

# LẬP TRÌNH TRÍ TUỆ NHÂN TẠO

THS. VƯƠNG XUÂN CHÍ

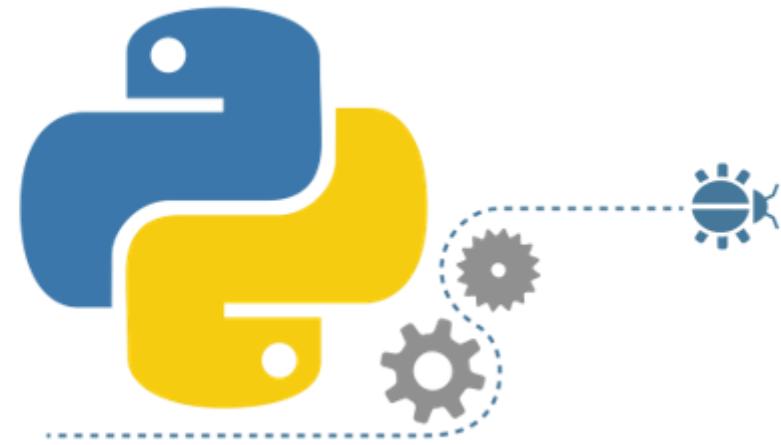
VXCHI@NTT.EDU.VN

0903 270 567



# CHƯƠNG 8

---



# MATPLOLIB



# Nội dung

- 1. Giới thiệu và cài đặt matplotlib*
- 2. Vẽ biểu đồ đơn giản*
- 3. Một số loại biểu đồ thông dụng trong matplotlib*
  - 3.1 Biểu đồ dạng cột (bar plot)*
  - 3.2 Biểu đồ đường (line chart)*
  - 3.3 Biểu đồ bánh (pie chart)*
- 4. Một số chức năng hữu ích*



# **Giới thiệu và cài đặt matplotlib**



# Giới thiệu matplotlib

- “**matplotlib**” là thư viện chuyên về vẽ biểu đồ, mở rộng từ [numpy](#)
- Có mục tiêu đơn giản hóa tối đa công việc vẽ biểu đồ để “chỉ cần vài dòng lệnh”
- Hỗ trợ rất nhiều loại biểu đồ, đặc biệt là các loại được sử dụng trong nghiên cứu hoặc kinh tế như biểu đồ dòng, đường, tần suất ([histograms](#)), phổ, tương quan, [errorcharts](#), [scatterplots](#),...
- Cấu trúc của [matplotlib](#) gồm nhiều phần, phục vụ cho các mục đích sử dụng khác nhau.



# Giới thiệu matplotlib

- Ngoài các [API](#) liên quan đến vẽ biểu đồ, matplotlib còn bao gồm một số interface: [Object-Oriented API](#), [The Scripting Interface \(pyplot\)](#), [The MATLAB Interface \(pylab\)](#)
  - ✓ Các interface này giúp chúng ta thuận tiện trong việc thiết lập chỉ số trước khi thực hiện vẽ biểu đồ Interface pylab hiện đã không còn được phát triển
  - ✓ Hầu hết các ví dụ trong slide này đều sử dụng pyplot
  - ✓ Sử dụng [Object-Oriented API](#) hoặc trực tiếp các API của matplotlib sẽ cho phép can thiệp sâu hơn vào việc vẽ biểu đồ (hầu hết project sẽ không có nhu cầu này).

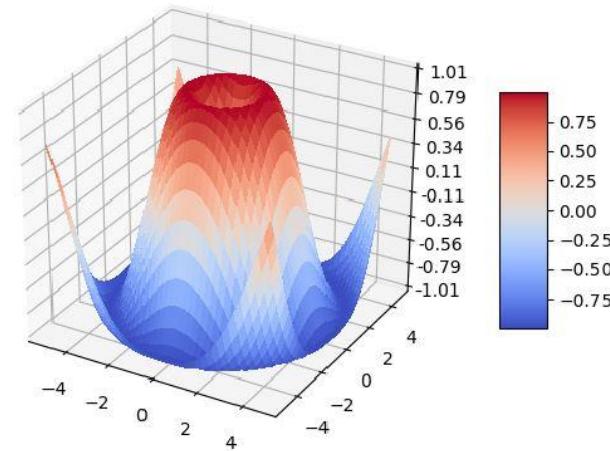
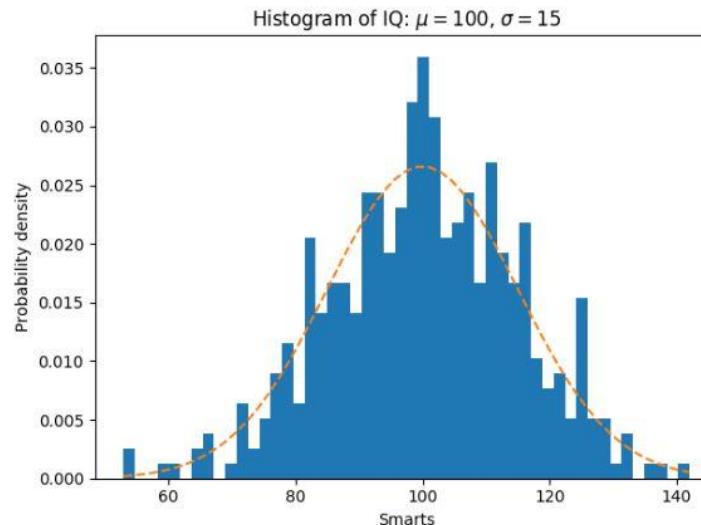
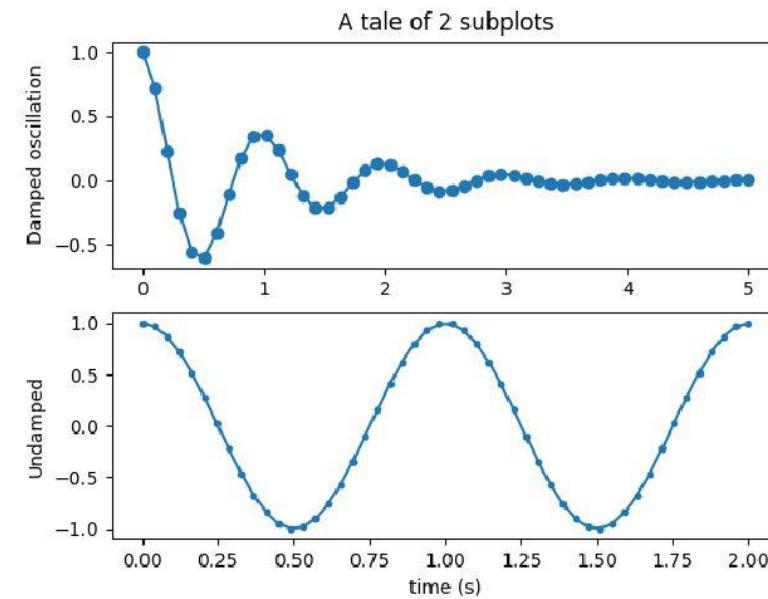
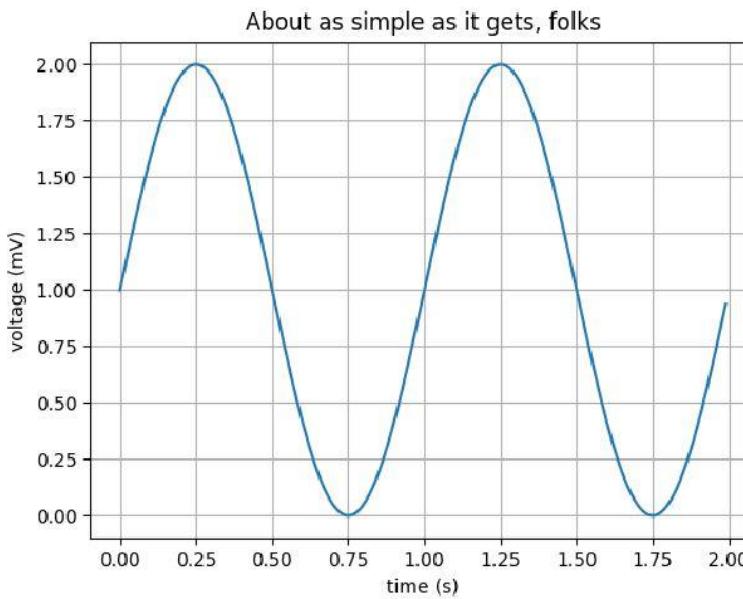


# Cài đặt: “pip install matplotlib”

```
Command Prompt
C:\Users\ADMIN>pip install matplotlib
Collecting matplotlib
  Downloading matplotlib-3.5.1-cp39-cp39-win_amd64.whl (7.2 MB)
    |████████| 7.2 MB 3.3 MB/s
Collecting packaging>=20.0
  Downloading packaging-21.3-py3-none-any.whl (40 kB)
    |████████| 40 kB 2.7 MB/s
Collecting pillow>=6.2.0
  Downloading Pillow-9.0.0-cp39-cp39-win_amd64.whl (3.2 MB)
    |████████| 3.2 MB 3.2 MB/s
Collecting fonttools>=4.22.0
  Downloading fonttools-4.29.0-py3-none-any.whl (895 kB)
    |████████| 895 kB 6.4 MB/s
Collecting pyparsing>=2.2.1
  Downloading pyparsing-3.0.7-py3-none-any.whl (98 kB)
    |████████| 98 kB 2.4 MB/s
Collecting python-dateutil>=2.7
  Downloading python_dateutil-2.8.2-py2.py3-none-any.whl (247 kB)
    |████████| 247 kB 6.4 MB/s
Requirement already satisfied: numpy>=1.17 in c:\users\admin\appdata\local\programs\python\python39\lib\site-packages (from matplotlib) (1.22.1)
Collecting cycler>=0.10
  Downloading cycler-0.11.0-py3-none-any.whl (6.4 kB)
Collecting kiwisolver>=1.0.1
  Downloading kiwisolver-1.3.2-cp39-cp39-win_amd64.whl (52 kB)
    |████████| 52 kB 314 kB/s
Collecting six>=1.5
  Downloading six-1.16.0-py2.py3-none-any.whl (11 kB)
Installing collected packages: six, pyparsing, python-dateutil, pillow, packaging, kiwisolver, fonttools, cycler, matplotlib
Successfully installed cycler-0.11.0 fonttools-4.29.0 kiwisolver-1.3.2 matplotlib-3.5.1 packaging-21.3 pillow-9.0.0 pyparsing-3.0.7 python-dateutil-2.8.2 six-1.16.0
WARNING: You are using pip version 21.1.1; however, version 22.0.2 is available.
You should consider upgrading via the 'c:\users\admin\appdata\local\programs\python\python39\python.exe -m pip install --upgrade pip' command.
C:\Users\ADMIN>
```

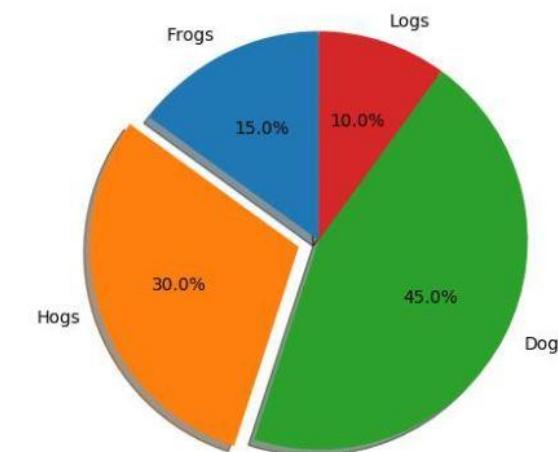
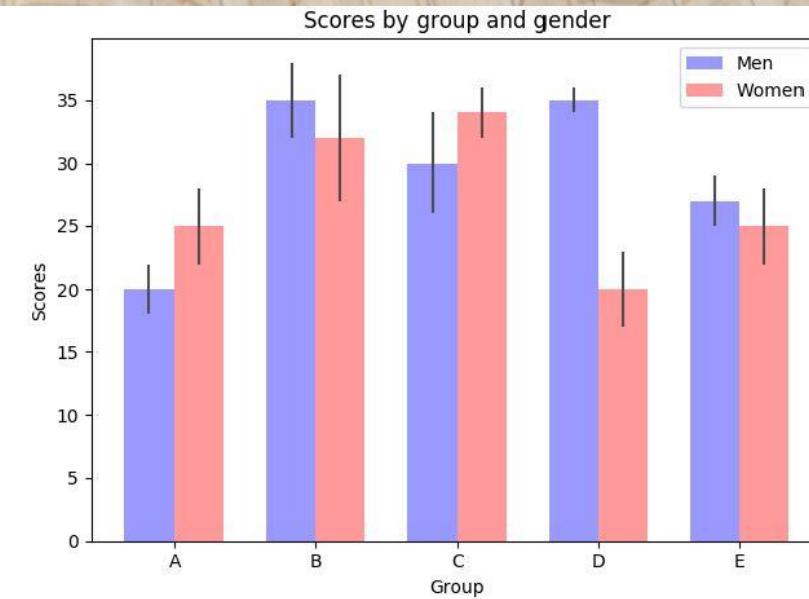
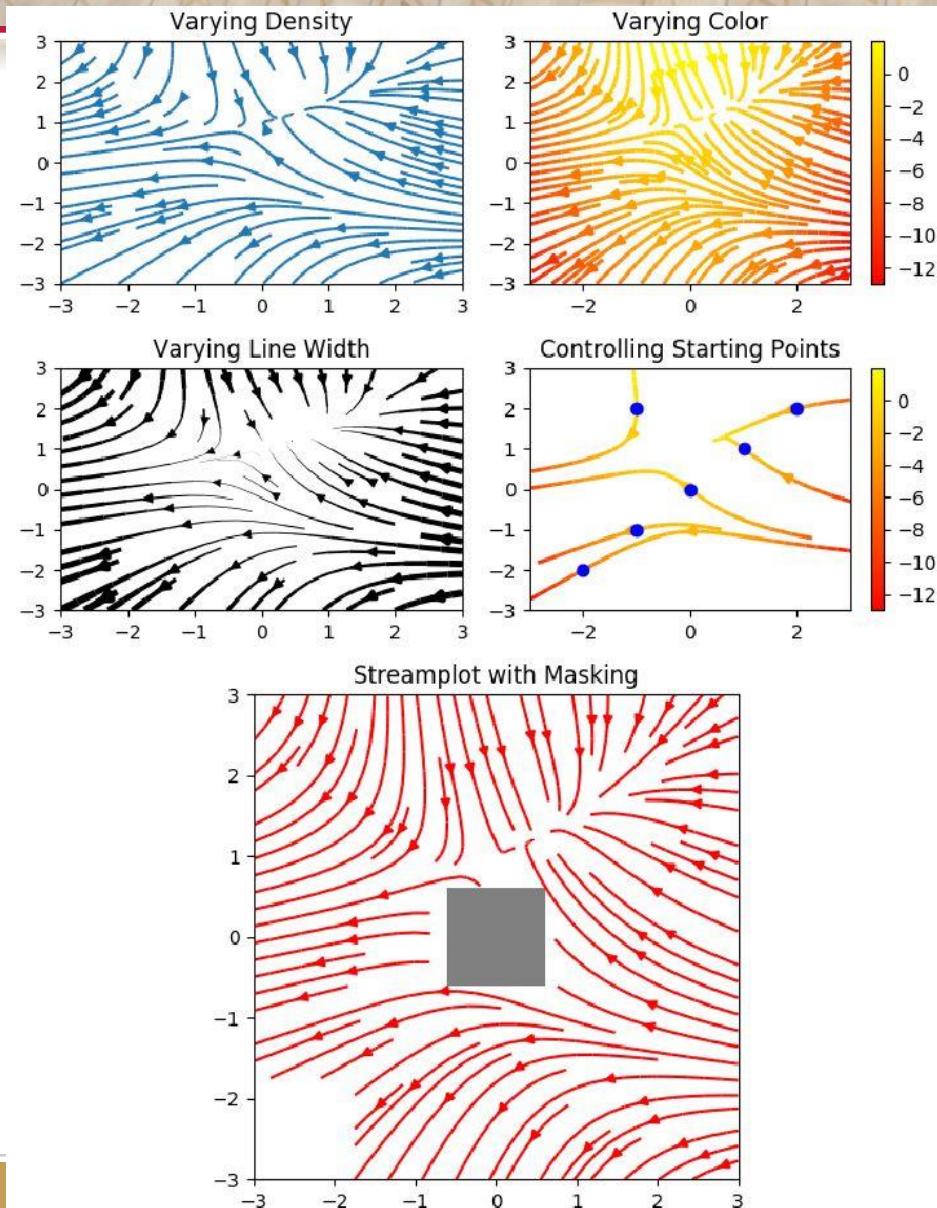


