

Data visualization▼ **step 1: Import library**

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

▼ **step 2 Load Dataset**

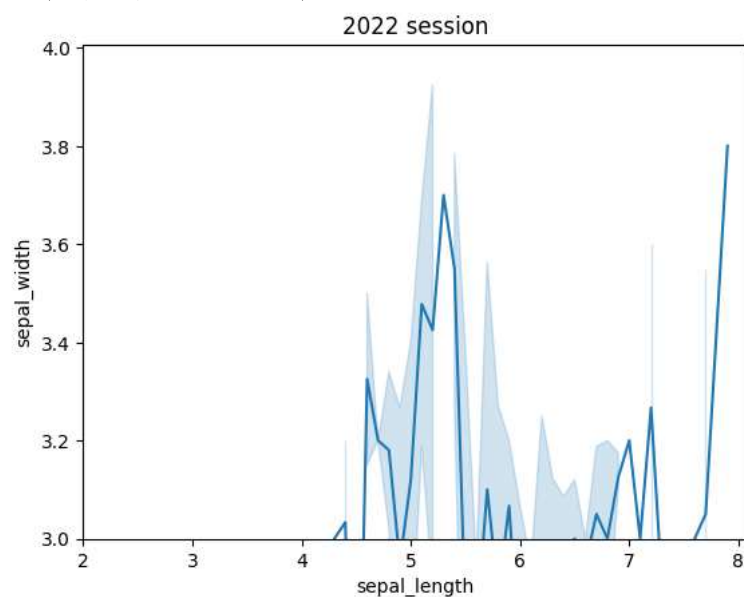
```
iris= sns.load_dataset("iris")
iris.head()
```

	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

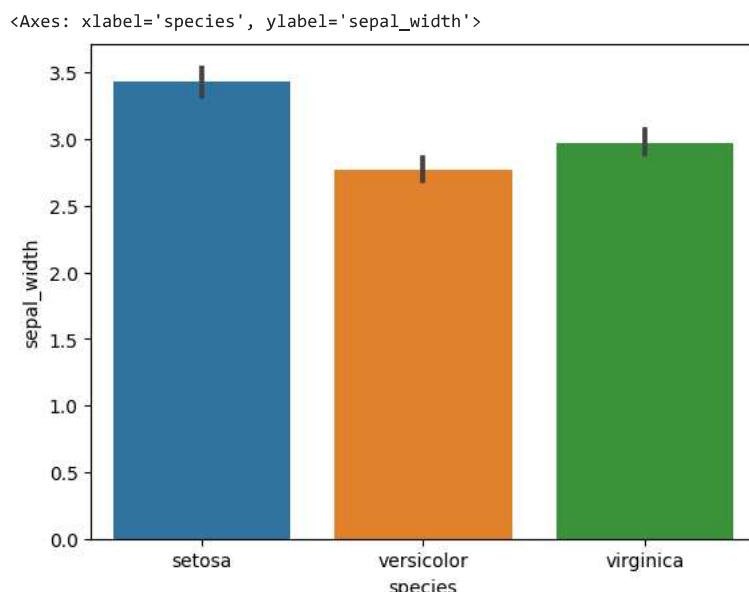
▼ **Step 3: plot a graph**

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

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

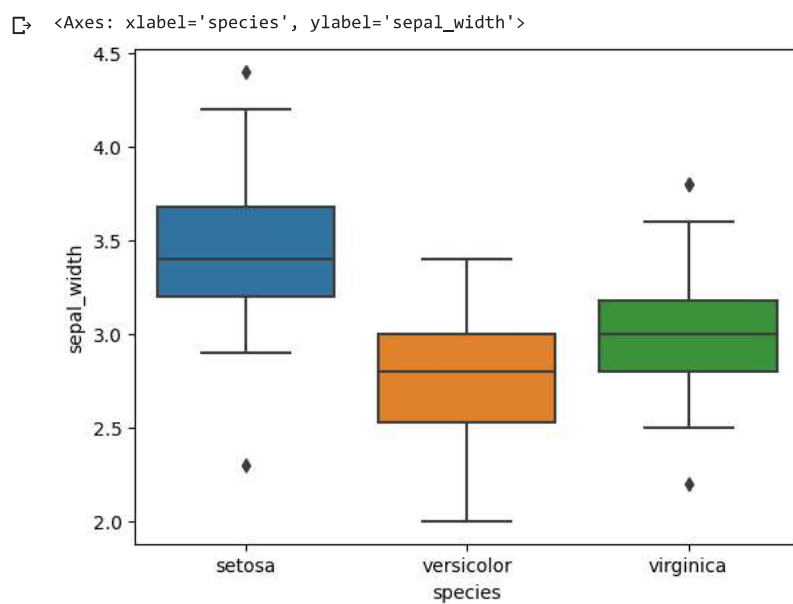
▼ **bar plot**

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



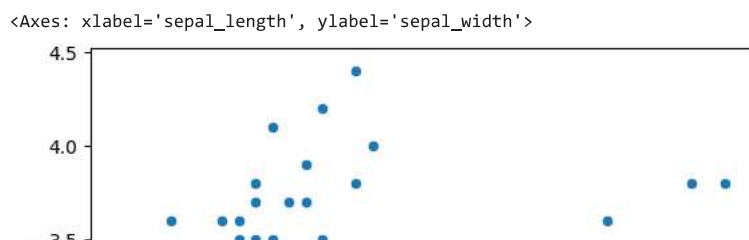
boxplot

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



scatter plot

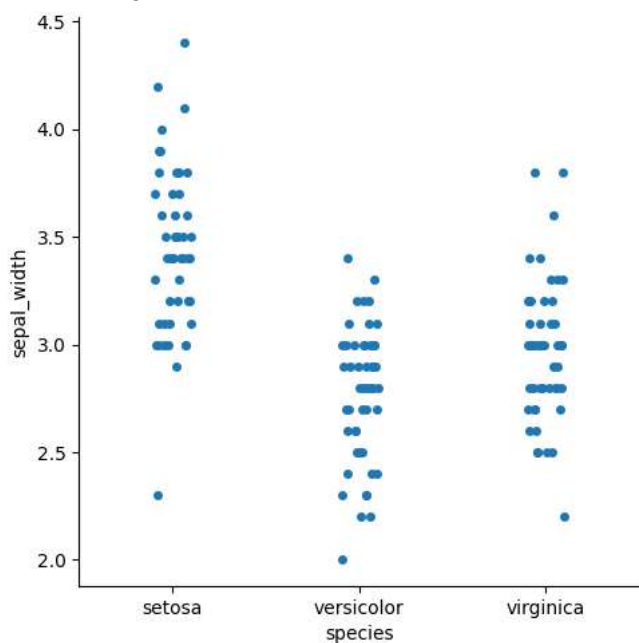
```
sns.scatterplot(x="sepal_length",y="sepal_width",data=iris)
```



▼ catplot

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

```
<seaborn.axisgrid.FacetGrid at 0x7fc8f9acc070>
```

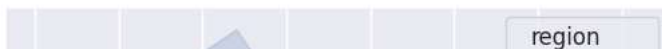


```
import seaborn as sns
sns.set_theme(style="darkgrid")

# Load an example dataset with long-form data
fmri = sns.load_dataset("fmri")

# Plot the responses for different events and regions
sns.lineplot(x="timepoint", y="signal",
             hue="region", style="event",
             data=fmri)
```

<Axes: xlabel='timepoint', ylabel='signal'>

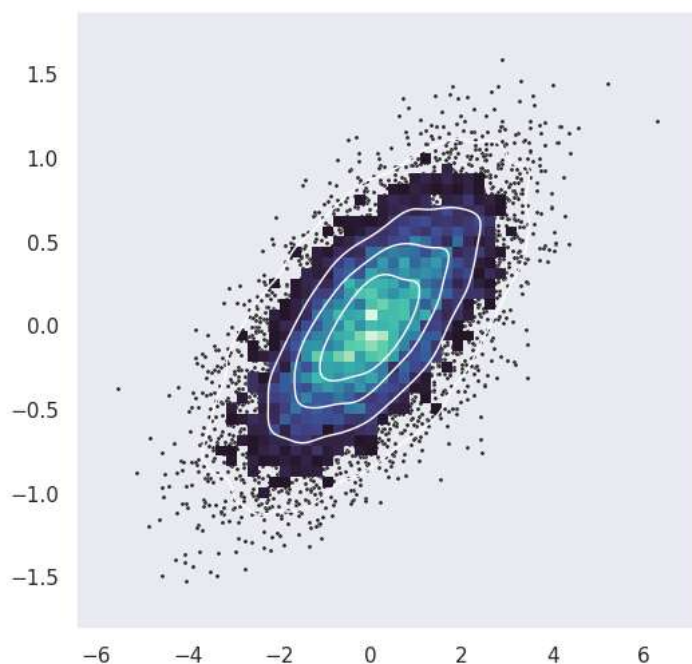


```
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
sns.set_theme(style="dark")

# Simulate data from a bivariate Gaussian
n = 10000
mean = [0, 0]
cov = [(2, .4), (.4, .2)]
rng = np.random.RandomState(0)
x, y = rng.multivariate_normal(mean, cov, n).T

# Draw a combo histogram and scatterplot with density contours
f, ax = plt.subplots(figsize=(6, 6))
sns.scatterplot(x=x, y=y, s=5, color=".15")
sns.histplot(x=x, y=y, bins=50, pthresh=.1, cmap="mako")
sns.kdeplot(x=x, y=y, levels=5, color="w", linewidths=1)
```

<Axes: >



▼ heatmap

```
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-21-fd553bdfde69>:7: FutureWarning: In a future version of pandas all arguments of DataFrame.pivot will be keyword-only.  
  flights = flights_long.pivot("month", "year", "passengers")  
<Axes: xlabel='year', ylabel='month'>
```

