In [1]:
```python
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
```

In [2]:
```python
#importing our dataset
df = pd.read_csv('C:\\Users\\pujit\\Downloads\\heart.csv')
```

In [3]:
```python
#import first 5 rows by calling df.head()
df.head()
```

Out[3]:

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
0	52	1	0	125	212	0	1	168	0	1.0	2	2	3	0
1	53	1	0	140	203	1	0	155	1	3.1	0	0	3	0
2	70	1	0	145	174	0	1	125	1	2.6	0	0	3	0
3	61	1	0	148	203	0	1	161	0	0.0	2	1	3	0
4	62	0	0	138	294	1	1	106	0	1.9	1	3	2	0

In [4]:
```python
#it gives bottom 5 rows of an dataset
df.tail()
```

Out[4]:

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
1020	59	1	1	140	221	0	1	164	1	0.0	2	0	2	1
1021	60	1	0	125	258	0	0	141	1	2.8	1	1	3	0
1022	47	1	0	110	275	0	0	118	1	1.0	1	1	2	0
1023	50	0	0	110	254	0	0	159	0	0.0	2	0	2	1
1024	54	1	0	120	188	0	1	113	0	1.4	1	1	3	0

In [5]:
```python
#take a look on colun names
df.columns.values
```

Out[5]:
```
array(['age', 'sex', 'cp', 'trestbps', 'chol', 'fbs', 'restecg',
       'thalach', 'exang', 'oldpeak', 'slope', 'ca', 'thal', 'target'],
      dtype=object)
```

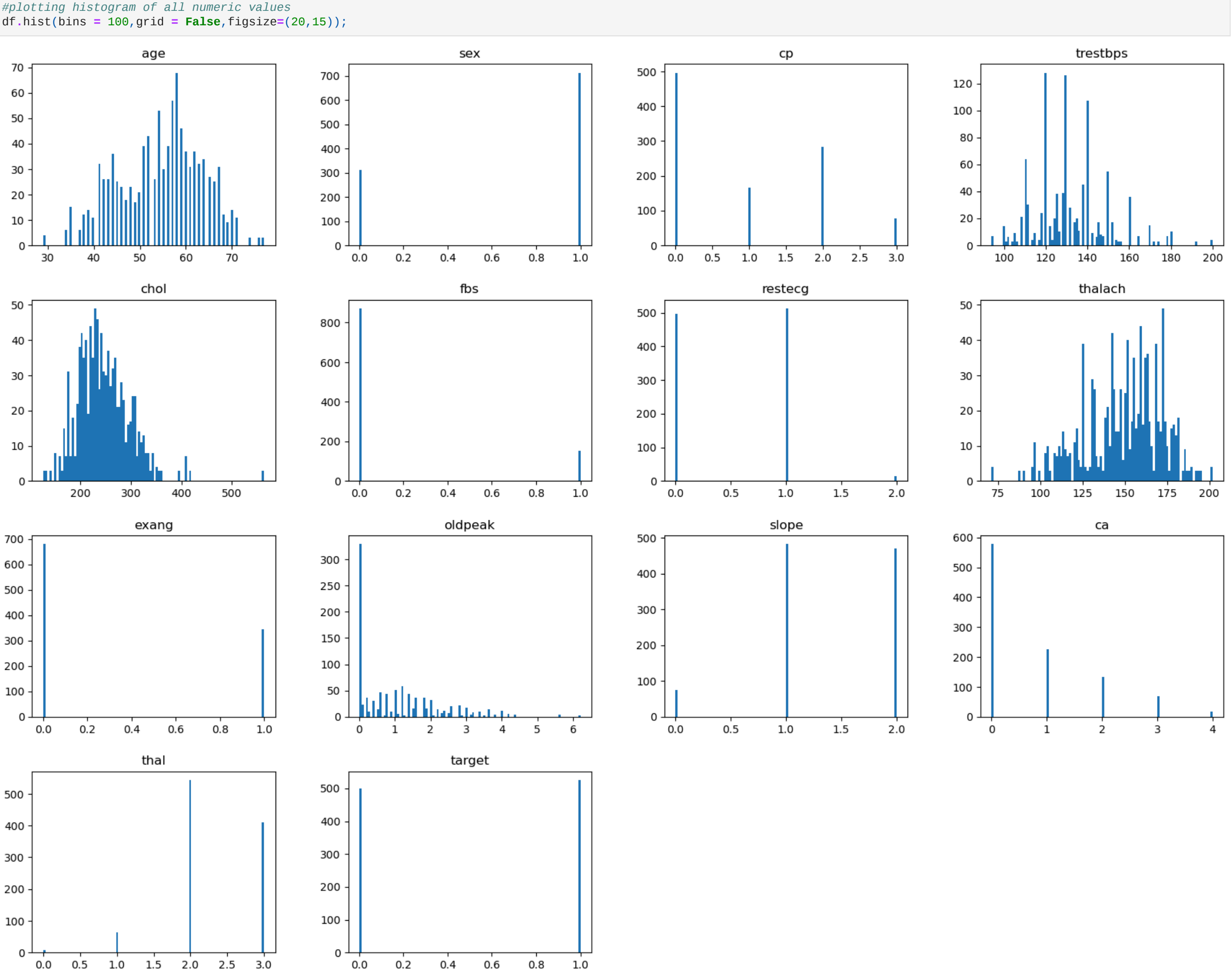
In [6]:
```python
#checking for the null values(how many null values or present in the particylar column)
df.isna().sum()
```

Out[6]:
```
age         0
sex         0
cp          0
trestbps    0
chol        0
fbs         0
restecg     0
thalach     0
exang       0
oldpeak     0
slope       0
ca          0
thal        0
target      0
dtype: int64
```

In [7]:
```python
#tells the basic information of the column
#concise summary of our dataset
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1025 entries, 0 to 1024
Data columns (total 14 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   age       1025 non-null   int64
 1   sex       1025 non-null   int64
 2   cp        1025 non-null   int64
 3   trestbps  1025 non-null   int64
 4   chol      1025 non-null   int64
 5   fbs       1025 non-null   int64
 6   restecg   1025 non-null   int64
 7   thalach   1025 non-null   int64
 8   exang     1025 non-null   int64
 9   oldpeak   1025 non-null   float64
 10  slope     1025 non-null   int64
 11  ca        1025 non-null   int64
 12  thal      1025 non-null   int64
 13  target    1025 non-null   int64
dtypes: float64(1), int64(13)
memory usage: 112.2 KB
```

In [8]:
```python
#plotting histogram of all numeric values
df.hist(bins = 100,grid = False,figsize=(20,15));
```



In [9]:
```python
#generating descriptive statistics
df.describe()
```

Out[9]:

	age	sex	cp	trestbps	chol	fbs	restecg
count	1025.000000	1025.000000	1025.000000	1025.000000	1025.00000	1025.000000	1025.000000
mean	54.434146	0.695610	0.942439	131.611707	246.00000	0.149268	0.529756
std	9.072290	0.460373	1.029641	17.516718	51.59251	0.356527	0.527878
min	29.000000	0.000000	0.000000	94.000000	126.00000	0.000000	0.000000
25%	48.000000	0.000000	0.000000	120.000000	211.00000	0.000000	0.000000
50%	56.000000	1.000000	1.000000	130.000000	240.00000	0.000000	1.000000
75%	61.000000	1.000000	2.000000	140.000000	275.00000	0.000000	1.000000
max	77.000000	1.000000	3.000000	200.000000	564.00000	1.000000	2.000000

	thalach	exang	oldpeak	slope	ca	thal	target
count	1025.000000	1025.000000	1025.000000	1025.000000	1025.000000	1025.000000	1025.000000
mean	149.114146	0.336585	1.071512	1.385366	0.754146	2.323902	0.513171
std	23.005724	0.472772	1.175053	0.617755	1.030798	0.620660	0.500070
min	71.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
25%	132.000000	0.000000	0.000000	1.000000	0.000000	2.000000	0.000000
50%	152.000000	0.000000	0.800000	1.000000	0.000000	2.000000	1.000000
75%	166.000000	1.000000	1.800000	2.000000	1.000000	3.000000	1.000000
max	202.000000	1.000000	6.200000	2.000000	4.000000	3.000000	1.000000

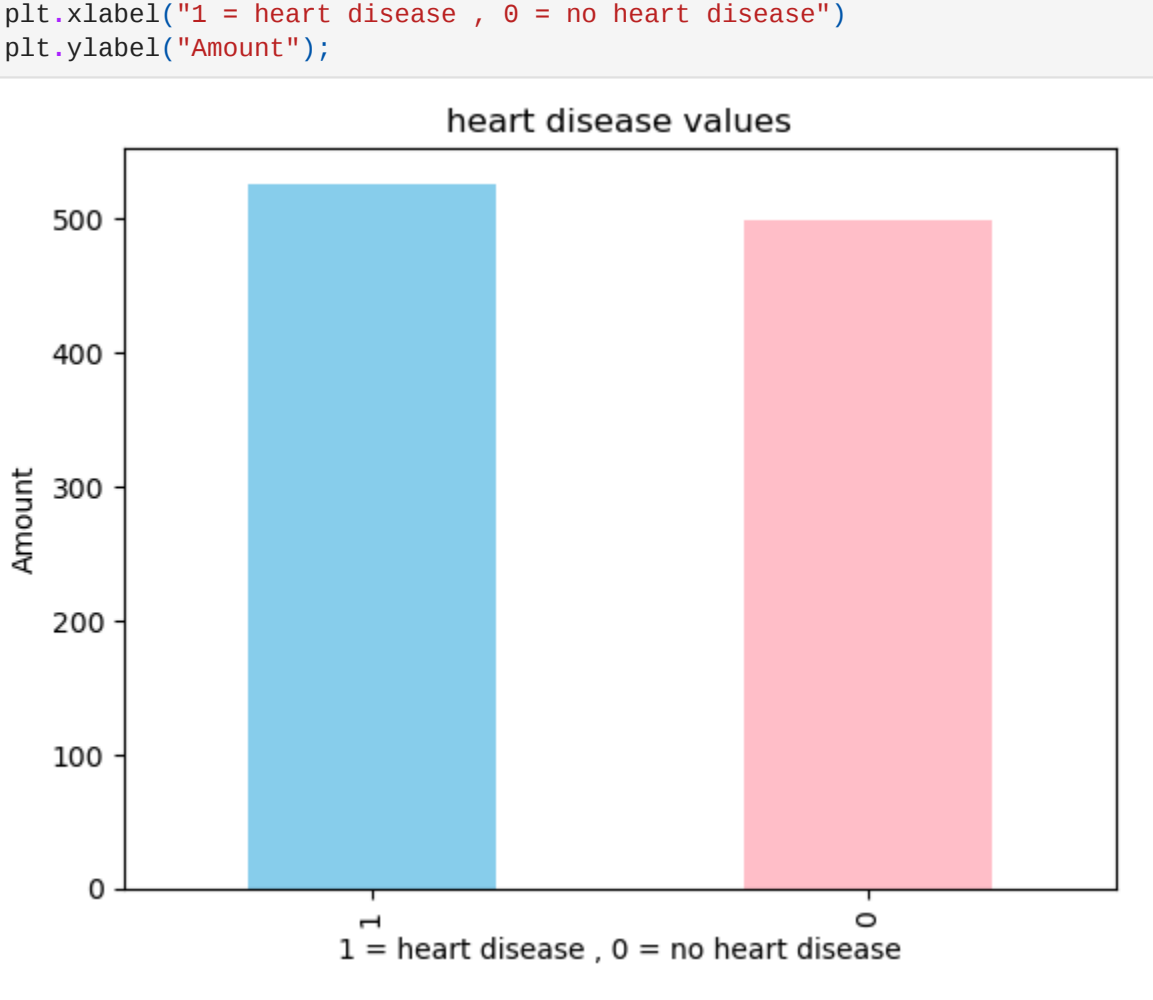
In [10]:
```python
questions = ["1,how many people have heart disease and how many people doesnt have heart disease?",
             "2.people of which sex has most heart disease",
             "3.people of which sex has which type of chest pain most",
             "4.people with which chest pain are most pron to have heart disease"]
questions
```

Out[10]:
```
['1,how many people have heart disease and how many people doesnt have heart disease?',
 '2.people of which sex has most heart disease',
 '3.people of which sex has which type of chest pain most',
 '4.people with which chest pain are most pron to have heart disease']
```

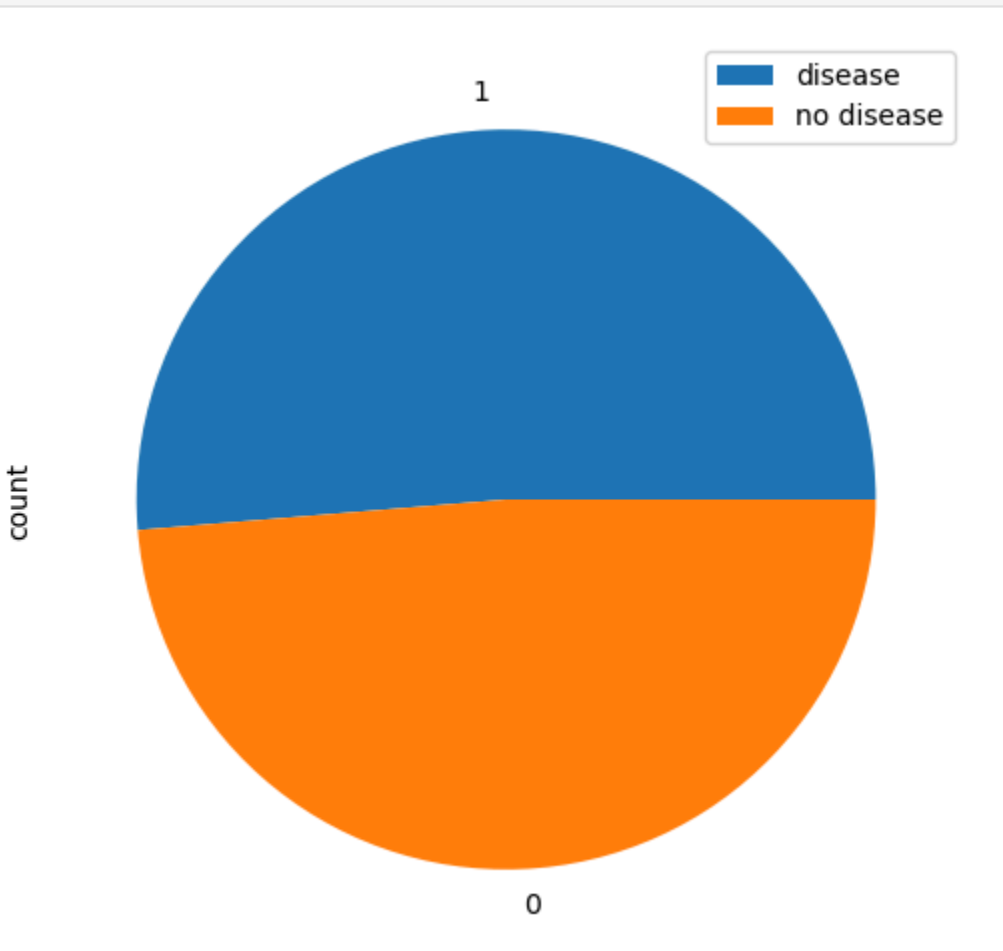
In [11]:
```python
#first ques

#getting values
df.target.value_counts()
```

Out[11]:
```
target
1    526
0    499
Name: count, dtype: int64
```

In [12]:
```python
#plotting bar chart
df.target.value_counts().plot(kind = 'bar',color=["skyblue" , "pink"])
plt.title("heart disease values")
plt.xlabel("1 = heart disease , 0 = no heart disease")
plt.ylabel("Amount");
```



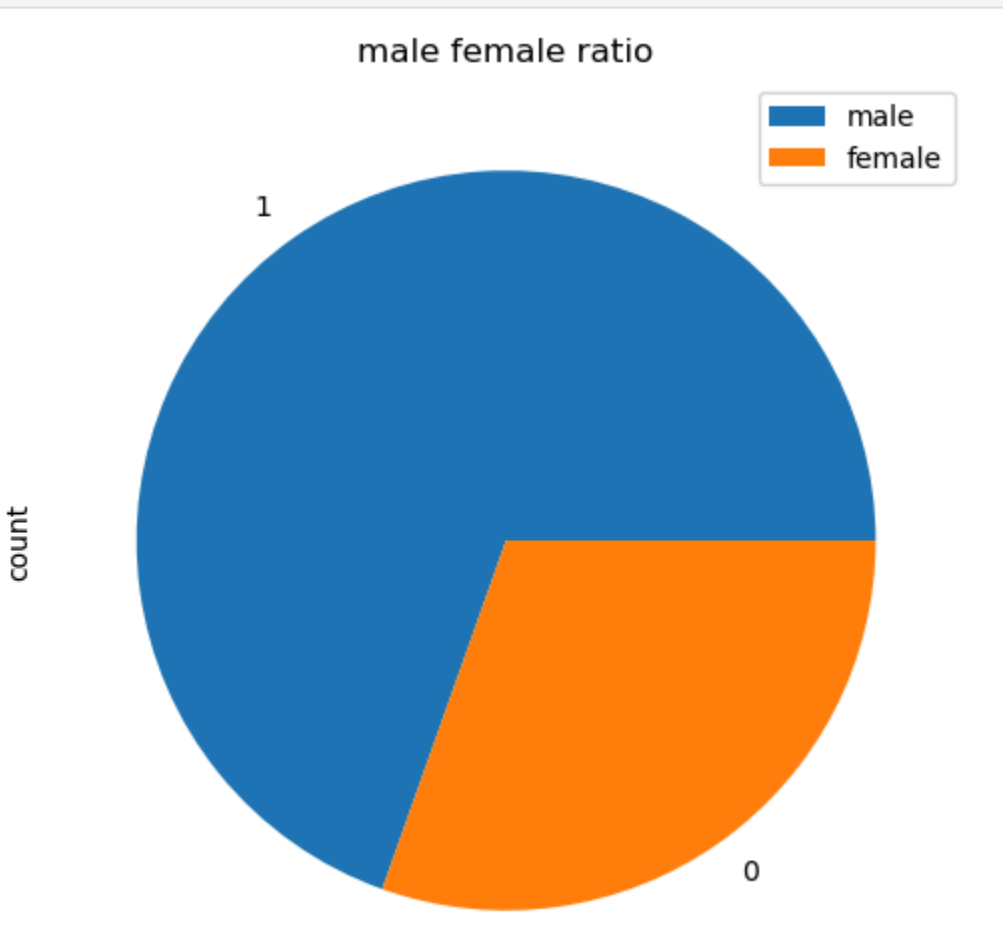
In [13]:
```python
#plotting a pie chart
df.target.value_counts().plot(kind = 'pie',figsize = (7,6))
plt.legend(["disease","no disease"]);
```



In [14]:
```python
#'0' = female, '1'= male
#'0' = no heart disease, '1'=heart disease


#checking how many male and females are int he dataset
df.sex.value_counts()
```

Out[14]:
```
sex
1    713
0    312
Name: count, dtype: int64
```

In [15]:
```python
#plotting a pie chart
df.sex.value_counts().plot(kind = 'pie',figsize = (8,6))
plt.title('male female ratio')
plt.legend(['male','female']);
```

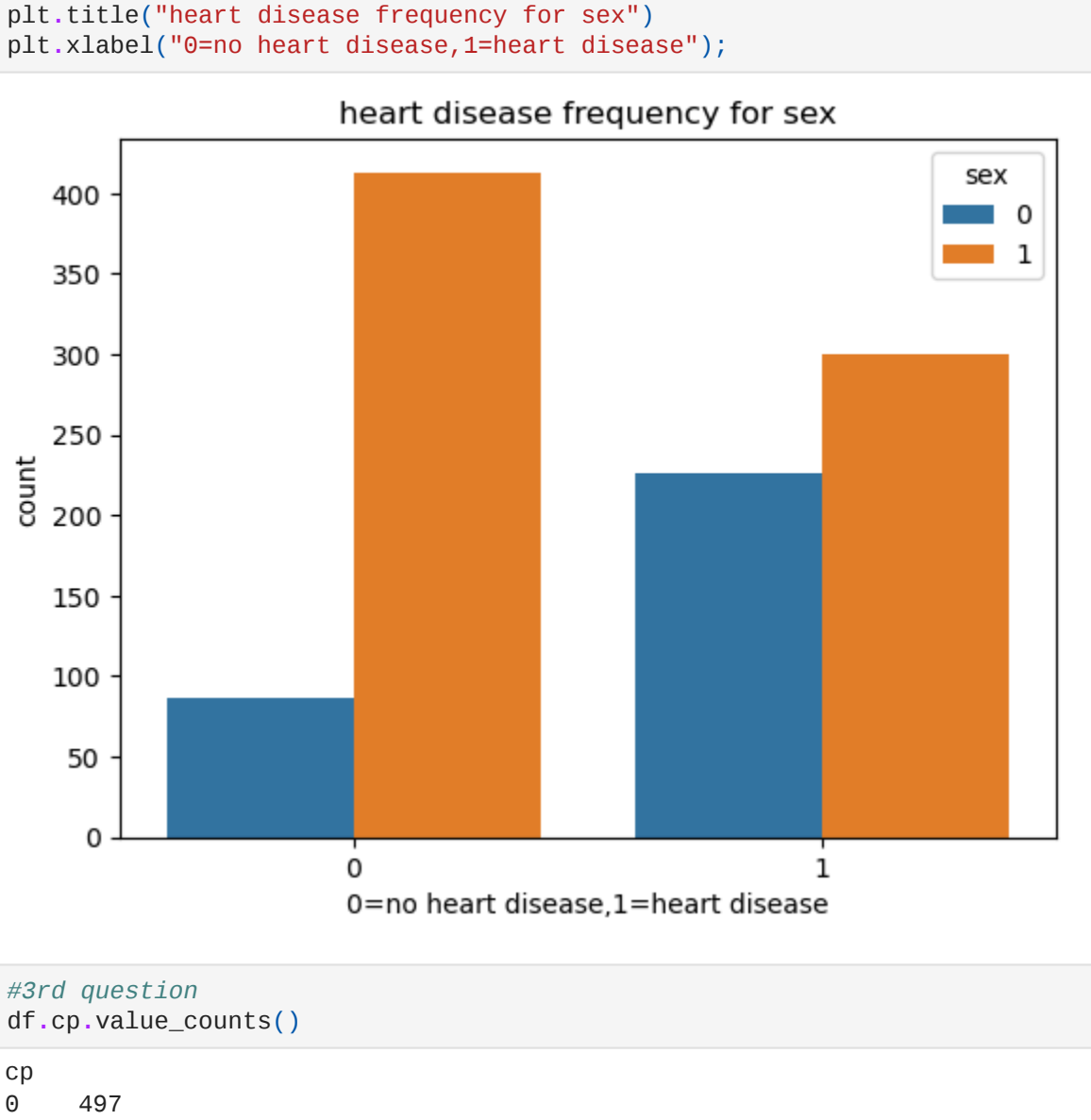


In [16]:
```python
#2nd question
pd.crosstab(df.target,df.sex)
```

Out[16]:

target \ sex	0	1
0	86	413
1	226	300

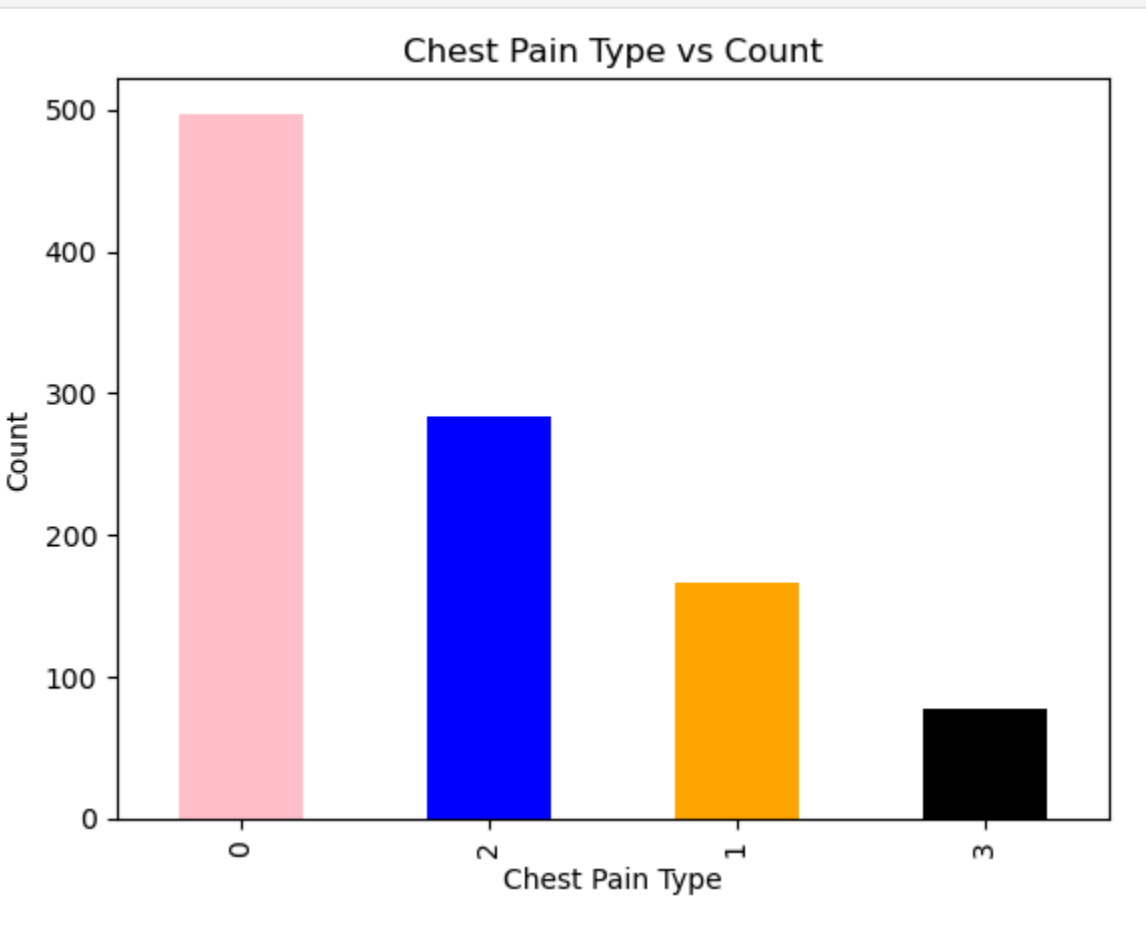
In [17]:
```python
sns.countplot(x='target',data = df,hue = 'sex')
plt.title("heart disease frequency for sex")
plt.xlabel("0=no heart disease,1=heart disease");
```



In [18]:
```python
#3rd question
df.cp.value_counts()
```

Out[18]:
```
cp
0    497
2    284
1    167
3     77
Name: count, dtype: int64
```

In [23]:
```python
#plotting a bar chart
df.cp.value_counts().plot(kind='bar', color=['pink', 'blue', 'orange', 'black'])
plt.title('Chest Pain Type vs Count')
plt.xlabel('Chest Pain Type')
plt.ylabel('Count')
plt.show()
```

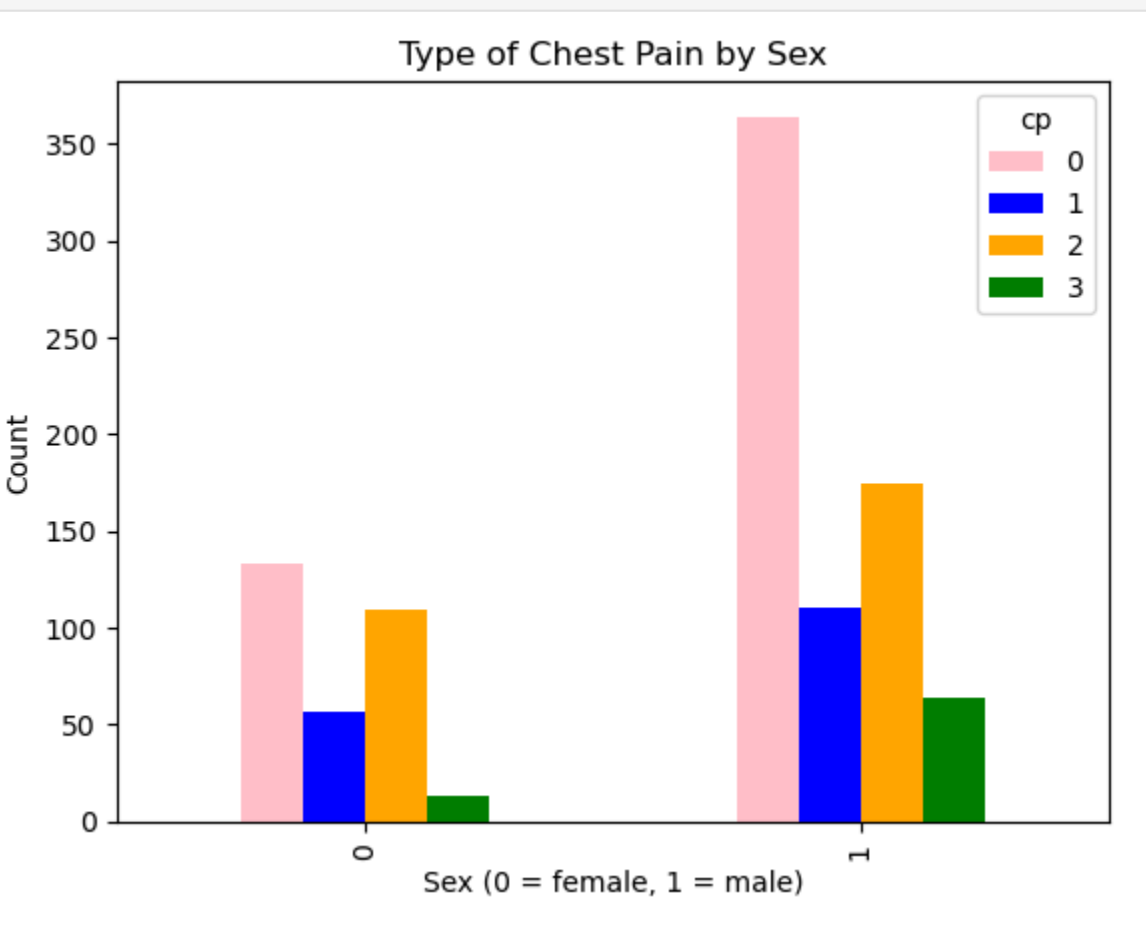


In [24]:
```python
pd.crosstab(df.sex,df.cp)
```

Out[24]:

sex \ cp	0	1	2	3
0	133	57	109	13
1	364	110	175	64

In [27]:
```python
pd.crosstab(df.sex, df.cp).plot(kind='bar', color=['pink', 'blue', 'orange', 'green'])
plt.title('Type of Chest Pain for Sex')
plt.xlabel('Sex (0 = female, 1 = male)')
plt.ylabel('Count')
plt.show()
```



In [28]:
```python
#question 4
pd.crosstab(df.cp, df.target)
```

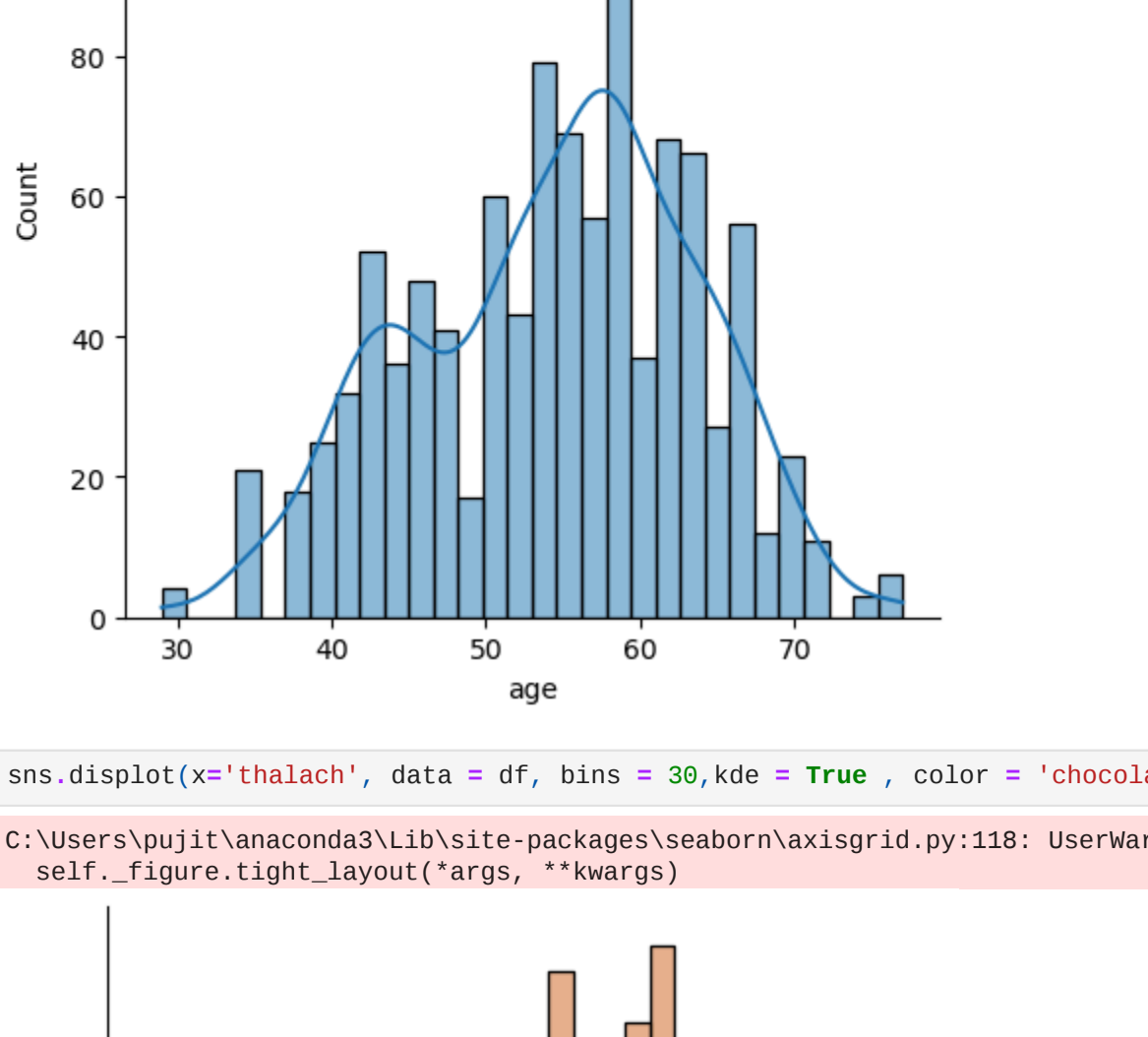
Out[28]:

cp \ target	0	1
0	375	122
1	33	134
2	65	219
3	26	51

In [29]:
```python
sns.countplot(x='cp',data = df,hue = 'target');
```

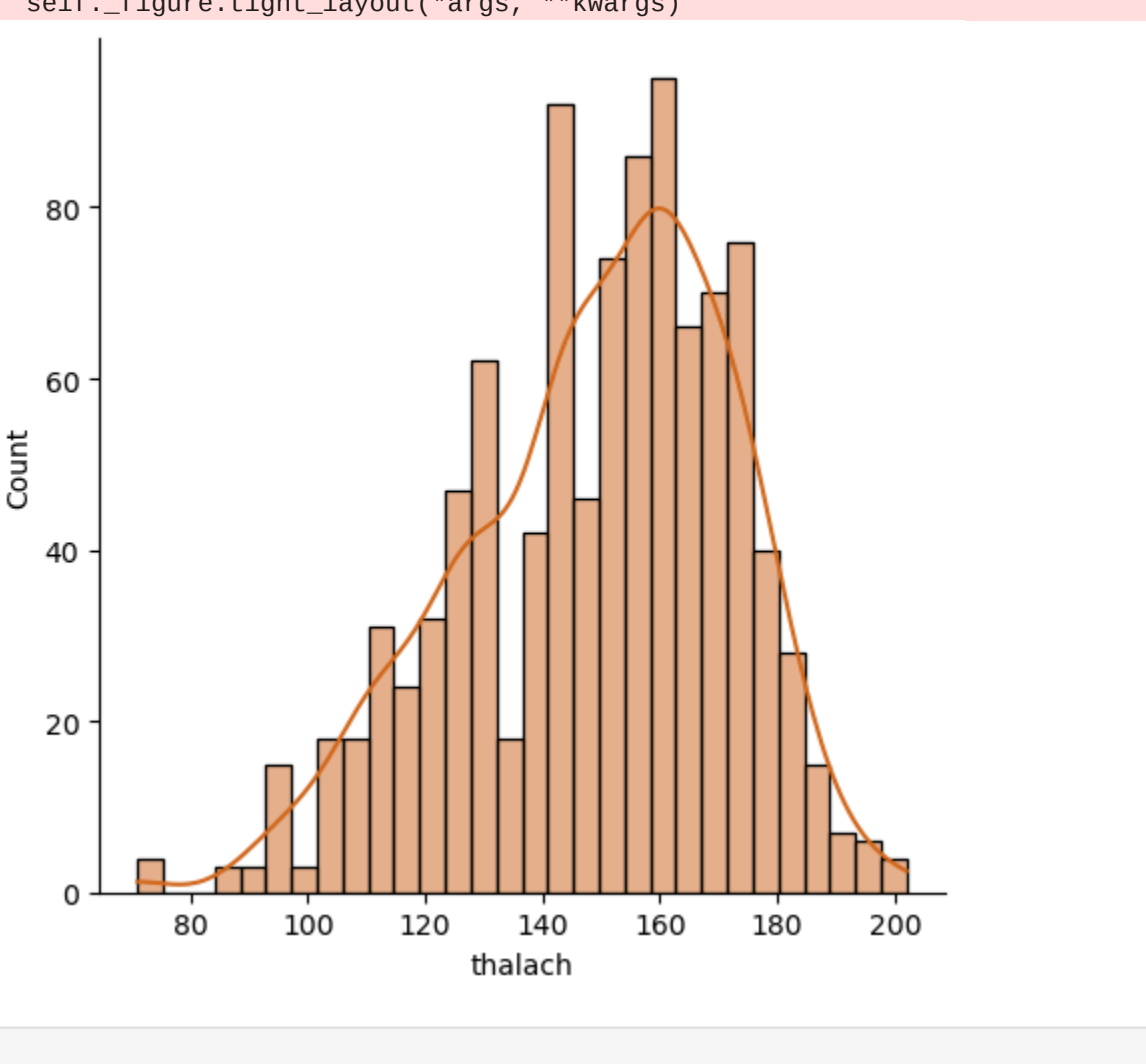
In [32]:
```python
sns.displot(x='age',data = df,bins = 30, kde = True);
```

```
C:\Users\pujit\anaconda3\Lib\site-packages\seaborn\axisgrid.py:118: UserWarning: The figure layout has changed to tight
  self._figure.tight_layout(*args, **kwargs)
```



In [34]:
```python
sns.displot(x='thalach', data = df, bins = 30,kde = True , color = 'chocolate');
```

```
C:\Users\pujit\anaconda3\Lib\site-packages\seaborn\axisgrid.py:118: UserWarning: The figure layout has changed to tight
  self._figure.tight_layout(*args, **kwargs)
```



In [ ]: