

```
# Name:Atharva Kangralkar
# Class:CS(AIML) - A
# Roll: 54
# PRN:12311493
# LINK:- https://colab.research.google.com/drive/1cqCtX0rNbHgZ0TPXYpazKyzbrTIqg00I?usp=sharing
```

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np
```

```
df = pd.read_csv("/content/Iris.csv")
```

```
df.head()
```

	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

Next steps:

[Generate code with df](#)
[View recommended plots](#)
[New interactive sheet](#)

#Q1) Find out range of PL, SL, SW, PW columns for setosa, versicolor, virginica data frame.

```
species_groups = df.groupby('Species')
species_groups[['SepalLengthCm', 'SepalWidthCm', 'PetalLengthCm', 'PetalWidthCm']].apply(lambda x: x.max() - x.min())
```

	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm
Species				
Iris-setosa	1.5	2.1	0.9	0.5
Iris-versicolor	2.1	1.4	2.1	0.8
Iris-virginica	3.0	1.6	2.4	1.1

# Q2)Find 95 percentile values for PL, SW, SL, PW column in setosa versicolor, virginica data frame.

```
species_groups[['SepalLengthCm', 'SepalWidthCm', 'PetalLengthCm', 'PetalWidthCm']].quantile(0.95)
```

	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm
Species				
Iris-setosa	5.610	4.055	1.700	0.400
Iris-versicolor	6.755	3.200	4.900	1.600
Iris-virginica	7.700	3.510	6.655	2.455

#Q3)Find out Quartile values for PL, SW, SL, PW column in setosa versicolor, virginica data frame.

```
df.groupby('Species').quantile([0.25, 0.5, 0.75])
```

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm
Species					
Iris-setosa	0.25	13.25	4.800	3.125	1.400
	0.50	25.50	5.000	3.400	1.500
	0.75	37.75	5.200	3.675	1.575
Iris-versicolor	0.25	63.25	5.600	2.525	4.000
	0.50	75.50	5.900	2.800	4.350
	0.75	87.75	6.300	3.000	4.600
Iris-virginica	0.25	113.25	6.225	2.800	5.100
	0.50	125.50	6.500	3.000	5.550
	0.75	137.75	6.900	3.175	5.875

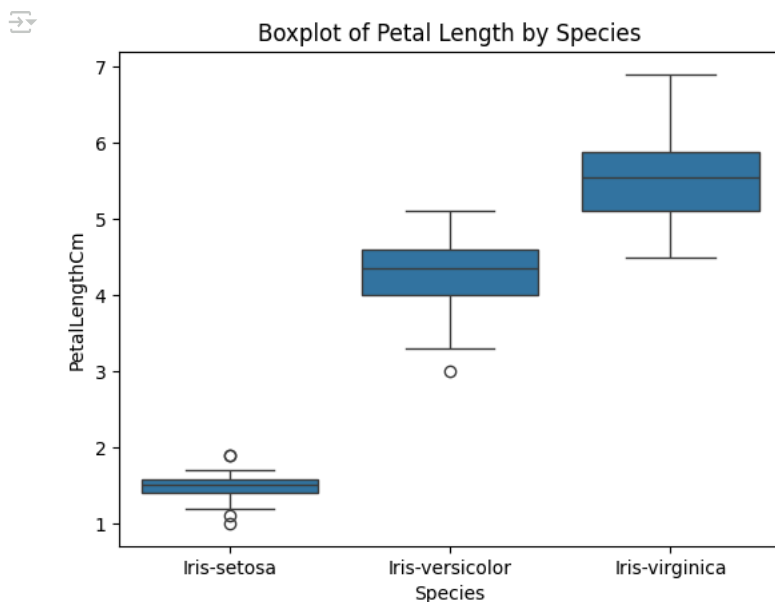
#Q4) Find out Interquartile Range for PL, SW, SL, PW column in setosa versicolor, virginica data frame.

```
species_groups[['SepalLengthCm', 'SepalWidthCm', 'PetalLengthCm', 'PetalWidthCm']].apply(lambda x: x.quantile(0.75) - x.quantile(0.25))
```

	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm
Species				
Iris-setosa	0.400	0.550	0.175	0.1
Iris-versicolor	0.700	0.475	0.600	0.3
Iris-virginica	0.675	0.375	0.775	0.5

#Q5) Plot box plot for PL column in setosa, versicolor, virginica data frame. Find out min and max values in the box plot.

