```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import plotly.express as px

# Step 1: Extract the Data
import pandas as pd
df = pd.read_csv('Heart Disease Data.csv')
df.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
```

```
# Step 2: Transform the Data
# Display basic info
print(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
```

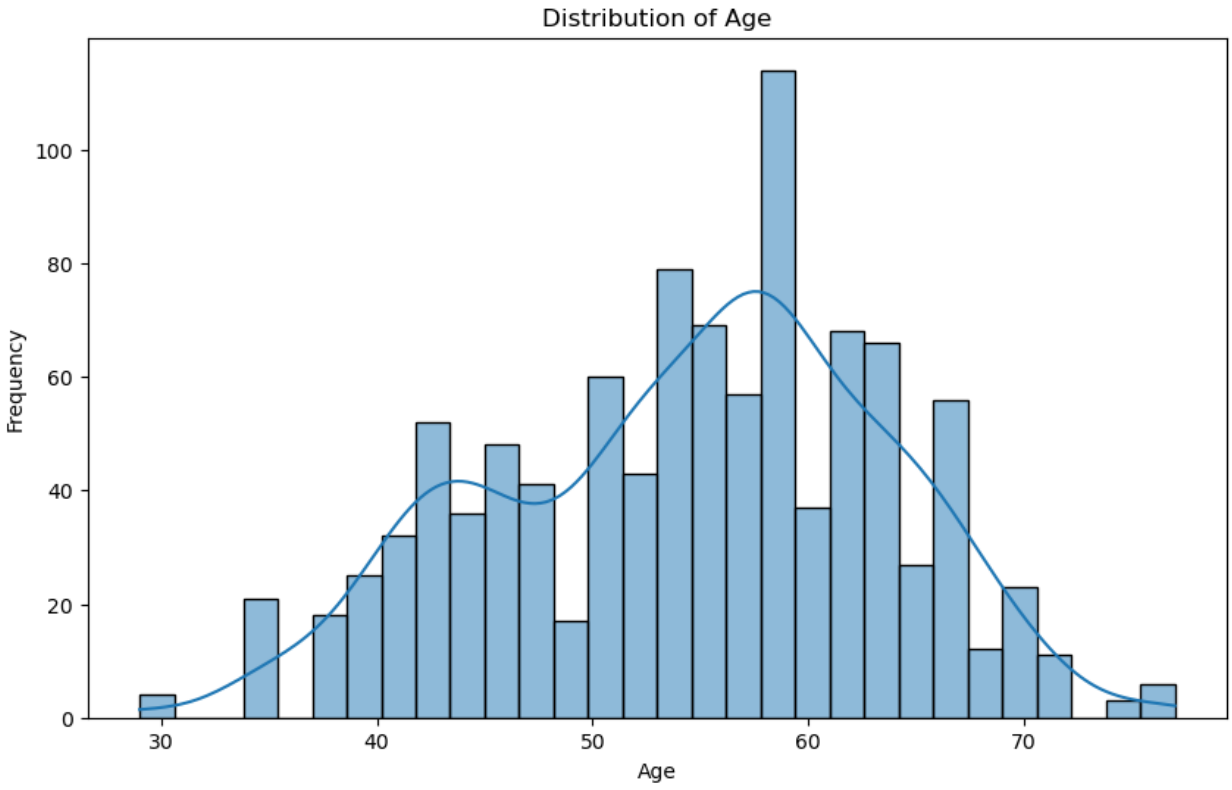
```
 11  ca        1025 non-null   int64
 12  thal      1025 non-null   int64
 13  target    1025 non-null   int64
dtypes: float64(1), int64(13)
memory usage: 112.2 KB
None
```

```
# Step 4:Handling missing values (if any)
df = df.dropna()

# Displaying summary statistics
print(df.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
```

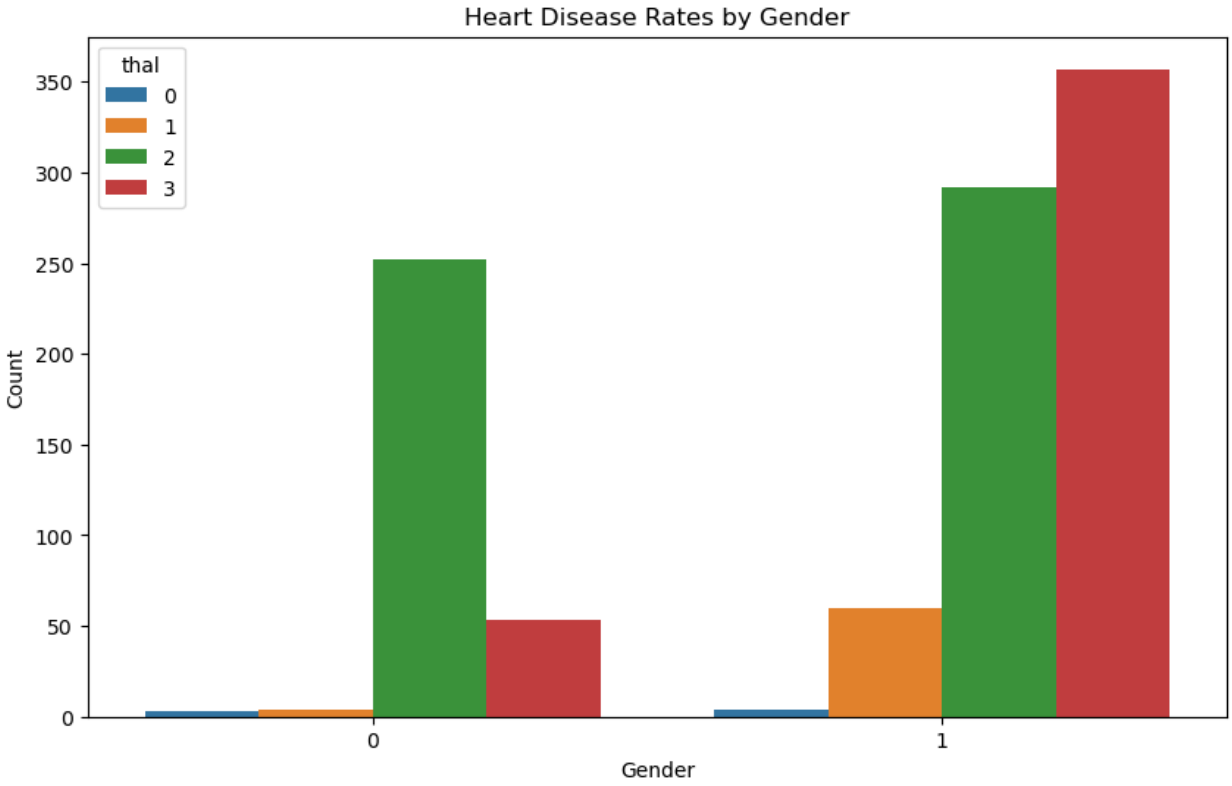
```
count  1025.000000  1025.000000  1025.000000  1025.000000
mean      1.385366     0.754146     2.323902     0.513171
std       0.617755     1.030798     0.620660     0.500070
min       0.000000     0.000000     0.000000     0.000000
25%       1.000000     0.000000     2.000000     0.000000
50%       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
```

```python
# Step 4: EDA

# Distribution of age
plt.figure(figsize=(10, 6))
sns.histplot(df['age'], bins=30, kde=True)
plt.title('Distribution of Age')
plt.xlabel('Age')
plt.ylabel('Frequency')
plt.show()
```

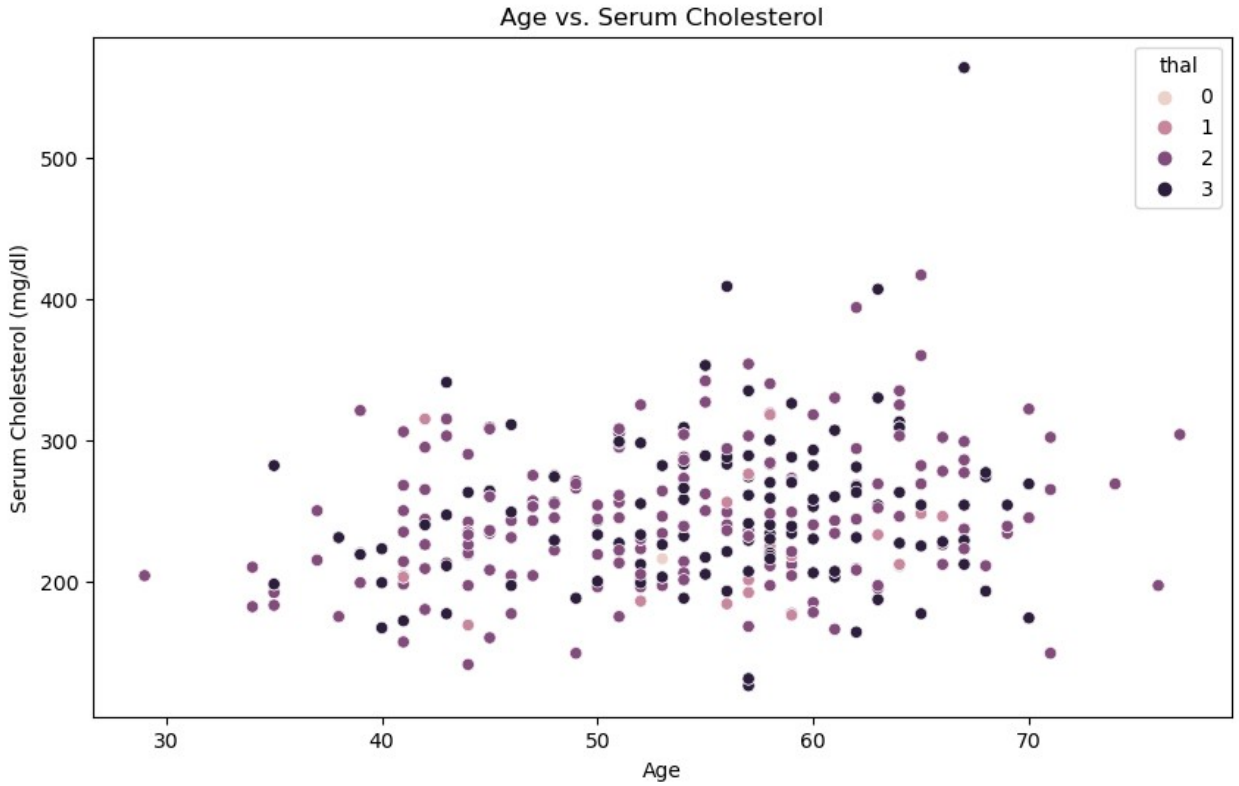



```python
# Heart disease rates by gender
plt.figure(figsize=(10, 6))
sns.countplot(data=df, x='sex', hue='thal')
plt.title('Heart Disease Rates by Gender')
plt.xlabel('Gender')
```

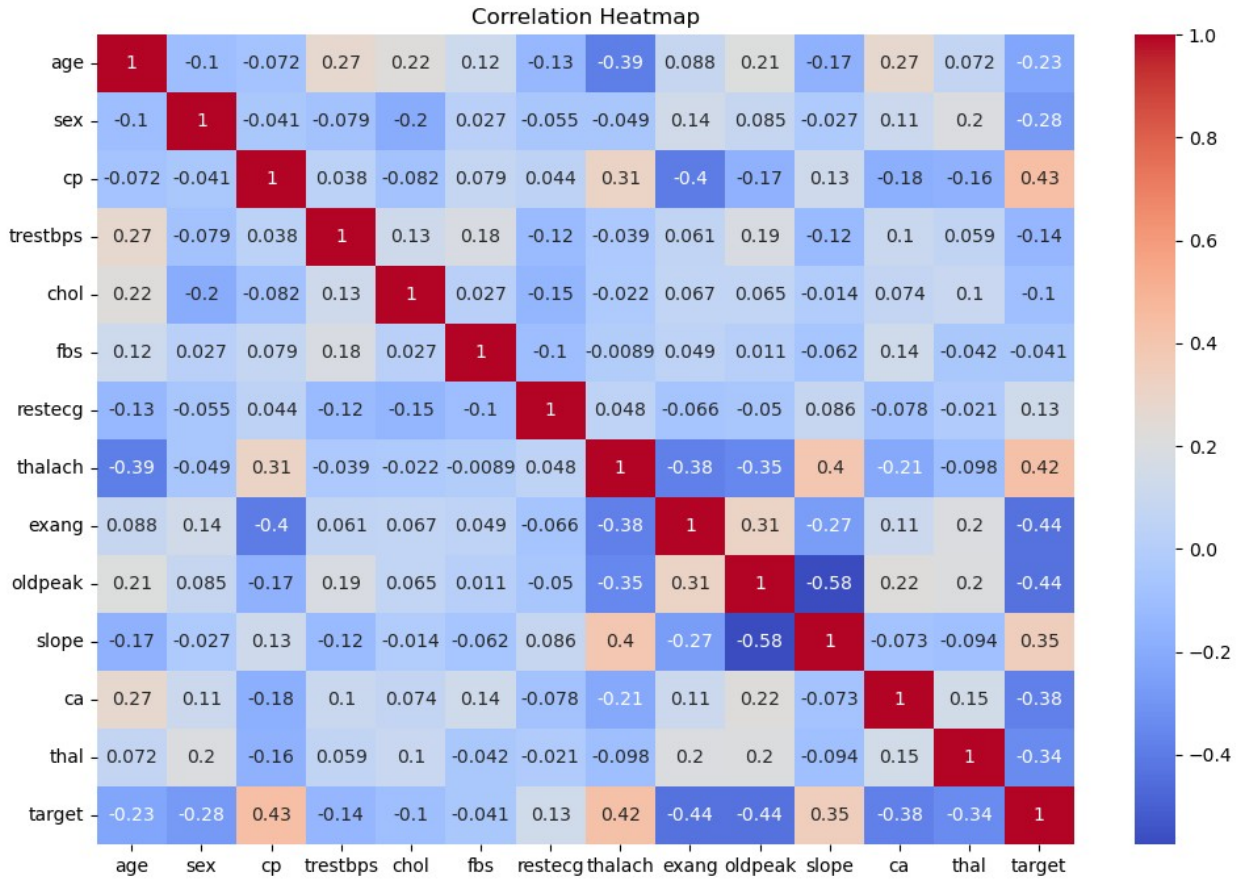
```
plt.ylabel('Count')
plt.show()
```




```
# Relationship between age and serum cholesterol
plt.figure(figsize=(10, 6))
sns.scatterplot(data=df, x='age', y='chol', hue='thal')
plt.title('Age vs. Serum Cholesterol')
plt.xlabel('Age')
plt.ylabel('Serum Cholesterol (mg/dl)')
plt.show()
```




```
# Correlation heatmap
plt.figure(figsize=(12, 8))
sns.heatmap(df.corr(), annot=True, cmap='coolwarm')
plt.title('Correlation Heatmap')
plt.show()
```




```
# Interactive visualization using Plotly
fig = px.scatter(df, x='age', y='thalach', color='thal', title='Age
vs. Maximum Heart Rate Achieved')
fig.show()
```


```
{"config":{"plotlyServerURL":"https://plot.ly"},"data":
[{"hovertemplate":"age=%{x}<br>thalach=%{y}<br>thal=%
{marker.color}<extra></extra>","legendgroup":"","marker":{"color":
[3,3,3,3,2,2,1,3,3,2,2,3,2,3,0,2,2,3,2,2,2,2,2,2,2,3,2,2,1,2,1,2,3,3,2
,2,2,2,3,3,2,3,2,3,2,2,1,3,2,3,2,3,2,3,3,3,3,2,3,2,2,2,2,2,2,3,2,1,2,3
,3,3,2,3,3,2,2,3,2,2,3,2,2,2,2,2,2,2,2,3,2,2,3,1,2,2,2,3,1,2,2,3,3,2,1
,2,3,3,3,2,2,3,3,3,1,3,3,3,2,2,3,1,3,2,3,2,2,2,2,1,2,2,1,2,3,1,2,2,2,2
,3,2,2,2,2,3,2,3,3,2,1,3,3,3,3,3,3,3,3,2,2,2,2,3,2,3,1,2,2,2,2,3,2,2,3
,3,3,2,2,3,3,2,3,2,2,3,3,2,3,3,2,3,3,2,2,2,3,2,2,3,2,2,3,3,2,2,1,2,2,3
,3,2,3,2,2,2,3,2,3,2,2,3,2,2,2,3,2,2,2,3,3,2,3,2,2,3,2,3,3,2,2,3,2,2,2
,2,3,3,2,2,2,2,3,3,3,2,2,2,3,2,2,3,2,2,1,2,3,2,3,2,3,3,2,2,1,3,3,2,3,2
,2,2,2,2,3,2,3,2,2,3,2,3,2,2,3,2,2,3,2,2,2,3,2,3,2,3,2,2,1,2,3,3,3,2,3
,2,3,2,3,0,2,2,3,1,2,2,3,1,2,0,2,2,2,2,3,3,3,2,1,3,2,2,2,3,2,1,3,2,3,3
,2,3,3,1,3,2,1,2,2,0,2,3,2,2,2,2,3,3,3,2,3,3,2,2,2,1,2,2,2,1,3,3,3,3,3
,2,2,2,2,1,2,3,2,3,3,2,3,3,1,1,3,2,2,2,3,1,2,3,2,2,2,2,1,2,3,2,2,2,2,2
,2,2,2,2,3,3,2,2,3,2,3,3,2,2,2,2,2,3,2,2,2,3,3,2,2,2,2,3,2,2,3,2,3,2,3
,2,2,2,2,2,3,2,1,3,3,2,2,2,2,2,2,3,2,2,3,3,3,3,2,3,3,3,3,2,1,3,3,3,2,2
,2,2,3,3,2,3,3,2,2,2,2,2,2,2,2,2,3,2,3,2,2,3,2,2,2,3,3,2,3,3,3,3,2,3,3
```

```
,2,3,2,3,2,3,2,2,2,2,2,2,2,3,3,3,2,2,2,2,3,2,2,2,3,3,2,3,2,3,3,3,2,2,2
,2,2,2,2,1,2,2,2,2,3,2,3,2,2,3,2,3,2,2,3,3,2,2,3,3,3,3,1,3,2,2,2,3,2,1
,3,2,2,2,2,2,3,3,2,2,2,2,2,2,1,1,2,3,3,1,3,3,2,2,2,2,1,3,3,2,2,3,3,3,3,2,2
,3,2,2,2,3,2,2,2,2,2,2,2,3,3,3,2,3,2,2,2,2,3,2,2,3,2,2,2,2,3,2,2,1,2,2,2
,2,2,3,2,1,2,3,3,3,3,3,3,3,2,3,2,3,1,3,2,3,0,3,3,2,3,2,3,2,3,3,2,3,2,3
,2,3,2,1,3,3,3,2,2,3,3,3,2,2,2,2,3,1,2,3,2,2,2,2,2,3,3,3,3,2,2,2,3,2,0
,2,3,3,3,3,2,2,3,3,2,2,2,3,2,3,2,2,2,3,2,2,2,1,2,2,1,2,3,2,3,3,3,3,3,2
,1,2,2,3,2,2,3,3,2,2,1,3,2,2,3,3,3,3,2,3,3,3,3,2,2,2,1,3,3,2,3,1,2,2,2
,3,1,2,3,2,3,3,2,3,2,3,2,2,2,1,3,2,3,2,2,2,2,3,2,3,2,3,3,3,1,2,2,3,2,2
,3,2,3,2,3,1,2,2,2,3,2,2,3,3,2,3,2,2,2,3,2,2,3,3,2,2,2,2,2,2,2,2,3,1,2
,3,3,2,3,2,3,2,3,3,3,3,3,1,3,3,3,2,3,0,3,3,3,2,2,3,2,2,2,1,2,2,2,2,3,3
,3,2,3,2,2,3,3,2,2,3,3,3,3,2,2,3,2,3,3,3,3,3,2,3,2,2,2,2,3,2,3,2,2,3,3
,3,2,2,3,3,2,3,2,2,2,3,2,3,2,2,2,2,3,3,2,2,3,2,3,2,2,3,1,2,3,3,2,2,2,3
,1,3,2,3,3,3,2,2,3,2,2,3,2,3,3,3,3,3,1,3,1,2,2,3,2,3,2,3,2,3,3,2,3,1,2
,3,2,3,3,2,2,3,2,2,3],"coloraxis":"coloraxis","symbol":"circle"},"mode
":"markers","name":"","showlegend":false,"type":"scattergl","x":
[52,53,70,61,62,58,58,55,46,54,71,43,34,51,52,34,51,54,50,58,60,67,45,
63,42,61,44,58,56,55,44,50,57,70,50,46,51,59,64,57,65,54,61,46,55,42,4
1,66,66,60,58,57,38,49,55,55,56,48,67,57,29,66,67,59,29,59,53,42,37,62
,59,61,56,59,48,47,48,63,52,52,50,49,46,38,37,44,58,59,62,68,54,62,63,
44,62,45,57,53,65,76,43,57,54,47,52,47,51,62,40,54,44,53,48,57,59,61,6
3,43,29,42,54,44,60,65,61,60,66,53,52,57,60,51,41,50,54,58,55,64,47,41
,57,63,61,34,47,40,51,41,52,39,58,54,58,54,63,54,40,54,67,41,77,51,77,
48,56,59,56,57,43,45,43,56,56,39,54,56,56,64,44,56,63,64,60,42,45,57,4
0,56,63,64,41,56,67,69,67,59,58,45,60,50,62,34,52,64,66,42,59,41,38,62
,42,67,50,43,45,49,65,41,46,54,57,63,64,39,51,54,49,44,52,66,58,56,60,
41,49,64,49,57,60,62,54,67,38,60,51,44,54,46,39,42,47,45,55,57,35,62,3
5,64,38,66,44,54,48,55,58,45,53,67,58,71,43,44,39,54,66,56,57,44,55,41
,45,35,41,64,58,71,64,71,58,40,52,58,61,59,56,67,67,58,35,52,46,51,55,
60,52,62,44,44,59,56,61,48,70,74,40,42,64,63,59,53,53,48,45,66,48,56,5
4,57,70,53,37,63,37,54,60,58,57,54,56,60,38,44,65,52,41,66,50,54,43,62
,66,50,57,57,57,46,59,67,59,53,48,62,43,53,57,53,58,48,58,51,43,55,54,
58,46,66,59,64,67,52,58,58,59,58,35,51,42,77,64,63,58,45,51,62,63,66,6
8,40,66,63,49,71,70,46,61,56,58,58,46,46,41,56,57,70,59,41,54,52,62,65
,57,42,57,58,45,51,54,57,43,47,67,65,60,37,41,57,59,67,47,58,62,60,57,
57,47,52,53,55,51,52,63,49,66,49,65,69,47,39,43,51,69,48,52,43,67,38,4
4,47,61,67,60,64,58,41,48,57,57,57,39,58,51,63,51,35,65,62,41,65,54,61
,57,57,64,55,51,59,68,57,54,46,71,35,46,45,47,44,61,41,56,55,54,43,44,
54,44,49,60,41,49,61,59,58,67,61,58,51,62,62,59,69,60,65,63,53,54,76,5
0,52,46,64,58,44,62,59,54,48,70,67,51,68,69,54,43,53,58,67,59,48,39,67
,58,65,42,44,56,53,50,41,54,42,54,60,34,44,60,43,52,70,62,58,46,44,55,
43,55,40,64,59,61,54,74,63,70,63,64,61,59,38,58,63,62,46,58,63,47,71,6
6,42,64,55,43,55,58,55,51,50,43,41,49,65,57,48,62,61,59,69,58,38,69,65
,45,49,45,61,52,53,59,58,65,58,46,56,64,65,44,50,47,64,71,45,62,41,66,
56,41,41,57,39,64,59,61,58,47,58,66,65,35,57,29,66,61,52,68,54,63,58,6
0,63,41,68,42,59,59,43,60,63,52,58,56,54,58,55,53,63,39,35,50,67,66,35
,41,35,71,57,51,58,57,56,60,45,56,35,45,66,51,70,55,56,55,52,64,45,67,
68,74,60,48,56,46,55,44,52,51,44,52,50,59,67,58,52,44,41,63,58,60,51,6
4,60,44,58,55,65,48,65,51,48,64,66,46,61,57,49,56,58,63,57,57,68,46,71
```

```
,41,45,62,65,48,51,61,53,59,34,44,58,64,56,54,55,67,51,62,62,54,54,68,
60,61,62,41,65,59,43,67,63,63,52,58,53,57,44,51,54,40,57,62,53,62,58,7
0,45,42,58,61,62,60,54,61,63,42,57,44,54,51,58,68,55,42,49,53,52,51,51
,70,35,58,59,60,56,35,61,58,52,58,37,52,67,65,46,68,43,68,51,52,64,59,
52,59,60,41,41,56,56,48,39,64,57,59,60,61,43,54,59,57,57,57,48,70,57,6
1,57,60,63,55,64,54,52,51,62,59,52,42,59,61,42,62,59,55,64,42,50,62,50
,50,58,35,45,66,46,65,47,59,38,39,58,44,54,61,57,54,42,54,60,65,40,51,
38,42,56,43,64,53,49,57,52,69,65,66,56,67,54,70,57,67,62,54,45,53,62,4
9,50,65,69,52,58,52,57,44,76,58,60,53,52,38,52,52,51,43,39,63,52,44,45
,57,39,67,64,59,62,55,57,60,71,56,60,50,43,59,44,56,54,42,67,64,42,66,
52,51,55,58,56,42,40,51,45,48,58,44,58,65,53,41,47,59,60,47,50,54],"xa
xis":"x","y":
[168,155,125,161,106,122,140,145,144,116,125,136,192,122,156,192,142,1
09,162,165,155,142,148,172,173,146,179,152,142,117,144,162,115,112,163
,147,125,182,105,150,151,147,169,144,166,178,132,165,152,160,162,123,1
73,139,111,111,162,180,150,164,202,152,125,157,202,159,152,125,170,145
,140,138,144,162,150,156,175,144,158,158,163,126,147,173,170,170,165,1
43,160,141,167,163,144,144,140,152,159,95,158,116,165,173,156,156,190,
118,173,103,181,108,177,111,150,143,134,146,147,120,202,162,156,144,14
2,157,140,171,151,152,169,148,171,149,132,163,156,146,166,154,156,153,
88,136,145,174,143,114,157,158,178,182,140,195,171,147,132,156,178,147
,160,182,162,143,162,166,144,140,105,163,162,138,171,103,169,179,126,1
33,103,96,177,150,144,144,170,150,170,124,114,144,147,131,179,163,160,
131,108,157,156,185,160,126,163,174,178,155,152,178,134,179,173,103,19
4,125,128,162,170,171,127,182,120,116,174,147,144,179,186,165,126,149,
184,165,171,178,141,163,163,155,126,141,160,154,159,150,173,155,166,18
8,195,120,152,150,118,138,132,124,130,157,182,154,182,151,169,165,186,
166,146,148,95,71,173,130,161,173,152,108,138,103,168,188,145,168,175,
174,172,144,173,130,133,162,172,181,169,130,137,157,133,125,71,111,174
,172,172,154,166,142,169,99,149,170,134,153,140,150,112,121,178,178,13
3,172,162,115,143,139,147,120,180,153,195,112,109,115,170,169,187,158,
132,111,173,152,142,144,173,169,148,162,182,120,128,167,143,97,151,126
,143,126,141,152,90,108,162,115,186,146,165,173,174,173,165,168,154,12
3,136,132,158,160,160,138,164,122,71,190,105,173,142,111,156,123,173,1
62,96,150,152,132,123,145,132,152,150,114,138,150,139,125,143,172,138,
142,165,156,152,147,168,144,112,109,159,172,152,169,163,151,159,122,16
3,152,147,142,163,174,143,152,150,114,160,187,172,163,143,129,179,160,
154,170,88,174,179,184,173,145,157,160,154,162,165,162,114,131,152,152
,171,186,146,175,190,161,160,173,188,118,137,108,171,133,131,168,168,1
73,124,150,182,131,142,154,122,174,158,103,158,127,159,137,159,174,131
,132,166,161,150,164,160,160,125,182,160,175,152,149,140,172,133,117,1
59,120,153,108,170,139,132,168,139,161,162,144,172,161,165,149,145,163
,145,131,157,148,136,160,167,116,158,172,144,155,130,180,157,162,152,1
68,109,142,157,151,146,116,136,152,156,163,162,186,179,71,131,148,173,
153,105,160,159,168,163,194,170,142,174,153,141,171,178,109,157,105,15
6,180,155,120,132,178,131,90,146,116,121,169,112,169,132,161,164,173,1
60,179,97,120,172,179,179,130,132,125,132,117,136,130,146,111,142,158,
181,163,171,158,88,150,106,145,145,146,173,182,151,174,132,171,175,125
,168,143,143,172,174,131,152,150,105,140,169,128,152,122,125,175,99,18
```

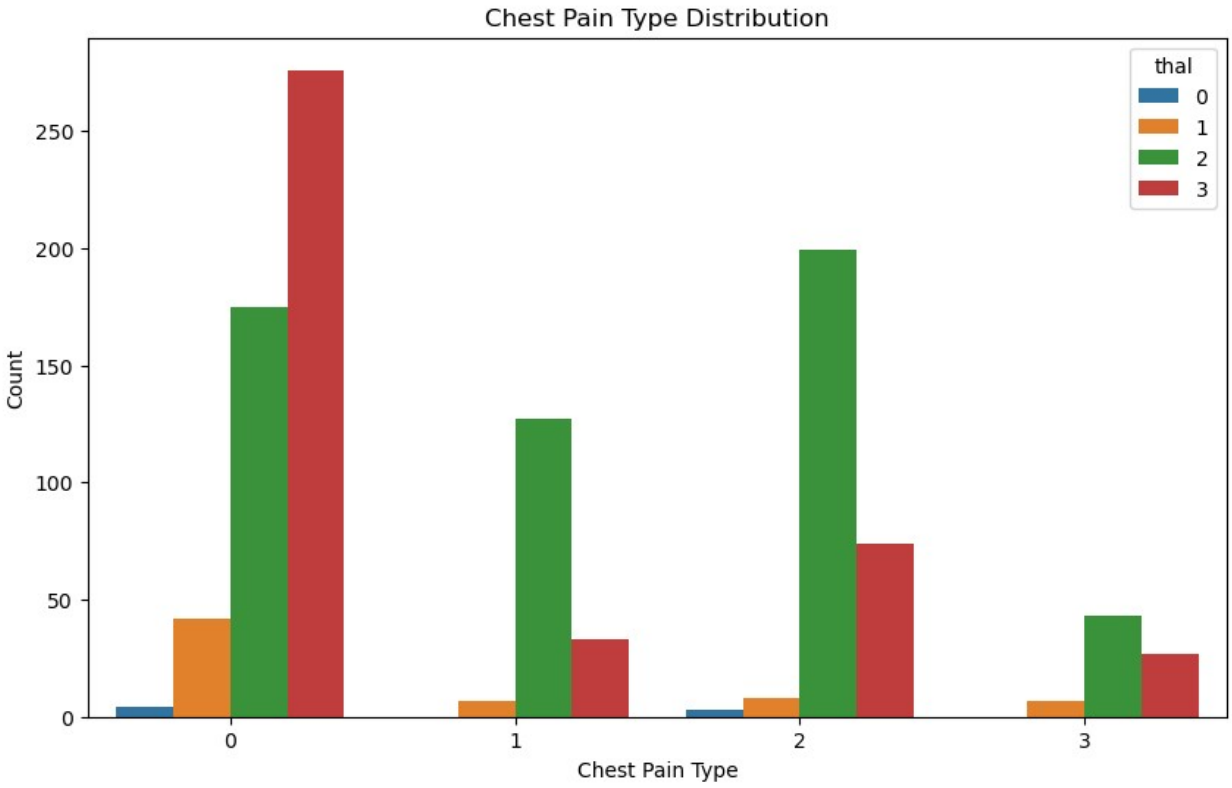
```
2,114,103,163,153,174,152,131,182,125,140,143,122,114,157,174,141,202,
120,169,168,151,147,132,156,144,147,172,150,162,140,90,120,155,154,156
,171,133,170,154,161,155,179,140,156,158,150,132,156,168,130,162,126,1
54,131,150,153,96,132,163,130,138,114,125,143,130,142,161,147,158,185,
172,115,121,157,166,163,120,161,170,160,154,175,156,163,142,129,171,16
1,180,172,132,144,160,143,158,160,169,144,155,151,180,140,149,175,96,1
38,172,169,112,163,163,162,144,143,143,141,156,130,132,152,154,114,139
,143,146,95,162,192,144,105,122,169,165,145,163,122,106,103,126,126,14
1,96,145,140,132,114,159,181,129,150,136,160,131,111,112,179,154,158,1
14,141,160,155,140,154,143,185,178,146,138,160,144,160,125,172,150,150
,179,109,143,144,151,111,125,126,160,162,125,186,125,182,165,125,170,1
05,182,125,162,161,160,187,161,129,174,156,115,181,115,173,160,158,159
,168,125,171,179,153,169,178,139,182,155,148,125,132,138,165,113,182,1
68,164,115,166,125,124,140,126,157,154,130,154,126,156,173,97,145,158,
122,161,145,122,160,134,155,96,178,162,99,126,163,152,130,148,132,144,
127,179,162,182,140,105,173,160,169,115,108,194,109,132,127,181,142,18
2,150,169,162,158,111,162,123,169,151,140,165,162,172,163,125,168,108,
97,170,152,143,146,163,163,157,151,169,122,147,123,175,116,130,96,155,
172,173,162,190,166,161,140,169,184,177,147,148,140,142,105,161,146,11
7,164,170,162,178,160,159,143,142,173,150,113,125,163,132,178,132,147,
142,130,165,162,162,181,173,170,168,140,175,131,174,95,158,143,164,141
,118,159,113],"yaxis":"y"}],"layout":{"coloraxis":{"colorbar":
{"title":{"text":"thal"}},"colorscale":[[0,"#0d0887"],
[0.1111111111111111,"#46039f"],[0.2222222222222222,"#7201a8"],
[0.3333333333333333,"#9c179e"],[0.4444444444444444,"#bd3786"],
[0.5555555555555556,"#d8576b"],[0.6666666666666666,"#ed7953"],
[0.7777777777777778,"#fb9f3a"],[0.8888888888888888,"#fdca26"],
[1,"#f0f921"]]},"legend":{"tracegroupgap":0},"template":{"data":
{"bar":[{"error_x":{"color":"#2a3f5f"},"error_y":
{"color":"#2a3f5f"},"marker":{"line":
{"color":"#E5ECF6","width":0.5},"pattern":
{"fillmode":"overlay","size":10,"solidity":0.2}},"type":"bar"}],"barpo
lar":[{"marker":{"line":{"color":"#E5ECF6","width":0.5},"pattern":
{"fillmode":"overlay","size":10,"solidity":0.2}},"type":"barpolar"}],"
carpet":[{"aaxis":
{"endlinecolor":"#2a3f5f","gridcolor":"white","linecolor":"white","min
orgridcolor":"white","startlinecolor":"#2a3f5f"},"baxis":
{"endlinecolor":"#2a3f5f","gridcolor":"white","linecolor":"white","min
orgridcolor":"white","startlinecolor":"#2a3f5f"},"type":"carpet"}],"ch
oropleth":[{"colorbar":
{"outlinewidth":0,"ticks":""},"type":"choropleth"}],"contour":
[{"colorbar":{"outlinewidth":0,"ticks":""},"colorscale":
[[0,"#0d0887"],[0.1111111111111111,"#46039f"],
[0.2222222222222222,"#7201a8"],[0.3333333333333333,"#9c179e"],
[0.4444444444444444,"#bd3786"],[0.5555555555555556,"#d8576b"],
[0.6666666666666666,"#ed7953"],[0.7777777777777778,"#fb9f3a"],
[0.8888888888888888,"#fdca26"],
[1,"#f0f921"]],"type":"contour"}],"contourcarpet":[{"colorbar":
{"outlinewidth":0,"ticks":""},"type":"contourcarpet"}],"heatmap":
```

```
[{"colorbar":{"outlinewidth":0,"ticks":""},"colorscale":
[[0,"#0d0887"],[0.1111111111111111,"#46039f"],
[0.2222222222222222,"#7201a8"],[0.3333333333333333,"#9c179e"],
[0.4444444444444444,"#bd3786"],[0.5555555555555556,"#d8576b"],
[0.6666666666666666,"#ed7953"],[0.7777777777777778,"#fb9f3a"],
[0.8888888888888888,"#fdca26"],
[1,"#f0f921"]],"type":"heatmap"}],"heatmapgl":[{"colorbar":
{"outlinewidth":0,"ticks":""},"colorscale":[[0,"#0d0887"],
[0.1111111111111111,"#46039f"],[0.2222222222222222,"#7201a8"],
[0.3333333333333333,"#9c179e"],[0.4444444444444444,"#bd3786"],
[0.5555555555555556,"#d8576b"],[0.6666666666666666,"#ed7953"],
[0.7777777777777778,"#fb9f3a"],[0.8888888888888888,"#fdca26"],
[1,"#f0f921"]],"type":"heatmapgl"}],"histogram":[{"marker":{"pattern":
{"fillmode":"overlay","size":10,"solidity":0.2}},"type":"histogram"}],
"histogram2d":[{"colorbar":{"outlinewidth":0,"ticks":""},"colorscale":
[[0,"#0d0887"],[0.1111111111111111,"#46039f"],
[0.2222222222222222,"#7201a8"],[0.3333333333333333,"#9c179e"],
[0.4444444444444444,"#bd3786"],[0.5555555555555556,"#d8576b"],
[0.6666666666666666,"#ed7953"],[0.7777777777777778,"#fb9f3a"],
[0.8888888888888888,"#fdca26"],
[1,"#f0f921"]],"type":"histogram2d"}],"histogram2dcontour":
[{"colorbar":{"outlinewidth":0,"ticks":""},"colorscale":
[[0,"#0d0887"],[0.1111111111111111,"#46039f"],
[0.2222222222222222,"#7201a8"],[0.3333333333333333,"#9c179e"],
[0.4444444444444444,"#bd3786"],[0.5555555555555556,"#d8576b"],
[0.6666666666666666,"#ed7953"],[0.7777777777777778,"#fb9f3a"],
[0.8888888888888888,"#fdca26"],
[1,"#f0f921"]],"type":"histogram2dcontour"}],"mesh3d":[{"colorbar":
{"outlinewidth":0,"ticks":""},"type":"mesh3d"}],"parcoords":[{"line":
{"colorbar":{"outlinewidth":0,"ticks":""}},"type":"parcoords"}],"pie":
[{"automargin":true,"type":"pie"}],"scatter":[{"fillpattern":
{"fillmode":"overlay","size":10,"solidity":0.2},"type":"scatter"}],"sc
atter3d":[{"line":{"colorbar":{"outlinewidth":0,"ticks":""}},"marker":
{"colorbar":
{"outlinewidth":0,"ticks":""}},"type":"scatter3d"}],"scattercarpet":
[{"marker":{"colorbar":
{"outlinewidth":0,"ticks":""}},"type":"scattercarpet"}],"scattergeo":
[{"marker":{"colorbar":
{"outlinewidth":0,"ticks":""}},"type":"scattergeo"}],"scattergl":
[{"marker":{"colorbar":
{"outlinewidth":0,"ticks":""}},"type":"scattergl"}],"scattermapbox":
[{"marker":{"colorbar":
{"outlinewidth":0,"ticks":""}},"type":"scattermapbox"}],"scatterpolar"
:[{"marker":{"colorbar":
{"outlinewidth":0,"ticks":""}},"type":"scatterpolar"}],"scatterpolargl
":[{"marker":{"colorbar":
{"outlinewidth":0,"ticks":""}},"type":"scatterpolargl"}],"scatterterna
ry":[{"marker":{"colorbar":
{"outlinewidth":0,"ticks":""}},"type":"scatterternary"}],"surface":
```

```
[{"colorbar":{"outlinewidth":0,"ticks":""},"colorscale":
[[0,"#0d0887"],[0.1111111111111111,"#46039f"],
[0.2222222222222222,"#7201a8"],[0.3333333333333333,"#9c179e"],
[0.4444444444444444,"#bd3786"],[0.5555555555555556,"#d8576b"],
[0.6666666666666666,"#ed7953"],[0.7777777777777778,"#fb9f3a"],
[0.8888888888888888,"#fdca26"],
[1,"#f0f921"]],"type":"surface"}],"table":[{"cells":{"fill":
{"color":"#EBF0F8"},"line":{"color":"white"}},"header":{"fill":
{"color":"#C8D4E3"},"line":
{"color":"white"}},"type":"table"}]},"layout":{"annotationdefaults":
{"arrowcolor":"#2a3f5f","arrowhead":0,"arrowwidth":1},"autotypenumbers
":"strict","coloraxis":{"colorbar":
{"outlinewidth":0,"ticks":""}},"colorscale":{"diverging":
[[0,"#8e0152"],[0.1,"#c51b7d"],[0.2,"#de77ae"],[0.3,"#f1b6da"],
[0.4,"#fde0ef"],[0.5,"#f7f7f7"],[0.6,"#e6f5d0"],[0.7,"#b8e186"],
[0.8,"#7fbc41"],[0.9,"#4d9221"],[1,"#276419"]],"sequential":
[[0,"#0d0887"],[0.1111111111111111,"#46039f"],
[0.2222222222222222,"#7201a8"],[0.3333333333333333,"#9c179e"],
[0.4444444444444444,"#bd3786"],[0.5555555555555556,"#d8576b"],
[0.6666666666666666,"#ed7953"],[0.7777777777777778,"#fb9f3a"],
[0.8888888888888888,"#fdca26"],[1,"#f0f921"]],"sequentialminus":
[[0,"#0d0887"],[0.1111111111111111,"#46039f"],
[0.2222222222222222,"#7201a8"],[0.3333333333333333,"#9c179e"],
[0.4444444444444444,"#bd3786"],[0.5555555555555556,"#d8576b"],
[0.6666666666666666,"#ed7953"],[0.7777777777777778,"#fb9f3a"],
[0.8888888888888888,"#fdca26"],[1,"#f0f921"]]},"colorway":
["#636efa","#EF553B","#00cc96","#ab63fa","#FFA15A","#19d3f3","#FF6692"
,"#B6E880","#FF97FF","#FECB52"],"font":{"color":"#2a3f5f"},"geo":
{"bgcolor":"white","lakecolor":"white","landcolor":"#E5ECF6","showlake
s":true,"showland":true,"subunitcolor":"white"},"hoverlabel":
{"align":"left"},"hovermode":"closest","mapbox":
{"style":"light"},"paper_bgcolor":"white","plot_bgcolor":"#E5ECF6","po
lar":{"angularaxis":
{"gridcolor":"white","linecolor":"white","ticks":""},"bgcolor":"#E5ECF
6","radialaxis":
{"gridcolor":"white","linecolor":"white","ticks":""}},"scene":
{"xaxis":
{"backgroundcolor":"#E5ECF6","gridcolor":"white","gridwidth":2,"lineco
lor":"white","showbackground":true,"ticks":"","zerolinecolor":"white"}
,"yaxis":
{"backgroundcolor":"#E5ECF6","gridcolor":"white","gridwidth":2,"lineco
lor":"white","showbackground":true,"ticks":"","zerolinecolor":"white"}
,"zaxis":
{"backgroundcolor":"#E5ECF6","gridcolor":"white","gridwidth":2,"lineco
lor":"white","showbackground":true,"ticks":"","zerolinecolor":"white"}
},"shapedefaults":{"line":{"color":"#2a3f5f"}},"ternary":{"aaxis":
{"gridcolor":"white","linecolor":"white","ticks":""},"baxis":
{"gridcolor":"white","linecolor":"white","ticks":""},"bgcolor":"#E5ECF
6","caxis":
```

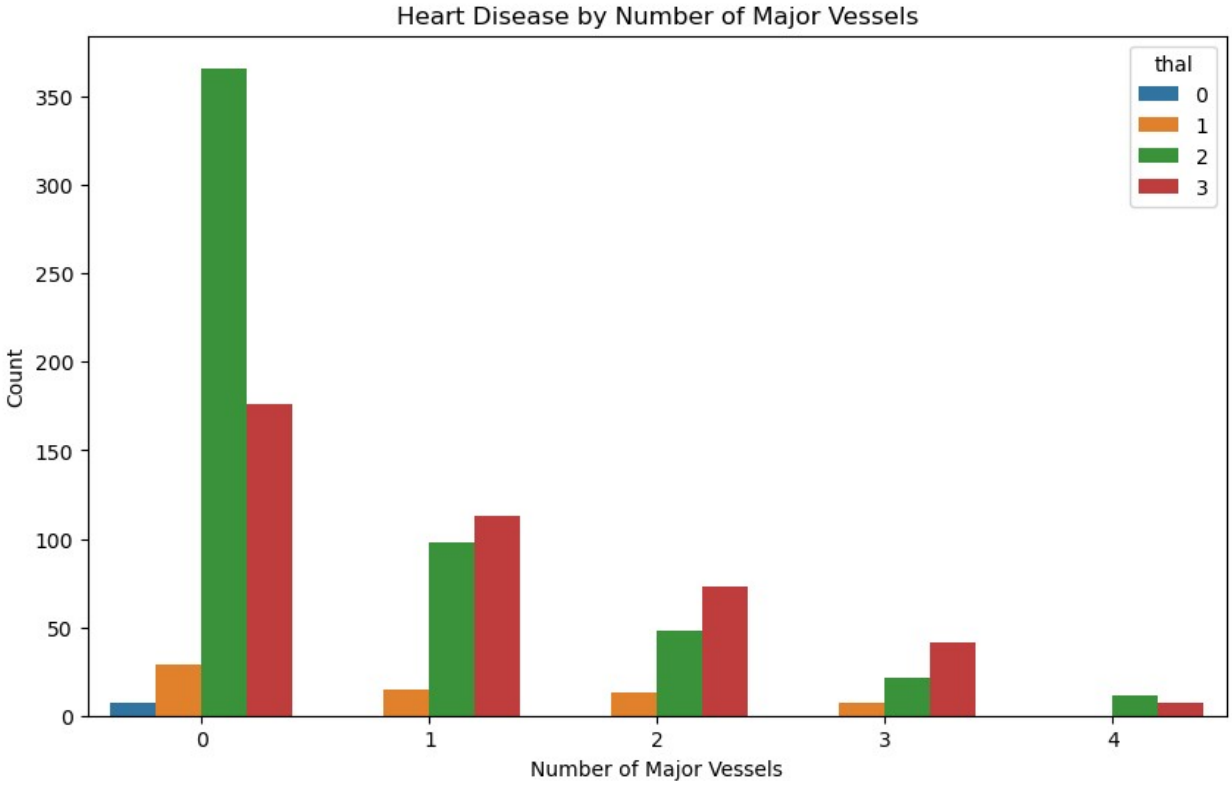
{"gridcolor":"white","linecolor":"white","ticks":""}},"title":
{"x":5.0e-2},"xaxis":
{"automargin":true,"gridcolor":"white","linecolor":"white","ticks":"",
"title":
{"standoff":15},"zerolinecolor":"white","zerolinewidth":2},"yaxis":
{"automargin":true,"gridcolor":"white","linecolor":"white","ticks":"",
"title":
{"standoff":15},"zerolinecolor":"white","zerolinewidth":2}}},"title":
{"text":"Age vs. Maximum Heart Rate Achieved"},"xaxis":
{"anchor":"y","domain":[0,1],"title":{"text":"age"}},"yaxis":
{"anchor":"x","domain":[0,1],"title":{"text":"thalach"}}}}

```
# More visualizations as needed for key metrics and relationships
# Distribution of chest pain types
plt.figure(figsize=(10, 6))
sns.countplot(data=df, x='cp', hue='thal')
plt.title('Chest Pain Type Distribution')
plt.xlabel('Chest Pain Type')
plt.ylabel('Count')
plt.show()
```

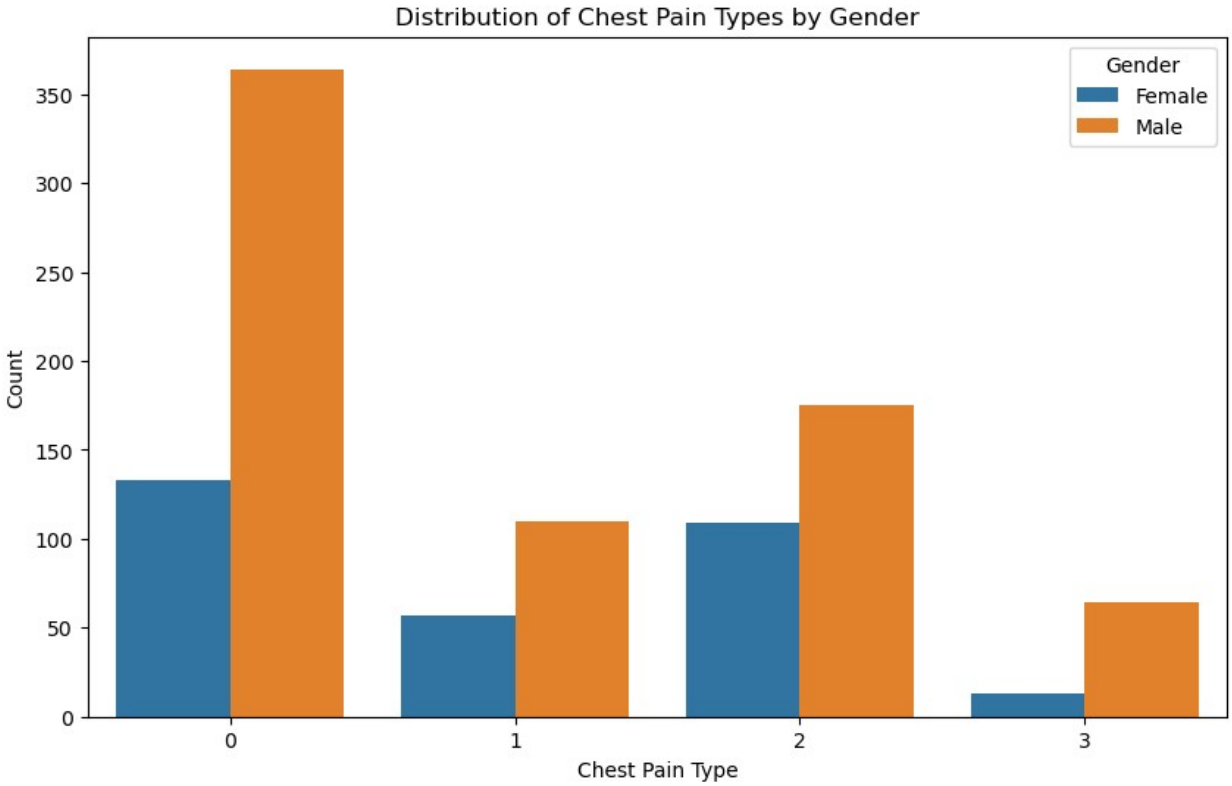



```
# Heart disease by number of major vessels
plt.figure(figsize=(10, 6))
sns.countplot(data=df, x='ca', hue='thal')
plt.title('Heart Disease by Number of Major Vessels')
```

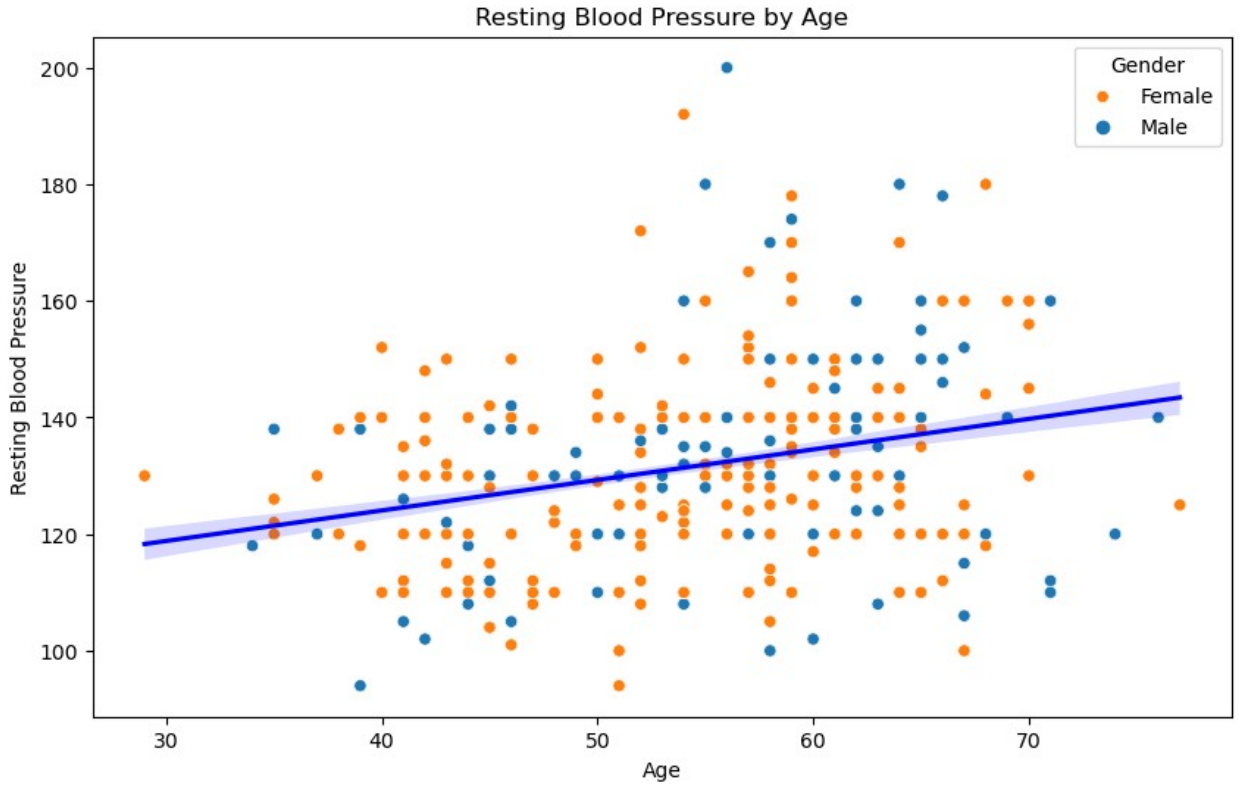
```
plt.xlabel('Number of Major Vessels')
plt.ylabel('Count')
plt.show()
```



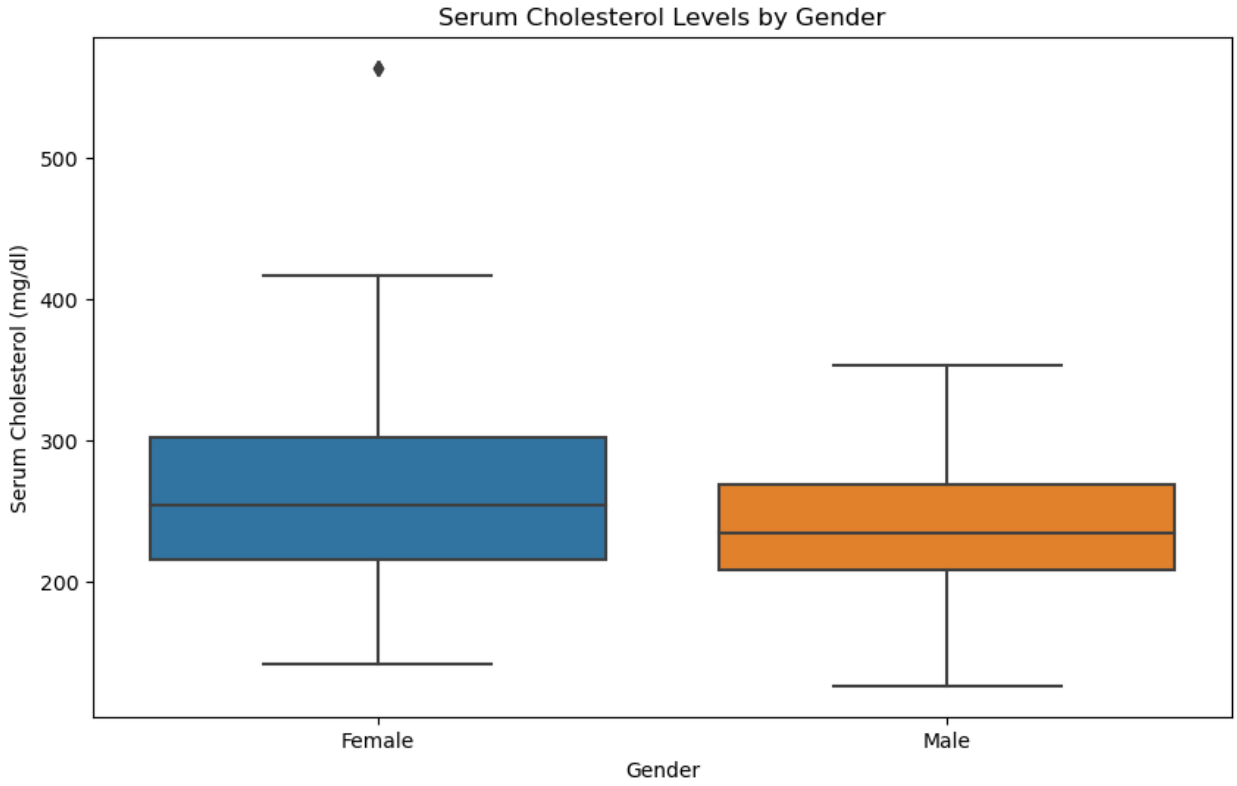

```
#Distribution of Chest Pain Types by Gender
plt.figure(figsize=(10, 6))
sns.countplot(data=df, x='cp', hue='sex')
plt.title('Distribution of Chest Pain Types by Gender')
plt.xlabel('Chest Pain Type')
plt.ylabel('Count')
plt.legend(title='Gender', labels=['Female', 'Male'])
plt.show()
```



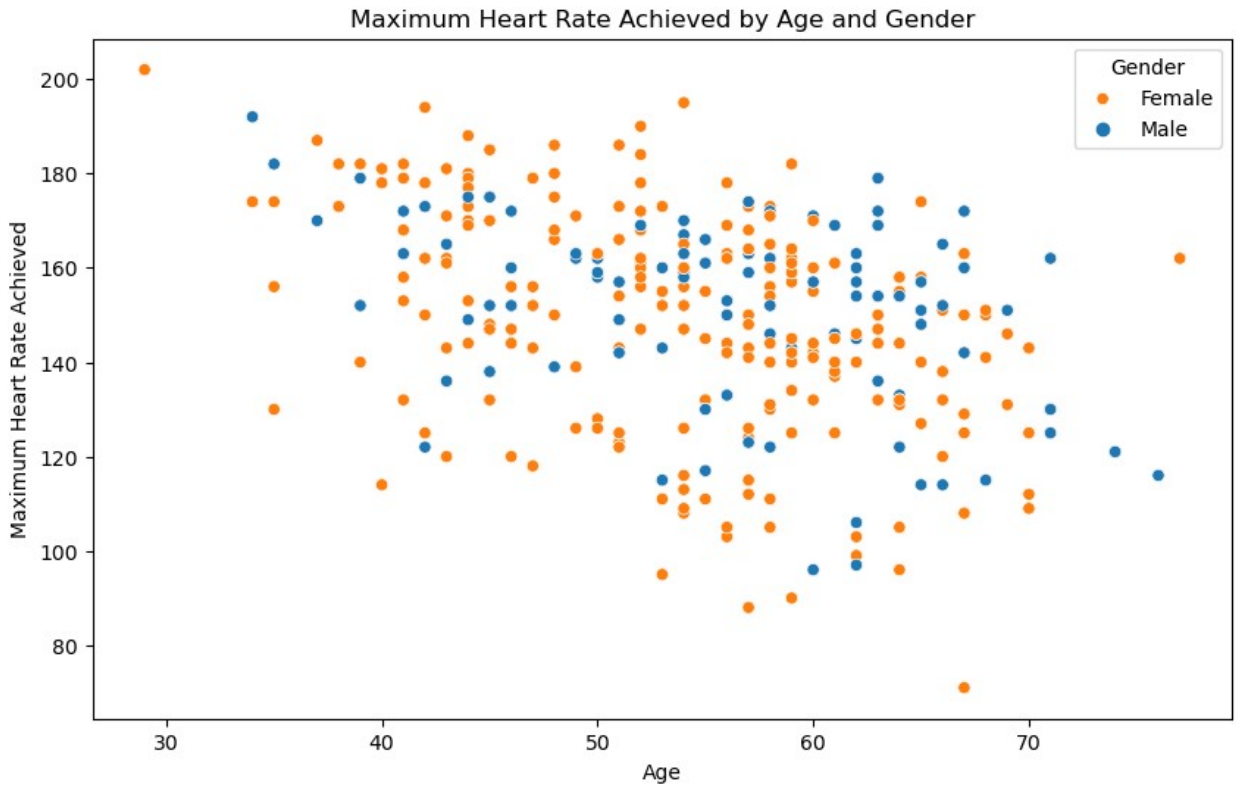

```
#Resting Blood Pressure by Age
plt.figure(figsize=(10, 6))
sns.scatterplot(data=df, x='age', y='trestbps', hue='sex')
sns.regplot(data=df, x='age', y='trestbps', scatter=False,
color='blue')
plt.title('Resting Blood Pressure by Age')
plt.xlabel('Age')
plt.ylabel('Resting Blood Pressure')
plt.legend(title='Gender', labels=['Female', 'Male'])
plt.show()
```




```
#Serum Cholesterol Levels by Gender
plt.figure(figsize=(10, 6))
sns.boxplot(data=df, x='sex', y='chol')
plt.title('Serum Cholesterol Levels by Gender')
plt.xlabel('Gender')
plt.ylabel('Serum Cholesterol (mg/dl)')
plt.xticks([0, 1], ['Female', 'Male'])
plt.show()
```



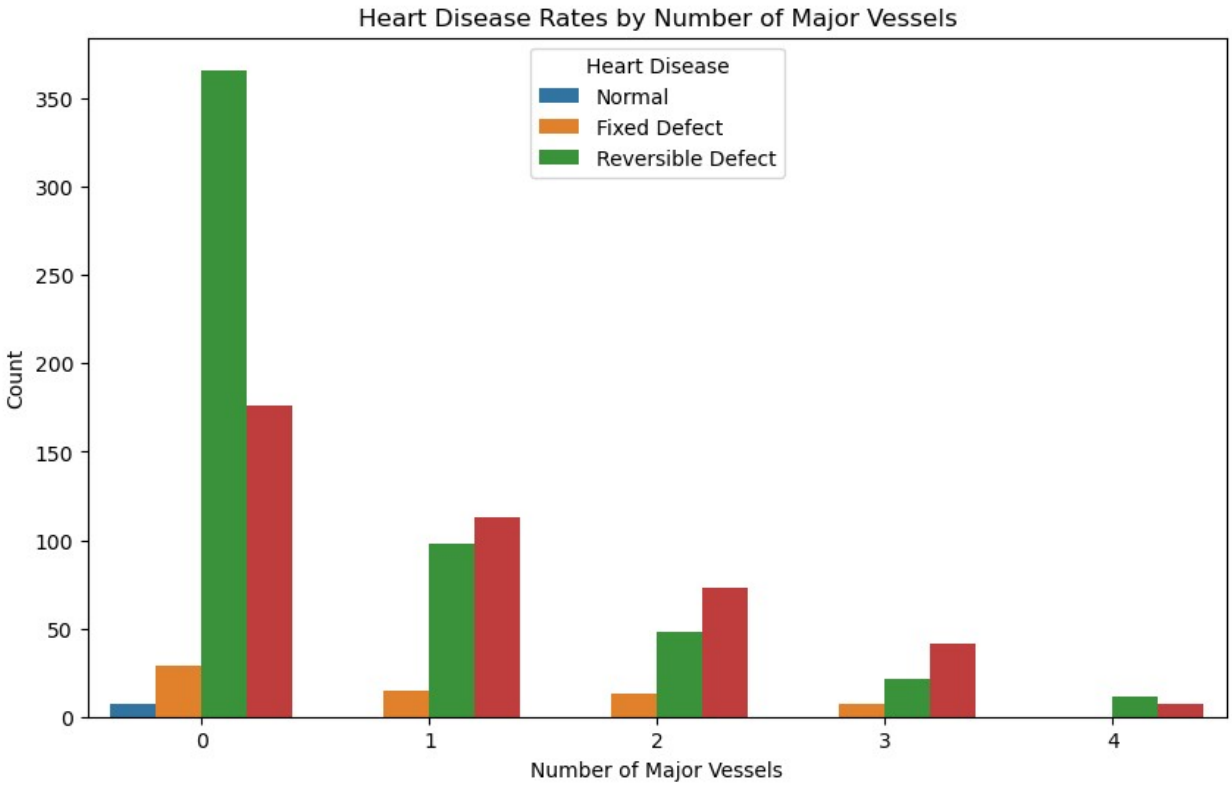

```
#Maximum Heart Rate Achieved by Age and Gender
plt.figure(figsize=(10, 6))
sns.scatterplot(data=df, x='age', y='thalach', hue='sex')
plt.title('Maximum Heart Rate Achieved by Age and Gender')
plt.xlabel('Age')
plt.ylabel('Maximum Heart Rate Achieved')
plt.legend(title='Gender', labels=['Female', 'Male'])
plt.show()
```



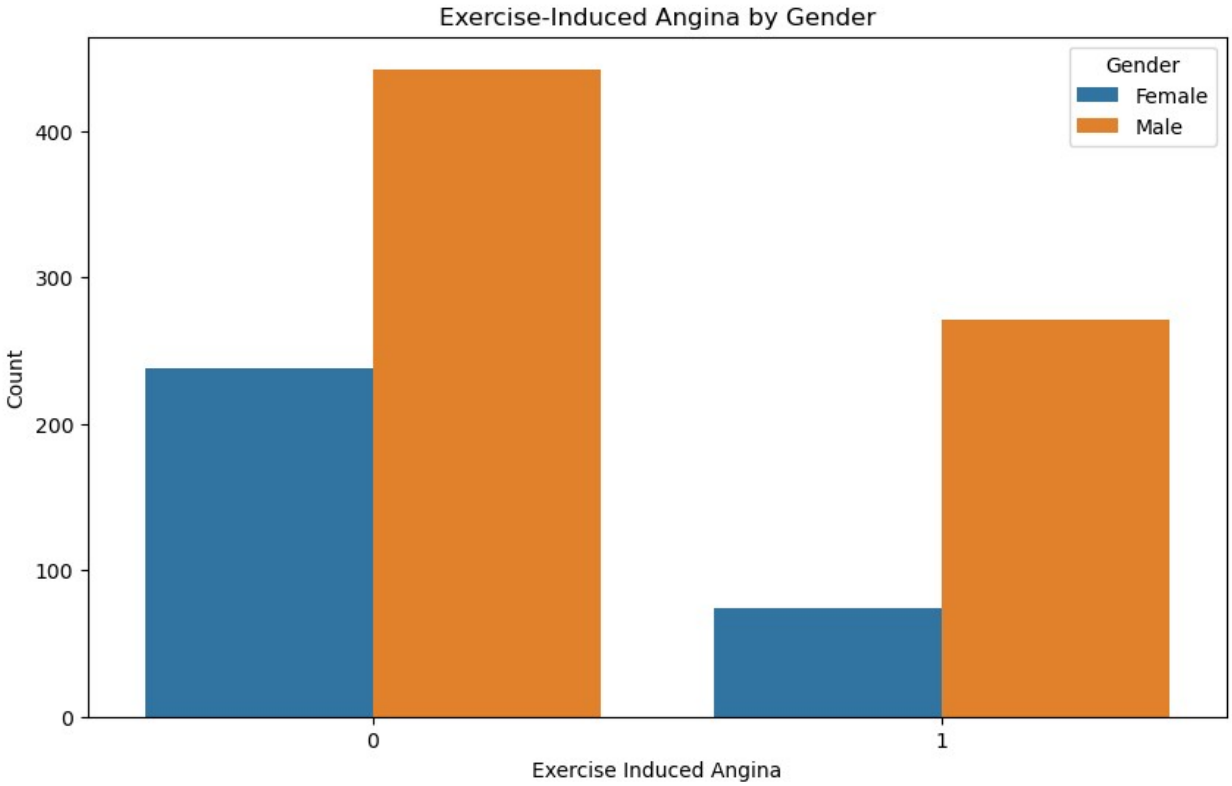

```
print(df.info())
```

```
<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
None
```

```
#Heart Disease Rates by Number of Major Vessels
plt.figure(figsize=(10, 6))
sns.countplot(data=df, x='ca', hue='thal')
plt.title('Heart Disease Rates by Number of Major Vessels')
plt.xlabel('Number of Major Vessels')
plt.ylabel('Count')
plt.legend(title='Heart Disease', labels=['Normal', 'Fixed Defect',
'Reversible Defect'])
plt.show()
```




```
#Exercise-Induced Angina by Gender
plt.figure(figsize=(10, 6))
sns.countplot(data=df, x='exang', hue='sex')
plt.title('Exercise-Induced Angina by Gender')
plt.xlabel('Exercise Induced Angina')
plt.ylabel('Count')
plt.legend(title='Gender', labels=['Female', 'Male'])
plt.show()
```


Exercise-Induced Angina by Gender
Gender
Female
Male
Count
400
300
200
100
0
0
1
Exercise Induced Angina