TE C Assignment No - 08

May 19, 2023

[3]:
```
import seaborn as sns
import pandas as pd
```

[4]:
```
titanic = sns.load_dataset("titanic")
```

[5]:
```
titanic
```

[5]:
```
     survived  pclass     sex   age  sibsp  parch     fare embarked   class  \
0           0       3    male  22.0      1      0   7.2500        S   Third
1           1       1  female  38.0      1      0  71.2833        C   First
2           1       3  female  26.0      0      0   7.9250        S   Third
3           1       1  female  35.0      1      0  53.1000        S   First
4           0       3    male  35.0      0      0   8.0500        S   Third
..        ...     ...     ...   ...    ...    ...      ...      ...     ...
886         0       2    male  27.0      0      0  13.0000        S  Second
887         1       1  female  19.0      0      0  30.0000        S   First
888         0       3  female   NaN      1      2  23.4500        S   Third
889         1       1    male  26.0      0      0  30.0000        C   First
890         0       3    male  32.0      0      0   7.7500        Q   Third

       who  adult_male deck  embark_town alive  alone
0      man        True  NaN  Southampton    no  False
1    woman       False    C    Cherbourg   yes  False
2    woman       False  NaN  Southampton   yes   True
3    woman       False    C  Southampton   yes  False
4      man        True  NaN  Southampton    no   True
..     ...         ...  ...          ...   ...    ...
886    man        True  NaN  Southampton    no   True
887  woman       False    B  Southampton   yes   True
888  woman       False  NaN  Southampton    no  False
889    man        True    C    Cherbourg   yes   True
890    man        True  NaN   Queenstown    no   True

[891 rows x 15 columns]
```

[6]:
```
titanic.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
```

[7]:
```
x = titanic["fare"]
x
```

[7]:
```
0       7.2500
1      71.2833
2       7.9250
3      53.1000
4       8.0500
        …
886    13.0000
887    30.0000
888    23.4500
889    30.0000
890     7.7500
Name: fare, Length: 891, dtype: float64
```

[8]:
```
#titanic.iloc[:,"fare"]
```

[9]:
```
titanic.describe()
```

[9]:

	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
```

[10]:
```
#First Part
#Data Cleanup
#inform us about empty fileds in column
titanic.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
```

[11]:
```
#Dropping the not required columns
titanic_cleaned = titanic.
  ↪drop(['pclass','embarked','deck','embark_town'],axis=1)
titanic_cleaned.head(15)
```

[11]:

	survived	sex	age	sibsp	parch	fare	class	who	adult_male	\
0	0	male	22.0	1	0	7.2500	Third	man	True	
1	1	female	38.0	1	0	71.2833	First	woman	False	
2	1	female	26.0	0	0	7.9250	Third	woman	False	
3	1	female	35.0	1	0	53.1000	First	woman	False	
4	0	male	35.0	0	0	8.0500	Third	man	True	
5	0	male	NaN	0	0	8.4583	Third	man	True	
6	0	male	54.0	0	0	51.8625	First	man	True	
7	0	male	2.0	3	1	21.0750	Third	child	False	

```
8          1  female  27.0      0      2  11.1333   Third  woman       False
9          1  female  14.0      1      0  30.0708  Second  child       False
10         1  female   4.0      1      1  16.7000   Third  child       False
11         1  female  58.0      0      0  26.5500   First  woman       False
12         0    male  20.0      0      0   8.0500   Third    man        True
13         0    male  39.0      1      5  31.2750   Third    man        True
14         0  female  14.0      0      0   7.8542   Third  child       False

   alive  alone
0     no  False
1    yes  False
2    yes   True
3    yes  False
4     no   True
5     no   True
6     no   True
7     no  False
8    yes  False
9    yes  False
10   yes  False
11   yes   True
12    no   True
13    no  False
14    no   True
```

[12]:
```
titanic_cleaned.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 11 columns):
 #   Column      Non-Null Count  Dtype
---  ------      --------------  -----
 0   survived    891 non-null    int64
 1   sex         891 non-null    object
 2   age         714 non-null    float64
 3   sibsp       891 non-null    int64
 4   parch       891 non-null    int64
 5   fare        891 non-null    float64
 6   class       891 non-null    category
 7   who         891 non-null    object
 8   adult_male  891 non-null    bool
 9   alive       891 non-null    object
 10  alone       891 non-null    bool
dtypes: bool(2), category(1), float64(2), int64(3), object(3)
memory usage: 58.6+ KB
```

[13]:
```
titanic_cleaned.isnull().sum()
```

[13]:
```
survived        0
sex             0
age           177
sibsp           0
parch           0
fare            0
class           0
who             0
adult_male      0
alive           0
alone           0
dtype: int64
```

[14]:
```
titanic_cleaned.corr(method='pearson')
```

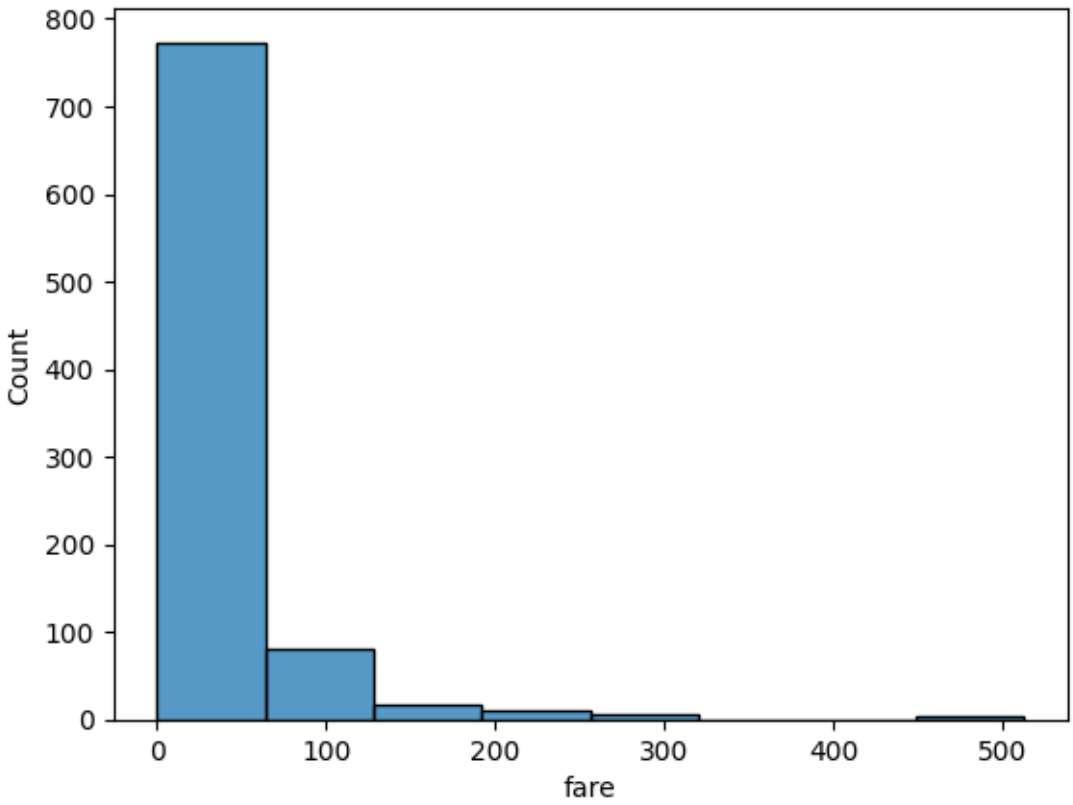
```
/var/folders/cs/hplqvnxd09bg_bgmf6zh8t3m0000gn/T/ipykernel_3389/2643438631.py:1:
FutureWarning: The default value of numeric_only in DataFrame.corr is
deprecated. In a future version, it will default to False. Select only valid
columns or specify the value of numeric_only to silence this warning.
  titanic_cleaned.corr(method='pearson')
```

[14]:

	survived	age	sibsp	parch	fare	adult_male	alone
survived	1.000000	-0.077221	-0.035322	0.081629	0.257307	-0.557080	-0.203367
age	-0.077221	1.000000	-0.308247	-0.189119	0.096067	0.280328	0.198270
sibsp	-0.035322	-0.308247	1.000000	0.414838	0.159651	-0.253586	-0.584471
parch	0.081629	-0.189119	0.414838	1.000000	0.216225	-0.349943	-0.583398
fare	0.257307	0.096067	0.159651	0.216225	1.000000	-0.182024	-0.271832
adult_male	-0.557080	0.280328	-0.253586	-0.349943	-0.182024	1.000000	0.404744
alone	-0.203367	0.198270	-0.584471	-0.583398	-0.271832	0.404744	1.000000

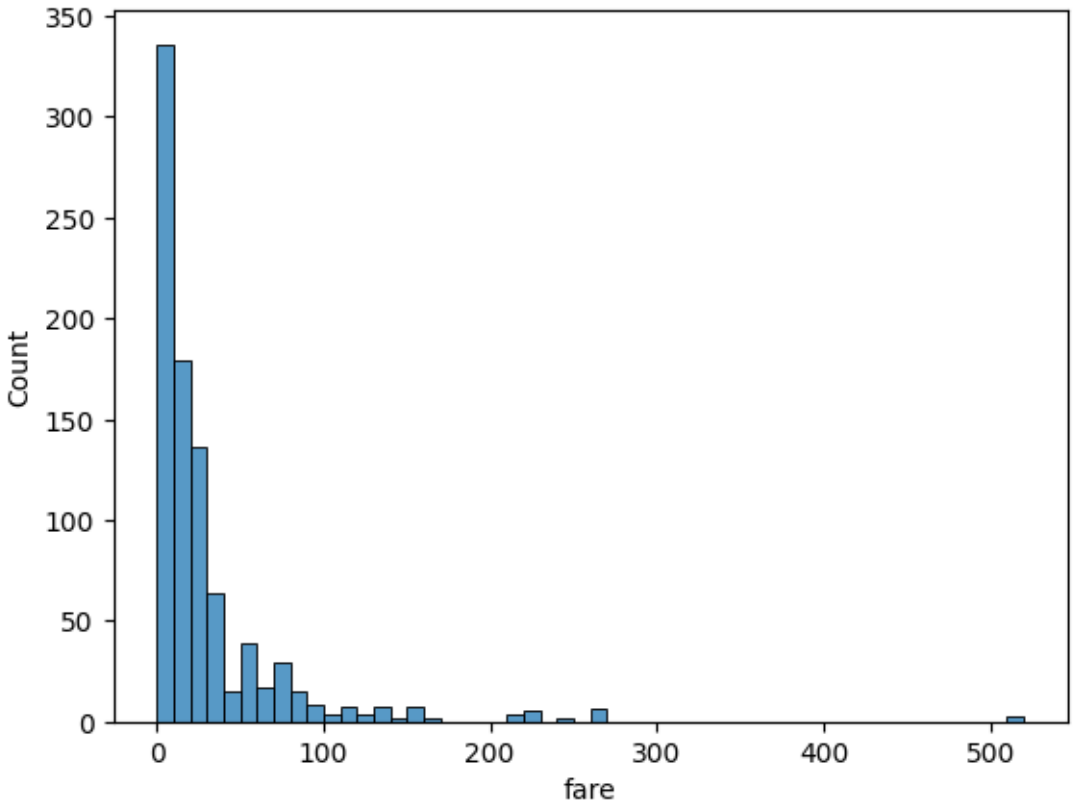
[15]:
```
sns.histplot(data=titanic,x="fare",bins=8)
```

[15]:
```
<Axes: xlabel='fare', ylabel='Count'>
```



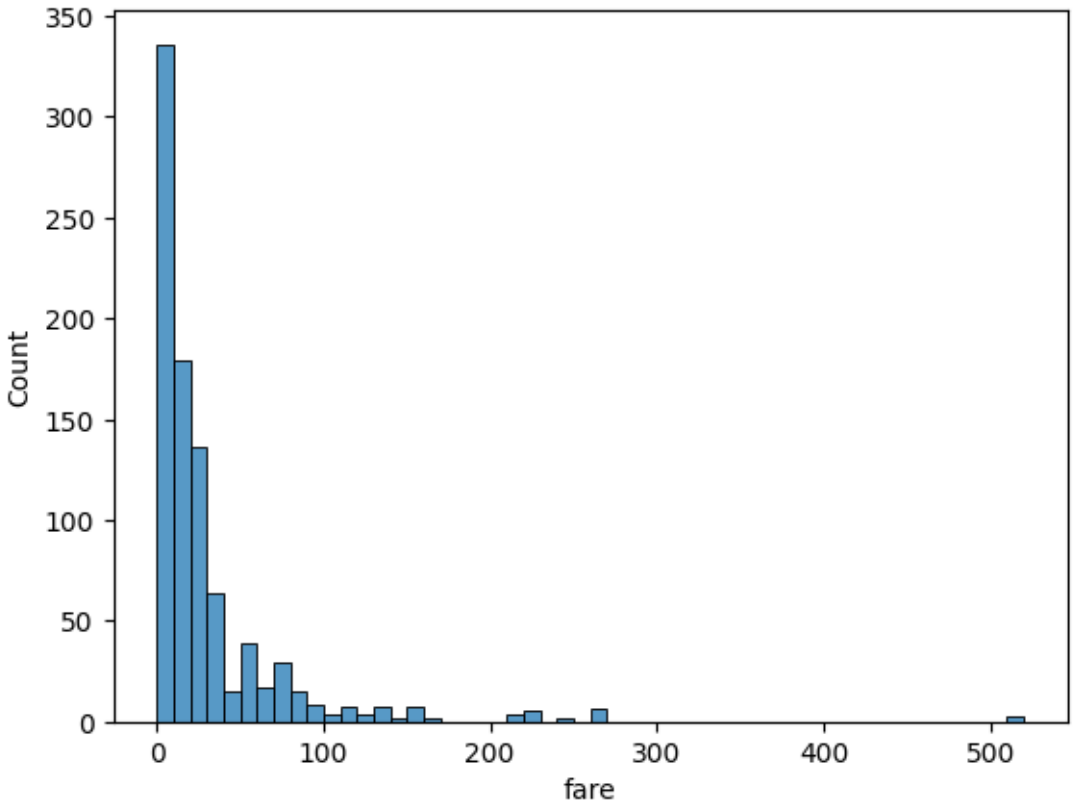
[16]:
```
sns.histplot(data=titanic,x="fare",binwidth=10)
```

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



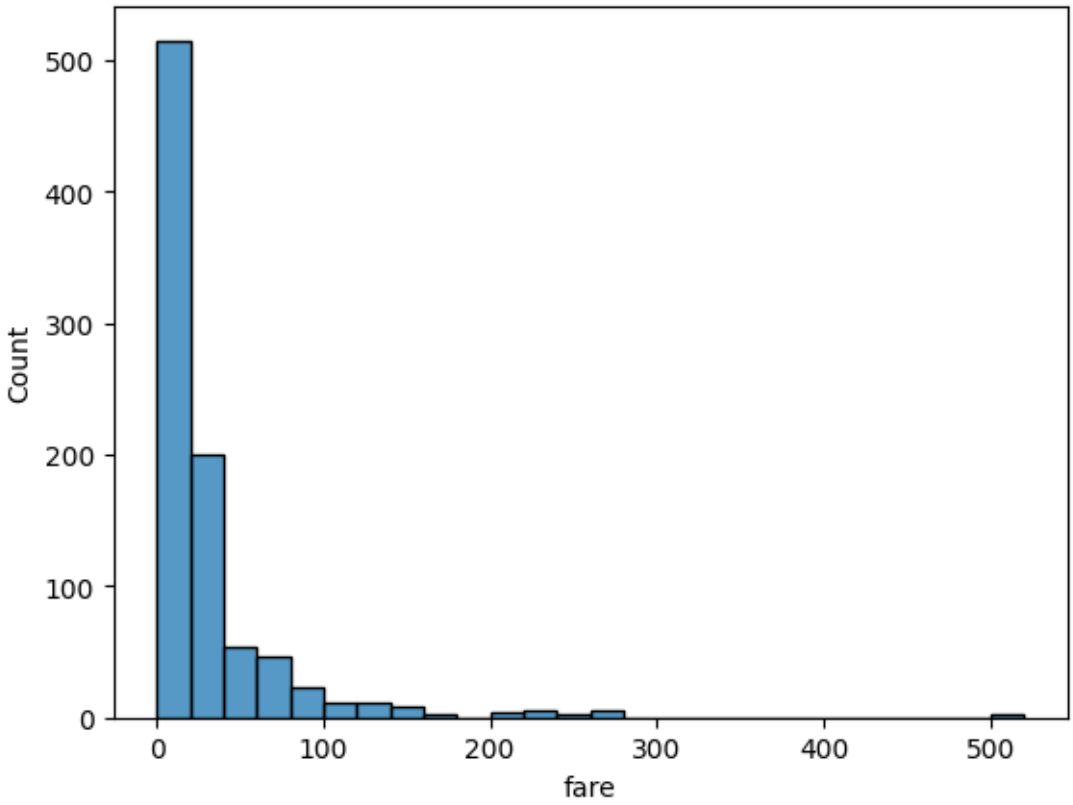
[17]:
```
sns.histplot(data=titanic,x="fare",bins=20,binwidth=10)
```

[17]:
```
<Axes: xlabel='fare', ylabel='Count'>
```



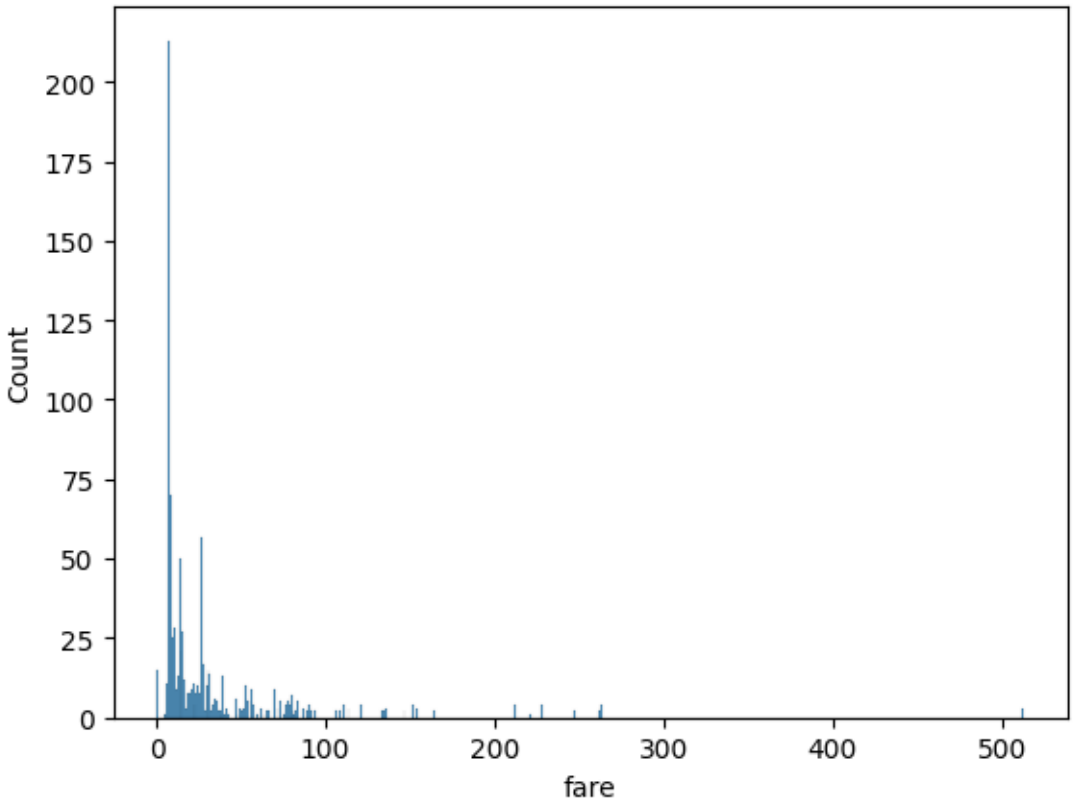
[18]:
```
sns.histplot(data=titanic,x="fare",binwidth=20)
```

[18]:
```
<Axes: xlabel='fare', ylabel='Count'>
```



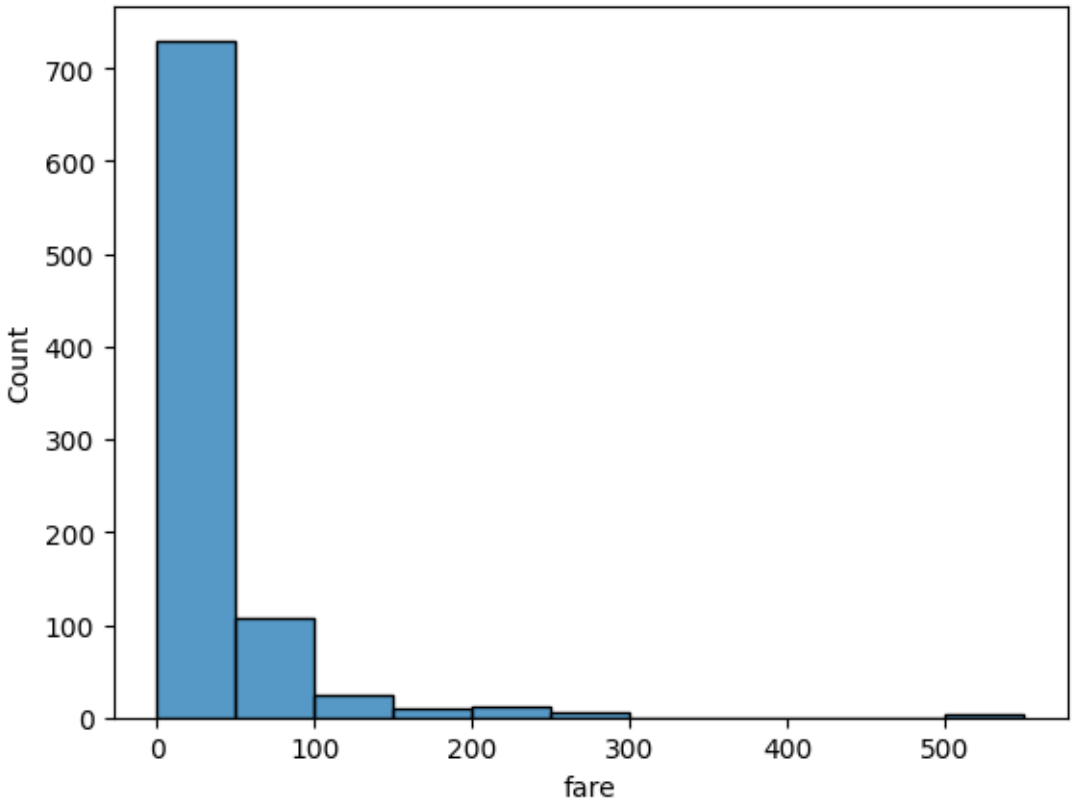
[19]:
```
sns.histplot(data=titanic,x="fare",binwidth=1)
```

[19]:
```
<Axes: xlabel='fare', ylabel='Count'>
```



[20]:
```
sns.histplot(data=titanic,x="fare", bins=20,binwidth=50)
```

[20]:
```
<Axes: xlabel='fare', ylabel='Count'>
```


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