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

dataset =pd.read_csv('titanic.csv')

dataset.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 418 entries, 0 to 417
Data columns (total 12 columns):
 #   Column       Non-Null Count  Dtype  
---  ------       --------------  -----  
 0   PassengerId  418 non-null    int64  
 1   Survived     418 non-null    int64  
 2   Pclass       418 non-null    int64  
 3   Name         418 non-null    object 
 4   Sex          418 non-null    object 
 5   Age          332 non-null    float64
 6   SibSp        418 non-null    int64  
 7   Parch        418 non-null    int64  
 8   Ticket       418 non-null    object 
 9   Fare         417 non-null    float64
 10  Cabin        91 non-null     object 
 11  Embarked     418 non-null    object 
dtypes: float64(2), int64(5), object(5)
memory usage: 39.3+ KB
```

```
dataset.isna().sum()
```

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

```
dataset['PassengerId']
```

```
0       892
1       893
2       894
3       895
```

```
4       896
       ...
413    1305
414    1306
415    1307
416    1308
417    1309
Name: PassengerId, Length: 418, dtype: int64
```

```
dataset['Ticket']
```

```
0                  330911
1                  363272
2                  240276
3                  315154
4                 3101298
              ...
413             A.5. 3236
414              PC 17758
415    SOTON/O.Q. 3101262
416                359309
417                  2668
Name: Ticket, Length: 418, dtype: object
```

```
dataset['Age']
```

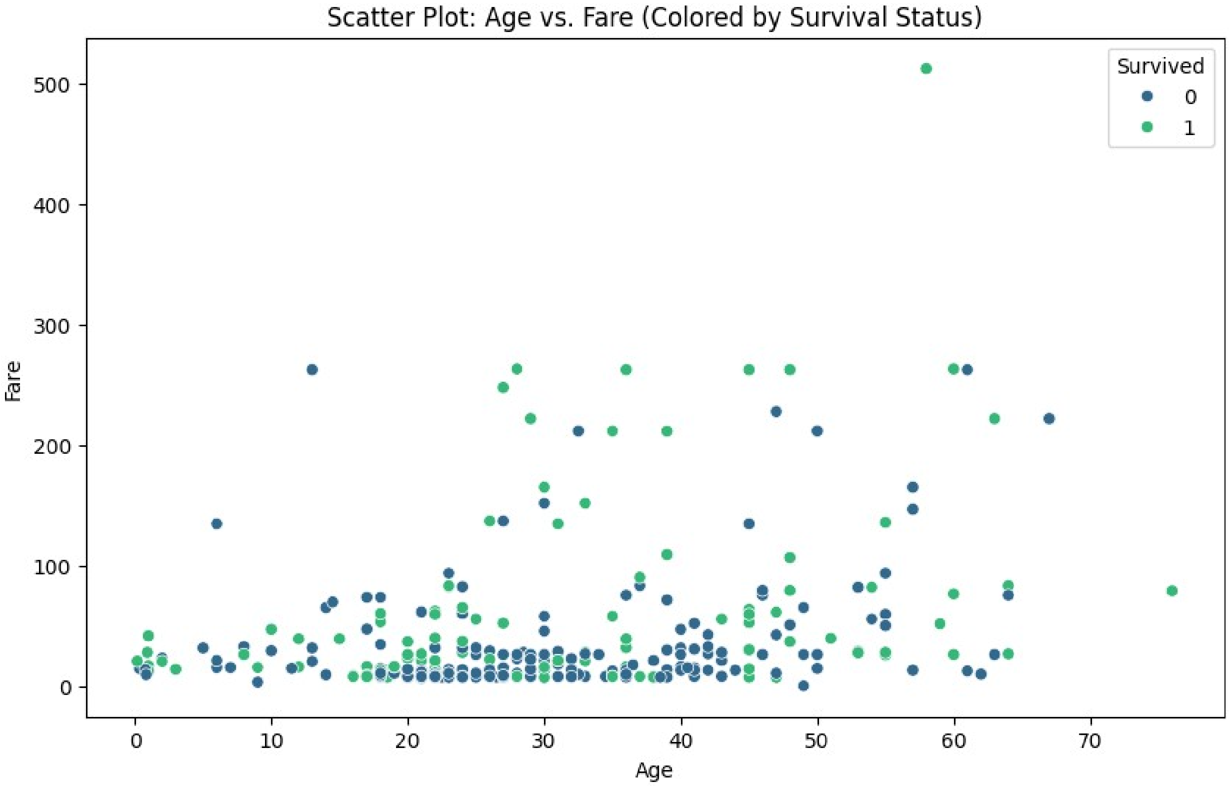
```
0      34.5
1      47.0
2      62.0
3      27.0
4      22.0
       ...
413     NaN
414    39.0
415    38.5
416     NaN
417     NaN
Name: Age, Length: 418, dtype: float64
```

```
dataset['Fare']
```

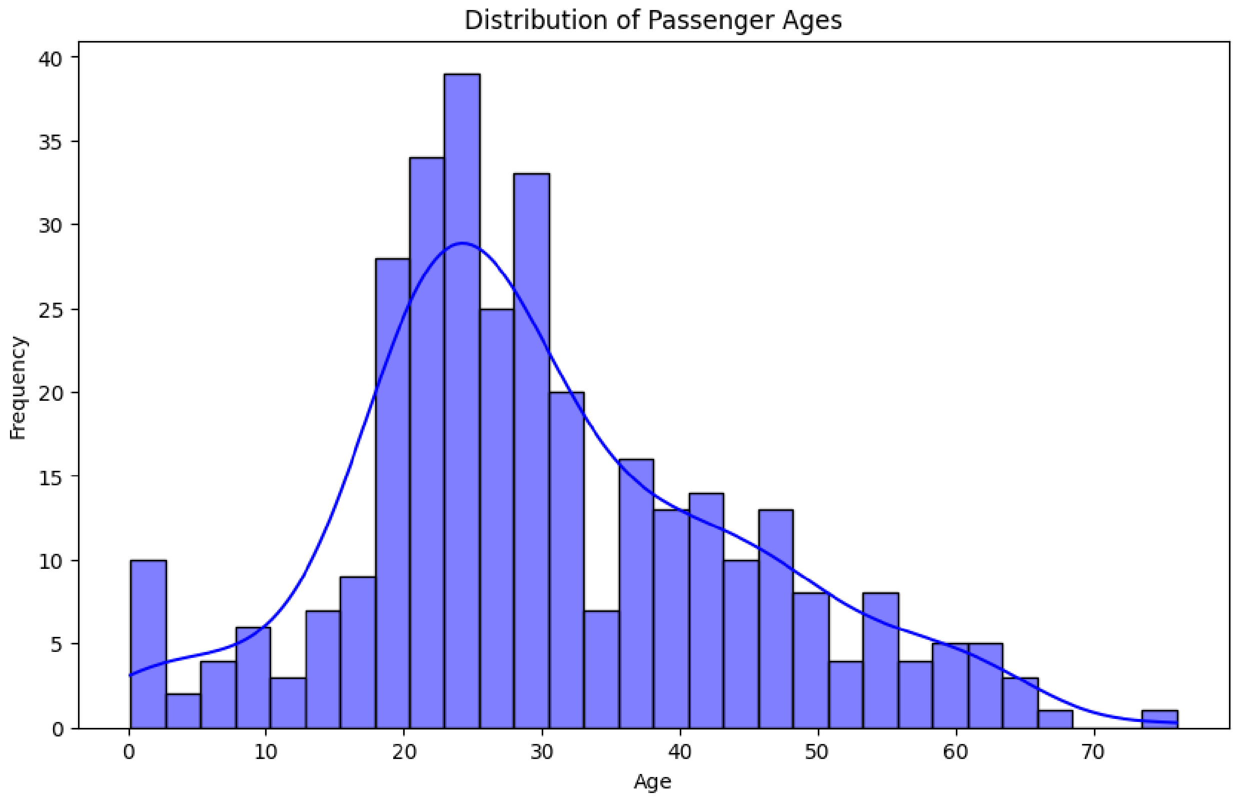
```
0        7.8292
1        7.0000
2        9.6875
3        8.6625
4       12.2875
         ...
413      8.0500
414    108.9000
415      7.2500
416      8.0500
```

```
417      22.3583
Name: Fare, Length: 418, dtype: float64
```

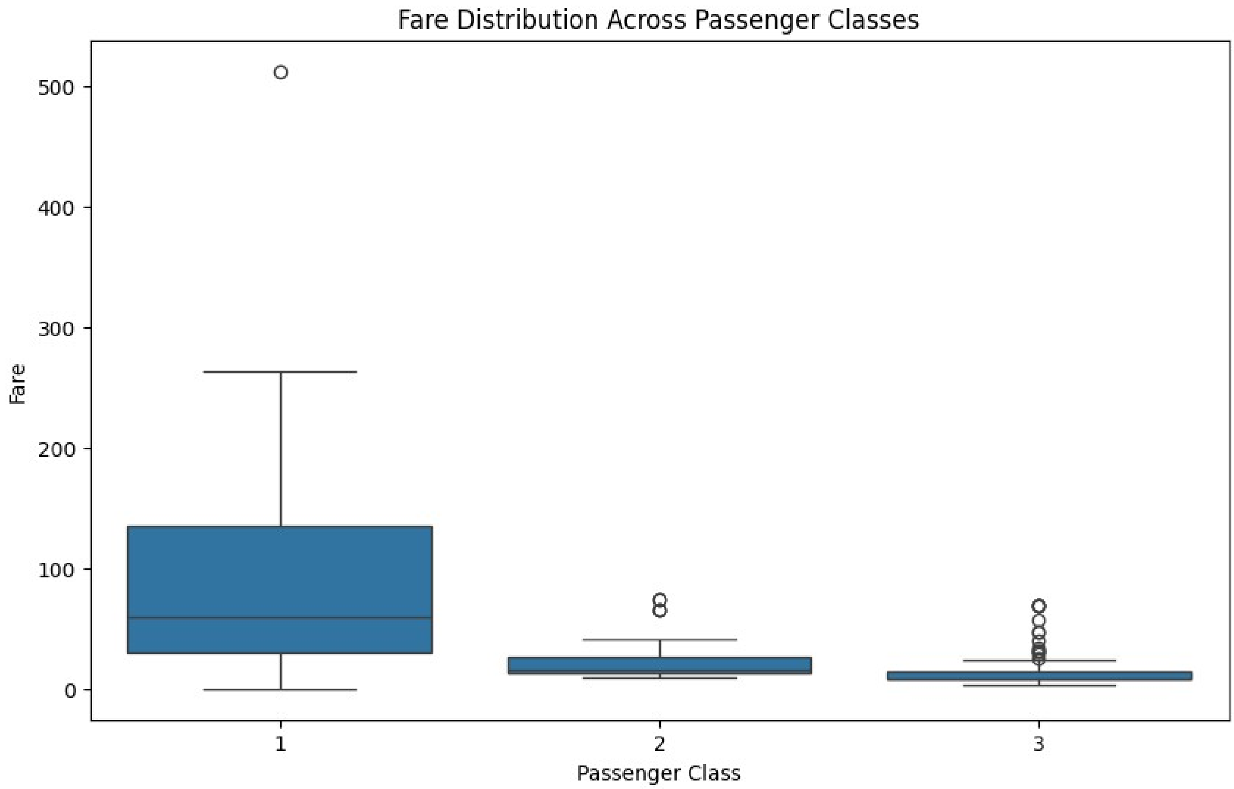
```python
plt.figure(figsize=(10, 6))
sns.scatterplot(x='Age', y='Fare', hue='Survived', data=dataset,
palette='viridis')
plt.title('Scatter Plot: Age vs. Fare (Colored by Survival Status)')
plt.xlabel('Age')
plt.ylabel('Fare')
plt.legend(title='Survived')
plt.show()
```



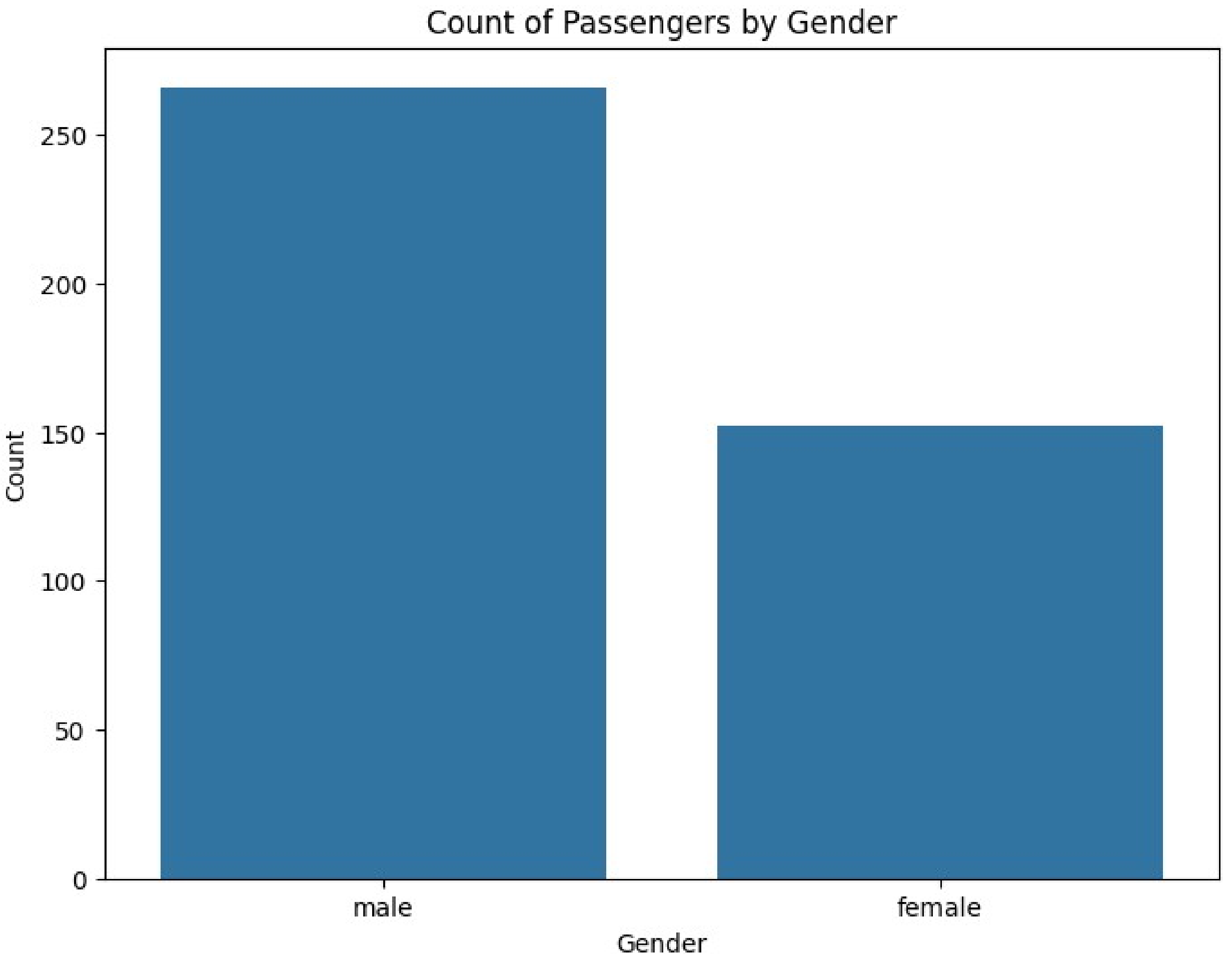
```python
plt.figure(figsize=(10, 6))
sns.histplot(dataset['Age'].dropna(), bins=30, kde=True, color='blue')
plt.title('Distribution of Passenger Ages')
plt.xlabel('Age')
plt.ylabel('Frequency')
plt.show()
```



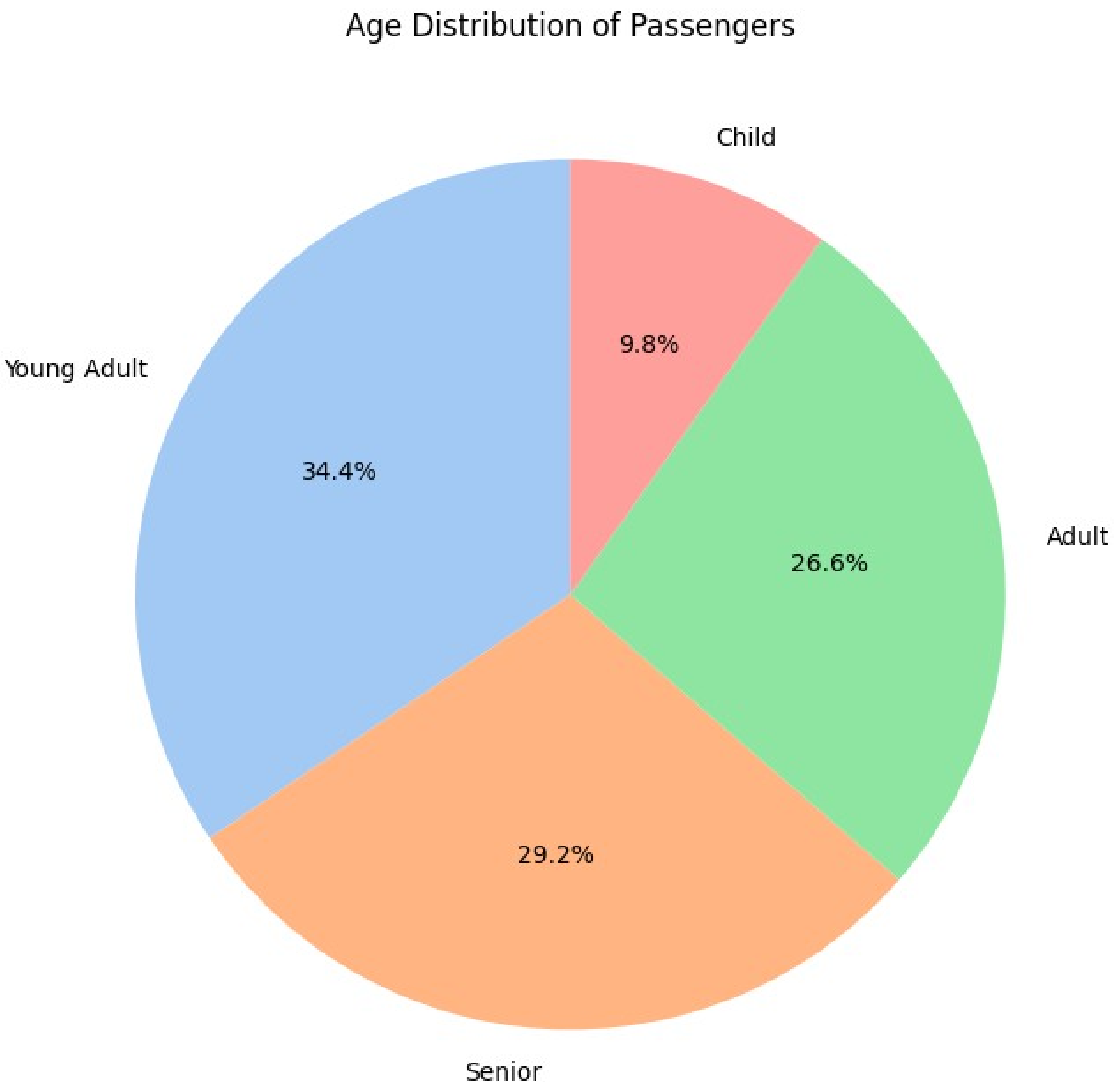
```
plt.figure(figsize=(10, 6))
sns.boxplot(x='Pclass', y='Fare', data=dataset)
plt.title('Fare Distribution Across Passenger Classes')
plt.xlabel('Passenger Class')
plt.ylabel('Fare')
plt.show()
```



```
plt.figure(figsize=(8, 6))
sns.countplot(x='Sex', data=dataset)
plt.title('Count of Passengers by Gender')
plt.xlabel('Gender')
plt.ylabel('Count')
plt.show()
```

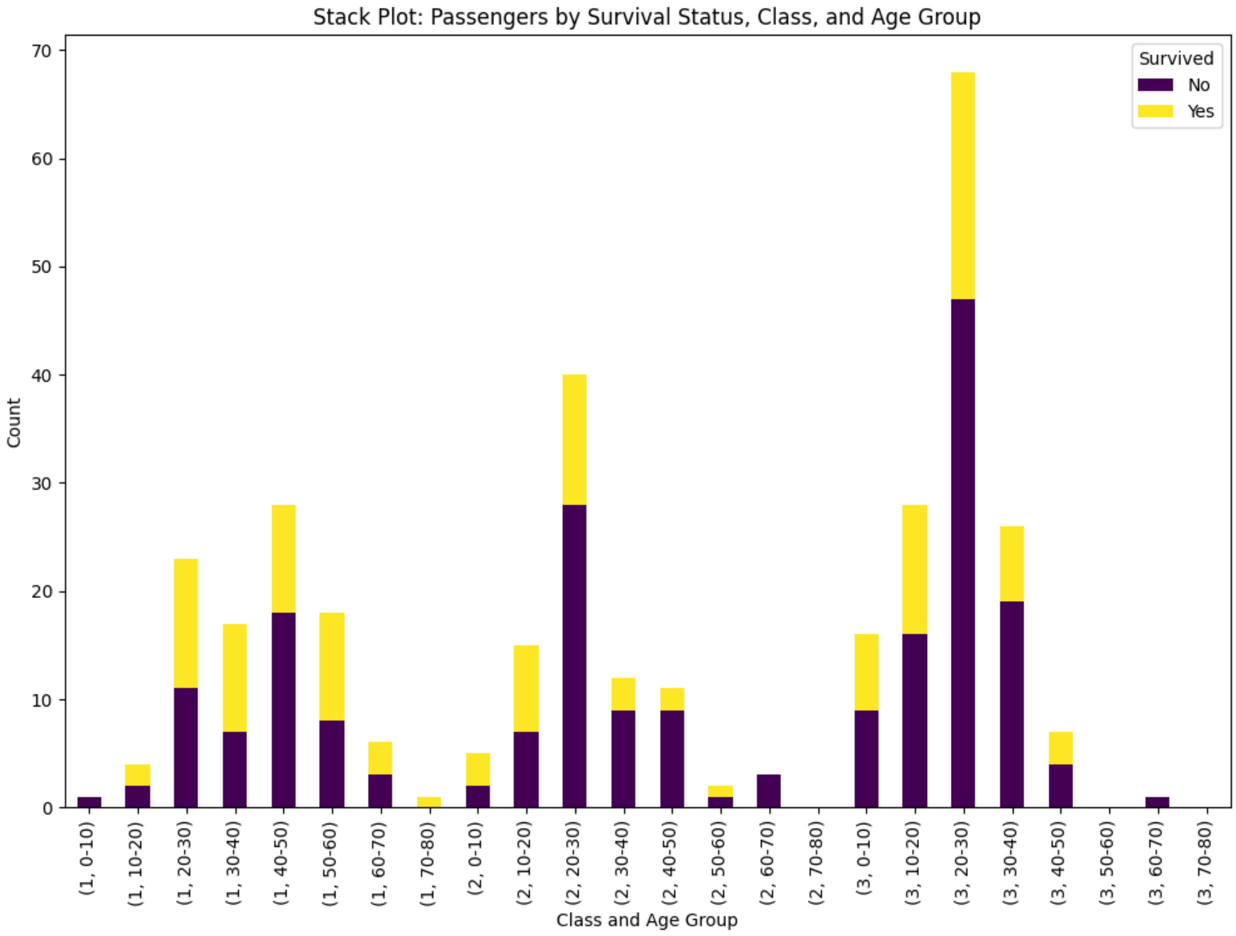


```python
def age_group(age):
    if age < 18:
        return 'Child'
    elif 18 <= age < 30:
        return 'Young Adult'
    elif 30 <= age < 50:
        return 'Adult'
    else:
        return 'Senior'

dataset['age_group'] = dataset['Age'].apply(age_group)

age_counts = dataset['age_group'].value_counts()
plt.figure(figsize=(8, 8))
plt.pie(age_counts, labels=age_counts.index, autopct='%1.1f%%',
colors=sns.color_palette('pastel'), startangle=90)
plt.title('Age Distribution of Passengers')
plt.show()
```



```
dataset['age_group'] = pd.cut(dataset['Age'], bins=[0, 10, 20, 30, 40,
50, 60, 70, 80], labels=['0-10', '10-20', '20-30', '30-40', '40-50',
'50-60', '60-70', '70-80'])

stack_data.plot(kind='bar', stacked=True, figsize=(12, 8),
colormap='viridis')
plt.title('Stack Plot: Passengers by Survival Status, Class, and Age
Group')
plt.xlabel('Class and Age Group')
plt.ylabel('Count')
plt.legend(title='Survived', labels=['No', 'Yes'])
plt.show()
```

```
<ipython-input-33-1ca25ea0bc85>:4: FutureWarning: The default of
observed=False is deprecated and will be changed to True in a future
version of pandas. Pass observed=False to retain current behavior or
observed=True to adopt the future default and silence this warning.
  stack_data = dataset.groupby(['Pclass', 'age_group',
'Survived']).size().unstack()
```


Stack Plot: Passengers by Survival Status, Class, and Age Group
Survived
No
Yes
Count
Class and Age Group
(1, 0-10)
(1, 10-20)
(1, 20-30)
(1, 30-40)
(1, 40-50)
(1, 50-60)
(1, 60-70)
(1, 70-80)
(2, 0-10)
(2, 10-20)
(2, 20-30)
(2, 30-40)
(2, 40-50)
(2, 50-60)
(2, 60-70)
(2, 70-80)
(3, 0-10)
(3, 10-20)
(3, 20-30)
(3, 30-40)
(3, 40-50)
(3, 50-60)
(3, 60-70)
(3, 70-80)