LAB 5

TASK 1

1.Catplot

o Create a bar catplot to compare average body mass across species and sex.

o Title: "Average Body Mass by Species and Sex"

o X-axis: Species

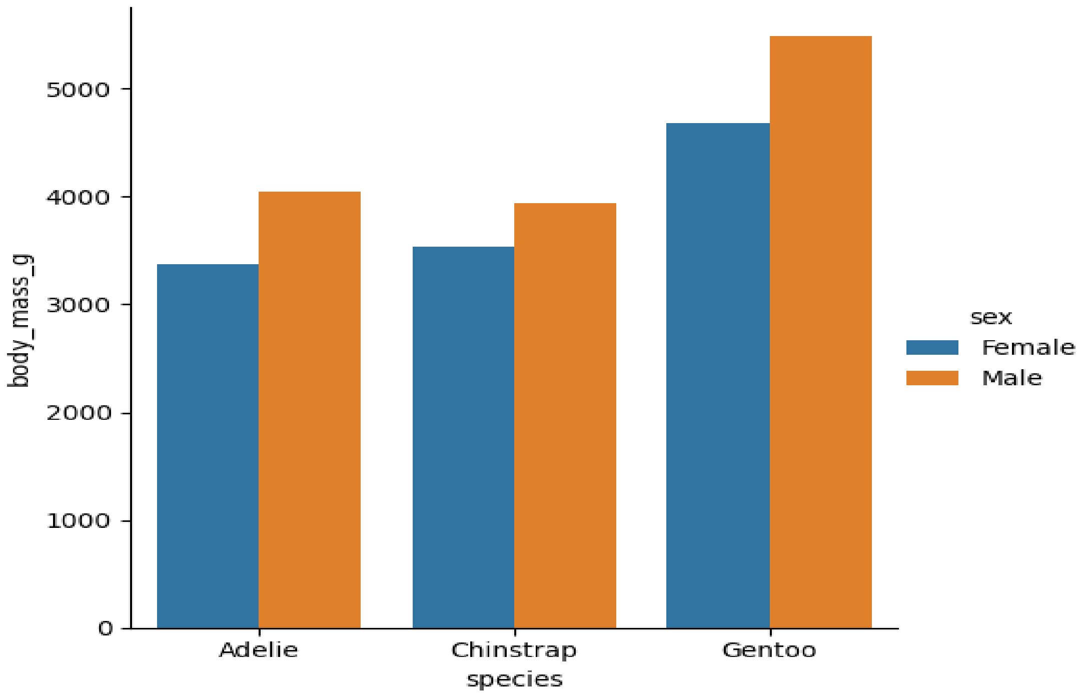
o Y-axis: Body Mass (g)

o Hue: Sex

INPUT

```
data_by_avg_body_mass =
data.groupby(["species","sex"])['body_mass_g'].mean().reset_index()

sns.catplot(data=data_by_avg_body_mass, x="species", y='body_mass_g', hue='sex',
kind='bar')

plt.savefig("Image1.png")
```

OUTPUT



TASK 2

2. Stripplot

o Visualize flipper length distribution by species and sex.

o Title: "Flipper Length Distribution by Species and Sex"

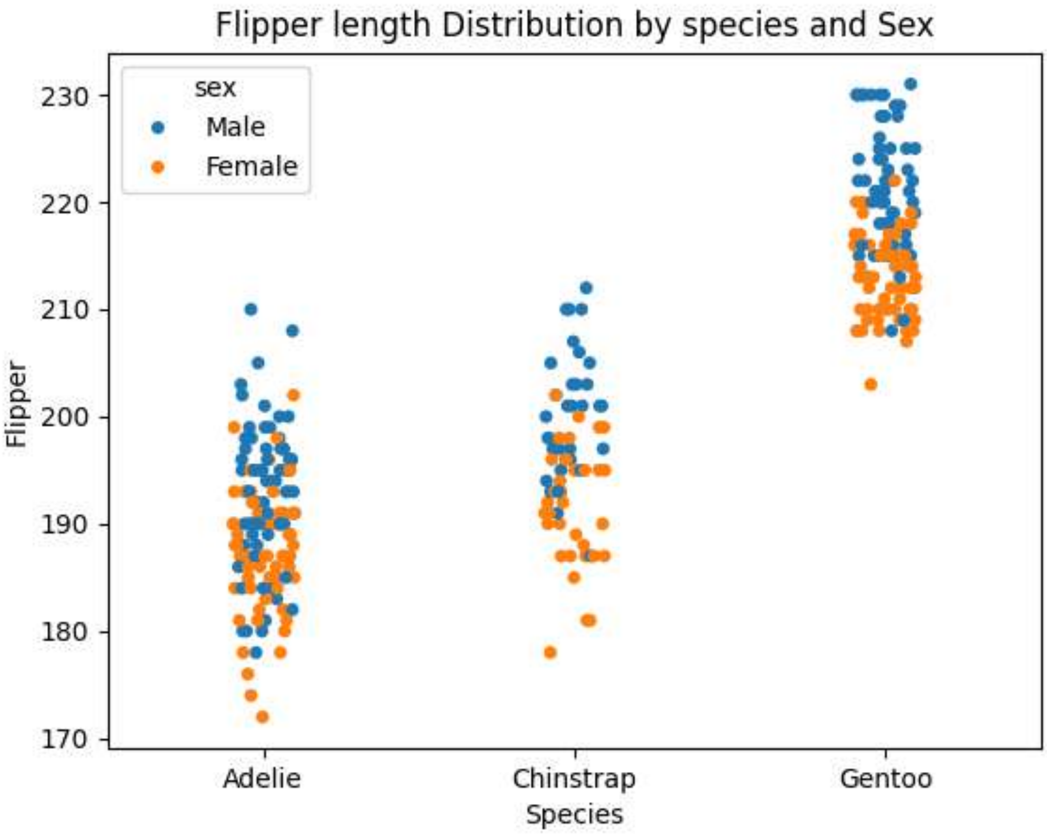
o X-axis: Species

o Y-axis: Flipper Length (mm)

INPUT

```
sns.stripplot(data=data, x="species", y='flipper_length_mm', hue='sex')
plt.title("Flipper length Distribution by species and Sex")
plt.xlabel("Species")
plt.ylabel("Flipper")
plt.savefig("Image2.png")
```

OUTPUT



TASK 3

3. Heatmap

- Display correlation matrix of numerical features (bill length, bill depth, flipper length, body , mass)

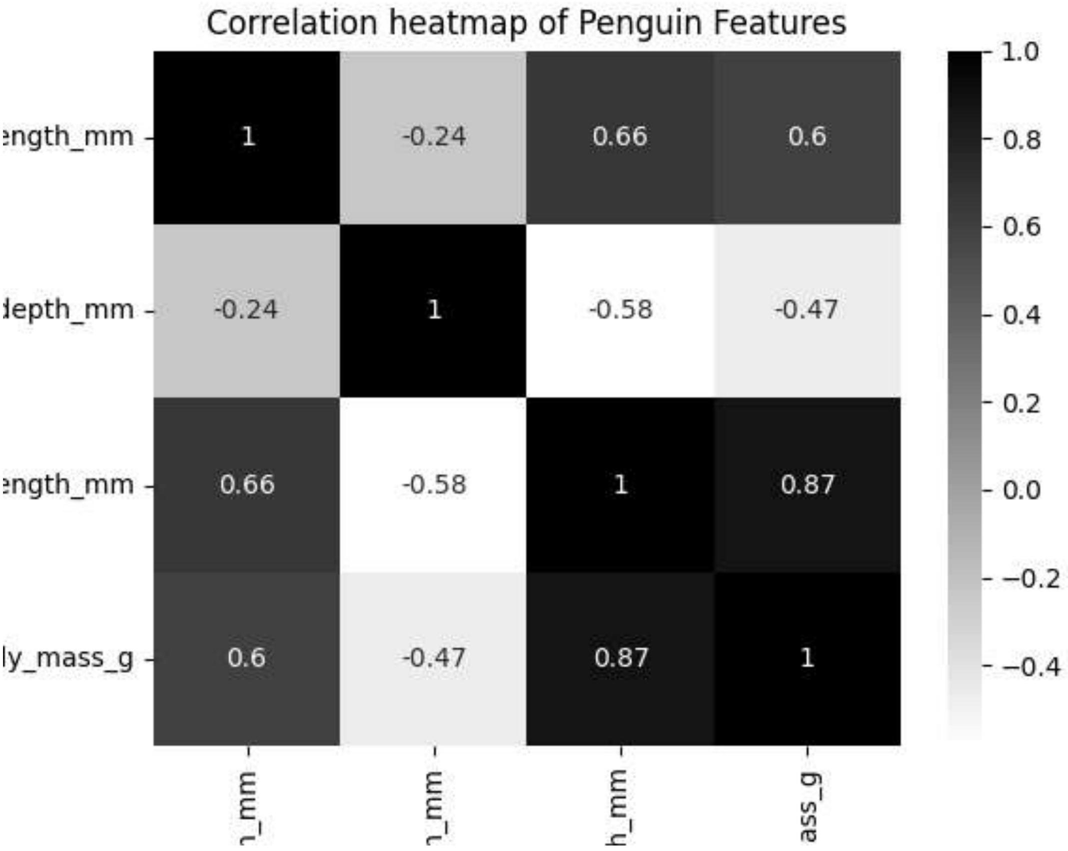
- Annotate values, use colorbar to represent correlation strength.

- Title: "Correlation Heatmap of Penguin Features

INPUT

```
new_data = data.drop(['species', 'island', 'sex'], axis = 1)
data_corr = new_data.corr()
sns.heatmap(data=data_corr, annot=True, cmap='binary')
plt.title("Correlation heatmap of Penguin Features")
plt.savefig("Image3.png")
```

OUTPUT



TASK 4

4. Clustermap

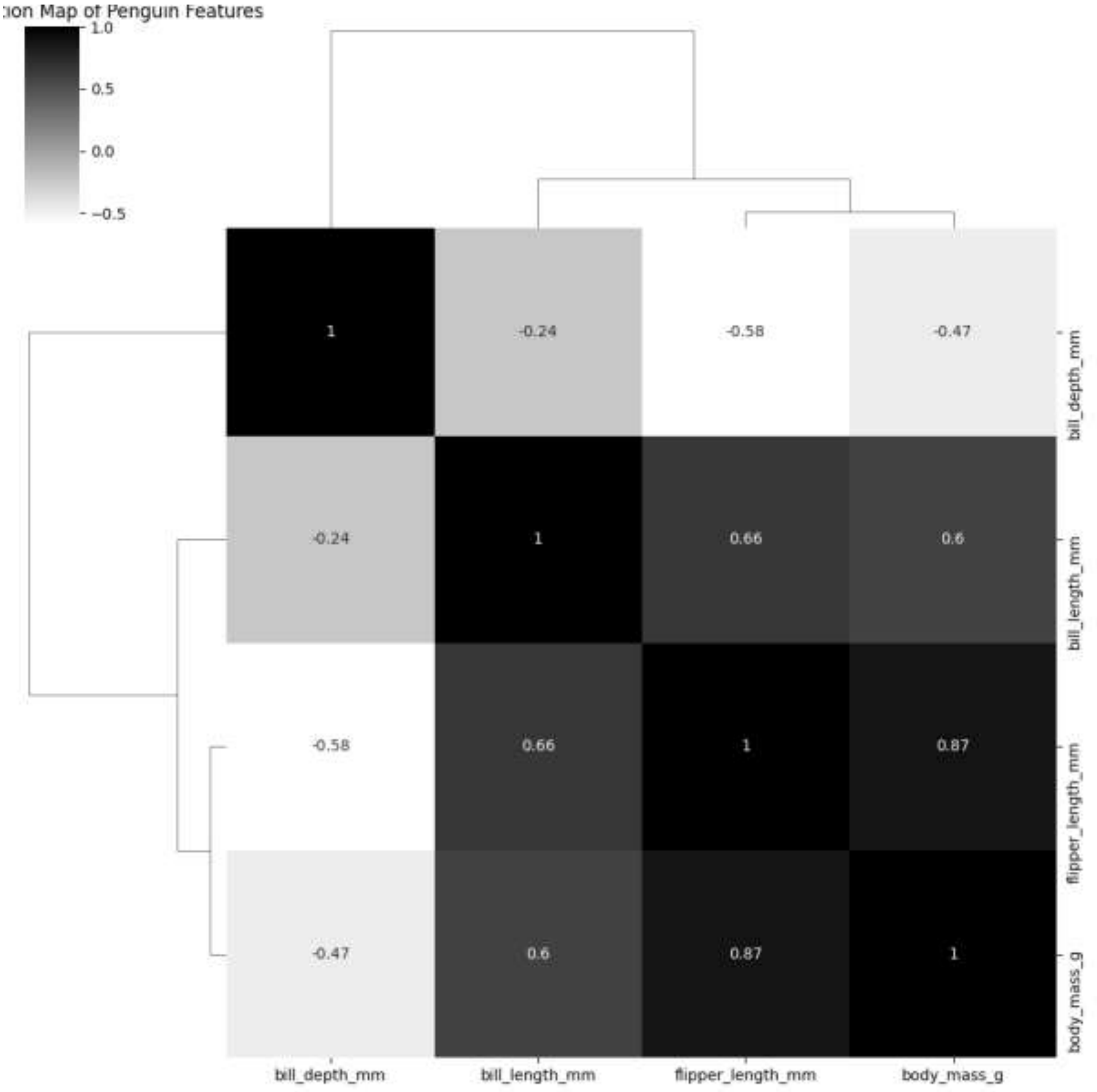
o Create a clustered heatmap to group features based on similarity.

o Title: "Clustered Correlation Map of Penguin Features”

INPUT

```
sns.clustermap(data=data_corr, annot=True, cmap='binary')
plt.title("Clustered Correlation Map of Penguin Features")
plt.savefig("Image4.png")
```

OUTPUT



TASK 5

5. Box Plot

o Compare total bill distribution across days.

o Title: "Total Bill Distribution by Day"

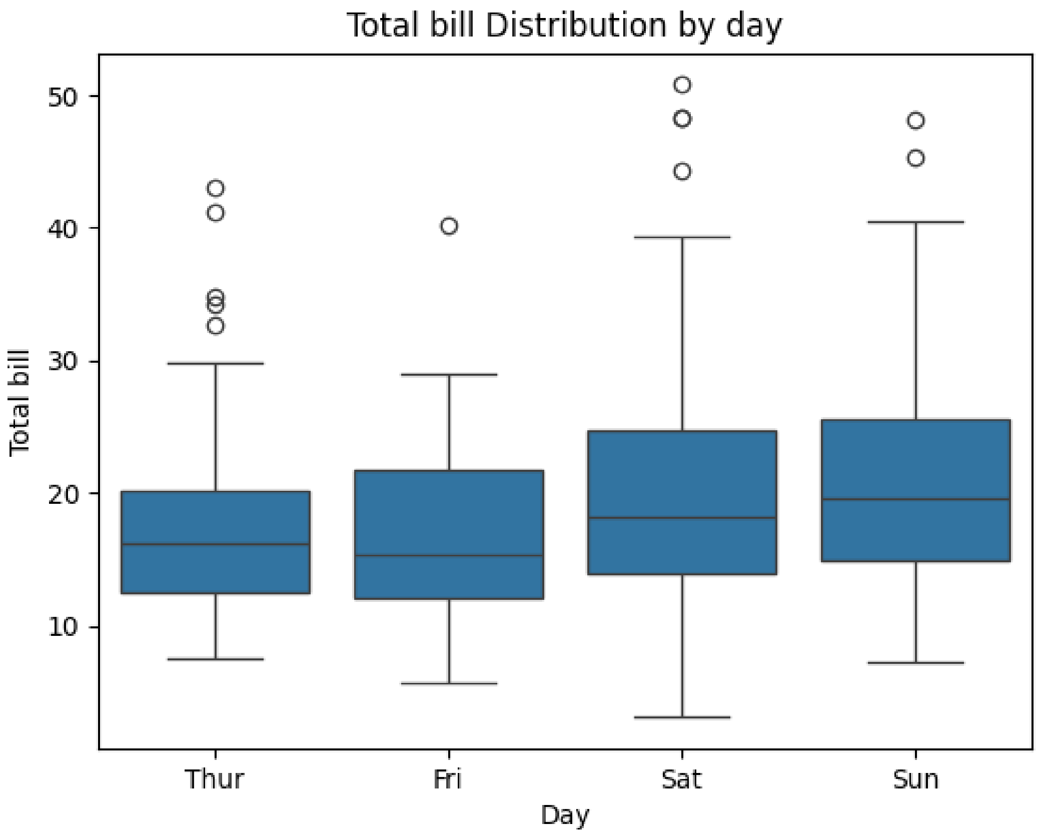
o X-axis: Day

o Y-axis: Total Bill

INPUT

```
sns.boxplot(data=data2, x='day', y='total_bill')

plt.title("Total bill Distribution by day")

plt.xlabel("Day")

plt.ylabel("Total bill")

plt.savefig("Image5.png")
```

OUTPUT



TASK 6

6. PViolin Plot

o Showtip distribution by smoker status.

o Title: "Tip Distribution by Smoker Status"

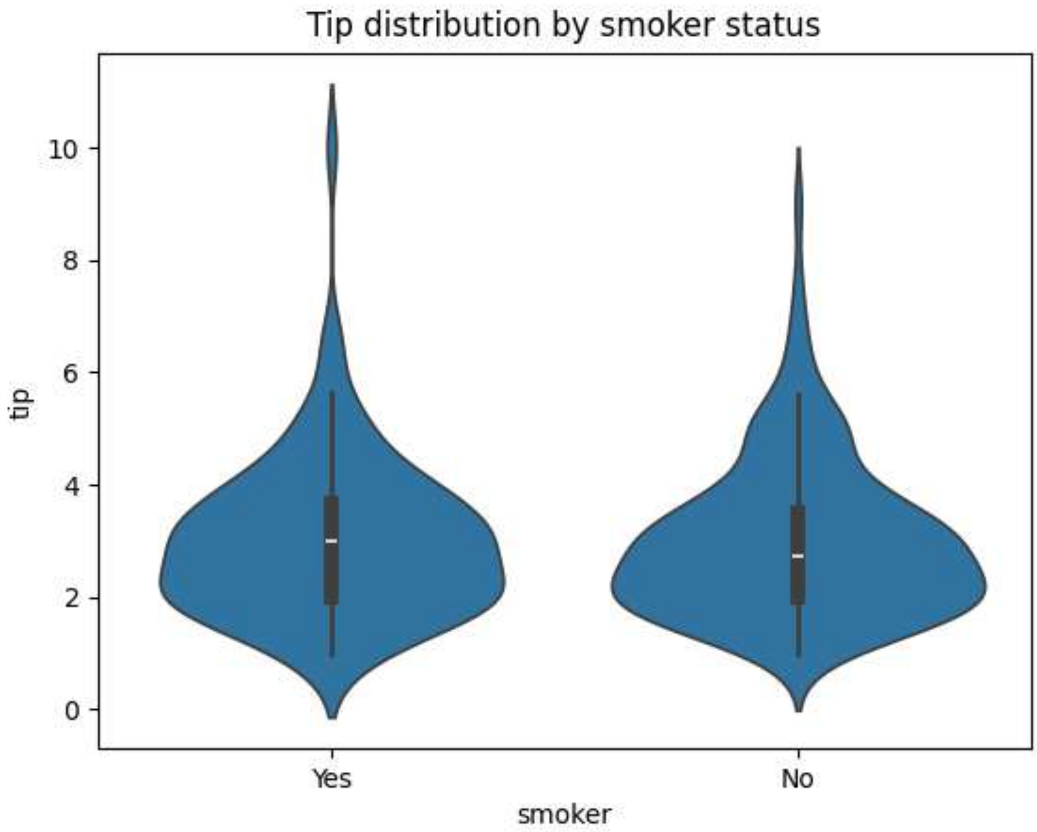
o X-axis: Smoker

o Y-axis: Tip

INPUT

```
sns.violinplot(data=data2,x='smoker', y='tip')
plt.title("Tip distribution by smoker status")
plt.savefig("Image6.png")
plt.show()
```

OUTPUT



TASK 7

7. KDEPlot

o Plot density of total bill amounts split by gender.

o Title: "Density of Total Bill by Gender"

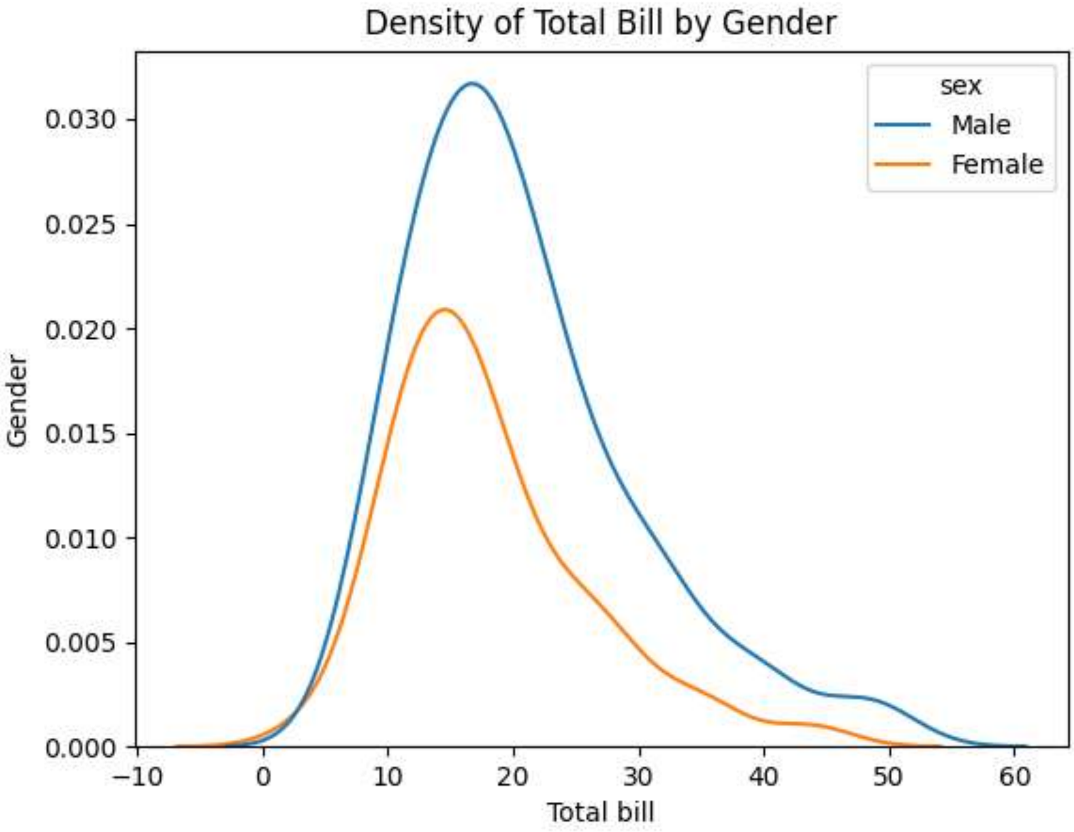
o X-axis: Total Bill

o Hue: Gender

INPUT

```
sns.kdeplot(data=data2, x="total_bill", hue='sex')

plt.title("Density of Total Bill by Gender")

plt.xlabel("Total bill")

plt.ylabel("Gender")

plt.savefig("Image7.png")
```

OUTPUT



TASK 8

8. Joint Plot

o Explore relationship between total bill and tip.

o Title: "Total Bill vs Tip Relationship"

o X-axis: Total Bill

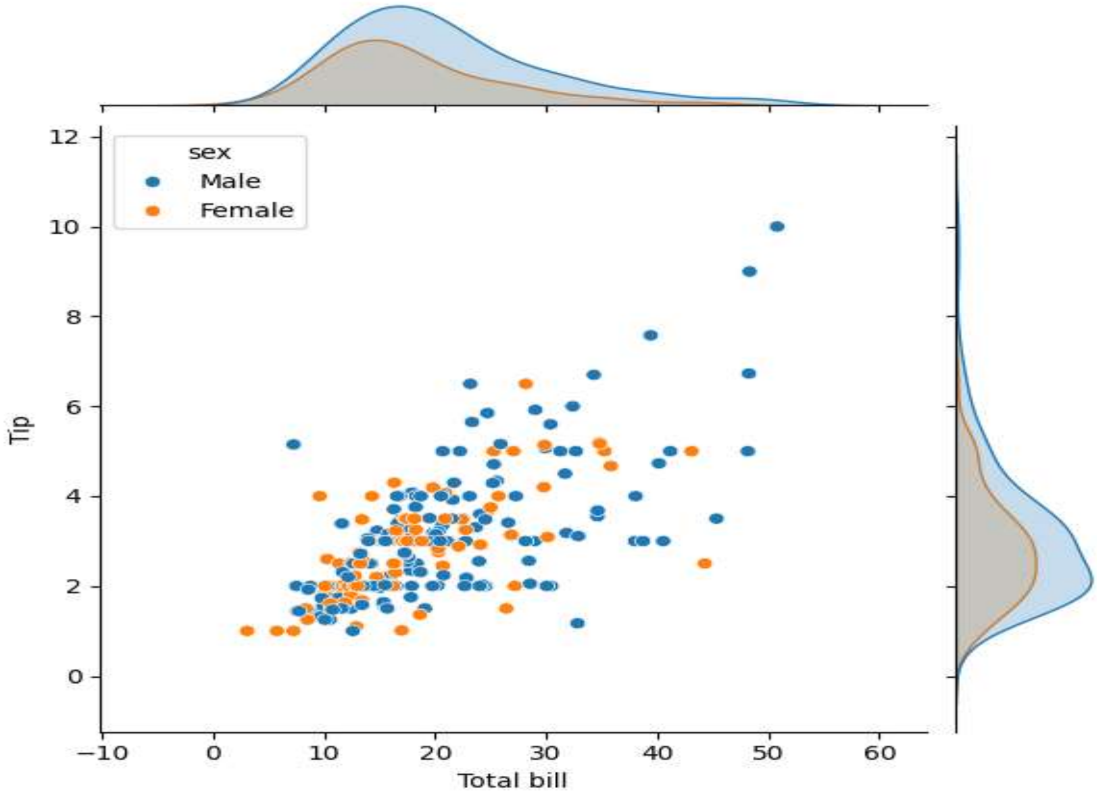
o Y-axis: Tip

o Hue: Gender

INPUT

```
sns.jointplot(data=data2, x="total_bill",y="tip",kind="scatter", hue='sex')
plt.suptitle("Total Bill vs Tip Relationship", y=1.05)
plt.xlabel("Total bill")
plt.ylabel("Tip")
plt.savefig("Image8.png")
```

OUTPUT



TASK 9

9. Pairplot

o Create pairwise relationships among all numeric features.

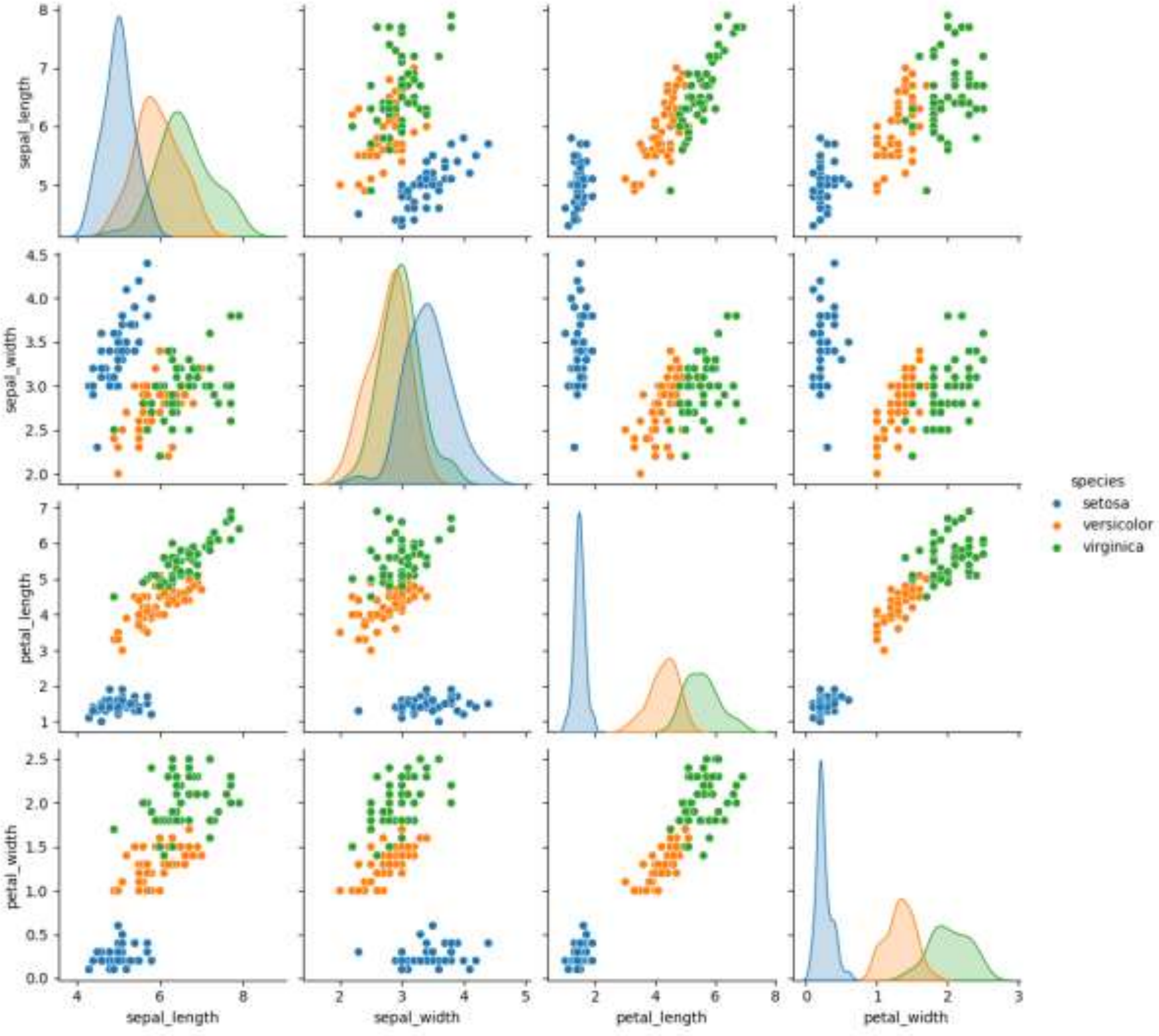
o Title: "Pairwise Relationships in Iris Dataset"

o Hue: Species

INPUT

```
sns.pairplot(data=data3, hue='species')

plt.suptitle("Pairwise Relationship in Iris Dataset", y=1.02)

plt.savefig("Image9.png")
```

OUTPUT



TASK 10

10. SwarmPlot

o Visualize petal length distribution across species.

o Title: "Petal Length Distribution by Species"

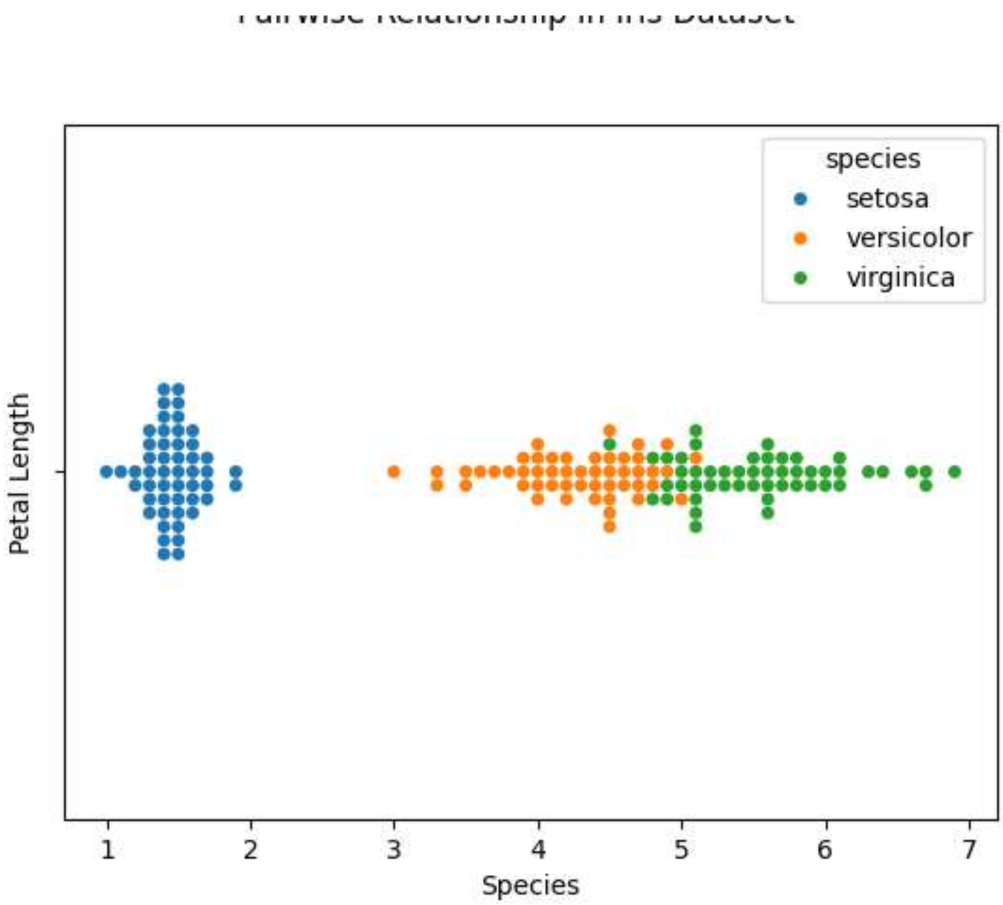
o X-axis: Species

o Y-axis: Petal Length

INPUT

```
sns.swarmplot(data=data3, hue='species', x='petal_length')
plt.suptitle("Pairwise Relationship in Iris Dataset", y=1.02)
plt.xlabel("Species")
plt.ylabel("Petal Length")
```

OUTPUT



TASK 11

10. Bubble Chart (Scatterplot with size)

o Sepal length vs sepal width

o bubble size = petal length

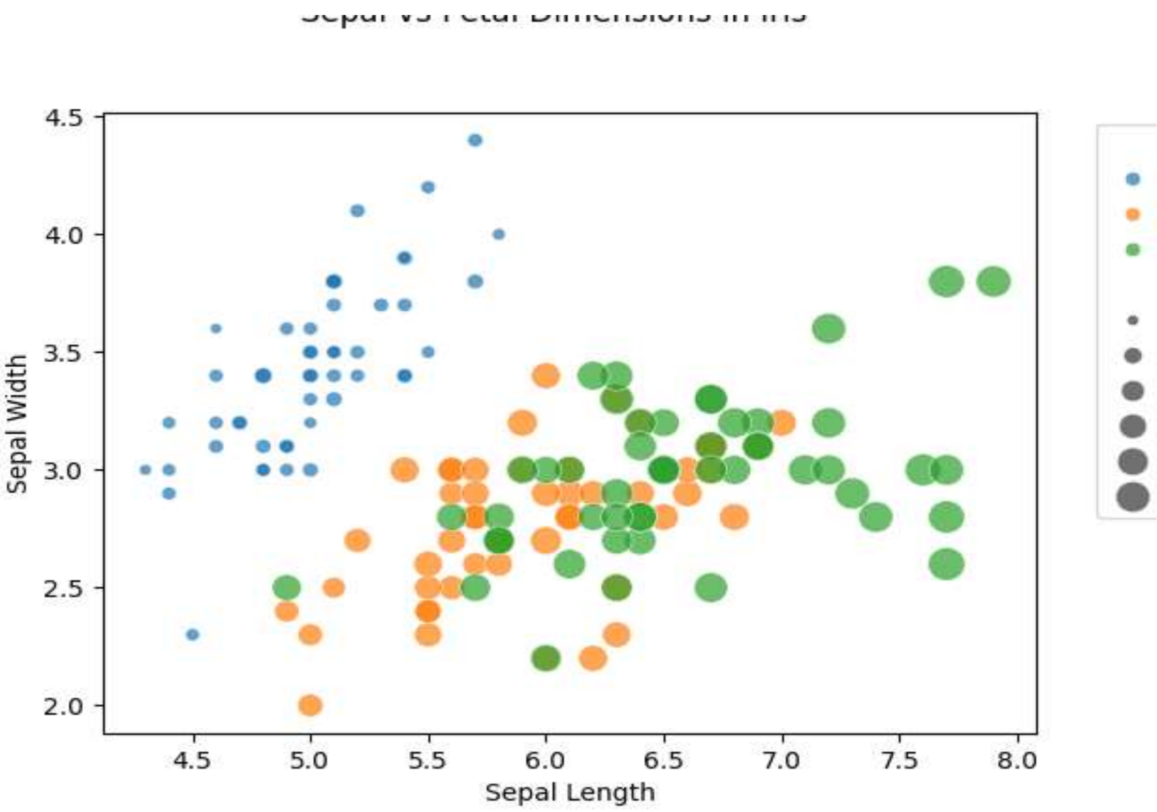
o hue = species.

o Title: "Sepal vs Petal Dimensions in Iris"

INPUT

```
sns.scatterplot(data=data3,x='sepal_length',y='sepal_width',size='petal_length',hue='species',sizes=(20, 200),  alpha=0.7)

plt.suptitle("Sepal vs Petal Dimensions in Iris", y=1.02)

plt.xlabel("Sepal Length")

plt.ylabel("Sepal Width")

plt.legend(bbox_to_anchor=(1.05, 1), loc='upper left')

plt.show()
```

OUTPUT


Sepal vs Petal Dimensions in Iris
Sepal Width
Sepal Length