```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline

# Importing our dataset

data = pd.read_csv(r"C:\Users\Raman\Downloads\heart.csv")
data
```

```
      age  sex  cp  trestbps  chol  fbs  restecg  thalach  exang
oldpeak  \
0      52    1   0       125   212    0        1      168      0
1.0
1      53    1   0       140   203    1        0      155      1
3.1
2      70    1   0       145   174    0        1      125      1
2.6
3      61    1   0       148   203    0        1      161      0
0.0
4      62    0   0       138   294    1        1      106      0
1.9
...   ...  ...  ..       ...   ...  ...      ...      ...    ...
...
1020   59    1   1       140   221    0        1      164      1
0.0
1021   60    1   0       125   258    0        0      141      1
2.8
1022   47    1   0       110   275    0        0      118      1
1.0
1023   50    0   0       110   254    0        0      159      0
0.0
1024   54    1   0       120   188    0        1      113      0
1.4

      slope  ca  thal  target
0         2   2     3       0
1         0   0     3       0
2         0   0     3       0
3         2   1     3       0
4         1   3     2       0
...     ...  ..   ...     ...
1020      2   0     2       1
1021      1   1     3       0
1022      1   1     2       0
1023      2   0     2       1
1024      1   1     3       0

[1025 rows x 14 columns]
```

```
# Accessing Only the column names
data.columns.values
```

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

```
# Checking for the Null values

data.isna().sum()
```

```
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
```

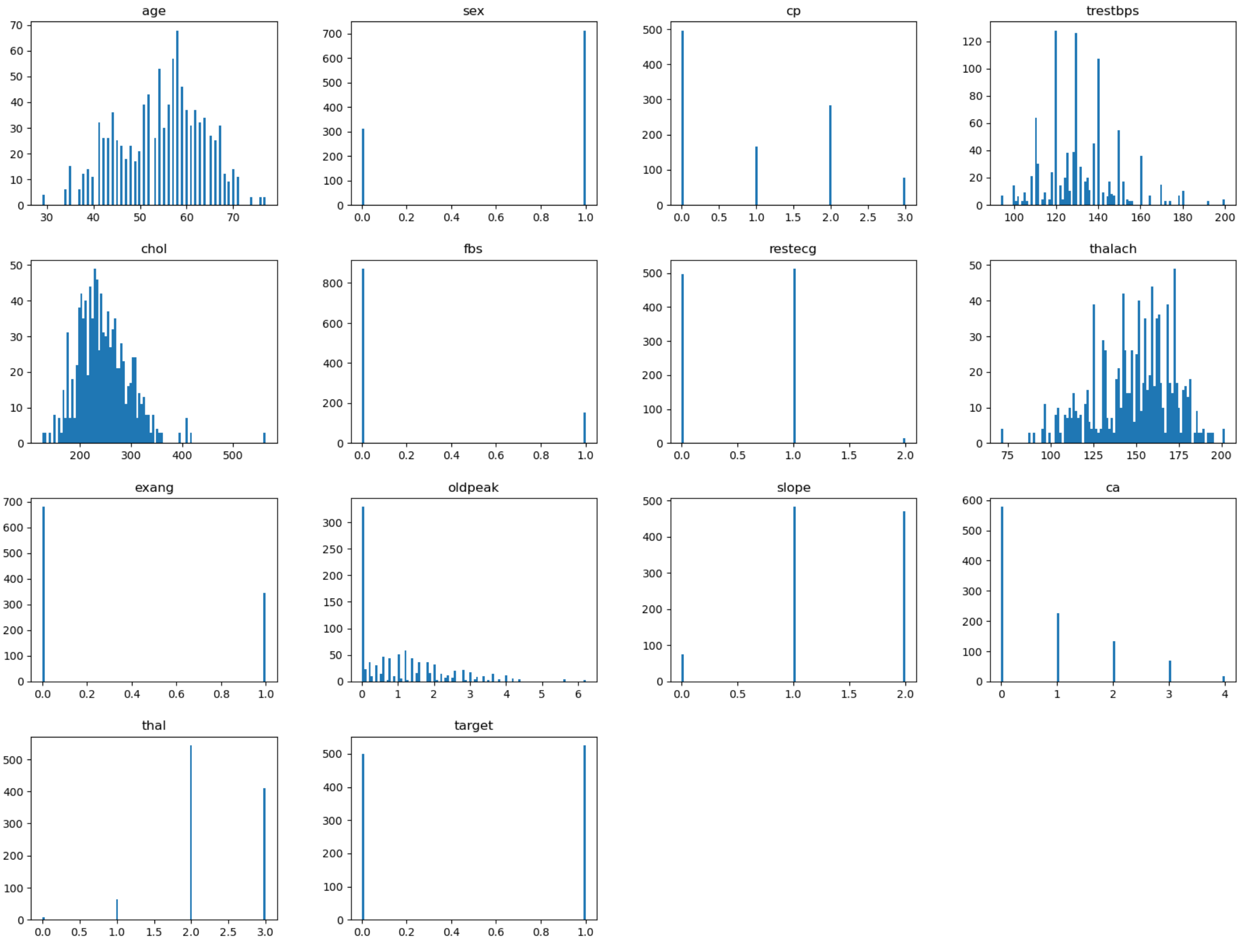
```
# Summary of our Dataset

data.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
```

```
# Plotting Histograms of All Numeric Values:

data.hist(bins = 100, grid = False, figsize = (20,15))
```

```
array([[<Axes: title={'center': 'age'}>, <Axes: title={'center':
'sex'}>,
        <Axes: title={'center': 'cp'}>,
        <Axes: title={'center': 'trestbps'}>],
       [<Axes: title={'center': 'chol'}>,
        <Axes: title={'center': 'fbs'}>,
        <Axes: title={'center': 'restecg'}>,
        <Axes: title={'center': 'thalach'}>],
       [<Axes: title={'center': 'exang'}>,
        <Axes: title={'center': 'oldpeak'}>,
        <Axes: title={'center': 'slope'}>,
        <Axes: title={'center': 'ca'}>],
       [<Axes: title={'center': 'thal'}>,
        <Axes: title={'center': 'target'}>, <Axes: >, <Axes: >]],
      dtype=object)
```



```
#Generating the Descriptive Statistics

data.describe()

               age          sex           cp     trestbps        chol
\
count  1025.000000  1025.000000  1025.000000  1025.000000  1025.00000

mean     54.434146     0.695610     0.942439   131.611707   246.00000

std       9.072290     0.460373     1.029641    17.516718    51.59251

min      29.000000     0.000000     0.000000    94.000000   126.00000

25%      48.000000     0.000000     0.000000   120.000000   211.00000

50%      56.000000     1.000000     1.000000   130.000000   240.00000

75%      61.000000     1.000000     2.000000   140.000000   275.00000

max      77.000000     1.000000     3.000000   200.000000   564.00000
```

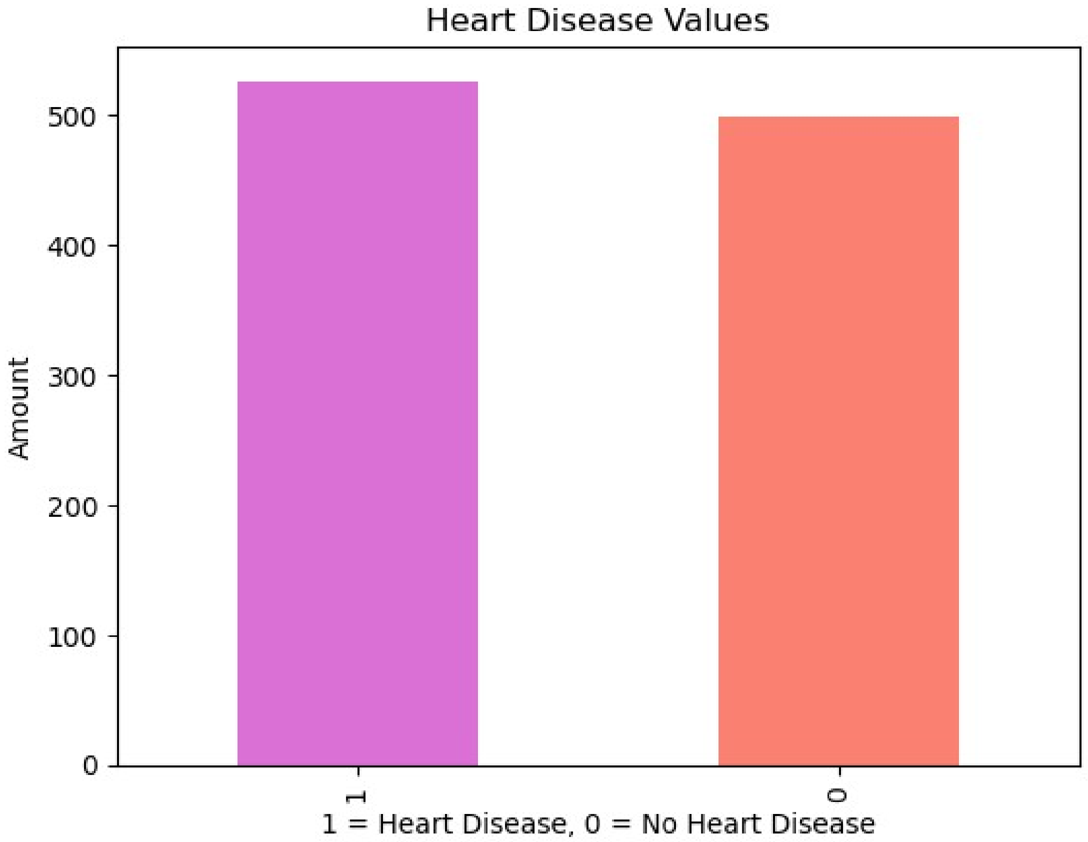
```
               fbs      restecg      thalach        exang      oldpeak  
\  
count  1025.000000  1025.000000  1025.000000  1025.000000  1025.000000   

mean      0.149268     0.529756   149.114146     0.336585     1.071512   

std       0.356527     0.527878    23.005724     0.472772     1.175053   

min       0.000000     0.000000    71.000000     0.000000     0.000000   

25%       0.000000     0.000000   132.000000     0.000000     0.000000   

50%       0.000000     1.000000   152.000000     0.000000     0.800000   

75%       0.000000     1.000000   166.000000     1.000000     1.800000   

max       1.000000     2.000000   202.000000     1.000000     6.200000   


             slope           ca         thal       target  
count  1025.000000  1025.000000  1025.000000  1025.000000  
mean      1.385366     0.754146     2.323902     0.513171  
std       0.617755     1.030798     0.620660     0.500070  
min       0.000000     0.000000     0.000000     0.000000  
25%       1.000000     0.000000     2.000000     0.000000  
50%       1.000000     0.000000     2.000000     1.000000  
75%       2.000000     1.000000     3.000000     1.000000  
max       2.000000     4.000000     3.000000     1.000000  
```

```
questions = ["1. How Many people have heart disease and how many 
people doesn't 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
```

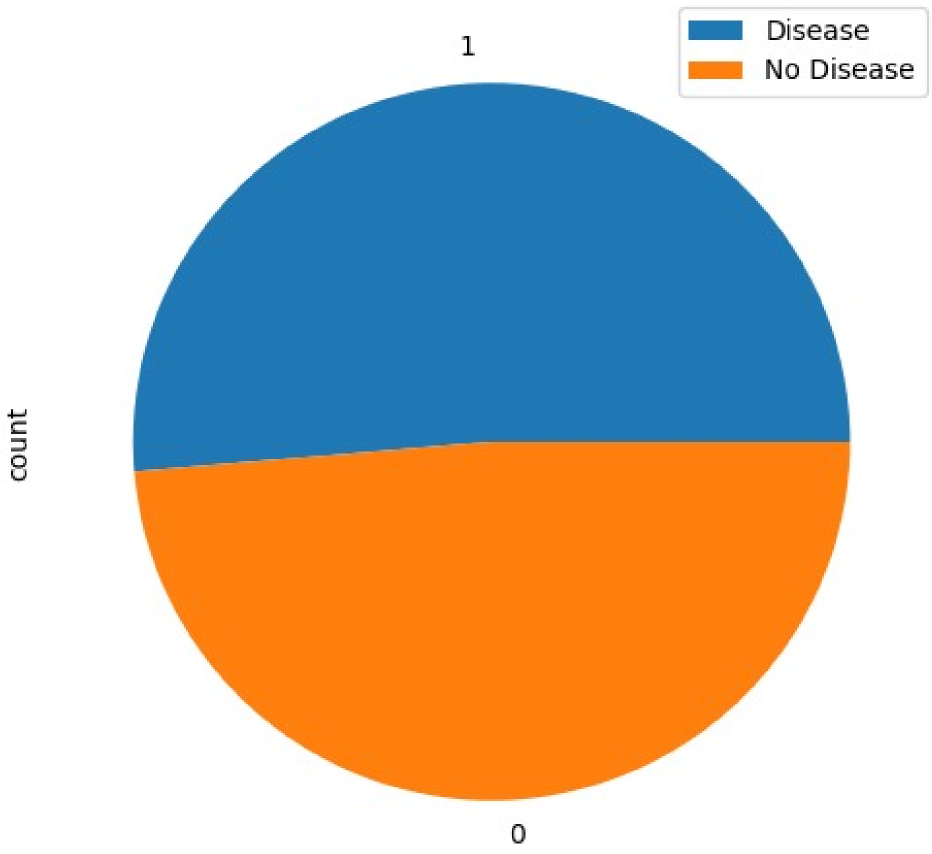
```
["1. How Many people have heart disease and how many people doesn't 
have heart disease ?",
 '2. People of which sex has most heart disease ?',
 '3. People of which sex has which type of chest pain most?',
 '4. People with which chest pain are most pron to have heart 
disease?']
```

