HEART DISEASE EXPLORATORY DATA ANALYSIS

Objective of the project:

1) Statistical insight of the dataset. 2) Gender distribution according to the target variable. 3) Age distribution of patients in the dataset. 4) Fasting blood sugar distribution according to the target variable. 5) Checking resting blood pressure distribution. 6) Distribution of Serum Cholesterol.

Tools used in the project:

1) NumPy 2) Pandas 3) Matplotlib 4) Seaborn

Importing the Dependencies

In [1]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
```

Importing the Dataset

In [2]:

```
data= pd.read_csv(r'C:\Users\HP\Downloads\archive (8)\heart.csv')
```

In [3]:

```
data.head()
```

Out[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

In [5]:

```
data.shape
```

Out[5]:

```
(1025, 14)
```

The given dataset has 1025 rows and 14 columns.

In [6]:

```
data.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
```

Checking for null values

In [8]:

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

Out[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
```

This dataset has no null values.

Checking for duplicate values and dropping the duplicate values (if any)

In [9]:

```
data_dup = data.duplicated().any()
print(data_dup)
```

```
True
```

In [10]:

```
data=data.drop_duplicates()
```

In [11]:

```
data.shape
```

Out[11]:

```
(302, 14)
```

The updated dataset has 302 rows and 14 coliumns.

Statistical insight of the dataset

In [12]:

```
data.describe()
```

Out[12]:

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope
count	302.00000	302.000000	302.000000	302.000000	302.000000	302.000000	302.000000	302.000000	302.000000	302.000000	302.000000
mean	54.42053	0.682119	0.963576	131.602649	246.500000	0.149007	0.526490	149.569536	0.327815	1.043046	1.397351
std	9.04797	0.466426	1.032044	17.563394	51.753489	0.356686	0.526027	22.903527	0.470196	1.161452	0.616274
min	29.00000	0.000000	0.000000	94.000000	126.000000	0.000000	0.000000	71.000000	0.000000	0.000000	0.000000
25%	48.00000	0.000000	0.000000	120.000000	211.000000	0.000000	0.000000	133.250000	0.000000	0.000000	1.000000
50%	55.50000	1.000000	1.000000	130.000000	240.500000	0.000000	1.000000	152.500000	0.000000	0.800000	1.000000
75%	61.00000	1.000000	2.000000	140.000000	274.750000	0.000000	1.000000	166.000000	1.000000	1.600000	2.000000
max	77.00000	1.000000	3.000000	200.000000	564.000000	1.000000	2.000000	202.000000	1.000000	6.200000	2.000000

Correlation matrix

In [13]:

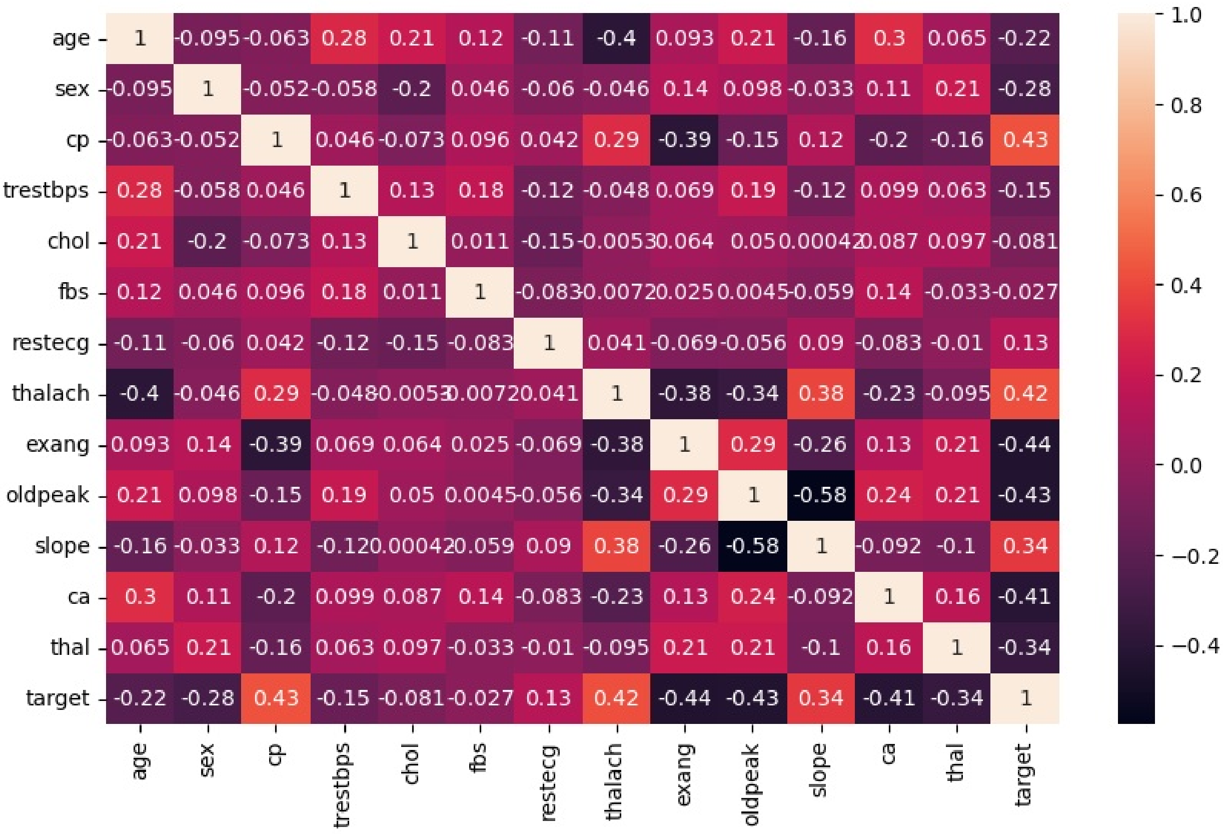
```
data.corr()
```

Out[13]:

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	
age	1.000000	-0.094962	-0.063107	0.283121	0.207216	0.119492	-0.111590	-0.395235	0.093216	0.206040	-0.164124	0.302261	0.0
sex	-0.094962	1.000000	-0.051740	-0.057647	-0.195571	0.046022	-0.060351	-0.046439	0.143460	0.098322	-0.032990	0.113060	0.2
cp	-0.063107	-0.051740	1.000000	0.046486	-0.072682	0.096018	0.041561	0.293367	-0.392937	-0.146692	0.116854	-0.195356	-0.1
trestbps	0.283121	-0.057647	0.046486	1.000000	0.125256	0.178125	-0.115367	-0.048023	0.068526	0.194600	-0.122873	0.099248	0.0
chol	0.207216	-0.195571	-0.072682	0.125256	1.000000	0.011428	-0.147602	-0.005308	0.064099	0.050086	0.000417	0.086878	0.0
fbs	0.119492	0.046022	0.096018	0.178125	0.011428	1.000000	-0.083081	-0.007169	0.024729	0.004514	-0.058654	0.144935	-0.0
restecg	-0.111590	-0.060351	0.041561	-0.115367	-0.147602	-0.083081	1.000000	0.041210	-0.068807	-0.056251	0.090402	-0.083112	-0.0
thalach	-0.395235	-0.046439	0.293367	-0.048023	-0.005308	-0.007169	0.041210	1.000000	-0.377411	-0.342201	0.384754	-0.228311	-0.0
exang	0.093216	0.143460	-0.392937	0.068526	0.064099	0.024729	-0.068807	-0.377411	1.000000	0.286766	-0.256106	0.125377	0.2
oldpeak	0.206040	0.098322	-0.146692	0.194600	0.050086	0.004514	-0.056251	-0.342201	0.286766	1.000000	-0.576314	0.236560	0.2
slope	-0.164124	-0.032990	0.116854	-0.122873	0.000417	-0.058654	0.090402	0.384754	-0.256106	-0.576314	1.000000	-0.092236	-0.1
ca	0.302261	0.113060	-0.195356	0.099248	0.086878	0.144935	-0.083112	-0.228311	0.125377	0.236560	-0.092236	1.000000	0.1
thal	0.065317	0.211452	-0.160370	0.062870	0.096810	-0.032752	-0.010473	-0.094910	0.205826	0.209090	-0.103314	0.160085	1.0
target	-0.221476	-0.283609	0.432080	-0.146269	-0.081437	-0.026826	0.134874	0.419955	-0.435601	-0.429146	0.343940	-0.408992	-0.3

In [17]:

```
plt.figure(figsize=(10,6))
sns.heatmap(data.corr(),annot=True)
plt.show()
```



	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
age	1	-0.095	-0.063	0.28	0.21	0.12	-0.11	-0.4	0.093	0.21	-0.16	0.3	0.065	-0.22
sex	-0.095	1	-0.052	-0.058	-0.2	0.046	-0.06	-0.046	0.14	0.098	-0.033	0.11	0.21	-0.28
cp	-0.063	-0.052	1	0.046	-0.073	0.096	0.042	0.29	-0.39	-0.15	0.12	-0.2	-0.16	0.43
trestbps	0.28	-0.058	0.046	1	0.13	0.18	-0.12	-0.048	0.069	0.19	-0.12	0.099	0.063	-0.15
chol	0.21	-0.2	-0.073	0.13	1	0.011	-0.15	-0.0053	0.064	0.05	0.00042	0.087	0.097	-0.081
fbs	0.12	0.046	0.096	0.18	0.011	1	-0.083	-0.0072	0.025	0.0045	-0.059	0.14	-0.033	-0.027
restecg	-0.11	-0.06	0.042	-0.12	-0.15	-0.083	1	0.041	-0.069	-0.056	0.09	-0.083	-0.01	0.13
thalach	-0.4	-0.046	0.29	-0.048	-0.0053	-0.0072	0.041	1	-0.38	-0.34	0.38	-0.23	-0.095	0.42
exang	0.093	0.14	-0.39	0.069	0.064	0.025	-0.069	-0.38	1	0.29	-0.26	0.13	0.21	-0.44
oldpeak	0.21	0.098	-0.15	0.19	0.05	0.0045	-0.056	-0.34	0.29	1	-0.58	0.24	0.21	-0.43
slope	-0.16	-0.033	0.12	-0.12	0.00042	-0.059	0.09	0.38	-0.26	-0.58	1	-0.092	-0.1	0.34
ca	0.3	0.11	-0.2	0.099	0.087	0.14	-0.083	-0.23	0.13	0.24	-0.092	1	0.16	-0.41
thal	0.065	0.21	-0.16	0.063	0.097	-0.033	-0.01	-0.095	0.21	0.21	-0.1	0.16	1	-0.34
target	-0.22	-0.28	0.43	-0.15	-0.081	-0.027	0.13	0.42	-0.44	-0.43	0.34	-0.41	-0.34	1

The number of people having heart disease, number of people not having heart disease

In [19]:

```
data.columns
```

Out[19]:

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

In [20]:

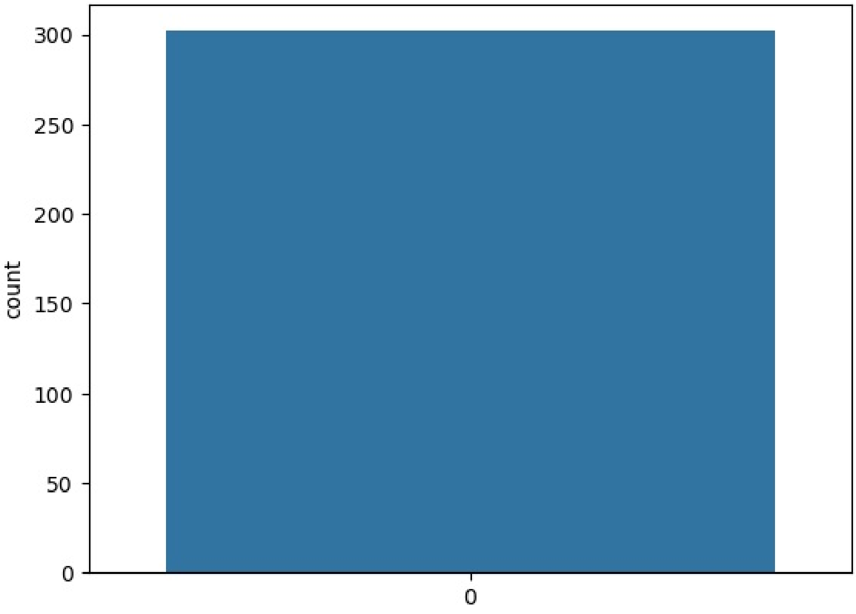
```
data['target'].value_counts()
```

Out[20]:

```
1    164
0    138
Name: target, dtype: int64
```

In [47]:

```
sns.countplot(data['target'])
plt.show()
```



Gender

In [23]:
```
data.columns
```

Out[23]:
```
Index(['age', 'sex', 'cp', 'trestbps', 'chol', 'fbs', 'restecg', 'thalach',
       'exang', 'oldpeak', 'slope', 'ca', 'thal', 'target'],
      dtype='object')
```

In [24]:
```
data['sex'].value_counts()
```

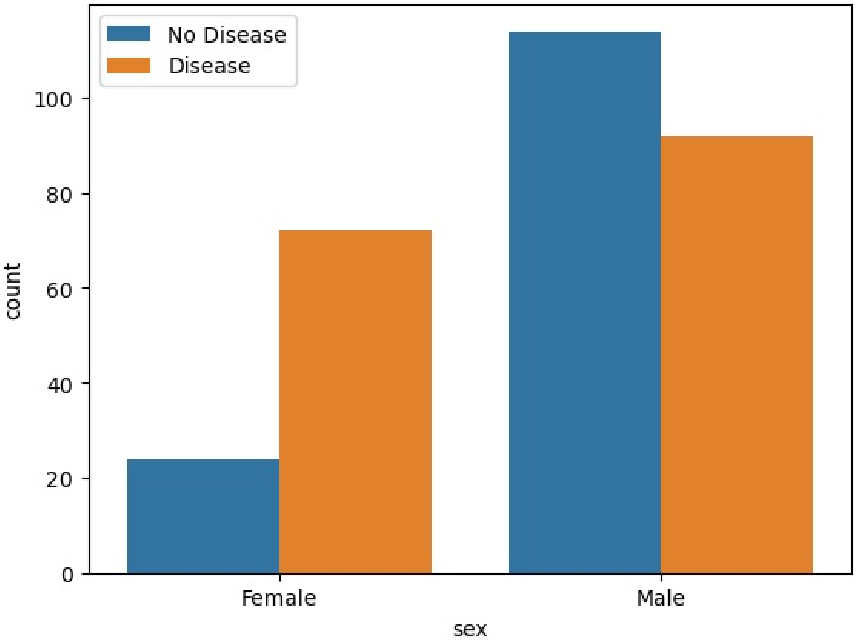
Out[24]:
```
1    206
0     96
Name: sex, dtype: int64
```

Gender distribution according to the target variable

In [32]:
```
data.columns
```

Out[32]:
```
Index(['age', 'sex', 'cp', 'trestbps', 'chol', 'fbs', 'restecg', 'thalach',
       'exang', 'oldpeak', 'slope', 'ca', 'thal', 'target'],
      dtype='object')
```

In [37]:
```
sns.countplot(x='sex',hue='target',data=data)
plt.xticks([1,0],['Male','Female'])
plt.legend(labels=['No Disease','Disease'])
plt.show()
```



There are more male patients who are suffering from heart disease than female patients. Apart from this the number of healthy male people is greater than those of male patients suffering from heart disease. The number of healthy women are lesser than the number of healthy male .

Age distributiuon in the dataset

```
In [40]: sns.distplot(data['age'])
         plt.show()
```

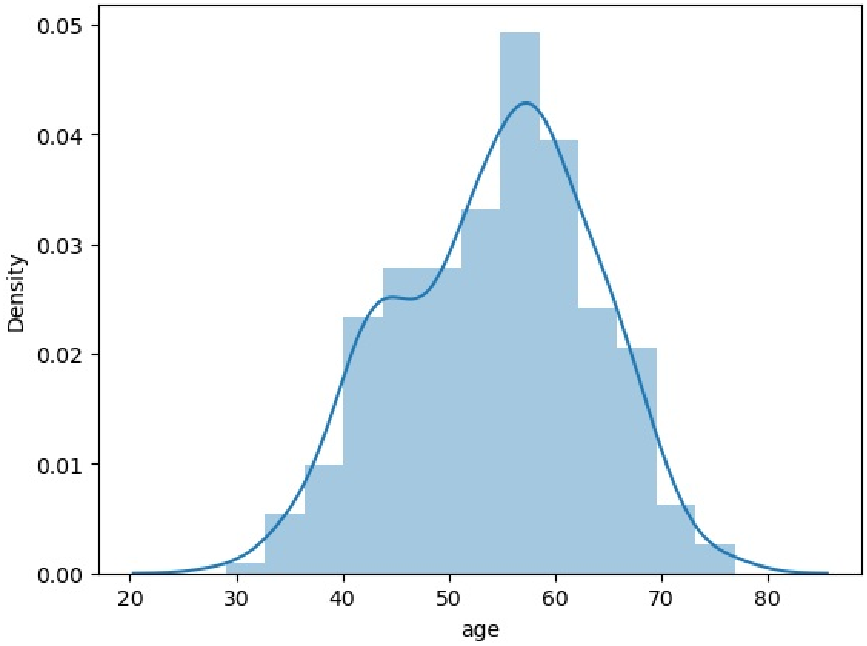
```
C:\Users\HP\AppData\Local\Temp\ipykernel_8356\3668578308.py:1: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  sns.distplot(data['age'])
```



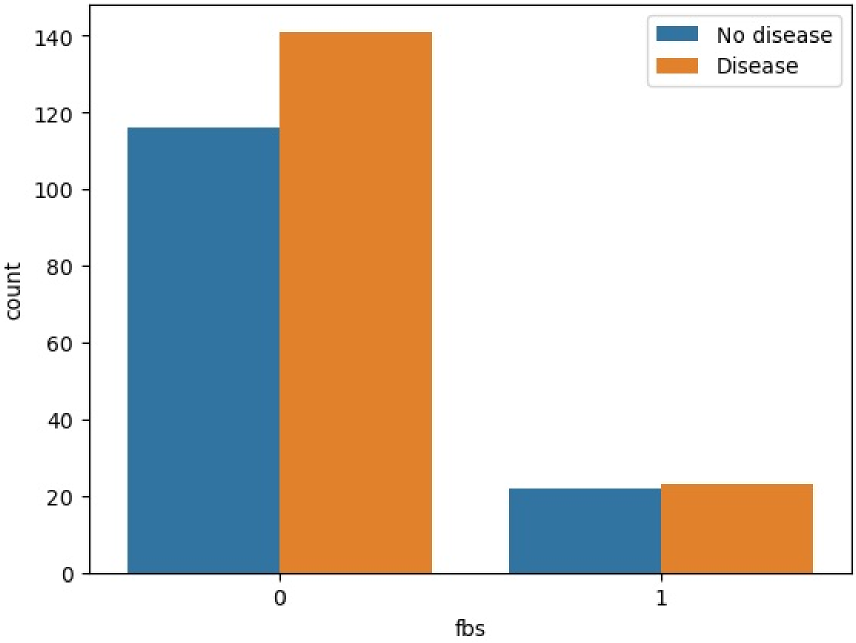
Most of the patients who are suffering from heart disease are of age 55 years (approximately).

Fasting Blood Sugar distribution according to the target

```
In [52]: data.columns
```

```
Out[52]: Index(['age', 'sex', 'cp', 'trestbps', 'chol', 'fbs', 'restecg', 'thalach',
                'exang', 'oldpeak', 'slope', 'ca', 'thal', 'target'],
               dtype='object')
```

```
In [54]: sns.countplot(x='fbs',hue='target',data=data)
         plt.legend(labels=['No disease','Disease'])
         plt.show()
```

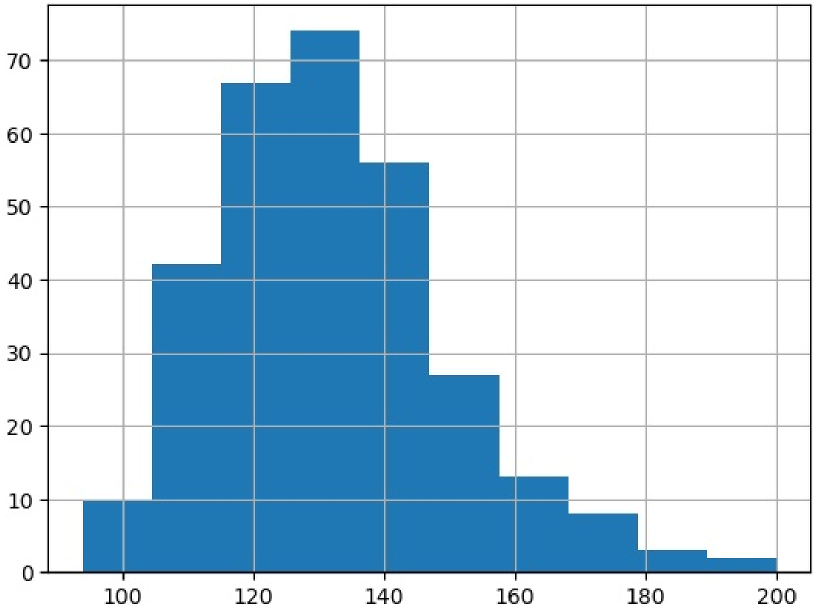


Checking resting blood pressure distribution

In [55]:
```
data.columns
```

Out[55]:
```
Index(['age', 'sex', 'cp', 'trestbps', 'chol', 'fbs', 'restecg', 'thalach',
       'exang', 'oldpeak', 'slope', 'ca', 'thal', 'target'],
      dtype='object')
```

In [57]:
```
data['trestbps'].hist()
plt.show()
```



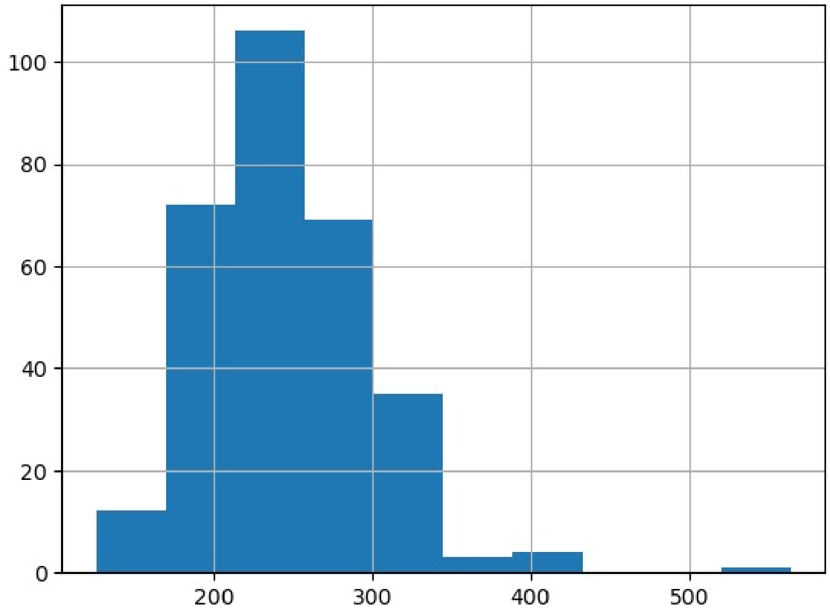
Most of the patients have resting blood pressure of 130.

Distribution of Serum Cholesterol

In [62]:
```
data.columns
```

Out[62]:
```
Index(['age', 'sex', 'cp', 'trestbps', 'chol', 'fbs', 'restecg', 'thalach',
       'exang', 'oldpeak', 'slope', 'ca', 'thal', 'target'],
      dtype='object')
```

In [63]:
```
data['chol'].hist()
plt.show()
```



Plot continuous variables

In [64]:

```
data.columns
```

Out[64]:

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

In [67]:

```
cate_val=[]
cont_val=[]

for column in data.columns:
    if data[column].nunique() <=10:
        cate_val.append(column)
    else:
        cont_val.append(column)
```

In [68]:

```
cate_val
```

Out[68]:

```
['sex', 'cp', 'fbs', 'restecg', 'exang', 'slope', 'ca', 'thal', 'target']
```

In [69]:

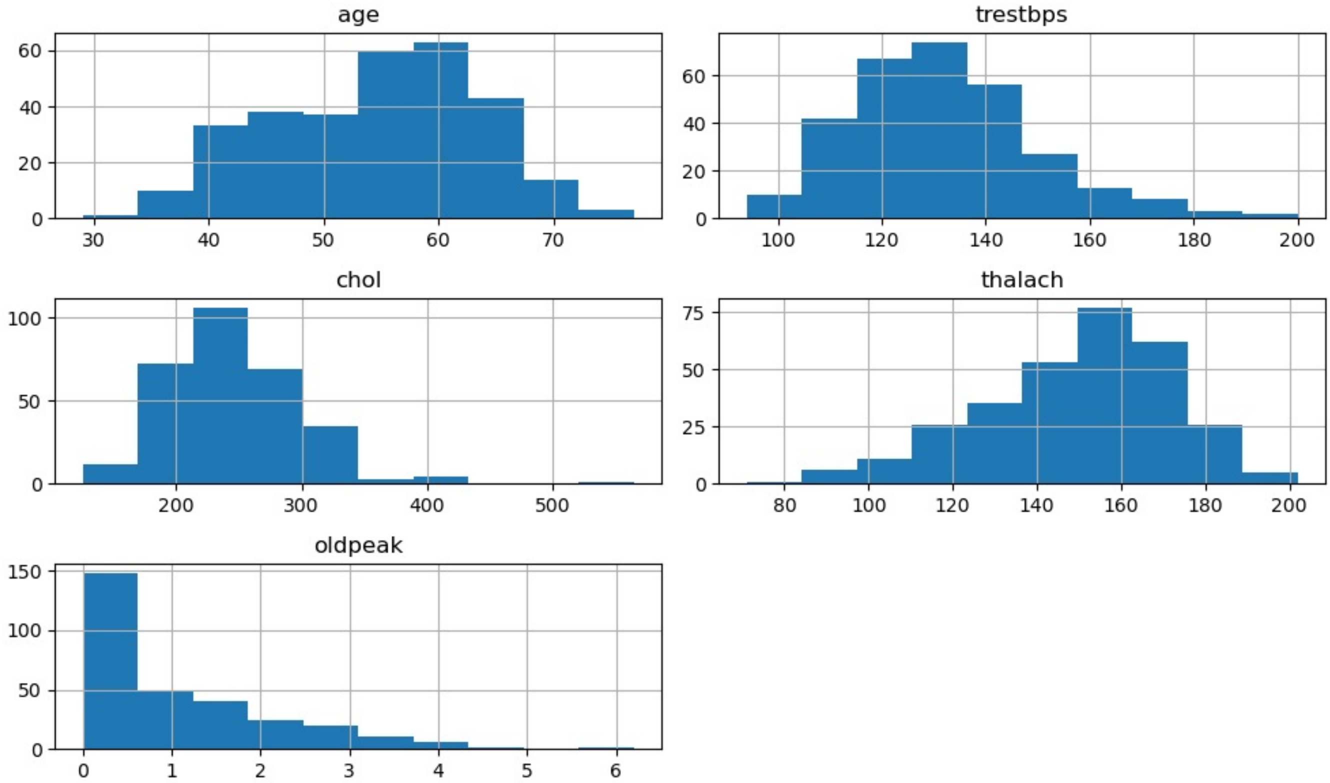
```
cont_val
```

Out[69]:

```
['age', 'trestbps', 'chol', 'thalach', 'oldpeak']
```

In [73]:

```
data.hist(cont_val,figsize=(10,6))
plt.tight_layout()
plt.show()
```



Conclusion :

1) There are more male patients who are suffering from heart disease than female patients. Apart from this the number of healthy male people is greater than those of male patients suffering from heart disease. The number of healthy women are lesser than the number of healthy male .

2) Most of the patients who are suffering from heart disease are of age 55 years (approximately). 3) 4) Most of the patients have resting blood pressure of 130 5)

```
In [ ]:
```

Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js