ASSIGNMENT-3

NAME:Aravinth K

Reg No:73771921108

Importing the necessary Libraries

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

```
pwd
```

```
'C:\\Users\\harih\\OneDrive\\Desktop\\IBMgithub\\IBM-Project-54753-1662453739
\\Assignments\\HARIHARAN M'
```

Loading the dataset

```
df=pd.read_csv('abalone.csv')
```

```
df.head(10)
```

	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

	Sex	Length	Diameter	Height	Whole weight	Shucked weight	Viscera weight	Shell weight	Rings
5	I	0.425	0.300	0.095	0.3515	0.1410	0.0775	0.120	8
6	F	0.530	0.415	0.150	0.7775	0.2370	0.1415	0.330	20
7	F	0.545	0.425	0.125	0.7680	0.2940	0.1495	0.260	16
8	M	0.475	0.370	0.125	0.5095	0.2165	0.1125	0.165	9
9	F	0.550	0.440	0.150	0.8945	0.3145	0.1510	0.320	19

```
Age=1.5+df.Rings
df["Age"]=Age
df=df.rename(columns = {'Whole weight':'Whole_weight','Shucked weight':
'Shucked_weight','Viscera weight': 'Viscera_weight',
                                'Shell weight': 'Shell_weight'})
df=df.drop(columns=["Rings"],axis=1)
df.head()
```

	Sex	Length	Diameter	Height	Whole_weight	Shucked_weight	Viscera_weight	Shell_weight	Age
0	M	0.455	0.365	0.095	0.5140	0.2245	0.1010	0.150	16.5
1	M	0.350	0.265	0.090	0.2255	0.0995	0.0485	0.070	8.5
2	F	0.530	0.420	0.135	0.6770	0.2565	0.1415	0.210	10.5
3	M	0.440	0.365	0.125	0.5160	0.2155	0.1140	0.155	11.5
4	I	0.330	0.255	0.080	0.2050	0.0895	0.0395	0.055	8.5

```
df.shape
```

```
(4177, 9)
```

```
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   Age             4177 non-null   float64
dtypes: float64(8), object(1)
memory usage: 293.8+ KB
```

Pre processing

```
df.isnull().any()
```

```
Sex               False
Length            False
Diameter          False
Height            False
Whole_weight      False
Shucked_weight    False
Viscera_weight    False
Shell_weight      False
Age               False
dtype: bool
```

```
df.head(10)
```

	Sex	Length	Diameter	Height	Whole_weight	Shucked_weight	Viscera_weight	Shell_weight	Age
0	M	0.455	0.365	0.095	0.5140	0.2245	0.1010	0.150	16.5
1	M	0.350	0.265	0.090	0.2255	0.0995	0.0485	0.070	8.5
2	F	0.530	0.420	0.135	0.6770	0.2565	0.1415	0.210	10.5

	Sex	Length	Diameter	Height	Whole_weight	Shucked_weight	Viscera_weight	Shell_weight	Age
3	M	0.440	0.365	0.125	0.5160	0.2155	0.1140	0.155	11.5
4	I	0.330	0.255	0.080	0.2050	0.0895	0.0395	0.055	8.5
5	I	0.425	0.300	0.095	0.3515	0.1410	0.0775	0.120	9.5
6	F	0.530	0.415	0.150	0.7775	0.2370	0.1415	0.330	21.5
7	F	0.545	0.425	0.125	0.7680	0.2940	0.1495	0.260	17.5
8	M	0.475	0.370	0.125	0.5095	0.2165	0.1125	0.165	10.5
9	F	0.550	0.440	0.150	0.8945	0.3145	0.1510	0.320	20.5

Descriptive statistics

```
df.describe()
```

	Length	Diameter	Height	Whole_weight	Shucked_weight	Viscera_weight	Shell_weight	Age
count	4177.000000	4177.000000	4177.000000	4177.000000	4177.000000	4177.000000	4177.000000	4177.000000
mean	0.523992	0.407881	0.139516	0.828742	0.359367	0.180594	0.238831	11.433684
std	0.120093	0.099240	0.041827	0.490389	0.221963	0.109614	0.139203	3.224169
min	0.075000	0.055000	0.000000	0.002000	0.001000	0.000500	0.001500	2.500000

	Length	Diameter	Height	Whole_weight	Shucked_weight	Viscera_weight	Shell_weight	Age
25%	0.450000	0.350000	0.115000	0.441500	0.186000	0.093500	0.130000	9.500000
50%	0.545000	0.425000	0.140000	0.799500	0.336000	0.171000	0.234000	10.500000
75%	0.615000	0.480000	0.165000	1.153000	0.502000	0.253000	0.329000	12.500000
max	0.815000	0.650000	1.130000	2.825500	1.488000	0.760000	1.005000	30.500000

```
df.Sex.unique()
```

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

```
df.Sex.value_counts()
```

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

Visualization

```
sns.displot(df.Length)
```

```
<seaborn.axisgrid.FacetGrid at 0x2995ebcbee0>
```

```
sns.displot(df.Diameter)
```

```
<seaborn.axisgrid.FacetGrid at 0x2996423e310>
```

```
sns.displot(df.Height)
```

```
<seaborn.axisgrid.FacetGrid at 0x2996462ad90>
```

```
sns.displot(df.Whole_weight)
```

```
<seaborn.axisgrid.FacetGrid at 0x2996469ab20>
```

```
sns.displot(df.Shucked_weight)
```

```
<seaborn.axisgrid.FacetGrid at 0x299648a37f0>
```

```
sns.displot(df.Viscera_weight)
```

```
<seaborn.axisgrid.FacetGrid at 0x29964804e50>
```

```
sns.displot(df.Shell_weight)
```

```
<seaborn.axisgrid.FacetGrid at 0x299645dec10>
```

```
sns.displot(df.Sex)
```

```
<seaborn.axisgrid.FacetGrid at 0x29964861310>
```

```
sns.barplot(df.Sex.value_counts().index,df.Sex.value_counts())
```

```
C:\Users\harih\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureW
arning: Pass the following variables as keyword args: x, y. From version 0.12
, the only valid positional argument will be `data`, and passing other argume
nts without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

```
<AxesSubplot:ylabel='Sex'>
```

```
sns.lineplot(df.Sex,df.Length)
```

```
C:\Users\harih\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureW
arning: Pass the following variables as keyword args: x, y. From version 0.12
, the only valid positional argument will be `data`, and passing other argume
nts without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

```
<AxesSubplot:xlabel='Sex', ylabel='Length'>
```

```
sns.scatterplot(df.Sex,df.Length)
```

```
C:\Users\harih\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureW
arning: Pass the following variables as keyword args: x, y. From version 0.12
, the only valid positional argument will be `data`, and passing other argume
nts without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

```
<AxesSubplot:xlabel='Sex', ylabel='Length'>
```

```
sns.lineplot(df.Length,df.Diameter)
```

```
C:\Users\harih\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureW
arning: Pass the following variables as keyword args: x, y. From version 0.12
, the only valid positional argument will be `data`, and passing other argume
nts without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

```
<AxesSubplot:xlabel='Length', ylabel='Diameter'>
```

```
sns.scatterplot(df.Length,df.Diameter)
```

```
C:\Users\harih\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureW
arning: Pass the following variables as keyword args: x, y. From version 0.12
, the only valid positional argument will be `data`, and passing other argume
nts without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

```
<AxesSubplot:xlabel='Length', ylabel='Diameter'>
```

```
sns.lineplot(df.Length,df.Height)
```

```
C:\Users\harih\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureW
arning: Pass the following variables as keyword args: x, y. From version 0.12
, the only valid positional argument will be `data`, and passing other argume
nts without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

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

```
sns.scatterplot(df.Length,df.Height)
```

```
C:\Users\harih\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureW
arning: Pass the following variables as keyword args: x, y. From version 0.12
, the only valid positional argument will be `data`, and passing other argume
nts without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

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

```
sns.lineplot(df.Length,df.Whole_weight)
```

```
C:\Users\harih\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureW
arning: Pass the following variables as keyword args: x, y. From version 0.12
, the only valid positional argument will be `data`, and passing other argume
nts without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

```
<AxesSubplot:xlabel='Length', ylabel='Whole_weight'>
```

```
sns.scatterplot(df.Length,df.Whole_weight)
```

```
C:\Users\harih\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureW
arning: Pass the following variables as keyword args: x, y. From version 0.12
, the only valid positional argument will be `data`, and passing other argume
nts without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

```
<AxesSubplot:xlabel='Length', ylabel='Whole_weight'>
```

```
sns.lineplot(df.Length,df.Shucked_weight)
```

```
C:\Users\harih\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureW
arning: Pass the following variables as keyword args: x, y. From version 0.12
, the only valid positional argument will be `data`, and passing other argume
nts without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

```
<AxesSubplot:xlabel='Length', ylabel='Shucked_weight'>
```

```
sns.scatterplot(df.Length,df.Shucked_weight)
```

```
C:\Users\harih\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureW
arning: Pass the following variables as keyword args: x, y. From version 0.12
, the only valid positional argument will be `data`, and passing other argume
nts without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

```
<AxesSubplot:xlabel='Length', ylabel='Shucked_weight'>
```

```
sns.lineplot(df.Length,df.Viscera_weight)
```

```
C:\Users\harih\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureW
arning: Pass the following variables as keyword args: x, y. From version 0.12
, the only valid positional argument will be `data`, and passing other argume
nts without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

```
<AxesSubplot:xlabel='Length', ylabel='Viscera_weight'>
```

```
sns.scatterplot(df.Length,df.Viscera_weight)
```

```
C:\Users\harih\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureW
arning: Pass the following variables as keyword args: x, y. From version 0.12
, the only valid positional argument will be `data`, and passing other argume
nts without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

```
<AxesSubplot:xlabel='Length', ylabel='Viscera_weight'>
```

```
sns.lineplot(df.Length,df.Shell_weight)
```

```
C:\Users\harih\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureW
arning: Pass the following variables as keyword args: x, y. From version 0.12
, the only valid positional argument will be `data`, and passing other argume
nts without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

```
<AxesSubplot:xlabel='Length', ylabel='Shell_weight'>
```

```
sns.scatterplot(df.Length,df.Shell_weight)
```

```
C:\Users\harih\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureW
arning: Pass the following variables as keyword args: x, y. From version 0.12
, the only valid positional argument will be `data`, and passing other argume
nts without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

```
<AxesSubplot:xlabel='Length', ylabel='Shell_weight'>
```

```
sns.lineplot(df.Diameter,df.Shell_weight)
```

```
C:\Users\harih\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureW
arning: Pass the following variables as keyword args: x, y. From version 0.12
, the only valid positional argument will be `data`, and passing other argume
nts without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

```
<AxesSubplot:xlabel='Diameter', ylabel='Shell_weight'>
```

```
sns.scatterplot(df.Diameter,df.Shell_weight)
```

```
C:\Users\harih\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureW
arning: Pass the following variables as keyword args: x, y. From version 0.12
, the only valid positional argument will be `data`, and passing other argume
nts without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

```
<AxesSubplot:xlabel='Diameter', ylabel='Shell_weight'>
```

```
sns.lineplot(df.Diameter,df.Whole_weight)
```

```
C:\Users\harih\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureW
arning: Pass the following variables as keyword args: x, y. From version 0.12
, the only valid positional argument will be `data`, and passing other argume
nts without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

```
<AxesSubplot:xlabel='Diameter', ylabel='Whole_weight'>
```

```
sns.scatterplot(df.Diameter,df.Whole_weight)
```

```
C:\Users\harih\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureW
arning: Pass the following variables as keyword args: x, y. From version 0.12
, the only valid positional argument will be `data`, and passing other argume
nts without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

```
<AxesSubplot:xlabel='Diameter', ylabel='Whole_weight'>
```

```
sns.lineplot(df.Shell_weight,df.Whole_weight)
```

```
C:\Users\harih\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureW
arning: Pass the following variables as keyword args: x, y. From version 0.12
, the only valid positional argument will be `data`, and passing other argume
nts without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

```
<AxesSubplot:xlabel='Shell_weight', ylabel='Whole_weight'>
```

```
sns.scatterplot(df.Shell_weight,df.Whole_weight)
```

```
C:\Users\harih\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureW
arning: Pass the following variables as keyword args: x, y. From version 0.12
, the only valid positional argument will be `data`, and passing other argume
nts without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

```
<AxesSubplot:xlabel='Shell_weight', ylabel='Whole_weight'>
```

```
sns.lineplot(df.Shucked_weight,df.Viscera_weight)
```

```
C:\Users\harih\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureW
arning: Pass the following variables as keyword args: x, y. From version 0.12
, the only valid positional argument will be `data`, and passing other argume
nts without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

```
<AxesSubplot:xlabel='Shucked_weight', ylabel='Viscera_weight'>
```

```
sns.scatterplot(df.Shucked_weight,df.Viscera_weight)
```

```
C:\Users\harih\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureW
arning: Pass the following variables as keyword args: x, y. From version 0.12
, the only valid positional argument will be `data`, and passing other argume
nts without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

```
<AxesSubplot:xlabel='Shucked_weight', ylabel='Viscera_weight'>
```

```
df.hist(figsize=(15,15))
```

```
array([[<AxesSubplot:title={'center':'Length'}>,
        <AxesSubplot:title={'center':'Diameter'}>,
        <AxesSubplot:title={'center':'Height'}>],
       [<AxesSubplot:title={'center':'Whole_weight'}>,
        <AxesSubplot:title={'center':'Shucked_weight'}>,
        <AxesSubplot:title={'center':'Viscera_weight'}>],
       [<AxesSubplot:title={'center':'Shell_weight'}>,
        <AxesSubplot:title={'center':'Age'}>, <AxesSubplot:>]],
      dtype=object)
```

```
sns.pairplot(df)
```

```
<seaborn.axisgrid.PairGrid at 0x2996786f580>
```

Handling Outliers

```
sns.boxplot(df.Length)
```

```
C:\Users\harih\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureW
arning: Pass the following variable as a keyword arg: x. From version 0.12, t
he only valid positional argument will be `data`, and passing other arguments
without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

```
<AxesSubplot:xlabel='Length'>
```

```
q1=df.Length.quantile(0.25)  #(Q1)
q3=df.Length.quantile(0.75)  #(Q3)
```

```
IQR=q3-q1
```

```
upper_limit= q3 + 1.5*IQR
```

```
lower_limit= q1 - 1.5*IQR
```

```
upper_limit
```

```
0.8624999999999999
```

```
lower_limit
```

```
0.20250000000000004
```

```
df.median()
```

```
C:\Users\harih\AppData\Local\Temp/ipykernel_16804/530051474.py:1: FutureWarni
ng: Dropping of nuisance columns in DataFrame reductions (with 'numeric_only=
None') is deprecated; in a future version this will raise TypeError.  Select
only valid columns before calling the reduction.
  df.median()
```

```
Length             0.5450
Diameter           0.4250
Height             0.1400
Whole_weight       0.7995
Shucked_weight     0.3360
Viscera_weight     0.1710
Shell_weight       0.2340
Age               10.5000
dtype: float64
```

```
df['Length']= np.where(df['Length']<lower_limit,0.5450,df['Length'])
```

```
sns.boxplot(df.Length)
```

```
C:\Users\harih\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureW
arning: Pass the following variable as a keyword arg: x. From version 0.12, t
he only valid positional argument will be `data`, and passing other arguments
without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

```
<AxesSubplot:xlabel='Length'>
```

```
sns.boxplot(df.Height)
```

```
C:\Users\harih\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureW
arning: Pass the following variable as a keyword arg: x. From version 0.12, t
```

```
he only valid positional argument will be `data`, and passing other arguments
without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

```
<AxesSubplot:xlabel='Height'>
```

```
q1=df.Height.quantile(0.25)  #(Q1
q3=df.Height.quantile(0.75)  #(Q3)
```

```
IQR=q3-q1
```

```
upper_limit= q3 + 1.5*IQR
```

```
lower_limit= q1 - 1.5*IQR
```

```
upper_limit
```

```
0.24000000000000002
```

```
lower_limit
```

```
0.039999999999999994
```

```
df.median()
```

```
C:\Users\harih\AppData\Local\Temp/ipykernel_16804/530051474.py:1: FutureWarni
ng: Dropping of nuisance columns in DataFrame reductions (with 'numeric_only=
None') is deprecated; in a future version this will raise TypeError.  Select
only valid columns before calling the reduction.
  df.median()
```

```
Length             0.5450
Diameter           0.4250
Height             0.1400
Whole_weight       0.7995
Shucked_weight     0.3360
Viscera_weight     0.1710
Shell_weight       0.2340
Age               10.5000
dtype: float64
```

```
df['Height']= np.where(df['Height']<lower_limit,0.1400,df['Height'])
```

```
df['Height']= np.where(df['Height']>upper_limit,0.1400,df['Height;])
```

```
sns.boxplot(df.Height)
```

```
C:\Users\harih\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureW
arning: Pass the following variable as a keyword arg: x. From version 0.12, t
he only valid positional argument will be `data`, and passing other arguments
without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

```
<AxesSubplot:xlabel='Height'>
```

```
sns.boxplot(df.Diameter)
```

```
C:\Users\harih\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureW
arning: Pass the following variable as a keyword arg: x. From version 0.12, t
he only valid positional argument will be `data`, and passing other arguments
without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

```
<AxesSubplot:xlabel='Diameter'>
```

```
q1=df.Diameter.quantile(0.25)  #(Q1)
q3=df.Diameter.quantile(0.75)  #(Q3)
```

```
IQR=q3-q1
```

```
upper_limit= q3 + 1.5*IQR
```

```
lower_limit= q1 - 1.5*IQR
```

```
upper_limit
```

```
0.675
```

```
lower_limit
```

```
0.15499999999999997
```

```
df.median()
```

```
C:\Users\harih\AppData\Local\Temp/ipykernel_16804/530051474.py:1: FutureWarni
ng: Dropping of nuisance columns in DataFrame reductions (with 'numeric_only=
None') is deprecated; in a future version this will raise TypeError.  Select
only valid columns before calling the reduction.
  df.median()
```

```
Length             0.5450
Diameter           0.4250
Height             0.1400
Whole_weight       0.7995
Shucked_weight     0.3360
```

```
Viscera_weight     0.1710
Shell_weight       0.2340
Age               10.5000
dtype: float64
```

```
df['Diameter']= np.where(df['Diameter']<lower_limit, 0.4250,df['Diameter'])
```

```
sns.boxplot(df.Diameter)
```

```
C:\Users\harih\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureW
arning: Pass the following variable as a keyword arg: x. From version 0.12, t
he only valid positional argument will be `data`, and passing other arguments
without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

```
<AxesSubplot:xlabel='Diameter'>
```



```
sns.boxplot(df.Whole_weight)
```

```
C:\Users\harih\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureW
arning: Pass the following variable as a keyword arg: x. From version 0.12, t
he only valid positional argument will be `data`, and passing other arguments
without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

```
<AxesSubplot:xlabel='Whole_weight'>
```



```
q1=df.Whole_weight.quantile(0.25)  #(Q1)
q3=df.Whole_weight.quantile(0.75)  #(Q3)
```

```
IQR=q3-q1
```

```
upper_limit= q3 + 1.5*IQR
```

```
lower_limit= q1 - 1.5*IQR
```

```
upper_limit
```

```
2.22025
```

```
lower_limit
```

```
-0.62575
```

```
df.median()
```

```
C:\Users\harih\AppData\Local\Temp/ipykernel_16804/530051474.py:1: FutureWarni
ng: Dropping of nuisance columns in DataFrame reductions (with 'numeric_only=
```

```
None') is deprecated; in a future version this will raise TypeError.  Select
only valid columns before calling the reduction.
  df.median()
```

```
Length             0.5450
Diameter           0.4250
Height             0.1400
Whole_weight       0.7995
Shucked_weight     0.3360
Viscera_weight     0.1710
Shell_weight       0.2340
Age               10.5000
dtype: float64
```

```
df['Whole_weight']=
np.where(df['Whole_weight']>upper_limit,0.7995,df['Whole_weight'])
```

```
sns.boxplot(df.Whole_weight)
```

```
C:\Users\harih\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureW
arning: Pass the following variable as a keyword arg: x. From version 0.12, t
he only valid positional argument will be `data`, and passing other arguments
without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

```
<AxesSubplot:xlabel='Whole_weight'>
```

```
sns.boxplot(df.Shucked_weight)
```

```
C:\Users\harih\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureW
arning: Pass the following variable as a keyword arg: x. From version 0.12, t
he only valid positional argument will be `data`, and passing other arguments
without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

```
<AxesSubplot:xlabel='Shucked_weight'>
```

```
q1=df.Shucked_weight.quantile(0.25)  #(Q1)
q3=df.Shucked_weight.quantile(0.75)  #(Q3)
```

```
IQR=q3-q1
```

```
upper_limit= q3 + 1.5*IQR
```

```
lower_limit= q1 - 1.5*IQR
```

```
upper_limit
```

```
0.976

lower_limit


-0.288

df.median()
```

```
C:\Users\harih\AppData\Local\Temp/ipykernel_16804/530051474.py:1: FutureWarni
ng: Dropping of nuisance columns in DataFrame reductions (with 'numeric_only=
None') is deprecated; in a future version this will raise TypeError.  Select
only valid columns before calling the reduction.
  df.median()
```

```
Length             0.5450
Diameter           0.4250
Height             0.1400
Whole_weight       0.7995
Shucked_weight     0.3360
Viscera_weight     0.1710
Shell_weight       0.2340
Age               10.5000
dtype: float64
```

```
df['Shucked_weight']=
np.where(df['Shucked_weight']>upper_limit,0.3360,df['Shucked_weight'])


sns.boxplot(df.Shucked_weight)
```

```
C:\Users\harih\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureW
arning: Pass the following variable as a keyword arg: x. From version 0.12, t
he only valid positional argument will be `data`, and passing other arguments
without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

```
<AxesSubplot:xlabel='Shucked_weight'>


sns.boxplot(df.Viscera_weight)
```

```
C:\Users\harih\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureW
arning: Pass the following variable as a keyword arg: x. From version 0.12, t
he only valid positional argument will be `data`, and passing other arguments
without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

```
<AxesSubplot:xlabel='Viscera_weight'>


q1=df.Viscera_weight.quantile(0.25)  #(Q1)
q3=df.Viscera_weight.quantile(0.75)  #(Q3)
```

```
IQR=q3-q1
```

```
upper_limit= q3 + 1.5*IQR
```

```
lower_limit= q1 - 1.5*IQR
```

```
upper_limit
```

```
0.49225
```

```
lower_limit
```

```
-0.14575000000000002
```

```
df.median()
```

```
C:\Users\harih\AppData\Local\Temp/ipykernel_16804/530051474.py:1: FutureWarni
ng: Dropping of nuisance columns in DataFrame reductions (with 'numeric_only=
None') is deprecated; in a future version this will raise TypeError.  Select
only valid columns before calling the reduction.
  df.median()
```

```
Length             0.5450
Diameter           0.4250
Height             0.1400
Whole_weight       0.7995
Shucked_weight     0.3360
Viscera_weight     0.1710
Shell_weight       0.2340
Age               10.5000
dtype: float64
```

```
df['Viscera_weight']=
np.where(df['Viscera_weight']>upper_limit,0.1710,df['Viscera_weight'])
```

```
sns.boxplot(df.Viscera_weight)
```

```
C:\Users\harih\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureW
arning: Pass the following variable as a keyword arg: x. From version 0.12, t
he only valid positional argument will be `data`, and passing other arguments
without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

```
<AxesSubplot:xlabel='Viscera_weight'>
```

```
sns.boxplot(df.Shell_weight)
```

```
C:\Users\harih\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureW
arning: Pass the following variable as a keyword arg: x. From version 0.12, t
```

```
he only valid positional argument will be `data`, and passing other arguments
without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

```
<AxesSubplot:xlabel='Shell_weight'>
```

```
q1=df.Shell_weight.quantile(0.25)  #(Q1)
q3=df.Shell_weight.quantile(0.75)  #(Q3)
```

```
IQR=q3-q1
```

```
upper_limit= q3 + 1.5*IQR
```

```
lower_limit= q1 - 1.5*IQR
```

```
upper_limit
```

```
0.6275
```

```
lower_limit
```

```
-0.16849999999999998
```

```
df.median()
```

```
C:\Users\harih\AppData\Local\Temp/ipykernel_16804/530051474.py:1: FutureWarni
ng: Dropping of nuisance columns in DataFrame reductions (with 'numeric_only=
None') is deprecated; in a future version this will raise TypeError.  Select
only valid columns before calling the reduction.
  df.median()
```

```
Length             0.5450
Diameter           0.4250
Height             0.1400
Whole_weight       0.7995
Shucked_weight     0.3360
Viscera_weight     0.1710
Shell_weight       0.2340
Age               10.5000
dtype: float64
```

```
df['Shell_weight']=
np.where(df['Shell_weight']>upper_limit,0.2340,df['Shell_weight'])
```

```
sns.boxplot(df.Shell_weight)
```

```
C:\Users\harih\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureW
arning: Pass the following variable as a keyword arg: x. From version 0.12, t
```

```
he only valid positional argument will be `data`, and passing other arguments
without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

```
<AxesSubplot:xlabel='Shell_weight'>
```

Label Encoder

```
from sklearn.preprocessing import LabelEncoder
```

```
le=LabelEncoder()
```

```
df.Sex=le.fit_transform(df.Sex)
```

```
df.head(10)
```

	Sex	Length	Diameter	Height	Whole_weight	Shucked_weight	Viscera_weight	Shell_weight	Age
0	2	0.455	0.365	0.095	0.5140	0.2245	0.1010	0.150	16.5
1	2	0.350	0.265	0.090	0.2255	0.0995	0.0485	0.070	8.5
2	0	0.530	0.420	0.135	0.6770	0.2565	0.1415	0.210	10.5
3	2	0.440	0.365	0.125	0.5160	0.2155	0.1140	0.155	11.5
4	1	0.330	0.255	0.080	0.2050	0.0895	0.0395	0.055	8.5
5	1	0.425	0.300	0.095	0.3515	0.1410	0.0775	0.120	9.5
6	0	0.530	0.415	0.150	0.7775	0.2370	0.1415	0.330	21.5
7	0	0.545	0.425	0.125	0.7680	0.2940	0.1495	0.260	17.5

	Sex	Length	Diameter	Height	Whole_weight	Shucked_weight	Viscera_weight	Shell_weight	Age
8	2	0.475	0.370	0.125	0.5095	0.2165	0.1125	0.165	10.5
9	0	0.550	0.440	0.150	0.8945	0.3145	0.1510	0.320	20.5

```
df.corr()
```

	Sex	Length	Diameter	Height	Whole_weight	Shucked_weight	Viscera_weight	Shell_weight	Age
Sex	1.000000	-0.032999	-0.035934	-0.041976	-0.028695	-0.020533	-0.037293	-0.039365	-0.034627
Length	-0.032999	1.000000	0.973234	0.871633	0.903973	0.878470	0.883640	0.885617	0.516360
Diameter	-0.035934	0.973234	1.000000	0.871498	0.898864	0.869731	0.876330	0.888250	0.528001
Height	-0.041976	0.871633	0.871498	1.000000	0.874416	0.821263	0.856081	0.880501	0.594727
Whole_weight	-0.028695	0.903973	0.898864	0.874416	1.000000	0.939880	0.944322	0.928024	0.540937
Shucked_weight	-0.020533	0.878470	0.869731	0.821263	0.939880	1.000000	0.901219	0.857370	0.431656
Viscera_weight	-0.037293	0.883640	0.876330	0.856081	0.944322	0.901219	1.000000	0.887565	0.507776

	Sex	Length	Diameter	Height	Whole_weight	Shucked_weight	Viscera_weight	Shell_weight	Age
Shell_weight	-0.039365	0.885617	0.888250	0.880501	0.928024	0.857370	0.887565	1.000000	0.606431
Age	-0.034627	0.516360	0.528001	0.594727	0.540937	0.431656	0.507776	0.606431	1.000000

```
plt.figure(figsize=(15,8))
sns.heatmap(df.corr(),annot=True)
```

```
<AxesSubplot:>
```



```
df.corr().Age.sort_values(ascending=False)
```

```
Age               1.000000
Shell_weight      0.606431
Height            0.594727
Whole_weight      0.540937
Diameter          0.528001
Length            0.516360
Viscera_weight    0.507776
Shucked_weight    0.431656
Sex              -0.034627
Name: Age, dtype: float64
```

```
df.head(
```

	Sex	Length	Diameter	Height	Whole_weight	Shucked_weight	Viscera_weight	Shell_weight	Age
0	2	0.455	0.365	0.095	0.5140	0.2245	0.1010	0.150	16.5
1	2	0.350	0.265	0.090	0.2255	0.0995	0.0485	0.070	8.5
2	0	0.530	0.420	0.135	0.6770	0.2565	0.1415	0.210	10.5

	Sex	Length	Diameter	Height	Whole_weight	Shucked_weight	Viscera_weight	Shell_weight	Age
3	2	0.440	0.365	0.125	0.5160	0.2155	0.1140	0.155	11.5
4	1	0.330	0.255	0.080	0.2050	0.0895	0.0395	0.055	8.5

X and Y split

```
y=df['Age']
y
```

```
0        16.5
1         8.5
2        10.5
3        11.5
4         8.5
        ...
4172    12.5
4173    11.5
4174    10.5
4175    11.5
4176    13.5
Name: Age, Length: 4177, dtype: float64
```

```
X=df.drop(columns=['Age'],axis=1)
X.head(10)
```

	Sex	Length	Diameter	Height	Whole_weight	Shucked_weight	Viscera_weight	Shell_weight
0	2	0.455	0.365	0.095	0.5140	0.2245	0.1010	0.150
1	2	0.350	0.265	0.090	0.2255	0.0995	0.0485	0.070
2	0	0.530	0.420	0.135	0.6770	0.2565	0.1415	0.210
3	2	0.440	0.365	0.125	0.5160	0.2155	0.1140	0.155
4	1	0.330	0.255	0.080	0.2050	0.0895	0.0395	0.055

	Sex	Length	Diameter	Height	Whole_weight	Shucked_weight	Viscera_weight	Shell_weight
5	1	0.425	0.300	0.095	0.3515	0.1410	0.0775	0.120
6	0	0.530	0.415	0.150	0.7775	0.2370	0.1415	0.330
7	0	0.545	0.425	0.125	0.7680	0.2940	0.1495	0.260
8	2	0.475	0.370	0.125	0.5095	0.2165	0.1125	0.165
9	0	0.550	0.440	0.150	0.8945	0.3145	0.1510	0.320

Scaling

```
from sklearn.preprocessing import MinMaxScaler
scale=MinMaxScaler()

X_scaled=pd.DataFrame(scale.fit_transform(X),columns=X.columns)
X_scaled.head()
```

	Sex	Length	Diameter	Height	Whole_weight	Shucked_weight	Viscera_weight	Shell_weight
0	1.0	0.409836	0.424242	0.275	0.231884	0.231726	0.204476	0.238172
1	1.0	0.237705	0.222222	0.250	0.101223	0.102125	0.097660	0.109864
2	0.0	0.532787	0.535354	0.475	0.305707	0.264904	0.286877	0.334403
3	1.0	0.385246	0.424242	0.425	0.232790	0.222395	0.230926	0.246191
4	0.5	0.204918	0.202020	0.200	0.091938	0.091757	0.079349	0.085806

Train Test Split

```
df.shape
```

```
(4177, 9)
```

```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=
train_test_split(X_scaled,y,test_size=0.2,random_state=0)
```

```
x_train.shape
```

```
(3341, 8)
```

```
x_test.shape
```

```
(836, 8)
```

```
y_train.shape
```

```
(3341,)
```

```
y_test.shape
```

```
(836,)
```

Model Building

Linear Regression

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

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

```
LinearRegression()
```

```
pred_test= lr.predict(x_test)
pred_test
```

```
array([13.80690384, 10.7601818 , 11.91266938,  7.74165577, 12.64640354,
       13.01902298,  9.31216917, 10.97864764, 10.20923554, 12.92069473,
       10.11949968,  8.40216639,  9.76181207, 10.38620015,  8.03417237,
       11.29885471,  9.24151631, 14.98455396, 13.10311588,  9.41490054,
        9.18125593,  8.45179656, 10.34075915,  9.24384733, 11.43509847,
       12.55855133,  9.91319508, 14.42316086, 11.71223636, 12.1205011 ,
        9.40211493,  9.87172948, 12.89129341, 16.67599634,  9.17428836,
```

10.49697682, 10.60909575, 11.98691159, 10.2029085 , 13.05440209,
13.12759813, 10.70654102, 12.77615623, 12.85314185, 13.51936135,
10.66760564, 10.98133112, 13.47861294, 13.63000606, 9.50624155,
12.47377656, 8.98233344, 10.17762269, 14.60384594, 10.77116619,
 9.17927021, 8.08496582, 8.90574462, 8.91635132, 8.74968419,
11.18490639, 10.79917287, 11.56382915, 9.74223427, 9.56411464,
13.93180286, 14.23135486, 14.11588799, 10.73885887, 15.78733251,
11.06119929, 13.45082024, 12.96777666, 10.82164   , 11.27119693,
10.50679686, 10.77294403, 11.08845824, 12.95452825, 9.53698564,
11.24610998, 8.0236658 , 9.37632552, 14.17192396, 11.70386473,
 9.66941609, 11.53008288, 14.41166642, 7.29702657, 9.05482723,
11.71175772, 12.21333058, 9.77130748, 6.56424892, 13.71222344,
 8.18781449, 11.49711147, 9.26884153, 15.51889861, 11.8602162 ,
11.91593506, 12.93740853, 11.94833644, 12.28670711, 7.82278415,
12.13473595, 9.35241095, 8.50569242, 9.57193784, 13.24596787,
18.95218178, 13.50110286, 12.62853953, 10.02703293, 14.17787387,
12.5621508 , 13.29486066, 13.12621809, 9.76382559, 11.42169873,
 8.89076078, 13.41758963, 8.50163795, 10.63896055, 12.37834304,
13.19585487, 12.51394541, 13.04427717, 9.40916396, 12.32441583,
12.04926934, 9.54357537, 11.17700894, 13.0505127 , 13.2256854 ,
12.43762735, 11.35042595, 10.63845249, 9.20319867, 15.13424128,
12.0164333 , 12.5713877 , 9.0236019 , 10.27617609, 13.16421152,
11.8069947 , 10.4553434 , 9.3564191 , 9.44023489, 8.76399393,
11.00836176, 15.82843433, 8.79536517, 12.18221518, 9.27512701,
 8.68330133, 12.91301968, 8.58294594, 14.24247208, 9.49441871,
12.12020577, 8.59202306, 11.56302436, 12.5538779 , 7.31995279,
14.01213518, 10.01122822, 8.56365544, 12.72270025, 11.34110648,
10.75142478, 9.25447876, 9.73016873, 12.23032821, 13.89078758,
12.25825704, 7.98855127, 10.32147078, 9.29234887, 12.22636942,
11.16125274, 12.13527407, 10.07514724, 10.61112663, 13.64548627,
11.52367746, 11.27650694, 10.32953904, 8.4766789 , 10.15885596,
11.75269723, 10.95261149, 10.62683643, 11.59555084, 11.94889148,
10.79564102, 9.72902619, 10.09861088, 7.49023182, 14.76800101,
10.79441947, 11.55179626, 18.5253285 , 11.29287193, 11.27410526,
13.58835542, 8.06614615, 13.95217783, 11.28459104, 13.61665395,
 7.92082169, 11.35547637, 12.13955191, 11.24183904, 12.79335724,
11.57433028, 11.04914375, 12.25416814, 7.81425418, 14.06800691,
12.53863693, 13.74035645, 9.97858091, 11.12715251, 13.66925709,
15.0784178 , 13.42902177, 10.71031075, 13.0824964 , 11.09433252,
10.93135934, 9.90163296, 7.72957225, 13.22136112, 9.01076243,
13.37447035, 9.43428542, 7.67997278, 10.37762157, 11.23029603,
10.05690291, 11.42463763, 12.40131039, 11.41441966, 11.81715146,
11.42875624, 12.50816994, 9.77953843, 11.89030729, 15.26426107,
13.79506105, 9.21513513, 14.43874165, 19.47627263, 12.25175959,
12.77093253, 9.17582256, 11.00182988, 11.78354077, 16.01181545,
12.20694708, 10.48082752, 12.82962982, 8.80279617, 12.75738752,
10.25472118, 9.97839527, 9.65563626, 9.35071337, 8.96721455,
11.29012784, 10.02709591, 9.696441 , 9.61683191, 8.41001813,
12.80376447, 13.9977188 , 10.51148793, 11.46307256, 11.56592187,
 9.41279068, 15.11106557, 9.54938521, 11.34966559, 11.536445 ,
12.07138851, 9.00893153, 12.06868774, 12.09850379, 13.8483699 ,
 9.22251272, 10.57826608, 7.64104106, 11.16934831, 9.17929865,
 9.64452266, 8.58557686, 8.47258835, 11.16894328, 15.84745106,

10.28864958, 9.37673548, 14.22988816, 11.48094093, 10.47148794,
10.01697235, 11.31103196, 11.27557892, 12.26264861, 10.96138875,
11.68826839, 12.65967708, 9.67618367, 11.26070226, 16.52462713,
13.47254621, 11.08509578, 8.49887538, 9.83863261, 11.87047817,
12.27597451, 11.36041968, 10.62381334, 12.58575043, 13.94513345,
10.83339099, 10.70887087, 12.27878962, 16.56286851, 12.05503805,
10.89149913, 13.27555067, 11.88720662, 13.16020504, 11.62050333,
13.72990855, 9.77571358, 9.72729834, 12.57917589, 12.3986216 ,
11.03683785, 11.14059976, 12.14099668, 12.68411753, 13.71847541,
 8.91256343, 7.89133508, 13.32148528, 14.41380229, 9.25331353,
 8.66915908, 10.4780174 , 14.43132097, 11.71780515, 12.69475699,
11.04224417, 12.58703017, 11.49510675, 9.82419761, 11.27096607,
10.05423913, 11.33890227, 9.97273039, 11.2267939 , 11.87763587,
 8.06591925, 8.65317539, 10.79767275, 9.453559 , 11.23066954,
12.45775879, 8.91475783, 12.96589848, 10.11485246, 13.90599852,
13.54697767, 14.55789748, 9.98271204, 12.19722653, 7.42838027,
10.39958499, 10.65451671, 11.48620481, 14.29330865, 8.26231782,
12.36163539, 16.71155077, 10.85224261, 9.60984543, 8.46205417,
17.96441475, 11.52585489, 11.13061133, 11.07568016, 12.2522251 ,
 9.17615028, 13.00014209, 10.40953832, 12.42979763, 12.02647835,
 9.67626229, 10.41295339, 8.86752593, 14.78442105, 9.50447537,
12.61121112, 8.4541589 , 14.01490474, 16.55015979, 11.02197996,
11.55085457, 9.44959499, 12.55054631, 12.11010251, 15.46998094,
12.52056265, 13.72313681, 12.84530911, 15.52170214, 12.43362296,
10.24481588, 10.8955776 , 12.1128833 , 12.53472271, 8.08794317,
10.5691944 , 11.86066197, 12.60430786, 12.72551875, 15.0173634 ,
10.93782827, 13.25811357, 10.44473985, 13.81671426, 10.64603563,
10.42522706, 13.61113514, 11.11106017, 11.09970591, 13.00363364,
 9.82241773, 8.66631264, 14.36423531, 10.1400024 , 8.51171084,
 9.00524228, 14.28604334, 7.91331587, 11.73104488, 11.93831402,
 6.96912671, 8.59681604, 11.23499011, 8.73014621, 4.27229336,
 9.64739904, 12.43348657, 11.47207361, 10.46508716, 13.37404965,
 9.1771865 , 15.74722809, 10.01274364, 11.55381225, 11.07476149,
14.17704797, 8.50911663, 18.34120143, 11.80171567, 10.03975293,
 8.75211898, 11.9293444 , 8.98252677, 15.94126231, 10.31262319,
10.61476998, 9.77734173, 9.4901215 , 9.21919125, 11.67178707,
11.61265991, 14.46934296, 10.09440319, 12.74709287, 15.62665447,
13.40222226, 12.03618508, 11.00417116, 9.59789191, 13.06100575,
 7.36626736, 10.91113598, 12.6674649 , 11.7403181 , 9.00264479,
 8.86932466, 10.12176723, 8.27106521, 12.81735832, 10.95183593,
 9.99839628, 12.82523413, 8.75175128, 13.36914109, 11.74100365,
12.33421622, 12.95460952, 15.5129496 , 9.7582816 , 7.73087612,
11.66686962, 10.72995716, 15.33911084, 15.77637128, 10.91546041,
 9.84483256, 12.1287358 , 13.07290984, 14.86773469, 11.85820809,
12.3149489 , 11.26509181, 11.97976154, 11.28508086, 9.13724401,
15.04792111, 10.66345124, 11.36994652, 10.35854875, 10.23486975,
 9.36785706, 9.63316964, 10.05306197, 12.15739154, 7.21220758,
11.3691406 , 13.75375036, 14.02835556, 10.76615235, 14.65910254,
 9.17636352, 11.83481951, 9.62803972, 13.43855229, 8.79616352,
11.55786899, 10.42127676, 9.97829228, 12.4776445 , 11.21453726,
12.07554627, 15.94172218, 9.9565626 , 15.04136528, 12.01449734,
 9.67325255, 10.13874736, 10.78711706, 12.10787363, 11.84280449,
12.94062751, 11.39472164, 13.02765193, 10.63354329, 19.27398122,

 8.27299201, 9.95325142, 9.63652467, 11.02516879, 9.61762969,
11.58045313, 10.38329388, 10.10542696, 11.02246736, 10.28008699,
 9.63459724, 12.47130079, 11.44453778, 13.58129408, 7.11649287,
13.44300837, 8.50261026, 7.62155066, 8.65017366, 9.84337196,
11.02203988, 13.9291393 , 11.65241654, 10.39310942, 13.91347048,
 8.02963568, 11.17988932, 11.56945205, 9.87423416, 13.27645097,
12.80406096, 9.8867954 , 9.89222656, 8.2965754 , 12.46360031,
10.64706414, 14.48069491, 11.24276194, 11.48514879, 13.71011328,
11.31216151, 11.12675379, 8.90357073, 11.52183684, 13.51147299,
 9.25568645, 14.82747311, 10.89662697, 10.74162373, 11.97600088,
14.4990726 , 10.60433202, 9.34850873, 15.57413301, 11.34890714,
11.73041964, 14.26621766, 8.9082601 , 11.35577227, 12.6291   ,
12.82042777, 10.32895282, 13.5817901 , 10.47789537, 11.16532374,
11.52053751, 8.88824562, 9.79193837, 9.71078009, 9.74356491,
11.48869438, 7.71972555, 16.4781371 , 10.00553554, 11.66780497,
11.69862067, 11.80322502, 12.8998245 , 10.88972158, 10.98254274,
10.83332125, 8.9560628 , 11.58480577, 14.19099292, 11.89028671,
10.08269388, 9.59698918, 9.68517163, 10.60716866, 8.70853692,
13.5982842 , 9.48810596, 10.31281598, 12.50676521, 13.712343 ,
12.35402498, 10.10333939, 14.29006333, 12.61087439, 12.87239072,
 9.97997869, 10.9065871 , 7.99980823, 8.65345923, 12.59301936,
11.16179791, 20.78366566, 15.93371178, 13.01235289, 8.03267721,
 9.63614139, 10.2771042 , 8.53131294, 9.35404102, 17.11380376,
14.83402824, 11.1053618 , 12.25647089, 11.51388742, 10.70142025,
 9.54916778, 12.91646992, 12.64706473, 12.02482068, 16.92847623,
11.28231673, 13.44070731, 13.09343678, 14.53619115, 10.29862412,
12.11701576, 15.0562041 , 13.95581573, 9.68252086, 9.69465075,
 8.72469001, 12.13678647, 10.77420497, 14.08672969, 10.48919628,
12.48128381, 10.88624923, 10.07083376, 14.63280029, 13.70605337,
 7.72024394, 10.06701108, 7.98535683, 11.910391 , 8.60086397,
10.76242463, 12.58436056, 9.88367787, 9.07223948, 9.67178584,
14.10936339, 9.70796269, 10.54402351, 8.82494373, 14.50254639,
 7.98866612, 10.61436011, 10.00776286, 13.29210894, 12.664191 ,
10.2997158 , 13.05785422, 13.13366715, 15.04683146, 10.3495999 ,
12.38172337, 9.38575916, 10.79829515, 11.08251837, 7.94925884,
12.48736843, 10.19456286, 8.8078886 , 8.60359297, 11.9064996 ,
 9.20761856, 11.16639574, 10.43183667, 10.38397608, 10.3748597 ,
11.19891928, 10.85230099, 9.1454066 , 13.12340888, 9.82675963,
13.30778245, 10.75303462, 8.90762907, 11.18199183, 14.8943254 ,
 8.76650039, 8.66428926, 9.48595897, 8.35120105, 12.60879735,
10.93131652, 12.8849545 , 9.528814 , 12.42491159, 9.74299385,
 8.73239324, 10.26942414, 14.28209957, 12.32766542, 10.99516202,
11.54819422, 8.88148679, 14.04812957, 9.24204119, 11.52602602,
15.45881133, 13.5933563 , 9.39830553, 9.17277312, 10.64380782,
10.02983786, 10.65409077, 10.03051955, 11.24966073, 12.24087974,
11.17525466, 12.38834829, 14.88745837, 12.82941044, 14.29063651,
10.93694762, 12.41670872, 11.32762928, 9.88414708, 11.80553981,
10.555068 , 9.75317226, 10.22474193, 8.03273089, 8.58945058,
10.04207917, 10.27323867, 13.21285435, 9.67517926, 9.85097167,
10.54531914, 9.26823783, 11.95684554, 13.50715682, 13.43460435,
10.51565927, 10.69504665, 12.77862904, 13.07501091, 12.07936668,
 7.88865785, 11.39550851, 12.94121775, 8.3377301 , 9.70349407,
10.78733012, 12.50376581, 11.56266984, 8.49986829, 8.88677056,

```
       10.46530989, 11.95438247, 10.37050597,  9.70218398, 13.87978636,
        7.76118504])
```

```
y_test
```

```
668     14.5
1580     9.5
3784    12.5
463      6.5
2615    13.5
        ...
575     12.5
3231    13.5
1084     8.5
290     18.5
2713     5.5
Name: Age, Length: 836, dtype: float64
```

```
pred_train = lr.predict(x_train)
pred_train
```

```
array([ 8.15639516,  9.40335451, 14.41784548, ..., 11.01140138,
       12.7207057 ,  9.87826696])
```

```
Rings=pd.DataFrame({'Actual_y_value':y_test,'Predicted_y_value':pred_test})
Rings.head(10)
```

	Actual_y_value	Predicted_y_value
668	14.5	13.806904
1580	9.5	10.760182
3784	12.5	11.912669
463	6.5	7.741656
2615	13.5	12.646404
1399	12.5	13.019023
2054	8.5	9.312169

	Actual_y_value	Predicted_y_value
2058	9.5	10.978648
217	8.5	10.209236
1931	10.5	12.920695

```
import sklearn.metrics as sm
print("Mean absolute error =", round(sm.mean_absolute_error(y_test,
pred_test), 2))
print("Mean squared error =", round(sm.mean_squared_error(y_test, pred_test),
2))
print("Median absolute error =", round(sm.median_absolute_error(y_test,
pred_test), 2))
print("Explain variance score =", round(sm.explained_variance_score(y_test,
pred_test), 2))
print("R2 score =", round(sm.r2_score(y_test, pred_test), 2))
```

```
Mean absolute error = 1.7
Mean squared error = 5.69
Median absolute error = 1.23
Explain variance score = 0.48
R2 score = 0.48
```