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

```
titanic = sn.load_dataset('titanic')
```

```
titanic.isnull().sum()
```

```
                 0
   survived      0
     pclass      0
        sex      0
        age    177
      sibsp      0
      parch      0
       fare      0
   embarked      2
      class      0
        who      0
 adult_male      0
       deck    688
embark_town      2
      alive      0
      alone      0

dtype: int64
```

```
titanic['age'].dropna()
```

```
       age
  0   22.0
  1   38.0
  2   26.0
  3   35.0
  4   35.0
 ...   ...
885   39.0
886   27.0
887   19.0
889   26.0
890   32.0
714 rows × 1 columns

dtype: float64
```

```
titanic['embarked'].dropna()
```

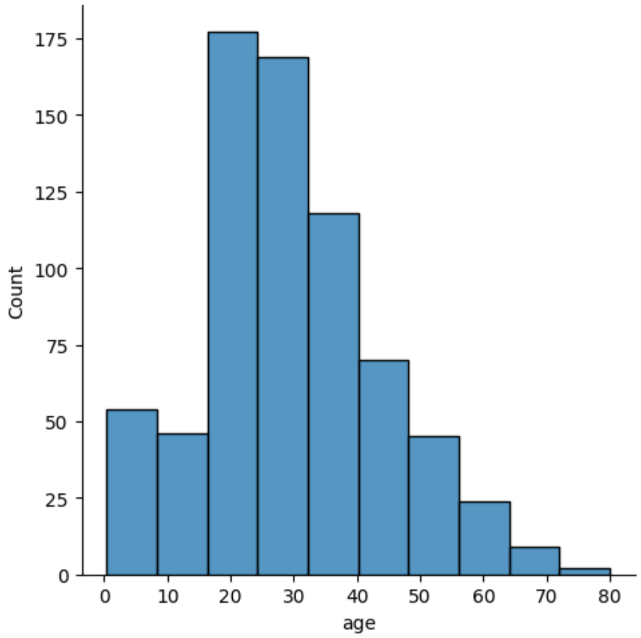
```
     embarked
0           S
1           C
2           S
3           S
4           S
...       ...
886         S
887         S
888         S
889         C
890         Q
889 rows × 1 columns

dtype: object
```

```
titanic['embark_town'].dropna()
```

```
     embark_town
0    Southampton
1      Cherbourg
2    Southampton
3    Southampton
4    Southampton
...          ...
886  Southampton
887  Southampton
888  Southampton
889    Cherbourg
890   Queenstown
889 rows × 1 columns

dtype: object
```

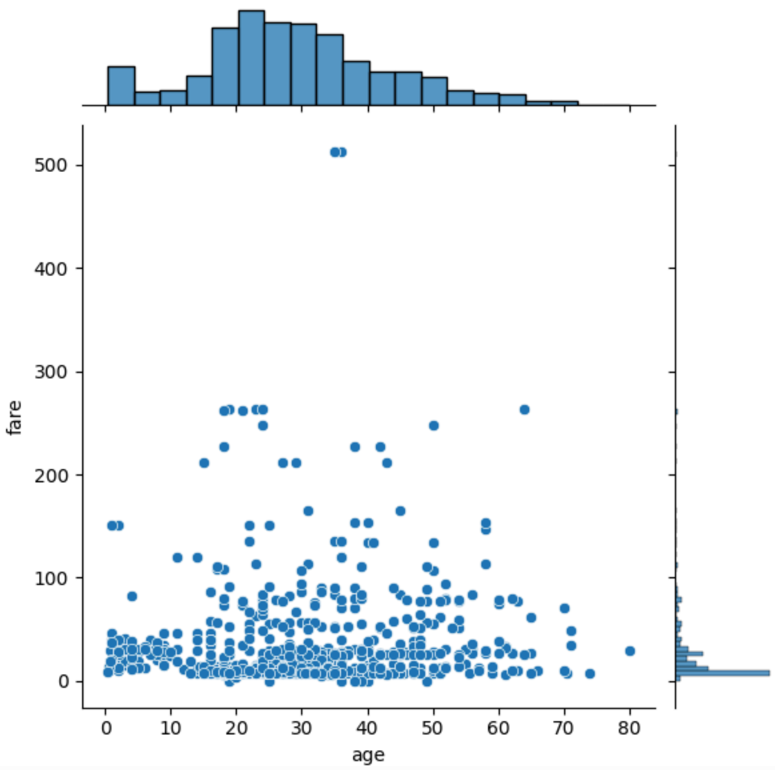
```
sn.displot(x = titanic['age'], bins = 10)
```

```
<seaborn.axisgrid.FacetGrid at 0x78ee954478d0>
```



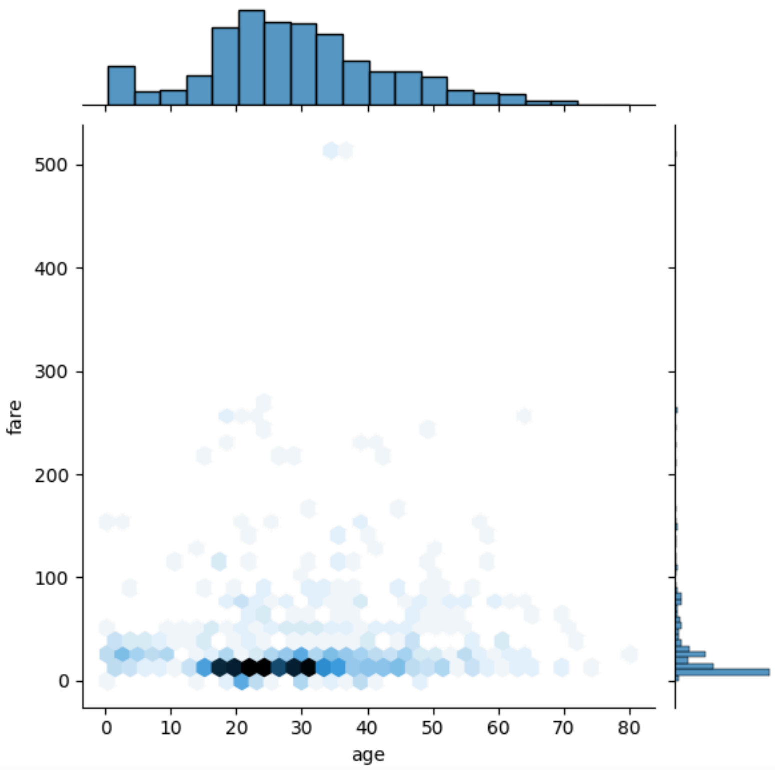
```
sn.jointplot(x = titanic['age'], y = titanic['fare'], kind = 'scatter')
```

```
<seaborn.axisgrid.JointGrid at 0x78ee95c14f10>
```



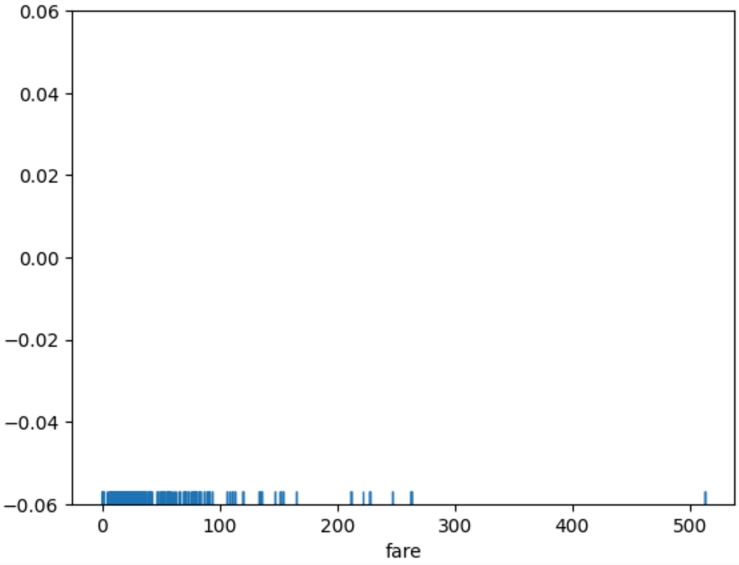
```
sn.jointplot(x = titanic['age'], y = titanic['fare'], kind = 'hex')
```

```
<seaborn.axisgrid.JointGrid at 0x78ee90f67790>
```



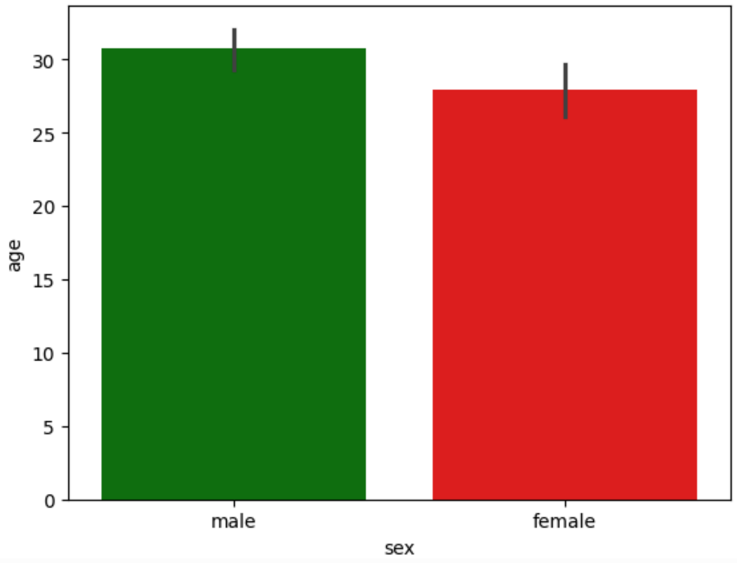
```
sn.rugplot(titanic['fare'])
```

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



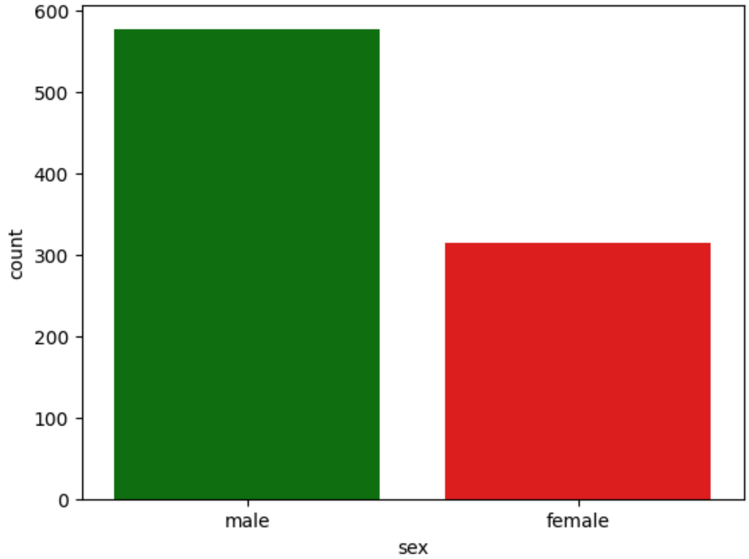
```
sn.barplot(x='sex', y='age', data=titanic, palette={'male': 'green', 'female': 'red'})
```

```
<ipython-input-30-8f5ed0905f91>:1: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `x` variable to `hue` and set `le

  sn.barplot(x='sex', y='age', data=titanic, palette={'male': 'green', 'female': 'red'})
<Axes: xlabel='sex', ylabel='age'>
```



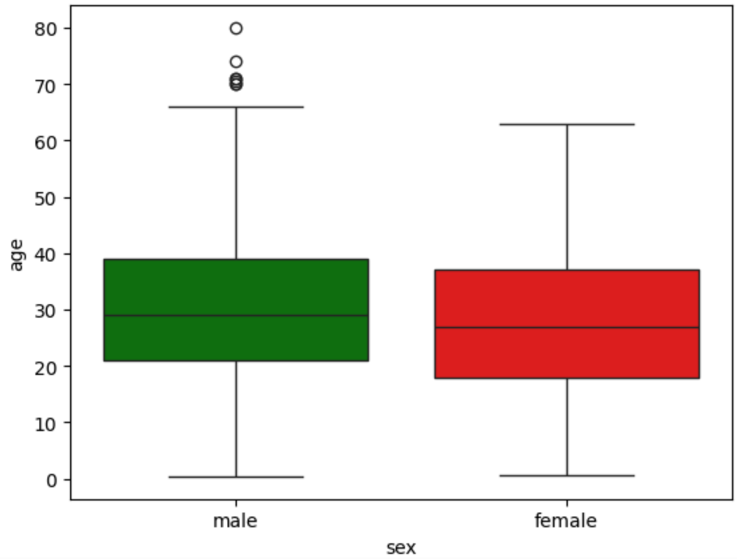
```
sn.countplot(x='sex', data=titanic, palette={'male': 'green', 'female': 'red'})
```

```
<ipython-input-31-d44eb98e1cc2>:1: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `x` variable to `hue` and set `le

  sn.countplot(x='sex', data=titanic, palette={'male': 'green', 'female': 'red'})
<Axes: xlabel='sex', ylabel='count'>
```



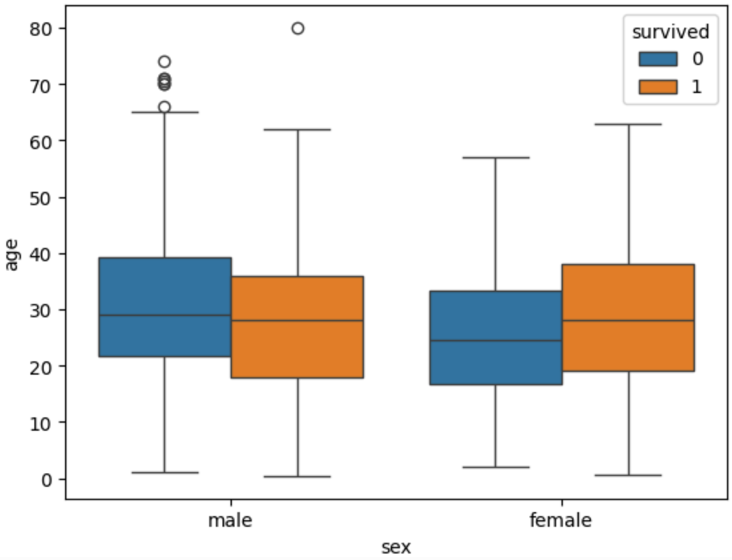
```
sn.boxplot(x='sex', y='age', data=titanic, palette={'male': 'green', 'female': 'red'})
```

```
<ipython-input-29-cbdd41592f1e>:1: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `x` variable to `hue` and set `le

  sn.boxplot(x='sex', y='age', data=titanic, palette={'male': 'green', 'female': 'red'})
<Axes: xlabel='sex', ylabel='age'>
```



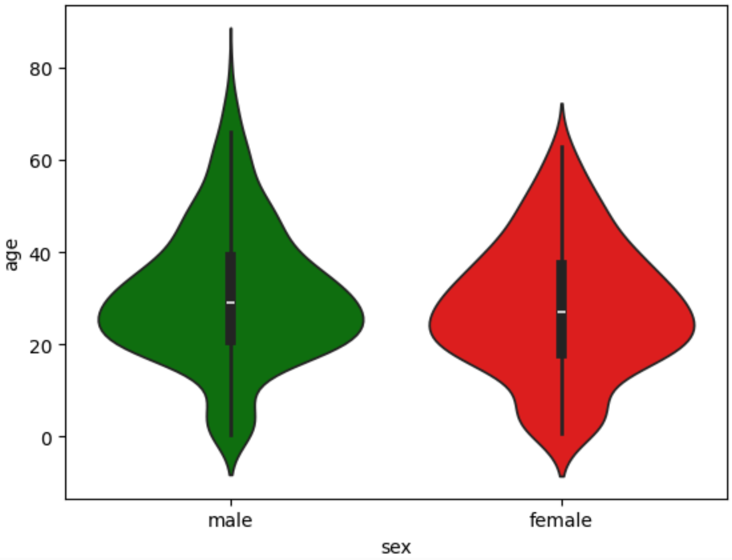
```
sn.boxplot(x='sex', y='age', data=titanic, hue='survived')
```

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



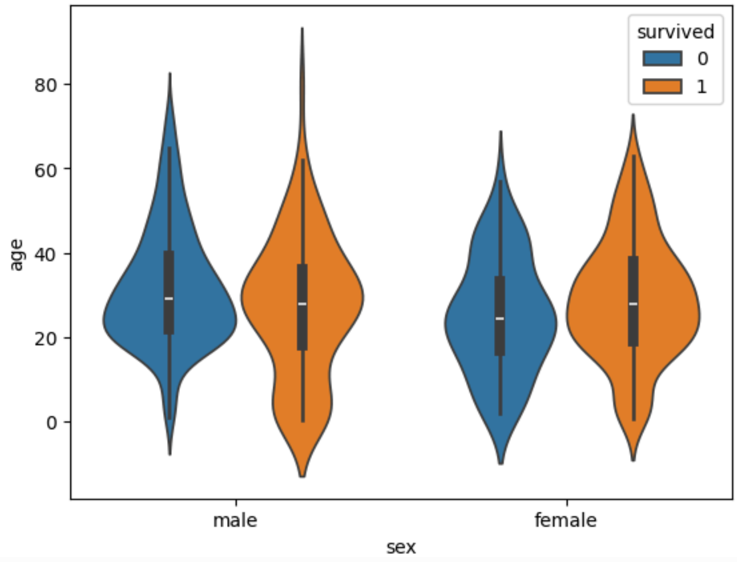
```
sn.violinplot(x='sex', y='age', data=titanic, palette={'male': 'green', 'female': 'red'})
```

```
<ipython-input-34-240f0d0ac587>:1: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `x` variable to `hue` and set `le

  sn.violinplot(x='sex', y='age', data=titanic, palette={'male': 'green', 'female': 'red'})
<Axes: xlabel='sex', ylabel='age'>
```



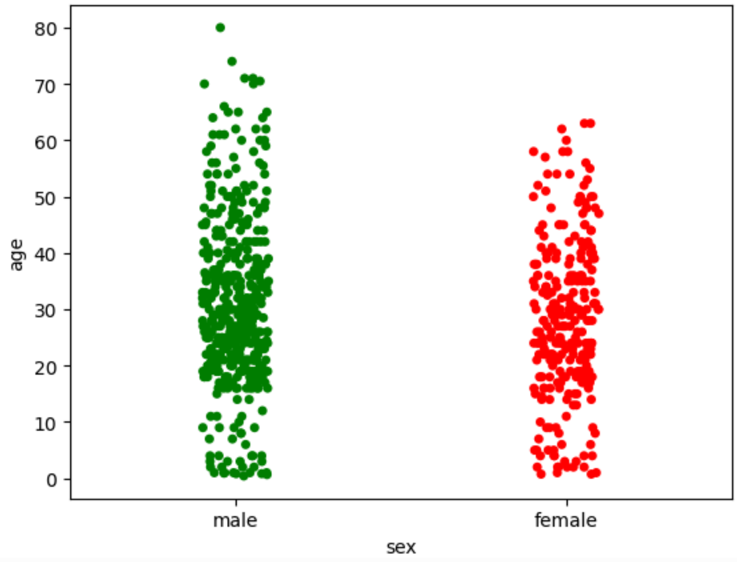
```
sn.violinplot(x='sex', y='age', data=titanic, hue='survived')
```

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



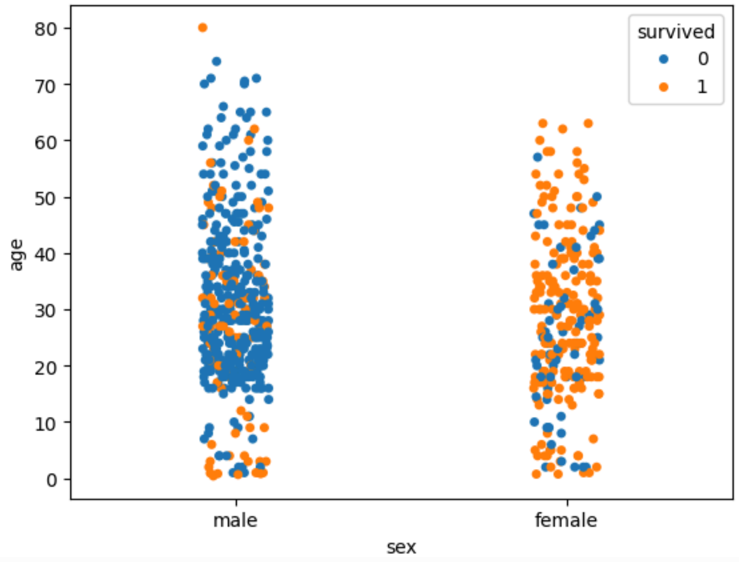
```
sn.stripplot(x='sex', y='age', data=titanic, jitter=True, palette={'male': 'green', 'female': 'red'})
```

```
<ipython-input-37-b1ca2420575a>:1: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `x` variable to `hue` and set `le

  sn.stripplot(x='sex', y='age', data=titanic, jitter=True, palette={'male': 'green', 'female': 'red'})
<Axes: xlabel='sex', ylabel='age'>
```



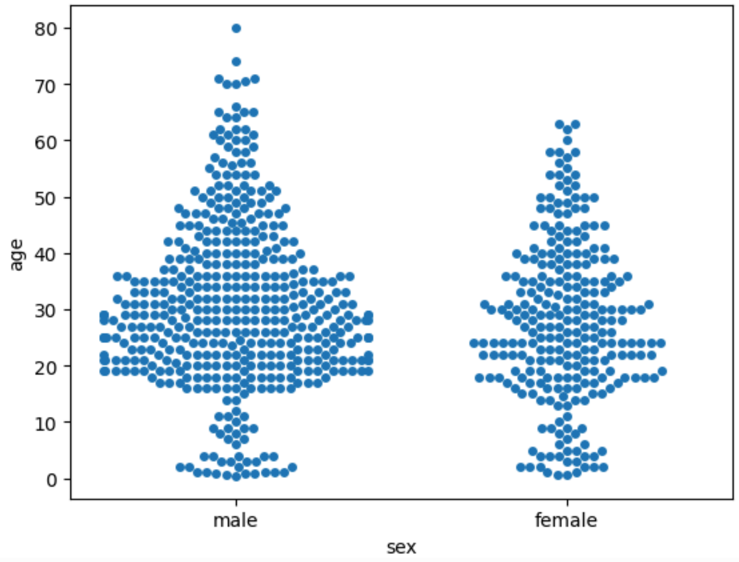
```
sn.stripplot(x='sex', y='age', data=titanic, jitter=True, hue='survived')
```

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



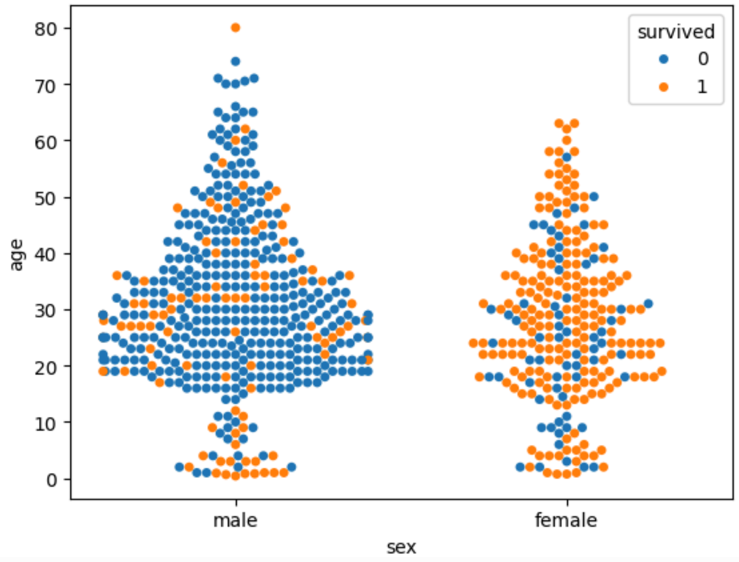
```
sn.swarmplot(x='sex', y='age', data=titanic)
```

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



```
sn.swarmplot(x='sex', y='age', data=titanic,hue='survived')
```

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



```
# Specify the columns you want to include in the correlation calculation
columns_to_include = ['survived', 'pclass', 'age', 'fare', 'sibsp','parch','fare','adult_male','alone'
```

```
# Calculate the correlation matrix for the selected columns
corr = titanic[columns_to_include].corr()

# Display the correlation matrix
print(corr)
```

```
            survived    pclass       age      fare     sibsp     parch  \
survived    1.000000 -0.338481 -0.077221  0.257307 -0.035322  0.081629   
pclass     -0.338481  1.000000 -0.369226 -0.549500  0.083081  0.018443   
age        -0.077221 -0.369226  1.000000  0.096067 -0.308247 -0.189119   
fare        0.257307 -0.549500  0.096067  1.000000  0.159651  0.216225   
sibsp      -0.035322  0.083081 -0.308247  0.159651  1.000000  0.414838   
parch       0.081629  0.018443 -0.189119  0.216225  0.414838  1.000000   
fare        0.257307 -0.549500  0.096067  1.000000  0.159651  0.216225   
adult_male -0.557080  0.094035  0.280328 -0.182024 -0.253586 -0.349943   
alone      -0.203367  0.135207  0.198270 -0.271832 -0.584471 -0.583398   

                fare  adult_male     alone  
survived    0.257307   -0.557080 -0.203367  
pclass     -0.549500    0.094035  0.135207  
age         0.096067    0.280328  0.198270  
fare        1.000000   -0.182024 -0.271832  
sibsp       0.159651   -0.253586 -0.584471  
parch       0.216225   -0.349943 -0.583398  
fare        1.000000   -0.182024 -0.271832  
adult_male -0.182024    1.000000  0.404744  
alone      -0.271832    0.404744  1.000000  
```

```
sn.heatmap(corr)
```

```
<Axes: >
```