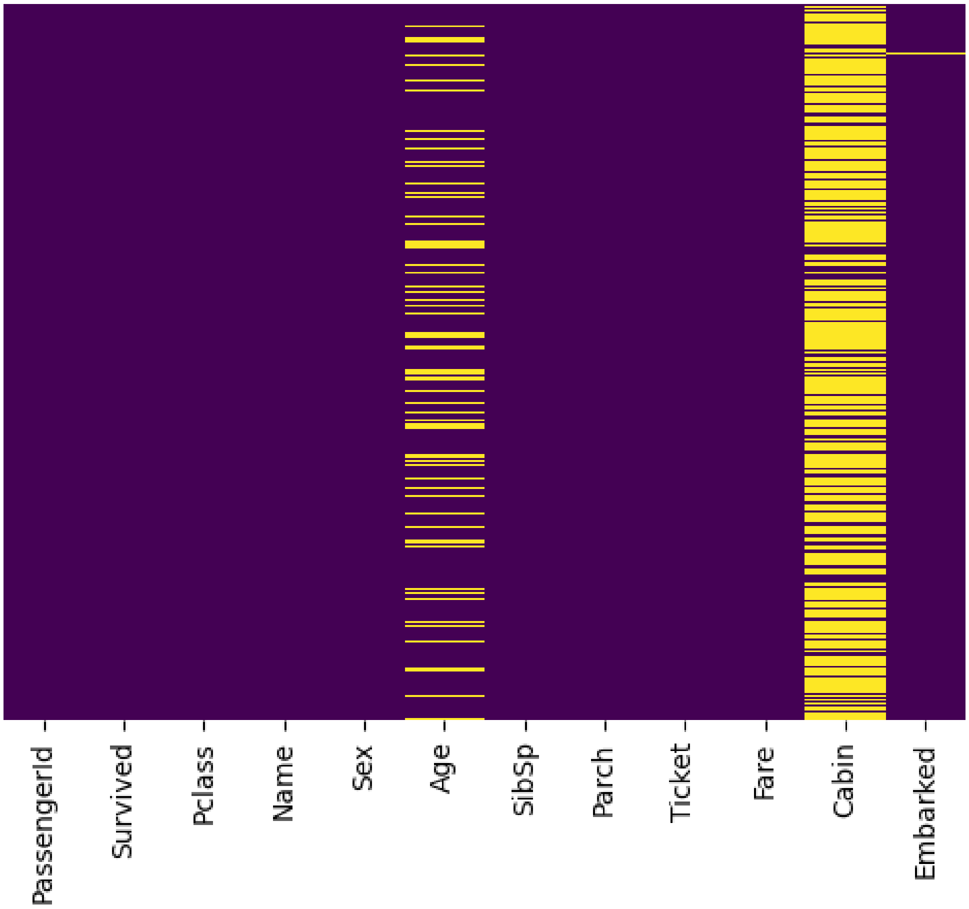
```
#Assignment 8

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline

pwd

'C:\\Users\\Stev3raj\\Desktop\\DSBDA Practical Assignments\\All ipynb'

train = pd.read_csv('titanic_train.csv')

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

train.isnull()

     PassengerId  Survived  Pclass   Name    Sex    Age  SibSp  Parch
Ticket  \
0          False     False   False  False  False  False  False  False
False
1          False     False   False  False  False  False  False  False
False
```

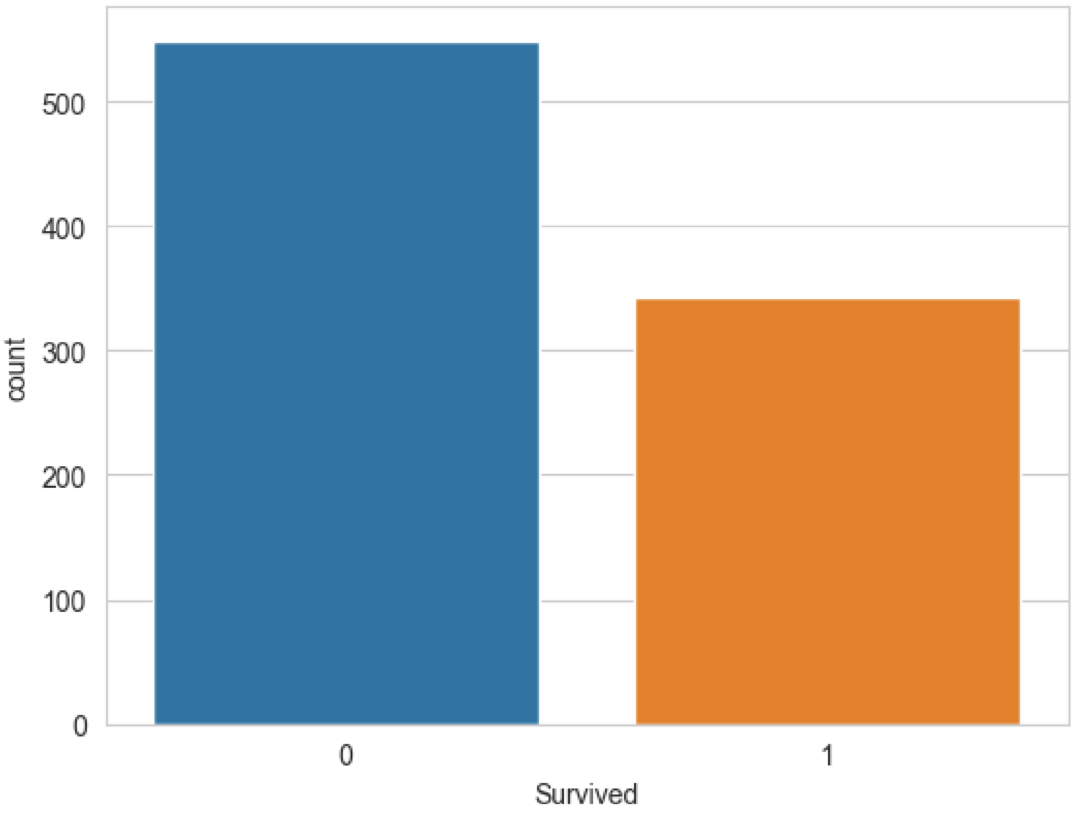
```
2          False     False   False  False  False  False  False  False
False
3          False     False   False  False  False  False  False  False
False
4          False     False   False  False  False  False  False  False
False
..           ...       ...     ...    ...    ...    ...    ...    ...
...
886        False     False   False  False  False  False  False  False
False
887        False     False   False  False  False  False  False  False
False
888        False     False   False  False  False   True  False  False
False
889        False     False   False  False  False  False  False  False
False
890        False     False   False  False  False  False  False  False
False

      Fare  Cabin  Embarked
0    False   True     False
1    False  False     False
2    False   True     False
3    False  False     False
4    False   True     False
..     ...    ...       ...
886  False   True     False
887  False  False     False
888  False   True     False
889  False  False     False
890  False   True     False

[891 rows x 12 columns]
```

```
sns.heatmap(train.isnull(),yticklabels = False, cbar = False, cmap =
'viridis')
```

```
<Axes: >
```

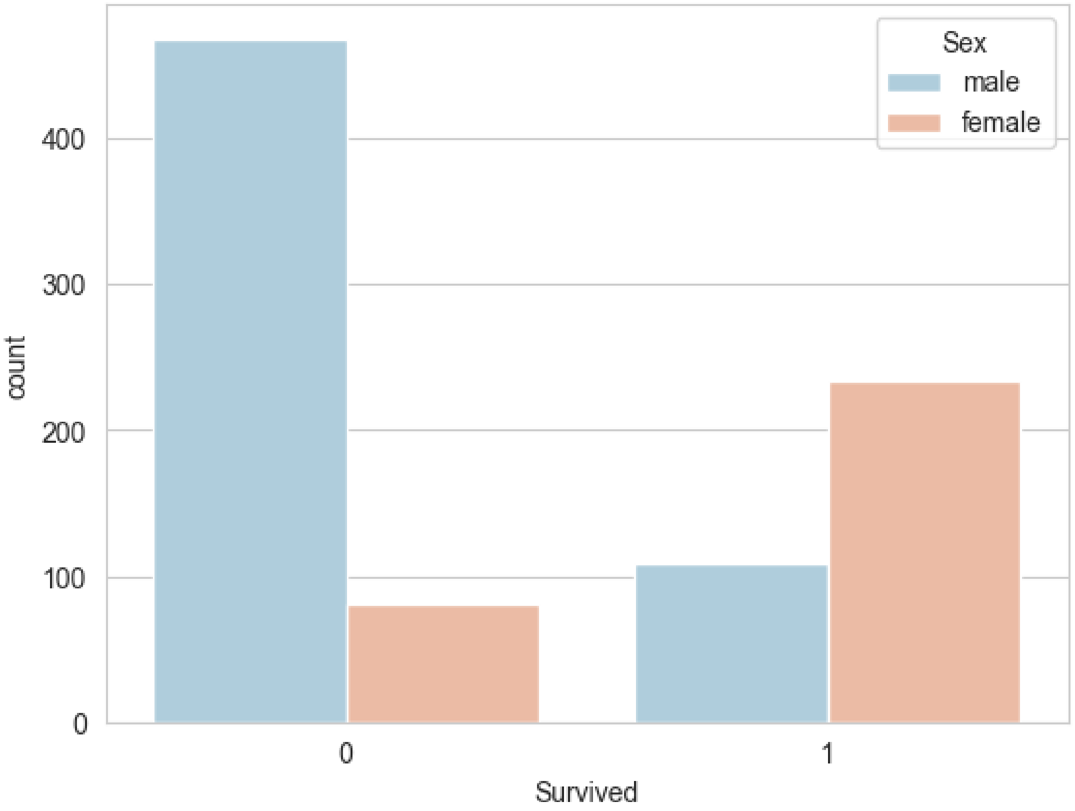


```
sns.set_style('whitegrid')
sns.countplot(x = 'Survived', data = train)

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

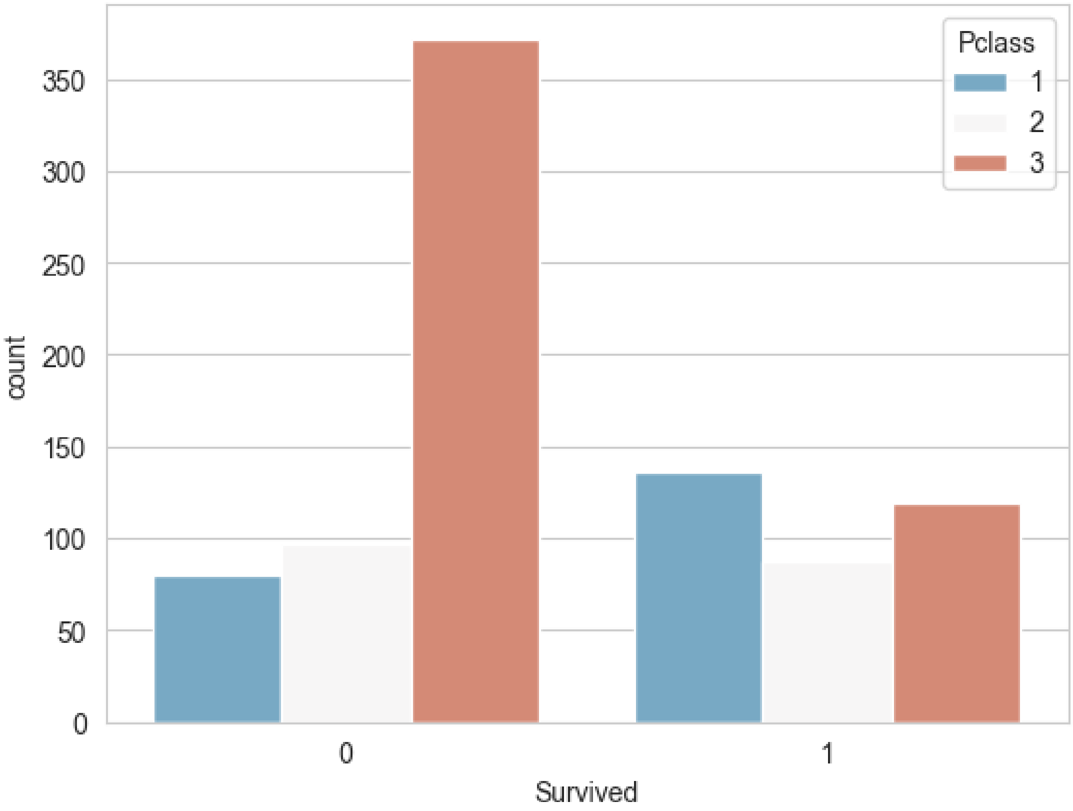


```
sns.set_style('whitegrid')
sns.countplot(x = 'Survived', hue = 'Sex', data = train, palette =
"RdBu_r" )
```

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



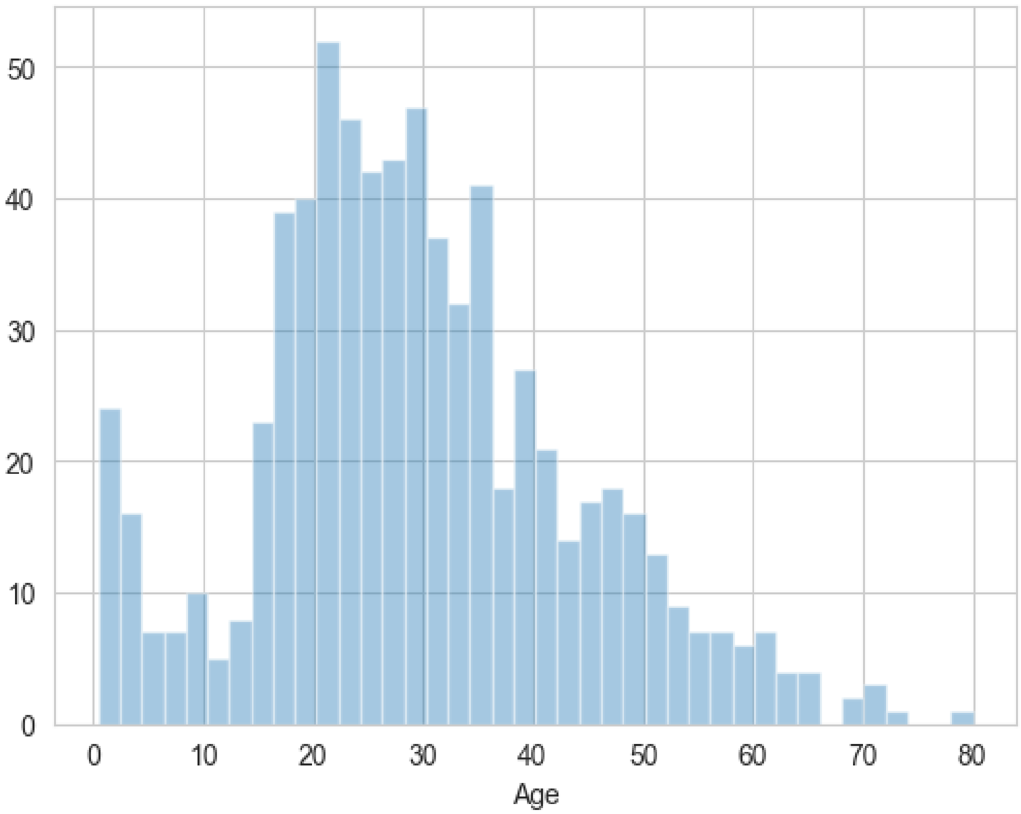
```
sns.set_style('whitegrid')
sns.countplot(x = 'Survived', hue = 'Pclass', data = train, palette =
"RdBu_r" )

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



```
sns.distplot(train["Age"].dropna(),kde = False, bins = 40)
```

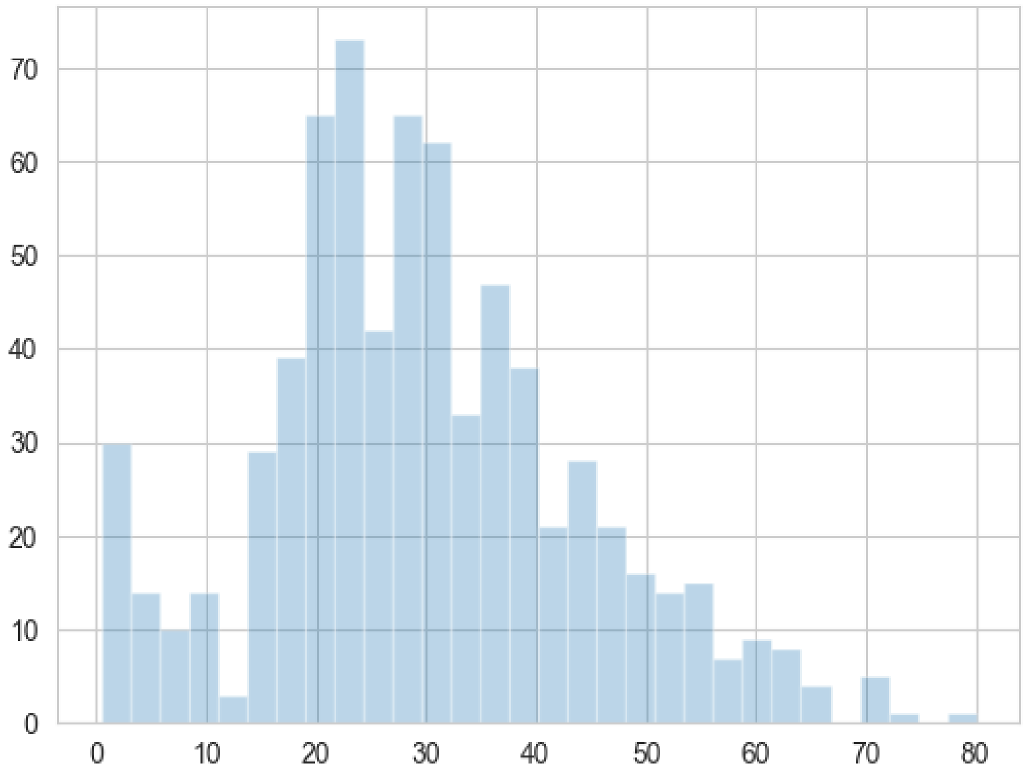
```
C:\Users\Stev3raj\AppData\Roaming\Python\Python311\site-packages\
seaborn\distributions.py:2551: FutureWarning: `distplot` is a
deprecated function and will be removed in a future version. Please
adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for
histograms).
  warnings.warn(msg, FutureWarning)
```

```
<Axes: xlabel='Age'>
```



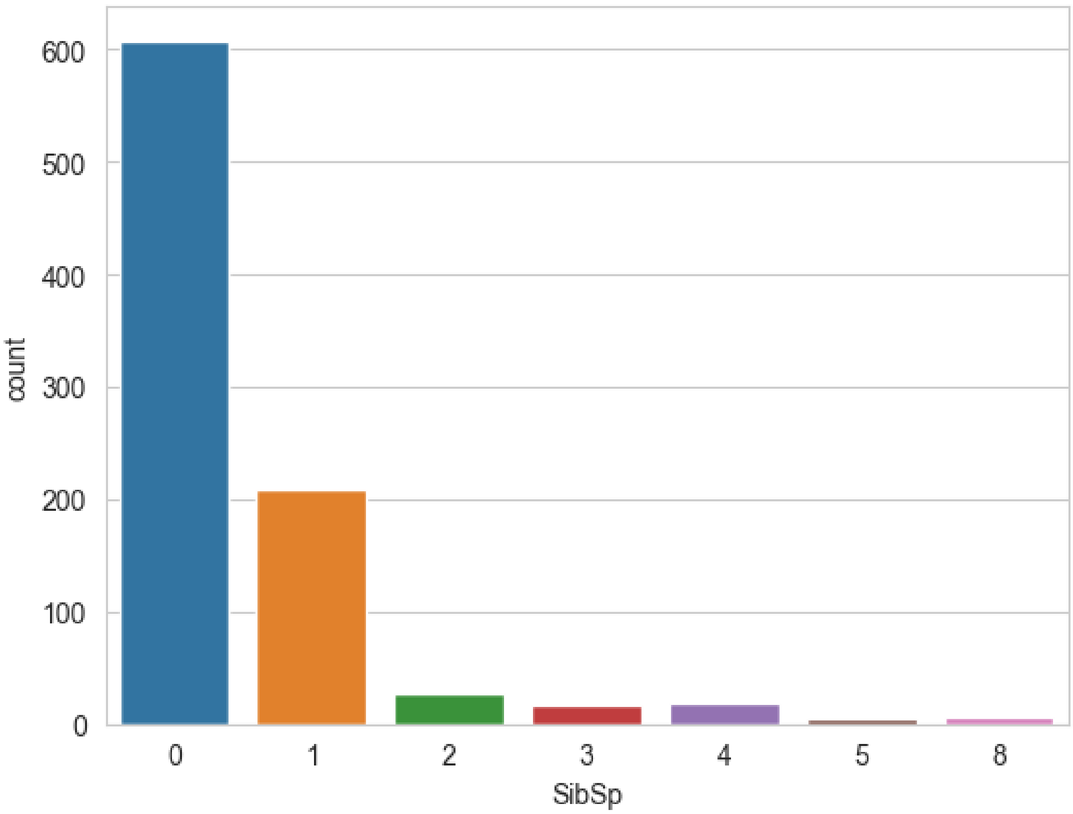
```
train['Age'].hist( bins= 30, alpha = 0.3)
```

```
<Axes: >
```



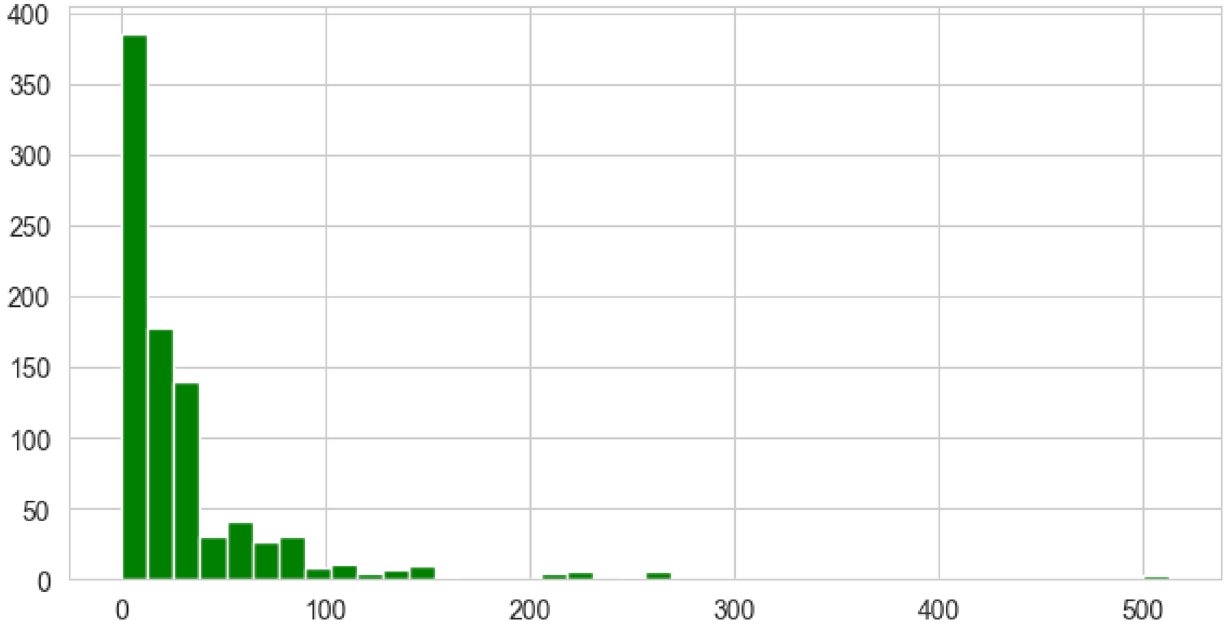
```
sns.countplot(x ='SibSp' , data = train)
```

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

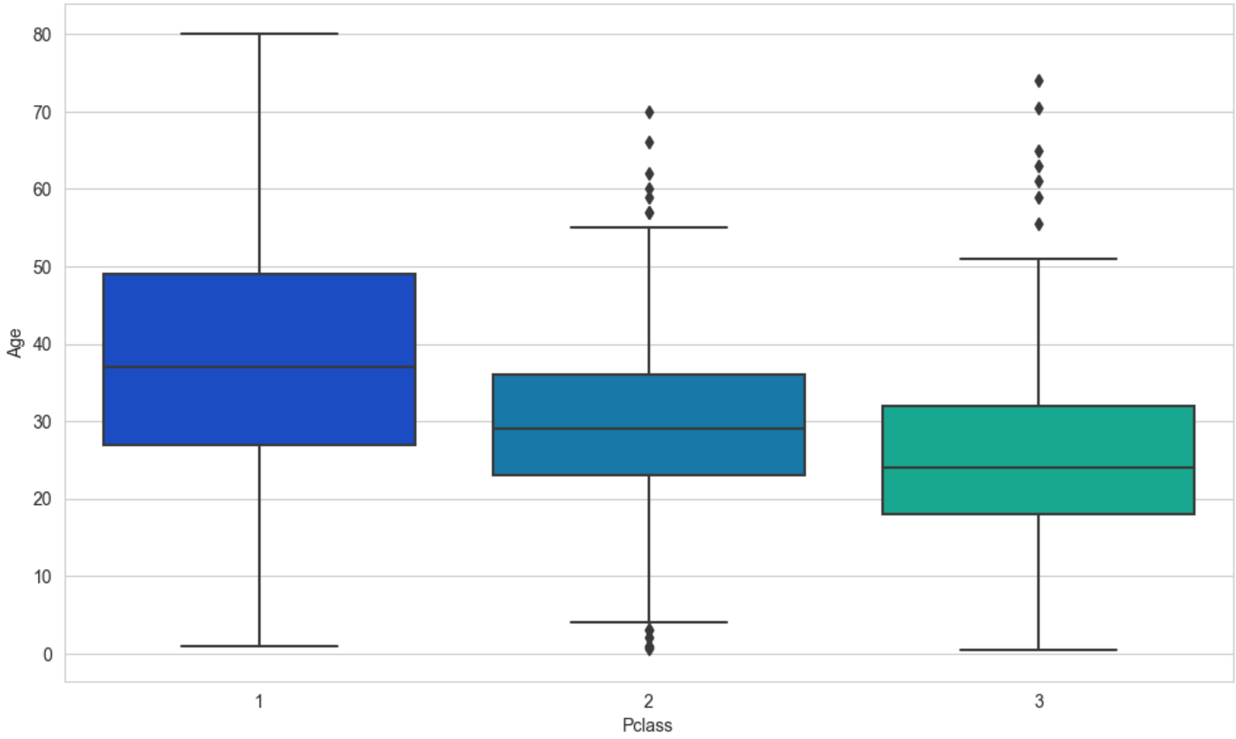


```
train['Fare'].hist(color = 'green', bins = 40, figsize = (8, 4))
```

```
<Axes: >
```

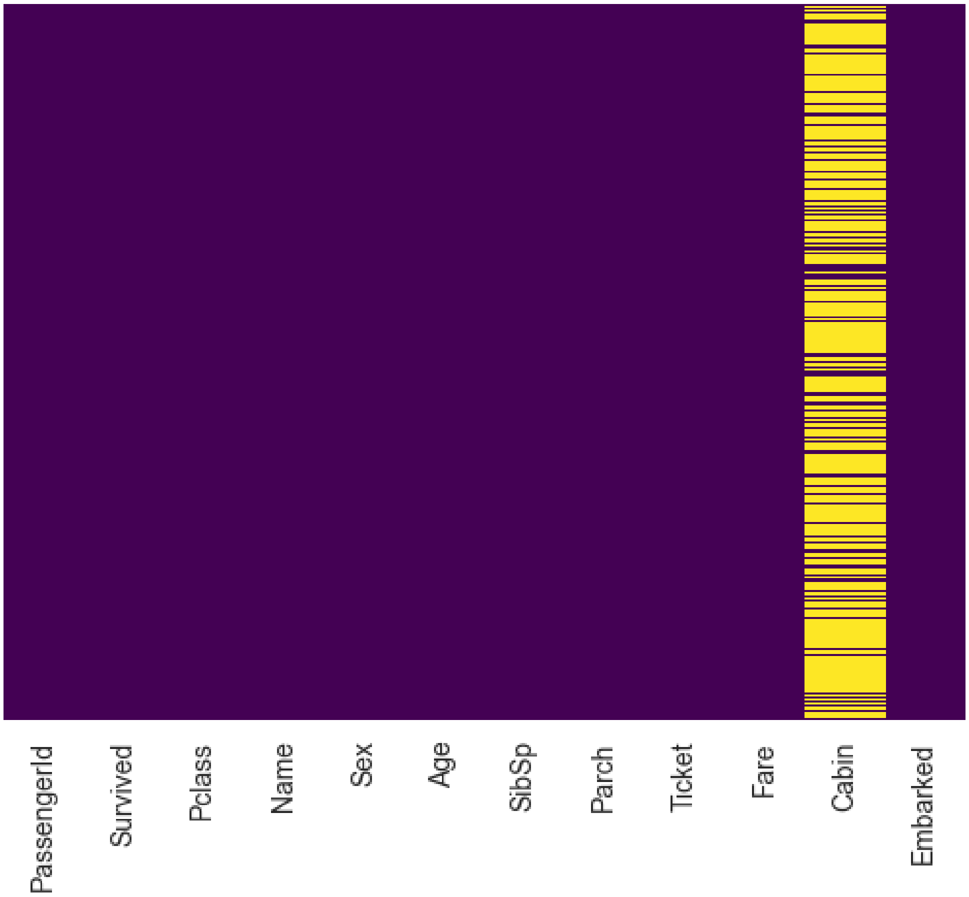


```
#Data Cleaning

plt.figure(figsize = (12, 7))
sns.boxplot(x = 'Pclass', y = 'Age', data = train, palette = 'winter')

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



```
def impute_age(cols):
    Age = cols[0]
    Pclass = cols[1]

    if pd.isnull(Age):

        if Pclass == 1:
            return 37

        elif Pclass == 2:
            return 29

        else:
            return 24

    else:
         return Age

train['Age'] = train[['Age', 'Pclass']].apply(impute_age, axis = 1)
```

```
sns.heatmap(train.isnull(), yticklabels = False, cbar = False, cmap =
'viridis')

<Axes: >
```



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

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 Embarked
0      0         A/5 21171   7.2500        S
1      0          PC 17599  71.2833        C
2      0  STON/O2. 3101282   7.9250        S
3      0            113803  53.1000        S
4      0            373450   8.0500        S
```

```
train.dropna(inplace = True)
```

```
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 Embarked
0      0         A/5 21171   7.2500        S
1      0          PC 17599  71.2833        C
2      0  STON/O2. 3101282   7.9250        S
3      0            113803  53.1000        S
4      0            373450   8.0500        S
```

```
pd.get_dummies(train ["Embarked"], drop_first = True).head()
```

```
       Q      S
0  False   True
1  False  False
2  False   True
```

```
3  False   True
4  False   True

sex = pd.get_dummies(train ["Sex"], drop_first = True)
embark = pd.get_dummies(train['Embarked'], drop_first = True)

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 Embarked
0      0         A/5 21171   7.2500        S
1      0          PC 17599  71.2833        C
2      0  STON/O2. 3101282   7.9250        S
3      0            113803  53.1000        S
4      0            373450   8.0500        S

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 Embarked
0      0         A/5 21171   7.2500        S
1      0          PC 17599  71.2833        C
2      0  STON/O2. 3101282   7.9250        S
3      0            113803  53.1000        S
4      0            373450   8.0500        S
```

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

```
train.head()
```

```
   PassengerId  Survived  Pclass   Age  SibSp  Parch     Fare
0            1         0       3  22.0      1      0   7.2500
1            2         1       1  38.0      1      0  71.2833
2            3         1       3  26.0      0      0   7.9250
3            4         1       1  35.0      1      0  53.1000
4            5         0       3  35.0      0      0   8.0500
```

```
train = pd.concat([train, sex, embark], axis = 1)
```

```
train.head()
```

```
   PassengerId  Survived  Pclass   Age  SibSp  Parch     Fare   male
Q  \
0            1         0       3  22.0      1      0   7.2500   True
False
1            2         1       1  38.0      1      0  71.2833  False
False
2            3         1       3  26.0      0      0   7.9250  False
False
3            4         1       1  35.0      1      0  53.1000  False
False
4            5         0       3  35.0      0      0   8.0500   True
False

       S
0   True
1  False
2   True
3   True
4   True
```

```
train.drop('Survived', axis = 1).head()
```

```
   PassengerId  Pclass   Age  SibSp  Parch     Fare   male      Q  
S
0            1       3  22.0      1      0   7.2500   True  False  
True
1            2       1  38.0      1      0  71.2833  False  False  
False
2            3       3  26.0      0      0   7.9250  False  False  
True
3            4       1  35.0      1      0  53.1000  False  False  
True
4            5       3  35.0      0      0   8.0500   True  False  
True
```

```python
train['Survived'].head()
```

```
0    0
1    1
2    1
3    1
4    0
Name: Survived, dtype: int64
```

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

```python
X_train, X_test, y_train, y_test = 
train_test_split(train.drop('Survived', axis = 1), 
                                                    train['Survived'], 
test_size = 0.3, random_state = 1)
```

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

```python
logmodel = LogisticRegression()
logmodel.fit(X_train, y_train)
```

```
C:\Users\Stev3raj\AppData\Roaming\Python\Python311\site-packages\
sklearn\linear_model\_logistic.py:460: ConvergenceWarning: lbfgs 
failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as 
shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:

https://scikit-learn.org/stable/modules/linear_model.html#logistic-
regression
  n_iter_i = _check_optimize_result(

LogisticRegression()
```

```python
predictions = logmodel.predict(X_test)
```

```
from sklearn.metrics import confusion_matrix

accuracy=confusion_matrix(y_test,predictions)

accuracy
```

```
array([[140,  26],
       [ 24,  77]], dtype=int64)
```

```
from sklearn.metrics import accuracy_score

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

```
0.8127340823970037
```

```
predictions
```

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

```
from sklearn.metrics import classification_report

print(classification_report(y_test,predictions))
```

```
              precision    recall  f1-score   support

           0       0.85      0.84      0.85       166
           1       0.75      0.76      0.75       101
```

```
    accuracy                           0.81       267
   macro avg       0.80      0.80      0.80       267
weighted avg       0.81      0.81      0.81       267
```