Step 1: Import the Required Libraries

In [1]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
warnings.filterwarnings("ignore")
```

Step 2: Load the Dataset's

In [2]:

```
penguins_iter_df = pd.read_csv('penguins_lter.csv')
penguins_size_df = pd.read_csv('penguins_size.csv')
```

Step 3: Explore the Datasets

In [3]:

```
# Display the first few rows of the DataFrames
penguins_iter_df.head()
```

Out[3]:

	studyName	Sample Number	Species	Region	Island	Stage	Individual ID	Clutch Completion	Dat Eg
0	PAL0708	1	Adelie Penguin (Pygoscelis adeliae)	Anvers	Torgersen	Adult, 1 Egg Stage	N1A1	Yes	11/11/0
1	PAL0708	2	Adelie Penguin (Pygoscelis adeliae)	Anvers	Torgersen	Adult, 1 Egg Stage	N1A2	Yes	11/11/0
2	PAL0708	3	Adelie Penguin (Pygoscelis adeliae)	Anvers	Torgersen	Adult, 1 Egg Stage	N2A1	Yes	11/16/0
3	PAL0708	4	Adelie Penguin (Pygoscelis adeliae)	Anvers	Torgersen	Adult, 1 Egg Stage	N2A2	Yes	11/16/0
4	PAL0708	5	Adelie Penguin (Pygoscelis adeliae)	Anvers	Torgersen	Adult, 1 Egg Stage	N3A1	Yes	11/16/0

In [4]:

```
penguins_size_df.head()
```

Out[4]:

	species	island	culmen_length_mm	culmen_depth_mm	flipper_length_mm	body_mass_
0	Adelie	Torgersen	39.1	18.7	181.0	3750
1	Adelie	Torgersen	39.5	17.4	186.0	3800
2	Adelie	Torgersen	40.3	18.0	195.0	3250
3	Adelie	Torgersen	NaN	NaN	NaN	Na
4	Adelie	Torgersen	36.7	19.3	193.0	3450

In [5]:

```
penguins_iter_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 344 entries, 0 to 343
Data columns (total 17 columns):
 #   Column               Non-Null Count  Dtype  
---  ------               --------------  -----  
 0   studyName            344 non-null    object 
 1   Sample Number        344 non-null    int64  
 2   Species              344 non-null    object 
 3   Region               344 non-null    object 
 4   Island               344 non-null    object 
 5   Stage                344 non-null    object 
 6   Individual ID        344 non-null    object 
 7   Clutch Completion    344 non-null    object 
 8   Date Egg             344 non-null    object 
 9   Culmen Length (mm)   342 non-null    float64
 10  Culmen Depth (mm)    342 non-null    float64
 11  Flipper Length (mm)  342 non-null    float64
 12  Body Mass (g)        342 non-null    float64
 13  Sex                  334 non-null    object 
 14  Delta 15 N (o/oo)    330 non-null    float64
 15  Delta 13 C (o/oo)    331 non-null    float64
 16  Comments             26 non-null     object 
dtypes: float64(6), int64(1), object(10)
memory usage: 45.8+ KB
```

In [6]:

```
penguins_size_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 344 entries, 0 to 343
Data columns (total 7 columns):
 #   Column             Non-Null Count  Dtype  
---  ------             --------------  -----  
 0   species            344 non-null    object 
 1   island             344 non-null    object 
 2   culmen_length_mm   342 non-null    float64
 3   culmen_depth_mm    342 non-null    float64
 4   flipper_length_mm  342 non-null    float64
 5   body_mass_g        342 non-null    float64
 6   sex                334 non-null    object 
dtypes: float64(4), object(3)
memory usage: 18.9+ KB
```

In [7]:

```
penguins_iter_df.describe()
```

Out[7]:

	Sample Number	Culmen Length (mm)	Culmen Depth (mm)	Flipper Length (mm)	Body Mass (g)	Delta 15 N (o/oo)	Delta 13 C (o/oo)
count	344.000000	342.000000	342.000000	342.000000	342.000000	330.000000	331.000000
mean	63.151163	43.921930	17.151170	200.915205	4201.754386	8.733382	-25.686292
std	40.430199	5.459584	1.974793	14.061714	801.954536	0.551770	0.793961
min	1.000000	32.100000	13.100000	172.000000	2700.000000	7.632200	-27.018540
25%	29.000000	39.225000	15.600000	190.000000	3550.000000	8.299890	-26.320305
50%	58.000000	44.450000	17.300000	197.000000	4050.000000	8.652405	-25.833520
75%	95.250000	48.500000	18.700000	213.000000	4750.000000	9.172123	-25.062050
max	152.000000	59.600000	21.500000	231.000000	6300.000000	10.025440	-23.787670

In [8]:

```
penguins_size_df.describe()
```

Out[8]:

	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

Step 4: Handle Missing Data

In [9]:

```
penguins_size_df.isnull().sum()
```

Out[9]:

```
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 [10]:

```
penguins_size_df = penguins_size_df.dropna()
```

In [11]:

```
penguins_size_df.isnull().sum()
```

Out[11]:

```
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 [12]:

```
# Check for missing values in Penguins iter dataset
print(penguins_iter_df.isnull().sum())
```

```
studyName              0
Sample Number          0
Species                0
Region                 0
Island                 0
Stage                  0
Individual ID          0
Clutch Completion      0
Date Egg               0
Culmen Length (mm)     2
Culmen Depth (mm)      2
Flipper Length (mm)    2
Body Mass (g)          2
Sex                   10
Delta 15 N (o/oo)     14
Delta 13 C (o/oo)     13
Comments             318
dtype: int64
```

In [13]:

```
penguins_iter_df = penguins_iter_df.dropna()
```

In [14]:

```
print(penguins_iter_df.isnull().sum())
```

```
studyName            0
Sample Number        0
Species              0
Region               0
Island               0
Stage                0
Individual ID        0
Clutch Completion    0
Date Egg             0
Culmen Length (mm)   0
Culmen Depth (mm)    0
Flipper Length (mm)  0
Body Mass (g)        0
Sex                  0
Delta 15 N (o/oo)    0
Delta 13 C (o/oo)    0
Comments             0
dtype: int64
```

Step 5: Data Cleaning

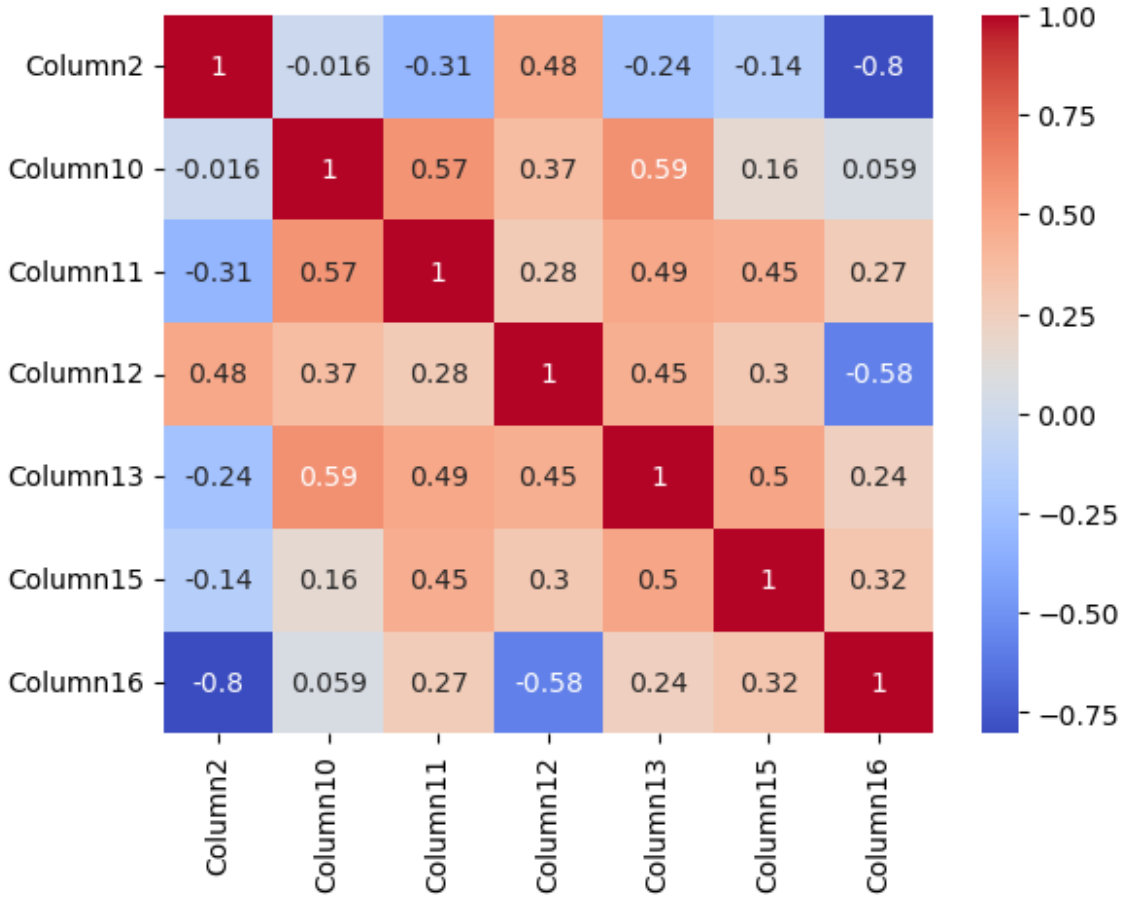
In [15]:

```
# Assign unique column names to Penguins Iter dataset
new_column_names = ['Column1', 'Column2', 'Column3', 'Column4', 'Column5', 'Column6', 'Co
penguins_iter_df.columns = new_column_names
```

Step 6: Exploratory Data Analysis

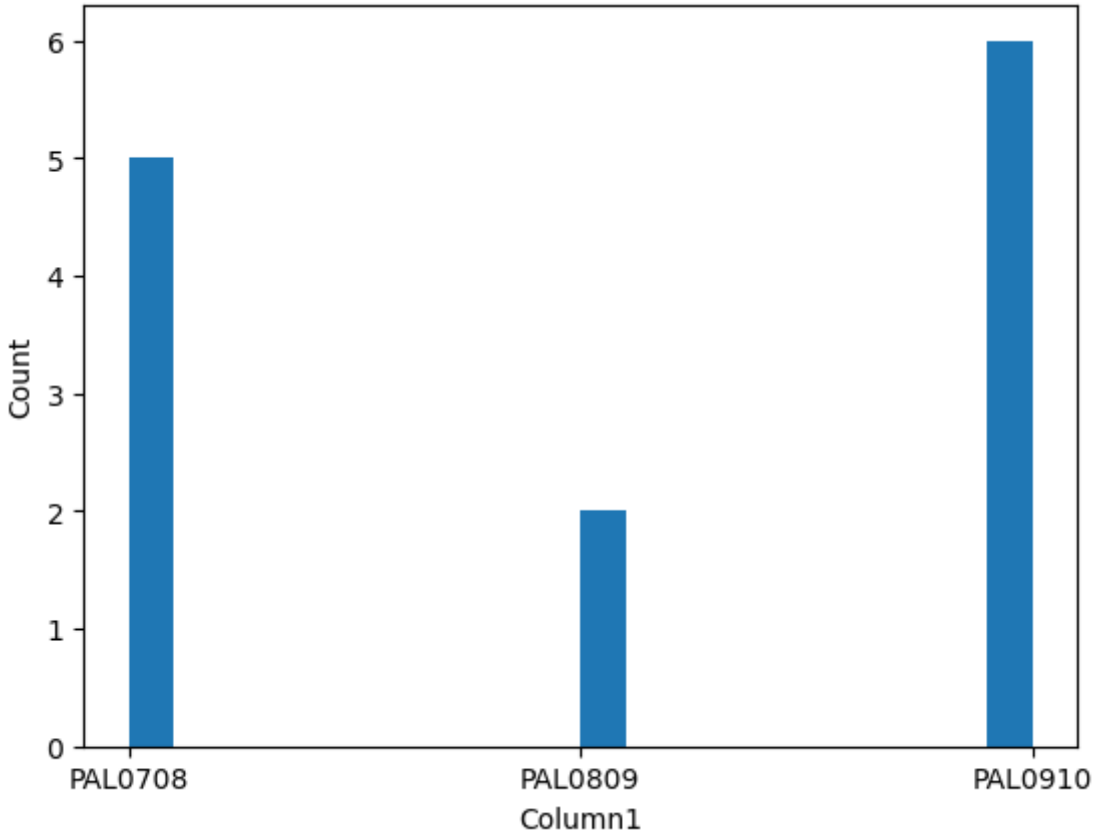
In [16]:

```
correlation = penguins_iter_df.corr()
sns.heatmap(correlation, annot=True, cmap='coolwarm')
plt.show()
```



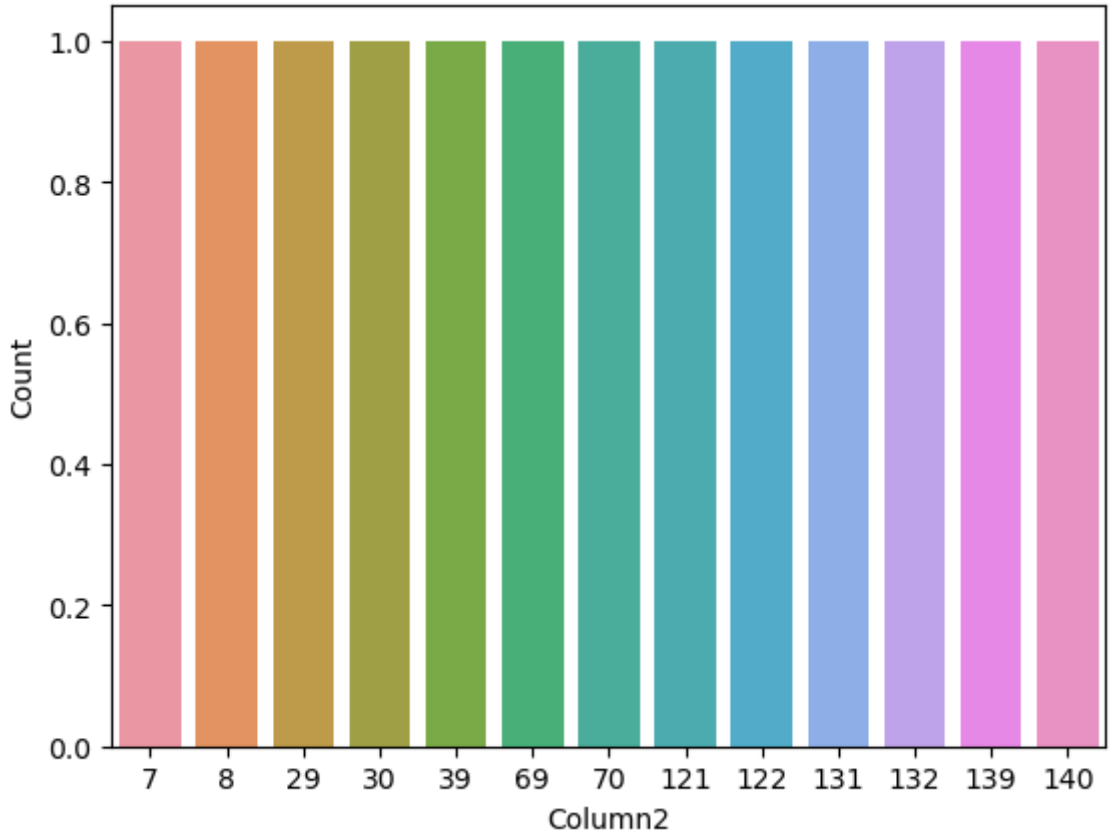
In [17]:

```
# Plot Different Features
plt.hist(penguins_iter_df['Column1'], bins=20)
plt.xlabel('Column1')
plt.ylabel('Count')
plt.show()
```



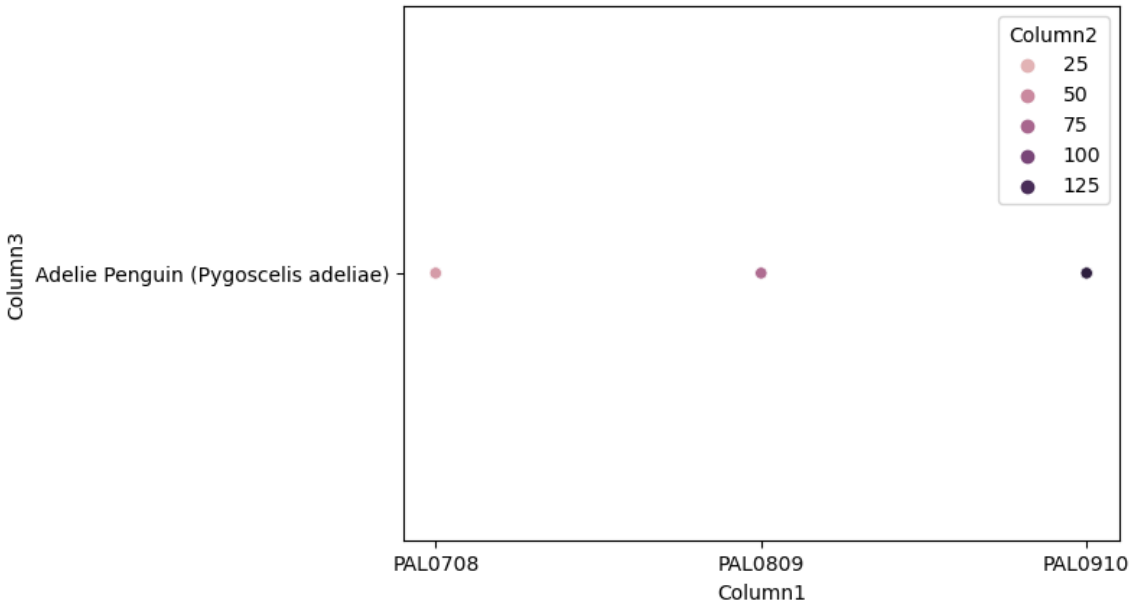
In [18]:

```
sns.countplot(x='Column2', data=penguins_iter_df)
plt.xlabel('Column2')
plt.ylabel('Count')
plt.show()
```



In [19]:

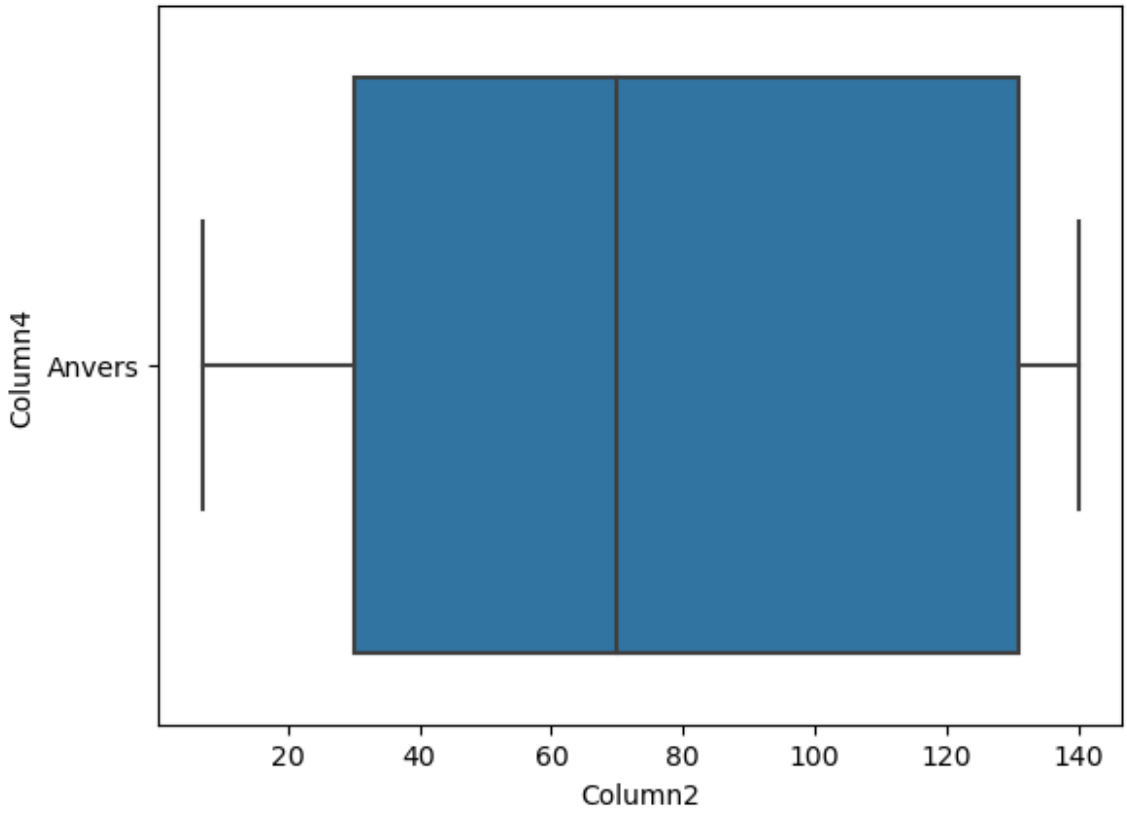
```
sns.scatterplot(x='Column1', y='Column3', hue='Column2', data=penguins_iter_df)
plt.xlabel('Column1')
plt.ylabel('Column3')
plt.show()
```



Detect Outliers / Missing Values

In [20]:

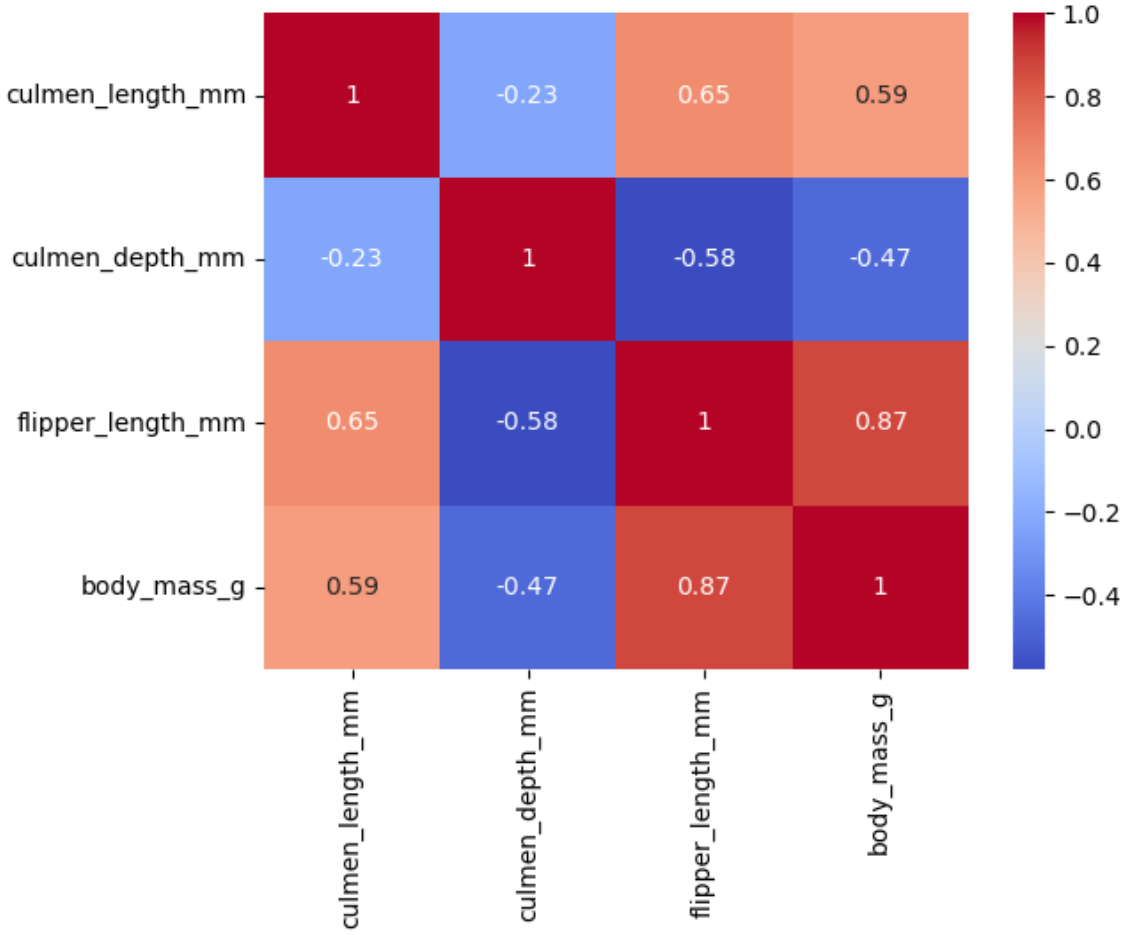
```
# Detect Outliers / Missing Values
sns.boxplot(x='Column2', y='Column4', data=penguins_iter_df)
plt.xlabel('Column2')
plt.ylabel('Column4')
plt.show()
```



Find Correlation Between Variables

In [21]:

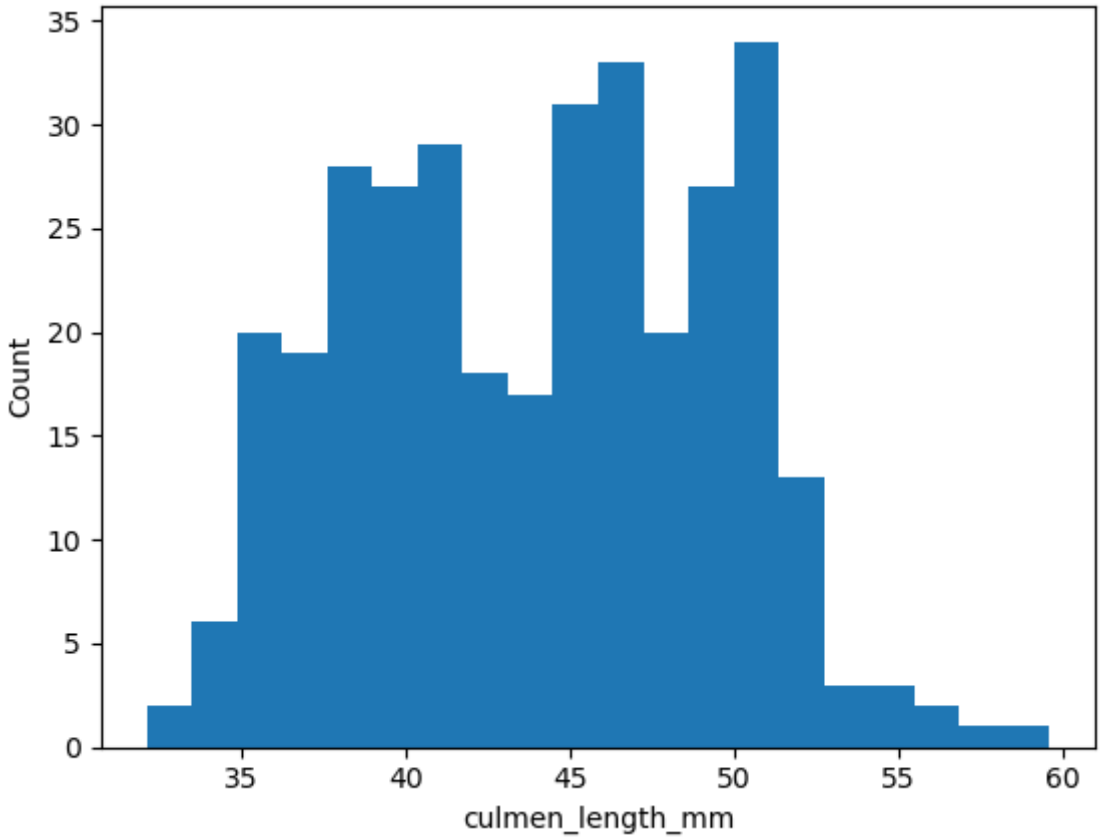
```
correlation = penguins_size_df.corr()
sns.heatmap(correlation, annot=True, cmap='coolwarm')
plt.show()
```



EDA ON PENGUINS SIZE

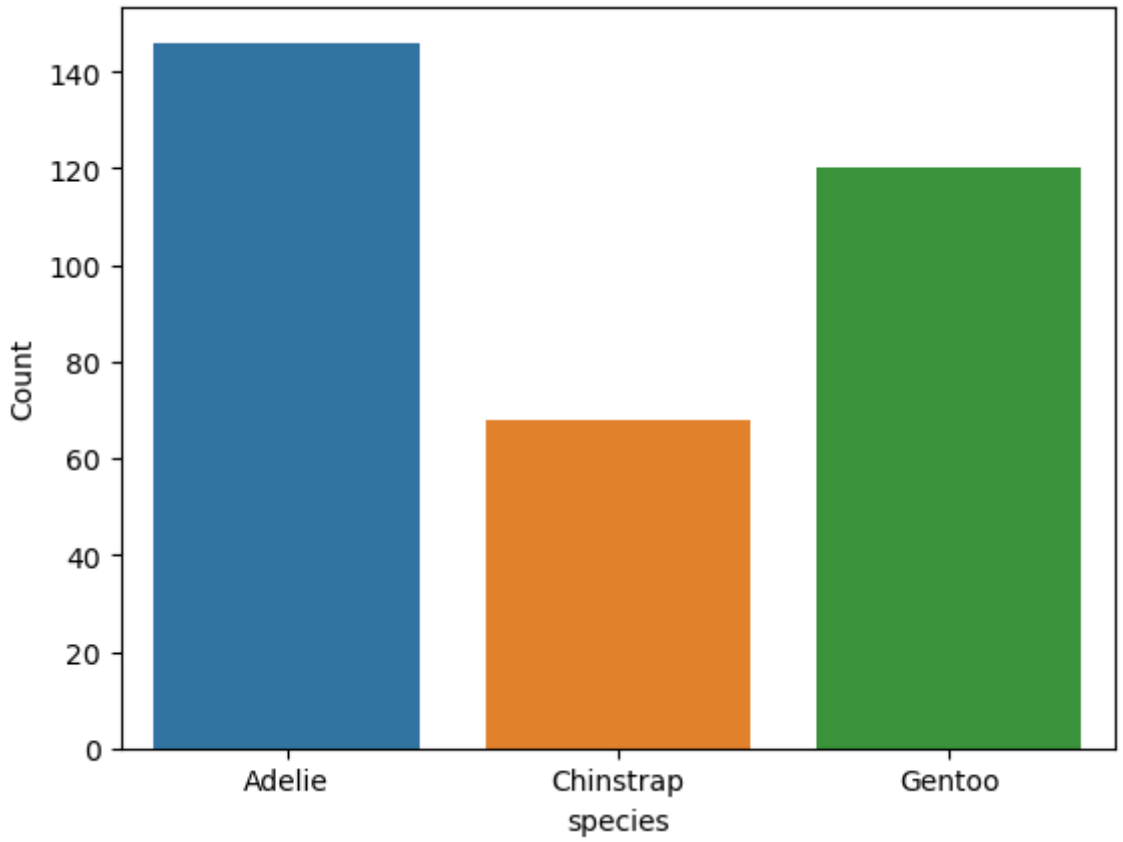
In [22]:

```
# Histograms
plt.hist(penguins_size_df['culmen_length_mm'], bins=20)
plt.xlabel('culmen_length_mm')
plt.ylabel('Count')
plt.show()
```



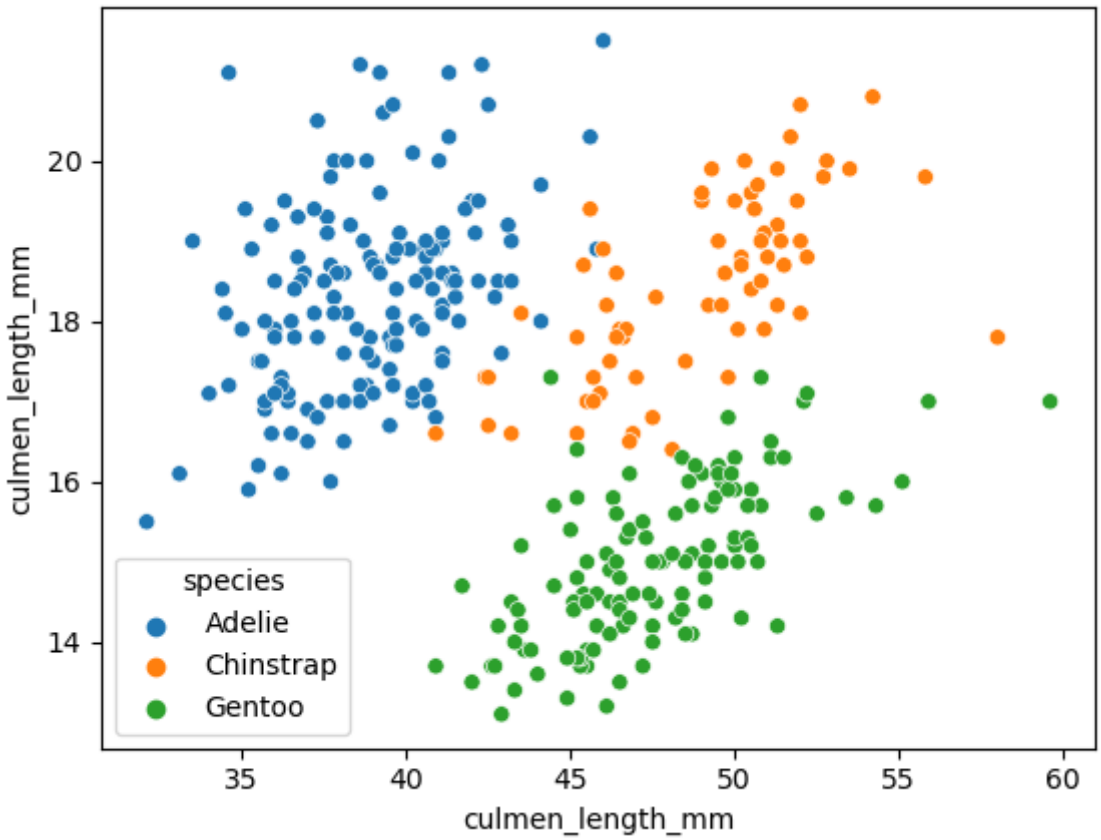
In [23]:

```
# Bar plots
sns.countplot(x='species', data=penguins_size_df)
plt.xlabel('species')
plt.ylabel('Count')
plt.show()
```



In [24]:

```
# Scatter plots
sns.scatterplot(x='culmen_length_mm', y='culmen_depth_mm', hue='species', data=penguins_s
plt.xlabel('culmen_length_mm')
plt.ylabel('culmen_length_mm')
plt.show()
```



Detect Outliers / Missing Values

In [25]:

```
# Detect outliers using box plots
sns.boxplot(x='species', y='body_mass_g', data=penguins_size_df)
plt.xlabel('species')
plt.ylabel('body_mass_g')
plt.show()
```