In [36]:
```
#Importing Libraries
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import seaborn as sns
sns.set_style('whitegrid')
import warnings
warnings.filterwarnings('ignore')
```

In [18]:
```
#Extracting CSV Dataset From System using Pandas Library

data=pd.read_csv(r"C:\Users\Prajakta Bose\Downloads\Heart Disease data\Hear
data
```

Out[18]:

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

1025 rows × 14 columns

In [19]:
```
#All Columns in the Dataset

data.columns
```

Out[19]:
```
Index(['age', 'sex', 'cp', 'trestbps', 'chol', 'fbs', 'restecg', 'thalac
h',
       'exang', 'oldpeak', 'slope', 'ca', 'thal', 'target'],
      dtype='object')
```

There are thirteen features in Dataset age: The person's age in years

sex: The person's sex (1 = male, 0 = female)

cp: The chest pain experienced (Value 1: typical angina, Value 2: atypical angina, Value 3: non-anginal pain, Value 4: asymptomatic)

trestbps: The person's resting blood pressure (mm Hg on admission to the hospital)

chol: The person's cholesterol measurement in mg/dl

fbs: The person's fasting blood sugar (> 120 mg/dl, 1 = true; 0 = false) restecg: Resting electrocardiographic measurement (0 = normal, 1 = having ST-T wave abnormality, 2 = showing probable or definite left ventricular hypertrophy by Estes' criteria)

thalach: The person's maximum heart rate achieved

exang: Exercise induced angina (1 = yes; 0 = no)

oldpeak: ST depression induced by exercise relative to rest

slope: the slope of the peak exercise ST segment (Value 1: upsloping, Value 2: flat, Value 3: downsloping)

ca: The number of major vessels (0-3)

thal: A blood disorder called thalassemia (3 = normal; 6 = fixed defect; 7 = reversable defect)

target: Heart disease (0 = no, 1 = yes)

In [20]:

```
#Checking NULL Values

data.isnull().sum()
```

Out[20]:

```
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
```

There is no missing values in our dataset

Percentage of people having heart diseases

In [22]:

```
target=data.groupby('target').size()
target
```

Out[22]:

```
target
0    499
1    526
dtype: int64
```

In [50]:

```
 #Converting Numerical Data into Categorical Data

def heart_disease(row):
    if row==0:
        return 'Absent'
    elif row==1:
        return 'Present'
```

In [51]:

```
#Applying converted data into our dataset with new column - Heart_Disease

data['Heart_Disease']=data['target'].apply(heart_disease)
data.head()
```

Out[51]:

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

In [25]:

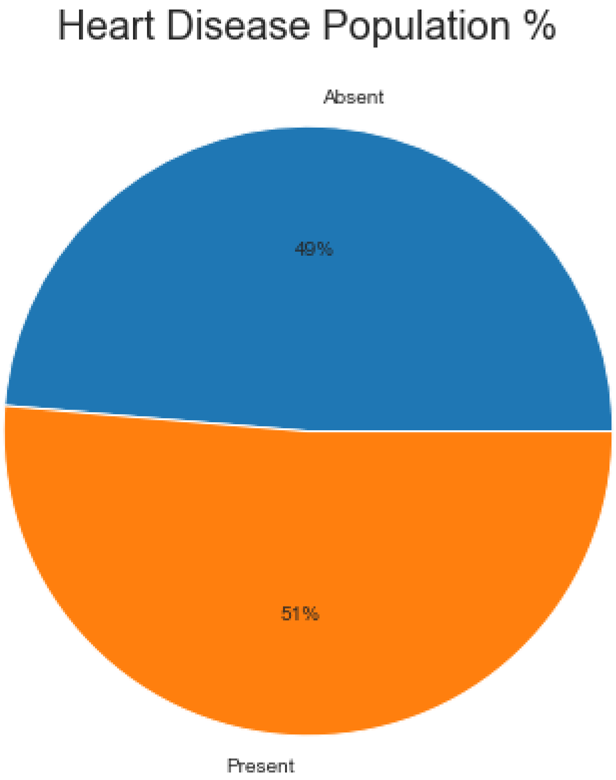
```
hd=data.groupby('Heart_Disease')['target'].count()
hd
```

Out[25]:

```
Heart_Disease
Absence     499
Presence    526
Name: target, dtype: int64
```

In [52]:

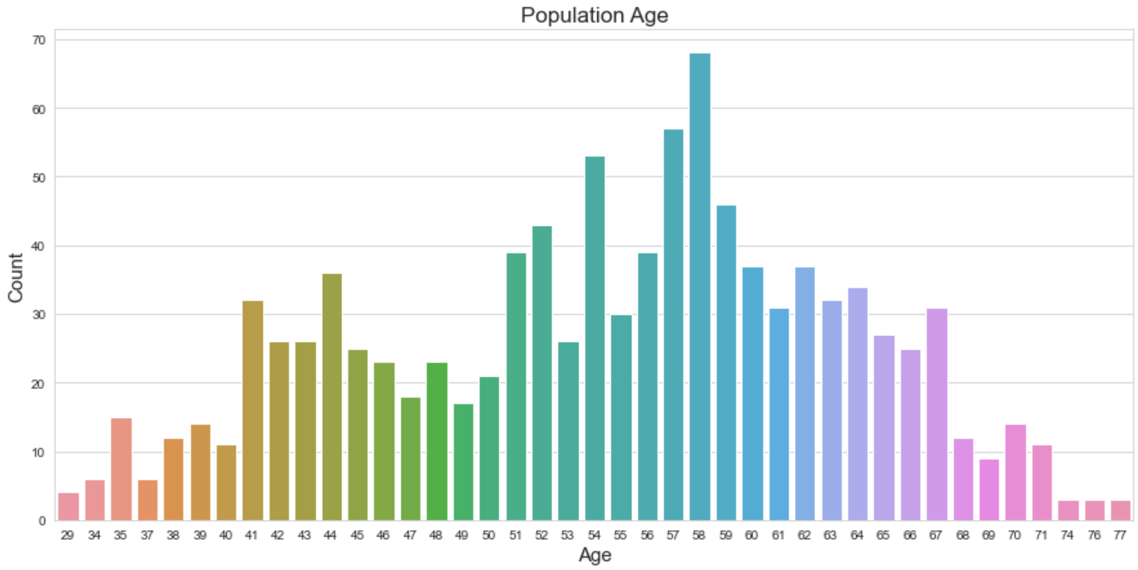
```
#Pie Chart Creation of Heart Disease Population % using MatplotLib

plt.figure(figsize=(10,7))
plt.pie(hd, labels=['Absent','Present'], autopct='%0.0f%%')
plt.title('Heart Disease Population %', fontsize=20)
plt.show()
```



From the overall population, people having heart disease (51%) are more than those who do not have heart disease(49%)

In [27]:

```
#Countplot Creation of Population Age using MatplotLib and Seaborn

plt.figure(figsize=(15,7))
sns.countplot(x='age', data=data)
plt.title('Population Age', fontsize=17)
plt.xlabel('Age', fontsize=15)
plt.ylabel('Count', fontsize=15)
plt.show()
```



-> In this section, the best analysis can be divided into the elderly,middle-aged, young people by looking at the age ranges.

In [28]:

```
#Statistical Analysis

Min_Age=data['age'].min()
Max_Age=data['age'].max()
Mean_Age=data['age'].mean()
print("Minimum Age =",Min_Age)
print("Maximum Age =",Max_Age)
print("Mean Age =",Mean_Age)
```

```
Minimum Age = 29
Maximum Age = 77
Mean Age = 54.43414634146342
```

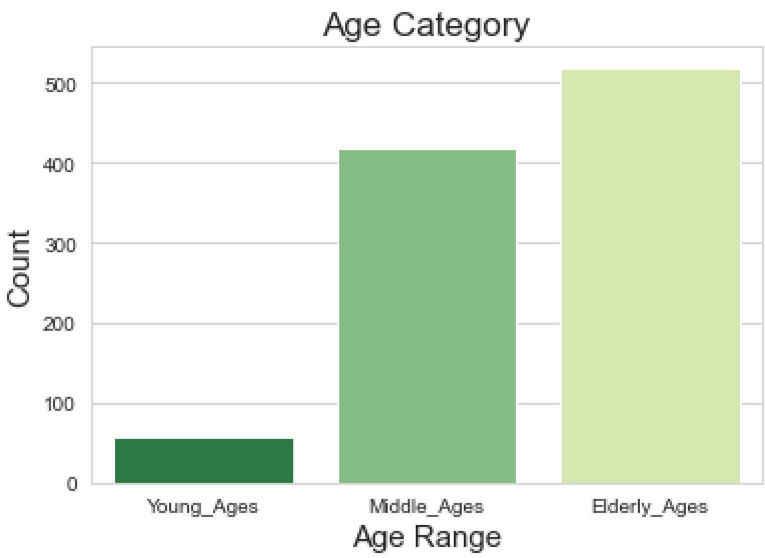
In [29]:

```
#Categorical Analysis

Young_Ages=data[(data['age']>=29) & (data['age']<40)]
Middle_Ages=data[(data['age']>=40) & (data['age']<55)]
Elderly_Ages=data[(data['age']>55)]
print('Young Ages =',len(Young_Ages))
print('Middle Ages =',len(Middle_Ages))
print('Elderly Ages =',len(Elderly_Ages))
```

```
Young Ages = 57
Middle Ages = 419
Elderly Ages = 519
```

In [30]:

```
#Bar Plot Creation of Age Category using MatplotLib and Seaborn

sns.barplot(x=['Young_Ages','Middle_Ages','Elderly_Ages'], y=[len(Young_Age
plt.title('Age Category', fontsize=17)
plt.xlabel('Age Range', fontsize=15)
plt.ylabel('Count', fontsize=15)
plt.show()
```



In [31]:

```
#Converting Numerical Data into Categorical Data

def gender(row):
    if row==1:
        return 'Male'
    elif row==0:
        return 'Female'
```

In [32]:

```
#Applying converted data into our dataset with new column - sex1

data['sex1']=data['sex'].apply(gender)
data.head()
```

Out[32]:

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

In [33]:

```
#Converting Numerical Data into Categorical Data

def age_range(row):
    if row>=29 and row<40:
        return 'Young Age'
    elif row>=40 and row<55:
        return 'Middle Age'
    elif row>55:
        return 'Elder Age'
```

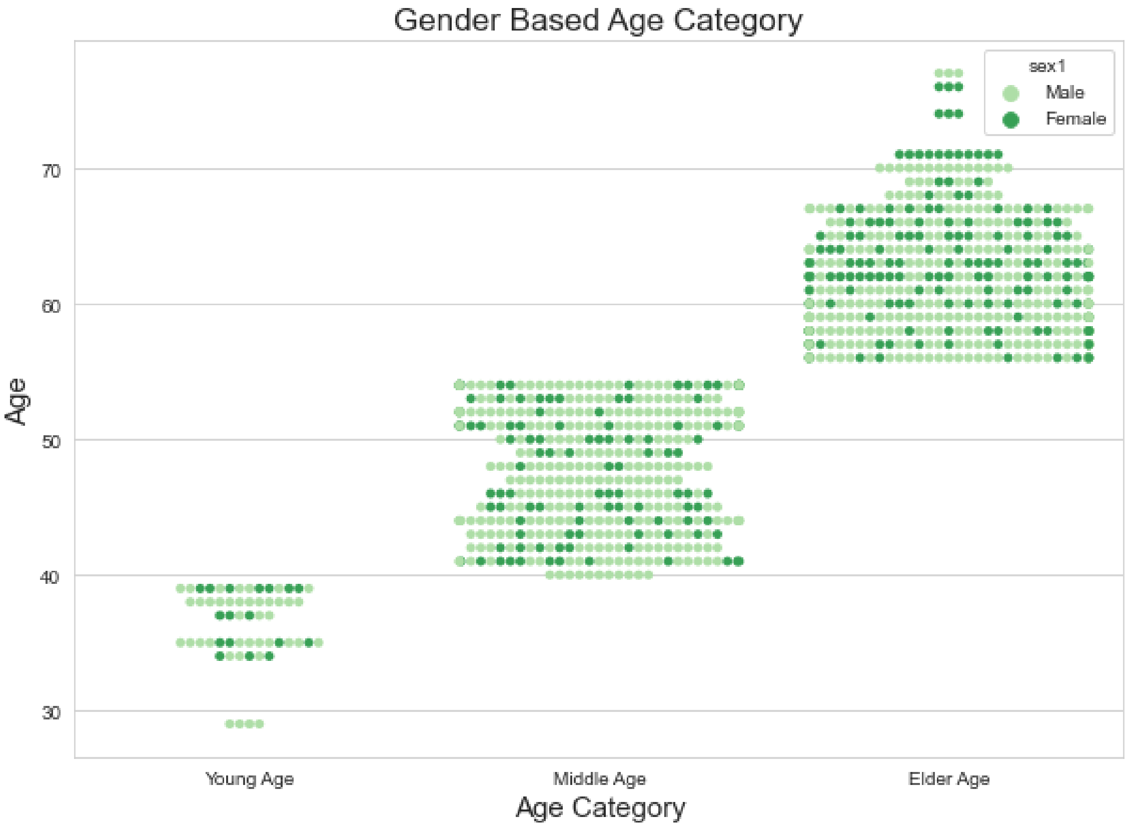
In [34]:

```
#Applying converted data into our dataset with new column - Age_Range

data['Age_Range']=data['age'].apply(age_range)
data.head()
```

Out[34]:

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

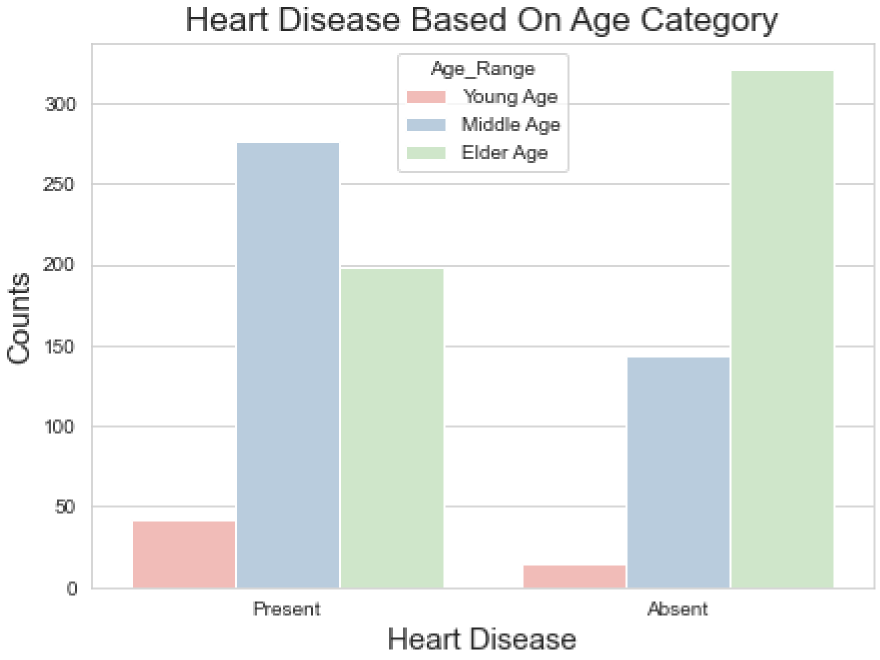
In [43]:
```
#Swarm Plot Creation of Gender Based Age Category using MatplotLib and Seab

plt.figure(figsize=(10,7))
sns.swarmplot(x='Age_Range', y='age', hue='sex1', data=data, order=['Young
plt.title('Gender Based Age Category', fontsize=17)
plt.xlabel('Age Category', fontsize=15)
plt.ylabel('Age', fontsize=15)
plt.show()
```



-> In Our Population Number Of Males are more in Middle Age Category and Females are more in Elder Age Category

In [53]:

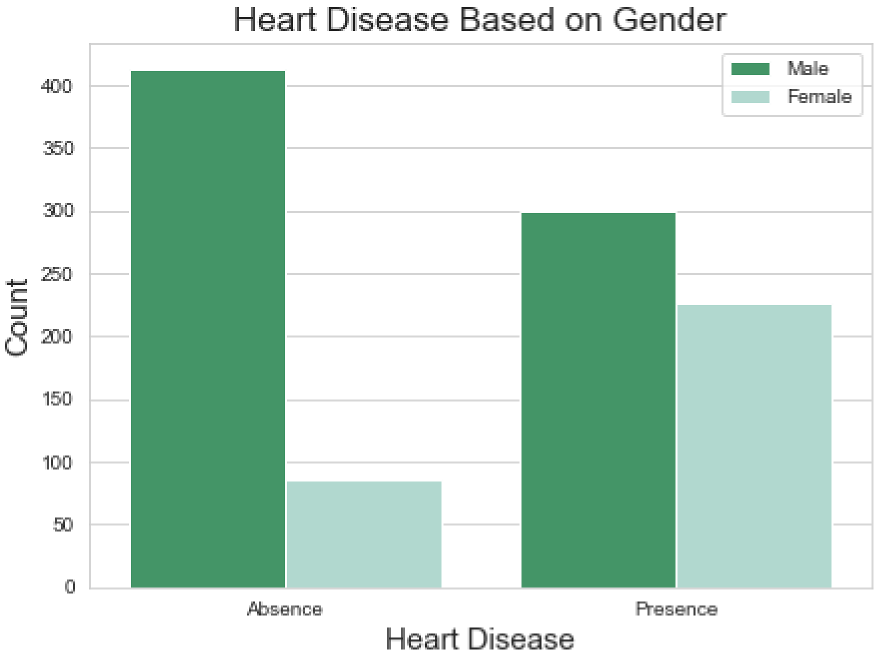
```
#Count Plot Creation of Heart Disease Based On Age Category using MatplotLi

plt.figure(figsize=(7,5))
hue_order=['Young Age', 'Middle Age', 'Elder Age']
sns.countplot(x='Heart_Disease', hue='Age_Range', data=data, order=['Presen
plt.title('Heart Disease Based On Age Category', fontsize=17)
plt.xlabel('Heart Disease', fontsize=15)
plt.ylabel('Counts', fontsize=15)
plt.show()
```



-> Elder Age People are most affected by Heart Disease AND Middle Age People are mostly FREE from any kind of Heart Disease

In [47]:

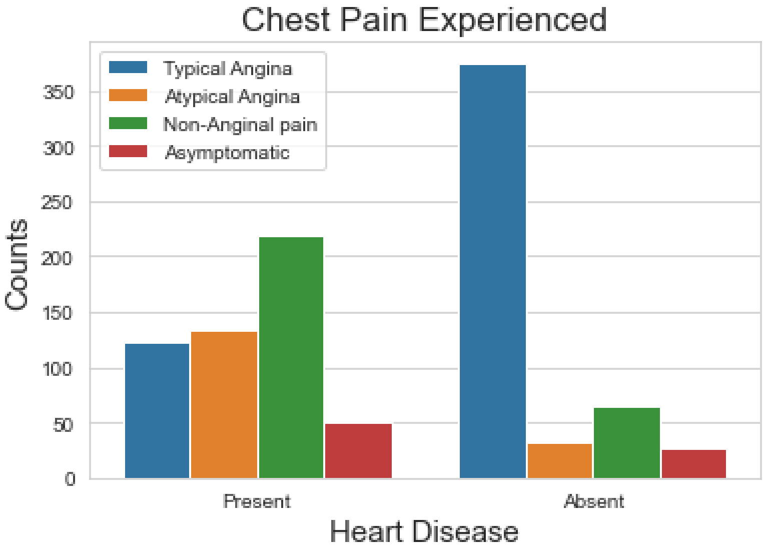
```
#Count Plot Creation of Heart Disease Based on Gender using MatplotLib and

plt.figure(figsize=(7,5))
sns.countplot(x=data['Heart_Disease'], hue='sex1', data=data, palette='BuGn
plt.xlabel('Heart Disease', fontsize=15)
plt.ylabel('Count',fontsize=15)
plt.legend(labels=['Male','Female'])
plt.title('Heart Disease Based on Gender',fontsize=17)
plt.show()
```



-> We can see that Males are more prone to Heart Disease

In [54]:

```
#Count Plot Creation of Chest Pain Experienced using MatplotLib and Seaborn

sns.countplot(x=data['Heart_Disease'], hue='cp', data=data, order=['Present
plt.title('Chest Pain Experienced', fontsize=17)
plt.xlabel('Heart Disease',fontsize=15)
plt.ylabel('Counts',fontsize=15)
plt.legend(labels=['Typical Angina','Atypical Angina','Non-Anginal pain','A
plt.show()
```

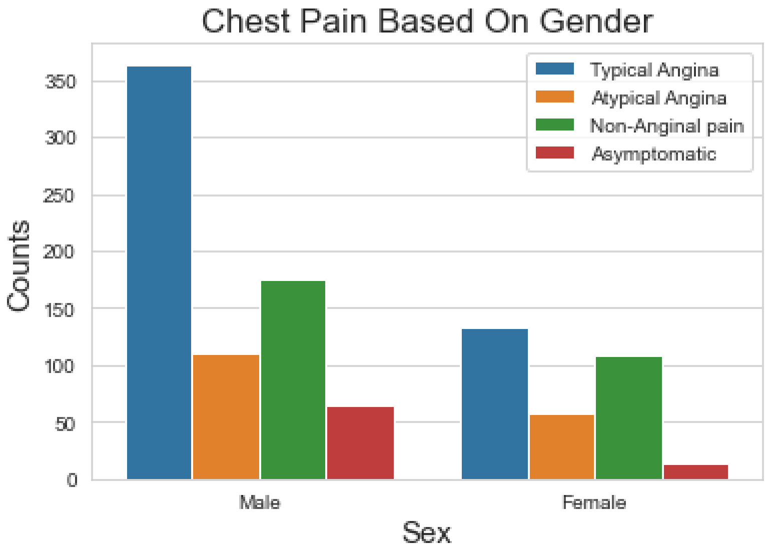


-> It seems people having asymptomatic chest pain have a higher chance of heart disease

-> Asymptomatic Chest pain means neither causing nor exhibiting symptoms of Heart disease.

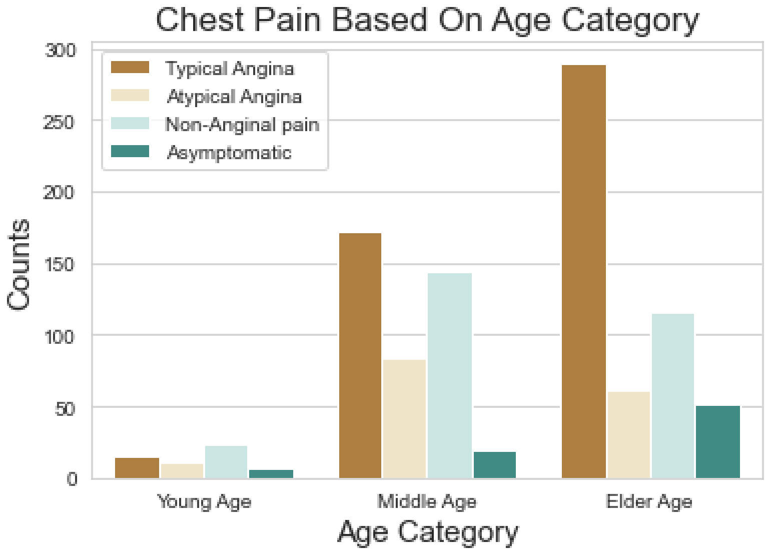
In [56]:

```
#Count Plot Creation of Chest Pain Based On Gender using MatplotLib and Sea

sns.countplot(x=data['sex1'], hue='cp', data=data)
plt.title('Chest Pain Based On Gender', fontsize=17)
plt.xlabel('Sex', fontsize=15)
plt.ylabel('Counts', fontsize=15)
plt.legend(labels=['Typical Angina','Atypical Angina','Non-Anginal pain','A
plt.show()
```



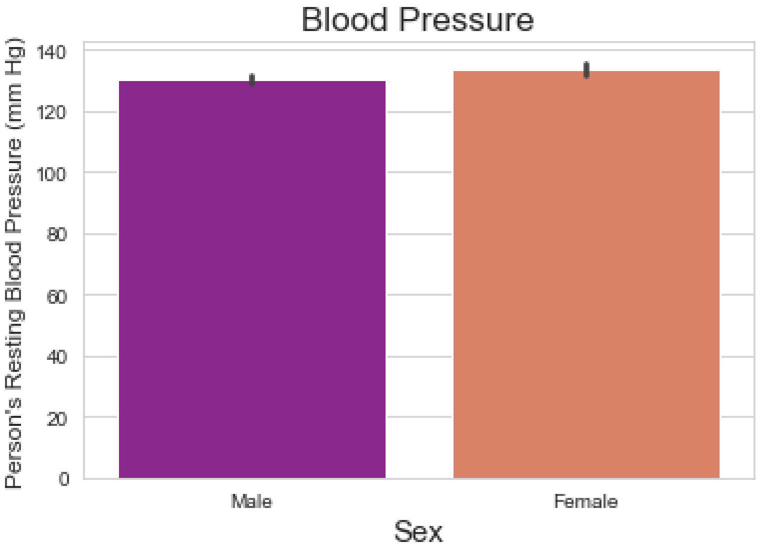
-> We can see that a higher number of men are suffering from Asymptomatic type of Chest Pain

In [57]:
```
#Count Plot Creation of Chest Pain Based On Age Category using MatplotLib a

sns.countplot(x=data['Age_Range'], hue='cp', data=data, order=['Young Age',
plt.title('Chest Pain Based On Age Category', fontsize=17)
plt.xlabel('Age Category', fontsize=15)
plt.ylabel('Counts', fontsize=15)
plt.legend(labels=['Typical Angina','Atypical Angina','Non-Anginal pain','A
plt.show()
```



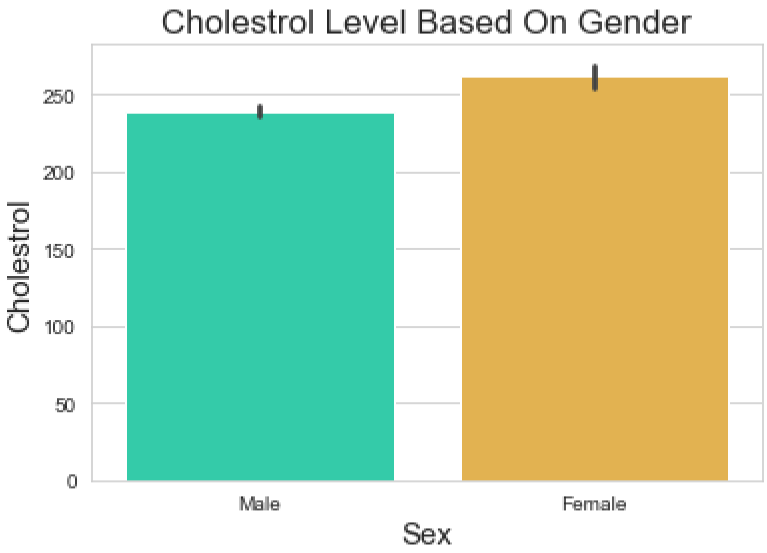
-> There is very high number of Asymptomatic Pain in Elderly age Category

In [58]:
```
#Bar Plot Creation of Person's Resting Blood Pressure (mm Hg) using Matplot

sns.barplot(x='sex1', y='trestbps', data=data, palette='plasma')
plt.title("Blood Pressure", fontsize=17)
plt.xlabel('Sex',fontsize=15)
plt.ylabel("Person's Resting Blood Pressure (mm Hg)", fontsize=12)
plt.show()
```



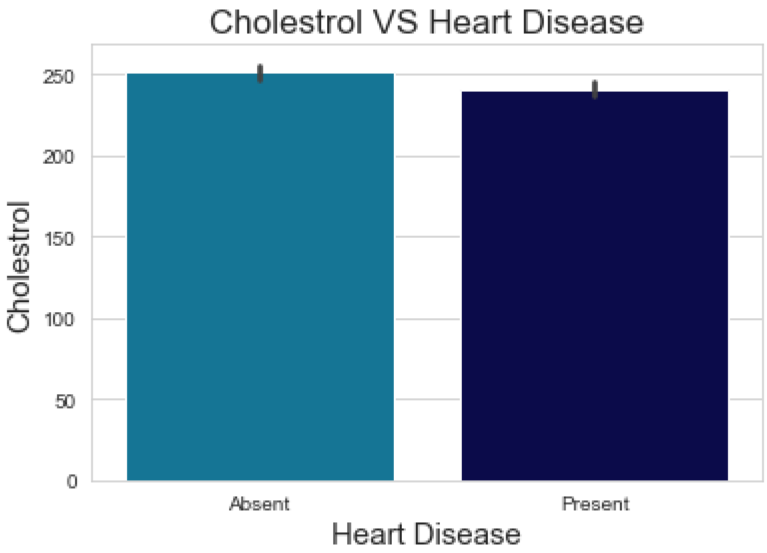
-> Blood Pressure Rate is almost equal in Males and Females

```
In [59]: #Bar Plot Creation of Cholestrol Level Based On Gender using MatplotLib and

         sns.barplot(x='sex1', y='chol', data=data, palette='turbo')
         plt.title("Cholestrol Level Based On Gender", fontsize=17)
         plt.xlabel('Sex',fontsize=15)
         plt.ylabel("Cholestrol", fontsize=15)
         plt.show()
```



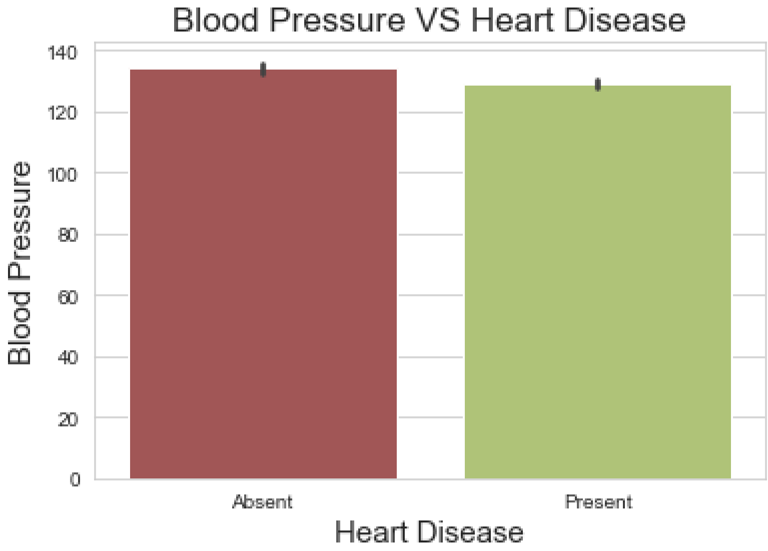
-> females comparatively have higher cholesterol than males

```
In [60]: #Bar Plot Creation of Cholestrol VS Heart Disease using MatplotLib and Seab

         sns.barplot(x='Heart_Disease', y='chol', data=data, palette='ocean_r')
         plt.title('Cholestrol VS Heart Disease', fontsize=17)
         plt.xlabel('Heart Disease', fontsize=15)
         plt.ylabel('Cholestrol', fontsize=15)
         plt.show()
```



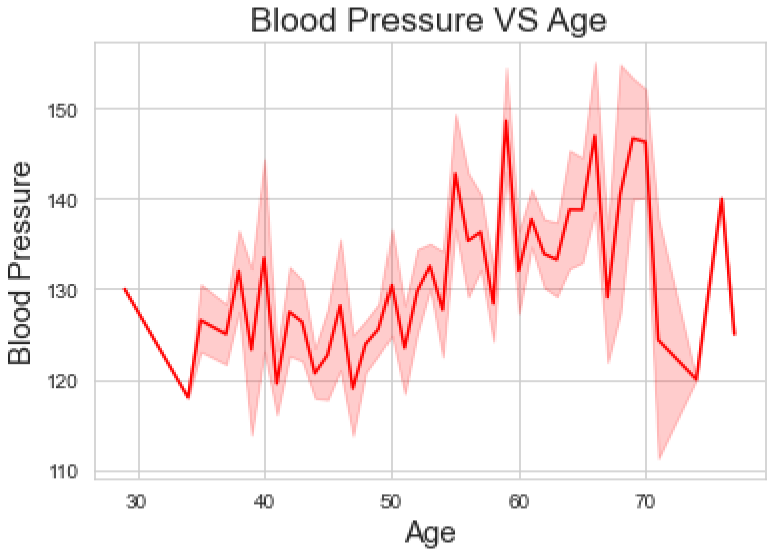
-> Higher Cholestrol level results in more chances Of heart disease

In [61]:
```
#Bar Plot Creation of Blood Pressure VS Heart Disease using MatplotLib and

sns.barplot(x='Heart_Disease', y='trestbps', data=data, palette='tab20b_r')
plt.title('Blood Pressure VS Heart Disease', fontsize=17)
plt.xlabel('Heart Disease', fontsize=15)
plt.ylabel('Blood Pressure', fontsize=15)
plt.show()
```



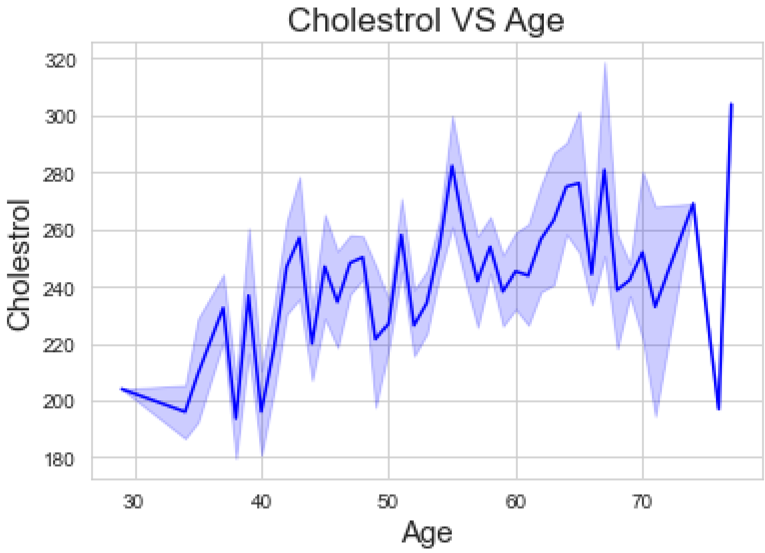
-> Higher Blood Pressure Level results in more chances of heart disease

In [62]:
```
#Line Plot Creation of Blood Pressure VS Age using MatplotLib and Seaborn

sns.lineplot(x='age', y='trestbps', data=data, color='r')
plt.title('Blood Pressure VS Age', fontsize=17)
plt.xlabel('Age', fontsize=15)
plt.ylabel('Blood Pressure', fontsize=15)
plt.show()
```



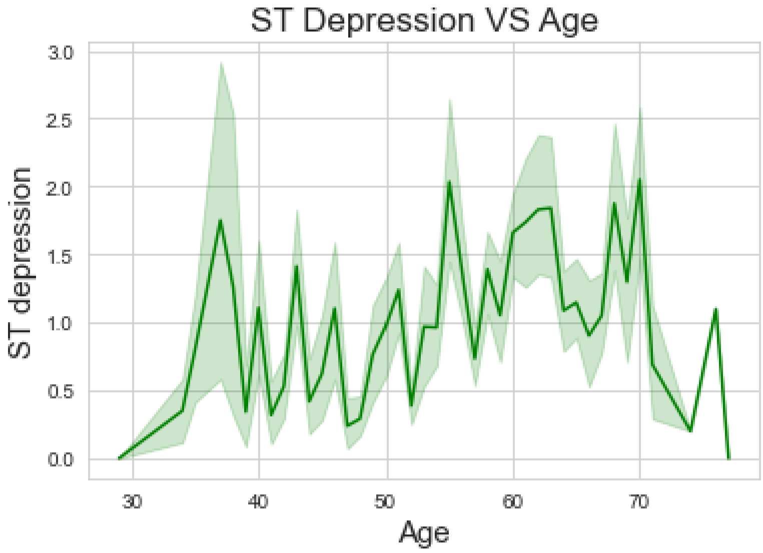
From the above pattern we can conclude that blood pressure increases in the age group of 50 to 60 and somehow continues till 70

```
In [64]: #Line Plot Creation of Cholestrol VS Age using MatplotLib and Seaborn

         sns.lineplot(x='age', y='chol', data=data, color='b')
         plt.title('Cholestrol VS Age', fontsize=17)
         plt.xlabel('Age', fontsize=15)
         plt.ylabel('Cholestrol', fontsize=15)
         plt.show()
```



-> Similarly cholestrol increases in the age group of 50-60

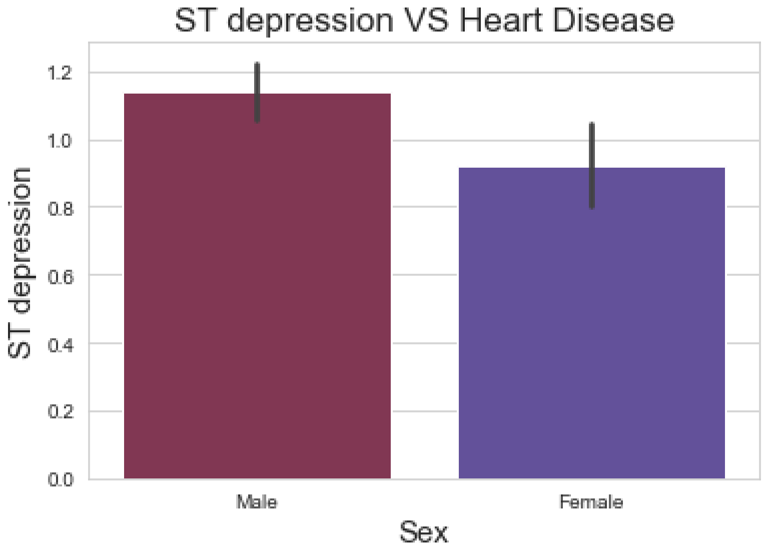
```
In [65]: #Line Plot Creation of ST Depression VS Age using MatplotLib and Seaborn

         sns.lineplot(x='age', y='oldpeak', data=data, color='g')
         plt.title('ST Depression VS Age', fontsize=17)
         plt.xlabel('Age', fontsize=15)
         plt.ylabel('ST depression', fontsize=15)
         plt.show()
```



-> we can observe from here that ST depression mostly increases between the age group of 30-40

In [69]:

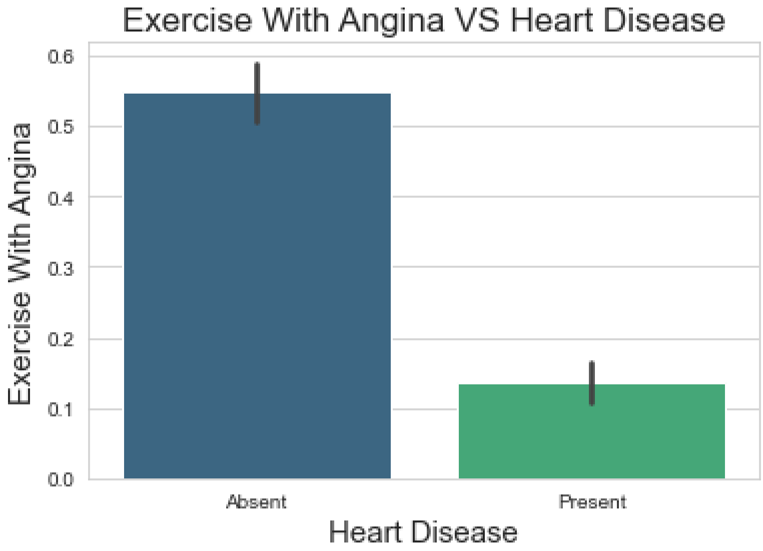
```
#Bar Plot Creation of ST depression VS Heart Disease using MatplotLib and S

sns.barplot(x='sex1', y='oldpeak', data=data, palette='twilight_r')
plt.title('ST depression VS Heart Disease', fontsize=17)
plt.xlabel('Sex', fontsize=15)
plt.ylabel('ST depression', fontsize=15)
plt.show()
```



-> More Males are prone to ST depression as compare to females

In [70]:

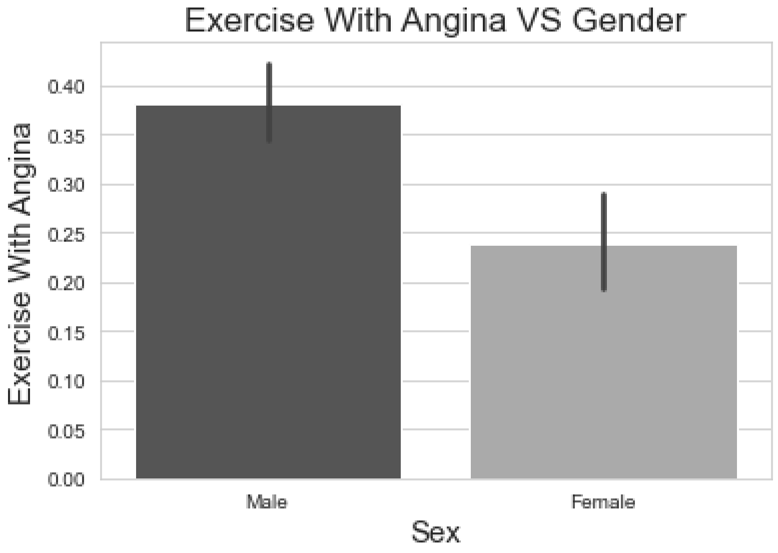
```
#Bar Plot Creation of Exercise With Angina VS Heart Disease using MatplotLi

sns.barplot(x='Heart_Disease', y='exang', data=data, palette='viridis')
plt.title('Exercise With Angina VS Heart Disease', fontsize=17)
plt.xlabel('Heart Disease', fontsize=15)
plt.ylabel('Exercise With Angina', fontsize=15)
plt.show()
```



> If you suffer from Angina, you may be concerned that exercise will make your symptoms worse.

In [71]:

```
#Bar Plot Creation of Exercise With Angina VS Gender using MatplotLib and S

sns.barplot(x='sex1', y='exang', data=data, palette='binary_r')
plt.title('Exercise With Angina VS Gender', fontsize=17)
plt.xlabel('Sex', fontsize=15)
plt.ylabel('Exercise With Angina', fontsize=15)
plt.show()
```

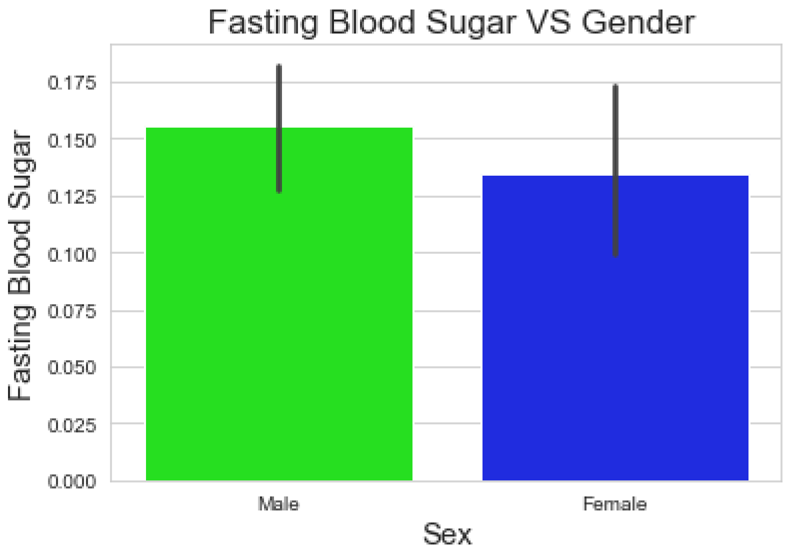


-> Males have have high Exercise Angina

-> A type of chest pain caused by reduced blood flow to the heart.

In [72]:

```
#Bar Plot Creation of Fasting Blood Sugar VS Gender using MatplotLib and Se

sns.barplot(y='fbs', x='sex1', data=data, palette='hsv')
plt.title(' Fasting Blood Sugar VS Gender', fontsize=17)
plt.xlabel('Sex', fontsize=15)
plt.ylabel('Fasting Blood Sugar', fontsize=15)
plt.show()
```



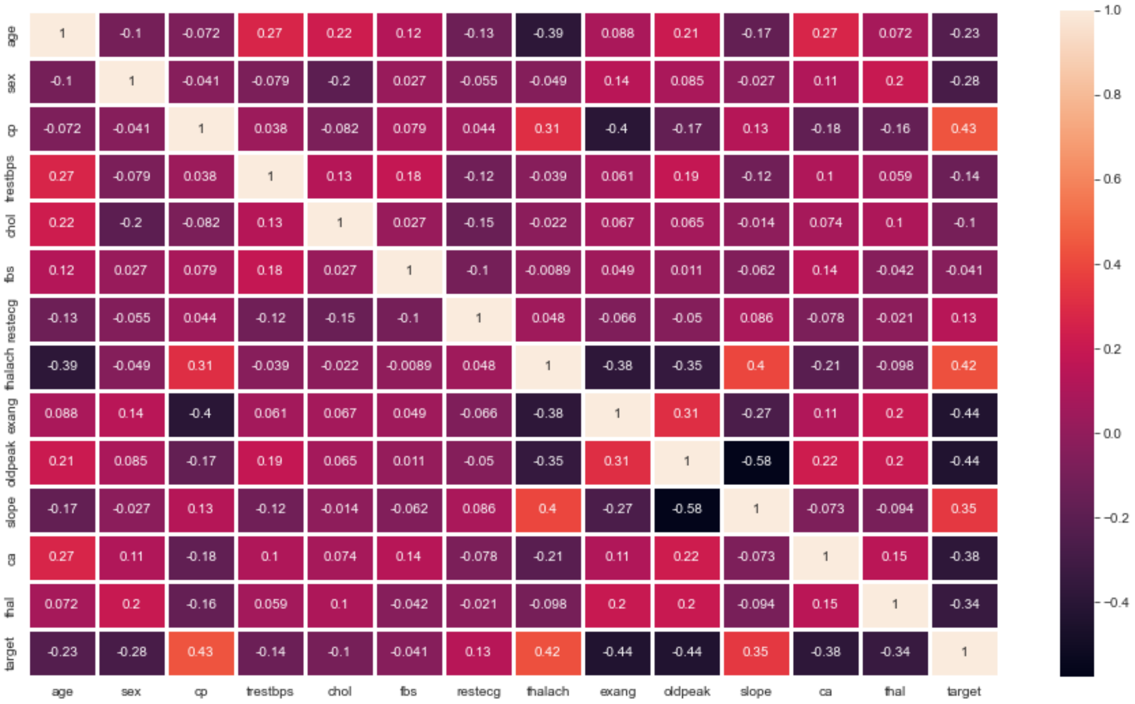
-> Males have high no of Fasting Blood Sugar over 120

In [73]:

```
#Heatmap Creation using Seaborn

plt.figure(figsize=(16,9))
sns.heatmap(data.corr(), annot=True, linewidth=3)
```

Out[73]:

```
<AxesSubplot:>
```



	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
age	1	-0.1	-0.072	0.27	0.22	0.12	-0.13	-0.39	0.088	0.21	-0.17	0.27	0.072	-0.23
sex	-0.1	1	-0.041	-0.079	-0.2	0.027	-0.055	-0.049	0.14	0.085	-0.027	0.11	0.2	-0.28
cp	-0.072	-0.041	1	0.038	-0.082	0.079	0.044	0.31	-0.4	-0.17	0.13	-0.18	-0.16	0.43
trestbps	0.27	-0.079	0.038	1	0.13	0.18	-0.12	-0.039	0.061	0.19	-0.12	0.1	0.059	-0.14
chol	0.22	-0.2	-0.082	0.13	1	0.027	-0.15	-0.022	0.067	0.065	-0.014	0.074	0.1	-0.1
fbs	0.12	0.027	0.079	0.18	0.027	1	-0.1	-0.0089	0.049	0.011	-0.062	0.14	-0.042	-0.041
restecg	-0.13	-0.055	0.044	-0.12	-0.15	-0.1	1	0.048	-0.066	-0.05	0.086	-0.078	-0.021	0.13
thalach	-0.39	-0.049	0.31	-0.039	-0.022	-0.0089	0.048	1	-0.38	-0.35	0.4	-0.21	-0.098	0.42
exang	0.088	0.14	-0.4	0.061	0.067	0.049	-0.066	-0.38	1	0.31	-0.27	0.11	0.2	-0.44
oldpeak	0.21	0.085	-0.17	0.19	0.065	0.011	-0.05	-0.35	0.31	1	-0.58	0.22	0.2	-0.44
slope	-0.17	-0.027	0.13	-0.12	-0.014	-0.062	0.086	0.4	-0.27	-0.58	1	-0.073	-0.094	0.35
ca	0.27	0.11	-0.18	0.1	0.074	0.14	-0.078	-0.21	0.11	0.22	-0.073	1	0.15	-0.38
thal	0.072	0.2	-0.16	0.059	0.1	-0.042	-0.021	-0.098	0.2	0.2	-0.094	0.15	1	-0.34
target	-0.23	-0.28	0.43	-0.14	-0.1	-0.041	0.13	0.42	-0.44	-0.44	0.35	-0.38	-0.34	1

In [ ]:

```
sns.heat
```