## Data Visualization I

1. Use the inbuilt dataset 'titanic'. The dataset contains 891 rows and contains informationabout the passengers who boarded the unfortunate Titanic ship. Use the Seaborn library tosee if we can find any patterns in the data.

```
import numpy as np
import pandas as pd
import seaborn as sns
```

```
df=sns.load_dataset('titanic')
```

```
df.head(3)
```

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

```
df.shape
```

```
(891, 15)
```

```
df.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
```

```
df.describe()
```

| | 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 |

```
df.isnull().sum()
```

```
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
```

## Data preproccessing

### Drop deck & embarked_town column

```
df.drop(columns=['embark_town','deck'],axis=1,inplace=True)
```

```
df.isnull().sum()
```

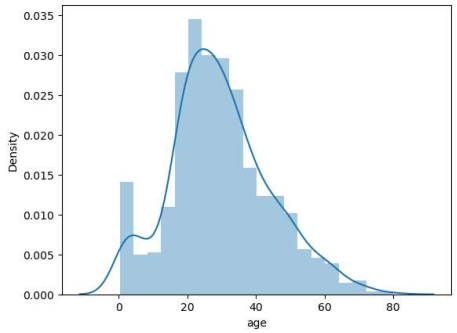
```
survived       0
pclass         0
sex            0
age          177
sibsp          0
parch          0
fare           0
embarked       2
class          0
who            0
adult_male     0
alive          0
alone          0
dtype: int64
```

### Impute values in age column

### check distribution of age column

```
sns.distplot(df['age'])
```

```
C:\Users\admin\AppData\Local\Temp\ipykernel_2984\3234920688.py:1: UserWarning:

`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

  sns.distplot(df['age'])
<Axes: xlabel='age', ylabel='Density'>
```




```
df['age'].skew()
```

```
0.38910778230082704
```

```
#age column has normal data distribution
```

```
df['age'].mean()
```

```
29.69911764705882
```

```
df['age'].fillna(df['age'].mean(),inplace=True)
```

```
df.isnull().sum()
```

```
survived      0
pclass        0
sex           0
age           0
sibsp         0
parch         0
fare          0
embarked      2
class         0
who           0
adult_male    0
alive         0
alone         0
dtype: int64
```

## impute values of embarked column

```
df['embarked'].nunique()
```

```
3
```

```
df['embarked'].unique()
```

```
array(['S', 'C', 'Q', nan], dtype=object)
```

```
df['embarked'].mode()
```

```
0    S
Name: embarked, dtype: object
```

```
df['embarked'].mode()[0]
```

```
'S'
```

```
df['embarked'].fillna(df['embarked'].mode()[0],inplace=True)
```

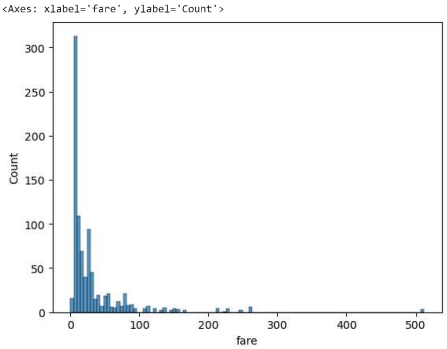
```
df.isnull().sum()
```

```
survived      0
pclass        0
sex           0
age           0
sibsp         0
parch         0
fare          0
embarked      0
class         0
who           0
adult_male    0
alive         0
alone         0
dtype: int64
```

## 2. Write a code to check how the price of the ticket (column name: 'fare') for each passenger is distributed by plotting a histogram.

```
sns.histplot(df['fare'])
```

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



```
df['fare'].skew()
```

```
4.787316519674893
```

```
# highly right distributated
```

```
df.nunique()
```

```
survived        2
pclass          3
sex             2
age            89
sibsp           7
parch           7
fare          248
embarked        3
class           3
who             3
adult_male      2
alive           2
alone           2
dtype: int64
```

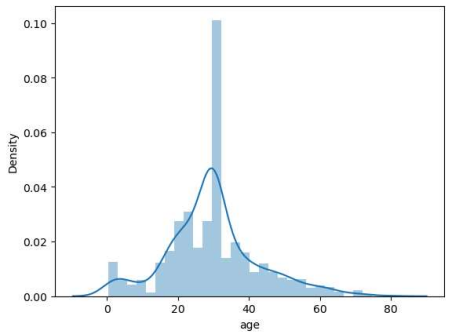
out of 13 columns age and fare are quanttative columns remaining are qualitative

- Quantitative= distplot, histplot,kdeplot,boxplot

Qualitative= countplot,pie chart

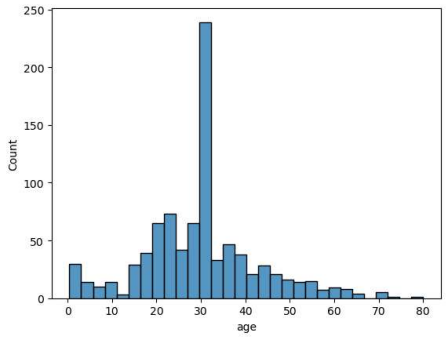
```
sns.distplot(df['age'])
```

```
C:\Users\admin\AppData\Local\Temp\ipykernel_2984\3234920688.py:1: UserWarning:

`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

  sns.distplot(df['age'])
<Axes: xlabel='age', ylabel='Density'>
```



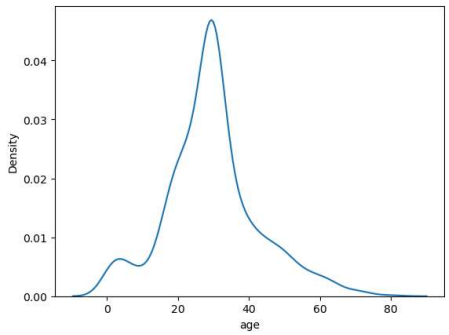

```
sns.histplot(df['age'])
```

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




```
sns.kdeplot(df['age'])
```

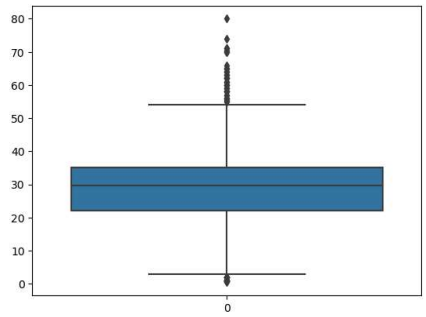
```
<Axes: xlabel='age', ylabel='Density'>
```




- age column is normally distributed

```
sns.boxplot(df['age'])
```

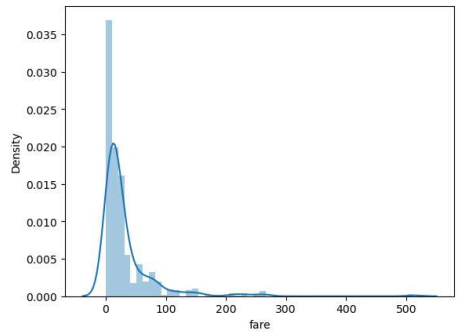
```
<Axes: >
```




there are outliers in age column

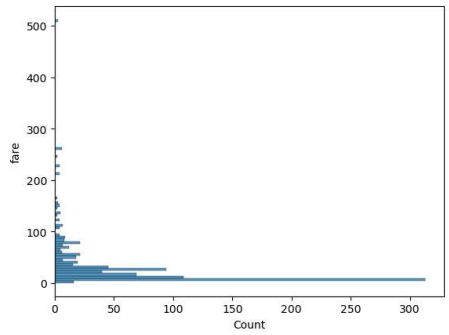
```
sns.distplot(df['fare'])
```

```
C:\Users\admin\AppData\Local\Temp\ipykernel_2984\1195996103.py:1: UserWarning:

`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

  sns.distplot(df['fare'])
<Axes: xlabel='fare', ylabel='Density'>
```



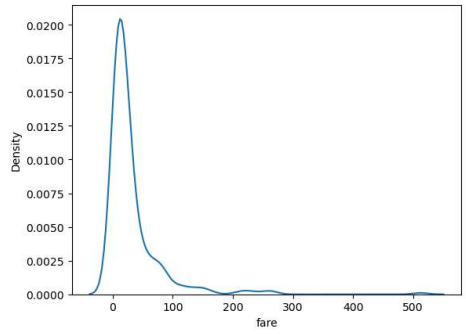

```
sns.histplot(y=df['fare'])
```

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




```
sns.kdeplot(df['fare'])
```

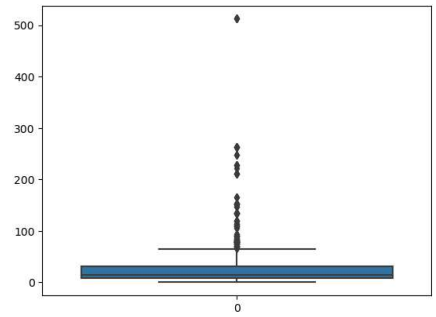
```
<Axes: xlabel='fare', ylabel='Density'>
```




fare column is rightly distributed

```
sns.boxplot(df['fare'])
```

```
<Axes: >
```




fare has also outliers

```
sns.countplot(x=df['survived'])
```

```
<Axes: xlabel='survived', ylabel='count'>
```