

**DATA SCIENCE AND MACHINE LEARNING**  
**LAB CYCLE 3**

## Mathplotlib

Demonstrate creating various types of charts and plots using functions in mathplotlib 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

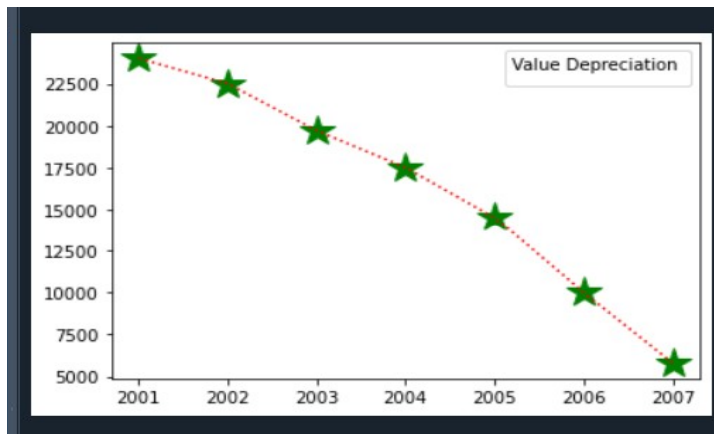
Represent the following information using a line graph with following style properties

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

Subplot() provides multiple plots in one figure.

### PROGRAM:

```
import matplotlib.pyplot as plt
import numpy as np
xpoints = np.array([2001, 2002,2003,2004,2005,2006,2007])
ypoints = np.array([24000, 22500,19700,17500,14500,10000,5800])
plt.plot(xpoints, ypoints, '*g',ms = 20)
plt.plot(xpoints, ypoints, ':r')
leg = plt.legend(title="Value Depreciation ")
leg. legend_box.align = "left"
plt.show()
```



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

	Mon	Tues	Wed	Thurs	Fri
	300	450	150	400	650
	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

**PROGRAM:**

```
import matplotlib.pyplot as plt
import numpy as np

#plot 1:
x = np.array(['mon', 'tue', 'wed', 'thur', 'fri'])
```

```
y = np.array([300, 450, 150, 400,65])
```

```
plt.subplot(1, 2, 1)
```

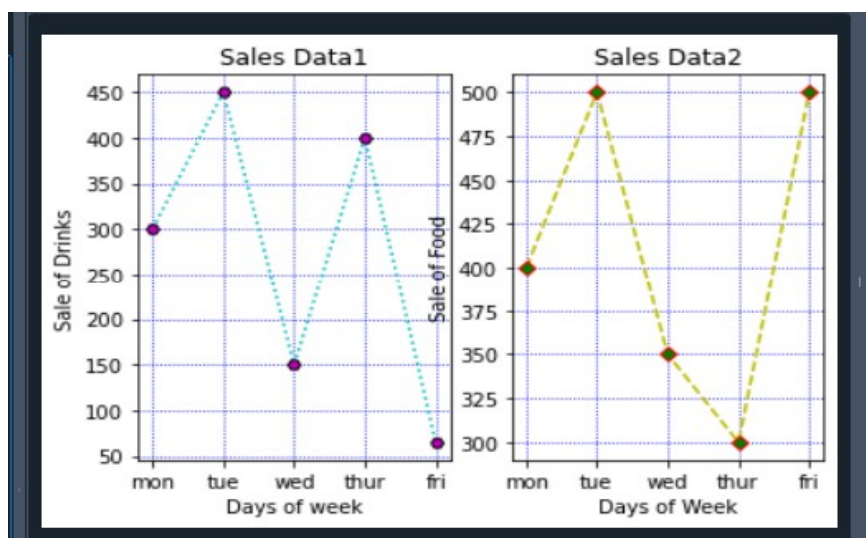
```
plt.title("Sales Data1")  
plt.xlabel("Days of week")  
plt.ylabel("Sale of Drinks")  
plt.plot(x,y,':c')  
plt.plot(x,y,'Hm',mec = 'k')
```

```
plt.grid(color = 'blue', linestyle = 'dotted')  
#plot 2:
```

```
c = np.array(['mon', 'tue', 'wed', 'thur','fri'])  
v = np.array([400, 500, 350, 300,500])  
plt.subplot(1, 2, 2)  
plt.title("Sales Data2")  
plt.xlabel("Days of Week")  
plt.ylabel("Sale of Food")
```

```
plt.plot(c,v,'--y')  
plt.plot(c,v,'Dg',mec = 'r')
```

```
plt.grid(color = 'blue', linestyle = 'dotted')  
plt.show()
```



### 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

#### PROGRAM:

```
import matplotlib.pyplot as plt
import numpy as np
```

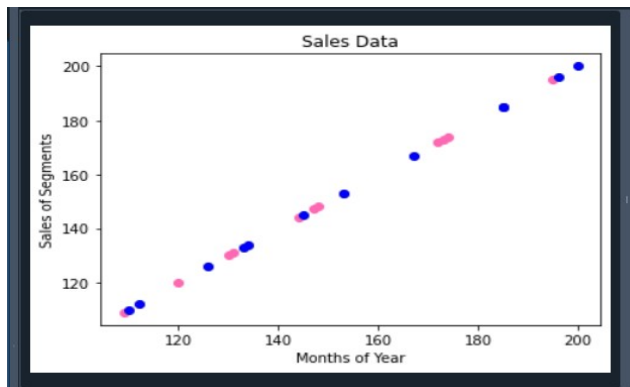
```
plt.title("Sales Data")
plt.xlabel("Months of Year")
plt.ylabel("Sales of Segments")
```

```
x = np.array([173,153,195,147,120,144,148,109,174,130,172,131])
y = np.array([173,153,195,147,120,144,148,109,174,130,172,131])
plt.scatter(x,y, color = 'hotpink')
```

```
x = np.array([185,185,126,134,196,153,112,133,200,145,167,110])
y = np.array([185,185,126,134,196,153,112,133,200,145,167,110])
plt.scatter(x,y, color = 'yellow')
```

```
x = np.array([185,185,126,134,196,153,112,133,200,145,167,110])
y = np.array([185,185,126,134,196,153,112,133,200,145,167,110])
plt.scatter(x,y, color = 'blue')
```

```
plt.show()
```



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())

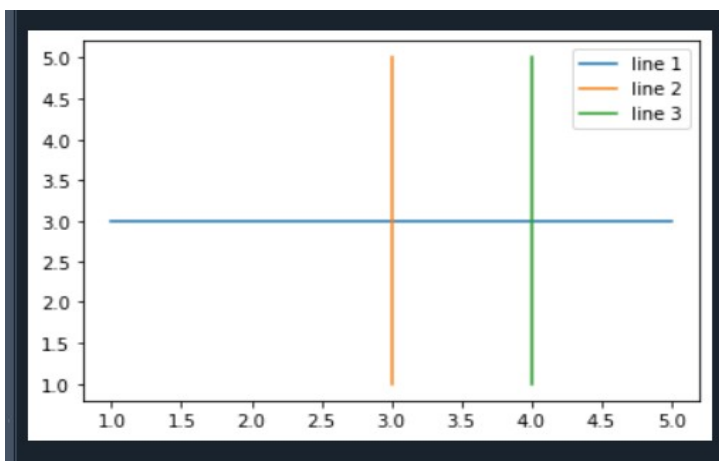
Use different colors and line styles for 3 different lines

#### PROGRAM:

```
import matplotlib.pyplot as plt
import numpy as np
```

```
x = [1,2,3,4,5]
y = [3,3,3,3,3]
z = [4,4,4,4,4]
```

```
plt.plot(x, y, label = "line 1")
plt.plot(y, x, label = "line 2")
plt.plot(z, x, label = "line 3")
plt.legend()
plt.show()
```



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

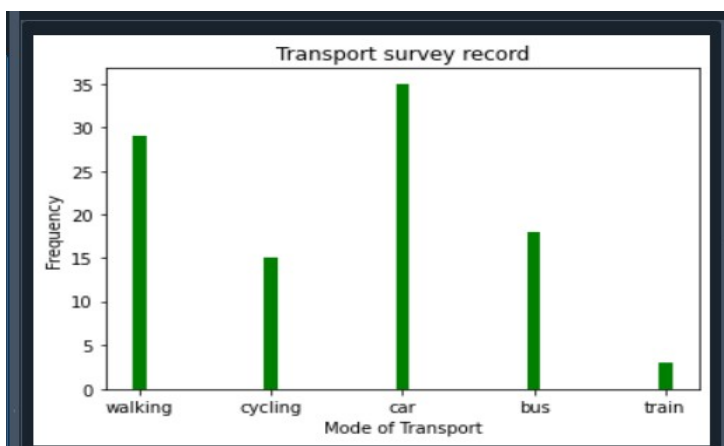
Create a bar graph with

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

**PROGRAM:**

```
import matplotlib.pyplot as plt
import numpy as np
x = np.array(["walking", "cycling", "car", "bus", "train"])
y = np.array([29, 15, 35, 18, 3])
plt.title("Transport survey record")
plt.xlabel("Mode of Transport")
plt.ylabel("Frequency")

plt.bar(x, y,color = "green", width = 0.1)
plt.show()
```



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

**PROGRAM:**

```
import matplotlib.pyplot as plt
height = [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(height, edgecolor='red', bins=5)
plt.show()
```

