

▼ EXPLORATORY DATA ANALYSIS

```
#IMPORT ALL NECESSARY MODULES
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

```
#import data set
df=sns.load_dataset("iris")
```

df

🔗

	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
...	...	...	...	...	...
145	6.7	3.0	5.2	2.3	virginica
146	6.3	2.5	5.0	1.9	virginica
147	6.5	3.0	5.2	2.0	virginica
148	6.2	3.4	5.4	2.3	virginica
149	5.9	3.0	5.1	1.8	virginica

150 rows × 5 columns

▼ SUMMARY OF DATA

```
df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 150 entries, 0 to 149
Data columns (total 5 columns):
#   Column          Non-Null Count  Dtype
---  ---
0   sepal_length    150 non-null   float64
1   sepal_width     150 non-null   float64
2   petal_length    150 non-null   float64
3   petal_width     150 non-null   float64
4   species         150 non-null   object
dtypes: float64(4), object(1)
memory usage: 6.0+ KB
```

▼ statistical summary

```
df.describe()
```

	sepal_length	sepal_width	petal_length	petal_width
count	150.000000	150.000000	150.000000	150.000000
mean	5.843333	3.057333	3.758000	1.199333
std	0.828066	0.435866	1.765298	0.762238
min	4.300000	2.000000	1.000000	0.100000
25%	5.100000	2.800000	1.600000	0.300000
50%	5.800000	3.000000	4.350000	1.300000
75%	6.400000	3.300000	5.100000	1.800000
max	7.900000	4.400000	6.900000	2.500000

```
#see all the columns
df.columns

Index(['sepal_length', 'sepal_width', 'petal_length', 'petal_width',
       'species'],
      dtype='object')
```

check for null values

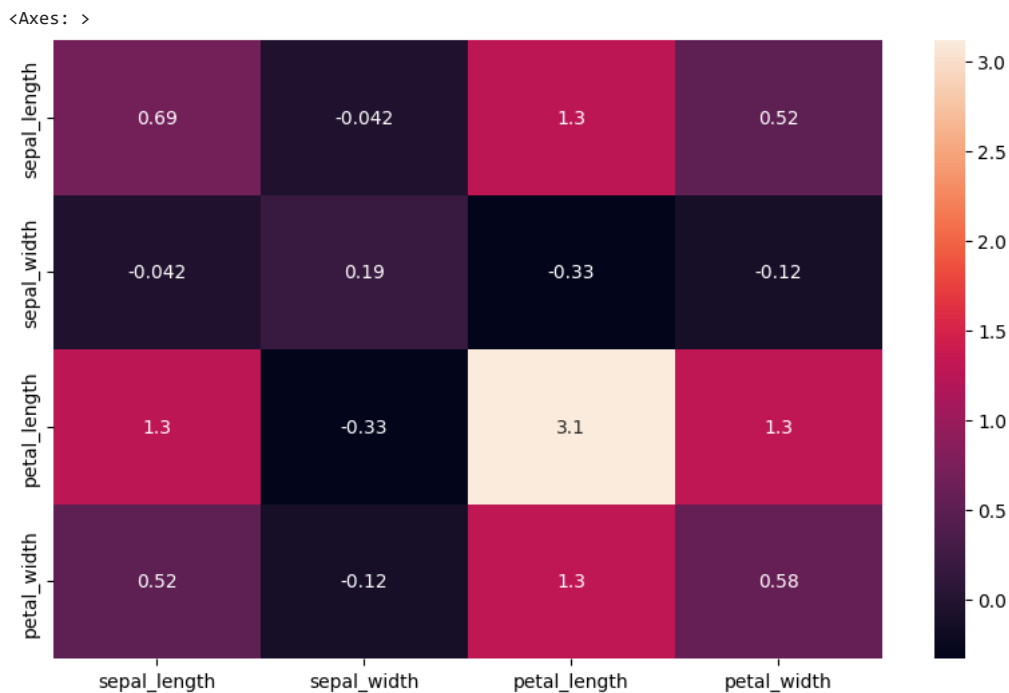
```
df.isnull().sum()

sepal_length    0
sepal_width     0
petal_length    0
petal_width     0
species         0
dtype: int64
```

```
#covariance of data
df.cov()
```

	sepal_length	sepal_width	petal_length	petal_width
sepal_length	0.685694	-0.042434	1.274315	0.516271
sepal_width	-0.042434	0.189979	-0.329656	-0.121639
petal_length	1.274315	-0.329656	3.116278	1.295609
petal_width	0.516271	-0.121639	1.295609	0.581006

```
#visualisation
plt.figure(figsize=(10,6))
sns.heatmap(df.cov(),annot=True)
```



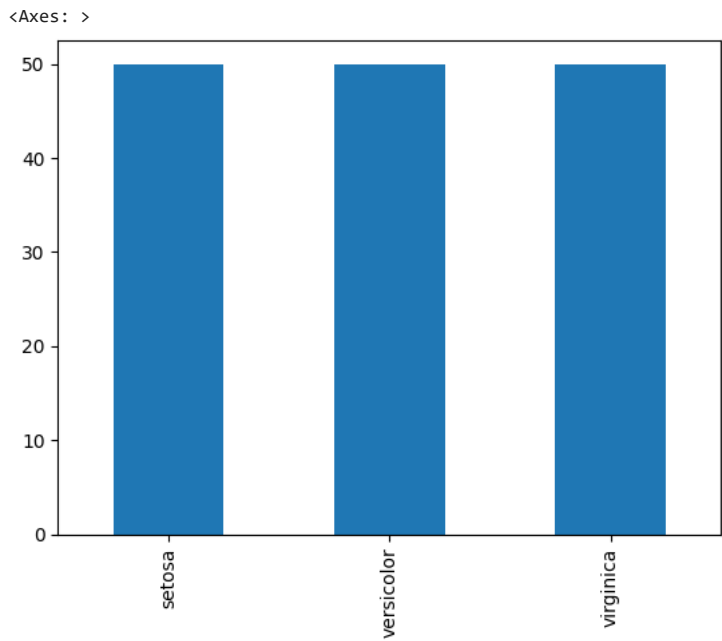
```
#check no of unique species
df.species.unique()

array(['setosa', 'versicolor', 'virginica'], dtype=object)
```

```
#no of species
df.species.value_counts()

setosa      50
versicolor  50
virginica   50
Name: species, dtype: int64
```

```
df.species.value_counts().plot(kind="bar")
```



```
#check for duplicates
df.duplicated()

0      False
1      False
2      False
3      False
4      False
...
145    False
146    False
147    False
148    False
149    False
Length: 150, dtype: bool
```

```
df.shape

(150, 5)
```

```
df[df.duplicated()]
```

	sepal_length	sepal_width	petal_length	petal_width	species
142	5.8	2.7	5.1	1.9	virginica

```
#drop the duplicates
df.drop_duplicates(inplace=True)
```

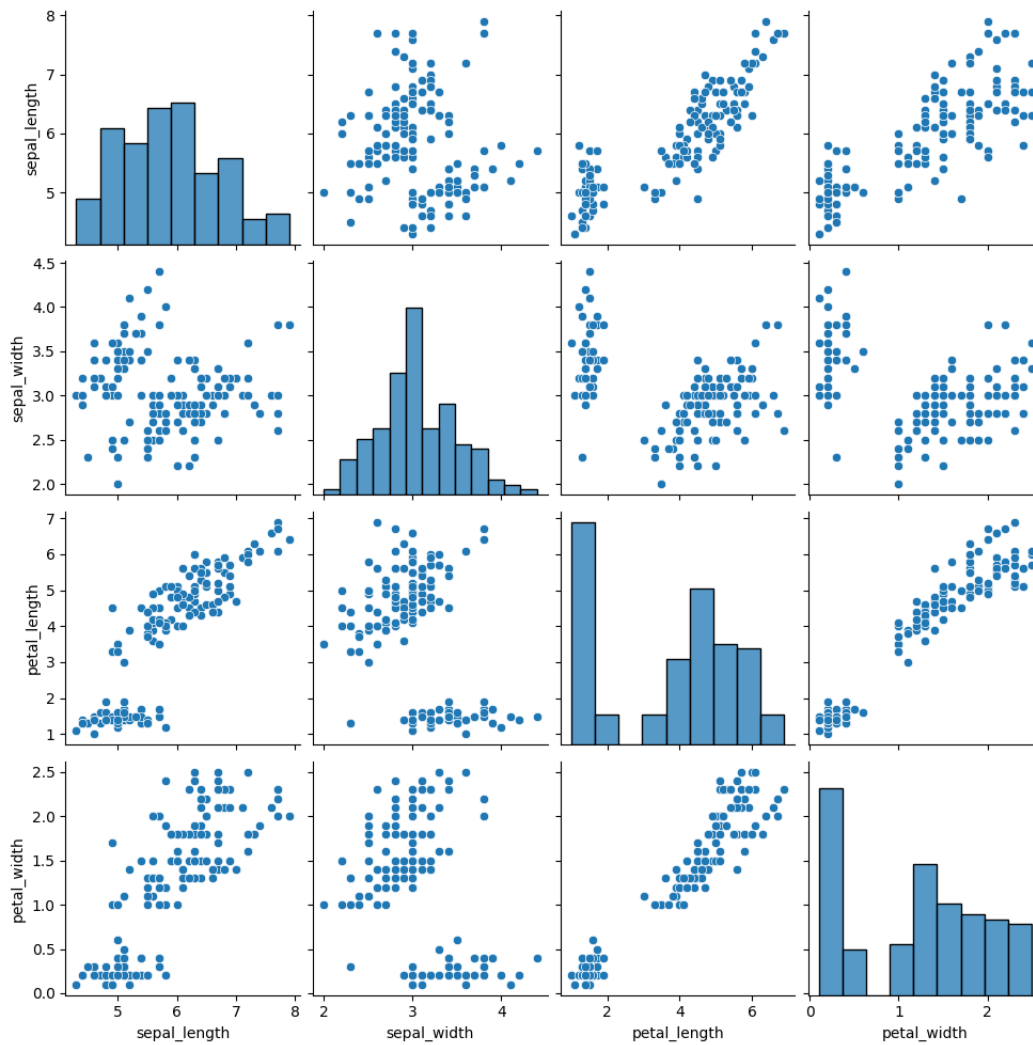
```
df
```

	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
...	...	...	...	...	...
145	6.7	3.0	5.2	2.3	virginica
146	6.3	2.5	5.0	1.9	virginica
147	6.5	3.0	5.2	2.0	virginica
148	6.2	3.4	5.4	2.3	virginica
149	5.9	3.0	5.1	1.8	virginica

149 rows × 5 columns

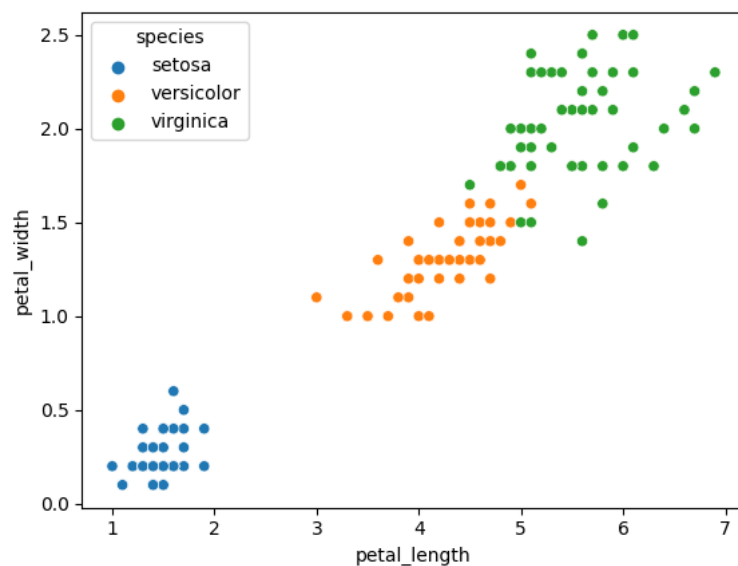
```
sns.pairplot(df)
```

<seaborn.axisgrid.PairGrid at 0x7f80abff2fa0>



```
sns.scatterplot(x=df.petal_length,y=df.petal_width,hue=df["species"])
```

<Axes: xlabel='petal\_length', ylabel='petal\_width'>



---

[Colab paid products](#) - [Cancel contracts here](#)

