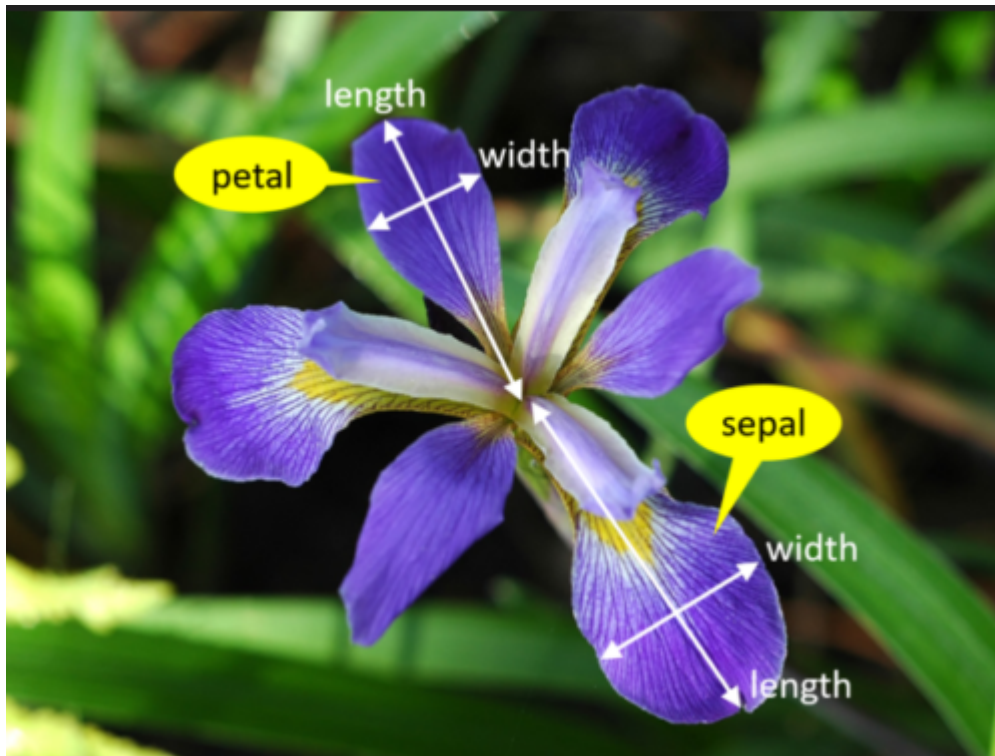


IRIS DATASET VISUALIZATION(SEABORN,MATPLOTLIB)



```
In [3]: import numpy as np  
import pandas as pd
```

```
In [4]: import seaborn as sns  
import matplotlib.pyplot as plt  
import warnings  
warnings.filterwarnings('ignore')
```

```
In [5]: iris=pd.read_csv(r'C:\Users\User\Downloads\Iris.csv')  
iris
```

Out[5]:

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	1	5.1	3.5	1.4	0.2	Iris-setosa
1	2	4.9	3.0	1.4	0.2	Iris-setosa
2	3	4.7	3.2	1.3	0.2	Iris-setosa
3	4	4.6	3.1	1.5	0.2	Iris-setosa
4	5	5.0	3.6	1.4	0.2	Iris-setosa
...
145	146	6.7	3.0	5.2	2.3	Iris-virginica
146	147	6.3	2.5	5.0	1.9	Iris-virginica
147	148	6.5	3.0	5.2	2.0	Iris-virginica
148	149	6.2	3.4	5.4	2.3	Iris-virginica
149	150	5.9	3.0	5.1	1.8	Iris-virginica

150 rows × 6 columns

In [6]: `iris.shape`

Out[6]: (150, 6)

In [7]: `len('iris')`

Out[7]: 4

In [8]: `print(type(iris))`

<class 'pandas.core.frame.DataFrame'>

In [9]: `iris.columns`Out[9]: Index(['Id', 'SepalLengthCm', 'SepalWidthCm', 'PetalLengthCm', 'PetalWidthCm',
'Species'],
dtype='object')In [10]: `iris.info()`

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 150 entries, 0 to 149
Data columns (total 6 columns):
#   Column                Non-Null Count  Dtype  
---  -
0   Id                     150 non-null   int64  
1   SepalLengthCm         150 non-null   float64
2   SepalWidthCm          150 non-null   float64
3   PetalLengthCm         150 non-null   float64
4   PetalWidthCm          150 non-null   float64
5   Species               150 non-null   object  
dtypes: float64(4), int64(1), object(1)
memory usage: 7.2+ KB
```

```
In [11]: iris.isna()
```

```
Out[11]:
```

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	False	False	False	False	False	False
1	False	False	False	False	False	False
2	False	False	False	False	False	False
3	False	False	False	False	False	False
4	False	False	False	False	False	False
...
145	False	False	False	False	False	False
146	False	False	False	False	False	False
147	False	False	False	False	False	False
148	False	False	False	False	False	False
149	False	False	False	False	False	False

150 rows × 6 columns

```
In [12]: iris.isna().sum()
```

```
Out[12]: Id                0
SepalLengthCm            0
SepalWidthCm             0
PetalLengthCm            0
PetalWidthCm             0
Species                  0
dtype: int64
```

```
In [13]: iris.describe()
```

Out[13]:

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm
count	150.000000	150.000000	150.000000	150.000000	150.000000
mean	75.500000	5.843333	3.054000	3.758667	1.198667
std	43.445368	0.828066	0.433594	1.764420	0.763161
min	1.000000	4.300000	2.000000	1.000000	0.100000
25%	38.250000	5.100000	2.800000	1.600000	0.300000
50%	75.500000	5.800000	3.000000	4.350000	1.300000
75%	112.750000	6.400000	3.300000	5.100000	1.800000
max	150.000000	7.900000	4.400000	6.900000	2.500000

In [14]: iris.head()

Out[14]:

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	1	5.1	3.5	1.4	0.2	Iris-setosa
1	2	4.9	3.0	1.4	0.2	Iris-setosa
2	3	4.7	3.2	1.3	0.2	Iris-setosa
3	4	4.6	3.1	1.5	0.2	Iris-setosa
4	5	5.0	3.6	1.4	0.2	Iris-setosa

In [15]: iris.tail()

Out[15]:

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
145	146	6.7	3.0	5.2	2.3	Iris-virginica
146	147	6.3	2.5	5.0	1.9	Iris-virginica
147	148	6.5	3.0	5.2	2.0	Iris-virginica
148	149	6.2	3.4	5.4	2.3	Iris-virginica
149	150	5.9	3.0	5.1	1.8	Iris-virginica

In [16]: iris.drop('Id',axis=1,inplace=True)

In [17]: iris.head()

```
Out[17]:
```

	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	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

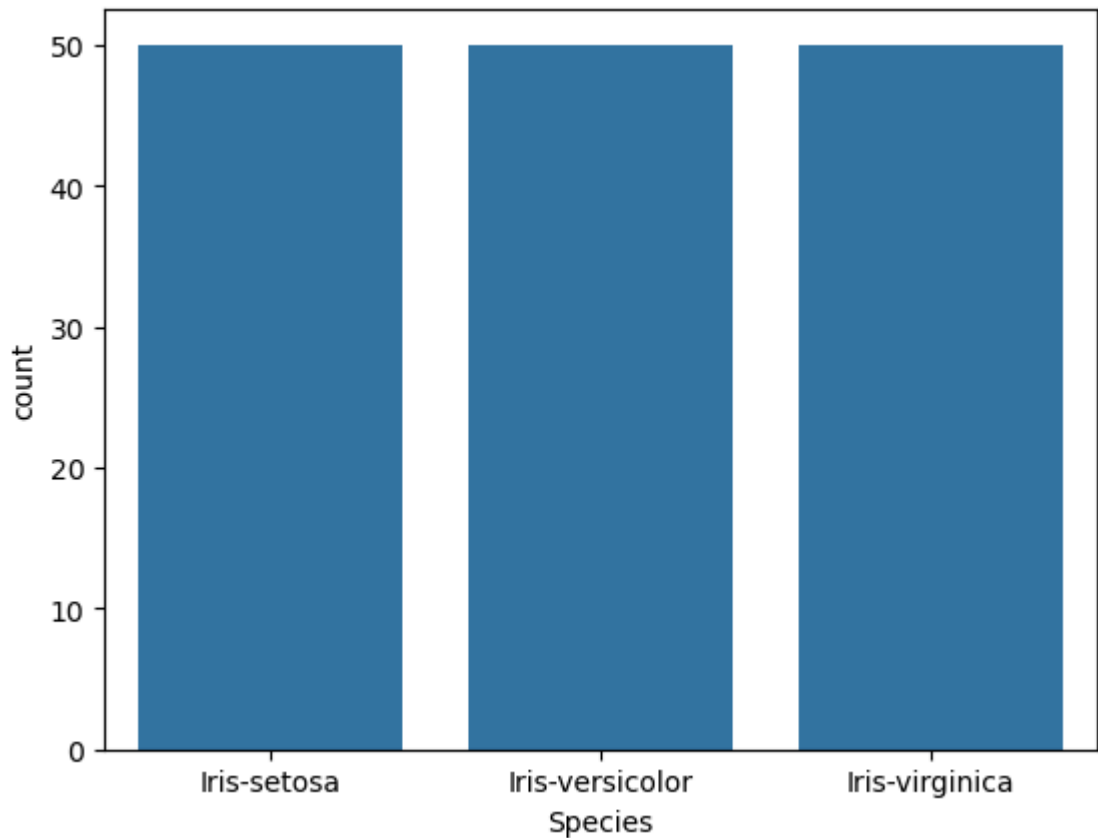
```
In [18]: iris.info()

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

```
In [19]: iris['Species'].value_counts()
```

```
Out[19]: Species
Iris-setosa      50
Iris-versicolor  50
Iris-virginica   50
Name: count, dtype: int64
```

```
In [20]: sns.countplot(x='Species',data=iris)
plt.show()
```

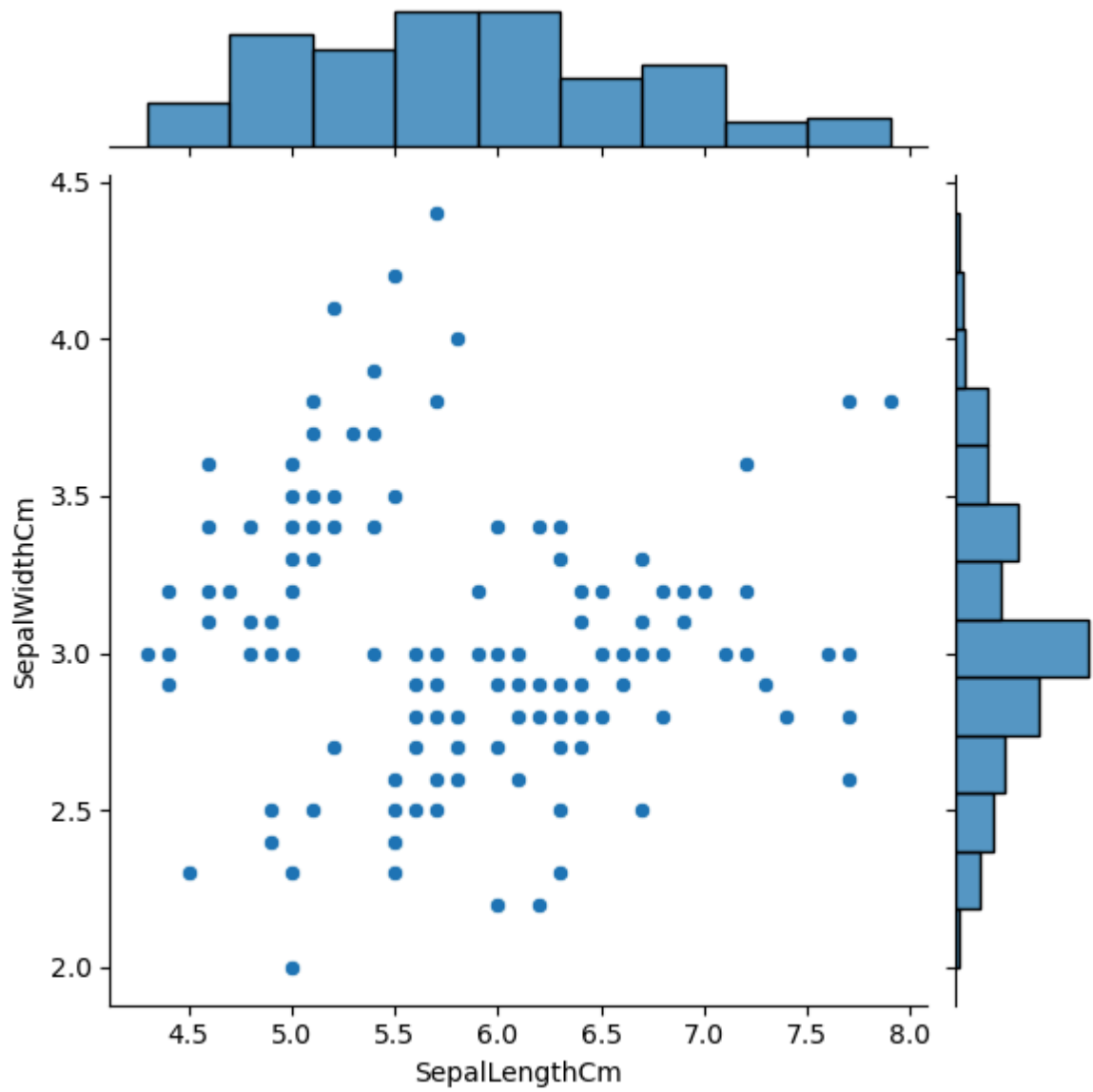


```
In [21]: iris.head()
```

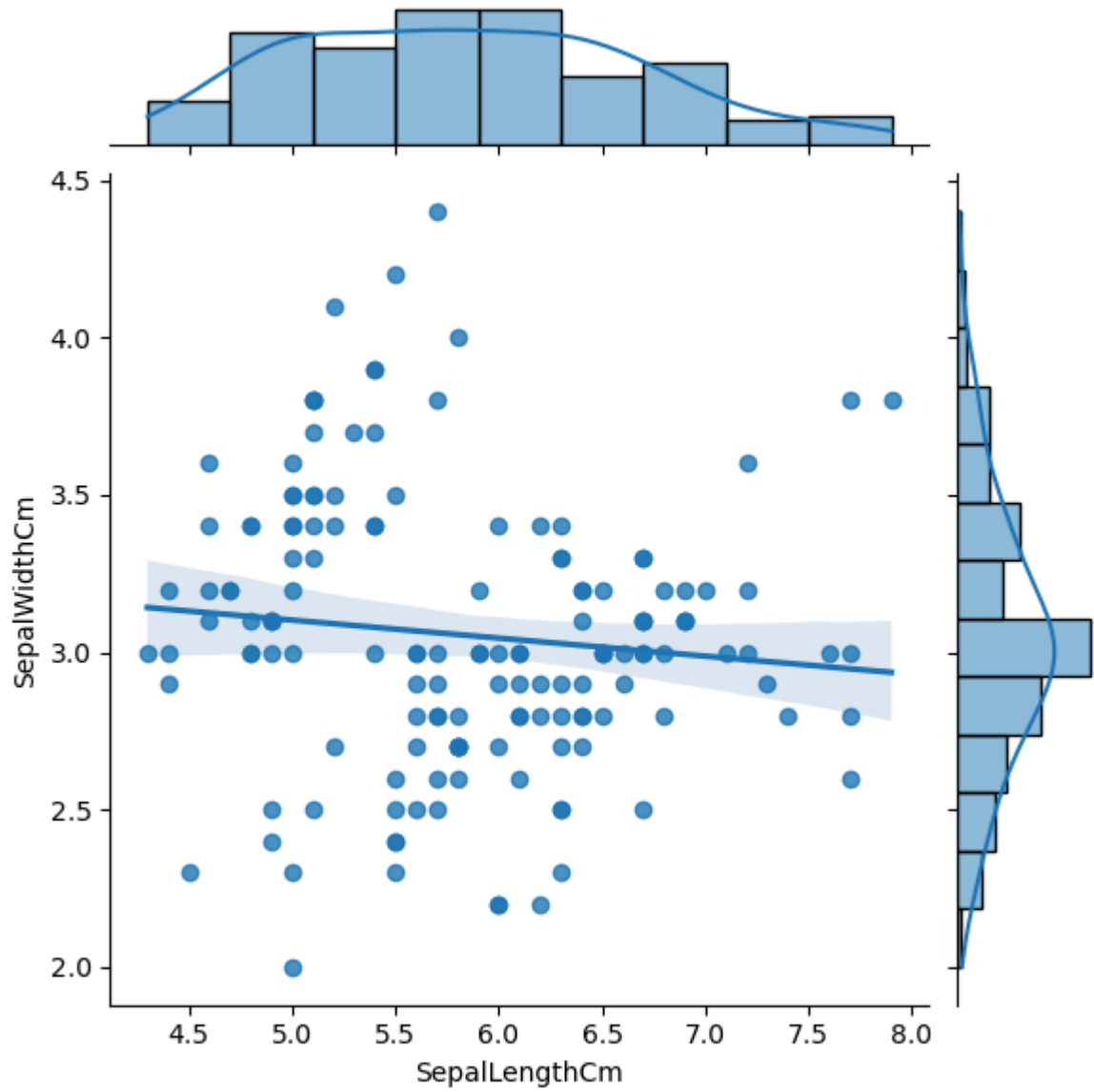
```
Out[21]:
```

	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	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

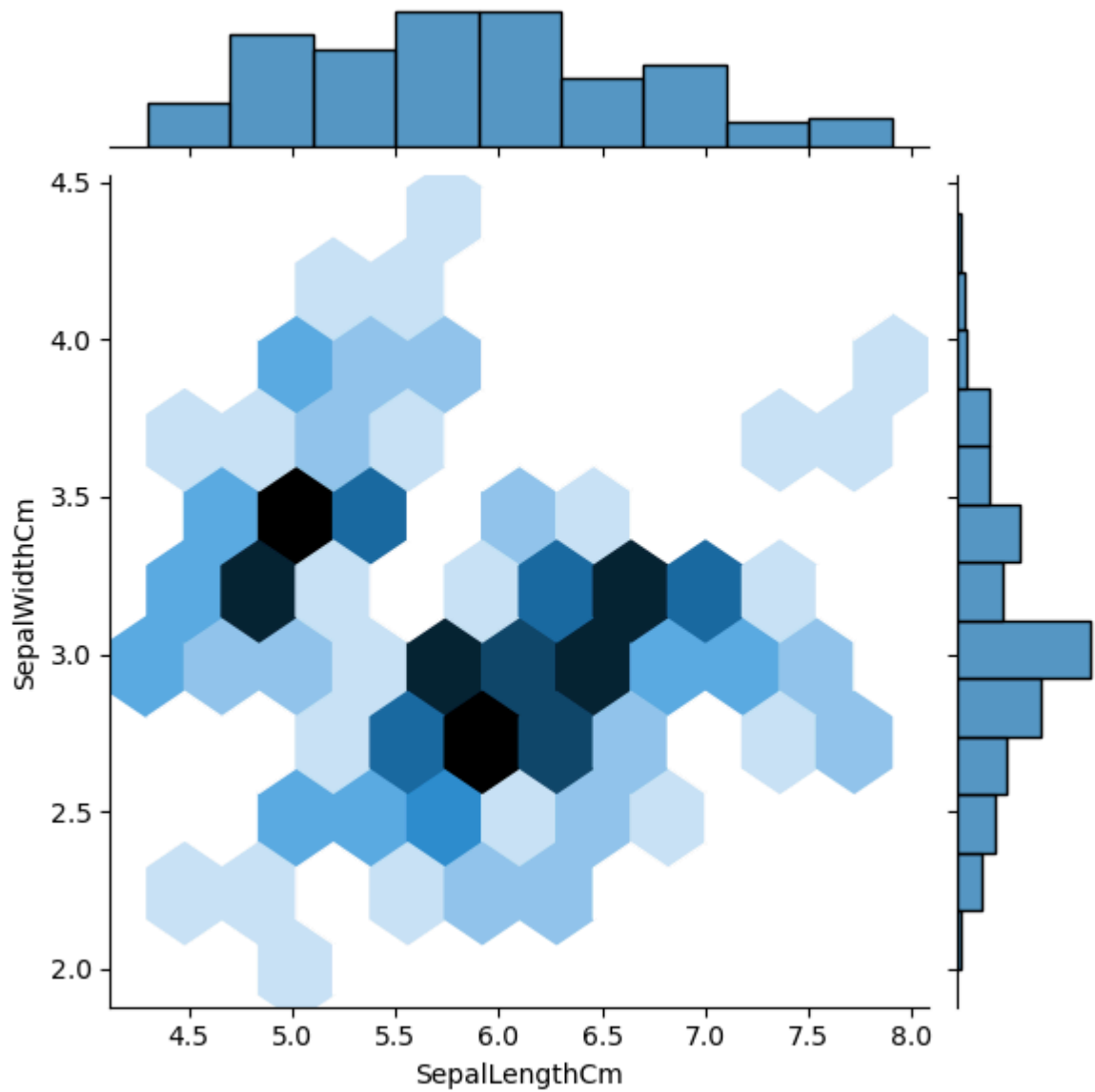
```
In [22]: fig=sns.jointplot(x='SepalLengthCm',y='SepalWidthCm',data=iris)
```



```
In [23]: fig=sns.jointplot(x='SepalLengthCm',y='SepalWidthCm',data=iris,kind='reg')
```



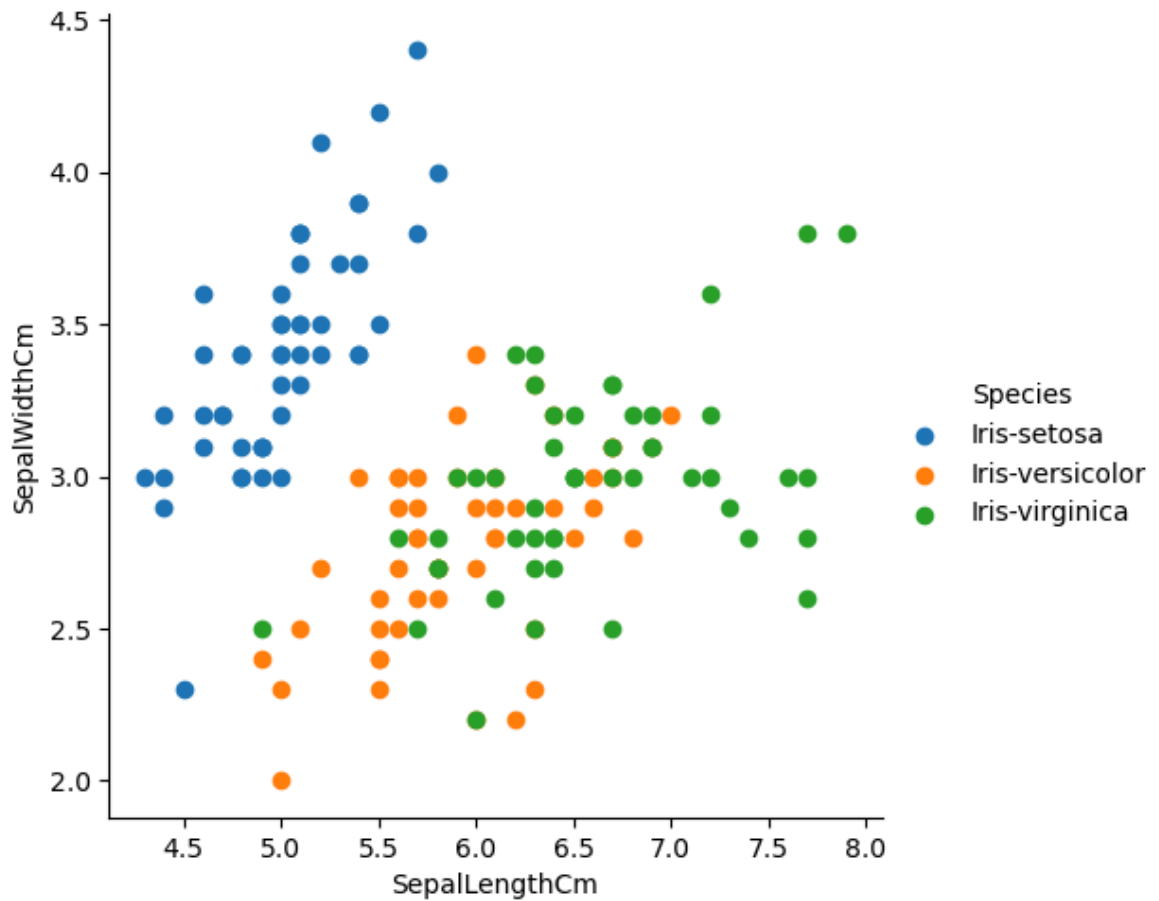
```
In [24]: fig=sns.jointplot(x='SepalLengthCm',y='SepalWidthCm',data=iris, kind='hex')
```

```
In [25]: import matplotlib.pyplot as plt
%matplotlib inline

sns.FacetGrid(iris,hue='Species',height=5)\
    .map(plt.scatter,'SepalLengthCm','SepalWidthCm')\
    .add_legend()
```

```
Out[25]: <seaborn.axisgrid.FacetGrid at 0x263d3e5fa40>
```

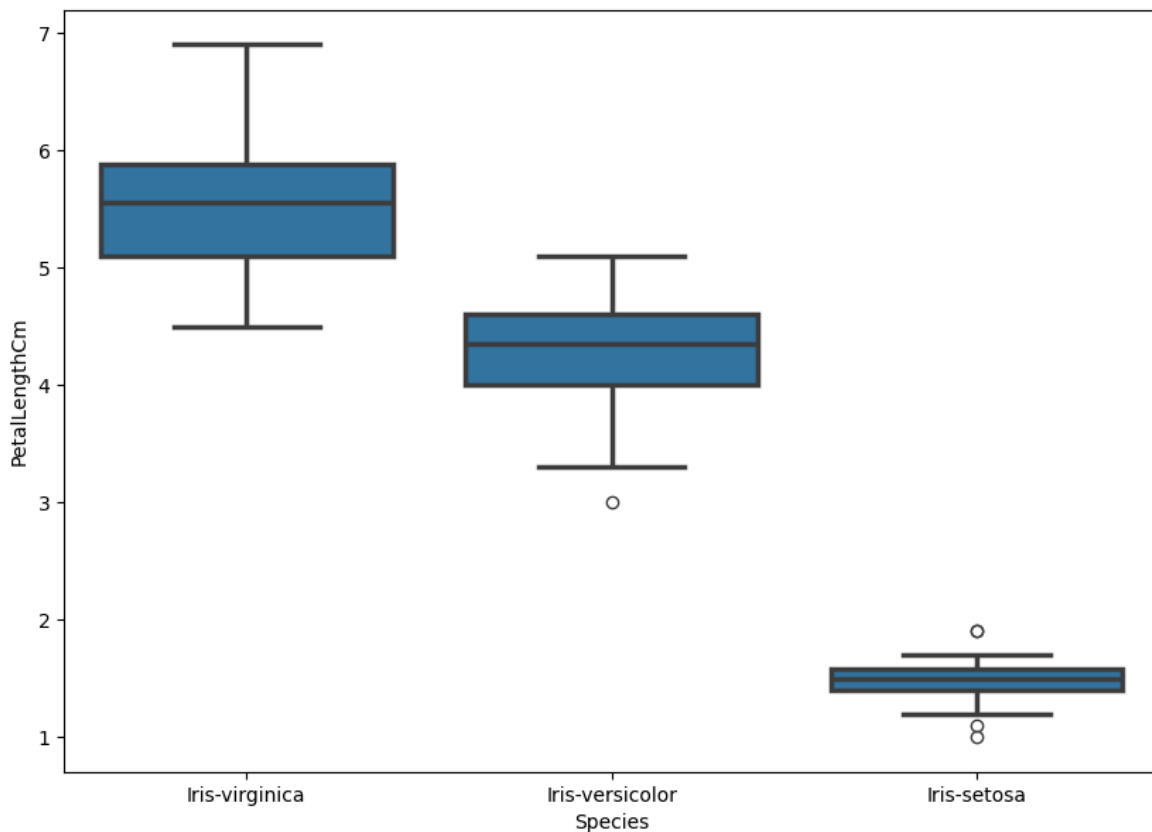


```
In [26]: iris.head()
```

```
Out[26]:
```

	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	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

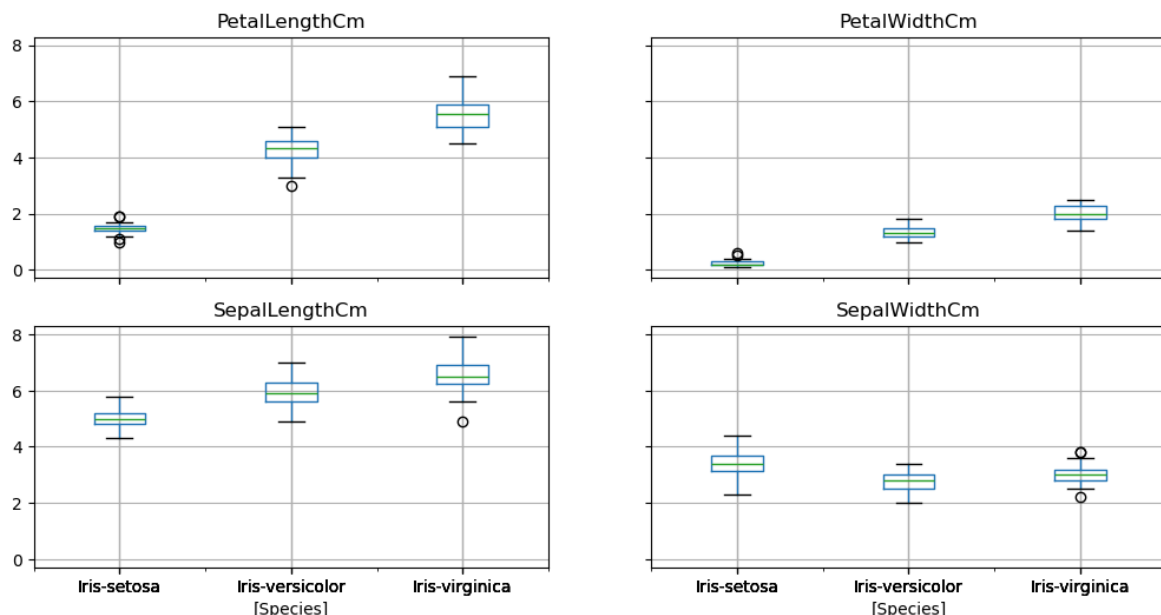
```
In [27]: fig=plt.gcf()
fig.set_size_inches(10,7)
fig=sns.boxplot(x='Species',y='PetalLengthCm',data=iris,
                order=['Iris-virginica','Iris-versicolor','Iris-setosa'],
                linewidth=2.5,orient='v',dodge=False)
```



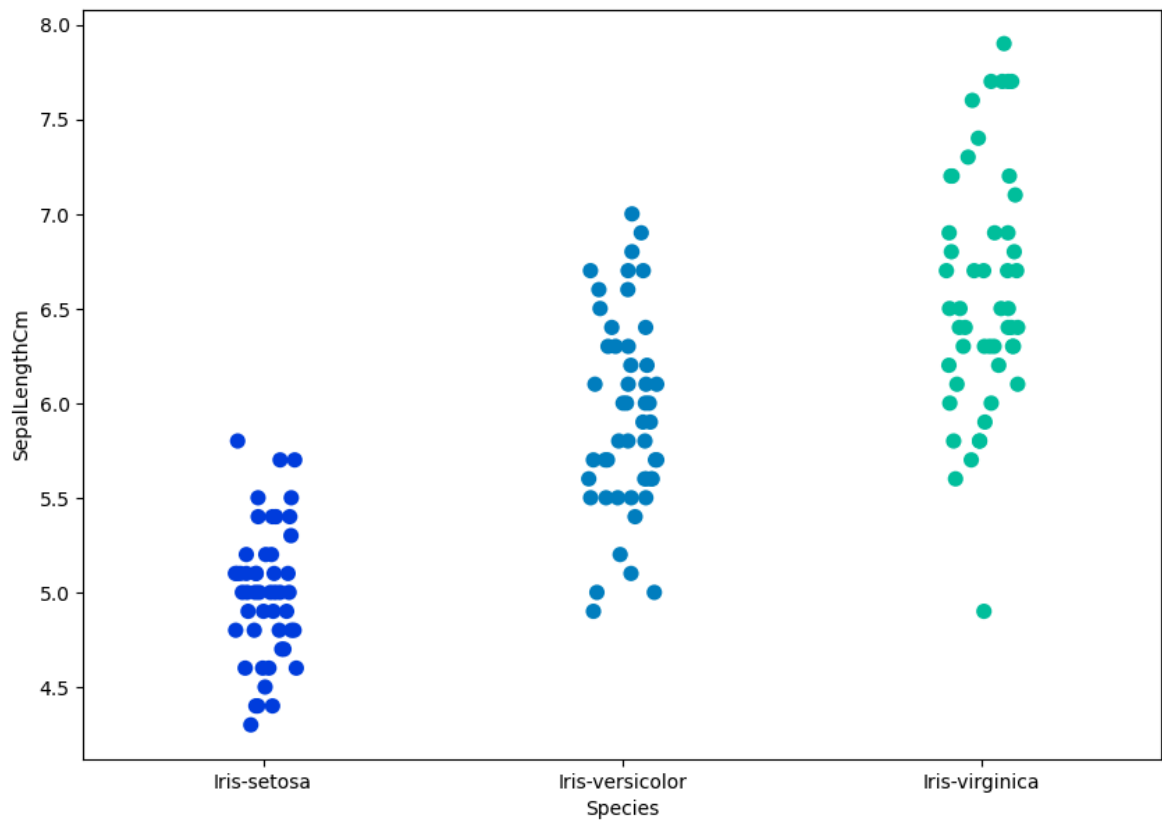
```
In [28]: iris.boxplot(by='Species',figsize=(12,6))
```

```
Out[28]: array([[<Axes: title={'center': 'PetalLengthCm'}, xlabel='[Species]>',
  <Axes: title={'center': 'PetalWidthCm'}, xlabel='[Species]>',
  <Axes: title={'center': 'SepalLengthCm'}, xlabel='[Species]>',
  <Axes: title={'center': 'SepalWidthCm'}, xlabel='[Species]>']],
  dtype=object)
```

