Assignment-4

```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import sklearn
```

```
data = pd.read_csv(r"C:\Users\LENOVO\Downloads\abalone.csv")
```

```
data.head()
```

	Sex	Length	Diameter	Height	Whole weight	Shucked weight	Viscera weight	Shell weight	Rings
0	M	0.455	0.365	0.095	0.5140	0.2245	0.1010	0.150	15
1	M	0.350	0.265	0.090	0.2255	0.0995	0.0485	0.070	7
2	F	0.530	0.420	0.135	0.6770	0.2565	0.1415	0.210	9
3	M	0.440	0.365	0.125	0.5160	0.2155	0.1140	0.155	10
4	I	0.330	0.255	0.080	0.2050	0.0895	0.0395	0.055	7

```
data['Height'].unique()
```

```
array([0.095, 0.09 , 0.135, 0.125, 0.08 , 0.15 , 0.14 , 0.11 , 0.145,
       0.1  , 0.13 , 0.085, 0.155, 0.165, 0.185, 0.18 , 0.175, 0.2  ,
       0.105, 0.045, 0.055, 0.05 , 0.12 , 0.07 , 0.16 , 0.06 , 0.17 ,
       0.195, 0.19 , 0.115, 0.075, 0.065, 0.215, 0.21 , 0.23 , 0.205,
       0.22 , 0.04 , 0.01 , 0.03 , 0.035, 0.225, 0.24 , 0.235, 0.02 ,
       0.025, 0.015, 0.   , 0.515, 0.25 , 1.13 ])
```



```
array(['M', 'F', 'I'], dtype=object)
```

```
data['Height'].value_counts()
```

```
0.150    267
0.140    220
0.155    217
0.175    211
0.160    205
0.125    202
0.165    193
0.135    189
0.145    182
0.130    169
0.120    169
```

```
0.170    160
0.100    145
0.110    135
0.115    133
0.180    131
0.090    124
0.105    114
0.185    103
0.190    103
0.095     91
0.195     78
0.080     76
0.085     74
0.200     68
0.075     61
0.070     47
0.205     45
0.065     39
0.215     31
0.060     26
0.055     25
0.210     23
0.050     18
0.220     17
0.040     13
0.225     13
0.045     11
0.230     10
0.030      6
0.035      6
0.235      6
0.025      5
0.240      4
0.250      3
0.020      2
0.015      2
0.000      2
0.010      1
0.515      1
1.130      1
```

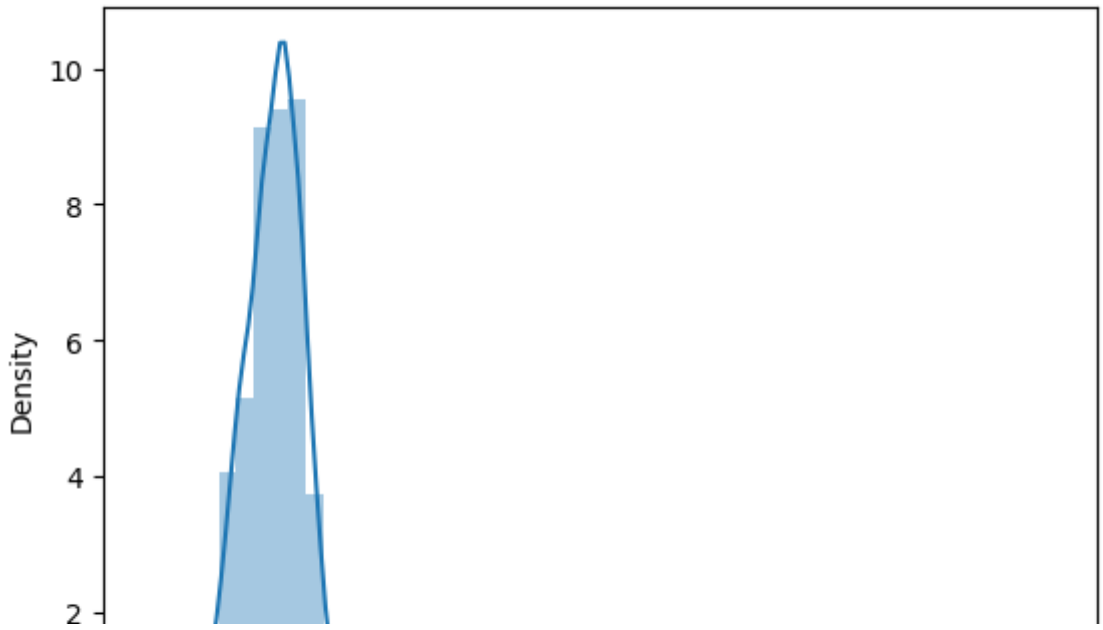
Saved successfully!

data Visualization

uni variate analysis

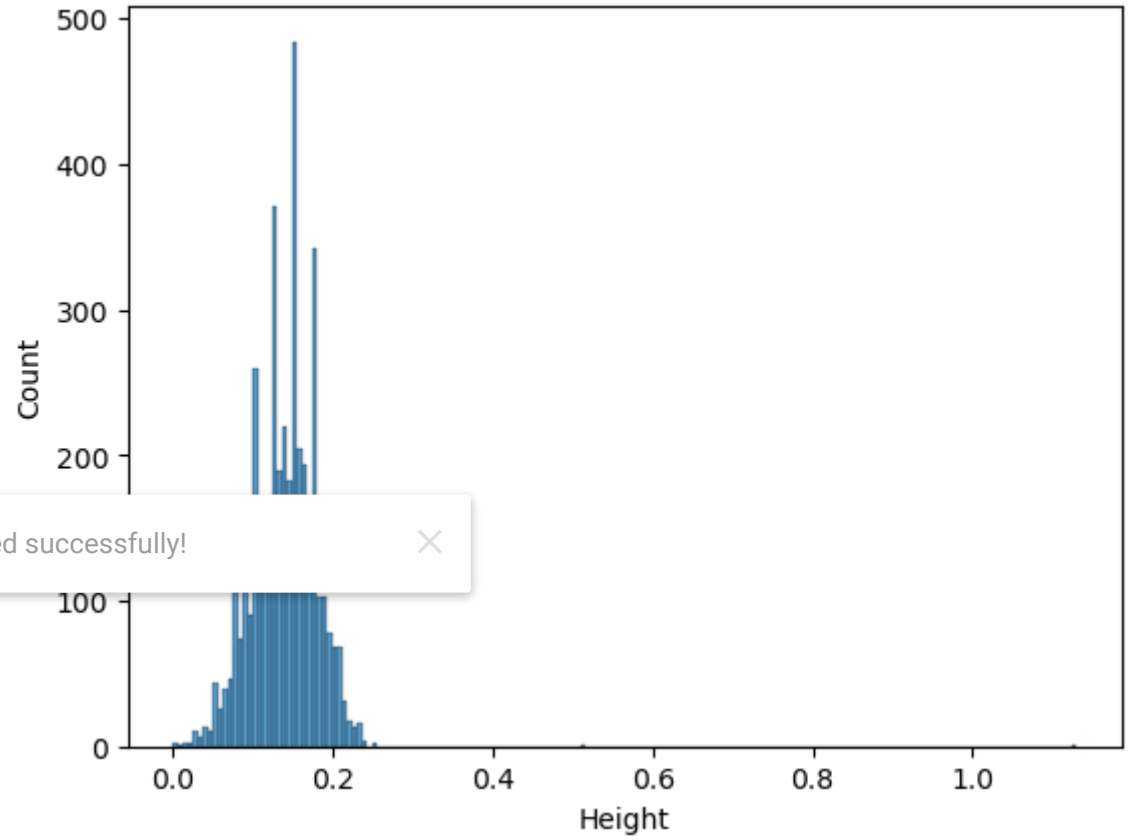
```
sns.distplot(data['Height'])
```

```
C:\Users\Admin\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarni
  warnings.warn(msg, FutureWarning)
<AxesSubplot:xlabel='Height', ylabel='Density'>
```




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

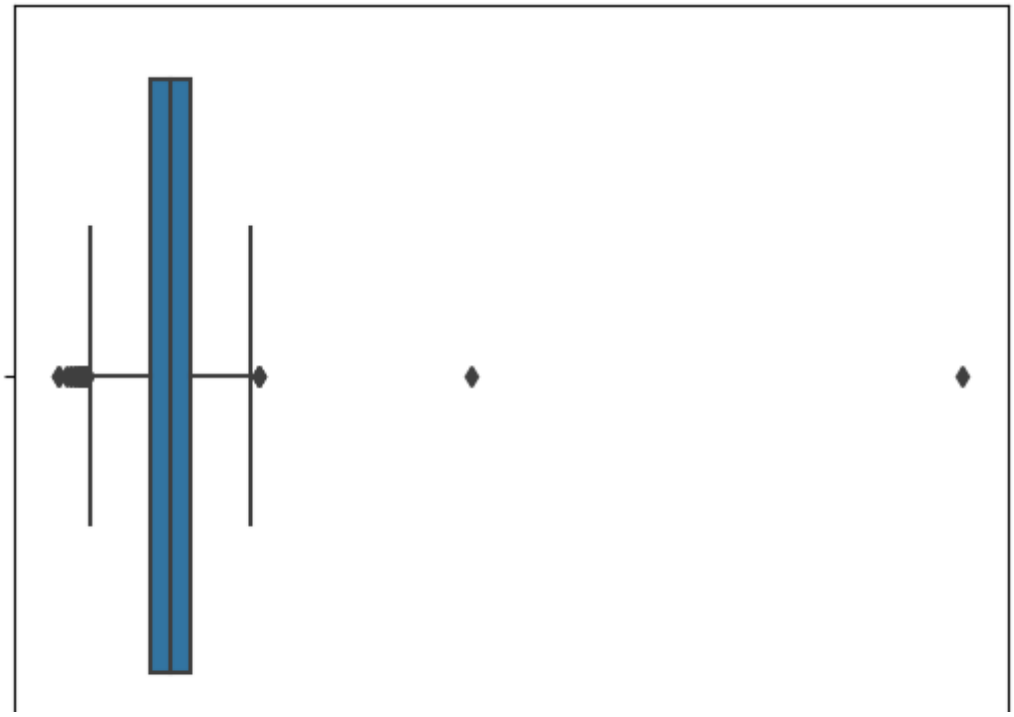
```
<AxesSubplot:xlabel='Height', ylabel='Count'>
```



Saved successfully!

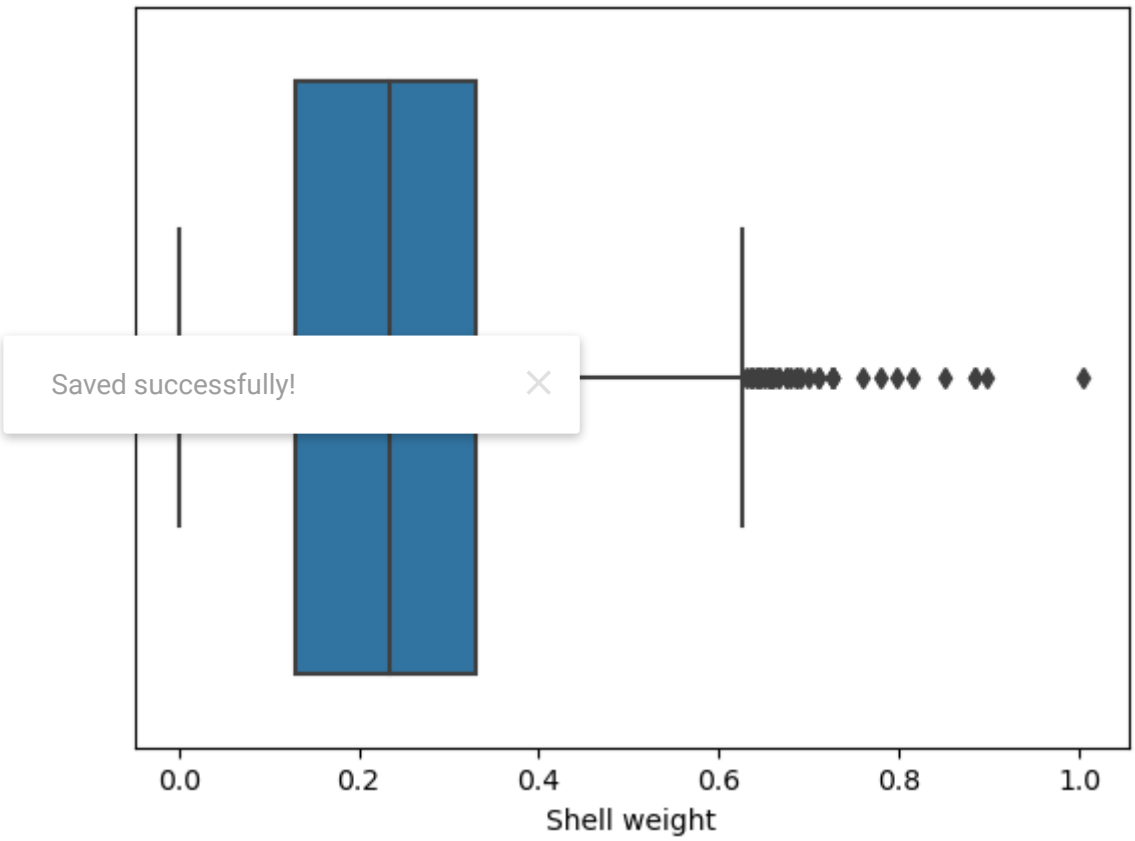
```
sns.boxplot(data['Height'])
```

```
C:\Users\Admin\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning:
  warnings.warn(
<AxesSubplot:xlabel='Height'>
```



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

```
C:\Users\Admin\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning:
  warnings.warn(
<AxesSubplot:xlabel='Shell weight'>
```

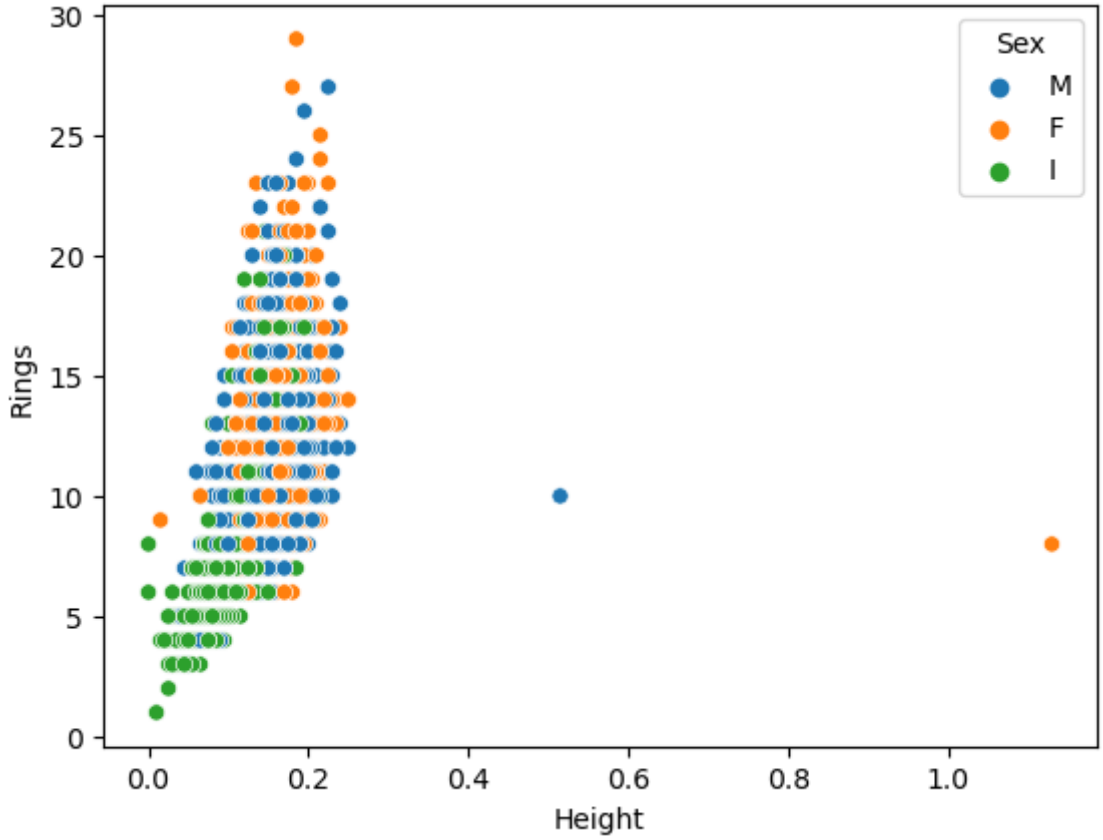


Bi variate analysis

Multivariate analysis

```
sns.scatterplot(data['Height'], data['Rings'], hue = data['Sex'])
```

```
C:\Users\Admin\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning:
  warnings.warn(
<AxesSubplot:xlabel='Height', ylabel='Rings'>
```



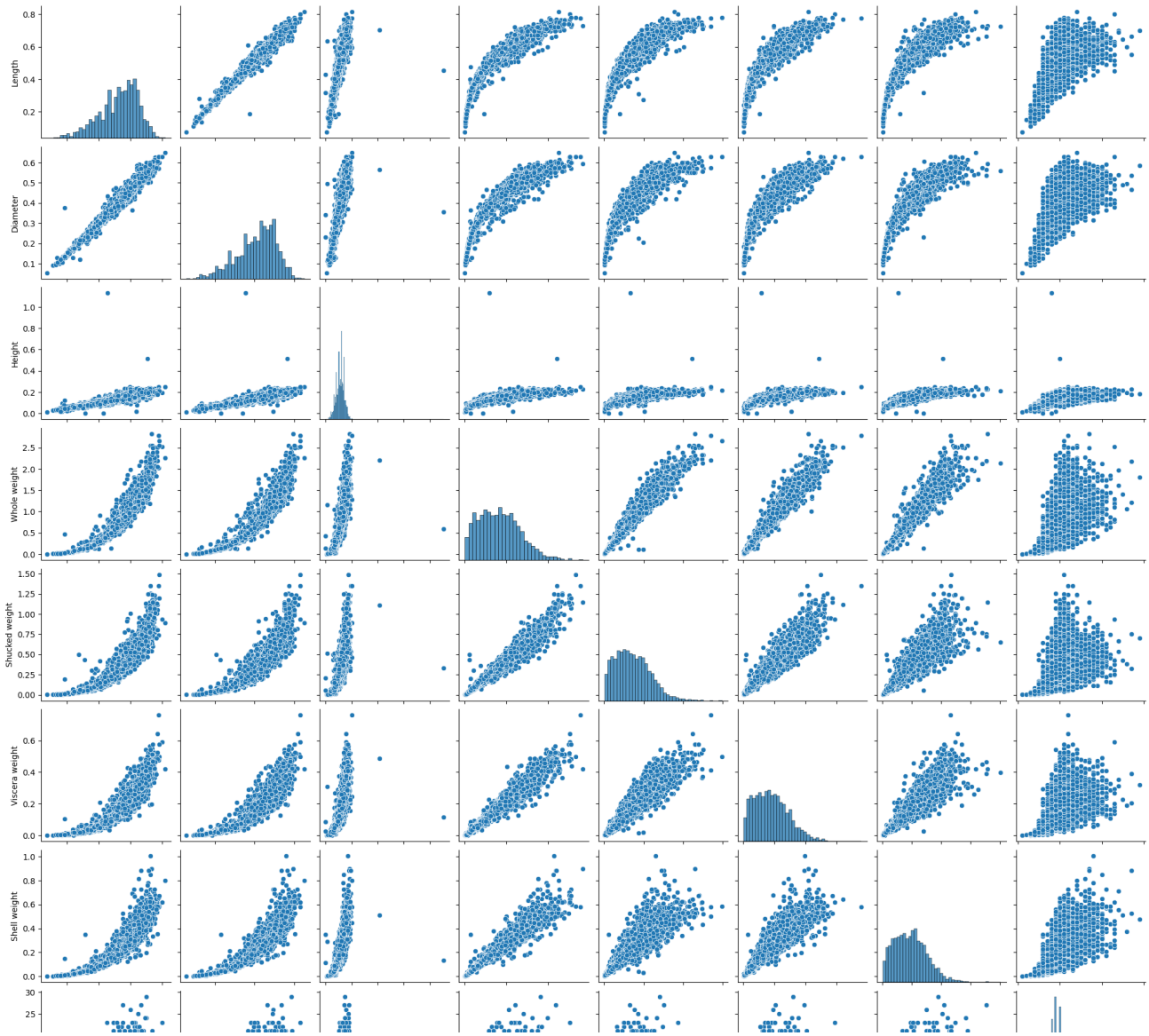

```
sns.scatterplot(data['Height'], data['Rings'], hue = data['Whole weight'])
```

Saved successfully!

```
C:\Users\Admin\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning:
  warnings.warn(
<AxesSubplot:xlabel='Height', ylabel='Rings'>
```


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

```
<seaborn.axisgrid.PairGrid at 0x2976ca76970>
```

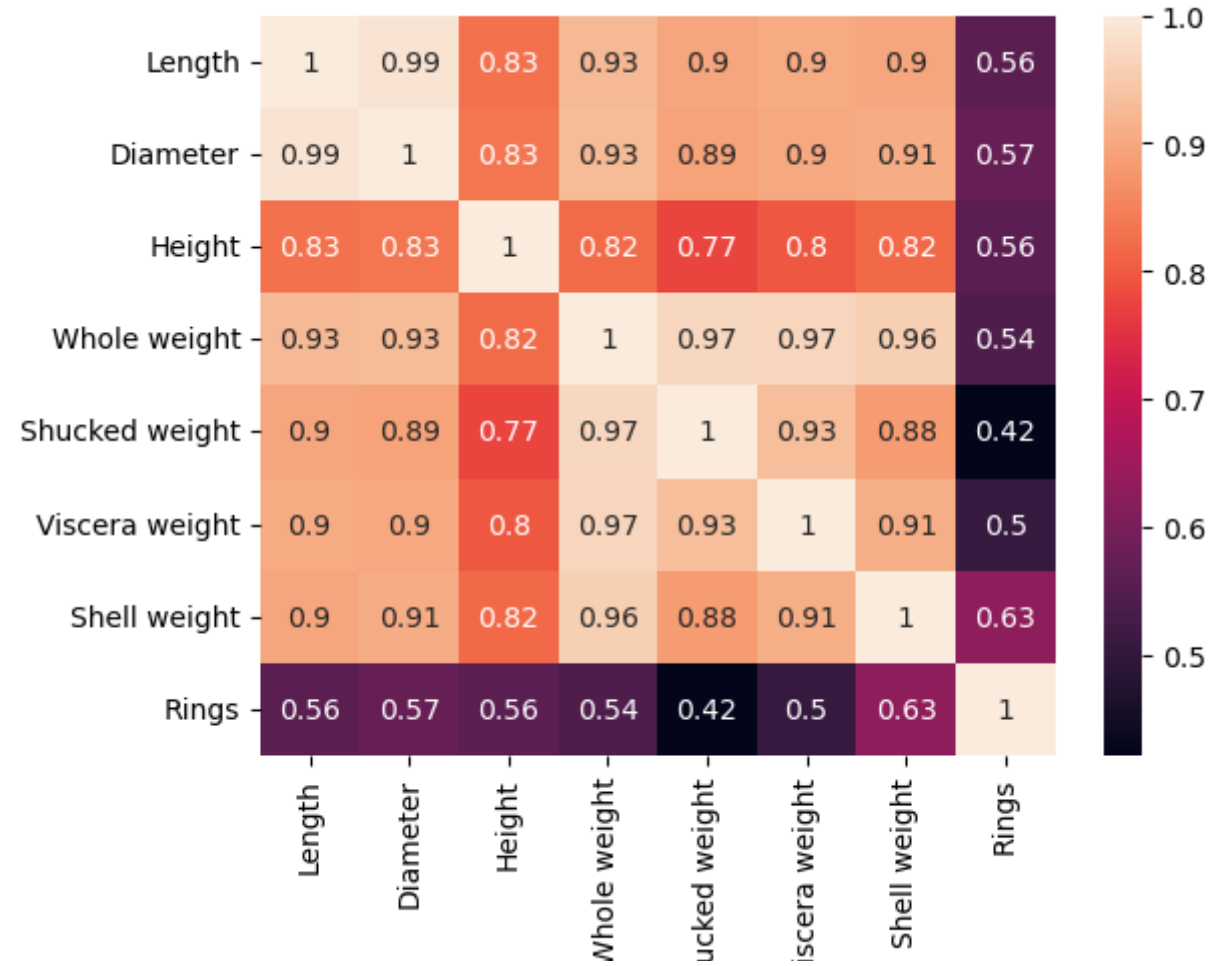



```
data.corr()
```

	Length	Diameter	Height	Whole weight	Shucked weight	Viscera weight	
			312	0.827554	0.925261	0.897914	0.903018	0
Diameter	0.986812	1.000000	0.833684	0.925452	0.893162	0.899724	0
Height	0.827554	0.833684	1.000000	0.819221	0.774972	0.798319	0
Whole weight	0.925261	0.925452	0.819221	1.000000	0.969405	0.966375	0
Shucked weight	0.897914	0.893162	0.774972	0.969405	1.000000	0.931961	0
Viscera weight	0.903018	0.899724	0.798319	0.966375	0.931961	1.000000	0


```
sns.heatmap(data.corr(), annot = True)
```

```
<AxesSubplot:>
```



Data preprocessing

```
data.isnull().any()
```

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

Saved successfully!

```
data.isnull().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
```

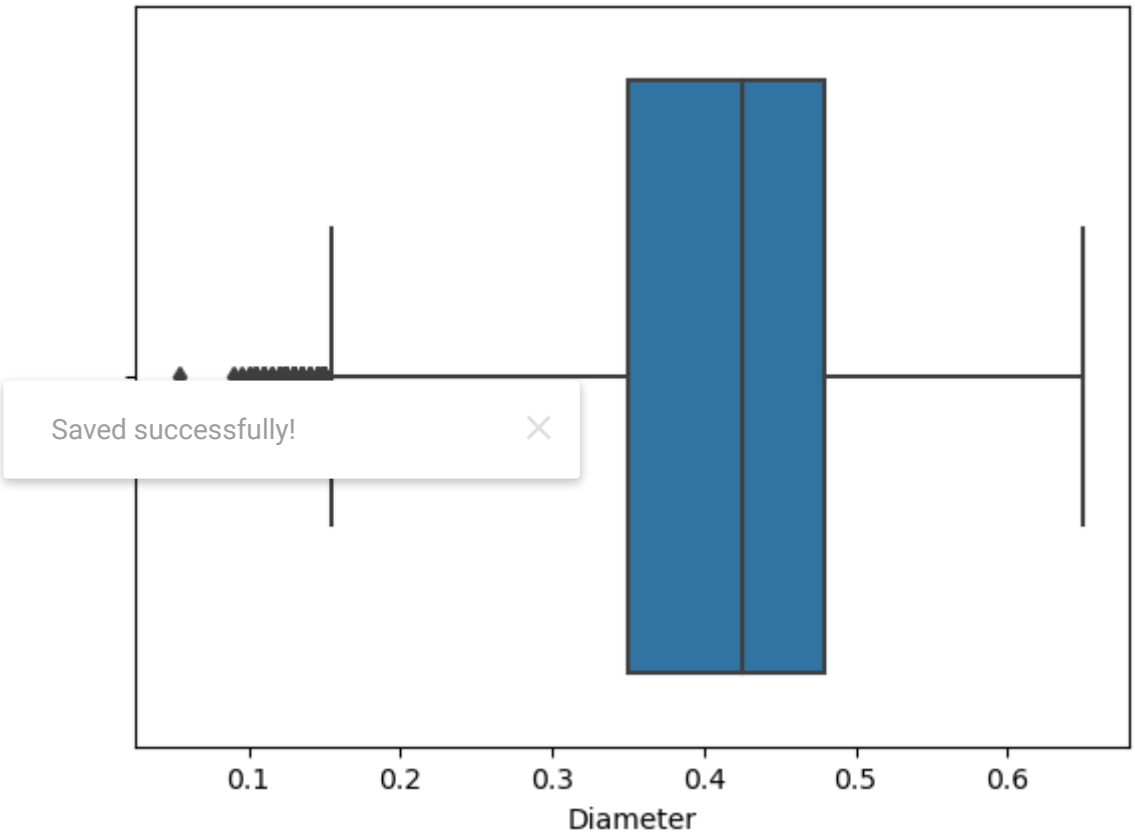
```
data.mean()
```

```
C:\Users\Admin\AppData\Local\Temp\ipykernel_10828\531903386.py:1: FutureWarning: Drop
  data.mean()
Length            0.523992
Diameter          0.407881
Height            0.139516
Whole weight      0.828742
Shucked weight    0.359367
Viscera weight    0.180594
Shell weight      0.238831
Rings             9.933684
dtype: float64
```

outliers

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

```
C:\Users\Admin\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning:
  warnings.warn(
<AxesSubplot:xlabel='Diameter'>
```




```
q  = data.quantile([0.75,0.25])
```

```
q
```

	Length	Diameter	Height	Whole weight	Shucked weight	Viscera weight	Shell
0.75	0.615	0.48	0.165	1.1530	0.502	0.2530	
0.25	0.450	0.35	0.115	0.4415	0.186	0.0935	

```
iqr =  q.iloc[0] - q.iloc[1]   #q3 -  q1 === 0.75 - 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
```

```
u = q.iloc[0] + (1.5*iqr)   # q3 + 1.5 * iqr
```

```
u
```

```
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
```

Saved successfully!

```
*n iqr
```

```
l
```

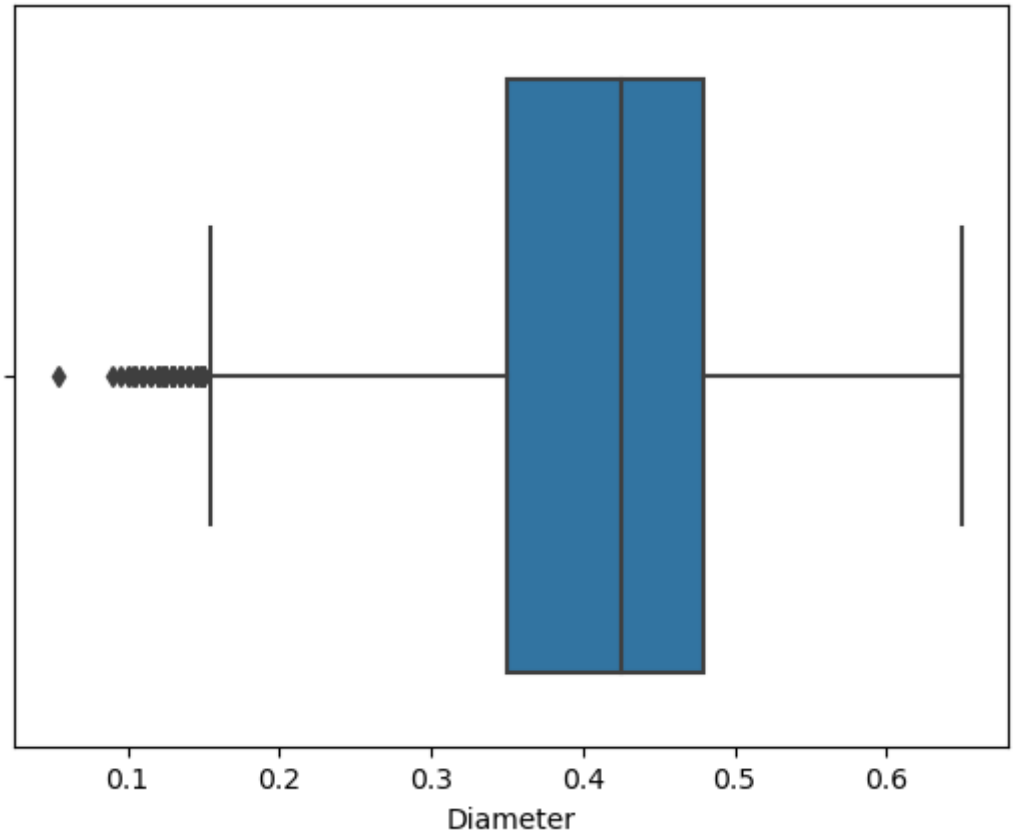
```
Length            0.20250
Diameter          0.15500
Height            0.04000
Whole weight     -0.62575
Shucked weight   -0.28800
Viscera weight   -0.14575
Shell weight     -0.16850
Rings             3.50000
dtype: float64
```

Handling outliers

```
data['Diameter'] = np.where(data['Diameter']>0.67500, 1.00000,data['Diameter'])
```

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

```
C:\Users\Admin\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning:
  warnings.warn(
<AxesSubplot:xlabel='Diameter'>
```



Encoding

Saved successfully!

```
belEncoder, OneHotEncoder
le = LabelEncoder()
oneh = OneHotEncoder()
data['Sex'] = le.fit_transform(data['Sex'])
```

```
data.head()
```

```
data['Whole weight'] = le.fit_transform(data['Whole weight'])
data['Viscera weight'] = le.fit_transform(data['Viscera weight'])
```

```
data.head()
```

Split the data into dependent and independent values

```
x = data.iloc[:, 0:6]
```

```
x
```

	Sex	Length	Diameter	Height	Whole weight	Shucked weight
0	2	0.455	0.365	0.095	715	0.2245
1	2	0.350	0.265	0.090	285	0.0995
2	0	0.530	0.420	0.135	962	0.2565
3	2	0.440	0.365	0.125	718	0.2155
4	1	0.330	0.255	0.080	253	0.0895
...	...	...	...	...	...	...
4172	0	0.565	0.450	0.165	1289	0.3700
4173	2	0.590	0.440	0.135	1411	0.4390
4174	2	0.600	0.475	0.205	1727	0.5255
4175	0	0.625	0.485	0.150	1610	0.5310
4176	2	0.710	0.555	0.195	2348	0.9455

4177 rows × 6 columns

```
y = data['Rings']
```

```
y
```

```
0       15
1        7
..
4172    11
4173    10
4174     9
4175    10
4176    12
Name: Rings, Length: 4177, dtype: int64
```

Saved successfully!

Scaling

```
from sklearn.preprocessing import StandardScaler, MinMaxScaler
sc = StandardScaler()
x_scaled = sc.fit_transform(x)
```

```
x_scaled
```

```
array([[ 1.15198011, -0.57455813, -0.43214879, -1.06442415, -0.6673621 ,
        -0.60768536],
       [ 1.15198011, -1.44898585, -1.439929  , -1.18397831, -1.31885041,
        -1.17090984],
       [-1.28068972,  0.05003309,  0.12213032, -0.10799087, -0.2931351 ,
        -0.4634999 ],
       ...,
       [ 1.15198011,  0.6329849 ,  0.67640943,  1.56576738,  0.86590805,
         0.74855917],
       [-1.28068972,  0.84118198,  0.77718745,  0.25067161,  0.68864263,
         0.77334105],
       [ 1.15198011,  1.54905203,  1.48263359,  1.32665906,  1.80677838,
         2.64099341]])
```

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_scaled, y, test_size = 0.3, random_s
```

```
x_train
```

```
array([[ 1.15198011,  0.79954256,  1.0291325 ,  0.84844242,  1.07499035,
         0.95582578],
       [-0.0643548 , -1.49062526, -1.54070702, -1.30353247, -1.37793889,
        -1.14612796],
       [-0.0643548 , -1.24078877, -1.33915098, -1.06442415, -1.30824479,
        -1.19118592],
       ...,
       [ 1.15198011,  0.59134549,  0.42446438,  0.13111745,  0.37198901,
         0.46694694],
       [-1.28068972,  0.84118198,  0.82757646,  0.6093341 ,  0.76136691,
         0.53002808],
       [                                    7, -0.83526087, -0.70576167, -1.10219267,
```

Saved successfully!

```
x_train.shape
```

```
(2923, 6)
```

```
x_test
```

```
array([[ 1.15198011,  0.21659075,  0.17251933,  0.37022577,  0.27047804,
        -0.36887819],
       [-0.0643548 , -0.1998034 , -0.07942572, -0.46665335, -0.42797803,
        -0.44322382],
       [ 1.15198011,  0.79954256,  0.72679844,  0.37022577,  1.02499241,
         0.75531787],
       ...,
       [-1.28068972,  0.92446081,  0.87796547, -2.97729071,  0.82500065,
         0.68547803],
```

```
            [ 1.15198011,  1.13265788,  0.97874349,  1.44621322,  1.58254519,
              0.57283314],
            [-0.0643548 ,  0.79954256,  0.77718745,  0.72888826,  0.93560215,
              0.54579836]])
```

```
x_test.shape
```

```
     (1254, 6)
```

```
y_train
```

```
     1376    10
     1225     5
     2722     7
     3387    18
     2773    11
             ..
     1033    10
     3264    12
     1653    10
     2607     9
     2732     8
     Name: Rings, Length: 2923, dtype: int64
```

```
y_test
```

```
     668     13
     1580     8
     3784    11
     463      5
     2615    12
             ..
     1052    12
     3439     8
```



```
     Name: Rings, Length: 1254, dtype: int64
```