Hands-On

Hands-On ini digunakan pada kegiatan Microcredential Associate Data Scientist 2021

Pertemuan 6

Pertemuan 6 (enam) pada Microcredential Associate Data Scientist 2021 menyampaikan materi mengenai Mengumpulkan Data, Menelaah Data dengan metode Visualisasi

Latihan (1)

Sebelum menelaah data dengan metode visualisasi, kita perlu memanggil modul visualisasi (seaborn & matplotlib) terlebih dahulu.

In [56]:

```python
# memanggil modul Pandas and Seaborn
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
plt.style.use('fivethirtyeight')

import warnings
warnings.filterwarnings('ignore')
```

In [57]:

```python
bunga ='Iris.csv'
df = pd.read_csv(bunga)
```

In [58]:

```python
# menampilkan 5 baris data
df.head()
```

Out[58]:

	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

Latihan (2)

Karena kita tidak membutuhkan kolom "Id" dalam melakukan visualisasi kita dapat menghapus kolom "Id" menggunakan fungsi .drop()

In [59]:

```python
# menghapus kolom "id"
df.drop('Id', axis=1, inplace=True)
```

Latihan (3)

Lakukan pengecekan nilai yang hilang (*missing value*) pada dataset. Dengan function info()

In [60]:

```python
# memeriksa missing values pada dataset
df.isna().sum()
```

Out[60]:

```
SepalLengthCm    0
SepalWidthCm     0
PetalLengthCm    0
PetalWidthCm     0
Species          0
dtype: int64
```

Latihan (4)

Tampilkan visualisasi dari data yang telah menggunakan fungsi describe() untuk mendapatkan informasi umum statistik tentang dataset

In [61]:

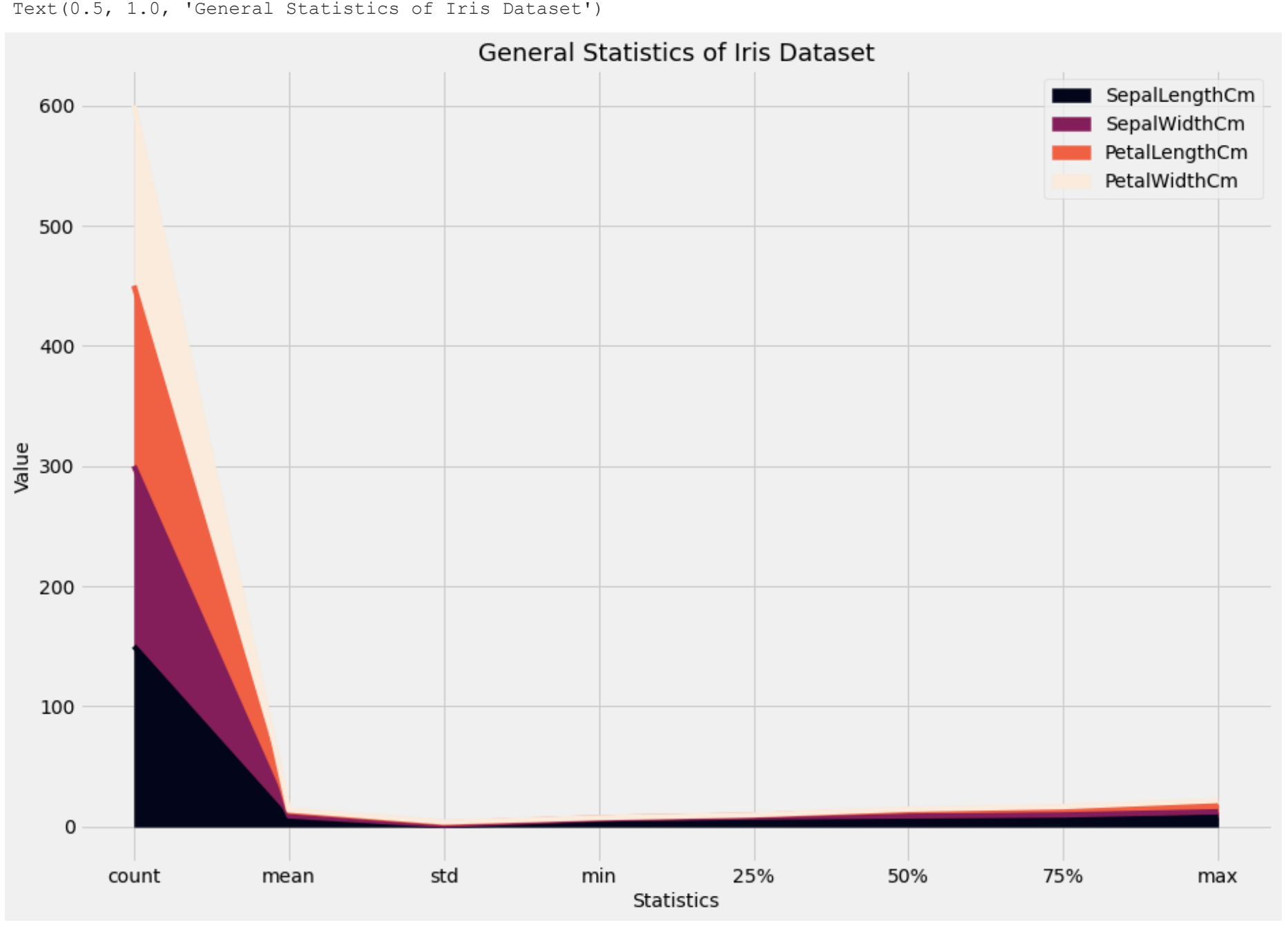
```python
# melakukan visualisasi dari data describe
```

In [62]:

```python
df.describe().plot(kind="area", figsize=(14, 10), colormap="rocket")
plt.xlabel('Statistics',fontsize=14)
plt.ylabel('Value',fontsize=14)
plt.title("General Statistics of Iris Dataset", fontsize=18)
```

Out[62]:

```
Text(0.5, 1.0, 'General Statistics of Iris Dataset')
```



Latihan (5)

Tampilkan visualisasi bar plot dan pie chart untuk menghitung frekuensi setiap species dalam dataset iris

In [63]:

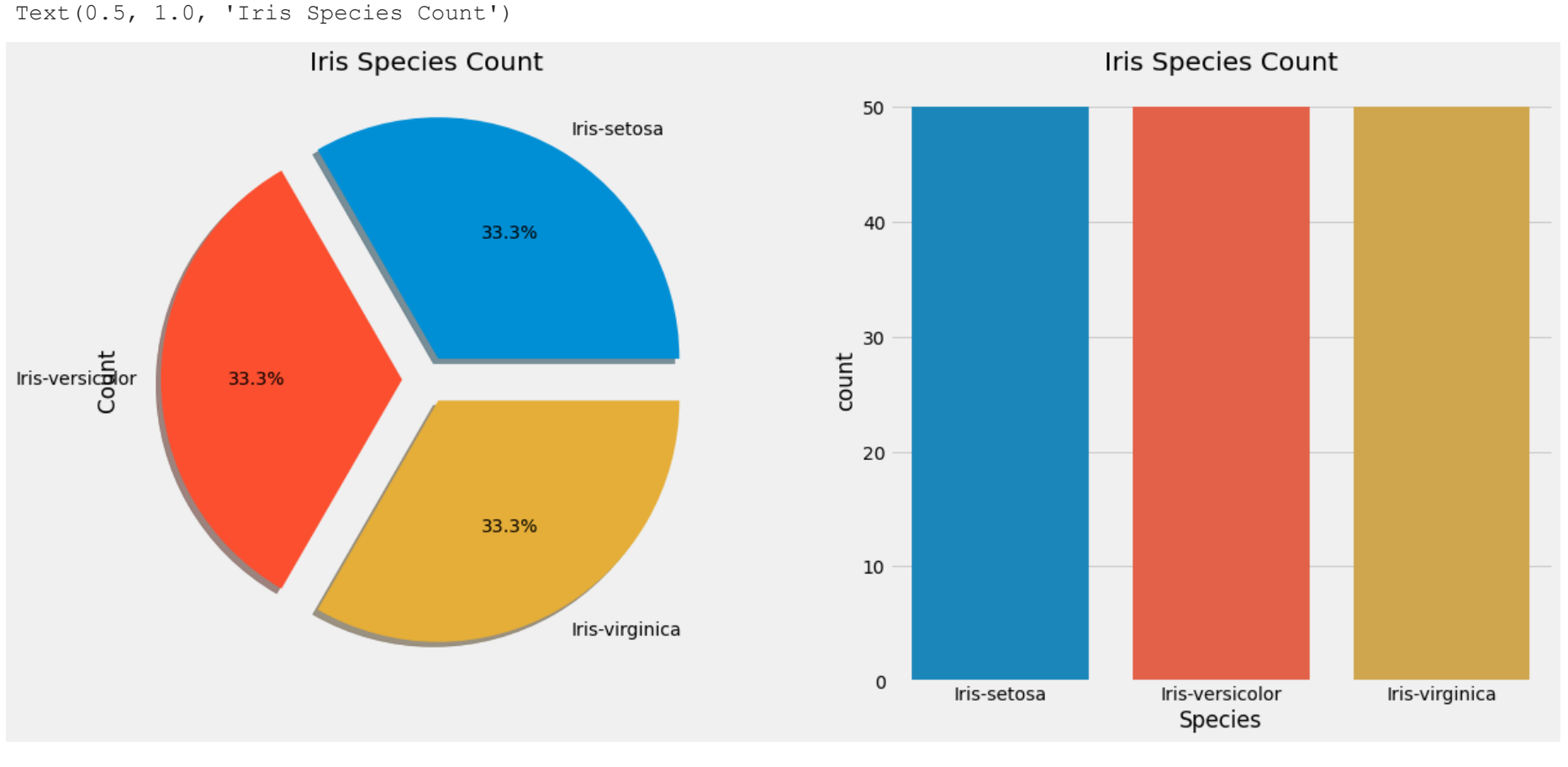
```python
# visualisasi bar plot dan pie chart
```

In [64]:

```python
f,ax=plt.subplots(1,2,figsize=(18,8))
df['Species'].value_counts().plot.pie(explode=[0.1,0.1,0.1],autopct='%1.1f%%',ax=ax[0],shadow=True)
ax[0].set_title('Iris Species Count')
ax[0].set_ylabel('Count')
sns.countplot('Species',data=df,ax=ax[1])
ax[1].set_title('Iris Species Count')
```

Out[64]:

```
Text(0.5, 1.0, 'Iris Species Count')
```



Visualisasi jointplot digunakan untuk menganalisis dua variable dan menggambarkan distribusi pada plot

Tampilkan visualisasi jointplot menggunakan fitur 'SepalLengthCm' dan 'SepalWidthCm' dalam visualisasi jointplot.

Latihan (6)

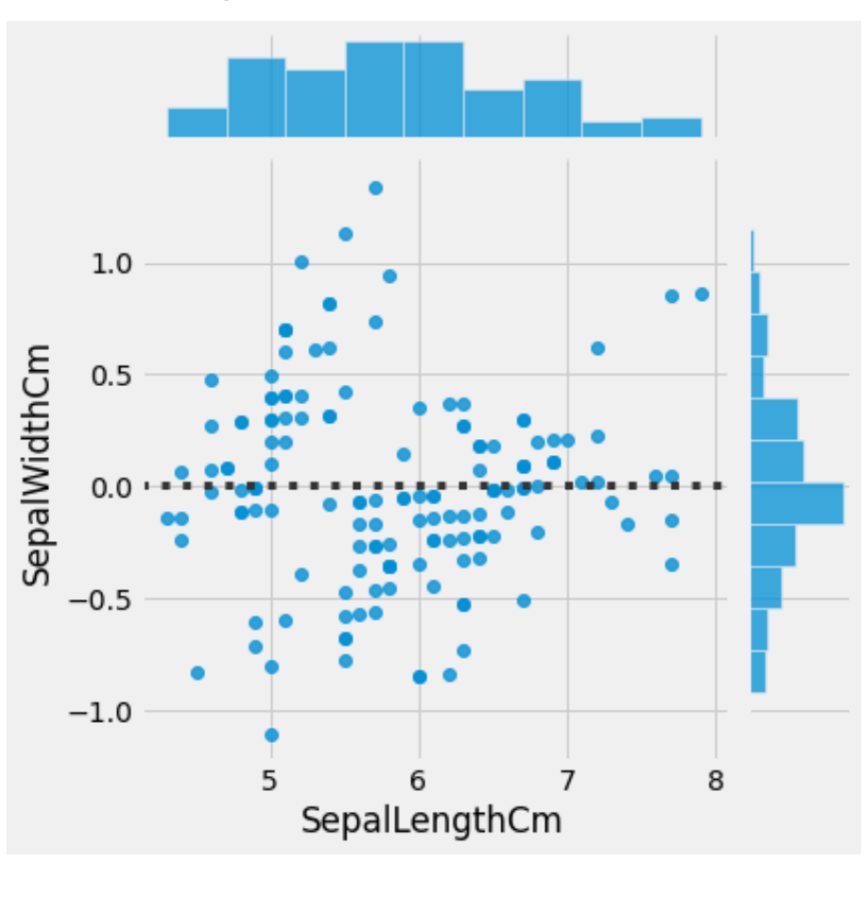
Gunakan jenis plot residual

In [65]:

```python
sns.jointplot(x='SepalLengthCm', y='SepalWidthCm', kind='resid', data=df)
```

Out[65]:

```
<seaborn.axisgrid.JointGrid at 0x2607a561be0>
```



Latihan (7)

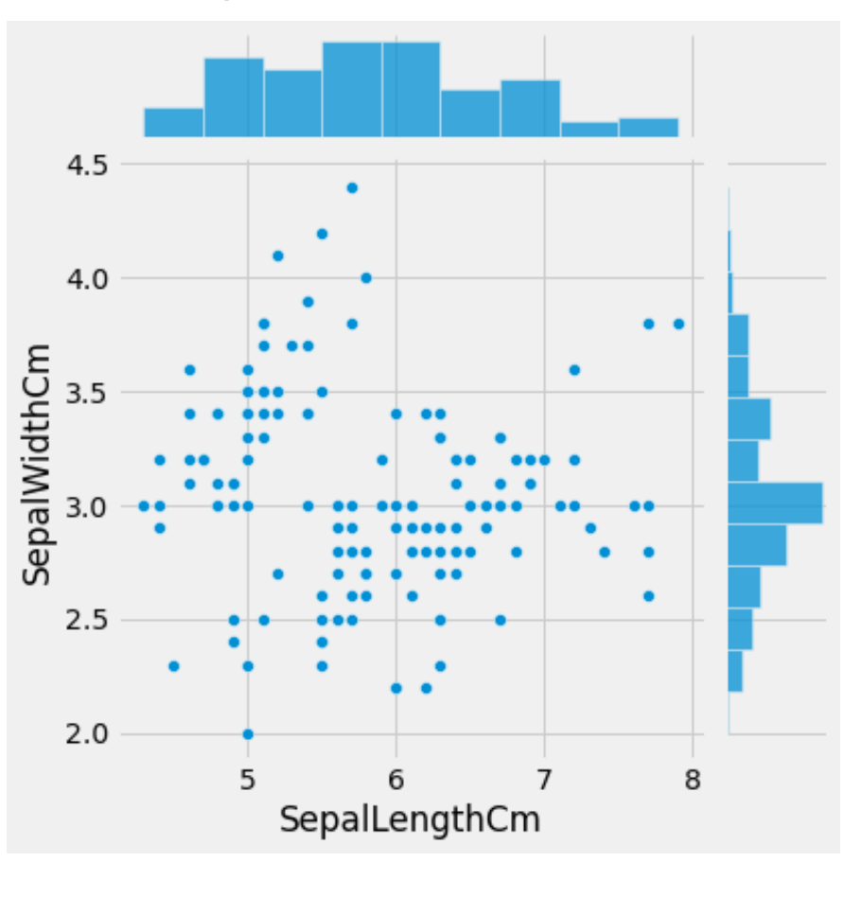
Gunakan jenis scatter plot

In [66]:

```python
sns.jointplot(x='SepalLengthCm',y='SepalWidthCm',kind="scatter", data=df)
```

Out[66]:

```
<seaborn.axisgrid.JointGrid at 0x2607b2ab910>
```



Latihan (8)

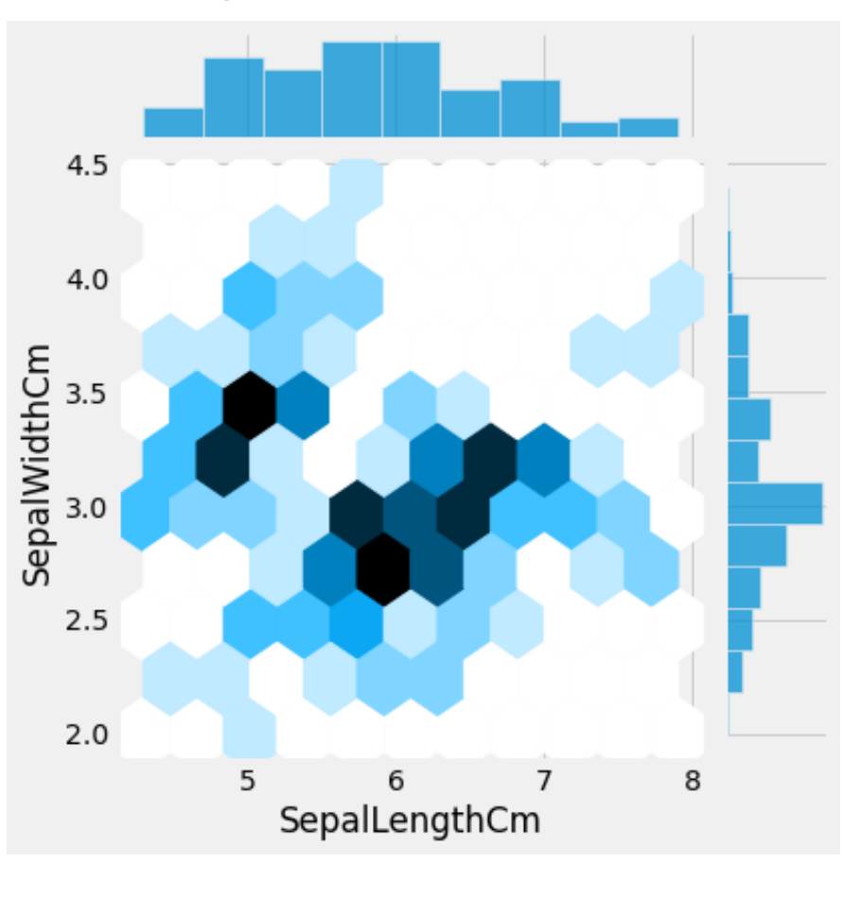
Gunakan jenis hexagons plot

In [67]:

```python
sns.jointplot('SepalLengthCm','SepalWidthCm',kind="hex",data=df)
```

Out[67]:

```
<seaborn.axisgrid.JointGrid at 0x2607b2ce910>
```



Latihan (9)

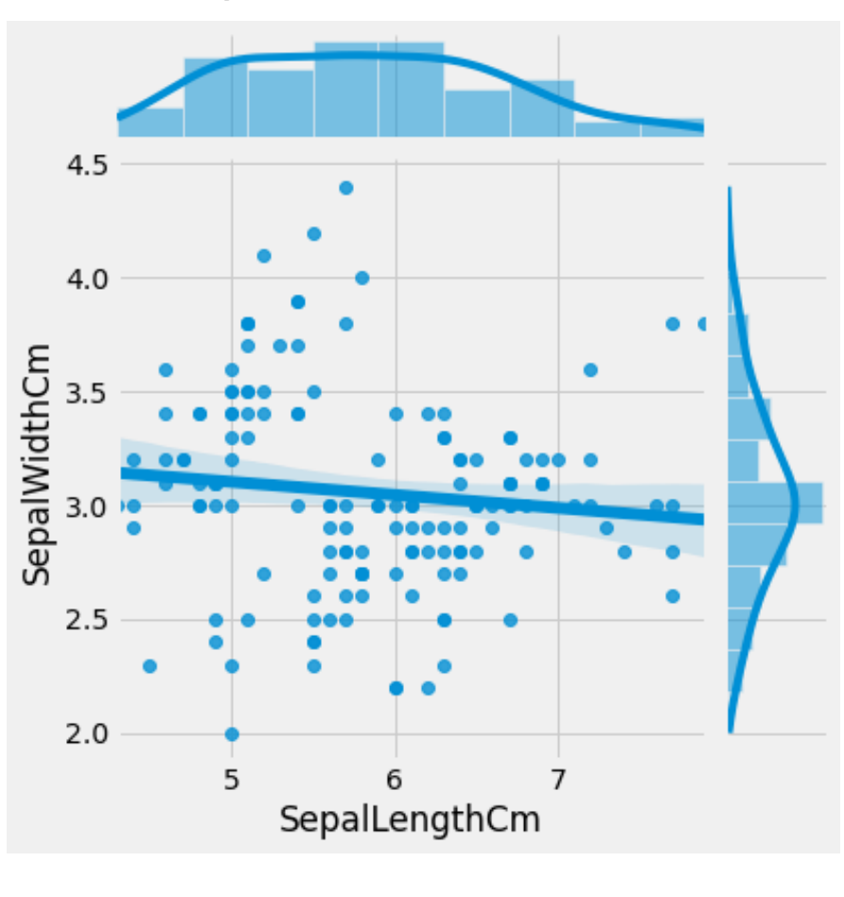
Gunakan jenis Linear regression line plot

In [68]:

```python
sns.jointplot('SepalLengthCm','SepalWidthCm', data=df, kind="reg")
```

Out[68]:

```
<seaborn.axisgrid.JointGrid at 0x2607a53a250>
```



Latihan (10)

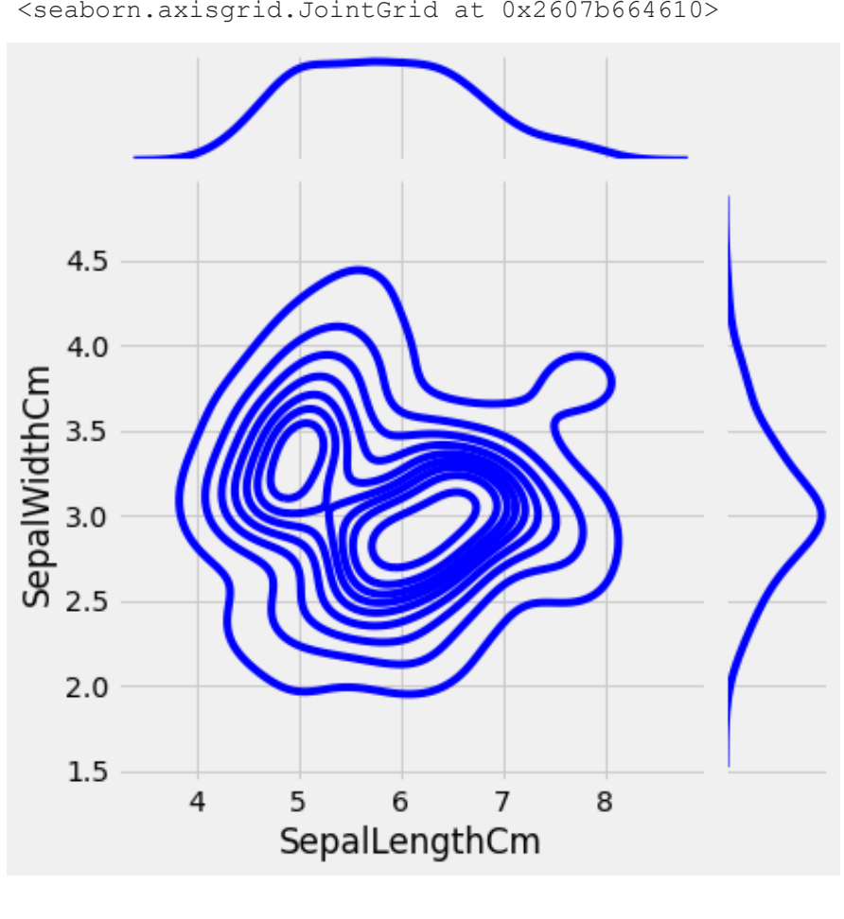
Gunakan jenis kernel density estimate plot

In [69]:

```python
sns.jointplot("SepalLengthCm", "SepalWidthCm", data=df, kind="kde", color='b')
```

Out[69]:

```
<seaborn.axisgrid.JointGrid at 0x2607b664610>
```



Latihan (11)

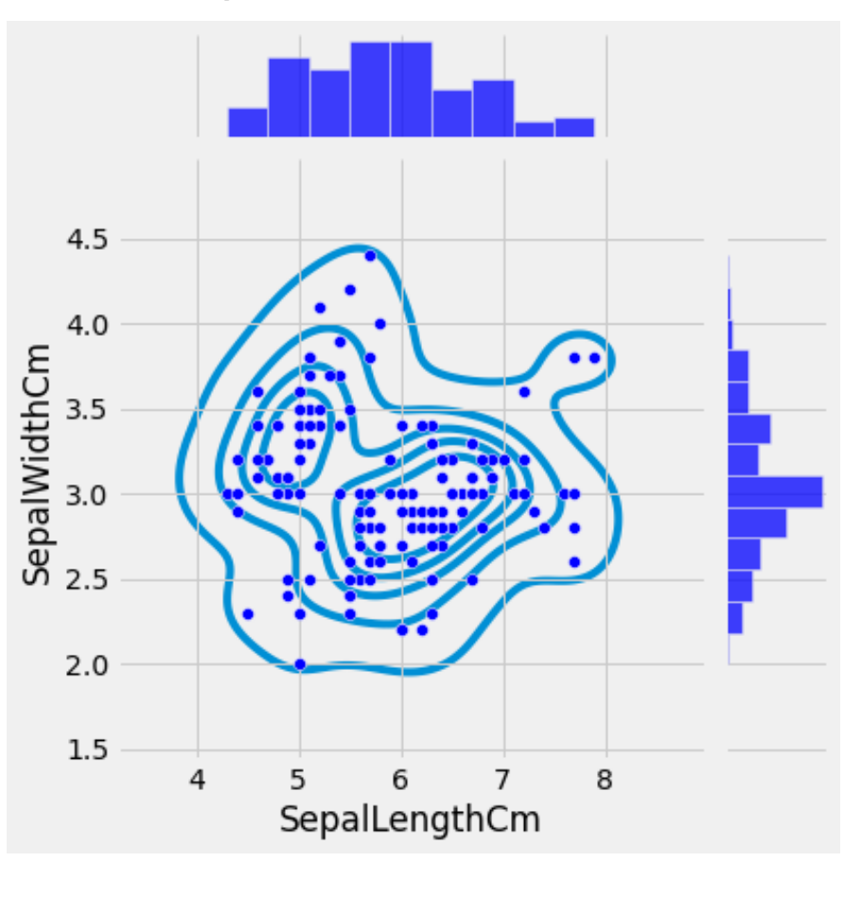
Lengkapi potongan code penggabungan plot

In [70]:

```python
sns.jointplot("SepalLengthCm", "SepalWidthCm", data=df, color="b").plot_joint(sns.kdeplot, zorder=0, n_levels=
```

Out[70]:

```
<seaborn.axisgrid.JointGrid at 0x2607c752f70>
```



Latihan (12)

Visualisasi Boxplot untuk memberikan ringkasan statis dari fitur yang diplot.

- Garis atas mewakili nilai maksimal
- Tepi atas kotak adalah Kuartil ketiga
- Tepi tengah adalah median,
- Tepi bawah adalah nilai kuartil pertama.
- Garis paling bawah adalah nilai minimum.
- Ketinggian kotak disebut sebagai rentang Interkuartil.
- Titik-titik hitam pada plot adalah nilai outlier dalam data.

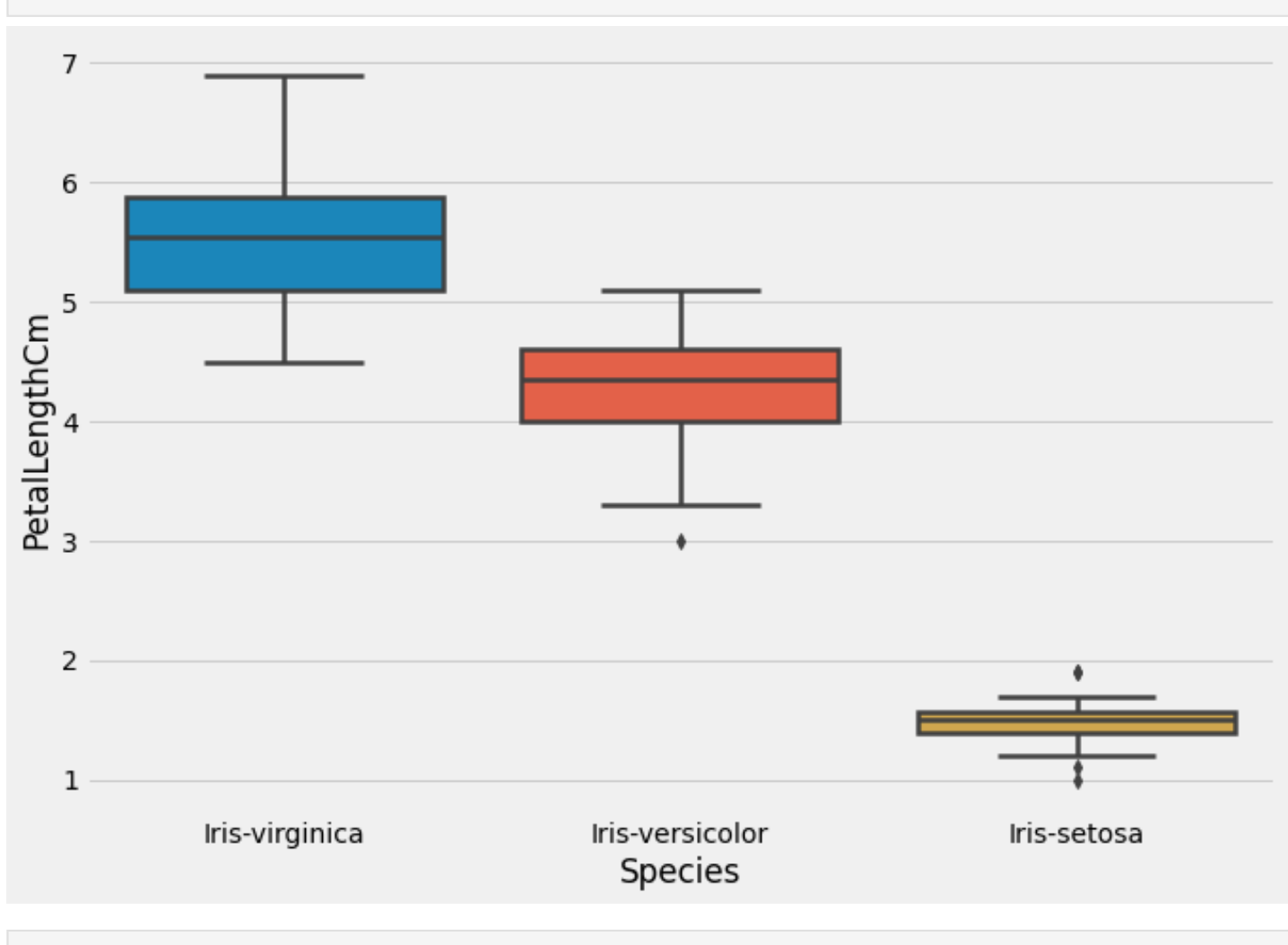
Tampilkan visualisasi boxplot menggunakan kolom "Species" dan "PetalLengthCm" dalam dataset iris

In [71]:

```python
# visualisasi Boxplot
```

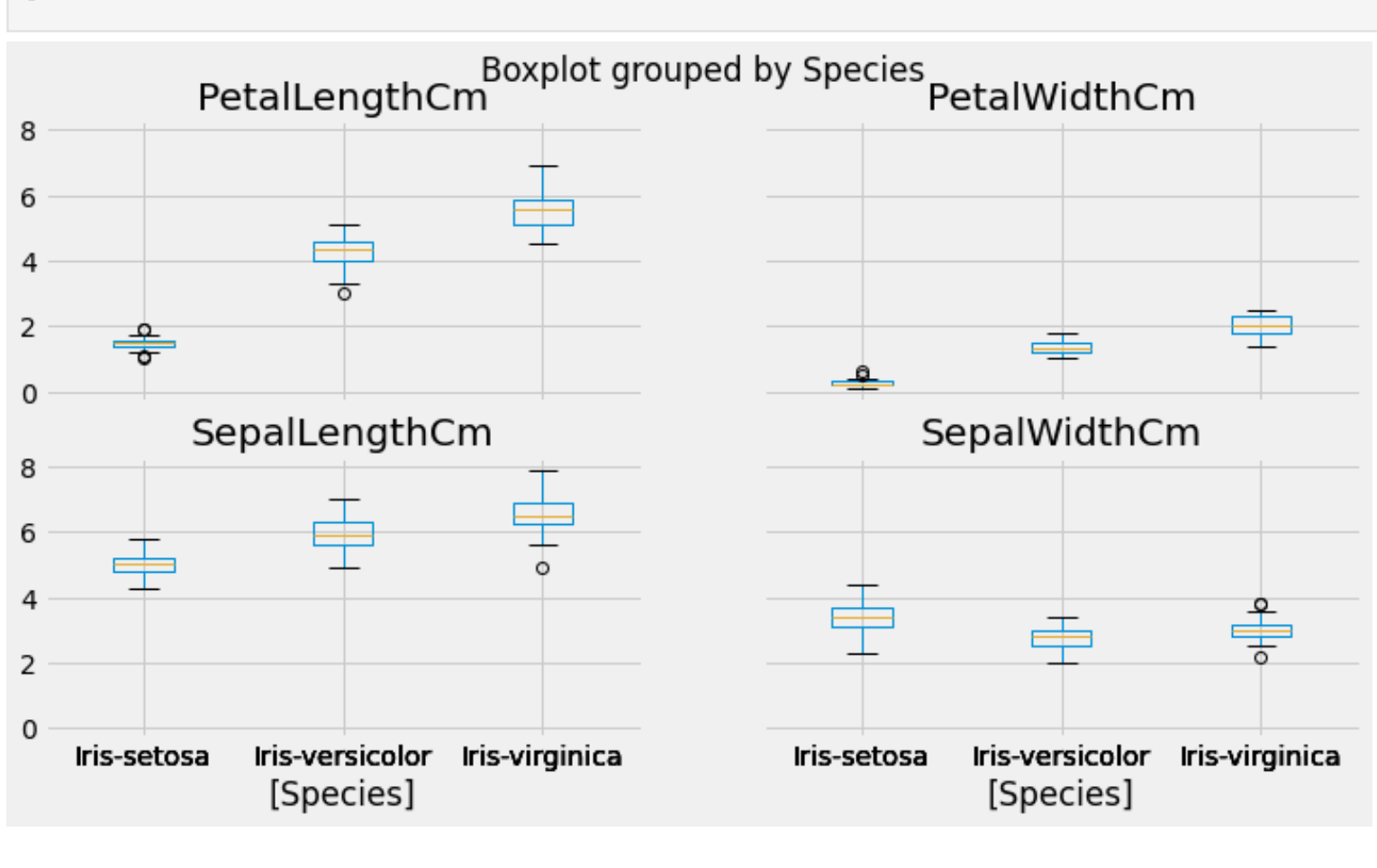
In [72]:

```python
fig=plt.gcf()
fig.set_size_inches(10,7)
fig=sns.boxplot(x='Species',y='PetalLengthCm',data=df,order=['Iris-virginica','Iris-versicolor','Iris-setosa'],
```



In [73]:

```python
# visualisasi Boxplot yang di kelompokkan berdasarkan "Species"
df.boxplot(by="Species", figsize=(12, 6))
pass
```



Latihan (13)

Visualisasi Violin Plot untuk memvisualisasikan sebaran data dan distribusi probabilitas.

- Bilah hitam tebal di tengah mewakili rentang interkuartil
- Garis hitam tipis yang memanjang darinya mewakili interval kepercayaan 95%
- Titik putih adalah median.

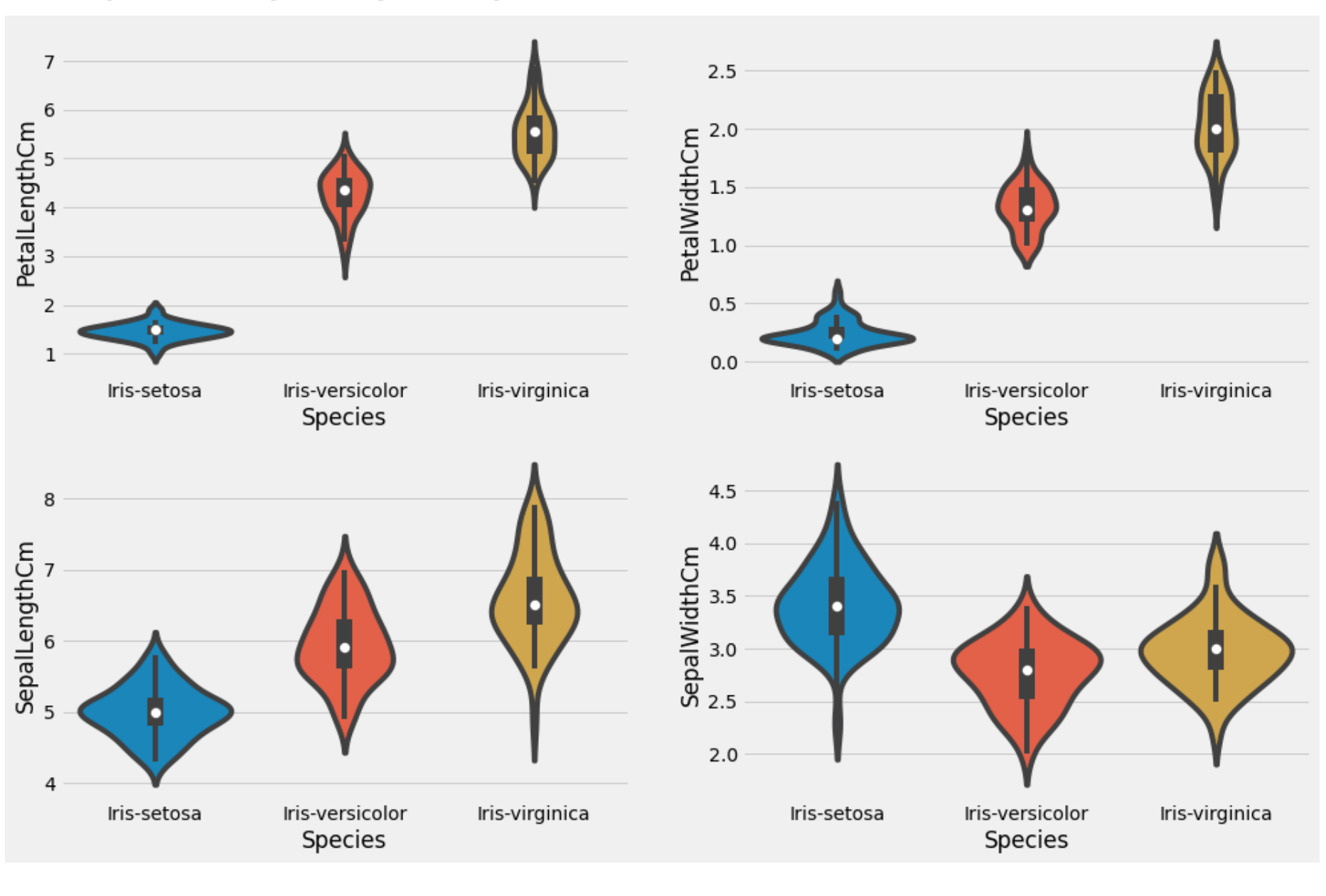
Tampilkan visualisasi Violin Plot dengan menggunakan setiap kolom yang ada untuk melihat sebaran data terhadap kolom "Species" dalam dataset iris

In [74]:

```python
plt.figure(figsize=(15, 10))
plt.subplot(2, 2, 1)
sns.violinplot(x='Species', y='PetalLengthCm', data=df)
plt.subplot(2, 2, 2)
sns.violinplot(x='Species', y='PetalWidthCm', data=df)
plt.subplot(2, 2, 3)
sns.violinplot(x='Species', y='SepalLengthCm', data=df)
plt.subplot(2, 2, 4)
sns.violinplot(x='Species', y='SepalWidthCm', data=df)
```

Out[74]:

```
<AxesSubplot:xlabel='Species', ylabel='SepalWidthCm'>
```



Latihan (14)

Visualisasi pairplot yang juga dikenal dengan scatterplot adalah visualisasi sebaran data yang menunjukan keterkaitan antar kolom.

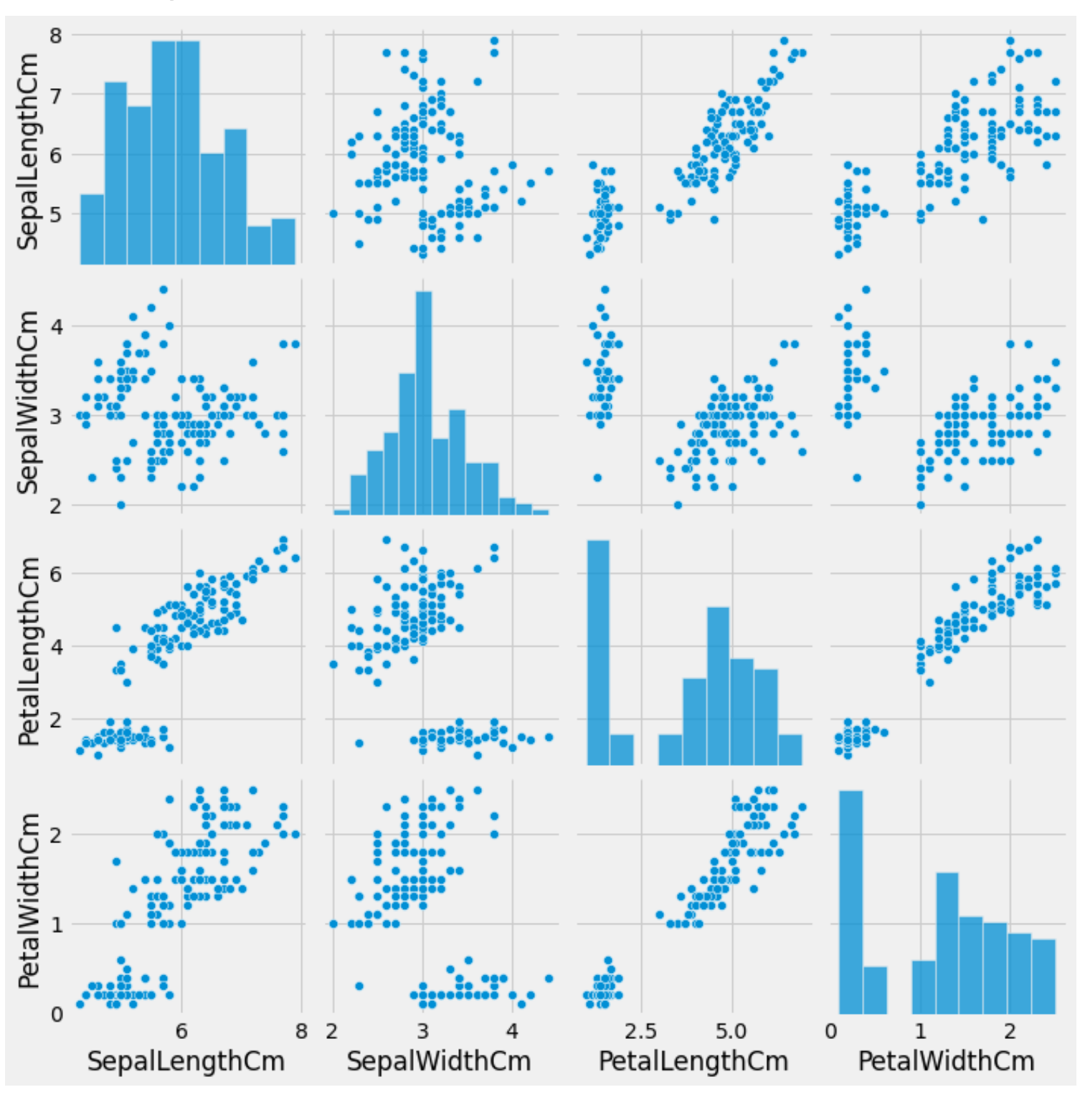
Tampilkan visualisasi pairplot dari setiap kolom yang ada untuk melihat sebaran data dalam dataset iris

In [75]:

```python
# visualisasi pairplot
sns.pairplot(data=df,kind='scatter')
```

Out[75]:

```
<seaborn.axisgrid.PairGrid at 0x260790cf1f0>
```



Latihan (15)

Visualisasi heatmap digunakan untuk mengetahui korelasi setiap kolom dalam dataset. Nilai positif atau negatif yang tinggi menunjukkan bahwa fitur tersebut memiliki korelasi yang tinggi. Hal ini membantu kita memilih parameter untuk machine learning.

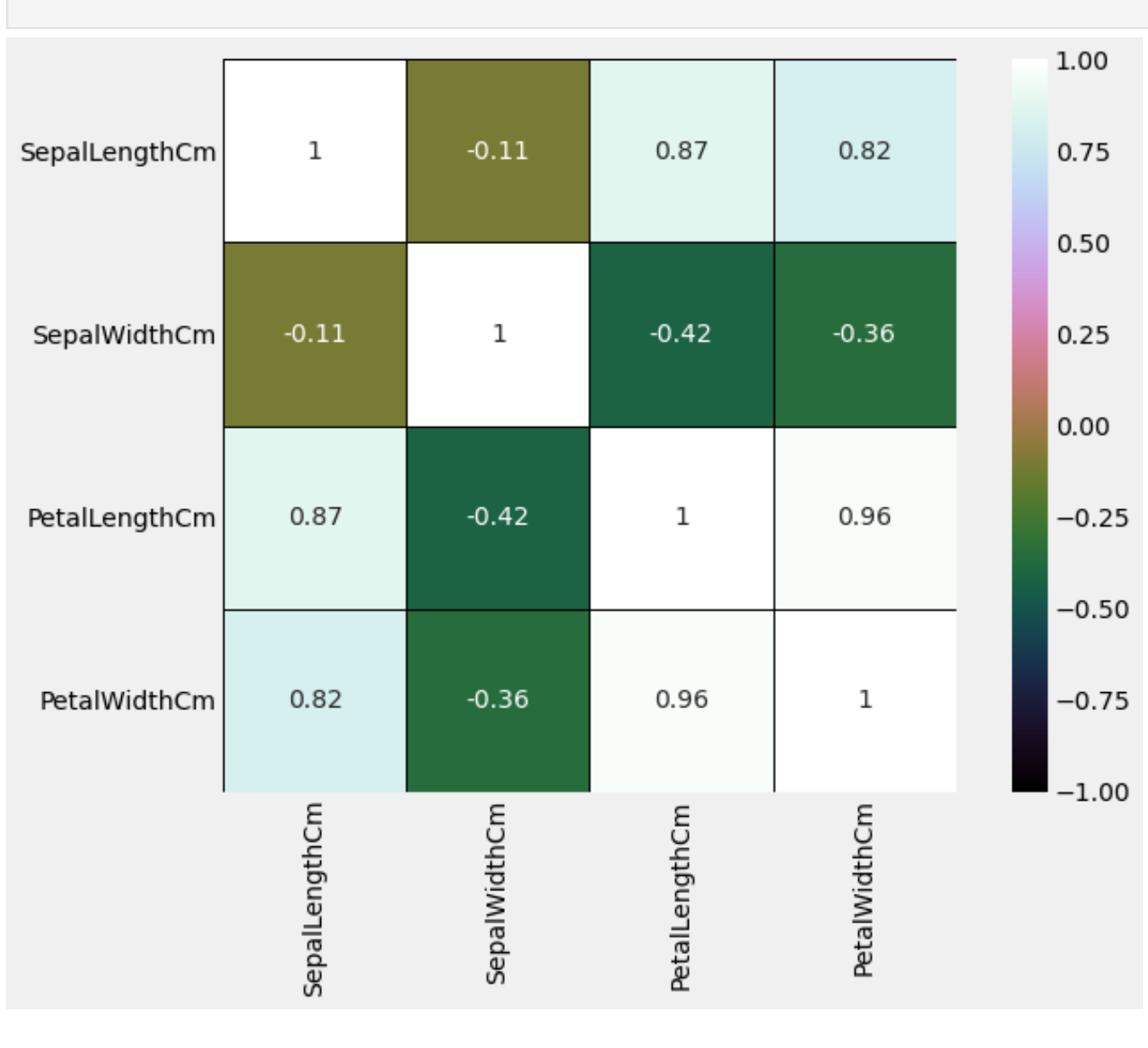
Tampilkan visualisasi heatmap dari korelasi setiap fitur dalam dataset iris

In [76]:

```python
# visualisasi heatmap
```

In [77]:

```python
fig=plt.gcf()
fig.set_size_inches(10,7)
fig=sns.heatmap(df.corr(),annot=True,cmap='cubehelix',linewidths=1,linecolor='k',square=True,mask=False, vmin=-
```



Latihan (16)

Visualisasi Distribution plot untuk membandingkan jangkauan dan distribusi untuk dataset numerik. Visualisasi Distribution plot tidak relevan untuk analisis data yang terperinci karena berkaitan dengan ringkasan distribusi data.

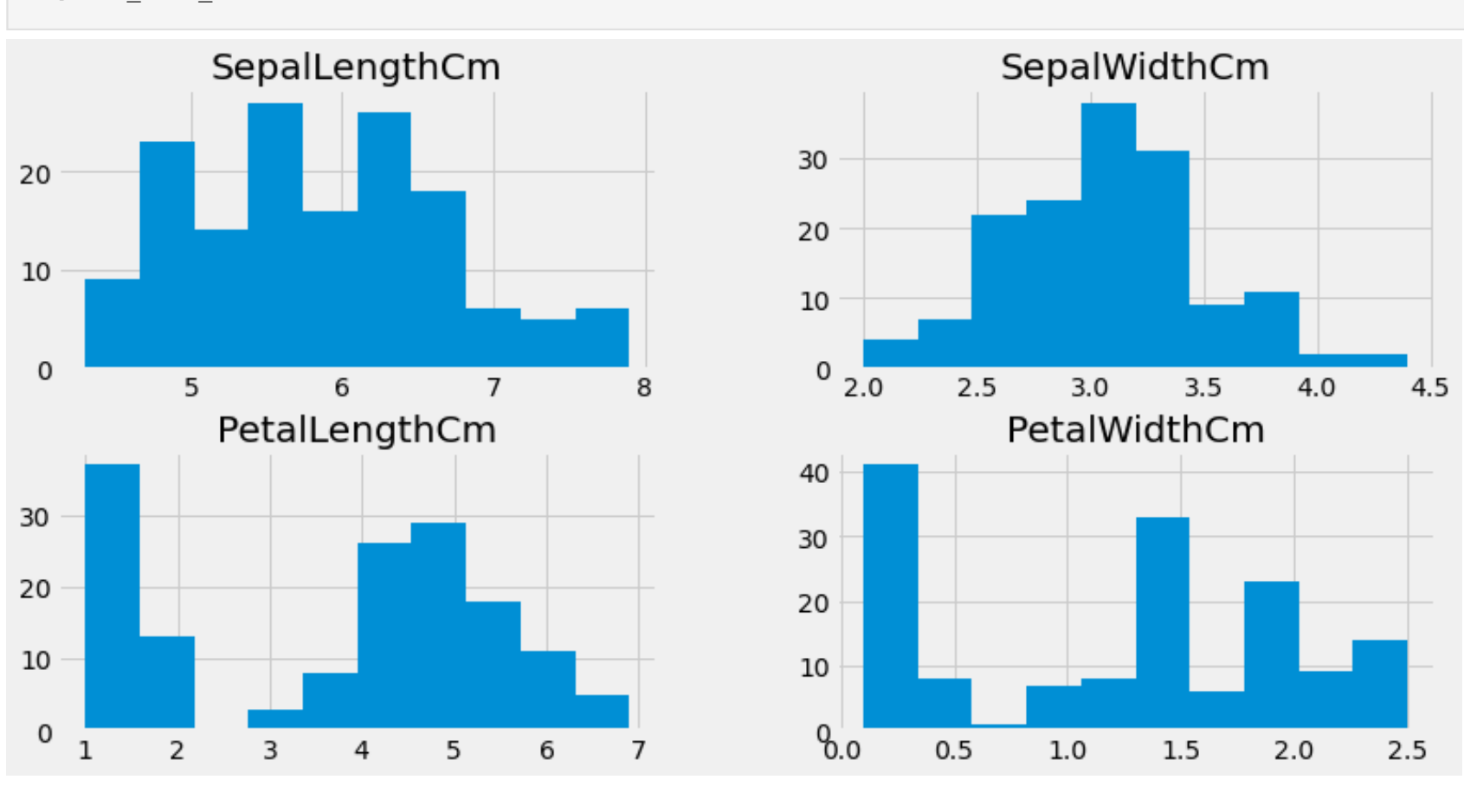
Tampilkan visualisasi Distribution plot setiap fitur dalam dataset iris

In [78]:

```python
# visualisasi distribution plot
```

In [79]:

```python
df.hist(linewidth=1.2)
fig=plt.gcf()
fig.set_size_inches(12,6)
```



Latihan (17)

Visualisasi Residual Plot adalah visualisasi untuk memplot residu dengan nilai prediksi pada sumbu x, dan residu pada sumbu y. Jarak dari garis 0 adalah seberapa buruk prediksi untuk nilai yang di plot.

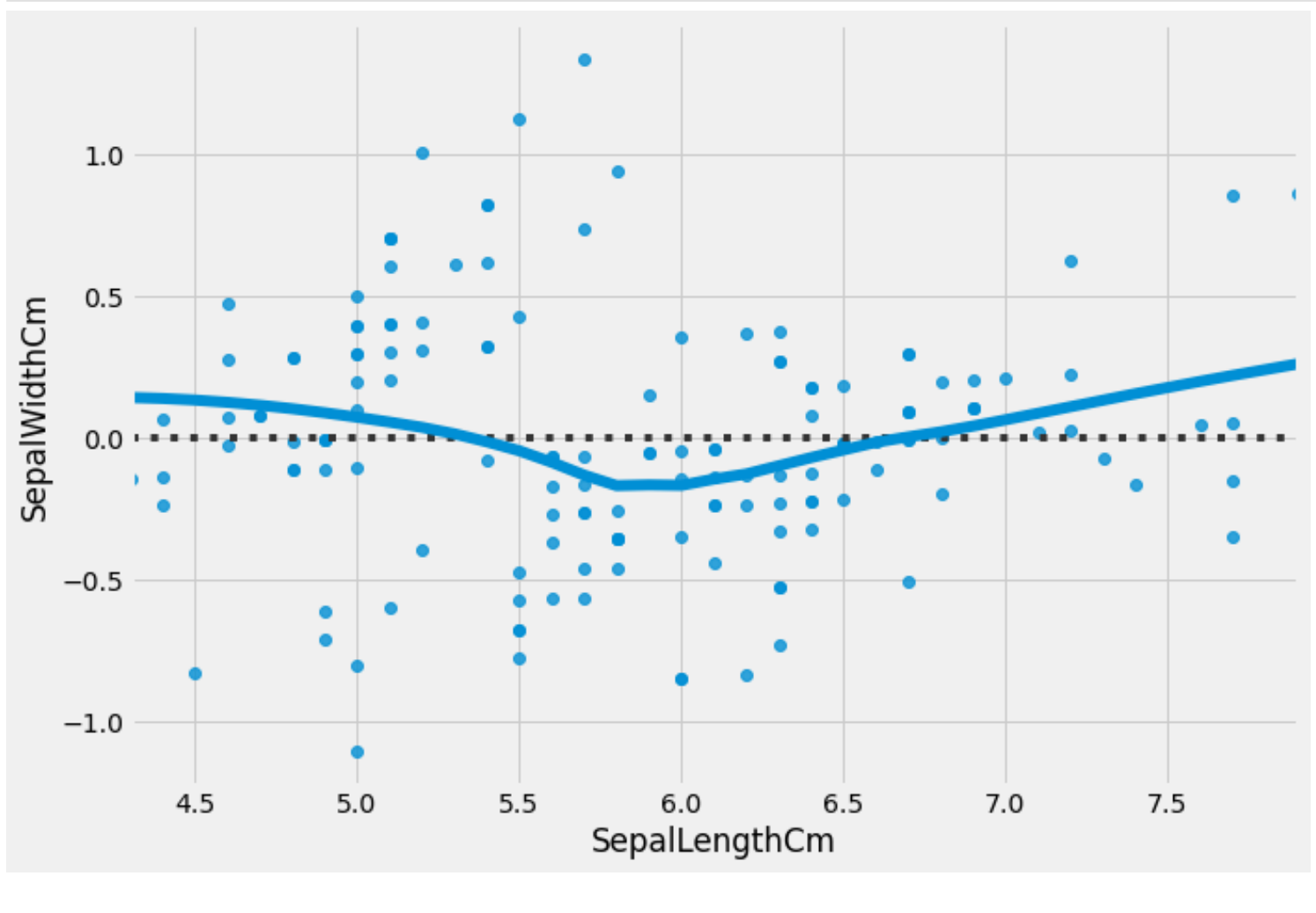
Tampilkan visualisasi Residual Plot dari fitur "SepalLengthCm" dan SepalWidthCm dalam dataset iris

In [80]:

```python
# visualisasi Residual Plot
```

In [81]:

```python
fig=plt.gcf()
fig.set_size_inches(10,7)
fig=sns.residplot('SepalLengthCm', 'SepalWidthCm',data=df,lowess=True)
```



Latihan (18)

Visualisasi Stacked Histogram digunakan untuk menunjukkan bagaimana fitur yang lebih besar dibagi menjadi fitur yang lebih kecil dan menunjukan hubungan masing-masing fitur terhadap jumlah total

Tampilkan visualisasi Stacked Histogram dari fitur "Species" dengan mengubah tipe datanya menjadi *category* (astype) dalam dataset iris

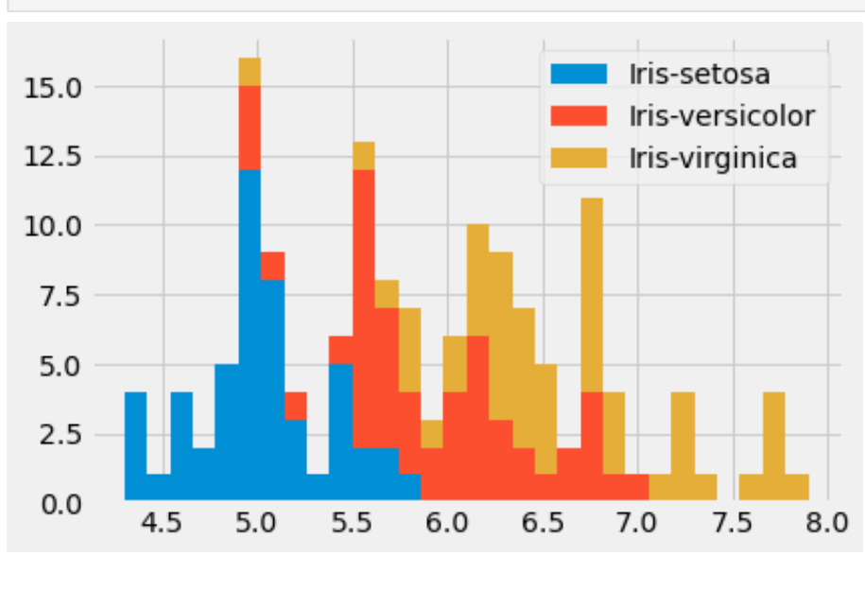
In [82]:

```python
# visualisasi stacked histogram
```

In [89]:

```python
df['Species'] = df['Species'].astype('category')
#df.head()
list1=list()
mylabels=list()
for gen in df.Species.cat.categories:
    list1.append(df[df.Species==gen].SepalLengthCm)
    mylabels.append(gen)

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



Latihan (19)

Visualisasi Area Plot memberi kita representasi visual dari Berbagai dimensi dalam dataset.

Tampilkan visualisasi Area Plot dari fitur 'SepalLengthCm','SepalWidthCm','PetalLengthCm','PetalWidthCm' dalam dataset iris

In [ ]:

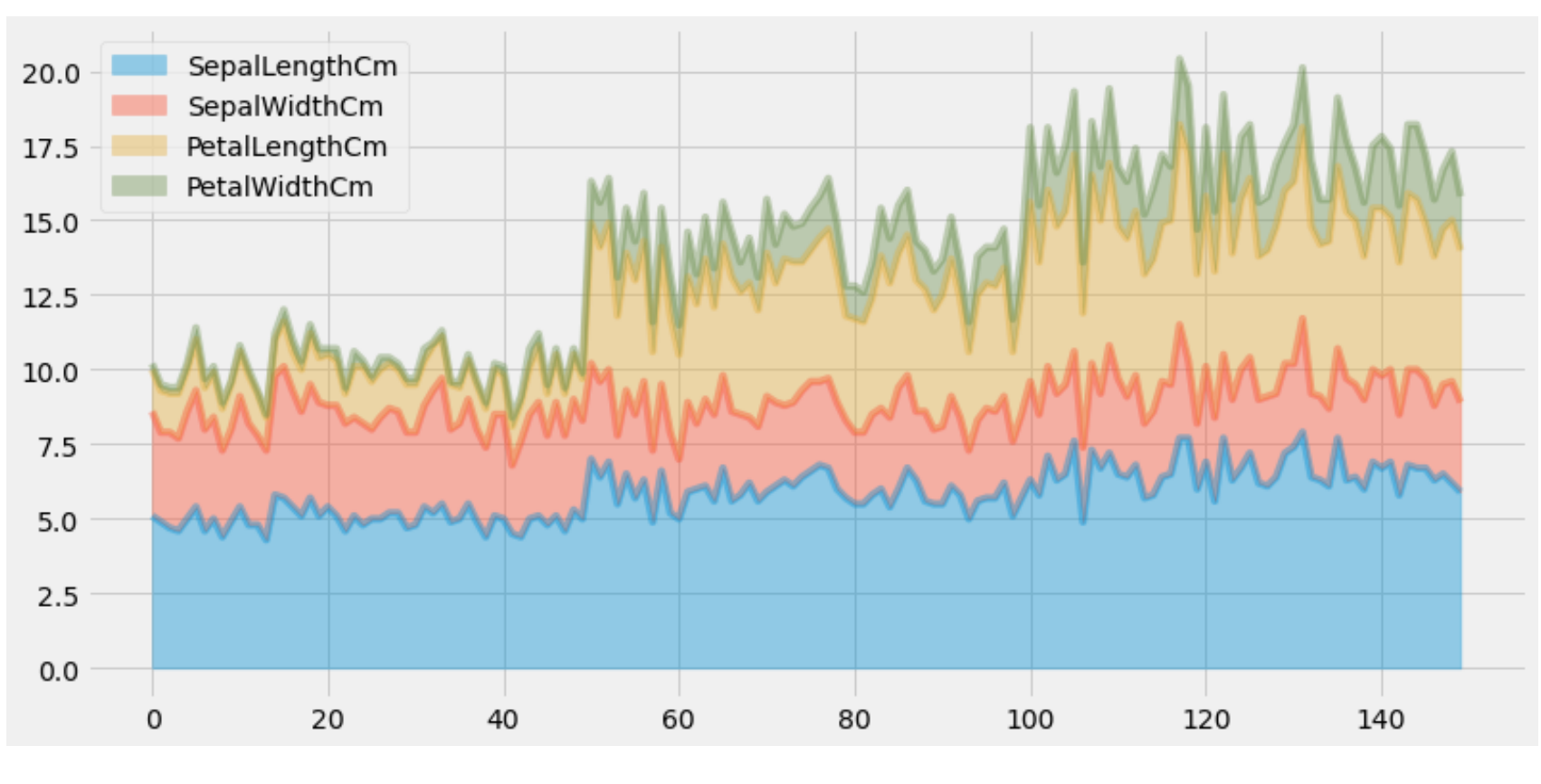
```python
# visualisasi Area Plot
```

In [88]:

```python
df.plot.area(y=['SepalLengthCm','SepalWidthCm','PetalLengthCm','PetalWidthCm'],alpha=0.4,figsize=(12, 6))
```

Out[88]:

```
<AxesSubplot:>
```



Latihan (20)

Visualisasi distplot membantu untuk melihat distribusi variabel tunggal. Kde menunjukkan kepadatan distribusi

Tampilkan visualisasi distplot dari fitur 'SepalLengthCm' dengan menggunakan *Kde* (kind) untuk menunjukkan kepadatan distribusi dalam dataset iris

In [ ]:

```python
# visualiasi distplot
```

In [90]:

```python
sns.distplot(df['SepalLengthCm'], kde=True, bins=20)
```

Out[90]:

```
<AxesSubplot:xlabel='SepalLengthCm', ylabel='Density'>
```