1)Loading Dataset into tool

```
from google.colab import files
uploaded = files.upload()
```

```
<IPython.core.display.HTML object>
```

```
Saving abalone.csv to abalone.csv
```

```
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")
```

2)Performing Visualization

Univariate Analysis, Bi-Variate Analysis, Multi-Variate Analysis

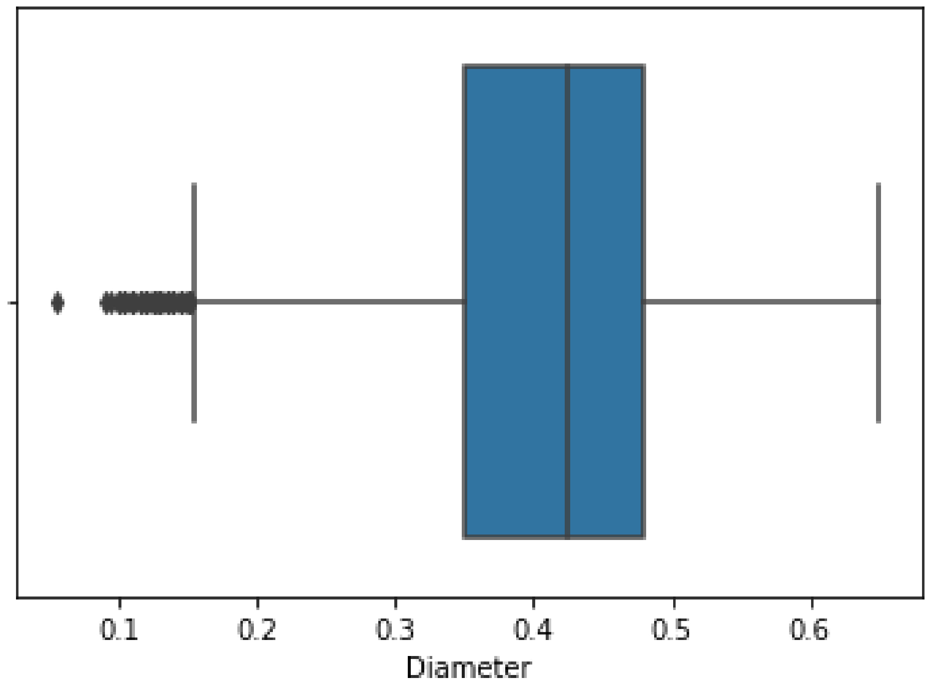
```
data.head()
```

```
  Sex  Length  Diameter  Height  Whole weight  Shucked weight  Viscera 
weight  \
0   M   0.455     0.365   0.095        0.5140          0.2245 
0.1010   
1   M   0.350     0.265   0.090        0.2255          0.0995 
0.0485   
2   F   0.530     0.420   0.135        0.6770          0.2565 
0.1415   
3   M   0.440     0.365   0.125        0.5160          0.2155 
0.1140   
4   I   0.330     0.255   0.080        0.2050          0.0895 
0.0395   

   Shell weight  Rings  
0         0.150     15  
1         0.070      7  
2         0.210      9  
3         0.155     10  
4         0.055      7  
```

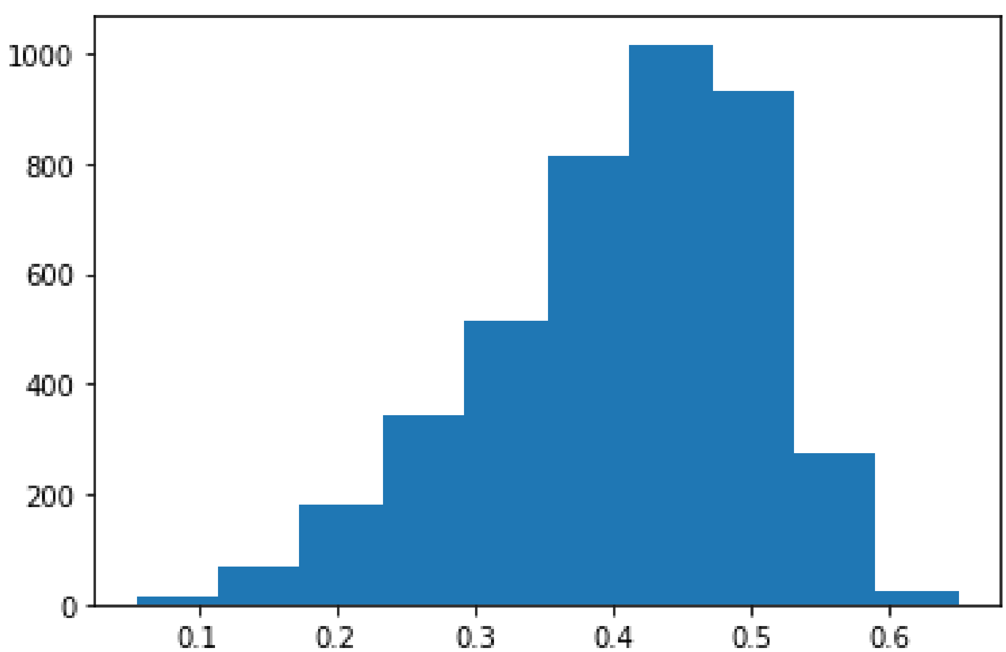
```
sns.boxplot(data['Diameter'])
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fe987eb58d0>
```



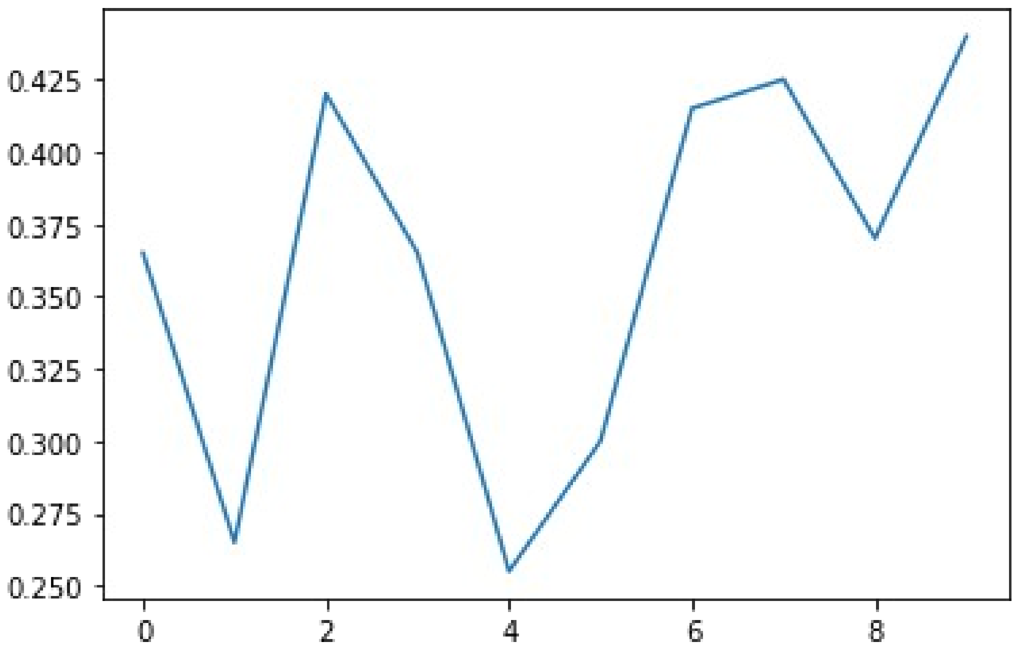

```
plt.hist(data['Diameter'])
```

```
(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>)
```



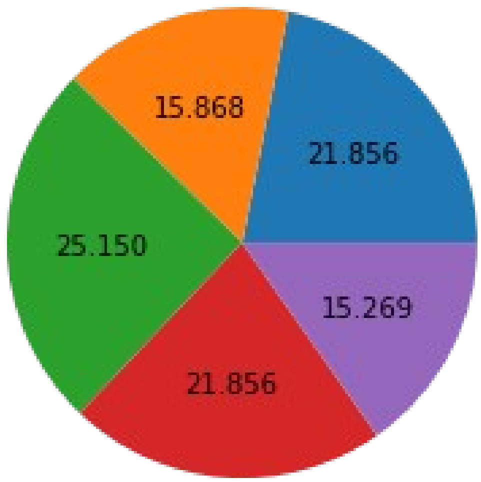
```
plt.plot(data['Diameter'].head(10))
```

```
[<matplotlib.lines.Line2D at 0x7fe987978510>]
```



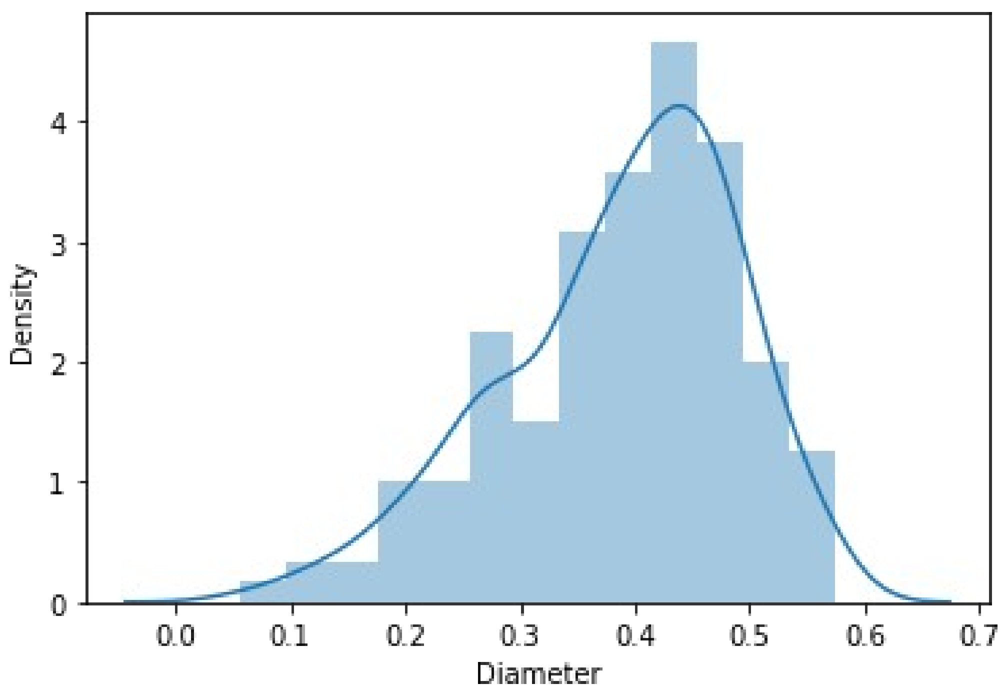

```
plt.pie(data['Diameter'].head(),autopct='%.3f')
```

```
([<matplotlib.patches.Wedge at 0x7fe98788cb50>,
  <matplotlib.patches.Wedge at 0x7fe987897350>,
  <matplotlib.patches.Wedge at 0x7fe987897bd0>,
  <matplotlib.patches.Wedge at 0x7fe9878a2510>,
  <matplotlib.patches.Wedge at 0x7fe98782d090>],
 [Text(0.8507215626110557, 0.6973326486753676, ''),
  Text(-0.32611344931648134, 1.0505474849691026, ''),
  Text(-1.0998053664078908, -0.020691931287471 44, ''),
  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')])
```



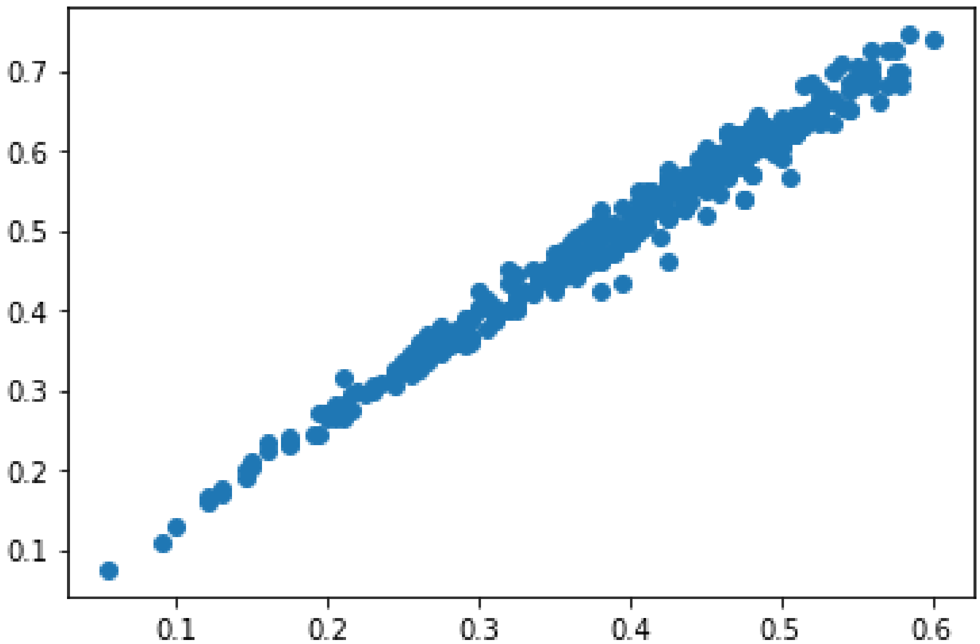

```
sns.distplot(data['Diameter'].head(300))
```

<matplotlib.axes._subplots.AxesSubplot at 0x7fe9878450d0>



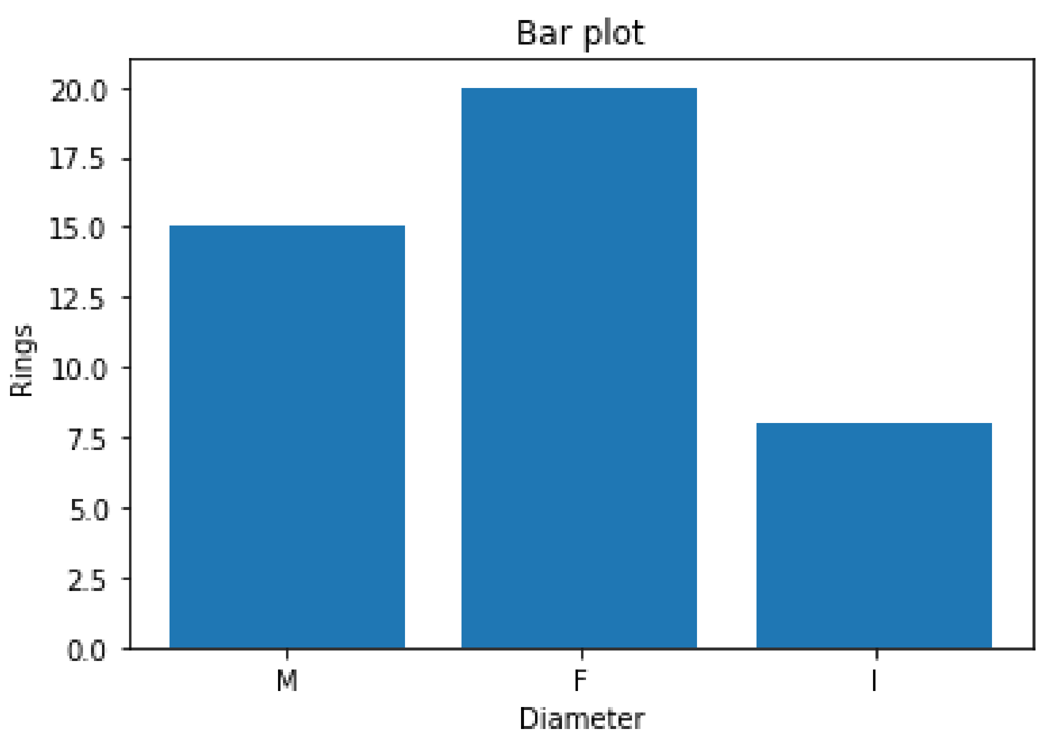

```
plt.scatter(data['Diameter'].head(400),data['Length'].head(400))
```

<matplotlib.collections.PathCollection at 0x7fe987732250>



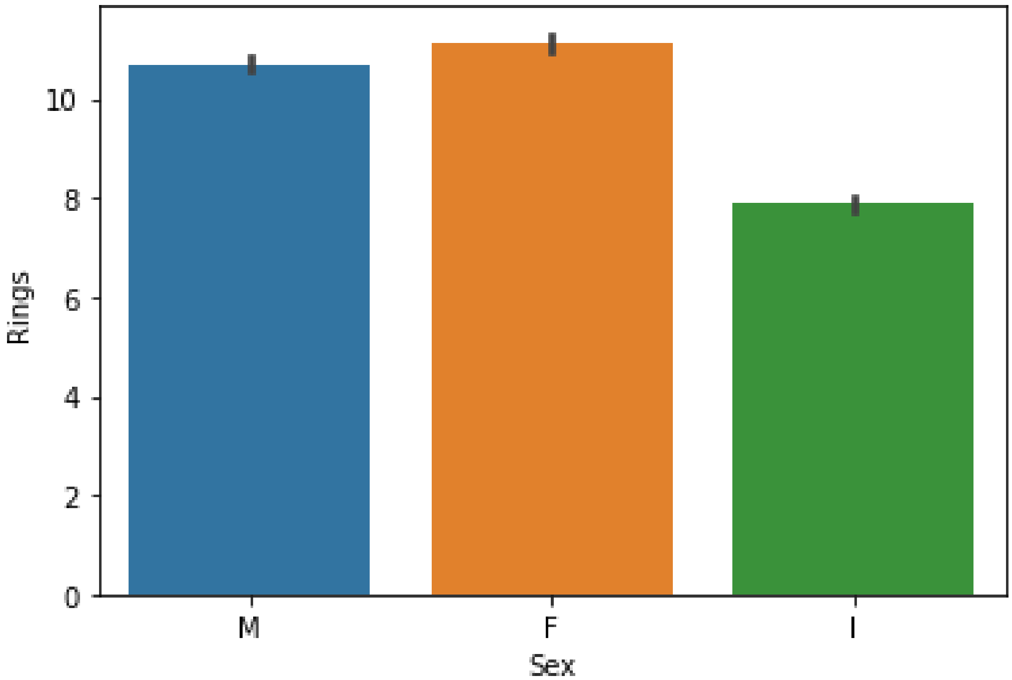

```
plt.bar(data['Sex'].head(20),data['Rings'].head(20))
plt.title('Bar plot')
plt.xlabel('Diameter')
plt.ylabel('Rings')
```

```
Text(0, 0.5, 'Rings')
```



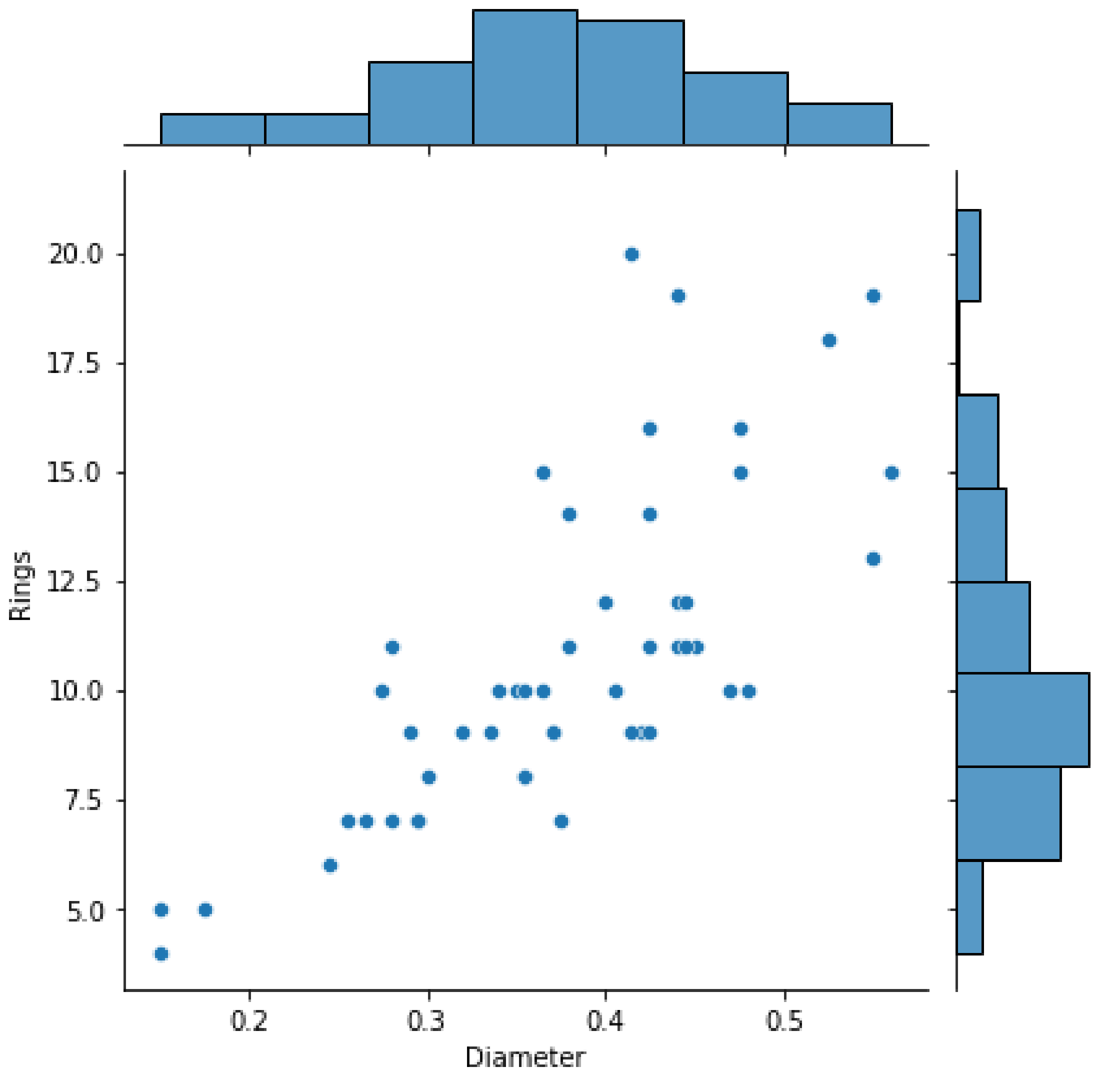
```
sns.barplot(data['Sex'], data['Rings'])
```

<matplotlib.axes._subplots.AxesSubplot at 0x7fe987629750>



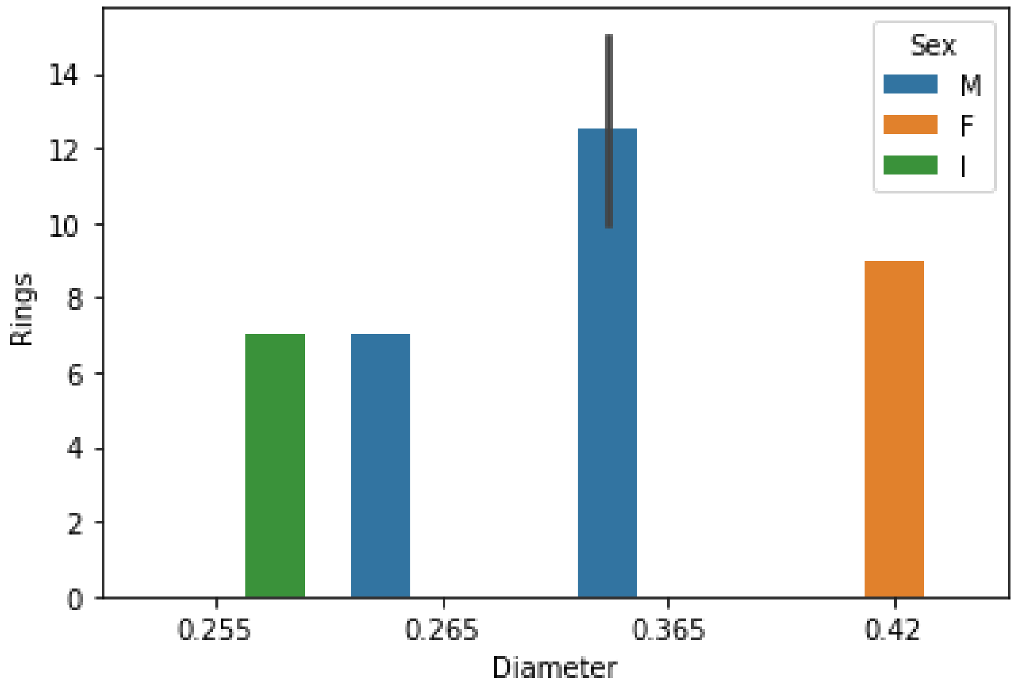

```
sns.jointplot(data['Diameter'].head(50),data['Rings'].head(100))
```

<seaborn.axisgrid.JointGrid at 0x7fe987621210>



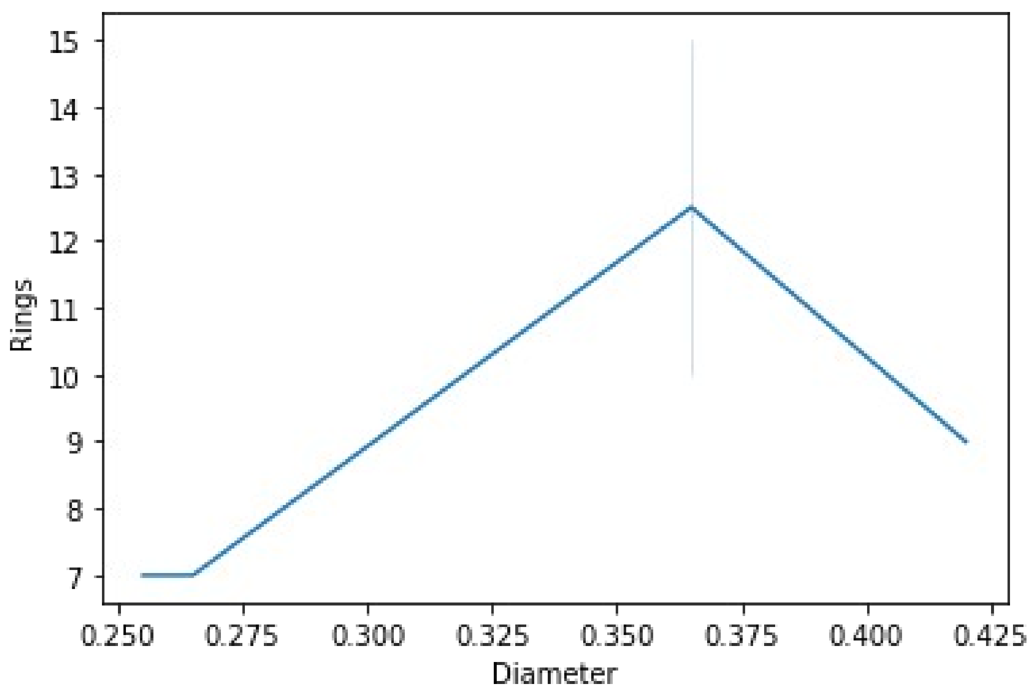

```
sns.barplot('Diameter','Rings',hue='Sex',data=data.head())
```

<matplotlib.axes._subplots.AxesSubplot at 0x7fe987622ad0>



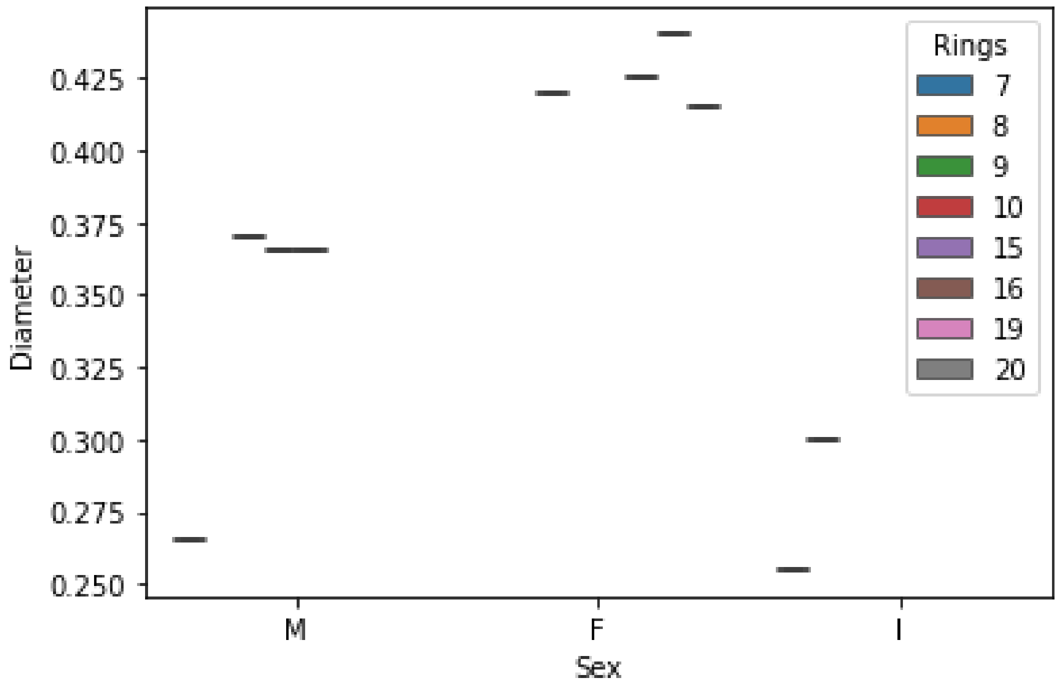

```
sns.lineplot(data['Diameter'].head(),data['Rings'].head())
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fe984bab810>
```



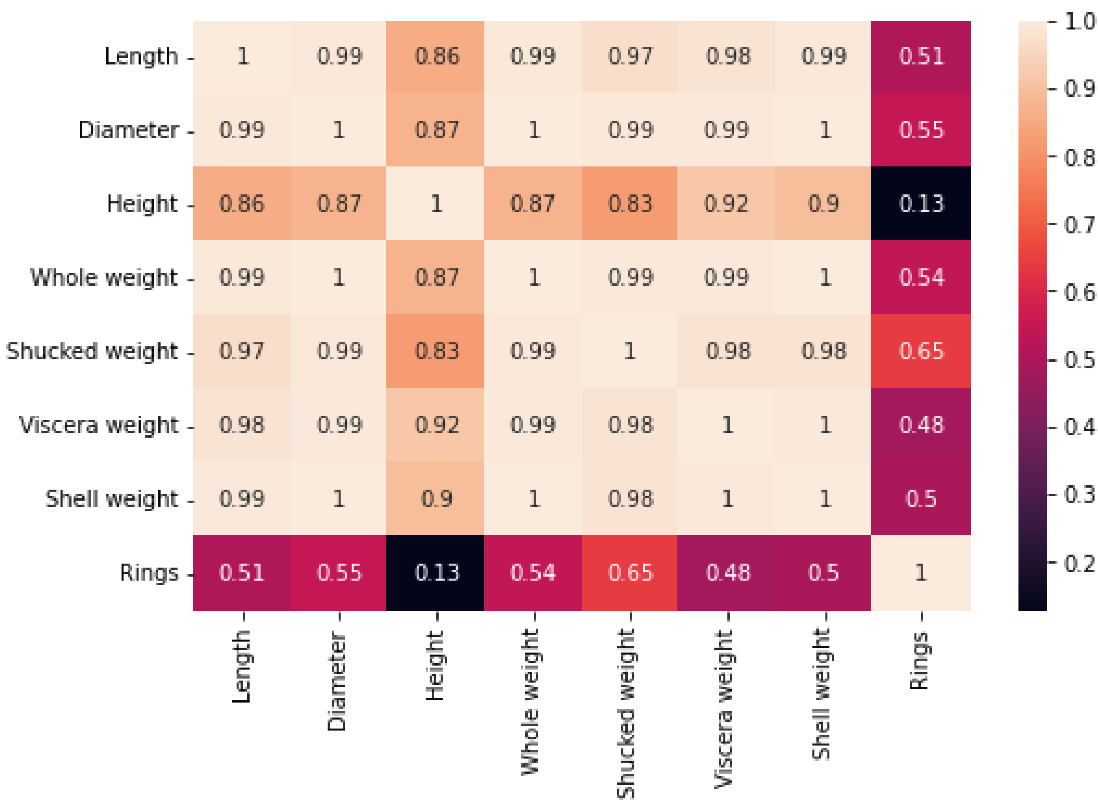

```
sns.boxplot(data['Sex'].head(10),data['Diameter'].head(10),data['Rings
'].head(10))
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fe984b57110>
```



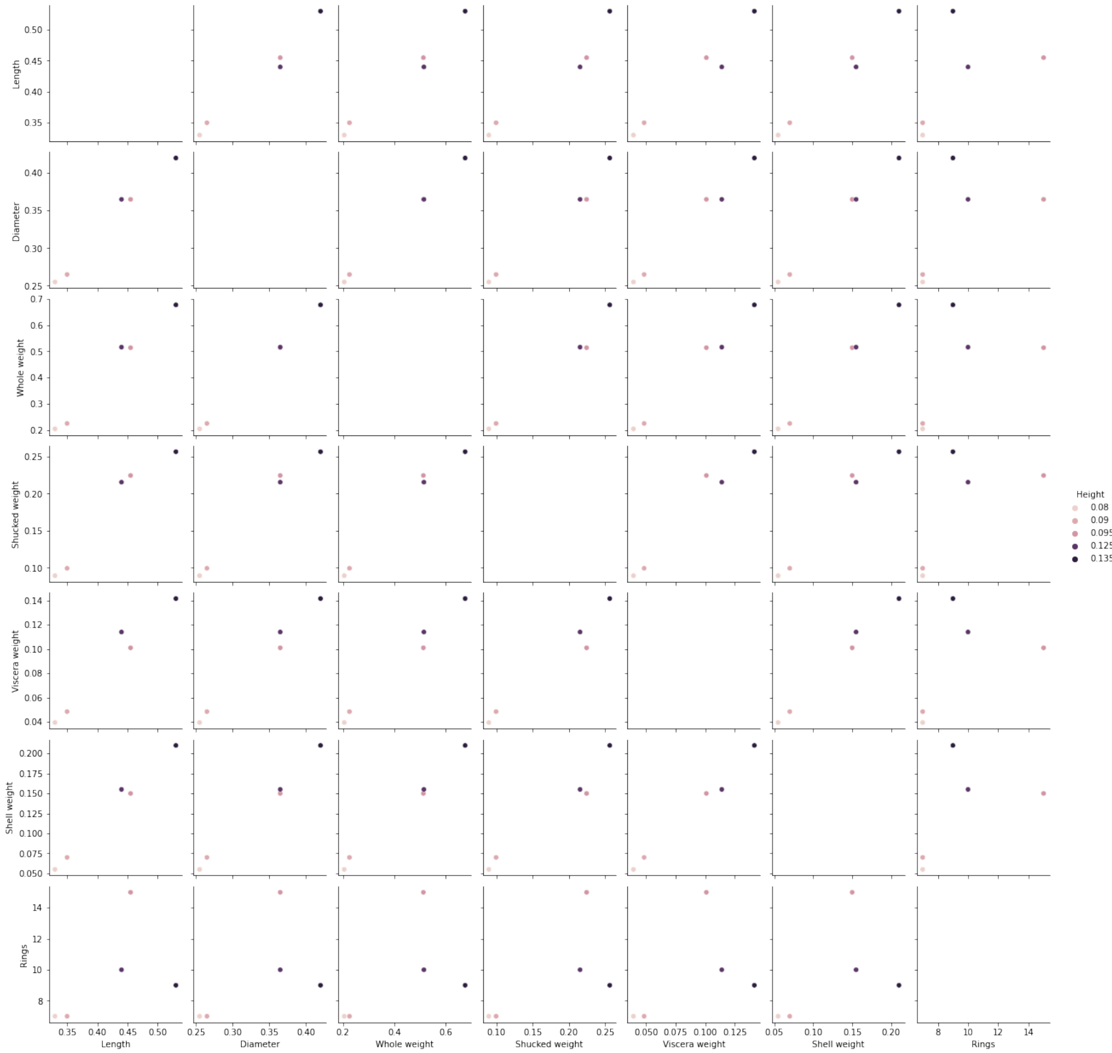

```
fig=plt.figure(figsize=(8,5))
sns.heatmap(data.head().corr(),annot=True)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fe984a004d0>
```



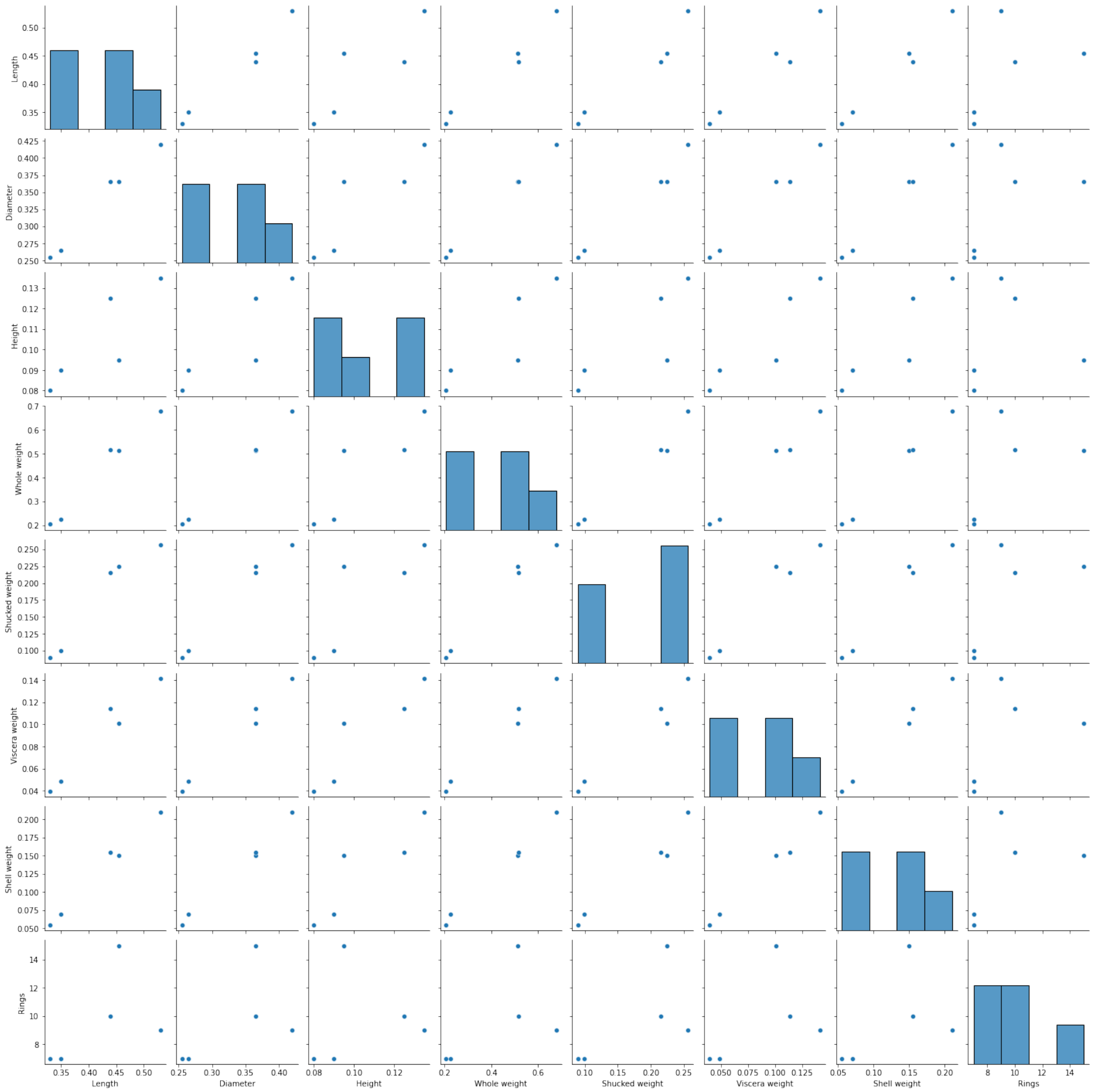
```
sns.pairplot(data.head(),hue='Height')
```

<seaborn.axisgrid.PairGrid at 0x7fe98488d410>




```
sns.pairplot(data.head())
```

<seaborn.axisgrid.PairGrid at 0x7fe983453c90>




3)Perform Descriptive Statistics on the dataset

```
data.head()

  Sex  Length  Diameter  Height  Whole weight  Shucked weight  Viscera
weight  \
0   M   0.455     0.365   0.095        0.5140          0.2245
0.1010
1   M   0.350     0.265   0.090        0.2255          0.0995
0.0485
2   F   0.530     0.420   0.135        0.6770          0.2565
0.1415
3   M   0.440     0.365   0.125        0.5160          0.2155
0.1140
4   I   0.330     0.255   0.080        0.2050          0.0895
0.0395
```

```
   Shell weight  Rings
0         0.150     15
1         0.070      7
2         0.210      9
3         0.155     10
4         0.055      7
```

```
data.tail()
```

```
      Sex  Length  Diameter  Height  Whole weight  Shucked weight  \
4172   F   0.565     0.450   0.165        0.8870          0.3700
4173   M   0.590     0.440   0.135        0.9660          0.4390
4174   M   0.600     0.475   0.205        1.1760          0.5255
4175   F   0.625     0.485   0.150        1.0945          0.5310
4176   M   0.710     0.555   0.195        1.9485          0.9455

      Viscera weight  Shell weight  Rings
4172          0.2390        0.2490     11
4173          0.2145        0.2605     10
4174          0.2875        0.3080      9
4175          0.2610        0.2960     10
4176          0.3765        0.4950     12
```

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4177 entries, 0 to 4176
Data columns (total 9 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   Sex             4177 non-null   object
 1   Length          4177 non-null   float64
 2   Diameter        4177 non-null   float64
 3   Height          4177 non-null   float64
 4   Whole weight    4177 non-null   float64
 5   Shucked weight  4177 non-null   float64
 6   Viscera weight  4177 non-null   float64
 7   Shell weight    4177 non-null   float64
 8   Rings           4177 non-null   int64
dtypes: float64(7), int64(1), object(1)
memory usage: 293.8+ KB
```

```
data.describe()
```

```
            Length     Diameter       Height  Whole weight  Shucked
weight  \
count  4177.000000  4177.000000  4177.000000   4177.000000
4177.000000
mean      0.523992     0.407881     0.139516      0.828742
0.359367
std       0.120093     0.099240     0.041827      0.490389
```

```
0.221963
min       0.075000     0.055000     0.000000      0.002000
0.001000
25%       0.450000     0.350000     0.115000      0.441500
0.186000
50%       0.545000     0.425000     0.140000      0.799500
0.336000
75%       0.615000     0.480000     0.165000      1.153000
0.502000
max       0.815000     0.650000     1.130000      2.825500
1.488000

       Viscera weight  Shell weight        Rings
count     4177.000000   4177.000000  4177.000000
mean         0.180594      0.238831     9.933684
std          0.109614      0.139203     3.224169
min          0.000500      0.001500     1.000000
25%          0.093500      0.130000     8.000000
50%          0.171000      0.234000     9.000000
75%          0.253000      0.329000    11.000000
max          0.760000      1.005000    29.000000
```

```
data.mode().T
```

```
                     0      1
Sex                  M    NaN
Length            0.55  0.625
Diameter          0.45    NaN
Height            0.15    NaN
Whole weight    0.2225    NaN
Shucked weight   0.175    NaN
Viscera weight  0.1715    NaN
Shell weight     0.275    NaN
Rings              9.0    NaN
```

```
data.shape
```

```
(4177, 9)
```

```
data.kurt()
```

```
Length             0.064621
Diameter          -0.045476
Height            76.025509
Whole weight      -0.023644
Shucked weight     0.595124
Viscera weight     0.084012
Shell weight       0.531926
Rings              2.330687
dtype: float64
```

```
data.skew()
```

```
Length           -0.639873
Diameter         -0.609198
Height            3.128817
Whole weight      0.530959
Shucked weight    0.719098
Viscera weight    0.591852
Shell weight      0.620927
Rings             1.114102
dtype: float64
```

```
data.var()
```

```
Length             0.014422
Diameter           0.009849
Height             0.001750
Whole weight       0.240481
Shucked weight     0.049268
Viscera weight     0.012015
Shell weight       0.019377
Rings             10.395266
dtype: float64
```

```
data.nunique()
```

```
Sex                  3
Length             134
Diameter           111
Height              51
Whole weight      2429
Shucked weight    1515
Viscera weight     880
Shell weight       926
Rings               28
dtype: int64
```

4)Check for missing values and deal with them

```
data.isna()
```

```
        Sex  Length  Diameter  Height  Whole weight  Shucked weight  \
0     False   False     False   False         False           False
1     False   False     False   False         False           False
2     False   False     False   False         False           False
3     False   False     False   False         False           False
4     False   False     False   False         False           False
...     ...     ...       ...     ...           ...             ...
4172  False   False     False   False         False           False
4173  False   False     False   False         False           False
4174  False   False     False   False         False           False
4175  False   False     False   False         False           False
4176  False   False     False   False         False           False
```

```
      Viscera weight  Shell weight  Rings
0              False         False  False
1              False         False  False
2              False         False  False
3              False         False  False
4              False         False  False
...              ...           ...    ...
4172           False         False  False
4173           False         False  False
4174           False         False  False
4175           False         False  False
4176           False         False  False

[4177 rows x 9 columns]
```

```
data.isna().any()
```

```
Sex               False
Length            False
Diameter          False
Height            False
Whole weight      False
Shucked weight    False
Viscera weight    False
Shell weight      False
Rings             False
dtype: bool
```

```
data.isna().sum()
```

```
Sex               0
Length            0
Diameter          0
Height            0
Whole weight      0
Shucked weight    0
Viscera weight    0
Shell weight      0
Rings             0
dtype: int64
```

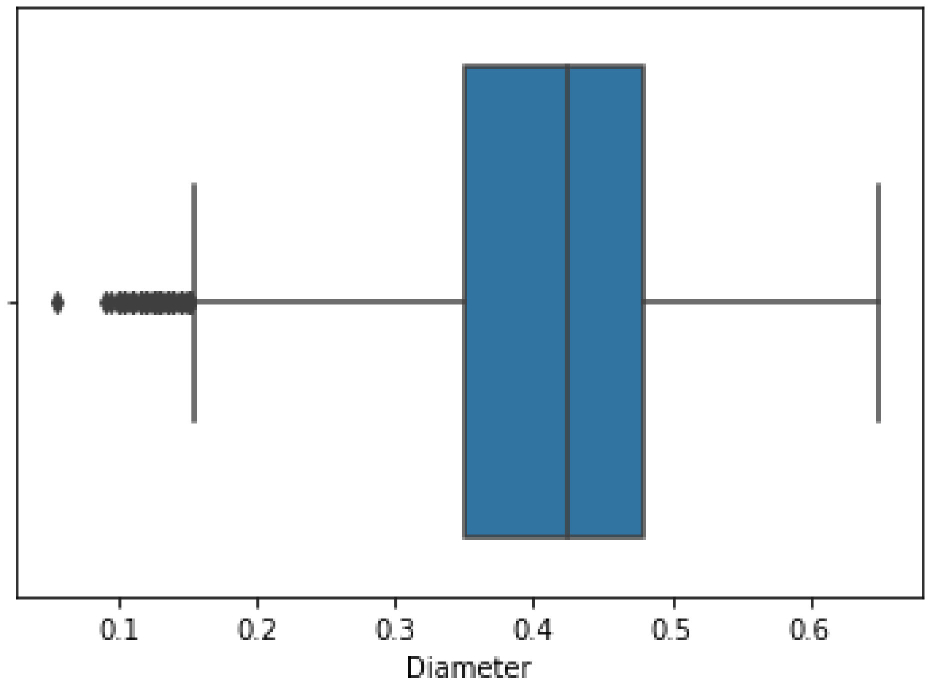
```
data.isna().any().sum()
```

```
0
```

5)Find the outliers and replace them outliers

```
sns.boxplot(data['Diameter'])
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fe9802a5490>
```




```
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           0.186
0.0935
0.75   0.615      0.48   0.165        1.1530           0.502
0.2530

      Shell weight  Rings
0.25         0.130    8.0
0.75         0.329   11.0
```

```
iqr=quant.loc[0.75]-quant.loc[0.25]
iqr
```

```
Length            0.1650
Diameter          0.1300
Height            0.0500
Whole weight      0.7115
Shucked weight    0.3160
Viscera weight    0.1595
Shell weight      0.1990
Rings             3.0000
dtype: float64
```

```
low=quant.loc[0.25]-(1.5*iqr)
low
```

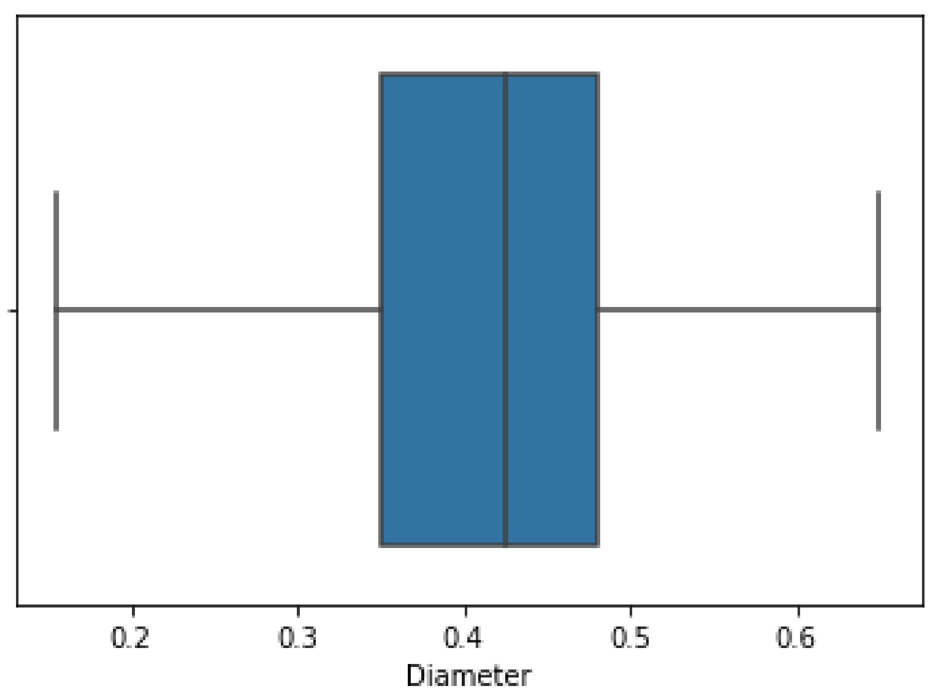
```
Length            0.20250
Diameter          0.15500
Height            0.04000
Whole weight     -0.62575
Shucked weight   -0.28800
Viscera weight   -0.14575
Shell weight     -0.16850
Rings             3.50000
dtype: float64
```

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

```
Length             0.86250
Diameter           0.67500
Height             0.24000
Whole weight       2.22025
Shucked weight     0.97600
Viscera weight     0.49225
Shell weight       0.62750
Rings             15.50000
dtype: float64
```

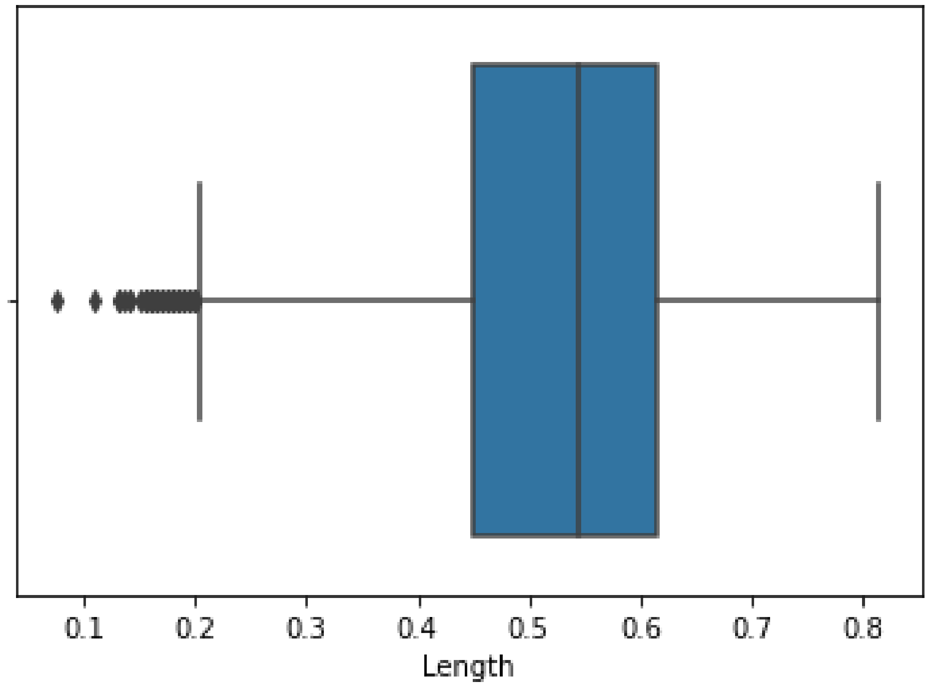
```
data['Diameter']=np.where(data['Diameter']<0.155,0.4078,data['Diameter
'])
sns.boxplot(data['Diameter'])
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fe980081110>
```



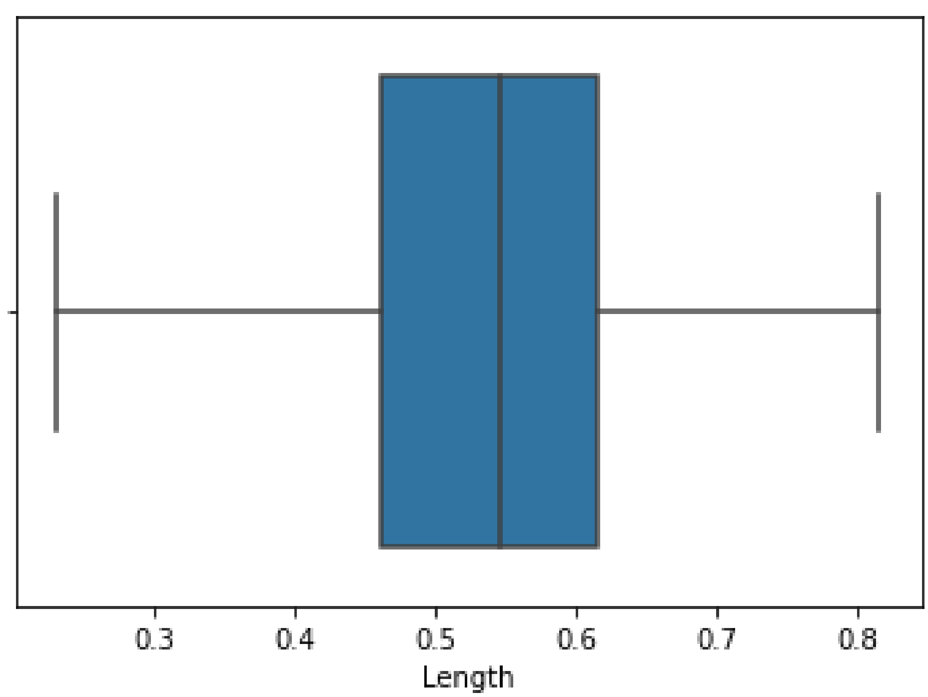

```
sns.boxplot(data['Length'])
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fe98005f8d0>
```



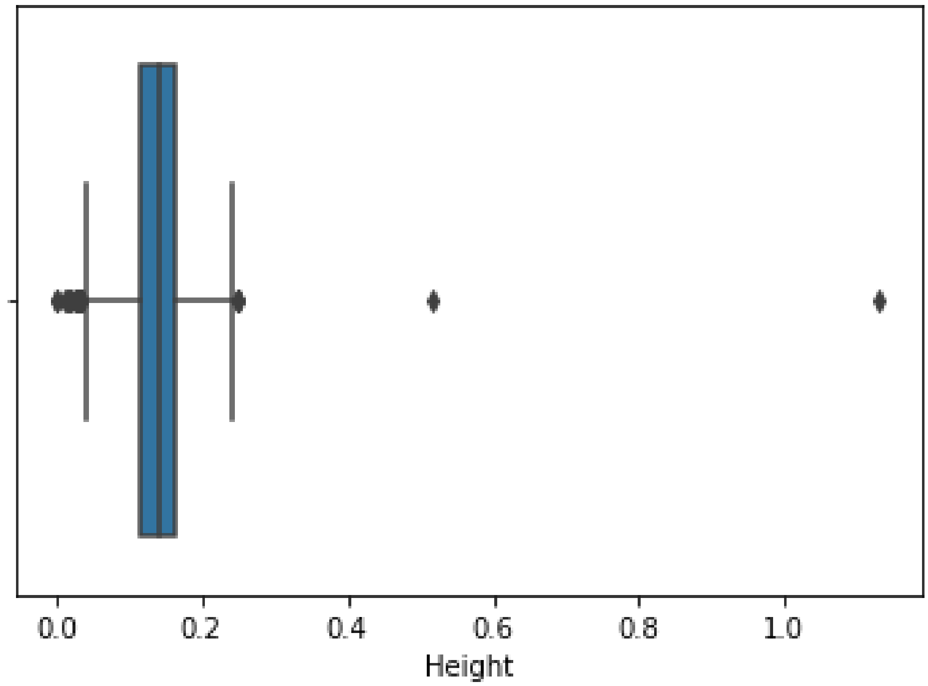

```
data['Length']=np.where(data['Length']<0.23,0.52, data['Length'])
sns.boxplot(data['Length'])
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fe97ffc43d0>
```



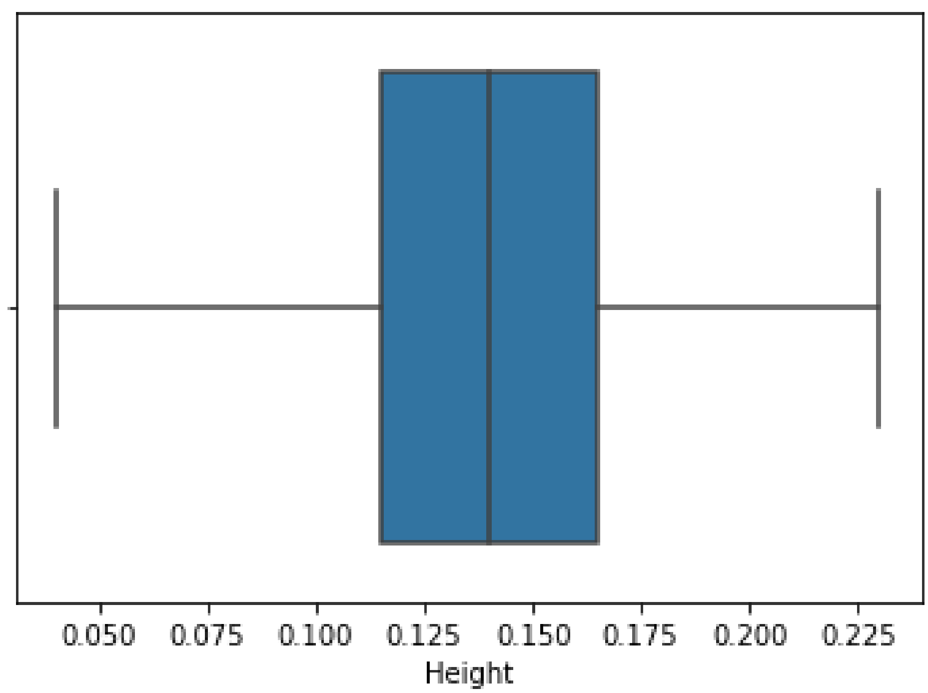

```
sns.boxplot(data['Height'])
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fe97ff3c6d0>
```



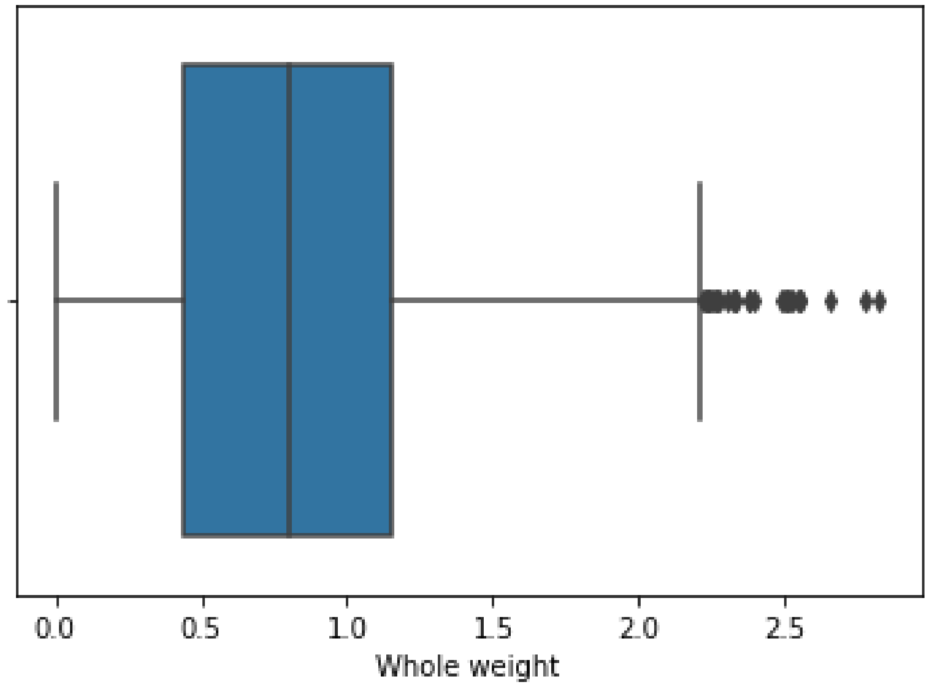

```
data['Height']=np.where(data['Height']<0.04,0.139, data['Height'])
data['Height']=np.where(data['Height']>0.23,0.139, data['Height'])
sns.boxplot(data['Height'])
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fe97fea80d0>
```



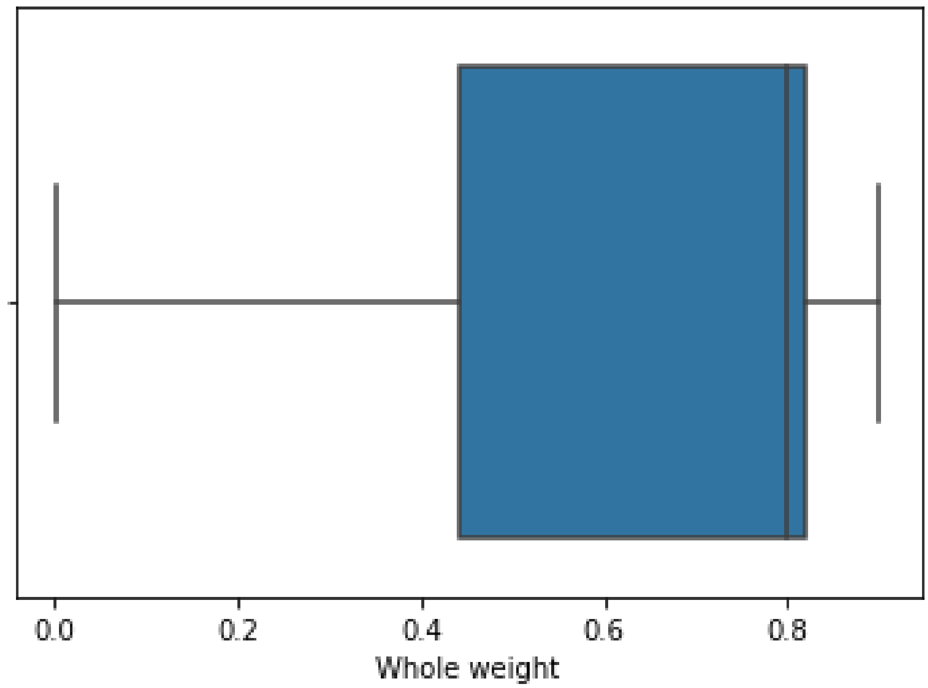
```
sns.boxplot(data['Whole weight'])
```

<matplotlib.axes._subplots.AxesSubplot at 0x7fe97fe74890>



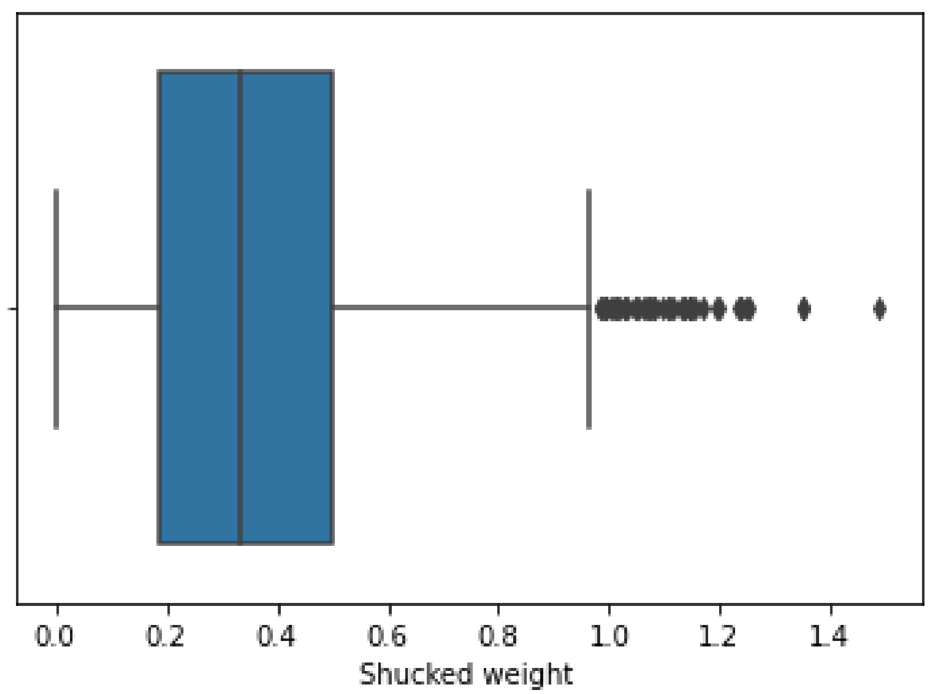

```
data['Whole weight']=np.where(data['Whole weight']>0.9,0.82,
data['Whole weight'])
sns.boxplot(data['Whole weight'])
```

<matplotlib.axes._subplots.AxesSubplot at 0x7fe97fe0e710>



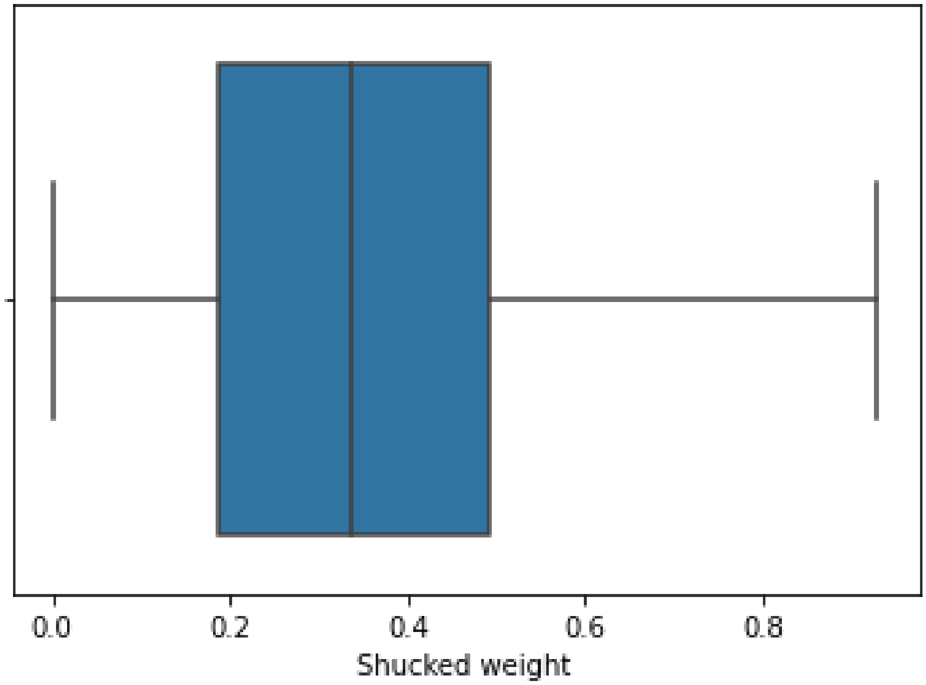

```
sns.boxplot(data['Shucked weight'])
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fe980853850>
```



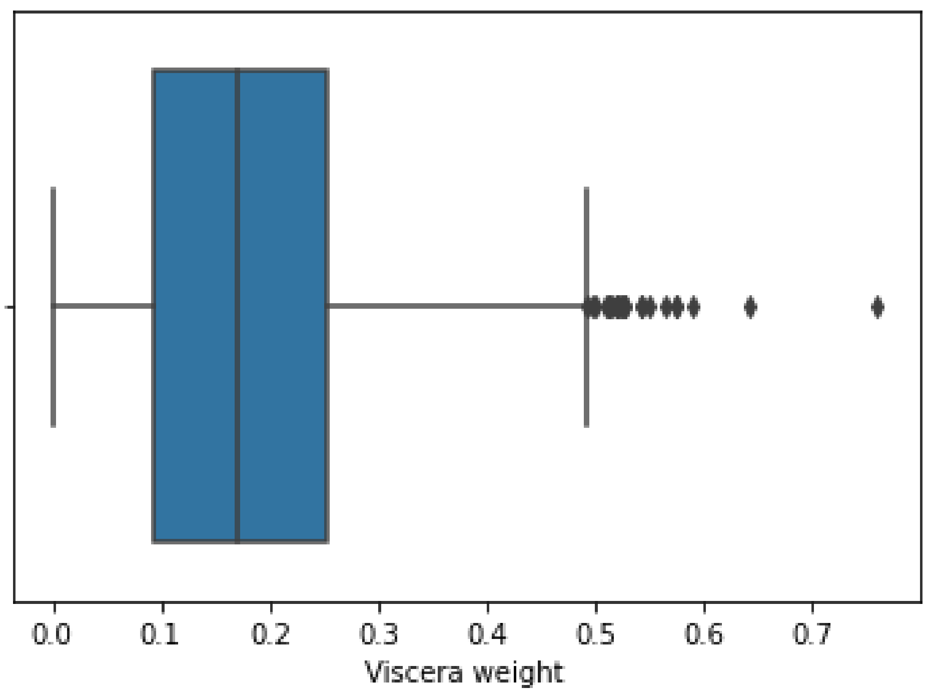

```
data['Shucked weight']=np.where(data['Shucked weight']>0.93,0.35,
data['Shucked weight'])
sns.boxplot(data['Shucked weight'])
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fe97fce5850>
```



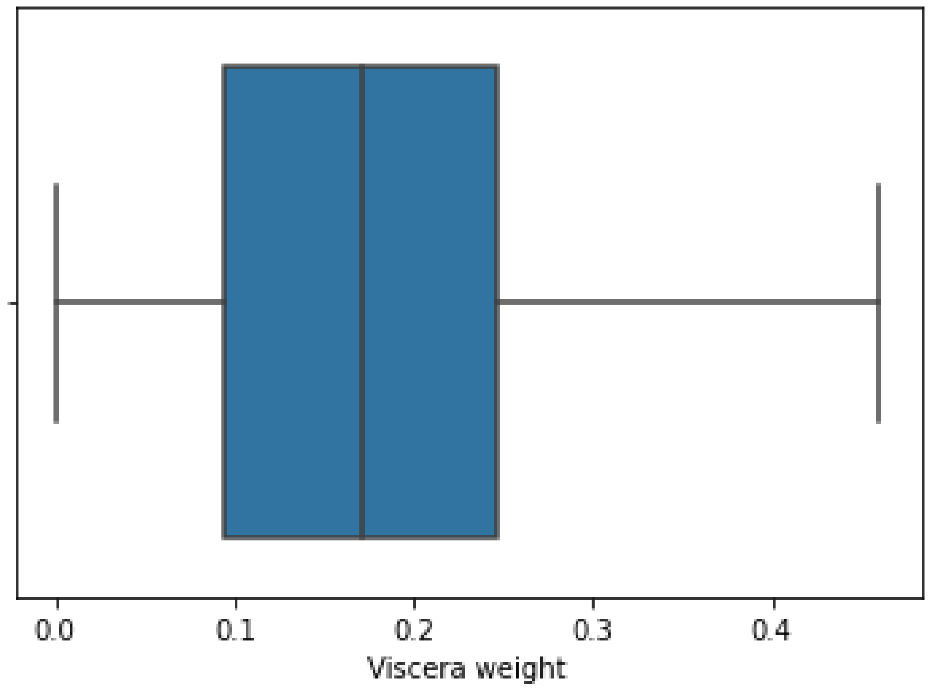

```
sns.boxplot(data['Viscera weight'])
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fe97fcc0f50>
```



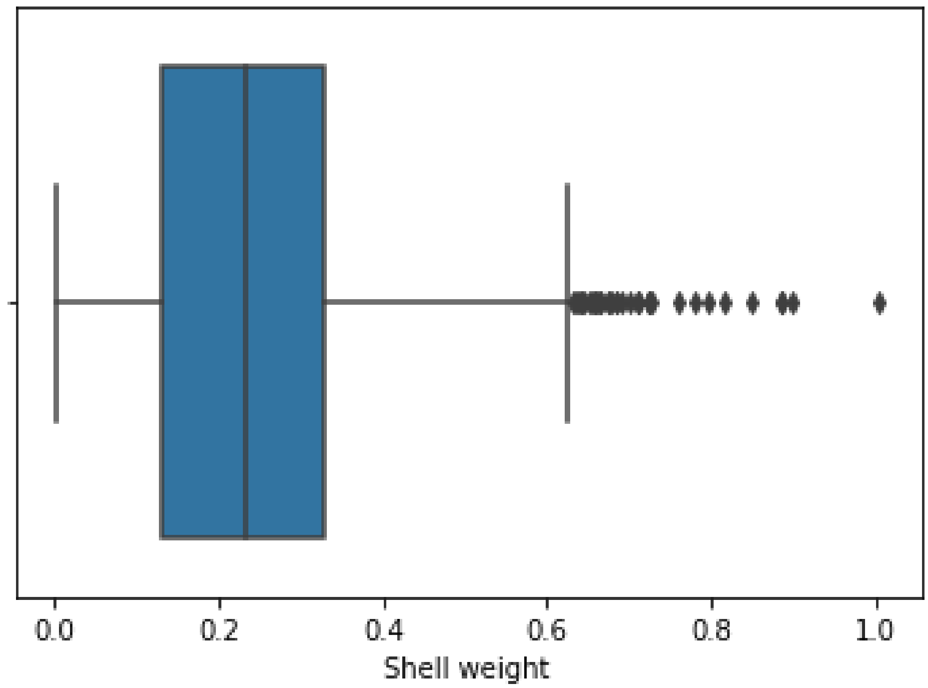
```
data['Viscera weight']=np.where(data['Viscera weight']>0.46,0.18,
data['Viscera weight'])
sns.boxplot(data['Viscera weight'])
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fe97fc38ad0>
```



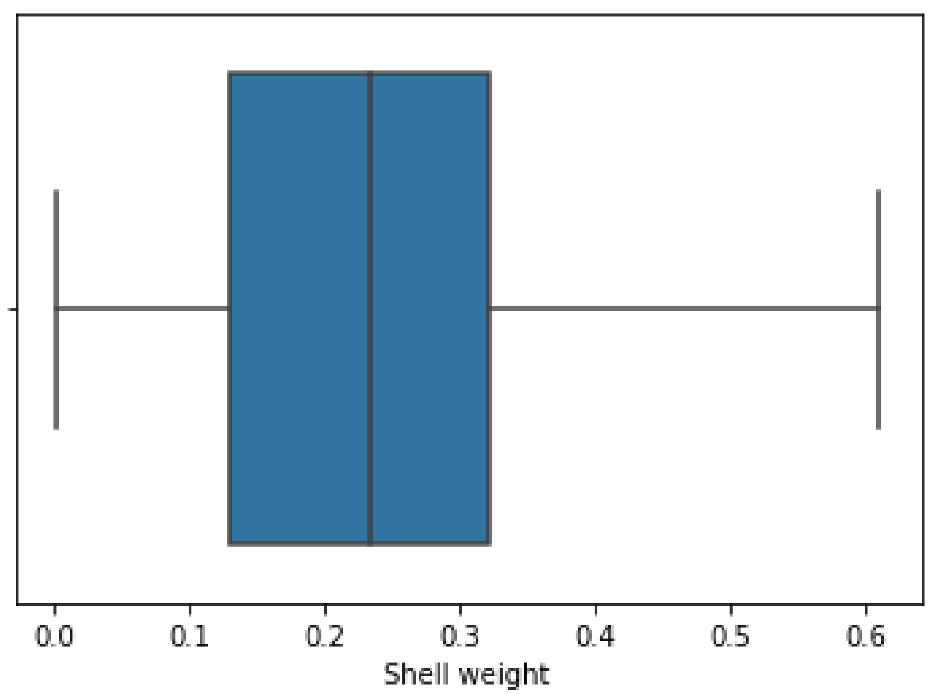

```
sns.boxplot(data['Shell weight'])
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fe97fc17a10>
```




```
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 0x7fe97fb81290>
```




6)Check for Categorical columns and perform encoding

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

```
      Sex  Length  Diameter  Height  Whole weight  Shucked weight  \
0       1   0.455     0.365   0.095        0.5140          0.2245
1       1   0.350     0.265   0.090        0.2255          0.0995
2       0   0.530     0.420   0.135        0.6770          0.2565
3       1   0.440     0.365   0.125        0.5160          0.2155
4       2   0.330     0.255   0.080        0.2050          0.0895
...   ...     ...       ...     ...           ...             ...
4172    0   0.565     0.450   0.165        0.8870          0.3700
4173    1   0.590     0.440   0.135        0.8200          0.4390
4174    1   0.600     0.475   0.205        0.8200          0.5255
4175    0   0.625     0.485   0.150        0.8200          0.5310
4176    1   0.710     0.555   0.195        0.8200          0.3500

      Viscera weight  Shell weight  Rings
0             0.1010        0.1500     15
1             0.0485        0.0700      7
2             0.1415        0.2100      9
3             0.1140        0.1550     10
4             0.0395        0.0550      7
...              ...           ...    ...
4172          0.2390        0.2490     11
4173          0.2145        0.2605     10
4174          0.2875        0.3080      9
4175          0.2610        0.2960     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  Whole weight  Shucked weight  \
0       1   0.455     0.365   0.095        0.5140          0.2245
1       1   0.350     0.265   0.090        0.2255          0.0995
2       0   0.530     0.420   0.135        0.6770          0.2565
3       1   0.440     0.365   0.125        0.5160          0.2155
4       2   0.330     0.255   0.080        0.2050          0.0895
...   ...     ...       ...     ...           ...             ...
4172    0   0.565     0.450   0.165        0.8870          0.3700
4173    1   0.590     0.440   0.135        0.8200          0.4390
4174    1   0.600     0.475   0.205        0.8200          0.5255
4175    0   0.625     0.485   0.150        0.8200          0.5310
4176    1   0.710     0.555   0.195        0.8200          0.3500

      Viscera weight  Shell weight
```

```
0             0.1010       0.1500
1             0.0485       0.0700
2             0.1415       0.2100
3             0.1140       0.1550
4             0.0395       0.0550
...              ...          ...
4172          0.2390       0.2490
4173          0.2145       0.2605
4174          0.2875       0.3080
4175          0.2610       0.2960
4176          0.3765       0.4950

[4177 rows x 8 columns]
```

```
y
```

```
0       15
1        7
2        9
3       10
4        7
        ..
4172    11
4173    10
4174     9
4175    10
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,  0.08738942, ..., -0.45300269,
        -0.33890749, -0.18321163],
       ...,
       [-0.0105225 ,  0.63117159,  0.67657577, ...,  0.86994729,
         1.08111018,  0.56873549],
       [-1.26630752,  0.85566483,  0.78370057, ...,  0.89699645,
         0.82336724,  0.47666033],
       [-0.0105225 ,  1.61894185,  1.53357412, ...,  0.00683308,
         1.94673739,  2.00357336]])
```

9)Split the data into training and testing

```
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([ 8.46487717,  6.85421951,  8.95625639,  9.14260428,
5.43454931,
       11.63142635,  9.78344305, 10.53287674,  9.69136   ,
9.56996713,
        8.02067572,  7.29056853,  7.21812634,  9.64532915,
8.90042893,
        8.61853752, 13.18664105,  9.63634027, 13.19697259,
12.33079685,
        7.12952139, 11.26637332,  9.40366388,  8.48062755,
10.56127866,
        7.88958627, 11.55052892,  9.58596809, 10.99926317,
10.783467  ,
       13.14570738,  7.53830403,  9.10730511,  9.23125992,
10.48229447,
        6.52779558, 10.13972137, 10.81104503, 12.38683084,
9.93455799,
       11.60413722, 10.34459185, 12.9264024 , 11.77611575,
8.54658576,
       11.18007798,  8.14339786, 10.67384382, 12.15622333,
7.17544099,
        7.16163277, 11.61472889,  9.51106204,  9.29911717,
7.14764444,
        7.57073367, 10.7237167 , 13.42254621, 10.262429  ,
11.91964081,
        9.07592109, 13.02194436,  6.45058692, 10.01267595,
11.49765613,
        7.41430973,  6.93209817, 11.35107298, 10.17515673,
9.78720055,
       10.51859254, 10.05921705, 13.58764674, 12.8102668 ,
```

```
9.86523619,
        10.34013227,  6.0609183 ,  6.0357537 ,  8.44368488,
10.4940275 ,
         7.89451441, 10.66006869,  6.91531125,  8.03673286,
3.50084352,
         8.13125373,  6.10647587,  9.00410788,  8.46363802,
15.50087024,
         7.28951679, 11.82805296,  9.28897402, 13.24664801,
11.2696987 ,
         7.57128004, 11.11340171, 11.40192884,  7.89076616,
4.18879008,
        10.24188054, 10.20915666,  9.89380793,  8.22127719,
10.83531754,
         8.91974918,  6.85671231, 11.36531432, 13.87649217,
10.67946672,
         7.69926618, 11.80696945, 10.67381983,  9.80760516,
7.77939824,
        11.6414938 ,  6.90835513, 10.79391878,  8.70140363,
9.56577457,
         7.20841861, 10.68570947, 10.44427932,  9.30025576,
11.17821246,
        11.05640552,  9.42633124,  6.90395177, 12.44933464,
6.61712181,
        11.26109344, 11.87995526, 10.44179912, 10.86149639,
7.27480096,
         9.18993323,  8.00751678, 11.03500107,  7.16086218,
7.3781279 ,
         9.97987697, 13.08503607,  8.88472108, 12.01309057,
11.25185163,
        12.62230109,  9.5771331 ,  6.64638774,  6.61454727,
13.76117117,
        12.58901257, 11.34307923,  9.96256615, 12.63021511,
7.71765795,
         7.01468052, 10.75517803, 11.44329807, 11.50602651,
10.03515615,
         9.05796967,  9.48807525,  5.85396862, 10.07137588,
7.04055711,
         7.20455189,  7.86553059,  8.82228567,  8.83548405,
10.20466389,
         9.87013976, 11.57861856, 10.70409019, 11.23589087,
9.6801916 ,
        14.73772844, 14.03134515, 11.72436402, 12.28978571,
14.46588442,
        12.49196345, 10.7540761 , 12.12376873,  6.4106872 ,
8.79636504,
        10.87989137, 11.09003265,  8.37290084,  7.13256146,
9.81202353,
        11.61631095,  6.20520895, 13.44670973,  8.23801051,
13.63946475,
        10.89730638, 10.51160004,  9.91014401,  8.82067439,
```

```
11.28185205,
        10.9743134 , 12.40038233, 10.24754027, 10.75822942,
11.04518827,
        10.02158365, 13.1111805 , 15.40556848,  8.65424048,
11.07808045,
        11.43428819,  9.29319742,  8.50536489, 11.72407875,
11.87436472,
        10.86040756,  6.83517474, 10.6800387 , 13.68599697,
7.85141387,
         9.61500118,  8.13639366,  7.12547327, 11.37572754,
11.08124523,
         8.38975998,  8.93125248, 11.25031435,  8.92650784,
10.77453433,
         7.26404102, 11.7050228 ,  7.33967403, 12.89906297,
10.05072602,
        12.30928037,  6.4696131 ,  7.02885083,  8.17760577,
8.36722009,
        12.27788251, 14.86689854,  9.03555045,  7.89516837,
11.46042259,
        11.97284328, 10.23782299,  8.82108824, 11.66005622,
10.46672855,
         9.6020531 ,  6.64546934,  9.30679254,  7.72084752,
6.9425281 ,
        11.22613892,  6.88233109,  6.12276549,  6.77762648,
11.93365948,
         9.79963888,  9.88494407, 11.07637015, 13.71894731,
11.33173588,
        12.46722047, 10.24469564,  9.72052293, 15.72757266,
10.71541052,
        10.35583651,  9.37937965,  9.4762949 , 11.32994445,
8.9365591 ,
        11.32457772,  8.90429742,  7.20777047,  7.85802467,
12.18367637,
        10.6047639 , 11.6284761 , 12.92381795, 12.9011051 ,
10.42630298,
        10.87901235, 11.95251887,  7.59061753, 11.13136573,
11.14587304,
        10.84922924, 10.56523496,  9.23890243,  7.05324141,
12.28183223,
        12.94505248, 10.68366806,  8.93243319, 10.23301085,
11.932904  ,
         8.09927623, 11.4636168 ,  8.56271226, 10.31112106,
11.84760778,
         6.22788608,  9.93223502, 11.72134752, 10.41244677,
7.77705748,
         8.02055319, 11.04293385,  6.71705862,  8.51011777,
12.48133396,
         6.92876306,  7.42973865,  6.32702252, 10.16489746,
10.04617246,
        10.74849629, 12.21585654,  9.20699277, 14.71634501,
```

```
9.30535188,
       11.15719953, 11.91262178,  6.16915888,  5.90941624,
12.0347141 ,
       10.39279186, 10.36893882,  4.25228089, 10.8949049 ,
7.14878233,
       11.04075635,  7.02554729, 12.2645853 ,  8.73018795,
8.27768114,
       10.17189596, 11.19158236, 10.1676116 ,  6.96869979,
8.66646692,
        7.08119382, 10.21851345,  9.34915128, 10.70912124,
5.95636562,
       10.61560182,  6.14184347,  8.71865978, 10.45526229,
10.82400642,
        9.23872278, 11.05431539, 10.90597747, 10.73342796,
8.86599071,
       10.48094551,  6.64578215, 11.23987205,  8.54182669,
11.7477742 ,
        6.71124909,  8.06292216, 11.22485284, 11.06111411,
7.34683644,
       12.79722726, 11.66605496,  6.38621258,  7.50561393,
10.75047708,
        6.78151503, 10.86834383, 12.05925882,  8.78559763,
11.60744044,
       14.34061908, 10.9173401 ,  3.90759341, 10.50838795,
11.23024492,
       10.76708491, 11.86291673, 10.08910641,  9.9153805 ,
10.42164311,
       11.72066467, 10.10311211, 10.85201146, 11.36908829,
11.63659406,
        8.98471533,  8.83415033, 16.31795918,  9.31796913,
8.19565201,
        6.52699703, 14.22263752, 10.90773594, 11.50174563,
11.3565975 ,
       13.04693705, 11.29211322,  8.8315844 , 10.62510236,
11.238575  ,
        8.51948719,  7.53761462,  6.59746543,  6.57494504,
6.45554154,
        9.37175698,  9.87591829,  7.77218791, 12.13270427,
8.28461486,
        6.8227227 ,  6.34295778, 11.49187889,  6.1373754 ,
9.78206148,
       10.25429389,  9.47152687,  7.0513122 , 10.75443284,
11.6828135 ,
        8.78994912,  9.72926132, 10.2007595 ,  4.0764625 ,
7.49944418,
        6.78909037,  9.28519643,  8.97182792, 11.78003264,
12.00495987,
       10.47520576, 10.87560979, 11.50717387,  9.55398665,
10.54263078,
        8.90294382,  9.99202711,  5.99682874,  9.77640612,
```

```
9.92853352,
        13.46987401, 13.75199765, 10.90704866,  7.42916317,
8.11278768,
        10.57797517, 15.12899917, 12.50580434,  8.94643312,
10.54930716,
         8.34871064,  8.43353409,  9.96389655, 10.83401386,
12.86793656,
         9.96423822, 12.06335062,  8.20073132, 12.29903926,
9.8971067 ,
        10.889334  , 10.86391965, 12.53094373, 10.40614429,
9.65908436,
         7.70080247, 11.90677891,  6.21309604, 11.00797514,
9.23353315,
         6.85219652, 13.18209709,  8.89606296, 11.25131005,
6.74388309,
         9.8844426 , 10.55742003,  6.13770368, 11.71967505,
7.80027388,
         7.53048031, 10.89143988, 12.91848935, 11.21615902,
10.94167564,
         7.8590129 , 11.29626773,  6.58992354,  8.04666216,
12.89628633,
         6.47545134,  8.05547525,  8.84402896,  9.73481452,
12.12947479,
        10.06005104, 11.35736253, 12.48180372, 11.50302573,
9.71481999,
         8.46755936, 11.29609761, 13.09773241, 11.98082425,
7.94237565,
         9.57004131, 12.6867358 , 11.36450828,  8.59556205,
10.07636243,
        11.83410963, 12.40510674, 12.23257326, 11.49580823,
9.52101315,
        10.7704872 , 11.73495616, 13.23074956,  7.46741936,
7.63822026,
         7.9388726 ,  9.98462497,  6.56149275,  8.07170379,
9.12274171,
        12.30352806, 10.92652663, 13.60545394,  7.15723   ,
13.20890459,
         9.09116435,  9.3475761 ,  9.94177873,  8.16303445,
10.75246769,
         9.78790049,  6.98429081,  8.80232765, 11.16940043,
11.16748789,
        11.5084373 , 12.61652019, 11.05372047,  8.39387331,
7.64577561,
        10.420515  , 10.77104027, 12.08290858,  8.22931526,
9.3068693 ,
        10.42108415,  9.18882448,  8.56945727, 11.52409994,
11.9105676 ,
         7.44172696,  8.64489551, 13.61519649, 13.67846316,
11.44844897,
        10.45997264, 10.09638146, 13.07650217, 10.61149523,
```

```
10.2934964 ,
        10.04355643, 10.1712392 ,  9.3428371 , 12.36137926,
8.42267952,
        12.73481319, 10.33750503,  9.59549169,  9.96288491,
8.52552219,
        12.80784202, 10.46470069, 11.12407983,  9.95996223,
11.17528982,
        10.23597105, 12.47337336, 11.86600396,  9.77051788,
12.69633846,
        12.09500479,  9.64831927,  7.54427062,  9.87746958,
12.56478991,
        11.61214474, 12.27412411,  9.39013705,  7.59847655,
7.32063078,
         7.59449664, 12.05301138,  9.10588556,  7.43686784,
11.64896449,
         8.98271335, 13.42115241,  9.83241432,  7.25440857,
9.1331702 ,
        15.03580216, 10.22099749, 12.78257901,  6.85855406,
10.99019741,
        10.90217763,  7.16461383, 11.34640628,  9.95172489,
10.7859926 ,
        10.89836889, 11.59883603, 10.84969072, 10.24309041,
12.14539915,
         6.58941215,  8.22392344,  7.86444329,  9.14734334,
8.27381518,
         8.30313499, 10.65318114,  7.40887461, 11.8987115 ,
10.59590781,
         8.92318523, 10.98976977,  8.38086598, 10.6737679 ,
6.53178251,
        15.04243701,  6.81518326, 10.79435693, 10.58011187,
11.39212732,
         7.37676752, 12.03697754, 15.87297392, 11.75402087,
10.78101802,
         9.63780568, 14.69206828, 12.44241419, 14.61933981,
7.67794892,
         8.7577727 , 11.3715803 , 11.22491678,  9.56810183,
9.01153038,
         8.34389845, 10.58101448, 11.11574463,  7.69374216,
10.9169358 ,
        11.17006925,  9.96526871, 11.86403014,  9.41866573,
10.54833461,
         6.24015159, 10.59409596, 10.63559004,  6.05565676,
7.73194573,
        10.06452362, 10.21726828, 11.18842124, 10.02037767,
6.8983916 ,
         6.36645658,  7.75408898,  5.54503996, 10.81996234,
9.17719741,
         6.58527865, 10.0591114 , 12.01911827,  9.83127433,
8.61867397,
         6.45370204, 12.25845283, 13.51665274, 12.53696302,
```

```
8.79457681,
        11.07606283, 12.69147809, 10.98239379,  7.38463225,
5.93729039,
        10.79969392, 10.76733499,  8.2188618 , 11.34080452,
9.39954335,
        12.38636467, 10.94465334, 12.49588537, 10.48451765,
11.90063924,
         9.65495392,  8.81235724, 12.45181269, 12.44448104,
7.98632536,
        10.14324749,  8.54918695,  8.59498093, 12.02501894,
6.66842172,
        11.25150714,  9.37761116, 10.28496259, 12.63282811,
9.60359349,
         6.61052099, 11.85826594, 13.64228714, 11.3979238 ,
10.73116119,
         6.43630759,  8.50229815,  8.64302668,  8.0061383 ,
8.78968059,
         8.10061947, 13.26874059,  7.89933025, 12.5391302 ,
11.85809903,
         7.66978607, 10.4076772 ,  6.94196399,  6.6926532 ,
6.60228052,
         9.72153959, 10.78023247, 10.11756144,  7.10824012,
9.07510535,
         6.72382864,  6.59857561, 12.80909162,  3.35319772,
9.89692324,
         8.82422964,  7.49892018, 10.88961498,  8.89660793,
10.15290882,
        12.19621774, 12.71939235,  6.96827699, 11.17294877,
13.50861155,
         9.34222195, 11.85379293, 10.69797445, 13.57290659,
6.41991576,
        10.14440251, 11.62199   ,  6.89268413,  8.35365527,
13.26808405,
        11.18290372, 11.31643501, 11.3907173 , 12.85970416,
9.20403402,
        12.81107736, 11.45491475, 15.13109672, 11.10930881,
11.09411055,
        12.34799951, 10.42347022, 10.68901032, 12.14200968,
8.02281006,
        11.14300487,  9.3448431 ,  9.75775715, 13.52895961,
8.98616732,
         6.72413552,  6.43212626,  9.19474859, 11.71208431,
6.73035855,
         9.85921463, 17.52967498,  5.81169767,  9.72587011,
11.98452588,
         9.04481813,  8.32260293,  9.51284017,  7.78462408,
9.06042797,
        12.10929291, 10.34672569, 10.85365671, 10.17507904,
10.30874898,
         8.43428622,  9.63925713,  8.1219991 ,  7.90922074,
```

```
12.10038436,
        7.52645508, 11.10927621, 12.52243803,  9.85758436,
6.60845382,
        9.37357726,  8.85642695, 11.27675023, 10.41924374,
6.95911896,
       10.36045744])
```

```
pred=MLR.predict(x_train)
pred
```

```
array([12.21776113,  6.22680978, 11.39442787, ..., 11.74115565,
        7.97596711, 10.31552481])
```

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

```
0.4576702588960275
```

```
MLR.predict([[1,0.455,0.365,0.095,0.5140,0.2245,0.1010,0.150]])
```

```
array([9.9126927])
```

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))
```

```
2.3858961843763398
```

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.        ,  0.        ,  0.        ,  0.46237759,
0.18052476,
        0.        ,  0.        ,  0.8091466 ])
```

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

```
0.35506852482869433
```

```
np.sqrt(mean_squared_error(y_test,lso_pred))
```

```
2.601816228364118
```

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([ 8.49516314,  6.84069169,  9.00164784,  9.11776713,
5.41885998,
       11.64656575,  9.74827657, 10.53153907,  9.84044012,
9.5407592 ,
        7.96276712,  7.30392808,  7.27032782,  9.62029171,
8.87182748,
        8.62544042, 13.04564037,  9.64744857, 13.04401702,
12.33748184,
        7.16667845, 11.2239785 ,  9.42545469,  8.45961271,
10.62460762,
        7.86059328, 11.54021395,  9.53109125, 10.94027279,
10.81326348,
       12.97306517,  7.55940943,  9.1112011 ,  9.12818315,
10.50200052,
        6.49876184, 10.16495911, 10.72545016, 12.29896769,
10.06098695,
       11.64382558, 10.35734716, 12.85835207, 11.69086713,
8.52179953,
       11.03439784,  8.15296371, 10.64855162, 12.21512591,
7.18141472,
        7.18147173, 11.72002423,  9.54129972,  9.41805986,
7.20352821,
        7.55194779, 10.81886721, 13.32239626, 10.28729455,
11.89730699,
        9.02919295, 12.87341569,  6.3987705 , 10.12960125,
11.38814359,
        7.40985498,  6.93798509, 11.36066551, 10.09022377,
9.70007752,
       10.46951084, 10.20001108, 13.52472735, 12.64591637,
9.87011565,
       10.39096561,  6.0249064 ,  6.01246466,  8.50917022,
10.51456196,
        7.90250973, 10.6859311 ,  6.9536912 ,  8.02557367,
3.96028154,
```

```
        8.11749354,  6.06226403,  9.05528071,  8.47376851,
15.30482512,
        7.32338123, 11.76018515,  9.27888042, 13.20013485,
11.35446324,
        7.56104201, 11.17855506, 11.37224751,  7.94834902,
4.32845968,
       10.22481542, 10.26567803, 10.00548715,  8.16419069,
10.91760616,
        8.90695218,  6.8500505 , 11.37885998, 13.79792021,
10.66521543,
        7.7159157 , 11.7868206 , 10.73456838,  9.82138402,
7.83754785,
       11.66866027,  6.9328725 , 10.86553534,  8.67411915,
9.53291103,
        7.20218147, 10.73414962, 10.45026994,  9.30181368,
11.3011259 ,
       11.04516262,  9.44775074,  6.87024509, 12.45741206,
6.59390973,
       11.18698539, 11.92968643, 10.56359614, 10.84546368,
7.3006536 ,
        9.24160393,  8.06709395, 11.0042354 ,  7.18055181,
7.37817274,
        9.94911459, 13.01112576,  8.87818904, 11.96263797,
11.19448547,
       12.56948249,  9.58934909,  6.62153005,  6.57607899,
13.62991351,
       12.65911276, 11.33031441,  9.93416104, 12.59788809,
7.72573275,
        7.02319578, 10.74886404, 11.42884765, 11.45041223,
9.94874746,
        9.10074757,  9.49034977,  5.83924305, 10.03113568,
7.03403912,
        7.22949628,  7.88537341,  8.86080022,  8.9054791 ,
10.23370471,
        9.83596213, 11.50208545, 10.68779757, 11.32192687,
9.74572809,
       14.63888442, 13.95437442, 11.66853579, 12.23009157,
14.33236535,
       12.49020704, 10.7277561 , 11.9585267 ,  6.40434756,
8.83089923,
       10.88217612, 11.0088549 ,  8.3981855 ,  7.18010375,
9.8166676 ,
       11.58264775,  6.16841904, 13.27706961,  8.27948815,
13.51346988,
       10.89617573, 10.4945917 ,  9.8633294 ,  9.01628529,
11.30241403,
       10.92121497, 12.31299345, 10.39514285, 10.81035689,
11.00117985,
       10.08069311, 13.04951681, 15.25168293,  8.67752387,
11.14860861,
```

```
       11.40087117,  9.27655335,  8.46071839, 11.70876795,
11.83981158,
       10.77314864,  6.84756301, 10.73411281, 13.57498873,
7.85057274,
        9.72982366,  8.33687863,  7.07753674, 11.31589331,
11.09113634,
        8.44076635,  8.93013968, 11.15545327,  8.96245904,
10.78597217,
        7.26194253, 11.57780507,  7.350867  , 12.82813225,
10.01944788,
       12.25983259,  6.43767529,  7.06035273,  8.21084461,
8.32923234,
       12.19474261, 14.7991909 ,  9.02397484,  7.93248676,
11.34819695,
       11.87140346, 10.29886915,  8.8224516 , 11.60936866,
10.47122564,
        9.53153676,  6.64489771,  9.3786179 ,  7.7290699 ,
6.94901198,
       11.21320739,  6.88733953,  6.11047672,  6.77574642,
11.8733077 ,
        9.76874027,  9.95762107, 11.12108011, 13.51756159,
11.36837274,
       12.39926513, 10.20842333,  9.6837534 , 15.59882467,
10.79464928,
       10.35749497,  9.4906762 ,  9.60305937, 11.40722826,
8.98156312,
       11.29352642,  8.88537265,  7.22720377,  7.90399528,
12.16928872,
       10.63450602, 11.58571964, 12.791631  , 12.8121307 ,
10.50062052,
       10.90774117, 12.03386007,  7.62223752, 11.22580451,
11.06300686,
       10.90453595, 10.57886765,  9.19592924,  7.07310628,
12.18351725,
       12.82513366, 10.72641865,  8.92834963, 10.18921756,
11.98058738,
        8.12925485, 11.4598942 ,  8.54266348, 10.35744267,
11.87902492,
        6.22536288,  9.94500225, 11.62480868, 10.37586651,
7.75092116,
        8.01989856, 10.97461635,  6.77607485,  8.46678951,
12.45143457,
        6.92838551,  7.3967379 ,  6.33785794, 10.15301633,
9.98582556,
       10.71842705, 12.17080443,  9.22237014, 14.57918302,
9.26458553,
       11.10168521, 11.83288408,  6.16755336,  5.90945688,
11.99584366,
       10.42222102, 10.40472494,  4.3719043 , 10.82316731,
7.15194322,
```

```
       11.0568255 ,  7.05310577, 12.21122626,  8.72851722,
8.27358488,
       10.19258325, 11.20422819, 10.21148306,  6.99461486,
8.65270263,
        7.07180318, 10.19853357,  9.26385328, 10.8557896 ,
5.93832829,
       10.69854746,  6.0762675 ,  8.74403165, 10.4597107 ,
10.96117082,
        9.23298651, 11.1045042 , 10.94969847, 10.81762435,
8.84523347,
       10.56924815,  6.7135542 , 11.22390704,  8.52003473,
11.78931642,
        6.75282559,  8.09887519, 11.16293872, 11.00779498,
7.31674411,
       12.69709303, 11.6162036 ,  6.51419248,  7.52806623,
10.72016103,
        6.79364415, 10.87780489, 11.94124364,  8.80837458,
11.7044664 ,
       14.22082662, 10.88826112,  4.03124081, 10.43791643,
11.19623026,
       10.86626341, 11.78601239, 10.18097361,  9.91642336,
10.51233388,
       11.70258168, 10.1293405 , 10.91840844, 11.38642042,
11.60076272,
        9.00100222,  8.81791682, 16.10103323,  9.41890986,
8.19721313,
        6.50835697, 14.14119553, 10.95071028, 11.55298414,
11.28817571,
       12.94265786, 11.2485466 ,  8.87589521, 10.82664455,
11.26668668,
        8.87033995,  7.56680418,  6.55721912,  6.59391714,
6.42365633,
        9.35064533,  9.86559989,  7.75619854, 12.10925428,
8.32173564,
        6.83189616,  6.47258823, 11.47025427,  6.12398329,
9.77084571,
       10.27804495,  9.44149687,  7.04658309, 10.66765321,
11.75585633,
        8.78197049,  9.69277195, 10.28247386,  4.19876999,
7.49450012,
        6.79197404,  9.27815163,  8.98062726, 11.7683474 ,
12.00462559,
       10.59500373, 10.85043786, 11.48617679,  9.5092461 ,
10.59783913,
        9.00246809,  9.96390135,  5.99034485,  9.76420447,
9.90896524,
       13.32705794, 13.70948256, 10.8932456 ,  7.46619036,
8.11986312,
       10.64902776, 14.99929428, 12.51777165,  8.97570756,
10.56744003,
```

```
         8.32813903,  8.43920511,  9.95667429, 10.76218999,
12.73502845,
         9.94378103, 12.02904446,  8.18710351, 12.28126567,
9.89745755,
        10.88327577, 10.81194218, 12.4304059 , 10.47875215,
9.74224438,
         7.69650322, 11.78238146,  6.18234823, 11.01613526,
9.19818821,
         6.83579802, 13.13476702,  8.89044929, 11.2763003 ,
6.73608328,
         9.92291956, 10.49533306,  6.11588427, 11.79001061,
8.01435546,
         7.49744675, 10.89986095, 12.90939801, 11.11789371,
10.80976916,
         7.84819514, 11.3152181 ,  6.58152599,  8.11841895,
12.81677596,
         6.53955455,  8.05514672,  8.97745818,  9.74424437,
12.0919466 ,
        10.13817585, 11.38976114, 12.46029988, 11.48525259,
9.83783712,
         8.52344098, 11.23694422, 13.00731758, 11.98383778,
7.89035523,
         9.50594259, 12.64850837, 11.3337587 ,  8.72347668,
10.15175863,
        11.78345911, 12.39232303, 12.17375997, 11.44017137,
9.55555895,
        10.81109264, 11.71629923, 13.1886598 ,  7.53422018,
7.64682795,
         7.94414496,  9.97299829,  6.52018636,  8.09615881,
9.1501659 ,
        12.24764232, 10.90920989, 13.47772254,  7.15557761,
13.20401574,
         9.06165038,  9.41519865,  9.89917674,  8.19080015,
10.77451514,
         9.80481832,  6.97047427,  8.80894438, 11.24108995,
11.08209722,
        11.49079416, 12.5381241 , 11.0704594 ,  8.33904797,
7.6716367 ,
        10.47903477, 10.8106835 , 12.03958909,  8.20929711,
9.29368328,
        10.57693332,  9.15957227,  8.54550938, 11.52194565,
11.9030039 ,
         7.44430791,  8.55920756, 13.47924627, 13.64289544,
11.47832677,
        10.50059311, 10.19004231, 12.93141951, 10.65976992,
10.26089688,
        10.04294156, 10.26228236,  9.31705027, 12.25780575,
8.42617095,
        12.60993828, 10.3459187 ,  9.62379902, 10.02899739,
8.56086346,
```

```
       12.72237956, 10.41861783, 11.06634062,  9.95082846,
11.13305377,
       10.33473805, 12.52979155, 11.72941524,  9.85859447,
12.53372504,
       12.14693745,  9.70202091,  7.54312864,  9.97879562,
12.56927004,
       11.52930369, 12.2666071 ,  9.48230474,  7.583882  ,
7.31030277,
        7.57579116, 11.99761925,  9.09087095,  7.47578595,
11.60665869,
        8.93946963, 13.34819844,  9.86495841,  7.28544145,
9.13277121,
       14.82599372, 10.21491659, 12.74513222,  6.86555643,
10.93023664,
       10.91469433,  7.15234891, 11.35441999,  9.92453987,
10.79545212,
       10.99340805, 11.57689778, 10.85593138, 10.2173154 ,
12.17092479,
        6.5815634 ,  8.18490921,  7.84865444,  9.18746673,
8.26637953,
        8.29808717, 10.73044385,  7.43311212, 11.78384955,
10.65080072,
        8.9401935 , 10.95898024,  8.35929206, 10.67641608,
6.51258477,
       14.8868583 ,  6.80156858, 10.75845316, 10.5426978 ,
11.36702837,
        7.39891178, 12.0255648 , 15.69138428, 11.70237753,
10.8179462 ,
        9.62984277, 14.57324262, 12.43949209, 14.46338536,
7.67392   ,
        8.72277026, 11.37953838, 11.24860745,  9.57185079,
8.98720212,
        8.40154441, 10.60468019, 11.15398252,  7.74157118,
10.95349237,
       11.18665001,  9.91352566, 11.79413023,  9.38712319,
10.59560391,
        6.19766215, 10.66787769, 10.66380577,  6.03187541,
7.72230324,
       10.13826737, 10.17650428, 11.1248837 , 10.04155115,
6.91675574,
        6.38148337,  7.77581373,  5.50761155, 10.75778254,
9.29482009,
        6.56161074, 10.06816906, 12.04766466,  9.8845791 ,
8.580779  ,
        6.45557143, 12.15821545, 13.47071875, 12.43116935,
8.7553012 ,
       11.14503588, 12.62113345, 11.03742739,  7.39147101,
5.91373942,
       10.74044122, 10.76118437,  8.24698   , 11.36735436,
9.54432797,
```

```
        12.14198003, 10.99237788, 12.44365837, 10.52032008,
11.9124589 ,
         9.69905245,  8.82684666, 12.34499562, 12.41677269,
8.05725243,
        10.14765395,  8.81173493,  8.64346351, 12.00512145,
6.69252127,
        11.26214019,  9.39949135, 10.35254823, 12.55940148,
9.57904793,
         6.62928902, 11.86507406, 13.56146012, 11.38199234,
10.75807812,
         6.48326652,  8.51159242,  8.68523124,  7.99852625,
8.8243379 ,
         8.0803471 , 13.17906671,  7.90555411, 12.40074747,
11.872995  ,
         7.69687431, 10.55984387,  6.93571761,  6.66462981,
6.62483401,
         9.68960716, 10.73878994, 10.19371516,  7.11217796,
9.05164841,
         6.74730739,  6.73554673, 12.83550478,  3.82774117,
9.92425955,
         8.78926896,  7.49394638, 10.95690827,  8.89633788,
10.12193577,
        12.17050243, 12.65382581,  6.9338426 , 11.11969097,
13.38568954,
         9.34037062, 11.77945219, 10.65705929, 13.51032139,
6.42714657,
        10.09279983, 11.58782308,  6.90204577,  8.38686119,
13.08419845,
        11.20235771, 11.33610572, 11.43517562, 12.72383264,
9.26028934,
        12.8248419 , 11.3964185 , 14.92382172, 11.08632516,
11.0906055 ,
        12.32892428, 10.44420753, 10.61430593, 12.09567335,
8.07481661,
        11.09040553,  9.28705661,  9.76502556, 13.51479546,
9.01471835,
         6.80235681,  6.49317209,  9.23694001, 11.69579537,
6.72267061,
        10.00602851, 17.24796093,  5.83061287,  9.70308402,
11.91497792,
         9.10265205,  8.29419747,  9.54736395,  7.77396338,
9.07678974,
        11.99706021, 10.38172391, 10.83567129, 10.31291474,
10.3059796 ,
         8.4367328 ,  9.73945245,  8.12059434,  7.89877523,
12.0568822 ,
         7.49833362, 11.12876435, 12.45065131,  9.89396318,
6.57817229,
         9.35968264,  8.83279147, 11.24059465, 10.44946547,
```

```
6.9483967 ,
       10.4638248 ])
```

```
rg.coef_
```

```
array([-0.30301698, -0.77060619,  0.25767474,  0.95539967,
0.97628841,
       -1.42719194,  0.05094781,  1.73422975])
```

```
metrics.r2_score(y_test,rg_pred)
```

```
0.4570874209477108
```

```
np.sqrt(mean_squared_error(y_test,rg_pred))
```

```
2.387177893010143
```