# Outliers Homework

- Shawn Goodin
- July 2022

In [35]:
```
import numpy as np
import pandas as pd
import matplotlib as mpl
import seaborn as sns
import warnings as w

import matplotlib.pyplot as plt
from matplotlib.cbook import boxplot_stats

w.filterwarnings('ignore')

df = pd.read_csv ('PEP1.csv')
pd.set_option('display.max_rows', None)
df.head()
```

Out[35]:

| | Id | MSSubClass | MSZoning | LotFrontage | LotArea | Street | Alley | LotShape | LandContour | Utilities | ... | PoolArea | PoolQC | Fence | Mi |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 1 | 60 | RL | 65.0 | 8450 | Pave | NaN | Reg | Lvl | AllPub | ... | 0 | NaN | NaN | |
| **1** | 2 | 20 | RL | 80.0 | 9600 | Pave | NaN | Reg | Lvl | AllPub | ... | 0 | NaN | NaN | |
| **2** | 3 | 60 | RL | 68.0 | 11250 | Pave | NaN | IR1 | Lvl | AllPub | ... | 0 | NaN | NaN | |
| **3** | 4 | 70 | RL | 60.0 | 9550 | Pave | NaN | IR1 | Lvl | AllPub | ... | 0 | NaN | NaN | |
| **4** | 5 | 60 | RL | 84.0 | 14260 | Pave | NaN | IR1 | Lvl | AllPub | ... | 0 | NaN | NaN | |

5 rows × 81 columns

In [36]:
```
df.shape
```

Out[36]: (1460, 81)

In [37]:
```
df.size
```

Out[37]: 118260

In [38]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1460 entries, 0 to 1459
Data columns (total 81 columns):
 #   Column          Non-Null Count  Dtype  
---  ------          --------------  -----  
 0   Id              1460 non-null   int64  
 1   MSSubClass      1460 non-null   int64  
 2   MSZoning        1460 non-null   object 
 3   LotFrontage     1201 non-null   float64
 4   LotArea         1460 non-null   int64  
 5   Street          1460 non-null   object 
 6   Alley           91 non-null     object 
 7   LotShape        1460 non-null   object 
 8   LandContour     1460 non-null   object 
 9   Utilities       1460 non-null   object 
 10  LotConfig       1460 non-null   object 
 11  LandSlope       1460 non-null   object 
 12  Neighborhood    1460 non-null   object 
 13  Condition1      1460 non-null   object 
 14  Condition2      1460 non-null   object 
 15  BldgType        1460 non-null   object 
 16  HouseStyle      1460 non-null   object 
 17  OverallQual     1460 non-null   int64  
 18  OverallCond     1460 non-null   int64  
 19  YearBuilt       1460 non-null   int64  
 20  YearRemodAdd    1460 non-null   int64  
 21  RoofStyle       1460 non-null   object 
 22  RoofMatl        1460 non-null   object 
 23  Exterior1st     1460 non-null   object 
 24  Exterior2nd     1460 non-null   object 
 25  MasVnrType      1452 non-null   object 
 26  MasVnrArea      1452 non-null   float64
 27  ExterQual       1460 non-null   object 
 28  ExterCond       1460 non-null   object 
 29  Foundation      1460 non-null   object 
 30  BsmtQual        1423 non-null   object 
 31  BsmtCond        1423 non-null   object 
 32  BsmtExposure    1422 non-null   object 
 33  BsmtFinType1    1423 non-null   object 
 34  BsmtFinSF1      1460 non-null   int64  
 35  BsmtFinType2    1422 non-null   object 
 36  BsmtFinSF2      1460 non-null   int64  
 37  BsmtUnfSF       1460 non-null   int64  
```

```
38  TotalBsmtSF    1460 non-null   int64
39  Heating        1460 non-null   object
40  HeatingQC      1460 non-null   object
41  CentralAir     1460 non-null   object
42  Electrical     1459 non-null   object
43  1stFlrSF       1460 non-null   int64
44  2ndFlrSF       1460 non-null   int64
45  LowQualFinSF   1460 non-null   int64
46  GrLivArea      1460 non-null   int64
47  BsmtFullBath   1460 non-null   int64
48  BsmtHalfBath   1460 non-null   int64
49  FullBath       1460 non-null   int64
50  HalfBath       1460 non-null   int64
51  BedroomAbvGr   1460 non-null   int64
52  KitchebvGr     1460 non-null   int64
53  KitchenQual    1460 non-null   object
54  TotRmsAbvGrd   1460 non-null   int64
55  Functiol       1460 non-null   object
56  Fireplaces     1460 non-null   int64
57  FireplaceQu    770 non-null    object
58  GarageType     1379 non-null   object
59  GarageYrBlt    1379 non-null   float64
60  GarageFinish   1379 non-null   object
61  GarageCars     1460 non-null   int64
62  GarageArea     1460 non-null   int64
63  GarageQual     1379 non-null   object
64  GarageCond     1379 non-null   object
65  PavedDrive     1460 non-null   object
66  WoodDeckSF     1460 non-null   int64
67  OpenPorchSF    1460 non-null   int64
68  EnclosedPorch  1460 non-null   int64
69  3SsnPorch      1460 non-null   int64
70  ScreenPorch    1460 non-null   int64
71  PoolArea       1460 non-null   int64
72  PoolQC         7 non-null      object
73  Fence          281 non-null    object
74  MiscFeature    54 non-null     object
75  MiscVal        1460 non-null   int64
76  MoSold         1460 non-null   int64
77  YrSold         1460 non-null   int64
78  SaleType       1460 non-null   object
79  SaleCondition  1460 non-null   object
80  SalePrice      1460 non-null   int64
```

```
dtypes: float64(3), int64(35), object(43)
memory usage: 924.0+ KB
```

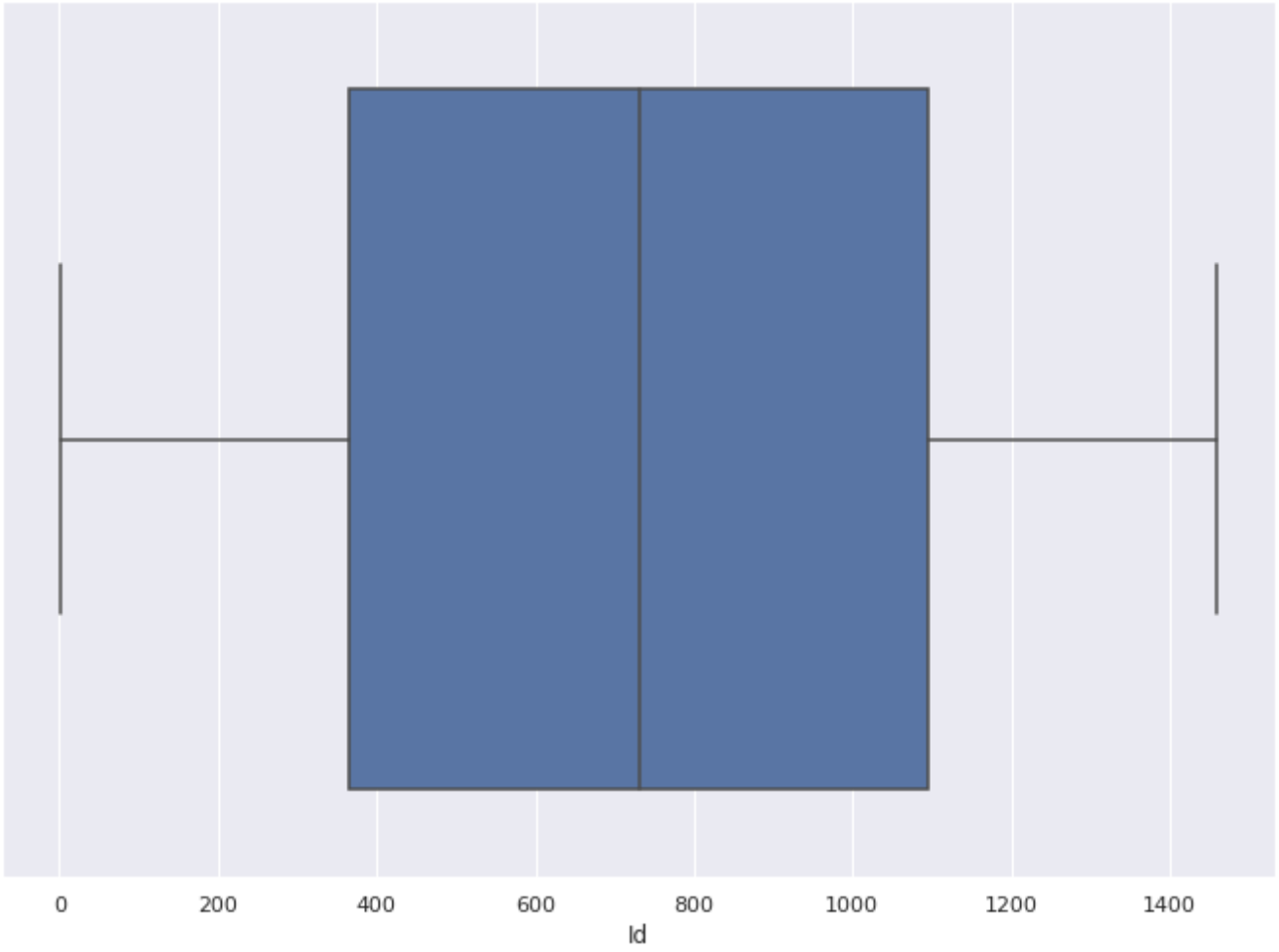
In [39]:
```
df.describe()
```

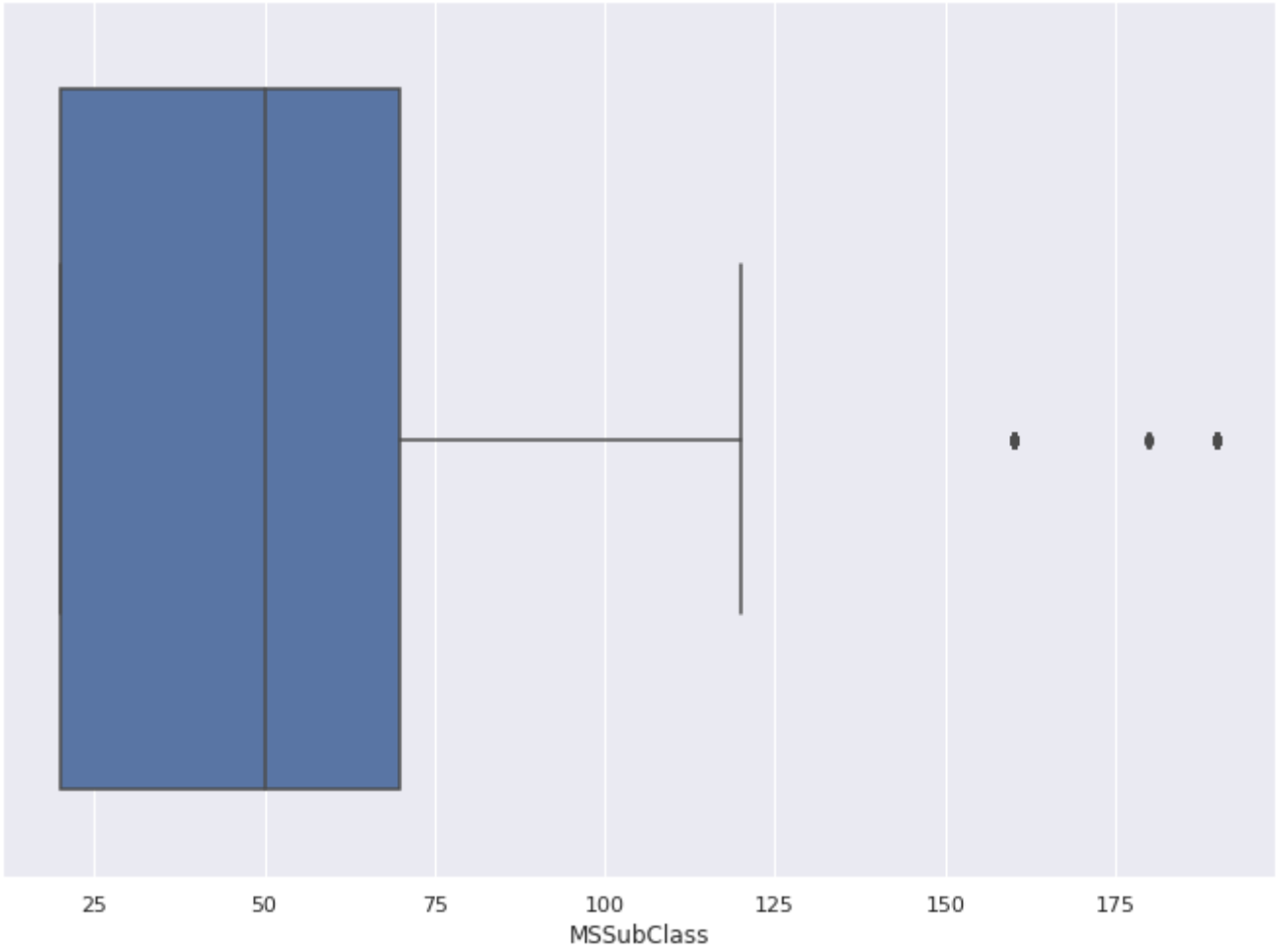
Out[39]:

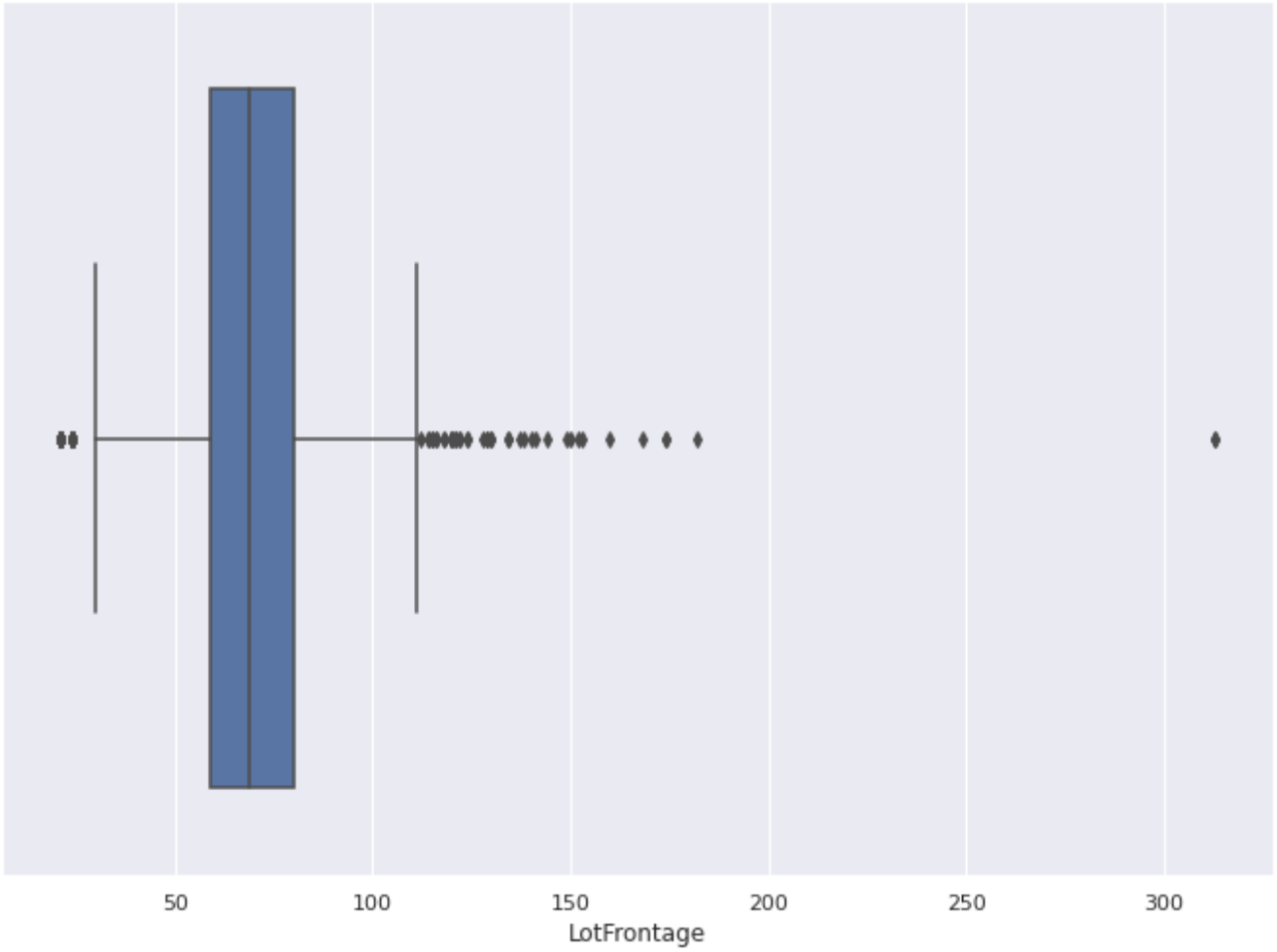
| | Id | MSSubClass | LotFrontage | LotArea | OverallQual | OverallCond | YearBuilt | YearRemodAdd | MasVnrArea | Bs |
|---|---|---|---|---|---|---|---|---|---|---|
| **count** | 1460.000000 | 1460.000000 | 1201.000000 | 1460.000000 | 1460.000000 | 1460.000000 | 1460.000000 | 1460.000000 | 1452.000000 | 14 |
| **mean** | 730.500000 | 56.897260 | 70.049958 | 10516.828082 | 6.099315 | 5.575342 | 1971.267808 | 1984.865753 | 103.685262 | 4 |
| **std** | 421.610009 | 42.300571 | 24.284752 | 9981.264932 | 1.382997 | 1.112799 | 30.202904 | 20.645407 | 181.066207 | 4 |
| **min** | 1.000000 | 20.000000 | 21.000000 | 1300.000000 | 1.000000 | 1.000000 | 1872.000000 | 1950.000000 | 0.000000 | |
| **25%** | 365.750000 | 20.000000 | 59.000000 | 7553.500000 | 5.000000 | 5.000000 | 1954.000000 | 1967.000000 | 0.000000 | |
| **50%** | 730.500000 | 50.000000 | 69.000000 | 9478.500000 | 6.000000 | 5.000000 | 1973.000000 | 1994.000000 | 0.000000 | 3 |
| **75%** | 1095.250000 | 70.000000 | 80.000000 | 11601.500000 | 7.000000 | 6.000000 | 2000.000000 | 2004.000000 | 166.000000 | 7 |
| **max** | 1460.000000 | 190.000000 | 313.000000 | 215245.000000 | 10.000000 | 9.000000 | 2010.000000 | 2010.000000 | 1600.000000 | 56 |

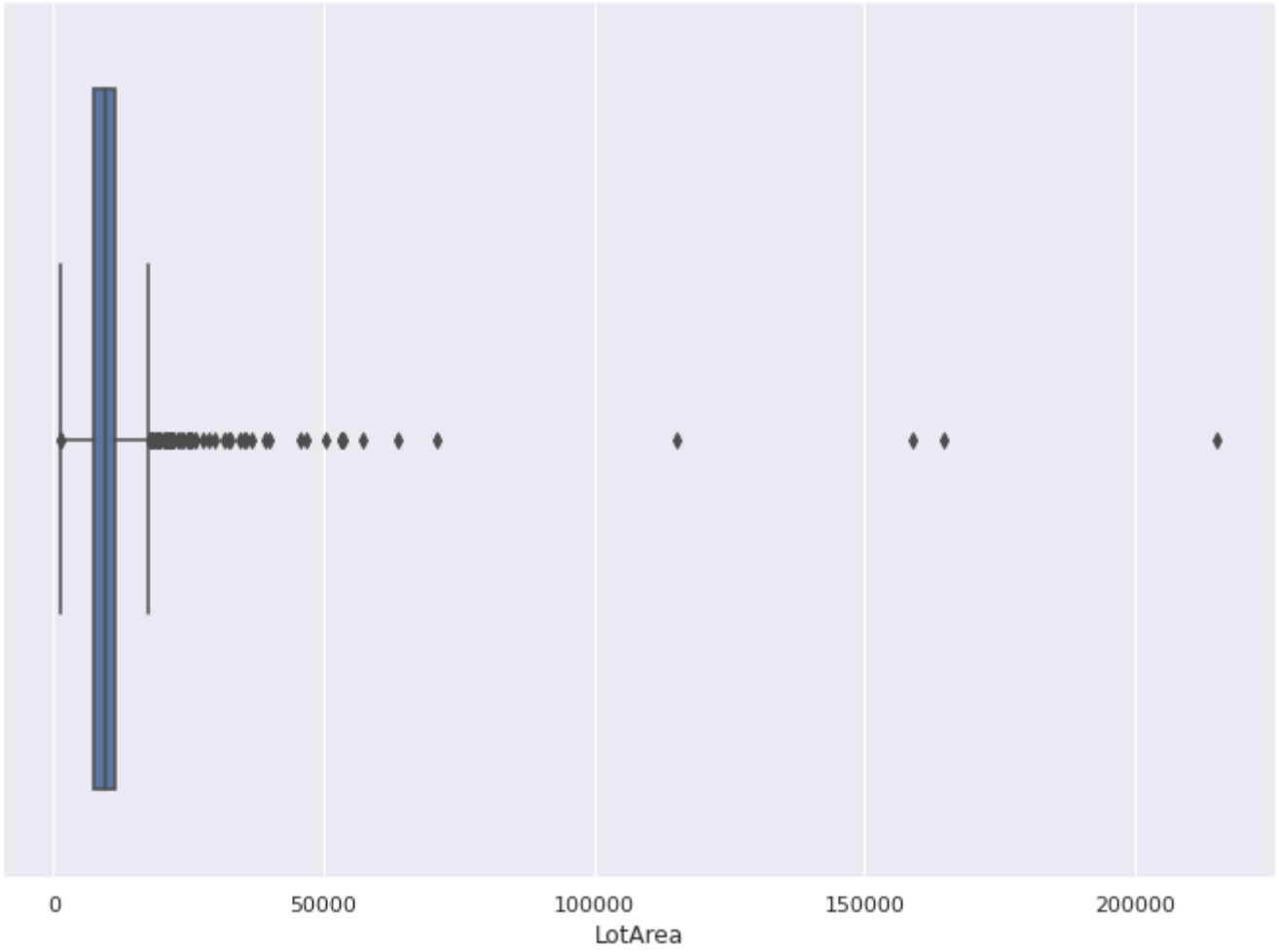
8 rows × 38 columns

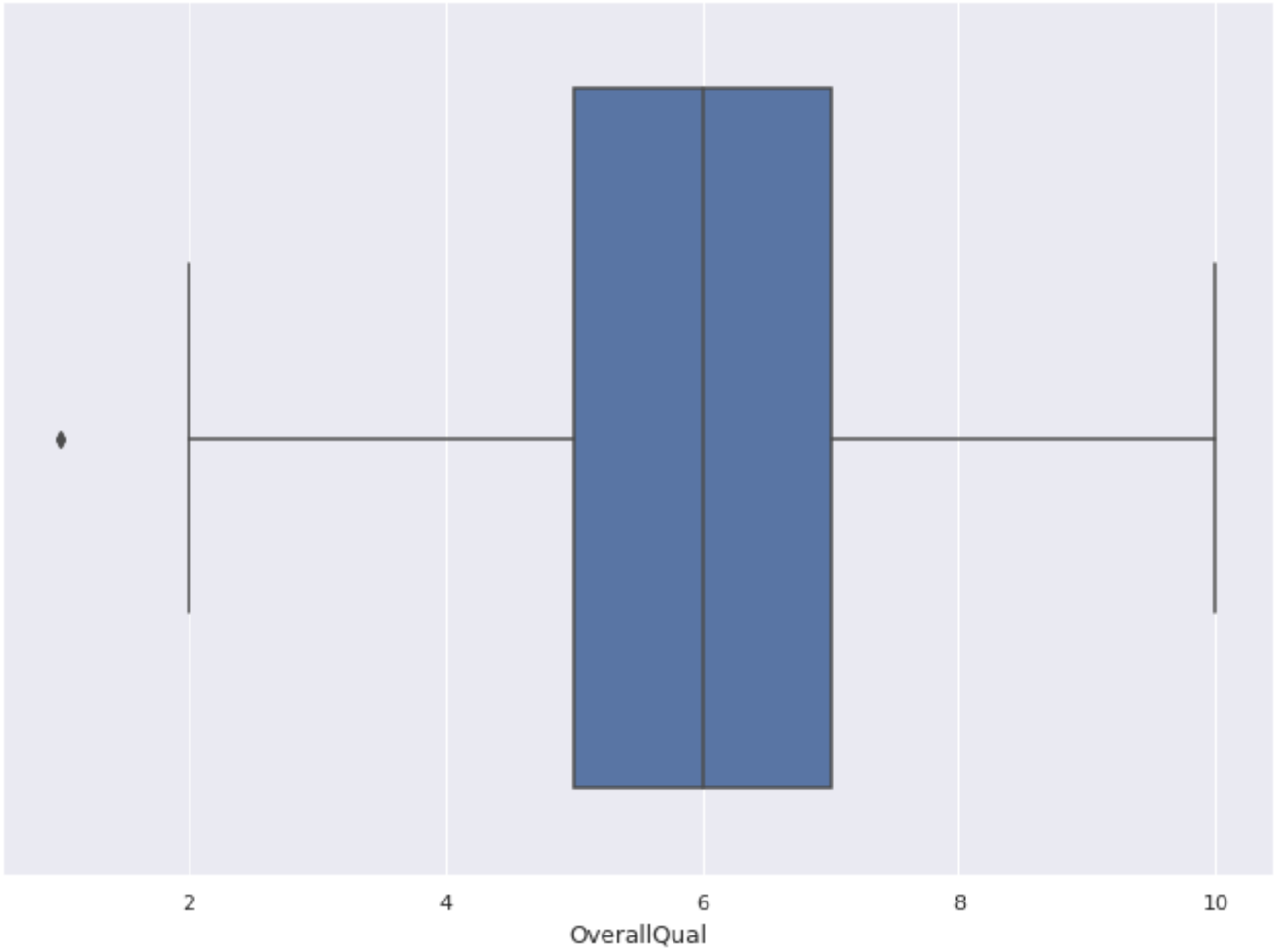
In [40]:
```
def boxplotloop(df, columns):
    for col in columns:
        if df[col].dtype != object:
            sns.set(rc={'figure.figsize':(11.7,8.27)})
            sns.boxplot(df[col])
            plt.show()



boxplotloop(df, df.describe().columns)
```

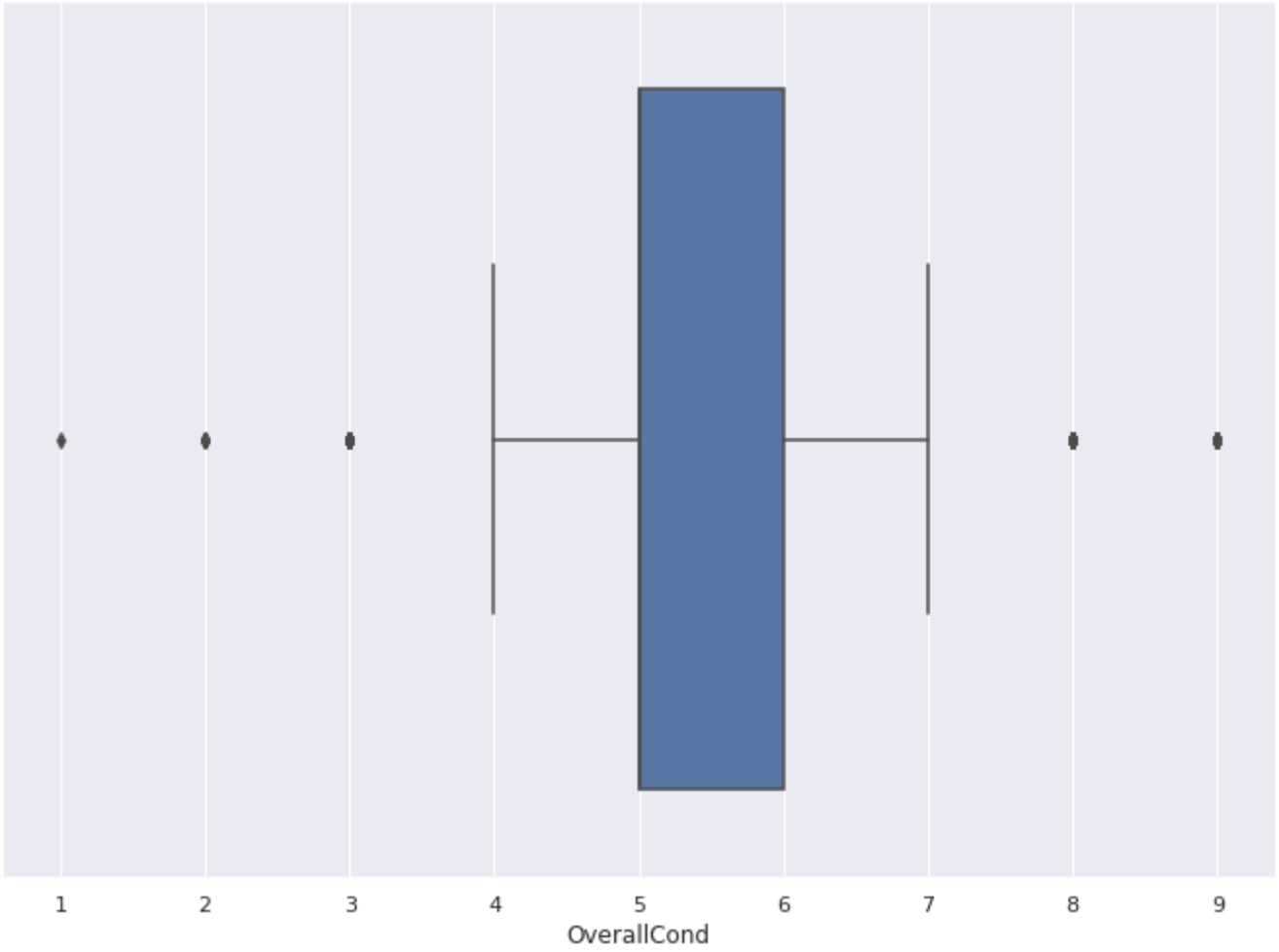

0
200
400
600
800
1000
1200
1400
Id

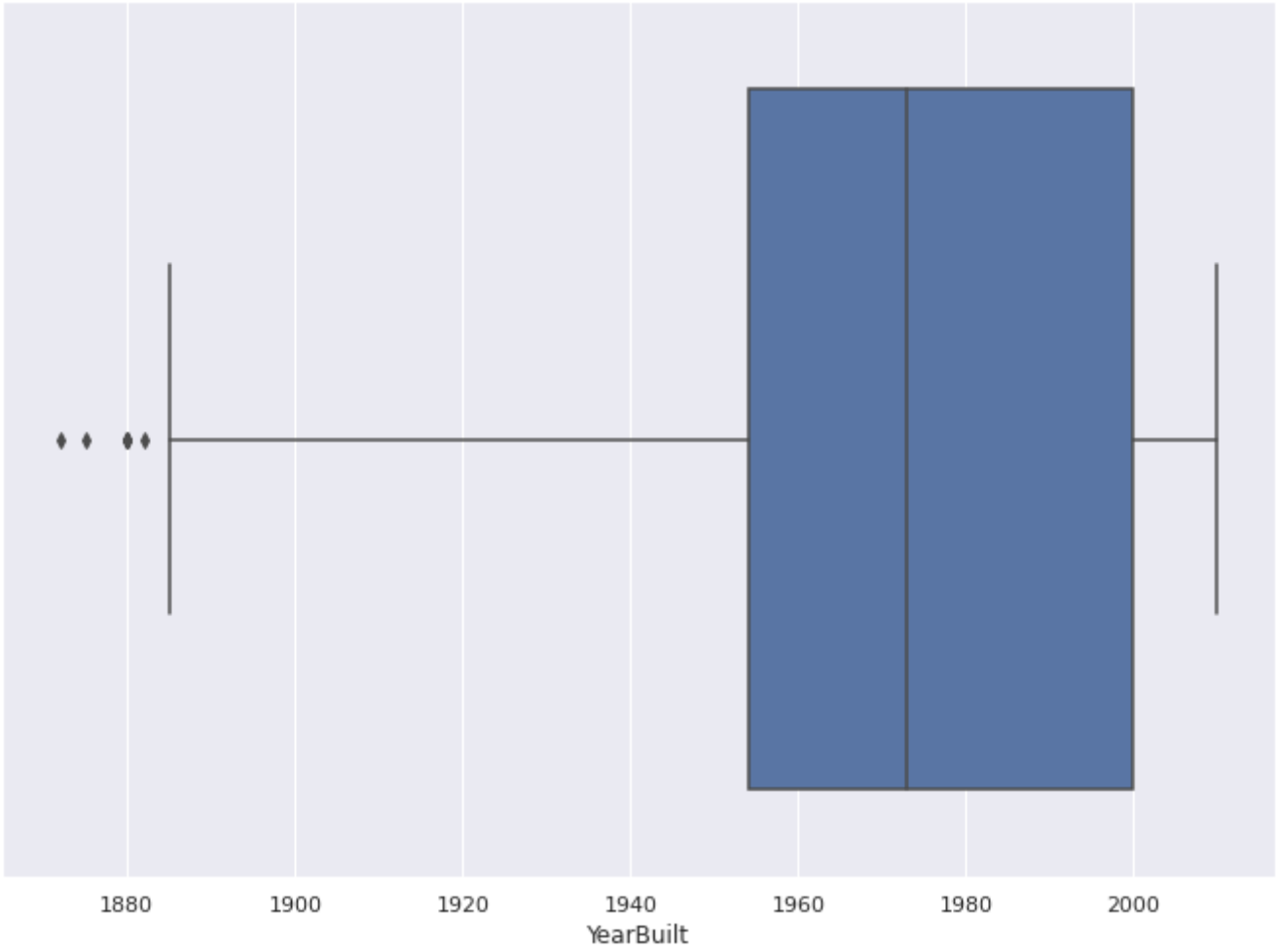

25
50
75
100
125
150
175
MSSubClass

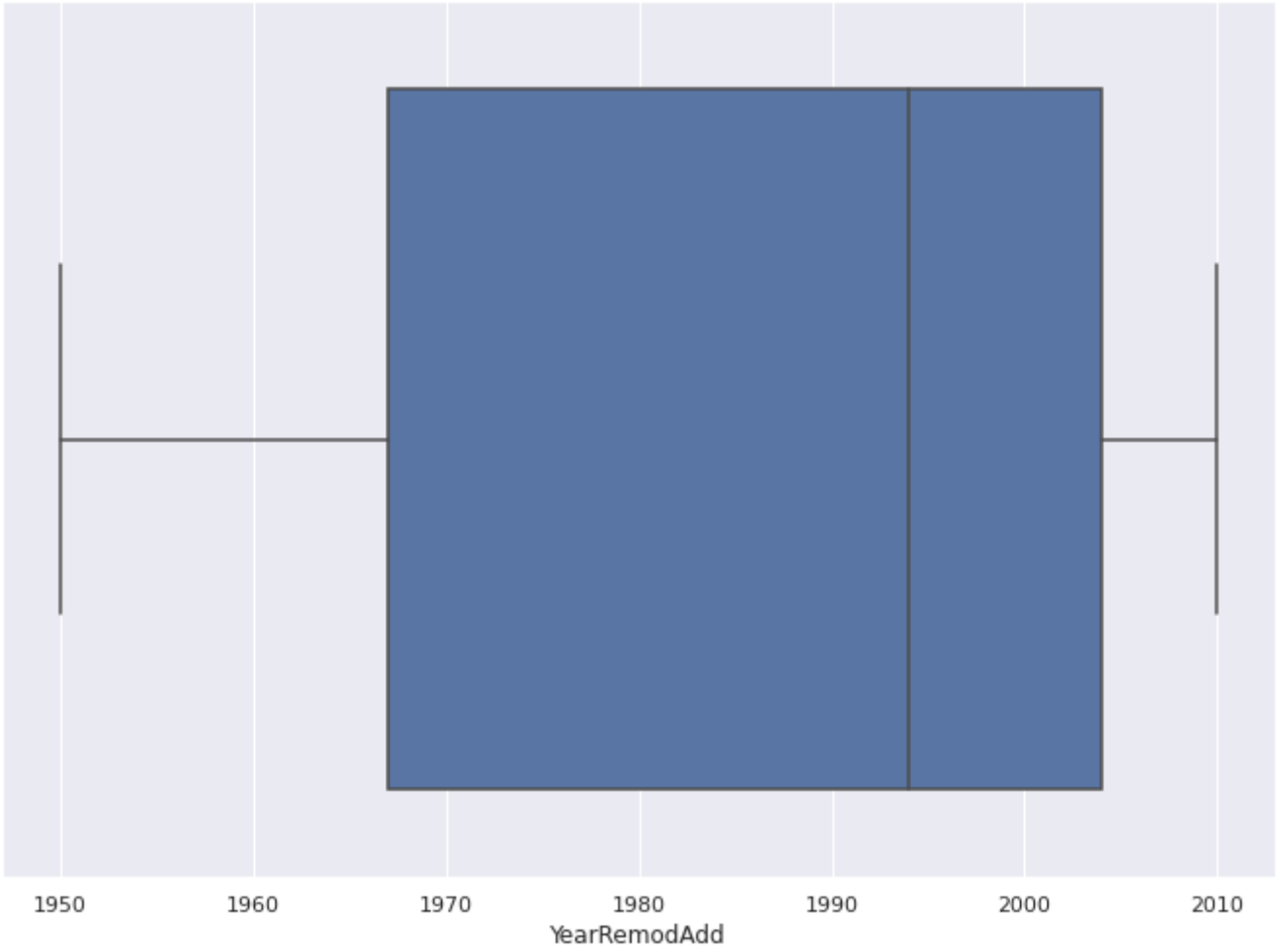

50
100
150
200
250
300
LotFrontage

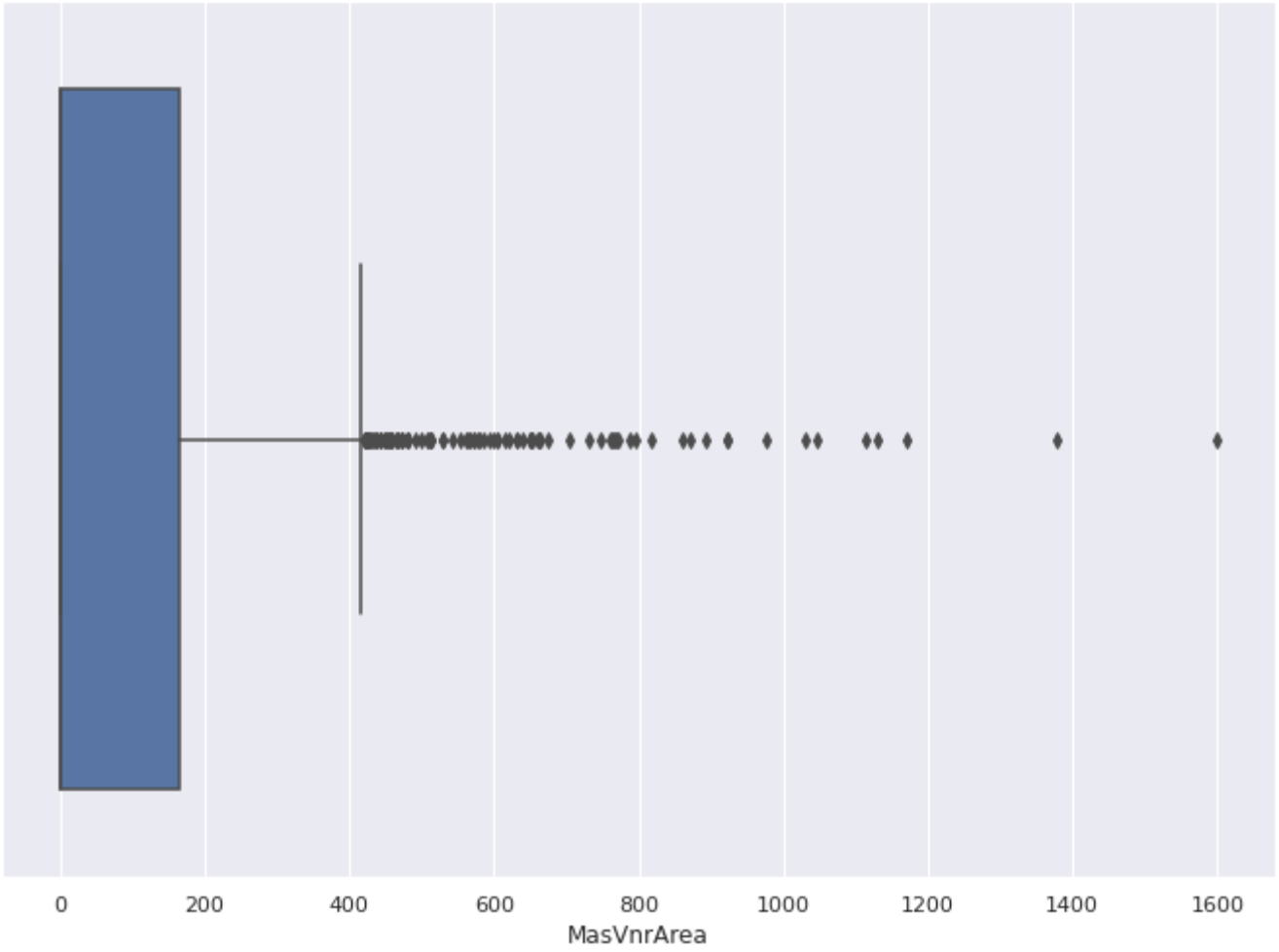

0
50000
100000
150000
200000
LotArea

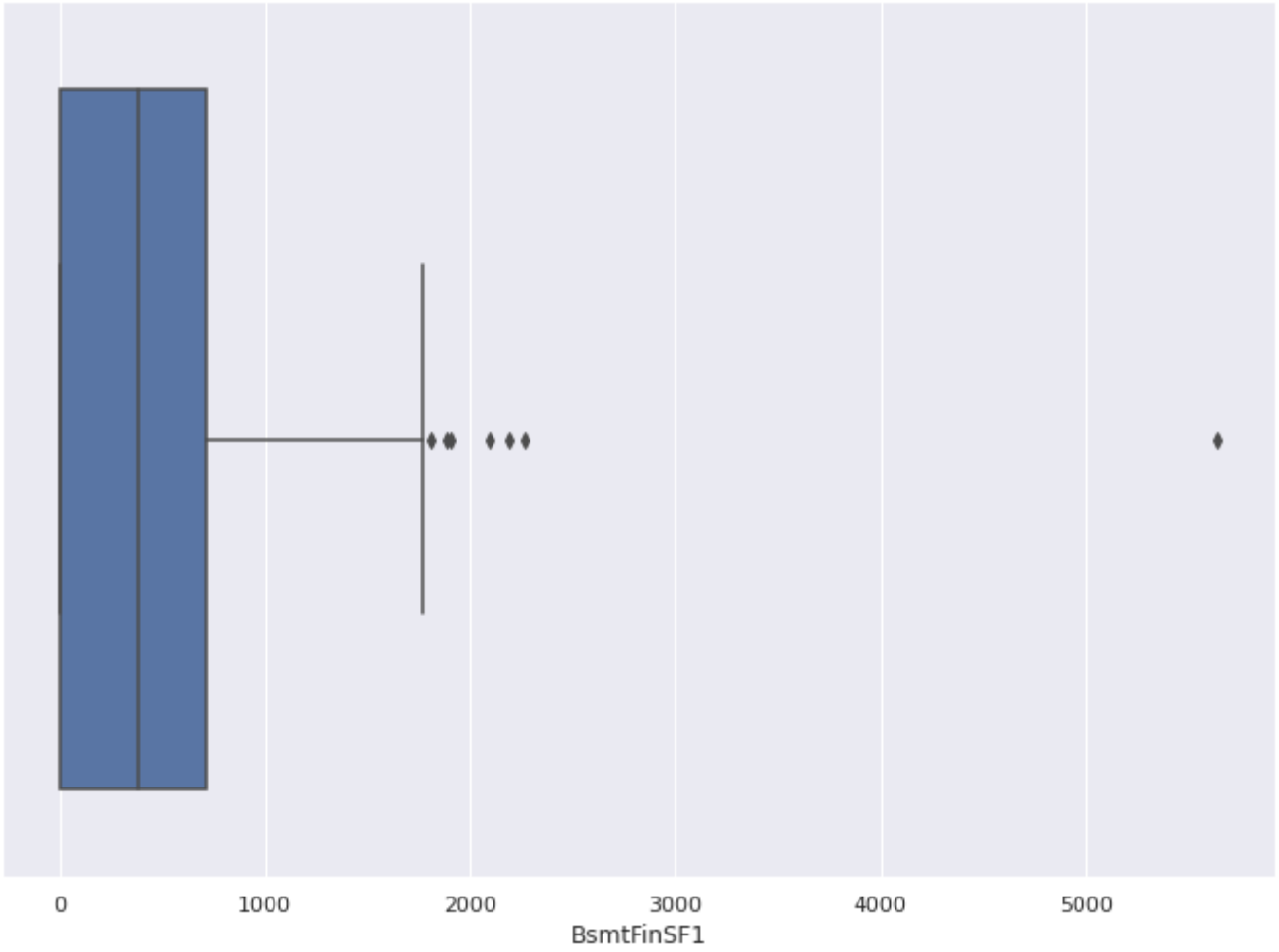

2
4
6
8
10
OverallQual

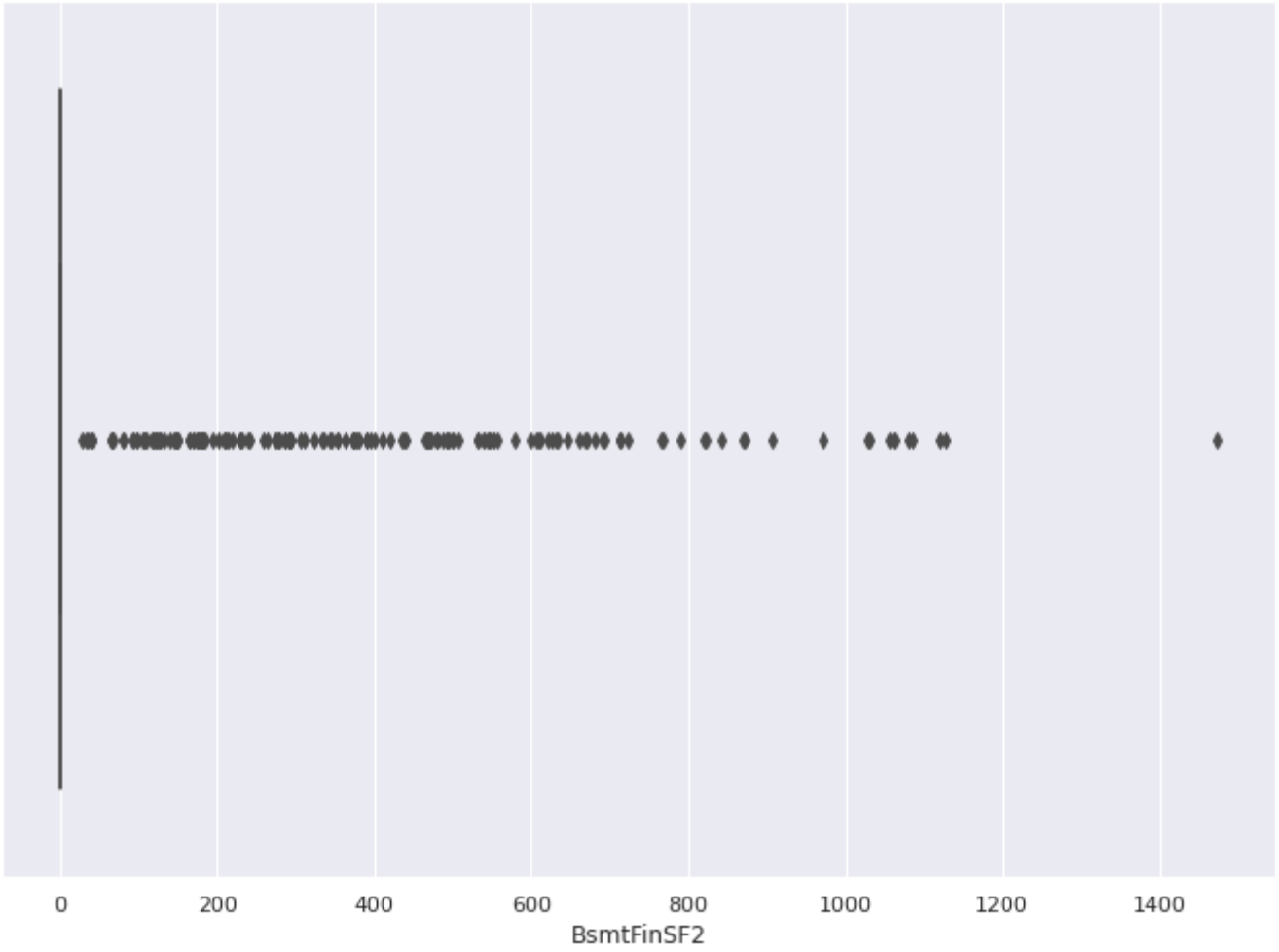

1
2
3
4
5
6
7
8
9
OverallCond

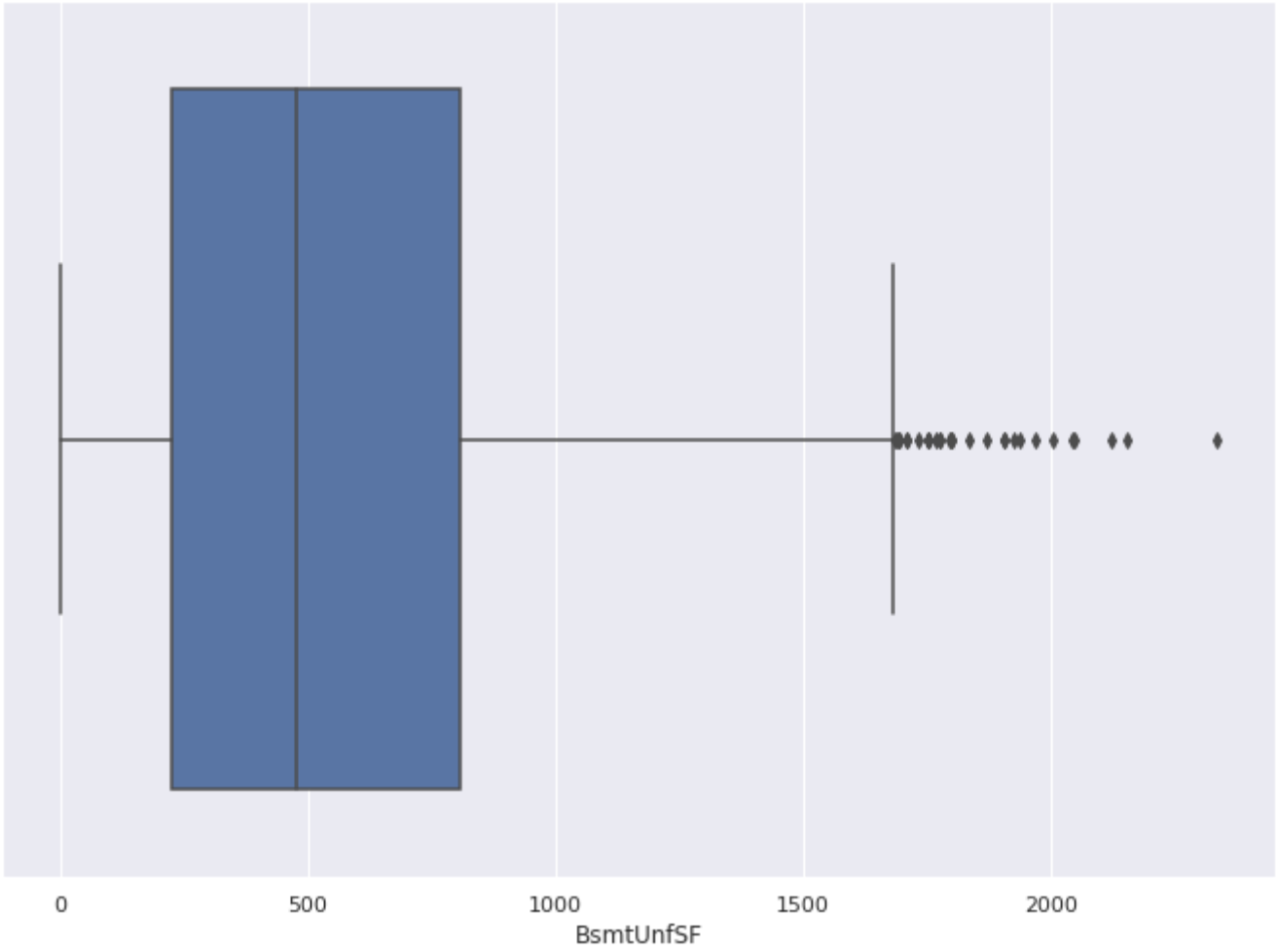

1880
1900
1920
1940
1960
1980
2000
YearBuilt

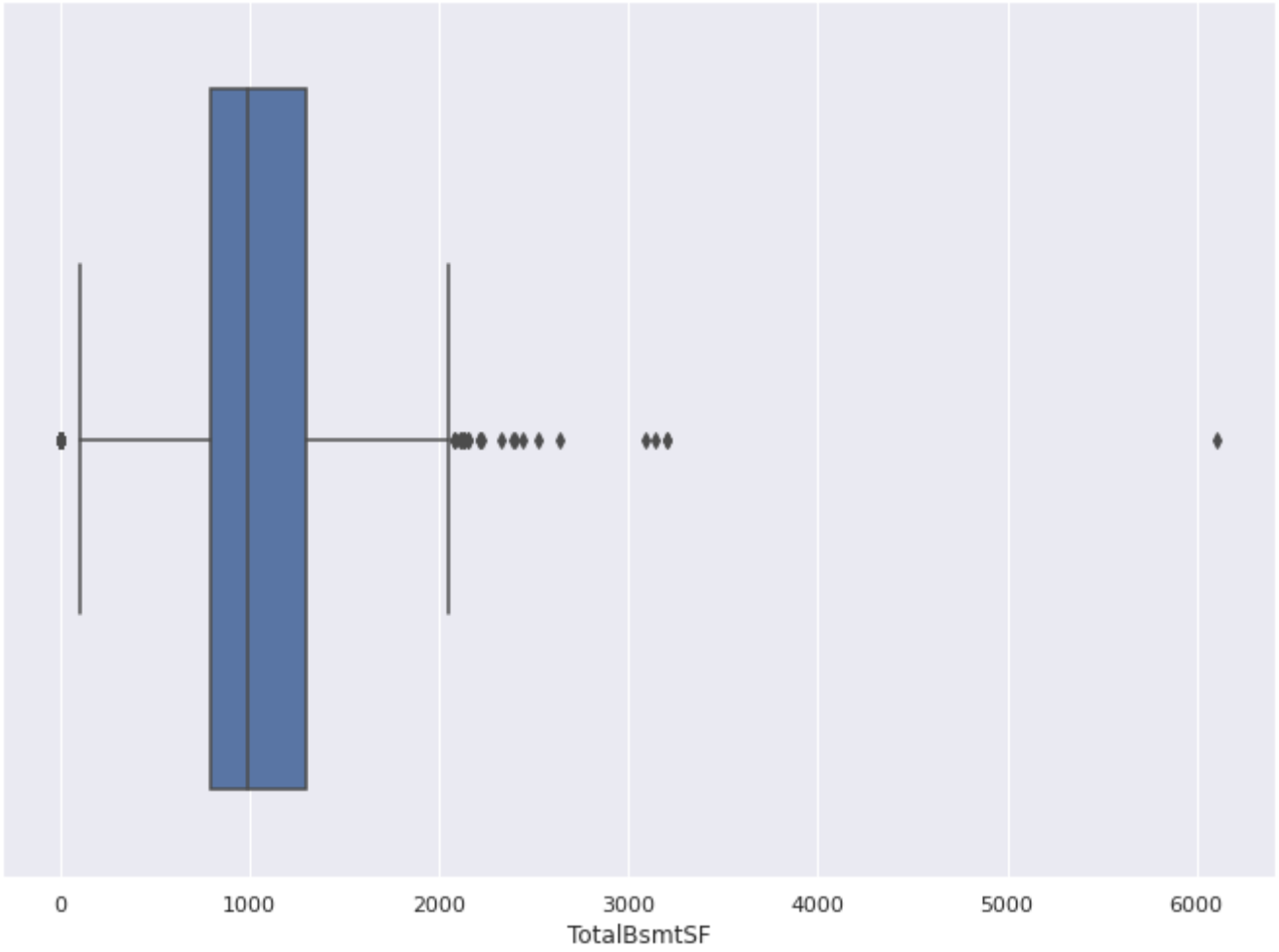

1950
1960
1970
1980
1990
2000
2010
YearRemodAdd

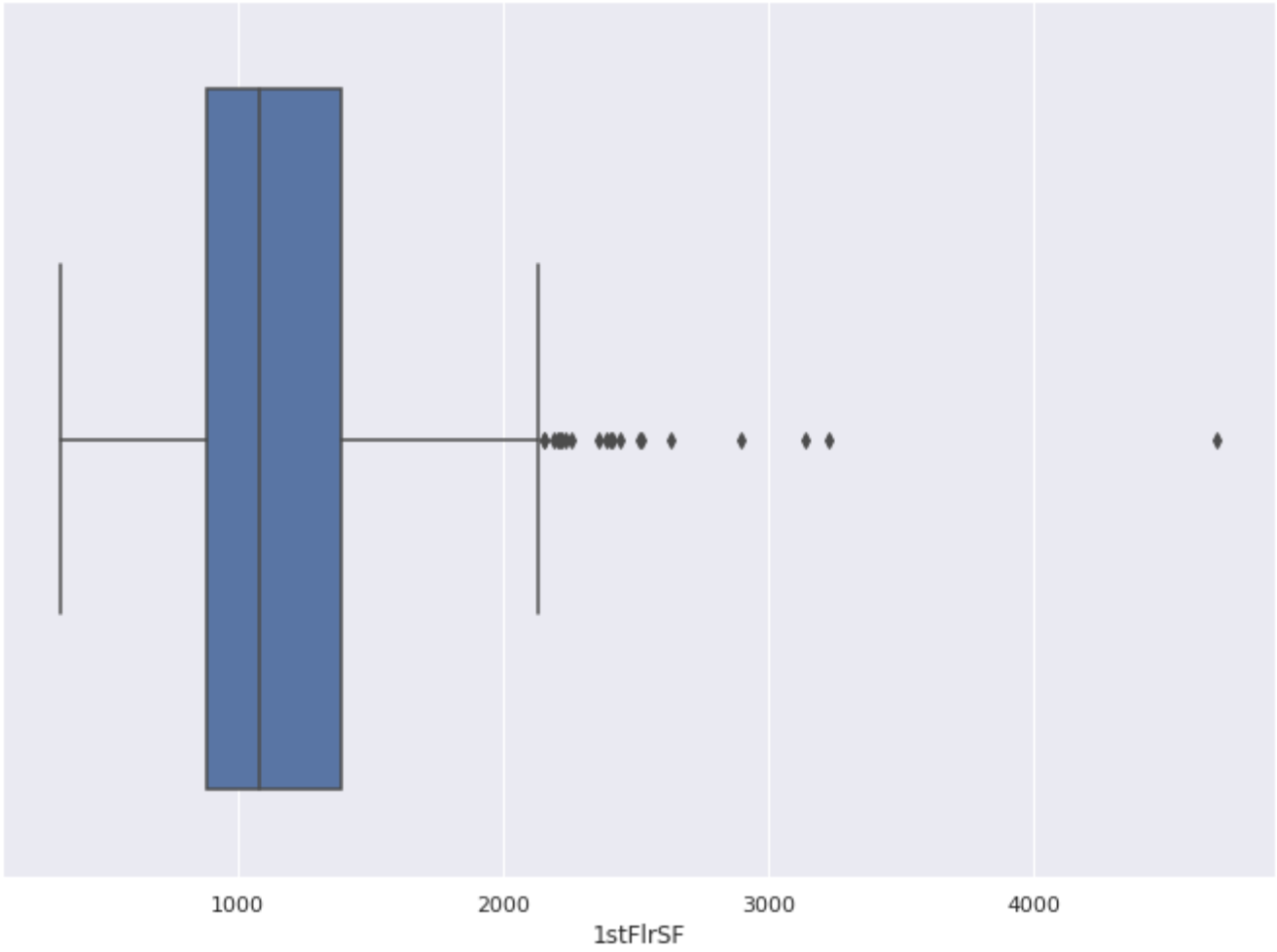

0
200
400
600
800
1000
1200
1400
1600
MasVnrArea

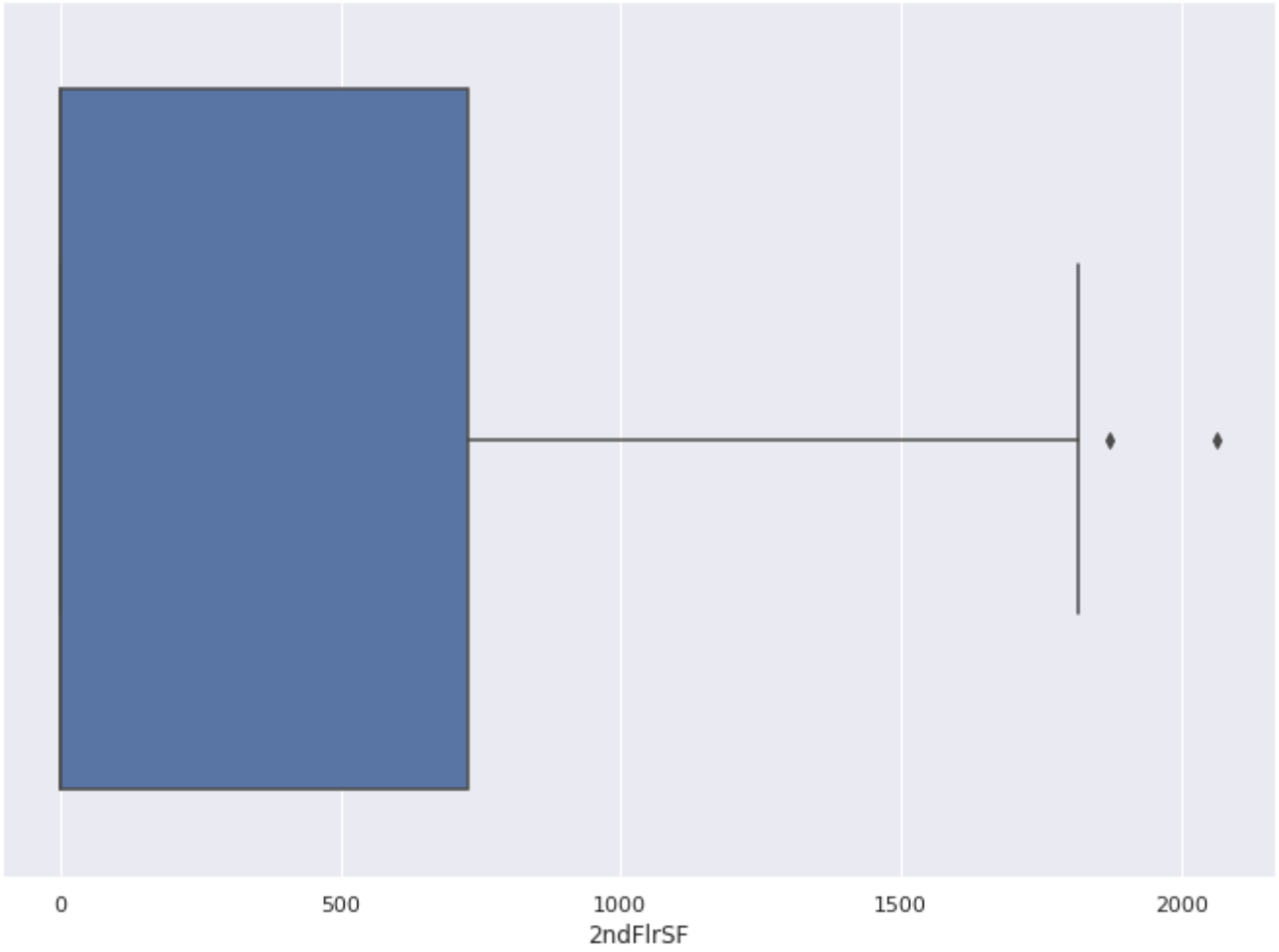

0
1000
2000
3000
4000
5000
BsmtFinSF1

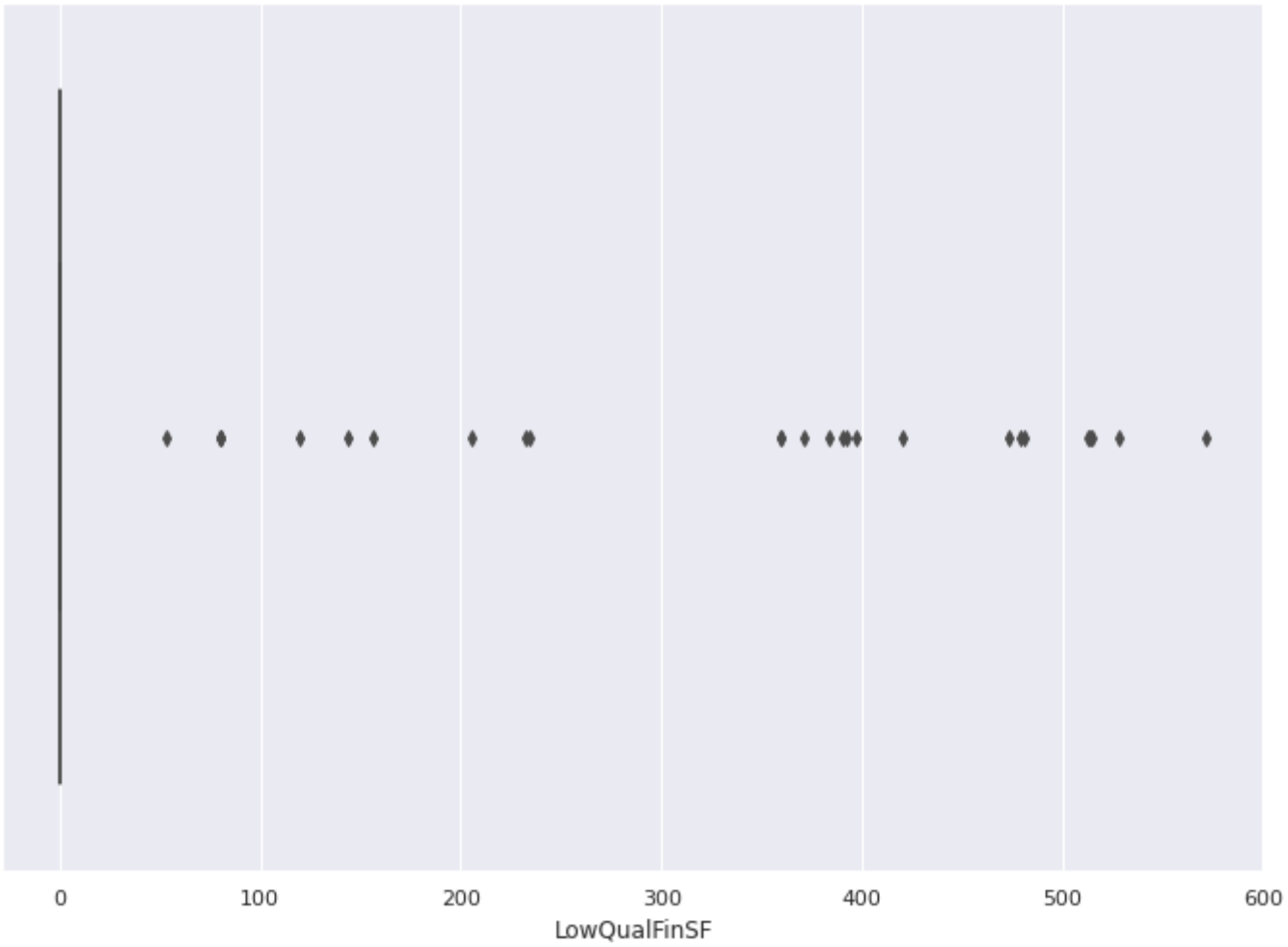

0
200
400
600
800
1000
1200
1400
BsmtFinSF2

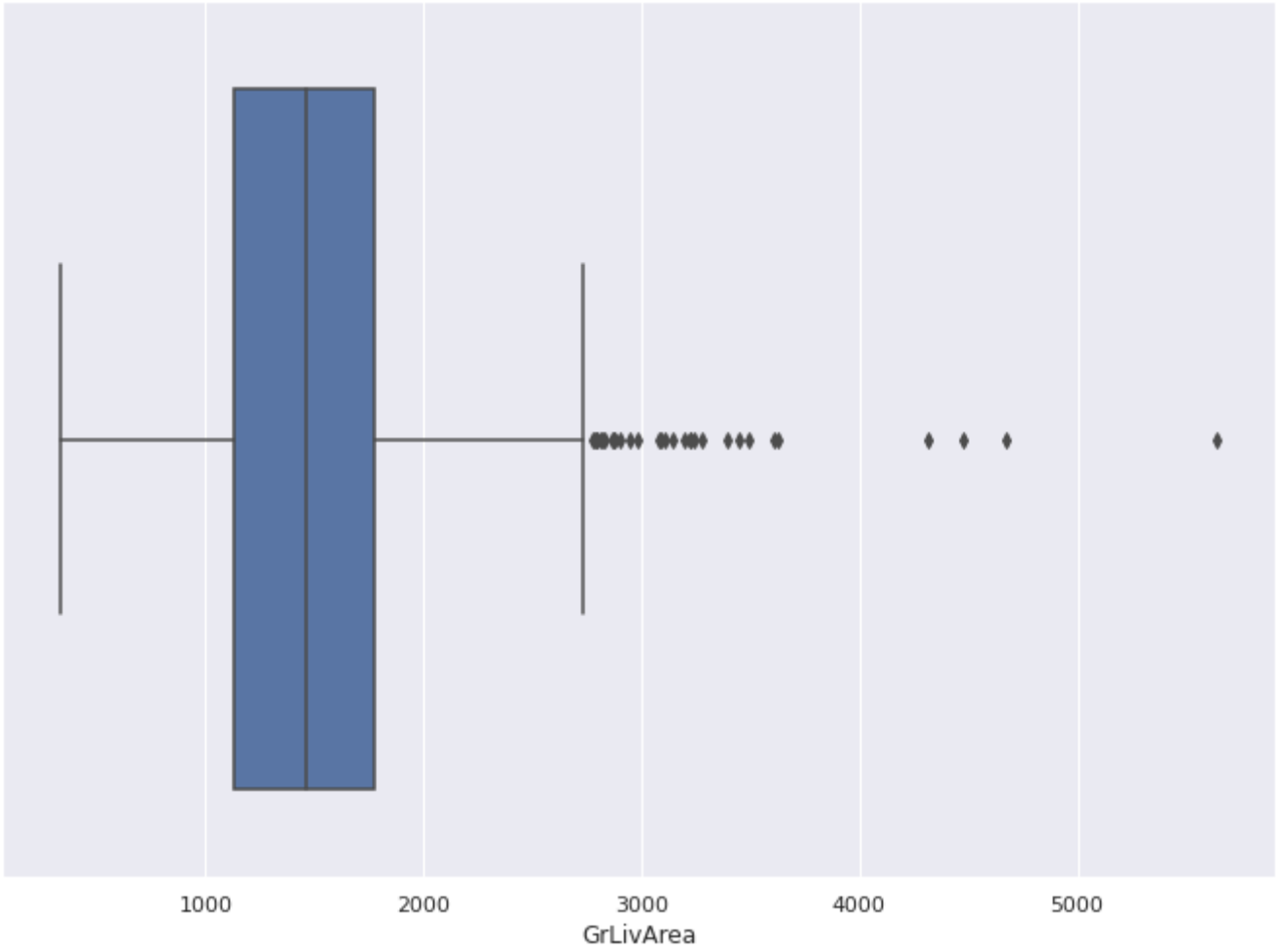

0
500
1000
1500
2000
BsmtUnfSF

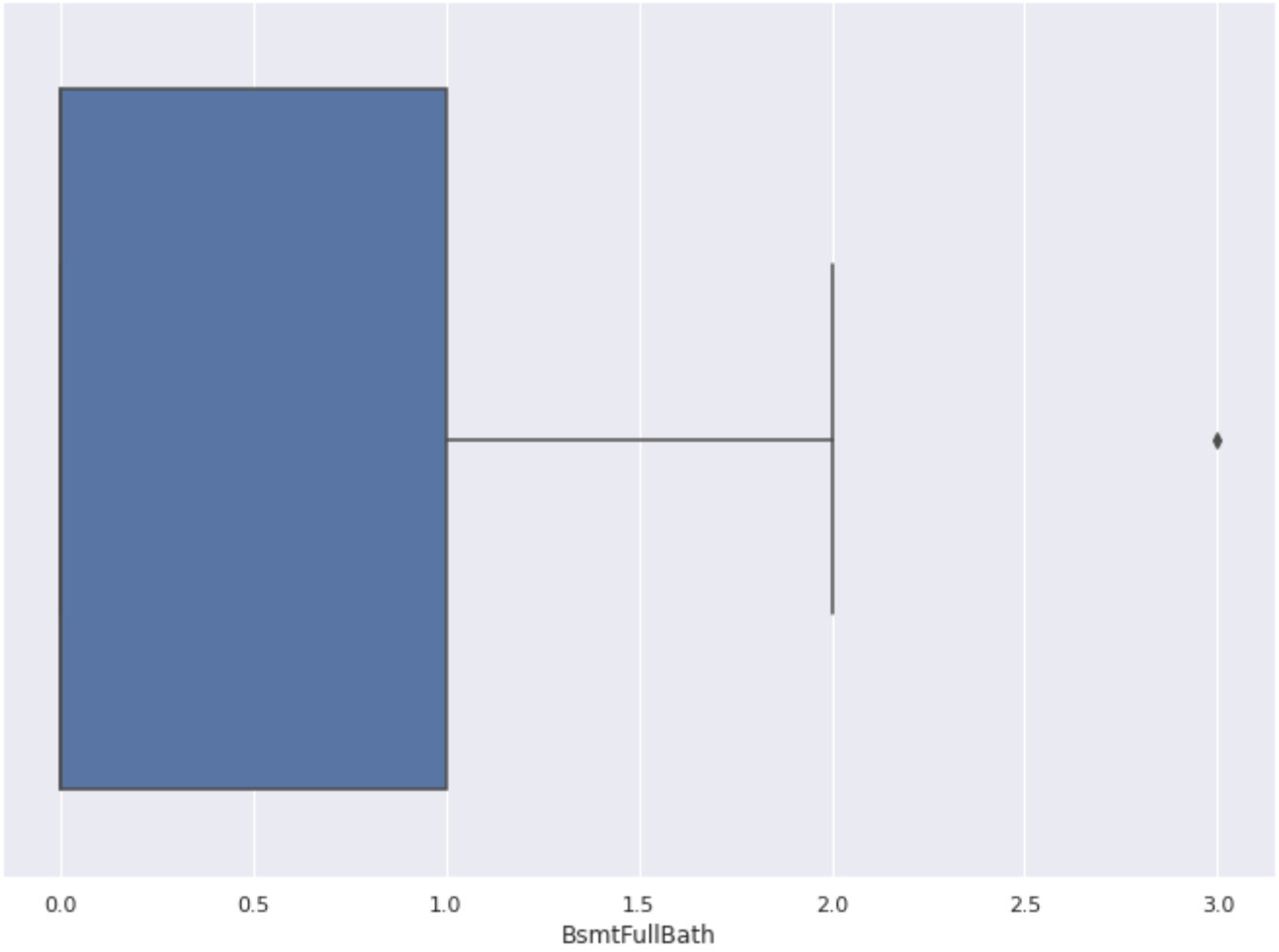

0
1000
2000
3000
4000
5000
6000
TotalBsmtSF

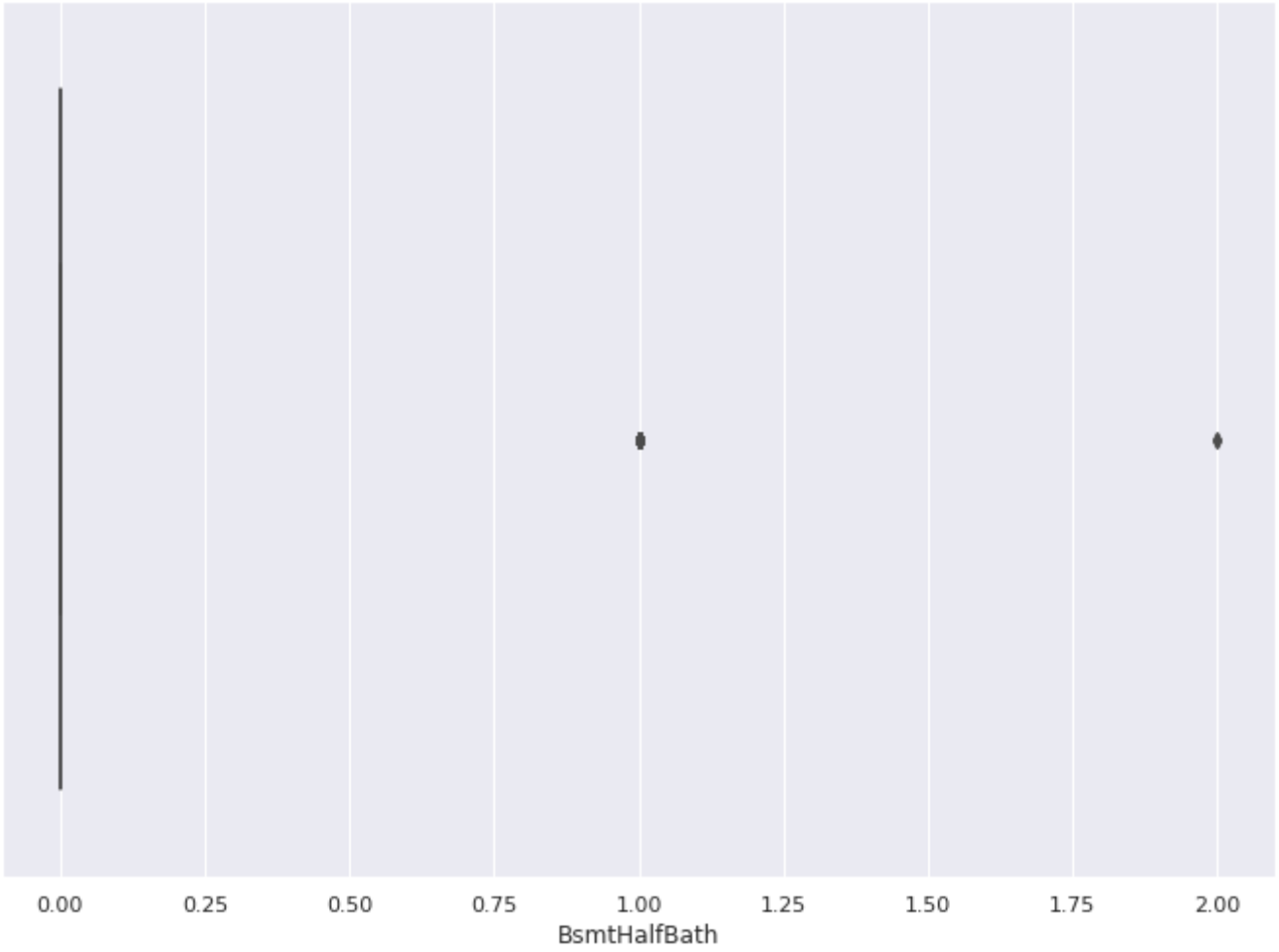

1000
2000
3000
4000
1stFlrSF

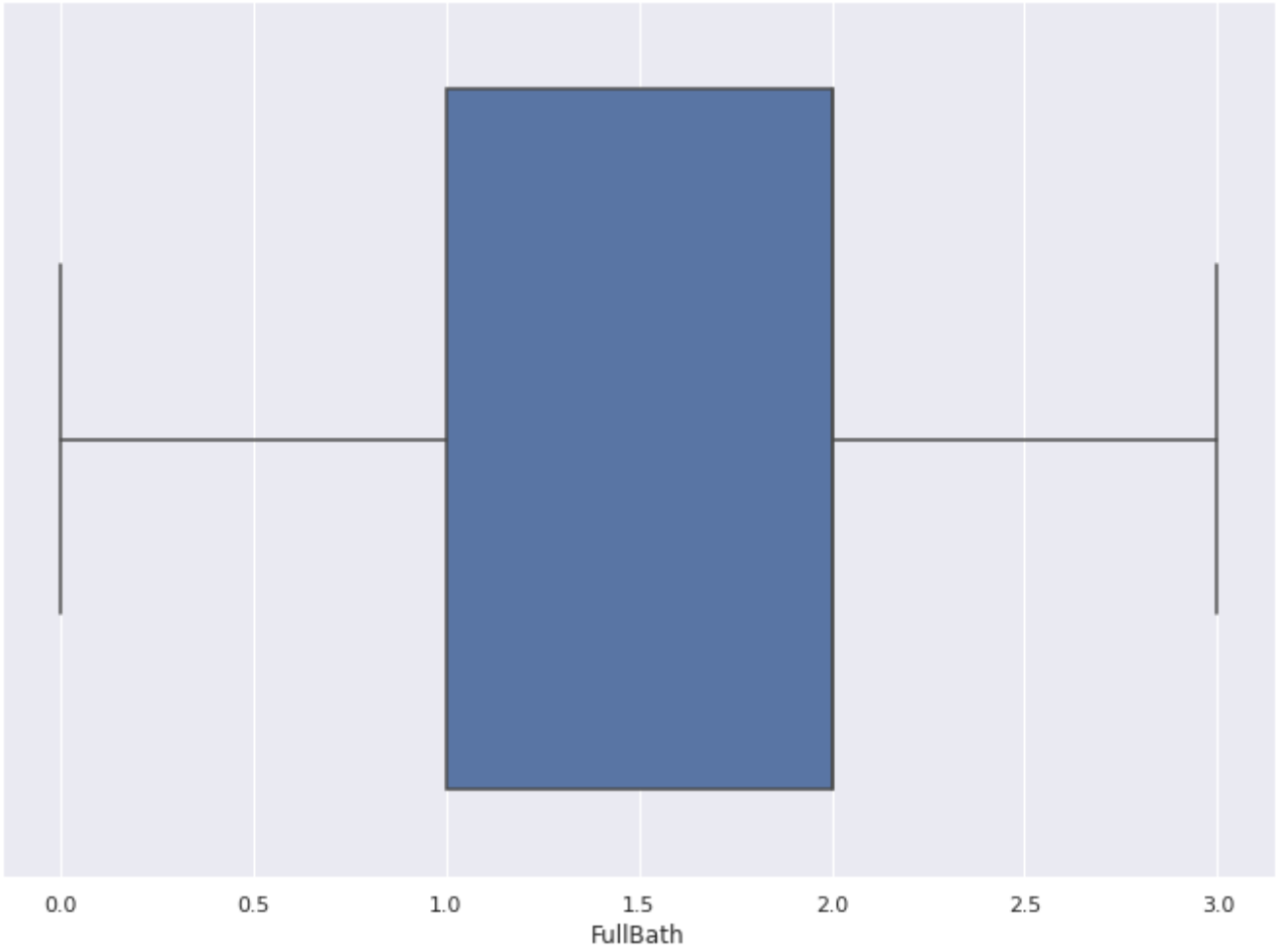

0
500
1000
1500
2000
2ndFlrSF

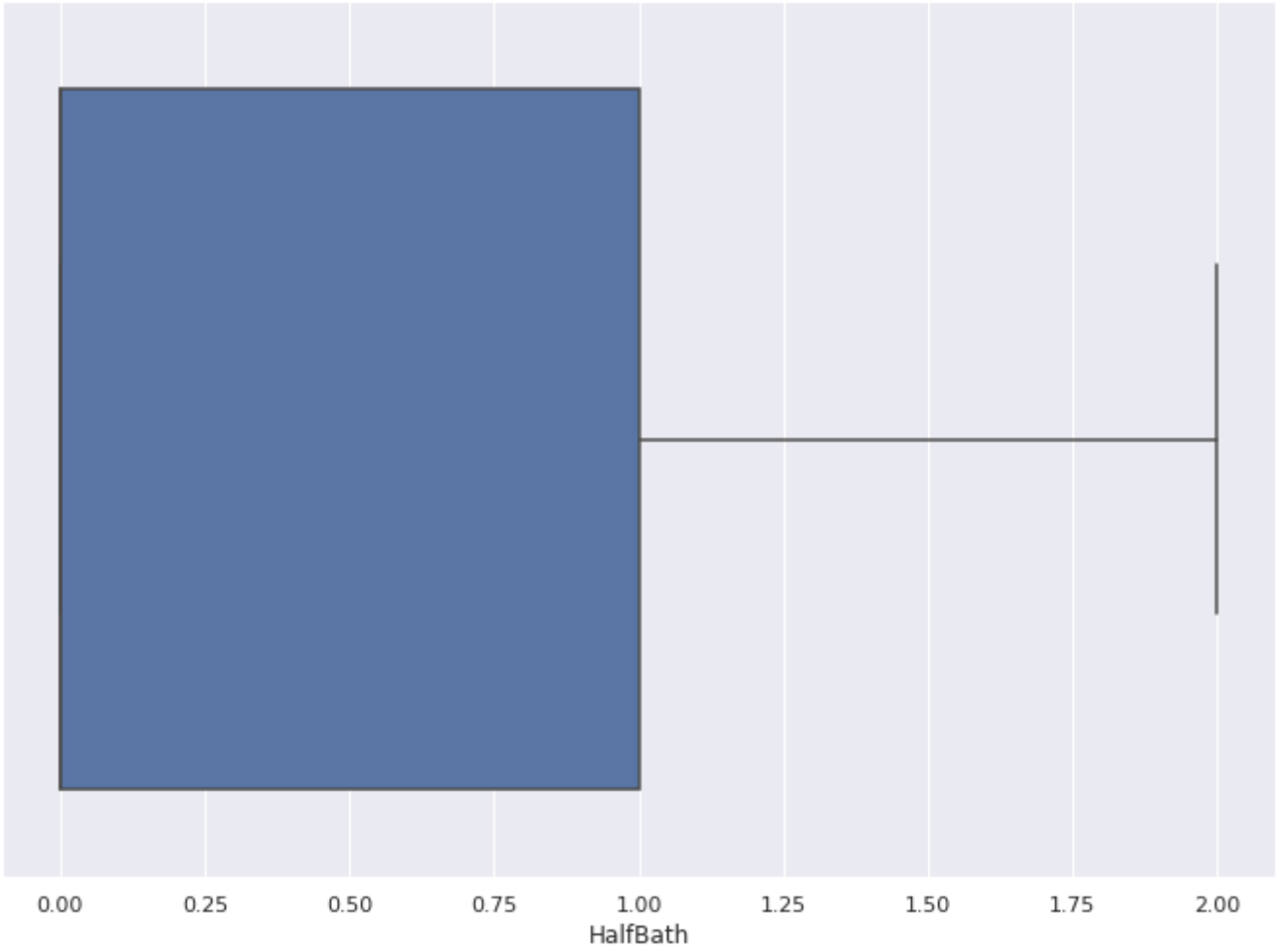

0
100
200
300
400
500
600
LowQualFinSF

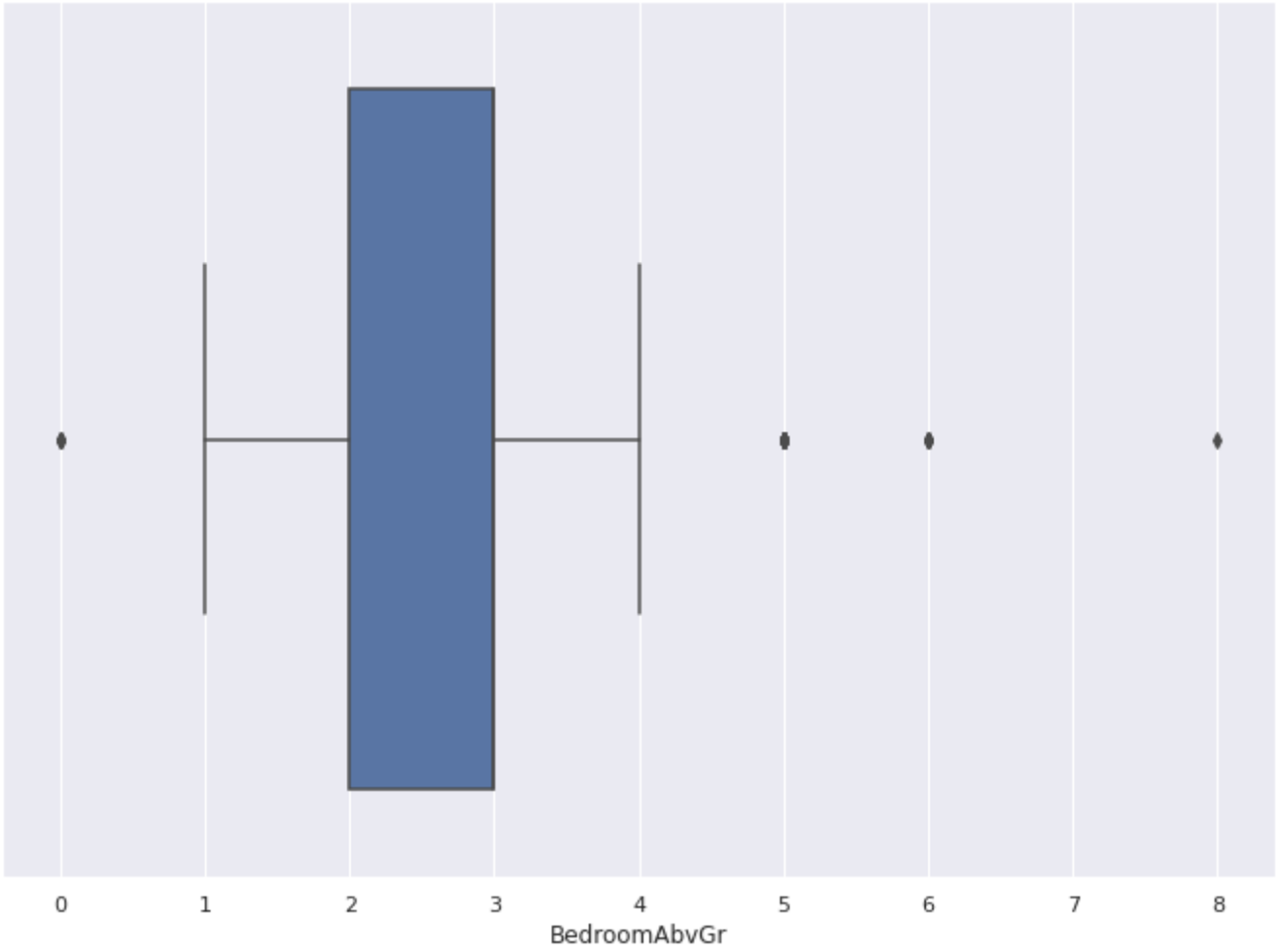

1000
2000
3000
4000
5000
GrLivArea

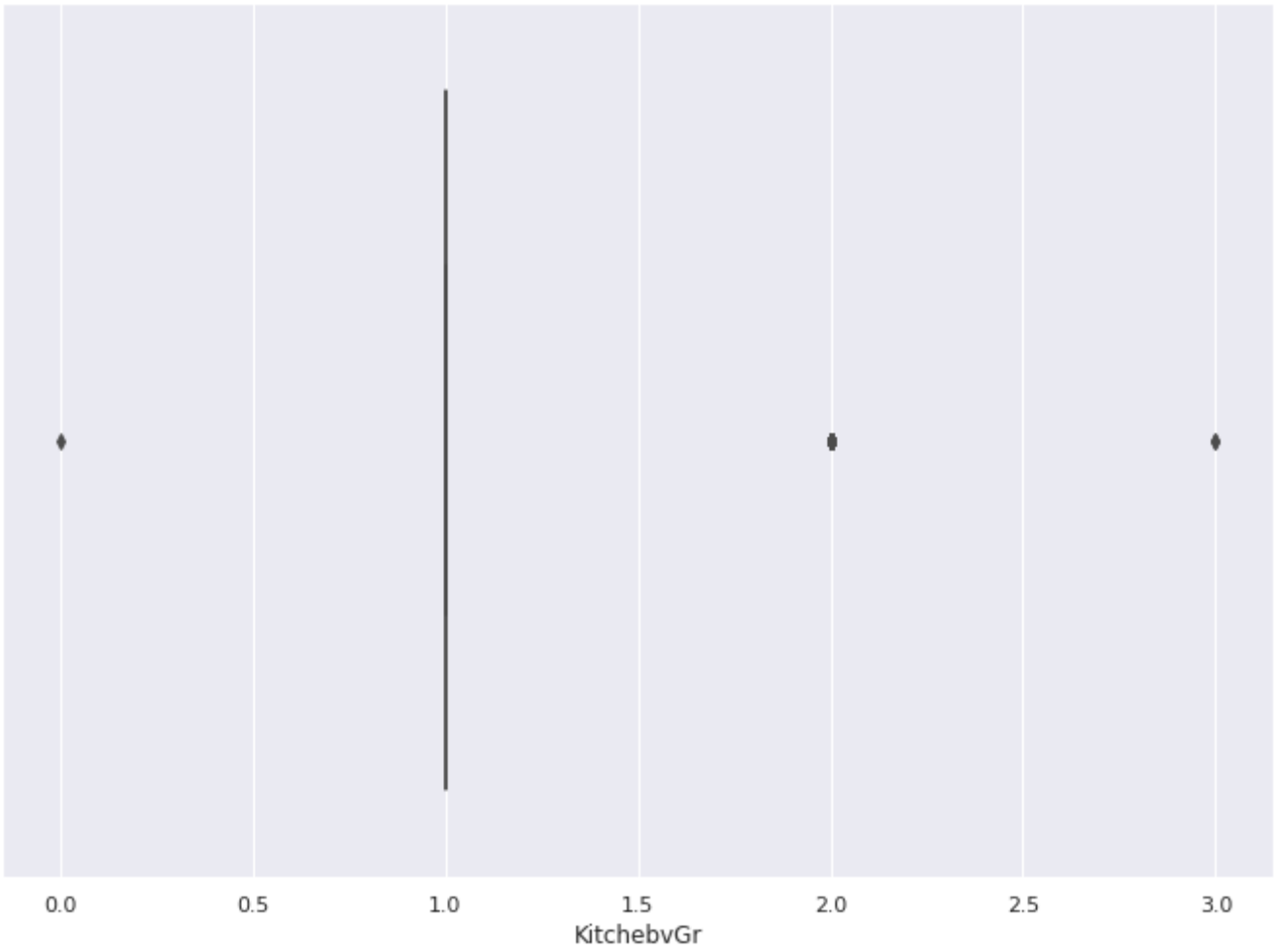

0.0
0.5
1.0
1.5
2.0
2.5
3.0
BsmtFullBath

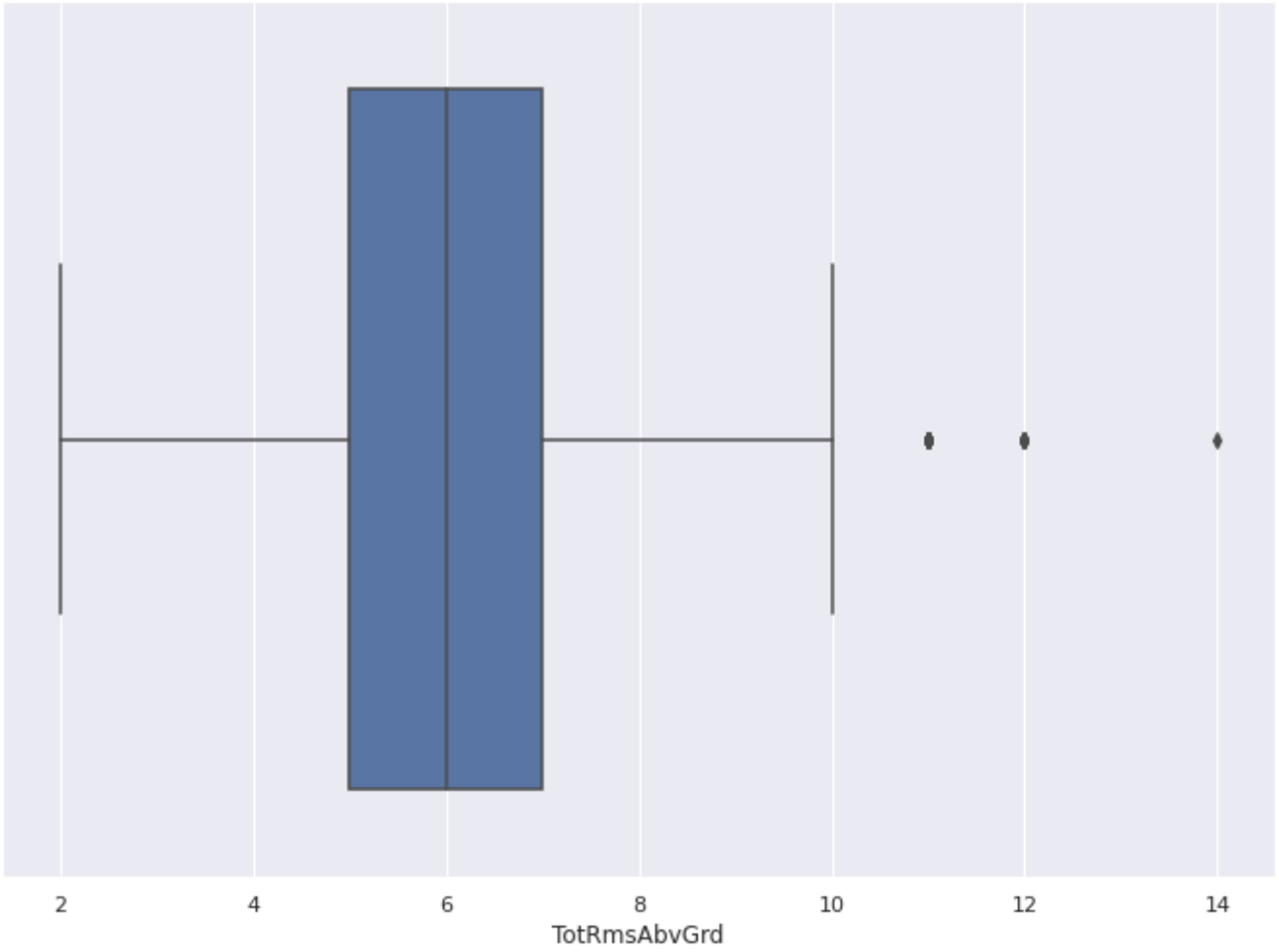

0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00
BsmtHalfBath

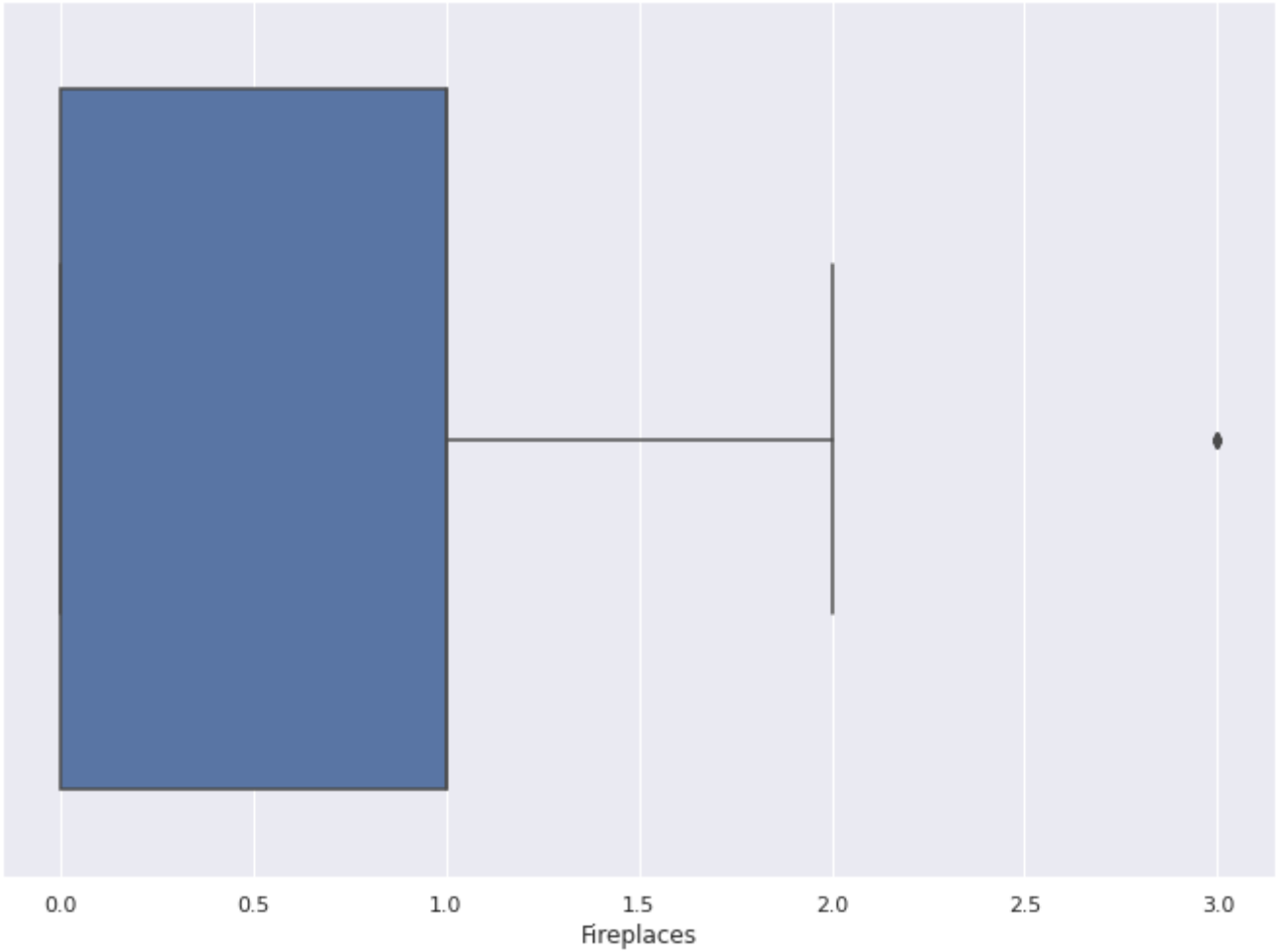

0.0
0.5
1.0
1.5
2.0
2.5
3.0
FullBath

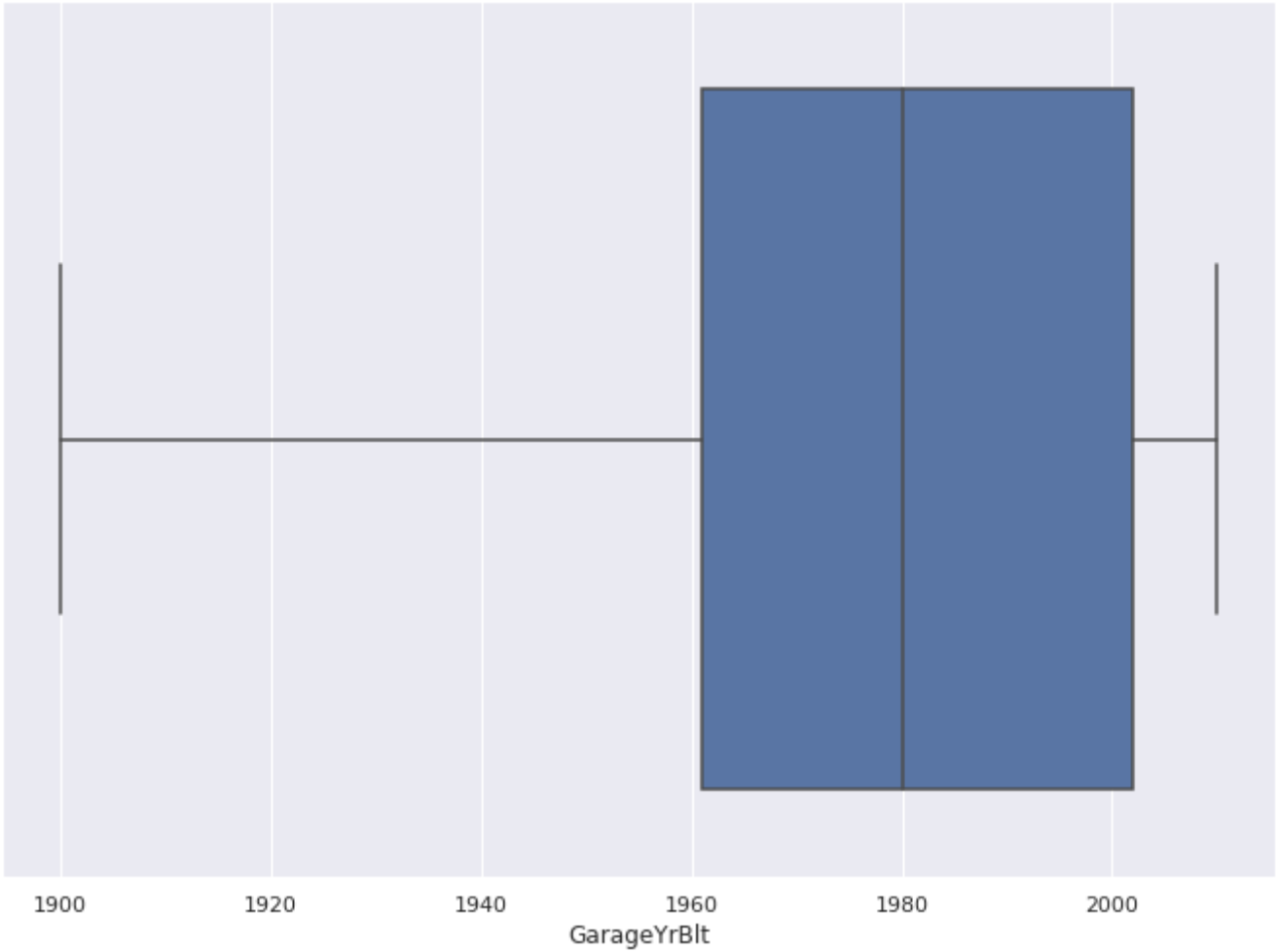

0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00
HalfBath

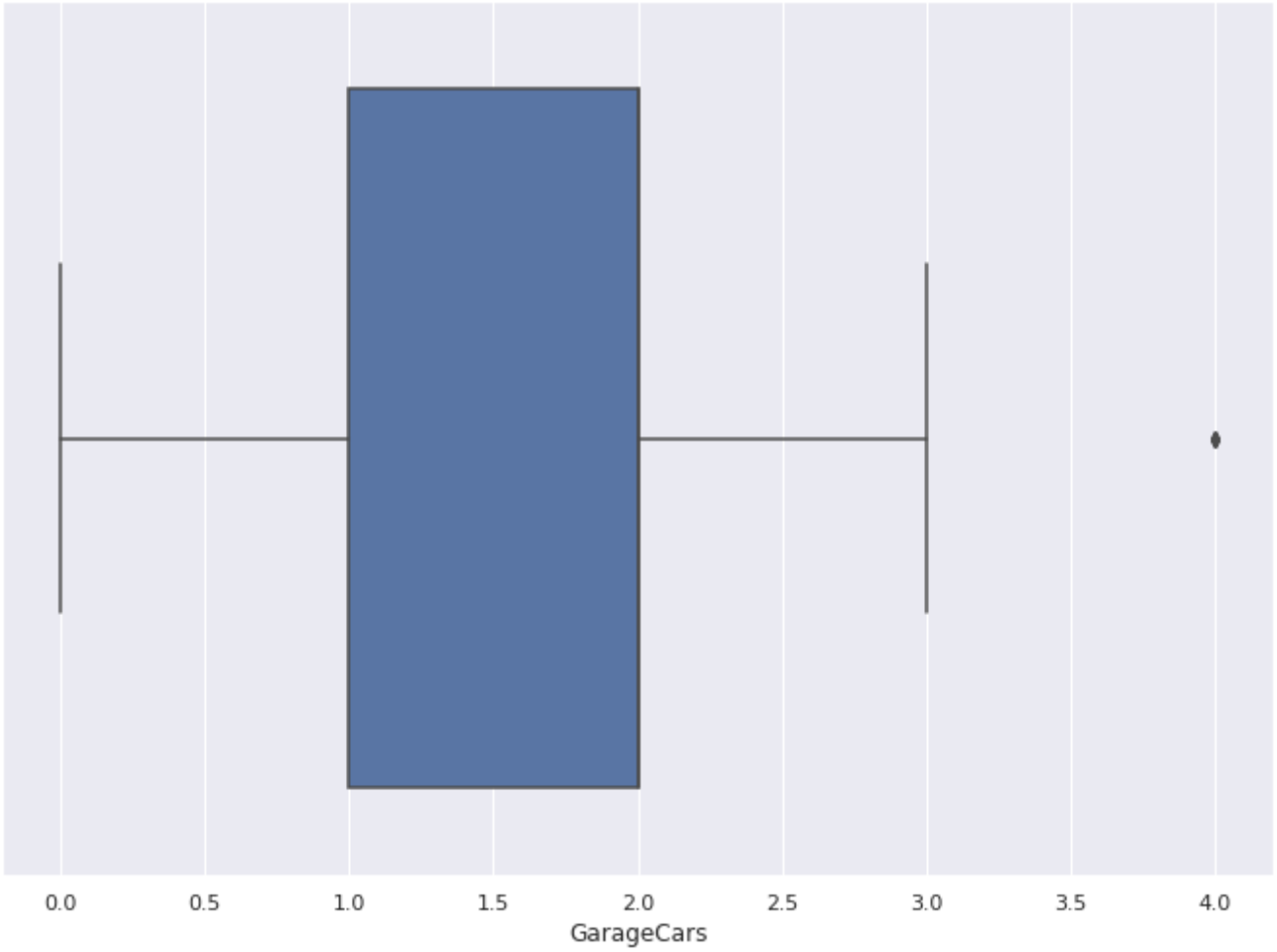

0
1
2
3
4
5
6
7
8
BedroomAbvGr

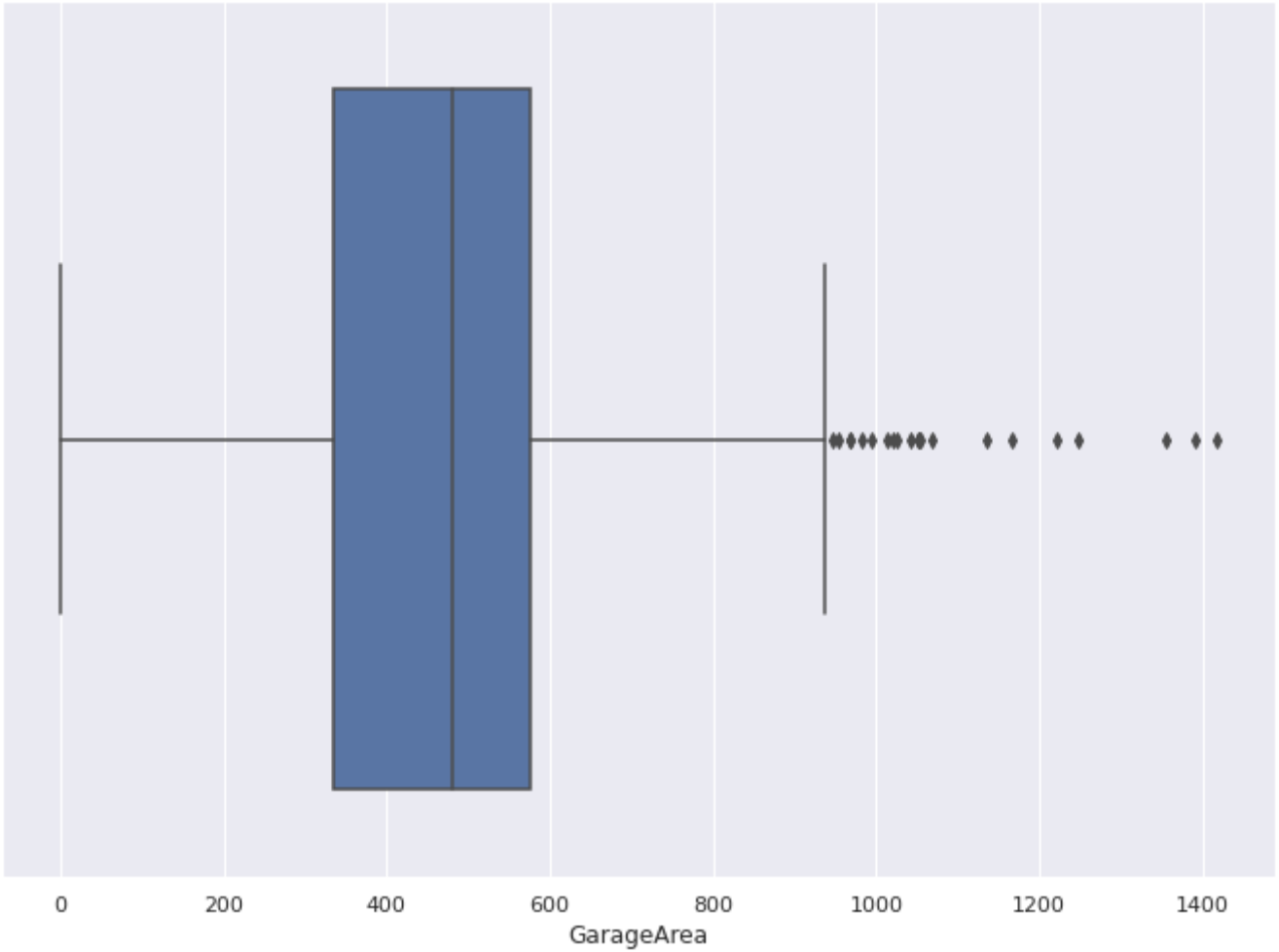

0.0
0.5
1.0
1.5
2.0
2.5
3.0
KitchebvGr

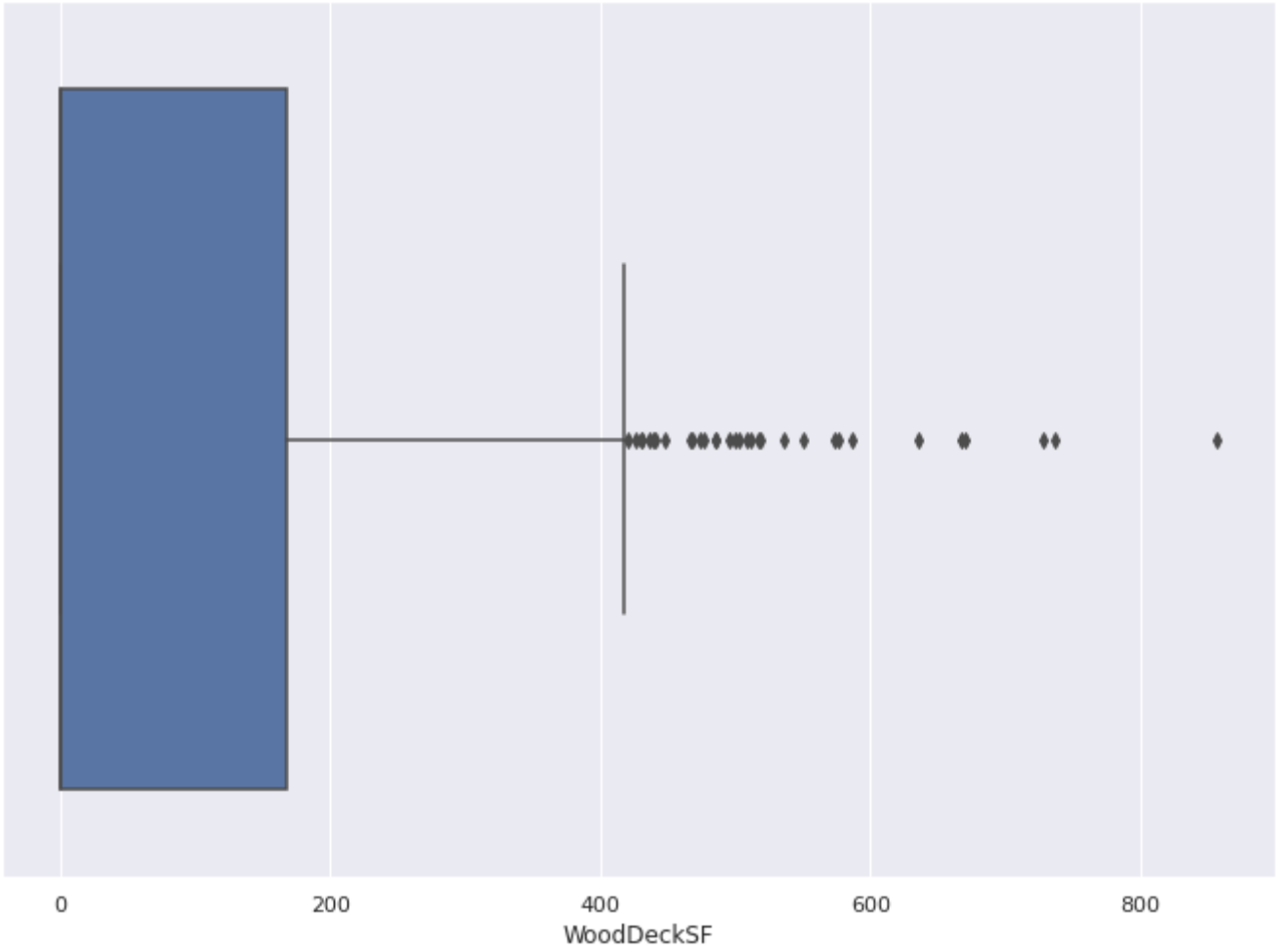

2
4
6
8
10
12
14
TotRmsAbvGrd

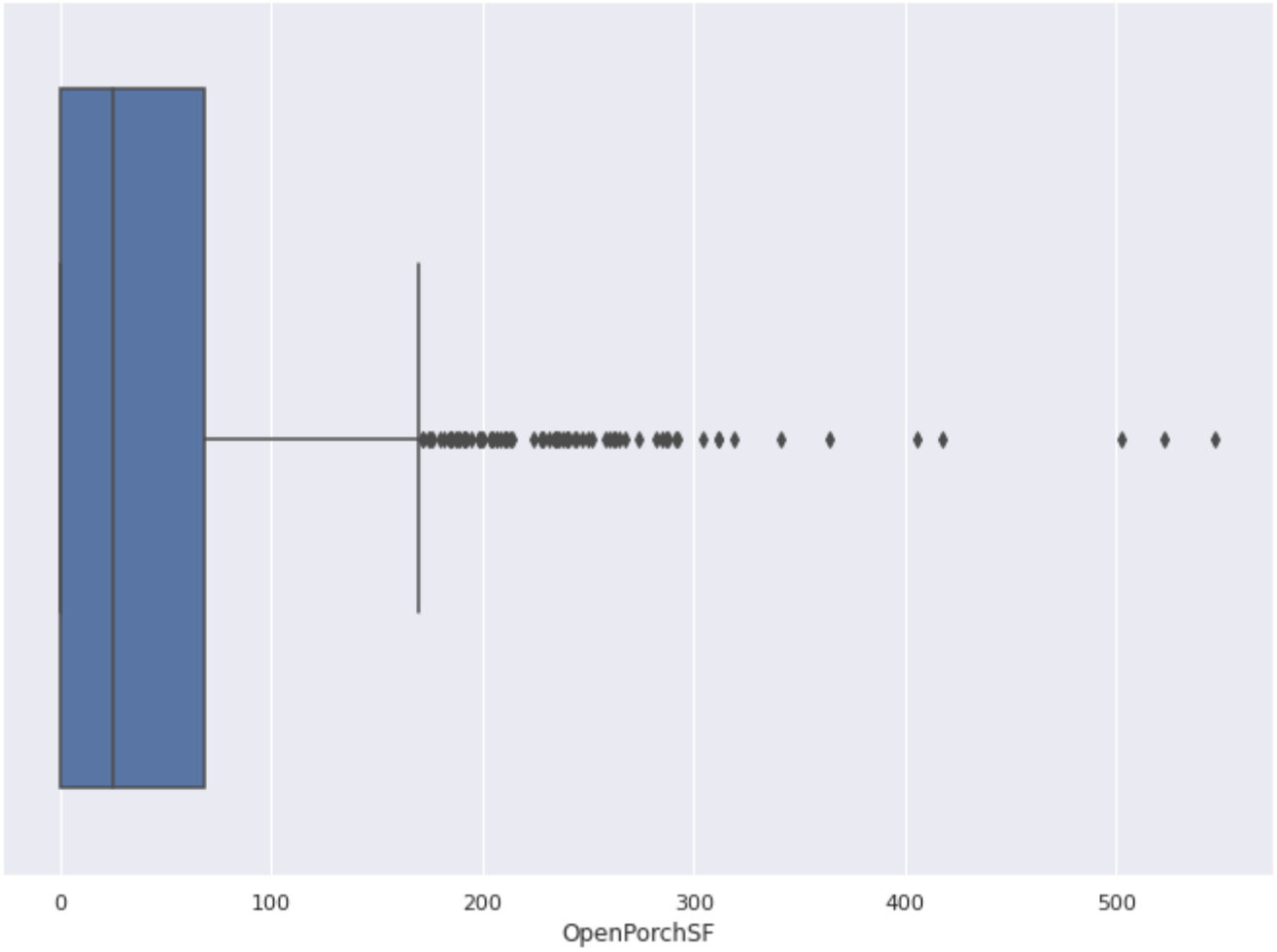

0.0
0.5
1.0
1.5
2.0
2.5
3.0
Fireplaces