Boxplot grouped by Species

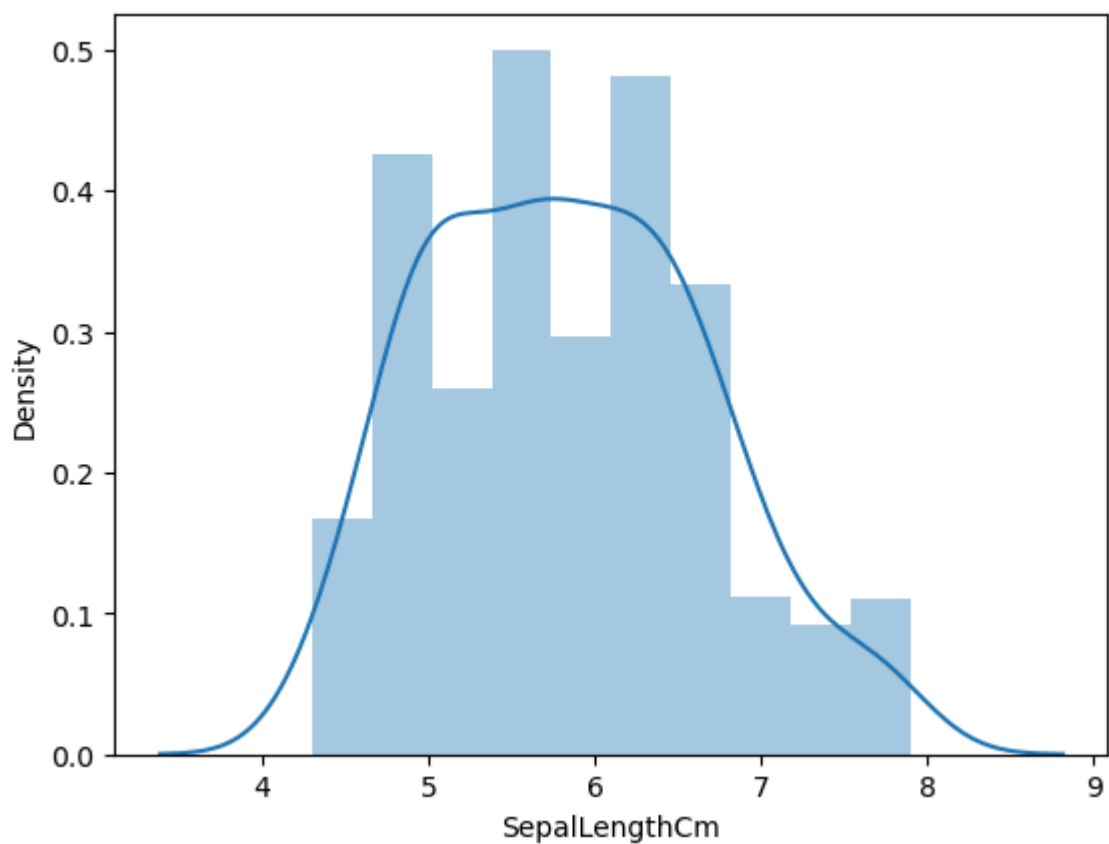


```
In [29]: fig=plt.gcf()
fig.set_size_inches(10,7)
fig=sns.stripplot(x='Species',y='SepalLengthCm',data=iris,
  jitter=True,edgecolor='gray',size=8,
  palette='winter',orient='v')
```



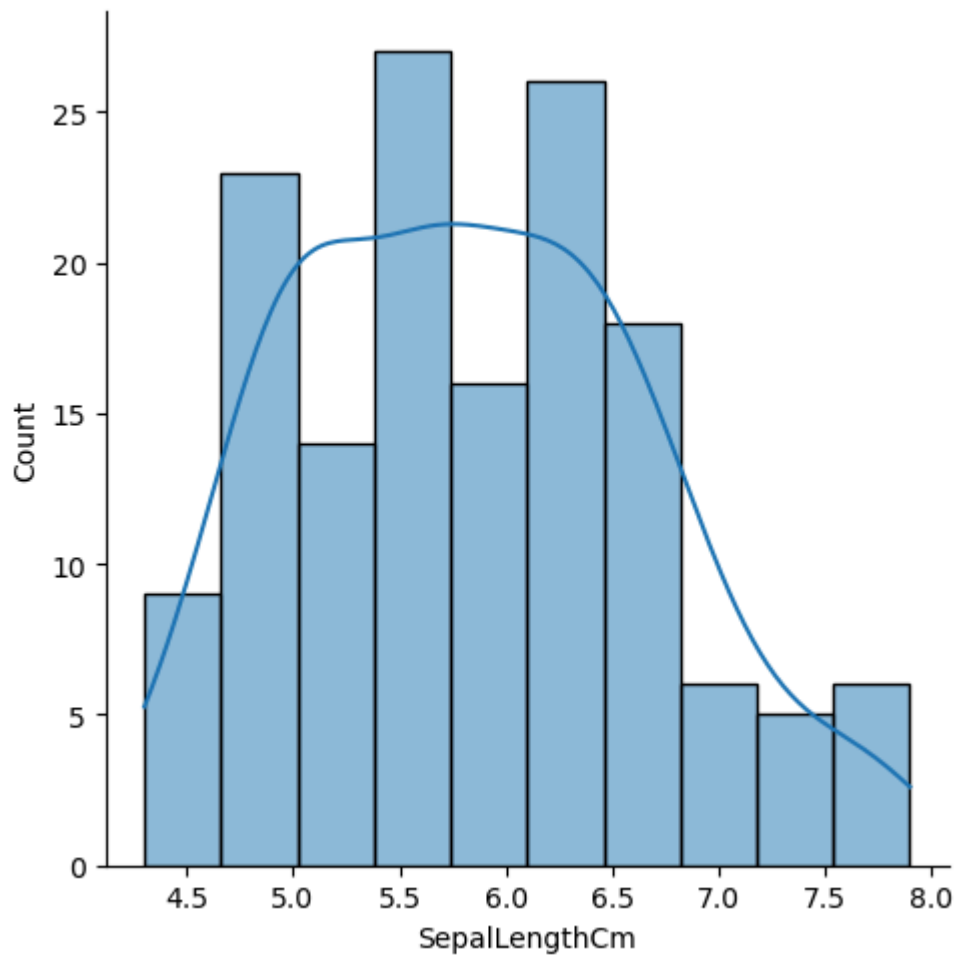
```
In [30]: sns.distplot(iris['SepalLengthCm'],kde=True, bins=10)
```

```
Out[30]: <Axes: xlabel='SepalLengthCm', ylabel='Density'>
```

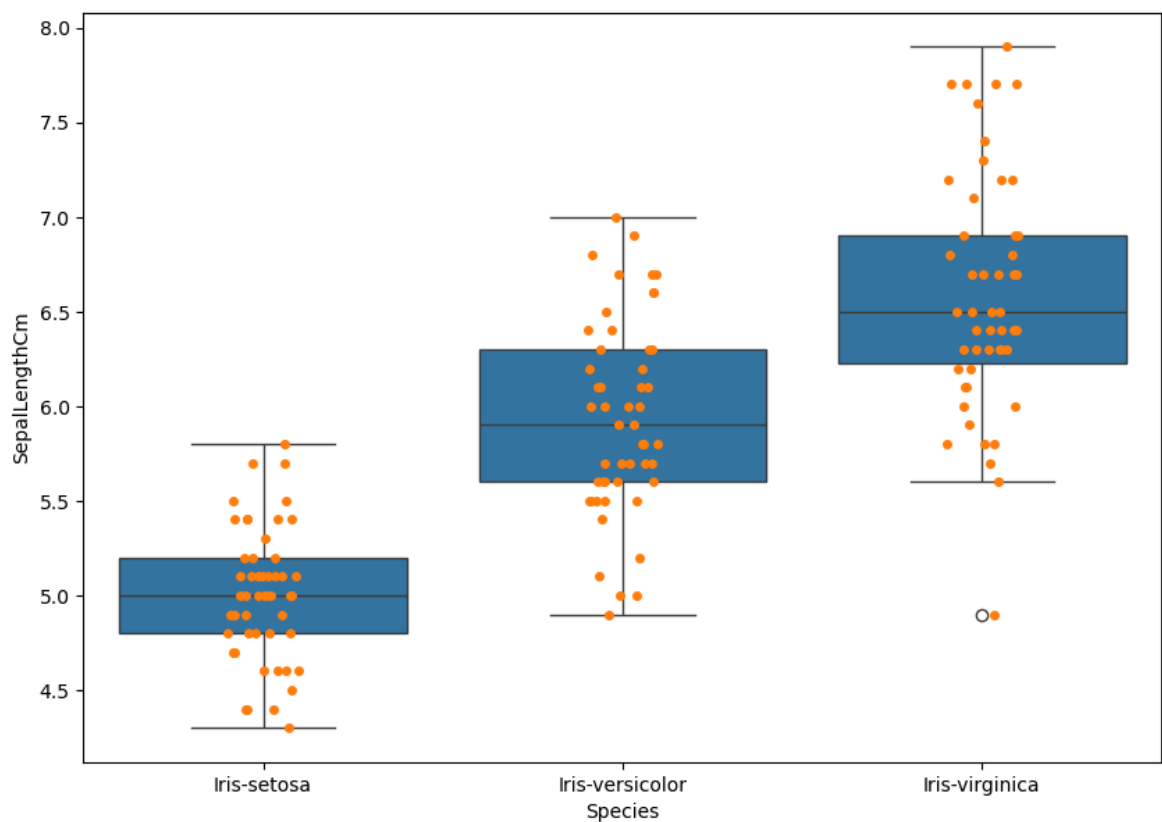


```
In [31]: sns.displot(iris['SepalLengthCm'],kde=True, bins=10)
```

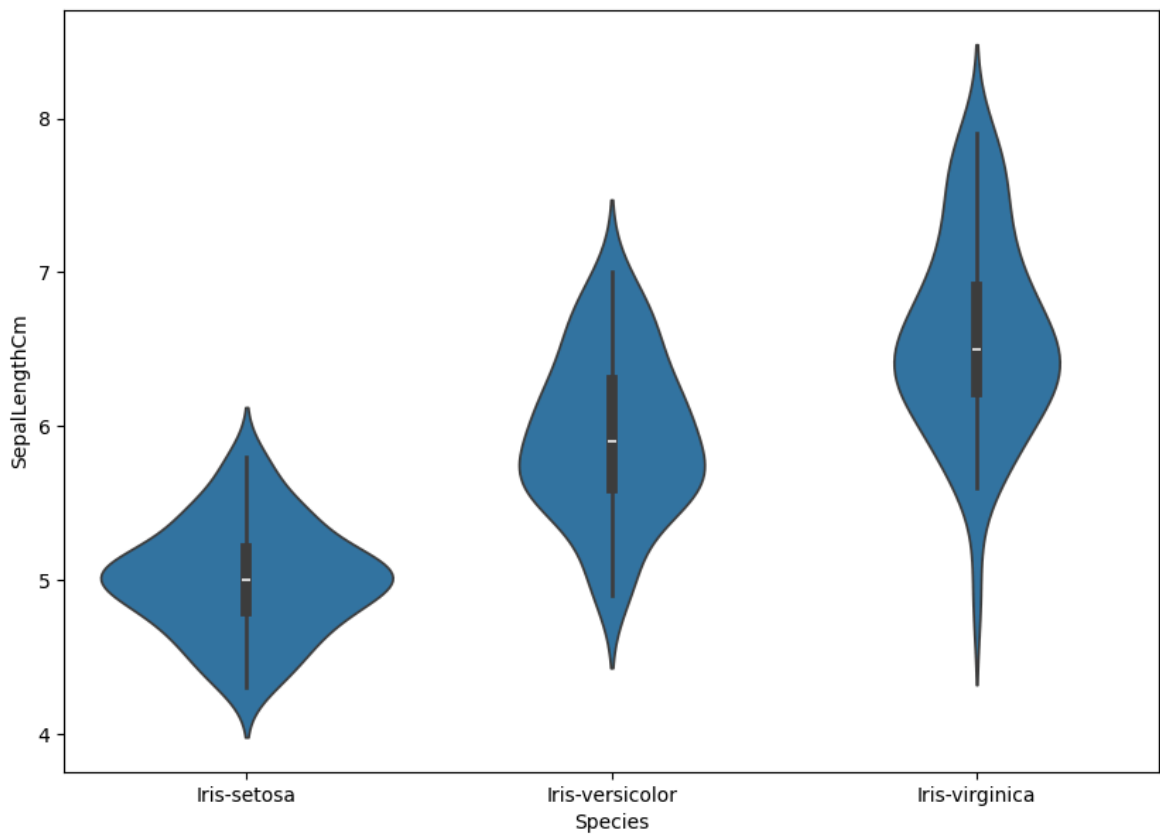
```
Out[31]: <seaborn.axisgrid.FacetGrid at 0x263d6136fc0>
```



```
In [32]: fig=plt.gcf()
fig.set_size_inches(10,7)
fig=sns.boxplot(x='Species',y='SepalLengthCm',data= iris)
fig=sns.stripplot(x='Species',y='SepalLengthCm',data=iris, jitter=True,edgecolor
```

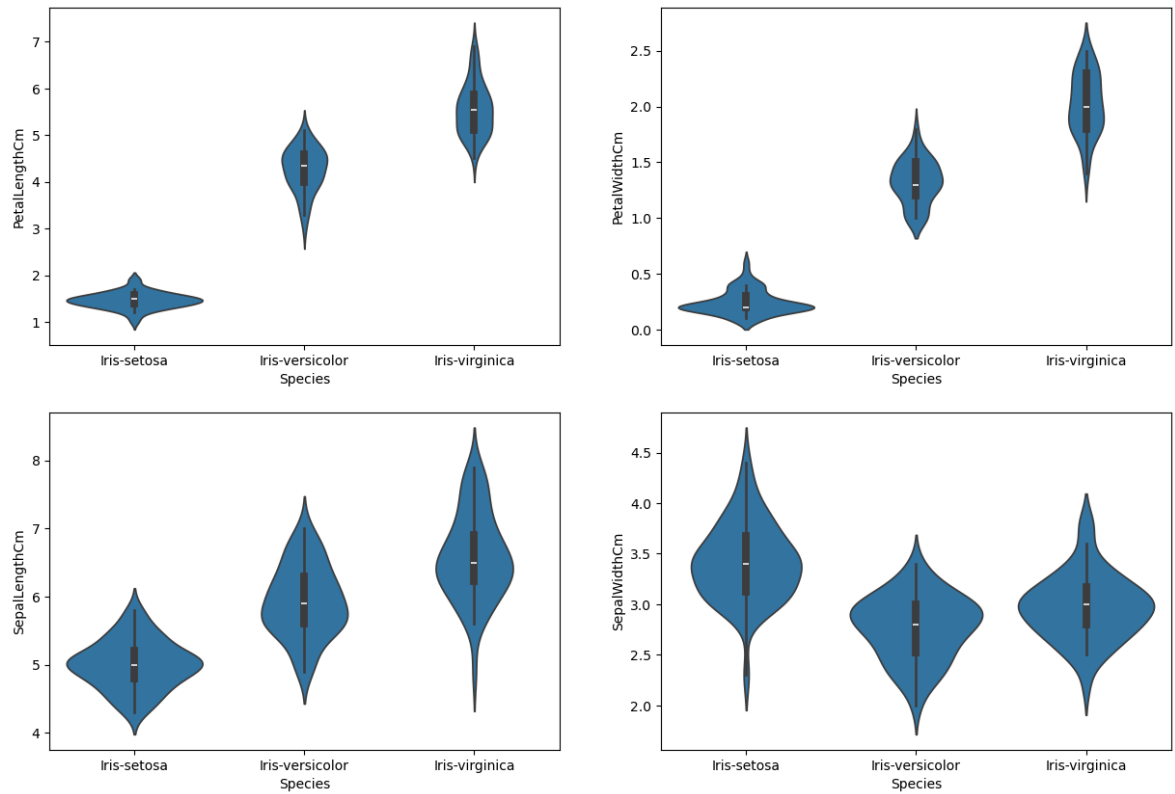


```
In [33]: fig=plt.gcf()
fig.set_size_inches(10,7)
fig=sns.violinplot(x='Species',y='SepallLengthCm',data=iris)
```



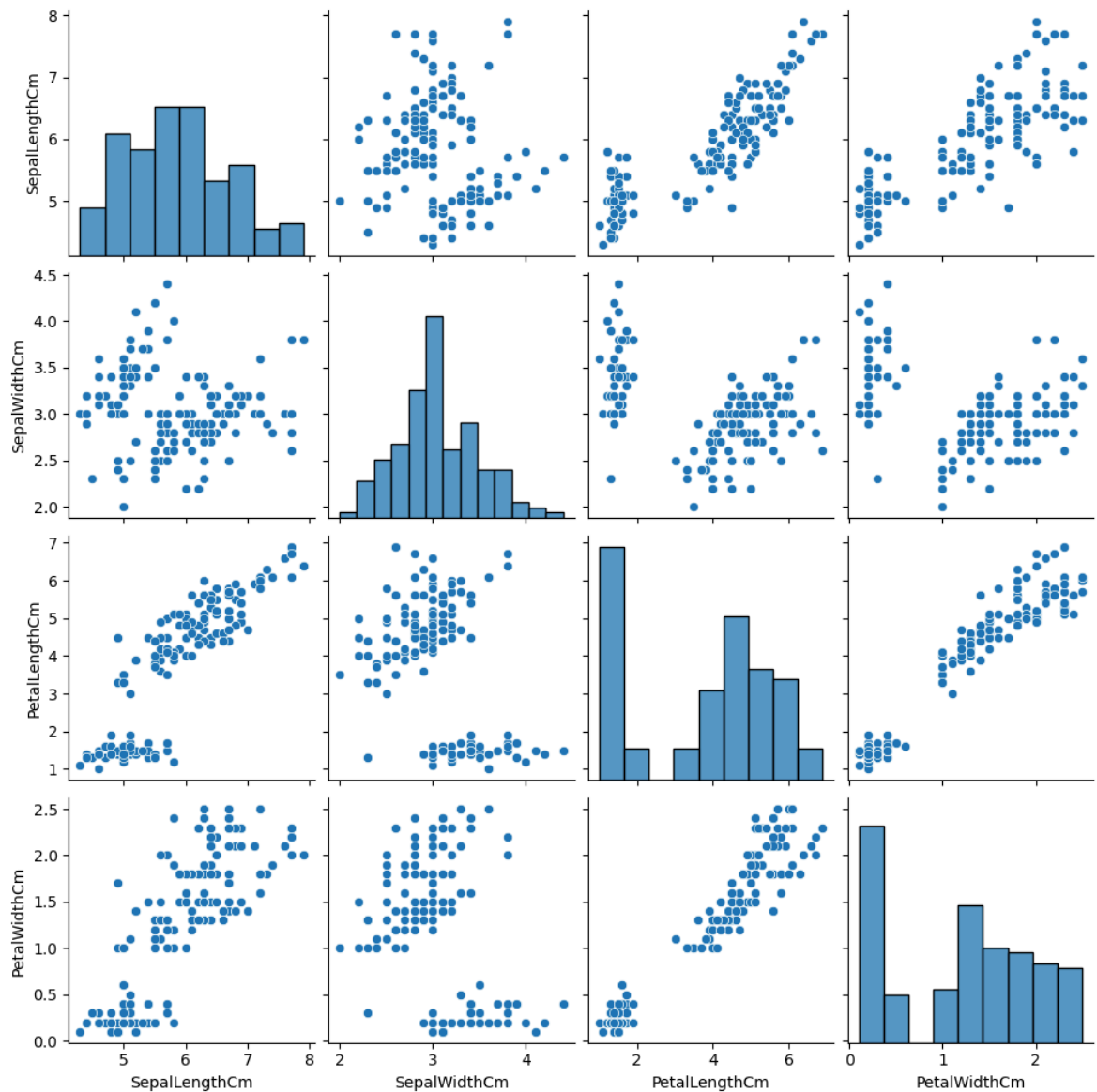
```
In [34]: plt.figure(figsize=(15,10))
plt.subplot(2,2,1)
sns.violinplot(x='Species',y='PetalLengthCm',data=iris)
plt.subplot(2,2,2)
sns.violinplot(x='Species',y='PetalWidthCm',data=iris)
plt.subplot(2,2,3)
sns.violinplot(x='Species',y='SepallLengthCm',data=iris)
plt.subplot(2,2,4)
sns.violinplot(x='Species',y='SepalWidthCm',data=iris)
```

```
Out[34]: <Axes: xlabel='Species', ylabel='SepalWidthCm'>
```

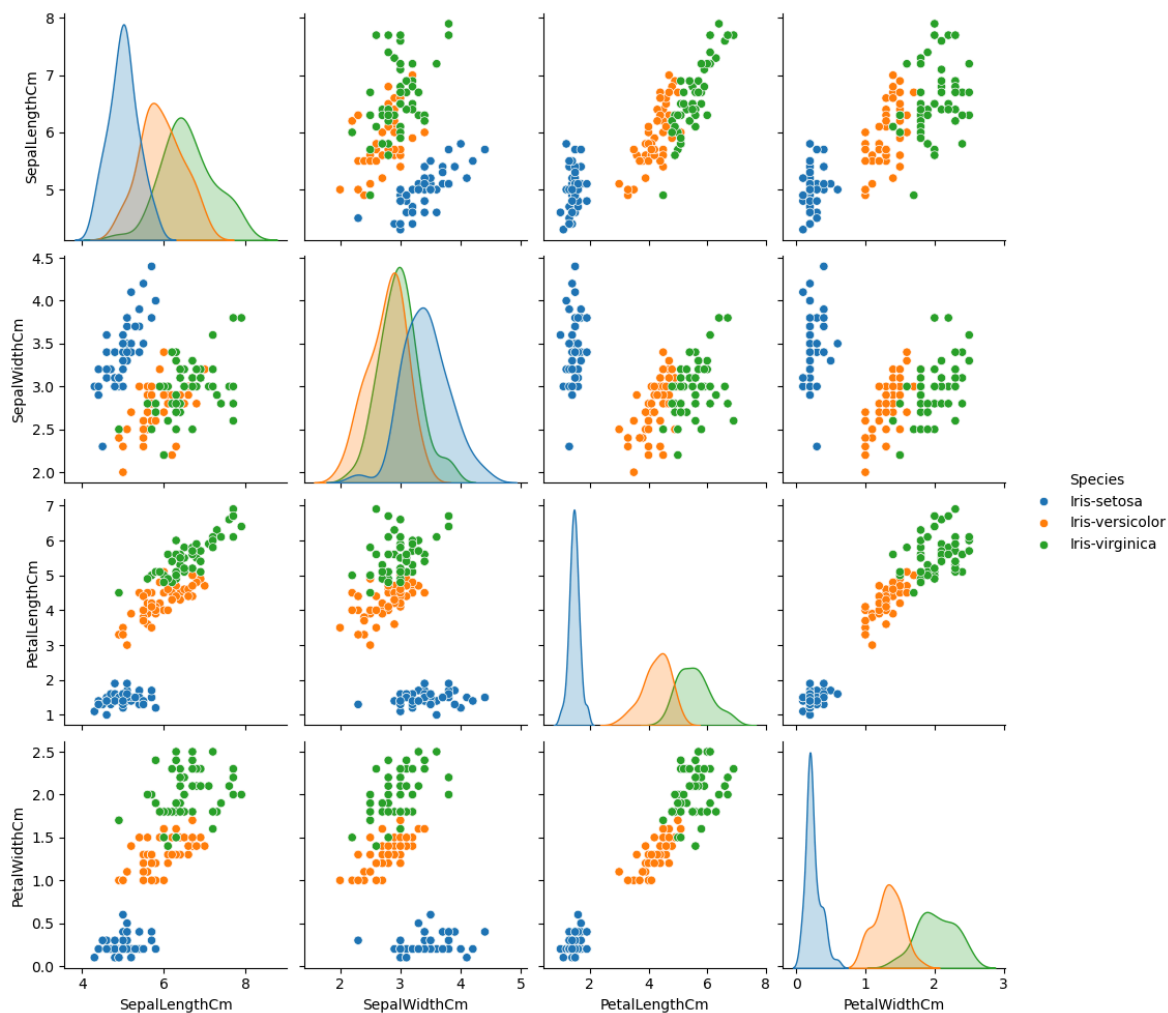


```
In [35]: sns.pairplot(data=iris,kind='scatter')
```

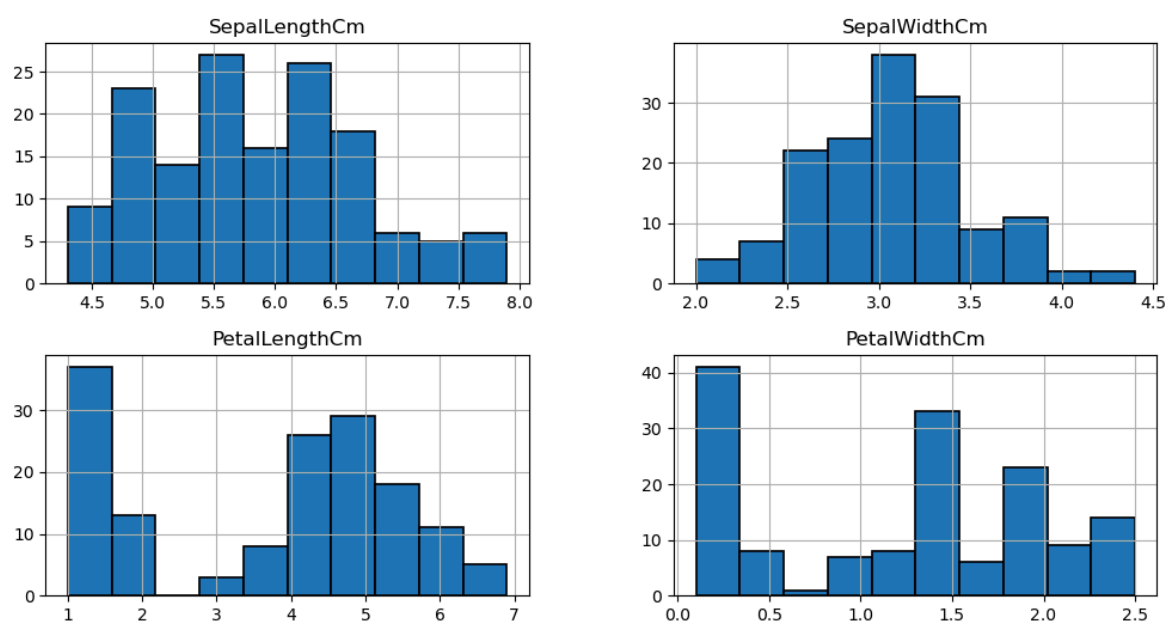
```
Out[35]: <seaborn.axisgrid.PairGrid at 0x263d697a540>
```



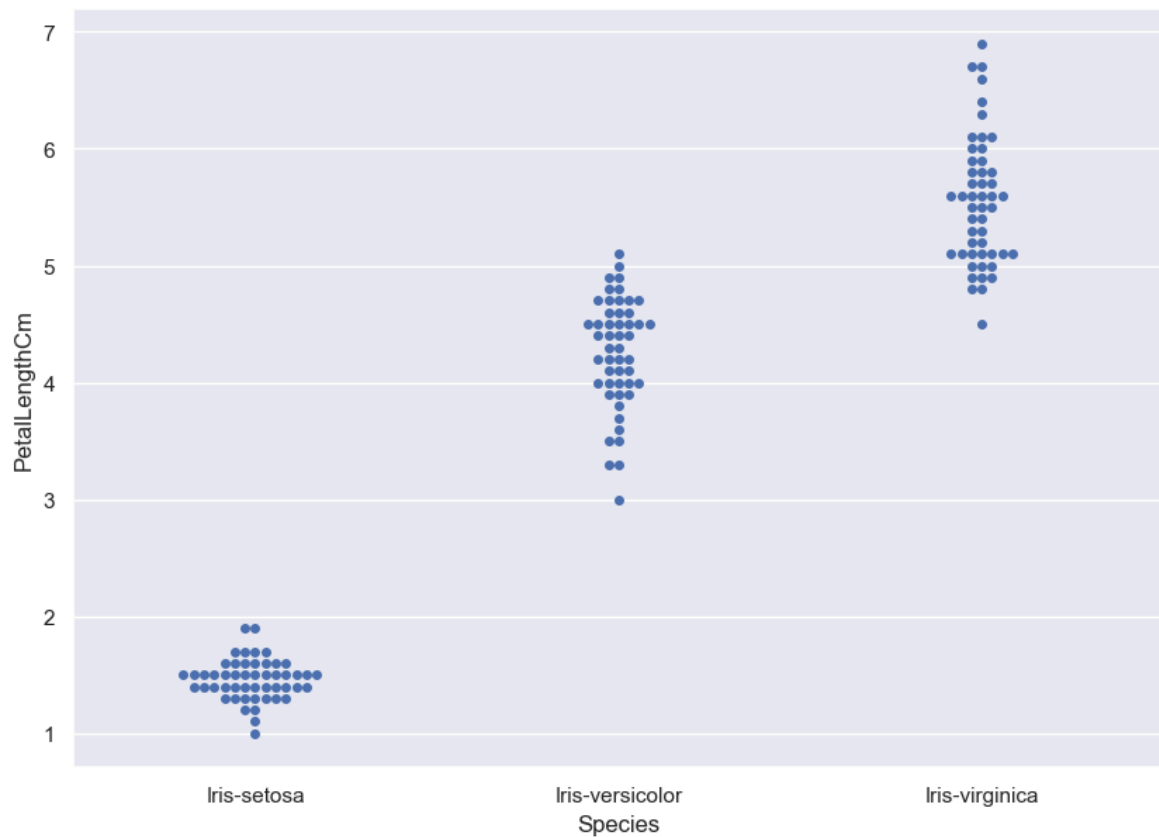
```
In [36]: sns.pairplot(iris,hue='Species');
```

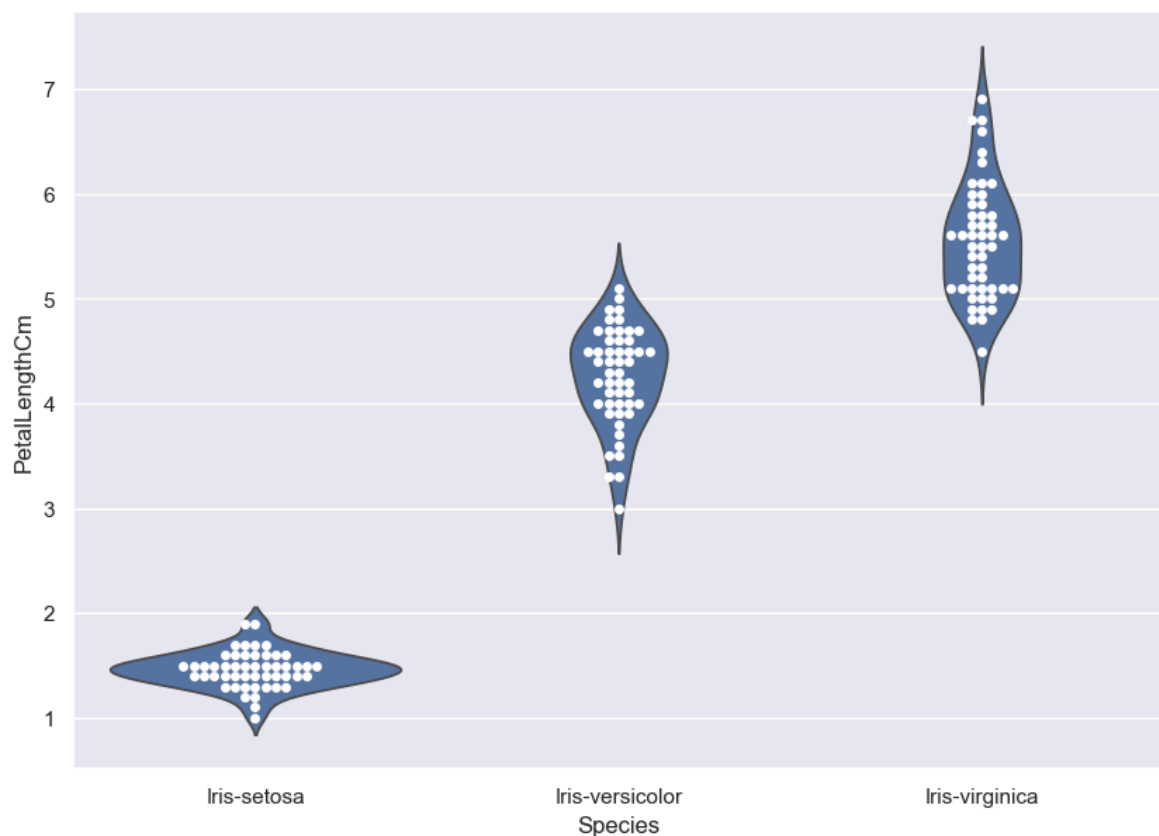
```
In [37]: iris.hist(edgecolor='black',linewidth=1.2)
fig=plt.gcf()
fig.set_size_inches(12,6)
```



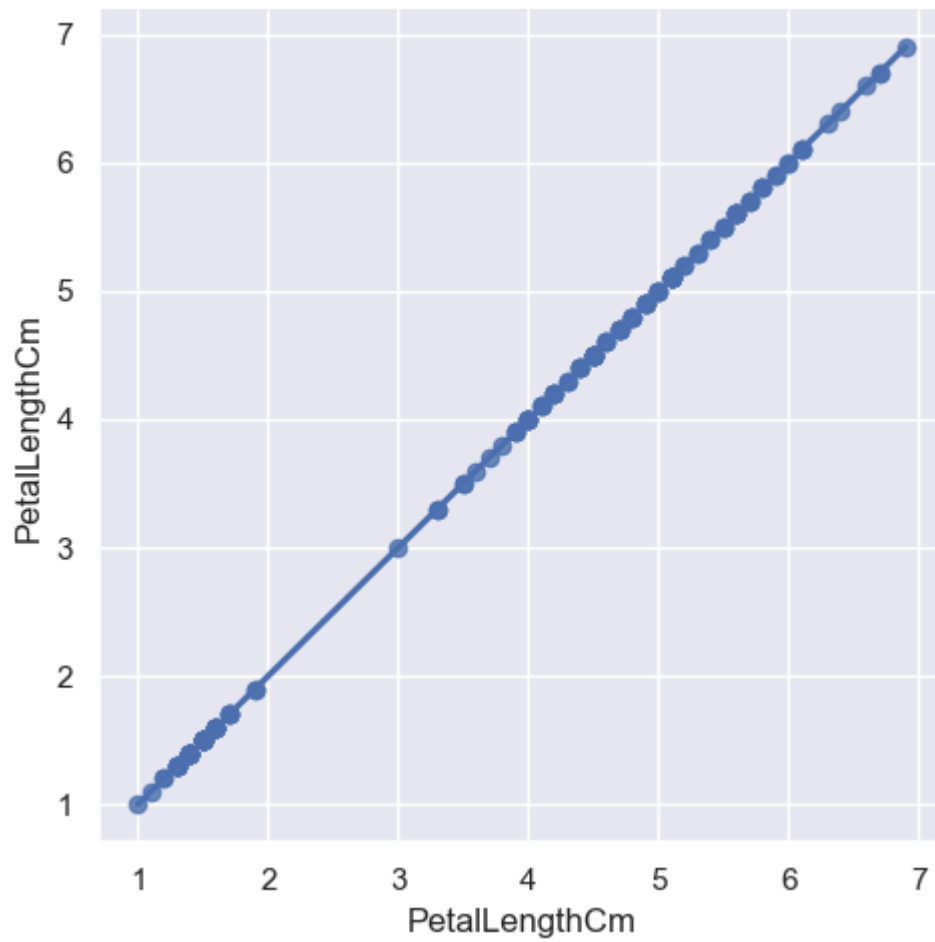
```
In [38]: sns.set(style='darkgrid')
fig=plt.gcf()
fig.set_size_inches(10,7)
fig=sns.swarmplot(x='Species',y='PetalLengthCm',data=iris)
```



```
In [39]: sns.set(style='darkgrid')
fig=plt.gcf()
fig.set_size_inches(10,7)
ax=sns.violinplot(x='Species',y='PetalLengthCm',data=iris,inner=None)
ax=sns.swarmplot(x='Species',y='PetalLengthCm',data=iris,color='white',edgecolor
```

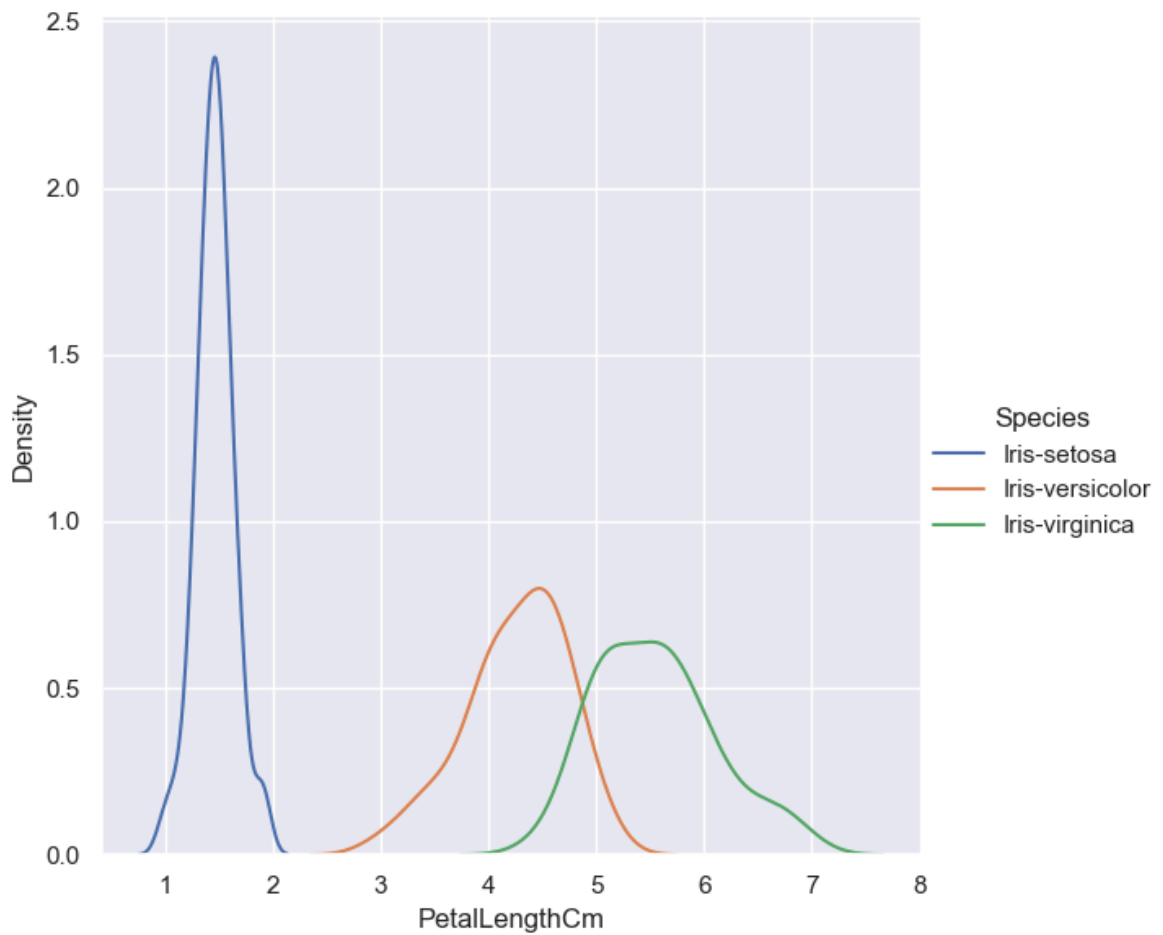


```
In [40]: ax=sns.lmplot(x='PetalLengthCm',y='PetalLengthCm',data=iris)
```

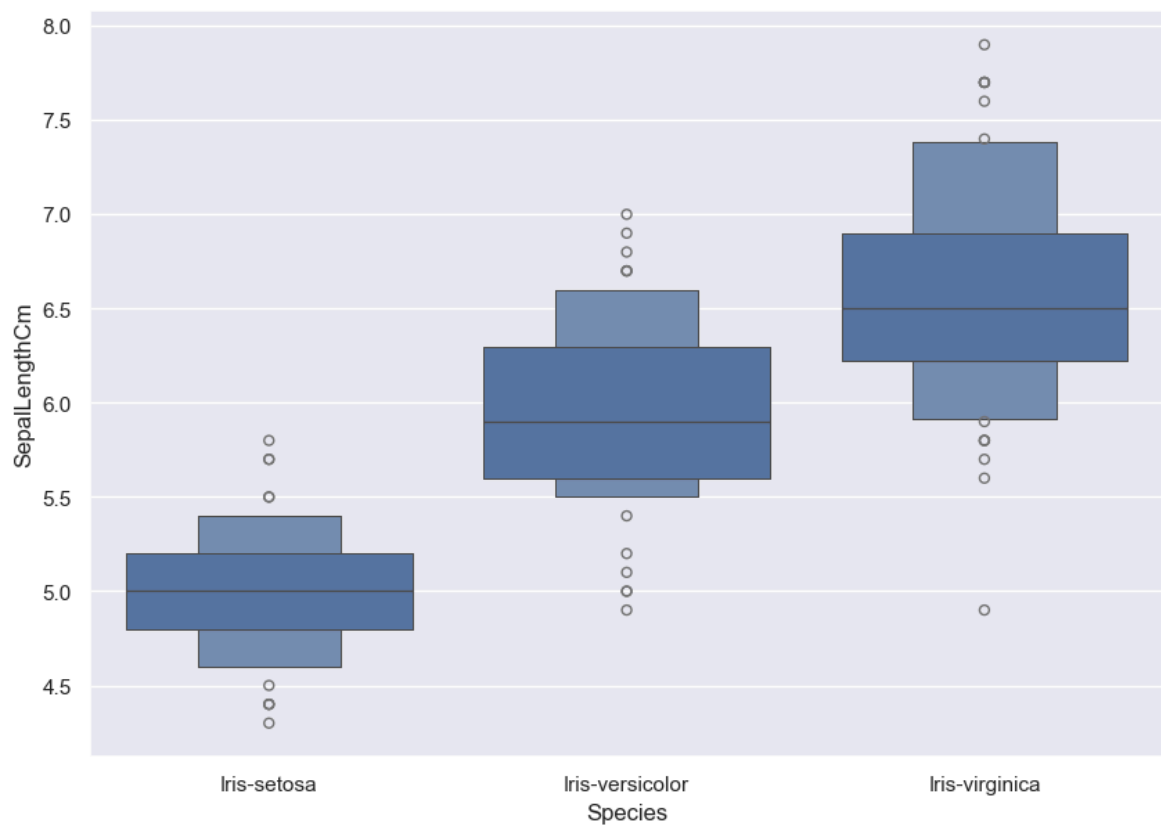


```
In [41]: sns.FacetGrid(iris, hue='Species',height=6)\
        .map(sns.kdeplot,"PetalLengthCm")\
        .add_legend()\
plt.ioff()
```

```
Out[41]: <contextlib.ExitStack at 0x263d81b2540>
```



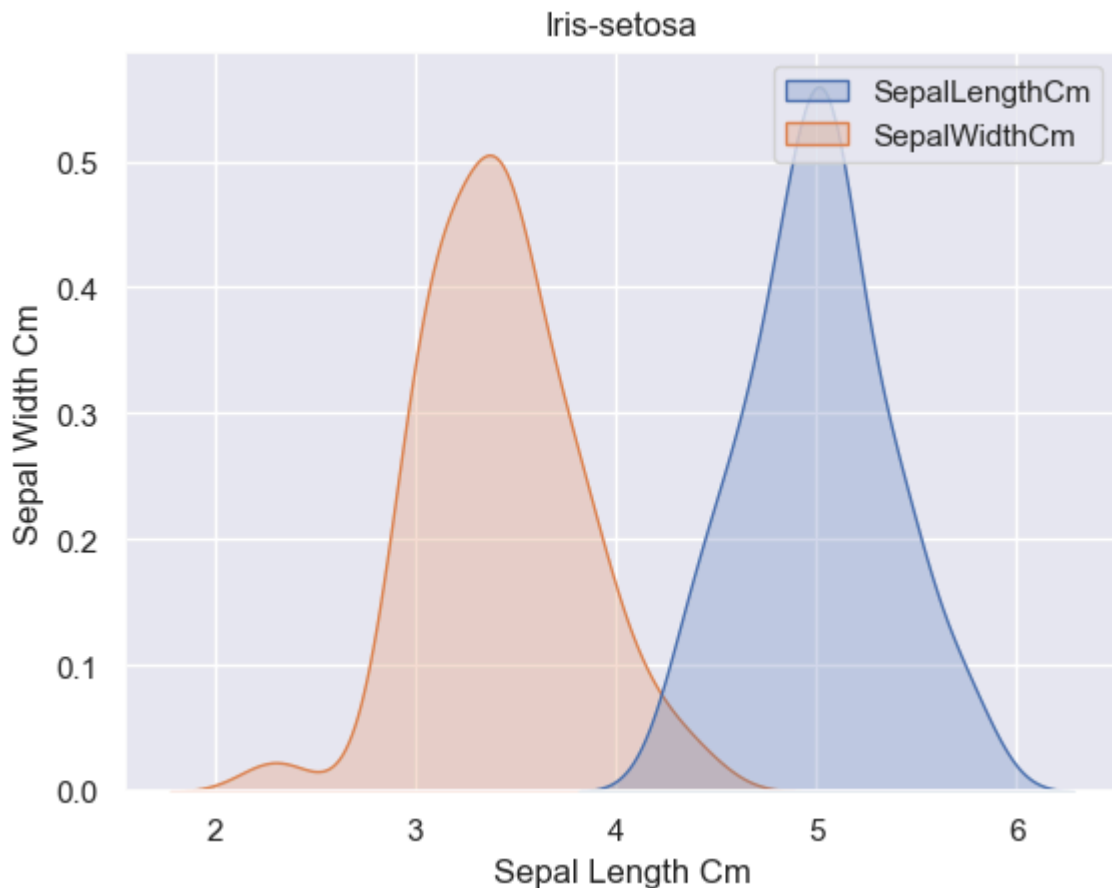
```
In [42]: fig=plt.gcf()
fig.set_size_inches(10,7)
fig=sns.boxenplot(x='Species',y='SepalLengthCm',data=iris)
plt.show()
```



```
In [43]: sub=iris[iris['Species']=='Iris-setosa']
sns.kdeplot(data=sub[['SepalLengthCm','SepalWidthCm']],
            shade=True,shade_lowest=False)
plt.title('Iris-setosa')
plt.xlabel('Sepal Length Cm')
plt.ylabel('Sepal Width Cm')
```

Out[43]: Text(0, 0.5, 'Sepal Width Cm')

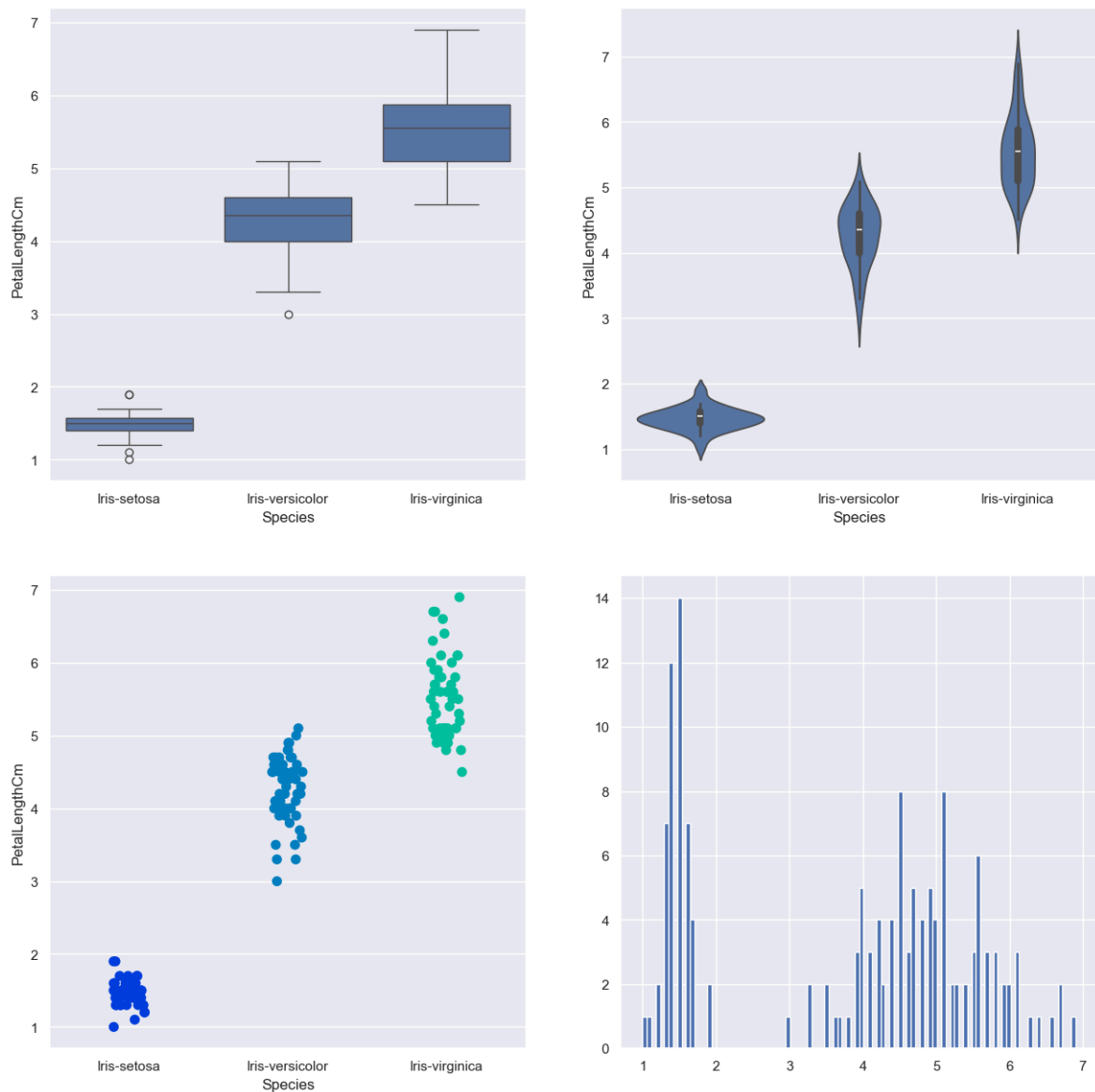
```
In [44]: plt.show()
```



```
In [45]: sns.set_style('darkgrid')
f,axes=plt.subplots(2,2,figsize=(15,15))

k1=sns.boxplot(x='Species',y='PetalLengthCm',data=iris,ax=axes[0,0])
k1=sns.violinplot(x='Species',y='PetalLengthCm',data=iris,ax=axes[0,1])
k1=sns.stripplot(x='Species',y='PetalLengthCm',data=iris,jitter=True,edgecolor='

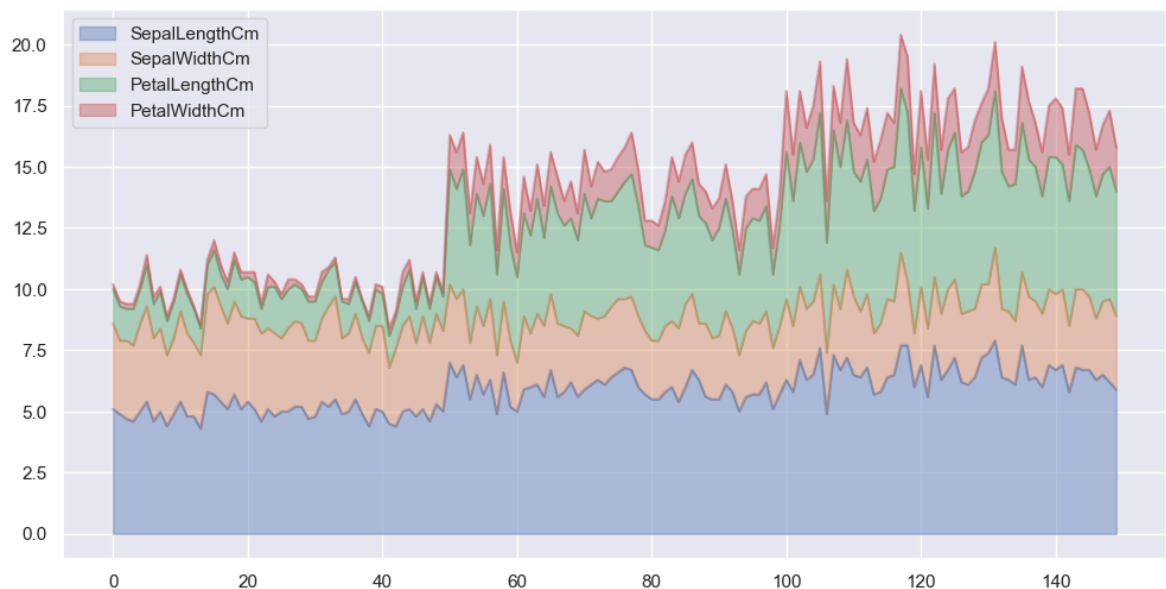
axes[1,1].hist(iris.PetalLengthCm,bins=100)
plt.show()
```



```
In [46]: iris['Spcies']=iris['Species'].astype('category')
```

```
In [70]: iris.plot.area(y=['SepalLengthCm','SepalWidthCm','PetalLengthCm','PetalWidthCm'])
```

```
In [72]: plt.show()
```



```
In [84]: sns.distplot(iris['SepalLengthCm'],kde=True,bins=20);
```

```
In [86]: plt.show()
```

