In [1]:
```
import numpy as np
import pandas as pd
import seaborn as sns
```

In [2]:
```
d_heart=pd.read_csv('heart.csv')
```

In [3]:
```
d_heart.head()
```

Out[3]:

	Age	Sex	ChestPainType	RestingBP	Cholesterol	FastingBS	RestingECG	MaxHR	ExerciseAngina	Oldpeak	ST_Slope	Heart
0	40	M	ATA	140	289	0	Normal	172	N	0.0	Up	
1	49	F	NAP	160	180	0	Normal	156	N	1.0	Flat	
2	37	M	ATA	130	283	0	ST	98	N	0.0	Up	
3	48	F	ASY	138	214	0	Normal	108	Y	1.5	Flat	
4	54	M	NAP	150	195	0	Normal	122	N	0.0	Up	

In [4]:
```
d_heart.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 918 entries, 0 to 917
Data columns (total 12 columns):
 #   Column          Non-Null Count  Dtype  
---  ------          --------------  -----  
 0   Age             918 non-null    int64  
 1   Sex             918 non-null    object 
 2   ChestPainType   918 non-null    object 
 3   RestingBP       918 non-null    int64  
 4   Cholesterol     918 non-null    int64  
 5   FastingBS       918 non-null    int64  
 6   RestingECG      918 non-null    object 
 7   MaxHR           918 non-null    int64  
 8   ExerciseAngina  918 non-null    object 
 9   Oldpeak         918 non-null    float64
 10  ST_Slope        918 non-null    object 
 11  HeartDisease    918 non-null    int64  
dtypes: float64(1), int64(6), object(5)
memory usage: 86.2+ KB
```

In [5]:
```
d_heart.isnull().sum()
```

Out[5]:
```
Age               0
Sex               0
ChestPainType     0
RestingBP         0
Cholesterol       0
FastingBS         0
RestingECG        0
MaxHR             0
ExerciseAngina    0
Oldpeak           0
ST_Slope          0
HeartDisease      0
dtype: int64
```

In [6]:
```
d_heart.describe()
```

Out[6]:

	Age	RestingBP	Cholesterol	FastingBS	MaxHR	Oldpeak	HeartDisease
count	918.000000	918.000000	918.000000	918.000000	918.000000	918.000000	918.000000
mean	53.510893	132.396514	198.799564	0.233115	136.809368	0.887364	0.553377
std	9.432617	18.514154	109.384145	0.423046	25.460334	1.066570	0.497414
min	28.000000	0.000000	0.000000	0.000000	60.000000	-2.600000	0.000000
25%	47.000000	120.000000	173.250000	0.000000	120.000000	0.000000	0.000000
50%	54.000000	130.000000	223.000000	0.000000	138.000000	0.600000	1.000000
75%	60.000000	140.000000	267.000000	0.000000	156.000000	1.500000	1.000000
max	77.000000	200.000000	603.000000	1.000000	202.000000	6.200000	1.000000

In [7]:
```
1 d_heart.shape
```

Out[7]:
```
(918, 12)
```

In [8]:
```
1 d_heart.nunique()
```

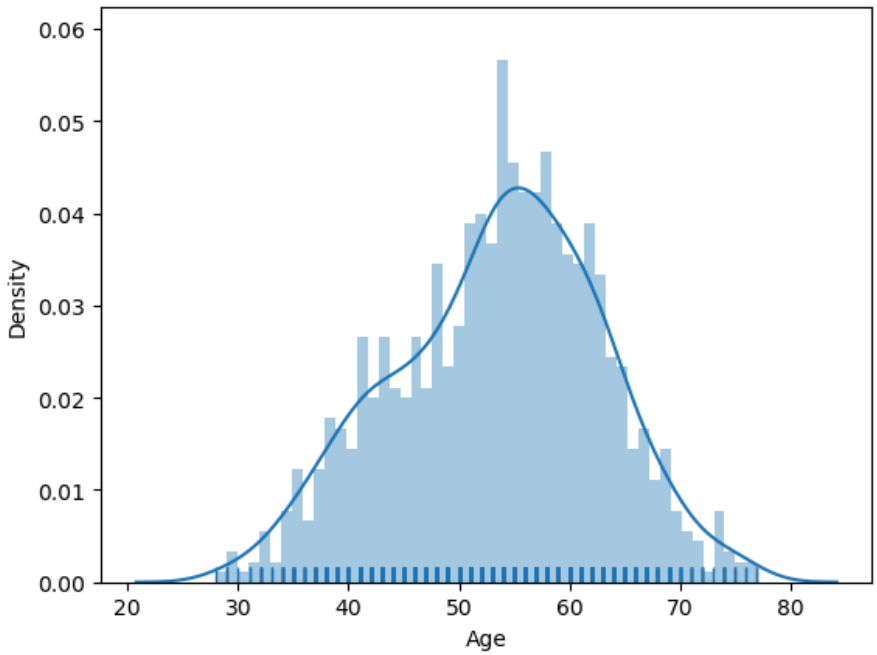
Out[8]:
```
Age               50
Sex                2
ChestPainType      4
RestingBP         67
Cholesterol      222
FastingBS          2
RestingECG         3
MaxHR            119
ExerciseAngina     2
Oldpeak           53
ST_Slope           3
HeartDisease       2
dtype: int64
```

Distribution Visualizations for Numerical Data

In [29]:
```
1 sns.distplot(d_heart['Age'], rug=True ,bins=50)
```

```
E:\AnacondaDATASCIENCE\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated function and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
E:\AnacondaDATASCIENCE\lib\site-packages\seaborn\distributions.py:2103: FutureWarning: The `axis` variable is no longer used and will be removed. Instead, assign variables directly to `x` or `y`.
  warnings.warn(msg, FutureWarning)
```

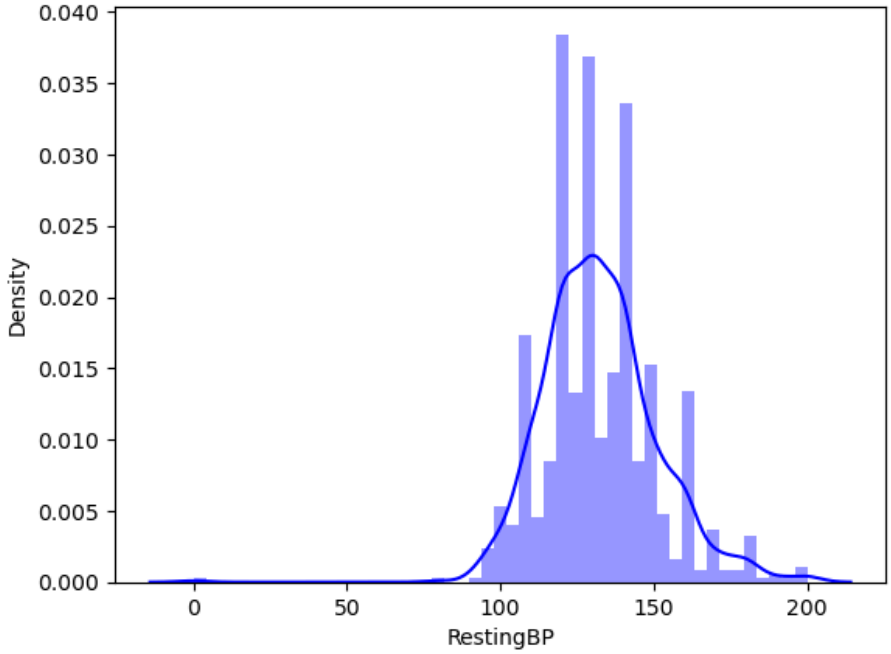
Out[29]:
```
<AxesSubplot:xlabel='Age', ylabel='Density'>
```



In [31]:
```
1 sns.distplot(d_heart['RestingBP'] , color='blue')
```

```
E:\AnacondaDATASCIENCE\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a
deprecated function and will be removed in a future version. Please adapt your code to use either `disp
lot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histo
grams).
  warnings.warn(msg, FutureWarning)
```

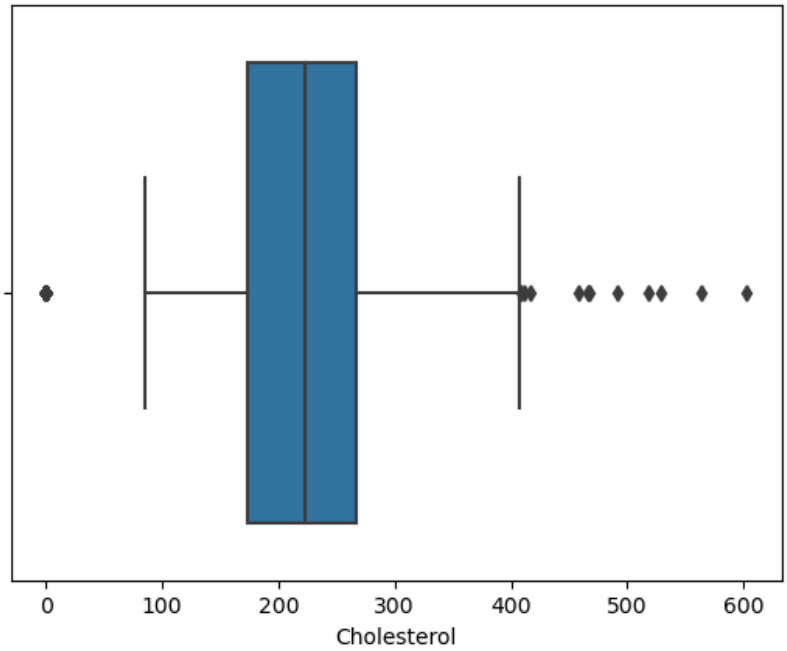
Out[31]: <AxesSubplot:xlabel='RestingBP', ylabel='Density'>



BOX PLOT

In [11]:
```
1 sns.boxplot(data=d_heart ,x= 'Cholesterol')
```

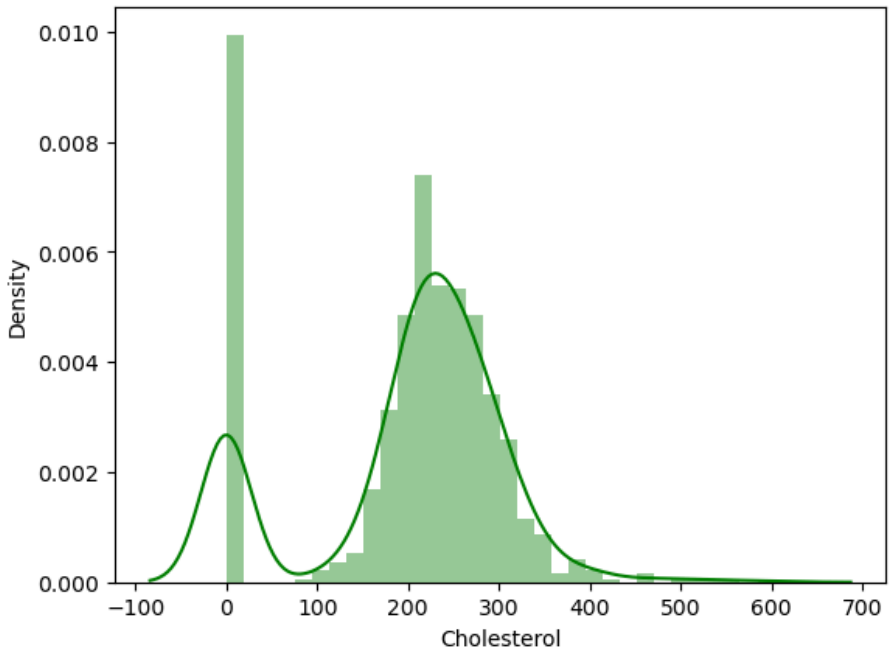
Out[11]: <AxesSubplot:xlabel='Cholesterol'>



In [34]:
```
1 sns.distplot(d_heart['Cholesterol'] , color='green')
```

```
E:\AnacondaDATASCIENCE\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a
deprecated function and will be removed in a future version. Please adapt your code to use either `disp
lot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histo
grams).
  warnings.warn(msg, FutureWarning)
```

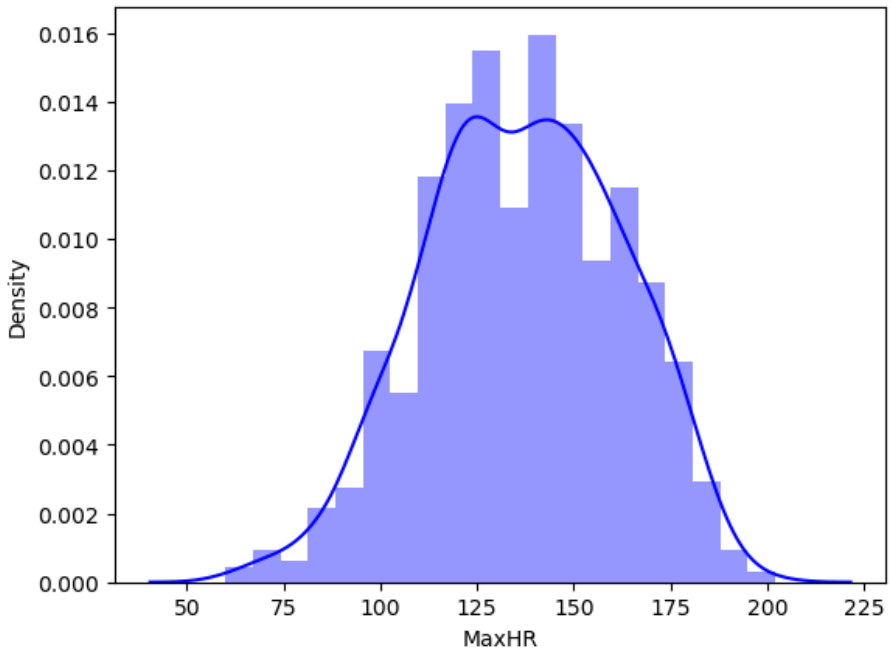
Out[34]: <AxesSubplot:xlabel='Cholesterol', ylabel='Density'>



In [13]:
```
1 sns.distplot(d_heart['MaxHR'] , color='blue')
```

```
E:\AnacondaDATASCIENCE\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a
deprecated function and will be removed in a future version. Please adapt your code to use either `disp
lot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histo
grams).
  warnings.warn(msg, FutureWarning)
```

Out[13]: <AxesSubplot:xlabel='MaxHR', ylabel='Density'>



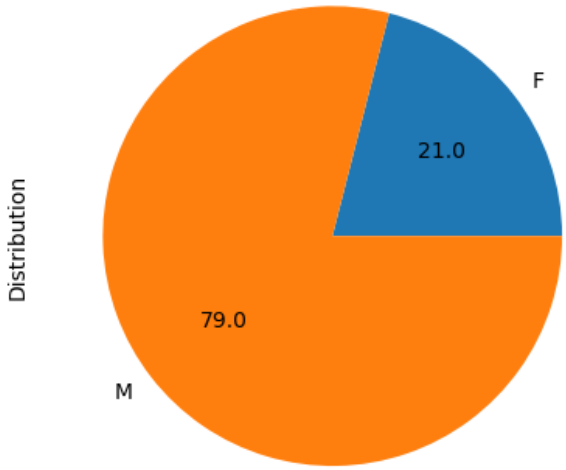
Pie Chart Representation for Categorical Data

In [14]:
```
1 d_heart.groupby('Sex').size()
```

Out[14]:
```
Sex
F    193
M    725
dtype: int64
```

In [40]:
```
1 d_heart.groupby('Sex').size().plot(kind='pie' , autopct='%0.1f', ylabel ="Distribution")
```

Out[40]: <AxesSubplot:ylabel='Distribution'>

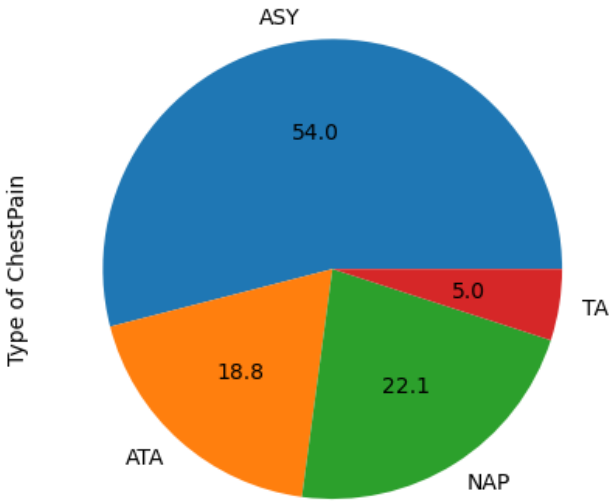


In [16]:
```
1 d_heart.groupby('ChestPainType').size()
```

Out[16]:
```
ChestPainType
ASY    496
ATA    173
NAP    203
TA      46
dtype: int64
```

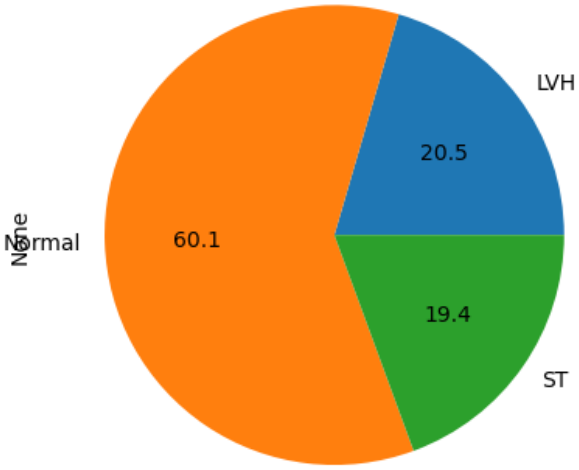
In [41]:
```
1 d_heart.groupby('ChestPainType').size().plot(kind='pie' , autopct='%0.1f', ylabel ="Type of ChestPai
```

Out[41]: <AxesSubplot:ylabel='Type of ChestPain'>



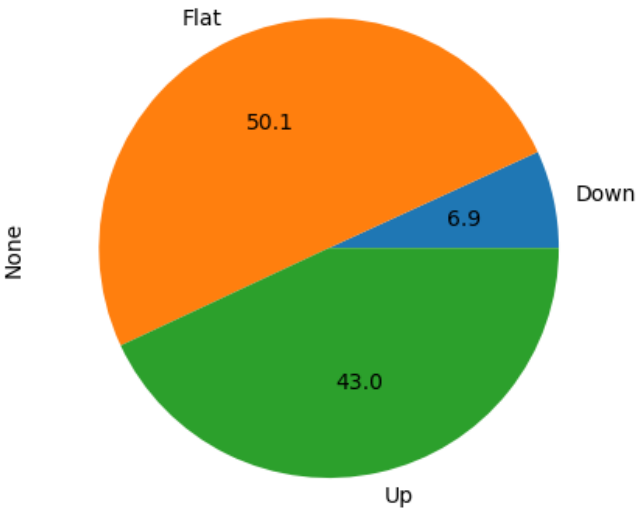
In [18]:
```
1 d_heart.groupby('RestingECG').size().plot(kind='pie' , autopct='%0.1f')
```

Out[18]: <AxesSubplot:ylabel='None'>



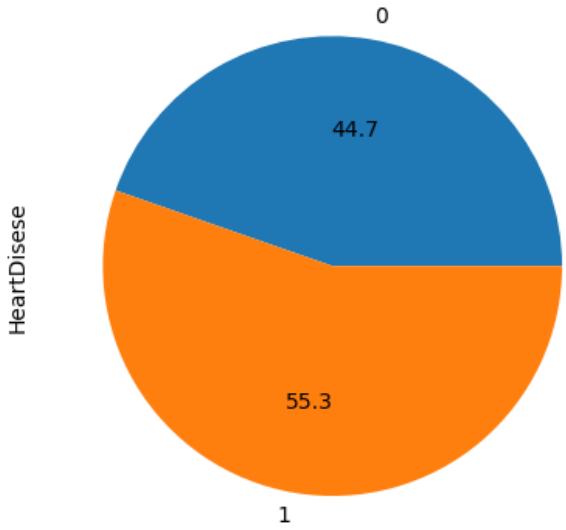
In [19]:
```
1 d_heart.groupby('ST_Slope').size().plot(kind='pie' , autopct='%0.1f')
```

Out[19]: <AxesSubplot:ylabel='None'>



In [42]:
```
1 d_heart.groupby('HeartDisease').size().plot(kind='pie' , autopct='%0.1f',ylabel= "HeartDisese")
```

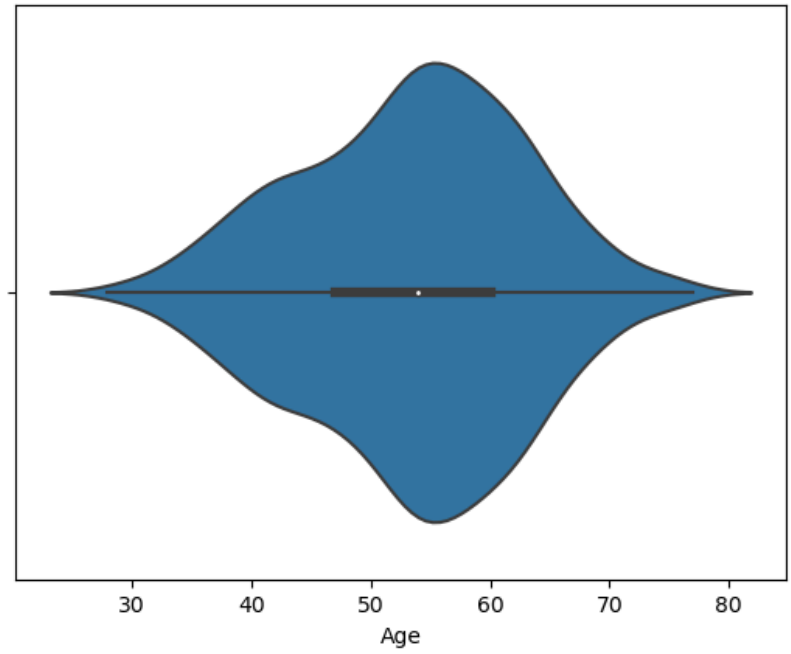
Out[42]: <AxesSubplot:ylabel='HeartDisese'>



Violin plot

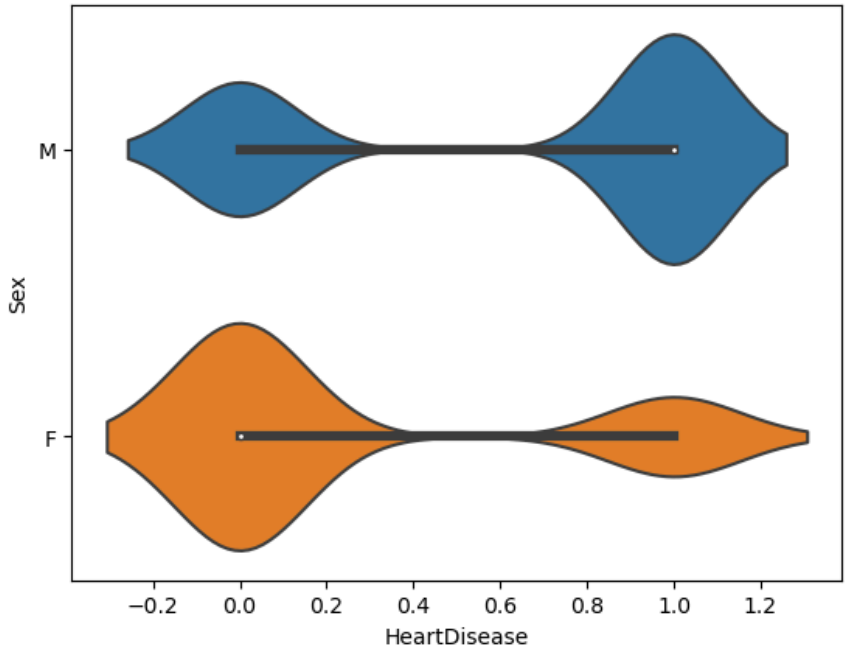
In [21]:
```
1 sns.violinplot(data=d_heart, x='Age')
```

Out[21]: <AxesSubplot:xlabel='Age'>



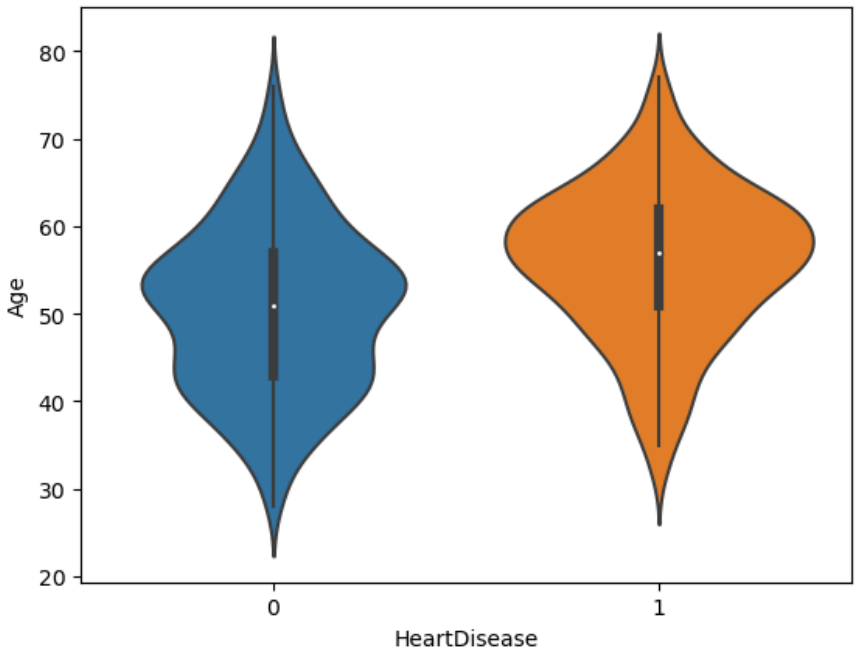
In [22]:
```
1 sns.violinplot(data=d_heart, y='Sex' , x='HeartDisease')
```

Out[22]: <AxesSubplot:xlabel='HeartDisease', ylabel='Sex'>



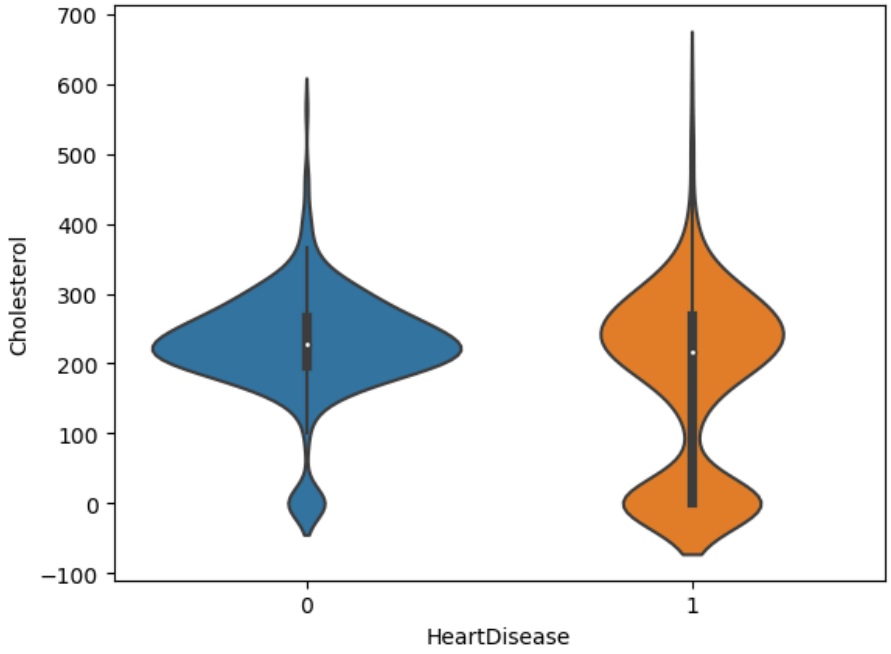
In [23]:
```
1 sns.violinplot(data=d_heart, y='Age' , x='HeartDisease')
```

Out[23]: <AxesSubplot:xlabel='HeartDisease', ylabel='Age'>



In [24]:
```
1 sns.violinplot(data=d_heart, y='Cholesterol' , x='HeartDisease')
```

Out[24]: <AxesSubplot:xlabel='HeartDisease', ylabel='Cholesterol'>



Correlation

In [25]:
```
1 d_heart.head(2)
```

Out[25]:

	Age	Sex	ChestPainType	RestingBP	Cholesterol	FastingBS	RestingECG	MaxHR	ExerciseAngina	Oldpeak	ST_Slope	Heart
0	40	M	ATA	140	289	0	Normal	172	N	0.0	Up	
1	49	F	NAP	160	180	0	Normal	156	N	1.0	Flat	

Heatmap

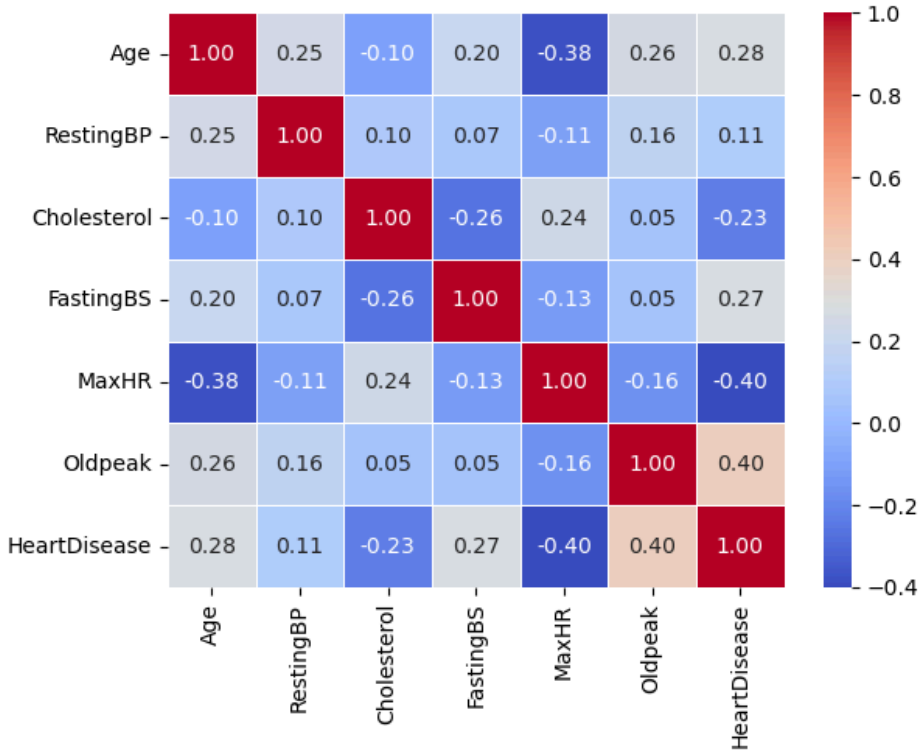
In [45]:
```
1 numeric_columns = d_heart.select_dtypes(include=[int, float])
2 cor_mat = numeric_columns.corr()
3 print(cor_mat)
4
```

```
                   Age  RestingBP  Cholesterol  FastingBS     MaxHR   Oldpeak  \
Age           1.000000   0.254399    -0.095282   0.198039 -0.382045  0.258612
RestingBP     0.254399   1.000000     0.100893   0.070193 -0.112135  0.164803
Cholesterol  -0.095282   0.100893     1.000000  -0.260974  0.235792  0.050148
FastingBS     0.198039   0.070193    -0.260974   1.000000 -0.131438  0.052698
MaxHR        -0.382045  -0.112135     0.235792  -0.131438  1.000000 -0.160691
Oldpeak       0.258612   0.164803     0.050148   0.052698 -0.160691  1.000000
HeartDisease  0.282039   0.107589    -0.232741   0.267291 -0.400421  0.403951

              HeartDisease
Age               0.282039
RestingBP         0.107589
Cholesterol      -0.232741
FastingBS         0.267291
MaxHR            -0.400421
Oldpeak           0.403951
HeartDisease      1.000000
```

```
In [46]: 1 sns.heatmap(cor_mat, cmap='coolwarm', annot=True, fmt=".2f", linewidths=.5)
```

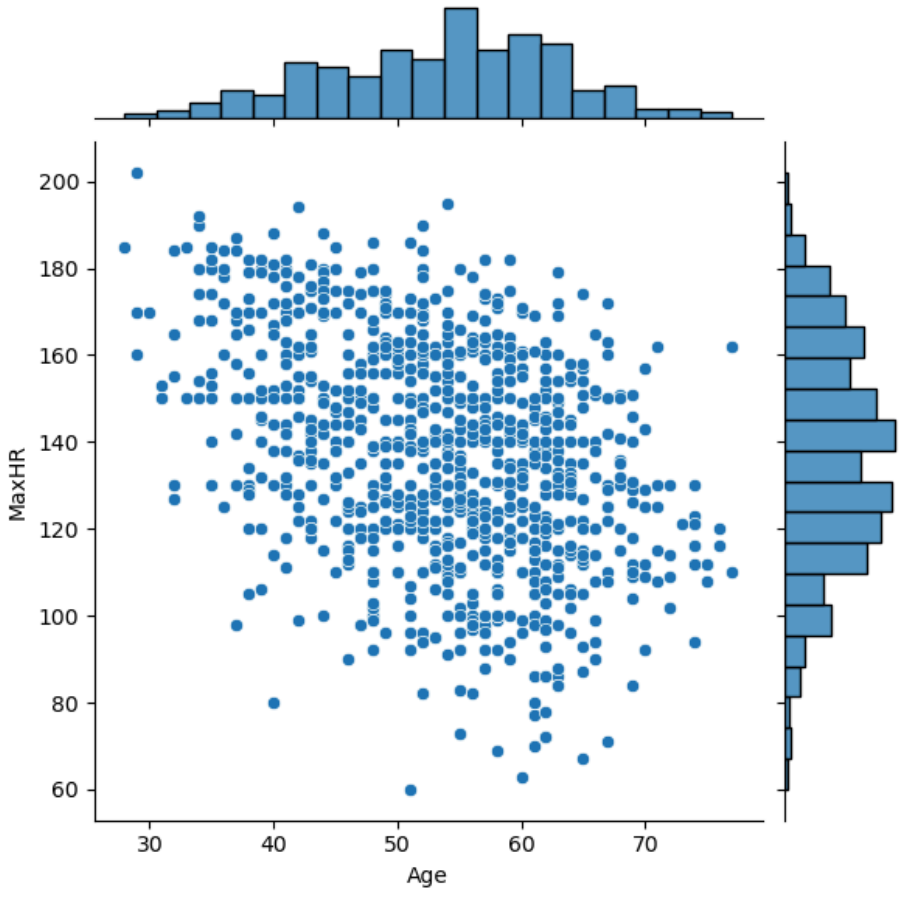
```
Out[46]: <AxesSubplot:>
```



Data Distribution on Joint Plot

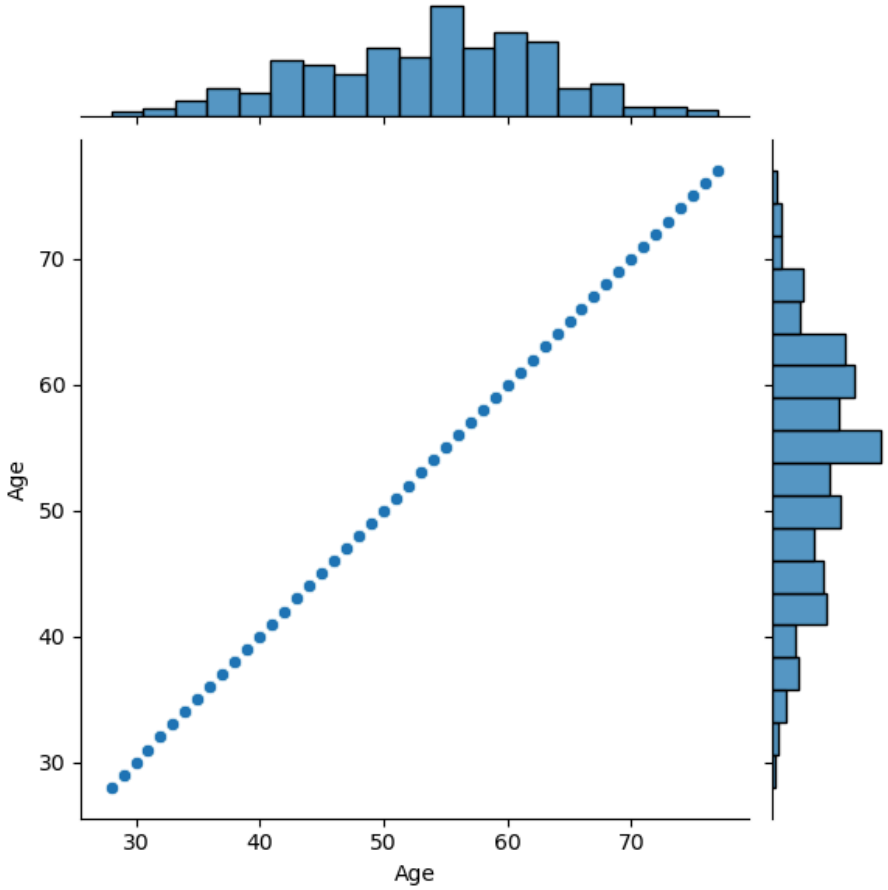
```
In [47]: 1 sns.jointplot(data=d_heart , x='Age' , y='MaxHR')
```

```
Out[47]: <seaborn.axisgrid.JointGrid at 0x194cfe345e0>
```



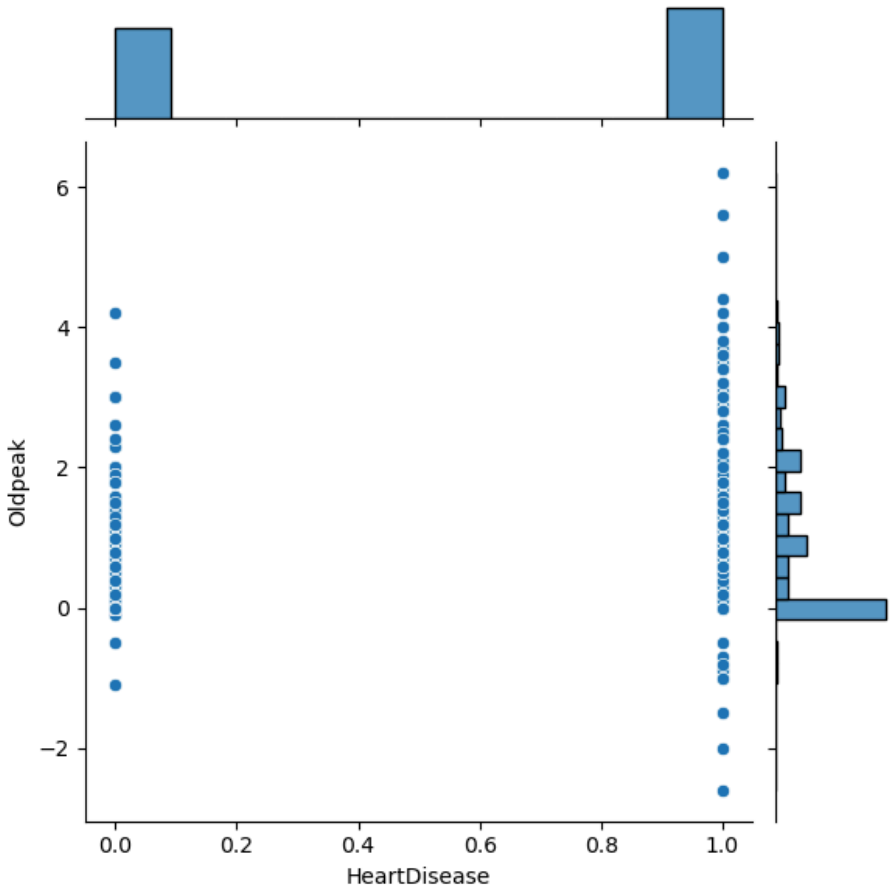
In [48]:
```
1 sns.jointplot(data=d_heart , x='Age' , y='Age')
```

Out[48]: <seaborn.axisgrid.JointGrid at 0x194cfe8c400>



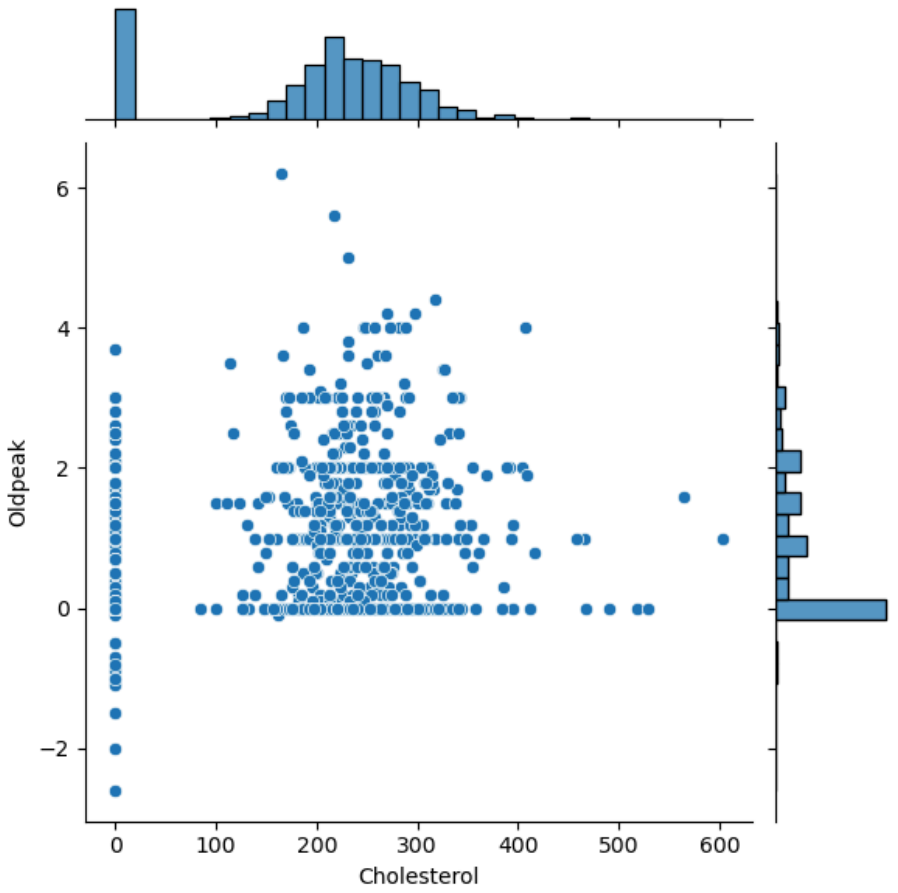
In [49]:
```
1 sns.jointplot(data=d_heart , x='HeartDisease' , y='Oldpeak')
```

Out[49]: <seaborn.axisgrid.JointGrid at 0x194cfe7e2b0>



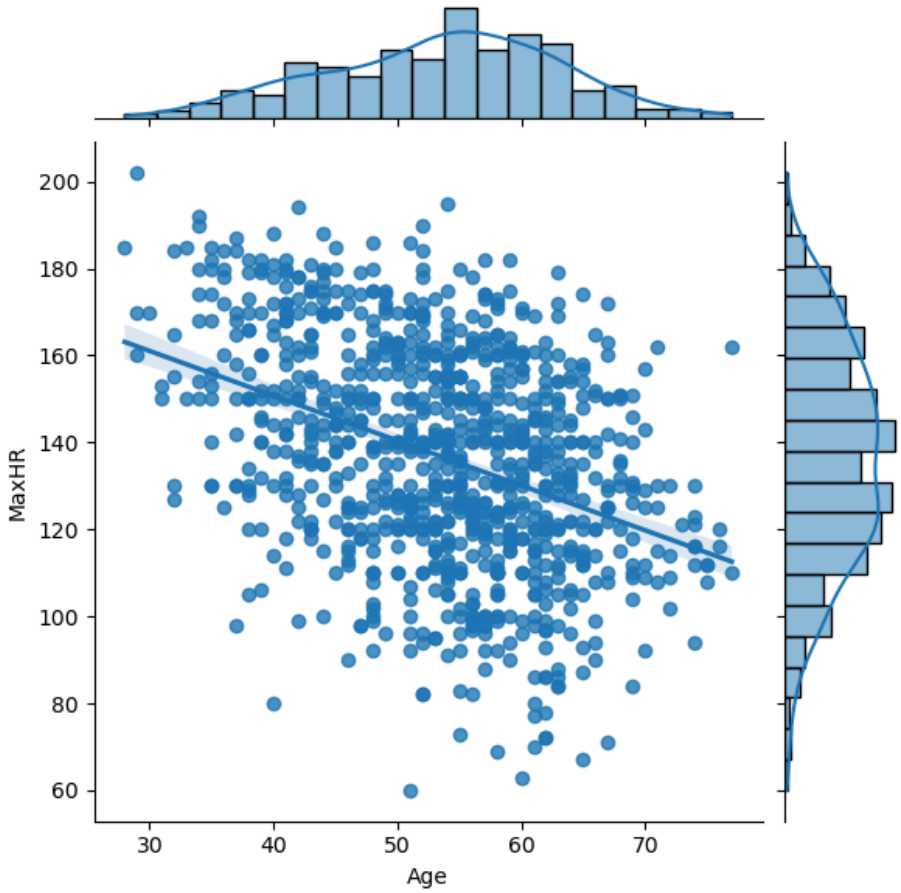
In [50]:
```
1 sns.jointplot(data=d_heart , x='Cholesterol' , y='Oldpeak')
```

Out[50]: <seaborn.axisgrid.JointGrid at 0x194d0b23130>



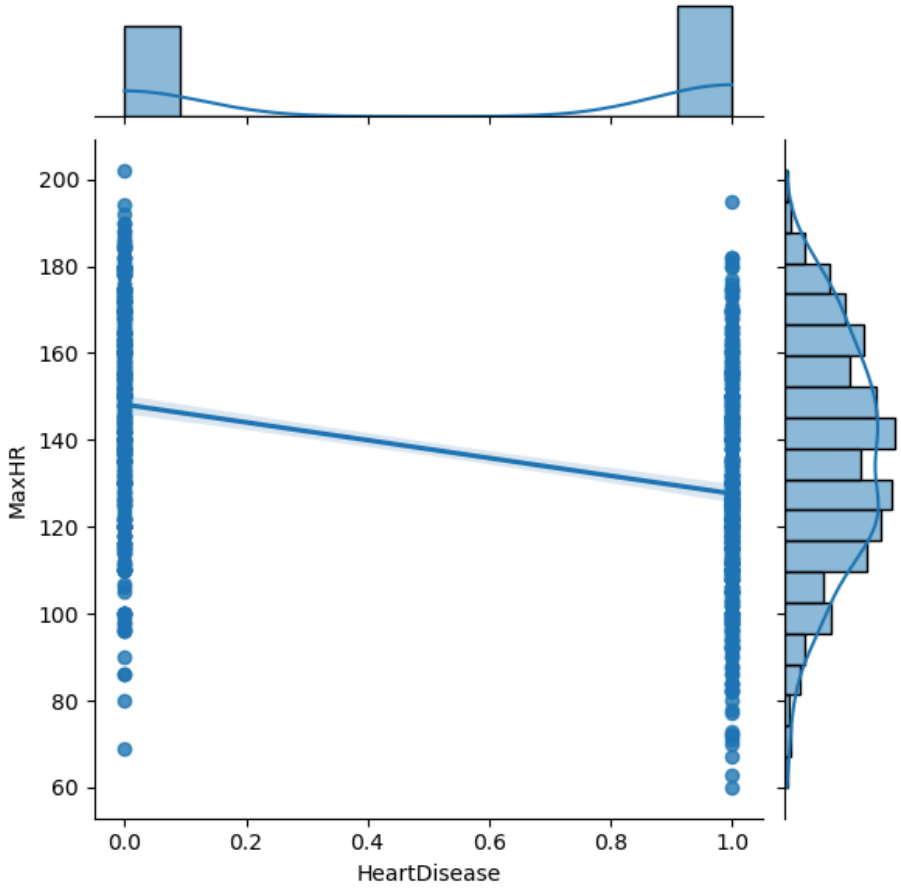
In [51]:
```
1 sns.jointplot(data=d_heart , x='Age' , y='MaxHR' , kind='reg')
```

Out[51]: <seaborn.axisgrid.JointGrid at 0x194d07493d0>



In [52]:
```
1 sns.jointplot(data=d_heart , y='MaxHR' , x='HeartDisease' , kind='reg')
```

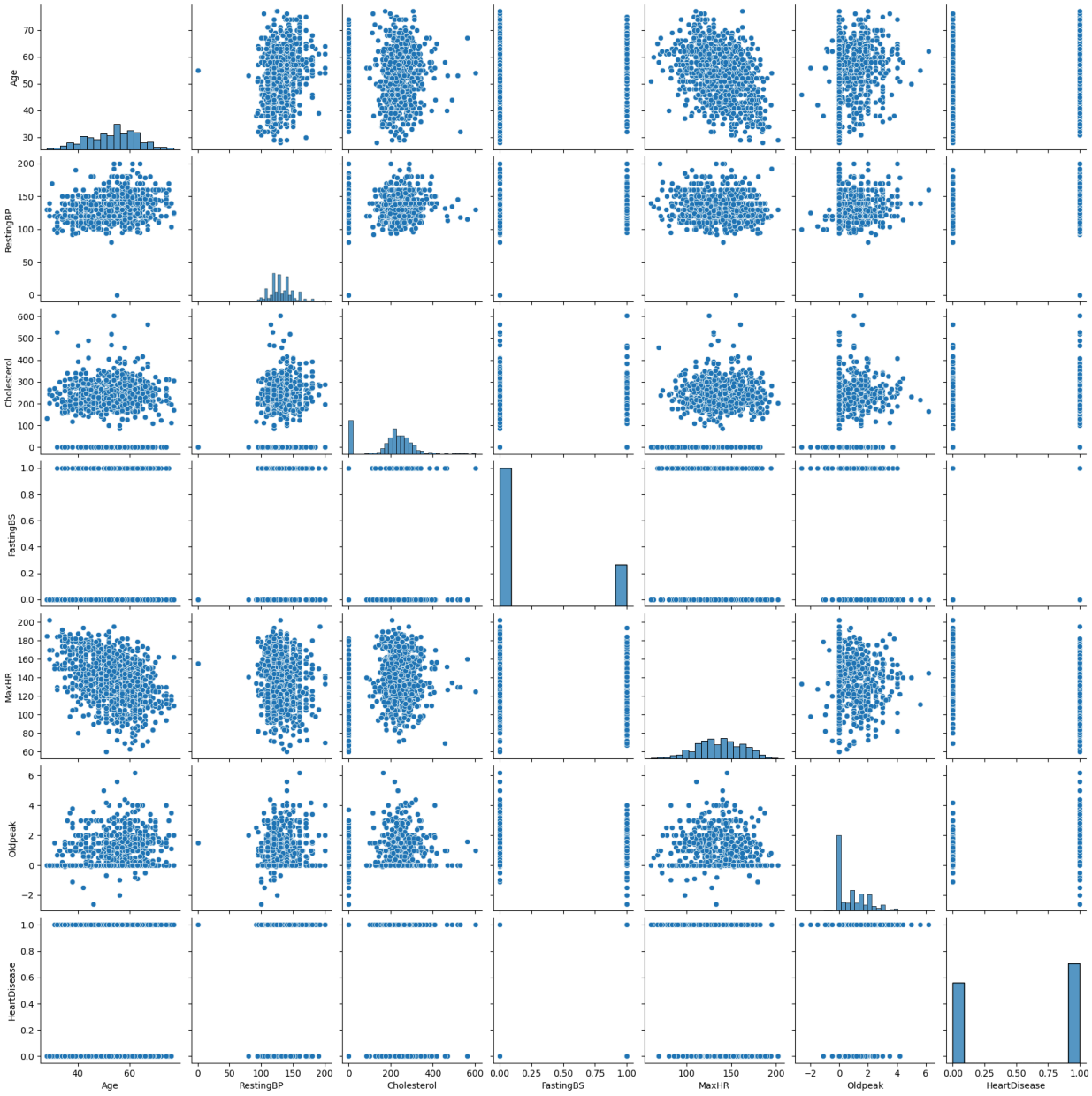
Out[52]: <seaborn.axisgrid.JointGrid at 0x194d0750f40>



Pairplot

In [53]:
```
1 sns.pairplot(d_heart)
```

Out[53]:
```
<seaborn.axisgrid.PairGrid at 0x194d2399940>
```



In [ ]:
```
1
```