

Iris Dataset Visualisation(Seaborn,Matplotlib)

```
In [1]: import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import warnings
warnings.filterwarnings('ignore')
```

```
In [2]: iris = pd.read_csv('iris.csv')
```

```
In [3]: iris
```

```
Out[3]:
```

	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 [4]: iris.head()
```

```
Out[4]:
```

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

```
In [6]: iris.head()
```

```
Out[6]:
```

	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

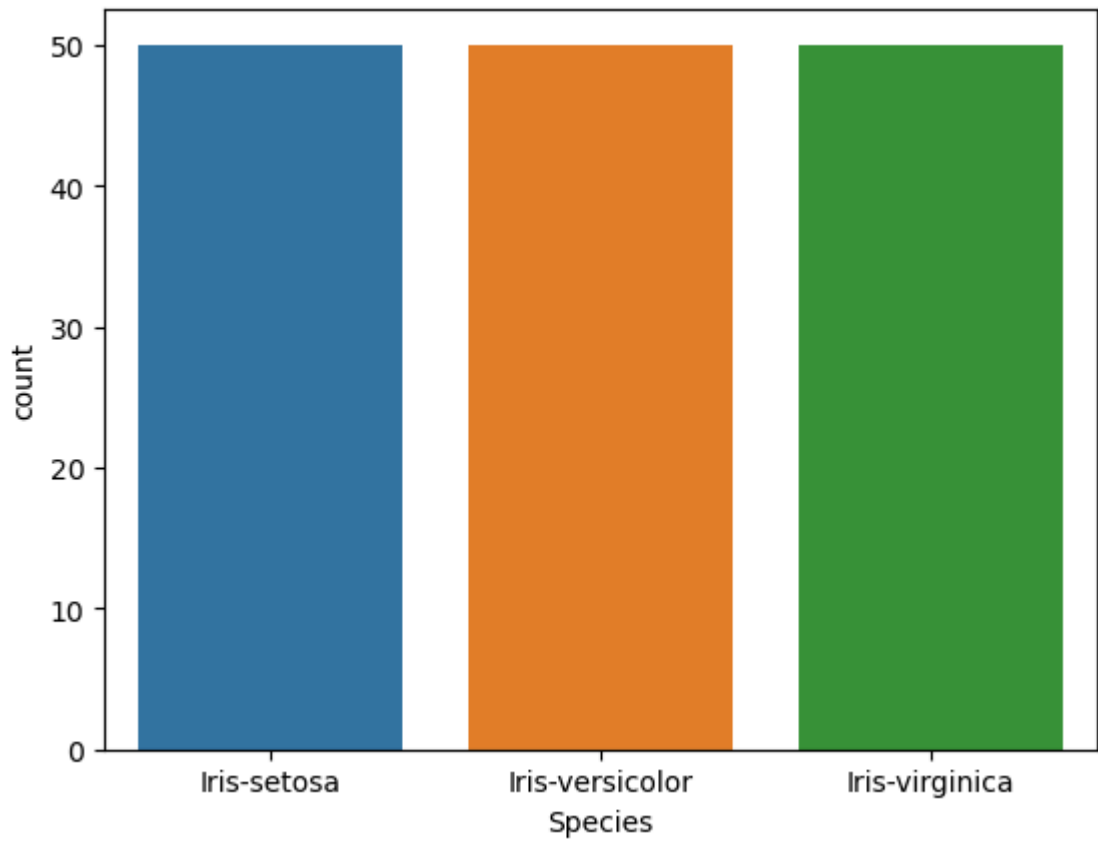
```
In [7]: iris.info()
```

```
<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 [8]: iris['Species'].value_counts()
```

```
Out[8]: Species
Iris-setosa      50
Iris-versicolor  50
Iris-virginica   50
Name: count, dtype: int64
```

```
In [9]: # Bar plot
sns.countplot(x='Species',data=iris)
plt.show()
```



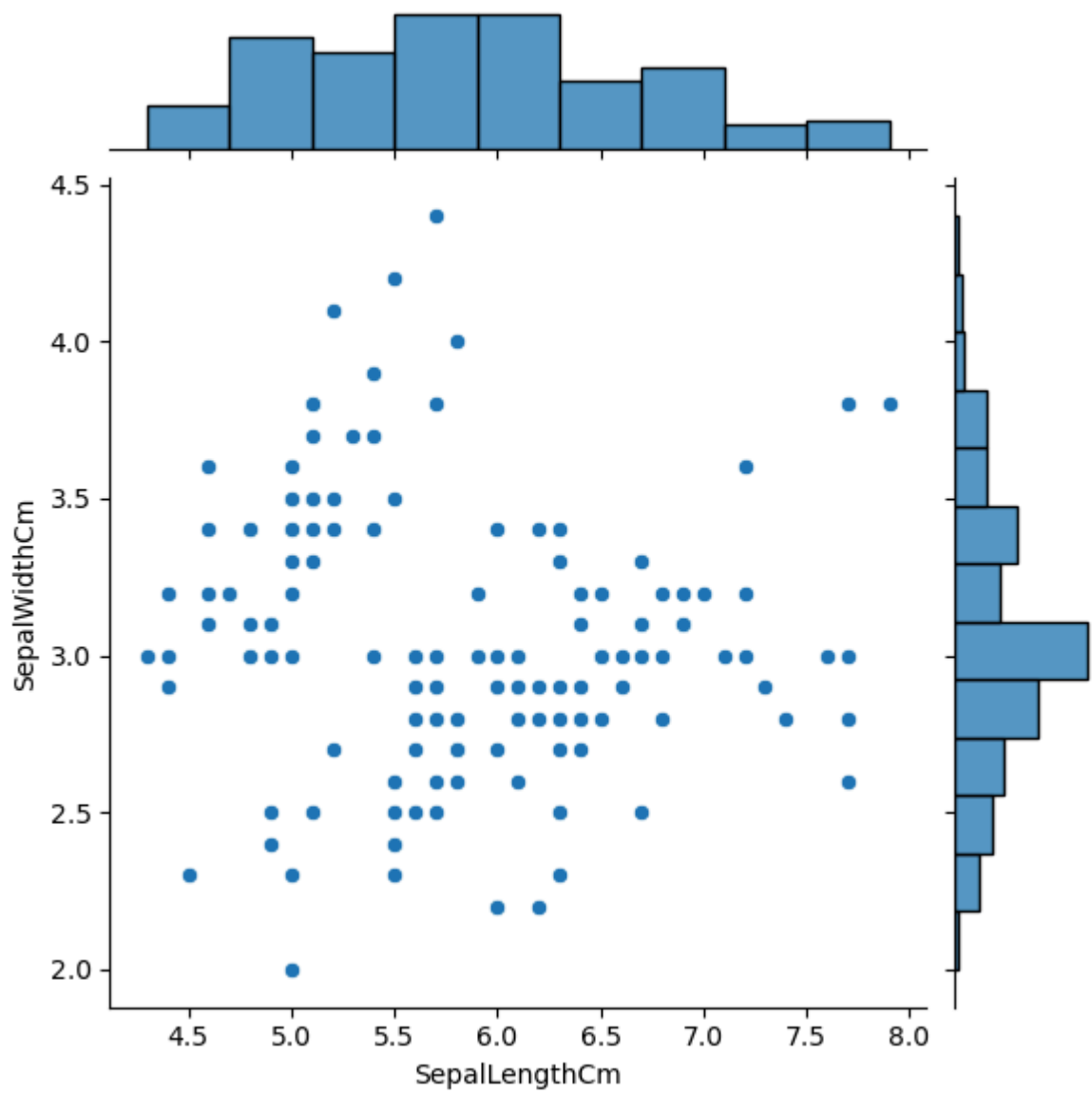
```
In [10]: # Joint plot
```

```
iris.head()
```

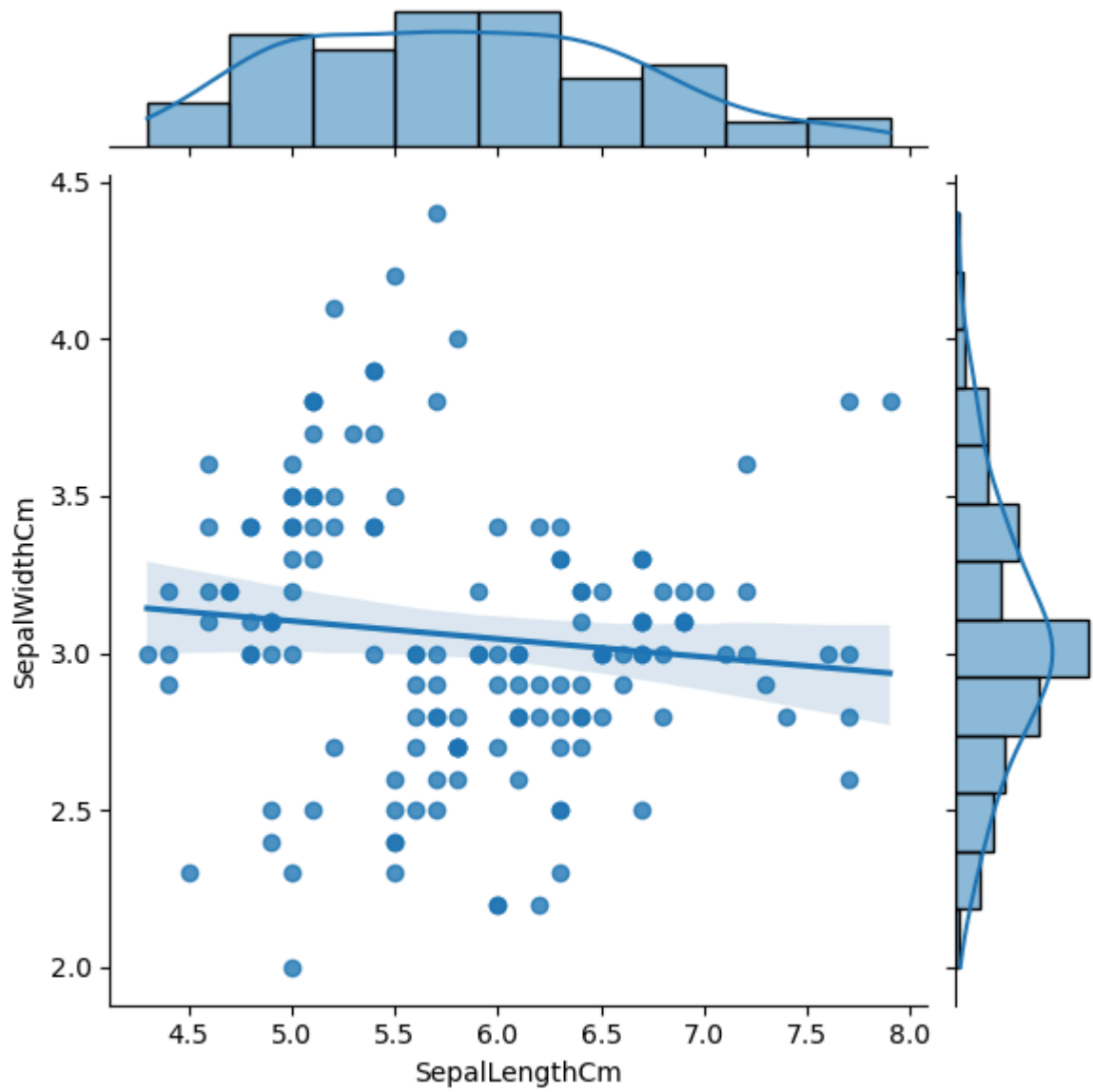
```
Out[10]:
```

	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

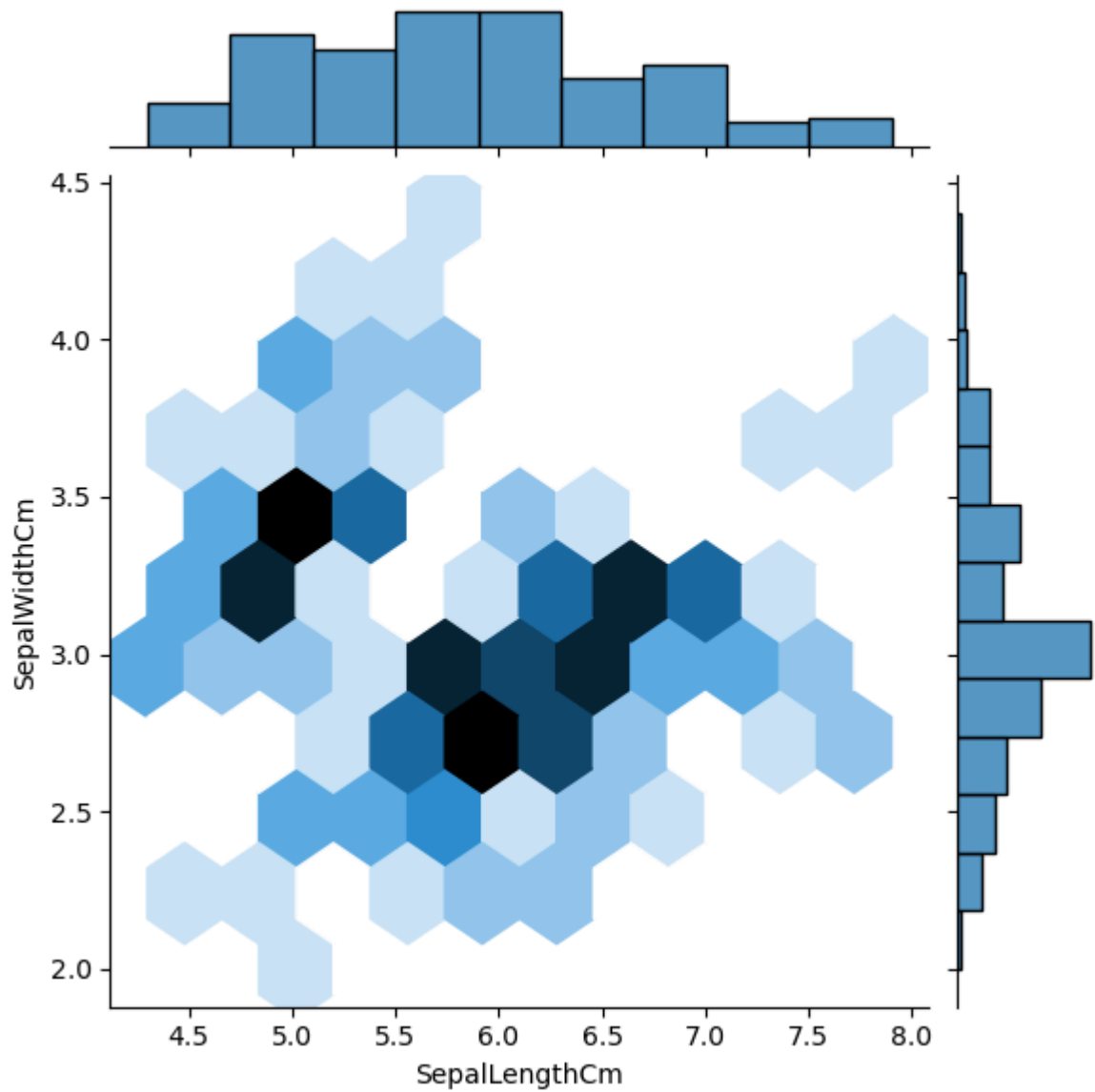
```
In [11]: fig = sns.jointplot(x='SepalLengthCm',y='SepalWidthCm',data=iris)
```



```
In [12]: fig1 = sns.jointplot(x='SepalLengthCm',y='SepalWidthCm',data=iris, kind="reg")
```

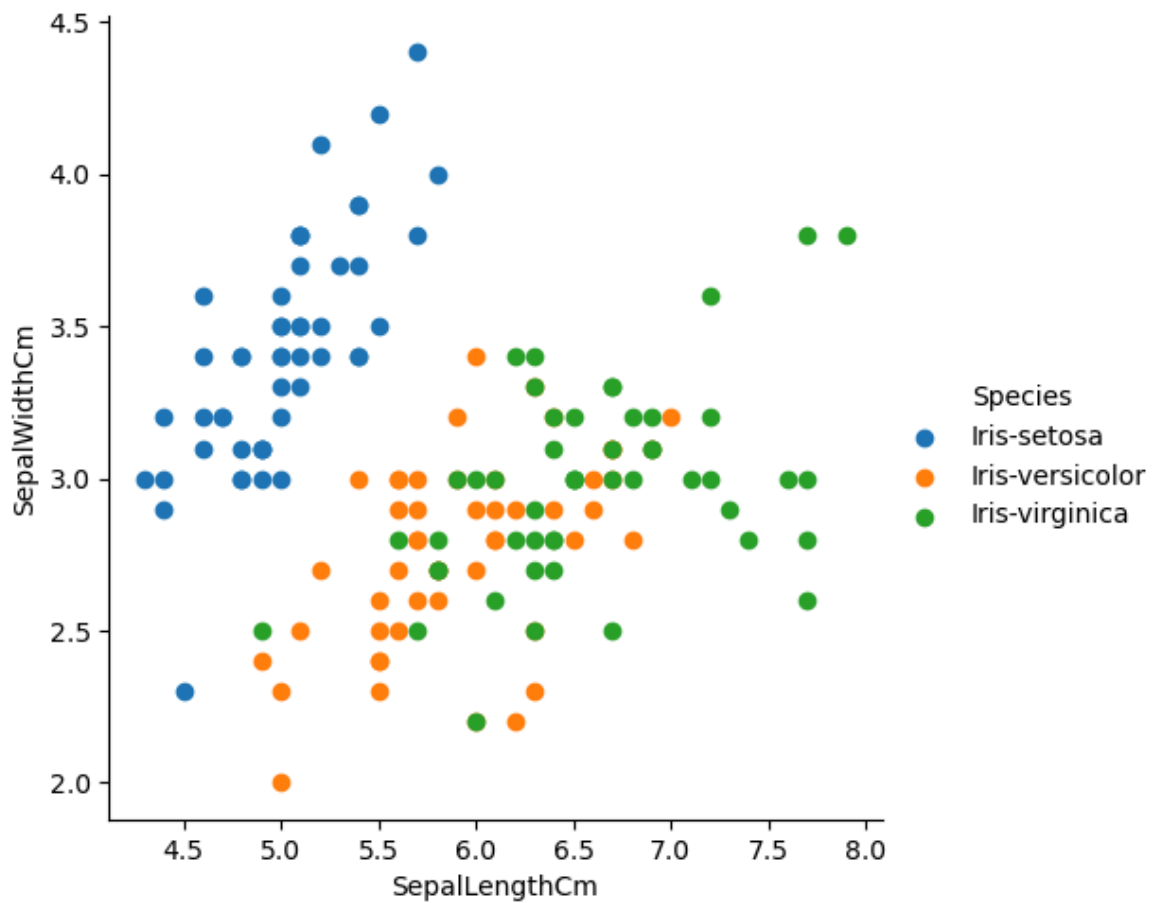


```
In [13]: fig2=sns.jointplot(x='SepalLengthCm',y='SepalWidthCm',kind='hex',data=iris)
```



```
In [14]: #FacetGrid Plot
sns.FacetGrid(iris,hue='Species',height=5)\
.map(plt.scatter,'SepalLengthCm','SepalWidthCm')\
.add_legend()
```

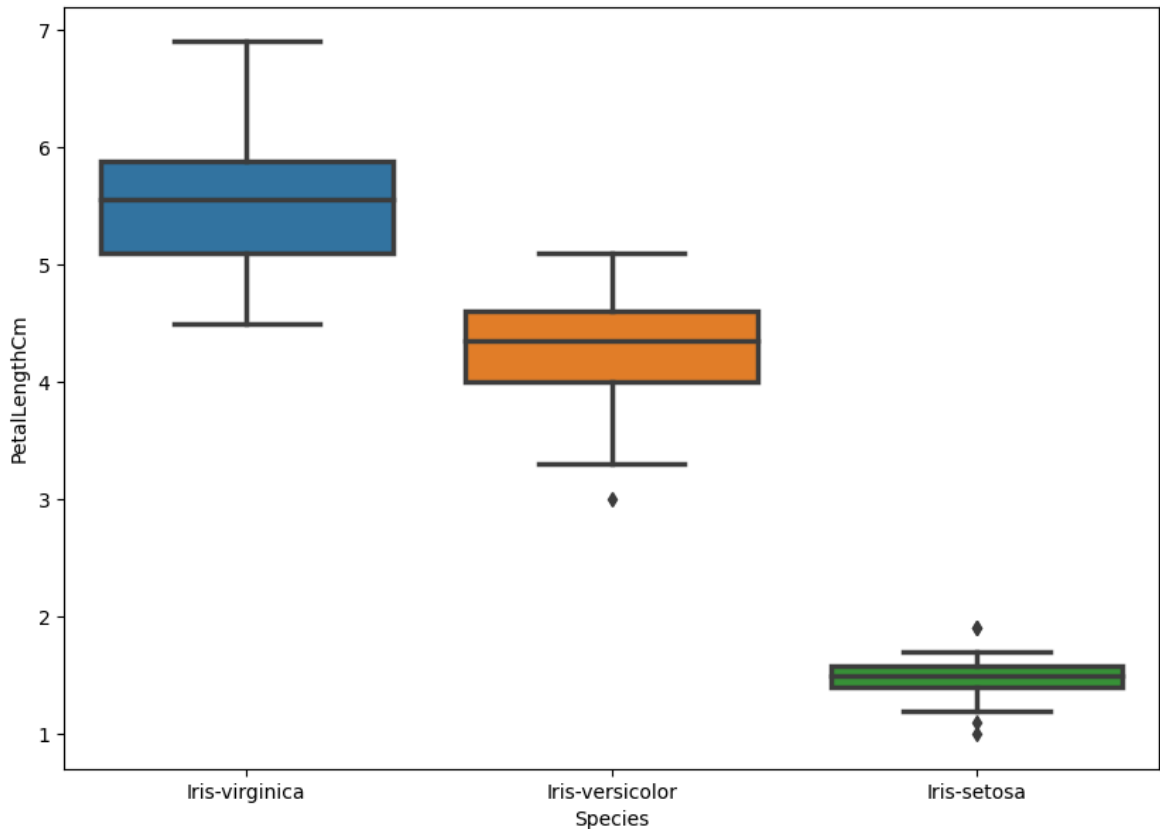
```
Out[14]: <seaborn.axisgrid.FacetGrid at 0x1a022e89350>
```



```
In [15]: # Box Plot
iris.head()
```

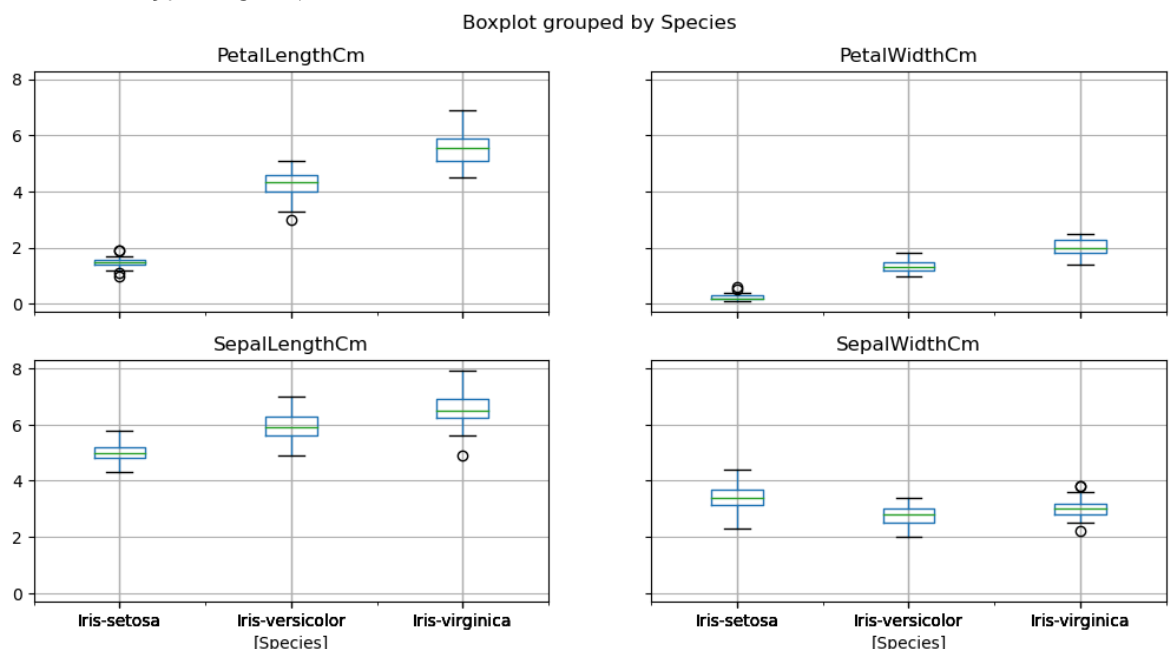
	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	5.1	3.5	1.4	0.2	Iris-setosa
1	4.9	3.0	1.4	0.2	Iris-setosa
2	4.7	3.2	1.3	0.2	Iris-setosa
3	4.6	3.1	1.5	0.2	Iris-setosa
4	5.0	3.6	1.4	0.2	Iris-setosa

```
In [16]: fig=plt.gcf()
fig.set_size_inches(10,7)
fig=sns.boxplot(x='Species',y='PetalLengthCm',data=iris,order=['Iris-virginica',
```

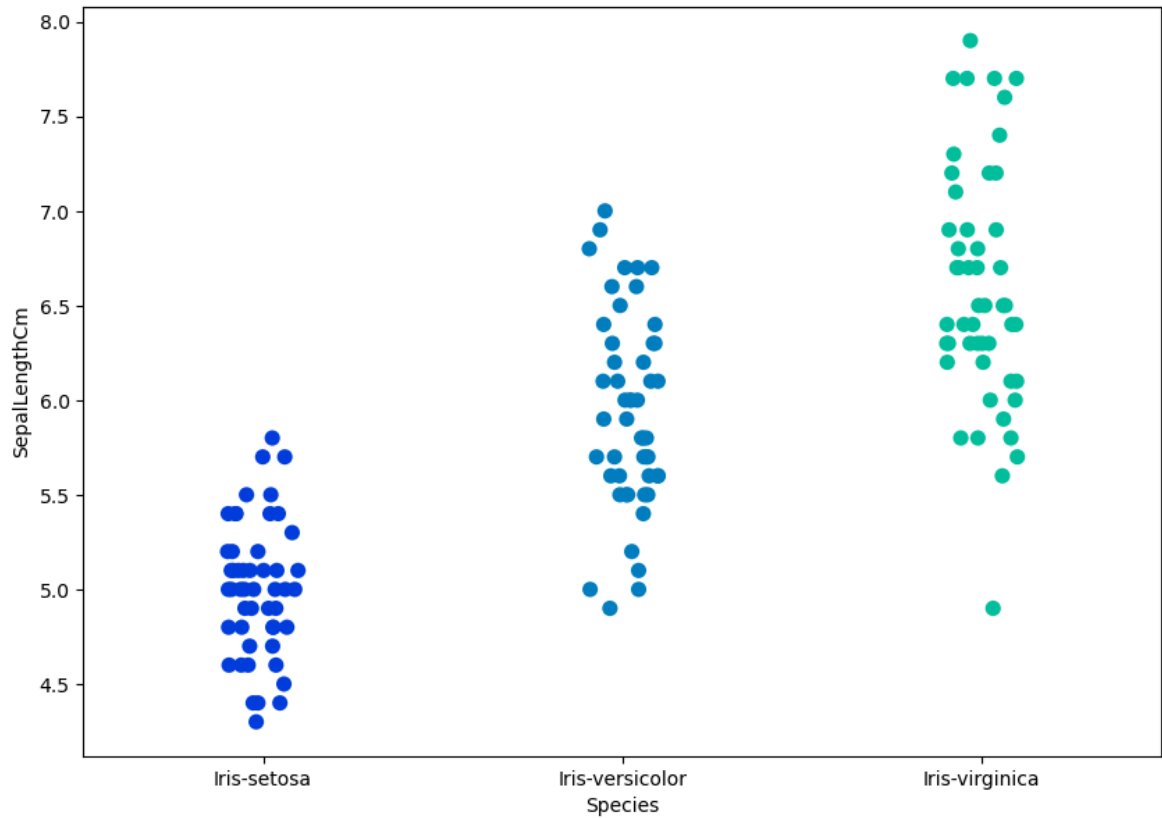


```
In [17]: iris.boxplot(by="Species", figsize=(12, 6))
```

```
Out[17]: 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)
```

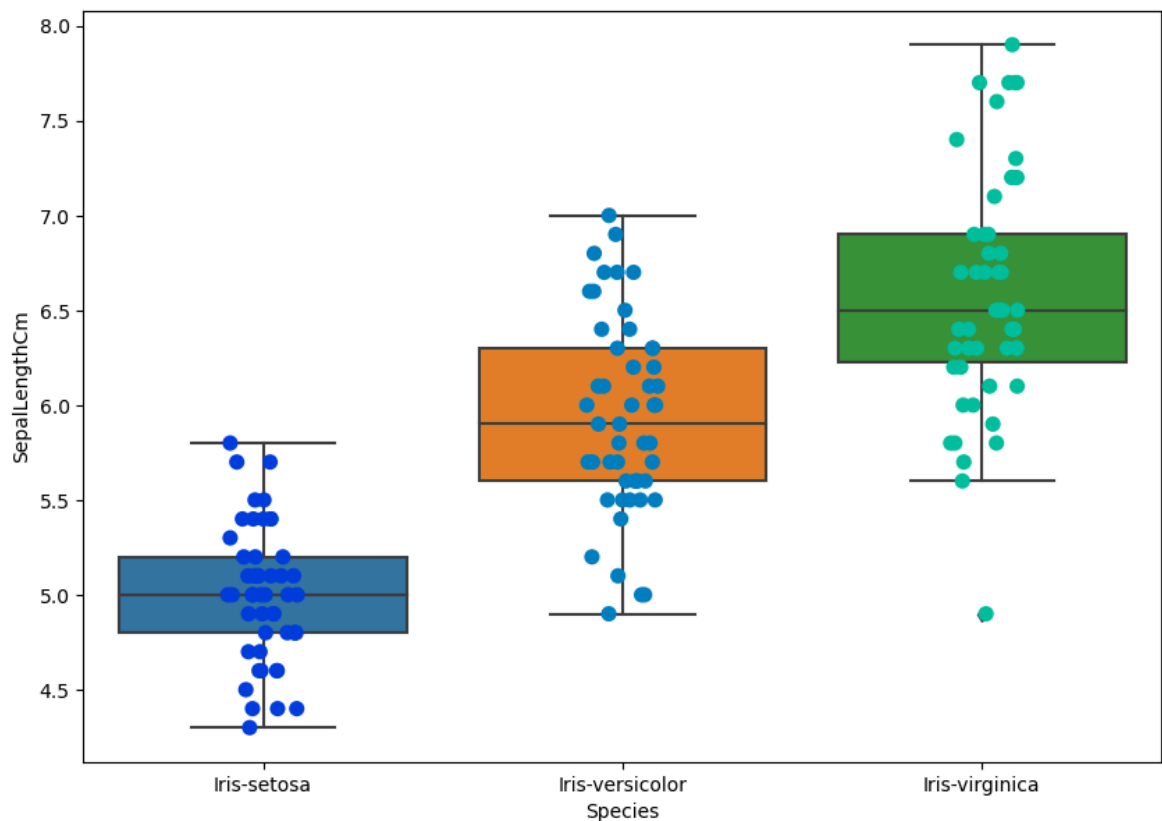


```
In [18]: # strip plot
fig=plt.gcf()
fig.set_size_inches(10,7)
fig=sns.stripplot(x='Species',y='SepalLengthCm',data=iris,jitter=True,edgecolor=
```

In [19]: *# Combining Box and Strip Plot*

```
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=
```



```
ax= sns.boxplot(x="Species", y="PetalLengthCm", data=iris)
ax= sns.stripplot(x="Species", y="PetalLengthCm", data=iris, jitter=True, edgeco
```

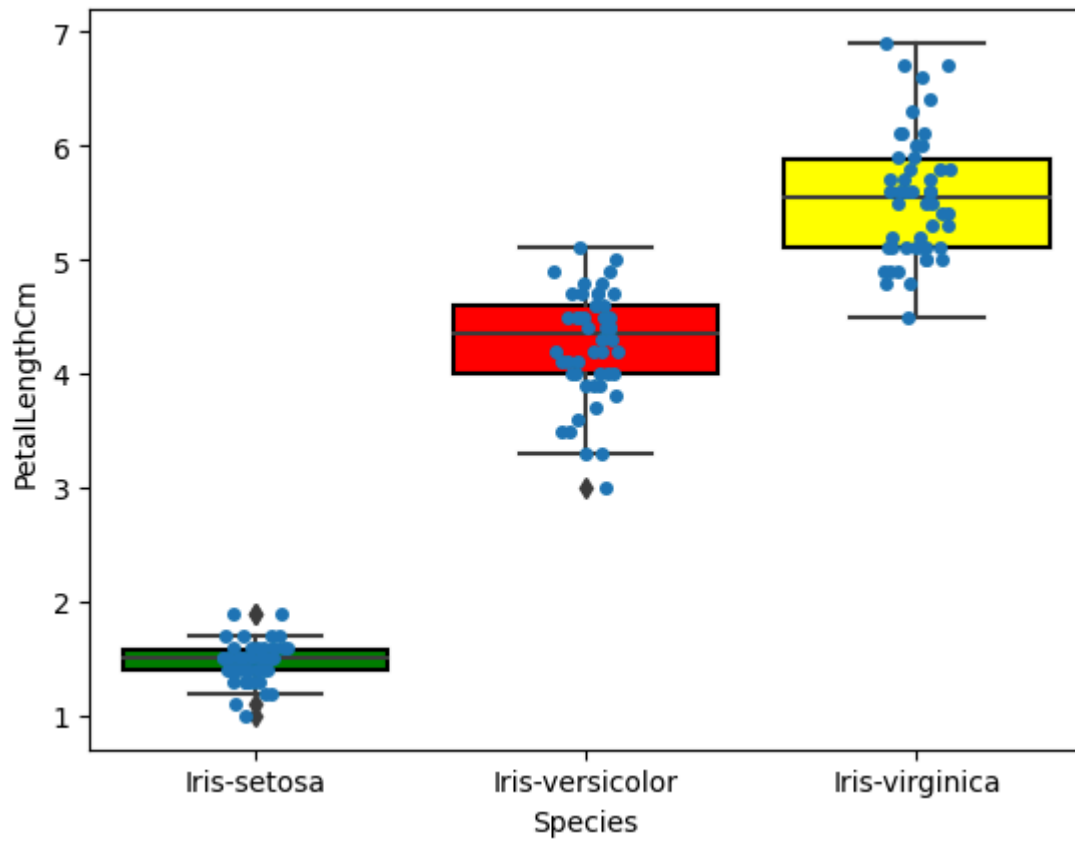
```

colors = ['green', 'red', 'yellow']

for i, patch in enumerate(ax.patches[:3]): # Only the first 3 patches are boxes
    patch.set_facecolor(colors[i])
    patch.set_edgecolor('black')

plt.show()

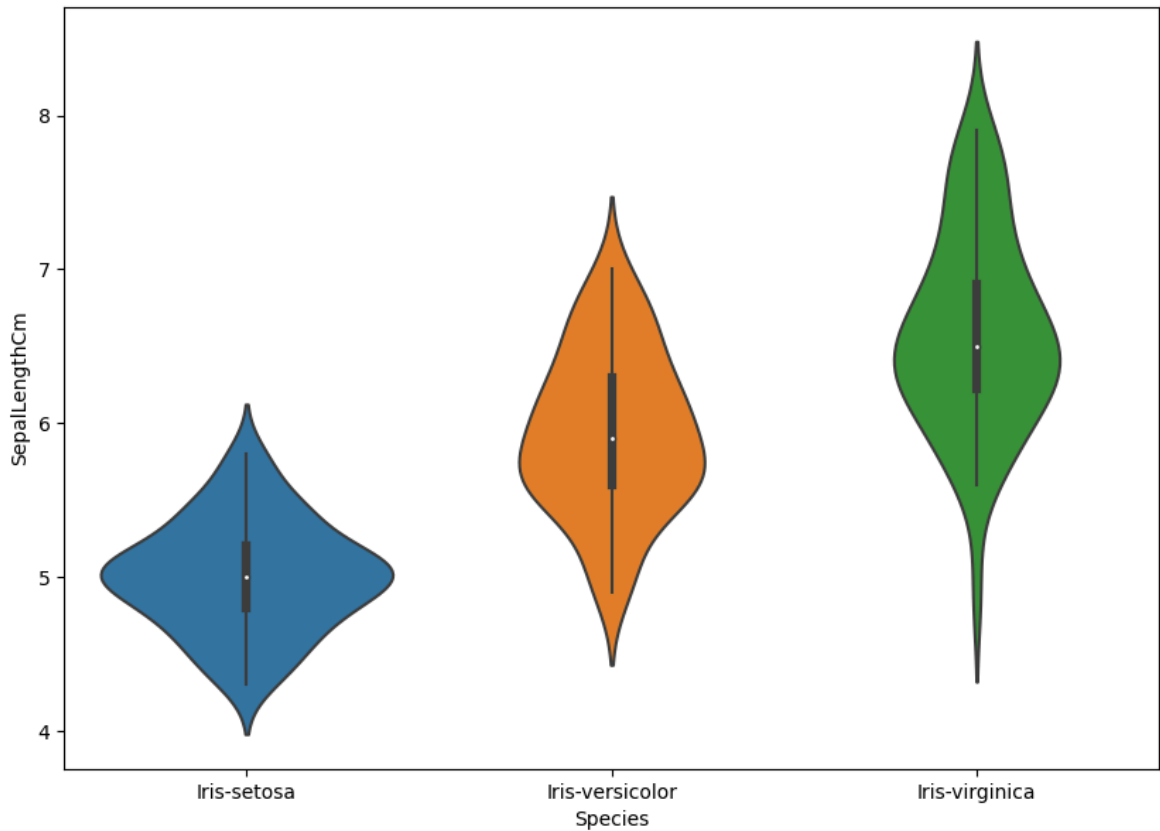
```



```

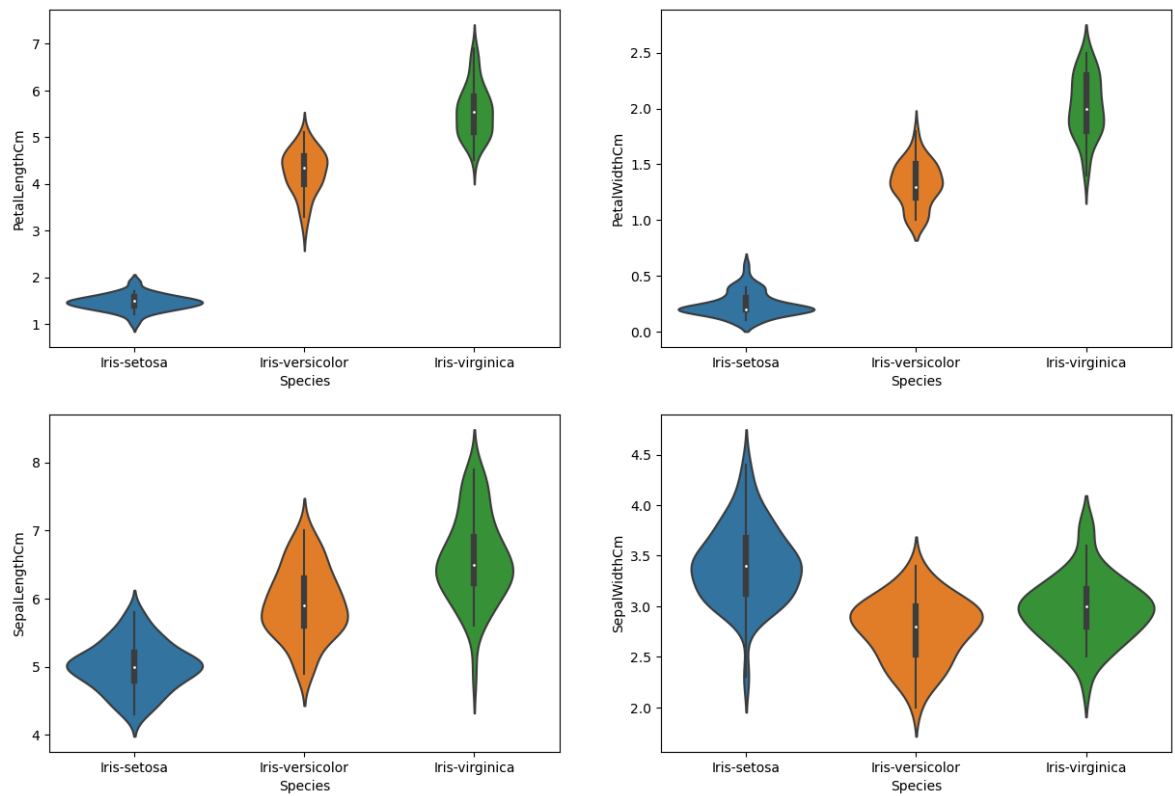
In [21]: fig=plt.gcf()
fig.set_size_inches(10,7)
fig=sns.violinplot(x='Species',y='SepalLengthCm',data=iris)

```



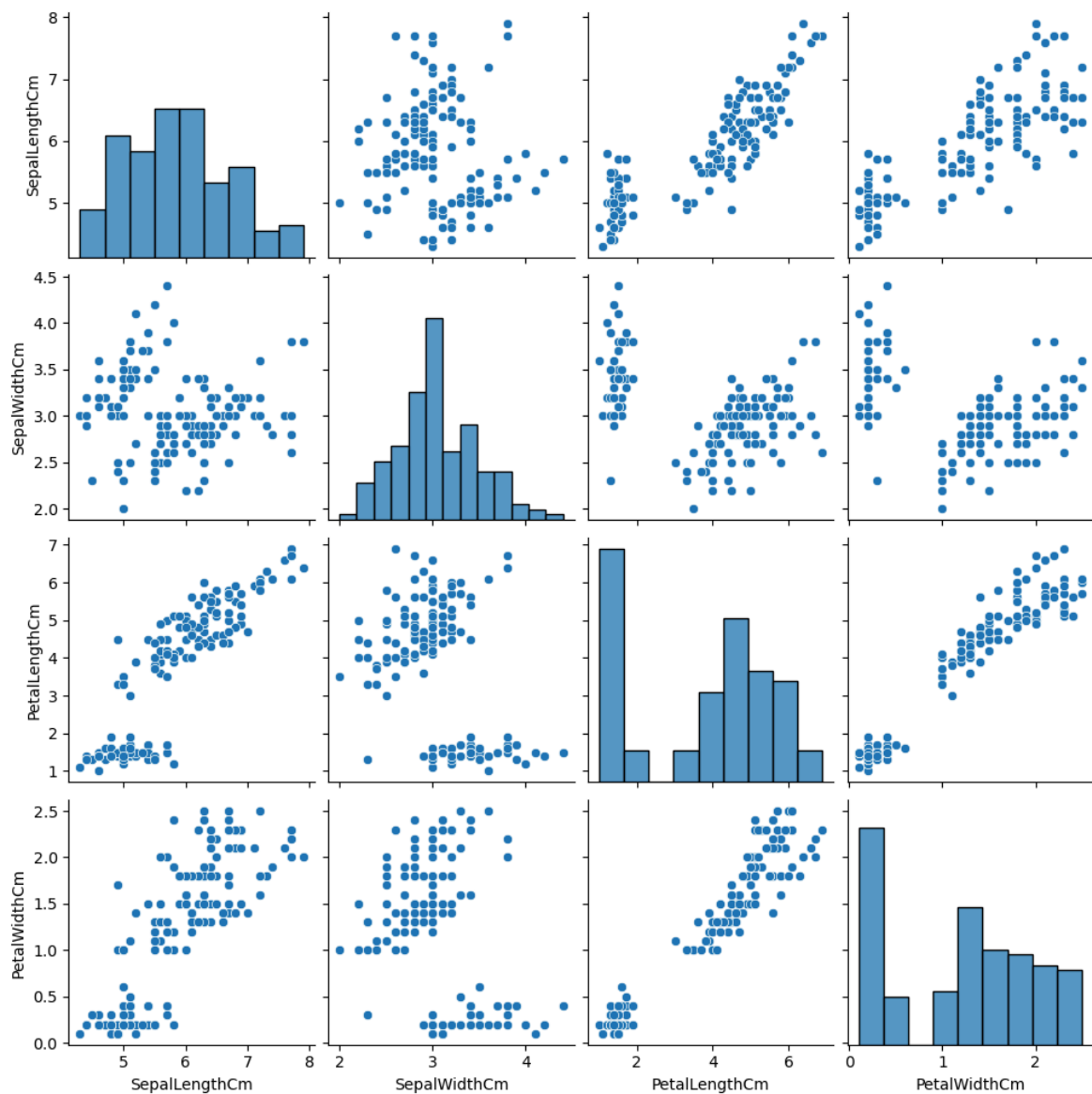
```
In [22]: plt.figure(figsize=(15,10))
plt.subplot(2,2,1)
sns.violinplot(x='Species',y='PetalLengthCm',data=iris)
plt.subplot(2,2,2)
sns.violinplot(x='Species',y='PetalWidthCm',data=iris)
plt.subplot(2,2,3)
sns.violinplot(x='Species',y='SepalLengthCm',data=iris)
plt.subplot(2,2,4)
sns.violinplot(x='Species',y='SepalWidthCm',data=iris)
```

```
Out[22]: <Axes: xlabel='Species', ylabel='SepalWidthCm'>
```



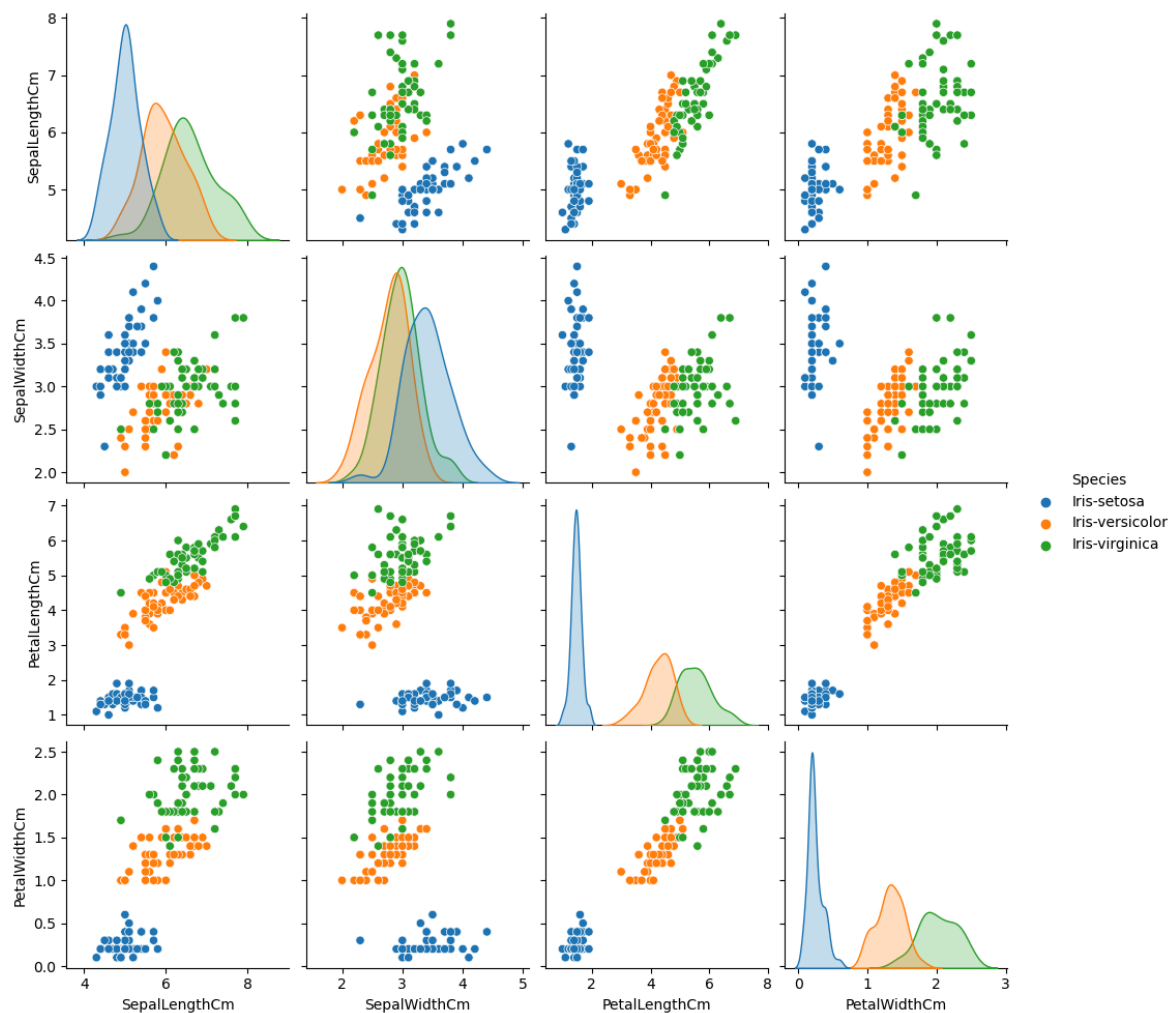
```
In [23]: # Pair Plot  
sns.pairplot(data=iris,kind='scatter')
```

```
Out[23]: <seaborn.axisgrid.PairGrid at 0x1a02343d110>
```



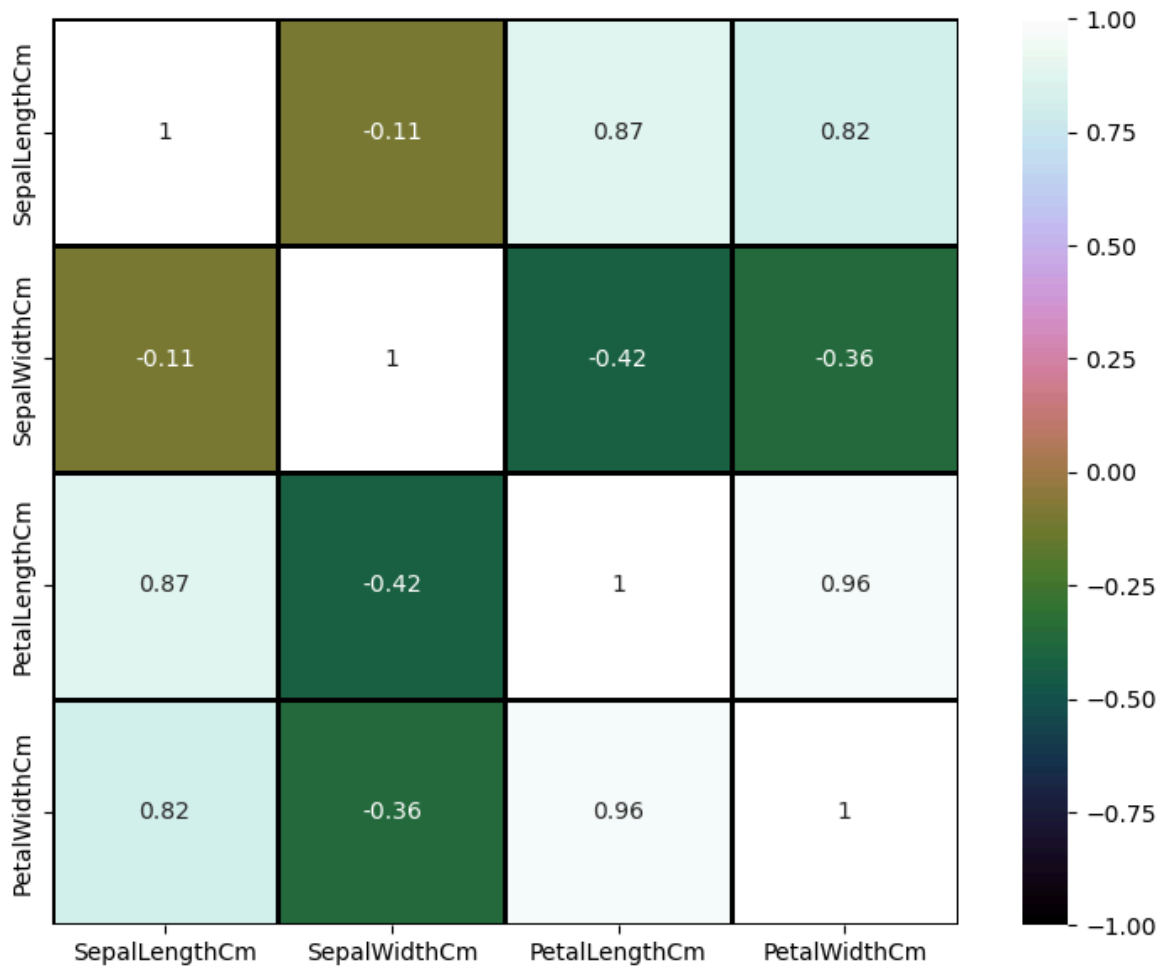
```
In [24]: sns.pairplot(data=iris,hue='Species')
```

```
Out[24]: <seaborn.axisgrid.PairGrid at 0x1a02325ced0>
```



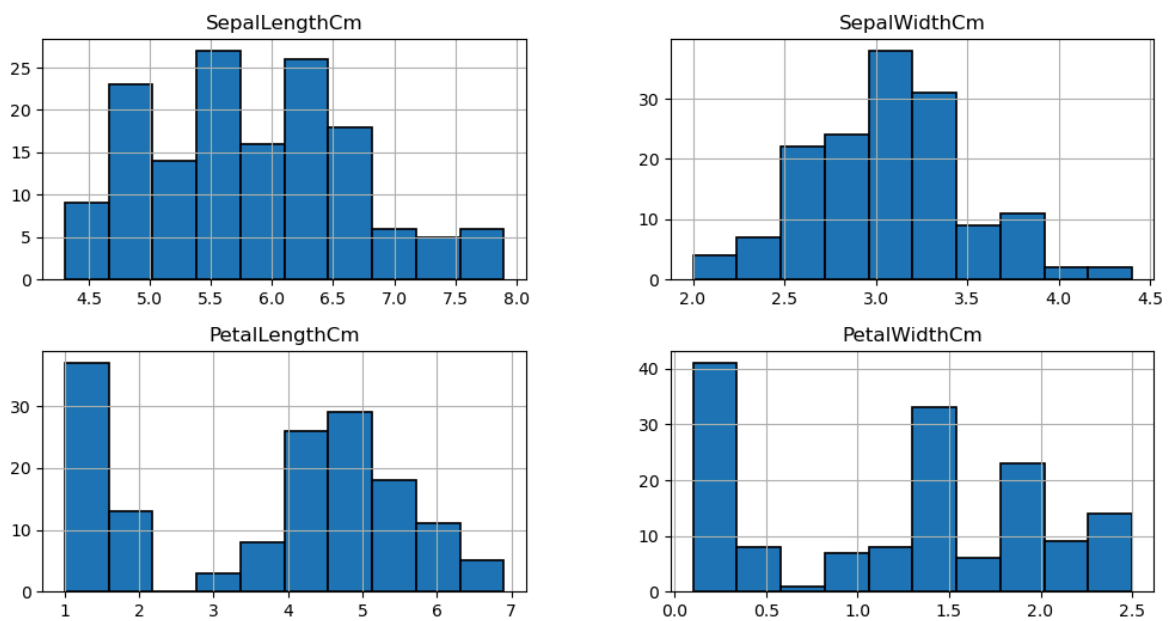
```
In [25]: # Heat Map
numeric_iris = iris.select_dtypes(include=['number'])

fig=plt.gcf()
fig.set_size_inches(10,7)
fig=sns.heatmap(numeric_iris.corr(),annot=True,cmap='cubehelix',linewidths=1,lin
```



```
In [26]: # Distribution Plot

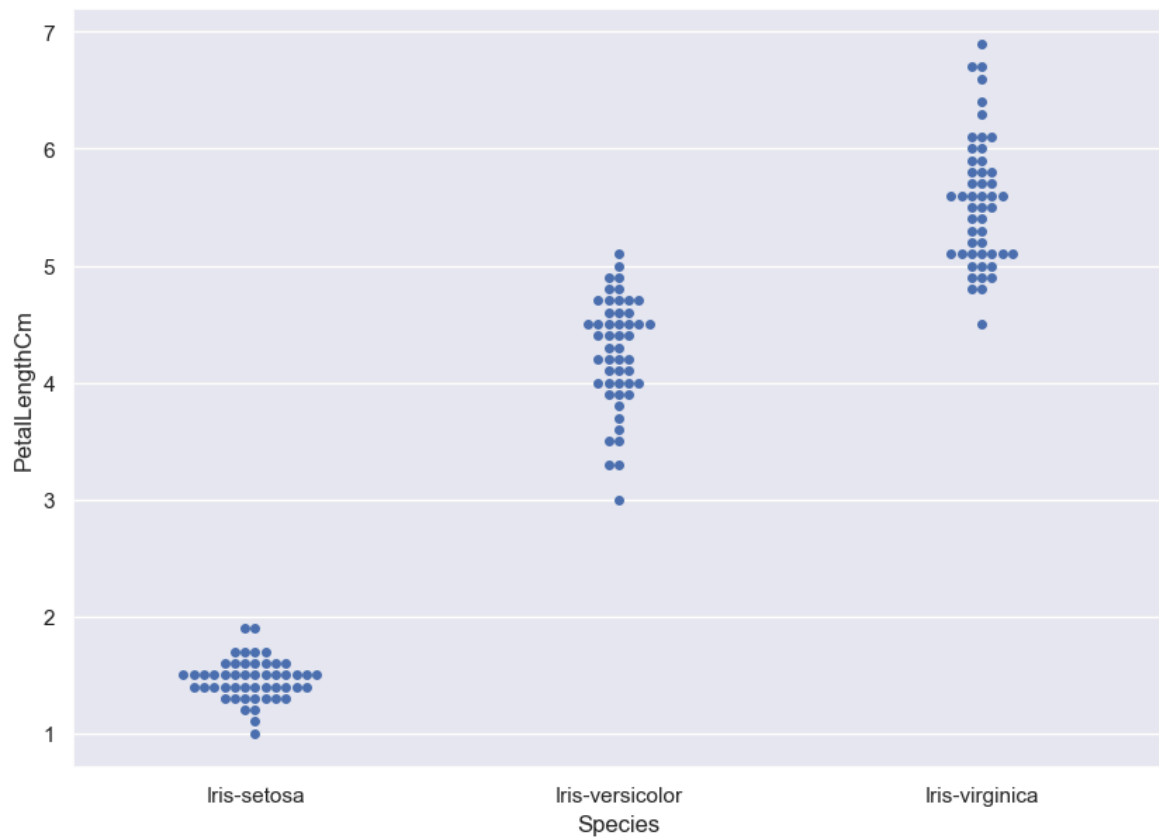
iris.hist(edgecolor='black', linewidth=1.2)
fig=plt.gcf()
fig.set_size_inches(12,6)
```



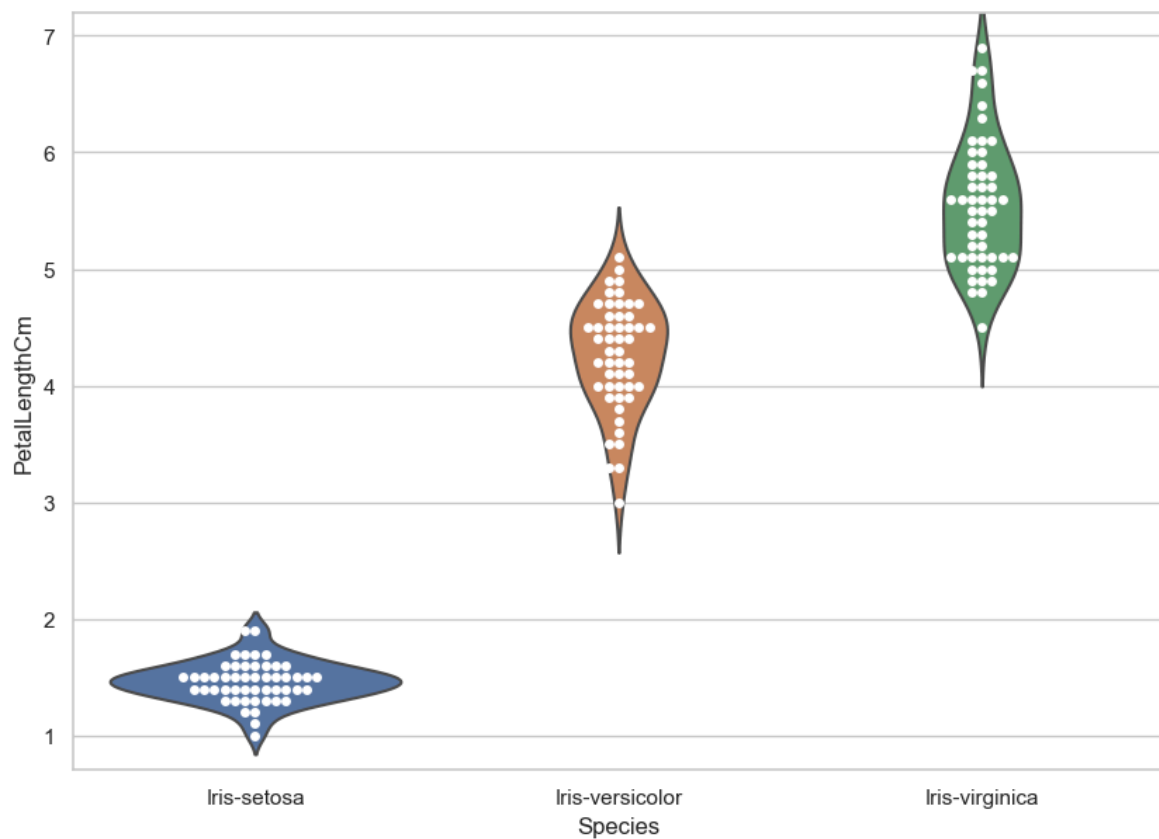
```
In [27]: # Swarm Plot

sns.set(style="darkgrid")
fig=plt.gcf()
```

```
fig.set_size_inches(10,7)
fig = sns.swarmplot(x="Species", y="PetalLengthCm", data=iris)
```

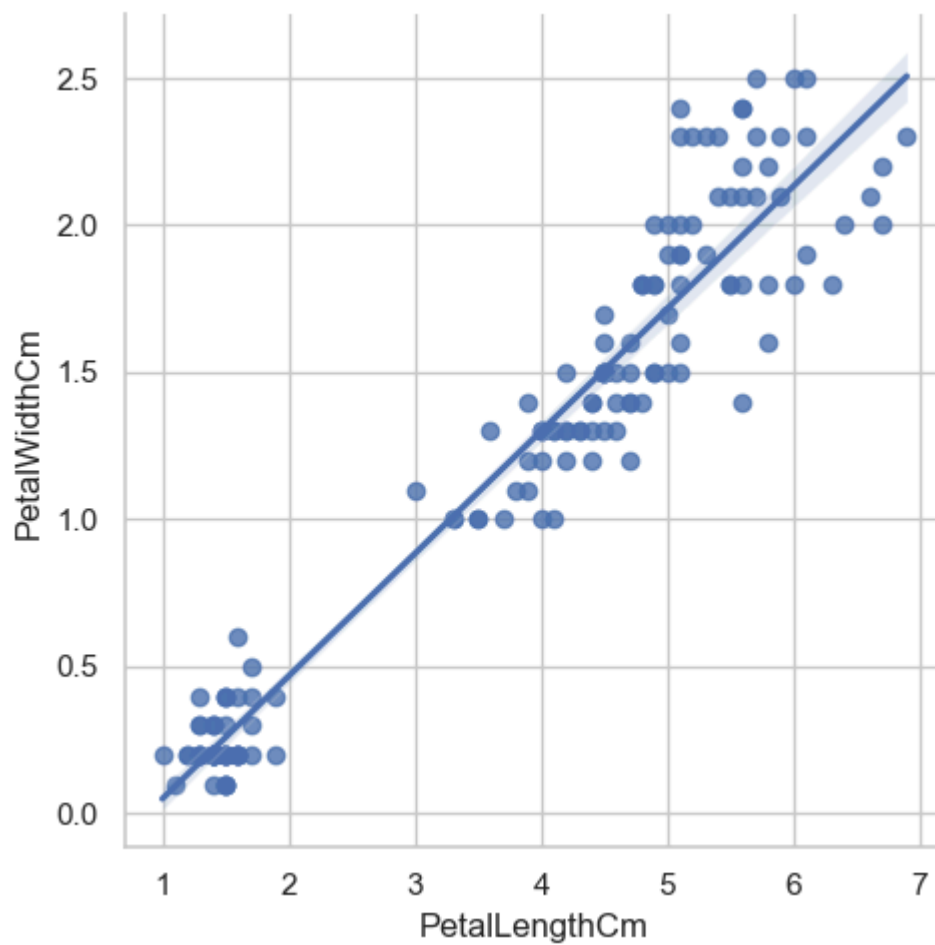


```
In [28]: sns.set(style="whitegrid")
fig=plt.gcf()
fig.set_size_inches(10,7)
ax = sns.violinplot(x="Species", y="PetalLengthCm", data=iris, inner=None)
ax = sns.swarmplot(x="Species", y="PetalLengthCm", data=iris,color="white", edge
```



In [29]: `# LM Plot`

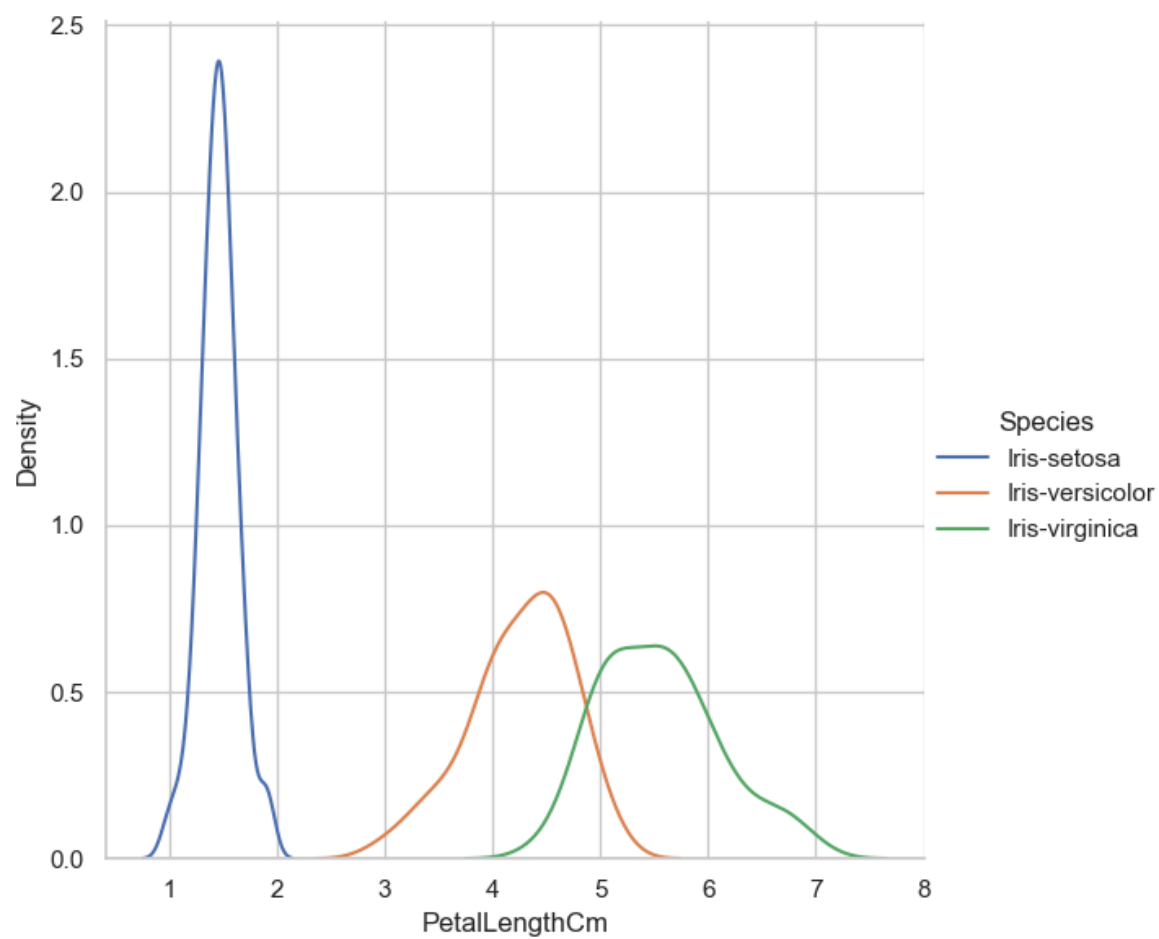
```
fig=sns.lmplot(x="PetalLengthCm", y="PetalWidthCm",data=iris)
```



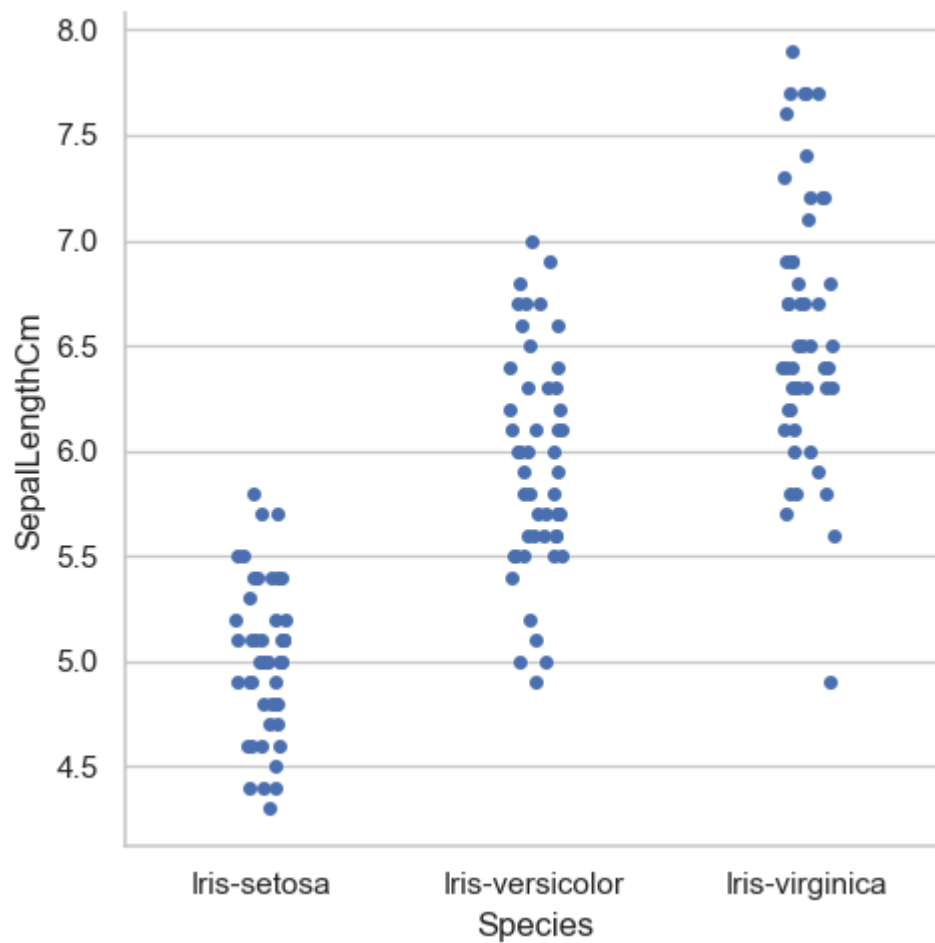
In [30]: `# FacetGrid`

```
sns.FacetGrid(iris, hue="Species", height=6) \
    .map(sns.kdeplot, "PetalLengthCm") \
    .add_legend()
plt.ioff()
```

Out[30]: `<contextlib.ExitStack at 0x1a02792c350>`



```
In [31]: # Factorplot
sns.catplot(x='Species',y='SepalLengthCm',data=iris)
plt.ioff()
plt.show()
```

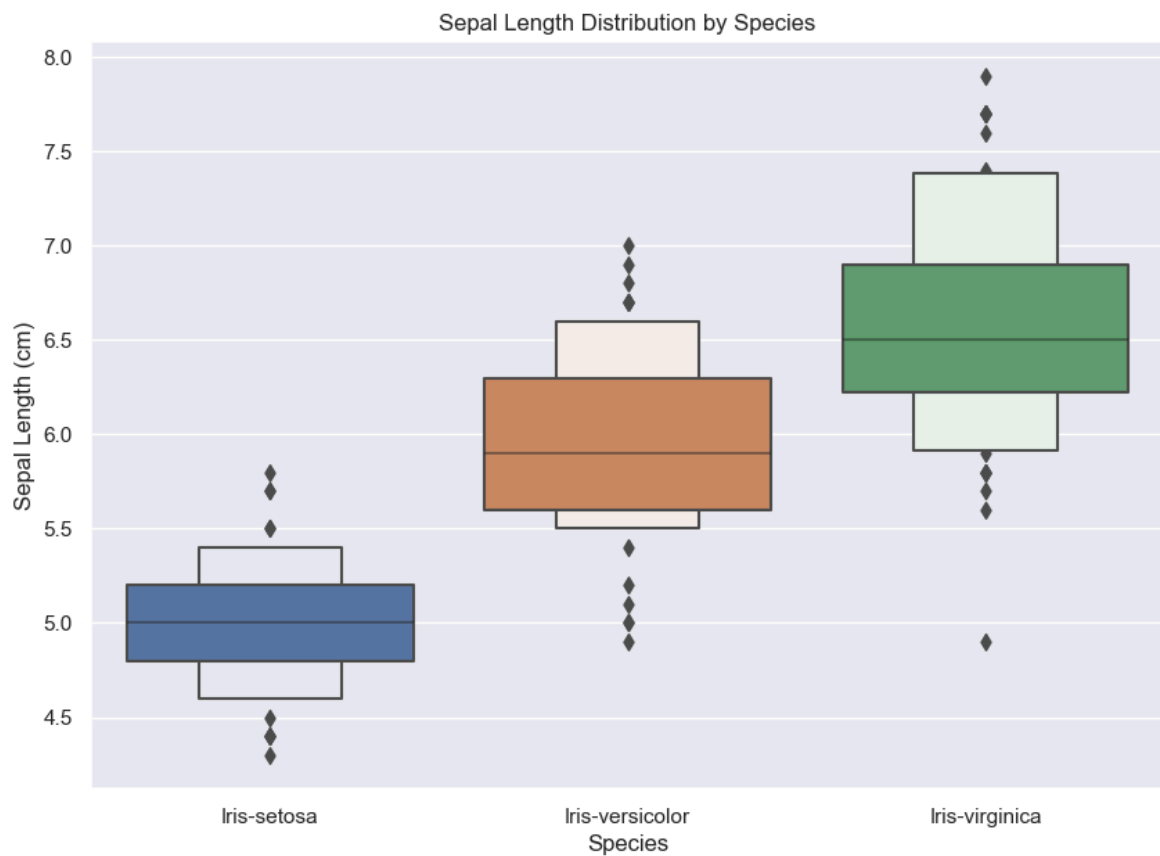


```
In [64]: # Boxen Plot

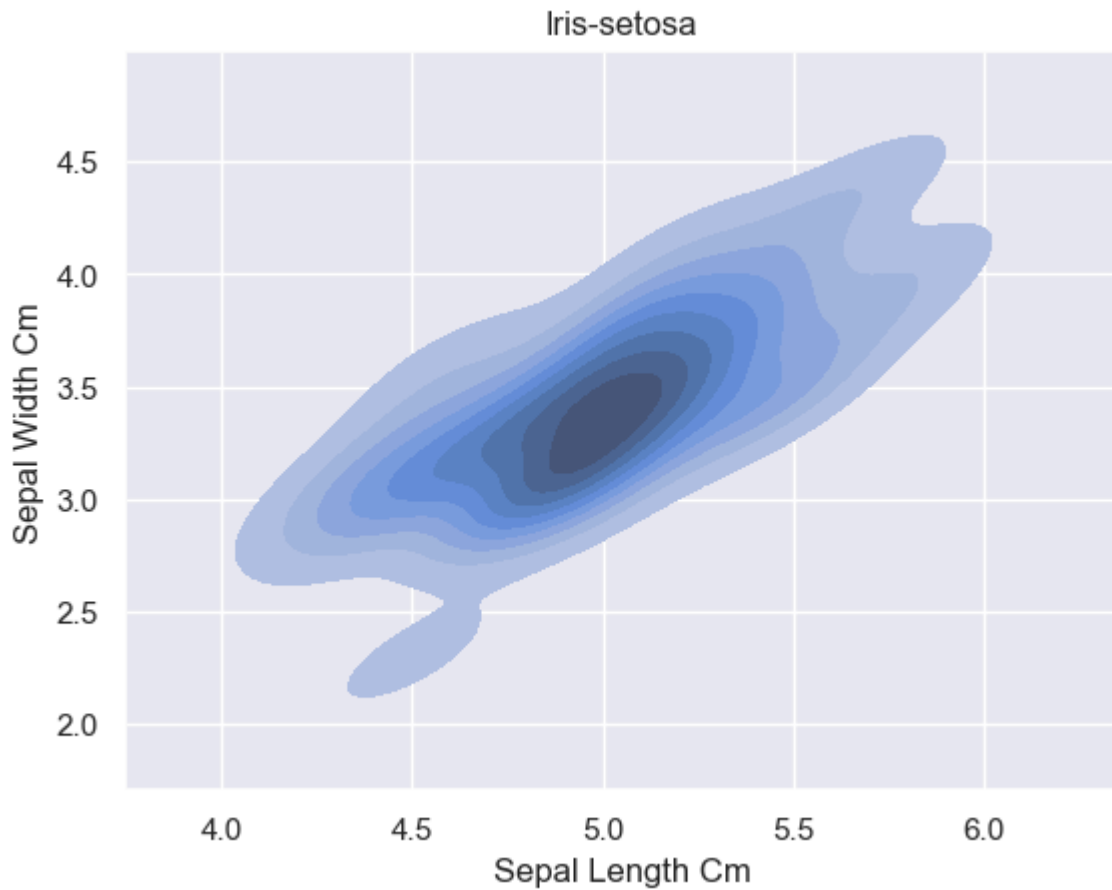
fig=plt.gcf()
fig.set_size_inches(10,7)
fig=sns.boxenplot(x='Species',y='SepalLengthCm',data=iris)

plt.title("Sepal Length Distribution by Species")
plt.xlabel("Species")
plt.ylabel("Sepal Length (cm)")

plt.show()
```



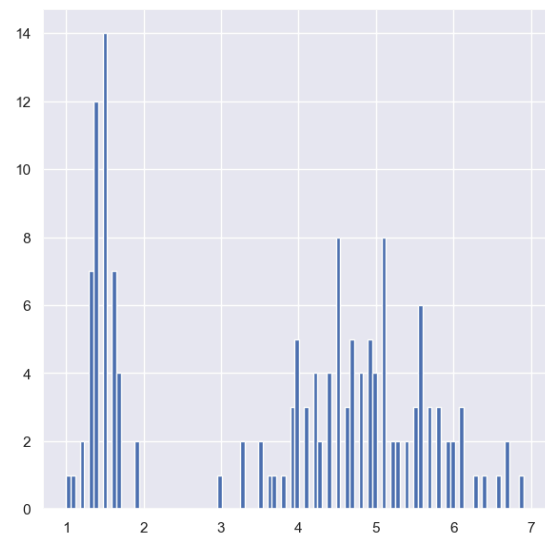
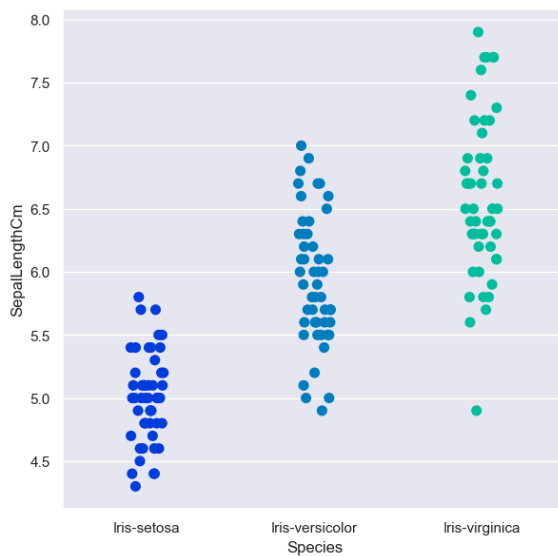
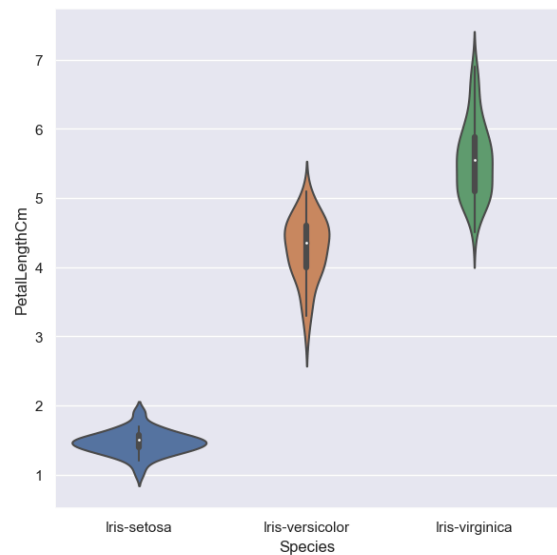
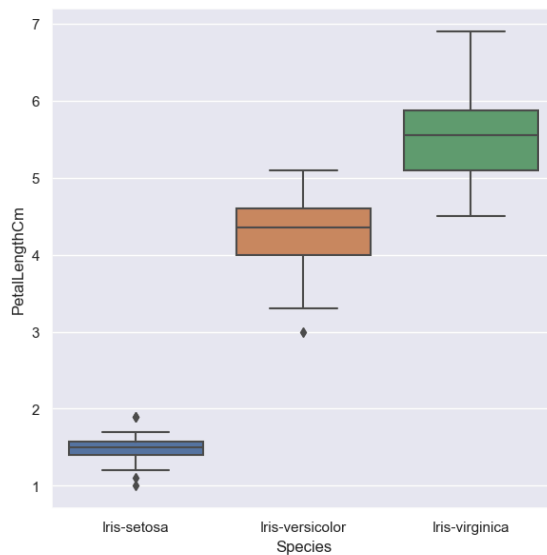
```
In [63]: # KDE Plot
sub=iris[iris['Species']=='Iris-setosa']
sns.kdeplot(data=sub,x='SepalLengthCm',y='SepalWidthCm', shade=True, shade_lowest=True)
plt.title('Iris-setosa')
plt.xlabel('Sepal Length Cm')
plt.ylabel('Sepal Width Cm')
plt.show()
```



```
In [40]: # Dashboard

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='
#axes[1, 1].hist(iris.hist, bin=10)
axes[1, 1].hist(iris.PetalLengthCm, bins=100)
#k2.set(xlim=(-1, 0.8))
plt.show()
```



```
In [43]: # Stacked Histogram

iris['Species'] = iris['Species'].astype('category')
iris.head()
```

```
Out[43]:
```

	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 [44]: 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   category
dtypes: category(1), float64(4)
memory usage: 5.1 KB

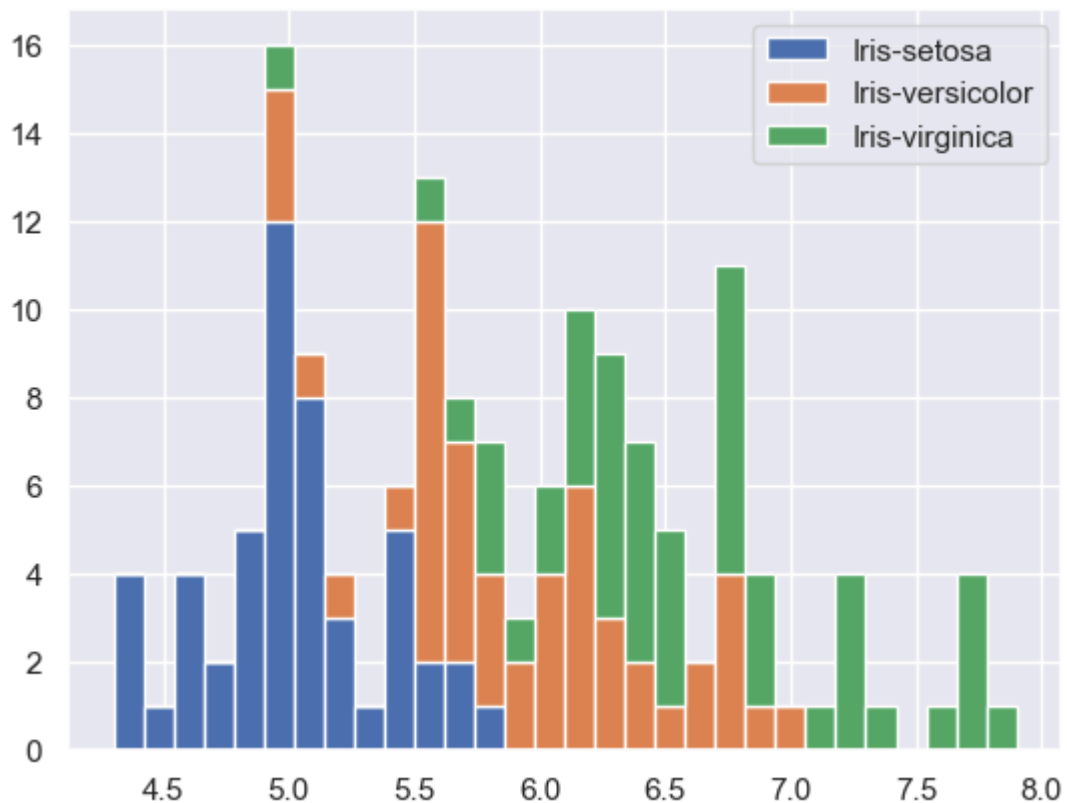
```

```

In [45]: 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()

```



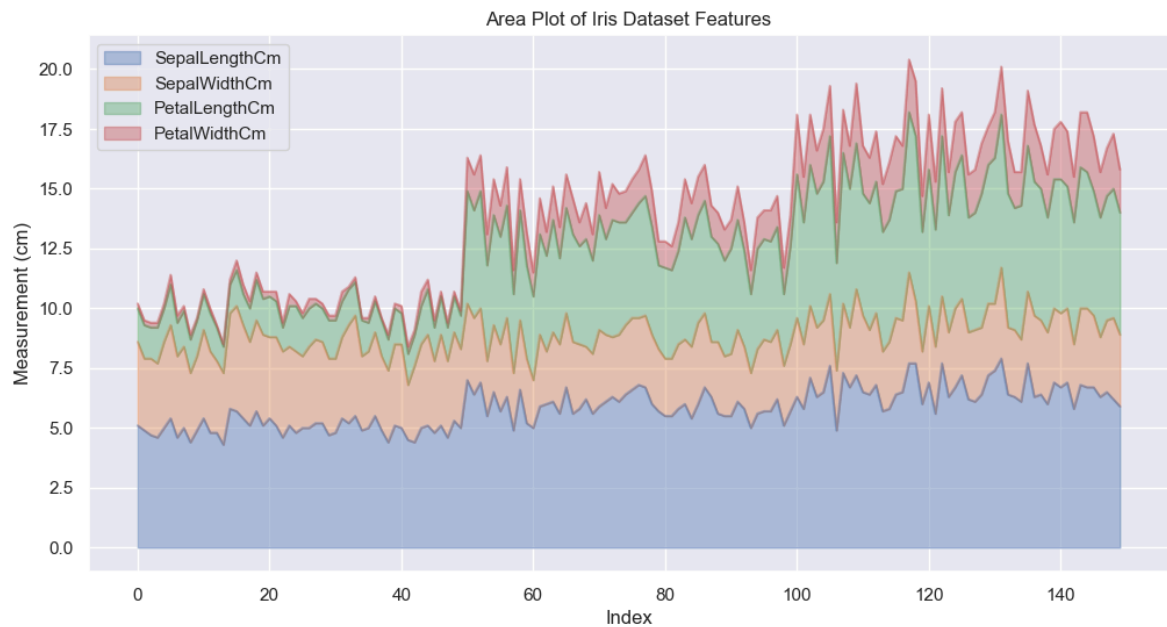
```

In [54]: # Area Plot

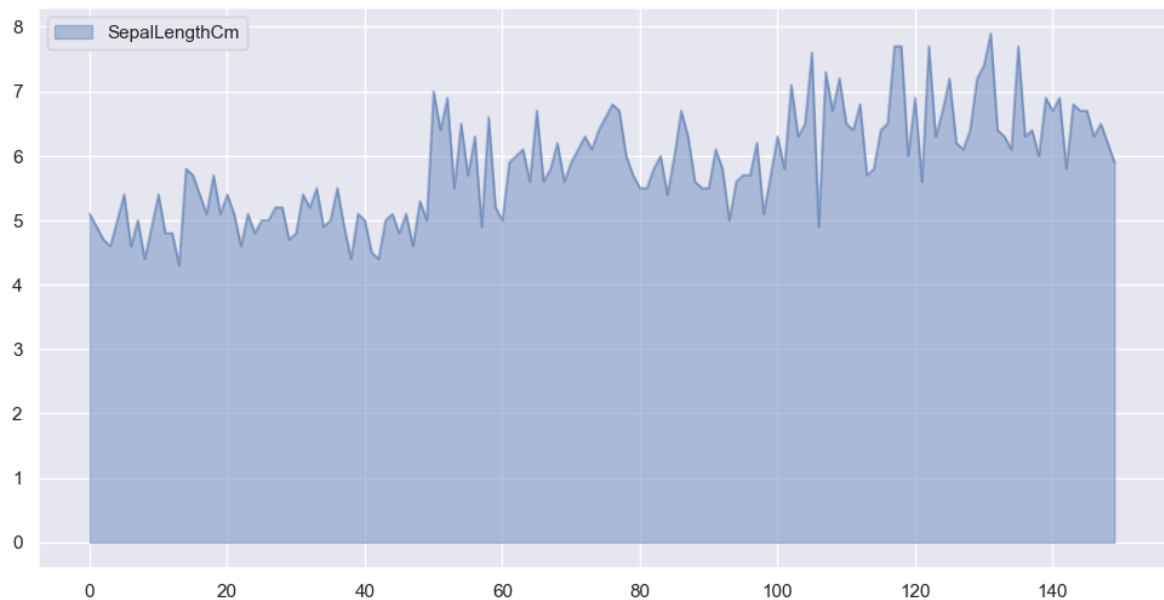
#iris['SepalLengthCm'] = iris['SepalLengthCm'].astype('category')
iris.plot.area(y=['SepalLengthCm', 'SepalWidthCm', 'PetalLengthCm', 'PetalWidthCm'],
plt.title("Area Plot of Iris Dataset Features")
plt.ylabel("Measurement (cm)")
plt.xlabel("Index")

plt.show()

```



```
In [55]: iris.plot.area(y='SepalLengthCm',alpha=0.4,figsize=(12, 6));  
plt.show()
```



```
In [58]: # Distplot  
sns.distplot(iris['SepalLengthCm'],kde=True,bins=20);  
plt.show()
```