DATA SCIENCE LAB

CYCLE-3

Mathplotlib

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

a) **X- axis - Year**
b) **Y –axis - Car Value**
c) **title –Value Depreciation (left Aligned)**
d) **Line Style dashdot and Line-color should be red**
e) **Point using * symbol with green color and size 20**

Subplot() provides multiple plots in one figure.

CODE:

```
import matplotlib.pyplot as plt
year=[2001,2002,2003,2004,2005,2006,2007]
car_value=[24000,22500,19700,17500,14500,10000,5800]
plt.figure(figsize=(8,4))
plt.subplot(1,1,1)
plt.plot(year,car_value,color='red',linestyle='-.',label='Car Value')
plt.scatter(year,car_value,color='green',marker='*',s=20,label='Data Point')
plt.xlabel('Year')
plt.ylabel('Car Value')
plt.title('Value Depreciation', loc='left')
plt.legend()
plt.show()
```

OUTPUT:

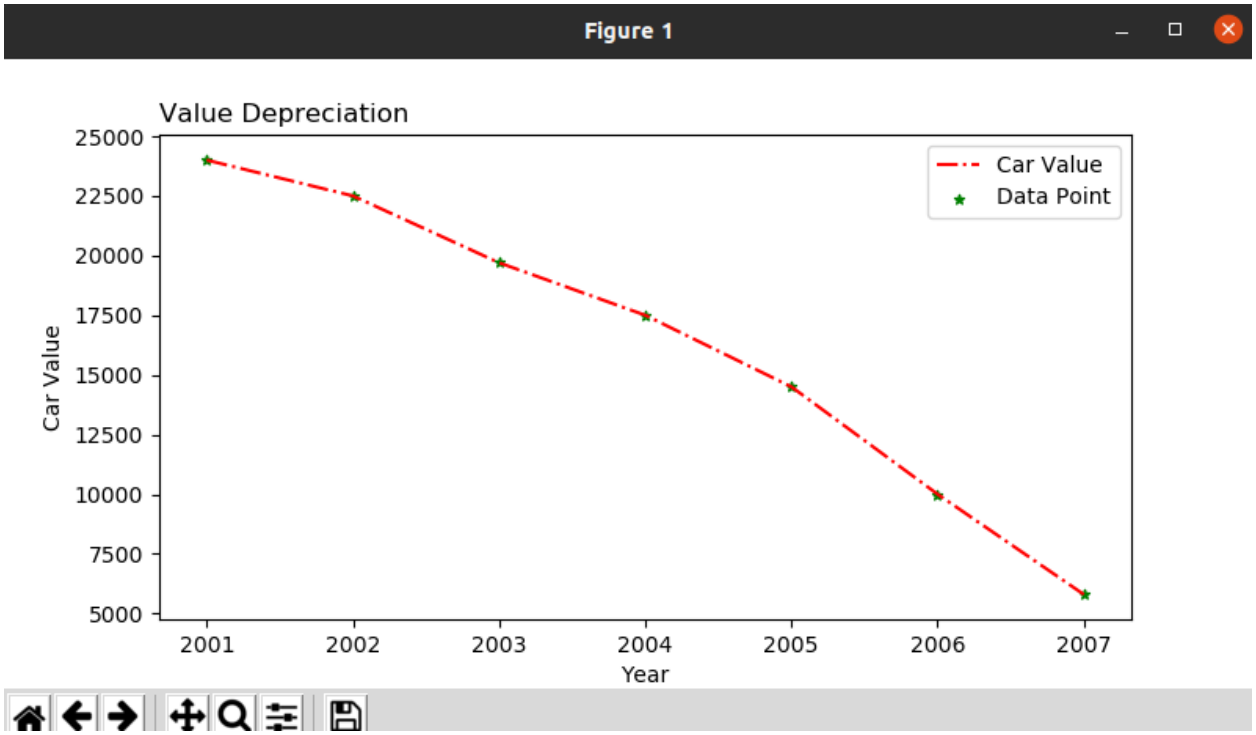


Figure 1
Value Depreciation
Car Value
Data Point
25000
22500
20000
17500
15000
12500
10000
7500
5000
Car Value
2001
2002
2003
2004
2005
2006
2007
Year

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:

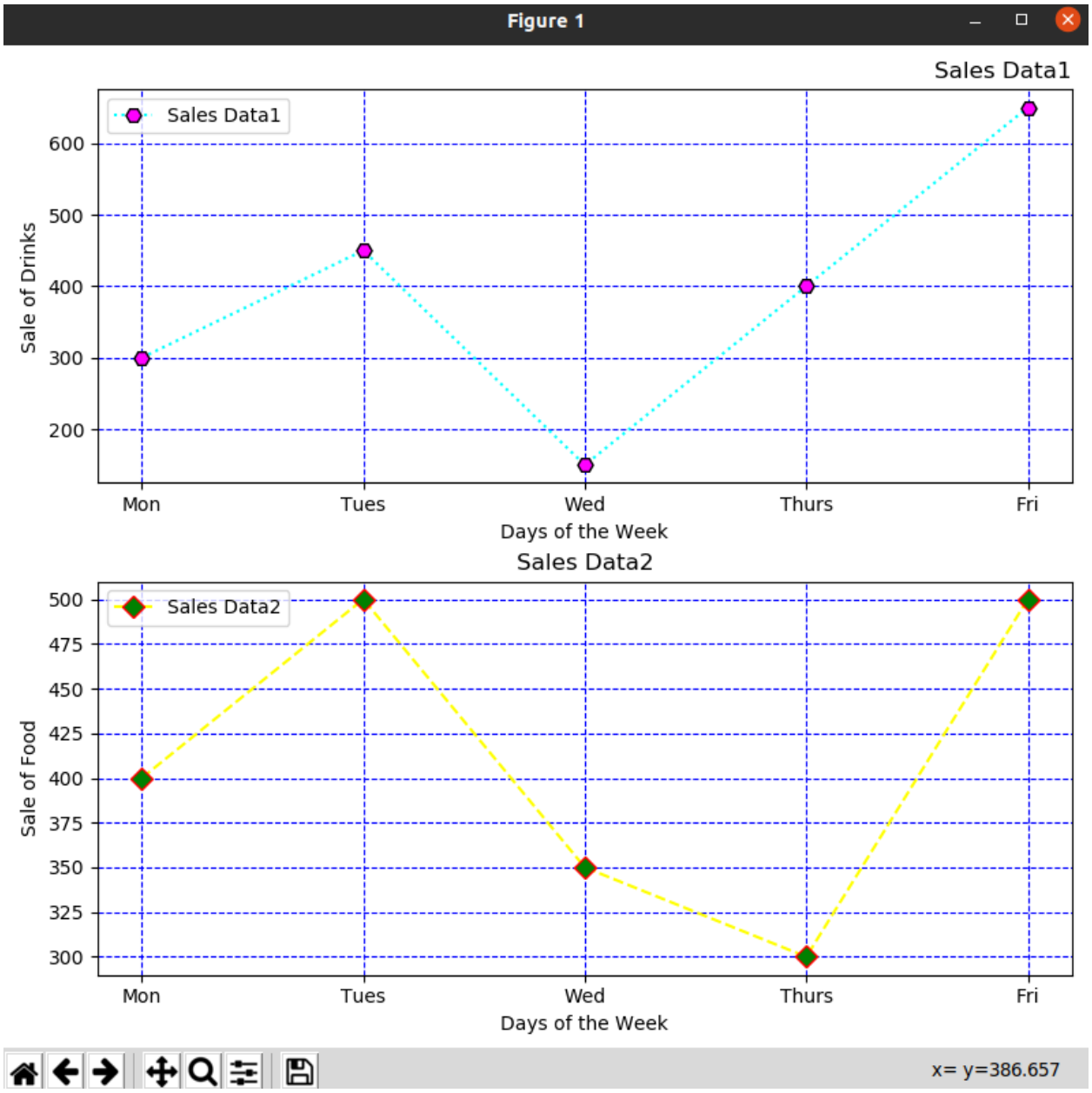
X label- Days of Week
Y label-Sale of Food
Title-Sales Data2 (center aligned)
Line –dashed with yellow color
Points- diamond shape with color green and outline red

CODE:

```
import matplotlib.pyplot as plt
days = ['Mon', 'Tues', 'Wed', 'Thurs', 'Fri']
drinks_sales = [300, 450, 150, 400, 650]
food_sales = [400, 500, 350, 300, 500]
plt.figure(figsize=(8, 8))
plt.subplot(2, 1, 1)
plt.plot(days, drinks_sales, linestyle='dotted', color='cyan', label='Sales Data1',
marker='H', markersize=8,
        markerfacecolor='magenta', markeredgecolor='black')
plt.xlabel('Days of the Week')
plt.ylabel('Sale of Drinks')
plt.title('Sales Data1', loc='right')
plt.grid(color='blue', linestyle='--')
plt.legend()
plt.subplot(2, 1, 2)
plt.plot(days, food_sales, linestyle='--', color='yellow', label='Sales Data2', marker='D',
markersize=8,
```

```
        markerfacecolor='green', markeredgecolor='red')
plt.xlabel('Days of the Week')
plt.ylabel('Sale of Food')
plt.title('Sales Data2', loc='center')
plt.grid(color='blue', linestyle='--')
plt.legend()
plt.tight_layout()
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

a) **X Label- Months of Year with font size 18**
b) **Y-Label- Sales of Segments**
c) **Title –Sales Data**
d) **Color for Affordable segment- pink**
e) **Color for Luxury Segment- Yellow**
f) **Color for Super luxury segment-blue**

CODE:

```
import matplotlib.pyplot as plt
month=["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]
plt.figure(figsize=(10, 6))
plt.scatter(month,affordable_segment,color='pink',label='Affordable',s=100)
plt.scatter(month,luxury_segment,color='yellow',label='Luxury',s=100)
plt.scatter(month,super_luxury_segment,color='blue',label='Super Luxury',s=100)
plt.xlabel('Months of Year', fontsize=18)
plt.ylabel('Sales of Segments')
plt.title('Sales Data', fontsize=18)
plt.legend()
plt.show()
```

OUTPUT:

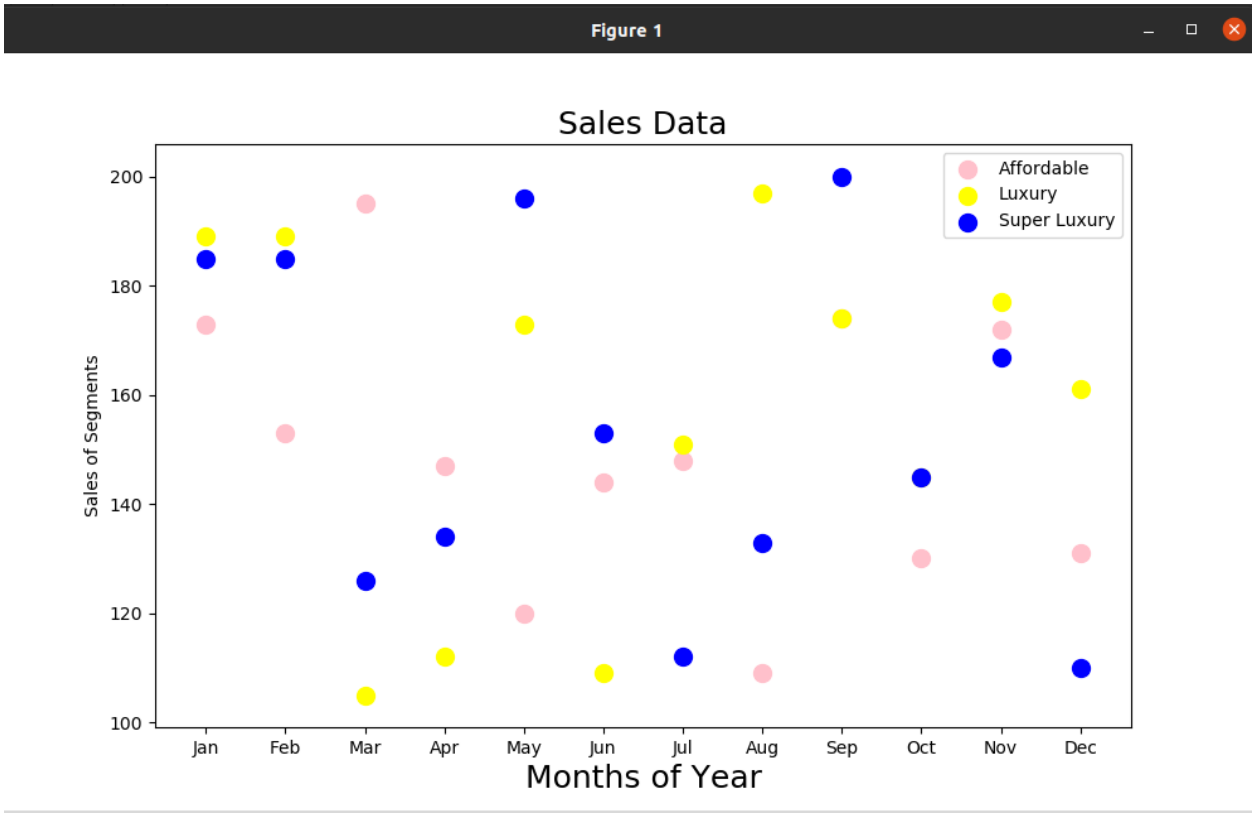


Figure 1
Sales Data
Affordable
Luxury
Super Luxury
Sales of Segments
200
180
160
140
120
100
Jan
Feb
Mar
Apr
May
Jun
Jul
Aug
Sep
Oct
Nov
Dec
Months of Year

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
months = ["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]
plt.figure(figsize=(10, 6))
plt.plot(months, affordable_segment, label='Affordable', color='red', linestyle='-',
marker='o')
plt.plot(months, luxury_segment, label='Luxury', color='green', linestyle='--', marker='s')
plt.plot(months, super_luxury_segment, label='Super Luxury', color='blue', linestyle='-.',
marker='^')
plt.xlabel('Months of Year')
plt.ylabel('Sales of Segments')
plt.title('Sales Data')
plt.legend(loc='upper right')
plt.show()
```

OUTPUT:

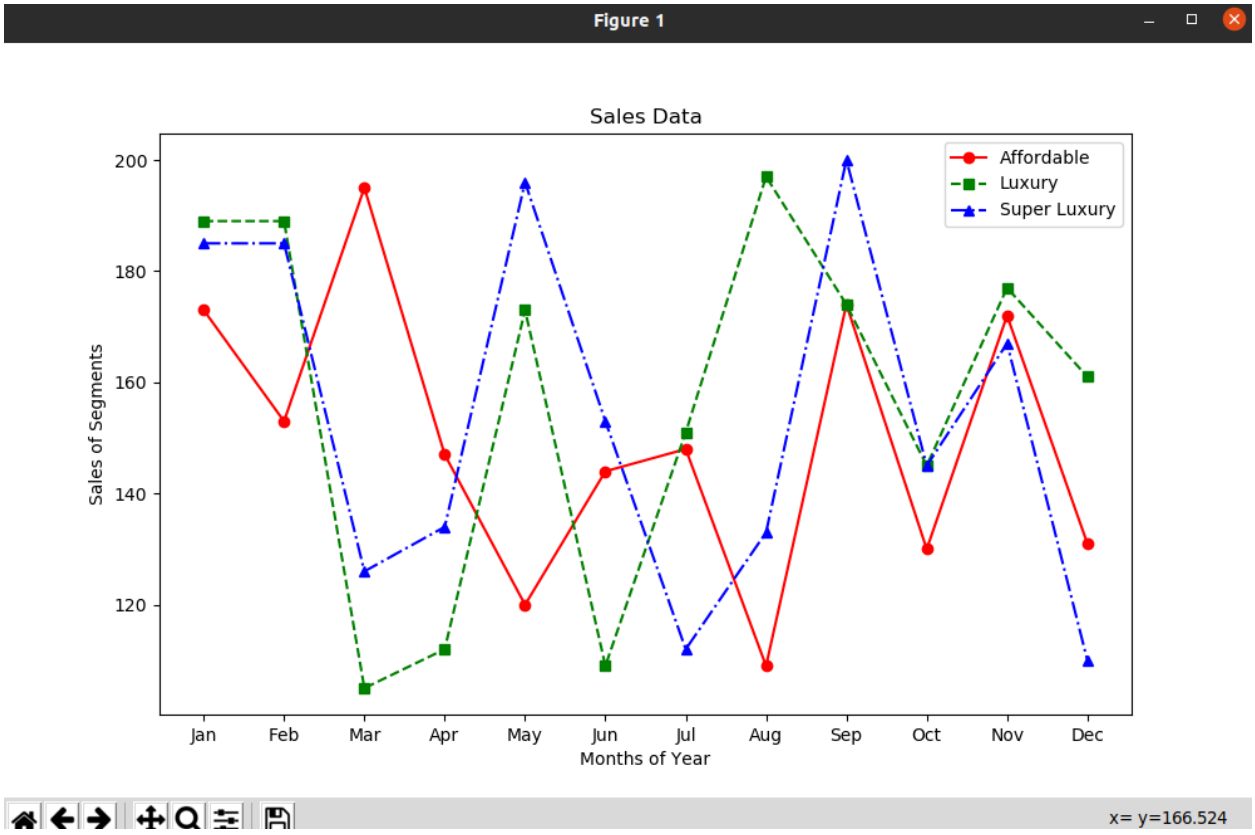


Figure 1
Sales Data
Affordable
Luxury
Super Luxury
Sales of Segments
200
180
160
140
120
Jan
Feb
Mar
Apr
May
Jun
Jul
Aug
Sep
Oct
Nov
Dec
Months of Year
x= y=166.524

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

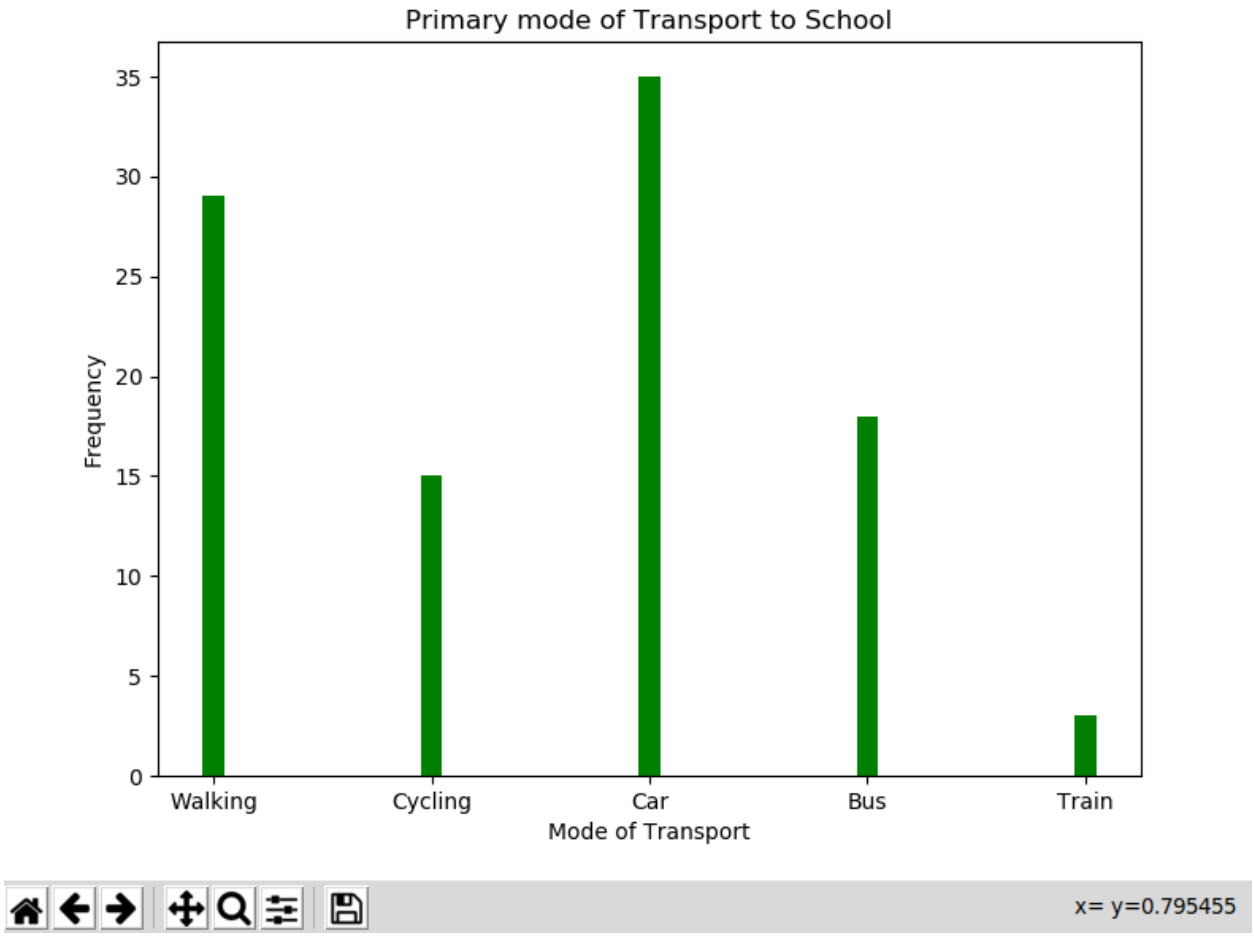
Create a bar graph with

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

CODE:

```
import matplotlib.pyplot as plt
modes_of_transport=["Walking","Cycling","Car","Bus","Train"]
frequency=[29,15,35,18,3]
plt.figure(figsize=(8, 6))
plt.bar(modes_of_transport,frequency,width=0.1,color='green')
plt.xlabel('Mode of Transport')
plt.ylabel('Frequency')
plt.title('Primary mode of Transport to School')
plt.show()
```

OUTPUT:

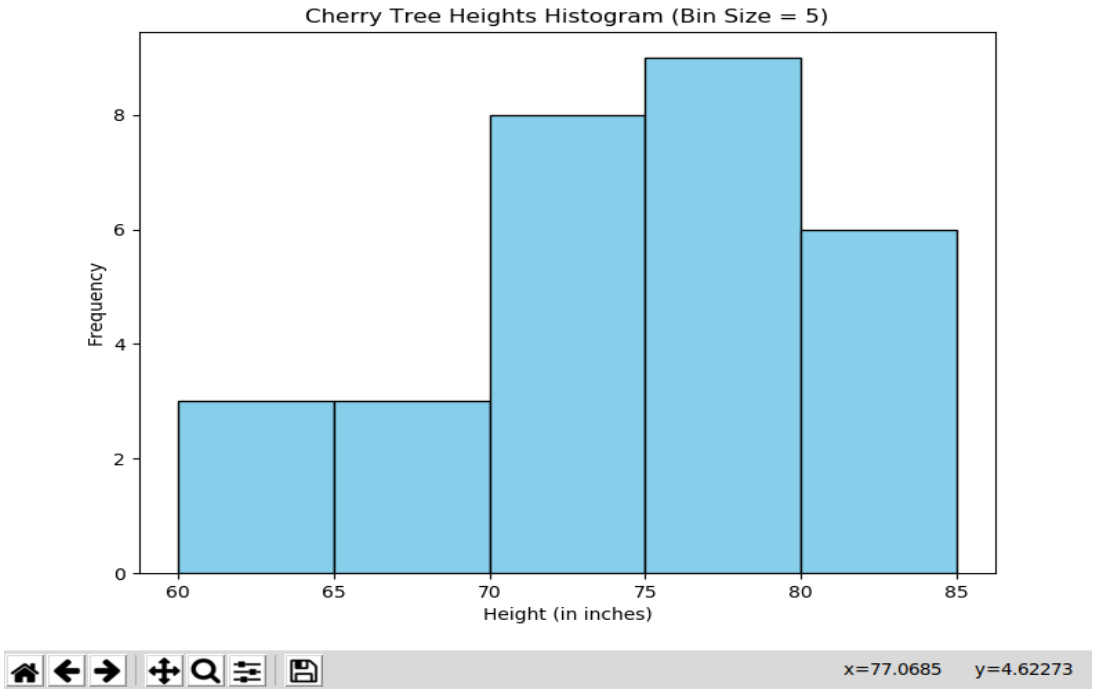

Primary mode of Transport to School
Frequency
0
5
10
15
20
25
30
35
Walking
Cycling
Car
Bus
Train
Mode of Transport
x= y=0.795455

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
heights=[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.figure(figsize=(8,6))
plt.hist(heights, bins=range(60, 90, 5), edgecolor='black', color='skyblue')
plt.xlabel('Height (in inches)')
plt.ylabel('Frequency')
plt.title('Cherry Tree Heights Histogram (Bin Size = 5)')
plt.show()
```

OUTPUT:



Import pandas and seaborn packages:

7. Using the pandas function read_csv(), read the given ‘iris’ data set.

i) Display Shape of the data set.
ii) First 5 and last five rows of data set(head and tail).
iii) Size of dataset.
iv) No. of samples available for each variety.
v) Description of the data set(use describe).

CODE:

```
print("Name:Gopika Unnikrishnan\nRoll No:22MCA030\nCourse Name:DATA SCIENCE LAB\nCourse Code:20MCA241\nDate:17/10/2023")
import pandas as pd
iris_data=pd.read_csv('iris.csv')
print("\ni) Shape of the dataset:")
print(iris_data.shape)
print("\nii) First 5 and last five rows of data set(head and tail):")
print(iris_data.head())
print(iris_data.tail())
dataset_size = iris_data.shape[0]
print("\niii)Size of the dataset:", dataset_size)
sample_counts = iris_data['variety'].value_counts()
print("\niv) Number of samples available for each variety:")
print(sample_counts)
dataset_description = iris_data.describe()
print("\nv) Description of the dataset:")
print(dataset_description)
```

OUTPUT:

```
Name:Gopika Unnikrishnan
Roll No:22MCA030
Course Name:DATA SCIENCE LAB
Course Code:20MCA241
Date:17/10/2023

i) Shape of the dataset:
(150, 5)

ii) First 5 and last five rows of data set(head and tail):
   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
     sepal.length  sepal.width  petal.length  petal.width    variety
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

iii)Size of the dataset: 150
```

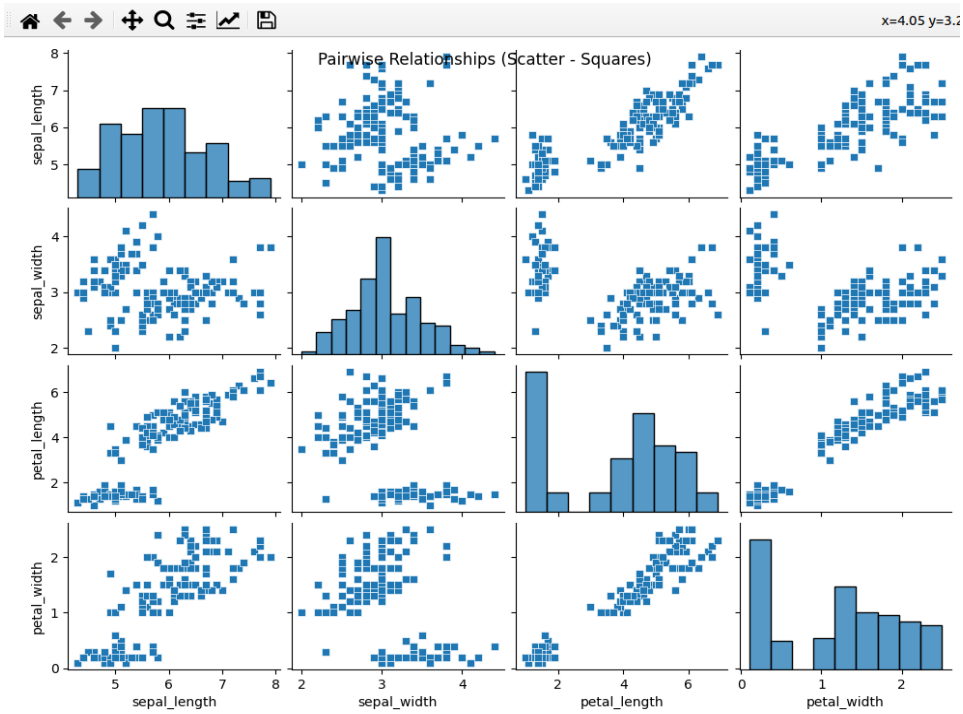
```
iv) Number of samples available for each variety:
variety
Setosa        50
Versicolor    50
Virginica     50
Name: count, dtype: int64

v) Description of the dataset:
       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
```

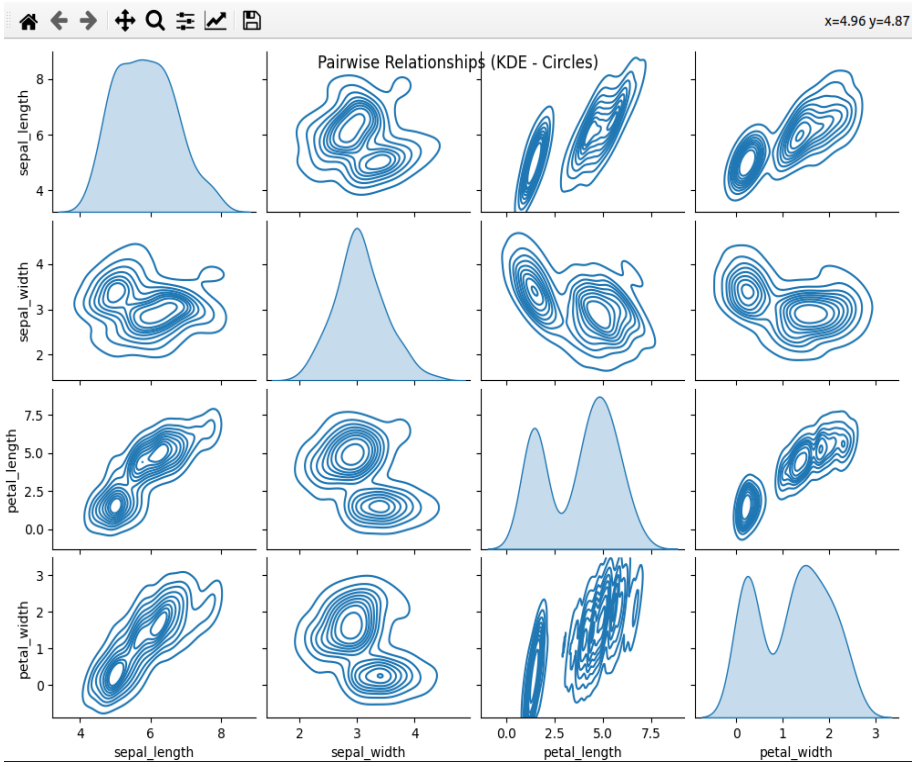
8. Use pairplot() function in seaborn to display pairwise relationships between attributes. Try different kind of plots {‘scatter’, ‘kde’, ‘hist’, ‘reg’} and different kind of markers.

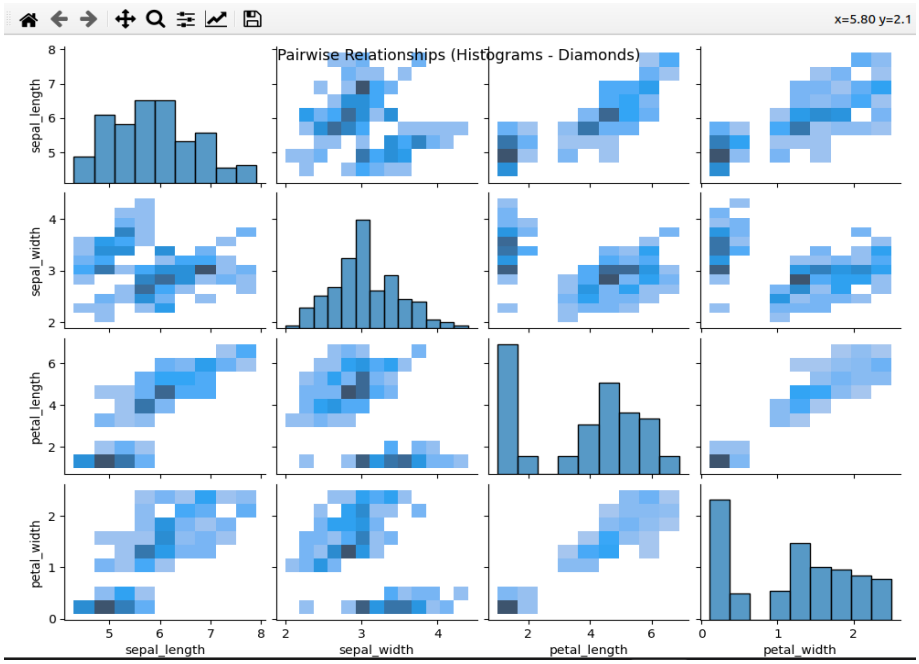
CODE:

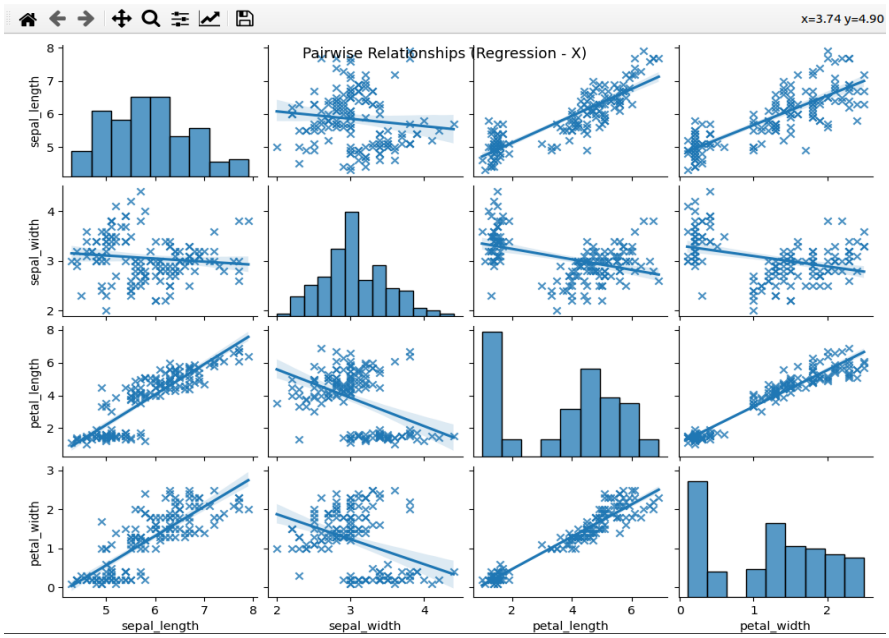
```
import seaborn as sns
import matplotlib.pyplot as plt
iris = sns.load_dataset("iris")
sns.pairplot(iris, kind='scatter', markers='s')
plt.suptitle("Pairwise Relationships (Scatter - Squares)")
plt.show()
sns.pairplot(iris, kind='kde', markers='o')
plt.suptitle("Pairwise Relationships (KDE - Circles)")
plt.show()
sns.pairplot(iris, kind='hist', markers='D')
plt.suptitle("Pairwise Relationships (Histograms - Diamonds)")
plt.show()
sns.pairplot(iris, kind='reg', markers='x')
plt.suptitle("Pairwise Relationships (Regression - X)")
plt.show()
```

OUTPUT:




x=4.96 y=4.87
Pairwise Relationships (KDE - Circles)
sepal_length
sepal_width
petal_length
petal_width


x=5.80 y=2.1
Pairwise Relationships (Histograms - Diamonds)
sepal_length
sepal_width
petal_length
petal_width


x=3.74 y=4.90
Pairwise Relationships (Regression - X)
sepal_length
sepal_width
petal_length
petal_width

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

1) displot()

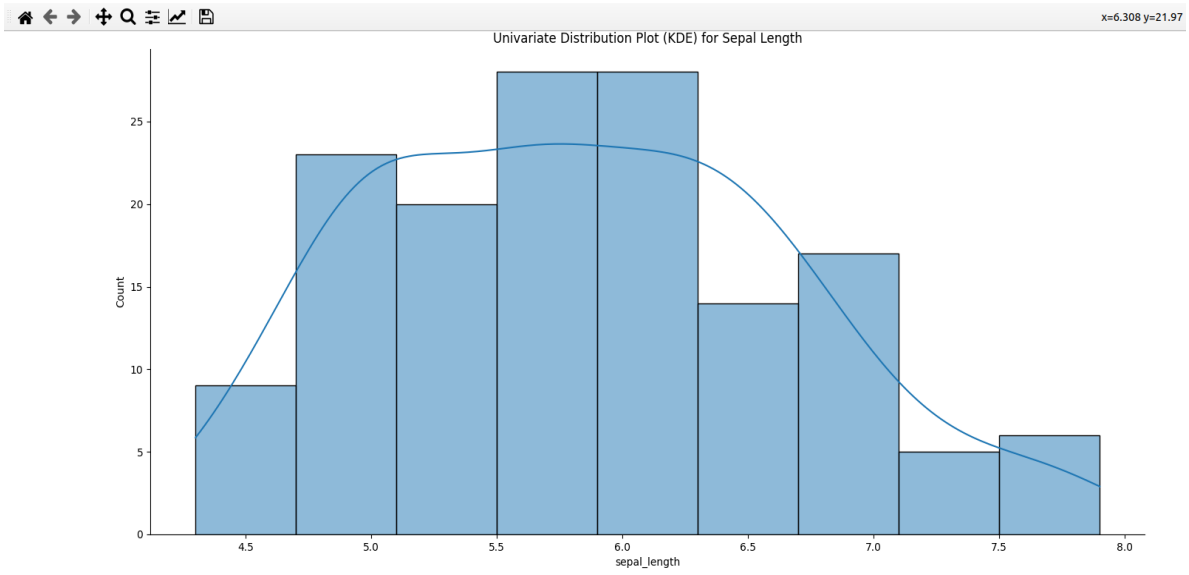
2) histplot()

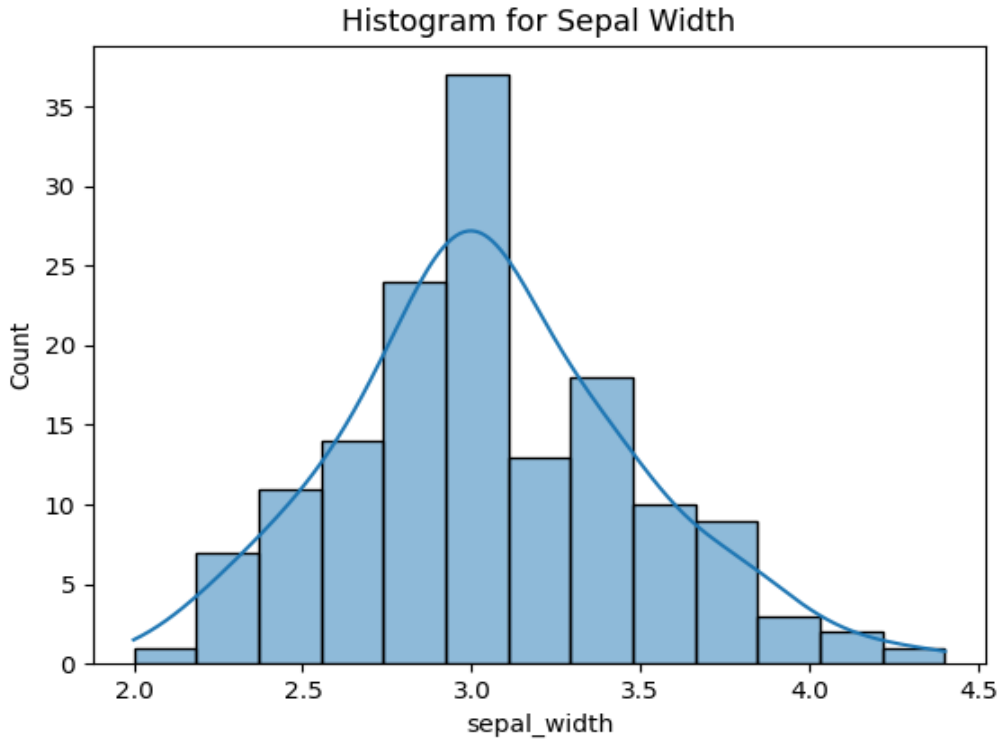
3) relplot()

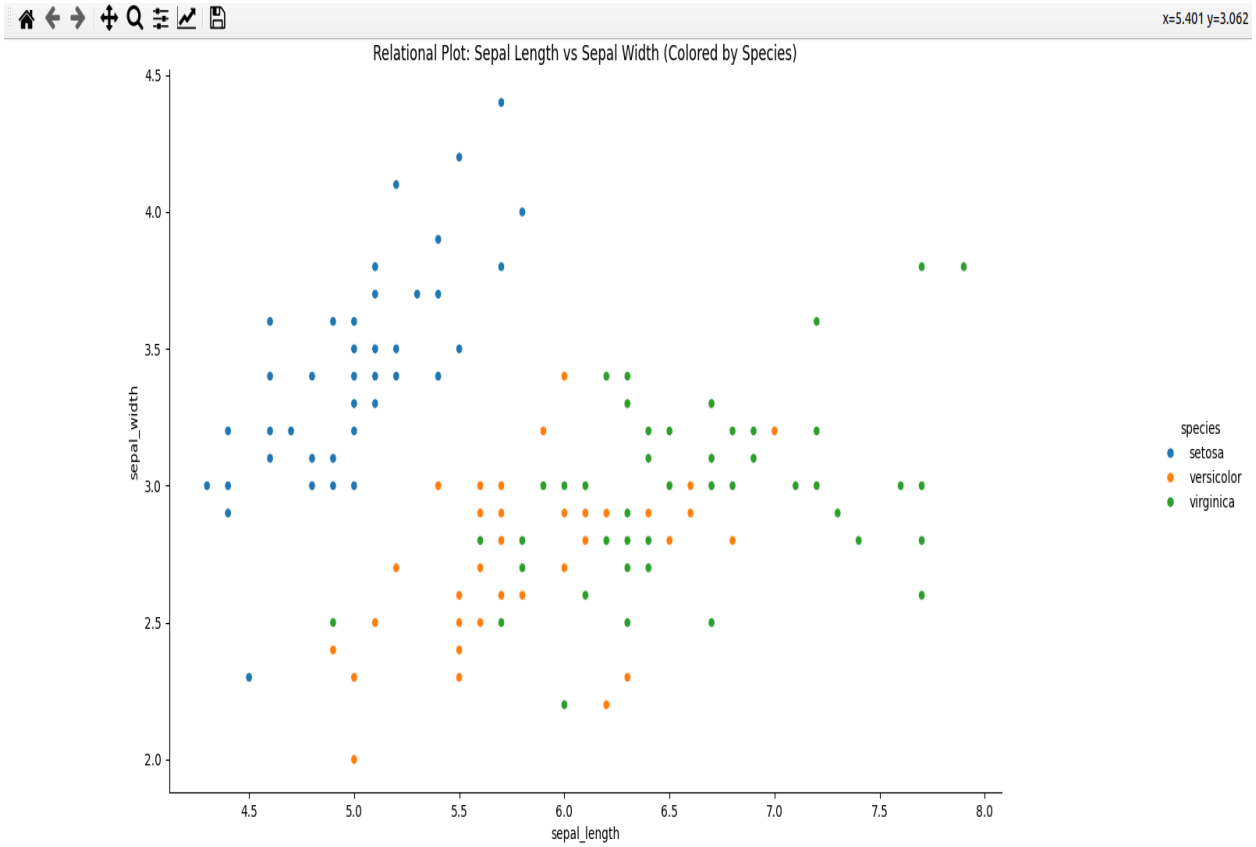
CODE:

```
import seaborn as sns
import matplotlib.pyplot as plt
iris = sns.load_dataset("iris")
sns.displot(iris['sepal_length'], kde=True)
plt.title("Univariate Distribution Plot (KDE) for Sepal Length")
plt.show()
sns.histplot(iris['sepal_width'], kde=True)
plt.title("Histogram for Sepal Width")
plt.show()
sns.relplot(x='sepal_length', y='sepal_width', data=iris, kind='scatter', hue='species')
plt.title("Relational Plot: Sepal Length vs Sepal Width (Colored by Species)")
plt.show()
```

OUTPUT:




Histogram for Sepal Width
Count
sepal_width
0
5
10
15
20
25
30
35
2.0
2.5
3.0
3.5
4.0
4.5


x=5.401 y=3.062
Relational Plot: Sepal Length vs Sepal Width (Colored by Species)
sepal_width
sepal_length
species
setosa
versicolor
virginica
2.0
2.5
3.0
3.5
4.0
4.5
4.5
5.0
5.5
6.0
6.5
7.0
7.5
8.0