heart Last Checkpoint: 7 minutes ago

View Run Kernel Settings Help

Code JupyterLab

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

```
df = pd.read_csv('C:\\Users\\narasimha\\Downloads\\heart.csv')
```

]:

	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

]:
```
df.tail()
```

	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

]:
```
#take a look at the column names
df.columns.values
```

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

]:
```
#checking for null values
df.isna().sum()
```

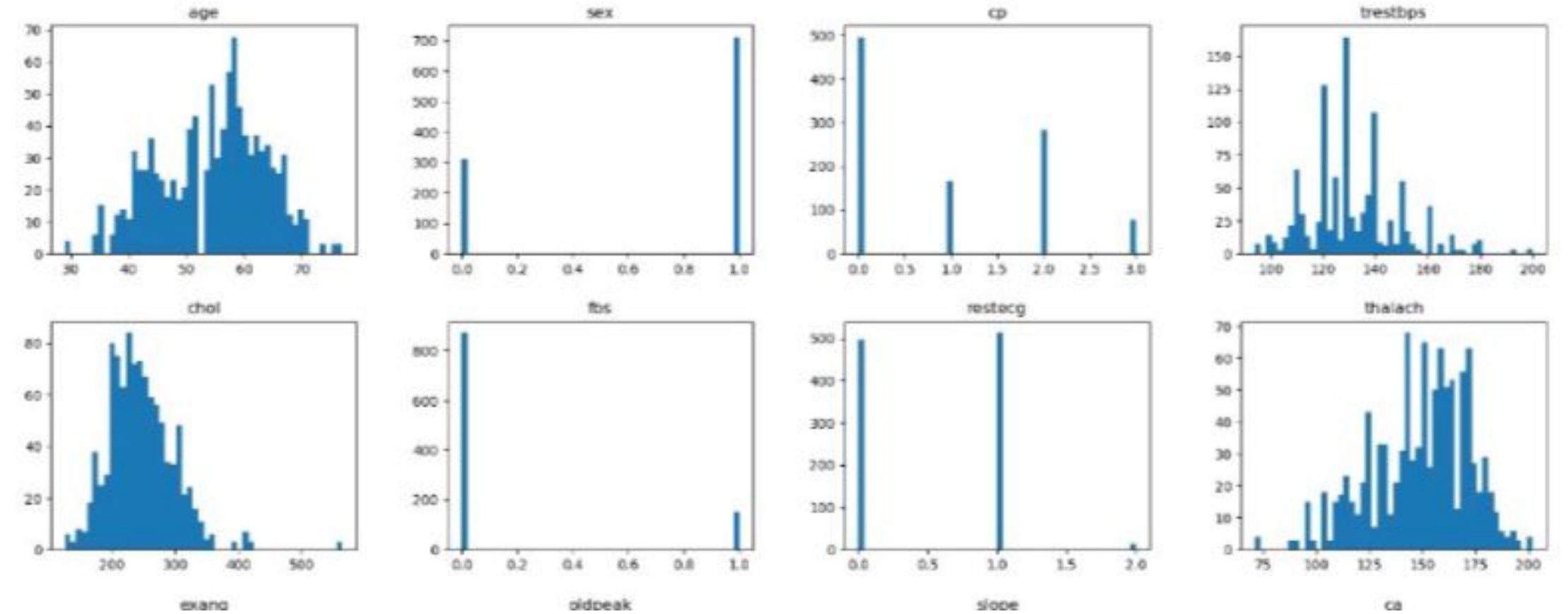
]:
```
age         0
sex         0
cp          0
trestbps    0
chol        0
fbs         0
restecg     0
thalach     0
```

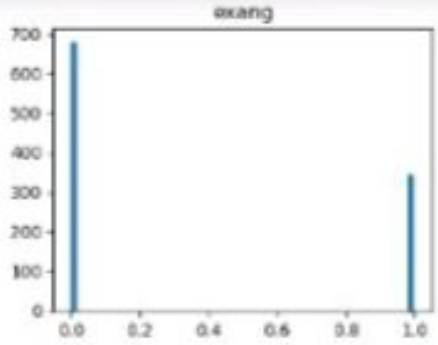
```
exang       0
oldpeak     0
slope       0
ca          0
thal        0
target      0
dtype: int64
```

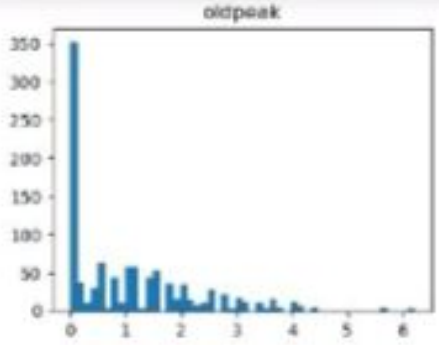
In [8]:
```
#concise summary of our dataset
df.info()
```

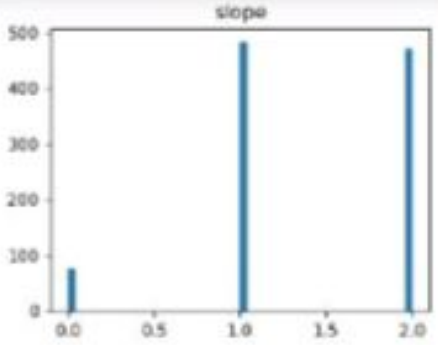
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1025 entries, 0 to 1024
Data columns (total 14 columns):
 #   Column    Non-Null Count  Dtype  
---  ------    --------------  -----  
 0   age       1025 non-null   int64  
 1   sex       1025 non-null   int64  
 2   cp        1025 non-null   int64  
 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
```

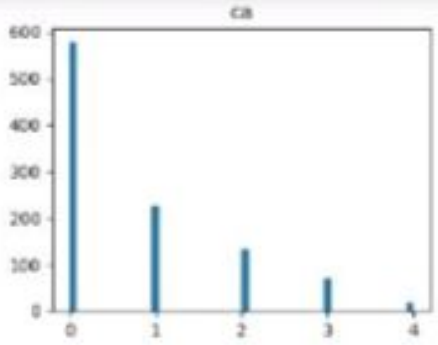
```
#plotting histogram of all numeric values
df.hist(bins = 50, grid = False, figsize=(20,15));
```

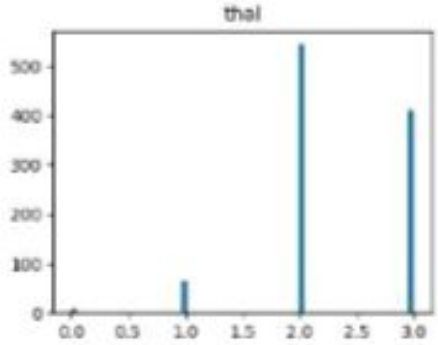


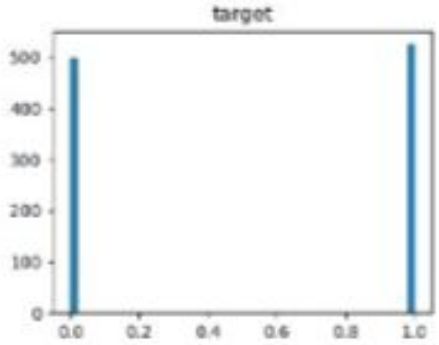













```
: #Generating descriptive statistics
  df.describe()
```

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	
count	1025.000000	1025.000000	1025.000000	1025.000000	1025.00000	1025.000000	1025.000000	1025.000000	1025.000000	1025.000000	1025.000000	1025.00
mean	54.434146	0.695610	0.942439	131.611707	246.00000	0.149268	0.529756	149.114146	0.336585	1.071512	1.385366	0.75
std	9.072290	0.460373	1.029641	17.516718	51.59251	0.356527	0.527878	23.005724	0.472772	1.175053	0.617755	1.03
min	29.000000	0.000000	0.000000	94.000000	126.00000	0.000000	0.000000	71.000000	0.000000	0.000000	0.000000	0.00
25%	48.000000	0.000000	0.000000	120.000000	211.00000	0.000000	0.000000	132.000000	0.000000	0.000000	1.000000	0.00
50%	56.000000	1.000000	1.000000	130.000000	240.00000	0.000000	1.000000	152.000000	0.000000	0.800000	1.000000	0.00
75%	61.000000	1.000000	2.000000	140.000000	275.00000	0.000000	1.000000	166.000000	1.000000	1.800000	2.000000	1.00
max	77.000000	1.000000	3.000000	200.000000	564.00000	1.000000	2.000000	202.000000	1.000000	6.200000	2.000000	4.00

1]:
```
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?"
             "5. What Dietary Changes Can I Make to Reduce My Heart Disease Risk?"
             "6. How Common Is Heart Disease Among Women?"
             "7. What treatment options do I have for heart disease?"]
questions
```

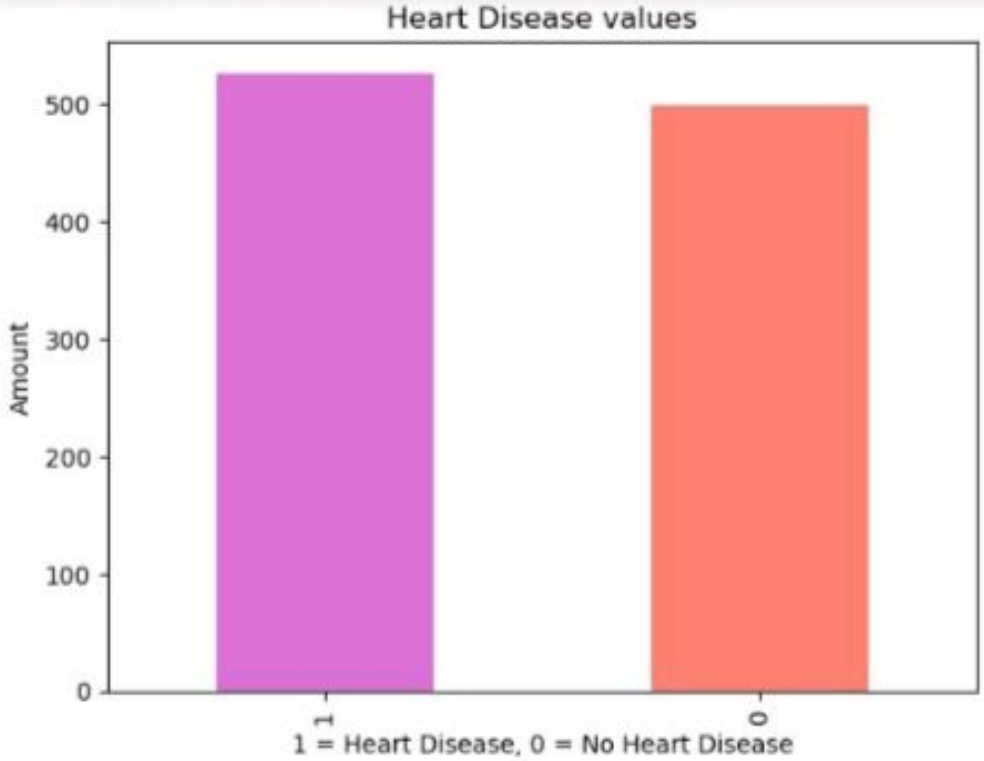
1]:
```
["1. How many people have heart disease and how many people doesn't have heart disease?2. People of which sex has most heart di
sease?3. People of which sex has which type of chest pain most?4. People with which chest pain are most pron to have heart dise
ase?5. What Dietary Changes Can I Make to Reduce My Heart Disease Risk?6. How Common Is Heart Disease Among Women?7. What treat
```

ment options do I have for heart disease?"]

2]:
```
#Let's find the answer of first question.

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

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

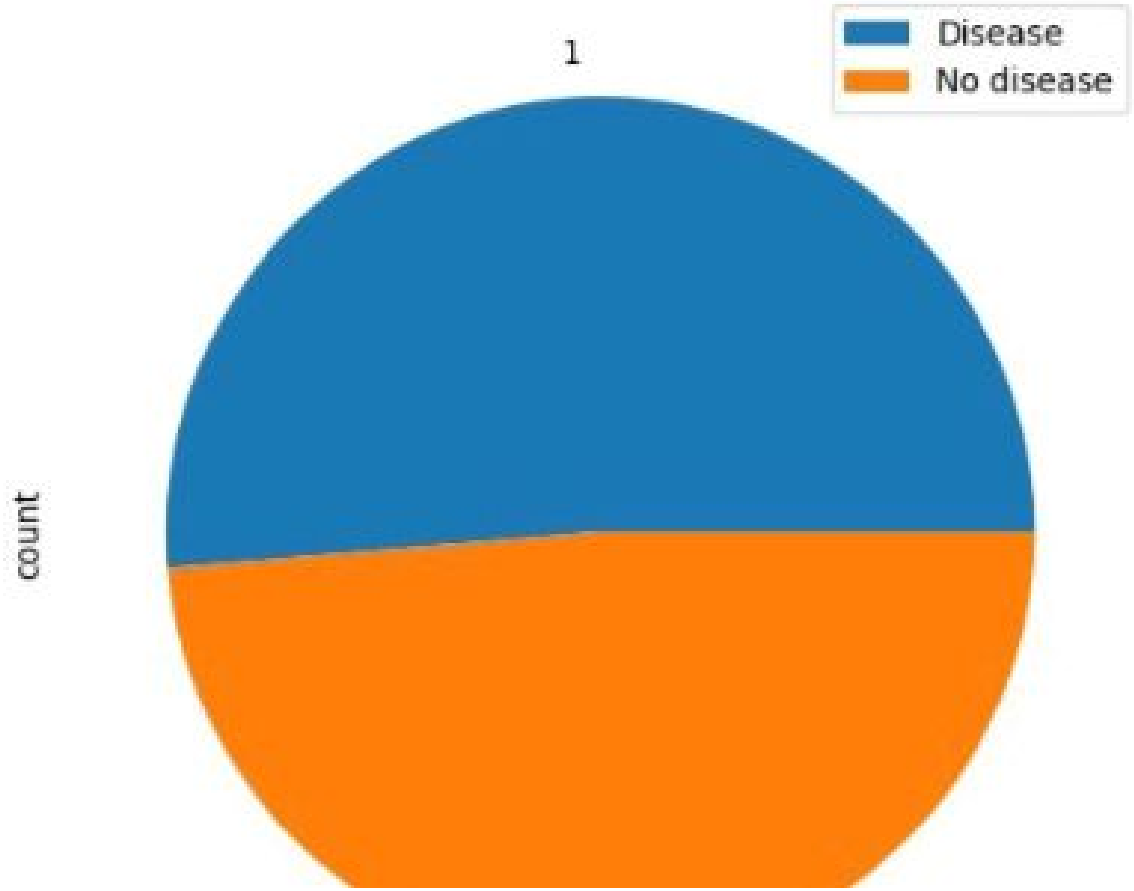
2]:
```
target
1    526
0    499
Name: count, dtype: int64
```

3]:
```
#plotting bar chart
df.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");
```



In [14]: #plotting a pie chart

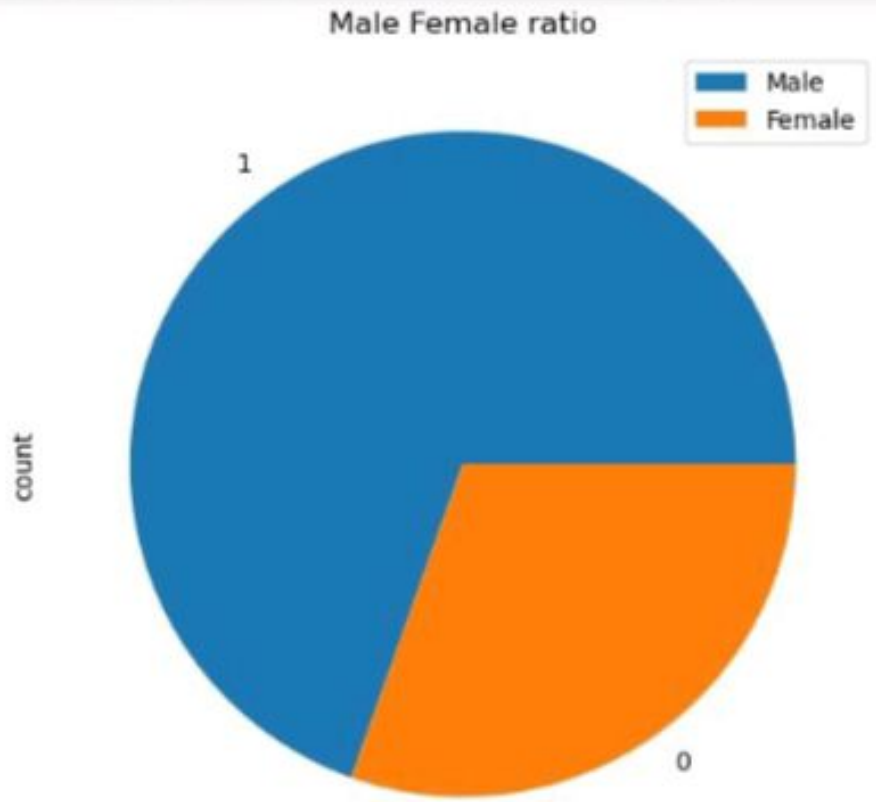
4]:
```
#plotting a pie chart
df.target.value_counts().plot(kind = 'pie', figsize = (8, 6))
plt.legend(["Disease", "No disease"]);
```



```
#'0' represent 'Female'

#'1' represent 'Male'

#SEX column part

#'0' represent 'No disease'

#'1' represent 'Disease'

#Target column part

#Now Let's check how many 'Male' and 'Female' are in the dataset

df.sex.value_counts()
```

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

```
#plotting a pie chart
df.sex.value_counts().plot(kind = 'pie', figsize = (8, 6))
plt.title('Male Female ratio')
plt.legend(['Male', 'Female']);
```


Male Female ratio
Male
Female
1
0
count

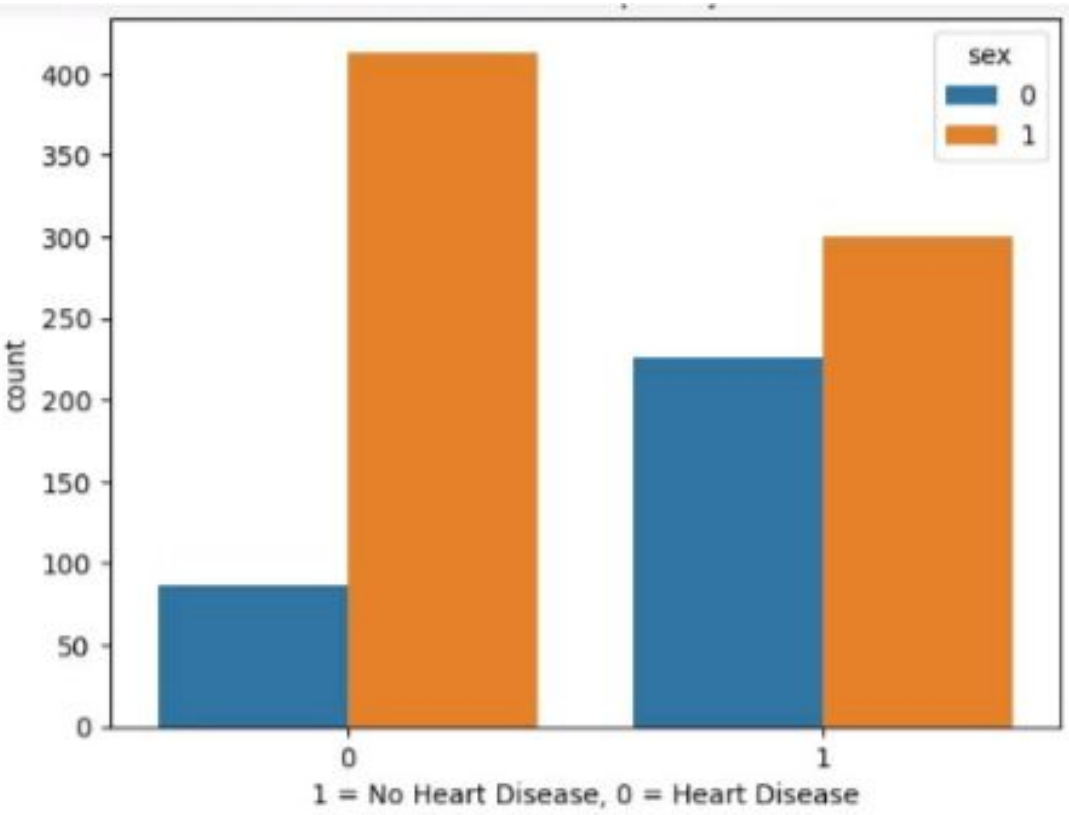
]:
```
#Let's find the answer of our 2nd question

#2.People of which sex has most heart disease?

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

]:

target	sex 0	1
0	86	413
1	226	300

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

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



20]: `#Number of male is more than double in our dataset than female.`

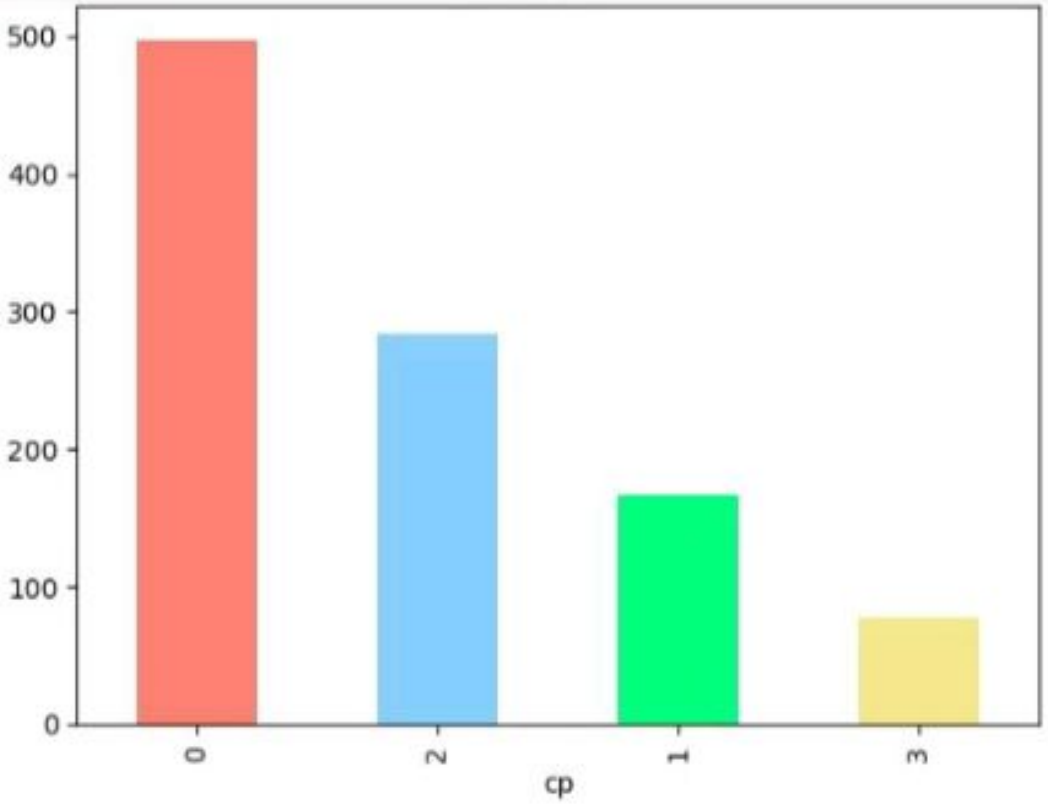
```
#More than 45% male has heart disease and 75% female has heart disease
```

27]:
```
#Let's find the answer of our 3rd question

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

df.cp.value_counts()
```

27]:
```
cp
0    497
2    284
1    167
3     77
Name: count, dtype: int64
```

28]:
```
#plotting a bar chart
df.cp.value_counts().plot(kind = 'bar', color = ['salmon', 'lightskyblue', 'springgreen', 'khaki'])
plt.title('Chest pain type vs count');
```

Chest pain type vs count

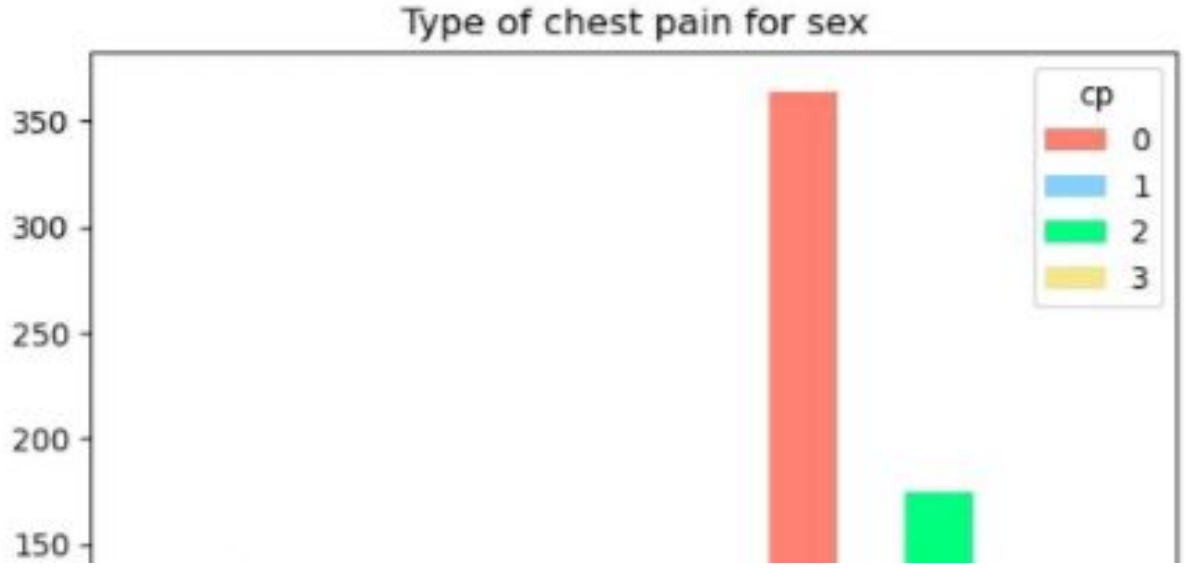



```
]: pd.crosstab(df.sex, df.cp)
```

29]:

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

30]:
```
pd.crosstab(df.sex, df.cp).plot(kind= 'bar', color= ['salmon', 'lightskyblue', 'springgreen', 'khaki'])
plt.title('Type of chest pain for sex')
plt.xlabel('0 = Female, 1 = Male');
```

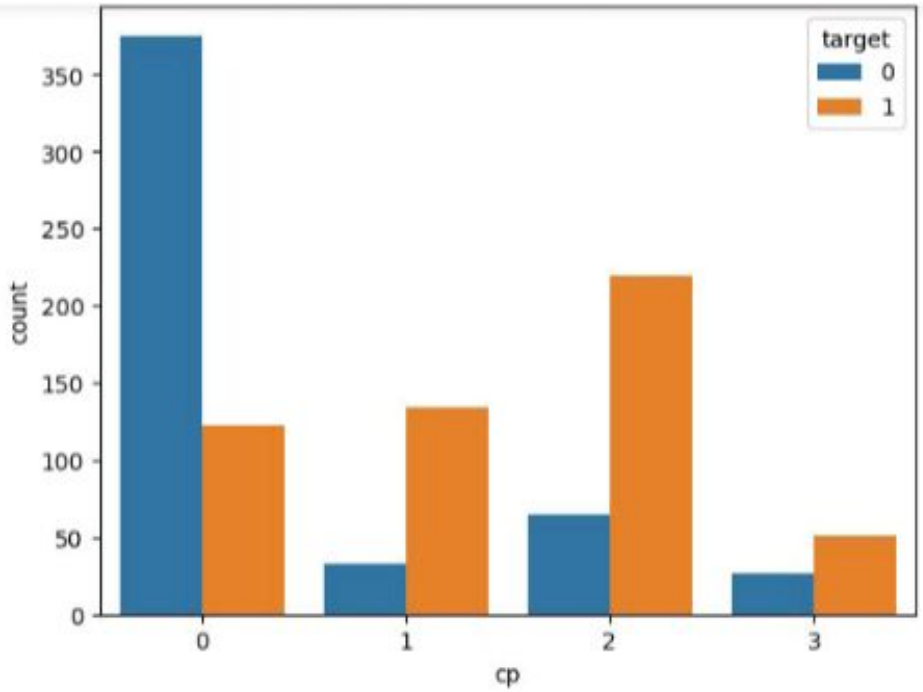


```
#Most of male has 'type 0' chest pain and least of 'Male' has 'type 4' pain.

#in case of 'Female' 'type 0' and 'type 2' percentage is almost same
```

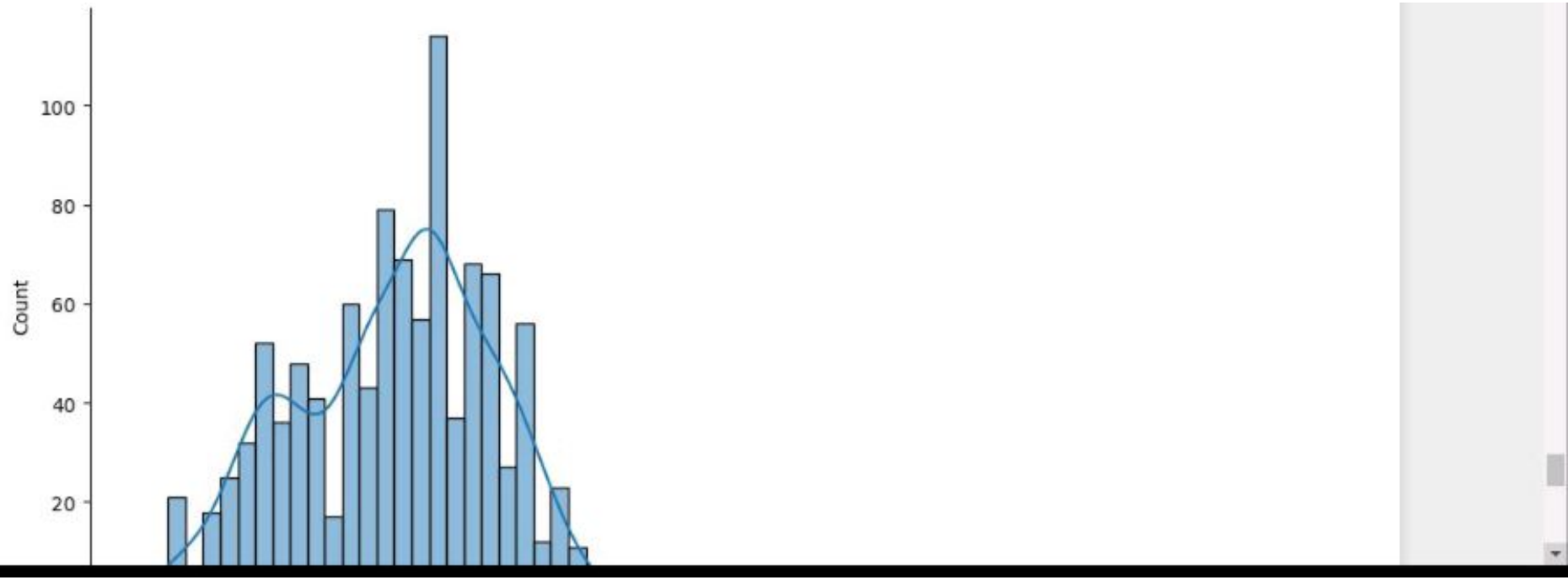
```
#Let's find the answer of our 4th question

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

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

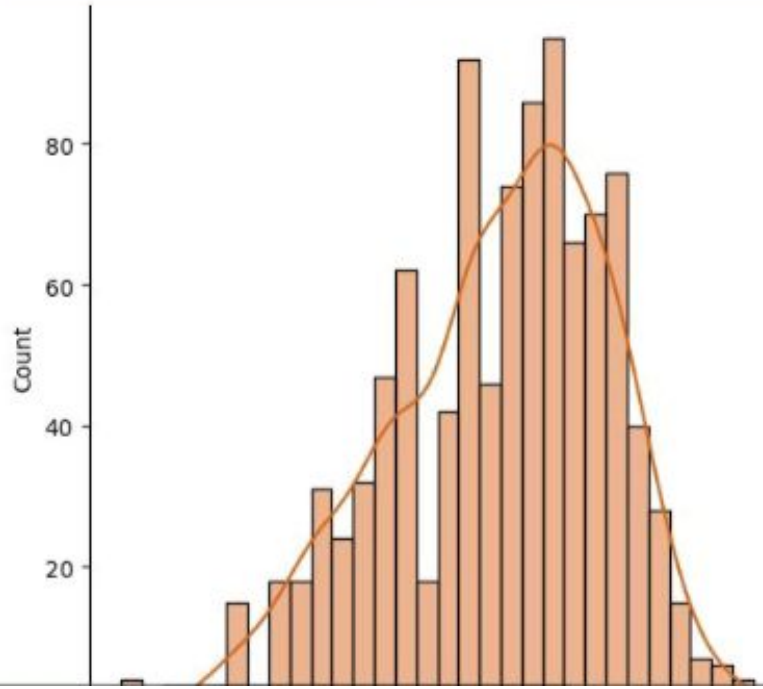
target	0	1
cp		
0	375	122
1	33	134
2	65	219
3	26	51

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



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


Count
100
80
60
40
20


Count
80
60
40
20