```
## Name:Atharva Kangralkar
## Roll no : 54
## PRN:12311493
## CS-AIML-A
# Colab link : - https://colab.research.google.com/drive/1togzSAwKwzkI-TTWs0n2pjeYk7kg4JZu?usp=sharing


import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns


#Q1) Write features of Iris Dataset.
features = ['SepalLengthCm', 'SepalWidthCm', 'PetalLengthCm', 'PetalWidthCm']
print("Features of Iris Dataset:", features)
```

```
Features of Iris Dataset: ['SepalLengthCm', 'SepalWidthCm', 'PetalLengthCm', 'PetalWidthCm']
```

```
#Q2) Read Iris csv data file and assign as df data frame.
df = pd.read_csv("Iris.csv")


df
```

	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

Next steps: Generate code with df | View recommended plots | New interactive sheet

```
#Q3) Show preview of df data frame.
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

```
# Q4) Rename columns of Iris dataset as SL, PL, SW, PW.
df=df.rename(columns={'SepalLengthCm':'SL'})
df=df.rename(columns={'SepalWidthCm':'SW'})
df=df.rename(columns={'PetalLengthCm':'PL'})
df=df.rename(columns={'PetalWidthCm':'PW'})


df
```

	Id	SL	SW	PL	PW	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

Next steps: Generate code with df | View recommended plots | New interactive sheet

```
#Q5) Find out the number of setosa, versicolor, virginica flowers from the dataset.
df["Species"].value_counts()
```

Species	count
Iris-setosa	50
Iris-versicolor	50
Iris-virginica	50

dtype: int64

```
#Q6) Find a separate data frame for setosa, versicolor and virginica flowers data frame from Iris dataset.
setosa_df = df[df["Species"] == "Iris-setosa"]
versicolor_df = df[df["Species"] == "Iris-versicolor"]
virginica_df = df[df["Species"] == "Iris-virginica"]


versicolor_df.head()
```

	Id	SL	SW	PL	PW	Species
50	51	7.0	3.2	4.7	1.4	Iris-versicolor
51	52	6.4	3.2	4.5	1.5	Iris-versicolor
52	53	6.9	3.1	4.9	1.5	Iris-versicolor
53	54	5.5	2.3	4.0	1.3	Iris-versicolor
54	55	6.5	2.8	4.6	1.5	Iris-versicolor

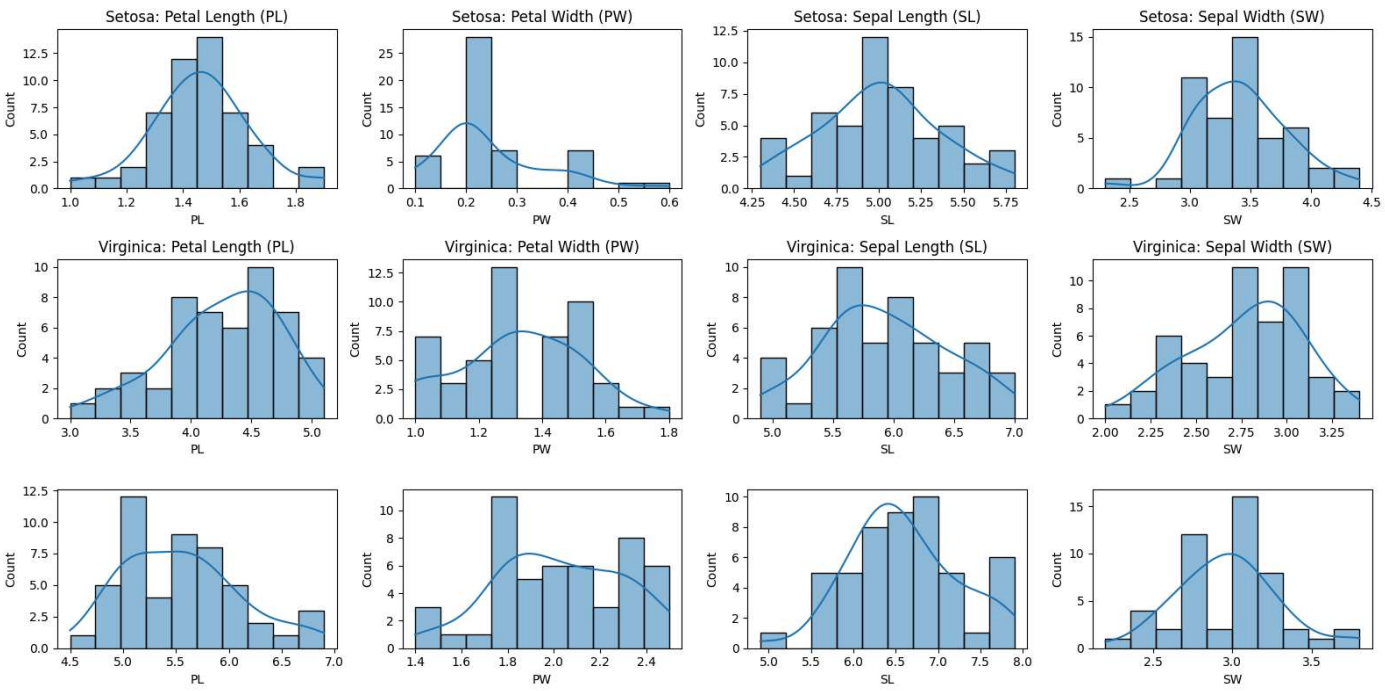
Next steps: Generate code with versicolor_df | View recommended plots | New interactive sheet

