Assignment -3

Python Programming

Building a Regression Model

1. Perform Below Visualizations.

Univariate Analysis

1. Summary Statistics

In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import statsmodels.api as sm
```

In [2]:
```
file_data = pd.read_csv('C:/KavinKumar/abalone.csv')
file_data
```

Out[2]:

	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	0.1500	15
1	M	0.350	0.265	0.090	0.2255	0.0995	0.0485	0.0700	7
2	F	0.530	0.420	0.135	0.6770	0.2565	0.1415	0.2100	9
3	M	0.440	0.365	0.125	0.5160	0.2155	0.1140	0.1550	10
4	I	0.330	0.255	0.080	0.2050	0.0895	0.0395	0.0550	7
...	...	...	...	...	...	...	...	...	...
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

4177 rows × 9 columns

Add a Age column in a dataset

In [3]:
```
file_data['Age']=''
file_data.head()
```

Out[3]:

	Sex	Length	Diameter	Height	Whole weight	Shucked weight	Viscera weight	Shell weight	Rings	Age
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	

In [4]:
```
file_data['Age']=file_data['Rings']+1.5
file_data.head()
```

Out[4]:

	Sex	Length	Diameter	Height	Whole weight	Shucked weight	Viscera weight	Shell weight	Rings	Age
0	M	0.455	0.365	0.095	0.5140	0.2245	0.1010	0.150	15	16.5
1	M	0.350	0.265	0.090	0.2255	0.0995	0.0485	0.070	7	8.5
2	F	0.530	0.420	0.135	0.6770	0.2565	0.1415	0.210	9	10.5
3	M	0.440	0.365	0.125	0.5160	0.2155	0.1140	0.155	10	11.5
4	I	0.330	0.255	0.080	0.2050	0.0895	0.0395	0.055	7	8.5

Drop the Rings Column

In [5]:
```
file_data = file_data.drop(columns=['Rings'],axis=1)
file_data
```

Out[5]:

	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	0.1500	16.5
1	M	0.350	0.265	0.090	0.2255	0.0995	0.0485	0.0700	8.5
2	F	0.530	0.420	0.135	0.6770	0.2565	0.1415	0.2100	10.5
3	M	0.440	0.365	0.125	0.5160	0.2155	0.1140	0.1550	11.5
4	I	0.330	0.255	0.080	0.2050	0.0895	0.0395	0.0550	8.5
...	...	...	...	...	...	...	...	...	...
4172	F	0.565	0.450	0.165	0.8870	0.3700	0.2390	0.2490	12.5
4173	M	0.590	0.440	0.135	0.9660	0.4390	0.2145	0.2605	11.5
4174	M	0.600	0.475	0.205	1.1760	0.5255	0.2875	0.3080	10.5
4175	F	0.625	0.485	0.150	1.0945	0.5310	0.2610	0.2960	11.5
4176	M	0.710	0.555	0.195	1.9485	0.9455	0.5765	0.4850	13.5

4177 rows × 9 columns

In [6]:
```
file_data['Height'].mean()
```

Out[6]:
```
0.13951639932966614
```

In [7]:
```
file_data['Height'].median()
```

Out[7]:
```
0.14
```

In [8]:
```
file_data['Height'].std()
```

Out[8]:
```
0.041827056669725703
```

2. Frequency Table

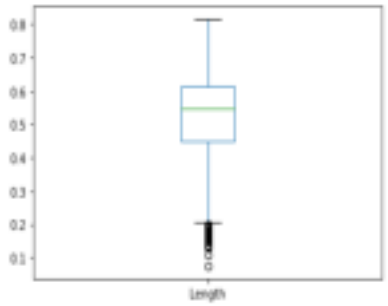
In [9]:
```
file_data['Sex'].value_counts()
```

Out[9]:
```
M    1528
I    1342
F    1307
Name: Sex, dtype: int64
```

3. Create Charts

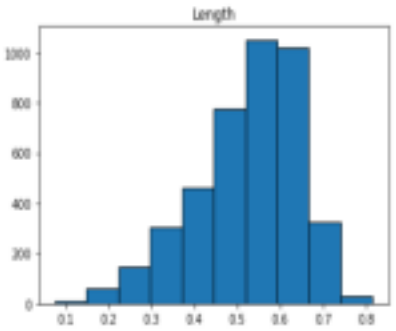
In [10]:
```
file_data.boxplot(column=['Length'], grid=False)
```

Out[10]:
```
<AxesSubplot:>
```



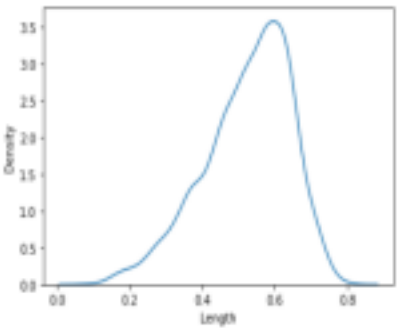
In [11]:
```
file_data.hist(column='Length', grid=False, edgecolor='black')
```

Out[11]:
```
array([[<AxesSubplot:title={'center':'Length'}>]], dtype=object)
```



In [12]:
```
sns.kdeplot(file_data['Length'])
```

Out[12]:
```
<AxesSubplot:xlabel='Length', ylabel='Density'>
```

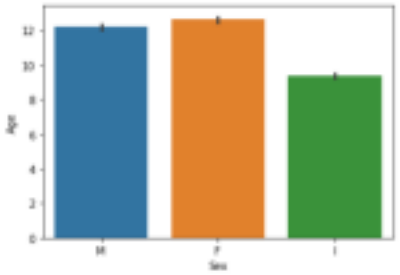


Bi - Variate Analysis

1. Barplot

In [13]:
```
data = sns.barplot(x = file_data["Sex"], y = file_data["Age"])
data
```

Out[13]:
```
<AxesSubplot:xlabel='Sex', ylabel='Age'>
```



2. Correlation Coefficients

In [14]:
```
file_data.corr()
```

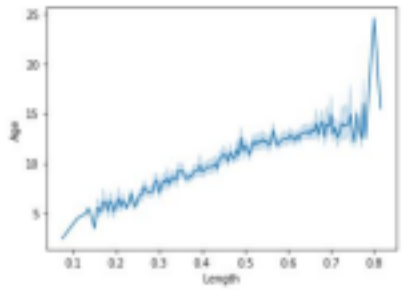
Out[14]:

	Length	Diameter	Height	Whole weight	Shucked weight	Viscera weight	Shell weight	Age
Length	1.000000	0.986812	0.827554	0.925261	0.897914	0.903018	0.897706	0.556720
Diameter	0.986812	1.000000	0.833684	0.925452	0.893162	0.899724	0.905330	0.574660
Height	0.827554	0.833684	1.000000	0.819221	0.774972	0.798319	0.817338	0.557467
Whole weight	0.925261	0.925452	0.819221	1.000000	0.968405	0.966375	0.955355	0.540390
Shucked weight	0.897914	0.893162	0.774972	0.968405	1.000000	0.931961	0.882617	0.420884
Viscera weight	0.903018	0.899724	0.798319	0.966375	0.931961	1.000000	0.907656	0.503819
Shell weight	0.897706	0.905330	0.817338	0.955355	0.882617	0.907656	1.000000	0.627574
Age	0.556720	0.574660	0.557467	0.540390	0.420884	0.503819	0.627574	1.000000

3.Linear Plot

In [15]:
```
data = sns.lineplot(x = file_data["Length"], y = file_data["Age"])
data
```

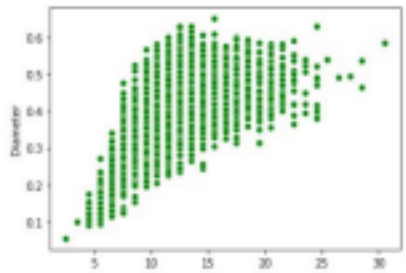
Out[15]:
```
<AxesSubplot:xlabel='Length', ylabel='Age'>
```



4. Scatter Plot

In [16]:
```
data = sns.scatterplot(x = file_data['Age'],y = file_data['Diameter'], color="green")
data
```

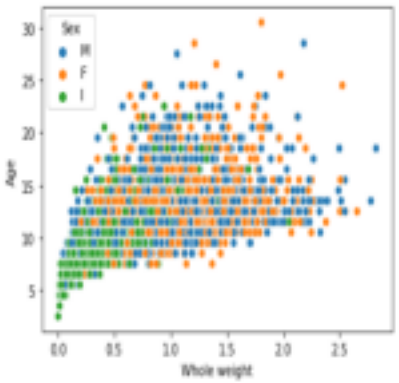
Out[16]:
```
<AxesSubplot:xlabel='Age', ylabel='Diameter'>
```



Multi - Variate Analysis

In [17]:
```
x = sns.scatterplot(x=file_data['Whole weight'],y=file_data['Age'],hue=file_data["Sex"])
x
```

Out[17]:
```
<AxesSubplot:xlabel='Whole weight', ylabel='Age'>
```



4. Perform descriptive statistics on the dataset.

In [18]:
```
file_data.shape
```

Out[18]:
```
(4177, 9)
```

In [19]:
```
file_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   Age             4177 non-null   float64
dtypes: float64(8), object(1)
memory usage: 293.8+ KB
```

In [20]:
```
file_data.describe()
```

Out[20]:

	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
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

In [21]:
```
file_data.head()
```

Out[21]:

	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

In [22]:
```
file_data.tail()
```

Out[22]:

	Sex	Length	Diameter	Height	Whole weight	Shucked weight	Viscera weight	Shell weight	Age
4172	F	0.565	0.450	0.165	0.8870	0.3700	0.2390	0.2490	12.5
4173	M	0.590	0.440	0.135	0.9660	0.4390	0.2145	0.2605	11.5
4174	M	0.600	0.475	0.205	1.1760	0.5255	0.2875	0.3080	10.5
4175	F	0.625	0.485	0.150	1.0945	0.5310	0.2610	0.2960	11.5
4176	M	0.710	0.555	0.195	1.9485	0.9455	0.3765	0.4950	13.5

In [23]:
```
file_data.mean(numeric_only=True)
```

Out[23]:
```
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
Age              11.433684
dtype: float64
```

In [24]:
```
file_data.median(numeric_only=True)
```

Out[24]:
```
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
```

In [25]:
```
file_data.mode()
```

Out[25]:

	Sex	Length	Diameter	Height	Whole weight	Shucked weight	Viscera weight	Shell weight	Age
0	M	0.550	0.45	0.15	0.2225	0.175	0.1715	0.275	10.5
1	NaN	0.625	NaN	NaN	NaN	NaN	NaN	NaN	NaN

In [26]:
```
file_data.var(numeric_only=True)
```

Out[26]:
```
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
Age              10.395266
dtype: float64
```

In [27]:
```
file_data.std(numeric_only=True)
```

Out[27]:
```
Length            0.120093
Diameter          0.099240
Height            0.041827
Whole weight      0.490389
Shucked weight    0.221963
Viscera weight    0.109634
Shell weight      0.139203
Age               3.224169
dtype: float64
```

In [28]:
```
file_data.skew(numeric_only=True)
```

Out[28]:
```
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
Age               1.114102
dtype: float64
```

In [29]:
```
file_data.kurt(numeric_only=True)
```

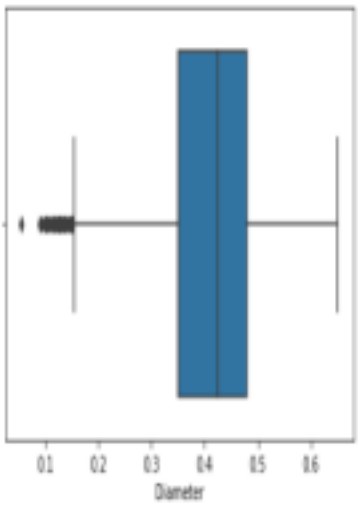
Out[29]:
```
Length            0.064621
Diameter         -0.045476
Height           76.025509
Whole weight     -0.023644
Shucked weight    0.595124
Viscera weight    0.084812
Shell weight      0.531926
Age               2.330687
dtype: float64
```

In [30]:
```
quantile = file_data['Whole weight'].quantile(q=[0.75, 0.25])
quantile
```

Out[30]:
```
0.75    1.1530
0.25    0.4415
Name: Whole weight, dtype: float64
```

In [31]:
```
x = file_data.Diameter
sns.boxplot(x=x)
```

Out[31]:



5. Handle the Missing values.

In [32]:
```
print(file_data.isnull())
```

```
       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    Age
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]
```

In [33]:
```
print(file_data.isnull().sum())
```

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

In [34]:
```
file_data.isna().any()
```

Out[34]:
```
Sex               False
Length            False
Diameter          False
Height            False
Whole weight      False
Shucked weight    False
Viscera weight    False
Shell weight      False
Age               False
dtype: bool
```

6. Find the outliers and replace the outliers

In [35]:
```
x = sns.boxplot(x=file_data["Age"])
x
```

Out[35]: