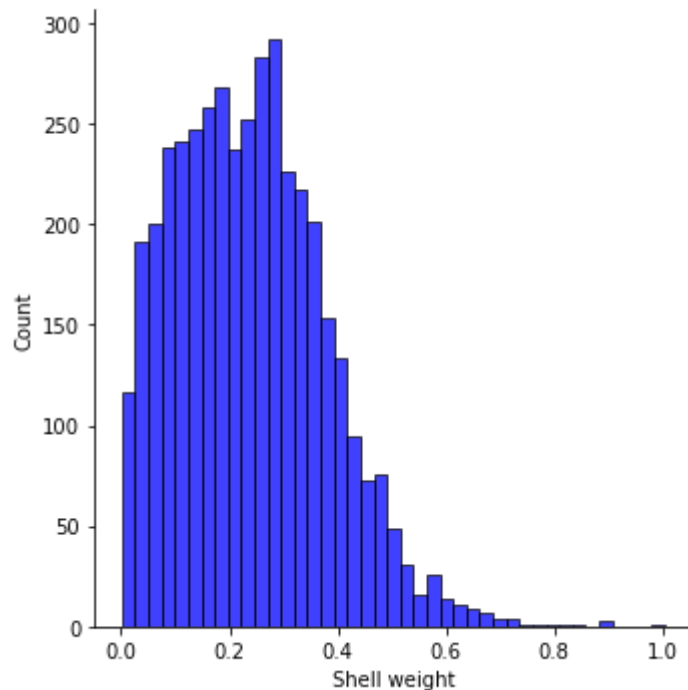


```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px
```

```
sns.displot( data["Shell weight"], color='blue')
```

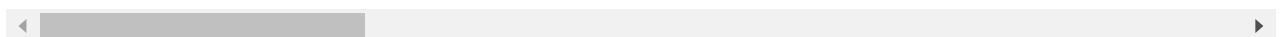
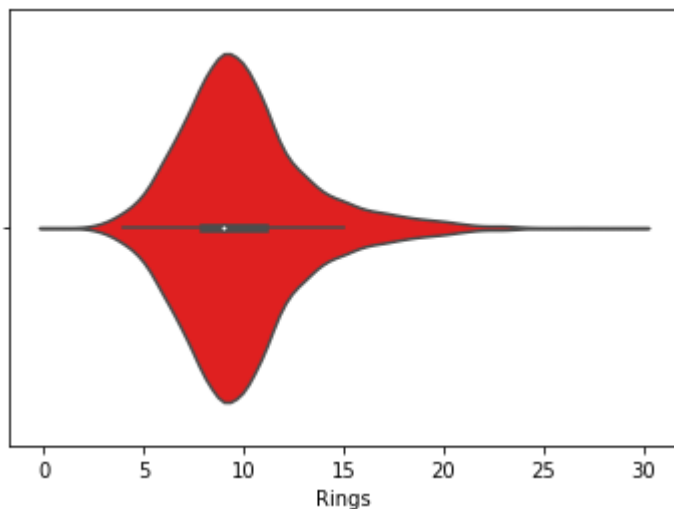


<seaborn.axisgrid.FacetGrid at 0x1b4a9c96fd0>



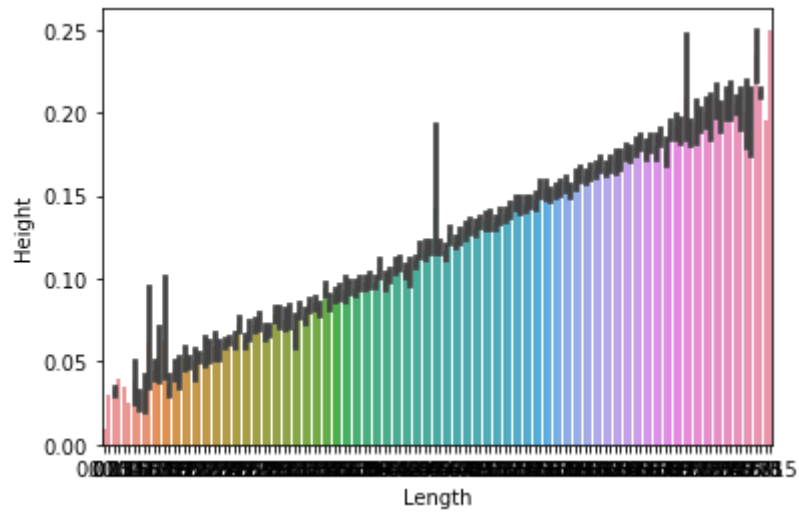
```
sns.violinplot(data["Rings"], color='red')
```

C:\Users\Dell\anaconda3\lib\site-packages\seaborn_decorators.py:36: FutureWarning
warnings.warn(
<AxesSubplot:xlabel='Rings'>



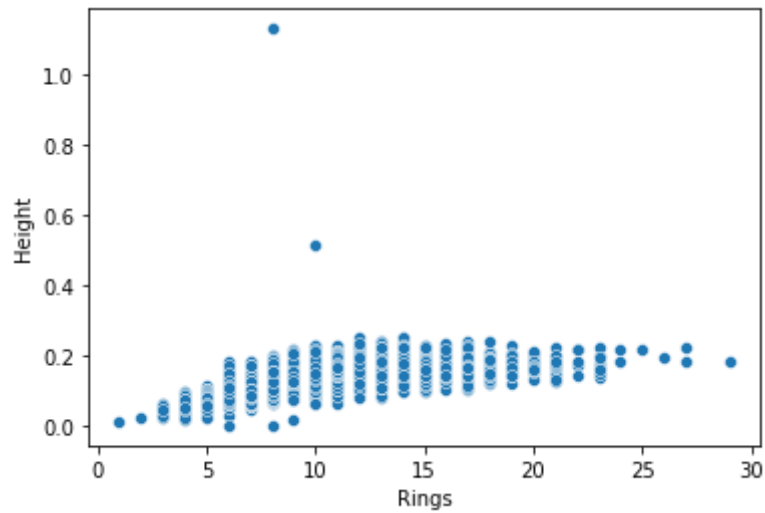
```
sns.barplot(x=data.Length,y=data.Height)
```

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



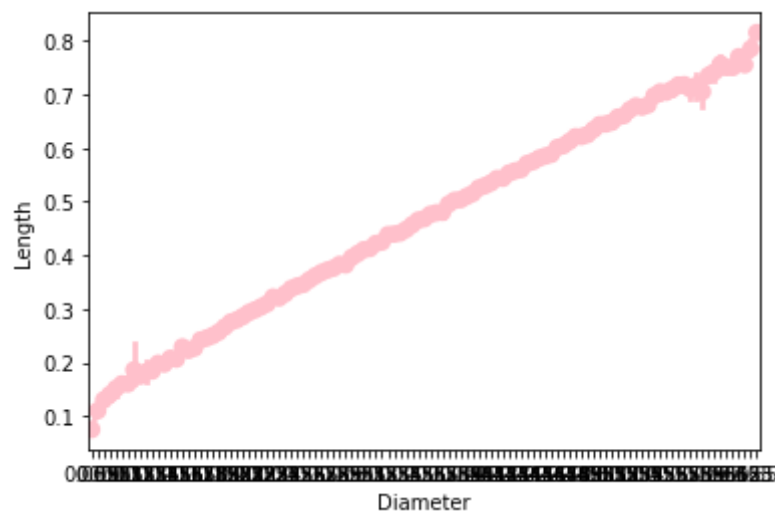
```
sns.scatterplot(x=data.Rings,y=data.Height)
```

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



```
sns.pointplot(x=data.Diameter,y=data.Length, color="pink")
```

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



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

<AxesSubplot:>



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   Rings            4177 non-null   int64
dtypes: float64(7), int64(1), object(1)
memory usage: 293.8+ KB
```

data.describe()

	Length	Diameter	Height	Whole weight	Shucked weight	Viscera weight
count	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
std	0.120093	0.099240	0.041827	0.490389	0.221963	0.109614
min	0.075000	0.055000	0.000000	0.002000	0.001000	0.000500
25%	0.450000	0.350000	0.115000	0.441500	0.186000	0.093500
50%	0.545000	0.425000	0.140000	0.799500	0.336000	0.171000
75%	0.615000	0.480000	0.165000	1.153000	0.502000	0.253000

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

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

```
fig = px.histogram(data, x='Whole weight')
fig.show()
```

```
from sklearn.preprocessing import LabelEncoder
le = LabelEncoder()
data["Sex"] = le.fit_transform(data["Sex"])
data["Sex"]
```

```
0      2
1      2
```

```

2      0
3      2
4      1
..
4172   0
4173   2
4174   2
4175   0
4176   2
Name: Sex, Length: 4177, dtype: int32

```

```

x=data.iloc[:,0:8].values
y=data.iloc[:,8:9].values

```

```

x=data.iloc[:,0:8]

```

```

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=0)

```

```

x_train.shape

(2923, 8)

```

```

x_test.shape

(1254, 8)

```

```

from sklearn.linear_model import LinearRegression
lr = LinearRegression()

```

```

lr.fit(x_train, y_train)

LinearRegression()

```

```

y_pred = lr.predict(x_test)
print((y_test)[0:6])
print((y_pred)[0:6])

```

```

[[13]
 [ 8]
 [11]
 [ 5]
 [12]
 [11]]
[[13.11640829]
 [ 9.65691091]
 [10.35350972]
 [ 5.63648715]
 [10.67436485]
 [11.95341338]]

```

```
from sklearn.metrics import mean_squared_error
mse = mean_squared_error(y_test, y_pred)
rmse = np.sqrt(mse)
print("RMSE value : {:.2f}".format(rmse))
```

RMSE value : 2.26

```
from sklearn.model_selection import cross_val_score
cv_scores = cross_val_score(lr, x, y, cv=5)
sco=cv_scores.round(4)
print(cv_scores.round(4))
print("Average",sco.sum()/5)
```

[0.4113 0.1574 0.4807 0.5046 0.4362]
Average 0.39803999999999995