Assignment - 4

Assignment Date	5 November 2022
Student Name	M.Raghul
Student Roll Number	717819P125
Maximum Marks	2 Marks

1.Loading Dataset into tool

from google.colab import files uploaded = files.upload() import pandas as pd
import numpy as np

import matplotlib.pyplot as plt import seaborn as sns
import warnings warnings.filterwarnings('ignore') data = pd.read_csv("abalone.csv")

<IPython.core.display.HTML object> Saving abalone.csv to abalone.csv 2.Performing Visualization

Univariate Analysis

data.head()

Sex Length Diameter Height weight \
0 M 0.455 0.365 0.095 0.1010
1 M 0.350 0.265 0.090 0.0485
2 F 0.530 0.420 0.135 0.1415
3 M 0.440 0.365 0.125 0.1140
4 I 0.330 0.255 0.080 0.0395

Shell weight Rings

1. 0 0.150 15
2. 1 0.070 7
3. 2 0.210 9
4. 3 0.155 10
5. 4 0.055 7

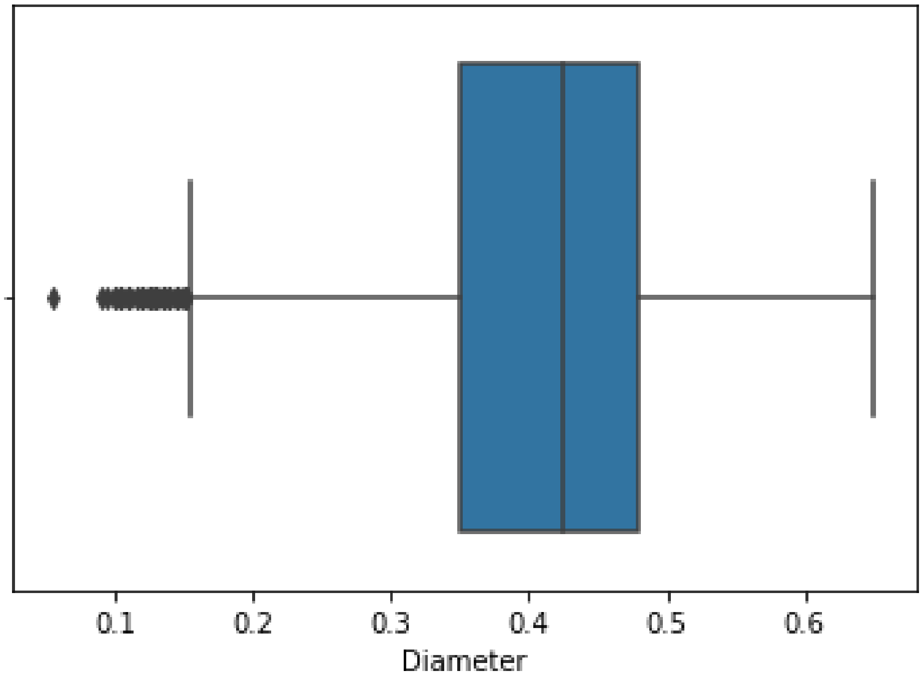
Whole

weight Shucked
0.5140
0.2255
0.6770
0.5160
0.2050
weight
0.2245

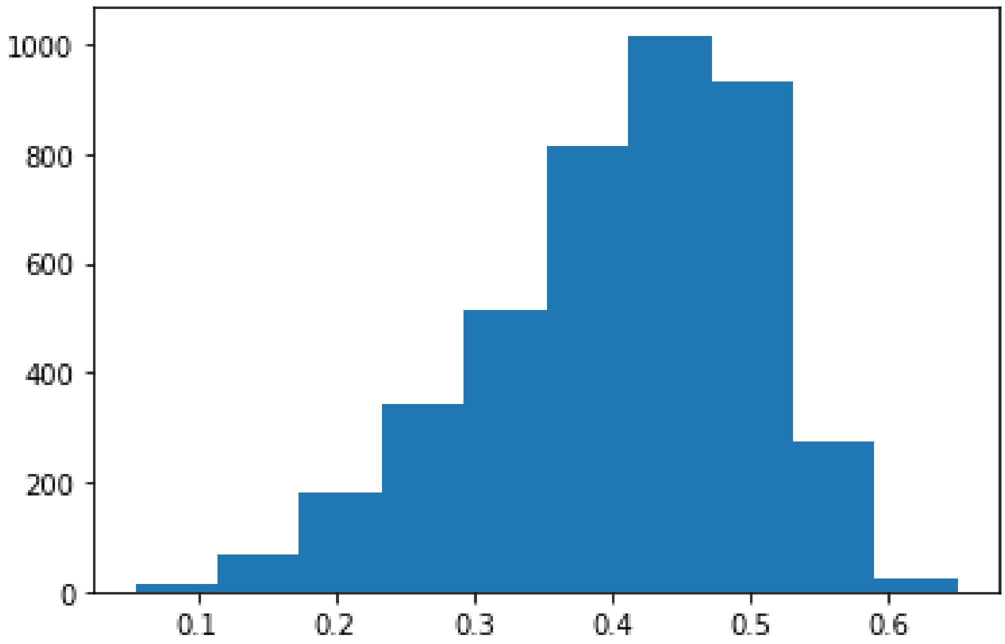
0.0995
0.2565
0.2155
0.0895

Viscera

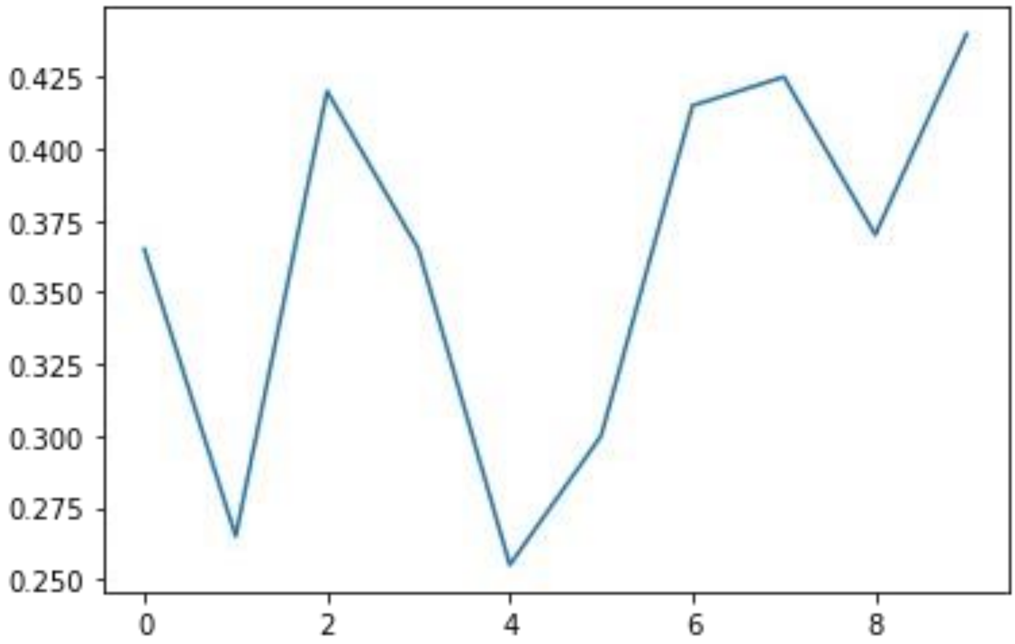
```
sns.boxplot(data['Diameter'])
<matplotlib.axes._subplots.AxesSubplot at 0x7f9624513c90>
```

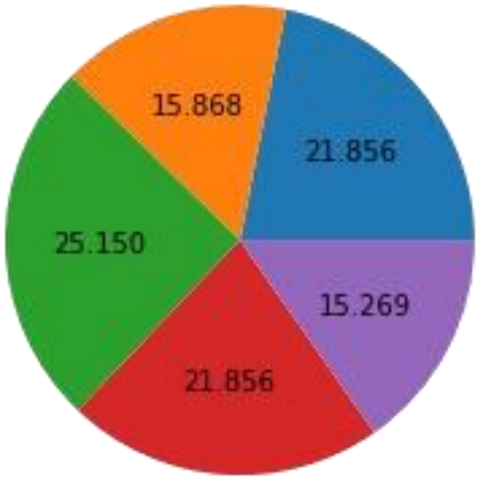



```
plt.hist(data['Diameter'])

(array([ 13., 66., 180., 344., 513., 812., 1017., 934., 275., 23.]),

array([0.055 , 0.1145, 0.174 , 0.2335, 0.293 , 0.3525, 0.412 , 0.4715,

       0.531 , 0.5905, 0.65  ]),
 <a list of 10 Patch objects>)
```

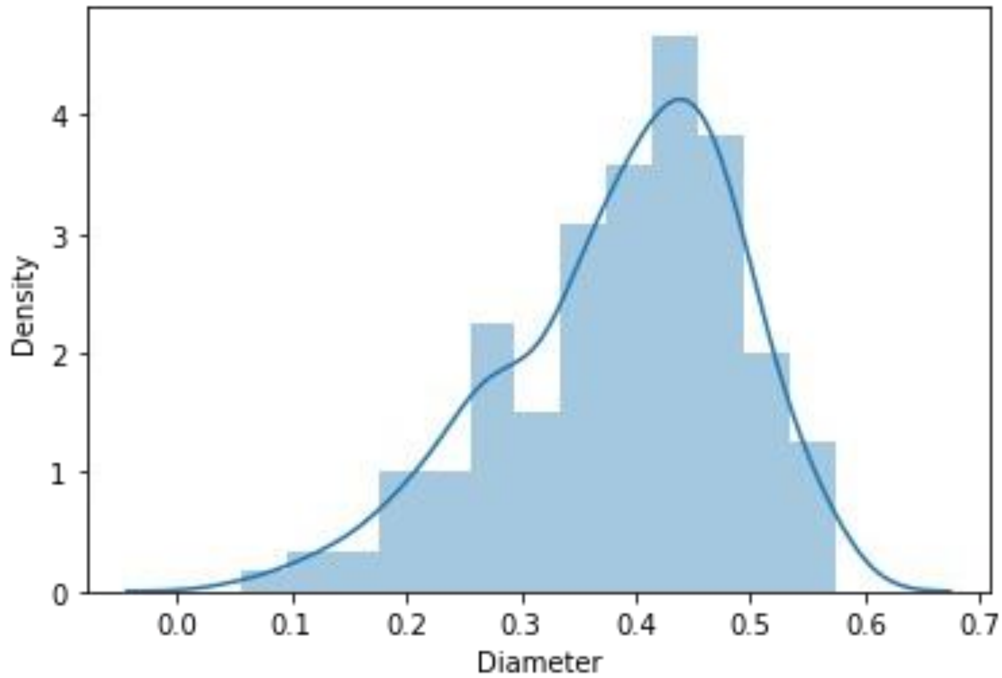


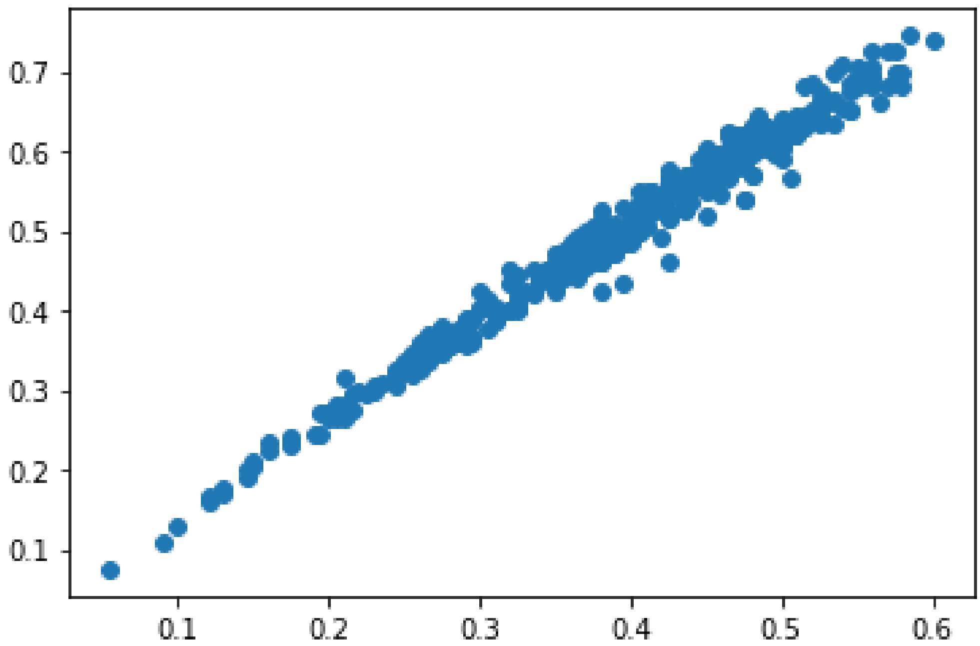

```
plt.plot(data['Diameter'].head(10))
[<matplotlib.lines.Line2D at 0x7f9623fc7ed0>]
plt.pie(data['Diameter'].head(),autopct='%.3f')
([<matplotlib.patches.Wedge at 0x7f9623edc590>,
  <matplotlib.patches.Wedge at 0x7f9623edcc10>,
  <matplotlib.patches.Wedge at 0x7f9623ee7650>,
  <matplotlib.patches.Wedge at 0x7f9623ee7e90>,
  <matplotlib.patches.Wedge at 0x7f9623e72990>],
 [Text(0.8507215626110557, 0.6973326486753676, ''),
  Text(-0.32611344931648134, 1.0505474849691026, ''),
  Text(-1.0998053664078908, -0.02069193128747144, ''),
  Text(-0.08269436219656089, -1.096887251480709, ''),
  Text(0.9758446362287218, -0.5076684409569241, '')],
 [Text(0.46402994324239394, 0.3803632629138369, '21.856'),
  Text(-0.17788006326353525, 0.5730259008922377, '15.868'),
  Text(-0.5998938362224858, -0.011286507974984419, '25.150'),
  Text(-0.045106015743578656, -0.5983021371712958, '21.856'),
  Text(0.5322788924883937, -0.2769100587037768, '15.269')]])
```



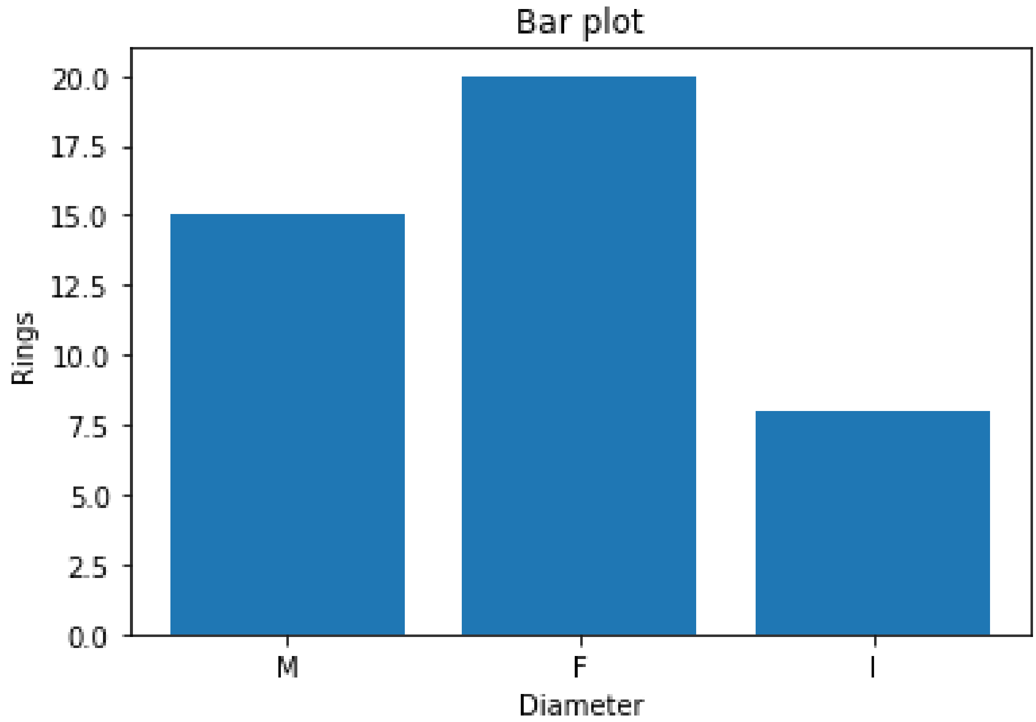



```
sns.distplot(data['Diameter'].head(300))
<matplotlib.axes._subplots.AxesSubplot at 0x7f9623e90250>
plt.scatter(data['Diameter'].head(400),data['Length'].head(400))
<matplotlib.collections.PathCollection at 0x7f9623d79c10>
```

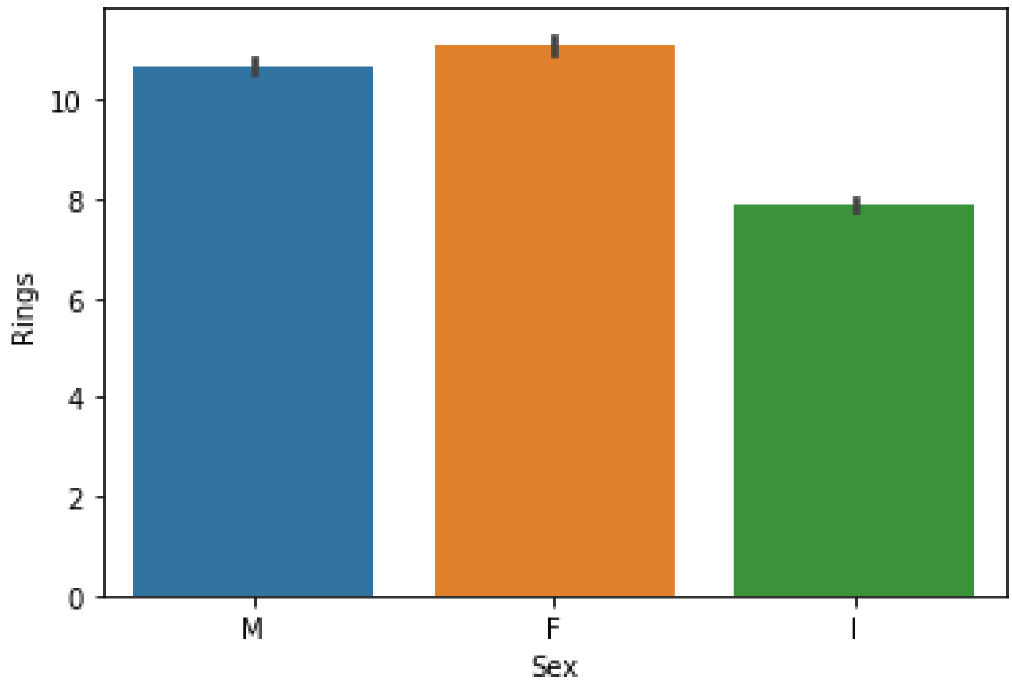


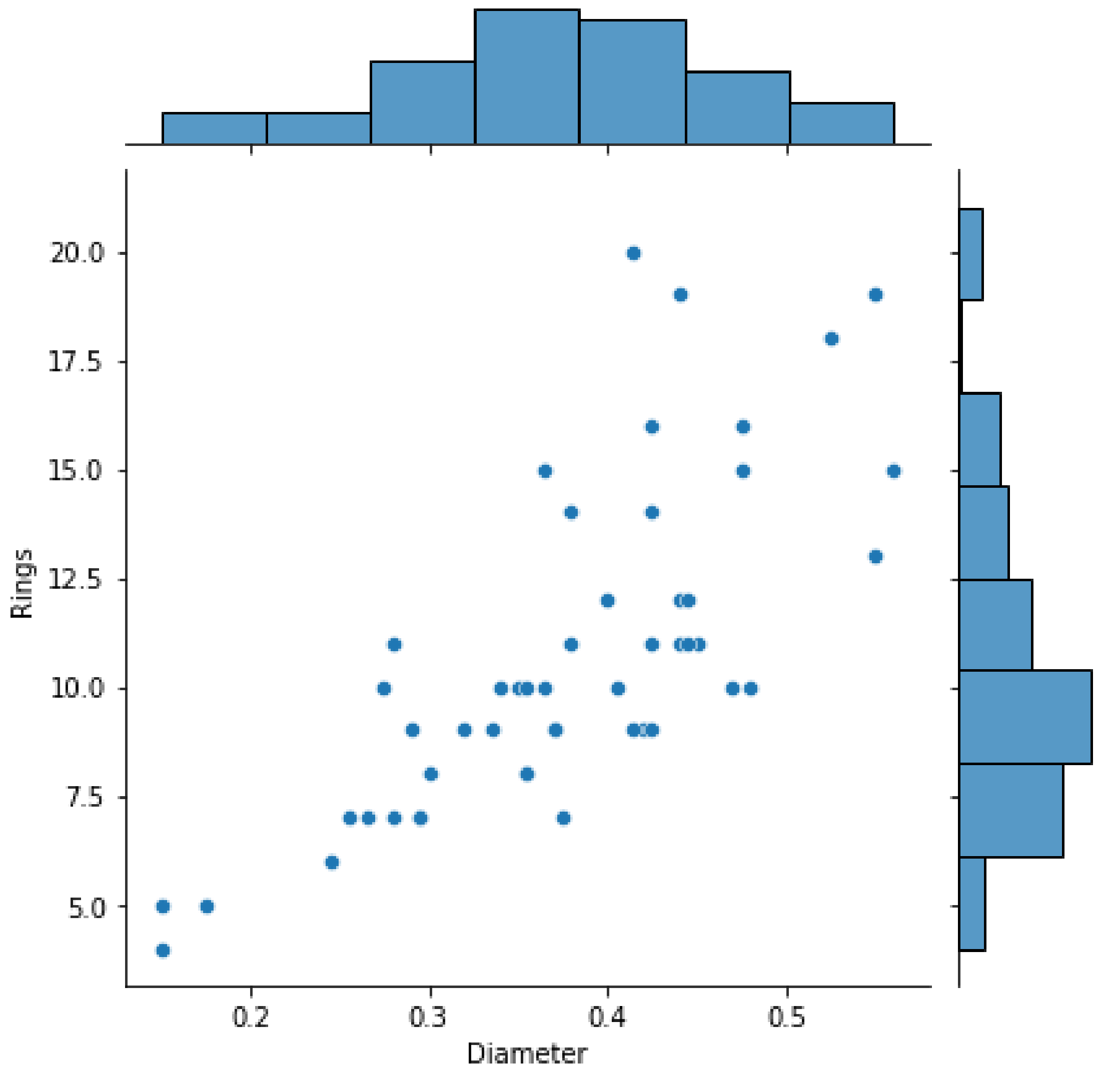



```
plt.bar(data['Sex'].head(20),data['Rings'].head(20))
plt.title('Bar plot')
plt.xlabel('Diameter')
plt.ylabel('Rings')
Text(0, 0.5, 'Rings')
```

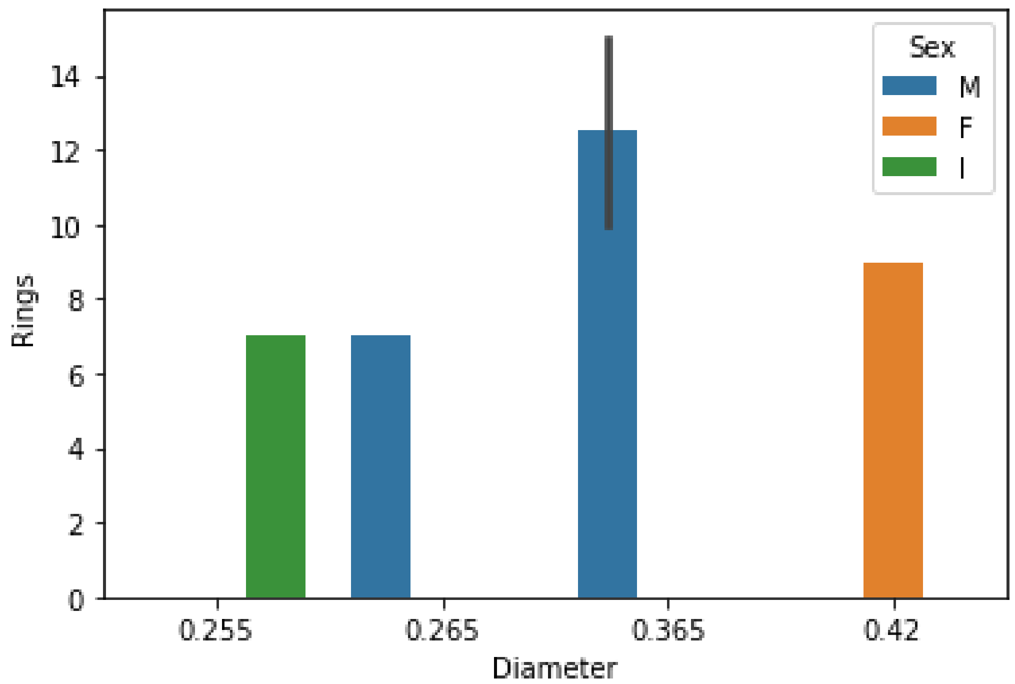



```
sns.barplot(data['Sex'], data['Rings'])
<matplotlib.axes._subplots.AxesSubplot at 0x7f9623ce8450>
sns.jointplot(data['Diameter'].head(50),data['Rings'].head(100))
<seaborn.axisgrid.JointGrid at 0x7f9623c44c50>
```

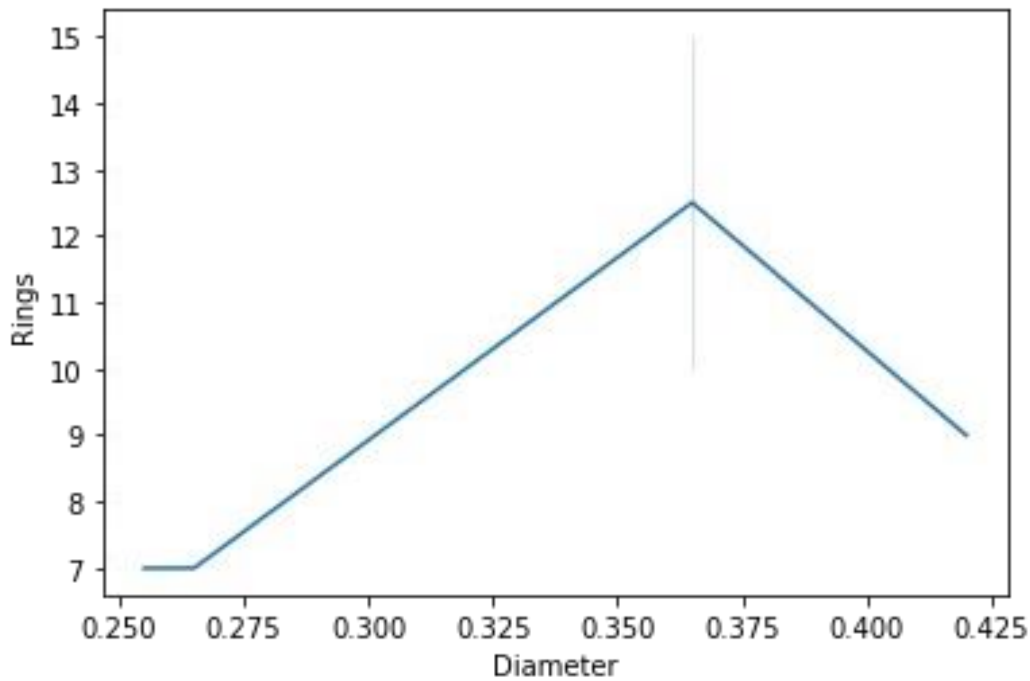


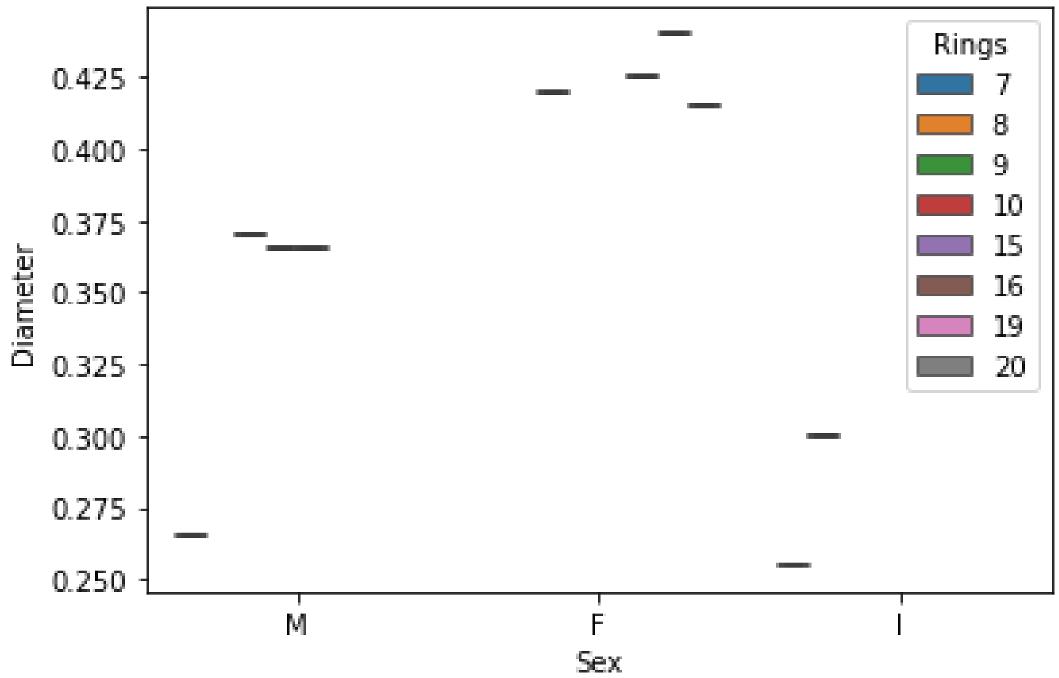



```
sns.barplot('Diameter','Rings',hue='Sex',data=data.head())
<matplotlib.axes._subplots.AxesSubplot at 0x7f9623b9e310>
```

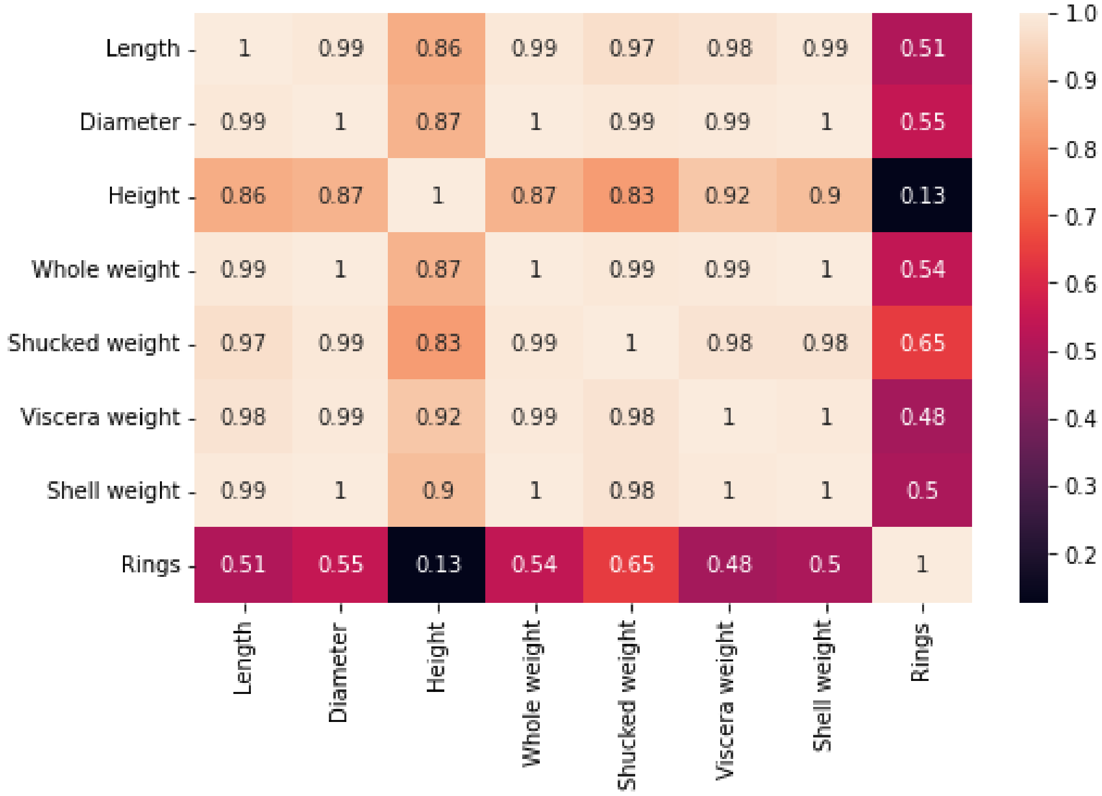


sns.lineplot(data['Diameter'].head(),data['Rings'].head())
<matplotlib.axes._subplots.AxesSubplot at 0x7f96212682d0>
sns.boxplot(data['Sex'].head(10),data['Diameter'].head(10),data['Rings
'].head(10))
<matplotlib.axes._subplots.AxesSubplot at 0x7f962116c4d0>

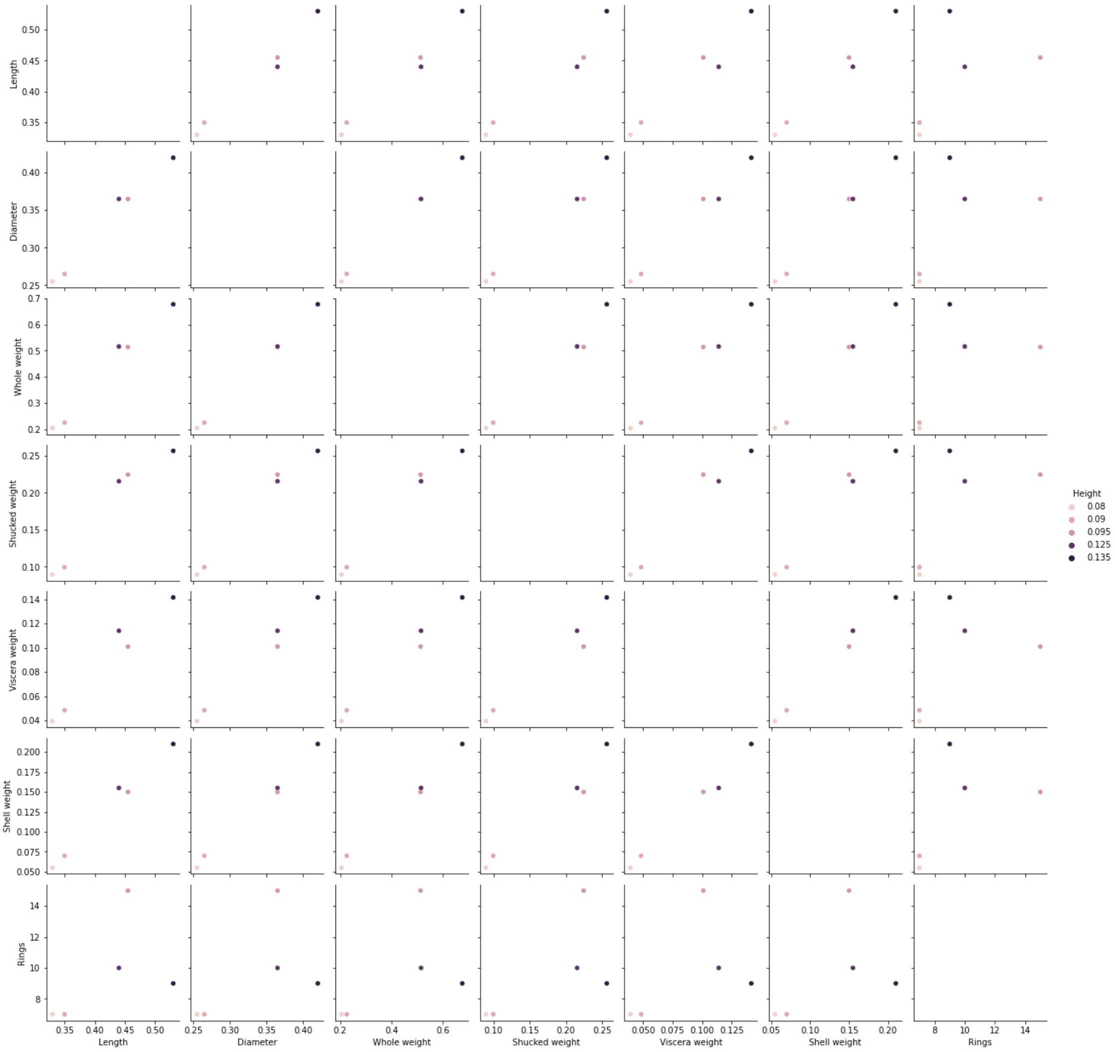




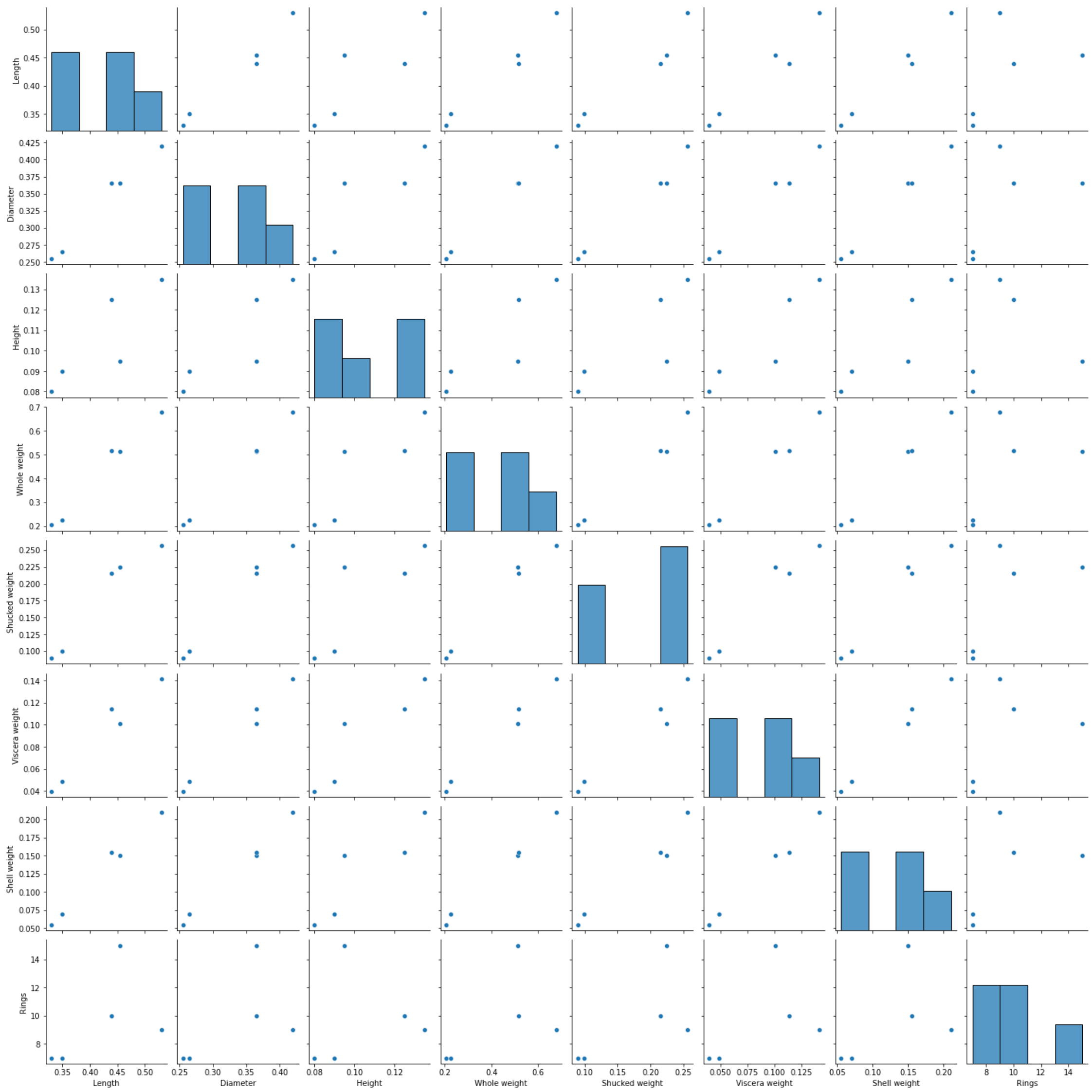
fig=plt.figure(figsize=(8,5))
sns.heatmap(data.head().corr(),annot=True)
<matplotlib.axes._subplots.AxesSubplot at 0x7f9621046fd0>



sns.pairplot(data.head(),hue='Height')
<seaborn.axisgrid.PairGrid at 0x7f9620eb5910>



sns.pairplot(data.head())
<seaborn.axisgrid.PairGrid at 0x7f961ff9fcd0>



3.Perform Descriptive Statistics on the dataset

data.head()

Sex Length Diameter Height weight \
0M 0.455 0.365 0.095 0.1010

1M 0.350 0.265 0.090 0.0485
2F 0.530 0.420 0.135 0.1415

3M 0.440 0.365 0.125 0.1140
4I 0.330 0.255 0.080 0.0395

Whole

weight Shucked
0.5140
0.2255
0.6770

0.5160
0.2050
weight
0.2245
0.0995
0.2565
0.2155
0.0895

Viscera

Shell weight Rings

1. 0 0.150 15
2. 1 0.070 7
3. 2 0.210 9
4. 3 0.155 10
5. 4 0.055 7

data.tail()
Sex Length Diameter

Whole

Rings
11
10
9
10
12
weight Shucked
0.8870
0.9660
1.1760
1.0945
1.9485
weight \
0.3700
0.4390
0.5255
0.5310
0.9455
4172
4173
4174
4175
4176
4172
4173
4174
4175
4176
F 0.565

M   0.590
M   0.600
F   0.625
M   0.710
Height
0.450   0.165
0.440   0.135
0.475   0.205
0.485   0.150
0.555   0.195

Viscera

weight Shell
0.2390
0.2145
0.2875
0.2610
0.3765
weight
0.2490
0.2605
0.3080
0.2960
0.4950
data.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4177 entries, 0 to 4176
Data columns (total 9 columns):
 #   Column
--- ------

1. 0 Sex
2. 1 Length
3. 2 Diameter
4. 3 Height
5. 4   Whole weight
6.
7. 5   Shucked weight
8.
9. 6   Viscera weight
10.
11. 7   Shell weight
12.

Non-Null Count
------------- -----
4177 non-null
4177 non-null
4177 non-null
4177 non-null
4177 non-null
4177 non-null

4177 non-null
4177 non-null
object
float64
float64
float64
float64
float64
float64
float64

Dtype

8 Rings 4177 non-null int64 dtypes: float64(7), int64(1), object(1)

memory usage: 293.8+ KB data.describe()

Length

weight \
count 4177.000000
4177.000000
mean 0.523992
0.359367
std 0.120093
0.221963
min 0.075000
0.001000
25% 0.450000
0.186000
50% 0.545000
0.336000
75% 0.615000
0.502000
max 0.815000
1.488000
Diameter
4177.000000
0.407881
0.099240
0.055000
0.350000
0.425000
0.480000
0.650000
Height
4177.000000
0.139516
0.041827
0.000000
0.115000
0.140000
0.165000

1.130000
Whole weight
4177.000000
0.828742
0.490389
0.002000
0.441500
0.799500
1.153000
2.825500

Shucked

Viscera weight
count 4177.000000
Shell weight
4177.000000
0.238831
0.139203
0.001500
0.130000
0.234000
0.329000
1.005000

01

Sex
Length
Diameter
Height
Whole weight
Shucked weight
Viscera weight 0.1715 NaN Shell weight 0.275 NaN Rings 9.0 NaN

Rings
4177.000000
9.933684
3.224169
1.000000
8.000000
9.000000
11.000000
29.000000
mean
std
min
25%
50%
75%
max
data.mode().T
0.180594

```
0.109614
0.000500
0.093500
0.171000
0.253000
0.760000
   M    NaN
  0.55  0.625
  0.45    NaN
  0.15    NaN
0.2225    NaN
 0.175    NaN
data.shape
(4177, 9)
data.kurt()
Length
Diameter
Height
Whole weight
Shucked weight
Viscera weight
Shell weight
Rings
dtype: float64
data.skew()
Length
Diameter
Height
Whole weight
Shucked weight
Viscera weight
Shell weight
Rings
dtype: float64
data.var()
Length
Diameter
Height
Whole weight
Shucked weight
Viscera weight
Shell weight
Rings
dtype: float64
data.nunique()
Sex
Length
Diameter
Height
Whole weight
Shucked weight
Viscera weight
Shell weight
Rings
```

```
dtype: int64
 0.064621
-0.045476
76.025509
-0.023644
 0.595124
 0.084012
 0.531926
 2.330687
-0.639873
-0.609198
 3.128817
 0.530959
 0.719098
 0.591852
 0.620927
 1.114102
 0.014422
 0.009849
 0.001750
 0.240481
 0.049268
 0.012015
 0.019377

10.395266

   3
 134
 111
  51
2429
1515
 880
 926
  28
```

4.Check for missing values and deal with them

data.isna()

Sex

1. 0 False
2. 1 False
3. 2 False
4. 3 False
5. 4 False

... ...

4172. 4172 False

4173. 4173 False
4174. 4174 False
4175. 4175 False
4176. 4176 False

Length
False
False
False
False
False
...
False
False
False
False
False
Diameter
False
False
False
False
False
...
False
False
False
False
False
Height
False
False
False
False
False
...
False
False
False
False
False
Whole weight
False
False
False
False
False
...
False
False
False
False
False

Shucked weight \
False
False
False
False
False
...
False
False
False
False
False
Viscera weight

1. 0 False
2. 1 False
3. 2 False
4. 3 False
5. 4 False

... ...

4172. 4172 False
4173. 4173 False
4174. 4174 False
4175. 4175 False
4176. 4176 False

Shell weight Rings
False False
False False
False False
False False
False False
... ...
False False
False False
False False
False False
False False
[4177 rows x 9 columns]
data.isna().any()
Sex
Length
Diameter
Height
Whole weight
Shucked weight
Viscera weight
Shell weight
Rings
dtype: bool
data.isna().sum()

Sex
Length
Diameter
Height
False
False
False
False
False
False
False
False
False

0 0 0 0

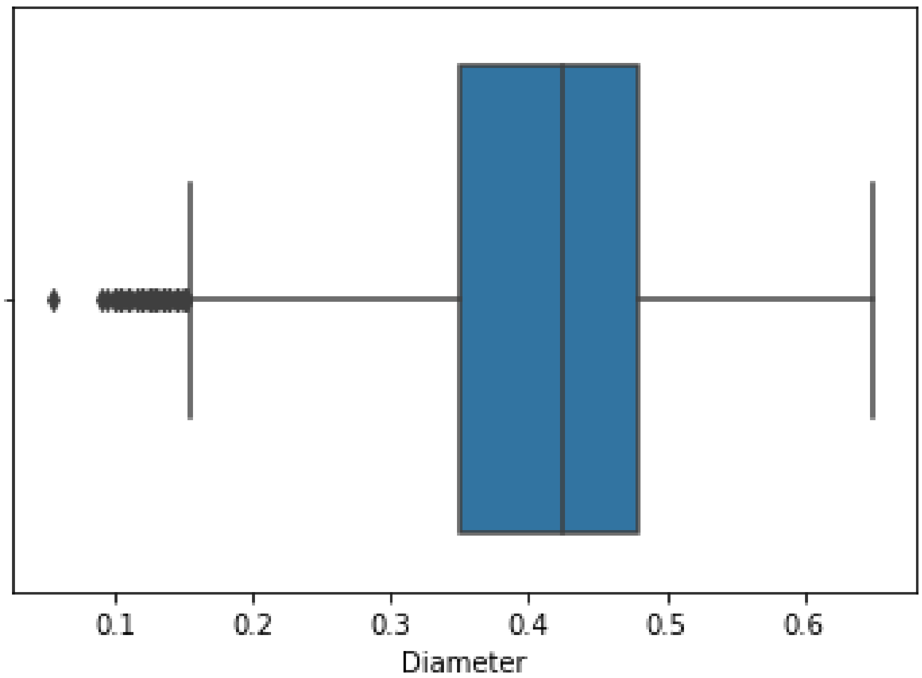
Whole weight 0
Shucked weight 0
Viscera weight 0
Shell weight 0
Rings 0
dtype: int64
data.isna().any().sum()

0

5.Find the outliers and replace them outliers

```
sns.boxplot(data['Diameter'])
<matplotlib.axes._subplots.AxesSubplot at 0x7f961c642c10>
quant=data.quantile(q=[0.25,0.75])
quant
```

Length Diameter Height Whole weight Shucked weight Viscera weight \



0.25 0.450 0.35 0.115 0.4415 0.0935
0.75 0.615 0.48 0.165 1.1530 0.2530

0.186 0.502

0.25 0.75

Shell weight Rings
0.130 8.0
0.329 11.0
iqr=quant.loc[0.75]-quant.loc[0.25]
iqr
Length
Diameter
Height
Whole weight
Shucked weight
Viscera weight
Shell weight
Rings
dtype: float64
0.1650
0.1300
0.0500
0.7115
0.3160
0.1595
0.1990
3.0000
low=quant.loc[0.25]-(1.5*iqr)
low
Length

Diameter
Height
Whole weight
Shucked weight
Viscera weight
Shell weight
Rings
dtype: float64
 0.20250
 0.15500
 0.04000
-0.62575
-0.28800
-0.14575
-0.16850

3.50000

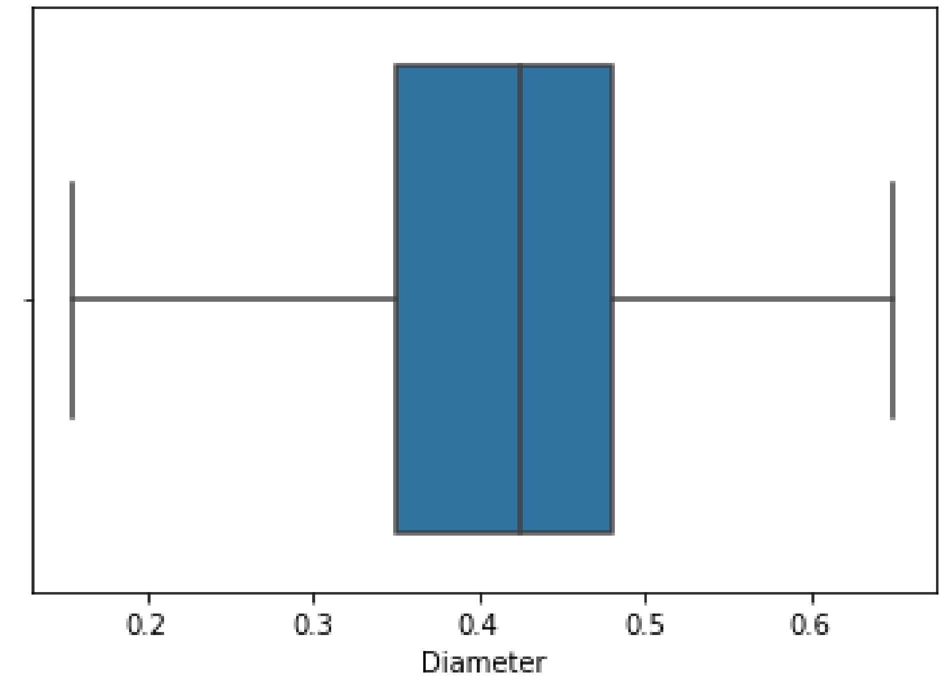
```
up=quant.loc[0.75]+(1.5*iqr)
up
```

Length
Diameter
Height
Whole weight
Shucked weight
Viscera weight
Shell weight
Rings
dtype: float64
 0.86250
 0.67500
 0.24000
 2.22025
 0.97600
 0.49225
 0.62750

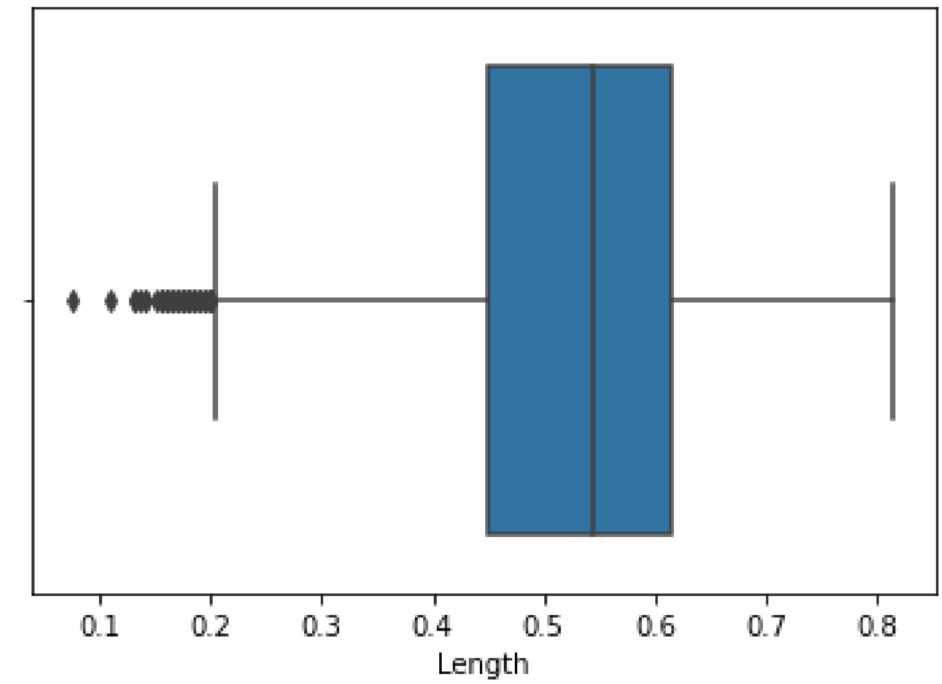
15.50000

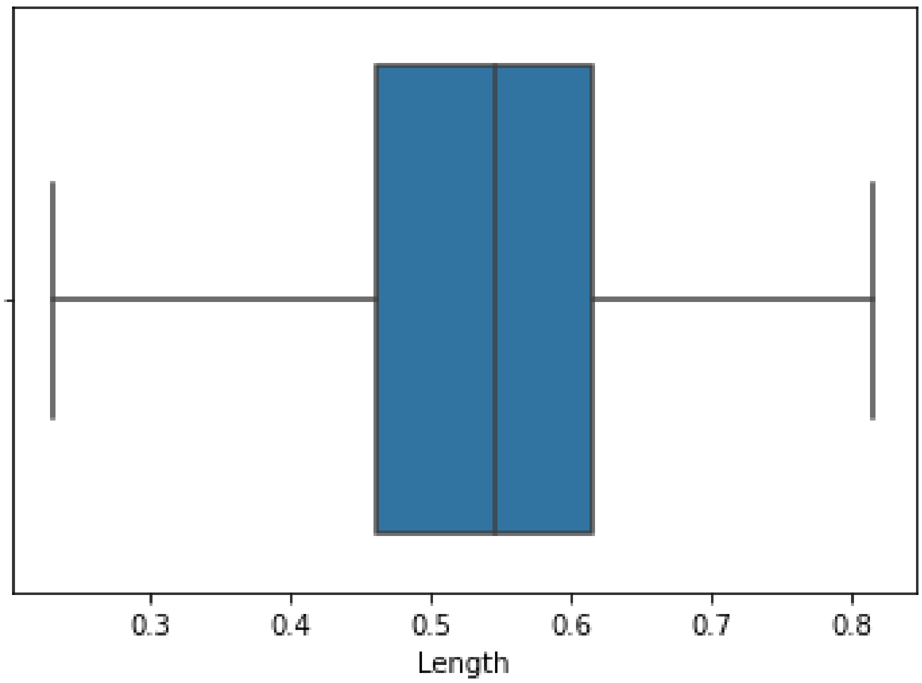
```
data['Diameter']=np.where(data['Diameter']<0.155,0.4078,data['Diameter
'])
sns.boxplot(data['Diameter'])
```

<matplotlib.axes._subplots.AxesSubplot at 0x7f961c621090>

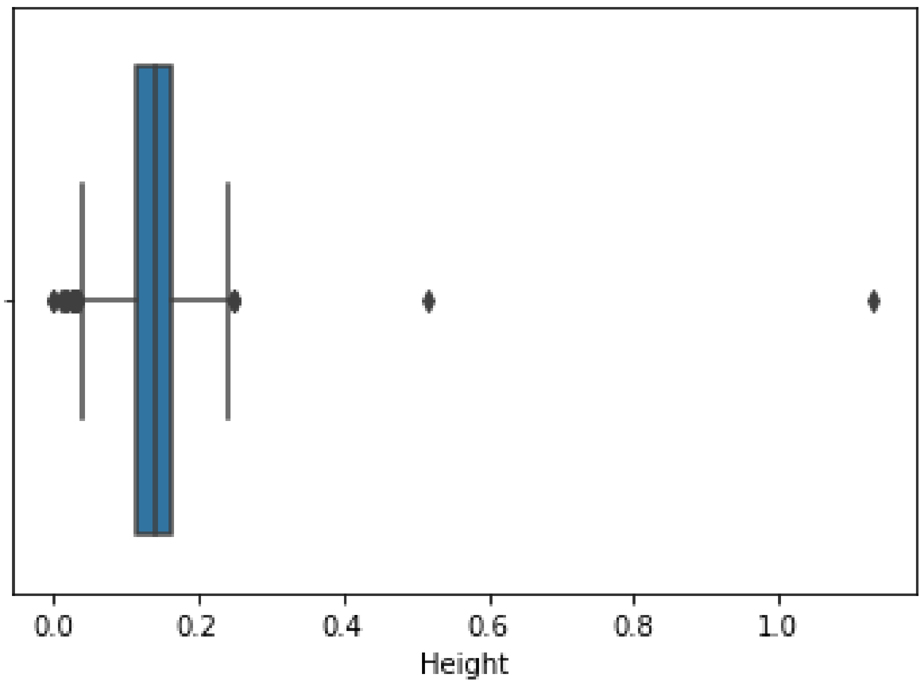



```
sns.boxplot(data['Length'])
<matplotlib.axes._subplots.AxesSubplot at 0x7f961c5ba1d0>
data['Length']=np.where(data['Length']<0.23,0.52, data['Length'])
sns.boxplot(data['Length'])
<matplotlib.axes._subplots.AxesSubplot at 0x7f961c5618d0>
```

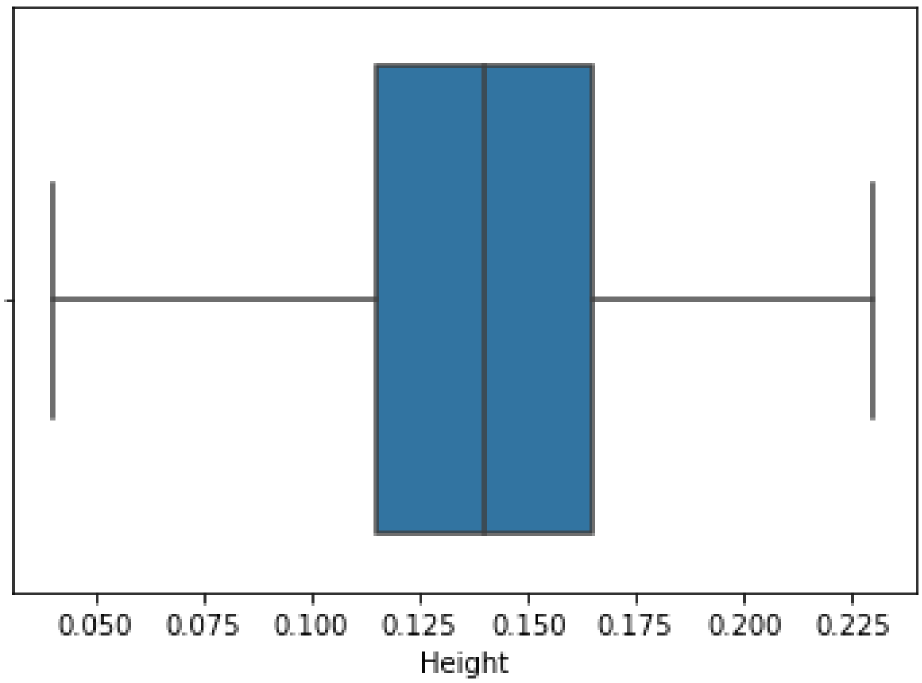


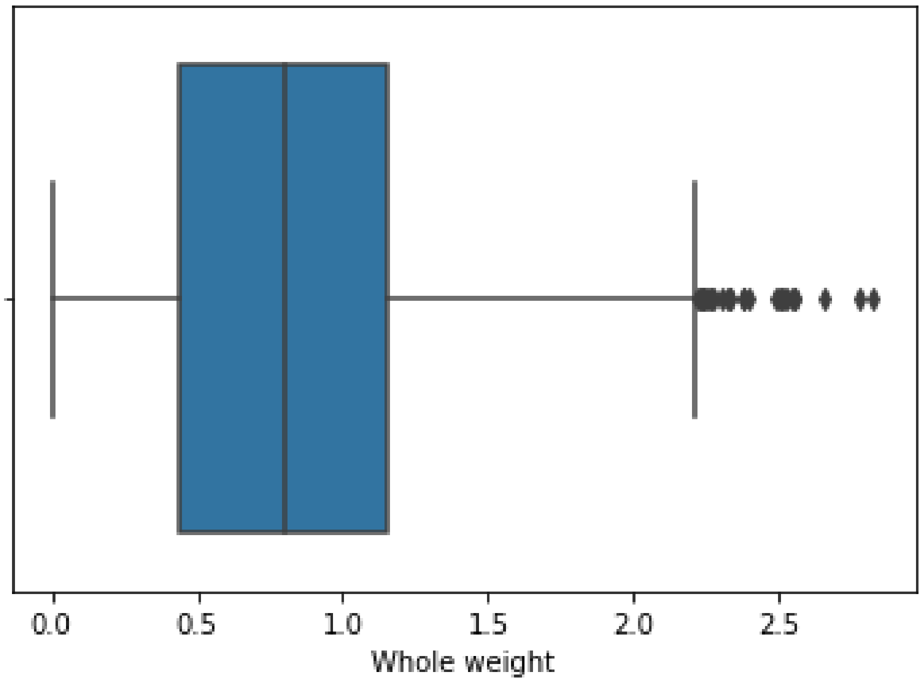


sns.boxplot(data['Height'])
<matplotlib.axes._subplots.AxesSubplot at 0x7f961c4b55d0>
data['Height']=np.where(data['Height']<0.04,0.139, data['Height'])
data['Height']=np.where(data['Height']>0.23,0.139, data['Height'])
sns.boxplot(data['Height'])

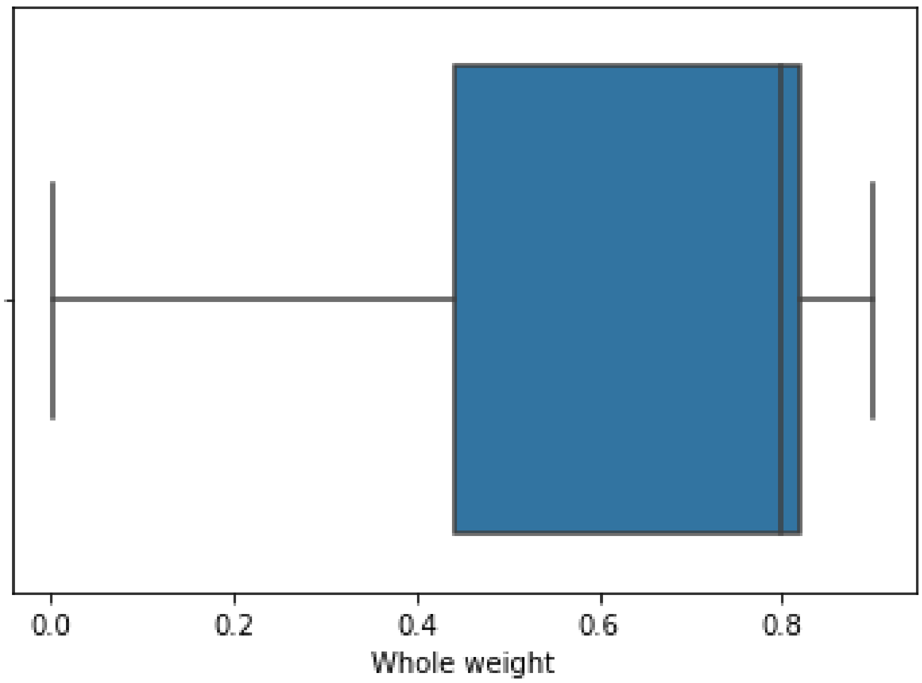


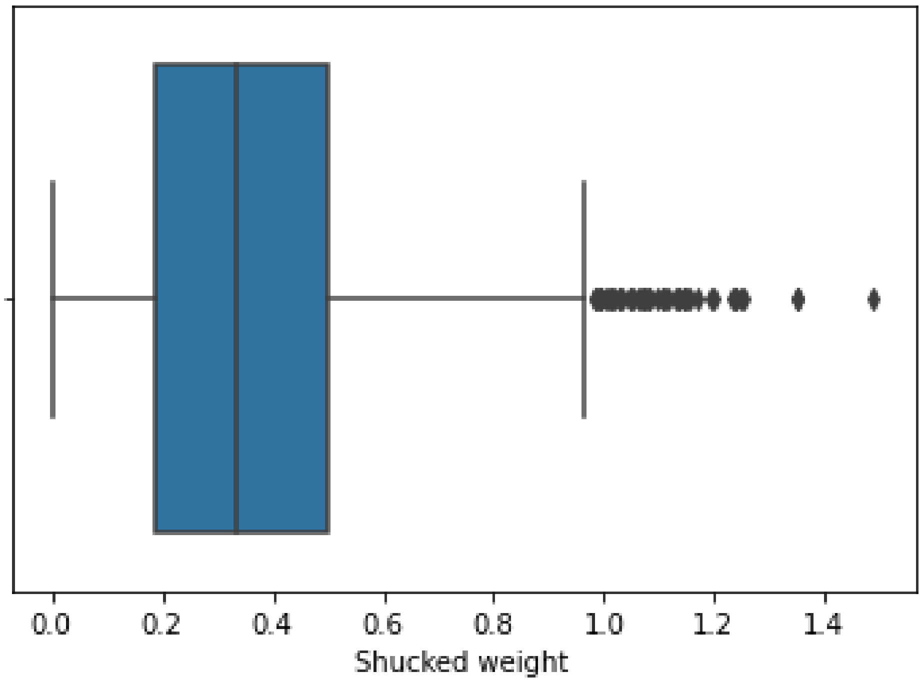
<matplotlib.axes._subplots.AxesSubplot at 0x7f961c4325d0>
sns.boxplot(data['Whole weight'])
<matplotlib.axes._subplots.AxesSubplot at 0x7f961c3eba10>



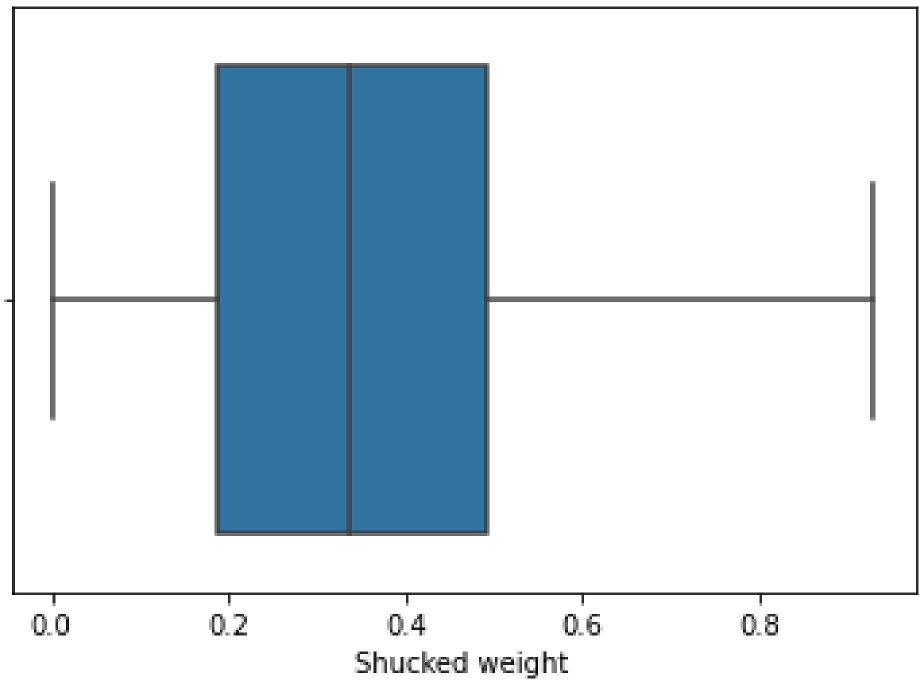



```
data['Whole weight']=np.where(data['Whole weight']>0.9,0.82,
data['Whole weight'])
sns.boxplot(data['Whole weight'])
<matplotlib.axes._subplots.AxesSubplot at 0x7f961c383d50>
sns.boxplot(data['Shucked weight'])
<matplotlib.axes._subplots.AxesSubplot at 0x7f961c2f6150>
```

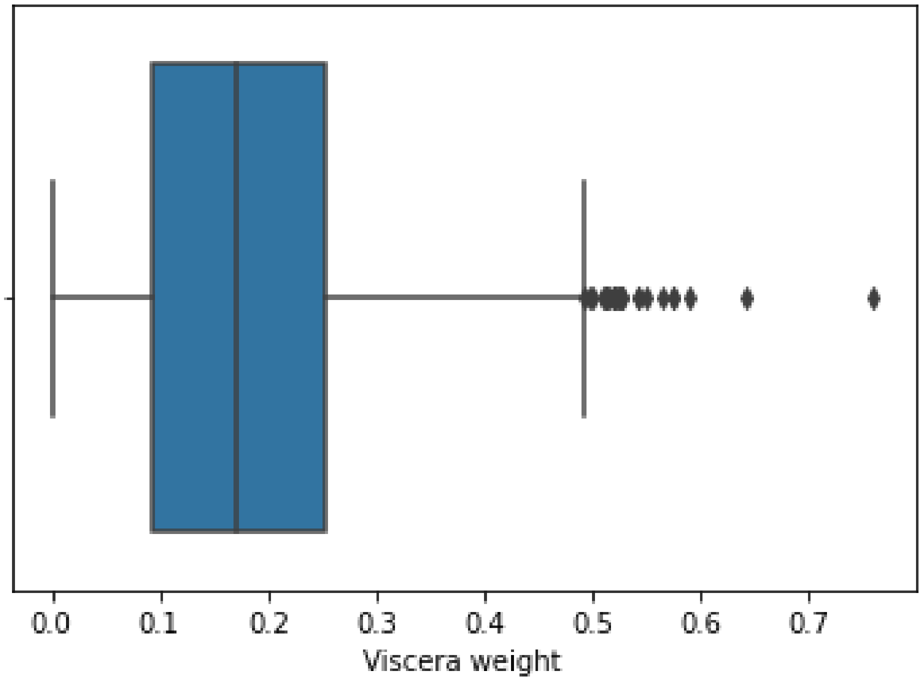


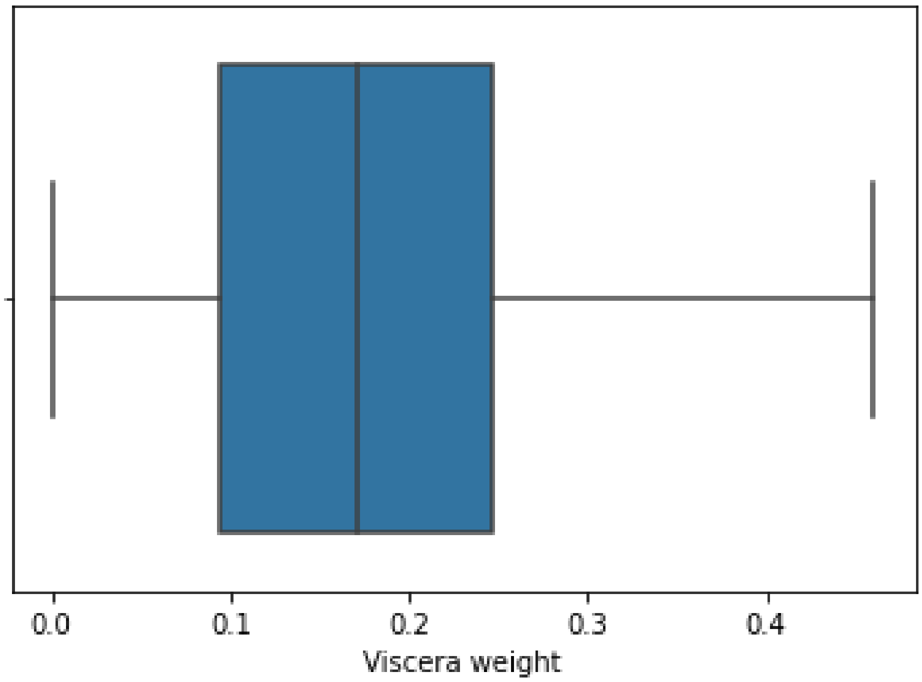



```
data['Shucked weight']=np.where(data['Shucked weight']>0.93,0.35,
data['Shucked weight'])
sns.boxplot(data['Shucked weight'])
<matplotlib.axes._subplots.AxesSubplot at 0x7f961c266bd0>
sns.boxplot(data['Viscera weight'])
```

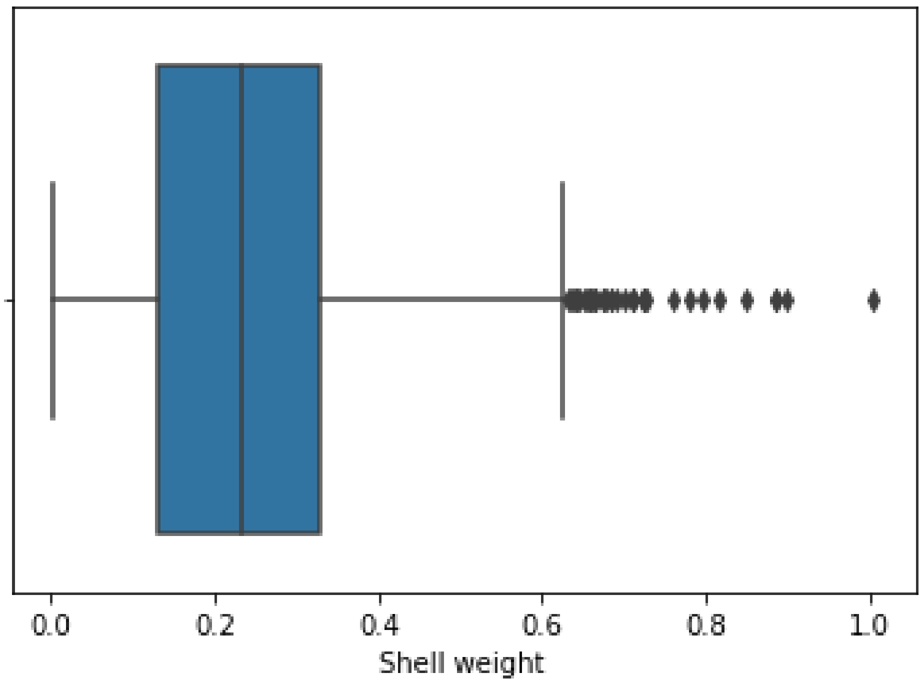


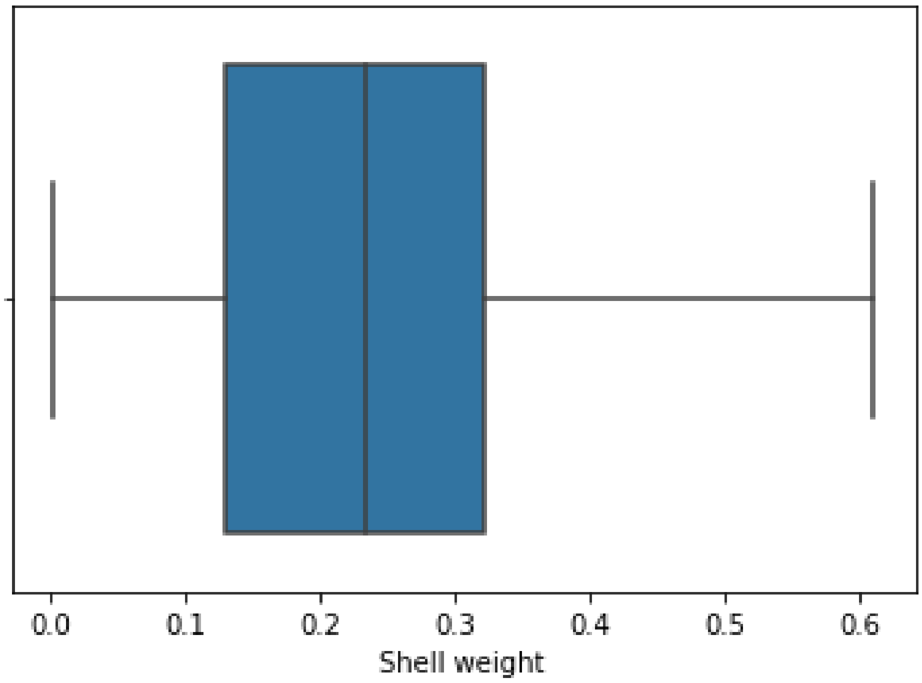

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f961c298e10>
data['Viscera weight']=np.where(data['Viscera weight']>0.46,0.18,
data['Viscera weight'])
sns.boxplot(data['Viscera weight'])
<matplotlib.axes._subplots.AxesSubplot at 0x7f961c396ad0>
```





sns.boxplot(data['Shell weight'])
<matplotlib.axes._subplots.AxesSubplot at 0x7f961c191f90>
data['Shell weight']=np.where(data['Shell weight']>0.61,0.2388,
data['Shell weight'])
sns.boxplot(data['Shell weight'])
<matplotlib.axes._subplots.AxesSubplot at 0x7f961c10dd50>





6.Check for Categorical columns and perform encoding.

```
data['Sex'].replace({'M':1,'F':0,'I':2},inplace=True)
data
```

Sex Length

1. 0 1 0.455
2. 1 1 0.350
3. 2 0 0.530
4. 3 1 0.440

5. 4 2 0.330

... ...

4172. 4172 0 0.565
4173. 4173 1 0.590
4174. 4174 1 0.600
4175. 4175 0 0.625
4176. 4176 1 0.710

Diameter Height
0.365 0.095
0.265 0.090
0.420 0.135
0.365 0.125
0.255 0.080
... ...
0.450 0.165
0.440 0.135
0.475 0.205
0.485 0.150
0.555 0.195
Shell weight
0.1500
0.0700
0.2100
0.1550
0.0550
...
0.2490
0.2605
0.3080
0.2960

Whole

weight Shucked
0.5140
0.2255
0.6770
0.5160
0.2050
...
0.8870
0.8200
0.8200
0.8200
0.8200
weight \
0.2245
0.0995
0.2565
0.2155

0.0895
...
0.3700
0.4390
0.5255
0.5310
0.3500
0
1
2
3
4
...
4172
4173
4174
4175

Viscera

weight
0.1010
0.0485
0.1415
0.1140
0.0395
...
0.2390
0.2145
0.2875
0.2610
Rings
15
7
9
10
7
...
11
10
9
10

4176 0.3765 0.4950 12 [4177 rows x 9 columns]

7.Split the data into dependent and independent variables.

```
x=data.drop(columns= ['Rings'])
y=data['Rings']
x
```

Sex Length Diameter Height

1. 0 1 0.455 0.365 0.095

2. 1 1 0.350 0.265 0.090
3. 2 0 0.530 0.420 0.135
4. 3 1 0.440 0.365 0.125
5. 4 2 0.330 0.255 0.080

... ...

4172. 4172 0 0.565 0.450 0.165
4173. 4173 1 0.590 0.440 0.135
4174. 4174 1 0.600 0.475 0.205
4175. 4175 0 0.625 0.485 0.150
4176. 4176 1 0.710 0.555 0.195
4177. Viscera weight Shell weight

Whole

weight Shucked
0.5140
0.2255
0.6770
0.5160
0.2050
...
0.8870
0.8200
0.8200
0.8200
0.8200
weight \
0.2245
0.0995
0.2565
0.2155
0.0895
...
0.3700
0.4390
0.5255
0.5310
0.3500

1. 0 0.1010
2. 1 0.0485
3. 2 0.1415
4. 3 0.1140
5. 4 0.0395

... ...

4172. 4172 0.2390
4173. 4173 0.2145
4174. 4174 0.2875

4175.   4175 0.2610
4176.   4176 0.3765

[4177 rows x 8 columns]

y

0 15 17 29 3 10 47

..
4172   11
4173   10
4174   9
4175   10
0.1500
0.0700
0.2100
0.1550
0.0550
...
0.2490
0.2605
0.3080
0.2960
0.4950

4176 12
Name: Rings, Length: 4177, dtype: int64

8.Scale the independent variables

from sklearn.preprocessing import scale x = scale(x)
x

array([[-0.0105225 , -0.67088921, -0.50179694, ..., -0.61037964, -0.7328165 , -0.64358742],

[-0.0105225 , -1.61376082, -1.57304487, ..., -1.22513334, -1.24343929, -1.25742181],

[-1.26630752, 0.00259051,
-0.33890749, -0.18321163],
...,
[-0.0105225 , 0.63117159,
1.08111018, 0.56873549],
[-1.26630752, 0.85566483,
0.82336724, 0.47666033],
[-0.0105225 , 1.61894185,
1.94673739, 2.00357336]])

9.Split the data into training and testing

0.08738942, ..., -0.45300269,
0.67657577, ..., 0.86994729,

0.78370057, ..., 0.89699645,
1.53357412, ..., 0.00683308,

```
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(x,y, test_size = 0.2)
print(x_train.shape, x_test.shape)
```

(3341, 8) (836, 8)

10.Build the Model

```
from sklearn.linear_model import LinearRegression MLR=LinearRegression()
```

11.Train the model

```
MLR.fit(x_train,y_train)
```

LinearRegression()

12.Test the model

```
y_pred=MLR.predict(x_test)
y_pred
```

array([11.54069678, 9.49895399, 6.52443921, 8.83112905, 7.17856508,

11.05657152, 11.64851477, 8.14081022, 10.27889967,
6.95196725,

8.55440106, 6.83326006, 13.65739688, 9.64587342,

8.9877502 ,
6.6198843 , 7.80248655, 8.80790722, 8.69792834,

11.26489978,
12.58112011, 15.56881783, 12.24601581, 7.78309109,
12.33762125,
11.95248582, 10.96621889, 10.29262391, 6.41772893,
12.64866118,
9.42259048, 11.69398584, 13.81562522, 10.56547208,
13.77344532,
9.46104164, 11.50841302, 9.09427682, 11.52031183,

9.70834125,
8.75237909, 10.55322532, 6.848486 , 10.7103863 ,

8.06918025,
9.7279291 , 9.73358905, 11.56150296, 8.6842891 ,

6.37816315,
7.73366501, 11.2405997 , 7.49066102, 9.62782361,

15.33711265,
9.8808388 , 13.66376956, 12.11126275, 8.49532274,

9.8658764 ,
14.14171599, 10.31795531, 9.16754365, 13.22474541,
6.92352778,
12.04355879, 11.15099777, 12.75412711, 9.20082015,

12.1712762 ,
8.89864407, 12.43181958, 6.81709844, 6.5583841 ,

9.980448 ,
7.51277931, 10.74486258, 11.35120432, 9.69690972,

7.8923869 ,
8.13969586, 8.8475261 , 7.36041772, 6.64092439,

11.26322908,
7.71181081, 8.97342293, 10.94593504, 10.4868579 ,

11.73430544,
8.5731762 , 10.51921164, 9.73482231, 9.15055446,

7.04296576,
10.13047939, 8.93314545, 7.37894174, 10.64731869,

11.36743605,
12.38734199, 9.24562918, 11.19180978, 11.07303436,

9.68069916,
7.58254393, 9.044852 , 9.95115811, 11.0582989 ,

11.72043006,
9.67082246, 11.09766282, 9.96802379, 7.53600837,
10.51973965,
9.17255415, 11.13257636, 10.63089427, 8.75554344,

13.41820769,
11.5738427 , 10.58324109, 7.60918431, 8.7857617 ,

7.7827081 ,
7.36470215, 9.94393466, 6.66549046, 7.69276842,

10.08500178,
11.4109651 , 10.62891559, 10.82433697, 10.59739469,

10.01546589,
11.71960777, 6.93158646, 6.21466601, 10.06788277,

10.72273275,
8.9801887 , 12.10745006, 12.04020196, 6.88791044,

13.29018362,
13.22568905, 9.27016388, 10.08534367, 11.90450271,

4.64919239,
13.61539616, 9.11384336, 12.8187526 , 4.31801239,

7.15719192,
4.8870934 , 12.90531414, 11.12777407, 10.02845693,

6.15250609,
13.09957628, 10.65641679, 13.42556673, 9.89013183,
7.78006145,
11.23521899, 14.51022451, 13.47270706, 11.85214133,
9.37453725,
11.65696338, 10.11378553, 5.86883913, 9.29581286,

13.19204926,
11.17191568, 6.5645379 , 15.48418468, 10.62798278,

12.2998294 ,
9.70337247, 13.01686409, 8.49448202, 9.16106491,
10.63770327,
8.35417634, 13.10041415, 13.42677458, 5.79134174,
11.33063144,
10.39325912, 13.38584917, 13.40916376, 10.42568761,
7.73749088,
9.16003971, 11.30348186, 7.95272523, 7.93512011,

7.86977829,
9.1494961 , 10.8396161 , 10.42791526, 12.06135993,

11.31430079,
13.98523552, 10.72480273, 10.76613282, 10.12098284,
6.66718688,
10.58914884, 10.93595723, 9.91502263, 11.29411918,

7.88521276,
9.33995138, 6.50229808, 11.58257584, 6.87771628,

12.67517329,
16.55763652, 10.69159895, 11.24279756, 9.13442737,
8.72323611,
10.22601629, 12.05258808, 8.71989817, 12.93656197,
6.45097046,
6.98213794, 11.52635762, 11.40841129, 10.76663776,

9.38412642,
10.89935953, 7.31982532, 11.97448724, 9.72507277,

12.26438271,
9.37215795, 7.12058354, 9.27677394, 8.63066054,

11.30215674,

6.08077549, 7.92229701, 7.2767753 , 10.88676783,

9.21305141,
12.15461642, 9.63110367, 7.70126284, 12.46114346,

7.59480401,
12.00577623, 10.28743849, 8.82975649, 11.45512588,
8.3442881 ,
10.02688931, 10.02630384, 11.44411395, 11.90229353,
12.00368773,
7.98148357, 11.33917258, 9.32368155, 7.93424756,

10.99753962,
7.91754348, 6.16968169, 8.42003195, 10.57324381,

8.73373817,
8.80732335, 10.8936687 , 8.61627177, 12.0350052 ,

12.42387851,
15.51804693, 9.82438264, 6.61711709, 8.53958333,

10.29161468,
5.94139478, 9.97690317, 10.336068 , 10.16515659,

9.15840085,
10.32360899, 11.30860679, 11.53640835, 7.85714689,

9.15165221,
6.38278905, 7.59652161, 10.16159927, 10.07843594,

7.97044259,
8.03569699, 8.02497031, 6.4322271 , 11.25501593,

11.83823874,
11.28503596, 6.87043255, 10.62894104, 10.99084524,

10.87908847,
12.06087707, 9.72764585, 7.79195275, 11.64611316,

10.57040484,
8.35775137, 12.23658447, 10.99945952, 15.50560555,

6.87374354,
8.60776074, 10.8593621 , 10.92229355, 11.07865536,

11.18738022,
10.93889866, 11.02435426, 10.99694487, 12.24766539,
7.87130066,
11.28334764, 13.07071067, 14.55709035, 8.95150329,

9.64381581,
8.53787374, 9.49814311, 11.04742947, 13.38410997,

10.43320536,
7.60519427, 8.32091959, 13.06360894, 14.50773568,

11.73965713,
12.48794147, 10.53951409, 11.04174252, 6.71936628,

11.1749459 ,
8.80586382, 8.26446109, 8.00674414, 11.58885536,

6.28875443,
8.33768758, 12.16236891, 10.08951878, 11.23161672,

11.92167545,
9.72787884, 9.89092893, 11.41601555, 8.75136282,

8.40485261,
6.99332406, 12.65369418, 9.28696132, 12.34199251,

11.55961076,
9.49910065, 7.71538709, 7.66748446, 7.92104409,

9.99653039,
12.10311262, 13.27010447, 13.89632175, 7.96276005,
8.28105499,
11.92062257, 11.86485603, 11.30236942, 11.72699485,

7.07287823,
10.01222378, 9.07595321, 10.99479604, 6.89937414,

14.57880665,
9.66840208, 9.34428944, 11.97554553, 7.06939809,

10.02929938,
10.01331059, 11.15814941, 13.12856179, 11.20333523,
7.57481498,
13.24071771, 10.50038133, 9.41093928, 11.99729048,
7.34077605,
7.85625332, 10.20192578, 10.00426864, 6.20358916,

10.16448511,
6.55642685, 10.77799647, 11.1162666 , 13.3863546 ,

11.92437896,
9.12040391, 5.99458465, 10.1199788 , 9.75812607,

7.02392668,
13.26021246, 9.4738486 , 10.97756881, 8.30744953,

10.68386186,
12.40904103, 7.22187259, 12.07846073, 9.60276861,

10.5635443 ,

7.73462142, 13.59163377, 9.06150914, 14.06291771,
10.43109922,
11.25463293, 11.92593621, 12.17267906, 8.77355942,

10.24123609,
13.80654218, 6.3145801 , 9.96318075, 10.62635096,

12.9487035 ,
8.78554027, 12.65575278, 11.00725139, 14.78443748,

8.43861728,
10.6263765 , 10.28836292, 10.01449497, 8.25546854,

12.1758287 ,
6.95523486, 8.90112605, 10.04798264, 10.43414356,

9.99962877,
11.03583022, 9.52482548, 10.7294708 , 8.86458229,
10.65738116,
11.12856712, 13.64794822, 9.60929351, 7.96351326,
9.32198228,
6.85888818, 10.87625026, 10.08503211, 11.25642064,
10.63925411,
10.21518367, 10.23375535, 12.56757299, 8.73411142,
10.28266098,
9.09557167, 12.33719483, 10.23237896, 12.03494537,

7.20341854,
9.04576135, 13.45789077, 8.05741121, 11.2391486 ,

9.90515751,
11.63561003, 10.82240081, 10.02462766, 10.37319587,

9.2853041 ,
7.32561844, 7.74274613, 7.96664097, 12.12810876,

9.65800013,
7.05560025, 7.0765866 , 8.08674607, 6.1525542 ,

12.74588665,
4.48862357, 9.35569147, 10.06336609, 10.56911428,

11.9991985 ,
11.031902 , 9.10207106, 9.16700788, 4.25036511,

10.44505832,
10.5248957 , 6.68575993, 15.96716935, 10.82662125,

8.09924411,
10.70869237, 12.35006318, 10.11425341, 9.28638073,
11.28456034,
6.88890746, 12.26664829, 8.33975993, 8.63395437,
12.7534647 ,

11.58794713, 12.32931121, 11.03947718, 14.44278997,
9.1526754 ,
9.96753573, 12.26568522, 11.35694972, 12.31490192,

12.81720079,
10.21722021, 16.29976957, 12.370311 , 12.24447926,

11.173701 ,
10.179486 , 11.81898172, 9.86346586, 9.27256167,
6.39317203,
7.96060048, 11.98230431, 7.20884879, 10.19870416,

10.54332877,
9.60829602, 8.90425344, 8.94845116, 9.35246022,

14.56196371,
11.03963102, 12.5023797 , 9.67399269, 9.3511401 ,

10.96072018,
10.69721757, 7.51690148, 11.511517 , 9.67811192,

7.33734619,
11.00287908, 11.87786789, 8.0526612 , 11.01033907,
13.36181267,
10.62125768, 10.75647649, 9.05527551, 9.39940694,

11.25291985,
9.67323418, 7.80449237, 9.04200461, 11.76068937,

8.95085535,
15.56892138, 10.43707649, 12.59822203, 12.83931103,

11.14957871,
11.93105691, 7.80490952, 10.47569057, 7.80852019,

10.03441283,
11.00290668, 6.3213835 , 13.28008836, 5.44262249,

10.0072763 ,
6.90588402, 10.62869477, 8.71239076, 10.87123808,

11.72001852,
7.19167124, 7.92167321, 7.68634043, 9.96813104,

10.21169043,
8.96348928, 10.75308121, 11.65412869, 14.58618679,
14.05054621,
12.02425713, 10.66056363, 8.00168655, 8.56812593,

6.02905054,
12.895106 , 9.4923062 , 11.47145176, 10.53720392,

12.83304005,
10.33670183, 16.03943759, 12.02522946, 8.5739522 ,

11.98857379,
8.82110981, 10.45519425, 9.1200697 , 10.52044421,

8.87813041,
6.72495962, 8.06392527, 9.19981327, 13.39096262,

9.26367146,
8.14837182, 8.85554945, 6.30515766, 10.25859235,

11.04432486,
11.93588945, 9.36256843, 9.68425387, 11.76199261,

9.00569387,
9.23569088, 11.51970577, 9.12069266, 7.94013073,

11.06286984,
12.92030963, 13.8943073 , 10.91871503, 8.74697235,

6.71089935,
14.71731037, 12.09769803, 8.54391856, 11.68705443,

11.08870433,
10.27946584, 10.51998226, 12.85751036, 9.9292065 ,

10.70116006,
8.89267373, 7.05760295, 11.68713402, 6.20648635,

10.82224239,
11.04772861, 11.3995166 , 10.33354214, 6.34144409,

15.50105519,
7.92176241, 9.87279381, 6.70097096, 7.83487912,

6.55063095,
5.93984218, 11.28170239, 7.97955425, 7.27822732,

10.22515415,
4.28748681, 7.5934957 , 12.16089328, 13.19196436,

11.89672074,
7.14104358, 10.51890117, 6.9754901 , 10.98348927,
11.85615632,
12.27658476, 10.26603983, 11.40583227, 7.99224812,

12.32434175,
11.09305975, 10.71953854, 10.25748861, 10.4759517 ,

8.49307286,
12.12488102, 8.65438527, 8.01858675, 7.14776458,

8.63032337,
11.2980198 , 11.29170328, 10.86003592, 6.26014084,

12.08001959,
10.43953314, 7.73987972, 11.99588048, 13.72401048,

8.63330819,
7.49253125, 10.20135478, 9.58333657, 10.5842399 ,

11.62679412,
7.68972791, 9.4646659 , 10.75059857, 6.69462086,

10.68554059,
7.55337542, 10.11978428, 11.15513888, 9.78756708,

7.86669589,
9.40048379, 5.66182888, 7.03918958, 11.57230128,

6.92239735,
10.71587496, 11.52892599, 7.79487777, 6.4461293 ,

11.59679899,
10.91807069, 11.65768382, 11.5984792 , 12.56106231,

10.52476051,
9.20765166, 14.4122295 , 8.04542852, 11.0218053 ,

6.12226262,
10.45628958, 11.0452751 , 10.65411905, 11.95752265,

11.4082904 ,
10.44351749, 5.87083955, 12.67297728, 9.14008833,

11.14123184,
10.97716762, 7.66987454, 10.21859958, 8.11501349,

4.03027355,
9.84334321, 11.14981452, 12.15089576, 7.67781124,

11.38619315,
13.20374376, 7.11300851, 8.49650293, 14.82672519,

9.7862761 ,
11.59659898, 7.85166071, 8.49747448, 7.43724277,

12.13105014,
10.07690974, 6.28311991, 10.87273209, 9.28976732,

7.25922486,
10.9465525 , 15.16016343, 7.79559915, 12.09636916,

10.19603123,
7.37850565, 6.6609198 , 6.75683481, 6.00570726,

10.00832715,
8.4120339 , 9.89161085, 13.39563291, 10.02679784,

10.73800476,
8.44704692, 12.26196649, 11.45369701, 9.22437065,
9.12619882,
7.25844296, 8.19338082, 10.0747906 , 8.26008144,
10.92636769,
7.66241316, 10.30171643, 11.85160087, 8.24892183,
11.82981831,
15.76865947, 8.55261702, 11.5525222 , 8.34384644,
6.62327531,
8.88634835, 13.30231386, 11.10002118, 9.72346458,
13.64771642,
10.26539644])
pred=MLR.predict(x_train)
pred

array([11.08442404, 7.61246988, 11.40560108, ..., 6.23338339, 6.68160695, 11.62326458])

from sklearn.metrics import r2_score accuracy=r2_score(y_test,y_pred) accuracy

0.4563299997265451 MLR.predict([[1,0.455,0.365,0.095,0.5140,0.2245,0.1010,0.150]])
array([9.87256759])
13.Measure the performance using Metrics

from sklearn import metrics
from sklearn.metrics import mean_squared_error np.sqrt(mean_squared_error(y_test,y_pred))

2.400658687128463

LASSO

from sklearn.linear_model import Lasso, Ridge

#intialising model

lso=Lasso(alpha=0.01,normalize=True) *#fit the model* lso.fit(x_train,y_train) Lasso(alpha=0.01, normalize=True) *#prediction on test data* lso_pred=lso.predict(x_test)

#coef

coef=lso.coef_
coef
array([-0.00867704, 0.
0.14076131,

0. , 0.

from sklearn import metrics
from sklearn.metrics import mean_squared_error metrics.r2_score(y_test,lso_pred)

0.35217661094369934 np.sqrt(mean_squared_error(y_test,lso_pred)) 2.6205415255898603
RIDGE

#initialising model

rg=Ridge(alpha=0.01,normalize=True) *#fit the model* rg.fit(x_train,y_train) Ridge(alpha=0.01, normalize=True) *#prediction* rg_pred=rg.predict(x_test)

rg_pred

array([11.48829185, 9.49185638, 6.52855715, 8.88572713, 7.1794515 ,

11.05400464, 11.61368185, 8.12311058, 10.32334947,
6.97106698,

8.57358187, 6.92975464, 13.52565076, 9.71367977, 8.99229802,

6.59818005, 7.83324476, 8.82713334, 8.70295194, 11.31079488,

12.46405845, 15.41781124, 12.20179819, 7.80859249,
12.21941921,
11.89746333, 11.02051962, 10.37978812, 6.40274613,
, 0.
, 0.81943443])
, 0.47886483,
12.5752096 ,
9.49536694, 11.73343905, 13.70396588, 10.53086589,

13.5982056 ,
9.45932051, 11.46540406, 9.32111182, 11.6327897 ,

9.67613273,
8.77272442, 10.55104433, 6.94417975, 10.67002081,

8.22482202,
9.69096894, 9.80602087, 11.60076362, 8.7238234 ,

6.37643978,
7.71860431, 11.30796156, 7.51879937, 9.69349058,
15.22763351,
9.86282438, 13.54770668, 12.0519293 , 8.51635754,
10.01110922,
13.96406923, 10.35099925, 9.17449114, 13.06598803,
6.91933105,
11.940958 , 11.09800171, 12.64262738, 9.17196555,

12.14184668,
8.86549075, 12.32471479, 6.91419185, 6.5446013 ,

9.87679619,
7.49979378, 10.76058357, 11.25199274, 9.86106422,

7.89388401,
8.37874926, 8.85212072, 7.35881066, 6.62699443,

11.2281232 ,
7.6803694 , 8.94168637, 10.95538742, 10.53536396,

11.75822781,
8.6316109 , 10.57898259, 9.72618332, 9.19702998,

7.05287605,
10.1853031 , 8.93128132, 7.38872276, 10.65090156,

11.36569117,
12.41541576, 9.2367783 , 11.11851486, 11.10208616,

9.68954224,
7.75166796, 9.05878184, 9.93400923, 11.07427757,

11.68715435,
9.66808682, 11.04776757, 9.91355145, 7.53134185,
10.56537734,
9.18490047, 11.15626926, 10.63660332, 8.73460768,

13.27876897,
11.56363276, 10.5354652 , 7.63624197, 8.78808886,

7.84197225,
7.36098072, 9.92803929, 6.69636345, 7.68963635,

10.03629225,
11.40370911, 10.62421834, 10.80896965, 10.65490043,

10.11986272,
11.69670119, 6.96258016, 6.20255724, 10.06623424,

10.72316963,
8.99348206, 12.09107221, 11.97218142, 6.91257446,

13.19318951,
13.13463583, 9.28686352, 10.0689656 , 11.91721126,

4.7631824 ,
13.51087731, 9.11531517, 12.81740474, 4.43780222,

7.16790279,
5.00112592, 12.86453911, 11.10478879, 10.01843888,

6.1461971 ,
13.0028451 , 10.79678097, 13.36006937, 9.94739918,

7.83117841,
11.18475737, 14.32890667, 13.39670664, 11.85487015,
9.33234782,
11.59427795, 10.15109076, 5.85820739, 9.30382537,

13.18305534,
11.11248639, 6.54275907, 15.29214822, 10.55212795,

12.32715875,
9.68922795, 12.94650132, 8.54586199, 9.15940396,

10.60976459,
8.3247563 , 12.96480032, 13.40086387, 5.78719843,

11.3238015 ,
10.45312483, 13.37895337, 13.32003513, 10.36924346,
7.76356965,
9.17163962, 11.30404613, 7.95803521, 7.94620503,
7.83062425,
9.15279212, 10.82942256, 10.43350984, 12.06239446,
11.29198439,
13.86994304, 10.77448157, 10.84682398, 10.08766098,

6.68154558,
10.6373005 , 11.04170201, 9.90525856, 11.29544118,

7.9058032 ,
9.35249912, 6.48343295, 11.56325071, 6.91258952,

12.63592168,
16.38940653, 10.70491488, 11.16838137, 9.10528211,
8.85104643,
10.23729585, 11.89814771, 8.71838891, 12.79383571,

6.44079822,
6.9707811 , 11.43537137, 11.40028719, 10.7716477 ,

9.36116658,
10.94608619, 7.34322513, 11.8331405 , 9.84685072,

12.11155414,
9.45815459, 7.11376981, 9.25863825, 8.65726028,

11.31982481,
6.07524434, 7.97503133, 7.30637776, 10.81806281,

9.20357185,
12.07204695, 9.61627488, 7.74887857, 12.442504 ,

7.58886688,
11.99380739, 10.24772157, 8.77839811, 11.45146639,

8.39899075,
10.11320108, 10.0245763 , 11.41739069, 11.82376497,

11.96302391,
7.97452309, 11.30414343, 9.44022391, 7.94857294,
10.97816286,
7.89419368, 6.17345341, 8.4501155 , 10.59587552,
8.85373305,
8.87748269, 10.77863866, 8.61787217, 12.01199382,

12.42080795,
15.40130648, 9.86589674, 6.65727677, 8.57728889,

10.217948 ,
5.90972332, 10.00264519, 10.41249197, 10.26984907,
9.21742817,
10.37729794, 11.33307884, 11.55995441, 7.91042502,

9.15348908,
6.3819584 , 7.61432151, 10.13642697, 10.19031162,

7.99443324,
8.0825615 , 8.08070422, 6.47164249, 11.24267189,

11.84608469,
11.29310399, 6.89175627, 10.56923366, 11.09699615,

10.90404062,
11.99250267, 9.80642905, 7.76902873, 11.57262024,

10.6298646 ,
8.3825652 , 12.26505129, 10.94913874, 15.33660141,

6.9182721 ,
8.66492578, 10.8733721 , 10.93365862, 11.02661621,

11.16986407,
10.96052154, 11.02382056, 10.97735488, 12.20802827,

7.8451432 ,
11.2313359 , 13.01478444, 14.40754815, 9.06007376,

9.72137464,
8.56117855, 9.62320742, 11.06000929, 13.33161608,

10.43813182,
7.56920004, 8.35508948, 12.99272773, 14.37281313,

11.7633051 ,
12.39452222, 10.47734612, 11.05119654, 6.71210553,

11.12718781,
8.83418598, 8.29063699, 8.04954168, 11.53899331,

6.25774546,
8.35866575, 12.16360395, 10.08106856, 11.17571594,

11.91187196,
9.8289716 , 9.93515592, 11.32459497, 8.75770579,

8.45203268,
7.01934846, 12.66381774, 9.28363872, 12.23227455,

11.57595187,
9.53096951, 7.73378041, 7.68983706, 7.91609618,

10.07741908,
12.02544182, 13.19229437, 13.75259075, 7.99893428,
8.30439167,
11.86533153, 11.86727546, 11.24065322, 11.73004046,

7.07480034,
10.10675942, 9.20759516, 11.05798055, 6.87157171,

14.43409935,
9.64182668, 9.38025051, 11.97495499, 7.08438188,

10.03435746,
10.04408419, 11.14318561, 13.05714042, 11.20794865,
7.56695973,
13.13045988, 10.55064571, 9.33754557, 11.98287976,

7.33022646,
7.8561511 , 10.23271788, 9.97719298, 6.21740842,

10.22347911,
6.53635592, 10.74978627, 11.11174262, 13.3400204 ,

11.85700009,
9.18547194, 5.99196026, 10.11339035, 9.75578879,

7.01064331,
13.12775642, 9.44123659, 10.94166899, 8.30328587,

10.65615788,
12.3805864 , 7.24653988, 12.02247978, 9.65949225,

10.5457433 ,
7.76029501, 13.44617552, 9.04120181, 13.93357462,
10.41136326,
11.25612879, 11.94969346, 12.15010205, 8.78209152,

10.17434646,
13.7477988 , 6.32931606, 9.92530257, 10.65781341,

12.87900268,
8.80958739, 12.53166118, 11.10652958, 14.65226729,

8.48241303,
10.61712776, 10.28878344, 10.15604084, 8.244834 ,

12.18860294,
6.930467 , 8.91382648, 10.02706252, 10.47403862,

10.03477945,
11.02756519, 9.48187924, 10.64974581, 8.86664196,

10.7649816 ,
11.1032976 , 13.51835898, 9.75094048, 8.02888855,

9.33012471,
6.83033455, 10.94304311, 10.21005067, 11.19062782,
10.66915739,
10.15175642, 10.24459739, 12.45319066, 8.72214027,
10.29686448,
9.13018844, 12.27167032, 10.20668047, 12.00545148,

7.18755909,
9.0397991 , 13.37428074, 8.10959737, 11.25904744,

9.94240709,
11.62649409, 10.78812888, 10.01422036, 10.32715966,

9.29001675,
7.3266678 , 7.75068445, 7.97596299, 12.15602707,

9.73032029,
7.05584208, 7.09625707, 8.11239959, 6.16027227,

12.66807021,
4.60567084, 9.343517 , 10.06244919, 10.65287057,

12.03523628,
10.99342212, 9.09767521, 9.17552459, 4.36839568,

10.42384283,
10.60574791, 6.68956059, 15.80742279, 10.7951137 ,

8.08189712,
10.6926337 , 12.21128071, 10.14654396, 9.41739236,

11.28195112,
6.93248938, 12.26885255, 8.39572045, 8.7406461 ,

12.69034168,
11.49318133, 12.25964808, 11.08014253, 14.37427062,
9.21617514,
9.96818875, 12.27758929, 11.28015796, 12.22329462,
12.77737085,
10.23352345, 16.08497568, 12.37872426, 12.14813951,

11.18622756,
10.23787776, 11.8158512 , 9.82904188, 9.33779071,

6.3901006 ,
8.03757195, 11.94852768, 7.22909104, 10.18668172,

10.5022804 ,
9.72012851, 8.88451057, 8.92805863, 9.33431875,

14.39529363,
11.10718108, 12.2307042 , 9.65006821, 9.36205093,

11.05877757,
10.69884279, 7.53724444, 11.50104215, 9.72891912,

7.35115781,
11.04419274, 11.79715097, 8.05207826, 11.06362993,
13.2671064 ,
10.64596476, 10.69952882, 9.24874574, 9.51034352,

11.2793115 ,
9.81975112, 7.81825267, 9.03856725, 11.76784756,

9.02808058,
15.44633968, 10.40604045, 12.57636427, 12.85224006,

11.09291857,
11.95890952, 7.83312891, 10.44266764, 7.79073352,

10.0046309 ,
11.02375359, 6.33725408, 13.11043708, 5.4294093 ,

10.04486242,
6.99905506, 10.67360861, 8.74171042, 10.84105225,

11.66436845,
7.20706858, 7.92164592, 7.68446934, 10.02502452,

10.23595405,
9.0013227 , 10.68778694, 11.66251239, 14.44788625,

14.05511425,
11.94679343, 10.64891309, 7.9968667 , 8.56823768,

6.00915406,
12.76016519, 9.50907796, 11.48713264, 10.5648931 ,

12.79860507,
10.38920132, 15.85105763, 12.01034026, 8.57200724,

11.99949995,
8.82714977, 10.4553905 , 9.13303597, 10.57447283,

8.86155487,
6.73062625, 8.07342506, 9.17215383, 13.2858116 ,

9.26436748,
8.18105105, 9.0107502 , 6.26498889, 10.21779876,

11.0409425 ,
11.92544274, 9.33789576, 9.67647565, 11.77542023,

9.02666917,
9.25561352, 11.57869119, 9.12013424, 8.20186749,

11.06676989,
12.7788211 , 13.76491299, 10.94068 , 8.75672159,

6.690432 ,
14.4956619 , 12.06751912, 8.57345473, 11.64608222,

11.04176895,
10.36441448, 10.56046046, 12.86718174, 9.99253427,

10.603819 ,
8.90708521, 7.08501527, 11.61623972, 6.20261783,

10.88939725,
11.00938563, 11.41003739, 10.32121915, 6.34835823,
15.26133349,
7.92926401, 9.85697784, 6.7916182 , 7.80934371,

6.56335507,
5.92765412, 11.25343232, 8.01402096, 7.2498473 ,

10.40425517,
4.40903645, 7.614689 , 12.13145246, 13.12946032,

11.86873141,
7.16295937, 10.57670554, 6.98745094, 11.02971488,
11.86152536,
12.25054238, 10.35007946, 11.36374006, 8.00953907,
12.26949318,
11.05956441, 10.64906129, 10.20220601, 10.52735951,
8.51091687,
12.11579486, 8.65857757, 8.0365289 , 7.16009108,

8.59460134,
11.23746879, 11.33333432, 10.92356515, 6.28305417,

12.07855713,
10.48508398, 7.72373265, 11.95374443, 13.60155892,

8.61754843,
7.52310305, 10.27127525, 9.69304382, 10.61604583,

11.67588014,
7.67551672, 9.50184394, 10.77950122, 6.71044345,

10.68353891,
7.58866907, 10.14479145, 11.15488106, 9.87437438,

7.86423978,
9.49931356, 5.6554684 , 7.03718954, 11.54741398,

7.1869371 ,
10.71132375, 11.49623964, 7.84589086, 6.44741686,

11.5387835 ,
10.95640556, 11.7450581 , 11.61736114, 12.42439193,

10.51406192,
9.21374258, 14.28205012, 8.02749499, 11.10564932,

6.10609667,
10.352845 , 11.06878526, 10.66838316, 12.01038164,

11.34981665,
10.47836566, 5.87230133, 12.69151735, 9.15469969,

11.21324205,
11.03403314, 7.69318537, 10.29300193, 8.17806607,

4.43535199,
9.82008619, 11.24436145, 12.12476841, 7.66668345,
11.36774655,
13.16987311, 7.09171718,
9.83090434,
11.59774925, 7.85011687,
12.13273433,
10.0800497 , 6.25534426,

7.28333626,

8.51840563, 14.70366035,
8.52249436, 7.49062757,
10.7752602 , 9.29336829,
11.00926295, 15.00915824, 7.80843969, 12.03795447,

```
10.16456012,
7.38610632, 6.67694658, 6.76395885, 6.00137317,

10.07374731,
8.40384552, 9.90397019, 13.25680879, 10.10226594,

10.79620112,
       8.47371473, 12.27215101, 11.37223303,  9.24073609,

9.26123534,
7.30210992, 8.18668614, 10.01057952, 8.16795417,

10.85423587,
       7.67387938, 10.27239999, 11.84077286,  8.26251054,

11.87670125,
15.55387759, 8.61521603, 11.54492611, 8.36432299,

6.61923449,
       8.89563542, 13.16830017, 11.16941883,  9.72452574,
13.57075134,
      10.31189721])

rg.coef_

array([-0.35348315, -0.6483044 , 0.30214801, 1.01384178, 0.86876941,

      -1.37261901, -0.17689802,  1.75483723])
metrics.r2_score(y_test,rg_pred)
0.45466391197604406
np.sqrt(mean_squared_error(y_test,rg_pred))
2.4043343069157084
```