

Data Science & Machine Learning: Lab Cycle 3

Matplotlib

Demonstrate creating various types of charts and plots using functions in matplotlib library

1. Sarah bought a new car in 2001 for \$24,000. The dollar value of her car changed each year as shown in the table below.

Value of Sarah's Car

Year	Value
2001	\$24,000
2002	\$22,500
2003	\$19,700
2004	\$17,500
2005	\$14,500
2006	\$10,000
2007	\$ 5,800

Represent the following information using a line graph with following style properties

- **X- axis - Year**
- **Y –axis - Car Value**
- **title –Value Depreciation (left Aligned)**
- **Line Style dash dot and Line-color should be red**
- **point using * symbol with green color and size 20**

Subplot() provides multiple plots in one figure.

CODE:

```
import matplotlib.pyplot as plt

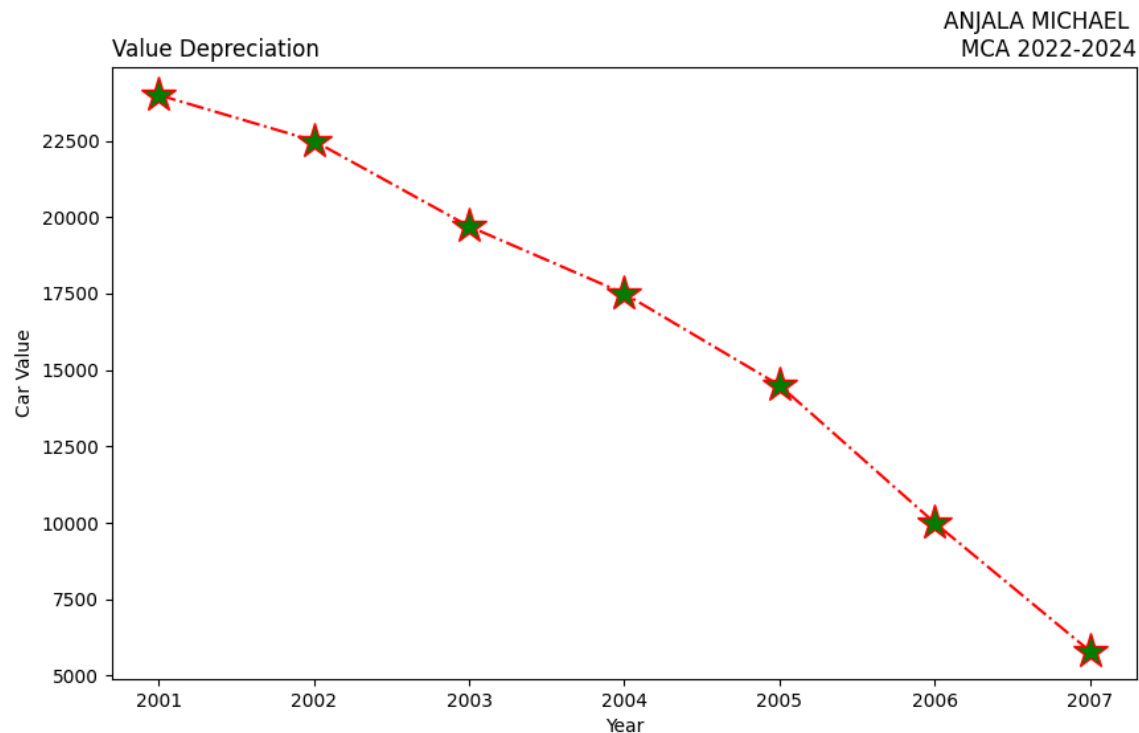
# Data
years = [2001, 2002, 2003, 2004, 2005, 2006, 2007]
car_values = [24000, 22500, 19700, 17500, 14500, 10000, 5800]

# Create a new figure
plt.figure(figsize=(10, 6))

# Create a subplot with the specified properties
plt.subplot(111) # Only one subplot in this example
plt.plot(years, car_values, linestyle='-.', color='red', marker='*',
         markersize=20, markerfacecolor='green')
plt.title("ANJALA MICHAEL \nMCA 2022-2024", loc="right")
plt.title("Value Depreciation", loc="left")
plt.xlabel("Year")
plt.ylabel("Car Value")

# Show the plot
plt.show()
```

OUTPUT:



2. Following table gives the daily sales of the following items in a shop

Day	Mon	Tues	Wed	Thurs	Fri
Drinks	300	450	150	400	650
Food	400	500	350	300	500

Use subplot function to draw the line graphs with **grids(color as blue and line style dotted)** for the above information as 2 separate graphs in two rows

a) Properties for the Graph 1:

- X label- Days of week
- Y label-Sale of Drinks
- Title-Sales Data1 (right aligned)
- Line –dotted with cyan color
- Points- hexagon shape with color magenta and outline black

b) Properties for the Graph 2:

- X label- Days of Week
- Y label-Sale of Food
- Title-Sales Data2 (center aligned)
- Line –dashed with yellow color
- Points- diamond shape with color green and outline red

CODE:

```
import matplotlib.pyplot as plt
# Data
days = ['Mon', 'Tues', 'Wed', 'Thurs', 'Fri']
drinks_sales = [300, 450, 150, 400, 650]
food_sales = [400, 500, 350, 300, 500]

# Create subplots with two rows
fig, axs = plt.subplots(2, 1, figsize=(8, 8))

# Graph 1: Drinks
axs[0].plot(days, drinks_sales, linestyle='--', color='cyan', marker='H',
            markersize=8, markerfacecolor='magenta', markeredgecolor='black')
axs[0].set_xlabel('Days of Week')
axs[0].set_ylabel('Sale of Drinks')
axs[0].set_title('Sales Data1', loc='right')
axs[0].set_title('ANJALA MICHAEL\nMCA 2022-2023', loc='left')
axs[0].grid(True, color='blue', linestyle='dotted')

# Graph 2: Food
axs[1].plot(days, food_sales, linestyle='-', color='yellow', marker='D',
            markersize=8, markerfacecolor='green', markeredgecolor='red')
axs[1].set_xlabel('Days of Week')
axs[1].set_ylabel('Sale of Food')
```

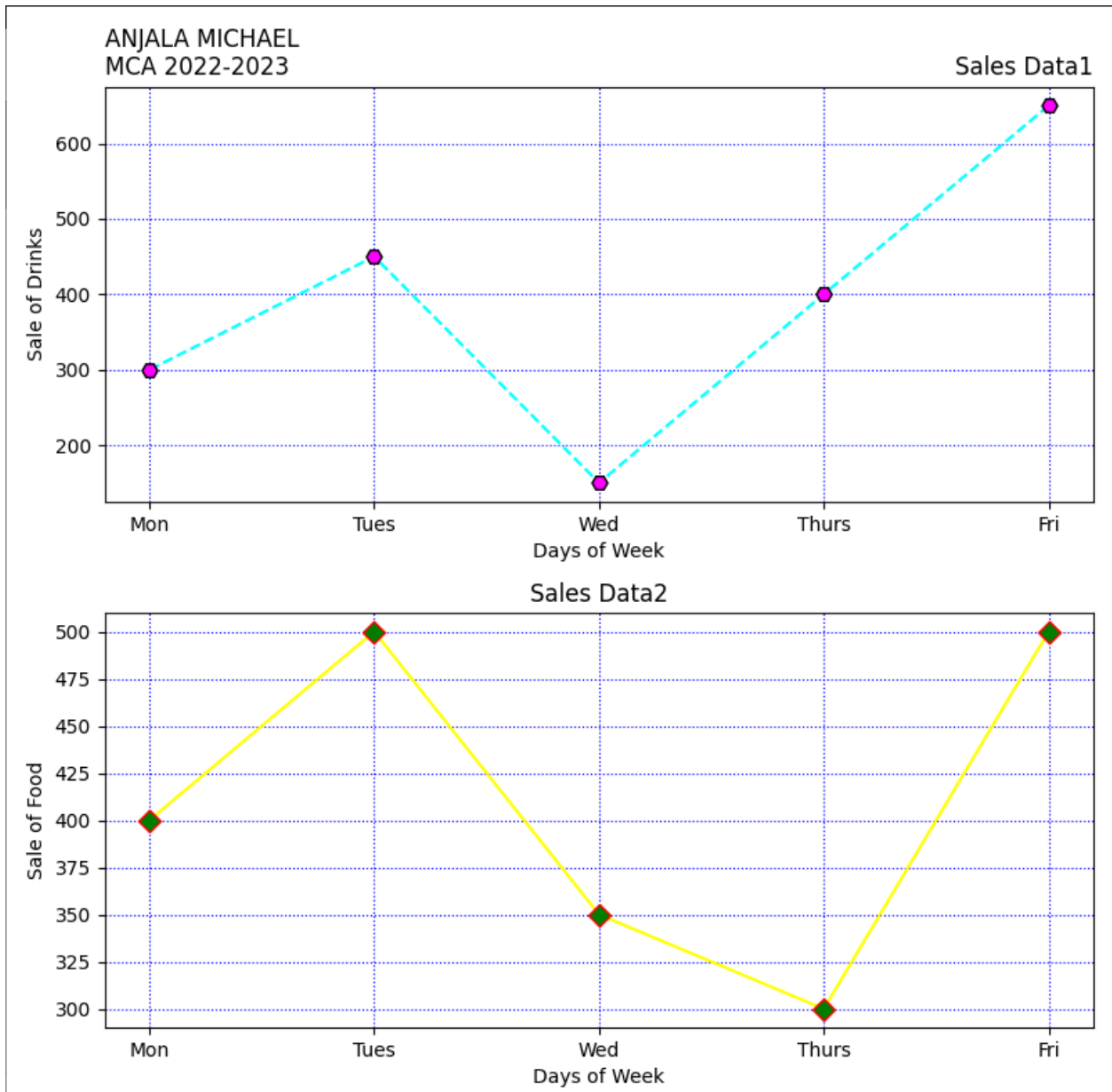
```

axs[1].set_title('Sales Data2', loc='center')
axs[1].grid(True, color='blue', linestyle='dotted')

# Adjust layout
plt.tight_layout()
# Display the plots
plt.show()

```

OUTPUT:



3. Create scatter plot for the below data:(use Scatter function)

Product	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
Affordable Segment	173	153	195	147	120	144	148	109	174	130	172	131
Luxury Segment	189	189	105	112	173	109	151	197	174	145	177	161
Super Luxury Segment	185	185	126	134	196	153	112	133	200	145	167	110

Create scatter plot for each Segment with following properties within one graph

- X Label- Months of Year with font size 18
- Y-Label- Sales of Segments
- Title –Sales Data
- Color for Affordable segment- pink
- Color for Luxury Segment- Yellow
- Color for Super luxury segment-blue

CODE:

```
import matplotlib.pyplot as plt
import numpy as np
month =
np.array(['Jan', 'Feb', 'Mar', 'Apr', 'May', 'Jun', 'Jul', 'Aug', 'Sep', 'Oct', 'Nov', 'D
ec'])
AS = np.array([173,153,195,147,120,144,148,109,174,130,172,131])
LS = np.array([189,189,105,112,173,109,151,197,174,145,177,161])
SLS = np.array([185,185,126,134,196,153,112,133,200,145,167,110])

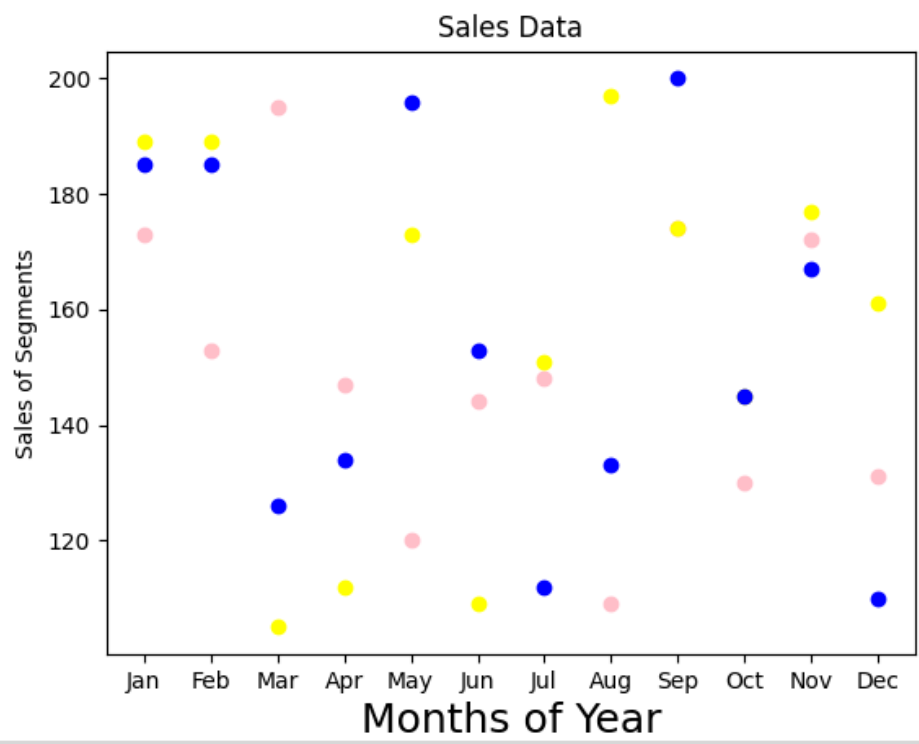
plt.xlabel('Months of Year', fontsize=18)
plt.ylabel('Sales of Segments')

plt.title('Sales Data')
plt.title('ANJALA MICHAEL\nMCA 2022-2024', loc='right')

plt.scatter(month,AS, label='Affordable Segment', color='pink')
plt.scatter(month,LS, label='Luxury Segment', color='yellow')
plt.scatter(month,SLS, label='Super Luxury Segment', color='blue')

plt.show()
```

OUTPUT:



4. Display the above data using multiline plot(3 different lines in same graph)
- Display the description of the graph in upper right corner(use **legend()**)
 - Use different colors and line styles for 3 different lines

CODE:

```
import matplotlib.pyplot as plt
import numpy as np
month =
np.array(['Jan', 'Feb', 'Mar', 'Apr', 'May', 'Jun', 'Jul', 'Aug', 'Sep', 'Oct', 'Nov', 'Dec'])
AS = np.array([173,153,195,147,120,144,148,109,174,130,172,131])
LS = np.array([189,189,105,112,173,109,151,197,174,145,177,161])
SLS = np.array([185,185,126,134,196,153,112,133,200,145,167,110])

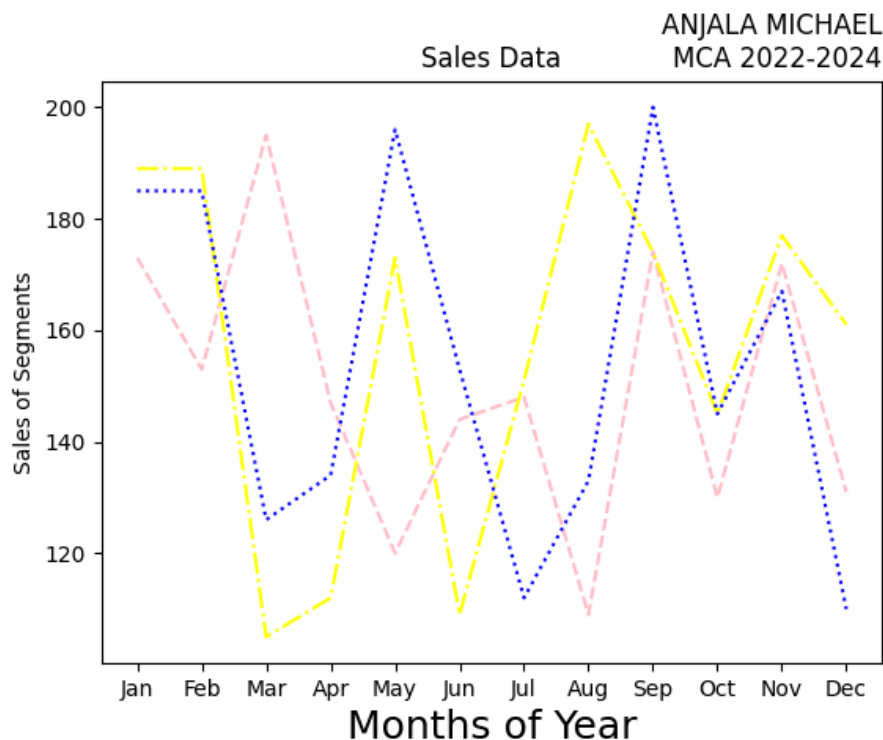
plt.plot(month,AS, label='Affordable', color='pink',linestyle='--')
plt.plot(month,LS, label='Luxury', color='yellow',linestyle='-.')
plt.plot(month,SLS, label='Super Luxury', color='blue',linestyle=':')

plt.xlabel('Months of Year', fontsize=18)
plt.ylabel('Sales of Segments')

plt.title('Sales Data')
plt.title('ANJALA MICHAEL\nMCA 2022-2024', loc='right')

plt.show()
```

OUTPUT:



5. 100 students were asked what their primary mode of transport for getting to school was. The results of this survey are recorded in the table below. Construct a bar graph representing this information.

Mode of transport	Frequency
Walking	29
Cycling	15
Car	35
Bus	18
Train	3

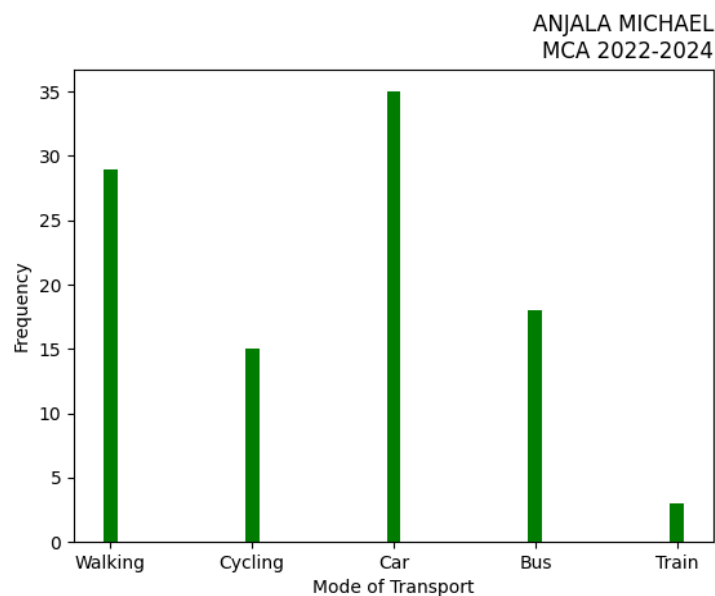
Create a bar graph with

- X axis -mode of Transport and Y axis 'frequency'
- Provide appropriate labels and title
- Width .1, color green

CODE:

```
import matplotlib.pyplot as plt
import numpy as np
mode_transport = np.array(['Walking', 'Cycling', 'Car', 'Bus', 'Train'])
freq = np.array([29, 15, 35, 18, 3])
plt.xlabel('Mode of Transport')
plt.ylabel('Frequency')
plt.title('ANJALA MICHAEL\nMCA 2022-2024', loc='right')
plt.bar(mode_transport, freq, width=0.1, color='green')
plt.show()
```

OUTPUT:



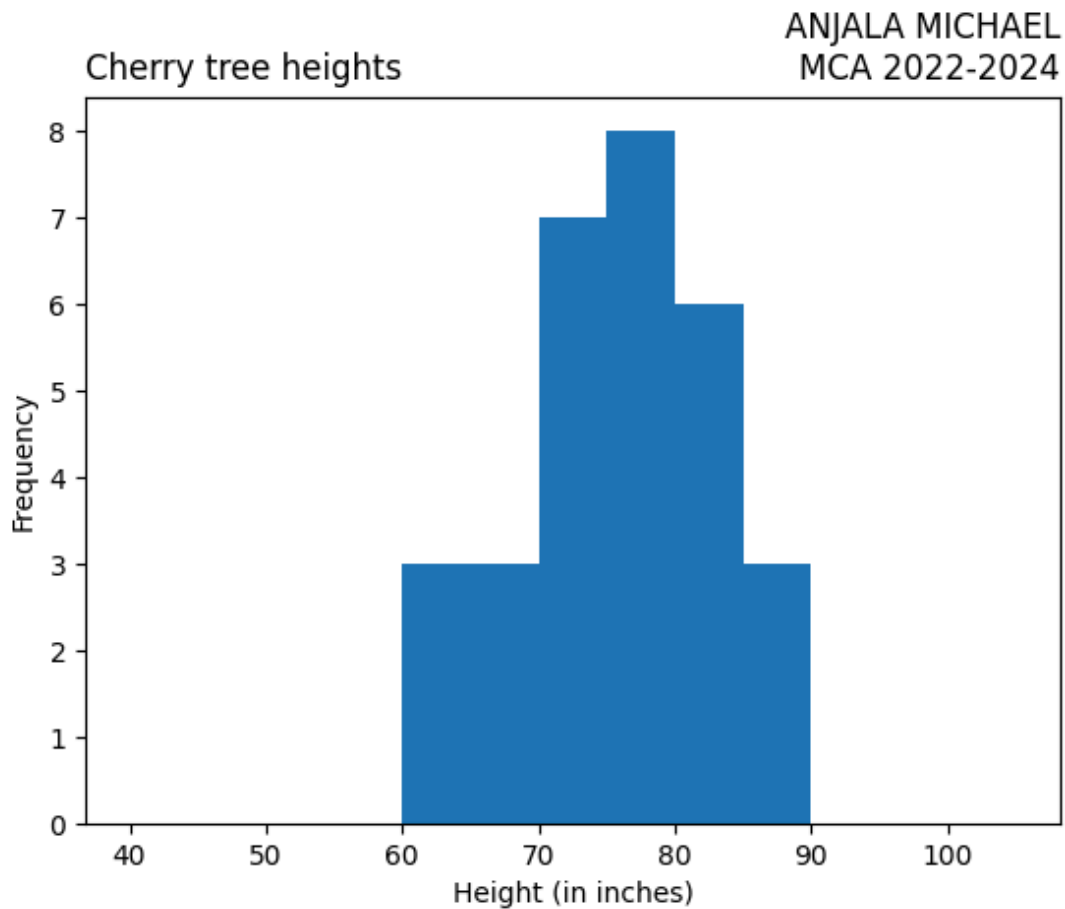
6. We are provided with the height of 30 cherry trees. The height of the trees (in inches): 61, 63, 64, 66, 68, 69, 71, 71.5, 72, 72.5, 73, 73.5, 74, 74.5, 76, 76.2, 76.5, 77, 77.5, 78, 78.5, 79, 79.2, 80, 81, 82, 83, 84, 85, 87. Create a histogram with a bin size of 5

CODE:

```
import matplotlib.pyplot as plt
import numpy as np
x = np.random.normal([61, 63, 64, 66, 68, 69, 71, 71.5, 72, 72.5, 73, 73.5, 74, 74.5, 76, 76.2, 76.5, 77, 77.5, 78, 78.5, 79, 79.2, 80, 81, 82, 83, 84, 85, 87])

plt.hist(x, bins=range(40,110,5), )
plt.title('Cherry tree heights',loc='left')
plt.title('ANJALA MICHAEL\nMCA 2022-2024', loc='right')
plt.xlabel('Height (in inches)')
plt.ylabel('Frequency')
plt.show()
```

OUTPUT:



7. Using the pandas function read_csv(), read the given 'iris' data set.
- Display Shape of the data set.
 - First 5 and last five rows of data set(head and tail).
 - Size of dataset.
 - No. of samples available for each variety.
 - Description of the data set(use describe).

CODE:

```
print("SJC22MCA-2007 : ANJALA MICHAEL")
print("Batch : MCA 2022-24")
import pandas as pd
df = pd.read_csv('iris.csv')
print("Shape of the dataset is : ",df.shape)
print("First 5 and last five rows of data set\n",df)
print("Size of dataset : ",df.size)
print("No. of samples available for each variety\n",df.count())
print("Description of the data set\n",df.describe())
```

OUTPUT:

SJC22MCA-2007 : ANJALA MICHAEL

Batch : MCA 2022-24

Shape of the dataset is : (150, 5)

First 5 and last five rows of data set

	sepal.length	sepal.width	petal.length	petal.width	variety
0	5.1	3.5	1.4	0.2	Setosa
1	4.9	3.0	1.4	0.2	Setosa
2	4.7	3.2	1.3	0.2	Setosa
3	4.6	3.1	1.5	0.2	Setosa
4	5.0	3.6	1.4	0.2	Setosa
..
145	6.7	3.0	5.2	2.3	Virginica
146	6.3	2.5	5.0	1.9	Virginica
147	6.5	3.0	5.2	2.0	Virginica
148	6.2	3.4	5.4	2.3	Virginica
149	5.9	3.0	5.1	1.8	Virginica

[150 rows x 5 columns]

Size of dataset : 750

No. of samples available for each variety

```
sepal.length    150
sepal.width     150
petal.length    150
petal.width     150
variety         150
dtype: int64
```

Description of the data set

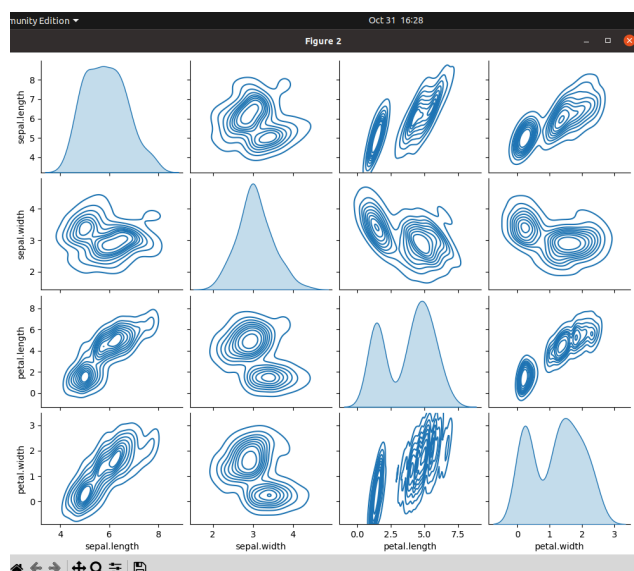
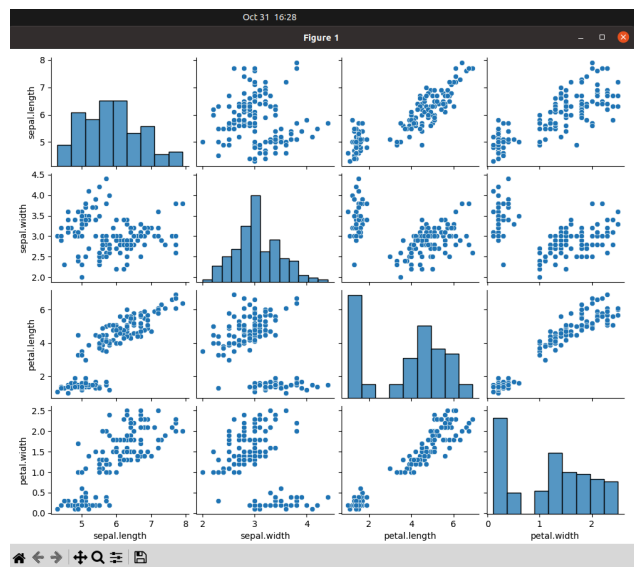
	sepal.length	sepal.width	petal.length	petal.width
count	150.000000	150.000000	150.000000	150.000000
mean	5.843333	3.057333	3.758000	1.199333
std	0.828066	0.435866	1.765298	0.762238
min	4.300000	2.000000	1.000000	0.100000
25%	5.100000	2.800000	1.600000	0.300000
50%	5.800000	3.000000	4.350000	1.300000
75%	6.400000	3.300000	5.100000	1.800000
max	7.900000	4.400000	6.900000	2.500000

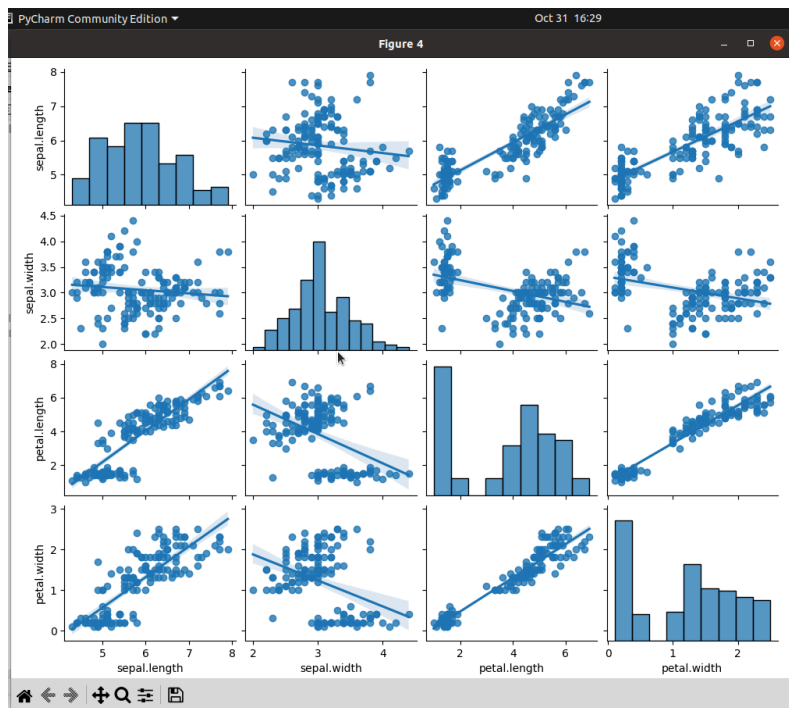
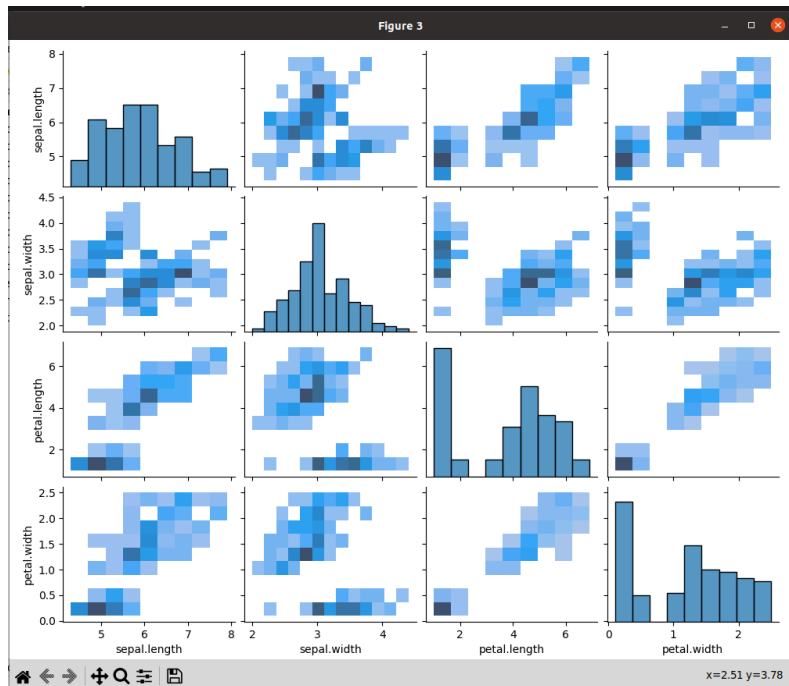
8. Use the `pairplot()` function in seaborn to display pairwise relationships between attributes. Try different kind of plots {'scatter', 'kde', 'hist', 'reg'} and different kind of markers.

CODE:

```
import pandas as pd
import seaborn
import matplotlib.pyplot as plt
df = pd.read_csv('iris.csv')
seaborn.pairplot(df, kind='scatter')
seaborn.pairplot(df, kind='kde')
seaborn.pairplot(df, kind='hist')
seaborn.pairplot(df, kind='reg')
# to show
plt.show()
```

OUTPUT:





9. Using the iris data set, get familiarized with functions:

- 1) `displot()`
- 2) `histplot()`
- 3) `relplot()`

Note: import pandas and seaborn packages

CODE:

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

iris=pd.read_csv("iris.csv")
sns.displot(iris['sepal.length'],kde=True,rug=True)
plt.title("Distribution of Sepal length")
plt.show()

sns.histplot(iris['petal.width'],kde=True,bins=20)
plt.title("Histogram of Petal width")
plt.show()

sns.relplot(x="sepal.length",y="sepal.width",data=iris,hue="variety",style="variety")
plt.title("Sepal Length v/s Sepal Width")
plt.show()
```

OUTPUT:

