```
import pandas as pd
import numpy  as np
import matplotlib.pyplot as plt
import seaborn as sns
from scipy.stats import stats


df = pd.read_csv("/content/abalone.csv")


df.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

```
df.tail()
```

	Sex	Length	Diameter	Height	Whole weight	Shucked weight	Viscera weight	Shell weight	Rings
4172	F	0.565	0.450	0.165	0.8870	0.3700	0.2390	0.2490	11
4173	M	0.590	0.440	0.135	0.9660	0.4390	0.2145	0.2605	10
4174	M	0.600	0.475	0.205	1.1760	0.5255	0.2875	0.3080	9
4175	F	0.625	0.485	0.150	1.0945	0.5310	0.2610	0.2960	10
4176	M	0.710	0.555	0.195	1.9485	0.9455	0.3765	0.4950	12

```
df.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
```

```
df.describe()
```

1 to 8 of 8 entries Filter

index	Length	Diameter	Height	Whole weight
count	4177.0	4177.0	4177.0	4177.0
mean	0.5239920995930094	0.40788125448886764	0.13951639932966242	0.8287421594445774
std	0.12009291256479956	0.09923986613365945	0.041827056607257274	0.4903890182309977
min	0.075	0.055	0.0	0.002
25%	0.45	0.35	0.115	0.4415
50%	0.545	0.425	0.14	0.7995
75%	0.615	0.48	0.165	1.153
max	0.815	0.65	1.13	2.8255

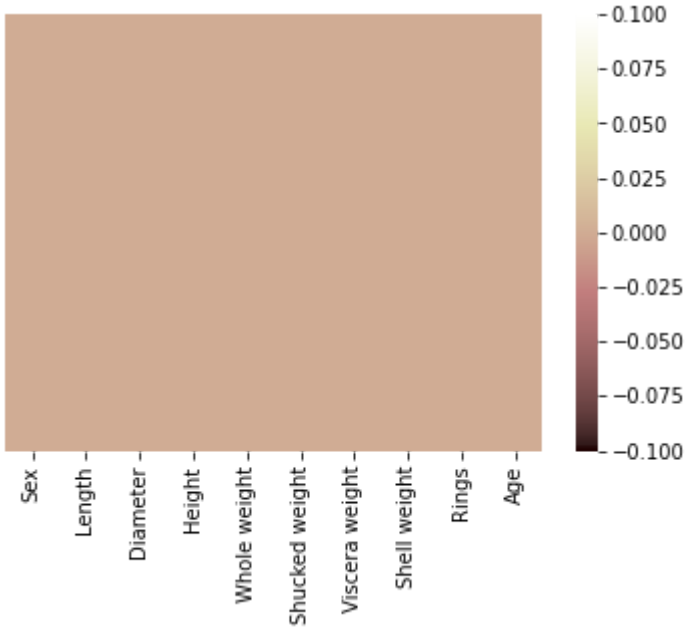
Show 25 per page
Like what you see? Visit the data table notebook to learn more about interactive tables.

```
df.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
```

```
sns.heatmap(df.isnull(),yticklabels=False,cmap='pink')
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f57bf3ff290>
```



```
df.corr()
```

1 to 8 of 8 entries Filter

index	Length	Diameter	Height	Whole weigh
Length	1.0	0.9868115846025	0.8275536093192142	0.9252611721489
Diameter	0.9868115846025	1.0	0.8336836879586538	0.925452101507
Height	0.8275536093192142	0.8336836879586538	1.0	0.8192207728553
Whole weight	0.9252611721489467	0.9254521015071313	0.8192207728553582	
Shucked weight	0.8979136582496655	0.8931624751432796	0.7749722929028299	0.969405456703
Viscera weight	0.9030176990077563	0.8997244291071196	0.79831929752753	0.966375078273
Shell weight	0.8977055691879205	0.905329781213082	0.8173380147032083	0.9553554421763
Rings	0.5567195769296182	0.5746598513059198	0.5574673244580344	0.5403896769238

Show 25 per page
Like what you see? Visit the data table notebook to learn more about interactive tables.

```
df['Sex'].value_counts()
```

```
M    1528
I    1342
F    1307
Name: Sex, dtype: int64
```

```
df['Sex'].unique()
```

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

```
df['Sex'] = df['Sex'].map({'M': 0, 'I': 1, 'F':2})
```

```
df.head()
```

1 to 5 of 5 entries Filter

index	Sex	Length	Diameter	Height	Whole weight	Shucked weight	Viscera weight	Shell weigh
0	0	0.455	0.365	0.095	0.514	0.2245	0.101	0.1
1	0	0.35	0.265	0.09	0.2255	0.0995	0.0485	0.0
2	2	0.53	0.42	0.135	0.677	0.2565	0.1415	0.2
3	0	0.44	0.365	0.125	0.516	0.2155	0.114	0.15
4	1	0.33	0.255	0.08	0.205	0.0895	0.0395	0.05

Show 25 per page
Like what you see? Visit the data table notebook to learn more about interactive tables.

```
df.tail()
```

1 to 5 of 5 entries Filter

index	Sex	Length	Diameter	Height	Whole weight	Shucked weight	Viscera weight	Shell weigh
4172	2	0.565	0.45	0.165	0.887	0.37	0.239	0.24
4173	0	0.59	0.44	0.135	0.966	0.439	0.2145	0.260
4174	0	0.6	0.475	0.205	1.176	0.5255	0.2875	0.30

ADDING AGE COLUMN

```
df['Age'] = df['Rings'] + 2.5
```

```
df.head()
```

1 to 5 of 5 entries Filter

index	Sex	Length	Diameter	Height	Whole weight	Shucked weight	Viscera weight	Shell weigh
0	0	0.455	0.365	0.095	0.514	0.2245	0.101	0.1
1	0	0.35	0.265	0.09	0.2255	0.0995	0.0485	0.0
2	2	0.53	0.42	0.135	0.677	0.2565	0.1415	0.2
3	0	0.44	0.365	0.125	0.516	0.2155	0.114	0.15
4	1	0.33	0.255	0.08	0.205	0.0895	0.0395	0.05

Show 25 per page

Like what you see? Visit the data table notebook to learn more about interactive tables.

```
df.columns
```

```
Index(['Sex', 'Length', 'Diameter', 'Height', 'Whole weight', 'Shucked weight',
       'Viscera weight', 'Shell weight', 'Rings', 'Age'],
      dtype='object')
```

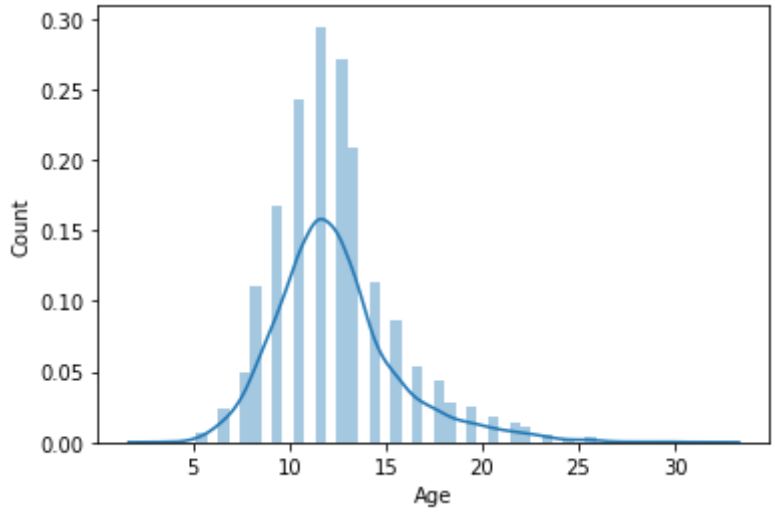
Data visualization

```
df['Sex'].value_counts().plot(kind='pie')
plt.legend()
plt.xlabel('Sex')
plt.ylabel('Count')
```

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

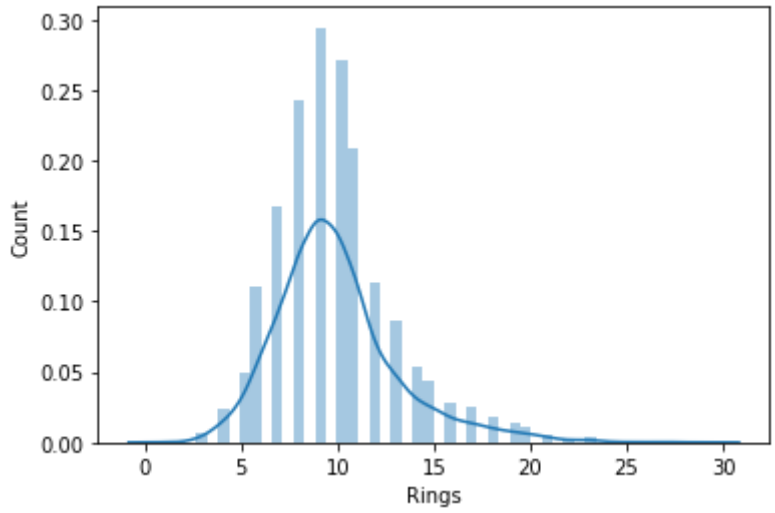
```
sns.distplot(df['Age'])
plt.xlabel('Age')
plt.ylabel('Count')
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/distributions.py:2619: FutureWarning:
  warnings.warn(msg, FutureWarning)
Text(0, 0.5, 'Count')
```




```
sns.distplot(df['Rings'])
plt.xlabel('Rings')
plt.ylabel('Count')
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/distributions.py:2619: FutureWarning:
  warnings.warn(msg, FutureWarning)
Text(0, 0.5, 'Count')
```



Bi-variate analysis

```
df.head()
```

1 to 5 of 5 entries Filter

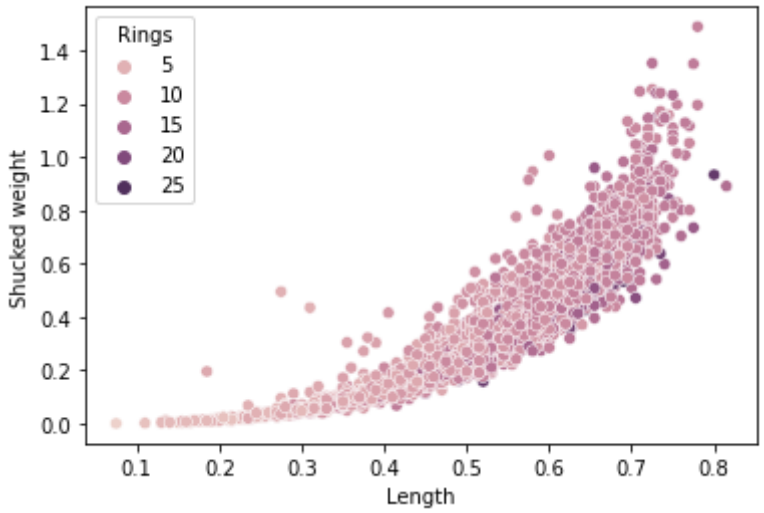
index	Sex	Length	Diameter	Height	Whole weight	Shucked weight	Viscera weight	Shell weigh
0	0	0.455	0.365	0.095	0.514	0.2245	0.101	0.1
1	0	0.35	0.265	0.09	0.2255	0.0995	0.0485	0.0
2	2	0.53	0.42	0.135	0.677	0.2565	0.1415	0.2
3	0	0.44	0.365	0.125	0.516	0.2155	0.114	0.15
4	1	0.33	0.255	0.08	0.205	0.0895	0.0395	0.05

Show 25 per page

Like what you see? Visit the data table notebook to learn more about interactive tables.

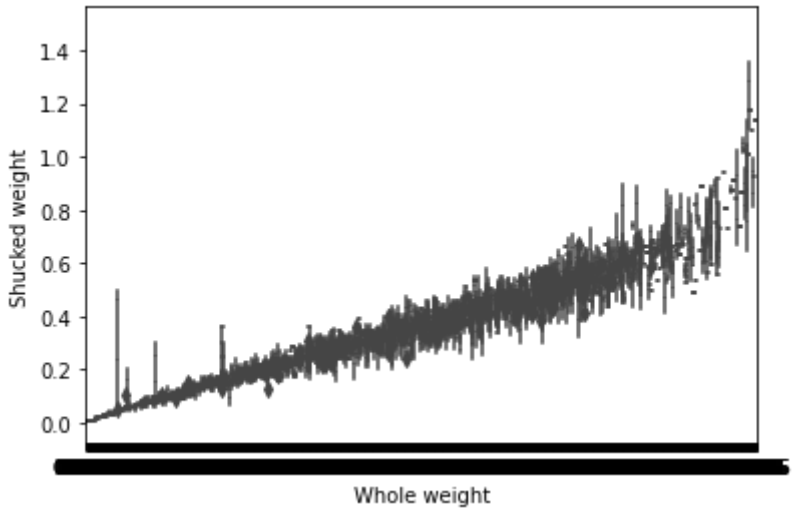
```
sns.scatterplot(data=df, x='Length', y='Shucked weight', hue='Rings',)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f57befe3ed0>
```



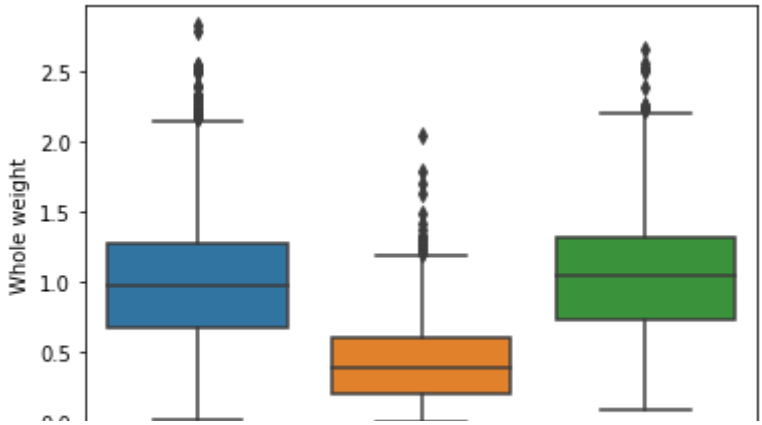

```
sns.boxplot(data=df, x='Whole weight', y='Shucked weight')
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f57bef71090>
```



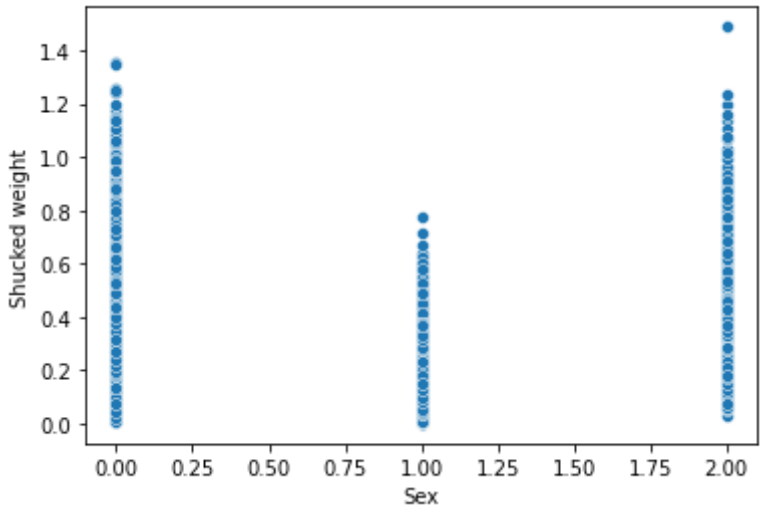

```
sns.boxplot(data=df, x='Sex', y='Whole weight')
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f57beb39810>
```



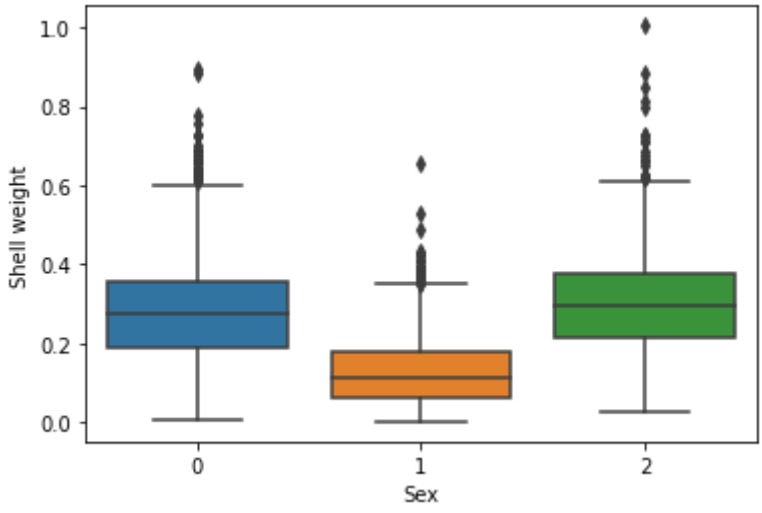

```
sns.scatterplot(data=df, x='Sex', y='Shucked weight')
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f57a17fffd0>
```




```
sns.boxplot(data=df, x='Sex', y='Shell weight')
```

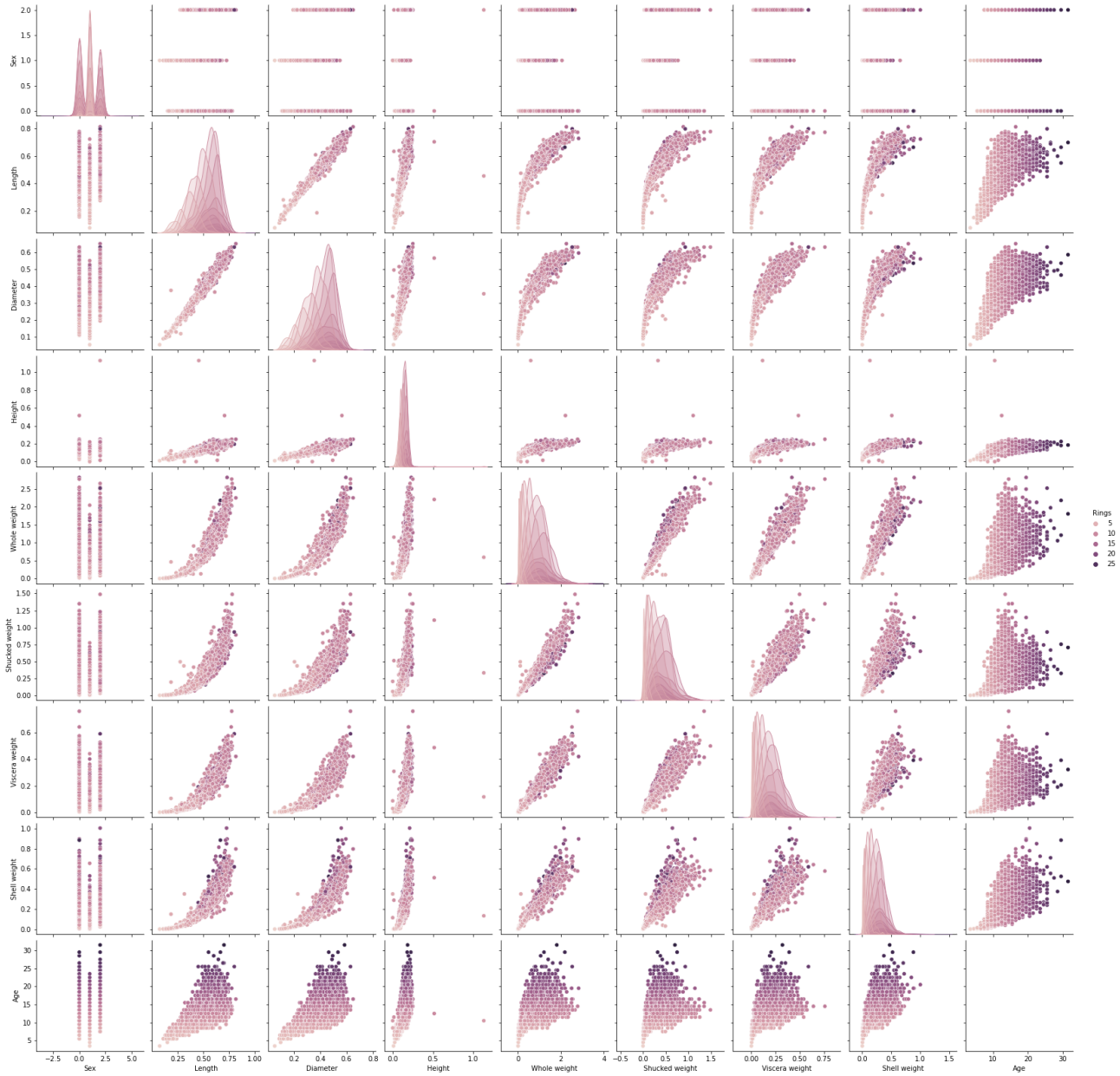
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f57beb61b50>
```



Univariate analysis

```
sns.pairplot(data=df, hue='Rings')
```

```
<seaborn.axisgrid.PairGrid at 0x7f57a111acd0>
```




```
df.describe()
```

1 to 8 of 8 entries Filter

index	Sex	Length	Diameter	Height
count	4177.0	4177.0	4177.0	4177.0
mean	0.9470912137898013	0.5239920995930094	0.40788125448886764	0.13951639932966242
std	0.8222404151746827	0.12009291256479956	0.09923986613365945	0.041827056607257274
min	0.0	0.075	0.055	0.0
25%	0.0	0.45	0.35	0.115
50%	1.0	0.545	0.425	0.14
75%	2.0	0.615	0.48	0.165
max	2.0	0.815	0.65	1.13

Show 25 per page

Like what you see? Visit the data table notebook to learn more about interactive tables.

```
df.corr()['Age']
```

```
Sex               0.034627
Length            0.556720
Diameter          0.574660
Height            0.557467
Whole weight      0.540390
Shucked weight    0.420884
Viscera weight    0.503819
Shell weight      0.627574
Rings             1.000000
Age               1.000000
Name: Age, dtype: float64
```

```
df.shape
```

```
(4177, 10)
```

Checking outliers for the data

```
df.head()
```

1 to 5 of 5 entries Filter

index	Sex	Length	Diameter	Height	Whole weight	Shucked weight	Viscera weight	Shell weigh
0	0	0.455	0.365	0.095	0.514	0.2245	0.101	0.1
1	0	0.35	0.265	0.09	0.2255	0.0995	0.0485	0.0
2	2	0.53	0.42	0.135	0.677	0.2565	0.1415	0.2
3	0	0.44	0.365	0.125	0.516	0.2155	0.114	0.15
4	1	0.33	0.255	0.08	0.205	0.0895	0.0395	0.05

Show 25 per page

Like what you see? Visit the data table notebook to learn more about interactive tables

```
df.drop('Age',axis=1,inplace=True)
```

```
df.head()
```

1 to 5 of 5 entries Filter

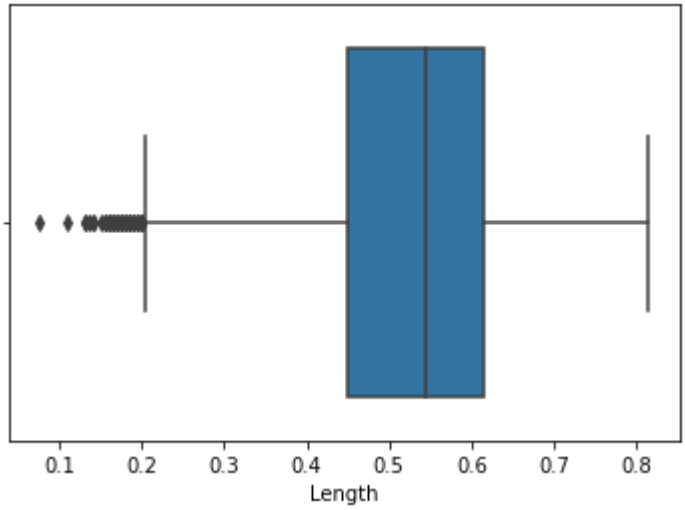
index	Sex	Length	Diameter	Height	Whole weight	Shucked weight	Viscera weight	Shell weigh
0	0	0.455	0.365	0.095	0.514	0.2245	0.101	0.1
1	0	0.35	0.265	0.09	0.2255	0.0995	0.0485	0.0
2	2	0.53	0.42	0.135	0.677	0.2565	0.1415	0.2
3	0	0.44	0.365	0.125	0.516	0.2155	0.114	0.15
4	1	0.33	0.255	0.08	0.205	0.0895	0.0395	0.05

Show 25 per page

Like what you see? Visit the data table notebook to learn more about interactive tables.

```
sns.boxplot(x=df['Length'])
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f579f241250>
```

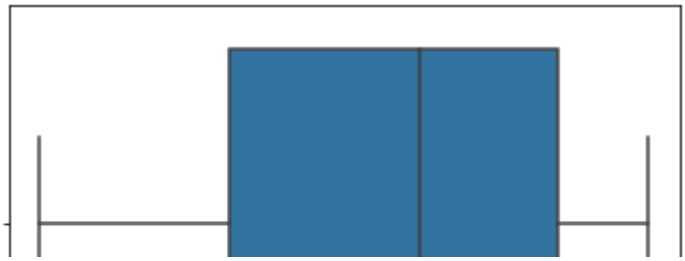



```
tenth_per = np.percentile(df['Length'], 10)
nine_per  = np.percentile(df['Length'], 90)

df['Length'] = np.where(df['Length'] < tenth_per, tenth_per, df['Length'])
df['Length'] = np.where(df['Length'] > nine_per, nine_per, df['Length'])
```

IQR

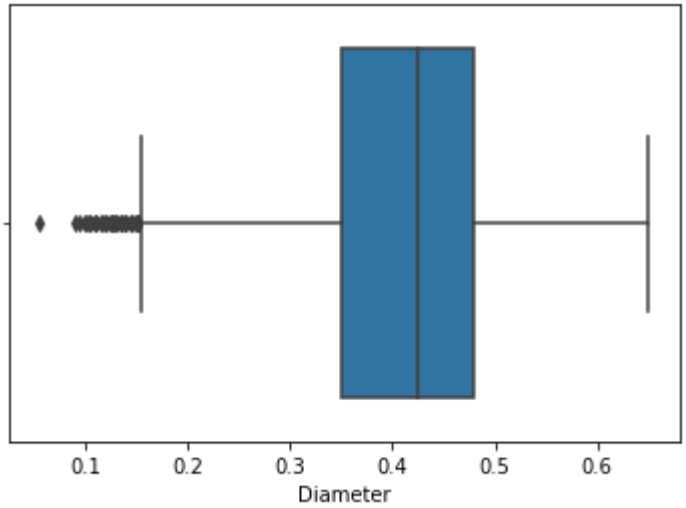
```
sns.boxplot(x=df['Length'])
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f579f24e390>
```



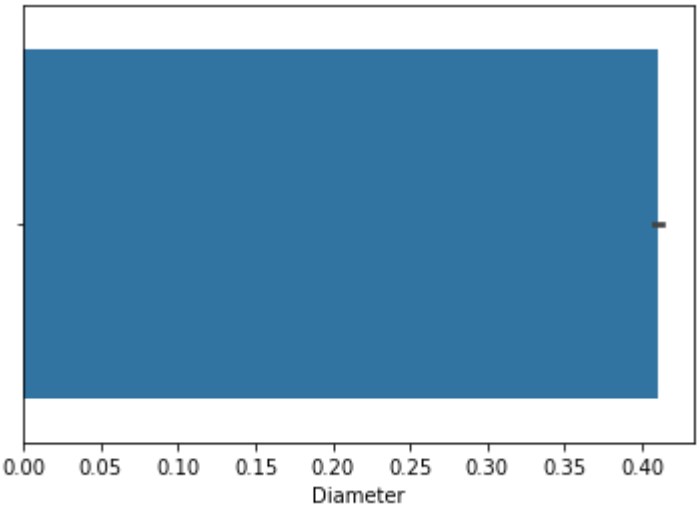
```
sns.boxplot(x=df['Diameter'])
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f579f1e90d0>
```



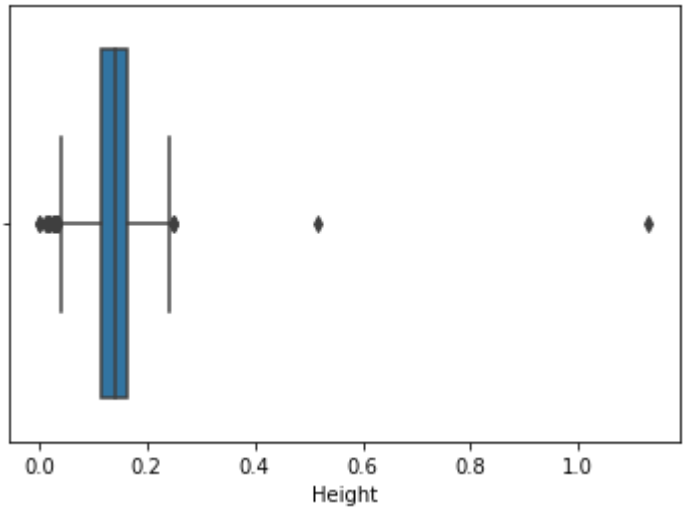

```
tenth_per = np.percentile(df['Diameter'], 10)
nine_per  = np.percentile(df['Diameter'], 90)

df['Diameter'] = np.where(df['Diameter'] < tenth_per, tenth_per, df['Diameter'])
df['Diameter'] = np.where(df['Diameter'] > nine_per, nine_per, df['Diameter'])


sns.barplot(x=df['Diameter'])
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f579d9c2250>
```



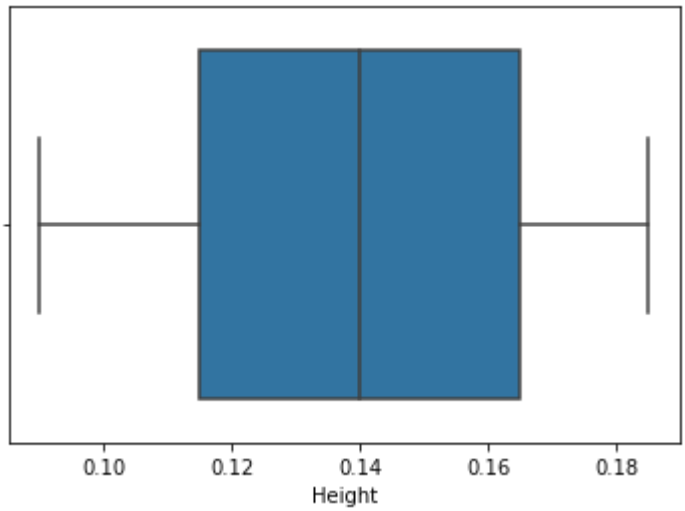

```
sns.boxplot(x=df['Height'])
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f579d9b9410>
```




```
tenth_per = np.percentile(df['Height'], 10)
nine_per  = np.percentile(df['Height'], 90)

df['Height'] = np.where(df['Height'] < tenth_per, tenth_per, df['Height'])
df['Height'] = np.where(df['Height'] > nine_per, nine_per, df['Height'])


sns.boxplot(x=df['Height'])
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f579d918ed0>
```



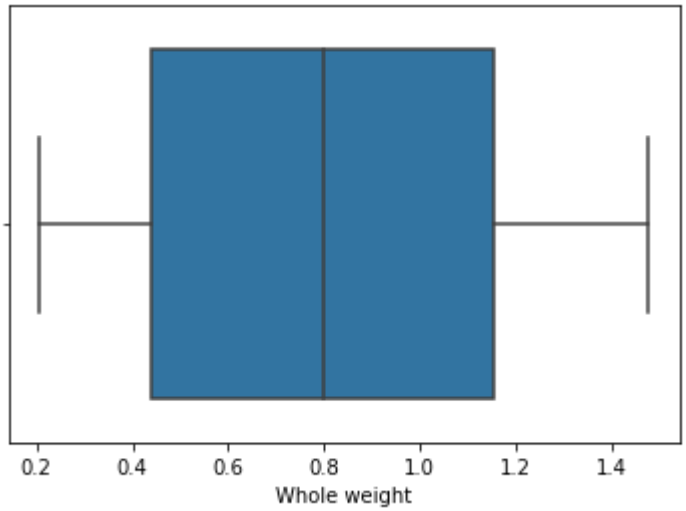

```
sns.boxplot(x=df['Whole weight'])
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f579d869410>
```

```
tenth_per = np.percentile(df['Whole weight'], 10)
nine_per  = np.percentile(df['Whole weight'], 90)

df['Whole weight'] = np.where(df['Whole weight'] < tenth_per, tenth_per, df['Whole weight'
df['Whole weight'] = np.where(df['Whole weight'] > nine_per, nine_per, df['Whole weight'])
```

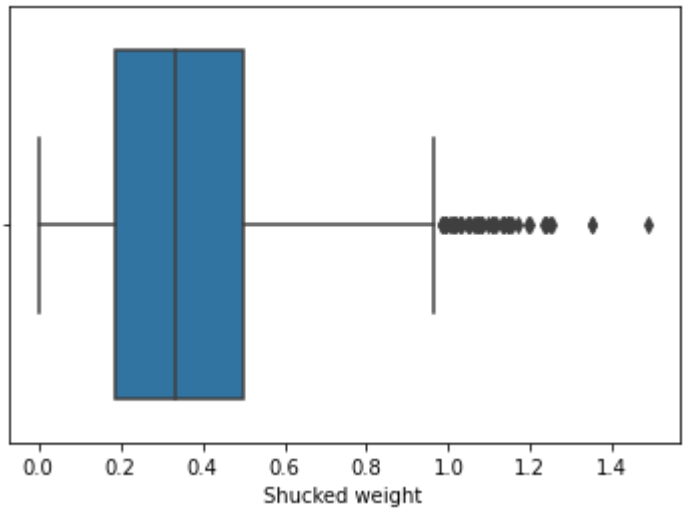
```
sns.boxplot(x=df['Whole weight'])
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f579d992ad0>
```




```
sns.boxplot(x=df['Shucked weight'])
```

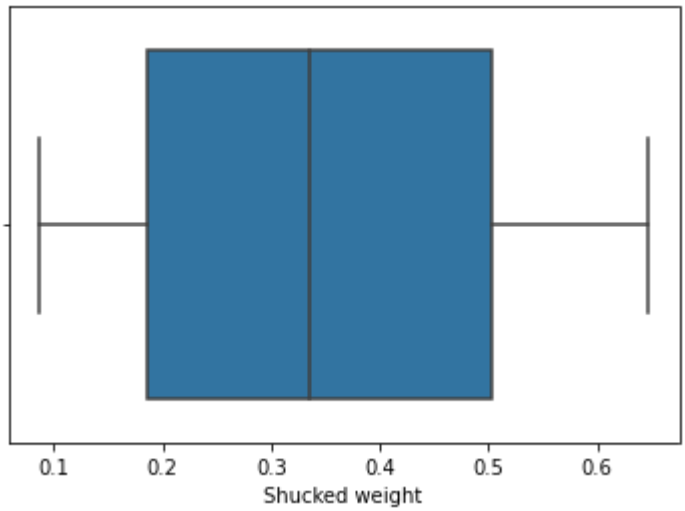
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f579d7ccf10>
```




```
tenth_per = np.percentile(df['Shucked weight'], 10)
nine_per  = np.percentile(df['Shucked weight'], 90)

df['Shucked weight'] = np.where(df['Shucked weight'] < tenth_per, tenth_per, df['Shucked w
df['Shucked weight'] = np.where(df['Shucked weight'] > nine_per, nine_per, df['Shucked wei
```

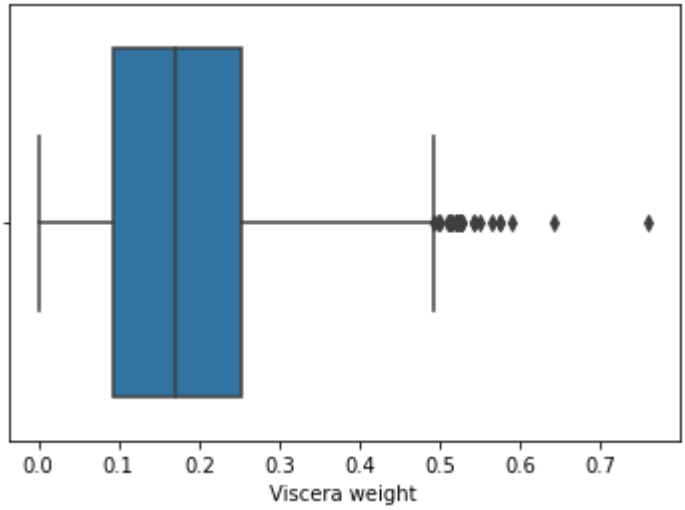
```
sns.boxplot(x=df['Shucked weight'])
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f579d740e90>
```



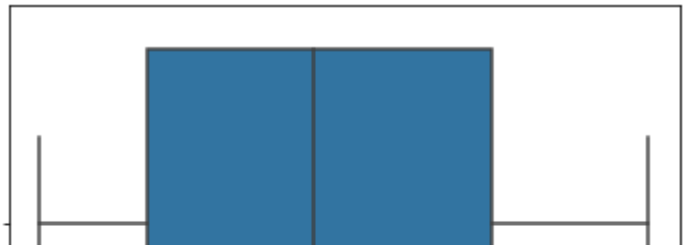

```
sns.boxplot(x=df['Viscera weight'])
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f579d7b99d0>
```



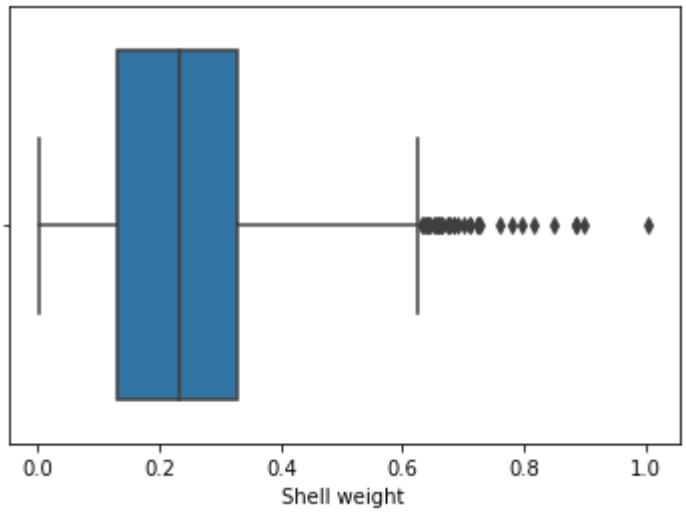

```
tenth_per = np.percentile(df['Viscera weight'], 10)
nine_per  = np.percentile(df['Viscera weight'], 90)

df['Viscera weight'] = np.where(df['Viscera weight'] < tenth_per, tenth_per, df['Viscera w
df['Viscera weight'] = np.where(df['Viscera weight'] > nine_per, nine_per, df['Viscera wei
sns.boxplot(x=df['Viscera weight'])
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f579d685210>
```



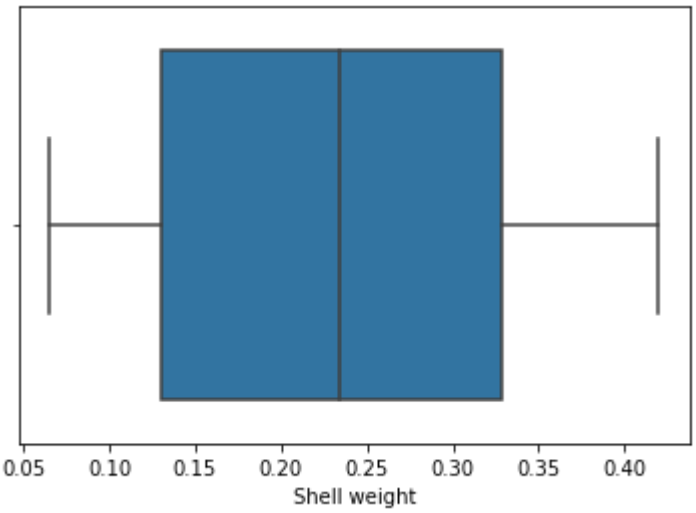
```
sns.boxplot(df['Shell weight'])
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/_decorators.py:43: FutureWarning: Pas
  FutureWarning
<matplotlib.axes._subplots.AxesSubplot at 0x7f579d5ff150>
```



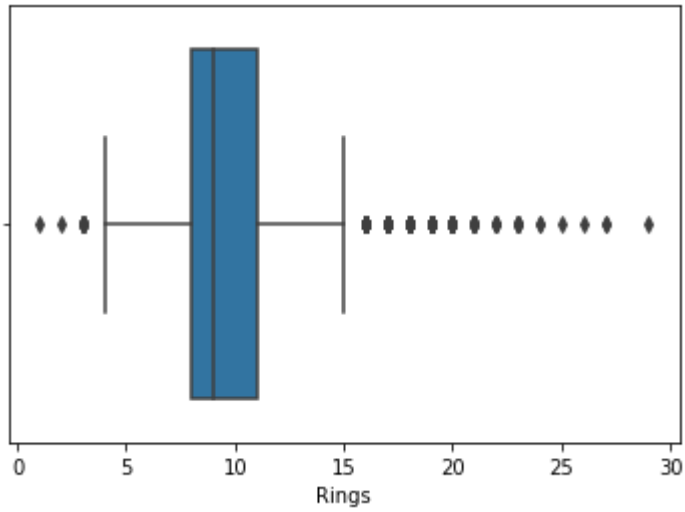

```
tenth_per = np.percentile(df['Shell weight'], 10)
nine_per  = np.percentile(df['Shell weight'], 90)

df['Shell weight'] = np.where(df['Shell weight'] < tenth_per, tenth_per, df['Shell weight'
df['Shell weight'] = np.where(df['Shell weight'] > nine_per, nine_per, df['Shell weight'])
sns.boxplot(df['Shell weight'])
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/_decorators.py:43: FutureWarning: Pas
  FutureWarning
<matplotlib.axes._subplots.AxesSubplot at 0x7f579d569a10>
```



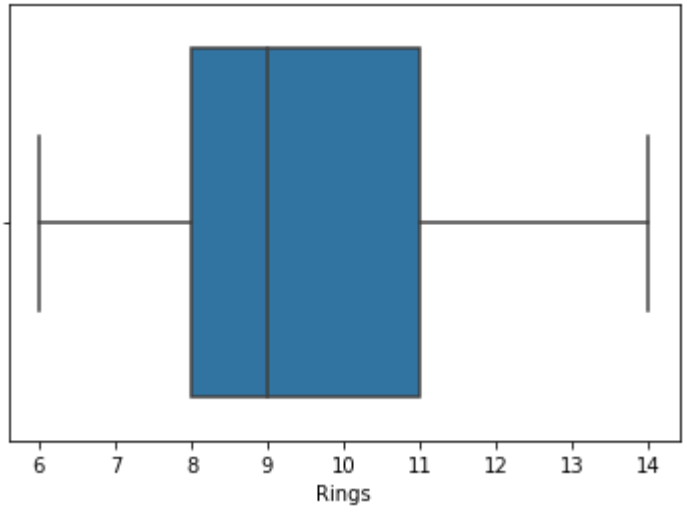
```
sns.boxplot(df['Rings'])
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/_decorators.py:43: FutureWarning: Pas
  FutureWarning
<matplotlib.axes._subplots.AxesSubplot at 0x7f579d556f50>
```




```
tenth_per = np.percentile(df['Rings'], 10)
nine_per  = np.percentile(df['Rings'], 90)

df['Rings'] = np.where(df['Rings'] < tenth_per, tenth_per, df['Rings'])
df['Rings'] = np.where(df['Rings'] > nine_per, nine_per, df['Rings'])
sns.boxplot(df['Rings'])
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/_decorators.py:43: FutureWarning: Pas
  FutureWarning
<matplotlib.axes._subplots.AxesSubplot at 0x7f579d6b5d10>
```




```
df.describe()
```

1 to 8 of 8 entries Filter

index	Sex	Length	Diameter	Height
count	4177.0	4177.0	4177.0	4177.0
mean	0.9470912137898013	0.5274910222647833	0.4104658846061767	0.13970074215944456
std	0.8222404151746827	0.09987349027498059	0.08371269310971387	0.031559391524783625
min	0.0	0.355	0.265	0.09
25%	0.0	0.45	0.35	0.115
50%	1.0	0.545	0.425	0.14
75%	2.0	0.615	0.48	0.165
max	2.0	0.66	0.5220000000000005	0.185

```
df.head()
```

	Sex	Length	Diameter	Height	Whole weight	Shucked weight	Viscera weight	Shell weight	Rings
0	0	0.455	0.365	0.095	0.5140	0.2245	0.1010	0.150	14.0
1	0	0.355	0.265	0.090	0.2255	0.0995	0.0485	0.070	7.0
2	2	0.530	0.420	0.135	0.6770	0.2565	0.1415	0.210	9.0
3	0	0.440	0.365	0.125	0.5160	0.2155	0.1140	0.155	10.0
4	1	0.355	0.265	0.090	0.2050	0.0895	0.0433	0.065	7.0

Outlier treatment

```
df['Age'] = df['Rings'] + 2.5
```

```
df.head()
```

	Sex	Length	Diameter	Height	Whole weight	Shucked weight	Viscera weight	Shell weight	Rings	Age
0	0	0.455	0.365	0.095	0.5140	0.2245	0.1010	0.150	14.0	16.5
1	0	0.355	0.265	0.090	0.2255	0.0995	0.0485	0.070	7.0	9.5
2	2	0.530	0.420	0.135	0.6770	0.2565	0.1415	0.210	9.0	11.5
3	0	0.440	0.365	0.125	0.5160	0.2155	0.1140	0.155	10.0	12.5
4	1	0.355	0.265	0.090	0.2050	0.0895	0.0433	0.065	7.0	9.5

```
X = df.drop('Age', axis=1)
y = df['Age']
```

```
from sklearn.model_selection import train_test_split
X_train,X_test,y_train,y_test  = train_test_split(X, y, test_size=0.3, random_state=101)
```

```
X_train.shape
```

```
(2923, 9)
```

```
X_test.shape
```

```
(1254, 9)
```

```
y_train.shape
```

```
(2923,)
```

```
y_test.shape
```

```
(1254,)
```

```
from sklearn.linear_model import LinearRegression
model1 = LinearRegression()
model1.fit(X_train,y_train)
```

```
LinearRegression()
```

```
y_pred1 = model1.predict(X_test)
```

```
y_pred1
```

```
array([12.5,  9.5, 12.5, ...,  8.5, 16.5, 12.5])
```

```
from sklearn.metrics import mean_squared_error, mean_absolute_error, r2_score
```

```
print(mean_absolute_error( y_test, y_pred1))
print(mean_squared_error(y_test, y_pred1))
```

```
7.592721418808088e-16
1.4544229767711159e-30
```

```
print(r2_score( y_test,y_pred1))
```

```
1.0
```

```
from sklearn.ensemble import RandomForestRegressor
model2 = RandomForestRegressor(n_estimators=500)
model2.fit(X_train, y_train)
```

```
RandomForestRegressor(n_estimators=500)
```

```
y_pred2 = model2.predict(X_test)
```

```
y_pred2
```

```
array([12.5,  9.5, 12.5, ...,  8.5, 16.5, 12.5])
```

```
print(r2_score( y_test,y_pred2))
```

```
1.0
```