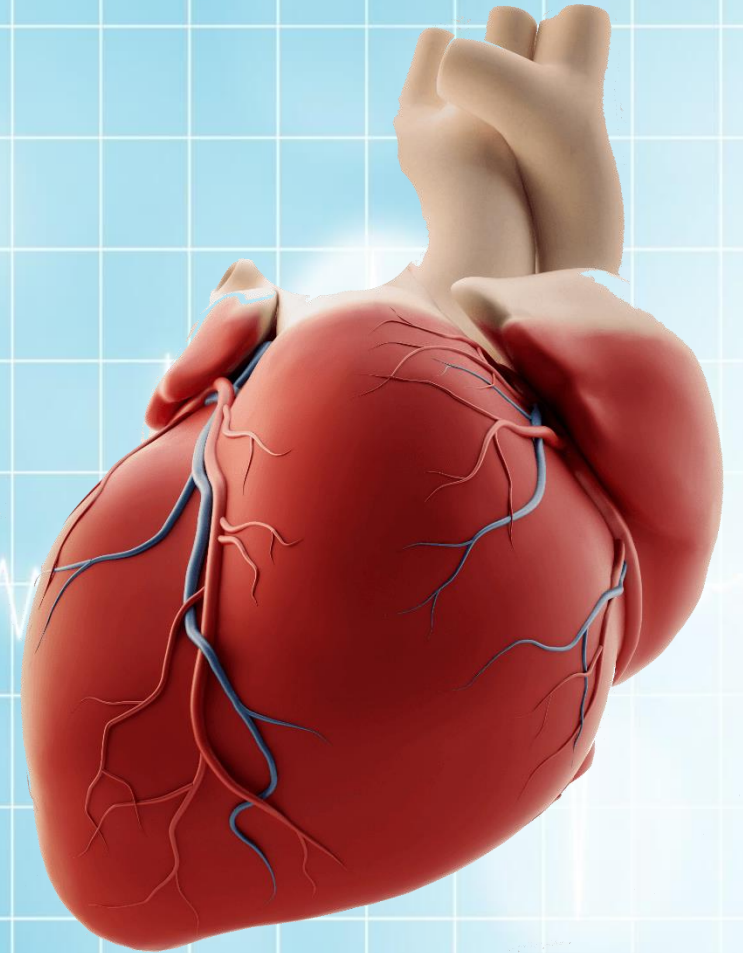




# 심장병 EDA 와 예측

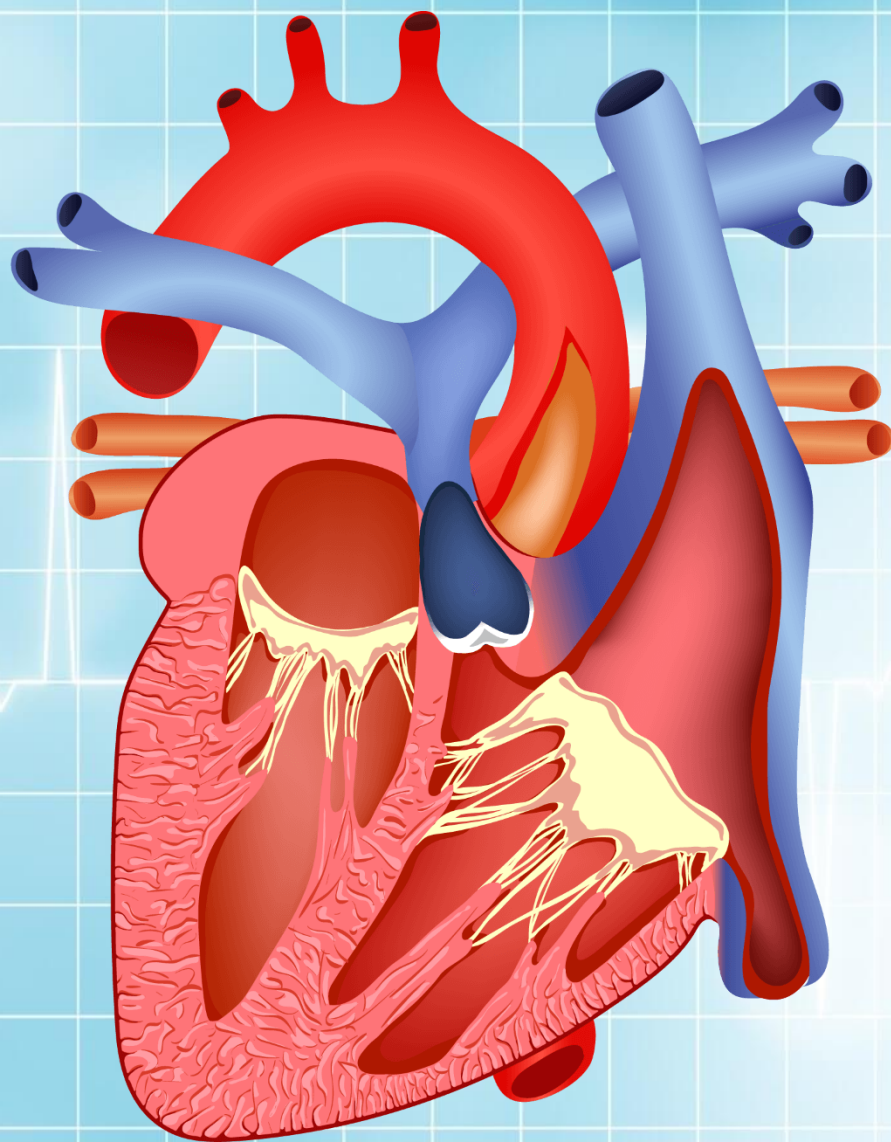
인공지능-컴퓨터공학전공-2020308002-등원호



# 목록 CONTENTS

1. 개요.
2. Data explanation데이터 설명
3. Data library데이터 라이브러리
4. Initial Data Exploration초기 데이터 탐색
- 5.EDA
6. Dataset Pre-processing데이터세트 전처리
7. Model Implementation모델 구현
- 8.결론
- 9.참고 문헌.









## 1.개요 및 필요성

### 개요

- 심장병은 개개인의 성별, 연령, 혈압, 기타 데이터를 통해 분석됩니다.
- 전처리를 통한 데이터 셋 정리
- 알고리즘을 통해 심장병 예측



### 필요성

- 심장병은 한국에서 두 번째로 큰 비정상 사망 원인입니다.
- 심장병은 매년 1790만 명의 생명을 앗아갑니다
- 심장병의 갈수록 높아지는 발병률은 더 많은 관심을 필요로 합니다.

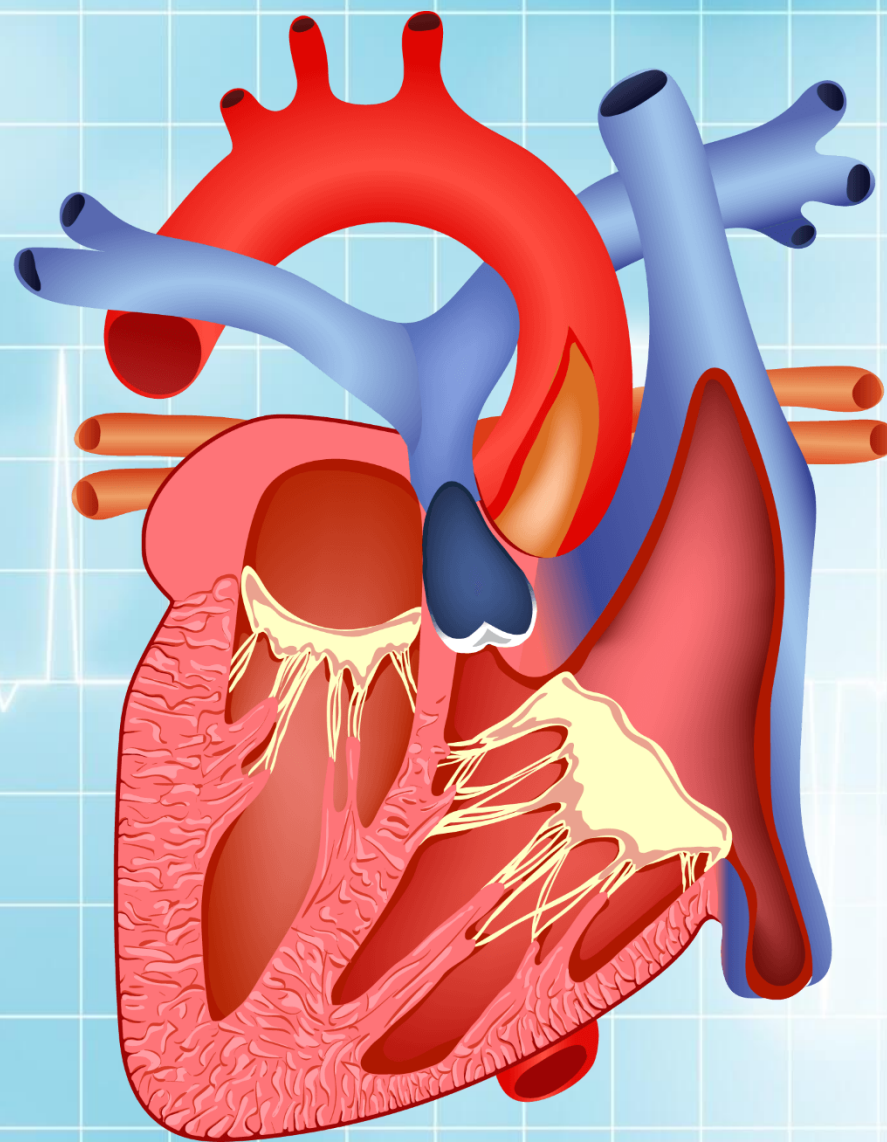


## 1.개요 및 필요성

# 인공지능과 심장병

- ◆ 심장병 환자의 데이터를 인공지능으로 분석하여 어떤 사람이 심장병에 더 잘 걸리는지, 심장병 환자는 어떤 특징을 가지고 있는지 알 수 있으며, 심장병에 걸리지 않았더라도 데이터를 통해 심장병을 예방할 수 있습니다.







## 2.Data explanation데이터 설명

Variable Name	Description	Sample Data
Age	Patient Age (in years)	63; 37; ...
Sex	Gender of patient (0 = male; 1 = female)	1; 0; ...
cp	Chest pain type (4 values: 0, 1, 2, 3)	3; 1; 2; ...
trestbps	resting blood pressure (in mm Hg)	145; 130; ...
chol	Serum cholestoral (in mg/dl)	233; 250; ...
fbs	Fasting blood sugar > 120 mg/dl (1 = true; 0 = false)	1; 0; ...
restecg	Resting electrocardiographic results (values 0, 1, 2)	0; 1; ...
thalach	Maximum heart rate achieved	150; 187; ...
exang	Exercise induced angina (1 = yes; 0 = no)	1; 0; ...
oldpeak	ST depression induced by exercise relative to rest	2.3; 3.5; ...
slope	The slope of the peak exercise ST segment (values 0, 1, 2)	0; 2; ...
ca	number of major vessels (0-4) colored by flourosopy	0; 3; ...
thal	(3 = normal; 6 = fixed defect; 7 = reversable defect)	1; 3; ...
Target	Target column (1 = Yes; 0 = No)	1; 0; ...

이 데이터 세트에는 14개의 변수가 있습니다.  
9개의 범주형 변수와 5개의 연속형 변수.  
다음은 데이터셋의 구조입니다.

Age-환자 나이

Sex-환자의 성별

Cp-흉통 유형

Trestbps-안정시 혈압

Chol-혈청 콜레스테롤

Fbs-공복 혈당

Restecg-안정시 심전도 결과

Thalach-최대 심박수

Exang-운동 유발 협심증

Oldpeak-휴식에 비해 운동으로 인한 ST 저하

Slope-피크 운동 ST 세그먼트의 기울기

Ca-플로로소피 0에 의해 착색된 주요 혈관(0-4)의 수

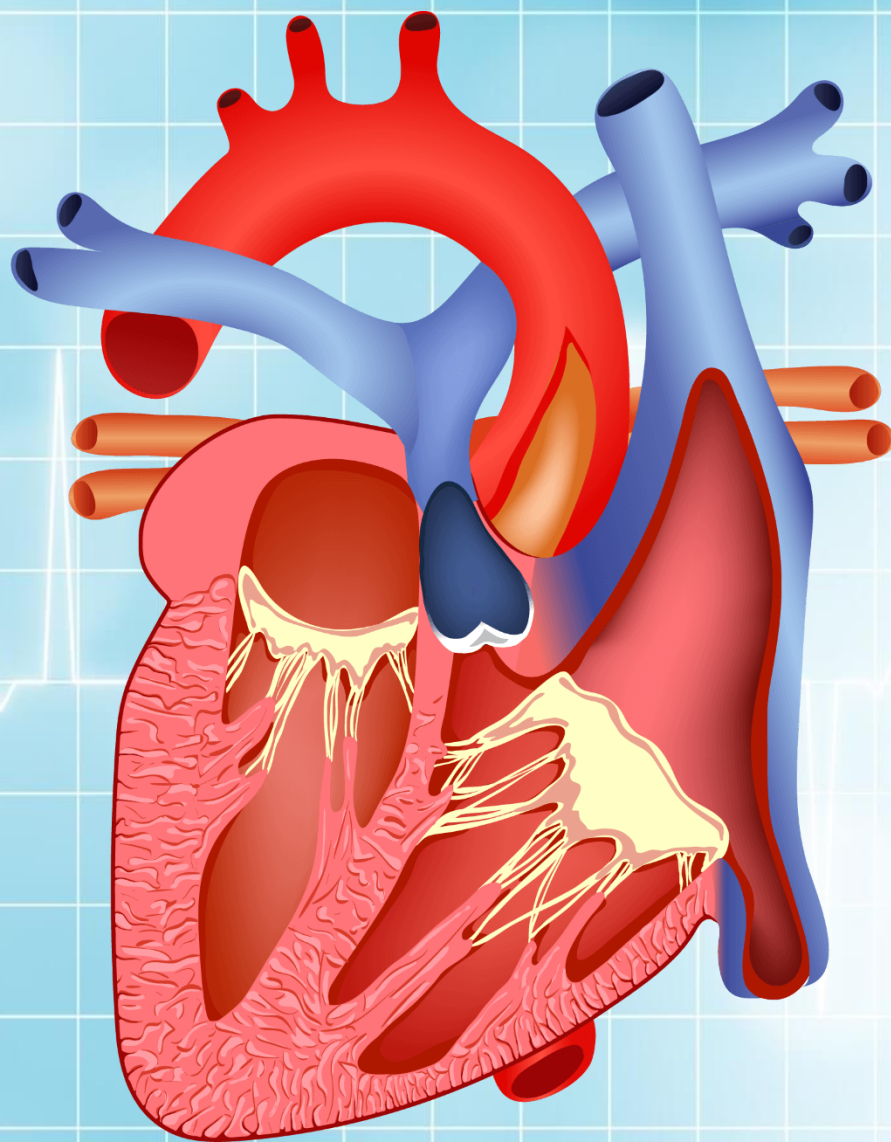
Thal-(3 = 정상, 6 = 고정 결함, 7 = 회복 가능한 결함)

Target-대상 열(1 = 예, 0 = 아니요)





Data library  
데이터 라이브러리







## 3Data library데이터 라이브러리

# Data library import

```
# --- Importing Libraries ---
```

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
import os
import yellowbrick
import pickle
```

```
from matplotlib.collections import PathCollection
from statsmodels.graphics.gofplots import qqplot
from sklearn.preprocessing import MinMaxScaler
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.neighbors import KNeighborsClassifier
from sklearn.svm import SVC
from sklearn.naive_bayes import GaussianNB
from sklearn.tree import DecisionTreeClassifier
```

Yellowbrick:본질적으로 scikit-learn의 확장은 매우 실용적인 ML 모델 시각화 도구를 제공합니다.

Pickle:Python 개체와 해당 개체를 바이트 스트림으로 변환합니다

numpy-다양한 수치 연산, 변환 기능 등을 갖는 모듈

Pandas-데이터를 읽어 드리고 유지하고 관리 할 수있는 모듈

OS : 파일을 복사하거나 디렉토리를 생성하고 특정 디렉토리 내의 파일 목록을 구하고자 할 때 이용

Matplotlib.pyplot :Pyplot은 MATLAB과 유사한 그리기 API를 제공하는 Matplotlib의 하위 라이브러리입니다.

seaborn-데이터를 멋지게 표시하는 모듈

Warnings:프로그램이 경고를 보냅니다

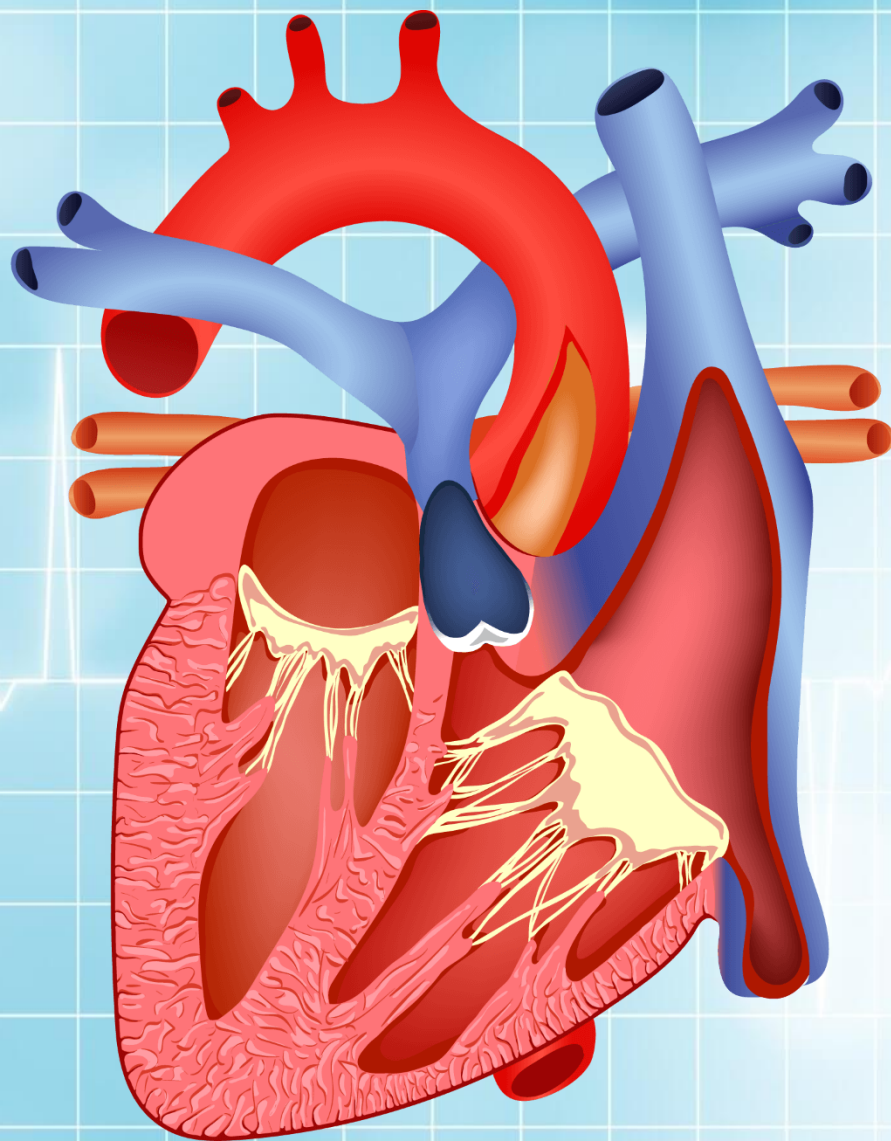
```
from sklearn.ensemble import RandomForestClassifier, GradientBoostingClassifier, AdaBoostClassifier, ExtraTreesClassifier
from sklearn.metrics import classification_report, accuracy_score
from xgboost import XGBClassifier
from yellowbrick.classifier import PrecisionRecallCurve, ROCAUC, ConfusionMatrix
from yellowbrick.style import set_palette
from yellowbrick.model_selection import LearningCurve, FeatureImportances
from yellowbrick.contrib.wrapper import wrap
```

```
# --- Libraries Settings ---
```

```
warnings.filterwarnings('ignore')
sns.set_style('whitegrid')
plt.rcParams['figure.dpi']=100
set_palette('dark')
```



Initial Data  
Exploration  
초기 데이터 탐색





## 4.1. 초기 데이터 탐색

◆ 탐색할 첫 번째 유형의 변수는 범주형 변수입니다.







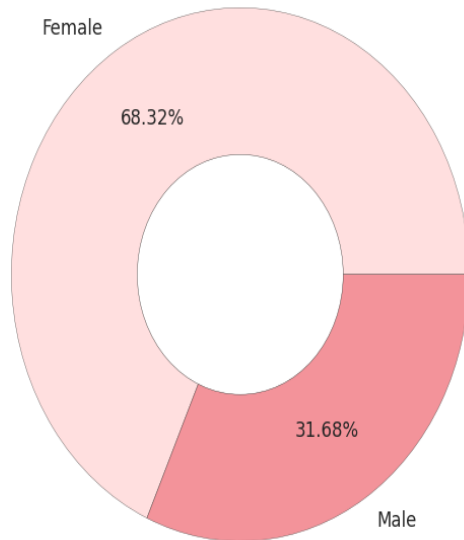
## 4.1.1sex (Gender)성별

```
df.sex.value_counts(dropna=False)
```

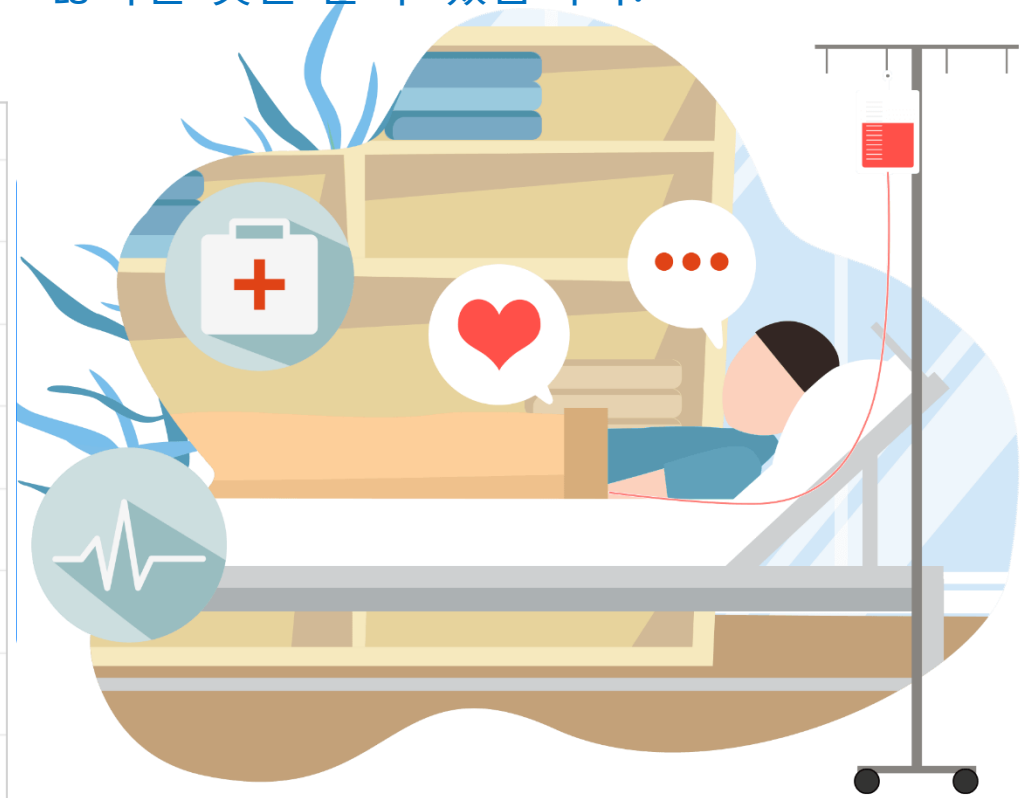
데이터에서 우리는 심장병이 있는 여성보다 심장병이 있는 남성이 더 많다는 것을 알 수 있습니다.

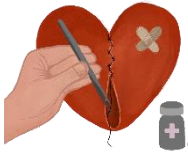
Sex (Gender) Distribution

Pie Chart



Histogram



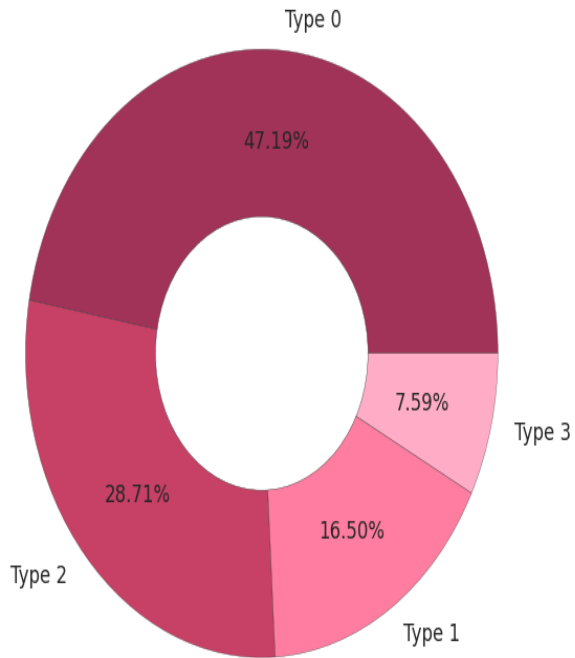


## 4.1.2cp (Chest Pain Type) 흉통 유형

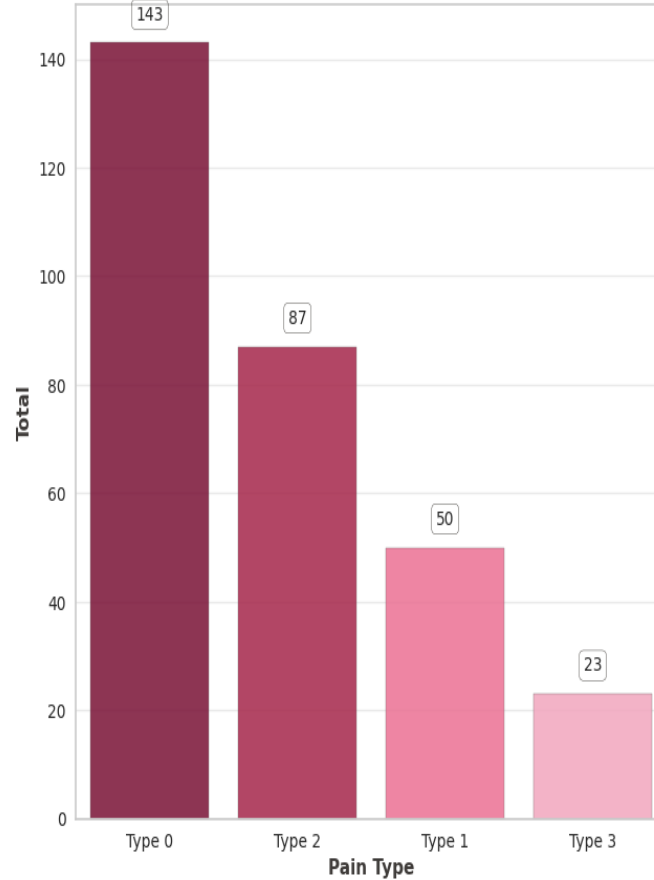
```
df.cp.value_counts(dropna=False)
```

Sex (Gender) Distribution  
Chest Pain Type Distribution

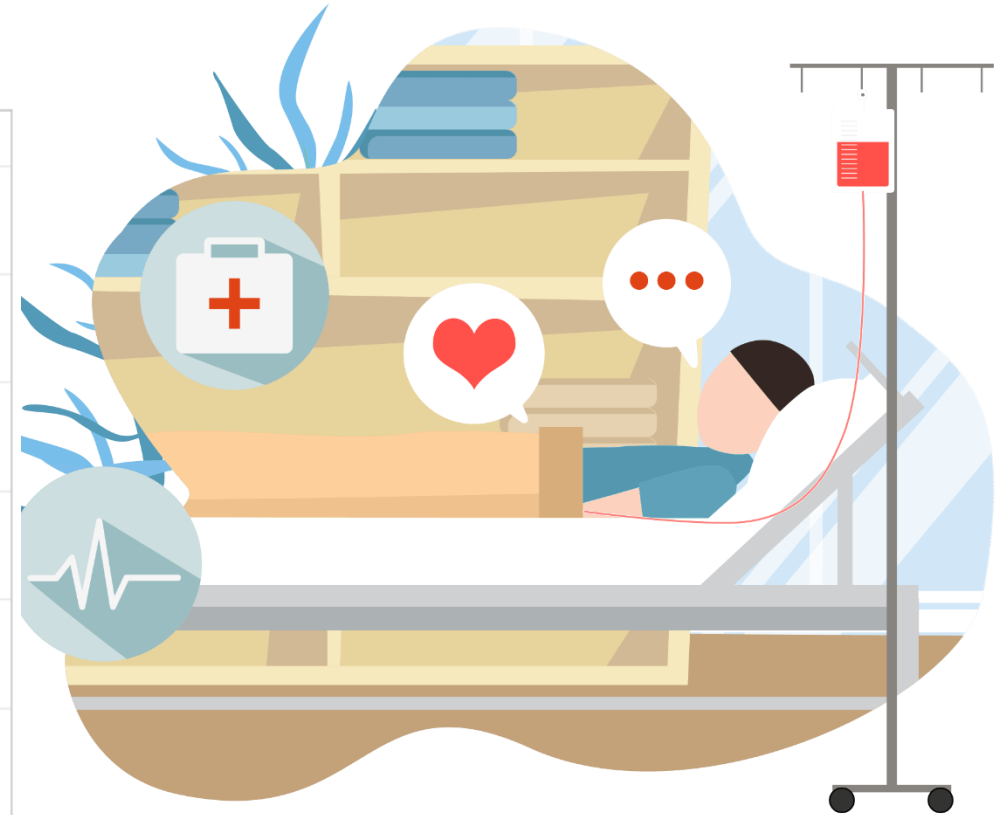
Pie Chart



Histogram



흉통 유형 0은 다른 유형의 흉통에 비해 가장 높은 수치를 나타냅니다.



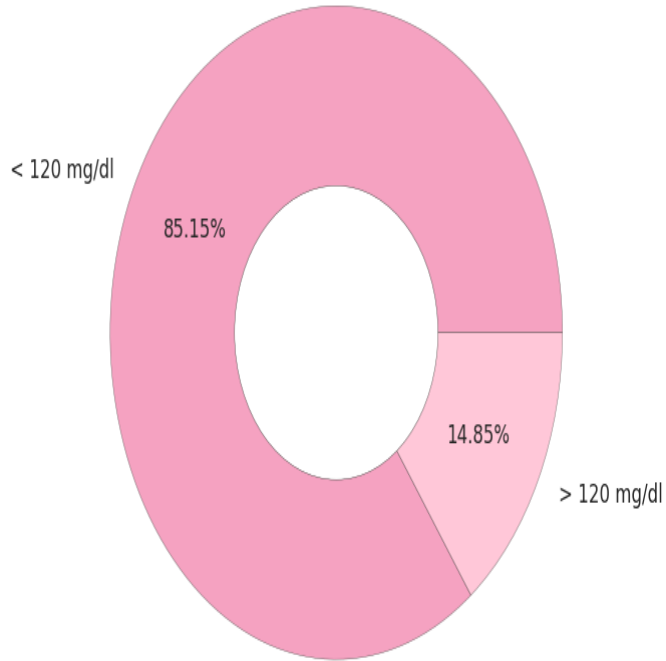


### 4.1.3fbs (Fasting Blood Sugar)공복 혈당

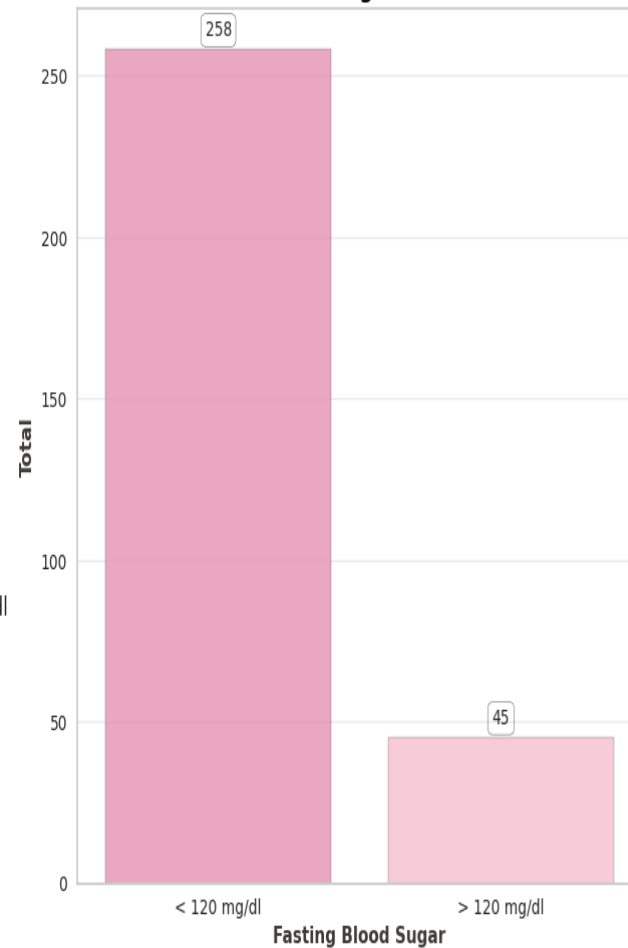
```
df.fbs.value_counts(dropna=False)
```

Fasting Blood Sugar Distribution

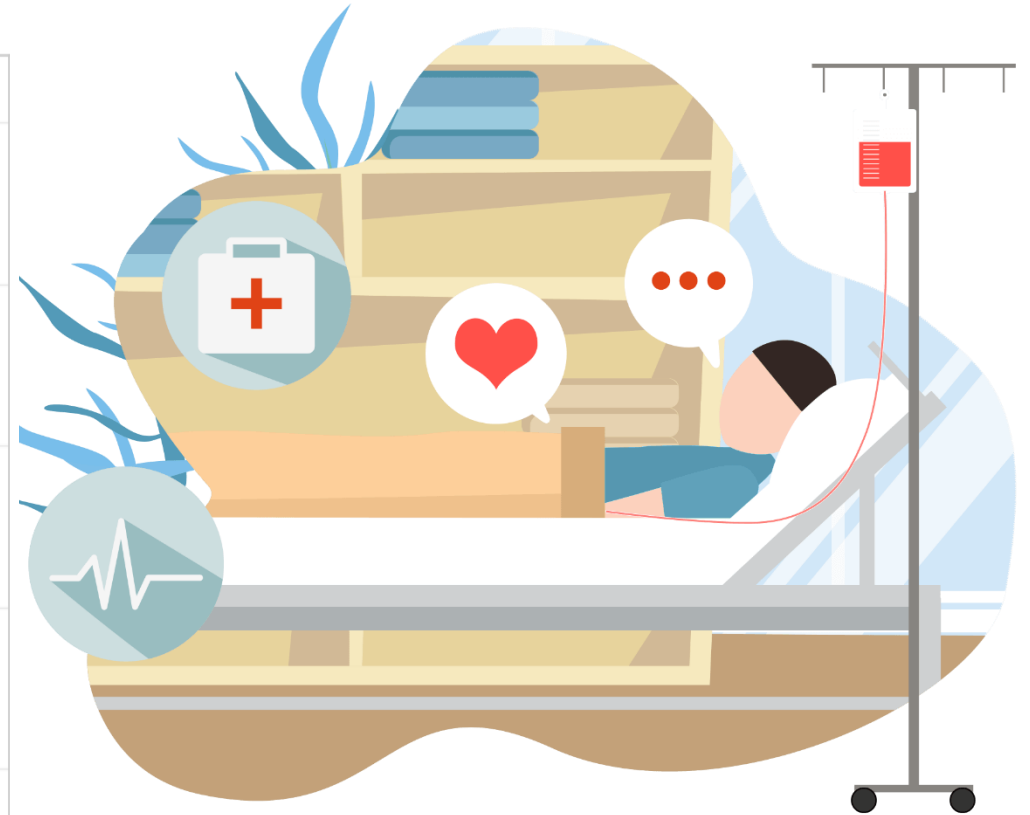
Pie Chart



Histogram



공복혈당이 120mg/dl 미만인 환자의 수가 가장 많은 것을 알 수 있습니다.



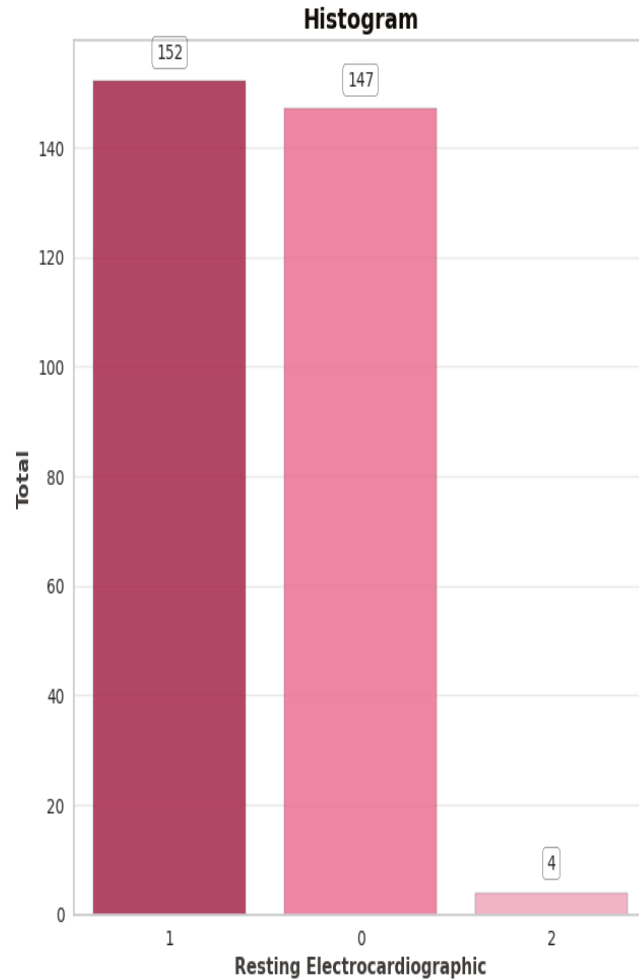
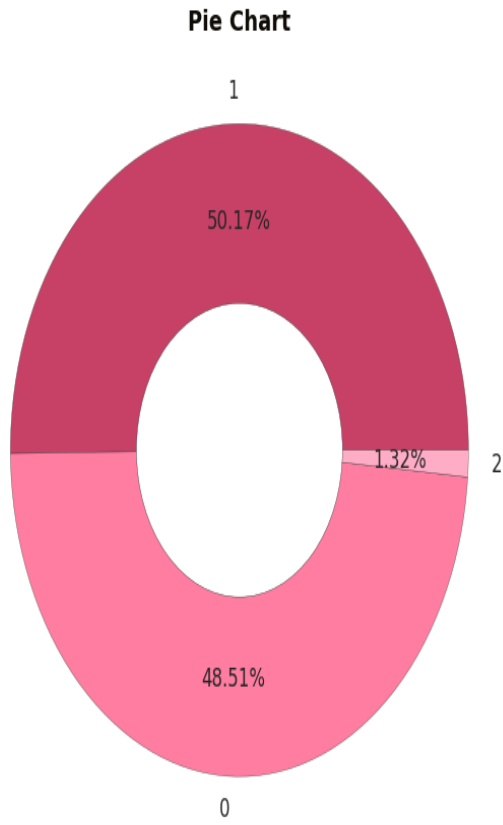




#### 4.1.4 restecg (Resting Electrocardiographic Results)안정시 심전도 결과

```
df.restecg.value_counts(dropna=False)
```

Resting Electrocardiographic Distribution



결과가 1 및 0인 정지 심전도는 결과 2보다 분포가 높습니다.  
또한 결과 1의 분포가 다른 결과와 비교하여 가장 높습니다.



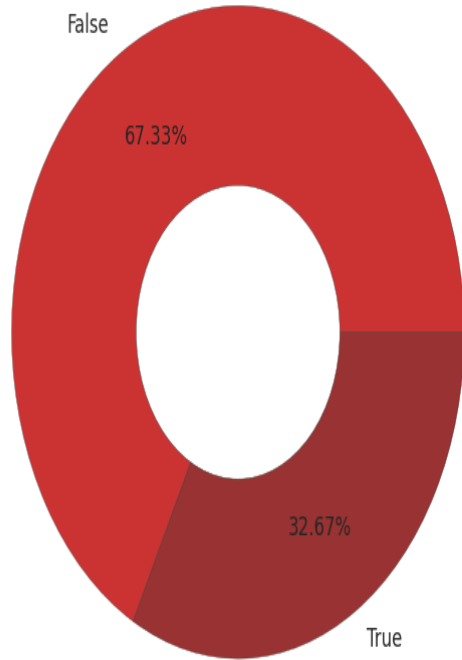


## 4.1.5exang (Exercise Induced Angina)운동 유발 협심증

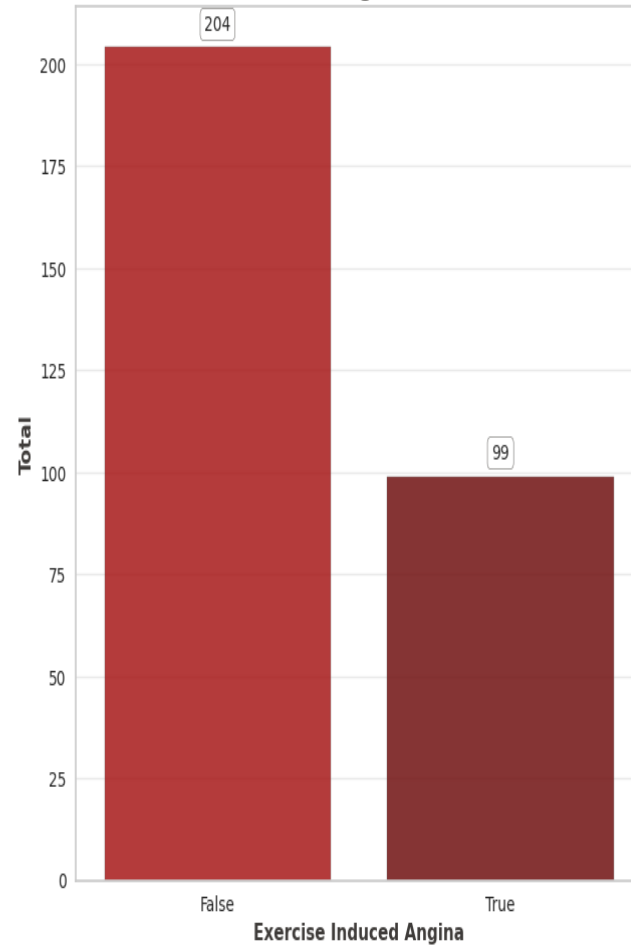
```
df.exang.value_counts(dropna=False)
```

Exercise Induced Angina Distribution

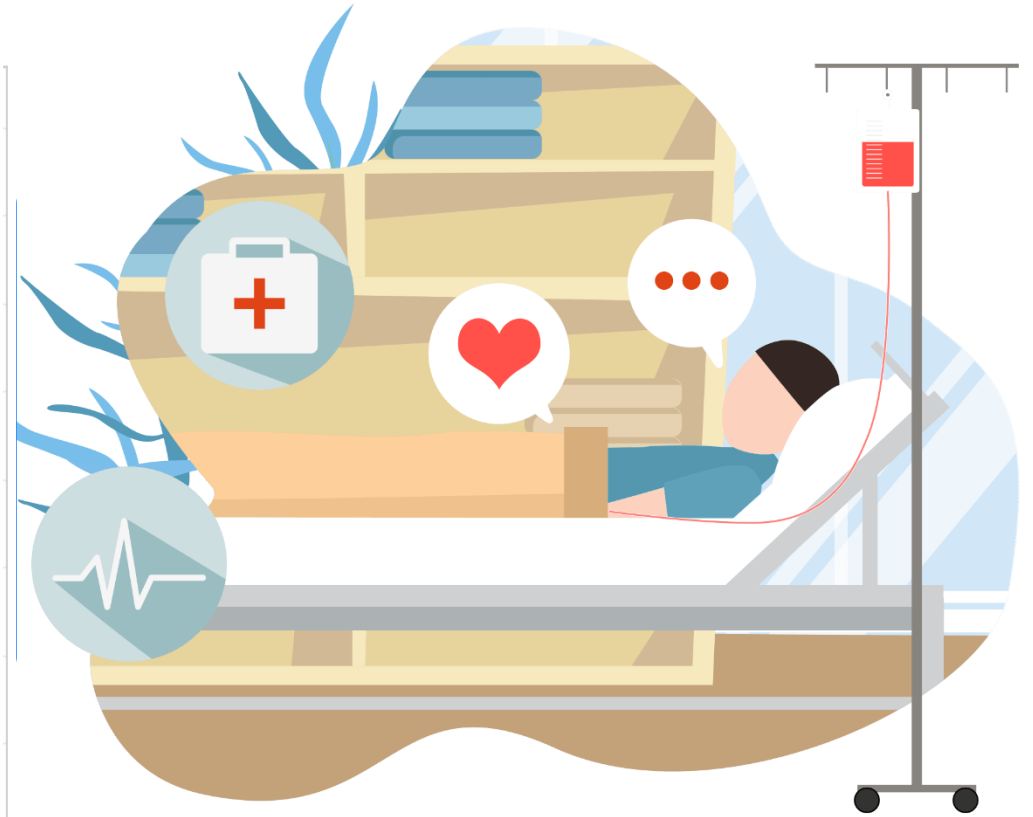
Pie Chart



Histogram



운동 유발 협심증이 없는 환자는 운동 유발 협심증 환자에 비해 가장 높습니다.

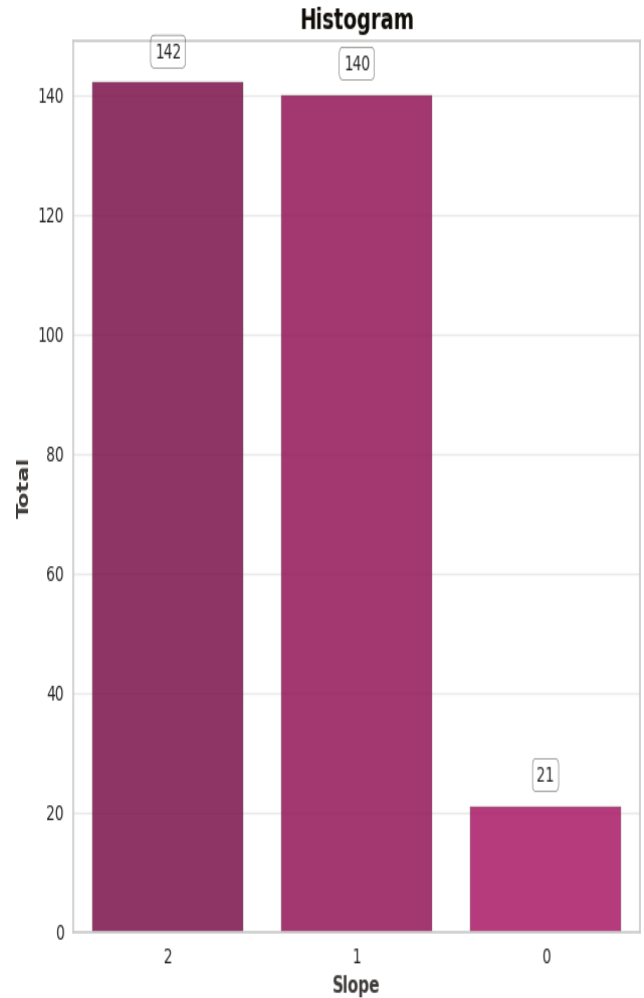
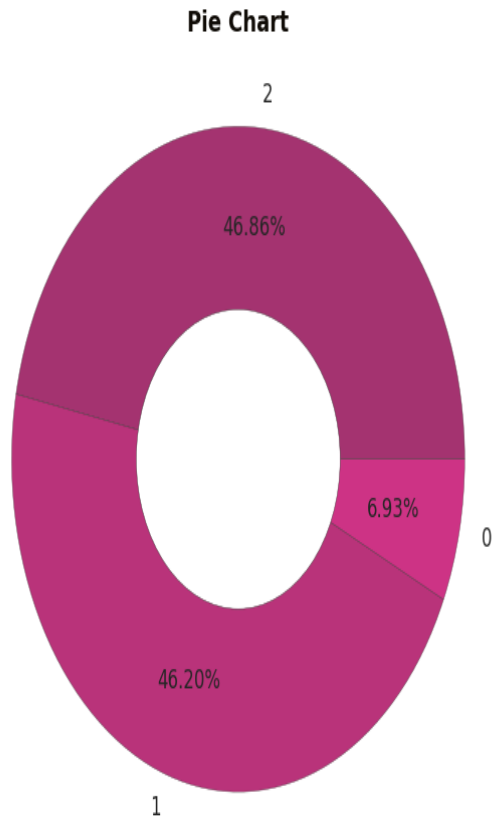




## 4.1.6slope (Slope of the Peak Exercise)피크 운동 ST 세그먼트의 기울기

```
df.slope.value_counts(dropna=False)
```

Slope of the Peak Exercise Distribution



기울기 1과 기울기 2의 분포는 거의 같습니다.  
또한 기울기 2는 다른 것에 비해 분포가 가장 높습니다.



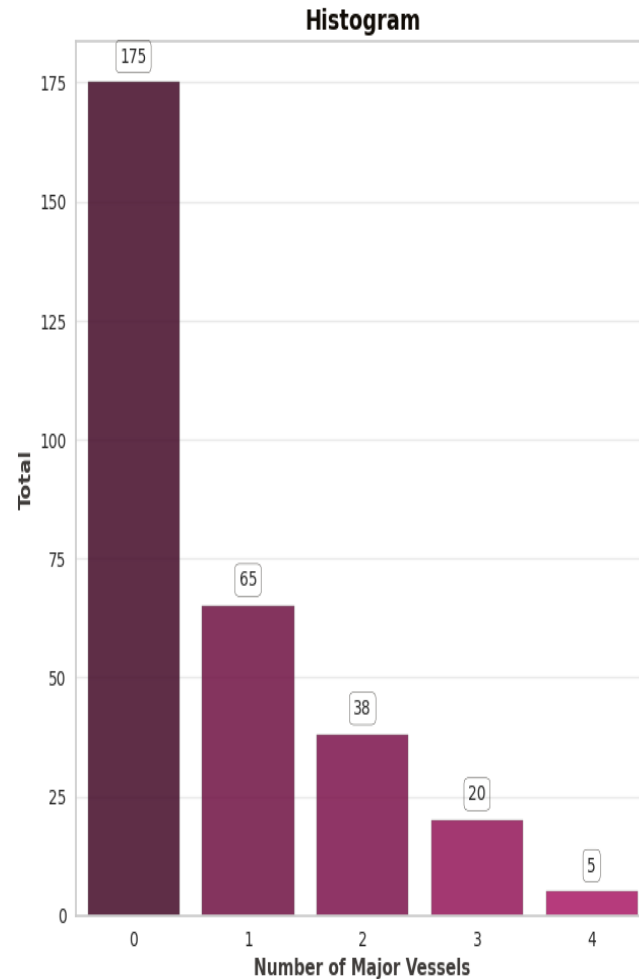
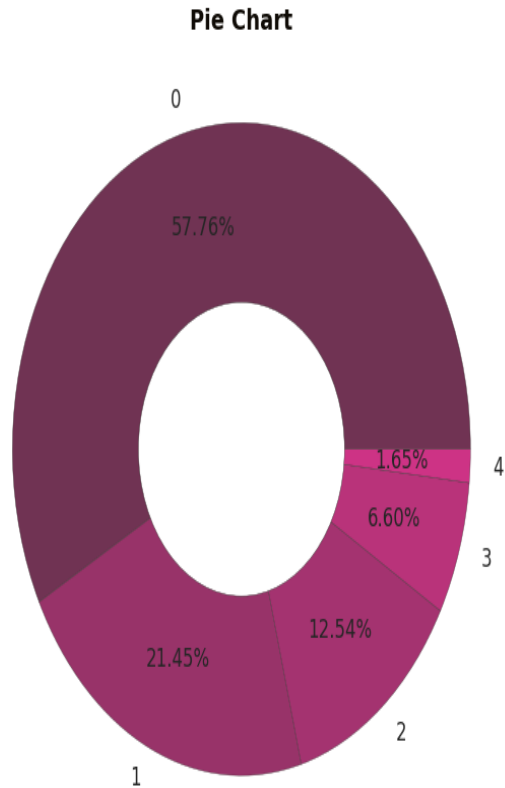




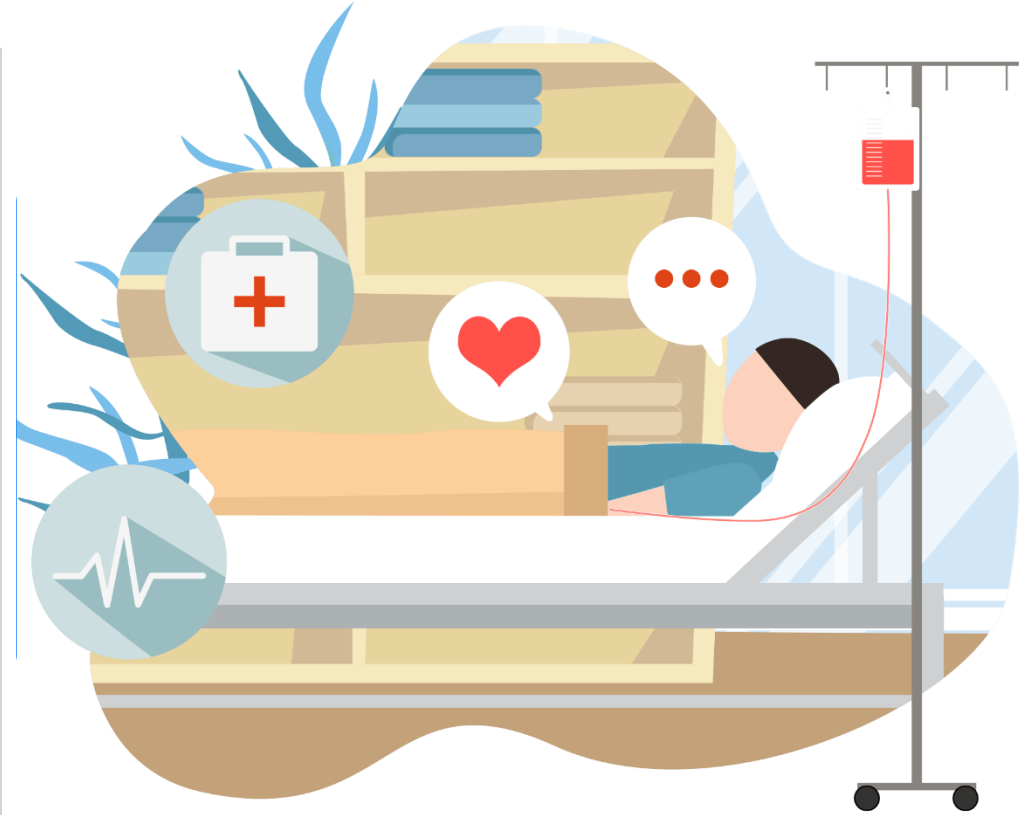
#### 4.1.7ca (Number of Major Vessels)플로로소피 0에 의해 착색된 주요 혈관(0-4)의 수

```
df.ca.value_counts(dropna=False)
```

Number of Major Vessels Distribution



주요 혈관이 0개인 사람은 다른 사람에 비해 분포가 가장 높습니다..



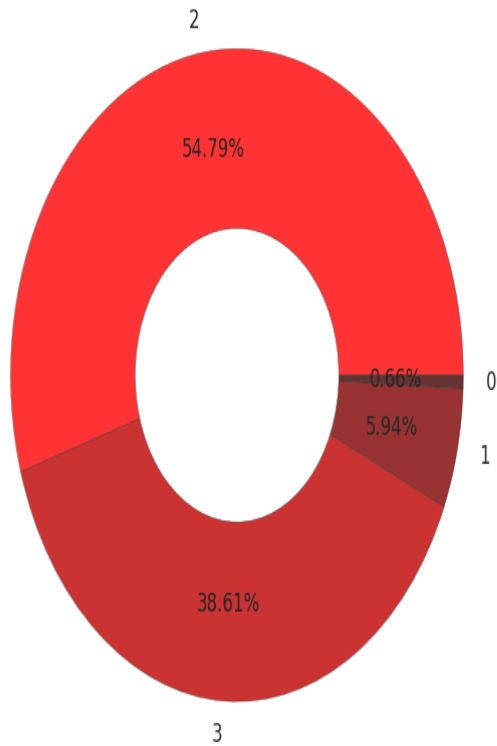


## 4.1.8thal

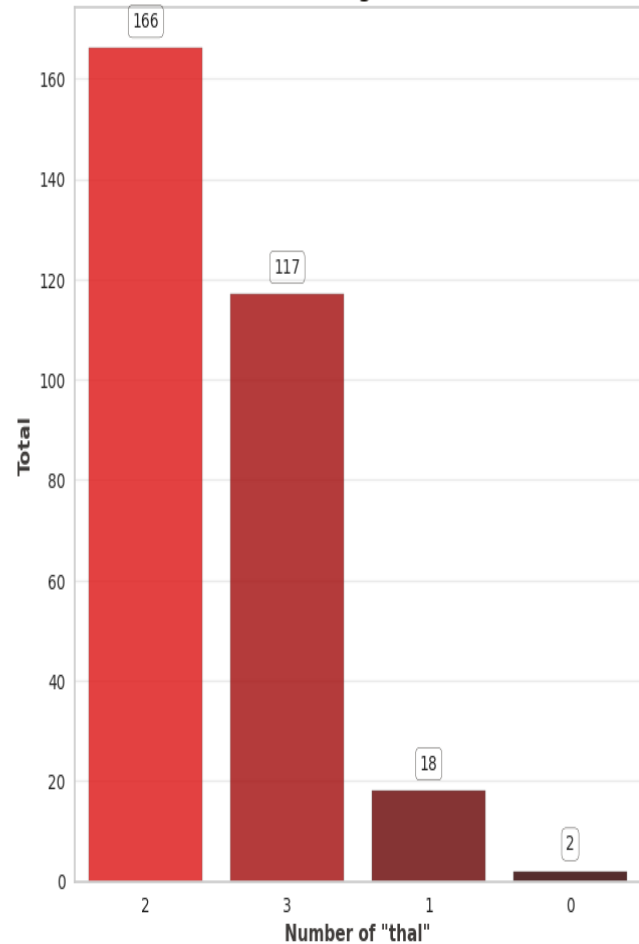
```
df.thal.value_counts(dropna=False)
```

"thal" Distribution

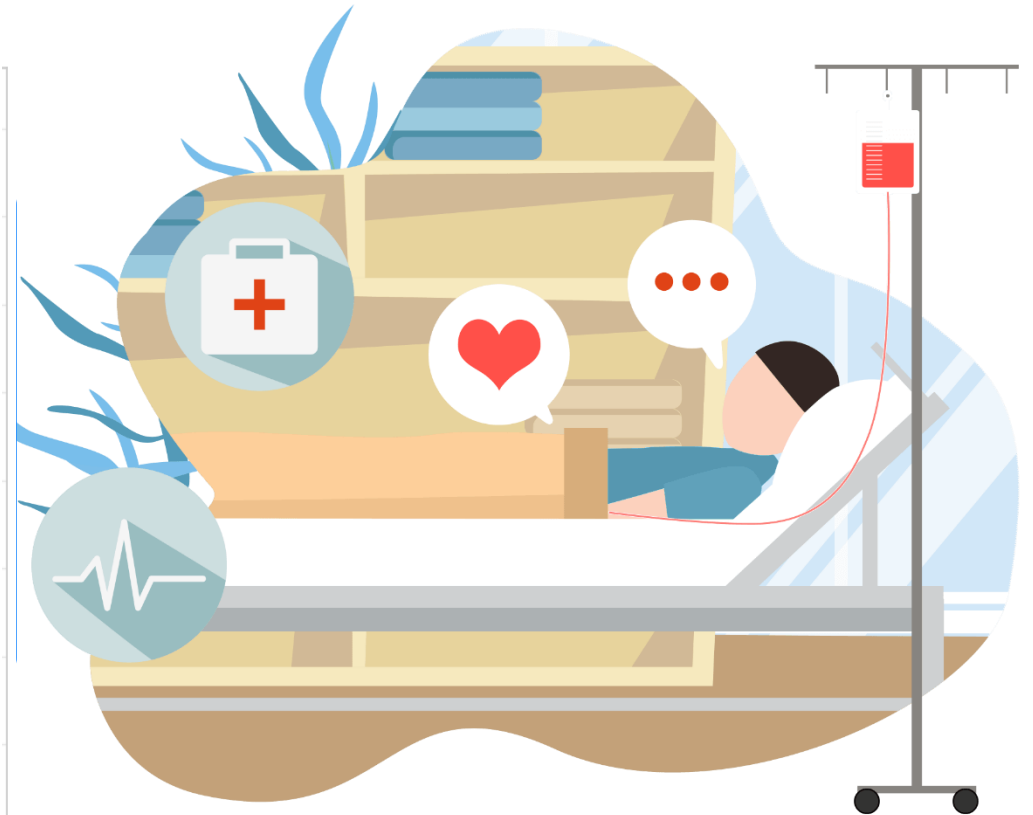
Pie Chart



Histogram



2개의 "thal"이 있는 환자는 다른 환자에 비해 분포가 가장 높습니다.



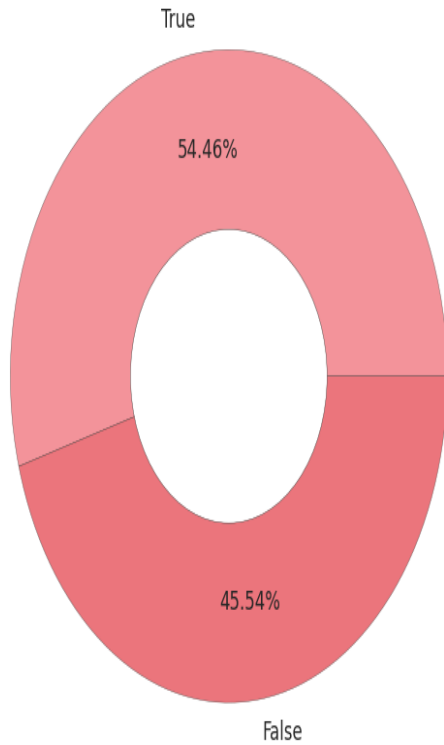


## 4.1.9target (Heart Diseases Status)

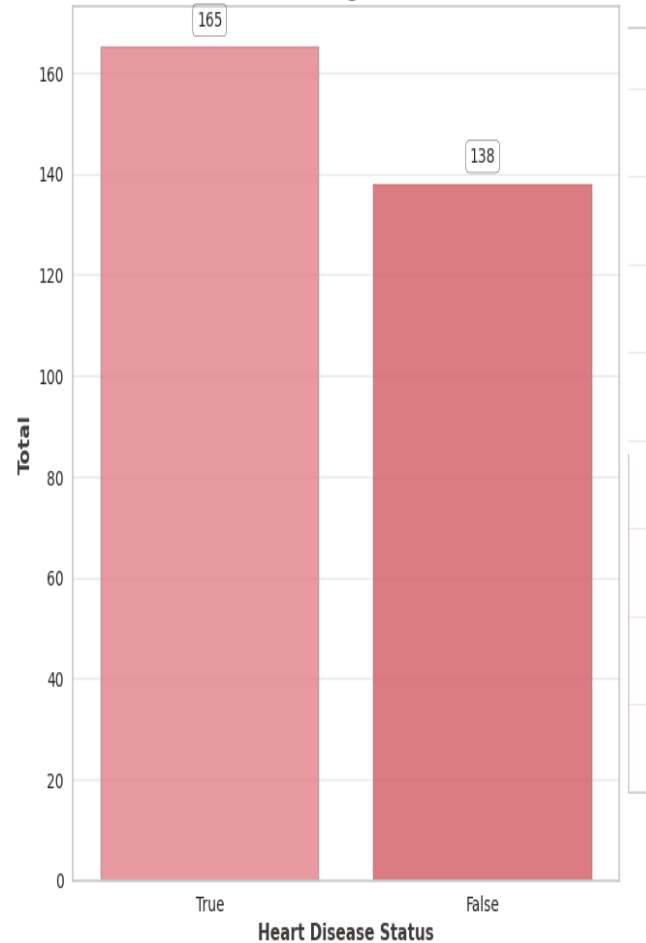
```
df.target.value_counts(dropna=False)
```

Heart Diseases Distribution

Pie Chart



Histogram



심장 질환이 없는 환자보다 심장 질환이 있는 환자의 수가 더 많습니다.



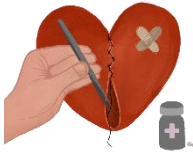




## 4.2.초기 데이터 탐색

◆ 탐색할 두 번째 유형의 변수는 연속형 변수입니다.





## 4.2.1 Descriptive Statistics

```
df.select_dtypes(exclude='object').describe().I.style.background_gradient(cmap='PuRd').set_properties(**{'font-family': 'Segoe UI'})
```

	count	mean	std	min	25%	50%	75%	max
age	303.000000	54.366337	9.082101	29.000000	47.500000	55.000000	61.000000	77.000000
trestbps	303.000000	131.623762	17.538143	94.000000	120.000000	130.000000	140.000000	200.000000
chol	303.000000	246.264026	51.830751	126.000000	211.000000	240.000000	274.500000	564.000000
thalach	303.000000	149.646865	22.905161	71.000000	133.500000	153.000000	166.000000	202.000000
oldpeak	303.000000	1.039604	1.161075	0.000000	0.000000	0.800000	1.600000	6.200000
target	303.000000	0.544554	0.498835	0.000000	0.000000	1.000000	1.000000	1.000000

기술통계를 통해 나이, 휴식혈압, 콜레스테린 및 달성된 최대 심박수는 변동이 없음을 알 수 있습니다.

✧ 표준 편차가 작다는 것은 데이터가 평균 주위에 군집화되어 있음을 의미하며, 표준 편차가 크면 데이터가 더 많이 분포되어 있음을 나타냅니다(더 많은 변동)..

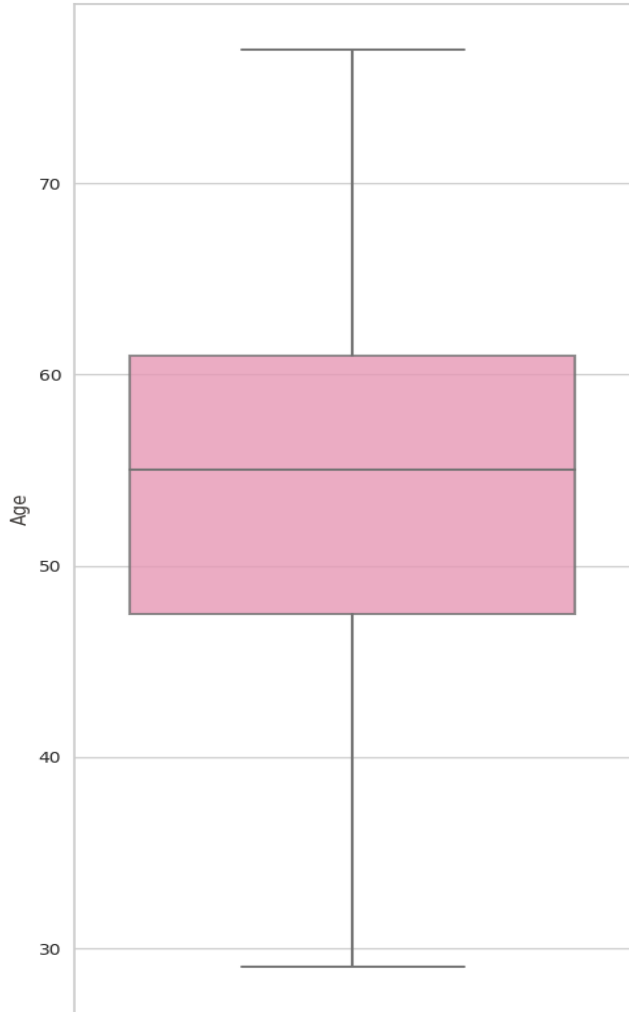




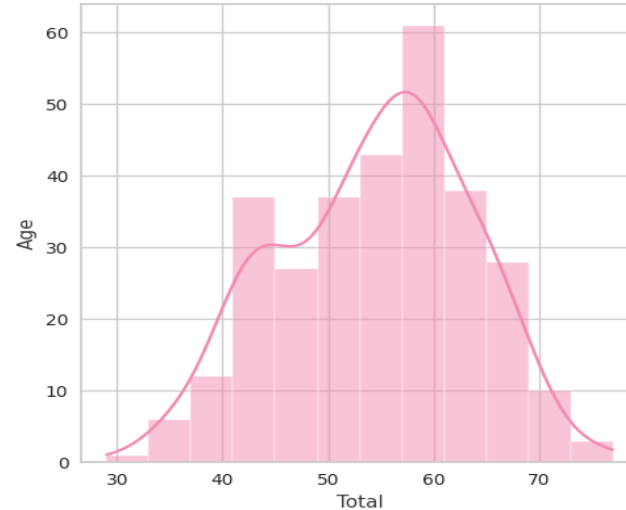
## 4.2.2age (Patient Age)나이

### Age Column Distribution

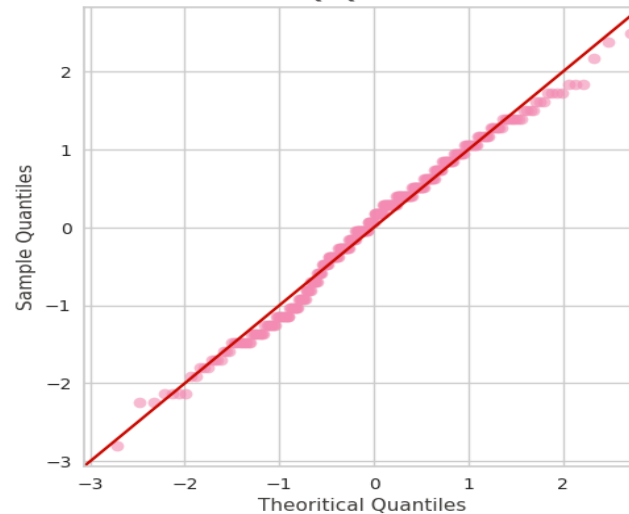
Box Plot



Histogram Plot



Q-Q Plot



```
plt.xlabel('Total', fontweight='regular', fontsize=11,
           fontfamily='sans-serif', color=black_grad[1])
plt.ylabel('Age', fontweight='regular', fontsize=11, fontfamily='sans-serif',
           color=black_grad[1])
```

```
plt.xlabel('Theoretical Quantiles', fontweight='regular', fontsize=11,
           fontfamily='sans-serif', color=black_grad[1])
plt.ylabel('Sample Quantiles', fontweight='regular', fontsize=11,
           fontfamily='sans-serif', color=black_grad[1])
```

```
plt.title('Box Plot', fontweight='bold', fontsize=14, fontfamily='sans-serif',
          color=black_grad[1])
sns.boxplot(data=df, y=var, color=color, boxprops=dict(alpha=0.8), linewidth=1.5)
plt.ylabel('Age', fontweight='regular', fontsize=11, fontfamily='sans-serif',
           color=black_grad[1])
```

히스토그램 및 상자 그림을 보면 이 열이 정규 분포를 따르고 있음을 알 수 있습니다. 이는 이 열의 왜도 값(-0.2)으로도 증명됩니다.

이 열에서 첨도 값은 -0.5이며, 이는 열이 플라티컬임을 나타냅니다.

Q-Q 그림에서 데이터 값은 45도를 따르는 경향이 있으며, 이는 데이터가 정규 분포를 따를 가능성이 높다는 것을 의미합니다(앞에서 설명한 바와 같이).

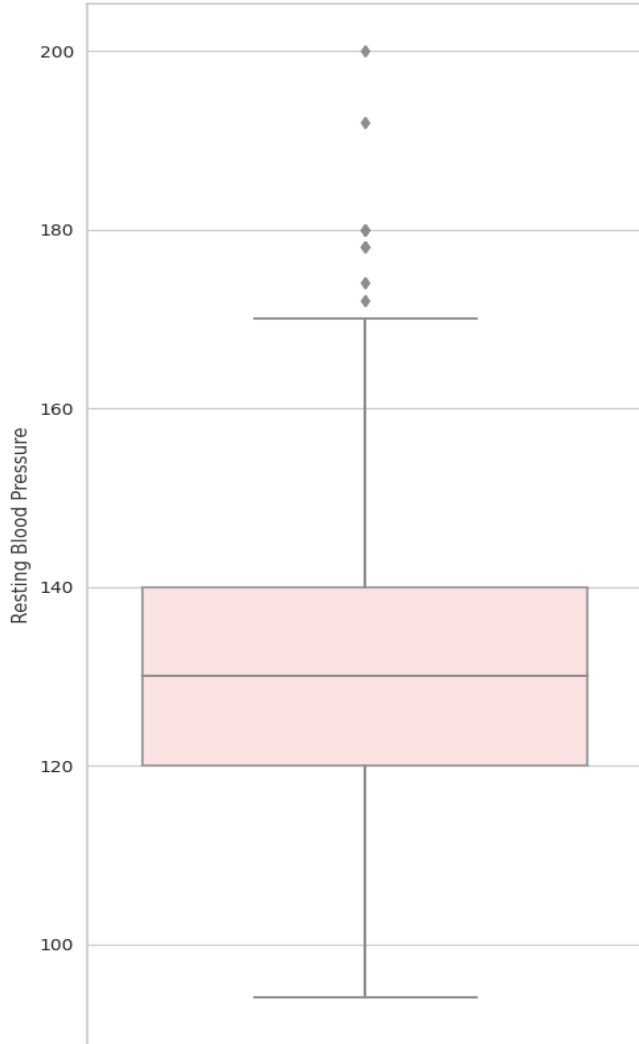
왜도가 -1보다 작거나 1보다 크면 분포가 심하게 치우쳐 있습니다. 왜도가 -1과 -0.5 사이 또는 0.5와 1 사이이면 분포가 적당히 치우쳐 있습니다. 왜도가 -0.5와 0.5 사이이면 분포가 거의 대칭입니다.



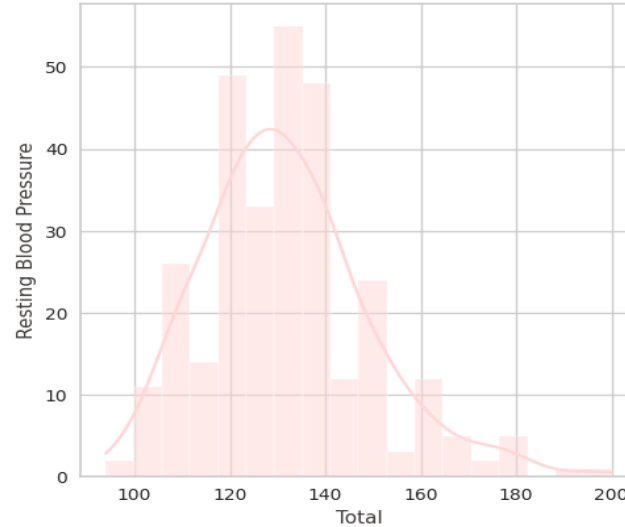
## 4.2.3 trestbps (Resting Blood Pressure in mm Hg) 안정시 혈압

Resting Blood Pressure Column Distribution

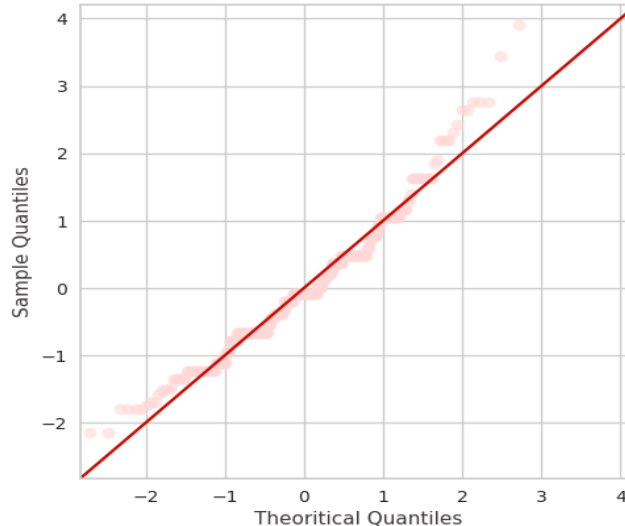
Box Plot



Histogram Plot



Q-Q Plot



```
plt.xlabel('Total', fontweight='regular', fontsize=11,
           fontfamily='sans-serif', color=black_grad[1])
plt.ylabel('Resting Blood Pressure', fontweight='regular', fontsize=11,
           fontfamily='sans-serif', color=black_grad[1])

plt.xlabel('Theoretical Quantiles', fontweight='regular', fontsize=11,
           fontfamily='sans-serif', color=black_grad[1])
plt.ylabel('Sample Quantiles', fontweight='regular', fontsize=11,
           fontfamily='sans-serif', color=black_grad[1])

plt.title('Box Plot', fontweight='bold', fontsize=14, fontfamily='sans-serif',
          color=black_grad[1])
sns.boxplot(data=df, y=var, color=color, boxprops=dict(alpha=0.8), linewidth=1.5)
plt.ylabel('Resting Blood Pressure', fontweight='regular', fontsize=11,
           fontfamily='sans-serif', color=black_grad[1])
```

히스토그램에서 이 열이 적당히 오른쪽으로 치우쳐 있음을 알 수 있습니다. 이 값은 이 열의 왜도 값(0.7)으로도 증명됩니다. 상자 그림의 위쪽에서 일부 특이치가 탐지되었습니다.

Q-Q 그림의 위쪽에서는 데이터 값이 45도에서 멀어지는 경향이 있습니다(45도 선이 있는 Q-Q 그림의 위쪽에는 간격이 있음). 즉, 데이터가 적당히 오른쪽으로 치우쳐 있습니다(앞에서 설명한 것처럼).

이 열에서 첨도 값은 0.9이며, 이는 열이 플라티컬임을 나타냅니다.

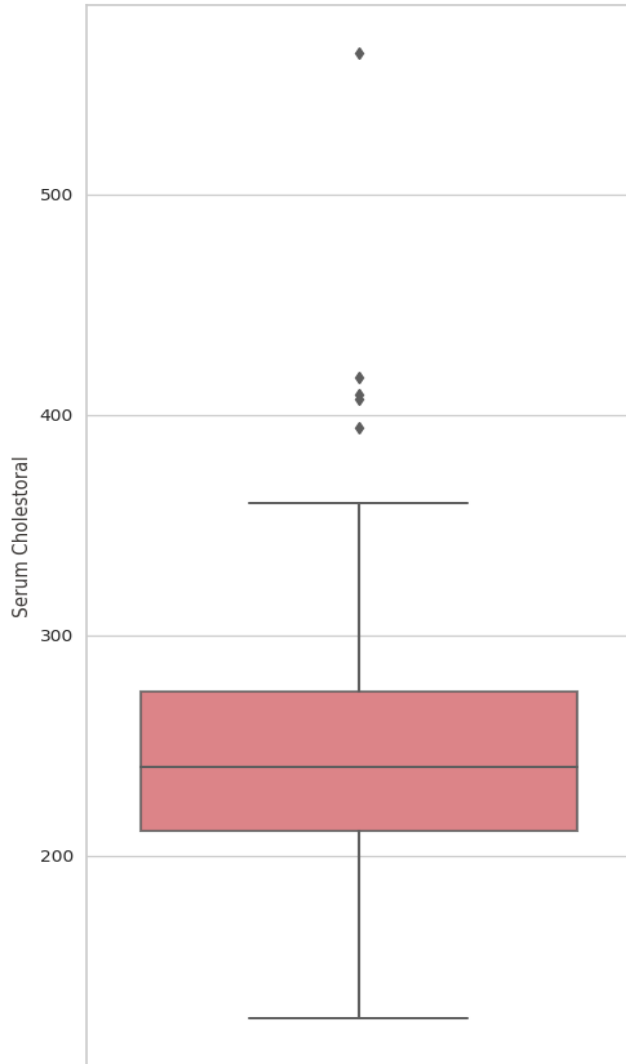




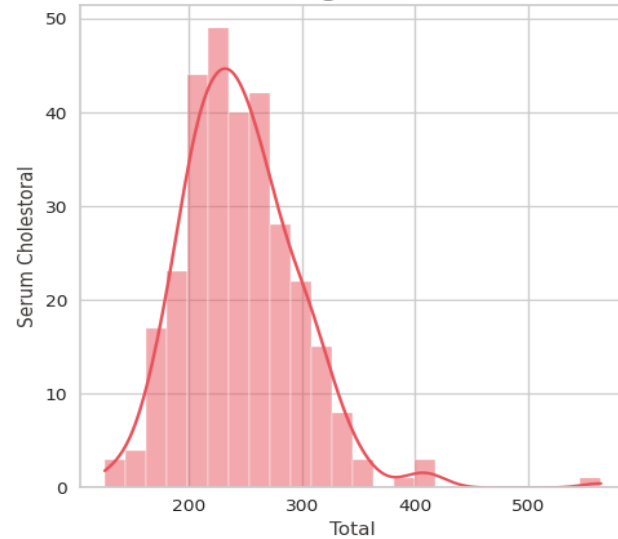
## 4.2.4chol (Serum Cholestoral in mg/dl)혈청 콜레스테롤

Serum Cholestoral Column Distribution

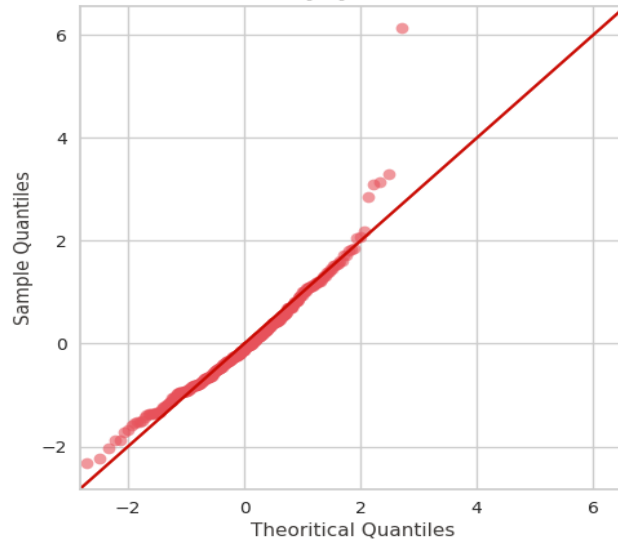
Box Plot



Histogram Plot



Q-Q Plot



```
plt.xlabel('Total', fontweight='regular', fontsize=11,
          fontfamily='sans-serif', color=black_grad[1])
plt.ylabel('Serum Cholestoral', fontweight='regular', fontsize=11,
          fontfamily='sans-serif', color=black_grad[1])
```

```
plt.title('Box Plot', fontweight='bold', fontsize=14,
         fontfamily='sans-serif', color=black_grad[1])
sns.boxplot(data=df, y=var, color=color, boxprops=dict(alpha=0.8), linewidth=1.5)
plt.ylabel('Serum Cholestoral', fontweight='regular', fontsize=11,
          fontfamily='sans-serif', color=black_grad[1])
```

```
plt.xlabel('Theoritical Quantiles', fontweight='regular', fontsize=11,
          fontfamily='sans-serif', color=black_grad[1])
plt.ylabel('Sample Quantiles', fontweight='regular', fontsize=11,
          fontfamily='sans-serif', color=black_grad[1])
```

히스토그램에서 이 열은 오른쪽으로 치우친 상태를 알 수 있습니다. 이는 이 열의 왜도 값(1.1)으로도 입증됩니다.

상자 그림의 위쪽에서 일부 특이치가 탐지되었습니다.

Q-Q 그림의 위쪽에는 45도 선이 있는 Q-Q 그림의 위쪽에 간격이 있으며, 이는 데이터가 오른쪽으로 크게 치우쳐 있음을 의미합니다(앞에서 설명한 바와 같이).

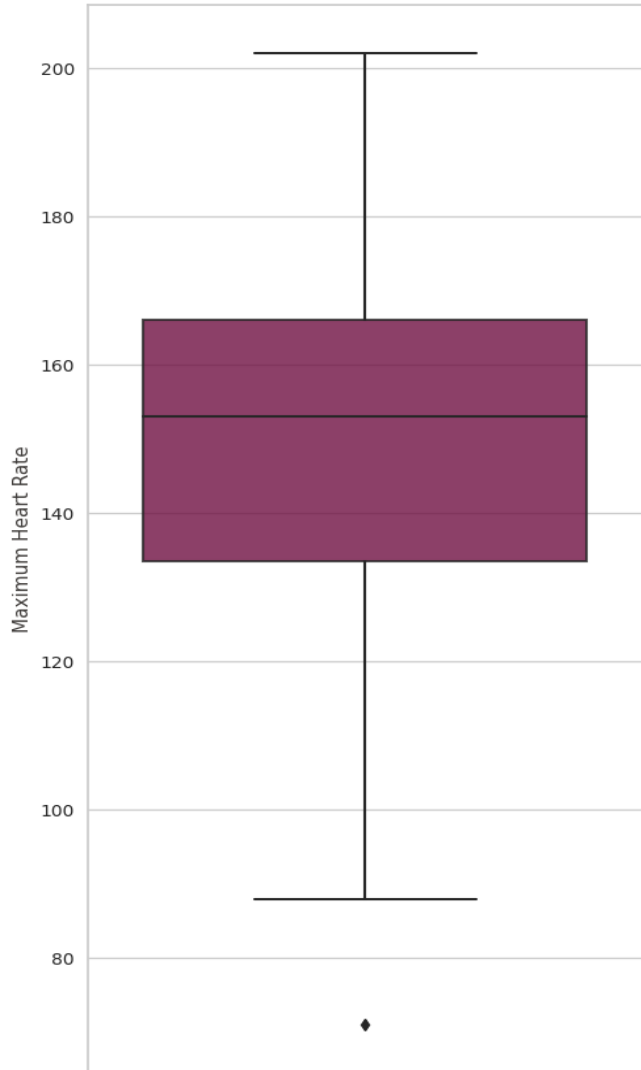
이 열의 첨도 값은 4.5이며, 이는 첨도가 레토크티임을 나타냅니다.



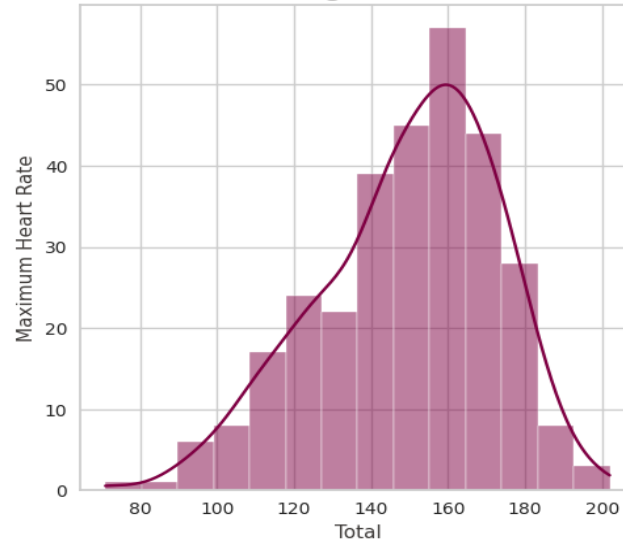
## 4.2.5thalach (Maximum Heart Rate)최대 심박수

Maximum Heart Rate Column Distribution

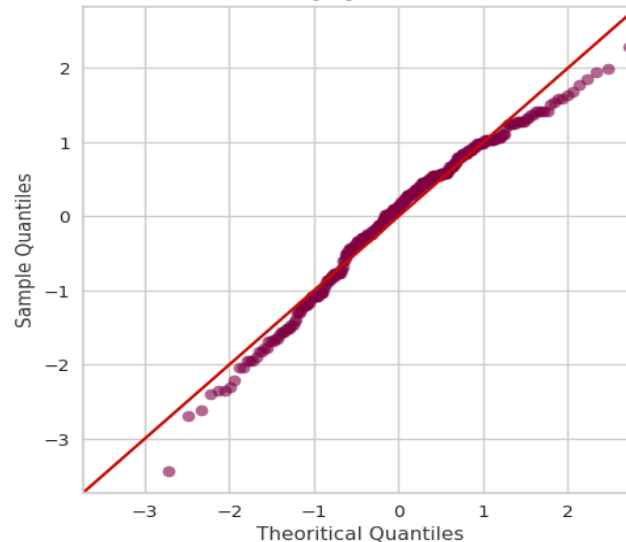
Box Plot



Histogram Plot



Q-Q Plot



```
plt.xlabel('Total', fontweight='regular', fontsize=11,
           fontfamily='sans-serif', color=black_grad[1])
plt.ylabel('Maximum Heart Rate', fontweight='regular', fontsize=11,
           fontfamily='sans-serif', color=black_grad[1])
```

```
plt.xlabel('Theoretical Quantiles', fontweight='regular', fontsize=11,
           fontfamily='sans-serif', color=black_grad[1])
plt.ylabel('Sample Quantiles', fontweight='regular', fontsize=11,
           fontfamily='sans-serif', color=black_grad[1])
```

```
plt.title('Box Plot', fontweight='bold', fontsize=14,
          fontfamily='sans-serif', color=black_grad[1])
sns.boxplot(data=df, y=var, color=color, boxprops=dict(alpha=0.8), linewidth=1.5)
plt.ylabel('Maximum Heart Rate', fontweight='regular', fontsize=11,
           fontfamily='sans-serif', color=black_grad[1])
```

히스토그램에서 이 열이 적당히 왼쪽으로 치우쳐 있음을 알 수 있습니다. 이 값은 이 열의 왜도 값(-0.5)으로도 증명됩니다.

상자 그림의 맨 아래 부분에서 특이치가 탐지되었습니다.

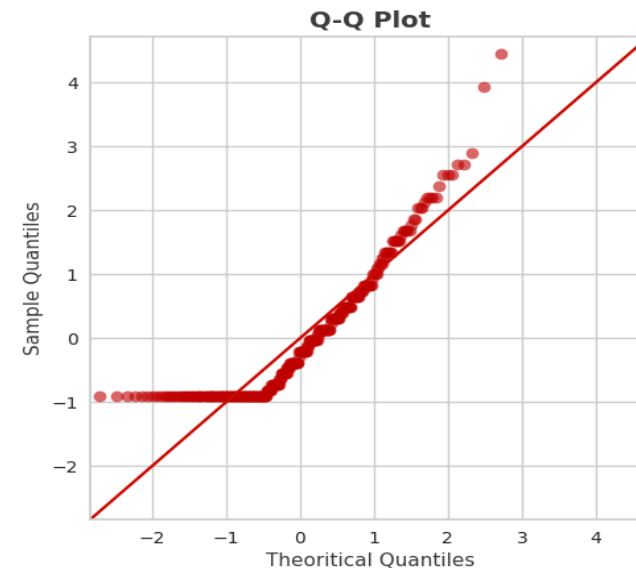
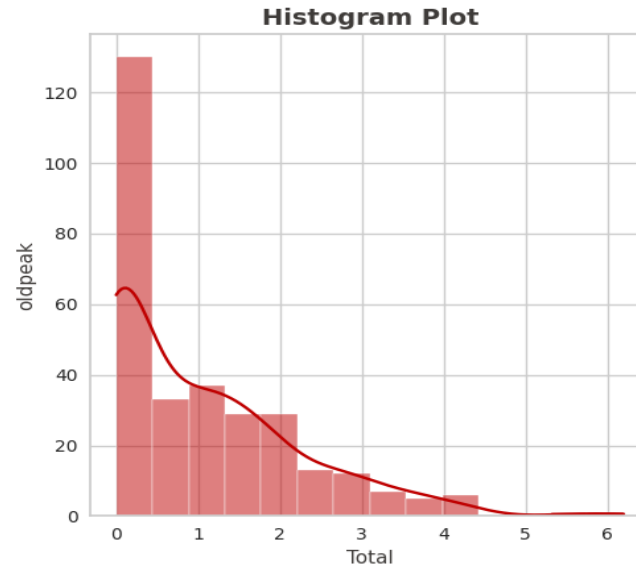
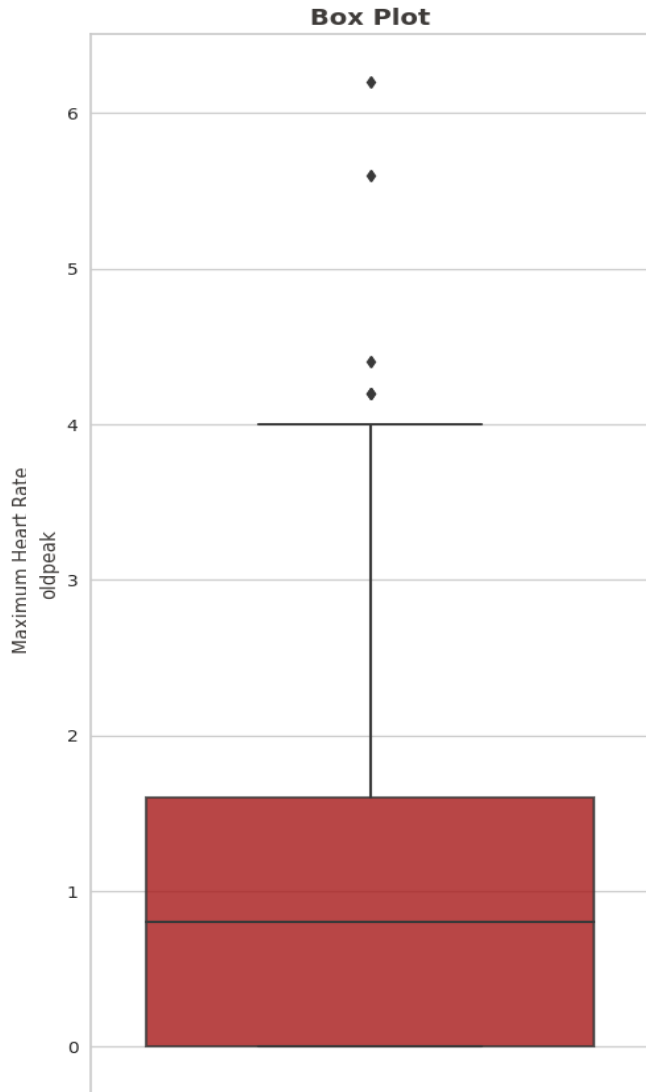
Q-Q 그림의 위쪽에는 45도 선이 있는 Q-Q 그림의 아래쪽 부분에 간격이 있으며, 이는 앞에서 설명한 대로 데이터가 적당히 왼쪽으로 치우쳐 있음을 의미합니다.

이 열에서 첨도 값은 -0.06이며, 이는 열이 플라티컬임을 나타냅니다.



## 4.2.6 oldpeak 휴식에 비해 운동으로 인한 ST 저하

"oldpeak" Column Distribution



```
plt.xlabel('Total', fontweight='regular', fontsize=11,
           fontfamily='sans-serif', color=black_grad[1])
plt.ylabel('oldpeak', fontweight='regular', fontsize=11,
           fontfamily='sans-serif', color=black_grad[1])
```

```
plt.xlabel('Theoretical Quantiles', fontweight='regular', fontsize=11,
           fontfamily='sans-serif', color=black_grad[1])
plt.ylabel('Sample Quantiles', fontweight='regular', fontsize=11,
           fontfamily='sans-serif', color=black_grad[1])
```

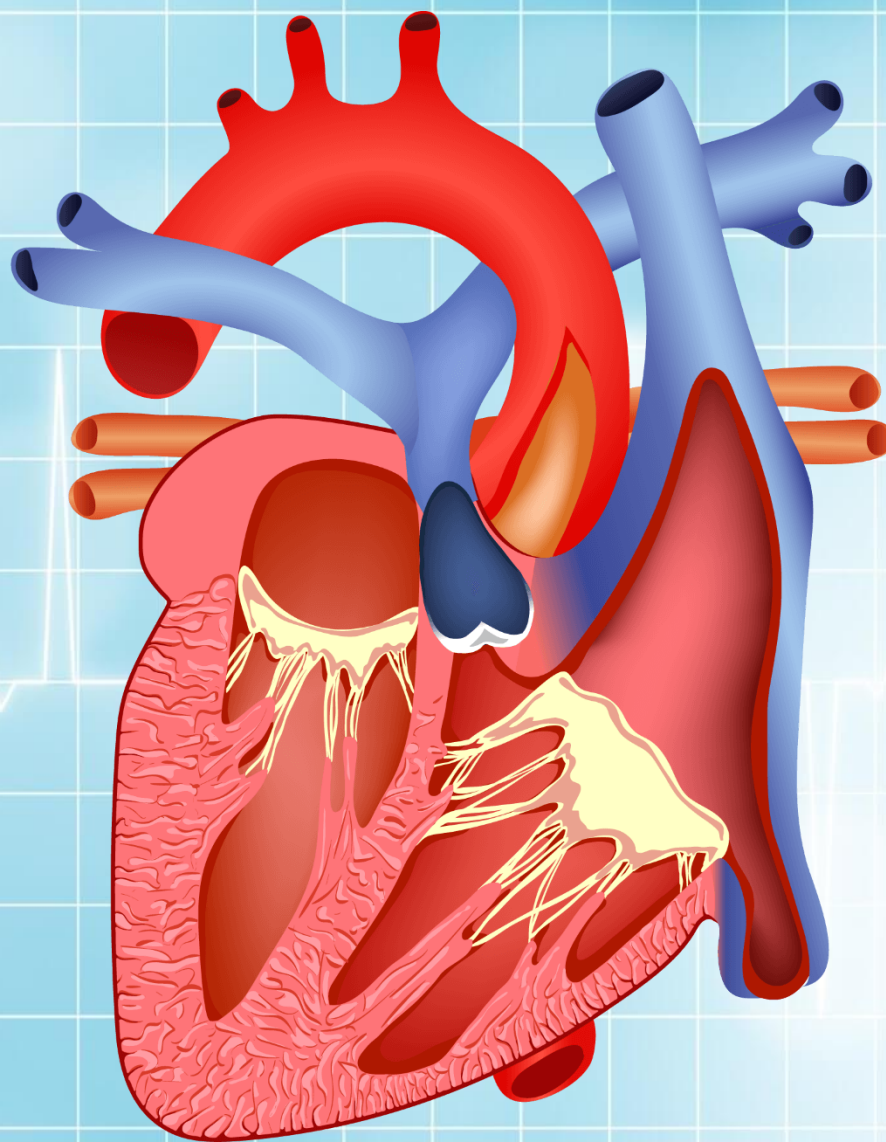
```
plt.title('Box Plot', fontweight='bold', fontsize=14,
          fontfamily='sans-serif', color=black_grad[1])
sns.boxplot(data=df, y=var, color=color, boxprops=dict(alpha=0.8), linewidth=1.5)
plt.ylabel('oldpeak', fontweight='regular', fontsize=11,
           fontfamily='sans-serif', color=black_grad[1])
```

히스토그램에서 이 열은 오른쪽으로 치우친 상태임을 알 수 있습니다. 이는 이 열의 왜도 값(1.3)으로도 입증됩니다.

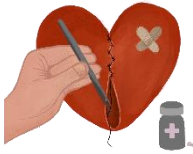
상자 그림의 위쪽에서 일부 특이치가 탐지되었습니다.

Q-Q 그림의 위쪽에는 45도 선이 있는 Q-Q 그림의 아래쪽 부분에 간격이 있으며, 이는 데이터가 오른쪽으로 크게 치우쳐 있음을 의미합니다(앞에서 설명한 바와 같이).

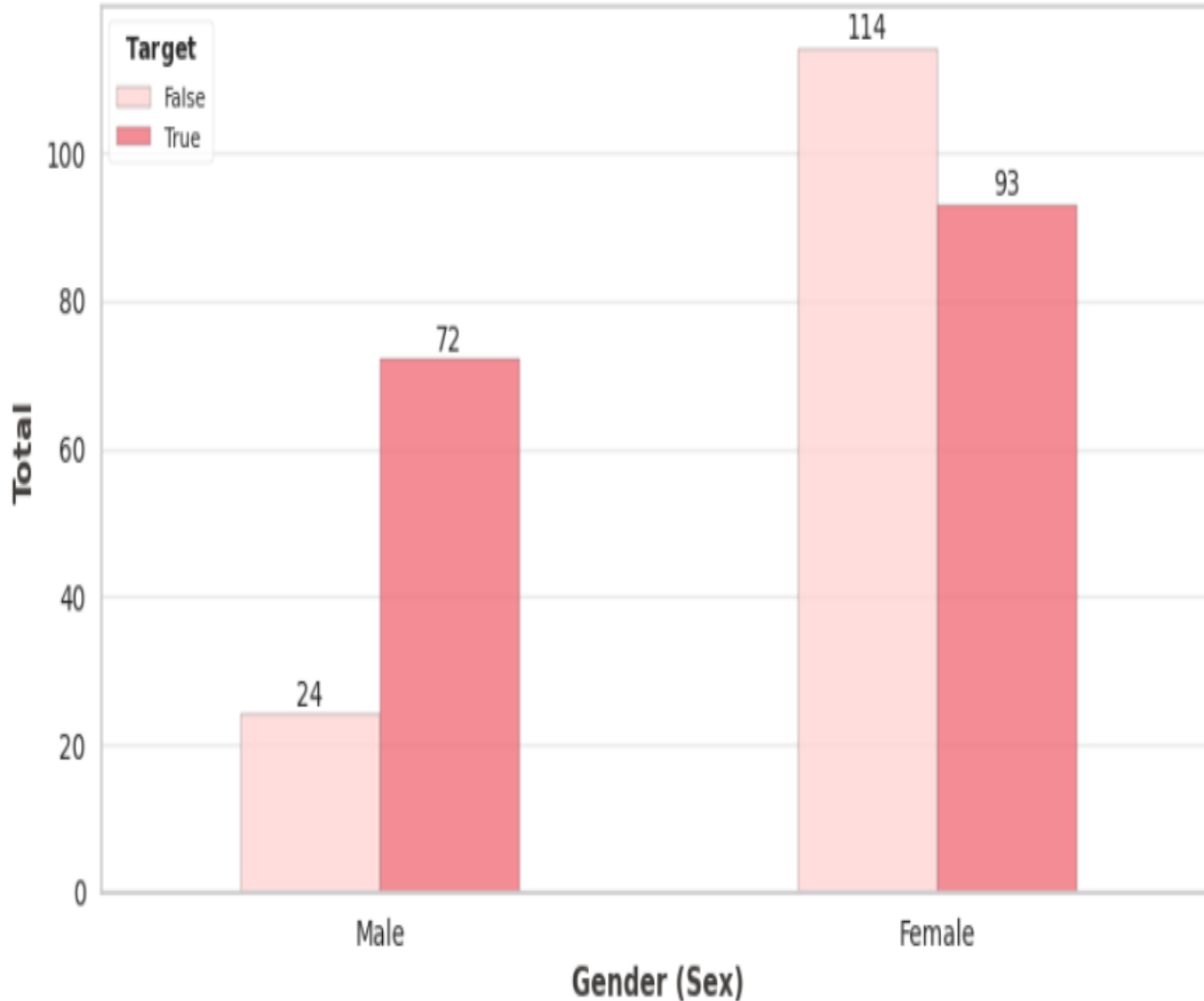
이 열에서 첨도 값은 1.57이며, 이는 열이 플라티컬임을 나타냅니다.







## 5.1 Heart Disease Distribution based on Gender성별에 따른 심장질환 분포



```
for rect in ax.patches:
    ax.text (rect.get_x()+rect.get_width()/2,
            rect.get_height()+1.25,rect.get_height(),
            horizontalalignment='center', fontsize=10)

plt.suptitle('Heart Disease Distribution based on Gender', fontweight='heavy',
            x=0.065, y=0.98, ha='left', fontsize='16', fontfamily='sans-serif',
            color=black_grad[0])

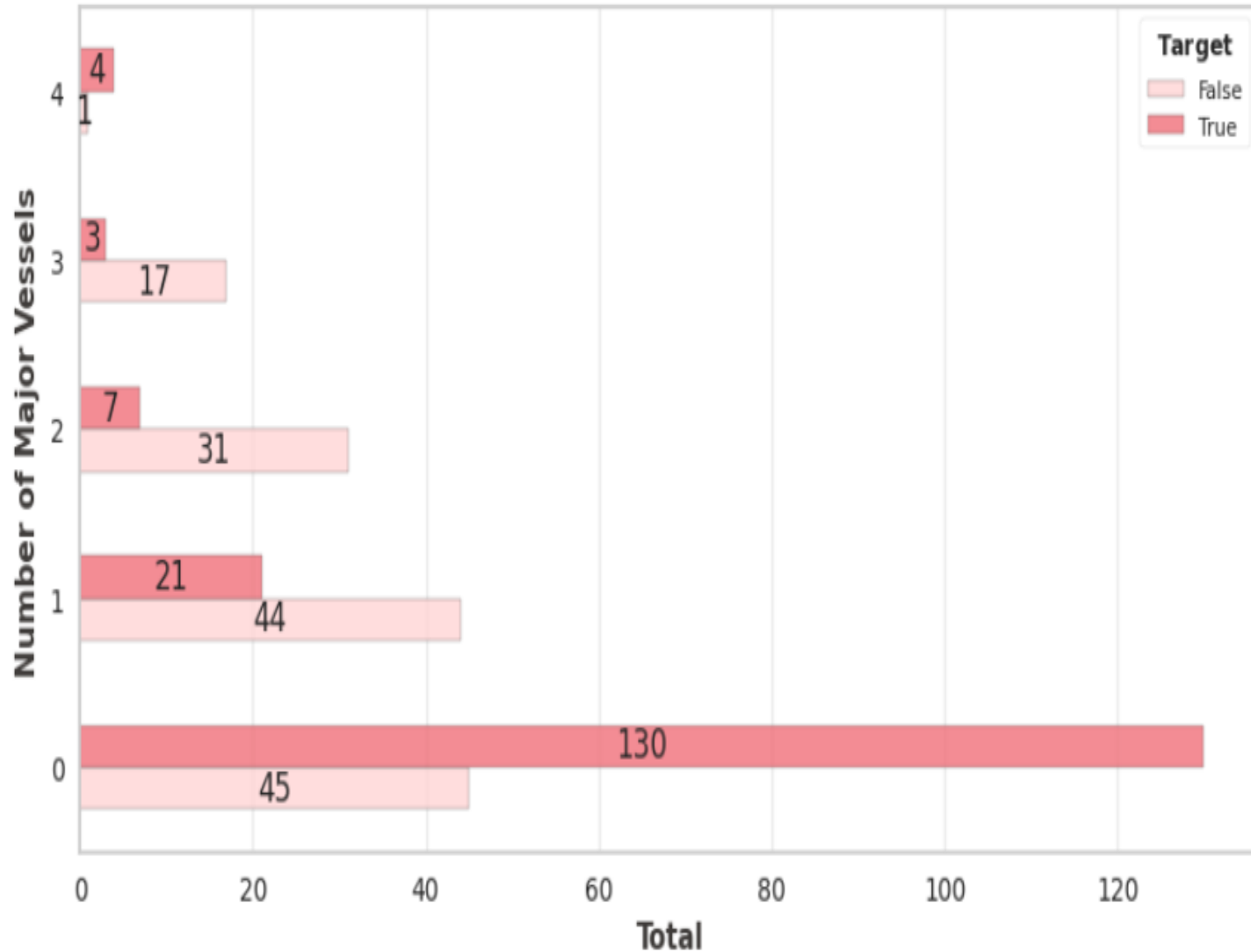
plt.title('Female tend to have heart diseases compared to Male. In male, the distribution is not imbalanced compared to female\nthat have almost the same distribution',
        fontsize='8', fontfamily='sans-serif', loc='left', color=black_grad[1])

plt.tight_layout(rect=[0, 0.04, 1, 1.025])
plt.xlabel('Gender (Sex)', fontfamily='sans-serif', fontweight='bold',
        color=black_grad[1])
plt.ylabel('Total', fontfamily='sans-serif', fontweight='bold',
        color=black_grad[1])
plt.xticks(label_gender, label_gender2, rotation=0)
plt.grid(axis='y', alpha=0.4)
plt.grid(axis='x', alpha=0)
plt.legend(labels=labels, title='$\\bf{Target}$', fontsize='8',
        title_fontsize='9', loc='upper left', frameon=True);
```

여성은 남성에 비해 심장질환이 있는 경향이 있으며, 남성의 경우 분포가 거의 같은 여성에 비해 불균형하지 않음



## 5.2 Heart Disease Distribution based on Major Vessels Total 주요혈관별 심장질환 분포 합계



```
for rect in ax.patches:
    width, height = rect.get_width(), rect.get_height()
    x, y = rect.get_xy()
    ax.text(x+width/2, y+height/2, '{:.0f}'.format(width),
            horizontalalignment='center', verticalalignment='center')

plt.suptitle('Heart Disease Distribution based on Major Vessels Total',
             fontweight='heavy', x=0.069, y=0.98, ha='left', fontsize='16',
             fontfamily='sans-serif', color=black_grad[0])

plt.title('Patients with 0 and 4 major vessels tend to have heart diseases. However, patients
who have a number of vessels 1 to 3\ntend not to have heart diseases.',
          fontsize='8', fontfamily='sans-serif', loc='left', color=black_grad[1])

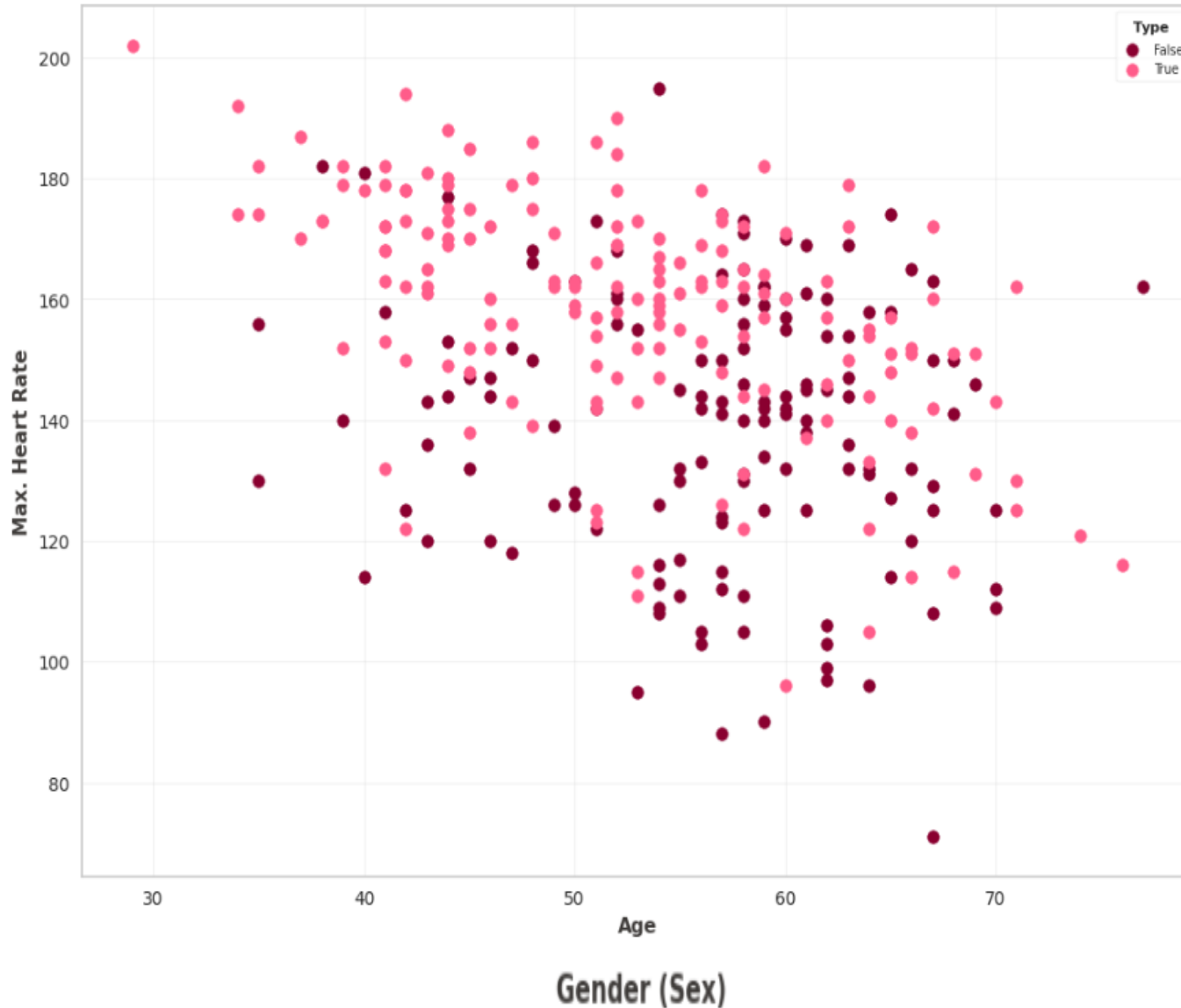
plt.tight_layout(rect=[0, 0.04, 1, 1.025])
plt.xlabel('Total', fontfamily='sans-serif', fontweight='bold', color=black_grad[1])
plt.ylabel('Number of Major Vessels', fontfamily='sans-serif', fontweight='bold',
           color=black_grad[1])

plt.yticks(rotation=0)
plt.grid(axis='x', alpha=0.4)
plt.grid(axis='y', alpha=0)
plt.legend(labels=labels, title='$\\bf{Target}$', fontsize='8', frameon=True,
           title_fontsize='9', loc='upper right');
```

대혈관 수가 0과 4개인 환자는  
심장병이 있는 경향이 있지만,  
1~3개의 주요혈관이 많은  
환자는 심장병이 없는 경향이  
있습니다.



## 5.3 Heart Disease Scatter Plot based on Age연령에 따른 심장 질환 산점도



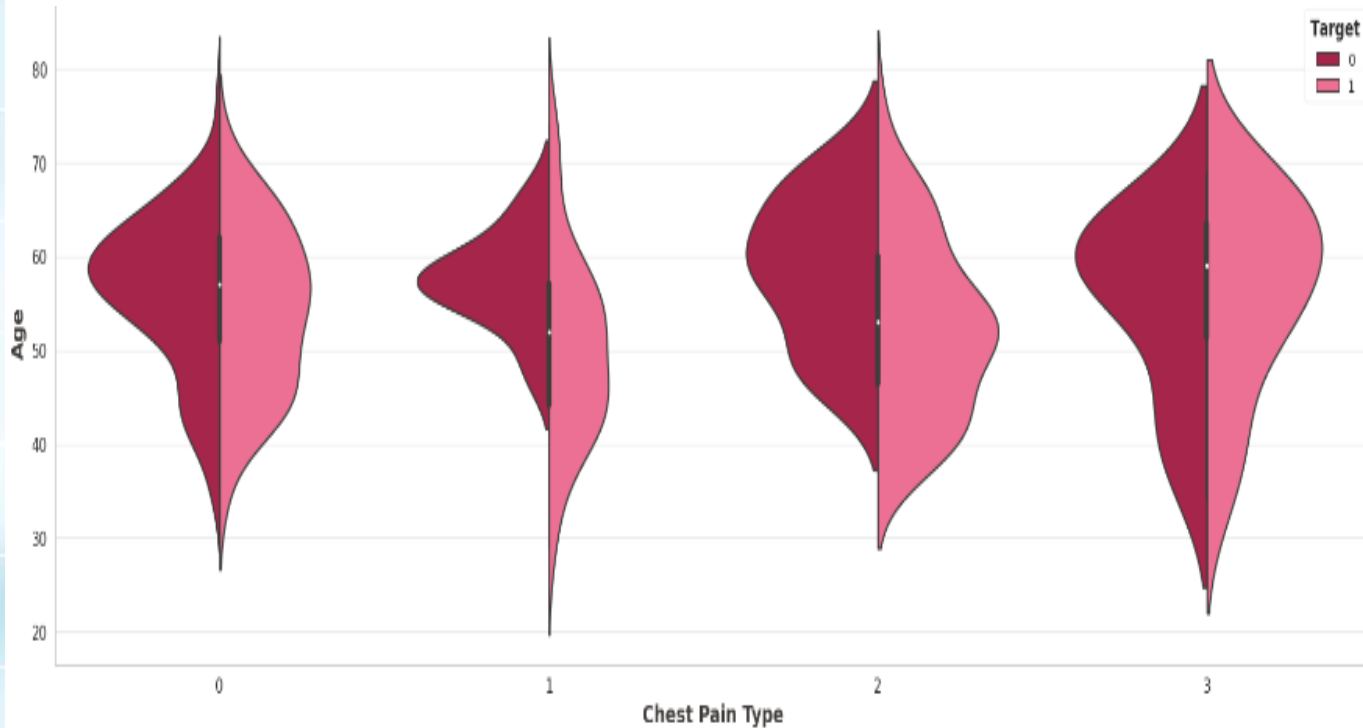
```
plt.scatter(x=df.age[df.target==0], y=df.thalach[(df.target==0)], c=pink_grad[0])
plt.scatter(x=df.age[df.target==1], y=df.thalach[(df.target==1)], c=pink_grad[2])

# --- Scatter Plot Legend & Labels Settings ---
plt.legend(['False', 'True'], title='$\\bf{Type}$', fontsize='7',
           title_fontsize='8', loc='upper right', frameon=True)
plt.xlabel('Age', fontweight='bold', fontsize='11',
           fontfamily='sans-serif', color=black_grad[1])
plt.ylabel('Max. Heart Rate', fontweight='bold', fontsize='11',
           fontfamily='sans-serif', color=black_grad[1])
plt.ticklabel_format(style='plain', axis='both')
plt.grid(axis='both', alpha=0.4, lw=0.5)
plt.show();
```

심장 질환이 있는 환자와 없는 환자의 연령을 기준으로 볼 때 대부분 50세 70세 사이의 심장 질환이 있는 환자는 심장 질환이 없는 환자에 비해 심장 박동률이 높은 경향이 있습니다.



## 5.4 Chest Pain Type based on Age나이에 따른 흉통 유형

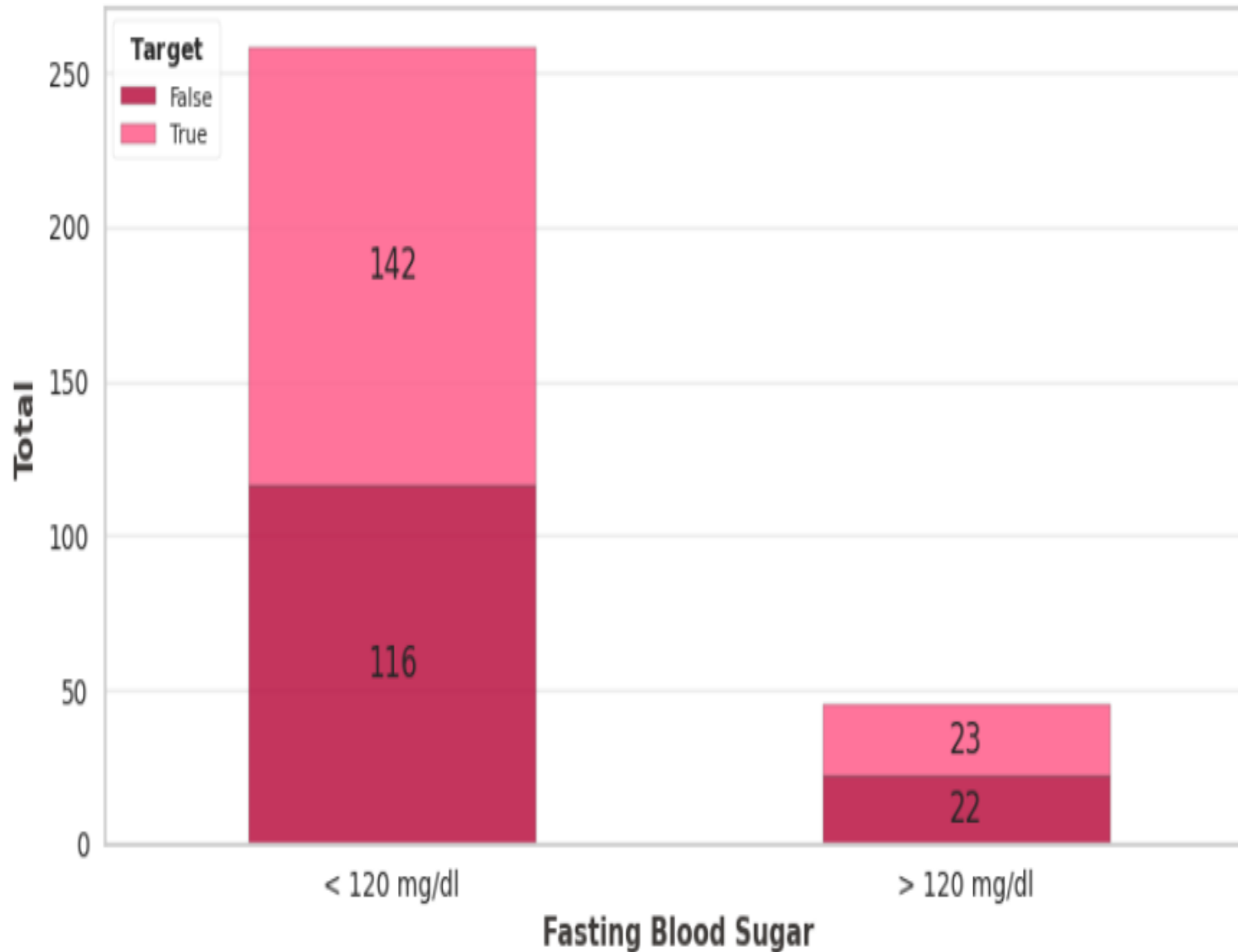


```
plt.legend(title='$\\bf{Target}$', fontsize='10', title_fontsize='12', frameon=True,
           loc='upper right')
plt.xlabel('Chest Pain Type', fontweight='bold', fontsize='14',
           fontfamily='sans-serif', color=black_grad[1])
plt.ylabel('Age', fontweight='bold', fontsize='14', fontfamily='sans-serif',
           color=black_grad[1])
plt.xticks(fontsize='11')
plt.yticks(fontsize='11')
plt.grid(axis='y', alpha=0.4)
plt.show();
```

환자 연령이 가장 낮은 것은 통증 유형 1과 3에서 확인할 수 있습니다. 추가적으로, 심장병을 가지고 있지 않은 환자들의 나이 분포는 대부분 약 60세입니다. 심장질환이 있는 환자가 심장질환이 없는 환자보다 젊다는 것도 알 수 있습니다.



## 5.5 Heart Disease Scatter Plot based on Age공복혈당에 따른 심장질환 분포



```
for rect in ax.patches:
    width, height = rect.get_width(), rect.get_height()
    x, y = rect.get_xy()
    ax.text(x+width/2, y+height/2, '{:.0f}'.format(height),
            horizontalalignment='center', verticalalignment='center')

plt.suptitle('Heart Disease Distribution based on Fasting Blood Sugar',
             fontweight='heavy', x=0.065, y=0.98, ha='left', fontsize='16',
             fontfamily='sans-serif', color=black_grad[0])
plt.title('The number of patients with low fasting blood sugar is higher compared to patients
with high fasting blood sugar. In low\fasting blood sugar, patients tend to have heart diseases.
Also, the distribution of heart diseases patients with high\fasting blood sugar is equally
distributed.',
          fontsize='8', fontfamily='sans-serif', loc='left', color=black_grad[1])
plt.tight_layout(rect=[0, 0.04, 1, 1.025])
plt.xlabel('Fasting Blood Sugar', fontfamily='sans-serif', fontweight='bold',
           color=black_grad[1])
plt.ylabel('Total', fontfamily='sans-serif', fontweight='bold',
           color=black_grad[1])
plt.xticks(label_gender, label_gender2, rotation=0)
plt.grid(axis='y', alpha=0.4)
plt.grid(axis='x', alpha=0)
plt.legend(labels=labels, title='${\\bf{Target}}$', fontsize='8',
           title_fontsize='9', loc='upper left', frameon=True);
```

낮은 공복혈당을 가진 환자들의 수는 높은 공복혈당을 가진 환자들에 비교하여 더 높습니다. 낮은 공복혈당에서, 환자들은 심장병을 가지고 있는 경향이 있습니다. 또한, 높은 공복혈당을 가진 심장병 환자들의 분포는 동등하게 분배됩니다.

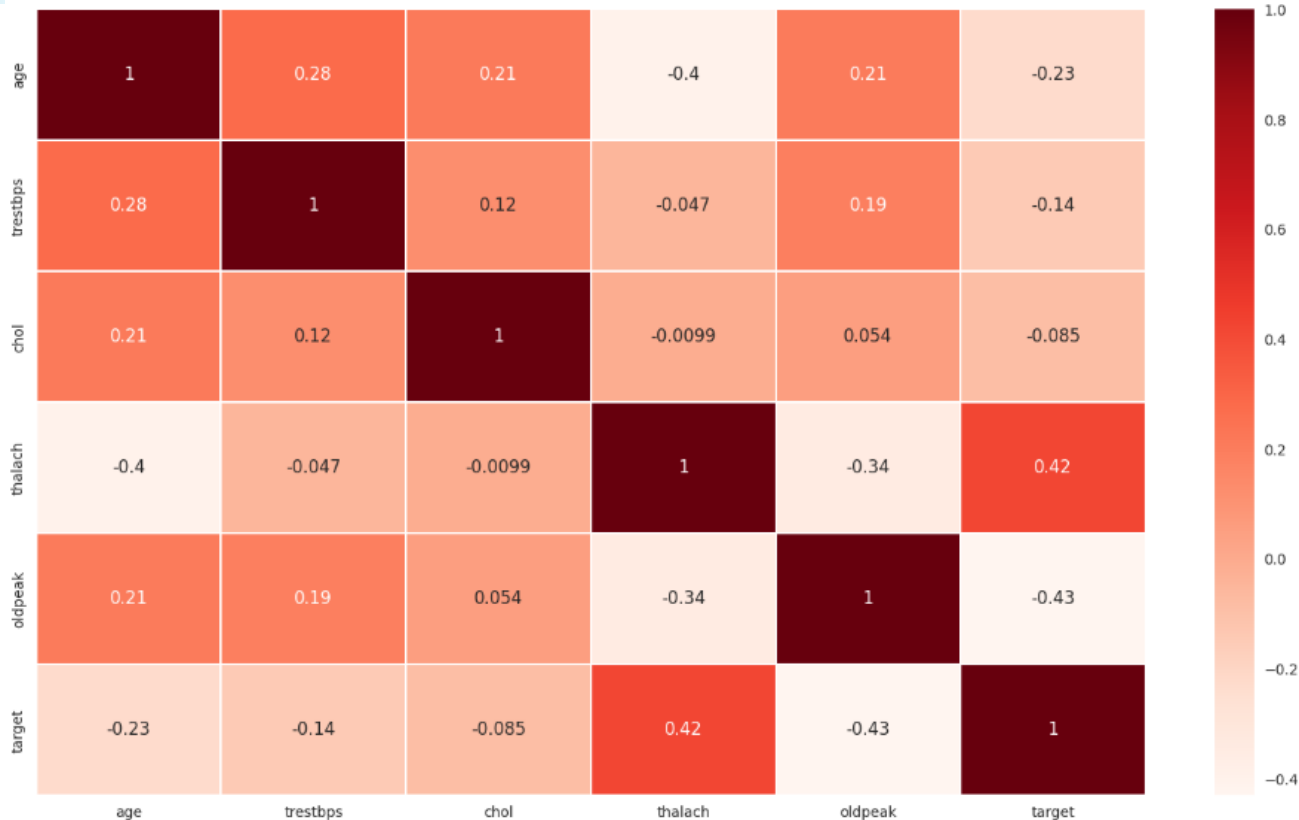




## 5.6 Heatmap

```
plt.figure(figsize=(14, 9))
sns.heatmap(df.corr(), annot=True, cmap='Reds', linewidths=0.1)
plt.suptitle('Correlation Map of Numerical Variables', fontweight='heavy',
             x=0.03, y=0.98, ha='left', fontsize='16', fontfamily='sans-serif',
             color=black_grad[0])

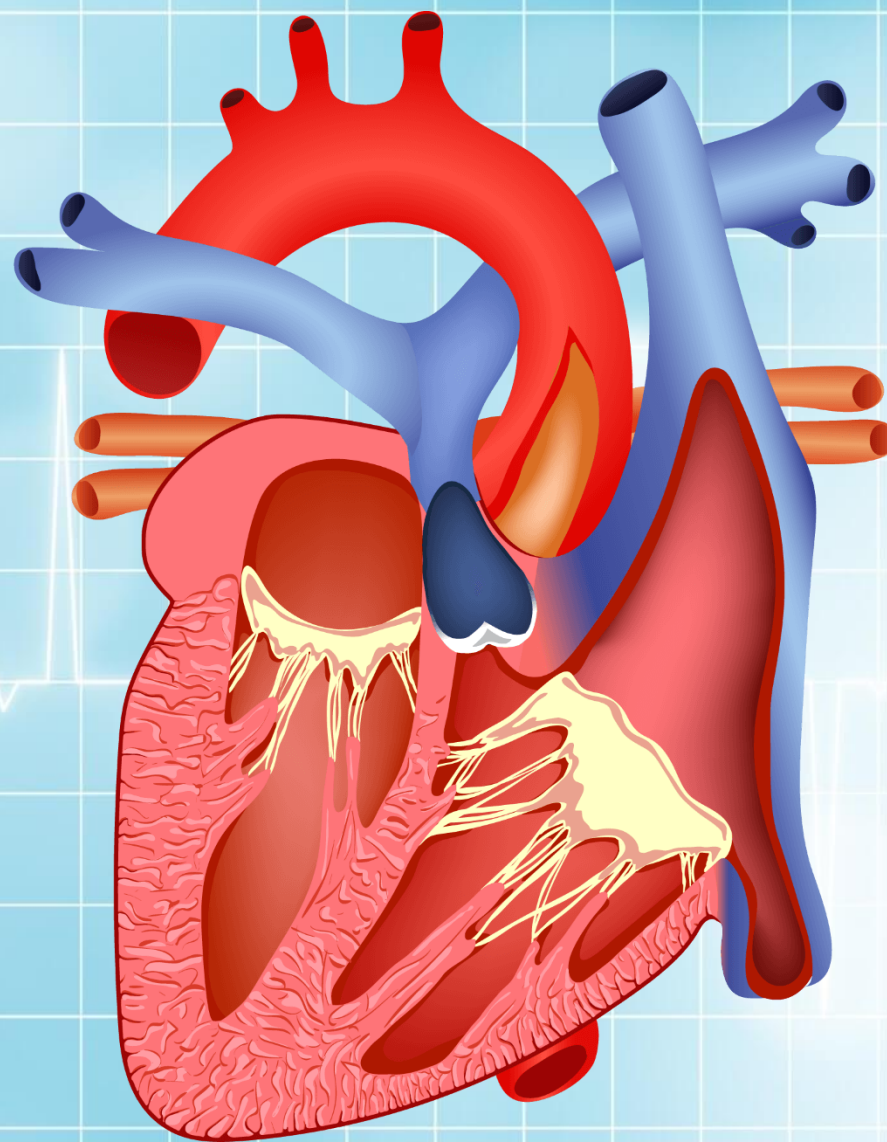
plt.title('Resting blood pressure, cholestoral, and "oldpeak" have moderate relationship with age.',
          fontsize='10', fontfamily='sans-serif', loc='left', color=black_grad[1])
plt.tight_layout(rect=[0, 0.04, 1, 1.01])
```



휴식중인 혈압, 콜레스테롤,  
그리고 "오래된 피크"는  
나와 적당한 관계를  
가지고 있습니다.



Dataset Pre-  
processing  
데이터세트 전처리





## 6.1 Dataset Pre-processing 데이터셋 전처리

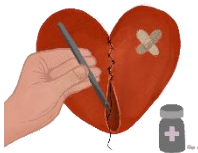
```
df.info(memory_usage = False)
```

0	age	303	non-null	int64
1	sex	303	non-null	int64
2	cp	303	non-null	int64
3	trestbps	303	non-null	int64
4	chol	303	non-null	int64
5	fbs	303	non-null	int64
6	restecg	303	non-null	int64
7	thalach	303	non-null	int64
8	exang	303	non-null	int64
9	oldpeak	303	non-null	float64
10	slope	303	non-null	int64
11	ca	303	non-null	int64
12	thal	303	non-null	int64
13	target	303	non-null	int64

```
dtypes: float64(1), int64(13)
```

13 개의 column과 303 개의  
데이터

non-null이므로 결측치는 없다.



## 6.1 Dataset Pre-processing 데이터세트 전처리

age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target	cp_0	cp_1	cp_2
63	1	3	145	233	1	0	150	0	2.300000	0	0	1	1	0	0	0
37	1	2	130	250	0	1	187	0	3.500000	0	0	2	1	0	0	1
41	0	1	130	204	0	0	172	0	1.400000	2	0	2	1	0	1	0
56	1	1	120	236	0	1	178	0	0.800000	2	0	2	1	0	1	0
57	0	0	120	354	0	1	163	1	0.600000	2	0	2	1	1	0	0

cp_3	thal_0	thal_1	thal_2	thal_3	slope_0	slope_1	slope_2
1	0	1	0	0	1	0	0
0	0	0	1	0	1	0	0
0	0	0	1	0	0	0	1
0	0	0	1	0	0	0	1
0	0	0	1	0	0	0	1

```
df.head().style.background_gradient(cmap='PuRd').hide_index().set_properties(**{'font-family':  
'Segoe UI'})
```



## 6.1 Dataset Pre-processing 데이터세트 전처리

```
df = df.drop(columns = ['cp', 'thal', 'slope'])
```

불필요한 변수는 삭제됩니다.

```
x = df.drop(['target'], axis=1)  
y = df['target']
```

target'(의존) 열은 독립 열에서 분리됩니다.





## 6.1 Dataset Pre-processing 데이터세트 전처리

```
# --- Data Normalization using Min-Max Method ---  
x = MinMaxScaler().fit_transform(x)
```

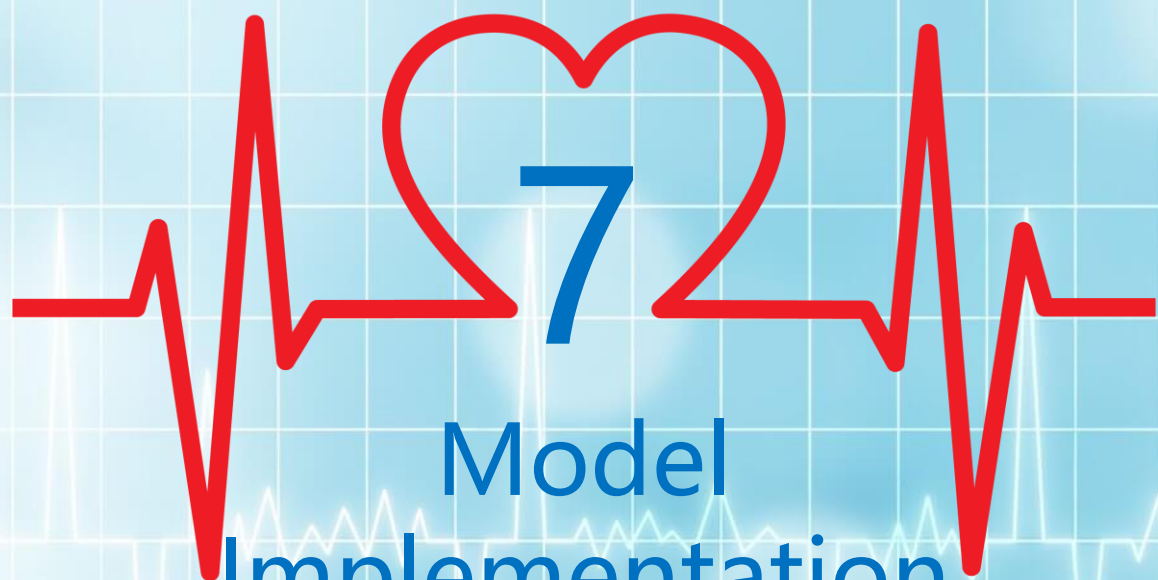
```
# --- Splitting Dataset into 80:20 ---  
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.2, random_state=4)
```

데이터 세트는 80:20 비율(80% 교육 및 20% 테스트)로 나뉩니다.

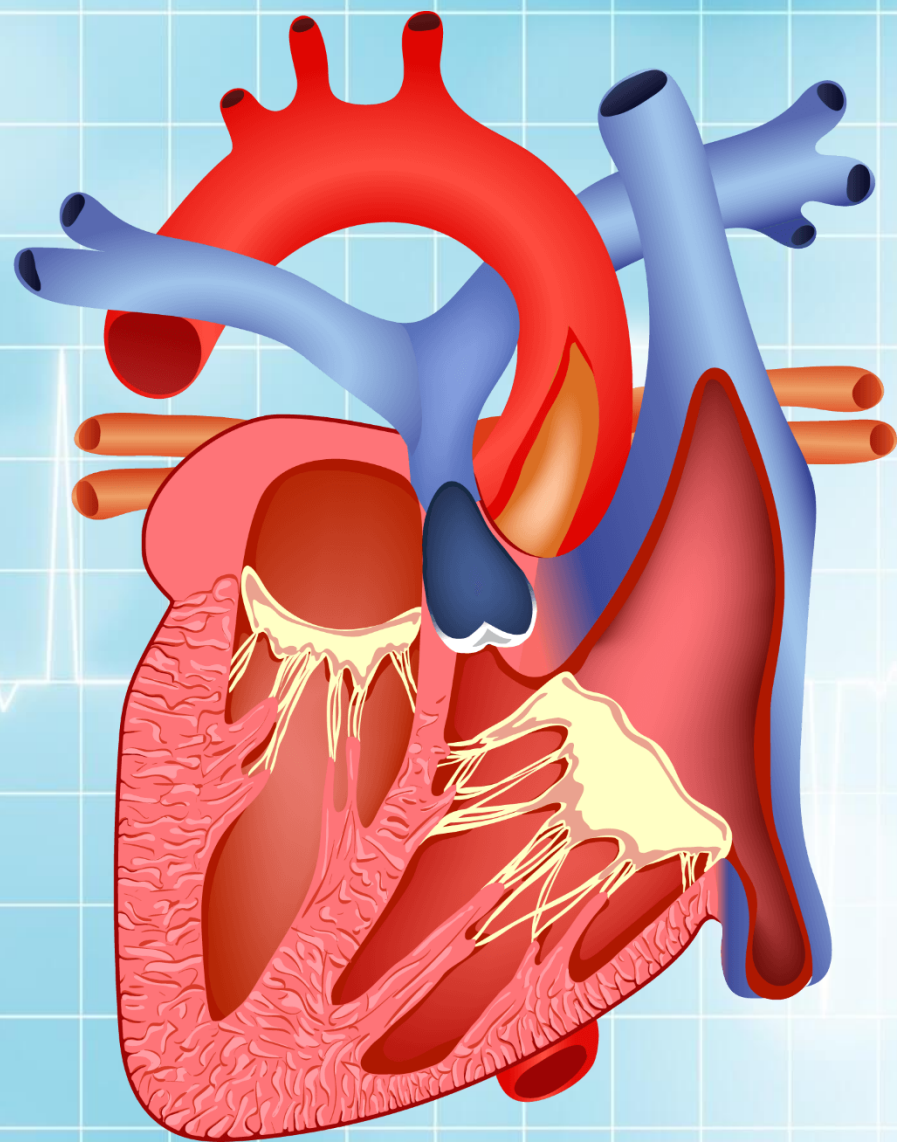
이 섹션에서는 데이터의 독립 변수 또는 특징의 범위를 정규화하기 위해 데이터 정규화를 수행합니다.

데이터 정규화는 최소-최대 정규화를 사용합니다.

최소-최대 정규화는 종종 데이터 기능의 숫자 범위 값이 0과 1 사이의 척도로 감소하는 기능 스케일링으로 알려져 있습니다.



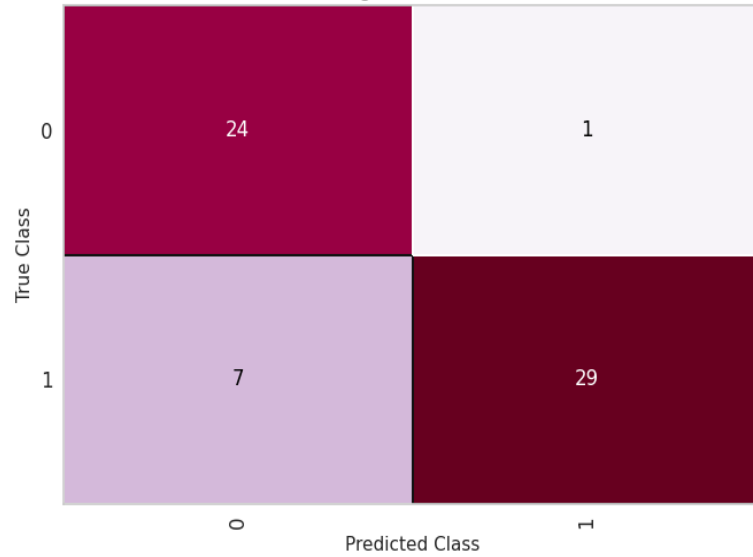
Model  
Implementation  
모델 구현



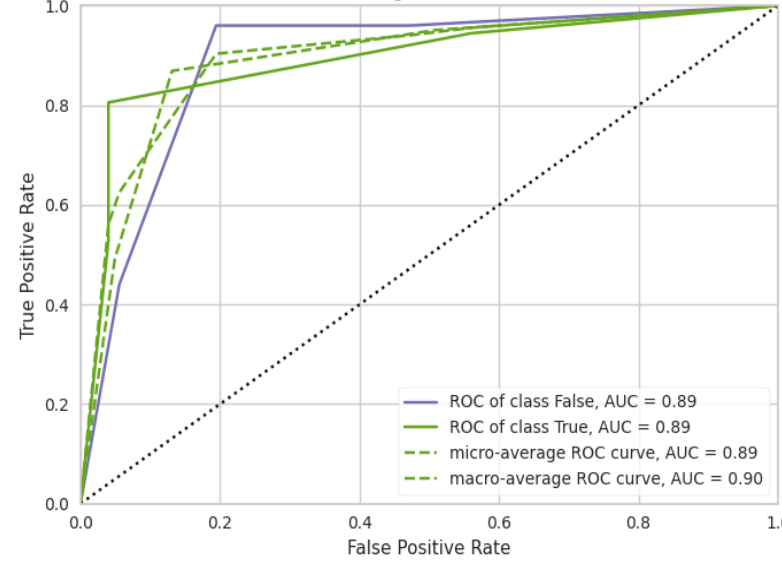


## 7. Model Implementation 모델 구현

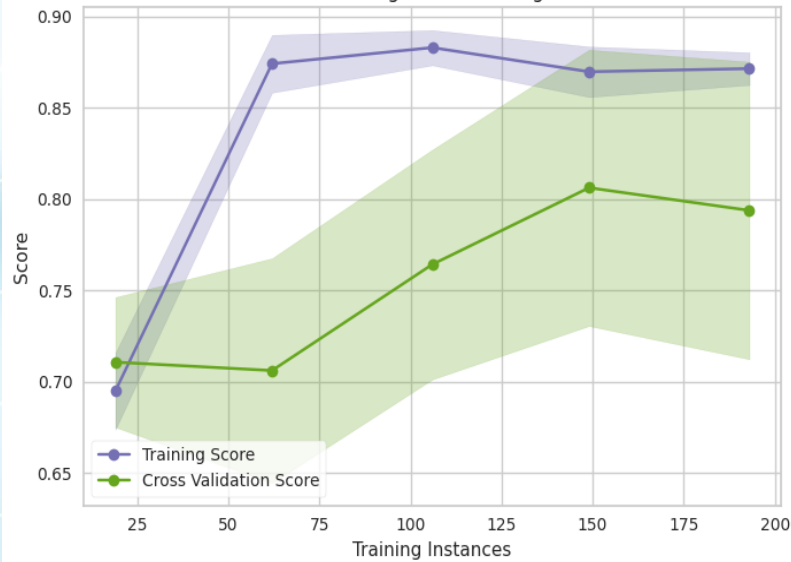
K-Nearest Neighbour Confusion Matrix



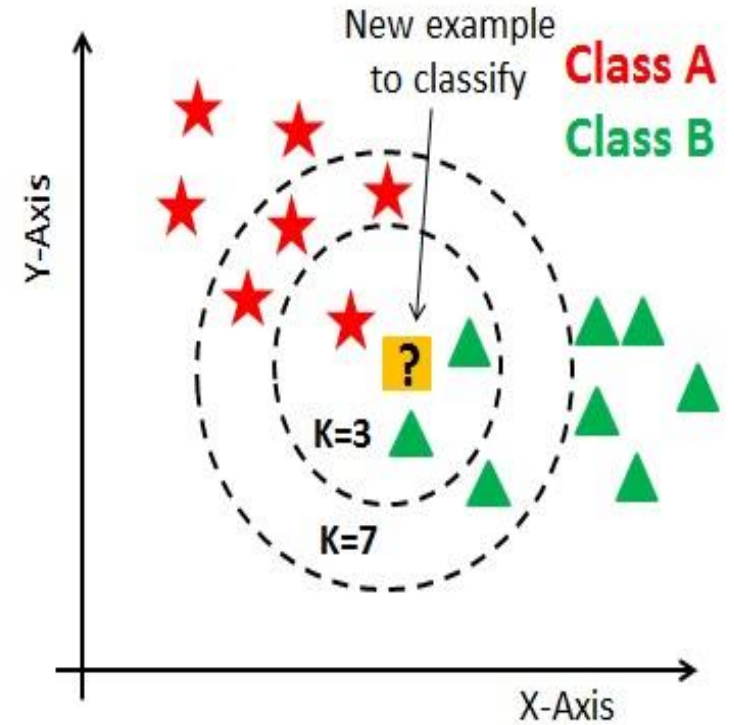
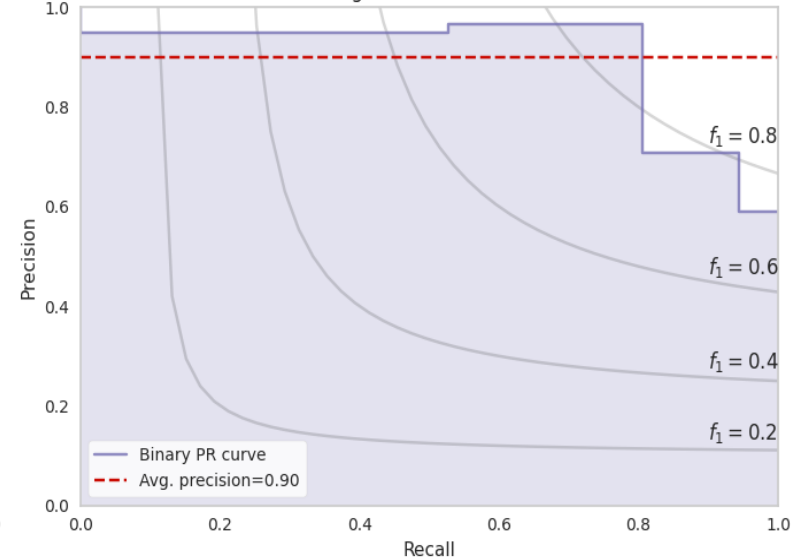
K-Nearest Neighbour ROC AUC Plot



K-Nearest Neighbour Learning Curve



K-Nearest Neighbour Precision-Recall Curve





## 7. Model Implementation 모델 구현

KNN(k-nearest neighbors) 알고리즘은 데이터 포인트가 가장 가까운 데이터 포인트가 속한 그룹을 기준으로 한 그룹 또는 다른 그룹의 멤버가 될 가능성을 추정하는 데이터 분류 방법입니다.

k-근접 이웃 알고리즘은 분류 및 회귀 문제를 해결하는 데 사용되는 감독 기계 학습 알고리즘의 한 유형입니다.

훈련 데이터를 제공할 때 어떠한 훈련도 수행하지 않기 때문에 게으른 학습자 알고리즘 또는 게으른 학습자라고 불립니다. 대신 교육 시간 동안 데이터만 저장하고 계산은 수행하지 않습니다. 데이터 세트에 대해 쿼리가 수행될 때까지 모델을 구축하지 않습니다. 따라서 KNN은 데이터 마이닝에 이상적입니다



```
# --- KNN Accuracy ---
KNNAcc = accuracy_score(y_pred_KNN, y_test)
print('... K-Nearest Neighbour Accuracy:+'\033[1m {:.2f}%'.format(KNNAcc*100)+' ...')

# --- KNN Classification Report ---
print('\n\033[1m'+': Classification Report'+'\033[0m')
print('* * 25)
print(classification_report(y_test, y_pred_KNN))

# --- Performance Evaluation ---
print('\n\033[1m'+': Performance Evaluation'+'\033[0m')
print('* * 26)
fig, ((ax1, ax2), (ax3, ax4)) = plt.subplots(2, 2, figsize=(14, 10))

# --- KNN Confusion Matrix ---
knnmatrix = ConfusionMatrix(KNNClassifier, ax=ax1, cmap='PuRd',
                             title='K-Nearest Neighbour Confusion Matrix')
knnmatrix.fit(x_train, y_train)
knnmatrix.score(x_test, y_test)
knnmatrix.finalize()

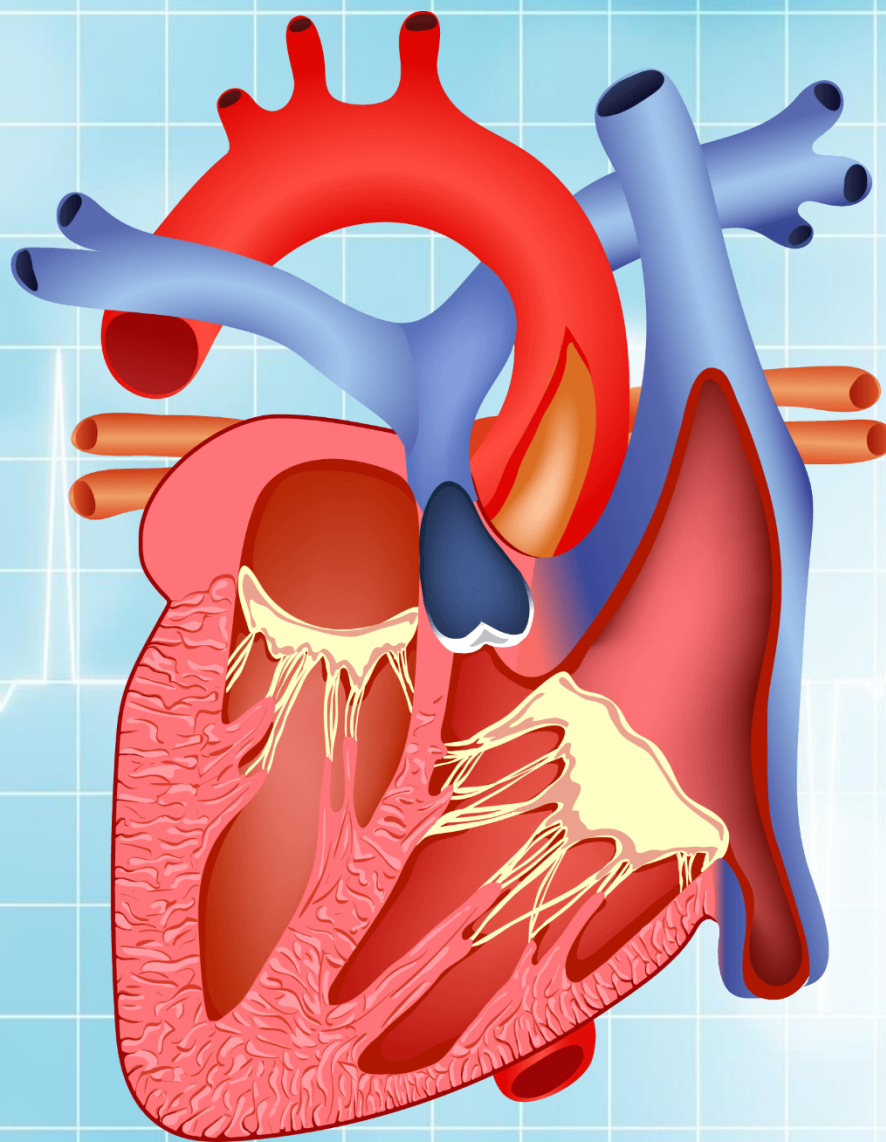
# --- KNN ROC AUC ---
knnrocauc = ROCAUC(KNNClassifier, classes=['False', 'True'], ax=ax2,
                   title='K-Nearest Neighbour ROC AUC Plot')
knnrocauc.fit(x_train, y_train)
knnrocauc.score(x_test, y_test)
knnrocauc.finalize()

# --- KNN Learning Curve ---
knnlc = LearningCurve(KNNClassifier, ax=ax3, title='K-Nearest Neighbour Learning Curve')
knnlc.fit(x_train, y_train)
knnlc.finalize()

# --- KNN Precision Recall Curve ---
knncurve = PrecisionRecallCurve(KNNClassifier, ax=ax4, ap_score=True, iso_f1_curves=True,
                                title='K-Nearest Neighbour Precision-Recall Curve')
knncurve.fit(x_train, y_train)
knncurve.score(x_test, y_test)
knncurve.finalize()

plt.tight_layout();
```









## 8. 결론

### 공복혈당

공복혈당에 따른 심장질환 분포  
공복혈당이 낮은 환자가  
공복혈당이 높은 환자보다  
많으며, 공복혈당이 낮은 환자는  
심장질환이 있는 경향이  
있습니다. .

### 주요혈관별

주요혈관별 심장질환 분포 전체  
주요혈관이 0과 4개인 환자는  
심장질환이 있는 경향이 있으나,  
1~3개의 혈관이 많은 환자는 심장질환이  
없는 경향이 있다.

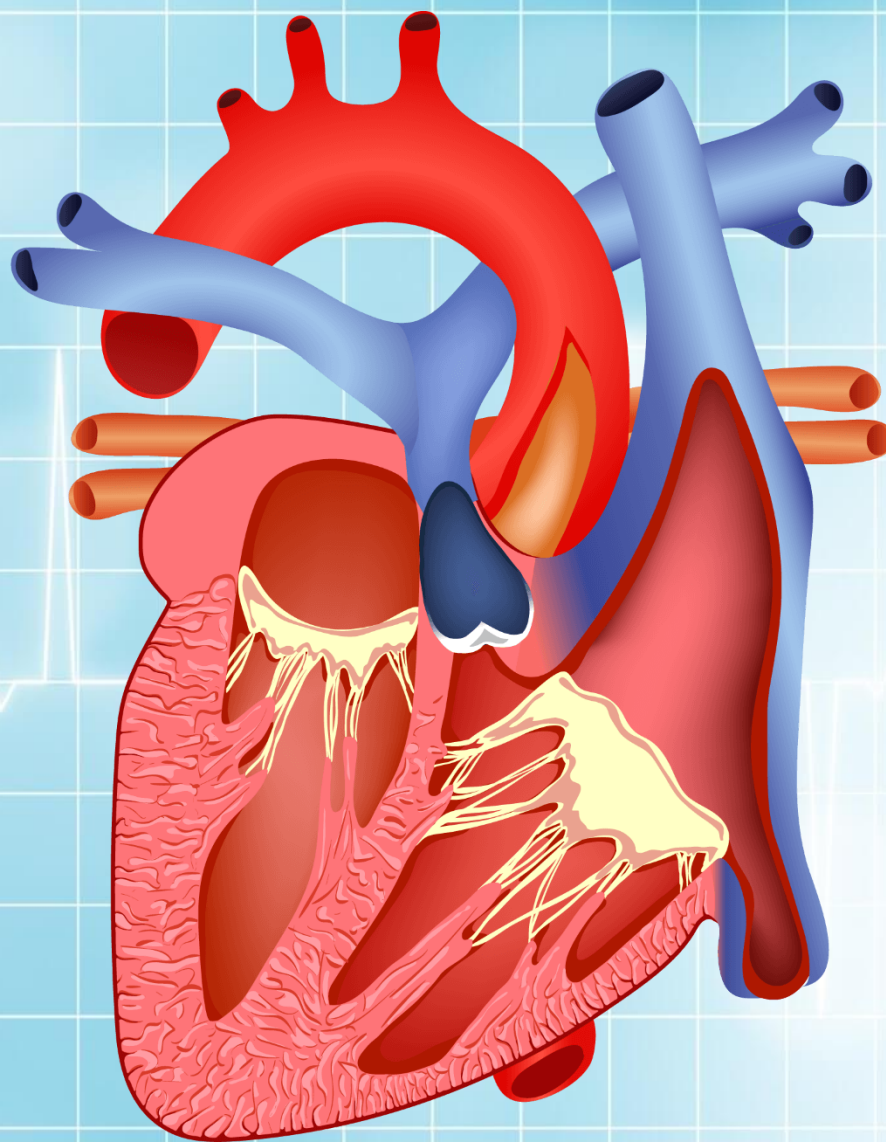


### 연령 기준

연령 기준 심장질환 유무에 관계없이  
대부분 50~70세 심장질환이 있는  
환자는 심장질환이 없는 환자에 비해  
심박수가 높은 경향이 있음



### 성별

성별에 따른 심장질환 분포  
여성은 남성에 비해 심장질환이  
있는 경향이 있습니다.





## 9. 참고 문헌.

- **Kaggle Notebook**  [What Visualizations Should You Use? by Vivek Chowdhury](#)
- [EDA On Train & Test Dataset+ !\[\]\(5daa6eee1904cb6b9d765700250de764\_img.jpg\) Price !\[\]\(d72e437c7cc5947bc0b147aba6602563\_img.jpg\) Prediction !\[\]\(0d2a89e6d0cbcd8e0459b972b9332401\_img.jpg\) by Sonali Singh](#)
- [Heart Disease - Classifications \(Machine Learning\) by Caner Dabakoglu](#)
- [Heart Disease UCI - EDA and ML w/LR by Asim Islam](#)
- [Heart Disease Predictions with Shapley by Kelli Belcher](#)
- **Online Articles**  [An Introduction to Logistic Regression in Python by Simplilearn](#)
- [What Is K-Nearest Neighbor? An ML Algorithm to Classify Data by Amal Joby](#)
- [Support Vector Machine Algorithm by Javatpoint](#)
- [Gaussian Naive Bayes by OpenGenus](#)
- [Decision Tree Classification Algorithm by Javatpoint](#)
- [Decision Tree vs. Random Forest – Which Algorithm Should you Use? by Abhishek Sharma](#)
- [Understanding Random Forest by Tony Yiu](#)
- [Gradient Boosting – What You Need to Know by Data Science . EU](#)
- [Understanding Gradient Boosting Machines by Harshdeep Singh](#)
- [AdaBoost Algorithm – A Complete Guide for Beginners by Anshul Saini](#)
- [ML | Extra Tree Classifier for Feature Selection by GeeksforGeeks](#)

이상입니다  
감사합니다.

제주대학교 컴퓨터공학전공  
등원호