1. How Many people have heart disease and how many people doesn't have heart disease ?

```
data.head()

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

   ca  thal  target
0   2     3       0
1   0     3       0
2   0     3       0
3   1     3       0
4   3     2       0

data['target'].value_counts()

target
1    526
0    499
Name: count, dtype: int64

data.target.value_counts().plot(kind = 'bar', color =
["orchid","salmon"])
plt.title("Heart Disease Values")
plt.xlabel("1 = Heart Disease, 0 = No Heart Disease")
plt.ylabel("Amount")

Text(0, 0.5, 'Amount')
```



```
# Plotting a Pie Chart

data.target.value_counts().plot(kind = 'pie', figsize = (8, 6))
plt.legend(["Disease", "No Disease"])

<matplotlib.legend.Legend at 0x1b8e9a34250>
```

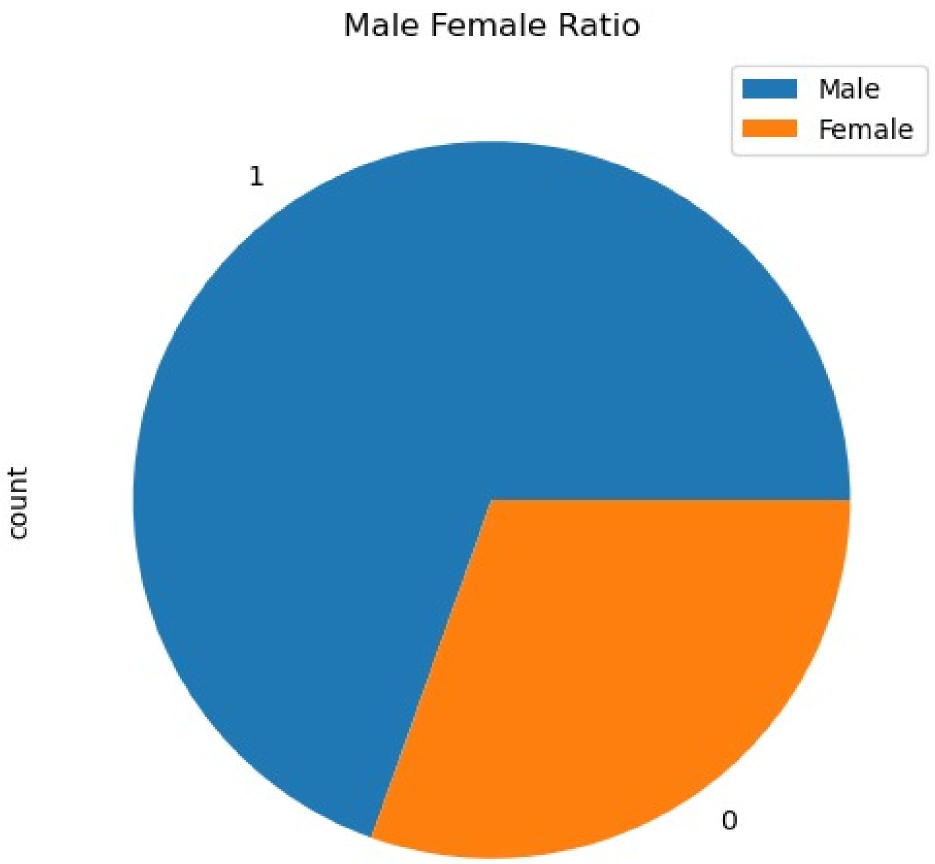


```python
# Checking the Number of Males and Females in the Dataset
# 1 - represents the MALE ; 0 - represents the FEMALE
# 0 - represents No Disease ; 1 - represents the Disease

data.sex.value_counts()
```

```
sex
1    713
0    312
Name: count, dtype: int64
```

```python
# Plotting a Pie Chart

data.sex.value_counts().plot(kind = 'pie', figsize = (8, 6))
plt.title("Male Female Ratio")
plt.legend(["Male","Female"])
```

```
<matplotlib.legend.Legend at 0x1b8ea4f40d0>
```



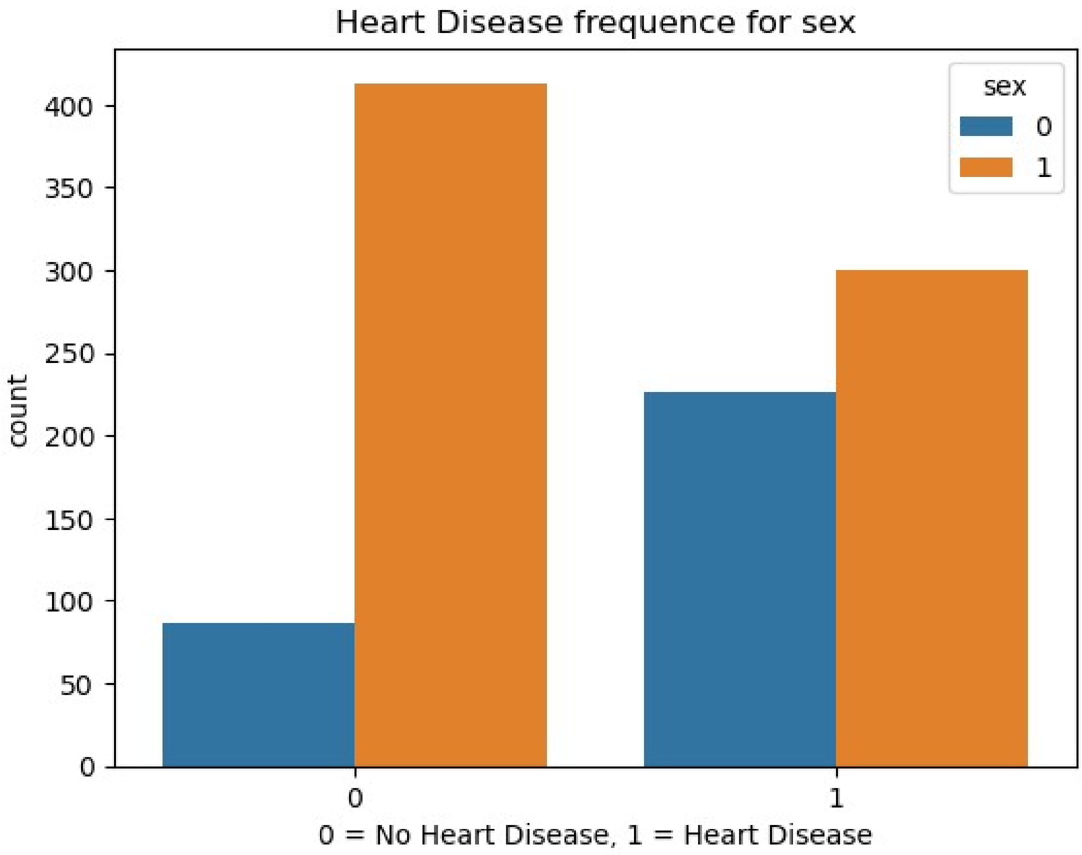
2. People of which sex has most heart disease ?

```
pd.crosstab(data.target,data.sex)
```

```
sex       0    1
target          
0        86  413
1       226  300
```

```
sns.countplot( x = 'target', data = data, hue = 'sex')
plt.title("Heart Disease frequence for sex")
plt.xlabel("0 = No Heart Disease, 1 = Heart Disease")
```

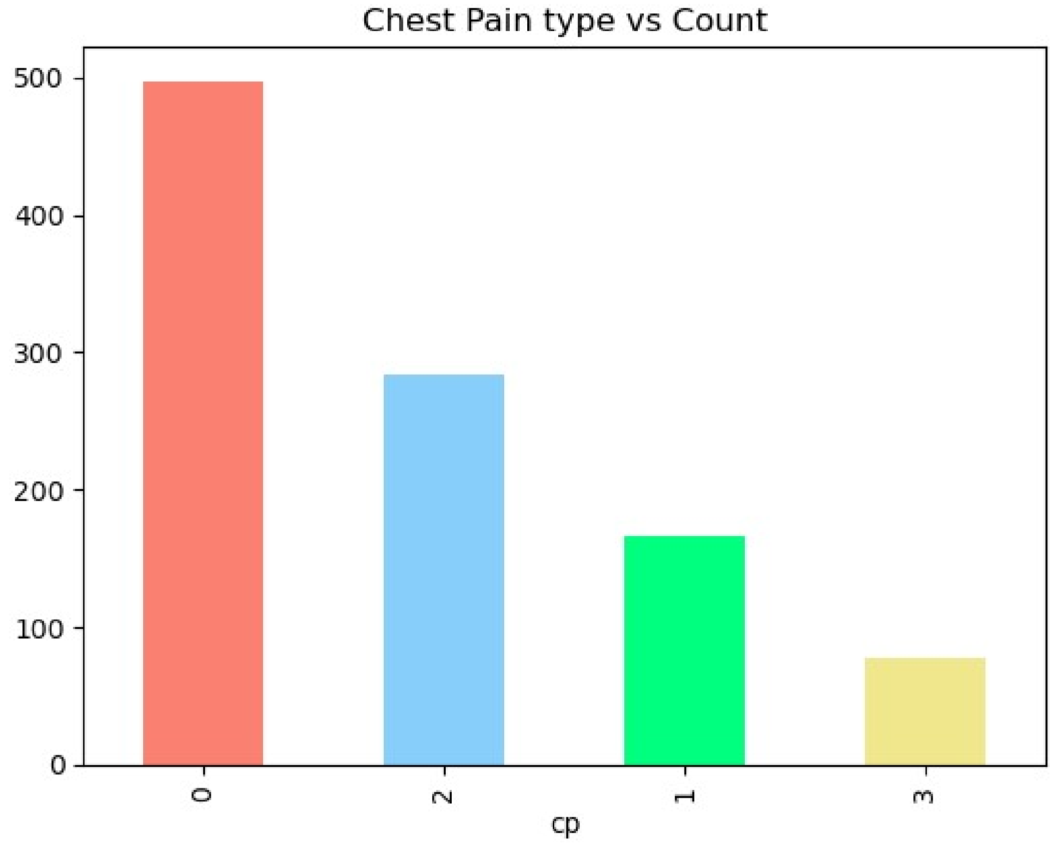
```
Text(0.5, 0, '0 = No Heart Disease, 1 = Heart Disease')
```



3. People of which sex has which type of chest pain most?

```
data.head()

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

   ca  thal  target  
0   2     3       0  
1   0     3       0  
2   0     3       0  
3   1     3       0  
4   3     2       0  
```

```
# Counting the values for different chest pain:

data.cp.value_counts()

cp
0    497
2    284
1    167
3     77
Name: count, dtype: int64

# Plottting a Bar Chart

data.cp.value_counts().plot(kind = 'bar', color =
['salmon','lightskyblue','springgreen','khaki'])
plt.title('Chest Pain type vs Count')

Text(0.5, 1.0, 'Chest Pain type vs Count')
```

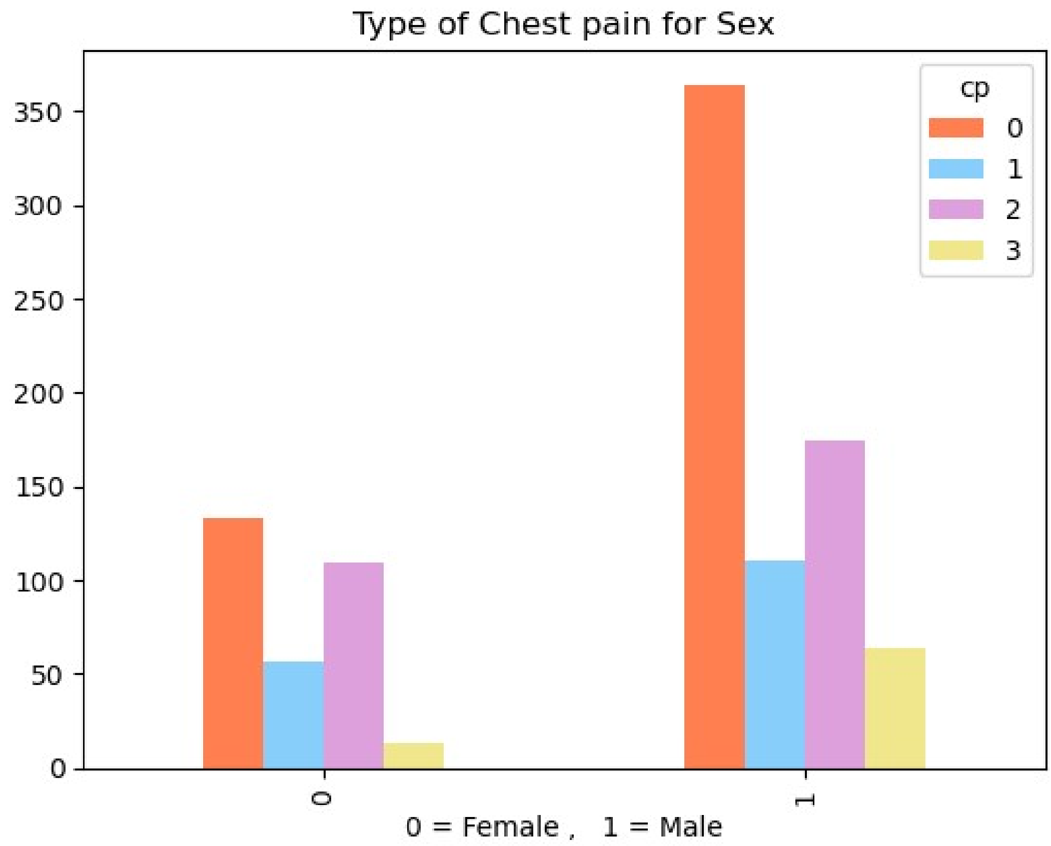


```
pd.crosstab(data.sex, data.cp)

cp      0    1    2   3
sex
```

```
0     133   57  109   13
1     364  110  175   64
```

```
pd.crosstab(data.sex, data.cp).plot(kind = 'bar', color = ['coral',
'lightskyblue', 'plum', 'khaki'])
plt.title('Type of Chest pain for Sex')
plt.xlabel("0 = Female,    1 = Male")
```

```
Text(0.5, 0, '0 = Female ,    1 = Male')
```



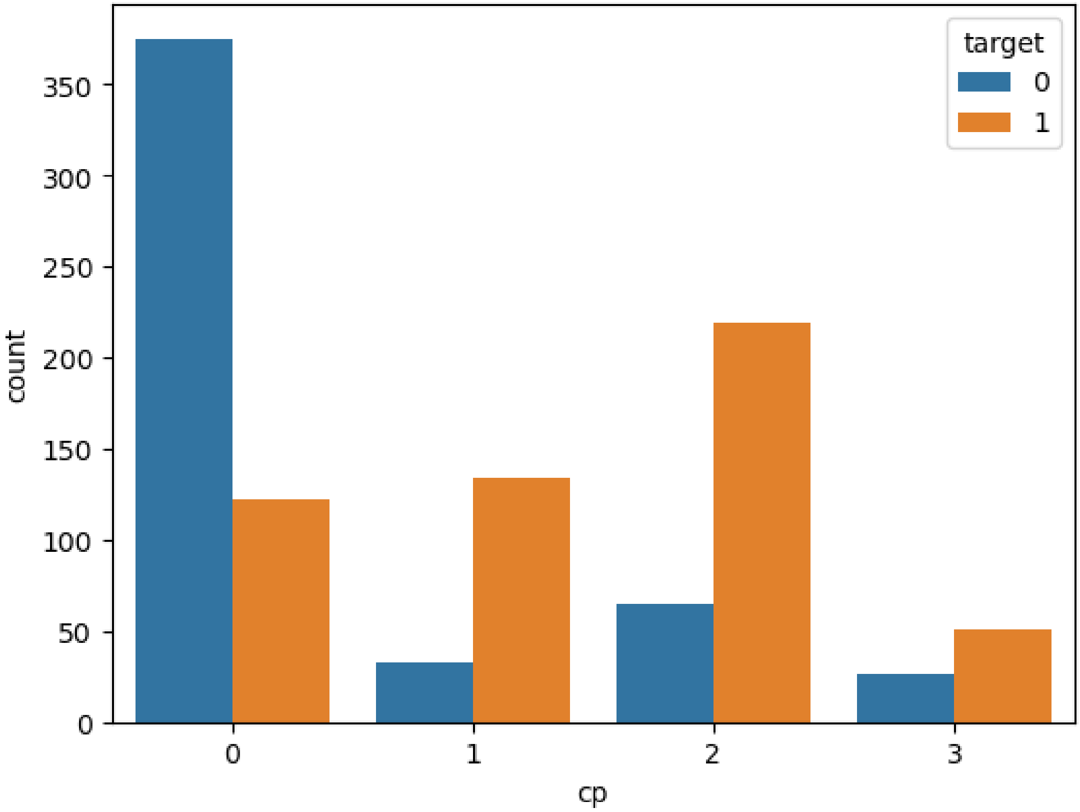
4. People with which chest pain are most pron to have heart disease?

```
pd.crosstab(data.cp, data.target)
```

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

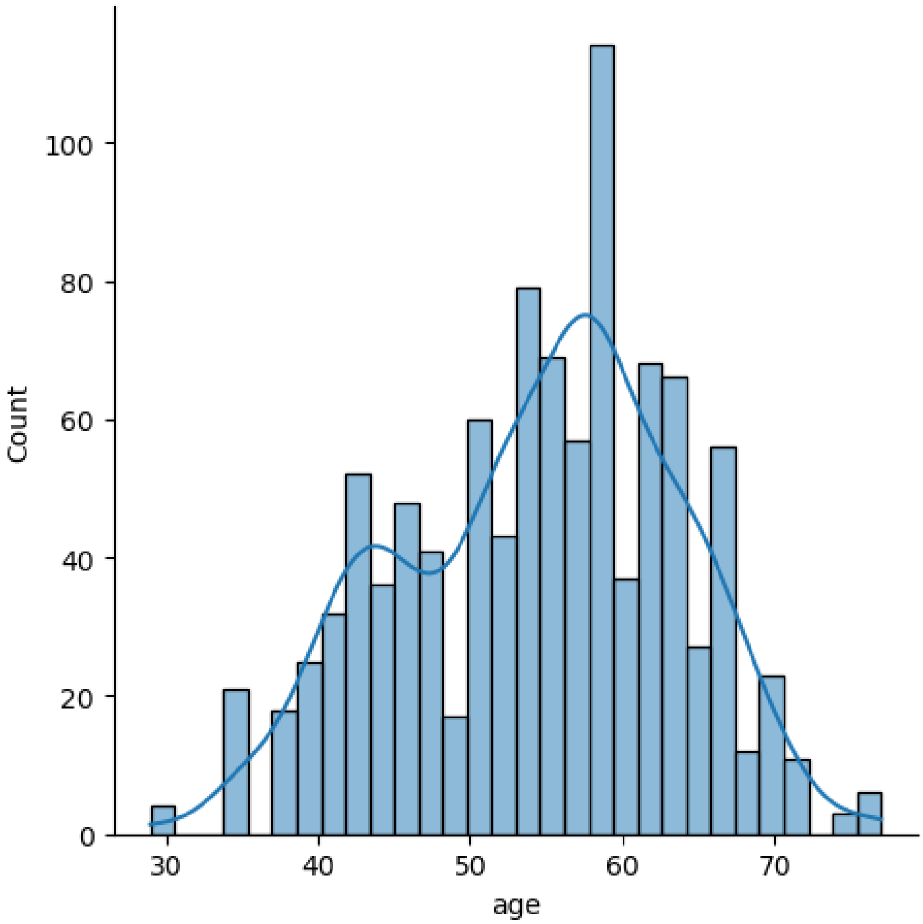
```
sns.countplot(x = 'cp', data = data, hue = 'target')
```

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



```
# Create a Distribution plot with a Normal Distribution curve

sns.displot( x = 'age', data = data, bins = 30, kde = True)
```

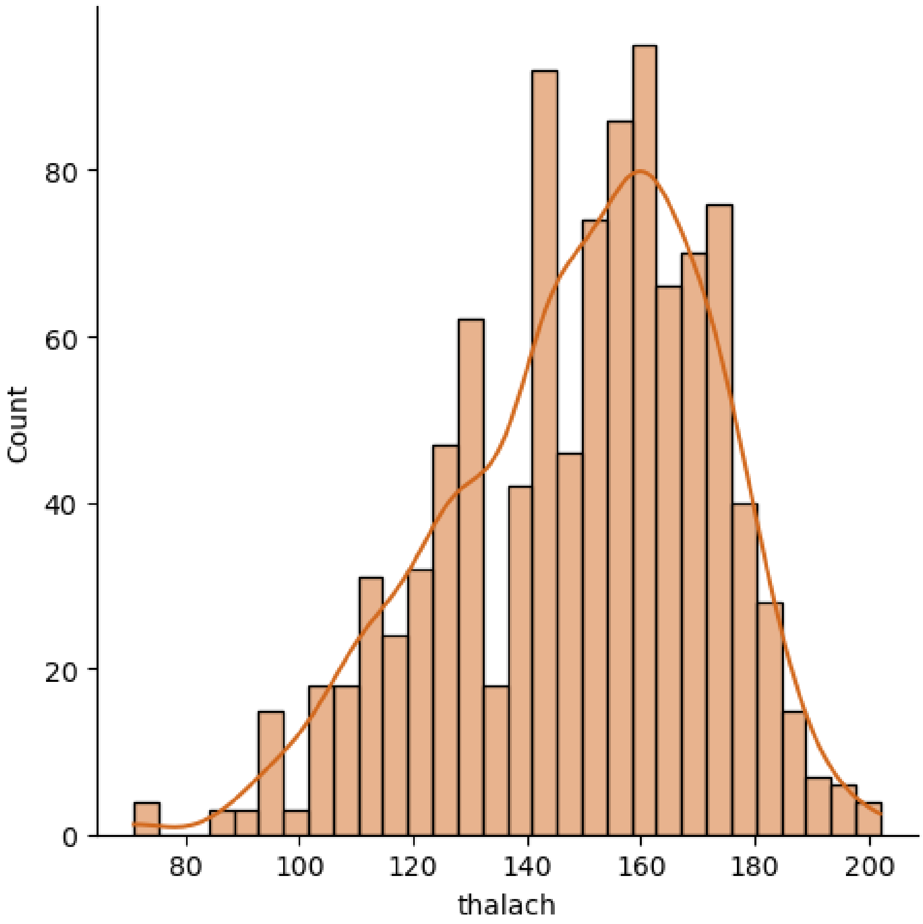
```
C:\Users\Raman\anaconda3\Lib\site-packages\seaborn\axisgrid.py:118:
UserWarning: The figure layout has changed to tight
  self._figure.tight_layout(*args, **kwargs)

<seaborn.axisgrid.FacetGrid at 0x1b8edb47b50>
```



```
# 58-59 years people are more in number according to the data set

sns.displot( x = 'thalach', data = data, bins = 30, kde = True, color
= 'chocolate')
```

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

```
<seaborn.axisgrid.FacetGrid at 0x1b8f313c5d0>
```


Count
0
20
40
60
80
80
100
120
140
160
180
200
thalach