In [2]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import os
import warnings
warnings.filterwarnings('ignore')
```

In [3]:

```
titanic=pd.read_csv("C:/Users/sweta/Downloads/Titanic.csv")
```

In [4]:

```
titanic.head(7)
```

Out[4]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500
5	6	0	3	Moran, Mr. James	male	NaN	0	0	330877	8.4583
6	7	0	1	McCarthy, Mr. Timothy J	male	54.0	0	0	17463	51.8625

In [5]:

```
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
```

In [6]:

```
titanic.shape
```

Out[6]:

```
(891, 12)
```

In [7]:

```
titanic.describe()
```

Out[7]:

	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

Exploratory Data Analysis

In [8]:

```
titanic.isnull().sum()
```

Out[8]:

```
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
```

In [9]:

```
titanic.dropna(subset='Embarked',inplace=True)
```

In [10]:

```
titanic.isnull().sum()
```

Out[10]:

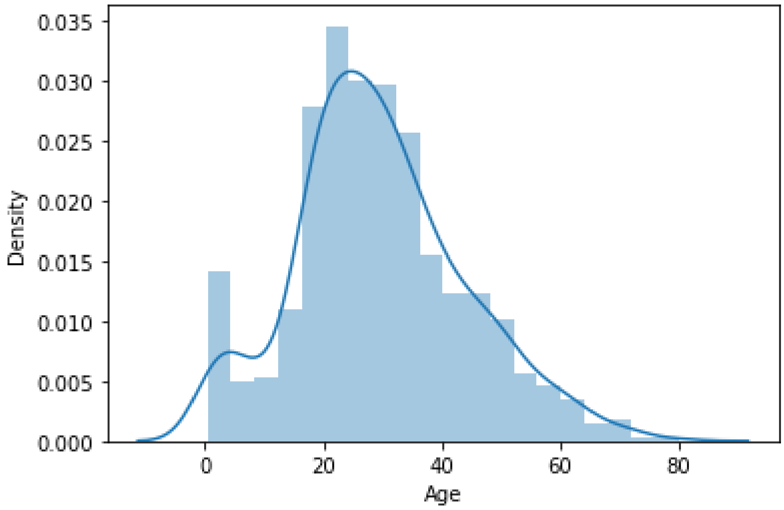
```
PassengerId      0
Survived         0
Pclass           0
Name             0
Sex              0
Age            177
SibSp            0
Parch            0
Ticket           0
Fare             0
Cabin          687
Embarked         0
dtype: int64
```

In [11]:

```
sns.distplot(titanic.Age)
```

Out[11]:

```
<AxesSubplot:xlabel='Age', ylabel='Density'>
```



In [12]:

```
titanic.Age.mean()
```

Out[12]:

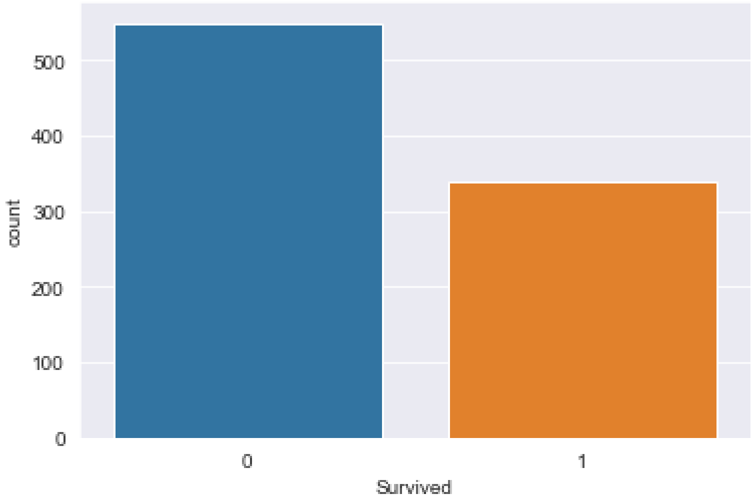
```
29.64209269662921
```

In [13]:

```
sns.set_style('darkgrid')
sns.countplot(x=titanic.Survived)
```

Out[13]:

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

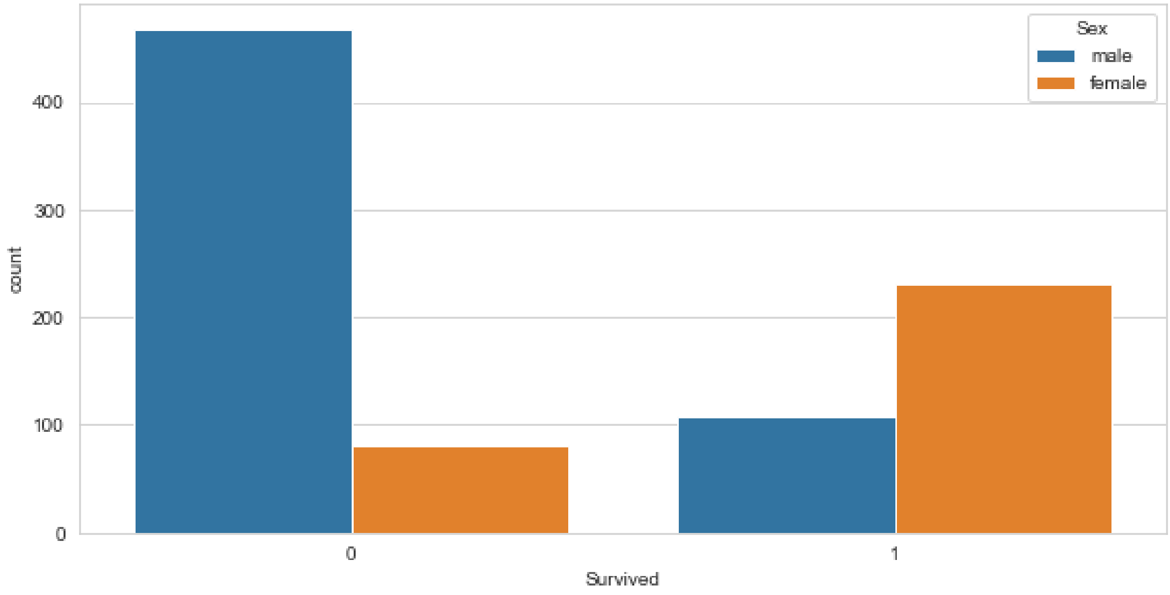


In [14]:

```
plt.figure(figsize=(10,5))
sns.set_style('whitegrid')
sns.countplot(x='Survived',hue='Sex',data=titanic)
```

Out[14]:

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

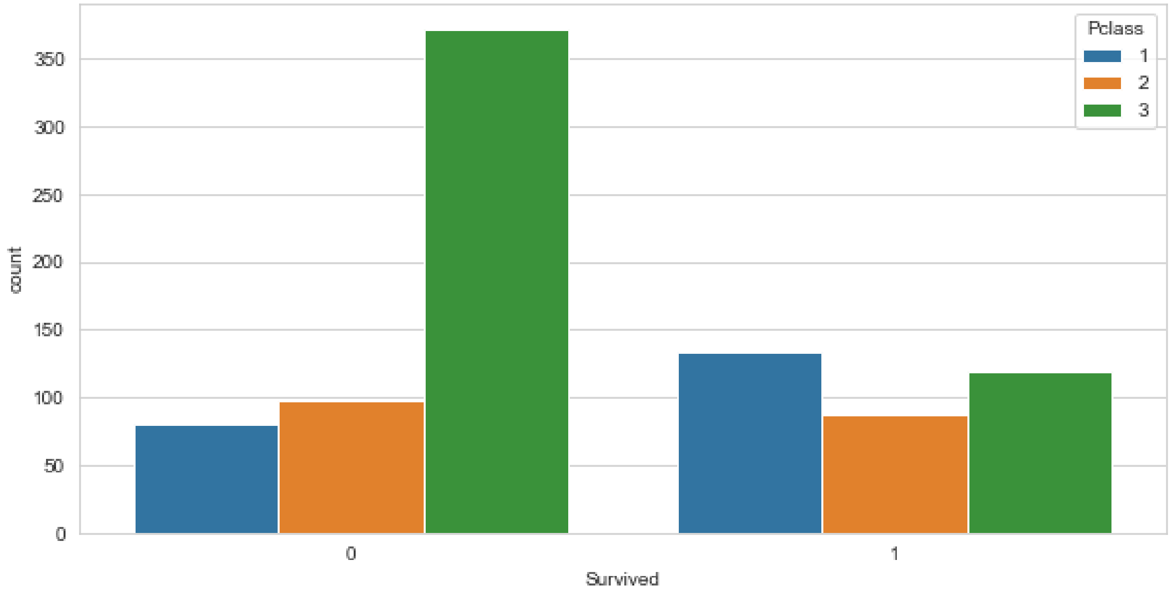


In [15]:

```
plt.figure(figsize=(10,5))
sns.set_style('whitegrid')
sns.countplot(x='Survived',hue='Pclass',data=titanic)
```

Out[15]:

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



In [16]:

```
titanic.Survived.value_counts()
```

Out[16]:

```
0    549
1    340
Name: Survived, dtype: int64
```

In [17]:

```
titanic.columns
```

Out[17]:

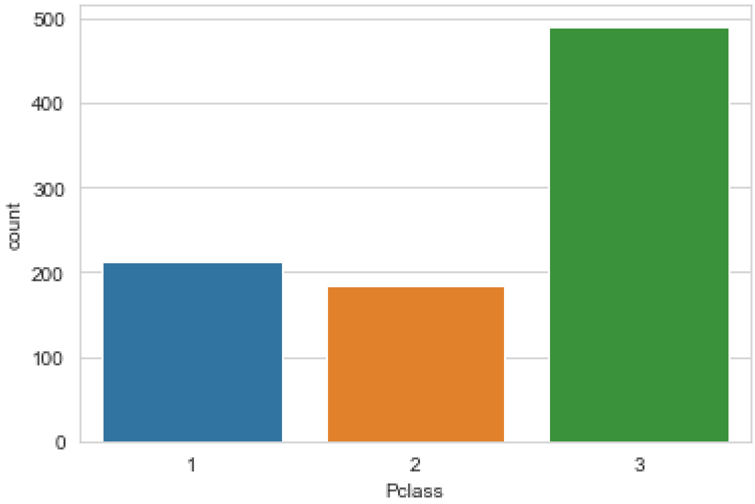
```
Index(['PassengerId', 'Survived', 'Pclass', 'Name', 'Sex', 'Age', 'SibSp',
       'Parch', 'Ticket', 'Fare', 'Cabin', 'Embarked'],
      dtype='object')
```

In [18]:

```
sns.countplot(x=titanic.Pclass)
```

Out[18]:

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

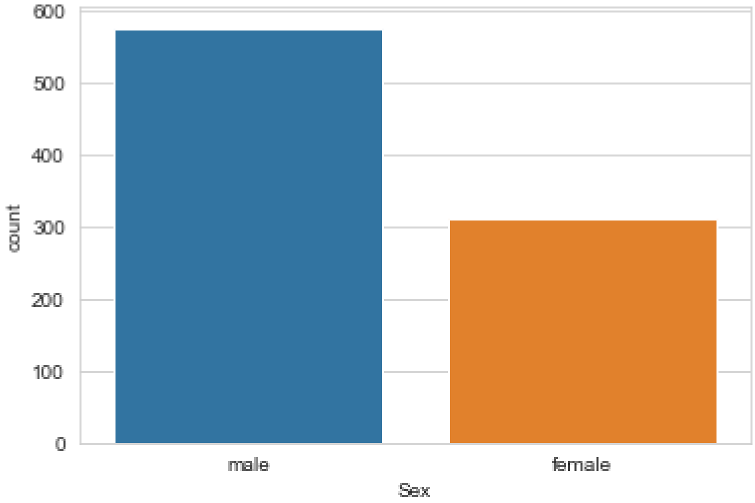


In [19]:

```
sns.countplot(x=titanic.Sex)
```

Out[19]:

```
<AxesSubplot:xlabel='Sex', ylabel='count'>
```

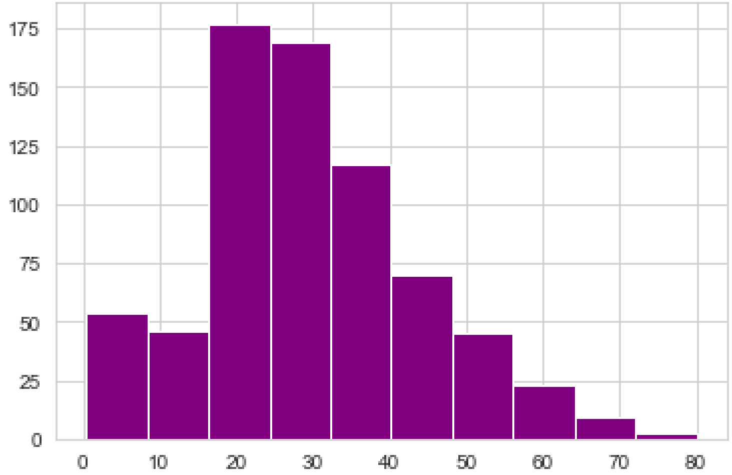


In [20]:

```
plt.hist(titanic.Age,color="purple")
```

Out[20]:

```
(array([ 54.,  46., 177., 169., 117.,  70.,  45.,  23.,   9.,   2.]),
 array([ 0.42 ,  8.378, 16.336, 24.294, 32.252, 40.21 , 48.168, 56.126,
        64.084, 72.042, 80.   ]),
 <BarContainer object of 10 artists>)
```

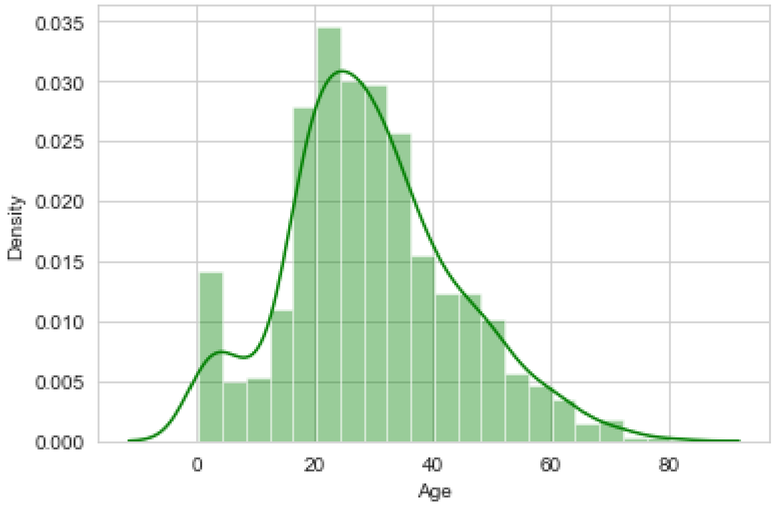


In [21]:

```
sns.distplot(titanic.Age,color="green")
```

Out[21]:

```
<AxesSubplot:xlabel='Age', ylabel='Density'>
```



In [22]:

```
titanic.SibSp.value_counts()
```

Out[22]:

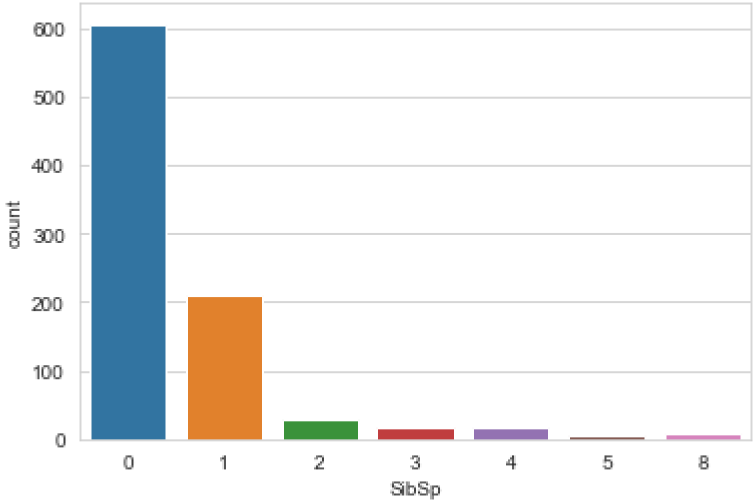
```
0    606
1    209
2     28
4     18
3     16
8      7
5      5
Name: SibSp, dtype: int64
```

In [23]:

```
sns.countplot(x=titanic.SibSp)
```

Out[23]:

```
<AxesSubplot:xlabel='SibSp', ylabel='count'>
```



In [24]:

```
titanic.Parch.value_counts()
```

Out[24]:

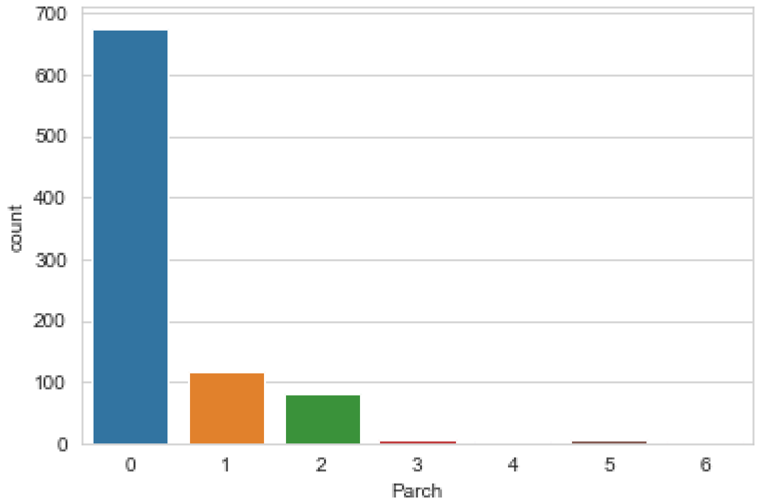
```
0    676
1    118
2     80
5      5
3      5
4      4
6      1
Name: Parch, dtype: int64
```

In [25]:

```
sns.countplot(x=titanic.Parch)
```

Out[25]:

```
<AxesSubplot:xlabel='Parch', ylabel='count'>
```

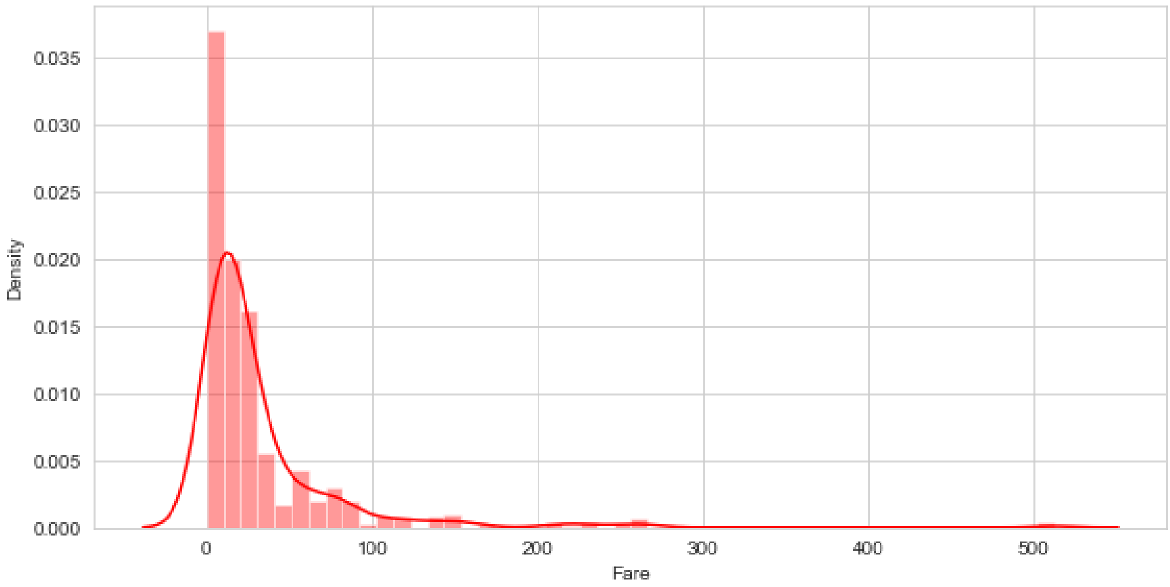


In [26]:

```
plt.figure(figsize=(10,5))
plt.xlabel("Fare")
sns.distplot(titanic.Fare,color="red")
```

Out[26]:

```
<AxesSubplot:xlabel='Fare', ylabel='Density'>
```

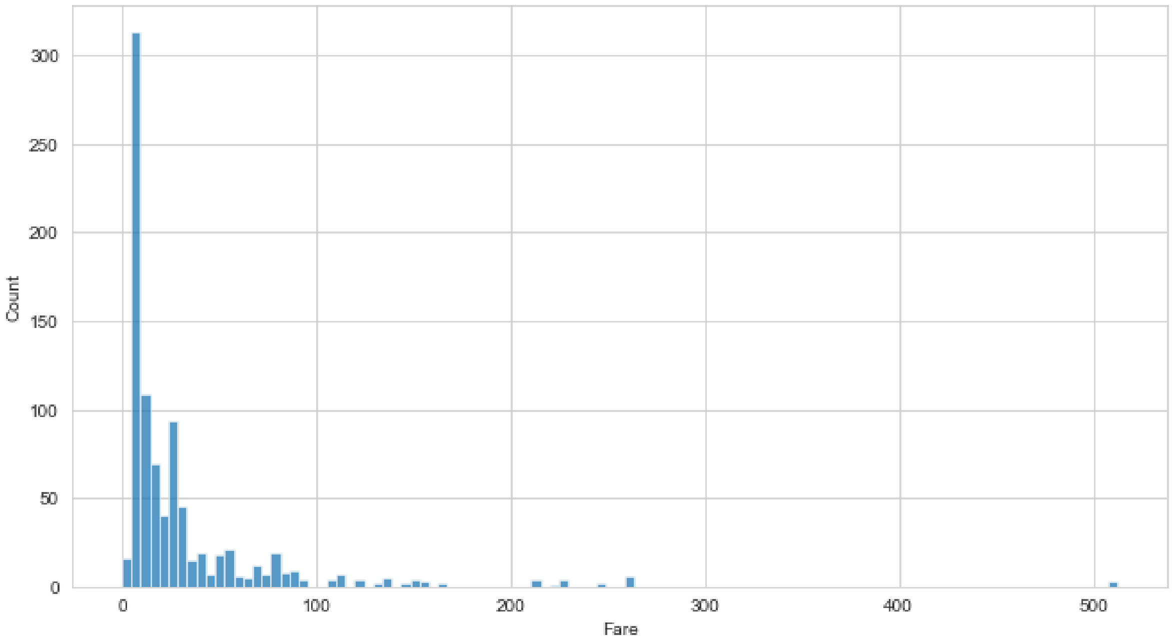


In [27]:

```
plt.figure(figsize=(11,6))
sns.histplot(x=titanic.Fare)
```

Out[27]:

```
<AxesSubplot:xlabel='Fare', ylabel='Count'>
```

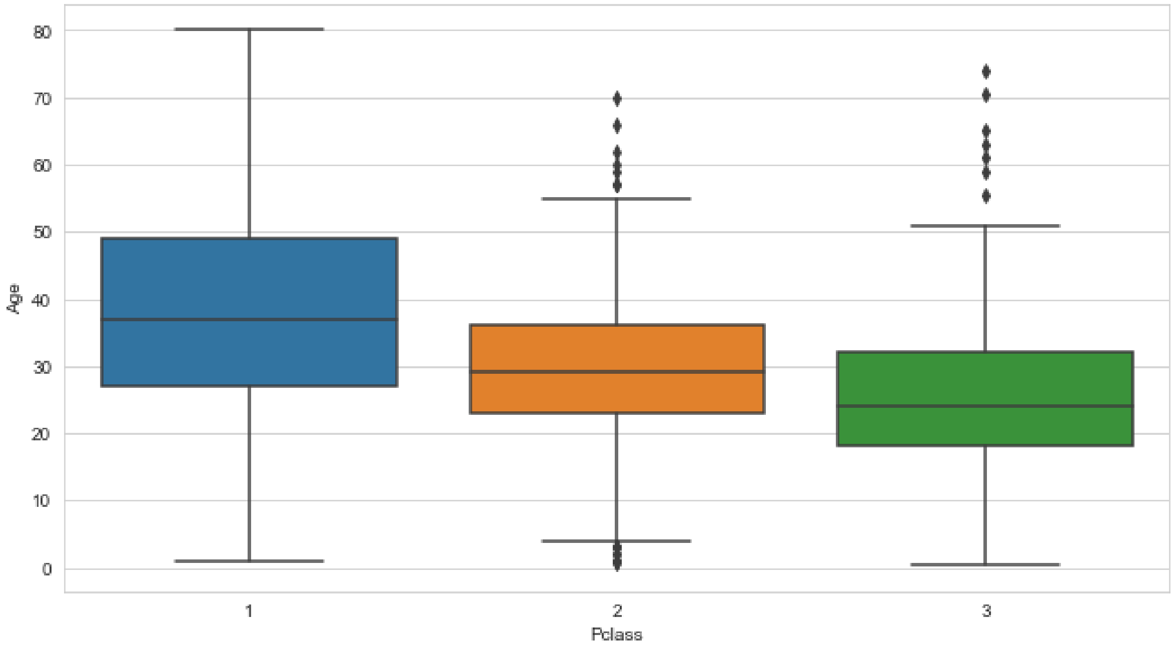


In [28]:

```
plt.figure(figsize=(11,6))
sns.boxplot(x='Pclass',y='Age',data=titanic)
```

Out[28]:

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



In [29]:

```
round(titanic.groupby("Pclass")["Age"].mean())
```

Out[29]:

```
Pclass
1    38.0
2    30.0
3    25.0
Name: Age, dtype: float64
```

In [30]:

```
titanic.loc[(titanic['Pclass']==1) & (titanic['Age'].isnull()),'Age']=38
```

In [31]:

```
titanic.loc[(titanic['Pclass']==2) & (titanic['Age'].isnull()),'Age']=30
```

In [32]:

```
titanic.loc[(titanic['Pclass']==3) & (titanic['Age'].isnull()),'Age']=25
```

In [33]:

```
titanic.Age.isnull().sum()
```

Out[33]:

```
0
```

In [34]:

```
titanic.info()
```

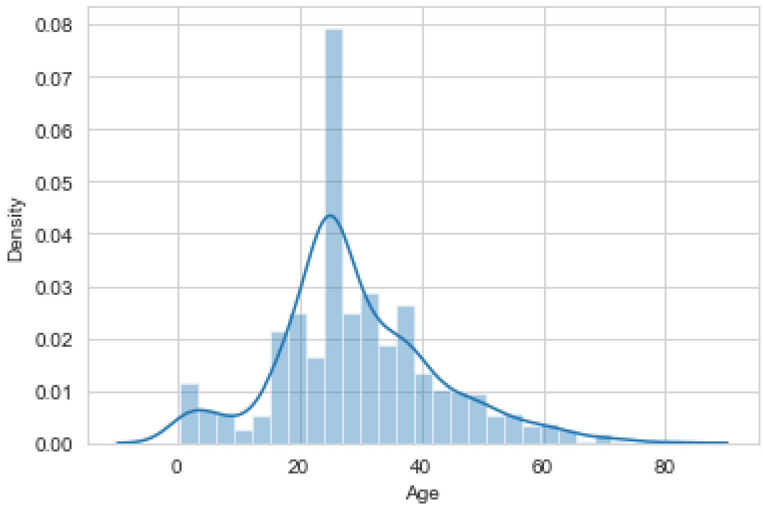
```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 889 entries, 0 to 890
Data columns (total 12 columns):
 #   Column       Non-Null Count  Dtype  
---  ------       --------------  -----  
 0   PassengerId  889 non-null    int64  
 1   Survived     889 non-null    int64  
 2   Pclass       889 non-null    int64  
 3   Name         889 non-null    object 
 4   Sex          889 non-null    object 
 5   Age          889 non-null    float64
 6   SibSp        889 non-null    int64  
 7   Parch        889 non-null    int64  
 8   Ticket       889 non-null    object 
 9   Fare         889 non-null    float64
 10  Cabin        202 non-null    object 
 11  Embarked     889 non-null    object 
dtypes: float64(2), int64(5), object(5)
memory usage: 90.3+ KB
```

In [35]:

```
sns.distplot(titanic.Age)
```

Out[35]:

```
<AxesSubplot:xlabel='Age', ylabel='Density'>
```



In [36]:

```
titanic.Cabin.isna().sum()/titanic.shape[0]*100 # Approx 77% data is missing in Cabin, so b
```

Out[36]:

```
77.27784026996626
```

In [37]:

```
titanic.drop('Cabin',axis=1,inplace=True)
```

In [38]:

```
titanic.drop(['Name','Ticket'],axis=1,inplace=True) # Also dropping Name and Ticket as they
```

In [39]:

```
titanic.head()
```

Out[39]:

	PassengerId	Survived	Pclass	Sex	Age	SibSp	Parch	Fare	Embarked
0	1	0	3	male	22.0	1	0	7.2500	S
1	2	1	1	female	38.0	1	0	71.2833	C
2	3	1	3	female	26.0	0	0	7.9250	S
3	4	1	1	female	35.0	1	0	53.1000	S
4	5	0	3	male	35.0	0	0	8.0500	S

In [40]:

```
dummies=pd.get_dummies(titanic[['Sex','Embarked']],drop_first=True)
```

In [41]:

```
titanic=pd.concat([titanic,dummies],axis=1)
```

In [42]:

```
titanic.head()
```

Out[42]:

	PassengerId	Survived	Pclass	Sex	Age	SibSp	Parch	Fare	Embarked	Sex_male
0	1	0	3	male	22.0	1	0	7.2500	S	1
1	2	1	1	female	38.0	1	0	71.2833	C	0
2	3	1	3	female	26.0	0	0	7.9250	S	0
3	4	1	1	female	35.0	1	0	53.1000	S	0
4	5	0	3	male	35.0	0	0	8.0500	S	1

In [43]:

```
titanic.drop(['Sex','Embarked','PassengerId'],axis=1,inplace=True)
```

In [44]:

```
titanic.head()
```

Out[44]:

	Survived	Pclass	Age	SibSp	Parch	Fare	Sex_male	Embarked_Q	Embarked_S
0	0	3	22.0	1	0	7.2500	1	0	1
1	1	1	38.0	1	0	71.2833	0	0	0
2	1	3	26.0	0	0	7.9250	0	0	1
3	1	1	35.0	1	0	53.1000	0	0	1
4	0	3	35.0	0	0	8.0500	1	0	1

Building a Logistic Regression Model

In [45]:

```
x=titanic.drop('Survived',axis=1)
y=titanic['Survived']
```

In [46]:

```
from sklearn.model_selection import train_test_split
```

In [47]:

```
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.3,random_state=10)
```

In [48]:

```
x_train.shape,y_train.shape,x_test.shape,y_test.shape
```

Out[48]:

```
((622, 8), (622,), (267, 8), (267,))
```

In [49]:

```
from sklearn.linear_model import LogisticRegression
```

In [50]:

```
logmodel=LogisticRegression()
logmodel.fit(x_train,y_train)
```

Out[50]:

```
LogisticRegression()
```

In [51]:

```
pred_y=logmodel.predict(x_test)
```

In [52]:

```
from sklearn.metrics import confusion_matrix
```

In [53]:

```
accuracy=confusion_matrix(y_test,pred_y)
```

In [54]:

```
accuracy
```

Out[54]:

```
array([[150,  19],
       [ 33,  65]], dtype=int64)
```

In [55]:

```
from sklearn.metrics import accuracy_score
```

In [56]:

```
accuracy=accuracy_score(y_test,pred_y)
accuracy
```

Out[56]:

```
0.8052434456928839
```

In [57]:

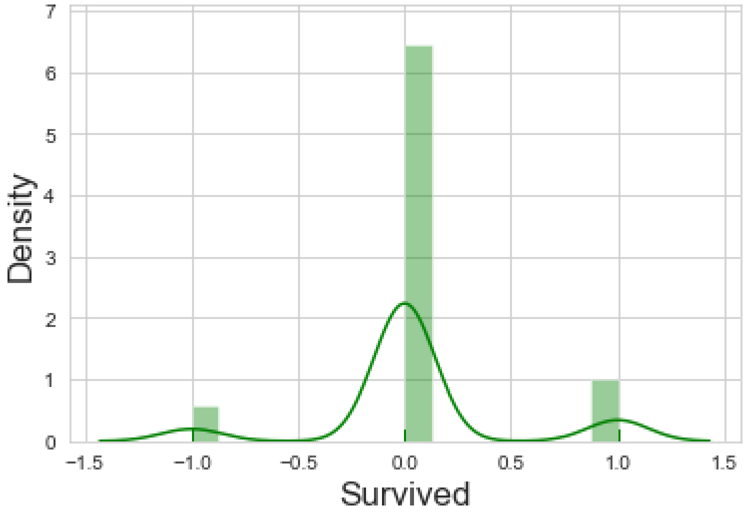
```
res=y_test-pred_y
```

In [58]:

```
plt.xlabel("errors",fontsize=17)
plt.ylabel("Density",fontsize=17)
sns.distplot(res,rug=True,color='green')
```

Out[58]:

```
<AxesSubplot:xlabel='Survived', ylabel='Density'>
```



In [ ]: