In [1]:
```
# Name:Atharv Santosh Danave
# Roll No.:11
# Practical No.:8
# Academic Year 2025-26
```

In [2]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

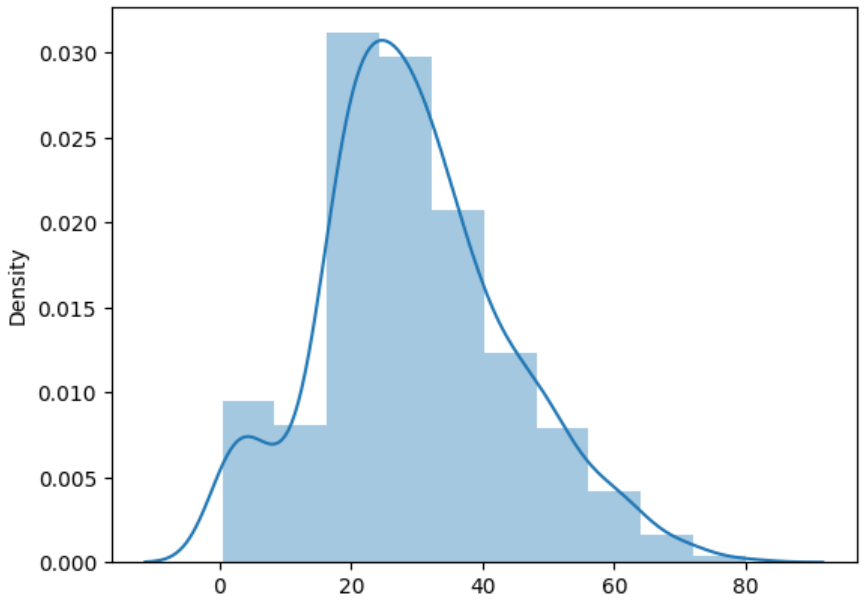
In [3]:
```
dataset = sns.load_dataset('titanic')
dataset.head()
```

Out[3]:

	survived	pclass	sex	age	sibsp	parch	fare	embarked	class	who	adult_male	deck	embark_town	alive
0	0	3	male	22.0	1	0	7.2500	S	Third	man	True	NaN	Southampton	no
1	1	1	female	38.0	1	0	71.2833	C	First	woman	False	C	Cherbourg	yes
2	1	3	female	26.0	0	0	7.9250	S	Third	woman	False	NaN	Southampton	yes
3	1	1	female	35.0	1	0	53.1000	S	First	woman	False	C	Southampton	yes
4	0	3	male	35.0	0	0	8.0500	S	Third	man	True	NaN	Southampton	no

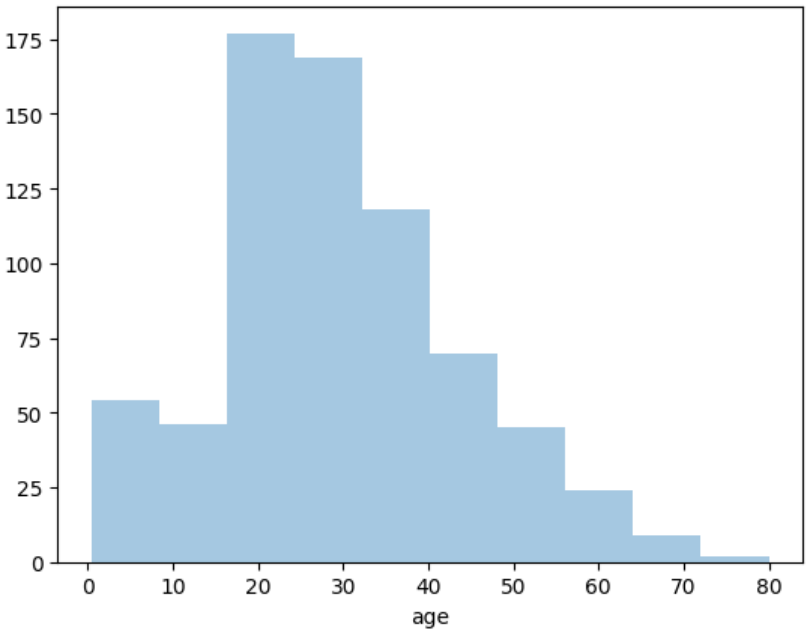
In [10]:
```
import warnings
warnings.filterwarnings("ignore")
sns.distplot(x = dataset['age'], bins = 10)
```

Out[10]: <Axes: ylabel='Density'>



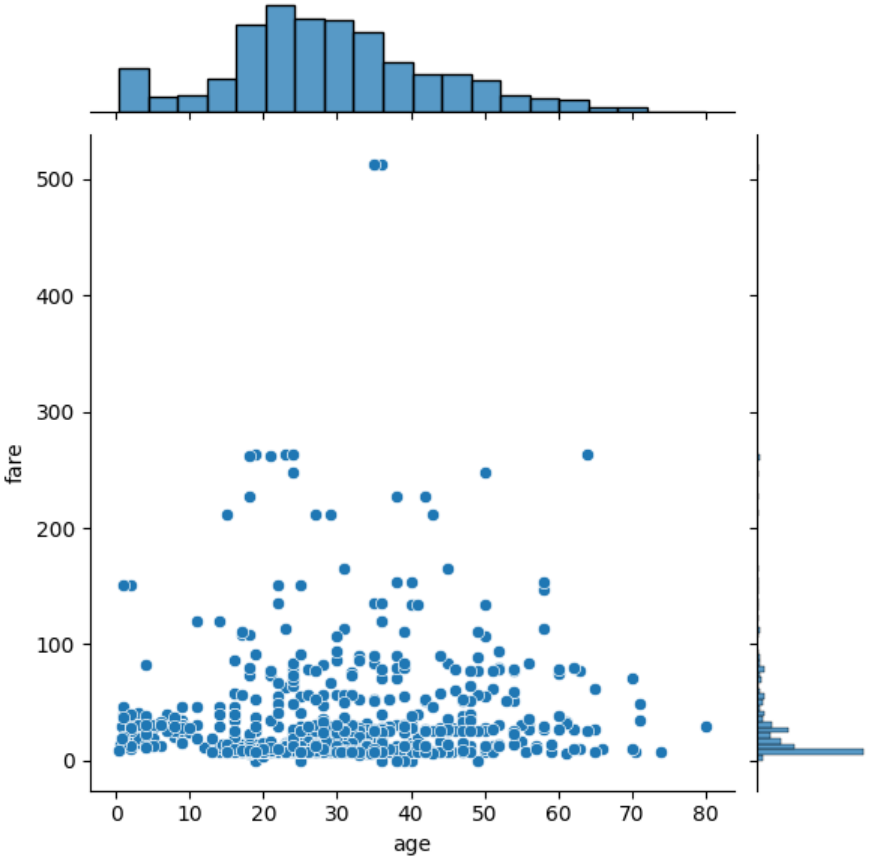
In [11]:
```
import warnings
warnings.filterwarnings("ignore")
sns.distplot(dataset['age'], bins = 10,kde=False)
```

Out[11]: <Axes: xlabel='age'>



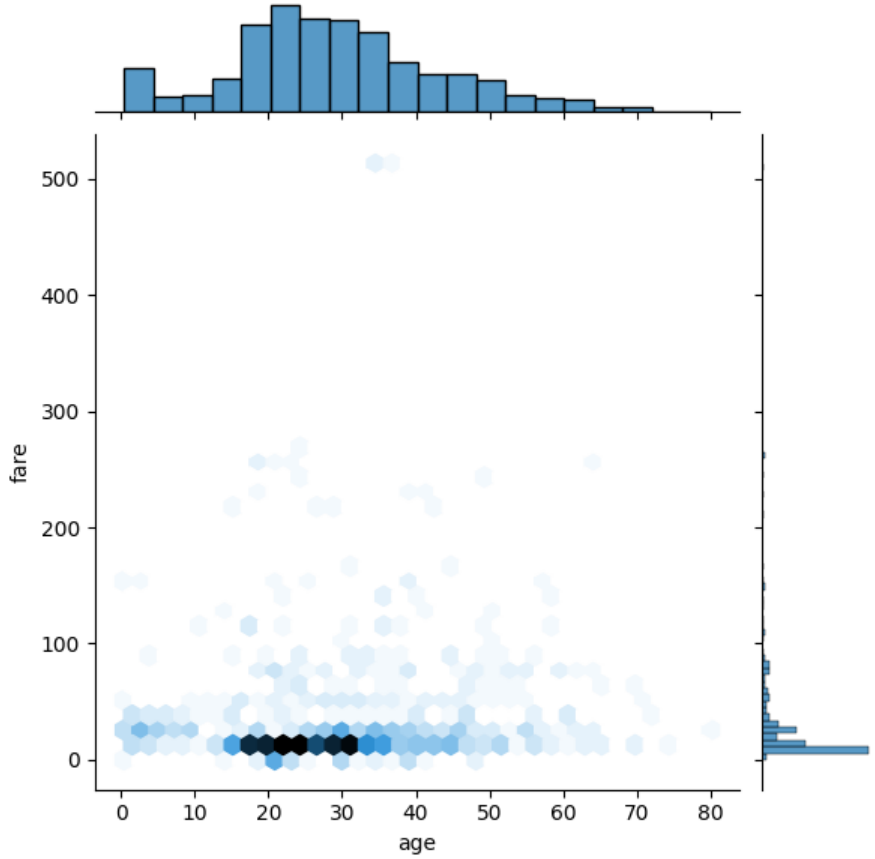
In [6]:
```
# For Plot 1
sns.jointplot(x = dataset['age'], y = dataset['fare'], kind ='scatter')
```

Out[6]: <seaborn.axisgrid.JointGrid at 0x79281c100710>



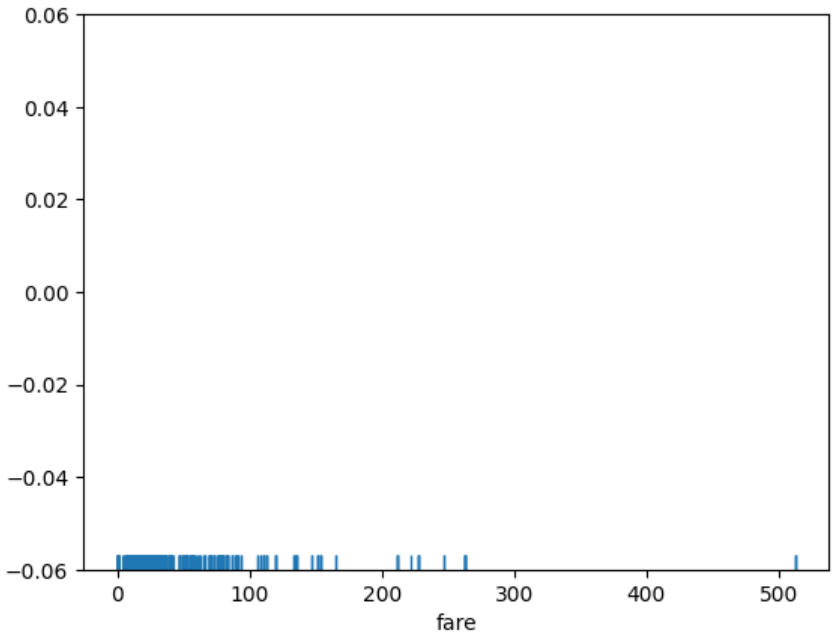
In [7]:
```
# For Plot 2
sns.jointplot(x = dataset['age'], y = dataset['fare'], kind = 'hex')
```

Out[7]: <seaborn.axisgrid.JointGrid at 0x79281c0f1490>



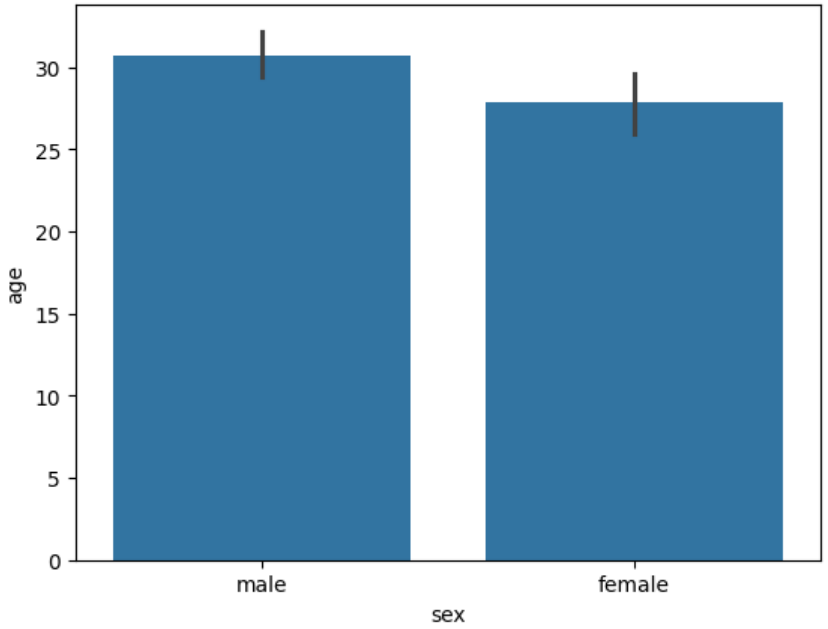
In [8]:
```
sns.rugplot(dataset['fare'])
```

Out[8]: <Axes: xlabel='fare'>



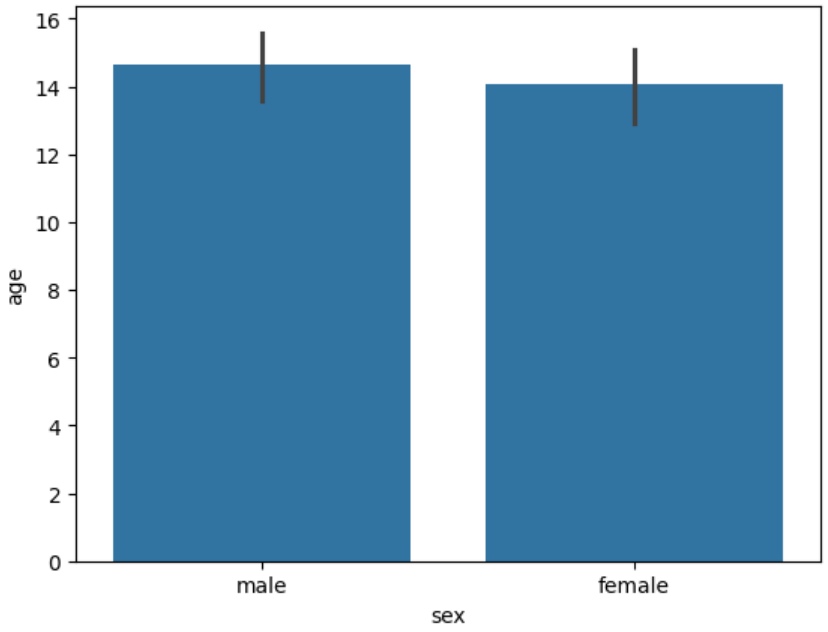
In [12]:
```
sns.barplot(x='sex', y='age', data=dataset)
```

Out[12]: <Axes: xlabel='sex', ylabel='age'>



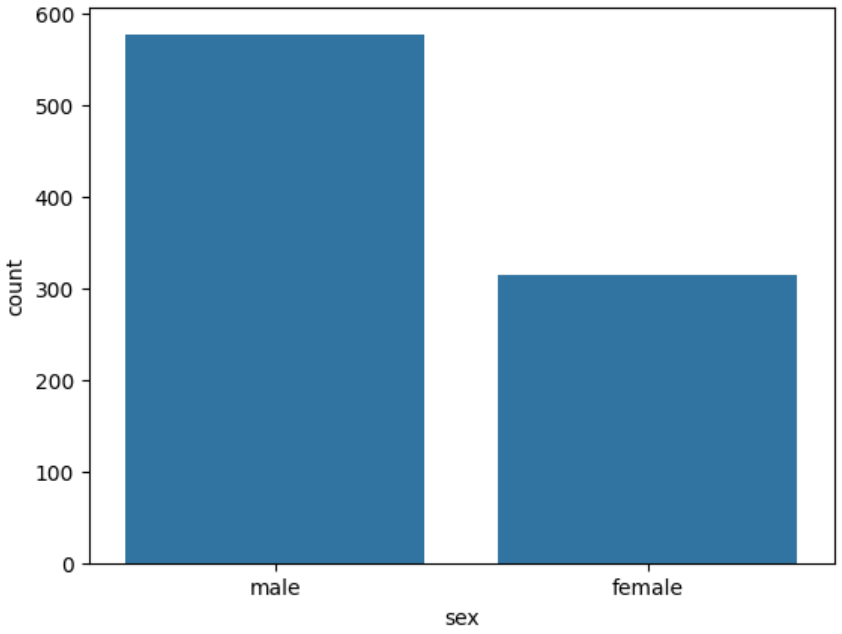
In [13]:
```
sns.barplot(x='sex', y='age', data=dataset, estimator=np.std)
```

Out[13]: <Axes: xlabel='sex', ylabel='age'>



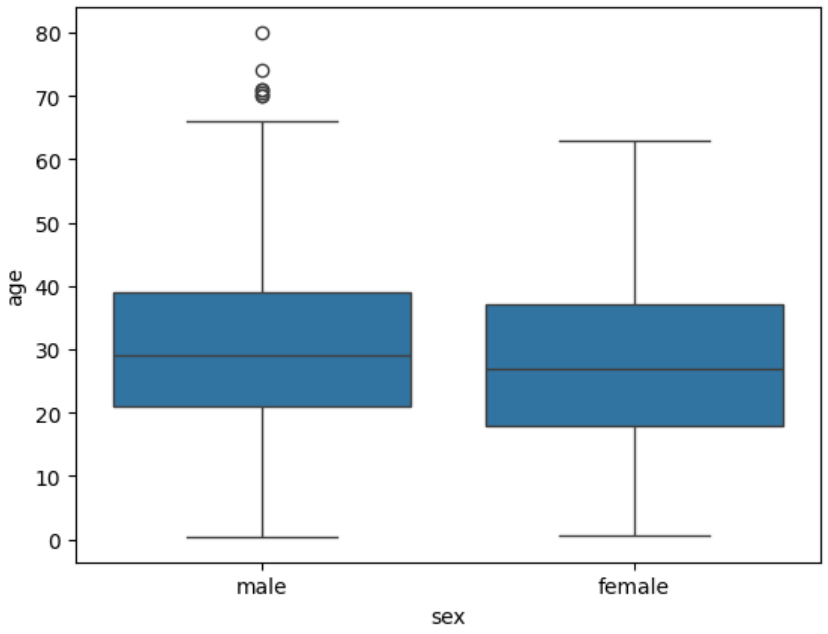
In [14]:
```
sns.countplot(x='sex', data=dataset)
```

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



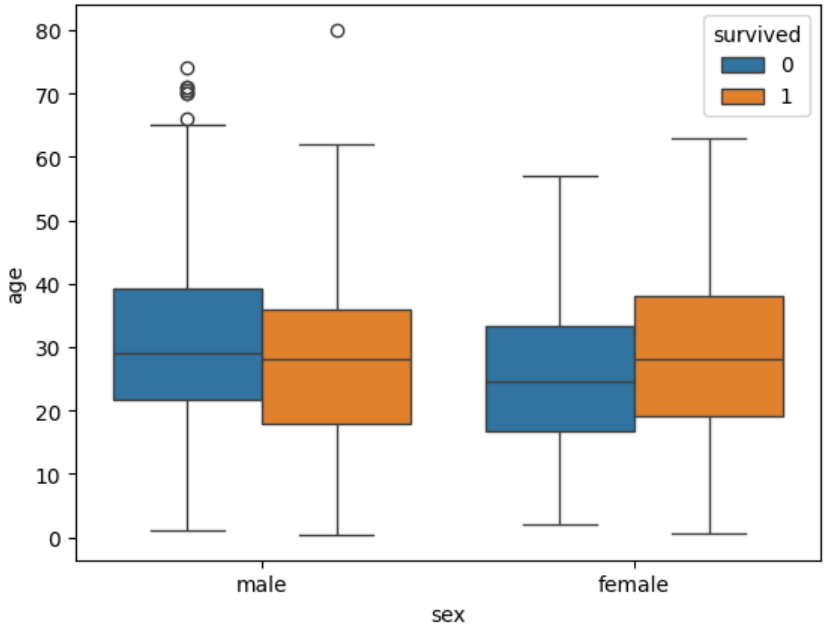
In [15]:
```
sns.boxplot(x='sex', y='age', data=dataset)
```

Out[15]: <Axes: xlabel='sex', ylabel='age'>



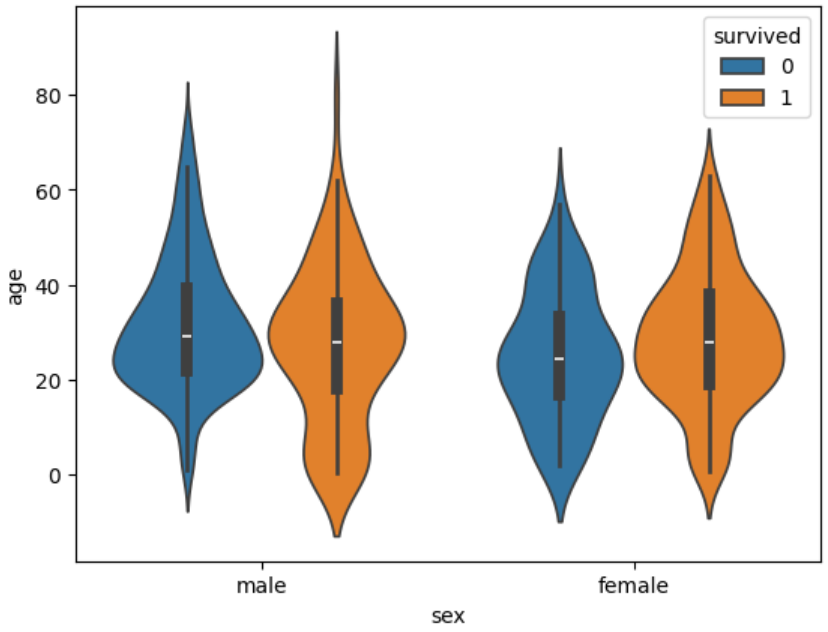
In [16]:
```
sns.boxplot(x='sex', y='age', data=dataset, hue="survived")
```

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



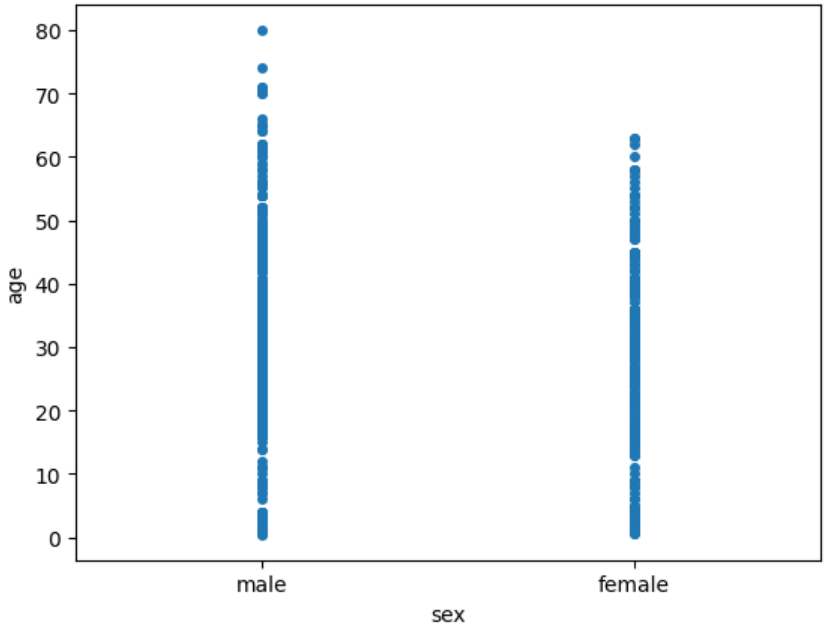
In [17]:
```
sns.violinplot(x='sex', y='age', data=dataset, hue='survived')
```

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



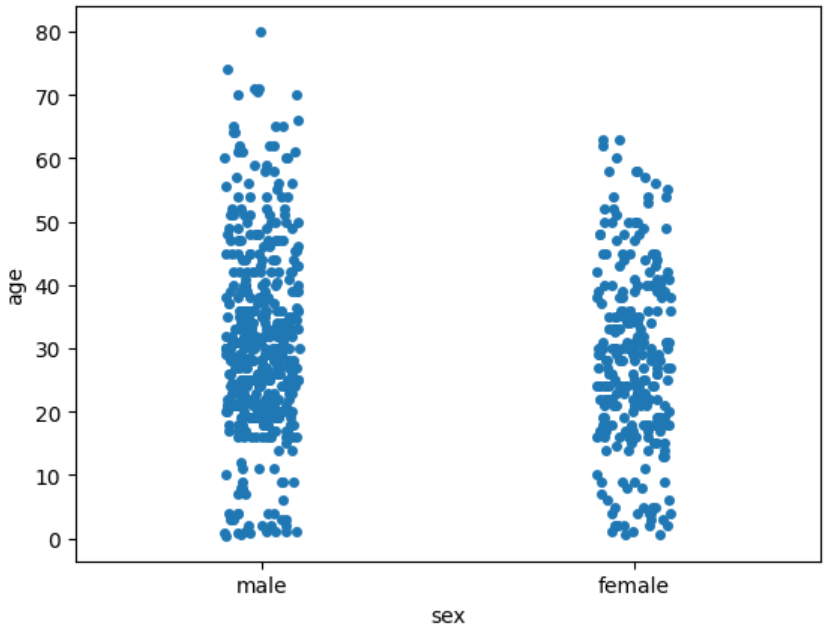
In [18]:
```
sns.stripplot(x='sex', y='age', data=dataset, jitter=False)
```

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



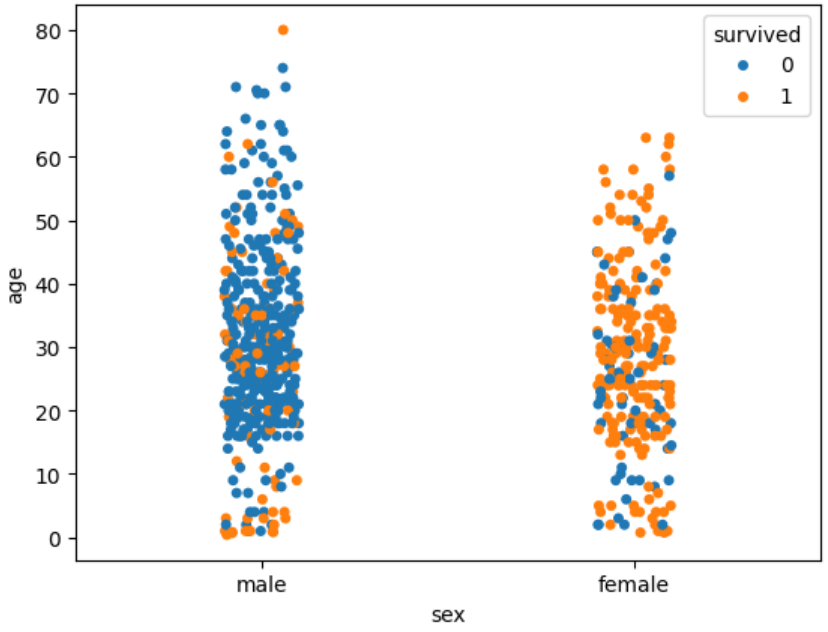
In [19]:
```
sns.stripplot(x='sex', y='age', data=dataset, jitter=True)
```

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



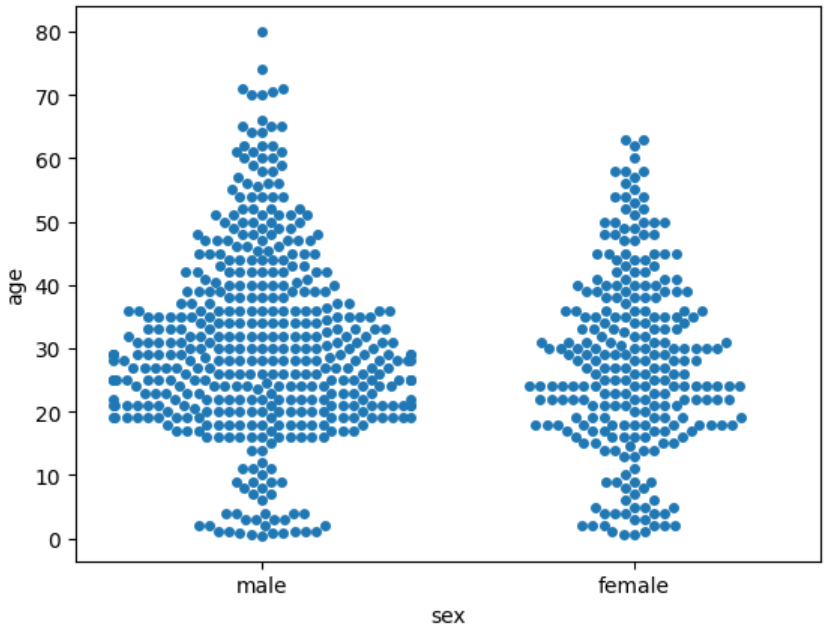
In [20]:
```
sns.stripplot(x='sex', y='age', data=dataset, jitter=True, hue='survived')
```

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



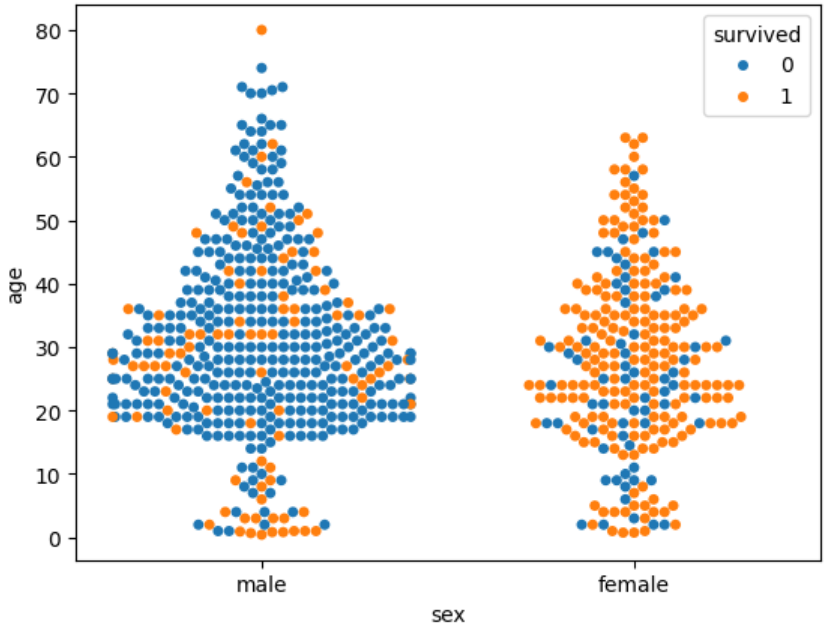
In [21]:
```
sns.swarmplot(x='sex', y='age', data=dataset)
```

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



In [22]:
```
sns.swarmplot(x='sex', y='age', data=dataset, hue='survived')
```

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



In [23]:
```
dataset = sns.load_dataset('titanic')
dataset.head()
```

Out[23]:

	survived	pclass	sex	age	sibsp	parch	fare	embarked	class	who	adult_male	deck	embark_town	alive
0	0	3	male	22.0	1	0	7.2500	S	Third	man	True	NaN	Southampton	no
1	1	1	female	38.0	1	0	71.2833	C	First	woman	False	C	Cherbourg	yes
2	1	3	female	26.0	0	0	7.9250	S	Third	woman	False	NaN	Southampton	yes
3	1	1	female	35.0	1	0	53.1000	S	First	woman	False	C	Southampton	yes
4	0	3	male	35.0	0	0	8.0500	S	Third	man	True	NaN	Southampton	no

In [24]:
```
dataset.select_dtypes(include=['number']).corr()
```

Out[24]:

	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 [26]:
```
print(dataset.dtypes)
```

```
survived         int64
pclass           int64
sex              int64
age            float64
sibsp            int64
parch            int64
fare           float64
embarked         int64
class            int64
who              int64
adult_male        bool
deck             int64
embark_town      int64
alive            int64
alone             bool
dtype: object
```

In [25]:
```
from sklearn.preprocessing import LabelEncoder
label_enc = LabelEncoder()
for col in dataset.select_dtypes(include=['object','category']).columns:
    dataset[col] = label_enc.fit_transform(dataset[col])
```

In [27]:
```
from sklearn.preprocessing import LabelEncoder
label_enc = LabelEncoder()
dataset['sex'] = label_enc.fit_transform(dataset['sex'])
print(dataset.corr())
```

```
             survived    pclass       sex       age     sibsp     parch  \
survived     1.000000 -0.338481 -0.543351 -0.077221 -0.035322  0.081629
pclass      -0.338481  1.000000  0.131900 -0.369226  0.083081  0.018443
sex         -0.543351  0.131900  1.000000  0.093254 -0.114631 -0.245489
age         -0.077221 -0.369226  0.093254  1.000000 -0.308247 -0.189119
sibsp       -0.035322  0.083081 -0.114631 -0.308247  1.000000  0.414838
parch        0.081629  0.018443 -0.245489 -0.189119  0.414838  1.000000
fare         0.257307 -0.549500 -0.182333  0.096067  0.159651  0.216225
embarked    -0.163517  0.157112  0.104057 -0.025252  0.066654  0.038322
class       -0.338481  1.000000  0.131900 -0.369226  0.083081  0.018443
who          0.325753 -0.196793 -0.639773  0.378685 -0.136003 -0.055682
adult_male  -0.557080  0.094035  0.908578  0.280328 -0.253586 -0.349943
deck        -0.294804  0.743251  0.118282 -0.267987  0.041333 -0.031308
embark_town -0.163517  0.157112  0.104057 -0.025252  0.066654  0.038322
alive        1.000000 -0.338481 -0.543351 -0.077221 -0.035322  0.081629
alone       -0.203367  0.135207  0.303646  0.198270 -0.584471 -0.583398

                fare  embarked     class       who  adult_male      deck  \
survived     0.257307 -0.163517 -0.338481  0.325753   -0.557080 -0.294804
pclass      -0.549500  0.157112  1.000000 -0.196793    0.094035  0.743251
sex         -0.182333  0.104057  0.131900 -0.639773    0.908578  0.118282
age          0.096067 -0.025252 -0.369226  0.378685    0.280328 -0.267987
sibsp        0.159651  0.066654  0.083081 -0.136003   -0.253586  0.041333
parch        0.216225  0.038322  0.018443 -0.055682   -0.349943 -0.031308
fare         1.000000 -0.221226 -0.549500  0.146290   -0.182024 -0.525994
embarked    -0.221226  1.000000  0.157112 -0.060177    0.088725  0.191735
class       -0.549500  0.157112  1.000000 -0.196793    0.094035  0.743251
who          0.146290 -0.060177 -0.196793  1.000000   -0.437532 -0.153766
adult_male  -0.182024  0.088725  0.094035 -0.437532    1.000000  0.098553
deck        -0.525994  0.191735  0.743251 -0.153766    0.098553  1.000000
embark_town -0.221226  1.000000  0.157112 -0.060177    0.088725  0.191735
alive        0.257307 -0.163517 -0.338481  0.325753   -0.557080 -0.294804
alone       -0.271832  0.065610  0.135207  0.006540    0.404744  0.137515

             embark_town     alive     alone
survived       -0.163517  1.000000 -0.203367
pclass          0.157112 -0.338481  0.135207
sex             0.104057 -0.543351  0.303646
age            -0.025252 -0.077221  0.198270
sibsp           0.066654 -0.035322 -0.584471
parch           0.038322  0.081629 -0.583398
fare           -0.221226  0.257307 -0.271832
embarked        1.000000 -0.163517  0.065610
class           0.157112 -0.338481  0.135207
who            -0.060177  0.325753  0.006540
adult_male      0.088725 -0.557080  0.404744
deck            0.191735 -0.294804  0.137515
embark_town     1.000000 -0.163517  0.065610
alive          -0.163517  1.000000 -0.203367
alone           0.065610 -0.203367  1.000000
```

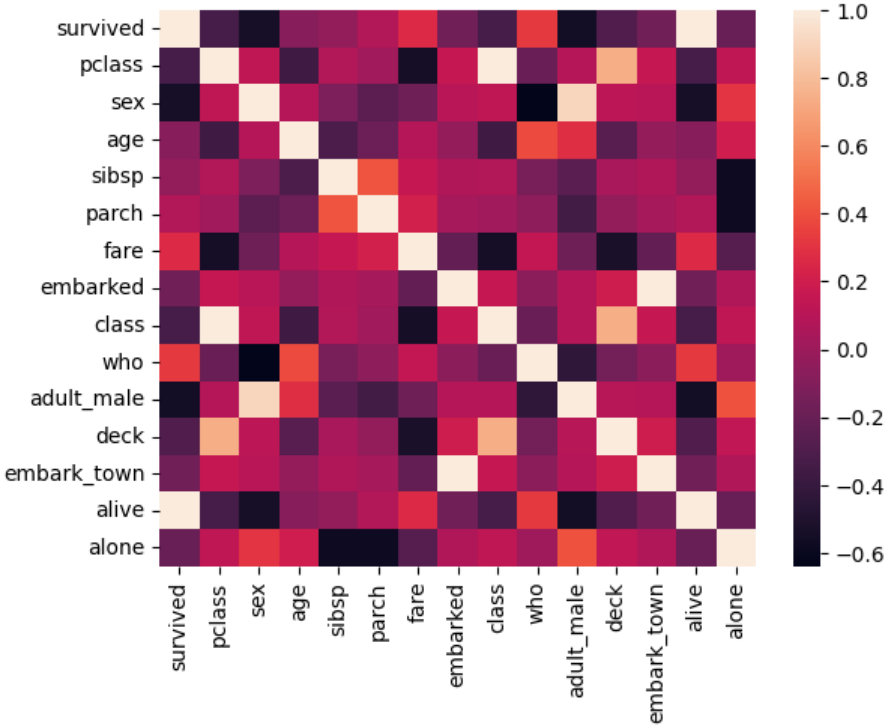
In [28]:
```
dataset.corr()
```

Out[28]:

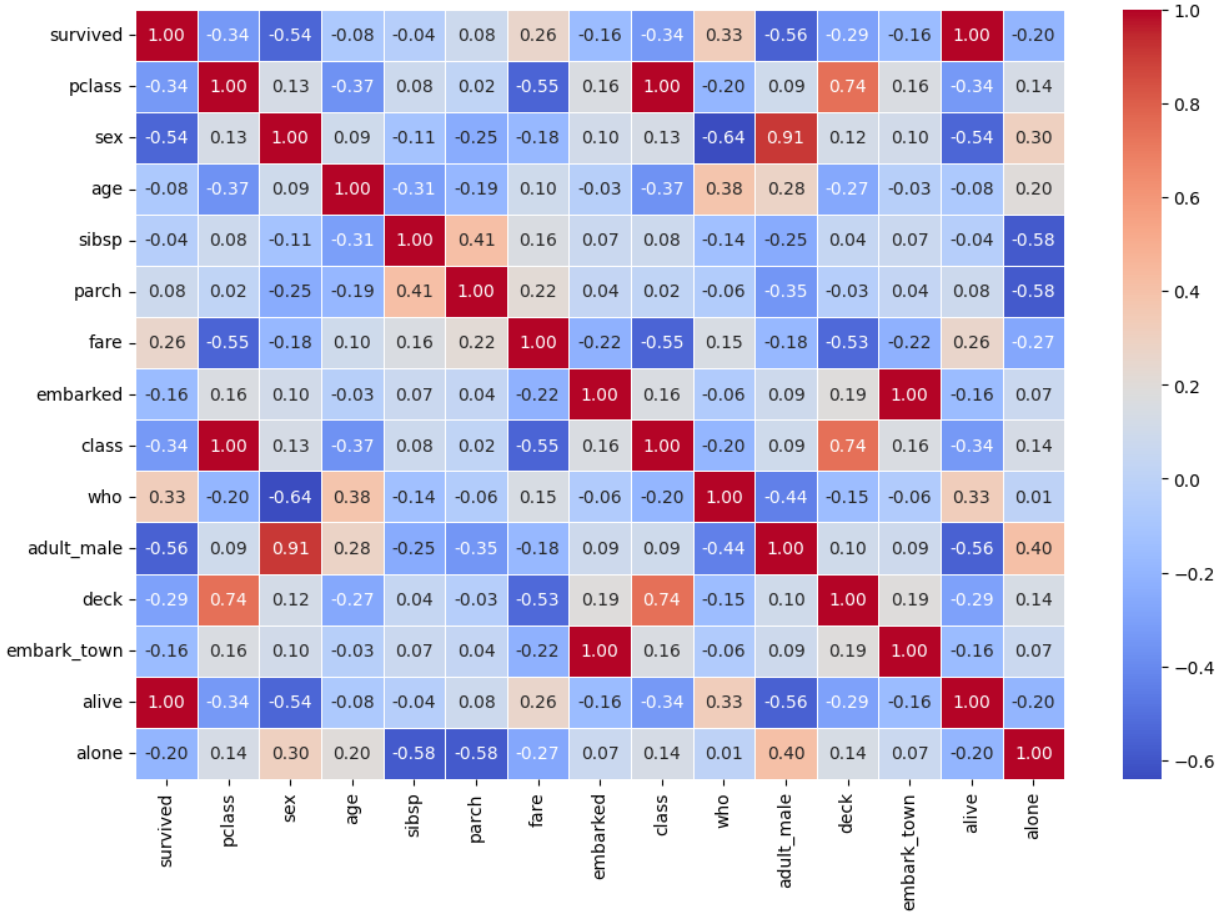
	survived	pclass	sex	age	sibsp	parch	fare	embarked	class	
survived	1.000000	-0.338481	-0.543351	-0.077221	-0.035322	0.081629	0.257307	-0.163517	-0.338481	0.325
pclass	-0.338481	1.000000	0.131900	-0.369226	0.083081	0.018443	-0.549500	0.157112	1.000000	-0.196
sex	-0.543351	0.131900	1.000000	0.093254	-0.114631	-0.245489	-0.182333	0.104057	0.131900	-0.639
age	-0.077221	-0.369226	0.093254	1.000000	-0.308247	-0.189119	0.096067	-0.025252	-0.369226	0.378
sibsp	-0.035322	0.083081	-0.114631	-0.308247	1.000000	0.414838	0.159651	0.066654	0.083081	-0.136
parch	0.081629	0.018443	-0.245489	-0.189119	0.414838	1.000000	0.216225	0.038322	0.018443	-0.055
fare	0.257307	-0.549500	-0.182333	0.096067	0.159651	0.216225	1.000000	-0.221226	-0.549500	0.146
embarked	-0.163517	0.157112	0.104057	-0.025252	0.066654	0.038322	-0.221226	1.000000	0.157112	-0.060
class	-0.338481	1.000000	0.131900	-0.369226	0.083081	0.018443	-0.549500	0.157112	1.000000	-0.196
who	0.325753	-0.196793	-0.639773	0.378685	-0.136003	-0.055682	0.146290	-0.060177	-0.196793	1.000
adult_male	-0.557080	0.094035	0.908578	0.280328	-0.253586	-0.349943	-0.182024	0.088725	0.094035	-0.437
deck	-0.294804	0.743251	0.118282	-0.267987	0.041333	-0.031308	-0.525994	0.191735	0.743251	-0.153
embark_town	-0.163517	0.157112	0.104057	-0.025252	0.066654	0.038322	-0.221226	1.000000	0.157112	-0.060
alive	1.000000	-0.338481	-0.543351	-0.077221	-0.035322	0.081629	0.257307	-0.163517	-0.338481	0.325
alone	-0.203367	0.135207	0.303646	0.198270	-0.584471	-0.583398	-0.271832	0.065610	0.135207	0.006

In [29]:
```
corr = dataset.corr()
sns.heatmap(corr)
```

Out[29]: <Axes: >

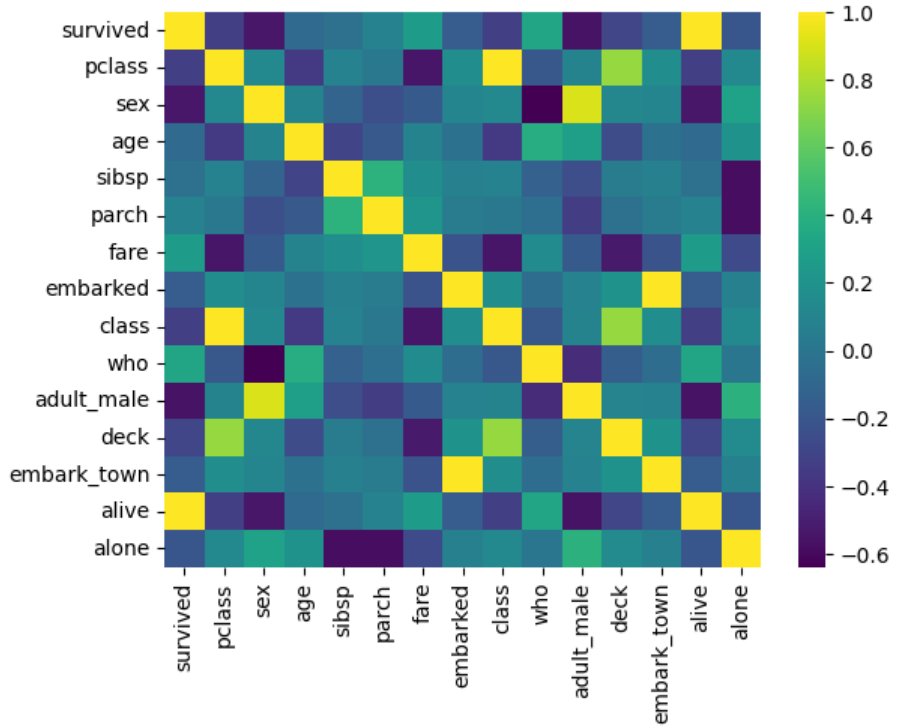


In [30]:
```
# Create the heatmap with better formatting
plt.figure(figsize=(12, 8))
sns.heatmap(dataset.corr(),
            annot=True,          # Show correlation values
            fmt=".2f",           # Limit values to 2 decimal places
            cmap="coolwarm",     # Choose a better color scheme
            linewidths=0.5,      # Add space between cells
            annot_kws={"size": 10})  # Adjust font size for better readability
plt.show()
```



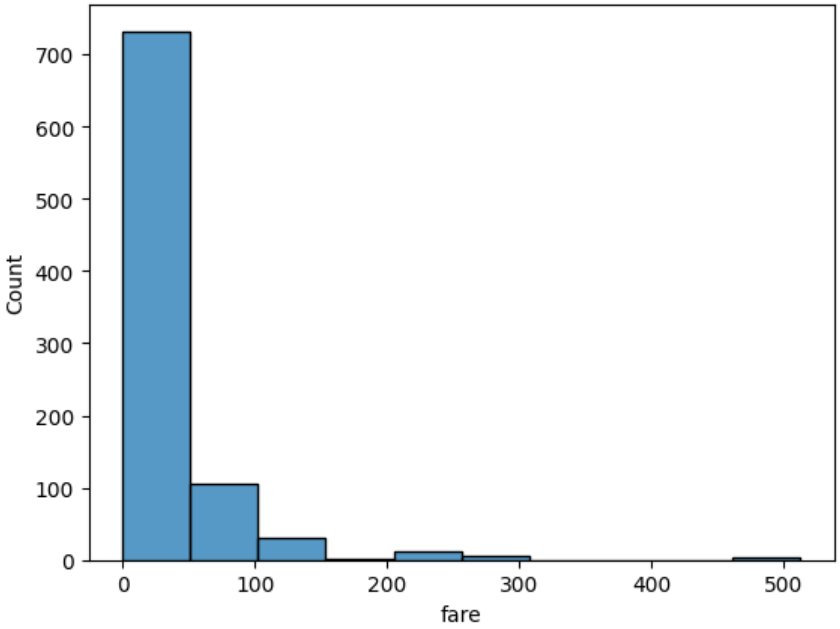
In [31]:
```
corr = dataset.corr()
sns.heatmap(corr, cmap='viridis')
```

Out[31]: <Axes: >



In [32]:
```
sns.histplot(dataset['fare'], kde=False, bins=10)
```

Out[32]: <Axes: xlabel='fare', ylabel='Count'>


700
600
500
400
300
200
100
0
Count
0
100
200
300
400
500
fare