



# Matplotlib

# Nội Dung

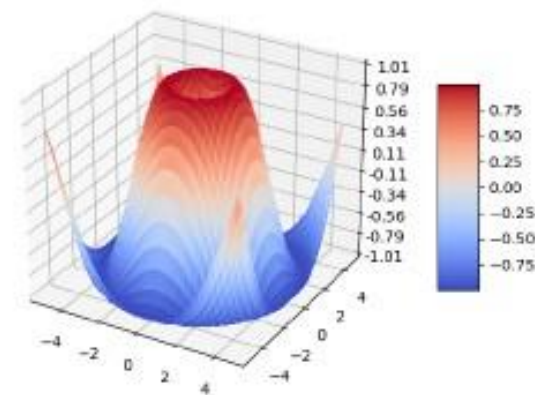
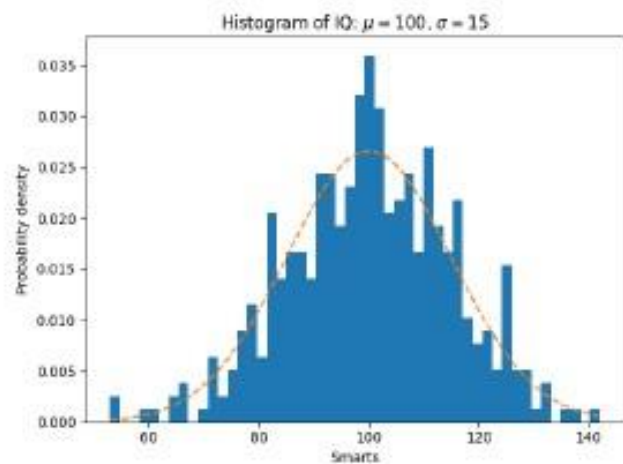
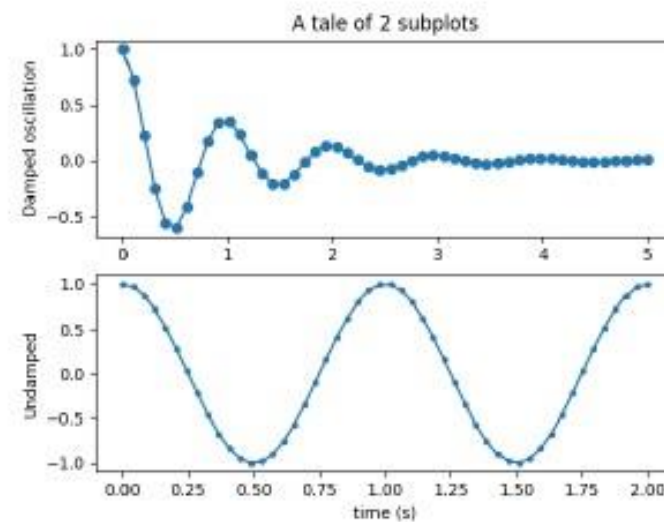
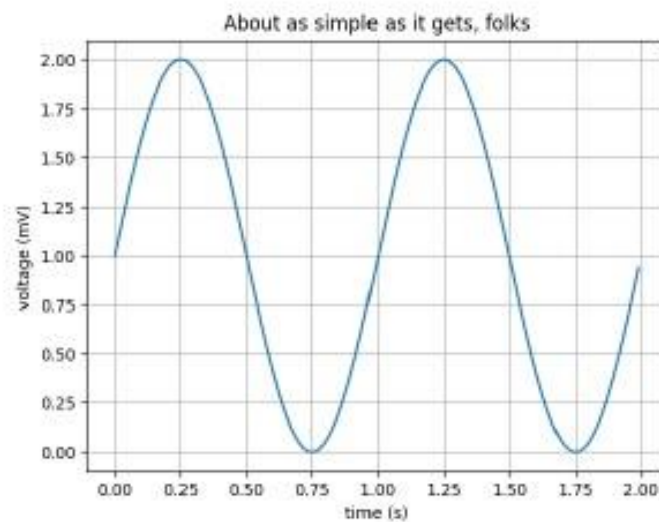
- Giới thiệu về `matplotlib`
- Các biểu đồ thông dụng

# Giới thiệu về matplotlib

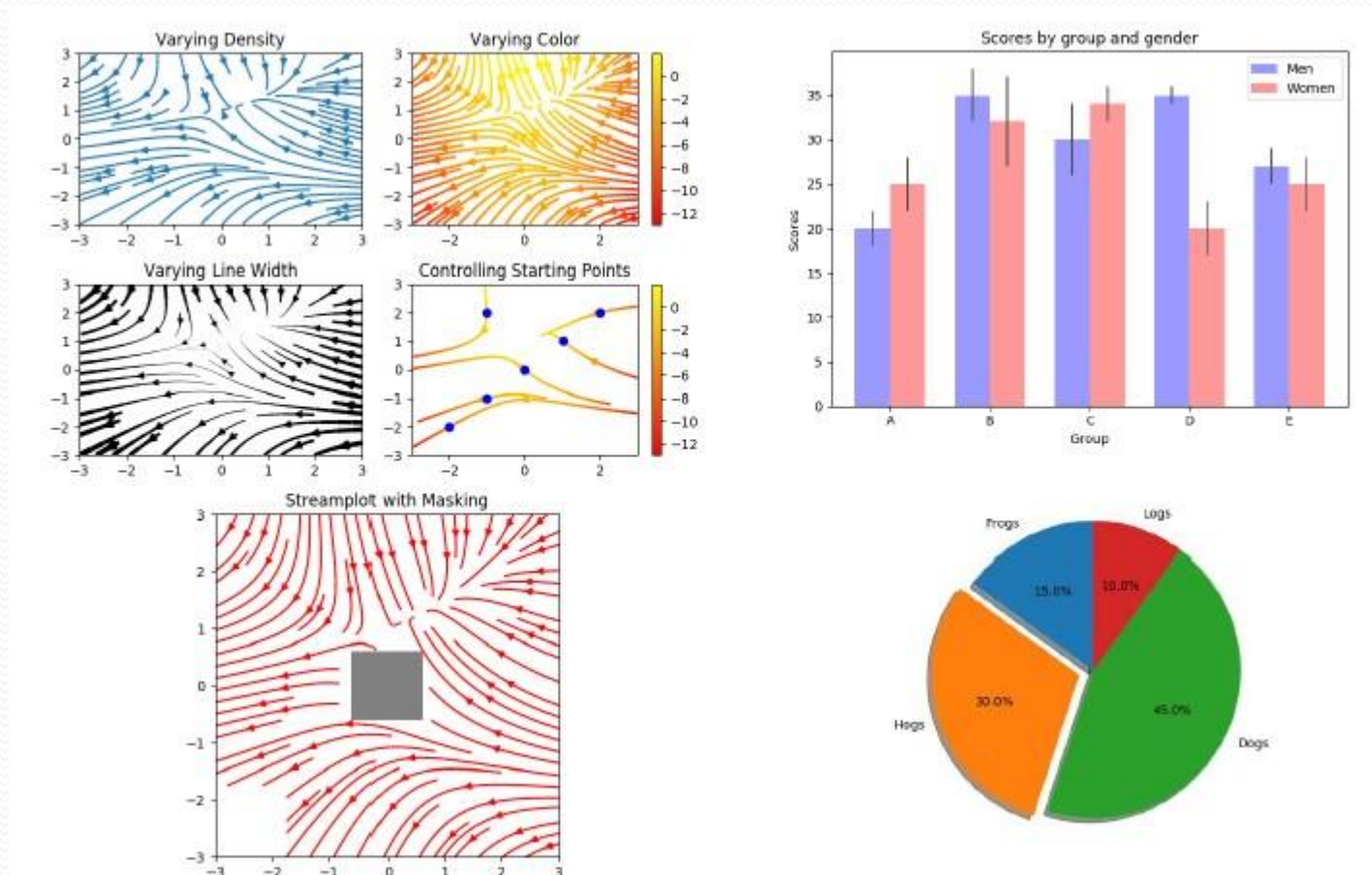
- “matplotlib” là thư viện chuyên về vẽ biểu đồ, mở rộng từ numpy.
- Hỗ trợ rất nhiều loại biểu đồ như biểu đồ dòng, đường, tần suất (histograms), phổ, tương quan, scatterplots,...
- Cài đặt :

**`pip install matplotlib`**

# Giới thiệu về matplotlib

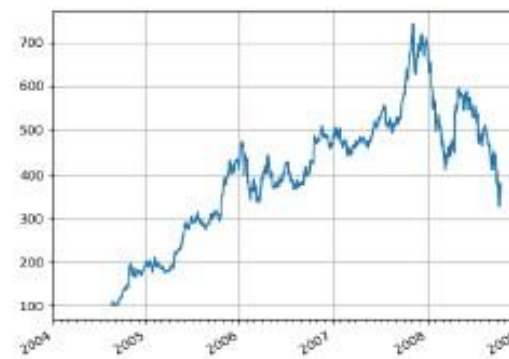
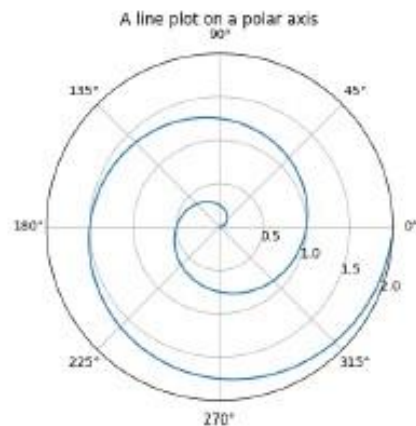
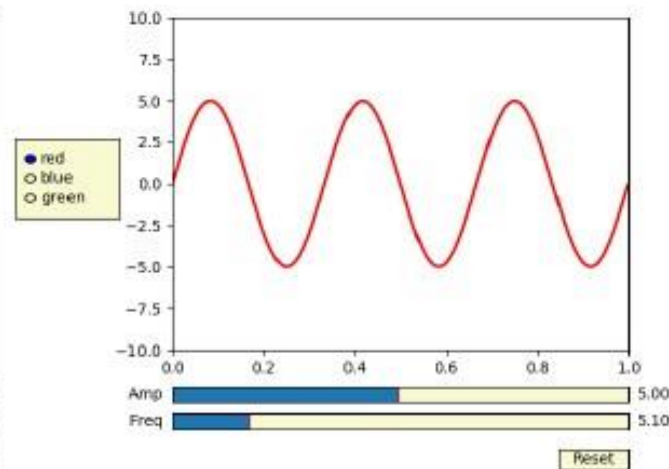
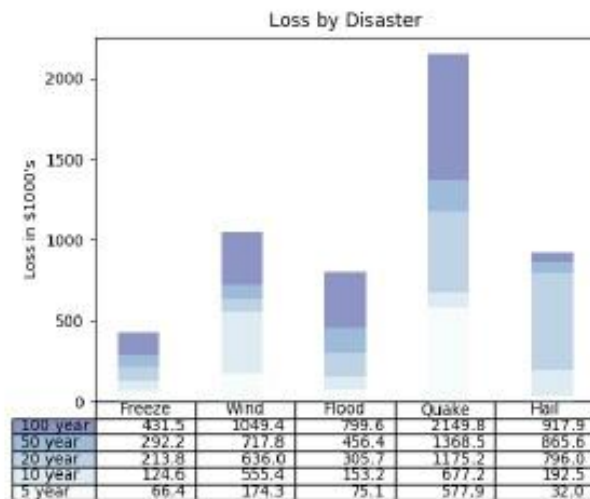


# Giới thiệu về matplotlib



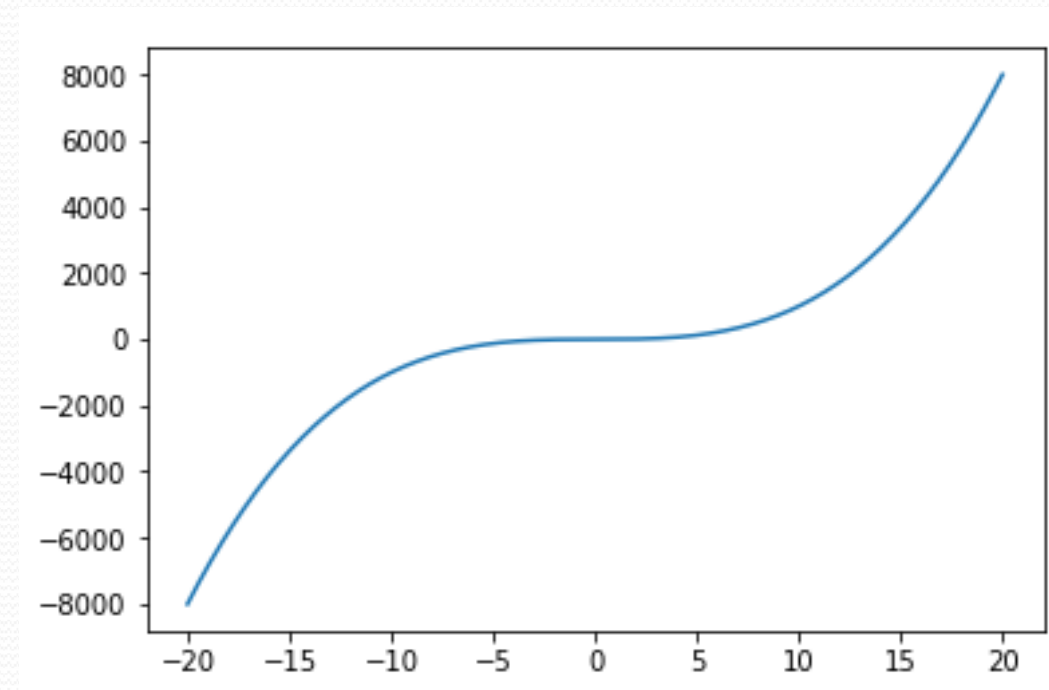


# Giới thiệu về matplotlib



# Các biểu đồ thông dụng

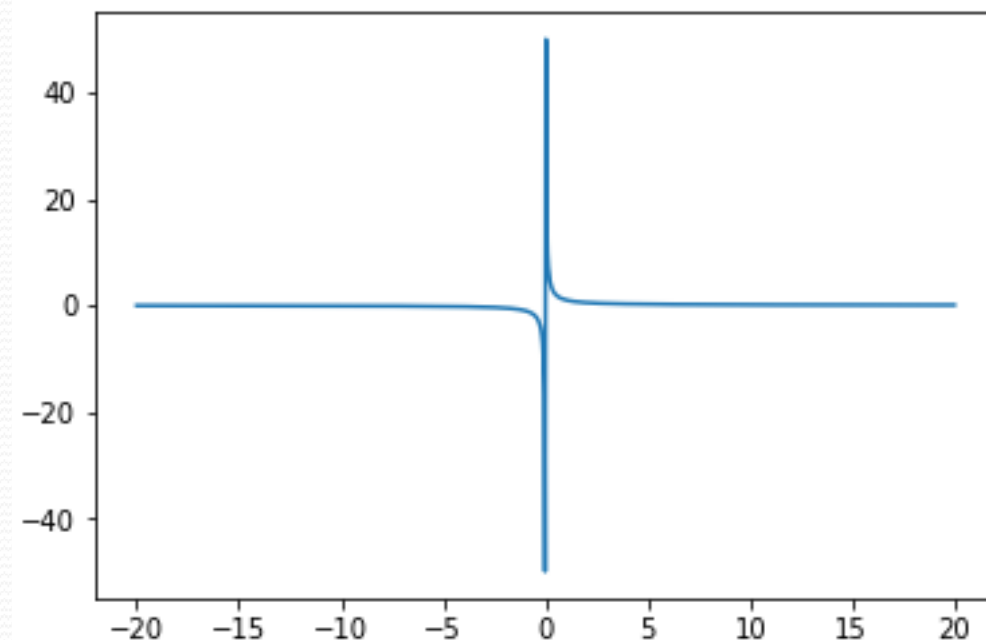
```
import numpy as np # thư viện numpy
import matplotlib.pyplot as plt # thư viện pyplot
# chia đoạn từ -20 đến 20 thành 1000 đoạn
x = np.linspace(-20, 20, 1000)
# tính y
y = x ** 3
# vẽ biểu đồ tương quan giữa x và y
plt.plot(x, y)
# hiển thị biểu đồ
plt.show()
```





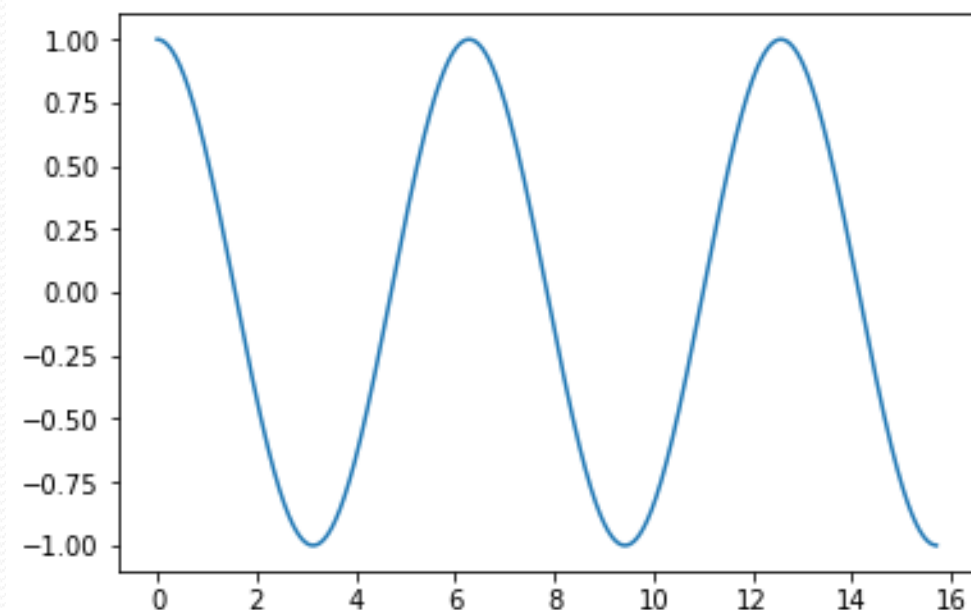
# Các biểu đồ thông dụng

```
import numpy as np # thư viện numpy
import matplotlib.pyplot as plt # thư viện pyplot
# chia đoạn từ -20 đến 20 thành 1000 đoạn
x = np.linspace(-20, 20, 1000)
# tính y
y = x ** -1
# vẽ biểu đồ tương quan giữa x và y
plt.plot(x, y)
# hiển thị biểu đồ
plt.show()
```



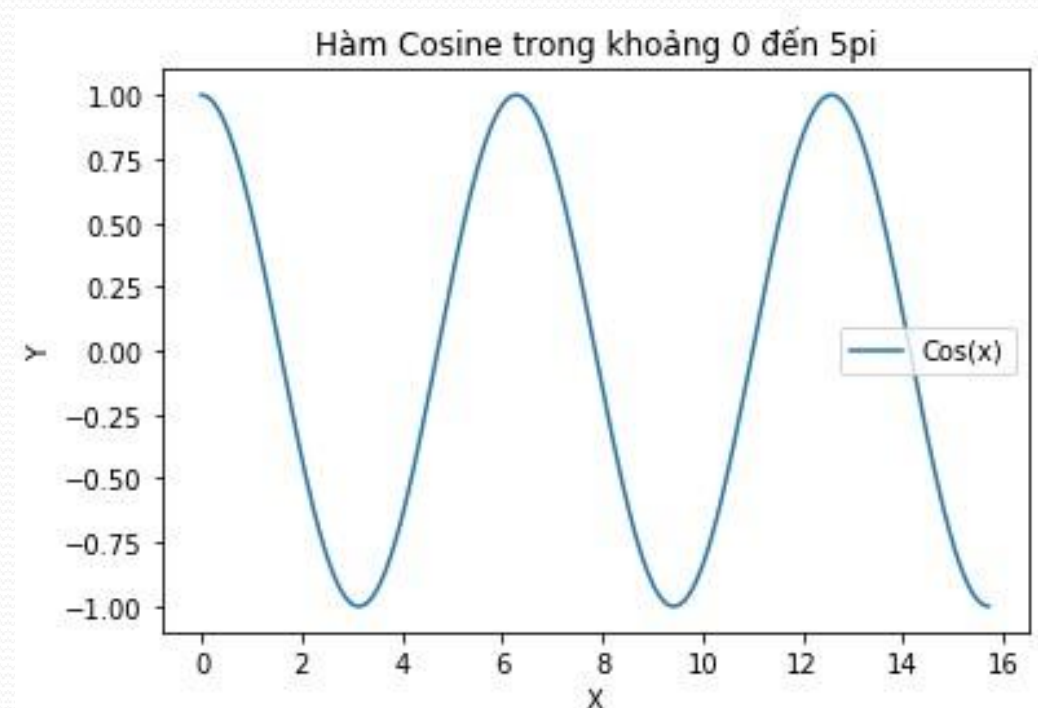
# Các biểu đồ thông dụng

```
import numpy as np # thư viện numpy
import matplotlib.pyplot as plt # thư viện pyplot
# chia đoạn từ 0 đến 5pi thành các đoạn con 0.01
x = np.arange(0, 5 * np.pi, 0.01)
# tính cosine tương ứng với từng phần tử của x
y = np.cos(x)
# vẽ biểu đồ tương quan giữa x và y
plt.plot(x, y)
# hiển thị biểu đồ
plt.show()
```



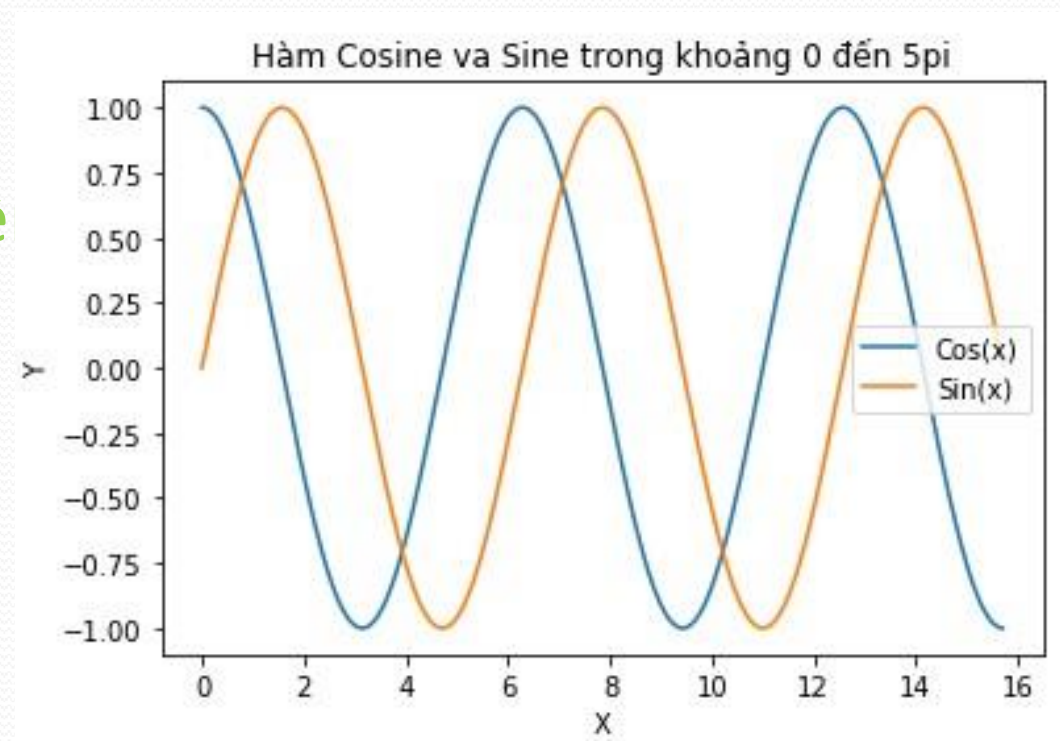
# Các biểu đồ thông dụng

```
import numpy as np # thư viện numpy
import matplotlib.pyplot as plt # thư viện pyplot
x = np.arange(0, 5 * np.pi, 0.01)
y = np.cos(x)
plt.plot(x, y)
# các thông tin bổ sung cho biểu đồ
plt.xlabel('X')
plt.ylabel('Y')
plt.title('Hàm Cosine trong khoảng 0 đến 5pi')
plt.legend(['Cos(x)'])
# hiển thị biểu đồ
plt.show()
```



# Các biểu đồ thông dụng

- `import numpy as np` # thư viện numpy
- `import matplotlib.pyplot as plt` # thư viện
- `5 * np.pi, 0.01)`
- `y = np.cos(x)` `z = np.sin(x)` `plt.plot(x,`  
`y)`
- `plt.plot(x, z)`
- `plt.xlabel('X')`
- `plt.ylabel('Y')`
- `plt.title('Hàm Cosine và Sine trong khoảng 0 đến 5pi')`
- `plt.legend(['Cos(x)', 'Sin(x)'])`





# Các biểu đồ thông dụng

- Điều kiện cần: đã có sẵn dữ liệu
- Có thể có 4 bước cơ bản:
  1. Chọn loại biểu đồ phù hợp
    - Tùy thuộc rất nhiều vào loại dữ liệu
    - Tùy thuộc vào mục đích sử dụng của người dùng
  2. Thiết lập các thông số cho biểu đồ
    - Thông số của các trục, ý nghĩa, tỉ lệ chia,...
    - Các điểm nhấn trên bản đồ
    - Góc nhìn, mẫu tô, màu và các chi tiết khác
    - Các thông tin bổ sung
  3. Vẽ biểu đồ
  4. Lưu ra file

# Biểu đồ dạng đường (line plot)

- Biểu đồ thể hiện tương quan giữa X và Y
- Cú pháp:
  - `plot([x], y, [fmt], data=None, **kwargs)`
  - `plot([x], y, [fmt], [x2], y2, [fmt2], ..., **kwargs)`
- “fmt” là quy cách vẽ đường
- “data” là nhãn của dữ liệu
- `**kwargs`: tham số vẽ đường
- Vẽ nhiều lần trên một biểu đồ
- Kết quả trả về là một list các đối tượng Line2D

[https://matplotlib.org/3.1.0/api/\\_as\\_gen/matplotlib.pyplot.plot.html](https://matplotlib.org/3.1.0/api/_as_gen/matplotlib.pyplot.plot.html)



# Biểu đồ dạng đường (line plot)

- “fmt” gồm 3 phần `fmt = '[color][marker][line]'`
- `[color]` –viết tắt tên màu:
  - ‘b’ –blue
  - ‘g’ –green
  - ‘r’ –red
  - ‘c’ –cyan
  - ‘m’ –magenta
  - ‘y’ –yellow
  - ‘b’ –black
  - ‘w’ –white
  - `#rrggbb` –chỉ ra mã màu theo hệ RGB

# Biểu đồ dạng đường (line plot)

□ [marker] –cách đánh dấu dữ liệu:

- ‘o’ –hình tròn
- ‘v’ –tam giác xuống (‘^’, ‘<’, ‘>’)
- ‘\*’ –ngôi sao
- ‘.’ –chấm
- ‘p’ –ngũ giác

□ [line] –cách vẽ đường:

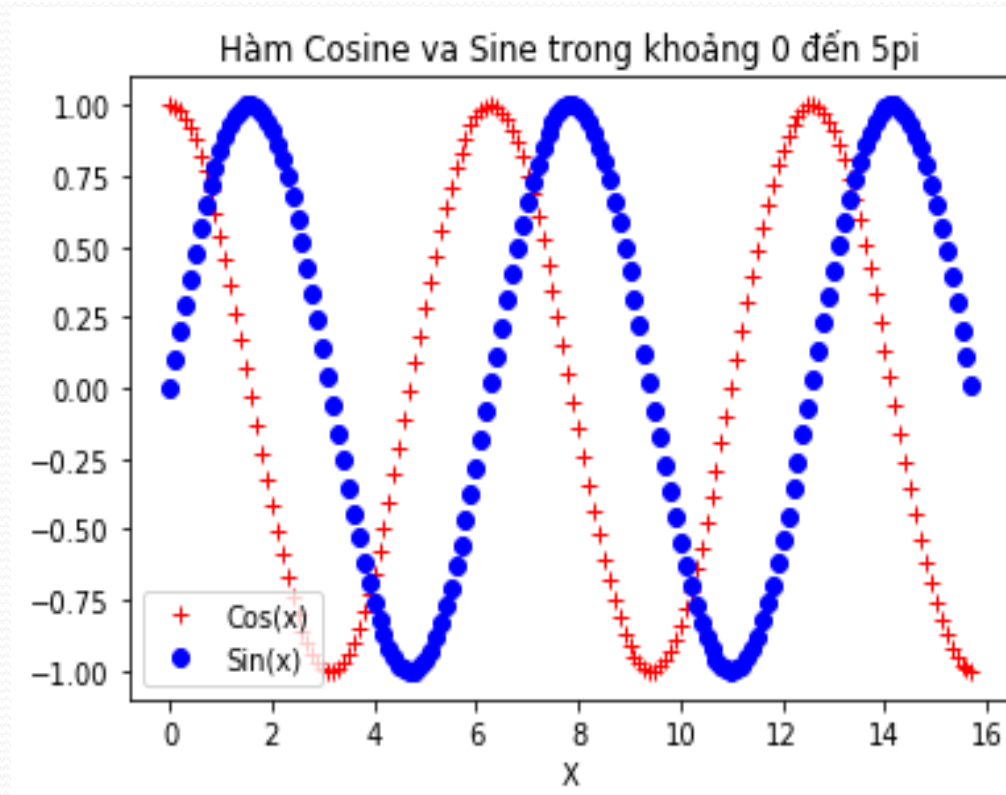
- ‘-’ –nét liền
- ‘--’ –nét đứt
- ‘-.’ –gạch chấm
- ‘:’ –đường chấm

# Biểu đồ dạng đường (line plot)

- `plot(x, y)`      # plot x and y using default line style and color
- `plot(x, y, 'bo')` # plot x and y using blue circle markers
- `plot(y)`          # plot y using x as index array 0..N-1
- `plot(y, 'r+')`    # ditto, but with red plusses

# Biểu đồ dạng đường (line plot)

```
import numpy as np # thư viện numpy
import matplotlib.pyplot as plt # thư viện pyplot
x = np.arange(0, 5 * np.pi, 0.1)
y = np.cos(x)
z = np.sin(x)
plt.plot(x, y, 'r+')
plt.plot(x, z, 'bo')
plt.xlabel('X')
plt.ylabel('Y')
plt.title('Hàm Cosine và Sine trong khoảng 0 đến 5pi')
plt.legend(['Cos(x)', 'Sin(x)'])
# hiển thị biểu đồ
plt.show()
```

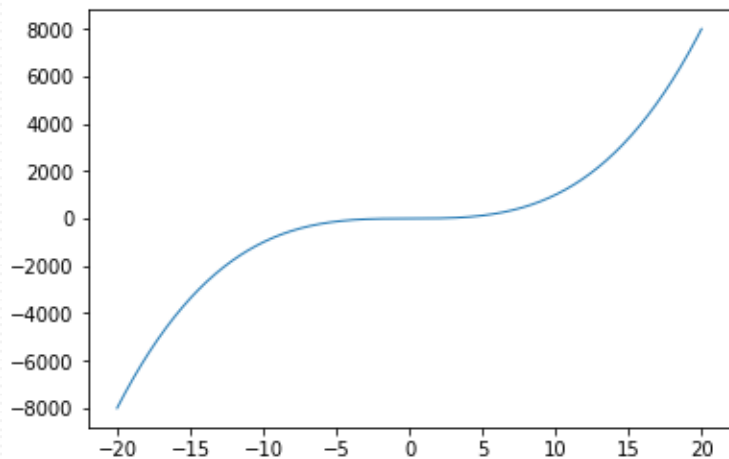


# Biểu đồ dạng đường (line plot)

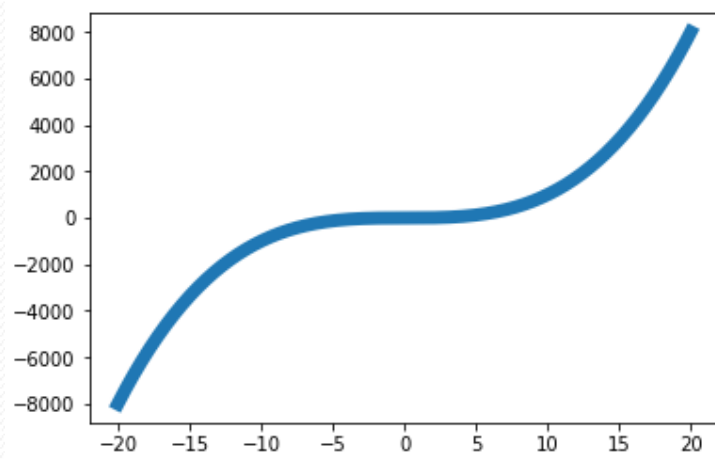
- `plot(x, y, 'go--', linewidth=2, markersize=12)`
- `plot(x, y, color='green', marker='o', linestyle='dashed', ... linewidth=2, markersize=12)`

```
import numpy as np # thư viện numpy
import matplotlib.pyplot as plt # thư viện pyplot
# chia đoạn từ -20 đến 20 thành 1000 đoạn
x = np.linspace(-20, 20, 1000)
# tính y
y = x ** 3
# vẽ biểu đồ tương quan giữa x và y
plt.plot(x, y, linewidth=7)
# hiển thị biểu đồ
plt.show()
```

linewidth=1



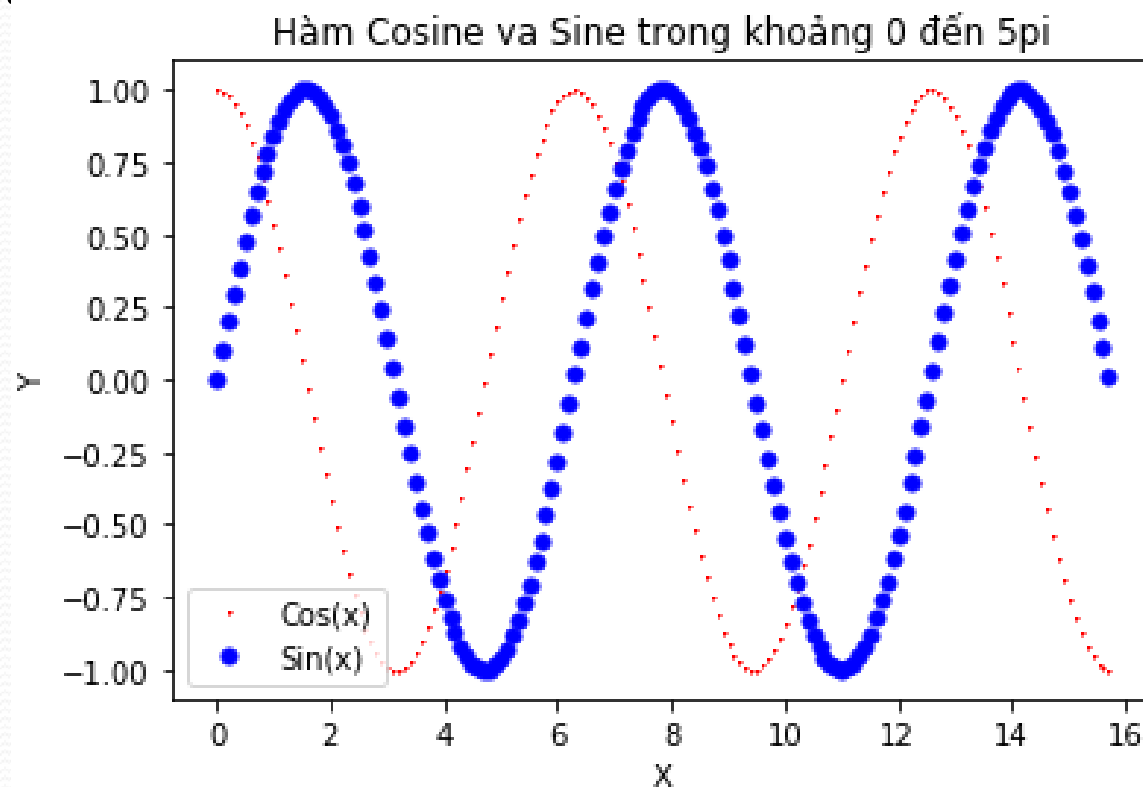
linewidth=7





# Biểu đồ dạng đường (line plot)

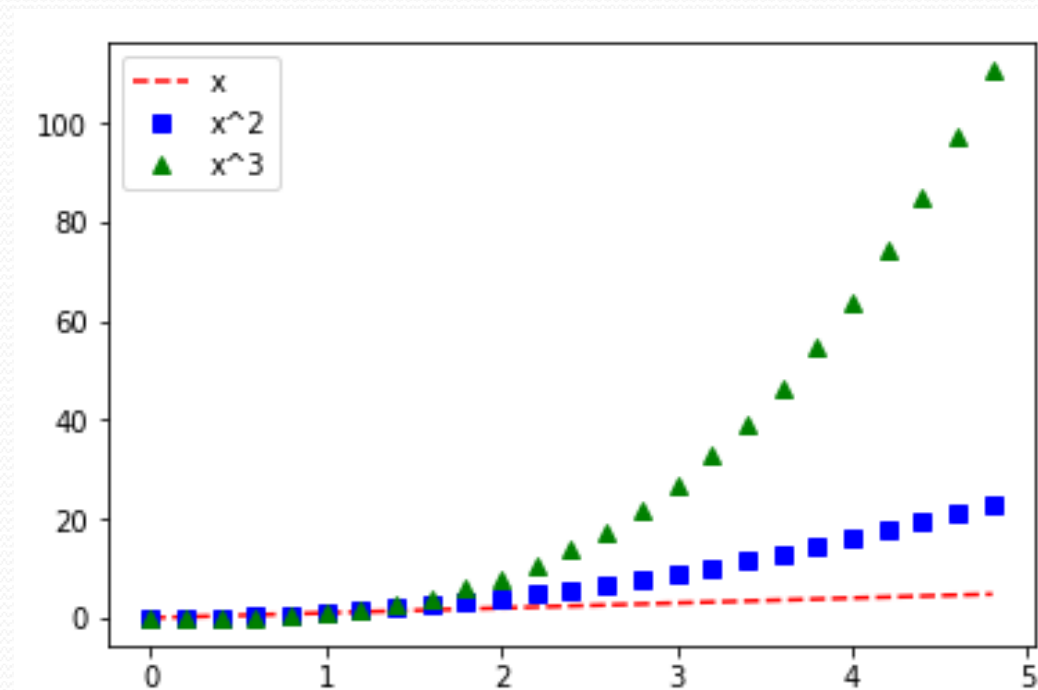
```
import numpy as np # thư viện numpy
import matplotlib.pyplot as plt # thư viện pyplot
x = np.arange(0, 5 * np.pi, 0.1)
y = np.cos(x)
z = np.sin(x)
plt.plot(x, y, 'r+', markersize=1)
plt.plot(x, z, 'bo', markersize=5)
plt.xlabel('X')
plt.ylabel('Y')
plt.title('Hàm Cosine và Sine trong khoảng 0 đến 5pi')
plt.legend(['Cos(x)', 'Sin(x)'])
# hiển thị biểu đồ
plt.show()
```





# Biểu đồ dạng đường (line plot)

```
import numpy as np
import matplotlib.pyplot as plt
# chia đoạn 0-5 thành các bước 0.2
x = np.arange(0., 5., 0.2)
# Vẽ 3 đường:
# -màu đỏ nét đứt:  $y = x$ 
# -màu xanh dương, đánh dấu ô vuông:  $y = x^2$ 
# -màu xanh lá, đánh dấu tam giác:  $y = x^3$ 
plt.plot(x, x, 'r--', x, x**2, 'bs', x, x**3, 'g^')
plt.legend(['x', 'x^2', 'x^3'])
plt.show()
```



# Biểu đồ dạng cột (bar plot)

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

```
D = {'CNTT': 500,
```

```
'Toan': 310,
```

```
'Hoa': 150,
```

```
'Sinh': 280,
```

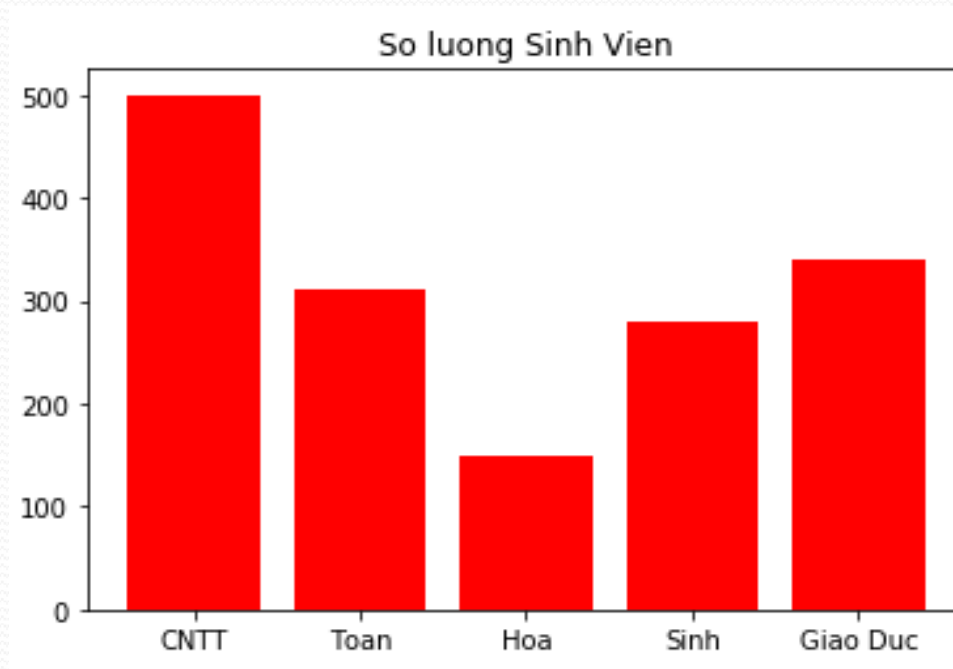
```
'Giao Duc': 340}
```

```
plt.bar(range(len(D)), list(D.values()),color='r')
```

```
plt.xticks(range(len(D)), D.keys())
```

```
plt.title('So luong Sinh Vien')
```

```
plt.show()
```



# Biểu đồ dạng cột (bar plot)

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

```
D = {'CNTT': 500,
```

```
'Toan': 310,
```

```
'Hoa': 150,
```

```
'Sinh': 280,
```

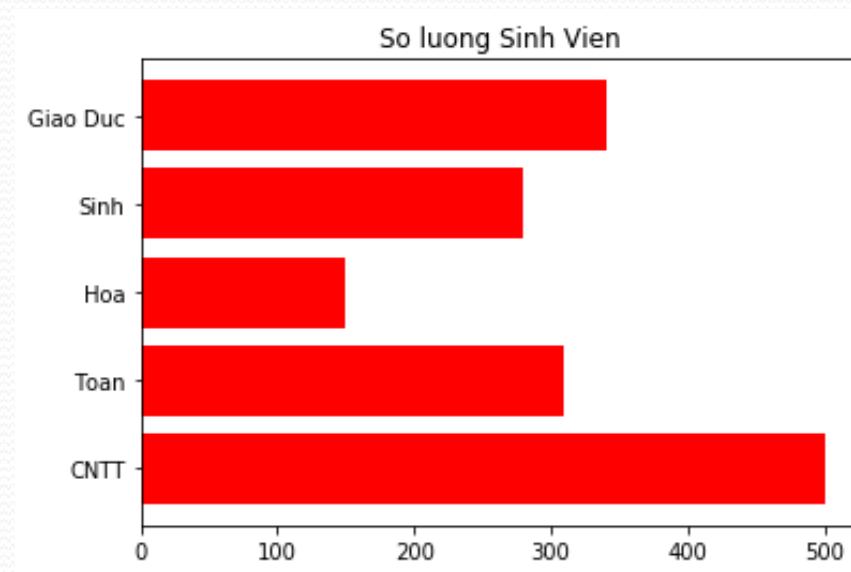
```
'Giao Duc': 340}
```

```
plt.barh(range(len(D)), list(D.values()),color='r')
```

```
plt.yticks(range(len(D)), D.keys())
```

```
plt.title('So luong Sinh Vien')
```

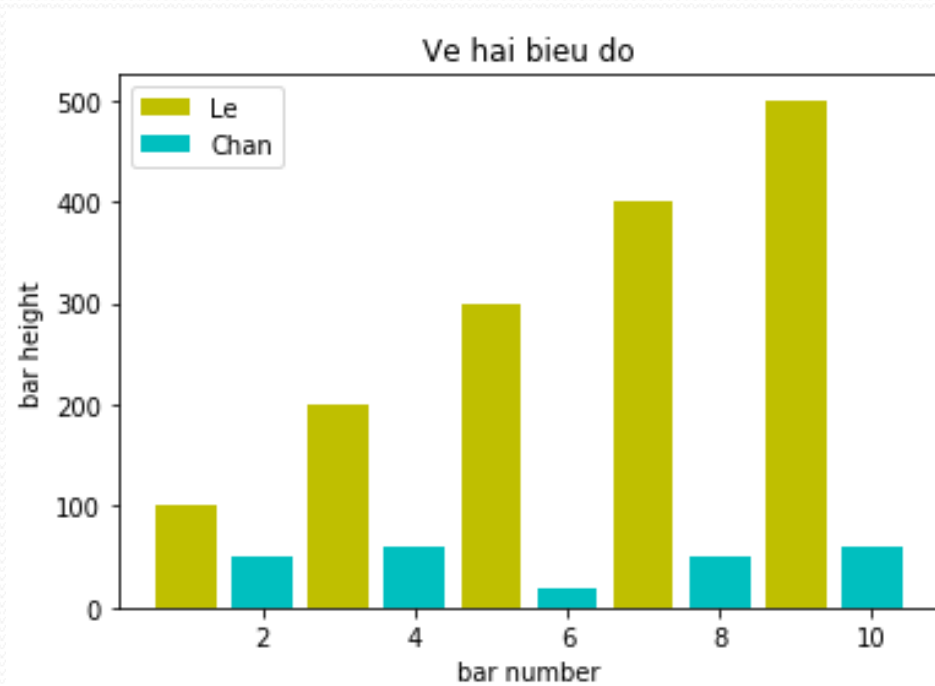
```
plt.show()
```



**barh()**: Plot a horizontal bar plot.

# Biểu đồ dạng cột (bar plot)

```
import matplotlib.pyplot as plt
plt.bar([1,3,5,7,9],[100,200,300,400,500], label="Le",color='y')
plt.bar([2,4,6,8,10],[50,60,20,50,60], label="Chan", color='c')
plt.legend()
plt.xlabel('bar number')
plt.ylabel('bar height')
plt.title('Ve hai bieu do')
plt.show()
```



# Biểu đồ bánh (pie chart)

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

```
D = {'CNTT': 500,
```

```
'Toan': 310,
```

```
'Hoa': 150,
```

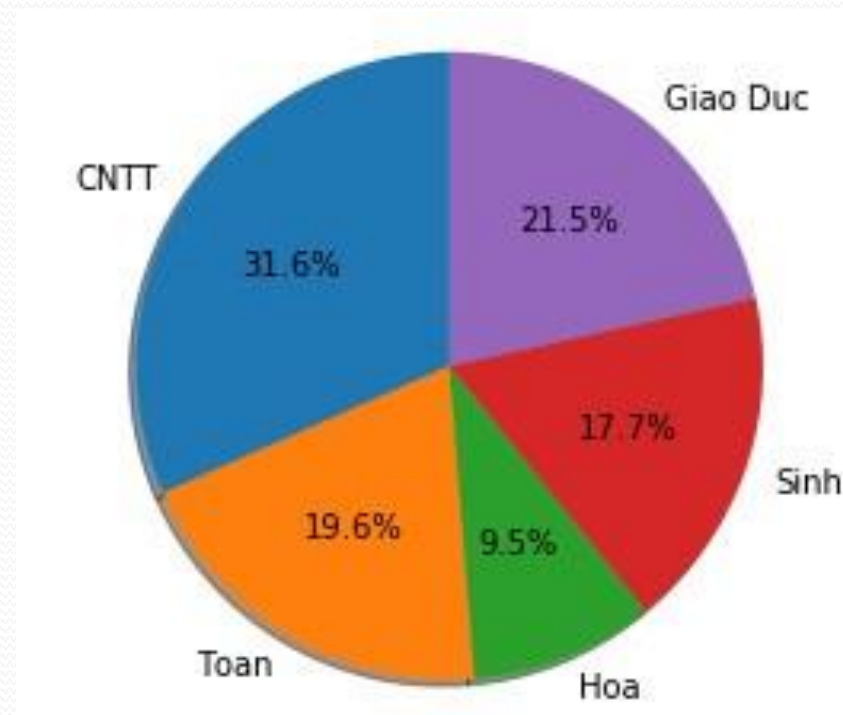
```
'Sinh': 280,
```

```
'Giao Duc': 340}
```

```
plt.pie(D.values(), labels=D.keys(), autopct='%1.1f%%', shadow=True, startangle=90)
```

```
plt.axis('equal')# trục x = trục y
```

```
plt.show()
```





# Biểu đồ bánh (pie chart)

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

```
D = {'CNTT': 500,
```

```
'Toan': 310,
```

```
'Hoa': 150,
```

```
'Sinh': 280,
```

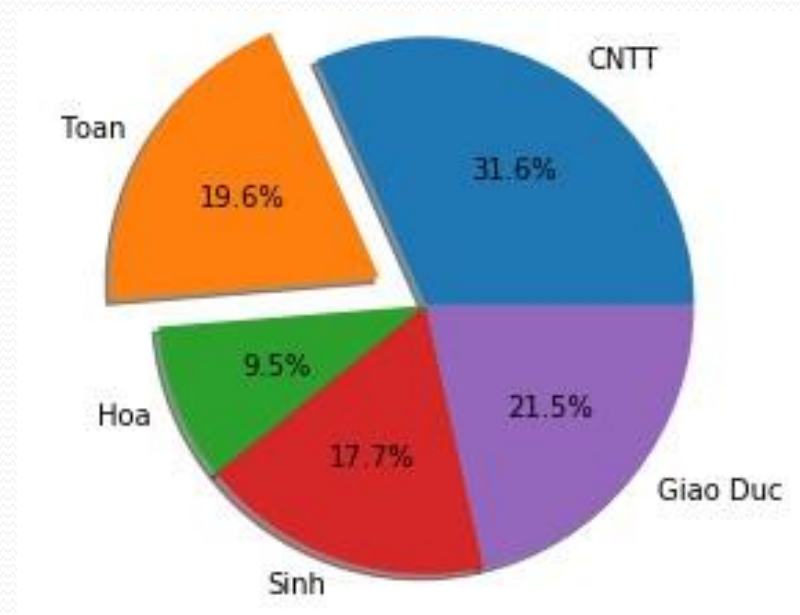
```
'Giao Duc': 340}
```

```
explode = (0, 0.2, 0, 0, 0) # only "explode" the 2nd slice (i.e. 'Toan')
```

```
plt.pie(D.values(),explode=explode, labels=D.keys(), autopct='%1.1f%%', shadow=True, startangle=0)
```

```
plt.axis('equal')# trục x = trục y
```

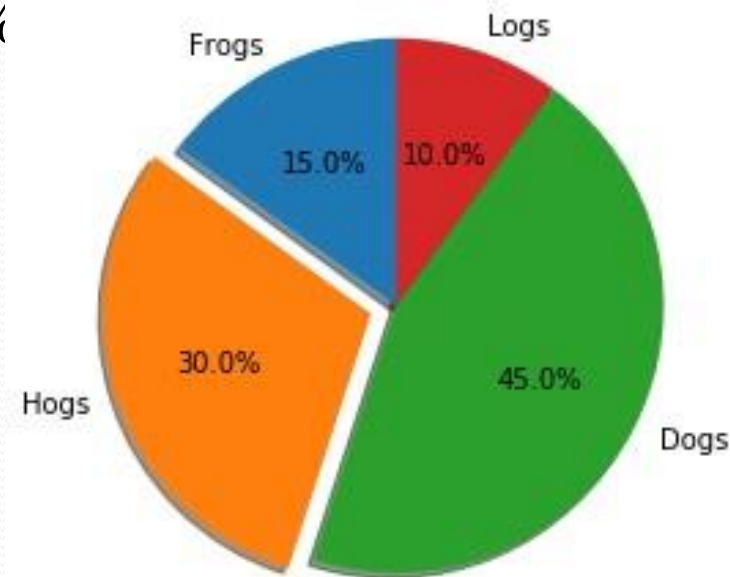
```
plt.show()
```





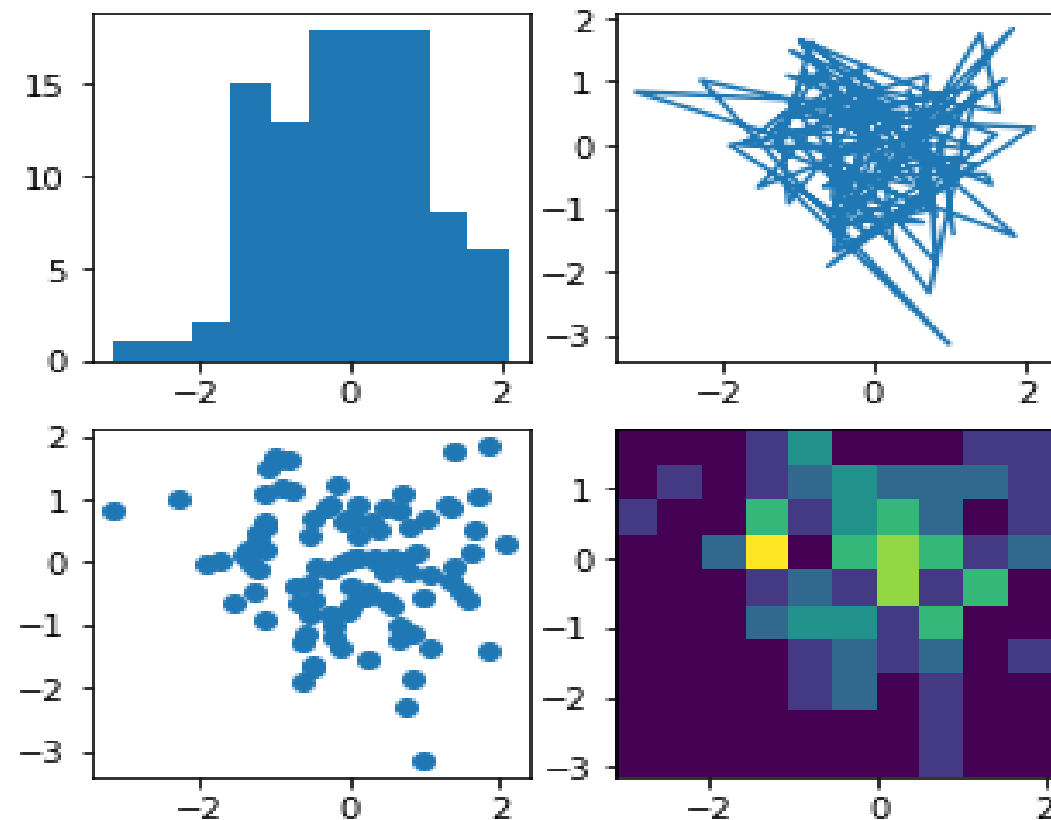
# Biểu đồ bánh (pie chart)

```
import matplotlib.pyplot as plt
# Pie chart, where the slices will be ordered and plotted counter-clockwise:
labels = 'Frogs', 'Hogs', 'Dogs', 'Logs'
sizes = [15, 30, 45, 10]
explode = (0, 0.1, 0, 0) # only "explode" the 2nd slice (i.e. 'Hogs')
plt.pie(sizes, explode=explode, labels=labels, autopct='%1.1f%%',
        shadow=True, startangle=90)
plt.axis('equal')
plt.show()
```



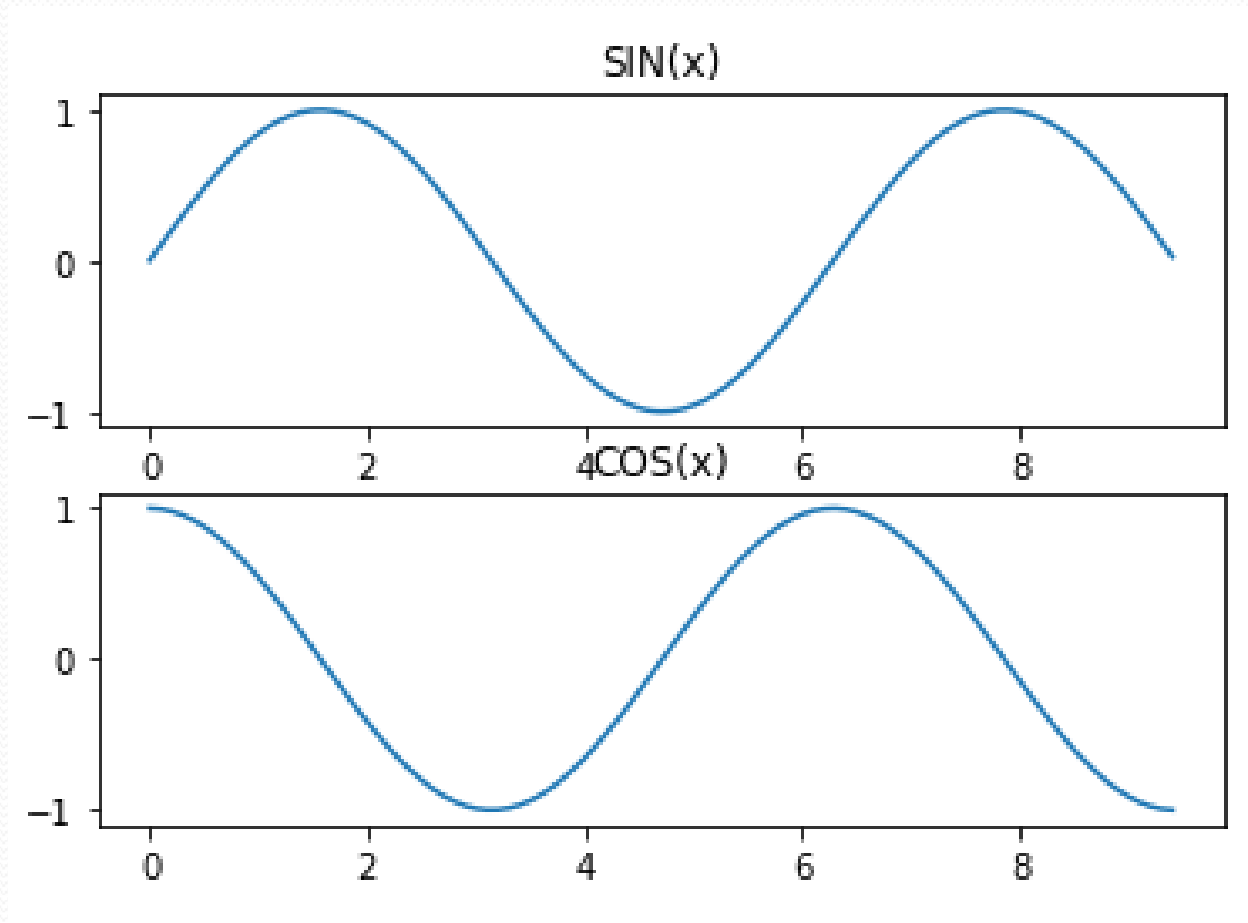
# Chia thành các biểu đồ con

```
import matplotlib.pyplot as plt
import numpy as np
np.random.seed(19680801)
data = np.random.randn(2, 100)
fig, axs = plt.subplots(2, 2, figsize=(5, 5))
axs[0, 0].hist(data[0])
axs[1, 0].scatter(data[0], data[1])
axs[0, 1].plot(data[0], data[1])
axs[1, 1].hist2d(data[0], data[1])
plt.show()
```



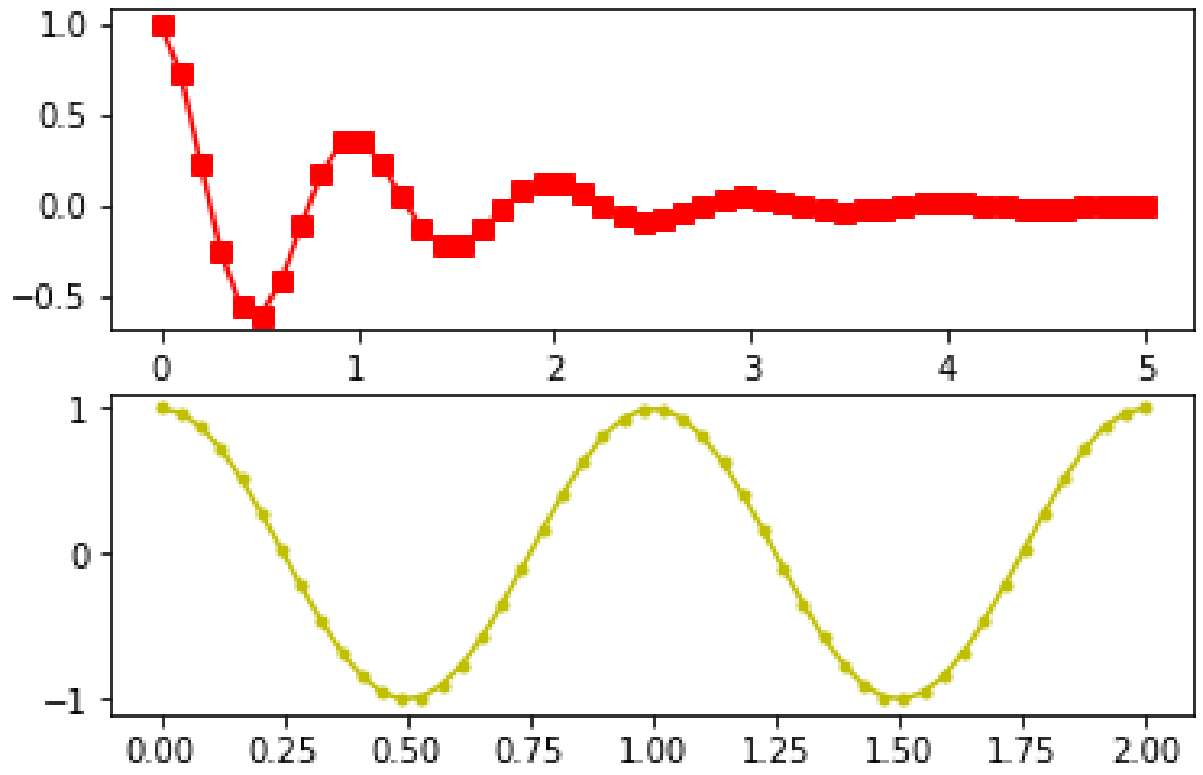
# Chia thành các biểu đồ con

```
import numpy as np
import matplotlib.pyplot as plt
x = np.arange(0, 3 * np.pi, 0.1)
y_sin = np.sin(x)
y_cos = np.cos(x)
plt.subplot(2, 1, 1) # biểu đồ 1
plt.plot(x, y_sin)
plt.title('SIN(x)')
plt.subplot(2, 1, 2) # biểu đồ 2
plt.plot(x, y_cos)
plt.title('COS(x)')
plt.show()
```



# Chia thành các biểu đồ con

```
import numpy as np
import matplotlib.pyplot as plt
x1 = np.linspace(0.0, 5.0)
x2 = np.linspace(0.0, 2.0)
y1 = np.cos(2 * np.pi * x1) * np.exp(-x1)
y2 = np.cos(2 * np.pi * x2)
plt.subplot(2, 1, 1)
plt.plot(x1, y1, 'rs-')
plt.subplot(2, 1, 2)
plt.plot(x2, y2, 'y.-')
plt.show()
```



# Tham khảo

- <https://matplotlib.org/>
- <https://matplotlib.org/stable/tutorials/index>
- [https://phamdinhkhanh.github.io/deepai-book/ch\\_appendix/appendix\\_matplotlib.html](https://phamdinhkhanh.github.io/deepai-book/ch_appendix/appendix_matplotlib.html)