

```
In [3]: import numpy as np
import pandas as pd
import os
for dirname, _, filenames in os.walk('/kaggle/input'):
    for filename in filenames:
        print(os.path.join(dirname, filename))
```

```
In [5]: import seaborn as sns
import matplotlib.pyplot as plt
import scipy.stats as st
%matplotlib inline
sns.set(style='whitegrid')
```

```
In [7]: import warnings
warnings.filterwarnings('ignore')
```

import dataset

```
In [16]: df=pd.read_csv(r"C:\Users\MANISHA\OneDrive\Documents\heart.csv")
```

Exploratory data analysis

```
In [21]: print('The shape of the dataset:',df.shape)
```

The shape of the dataset: (303, 14)

```
In [23]: df.head()
```

```
Out[23]:   age  sex  cp  trestbps  chol  fbs  restecg  thalach  exang  oldpeak  slope  ca  thal  target
0     63     1     3      145    233     1      0     150      0      2.3      0     0     1
1     37     1     2      130    250     0      1     187      0      3.5      0     0     2
2     41     0     1      130    204     0      0     172      0      1.4      2     0     2
3     56     1     1      120    236     0      1     178      0      0.8      2     0     2
4     57     0     0      120    354     0      1     163      1      0.6      2     0     2
```



```
In [25]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 303 entries, 0 to 302
Data columns (total 14 columns):
 #   Column      Non-Null Count  Dtype  
--- 
 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)
memory usage: 33.3 KB
```

In [27]: `df.dtypes`

```
Out[27]: age        int64
          sex        int64
          cp         int64
          trestbps  int64
          chol       int64
          fbs        int64
          restecg   int64
          thalach   int64
          exang     int64
          oldpeak   float64
          slope     int64
          ca         int64
          thal       int64
          target    int64
          dtype: object
```

Statistical properties of dataset

In [29]: `df.describe()`

Out[29]:

	age	sex	cp	trestbps	chol	fbs	restecg
count	303.000000	303.000000	303.000000	303.000000	303.000000	303.000000	303.000000
mean	54.366337	0.683168	0.966997	131.623762	246.264026	0.148515	0.528053
std	9.082101	0.466011	1.032052	17.538143	51.830751	0.356198	0.525860
min	29.000000	0.000000	0.000000	94.000000	126.000000	0.000000	0.000000
25%	47.500000	0.000000	0.000000	120.000000	211.000000	0.000000	0.000000
50%	55.000000	1.000000	1.000000	130.000000	240.000000	0.000000	1.000000
75%	61.000000	1.000000	2.000000	140.000000	274.500000	0.000000	1.000000
max	77.000000	1.000000	3.000000	200.000000	564.000000	1.000000	2.000000



In [31]: `df.columns`

Out[31]: `Index(['age', 'sex', 'cp', 'trestbps', 'chol', 'fbs', 'restecg', 'thalach', 'exang', 'oldpeak', 'slope', 'ca', 'thal', 'target'], dtype='object')`

Univariate analysis

In [34]: `df['target'].nunique()`

Out[34]: 2

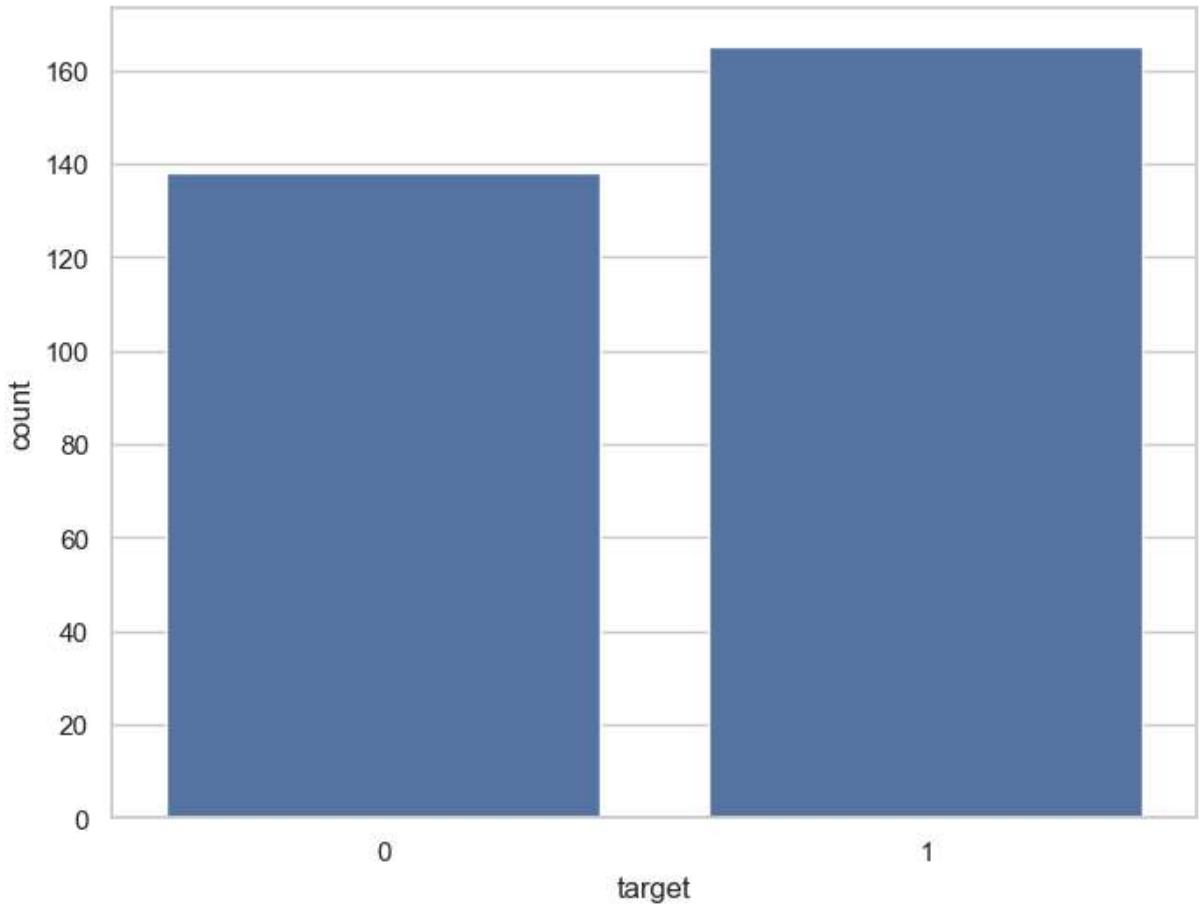
In [36]: `df['target'].unique()`

Out[36]: `array([1, 0], dtype=int64)`

In [38]: `df['target'].value_counts()`

Out[38]: `target`
1 165
0 138
Name: count, dtype: int64

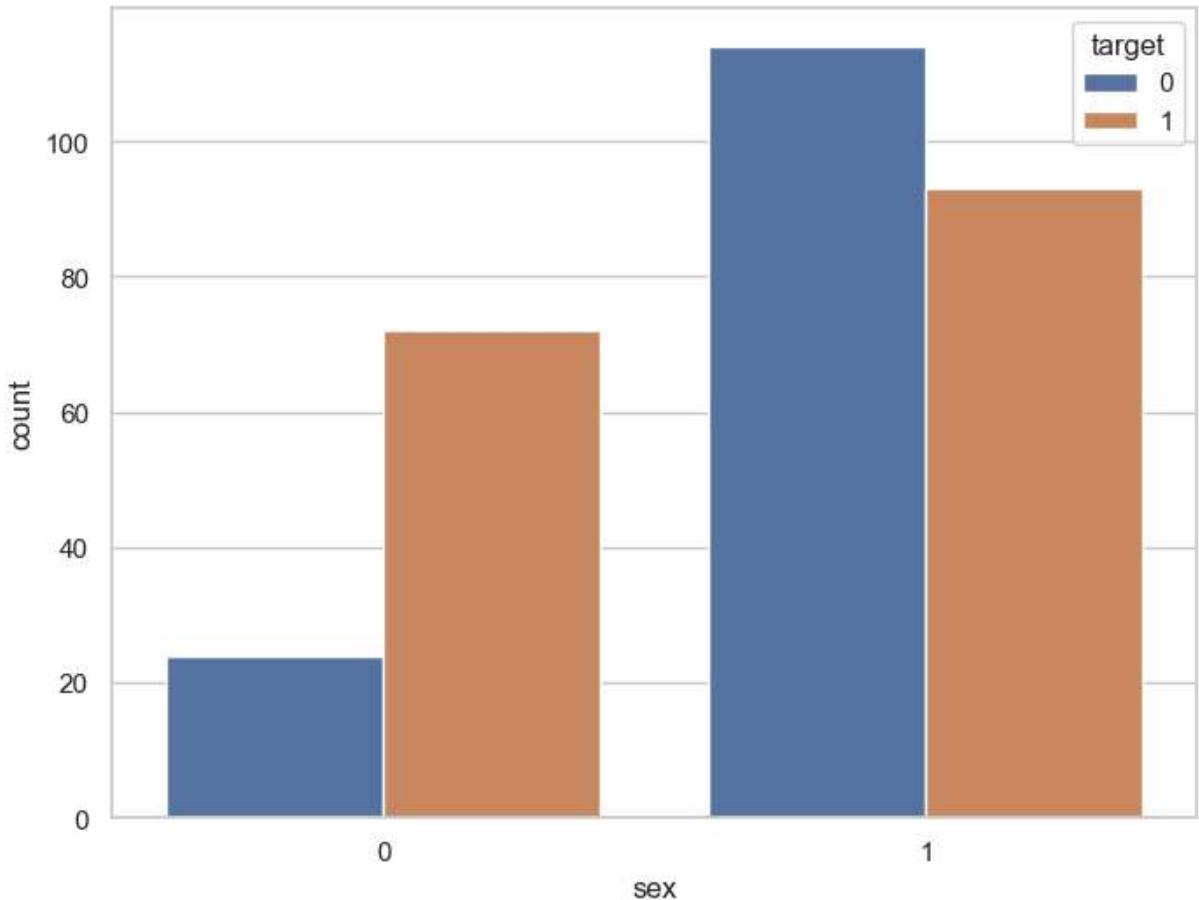
In [40]: `f,ax=plt.subplots(figsize=(8,6))
ax=sns.countplot(x="target",data=df)
plt.show()`



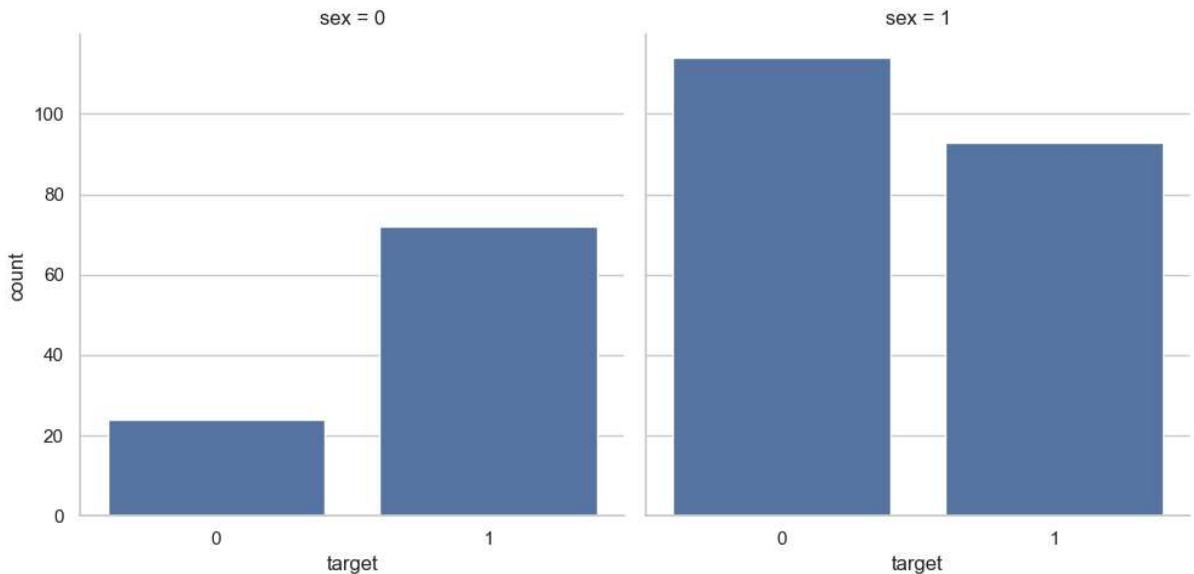
```
In [42]: df.groupby('sex')['target'].value_counts()
```

```
Out[42]: sex  target
          0      1        72
                  0        24
          1      0       114
                  1        93
Name: count, dtype: int64
```

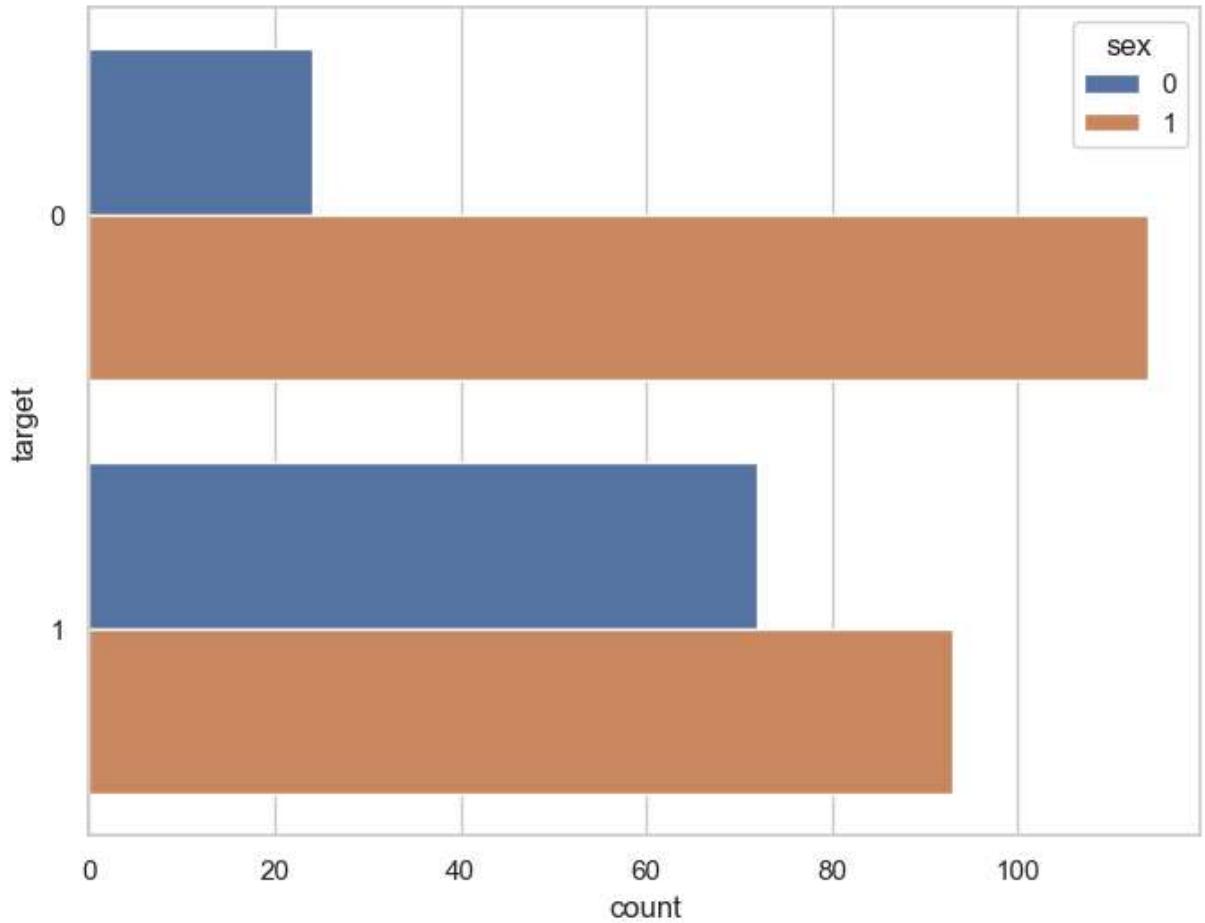
```
In [44]: f,ax=plt.subplots(figsize=(8,6))
ax=sns.countplot(x='sex',hue='target',data=df)
plt.show()
```



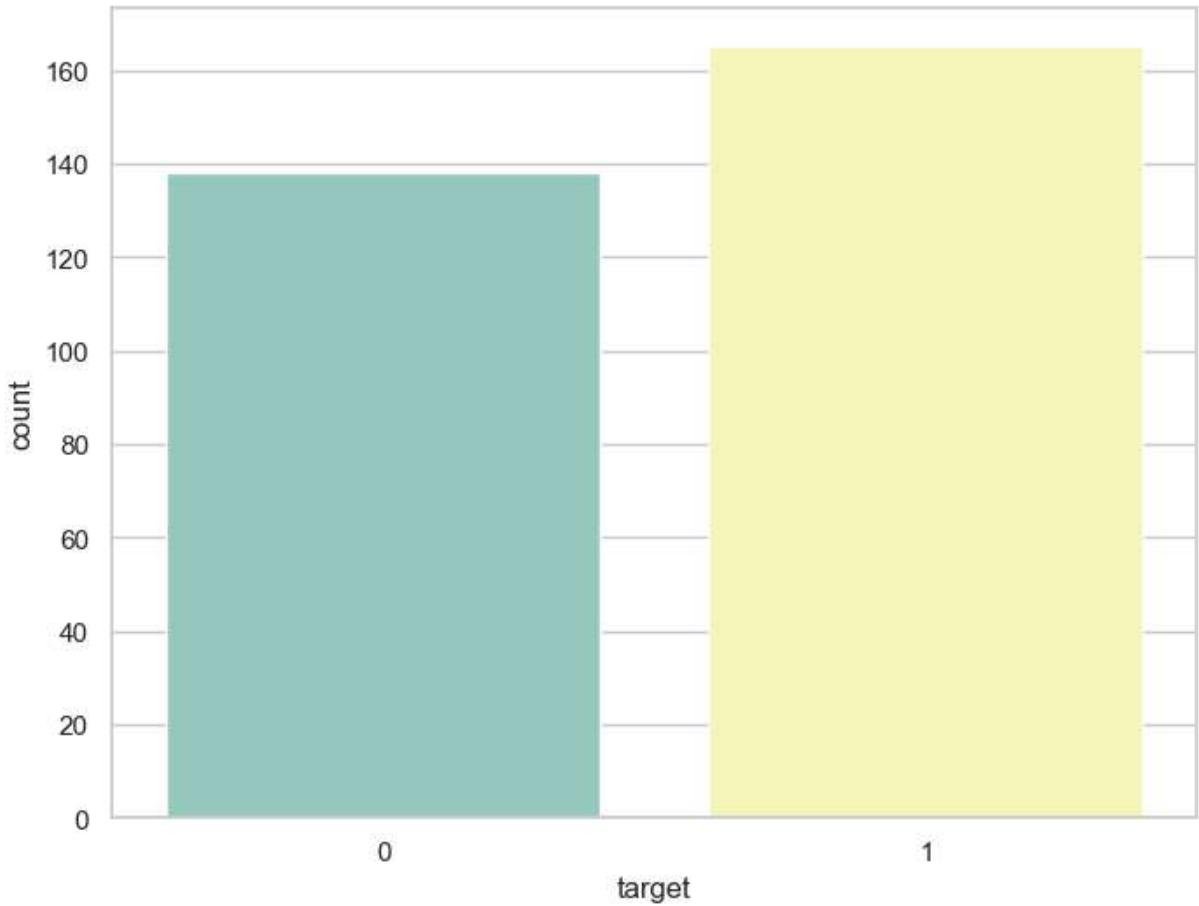
```
In [46]: ax=sns.catplot(x='target',col='sex',data=df,kind='count',height=5,aspect=1)
```



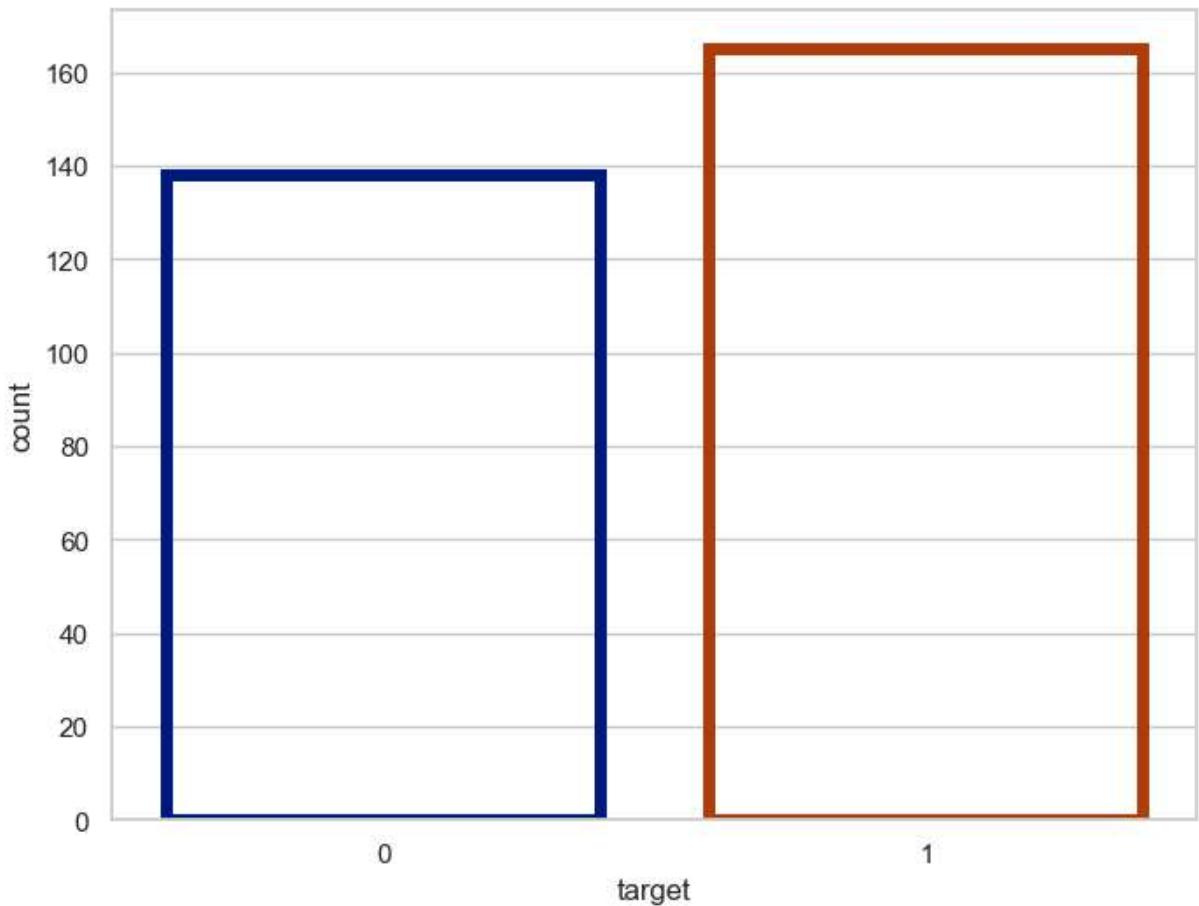
```
In [48]: f, ax = plt.subplots(figsize=(8, 6))
ax = sns.countplot(y="target", hue="sex", data=df)
plt.show()
```



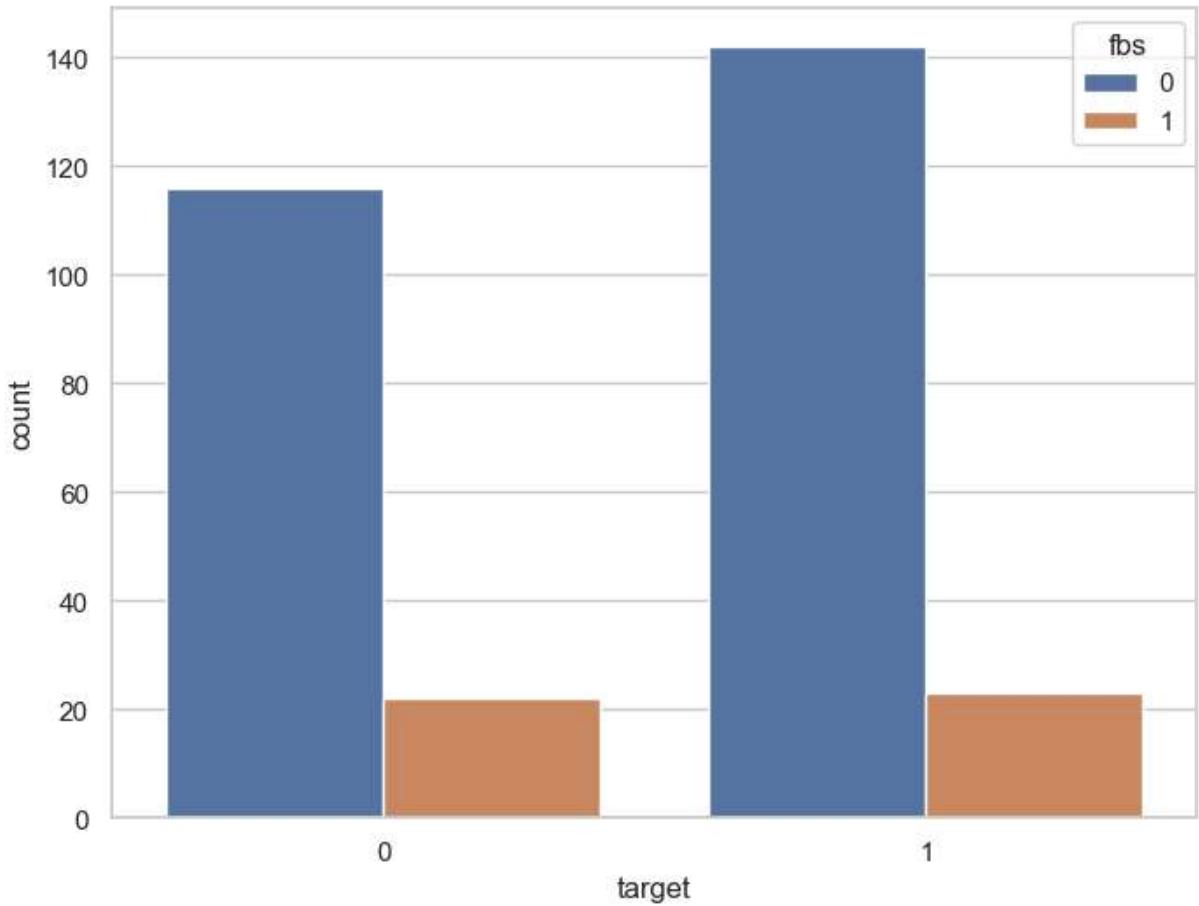
```
In [50]: f, ax = plt.subplots(figsize=(8, 6))
ax = sns.countplot(x="target", data=df, palette="Set3")
plt.show()
```



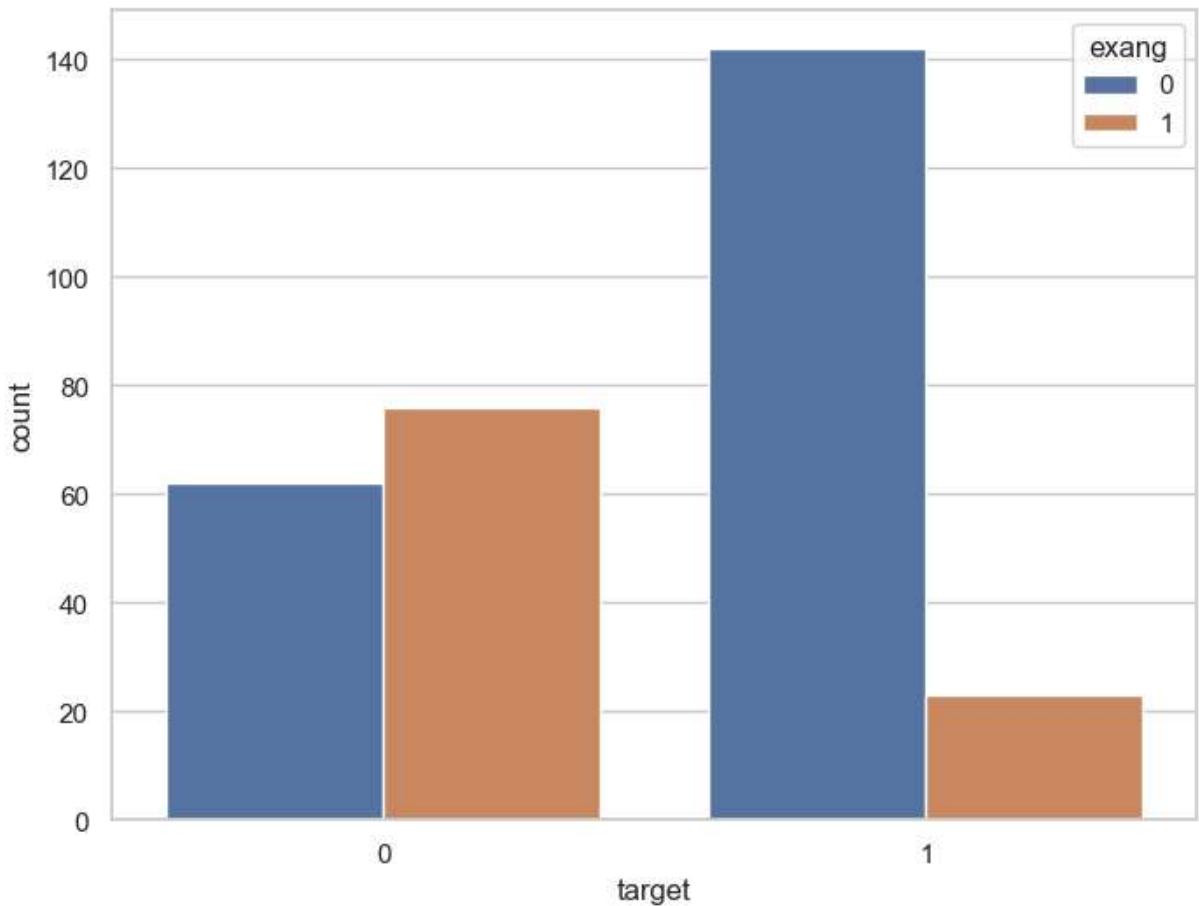
```
In [52]: f, ax = plt.subplots(figsize=(8, 6))
ax = sns.countplot(x="target", data=df, facecolor=(0, 0, 0, 0), linewidth=5, edgecolor='black')
plt.show()
```



```
In [54]: f, ax = plt.subplots(figsize=(8, 6))
ax = sns.countplot(x="target", hue="fbs", data=df)
plt.show()
```



```
In [56]: f, ax = plt.subplots(figsize=(8, 6))
ax = sns.countplot(x="target", hue="exang", data=df)
plt.show()
```



Bivariate Analysis

```
In [60]: correlation=df.corr()
```

```
In [62]: correlation['target'].sort_values(ascending=False)
```

```
Out[62]: target      1.000000
cp          0.433798
thalach    0.421741
slope       0.345877
restecg    0.137230
fbs        -0.028046
chol        -0.085239
trestbps   -0.144931
age         -0.225439
sex         -0.280937
thal        -0.344029
ca          -0.391724
oldpeak    -0.430696
exang      -0.436757
Name: target, dtype: float64
```

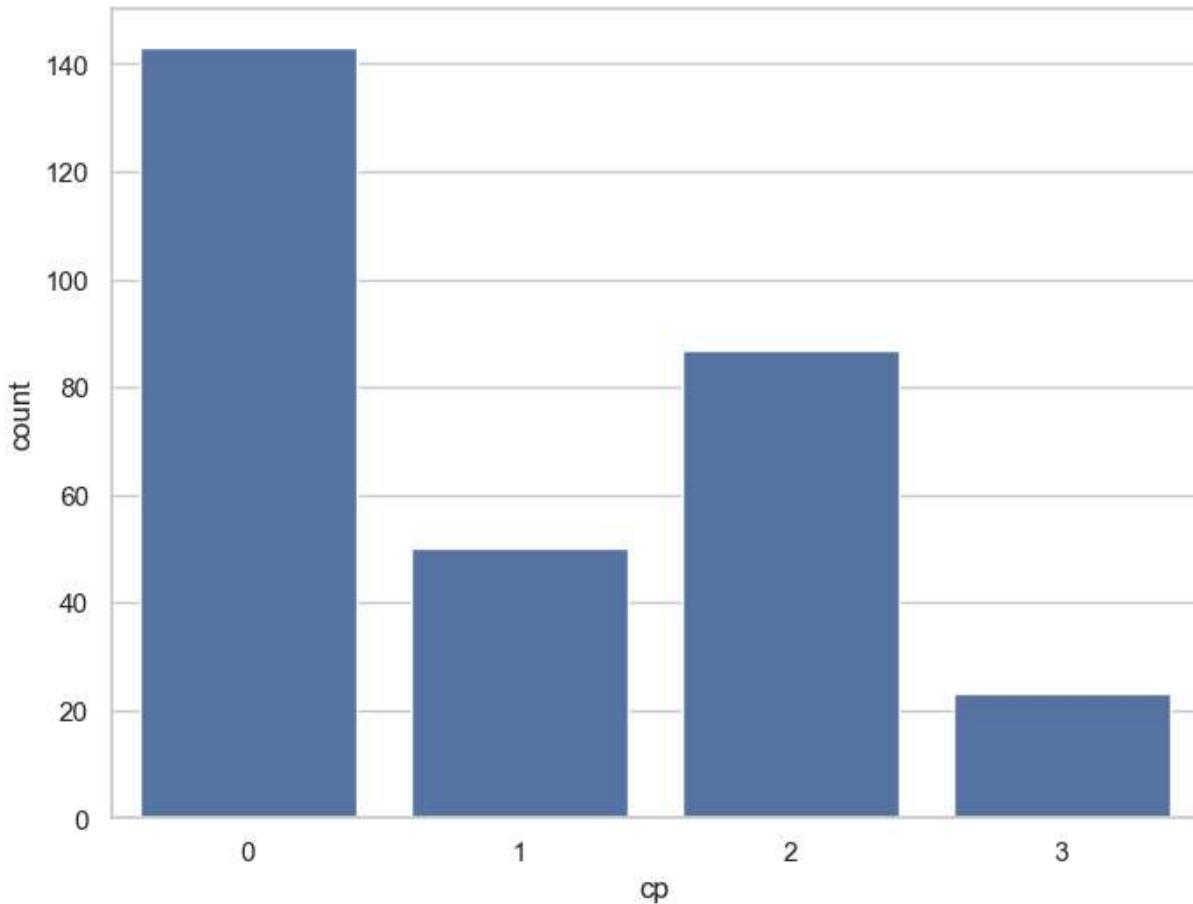
```
In [64]: df['cp'].nunique()
```

```
Out[64]: 4
```

```
In [66]: df['cp'].value_counts()
```

```
Out[66]: cp
0    143
2     87
1     50
3     23
Name: count, dtype: int64
```

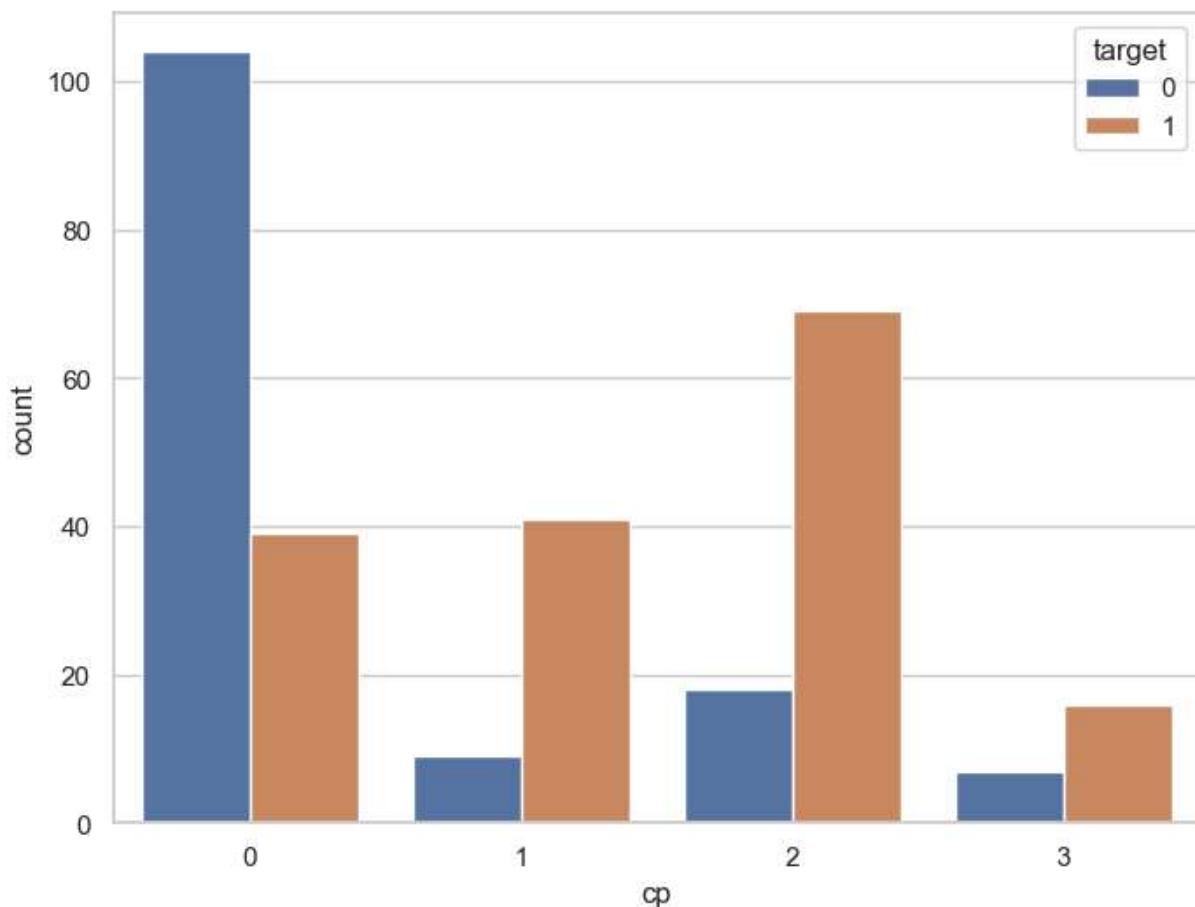
```
In [68]: f, ax = plt.subplots(figsize=(8, 6))
ax = sns.countplot(x="cp", data=df)
plt.show()
```



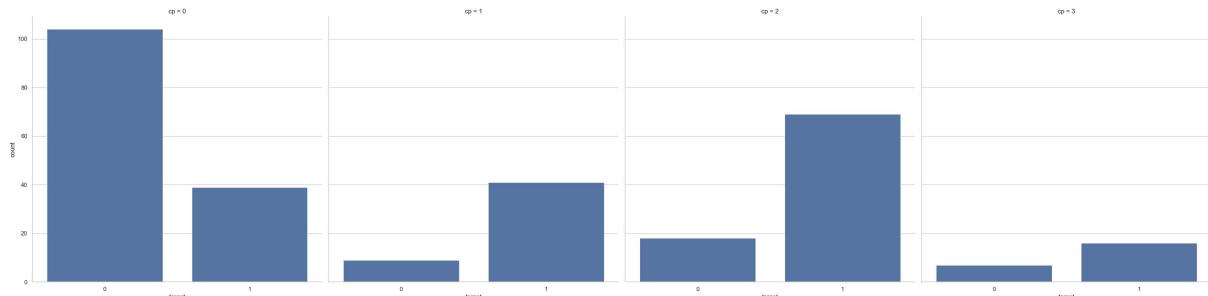
```
In [72]: df.groupby('cp')['target'].value_counts()
```

```
Out[72]: cp  target
0   0        104
      1        39
1   1        41
      0         9
2   1        69
      0        18
3   1        16
      0         7
Name: count, dtype: int64
```

```
In [74]: f, ax = plt.subplots(figsize=(8, 6))
ax = sns.countplot(x="cp", hue="target", data=df)
plt.show()
```



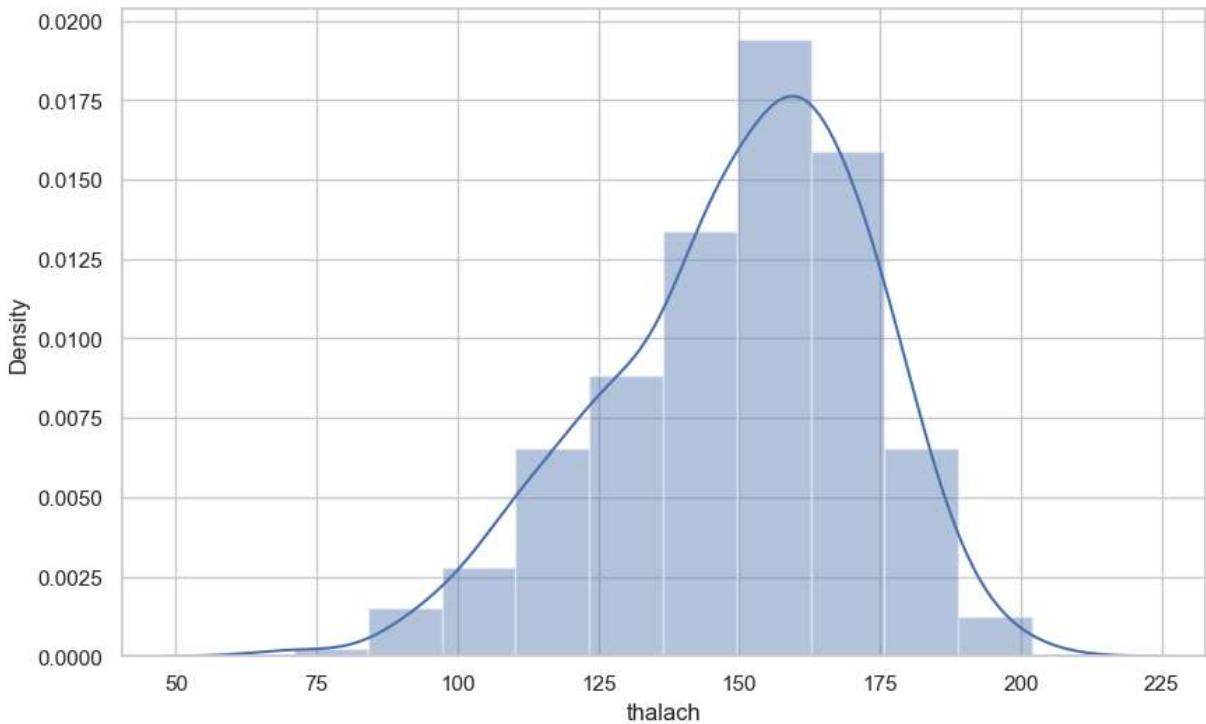
```
In [76]: ax = sns.catplot(x="target", col="cp", data=df, kind="count", height=8, aspect=1)
```



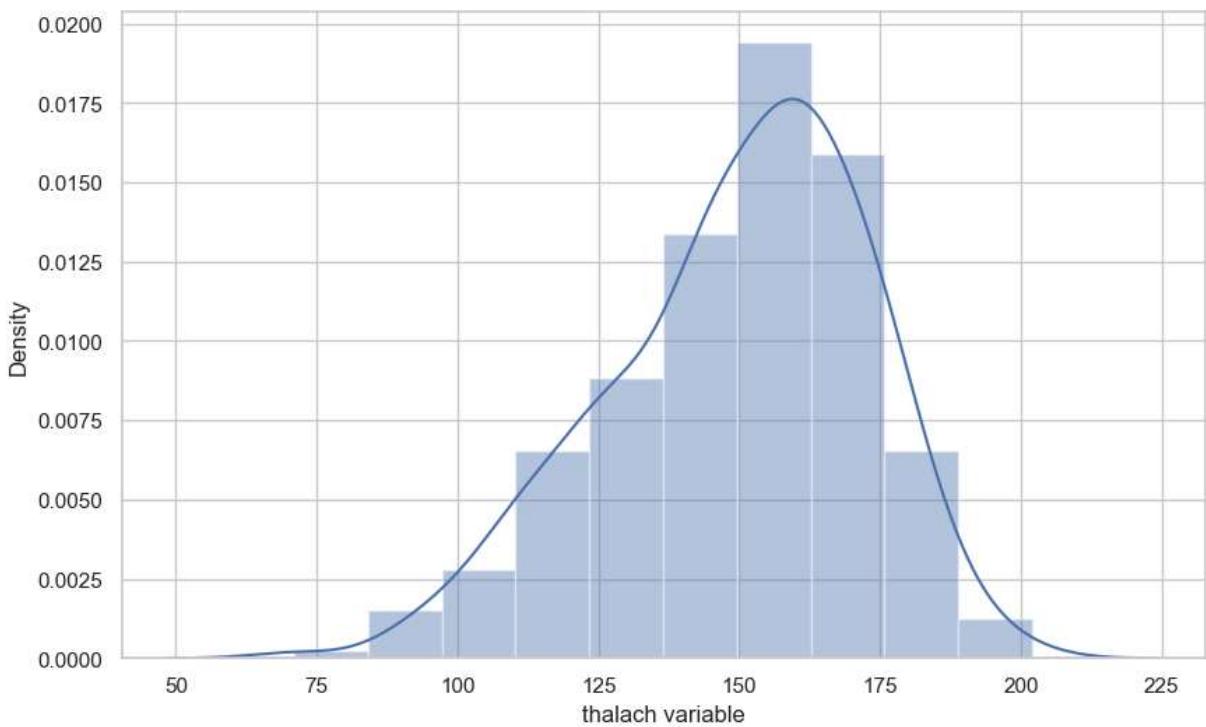
```
In [78]: df['thalach'].nunique()
```

```
Out[78]: 91
```

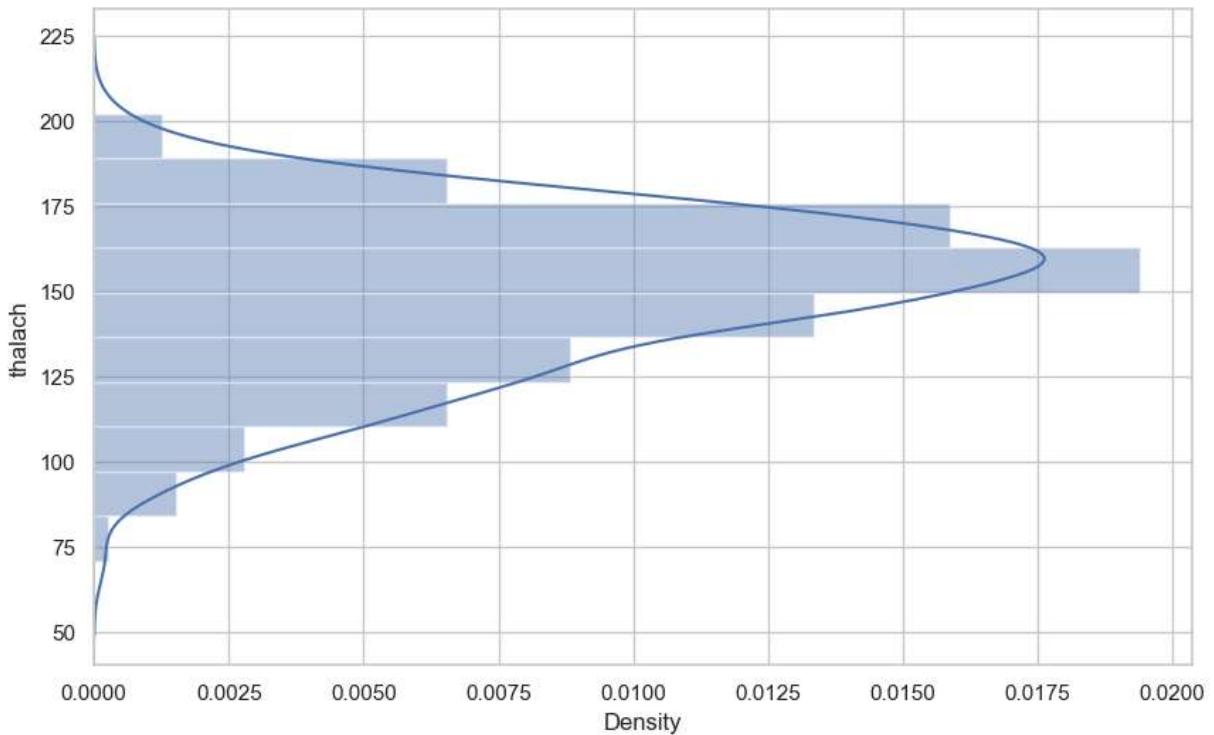
```
In [80]: f, ax = plt.subplots(figsize=(10,6))
x = df['thalach']
ax = sns.distplot(x, bins=10)
plt.show()
```



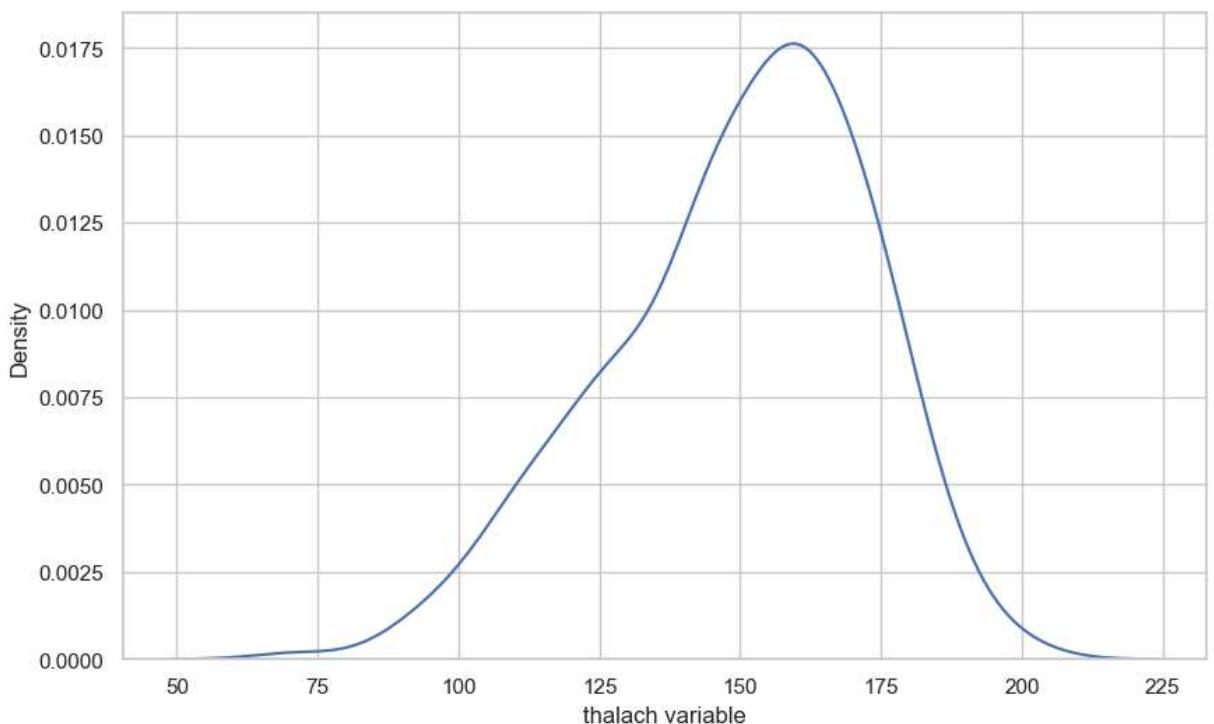
```
In [82]: f, ax = plt.subplots(figsize=(10,6))
x = df['thalach']
x = pd.Series(x, name="thalach variable")
ax = sns.distplot(x, bins=10)
plt.show()
```



```
In [84]: f, ax = plt.subplots(figsize=(10,6))
x = df['thalach']
ax = sns.distplot(x, bins=10, vertical=True)
plt.show()
```

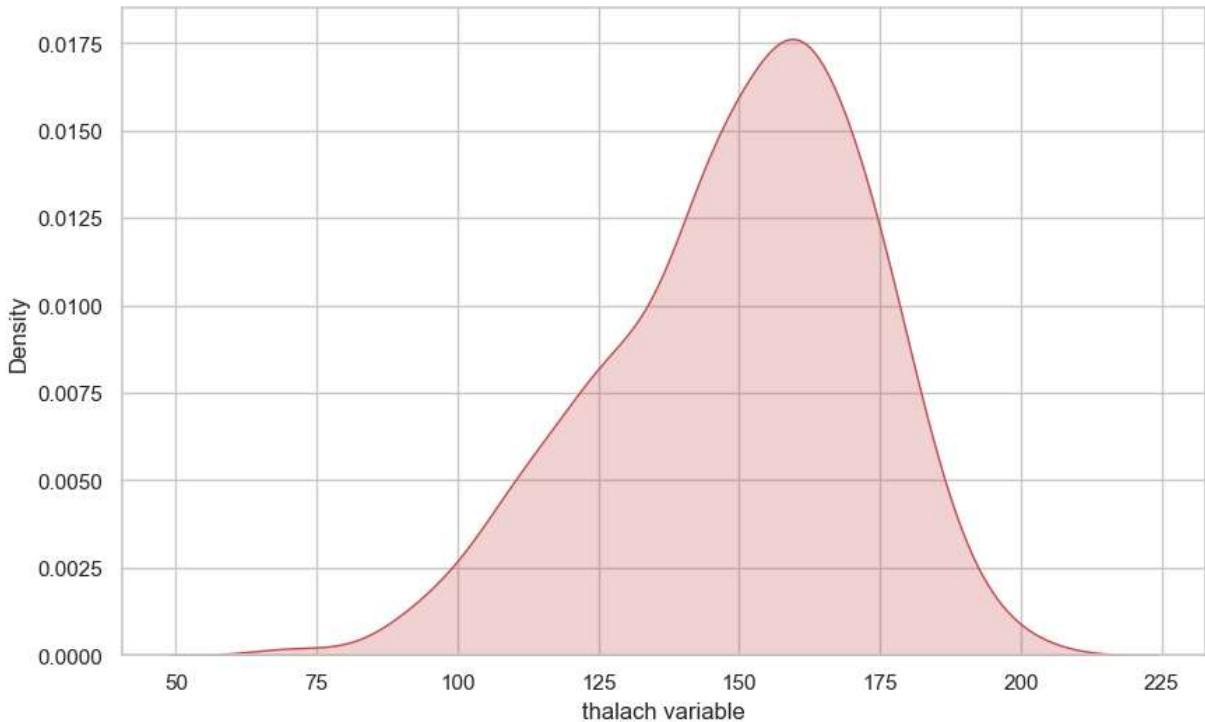


```
In [86]: f, ax = plt.subplots(figsize=(10,6))
x = df['thalach']
x = pd.Series(x, name="thalach variable")
ax = sns.kdeplot(x)
plt.show()
```

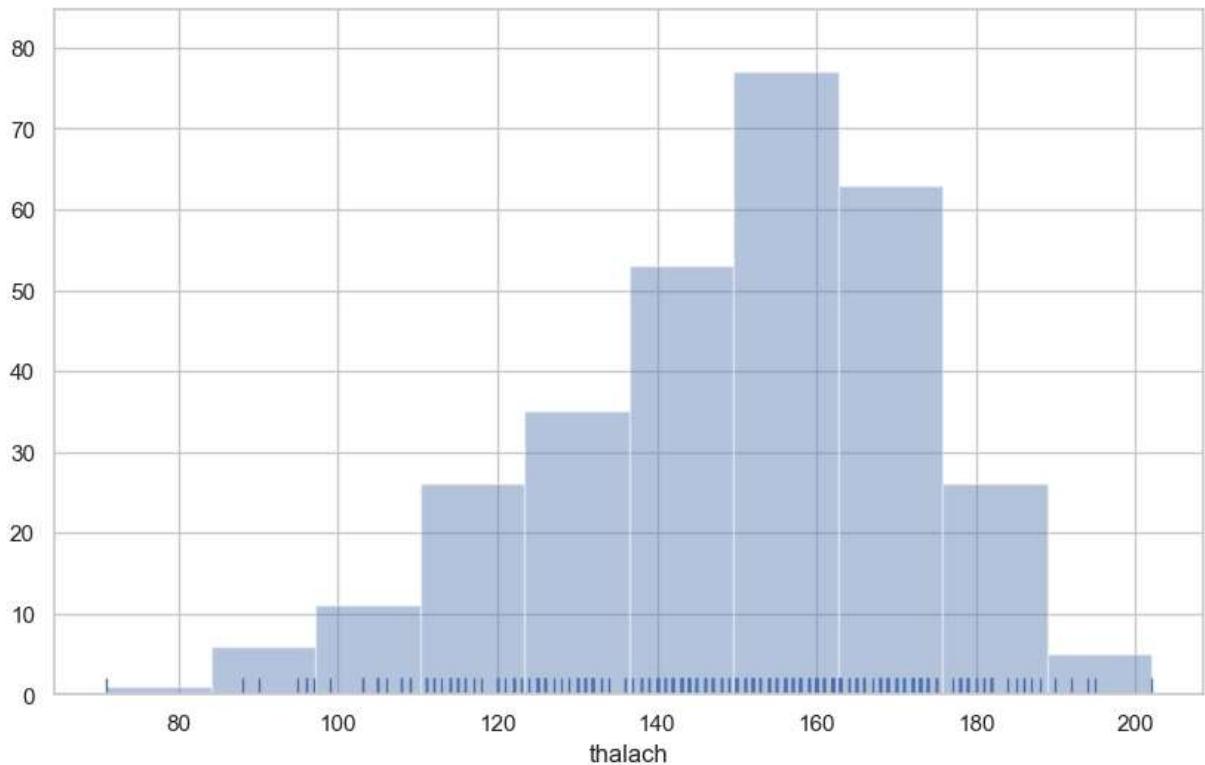


```
In [88]: f, ax = plt.subplots(figsize=(10,6))
x = df['thalach']
x = pd.Series(x, name="thalach variable")
```

```
ax = sns.kdeplot(x, shade=True, color='r')
plt.show()
```

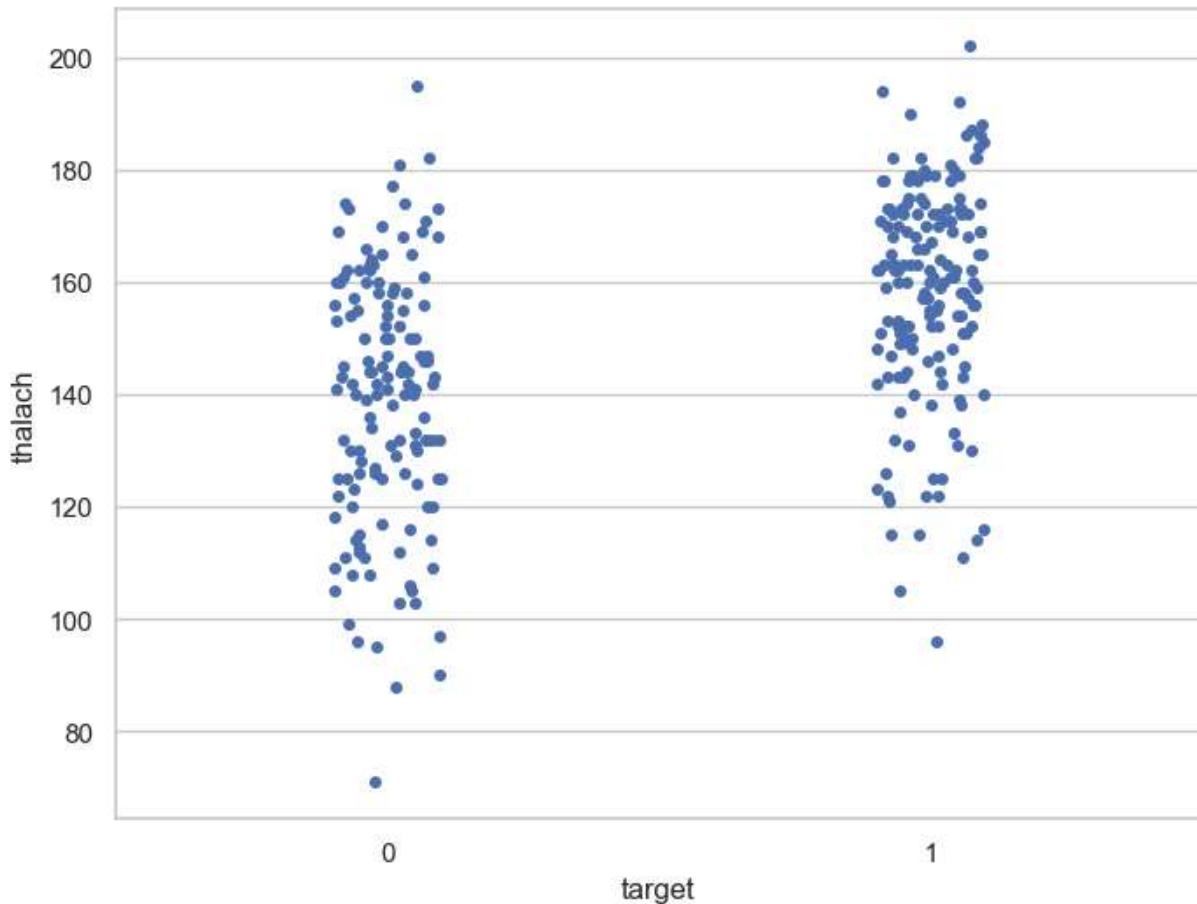


```
In [90]: f, ax = plt.subplots(figsize=(10,6))
x = df['thalach']
ax = sns.distplot(x, kde=False, rug=True, bins=10)
plt.show()
```

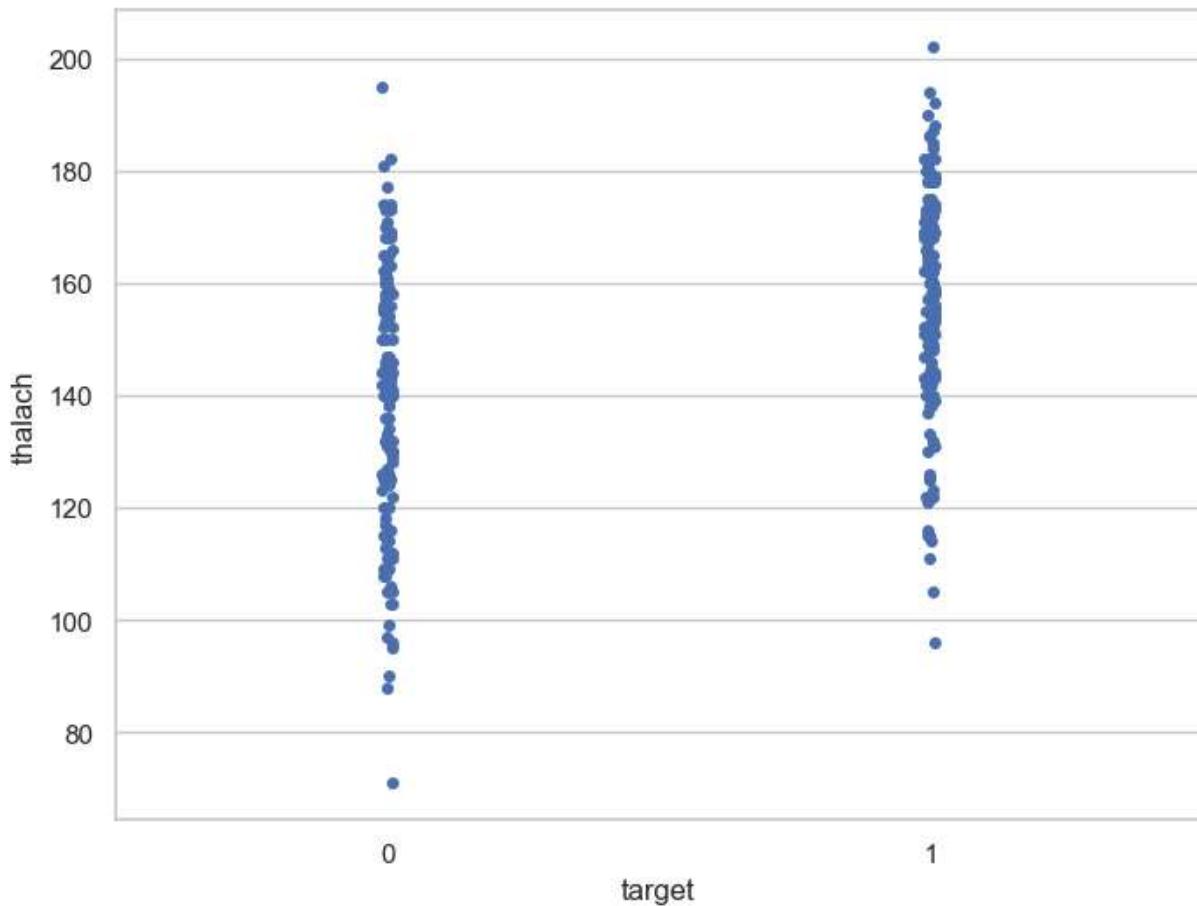


```
In [92]: f, ax = plt.subplots(figsize=(8, 6))
sns.stripplot(x="target", y="thalach", data=df)
```

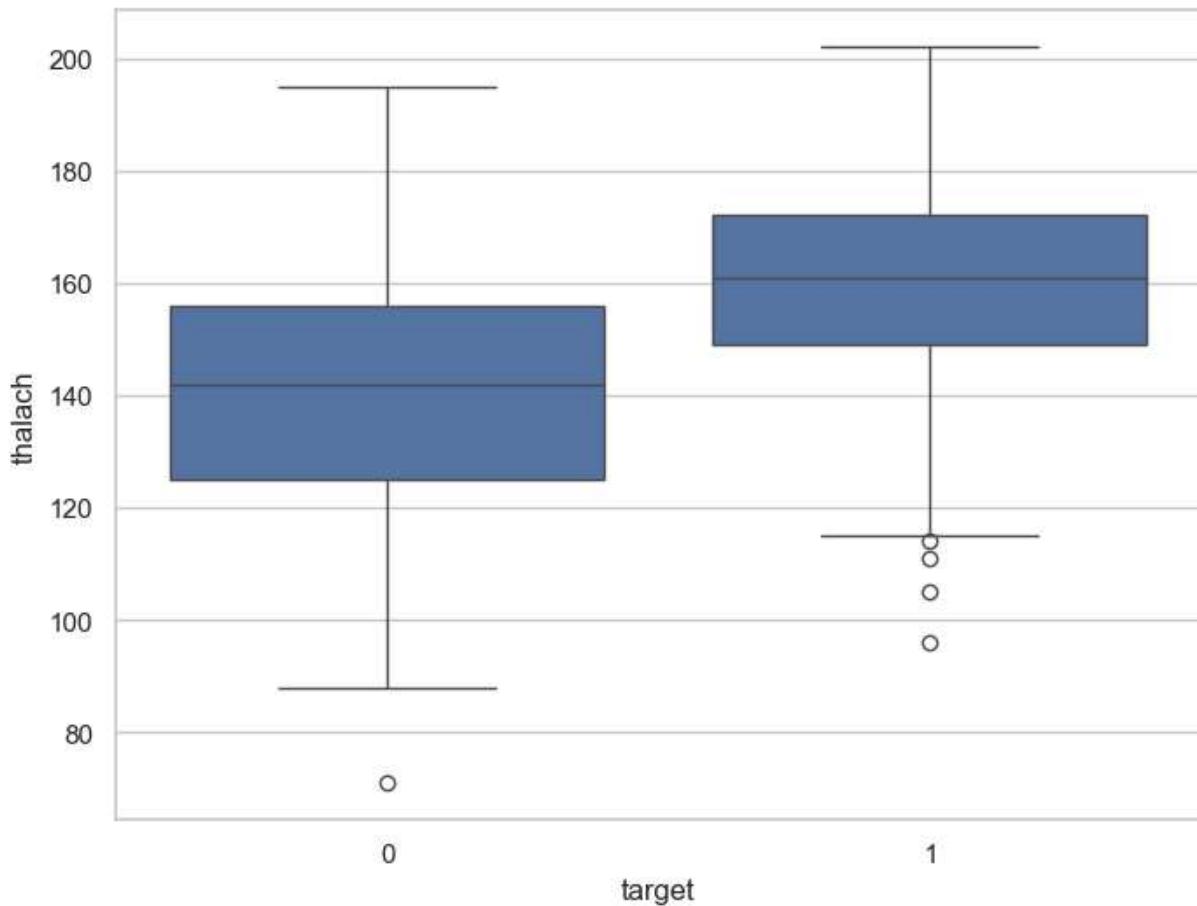
```
plt.show()
```



```
In [94]: f, ax = plt.subplots(figsize=(8, 6))
sns.stripplot(x="target", y="thalach", data=df, jitter = 0.01)
plt.show()
```



```
In [96]: f, ax = plt.subplots(figsize=(8, 6))
sns.boxplot(x="target", y="thalach", data=df)
plt.show()
```

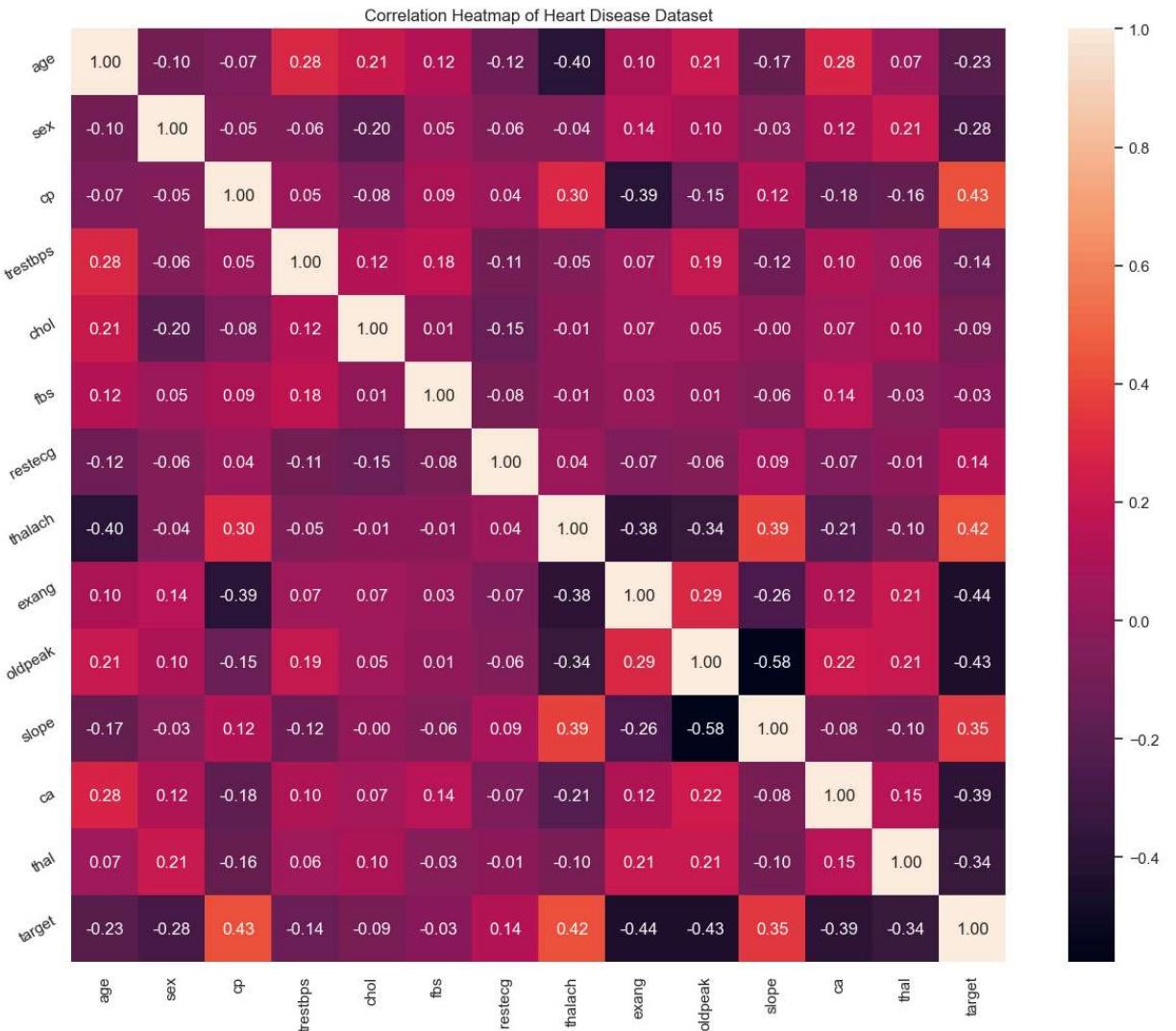


Multivariate analysis

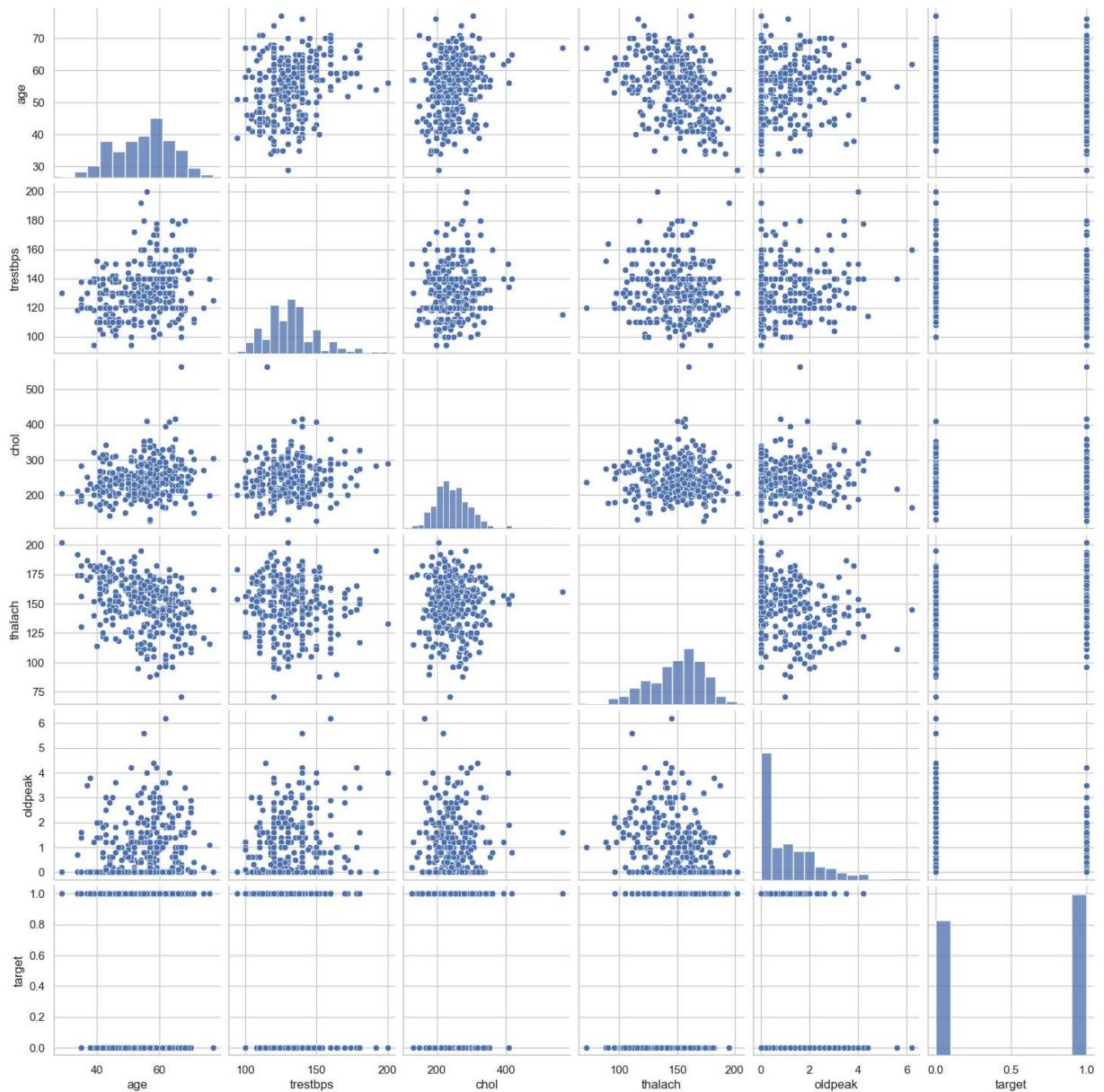
Heat map

In [103...]

```
plt.figure(figsize=(16,12))
plt.title('Correlation Heatmap of Heart Disease Dataset')
a = sns.heatmap(correlation, square=True, annot=True, fmt='.{2f}', linecolor='white')
a.set_xticklabels(a.get_xticklabels(), rotation=90)
a.set_yticklabels(a.get_yticklabels(), rotation=30)
plt.show()
```



```
In [105...]: num_var = ['age', 'trestbps', 'chol', 'thalach', 'oldpeak', 'target']
sns.pairplot(df[num_var], kind='scatter', diag_kind='hist')
plt.show()
```



```
In [107...]: df['age'].nunique()
```

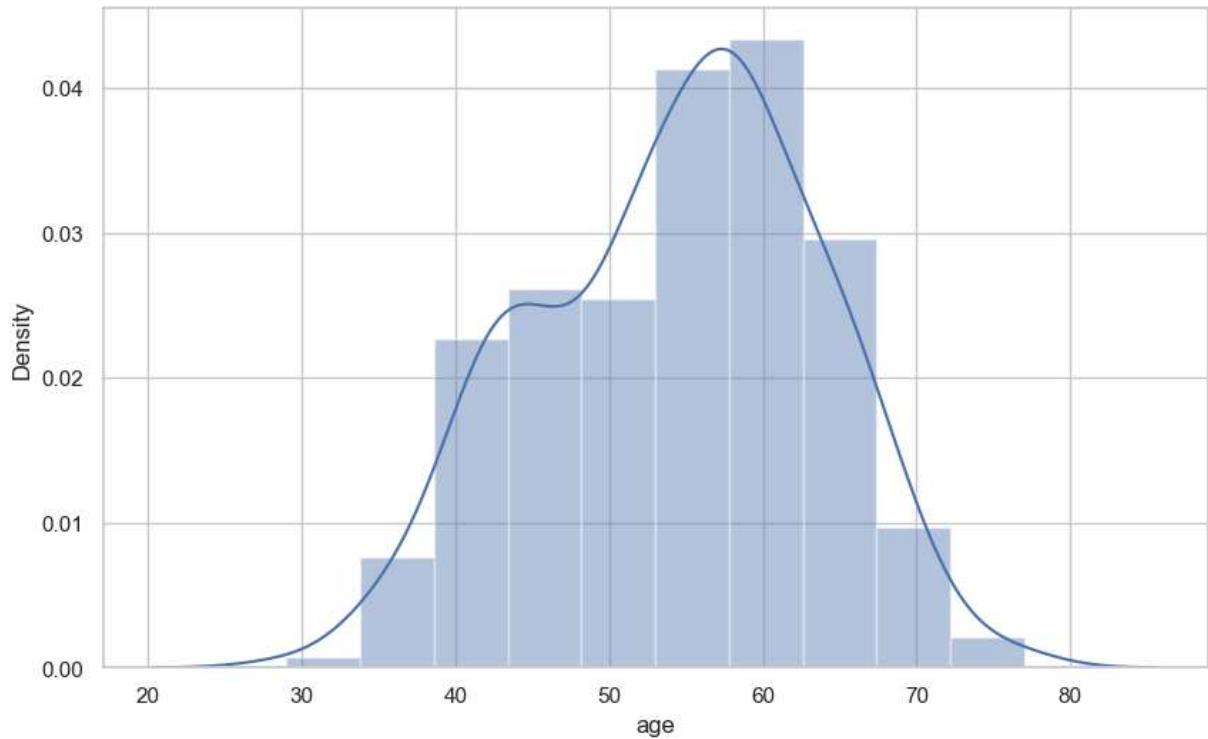
```
Out[107...]: 41
```

```
In [109...]: df['age'].describe()
```

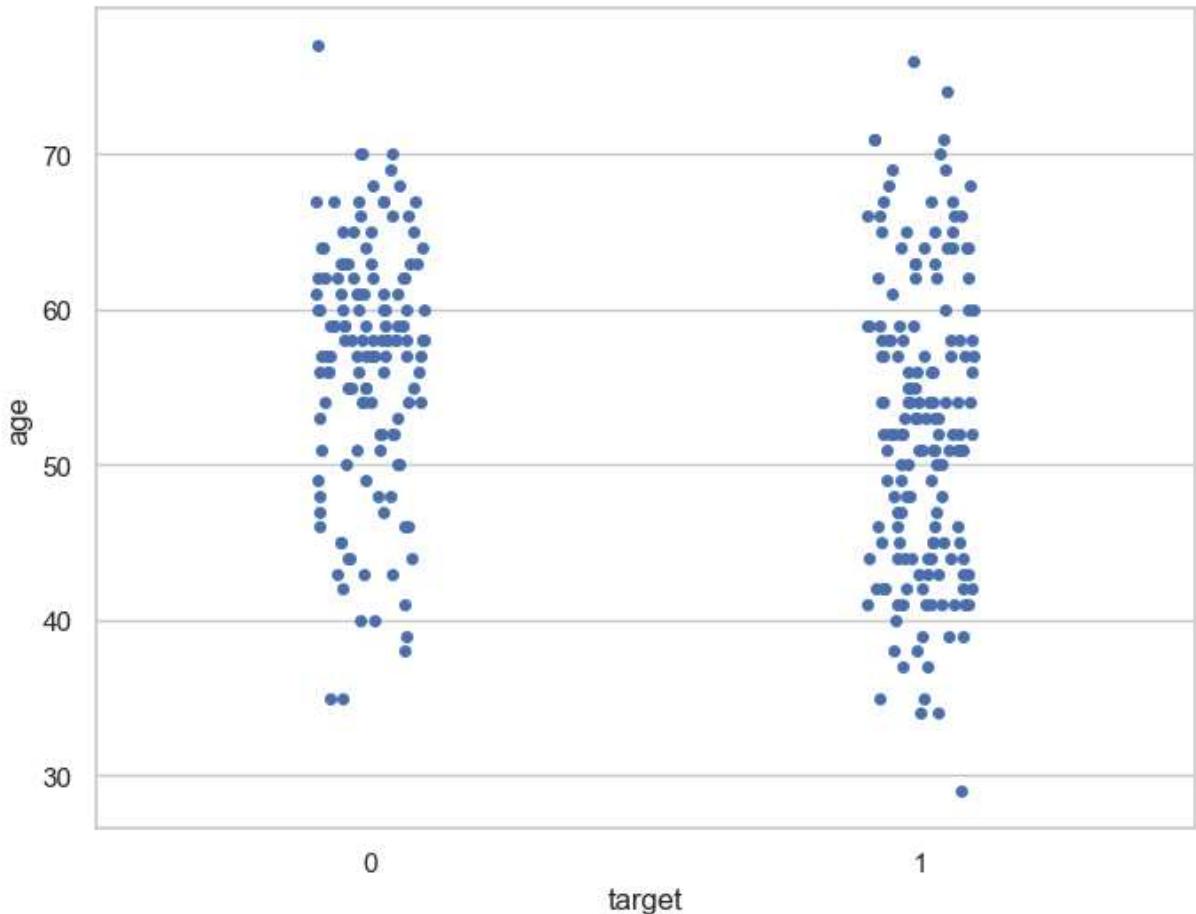
```
Out[109...]: count    303.000000
mean      54.366337
std       9.082101
min      29.000000
25%      47.500000
50%      55.000000
75%      61.000000
max      77.000000
Name: age, dtype: float64
```

```
In [111...]: f, ax = plt.subplots(figsize=(10,6))
x = df['age']
```

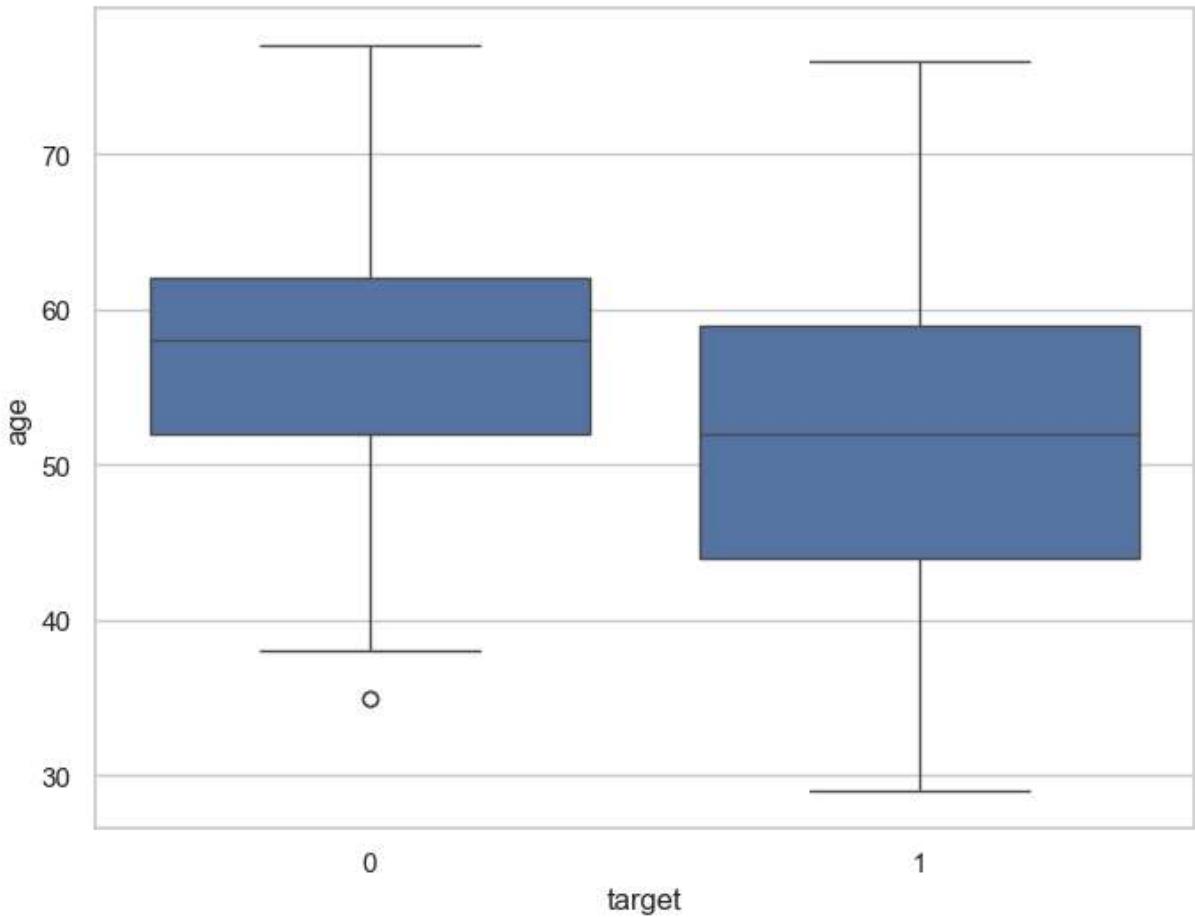
```
ax = sns.distplot(x, bins=10)
plt.show()
```



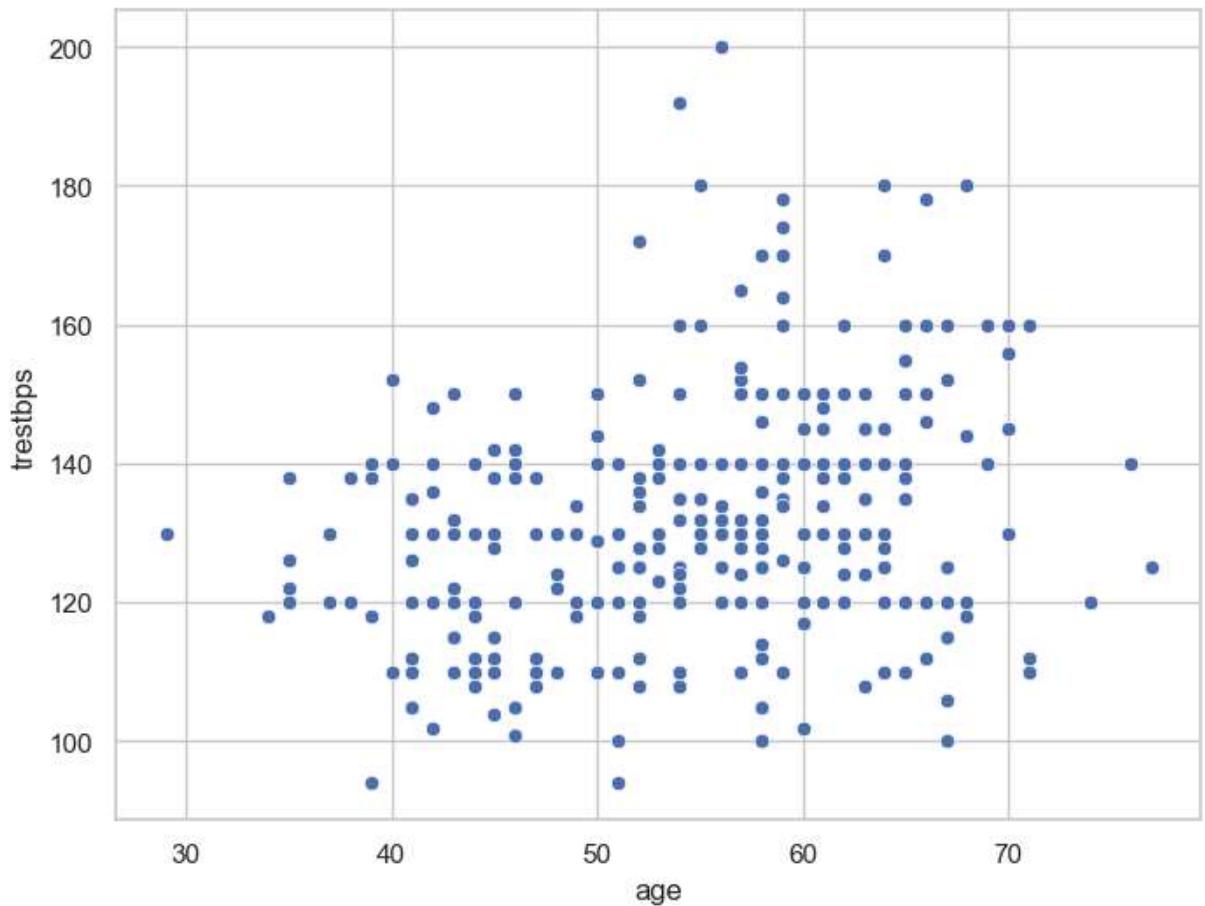
```
In [113]: f, ax = plt.subplots(figsize=(8, 6))
sns.stripplot(x="target", y="age", data=df)
plt.show()
```



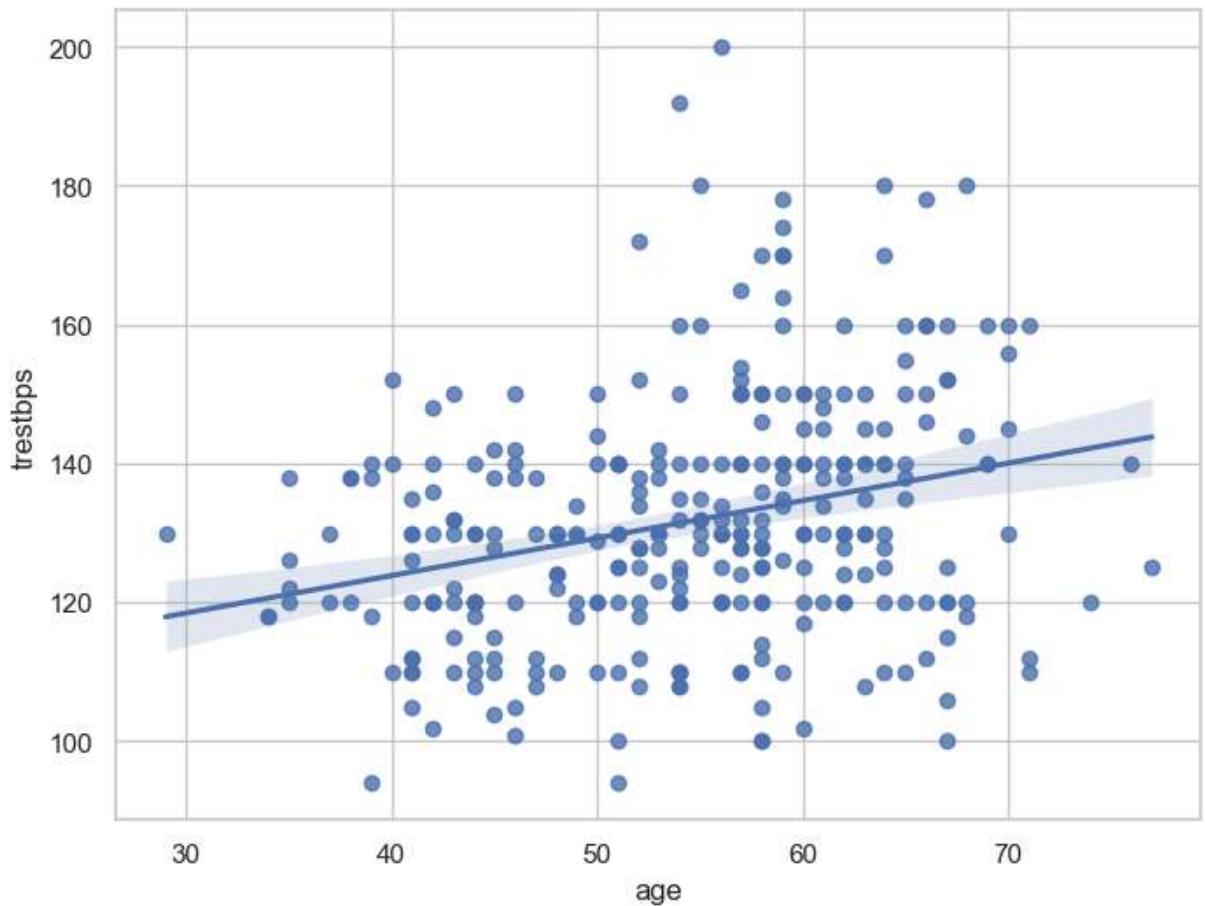
```
In [115]:  
f, ax = plt.subplots(figsize=(8, 6))  
sns.boxplot(x="target", y="age", data=df)  
plt.show()
```



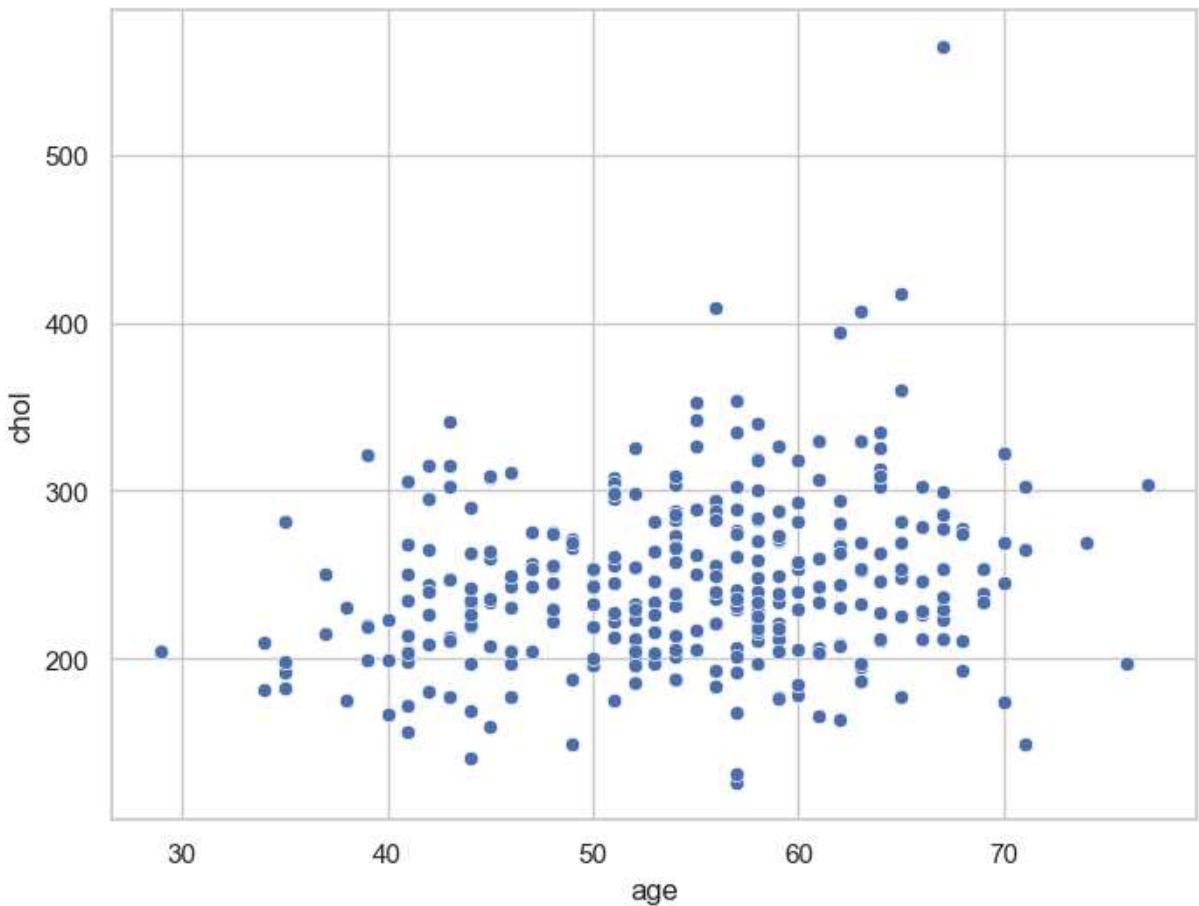
```
In [117]:  
f, ax = plt.subplots(figsize=(8, 6))  
ax = sns.scatterplot(x="age", y="trestbps", data=df)  
plt.show()
```



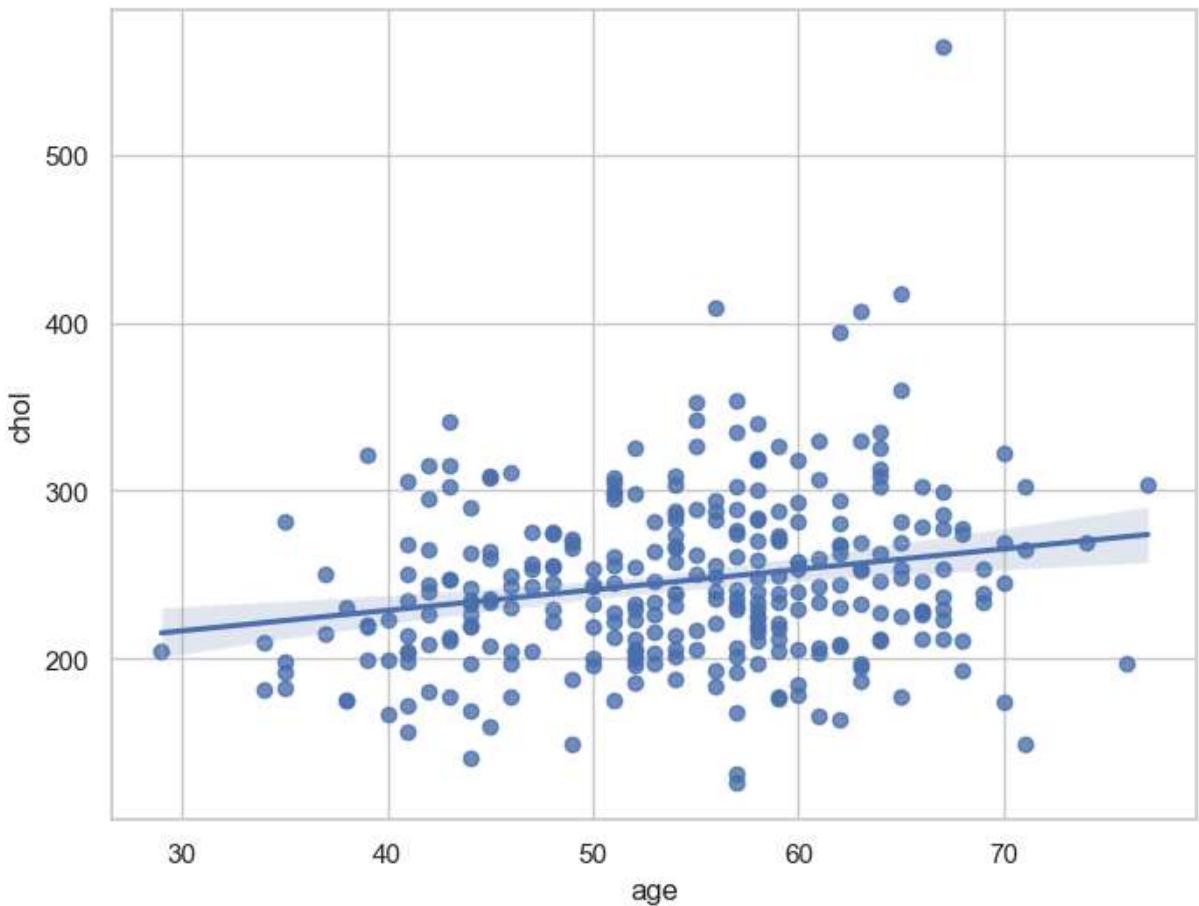
```
In [119]:  
f, ax = plt.subplots(figsize=(8, 6))  
ax = sns.regplot(x="age", y="trestbps", data=df)  
plt.show()
```



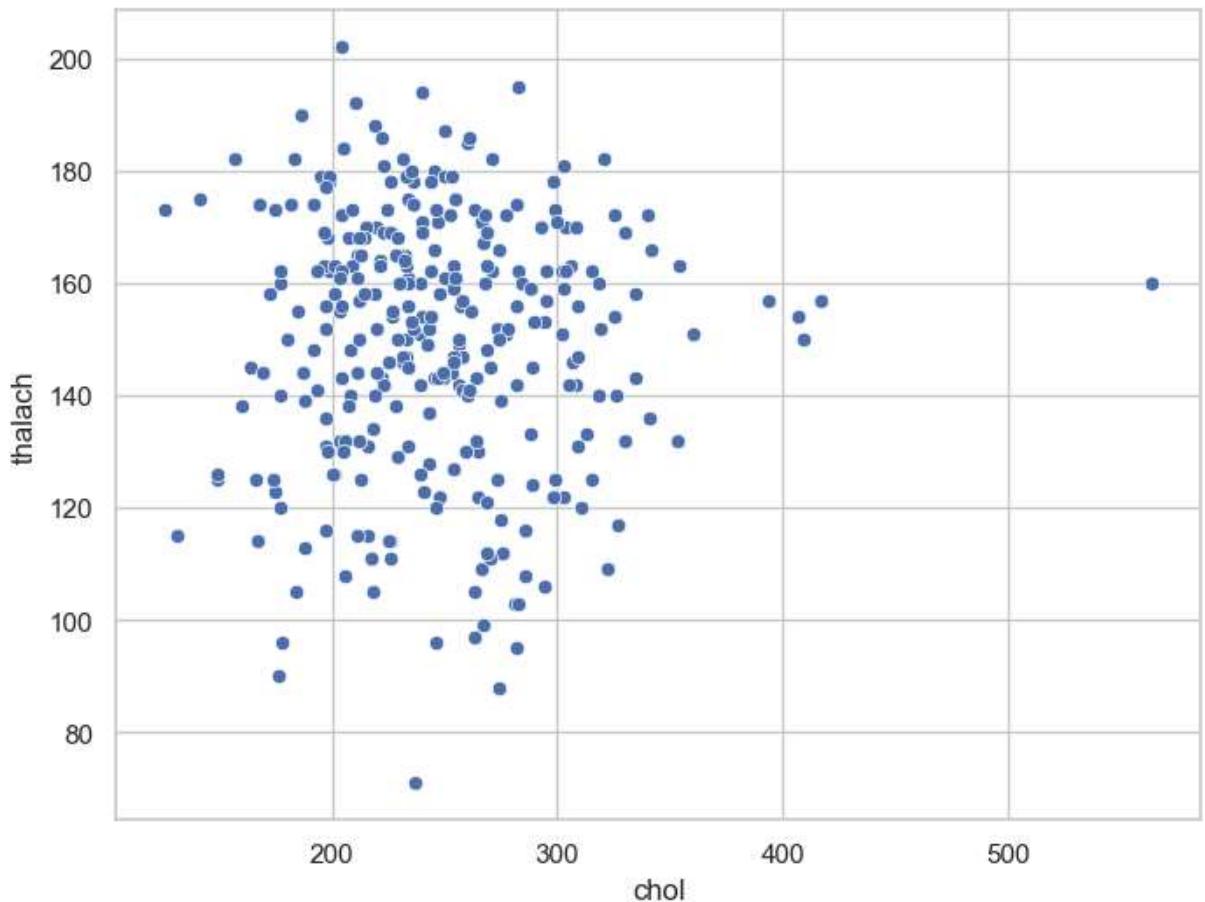
```
In [121]:  
f, ax = plt.subplots(figsize=(8, 6))  
ax = sns.scatterplot(x="age", y="trestbps", data=df)  
plt.show()
```



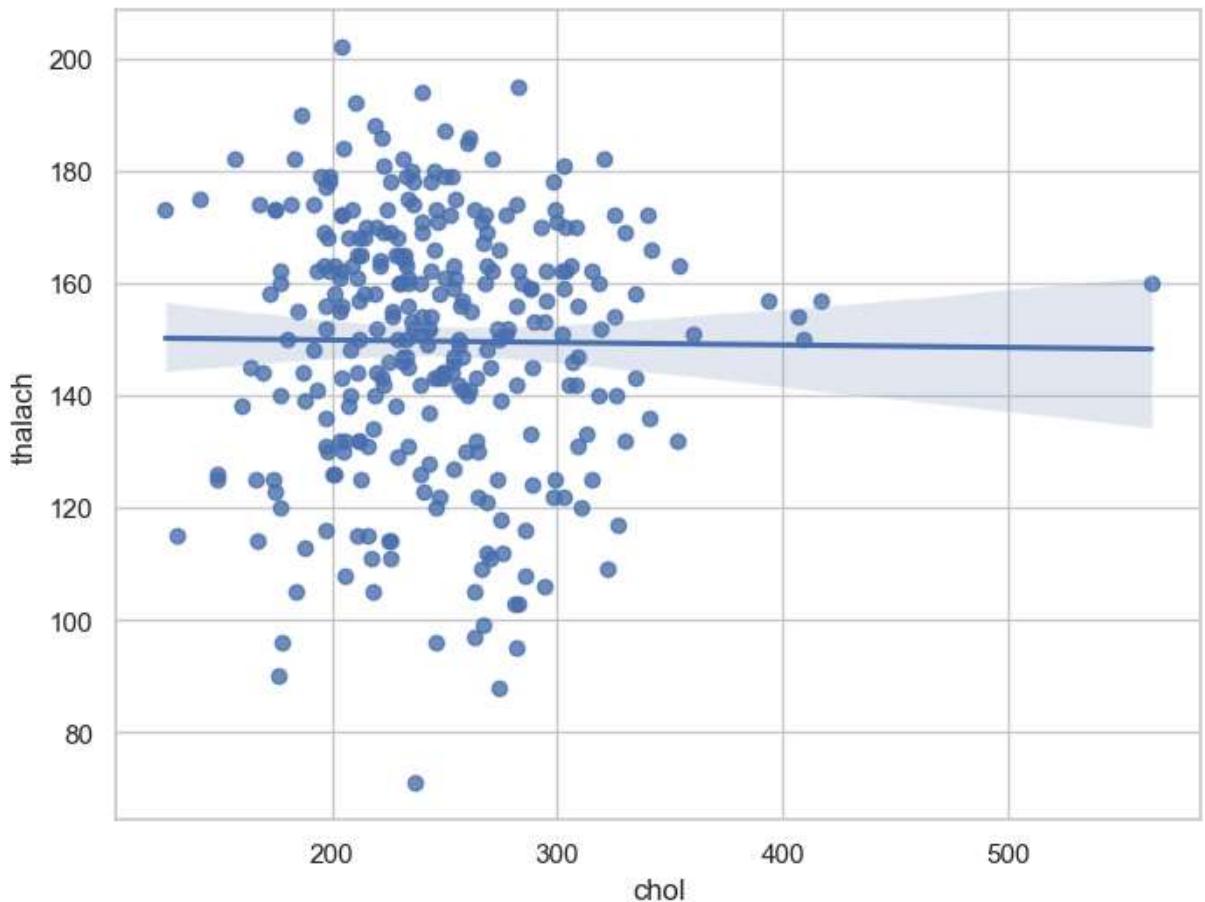
```
In [123...]:  
f, ax = plt.subplots(figsize=(8, 6))  
ax = sns.regplot(x="age", y="chol", data=df)  
plt.show()
```



```
In [125...]:  
f, ax = plt.subplots(figsize=(8, 6))  
ax = sns.scatterplot(x="chol", y = "thalach", data=df)  
plt.show()
```



```
In [127]:  
f, ax = plt.subplots(figsize=(8, 6))  
ax = sns.regplot(x="chol", y="thalach", data=df)  
plt.show()
```



```
In [129...]: df.isnull().sum()
```

```
Out[129...]: 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
```

```
In [131...]: assert pd.notnull(df).all().all()
```

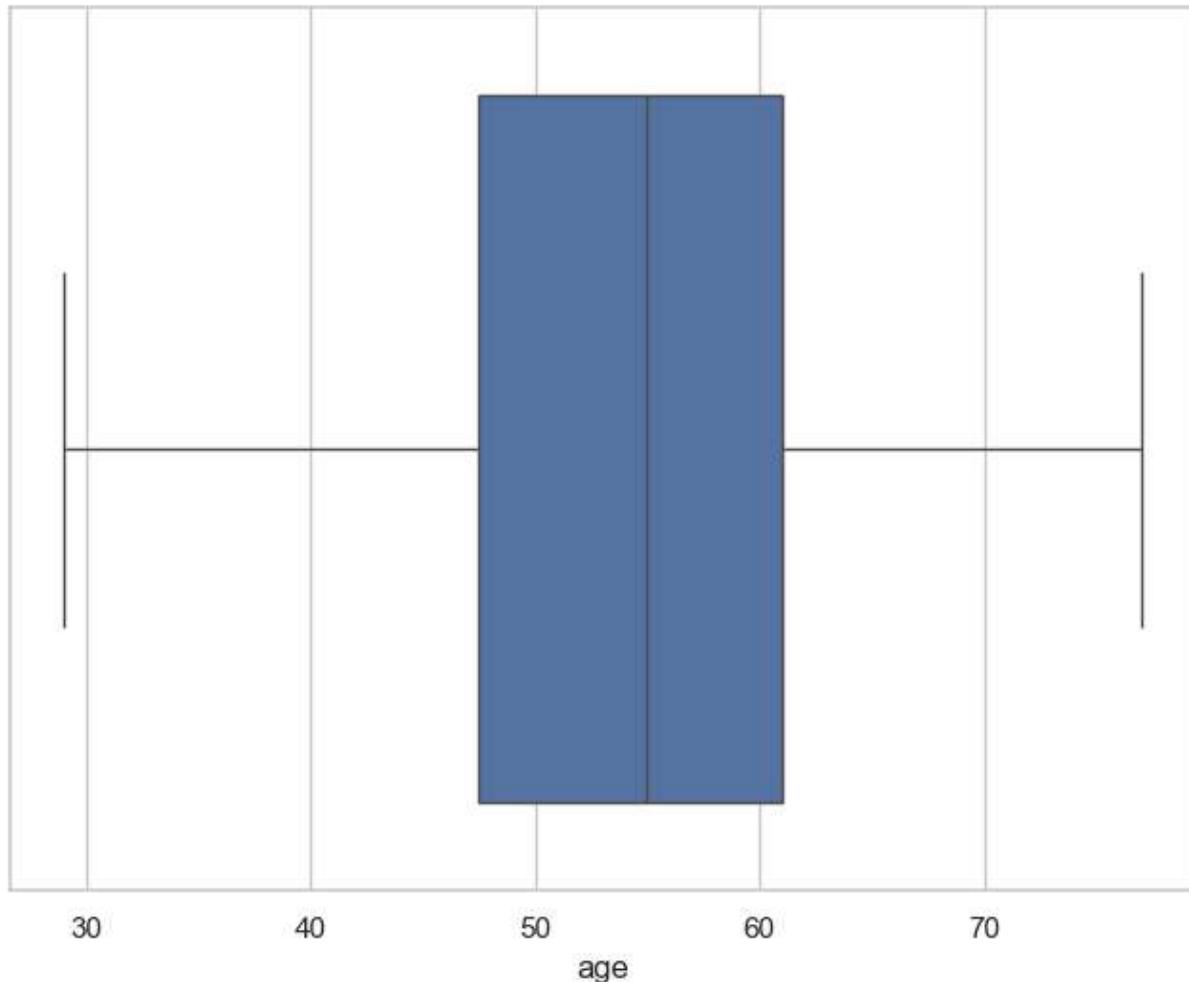
```
In [133...]: assert (df>=0).all().all()
```

outlier detection

```
In [136...]: df['age'].describe()
```

```
Out[136...]: count    303.000000  
mean      54.366337  
std       9.082101  
min      29.000000  
25%     47.500000  
50%     55.000000  
75%     61.000000  
max     77.000000  
Name: age, dtype: float64
```

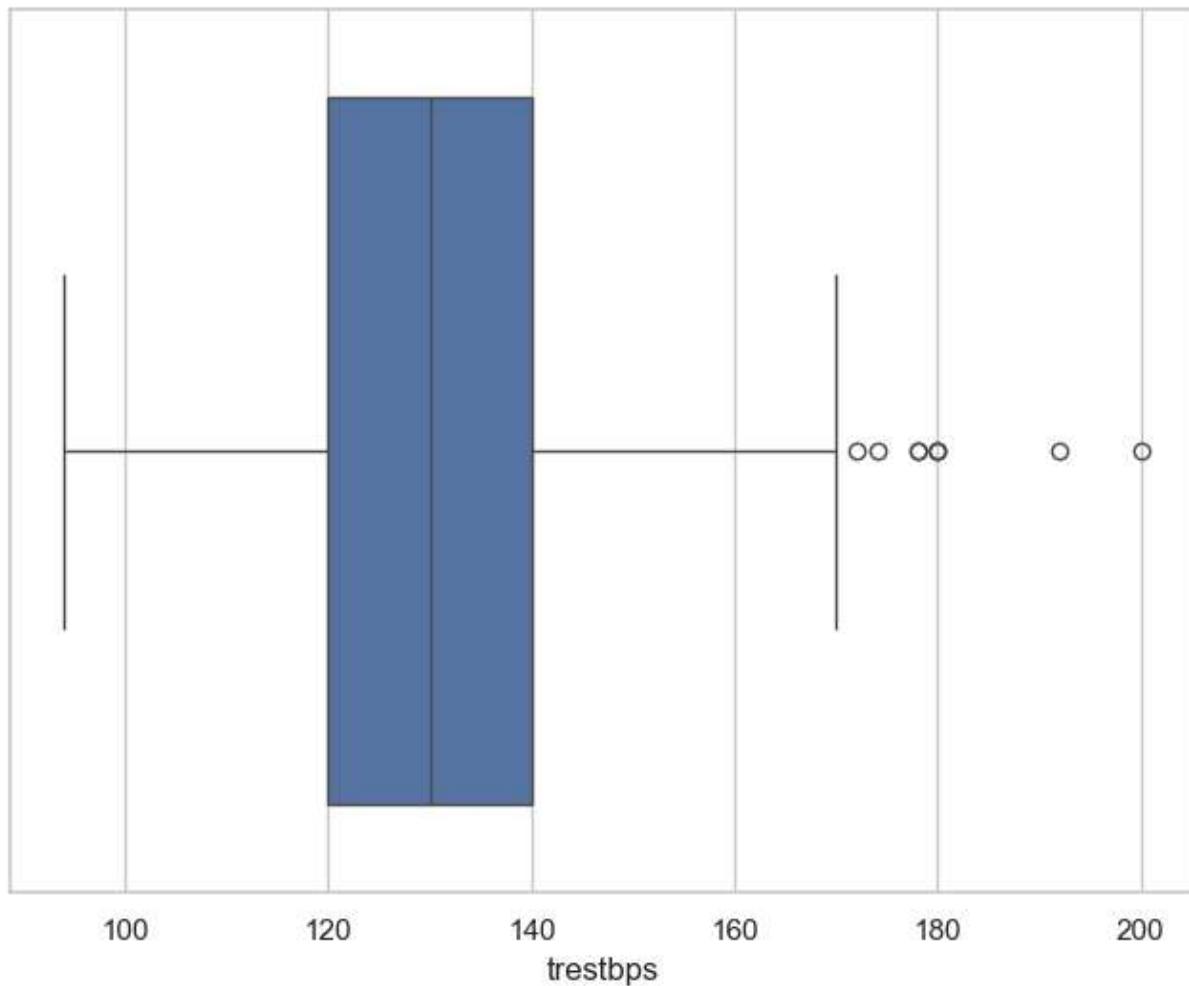
```
In [138...]: f, ax = plt.subplots(figsize=(8, 6))  
sns.boxplot(x=df["age"])  
plt.show()
```



```
In [140...]: df['trestbps'].describe()
```

```
Out[140...]: count    303.000000  
mean      131.623762  
std       17.538143  
min      94.000000  
25%     120.000000  
50%     130.000000  
75%     140.000000  
max     200.000000  
Name: trestbps, dtype: float64
```

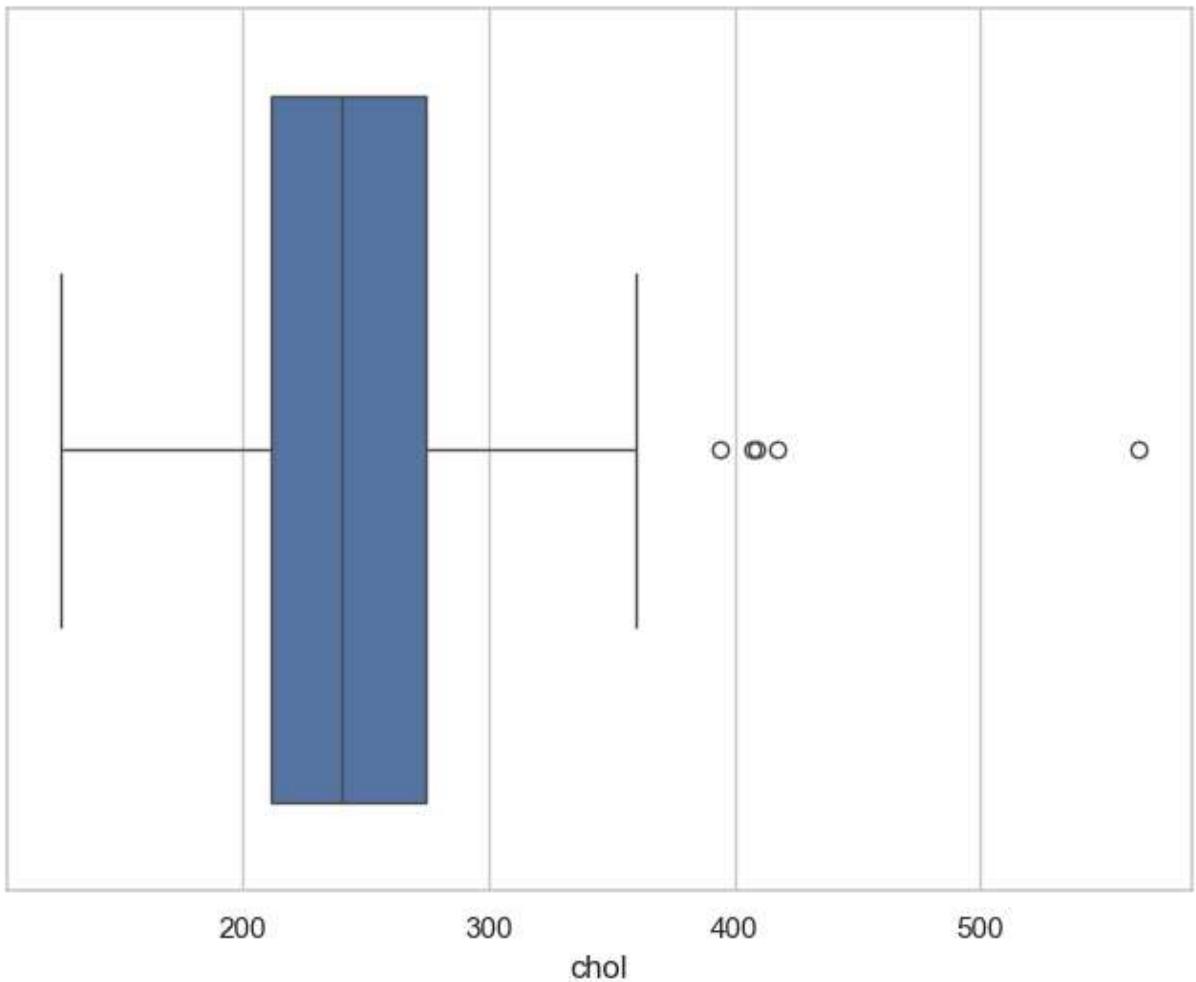
```
In [142...  
f, ax = plt.subplots(figsize=(8, 6))  
sns.boxplot(x=df["trestbps"])  
plt.show()
```



```
In [144... df['chol'].describe()
```

```
Out[144... count    303.000000  
mean     246.264026  
std      51.830751  
min     126.000000  
25%     211.000000  
50%     240.000000  
75%     274.500000  
max     564.000000  
Name: chol, dtype: float64
```

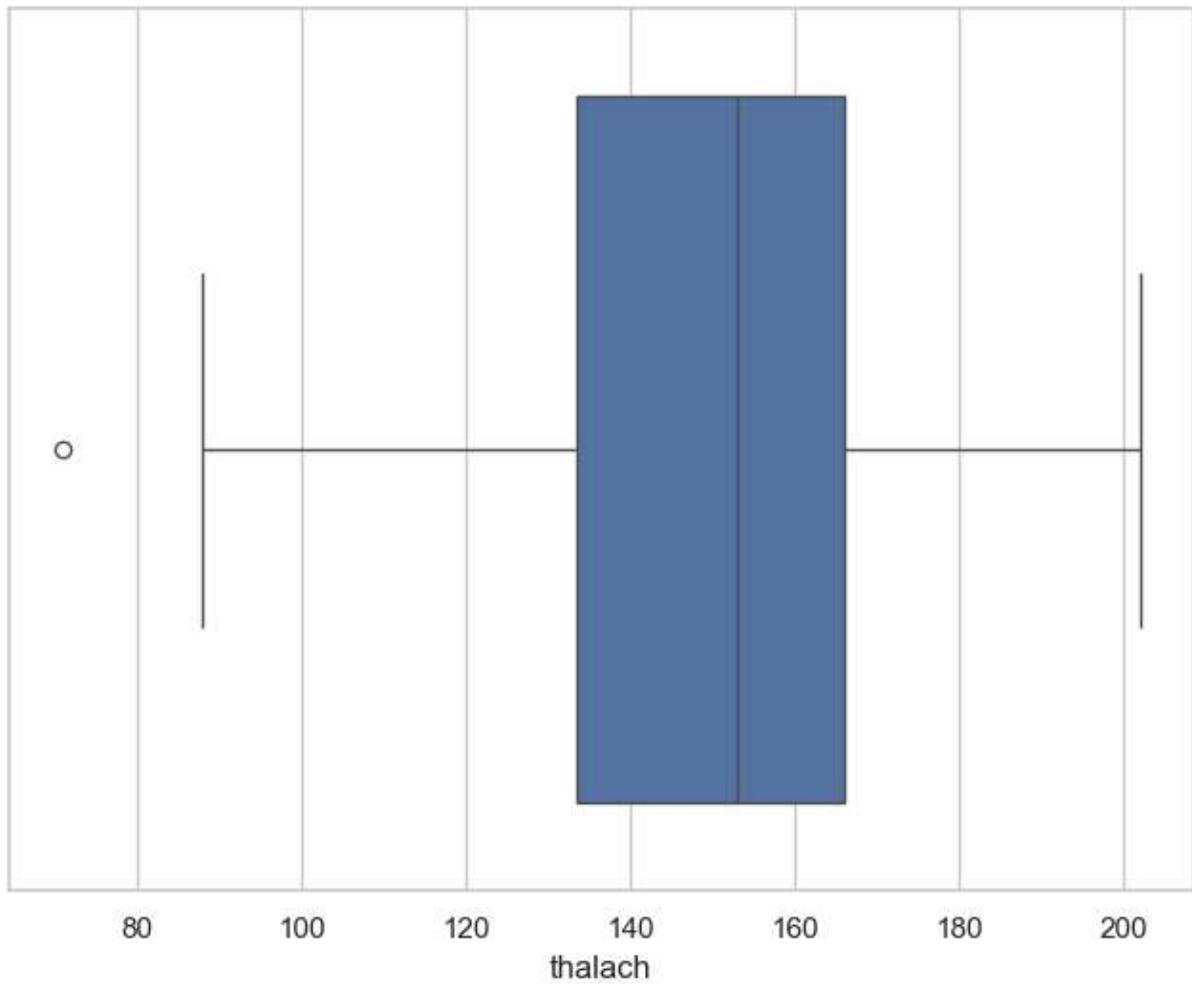
```
In [146... f, ax = plt.subplots(figsize=(8, 6))  
sns.boxplot(x=df["chol"])  
plt.show()
```



```
In [148...]: df['thalach'].describe()
```

```
Out[148...]: count    303.000000
mean      149.646865
std       22.905161
min       71.000000
25%      133.500000
50%      153.000000
75%      166.000000
max      202.000000
Name: thalach, dtype: float64
```

```
In [150...]: f, ax = plt.subplots(figsize=(8, 6))
sns.boxplot(x=df["thalach"])
plt.show()
```



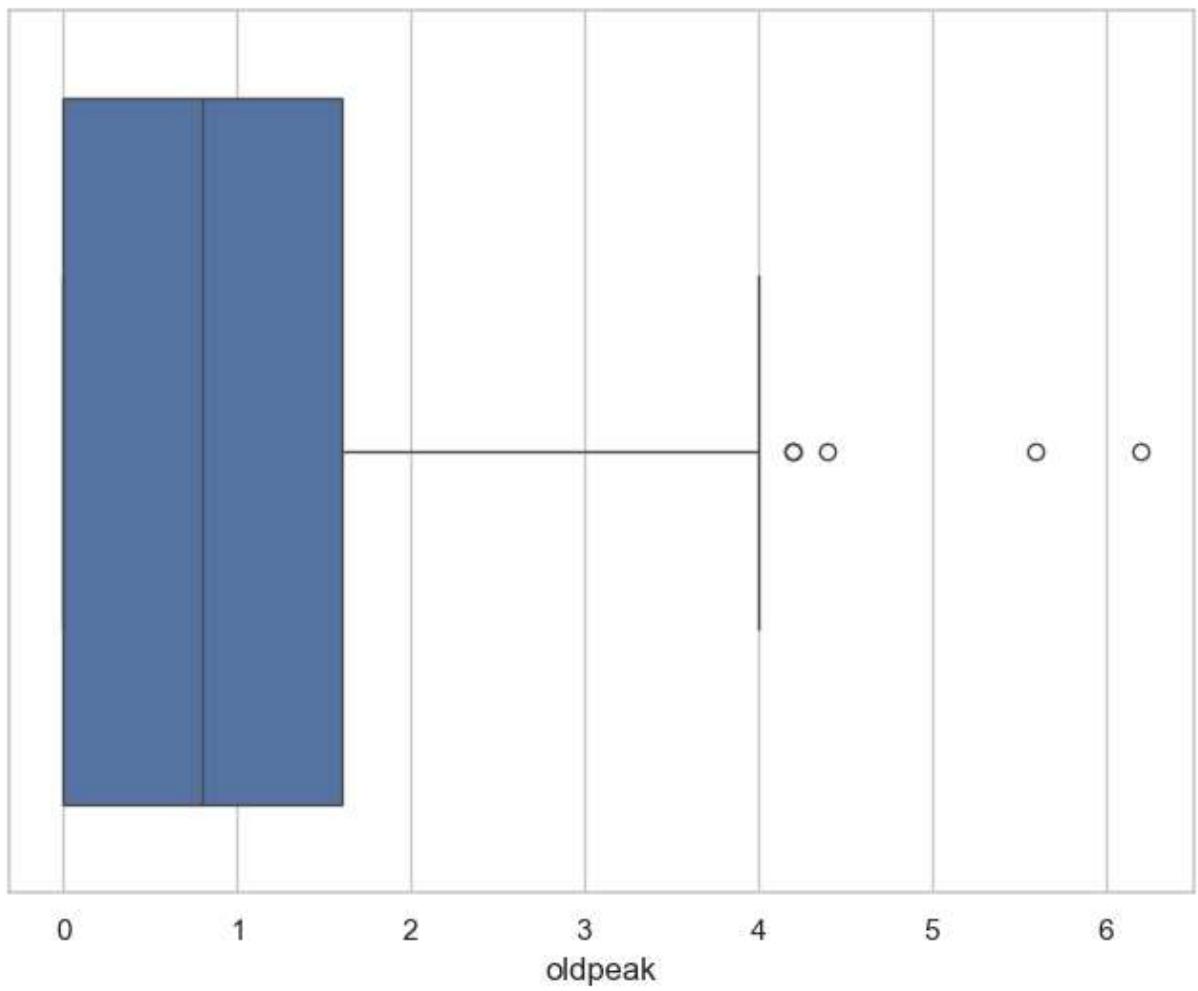
```
In [152...]: df['oldpeak'].describe()
```

```
Out[152...]:
```

	count	303.000000
mean	1.039604	
std	1.161075	
min	0.000000	
25%	0.000000	
50%	0.800000	
75%	1.600000	
max	6.200000	

Name: oldpeak, dtype: float64

```
In [154...]: f, ax = plt.subplots(figsize=(8, 6))
sns.boxplot(x=df["oldpeak"])
plt.show()
```



In []: