

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

```
df = pd.read_csv('/content/train (2).csv')
```

```
df.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
```

```
df.describe()
```

```
>>>
```

	PassengerId	Survived	Pclass	Age	SibSp	Parch	Fare
<b>count</b>	891.000000	891.000000	891.000000	714.000000	891.000000	891.000000	891.000000
<b>mean</b>	446.000000	0.383838	2.308642	29.699118	0.523008	0.381594	32.204200
<b>std</b>	257.353842	0.486592	0.836071	14.526497	1.102743	0.806057	49.693422
<b>min</b>	1.000000	0.000000	1.000000	0.420000	0.000000	0.000000	0.000000
<b>25%</b>	223.500000	0.000000	2.000000	20.125000	0.000000	0.000000	7.910450
<b>50%</b>	446.000000	0.000000	3.000000	28.000000	0.000000	0.000000	14.454200
<b>75%</b>	668.500000	1.000000	3.000000	38.000000	1.000000	0.000000	31.000000
<b>max</b>	891.000000	1.000000	3.000000	80.000000	8.000000	6.000000	512.329000

```
df['Survived'].value_counts()  
df['Pclass'].value_counts()  
df['Sex'].value_counts()
```

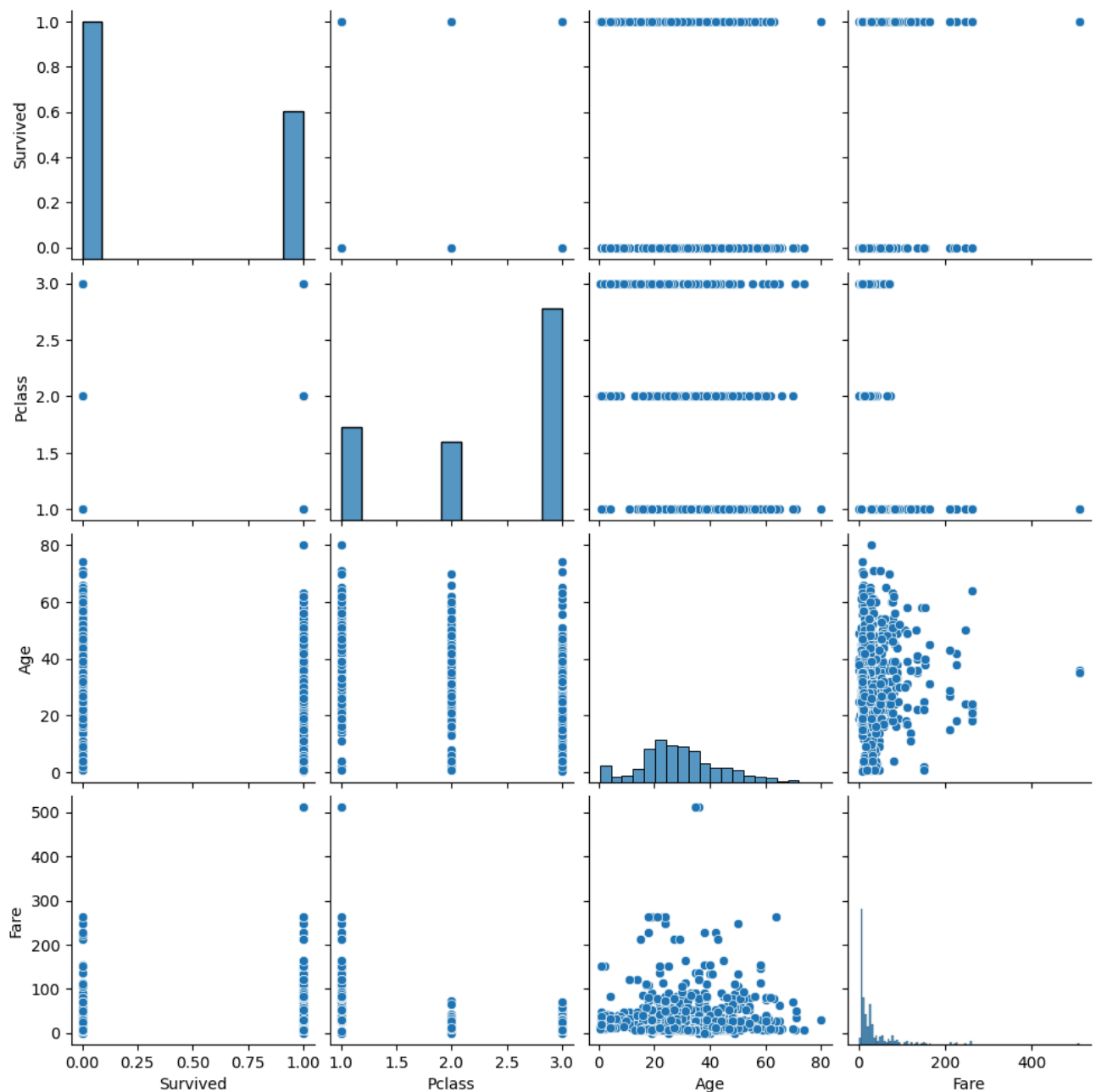


	count
<b>Sex</b>	
<b>male</b>	577
<b>female</b>	314

**dtype:** int64

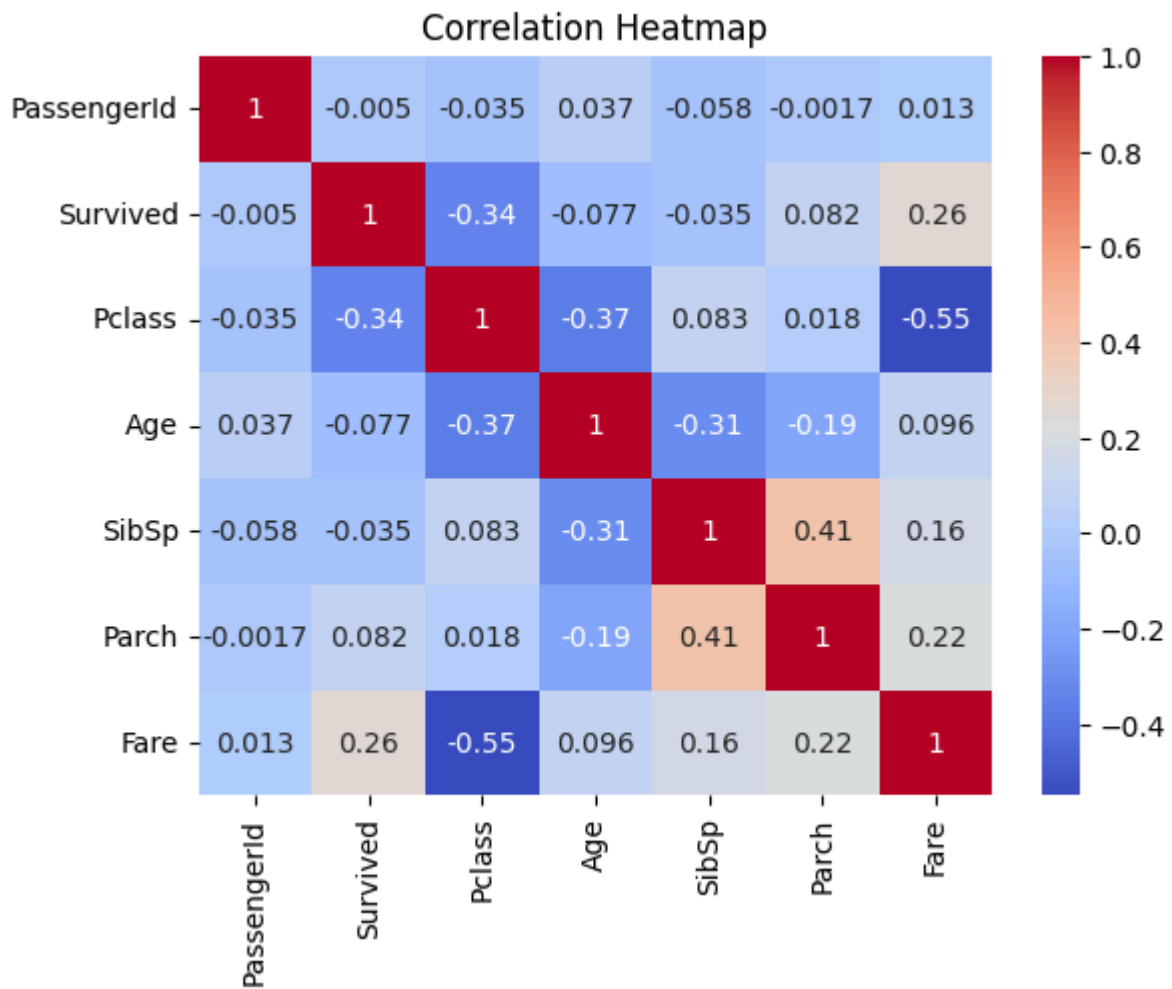
Pairplot: Survivors are more frequent in higher classes and younger ages.

```
sns.pairplot(df[['Survived', 'Pclass', 'Age', 'Fare']])  
plt.show()
```



