⚡ NAME 'MRITYUNJAY JOSHI'

⬇⬇⬇⬇⬇⬇⬇⬇⬇⬇⬇⬇⬇⬇⬇⬇⬇⬇⬇⬇⬇⬇⬇⬇⬇⬇⬇⬇⬇⬇⬇⬇⬇⬇⬇⬇

🟥 PROJECT - 'Analyze Deaths in Titanic Ship Dataset':

import Libraries_

```
In [1]:  import numpy as N
         import pandas as pd
         import matplotlib.pyplot as plt
         import seaborn as sns
```

Load Dataset From Seaborn Library & Read :

```
In [2]:  titanic=sns.load_dataset('Titanic')
```

```
In [3]:  titanic.index
```

```
Out[3]:  RangeIndex(start=0, stop=891, step=1)
```

```
In [4]:  titanic.size
```

```
Out[4]:  13365
```

```
In [5]:  titanic.shape
```

```
Out[5]:  (891, 15)
```

```
In [6]:  titanic.columns
```

```
Out[6]: Index(['survived', 'pclass', 'sex', 'age', 'sibsp', 'parch', 'fare',
               'embarked', 'class', 'who', 'adult_male', 'deck', 'embark_town',
               'alive', 'alone'],
              dtype='object')
```

In [7]: `titanic.head(2)`

Out[7]:

	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

In [8]: `titanic.tail(3)`

Out[8]:

	survived	pclass	sex	age	sibsp	parch	fare	embarked	class	who	adult_male	deck	embark_town	alive	alone
888	0	3	female	NaN	1	2	23.45	S	Third	woman	False	NaN	Southampton	no	False
889	1	1	male	26.0	0	0	30.00	C	First	man	True	C	Cherbourg	yes	True
890	0	3	male	32.0	0	0	7.75	Q	Third	man	True	NaN	Queenstown	no	True

Data Cleaning & Other Operations _

In [9]: `pd.isnull(titanic).sum()`

```
Out[9]:  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
```

```
In [10]:  titanic.drop(['deck'],axis=1,inplace=True)
```

```
In [11]:  pd.isnull(titanic).sum()
```

```
Out[11]:  survived          0
          pclass            0
          sex               0
          age             177
          sibsp             0
          parch             0
          fare              0
          embarked          2
          class             0
          who               0
          adult_male        0
          embark_town       2
          alive             0
          alone             0
          dtype: int64
```

```
In [12]:  titanic.dropna(inplace=True)
```

```
In [13]:  titanic.to_csv('Titanic New.csv')
```

```
In [14]:  titanic.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 712 entries, 0 to 890
Data columns (total 14 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   survived     712 non-null    int64
 1   pclass       712 non-null    int64
 2   sex          712 non-null    object
 3   age          712 non-null    float64
 4   sibsp        712 non-null    int64
 5   parch        712 non-null    int64
 6   fare         712 non-null    float64
 7   embarked     712 non-null    object
 8   class        712 non-null    object
 9   who          712 non-null    object
 10  adult_male   712 non-null    bool
 11  embark_town  712 non-null    object
 12  alive        712 non-null    object
 13  alone        712 non-null    bool
dtypes: bool(2), float64(2), int64(4), object(6)
memory usage: 73.7+ KB
```

In [15]: `titanic.dtypes`

Out[15]:
```
survived         int64
pclass           int64
sex             object
age            float64
sibsp            int64
parch            int64
fare           float64
embarked        object
class           object
who             object
adult_male        bool
embark_town     object
alive           object
alone             bool
dtype: object
```

In [16]: `titanic['age']=titanic['age'].astype(int)`

In [17]: `titanic['age'].dtypes`

```
Out[17]:  dtype('int32')
```

```
In [18]:  pd.unique(titanic['embarked'])
```

```
Out[18]:  array(['S', 'C', 'Q'], dtype=object)
```

```
In [19]:  pd.unique(titanic['class'])
```

```
Out[19]:  array(['Third', 'First', 'Second'], dtype=object)
```

```
In [20]:  titanic.nunique()
```

```
Out[20]:  survived        2
          pclass          3
          sex             2
          age            71
          sibsp           6
          parch           7
          fare          219
          embarked        3
          class           3
          who             3
          adult_male      2
          embark_town     3
          alive           2
          alone           2
          dtype: int64
```

```
In [21]:  titanic.rename(columns={'embark_town':'E Town'},inplace=True)
```

```
In [22]:  titanic.columns
```

```
Out[22]:  Index(['survived', 'pclass', 'sex', 'age', 'sibsp', 'parch', 'fare',
                 'embarked', 'class', 'who', 'adult_male', 'E Town', 'alive', 'alone'],
                dtype='object')
```

```
In [23]:  titanic.describe()
```

	survived	pclass	age	sibsp	parch	fare
count	712.000000	712.000000	712.000000	712.000000	712.000000	712.000000
mean	0.404494	2.240169	29.622191	0.514045	0.432584	34.567251
std	0.491139	0.836854	14.502891	0.930692	0.854181	52.938648
min	0.000000	1.000000	0.000000	0.000000	0.000000	0.000000
25%	0.000000	1.000000	20.000000	0.000000	0.000000	8.050000
50%	0.000000	2.000000	28.000000	0.000000	0.000000	15.645850
75%	1.000000	3.000000	38.000000	1.000000	1.000000	33.000000
max	1.000000	3.000000	80.000000	5.000000	6.000000	512.329200

```python
titanic.describe(include=object)
```

	sex	embarked	class	who	E Town	alive
count	712	712	712	712	712	712
unique	2	3	3	3	3	2
top	male	S	Third	man	Southampton	no
freq	453	554	355	413	554	424

Visualization _

```python
sns.set(style="darkgrid")
```

```python
titanic.head(3)
```

	survived	pclass	sex	age	sibsp	parch	fare	embarked	class	who	adult_male	E Town	alive	alone
0	0	3	male	22	1	0	7.2500	S	Third	man	True	Southampton	no	False
1	1	1	female	38	1	0	71.2833	C	First	woman	False	Cherbourg	yes	False
2	1	3	female	26	0	0	7.9250	S	Third	woman	False	Southampton	yes	True

```python
titanic.groupby(['sex'])['sex'].count().reset_index(name='count')
```

	sex	count
0	female	259
1	male	453

```python
ax=sns.countplot(x='sex',palette='rocket',data=titanic)
for bars in ax.containers:
    ax.bar_label(bars)
plt.xticks(rotation=90)
plt.title('count of sex')
plt.show()
```

count of sex
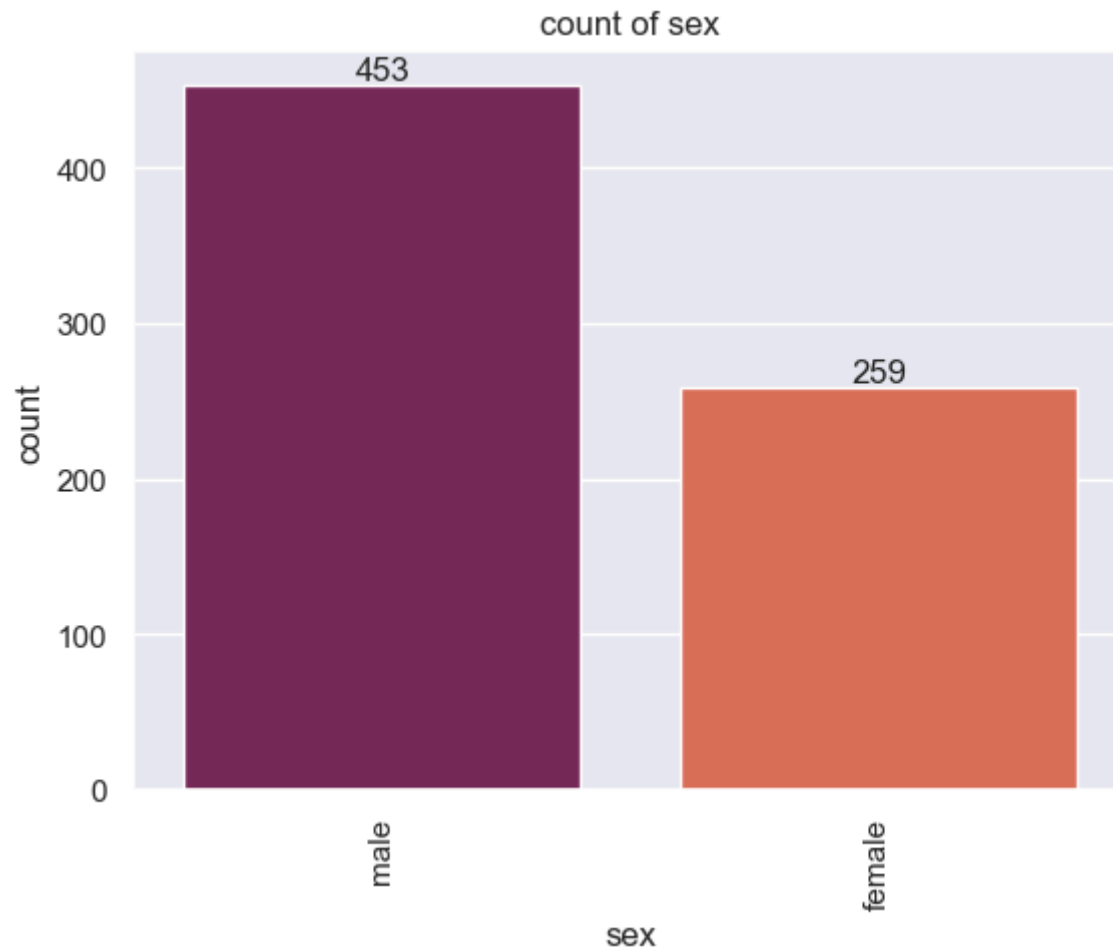
There were a total of 453 males and 259 females on the Titanic …

```
In [29]:  gr=titanic.groupby(['alive','sex'])['sex'].count().reset_index(name='count').sort_values(by='count',ascending=False)
```

```
In [30]:  gr
```

	alive	sex	count
1	no	male	360
2	yes	female	195
3	yes	male	93
0	no	female	64

```python
color=['red','green']
```

```python
ax=sns.barplot(x='sex',hue='alive',y='count',data=gr,palette=color)
for p in ax.patches:
    ax.annotate(format(p.get_height(), '.0f'),
                (p.get_x() + p.get_width() / 2., p.get_height()),
                ha = 'center', va = 'center',
                xytext = (0, 8),
                textcoords = 'offset points')
plt.xticks(rotation=90)
plt.title('How many Peoples are alive based on Gender')
plt.show()
```

How many Peoples are alive based on Gender
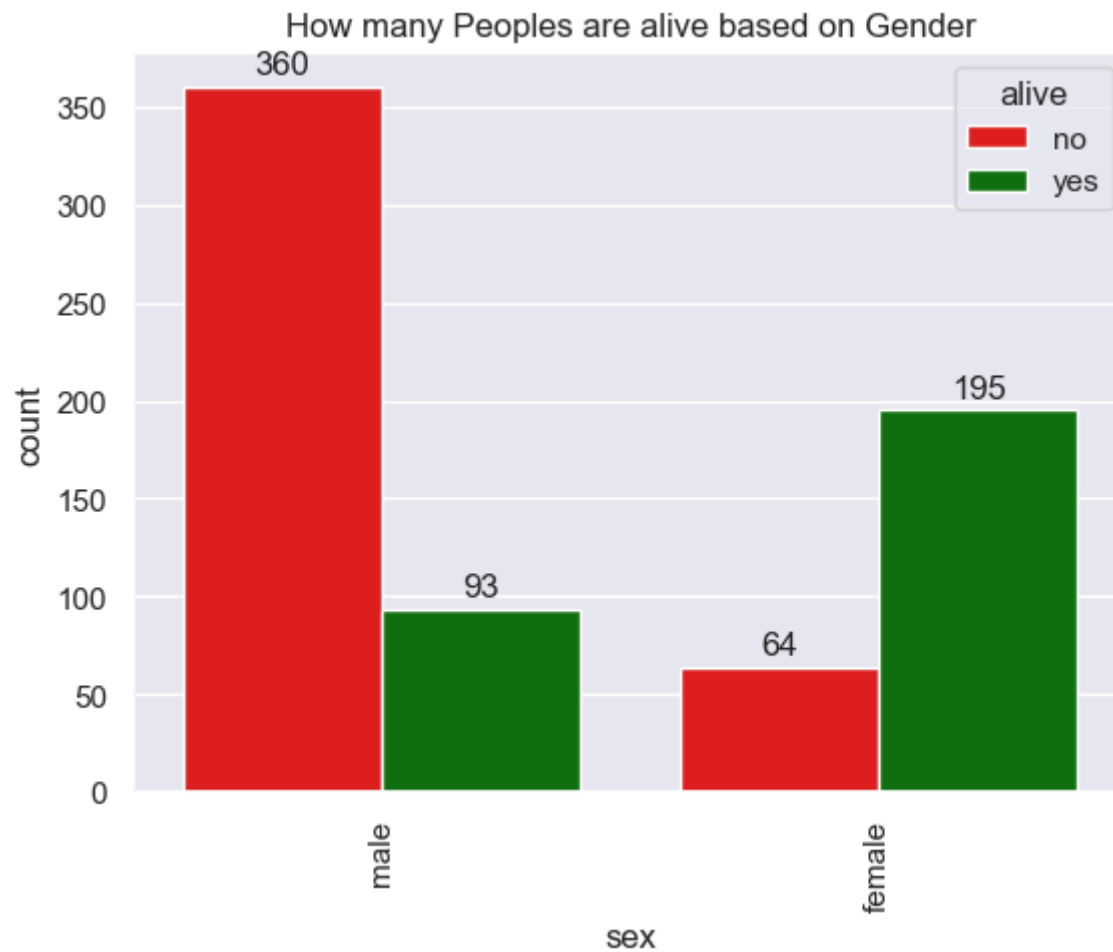
360 males and 64 females have lost their lives ...

```
In [33]: titanic.head(3)
```

Out[33]:

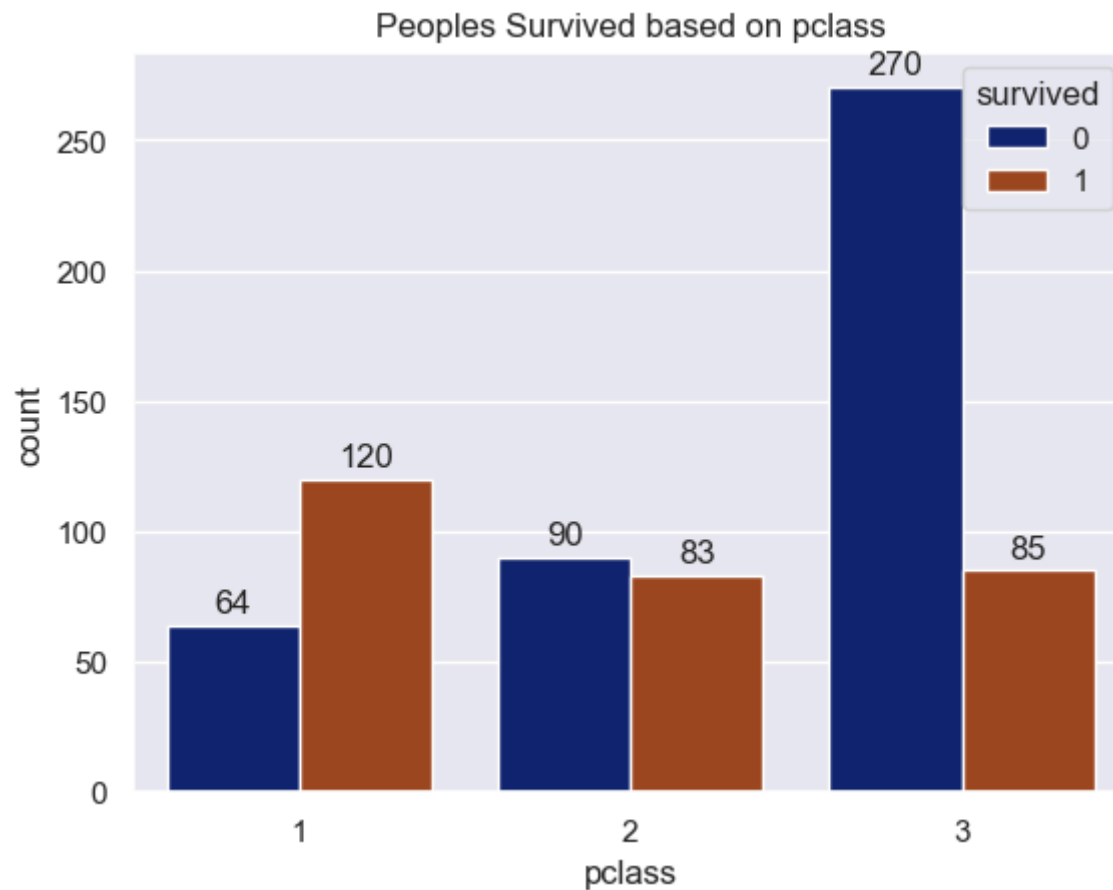|	survived	pclass	sex	age	sibsp	parch	fare	embarked	class	who	adult_male	E Town	alive	alone
0	0	3	male	22	1	0	7.2500	S	Third	man	True	Southampton	no	False
1	1	1	female	38	1	0	71.2833	C	First	woman	False	Cherbourg	yes	False
2	1	3	female	26	0	0	7.9250	S	Third	woman	False	Southampton	yes	True

```
In [34]:  gr=titanic.groupby(['pclass','sex'])['alive'].count().reset_index(name='count').sort_values(by='pclass',ascending=False)
```

```
In [35]:  gr
```

Out[35]:

	pclass	sex	count
4	3	female	102
5	3	male	253
2	2	female	74
3	2	male	99
0	1	female	83
1	1	male	101

```
In [36]:  ax=sns.barplot(x='sex',hue='pclass',y='count',data=gr,palette='winter')
          for p in ax.patches:
              ax.annotate(format(p.get_height(), '.0f'),
                          (p.get_x() + p.get_width() / 2., p.get_height()),
                          ha = 'center', va = 'center',
                          xytext = (0, 8),
                          textcoords = 'offset points')
          plt.title('Count Of Sex Based On PClass')
          plt.xticks(rotation=45)
          plt.show()
```

Count Of Sex Based On PClass
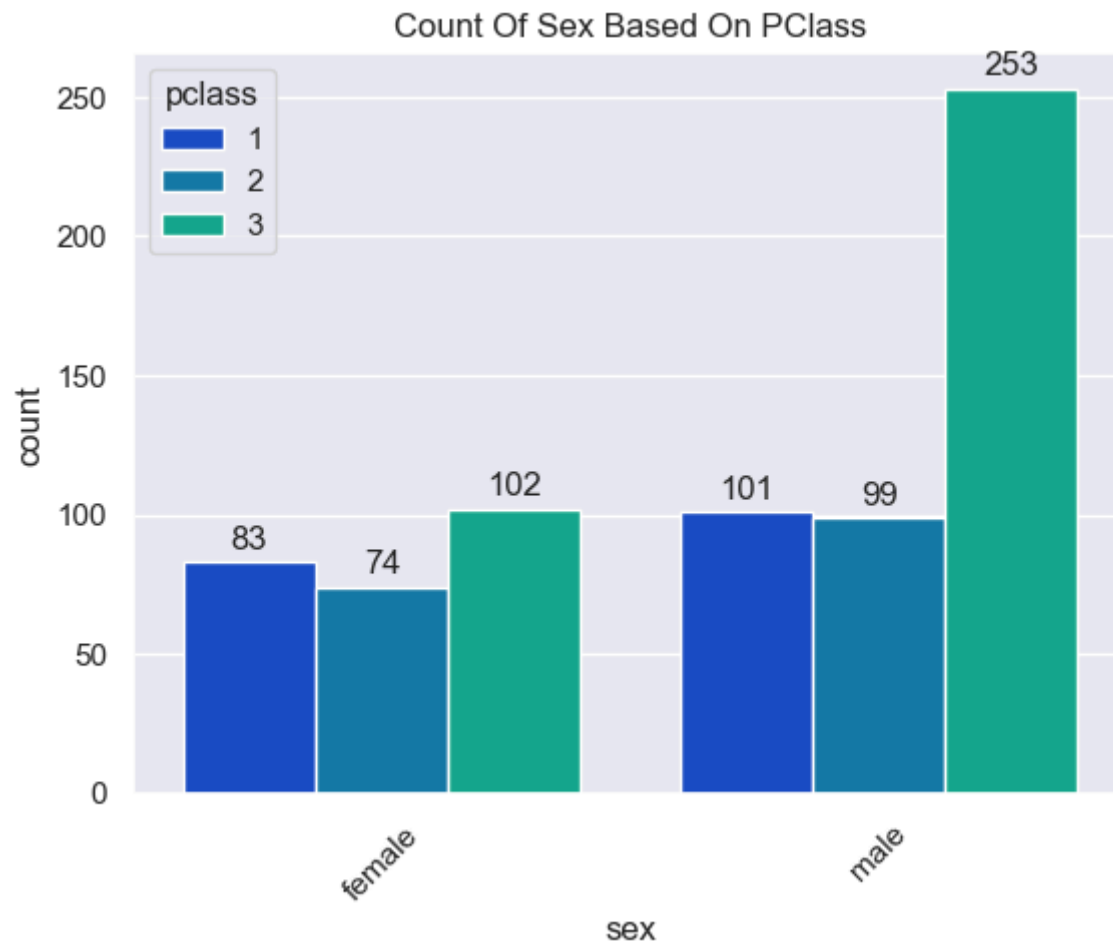

```
In [ ]:

In [37]: gr=titanic.groupby(['pclass','survived'])['survived'].count().reset_index(name='count').sort_values(by='pclass',ascending=True)

In [38]: gr
```

	pclass	survived	count
0	1	0	64
1	1	1	120
2	2	0	90
3	2	1	83
4	3	0	270
5	3	1	85

```python
ax=sns.barplot(x='pclass',y='count',hue='survived',data=gr,palette='dark')
for p in ax.patches:
    ax.annotate(format(p.get_height(), '.0f'),
                (p.get_x() + p.get_width() / 2., p.get_height()),
                ha = 'center', va = 'center',
                xytext = (0, 8),
                textcoords = 'offset points')
plt.title('Peoples Survived based on pclass')
plt.show();
```

Peoples Survived based on pclass

Most people (270 peoples) have lost their lives in pclass 3 ...

```
In [40]: titanic.head(3)
```

Out[40]:

	survived	pclass	sex	age	sibsp	parch	fare	embarked	class	who	adult_male	E Town	alive	alone
0	0	3	male	22	1	0	7.2500	S	Third	man	True	Southampton	no	False
1	1	1	female	38	1	0	71.2833	C	First	woman	False	Cherbourg	yes	False
2	1	3	female	26	0	0	7.9250	S	Third	woman	False	Southampton	yes	True

```
In [41]: sns.boxplot(y='age',data=titanic,palette='summer')
```

`<Axes: ylabel='age'>`
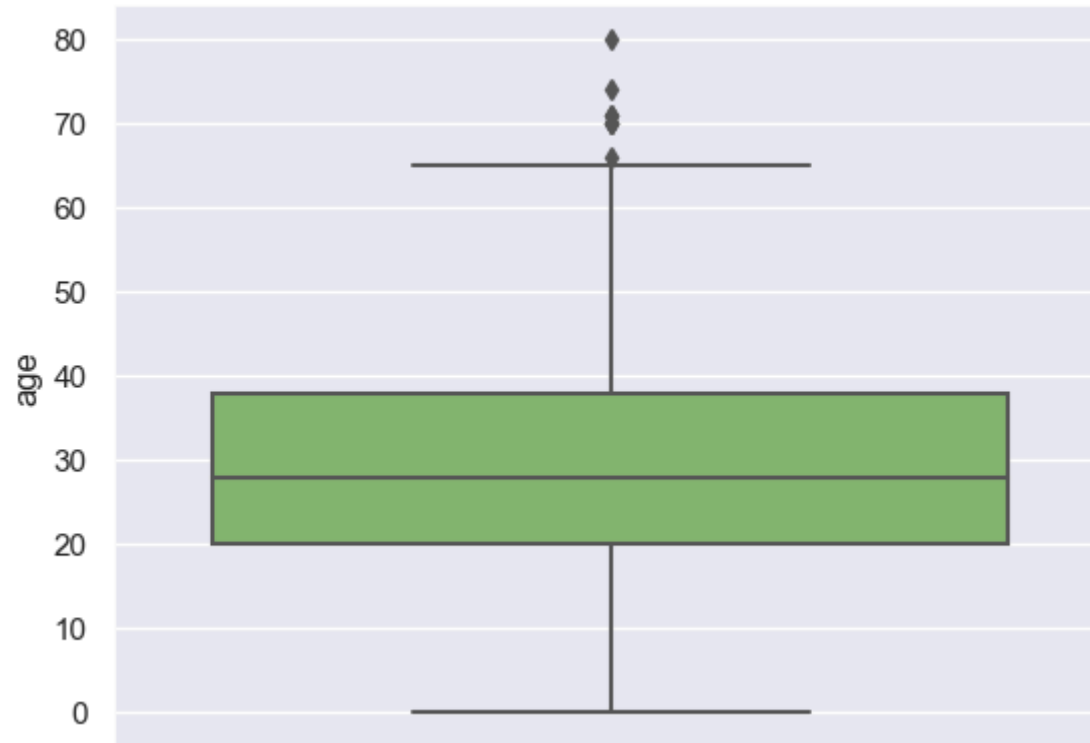
`sns.histplot(titanic['age'],kde=True,color='red')`

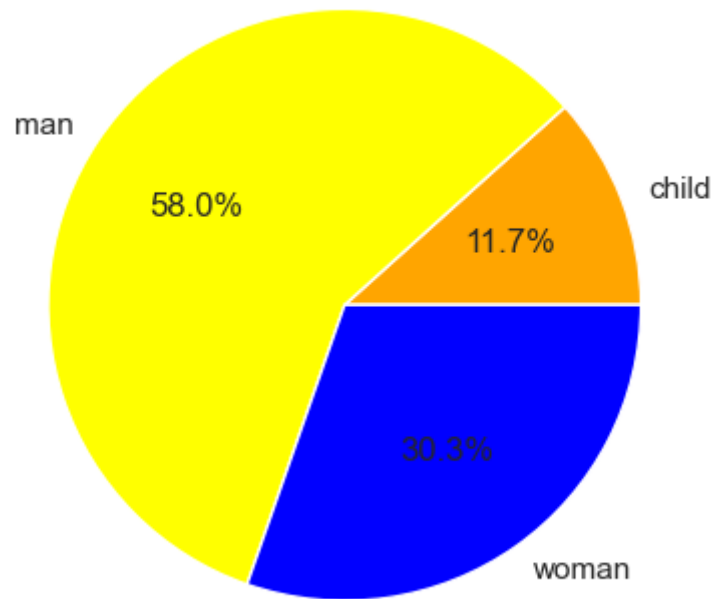`<Axes: xlabel='age', ylabel='Count'>`
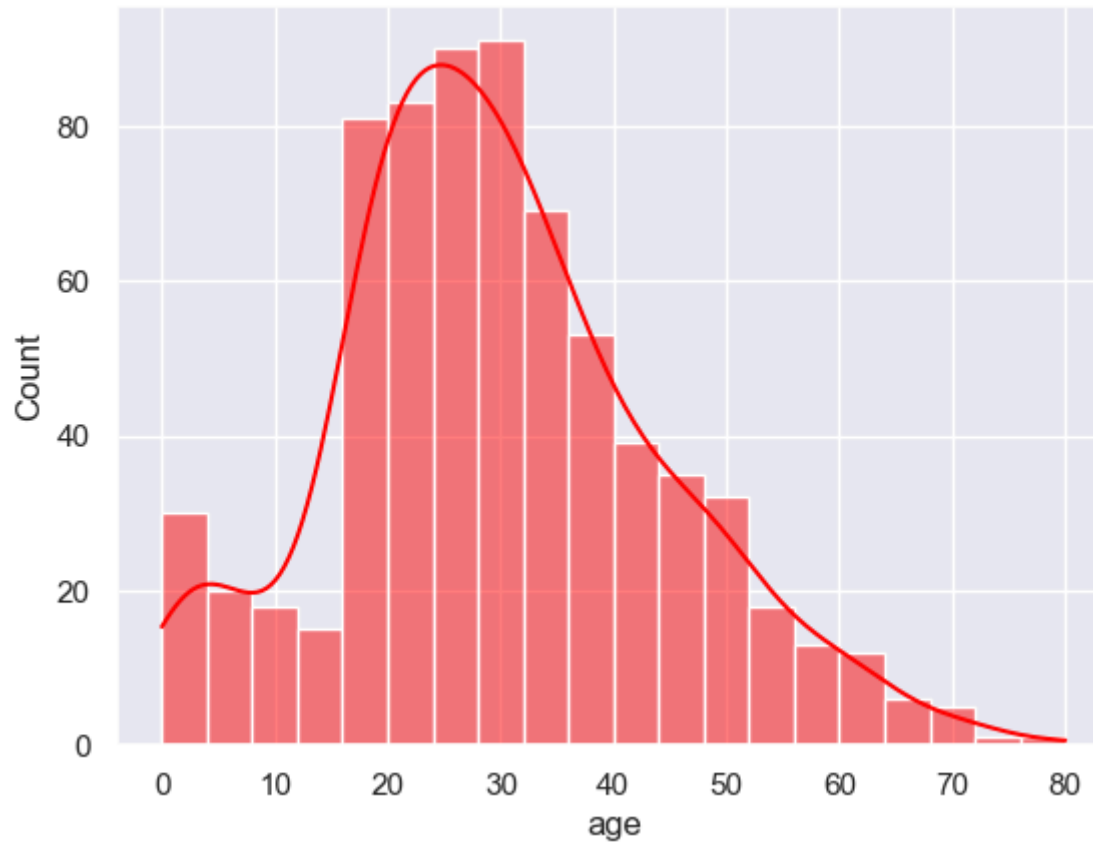
Most people are between 18 to 36 years ...

```
In [43]: gr=titanic.groupby(['who'])['who'].count().reset_index(name='count')
```

```
In [44]: beautiful=['orange','yellow','blue']
```

```
In [45]: plt.pie(x='count',labels='who',autopct='%1.1f%%',data=gr,colors=beautiful);
```

`gr=titanic.groupby(['who','survived'])['survived'].count().reset_index(name='count').sort_values(by='survived',ascending=True)`

`gr`

	who	survived	count
0	child	0	34
2	man	0	341
4	woman	0	49
1	child	1	49
3	man	1	72
5	woman	1	167

```
In [48]: ax=sns.barplot(x='who',y='count',hue='survived',data=gr,palette='spring')
         for p in ax.patches:
             ax.annotate(format(p.get_height(), '.0f'),
                         (p.get_x() + p.get_width() / 2., p.get_height()),
                         ha = 'center', va = 'center',
                         xytext = (0, 8),
                         textcoords = 'offset points')
         plt.xticks(rotation=90)
         plt.xlabel('who')
         plt.ylabel('count of who')
         plt.title('Peoples Survived based on pclass');
         plt.show()
```
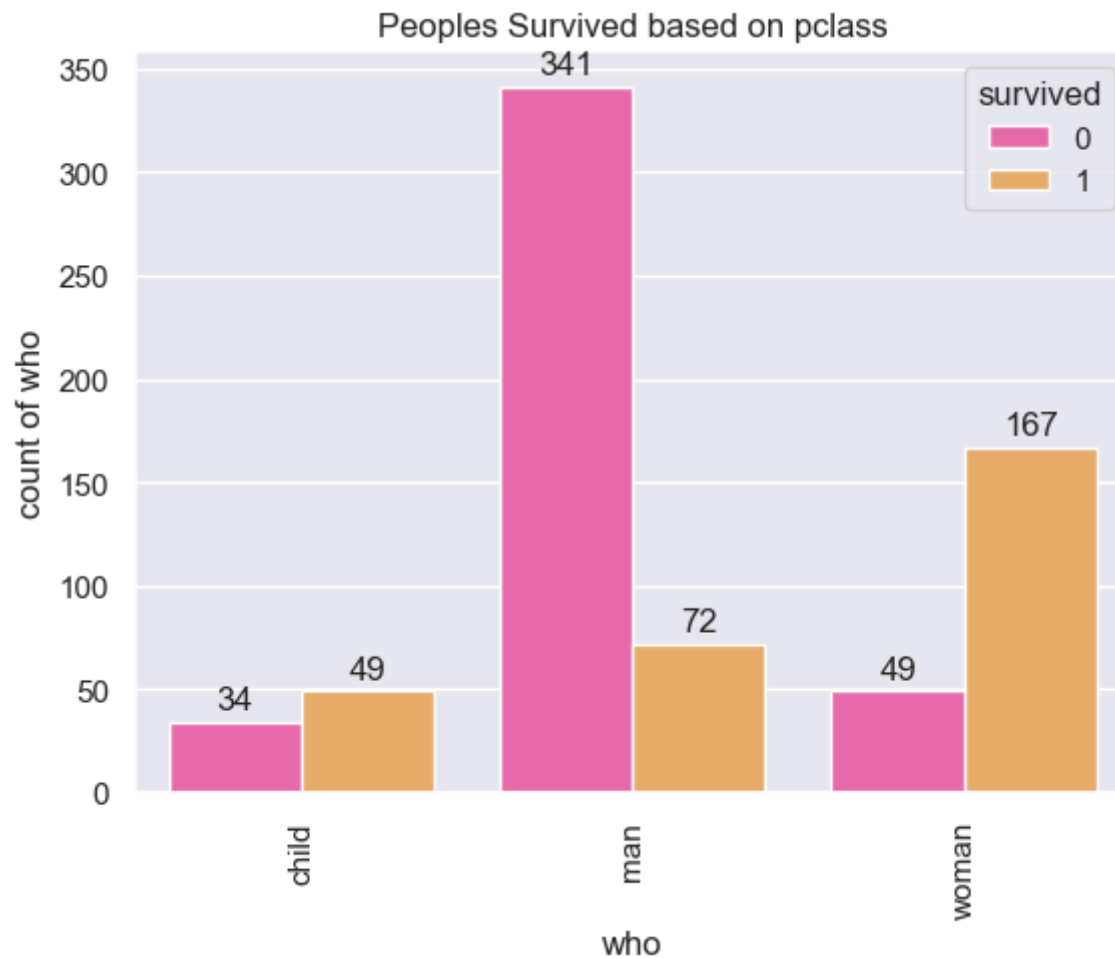
34 childs have lost their lives ...

```
In [49]: titanic.head(2)
```

Out[49]:

	survived	pclass	sex	age	sibsp	parch	fare	embarked	class	who	adult_male	E Town	alive	alone
0	0	3	male	22	1	0	7.2500	S	Third	man	True	Southampton	no	False
1	1	1	female	38	1	0	71.2833	C	First	woman	False	Cherbourg	yes	False

```
In [50]: titanic.groupby(['adult_male'])['adult_male'].aggregate(['count'])
```

Out[50]:

	count
adult_male	
False	299
True	413

```
In [51]: ax=sns.countplot(y='adult_male',data=titanic,palette='plasma')
         for bars in ax.containers:
             ax.bar_label(bars)
         plt.xlabel('ADULT_MALES')
         plt.ylabel('Counts')
         plt.title('Count of adult_males')
         plt.xticks(rotation=0)
         plt.show()
```

Count of adult_males
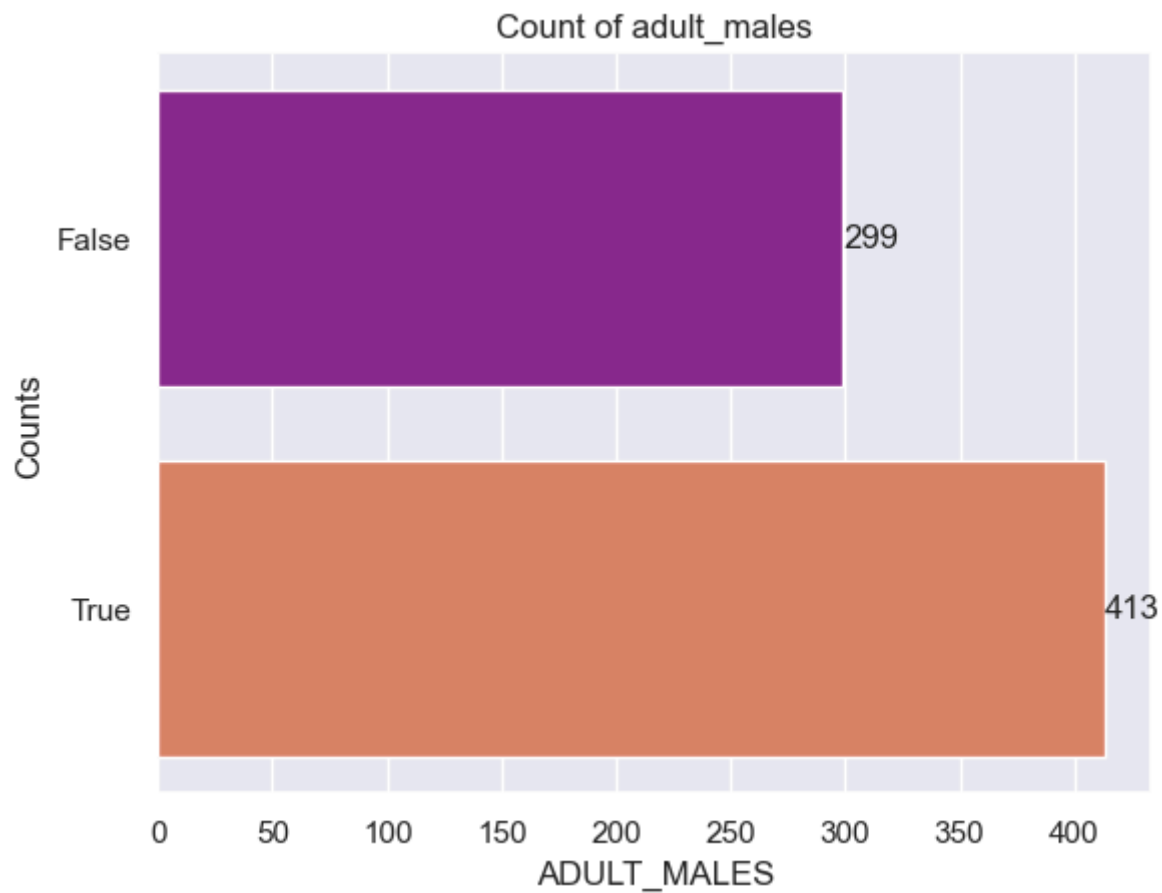
```
In [52]: gr=titanic.groupby(['adult_male','alive'])['alive'].count().reset_index(name='count')
```

```
In [53]: gr
```

Out[53]:

	adult_male	alive	count
0	False	no	83
1	False	yes	216
2	True	no	341
3	True	yes	72

```
In [54]: x=['brown','orange']
```

```
In [55]: ax=sns.barplot(x='adult_male',y='count',hue='alive',data=gr,palette=x)
         for p in ax.patches:
             ax.annotate(format(p.get_height(), '.0f'),
                         (p.get_x() + p.get_width() / 2., p.get_height()),
                         ha = 'center', va = 'center',
                         xytext = (0, 8),
                         textcoords = 'offset points')
         plt.xticks(rotation=90)
         plt.title('how many adult males alive & not');
         plt.show()
```
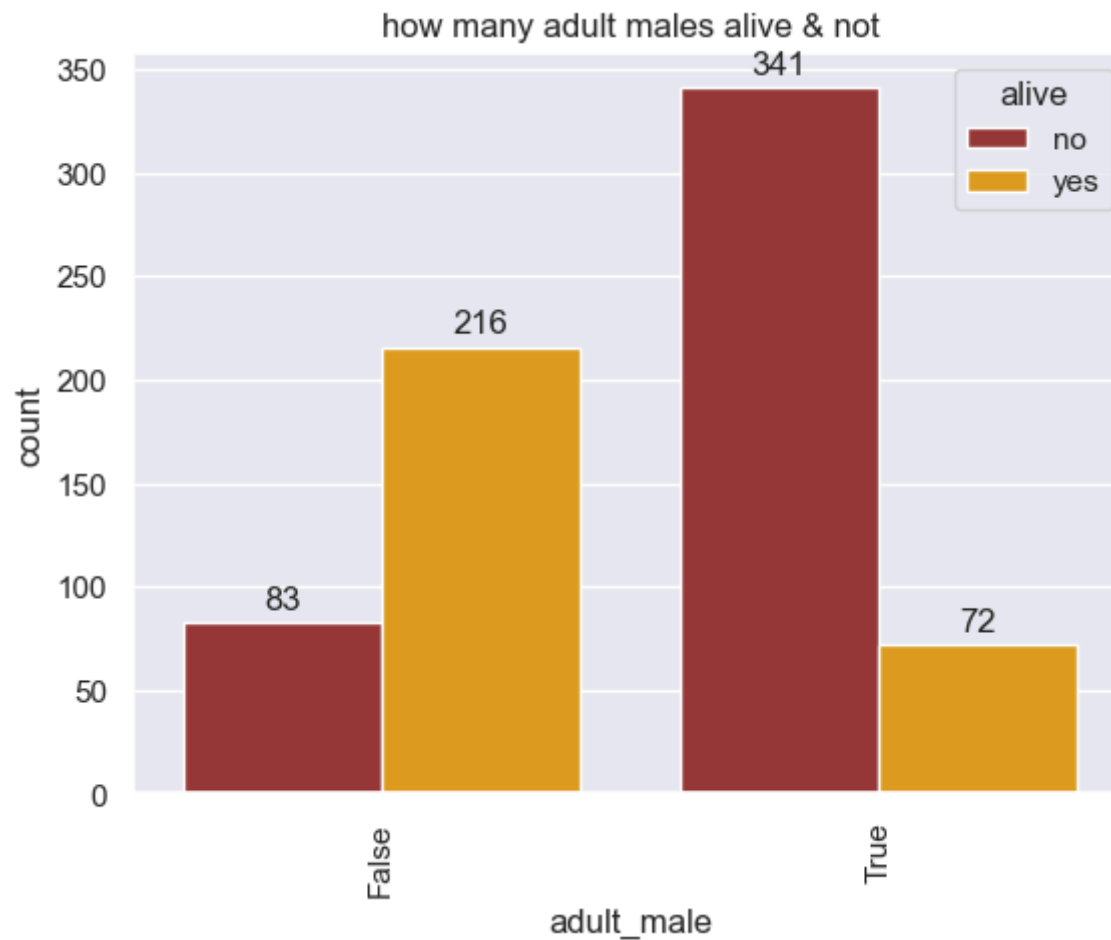
341 adult males have lost their lifes...

```
In [56]: titanic.head(2)
```

Out[56]:

	survived	pclass	sex	age	sibsp	parch	fare	embarked	class	who	adult_male	E Town	alive	alone
0	0	3	male	22	1	0	7.2500	S	Third	man	True	Southampton	no	False
1	1	1	female	38	1	0	71.2833	C	First	woman	False	Cherbourg	yes	False

```
In [57]: gr=titanic.groupby(['who']).get_group('child').groupby(['alive','sex'])['sex'].count().reset_index(name='count')
```

```
In [58]: gr
```

Out[58]:

	alive	sex	count
0	no	female	15
1	no	male	19
2	yes	female	28
3	yes	male	21

```
In [59]: ax=sns.barplot(x='sex',hue='alive',y='count',data=gr,palette='cubehelix')
         for p in ax.patches:
             ax.annotate(format(p.get_height(), '.0f'),
                         (p.get_x() + p.get_width() / 2., p.get_height()),
                         ha = 'center', va = 'center',
                         xytext = (0, 8),
                         textcoords = 'offset points')
         plt.xticks(rotation=90)
         plt.title('how many child lost lifes');
         plt.show()
```

how many child lost lifes
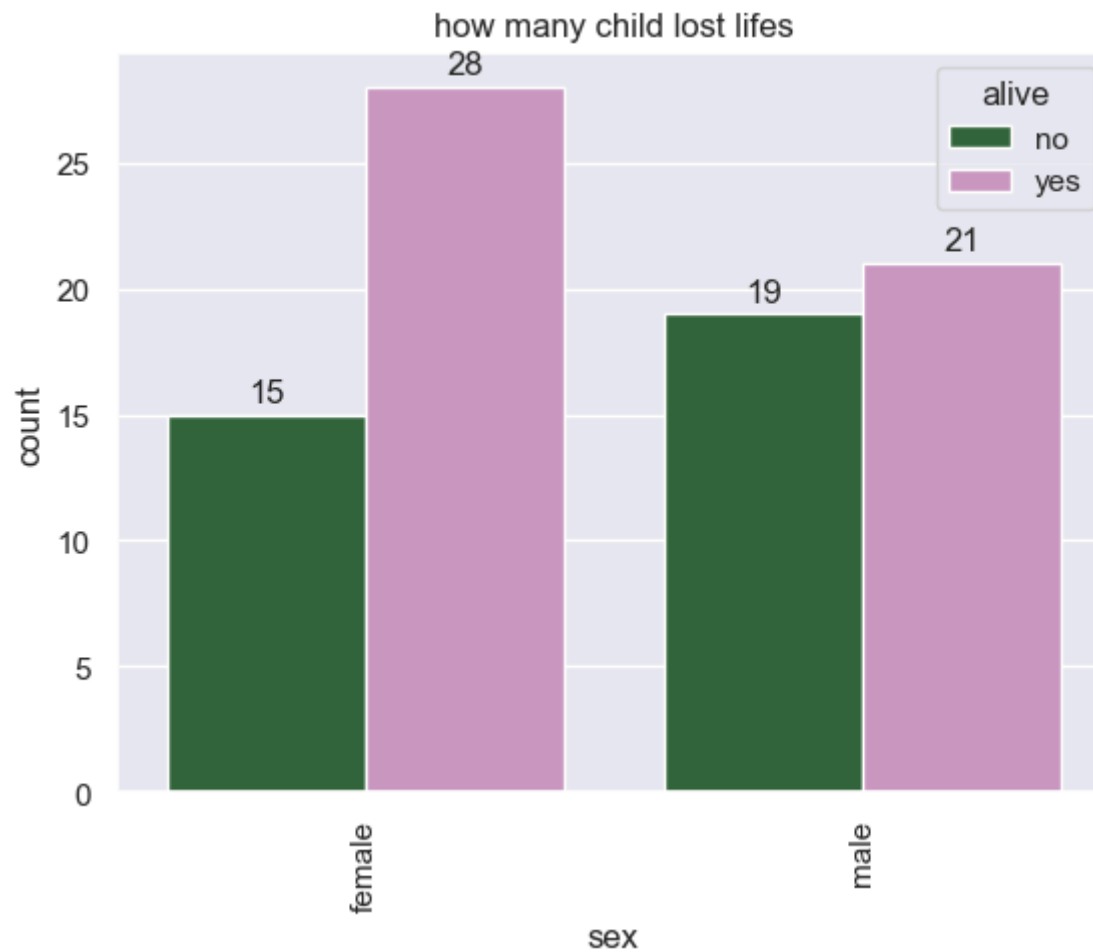
In [60]: grouping=titanic.groupby(['sex']).get_group('male').groupby(['who']).get_group('child').groupby(['alive'])['alive'].count().reset
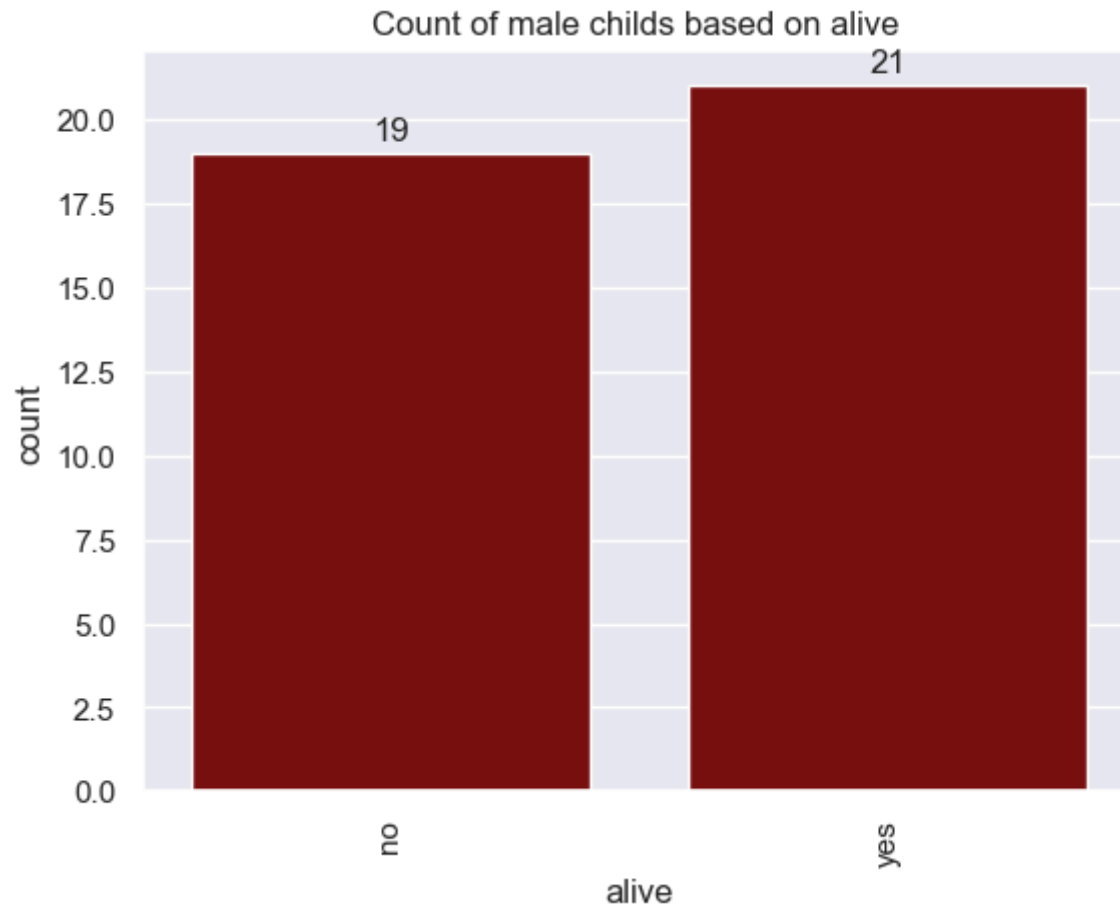
In [61]: grouping

Out[61]:

	alive	count
0	no	19
1	yes	21

```python
ax=sns.barplot(x='alive',y='count',data=grouping,color='darkred')
for p in ax.patches:
    ax.annotate(format(p.get_height(), '.0f'),
                (p.get_x() + p.get_width() / 2., p.get_height()),
                ha = 'center', va = 'center',
                xytext = (0, 8),
                textcoords = 'offset points')
plt.xticks(rotation=90)
plt.title('Count of male childs based on alive ');
plt.show()
```

Count of male childs based on alive

19 male childs have lost their lives...

```
In [63]: grouping=titanic.groupby(['sex']).get_group('female').groupby(['who']).get_group('child').groupby(['alive'])['alive'].count().res
```

```
In [64]: grouping
```

Out[64]:

	alive	count
0	no	15
1	yes	28

```
In [65]: ax=sns.barplot(x='alive',y='count',data=grouping,color='blue')
         for p in ax.patches:
             ax.annotate(format(p.get_height(), '.0f'),
                         (p.get_x() + p.get_width() / 2., p.get_height()),
                         ha = 'center', va = 'center',
                         xytext = (0, 8),
                         textcoords = 'offset points')
         plt.xticks(rotation=90)
         plt.title('Count of Fmale childs based on alive ');
         plt.show()
```

Count of Fmale childs based on alive
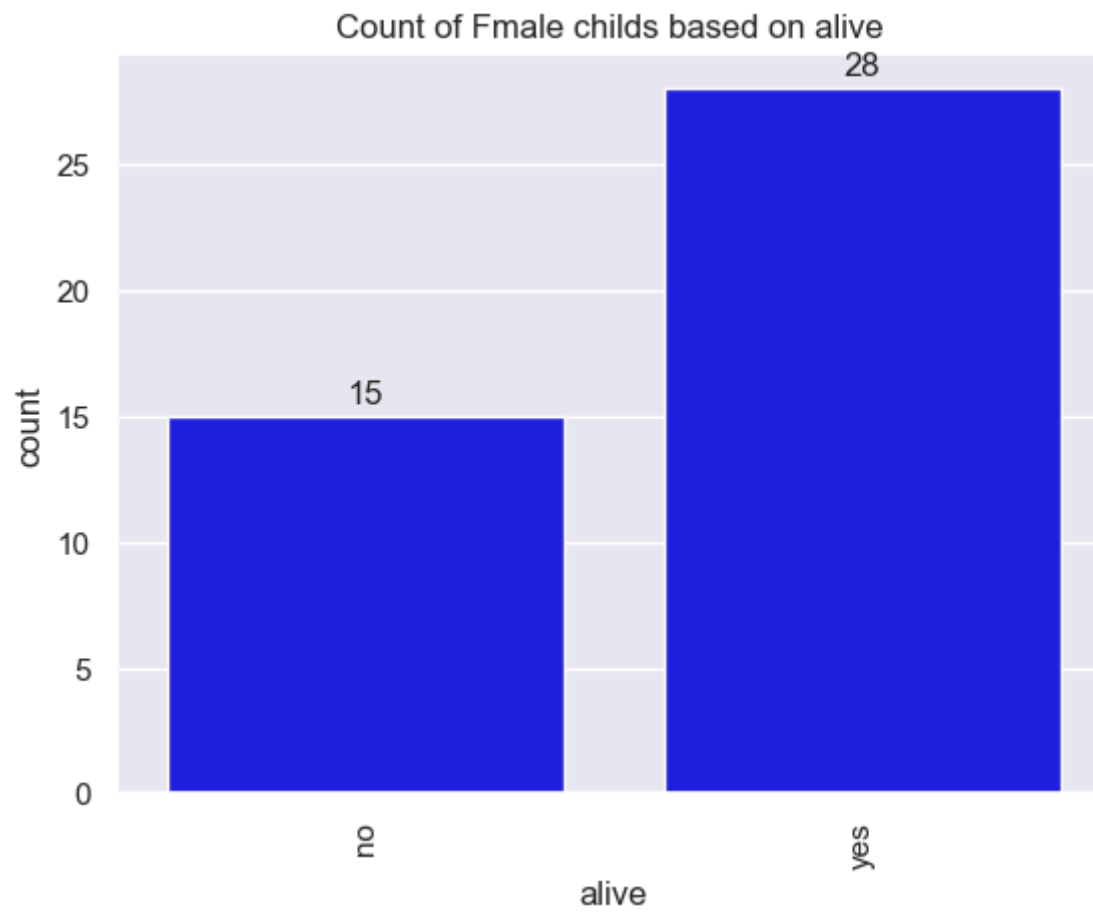
15 female childs have lost their lives ...

`sns.distplot(titanic['fare'],color='red')`

Out[66]: `<Axes: xlabel='fare', ylabel='Density'>`
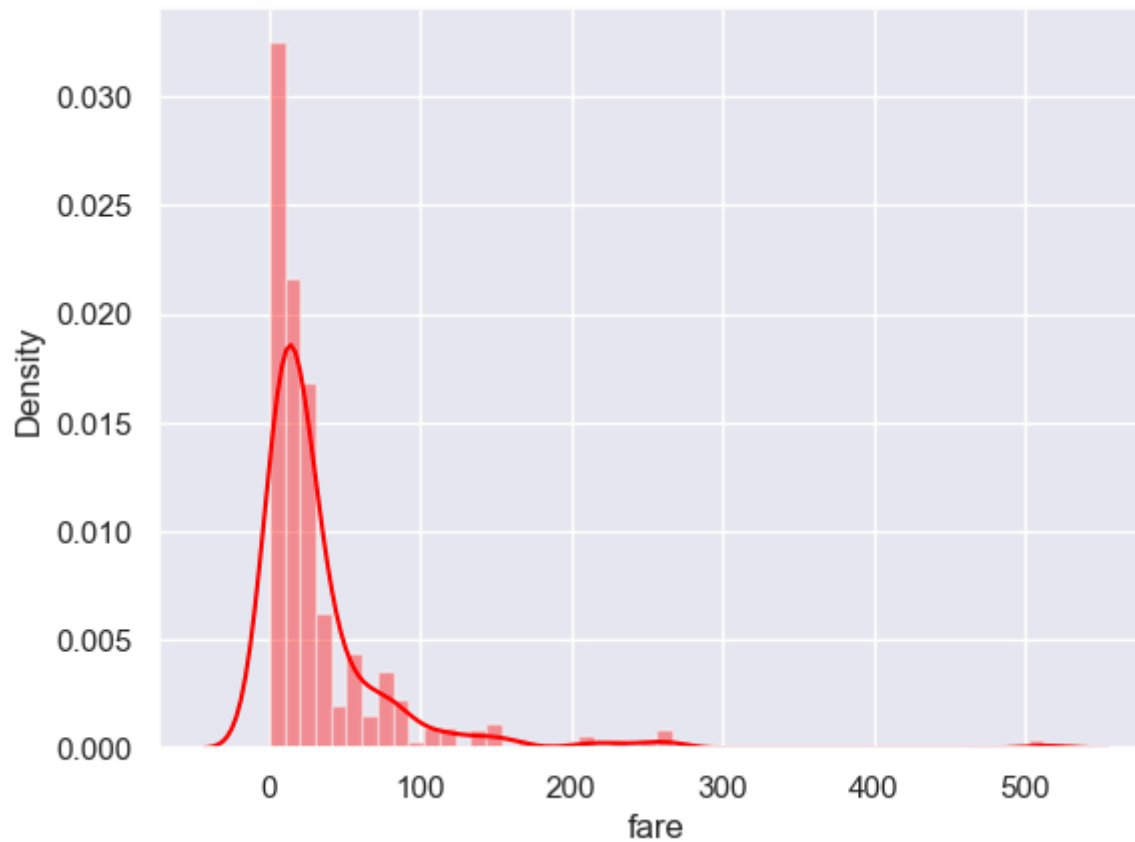
In [67]:
```python
titanic.tail(3)
```

	survived	pclass	sex	age	sibsp	parch	fare	embarked	class	who	adult_male	E Town	alive	alone
887	1	1	female	19	0	0	30.00	S	First	woman	False	Southampton	yes	True
889	1	1	male	26	0	0	30.00	C	First	man	True	Cherbourg	yes	True
890	0	3	male	32	0	0	7.75	Q	Third	man	True	Queenstown	no	True

In [68]:
```python
alone=titanic.groupby(['alone'])['alone'].count().reset_index(name='count')
```

In [69]:
```python
color=('white','violet')
```

In [70]:
```python
plt.pie(x='count',labels='alone',autopct='%1.1f%%',data=alone,colors=color);
```

False

43.5%

56.5%
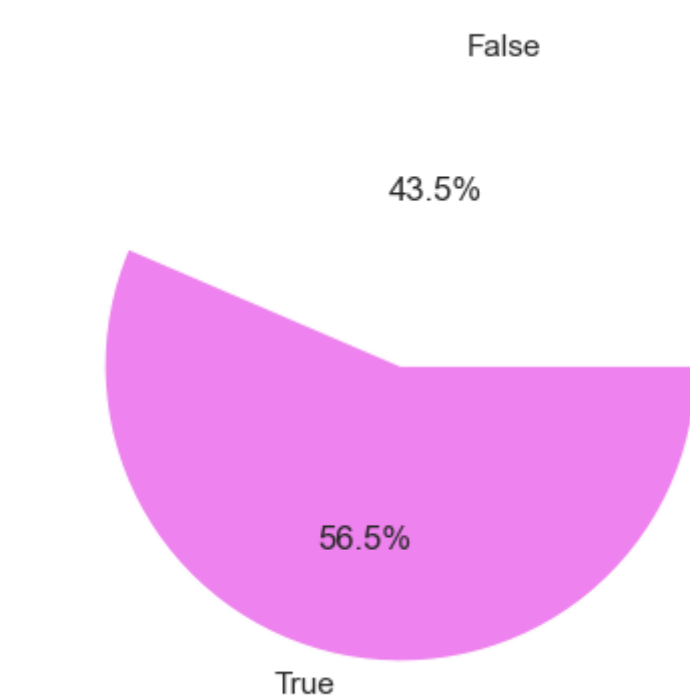
True

Most of the Peoples were Alone(Single)

```
In [71]: G=titanic.groupby(['alone']).get_group(True).groupby(['sex','survived'])['survived'].count().reset_index(name='count').sort_value
```

```
In [72]: G
```

Out[72]:

	sex	survived	count
2	male	0	253
1	female	1	77
3	male	1	51
0	female	0	21

```
In [73]: ax=sns.barplot(x='sex',hue='survived',y='count',data=G,palette='dark')
         for p in ax.patches:
             ax.annotate(format(p.get_height(), '.0f'),
                         (p.get_x() + p.get_width() / 2., p.get_height()),
                         ha = 'center', va = 'center',
                         xytext = (0, 8),
                         textcoords = 'offset points')
         plt.xticks(rotation=45)
         plt.title('Count of alone peoples ( Sex ) based on Survived')
         plt.show()
```

Count of alone peoples (Sex) based on Survived
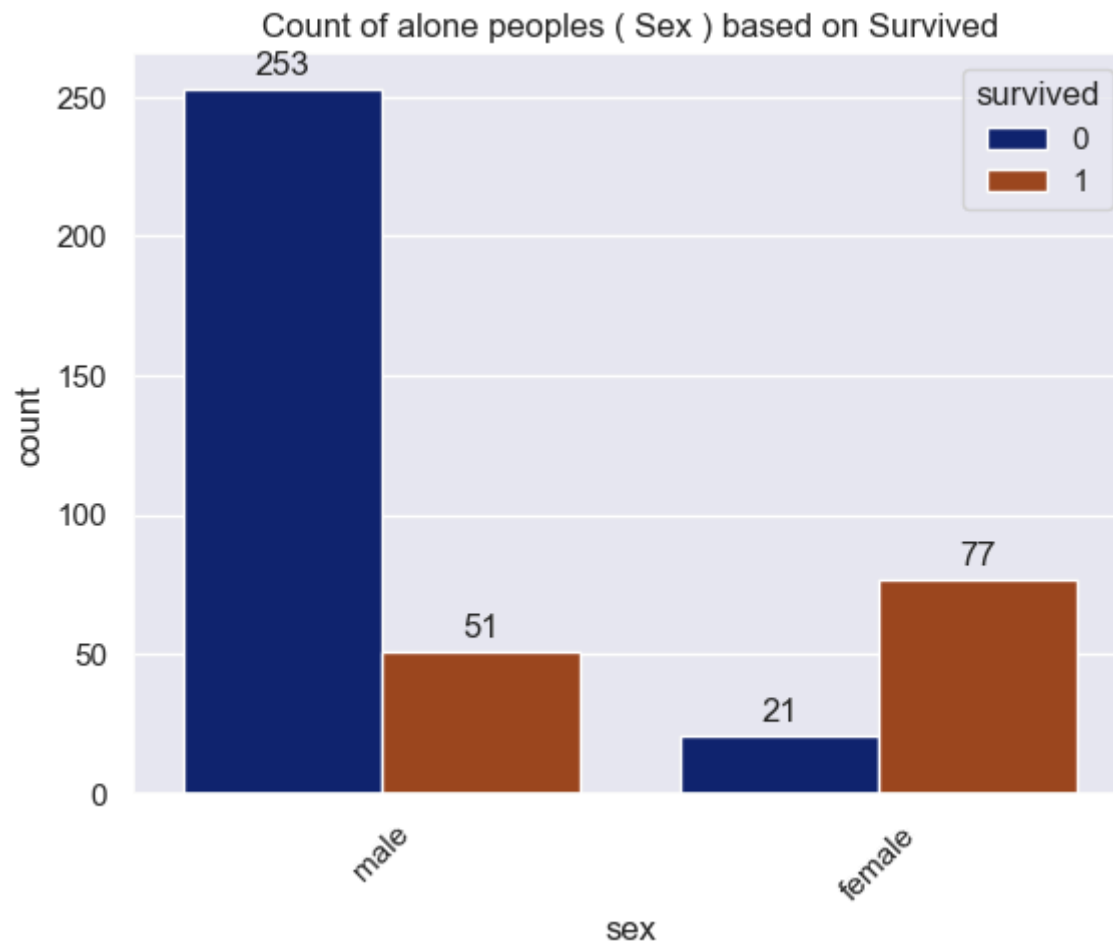
253 alone males & 21 alone females lost their lives ...

In [74]: `titanic.columns`

Out[74]:
```
Index(['survived', 'pclass', 'sex', 'age', 'sibsp', 'parch', 'fare',
       'embarked', 'class', 'who', 'adult_male', 'E Town', 'alive', 'alone'],
      dtype='object')
```

In [75]: `titanic.corr(numeric_only=True)`

|	survived	pclass	age	sibsp	parch	fare	adult_male	alone
survived	1.000000	-0.356462	-0.082377	-0.015523	0.095265	0.266100	-0.551151	-0.199741
pclass	-0.356462	1.000000	-0.366125	0.065187	0.023666	-0.552893	0.094635	0.150576
age	-0.082377	-0.366125	1.000000	-0.307050	-0.188084	0.093364	0.286488	0.195827
sibsp	-0.015523	0.065187	-0.307050	1.000000	0.383338	0.139860	-0.313016	-0.629408
parch	0.095265	0.023666	-0.188084	0.383338	1.000000	0.206624	-0.365580	-0.577109
fare	0.266100	-0.552893	0.093364	0.139860	0.206624	1.000000	-0.177446	-0.262799
adult_male	-0.551151	0.094635	0.286488	-0.313016	-0.365580	-0.177446	1.000000	0.400718
alone	-0.199741	0.150576	0.195827	-0.629408	-0.577109	-0.262799	0.400718	1.000000

```python
plt.figure(figsize=(8,6))
sns.heatmap(titanic.corr(numeric_only=True),annot=True,cmap='coolwarm')
```

```
<Axes: >
```
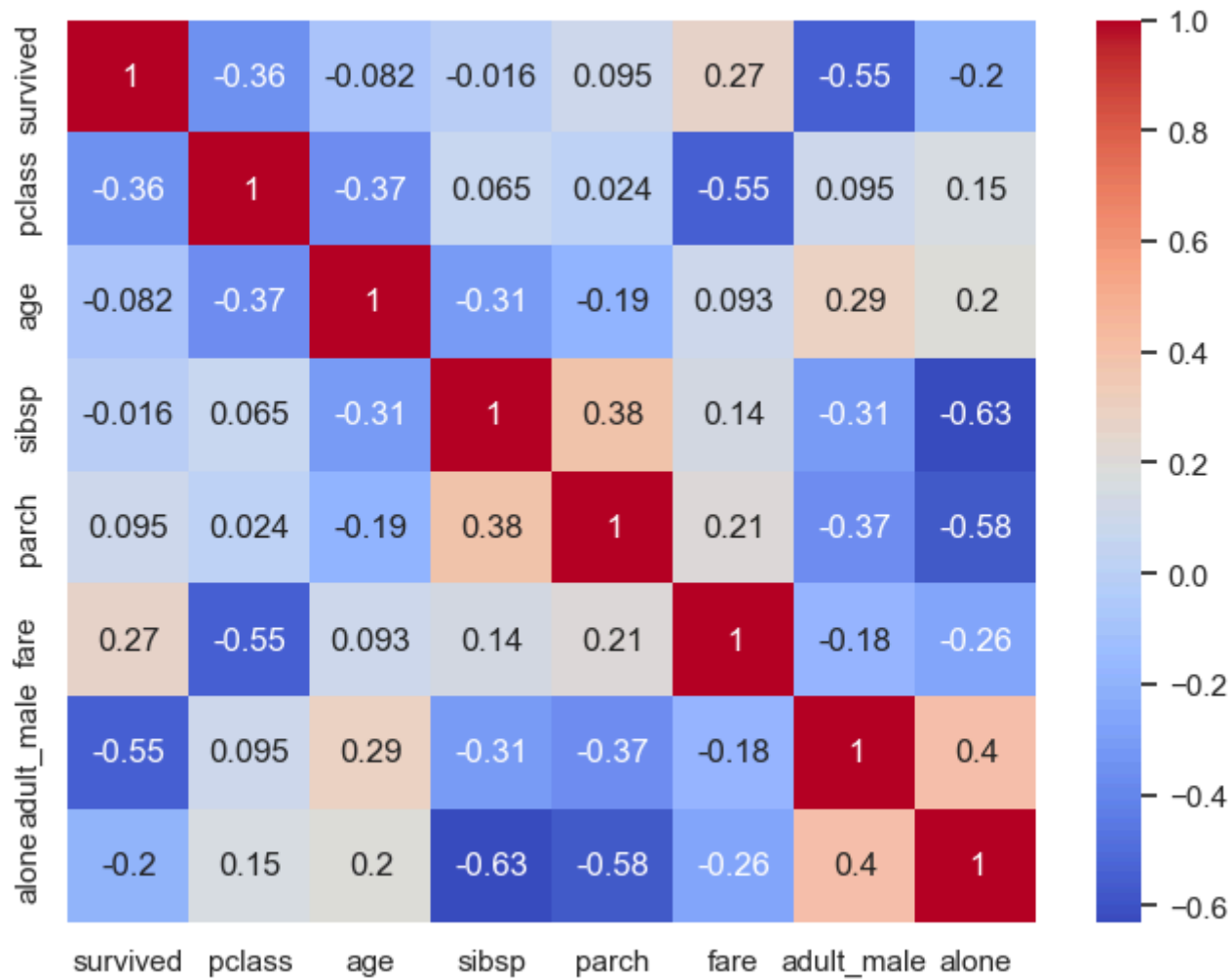
No Good Relation Found Between Columns__

FINAL CONCLUSION: 🔍

➡️ Most of the people on Titanic were between the ages of 18 to 37.

➡️ Titanic was carrying 453 males and 259 females, out of which 360 males and 64 females lost their lives.

➡️ Out of 360 death males, 341 were young.

➡️ In the Titanic ship, 253 males and 21 female who were alone (single), lost their lives.

➡️ Mostly people died (Total 270 peoples) in P-Class 3, in which 55 female and 215 male.

➡️ In this incident, 34 children also lost their lives in which 19 were boys and 15 were girls. .

⬇️