

```
In [1]: import numpy as np
import pandas as pd
```

```
In [3]: import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import warnings
warnings.filterwarnings('ignore')
```

```
In [4]: iris=pd.read_csv(r'C:\Users\Hp\OneDrive\Desktop\25th, 26th- Advanced EDA project\iris.csv')
```

```
In [5]: iris
```

	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.head()
```

Out[6]:

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

In [8]: `iris.head()`

Out[8]:

	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 [9]: `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
```

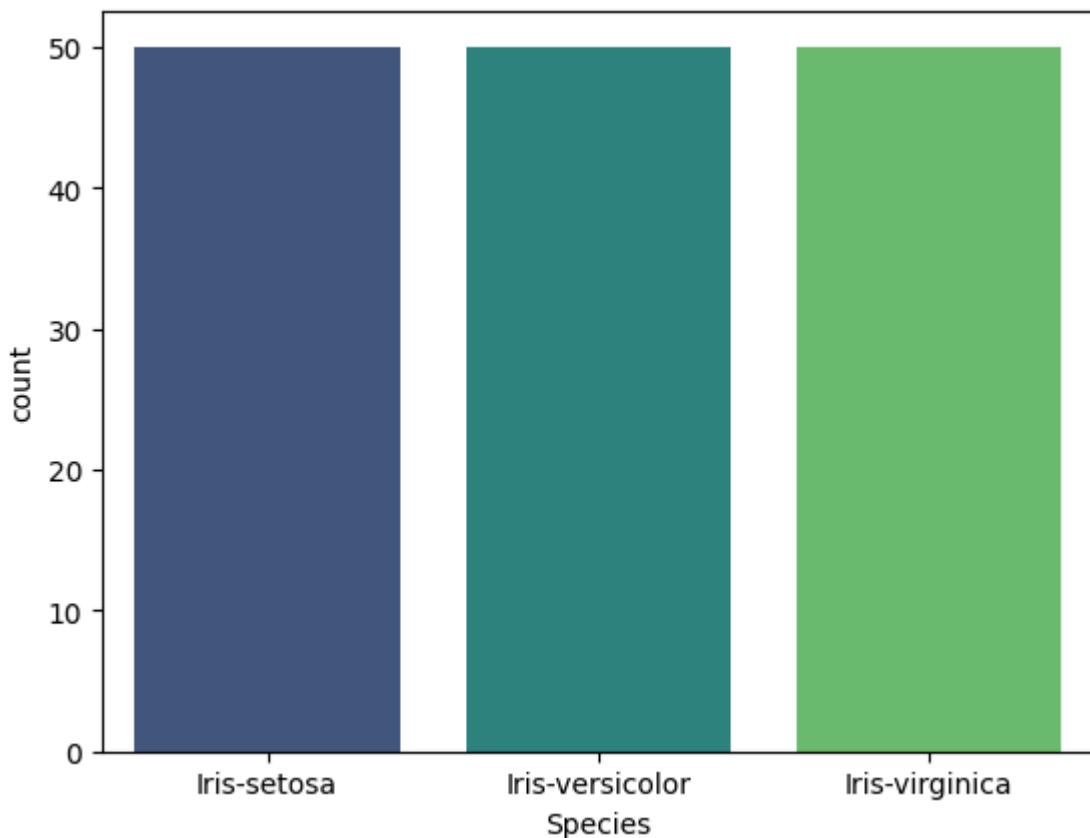
In [10]: `iris['Species'].value_counts()`

Out[10]:

Species	count
Iris-setosa	50
Iris-versicolor	50
Iris-virginica	50

Name: count, dtype: int64

In [15]: `sns.countplot(x='Species', data=iris, palette='viridis')`
`plt.show()`

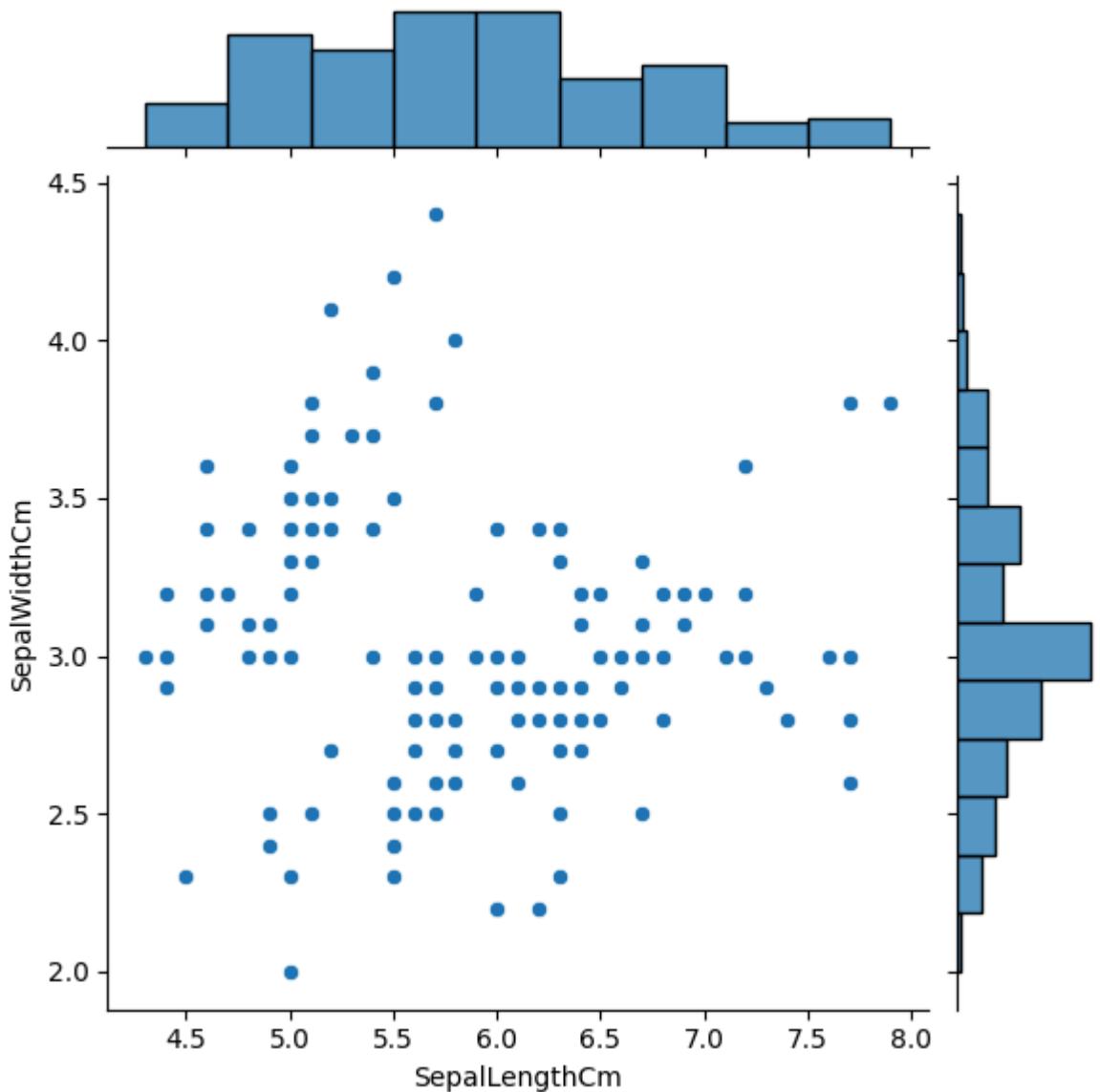


```
In [16]: iris.head()
```

```
Out[16]:
```

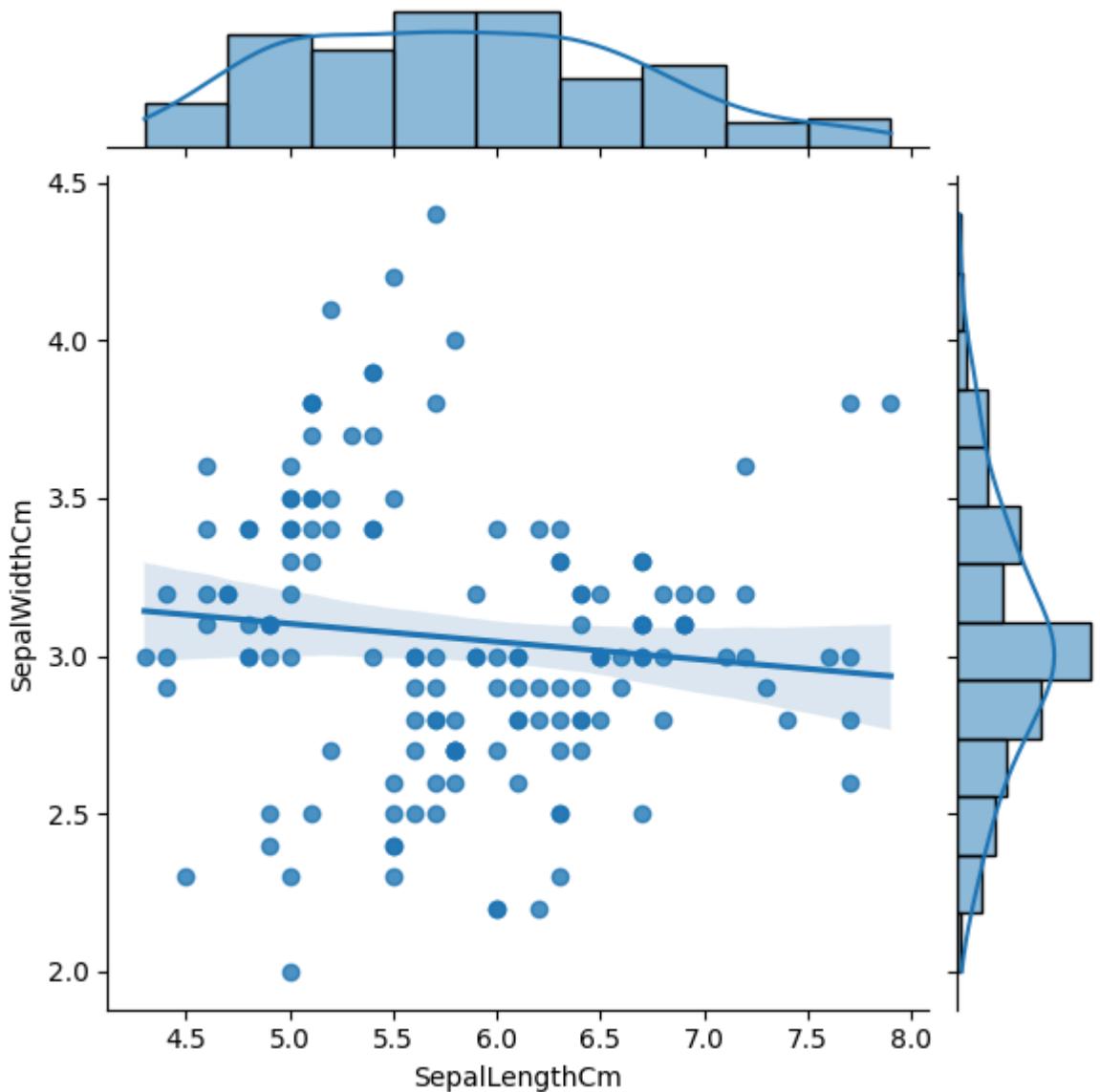
	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	5.1	3.5	1.4	0.2	Iris-setosa
1	4.9	3.0	1.4	0.2	Iris-setosa
2	4.7	3.2	1.3	0.2	Iris-setosa
3	4.6	3.1	1.5	0.2	Iris-setosa
4	5.0	3.6	1.4	0.2	Iris-setosa

```
In [17]: fig=sns.jointplot(x='SepalLengthCm',y='SepalWidthCm',data=iris)
```

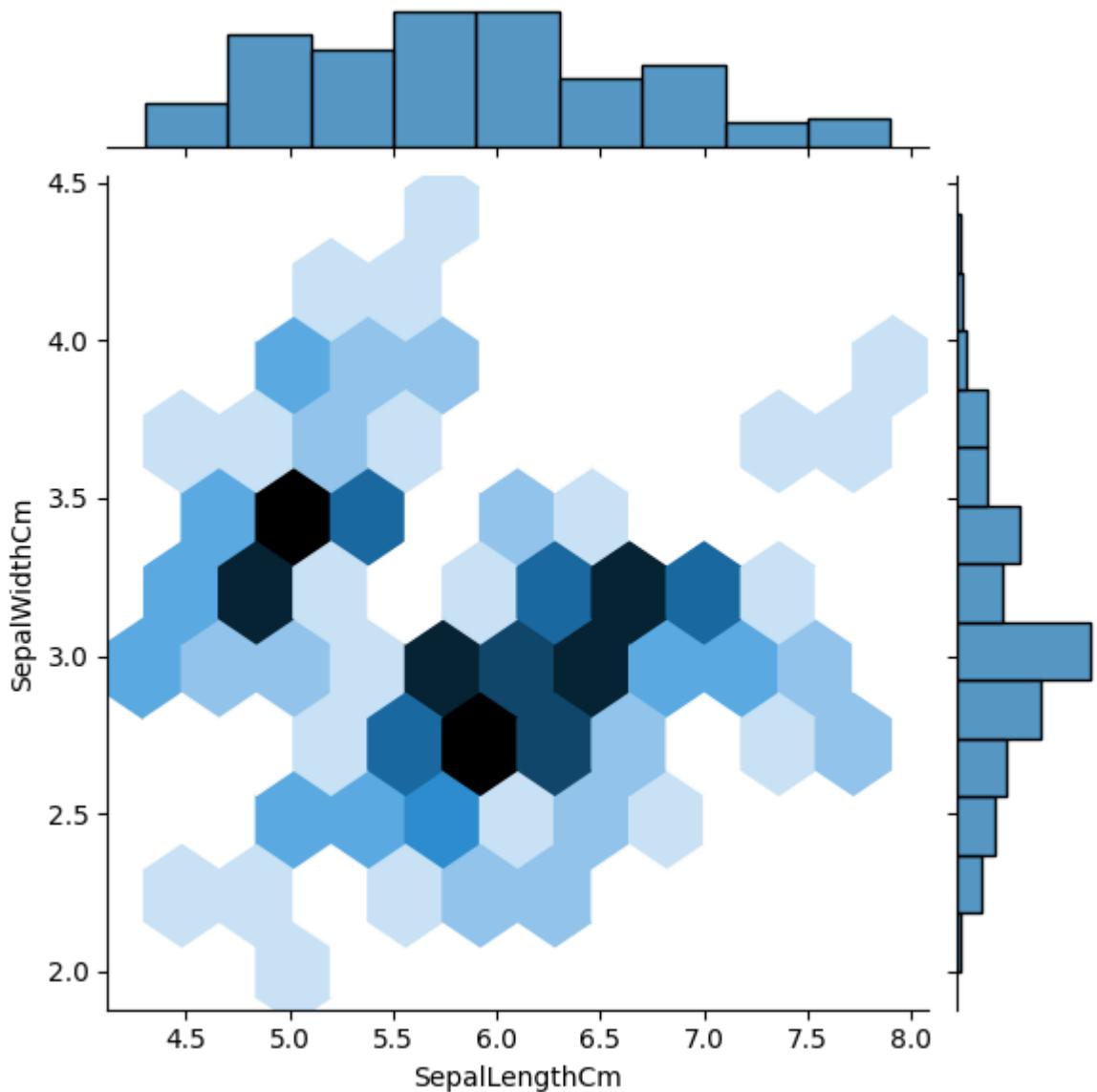


```
In [20]: sns.jointplot(x="SepalLengthCm",y= "SepalWidthCm", data=iris, kind="reg")
```

```
Out[20]: <seaborn.axisgrid.JointGrid at 0x1d9dfecbc50>
```

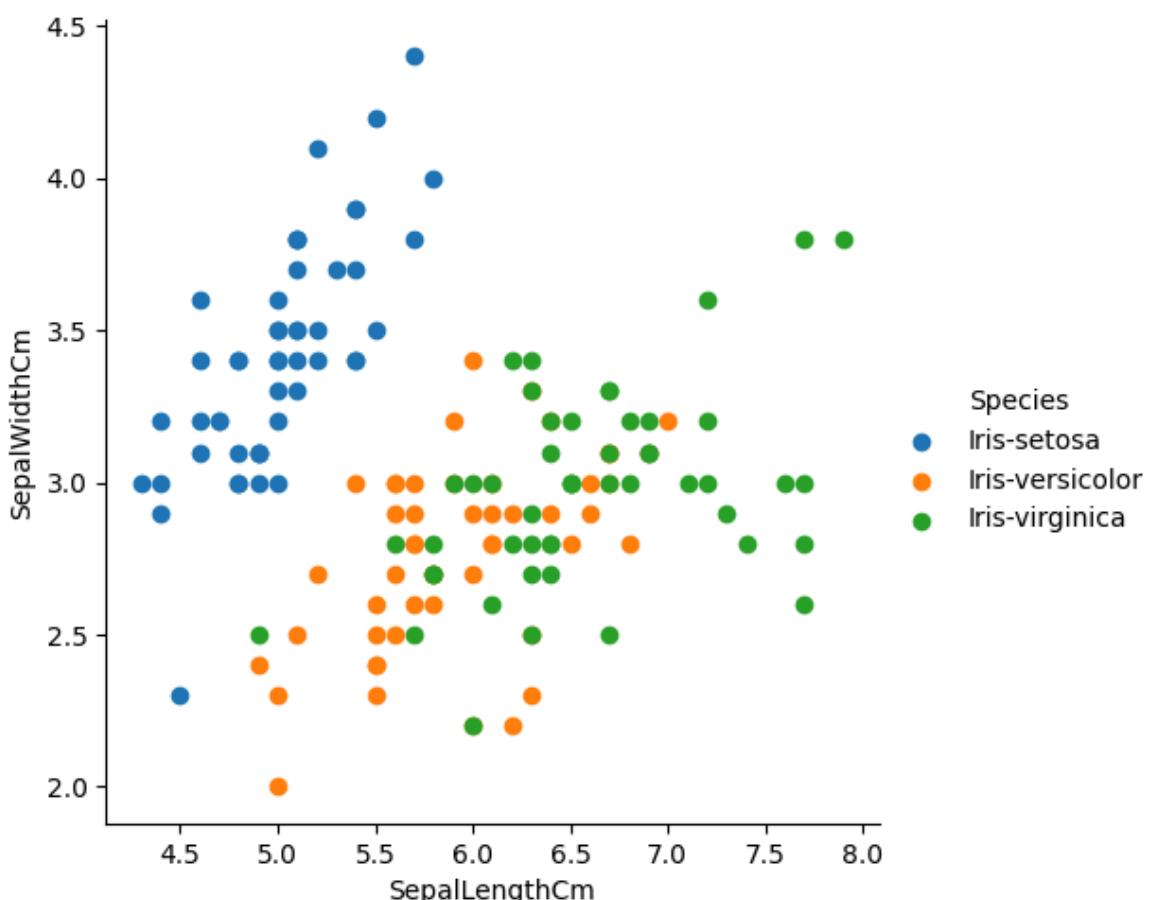
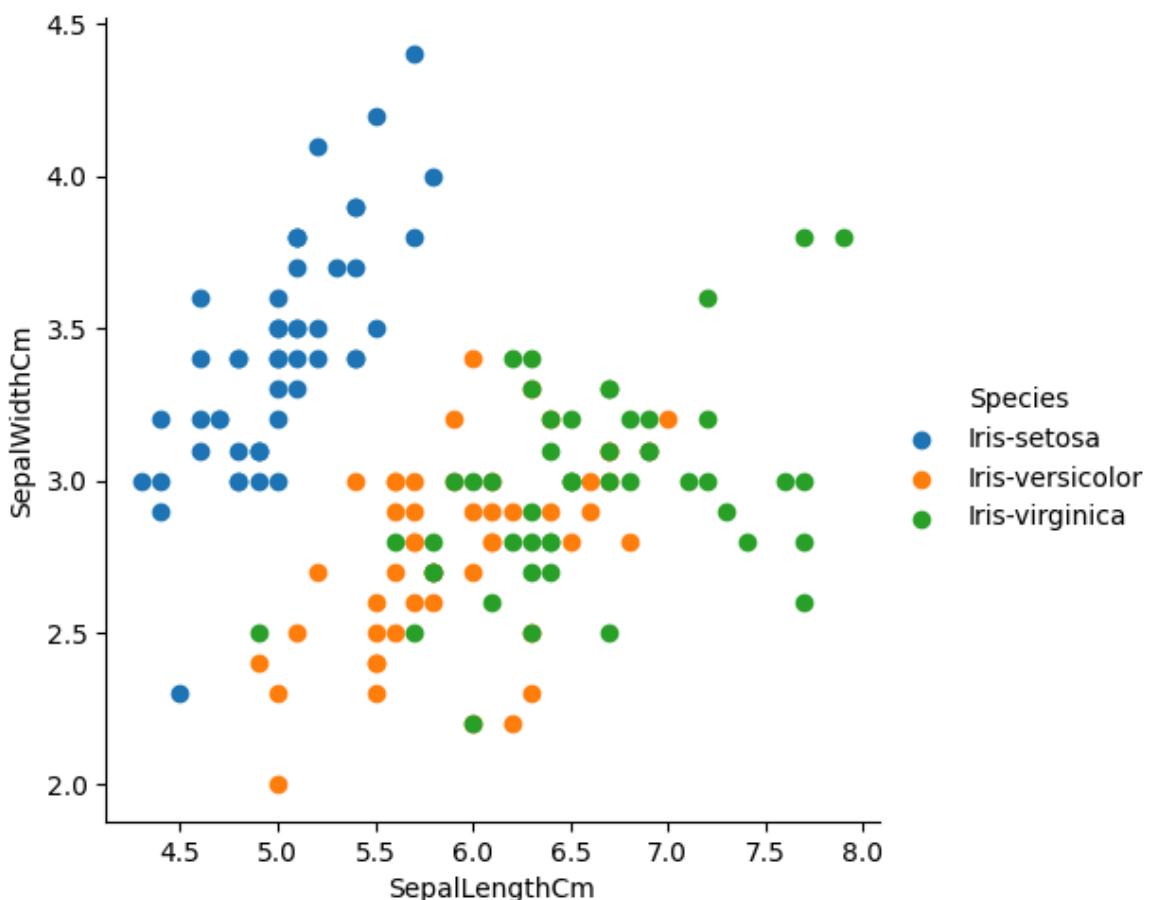


```
In [21]: fig=sns.jointplot(x='SepalLengthCm',y='SepalWidthCm',kind='hex',data=iris)
```



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

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



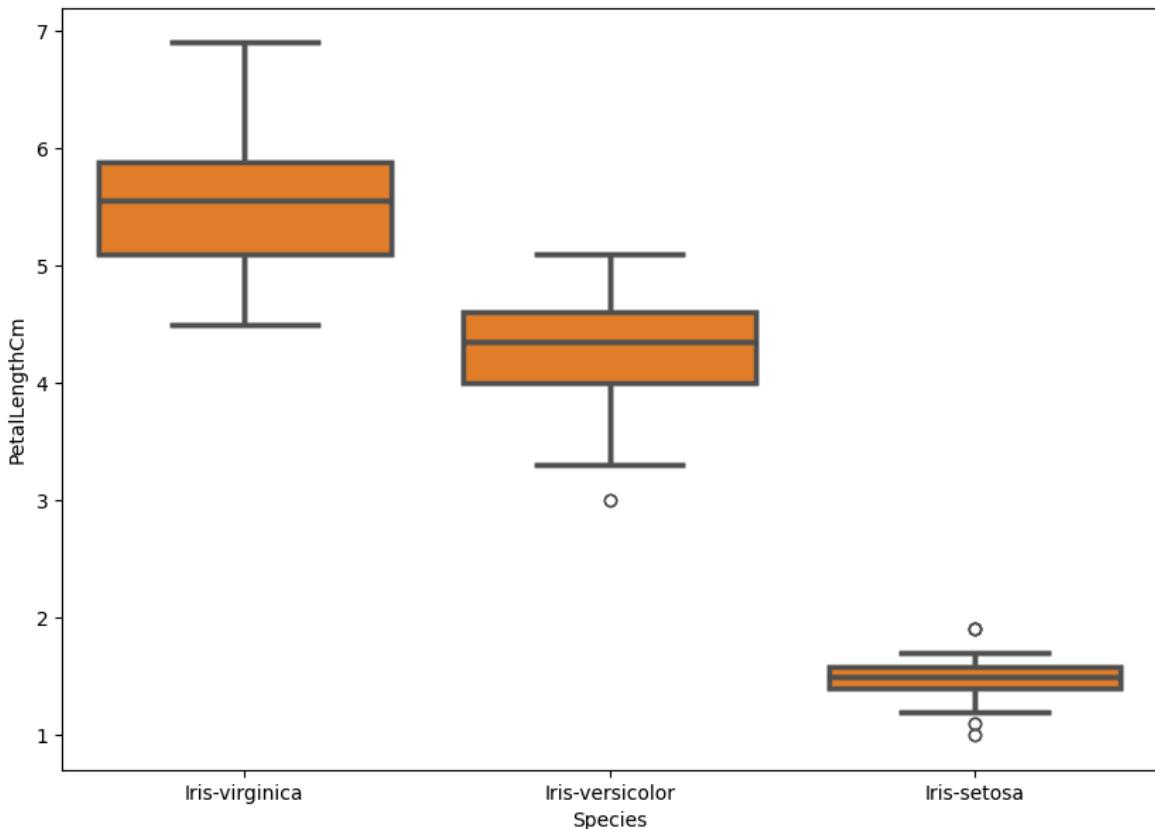
```
In [26]: iris.head()
```

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

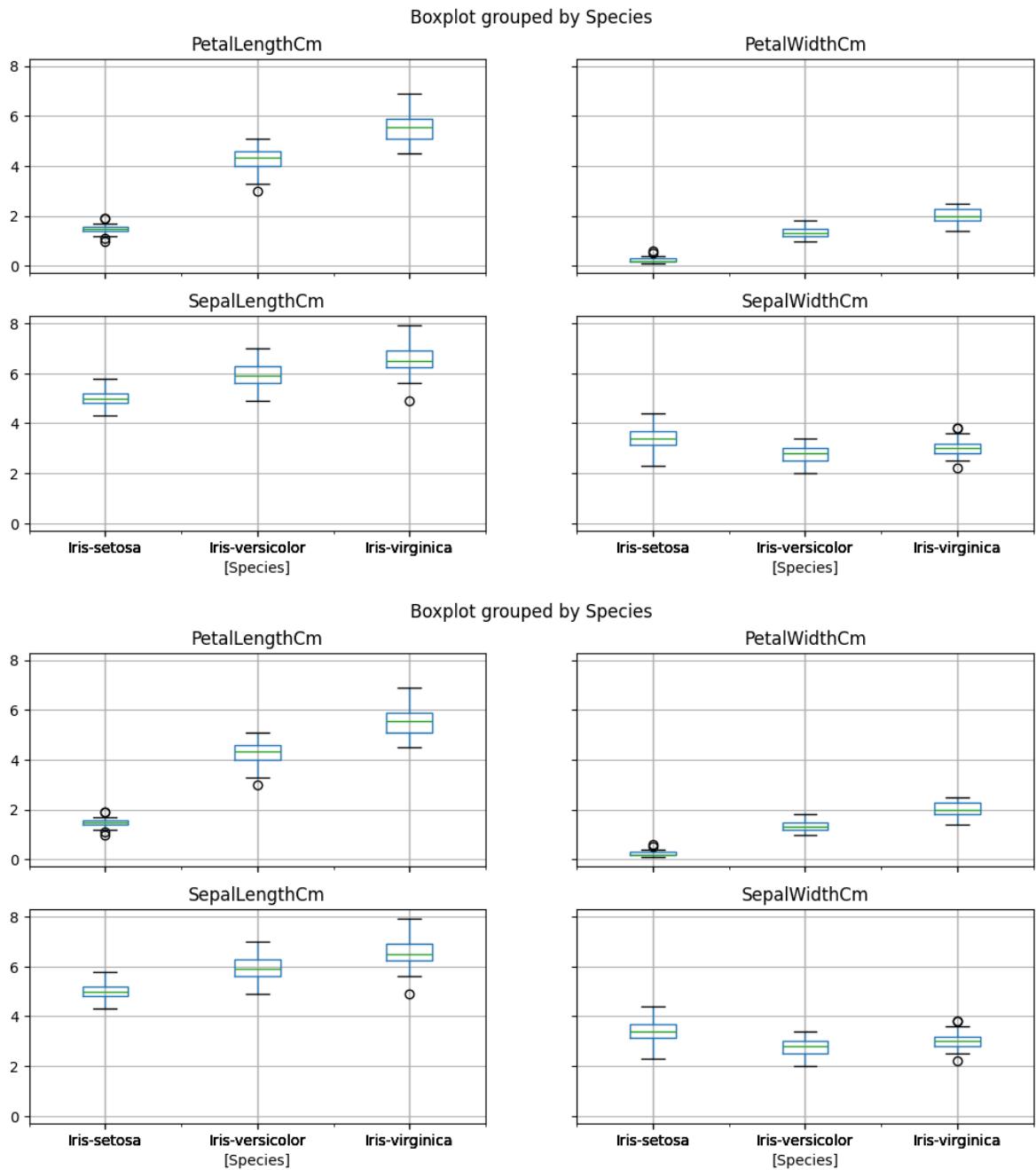
In [28]:

```
fig=plt.gcf()
fig.set_size_inches(10,7)
fig=sns.boxplot(x='Species',y='PetalLengthCm',data=iris,order=['Iris-virginica',
plt.show()
```

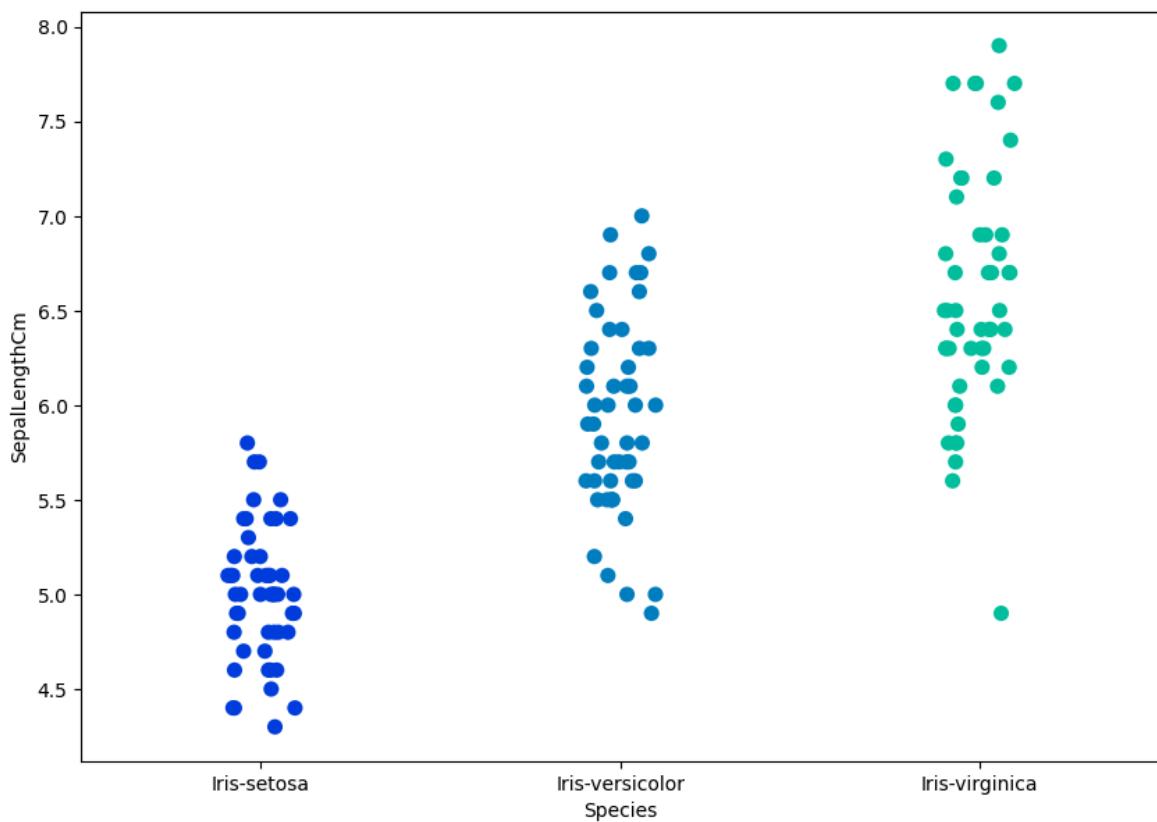


In [30]:

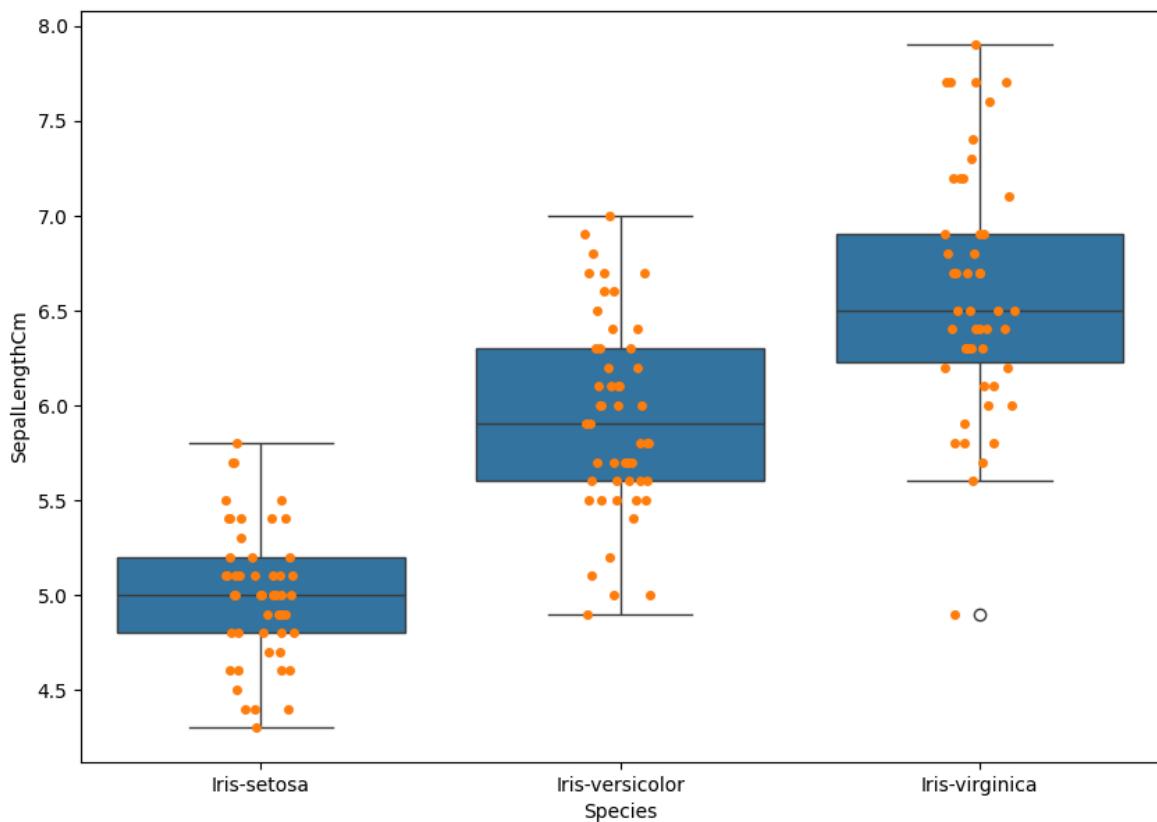
```
#iris.drop("Id", axis=1).boxplot(by="Species", figsize=(12, 6))
iris.boxplot(by="Species", figsize=(12, 6))
plt.show()
```



```
In [31]: fig=plt.gcf()
fig.set_size_inches(10,7)
fig=sns.stripplot(x='Species',y='SepalLengthCm',data=iris,jitter=True,edgecolor=plt.show()
```

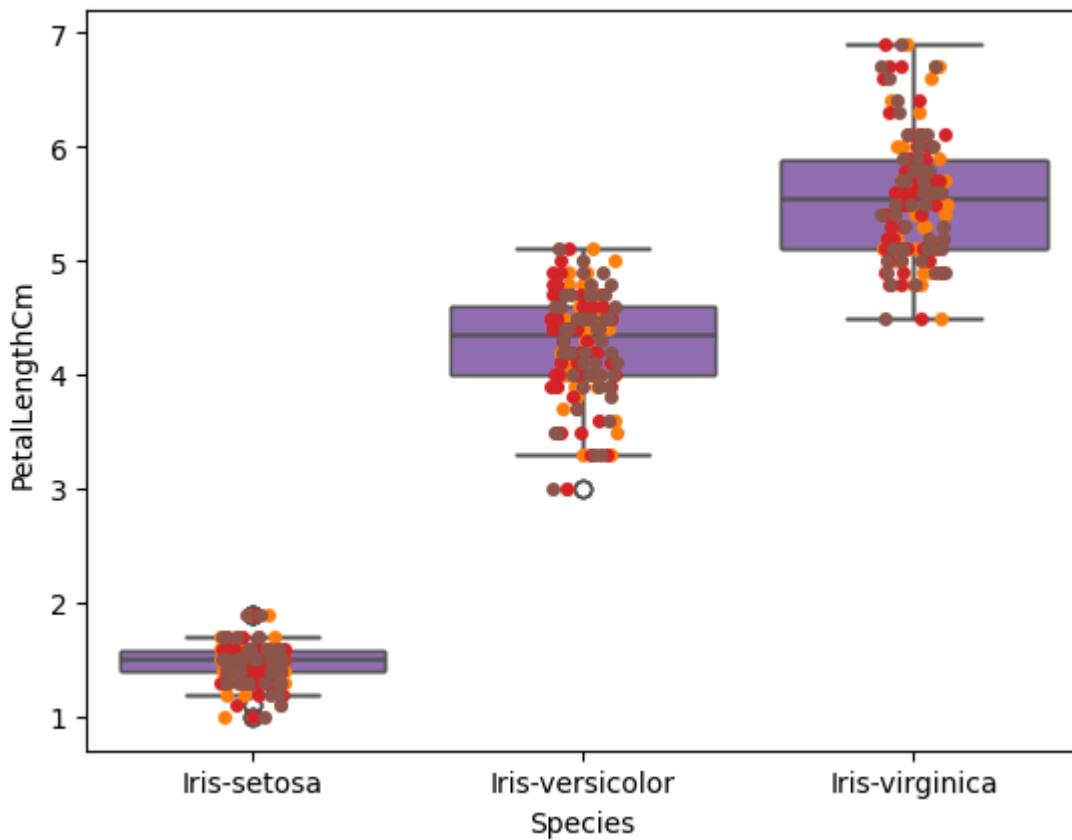


```
In [32]: fig=plt.gcf()
fig.set_size_inches(10,7)
fig=sns.boxplot(x='Species',y='SepalLengthCm',data=iris)
fig=sns.stripplot(x='Species',y='SepalLengthCm',data=iris,jitter=True,edgecolor='black')
plt.show()
```

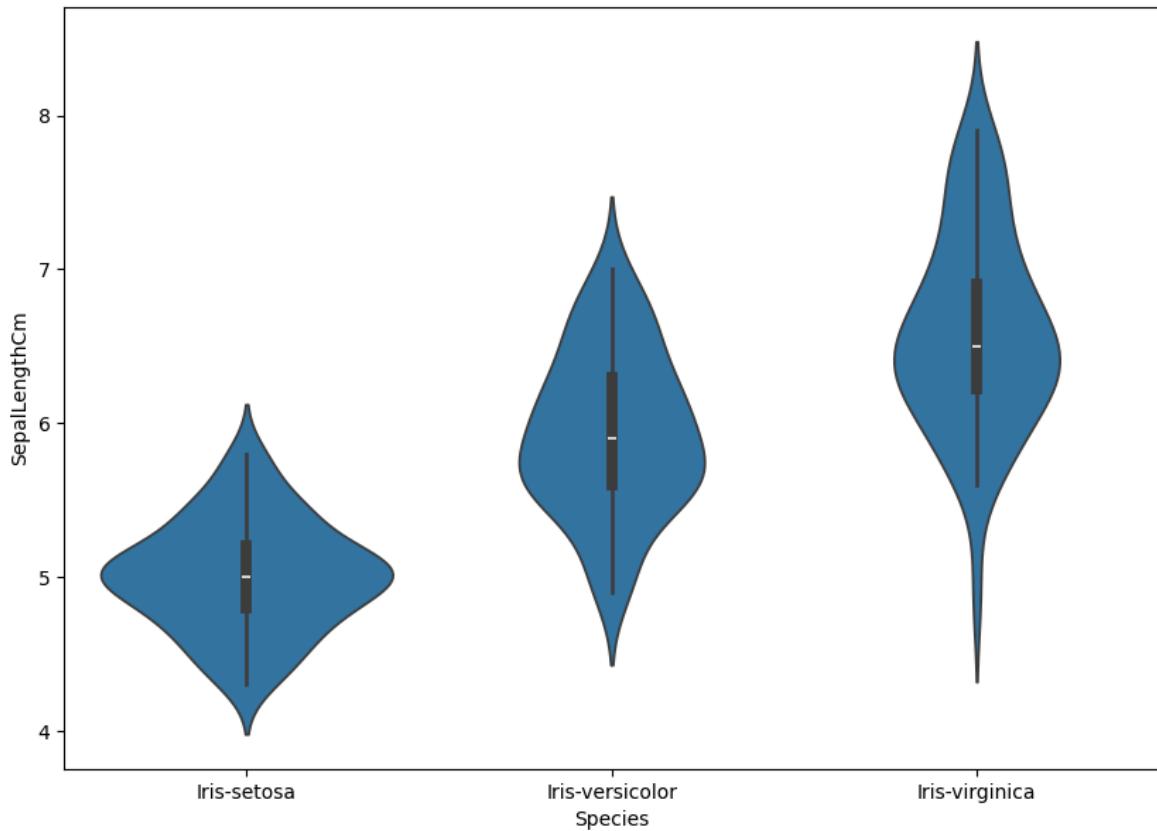


```
In [35]: ax = sns.boxplot(x="Species", y="PetalLengthCm", data=iris)
ax = sns.stripplot(x="Species", y="PetalLengthCm", data=iris, jitter=True, edgecolor='black')
```

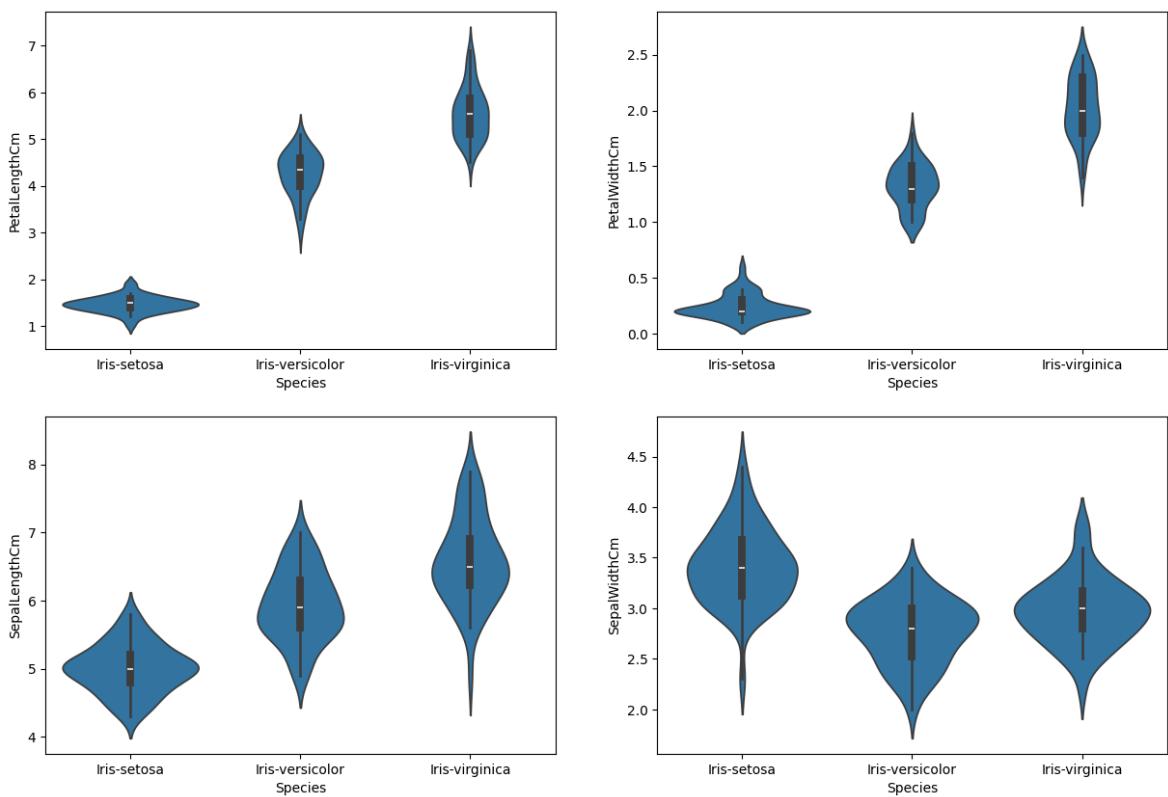
```
# Accessing the 3rd box (index 2)
boxtwo = ax.patches[2]
boxtwo.set_facecolor('yellow')
boxtwo.set_edgecolor('black')
plt.show()
```



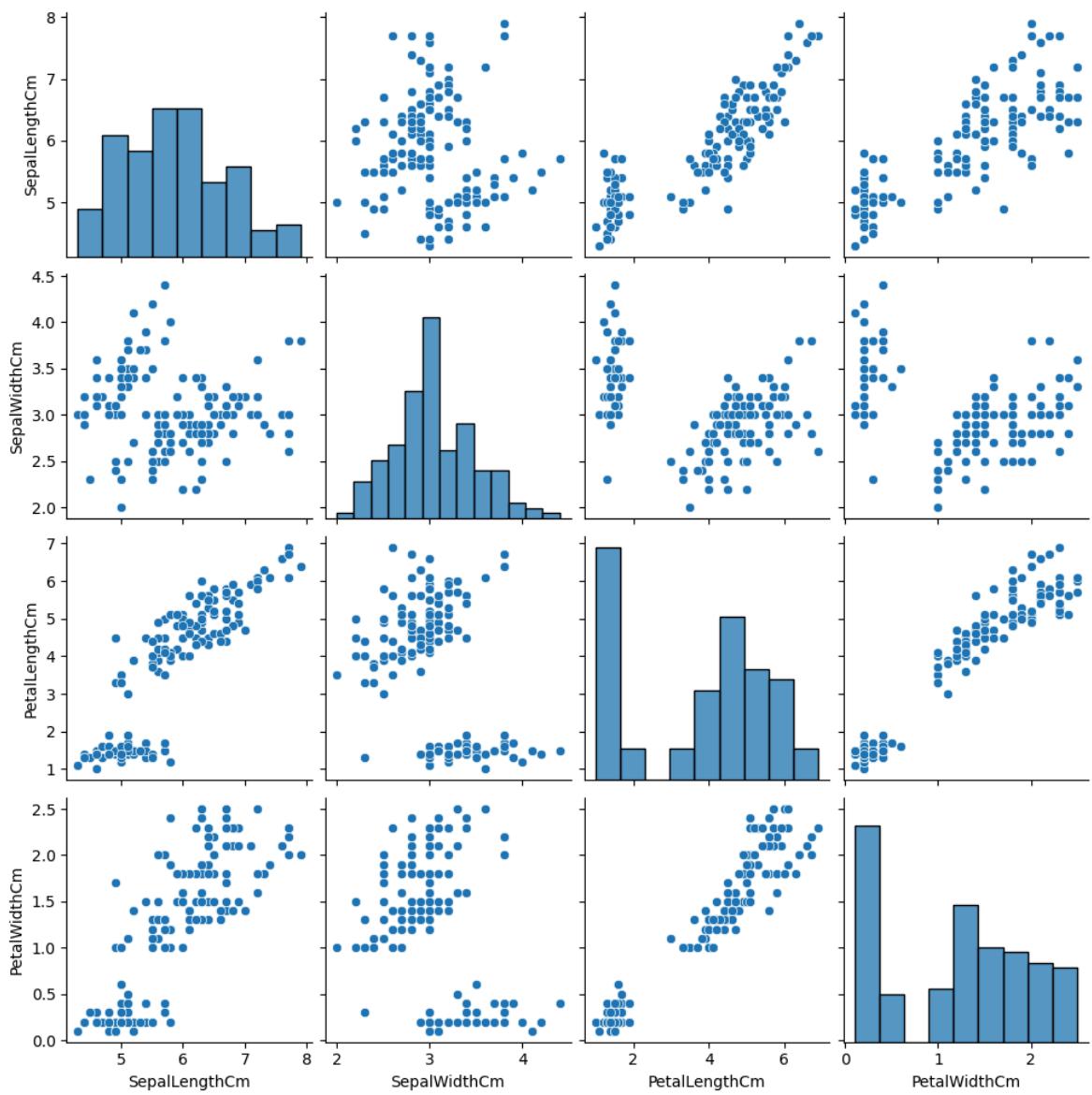
```
In [36]: fig=plt.gcf()
fig.set_size_inches(10,7)
fig=sns.violinplot(x='Species',y='SepalLengthCm',data=iris)
plt.show()
```



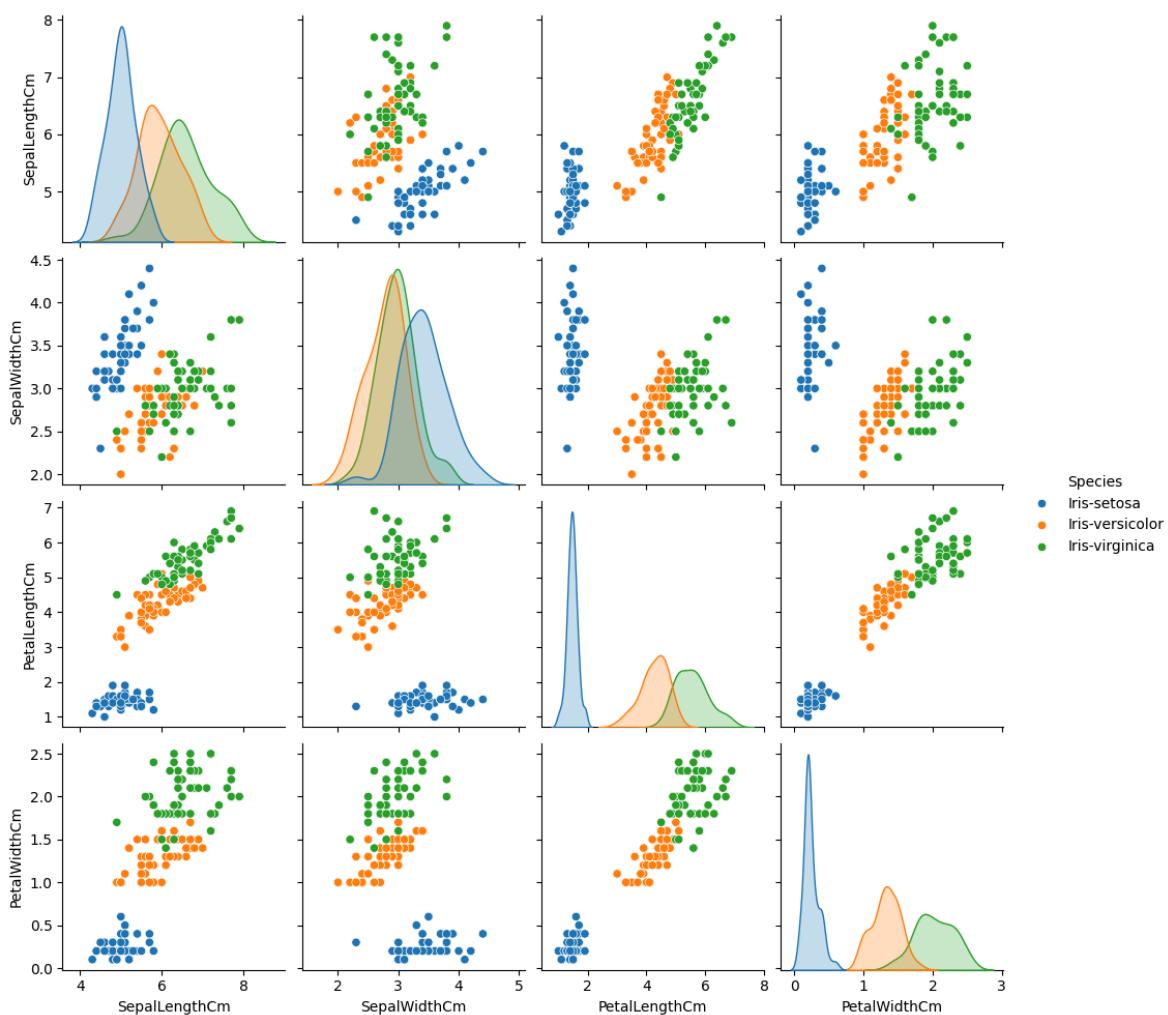
```
In [37]: plt.figure(figsize=(15,10))
plt.subplot(2,2,1)
sns.violinplot(x='Species',y='PetalLengthCm',data=iris)
plt.subplot(2,2,2)
sns.violinplot(x='Species',y='PetalWidthCm',data=iris)
plt.subplot(2,2,3)
sns.violinplot(x='Species',y='SepalLengthCm',data=iris)
plt.subplot(2,2,4)
sns.violinplot(x='Species',y='SepalWidthCm',data=iris)
plt.show()
```



```
In [38]: sns.pairplot(data=iris, kind='scatter')
plt.show()
```



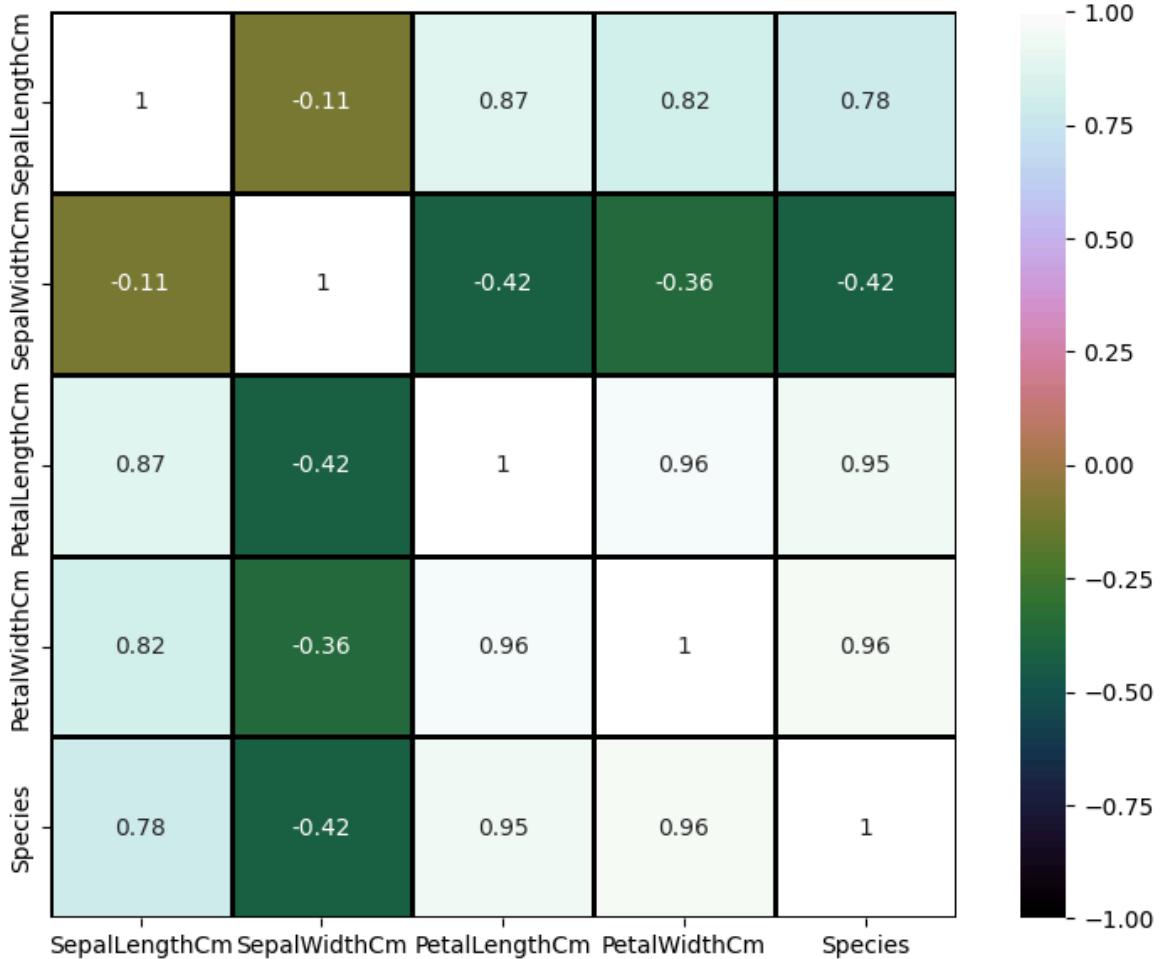
```
In [39]: sns.pairplot(iris,hue='Species');
plt.show()
```



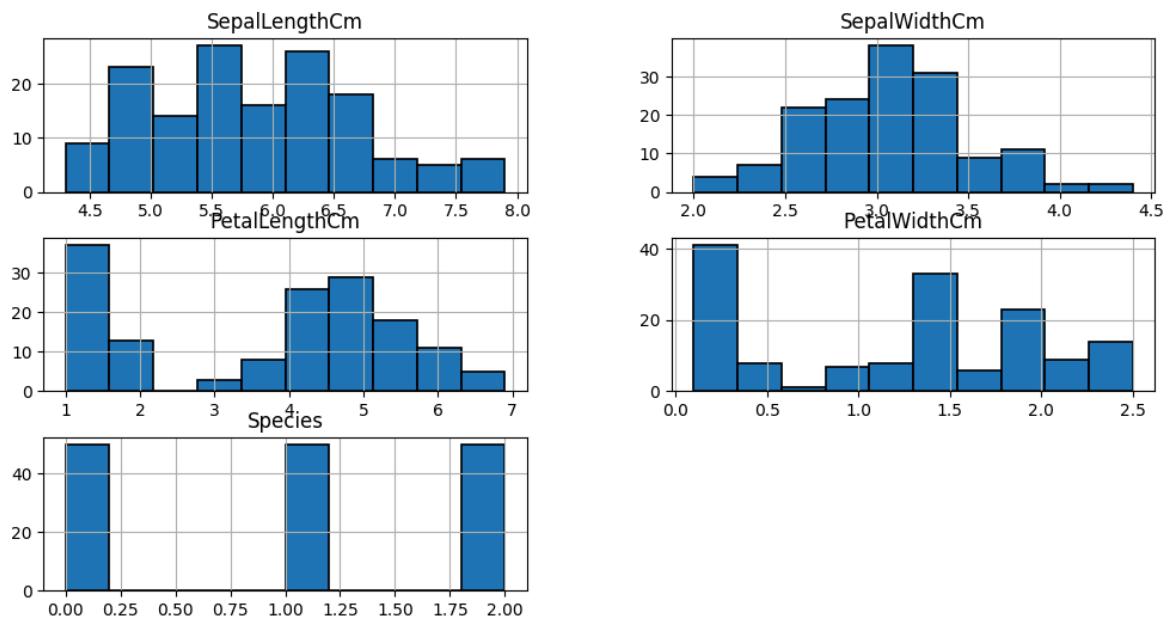
```
In [41]: from sklearn.preprocessing import LabelEncoder

le = LabelEncoder()
iris['Species'] = le.fit_transform(iris['Species'])

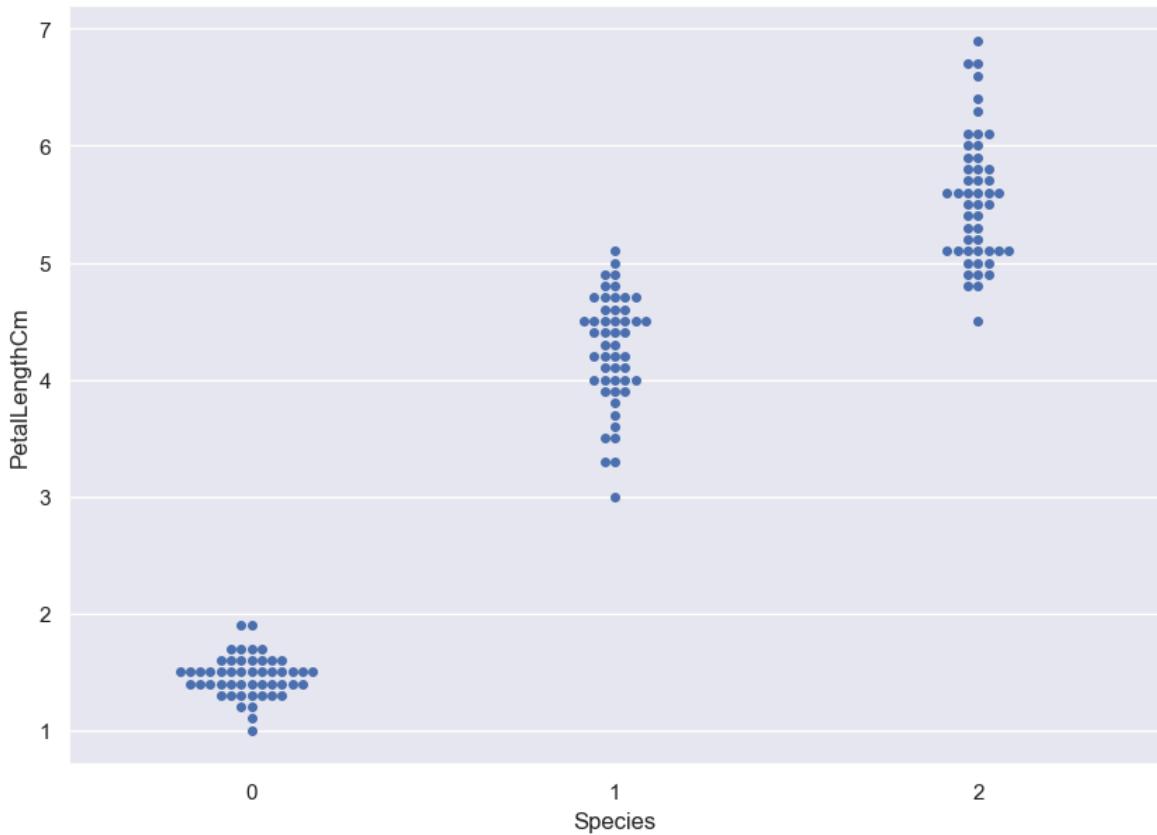
fig = plt.gcf()
fig.set_size_inches(10, 7)
sns.heatmap(iris.corr(), annot=True, cmap='cubehelix',
            linewidths=1, linecolor='k', square=True,
            vmin=-1, vmax=1, cbar_kws={"orientation": "vertical"})
plt.show()
```



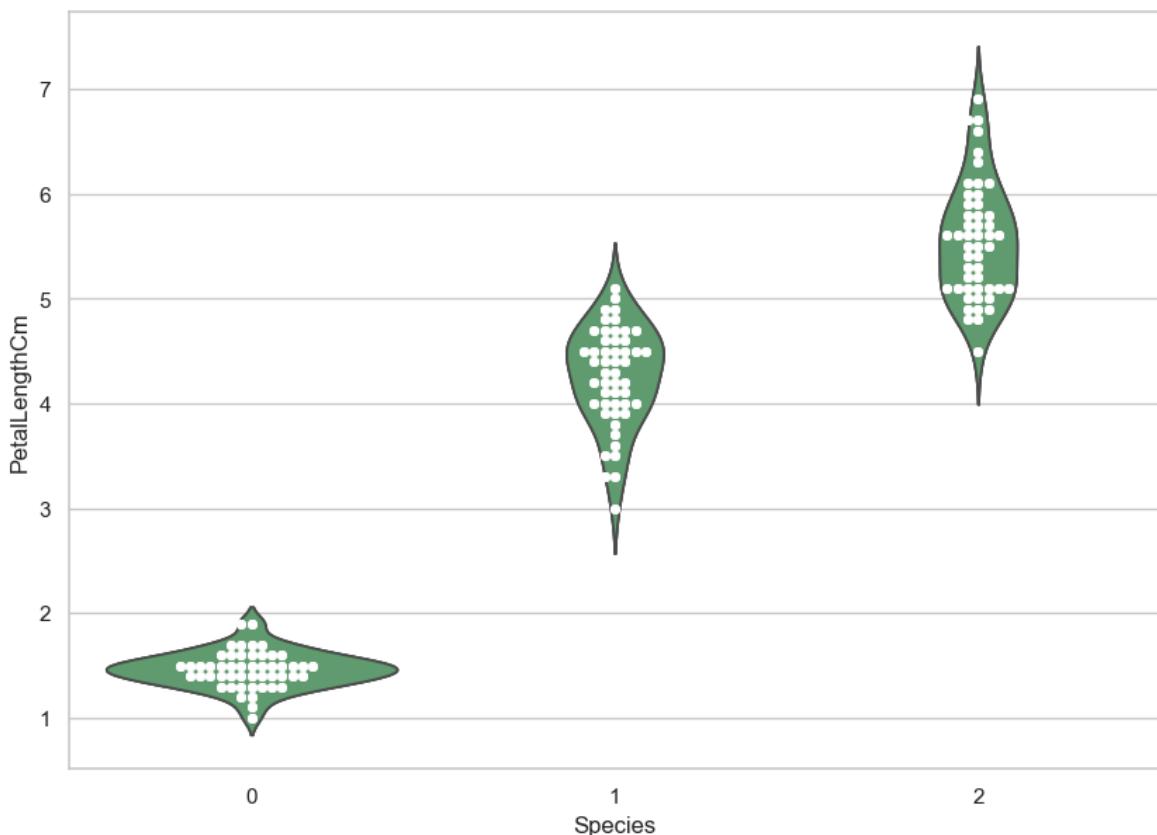
```
In [42]: iris.hist(edgecolor='black', linewidth=1.2)
fig=plt.gcf()
fig.set_size_inches(12,6)
plt.show()
```



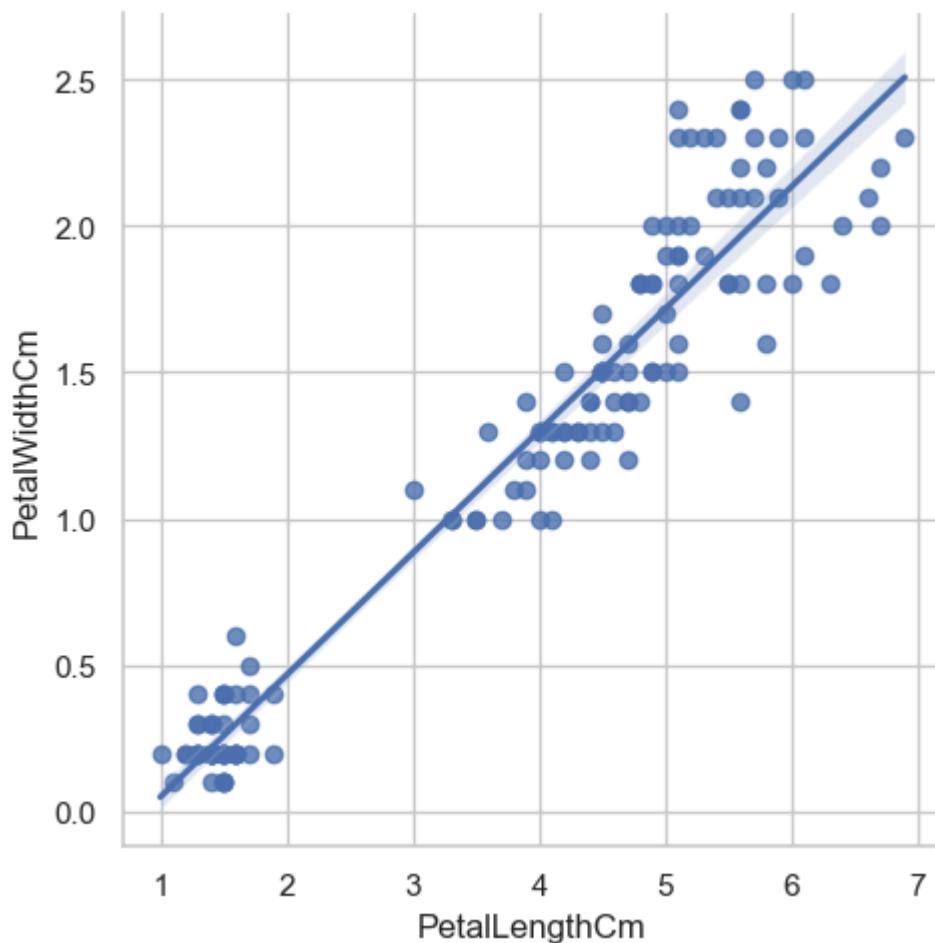
```
In [43]: sns.set(style="darkgrid")
fig=plt.gcf()
fig.set_size_inches(10,7)
fig = sns.swarmplot(x="Species", y="PetalLengthCm", data=iris)
plt.show()
```

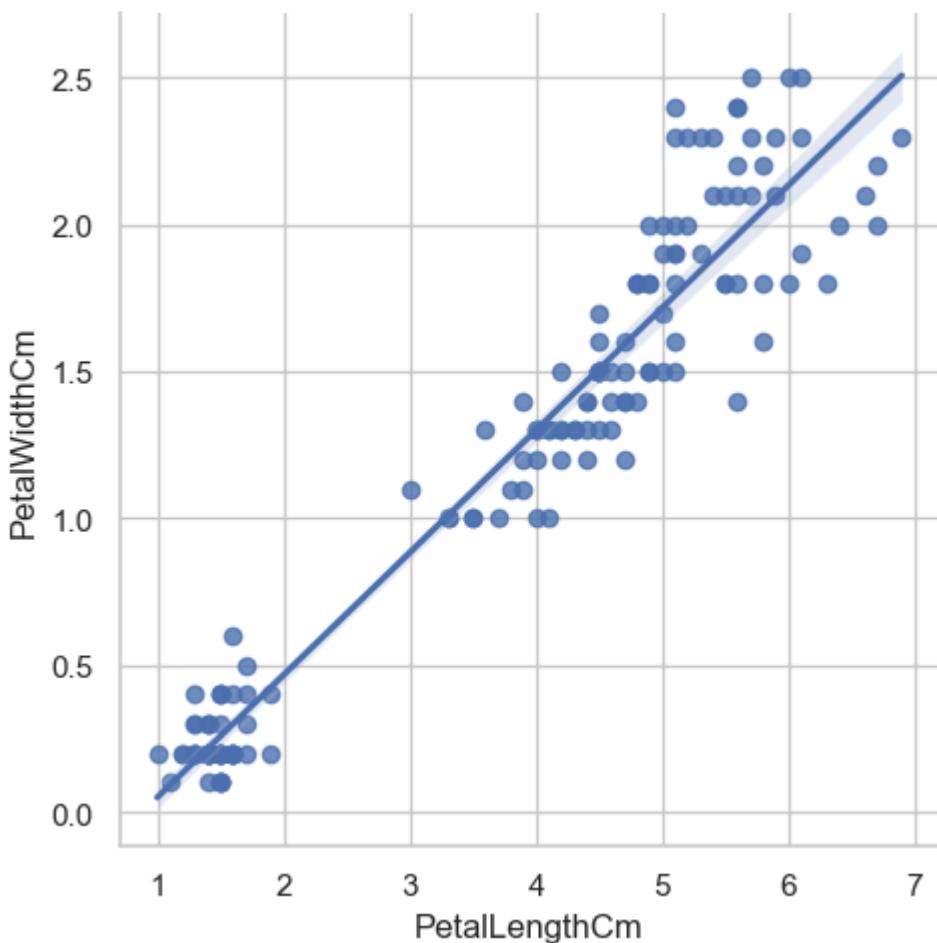


```
In [46]: sns.set(style="whitegrid")
fig=plt.gcf()
fig.set_size_inches(10,7)
ax = sns.violinplot(x="Species", y="PetalLengthCm", data=iris, inner=None)
ax = sns.swarmplot(x="Species", y="PetalLengthCm", data=iris,color="white", edgecolor="black")
plt.show()
```



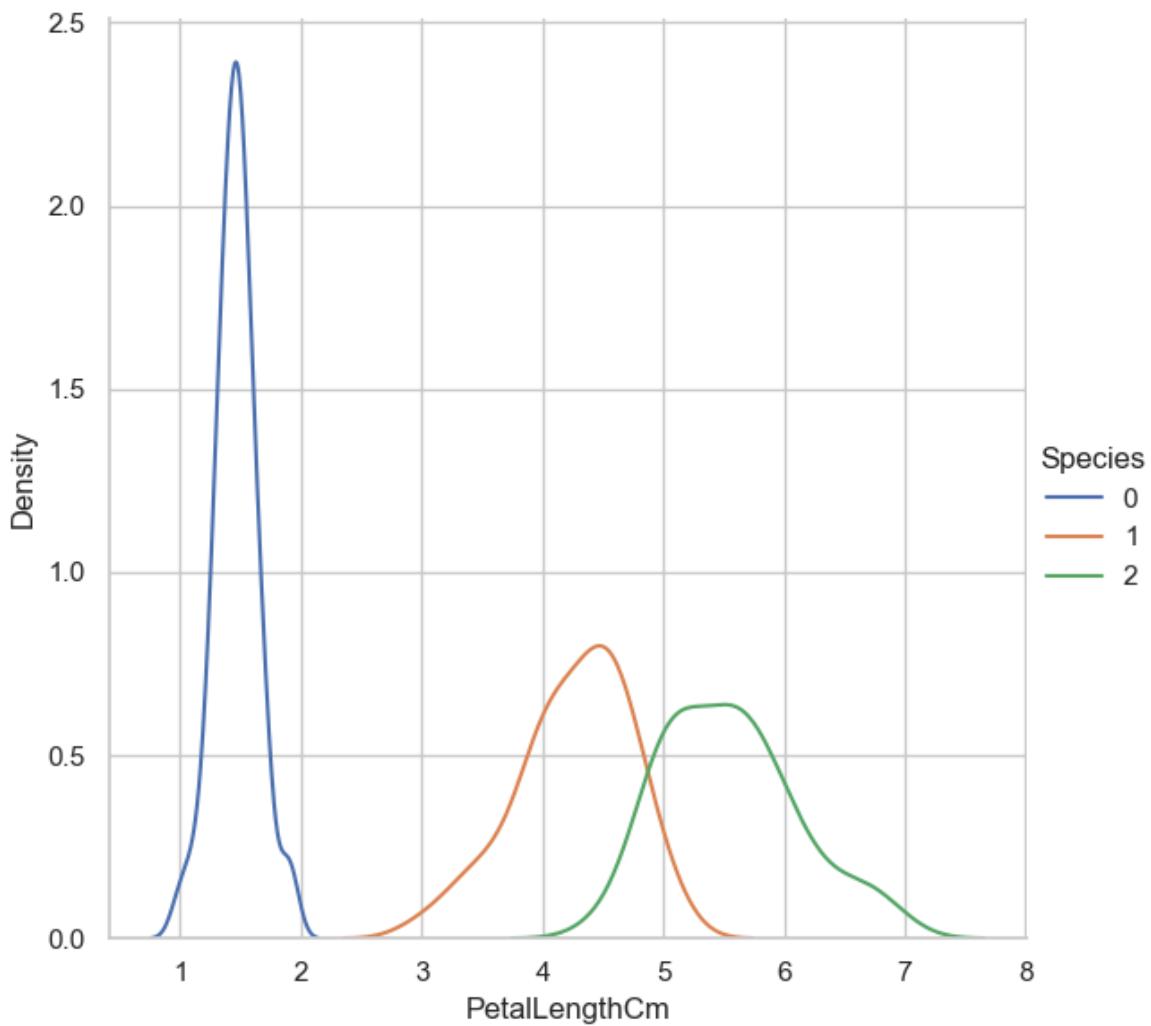
```
In [48]: fig=sns.lmplot(x="PetalLengthCm", y="PetalWidthCm", data=iris)
plt.show()
```



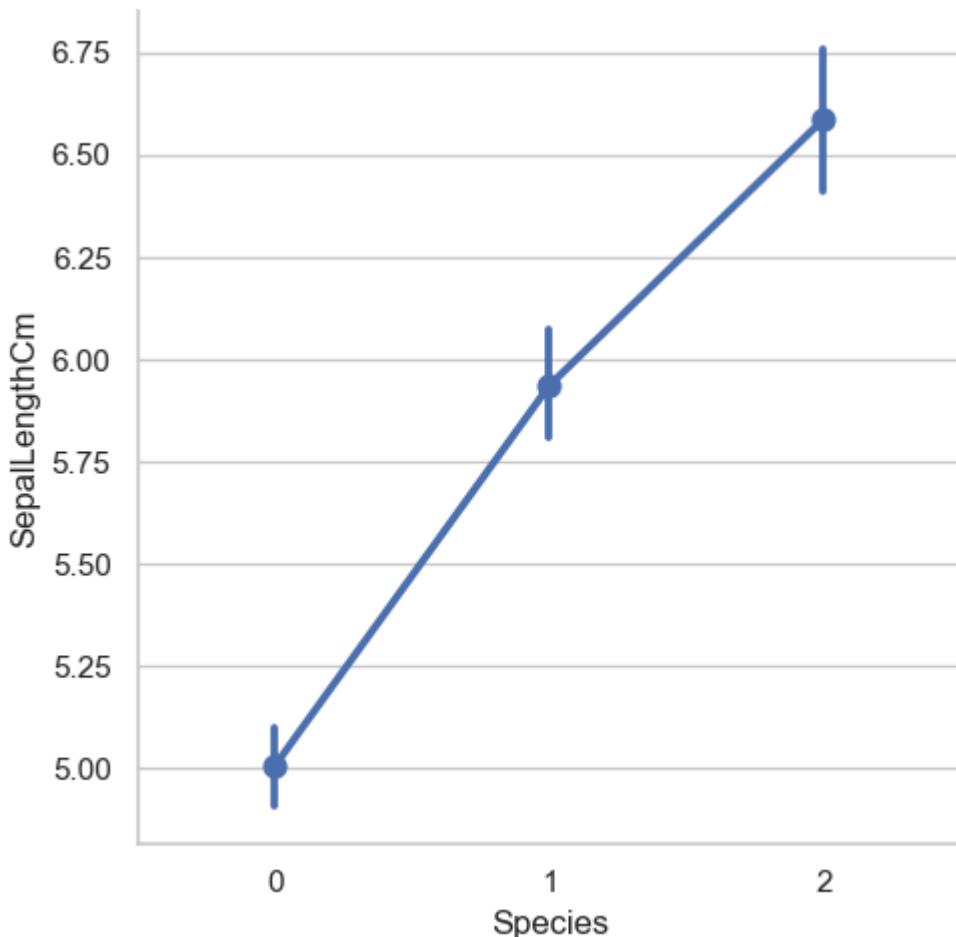


```
In [51]: sns.FacetGrid(iris, hue="Species", height=6) \
    .map(sns.kdeplot, "PetalLengthCm") \
    .add_legend()

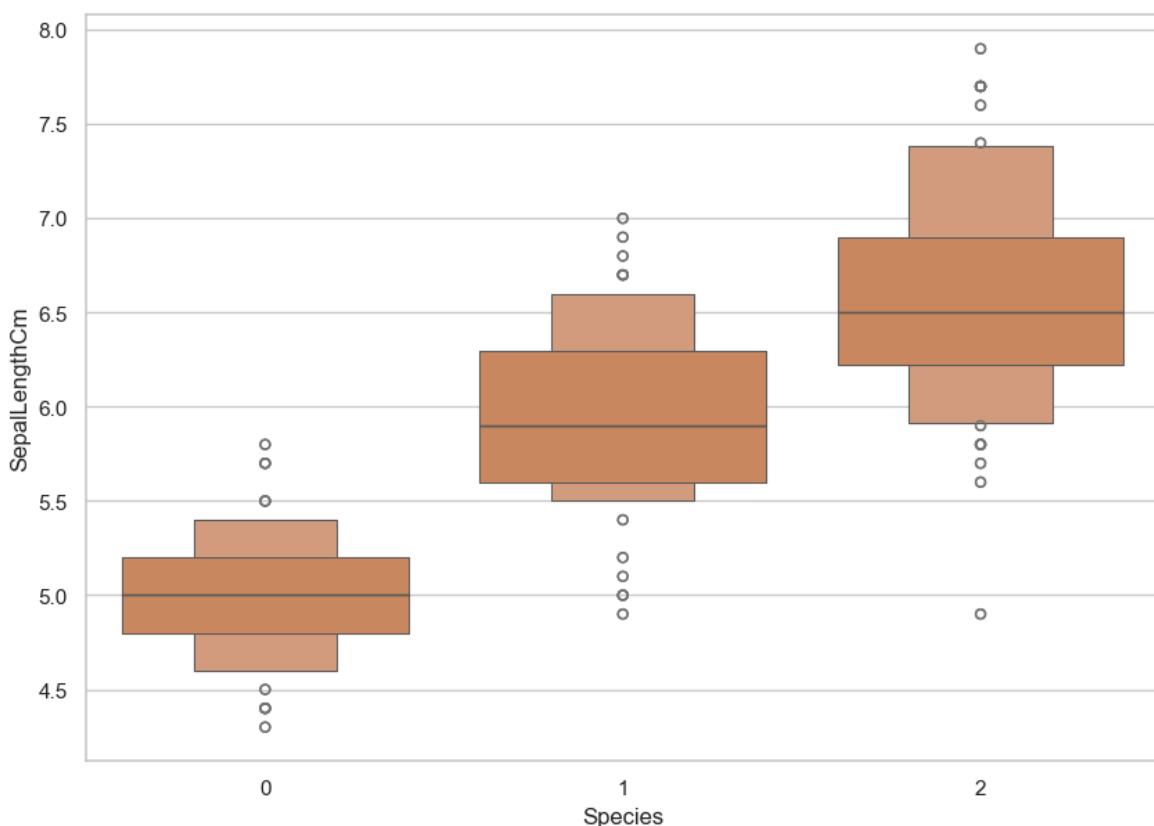
plt.show()
```



```
In [53]: sns.catplot(x='Species', y='SepalLengthCm', data=iris, kind='point')
plt.show()
```



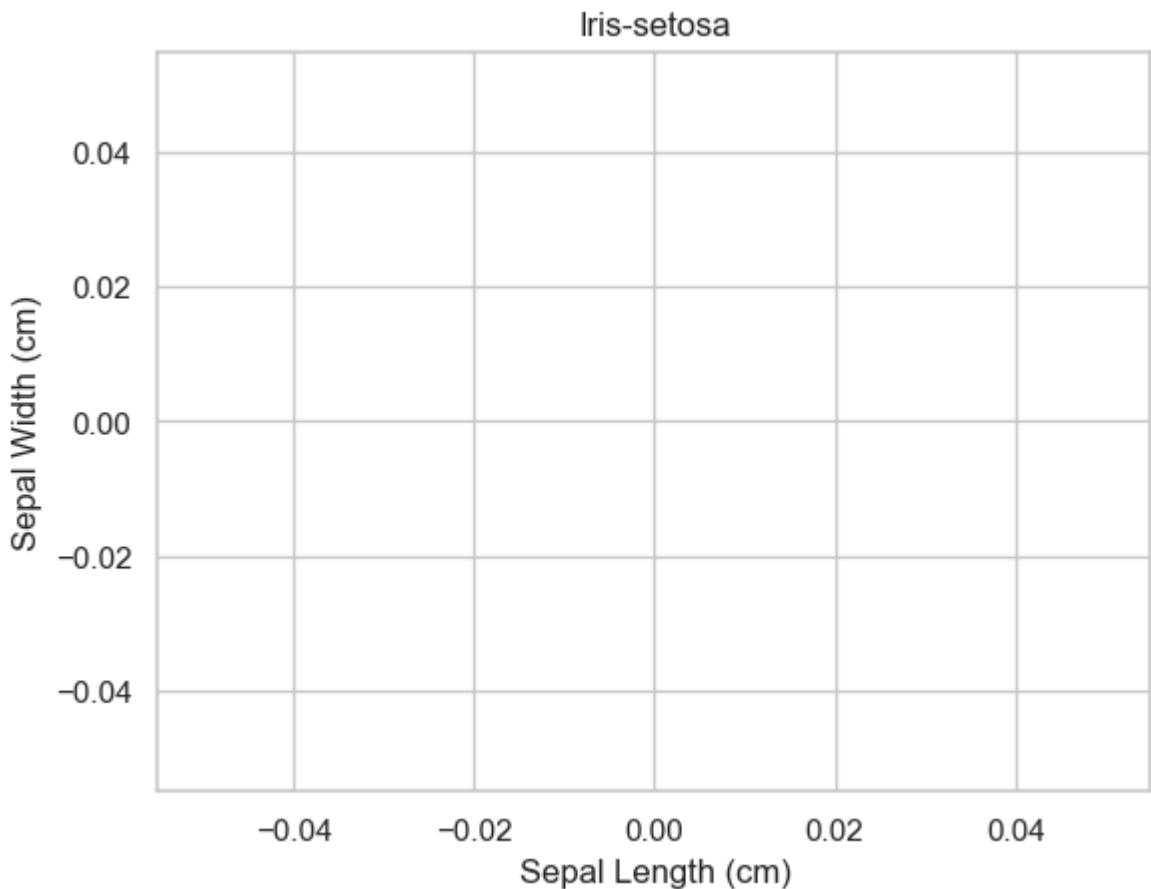
```
In [55]: fig=plt.gcf()
fig.set_size_inches(10,7)
fig=sns.boxenplot(x='Species',y='SepalLengthCm',data=iris)
plt.show()
```



```
In [60]: sub = iris[iris['Species'] == 'Iris-setosa']

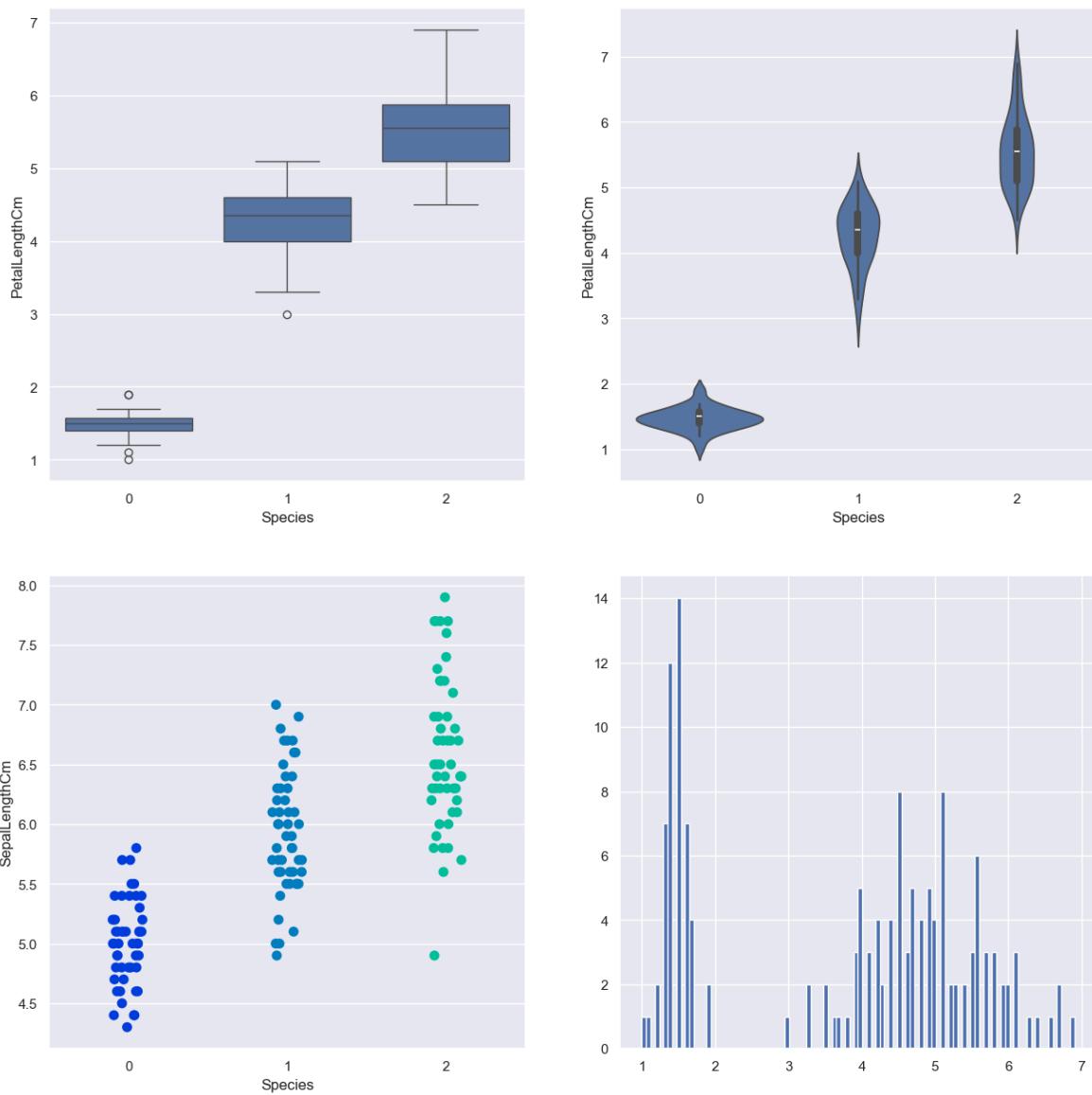
sns.kdeplot(
    data=sub,
    x="SepalLengthCm",
    y="SepalWidthCm",
    fill=True,      # replaces shade=True
    cmap="plasma",
    thresh=0        # replaces shade_lowest=False
)

plt.title('Iris-setosa')
plt.xlabel('Sepal Length (cm)')
plt.ylabel('Sepal Width (cm)')
plt.show()
```



```
In [61]: sns.set_style('darkgrid')
f,axes=plt.subplots(2,2,figsize=(15,15))

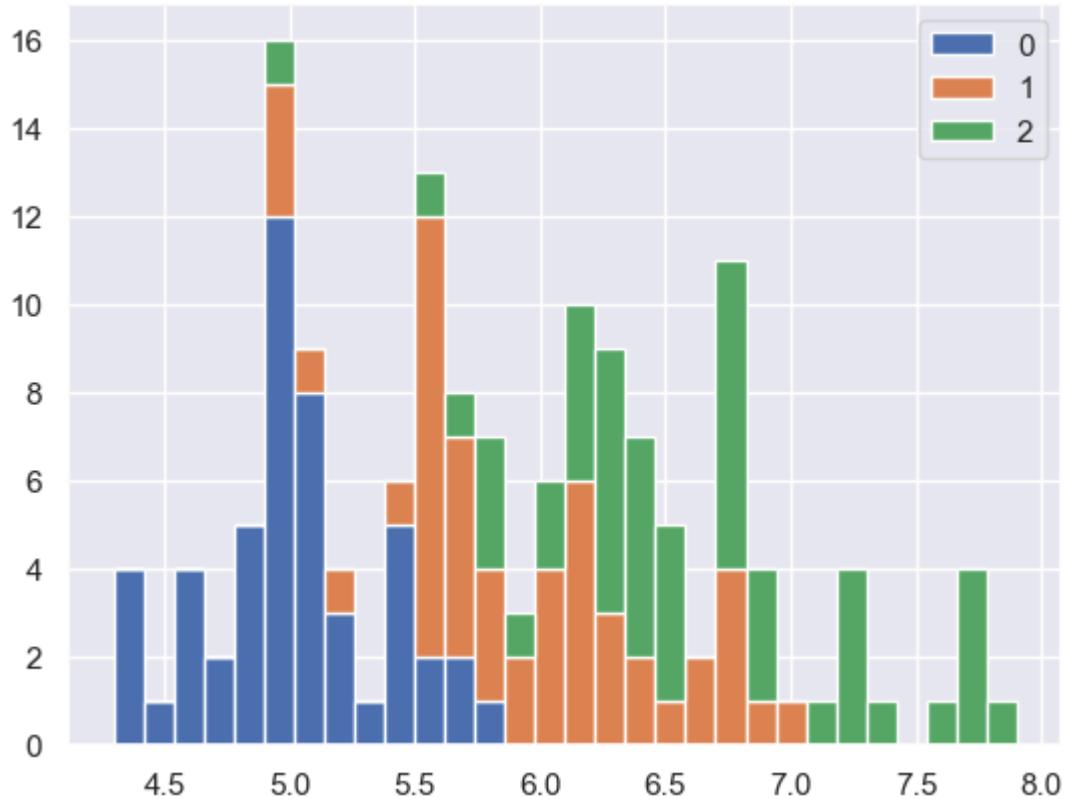
k1=sns.boxplot(x="Species", y="PetalLengthCm", data=iris,ax=axes[0,0])
k2=sns.violinplot(x='Species',y='PetalLengthCm',data=iris,ax=axes[0,1])
k3=sns.stripplot(x='Species',y='SepalLengthCm',data=iris,jitter=True,edgecolor='black',size=10)
#axes[1,1].hist(iris.hist,bins=10)
#k2.set(xlim=(-1,0.8))
plt.show()
```



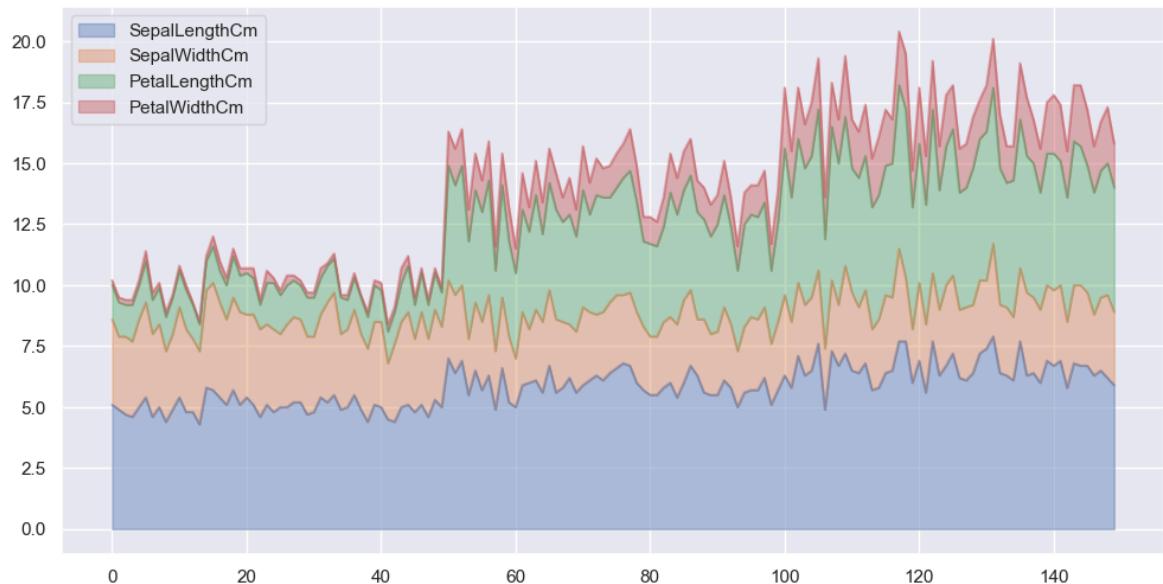
```
In [62]: iris['Species'] = iris['Species'].astype('category')
#iris.head()
plt.show()
```

```
In [63]: list1=list()
mylabels=list()
for gen in iris.Species.cat.categories:
    list1.append(iris[iris.Species==gen].SepalLengthCm)
    mylabels.append(gen)

h=plt.hist(list1,bins=30,stacked=True,rwidth=1,label=mylabels)
plt.legend()
plt.show()
```

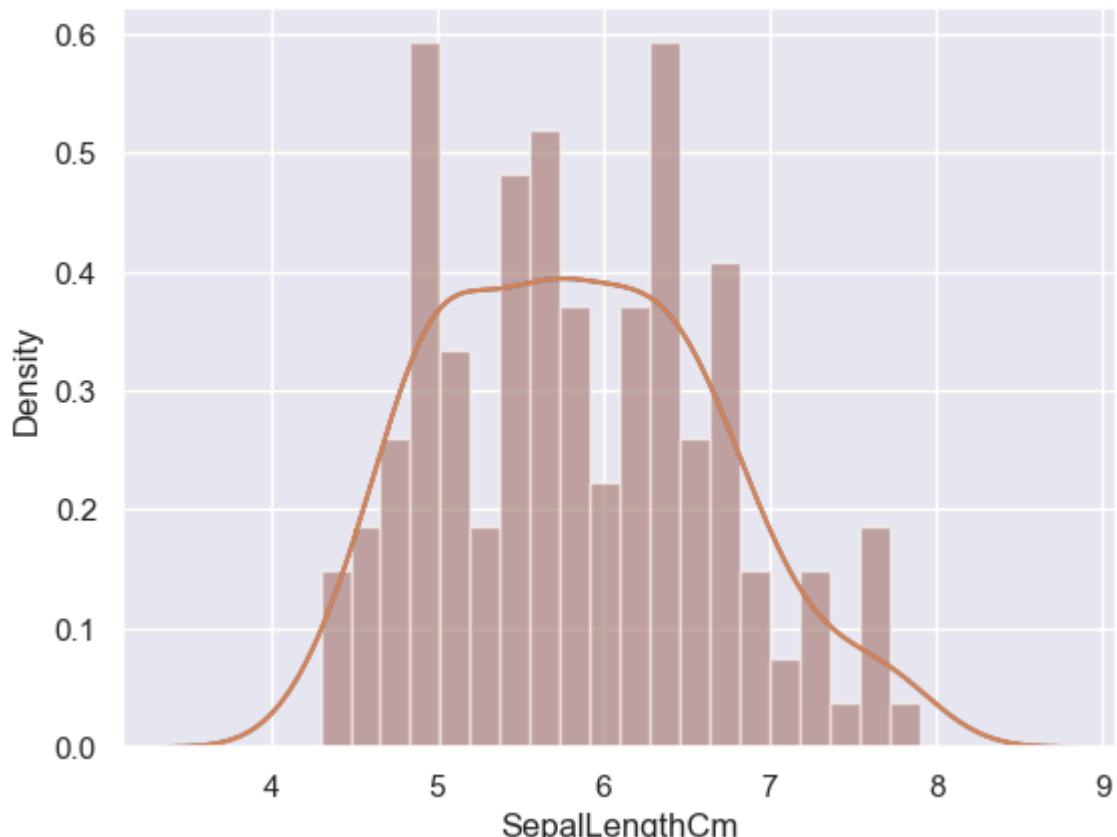


```
In [65]: #iris['SepalLengthCm'] = iris['SepalLengthCm'].astype('category')
#iris.head()
#iris.plot.area(y='SepalLengthCm',alpha=0.4,figsize=(12, 6));
iris.plot.area(y=['SepalLengthCm','SepalWidthCm','PetalLengthCm','PetalWidthCm'])
plt.show()
```





```
In [67]: sns.distplot(iris['SepalLengthCm'], kde=True, bins=20);  
plt.show()
```



THIS IS ALL ABOUT EDA COMPLETE

```
In [ ]:
```