```
#Q7) Find out mean for PL, SW, PW, SL columns in setosa, versicolor and virginica data frame. Also write conclusions from  mean result.
df.groupby("Species")[["PL", "SW", "PW", "SL"]].mean()
## setosa has the shortest petal length and width on average,versicolor has longer and wider petals and virginica has the longest and widest
## sepal length is similar for setosa,versicolor and virginica.With virginica having longest sepal length
## sepal length is similar for setosa,versicolor and virginica.With setosa having longest sepal length
```

Species	PL	SW	PW	SL
Iris-setosa	1.464	3.418	0.244	5.006
Iris-versicolor	4.260	2.770	1.326	5.936
Iris-virginica	5.552	2.974	2.026	6.588

Double-click (or enter) to edit

```
#Q8) Plot graphs based on PL, SW, PW, SL columns in setosa, versicolor and virginica data frame.

# Create a 2x4 grid of subplots (2 rows, 4 columns)
fig, axes = plt.subplots(3, 4, figsize=(16, 8))

# Set the titles for each subplot
axes[0, 0].set_title("Setosa: Petal Length (PL)")
axes[0, 1].set_title("Setosa: Petal Width (PW)")
axes[0, 2].set_title("Setosa: Sepal Length (SL)")
axes[0, 3].set_title("Setosa: Sepal Width (SW)")

axes[1, 0].set_title("Versicolor: Petal Length (PL)")
axes[1, 1].set_title("Versicolor: Petal Width (PW)")
axes[1, 2].set_title("Versicolor: Sepal Length (SL)")
axes[1, 3].set_title("Versicolor: Sepal Width (SW)")

axes[1, 0].set_title("Virginica: Petal Length (PL)")
axes[1, 1].set_title("Virginica: Petal Width (PW)")
axes[1, 2].set_title("Virginica: Sepal Length (SL)")
axes[1, 3].set_title("Virginica: Sepal Width (SW)")

# Plot histograms for Setosa
sns.histplot(setosa_df['PL'], bins=10, kde=True, ax=axes[0, 0])
sns.histplot(setosa_df['PW'], bins=10, kde=True, ax=axes[0, 1])
sns.histplot(setosa_df['SL'], bins=10, kde=True, ax=axes[0, 2])
sns.histplot(setosa_df['SW'], bins=10, kde=True, ax=axes[0, 3])

# Plot histograms for Versicolor
sns.histplot(versicolor_df['PL'], bins=10, kde=True, ax=axes[1, 0])
sns.histplot(versicolor_df['PW'], bins=10, kde=True, ax=axes[1, 1])
sns.histplot(versicolor_df['SL'], bins=10, kde=True, ax=axes[1, 2])
sns.histplot(versicolor_df['SW'], bins=10, kde=True, ax=axes[1, 3])

# Plot histograms for Virginica
sns.histplot(virginica_df['PL'], bins=10, kde=True, ax=axes[2, 0])
sns.histplot(virginica_df['PW'], bins=10, kde=True, ax=axes[2, 1])
sns.histplot(virginica_df['SL'], bins=10, kde=True, ax=axes[2, 2])
sns.histplot(virginica_df['SW'], bins=10, kde=True, ax=axes[2, 3])

# Adjust layout for better spacing
plt.tight_layout()
plt.show()
```




```
#Q9) Add 100 values in PL, SW columns and observe the result. Also comment on the result
modified_df = df.copy()
modified_df[['PL', 'SW']] += 100
# modified_df.head()
modified_df.groupby("Species")[["PL", "SW", "PW", "SL"]].mean()
## Adding a constant value 100 shifts the mean of PL and SW
## but does not affect other the difference among setosa,versicolor and virginica for PL,SW features ie the affected features.
```

Species	PL	SW	PW	SL
Iris-setosa	101.464	103.418	0.244	5.006
Iris-versicolor	104.260	102.770	1.326	5.936
Iris-virginica	105.552	102.974	2.026	6.588

```
#Q10) Find out median for PL, SW, PW, SL columns in setosa, versicolor and virginica data frame. Also write conclusions from median result.
df.groupby('Species')[['PL', 'SW', 'PW', 'SL']].median()
## The median values are close to the mean values, suggesting that the data distribution for these features is fairly symmetric.
## Petal Length increases progressively from Setosa to Virginica
## Sepal Width is highest in Setosa and lowest in Versicolor
## Petal Width increases significantly from Setosa to Virginica
## Sepal Length increases gradually from Setosa to Virginica but has less variation compared to petal-related features.
```

Species	PL	SW	PW	SL
Iris-setosa	1.50	3.4	0.2	5.0
Iris-versicolor	4.35	2.8	1.3	5.9
Iris-virginica	5.55	3.0	2.0	6.5

```
#Q11)Find out mode for PL, SW, PW, SL columns in setosa, versicolor and virginica data frame. Also write conclusions from the median result.
from scipy import stats

# Calculate mode for each species
mode_setosa = stats.mode(setosa_df[['SL', 'SW', 'PL', 'PW']], axis=0).mode
```

```
mode_versicolor = stats.mode(versicolor_df[['SL', 'SW', 'PL', 'PW']], axis=0).mode
mode_virginica = stats.mode(virginica_df[['SL', 'SW', 'PL', 'PW']], axis=0).mode

print("Setosa Mode:\n", mode_setosa)
print("Versicolor Mode:\n", mode_versicolor)
print("Virginica Mode:\n", mode_virginica)

##Iris-setosa has a mode close to its median for all features, indicating its values are mostly concentrated around those central values.
# Iris-versicolor has a mode that is slightly different from the median, suggesting a bit more spread in the data.
# Iris-virginica shows a mode higher than its median for Petal Length and Petal Width, indicating a tendency for larger values in these featu
```

```
Setosa Mode:
 [5.  3.4 1.5 0.2]
Versicolor Mode:
 [5.5 3.  4.5 1.3]
Virginica Mode:
 [6.3 3.  5.1 1.8]
```

```
#Q12) Find out the range of PL,SW,PW,SL columns in the data frame.
range_PL = df['PL'].max() - df['PL'].min()
range_SW = df['SW'].max() - df['SW'].min()
range_PW = df['PW'].max() - df['PW'].min()
range_SL = df['SL'].max() - df['SL'].min()

print("Range of PL:", range_PL)
print("Range of SW:", range_SW)
print("Range of PW:", range_PW)
print("Range of SL:", range_SL)
```

```
Range of PL: 5.9
Range of SW: 2.4000000000000004
Range of PW: 2.4
Range of SL: 3.6000000000000005
```

```
#Q13) Find out variance for PL, SW, PW, SL columns in setosa, versicolor and virginica data frame. Also write conclusions from  variance res
variance_values = df.groupby('Species')[['PL', 'SW', 'PW', 'SL']].var()
variance_values
## Iris-virginica has the most variation in petal length, while Iris-setosa has the most uniform petal lengths.
## Sepal width varies the most in Iris-setosa and the least in Iris-versicolor, showing that Versicolor's sepal width is more stable.
## Iris-setosa has very stable petal widths, whereas Iris-virginica has a much wider spread in petal width measurements.
## Sepal length is most consistent in Iris-setosa and varies the most in Iris-virginica, indicating that Virginica specimens have more diver
```

Species	PL	SW	PW	SL
Iris-setosa	0.030106	0.145180	0.011494	0.124249
Iris-versicolor	0.220816	0.098469	0.039106	0.266433
Iris-virginica	0.304588	0.104004	0.075433	0.404343

Next steps: Generate code with variance_values | View recommended plots | New interactive sheet

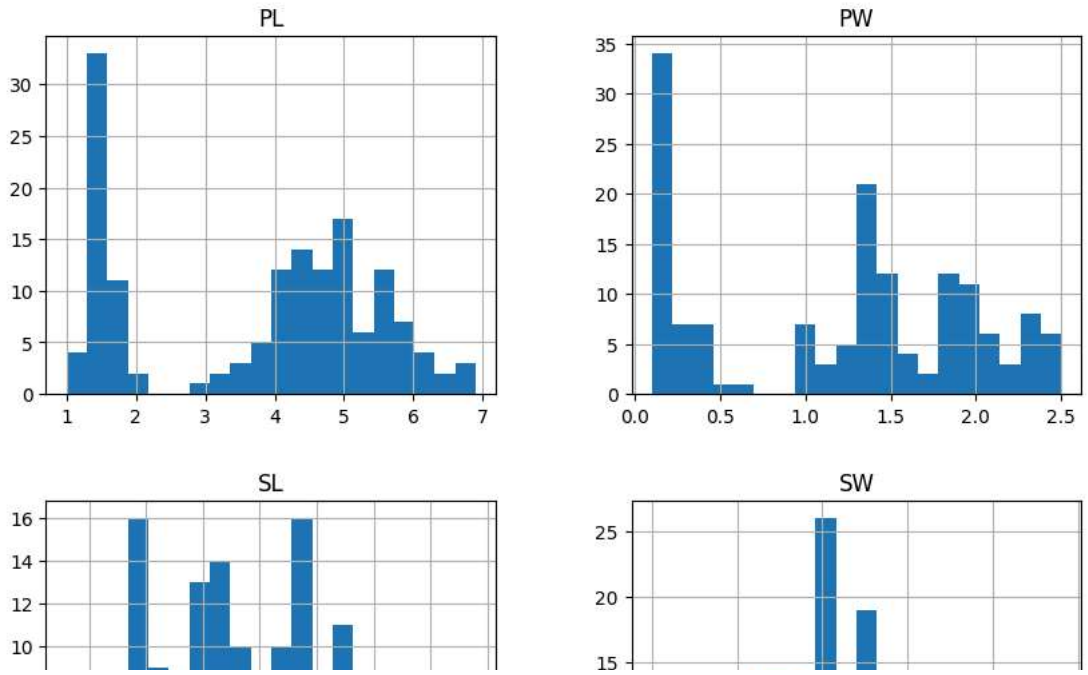
```
#Q14)Find out standard deviation for PL, SW, PW, SL columns in setosa, versicolor and virginica data frame. Also write conclusions from stan
std_values = df.groupby('Species')[['PL', 'SW', 'PW', 'SL']].std()
std_values
## Iris-virginica has the most diverse petal lengths, while Iris-setosa has the most uniform petal length.
## Sepal Width varies the most in Iris-setosa and the least in Iris-versicolor, showing that Versicolor has the most stable sepal width.
## Iris-setosa has very stable petal widths, while Iris-virginica has the most variation in petal width measurements.
## Sepal length is most consistent in Iris-setosa and varies the most in Iris-virginica, showing that Virginica specimens have more diversit
```

Species	PL	SW	PW	SL
Iris-setosa	0.173511	0.381024	0.107210	0.352490
Iris-versicolor	0.469911	0.313798	0.197753	0.516171
Iris-virginica	0.551895	0.322497	0.274650	0.635880

Next steps: Generate code with std_values | View recommended plots | New interactive sheet

```
#Q15) Perform univariate analysis on PL,PW,SL,SW ,Species column from df Iris dataset.
```

```
# Univariate analysis using histograms
df.hist(column=['PL', 'PW', 'SL', 'SW'], bins=20, figsize=(10, 8))
plt.suptitle('Univariate Analysis of Iris Dataset')
plt.show()

# Univariate analysis using boxplots
plt.figure(figsize=(10, 8))
sns.boxplot(data=df[['PL', 'PW', 'SL', 'SW']])
plt.title('Boxplot of Iris Dataset')
plt.show()

# Univariate analysis for Species (categorical data)
sns.countplot(data=df, x='Species')
plt.title('Count of Each Species in Iris Dataset')
plt.show()
```


Univariate Analysis of Iris Dataset
PL
PW
SL
SW