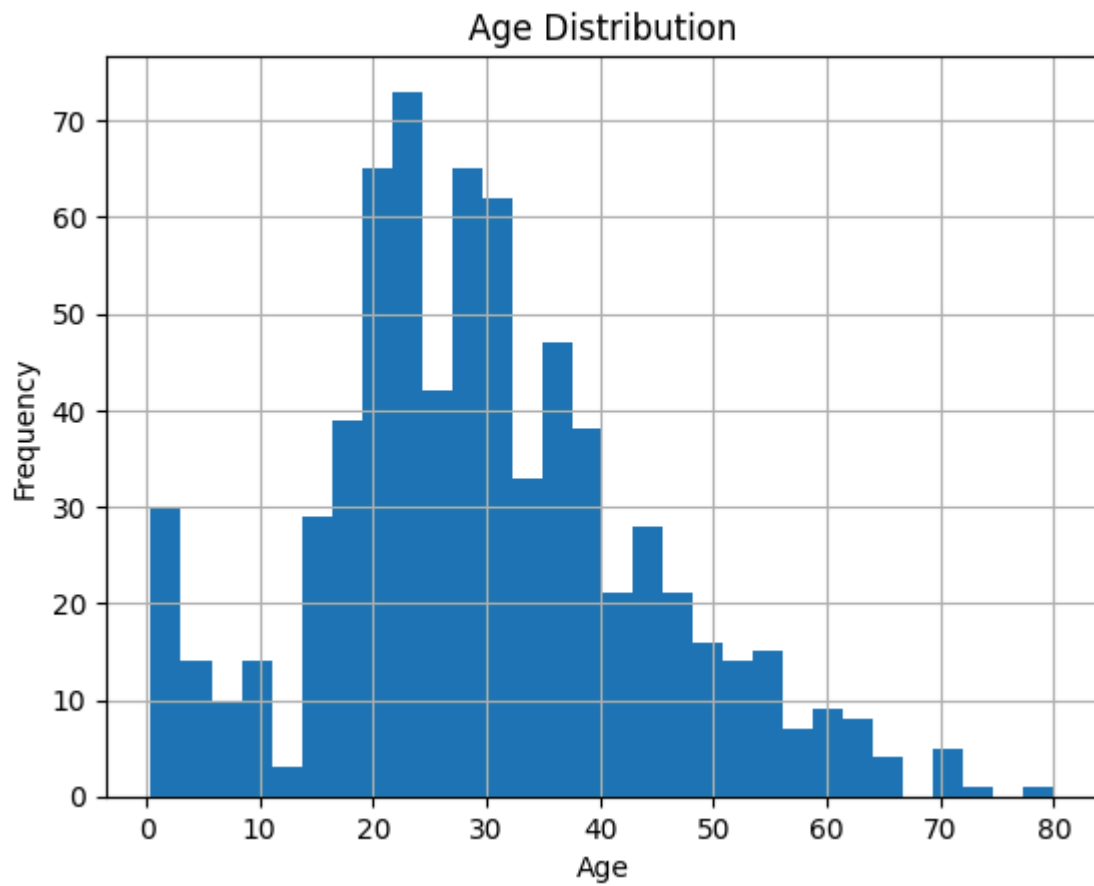
Heatmap: Strong positive correlation between Fare and Pclass (negative because lower class has higher fare).

```
corr = df.corr(numeric_only=True) # Calculate correlation only on numeric columns
sns.heatmap(corr, annot=True, cmap='coolwarm')
plt.title('Correlation Heatmap')
plt.show()
```



Histogram: Most passengers are in the 20–40 age range.

```
df['Age'].hist(bins=30)
plt.title('Age Distribution')
plt.xlabel('Age')
plt.ylabel('Frequency')
plt.show()
```



Boxplot: Outliers present in Fare; higher fares usually in first class.

```
sns.boxplot(x='Pclass', y='Age', data=df)
plt.title('Age by Passenger Class')
plt.show()
```



## Age by Passenger Class

80

```
plt.scatter(df['Age'], df['Fare'])  
plt.title('Fare vs Age')
```