

```
[1]: #Importing all the libraries that we need.
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline

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

[3]: df.head()

[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

[4]: df.tail()

[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

[7]: df.columns.values

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

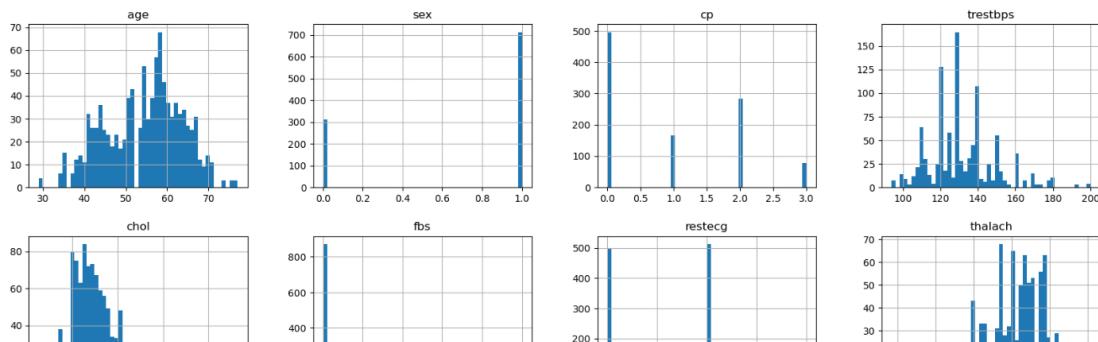
[8]: df.isna().sum()

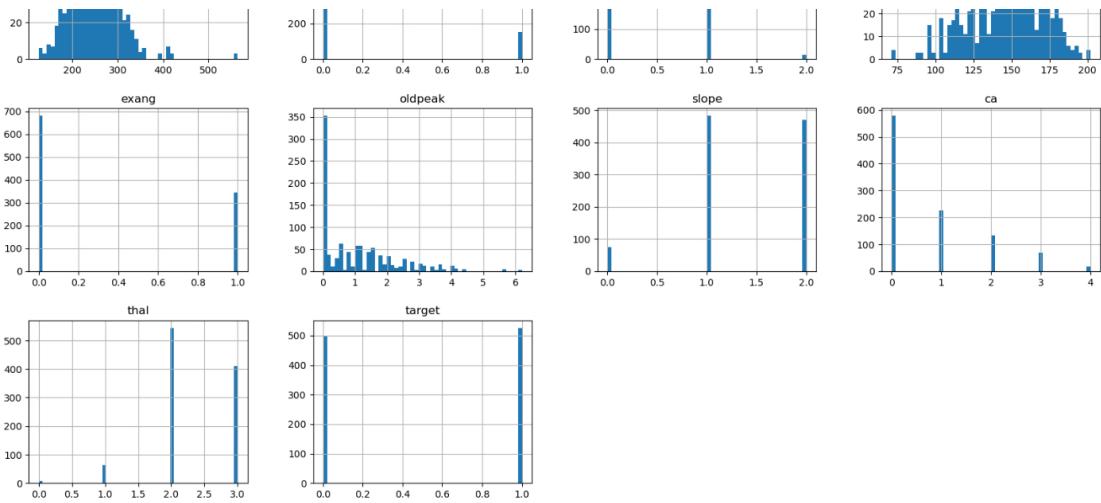
[8]:   age      0
       sex      0
       cp      0
       trestbps  0
       chol      0
       fbs      0
       restecg  0
       thalach  0
       exang      0
       oldpeak  0
       slope      0
       ca      0
       thal      0
       target     0
       dtype: int64

[9]: 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
```

```
[15]: df.hist(bins = 50, grid = True, figsize = (20,15));
```





```
[16]: df.describe()
```

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca
<b>count</b>	1025.000000	1025.000000	1025.000000	1025.000000	1025.000000	1025.000000	1025.000000	1025.000000	1025.000000	1025.000000	1025.000000	1025.000000
<b>mean</b>	54.434146	0.695610	0.942439	131.611707	246.000000	0.149268	0.529756	149.114146	0.336585	1.071512	1.385366	0.754146
<b>std</b>	9.072290	0.460373	1.029641	17.516718	51.59251	0.356527	0.527878	23.005724	0.472772	1.175053	0.617755	1.030798
<b>min</b>	29.000000	0.000000	0.000000	94.000000	126.000000	0.000000	0.000000	71.000000	0.000000	0.000000	0.000000	0.000000
<b>25%</b>	48.000000	0.000000	0.000000	120.000000	211.000000	0.000000	0.000000	132.000000	0.000000	0.000000	1.000000	0.000000
<b>50%</b>	56.000000	1.000000	1.000000	130.000000	240.000000	0.000000	1.000000	152.000000	0.000000	0.800000	1.000000	0.000000
<b>75%</b>	61.000000	1.000000	2.000000	140.000000	275.000000	0.000000	1.000000	166.000000	1.000000	1.800000	2.000000	1.000000
<b>max</b>	77.000000	1.000000	3.000000	200.000000	564.000000	1.000000	2.000000	202.000000	1.000000	6.200000	2.000000	4.000000

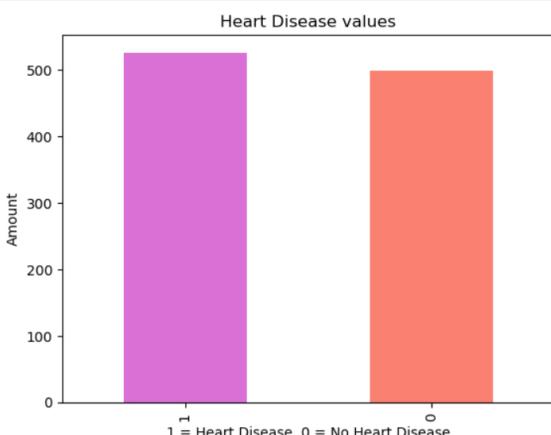
```
[16]: df.describe()
```

```
[17]: 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. People of what age has most heart disease?",  
    "6. People of what age has most chest pain?"]
```

```
[18]: # question 1  
df.target.value_counts()
```

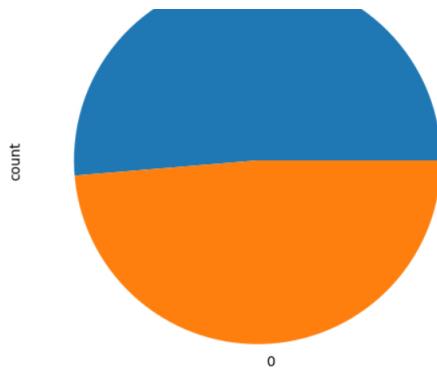
```
[18]: target  
1    526  
0    499  
Name: count, dtype: int64
```

```
[54]: import matplotlib.pyplot as plt  
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")  
plt.show()
```



```
[24]: df.target.value_counts().plot(kind = 'pie', figsize = (8,6))  
plt.legend(["Disease","No Disease"]);
```

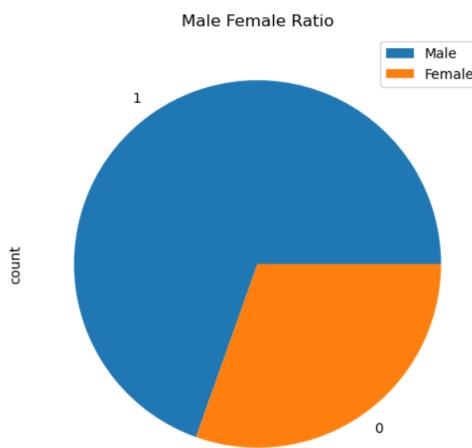




```
[25]: # "0" represent 'female'
# "1" represent 'male'
# '0' represent 'No disease'
# '1' represent 'Disease'
df.sex.value_counts()
```

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

```
[26]: df.sex.value_counts().plot(kind = 'pie', figsize = (8,6))
plt.title('Male Female Ratio')
plt.legend(['Male','Female']);
```

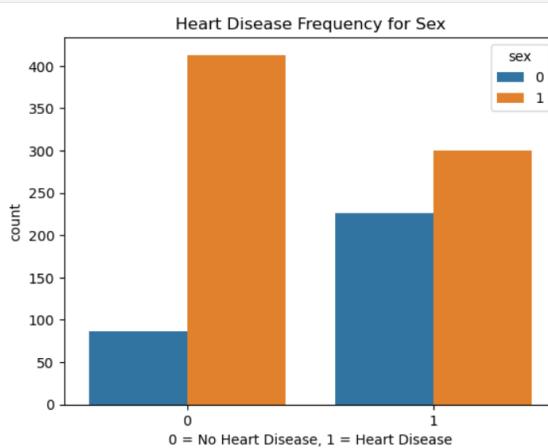


```
[27]: # question 2
pd.crosstab(df.target, df.sex)
```

target	0	1
0	86	413
1	226	300

```
[55]: import seaborn as sns
import matplotlib.pyplot as plt

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

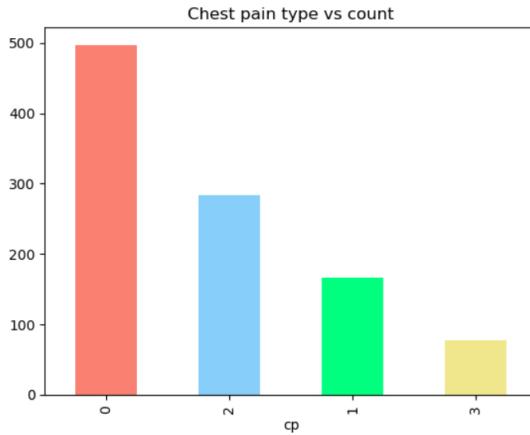


```
[55]: # question 3
```

```
[31]: df.cp.value_counts()

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

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

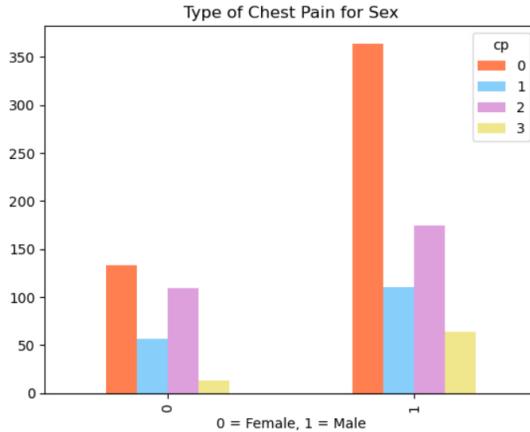


```
[37]: pd.crosstab(df.sex,df.cp)

[37]: cp   0   1   2   3
sex
0    133  57  109  13
1    364  110  175  64
```

```
[56]: import pandas as pd
import matplotlib.pyplot as plt

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



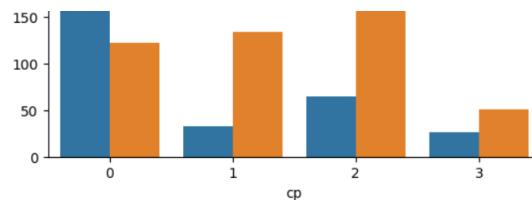
```
[40]: # question 4
pd.crosstab(df.cp,df.target)

[40]: target    0   1
cp
0    375  122
1     33  134
2     65  219
3     26   51
```

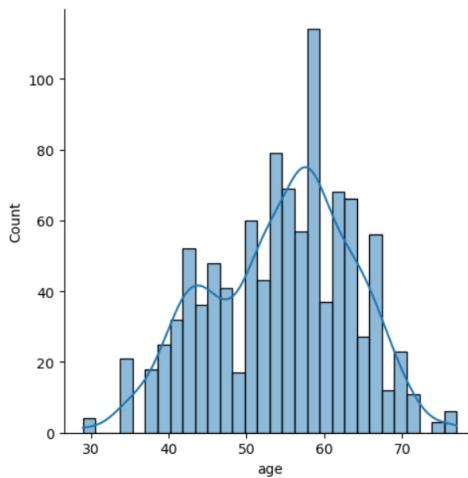
```
[41]: sns.countplot(x = 'cp', data = df, hue ='target')

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

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

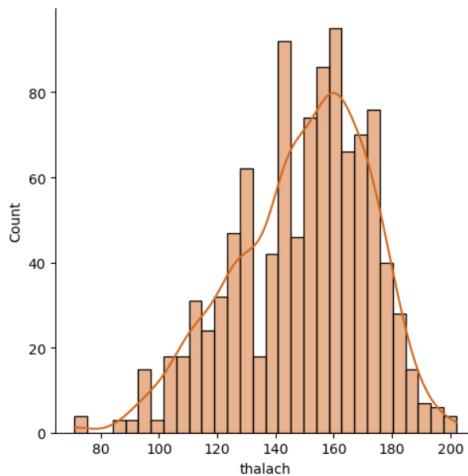


```
[42]: # create distribution plot with normal distribution curve
sns.displot(x = 'age', data=df, bins=30, kde=True);
```



```
[46]: # 58-59 year old people are most in the dataset.
```

```
# Let's plot another distribution plot for Maximum heart rate
sns.displot(x='thalach', data=df, bins=30, kde=True, color='chocolate');
```



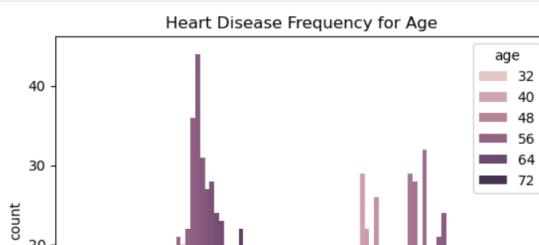
```
[47]: # question 5
pd.crosstab(df.target, df.age)
```

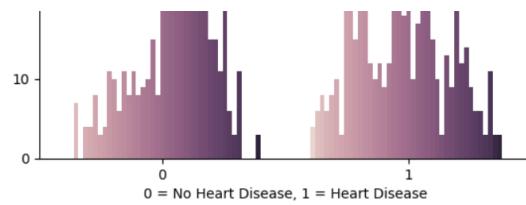
```
[47]: age 29 34 35 37 38 39 40 41 42 43 ... 65 66 67 68 69 70 71 74 76 77
      target
      0 0 0 7 0 4 4 8 3 4 11 ... 15 11 22 6 3 11 0 0 0 3
      1 4 6 8 6 8 10 3 29 22 15 ... 12 14 9 6 6 3 11 3 3 0
```

2 rows × 41 columns

```
[57]: import seaborn as sns
import matplotlib.pyplot as plt

sns.countplot(x='target', data=df, hue='age')
plt.title('Heart Disease Frequency for Age')
plt.xlabel('0 = No Heart Disease, 1 = Heart Disease')
plt.show()
```





```
[49]: # question 6
df.cp.value_counts()
```

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

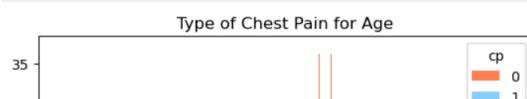
```
[50]: pd.crosstab(df.age,df.cp)
```

```
[50]: cp  0   1   2   3
```

age	0	1	2	3
29	0	4	0	0
34	0	3	0	3
35	11	4	0	0
37	0	0	6	0
38	0	0	8	4
39	4	0	10	0
40	8	0	0	3
41	3	20	9	0
42	10	3	10	3
43	20	0	6	0
44	10	10	16	0
45	12	10	0	3
46	11	6	6	0
47	7	0	11	0
48	10	7	6	0
49	3	6	8	0
50	9	3	9	0
51	13	0	23	3
52	18	9	9	7
53	17	0	9	0
54	21	10	22	0
55	20	10	0	0
56	18	14	4	3
57	36	11	10	0
58	36	12	17	3
59	22	3	7	14
60	24	0	9	4
61	24	0	3	4
62	23	7	7	0
63	23	3	3	3
64	16	0	11	7
65	14	0	9	4
66	15	3	4	3
67	21	0	10	0
68	3	0	9	0
69	0	0	3	6
70	8	3	3	0
71	4	3	4	0
74	0	3	0	0
76	0	0	3	0
77	3	0	0	0

```
[58]: import pandas as pd
import matplotlib.pyplot as plt

pd.crosstab(df.age, df.cp).plot(kind='bar', color=['coral', "lightskyblue", "plum", "khaki"])
plt.title('Type of Chest Pain for Age')
plt.xlabel('0 = Female, 1 = Male')
plt.show()
```



Type of Chest Pain for Age



35

