

BÀI TẬP THỰC HÀNH PTDL1: SEABORN

Import thư viện

```
In [1]: import pandas as pd  
import seaborn as sns  
import matplotlib.pyplot as plt  
import numpy as np
```

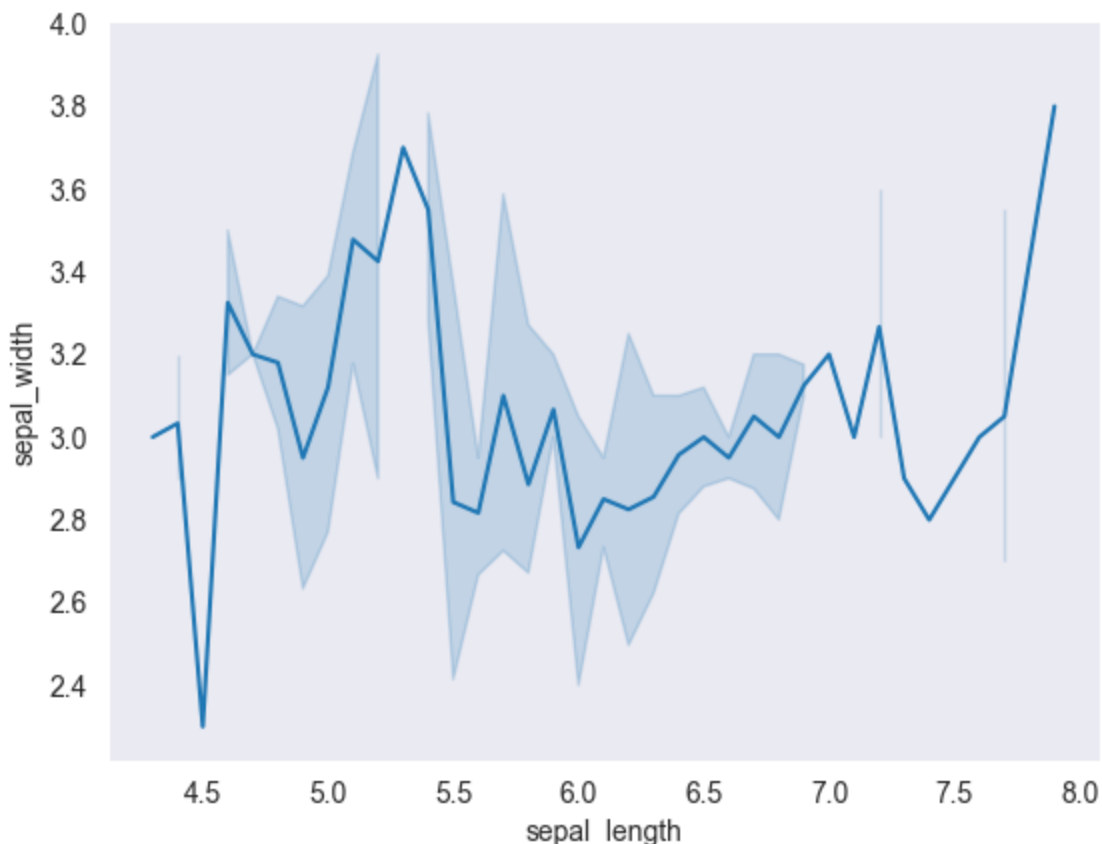
1.

```
In [12]: # loading dataset
        """
        Iris dataset contains five columns such as Petal Length, Petal Width, Sepal
        Length, Sepal Width and Species Type. Iris is a flowering plant, the researchers
        have measured various features of the different iris flowers and recorded them
        digitally.
        """
        data = sns.load_dataset("iris")

        # draw lineplot
        sns.lineplot(x="sepal_length", y="sepal_width", data=data)
```

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1498: FutureWarning: is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead
 if pd.api.types.is_categorical_dtype(vector):
 e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1498: FutureWarning: is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead
 if pd.api.types.is_categorical_dtype(vector):
 e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.
 with pd.option_context('mode.use_inf_as_na', True):
 e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.
 with pd.option_context('mode.use_inf_as_na', True):

Out[12]: <Axes: xlabel='sepal_length', ylabel='sepal_width'>



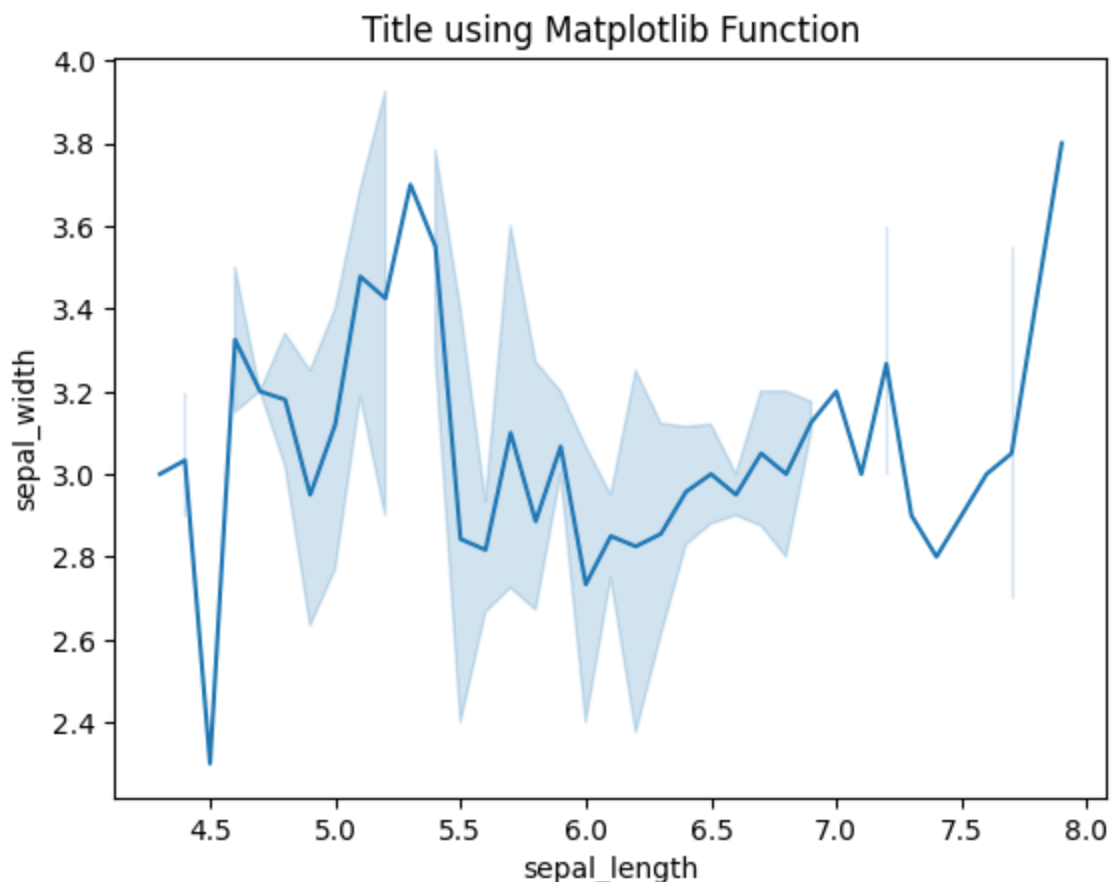
=> Kết quả của đoạn mã này sẽ là một biểu đồ lineplot hiển thị mối quan hệ giữa độ dài của lá đài (sepal_length) và độ rộng của lá đài (sepal_width) của các mẫu hoa Iris. Biểu đồ lineplot thường được sử dụng để thể hiện sự biến thiên của dữ liệu theo thời gian hoặc giữa các giá trị liên quan. Trong trường hợp này, nó giúp bạn xem xét mối quan hệ giữa hai đặc trưng của hoa Iris.

2.

```
In [3]: # Seaborn is built on the top of Matplotlib. It means that Seaborn can be used with Matplotlib.
# Using Seaborn with Matplotlib
# We will be using the above example and will add the title to the plot using the Matplotlib.
# Draw lineplot
sns.lineplot(x="sepal_length",y="sepal_width",data=data)

#setting the title using Matplotlib
plt.title('Title using Matplotlib Function')
plt.show()
```

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1498: FutureWarning: is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead
if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1498: FutureWarning: is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead
if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.
with pd.option_context('mode.use_inf_as_na', True):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.
with pd.option_context('mode.use_inf_as_na', True):



- `sns.lineplot(x="sepal_length", y="sepal_width", data=data)`: Đoạn này sử dụng Seaborn để vẽ biểu đồ dạng lineplot. Nó lấy dữ liệu từ cột "sepal_length" (độ dài của lá đài) làm trục x và từ cột "sepal_width" (độ rộng của lá đài) làm trục y trong tập dữ liệu "data". Kết quả là một biểu đồ lineplot hiển thị mối quan hệ giữa độ dài lá đài và độ rộng lá đài của các mẫu hoa Iris.
- `plt.xlim(5)`: Dòng này sử dụng hàm xlim của Matplotlib để thiết lập giới hạn cho trục x của biểu đồ. Ở đây, trục x sẽ bắt đầu từ giá trị 0 (mặc định) và được giới hạn ở giá trị 5. Điều này có nghĩa rằng biểu đồ sẽ chỉ hiển thị dữ liệu cho các mẫu có độ dài lá đài từ 0 đến 5.
- `plt.show()`: Dòng này sử dụng để hiển thị biểu đồ đã được vẽ sau khi bạn đã thiết lập giới hạn cho trục x.

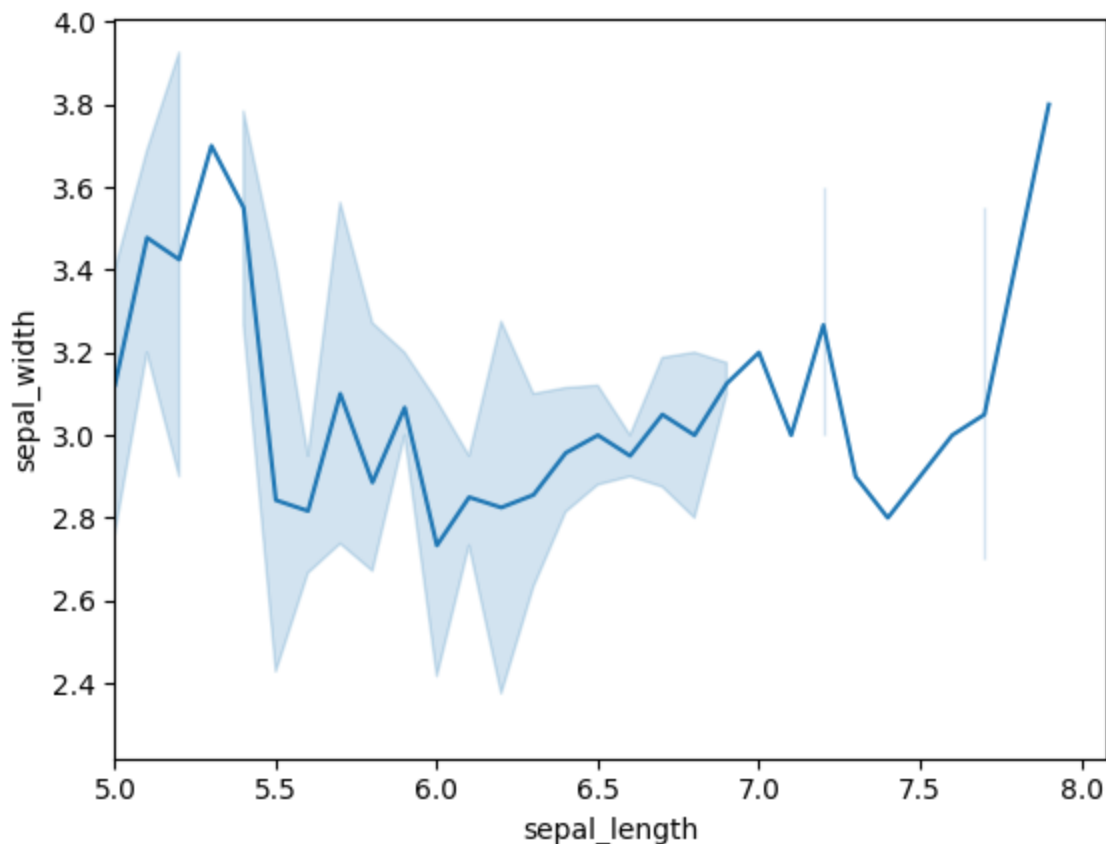
=> Kết quả của đoạn mã này sẽ là một biểu đồ lineplot hiển thị mối quan hệ giữa độ dài lá đài và độ rộng lá đài của các mẫu hoa Iris, và biểu đồ này sẽ bắt đầu từ 0 đến 5 trên trục x.

3.

```
In [4]: # Setting the xlim and ylim
# draw lineplot
sns.lineplot(x="sepal_length", y="sepal_width", data=data)

# setting the x limit of the plot vẽ bd với x=5
plt.xlim(5)
plt.show()
```

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1498: FutureWarning: is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead
 if pd.api.types.is_categorical_dtype(vector):
 e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1498: FutureWarning: is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead
 if pd.api.types.is_categorical_dtype(vector):
 e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.
 with pd.option_context('mode.use_inf_as_na', True):
 e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.
 with pd.option_context('mode.use_inf_as_na', True):



- `sns.lineplot(x="sepal_length", y="sepal_width", data=data)`: Đoạn này sử dụng Seaborn để vẽ biểu đồ dạng lineplot. Nó lấy dữ liệu từ cột "sepal_length" (độ dài của lá đài) làm trục x và từ cột "sepal_width" (độ rộng của lá đài) làm trục y trong tập dữ liệu "data". Kết quả là một

biểu đồ lineplot hiển thị mối quan hệ giữa độ dài lá đài và độ rộng lá đài của các mẫu hoa Iris.

- `plt.xlim(5)`: Dòng này sử dụng hàm `xlim` của Matplotlib để thiết lập giới hạn cho trục x của biểu đồ. Ở đây, trục x sẽ bắt đầu từ giá trị 0 (mặc định) và được giới hạn ở giá trị 5. Điều này có nghĩa rằng biểu đồ sẽ chỉ hiển thị dữ liệu cho các mẫu có độ dài lá đài từ 0 đến 5.
- `plt.show()`: Dòng này sử dụng để hiển thị biểu đồ đã được vẽ sau khi bạn đã thiết lập giới hạn cho trục x.

=> Kết quả của đoạn mã này sẽ là một biểu đồ lineplot hiển thị mối quan hệ giữa độ dài lá đài và

4.

In [5]: *# Customizing Seaborn Plots*

```
"""
set_style() method is used to set the aesthetic of the plot. It means it affects
things like the color of the axes, whether the grid is active or not, or other
aesthetic elements. There are five themes available in Seaborn:
"""

# darkgrid, whitegrid, dark, white, ticks
# set_style(style=None, rc=None)

# draw lineplot
sns.lineplot(x="sepal_length", y="sepal_width", data=data)

# changing the theme to dark
sns.set_style("dark")
plt.show()
```

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1498: FutureWarning: is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is_categorical_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1498: FutureWarning: is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

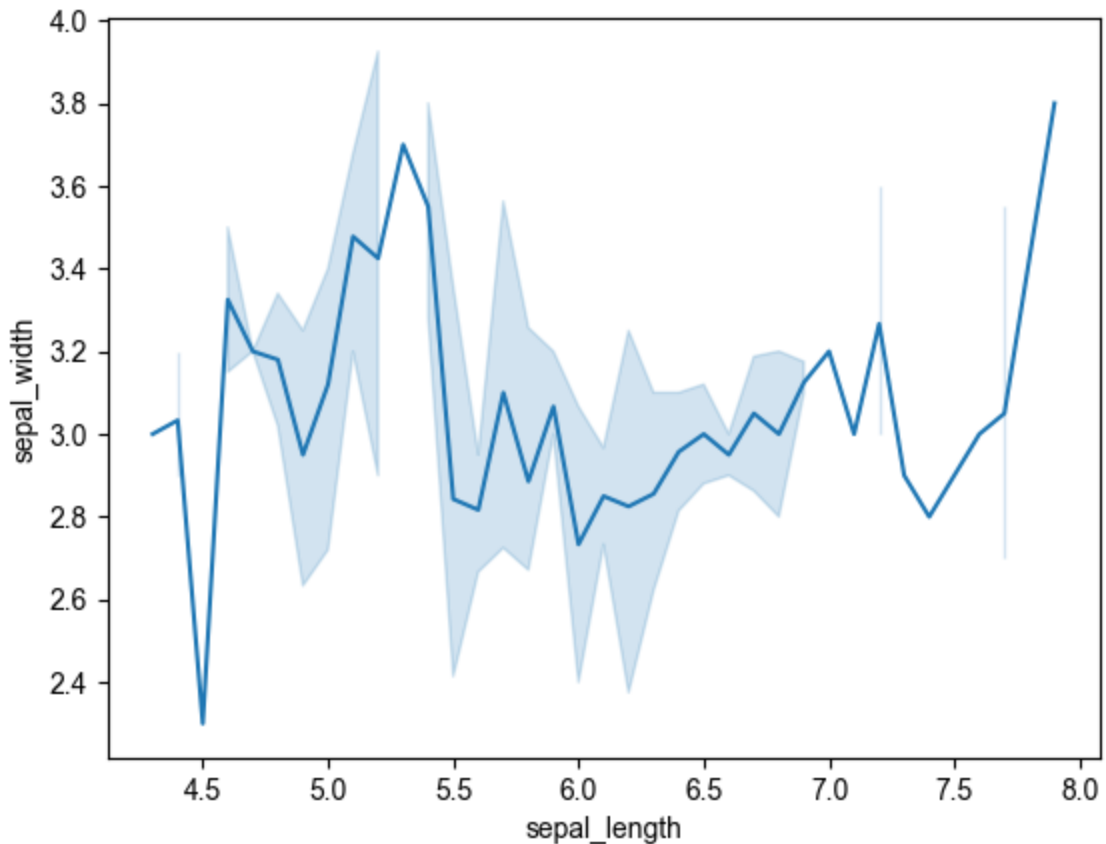
if pd.api.types.is_categorical_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.

with pd.option_context('mode.use_inf_as_na', True):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.

with pd.option_context('mode.use_inf_as_na', True):



- `sns.lineplot(x="sepal_length", y="sepal_width", data=data)`: Đoạn này sử dụng Seaborn để vẽ biểu đồ dạng lineplot. Nó lấy dữ liệu từ cột "sepal_length" (độ dài của lá đài) làm trục x và từ cột "sepal_width" (độ rộng của lá đài) làm trục y trong tập dữ liệu "data". Kết quả là một biểu đồ lineplot hiển thị mối quan hệ giữa độ dài lá đài và độ rộng lá đài của các mẫu hoa Iris.
- `plt.xlim(5)`: Dòng này sử dụng hàm `xlim` của Matplotlib để thiết lập giới hạn cho trục x của biểu đồ. Ở đây, trục x sẽ bắt đầu từ giá trị 0 (mặc định) và được giới hạn ở giá trị 5. Điều này có nghĩa rằng biểu đồ sẽ chỉ hiển thị dữ liệu cho các mẫu có độ dài lá đài từ 0 đến 5.
- `plt.show()`: Dòng này sử dụng để hiển thị biểu đồ đã được vẽ sau khi bạn đã thiết lập giới hạn cho trục x.

=> Kết quả của đoạn mã này sẽ là một biểu đồ lineplot hiển thị mối quan hệ giữa độ dài lá đài và độ rộng lá đài của các mẫu hoa Iris, và biểu đồ này sẽ bắt đầu từ 0 đến 5 trên trục x.

5.

```
In [ ]: #Removal of Spines
#Spines are the lines noting the data boundaries and connecting the axis tick marks. It can be removed using
#sns.despine(left = True)
# draw lineplot
sns.lineplot(x="sepal_length", y="sepal_width", data=data)
# Removing the spines
sns.despine(left=True)
plt.show()
```

- `sns.lineplot(x="sepal_length", y="sepal_width", data=data)`: Dòng này sử dụng Seaborn để vẽ biểu đồ dạng lineplot dựa trên dữ liệu từ tập dữ liệu "data". Nó lấy dữ liệu từ cột "sepal_length" làm trục x và từ cột "sepal_width" làm trục y để hiển thị mối quan hệ giữa độ dài lá đài và độ rộng lá đài của các mẫu hoa Iris.
- `sns.despine(left=True)`: Dòng này sử dụng hàm `despine` của Seaborn để tùy chỉnh biểu đồ bằng cách loại bỏ đường viền (spines) của biểu đồ. Tham số `left=True` chỉ định rằng đường viền ở bên trái của biểu đồ sẽ được loại bỏ. Điều này tạo ra một biểu đồ chỉ có các đường viền ở bên phải và phía trên.
- `plt.show()`: Dòng này sử dụng để hiển thị biểu đồ đã được vẽ, sau khi đã loại bỏ đường viền ở bên trái.

=> Kết quả của đoạn mã này sẽ là một biểu đồ lineplot hiển thị mối quan hệ giữa độ dài lá đài và độ rộng lá đài của các mẫu hoa Iris. Biểu đồ này sẽ không có đường viền ở phía bên trái, như đã được tùy chỉnh bằng hàm `despine`.

6.

```
In [ ]: #Changing the figure Size
# changing the figure size
plt.figure(figsize = (2, 4))
# draw lineplot
sns.lineplot(x="sepal_length", y="sepal_width", data=data)
# Removing the spines
sns.despine()
plt.show()
```

- `plt.figure(figsize=(2, 4))`: Dòng này sử dụng `plt.figure()` để tạo một hình (figure) mới để vẽ biểu đồ. Tham số `figsize` được sử dụng để thiết lập kích thước của hình. Trong trường hợp này, kích thước được thiết lập thành (2, 4), tức là chiều rộng là 2 đơn vị và chiều cao là 4 đơn vị.
- `sns.lineplot(x="sepal_length", y="sepal_width", data=data)`: Dòng này sử dụng Seaborn để vẽ biểu đồ dạng lineplot dựa trên dữ liệu từ tập dữ liệu "data". Nó lấy dữ liệu từ cột "sepal_length" làm trục x và từ cột "sepal_width" làm trục y để hiển thị mối quan hệ giữa độ dài lá đài và độ rộng lá đài của các mẫu hoa Iris.
- `sns.despine()`: Dòng này sử dụng hàm `despine` của Seaborn để loại bỏ đường viền (spines) của biểu đồ. Mặc định, nếu bạn không chỉ định tham số nào cho `despine()`, thì nó sẽ loại bỏ tất cả các đường viền ở bên trên và bên phải của biểu đồ, giữ lại đường viền ở bên trái và bên dưới.
- `plt.show()`: Dòng này sử dụng để hiển thị biểu đồ đã được vẽ, với kích thước tùy chỉnh và đường viền đã được loại bỏ.

=> Kết quả của đoạn mã này sẽ là một biểu đồ lineplot hiển thị mối quan hệ giữa độ dài lá đài và độ rộng lá đài của các mẫu hoa Iris. Biểu đồ này sẽ có kích thước là (2, 4) và sẽ không có đường viền ở bên trên và bên phải.

7.

```
In [ ]: #Scaling the plots
        """
        It can be done using the set_context() method. It allows us to override default
        parameters. This affects things like the size of the labels, lines, and other
        elements of the plot, but not the overall style. The base context is "notebook",
        and the other contexts are "paper", "talk", and "poster". font_scale sets the
        font size.
        """
        #set_context(context=None, font_scale=1, rc=None)
        # draw lineplot
        sns.lineplot(x="sepal_length", y="sepal_width", data=data)
        # Setting the scale of the plot
        sns.set_context("poster")
        plt.show()
```

- `sns.lineplot(x="sepal_length", y="sepal_width", data=data)`: Dòng này sử dụng Seaborn để vẽ biểu đồ dạng lineplot dựa trên dữ liệu từ tập dữ liệu "data". Nó lấy dữ liệu từ cột "sepal_length" làm trục x và từ cột "sepal_width" làm trục y để hiển thị mối quan hệ giữa độ dài lá đài và độ rộng lá đài của các mẫu hoa Iris.
- `sns.set_context("poster")`: Dòng này sử dụng hàm `set_context` của Seaborn để thiết lập ngữ cảnh (context) của biểu đồ thành "poster". "Poster" là một trong các ngữ cảnh được Seaborn hỗ trợ, và nó tạo ra biểu đồ có kích thước lớn hơn và phù hợp cho việc hiển thị trên bảng thông báo hoặc in ấn lớn hơn. Ngữ cảnh "poster" thường được sử dụng để làm cho các biểu đồ trở nên rõ ràng và dễ đọc hơn trong các báo cáo lớn.
- `plt.show()`: Dòng này sử dụng để hiển thị biểu đồ đã được vẽ, với ngữ cảnh "poster" đã được áp dụng.

=> Kết quả của đoạn mã này sẽ là một biểu đồ lineplot hiển thị mối quan hệ giữa độ dài lá đài và độ rộng lá đài của các mẫu hoa Iris. Biểu đồ này sẽ có kích thước lớn hơn so với mặc định và phù hợp cho việc hiển thị thông tin trên bảng thông báo hoặc in ấn lớn hơn.

8.

```
In [ ]: #Setting the Style Temporarily
#axes_style() method is used to set the style temporarily. It is used along with the with statement.
#axes_style(style=None, rc=None)

def plot():
    sns.lineplot(x="sepal_length", y="sepal_width", data=data)
with sns.axes_style('darkgrid'):
    # Adding the subplot
    plt.subplot(211)
    plot()

plt.subplot(212)
plot()
```

- `def plot():` - Đây là việc định nghĩa một hàm tên là `plot()`. Hàm này sẽ được sử dụng để vẽ biểu đồ `lineplot`.
- `sns.lineplot(x="sepal_length", y="sepal_width", data=data)` - Đây là cuộc gọi đến hàm `lineplot` của Seaborn để vẽ biểu đồ `lineplot`. Biểu đồ này sẽ sử dụng dữ liệu từ cột "sepal_length" làm trục x và từ cột "sepal_width" làm trục y, sử dụng dữ liệu từ biến "data".
- `with sns.axes_style('darkgrid'):` - Đây là một khối `with` sử dụng `sns.axes_style()` của Seaborn để thiết lập kiểu (style) của trục cho subplot. Trong trường hợp này, kiểu đã được thiết lập thành "darkgrid", có nghĩa là biểu đồ sẽ có một lưới màu sáng trên nền đậm.
- `plt.subplot(211)` - Dòng này sử dụng Matplotlib để tạo một subplot có kích thước 2x1 và chọn subplot thứ nhất (2 hàng, 1 cột, chọn subplot ở hàng thứ nhất).
- `plot()` - Dòng này gọi hàm `plot()` để vẽ biểu đồ `lineplot` trong subplot thứ nhất.
- `plt.subplot(212)` - Dòng này sử dụng Matplotlib để chọn subplot thứ hai (2 hàng, 1 cột, chọn subplot ở hàng thứ hai).
- `plot()` - Dòng này gọi hàm `plot()` một lần nữa để vẽ biểu đồ `lineplot` trong subplot thứ hai.

=> Kết quả của đoạn mã này sẽ là hai biểu đồ `lineplot` nằm dưới nhau trong một khung subplot. Biểu đồ trên có kiểu trục "darkgrid", trong khi biểu đồ dưới không có kiểu trục được thiết lập. Cả hai biểu đồ sẽ sử dụng cùng tập dữ liệu từ biến "data" và hiển thị mối quan hệ giữa độ dài lá đài và độ rộng lá đài của các mẫu hoa Iris.

9.

```
In [ ]: #Color Palette
#color_palette() method is used to give colors to the plot.
#palplot() is used to deal with the color palettes and plots the color palette as a horizontal array.
# current color palette
palette = sns.color_palette()
# plots the color palette as a
# horizontal array
sns.palplot(palette)
plt.show()
```

- `palette = sns.color_palette():` Dòng này sử dụng hàm `color_palette()` của Seaborn để tạo một bảng màu (color palette). Màu sắc trong bảng màu này là các màu mặc định được sử dụng bởi Seaborn. Khi bạn gọi `color_palette()` mà không truyền bất kỳ tham số nào, nó sẽ trả về bảng màu mặc định của Seaborn và lưu nó trong biến `palette`.

- `sns.palplot(palette)`: Dòng này sử dụng hàm `palplot()` của Seaborn để hiển thị bảng màu. `palplot()` cho phép bạn xem trước và hiển thị tất cả các màu trong bảng màu. Bạn truyền biến `palette` đã tạo từ bước trước vào hàm `palplot()` để hiển thị bảng màu này.
- `plt.show()`: Dòng này sử dụng để hiển thị bảng màu trên biểu đồ.

=> Kết quả của đoạn mã này sẽ là một biểu đồ hiển thị bảng màu mặc định của Seaborn. Bảng màu này chứa các màu sắc được sử dụng để vẽ biểu đồ và trang trí trong Seaborn.

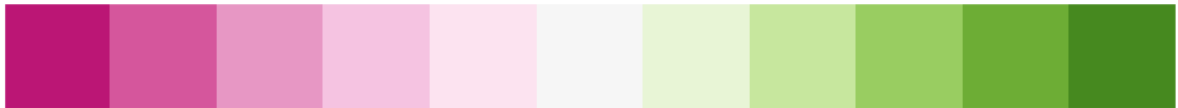
10.

In [15]:

```
"""
Diverging Color Palette: uses two different colors where each color depicts
different points ranging from a common point in either direction. Consider a
range of -10 to 10 so the value from -10 to 0 takes one color and values from 0
to 10 take another.
"""

# current color palette
palette = sns.color_palette('PiYG', 11)

# diverging color palette
sns.palplot(palette)
plt.show()
```



- `palette = sns.color_palette('PiYG', 11)`: Dòng này sử dụng hàm `color_palette()` của Seaborn để tạo một bảng màu mới. Tham số đầu tiên 'PiYG' xác định chuỗi màu bạn muốn sử dụng. Trong trường hợp này, 'PiYG' đại diện cho một chuỗi màu có tên là "Pink to Yellow-Green", có nghĩa là bảng màu này sẽ chứa các màu từ hồng (pink) đến màu xanh lá cây và chia thành 11 màu sắc khác nhau.
- `sns.palplot(palette)`: Dòng này sử dụng hàm `palplot()` của Seaborn để hiển thị bảng màu với các màu sắc từ bảng màu được tạo ở bước trước. Bảng màu này sẽ hiển thị 11 màu sắc được lấy từ chuỗi màu 'PiYG'.
- `plt.show()`: Dòng này sử dụng để hiển thị bảng màu trên biểu đồ.

=> Kết quả của đoạn mã này sẽ là một biểu đồ hiển thị bảng màu được tạo dựa trên chuỗi màu 'PiYG' với 11 màu sắc khác nhau, từ hồng đến xanh lá cây. Bảng màu này có thể được sử dụng để tùy chỉnh màu sắc trong các biểu đồ và đồ thị Seaborn.

11.

In []:

```
"""
Sequential Color Palette: is used where the distribution ranges from a lower
value to a higher value. To do this add the character 's' to the color passed in
the color palette.
"""

# current color palette
palette = sns.color_palette('Greens', 11)
# sequential color palette
sns.palplot(palette)
plt.show()
```

- `palette = sns.color_palette('Greens', 11)`: Dòng này sử dụng hàm `color_palette()` của Seaborn để tạo một bảng màu mới. Tham số đầu tiên 'Greens' xác định chuỗi màu bạn muốn sử dụng. Trong trường hợp này, 'Greens' đại diện cho một chuỗi màu có tên là "Greens", chứa các màu sắc xanh lá cây và chia thành 11 màu sắc khác nhau.
- `sns.palplot(palette)`: Dòng này sử dụng hàm `palplot()` của Seaborn để hiển thị bảng màu với các màu sắc từ bảng màu được tạo ở bước trước. Bảng màu này sẽ hiển thị 11 màu sắc xanh lá cây khác nhau, lấy từ chuỗi màu 'Greens'.
- `plt.show()`: Dòng này sử dụng để hiển thị bảng màu trên biểu đồ.

=> Kết quả của đoạn mã này sẽ là một biểu đồ hiển thị bảng màu được tạo dựa trên chuỗi màu 'Greens' với 11 màu sắc khác nhau, tạo thành một chuỗi màu liên tục chuyển từ màu xanh nhạt đến màu xanh đậm. Bảng màu này có thể được sử dụng để tùy chỉnh màu sắc trong các biểu đồ và đồ thị Seaborn.

12.

```

In [ ]: #Setting the default Color Palette
        """
        set_palette() method is used to set the default color palette for all the plots.
        The arguments for both color_palette() and set_palette() is same. set_palette()
        changes the default matplotlib parameters.
        """

        def plot():
            sns.lineplot(x="sepal_length", y="sepal_width", data=data)

        # setting the default color palette
        sns.set_palette('vlag')
        plt.subplot(211)

        # plotting with the color palette
        # as vlag
        plot()

        # setting another default color palette
        sns.set_palette('Accent')
        plt.subplot(212)
        plot()

        plt.show()

```

- `def plot():` - Đây là việc định nghĩa một hàm tên là `plot()`. Hàm này sẽ được sử dụng để vẽ biểu đồ `lineplot`.
- `sns.set_palette('vlag')` - Dòng này sử dụng hàm `set_palette()` của Seaborn để thiết lập bảng màu mặc định thành "vlag". "vlag" là một chuỗi màu cụ thể của Seaborn.
- `plt.subplot(211)` - Dòng này sử dụng Matplotlib để tạo một subplot có kích thước 2x1 và chọn subplot thứ nhất (2 hàng, 1 cột, chọn subplot ở hàng thứ nhất).
- `plot()` - Dòng này gọi hàm `plot()` để vẽ biểu đồ `lineplot` trong subplot thứ nhất, với bảng màu "vlag" đã được thiết lập.
- `sns.set_palette('Accent')` - Dòng này sử dụng hàm `set_palette()` của Seaborn để thay đổi bảng màu mặc định thành "Accent", một chuỗi màu khác.
- `plt.subplot(212)` - Dòng này sử dụng Matplotlib để chọn subplot thứ hai (2 hàng, 1 cột, chọn subplot ở hàng thứ hai).
- `plot()` - Dòng này gọi hàm `plot()` một lần nữa để vẽ biểu đồ `lineplot` trong subplot thứ hai, với bảng màu "Accent" đã được thiết lập.
- `plt.show()` - Dòng này sử dụng để hiển thị cả hai biểu đồ `lineplot` trong hai subplot khác nhau, mỗi biểu đồ được vẽ bằng một bảng màu mặc định khác nhau.

=> Kết quả của đoạn mã này sẽ là hai biểu đồ `lineplot` nằm dưới nhau trong một khung subplot, mỗi biểu đồ sử dụng một bảng màu mặc định khác nhau ("vlag" và "Accent").

13.

```
In [7]: #Multiple plots with Seaborn
#Using Matplotlib: add_axes(), subplot(), and subplot2grid().
def graph():
    sns.lineplot(x="sepal_length", y="sepal_width", data=data)

# Creating a new figure with width = 5 inches
# and height = 4 inches
fig = plt.figure(figsize =(5, 4))

# Creating first axes for the figure
ax1 = fig.add_axes([0.1, 0.1, 0.8, 0.8])

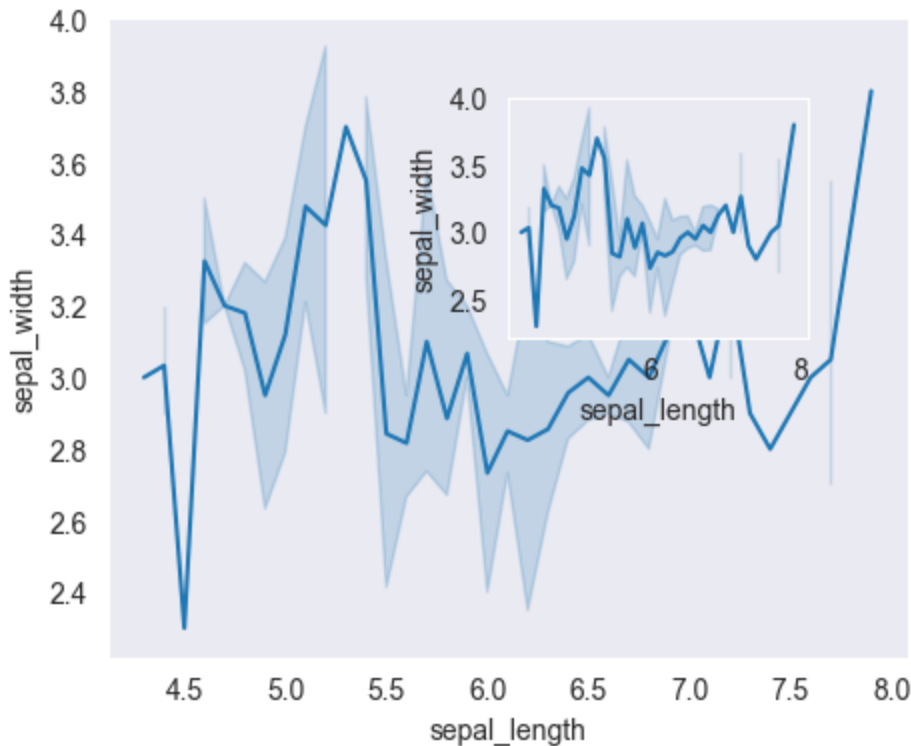
#plotting the graph
graph()

# Creating second axes for the figure
ax2 = fig.add_axes([0.5, 0.5, 0.3, 0.3])

#plotting the graph
graph()

plt.show()
```

```
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, Categori
calDtype) instead
    if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, Categori
calDtype) instead
    if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning:
use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN bef
ore operating instead.
    with pd.option_context('mode.use_inf_as_na', True):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning:
use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN bef
ore operating instead.
    with pd.option_context('mode.use_inf_as_na', True):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, Categori
calDtype) instead
    if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, Categori
calDtype) instead
    if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning:
use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN bef
ore operating instead.
    with pd.option_context('mode.use_inf_as_na', True):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning:
use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN bef
ore operating instead.
    with pd.option_context('mode.use_inf_as_na', True):
```

- `def graph():` - Đây là việc định nghĩa một hàm tên là `graph()`. Hàm này sẽ được sử dụng để vẽ biểu đồ lineplot.
- `fig = plt.figure(figsize=(5, 4))` - Dòng này tạo một hình vẽ (figure) mới bằng Matplotlib với kích thước là 5 inches (chiều rộng) x 4 inches (chiều cao).
- `ax1 = fig.add_axes([0.1, 0.1, 0.8, 0.8])` - Dòng này tạo một đối tượng trục (axes) đầu tiên bên trong hình vẽ. Tham số trong `add_axes()` xác định vị trí và kích thước của đối tượng trục. Trong trường hợp này, `[0.1, 0.1, 0.8, 0.8]` xác định rằng đối tượng trục này sẽ nằm ở vị trí bắt đầu từ `(0.1, 0.1)` và có kích thước chiếm 80% chiều rộng và 80% chiều cao của hình vẽ.
- `graph()` - Dòng này gọi hàm `graph()` để vẽ biểu đồ lineplot trong đối tượng trục `ax1`.
- `ax2 = fig.add_axes([0.5, 0.5, 0.3, 0.3])` - Dòng này tạo một đối tượng trục (axes) thứ hai bên trong hình vẽ. Cũng giống như trước, `[0.5, 0.5, 0.3, 0.3]` xác định vị trí và kích thước của đối tượng trục này.
- `graph()` - Dòng này gọi hàm `graph()` một lần nữa để vẽ biểu đồ lineplot trong đối tượng trục `ax2`.
- `plt.show()` - Dòng này sử dụng để hiển thị hình vẽ với hai đối tượng trục và hai biểu đồ lineplot được vẽ bên trong chúng.

=> Kết quả của đoạn mã này sẽ là một hình vẽ (figure) có hai đối tượng trục (axes) khác nhau. Mỗi đối tượng trục chứa một biểu đồ lineplot, và chúng nằm ở các vị trí và kích thước khác nhau trong hình vẽ.

14.

In [16]: *#Using subplot() method*

```
def graph():
    sns.lineplot(x="sepal_length", y="sepal_width", data=data)

# Adding the subplot at the specified
# grid position
plt.subplot(121)
graph()

# Adding the subplot at the specified
# grid position
plt.subplot(122)
graph()

plt.show()
```

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1498: FutureWarning: is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is_categorical_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1498: FutureWarning: is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is_categorical_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.

with pd.option_context('mode.use_inf_as_na', True):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.

with pd.option_context('mode.use_inf_as_na', True):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1498: FutureWarning: is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is_categorical_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1498: FutureWarning: is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

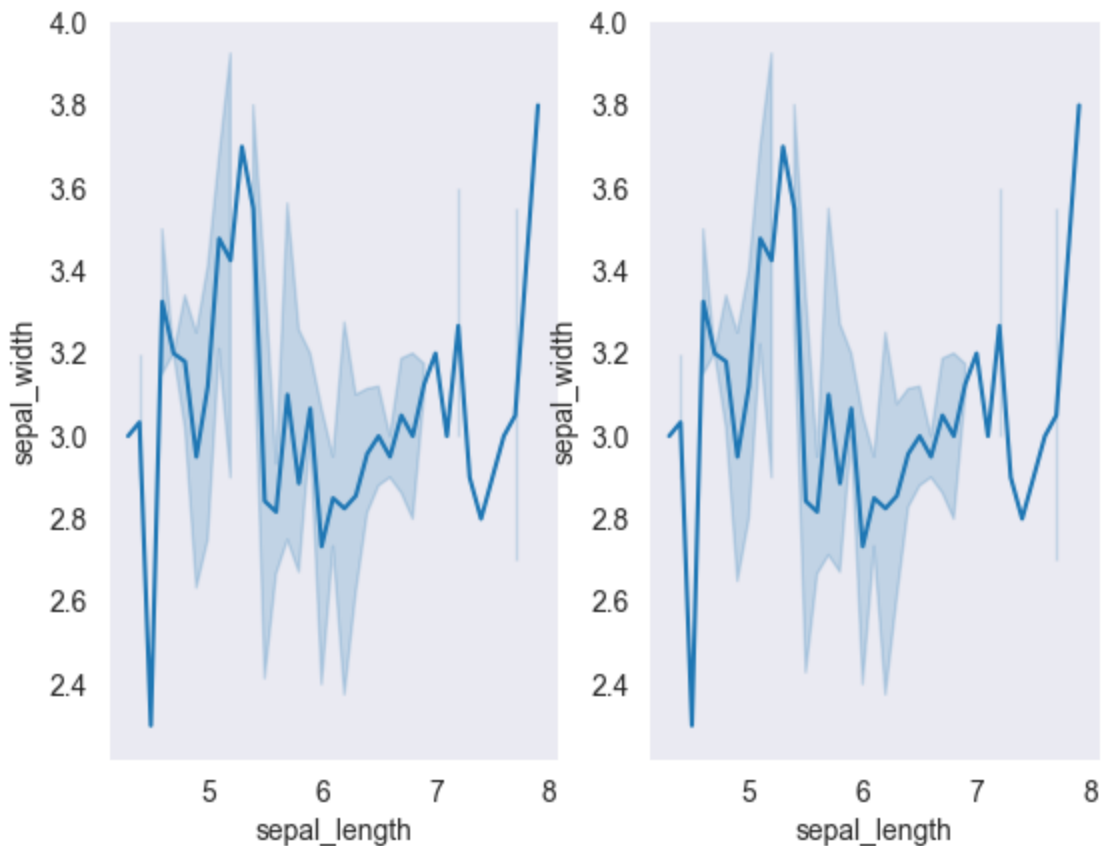
if pd.api.types.is_categorical_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.

with pd.option_context('mode.use_inf_as_na', True):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.

with pd.option_context('mode.use_inf_as_na', True):



- `def graph():` - Đây là việc định nghĩa một hàm tên là `graph()`. Hàm này sẽ được sử dụng để vẽ biểu đồ lineplot.
- `plt.subplot(121)` - Dòng này sử dụng Matplotlib để tạo một đối tượng trục (axes) ở vị trí đầu tiên trong một lưới 1x2 (một hàng và hai cột). Số 121 cho biết rằng chúng ta đang tạo đối tượng trục ở hàng thứ nhất và cột thứ nhất trong lưới 1x2.
- `graph()` - Dòng này gọi hàm `graph()` để vẽ biểu đồ lineplot trong đối tượng trục ở vị trí (1,1).
- `plt.subplot(122)` - Dòng này sử dụng Matplotlib để tạo một đối tượng trục (axes) ở vị trí thứ hai trong lưới 1x2. Số 122 cho biết rằng chúng ta đang tạo đối tượng trục ở hàng thứ nhất và cột thứ hai trong lưới 1x2.
- `graph()` - Dòng này gọi hàm `graph()` một lần nữa để vẽ biểu đồ lineplot trong đối tượng trục ở vị trí (1,2).
- `plt.show()` - Dòng này sử dụng để hiển thị hình vẽ với hai đối tượng trục và hai biểu đồ lineplot được vẽ bên trong chúng.

=> Kết quả của đoạn mã này sẽ là một hình vẽ (figure) chứa hai đối tượng trục (axes) nằm cạnh nhau trong lưới 1x2. Mỗi đối tượng trục chứa một biểu đồ lineplot, và chúng được hiển thị trong hai cột riêng biệt.

15.

```
In [ ]: #Using subplot() method
def graph():
    sns.lineplot(x="sepal_length", y="sepal_width", data=data)

# Adding the subplot at the specified
# grid position
plt.subplot(121)
graph()

# Adding the subplot at the specified
# grid position
plt.subplot(122)
graph()

plt.show()
```

- `def graph():` - Đây là việc định nghĩa một hàm tên là `graph()`. Hàm này sẽ được sử dụng để vẽ biểu đồ `lineplot`.
- `plt.subplot(121)` - Dòng này sử dụng Matplotlib để tạo một đối tượng trục (axes) ở vị trí đầu tiên trong một lưới 1x2 (một hàng và hai cột). Số 121 cho biết rằng chúng ta đang tạo đối tượng trục ở hàng thứ nhất và cột thứ nhất trong lưới 1x2.
- `graph()` - Dòng này gọi hàm `graph()` để vẽ biểu đồ `lineplot` trong đối tượng trục ở vị trí (1,1).
- `plt.subplot(122)` - Dòng này sử dụng Matplotlib để tạo một đối tượng trục (axes) ở vị trí thứ hai trong lưới 1x2. Số 122 cho biết rằng chúng ta đang tạo đối tượng trục ở hàng thứ nhất và cột thứ hai trong lưới 1x2.
- `graph()` - Dòng này gọi hàm `graph()` một lần nữa để vẽ biểu đồ `lineplot` trong đối tượng trục ở vị trí (1,2).
- `plt.show()` - Dòng này sử dụng để hiển thị hình vẽ với hai đối tượng trục và hai biểu đồ `lineplot` được vẽ bên trong chúng.

=> Kết quả của đoạn mã này sẽ là một hình vẽ (figure) chứa hai đối tượng trục (axes) nằm cạnh nhau trong lưới 1x2. Mỗi đối tượng trục chứa một biểu đồ `lineplot`, và chúng được hiển thị trong hai cột riêng biệt. Tương tự như mã trước, có thể có lỗi do biến `data` không được định nghĩa trong phạm vi hiện tại.

16.

```
In [ ]: #Using subplot2grid() method
def graph():
    sns.lineplot(x="sepal_length", y="sepal_width", data=data)

# adding the subplots
axes1 = plt.subplot2grid ((7, 1), (0, 0), rowspan = 2, colspan = 1)
graph()

axes2 = plt.subplot2grid ((7, 1), (2, 0), rowspan = 2, colspan = 1)
graph()

axes3 = plt.subplot2grid ((7, 1), (4, 0), rowspan = 2, colspan = 1)
graph()

#Using Seaborn: Using FacetGrid() method
"""
FacetGrid class helps in visualizing distribution of one variable as well as the
relationship between multiple variables separately within subsets of your dataset
using multiple panels.
"""
#seaborn.FacetGrid( data, \**kwargs)
plot = sns.FacetGrid(data, col="species")
plot.map(plt.plot, "sepal_width")

plt.show()
```

17.

```
In [ ]: #Using PairGrid() method
#Subplot grid for plotting pairwise relationships in a dataset.
"""
This class maps each variable in a dataset onto a column and row in a grid of
multiple axes. Different axes-level plotting functions can be used to draw
bivariate plots in the upper and lower triangles, and the marginal distribution
of each variable can be shown on the diagonal.
"""
#seaborn.PairGrid( data, \**kwargs)
data = sns.load_dataset("flights")

plot = sns.PairGrid(data)
plot.map(plt.plot)

plt.show()
```

18.

In [9]: *#Creating Different Types of Plots**#Relplot()*

"""

This function provides us the access to some other different axes-level functions which shows the relationships between two variables with semantic mappings of subsets. It is plotted using the relplot() method

"""

*#seaborn.relplot(x=None, y=None, data=None, **kwargs)**# loading dataset*

data = sns.load_dataset("iris")

creating the relplot

sns.relplot(x='sepal_width', y='species', data=data)

plt.show()

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1498: FutureWarning: is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is_categorical_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1498: FutureWarning: is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is_categorical_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1498: FutureWarning: is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is_categorical_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1498: FutureWarning: is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is_categorical_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1498: FutureWarning: is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is_categorical_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1498: FutureWarning: is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

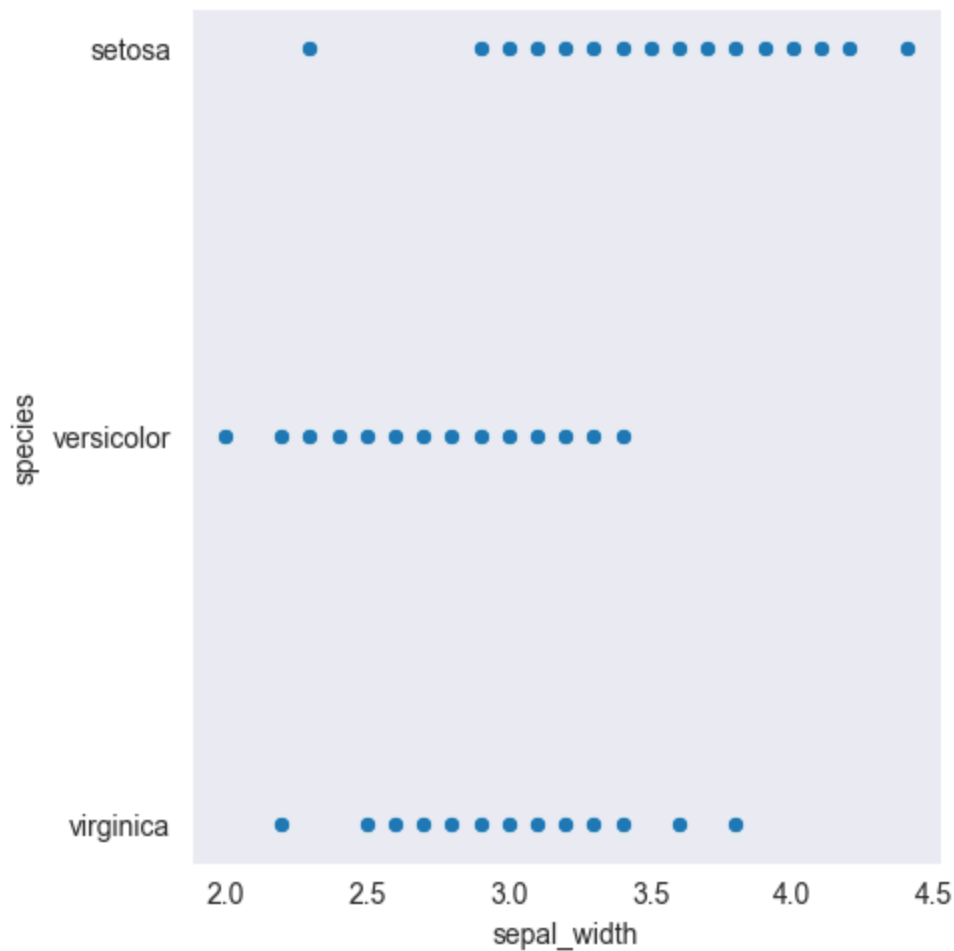
if pd.api.types.is_categorical_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1498: FutureWarning: is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is_categorical_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\axisgrid.py:118: UserWarning: The figure layout has changed to tight

self.figure.tight_layout(*args, **kwargs)



19.

```
In [ ]: """
The scatter plot is a mainstay of statistical visualization. It depicts the
joint distribution of two variables using a cloud of points, where each point
represents an observation in the dataset. This depiction allows the eye to infer
a substantial amount of information about whether there is any meaningful
relationship between them. It is plotted using the scatterplot() method
"""
#seaborn.scatterplot(x=None, y=None, data=None, **kwargs)

sns.scatterplot(x='sepal_length', y='sepal_width', data=data)
plt.show()
```

20.

In [14]:

```

"""
The scatter plot is a mainstay of statistical visualization. It depicts the
joint distribution of two variables using a cloud of points, where each point
represents an observation in the dataset. This depiction allows the eye to infer
a substantial amount of information about whether there is any meaningful
relationship between them. It is plotted using the scatterplot() method
"""

#seaborn.scatterplot(x=None, y=None, data=None, **kwargs)

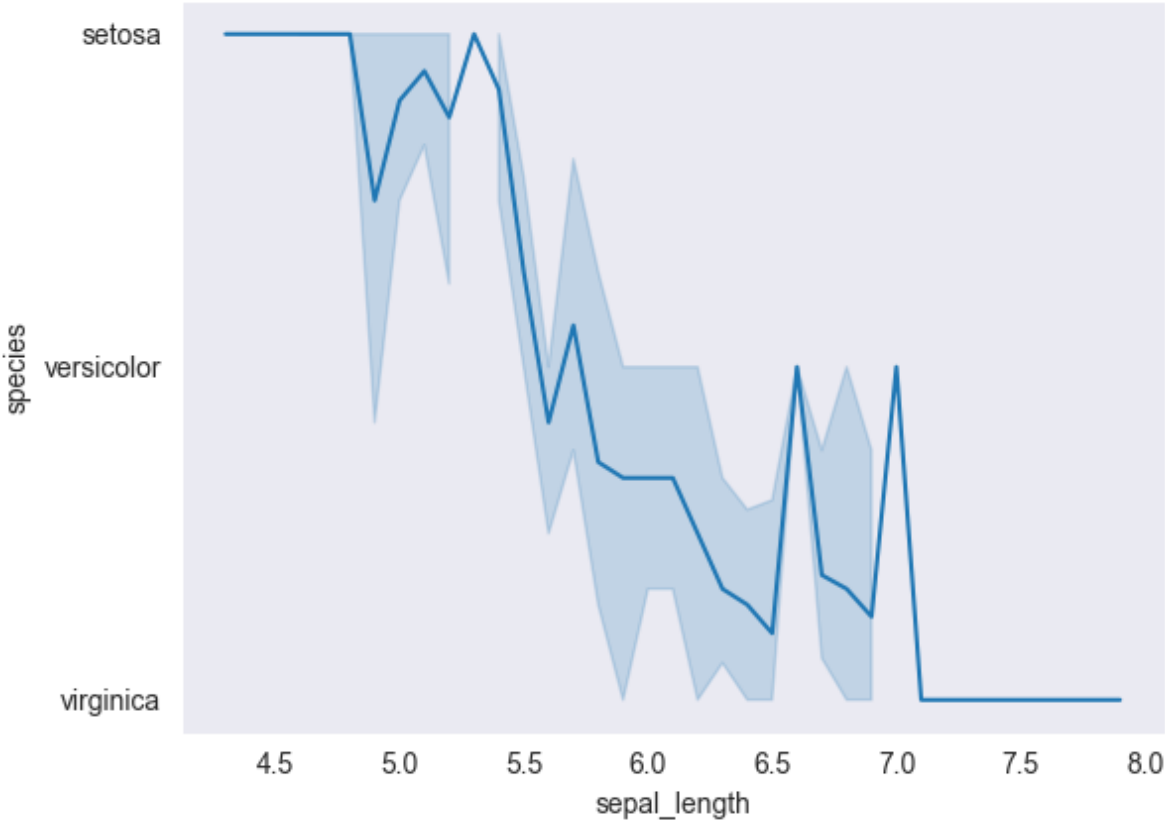
sns.lineplot(x='sepal_length', y='species', data=data)
plt.show()

```

```

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, Categori
calDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, Categori
calDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, Categori
calDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning:
use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN bef
ore operating instead.
  with pd.option_context('mode.use_inf_as_na', True):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning:
use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN bef
ore operating instead.
  with pd.option_context('mode.use_inf_as_na', True):

```

21.

In [13]:

```

"""
Categorical Plots are used where we have to visualize relationship between two
numerical values. A more specialized approach can be used if one of the main
variable is categorical which means such variables that take on a fixed and
limited number of possible values.
"""
"""
A barplot is basically used to aggregate the categorical data according to some
methods and by default its the mean. It can also be understood as a visualization
of the group by action. To use this plot we choose a categorical column for the x
axis and a numerical column for the y axis and we see that it creates a plot
taking a mean per categorical column. It can be created using the barplot()
method.
"""
#barplot([x, y, hue, data, order, hue_order, ...])

sns.barplot(x='species', y='sepal_length', data=data)
plt.show()

```

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1498: FutureWarning: is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

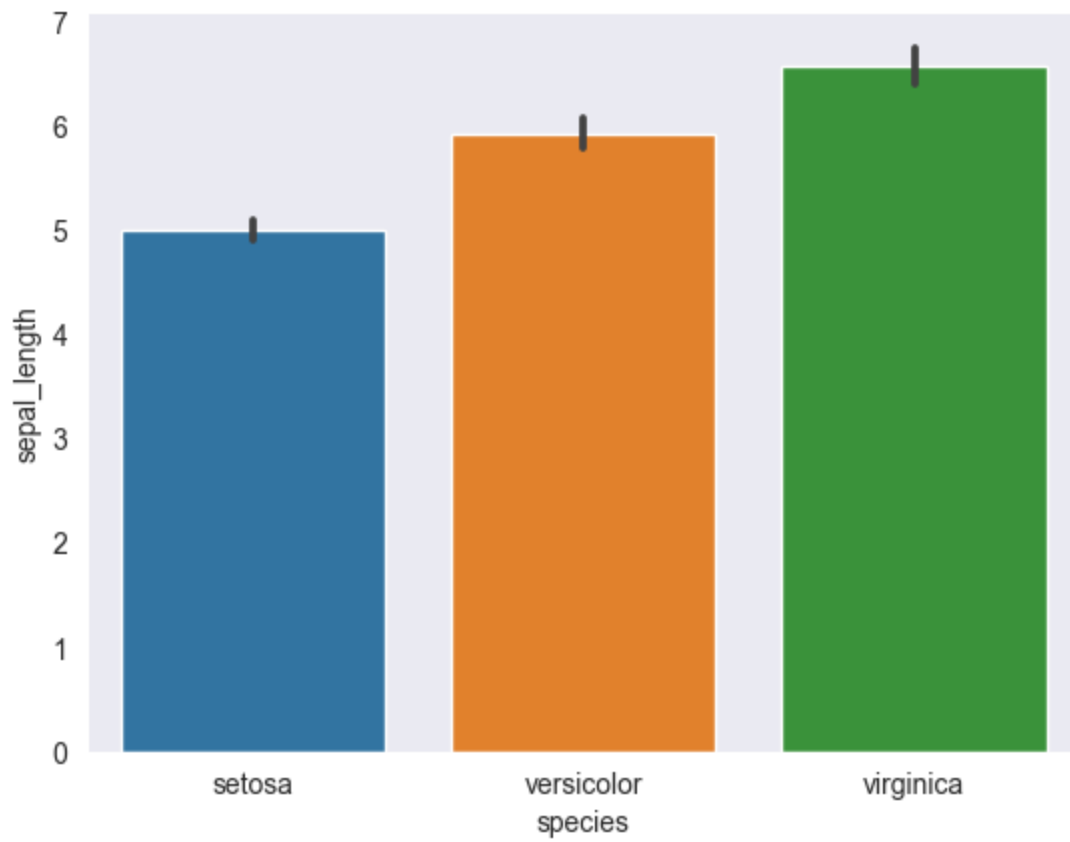
if pd.api.types.is_categorical_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1498: FutureWarning: is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is_categorical_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1498: FutureWarning: is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is_categorical_dtype(vector):



22.

In [17]:

```

"""
A countplot basically counts the categories and returns a count of their
occurrences. It is one of the most simple plots provided by the seaborn library.
It can be created using the countplot() method.
"""

#countplot([x, y, hue, data, order, ...])
sns.countplot(x='species', data=data)
plt.show()

"""
A boxplot is sometimes known as the box and whisker plot. It shows the
distribution of the quantitative data that represents the comparisons between
variables. boxplot shows the quartiles of the dataset while the whiskers extend
to show the rest of the distribution i.e. the dots indicating the presence of
outliers. It is created using the boxplot() method.
"""

#boxplot([x, y, hue, data, order, hue_order, ...])
sns.boxplot(x='species', y='sepal_width', data=data)
plt.show()

```

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1498: FutureWarning: is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

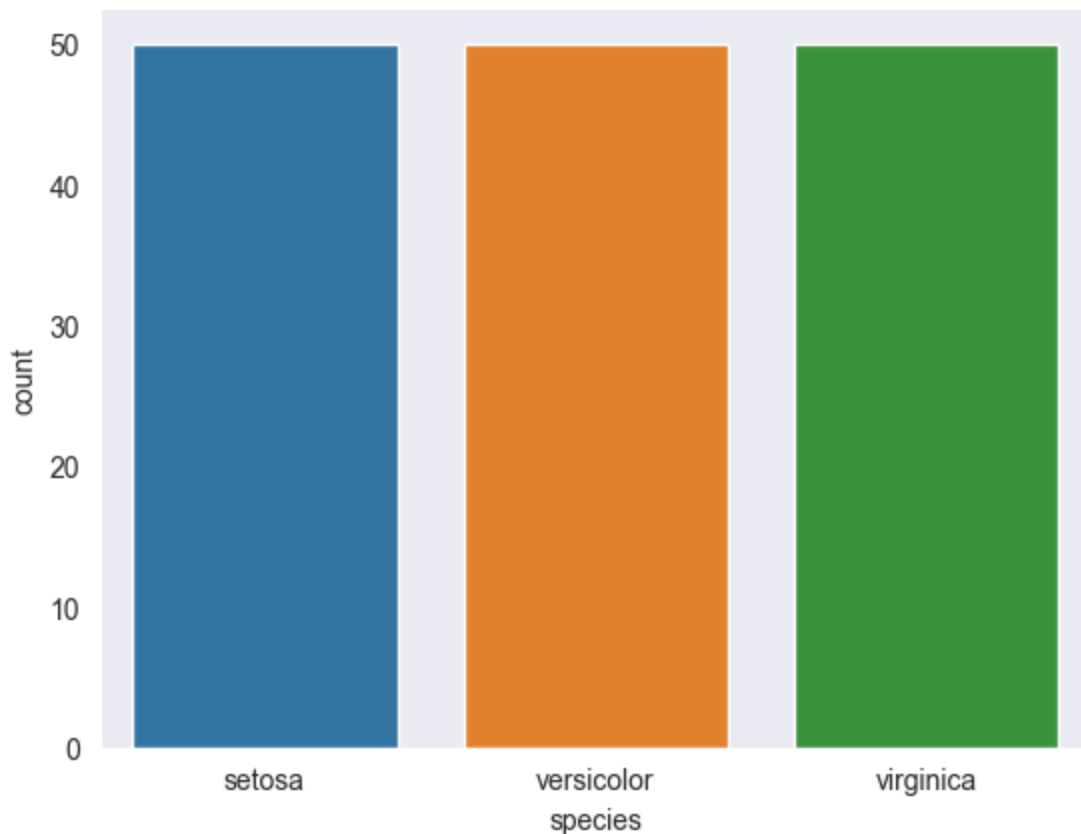
if pd.api.types.is_categorical_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1498: FutureWarning: is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is_categorical_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1498: FutureWarning: is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is_categorical_dtype(vector):



e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1498: FutureWarning: is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

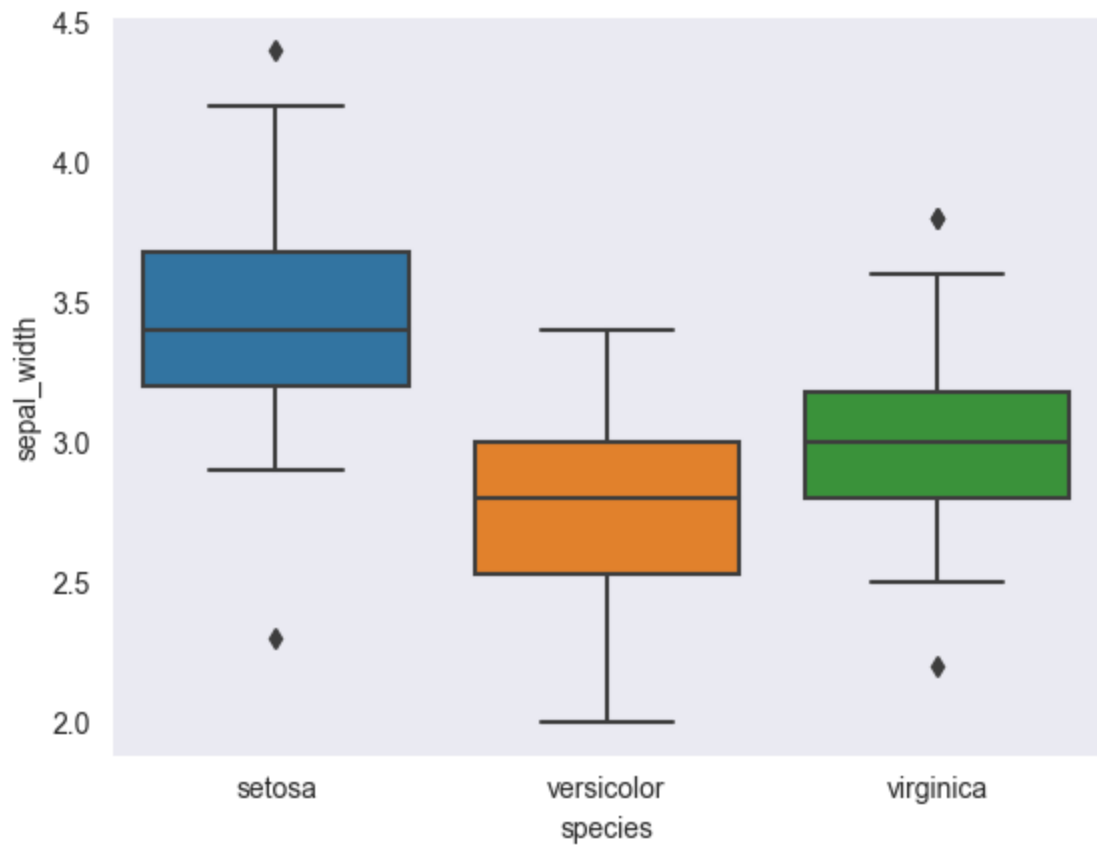
if pd.api.types.is_categorical_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1498: FutureWarning: is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is_categorical_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1498: FutureWarning: is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is_categorical_dtype(vector):



23.

In [18]: `#Violinplot`

"""

It is similar to the boxplot except that it provides a higher, more advanced visualization and uses the kernel density estimation to give a better description about the data distribution. It is created using the `violinplot()` method.

"""

`#violinplot([x, y, hue, data, order, ...])``sns.violinplot(x='species', y='sepal_width', data=data)``plt.show()`

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1498: FutureWarning: `is_categorical_dtype` is deprecated and will be removed in a future version. Use `isinstance(dtype, CategoricalDtype)` instead

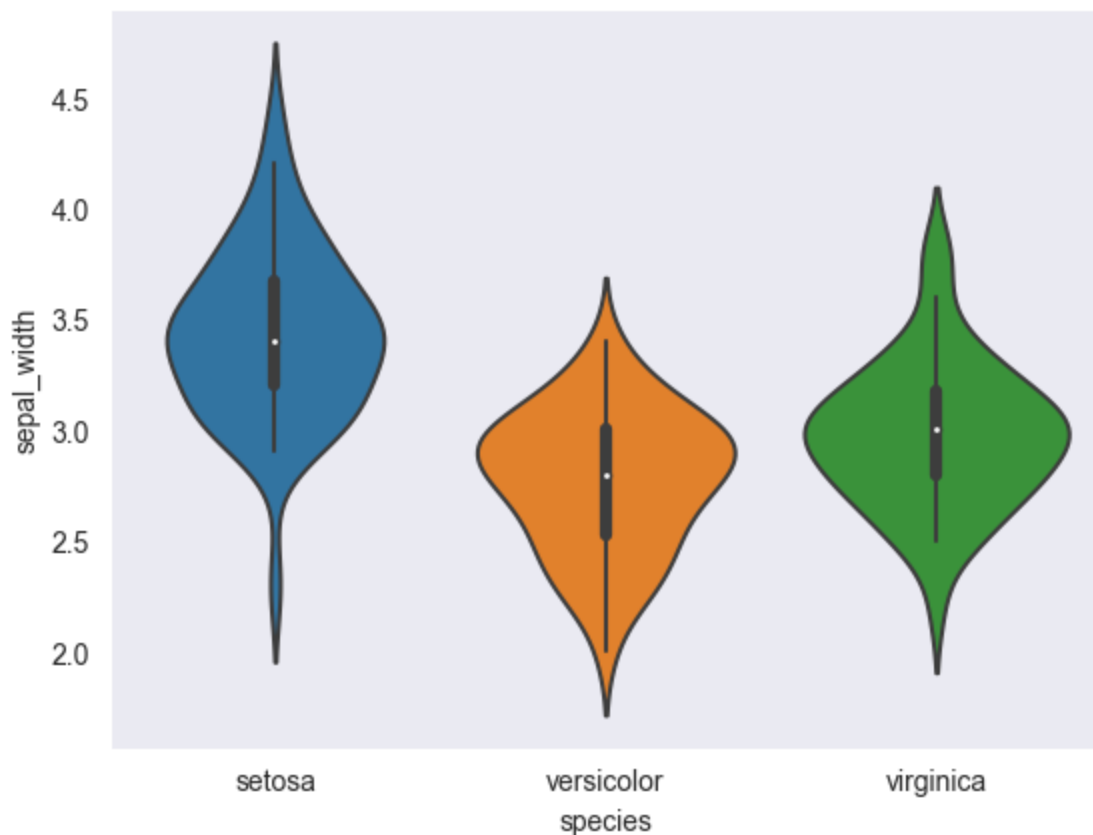
if `pd.api.types.is_categorical_dtype(vector)`:

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1498: FutureWarning: `is_categorical_dtype` is deprecated and will be removed in a future version. Use `isinstance(dtype, CategoricalDtype)` instead

if `pd.api.types.is_categorical_dtype(vector)`:

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1498: FutureWarning: `is_categorical_dtype` is deprecated and will be removed in a future version. Use `isinstance(dtype, CategoricalDtype)` instead

if `pd.api.types.is_categorical_dtype(vector)`:



24.

In [19]:

```

"""
Stripplot: It basically creates a scatter plot based on the category. It is
created using the stripplot() method.
"""
#stripplot([x, y, hue, data, order, ...])
sns.stripplot(x='species', y='sepal_width', data=data)
plt.show()

```

```

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, Categori
calDtype) instead
    if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, Categori
calDtype) instead
    if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, Categori
calDtype) instead
    if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, Categori
calDtype) instead
    if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, Categori
calDtype) instead
    if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, Categori
calDtype) instead
    if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, Categori
calDtype) instead
    if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning:
use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN bef
ore operating instead.
    with pd.option_context('mode.use_inf_as_na', True):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning:
use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN bef
ore operating instead.
    with pd.option_context('mode.use_inf_as_na', True):

```




25.

In [20]:

```

"""
Swarmplot is very similar to the stripplot except the fact that the points are
adjusted so that they do not overlap. Some people also like combining the idea of
a violin plot and a stripplot to form this plot. One drawback to using swarmplot
is that sometimes they don't scale well to really large numbers and takes a lot of
computation to arrange them. So in case we want to visualize a swarmplot properly
we can plot it on top of a violinplot. It is plotted using the swarmplot()
method.
"""
#swarmplot([x, y, hue, data, order, ...])
sns.swarmplot(x='species', y='sepal_width', data=data)
plt.show()

```

```

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, Categori
calDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, Categori
calDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, Categori
calDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, Categori
calDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, Categori
calDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, Categori
calDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, Categori
calDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning:
use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN bef
ore operating instead.
  with pd.option_context('mode.use_inf_as_na', True):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning:
use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN bef
ore operating instead.
  with pd.option_context('mode.use_inf_as_na', True):

```



26.

In [30]:

```

"""
catplot is the most general of all these plots and provides a parameter
called kind to choose the kind of plot we want thus saving us from the trouble of
writing these plots separately. The kind parameter can be bar, violin, swarm etc.
It is plotted using the catplot() method.
"""

#sns.catplot([x, y, hue, data, row, col, ...])
# loading dataset

data = sns.load_dataset("iris")
sns.catplot(x='species', y='sepal_width', data=data)

```

```

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning:
use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.
  with pd.option_context('mode.use_inf_as_na', True):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning:
use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.
  with pd.option_context('mode.use_inf_as_na', True):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\axisgrid.py:118: UserWarning: The figure layout has changed to tight
  self.figure.tight_layout(*args, **kwargs)

```

Out[30]: <seaborn.axisgrid.FacetGrid at 0x2294c673ad0>



27.

In [28]:

```

"""
A histogram is basically used to represent data provided in a form of some
groups. It is an accurate method for the graphical representation of numerical data
distribution. It can be plotted using the histplot() function.
"""

#histplot(data=None, *, x=None, y=None, hue=None, **kwargs)
sns.histplot(x='species', y='sepal_width', data=data)
plt.show()

```

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1498: FutureWarning: is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is_categorical_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1498: FutureWarning: is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is_categorical_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1498: FutureWarning: is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

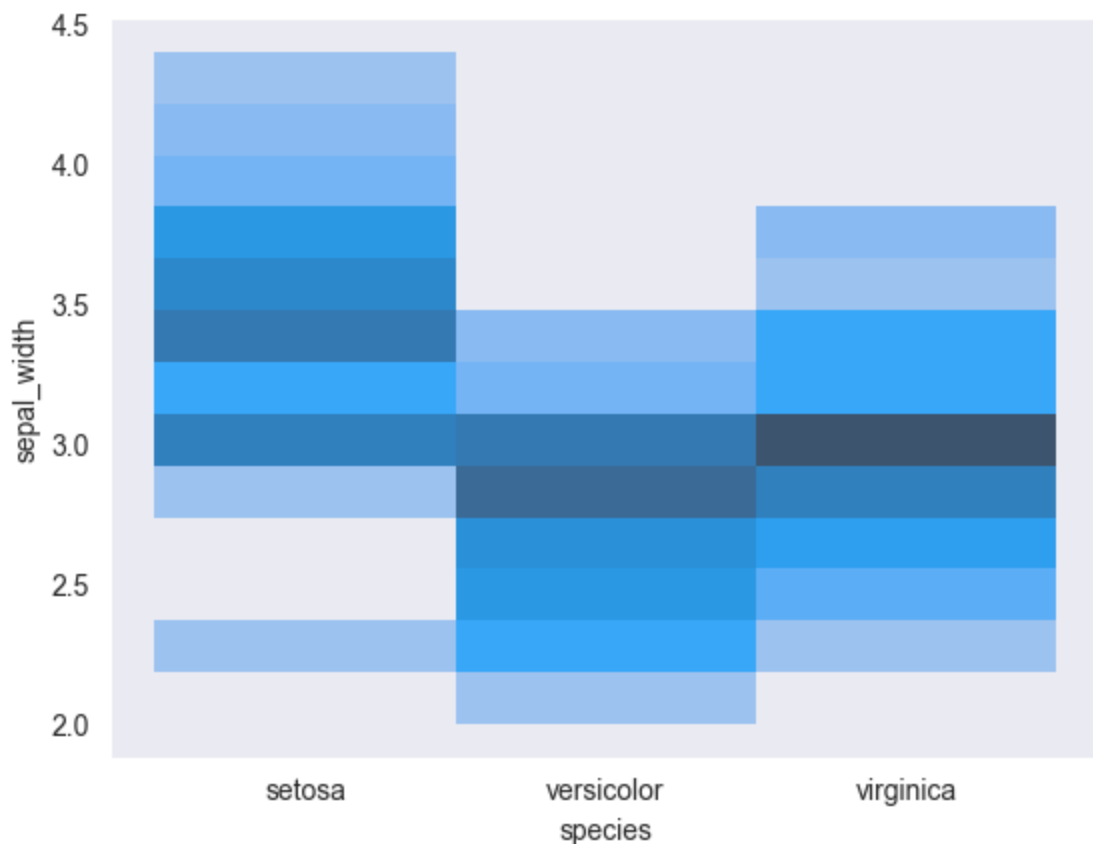
if pd.api.types.is_categorical_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.

with pd.option_context('mode.use_inf_as_na', True):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.

with pd.option_context('mode.use_inf_as_na', True):



28.

In [31]:

```

"""
Distplot is used basically for univariant set of observations and visualizes it
through a histogram i.e. only one observation and hence we choose one particular
column of the dataset. It is potted using the distplot() method.
"""

#distplot(a[, bins, hist, kde, rug, fit, ...])
import seaborn as sns
import matplotlib.pyplot as plt
# loading dataset
data = sns.load_dataset("iris")
sns.distplot(data['sepal_width'])
plt.show()

```

C:\Users\Lan Anh\AppData\Local\Temp\ipykernel_8316\68307821.py:11: 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).

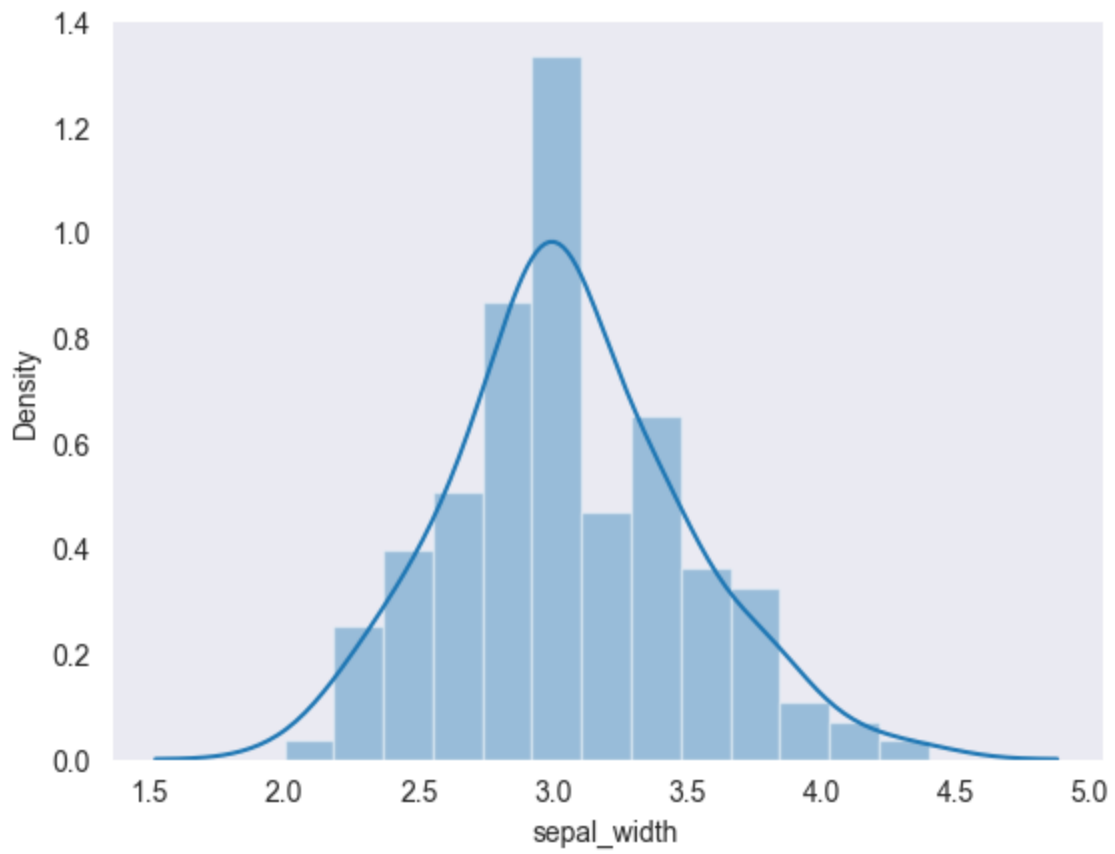
For a guide to updating your code to use the new functions, please see

<https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751> (<https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751>)

```

sns.distplot(data['sepal_width'])
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, Categori
calDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning:
use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN bef
ore operating instead.
  with pd.option_context('mode.use_inf_as_na', True):

```

29.

In [32]:

```

"""
Jointplot is used to draw a plot of two variables with bivariate and univariate
graphs. It basically combines two different plots. It is plotted using the
jointplot() method.
"""

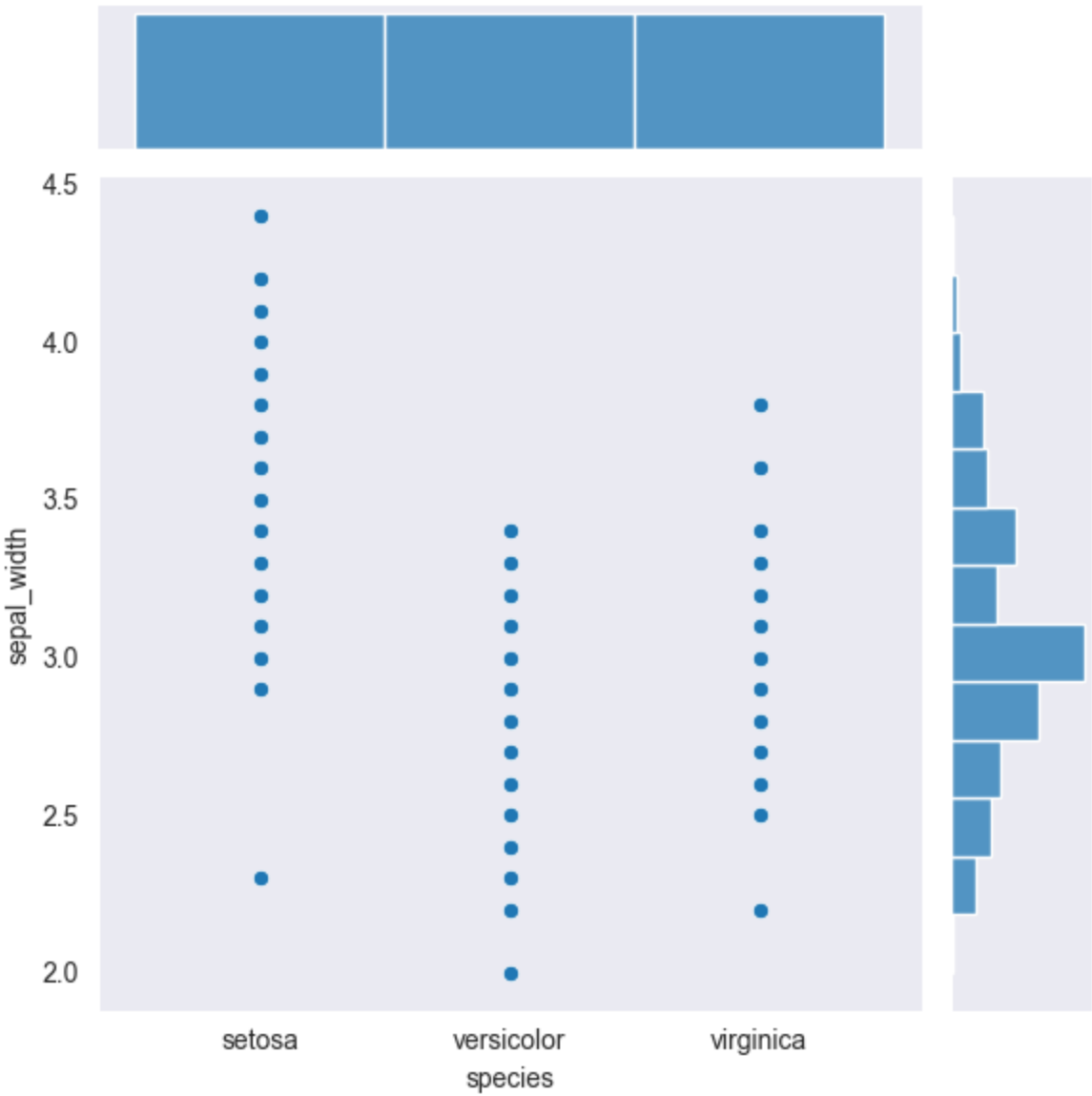
#jointplot(x, y[, data, kind, stat_func, ...])
sns.jointplot(x='species', y='sepal_width', data=data)
plt.show()

```

```

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, Categori
calDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, Categori
calDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, Categori
calDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, Categori
calDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, Categori
calDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, Categori
calDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, Categori
calDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning:
use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN bef
ore operating instead.
  with pd.option_context('mode.use_inf_as_na', True):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, Categori
calDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning:
use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN bef
ore operating instead.
  with pd.option_context('mode.use_inf_as_na', True):

```



30.

In [33]:

```

"""
Pairplot represents pairwise relation across the entire dataframe and supports
an additional argument called hue for categorical separation. What it does
basically is create a jointplot between every possible numerical column and takes
a while if the dataframe is really huge. It is plotted using the pairplot()
method.
"""

#pairplot(data[, hue, hue_order, palette, ...])
sns.pairplot(data=data, hue='species')
plt.show()

```

```

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarni
ng: is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, C
ategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarni
ng: is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, C
ategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarni
ng: is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, C
ategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarni
ng: is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, C
ategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarni
ng: is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, C
ategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):

```

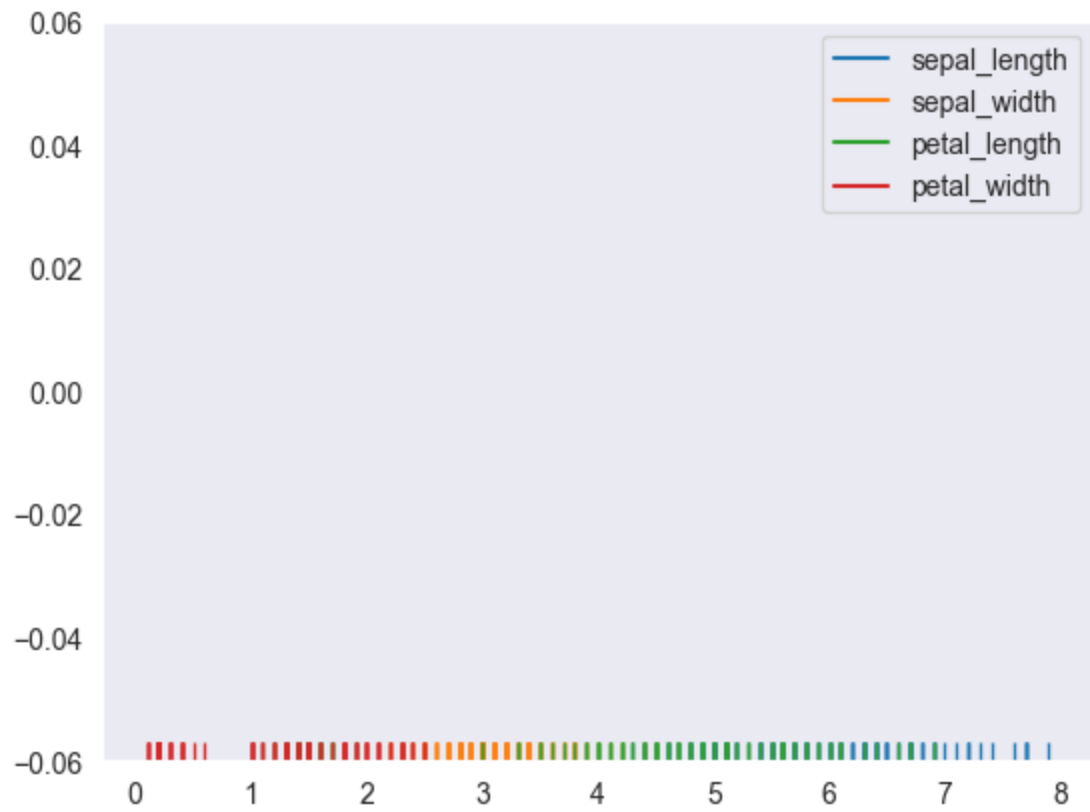
31.

In [34]:

```
"""
Rugplot plots datapoints in an array as sticks on an axis. Just like a distplot
it takes a single column. Instead of drawing a histogram it creates dashes all
across the plot. If you compare it with the joinplot you can see that what a
jointplot does is that it counts the dashes and shows it as bins. It is plotted
using the rugplot() method.
"""

#rugplot(a[, height, axis, ax])
sns.rugplot(data=data)
plt.show()
```

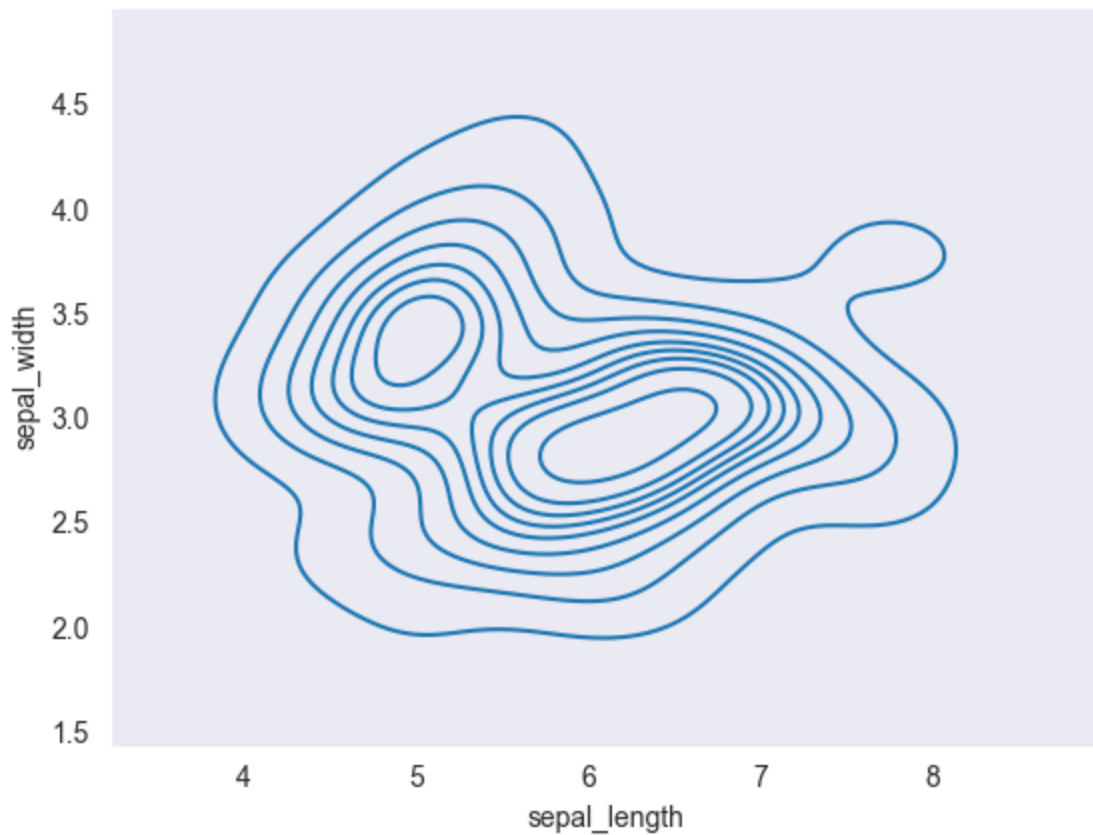
```
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:  
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, Categori  
calDType) instead  
    if pd.api.types.is_categorical_dtype(vector):  
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:  
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, Categori  
calDType) instead  
    if pd.api.types.is_categorical_dtype(vector):  
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:  
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, Categori  
calDType) instead  
    if pd.api.types.is_categorical_dtype(vector):  
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:  
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, Categori  
calDType) instead  
    if pd.api.types.is_categorical_dtype(vector):  
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:  
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, Categori  
calDType) instead  
    if pd.api.types.is_categorical_dtype(vector):  
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:  
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, Categori  
calDType) instead  
    if pd.api.types.is_categorical_dtype(vector):  
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:  
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, Categori  
calDType) instead  
    if pd.api.types.is_categorical_dtype(vector):  
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:  
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, Categori  
calDType) instead  
    if pd.api.types.is_categorical_dtype(vector):  
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning:  
use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN bef  
ore operating instead.  
    with pd.option_context('mode.use_inf_as_na', True):
```



32.

```
In [35]: """
KDE Plot described as Kernel Density Estimate is used for visualizing the
Probability Density of a continuous variable. It depicts the probability density
at different values in a continuous variable. We can also plot a single graph for
multiple samples which helps in more efficient data visualization.
"""
#seaborn.kdeplot(x=None, *, y=None, vertical=False, palette=None, **kwargs)
sns.kdeplot(x='sepal_length', y='sepal_width', data=data)
plt.show()
```

```
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning:
use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.
  with pd.option_context('mode.use_inf_as_na', True):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning:
use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.
  with pd.option_context('mode.use_inf_as_na', True):
```



34.

```
In [ ]: """
The regression plots are primarily intended to add a visual guide that helps to
emphasize patterns in a dataset during exploratory data analyses. Regression
plots as the name suggests creates a regression line between two parameters and
helps to visualize their linear relationships.
"""

"""
lmplo() method can be understood as a function that basically creates a linear
model plot. It creates a scatter plot with a linear fit on top of it.
"""

#seaborn.lmplot(x, y, data, hue=None, col=None, row=None, **kwargs)
# loading dataset

data = sns.load_dataset("tips")
sns.lmplot(x='total_bill', y='tip', data=data)
plt.show()
```

35.

```
In [37]: #regplot() method is also similar to lmplot which creates linear regression model.
#seaborn.regplot(x, y, data=None, x_estimator=None, **kwargs)
# loading dataset
data = sns.load_dataset("tips")
sns.regplot(x='total_bill', y='tip', data=data)
plt.show()
```

