**Task: Exploring Various Types of Visual Representations in Python**

The goal of this task is to help students learn about different types of visual representations available in Python. Each type of visualization has its specific purpose and is suited for particular types of data relationships. Students will practice creating these visualizations using the Iris dataset and understand the scenarios where each type is most effective.

**1. Histogram**

Used to show the distribution of a single variable. It divides the variable into bins and shows the frequency of values in each bin.

Example:

python

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

# Load dataset

df = sns.load\_dataset('iris')

# Histogram

plt.figure(figsize=(10, 5))

sns.histplot(df['sepal\_length'], kde=True)

plt.title('Histogram of Sepal Length')

plt.xlabel('Sepal Length')

plt.ylabel('Frequency')

plt.show()

**2. Bar Chart**

Used to compare different categories or groups. The height of the bars represents the value for each category.

Example:

python

# Load dataset

df = sns.load\_dataset('iris')

# Bar Chart

plt.figure(figsize=(10, 5))

sns.countplot(x='species', data=df)

plt.title('Count of Each Species')

plt.xlabel('Species')

plt.ylabel('Count')

plt.show()

**3. Line Chart**

Used to show trends over a period of time or a sequence. It connects data points with a line.

Example:

python

# Load dataset

df = sns.load\_dataset('iris')

# Simulating data over time

time\_data = pd.DataFrame({

'day': range(1, 11),

'sepal\_length': df['sepal\_length'][:10]

})

**Line Chart**

plt.figure(figsize=(10, 5))

sns.lineplot(x='day', y='sepal\_length', data=time\_data)

plt.title('Line Chart of Sepal Length Over Time')

plt.xlabel('Day')

plt.ylabel('Sepal Length')

plt.show()

**4. Bar and Line Chart**

Combines bar and line charts to show the relationship between two different types of data.

**Example:**

python

# Load dataset

df = sns.load\_dataset('iris')

fig, ax1 = plt.subplots(figsize=(10, 5))

# Bar chart for count of species

sns.countplot(x='species', data=df, ax=ax1, alpha=0.6, color='b')

ax1.set\_ylabel('Count', color='b')

# Line chart for average sepal length of species

ax2 = ax1.twinx()

sns.pointplot(x='species', y='sepal\_length', data=df, ax=ax2, color='r', markers='o', linestyles='-')

ax2.set\_ylabel('Average Sepal Length', color='r')

plt.title('Bar and Line Chart of Species Count and Sepal Length')

plt.show()

**5. Pie Chart**

Used to show the proportions of a whole. Each slice represents a category's proportion to the total.

**Example:**

python

# Load dataset

df = sns.load\_dataset('iris')

# Pie Chart

species\_count = df['species'].value\_counts()

plt.figure(figsize=(10, 5))

plt.pie(species\_count, labels=species\_count.index, autopct='%1.1f%%', startangle=140)

plt.title('Pie Chart of Species Distribution')

plt.show()

**6. Heatmap**

Used to show the relationship between two variables using color to represent the intensity of the relationship.

Example:

python

# Load dataset

df = sns.load\_dataset('iris')

# Heatmap

plt.figure(figsize=(10, 5))

sns.heatmap(df.corr(), annot=True, cmap='coolwarm', linewidths=0.5)

plt.title('Correlation Heatmap')

plt.show()

**7. Pair Plot**

Used to show the pairwise relationships between variables in a dataset.

Example:

python

# Load dataset

df = sns.load\_dataset('iris')

# Pair Plot

sns.pairplot(df)

plt.title('Pair Plot of Iris Dataset')

plt.show()

**8. Box Plot**

Used to show the distribution of a variable and identify outliers. It displays the median, quartiles, and outliers.

Example:

python

# Load dataset

df = sns.load\_dataset('iris')

# Box Plot

plt.figure(figsize=(10, 5))

sns.boxplot(x='species', y='sepal\_length', data=df)

plt.title('Box Plot of Sepal Length by Species')

plt.xlabel('Species')

plt.ylabel('Sepal Length')

plt.show()

**Task for Students**

1. Load the Iris dataset into a Pandas DataFrame.

2. Create each type of visualization (histogram, bar chart, line chart, bar and line chart, pie chart, heatmap, pair plot, box plot).

3. Write a brief explanation of the purpose of each visualization.

4. Submit a Jupyter notebook file (.ipynb) containing your code, visualizations, and explanations.