen', 'Sivola, Mr. Antti Wilhelm',
'Smith, Mr. James Clinch', 'Klasen, Mr. Klas Albin',
'Lefebre, Master. Henry Forbes', 'Isham, Miss. Ann Elizabeth',
'Hale, Mr. Reginald', 'Leonard, Mr. Lionel',
'Sage, Miss. Constance Gladys', 'Pernot, Mr. Rene',
'Asplund, Master. Clarence Gustaf Hugo',
'Becker, Master. Richard F', 'Kink-Heilmann, Miss. Luise Gretchen',
'Rood, Mr. Hugh Roscoe',
'O\'Brien, Mrs. Thomas (Johanna "Hannah" Godfrey)',
'Romaine, Mr. Charles Hallace ("Mr C Rolmane")',
'Bourke, Mr. John', 'Turcin, Mr. Stjepan', 'Pinsky, Mrs. (Rosa)',
'Carbines, Mr. William',
'Andersen-Jensen, Miss. Carla Christine Nielsine',
'Navratil, Master. Michel M',
'Brown, Mrs. James Joseph (Margaret Tobin)',
'Lurette, Miss. Elise', 'Mernagh, Mr. Robert',
'Olsen, Mr. Karl Siegwart Andreas',
'Madigan, Miss. Margaret "Maggie"',
'Yrois, Miss. Henriette ("Mrs Harbeck")',
'Vande Walle, Mr. Nestor Cyriel', 'Sage, Mr. Frederick',
'Johanson, Mr. Jakob Alfred', 'Youseff, Mr. Gerious',
'Cohen, Mr. Gurshon "Gus"', 'Strom, Miss. Telma Matilda',
'Backstrom, Mr. Karl Alfred', 'Albimona, Mr. Nassef Cassem',
'Carr, Miss. Helen "Ellen"', 'Blank, Mr. Henry', 'Ali, Mr. Ahmed',
'Cameron, Miss. Clear Annie', 'Perkin, Mr. John Henry',
```

'Givard, Mr. Hans Kristensen', 'Kiernan, Mr. Philip',
'Newell, Miss. Madeleine', 'Honkanen, Miss. Eliina',
'Jacobsohn, Mr. Sidney Samuel', 'Bazzani, Miss. Albina',
'Harris, Mr. Walter', 'Sunderland, Mr. Victor Francis',
'Bracken, Mr. James H', 'Green, Mr. George Henry',
'Nenkoff, Mr. Christo', 'Hoyt, Mr. Frederick Maxfield',
'Berglund, Mr. Karl Ivar Sven', 'Mellors, Mr. William John',
'Lovell, Mr. John Hall ("Henry")', 'Fahlstrom, Mr. Arne Jonas',
'Lefebre, Miss. Mathilde',
'Harris, Mrs. Henry Birkhardt (Irene Wallach)',
'Larsson, Mr. Bengt Edvin', 'Sjostedt, Mr. Ernst Adolf',
'Asplund, Miss. Lillian Gertrud',
'Leyson, Mr. Robert William Norman',
'Harknett, Miss. Alice Phoebe', 'Hold, Mr. Stephen',
'Collyer, Miss. Marjorie "Lottie"',
'Pengelly, Mr. Frederick William', 'Hunt, Mr. George Henry',
'Zabour, Miss. Thamine', 'Murphy, Miss. Katherine "Kate"',
'Coleridge, Mr. Reginald Charles', 'Maenpaa, Mr. Matti Alexanteri',
'Attalah, Mr. Sleiman', 'Minahan, Dr. William Edward',
'Lindahl, Miss. Agda Thorilda Viktoria',
'Hamalainen, Mrs. William (Anna)', 'Beckwith, Mr. Richard Leonard',
'Carter, Rev. Ernest Courtenay', 'Reed, Mr. James George',
'Strom, Mrs. Wilhelm (Elna Matilda Persson)',
'Stead, Mr. William Thomas', 'Lobb, Mr. William Arthur',
'Rosblom, Mrs. Viktor (Helena Wilhelmina)',
'Touma, Mrs. Darwis (Hanne Youssef Razi)',
'Thorne, Mrs. Gertrude Maybelle', 'Cherry, Miss. Gladys',
'Ward, Miss. Anna', 'Parrish, Mrs. (Lutie Davis)',
'Smith, Mr. Thomas', 'Asplund, Master. Edvin Rojj Felix',
'Taussig, Mr. Emil', 'Harrison, Mr. William', 'Henry, Miss. Delia',
'Reeves, Mr. David', 'Panula, Mr. Ernesti Arvid',
'Persson, Mr. Ernst Ulrik',
'Graham, Mrs. William Thompson (Edith Junkins)',
'Bissette, Miss. Amelia', 'Cairns, Mr. Alexander',
'Tornquist, Mr. William Henry',
'Mellinger, Mrs. (Elizabeth Anne Maidment)',
'Natsch, Mr. Charles H', 'Healy, Miss. Hanora "Nora"',
'Andrews, Miss. Kornelia Theodosia',
'Lindblom, Miss. Augusta Charlotta', 'Parkes, Mr. Francis "Frank"',
'Rice, Master. Eric', 'Abbott, Mrs. Stanton (Rosa Hunt)',
'Duane, Mr. Frank', 'Olsson, Mr. Nils Johan Goransson',
'de Pelsmaeker, Mr. Alfons', 'Dorking, Mr. Edward Arthur',
'Smith, Mr. Richard William', 'Stankovic, Mr. Ivan',
'de Mulder, Mr. Theodore', 'Naidenoff, Mr. Penko',
'Hosono, Mr. Masabumi', 'Connolly, Miss. Kate',
'Barber, Miss. Ellen "Nellie"',
'Bishop, Mrs. Dickinson H (Helen Walton)',
'Levy, Mr. Rene Jacques', 'Haas, Miss. Aloisia',
'Mineff, Mr. Ivan', 'Lewy, Mr. Ervin G', 'Hanna, Mr. Mansour',
'Allison, Miss. Helen Loraine', 'Saalfeld, Mr. Adolphe',
'Baxter, Mrs. James (Helene DeLaudeniere Chaput)',
'Kelly, Miss. Anna Katherine "Annie Kate"', 'McCoy, Mr. Bernard',
'Johnson, Mr. William Cahoone Jr', 'Keane, Miss. Nora A',
'Williams, Mr. Howard Hugh "Harry"',
'Allison, Master. Hudson Trevor', 'Fleming, Miss. Margaret',
'Penasco y Castellana, Mrs. Victor de Satode (Maria Josefa Perez de Soto y Vallejo)',
'Abelson, Mr. Samuel', 'Francatelli, Miss. Laura Mabel',
'Hays, Miss. Margaret Bechstein', 'Ryerson, Miss. Emily Borie',
'Lahtinen, Mrs. William (Anna Sylfven)', 'Hendekovic, Mr. Ignjac',
'Hart, Mr. Benjamin', 'Nilsson, Miss. Helmina Josefina',
'Kantor, Mrs. Sinai (Miriam Sternin)', 'Moraweck, Dr. Ernest',
'Wick, Miss. Mary Natalie',
'Spedden, Mrs. Frederic Oakley (Margaretta Corning Stone)',
'Dennis, Mr. Samuel', 'Danoff, Mr. Yoto',

'Slayter, Miss. Hilda Mary',
'Caldwell, Mrs. Albert Francis (Sylvia Mae Harbaugh)',
'Sage, Mr. George John Jr', 'Young, Miss. Marie Grice',
'Nysveen, Mr. Johan Hansen', 'Ball, Mrs. (Ada E Hall)',
'Goldsmith, Mrs. Frank John (Emily Alice Brown)',
'Hippach, Miss. Jean Gertrude', 'McCoy, Miss. Agnes',
'Partner, Mr. Austen', 'Graham, Mr. George Edward',
'Vander Planke, Mr. Leo Edmondus',
'Frauenthal, Mrs. Henry William (Clara Heinsheimer)',
'Denkoff, Mr. Mitto', 'Pears, Mr. Thomas Clinton',
'Burns, Miss. Elizabeth Margaret', 'Dahl, Mr. Karl Edwart',
'Blackwell, Mr. Stephen Weart', 'Navratil, Master. Edmond Roger',
'Fortune, Miss. Alice Elizabeth', 'Collander, Mr. Erik Gustaf',
'Sedgwick, Mr. Charles Frederick Waddington',
'Fox, Mr. Stanley Hubert', 'Brown, Miss. Amelia "Mildred"',
'Smith, Miss. Marion Elsie',
'Davison, Mrs. Thomas Henry (Mary E Finck)',
'Coutts, Master. William Loch "William"', 'Dimic, Mr. Jovan',
'Odahl, Mr. Nils Martin', 'Williams-Lambert, Mr. Fletcher Fellows',
'Elias, Mr. Tannous', 'Arnold-Franchi, Mr. Josef',
'Yousif, Mr. Wazli', 'Vanden Steen, Mr. Leo Peter',
'Bowerman, Miss. Elsie Edith', 'Funk, Miss. Annie Clemmer',
'McGovern, Miss. Mary', 'Mockler, Miss. Helen Mary "Ellie"',
'Skoog, Mr. Wilhelm', 'del Carlo, Mr. Sebastiano',
'Barbara, Mrs. (Catherine David)', 'Asim, Mr. Adola',
"O'Brien, Mr. Thomas", 'Adahl, Mr. Mauritz Nils Martin',
'Warren, Mrs. Frank Manley (Anna Sophia Atkinson)',
'Moussa, Mrs. (Mantoura Boulos)', 'Jermyn, Miss. Annie',
'Aubart, Mme. Leontine Pauline', 'Harder, Mr. George Achilles',
'Wiklund, Mr. Jakob Alfred', 'Beavan, Mr. William Thomas',
'Ringhini, Mr. Sante', 'Palsson, Miss. Stina Viola',
'Meyer, Mrs. Edgar Joseph (Leila Saks)',
'Landergren, Miss. Aurora Adelia', 'Widener, Mr. Harry Elkins',
'Betros, Mr. Tannous', 'Gustafsson, Mr. Karl Gideon',
'Bidois, Miss. Rosalie', 'Nakid, Miss. Maria ("Mary")',
'Tikkanen, Mr. Juho',
'Holverson, Mrs. Alexander Oskar (Mary Aline Towner)',
'Plotcharsky, Mr. Vasil', 'Davies, Mr. Charles Henry',
'Goodwin, Master. Sidney Leonard', 'Buss, Miss. Kate',
'Sadlier, Mr. Matthew', 'Lehmann, Miss. Bertha',
'Carter, Mr. William Ernest', 'Jansson, Mr. Carl Olof',
'Gustafsson, Mr. Johan Birger', 'Newell, Miss. Marjorie',
'Sandstrom, Mrs. Hjalmar (Agnes Charlotta Bengtsson)',
'Johansson, Mr. Erik', 'Olsson, Miss. Elina',
'McKane, Mr. Peter David', 'Pain, Dr. Alfred',
'Trout, Mrs. William H (Jessie L)', 'Niskanen, Mr. Juha',
'Adams, Mr. John', 'Jussila, Miss. Mari Aina',
'Hakkarainen, Mr. Pekka Pietari', 'Oreskovic, Miss. Marija',
'Gale, Mr. Shadrach', 'Widegren, Mr. Carl/Charles Peter',
'Richards, Master. William Rowe',
'Birkeland, Mr. Hans Martin Monsen', 'Lefebre, Miss. Ida',
'Sdycoff, Mr. Todor', 'Hart, Mr. Henry', 'Minahan, Miss. Daisy E',
'Cunningham, Mr. Alfred Fleming', 'Sundman, Mr. Johan Julian',
'Meek, Mrs. Thomas (Annie Louise Rowley)',
'Drew, Mrs. James Vivian (Lulu Thorne Christian)',
'Silven, Miss. Lyyli Karoliina', 'Matthews, Mr. William John',
'Van Impe, Miss. Catharina', 'Gheorgheff, Mr. Stanio',
'Charters, Mr. David', 'Zimmerman, Mr. Leo',
'Danbom, Mrs. Ernst Gilbert (Anna Sigrid Maria Brogren)',
'Rosblom, Mr. Viktor Richard', 'Wiseman, Mr. Phillippe',
'Clarke, Mrs. Charles V (Ada Maria Winfield)',
'Phillips, Miss. Kate Florence ("Mrs Kate Louise Phillips Marshall")',
'Flynn, Mr. James', 'Pickard, Mr. Berk (Berk Trembisky)',
'Bjornstrom-Steffansson, Mr. Mauritz Hakan',

```
'Thorneycroft, Mrs. Percival (Florence Kate White)',
'Louch, Mrs. Charles Alexander (Alice Adelaide Slow)',
'Kallio, Mr. Nikolai Erland', 'Silvey, Mr. William Baird',
'Carter, Miss. Lucile Polk',
'Ford, Miss. Doolina Margaret "Daisy"',
'Richards, Mrs. Sidney (Emily Hocking)', 'Fortune, Mr. Mark',
'Kvillner, Mr. Johan Henrik Johannesson',
'Hart, Mrs. Benjamin (Esther Ada Bloomfield)', 'Hampe, Mr. Leon',
'Petterson, Mr. Johan Emil', 'Reynaldo, Ms. Encarnacion',
'Johannesen-Bratthammer, Mr. Bernt', 'Dodge, Master. Washington',
'Mellinger, Miss. Madeleine Violet', 'Seward, Mr. Frederic Kimber',
'Baclini, Miss. Marie Catherine', 'Peuchen, Major. Arthur Godfrey',
'West, Mr. Edwy Arthur', 'Hagland, Mr. Ingvald Olai Olsen',
'Foreman, Mr. Benjamin Laventall', 'Goldenberg, Mr. Samuel L',
'Peduzzi, Mr. Joseph', 'Jalsevac, Mr. Ivan',
'Millet, Mr. Francis Davis', 'Kenyon, Mrs. Frederick R (Marion)',
'Toomey, Miss. Ellen', "O'Connor, Mr. Maurice",
'Anderson, Mr. Harry', 'Morley, Mr. William', 'Gee, Mr. Arthur H',
'Milling, Mr. Jacob Christian', 'Maisner, Mr. Simon',
'Goncalves, Mr. Manuel Estanslas', 'Campbell, Mr. William',
'Smart, Mr. John Montgomery', 'Scanlan, Mr. James',
'Baclini, Miss. Helene Barbara', 'Keefe, Mr. Arthur',
'Cacic, Mr. Luka', 'West, Mrs. Edwy Arthur (Ada Mary Worth)',
'Jerwan, Mrs. Amin S (Marie Marthe Thuillard)',
'Strandberg, Miss. Ida Sofia', 'Clifford, Mr. George Quincy',
'Renouf, Mr. Peter Henry', 'Braund, Mr. Lewis Richard',
'Karlsson, Mr. Nils August', 'Hirvonen, Miss. Hildur E',
'Goodwin, Master. Harold Victor',
'Frost, Mr. Anthony Wood "Archie"', 'Rouse, Mr. Richard Henry',
'Turkula, Mrs. (Hedwig)', 'Bishop, Mr. Dickinson H',
'Lefebre, Miss. Jeannie',
'Hoyt, Mrs. Frederick Maxfield (Jane Anne Forby)',
'Kent, Mr. Edward Austin', 'Somerton, Mr. Francis William',
'Coutts, Master. Eden Leslie "Neville"',
'Hagland, Mr. Konrad Mathias Reiersen', 'Windelov, Mr. Einar',
'Molson, Mr. Harry Markland', 'Artagaveytia, Mr. Ramon',
'Stanley, Mr. Edward Roland', 'Yousseff, Mr. Gerious',
'Eustis, Miss. Elizabeth Mussey',
'Shellard, Mr. Frederick William',
'Allison, Mrs. Hudson J C (Bessie Waldo Daniels)',
'Svensson, Mr. Olof', 'Calic, Mr. Petar', 'Canavan, Miss. Mary',
"O'Sullivan, Miss. Bridget Mary", 'Laitinen, Miss. Kristina Sofia',
'Maioni, Miss. Roberta',
'Penasco y Castellana, Mr. Victor de Satode',
'Quick, Mrs. Frederick Charles (Jane Richards)',
'Bradley, Mr. George ("George Arthur Brayton")',
'Olsen, Mr. Henry Margido', 'Lang, Mr. Fang',
'Daly, Mr. Eugene Patrick', 'Webber, Mr. James',
'McGough, Mr. James Robert',
'Rothschild, Mrs. Martin (Elizabeth L. Barrett)',
'Coleff, Mr. Satio', 'Walker, Mr. William Anderson',
'Lemore, Mrs. (Amelia Milley)', 'Ryan, Mr. Patrick',
'Angle, Mrs. William A (Florence "Mary" Agnes Hughes)',
'Pavlovic, Mr. Stefo', 'Perreault, Miss. Anne', 'Vovk, Mr. Janko',
'Lahoud, Mr. Sarkis',
'Hippach, Mrs. Louis Albert (Ida Sophia Fischer)',
'Kassem, Mr. Fared', 'Farrell, Mr. James', 'Ridsdale, Miss. Lucy',
'Farthing, Mr. John', 'Salonen, Mr. Johan Werner',
'Hocking, Mr. Richard George', 'Quick, Miss. Phyllis May',
'Toufik, Mr. Nakli', 'Elias, Mr. Joseph Jr',
'Peter, Mrs. Catherine (Catherine Rizk)', 'Cacic, Miss. Marija',
'Hart, Miss. Eva Miriam', 'Butt, Major. Archibald Willingham',
'LeRoy, Miss. Bertha', 'Risien, Mr. Samuel Beard',
'Frolicher, Miss. Hedwig Margaritha', 'Crosby, Miss. Harriet R',
```

'Andersson, Miss. Ingeborg Constanzia',
'Andersson, Miss. Sigrid Elisabeth', 'Beane, Mr. Edward',
'Douglas, Mr. Walter Donald', 'Nicholson, Mr. Arthur Ernest',
'Beane, Mrs. Edward (Ethel Clarke)', 'Padro y Manent, Mr. Julian',
'Goldsmith, Mr. Frank John', 'Davies, Master. John Morgan Jr',
'Thayer, Mr. John Borland Jr', 'Sharp, Mr. Percival James R',
"O'Brien, Mr. Timothy", 'Leeni, Mr. Fahim ("Philip Zenni")',
'Ohman, Miss. Velin', 'Wright, Mr. George',
'Duff Gordon, Lady. (Lucille Christiana Sutherland) ("Mrs Morgan")',
'Robbins, Mr. Victor', 'Taussig, Mrs. Emil (Tillie Mandelbaum)',
'de Messemaeker, Mrs. Guillaume Joseph (Emma)',
'Morrow, Mr. Thomas Rowan', 'Sivic, Mr. Husein',
'Norman, Mr. Robert Douglas', 'Simmons, Mr. John',
'Meanwell, Miss. (Marion Ogden)', 'Davies, Mr. Alfred J',
'Stoytcheff, Mr. Ilia',
'Palsson, Mrs. Nils (Alma Cornelia Berglund)',
'Doharr, Mr. Tannous', 'Jonsson, Mr. Carl', 'Harris, Mr. George',
'Appleton, Mrs. Edward Dale (Charlotte Lamson)',
'Flynn, Mr. John Irwin ("Irving")', 'Kelly, Miss. Mary',
'Rush, Mr. Alfred George John', 'Patchett, Mr. George',
'Garside, Miss. Ethel',
'Silvey, Mrs. William Baird (Alice Munger)',
'Caram, Mrs. Joseph (Maria Elias)', 'Jussila, Mr. Eiriik',
'Christy, Miss. Julie Rachel',
'Thayer, Mrs. John Borland (Marian Longstreth Morris)',
'Downton, Mr. William James', 'Ross, Mr. John Hugo',
'Paulner, Mr. Uscher', 'Taussig, Miss. Ruth',
'Jarvis, Mr. John Denzil', 'Frolicher-Stehli, Mr. Maxmillian',
'Gilinski, Mr. Eliezer', 'Murdlin, Mr. Joseph',
'Rintamaki, Mr. Matti',
'Stephenson, Mrs. Walter Bertram (Martha Eustis)',
'Elsbury, Mr. William James', 'Bourke, Miss. Mary',
'Chapman, Mr. John Henry', 'Van Impe, Mr. Jean Baptiste',
'Leitch, Miss. Jessie Wills', 'Johnson, Mr. Alfred',
'Boulos, Mr. Hanna',
'Duff Gordon, Sir. Cosmo Edmund ("Mr Morgan")',
'Jacobsohn, Mrs. Sidney Samuel (Amy Frances Christy)',
'Slabenoff, Mr. Petco', 'Harrington, Mr. Charles H',
'Torber, Mr. Ernst William', 'Homer, Mr. Harry ("Mr E Haven")',
'Lindell, Mr. Edvard Bengtsson', 'Karaic, Mr. Milan',
'Daniel, Mr. Robert Williams',
'Laroche, Mrs. Joseph (Juliette Marie Louise Lafargue)',
'Shutes, Miss. Elizabeth W',
'Andersson, Mrs. Anders Johan (Alfrida Konstantia Brogren)',
'Jardin, Mr. Jose Neto', 'Murphy, Miss. Margaret Jane',
'Horgan, Mr. John', 'Brocklebank, Mr. William Alfred',
'Herman, Miss. Alice', 'Danbom, Mr. Ernst Gilbert',
'Lobb, Mrs. William Arthur (Cordelia K Stanlick)',
'Becker, Miss. Marion Louise', 'Gavey, Mr. Lawrence',
'Yasbeck, Mr. Antoni', 'Kimball, Mr. Edwin Nelson Jr',
'Nakid, Mr. Sahid', 'Hansen, Mr. Henry Damsgaard',
'Bowen, Mr. David John "Dai"', 'Sutton, Mr. Frederick',
'Kirkland, Rev. Charles Leonard', 'Longley, Miss. Gretchen Fiske',
'Bostandyeff, Mr. Guentcho', "O'Connell, Mr. Patrick D",
'Barkworth, Mr. Algernon Henry Wilson',
'Lundahl, Mr. Johan Svensson', 'Stahelin-Maeglin, Dr. Max',
'Parr, Mr. William Henry Marsh', 'Skoog, Miss. Mabel',
'Davis, Miss. Mary', 'Leinonen, Mr. Antti Gustaf',
'Collyer, Mr. Harvey', 'Panula, Mrs. Juha (Maria Emilia Ojala)',
'Thorneycroft, Mr. Percival', 'Jensen, Mr. Hans Peder',
'Sagesser, Mlle. Emma', 'Skoog, Miss. Margit Elizabeth',
'Foo, Mr. Choong', 'Baclini, Miss. Eugenie',
'Harper, Mr. Henry Sleeper', 'Cor, Mr. Liudevit',
'Simonius-Blumer, Col. Oberst Alfons', 'Willey, Mr. Edward',

'Stanley, Miss. Amy Zillah Elsie', 'Mitkoff, Mr. Mito',
'Doling, Miss. Elsie', 'Kalvik, Mr. Johannes Halvorsen',
'O\'Leary, Miss. Hanora "Norah"', 'Hegarty, Miss. Hanora "Nora"',
'Hickman, Mr. Leonard Mark', 'Radeff, Mr. Alexander',
'Bourke, Mrs. John (Catherine)', 'Eitemiller, Mr. George Floyd',
'Newell, Mr. Arthur Webster', 'Frauenthal, Dr. Henry William',
'Badt, Mr. Mohamed', 'Colley, Mr. Edward Pomeroy',
'Coleff, Mr. Peju', 'Lindqvist, Mr. Eino William',
'Hickman, Mr. Lewis', 'Butler, Mr. Reginald Fenton',
'Rommetvedt, Mr. Knud Paust', 'Cook, Mr. Jacob',
'Taylor, Mrs. Elmer Zebley (Juliet Cummins Wright)',
'Brown, Mrs. Thomas William Solomon (Elizabeth Catherine Ford)',
'Davidson, Mr. Thornton', 'Mitchell, Mr. Henry Michael',
'Wilhelms, Mr. Charles', 'Watson, Mr. Ennis Hastings',
'Edvardsson, Mr. Gustaf Hjalmar', 'Sawyer, Mr. Frederick Charles',
'Turja, Miss. Anna Sofia',
'Goodwin, Mrs. Frederick (Augusta Tyler)',
'Cardeza, Mr. Thomas Drake Martinez', 'Peters, Miss. Katie',
'Hassab, Mr. Hammad', 'Olsvigen, Mr. Thor Anderson',
'Goodwin, Mr. Charles Edward', 'Brown, Mr. Thomas William Solomon',
'Laroche, Mr. Joseph Philippe Lemercier',
'Panula, Mr. Jaako Arnold', 'Dakic, Mr. Branko',
'Fischer, Mr. Eberhard Thelander',
'Madill, Miss. Georgette Alexandra', 'Dick, Mr. Albert Adrian',
'Karun, Miss. Manca', 'Lam, Mr. Ali', 'Saad, Mr. Khalil',
'Weir, Col. John', 'Chapman, Mr. Charles Henry',
'Kelly, Mr. James', 'Mullens, Miss. Katherine "Katie"',
'Thayer, Mr. John Borland',
'Humblen, Mr. Adolf Mathias Nicolai Olsen',
'Astor, Mrs. John Jacob (Madeleine Talmadge Force)',
'Silverthorne, Mr. Spencer Victor', 'Barbara, Miss. Saiide',
'Gallagher, Mr. Martin', 'Hansen, Mr. Henrik Juul',
'Morley, Mr. Henry Samuel ("Mr Henry Marshall")',
'Kelly, Mrs. Florence "Fannie"',
'Calderhead, Mr. Edward Pennington', 'Cleaver, Miss. Alice',
'Moubarek, Master. Halim Gonios ("William George")',
'Mayne, Mlle. Berthe Antonine ("Mrs de Villiers")',
'Klaber, Mr. Herman', 'Taylor, Mr. Elmer Zebley',
'Larsson, Mr. August Viktor', 'Greenberg, Mr. Samuel',
'Soholt, Mr. Peter Andreas Lauritz Andersen',
'Endres, Miss. Caroline Louise',
'Troutt, Miss. Edwina Celia "Winnie"', 'McEvoy, Mr. Michael',
'Johnson, Mr. Malkolm Joackim',
'Harper, Miss. Annie Jessie "Nina"', 'Jensen, Mr. Svend Lauritz',
'Gillespie, Mr. William Henry', 'Hodges, Mr. Henry Price',
'Chambers, Mr. Norman Campbell', 'Oreskovic, Mr. Luka',
'Renouf, Mrs. Peter Henry (Lillian Jefferys)',
'Mannion, Miss. Margareth', 'Bryhl, Mr. Kurt Arnold Gottfrid',
'Ilmakangas, Miss. Pieta Sofia', 'Allen, Miss. Elisabeth Walton',
'Hassan, Mr. Houssein G N', 'Knight, Mr. Robert J',
'Berriman, Mr. William John', 'Troupiansky, Mr. Moses Aaron',
'Williams, Mr. Leslie', 'Ford, Mrs. Edward (Margaret Ann Watson)',
'Lesurer, Mr. Gustave J', 'Ivanoff, Mr. Kanio',
'Nankoff, Mr. Minko', 'Hawksford, Mr. Walter James',
'Cavendish, Mr. Tyrell William',
'Ryerson, Miss. Susan Parker "Suzette"', 'McNamee, Mr. Neal',
'Stranden, Mr. Juho', 'Crosby, Capt. Edward Gifford',
'Abbott, Mr. Rossmore Edward', 'Sinkkonen, Miss. Anna',
'Marvin, Mr. Daniel Warner', 'Connaghton, Mr. Michael',
'Wells, Miss. Joan', 'Moor, Master. Meier',
'Vande Velde, Mr. Johannes Joseph', 'Jonkoff, Mr. Lalio',
'Herman, Mrs. Samuel (Jane Laver)', 'Hamalainen, Master. Viljo',
'Carlsson, Mr. August Sigfrid', 'Bailey, Mr. Percy Andrew',
'Theobald, Mr. Thomas Leonard',

'Rothes, the Countess. of (Lucy Noel Martha Dyer-Edwards)',
'Garfirth, Mr. John', 'Nirva, Mr. Iisakki Antino Aijo',
'Barah, Mr. Hanna Assi',
'Carter, Mrs. William Ernest (Lucile Polk)',
'Eklund, Mr. Hans Linus', 'Hogeboom, Mrs. John C (Anna Andrews)',
'Brewe, Dr. Arthur Jackson', 'Mangan, Miss. Mary',
'Moran, Mr. Daniel J', 'Gronnestad, Mr. Daniel Danielsen',
'Lievens, Mr. Rene Aime', 'Jensen, Mr. Niels Peder',
'Mack, Mrs. (Mary)', 'Elias, Mr. Dibo',
'Hocking, Mrs. Elizabeth (Eliza Needs)',
'Myhrman, Mr. Pehr Fabian Oliver Malkolm', 'Tobin, Mr. Roger',
'Emanuel, Miss. Virginia Ethel', 'Kilgannon, Mr. Thomas J',
'Robert, Mrs. Edward Scott (Elisabeth Walton McMillan)',
'Ayoub, Miss. Banoura',
'Dick, Mrs. Albert Adrian (Vera Gillespie)',
'Long, Mr. Milton Clyde', 'Johnston, Mr. Andrew G',
'Ali, Mr. William', 'Harmer, Mr. Abraham (David Lishin)',
'Sjoblom, Miss. Anna Sofia', 'Rice, Master. George Hugh',
'Dean, Master. Bertram Vere', 'Guggenheim, Mr. Benjamin',
'Keane, Mr. Andrew "Andy"', 'Gaskell, Mr. Alfred',
'Sage, Miss. Stella Anna', 'Hoyt, Mr. William Fisher',
'Dantcheff, Mr. Ristiu', 'Otter, Mr. Richard',
'Leader, Dr. Alice (Farnham)', 'Osman, Mrs. Mara',
'Ibrahim Shawah, Mr. Yousseff',
'Van Impe, Mrs. Jean Baptiste (Rosalie Paula Govaert)',
'Ponesell, Mr. Martin',
'Collyer, Mrs. Harvey (Charlotte Annie Tate)',
'Carter, Master. William Thornton II',
'Thomas, Master. Assad Alexander', 'Hedman, Mr. Oskar Arvid',
'Johansson, Mr. Karl Johan', 'Andrews, Mr. Thomas Jr',
'Pettersson, Miss. Ellen Natalia', 'Meyer, Mr. August',
'Chambers, Mrs. Norman Campbell (Bertha Griggs)',
'Alexander, Mr. William', 'Lester, Mr. James',
'Slemen, Mr. Richard James', 'Andersson, Miss. Ebba Iris Alfrida',
'Tomlin, Mr. Ernest Portage', 'Fry, Mr. Richard',
'Heininen, Miss. Wendla Maria', 'Mallet, Mr. Albert',
'Holm, Mr. John Fredrik Alexander', 'Skoog, Master. Karl Thorsten',
'Hays, Mrs. Charles Melville (Clara Jennings Gregg)',
'Lulic, Mr. Nikola', 'Reuchlin, Jonkheer. John George',
'Moor, Mrs. (Beila)', 'Panula, Master. Urho Abraham',
'Flynn, Mr. John', 'Lam, Mr. Len', 'Mallet, Master. Andre',
'McCormack, Mr. Thomas Joseph',
'Stone, Mrs. George Nelson (Martha Evelyn)',
'Yasbeck, Mrs. Antoni (Selini Alexander)',
'Richards, Master. George Sibley', 'Saad, Mr. Amin',
'Augustsson, Mr. Albert', 'Allum, Mr. Owen George',
'Compton, Miss. Sara Rebecca', 'Pasic, Mr. Jakob',
'Sirota, Mr. Maurice', 'Chip, Mr. Chang', 'Marechal, Mr. Pierre',
'Alhomaki, Mr. Ilmari Rudolf', 'Mudd, Mr. Thomas Charles',
'Serepeca, Miss. Augusta', 'Lemberopolous, Mr. Peter L',
'Culumovic, Mr. Jeso', 'Abbing, Mr. Anthony',
'Sage, Mr. Douglas Bullen', 'Markoff, Mr. Marin',
'Harper, Rev. John',
'Goldenberg, Mrs. Samuel L (Edwiga Grabowska)',
'Andersson, Master. Sigvard Harald Elias', 'Svensson, Mr. Johan',
'Boulos, Miss. Nourelain', 'Lines, Miss. Mary Conover',
'Carter, Mrs. Ernest Courtenay (Lilian Hughes)',
'Aks, Mrs. Sam (Leah Rosen)',
'Wick, Mrs. George Dennick (Mary Hitchcock)',
'Daly, Mr. Peter Denis ', 'Baclini, Mrs. Solomon (Latifa Qurban)',
'Razi, Mr. Raihed', 'Hansen, Mr. Claus Peter',
'Giles, Mr. Frederick Edward',
'Swift, Mrs. Frederick Joel (Margaret Welles Barron)',
'Sage, Miss. Dorothy Edith "Dolly"', 'Gill, Mr. John William',

```
'Bystrom, Mrs. (Karolina)', 'Duran y More, Miss. Asuncion',
'Roebling, Mr. Washington Augustus II',
'van Melkebeke, Mr. Philemon', 'Johnson, Master. Harold Theodor',
'Balkic, Mr. Cerin',
'Beckwith, Mrs. Richard Leonard (Sallie Monypeny)',
'Carlsson, Mr. Frans Olof', 'Vander Cruyssen, Mr. Victor',
'Abelson, Mrs. Samuel (Hannah Wizosky)',
'Najib, Miss. Adele Kiamie "Jane"',
'Gustafsson, Mr. Alfred Ossian', 'Petroff, Mr. Nedelio',
'Laleff, Mr. Kristo',
'Potter, Mrs. Thomas Jr (Lily Alexenia Wilson)',
'Shelley, Mrs. William (Imanita Parrish Hall)',
'Markun, Mr. Johann', 'Dahlberg, Miss. Gerda Ulrika',
'Banfield, Mr. Frederick James', 'Sutehall, Mr. Henry Jr',
'Rice, Mrs. William (Margaret Norton)', 'Montvila, Rev. Juozas',
'Graham, Miss. Margaret Edith',
'Johnston, Miss. Catherine Helen "Carrie"',
'Behr, Mr. Karl Howell', 'Dooley, Mr. Patrick'], dtype=object)
```

In [93]:
```
titanic_data['Name'].value_counts()
```

Out[93]:
```
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 Column

In [94]:
```
titanic_data['Sex']
```

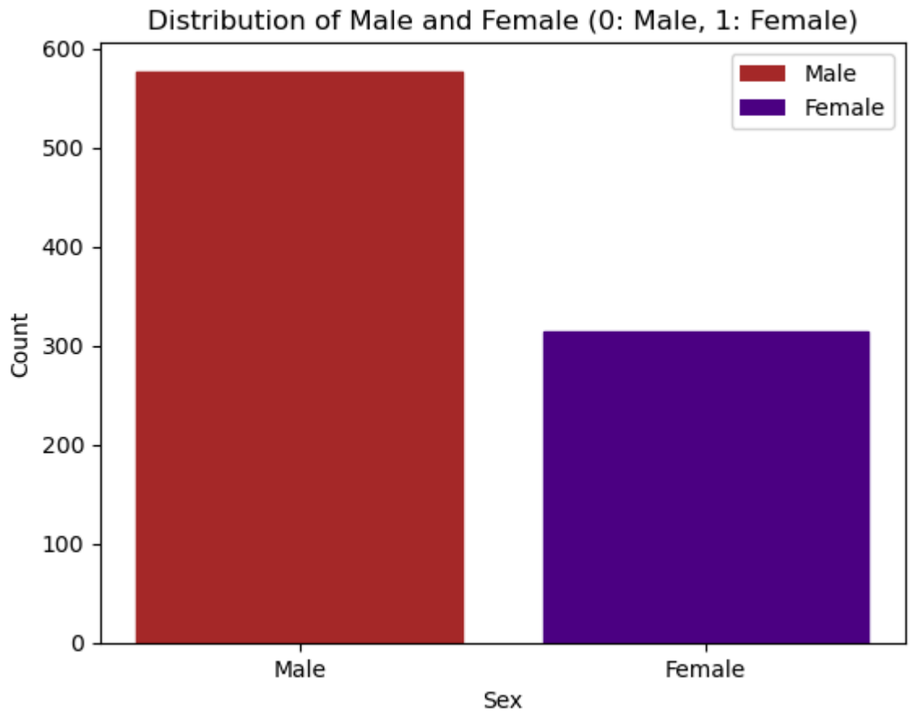
Out[94]:
```
0        male
1      female
2      female
3      female
4        male
       ...
886      male
887    female
888    female
889      male
890      male
Name: Sex, Length: 891, dtype: object
```

In [95]:
```
titanic_data['Sex'].unique()
```

Out[95]:
```
array(['male', 'female'], dtype=object)
```

In [96]:
```
from matplotlib.patches import Patch
sex_distribution = titanic_data['Sex'].value_counts()

# Plotting the distribution
bars=plt.bar(sex_distribution.index, sex_distribution.values)
bars[0].set_color('Brown')
bars[1].set_color('indigo')
plt.xlabel('Sex')
```

```
plt.ylabel('Count')
handles = [
    Patch(facecolor="Brown", label="Male"),
    Patch(facecolor="indigo", label="Female")
]

plt.legend(handles=handles)
plt.title('Distribution of Male and Female (0: Male, 1: Female)')
plt.xticks([0, 1], ['Male', 'Female'])
plt.show()
```



SibSp Column

In [97]:
```
titanic_data['SibSp']
```

Out[97]:
```
0      1
1      1
2      0
3      1
4      0
      ..
886    0
887    0
888    1
889    0
890    0
Name: SibSp, Length: 891, dtype: int64
```

In [98]:
```
titanic_data['SibSp'].unique()
```

Out[98]:
```
array([1, 0, 3, 4, 2, 5, 8], dtype=int64)
```

In [99]:
```
sibsp_distribution = titanic_data['SibSp'].value_counts().sort_index()

# Plotting the distribution
```

```
plt.bar(sibsp_distribution.index, sibsp_distribution.values, color='green')
plt.xlabel('Number of Siblings/Spouses')
plt.ylabel('Count')
plt.title('Distribution of Number of Siblings/Spouses')
plt.xticks(sibsp_distribution.index)
plt.show()
```

Parch Column

In [101…
```
titanic_data['Parch']
```

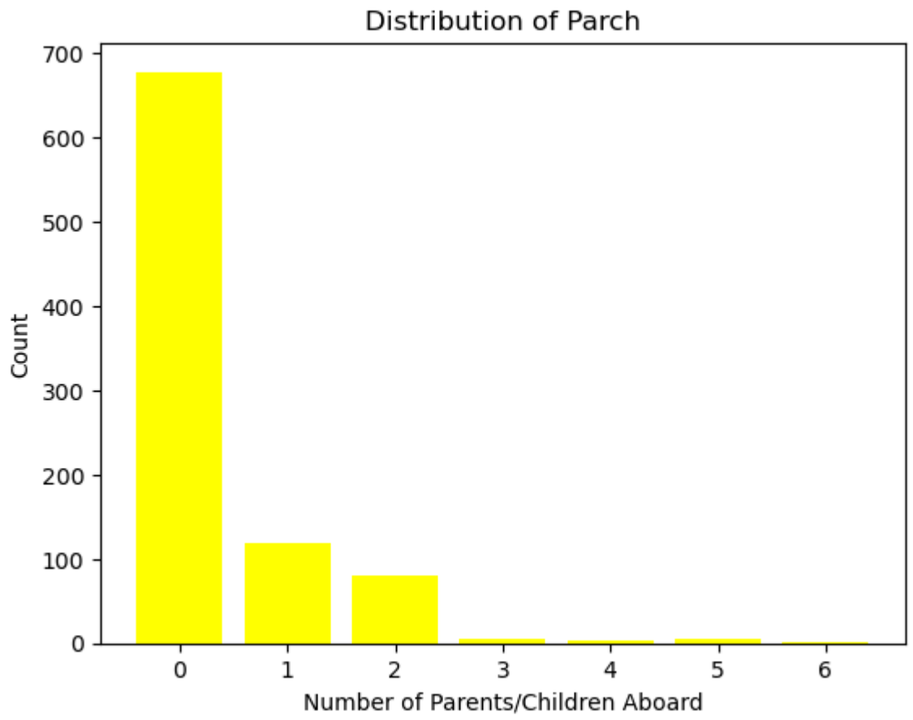
Out[101]:
```
0      0
1      0
2      0
3      0
4      0
      ..
886    0
887    0
888    2
889    0
890    0
Name: Parch, Length: 891, dtype: int64
```

In [102…
```
titanic_data['Parch'].unique()
```

Out[102]:
```
array([0, 1, 2, 5, 3, 4, 6], dtype=int64)
```

In [103…
```
 #Count the occurrences of each unique value in 'Parch'
parch_counts = titanic_data['Parch'].value_counts()

# Create a bar plot
plt.bar(parch_counts.index, parch_counts.values,color='yellow')

# Add labels and title
```

```
plt.xlabel('Number of Parents/Children Aboard')
plt.ylabel('Count')
plt.title('Distribution of Parch')

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



In [104…
```
titanic_data['Ticket']
```

Out[104]:
```
0            A/5 21171
1             PC 17599
2     STON/O2. 3101282
3               113803
4               373450
            ...
886             211536
887             112053
888         W./C. 6607
889             111369
890             370376
Name: Ticket, Length: 891, dtype: object
```

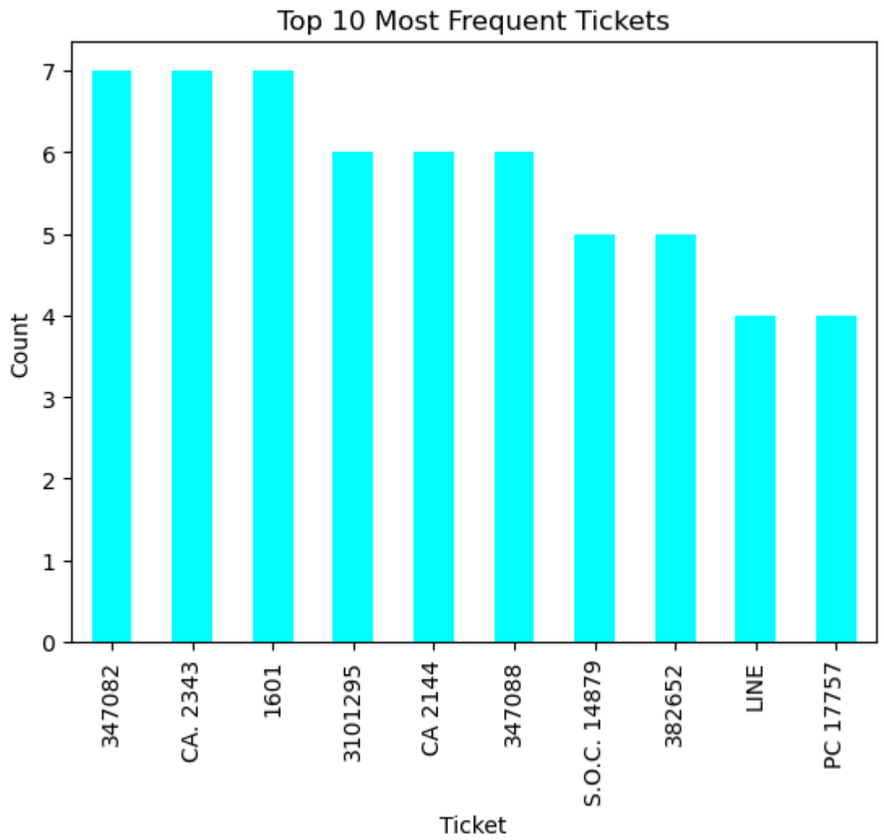
Ticket Column

In [106…
```
titanic_data['Ticket'].nunique()
```

Out[106]:
```
681
```

In [115…
```
from matplotlib.patches import Patch

# Get the top 10 most frequent ticket values
top_tickets = titanic_data['Ticket'].value_counts().head(10)

# Create a bar plot
top_tickets.plot(kind='bar', color='Cyan')
```

```
# Add labels and title
plt.xlabel('Ticket')
plt.ylabel('Count')
plt.title('Top 10 Most Frequent Tickets')

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



In [116…
```
titanic_data.columns
```

Out[116]:
```
Index(['PassengerId', 'Survived', 'Pclass', 'Name', 'Sex', 'Age', 'SibSp',
       'Parch', 'Ticket', 'Fare', 'Cabin', 'Embarked'],
      dtype='object')
```

Fare Column

In [118…
```
titanic_data['Fare']
```

Out[118]:
```
0       7.2500
1      71.2833
2       7.9250
3      53.1000
4       8.0500
        ...
886    13.0000
887    30.0000
888    23.4500
889    30.0000
890     7.7500
Name: Fare, Length: 891, dtype: float64
```

In [119…
```
titanic_data['Fare'].unique()
```

Out[119]:
```
array([  7.25  ,  71.2833,   7.925 ,  53.1   ,   8.05  ,   8.4583,
        51.8625,  21.075 ,  11.1333,  30.0708,  16.7   ,  26.55  ,
        31.275 ,   7.8542,  16.    ,  29.125 ,  13.    ,  18.    ,
         7.225 ,  26.    ,   8.0292,  35.5   ,  31.3875, 263.    ,
         7.8792,   7.8958,  27.7208, 146.5208,   7.75  ,  10.5   ,
        82.1708,  52.    ,   7.2292,  11.2417,   9.475 ,  21.    ,
        41.5792,  15.5   ,  21.6792,  17.8   ,  39.6875,   7.8   ,
        76.7292,  61.9792,  27.75  ,  46.9   ,  80.    ,  83.475 ,
        27.9   ,  15.2458,   8.1583,   8.6625,  73.5   ,  14.4542,
        56.4958,   7.65  ,  29.    ,  12.475 ,   9.    ,   9.5   ,
         7.7875,  47.1   ,  15.85  ,  34.375 ,  61.175 ,  20.575 ,
        34.6542,  63.3583,  23.    ,  77.2875,   8.6542,   7.775 ,
        24.15  ,   9.825 ,  14.4583, 247.5208,   7.1417,  22.3583,
         6.975 ,   7.05  ,  14.5   ,  15.0458,  26.2833,   9.2167,
        79.2   ,   6.75  ,  11.5   ,  36.75  ,   7.7958,  12.525 ,
        66.6   ,   7.3125,  61.3792,   7.7333,  69.55  ,  16.1   ,
        15.75  ,  20.525 ,  55.    ,  25.925 ,  33.5   ,  30.6958,
        25.4667,  28.7125,   0.    ,  15.05  ,  39.    ,  22.025 ,
        50.    ,   8.4042,   6.4958,  10.4625,  18.7875,  31.    ,
       113.275 ,  27.    ,  76.2917,  90.    ,   9.35  ,  13.5   ,
         7.55  ,  26.25  ,  12.275 ,   7.125 ,  52.5542,  20.2125,
        86.5   , 512.3292,  79.65  , 153.4625, 135.6333,  19.5   ,
        29.7   ,  77.9583,  20.25  ,  78.85  ,  91.0792,  12.875 ,
         8.85  , 151.55  ,  30.5   ,  23.25  ,  12.35  , 110.8833,
       108.9   ,  24.    ,  56.9292,  83.1583, 262.375 ,  14.    ,
       164.8667, 134.5   ,   6.2375,  57.9792,  28.5   , 133.65  ,
        15.9   ,   9.225 ,  35.    ,  75.25  ,  69.3   ,  55.4417,
       211.5   ,   4.0125, 227.525 ,  15.7417,   7.7292,  12.    ,
       120.    ,  12.65  ,  18.75  ,   6.8583,  32.5   ,   7.875 ,
        14.4   ,  55.9   ,   8.1125,  81.8583,  19.2583,  19.9667,
        89.1042,  38.5   ,   7.725 ,  13.7917,   9.8375,   7.0458,
         7.5208,  12.2875,   9.5875,  49.5042,  78.2667,  15.1   ,
         7.6292,  22.525 ,  26.2875,  59.4   ,   7.4958,  34.0208,
        93.5   , 221.7792, 106.425 ,  49.5   ,  71.    ,  13.8625,
         7.8292,  39.6   ,  17.4   ,  51.4792,  26.3875,  30.    ,
        40.125 ,   8.7125,  15.    ,  33.    ,  42.4   ,  15.55  ,
        65.    ,  32.3208,   7.0542,   8.4333,  25.5875,   9.8417,
         8.1375,  10.1708, 211.3375,  57.    ,  13.4167,   7.7417,
         9.4833,   7.7375,   8.3625,  23.45  ,  25.9292,   8.6833,
         8.5167,   7.8875,  37.0042,   6.45  ,   6.95  ,   8.3   ,
         6.4375,  39.4   ,  14.1083,  13.8583,  50.4958,   5.    ,
         9.8458,  10.5167])
```

Normalizing fare column for readability

In [120…
```
titanic_data['Fare'] = titanic_data['Fare'].apply(lambda x: "${:.2f}".format(x))
```

```
# Displaying the formatted 'Fare' column
print(titanic_data['Fare'])
```

```
0       $7.25
1      $71.28
2       $7.92
3      $53.10
4       $8.05
        ...
886    $13.00
887    $30.00
888    $23.45
889    $30.00
890     $7.75
Name: Fare, Length: 891, dtype: object
```

In [121…

```
titanic_data
```

Out[121]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22	1	0	A/5 21171	$7.25	NaN	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38	1	0	PC 17599	$71.28	C85	C
2	3	1	3	Heikkinen, Miss. Laina	female	26	0	0	STON/O2. 3101282	$7.92	NaN	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35	1	0	113803	$53.10	C123	S
4	5	0	3	Allen, Mr. William Henry	male	35	0	0	373450	$8.05	NaN	S
...	...	...	...	...	...	...	...	...	...	...	...	...
886	887	0	2	Montvila, Rev. Juozas	male	27	0	0	211536	$13.00	NaN	S
887	888	1	1	Graham, Miss. Margaret Edith	female	19	0	0	112053	$30.00	B42	S
888	889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	23	1	2	W./C. 6607	$23.45	NaN	S
889	890	1	1	Behr, Mr. Karl Howell	male	26	0	0	111369	$30.00	C148	C
890	891	0	3	Dooley, Mr. Patrick	male	32	0	0	370376	$7.75	NaN	Q

891 rows × 12 columns

Data Analysis : Insights from the given dataset

Survival Rate of the Passengers

In [122…

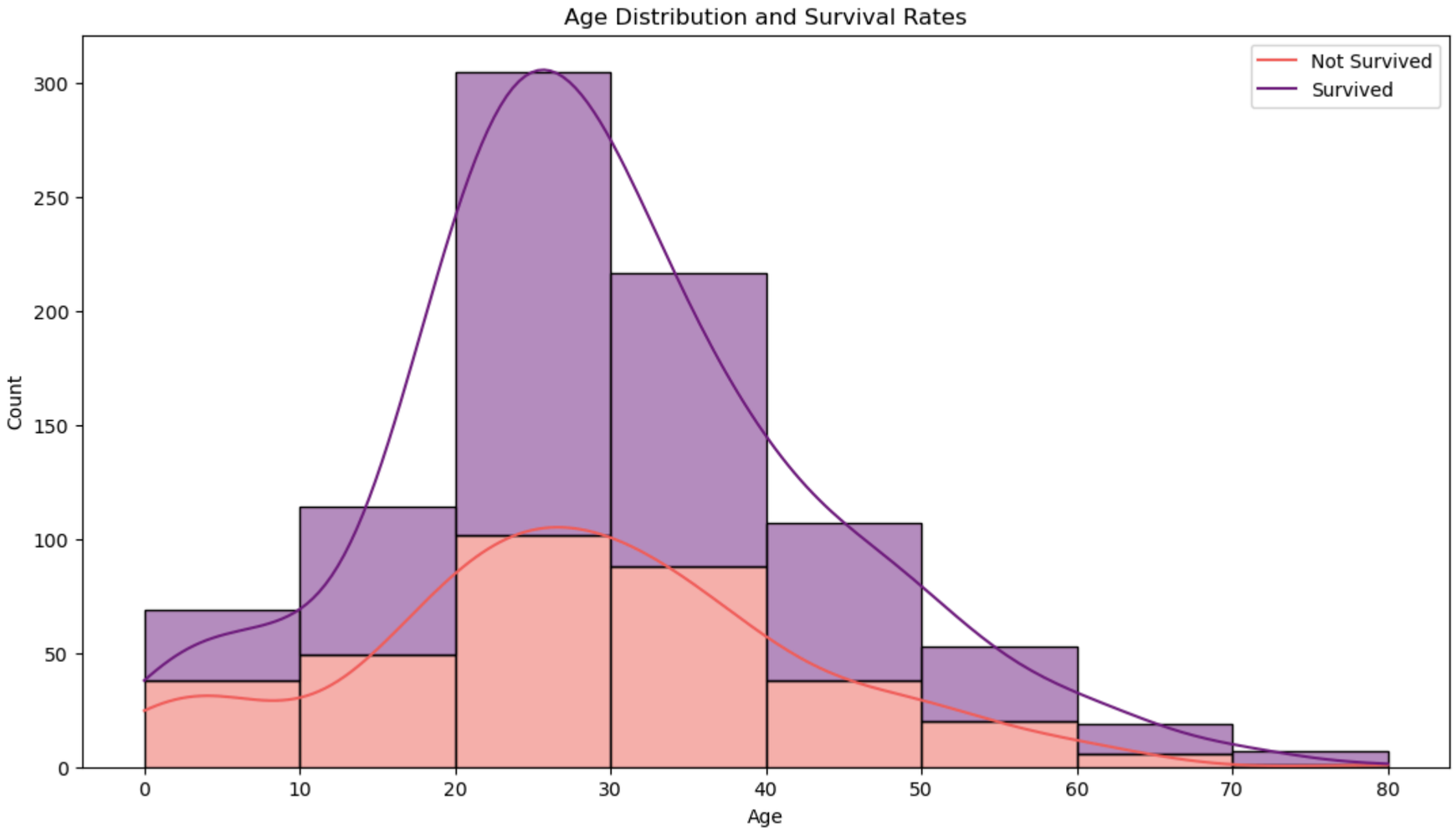
```
# Count total number of passengers
total_passengers = len(titanic_data)

# Count number of survivors
survivors = titanic_data['Survived'].sum()

# Calculate survival rate
survival_rate = (survivors / total_passengers) * 100

print("Survival Rate: {:.2f}%".format(survival_rate))
```

```
Survival Rate: 38.38%
```

Survival Rate by age groups

In [124…

```
import matplotlib.pyplot as plt
import seaborn as sns

# Define age groups
age_bins = [0, 10, 20, 30, 40, 50, 60, 70, 80]
age_labels = ['0-10', '11-20', '21-30', '31-40', '41-50', '51-60', '61-70', '71-80']

# Categorize passengers into age groups
titanic_data['AgeGroup'] = pd.cut(titanic_data['Age'], bins=age_bins, labels=age_labels)

# Calculate survival rate for each age group
survival_by_age = titanic_data.groupby('AgeGroup')['Survived'].mean() * 100

# Plot age distribution and survival rates
plt.figure(figsize=(13, 7))
sns.histplot(data=titanic_data, x='Age', bins=age_bins, kde=True, hue='Survived', multiple='stack', palette='magma')
plt.xlabel('Age')
plt.ylabel('Count')
plt.title('Age Distribution and Survival Rates')
plt.legend(labels=['Not Survived', 'Survived'], loc='upper right')
plt.show()

# Print survival rate by age group
print("Survival Rate by Age Group:")
print(survival_by_age)
```



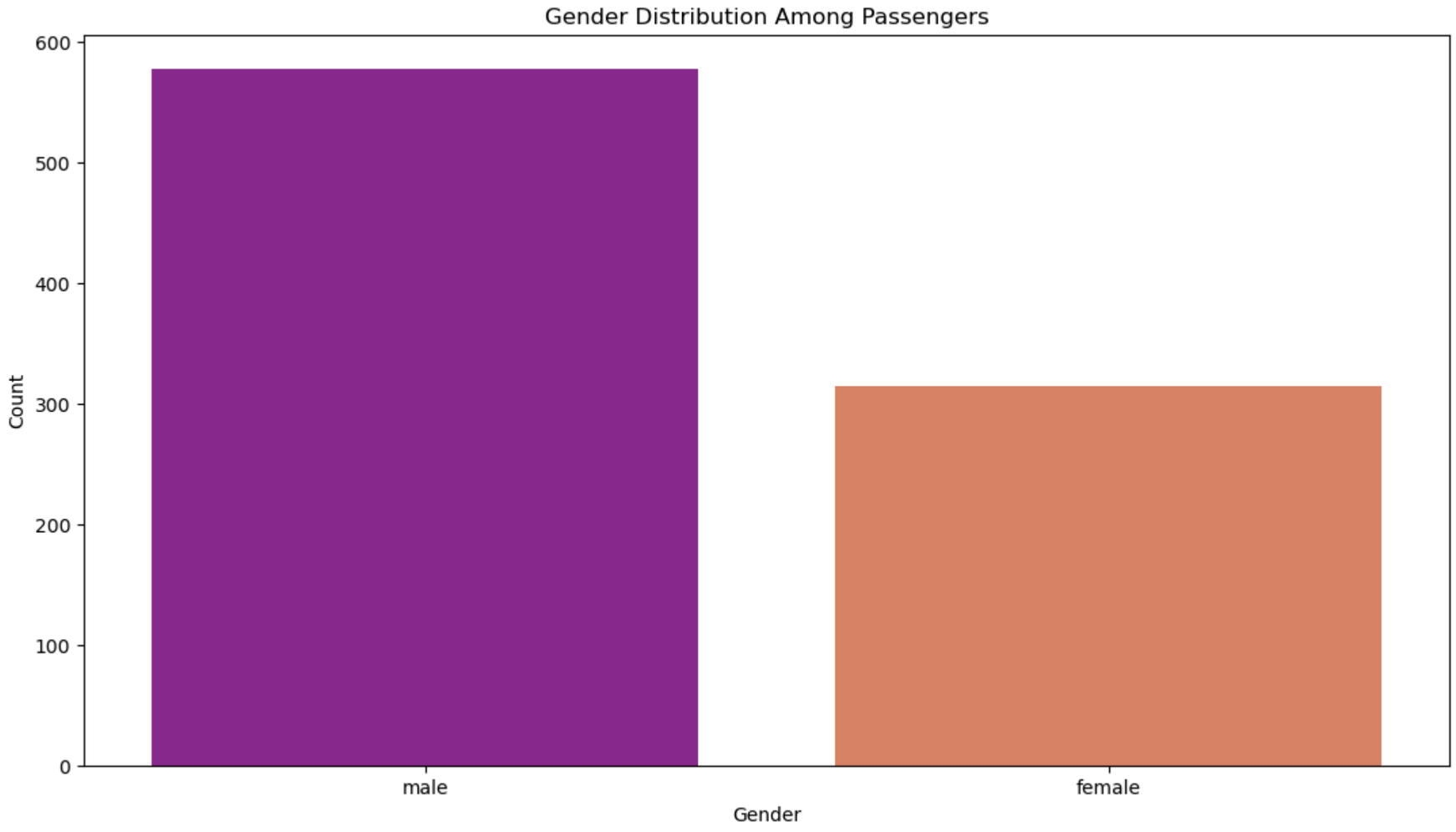

```
Survival Rate by Age Group:
AgeGroup
0-10     49.253731
11-20    40.625000
21-30    33.639144
31-40    42.187500
41-50    36.274510
51-60    36.956522
61-70    22.222222
71-80    25.000000
Name: Survived, dtype: float64
```

Survival Rate By Gender

In [126…

```python
import matplotlib.pyplot as plt
import seaborn as sns

# Analyze gender distribution
gender_distribution = titanic_data['Sex'].value_counts()

# Calculate survival rate for each gender
survival_by_gender = titanic_data.groupby('Sex')['Survived'].mean() * 100

# Plot gender distribution
plt.figure(figsize=(13, 7))
```

```
sns.countplot(data=titanic_data, x='Sex', palette='plasma')
plt.xlabel('Gender')
plt.ylabel('Count')
plt.title('Gender Distribution Among Passengers')
plt.show()

# Print survival rate by gender
print("Survival Rate by Gender:")
print(survival_by_gender)
```



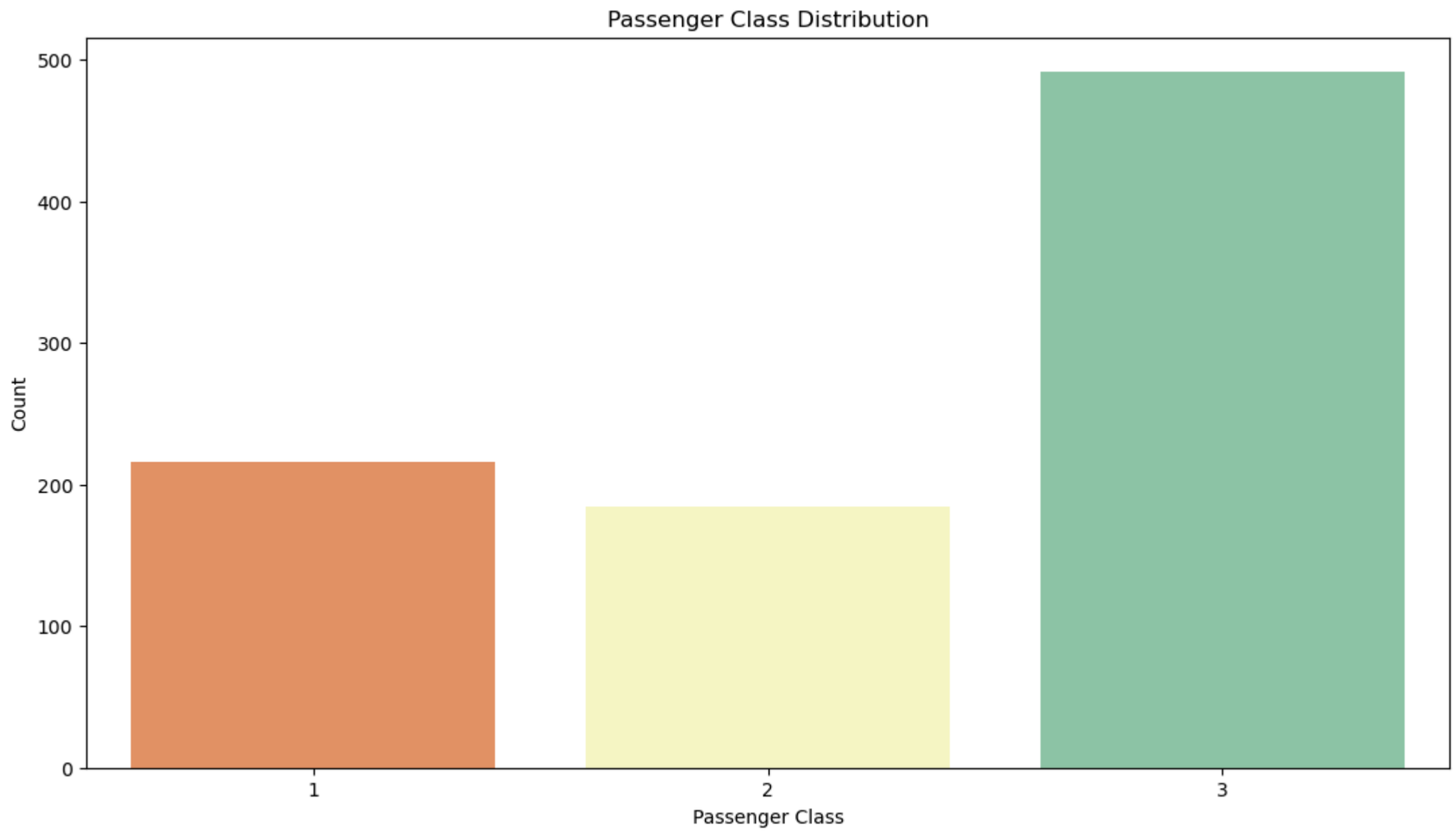

```
Survival Rate by Gender:
Sex
female    74.203822
male      18.890815
Name: Survived, dtype: float64
```

Survival Rate by Passenger Class

In [129…

```
import matplotlib.pyplot as plt
import seaborn as sns

# Analyze class distribution
class_distribution = titanic_data['Pclass'].value_counts()

# Calculate survival rate for each class
```

```
survival_by_class = titanic_data.groupby('Pclass')['Survived'].mean() * 100

# Plot class distribution
plt.figure(figsize=(13, 7))
sns.countplot(data=titanic_data, x='Pclass', palette='Spectral')
plt.xlabel('Passenger Class')
plt.ylabel('Count')
plt.title('Passenger Class Distribution')
plt.show()

# Print survival rate by class
print("Survival Rate by Passenger Class:")
print(survival_by_class)
```




```
Survival Rate by Passenger Class:
Pclass
1    62.962963
2    47.282609
3    24.236253
Name: Survived, dtype: float64
```

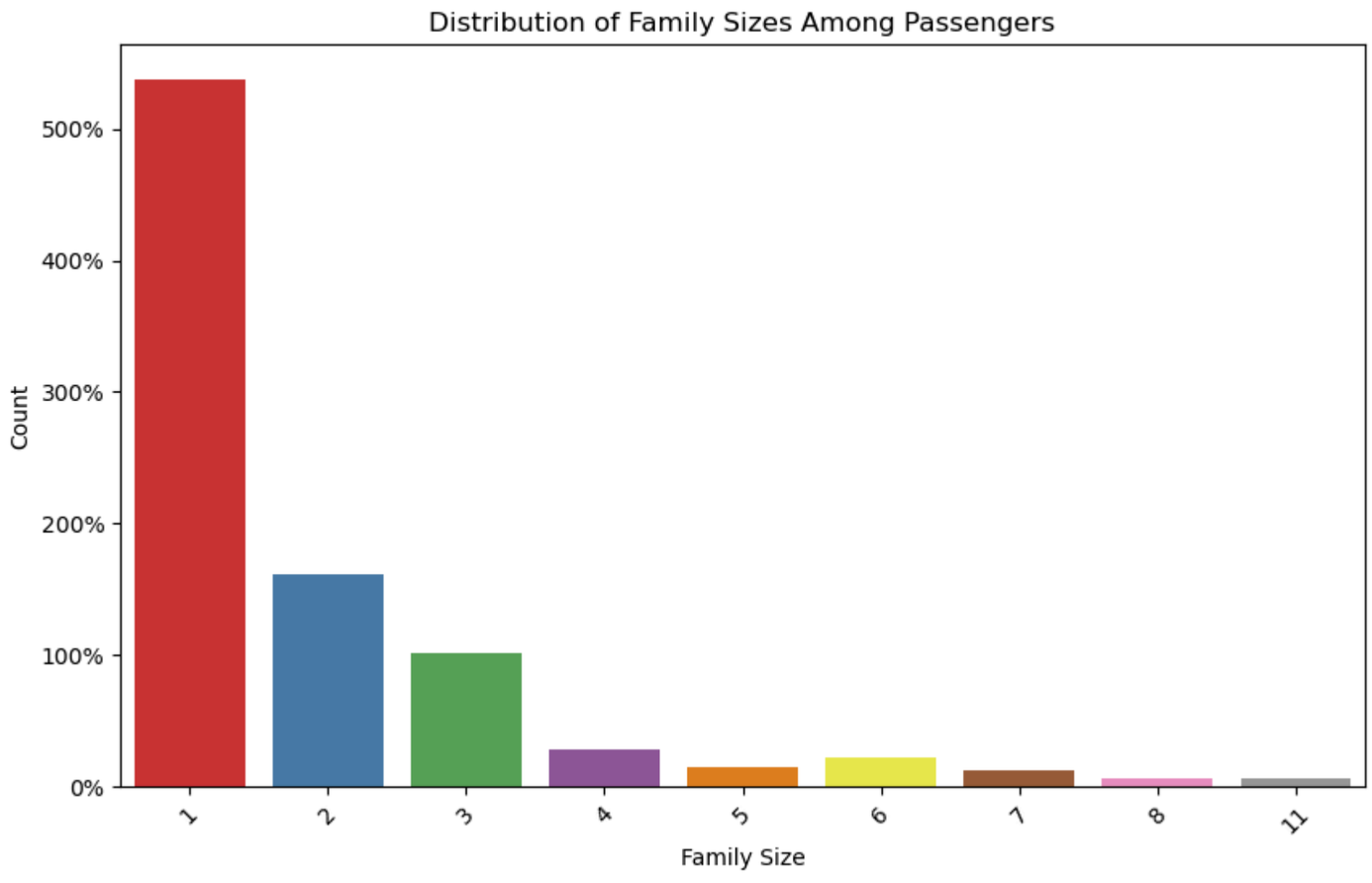
Family Size Analysis

In [134…

```
# Calculate family size
titanic_data['FamilySize'] = titanic_data['SibSp'] + titanic_data['Parch'] + 1
```

```python
# Analyze distribution of family sizes
family_size_distribution = titanic_data['FamilySize'].value_counts().sort_index()

# Calculate survival rate based on family size
survival_by_family_size = titanic_data.groupby('FamilySize')['Survived'].mean() * 100

# Plot distribution of family sizes
plt.figure(figsize=(10, 6))
sns.barplot(x=family_size_distribution.index, y=family_size_distribution.values, palette='Set1')
plt.xlabel('Family Size')
plt.ylabel('Count')
plt.title('Distribution of Family Sizes Among Passengers')
plt.xticks(rotation=45)

# Change y-axis labels to percentages
plt.gca().set_yticklabels(['{:.0f}%'.format(x) for x in plt.gca().get_yticks()])

plt.show()

# Print survival rate by family size
print("Survival Rate by Family Size:")
print(survival_by_family_size)
```

```
C:\Users\Ayush\AppData\Local\Temp\ipykernel_11808\1780016386.py:19: UserWarning: FixedFormatter should only be used together with FixedLocator
  plt.gca().set_yticklabels(['{:.0f}%'.format(x) for x in plt.gca().get_yticks()])
```



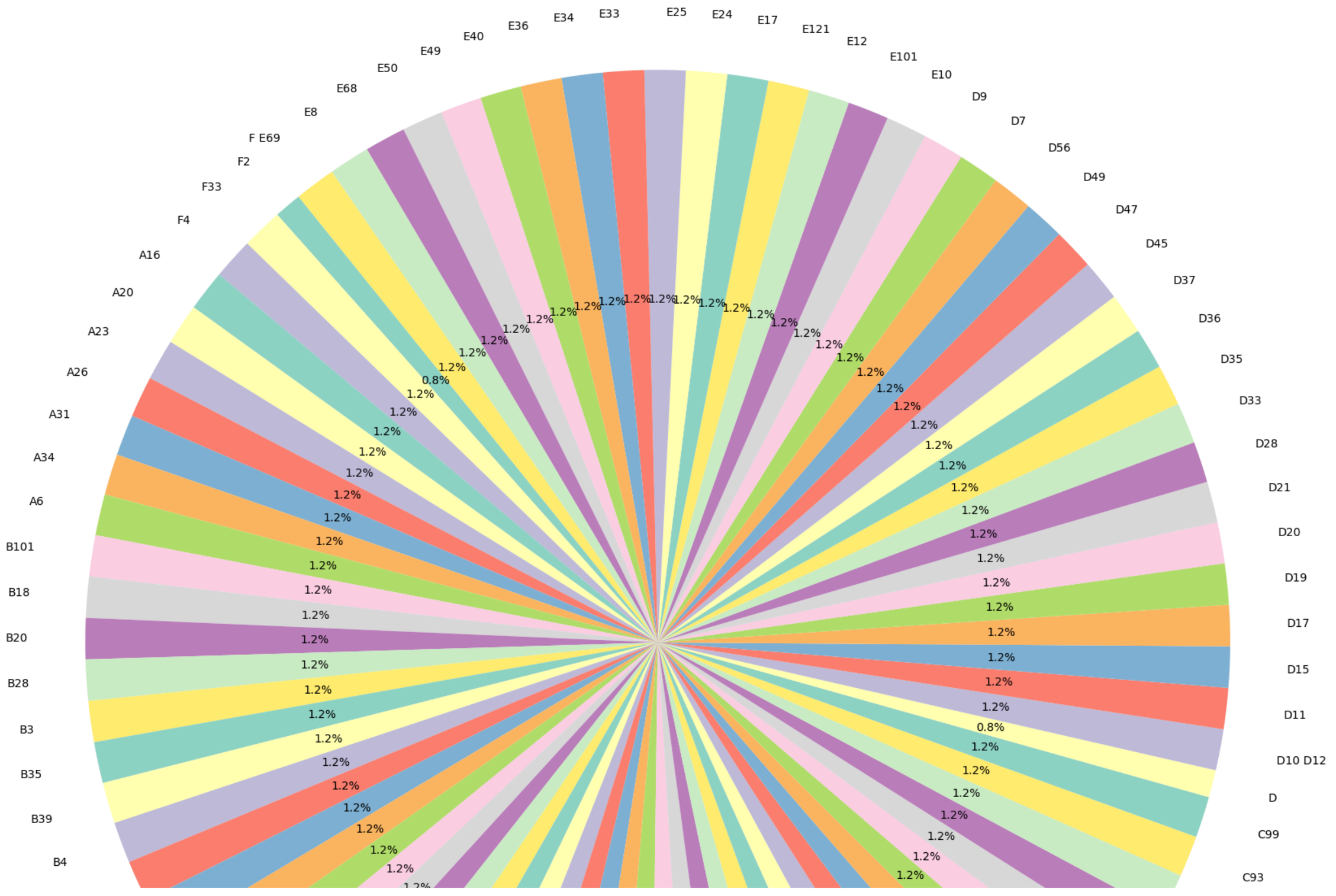
```
Survival Rate by Family Size:
FamilySize
1     30.353818
2     55.279503
3     57.843137
4     72.413793
5     20.000000
6     13.636364
7     33.333333
8      0.000000
11     0.000000
Name: Survived, dtype: float64
```

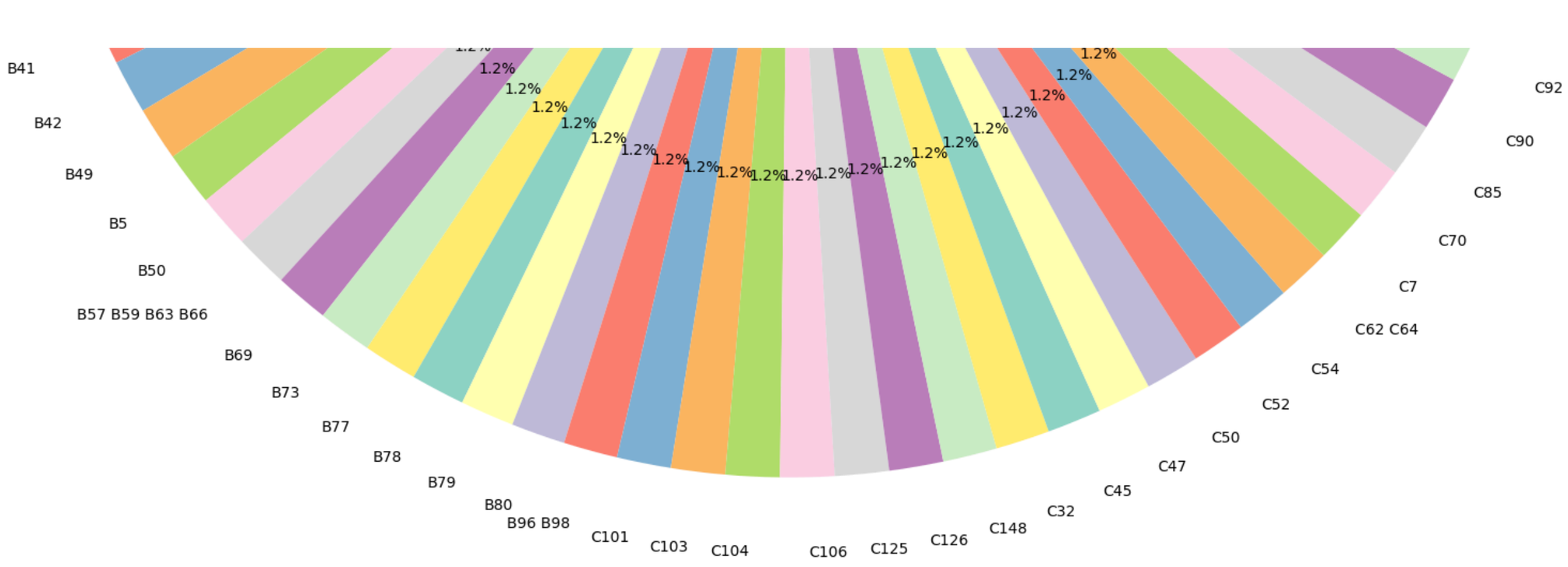
High survival Rate Cabins

In [165…

```
cabin_survival_rate = titanic_data.groupby('Cabin')['Survived'].mean() * 100

# Filter out cabins with survival rates less than a certain threshold (e.g., 50%)
high_survival_cabins = cabin_survival_rate[cabin_survival_rate > 50]

# Plot the survival rate for selected cabins in a pie chart
plt.figure(figsize=(20, 20))
plt.pie(high_survival_cabins, labels=high_survival_cabins.index, autopct='%1.1f%%', startangle=140, colors=sns.color_palette('Set3', len(high_survival_cabins)))
plt.title('Survival Rate by Cabin (Cabins with >50% Survival Rate)', fontsize=25)
plt.subplots_adjust(top=1.0)
plt.axis('equal')  # Equal aspect ratio ensures that pie is drawn as a circle.
plt.show()
```

Survival Rate by Cabin (Cabins with >50% Survival Rate)

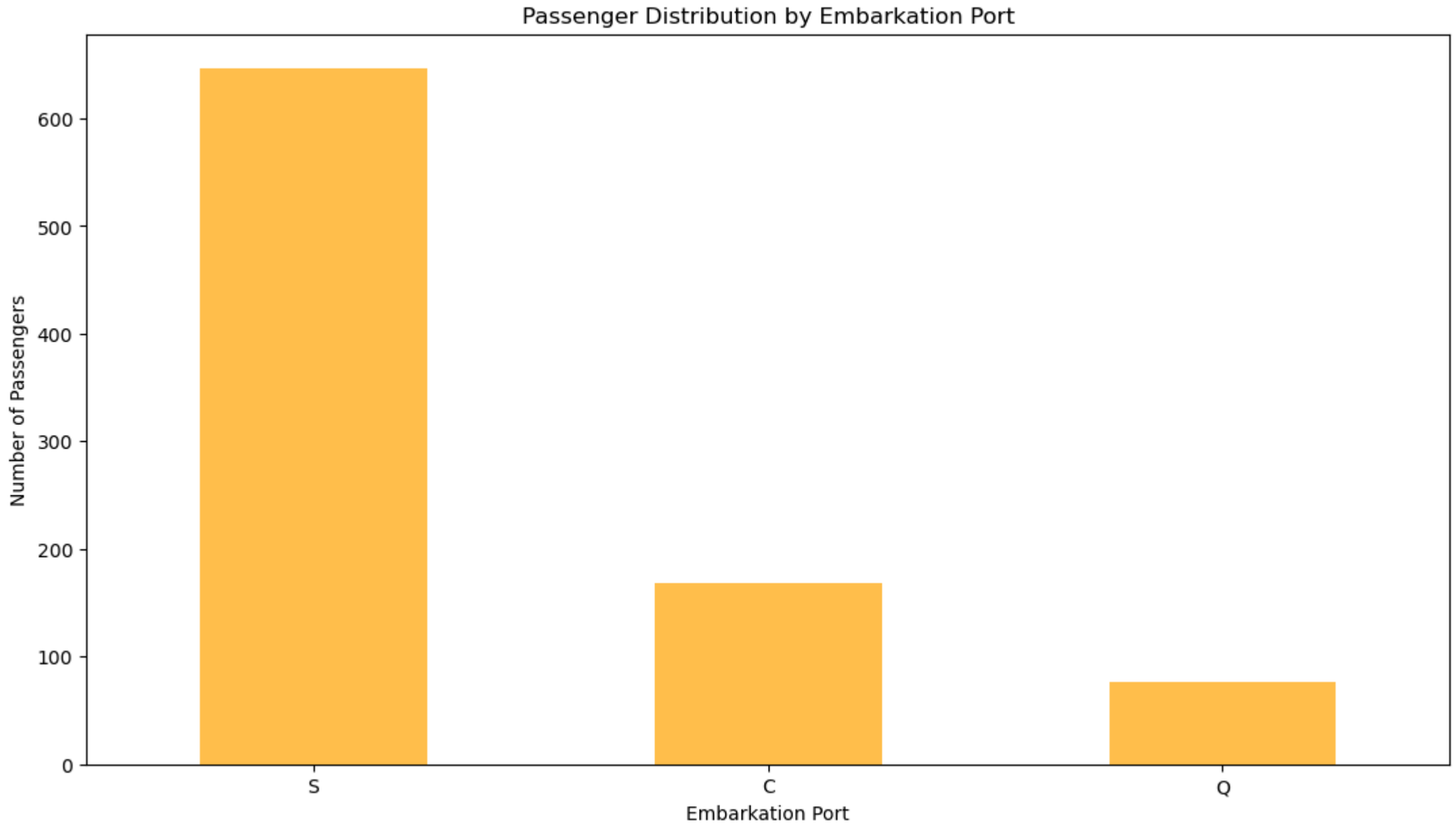




Embarkation Port Analysis

In [167…

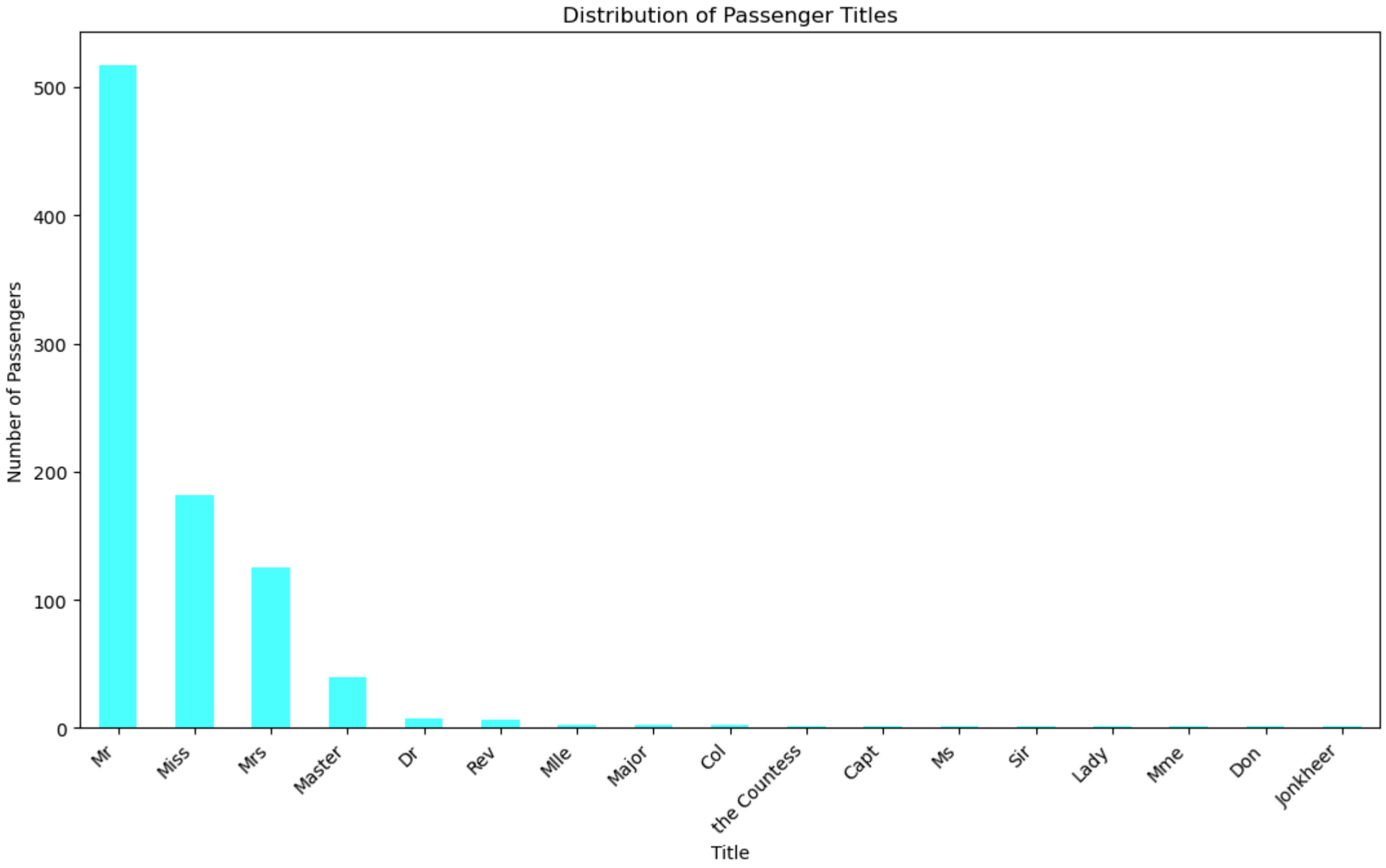
```
# Count passengers embarked at each port
embarkation_counts = titanic_data['Embarked'].value_counts()

# Plot
plt.figure(figsize=(13, 7))
embarkation_counts.plot(kind='bar', color='orange', alpha=0.7)
plt.title('Passenger Distribution by Embarkation Port')
plt.xlabel('Embarkation Port')
plt.ylabel('Number of Passengers')
plt.xticks(rotation=0)
plt.show()
```



Name Analysis

In [168…

```python
# Extract titles from names
titanic_data['Title'] = titanic_data['Name'].apply(lambda name: name.split(',')[1].split('.')[0].strip())

# Count occurrences of each title
title_counts = titanic_data['Title'].value_counts()

# Plot
plt.figure(figsize=(13, 7))
title_counts.plot(kind='bar', color='cyan', alpha=0.7)
plt.title('Distribution of Passenger Titles')
plt.xlabel('Title')
plt.ylabel('Number of Passengers')
plt.xticks(rotation=45, ha='right')
plt.show()
```


Distribution of Passenger Titles
Number of Passengers
0
100
200
300
400
500
Mr
Miss
Mrs
Master
Dr
Rev
Mlle
Major
Col
the Countess
Capt
Ms
Sir
Lady
Mme
Don
Jonkheer
Title