In [2]:

```
#importing various packages
import pandas as pd
import numpy as np
import seaborn as sns
```

In [2]:

```
#reading the dataset
df=pd.read_csv('/content/Heart Disease data.csv')
```

In [3]:

```
#checking the data
df.head()
```

Out[3]:

	Sl_No	Age	Gender	CP	RBP(Diastolic)	SC	FBS	RER	MHRA	EIE	Old peak	Slope	Flourosopy	Thallium Test	HD
0	1	52	Male	0	125	212	0	1	168	0	1.0	2	2	2	0
1	2	53	Male	0	140	203	1	0	155	1	3.1	0	0	2	0
2	3	70	Male	0	145	174	0	1	125	1	2.6	0	0	2	0
3	4	61	Male	0	148	203	0	1	161	0	0.0	2	1	2	0
4	5	62	Female	0	138	294	1	1	106	0	1.9	1	3	2	0

In [4]:

```
#checking null values
df.isna().sum()
```

Out[4]:

```
Sl_No              0
Age                0
Gender             0
CP                 0
RBP(Diastolic)     0
SC                 0
FBS                0
```

```
RER               0
MHRA              0
EIE               0
Oldpeak           0
Slope             0
Flourosopy        0
Thallium Test     0
HD                0
dtype: int64
```

In [5]:

```
#There are no null values in any of the columns
```

In [6]:

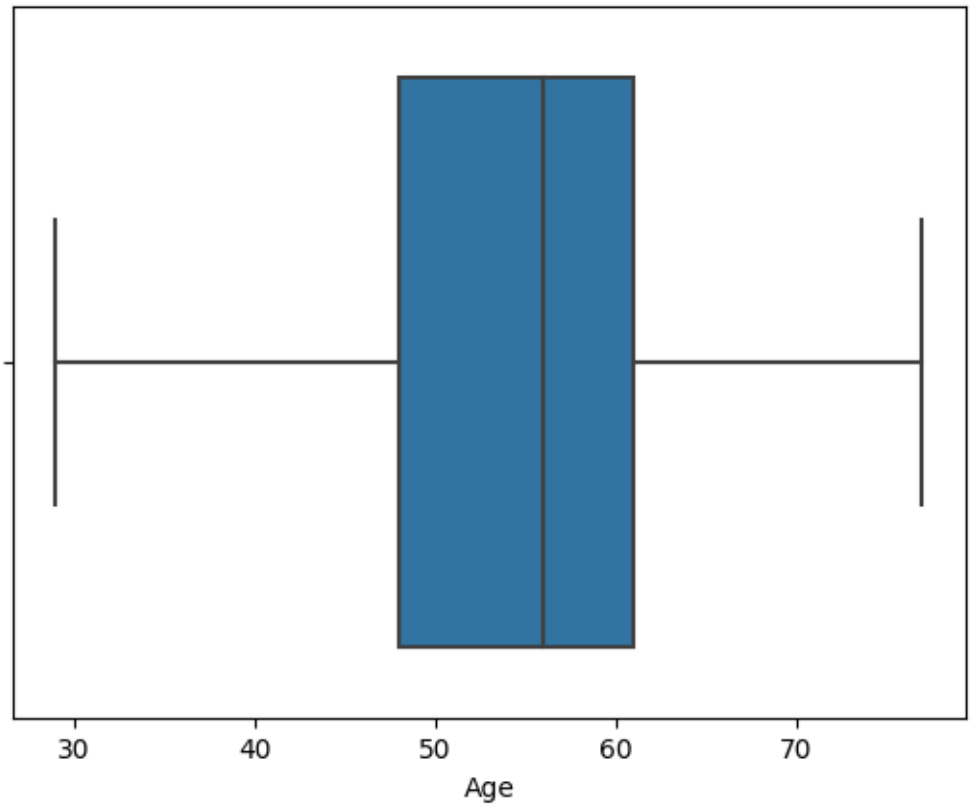
```
#now we will check for outliers
```

In [7]:

```
#checking outliers in Age Column
sns.boxplot(data=df,x='Age')
```

Out[7]:

```
<AxesSubplot:xlabel='Age'>
```



In [8]:

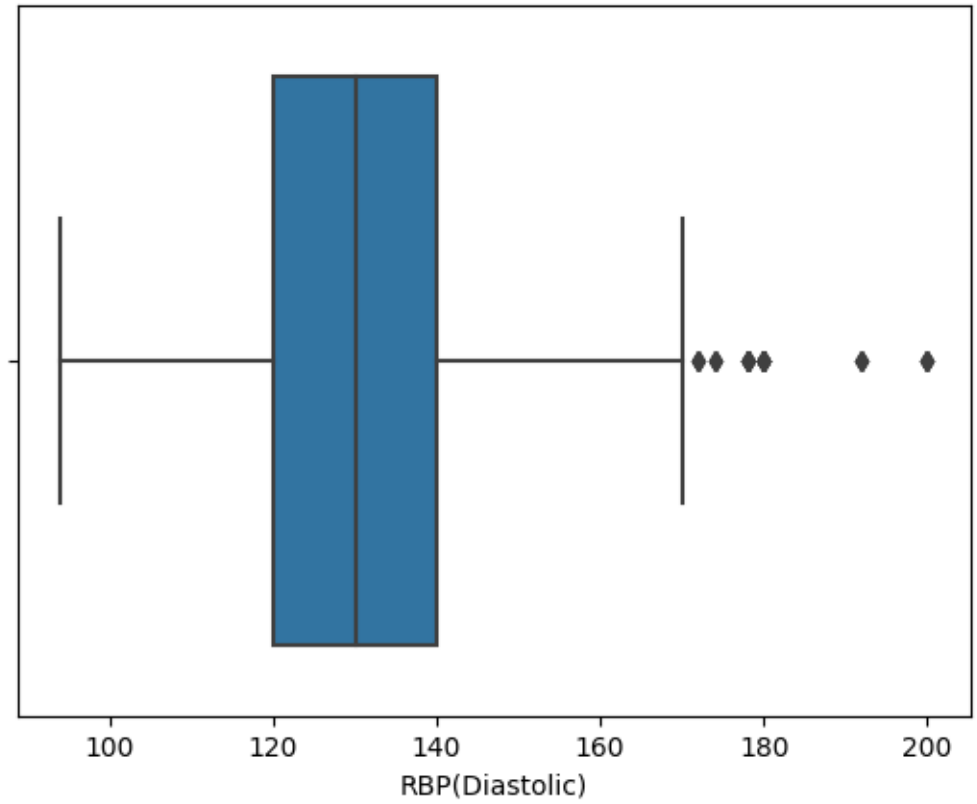
```
#there are no outliers in Age column
```

In [9]:

```
#checking outliers in RBP(Diastolic) Column
sns.boxplot(data=df,x='RBP(Diastolic)')
```

Out[9]:

```
<AxesSubplot:xlabel='RBP(Diastolic)'>
```



In [10]:

```
#we can see there are some outliers in RBP(Diastolic) column. But we will
not remove them.
#We will replace those outliers with median value
```

In [11]:

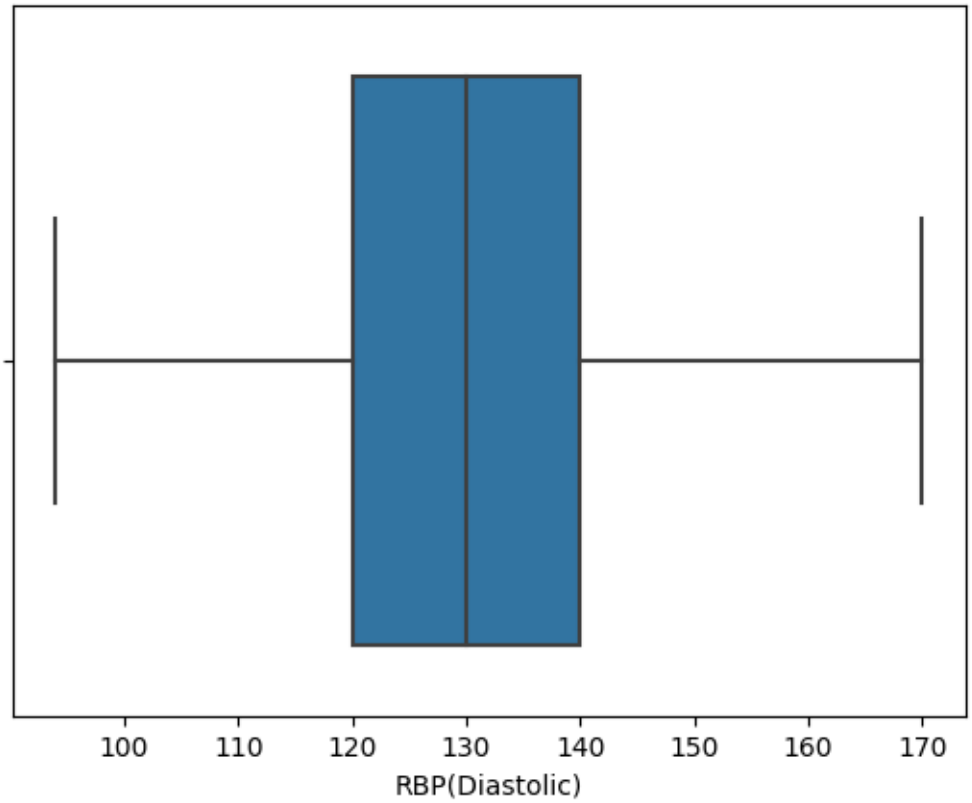
```
# Calculating the Interquartile Range (IQR)
q1 = df["RBP(Diastolic)"].quantile(0.25)
q3 = df["RBP(Diastolic)"].quantile(0.75)
iqr = q3 - q1

# Defining the lower and upper bounds for outliers
lower_bound = q1 - 1.5 * iqr
upper_bound = q3 + 1.5 * iqr

# Identifying outliers
outliers = (df["RBP(Diastolic)"] < lower_bound) | (df["RBP(Diastolic)"] >
upper_bound)

# Replace outliers with the median value
df.loc[outliers, "RBP(Diastolic)"] = df["RBP(Diastolic)"].median()
```

In [12]:

```
#checking outliers in RBP(Diastolic) Column
sns.boxplot(data=df,x='RBP(Diastolic)')
```

Out[12]:

```
<AxesSubplot:xlabel='RBP(Diastolic)'>
```



In [13]:

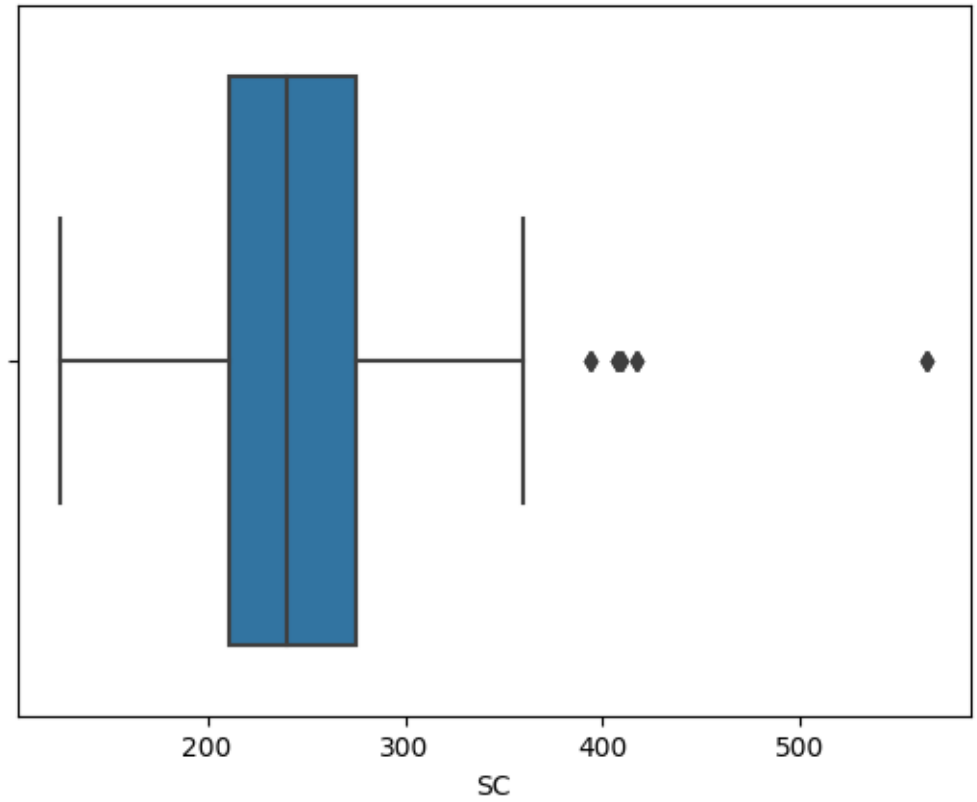
```
#there are no outliers left in the RBP(Diastolic) column
```

In [14]:

```
#checking outliers in SC Column
sns.boxplot(data=df,x='SC')
```

Out[14]:

```
<AxesSubplot:xlabel='SC'>
```



In [15]:

```
#there are some outliers.We will replace them with median value
```

In [16]:

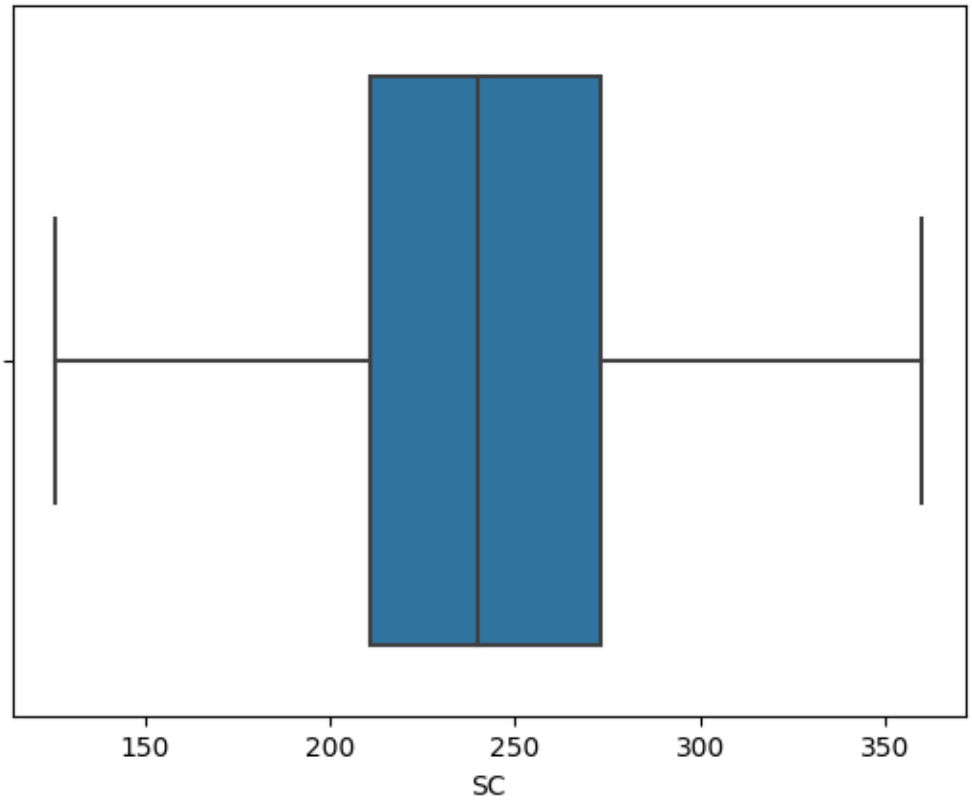
```
# Calculating the Interquartile Range (IQR)
q1 = df["SC"].quantile(0.25)
q3 = df["SC"].quantile(0.75)
iqr = q3 - q1

# Defining the lower and upper bounds for outliers
lower_bound = q1 - 1.5 * iqr
upper_bound = q3 + 1.5 * iqr

# Identifying outliers
outliers = (df["SC"] < lower_bound) | (df["SC"] > upper_bound)

# Replace outliers with the median value
df.loc[outliers, "SC"] = df["SC"].median()
```

In [17]:

```
#checking outliers in SC Column
sns.boxplot(data=df,x='SC')
```

Out[17]:

```
<AxesSubplot:xlabel='SC'>
```



In [18]:

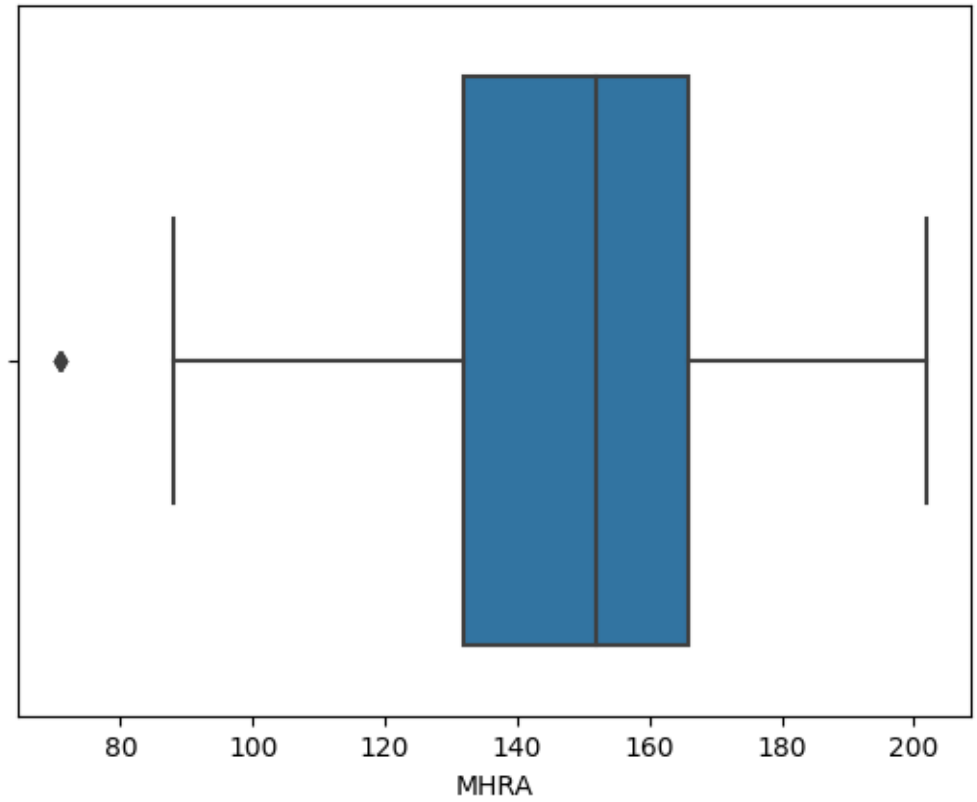
```
#there are no outliers left in the SC column
```

In [19]:

```
#checking outliers in MHRA Column
sns.boxplot(data=df,x='MHRA')
```

Out[19]:

```
<AxesSubplot:xlabel='MHRA'>
```



In [20]:

```
#there are some outliers.We will replace them with median value
```

In [21]:

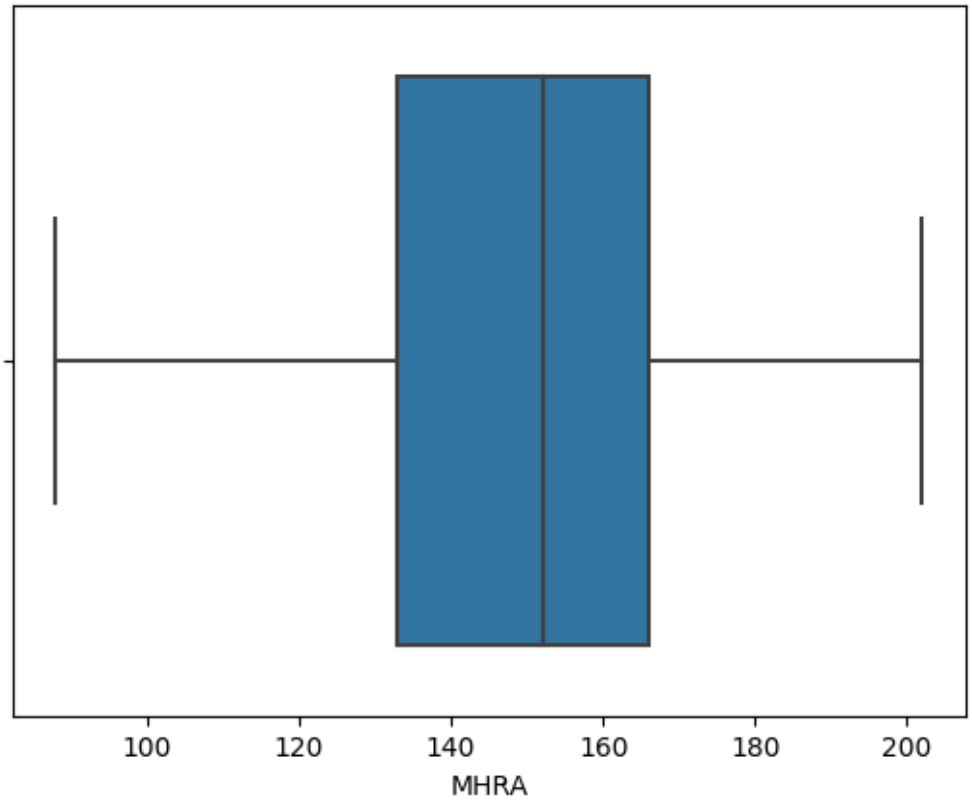
```
# Calculating the Interquartile Range (IQR)
q1 = df["MHRA"].quantile(0.25)
q3 = df["MHRA"].quantile(0.75)
iqr = q3 - q1

# Defining the lower and upper bounds for outliers
lower_bound = q1 - 1.5 * iqr
upper_bound = q3 + 1.5 * iqr

# Identifying outliers
outliers = (df["MHRA"] < lower_bound) | (df["MHRA"] > upper_bound)

# Replace outliers with the median value
df.loc[outliers, "MHRA"] = df["MHRA"].median()
```

In [22]:

```
#checking outliers in MHRA Column
sns.boxplot(data=df,x='MHRA')
```

Out[22]:

```
<AxesSubplot:xlabel='MHRA'>
```



In [23]:

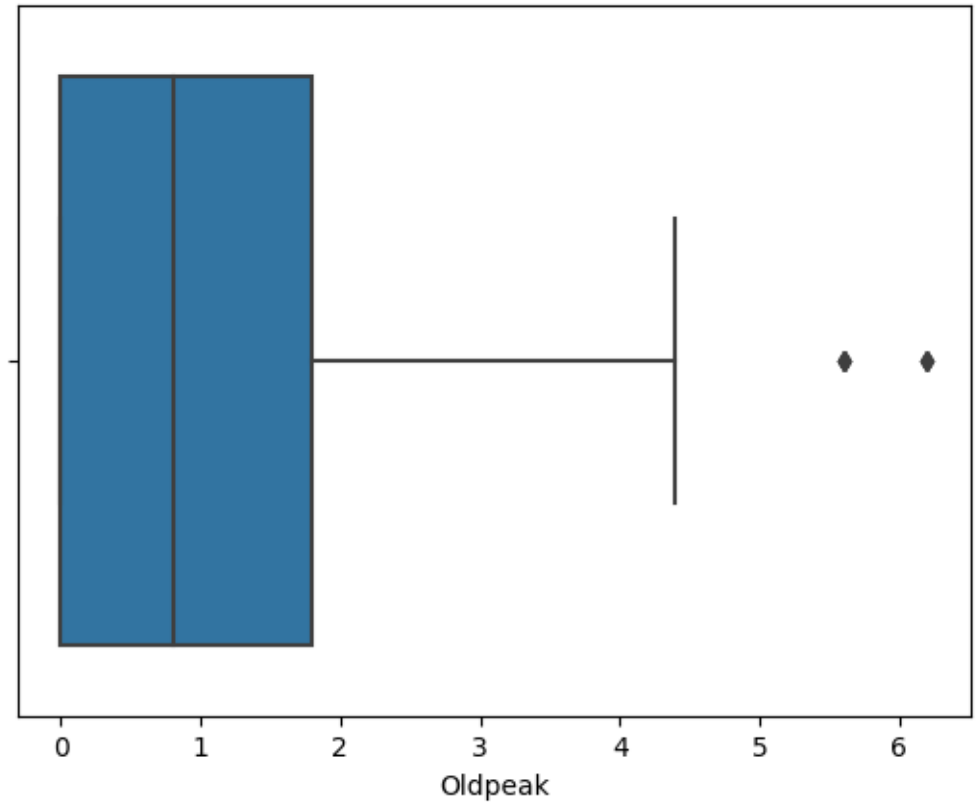
```
#there are no outliers left in the MHRA column
```

In [24]:

```
#checking outliers in Oldpeak Column
sns.boxplot(data=df,x='Oldpeak')
```

Out[24]:

```
<AxesSubplot:xlabel='Oldpeak'>
```



In [25]:

```
#there are some outliers.We will replace them with median value
```

In [26]:

