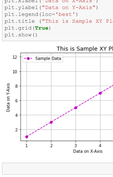
Matplotlib in Python

Plotting with one-dimension data (without defining x-axis)

In [13]:
```
import matplotlib.pyplot as plt
data = [1, 3, 5, 7, 9, 12]
plt.plot(data)
plt.show()
```

Plotting a fixed x and y range

In [14]:
```
import matplotlib.pyplot as plt
data = [1, 2, 3, 7, 9, 12]
x_data = [1,2,3,4,5,6]
plt.plot(x_data, data)
plt.show()
```



Adding various additional features to a plot (Customization)

In [23]:
```
import matplotlib.pyplot as plt
data = [1, 3, 5, 7, 9, 12]
x_data = [1,2,3,4,5,6]
plt.plot(x_data, data, "o--m",label='Sample Data')
plt.xlabel("Data on X-Axis")
plt.ylabel("Data on Y-Axis")
plt.legend(loc='best')
plt.title ("This is Sample XY Plot", fontsize=14)
plt.grid(True)
plt.show()
```

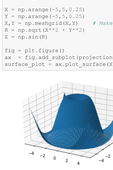
In [ ]:
```

```

Histogram in Matplotlib

In [25]:
```
import matplotlib.pyplot as plt
import numpy as np
data = np.random.randn(1000)
hl = 0.2*data+4
plt.hist(hl, 5, align= 'mid', color = 'g')
plt.title("Histogram plot for Random Data", fontsize=12)
plt.show()
```

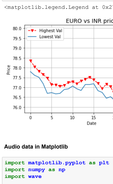
Sactter plot

In [26]:
```
import matplotlib.pyplot as plt
import numpy as np
x = np.arange(70)
y1 = np.random.randn(70)
y2 = np.random.randn(70)
plt.scatter(x, y1, marker ='o', label='y1 with o')
plt.scatter(x, y2, marker ='v', label='y2 with v')
plt.legend(loc = 'upper left')
plt.show()
```



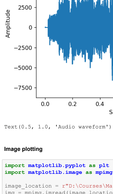
Pie Chart

In [27]:
```
import matplotlib.pyplot as plt

percent_result = [12, 42, 20, 18, 8]
sub = ['Math','Phy','Chem','Bio','IT']

plt.pie(percent_result,  explode = (0,0.1,0.1,0,0), labels = sub)
plt.show()
```



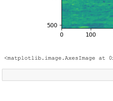
Stacked bar chart

In [28]:
```
import matplotlib.pyplot as plt

avg_score = [50, 20, 57, 60, 26, 55, 88, 19, 74, 60]
sub = ['Math','Phy','Chem','Bio','IT', 'Sci', 'Tec','PT','ECE', 'Comp']
grace_marks = [10, 34, 6, 5, 7, 15, 7, 20, 6, 9]

plt.title('Average score for Subjects', fontsize =16)
plt.xlabel('Subjects',fontsize = 14, color = 'm')
plt.ylabel('Scores', fontsize = 14, color = 'g')
colors = ['blue', 'orange', 'purple', 'cyan', 'magenta', 'yellow', 'black', 'purple', 'red', 'orange']

plt.bar(sub,avg_score, width = 0.7,color= colors)
plt.bar(sub, grace_marks, width = 0.7, bottom = avg_score, color = 'green')

plt.show()
```



3D Plots

In [29]:
```
from mpl_toolkits.mplot3d import Axes3D

X = np.arange(-5,5,0.25)
Y = np.arange(-5,5,0.25)
X,Y = np.meshgrid(X,Y)      # Make N-D coordinate arrays for vectorised evaluations
R = np.sqrt(X**2 + Y**2)
Z = np.sin(R)

fig = plt.figure()
ax  = fig.add_subplot(projection='3d')
surface_plot = ax.plot_surface(X,Y,Z, antialiased=True, rstride=1, cstride=1)
```



Time series data

In [30]:
```
import matplotlib.pyplot as plt
import pandas as pd

file_path = r"D:\Courses\Matplotlib_Python\Code\EUR_INR.csv'
data = pd.read_csv(file_path)
high_val = data['High']
low_val = data['Low']


plt.plot(high_val,"r--.r" ,label = 'Highest Val')
plt.plot(low_val, label = 'Lowest Val')
plt.xlabel('Date')
plt.ylabel('Price')
plt.title('EURO vs INR price', fontsize =14)
plt.grid()
plt.legend()
```

Out[30]:
```
<matplotlib.legend.Legend at 0x2760df274c8>
```


Audio data in Matplotlib

In [32]:
```
import matplotlib.pyplot as plt
import numpy as np
import wave

file_location = r"D:\Courses\Matplotlib_Python\Code\PinkPanther60.wav"
wav_data = wave.open(file_location, 'r')

sample_length = 352 * 1000 * 60   #bitrate 352kbps , audio length 60 seconds
read_file_at_sample = wav_data.readframes(sample_length)
sig1 = np.frombuffer(read_file_at_sample, dtype=np.int16)

plt.plot(sig1)
plt.xlabel('Sample rate * time')
plt.ylabel('Amplitude')
plt.title('Audio waveform', fontsize=14)
```


Out[32]:
```
Text(0.5, 1.0, 'Audio waveform')
```

Image plotting

In [34]:
```
import matplotlib.pyplot as plt
import matplotlib.image as mpimg

image_location = r"D:\Courses\Matplotlib_Python\Code\ship.png"
img = mpimg.imread(image_location)

plt.imshow(img)
```


Out[34]:
```
<matplotlib.image.AxesImage at 0x2760dff8288>
```

In [ ]:
```

```