

```
In [ ]: import seaborn as sns
import ssl
ssl._create_default_https_context = ssl._create_unverified_context
```

```
In [ ]: data = sns.load_dataset("penguins")
```

```
In [ ]: data[:5]
```

```
Out[ ]:   species      island  bill_length_mm  bill_depth_mm  flipper_length_mm  body_mass_g  s
          0    Adelie  Torgersen         39.1           18.7            181.0       3750.0  Ma
          1    Adelie  Torgersen         39.5           17.4            186.0       3800.0  Fem
          2    Adelie  Torgersen         40.3           18.0            195.0       3250.0  Fem
          3    Adelie  Torgersen          NaN            NaN            NaN          NaN        NaN
          4    Adelie  Torgersen         36.7           19.3            193.0       3450.0  Fem
```

```
In [ ]: data.shape
```

```
Out[ ]: (344, 7)
```

```
In [ ]: # Set the Seaborn theme
sns.set_theme()

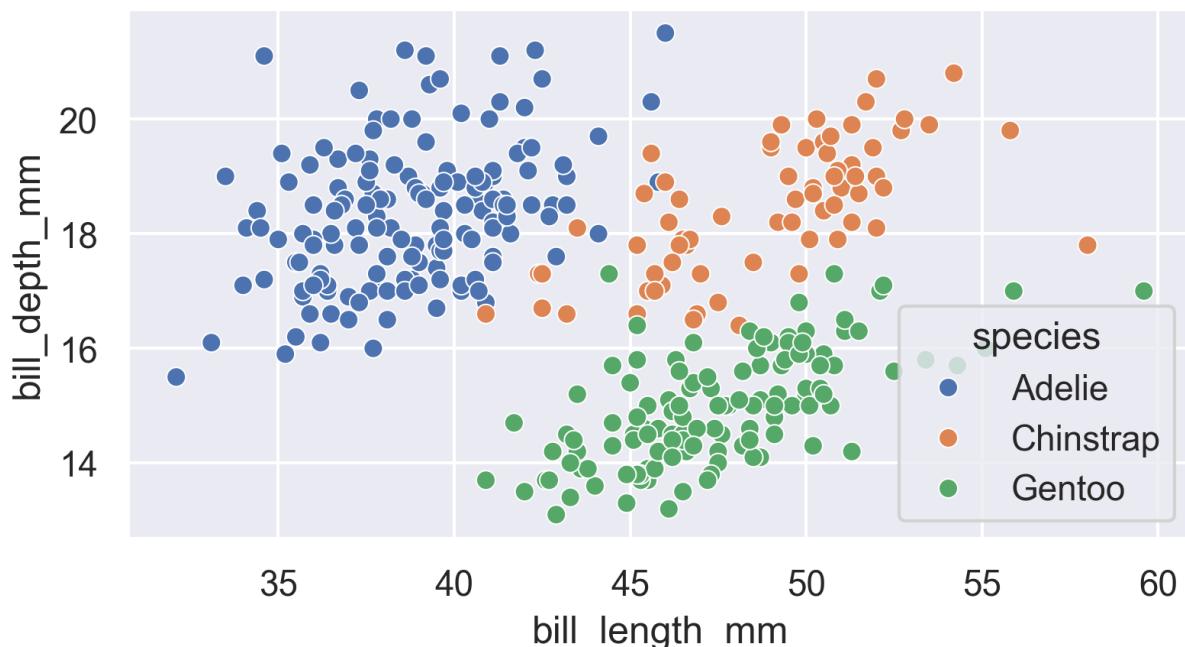
# Set the dpi (dots per inch) value of the graphs to 300
sns.set(rc={"figure.dpi": 300})

# Define the width and height of the graphs in inches. Here, width is set to
sns.set(rc={"figure.figsize": (6, 3)})
```

```
In [ ]: # Scatter plot
scatter = sns.scatterplot(x="bill_length_mm", y="bill_depth_mm", data=data, h
scatter.set_title("Bill Length vs Bill Depth")
```

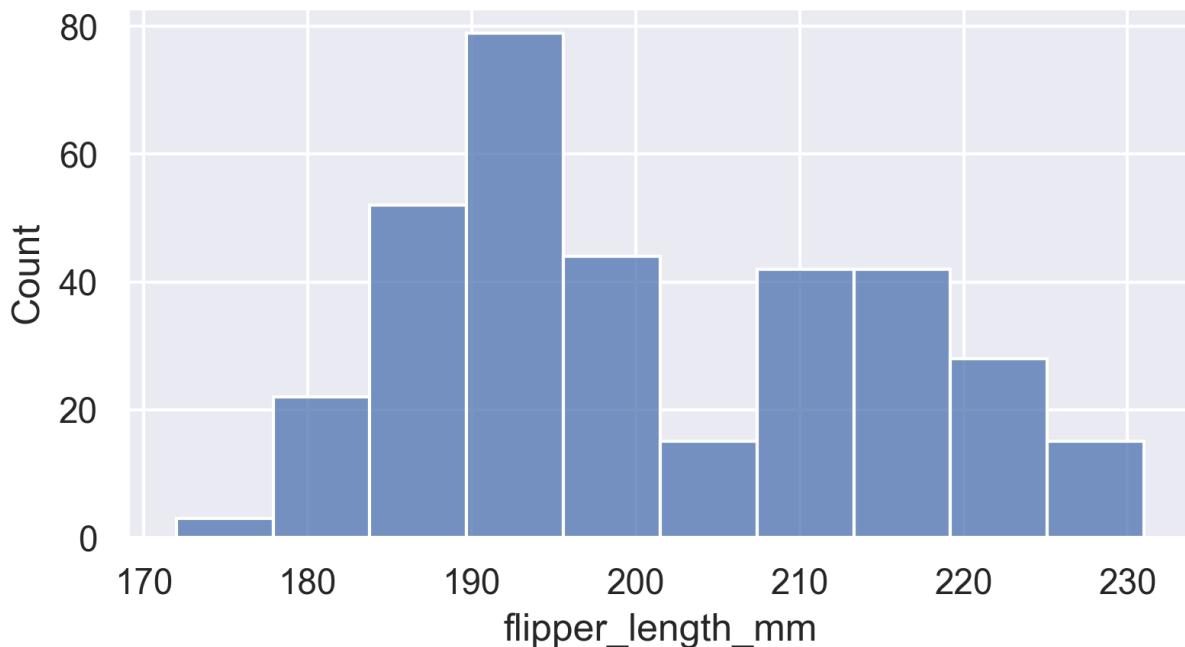
```
Out[ ]: Text(0.5, 1.0, 'Bill Length vs Bill Depth')
```

Bill Length vs Bill Depth



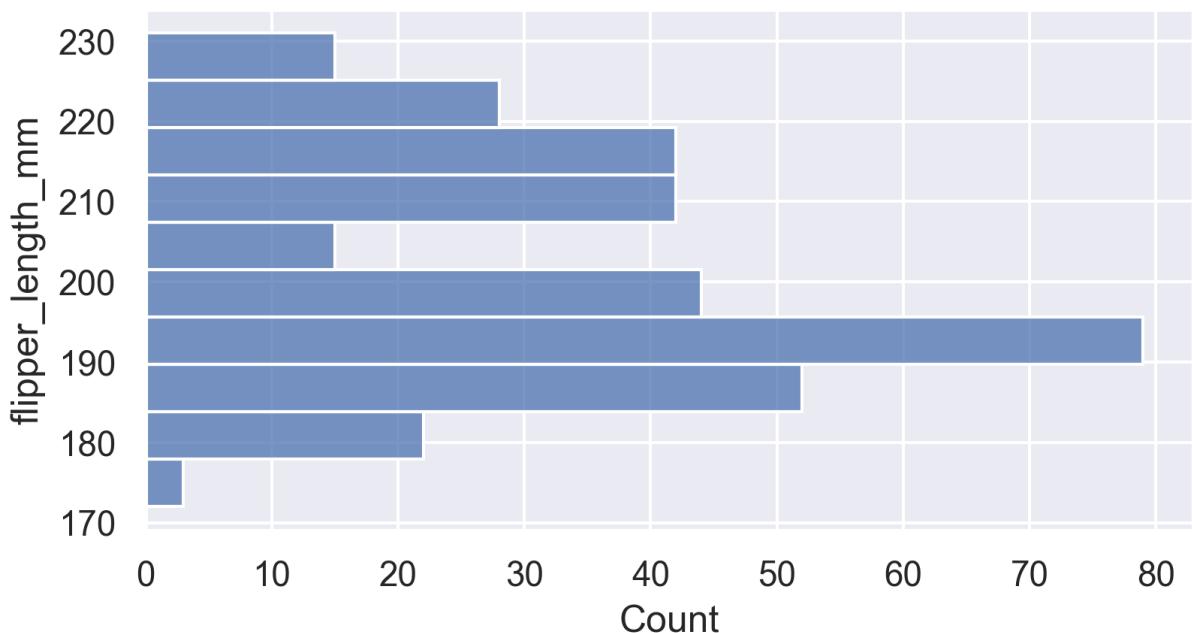
```
In [ ]: sns.histplot(x= "flipper_length_mm" , data=data)
#sns.set_theme(style='dark')
```

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



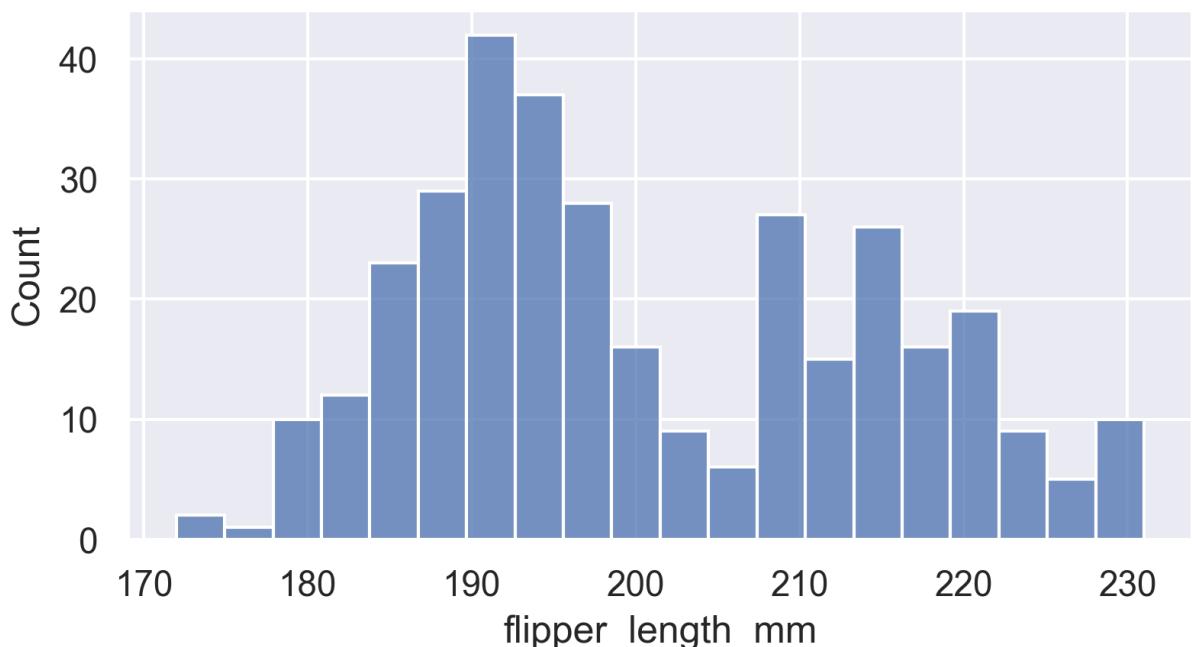
```
In [ ]: sns.histplot(y= "flipper_length_mm" , data=data)
```

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



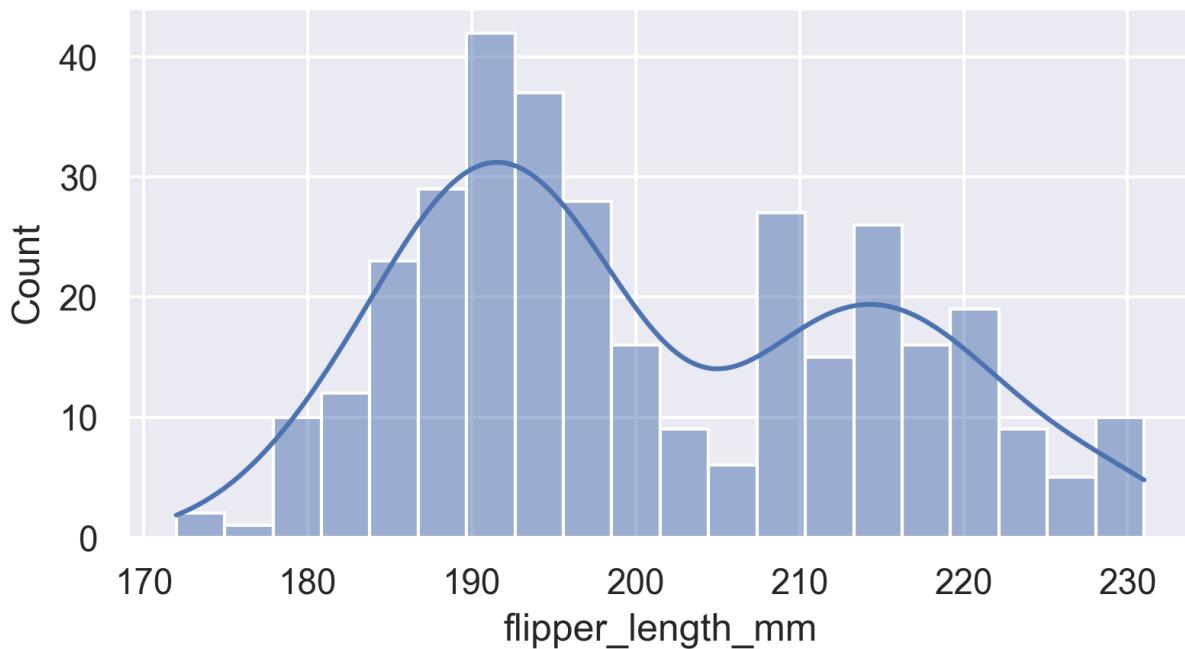
```
In [ ]: sns.histplot(x= "flipper_length_mm" , data=data,binwidth=3)
```

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



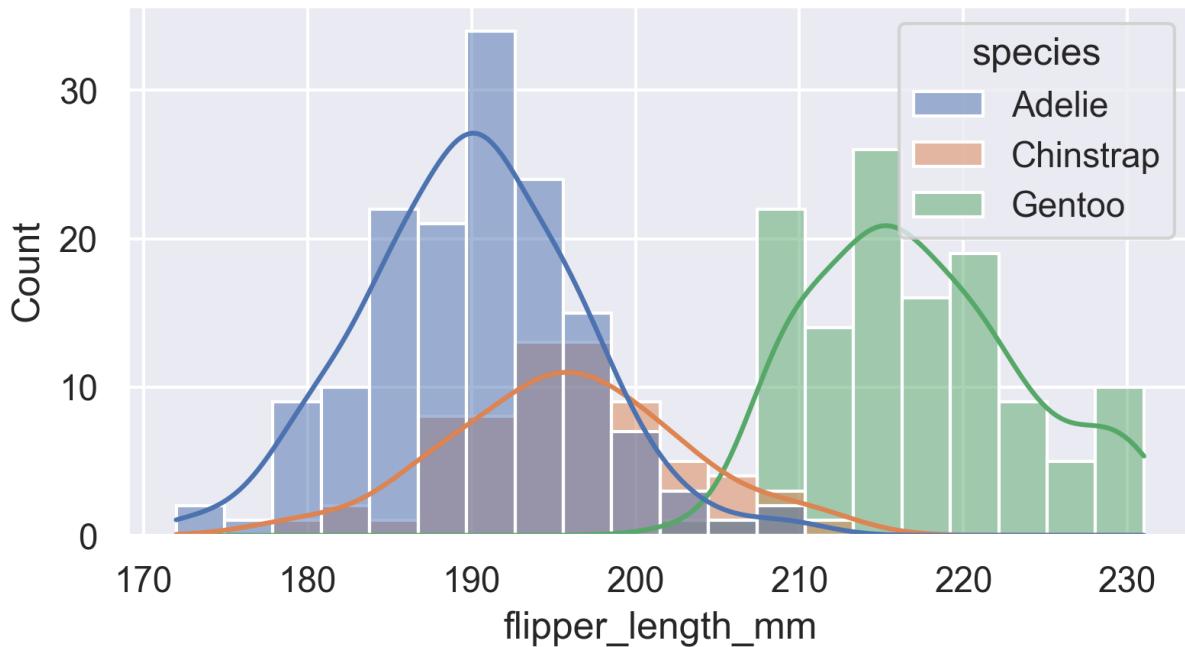
```
In [ ]: sns.histplot(x= "flipper_length_mm" , data=data,binwidth=3,kde=True)
```

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



```
In [ ]: sns.histplot(x= "flipper_length_mm" , data=data, binwidth=3,kde=True,hue="spec
```

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

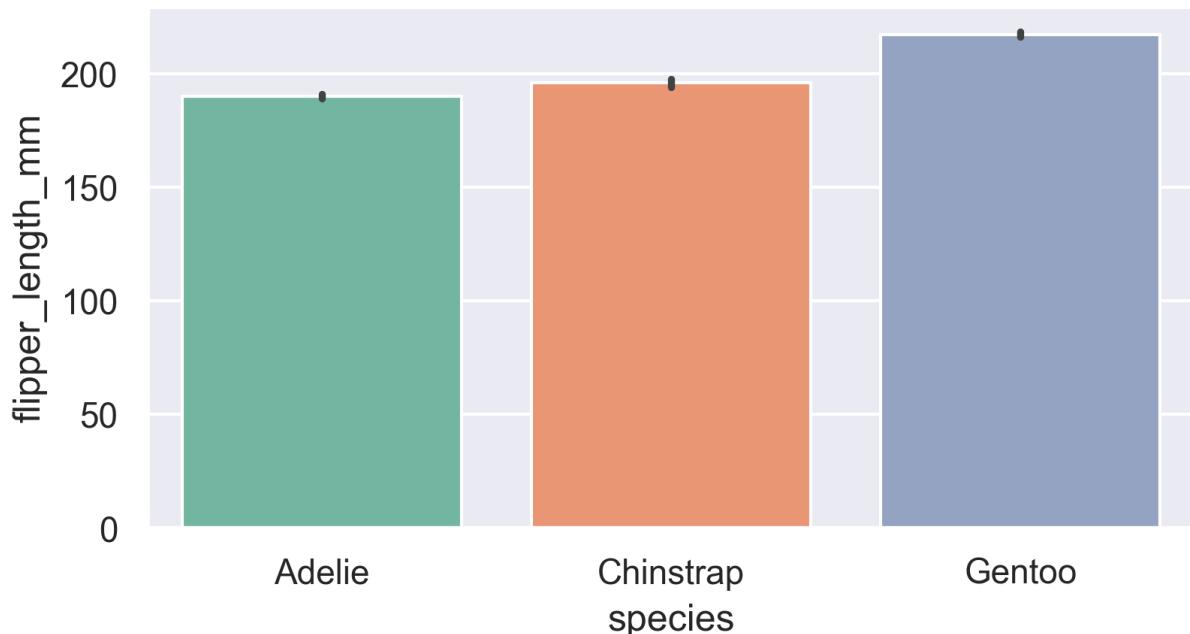


```
In [ ]: # Bar plot  
bar = sns.barplot(x="species", y="flipper_length_mm", data=data, palette="Set
```

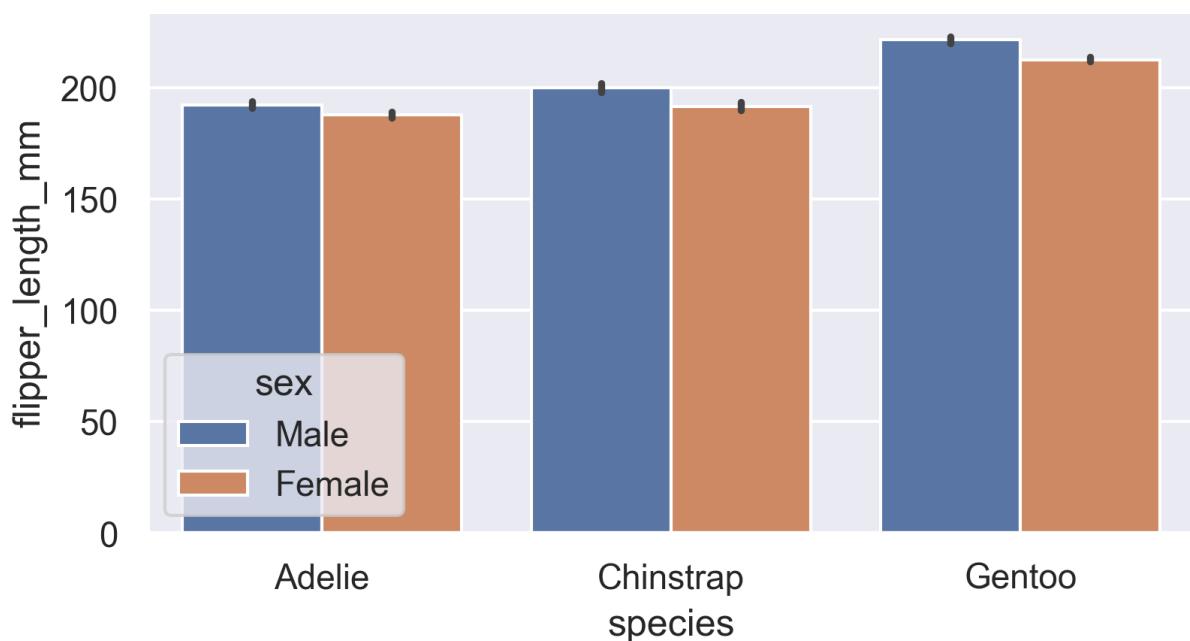
```
/var/folders/26/pvvz5dxx7lb978b1_d113_r40000gn/T/ipykernel_4463/3284859643.p  
y:3: FutureWarning:
```

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `x` variable to `hue` and set `legend=False` for the same effect.

```
bar = sns.barplot(x="species", y="flipper_length_mm", data=data, palette="S  
et2")
```



```
In [ ]: bar = sns.barplot(x="species", y="flipper_length_mm", data=data, hue="sex")
```



```
In [ ]: box = sns.boxplot(x="species", y="flipper_length_mm", data=data, palette="Set2")
box.set_title("Flipper Length by Species")
```

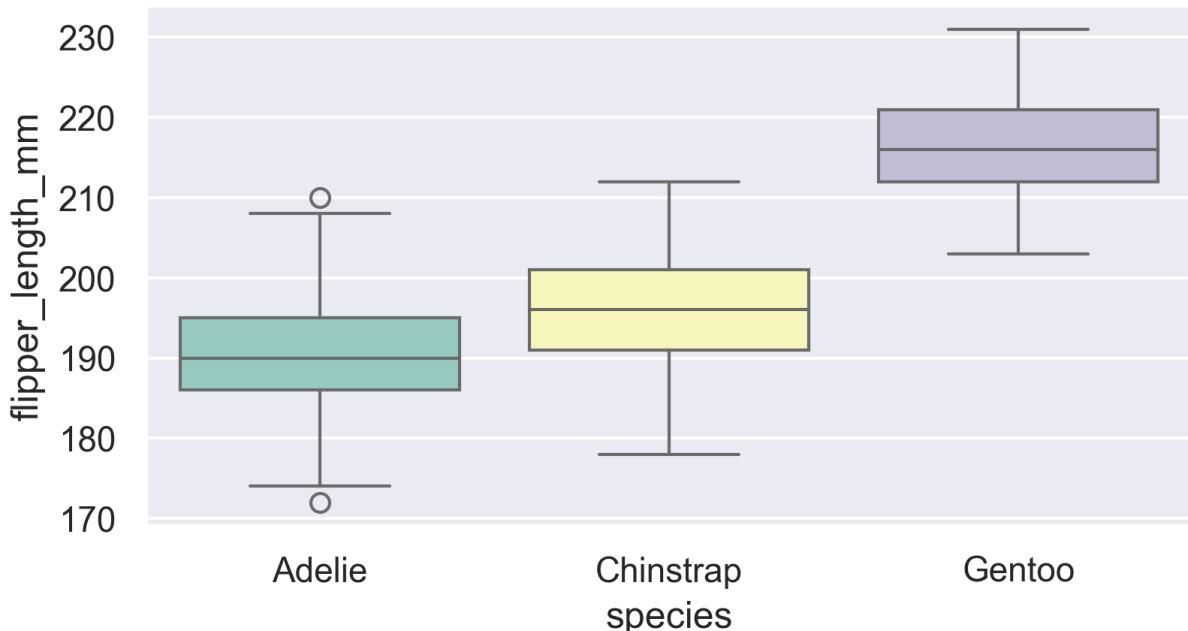
/var/folders/26/pvvz5dxx7lb978b1_d113_r4000gn/T/ipykernel_4463/345476460.py:
1: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `x` variable to `hue` and set `legend=False` for the same effect.

```
box = sns.boxplot(x="species", y="flipper_length_mm", data=data, palette="Set3")
```

Out[]: Text(0.5, 1.0, 'Flipper Length by Species')

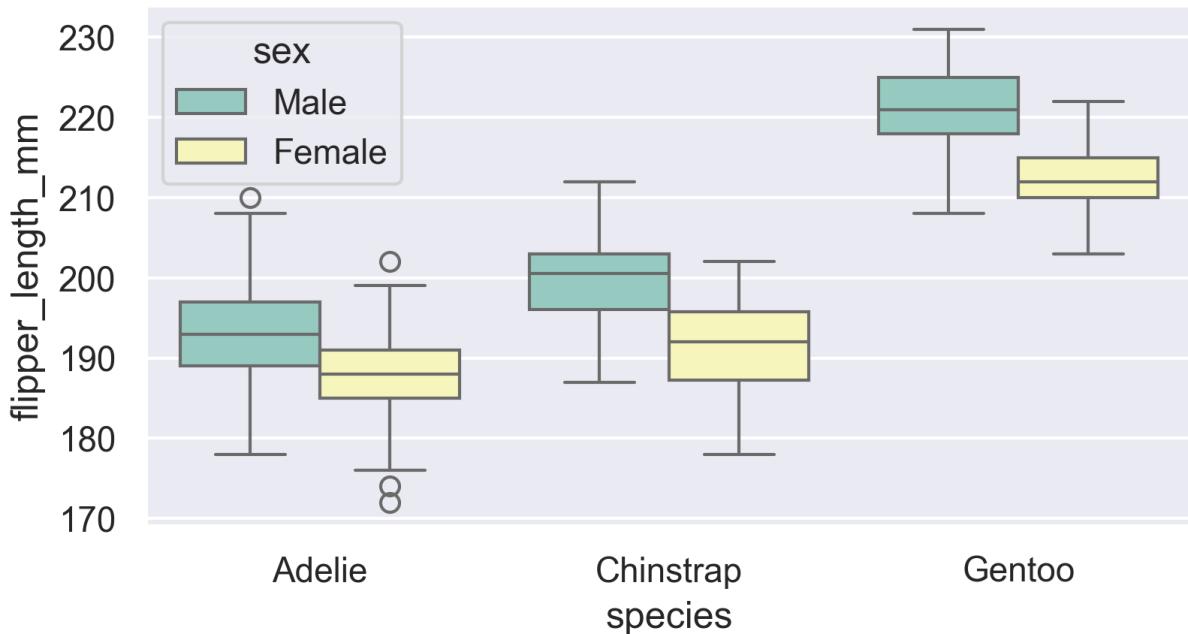
Flipper Length by Species



```
In [ ]: box = sns.boxplot(x="species", y="flipper_length_mm", data=data, palette="Set2")
box.set_title("Flipper Length by Species") # Grafik başlığı
```

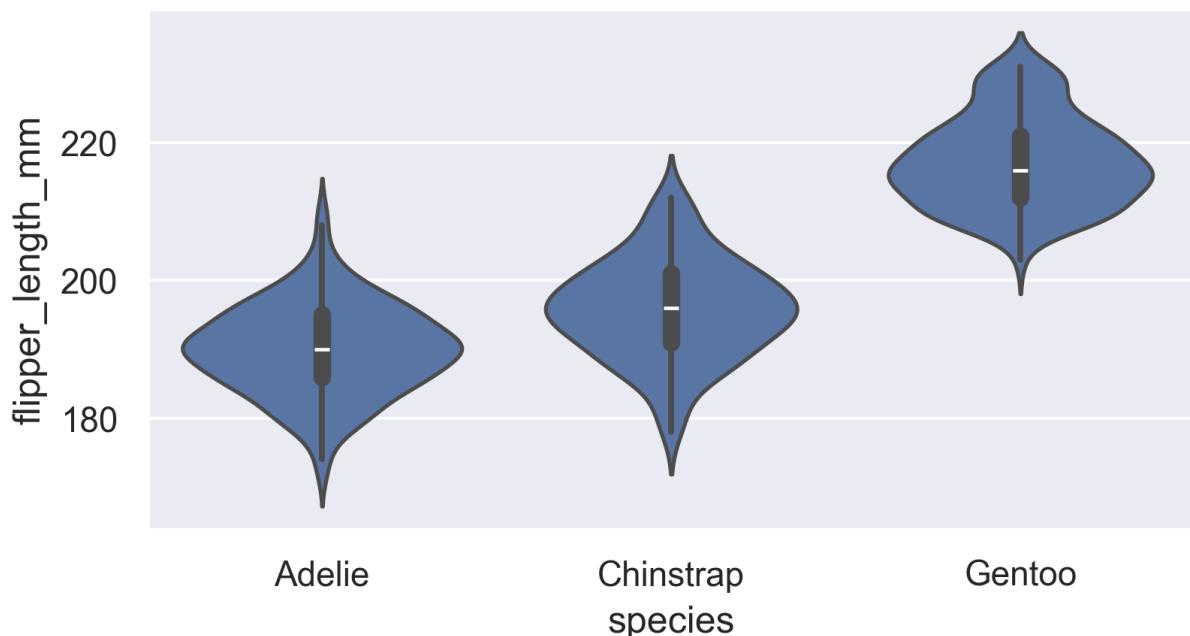
```
Out[ ]: Text(0.5, 1.0, 'Flipper Length by Species')
```

Flipper Length by Species



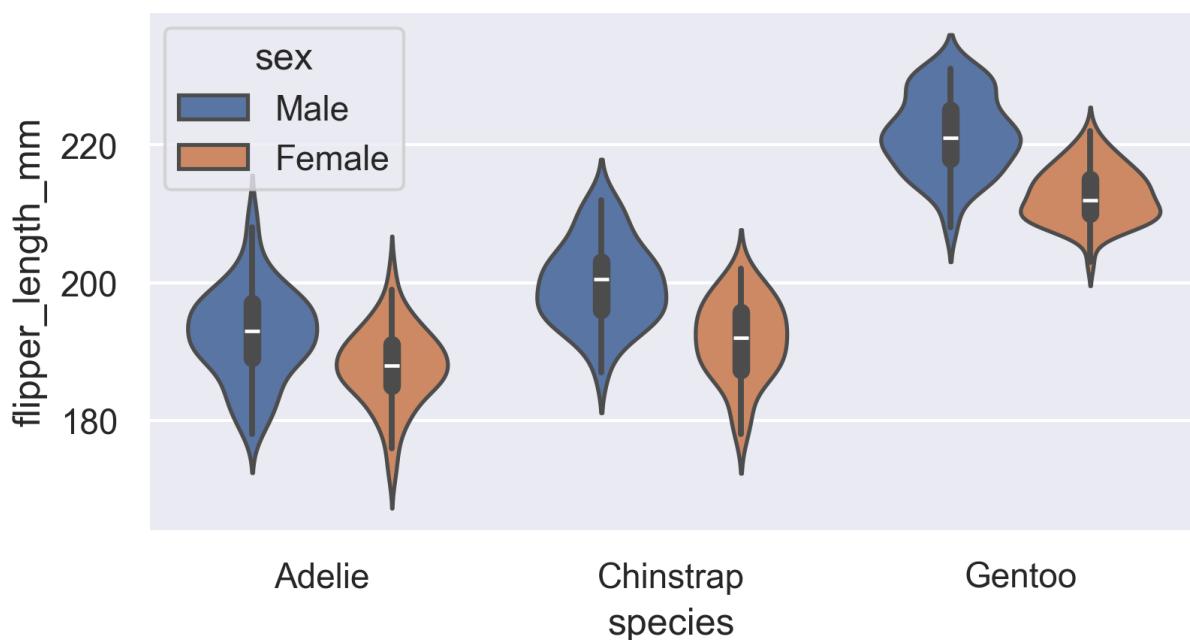
```
In [ ]: sns.violinplot(x="species", y="flipper_length_mm", data=data)
```

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

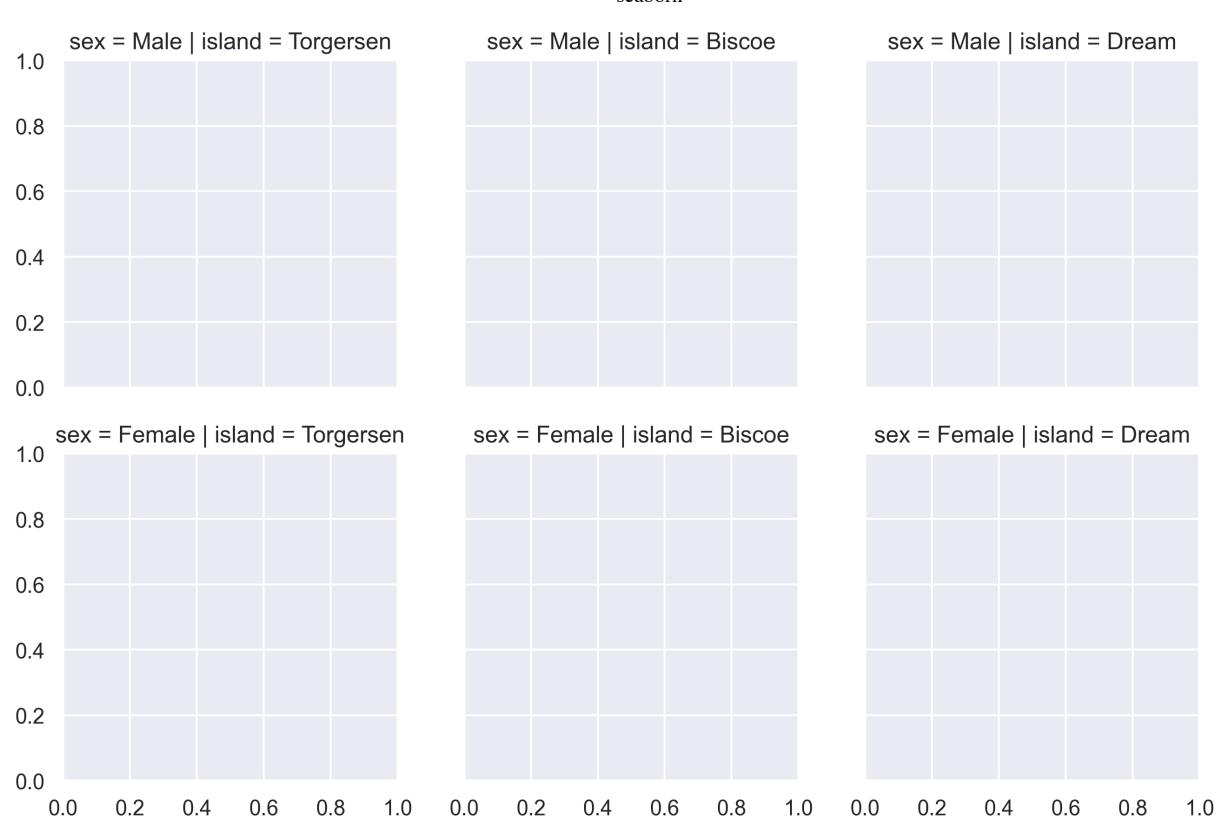


```
In [ ]: sns.violinplot(x="species", y="flipper_length_mm", data=data, hue="sex")
```

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



```
In [ ]: grid = sns.FacetGrid(data, col="island", row="sex", palette="Set2")
```



```
In [ ]: sns.FacetGrid(data,col="island",row="sex").map(sns.histplot,"flipper_length_mm")
```

Out[]: <seaborn.axisgrid.FacetGrid at 0x14f9f04f0>



```
In [ ]: sns.FacetGrid(data,col="island",row="sex").map(sns.distplot, "flipper_length_mm")
```

/Users/onurgumus/Desktop/Python ile Projeler/seaborn/.venv/lib/python3.10/site-packages/seaborn/axisgrid.py:854: 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>

```
func(*plot_args, **plot_kwargs)
/Users/onurgumus/Desktop/Python ile Projeler/seaborn/.venv/lib/python3.10/site-packages/seaborn/axisgrid.py:854: 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>

```
func(*plot_args, **plot_kwargs)
/Users/onurgumus/Desktop/Python ile Projeler/seaborn/.venv/lib/python3.10/site-packages/seaborn/axisgrid.py:854: 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>

```
func(*plot_args, **plot_kwargs)
/Users/onurgumus/Desktop/Python ile Projeler/seaborn/.venv/lib/python3.10/site-packages/seaborn/axisgrid.py:854: 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>

```
func(*plot_args, **plot_kwargs)
/Users/onurgumus/Desktop/Python ile Projeler/seaborn/.venv/lib/python3.10/site-packages/seaborn/axisgrid.py:854: 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>

```
func(*plot_args, **plot_kwargs)
/Users/onurgumus/Desktop/Python ile Projeler/seaborn/.venv/lib/python3.10/site-packages/seaborn/axisgrid.py:854: UserWarning:
```

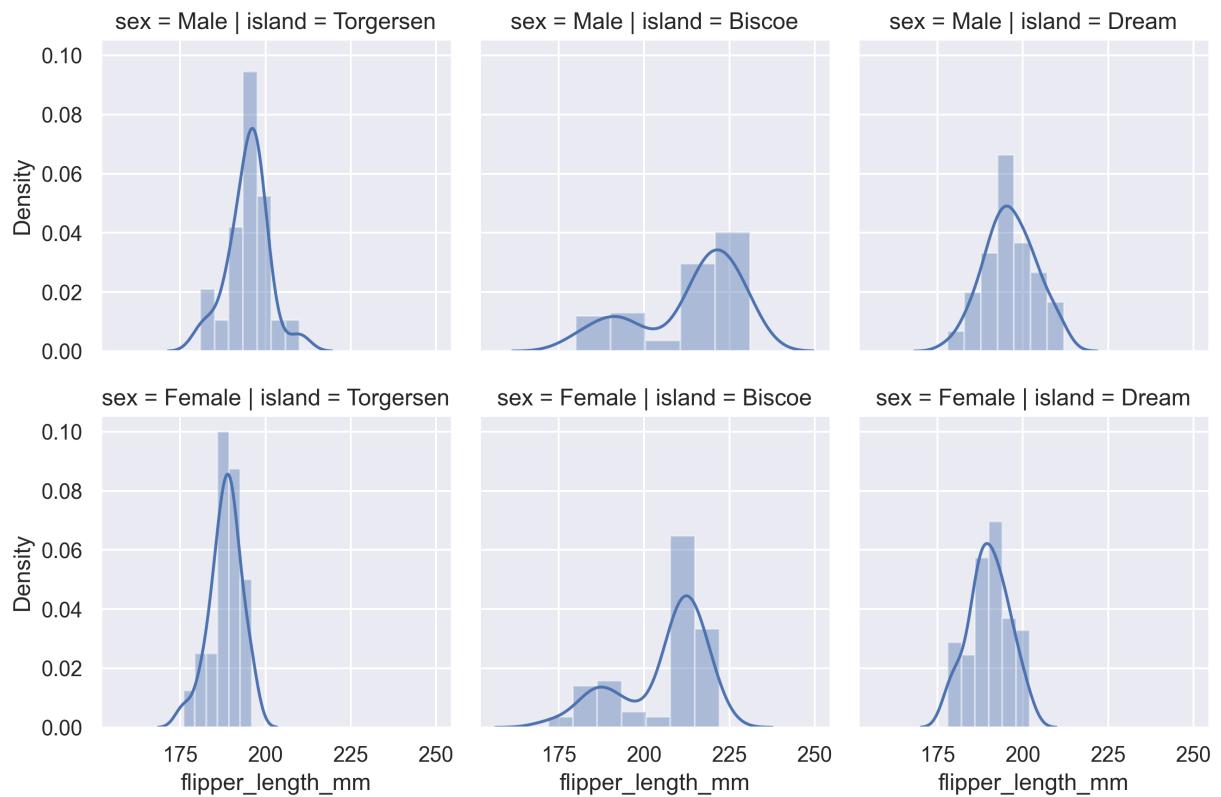
`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

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>

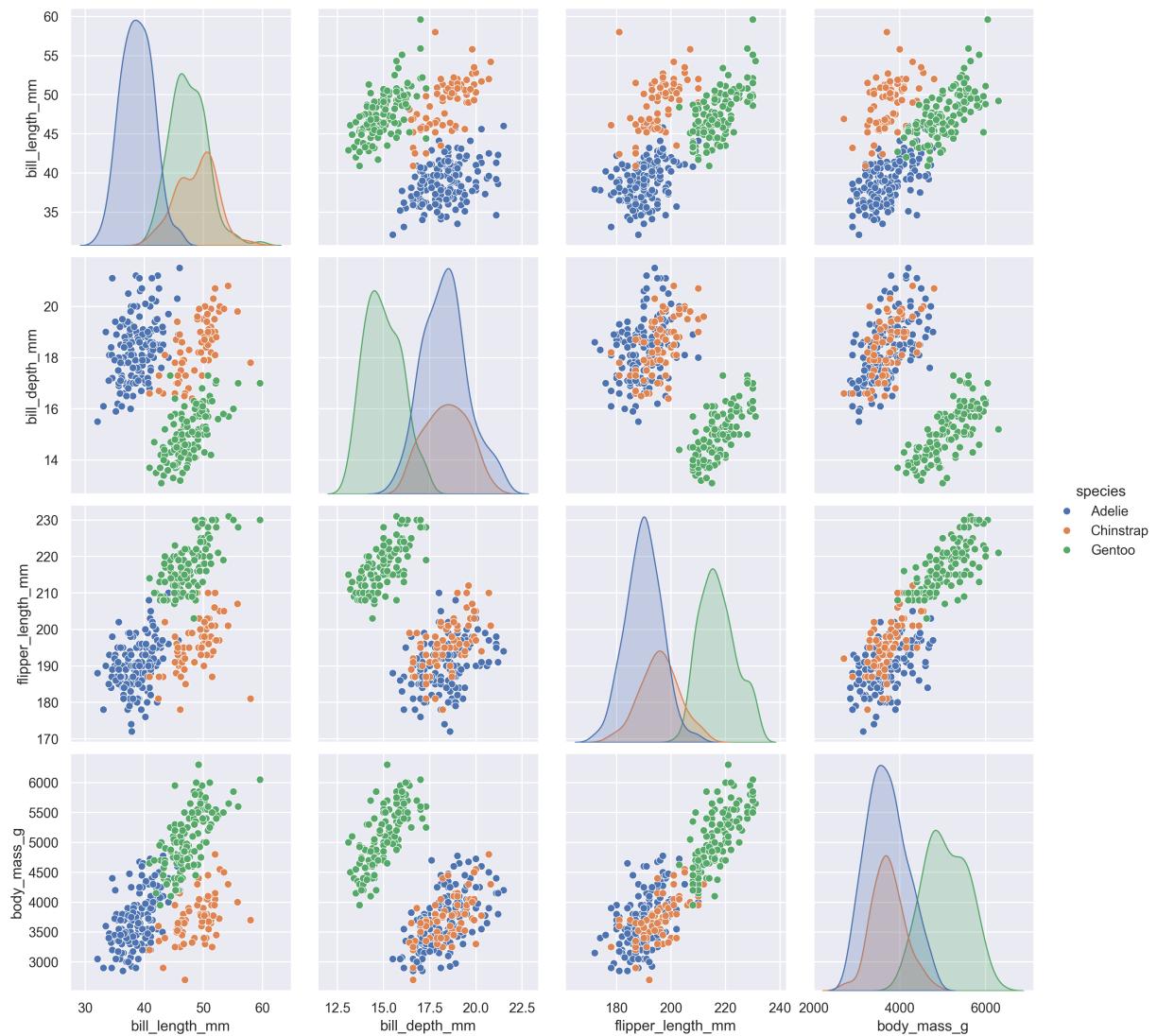
```
func(*plot_args, **plot_kwargs)
```

Out[]: <seaborn.axisgrid.FacetGrid at 0x14f4c62f0>



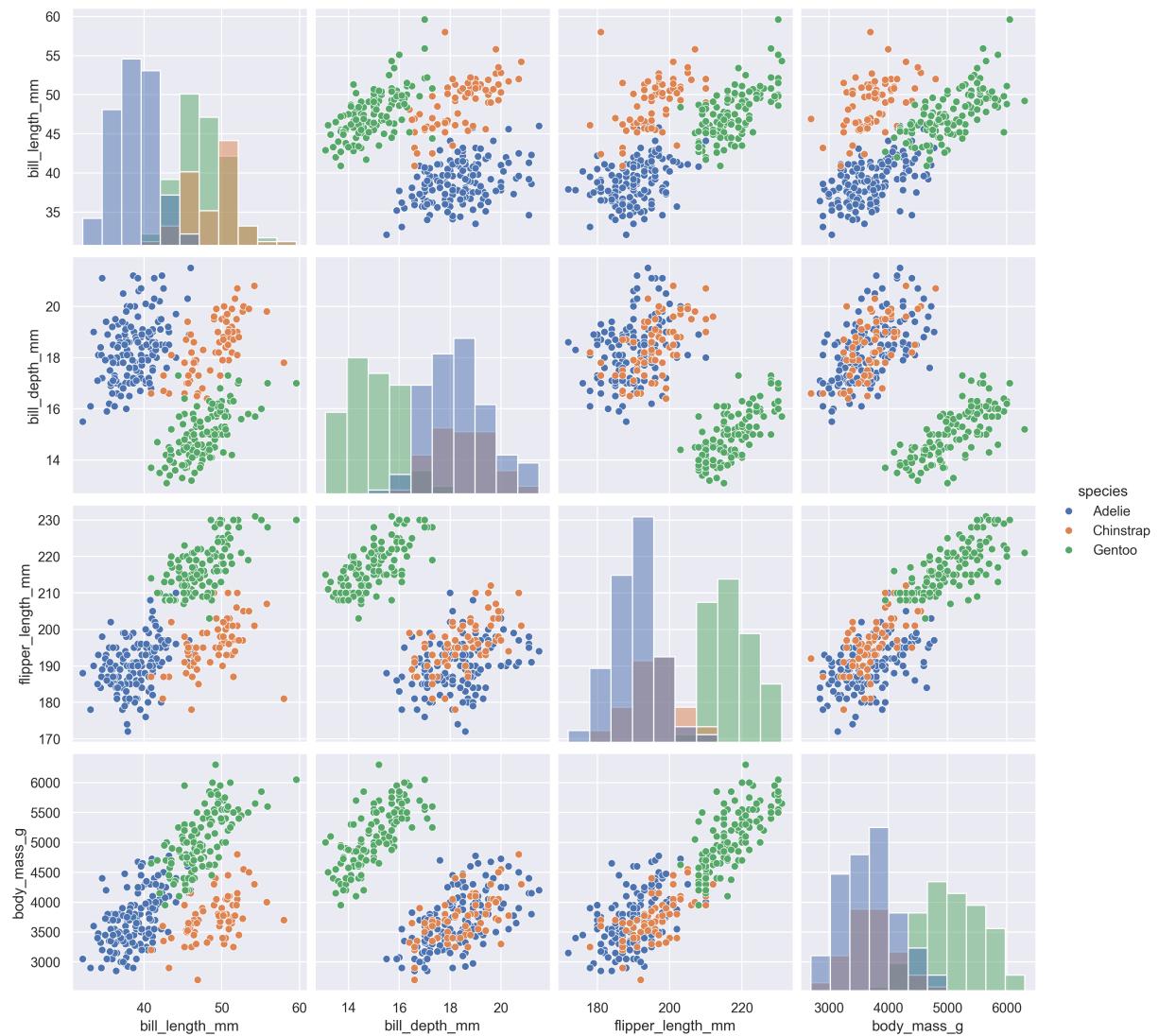
```
In [ ]: sns.pairplot(data,hue="species",height=3)
```

```
Out[ ]: <seaborn.axisgrid.PairGrid at 0x14fa59d20>
```



```
In [ ]: sns.pairplot(data,hue="species",height=3,diag_kind="hist")
```

```
Out[ ]: <seaborn.axisgrid.PairGrid at 0x28c3472e0>
```



In []:

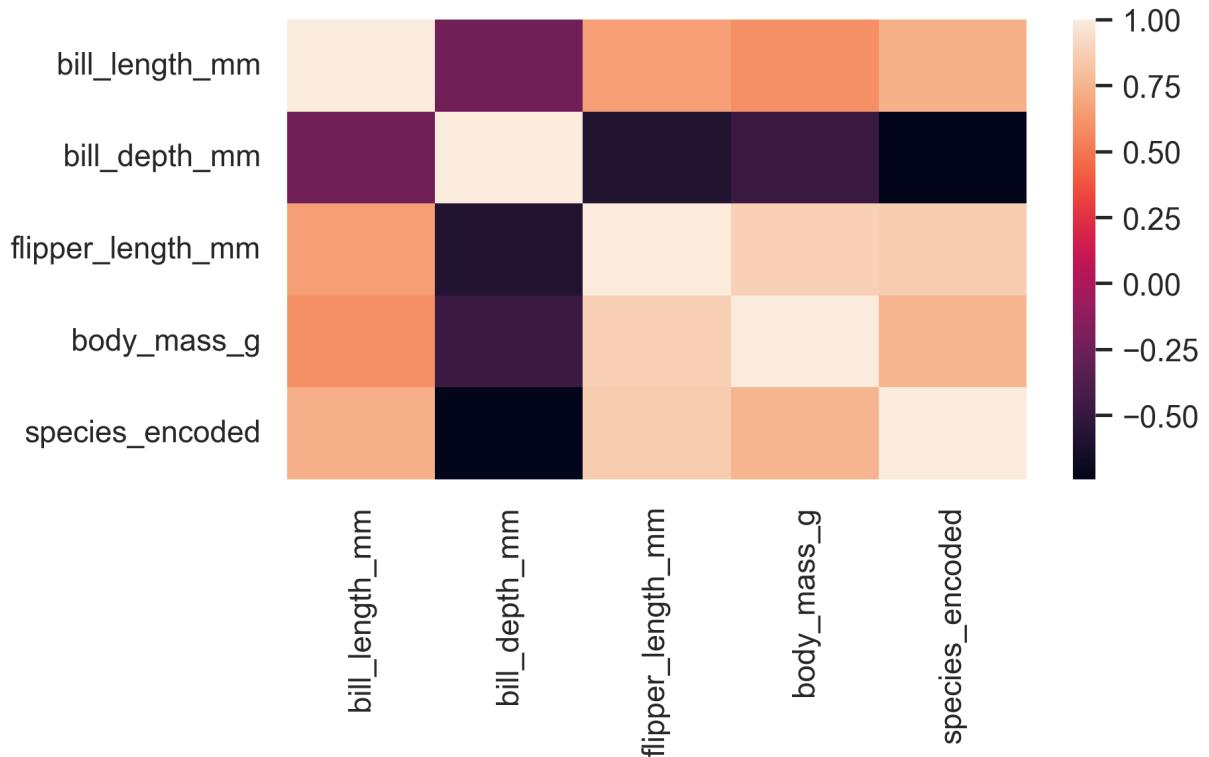
```
from sklearn.preprocessing import LabelEncoder

label_encoder = LabelEncoder()
data["species_encoded"] = label_encoder.fit_transform(data["species"])

# Select numeric columns only for correlation analysis
numeric_data = data.select_dtypes(include=['float64', 'int64'])

# Create heatmap of correlations using numeric data
sns.heatmap(numeric_data.corr())
```

Out[]: <Axes: >



In []:

```
# Create heatmap of correlations using numeric data
sns.heatmap(numeric_data.corr(), annot=True)
```

Out[]: <Axes: >

