In [3]:

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


data = pd.read_csv("titanic.csv")
```

In [4]:

```python
data.head()
```

Out[4]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S

In [5]:

```python
data.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]:

```
data.describe()
```

Out[6]:

	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

In [7]:

```
data.isnull()
```

Out[7]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	False	False	False	False	False	False	False	False	False	False	True	False
1	False	False	False	False	False	False	False	False	False	False	False	False
2	False	False	False	False	False	False	False	False	False	False	True	False
3	False	False	False	False	False	False	False	False	False	False	False	False
4	False	False	False	False	False	False	False	False	False	False	True	False
...	...	...	...	...	...	...	...	...	...	...	...	...
886	False	False	False	False	False	False	False	False	False	False	True	False
887	False	False	False	False	False	False	False	False	False	False	False	False
888	False	False	False	False	False	True	False	False	False	False	True	False
889	False	False	False	False	False	False	False	False	False	False	False	False
890	False	False	False	False	False	False	False	False	False	False	True	False

891 rows × 12 columns

In [8]:

```
data.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]:

```
data = data.bfill()
```

In [10]:

```
data.isnull().sum()
```

Out[10]:

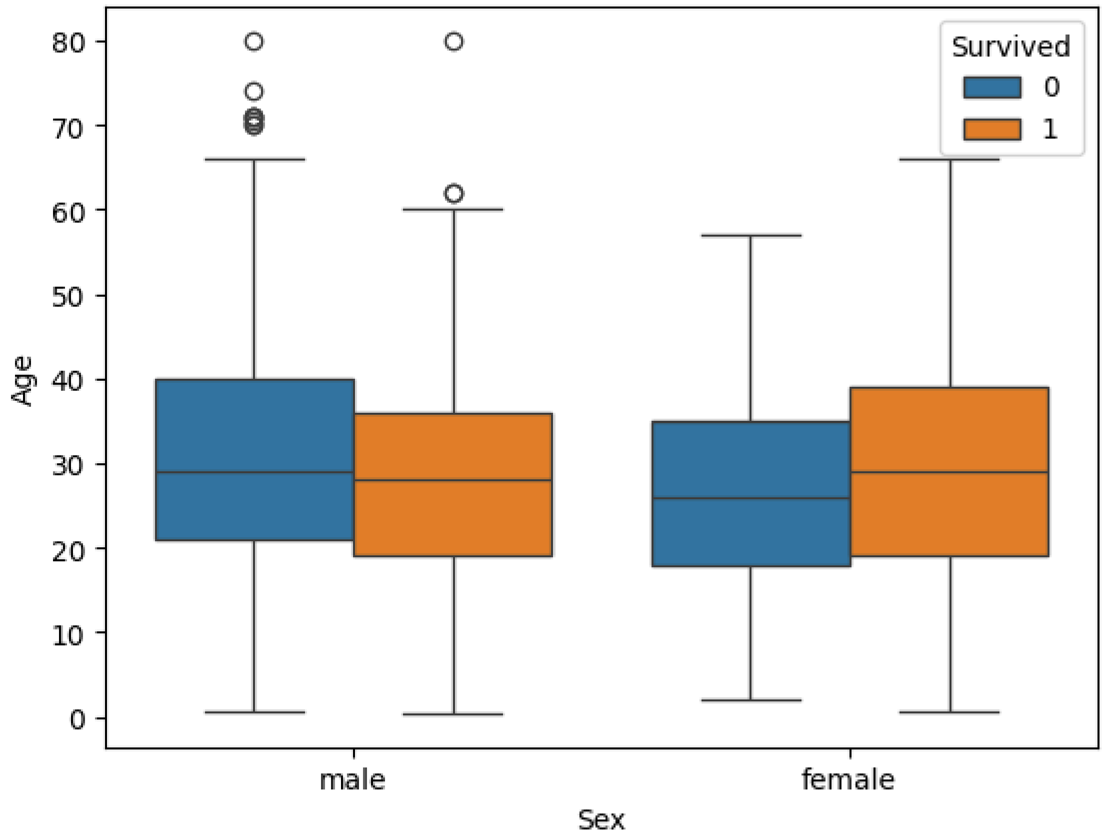
```
PassengerId    0
Survived       0
Pclass         0
Name           0
Sex            0
Age            0
SibSp          0
Parch          0
Ticket         0
Fare           0
Cabin          1
Embarked       0
dtype: int64
```

In [11]:

```
sns.boxplot(data= data, x="Sex", y="Age", hue="Survived")
```

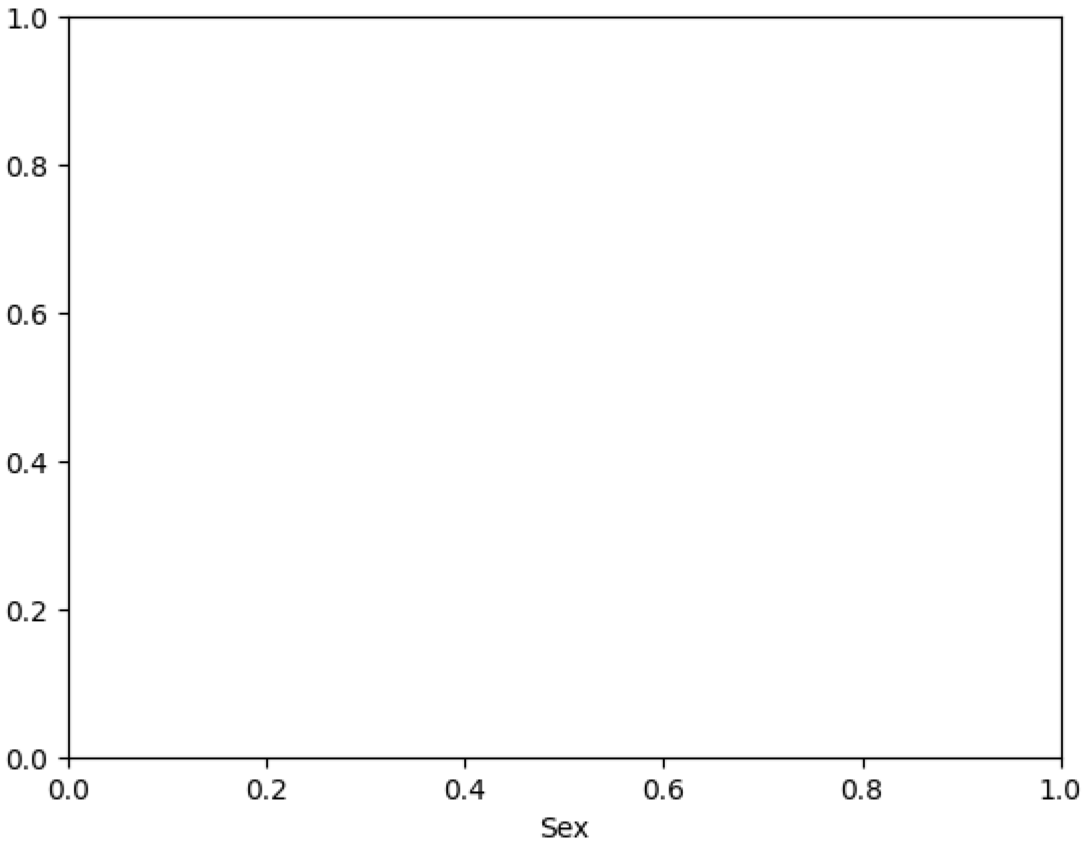
Out[11]:

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



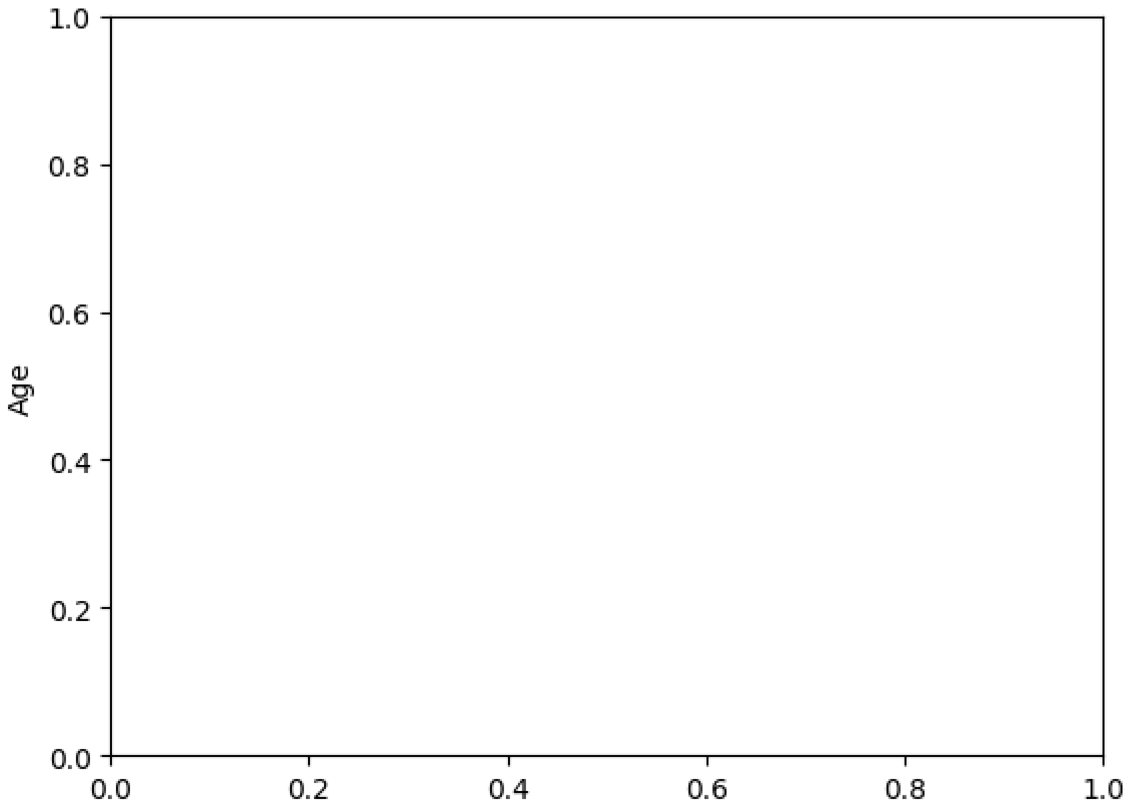
```
In [12]: plt.xlabel("Sex")
```

