Written by: M.Danish Azeem Date: 20.11.2023 Email:m danishazeem365@gmail.com

pandas tips and tricks

```
import pandas as pd

# load dataset to work with
import seaborn as sns

# titanic dataset ko load karen
kashti = sns.load_dataset('titanic')

kashti.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
```

what every column means in kashti Dataset?

how many datasets are there in sns.load_dataset()?and how to fetch.

```
kashti.head()
```

```
   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   

     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  
```

```
kashti.tail()
```

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

       who  adult_male deck  embark_town alive  alone  
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  
```

```
# download phool ka data
phool = sns.load_dataset('iris')
phool.sample(100)
```

```
     sepal_length  sepal_width  petal_length  petal_width     species
62            6.0          2.2           4.0          1.0  versicolor
30            4.8          3.1           1.6          0.2      setosa
146           6.3          2.5           5.0          1.9   virginica
```

```
81            5.5          2.4           3.7          1.0  versicolor
132           6.4          2.8           5.6          2.2   virginica
..            ...          ...           ...          ...         ...
75            6.6          3.0           4.4          1.4  versicolor
59            5.2          2.7           3.9          1.4  versicolor
126           6.2          2.8           4.8          1.8   virginica
54            6.5          2.8           4.6          1.5  versicolor
31            5.4          3.4           1.5          0.4      setosa

[100 rows x 5 columns]
```

```
kashti.to_excel('kashti.xlsx')
```

```
phool.to_excel('phool.xlsx')
```

```
phool.to_excel('../day_7/phool.xlsx')
```

```
df = pd.read_excel('kashti.xlsx')
```

```
df.head()
```

```
   Unnamed: 0  survived  pclass     sex   age  sibsp  parch     fare 
embarked  \
0           0         0       3    male  22.0      1      0   7.2500 
S   
1           1         1       1  female  38.0      1      0  71.2833 
C   
2           2         1       3  female  26.0      0      0   7.9250 
S   
3           3         1       1  female  35.0      1      0  53.1000 
S   
4           4         0       3    male  35.0      0      0   8.0500 
S   

   class    who  adult_male deck  embark_town alive  alone  
0  Third    man        True  NaN  Southampton    no  False  
1  First  woman       False    C    Cherbourg   yes  False  
2  Third  woman       False  NaN  Southampton   yes   True  
3  First  woman       False    C  Southampton   yes  False  
4  Third    man        True  NaN  Southampton    no   True  
```

```
df.columns
```

```
Index(['Unnamed: 0', 'survived', 'pclass', 'sex', 'age', 'sibsp', 
'parch',
       'fare', 'embarked', 'class', 'who', 'adult_male', 'deck', 
'embark_town',
       'alive', 'alone'],
      dtype='object')
```

```
df['embark_town']
```

```
0      Southampton
1        Cherbourg
2      Southampton
3      Southampton
4      Southampton
          ...
886    Southampton
887    Southampton
888    Southampton
889      Cherbourg
890     Queenstown
Name: embark_town, Length: 891, dtype: object
```

```
df[['embark_town', 'survived']]
```

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

[891 rows x 2 columns]
```

```
df.embark_town.unique()
```

```
array(['Southampton', 'Cherbourg', 'Queenstown', nan], dtype=object)
```

what is nan, kitna importent hy or is ka hona ya na hna kitna zaroori ha?

```
df.deck.unique()
```

```
array([nan, 'C', 'E', 'G', 'D', 'A', 'B', 'F'], dtype=object)
```

```
df.describe()
```

```
       Unnamed: 0    survived      pclass         age       sibsp
parch  \
count  891.000000  891.000000  891.000000  714.000000  891.000000
891.000000
mean   445.000000    0.383838    2.308642   29.699118    0.523008
```

```
0.381594
std    257.353842    0.486592    0.836071   14.526497    1.102743
0.806057
min      0.000000    0.000000    1.000000    0.420000    0.000000
0.000000
25%    222.500000    0.000000    2.000000   20.125000    0.000000
0.000000
50%    445.000000    0.000000    3.000000   28.000000    0.000000
0.000000
75%    667.500000    1.000000    3.000000   38.000000    1.000000
0.000000
max    890.000000    1.000000    3.000000   80.000000    8.000000
6.000000

             fare
count  891.000000
mean    32.204208
std     49.693429
min      0.000000
25%      7.910400
50%     14.454200
75%     31.000000
max    512.329200
```

```
df = df.drop(columns='Unnamed: 0')
```

```
df
```

```
     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]
```

```
df.groupby(['sex']).describe().T
```

```
sex                    female        male
survived count  314.000000  577.000000
         mean     0.742038    0.188908
         std      0.438211    0.391775
         min      0.000000    0.000000
         25%      0.000000    0.000000
         50%      1.000000    0.000000
         75%      1.000000    0.000000
         max      1.000000    1.000000
pclass   count  314.000000  577.000000
         mean     2.159236    2.389948
         std      0.857290    0.813580
         min      1.000000    1.000000
         25%      1.000000    2.000000
         50%      2.000000    3.000000
         75%      3.000000    3.000000
         max      3.000000    3.000000
age      count  261.000000  453.000000
         mean    27.915709   30.726645
         std     14.110146   14.678201
         min      0.750000    0.420000
         25%     18.000000   21.000000
         50%     27.000000   29.000000
         75%     37.000000   39.000000
         max     63.000000   80.000000
sibsp    count  314.000000  577.000000
         mean     0.694268    0.429809
         std      1.156520    1.061811
         min      0.000000    0.000000
         25%      0.000000    0.000000
         50%      0.000000    0.000000
         75%      1.000000    0.000000
         max      8.000000    8.000000
parch    count  314.000000  577.000000
         mean     0.649682    0.235702
         std      1.022846    0.612294
         min      0.000000    0.000000
         25%      0.000000    0.000000
         50%      0.000000    0.000000
         75%      1.000000    0.000000
         max      6.000000    5.000000
fare     count  314.000000  577.000000
         mean    44.479818   25.523893
         std     57.997698   43.138263
         min      6.750000    0.000000
         25%     12.071875    7.895800
         50%     23.000000   10.500000
         75%     55.000000   26.550000
         max    512.329200  512.329200
```

```
df.count()
```

```
survived       891
pclass         891
sex            891
age            714
sibsp          891
parch          891
fare           891
embarked       889
class          891
who            891
adult_male     891
deck           203
embark_town    889
alive          891
alone          891
dtype: int64
```

```
df.groupby(["sex"]).count()
```

```
        survived  pclass  age  sibsp  parch  fare  embarked  class
who  \
sex

female       314     314  261    314    314   314       312    314
314
male         577     577  453    577    577   577       577    577
577
```

```
        adult_male  deck  embark_town  alive  alone
sex                                                
female         314    97          312    314    314
male           577   106          577    577    577
```

```
df.groupby(["sex", ("survived")]).count()
```

```
                 pclass  age  sibsp  parch  fare  embarked  class  who  
\
sex    survived                                                          

female 0             81   64     81     81    81        81     81   81   

       1            233  197    233    233   233       231    233  233   

male   0            468  360    468    468   468       468    468  468   

       1            109   93    109    109   109       109    109  109   


                 adult_male  deck  embark_town  alive  alone
sex    survived                                             
female 0                 81     6           81     81     81
       1                233    91          231    233    233
male   0                468    61          468    468    468
       1                109    45          109    109    109
```

```
df.groupby(["survived", "sex"])["survived"].count()
```

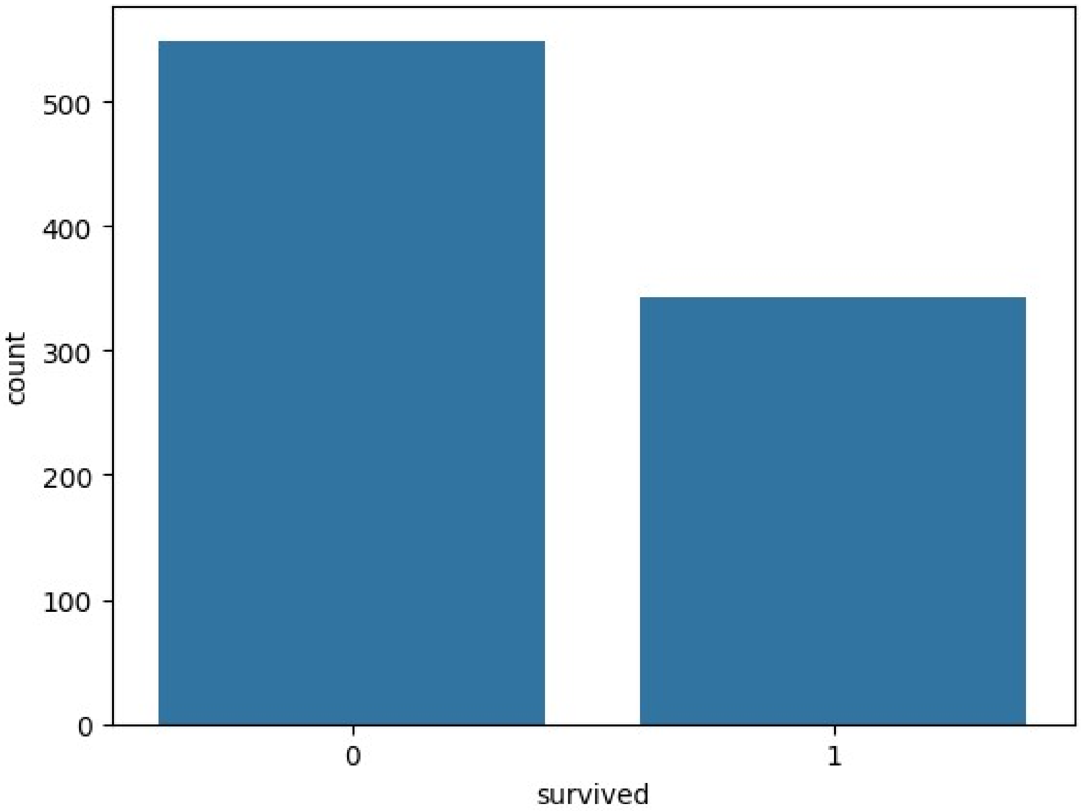
```
survived  sex   
0         female     81
          male      468
1         female    233
          male      109
Name: survived, dtype: int64
```

```
df.groupby(["survived", "sex", "who"])["who"].count()
```

```
survived  sex     who  
0         female  child     15
                  woman     66
          male    child     19
                  man      449
1         female  child     28
                  woman    205
          male    child     21
                  man       88
Name: who, dtype: int64
```

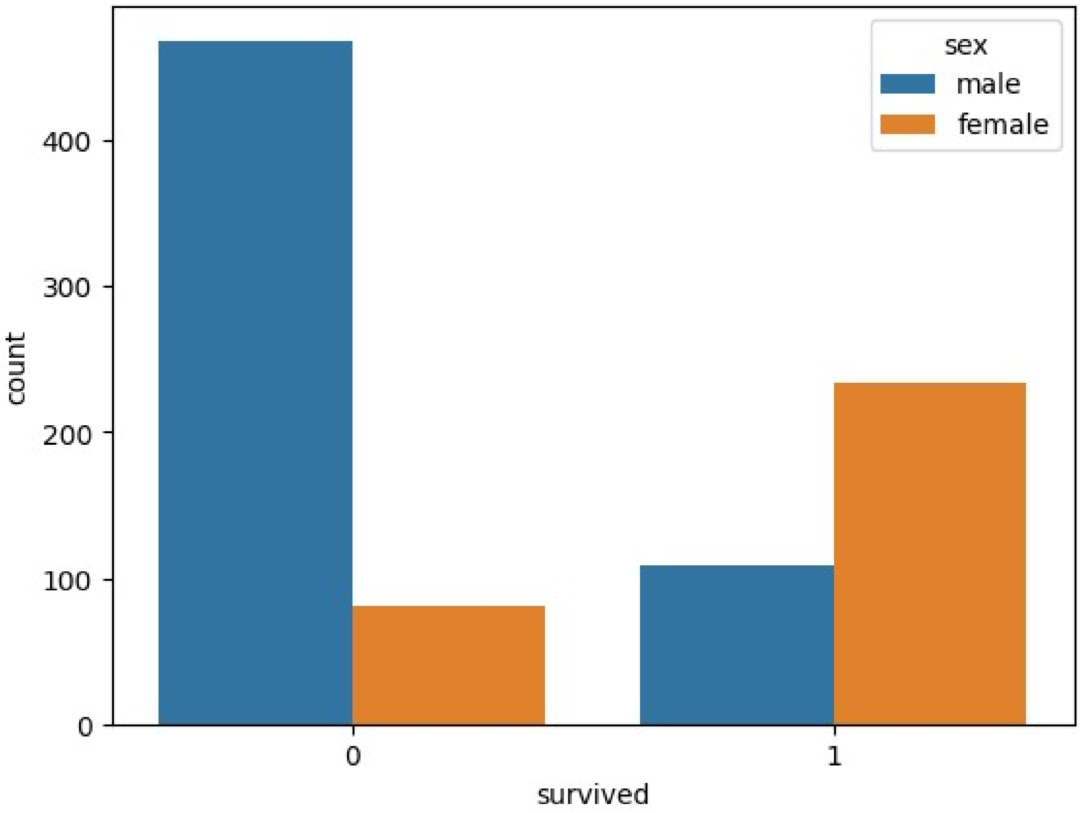
```
sns.countplot(x="survived", data=df)
```

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



```
sns.countplot(x="survived", hue="sex", data=df)
```

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


sex
male
female
400
300
200
100
0
count
0
1
survived