Question 5 IRIS

Python for Data Science - Perform Data Visualization on Iris Dataset

a)Load the Titanic dataset into one of the data structures (NumPy or Pandas).

b)Display header rows and description of the loaded dataset.

c) Clean the data if applicable

d) Find the average petal width of each category of IRIS Species

e) Data Visualization for:

(i) How many flowers of each species exists for each value of sepal width

(ii) How many flowers are there whose petal width is <1, between 1 to 2 and >2

(iii) Tally the Iris-Versicolour and Iris-Virginica species according to the value of Sepal Width

Click here to download dataset

In [36]:
```
#numpy - Deals multi-dimensional arrays and matrices
#seaborn - Deals with data visualization
#matplotlib - Plotting; pyplot-interactive plotting
#pandas - data structures and data analysis tools
import seaborn as sns
import pandas as pd
import matplotlib.pyplot as plot
```

In [45]:
```
#Import csv file into variable (dataframe)
iris_df = pd.read_csv('iris.csv')
iris_df.head()
```

Out[45]:

	Sepal_Length	Sepal_Width	Petal_Length	Petal_Width	Class
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

In [38]:
```
#print info about dataframe
print("This is info() output\n")
print(iris_df.info())
print("\nThis is describe() output\n")
print(iris_df.describe())
```

```
This is info() output

<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   Class          150 non-null    object 
dtypes: float64(4), object(1)
memory usage: 6.0+ KB
None

This is describer() output

       Sepal_Length   Sepal_Width   Petal_Length   Petal_Width
count    150.000000    150.000000     150.000000    150.000000
mean       5.843333      3.054000       3.758667      1.198667
std        0.828066      0.433594       1.764420      0.763161
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
```

In [39]:
```
#drop sepal_length
iris_df.drop(['Sepal_Length'],axis=1,inplace=True)
iris_df.head()
```

Out[39]:

	Sepal_Width	Petal_Length	Petal_Width	Class
0	3.5	1.4	0.2	Iris-setosa
1	3.0	1.4	0.2	Iris-setosa
2	3.2	1.3	0.2	Iris-setosa
3	3.1	1.5	0.2	Iris-setosa
4	3.6	1.4	0.2	Iris-setosa

In [63]:
```
iris_df.groupby('Class',as_index=False)[' Petal_Width'].mean()
```

Out[63]:

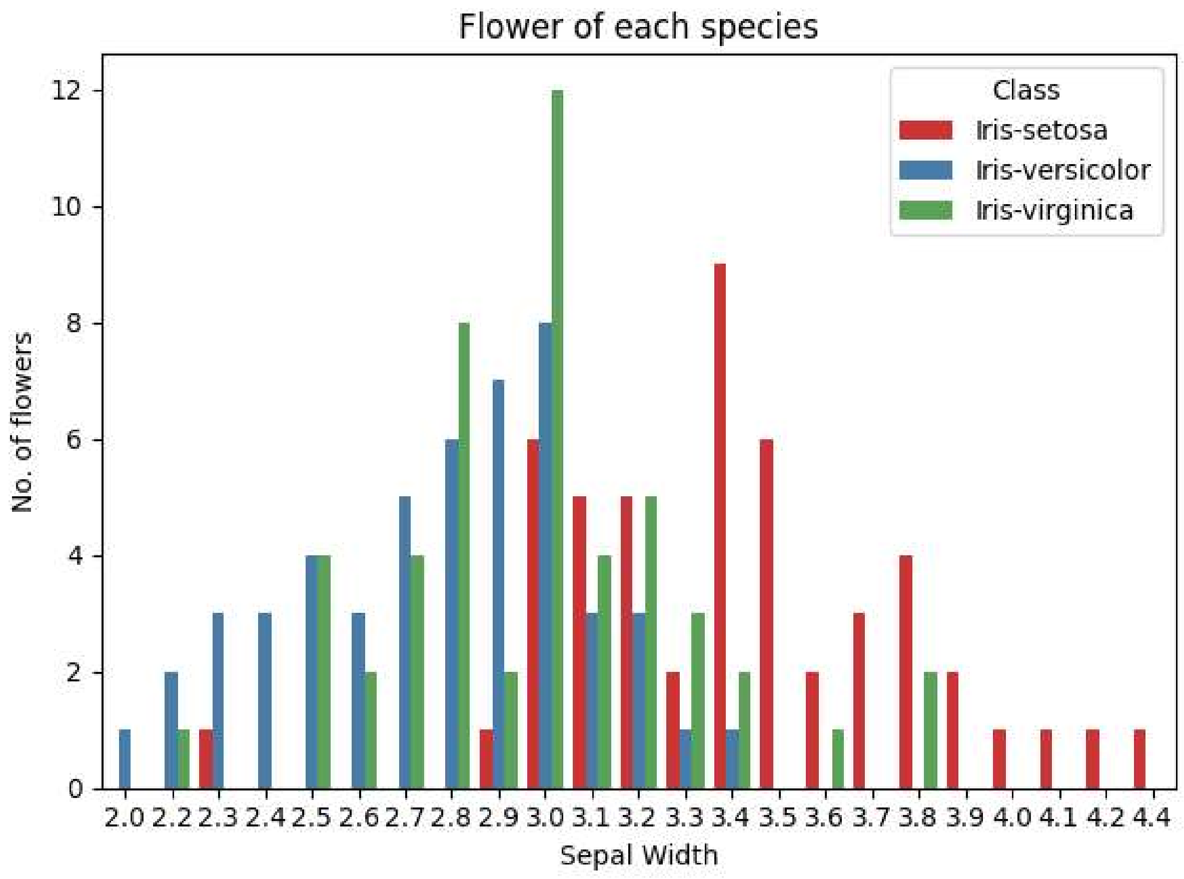
	Class	Petal_Width
0	Iris-setosa	0.244
1	Iris-versicolor	1.326
2	Iris-virginica	2.026

In [40]:
```
# we increase the size of output graph
plot.figure(figsize=[12,6])
```

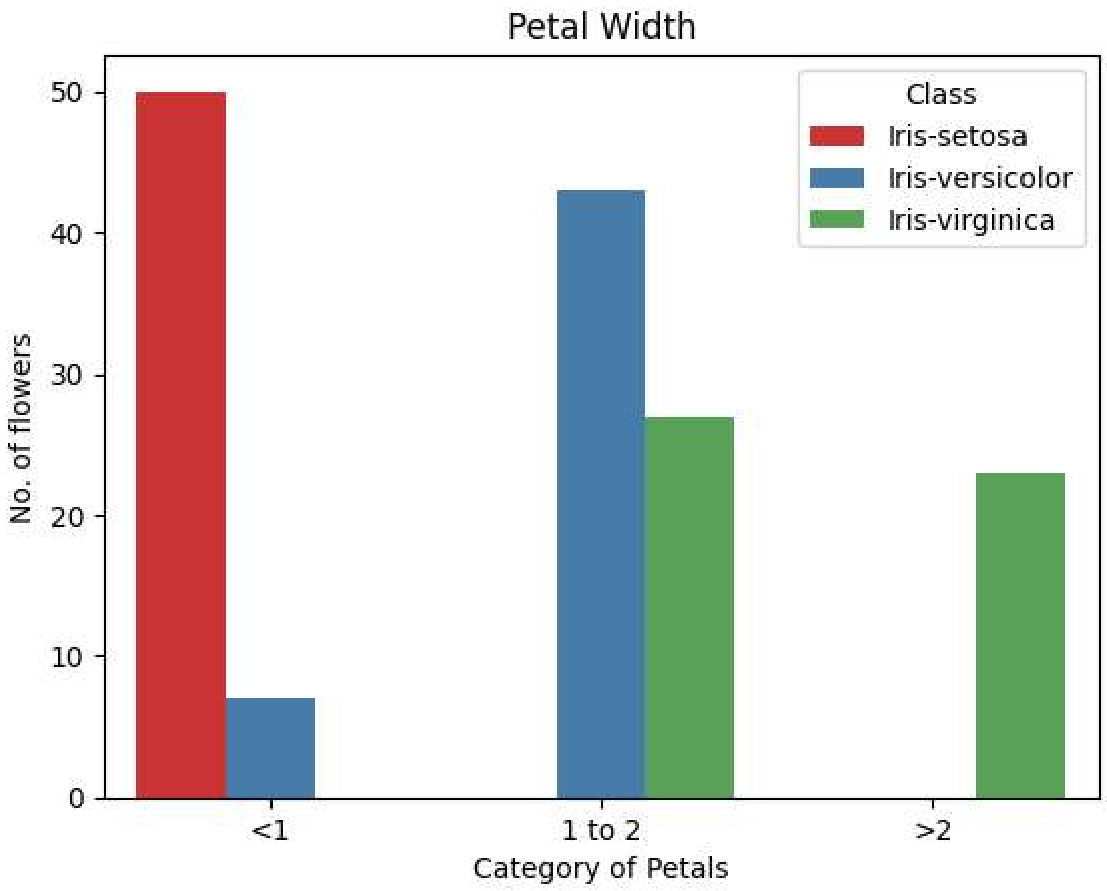
```
Out[40]: <Figure size 1200x600 with 0 Axes>

         <Figure size 1200x600 with 0 Axes>
```

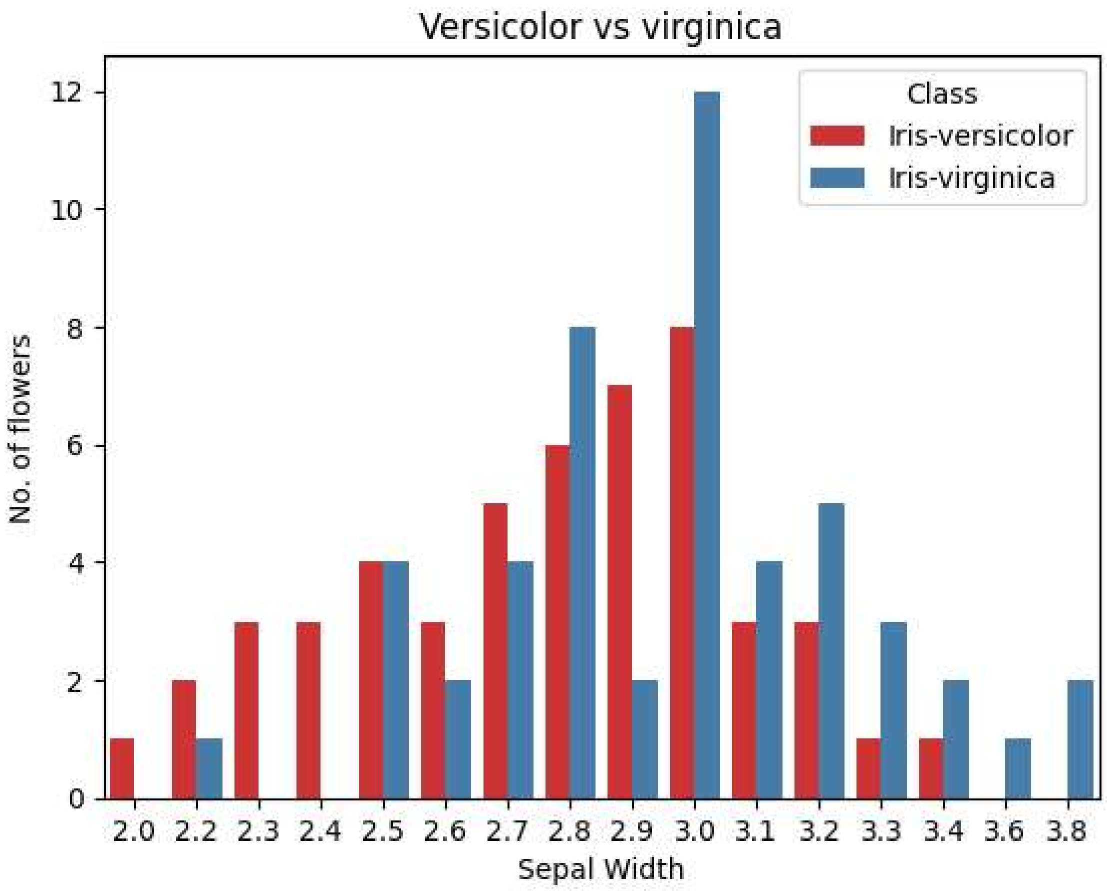
```
In [52]: #plot graph of class vs sepalwidth
         ax=sns.countplot(data=iris_df,x=' Sepal_Width',hue='Class',palette='Set1')
         ax.set(title='Flower of each species',xlabel='Sepal Width',ylabel='No. of flowers')
         plot.tight_layout()
         plot.show()
```



```
In [64]: #Cut petal width accoding to interval and give labels from categories
         interval = (0,1,2,4)
         category = ['<1','1 to 2','>2']
         iris_df['Petal_Catg'] = pd.cut(iris_df[' Petal_Width'],interval,labels=category)
         ax = sns.countplot(data = iris_df,x = 'Petal_Catg',hue='Class',palette='Set1')
         ax.set(title='Petal Width',xlabel='Category of Petals',ylabel='No. of flowers')
         plot.show()
```



```
In [68]: ax = sns.countplot(data = iris_df[iris_df['Class']!='Iris-setosa'],x = ' Sepal_Widt
         ax.set(title='Versicolor vs virginica',xlabel='Sepal Width',ylabel='No. of flowers'
         plot.show()
```


Versicolor vs virginica
Class
Iris-versicolor
Iris-virginica
No. of flowers
Sepal Width
0
2
4
6
8
10
12
2.0
2.2
2.3
2.4
2.5
2.6
2.7
2.8
2.9
3.0
3.1
3.2
3.3
3.4
3.6
3.8