LAB 4

TASK 1

1. Area Chart

o Create an area chart showing the average petal length for each species.

o Title: "Average Petal Length by Species"

o X-axis: Species

o Y-axis: Average Petal Length (cm)

o Add legend and gridlines (major and minor).

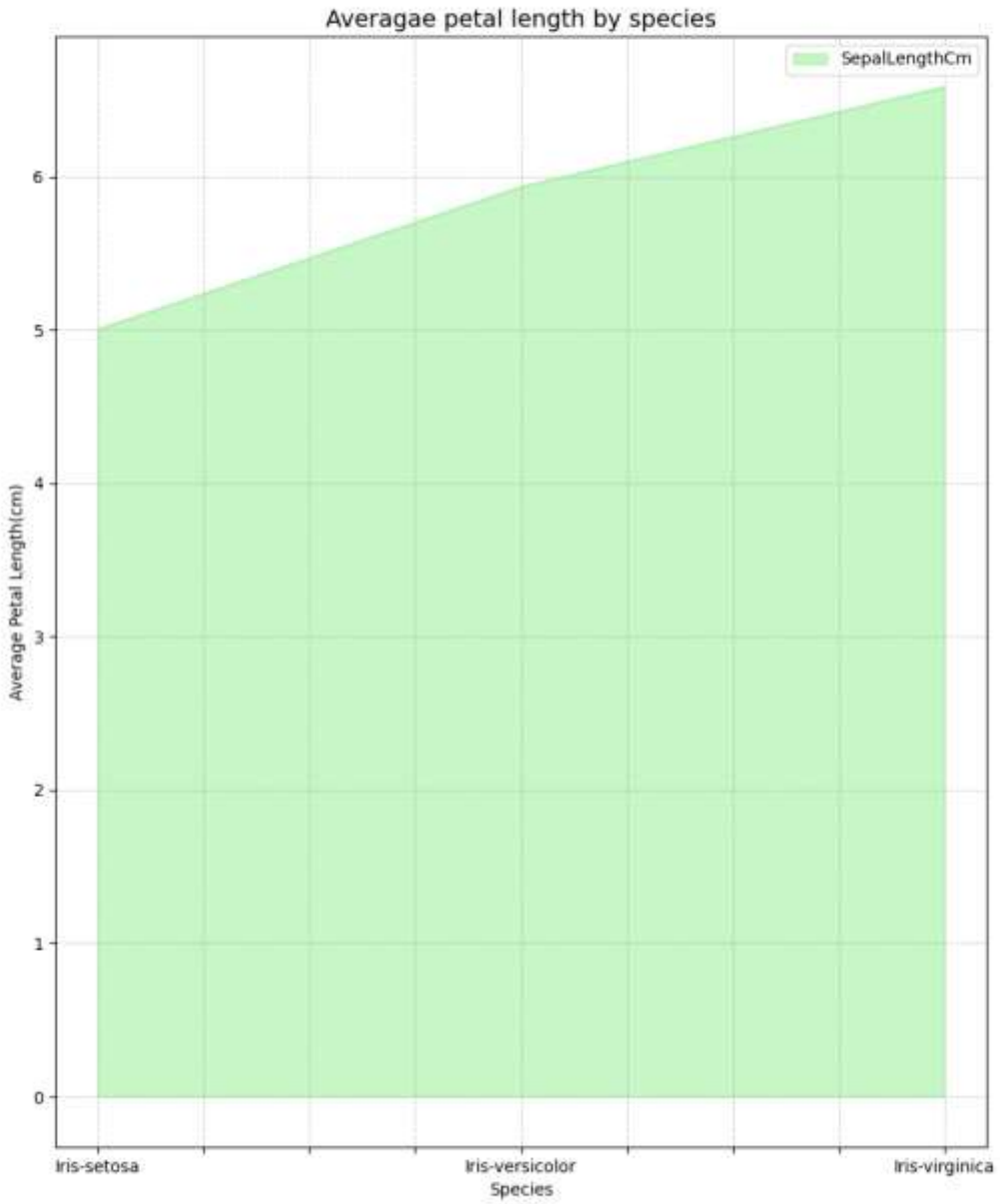
o Set the legend position to the bottom-right corner.

o Add a legend title "Species" and set the legend font size to 12.

INPUT

```
import matplotlib.pyplot as plt


avg_petal_length_data = data.groupby("Species")['SepalLengthCm'].mean()


plt.figure(figsize=(10, 12))
avg_petal_length_data.plot(kind='area', alpha=0.5, color='lightgreen')
plt.title("Averagae petal length by species", fontsize=14)
plt.xlabel("Species")
plt.ylabel("Average Petal Length(cm)")
plt.legend()


plt.grid(True, linestyle='--', alpha=0.6)
plt.savefig("Image1.png")
plt.show()
plt.plot()
```

OUTPUT



TASK 2

2. Bubble Chart

o X-axis: Sepal Length

o Y-axis: Sepal Width

o Bubble size: Petal Length

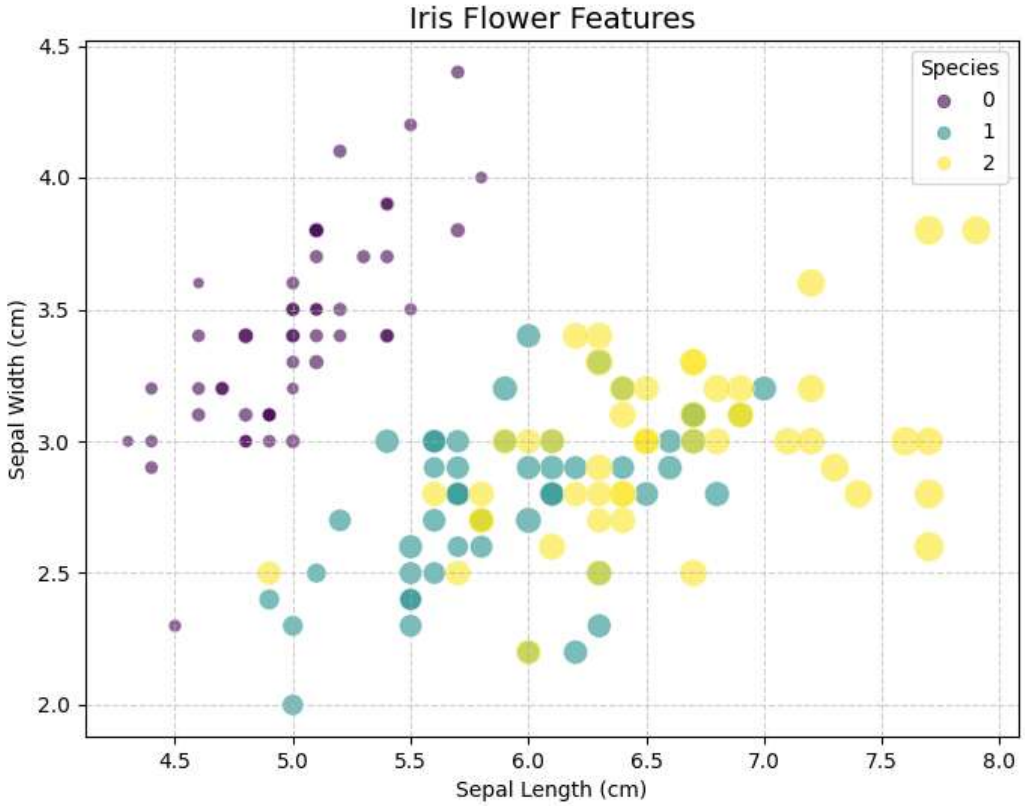
o Bubble color: Species

o Title: "Iris Flower Features"

INPUT

```
import matplotlib.pyplot as plt


x = data['SepalLengthCm']

y = data['SepalWidthCm']

sizes = data['PetalLengthCm'] * 30   # scale bubble size for visibility

colors = data['Species'].astype('category').cat.codes  # map species to numeric codes


plt.figure(figsize=(8, 6))

scatter = plt.scatter( x, y,  s=sizes,  c=colors, cmap='viridis', alpha=0.6,  edgecolors='w',
linewidth=0.5)

plt.title("Iris Flower Features", fontsize=14)

plt.xlabel("Sepal Length (cm)")

plt.ylabel("Sepal Width (cm)")


# Add legend for species

legend_labels = data['Species'].unique()

plt.legend(*scatter.legend_elements() , title="Species")


plt.grid(True, linestyle='--', alpha=0.6)

plt.savefig("Image2.png")

plt.show()
```

OUTPUT



TASK 3

3. Heat Map

o Create a correlation heatmap for all numeric features.

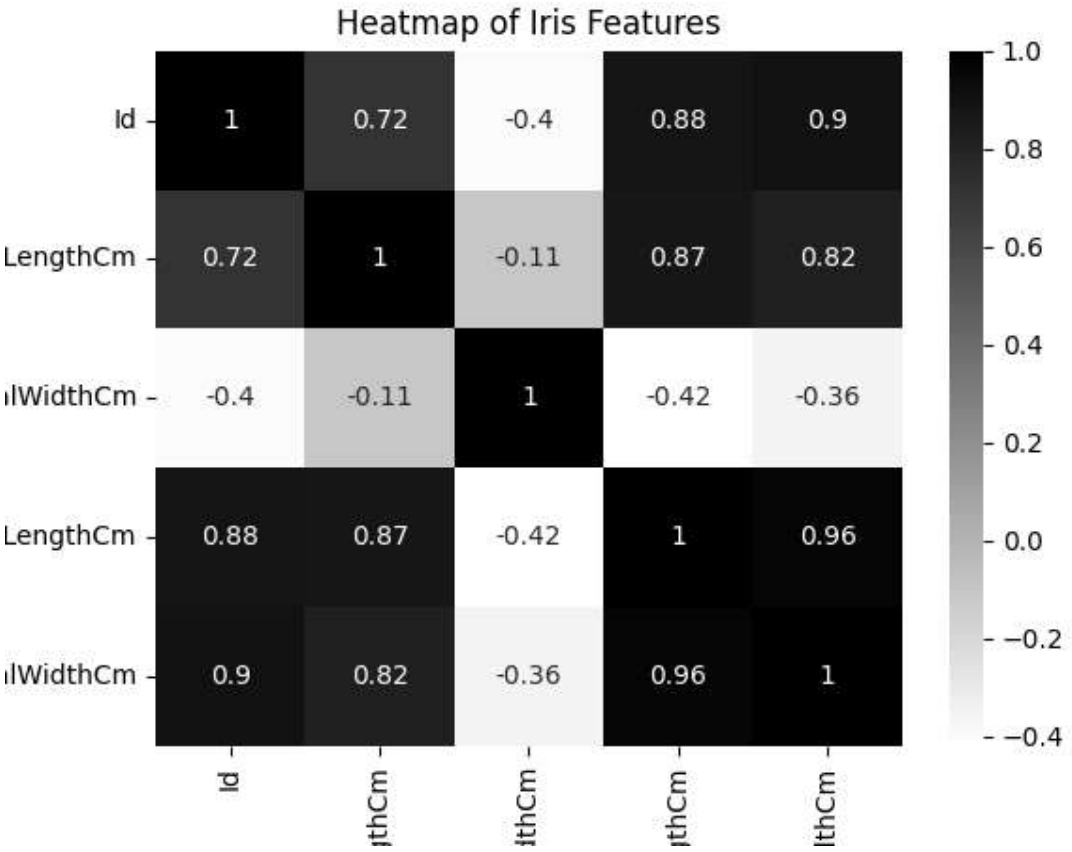
o Title: "Heatmap of Iris features"

INPUT

```
import seaborn as sns


new_data = data.drop("Species", axis=1)
corr_data = new_data.corr()


sns.heatmap(corr_data, cmap='binary', annot=True)
plt.title("Heatmap of Iris Features")
plt.savefig("Image3.png")
```

plt.show()

OUTPUT



TASK 4

4. Box Plot

o Create box plots for petal length by species.

o Title: "Petal Length Distribution by Species"

o Create box plots showing the distribution of petal length for each species.

o Clearly represent the median, first quartile (Q1), third quartile (Q3), and the interquartile range (IQR).

o Display outliers using distinct marker symbols for easy identification.

o Use different colors for each species to ensure clear visual distinction.

INPUT

Box plot of petal length by species

```
sns.boxplot(data=data,x="Species", y="PetalLengthCm",  palette="Set2", notch=True)
plt.title("Petal Length Distribution by Species")
plt.xlabel("Species")
plt.ylabel("Petal Length")
plt.savefig("Image4.png")
plt.show()
```

OUTPUT

TASK 5

5. Violin Plot

o Create violin plots for petal width across species.

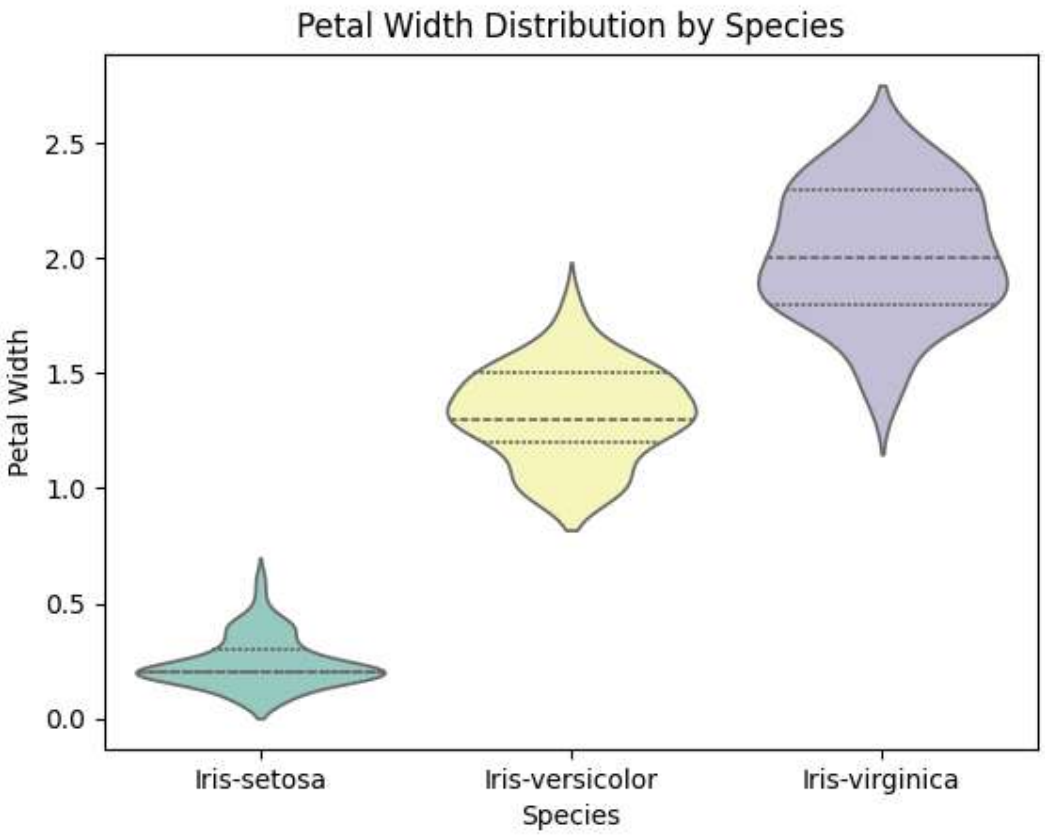
o Title: "Petal Width Distribution by Species"

INPUT

```
sns.violinplot(data=data,x="Species",y="PetalWidthCm",palette="Set3",  inner="quartile")
```

```
plt.title("Petal Width Distribution by Species")
plt.xlabel("Species")
plt.ylabel("Petal Width")
plt.savefig("Image5.png")
plt.show()
```

OUTPUT



TASK 6

6. Pair Plot

o Create pairwise scatter plots between all numeric features.

o Create pairwise scatter plots between all numeric features of the Iris dataset: Sepal Length, Sepal Width, Petal Length and Petal Width.

o Display feature-to-feature relationships across all possible combinations.

o Use a consistent axis scale for all subplots to ensure fair visual comparison. Apply

uniform marker style and size for visual consistency.

o Use different colors for different species to enable class-wise visual separation.

o Maintain color consistency for each species across all subplots.

INPUT

```
sns.pairplot(data=data,vars=["SepalLengthCm", "SepalWidthCm", "PetalLengthCm", "PetalWidthCm"],hue="Species",diag_kind="kde", markers="o", plot_kws={"s": 40, "alpha": 0.7} )

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

plt.savefig("Image6.png")

plt.show()
```

OUTPUT