

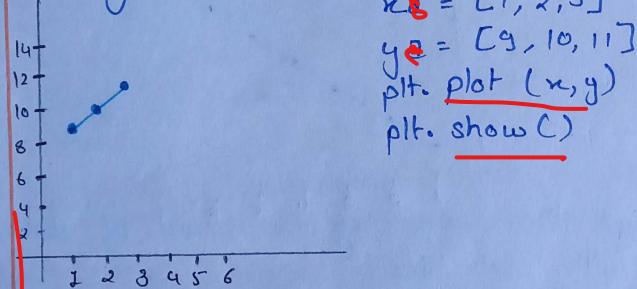
Ques 1

What is a graph? How can you show graph in python?
Name 5 graph you can use in python.

A graph is a data structure that consists of a collection of vertices (also called nodes) and edges.

- It's used to represent relationship or connections b/w different entities.
- Graphs are widely used in various applications, including Computer networks, Social Networks, transportation networks and data structures and algorithms.
- Graphs can be represented visually using diagrams.

Graphs in python can be plotted by using Matplotlib library for ex. import matplotlib.pyplot as plt



- # five graphs that can be created in python are:-
1. Line Graph:- A line graph is used to represent the relationship b/w two continuous variables over continuous intervals for example trees (see previous graph)
 2. Bar Graph:- A bar graph represents data using rectangular bars of varying height or lengths.

• Suitable for creating categories or discrete data.

Eg - cate = ['A', 'B', 'C', 'D']

val = [10, 7, 12, 8]

plot plt.plot(kind='bar', x=cate, y=val)



AI DUAL CAMERA

Shot by @mr_ravindra8896

3. Pie chart:- A pie chart is a circular statistical graph divided into slices to illustrate proportions.

- used to represent the composition or distribution of different categories as parts of a whole

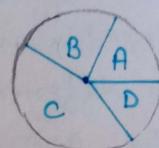
Eg

```
lab = ['A', 'B', 'C', 'D']
```

```
sizes = [15, 30, 45, 10]
```

```
plt.pie(sizes, labels=lab)
```

```
plt.show()
```



4. Scatter plot:- A scatter plot displays individual data points as dots in two-D space

- Suitable for visualizing the correlation or relationship

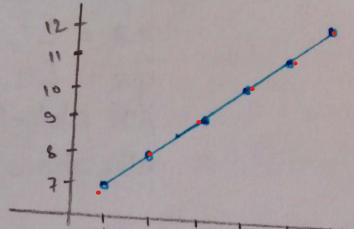
Eg

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

```
y = [7, 8, 9, 10, 11]
```

```
plt.scatter(x, y)
```

```
plt.show()
```



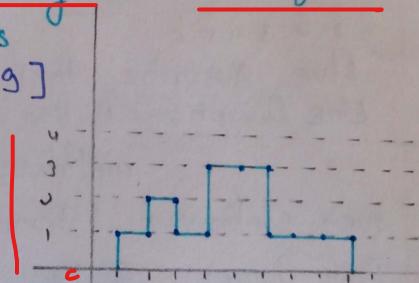
5. Histogram:- Displays the frequency or count of data points with each bins

Eg

```
data = [1, 2, 3, 3, 4, 5, 5, 5, 6, 6, 7, 8, 9]
```

```
plt.hist(data, bins=5)
```

```
plt.show()
```



Qn

What is the use of plot() functions in matplotlib?

plot() is a fundamental method used for creating line plots or line graphs.

- plot() provides a wide range of options for customizing the appearance of the plot including line styles, colors, markers, labels and more.



AI DUAL CAMERA

Shot by @mr_ravindra8896

Eg
import matplotlib.pyplot as plt

x = [1, 2, 3]

y = [4, 5, 6]

plt.plot(x, y) # will simply convert the points

plt.show() # into a line graph

will print the graph

~~Ques 3~~

Define labels, annotations and legends

i) Labels :- Labels refers to textual annotations associated with datapoints or axes

- These labels provide additional information about data

Eg
import matplotlib.pyplot as plt

lab = ['Devil', 'God']

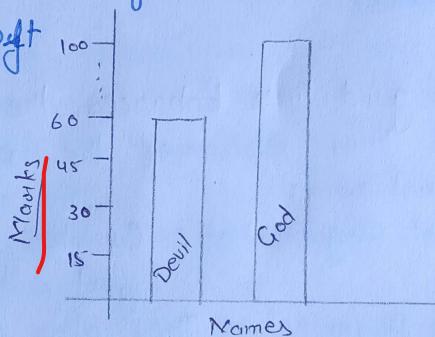
marks = [60, 100]

plt.bar(x, y, lab, marks)

plt.xlabel("Name")

plt.ylabel("Marks")

plt.show()



ii) Annotations :- annotations refers to textual or graphical elements that provides additional information about specific data points

- annotate() is used to add a text annotation with an arrow to point to a specific data.

Eg
import matplotlib.pyplot as plt

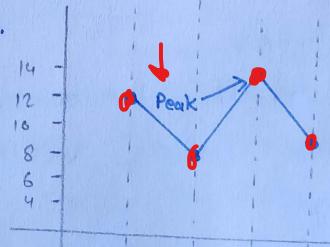
x = [1, 2, 3, 4]

y = [12, 8, 14, 9]

plt.plot(x, y)

plt.annotate('Peak', xy=2, xytext=(3, 14), arrowprops=dict(arrowstyle='->'))

plt.show()



AI DUAL CAMERA

Shot by @mr_ravindra8896

iii) Legends : A legend is a graphical representation of the elements present in a plot

- Legends are commonly used with multiple lines

Eg.

$x = [1, 2, 3]$

$y_1 = [10, 11, 12]$

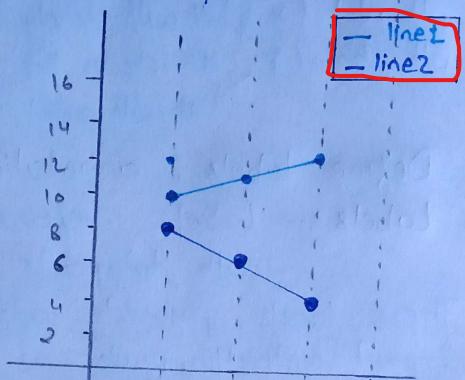
$y_2 = [8, 6, 4]$

✓ plt. plot (x, y_1 , label = 'line1')

✓ plt. plot (x, y_2 , label = 'line2')

plt. legend()

plt. show()



Ques

How grid lines Enhances the presentation of a graph?
Grid lines enhances the presentation of a graph in several way:

- i) Visual Organisation :- Creates an organised and structured layout within the plotting area.
- ii) Data Alignment :- They align data points with axes and each other so that it is easier to read and understand data accurately.
- iii) Data Range Assessment :- Grid lines helps in estimating the range of data values along the axes.
- By counting the number of grid lines b/w the minimum & max values, viewers can easily estimate the data range.



AI DUAL CAMERA

Shot by @mr_ravindra8896

Ques 5 Explain the significance of labelling the graph in python. Show with an example.

Labelling the graphs in python is significant as it provides essential information about the data being plotted.

- It enhances the overall understanding and interpretation of graph.
- labels convey the meaning of axes.
- provides context for the plotted data.
- Helps the reader or viewer to make sense of the visual representation.

For example:- let consider the population growth several years without proper labelling graph would lack crucial information so, adding labels is important.

```
import matplotlib.pyplot as plt
```

```
year = [2010, 2012, 2014, 2016, 2018]
```

```
pop = [15, 18, 20, 22, 25] # (in millions)
```

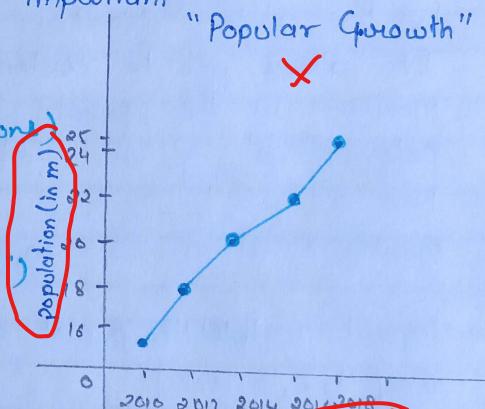
```
plt.plot(year, pop)
```

```
plt.xlabel("Year")
```

```
plt.ylabel("Population (in millions)")
```

```
plt.title("Population Growth")
```

```
plt.show()
```



Ques 6 How to create legends in python using Matplotlib? Explain the concept with following example by creating:-

a) one line graph representation sales in 2014

```
import matplotlib.pyplot as plt
```

```
Sales_2014 = [100, 500, 200, 2500, 300, 500]
```

```
plt.plot(Sales_2014, label='Sales_2014')
```

b) One line graph representing sales in 2015-

```
Sales_2015 = [100, 300, 5000, 600, 8000]
```

```
plt.plot(Sales_2015, label='Sales_2015')
```

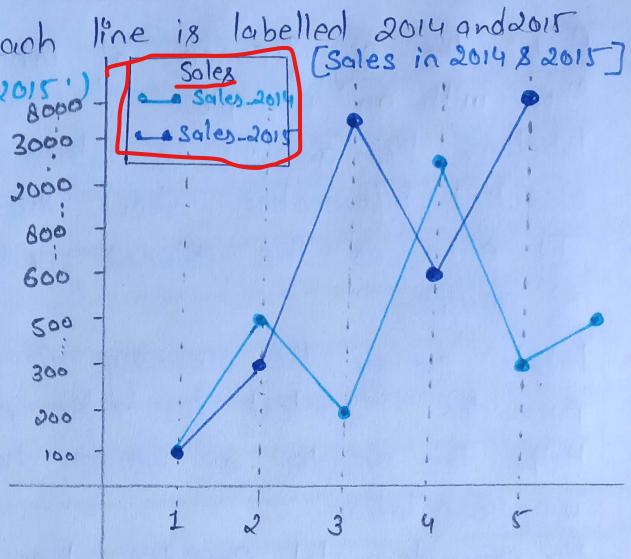


AI DUAL CAMERA

Shot by @mr_ravindra8896

(c)

an entry in legend for each line is labelled 2014 and 2015
plt. title ('Sales in 2014 & 2015')
plt. legend (Title = 'Sales')
plt. show ()



Qn7

What is format string 'Fmt' in `pyplot` `matplotlib`?

The format string 'Fmt' is a string parameter that specifies the line style, marker style and color of a plotted line

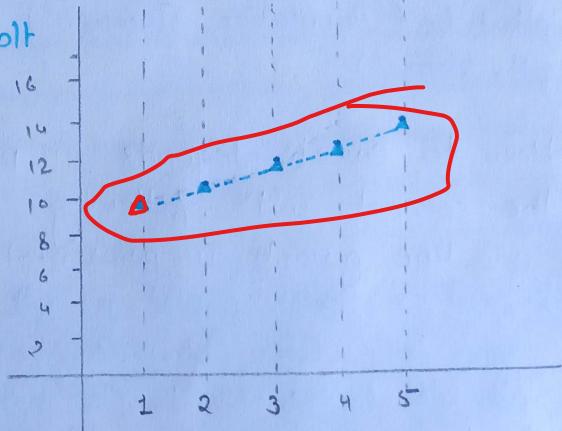
It allows you to customize the appearance of the line and marker in the graph.

③ color → 'b', 'g', 'r' ... etc

② line → '—', '--', '-.', ':'

① marker → 'o', 'v', 's' → square, 'x' → cross, '*' etc.

```
import matplotlib.pyplot as plt
x = [1, 2, 3, 4, 5]
y = [10, 11, 12, 13, 14]
plt.plot(x, y, '^-b')
plt.show()
```



Note: dotted vertical lines for our personal purpose



AI DUAL CAMERA

Shot by @mr_ravindra8896