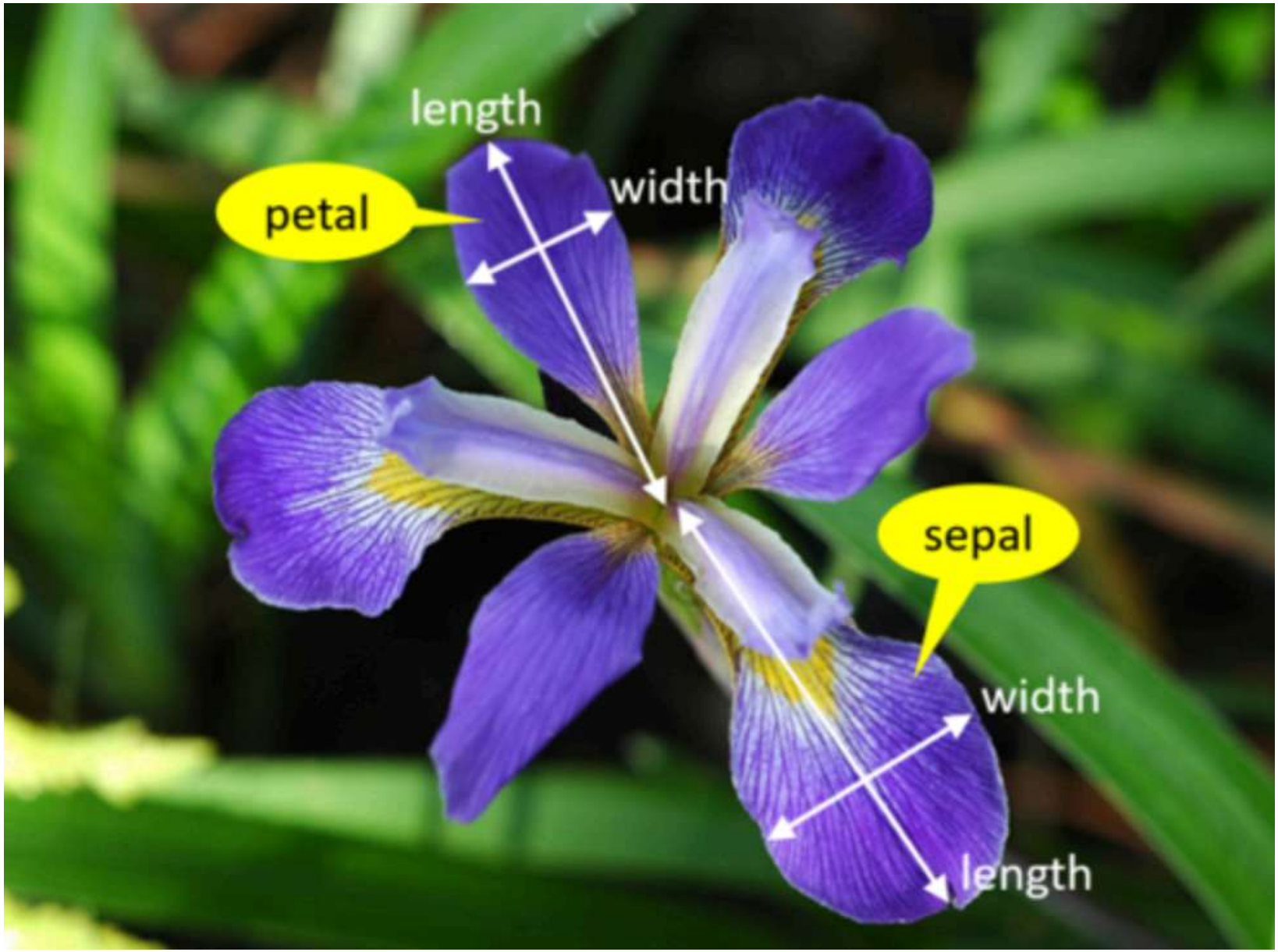
IRIS DATASET VISUALIZATION(SEABORN,MATPLOTLIB)



In [5]:
```
import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)
```

In [7]:
```
import seaborn as sns
import matplotlib.pyplot as plt
import warnings
warnings.filterwarnings('ignore')  #this will ignore the warnings.it wont display warnings in notebook
```

Importing Iris data set

In [14]:
```
iris=pd.read_csv(r"C:\Users\jayes\OneDrive\Desktop\NareshIT\28_mar\28th - Iris, movie analytics Project\28th -Seaborr
```

In [16]:
```
iris
```

Out[16]:

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

Out[18]:

	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 [22]:
```
iris.drop('Id',axis=1,inplace=True)
```

In [24]:
```
iris
```

Out[24]:

	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
...	...	...	...	...	...
145	6.7	3.0	5.2	2.3	Iris-virginica
146	6.3	2.5	5.0	1.9	Iris-virginica
147	6.5	3.0	5.2	2.0	Iris-virginica
148	6.2	3.4	5.4	2.3	Iris-virginica
149	5.9	3.0	5.1	1.8	Iris-virginica

150 rows × 5 columns

In [26]:
```
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
```

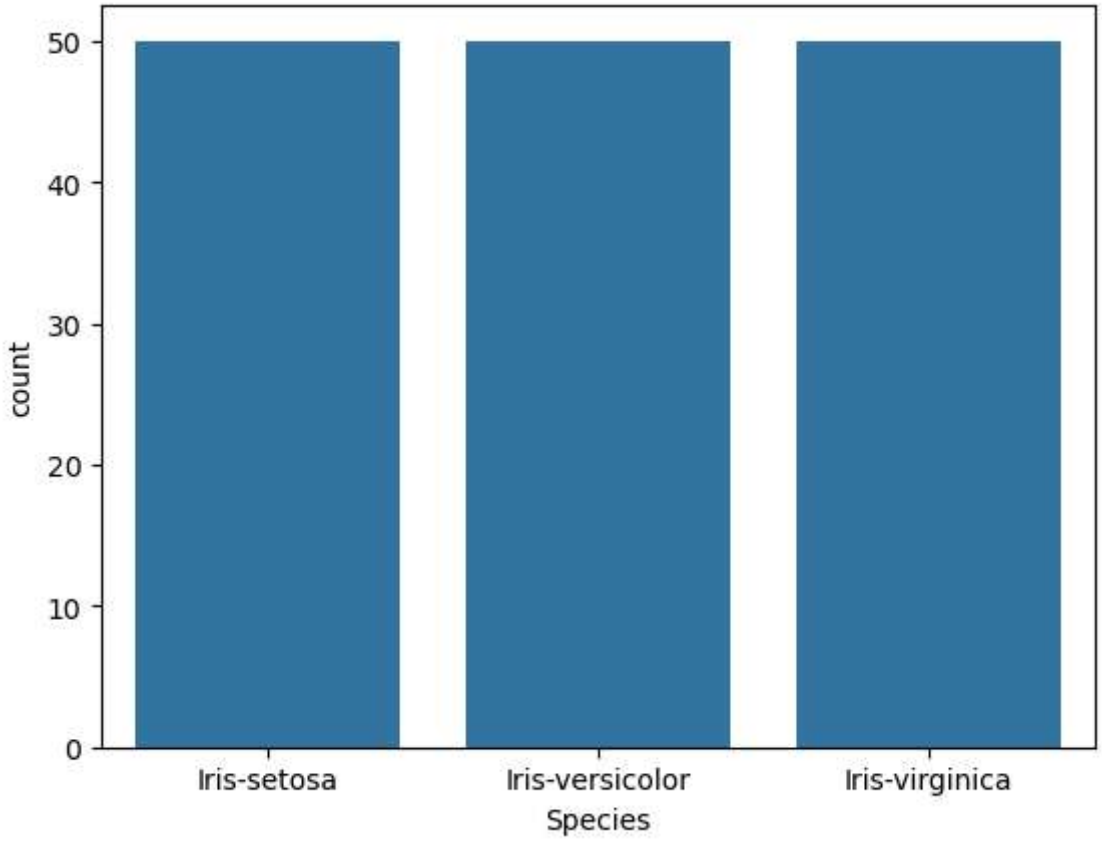
In [28]:
```
iris['Species'].value_counts()
```

Out[28]:
```
Species
Iris-setosa        50
Iris-versicolor    50
Iris-virginica     50
Name: count, dtype: int64
```

In [30]:
```
# This data set has three varities of Iris plant.
```

In [32]:
```
# 2.Bar Plot : Here the frequency of the observation is plotted.In this case we are plotting the frequency of the thr
```

In [46]:
```
sns.countplot(x=iris['Species'])   #default is horizontall
plt.show()
```

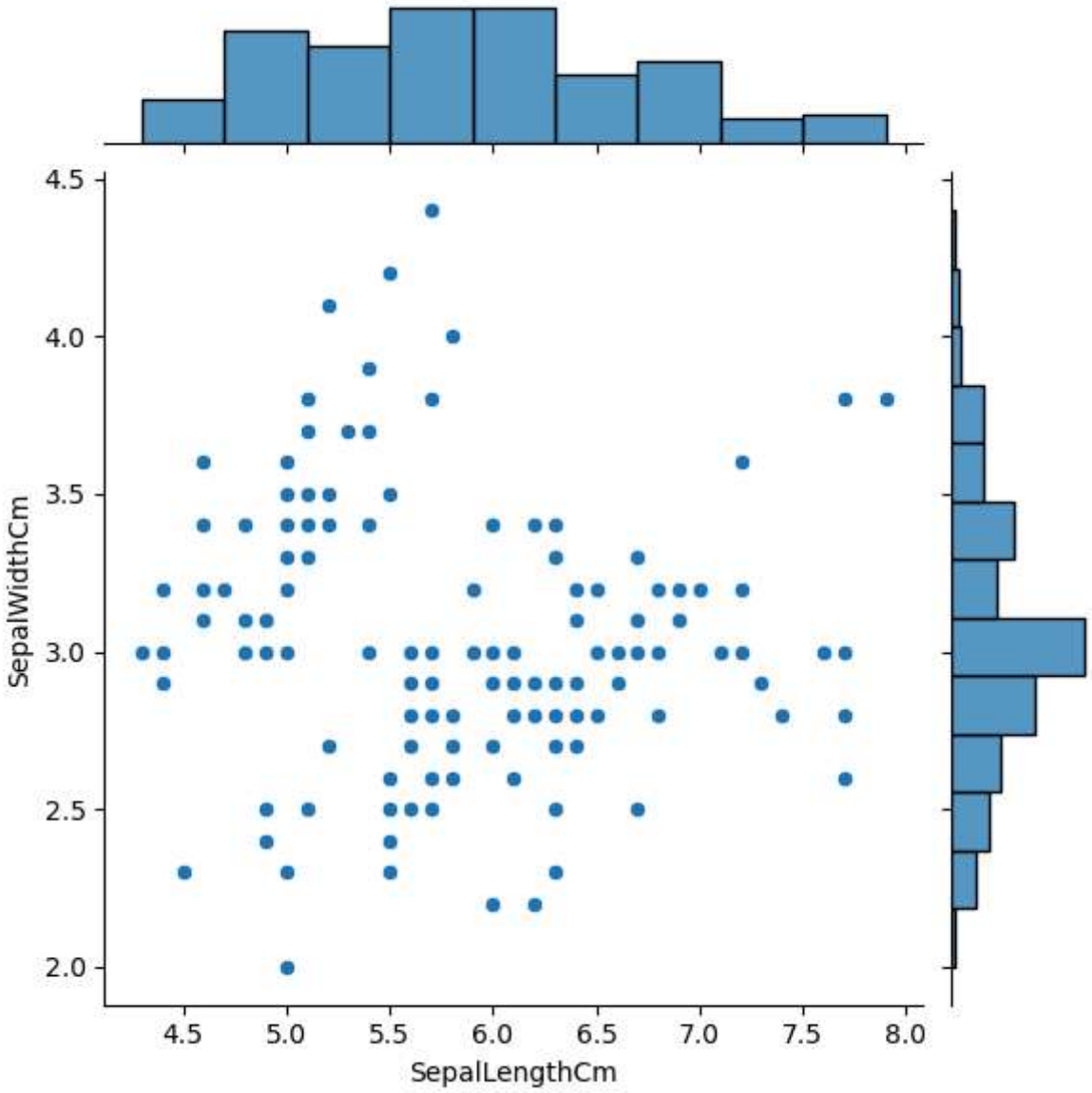


In [48]:
```
iris.head()
```

Out[48]:

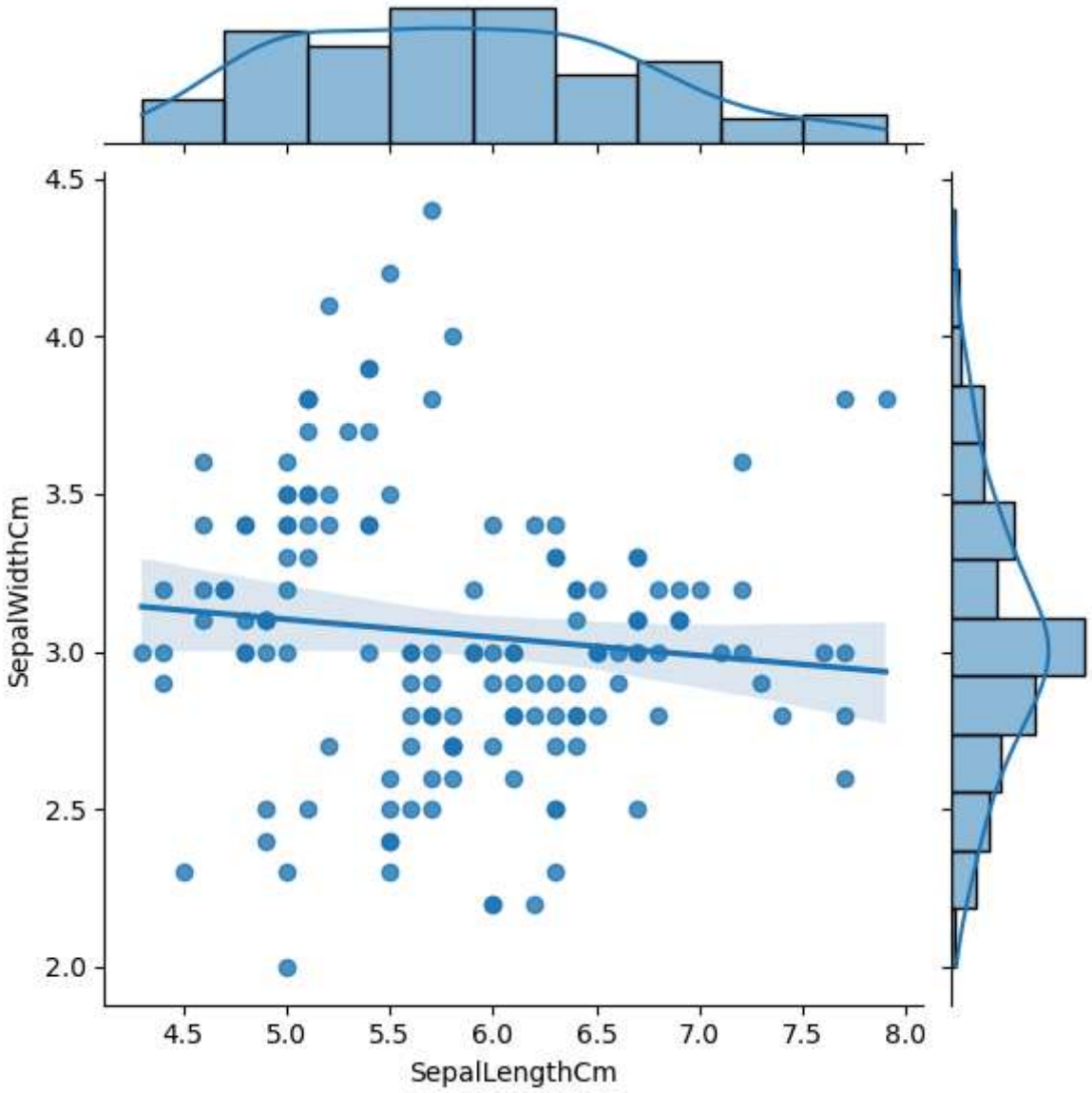
	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	5.1	3.5	1.4	0.2	Iris-setosa
1	4.9	3.0	1.4	0.2	Iris-setosa
2	4.7	3.2	1.3	0.2	Iris-setosa
3	4.6	3.1	1.5	0.2	Iris-setosa
4	5.0	3.6	1.4	0.2	Iris-setosa

In [52]:
```
fig=sns.jointplot(x='SepalLengthCm',y='SepalWidthCm',data=iris)
```

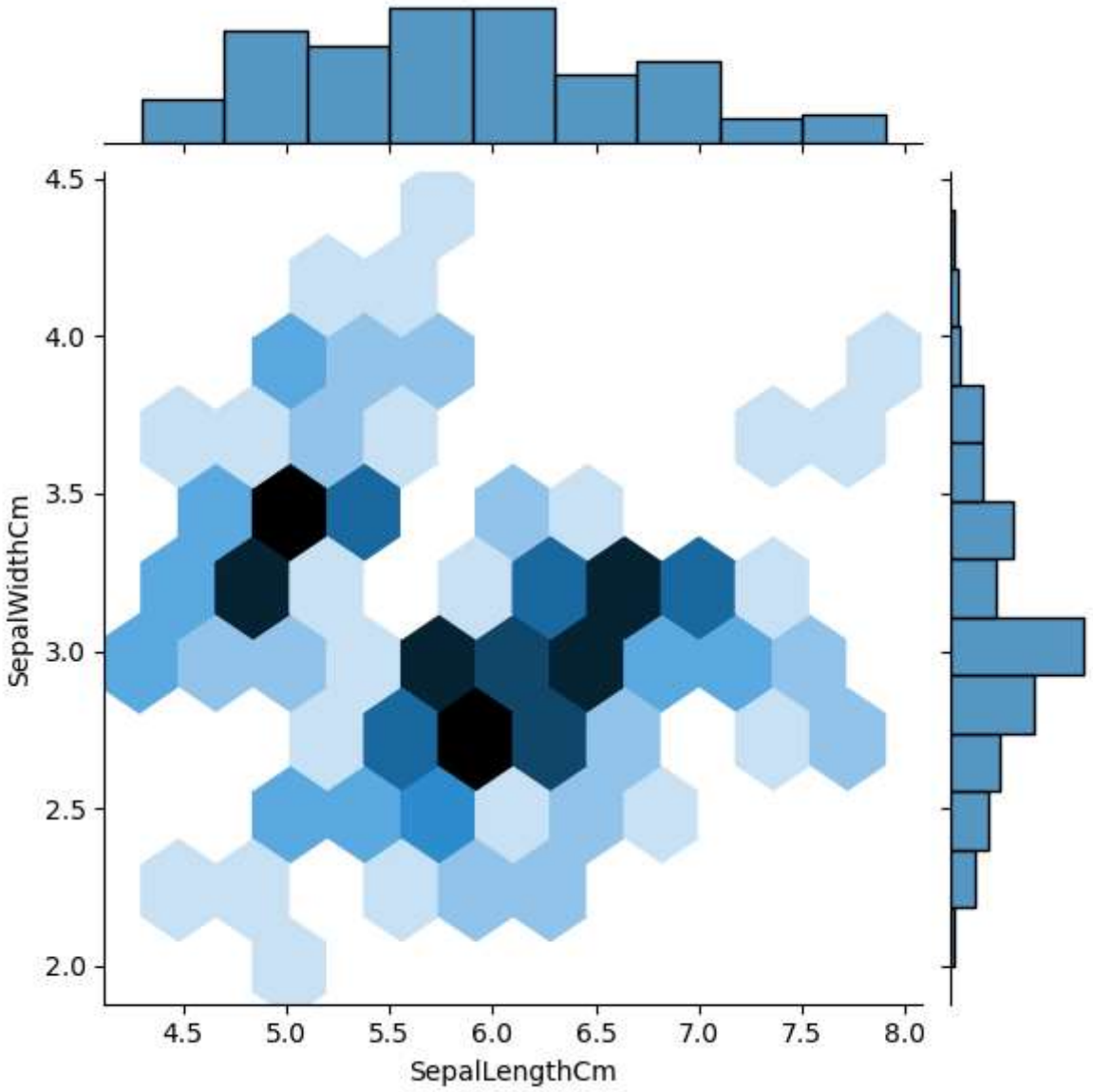


In [56]:
```
sns.jointplot(x="SepalLengthCm", y="SepalWidthCm", data=iris, kind="reg")
```

Out[56]: <seaborn.axisgrid.JointGrid at 0x1edd0347cb0>

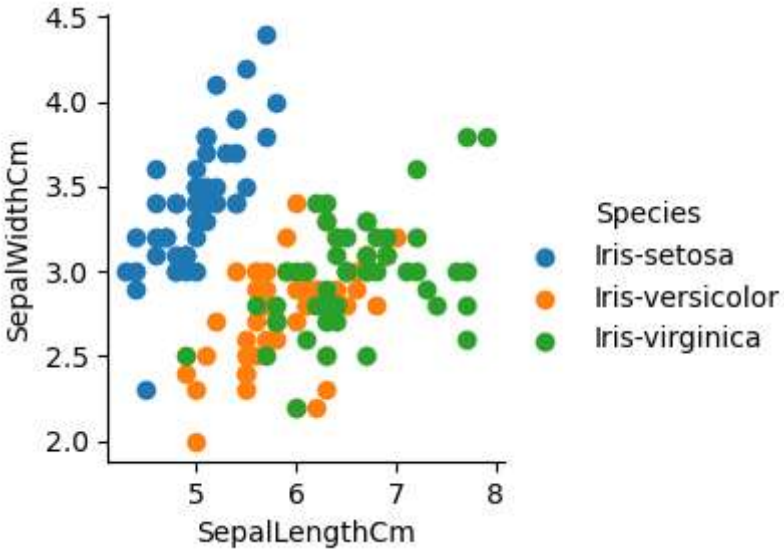


In [58]:
```
fig=sns.jointplot(x='SepalLengthCm',y='SepalWidthCm',kind='hex',data=iris)
```



In [66]:
```
import matplotlib.pyplot as plt
%matplotlib inline

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

