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

```
data=pd.read_csv("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.shape
```

```
(4177, 9)
```

Saving...

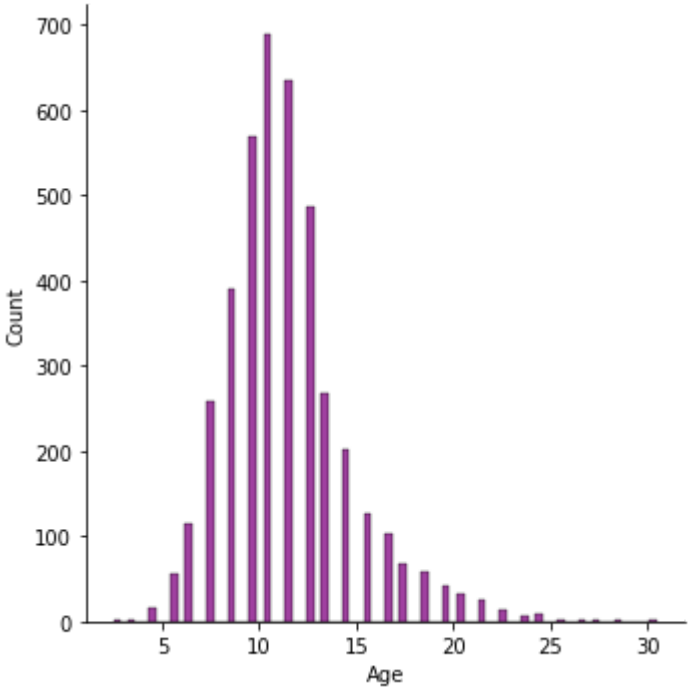
```
data=data.rename(columns = {'Whole weight':'Whole_weight','Shucked weight': 'Shucked_weigh
                            'Shell weight': 'Shell_weight'})
data=data.drop(columns=["Rings"],axis=1)
```

```
data.head()
```

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

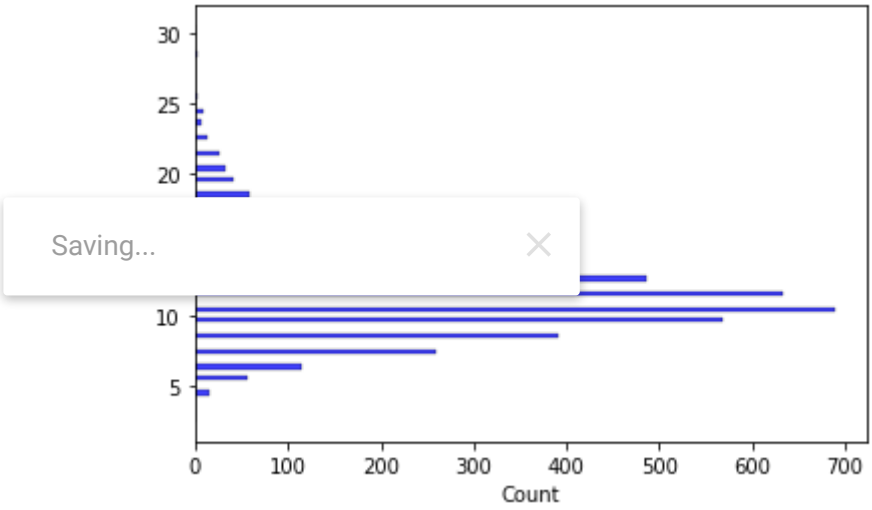
```
sns.displot(data["Age"], color='purple')
```

```
<seaborn.axisgrid.FacetGrid at 0x7f58955af550>
```




```
sns.histplot(y=data.Age,color='blue')
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f58954a2590>
```

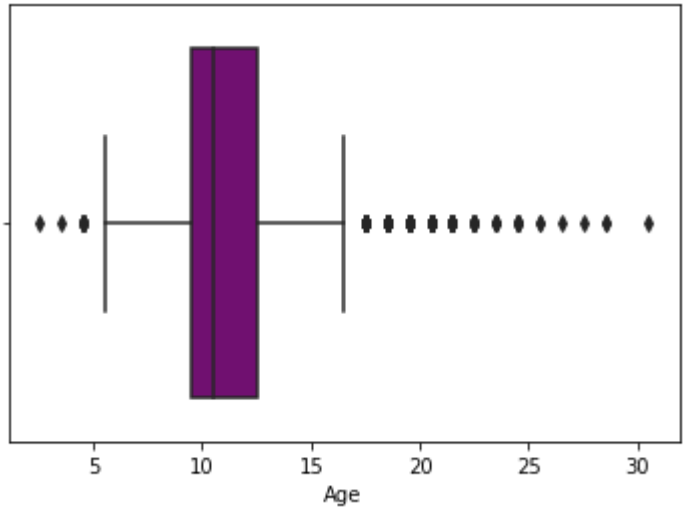



```
sns.histplot(x=data.Age,color='yellow')
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f5892599e10>
```

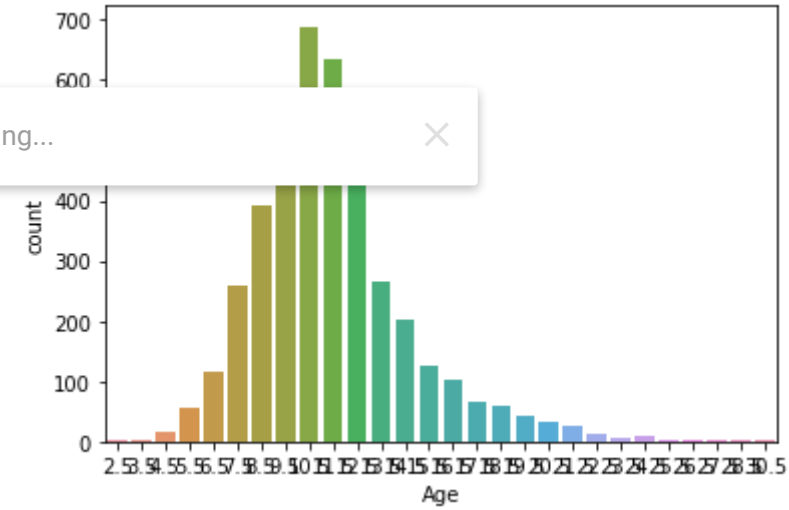

```
sns.boxplot(x=data.Age,color='purple')
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f58955fda50>
```




```
sns.countplot(x=data.Age)
```

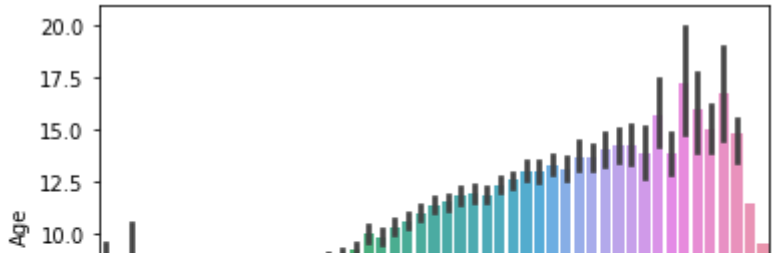
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f58923ef710>
```



Saving...

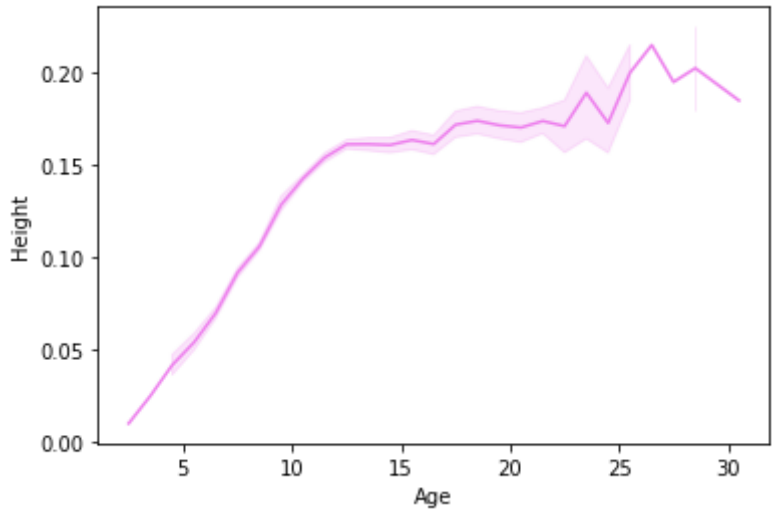
```
sns.barplot(x=data.Height,y=data.Age)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f5892306d50>
```




```
sns.lineplot(x=data.Age,y=data.Height, color='violet')
```

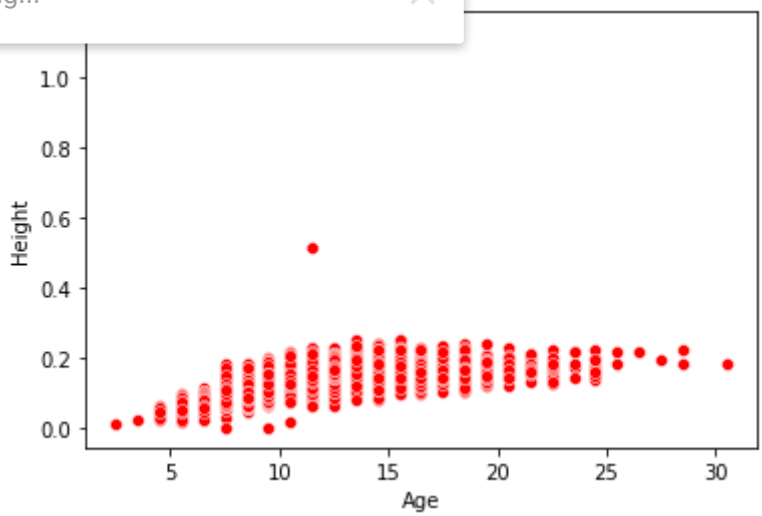
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f58920adf90>
```




```
sns.scatterplot(x=data.Age,y=data.Height,color='red')
```

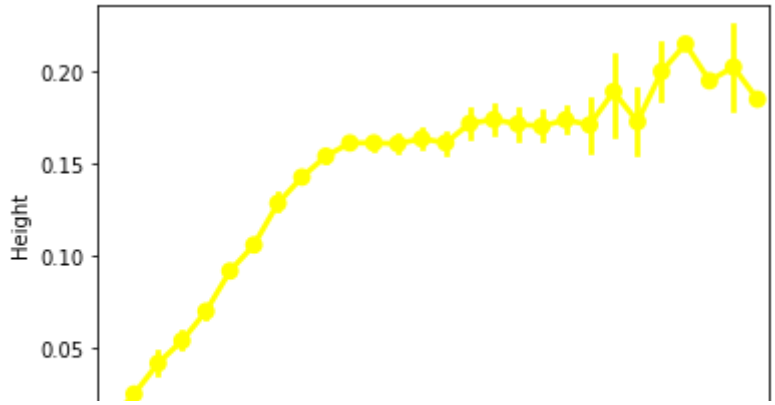
Saving...

```
sSubplot at 0x7f5892b88610>
```



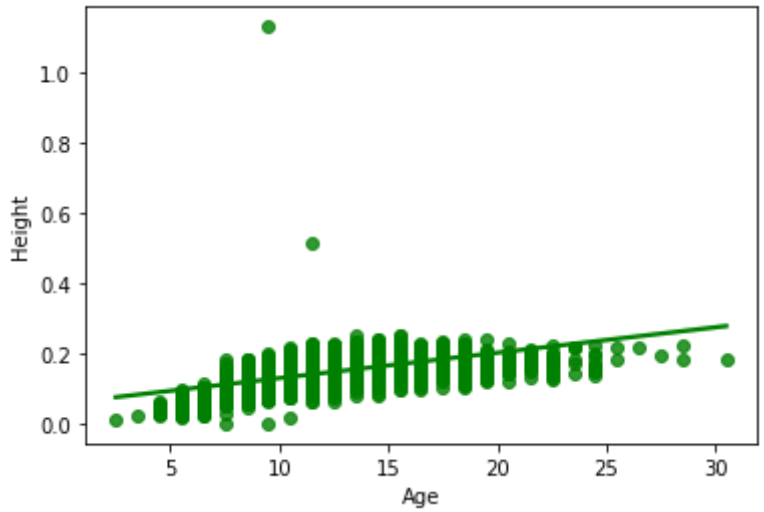

```
sns.pointplot(x=data.Age, y=data.Height, color="yellow")
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f58920e9050>
```




```
sns.regplot(x=data.Age,y=data.Height,color='green')
```

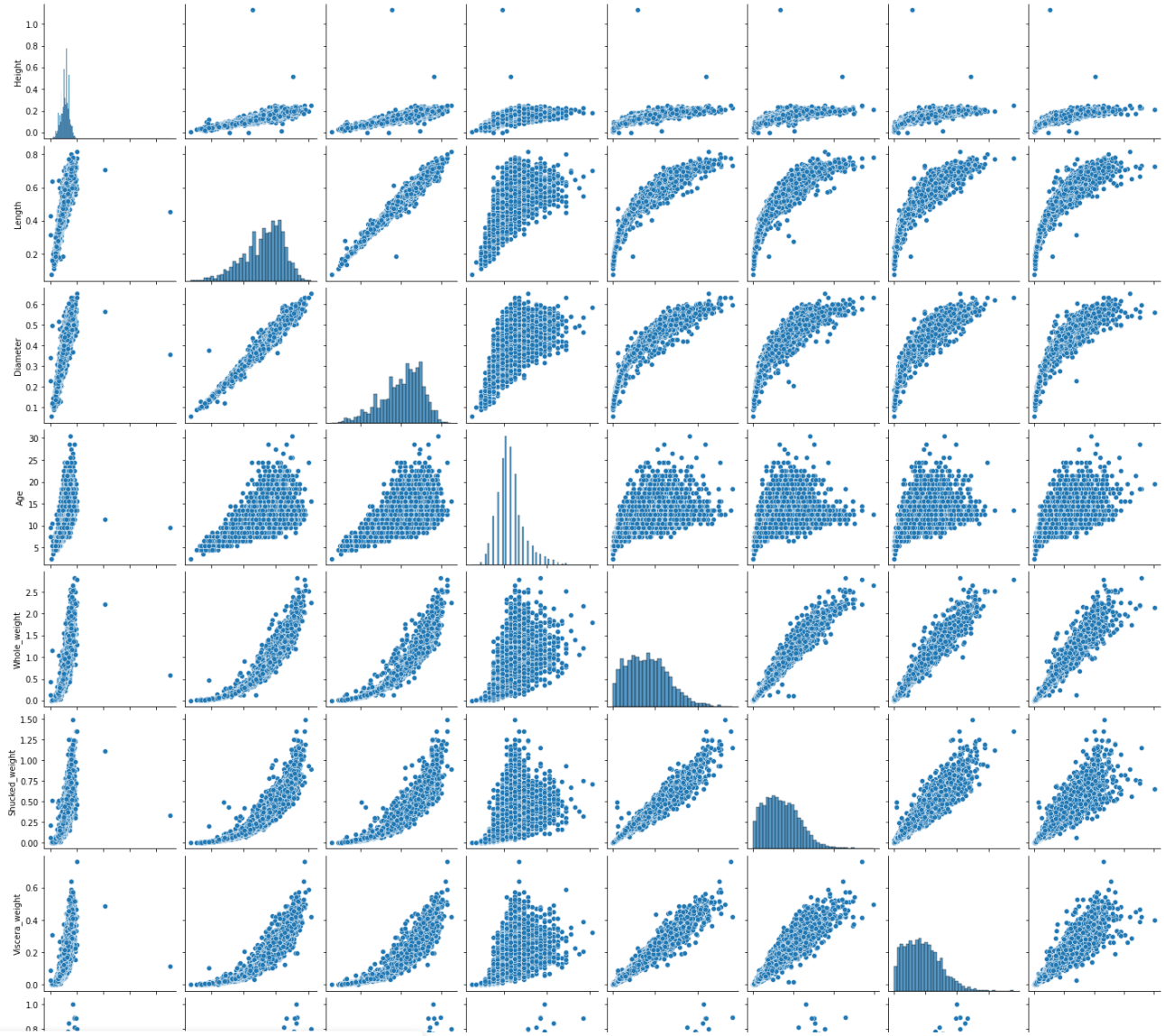
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f5896f0fbd0>
```



Saving...

```
ength","Diameter","Age","Whole_weight","Shucked_weight"
```

```
<seaborn.axisgrid.PairGrid at 0x7f5891e6dfd0>
```



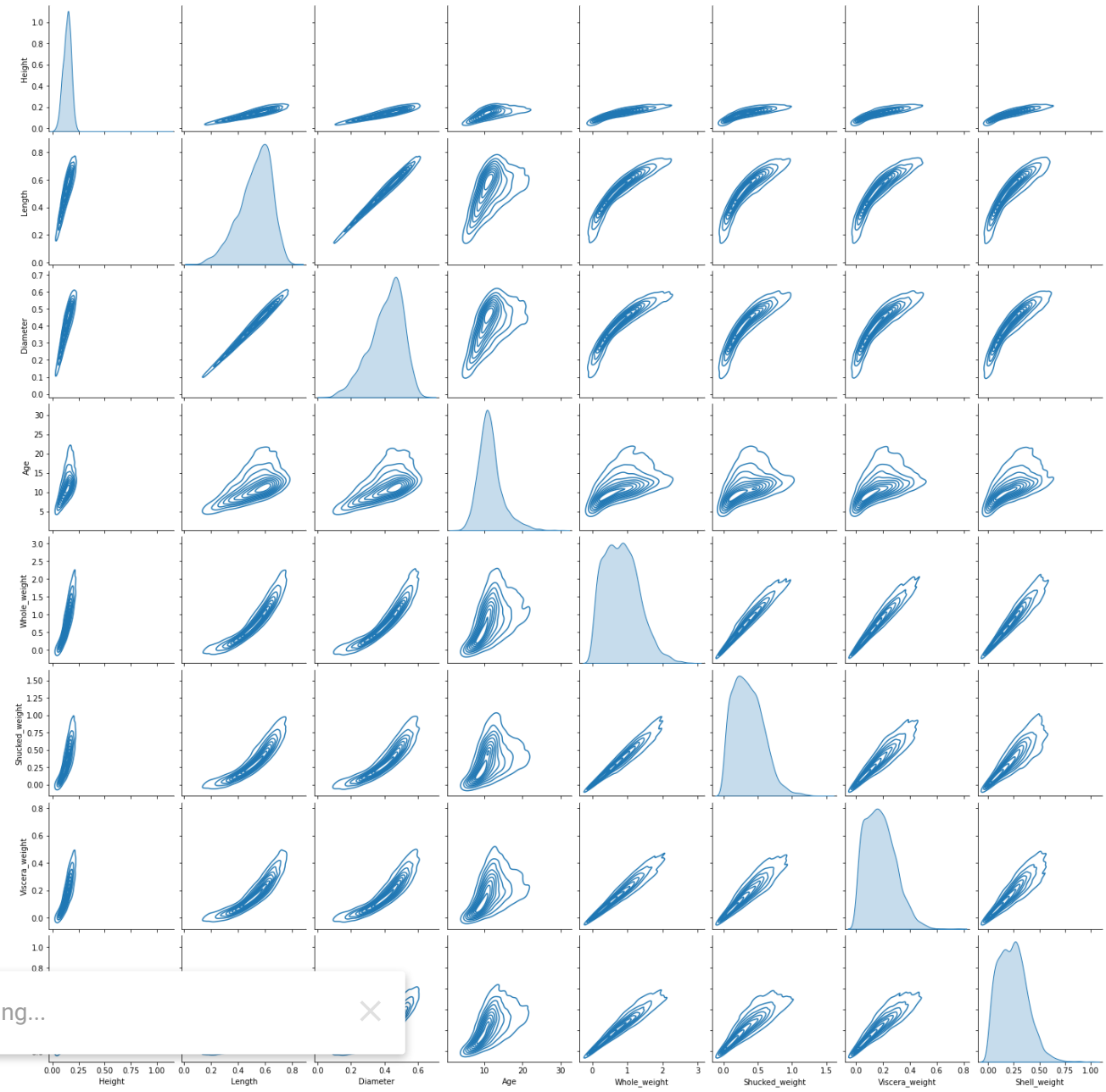
Saving...

```
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/
Requirement already satisfied: statsmodels in /usr/local/lib/python3.7/dist-packages
Requirement already satisfied: scipy>=1.3 in /usr/local/lib/python3.7/dist-packages
Requirement already satisfied: pandas>=0.25 in /usr/local/lib/python3.7/dist-package
Requirement already satisfied: patsy>=0.5.2 in /usr/local/lib/python3.7/dist-package
Requirement already satisfied: packaging>=21.3 in /usr/local/lib/python3.7/dist-pack
Requirement already satisfied: numpy>=1.17 in /usr/local/lib/python3.7/dist-packages
Requirement already satisfied: pyparsing!=3.0.5,>=2.0.2 in /usr/local/lib/python3.7/
Requirement already satisfied: pytz>=2017.3 in /usr/local/lib/python3.7/dist-package
Requirement already satisfied: python-dateutil>=2.7.3 in /usr/local/lib/python3.7/di
Requirement already satisfied: six in /usr/local/lib/python3.7/dist-packages (from p
```

```
import seaborn as sns
```

```
sns.pairplot(data=data[["Height","Length","Diameter","Age","Whole_weight","Shucked_weight"
```

```
<seaborn.axisgrid.PairGrid at 0x7f58907cd390>
```




```
data.describe(include='all')
```

	Sex	Length	Diameter	Height	Whole_weight	Shucked_weight	Vi
count	4177	4177.000000	4177.000000	4177.000000	4177.000000	4177.000000	
unique	3	NaN	NaN	NaN	NaN	NaN	
top	M	NaN	NaN	NaN	NaN	NaN	
freq	1528	NaN	NaN	NaN	NaN	NaN	
mean	NaN	0.523992	0.407881	0.139516	0.828742	0.359367	
std	NaN	0.120093	0.099240	0.041827	0.490389	0.221963	
min	NaN	0.075000	0.055000	0.000000	0.002000	0.001000	
25%	NaN	0.450000	0.350000	0.115000	0.441500	0.186000	
50%	NaN	0.545000	0.425000	0.140000	0.799500	0.336000	
75%	NaN	0.615000	0.480000	0.165000	1.153000	0.502000	
max	NaN	0.815000	0.650000	1.130000	2.825500	1.488000	

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

```
Sex               0
Length            0
Diameter          0
Height            0
Whole_weight      0
Shucked_weight    0
```



```
dtype: int64
```

```
outliers=data.quantile(q=(0.25,0.75))
```

```
outliers
```

	Length	Diameter	Height	Whole_weight	Shucked_weight	Viscera_weight	Shell_
0.25	0.450	0.35	0.115	0.4415	0.186	0.0935	
0.75	0.615	0.48	0.165	1.1530	0.502	0.2530	

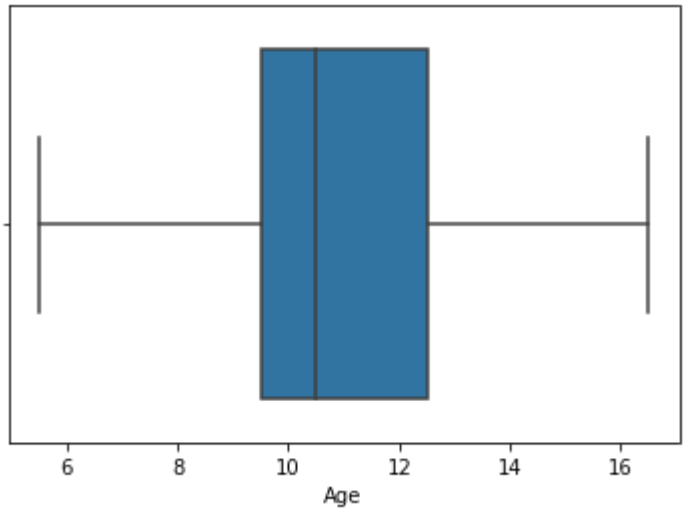
```
a = data.Age.quantile(0.25)
b = data.Age.quantile(0.75)
c = b - a
lower_limit = a - 1.5 * c
data.median(numeric_only=True)
```

```
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
```

```
data['Age'] = np.where(data['Age'] < lower_limit, 7, data['Age'])
sns.boxplot(x=data.Age,showfliers = False)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f58922b9e10>
```




```
from sklearn.preprocessing import LabelEncoder

lab = LabelEncoder()
                                    ex)
```

Saving...

```
data.head()
```

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

```
y = data["Sex"]
y.head()
```

```
0    2
1    2
```

```
2    0
3    2
4    1
Name: Sex, dtype: int64
```

```
x=data.drop(columns=["Sex"],axis=1)
x.head()
```

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

```
from sklearn.preprocessing import scale
X_Scaled = pd.DataFrame(scale(x), columns=x.columns)
X_Scaled.head()
```

	Length	Diameter	Height	Whole_weight	Shucked_weight	Viscera_weight	Shel
0	-0.574558	-0.432149	-1.064424	-0.641898	-0.607685	-0.726212	
1	-1.448986	-1.439929	-1.183978	-1.230277	-1.170910	-1.205221	
2	0.050033	0.122130	-0.107991	-0.309469	-0.463500	-0.356690	
3	-0.699476	-0.432149	-0.347099	-0.637819	-0.648238	-0.607600	
				87	-1.272086	-1.215968	-1.287337	

Saving...

```
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_sta
```

```
X_Train.shape,X_Test.shape
```

```
((3341, 8), (836, 8))
```

```
((3341, 8), (836, 8))
```

```
((3341, 8), (836, 8))
```

```
Y_Train.shape,Y_Test.shape
```

```
((3341,), (836,))
```

```
X_Train.head()
```

	Length	Diameter	Height	Whole_weight	Shucked_weight	Viscera_weight	S
3141	-2.864726	-2.750043	-1.423087	-1.622870	-1.553902	-1.583867	
3521	-2.573250	-2.598876	-2.020857	-1.606554	-1.551650	-1.565619	
883	1.132658	1.230689	0.728888	1.145672	1.041436	0.286552	
3627	1.590691	1.180300	1.446213	2.164373	2.661269	2.330326	
2106	0.591345	0.474853	0.370226	0.432887	0.255175	0.272866	

```
X_Test.head()
```

	Length	Diameter	Height	Whole_weight	Shucked_weight	Viscera_weight	S
668	0.216591	0.172519	0.370226	0.181016	-0.368878	0.569396	
1580	-0.199803	-0.079426	-0.466653	-0.433875	-0.443224	-0.343004	
3784	0.799543	0.726798	0.370226	0.870348	0.755318	1.764639	
463	-2.531611	-2.447709	-2.020857	-1.579022	-1.522362	-1.538247	
2615	1.007740	0.928354	0.848442	1.390405	1.415417	1.778325	

```
Y_Train.head()
```

```
3141    1
3521    1
883     2
3627    2
2106    2
```

Saving...

```
Y_Test.head()
```

```
668     2
1580    1
3784    2
463     1
2615    2
Name: Sex, dtype: int64
```

```
from sklearn.ensemble import RandomForestClassifier
model = RandomForestClassifier(n_estimators=10,criterion='entropy')
```

Colab paid products - Cancel contracts here

✓ 0s completed at 11:12 AM

Saving...