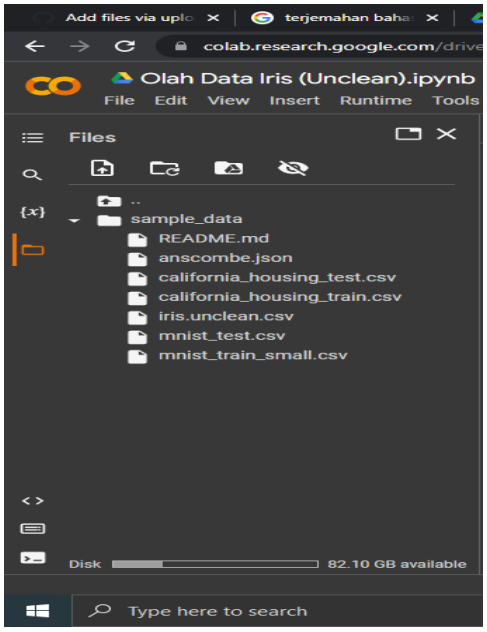
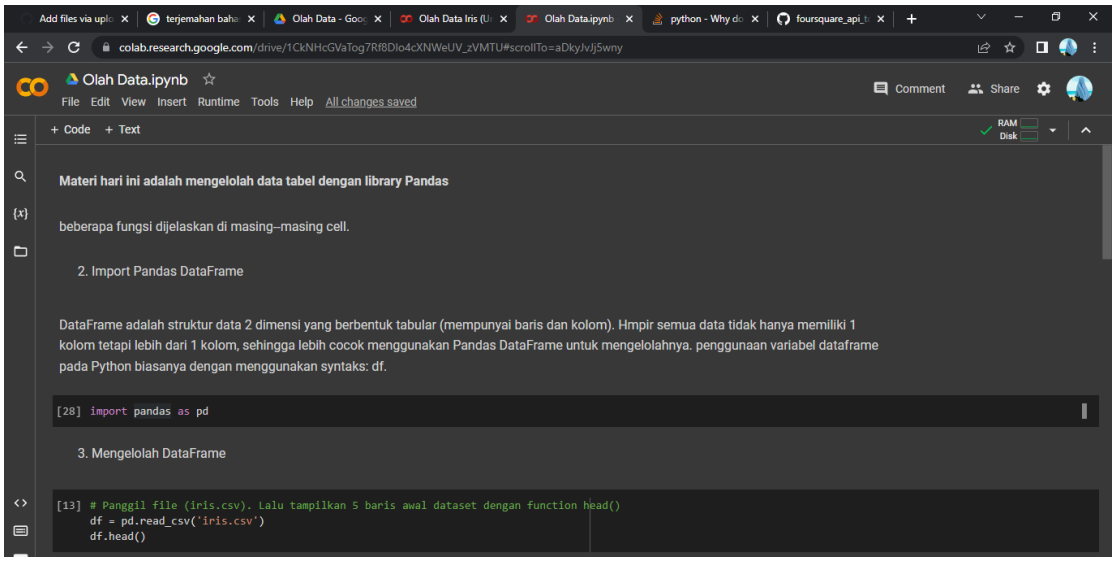
Nama : Nurmala

Nim : 20.01.013.069

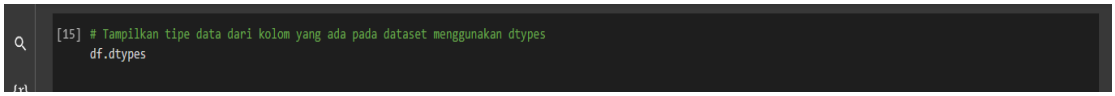
Tugas : Pemrograman Python

1. Pemrograman Python Olah Data





	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



```
Id                int64
SepalLengthCm     float64
SepalWidthCm      float64
PetalLengthCm     float64
PetalWidthCm      float64
Species            object
dtype: object
```

```
[27] # Hitung ukuran (jumlah baris dan kolom) dari dataset
     df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 150 entries, 0 to 149
Data columns (total 6 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   Id             150 non-null    int64
 1   SepalLengthCm  150 non-null    float64
 2   SepalWidthCm   150 non-null    float64
 3   PetalLengthCm  150 non-null    float64
 4   PetalWidthCm   150 non-null    float64
 5   Species        150 non-null    object
dtypes: float64(4), int64(1), object(1)
memory usage: 7.2+ KB
```

```
[19] # Tampilkan data untuk kolom "Id" dan kolom "Species" dalam bentuk dataframe
     df[["Id","Species"]]
```

	Id	Species
0	1	Iris-setosa
1	2	Iris-setosa
2	3	Iris-setosa
3	4	Iris-setosa
4	5	Iris-setosa
...	...	...
145	146	Iris-virginica
146	147	Iris-virginica
147	148	Iris-virginica
148	149	Iris-virginica
149	150	Iris-virginica

150 rows × 2 columns

```
[20] # Tampilkan data baris indexs ke-0 (nol) sampai dengan index ke-9 (sembilan)
     df.iloc[:10]
```

	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
5	6	5.4	3.9	1.7	0.4	Iris-setosa
6	7	4.6	3.4	1.4	0.3	Iris-setosa
7	8	5.0	3.4	1.5	0.2	Iris-setosa
8	9	4.4	2.9	1.4	0.2	Iris-setosa
9	10	4.9	3.1	1.5	0.1	Iris-setosa

```
[25] # Tampilkan data hanya kolom "Id" dan kolom "Species", dan yang ditampilkan adalah data index ke-11 (sebelas)
     # sampai dengan index ke-15 (limabelas)
     df[["Id", "Species"]].iloc[11:16]
```

	Id	Species
11	12	Iris-setosa
12	13	Iris-setosa
13	14	Iris-setosa
14	15	Iris-setosa
15	16	Iris-setosa

```
[23] # Tampilkan data 8 baris pertama
     df.head(8)
```

	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
5	6	5.4	3.9	1.7	0.4	Iris-setosa
6	7	4.6	3.4	1.4	0.3	Iris-setosa
7	8	5.0	3.4	1.5	0.2	Iris-setosa

```
[24] # Tampilkan data 3 baris terakhir
     df.tail(3)
```

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
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

```
[29] # Hitung nilai mean dari dataset
     df.mean()
```

```
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:2: FutureWarning: Dropping of

Id               75.500000
SepalLengthCm     5.843333
SepalWidthCm      3.054000
PetalLengthCm     3.758667
PetalWidthCm      1.198667
dtype: float64
```

```
[30] # Hitung nilai mean untuk kolom PetalLengthCm
     df["PetalLengthCm"].mean()
```

```
3.7586666666666693
```

```
[32] # cari nilai minimal untuk kolom SepalWidthCm
     df["Species"].value_counts()
```

```
2.0
```

```
[33] # Hitung frekuensi pada kolom Species dengan mmenggunakan metode value_counts()
     df["Species"].value_count()
```

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

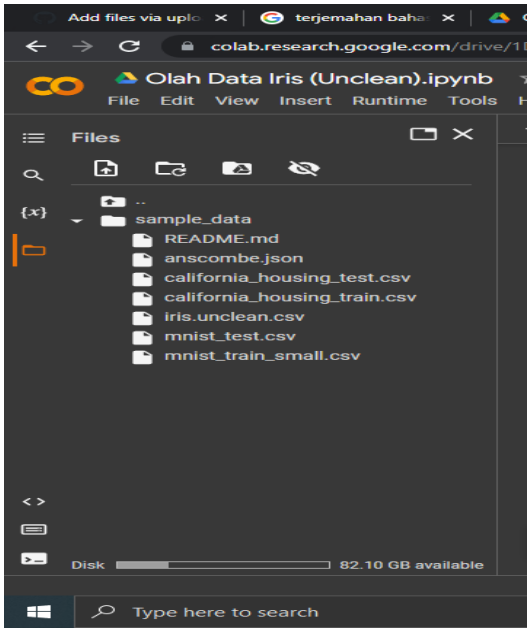
```
[36] # Tampilkan perhitungan frekuensi pada kolom Species dengan menggunakan value_counts() dalam bentuk dataframe
     dfValueCountsSpecies = df["Species"].value_counts().rename_axis('Species Value Counts').reset_index(name='Counts')
     dfValueCountsSpecies
```

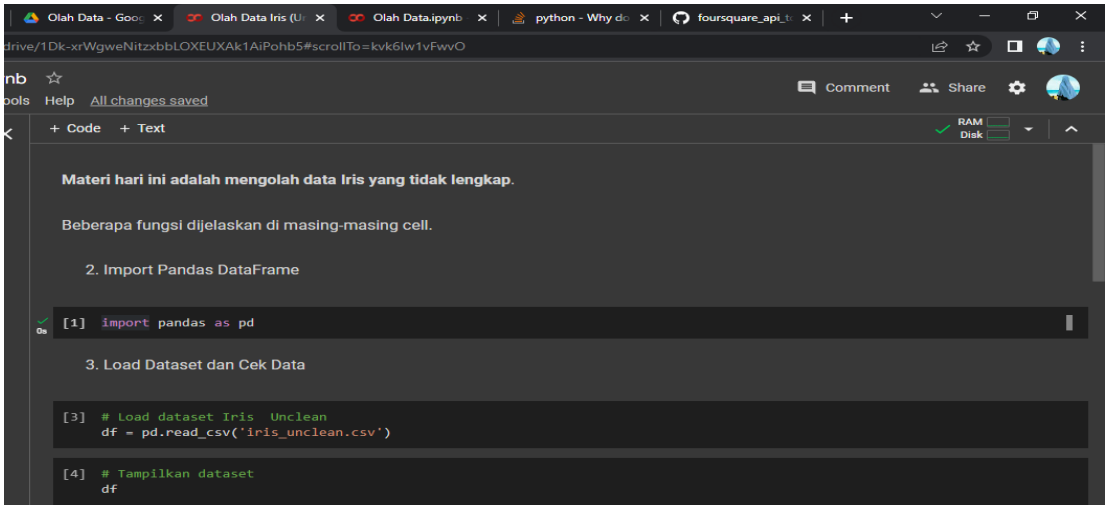
	Species Value Counts	Counts
0	Iris-setosa	50
1	Iris-versicolor	50
2	Iris-virginica	50

```
[37] # Hitung frekuensi pada kolom PetalLenghCm dengan menggunakan value_counts() dalam bentuk dataframe
     dfValueCountsPetalLenghCm = df["PetalLenghCm"].value_counts().rename_axis('PetalLenghCm Value Counts').reset_index(name='Counts')
     dfValueCountsPetalLenghCm
```

	PetalLengthCm Value Counts	Counts
0	1.5	14
1	1.4	12
2	5.1	8
3	4.5	8
4	1.6	7
5	1.3	7

2. Olah Data Iris Dataset (Unclean)



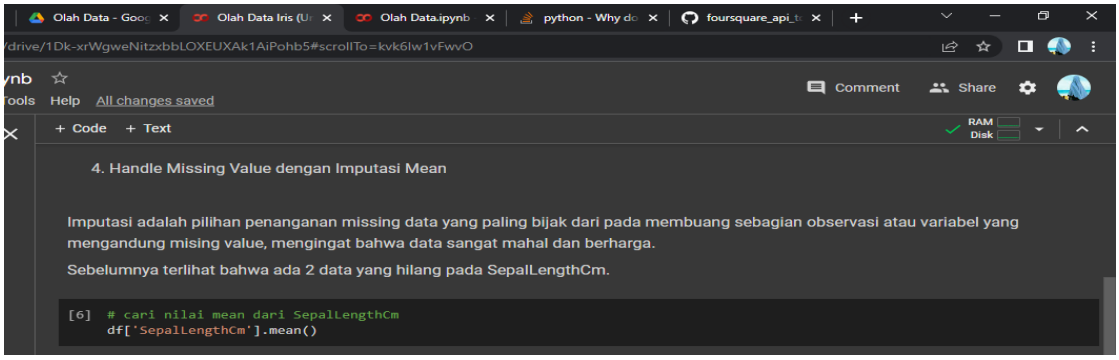


	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	NaN	3.5	1.4	0.2	Iris-setosa
1	4.9	2000.0	1.4	0.2	Iris-setosa
2	4.7	3.2	-1.3	0.2	Iris-setosa
3	4.6	3.1	1.5	0.2	Iris-setosa
4	5.0	3.6	1.4	0.2	Iris-setosa
...	...	...	...	...	...
145	6.7	3.0	5.2	2.3	Iris-virginica
146	6.3	2.5	5.0	1.9	Iris-virginica
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

```
[5] # Hitung jumlah nilai null pada dataset
    df.isna().sum()
```

```
SepalLengthCm    2
SepalWidthCm     0
PetalLengthCm    0
PetalWidthCm     0
Species          0
dtype: int64
```




```
5.856756756756758
```

```
[7] # Mengganti missing value dengan mean(), kemudian masukan ke variabel
    dfDataBaru = df["SepalLenghtCm"].fillna(df['SepalLenghtCm']).mean())
[8] # cek data
    dfDataBaru
```

```
0      5.856757
1      4.900000
2      4.700000
3      4.600000
4      5.000000
       ...
145    6.700000
146    6.300000
147    6.500000
148    6.200000
149    5.900000
Name: SepalLengthCm, Length: 150, dtype: float64
```

```
[9]  # Gabung data baru menjadi DataFrame
     df2 = pd.DataFrame({'SepalLengthCm': dfDataBaru, 'SepalWidthCm': df['SepalWidthCm'],
                         'PetalLengthCm': df['PetalLengthCm'], 'PetalWidthCm': df['PetalWidthCm'],
                         'Species': df['Species']})
```

	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	5.856757	3.5	1.4	0.2	Iris-setosa
1	4.900000	2000.0	1.4	0.2	Iris-setosa
2	4.700000	3.2	-1.3	0.2	Iris-setosa
3	4.600000	3.1	1.5	0.2	Iris-setosa
4	5.000000	3.6	1.4	0.2	Iris-setosa
...	...	...	...	...	...
145	6.700000	3.0	5.2	2.3	Iris-virginica
146	6.300000	2.5	5.0	1.9	Iris-virginica
147	6.500000	3.0	5.2	2.0	Iris-virginica
148	6.200000	3.4	5.4	2.3	Iris-virginica
149	5.900000	3.0	5.1	1.8	Iris-virginica

150 rows × 5 columns

```
[10] # cek jumlah baris dan kolom
     df2.info()
```

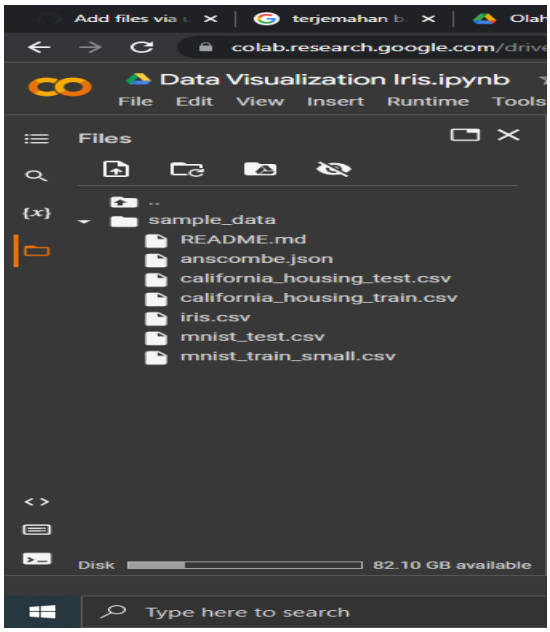
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 150 entries, 0 to 149
Data columns (total 5 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   SepalLengthCm  150 non-null    float64
 1   SepalWidthCm   150 non-null    float64
 2   PetalLengthCm  150 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
```

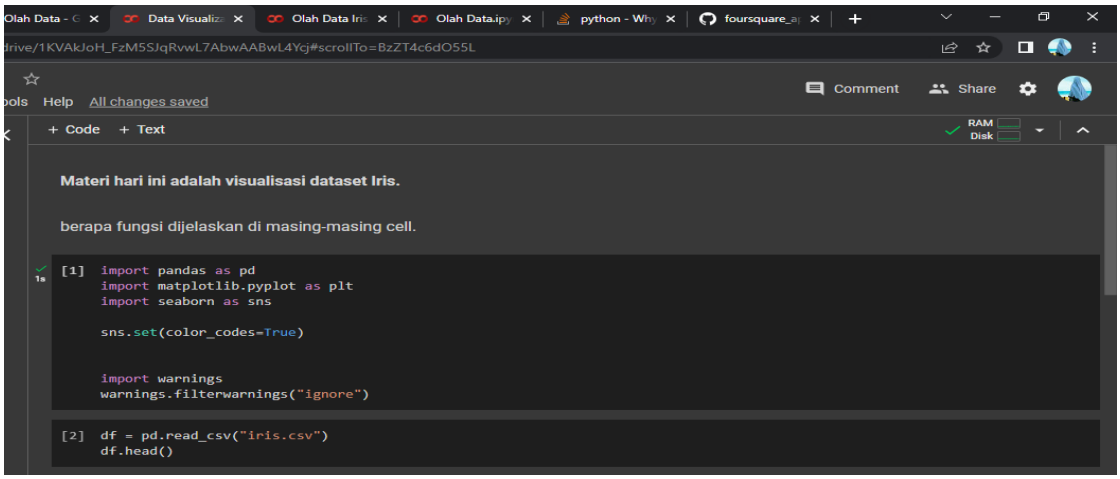
```
[11] # Hitung jumlah nilai null pada DataFrame baru
     df2.isna().sum()
```

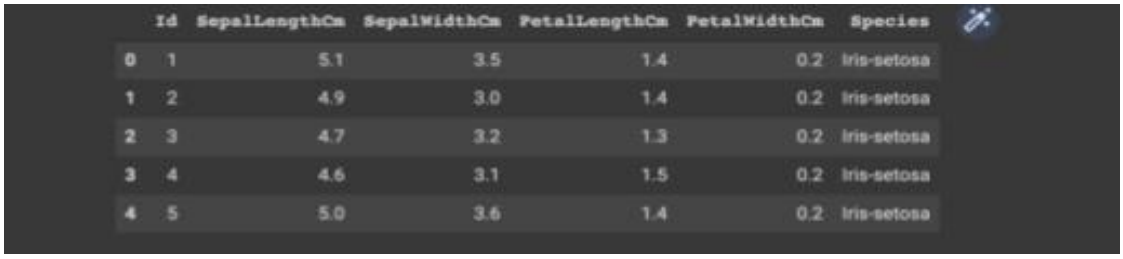
```
SepalLengthCm    0
SepalWidthCm     0
PetalLengthCm    0
PetalWidthCm     0
Species          0
dtype: int64
```

Nilai null sudah terisi semua

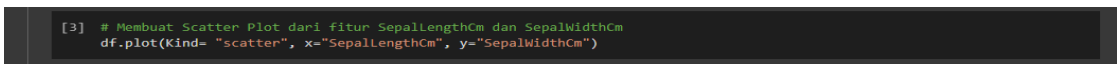
3. Data Visualization Iris



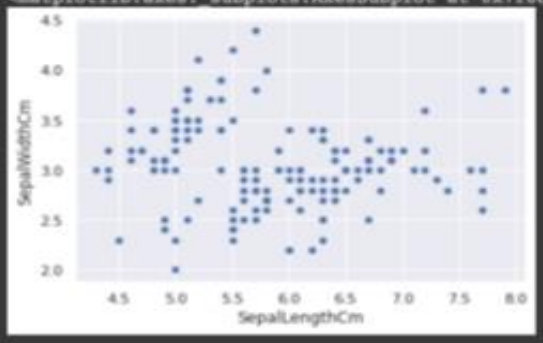




	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	1	5.1	3.5	1.4	0.2	Iris-setosa
1	2	4.9	3.0	1.4	0.2	Iris-setosa
2	3	4.7	3.2	1.3	0.2	Iris-setosa
3	4	4.6	3.1	1.5	0.2	Iris-setosa
4	5	5.0	3.6	1.4	0.2	Iris-setosa

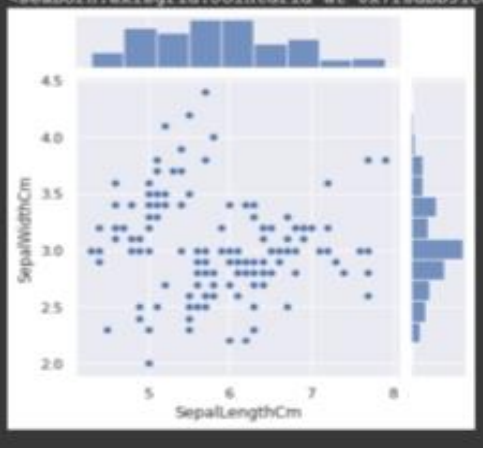


```
*c* argument looks like a single numeric RGB or RGBA sequence, which should be avoided a
<matplotlib.axes._subplots.AxesSubplot at 0x7f8db9c18910>
```



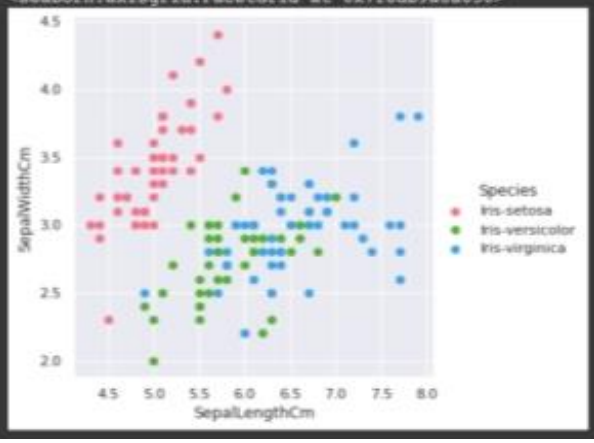

```
[4]  # Atau dengan library Seaborn
     sns.jointplot( x="SepalLengthCm", y="SepalWidthCm", data=df, size=5)
```

```
<seaborn.axisgrid.JointGrid at 0x7f8dbb51cdd0>
```

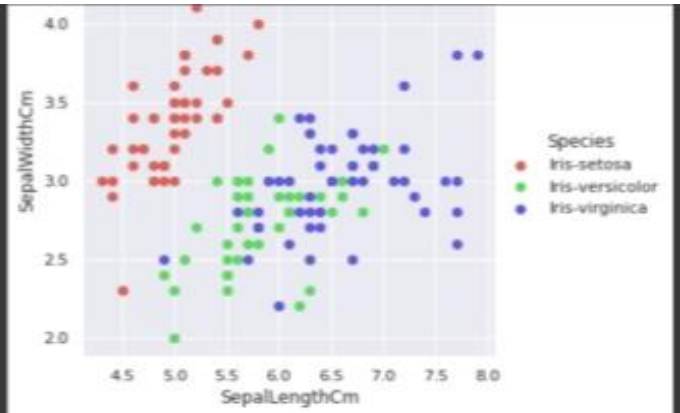


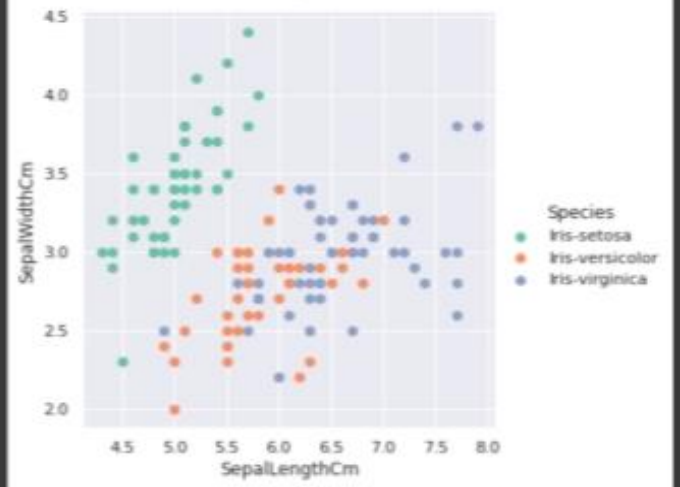

```
[5]  # Salah satu informasi yang hilang dalam plot di atas adalah jenis tanaman (Species)
     # Gunakan FacetGrid Seaborn untuk mewarnai sebaran Species
     sns.FacetGrid(df, hue="Species", palette= "husl", sizw=5) \
        .map(plt.scatter, "SepalLengthCm", "SepalWidthCm") \
        .add_legend()
```

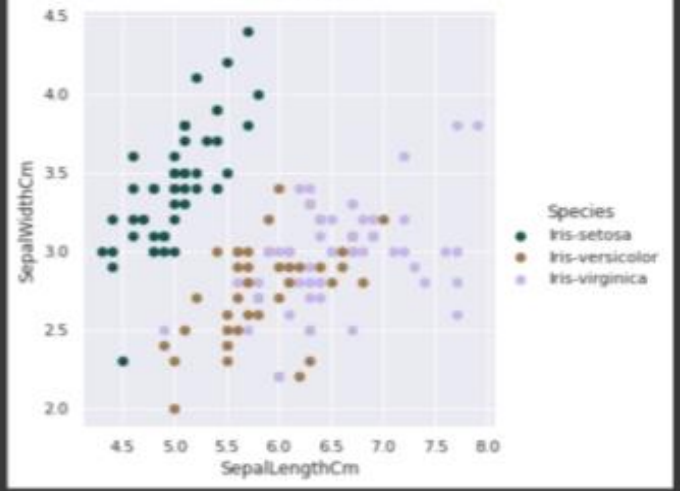
```
<seaborn.axisgrid.FacetGrid at 0x7f8db9a8a650>
```



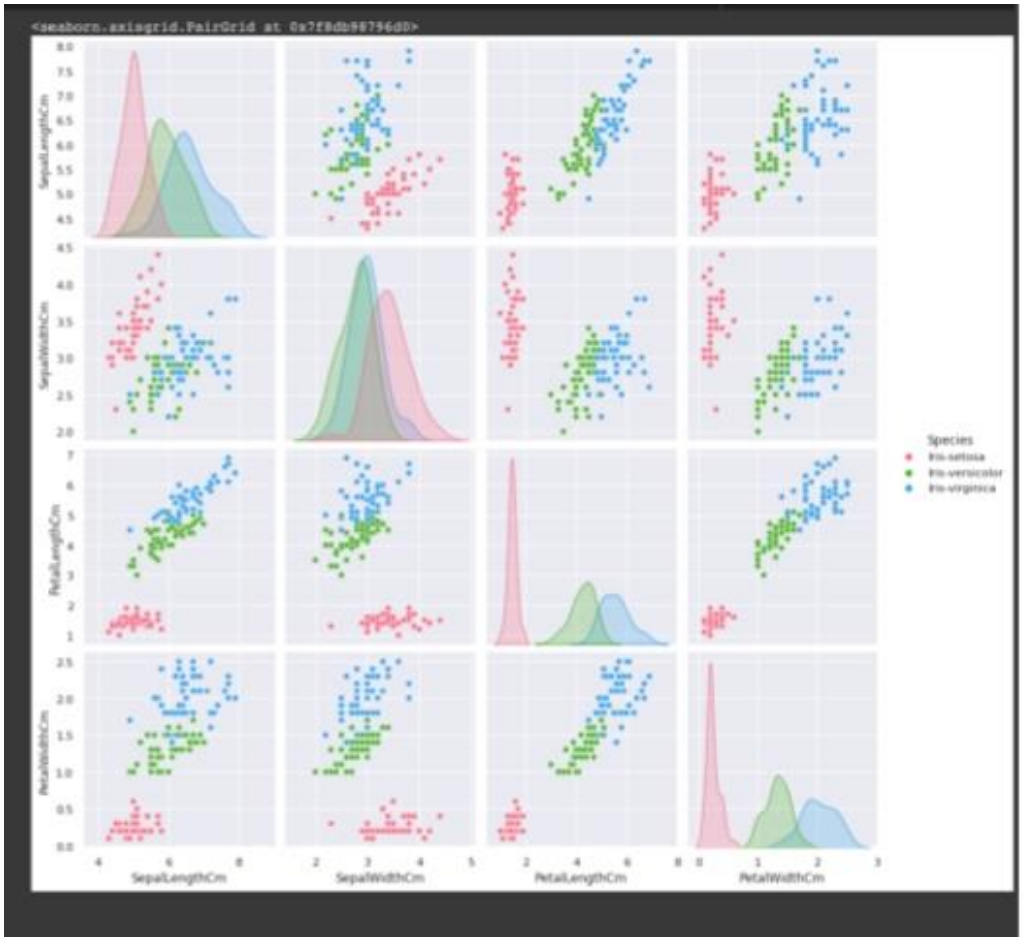

```
[6]  # 3 plot dengan jenis Pallete atau warna yang berbeda
     sns.FacetGrid(df, hue="Species", palette= "hls", sizw=5) \
        .map(plt.scatter, "SepalLengthCm", "SepalWidthCm") \
        .add_legend()

     sns.FacetGrid(df, hue="Species", palette= "Set2", sizw=5) \
        .map(plt.scatter, "SepalLengthCm", "SepalWidthCm") \
        .add_legend()

     sns.FacetGrid(df, hue="Species", palette= "cubehelix", sizw=5) \
        .map(plt.scatter, "SepalLengthCm", "SepalWidthCm") \
        .add_legend()
```








```
[7]  # pirplot
     # Dibawah ini ada kolom Id yang dihapus karena tidak memiliki korelasi dengan variabel lain
     sns.pairplot(df.drop("Id", axis=1), hue="Species", palette="husl", size=3)
```


<seaborn.axisgrid.PairGrid at 0x7f8db98796d0>
SepalLengthCm
SepalWidthCm
PetalLengthCm
PetalWidthCm
Species
Iris-setosa
Iris-versicolor
Iris-virginica