Assignment No. 8

Aim : Data Visualization I

1. Use the inbuilt dataset 'titanic'. The dataset contains 891 rows and contains information about the passengers who boarded the unfortunate Titanic ship. Use the Seaborn library to see if we can find any patterns in the data.
2. Write a code to check how the price of the ticket (column name: 'fare') for each passenger is distributed by plotting a histogram.

Code :

In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
df1 = sns.load_dataset('titanic')
df1
```

Out[1]:

	survived	pclass	sex	age	sibsp	parch	fare	embarked	class	who	adult_mal
0	0	3	male	22.0	1	0	7.2500	S	Third	man	Tru
1	1	1	female	38.0	1	0	71.2833	C	First	woman	Fals
2	1	3	female	26.0	0	0	7.9250	S	Third	woman	Fals
3	1	1	female	35.0	1	0	53.1000	S	First	woman	Fals
4	0	3	male	35.0	0	0	8.0500	S	Third	man	Tru
...	...	...	...	...	...	...	...	...	...	...	.
886	0	2	male	27.0	0	0	13.0000	S	Second	man	Tru
887	1	1	female	19.0	0	0	30.0000	S	First	woman	Fals
888	0	3	female	NaN	1	2	23.4500	S	Third	woman	Fals
889	1	1	male	26.0	0	0	30.0000	C	First	man	Tru
890	0	3	male	32.0	0	0	7.7500	Q	Third	man	Tru

891 rows × 15 columns

In [2]:

```
df1.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 15 columns):
 #   Column       Non-Null Count  Dtype   
---  ------       --------------  -----   
 0   survived     891 non-null    int64   
 1   pclass       891 non-null    int64   
 2   sex          891 non-null    object  
 3   age          714 non-null    float64 
 4   sibsp        891 non-null    int64   
 5   parch        891 non-null    int64   
 6   fare         891 non-null    float64 
 7   embarked     889 non-null    object  
 8   class        891 non-null    category
 9   who          891 non-null    object  
 10  adult_male   891 non-null    bool    
 11  deck         203 non-null    category
 12  embark_town  889 non-null    object  
 13  alive        891 non-null    object  
 14  alone        891 non-null    bool    
dtypes: bool(2), category(2), float64(2), int64(4), object(5)
memory usage: 80.7+ KB
```

In [3]:

```
df1.shape
```

Out[3]:

```
(891, 15)
```

In [4]:

```
df1.describe()
```

Out[4]:

	survived	pclass	age	sibsp	parch	fare
count	891.000000	891.000000	714.000000	891.000000	891.000000	891.000000
mean	0.383838	2.308642	29.699118	0.523008	0.381594	32.204208
std	0.486592	0.836071	14.526497	1.102743	0.806057	49.693429
min	0.000000	1.000000	0.420000	0.000000	0.000000	0.000000
25%	0.000000	2.000000	20.125000	0.000000	0.000000	7.910400
50%	0.000000	3.000000	28.000000	0.000000	0.000000	14.454200
75%	1.000000	3.000000	38.000000	1.000000	0.000000	31.000000
max	1.000000	3.000000	80.000000	8.000000	6.000000	512.329200

In [5]:
```
df1.head()
```

Out[5]:

	survived	pclass	sex	age	sibsp	parch	fare	embarked	class	who	adult_male	d
0	0	3	male	22.0	1	0	7.2500	S	Third	man	True	N
1	1	1	female	38.0	1	0	71.2833	C	First	woman	False	
2	1	3	female	26.0	0	0	7.9250	S	Third	woman	False	N
3	1	1	female	35.0	1	0	53.1000	S	First	woman	False	
4	0	3	male	35.0	0	0	8.0500	S	Third	man	True	N

In [6]:
```
df1.tail()
```

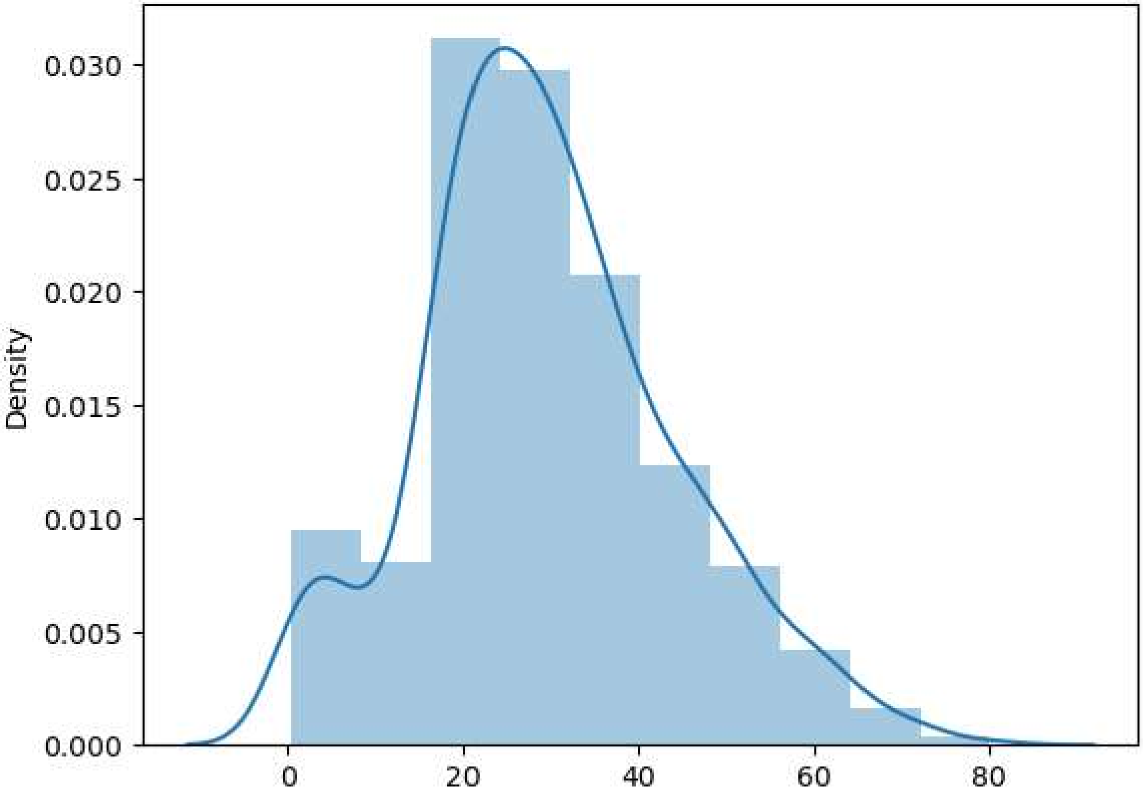
Out[6]:

	survived	pclass	sex	age	sibsp	parch	fare	embarked	class	who	adult_male
886	0	2	male	27.0	0	0	13.00	S	Second	man	True
887	1	1	female	19.0	0	0	30.00	S	First	woman	False
888	0	3	female	NaN	1	2	23.45	S	Third	woman	False
889	1	1	male	26.0	0	0	30.00	C	First	man	True
890	0	3	male	32.0	0	0	7.75	Q	Third	man	True

In [7]:
```
sns.distplot(x = df1['age'], bins = 10)
```

```
C:\Users\Admin\AppData\Local\Temp\ipykernel_17360\1321188604.py:1: UserWarnin
g:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751 (https://gis
t.github.com/mwaskom/de44147ed2974457ad6372750bbe5751)

  sns.distplot(x = df1['age'], bins = 10)
```

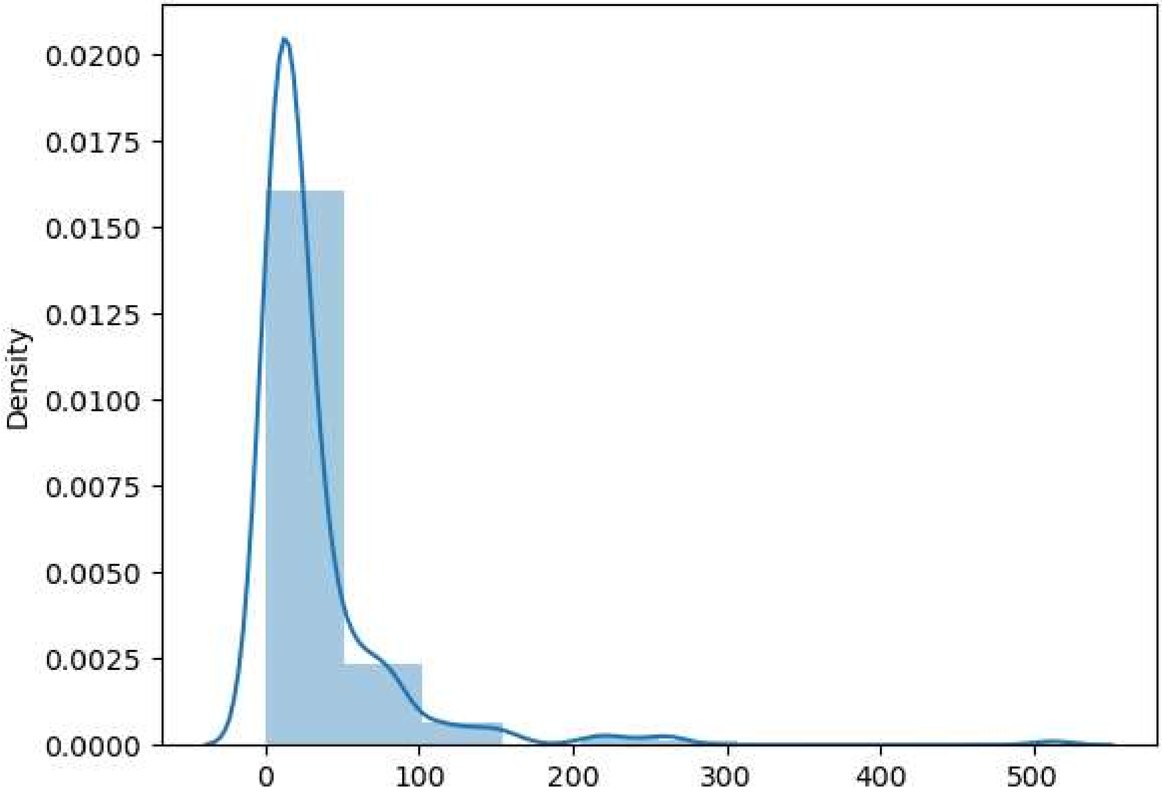
Out[7]:
```
<Axes: ylabel='Density'>
```



In [8]:
```
sns.distplot(x = df1['fare'], bins = 10)
```

```
C:\Users\Admin\AppData\Local\Temp\ipykernel_17360\1056606941.py:1: UserWarnin
g:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751 (https://gis
t.github.com/mwaskom/de44147ed2974457ad6372750bbe5751)

  sns.distplot(x = df1['fare'], bins = 10)
```

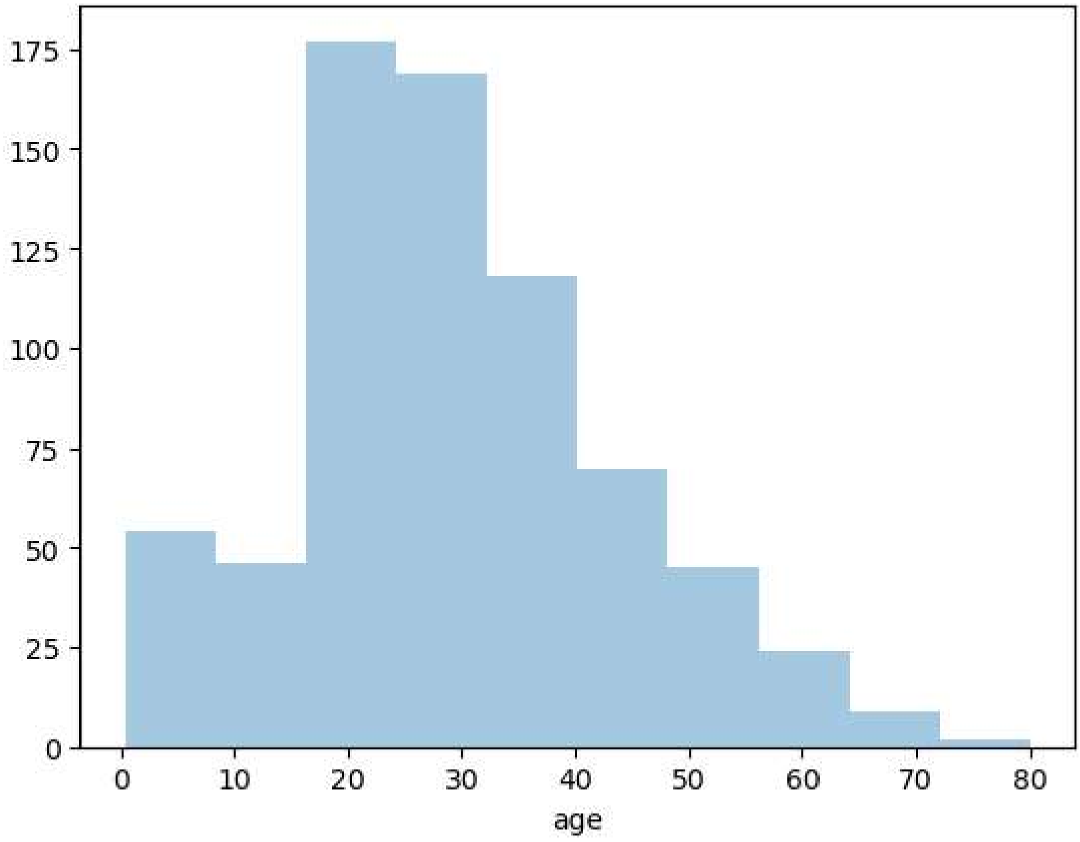
Out[8]:
```
<Axes: ylabel='Density'>
```



In [9]:
```
sns.distplot(df1['age'], bins = 10,kde=False)
```

```
C:\Users\Admin\AppData\Local\Temp\ipykernel_17360\3002249961.py:1: UserWarnin
g:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751 (https://gis
t.github.com/mwaskom/de44147ed2974457ad6372750bbe5751)

  sns.distplot(df1['age'], bins = 10,kde=False)
```

Out[9]:
```
<Axes: xlabel='age'>
```



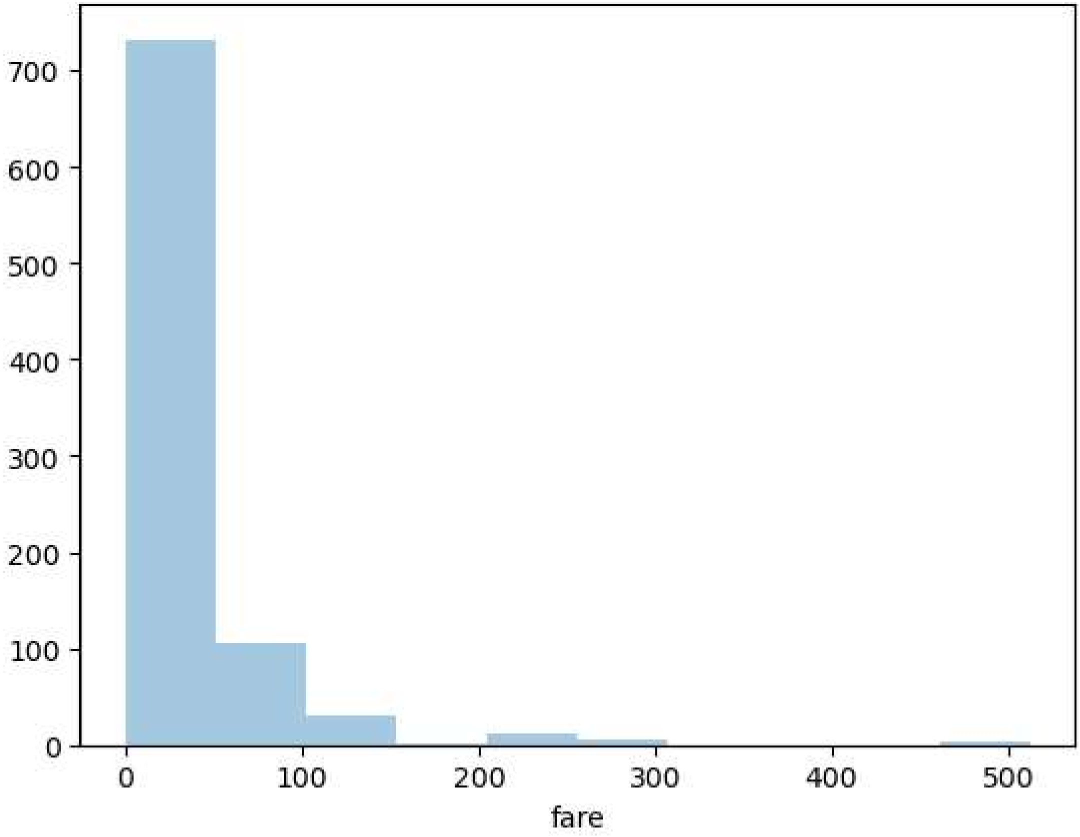
In [10]:

```
sns.distplot(df1['fare'], bins = 10,kde=False)
```

```
C:\Users\Admin\AppData\Local\Temp\ipykernel_17360\2862302602.py:1: UserWarnin
g:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751 (https://gis
t.github.com/mwaskom/de44147ed2974457ad6372750bbe5751)

  sns.distplot(df1['fare'], bins = 10,kde=False)
```

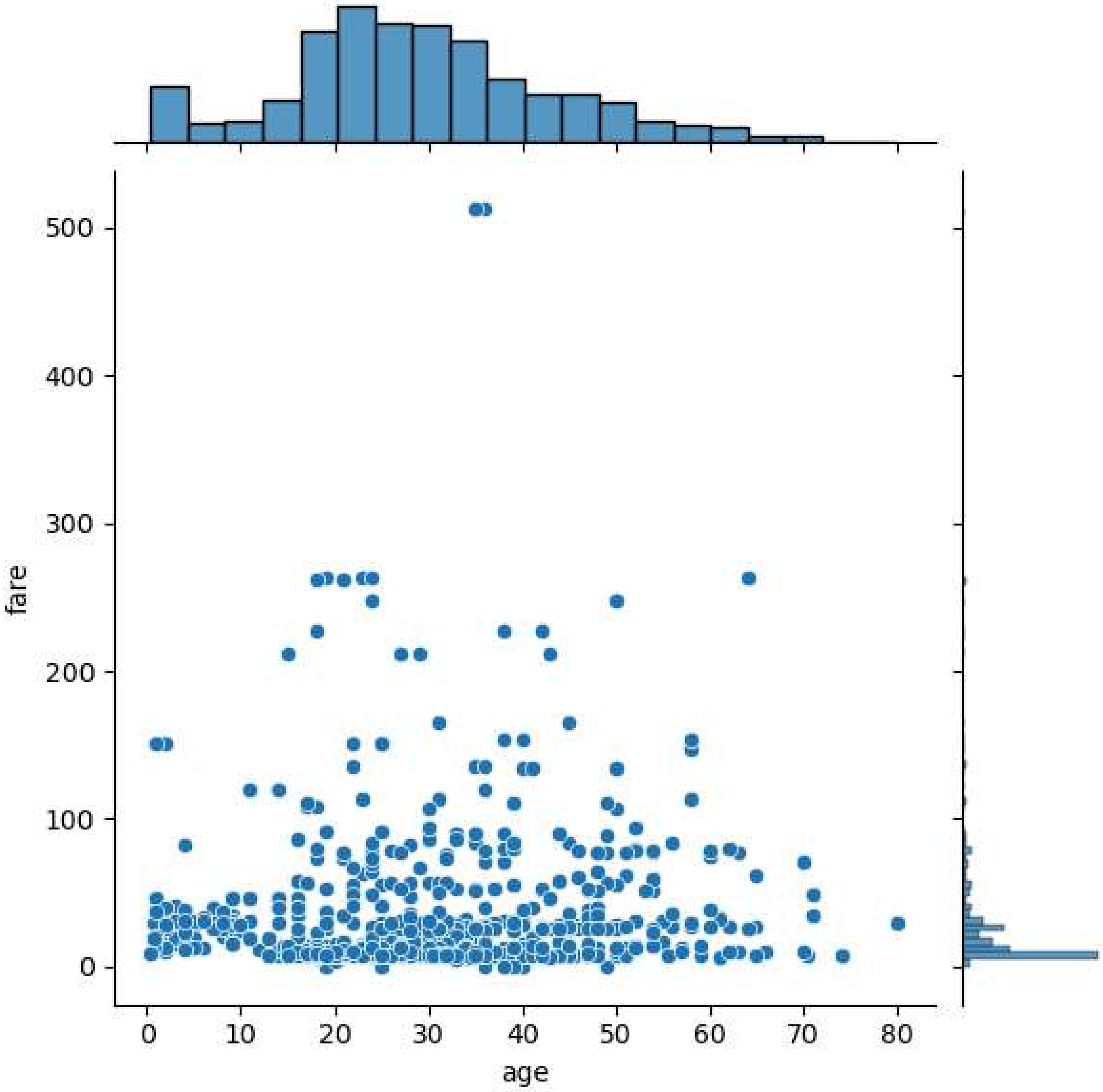
Out[10]:

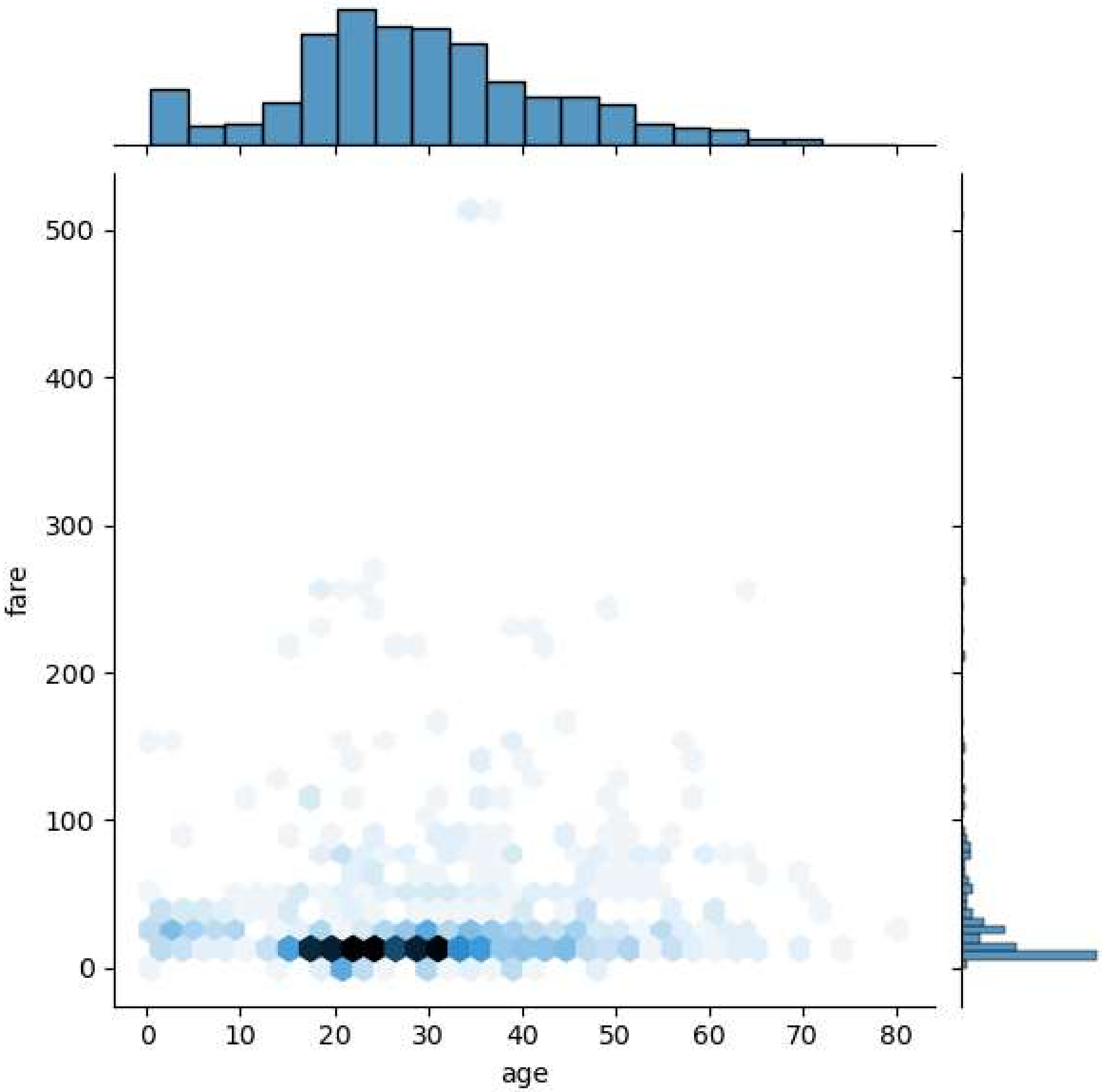
```
<Axes: xlabel='fare'>
```



```
In [11]: sns.jointplot(x = df1['age'], y = df1['fare'], kind ='scatter')
         sns.jointplot(x = df1['age'], y = df1['fare'], kind = 'hex')
```

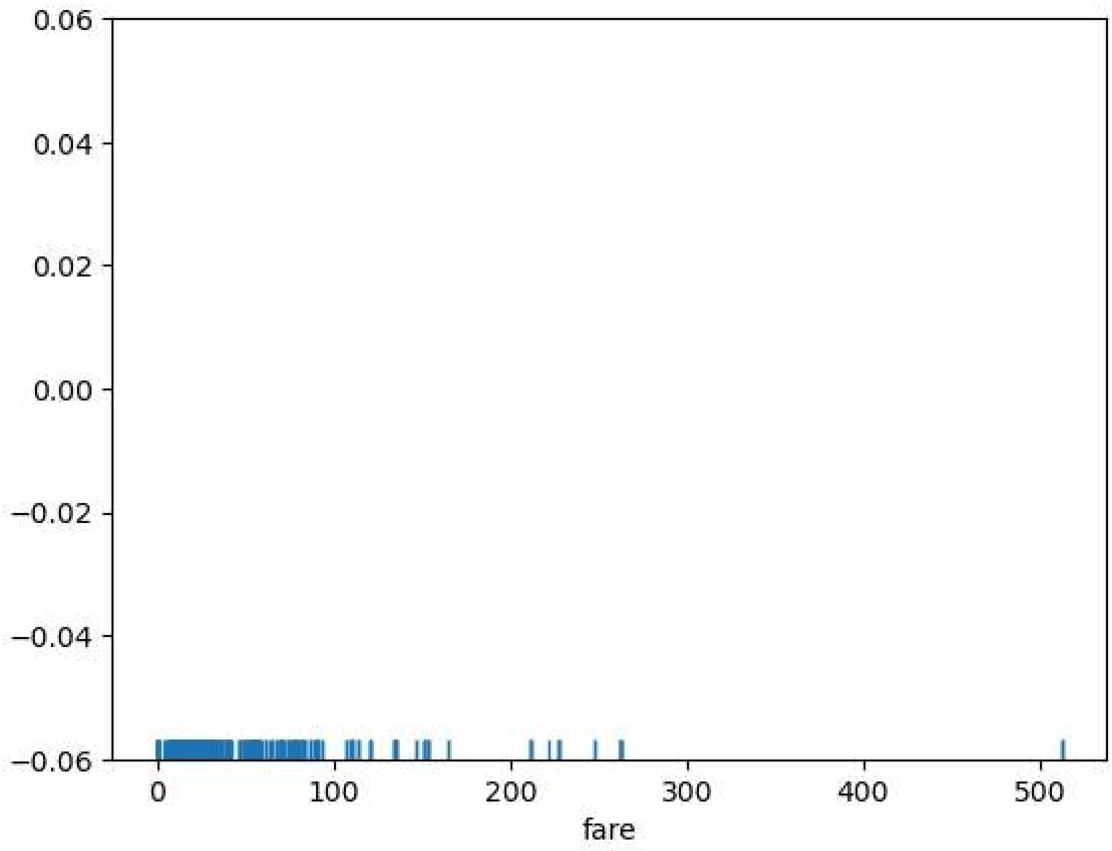
```
Out[11]: <seaborn.axisgrid.JointGrid at 0x2a2184b8690>
```




500
400
300
200
100
0
fare
0
10
20
30
40
50
60
70
80
age

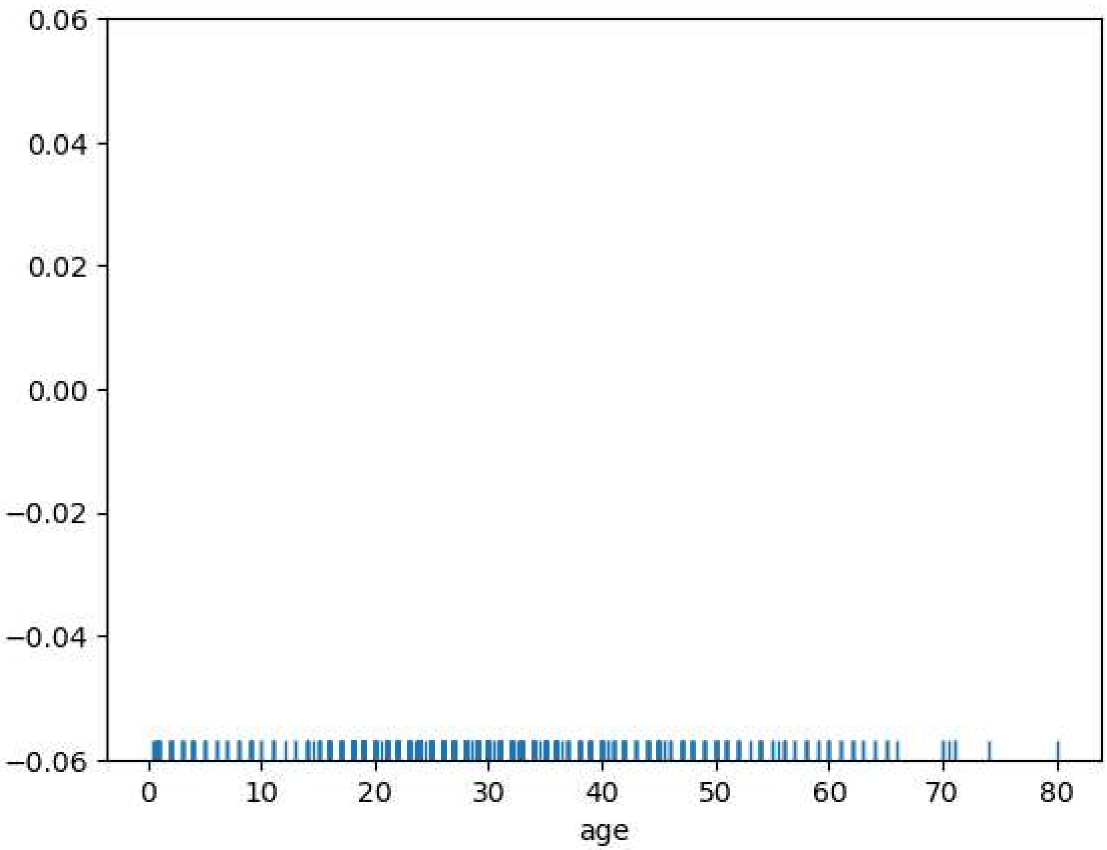
In [12]:
```
sns.rugplot(df1['fare'])
```

Out[12]:
```
<Axes: xlabel='fare'>
```



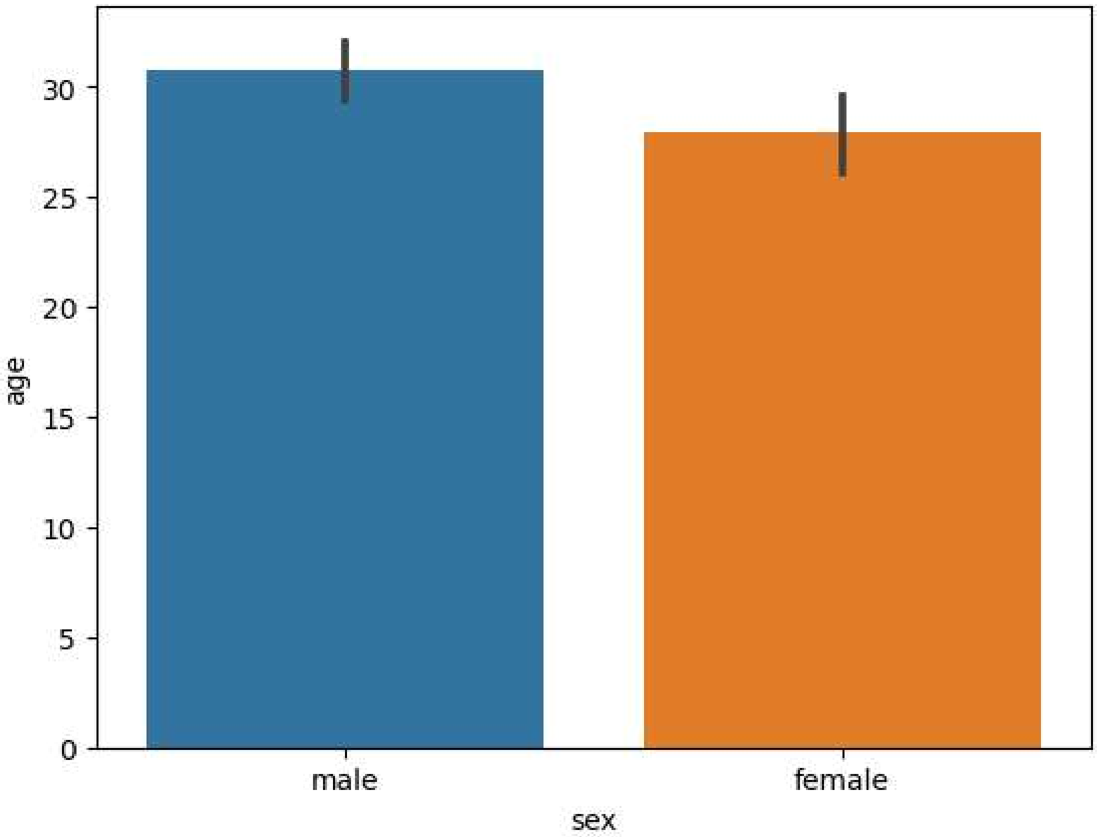
```
In [13]: sns.rugplot(df1['age'])
```

```
Out[13]: <Axes: xlabel='age'>
```



In [14]:
```
sns.barplot(x='sex', y='age', data=df1)
```

Out[14]:
```
<Axes: xlabel='sex', ylabel='age'>
```



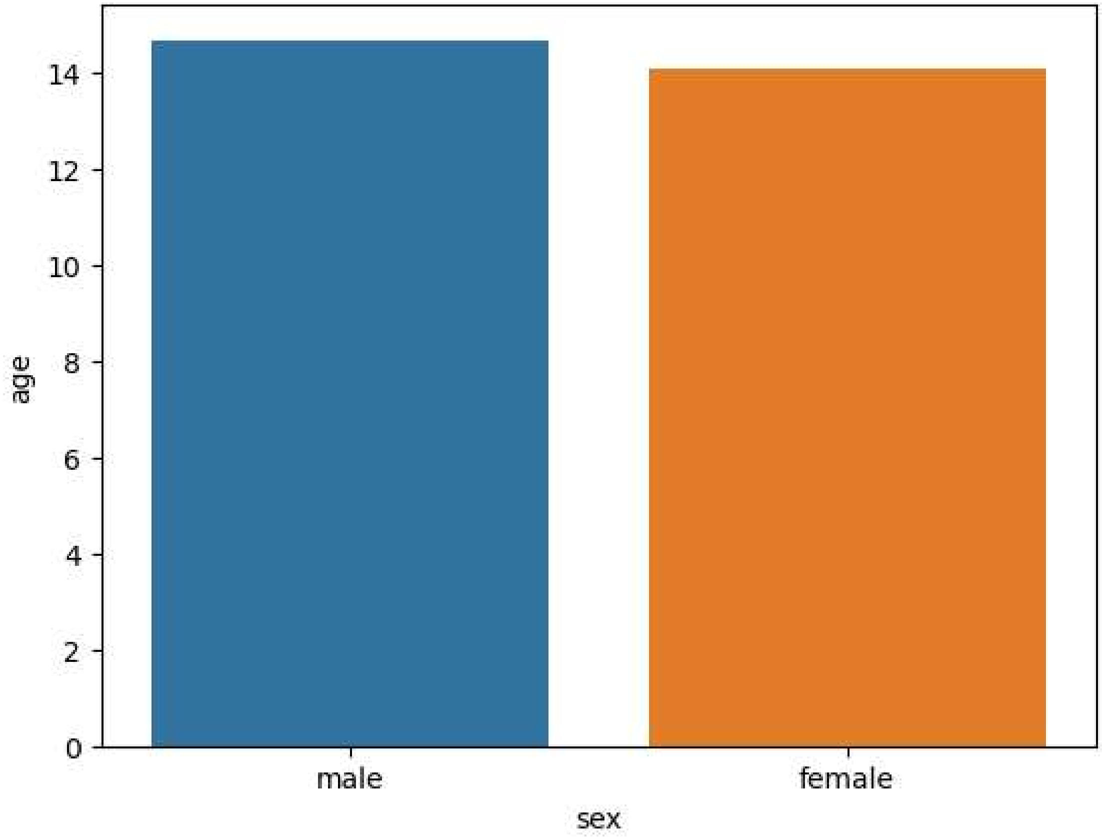
In [15]:

```
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
sns.barplot(x='sex', y='age', data=df1, estimator=np.std)
```

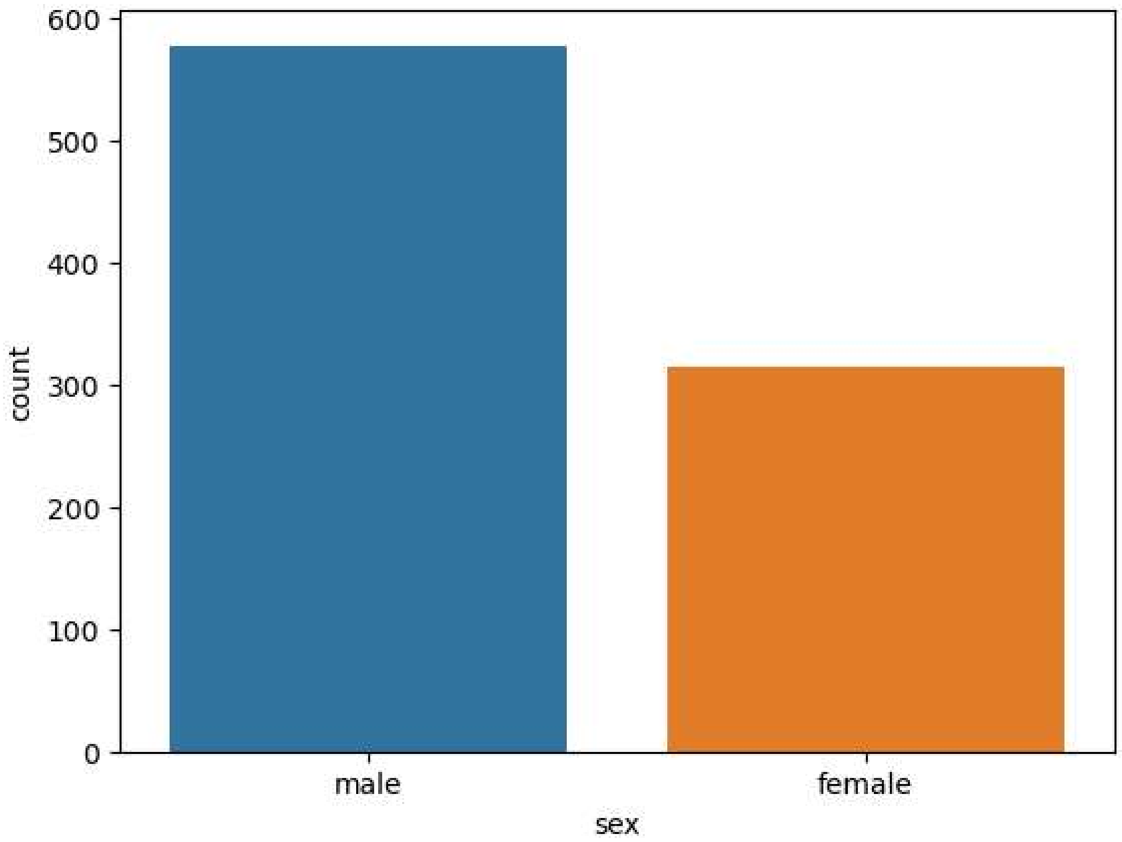
```
C:\ProgramData\anaconda3\Lib\site-packages\numpy\lib\nanfunctions.py:1556: Ru
ntimeWarning: All-NaN slice encountered
  return function_base._ureduce(a,
```

Out[15]:

```
<Axes: xlabel='sex', ylabel='age'>
```

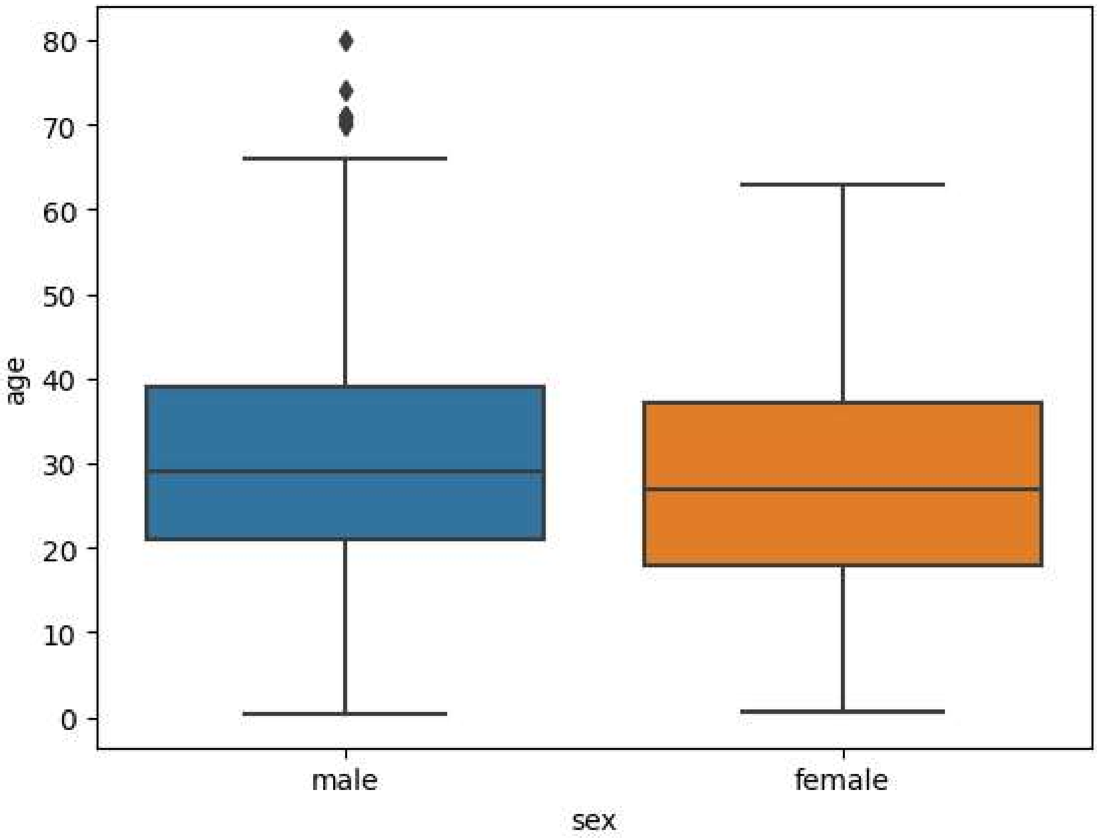


```
In [16]: sns.countplot(x='sex', data=df1)

Out[16]: <Axes: xlabel='sex', ylabel='count'>
```



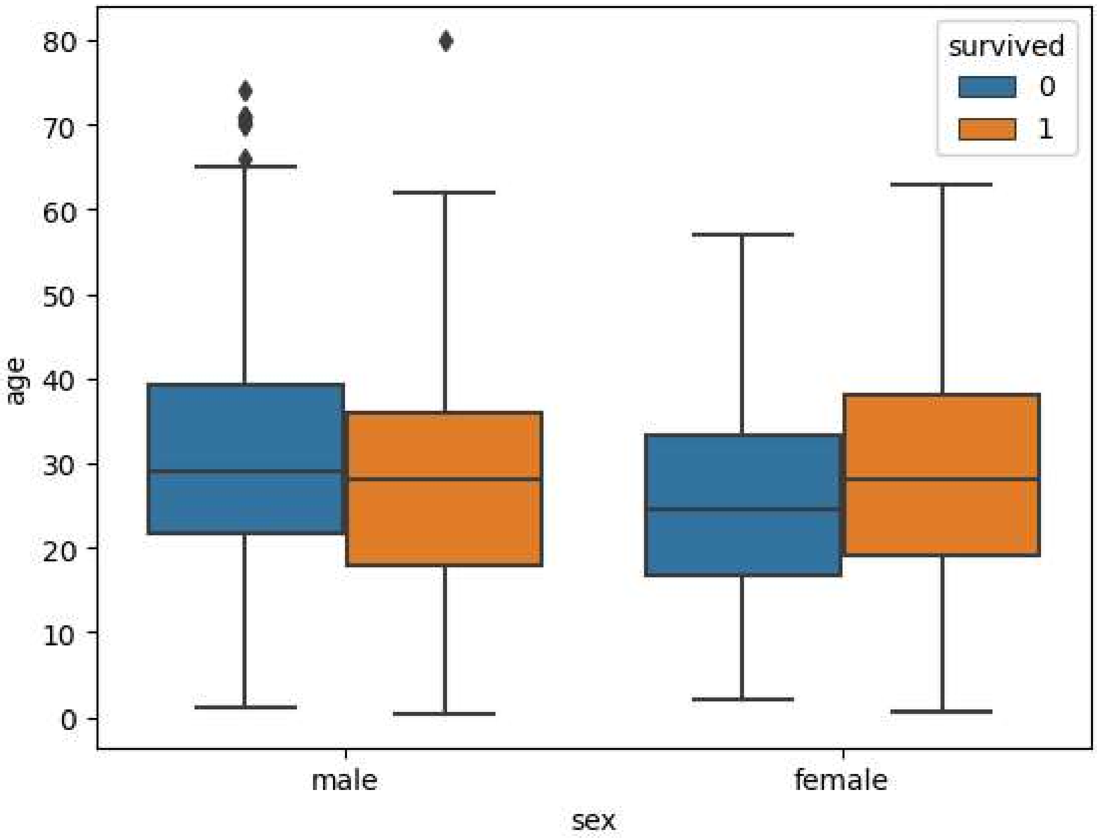
```
In [17]: sns.boxplot(x='sex', y='age', data=df1)
```

```
Out[17]: <Axes: xlabel='sex', ylabel='age'>
```



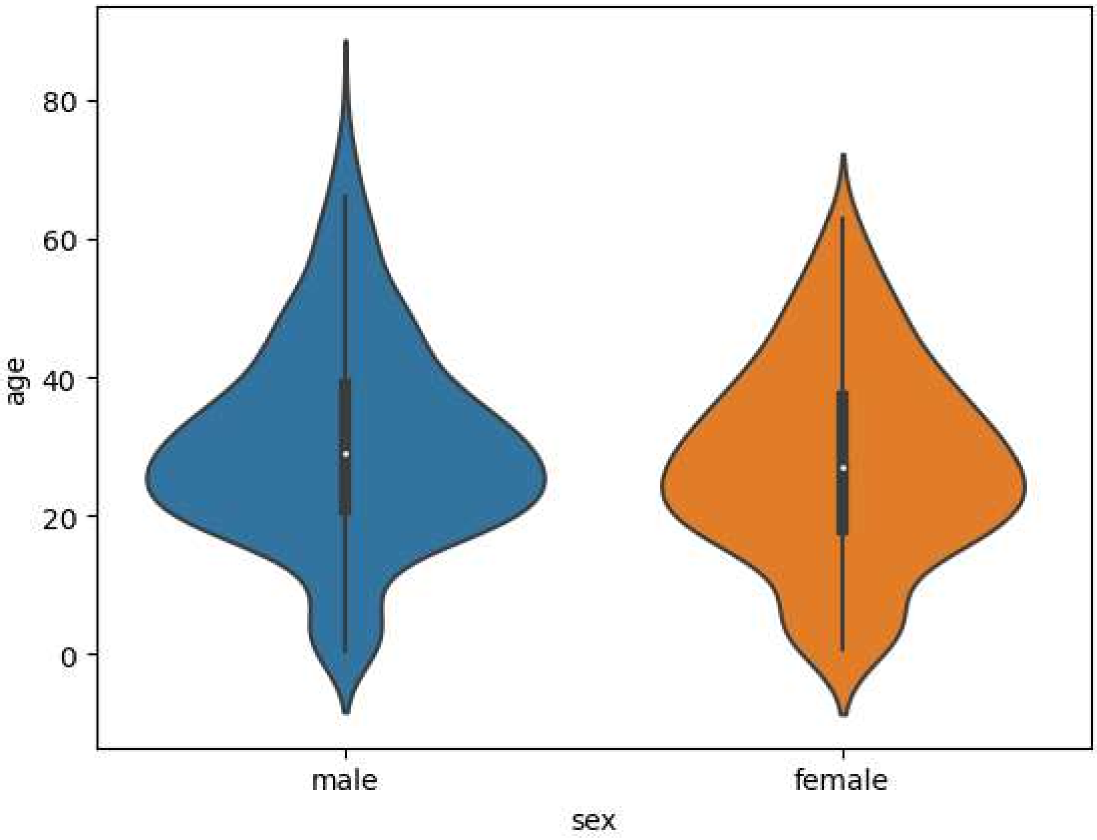
```
In [18]: sns.boxplot(x='sex', y='age', data=df1, hue="survived")
```

```
Out[18]: <Axes: xlabel='sex', ylabel='age'>
```



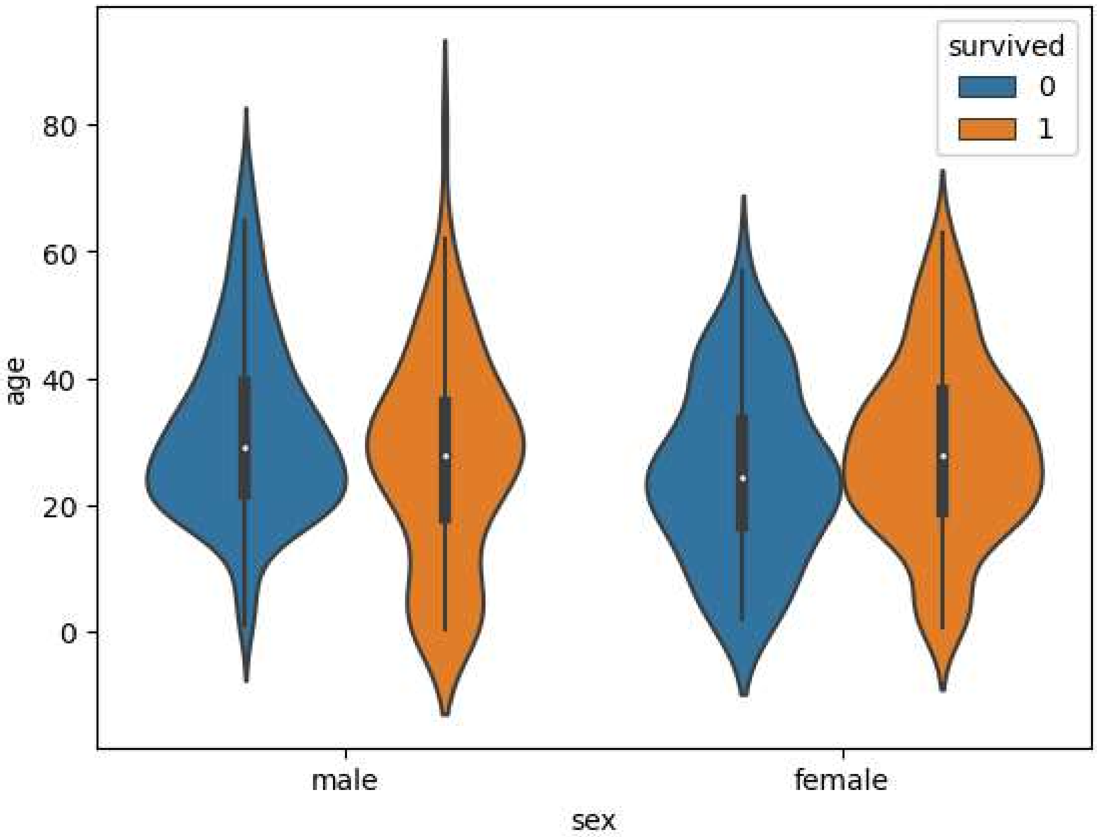
```
In [19]: sns.violinplot(x='sex', y='age', data=df1)
```

```
Out[19]: <Axes: xlabel='sex', ylabel='age'>
```



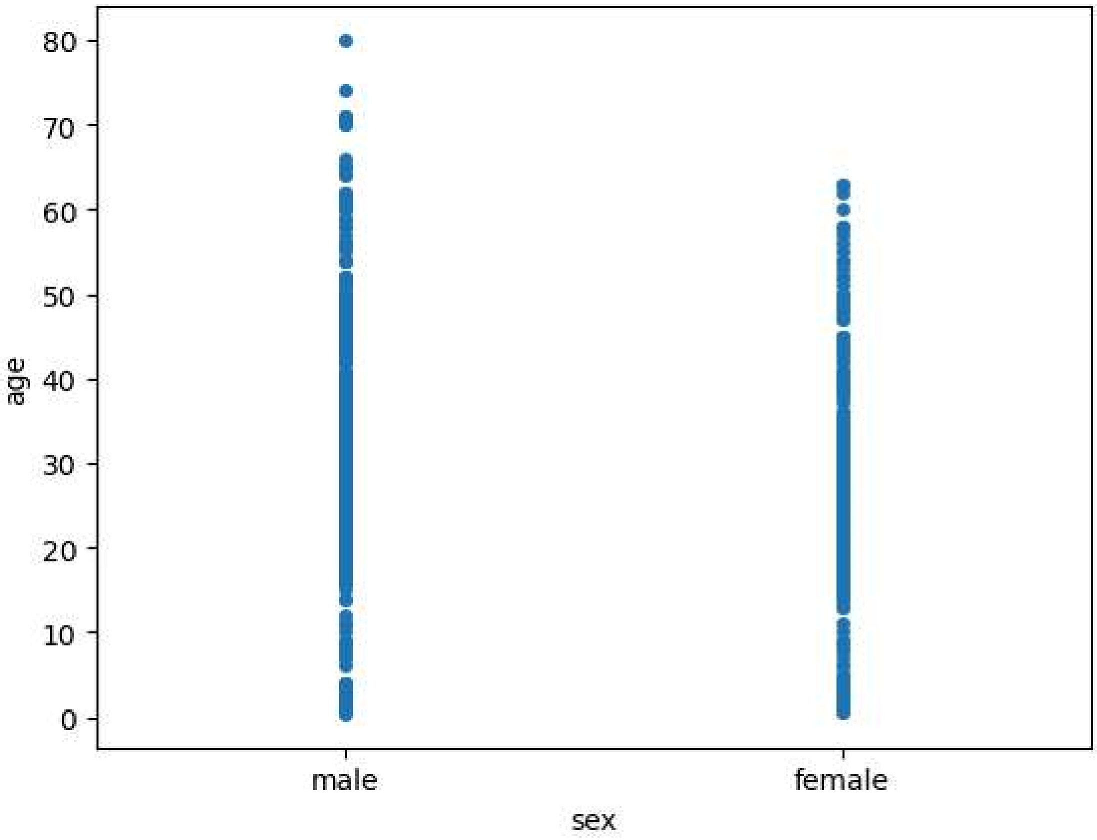
```
In [20]: sns.violinplot(x='sex', y='age', data=df1, hue='survived')
```

```
Out[20]: <Axes: xlabel='sex', ylabel='age'>
```



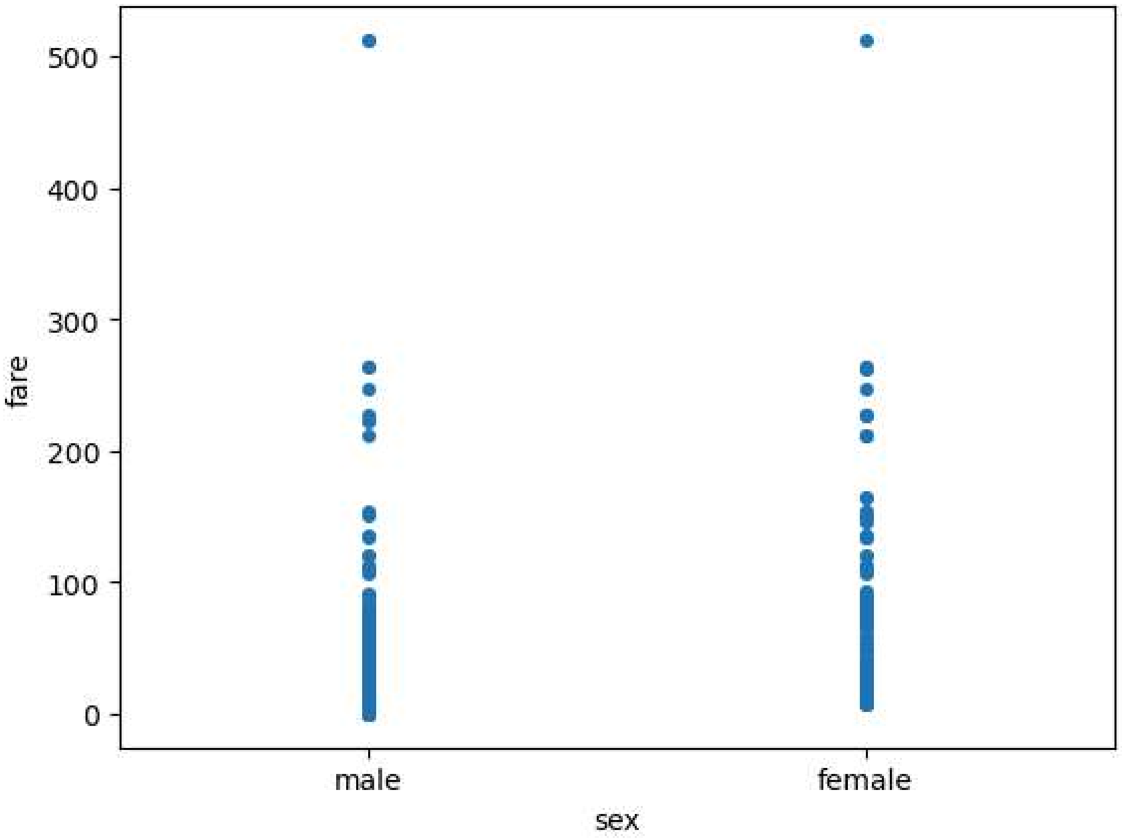
In [21]:
```
sns.stripplot(x='sex', y='age', data=df1, jitter=False)
```

Out[21]:
```
<Axes: xlabel='sex', ylabel='age'>
```



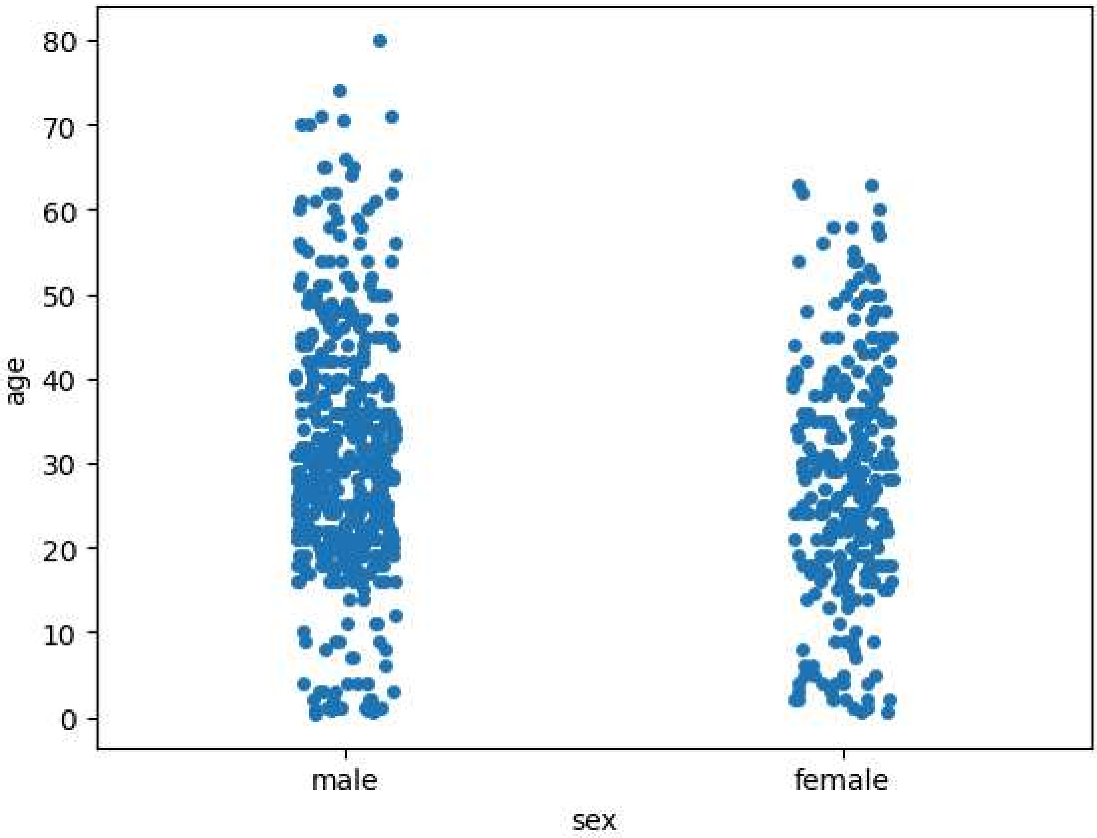
In [22]:
```
sns.stripplot(x='sex', y='fare', data=df1, jitter=False)
```

Out[22]:
```
<Axes: xlabel='sex', ylabel='fare'>
```

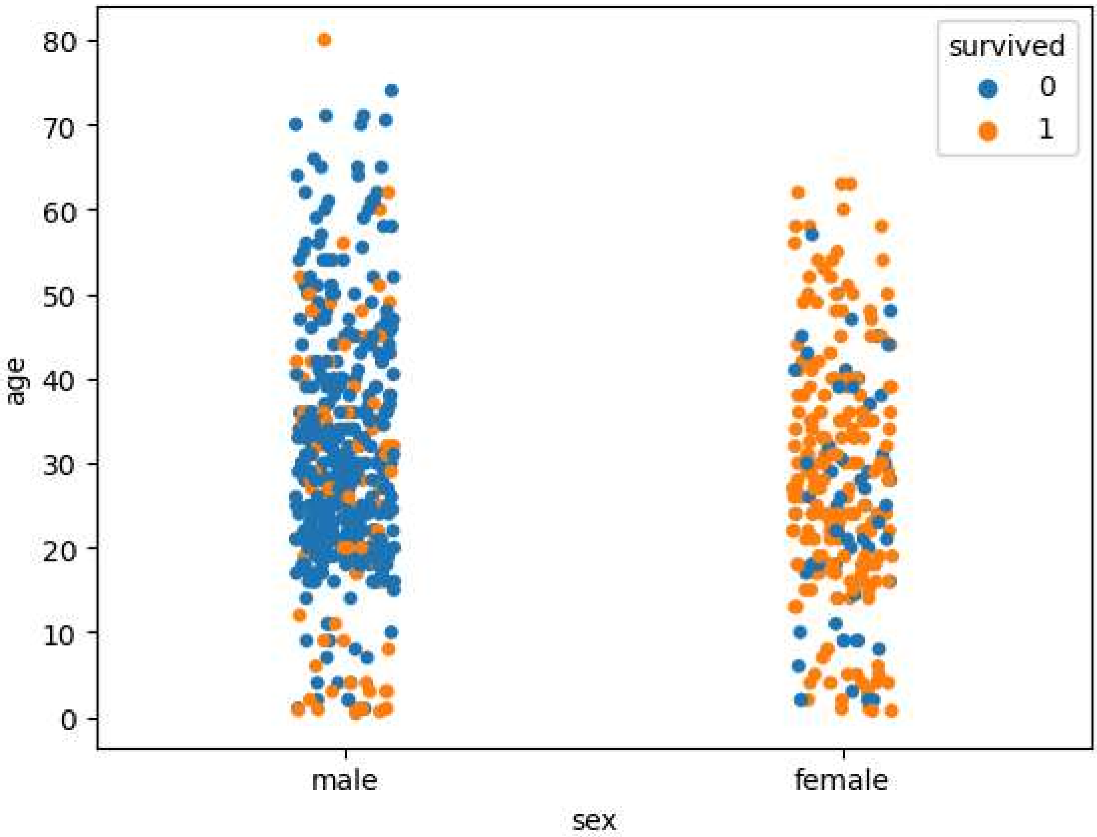


```
In [23]: sns.stripplot(x='sex', y='age', data=df1, jitter=True)
```

```
Out[23]: <Axes: xlabel='sex', ylabel='age'>
```

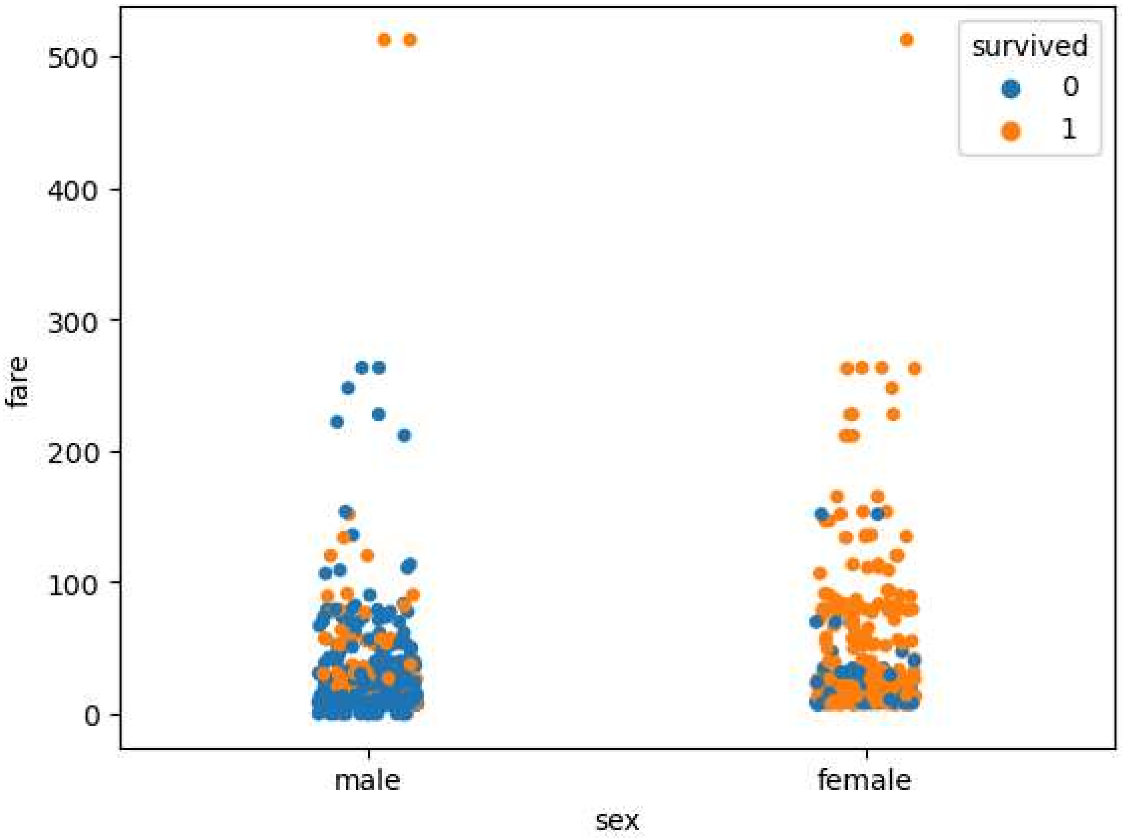


```
In [24]: sns.stripplot(x='sex', y='age', data=df1, jitter=True, hue='survived')

Out[24]: <Axes: xlabel='sex', ylabel='age'>
```



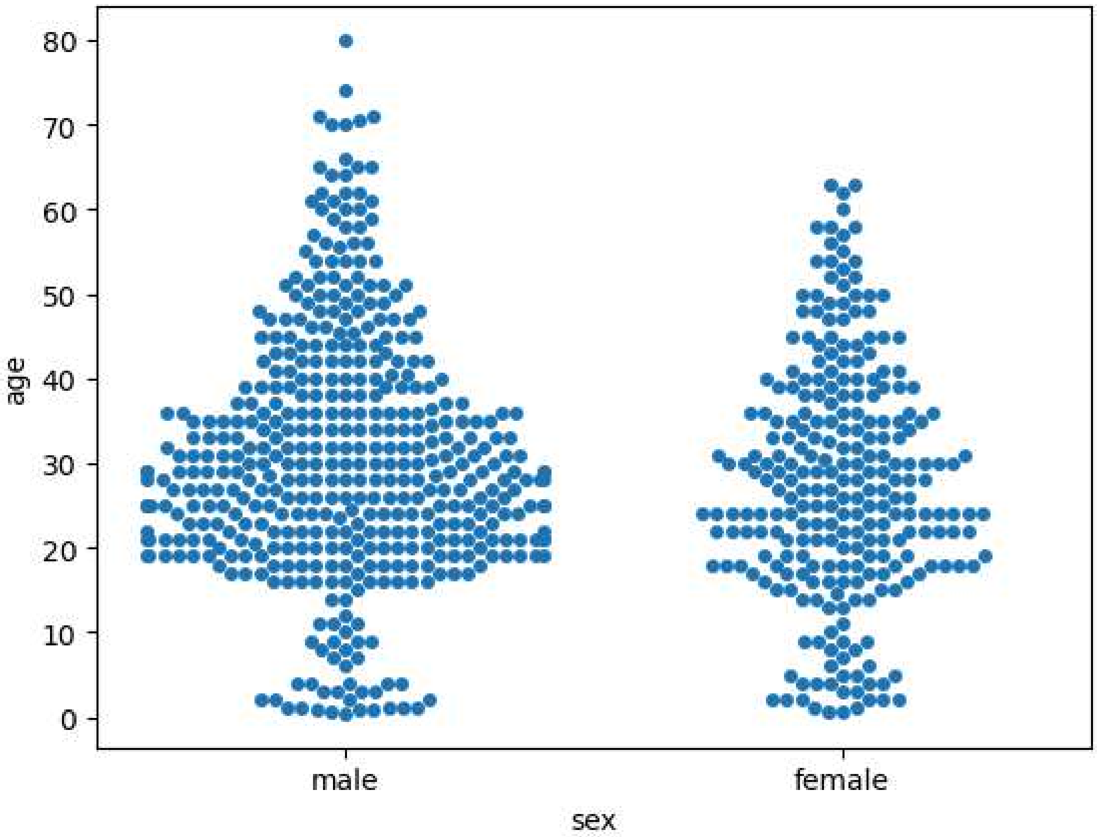
```
In [25]: sns.stripplot(x='sex', y='fare', data=df1, jitter=True, hue='survived')
```

```
Out[25]: <Axes: xlabel='sex', ylabel='fare'>
```



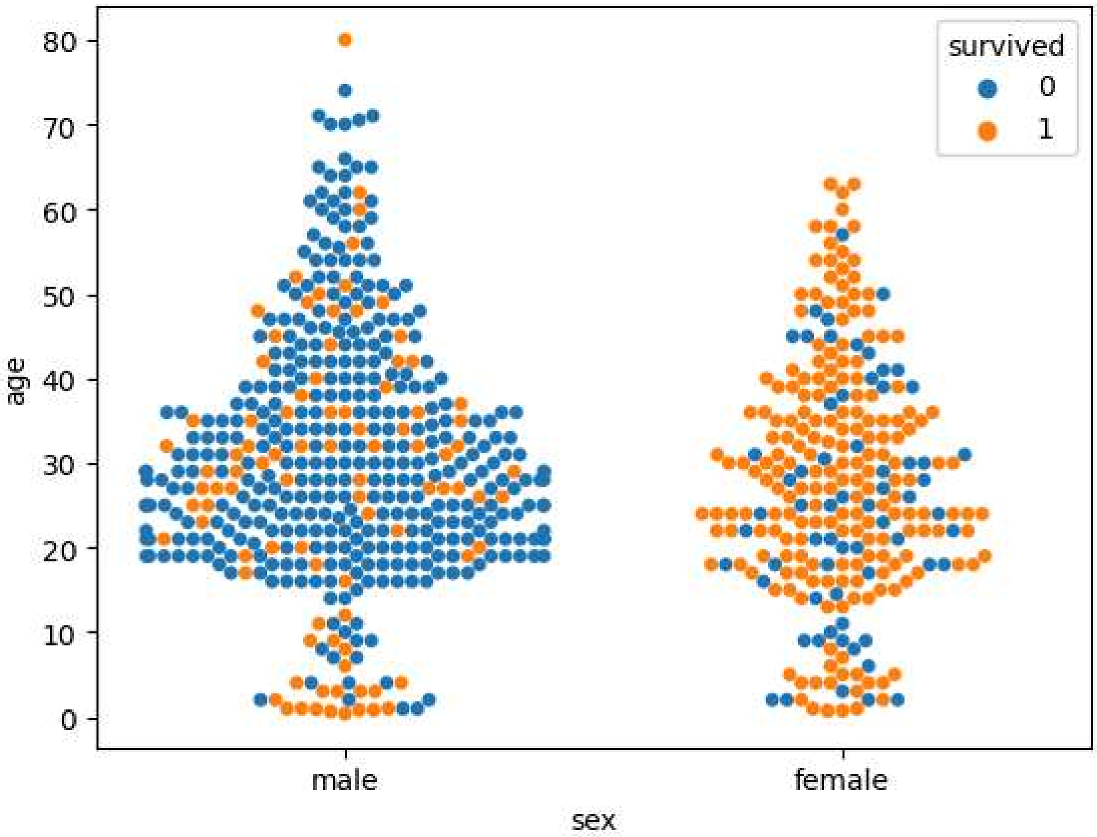
In [26]:
```
sns.swarmplot(x='sex', y='age', data=df1)
```

Out[26]:
```
<Axes: xlabel='sex', ylabel='age'>
```



In [27]:
```
sns.swarmplot(x='sex', y='age', data=df1, hue='survived')
```

Out[27]:
```
<Axes: xlabel='sex', ylabel='age'>
```



In [28]:
```
df1.head()
```

Out[28]:

	survived	pclass	sex	age	sibsp	parch	fare	embarked	class	who	adult_male	d
0	0	3	male	22.0	1	0	7.2500	S	Third	man	True	N
1	1	1	female	38.0	1	0	71.2833	C	First	woman	False	
2	1	3	female	26.0	0	0	7.9250	S	Third	woman	False	N
3	1	1	female	35.0	1	0	53.1000	S	First	woman	False	
4	0	3	male	35.0	0	0	8.0500	S	Third	man	True	N

In [29]:
```
numerical_df = df1.select_dtypes(include=['number'])
```

In [33]:

```
corr = numerical_df.corr()
corr
```

Out[33]:

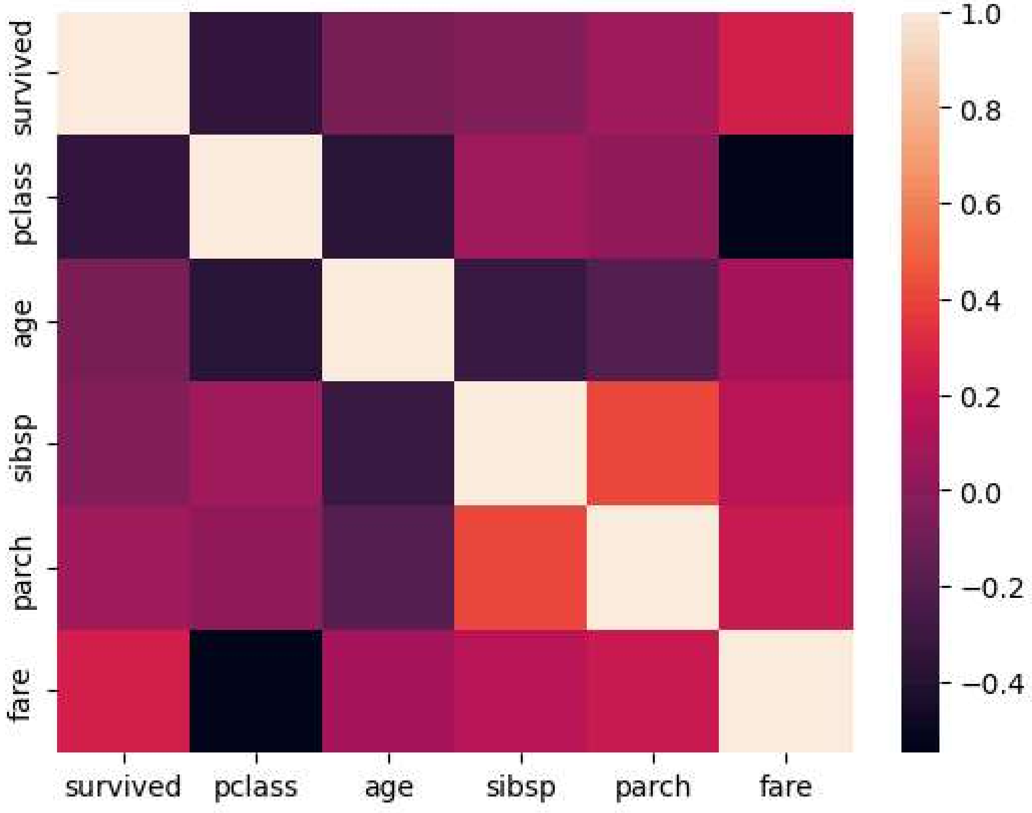
	survived	pclass	age	sibsp	parch	fare
survived	1.000000	-0.338481	-0.077221	-0.035322	0.081629	0.257307
pclass	-0.338481	1.000000	-0.369226	0.083081	0.018443	-0.549500
age	-0.077221	-0.369226	1.000000	-0.308247	-0.189119	0.096067
sibsp	-0.035322	0.083081	-0.308247	1.000000	0.414838	0.159651
parch	0.081629	0.018443	-0.189119	0.414838	1.000000	0.216225
fare	0.257307	-0.549500	0.096067	0.159651	0.216225	1.000000

In [34]:

```
sns.heatmap(corr)
```

Out[34]:

```
<Axes: >
```

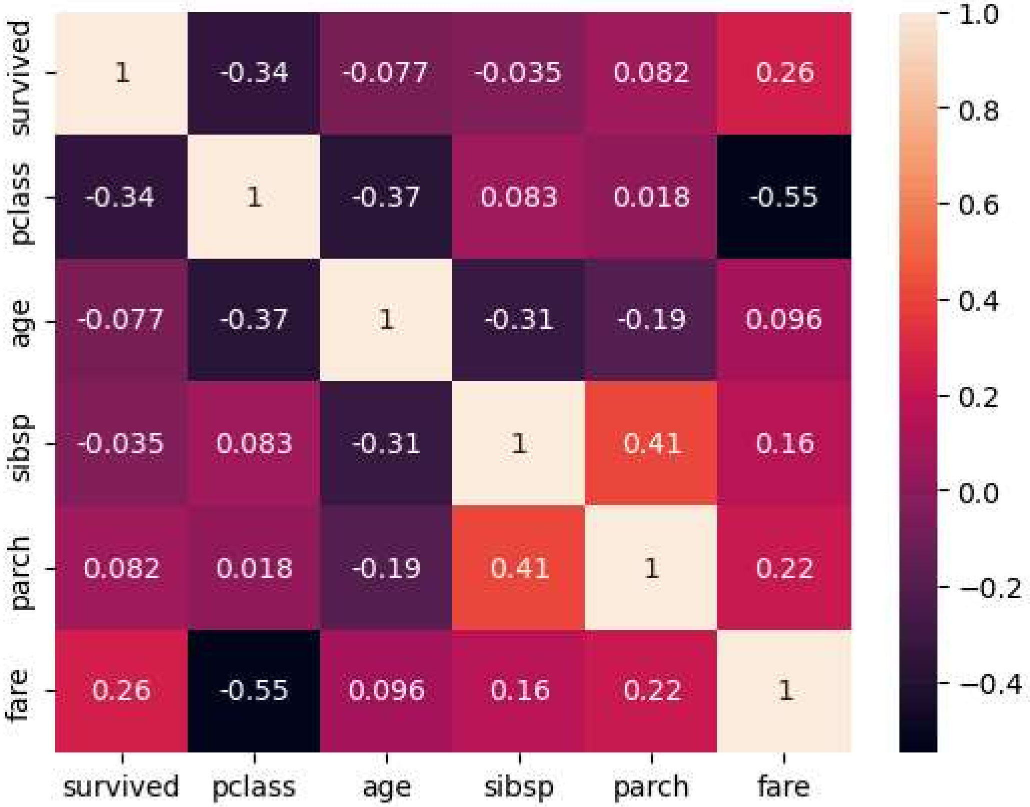


In [35]:

```
sns.heatmap(corr, annot=True)
```

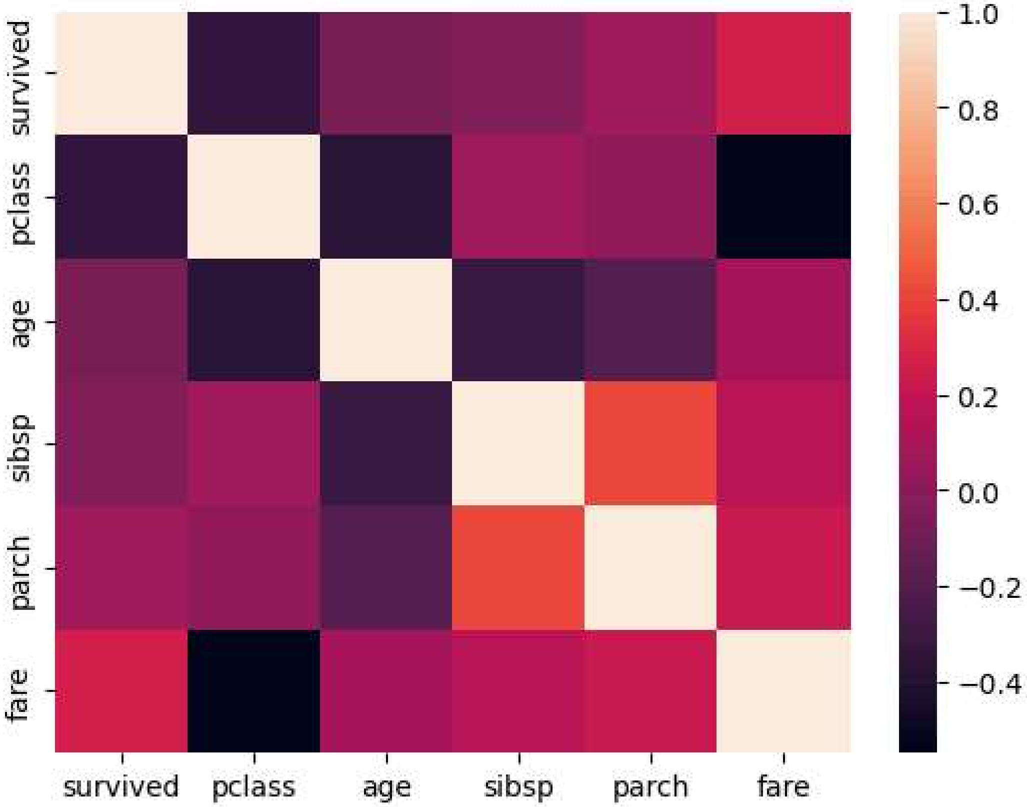
Out[35]:

```
<Axes: >
```



In [36]:
```
sns.heatmap(corr)
```

Out[36]:
```
<Axes: >
```

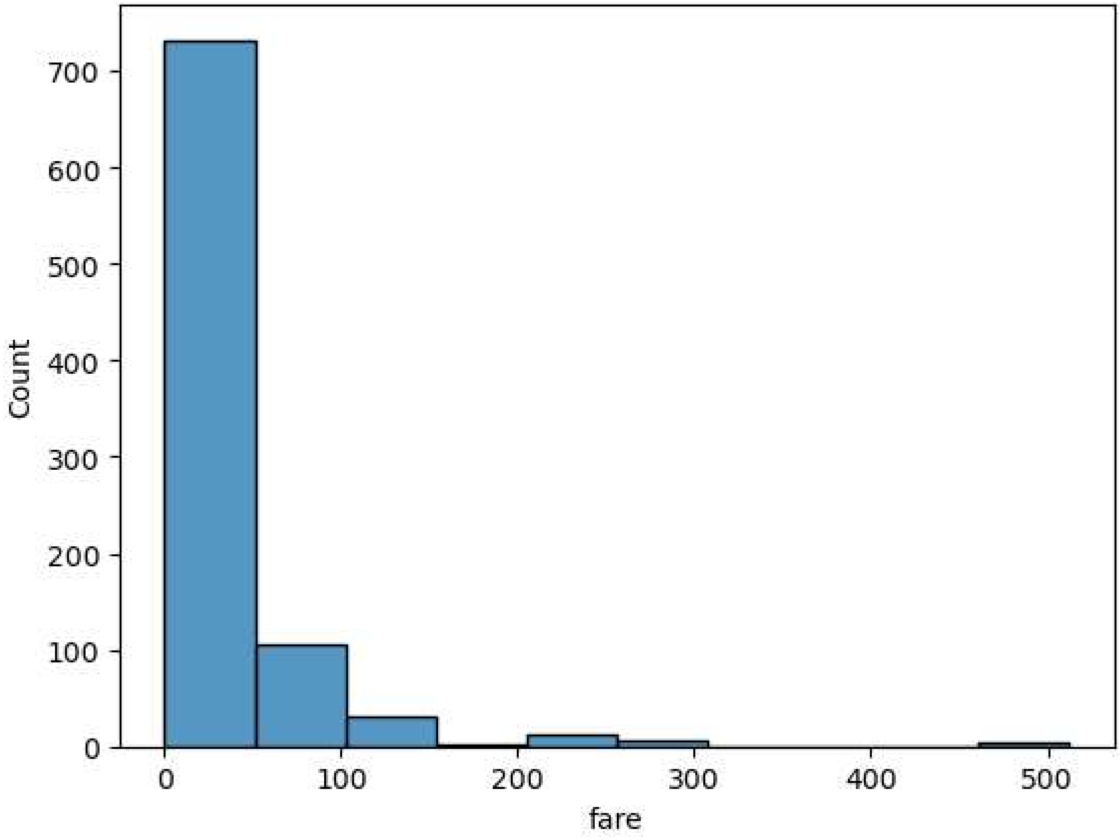


In [37]:

```
import seaborn as sns
df1 = sns.load_dataset('titanic')
sns.histplot(df1['fare'], kde=False, bins=10)
```

Out[37]:

```
<Axes: xlabel='fare', ylabel='Count'>
```



Name : Mohan Kadambande

Roll No. : 13212 (TECO-b1)