

DATA VISUALIZATION

STEP 1 IMPORT LIBRARIES

```
import seaborn as sns
import matplotlib.pyplot as plt
```

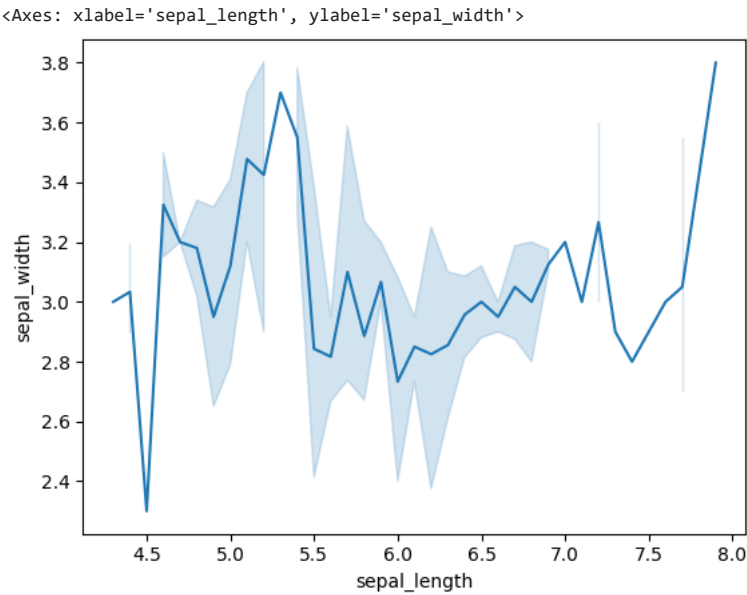
STEP 2 LOAD DATA SET

```
iris = sns.load_dataset("iris")
iris.head()
```

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3.0	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
3	4.6	3.1	1.5	0.2	setosa
4	5.0	3.6	1.4	0.2	setosa

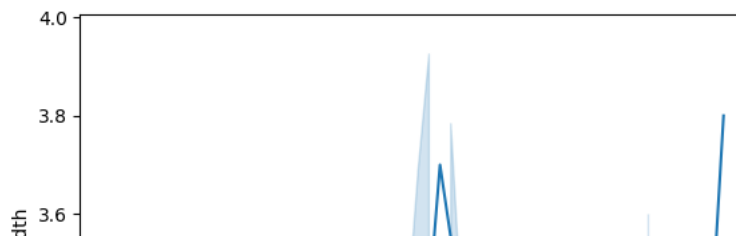
STEP-3 PLOT A GRAPH

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



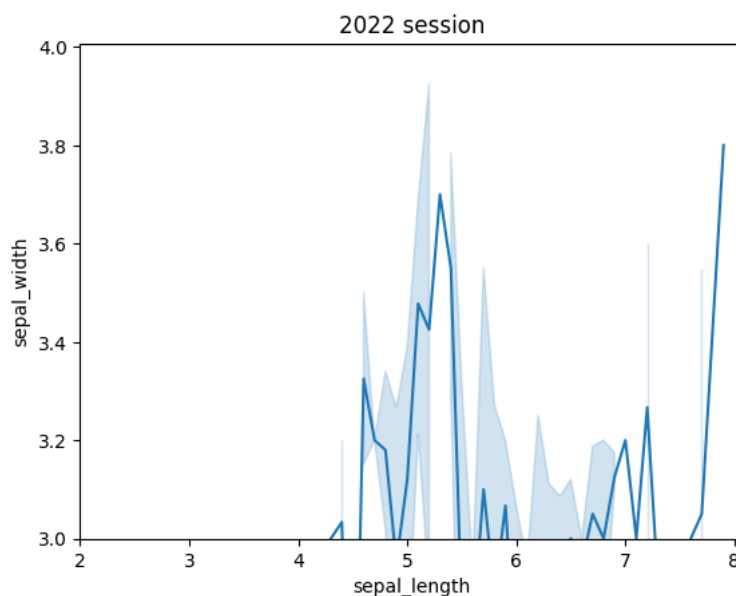
```
sns.lineplot(x="sepal_length" , y="sepal_width", data=iris)
plt.xlim(2)
plt.ylim(3)
```

(3.0, 4.00625)



```
sns.lineplot(x="sepal_length", y="sepal_width", data=iris)
plt.xlim(2)
plt.ylim(3)
plt.title("2022 session")
```

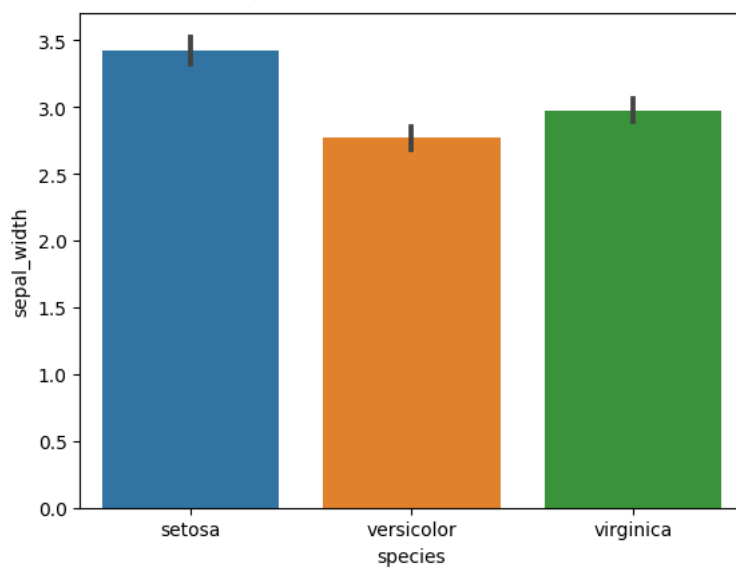
```
Text(0.5, 1.0, '2022 session')
```



BAR PLOT

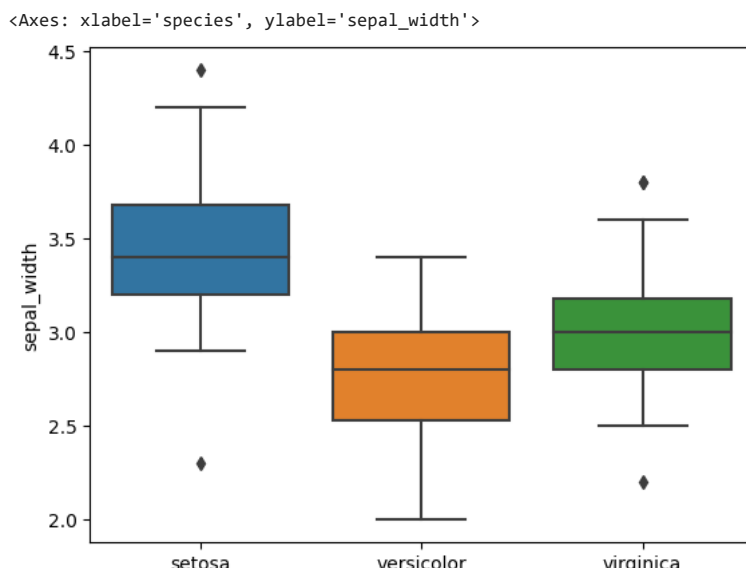
```
sns.barplot(x="species", y="sepal_width", data=iris)
```

<Axes: xlabel='species', ylabel='sepal_width'>



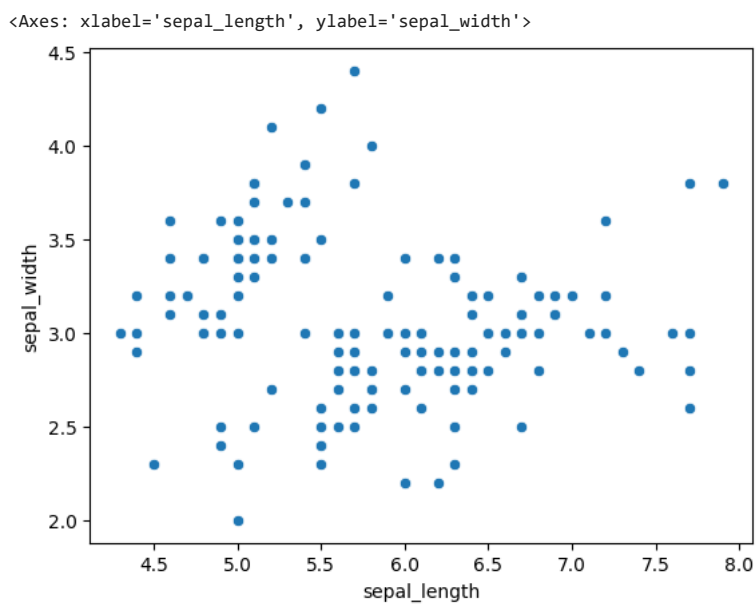
BOX PLOT

```
sns.boxplot(x="species", y="sepal_width", data=iris)
```



▼ SCATTER PLOT

```
sns.scatterplot(x="sepal_length" , y="sepal_width", data=iris)
```



▼ CAT PLOR

```
sns.catplot(x="species" , y="sepal_width", data=iris , color="gold")
```

```
<seaborn.axisgrid.FacetGrid at 0x7f0c44eff0d0>
```

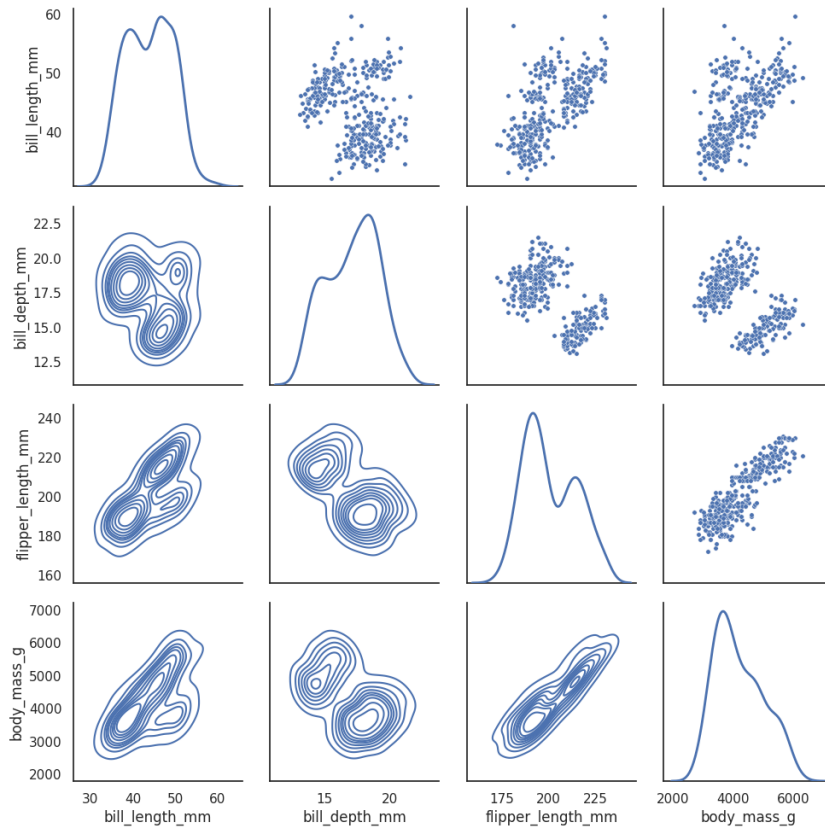


```
import seaborn as sns
sns.set_theme(style="white")
```

```
df = sns.load_dataset("penguins")
```

```
g = sns.PairGrid(df, diag_sharey=False)
g.map_upper(sns.scatterplot, s=15)
g.map_lower(sns.kdeplot)
g.map_diag(sns.kdeplot, lw=2)
```

```
<seaborn.axisgrid.PairGrid at 0x7f0c475c1ae0>
```



```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
sns.set_theme(style="white", rc={"axes.facecolor": (0, 0, 0, 0)})
```

```
# Create the data
rs = np.random.RandomState(1979)
x = rs.randn(500)
g = np.tile(list("ABCDEFGHJIJ"), 50)
df = pd.DataFrame(dict(x=x, g=g))
m = df.g.map(ord)
```

```

df["x"] += m

# Initialize the FacetGrid object
pal = sns.cubehelix_palette(10, rot=-.25, light=.7)
g = sns.FacetGrid(df, row="g", hue="g", aspect=15, height=.5, palette=pal)

# Draw the densities in a few steps
g.map(sns.kdeplot, "x",
      bw_adjust=.5, clip_on=False,
      fill=True, alpha=1, linewidth=1.5)
g.map(sns.kdeplot, "x", clip_on=False, color="w", lw=2, bw_adjust=.5)

# passing color=None to refline() uses the hue mapping
g.refline(y=0, linewidth=2, linestyle="-", color=None, clip_on=False)

# Define and use a simple function to label the plot in axes coordinates
def label(x, color, label):
    ax = plt.gca()
    ax.text(0, .2, label, fontweight="bold", color=color,
           ha="left", va="center", transform=ax.transAxes)

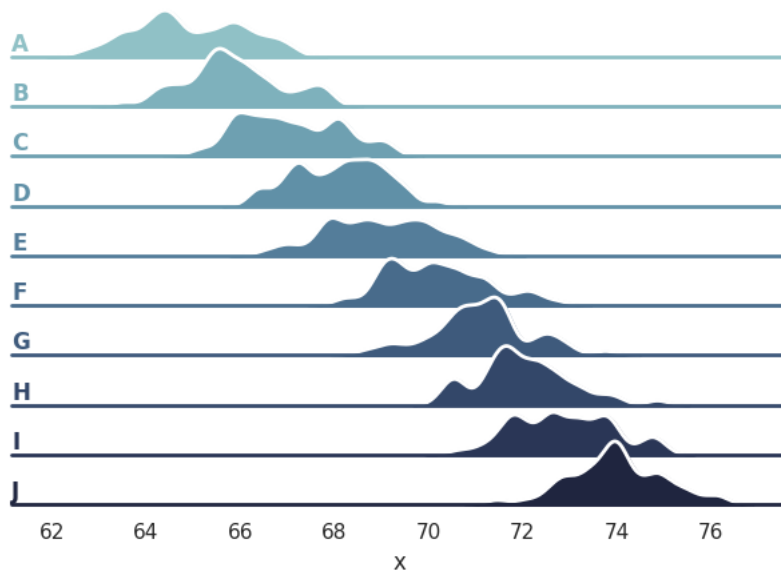
g.map(label, "x")

# Set the subplots to overlap
g.figure.subplots_adjust(hspace=-.25)

# Remove axes details that don't play well with overlap
g.set_titles("")
g.set(yticks=[], ylabel="")
g.despine(bottom=True, left=True)

/usr/local/lib/python3.10/dist-packages/seaborn/axisgrid.py:118: UserWarning: Tight
self._figure.tight_layout(*args, **kwargs)
/usr/local/lib/python3.10/dist-packages/seaborn/axisgrid.py:118: UserWarning: Tight
self._figure.tight_layout(*args, **kwargs)
/usr/local/lib/python3.10/dist-packages/seaborn/axisgrid.py:118: UserWarning: Tight
self._figure.tight_layout(*args, **kwargs)
/usr/local/lib/python3.10/dist-packages/seaborn/axisgrid.py:118: UserWarning: Tight
self._figure.tight_layout(*args, **kwargs)
/usr/local/lib/python3.10/dist-packages/seaborn/axisgrid.py:118: UserWarning: Tight
self._figure.tight_layout(*args, **kwargs)
/usr/local/lib/python3.10/dist-packages/seaborn/axisgrid.py:118: UserWarning: Tight
self._figure.tight_layout(*args, **kwargs)
<seaborn.axisgrid.FacetGrid at 0x7f0c41fed7e0>

```



HEAT MAP

```

import matplotlib.pyplot as plt
import seaborn as sns
sns.set_theme()

# Load the example flights dataset and convert to long-form
flights_long = sns.load_dataset("flights")
flights = flights_long.pivot("month", "year", "passengers")

```

```
# Draw a heatmap with the numeric values in each cell
f, ax = plt.subplots(figsize=(9, 6))
sns.heatmap(flights, annot=True, fmt="d", linewidths=.5, ax=ax)

<ipython-input-23-fd553bdfde69>:7: FutureWarning: In a future version of pandas all
  flights = flights_long.pivot("month", "year", "passengers")
<Axes: xlabel='year', ylabel='month'>
```