```
.add_legend()
plt.show()
```

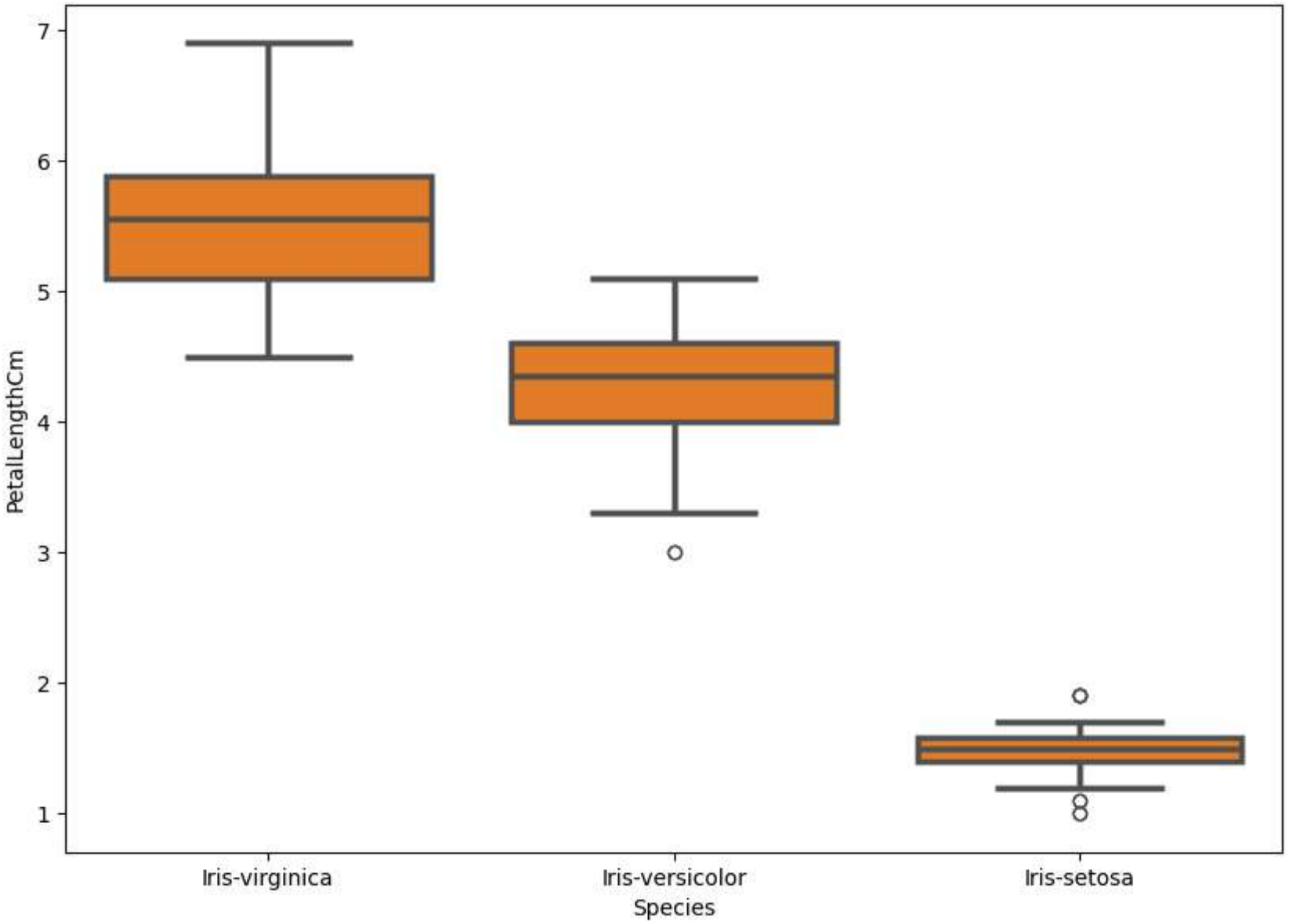


In [68]:
```
iris.head()
```

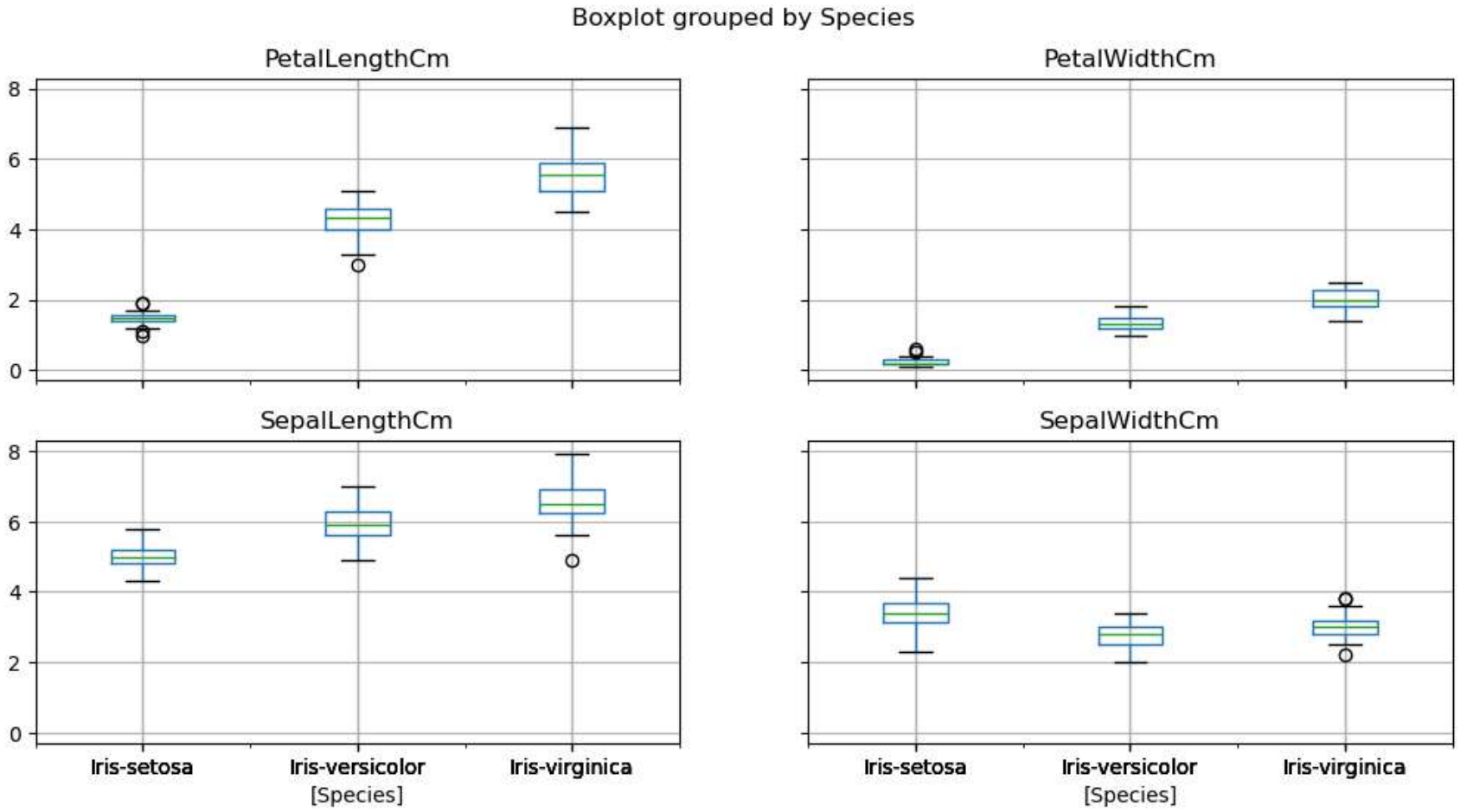
Out[68]:

	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

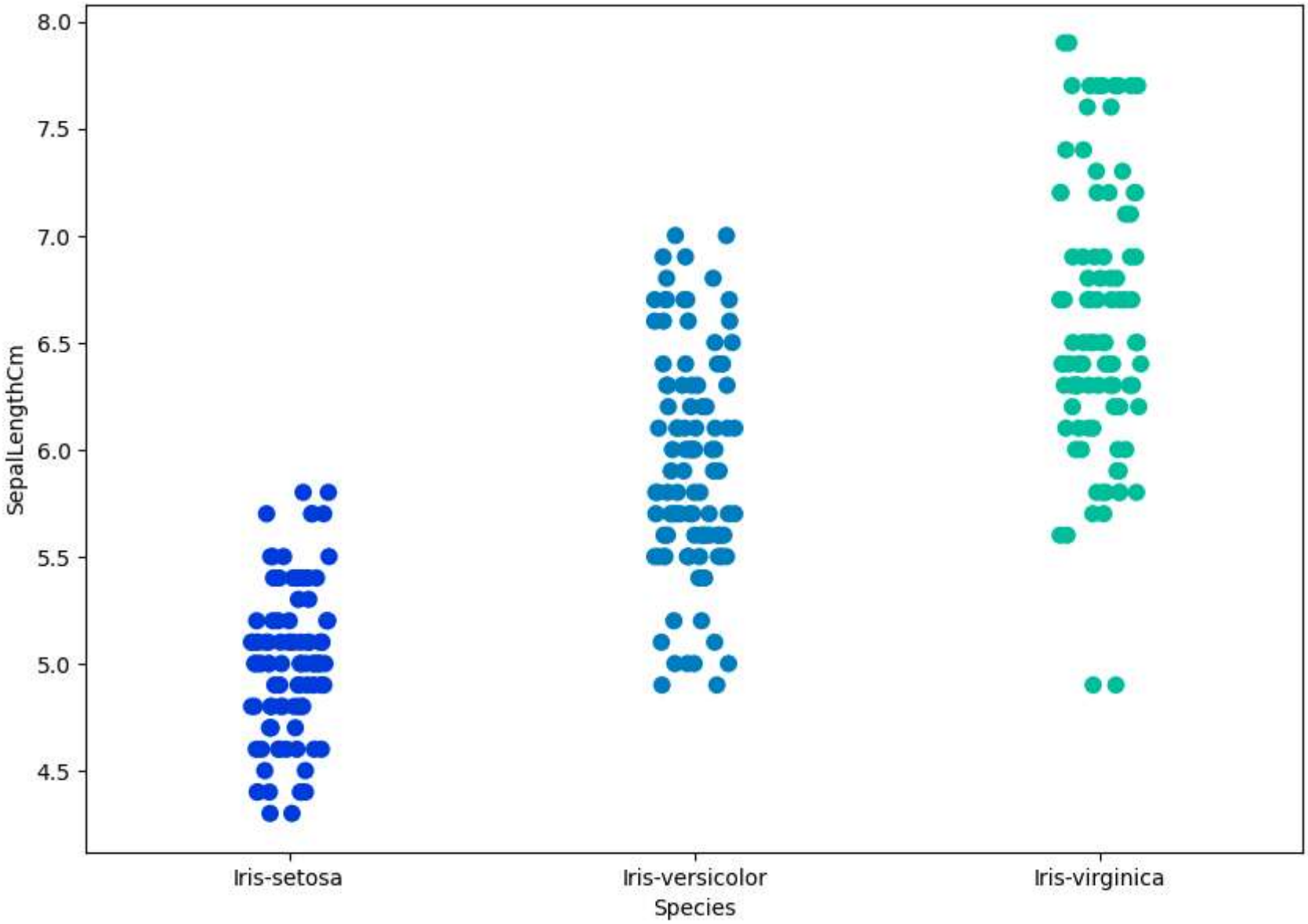
In [72]:
```
fig=plt.gcf()
fig.set_size_inches(10,7)
fig=sns.boxplot(x='Species',y='PetalLengthCm',data=iris,order=['Iris-virginica','Iris-versicolor','Iris-setosa'],line
plt.show()
```



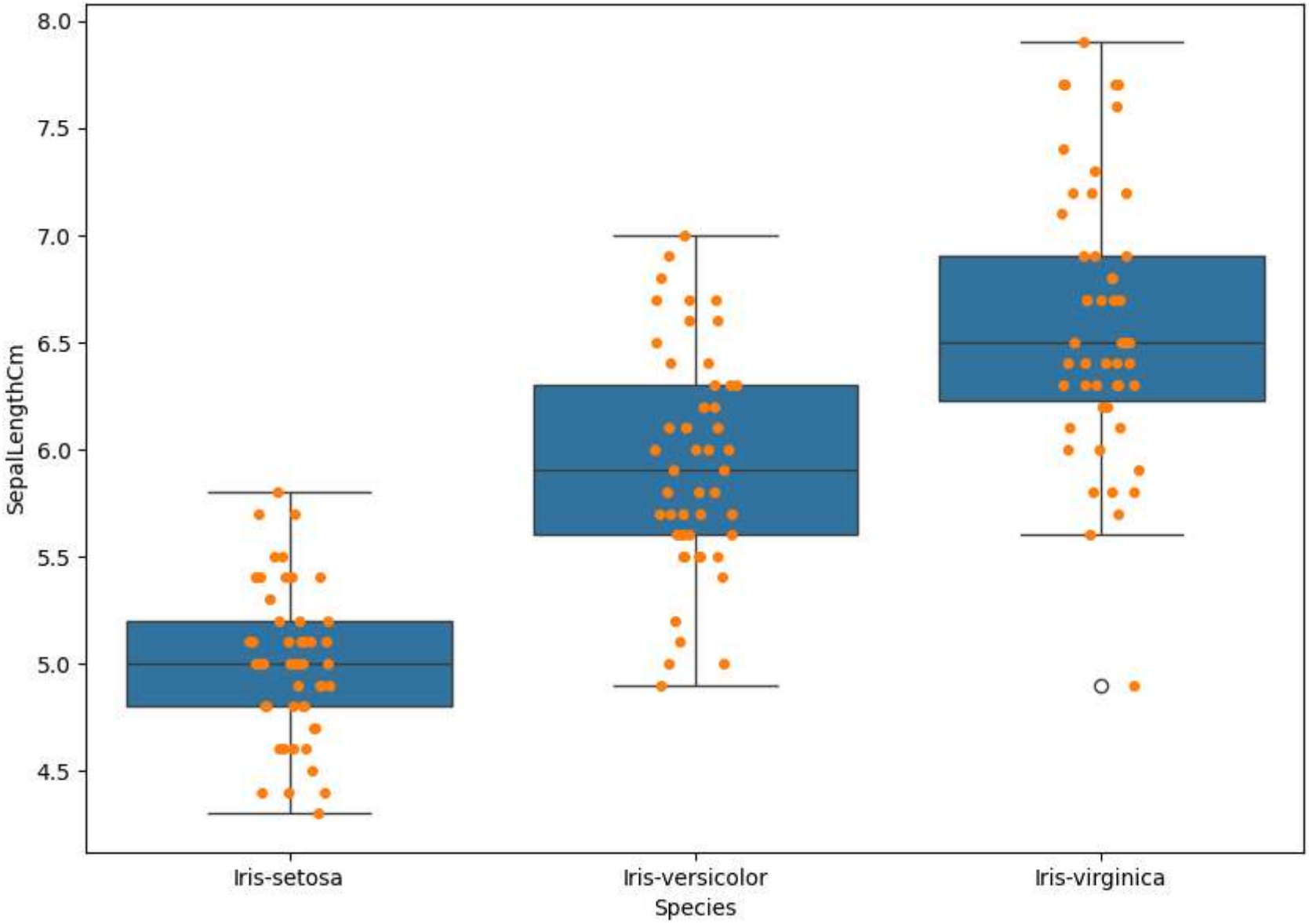
In [74]:
```
iris.boxplot(by="Species", figsize=(12, 6))
plt.show()
```



In [78]:
```
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
plt.show()
```

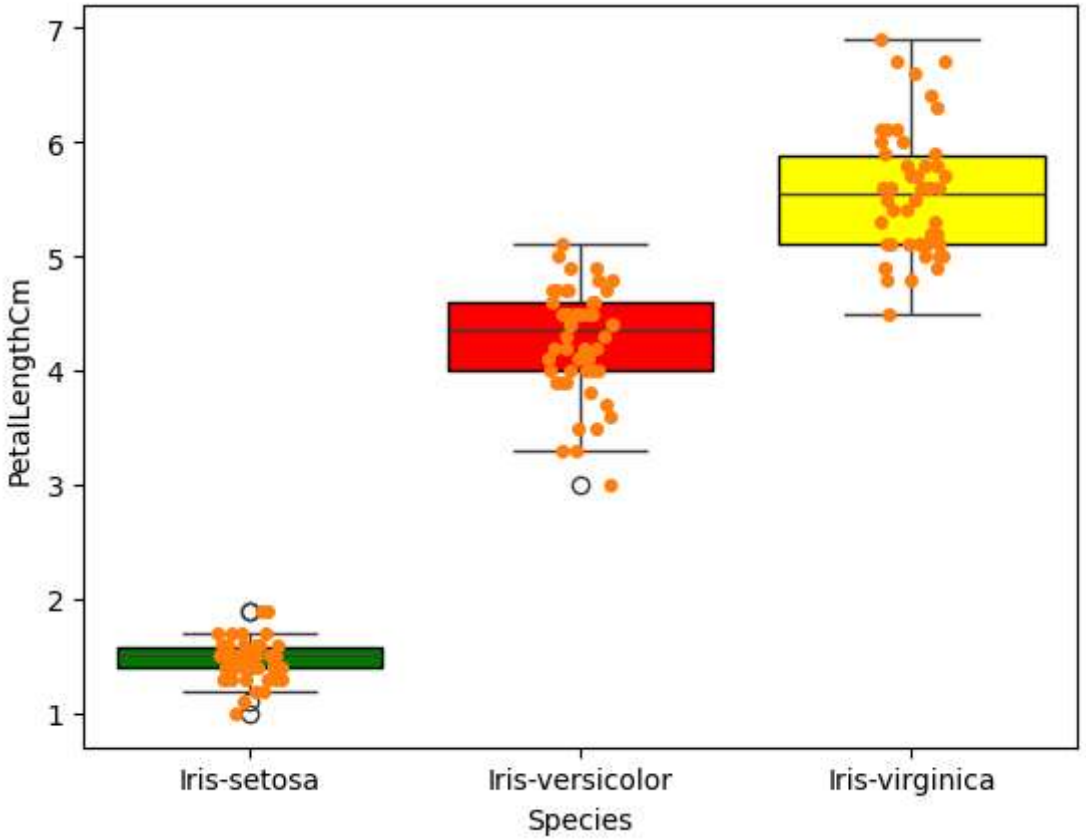


In [80]:
```
fig=plt.gcf()
fig.set_size_inches(10,7)
fig=sns.boxplot(x='Species',y='SepalLengthCm',data=iris)
fig=sns.stripplot(x='Species',y='SepalLengthCm',data=iris,jitter=True,edgecolor='gray')
plt.show()
```

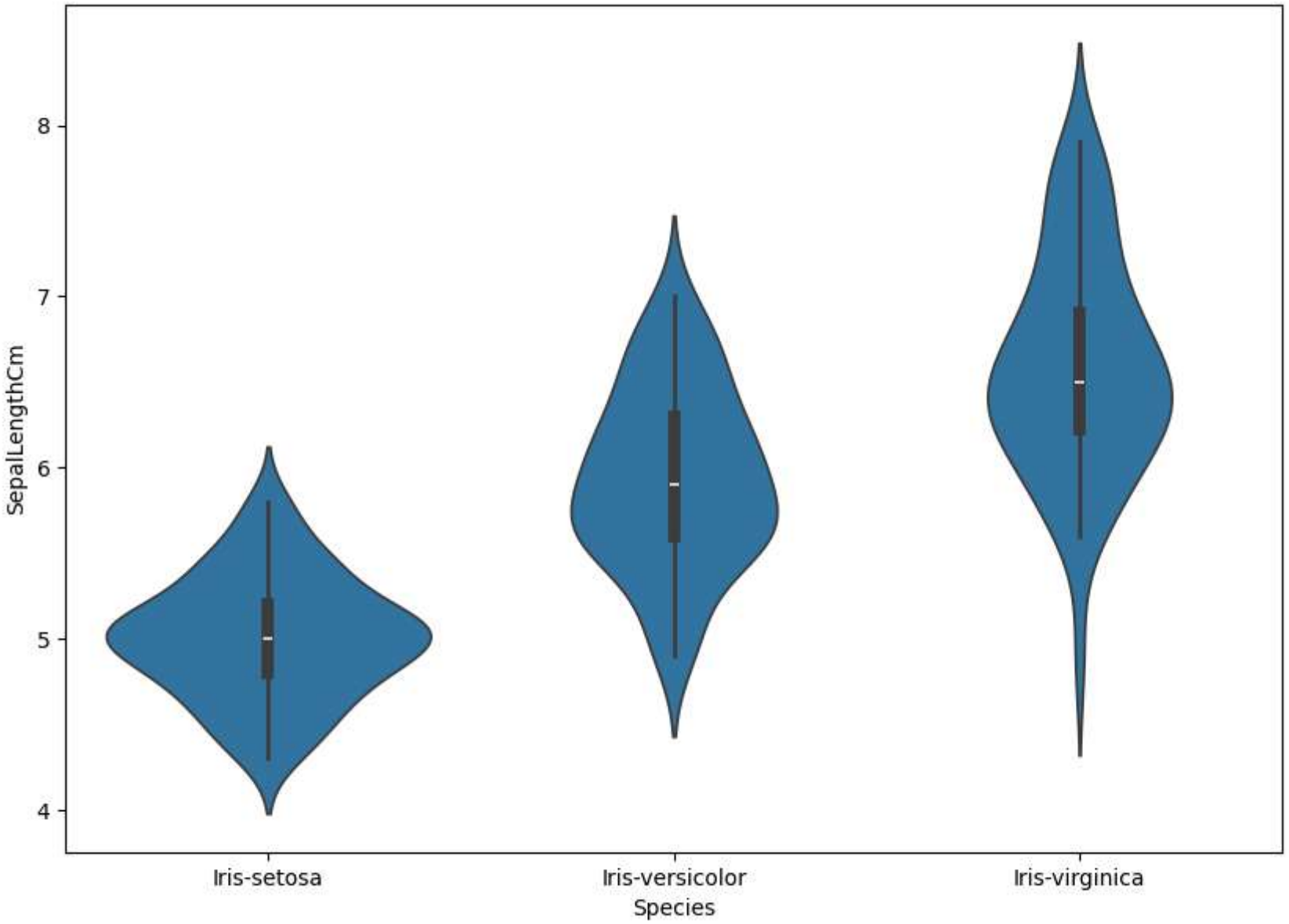


In [86]:
```
# Create the box plot
ax = sns.boxplot(x="Species", y="PetalLengthCm", data=iris)

# Overlay strip plot
sns.stripplot(x="Species", y="PetalLengthCm", data=iris, jitter=True, edgecolor="gray")
```

```
# Define colors for each species
colors = ['green', 'red', 'yellow']

# Change the colors of the boxes
for i, patch in enumerate(ax.patches[:3]):  # Ensure only the first three are changed
    patch.set_facecolor(colors[i])
    patch.set_edgecolor('black')

plt.show()
```

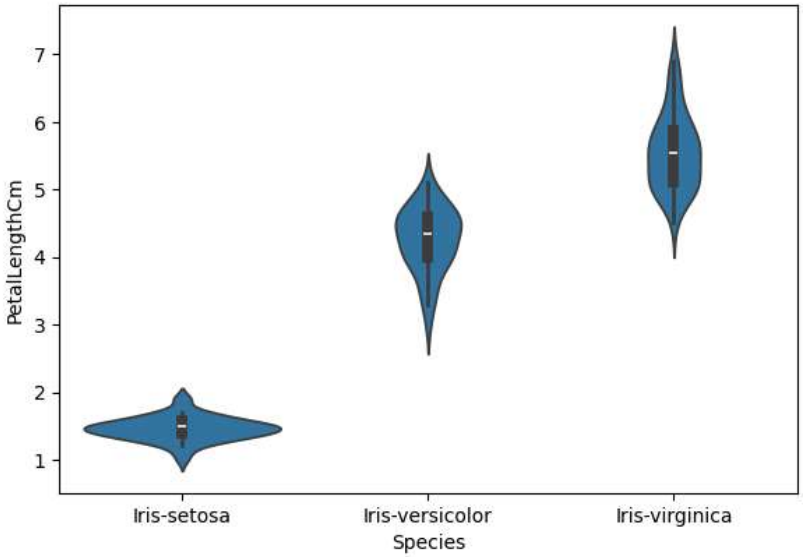


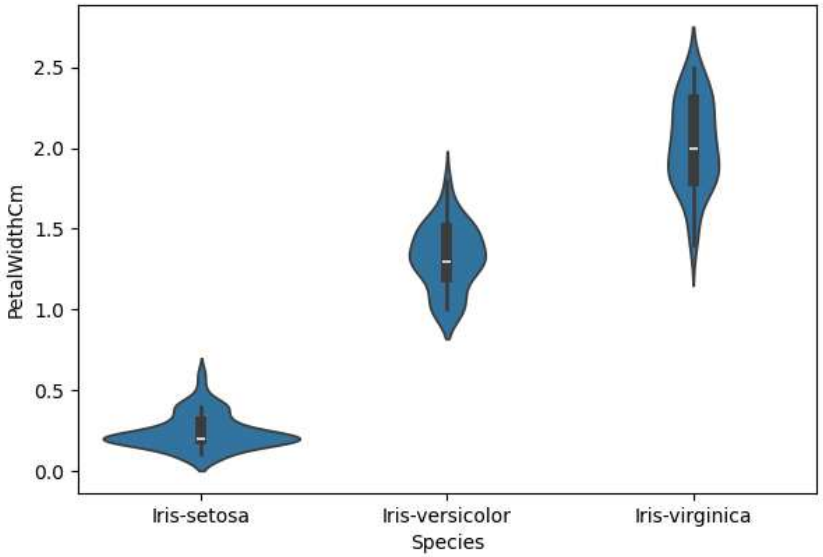
In [88]:
```
fig=plt.gcf()
fig.set_size_inches(10,7)
fig=sns.violinplot(x='Species',y='SepalLengthCm',data=iris)
plt.show()
```

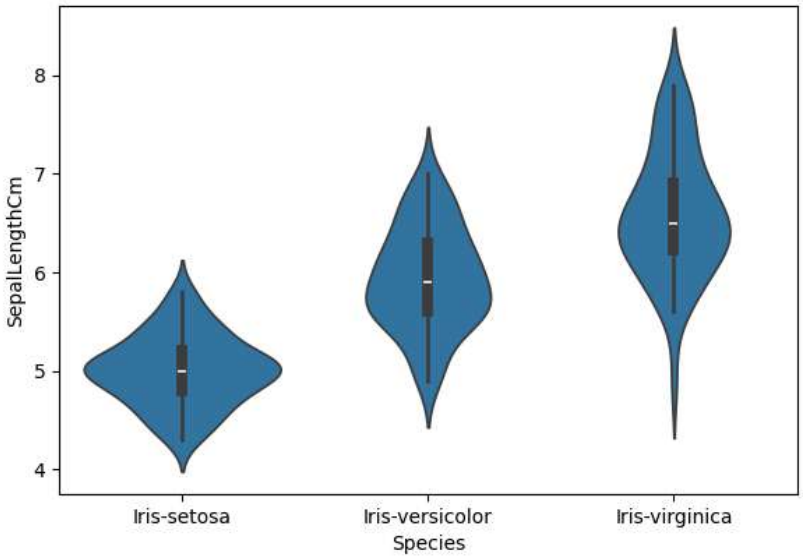


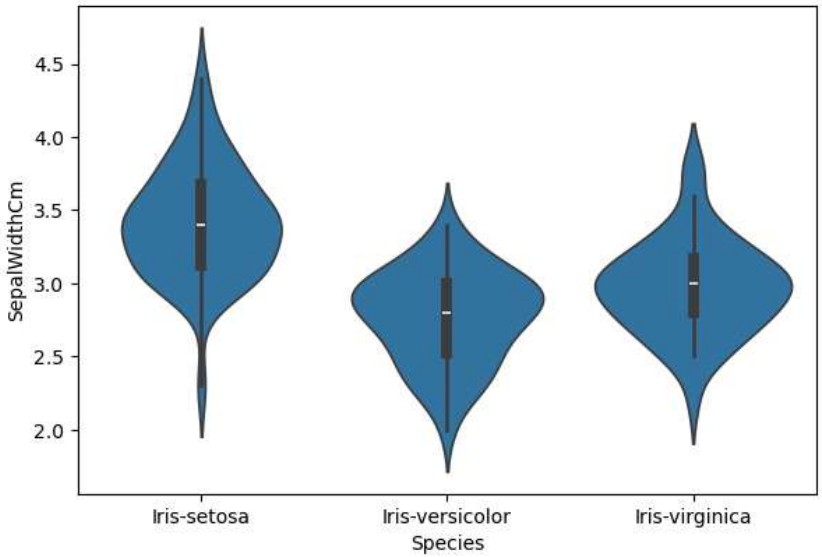
In [90]:
```
plt.figure(figsize=(15,10))
plt.subplot(2,2,1)
sns.violinplot(x='Species',y='PetalLengthCm',data=iris)
plt.subplot(2,2,2)
sns.violinplot(x='Species',y='PetalWidthCm',data=iris)
```

```
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()
```

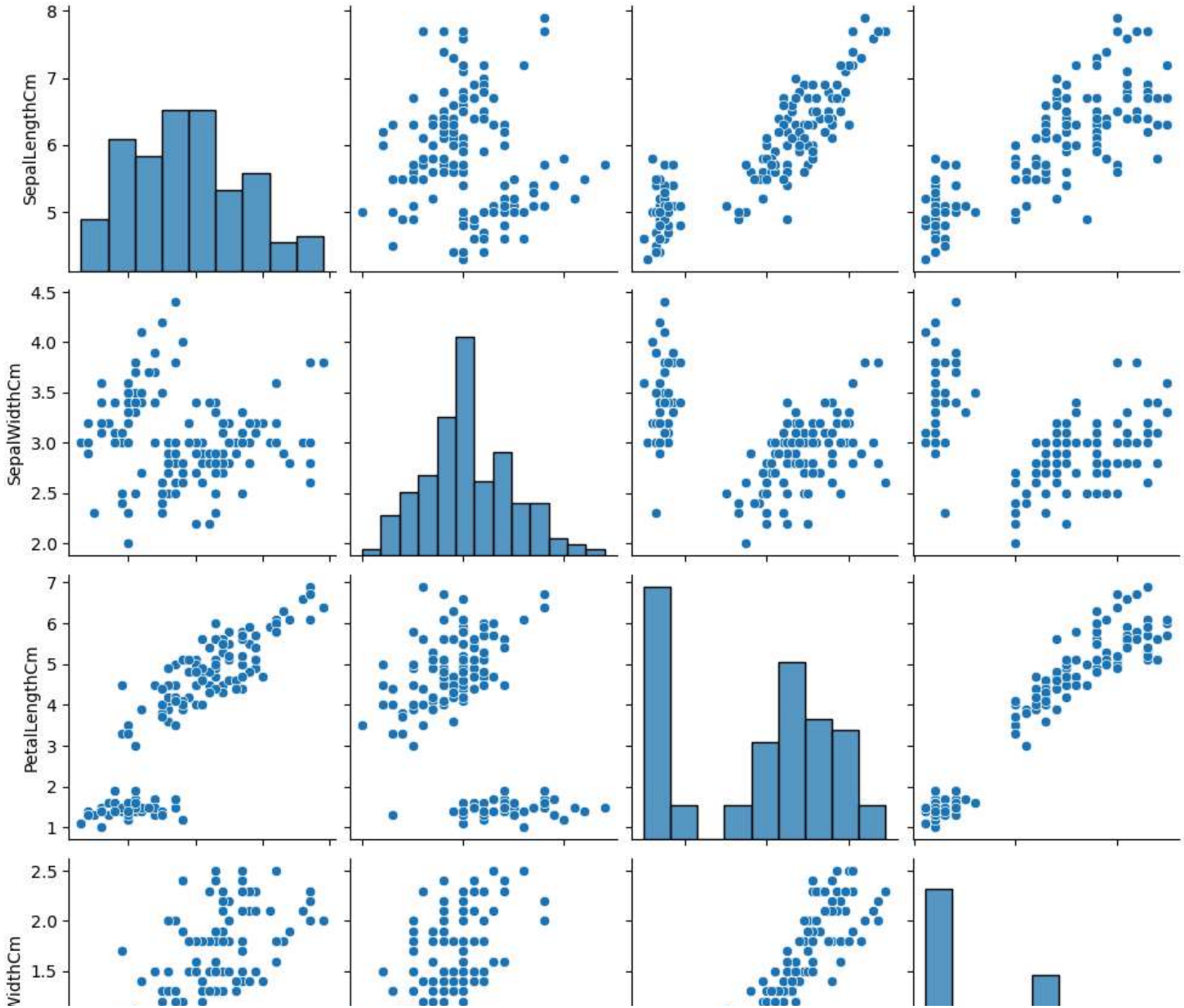


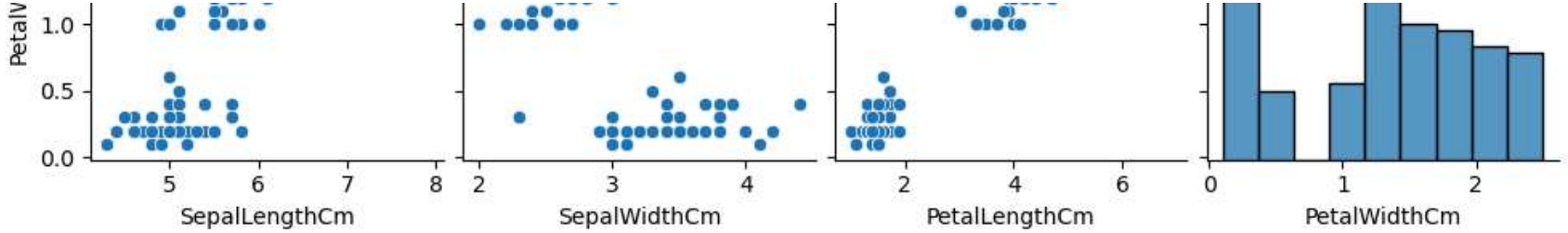






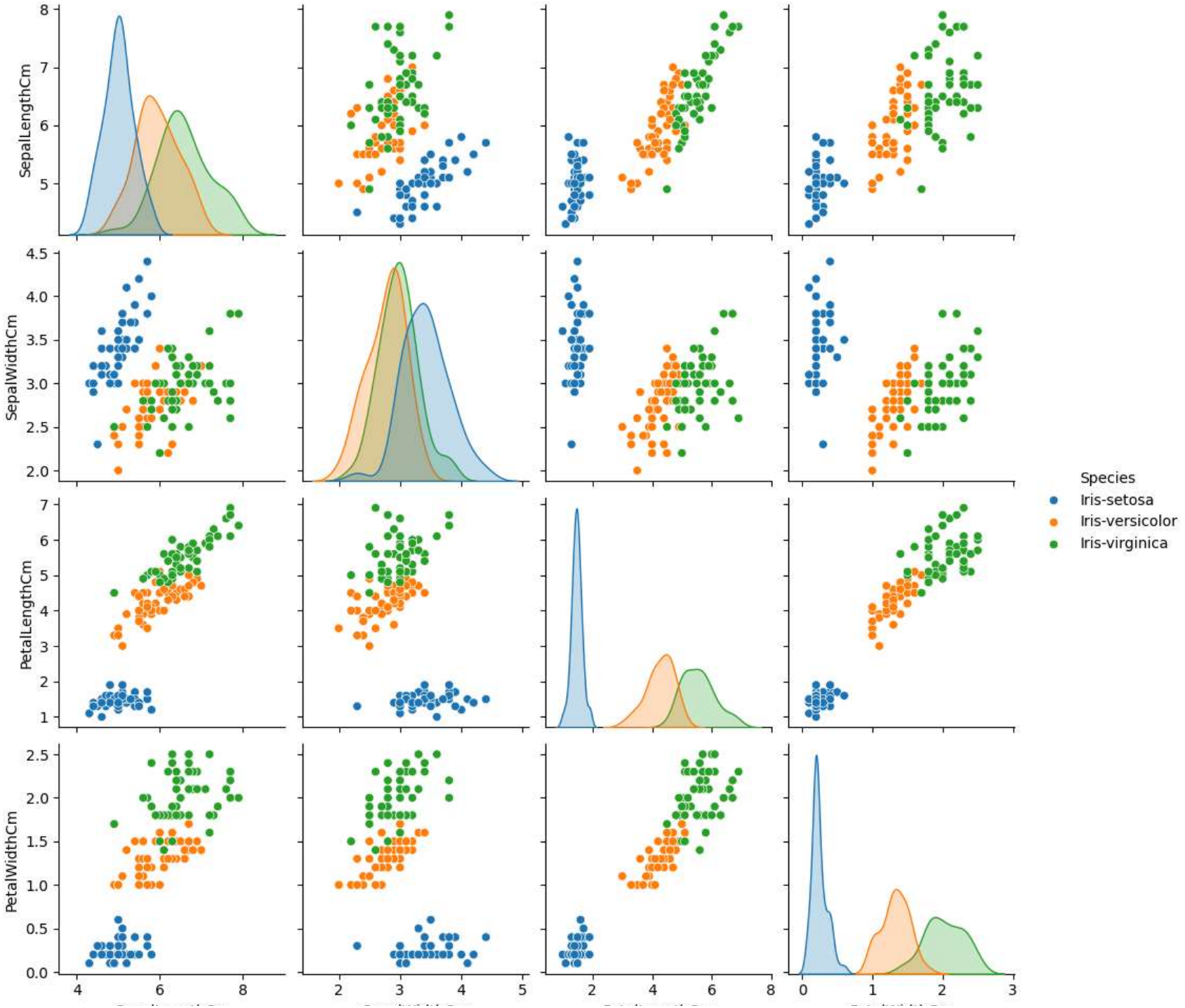
In [98]:
```
sns.pairplot(data=iris,kind='scatter')
plt.show()
```


SepalLengthCm
SepalWidthCm
PetalLengthCm
WidthCm



In [104…

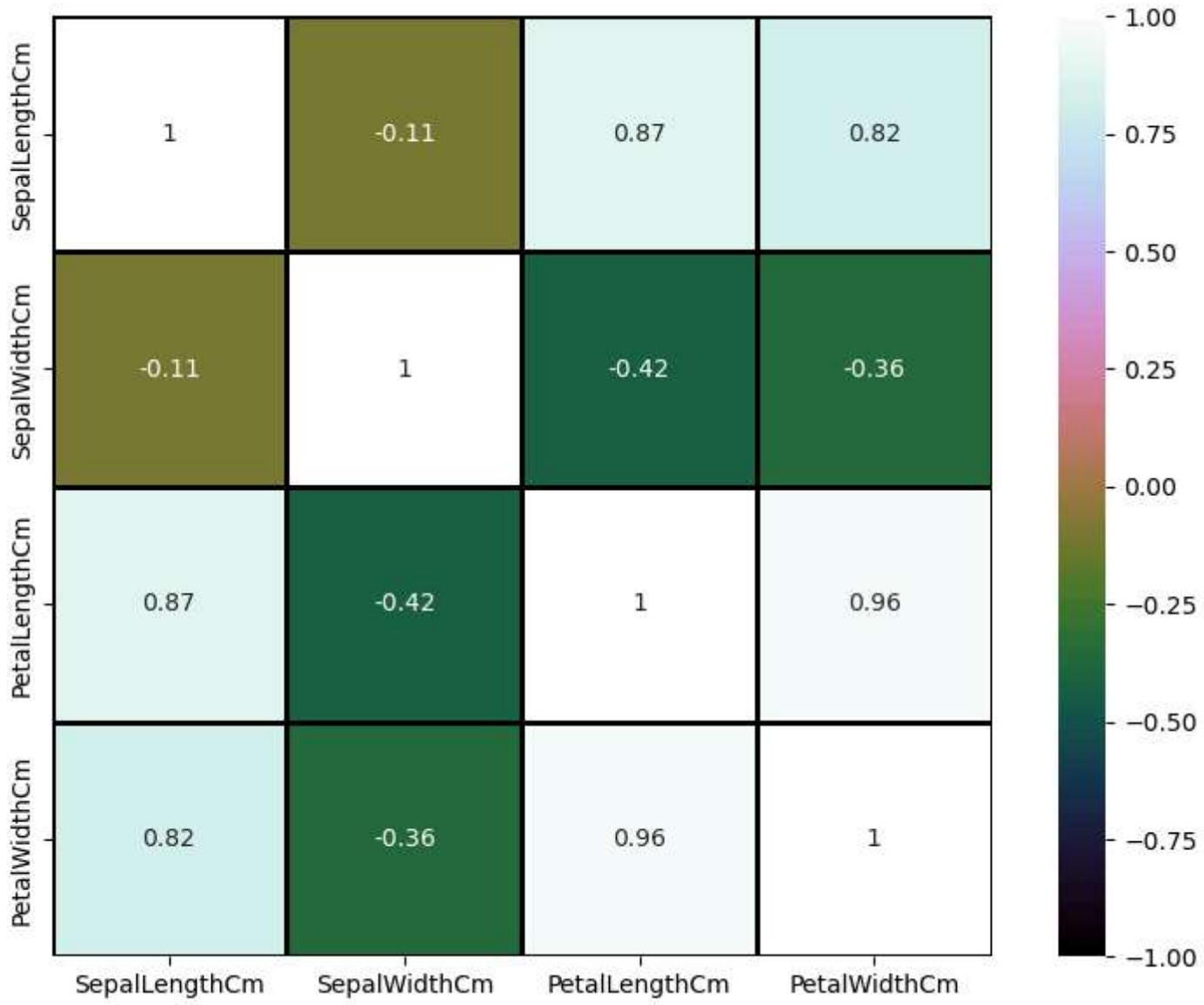
```
sns.pairplot(iris,hue='Species')
plt.show()
```


SepalLengthCm
SepalWidthCm
PetalLengthCm
PetalWidthCm
Species
Iris-setosa
Iris-versicolor
Iris-virginica

SepalLengthCm SepalWidthCm PetalLengthCm PetalWidthCm

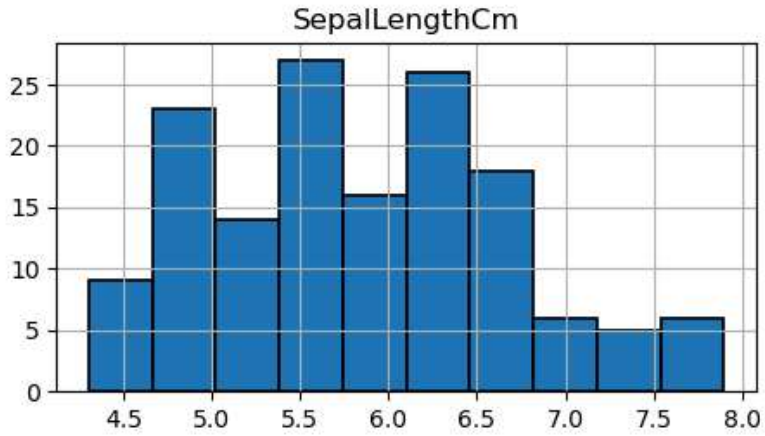
In [112…

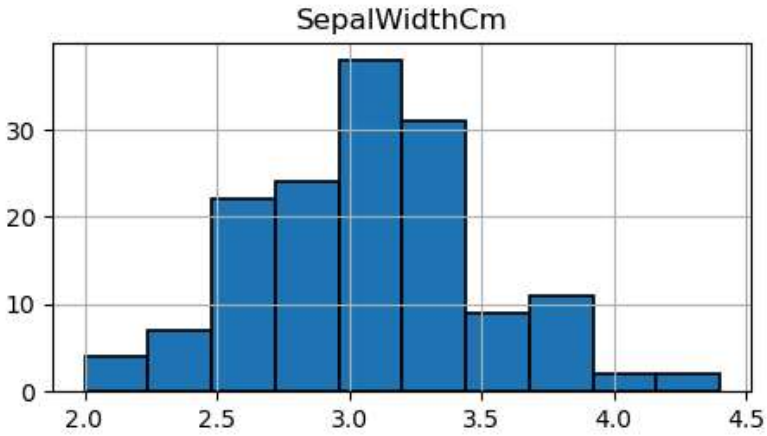
```
fig=plt.gcf()
fig.set_size_inches(10,7)
iris_numeric = iris.select_dtypes(include=['number'])
fig=sns.heatmap(iris_numeric.corr(),annot=True,cmap='cubehelix',linewidths=1,linecolor='k',square=True,mask=False, vr
plt.show()
```

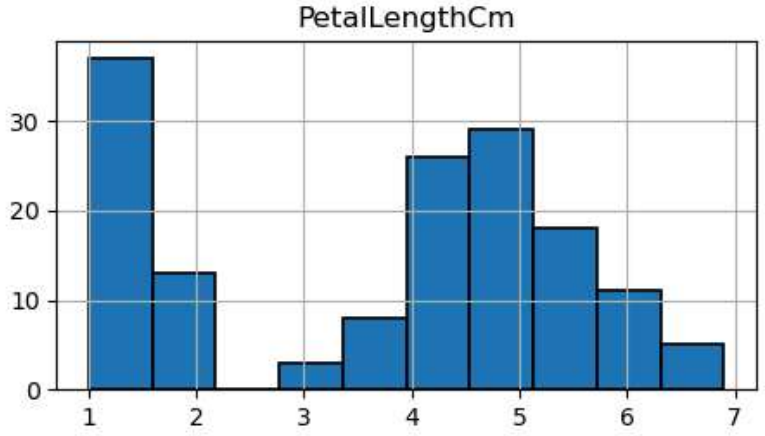


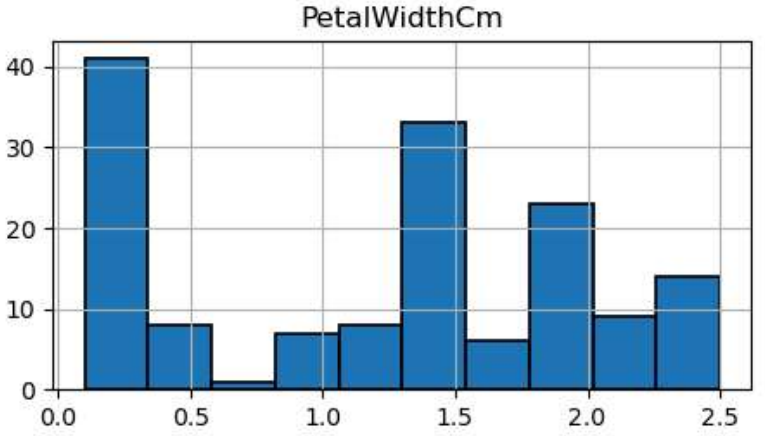
In [116…

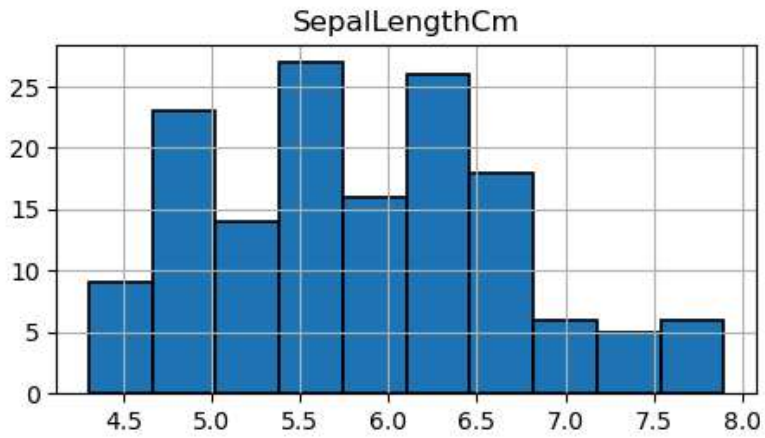
```
iris.hist(edgecolor='black', linewidth=1.2)
fig=plt.gcf()
fig.set_size_inches(12,6)
plt.show()
```

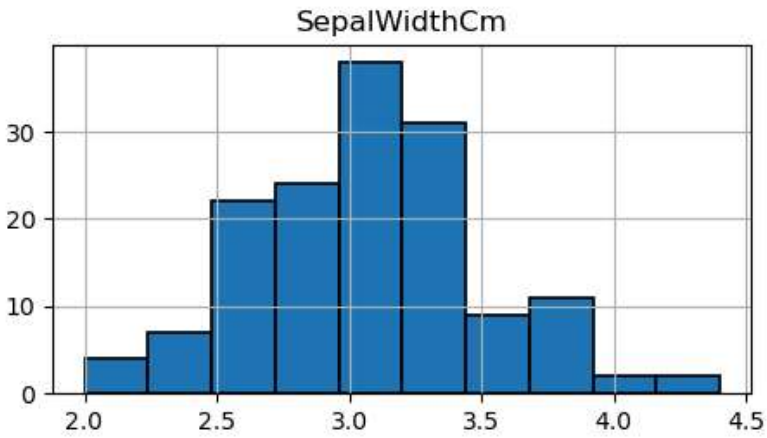

SepalLengthCm
25
20
15
10
5
0
4.5
5.0
5.5
6.0
6.5
7.0
7.5
8.0

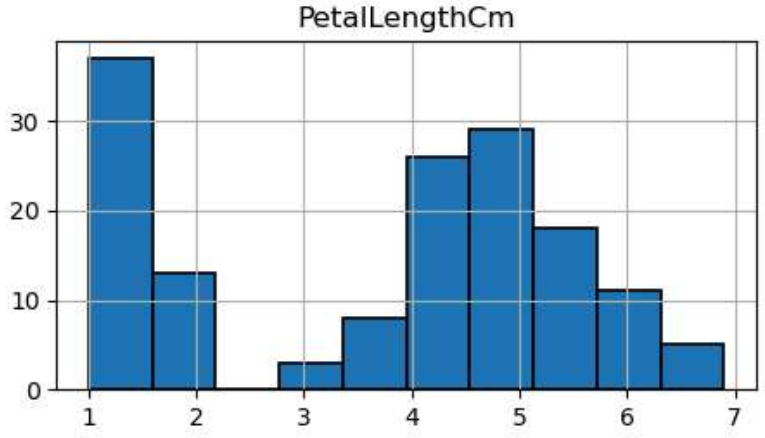

SepalWidthCm
30
20
10
0
2.0
2.5
3.0
3.5
4.0
4.5

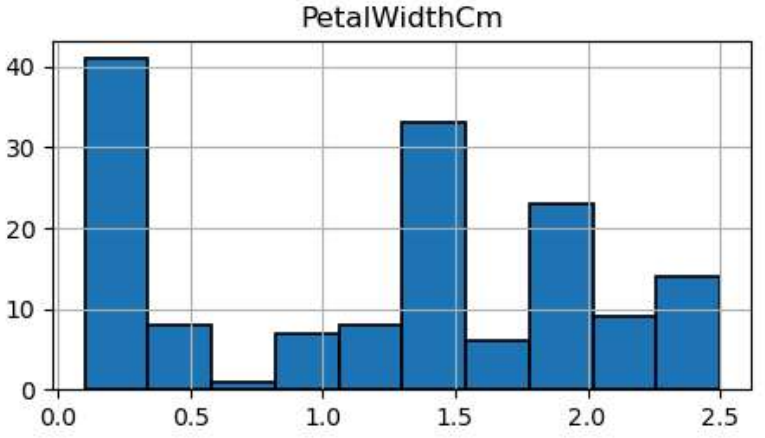

PetalLengthCm
30
20
10
0
1
2
3
4
5
6
7


PetalWidthCm
40
30
20
10
0
0.0
0.5
1.0
1.5
2.0
2.5



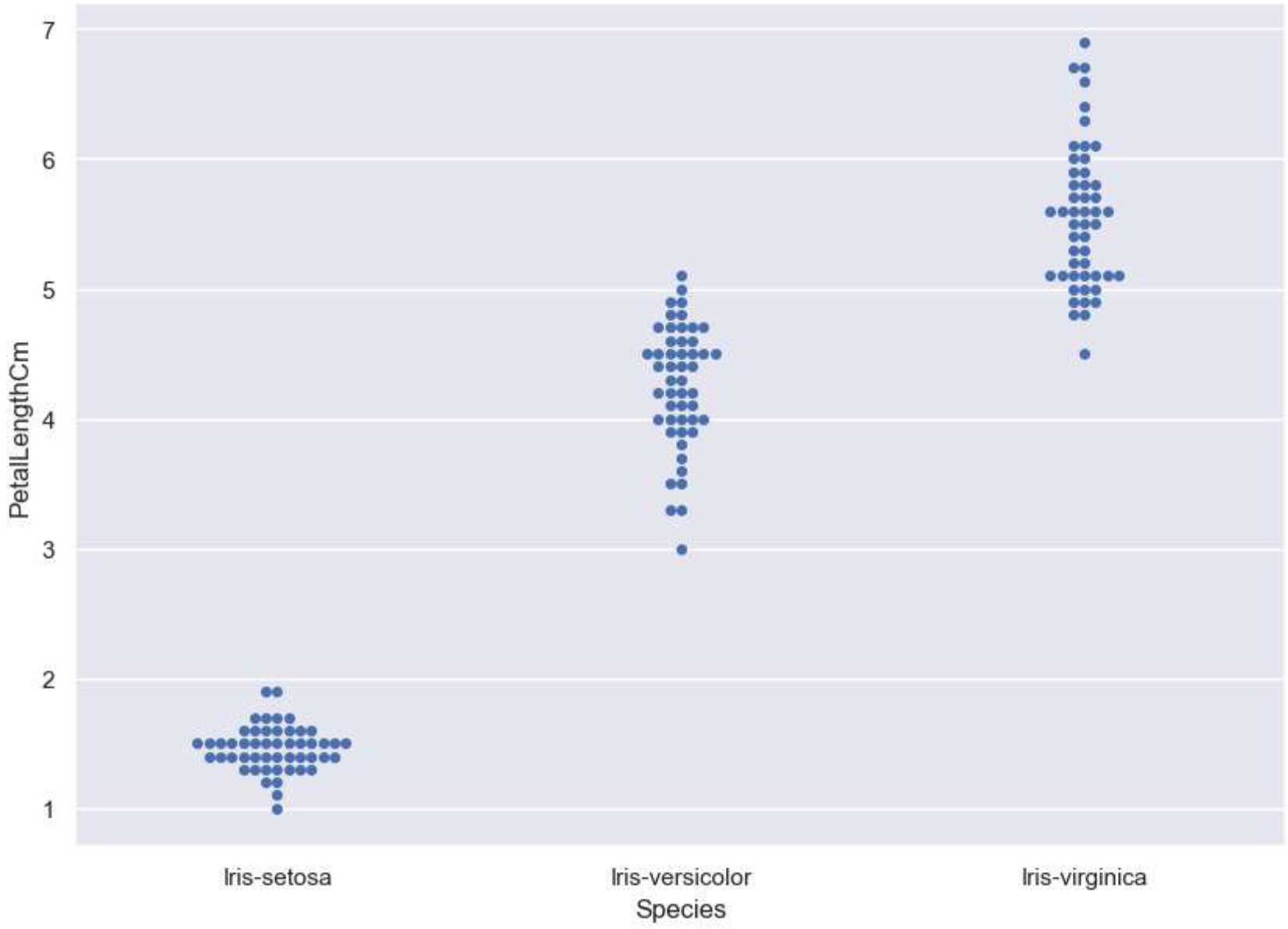






In [118…

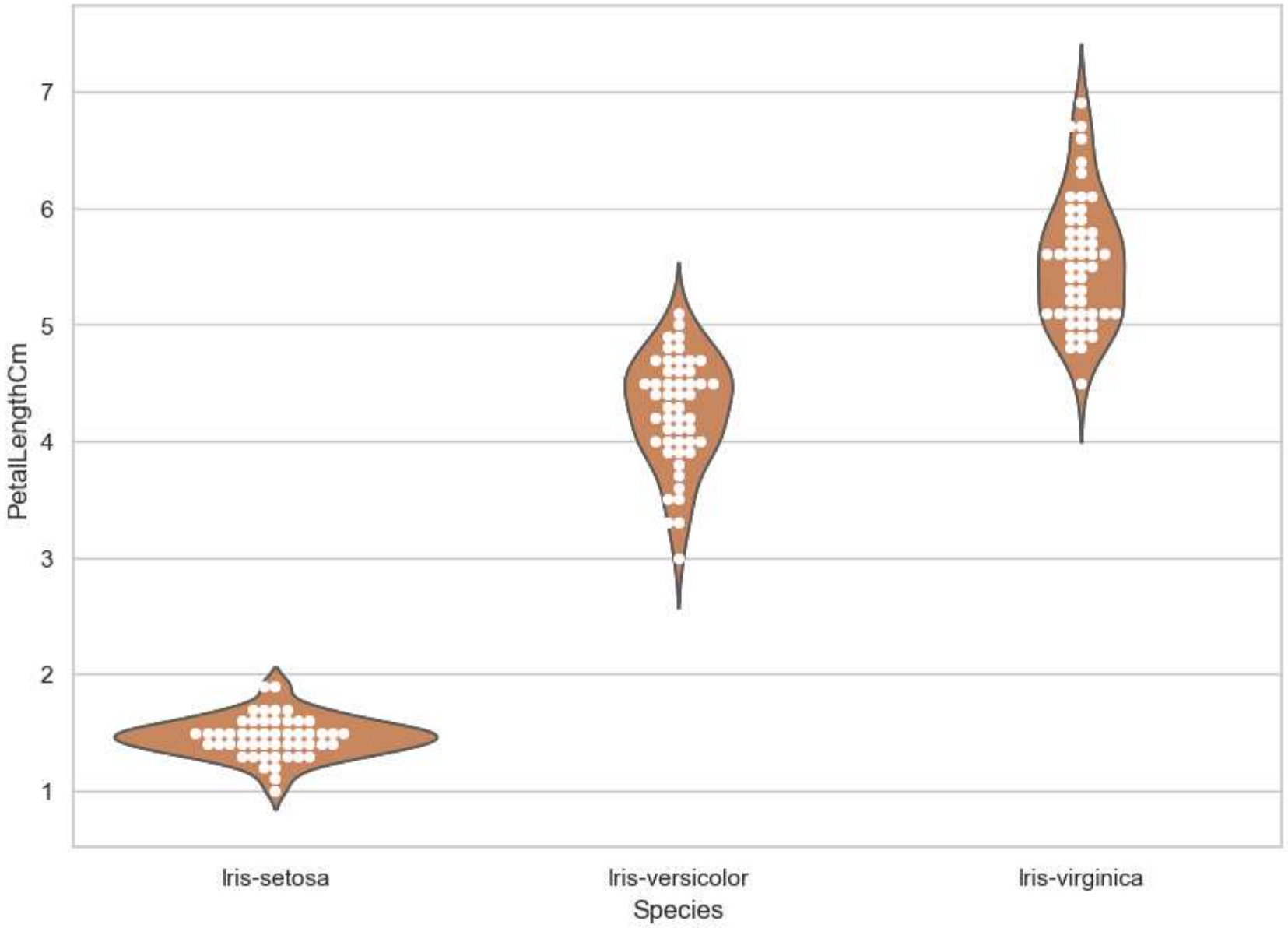
```
sns.set(style="darkgrid")
fig=plt.gcf()
fig.set_size_inches(10,7)
fig = sns.swarmplot(x="Species", y="PetalLengthCm", data=iris)
plt.show()
```



In [122…

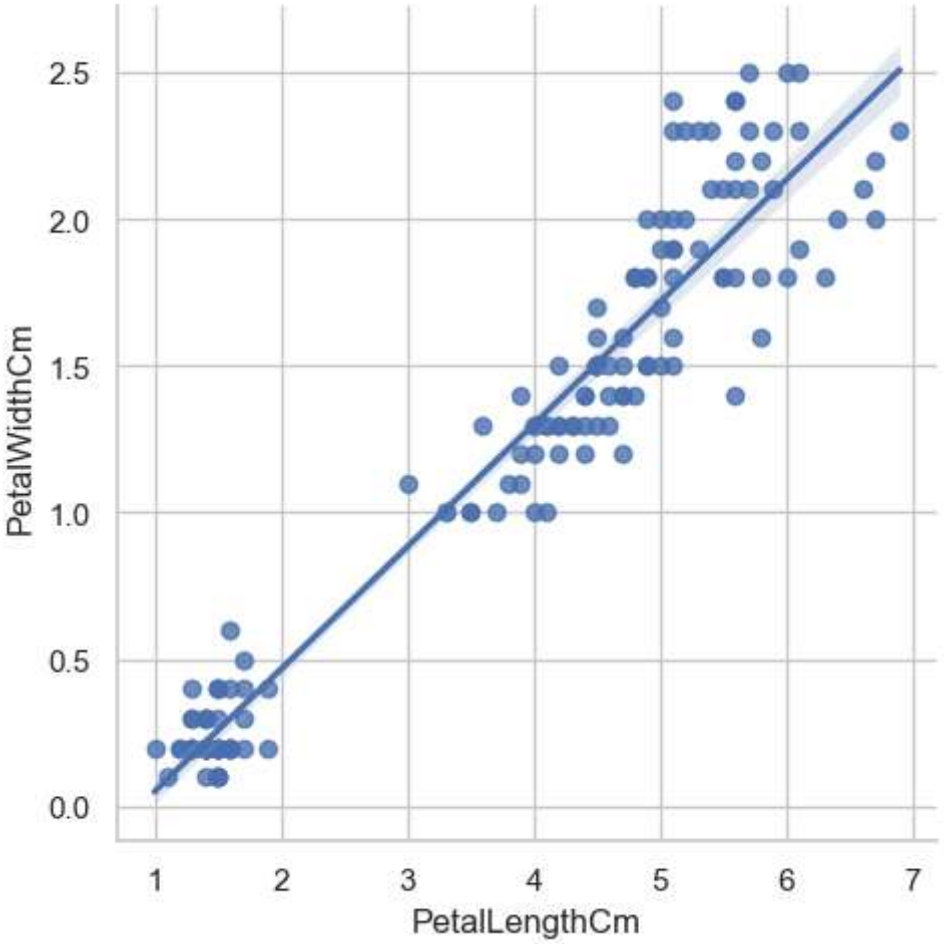
```
sns.set(style="whitegrid")
fig=plt.gcf()
fig.set_size_inches(10,7)
ax = sns.violinplot(x="Species", y="PetalLengthCm", data=iris, inner=None)
```

```
ax = sns.swarmplot(x="Species", y="PetalLengthCm", data=iris,color="white", edgecolor="black")
plt.show()
```



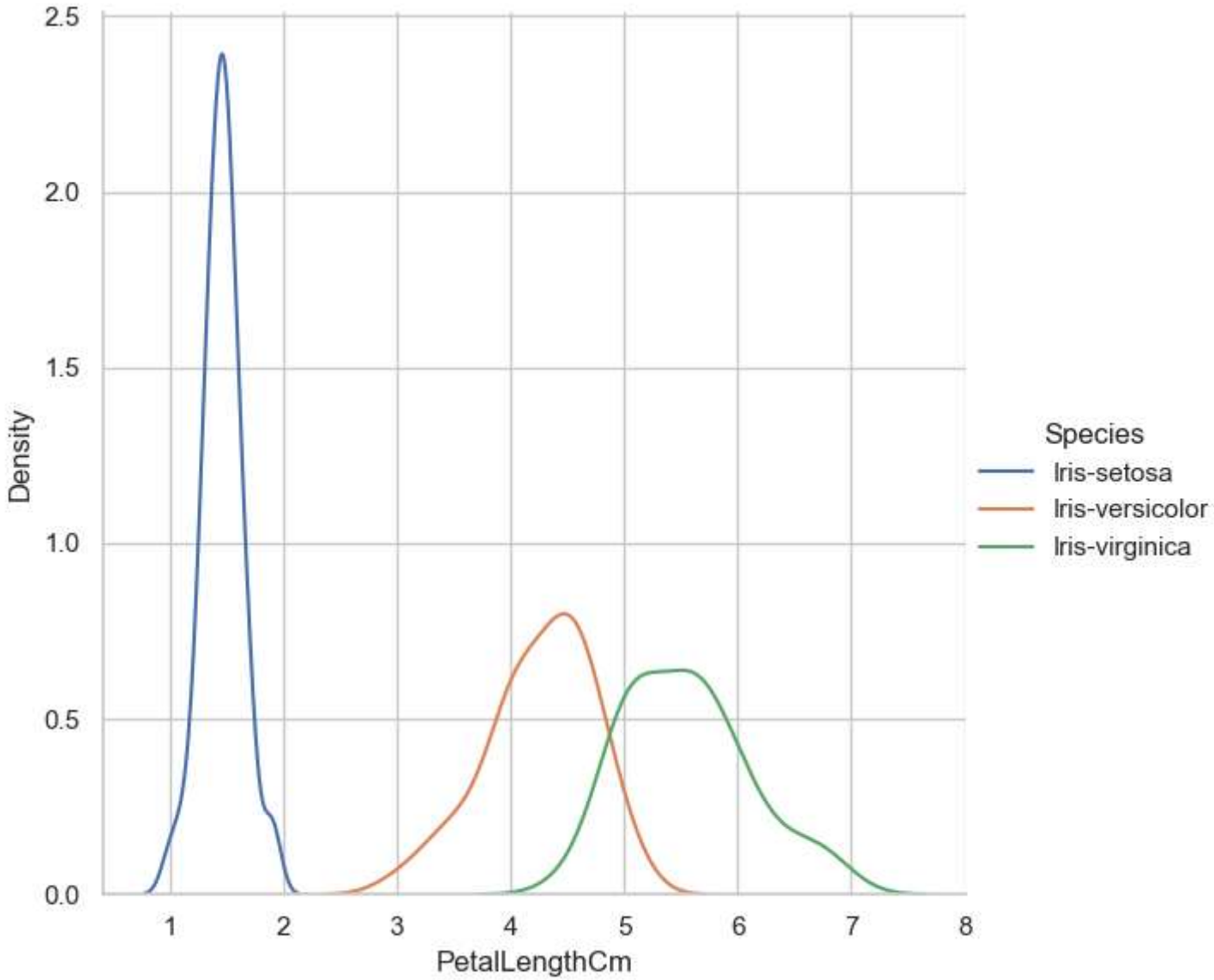
In [124…

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



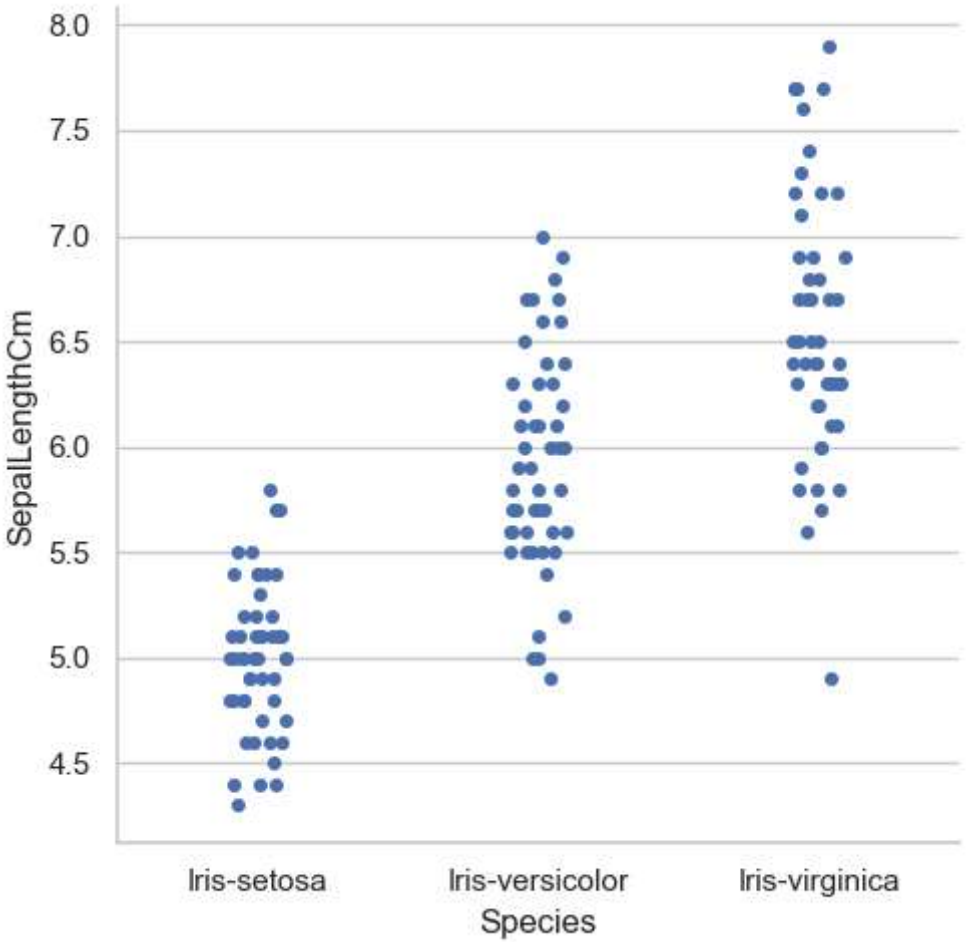
In [140…

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



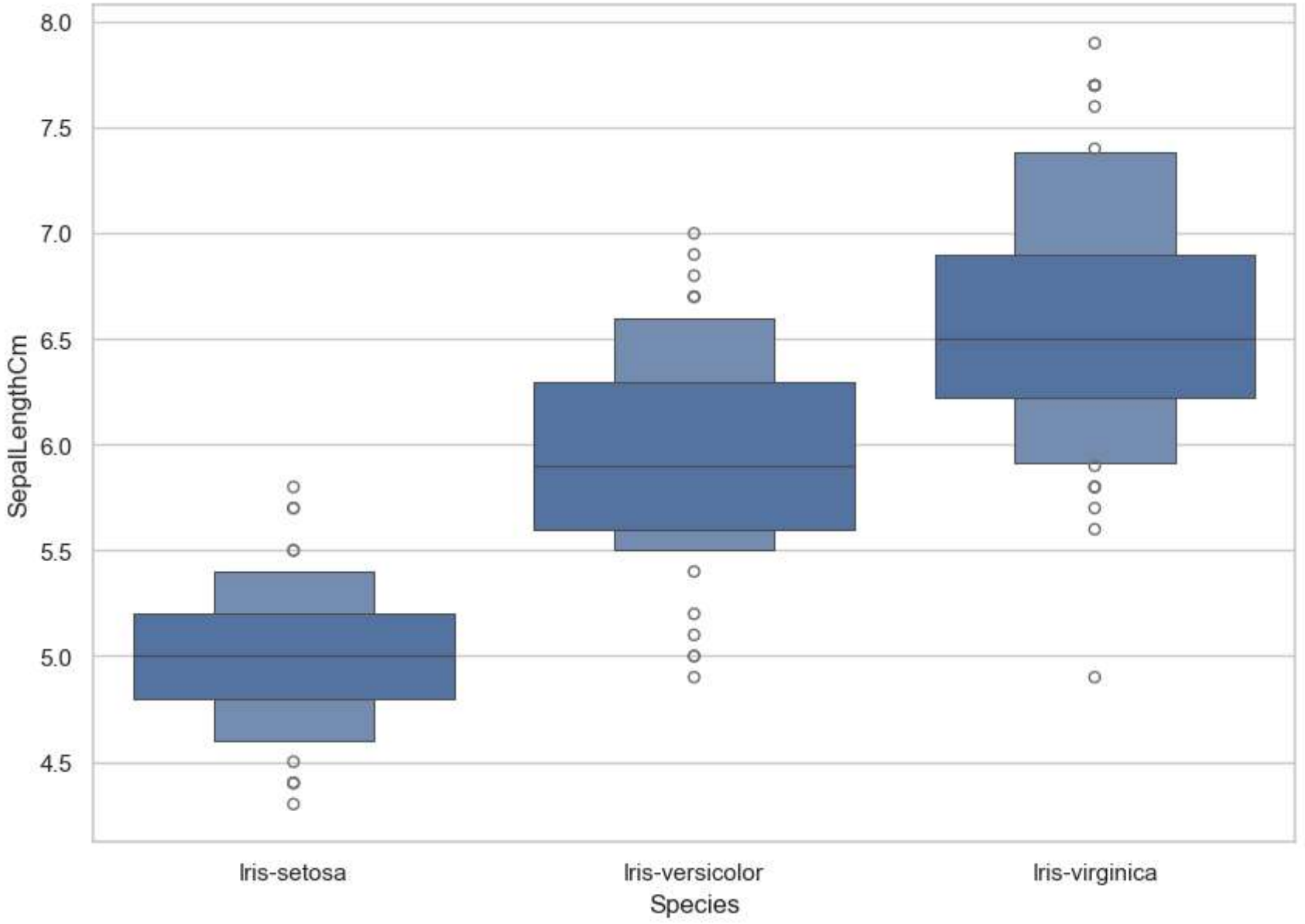
In [158…

```
sns.catplot(x='Species', y='SepalLengthCm', data=iris)
plt.ioff()
plt.show()
```



In [160…

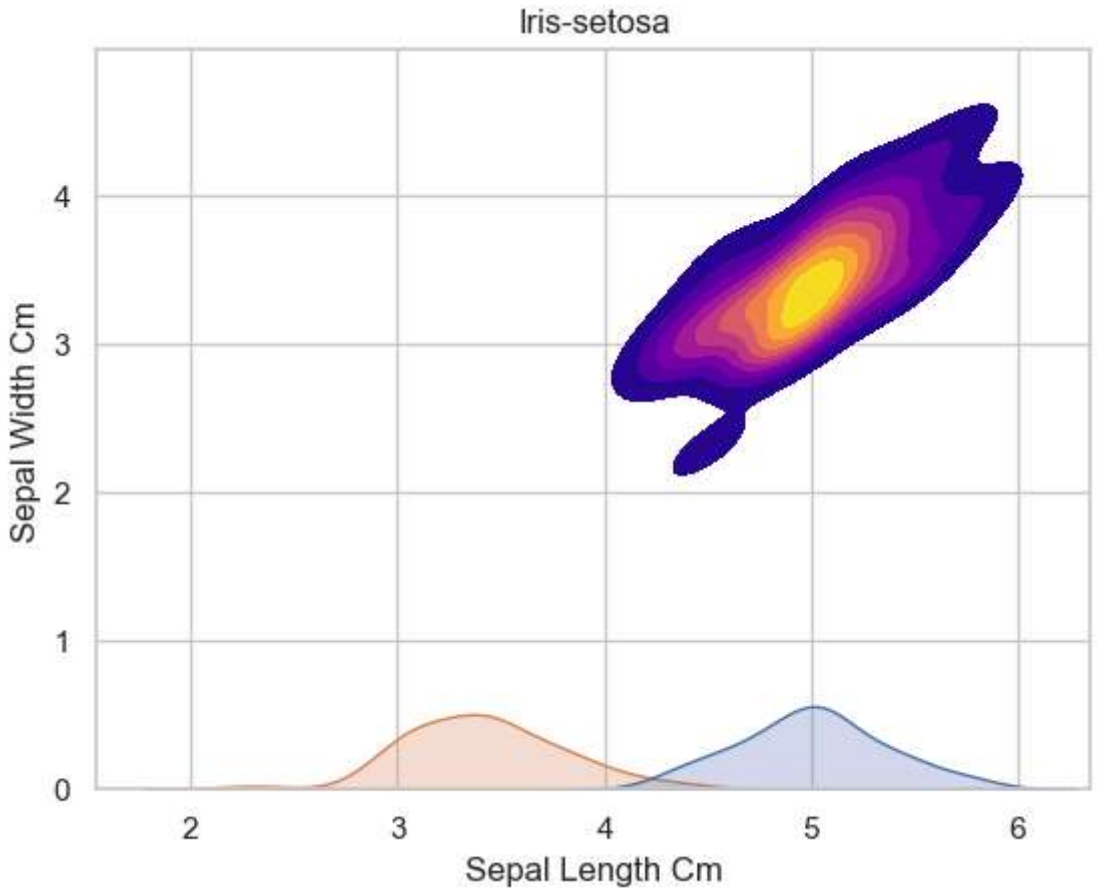
```
fig=plt.gcf()
fig.set_size_inches(10,7)
fig=sns.boxenplot(x='Species',y='SepalLengthCm',data=iris)
plt.show()
```



In [164…

```
sub = iris[iris['Species'] == 'Iris-setosa']

sns.kdeplot(
    x=sub['SepalLengthCm'],
    y=sub['SepalWidthCm'],
```

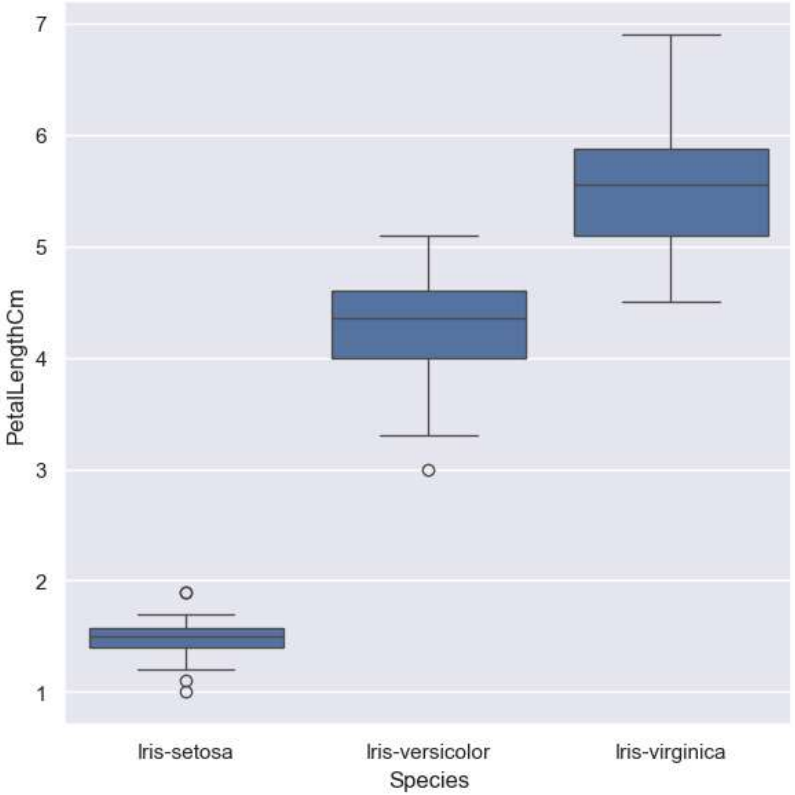
```
    cmap="plasma",
    fill=True  # Instead of 'shade'
)

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

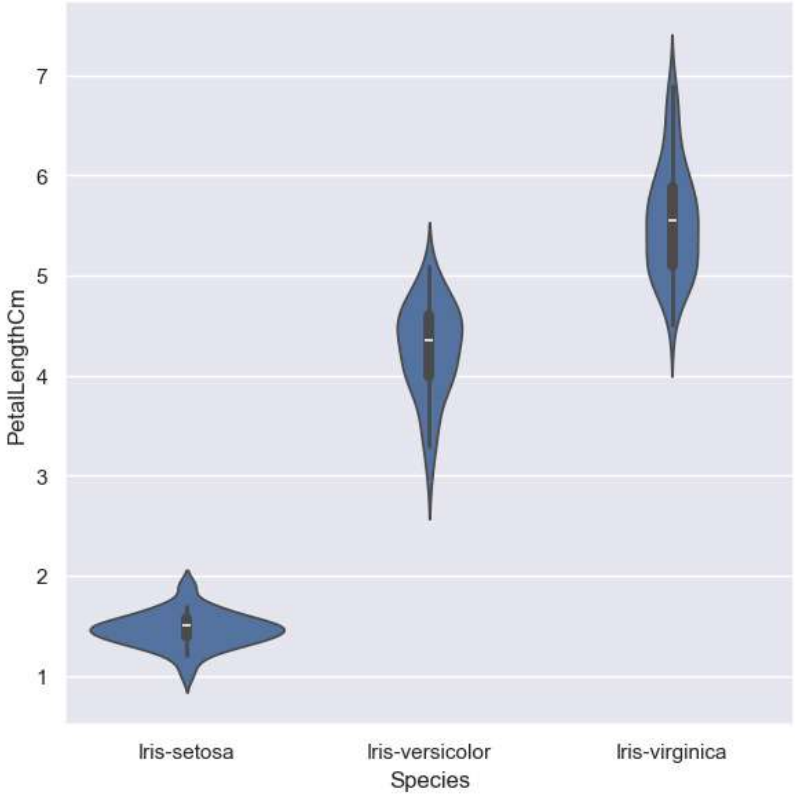


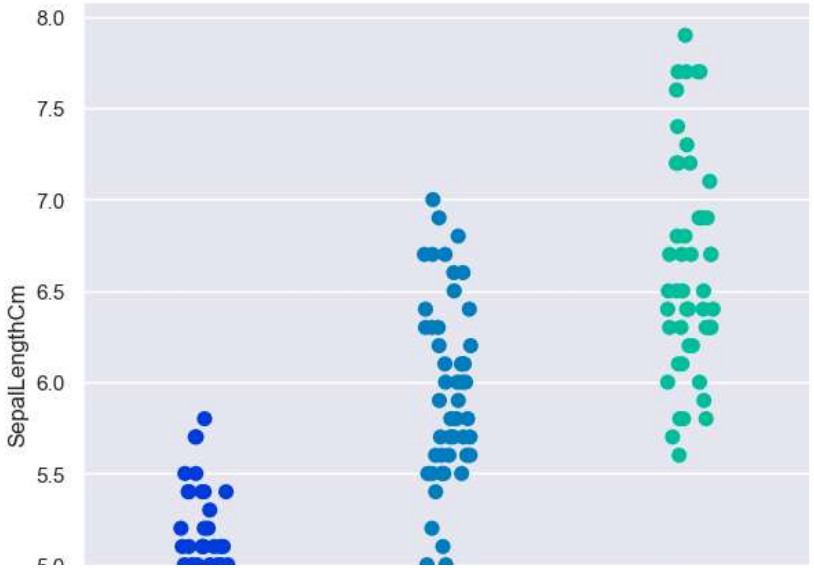
In [166…

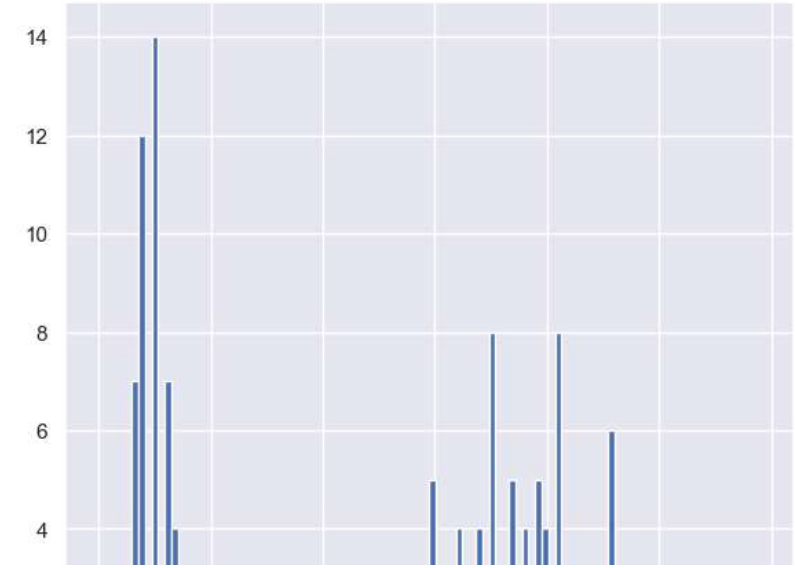
```
sns.set_style('darkgrid')
f,axes=plt.subplots(2,2,figsize=(15,15))

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])
```

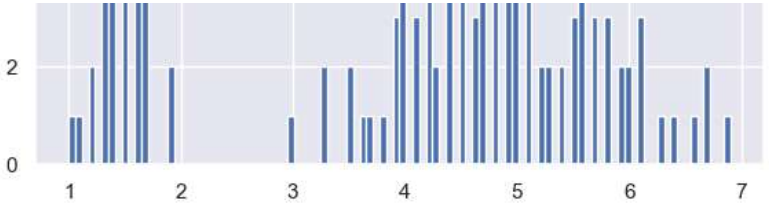
```
k3=sns.stripplot(x='Species',y='SepalLengthCm',data=iris,jitter=True,edgecolor='gray',size=8,palette='winter',orient=
#axes[1,1].hist(iris.hist,bin=10)
axes[1,1].hist(iris.PetalLengthCm,bins=100)
#k2.set(xlim=(-1,0.8))
plt.show()
```


PetalLengthCm
7
6
5
4
3
2
1
Iris-setosa
Iris-versicolor
Iris-virginica
Species


PetalLengthCm
7
6
5
4
3
2
1
Iris-setosa
Iris-versicolor
Iris-virginica
Species

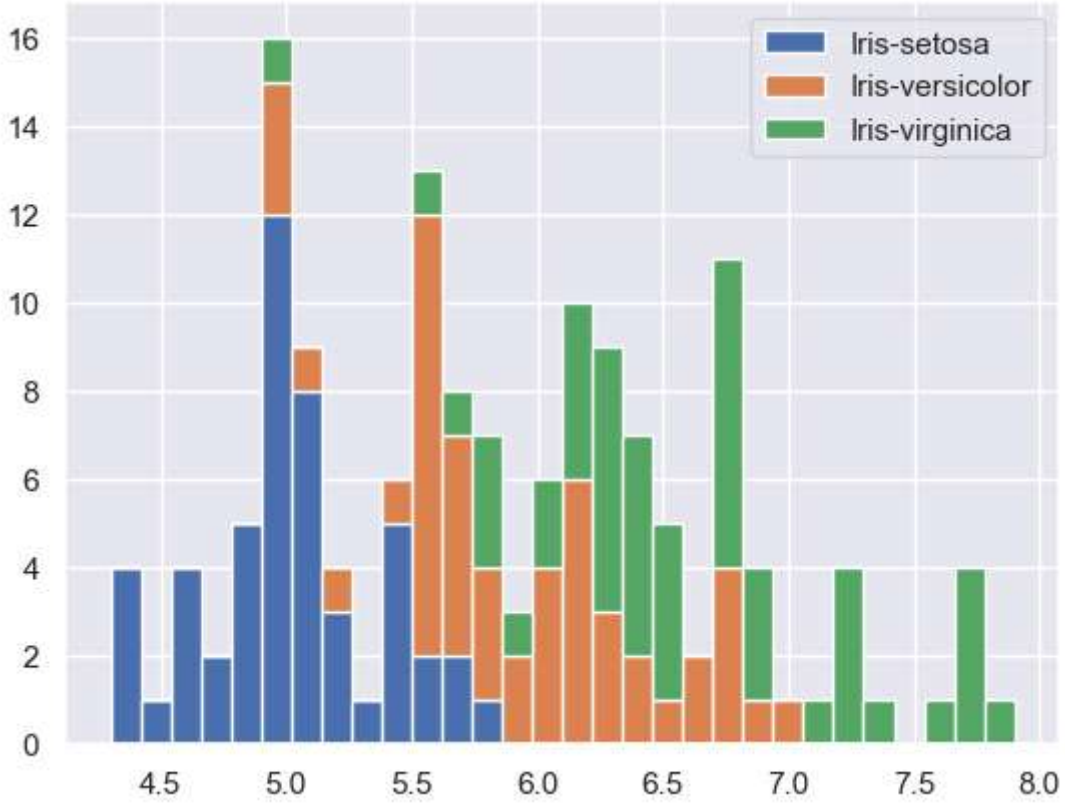

SepalLengthCm
8.0
7.5
7.0
6.5
6.0
5.5


14
12
10
8
6
4



In [168…
```
iris['Species'] = iris['Species'].astype('category')
#iris.head()
```

In [170…
```
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()
```



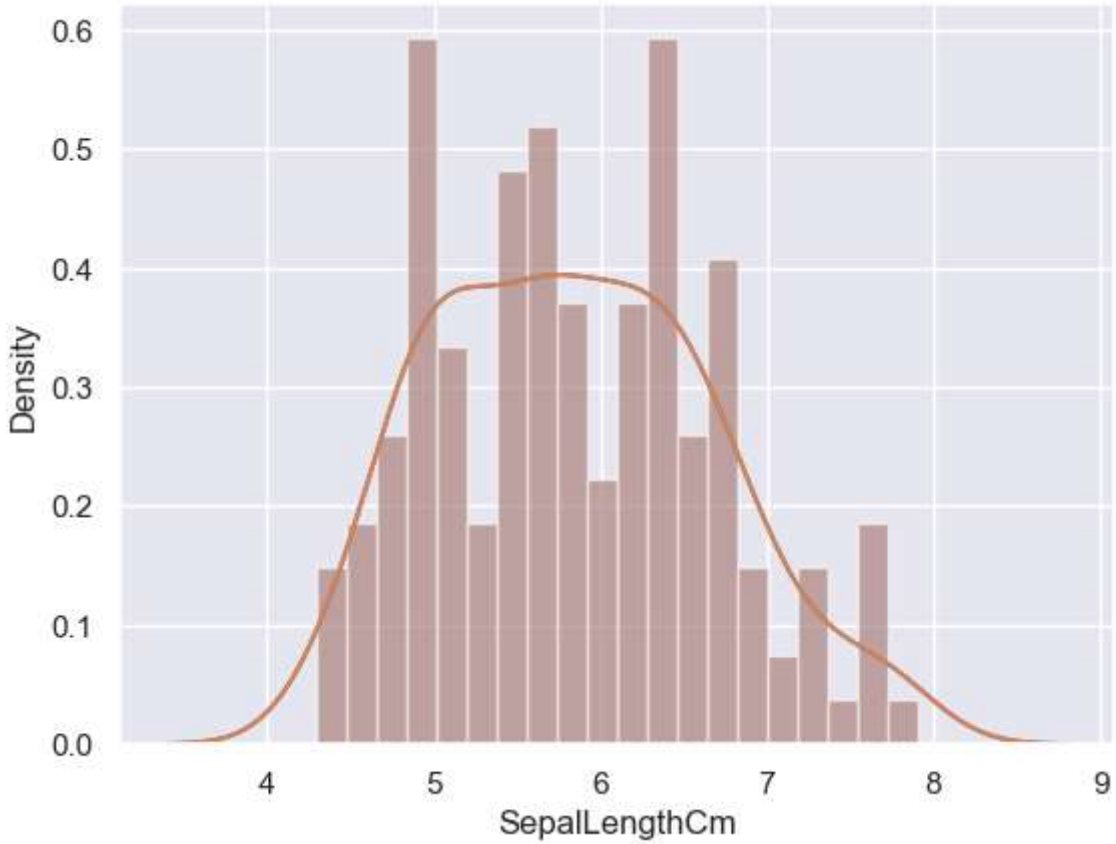
In [174…

```
iris.plot.area(y=['SepalLengthCm','SepalWidthCm','PetalLengthCm','PetalWidthCm'],alpha=0.4,figsize=(12, 6));
plt.show()
```


SepalLengthCm
SepalWidthCm
PetalLengthCm
PetalWidthCm
20.0
17.5
15.0
12.5
10.0
7.5
5.0
2.5
0.0
0
20
40
60
80
100
120
140

In [178…

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


0.6
0.5
0.4
0.3
0.2
0.1
0.0
Density
4
5
6
7
8
9
SepalLengthCm