```
sns.boxplot(x='Species', y='PetalLengthCm', data=df)
plt.title("Boxplot of Petal Length by Species")
plt.show()
print("Min and Max values from PetalLengthCm:")
print(df['PetalLengthCm'].agg(['min', 'max']))
```

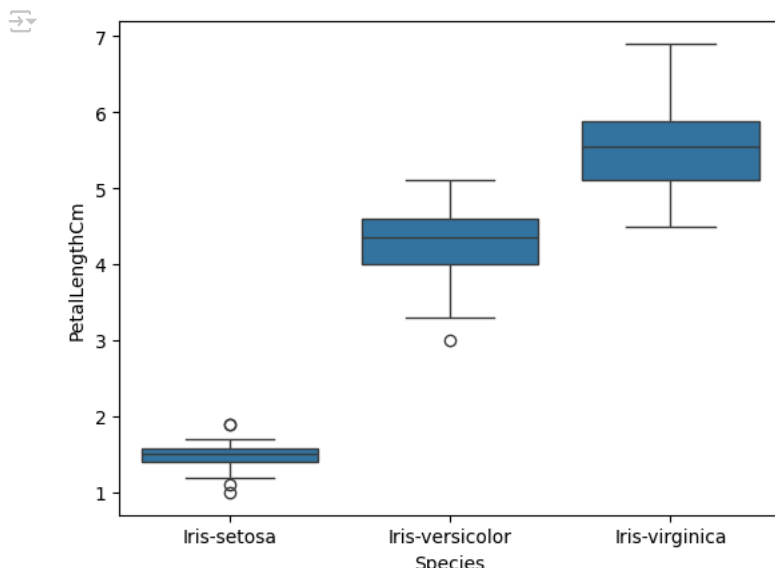


Min and Max values from PetalLengthCm:

```
min    1.0
max    6.9
Name: PetalLengthCm, dtype: float64
```

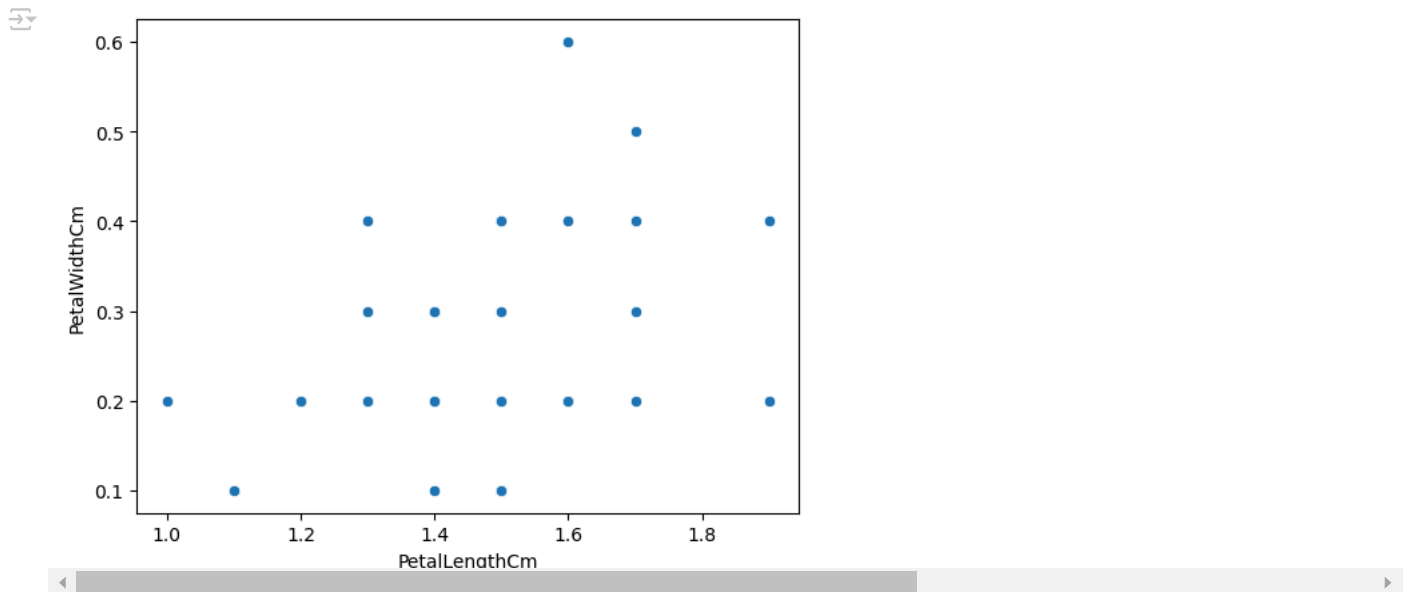
#Q6) Plot box plot using species and PL column in IRIS dataset. Find out min and max value in box plot.

```
sns.boxplot(x='Species', y='PetalLengthCm', data=df)
plt.show()
```

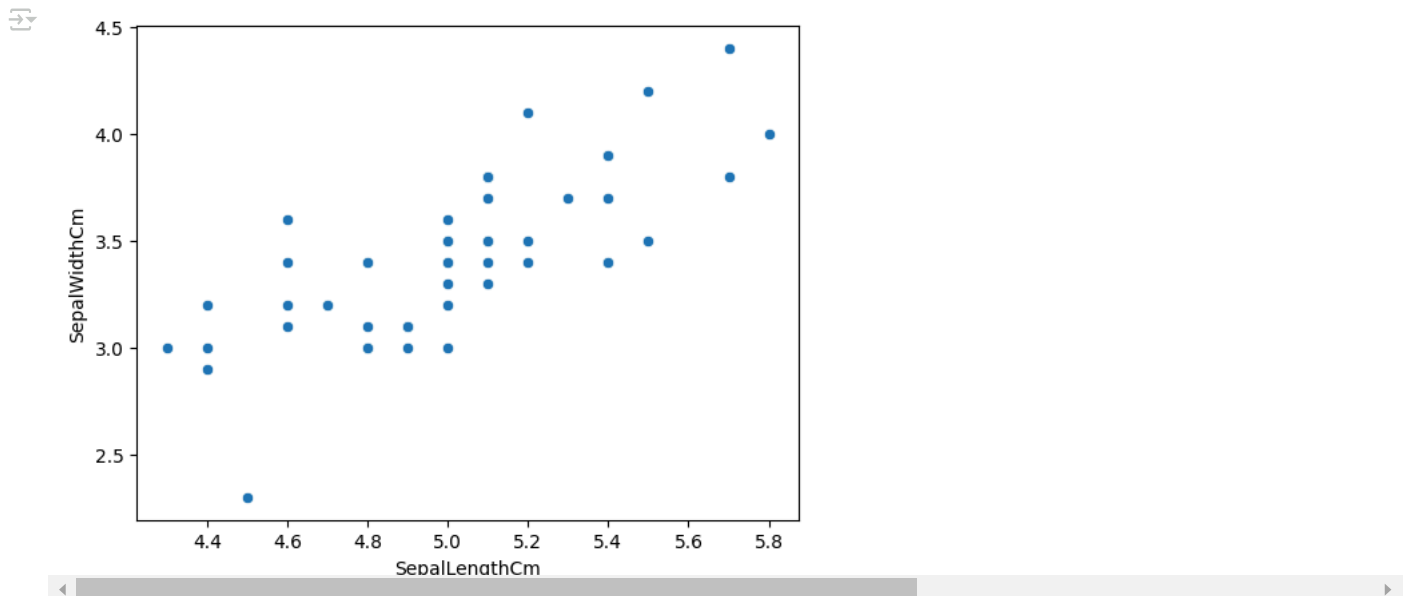


```
setosa = df[df["Species"] == "Iris-setosa"]  
versicolor = df[df["Species"] == "Iris-versicolor"]  
virginica = df[df["Species"] == "Iris-virginica"]
```

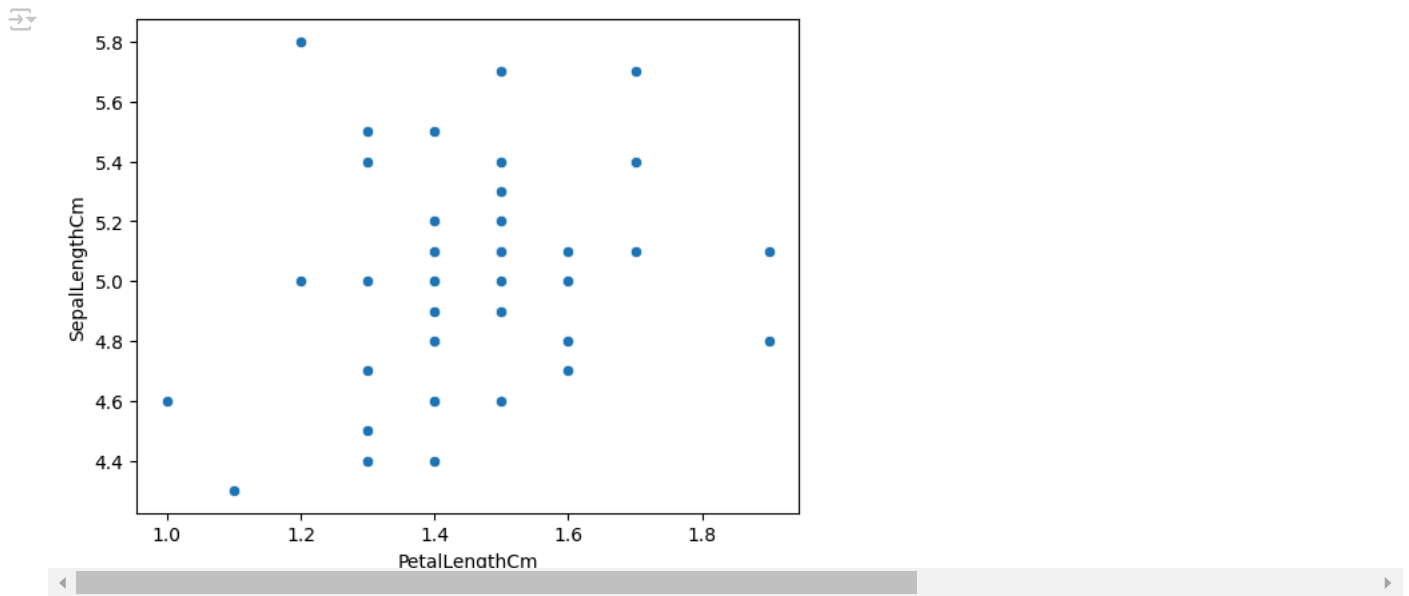
```
#Q7)Plot graph between PL and PW column in Iris_setosa dataframe.  
sns.scatterplot(x=setosa['PetalLengthCm'], y=setosa['PetalWidthCm'], data=df)  
plt.show()
```



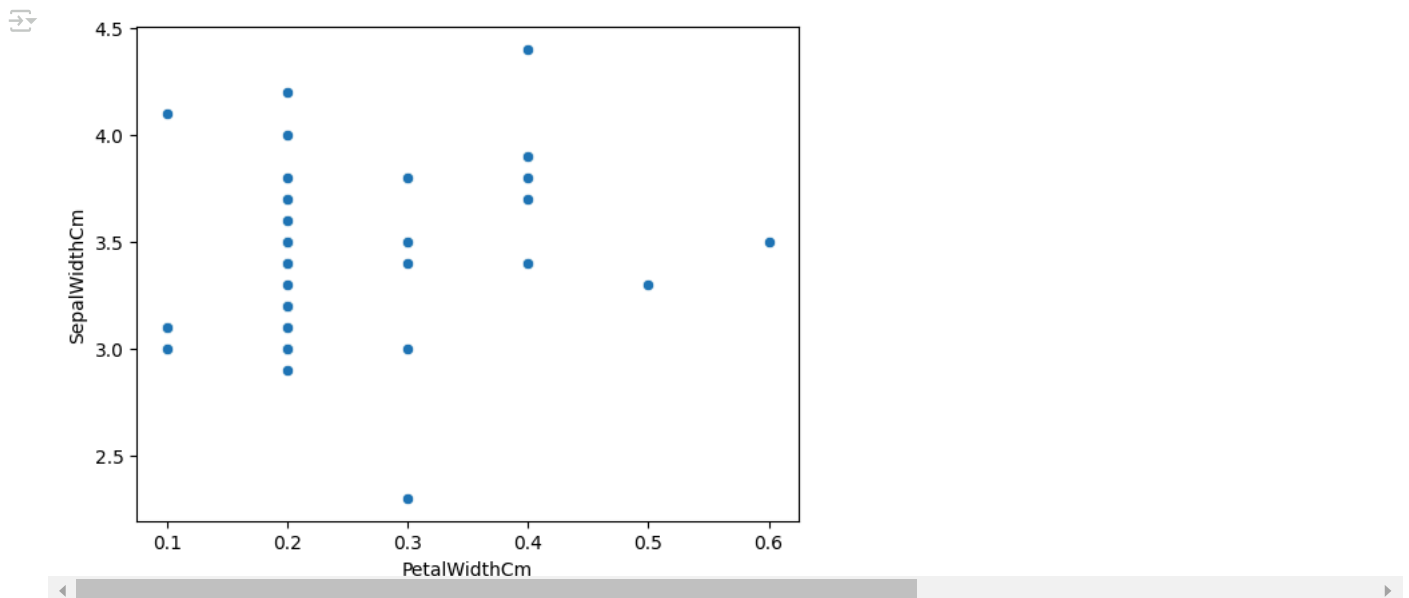
```
#Q8) Plot graph between SL and SW column in Iris_setosa dataframe.  
sns.scatterplot(x=setosa['SepalLengthCm'], y=setosa['SepalWidthCm'], data=df)  
plt.show()
```



```
#Q9)Plot graph between PL and SL column in Iris_setosa dataframe.  
sns.scatterplot(x=setosa['PetalLengthCm'], y=setosa['SepalLengthCm'], data=df)  
plt.show()
```



```
#Q10) Plot graph between PW and SW column in Iris_setosa dataframe.  
sns.scatterplot(x=setosa['PetalWidthCm'], y=setosa['SepalWidthCm'], data=df)  
plt.show()
```



```
#Q11) Find out covariances values for PL, SW, SL, PW column in setosa versicolor, virginica data frame.  
df.groupby('Species').cov()
```



Id SepalLengthCm SepalWidthCm PetalLengthCm PetalWidthCm



Species		Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm
Iris-setosa	Id	212.500000	-0.172449	-0.370408	0.134694	0.136735
	SepalLengthCm	-0.172449	0.124249	0.100298	0.016139	0.010547
	SepalWidthCm	-0.370408	0.100298	0.145180	0.011682	0.011437
	PetalLengthCm	0.134694	0.016139	0.011682	0.030106	0.005698
	PetalWidthCm	0.136735	0.010547	0.011437	0.005698	0.011494
Iris-versicolor	Id	212.500000	-2.024490	-0.374490	-1.297959	-0.486735
	SepalLengthCm	-2.024490	0.266433	0.085184	0.182898	0.055780
	SepalWidthCm	-0.374490	0.085184	0.098469	0.082653	0.041204
	PetalLengthCm	-1.297959	0.182898	0.082653	0.220816	0.073102
	PetalWidthCm	-0.486735	0.055780	0.041204	0.073102	0.039106
Iris-virginica	Id	212.500000	-0.116327	0.615306	-1.642857	0.145918
	SepalLengthCm	-0.116327	0.404343	0.093763	0.303290	0.049094
	SepalWidthCm	0.615306	0.093763	0.104004	0.071380	0.047629
	PetalLengthCm	-1.642857	0.303290	0.071380	0.304588	0.048824
	PetalWidthCm	0.145918	0.049094	0.047629	0.048824	0.075433

#Q12)Find out covariance between PL and PL columns in Iris\_setosa dataframe.Also Plot Graph.

```
print("Covariance between PetalLengthCm and itself:")
print(np.cov(df['PetalLengthCm'], df['PetalLengthCm']))
```

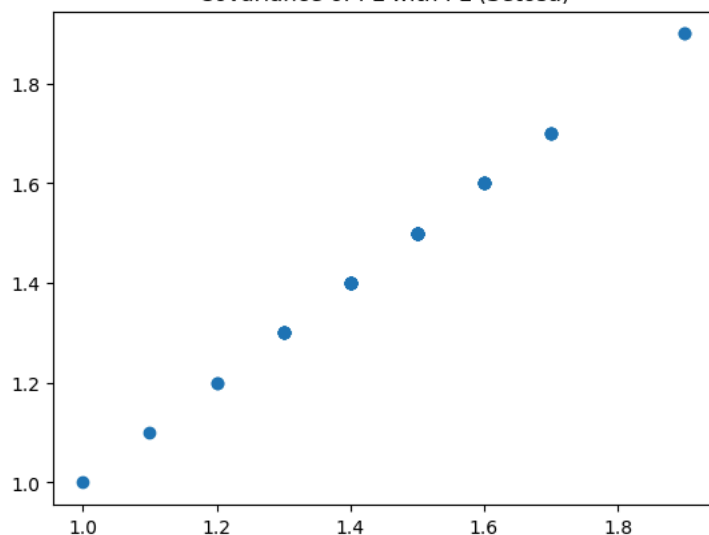
```
plt.scatter(setosa['PetalLengthCm'], setosa['PetalLengthCm'])
plt.title("Covariance of PL with PL (Setosa)")
plt.show()
```



Covariance between PetalLengthCm and itself:

```
[[3.11317942 3.11317942]
 [3.11317942 3.11317942]]
```

Covariance of PL with PL (Setosa)



#Q13)Scale the value of PL and PL column by 2. Find out covariance between PL and PL column in Iris\_setosa dataframe. Also Plot Graph.

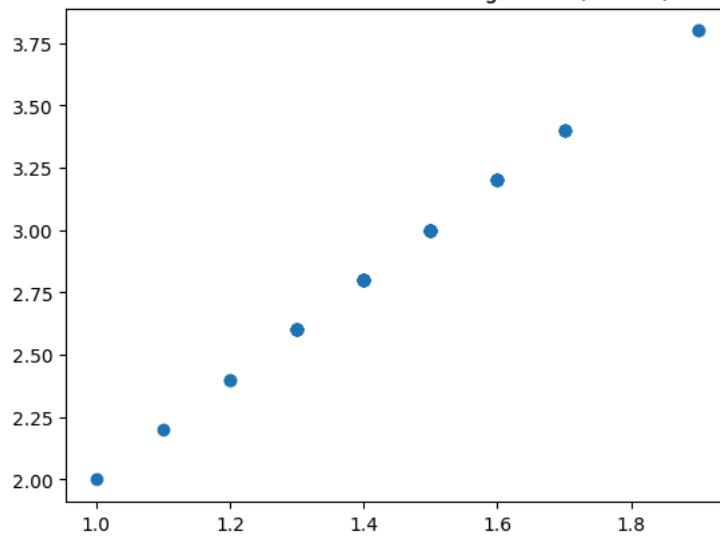
```
setosa['PL_scaled'] = setosa['PetalLengthCm'] * 2
cov_pl_scaled = setosa[['PetalLengthCm', 'PL_scaled']].cov()
print("Covariance after scaling PL by 2:", cov_pl_scaled)
plt.scatter(setosa['PetalLengthCm'], setosa['PL_scaled'])
plt.title("Covariance of Scaled PL with Original PL (Setosa)")
plt.show()
```

```
<ipython-input-17-c2002e218998>:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead
```

See the caveats in the documentation: [https://pandas.pydata.org/pandas-docs/stable/user\\_guide/indexing.html#returning-a-view-versus](https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus)

```
setosa['PL_scaled'] = setosa['PetalLengthCm'] * 2
Covariance after scaling PL by 2:
PetalLengthCm    0.030106    0.060212
PL_scaled        0.060212    0.120424
```

Covariance of Scaled PL with Original PL (Setosa)

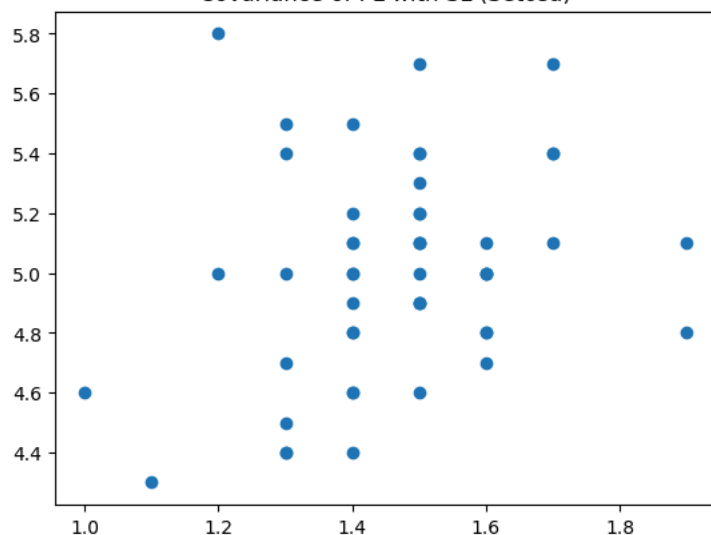


#Q14) Find out covariance between PL and SL columns in the Iris\_setosa dataframe. Also Plot Graph.

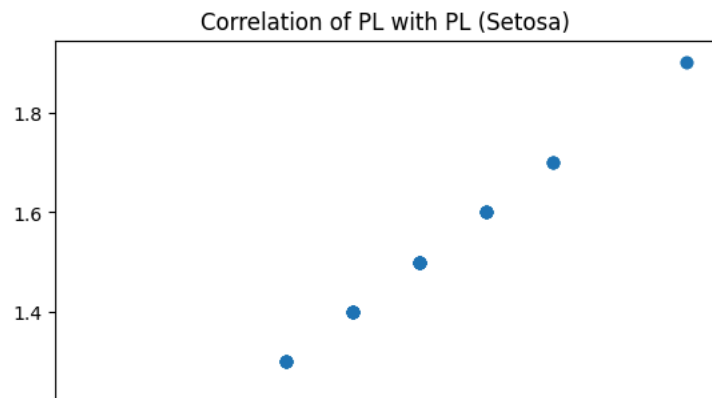
```
cov_pl_sl = setosa[['PetalLengthCm', 'SepalLengthCm']].cov()
print("Covariance between PL and SL:", cov_pl_sl)
plt.scatter(setosa['PetalLengthCm'], setosa['SepalLengthCm'])
plt.title("Covariance of PL with SL (Setosa)")
plt.show()
```

```
Covariance between PL and SL:
PetalLengthCm    0.030106    0.016139
SepalLengthCm    0.016139    0.124249
```

Covariance of PL with SL (Setosa)

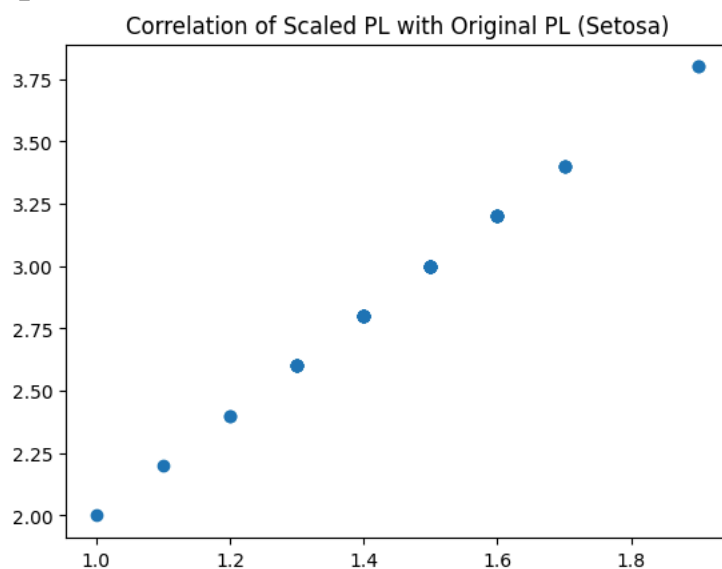


Correlation between PL and PL:  
PetalLengthCm 1.0



```
#Q16)Scale the value of PL and PL column by 2. Find correlation between PL and PL column in Iris_setosa dataframe. Also Plot Graph.
correlation_pl_scaled = setosa[['PetalLengthCm', 'PL_scaled']].corr()
print("Correlation after scaling PL by 2:", correlation_pl_scaled)
plt.scatter(setosa['PetalLengthCm'], setosa['PL_scaled'])
plt.title("Correlation of Scaled PL with Original PL (Setosa)")
plt.show()
```

Correlation after scaling PL by 2:  
PetalLengthCm 1.0 1.0  
PL\_scaled 1.0 1.0



```
#Q17)Find out correlation between PL and SL columns in Iris _setosa dataframe. Also Plot Graph.
correlation_pl_sl = setosa[['PetalLengthCm', 'SepalLengthCm']].corr()
print("Correlation between PL and SL:", correlation_pl_sl)
plt.scatter(setosa['PetalLengthCm'], setosa['SepalLengthCm'])
plt.title("Correlation of PL with SL (Setosa)")
plt.show()
```