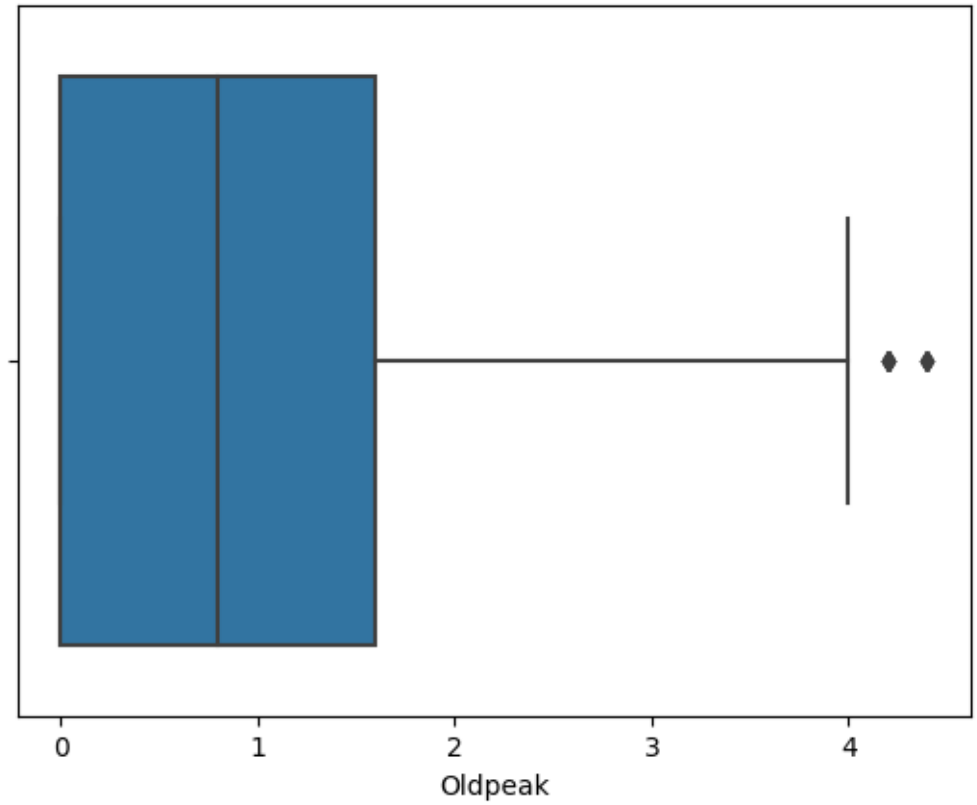
```
# Calculating the Interquartile Range (IQR)
q1 = df["Oldpeak"].quantile(0.25)
q3 = df["Oldpeak"].quantile(0.75)
iqr = q3 - q1

# Defining the lower and upper bounds for outliers
lower_bound = q1 - 1.5 * iqr
upper_bound = q3 + 1.5 * iqr

# Identifying outliers
outliers = (df["Oldpeak"] < lower_bound) | (df["Oldpeak"] > upper_bound)

# Replace outliers with the median value
df.loc[outliers, "Oldpeak"] = df["Oldpeak"].median()
```

In [27]:

```
#checking outliers in Oldpeak Column
sns.boxplot(data=df,x='Oldpeak')
```

Out[27]:

```
<AxesSubplot:xlabel='Oldpeak'>
```



In [28]:

```
#as the outliers are still there, we will repeat the same process one more
time
```

In [29]:

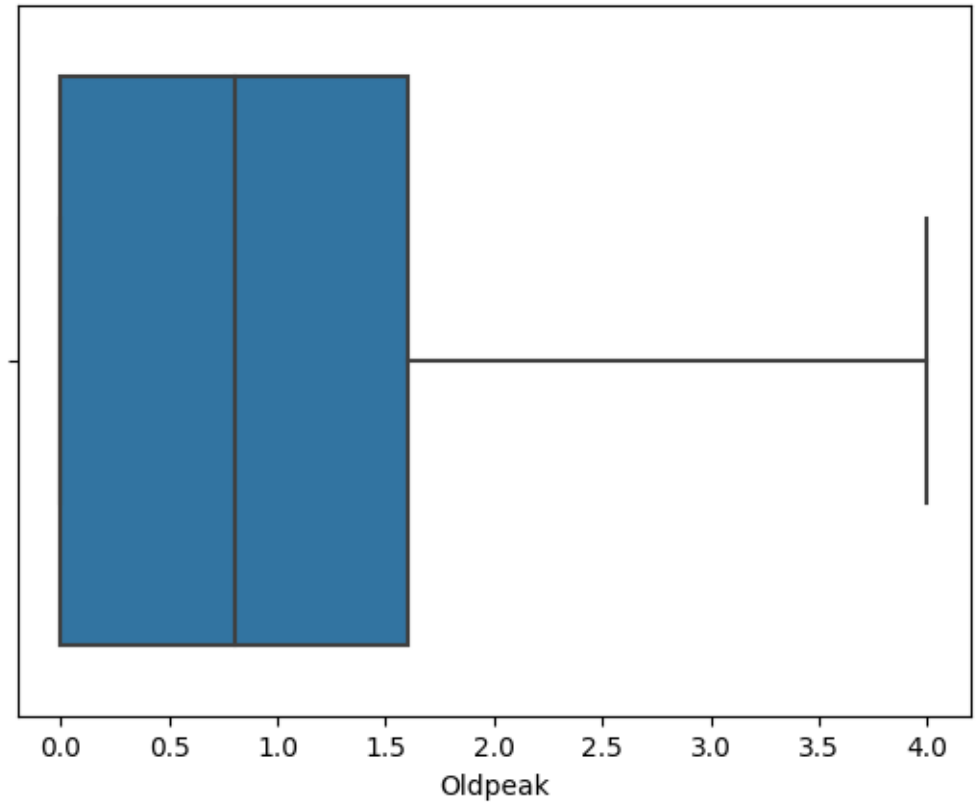
```
# Calculating the Interquartile Range (IQR)
q1 = df["Oldpeak"].quantile(0.25)
q3 = df["Oldpeak"].quantile(0.75)
iqr = q3 - q1

# Defining the lower and upper bounds for outliers
lower_bound = q1 - 1.5 * iqr
upper_bound = q3 + 1.5 * iqr

# Identifying outliers
outliers = (df["Oldpeak"] < lower_bound) | (df["Oldpeak"] > upper_bound)

# Replace outliers with the median value
df.loc[outliers, "Oldpeak"] = df["Oldpeak"].median()
```

In [30]:

```
#checking outliers in Oldpeak Column
sns.boxplot(data=df,x='Oldpeak')
```

Out[30]:

```
<AxesSubplot:xlabel='Oldpeak'>
```



In [31]:

```
#there are no outliers left in the Oldpeak column
```

In [34]:

```
#But some values of Oldpeak column has value 0, which is practically not
correct.
#So, we replace those values with the median of Oldpeak
# Replace 0 values in "Oldpeak" with the median
df.loc[df["Oldpeak"] == 0, "Oldpeak"] = df["Oldpeak"].median()
```

In [35]:

```
df.sample(20)
```

Out[35]:

	Sl_No	Age	Gender	CP	RBP(Diastolic)	SC	FBS	RER	MHRA	EIE	Old peak	Slope	Flourosopy	Thalliu m Test	HD
77	78	63	Male	0	140	187	0	0	144	1	4.0	2	2	2	0
75	76	47	Male	2	138	257	0	0	156	0	0.8	2	0	2	1

	Sl_No	Age	Gender	CP	RBP(Diastolic)	SC	FBS	RER	MHRA	EIE	Old peak	Slope	Flourosopy	Thalliu m Test	HD
939	940	49	Female	1	134	271	0	1	162	0	0.8	1	0	2	1
653	654	56	Male	0	130	283	1	0	103	1	1.6	0	0	2	0
530	531	60	Female	0	150	258	0	0	157	0	2.6	1	2	2	0
160	161	77	Male	0	125	304	0	0	162	1	0.8	2	3	2	0
825	826	63	Female	2	135	252	0	0	172	0	0.8	2	0	2	1
1008	1009	42	Male	1	120	295	0	1	162	0	0.8	2	0	2	1
303	304	60	Male	0	145	282	0	0	142	1	2.8	1	2	2	0
175	176	56	Female	0	130	288	1	0	133	1	4.0	0	2	2	0
130	131	60	Female	3	150	240	0	1	171	0	0.9	2	0	2	1

	Sl_No	Age	Gender	CP	RBP(Diastolic)	SC	FBS	RER	MHRA	EIE	Old peak	Slope	Flourosopy	Thalliu m Test	HD
38	39	64	Male	0	128	263	0	1	105	1	0.2	1	1	2	1
655	656	41	Male	1	110	235	0	1	153	0	0.8	2	0	2	1
342	343	65	Female	2	155	269	0	1	148	0	0.8	2	0	2	1
162	163	77	Male	0	125	304	0	0	162	1	0.8	2	3	2	0
404	405	61	Male	0	140	207	0	0	138	1	1.9	2	1	2	0
661	662	58	Male	0	114	318	0	2	140	0	0.8	0	3	1	0
1013	1014	58	Male	0	114	318	0	2	140	0	0.8	0	3	1	0
152	153	58	Male	0	125	300	0	0	171	0	0.8	2	2	2	0
135	136	58	Female	0	170	225	1	0	146	1	2.8	1	2	1	0

In [36]:

```
#now our dataset is clean and without any outliers. We can do the analysis
now
#lets export the dataframe as csv file
```

In [38]:

```
df.to_csv('/content/Heart Disease data.csv, index=False)
```

In [ ]: