

```
import numpy as np
import pandas as pd

iris = pd.read_csv('/content/IRIS.csv')
iris
```

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	Iris-setosa
1	4.9	3.0	1.4	0.2	Iris-setosa
2	4.7	3.2	1.3	0.2	Iris-setosa
3	4.6	3.1	1.5	0.2	Iris-setosa
4	5.0	3.6	1.4	0.2	Iris-setosa
...	...	...	...	...	...
145	6.7	3.0	5.2	2.3	Iris-virginica
146	6.3	2.5	5.0	1.9	Iris-virginica
147	6.5	3.0	5.2	2.0	Iris-virginica
148	6.2	3.4	5.4	2.3	Iris-virginica
149	5.9	3.0	5.1	1.8	Iris-virginica

150 rows × 5 columns

Next steps:

Generate code with iris

View recommended plots

```
iris.head()
```

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

Next steps:

Generate code with iris

View recommended plots

```
iris.columns
```

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

```
columns = len (list(iris))
```

```
columns
```

```
5
```

```
iris.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
```

```
np.unique(iris["species"])

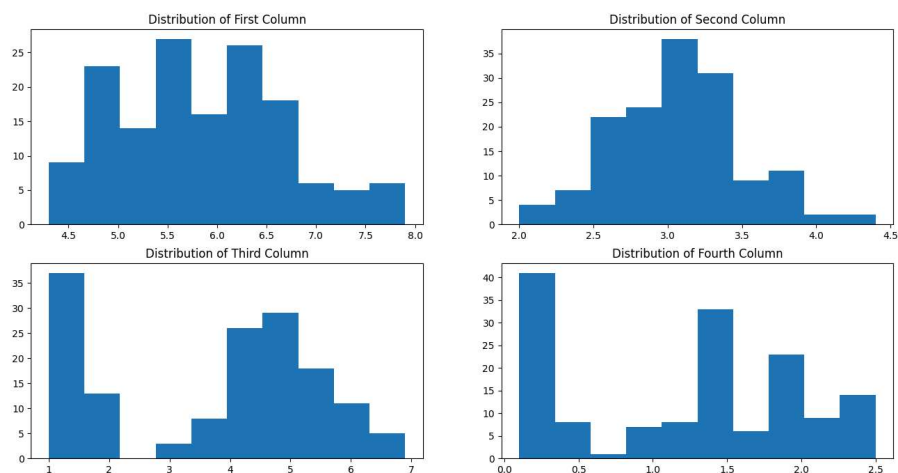
array(['Iris-setosa', 'Iris-versicolor', 'Iris-virginica'], dtype=object)

iris.describe()
```

	sepal_length	sepal_width	petal_length	petal_width
<b>count</b>	150.000000	150.000000	150.000000	150.000000
<b>mean</b>	5.843333	3.054000	3.758667	1.198667
<b>std</b>	0.828066	0.433594	1.764420	0.763161
<b>min</b>	4.300000	2.000000	1.000000	0.100000
<b>25%</b>	5.100000	2.800000	1.600000	0.300000
<b>50%</b>	5.800000	3.000000	4.350000	1.300000
<b>75%</b>	6.400000	3.300000	5.100000	1.800000
<b>max</b>	7.900000	4.400000	6.900000	2.500000

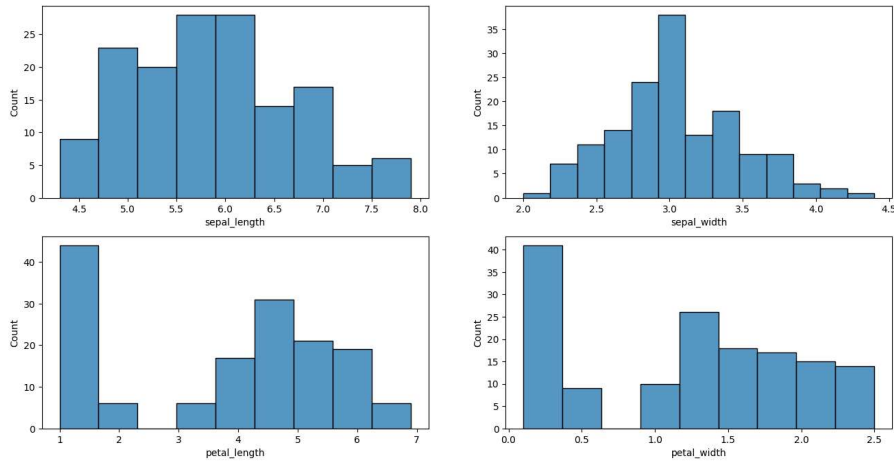
```
import seaborn as sns
import matplotlib
import matplotlib.pyplot as plt
%matplotlib inline
```

```
fig, axes = plt.subplots(2, 2, figsize = (16, 8))
axes[0,0].set_title("Distribution of First Column")
axes[0,0].hist(iris["sepal_length"]);
axes[0,1].set_title("Distribution of Second Column")
axes[0,1].hist(iris["sepal_width"]);
axes[1,0].set_title("Distribution of Third Column")
axes[1,0].hist(iris["petal_length"]);
axes[1,1].set_title("Distribution of Fourth Column")
axes[1,1].hist(iris["petal_width"]);
```



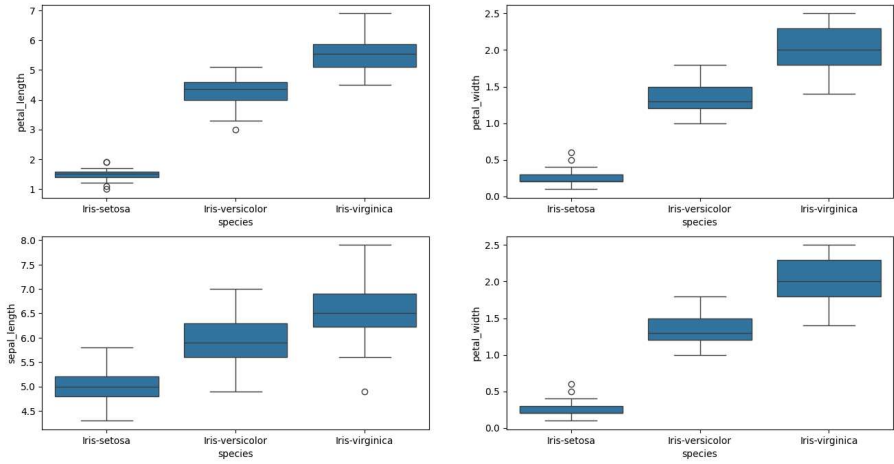
```
import matplotlib.pyplot as plt
fig, axes = plt.subplots(2, 2, figsize = (16, 8))
sns.histplot (iris['sepal_length'], ax = axes[0,0])
sns.histplot (iris['sepal_width'], ax = axes[0,1])
sns.histplot (iris['petal_length'], ax = axes[1,0])
sns.histplot (iris['petal_width'], ax = axes[1,1])
```

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



```
import matplotlib.pyplot as plt
fig, axes = plt.subplots(2, 2, figsize = (16, 8))
sns.boxplot(y='petal_length', x='species', data = iris,ax = axes[0,0])
sns.boxplot(y='petal_width', x='species', data = iris,ax = axes[0,1])
sns.boxplot(y='sepal_length', x='species', data = iris,ax = axes[1,0])
sns.boxplot(y='petal_width', x='species', data = iris,ax = axes[1,1])
```

```
<Axes: xlabel='species', ylabel='petal_width'>
```



Start coding or [generate](#) with AI.