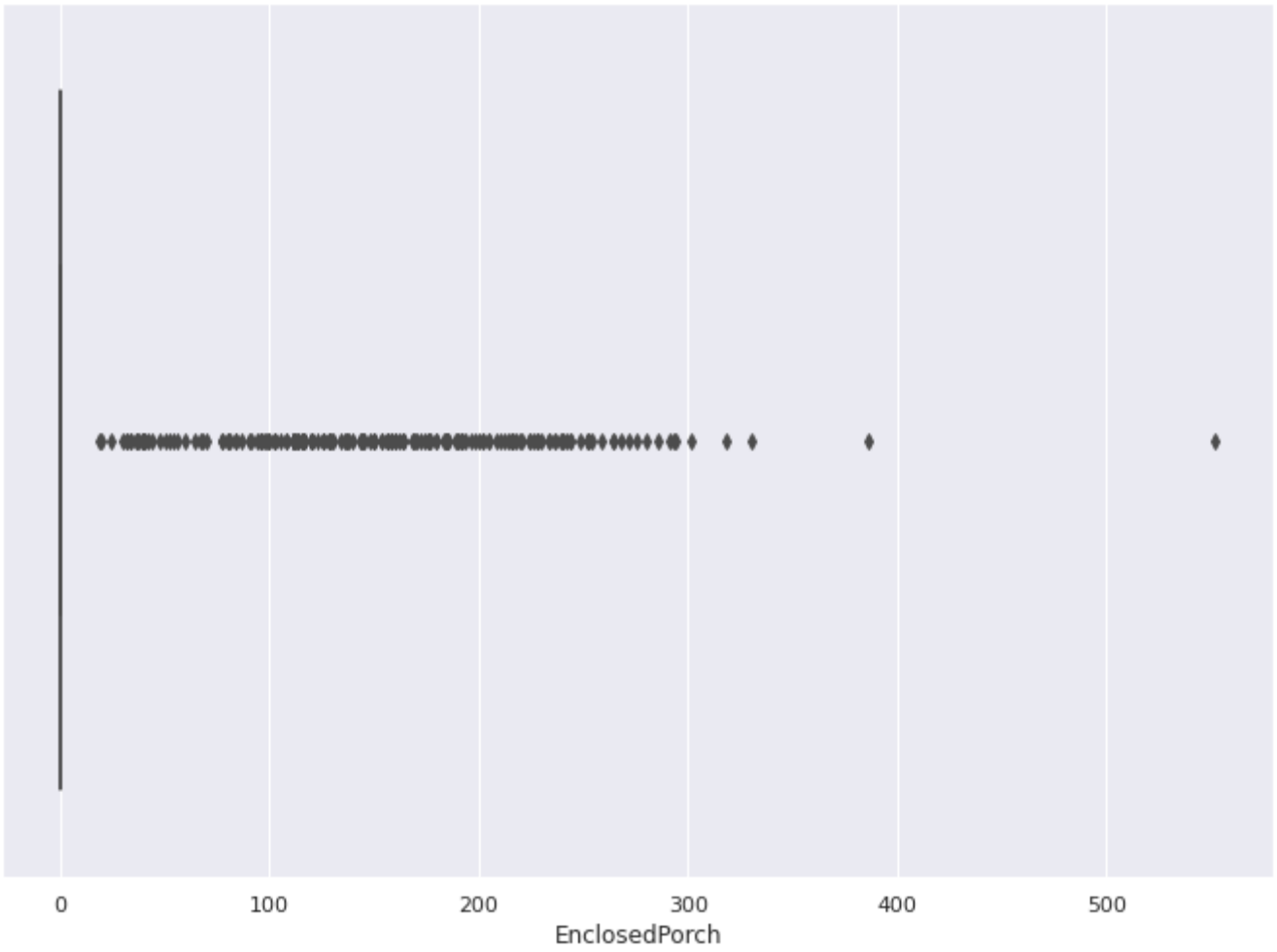

1900
1920
1940
1960
1980
2000
GarageYrBlt


0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
GarageCars

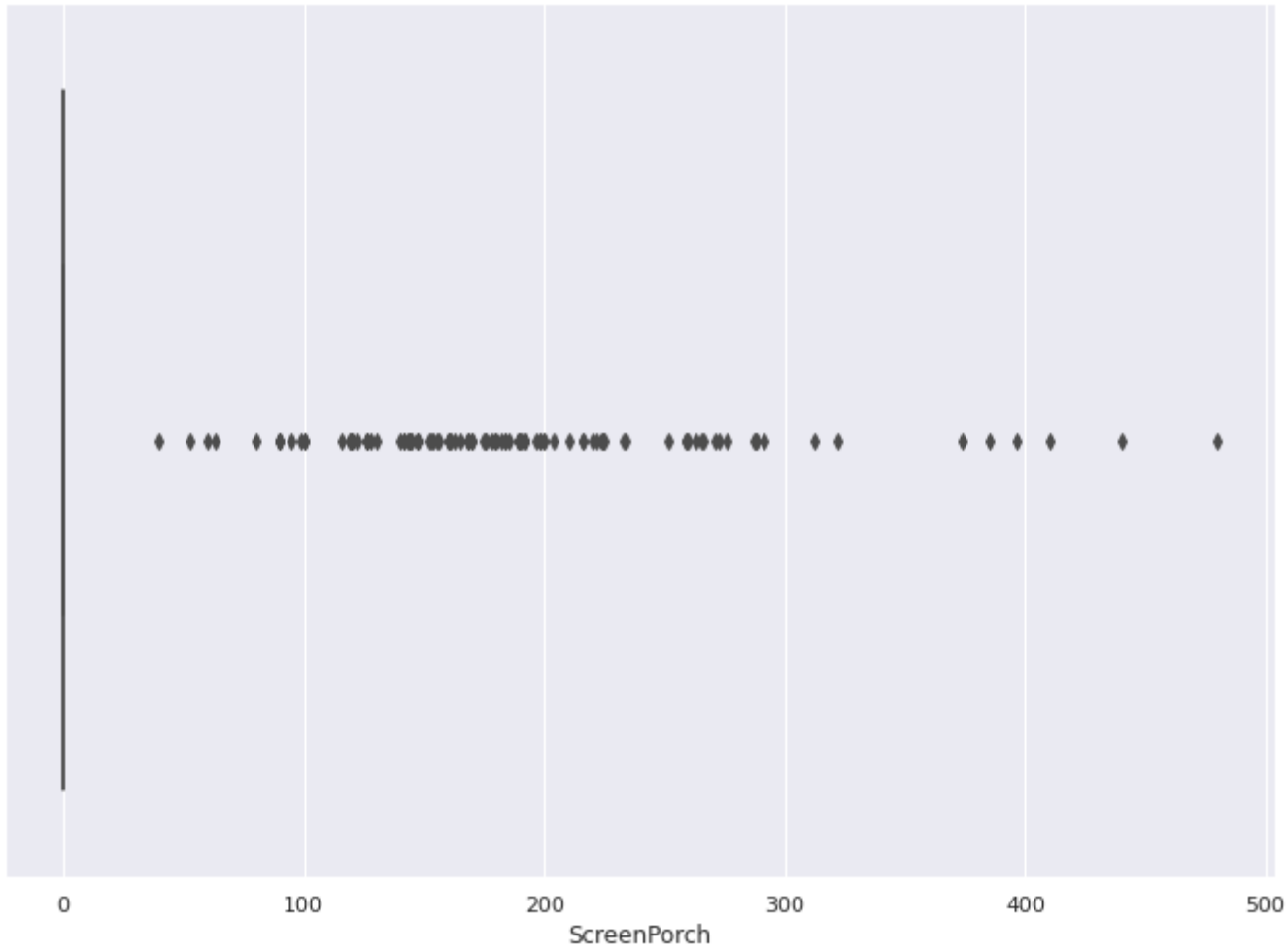

0
200
400
600
800
1000
1200
1400
GarageArea

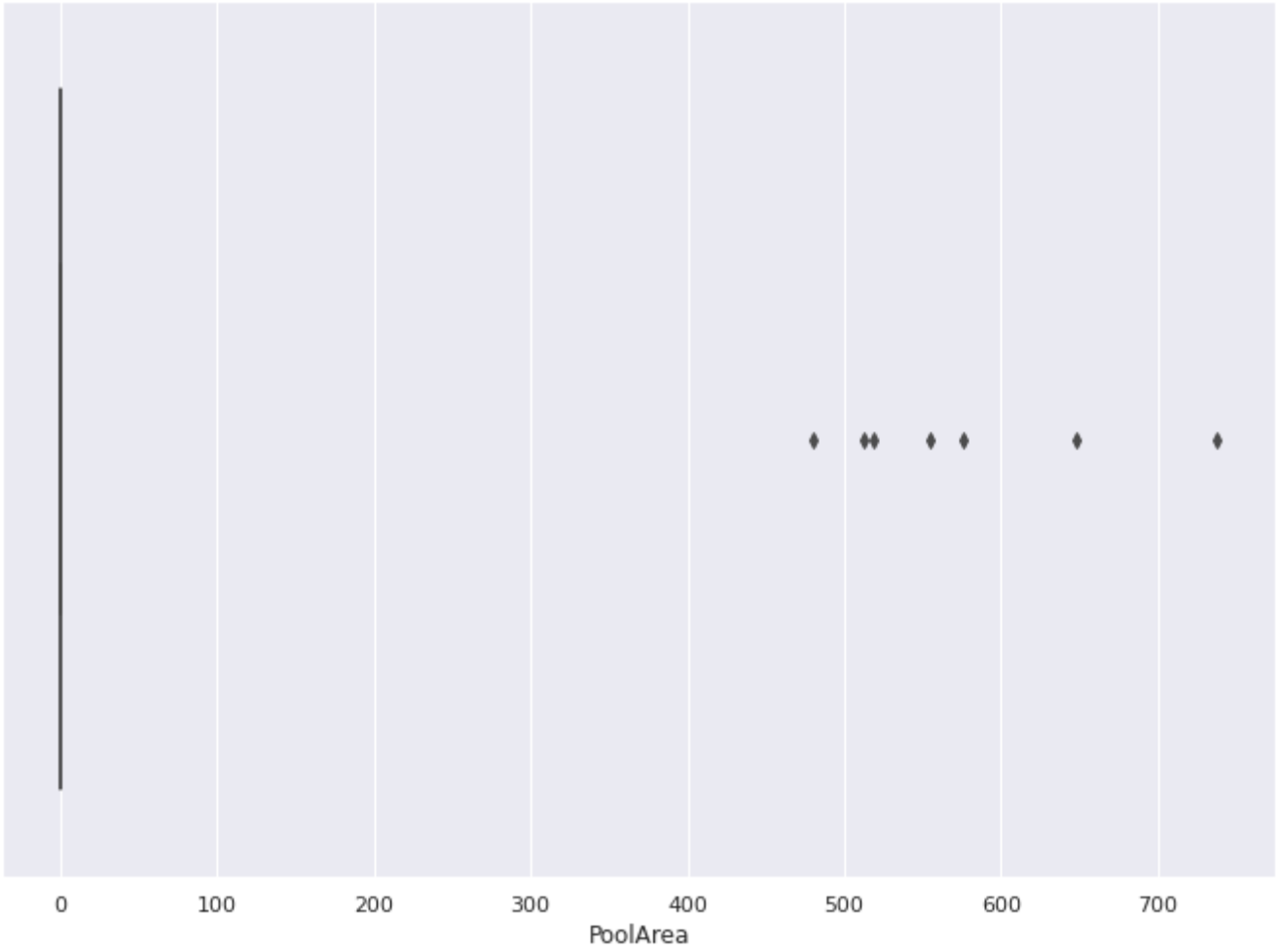

0
200
400
600
800
WoodDeckSF

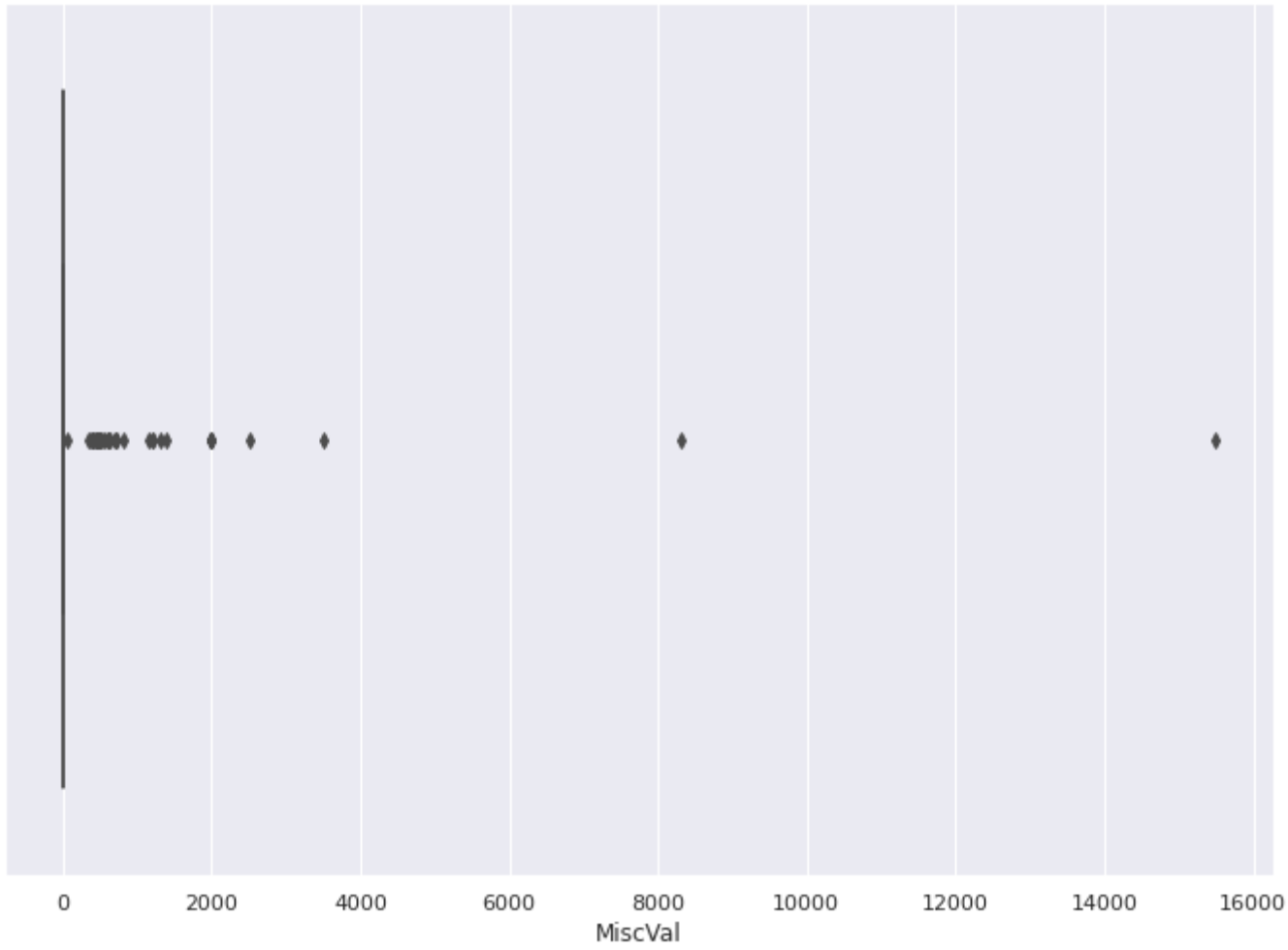

0
100
200
300
400
500
OpenPorchSF

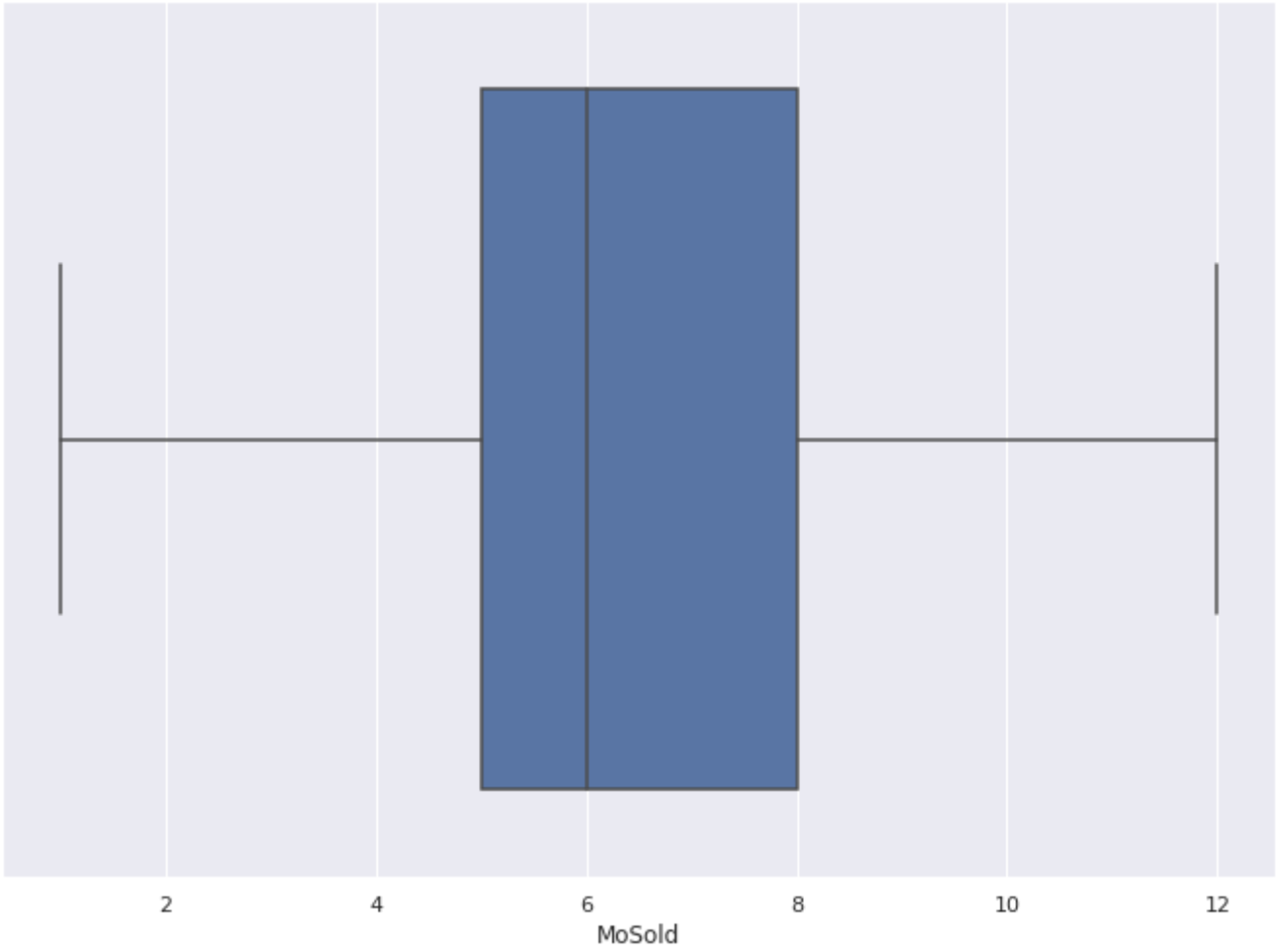

0
100
200
300
400
500
EnclosedPorch

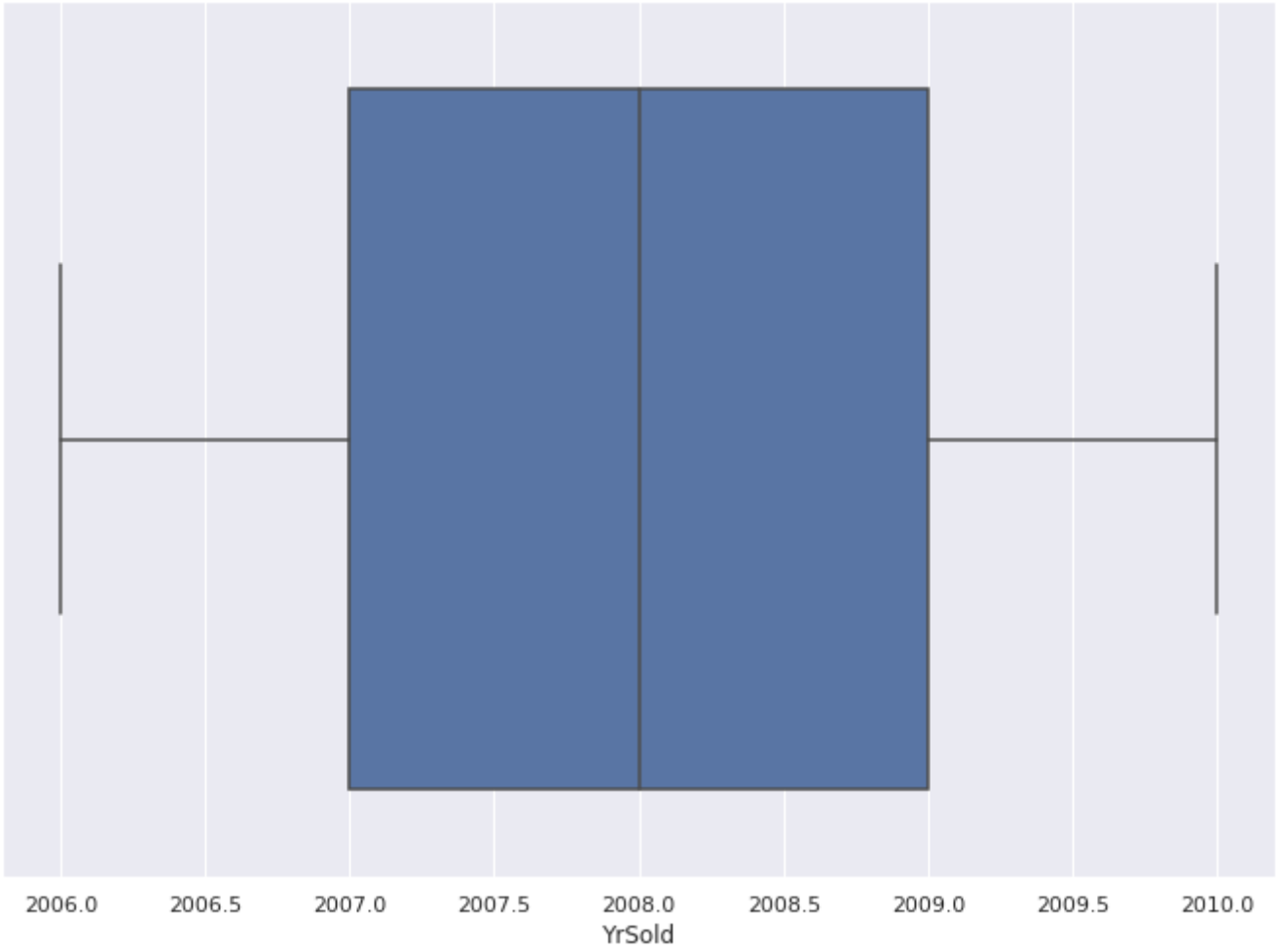

0
100
200
300
400
500
3SsnPorch

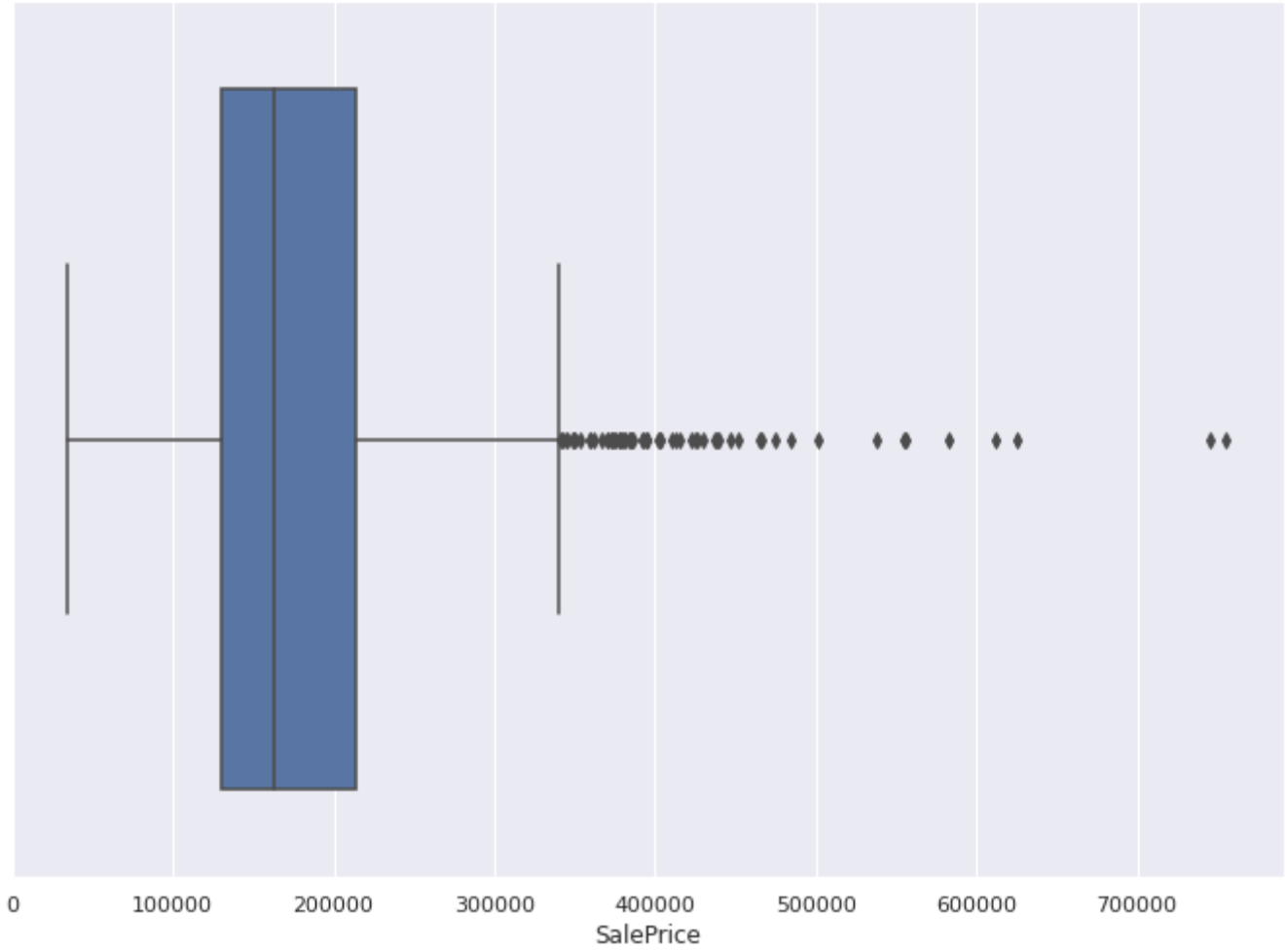

0
100
200
300
400
500
ScreenPorch


0
100
200
300
400
500
600
700
PoolArea


0
2000
4000
6000
8000
10000
12000
14000
16000
MiscVal


2
4
6
8
10
12
MoSold


2006.0
2006.5
2007.0
2007.5
2008.0
2008.5
2009.0
2009.5
2010.0
YrSold

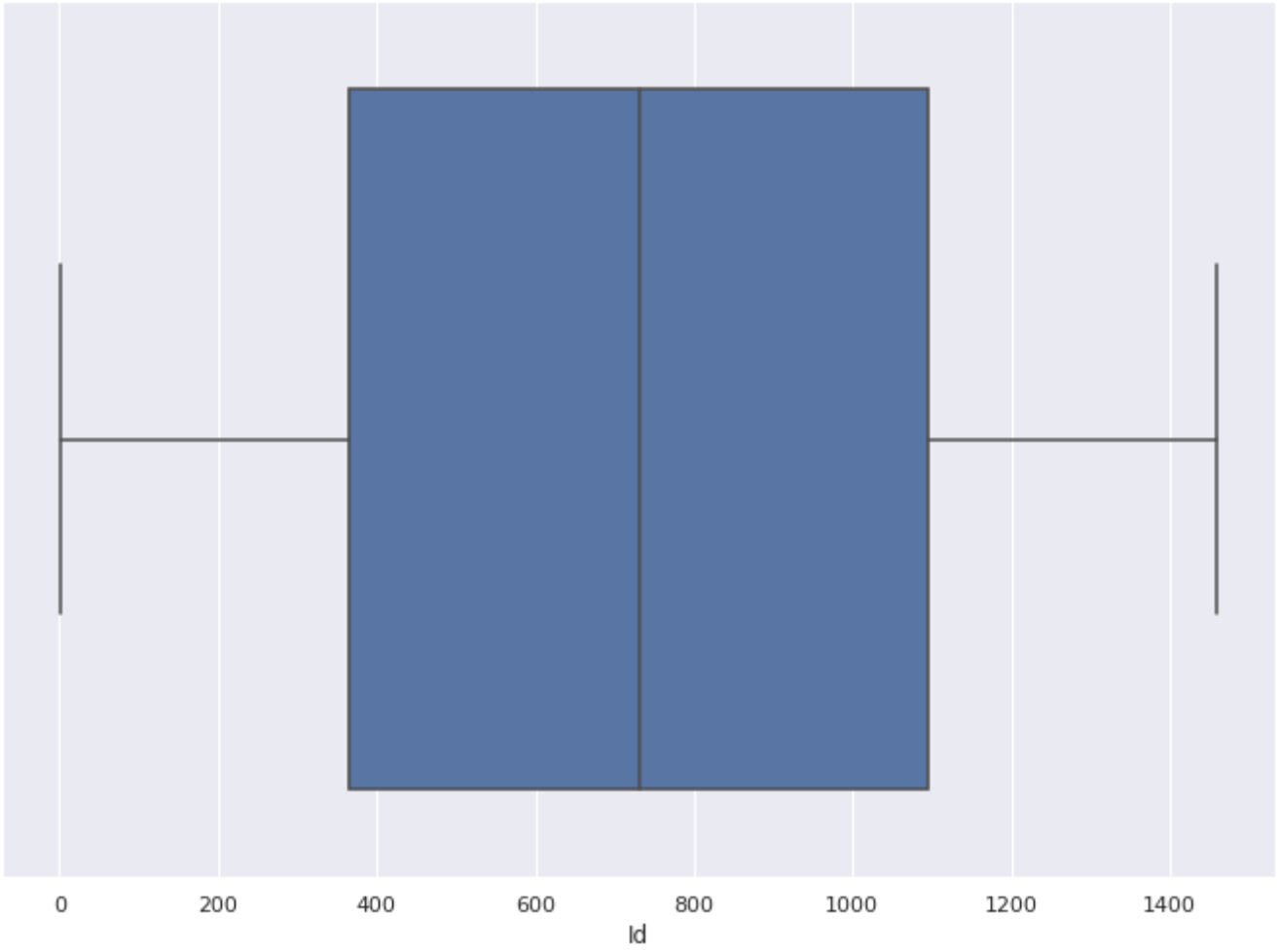

0
100000
200000
300000
400000
500000
600000
700000
SalePrice

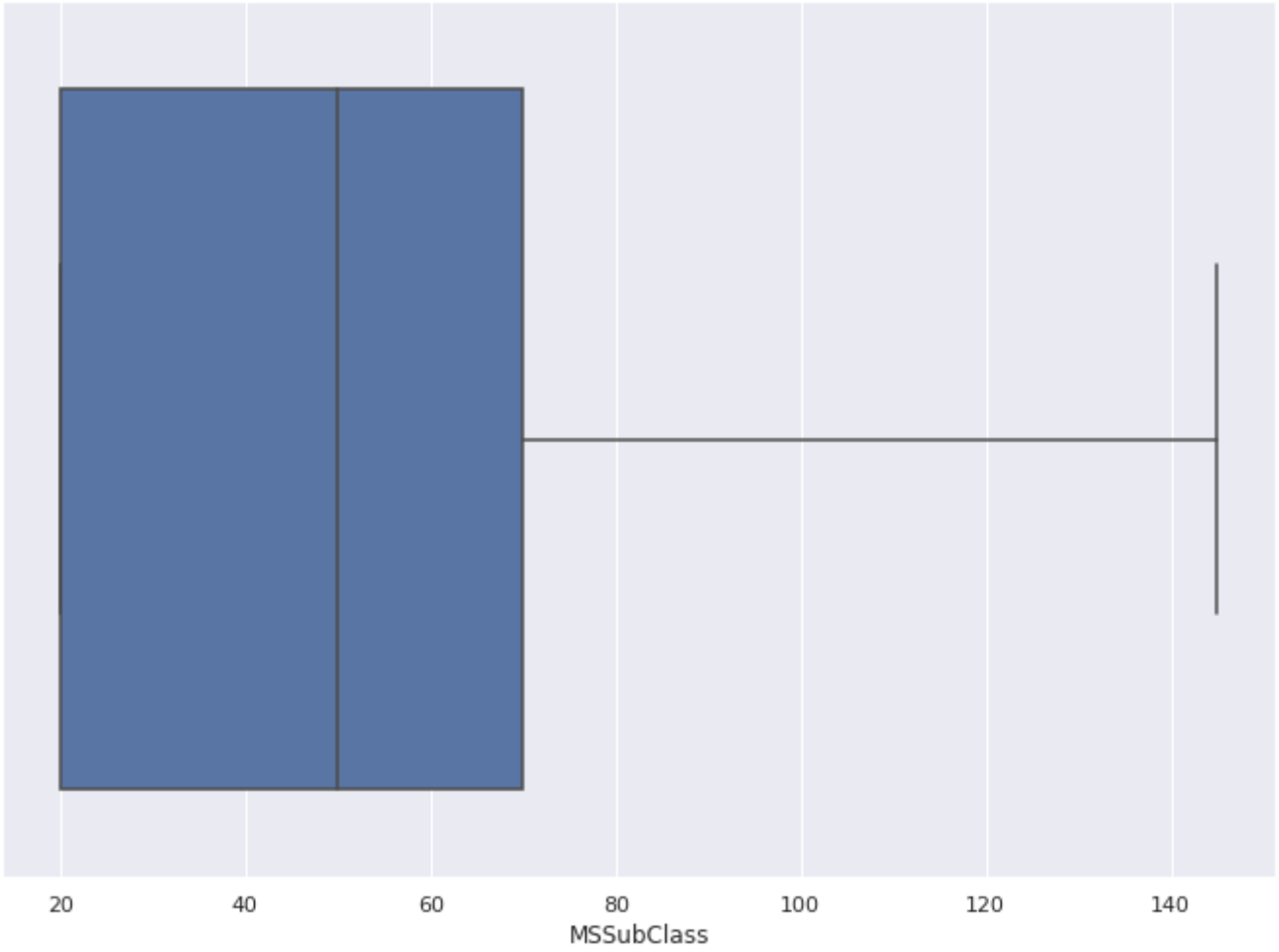
In [41]:
```
def remove_outliers(df,columns):
    # df = dataframe
    # column takes a list of numerical columns

    for col in columns:
        print('Working on column: {}'.format(col))
        if (df[col].dtype != object) :
            q1,q3 = np.percentile(df[col], [25,75])
            iqr = q3-q1
            minv = q1-(1.5*iqr)
            maxv = q3+(1.5*iqr)
            med  = df[col].median()
            #data[col] = data[col].apply(lambda x: maxv if x>maxv else minv if x<minv else x)
            df[col] = np.where(df[col]>maxv , maxv, df[col])
            df[col] = np.where(df[col]<minv , minv, df[col])

    return df
```

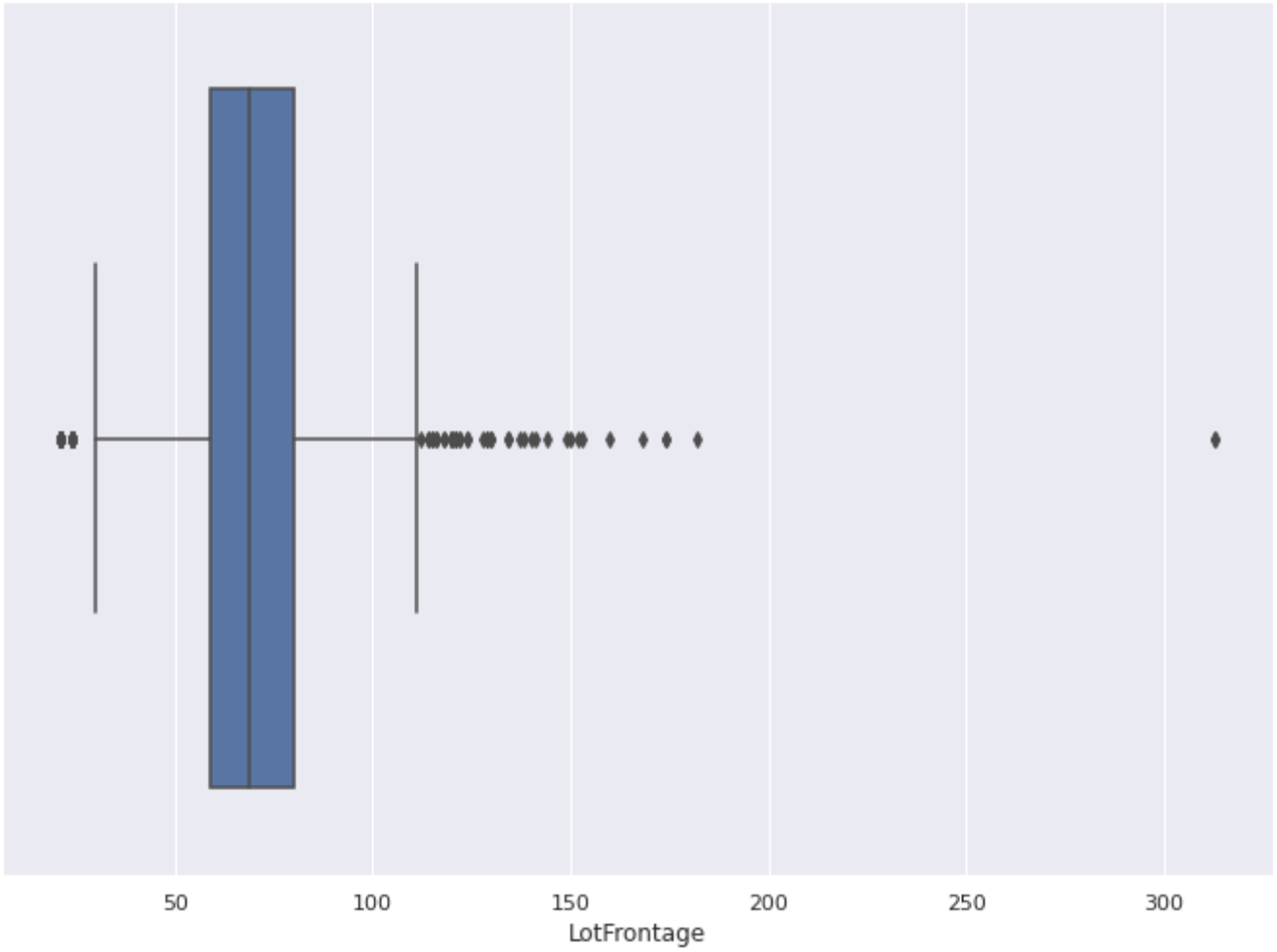
In [42]:
```
df2 = remove_outliers(df, df.describe().columns)
```

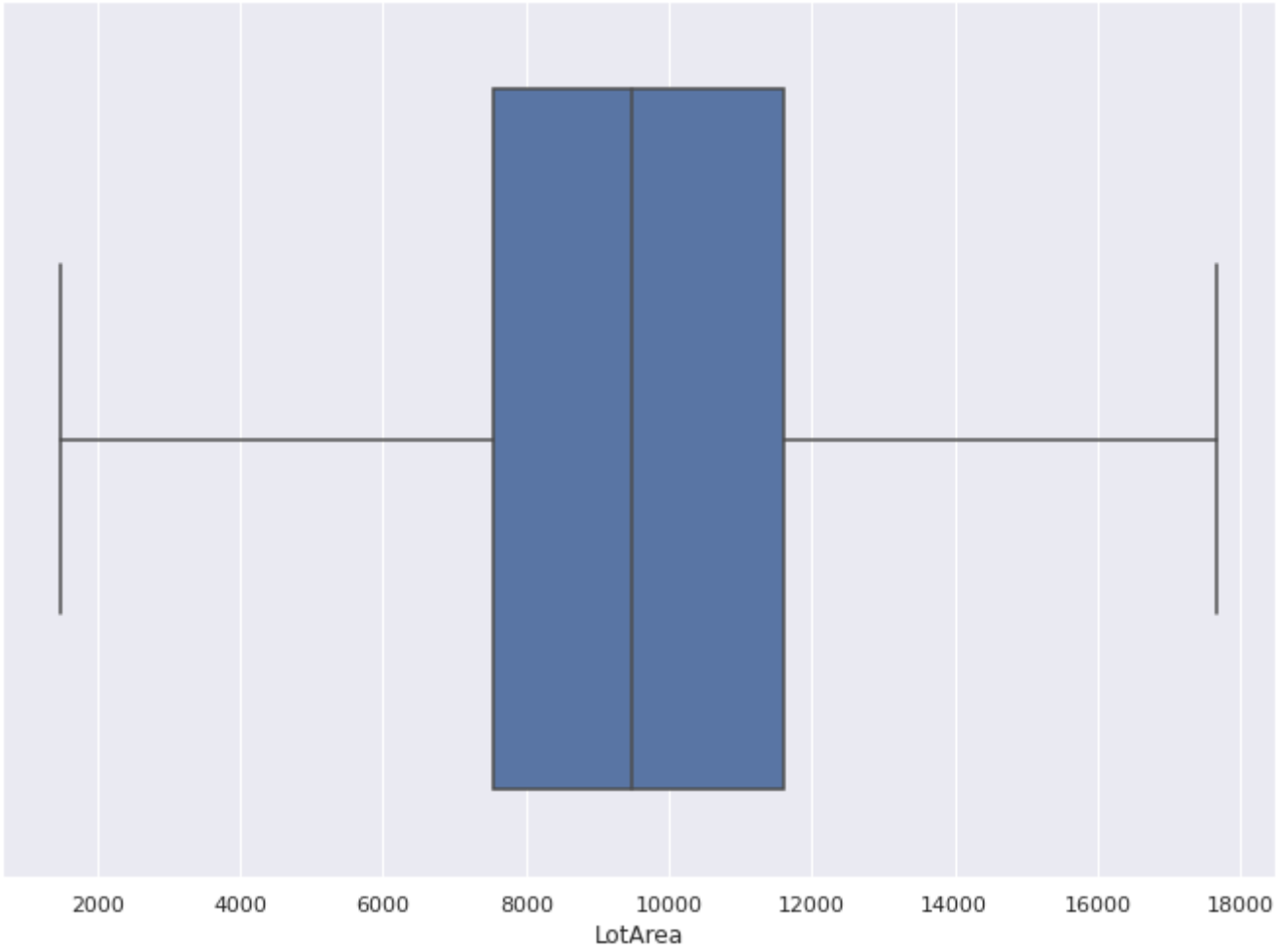
```
Working on column: Id
Working on column: MSSubClass
Working on column: LotFrontage
Working on column: LotArea
Working on column: OverallQual
Working on column: OverallCond
Working on column: YearBuilt
Working on column: YearRemodAdd
Working on column: MasVnrArea
Working on column: BsmtFinSF1
Working on column: BsmtFinSF2
Working on column: BsmtUnfSF
Working on column: TotalBsmtSF
Working on column: 1stFlrSF
Working on column: 2ndFlrSF
Working on column: LowQualFinSF
Working on column: GrLivArea
Working on column: BsmtFullBath
Working on column: BsmtHalfBath
Working on column: FullBath
Working on column: HalfBath
Working on column: BedroomAbvGr
Working on column: KitchebvGr
Working on column: TotRmsAbvGrd
Working on column: Fireplaces
Working on column: GarageYrBlt
Working on column: GarageCars
Working on column: GarageArea
Working on column: WoodDeckSF
Working on column: OpenPorchSF
Working on column: EnclosedPorch
Working on column: 3SsnPorch
Working on column: ScreenPorch
Working on column: PoolArea
Working on column: MiscVal
Working on column: MoSold
Working on column: YrSold
Working on column: SalePrice
```

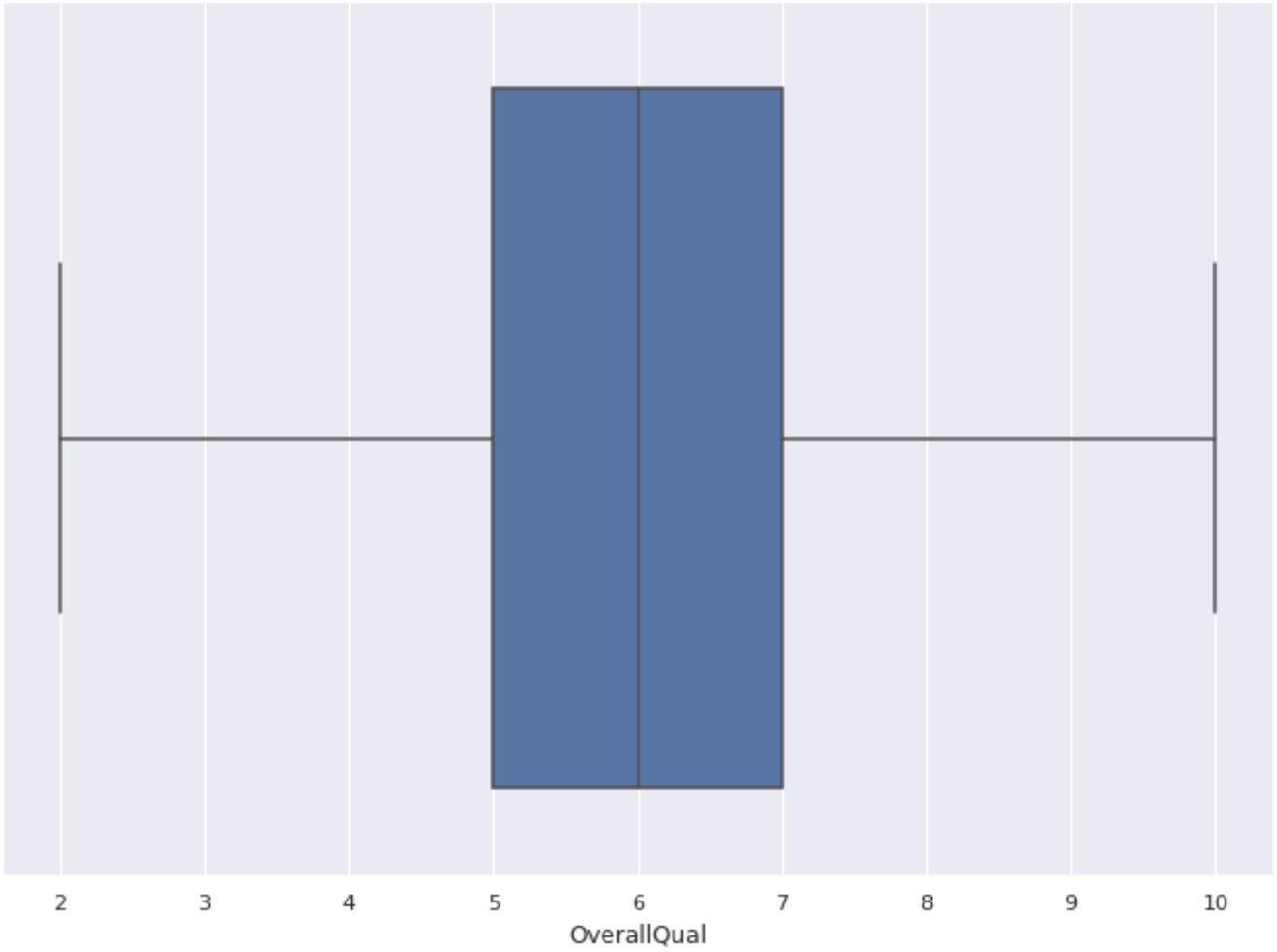
In [43]:
```
boxplotloop(df2, df2.describe().columns)
```

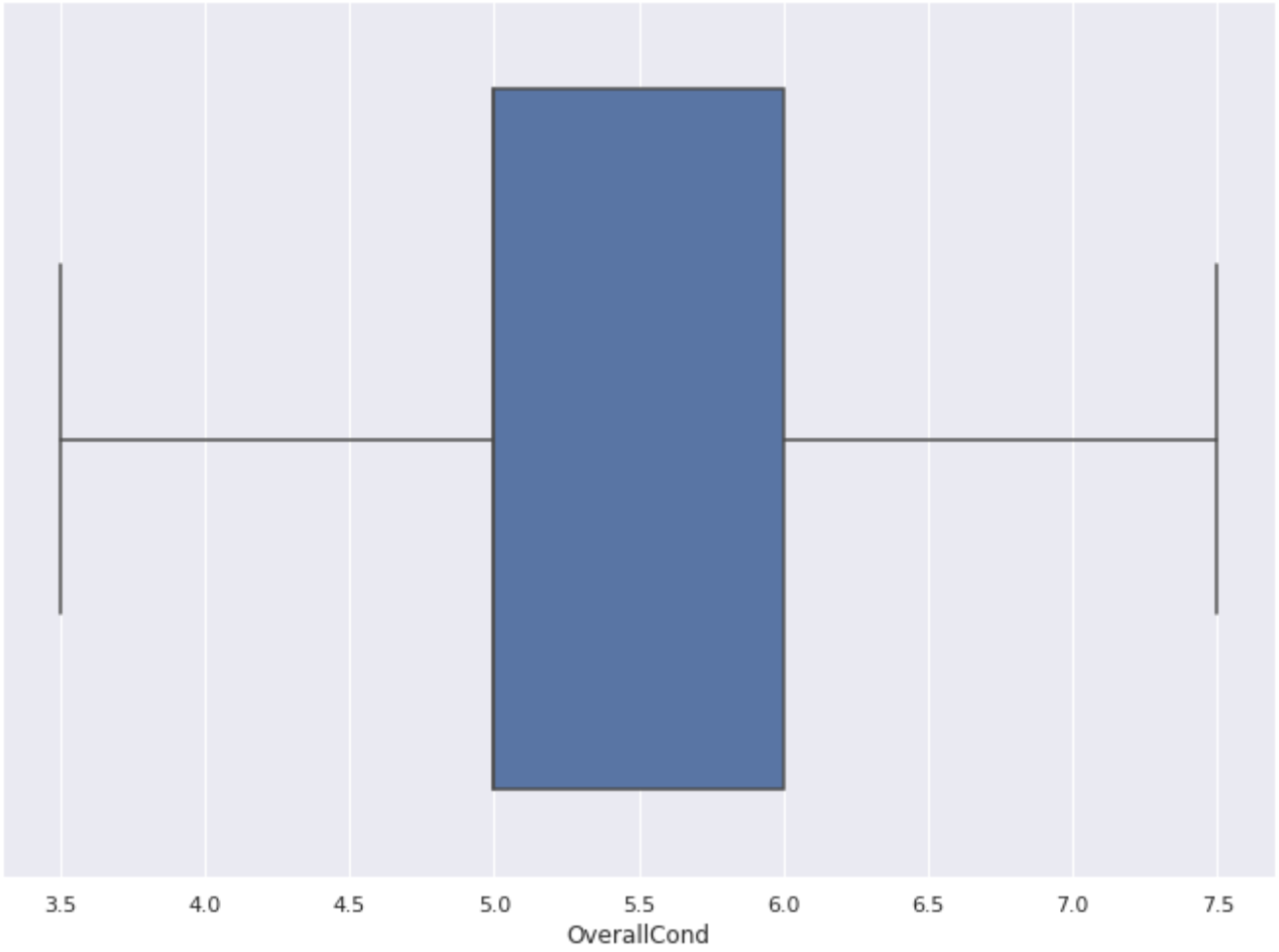

0
200
400
600
800
1000
1200
1400
Id

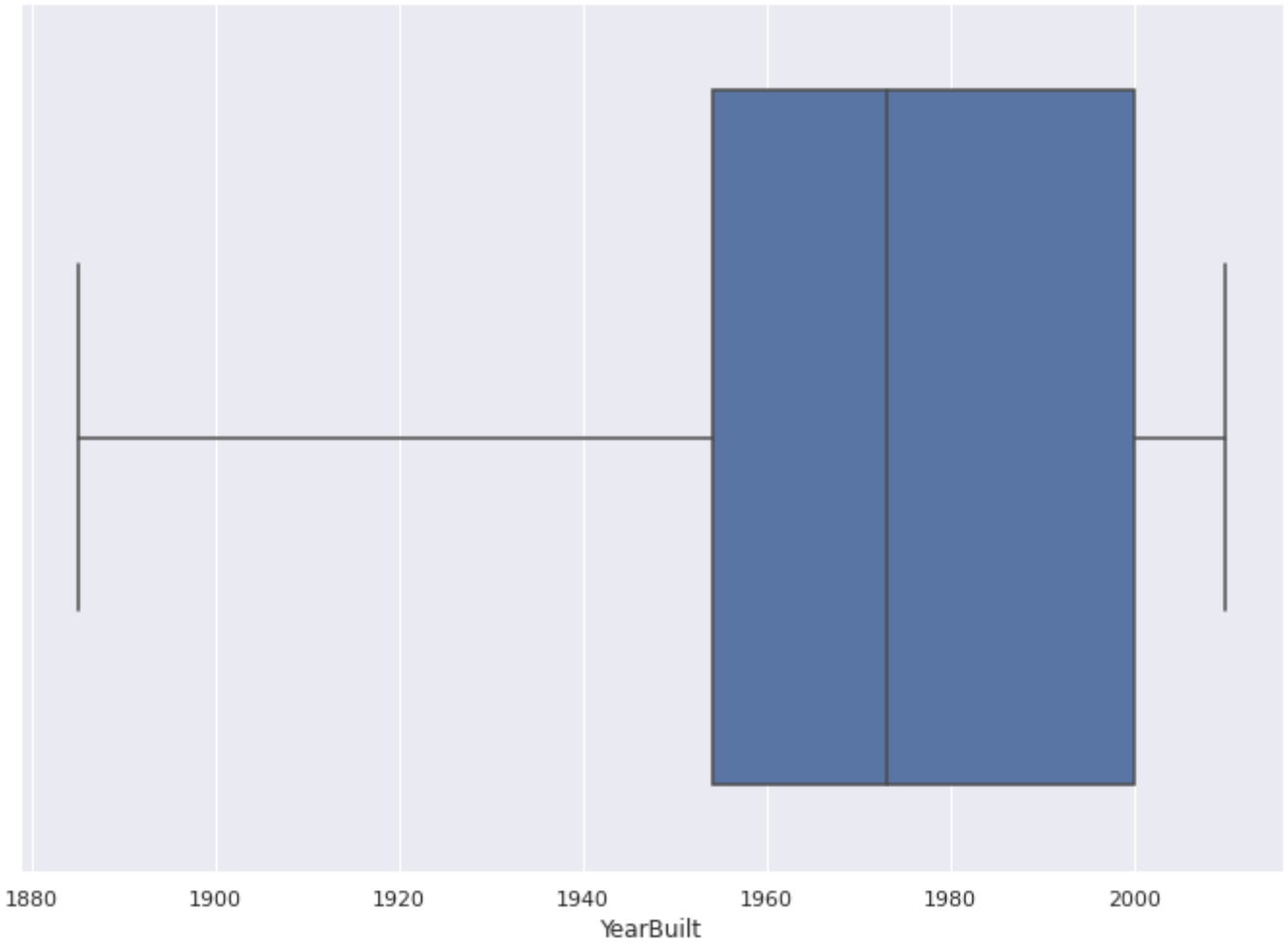

20
40
60
80
100
120
140
MSSubClass

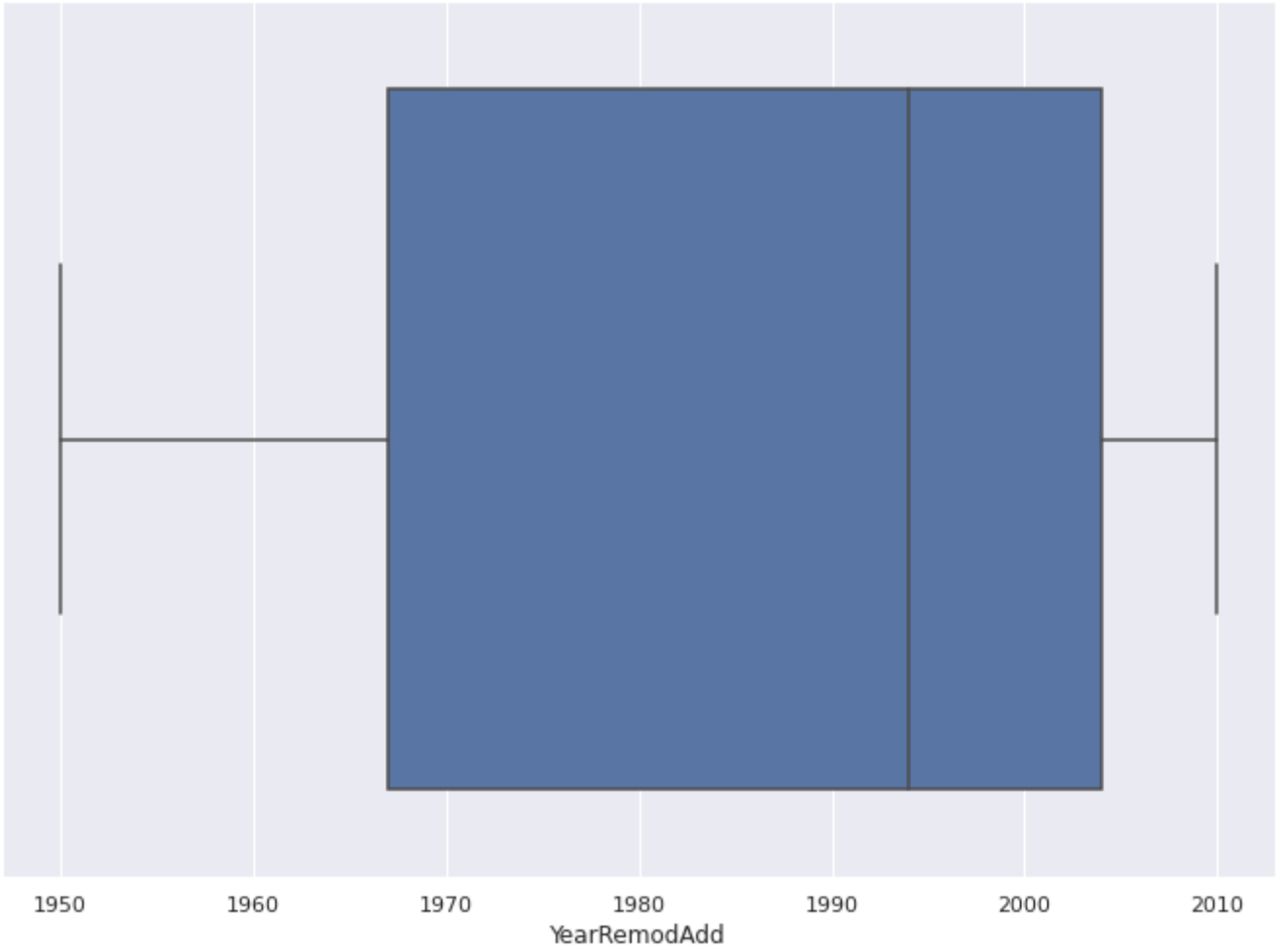

50
100
150
200
250
300
LotFrontage

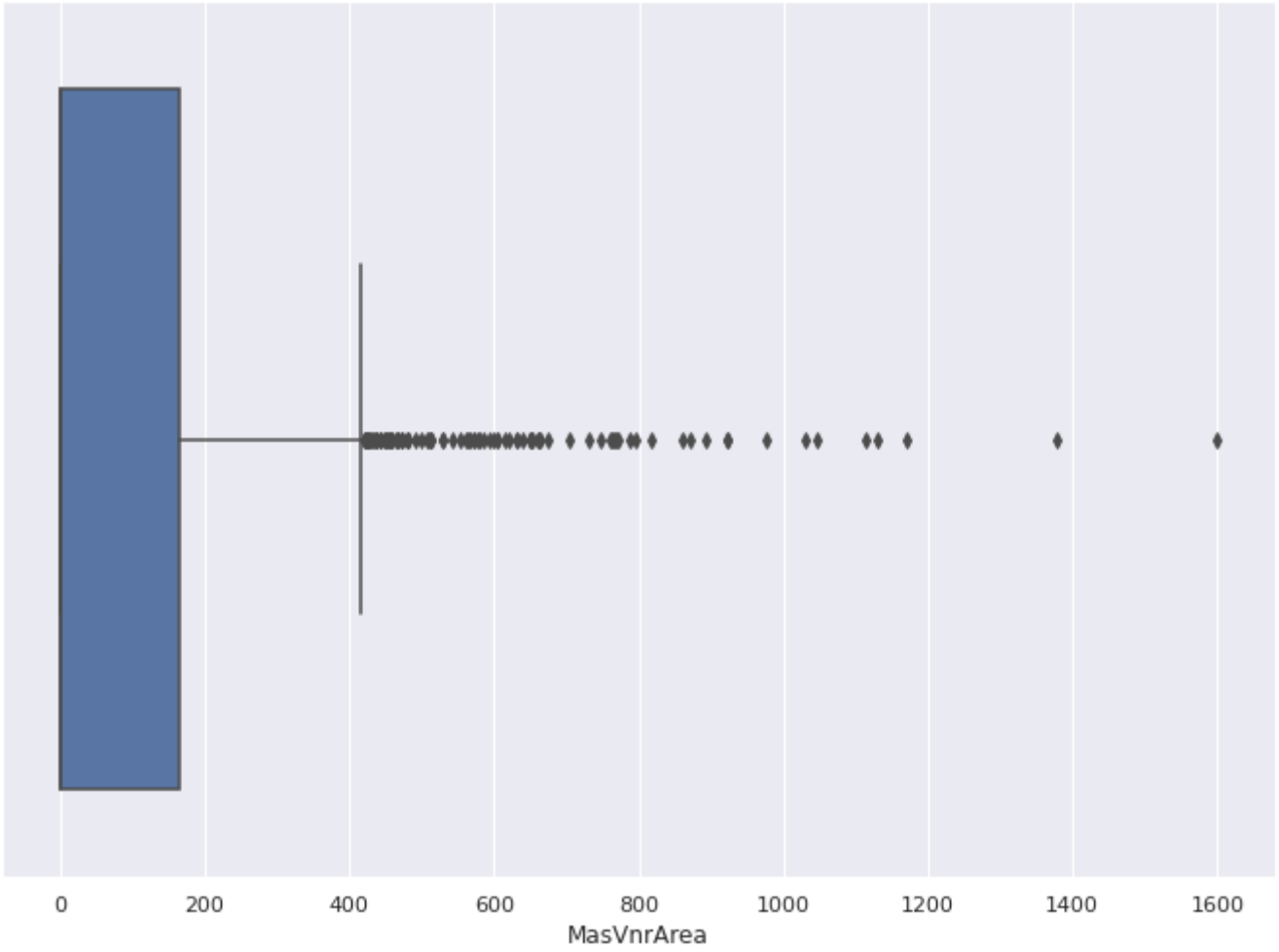

2000
4000
6000
8000
10000
12000
14000
16000
18000
LotArea

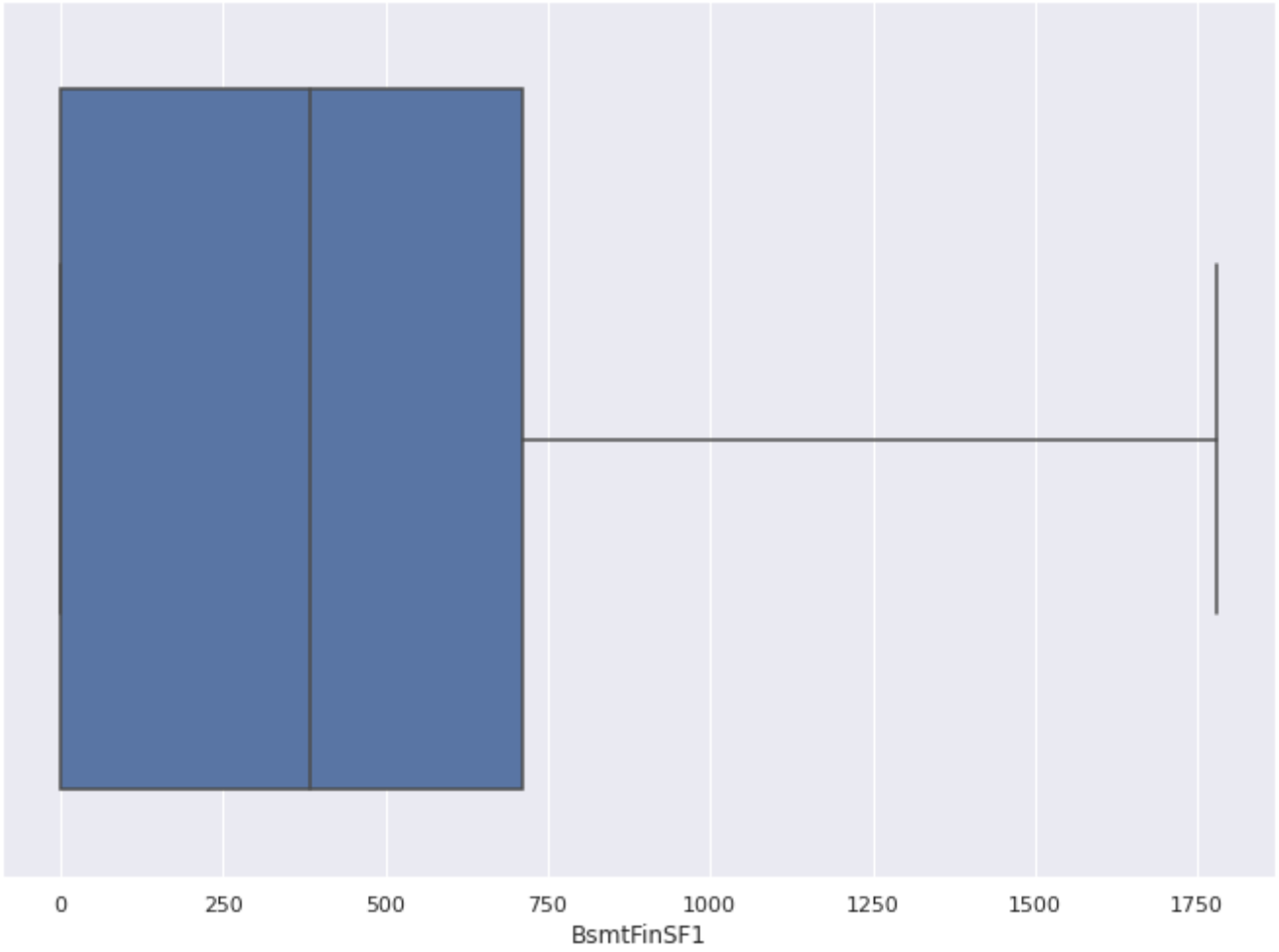

2
3
4
5
6
7
8
9
10
OverallQual

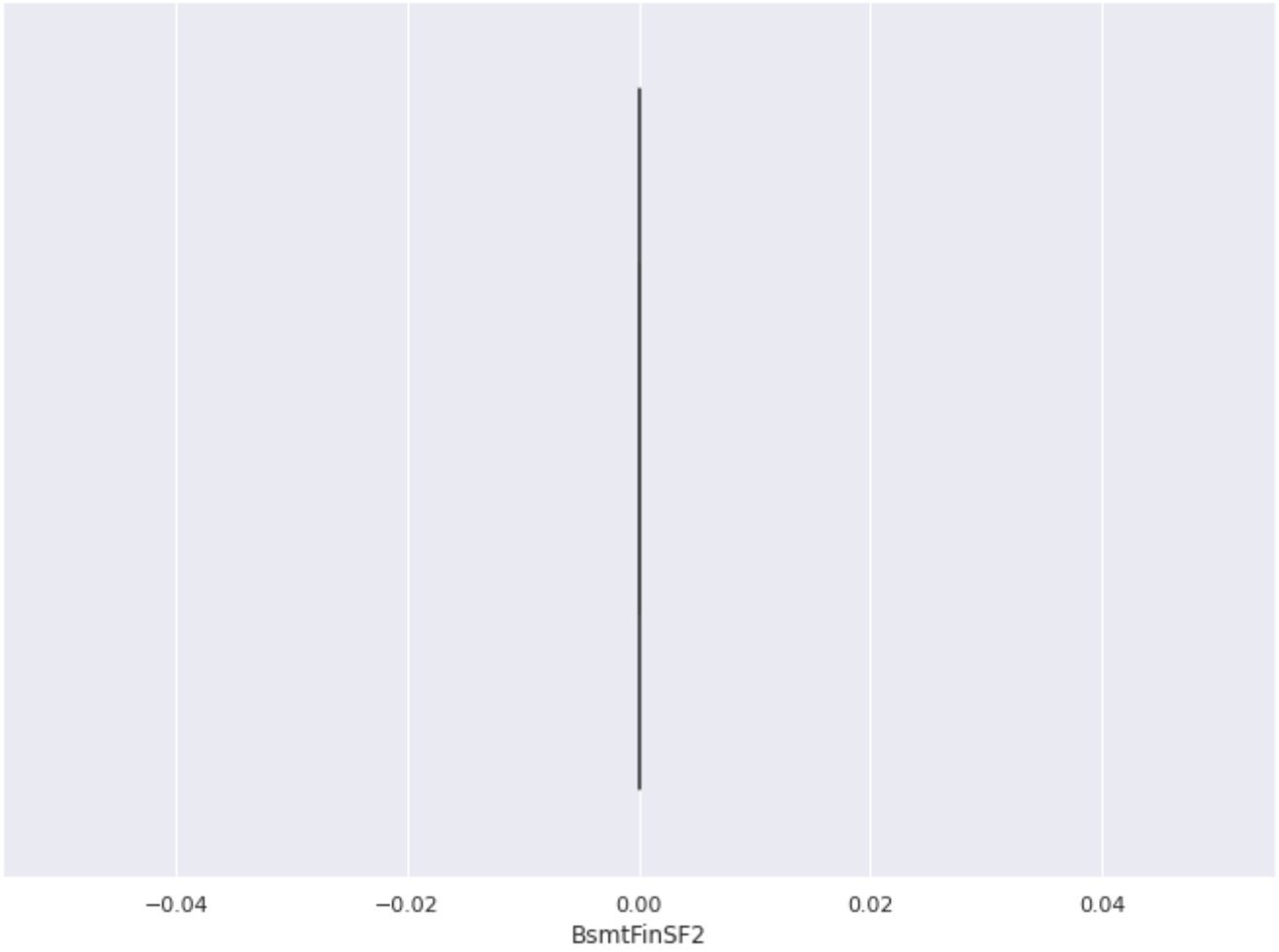

3.5
4.0
4.5
5.0
5.5
6.0
6.5
7.0
7.5
OverallCond

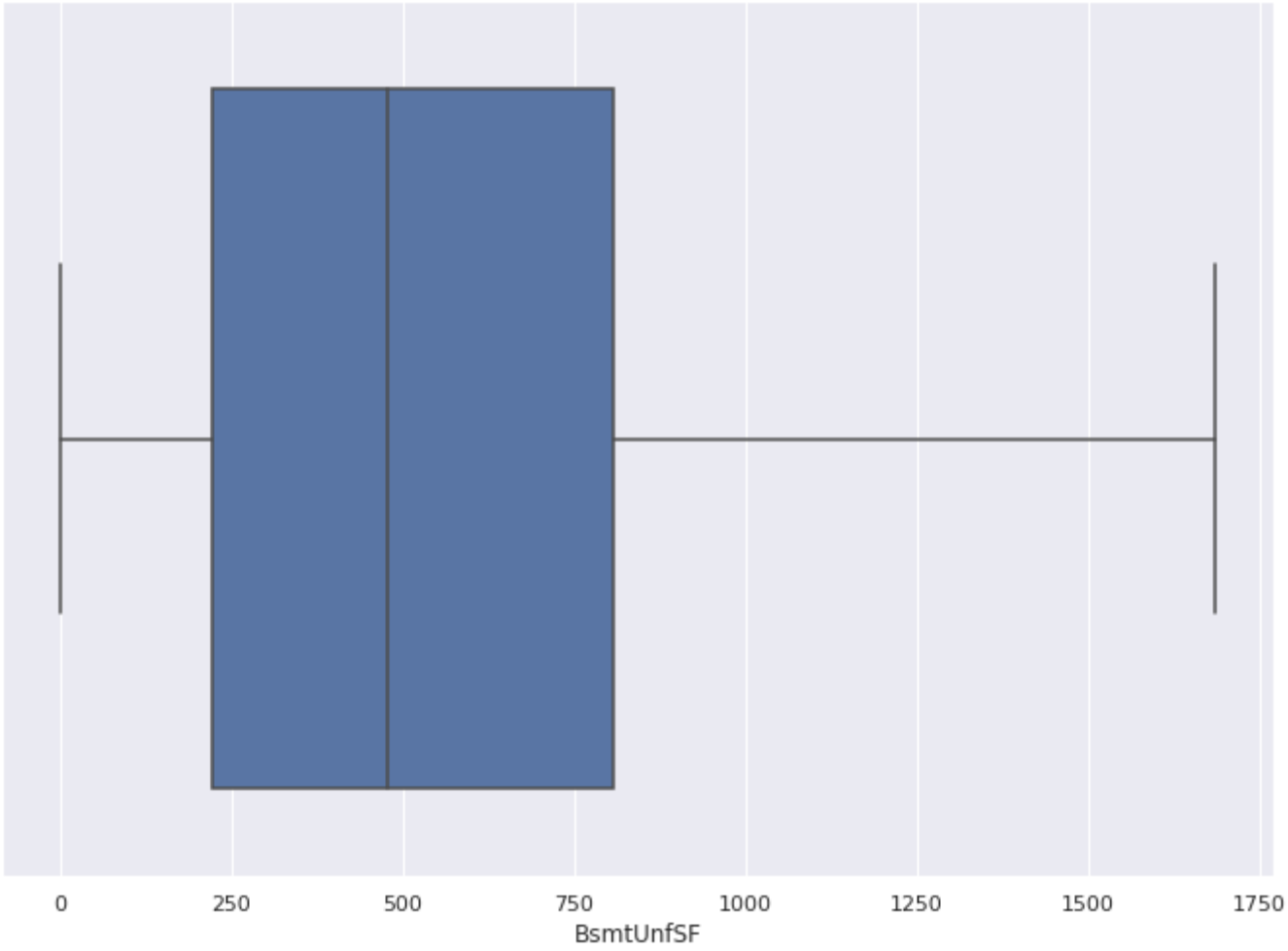

1880
1900
1920
1940
1960
1980
2000
YearBuilt

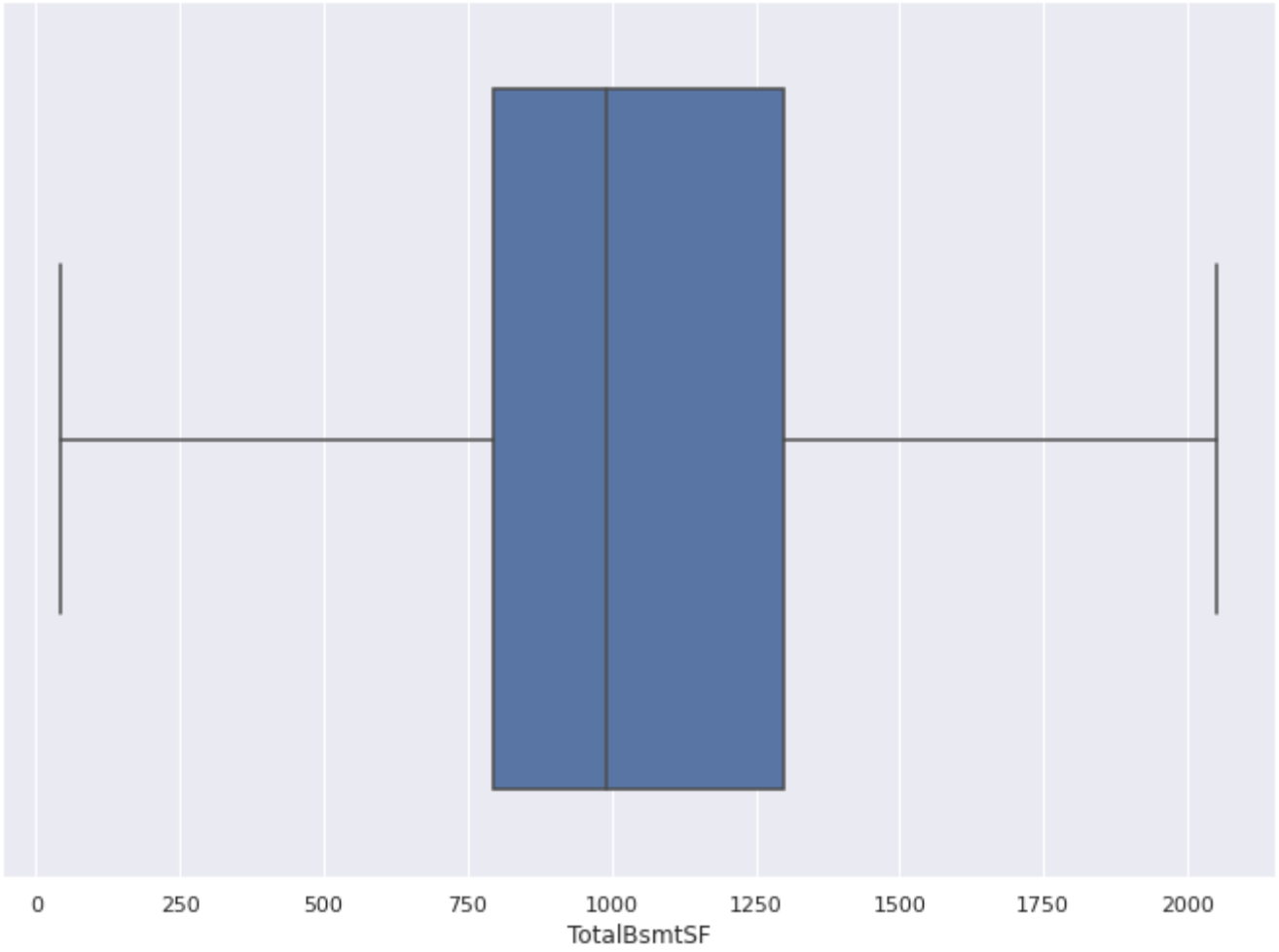

1950
1960
1970
1980
1990
2000
2010
YearRemodAdd

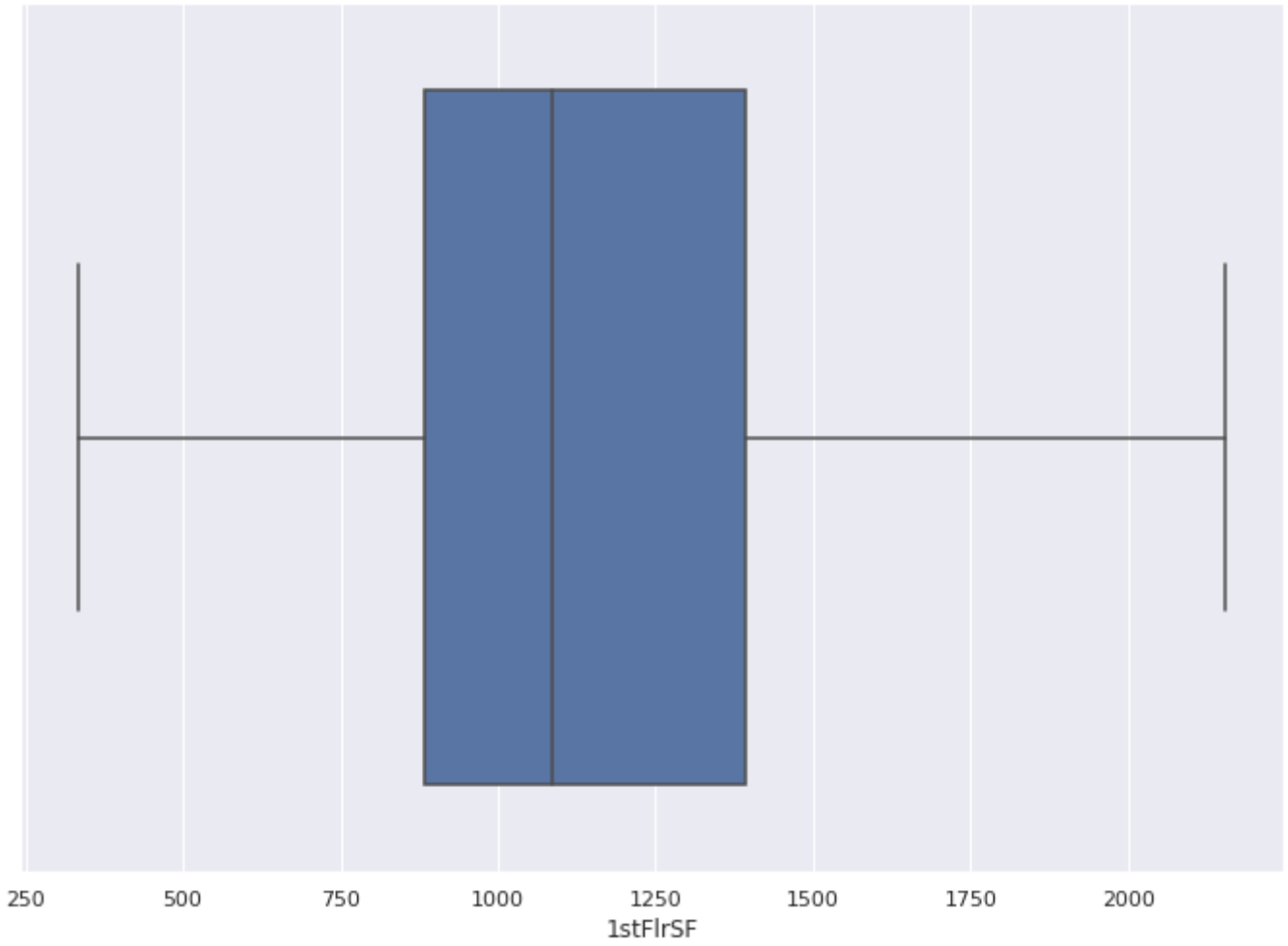

0
200
400
600
800
1000
1200
1400
1600
MasVnrArea

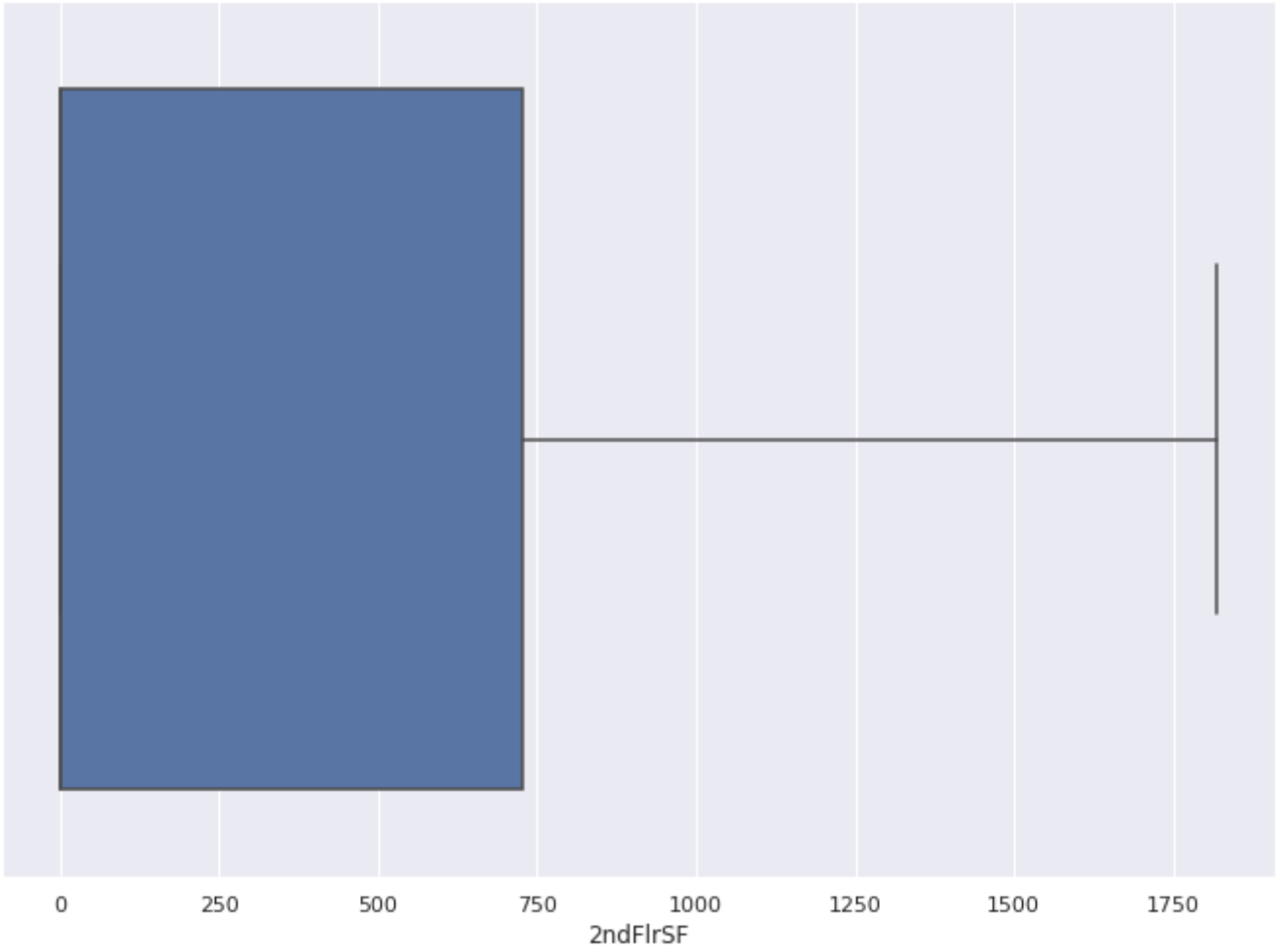

0
250
500
750
1000
1250
1500
1750
BsmtFinSF1

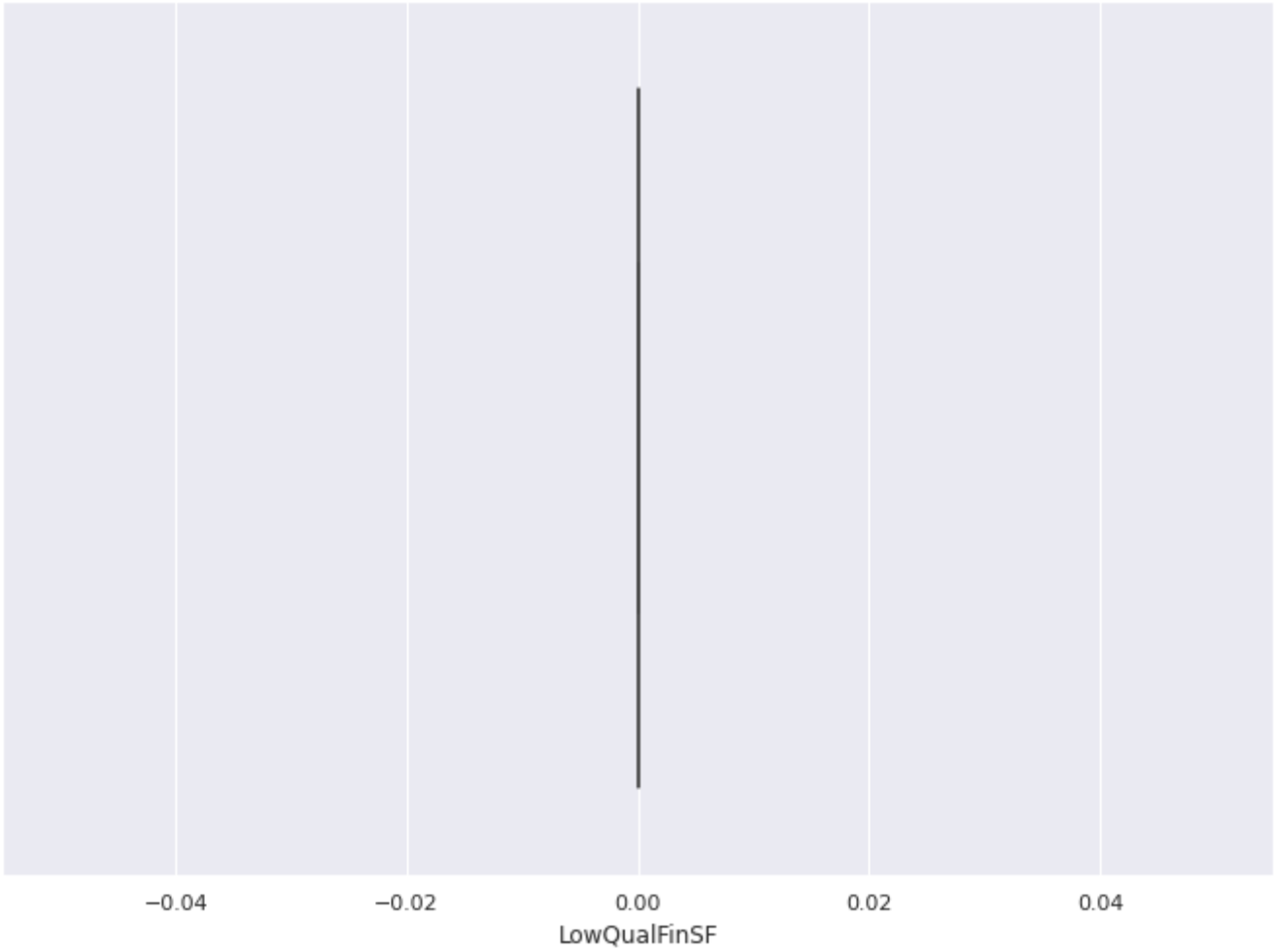

−0.04
−0.02
0.00
0.02
0.04
BsmtFinSF2

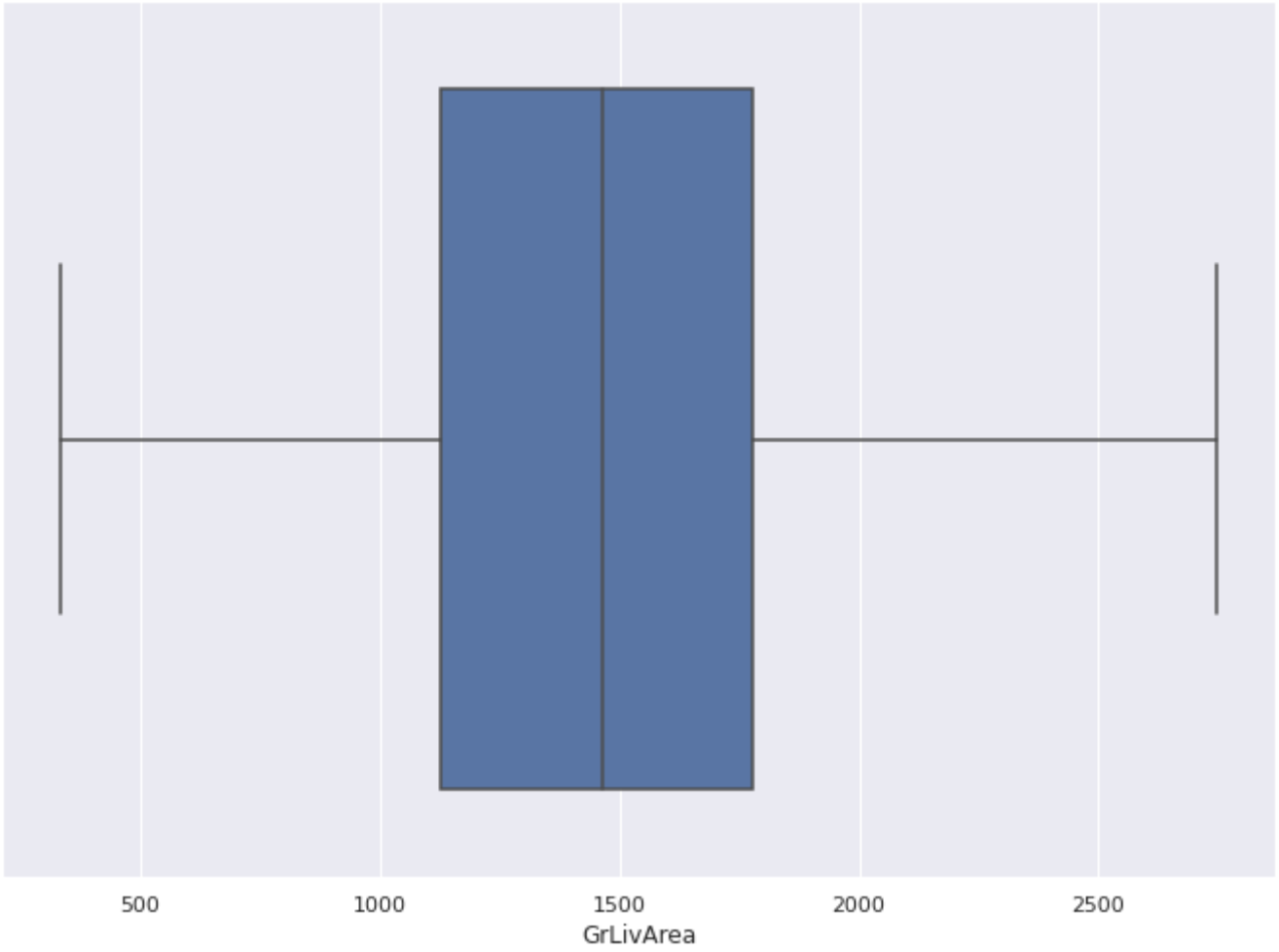

0
250
500
750
1000
1250
1500
1750
BsmtUnfSF

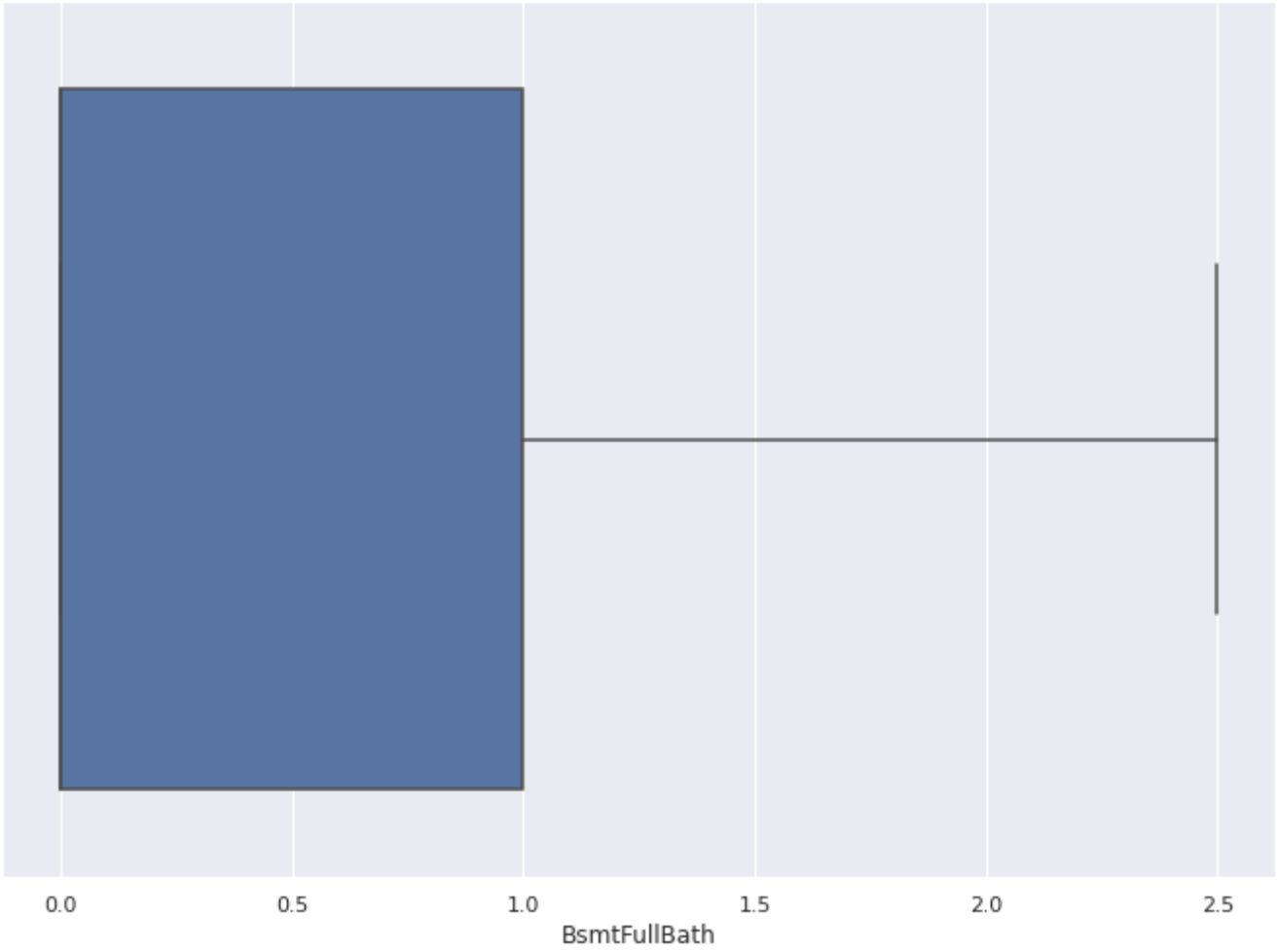

0
250
500
750
1000
1250
1500
1750
2000
TotalBsmtSF

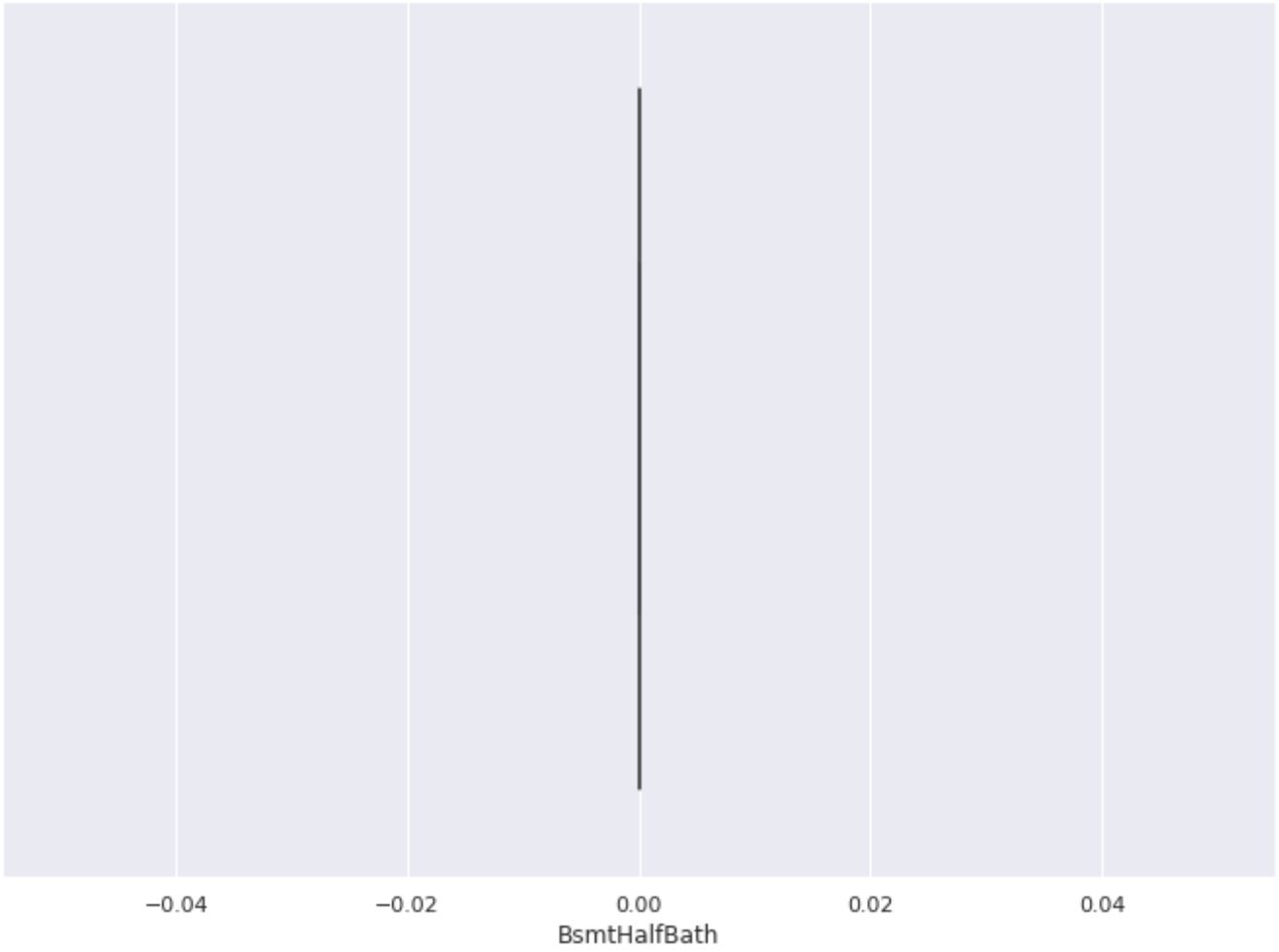

250
500
750
1000
1250
1500
1750
2000
1stFlrSF

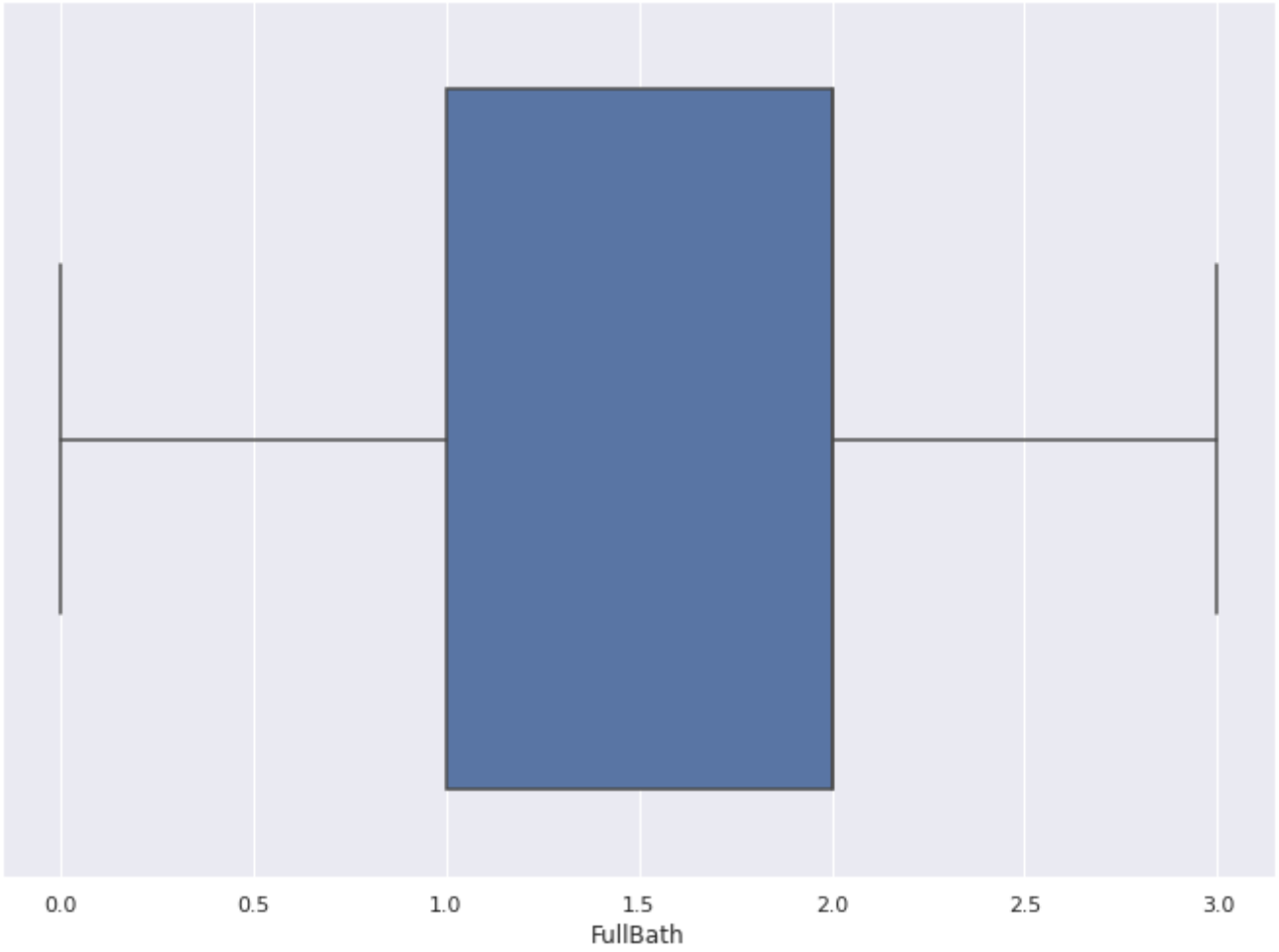

0
250
500
750
1000
1250
1500
1750
2ndFlrSF

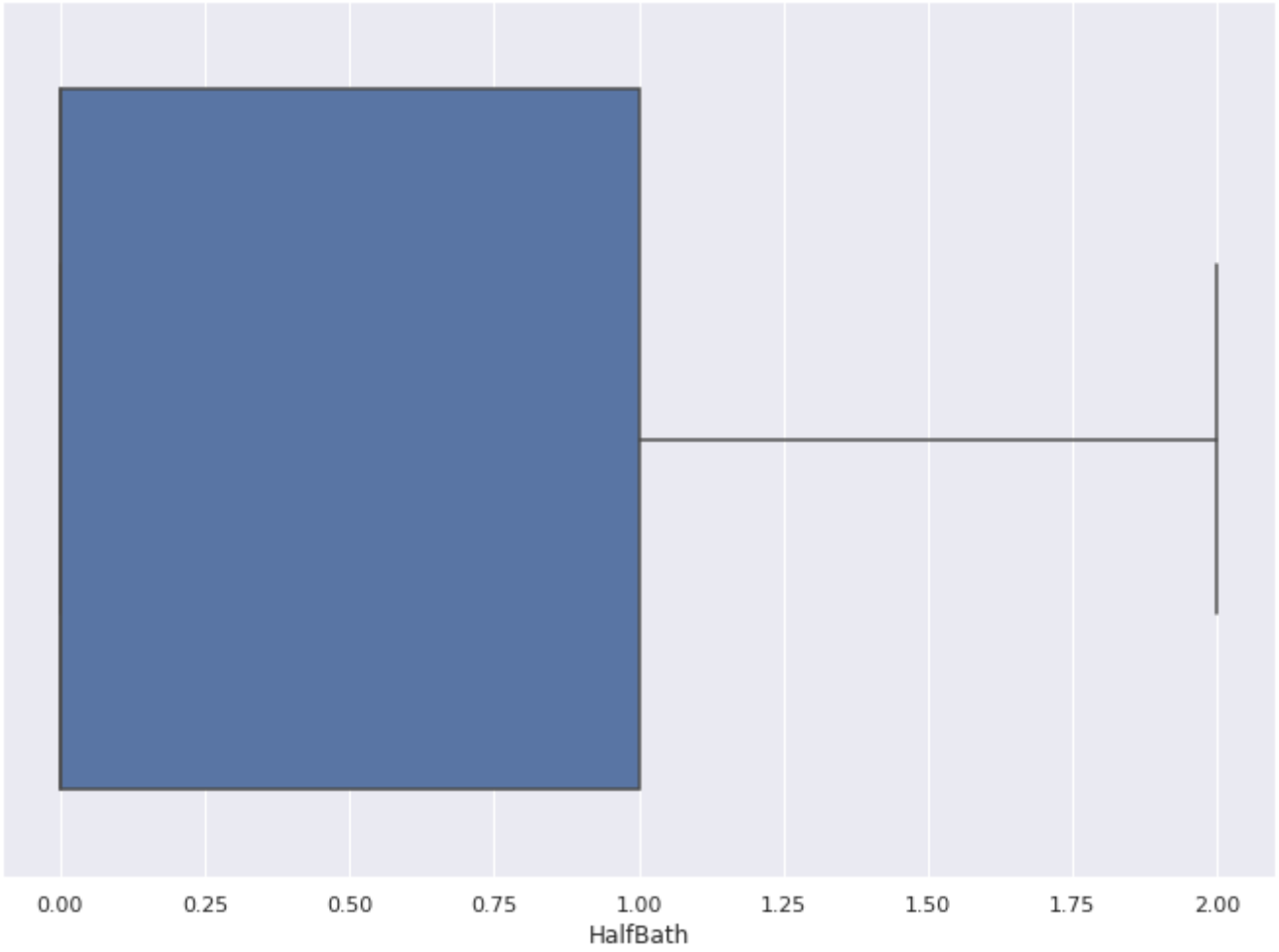

−0.04
−0.02
0.00
0.02
0.04
LowQualFinSF

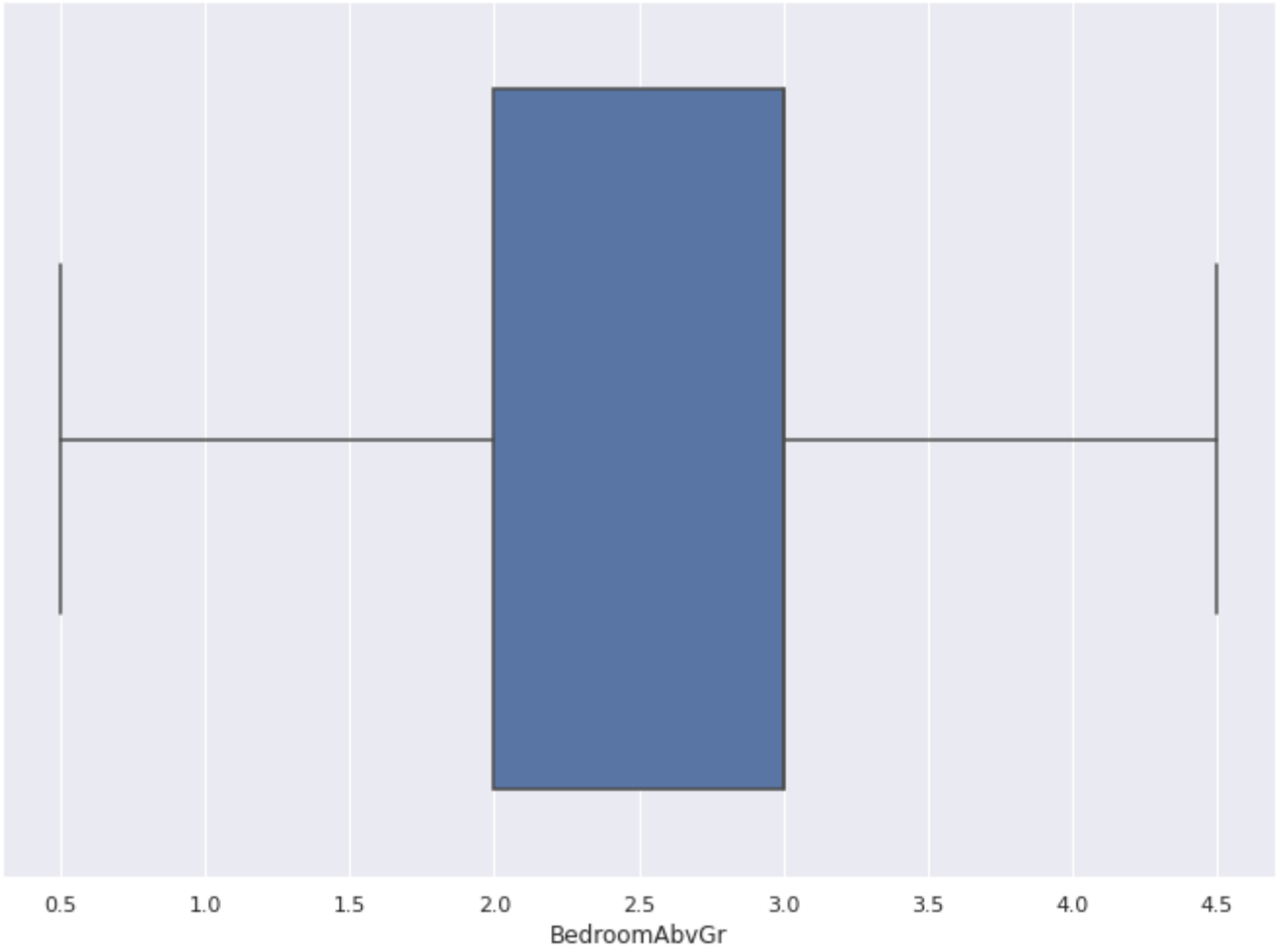

500
1000
1500
2000
2500
GrLivArea

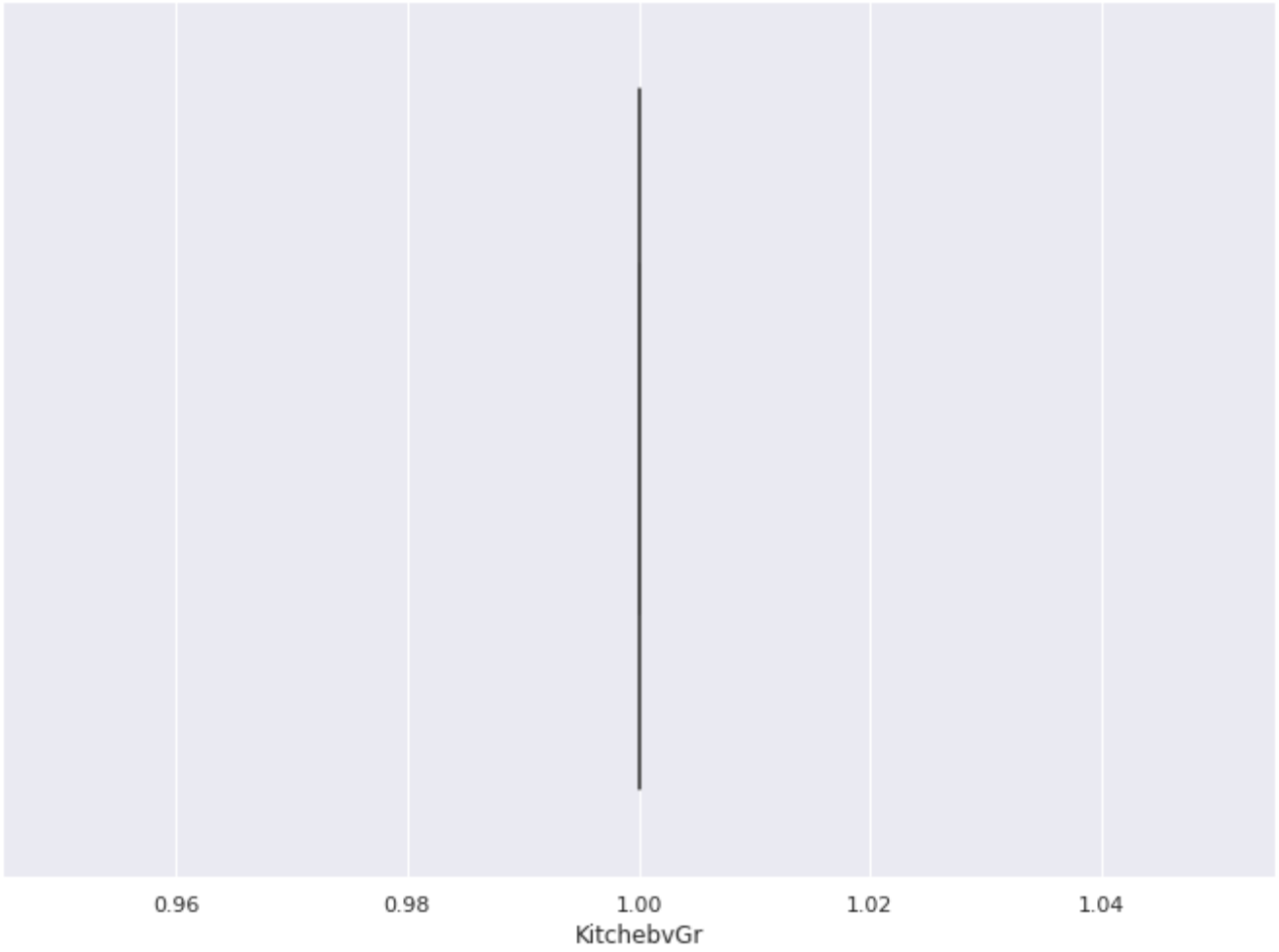

0.0
0.5
1.0
1.5
2.0
2.5
BsmtFullBath

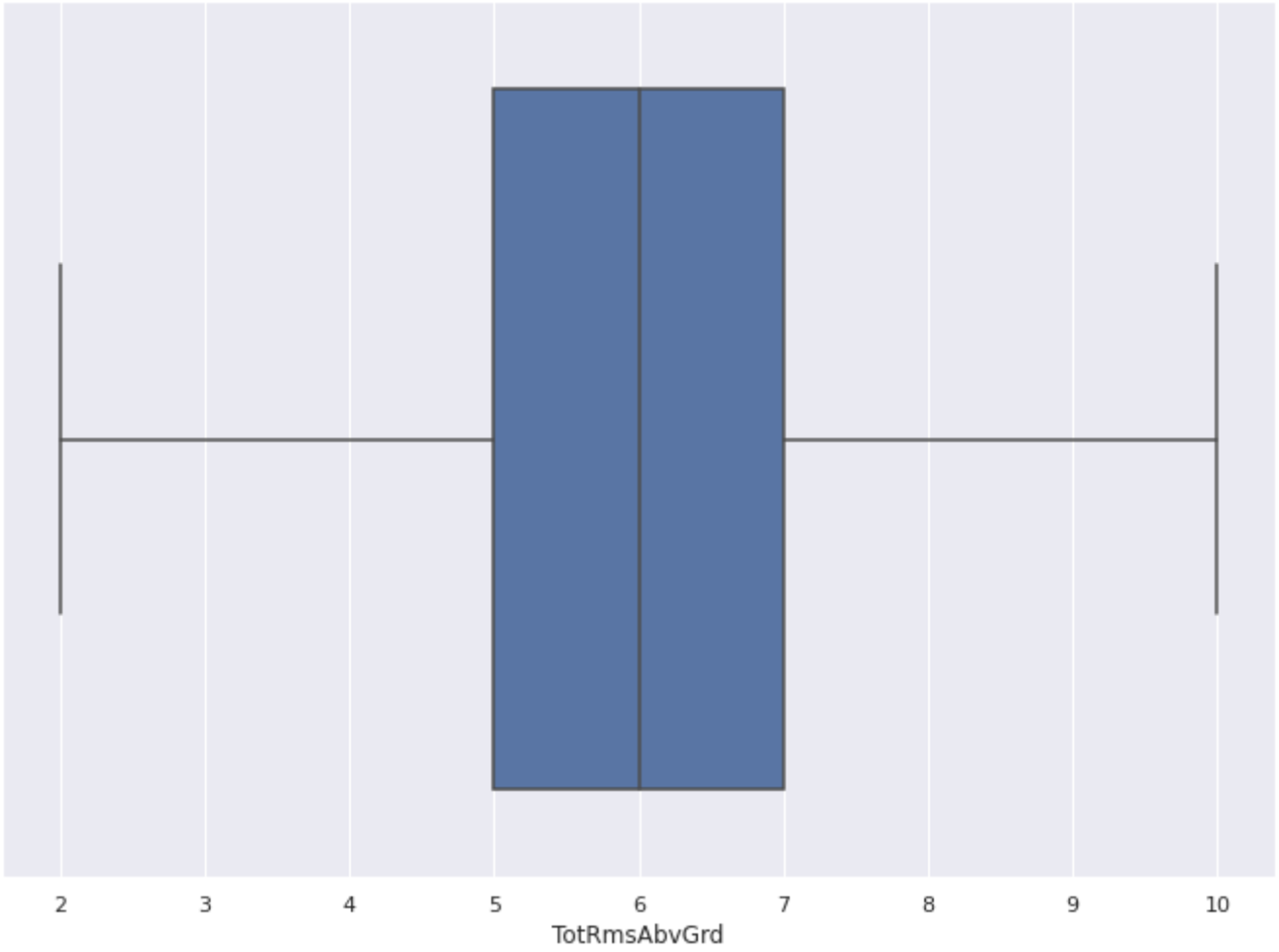

−0.04
−0.02
0.00
0.02
0.04
BsmtHalfBath

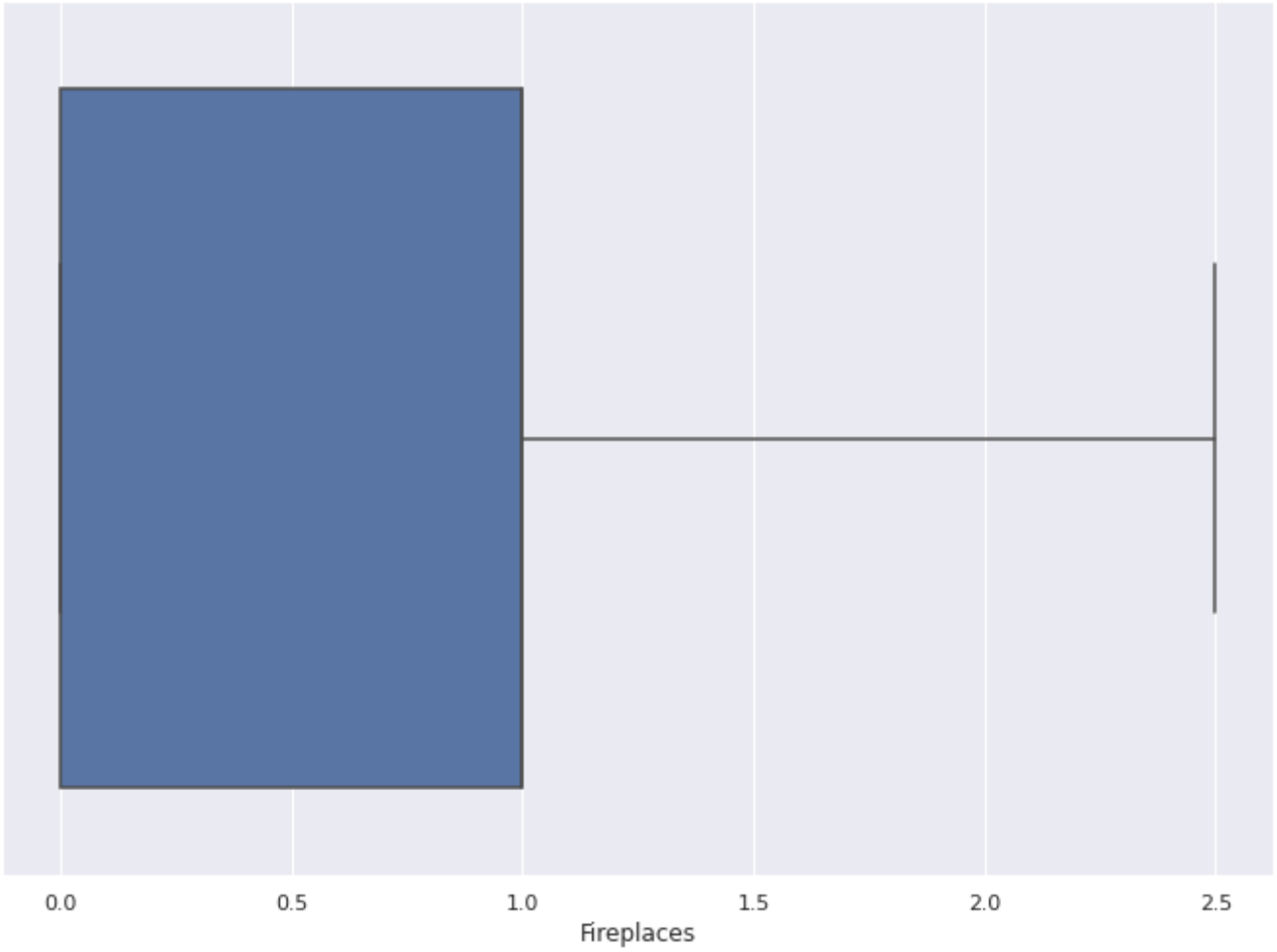

0.0
0.5
1.0
1.5
2.0
2.5
3.0
FullBath

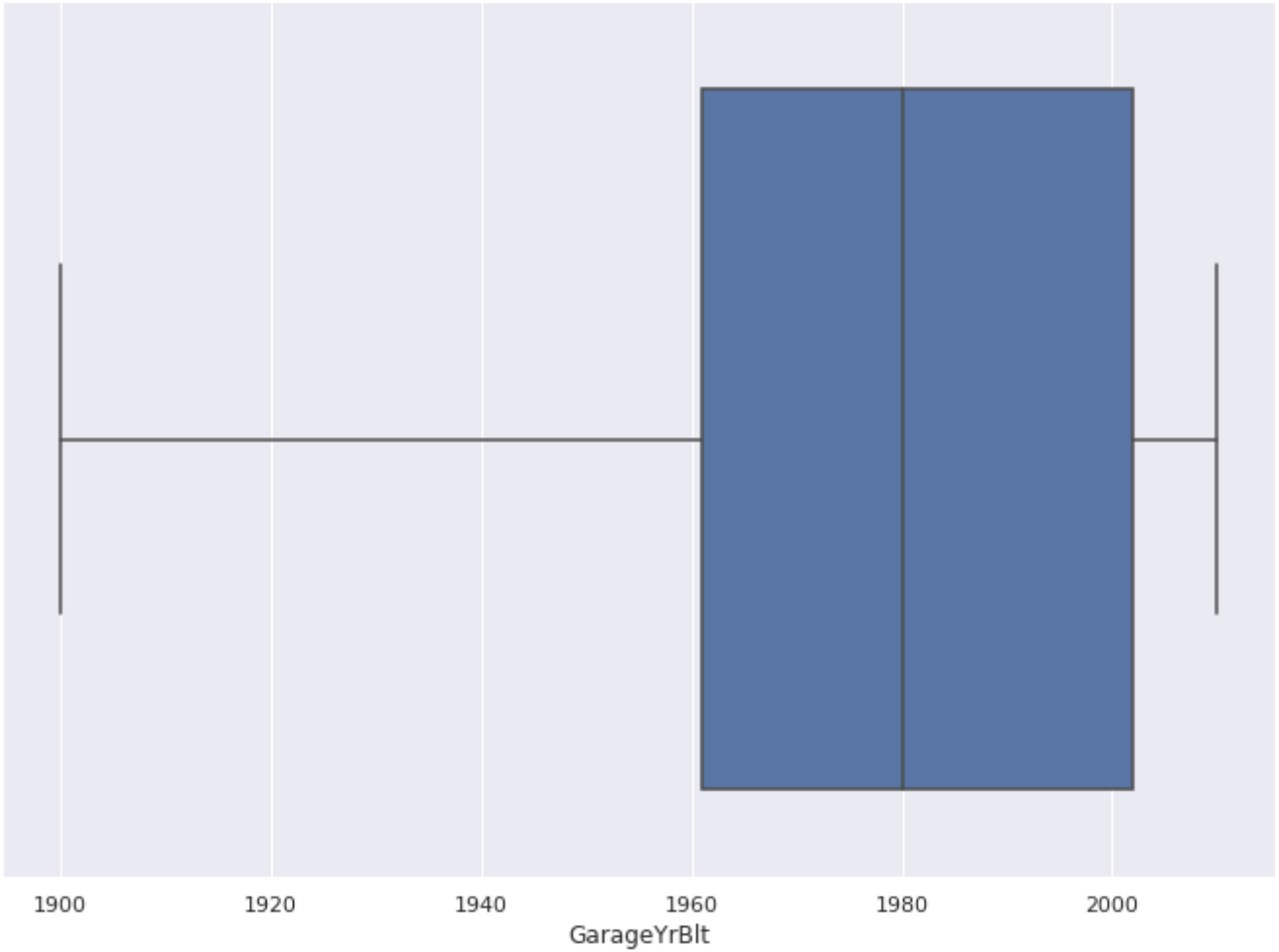

0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00
HalfBath

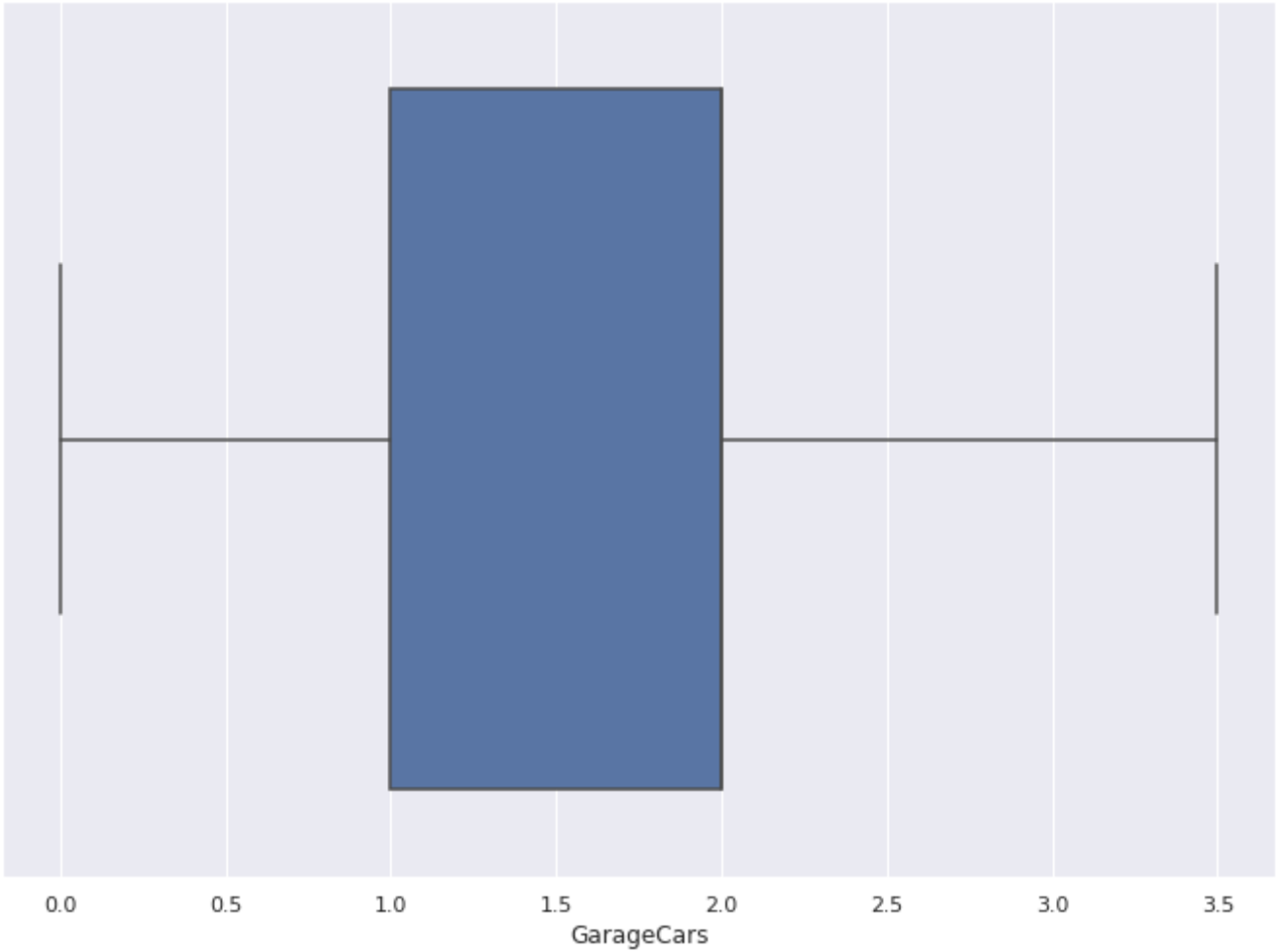

0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
BedroomAbvGr

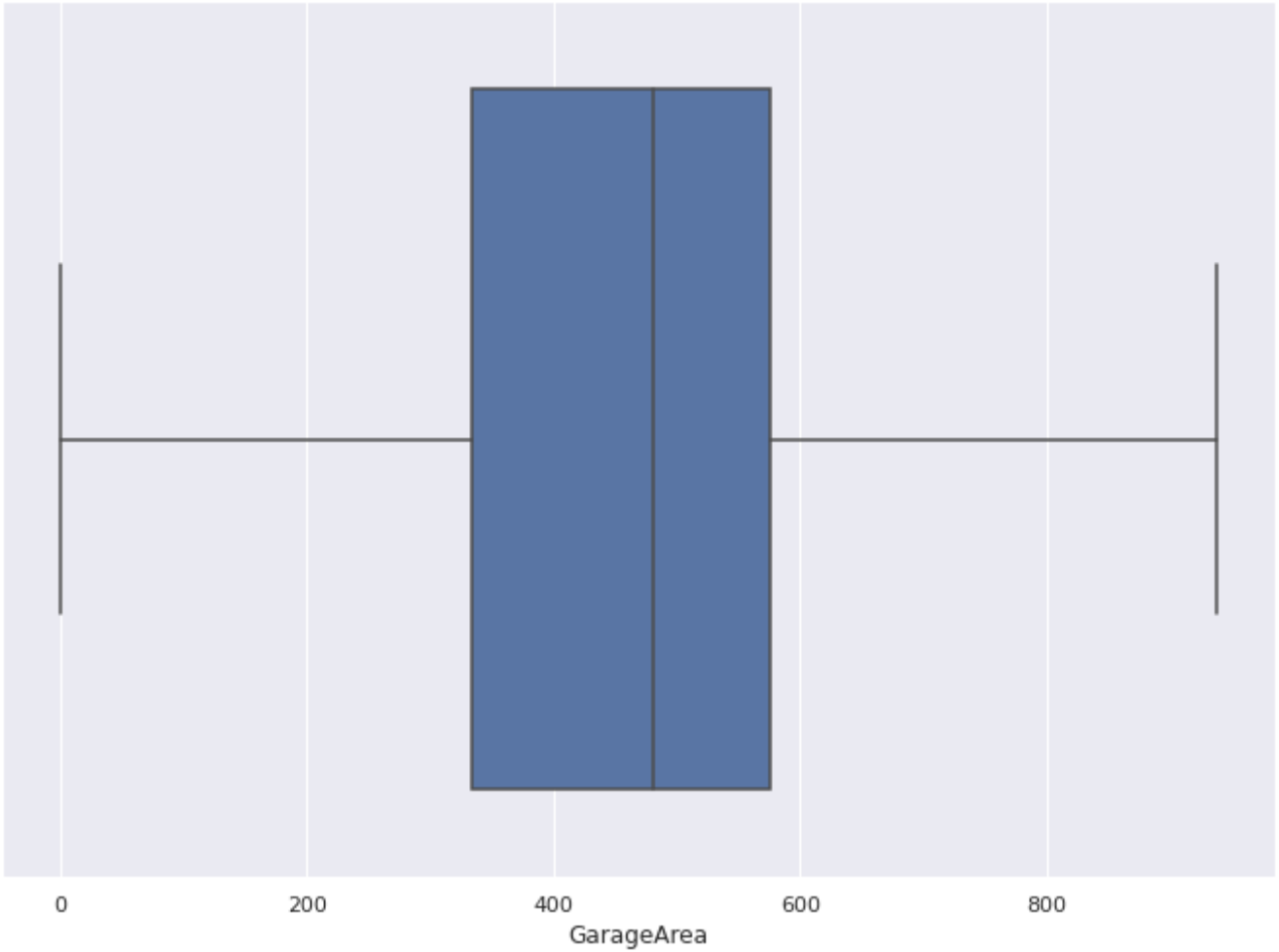

0.96
0.98
1.00
1.02
1.04
KitchebvGr

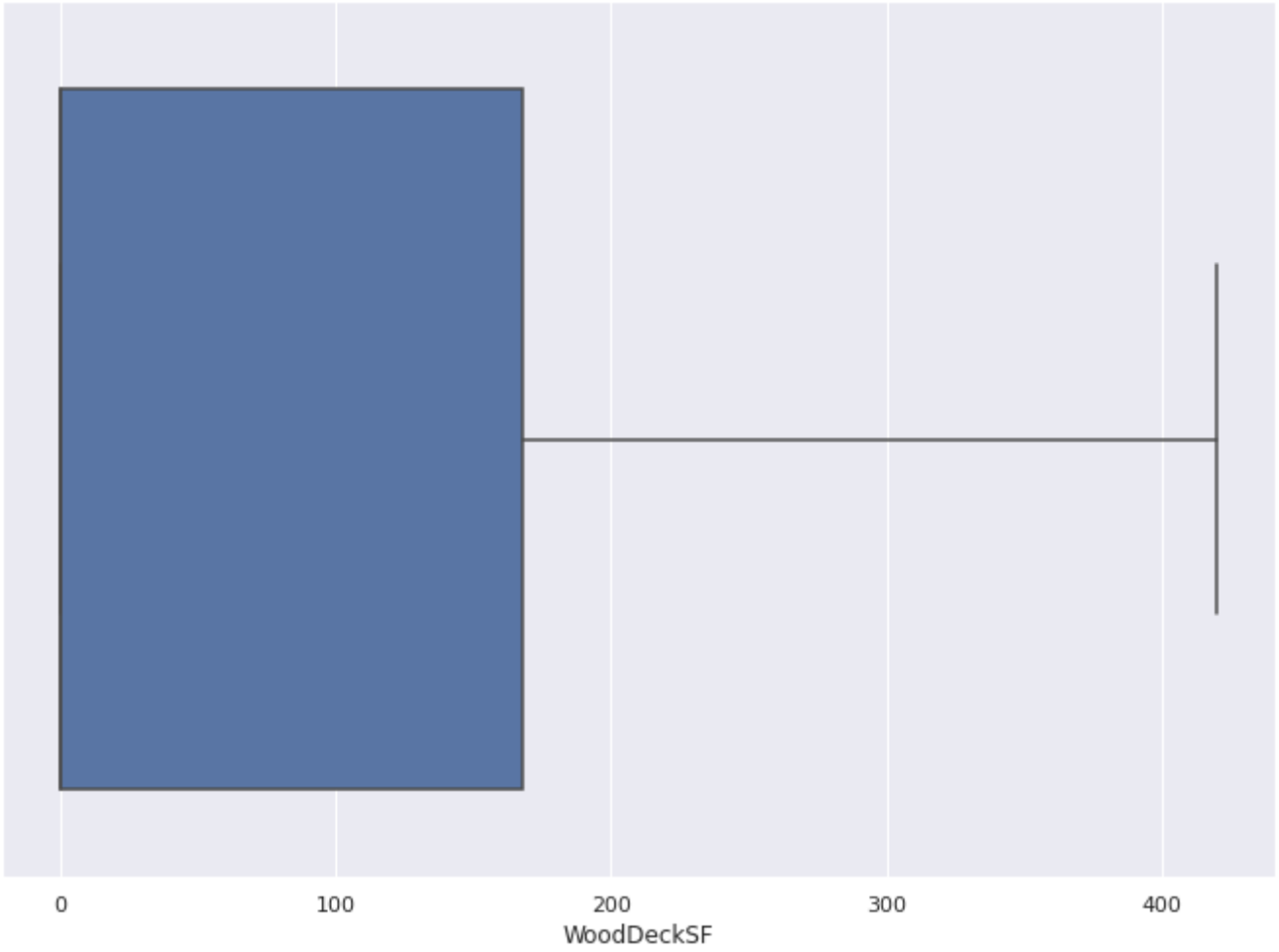

2
3
4
5
6
7
8
9
10
TotRmsAbvGrd

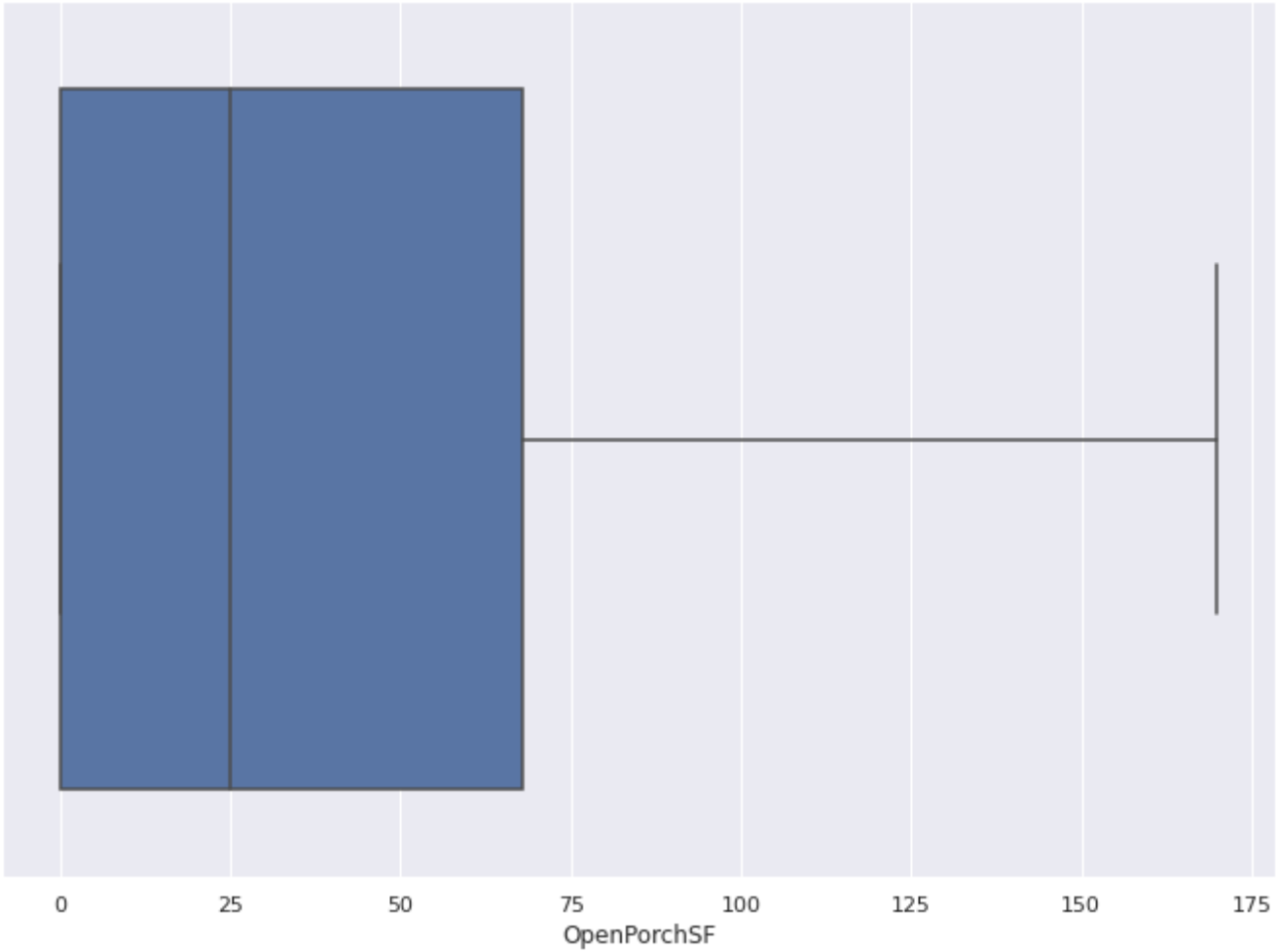

0.0
0.5
1.0
1.5
2.0
2.5
Fireplaces

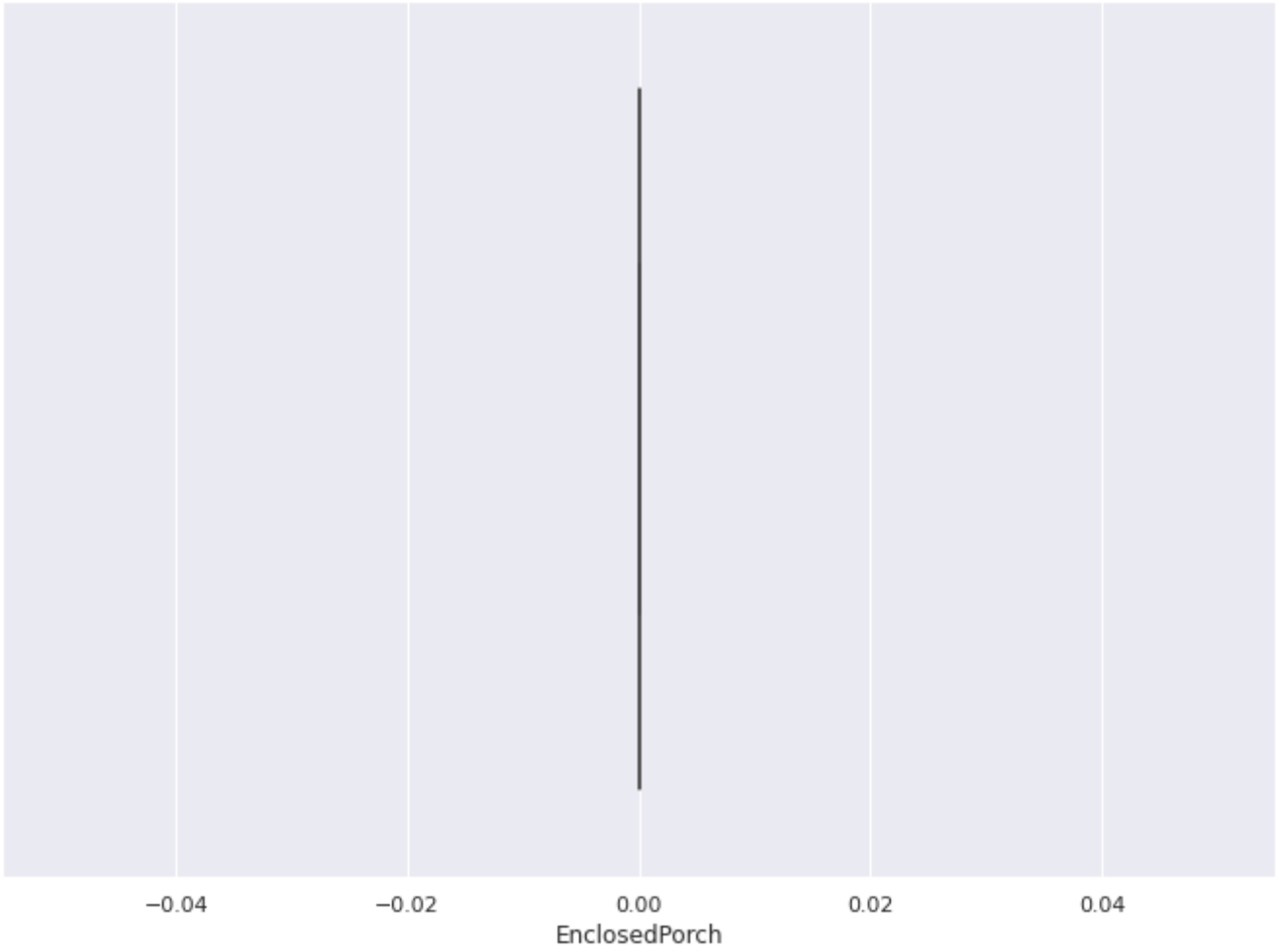

1900
1920
1940
1960
1980
2000
GarageYrBlt

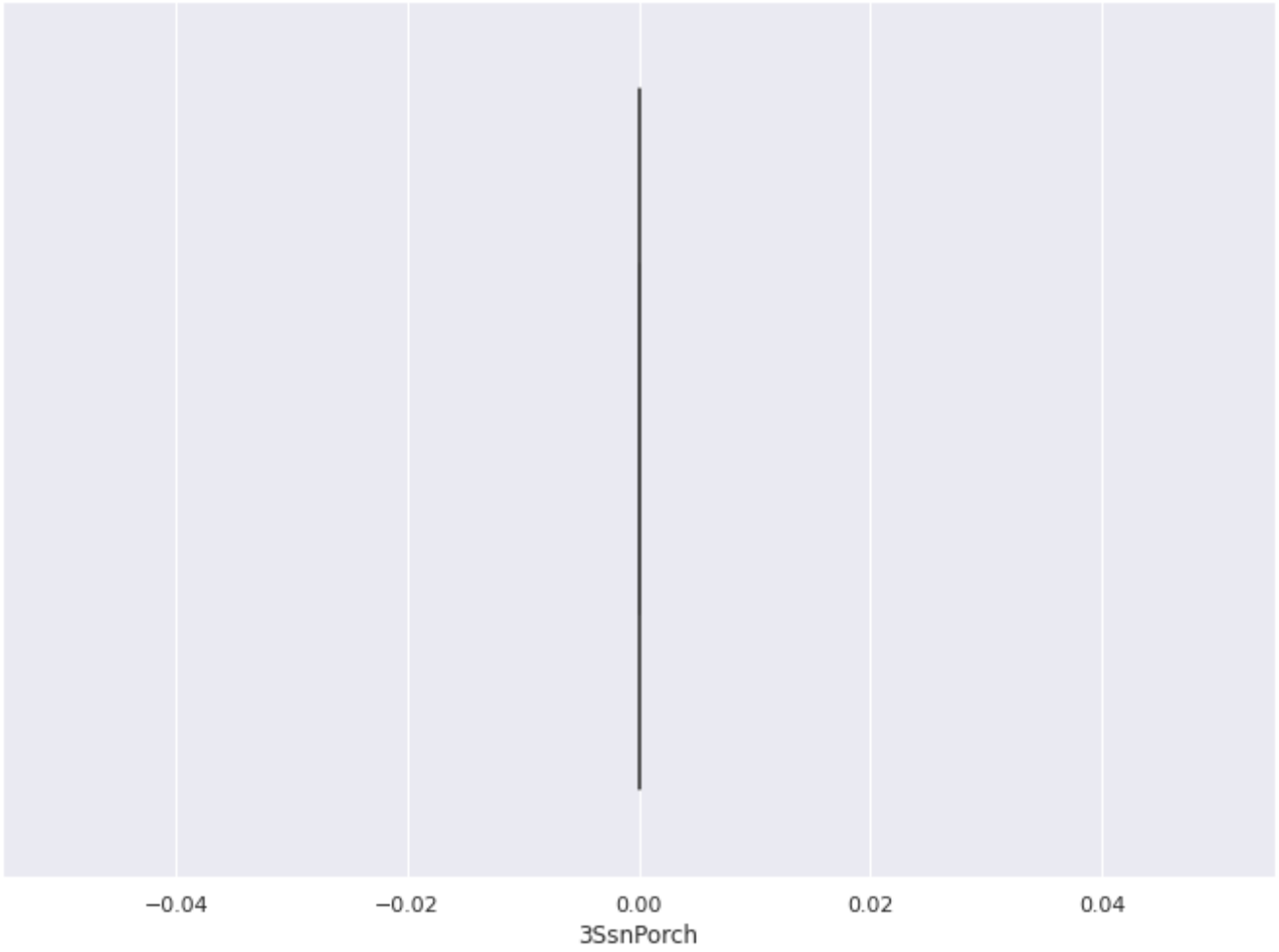

0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
GarageCars

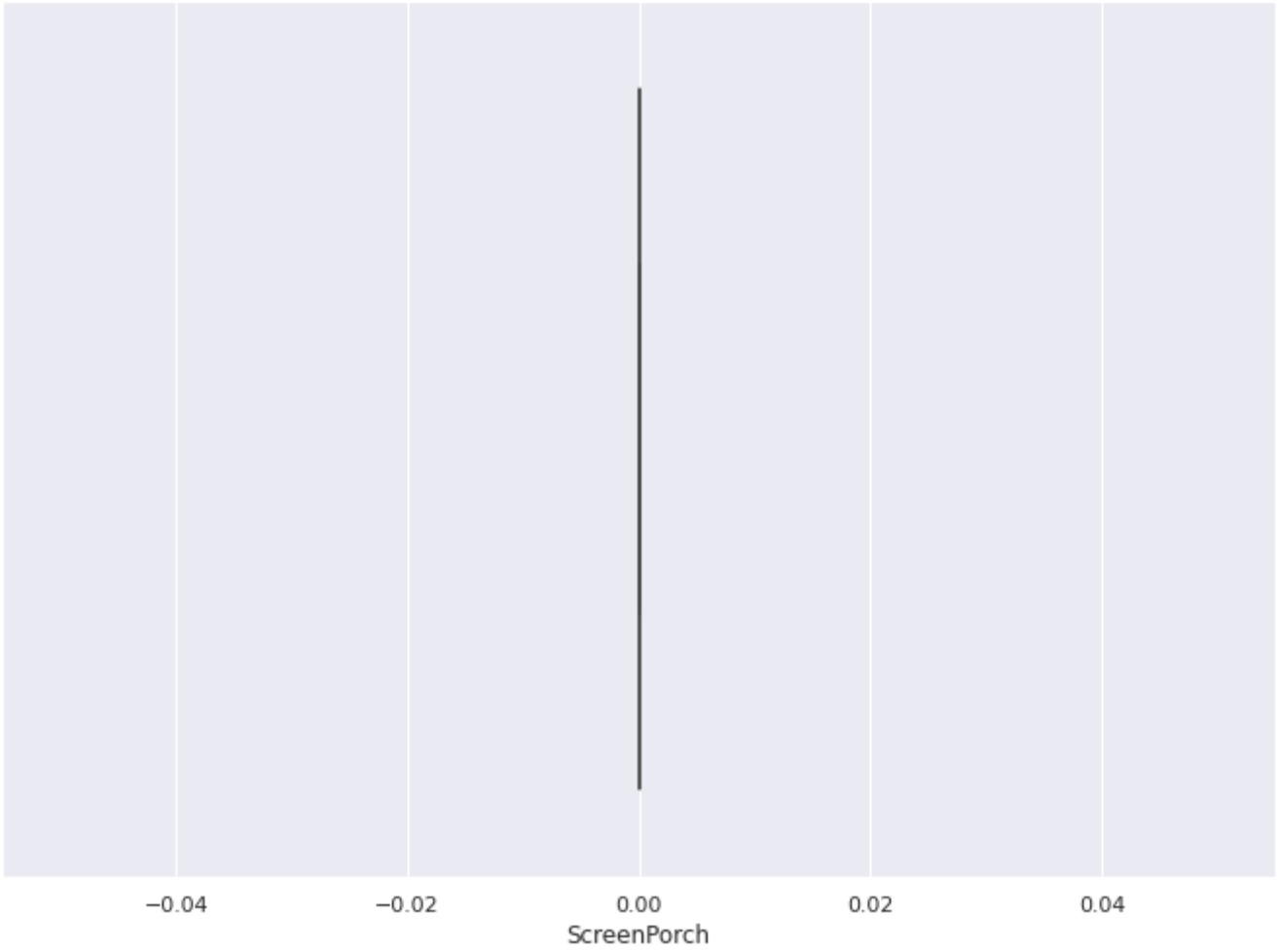

0
200
400
600
800
GarageArea

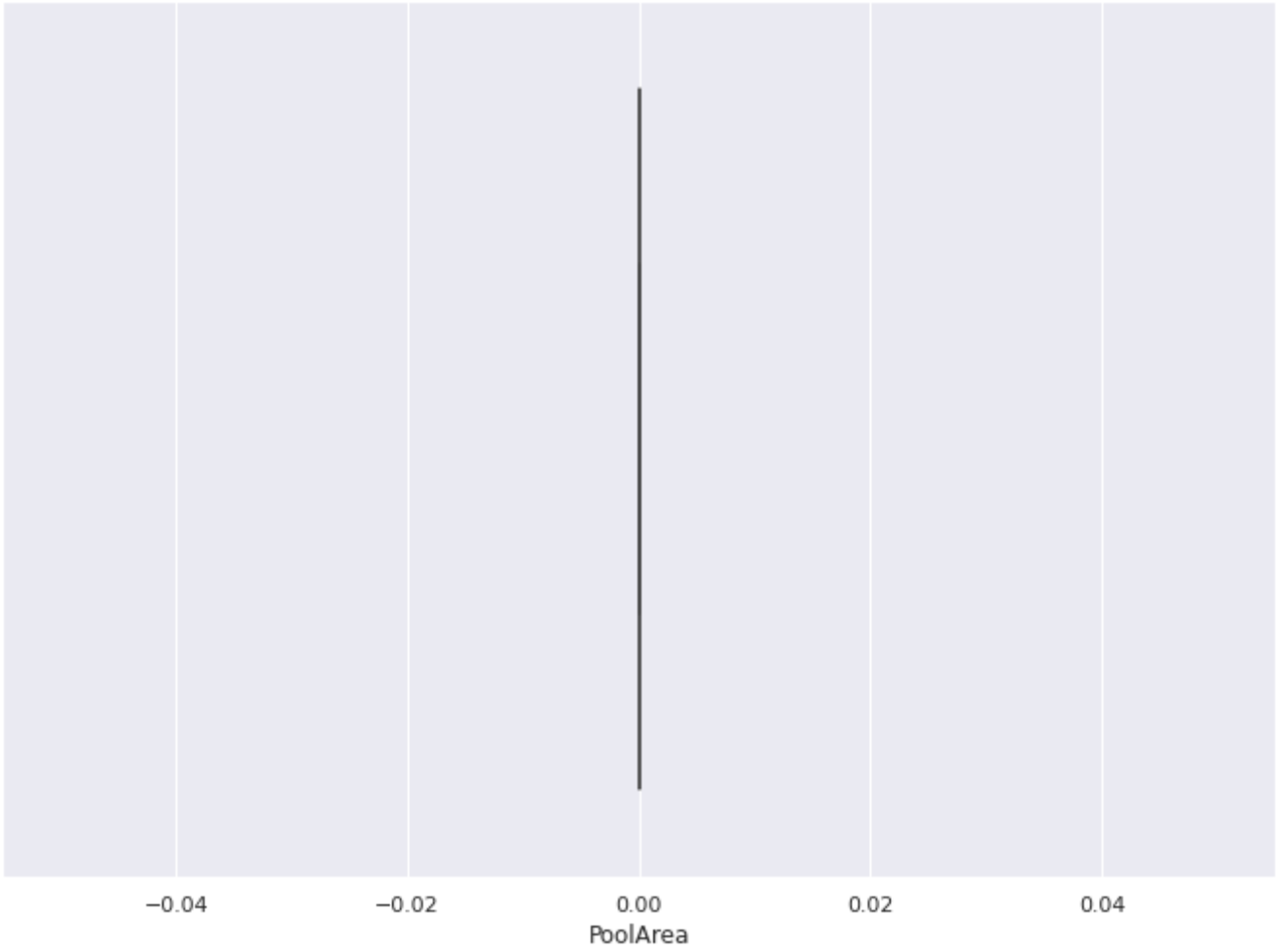

0
100
200
300
400
WoodDeckSF

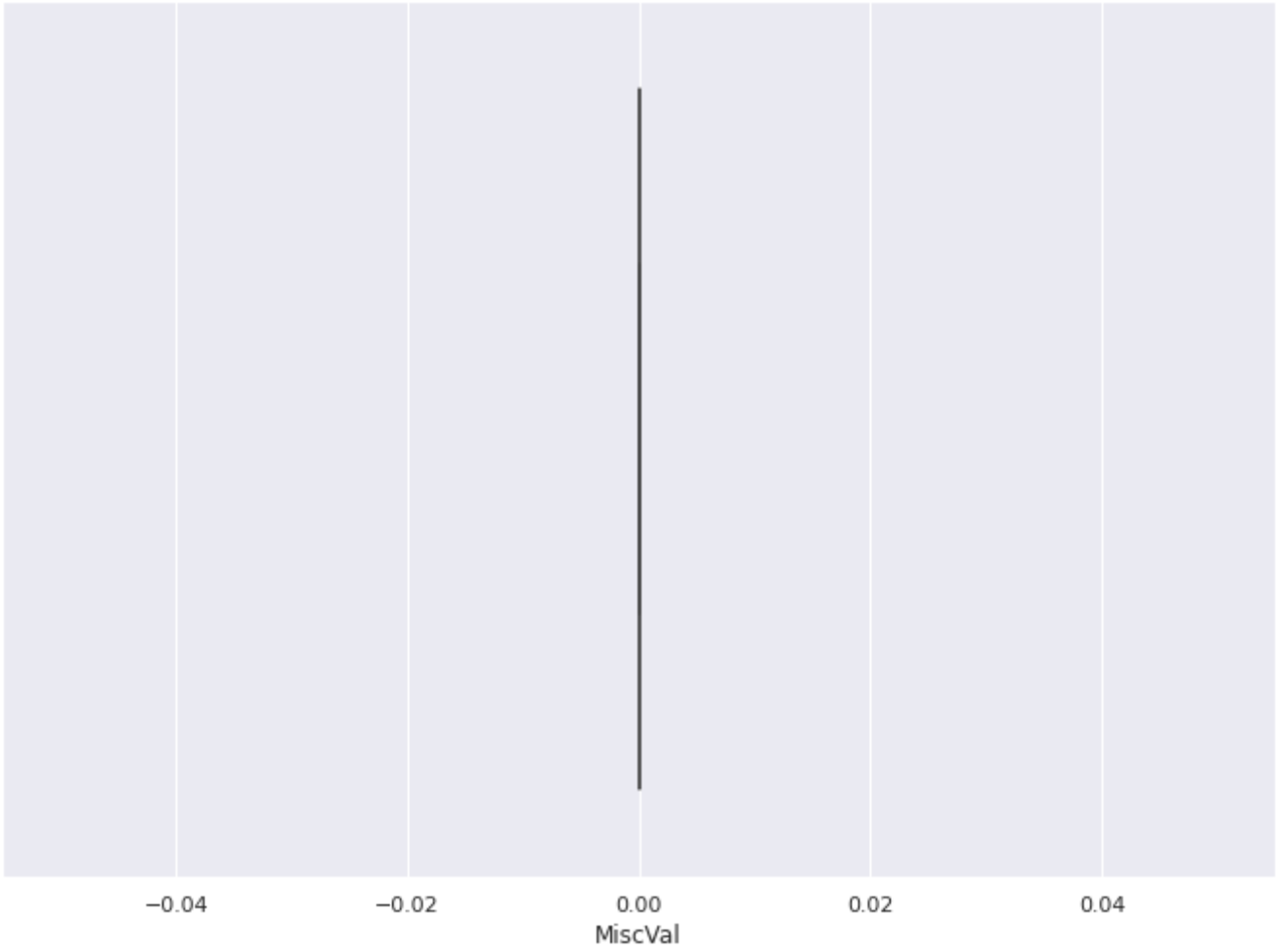

0
25
50
75
100
125
150
175
OpenPorchSF

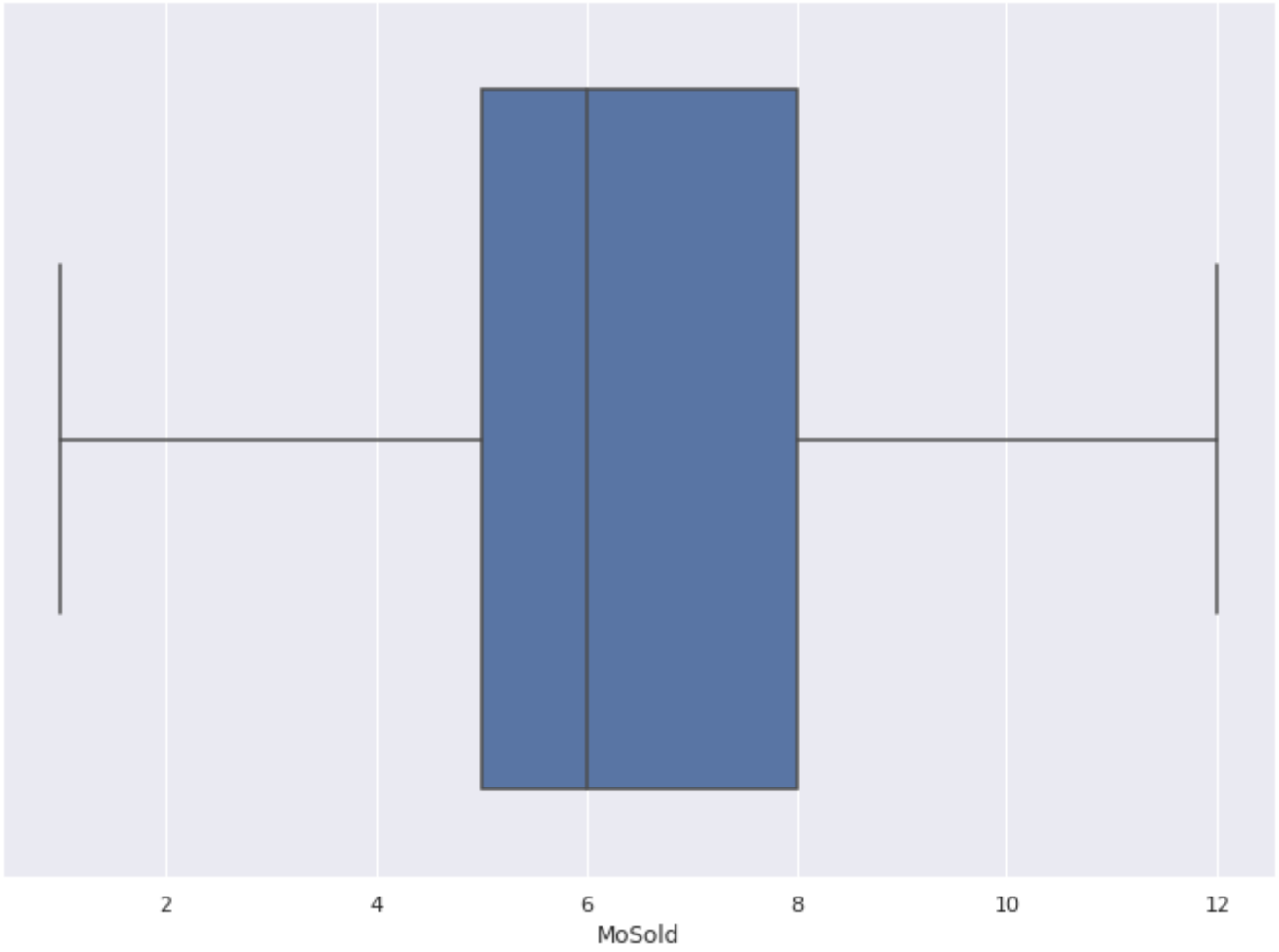

−0.04
−0.02
0.00
0.02
0.04
EnclosedPorch

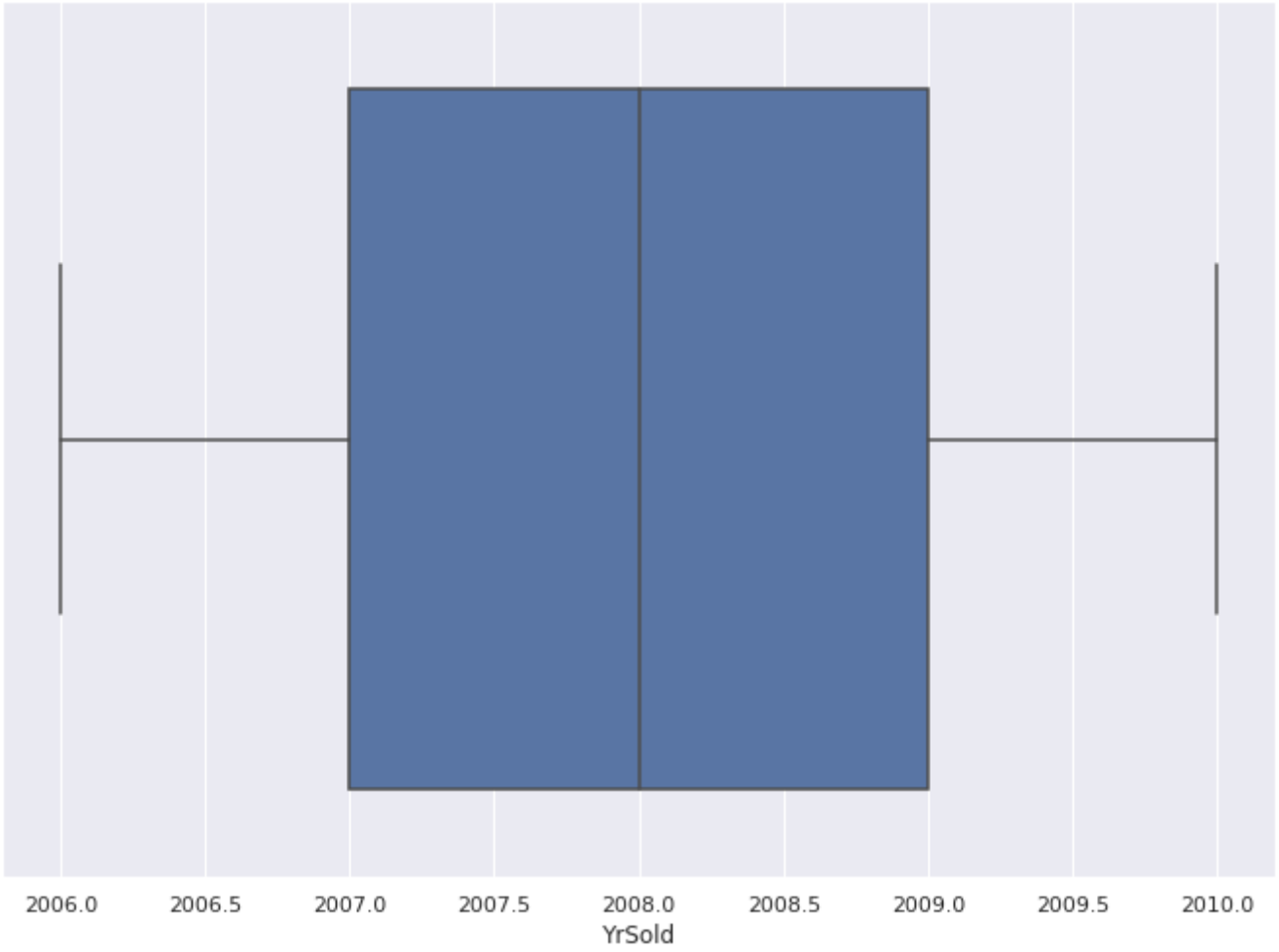

−0.04
−0.02
0.00
0.02
0.04
3SsnPorch

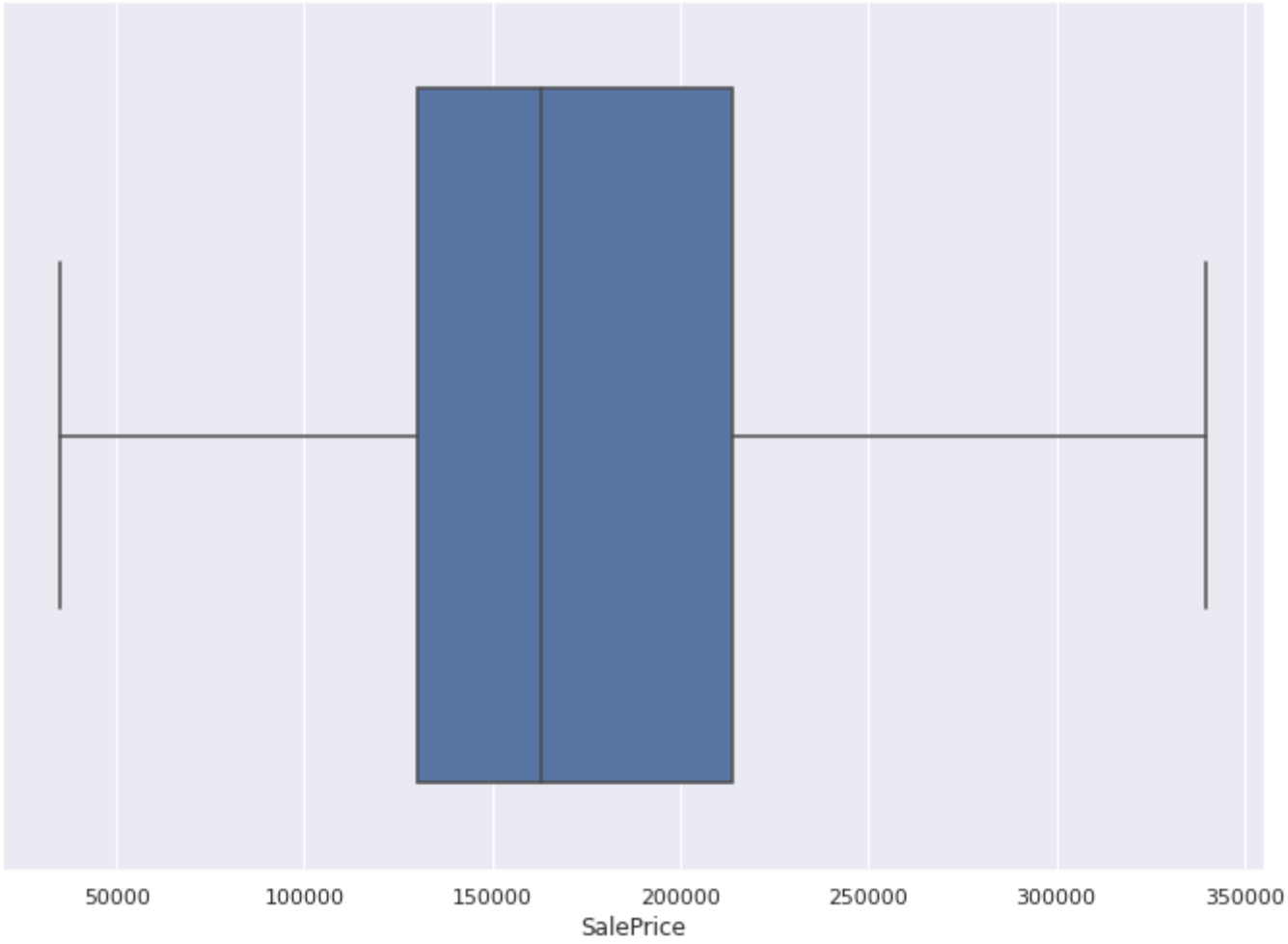

−0.04
−0.02
0.00
0.02
0.04
ScreenPorch


−0.04
−0.02
0.00
0.02
0.04
PoolArea


−0.04
−0.02
0.00
0.02
0.04
MiscVal

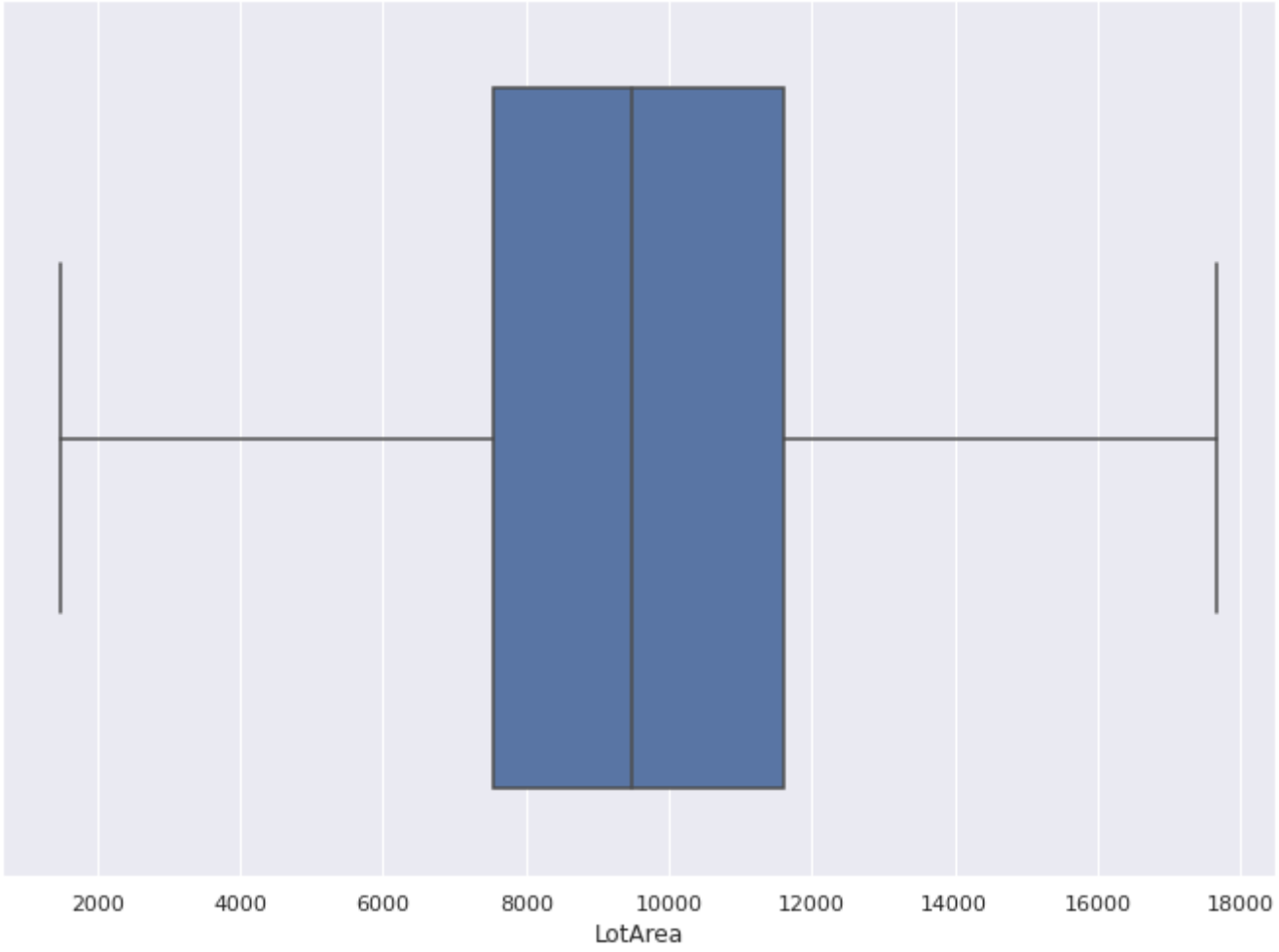

2
4
6
8
10
12
MoSold


2006.0
2006.5
2007.0
2007.5
2008.0
2008.5
2009.0
2009.5
2010.0
YrSold


50000
100000
150000
200000
250000
300000
350000
SalePrice

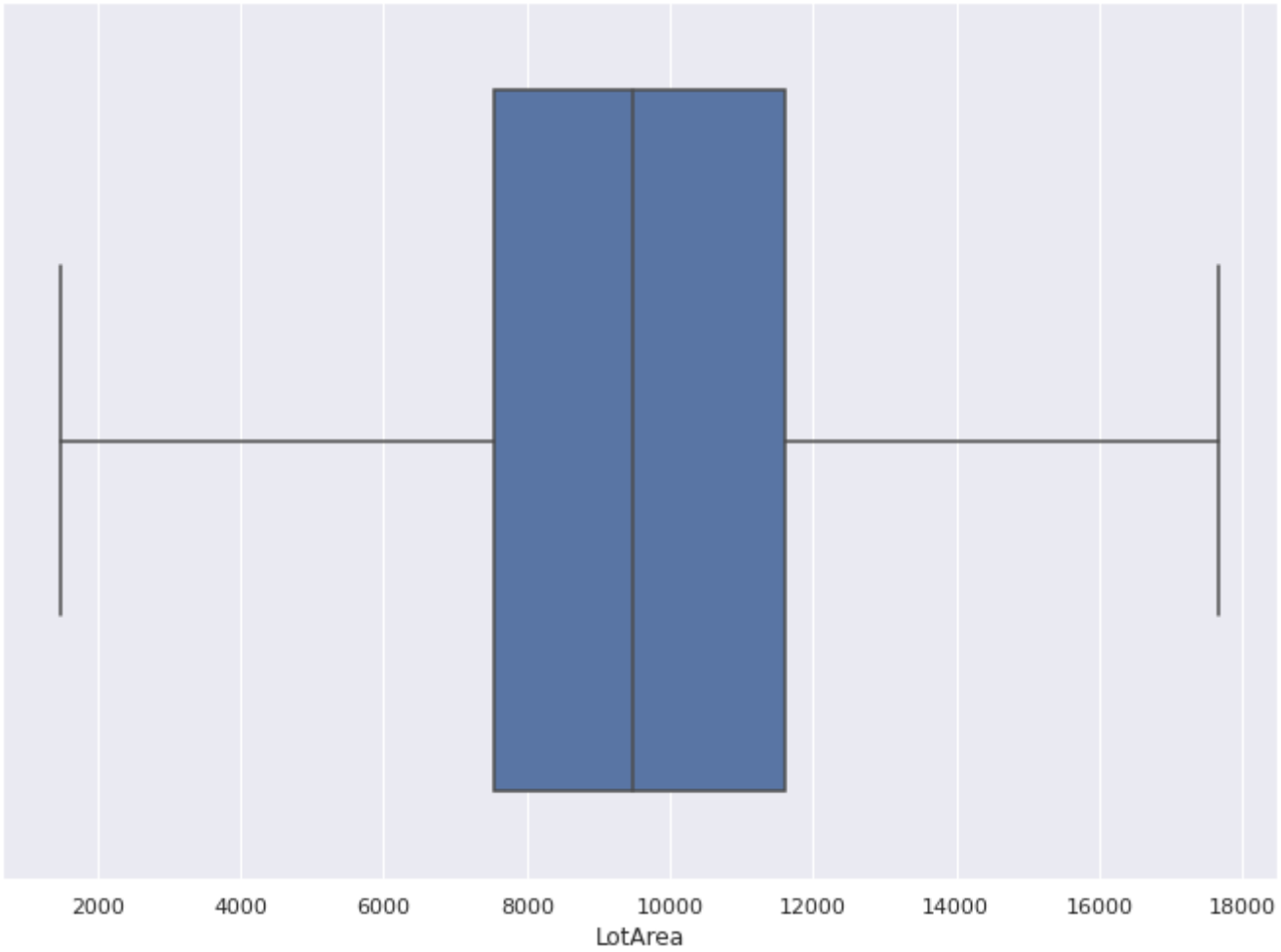
In [44]:
```
sns.set(rc={'figure.figsize':(11.7,8.27)})
sns.boxplot(df['LotArea'], showfliers = False)
```

Out[44]: <AxesSubplot:xlabel='LotArea'>

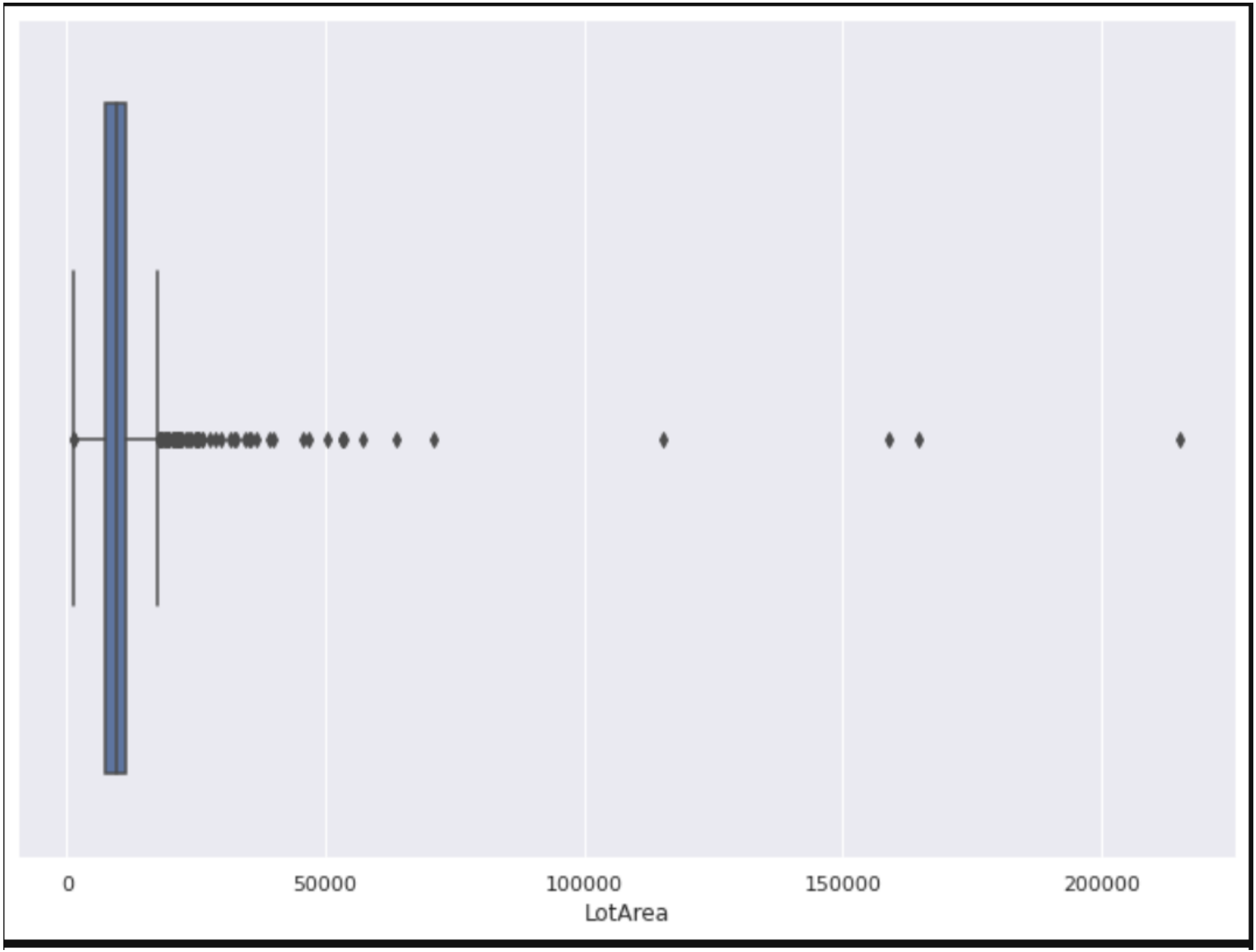


In [45]:
```
# Original DF using showfliers = False
sns.set(rc={'figure.figsize':(11.7,8.27)})
sns.boxplot(df['LotArea'], showfliers = False)
```

Out[45]: <AxesSubplot:xlabel='LotArea'>



## LOT AREA BEFORE


0
50000
100000
150000
200000
LotArea

## LOT AREA AFTER

In [ ]: