In [14]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from matplotlib import rcParams
import seaborn as sns
```

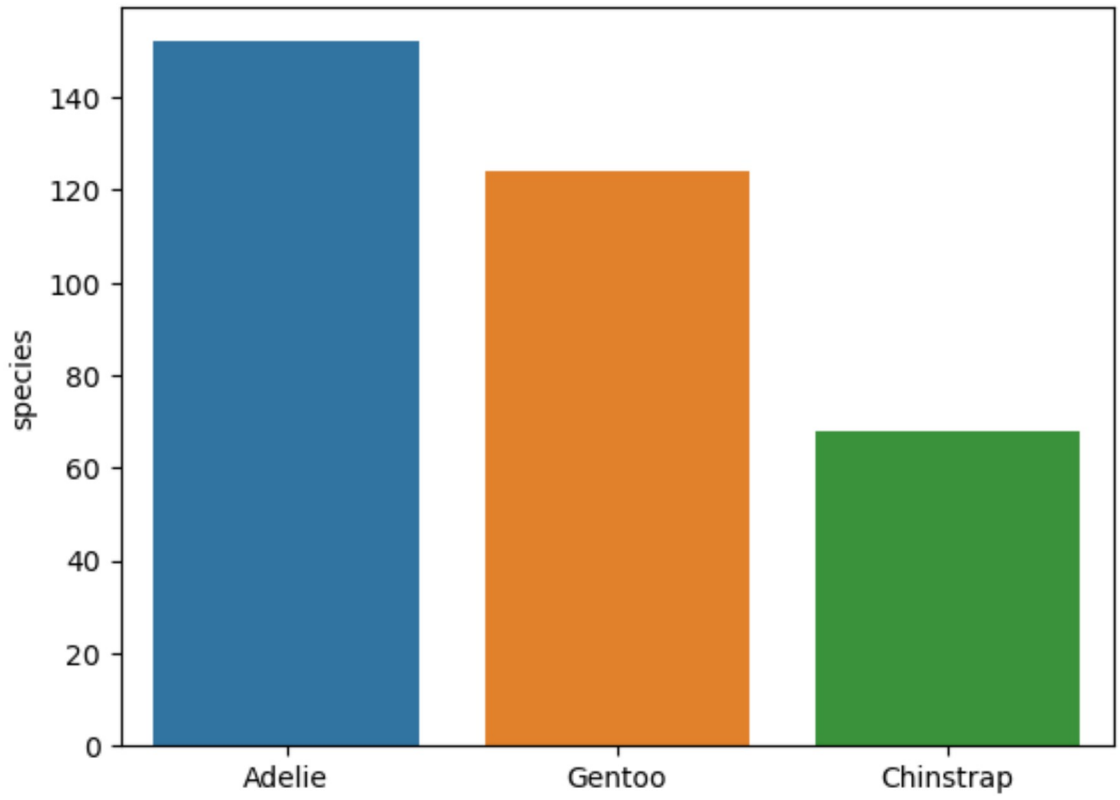
In [15]:
```
file_path = "C:/Users/utkar/Downloads/Untitled Folder 1/penguins_size.csv"
df = pd.read_csv(file_path)
df.head()
```

Out[15]:

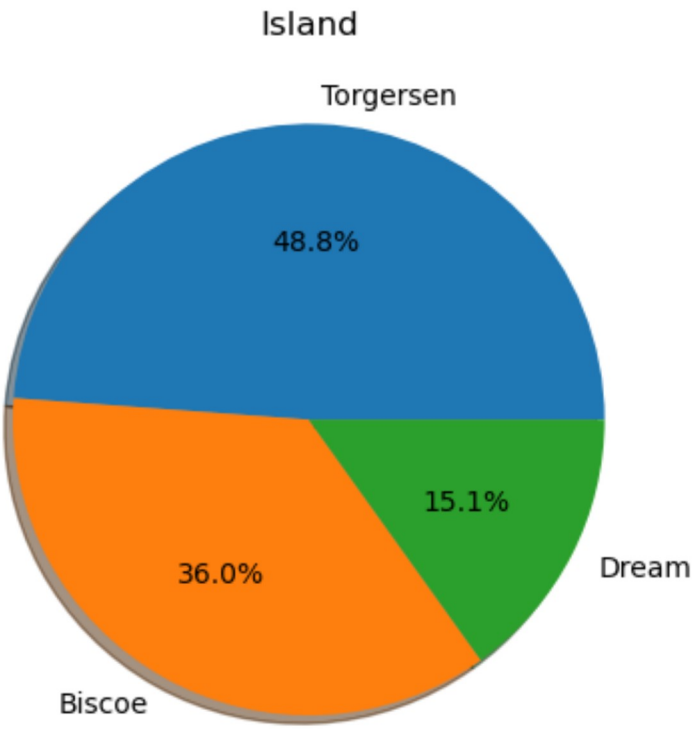
| | species | island | culmen_length_mm | culmen_depth_mm | flipper_length_mm | body_mass_g |
|---|---|---|---|---|---|---|
| **0** | Adelie | Torgersen | 39.1 | 18.7 | 181.0 | 3750.0 |
| **1** | Adelie | Torgersen | 39.5 | 17.4 | 186.0 | 3800.0 |
| **2** | Adelie | Torgersen | 40.3 | 18.0 | 195.0 | 3250.0 |
| **3** | Adelie | Torgersen | NaN | NaN | NaN | NaN |
| **4** | Adelie | Torgersen | 36.7 | 19.3 | 193.0 | 3450.0 |

In [16]:
```
sns.barplot(x =df.species.value_counts().index,y =df.species.value_counts() )
```

Out[16]:
```
<Axes: ylabel='species'>
```



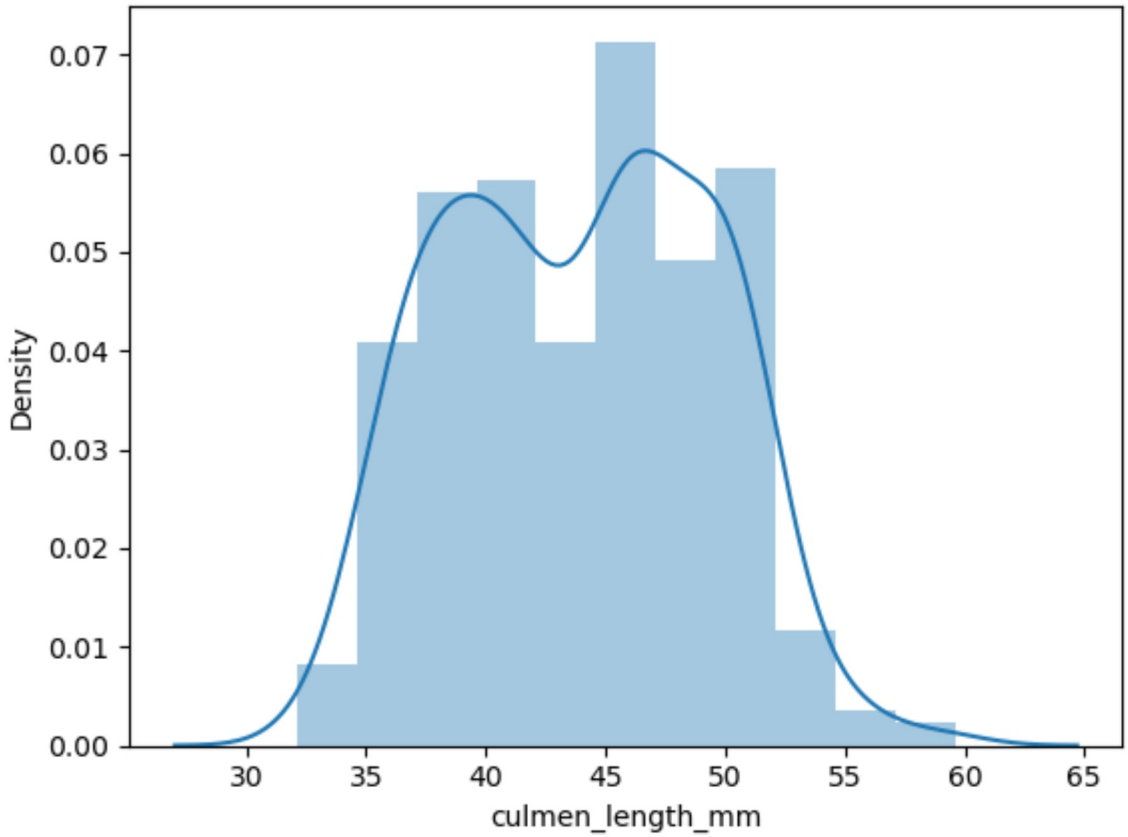

In [17]:
```
plt.pie(df.island.value_counts(),[0,0,0],labels = ['Torgersen','Biscoe','Dream'])
plt.title("Island")
plt.show()
```




In [18]:
```
sns.distplot(df.culmen_length_mm)
```

```
C:\Users\utkar\AppData\Local\Temp\ipykernel_15528\41153472.py:1: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  sns.distplot(df.culmen_length_mm)
```

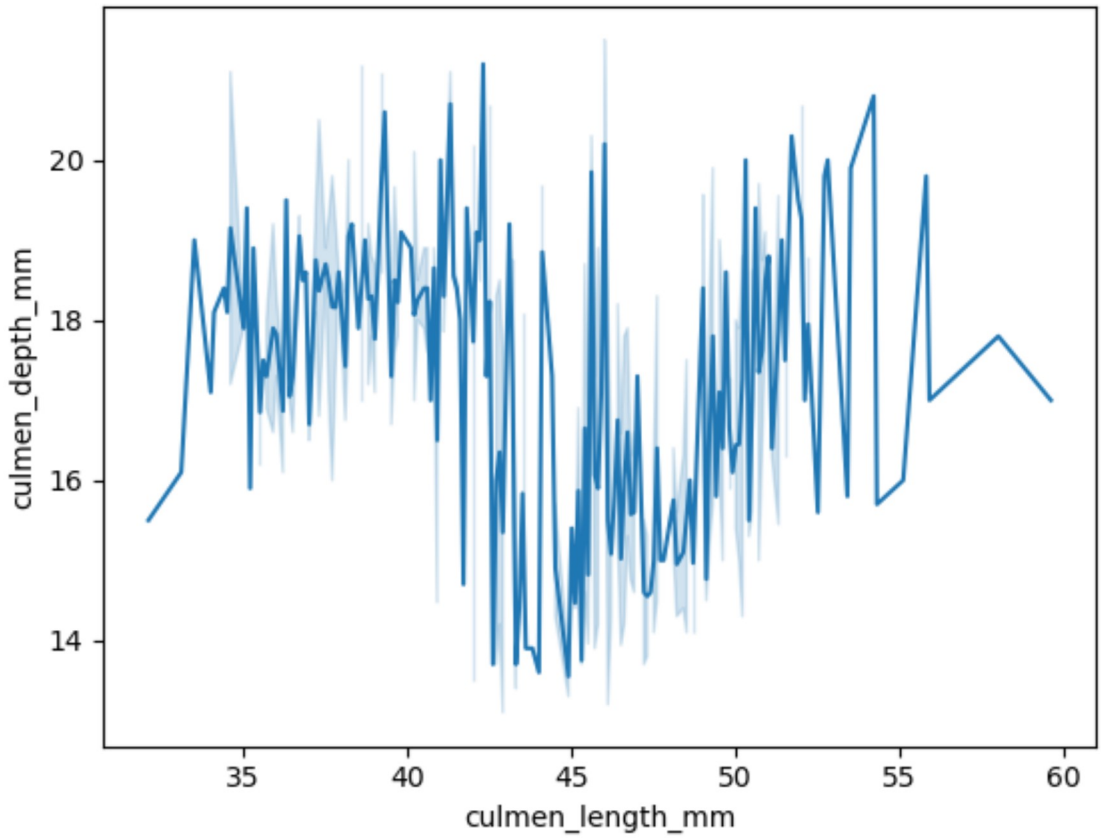
Out[18]:
```
<Axes: xlabel='culmen_length_mm', ylabel='Density'>
```




## Bi- Variate Analysis

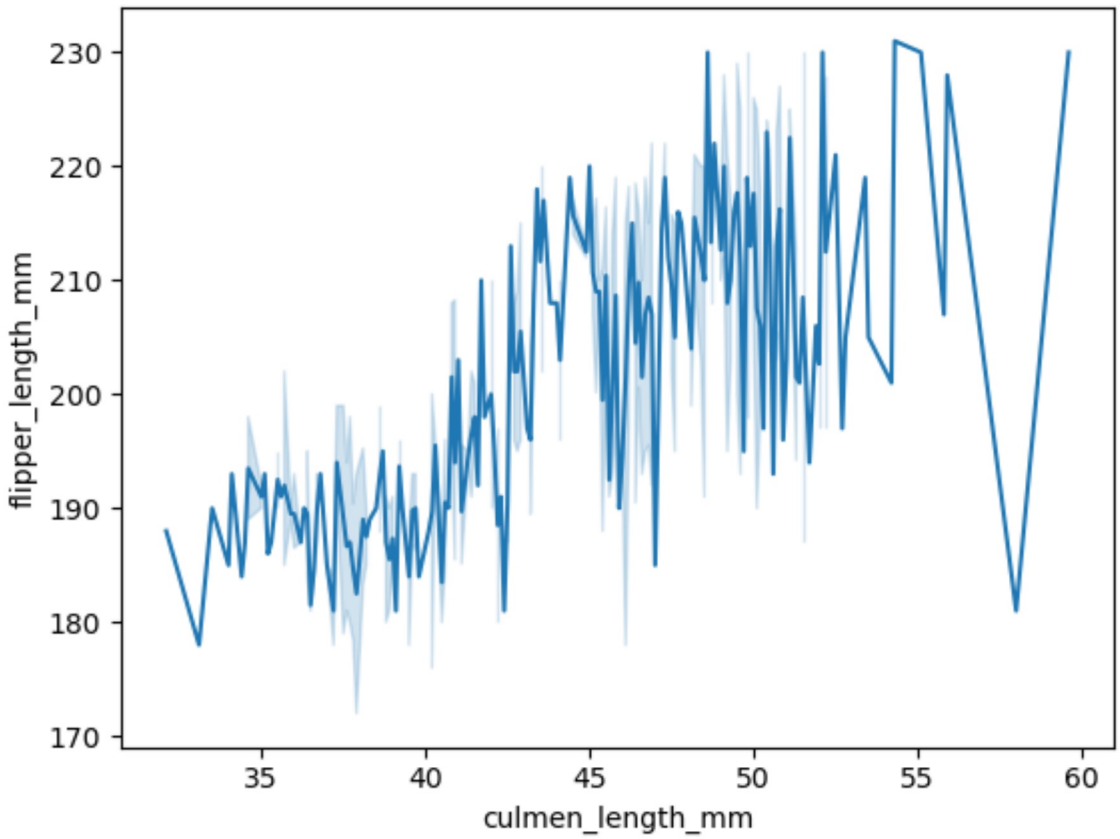
In [19]:
```
sns.lineplot(x=df.culmen_length_mm,y=df.culmen_depth_mm)
```

Out[19]:
```
<Axes: xlabel='culmen_length_mm', ylabel='culmen_depth_mm'>
```



In [20]:
```
sns.lineplot(x=df.culmen_length_mm,y=df.flipper_length_mm)
```

Out[20]:
```
<Axes: xlabel='culmen_length_mm', ylabel='flipper_length_mm'>
```

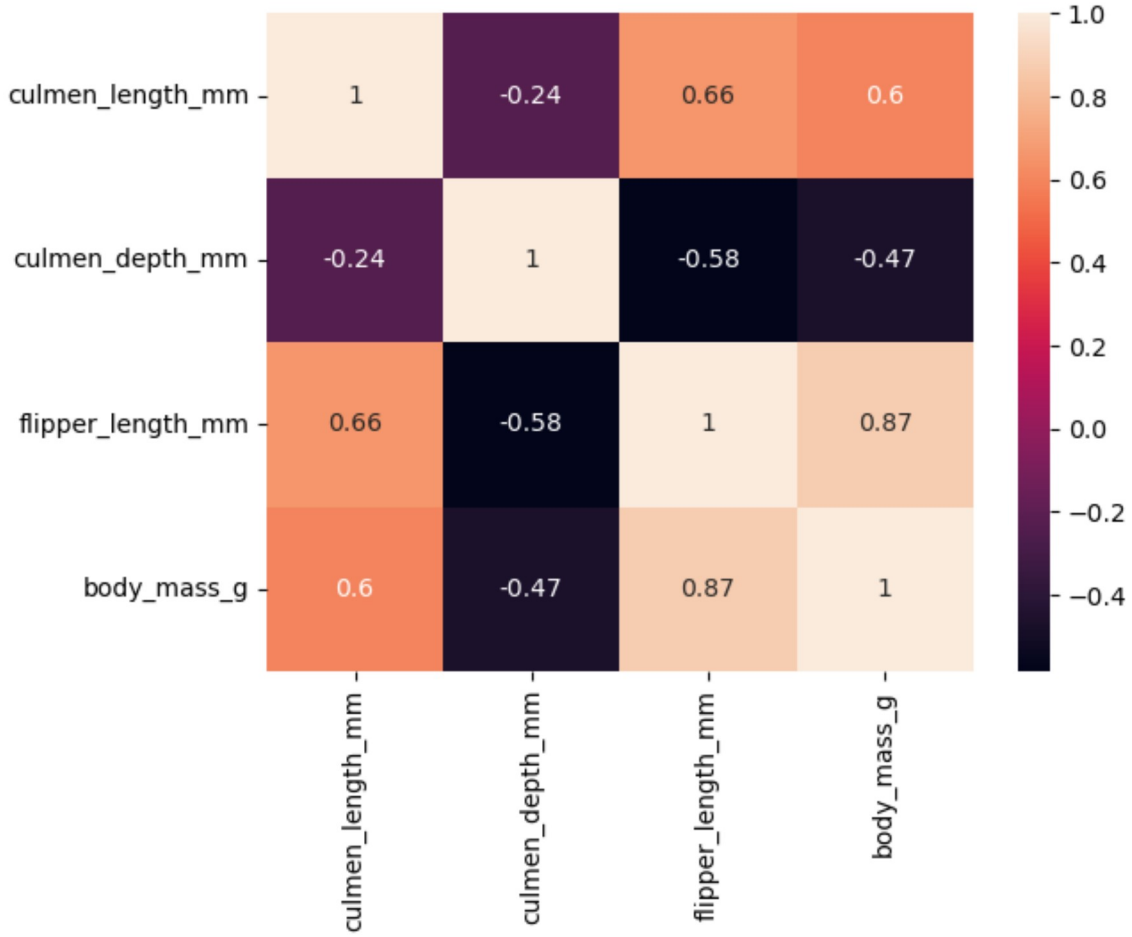



# MultiVariate Analysis

In [21]:
```
sns.heatmap(df.corr(),annot=True)
```

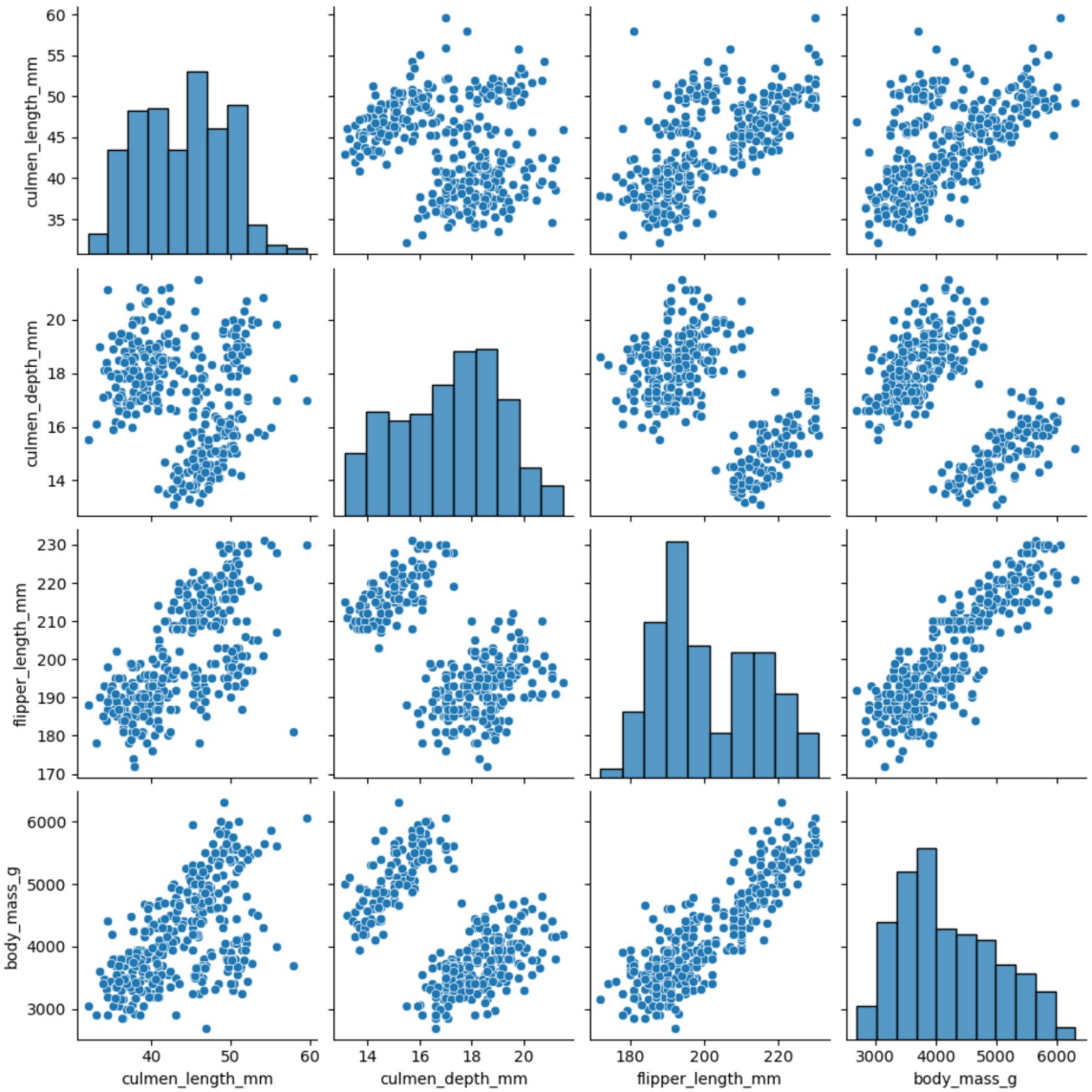
```
C:\Users\utkar\AppData\Local\Temp\ipykernel_15528\4277794465.py:1: FutureWarnin
g: The default value of numeric_only in DataFrame.corr is deprecated. In a futur
e version, it will default to False. Select only valid columns or specify the va
lue of numeric_only to silence this warning.
  sns.heatmap(df.corr(),annot=True)
```

Out[21]:
```
<Axes: >
```




In [22]:
```
sns.pairplot(df)
```

Out[22]:
```
<seaborn.axisgrid.PairGrid at 0x2e23be21540>
```




In [23]:
```
#descriptive statistics
df.describe()
```

Out[23]:

| | culmen_length_mm | culmen_depth_mm | flipper_length_mm | body_mass_g |
|---|---|---|---|---|
| **count** | 342.000000 | 342.000000 | 342.000000 | 342.000000 |
| **mean** | 43.921930 | 17.151170 | 200.915205 | 4201.754386 |
| **std** | 5.459584 | 1.974793 | 14.061714 | 801.954536 |
| **min** | 32.100000 | 13.100000 | 172.000000 | 2700.000000 |
| **25%** | 39.225000 | 15.600000 | 190.000000 | 3550.000000 |
| **50%** | 44.450000 | 17.300000 | 197.000000 | 4050.000000 |
| **75%** | 48.500000 | 18.700000 | 213.000000 | 4750.000000 |
| **max** | 59.600000 | 21.500000 | 231.000000 | 6300.000000 |

In [24]:
```
#check missing values and replace
df.isnull().sum()
```

Out[24]:
```
species               0
island                0
culmen_length_mm      2
culmen_depth_mm       2
flipper_length_mm     2
body_mass_g           2
sex                  10
dtype: int64
```

In [25]:
```
df['culmen_length_mm'].fillna(df['culmen_length_mm'].median(),inplace=True)
df['culmen_depth_mm'].fillna(df['culmen_depth_mm'].median(),inplace=True)
df['flipper_length_mm'].fillna(df['flipper_length_mm'].median(),inplace=True)
df['body_mass_g'].fillna(df['body_mass_g'].median(),inplace=True)
```

In [26]:
```
df['sex']=df['sex'].replace('.','MALE')
```

In [27]:
```
df.sex.value_counts()
```

Out[27]:
```
MALE      169
FEMALE    165
Name: sex, dtype: int64
```

In [28]:
```
df['sex'].fillna('MALE',inplace=True)
```

In [29]:
```
df.isnull().sum()
```

Out[29]:
```
species              0
island               0
culmen_length_mm     0
culmen_depth_mm      0
flipper_length_mm    0
body_mass_g          0
sex                  0
dtype: int64
```

In [30]:
```
df.island.value_counts()
```

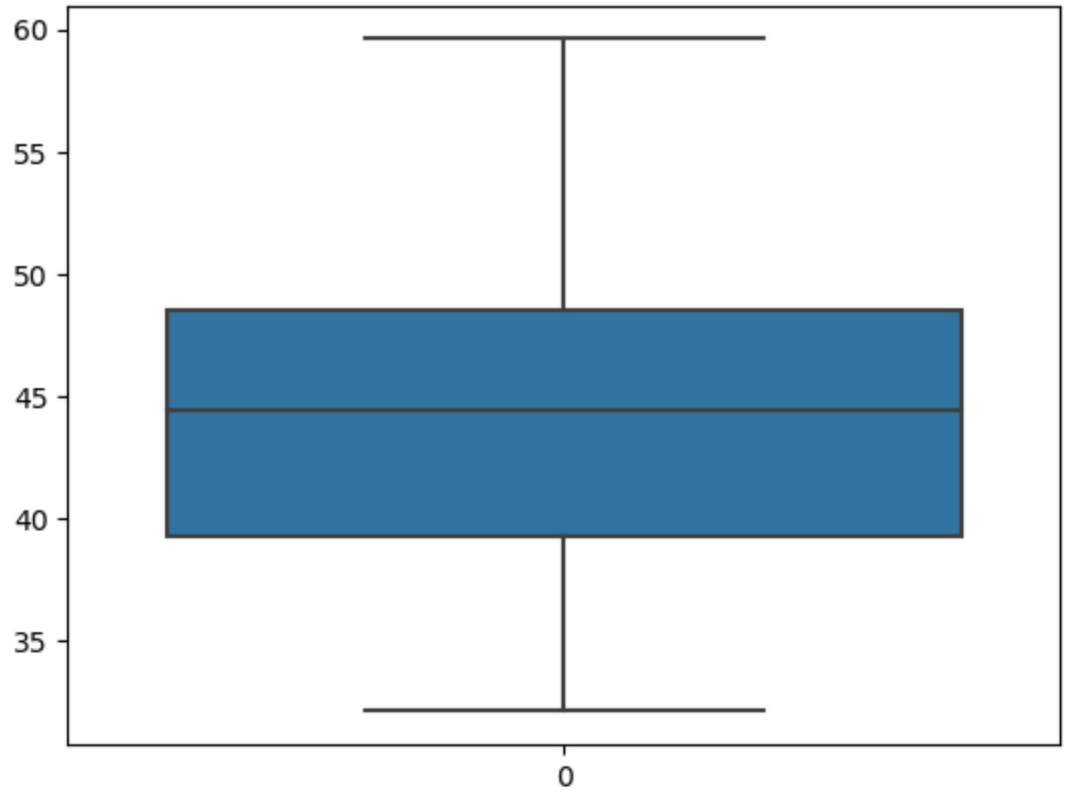
Out[30]:
```
Biscoe       168
Dream        124
Torgersen     52
Name: island, dtype: int64
```

In [31]:
```
df.species.value_counts()
```

Out[31]:
```
Adelie       152
Gentoo       124
Chinstrap     68
Name: species, dtype: int64
```

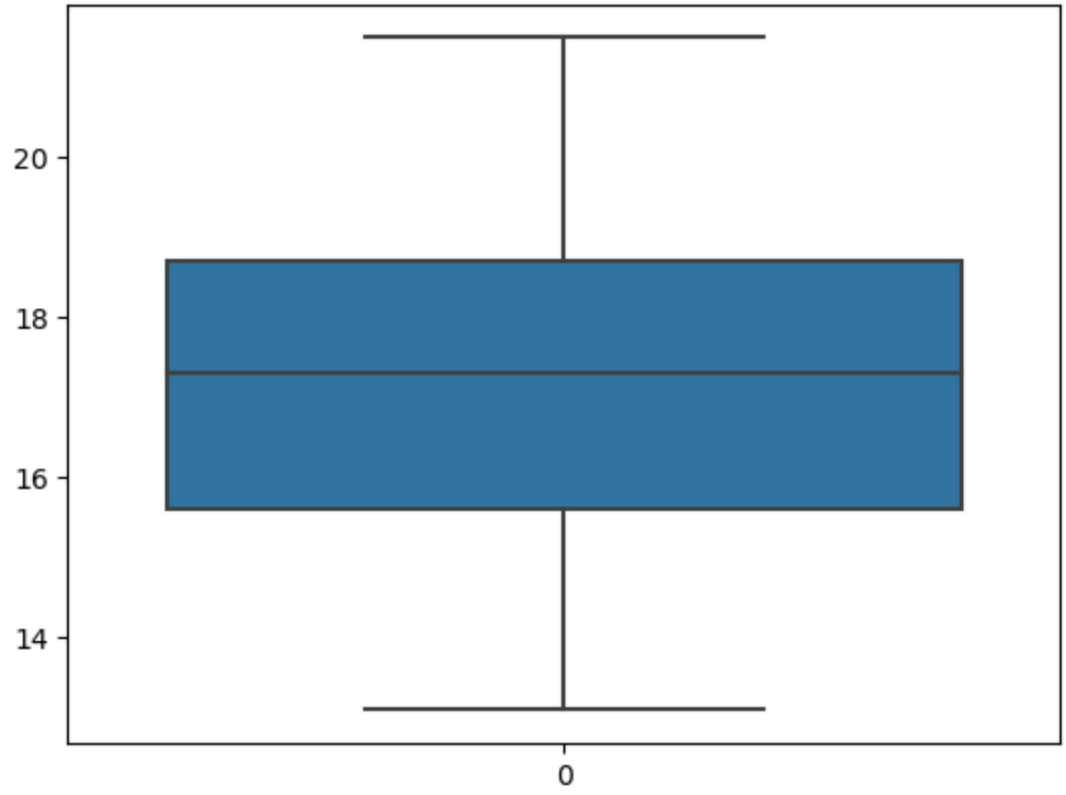
In [32]:
```
#find and replace outliers
sns.boxplot(df.culmen_length_mm)
```

Out[32]:
```
<Axes: >
```



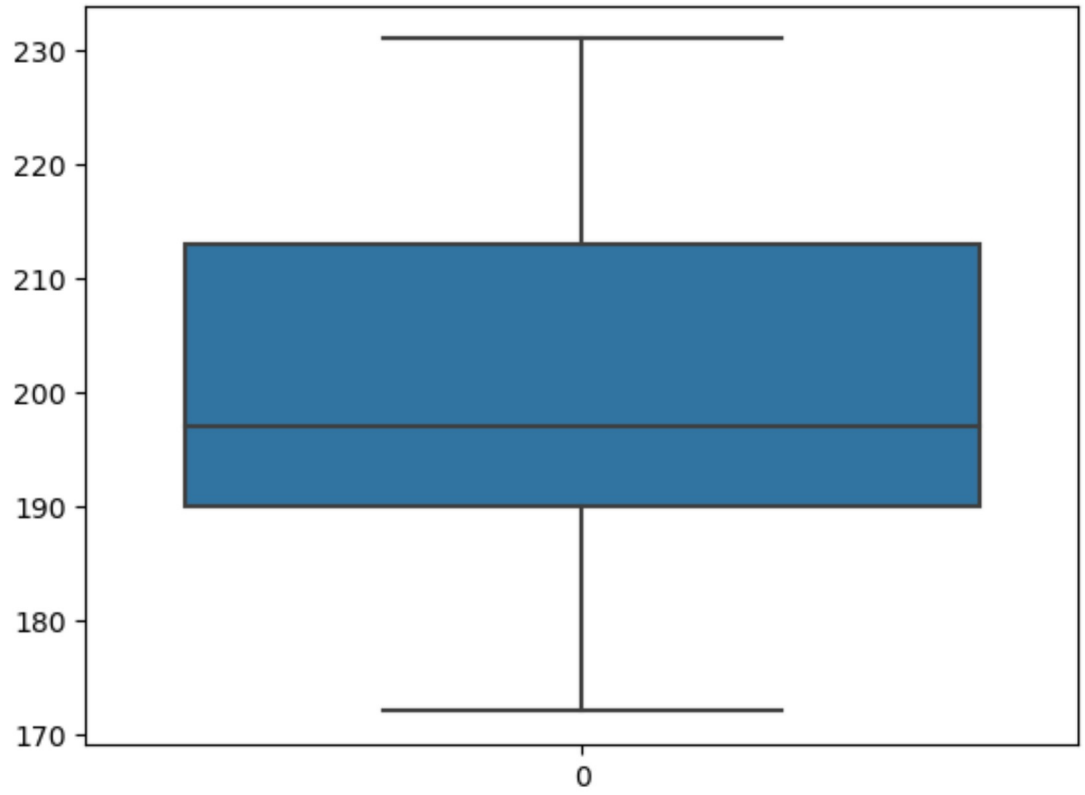

In [33]:
```
sns.boxplot(df.culmen_depth_mm)
```

Out[33]: <Axes: >



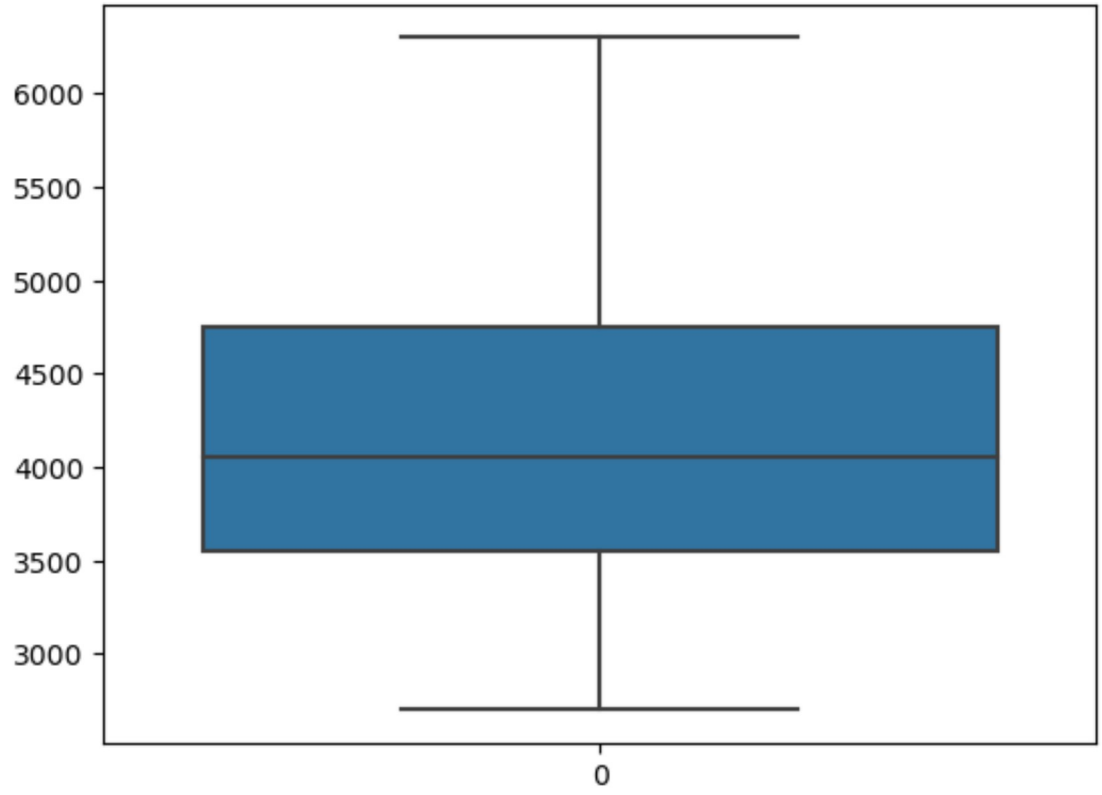

In [34]:
```
sns.boxplot(df.flipper_length_mm)
```

Out[34]: <Axes: >




In [35]:
```
sns.boxplot(df.body_mass_g)
```

Out[35]: <Axes: >




In [36]:
```
#correlation
df.corr()
```

```
C:\Users\utkar\AppData\Local\Temp\ipykernel_15528\3302858938.py:2: FutureWarnin
g: The default value of numeric_only in DataFrame.corr is deprecated. In a futur
e version, it will default to False. Select only valid columns or specify the va
lue of numeric_only to silence this warning.
  df.corr()
```

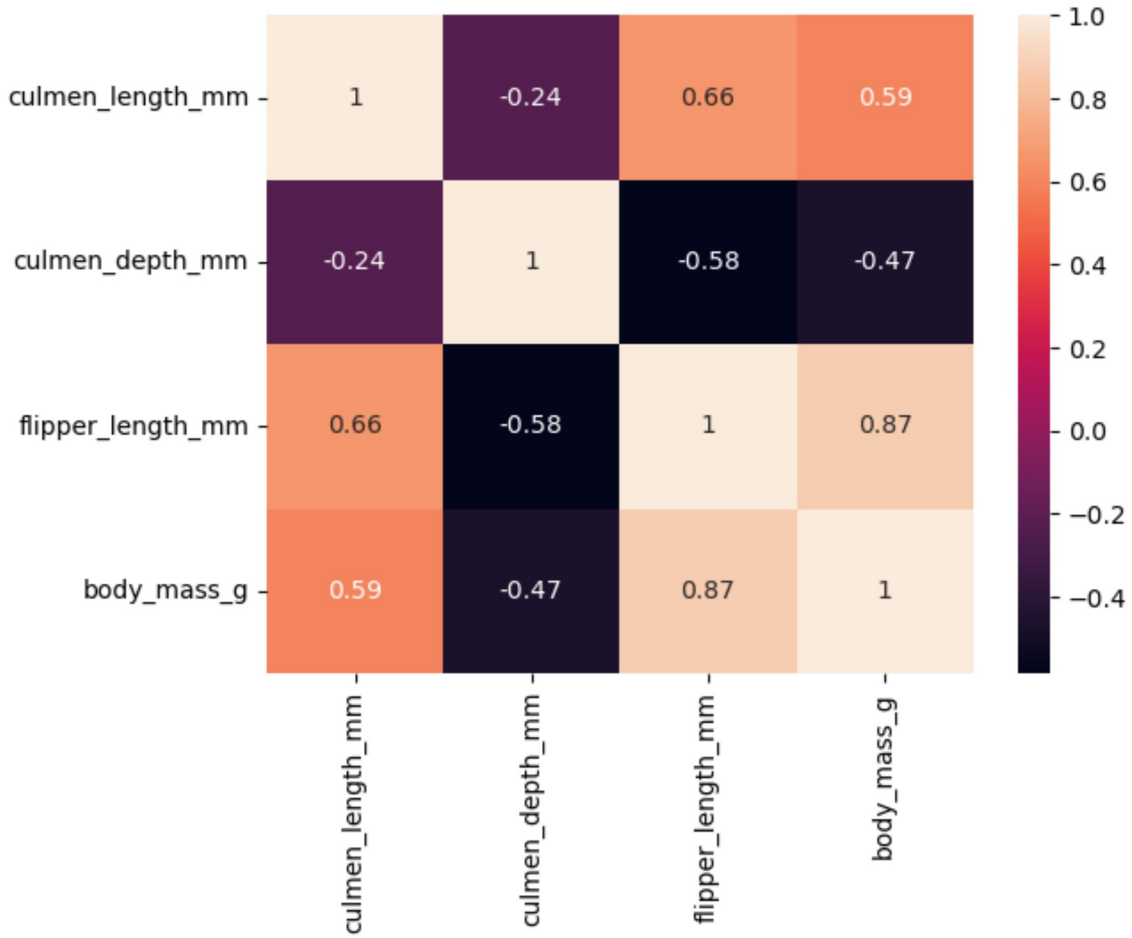
Out[36]:

| | culmen_length_mm | culmen_depth_mm | flipper_length_mm | body_mass_g |
|---|---|---|---|---|
| **culmen_length_mm** | 1.000000 | -0.235000 | 0.655858 | 0.594925 |
| **culmen_depth_mm** | -0.235000 | 1.000000 | -0.583832 | -0.471942 |
| **flipper_length_mm** | 0.655858 | -0.583832 | 1.000000 | 0.871221 |
| **body_mass_g** | 0.594925 | -0.471942 | 0.871221 | 1.000000 |

In [37]:
```
sns.heatmap(df.corr(),annot=True)
```

```
C:\Users\utkar\AppData\Local\Temp\ipykernel_15528\4277794465.py:1: FutureWarnin
g: The default value of numeric_only in DataFrame.corr is deprecated. In a futur
e version, it will default to False. Select only valid columns or specify the va
lue of numeric_only to silence this warning.
  sns.heatmap(df.corr(),annot=True)
```

Out[37]:
```
<Axes: >
```




In [38]:
```
#encoding
from sklearn.preprocessing import LabelEncoder
le = LabelEncoder()
```

In [39]:
```
df.species=le.fit_transform(df.species)
df.sex=le.fit_transform(df.sex)
df.island=le.fit_transform(df.island)
```

In [40]:
```
df.head()
```

Out[40]:

| | species | island | culmen_length_mm | culmen_depth_mm | flipper_length_mm | body_mass_g | se |
|---|---|---|---|---|---|---|---|
| **0** | 0 | 2 | 39.10 | 18.7 | 181.0 | 3750.0 | |
| **1** | 0 | 2 | 39.50 | 17.4 | 186.0 | 3800.0 | |
| **2** | 0 | 2 | 40.30 | 18.0 | 195.0 | 3250.0 | |
| **3** | 0 | 2 | 44.45 | 17.3 | 197.0 | 4050.0 | |
| **4** | 0 | 2 | 36.70 | 19.3 | 193.0 | 3450.0 | |

In [41]:
```
#split data in independent and dependent variable
X = df.drop(columns =['species'],axis=1)
X.head()
```

Out[41]:

| | island | culmen_length_mm | culmen_depth_mm | flipper_length_mm | body_mass_g | sex |
|---|---|---|---|---|---|---|
| **0** | 2 | 39.10 | 18.7 | 181.0 | 3750.0 | 1 |
| **1** | 2 | 39.50 | 17.4 | 186.0 | 3800.0 | 0 |
| **2** | 2 | 40.30 | 18.0 | 195.0 | 3250.0 | 0 |
| **3** | 2 | 44.45 | 17.3 | 197.0 | 4050.0 | 1 |
| **4** | 2 | 36.70 | 19.3 | 193.0 | 3450.0 | 0 |

In [42]:
```
y = df.species
df.head()
```

Out[42]:

| | species | island | culmen_length_mm | culmen_depth_mm | flipper_length_mm | body_mass_g | se |
|---|---|---|---|---|---|---|---|
| **0** | 0 | 2 | 39.10 | 18.7 | 181.0 | 3750.0 | |
| **1** | 0 | 2 | 39.50 | 17.4 | 186.0 | 3800.0 | |
| **2** | 0 | 2 | 40.30 | 18.0 | 195.0 | 3250.0 | |
| **3** | 0 | 2 | 44.45 | 17.3 | 197.0 | 4050.0 | |
| **4** | 0 | 2 | 36.70 | 19.3 | 193.0 | 3450.0 | |

In [43]:
```
#scaling
from sklearn.preprocessing import StandardScaler
scale = StandardScaler()
```

In [44]:
```
X_scaled = pd.DataFrame(scale.fit_transform(X),columns=X.columns)
X_scaled.head()
```

Out[44]:

| | island | culmen_length_mm | culmen_depth_mm | flipper_length_mm | body_mass_g | sex |
|---|---|---|---|---|---|---|
| **0** | 1.844076 | -0.887622 | 0.787289 | -1.420541 | -0.564625 | 0.96009 |
| **1** | 1.844076 | -0.814037 | 0.126114 | -1.063485 | -0.502010 | -1.04156 |
| **2** | 1.844076 | -0.666866 | 0.431272 | -0.420786 | -1.190773 | -1.04156 |
| **3** | 1.844076 | 0.096581 | 0.075255 | -0.277964 | -0.188936 | 0.96009 |
| **4** | 1.844076 | -1.329133 | 1.092447 | -0.563608 | -0.940314 | -1.04156 |

In [45]:
```
#split data
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_s
```

In [46]:
```
#check training and testing data shape
```

In [47]:
```
x_train.shape
```

Out[47]:
```
(275, 6)
```

In [48]:
```
x_test.shape
```

Out[48]:
```
(69, 6)
```

In [49]:
```
y_train.shape
```

Out[49]:
```
(275,)
```

In [50]:
```
y_test.shape
```

Out[50]:
```
(69,)
```

In [ ]: