Task 5

By Mallela Preethi

Importing all dependencies

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

```python
[2] from google.colab import files
    uploaded = files.upload()
```

Choose Files heart.csv
- **heart.csv**(text/csv) - 38114 bytes, last modified: 6/27/2024 - 100% done
Saving heart.csv to heart.csv

```python
df = pd.read_csv('heart.csv') # Reading file
df.head()
```

	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

```
[7] df.shape # Knowing the shape of dataframe(no.of rows and columns)
```

(1025, 14)

This dataset contains 1025 records with 14 columns

```
df.info() #concise summary of our dataset
```

```
<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
```

There are no missing values

```python
df.describe() # Summary statistics
```

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
count	1025.000000	1025.000000	1025.000000	1025.000000	1025.00000	1025.000000	1025.000000	1025.000000	1025.000000	1025.000000	1025.000000	1025.000000	1025.000000	1025.000000
mean	54.434146	0.695610	0.942439	131.611707	246.00000	0.149268	0.529756	149.114146	0.336585	1.071512	1.385366	0.754146	2.323902	0.513171
std	9.072290	0.460373	1.029641	17.516718	51.59251	0.356527	0.527878	23.005724	0.472772	1.175053	0.617755	1.030798	0.620660	0.500070
min	29.000000	0.000000	0.000000	94.000000	126.00000	0.000000	0.000000	71.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
25%	48.000000	0.000000	0.000000	120.000000	211.00000	0.000000	0.000000	132.000000	0.000000	0.000000	1.000000	0.000000	2.000000	0.000000
50%	56.000000	1.000000	1.000000	130.000000	240.00000	0.000000	1.000000	152.000000	0.000000	0.800000	1.000000	0.000000	2.000000	1.000000
75%	61.000000	1.000000	2.000000	140.000000	275.00000	0.000000	1.000000	166.000000	1.000000	1.800000	2.000000	1.000000	3.000000	1.000000
max	77.000000	1.000000	3.000000	200.000000	564.00000	1.000000	2.000000	202.000000	1.000000	6.200000	2.000000	4.000000	3.000000	1.000000

```python
df.columns # Knowing all the columns
```

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

```
questions=["1. How many people have heart disease problem and how many doesn't have heart problem? ",
           "2. People of which sex has most heart problem? ",
           "3. People of which sex has which types of chest pain most? ",
           "4. People with which chest pain are most prone to have heart failure?"]
questions

["1. How many people have heart disease problem an dhow many doesn't have heart problem? ",
 '2. People of which sex has most heart problem? ',
 '3. People of which sex has which types of chest pain most? ',
 '4. People with which chest pain are most prone to have heart failure?']
```
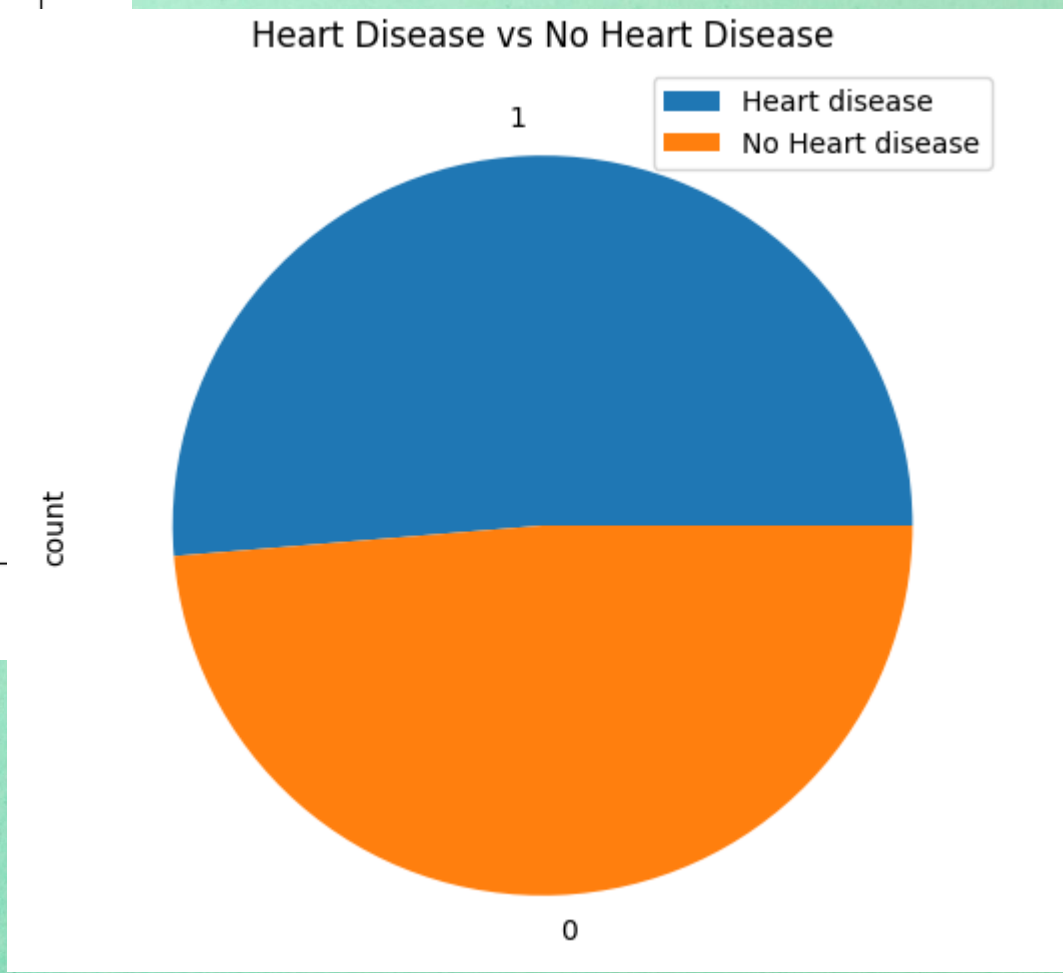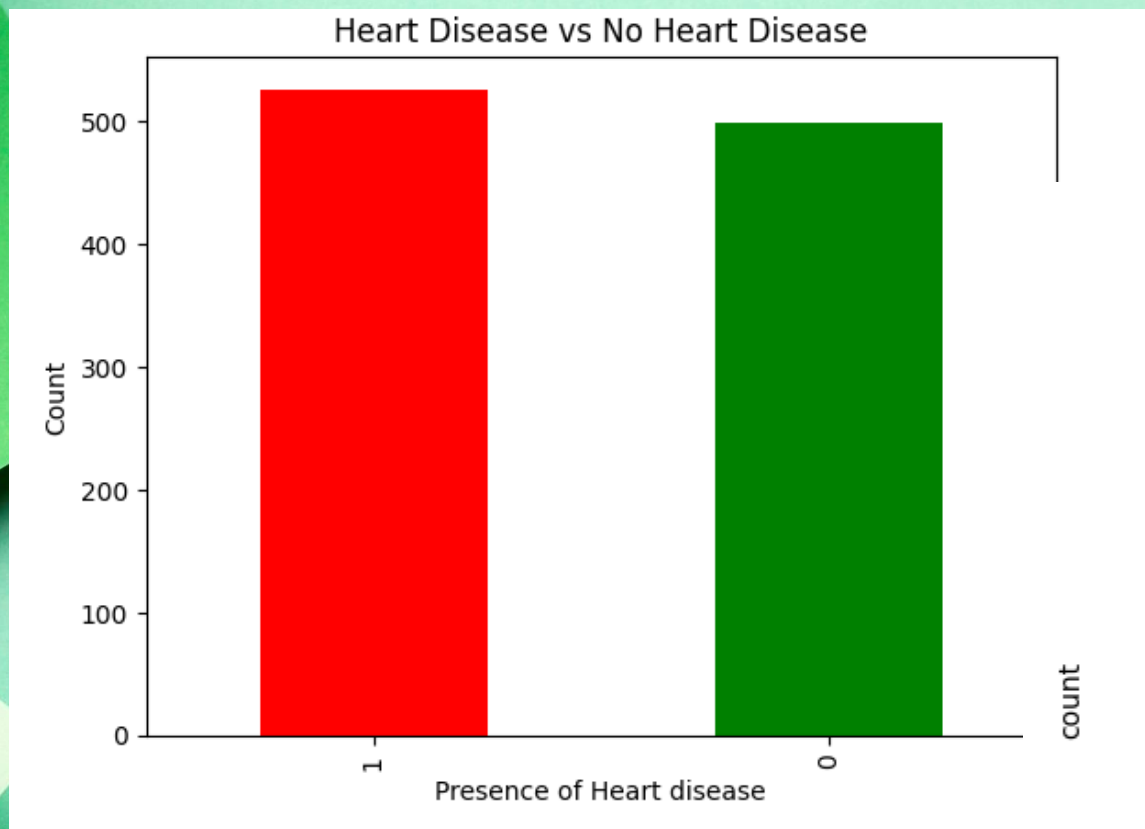
Answering all these questions:

```
# 1 How many people have heart disease problem and how many people doesn't have heart problem? "
df.target.value_counts()

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

Heart Disease vs No Heart Disease

Count

500

400

300

200

100

0

1   0

Presence of Heart disease

Heart Disease vs No Heart Disease
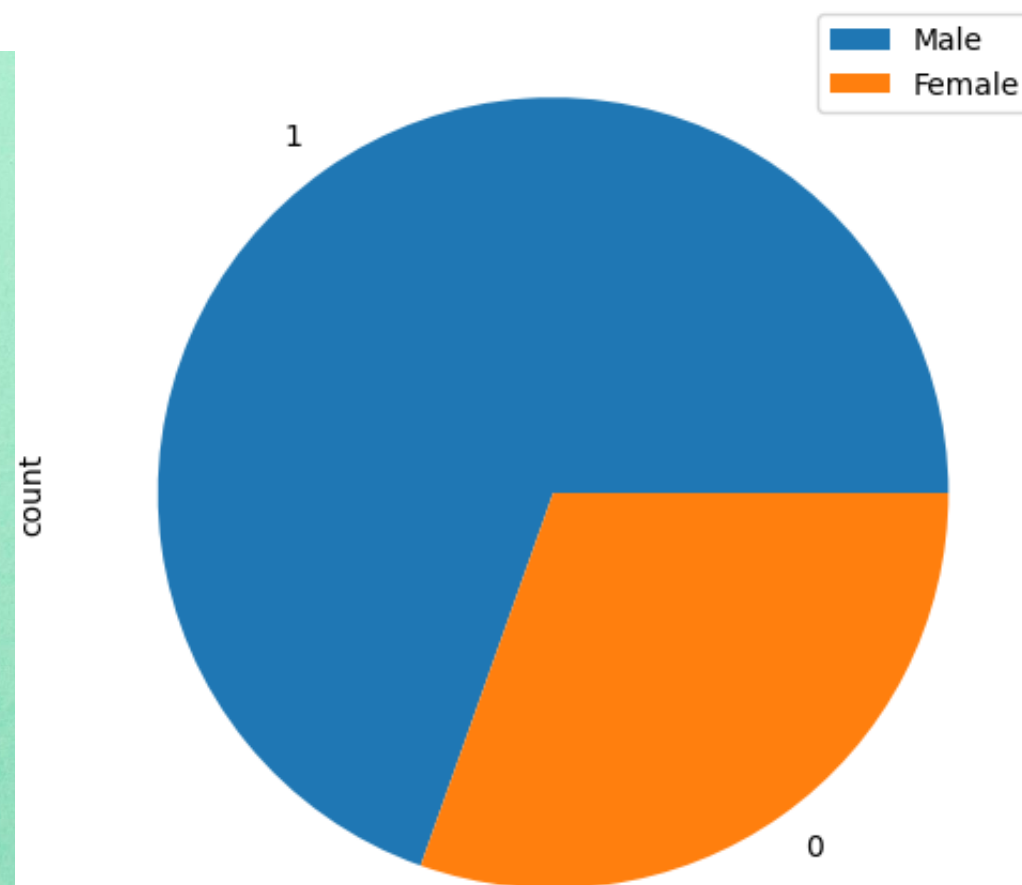
Heart disease
No Heart disease

1

count

0

```
# '0' in sex column represents 'Female'
# '1' in sex column represents 'Male'
# '0' in target column represents 'No Disease problem'
# '1' in target column represents 'Disease problem'
# 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
```

Male Female Ratio

```
#2. People of which sex has most heart problem?

# crosstab is used to analyse the relationship between two or more  categorical varables
pd.crosstab(df.target, df.sex)
```

sex	0	1
target		
0	86	413
1	226	300

Heart problems variation according to sex

```
[ ]   # Number of male is  more than double in our dataset than female.
      # More than '45% male' has heart problem and '75% female' has heart problem.
```
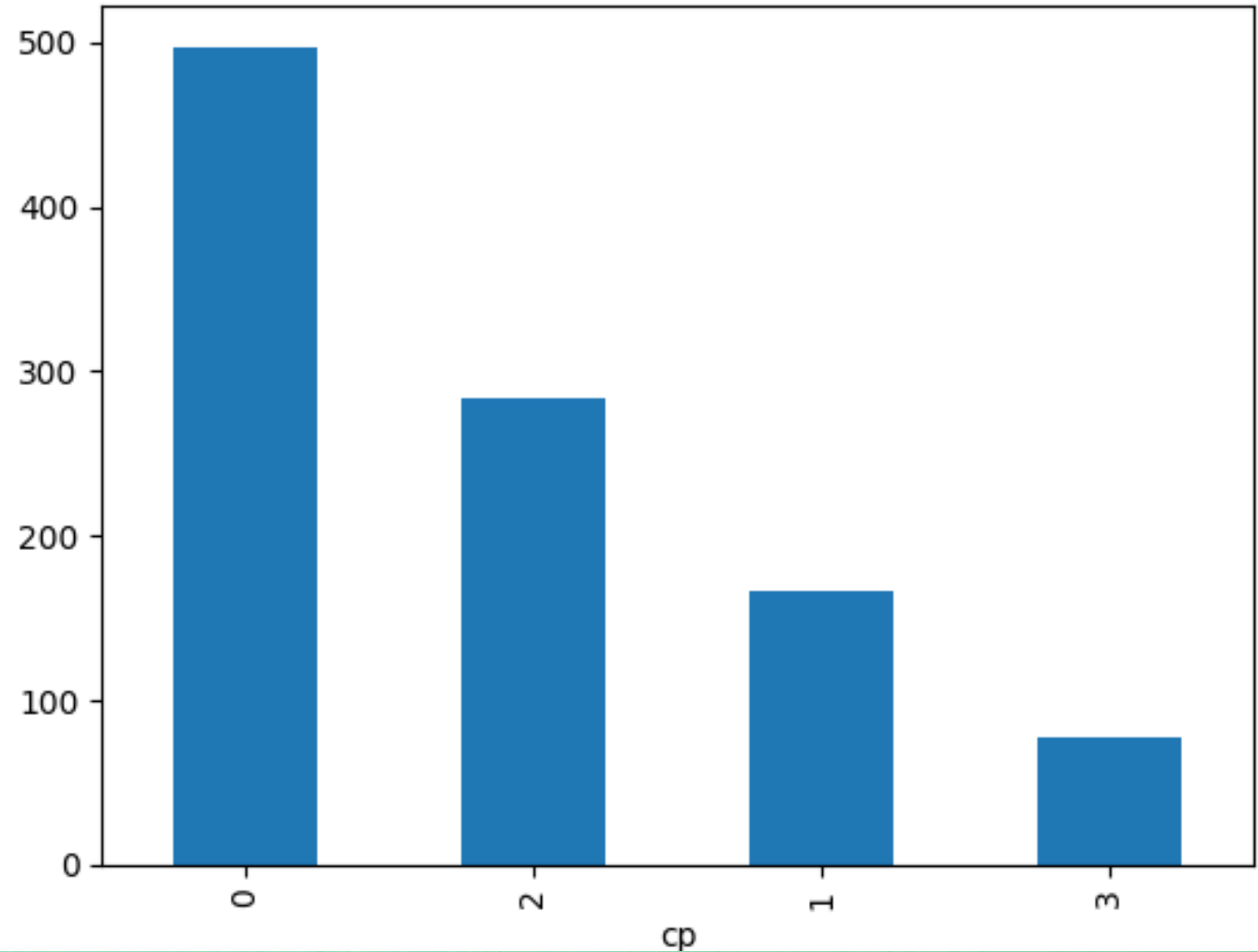
```
#3. People of which sex has which types of chest pain most?
#counting values for different chest pain.

df.cp.value_counts()

cp
0    497
2    284
1    167
3     77
Name: count, dtype: int64
```
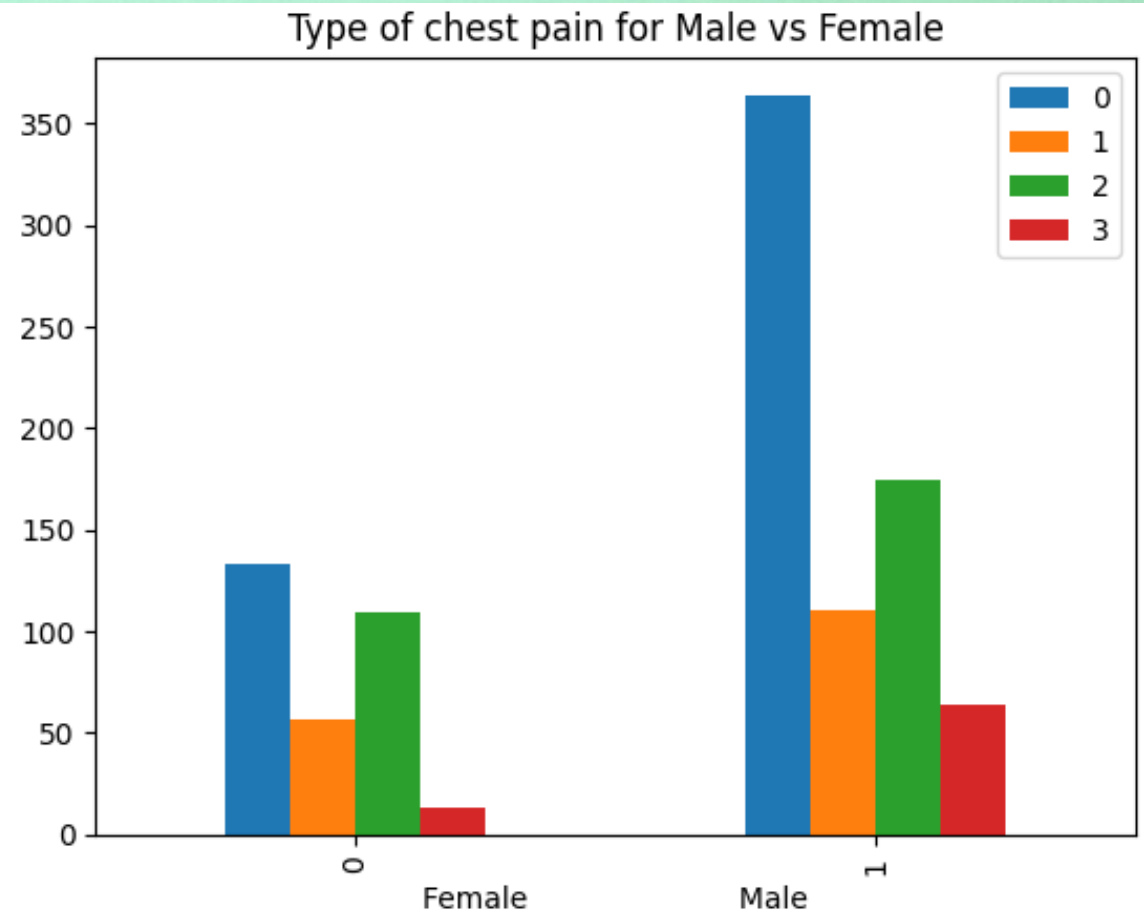
Chest pain type vs count

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

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

Type of chest pain for Male vs Female

```
#Most of 'Male' has 'type 0' chest pain and least of 'Male' has 'type 4' chest pain.
#In case of 'Female', 'type 0' and 'type 2' pain presence percentage is almost same.
```

```
#4. People with which chest pain are most prone to have heart failure?
pd.crosstab(df.cp, df.target)
```

target	0	1
cp		
0	375	122
1	33	134
2	65	219
3	26	51
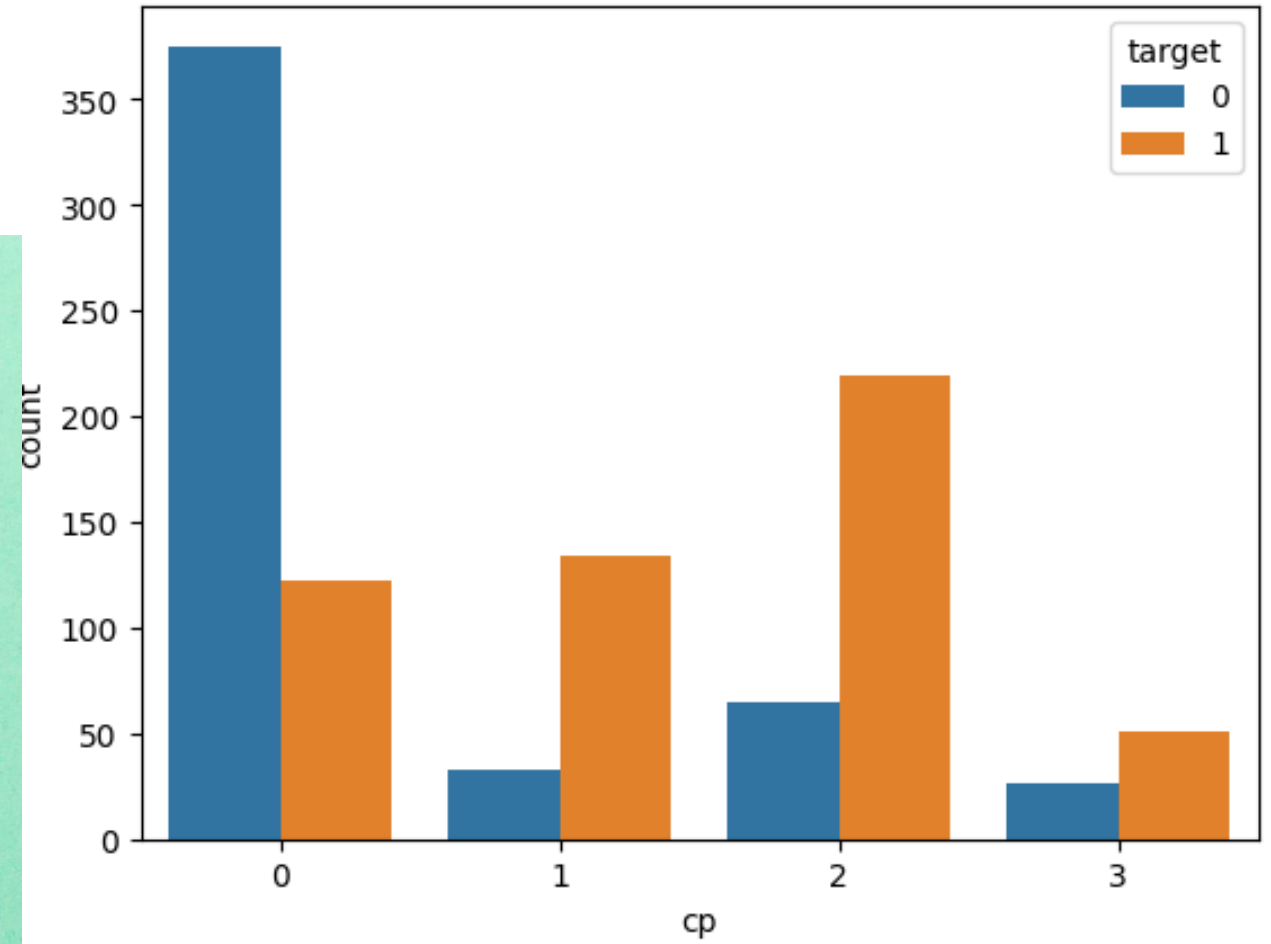
#Most of people who has 'type 0' chest pain has less chance of heart failure and we see the opposite for other types.