```
Out[12]: Text(0.5, 0, 'Sex')
```



```
In [13]: plt.ylabel("Age")
```

```
Out[13]: Text(0, 0.5, 'Age')
```



In [14]:
```
mean_age = data['Age'].mean()
print(mean_age)
```

```
29.87056116722783
```

In [15]:
```
std_age = data['Age'].std()
```

In [16]:
```
print(std_age)
```

```
14.597667657302386
```

In [17]:
```
data['zscore'] = (data['Age'] - mean_age) / std_age
```

In [18]:
```
data['zscore']
```

Out[18]:
```
0     -0.539166
1      0.556900
2     -0.265149
3      0.351388
4      0.351388
         ...   
886   -0.196645
887   -0.744678
888   -0.265149
889   -0.265149
890    0.145875
Name: zscore, Length: 891, dtype: float64
```

In [19]:
```
outliers = data[np.abs(data['zscore']) > 3]
```

In [20]:
```
outliers
```

Out[20]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked	zscore
629	630	0	3	O'Connell, Mr. Patrick D	male	80.0	0	0	334912	7.7333	A23	Q	3.434072
630	631	1	1	Barkworth, Mr. Algernon Henry Wilson	male	80.0	0	0	27042	30.0000	A23	S	3.434072
851	852	0	3	Svensson, Mr. Johan	male	74.0	0	0	347060	7.7750	D28	S	3.023047

In [21]:
```
print(outliers[['Age', 'Sex', 'Survived', 'zscore']])
```

```
      Age   Sex  Survived    zscore
629  80.0  male         0  3.434072
630  80.0  male         1  3.434072
851  74.0  male         0  3.023047
```

In [22]:
```
titanic_cleaned = data[np.abs(data['zscore']) <= 3]
```

In [24]:
```
titanic_cleaned = titanic_cleaned.drop(columns=['zscore'])
```

In [25]:

```
print("Original dataset size:", data.shape[0])
print("Cleaned dataset size:", titanic_cleaned.shape[0])
```

```
Original dataset size: 891
Cleaned dataset size: 888
```

In [26]:

```
titanic_cleaned.head()
```

Out[26]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	C85	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	C123	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	E46	S

In [ ]: