Iris Data Visualization

In [2]:
```
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\LENOVO\Downloads\28th - Iris, movie analytics Project\I
```

In [5]:
```
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.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]:

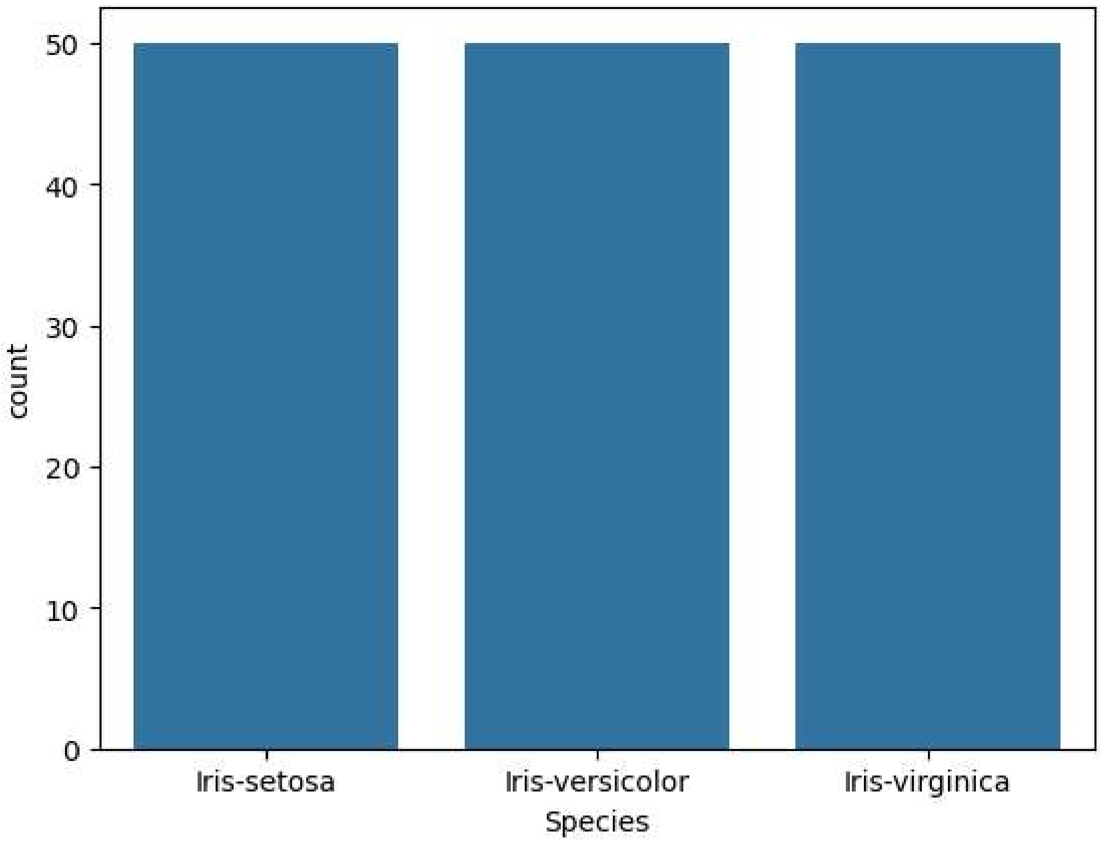
```
iris['Species'].value_counts()
```

Out[10]:

```
Species
Iris-setosa        50
Iris-versicolor    50
Iris-virginica     50
Name: count, dtype: int64
```

In [11]:

```
sns.countplot(data=iris,x='Species')
plt.show()
```



In [12]:

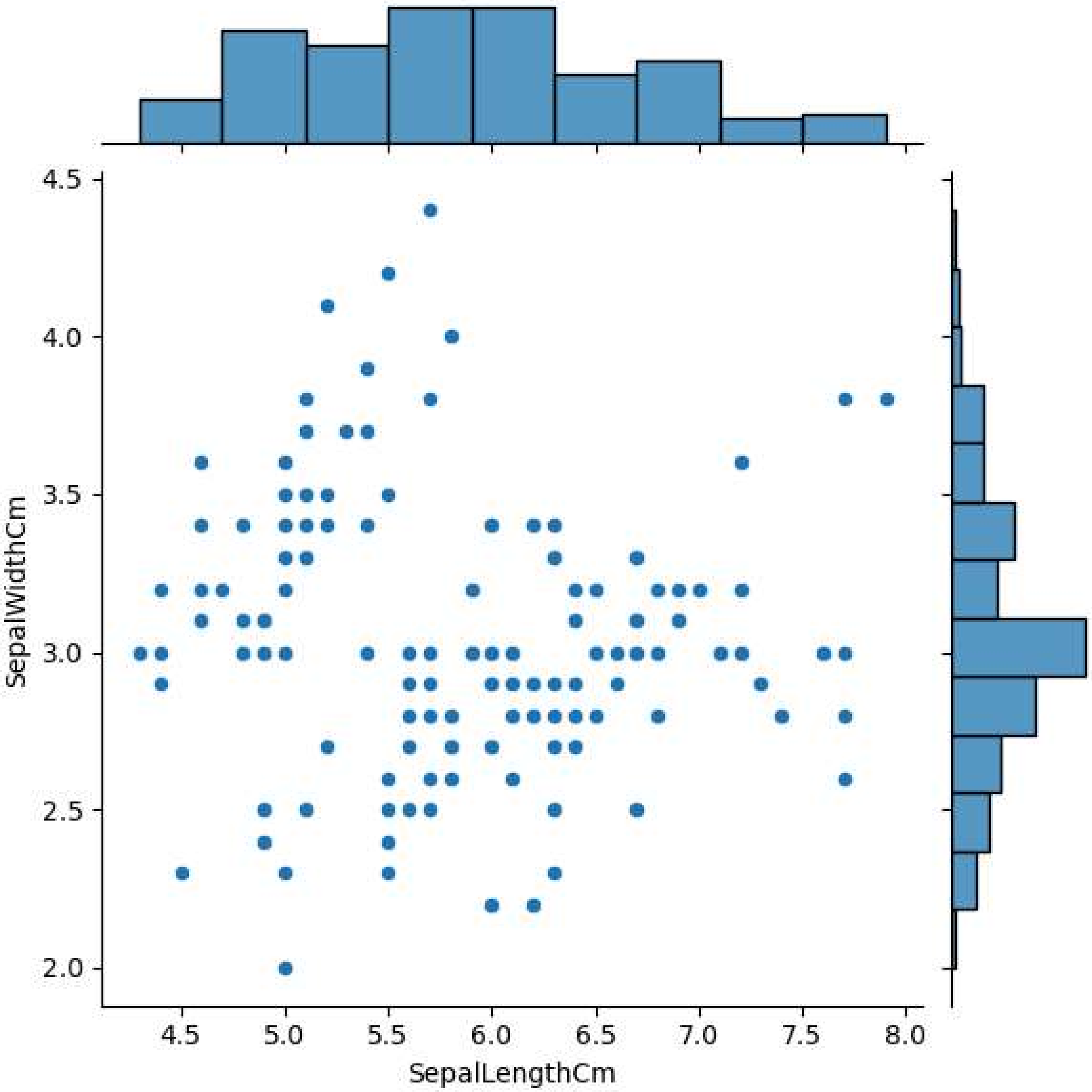
```
iris.head()
```

Out[12]:

	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

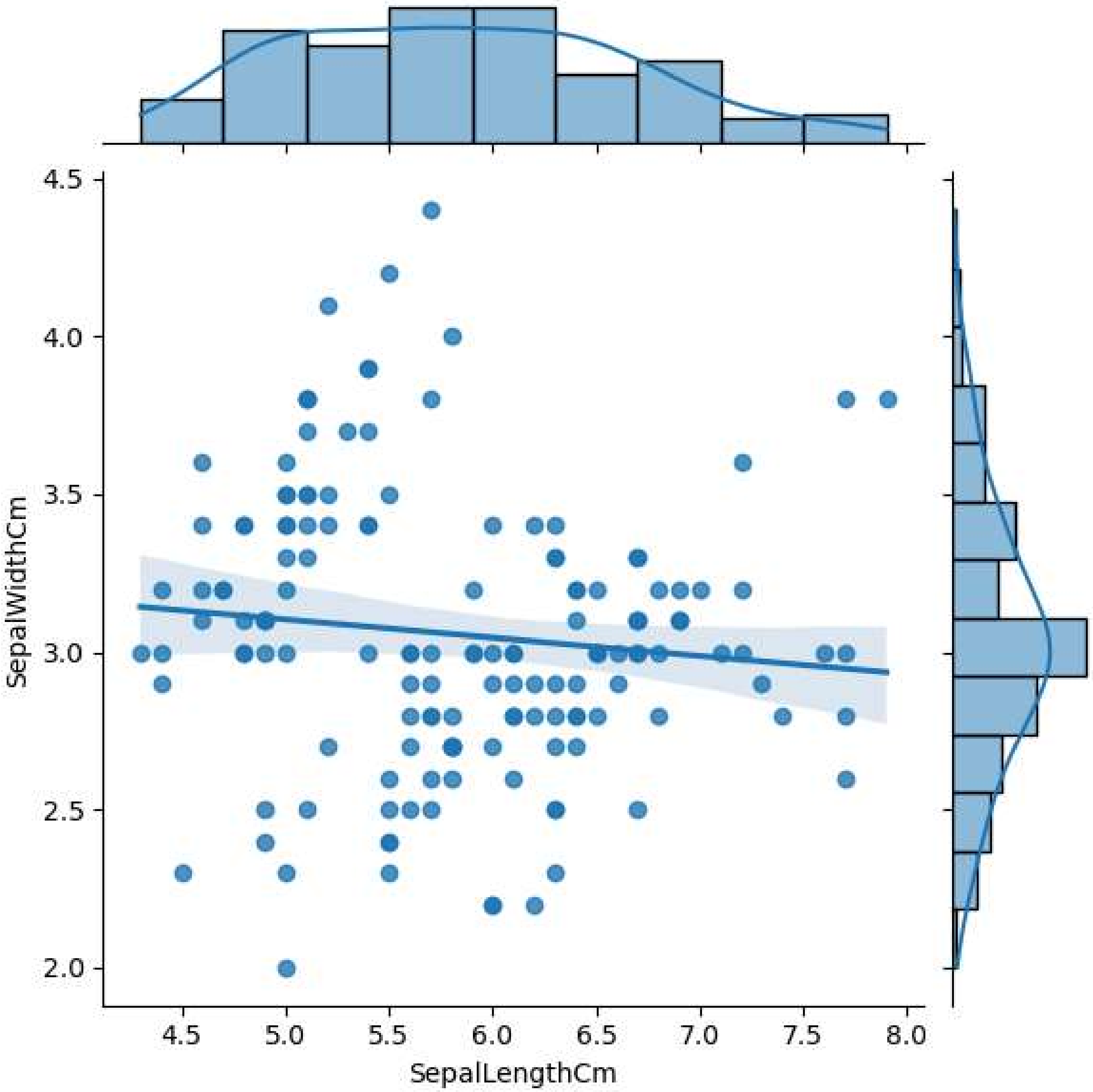
In [13]:

```
fig=sns.jointplot(x='SepalLengthCm',y='SepalWidthCm',data=iris)
plt.show()
```

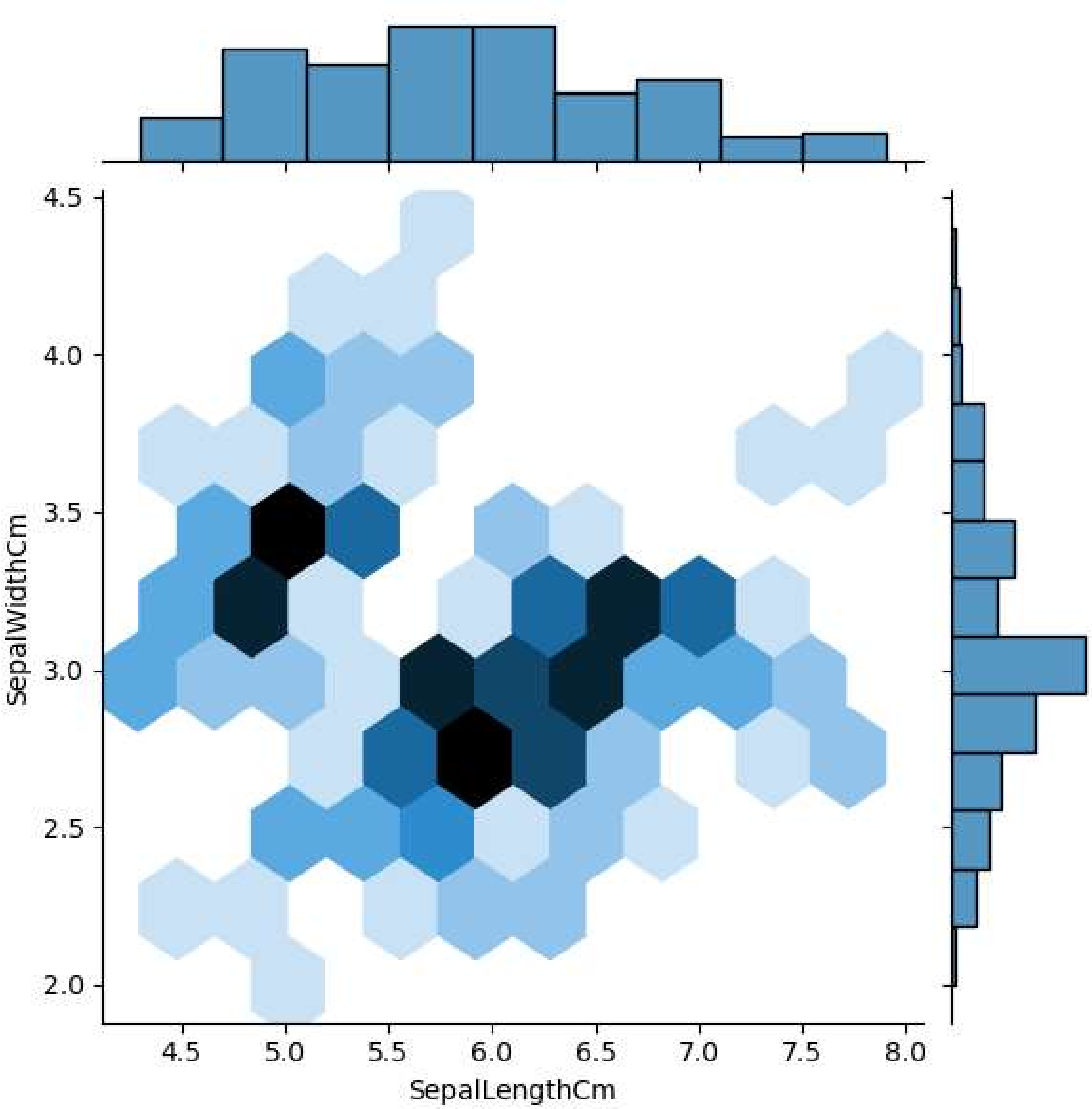


In [14]:

```
sns.jointplot(x='SepalLengthCm',y='SepalWidthCm',data=iris,kind="reg")
plt.show()
```

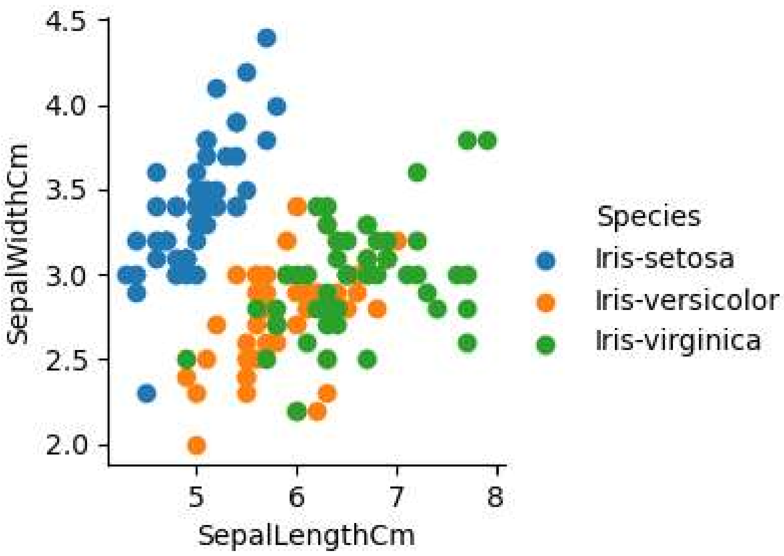


In [15]:
```
fig=sns.jointplot(x='SepalLengthCm',y='SepalWidthCm',kind='hex',data=iris)
plt.show()
```



5. FacetGrid Plot

In [17]:
```
import matplotlib.pyplot as plt
%matplotlib inline
sns.FacetGrid(iris,hue='Species')\
.map(plt.scatter,'SepalLengthCm','SepalWidthCm')\
.add_legend()
plt.show()
```

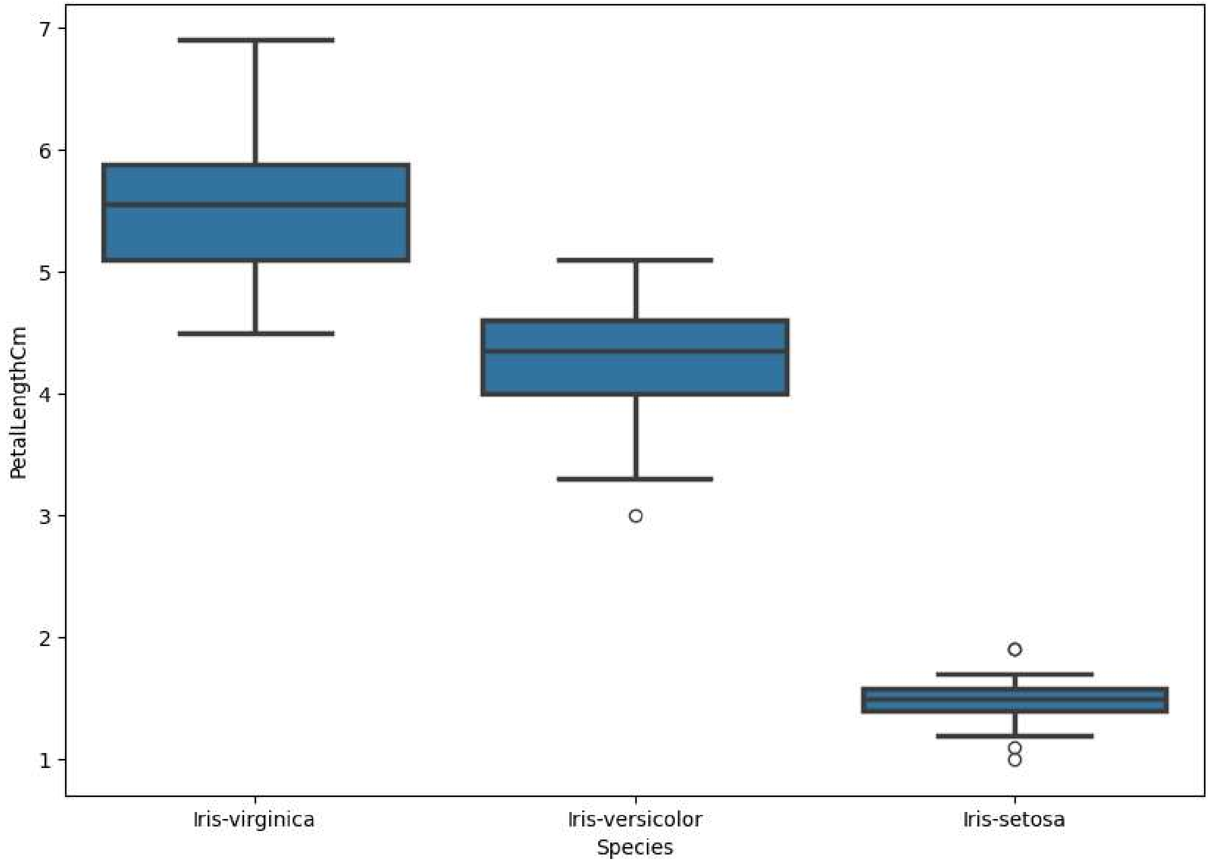


In [18]:
```
iris.head()
```

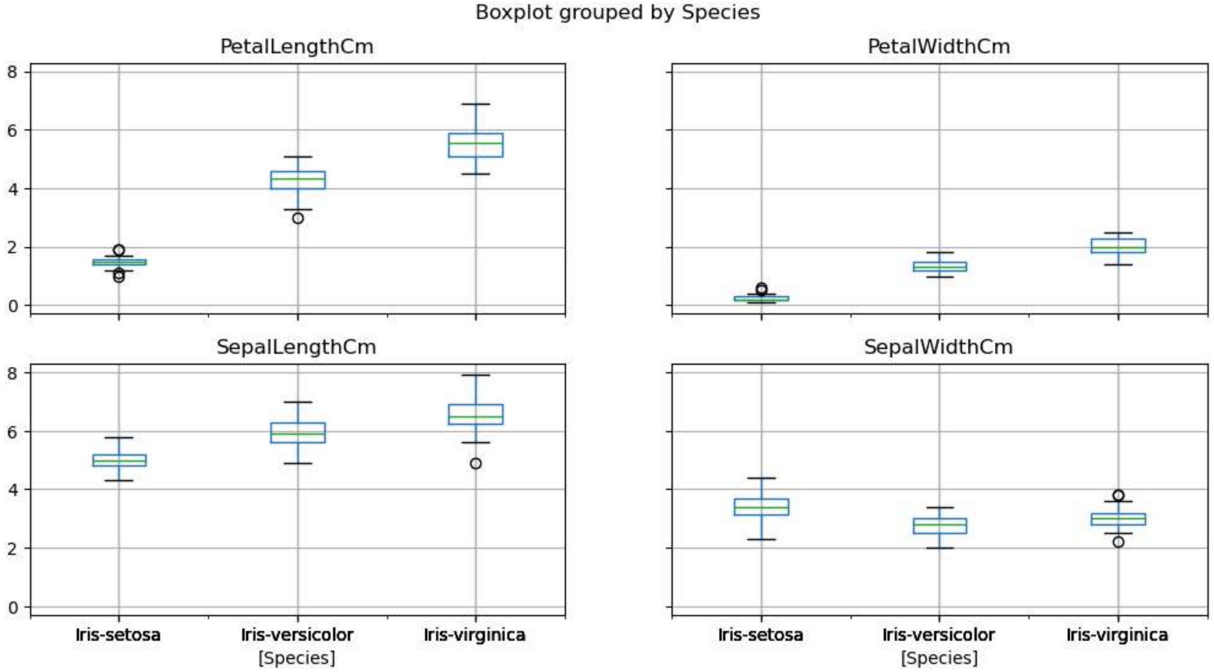
Out[18]:

	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 [19]:
```
fig=plt.gcf()
fig.set_size_inches(10,7)
fig=sns.boxplot(x='Species',y='PetalLengthCm',data=iris,order=['Iris-virginica','Ir
plt.show()
```



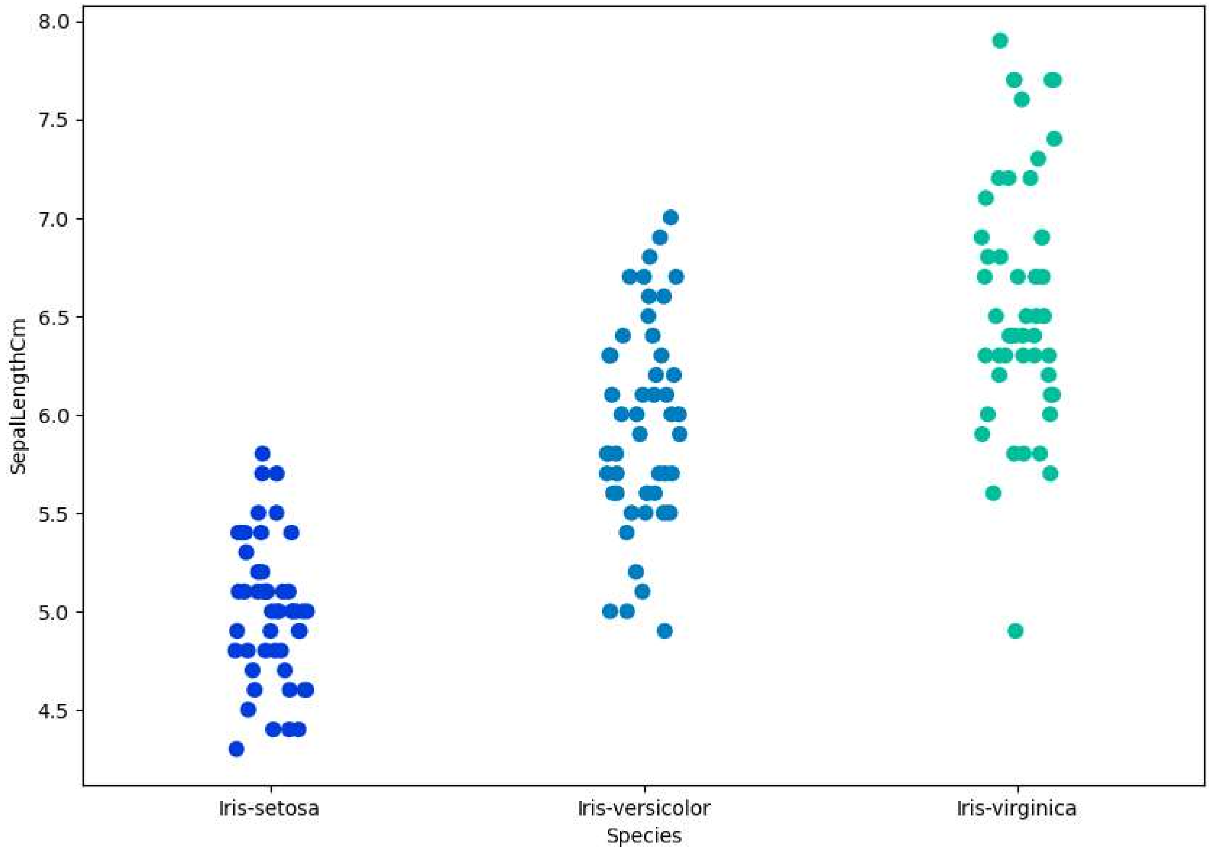
```
In [20]: iris.boxplot(by="Species",figsize=(12,6))
         plt.show()
```



7 Strip plot

In [22]:

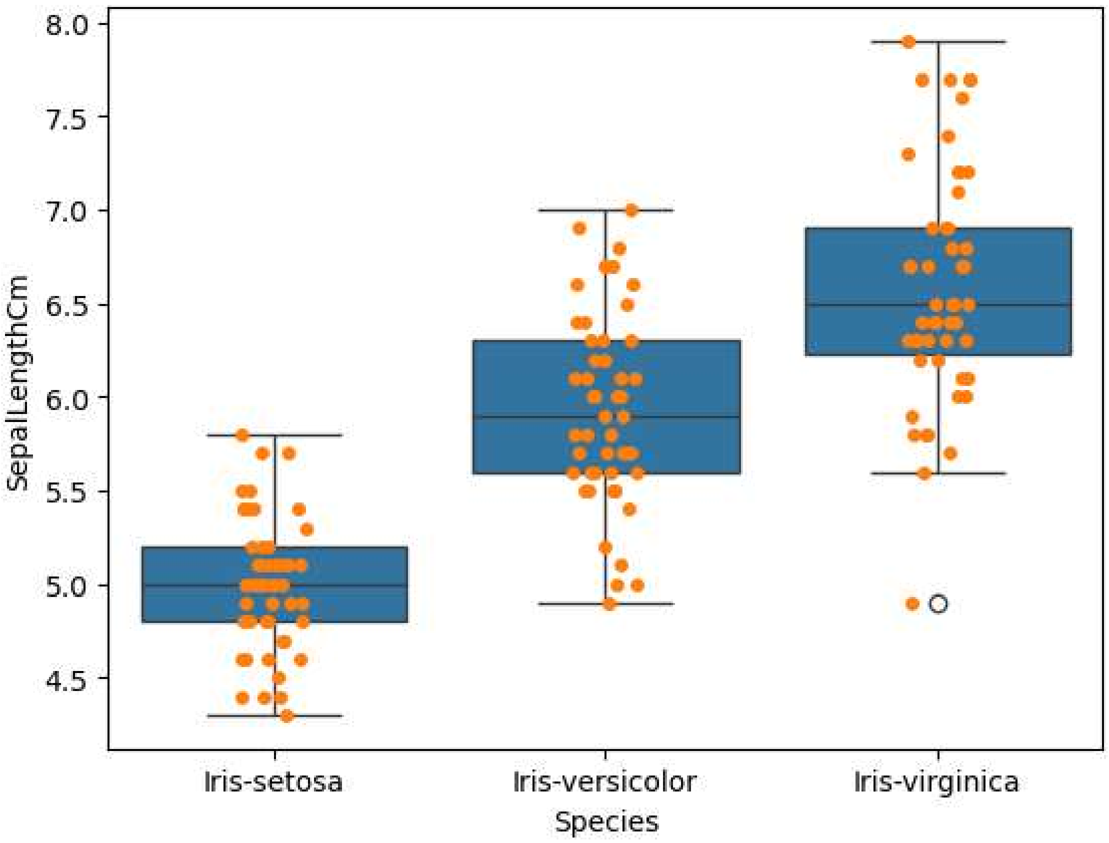
```
fig=plt.gcf()
fig.set_size_inches(10,7)
fig=sns.stripplot(x='Species',y='SepalLengthCm',data=iris,jitter=True,edgecolor='gr
plt.show()
```



8. Combining Box and Strip Plots

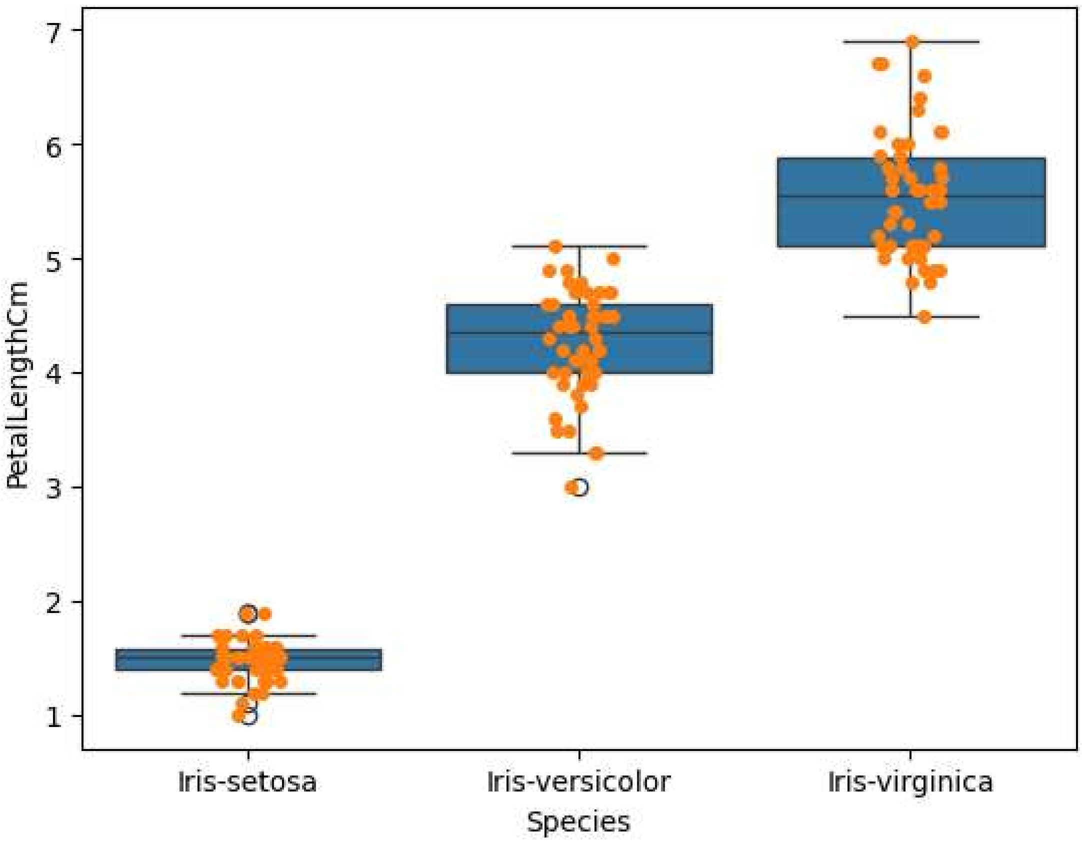
In [24]:

```
fig=plt.gcf
fig=sns.boxplot(x='Species',y='SepalLengthCm',data=iris)
fig=sns.stripplot(x='Species',y='SepalLengthCm',data=iris,jitter=True,edgecolor='gr
plt.show()
```



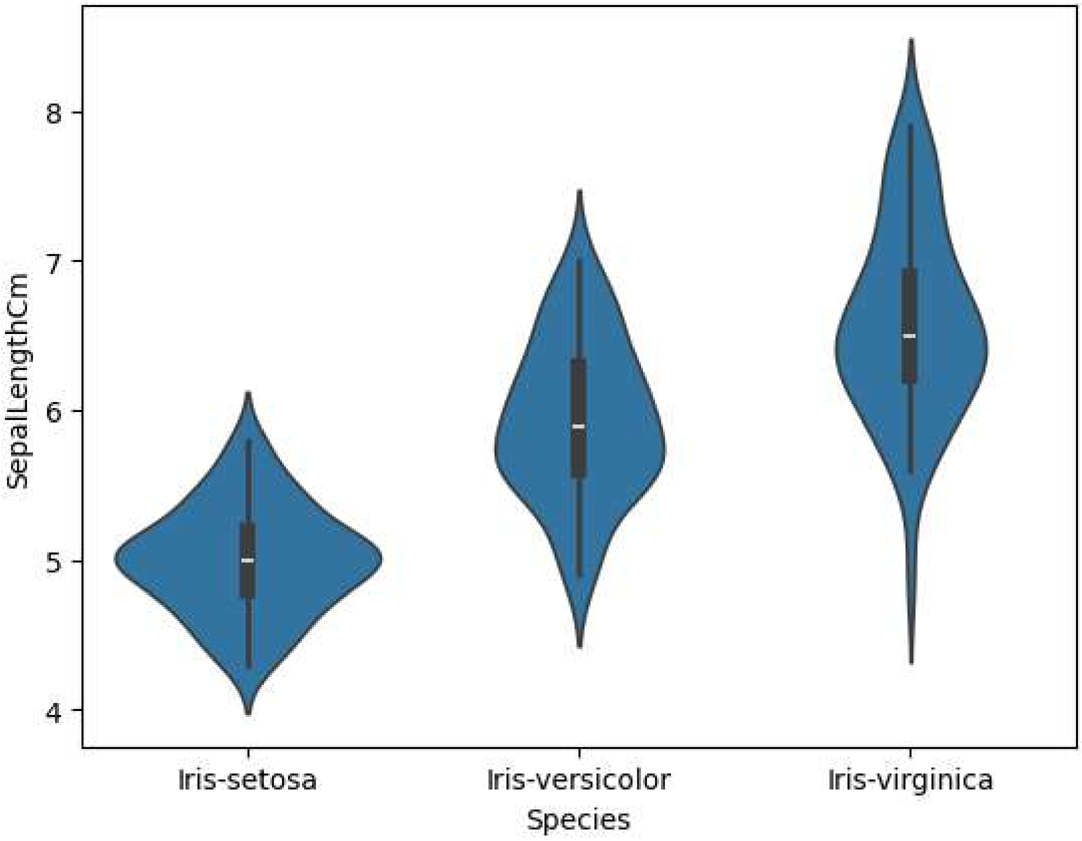
In [25]:

```
ax=sns.boxplot(x="Species",y="PetalLengthCm",data=iris)
ax=sns.stripplot(x="Species",y="PetalLengthCm",data=iris,jitter=True,edgecolor="gra
colors=['yellow',
        'red','grren']
for i,box in enumerate(ax.artists):
    box.set_facecolor(colors[i])
    box.set_edgecolor('black')

plt.show()
```

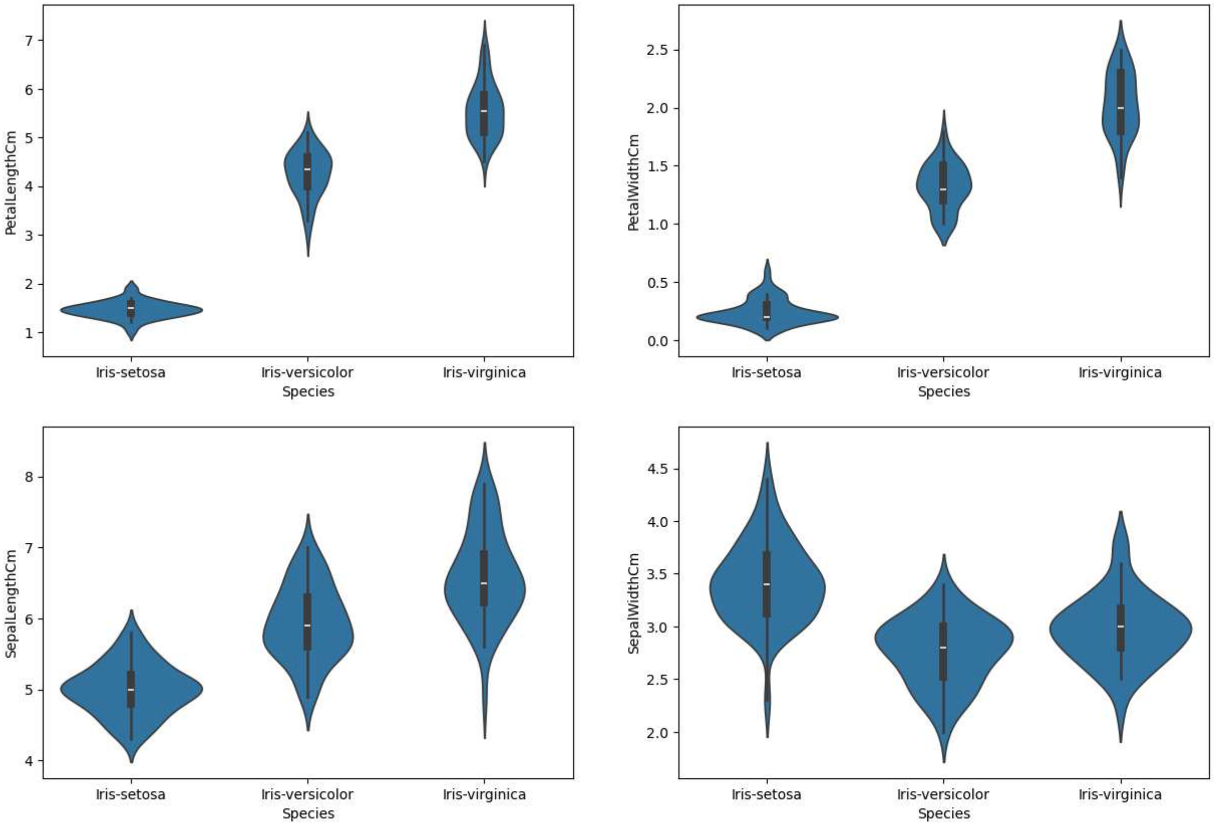


9. Violin Plot

```
In [27]: fif=plt.gcf
         fig=sns.violinplot(x='Species',y='SepalLengthCm',data=iris)
         plt.show()
```



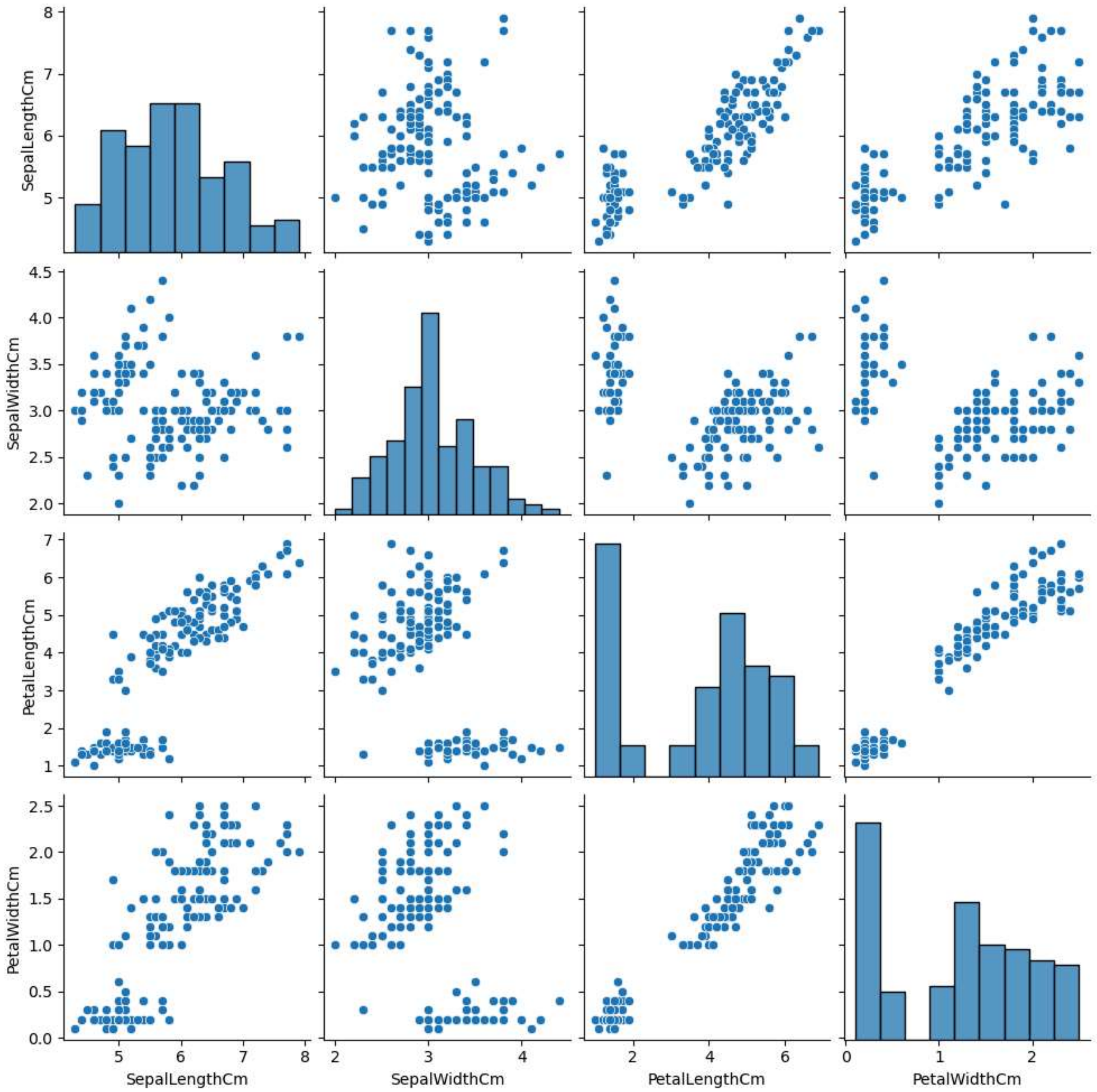
In [28]:
```
plt.figure(figsize=(15,10))
plt.subplot(2,2,1)
sns.violinplot(x='Species',y='PetalLengthCm',data=iris)
plt.subplot(2,2,2)
sns.violinplot(x='Species',y='PetalWidthCm',data=iris)
plt.subplot(2,2,3)
sns.violinplot(x='Species',y='SepalLengthCm',data=iris)
plt.subplot(2,2,4)
sns.violinplot(x='Species',y='SepalWidthCm',data=iris)
plt.show()
```



10. Pair Plot

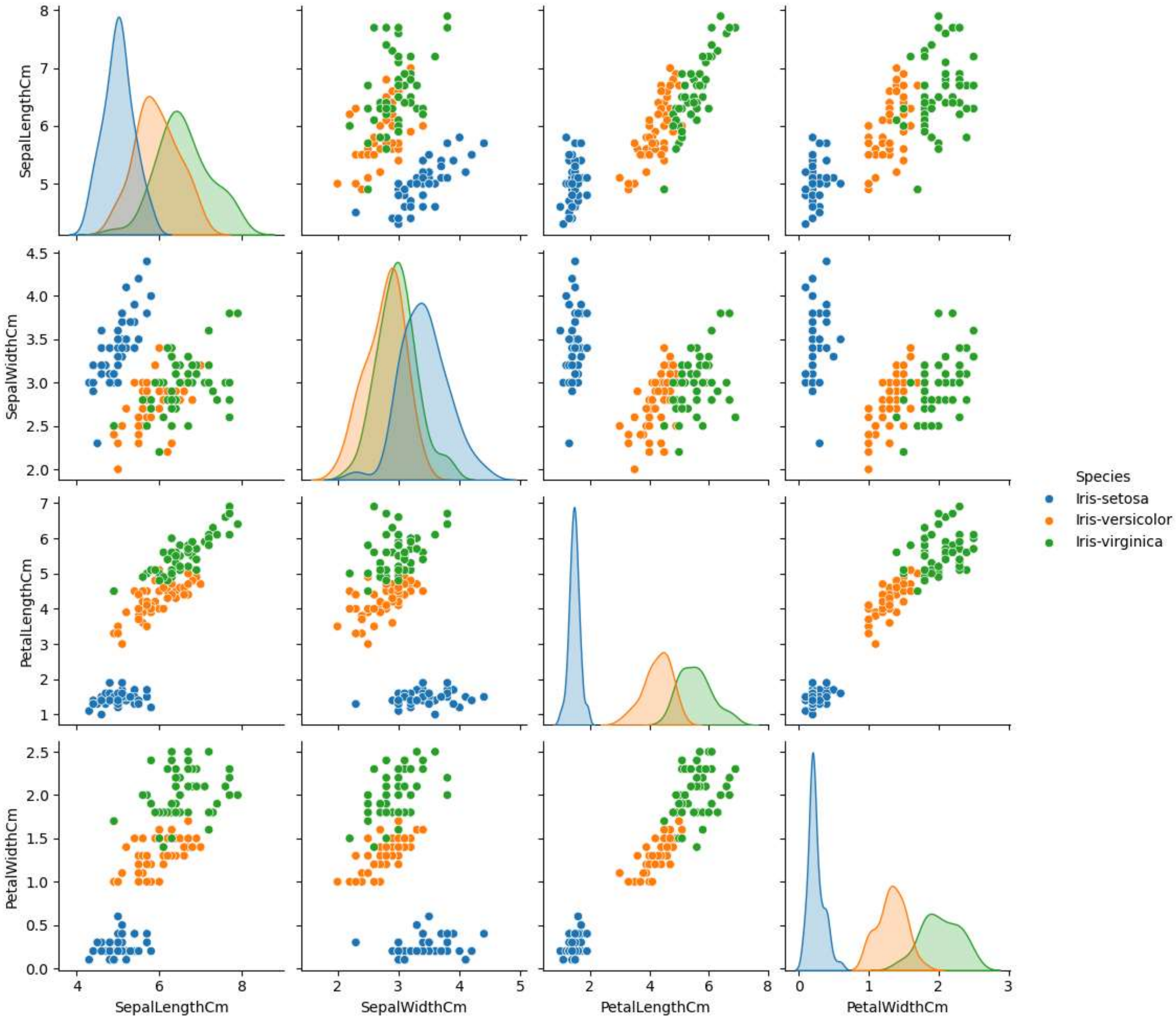
In [30]:

```
sns.pairplot(data=iris,kind='scatter')
plt.show()
```



In [31]:

```
sns.pairplot(data=iris,hue='Species')
plt.show()
```



11. Heat map

In [33]:

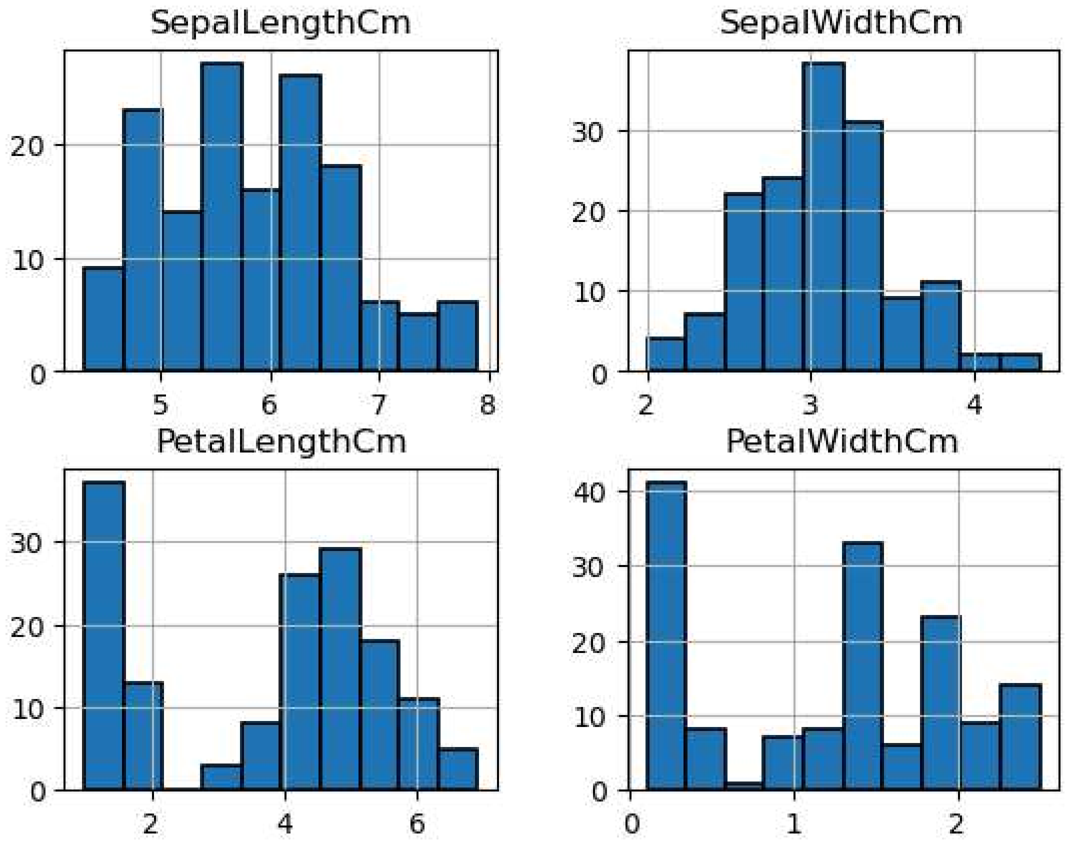
```
fig=plt.gcf
plt.figure(figsize=(15,10))

fig=sns.heatmap(iris.corr(),annot=True,cmap='cubehelix',linewidths=1,linecolor='k',
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
Cell In[33], line 4
      1 fig=plt.gcf
      2 plt.figure(figsize=(15,10))
----> 4 fig=sns.heatmap(iris.corr(),annot=True,cmap='cubehelix',linewidths=1,linecol
or='k',square=True,mask=False, vmin=-1, vmax=1,cbar_kws={"orientation": "vertical"},
cbar=True)

File ~\anaconda3\Lib\site-packages\pandas\core\frame.py:11049, in DataFrame.corr(sel
f, method, min_periods, numeric_only)
  11047 cols = data.columns
  11048 idx = cols.copy()
> 11049 mat = data.to_numpy(dtype=float, na_value=np.nan, copy=False)
  11051 if method == "pearson":
  11052     correl = libalgos.nancorr(mat, minp=min_periods)

File ~\anaconda3\Lib\site-packages\pandas\core\frame.py:1993, in DataFrame.to_numpy
(self, dtype, copy, na_value)
   1991 if dtype is not None:
   1992     dtype = np.dtype(dtype)
-> 1993 result = self._mgr.as_array(dtype=dtype, copy=copy, na_value=na_value)
   1994 if result.dtype is not dtype:
   1995     result = np.asarray(result, dtype=dtype)

File ~\anaconda3\Lib\site-packages\pandas\core\internals\managers.py:1694, in BlockM
anager.as_array(self, dtype, copy, na_value)
   1692         arr.flags.writeable = False
   1693 else:
-> 1694     arr = self._interleave(dtype=dtype, na_value=na_value)
   1695     # The underlying data was copied within _interleave, so no need
   1696     # to further copy if copy=True or setting na_value
   1698 if na_value is lib.no_default:

File ~\anaconda3\Lib\site-packages\pandas\core\internals\managers.py:1753, in BlockM
anager._interleave(self, dtype, na_value)
   1751     else:
   1752         arr = blk.get_values(dtype)
-> 1753     result[rl.indexer] = arr
   1754     itemmask[rl.indexer] = 1
   1756 if not itemmask.all():

ValueError: could not convert string to float: 'Iris-setosa'
```

12. Distribution plot:

In [34]:

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

```
<Figure size 1500x1000 with 0 Axes>
```



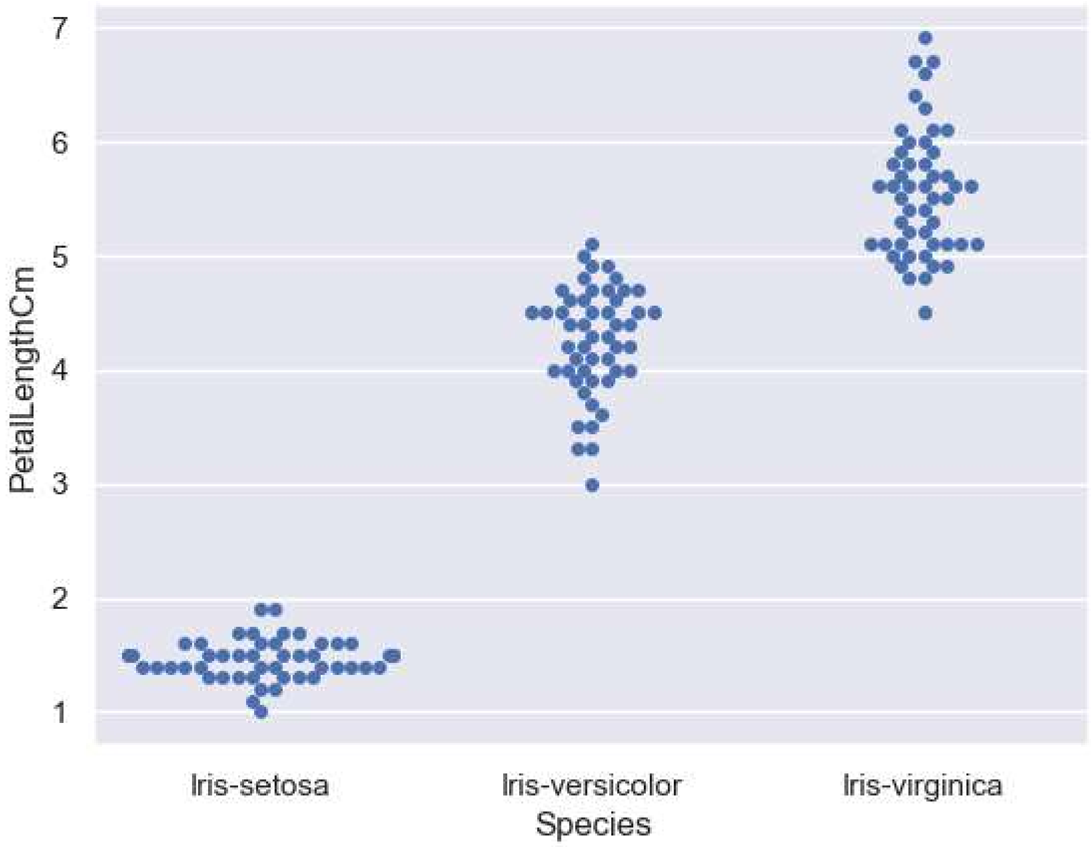
```
<Figure size 1200x600 with 0 Axes>
```

13. Swarm plot

In [36]:

```
sns.set(style="darkgrid")
fig=plt.gcf()
plt.figure(figure=(12,6))
fig=sns.swarmplot(x="Species",y="PetalLengthCm",data=iris)
plt.show()
```

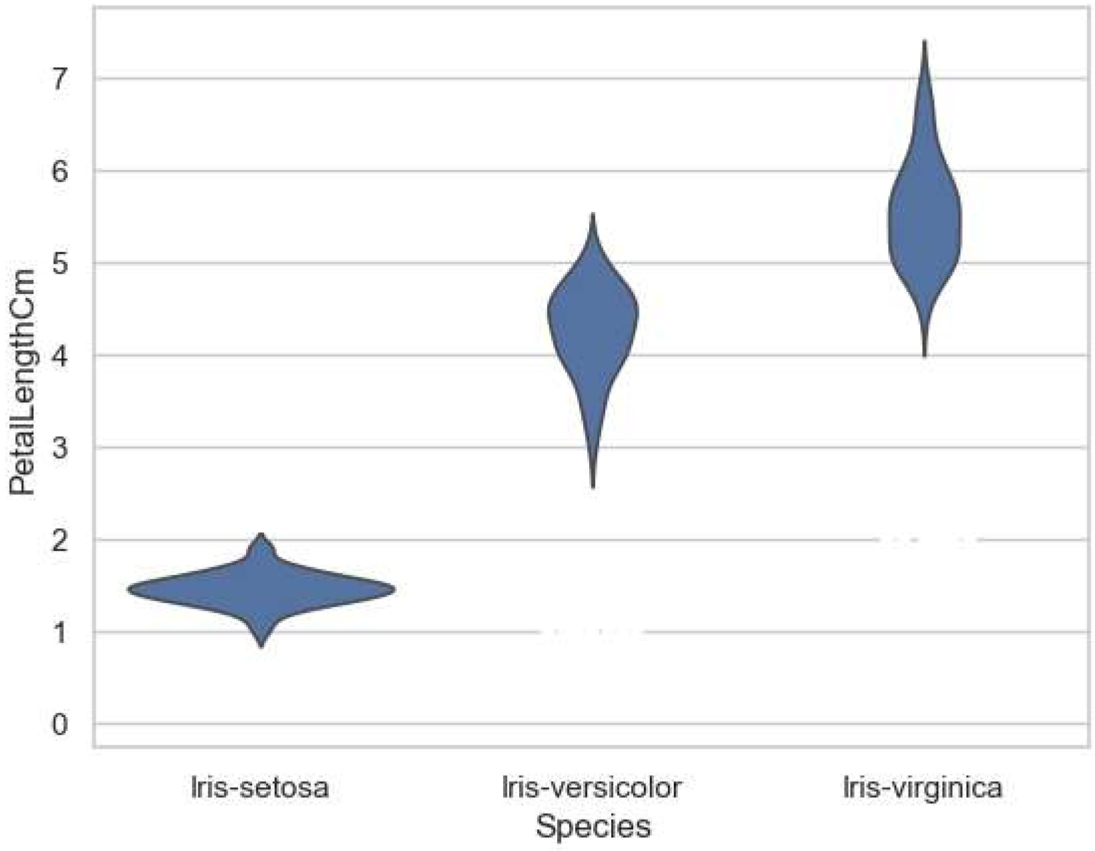
```
<Figure size 640x480 with 0 Axes>
```



In [38]:

```
sns.set(style="whitegrid")
fig=plt.gcf()
plt.figure(figure=(10,7))
ax=sns.violinplot(x="Species",y="PetalLengthCm",data=iris,inner=None)
ax=sns.swarmplot(x="Species",y="PetalWidthCm",data=iris,color="white",edgecolor="bl
plt.show()
```

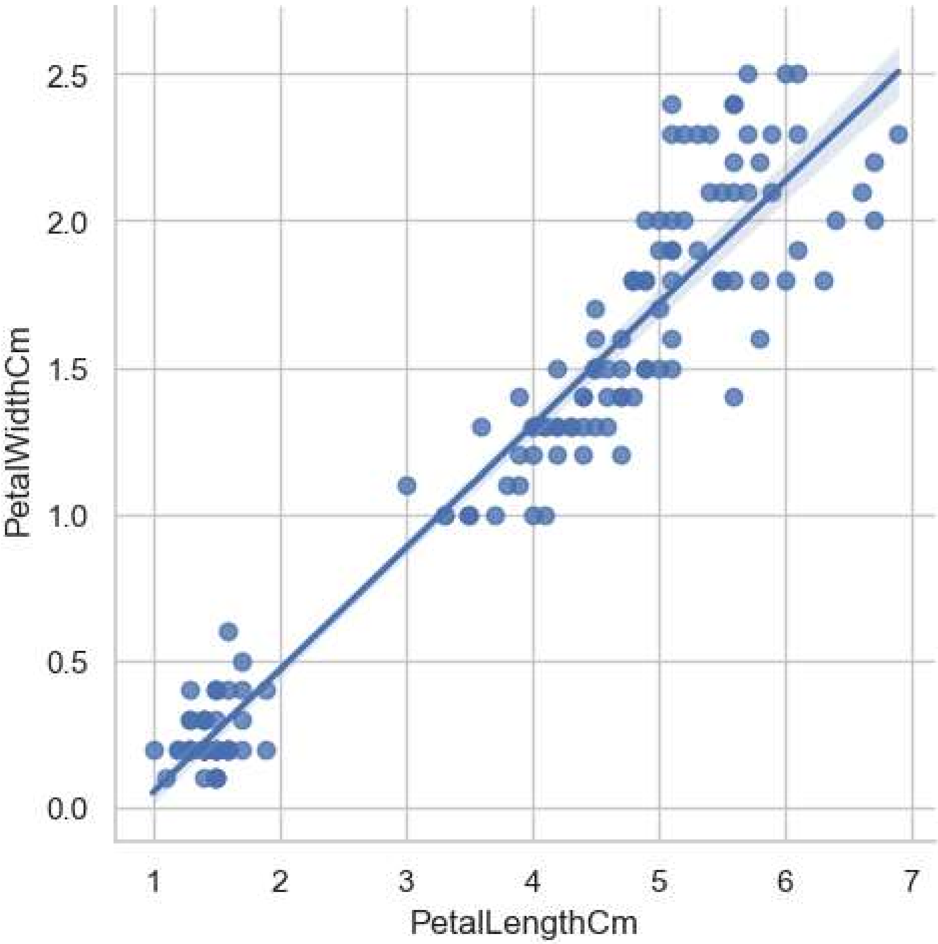
```
<Figure size 640x480 with 0 Axes>
```



17. LM PLot

In [40]:

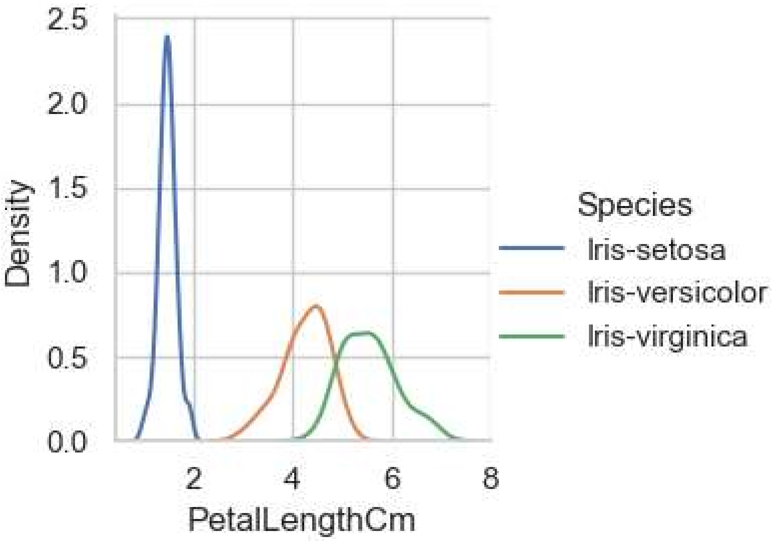
```
fig=sns.lmplot(x="PetalLengthCm",y="PetalWidthCm",data=iris)
plt.show()
```



18. FacetGrid

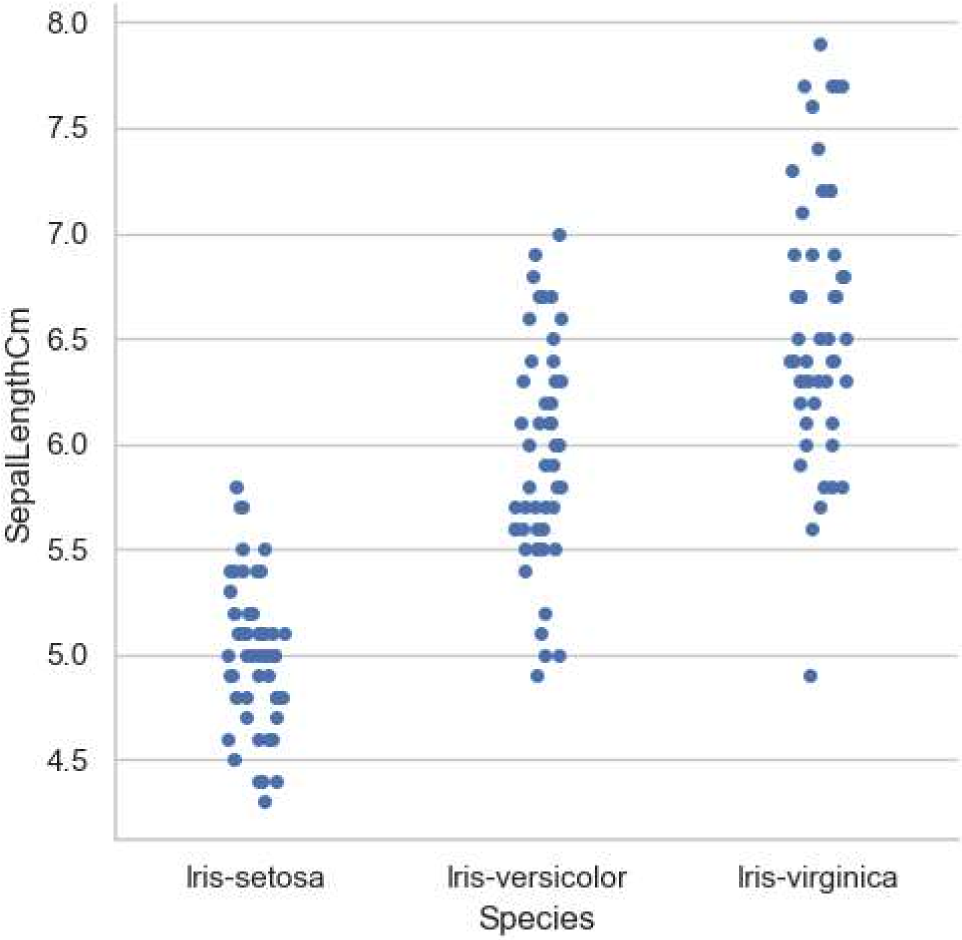
In [42]:

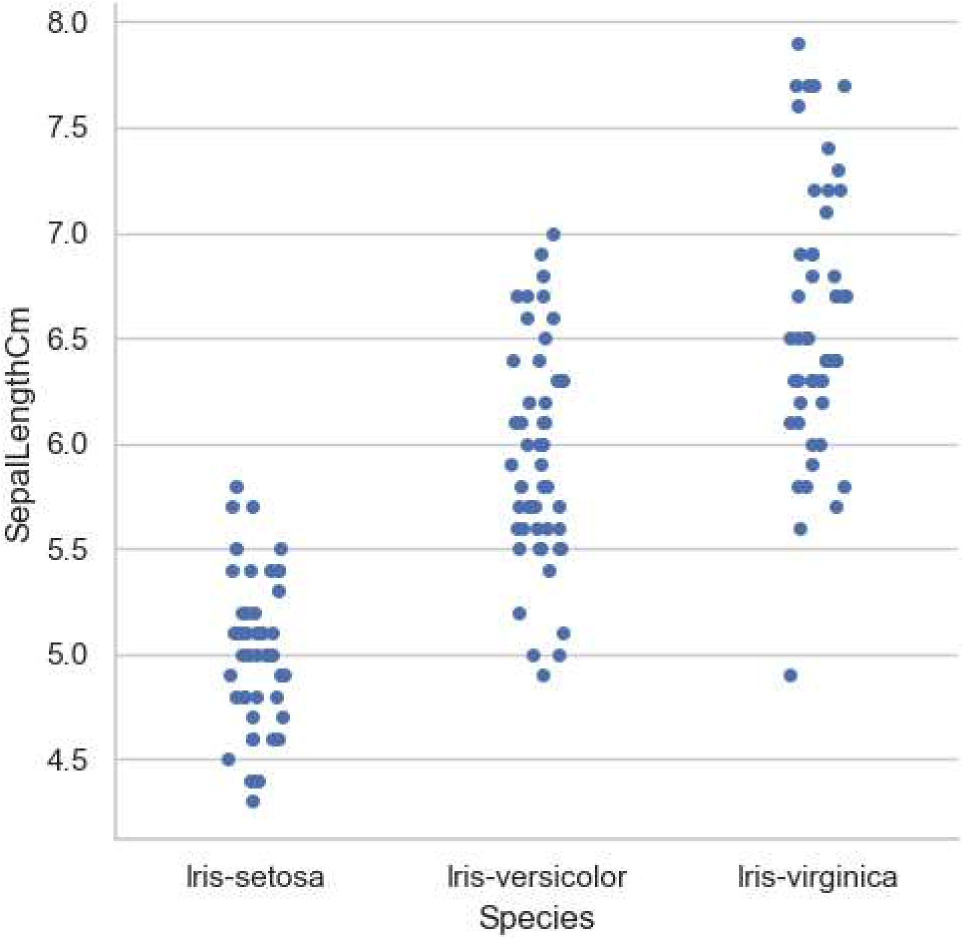
```
sns.FacetGrid(iris,hue='Species') \
   .map(sns.kdeplot,'PetalLengthCm') \
   .add_legend()
plt.ioff()
plt.show()
```



19. Factor Plot

In [77]:
```
import seaborn as sns
import matplotlib.pyplot as plt
sns.catplot(x='Species',y='SepalLengthCm',data=iris)
plt.show()
```

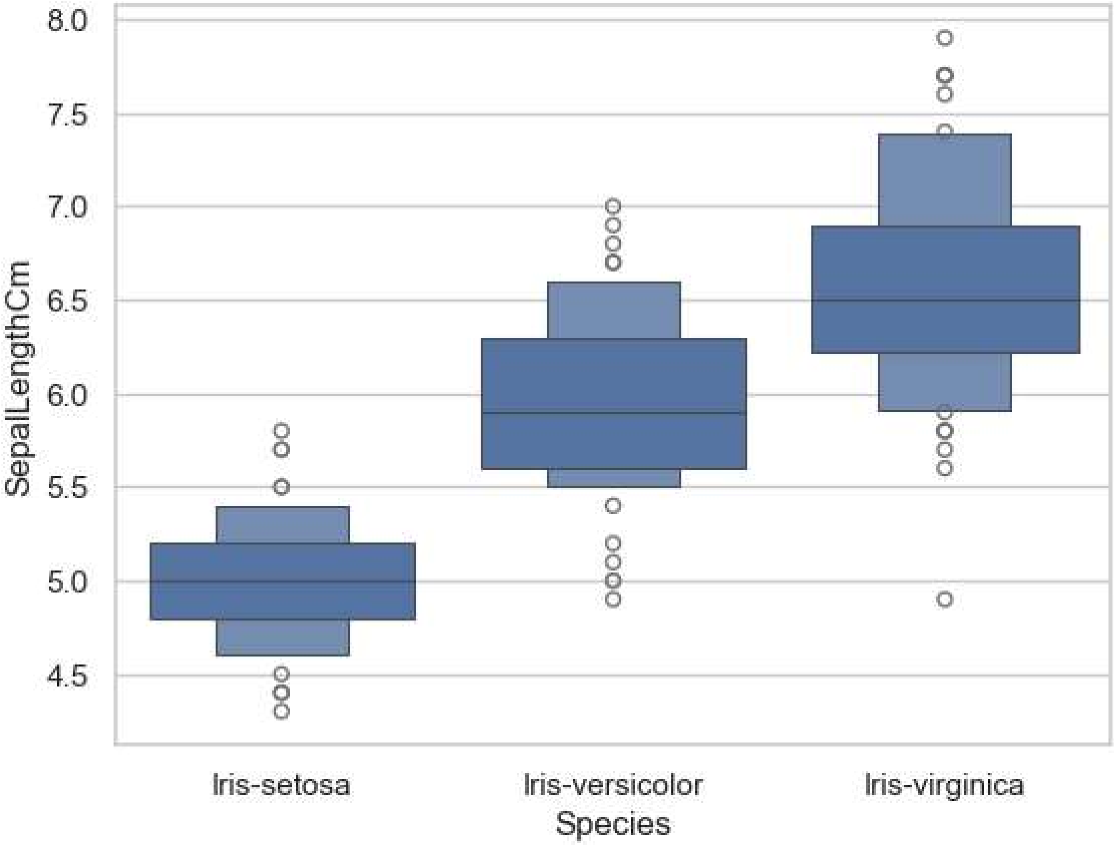




20.boxplot

```
In [48]: fig=plt.gcf()
         plt.figure(figure=(10,7))
         fig=sns.boxenplot(x="Species",y="SepalLengthCm",data=iris)
         plt.show()
```

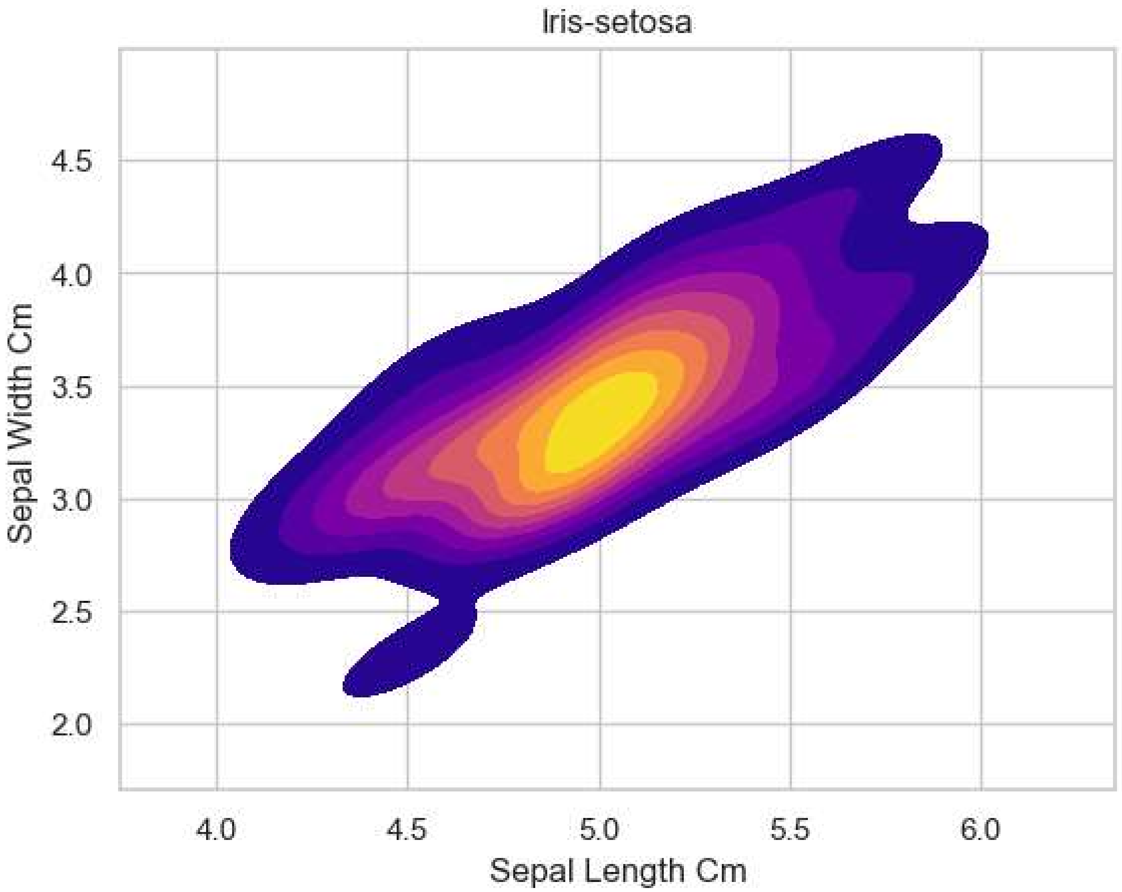
```
<Figure size 640x480 with 0 Axes>
```



21.KDE Plot

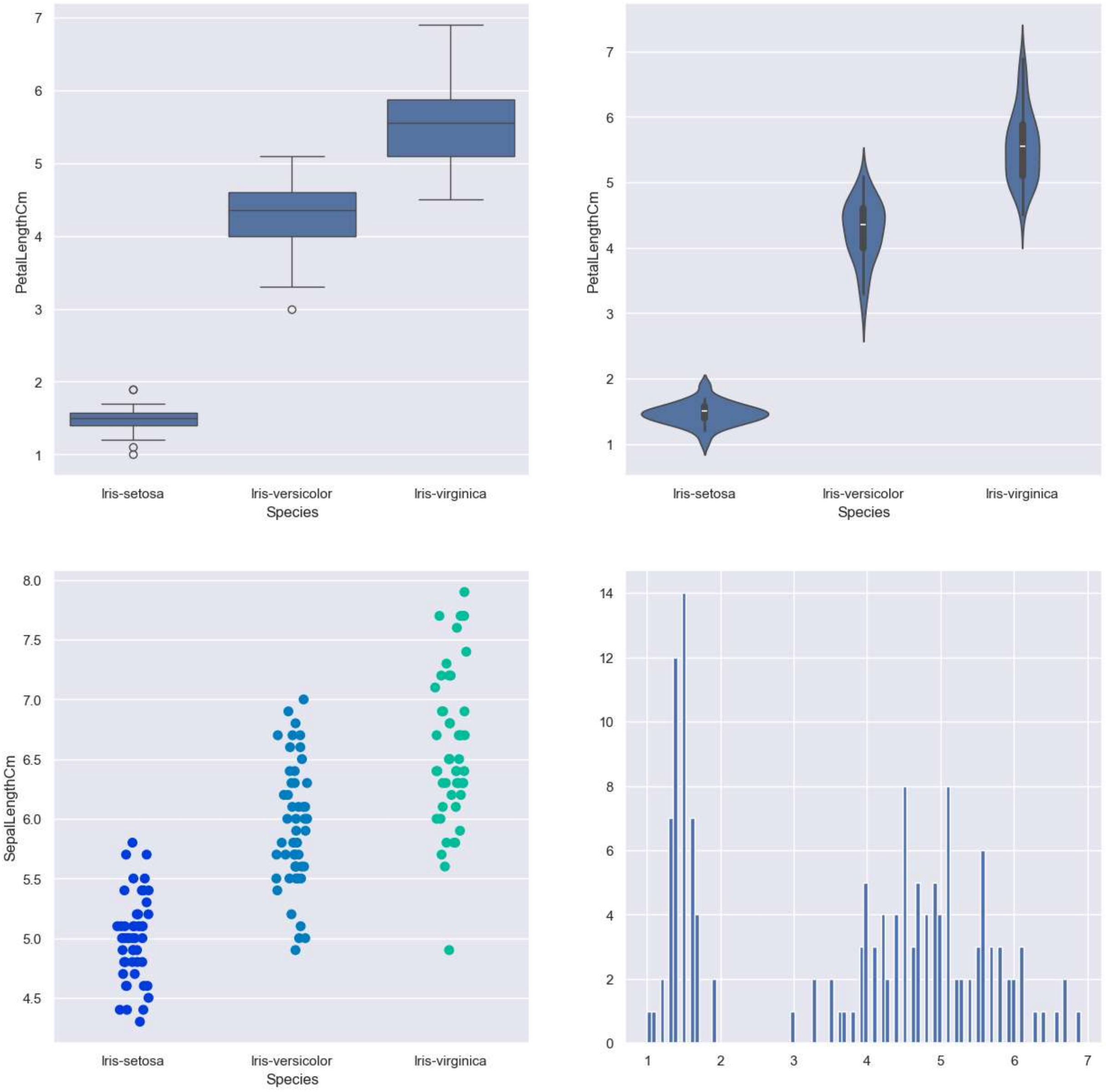
In [67]:

```
sub=iris[iris['Species']=='Iris-setosa']
sns.kdeplot(data=sub,x='SepalLengthCm',y='SepalWidthCm',cmap="plasma", shade=True,
plt.title('Iris-setosa')
plt.xlabel('Sepal Length Cm')
plt.ylabel('Sepal Width Cm')
plt.show()
```



In [81]:

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



In [85]:

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

In [87]:

```
iris.head()
```

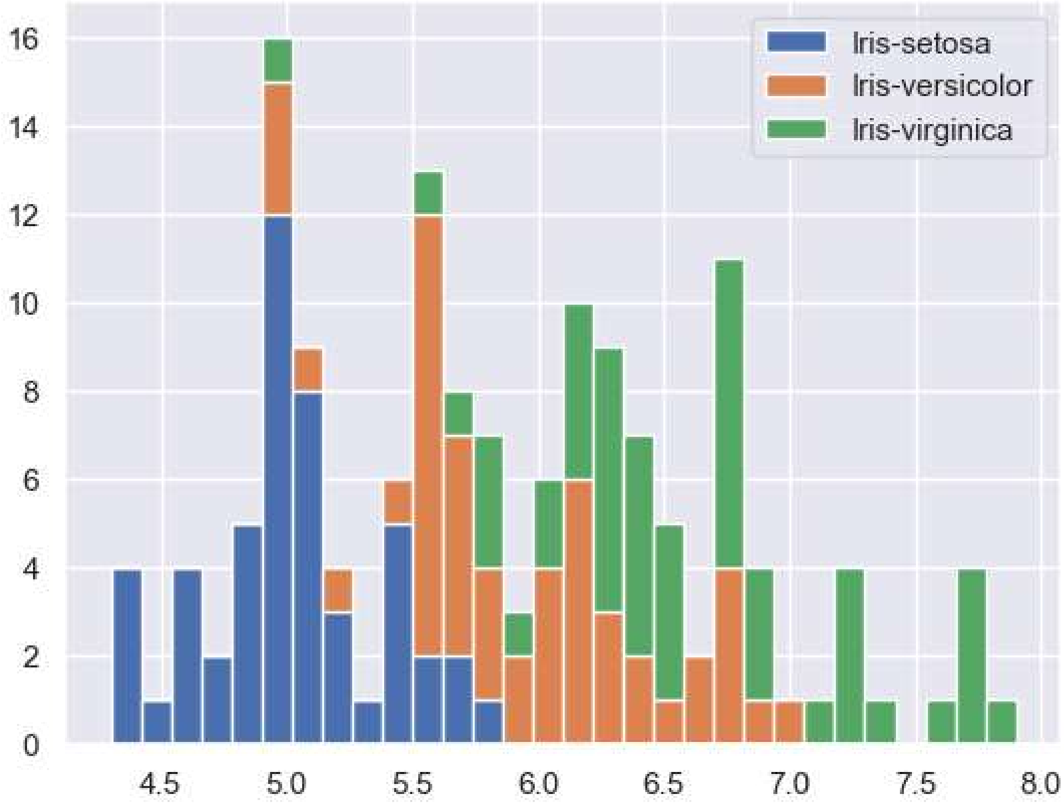
Out[87]:

	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

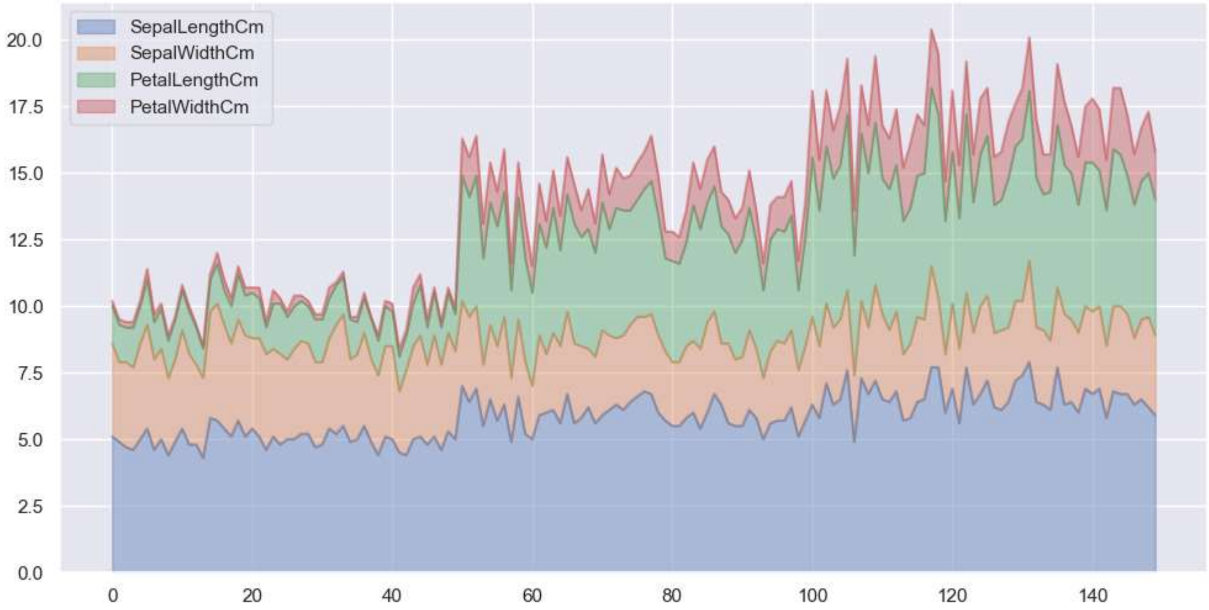
In [96]:

```
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()
```

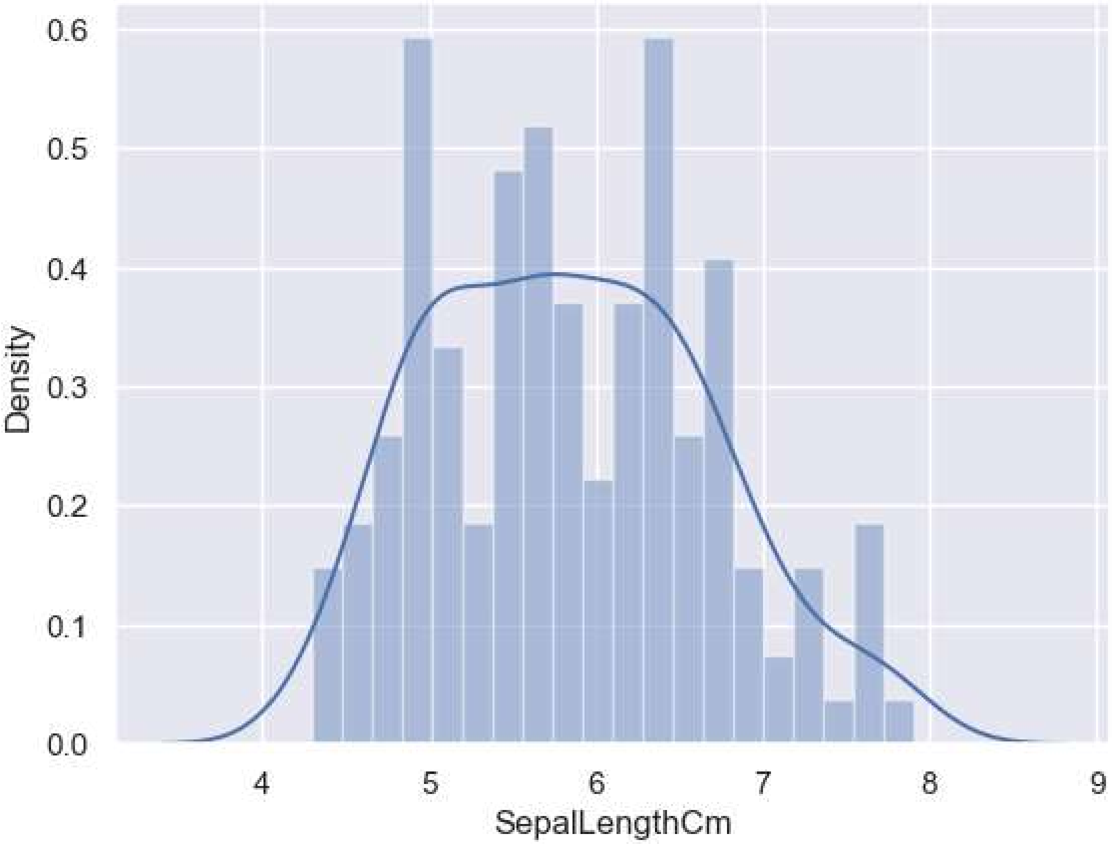


```
In [98]: iris.plot.area(y=['SepalLengthCm','SepalWidthCm','PetalLengthCm','PetalWidthCm'],al
         plt.show()
```



In [100…

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



In [ ]: