Importing Required Libraries

```
import pandas as pd
import numpy as ny
import matplotlib.pyplot as plt
import seaborn as sns
```

Loading the Dataset

```
train=pd.read_csv("C:/Users/BFL/Downloads/titanic/train.csv")
test=pd.read_csv("C:/Users/BFL/Downloads/titanic/test.csv")
```

Explore Train Dataset

Checking Top 5 Rows of Train Dataset

```
train.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
```

Checking the Bottom 5 Rows of Train Dataset

```
train.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
```

Checking Number of Rows and Columns

```
train.shape
```

```
(891, 12)
```

Checking the Non Null Values adn Data Types

```
train.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
```

Statistical Findings

```
train.describe()
```

```
       PassengerId    Survived      Pclass         Age       SibSp  \
count   891.000000  891.000000  891.000000  714.000000  891.000000
mean    446.000000    0.383838    2.308642   29.699118    0.523008
std     257.353842    0.486592    0.836071   14.526497    1.102743
min       1.000000    0.000000    1.000000    0.420000    0.000000
25%     223.500000    0.000000    2.000000   20.125000    0.000000
50%     446.000000    0.000000    3.000000   28.000000    0.000000
75%     668.500000    1.000000    3.000000   38.000000    1.000000
max     891.000000    1.000000    3.000000   80.000000    8.000000

            Parch        Fare
count  891.000000  891.000000
mean     0.381594   32.204208
std      0.806057   49.693429
min      0.000000    0.000000
25%      0.000000    7.910400
50%      0.000000   14.454200
75%      0.000000   31.000000
max      6.000000  512.329200
```

Checking the Value Counts

```
train.value_counts()
```

```
PassengerId  Survived  Pclass  Name
Sex     Age   SibSp  Parch  Ticket    Fare      Cabin  Embarked
2            1         1       Cumings, Mrs. John Bradley (Florence
Briggs Thayer)   female  38.0  1      0      PC 17599  71.2833   C85
C           1
572          1         1       Appleton, Mrs. Edward Dale (Charlotte
Lamson)         female  53.0  2      0      11769     51.4792   C101
S           1
578          1         1       Silvey, Mrs. William Baird (Alice
Munger)              female  39.0  1      0      13507     55.9000
E44    S           1
582          1         1       Thayer, Mrs. John Borland (Marian
Longstreth Morris)  female  39.0  1      1      17421     110.8833
C68    C           1
584          0         1       Ross, Mr. John Hugo
male    36.0  0      0      13049     40.1250   A10    C           1

..
328          1         2       Ball, Mrs. (Ada E Hall)
```

```
female  36.0  0      0      28551     13.0000   D      S            1
330          1        1         Hippach, Miss. Jean Gertrude
female  16.0  0      1      111361    57.9792   B18    C            1
332          0        1         Partner, Mr. Austen
male    45.5  0      0      113043    28.5000   C124   S            1
333          0        1         Graham, Mr. George Edward
male    38.0  0      1      PC 17582  153.4625  C91    S            1
890          1        1         Behr, Mr. Karl Howell
male    26.0  0      0      111369    30.0000   C148   C            1
Name: count, Length: 183, dtype: int64
```

Columnwise Number of Null Values in Train Dataset

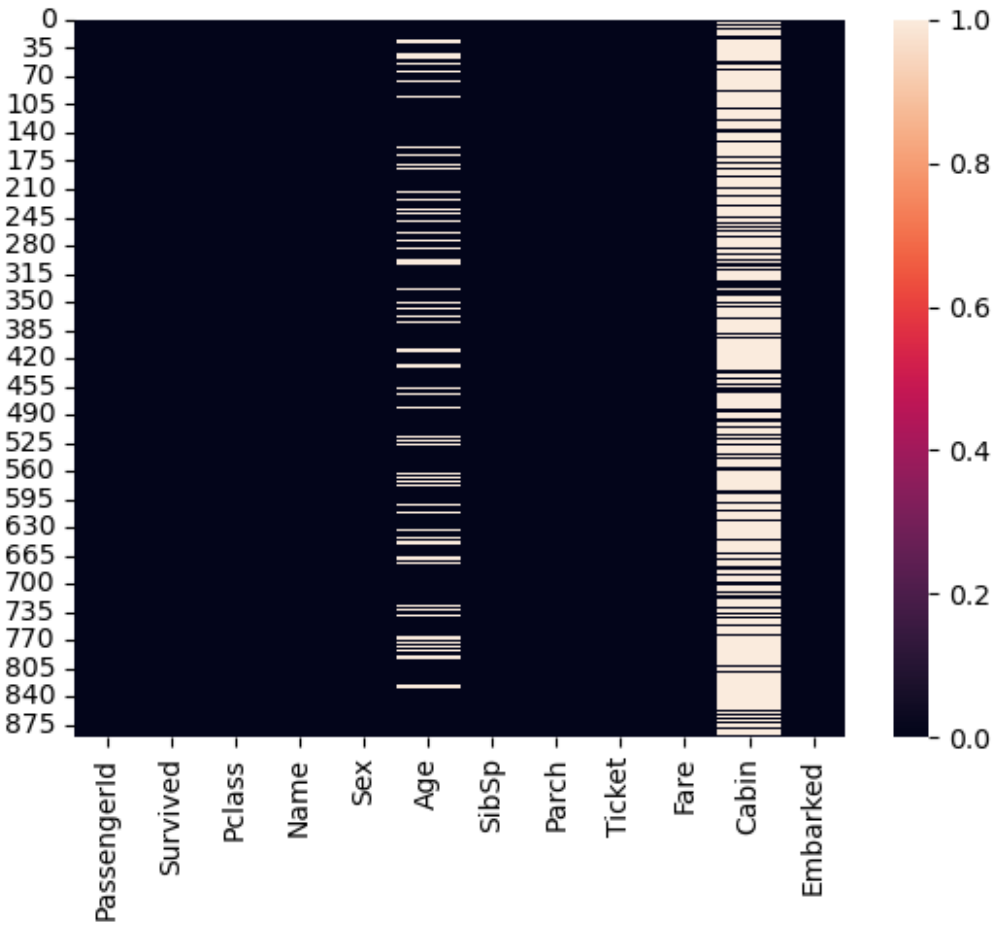
```
train.isnull().sum()
```

```
PassengerId      0
Survived         0
Pclass           0
Name             0
Sex              0
Age            177
SibSp            0
Parch            0
Ticket           0
Fare             0
Cabin          687
Embarked         2
dtype: int64
```

Plotting Heatmap for Null Values

```
sns.heatmap(train.isnull())
```

```
<Axes: >
```



Explore Test Dataset

Checking Top 5 Rows of Test Dataset

```
test.head()
```

```
   PassengerId  Pclass                                          Name
Sex  \
0          892       3                              Kelly, Mr. James
male
1          893       3              Wilkes, Mrs. James (Ellen Needs)
female
2          894       2                     Myles, Mr. Thomas Francis
male
3          895       3                              Wirz, Mr. Albert
male
4          896       3  Hirvonen, Mrs. Alexander (Helga E Lindqvist)
female

    Age   SibSp   Parch   Ticket     Fare Cabin Embarked
```

```
0  34.5      0      0   330911   7.8292   NaN        Q
1  47.0      1      0   363272   7.0000   NaN        S
2  62.0      0      0   240276   9.6875   NaN        Q
3  27.0      0      0   315154   8.6625   NaN        S
4  22.0      1      1  3101298  12.2875   NaN        S
```

Checking Bottom 5 Rows of Test Dataset

```
test.tail()
```

```
     PassengerId  Pclass                          Name     Sex   Age
SibSp  \
413         1305       3            Spector, Mr. Woolf    male   NaN
0
414         1306       1  Oliva y Ocana, Dona. Fermina  female  39.0
0
415         1307       3  Saether, Mr. Simon Sivertsen    male  38.5
0
416         1308       3           Ware, Mr. Frederick    male   NaN
0
417         1309       3      Peter, Master. Michael J    male   NaN
1

     Parch              Ticket      Fare Cabin Embarked
413      0           A.5. 3236    8.0500   NaN        S
414      0            PC 17758  108.9000  C105        C
415      0  SOTON/O.Q. 3101262    7.2500   NaN        S
416      0              359309    8.0500   NaN        S
417      1                2668   22.3583   NaN        C
```

Checking Number of Rows and Columns

```
test.shape
```

```
(418, 11)
```

Checking Data type and Not Null Values

```
test.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 418 entries, 0 to 417
Data columns (total 11 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   PassengerId  418 non-null    int64
 1   Pclass       418 non-null    int64
 2   Name         418 non-null    object
 3   Sex          418 non-null    object
 4   Age          332 non-null    float64
```

```
 5   SibSp        418 non-null    int64
 6   Parch        418 non-null    int64
 7   Ticket       418 non-null    object
 8   Fare         417 non-null    float64
 9   Cabin        91 non-null     object
 10  Embarked     418 non-null    object
dtypes: float64(2), int64(4), object(5)
memory usage: 36.1+ KB
```

Statistical Finding of Test Dataset

```
test.describe()
```

```
       PassengerId      Pclass         Age       SibSp       Parch
Fare
count   418.000000  418.000000  332.000000  418.000000  418.000000
417.000000
mean   1100.500000    2.265550   30.272590    0.447368    0.392344
35.627188
std     120.810458    0.841838   14.181209    0.896760    0.981429
55.907576
min     892.000000    1.000000    0.170000    0.000000    0.000000
0.000000
25%     996.250000    1.000000   21.000000    0.000000    0.000000
7.895800
50%    1100.500000    3.000000   27.000000    0.000000    0.000000
14.454200
75%    1204.750000    3.000000   39.000000    1.000000    0.000000
31.500000
max    1309.000000    3.000000   76.000000    8.000000    9.000000
512.329200
```

Value Counts of Test Dataset

```
test.value_counts()
```

```
PassengerId  Pclass  Name
Sex     Age   SibSp  Parch  Ticket    Fare      Cabin    Embarked
904          1       Snyder, Mrs. John Pillsbury (Nelle Stevenson)
female  23.0  1      0      21228     82.2667   B45      S           1
1164         1       Clark, Mrs. Walter Miller (Virginia McDowell)
female  26.0  1      0      13508     136.7792  C89      C           1
1213         3       Krekorian, Mr. Neshan
male    25.0  0      0      2654      7.2292    F E57    C           1
1208         1       Spencer, Mr. William Augustus
male    57.0  1      0      PC 17569  146.5208  B78      C           1
1206         1       White, Mrs. John Stuart (Ella Holmes)
female  55.0  0      0      PC 17760  135.6333  C32      C           1

..
```

```
1009          3       Sandstrom, Miss. Beatrice Irene
female  1.0   1       1       PP 9549    16.7000   G6       S            1
1006          1       Straus, Mrs. Isidor (Rosalie Ida Blun)
female  63.0  1       0       PC 17483   221.7792  C55 C57  S            1
1004          1       Evans, Miss. Edith Corse
female  36.0  0       0       PC 17531   31.6792   A29      C            1
1001          2       Swane, Mr. George
male    18.5  0       0       248734     13.0000   F        S            1
1306          1       Oliva y Ocana, Dona. Fermina
female  39.0  0       0       PC 17758   108.9000  C105     C            1
Name: count, Length: 87, dtype: int64
```

Columnwise Number of Null Values in Test Dataset

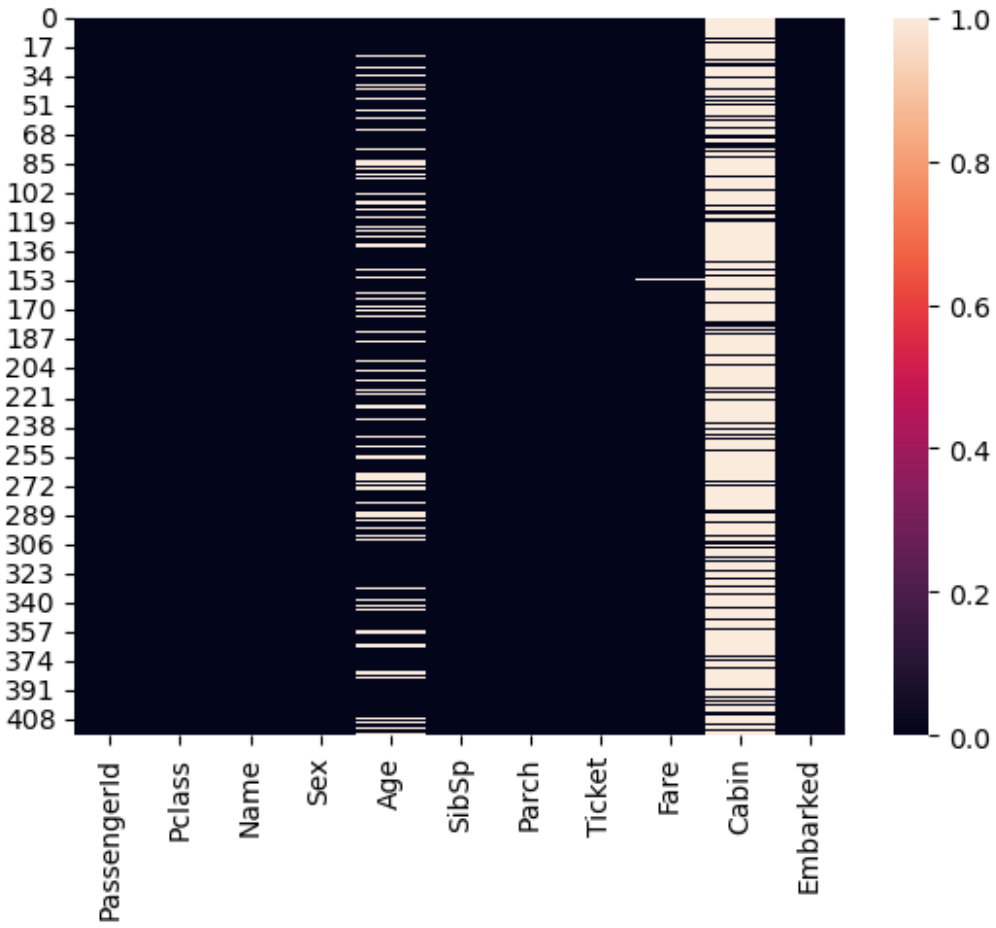
```
test.isnull().sum()
```

```
PassengerId      0
Pclass           0
Name             0
Sex              0
Age             86
SibSp            0
Parch            0
Ticket           0
Fare             1
Cabin          327
Embarked         0
dtype: int64
```

Heatmap for Null Values

```
sns.heatmap(test.isnull())
```

```
<Axes: >
```



Cleaning Train Dataset

Deleting the Unnecessary Columns

```
train.drop(['Name','Ticket','Cabin','Embarked'],axis=1,inplace=True)
```

Filling Median Age Values in Null Values

```
train.Age=train.Age.fillna(train.Age.median())
```

Deleting the Null Values in all Columns

```
train.dropna()
```

```
     PassengerId  Survived  Pclass     Sex   Age  SibSp  Parch
Fare
0              1         0       3    male  22.0      1      0
7.2500
1              2         1       1  female  38.0      1      0
71.2833
```

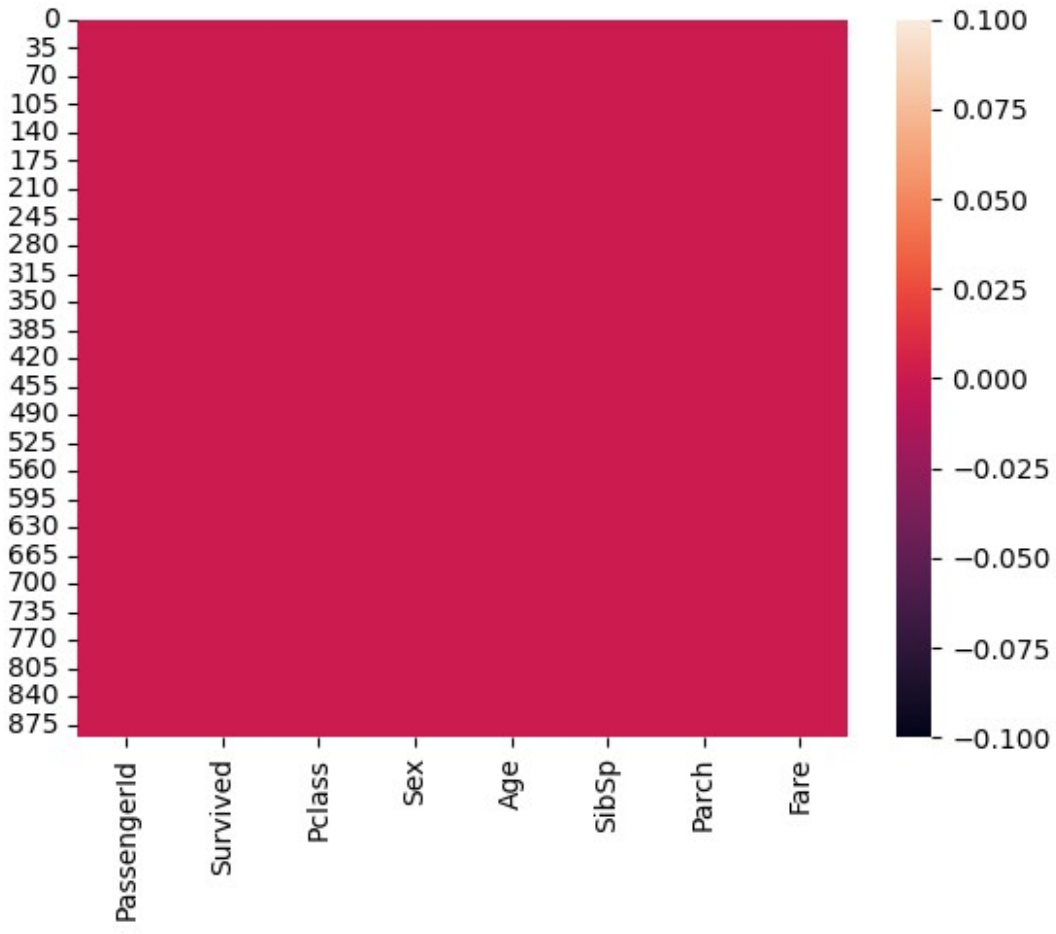
```
2              3         1       3  female  26.0      0      0
7.9250
3              4         1       1  female  35.0      1      0
53.1000
4              5         0       3    male  35.0      0      0
8.0500
..           ...       ...     ...     ...   ...    ...    ...       ..
.
886          887         0       2    male  27.0      0      0
13.0000
887          888         1       1  female  19.0      0      0
30.0000
888          889         0       3  female  28.0      1      2
23.4500
889          890         1       1    male  26.0      0      0
30.0000
890          891         0       3    male  32.0      0      0
7.7500

[891 rows x 8 columns]
```

Checking with Heatmap if all Null Values are deleted

```
sns.heatmap(train.isnull())
```

```
<Axes: >
```



Cleaning Test Dataset

Deleting Unnecessary Columns

```
test.drop(['Name','Ticket','Cabin','Embarked'],axis=1,inplace=True)
```

Filling Median Age Values in Null Values

```
test.Age=test.Age.fillna(test.Age.median())
```

Deleting the Null Values in all Columns

```
test.dropna()
```

	PassengerId	Pclass	Sex	Age	SibSp	Parch	Fare
0	892	3	male	34.5	0	0	7.8292
1	893	3	female	47.0	1	0	7.0000
2	894	2	male	62.0	0	0	9.6875
3	895	3	male	27.0	0	0	8.6625
4	896	3	female	22.0	1	1	12.2875

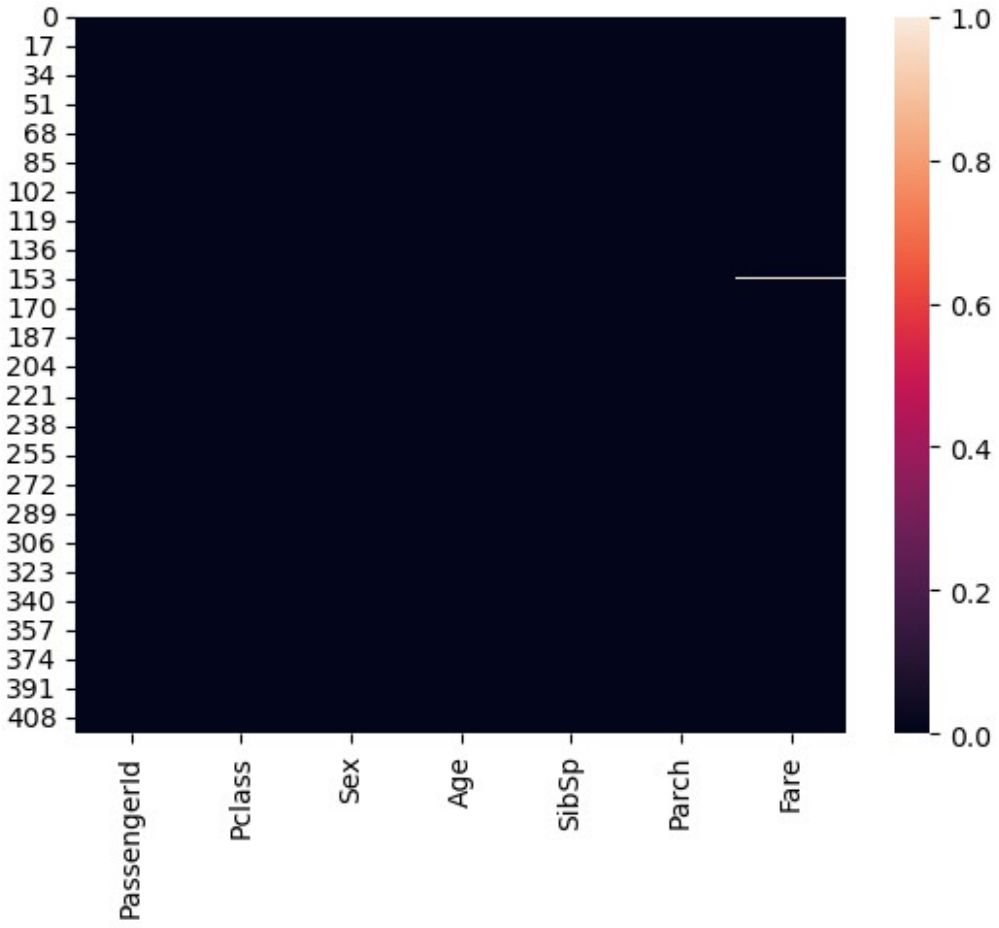
```
..           ...     ...     ...   ...    ...    ...       ...
413         1305       3    male  27.0      0      0    8.0500
414         1306       1  female  39.0      0      0  108.9000
415         1307       3    male  38.5      0      0    7.2500
416         1308       3    male  27.0      0      0    8.0500
417         1309       3    male  27.0      1      1   22.3583

[417 rows x 7 columns]
```

Checking with Heatmap if all Null Values are deleted

```
sns.heatmap(test.isnull())
```

```
<Axes: >
```



Changing Sex Column Data Type to int

```
train.Sex = train.Sex.map({'female': 0, 'male': 1})
```

Checking Data Types for Train Dataset

```
train.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 8 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   Sex          891 non-null    int64  
 4   Age          891 non-null    float64
 5   SibSp        891 non-null    int64  
 6   Parch        891 non-null    int64  
 7   Fare         891 non-null    float64
dtypes: float64(2), int64(6)
memory usage: 55.8 KB
```

Train Data Analysis

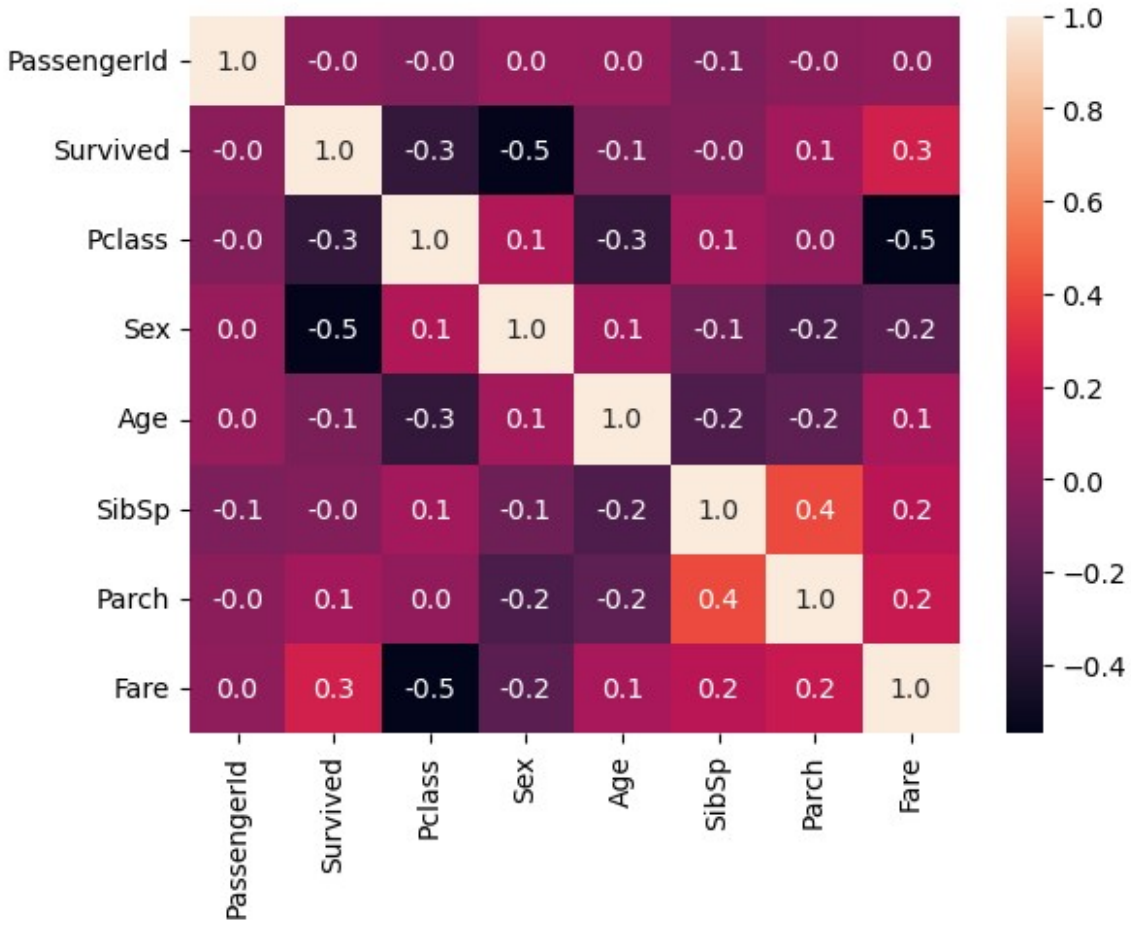
Checking linear relationships between numeric variables

```
corr=train.corr()
```

Visualising Relationship with Heatmap

```
sns.heatmap(corr,annot=True,fmt='.1f')
```

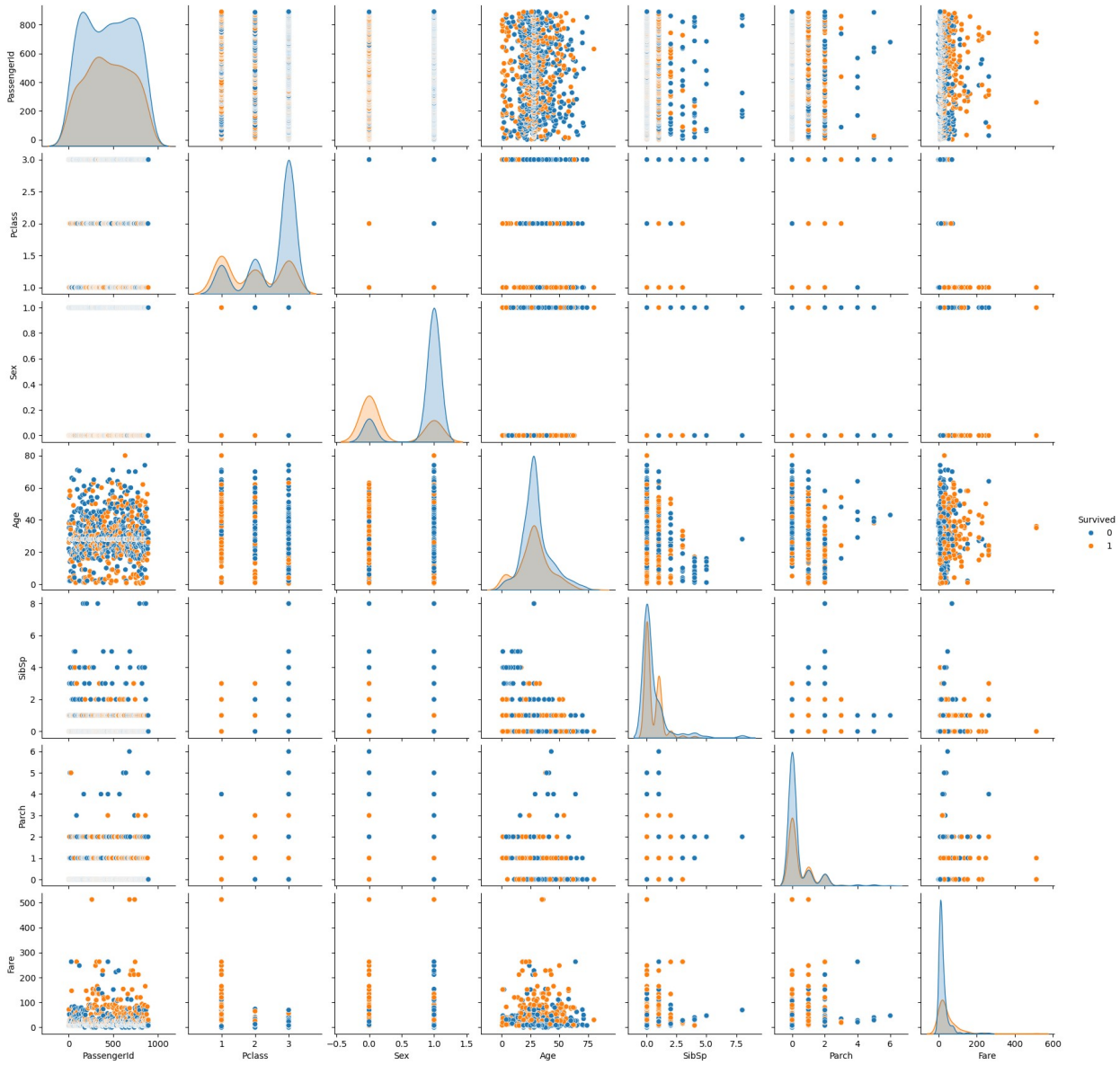
```
<Axes: >
```



Pairplot on Basis of Survival

```
sns.pairplot(train, hue='Survived')
```

```
<seaborn.axisgrid.PairGrid at 0x1ca04d098e0>
```

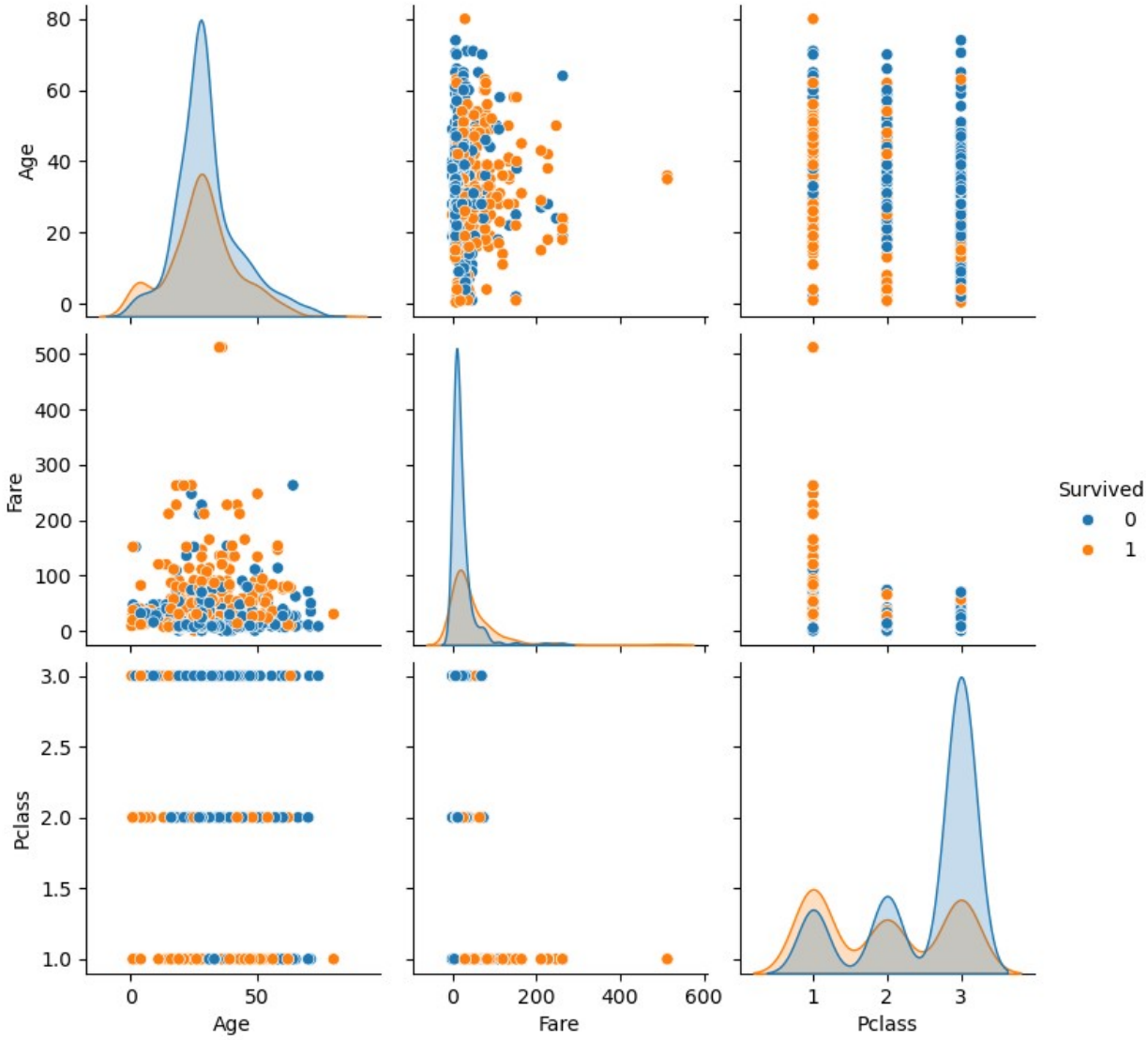



```
## Key Findings
# Younger passengers (children) had a higher survival rate.
# The higher the class (Pclass = 1), the higher the survival rate.
Most 1st class passengers survived, while 3rd class passengers had
much lower survival rates.
# A higher fare correlates with a higher chance of survival, which is
likely related to class.
# Having a family aboard (SibSp or Parch) doesn't directly show a
significant trend in survival, but it could affect the priority for
lifeboats.
```

Analysis of Passengers Survived on basis of Age,Fare and Pclass

```
sns.pairplot(train, vars=['Age', 'Fare', 'Pclass'], hue='Survived')
```

```
<seaborn.axisgrid.PairGrid at 0x1ca087396d0>
```




```
## Key Finding
# Younger passengers had a higher survival rate.
# The higher the class the higher the survival rate.
# The higher the Fare the higher the survival rate.
```

Number of Passengers Survived Vs Not Survived

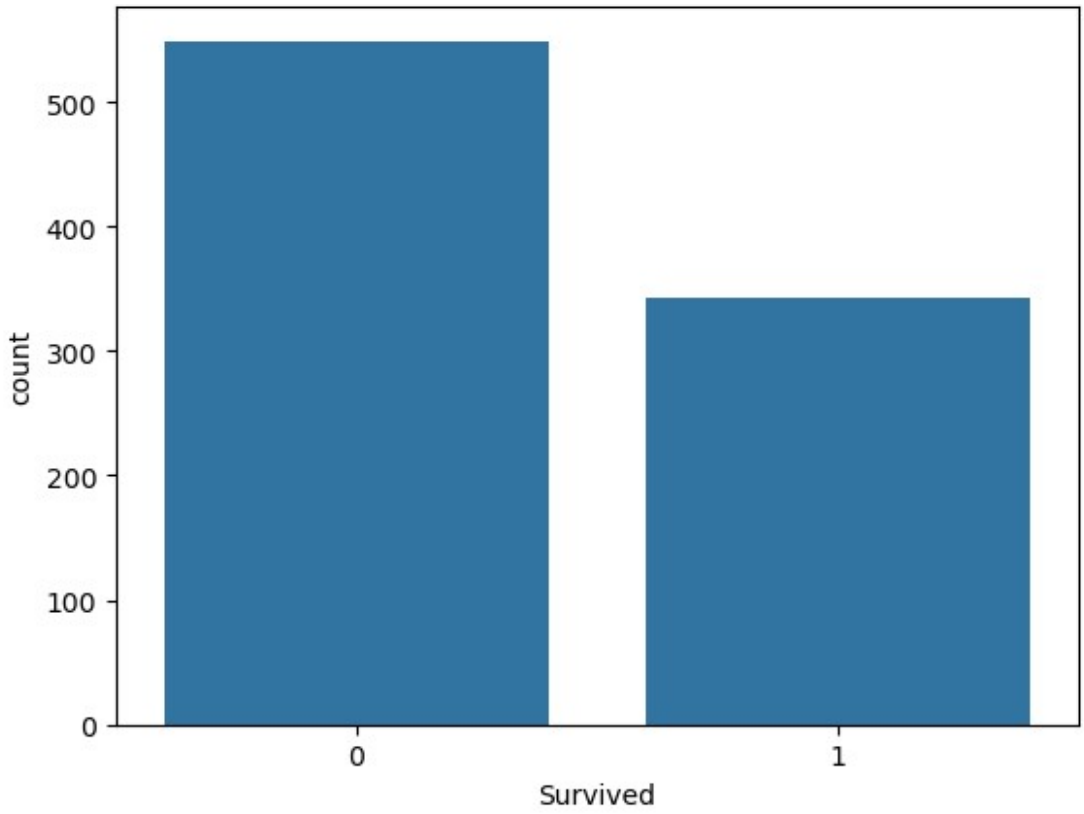
```
train.Survived.value_counts()
```

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

Countplot Visualisation of Number of Passengers Survived Vs Not Survived

```
sns.countplot(x='Survived', data=train)
```

```
<Axes: xlabel='Survived', ylabel='count'>
```




```
## Key Finding
# There were a much Higher number of non-surviving passengers than
survivors.
```

Number of Male Vs Female Passengers

```
train.Sex.value_counts()
```

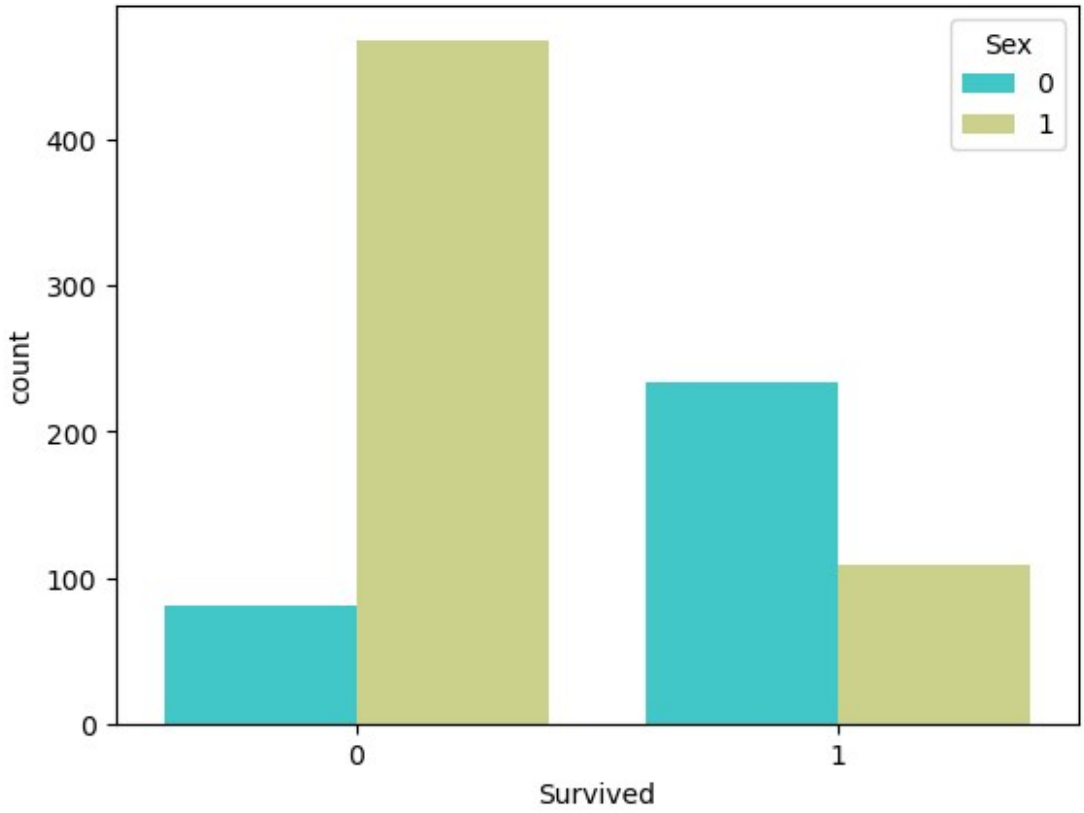
```
Sex
1    577
```

```
0    314
Name: count, dtype: int64
```

Countplot Visualisation Number of Male Vs Female Passengers Survived

```
sns.countplot(x='Survived',hue='Sex',data=train, palette='rainbow')
```

```
<Axes: xlabel='Survived', ylabel='count'>
```

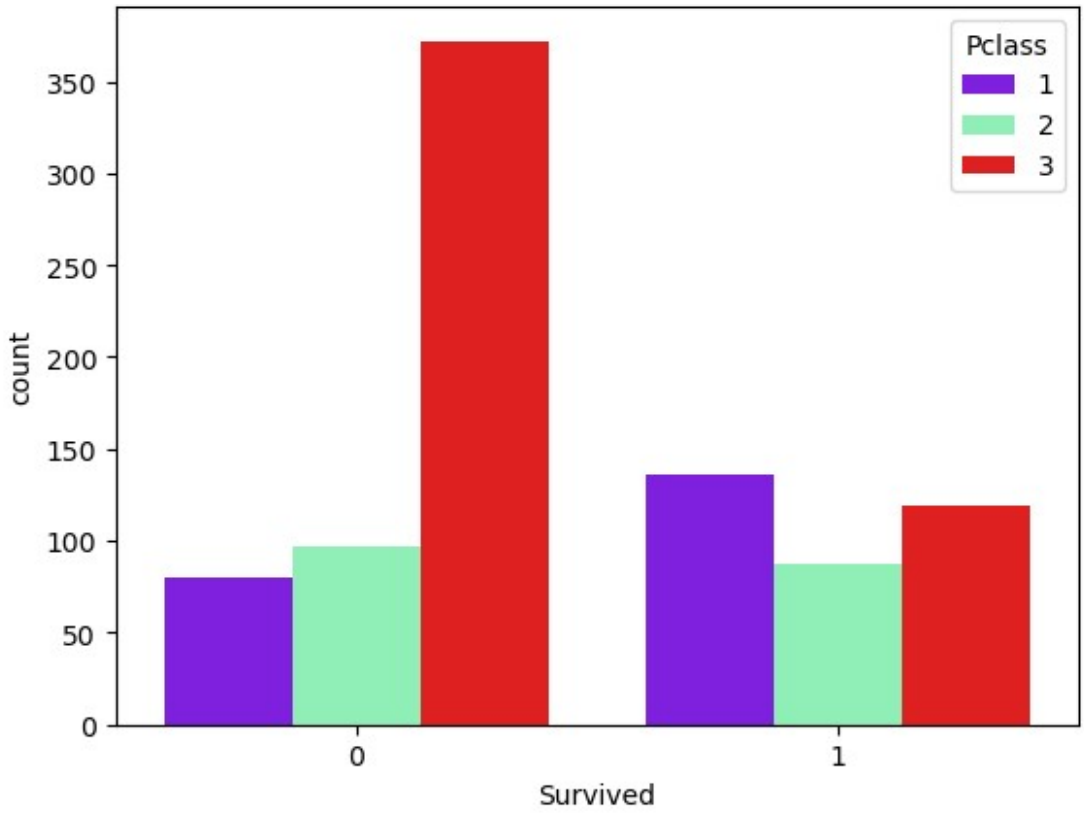



```
## Key Findings
# Gender Played a Significant role in survival, females have higher
survival rate than males.
```

Countplot Visualisation of Number of Passenger Survived on Basis of Pclass

```
sns.countplot(x='Survived', hue='Pclass',data=train,
palette='rainbow')
```

```
<Axes: xlabel='Survived', ylabel='count'>
```

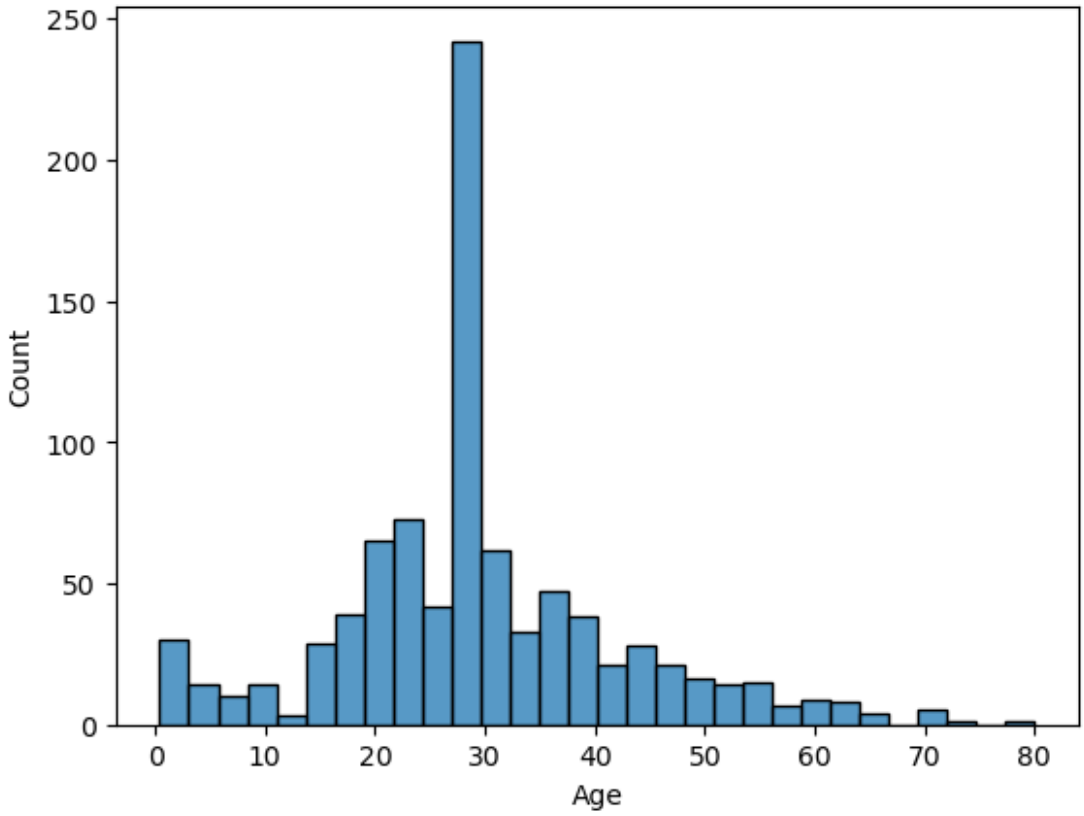



```
## Key Finding
#First-class passengers (Pclass 1) had a much higher survival rate.
#Second-class passengers (Pclass 2) had a moderate survival rate.
#Third-class passengers (Pclass 3) had a significantly lower survival
rate.
```

Histogram Visualisation of Passengers Age

```
sns.histplot(x='Age',data=train)
```

```
<Axes: xlabel='Age', ylabel='Count'>
```

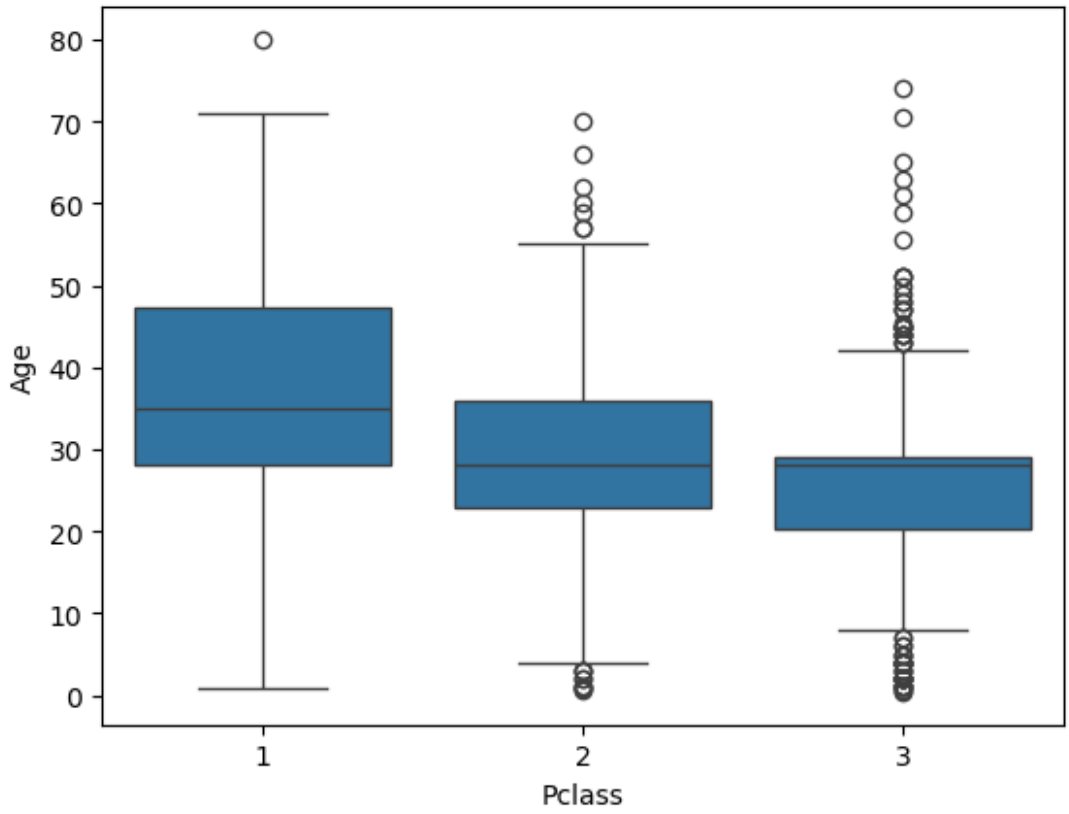



```
## Key Findings
#The distribution is slightly skewed towards younger passengers, with
fewer elderly passengers.
```

Boxplot Visualisation of Pclass Passengers on basis of Age

```
sns.boxplot(x='Pclass',y='Age',data=train)
```

```
<Axes: xlabel='Pclass', ylabel='Age'>
```

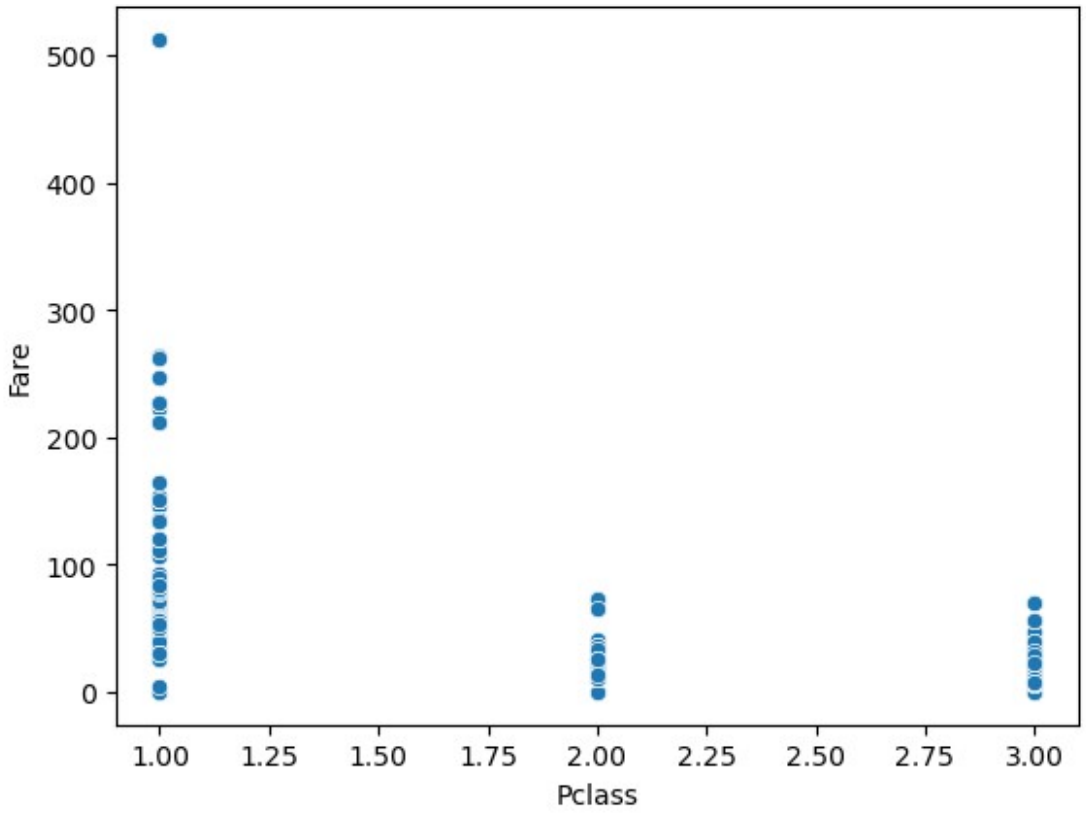



```
## Key Findings
# The age of passengers varies considerably across classes, and this
could be useful in understanding survival rates, as younger passengers
had a higher survival rate,while older passengers, especially in third
class, had lower survival rates.
```

Scatterplot of Pclass Passenger on basis of Fare

```
sns.scatterplot(x='Pclass', y='Fare', data=train)
```

```
<Axes: xlabel='Pclass', ylabel='Fare'>
```

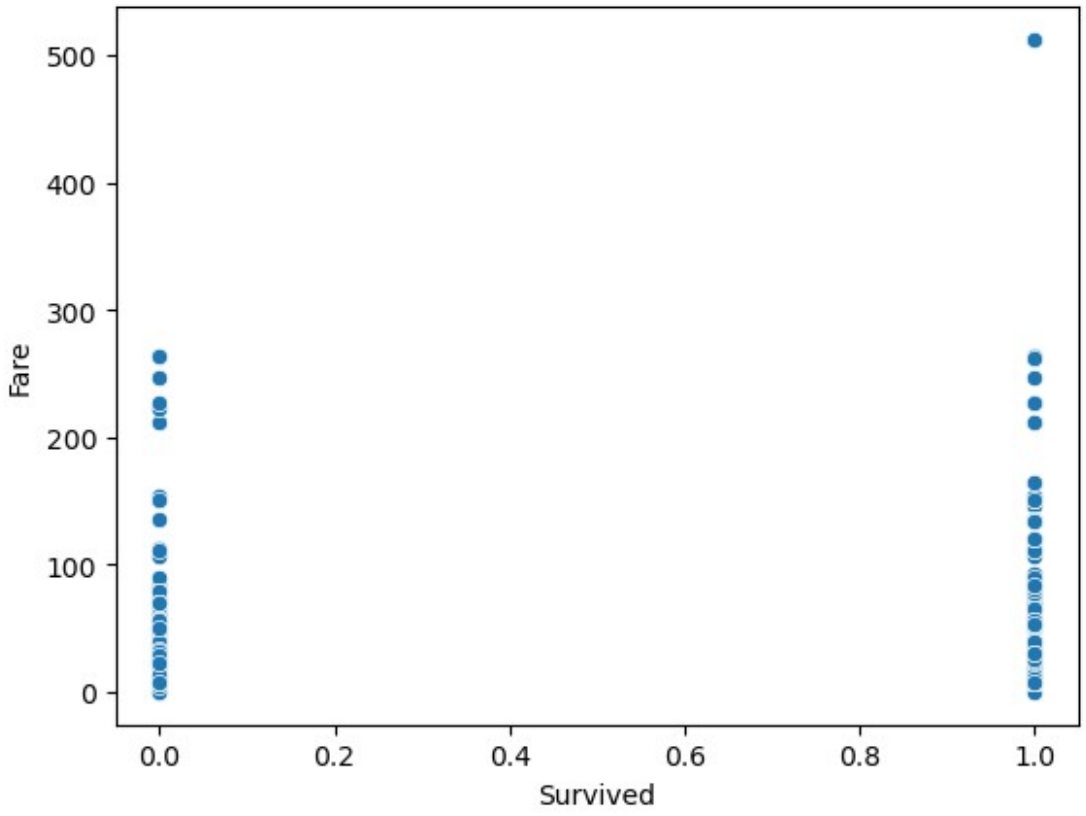



```
##Key Findings
#The scatterplot clearly shows that fare is strongly linked to class
(Pclass). Passengers in first class generally paid significantly
higher fares than those in second and third class.
```

Scatterplot of Passenger Survived on basis of Fare

```
sns.scatterplot(x='Survived', y='Fare', data=train)
```

```
<Axes: xlabel='Survived', ylabel='Fare'>
```




```
## Key Findings
# The scatterplot will likely show that passengers who paid higher
fares were more likely to survive. First-class passengers, who paid
the highest fares, had a higher chance of survival.
```

Changing Sex Column Data Type to int of Test Dataset

```
test.Sex= test.Sex.map({'female': 0, 'male': 1})
```

Checking the Data Type of Test Data

```
test.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 418 entries, 0 to 417
Data columns (total 7 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
```

```
 0   PassengerId  418 non-null    int64
 1   Pclass       418 non-null    int64
 2   Sex          418 non-null    int64
 3   Age          418 non-null    float64
 4   SibSp        418 non-null    int64
 5   Parch        418 non-null    int64
 6   Fare         417 non-null    float64
dtypes: float64(2), int64(5)
memory usage: 23.0 KB
```

Test Data Analysis

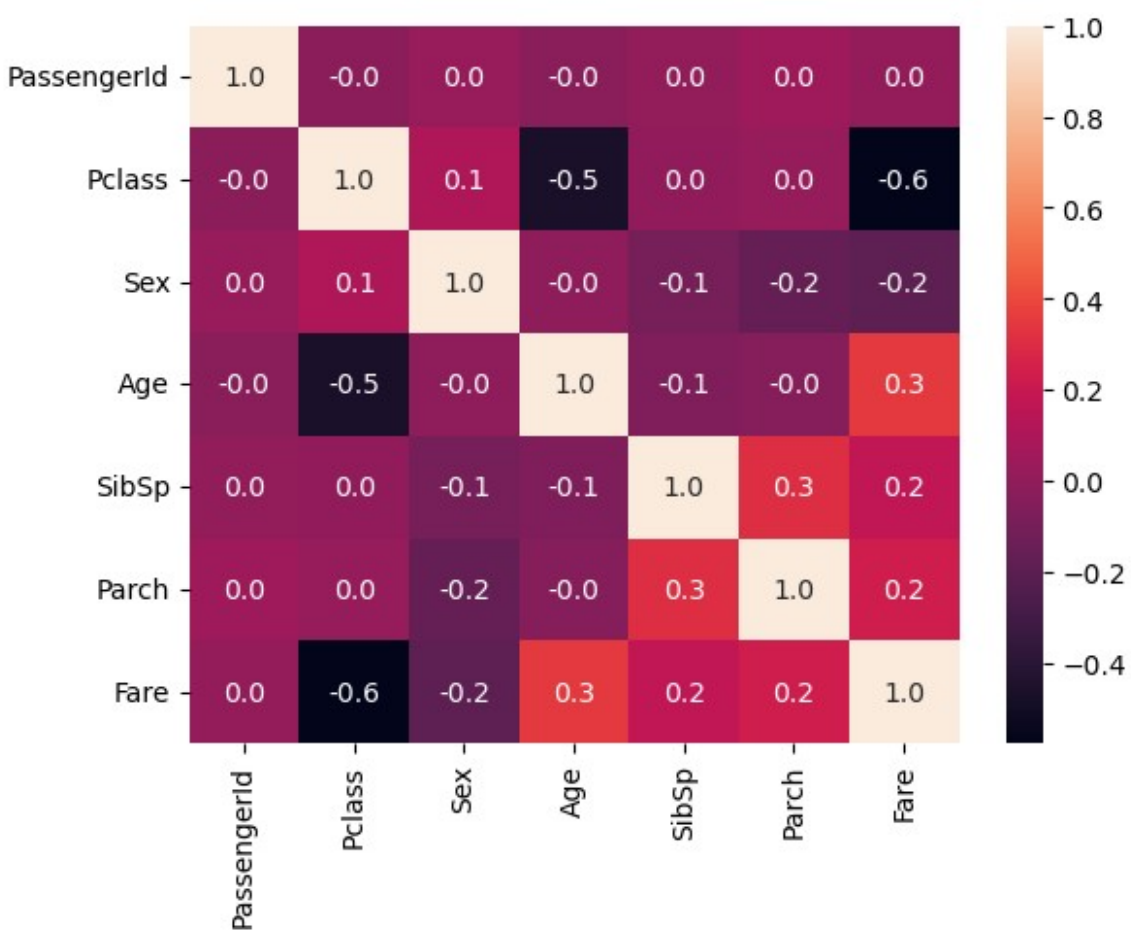
Checking linear relationships between numeric variables

```
Co=test.corr()
```

Visualisation Relationship between Numeric Variables

```
sns.heatmap(Co,annot=True,fmt='.1f')
```

```
<Axes: >
```



Checking Number of Males and Females

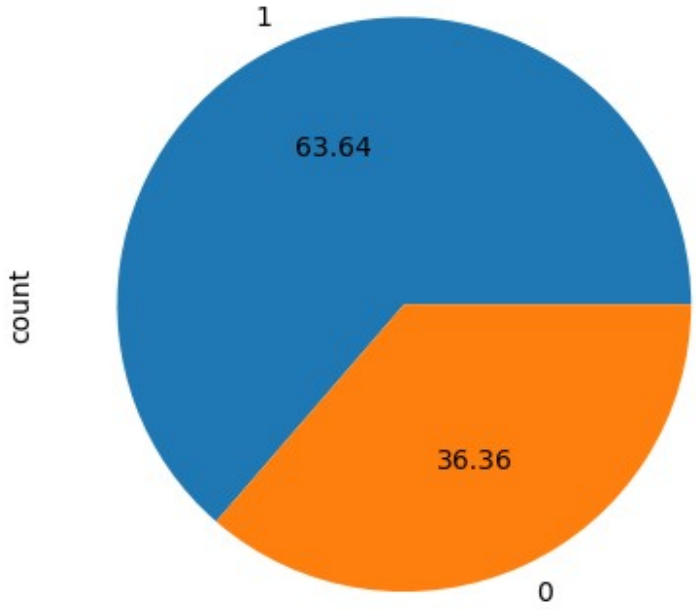
```
test.Sex.value_counts()
```

```
Sex
1    266
0    152
Name: count, dtype: int64
```

Visualisation of Males Female Ratio with Pie Chart

```
test.Sex.value_counts().plot.pie(autopct='%0.2f')
```

```
<Axes: ylabel='count'>
```

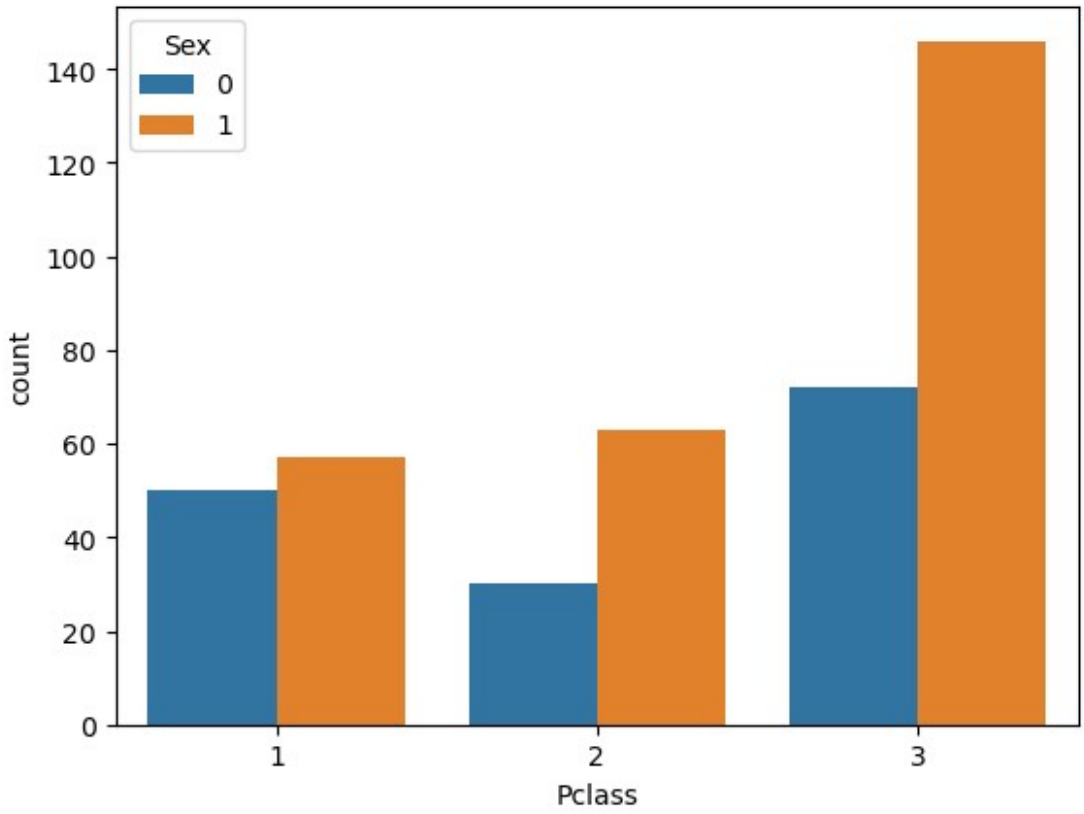



```
##Key Findings
# Represents Male Passenger are 64% While Female Passenger are 36%.
```

Countplot Visualisation of Pclass on basis of Sex

```
sns.countplot(x=test.Pclass, hue=test.Sex)
```

```
<Axes: xlabel='Pclass', ylabel='count'>
```

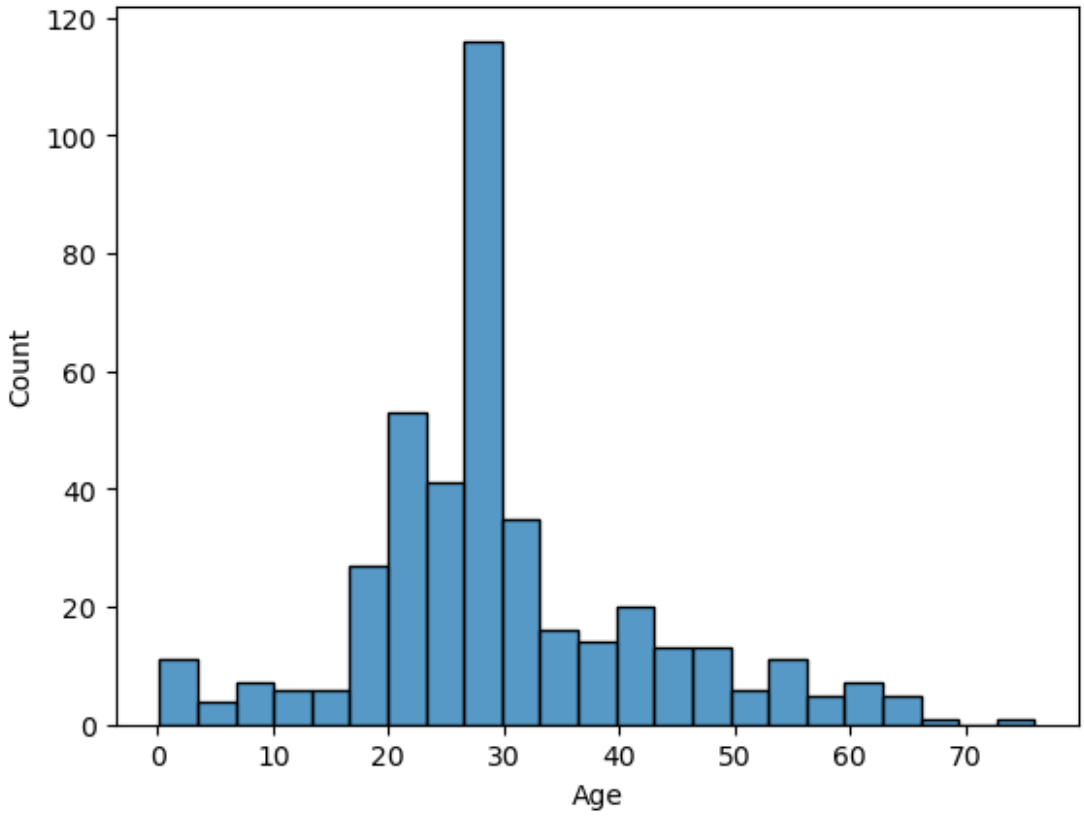



```
##Key Findings
#Third class is dominated by males.
#First class has a relatively higher proportion of females compared to
third class.
#Second class has a moderate distribution of males and females.
```

Histogram Visualisation basis on Age

```
sns.histplot(test.Age)
```

```
<Axes: xlabel='Age', ylabel='Count'>
```

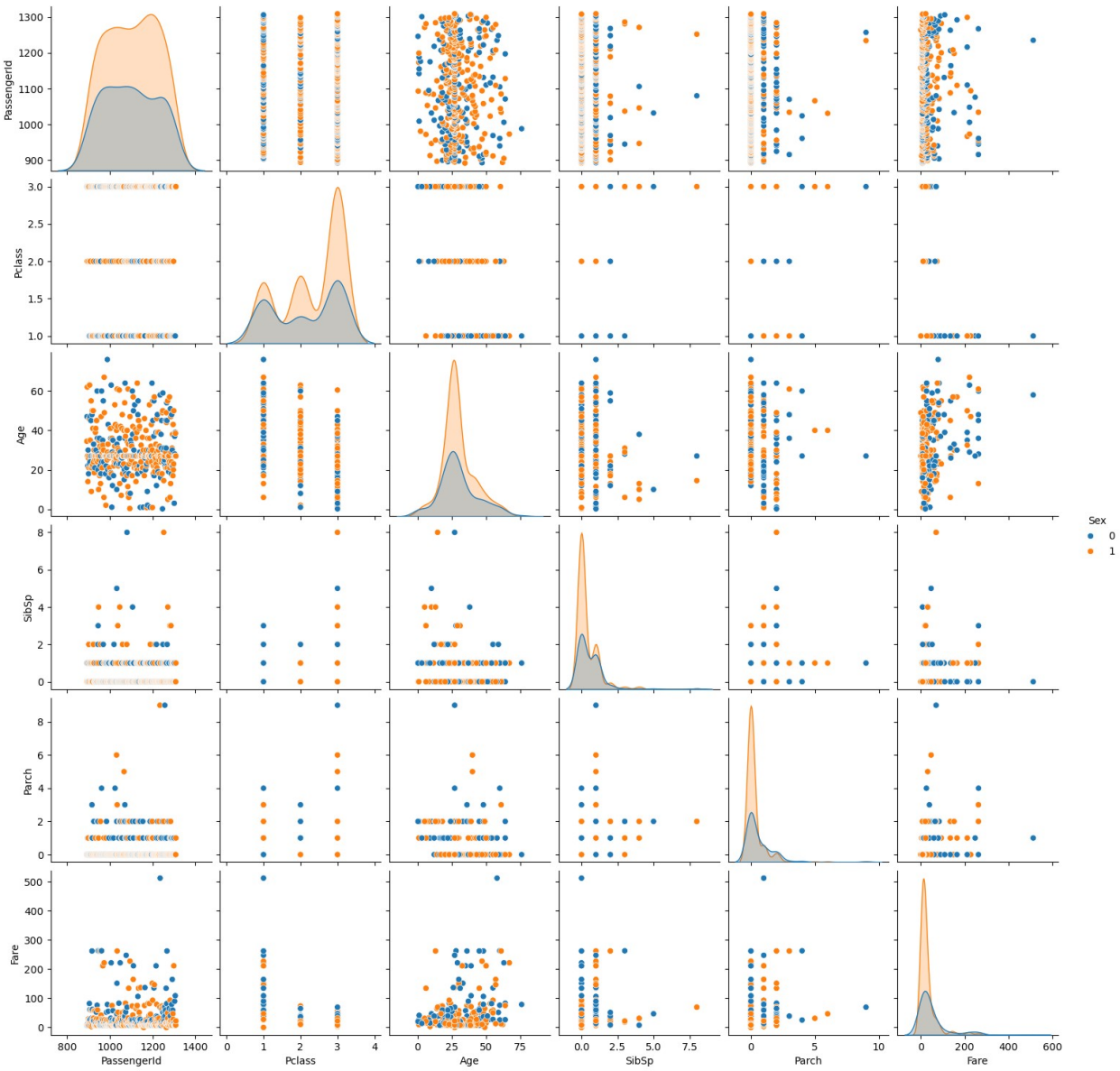



```
## Key Findings
#The distribution is slightly skewed towards younger passengers, with
fewer elderly passengers.
```

Pairplot on Basis of Sex

```
sns.pairplot(test, hue='Sex')
```

```
<seaborn.axisgrid.PairGrid at 0x1ca0b4ef560>
```




```
# Key Findings
# Sex has a strong influence on other variables like Pclass and Fare.
# Females are often associated with higher classes (Pclass 1) and
higher fares, possibly indicating more wealthy or family-linked
travelers.
# Males dominate third class, which had lower fares and a wider age
spread.
# Gender differences are clear and important predictors for survival
chances in the Titanic dataset.
```