# Một số biểu đồ vẽ bằng matplotlib



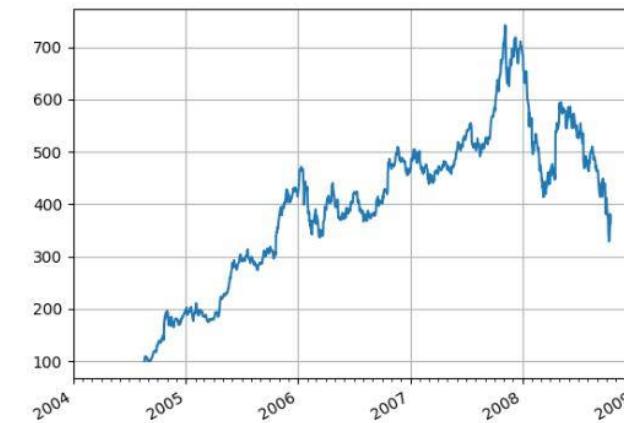
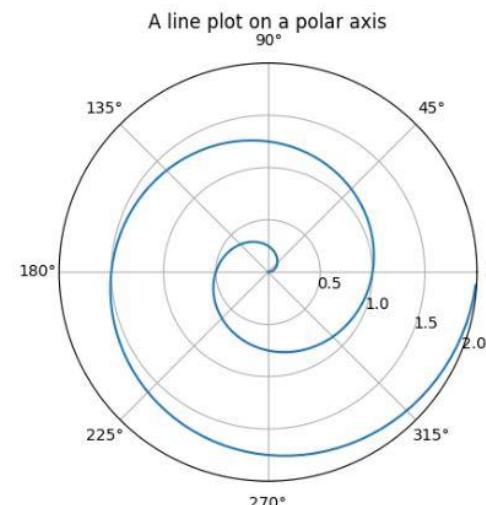
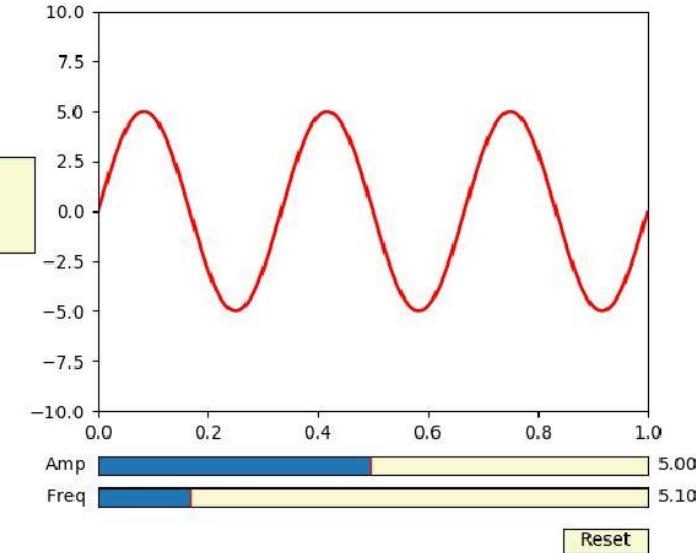
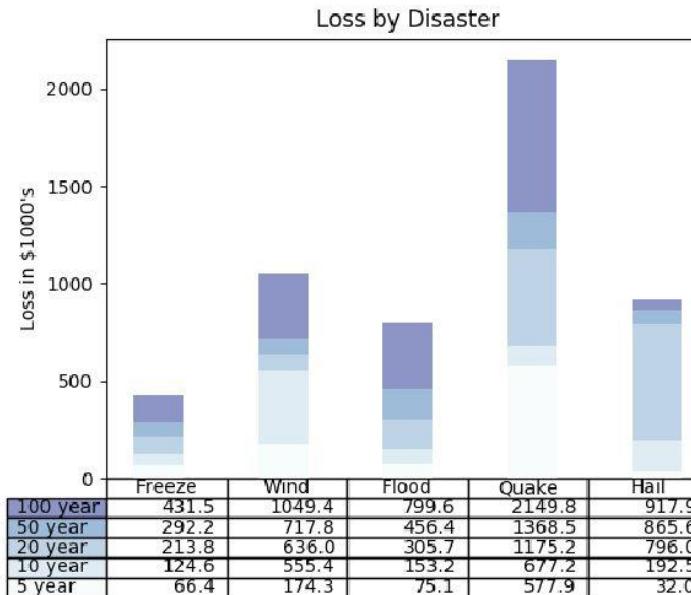


# Một số biểu đồ vẽ bằng matplotlib





# Một số biểu đồ vẽ bằng matplotlib



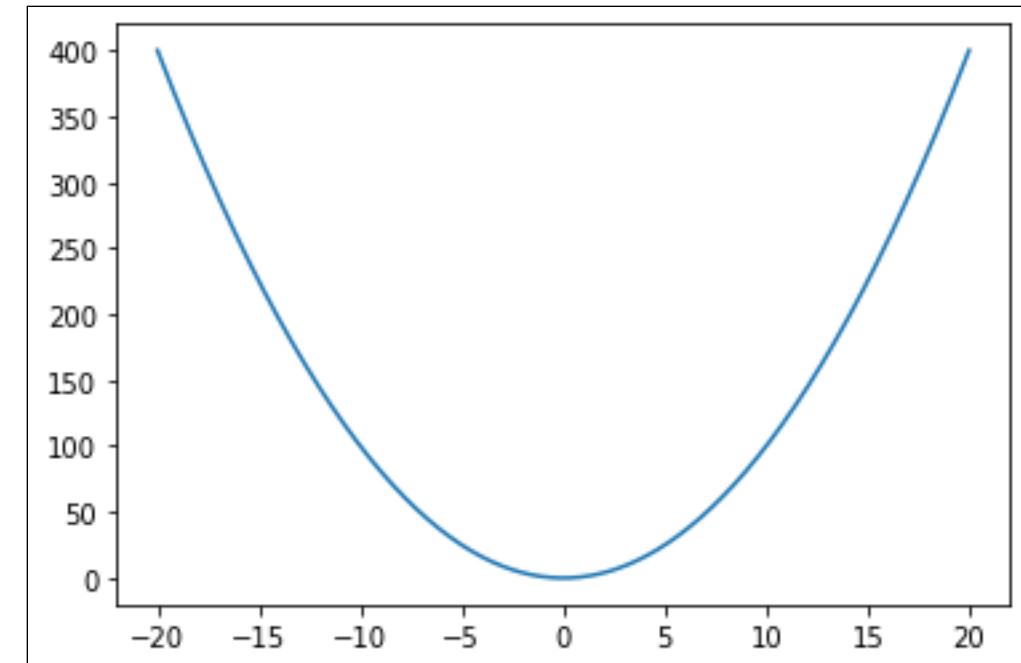


# Vẽ biểu đồ đơn giản

# Ví dụ: vẽ biểu đồ $y = x^2$

```
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 * x
# vẽ biểu đồ tương quan giữa x và y
plt.plot(x, y)
# hiển thị biểu đồ
plt.show()
```

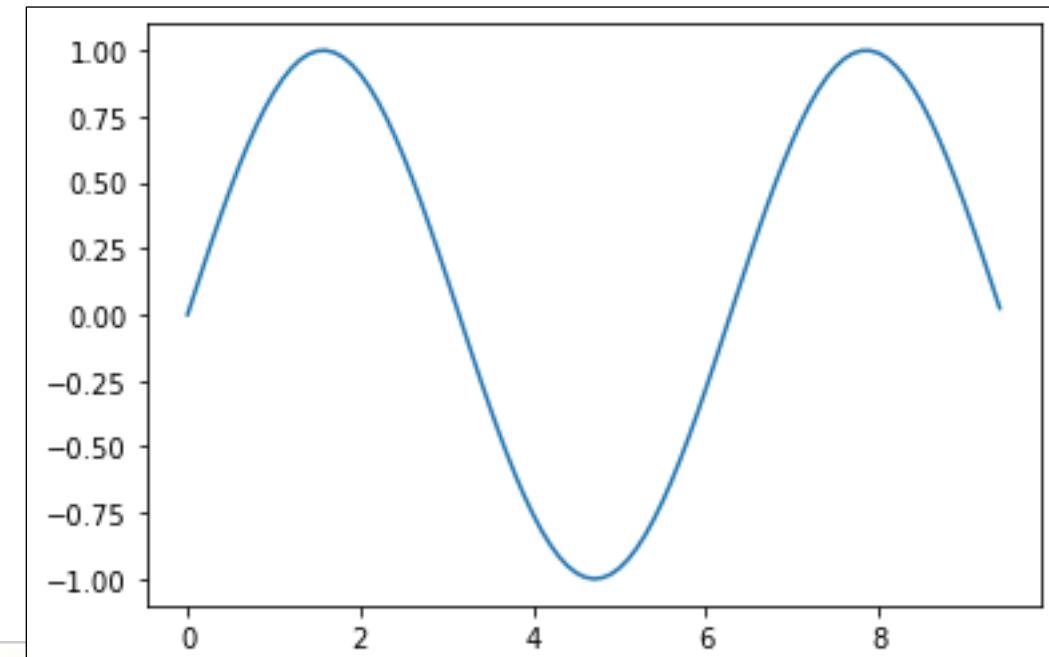




# Ví dụ: Vẽ biểu đồ hình sin

```
import numpy as np          # thư viện numpy
import matplotlib.pyplot as plt # thư viện pyplot

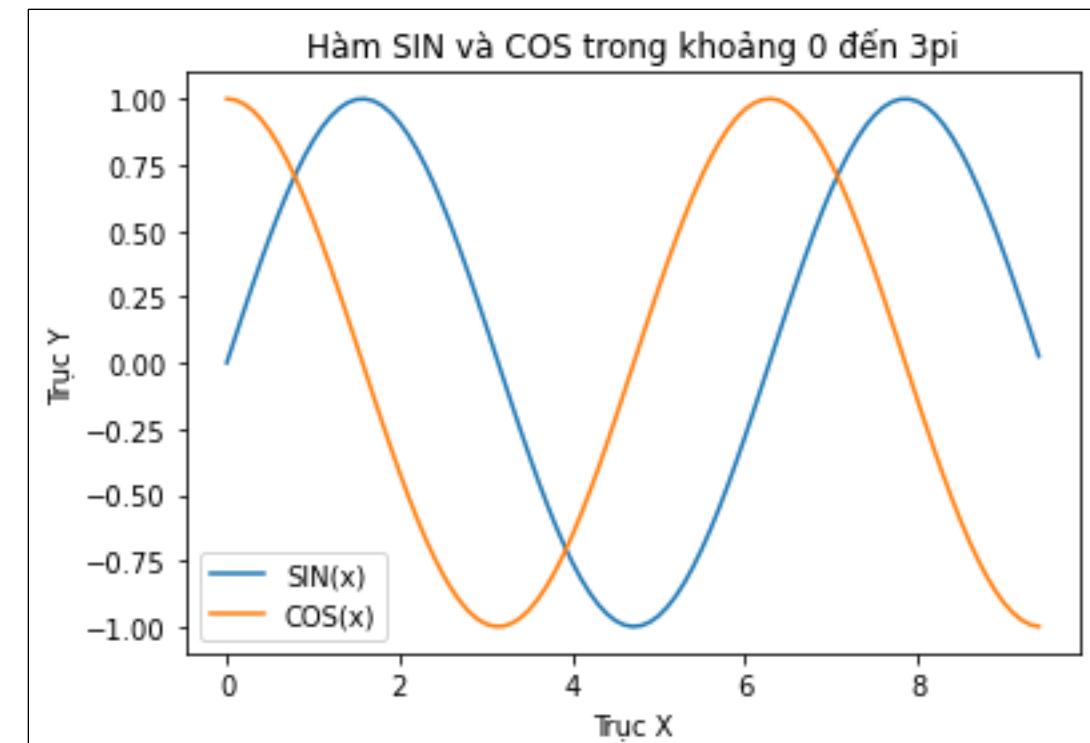
# chia đoạn từ 0 đến  $3\pi$  thành các đoạn con 0.1
x = np.arange(0,3 * np.pi, 0.1)
# tính sin tương ứng với từng phần tử của x
y = np.sin(x)
# vẽ biểu đồ tương quan giữa x và y
plt.plot(x, y)
# hiển thị biểu đồ
plt.show()
```





# Ví dụ: Biểu đồ cả SIN và COS

```
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.plot(x, y_sin)  
plt.plot(x, y_cos)  
plt.xlabel('Trục X')  
plt.ylabel('Trục Y')  
plt.title('Hàm SIN và COS  
trong khoảng 0 đến 3pi')  
plt.legend(['SIN(x)', 'COS(x)'])  
plt.show()
```





# Các bước vẽ biểu đồ với matplotlib

- 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



# Một số loại biểu đồ thông dụng trong matplotlib



I.

# Biểu đồ dạng đường (line plot)

# Line plot

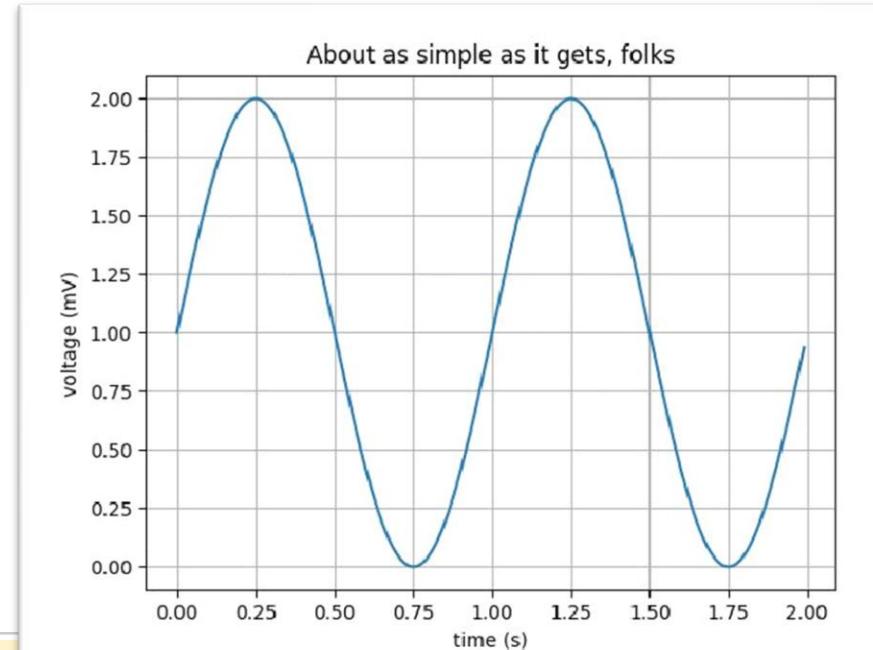
- Biểu đồ thể hiện tương quan giữa X và Y

## ■ 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**





# Line plot: fmt

- “**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**



# Line plot: fmt

- **[marker]** – cách đánh dấu dữ liệu:

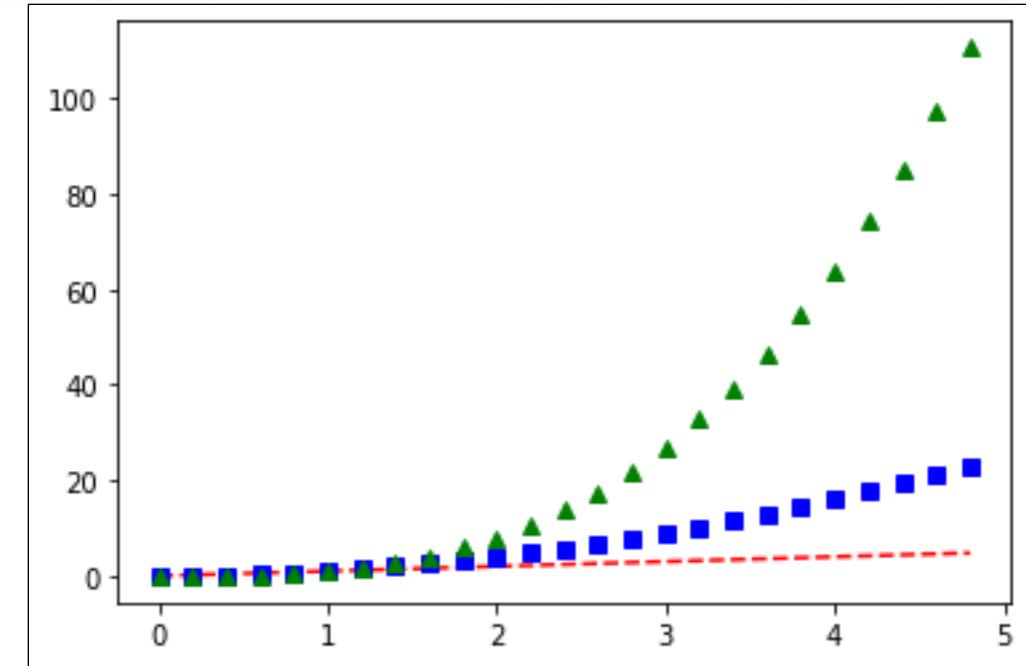
- ✓ 'o' – hình tròn
- ✓ 'v' – tam giác xuông ('^', '<', '>')
- ✓ '\*' – ngôi sao
- ✓ '.' – chấm
- ✓ 'p' – ngũ giác
- ✓ ...

- **[line]** – cách vẽ đường:

- ✓ '-' – nét liền
- ✓ '--' – nét đứt
- ✓ '-.' – gạch chấm
- ✓ ':' – đường chấm

# Ví dụ

```
import numpy as np
import matplotlib.pyplot as plt
# chia đoạn 0-5 thành các bước 0.2
t = 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(t,t,'r--',t,t**2,'bs',t, t**3,'g^')
plt.show()
```





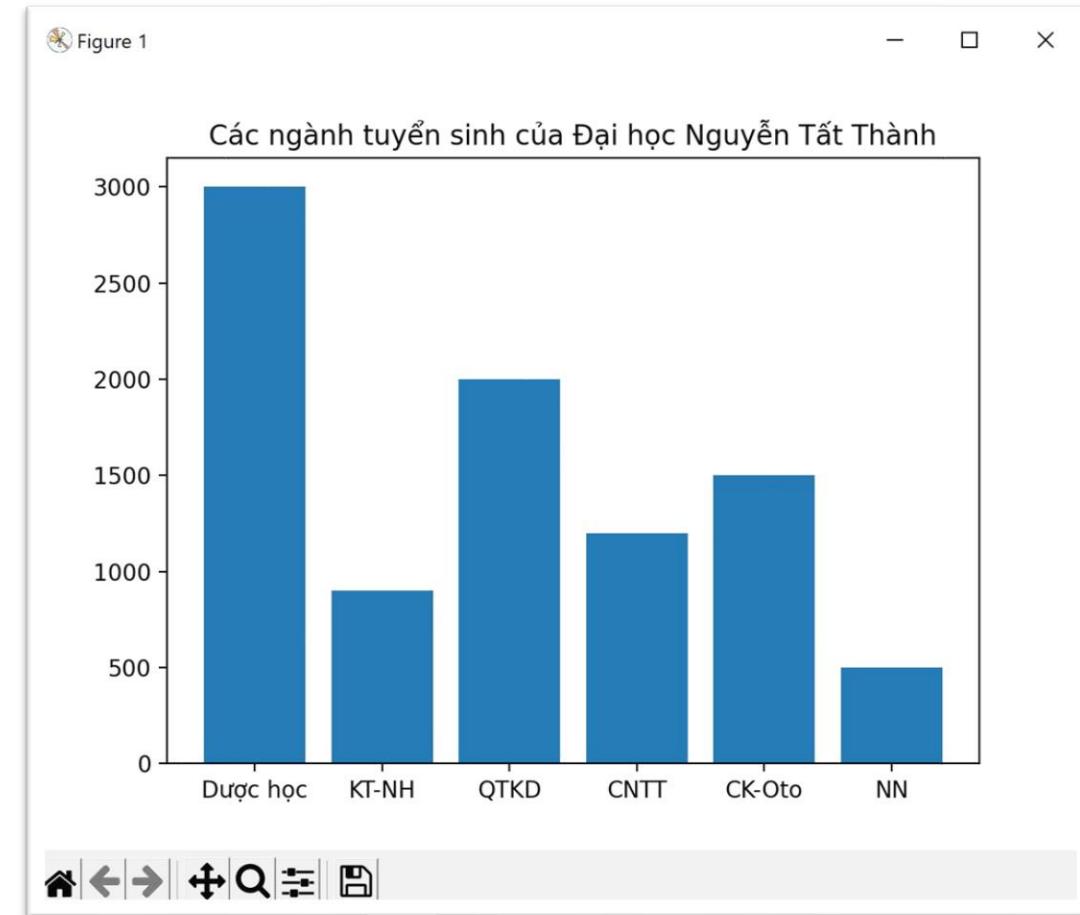
2.

# Biểu đồ dạng cột (bar plot)



# Biểu đồ dạng cột chiều đứng

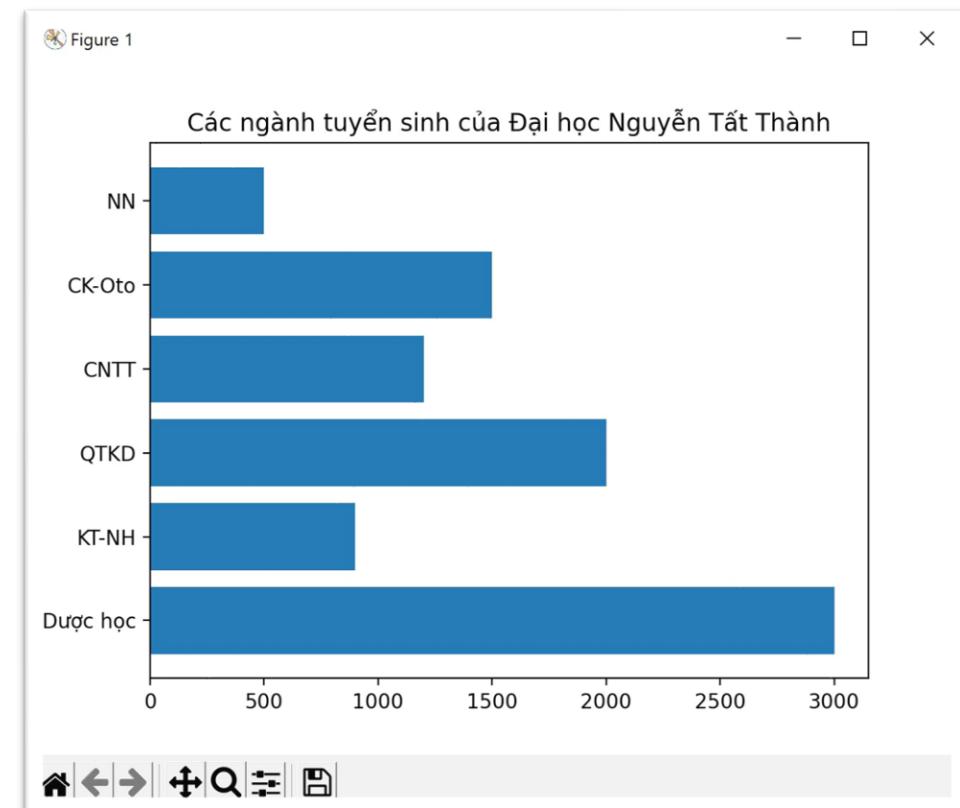
```
import matplotlib.pyplot as plt  
D = { 'Dược học': 3000,  
      'KT-NH': 900,  
      'QTKD': 2000,  
      'CNTT': 1200,  
      'CK-Oto': 1500,  
      'NN': 500 }  
  
plt.bar(range(len(D)), D.values(),  
        align='center')  
plt.xticks(range(len(D)), D.keys())  
plt.title('Các ngành tuyển sinh  
của Đại học Nguyễn Tất Thành')  
plt.show()
```





# Biểu đồ dạng cột chiều ngang

```
import matplotlib.pyplot as plt  
D = { 'Dược học': 3000,  
      'KT-NH': 900,  
      'QTKD': 2000,  
      'CNTT': 1200,  
      'CK-Oto': 1500,  
      'NN': 500 }  
plt.barh(range(len(D)), D.values(),  
        align='center')  
plt.yticks(range(len(D)), D.keys())  
plt.title('Các ngành tuyển sinh  
của Đại học Nguyễn Tất Thành')  
plt.show()
```

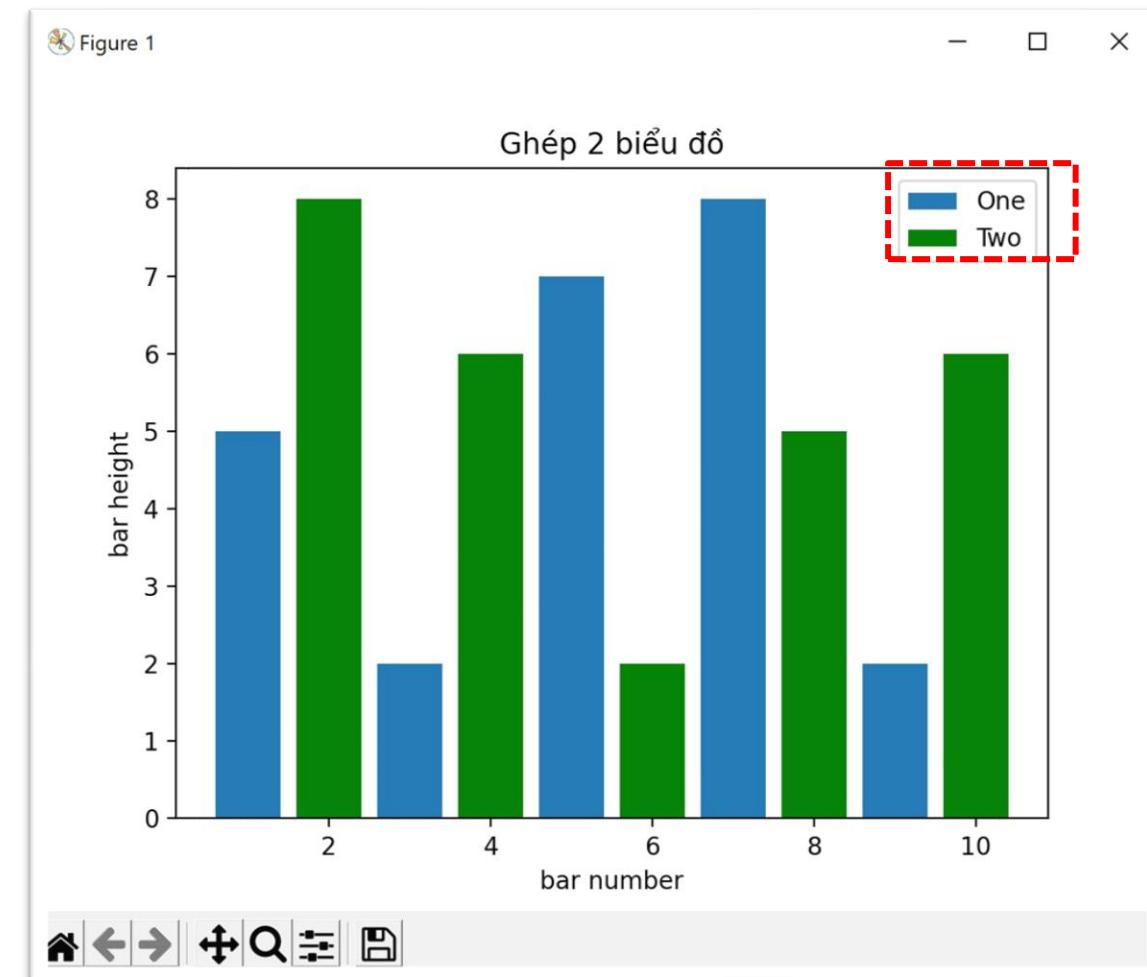




# Có thể ghép 2 biểu đồ

```
import matplotlib.pyplot as plt

plt.bar([1,3,5,7,9],[5,2,7,8,2],
label="One")
plt.bar([2,4,6,8,10],[8,6,2,5,6]
,label="Two",color='g')
[plt.legend()]
plt.xlabel('bar number')
plt.ylabel('bar height')
plt.title('Ghép 2 biểu đồ')
plt.show()
```





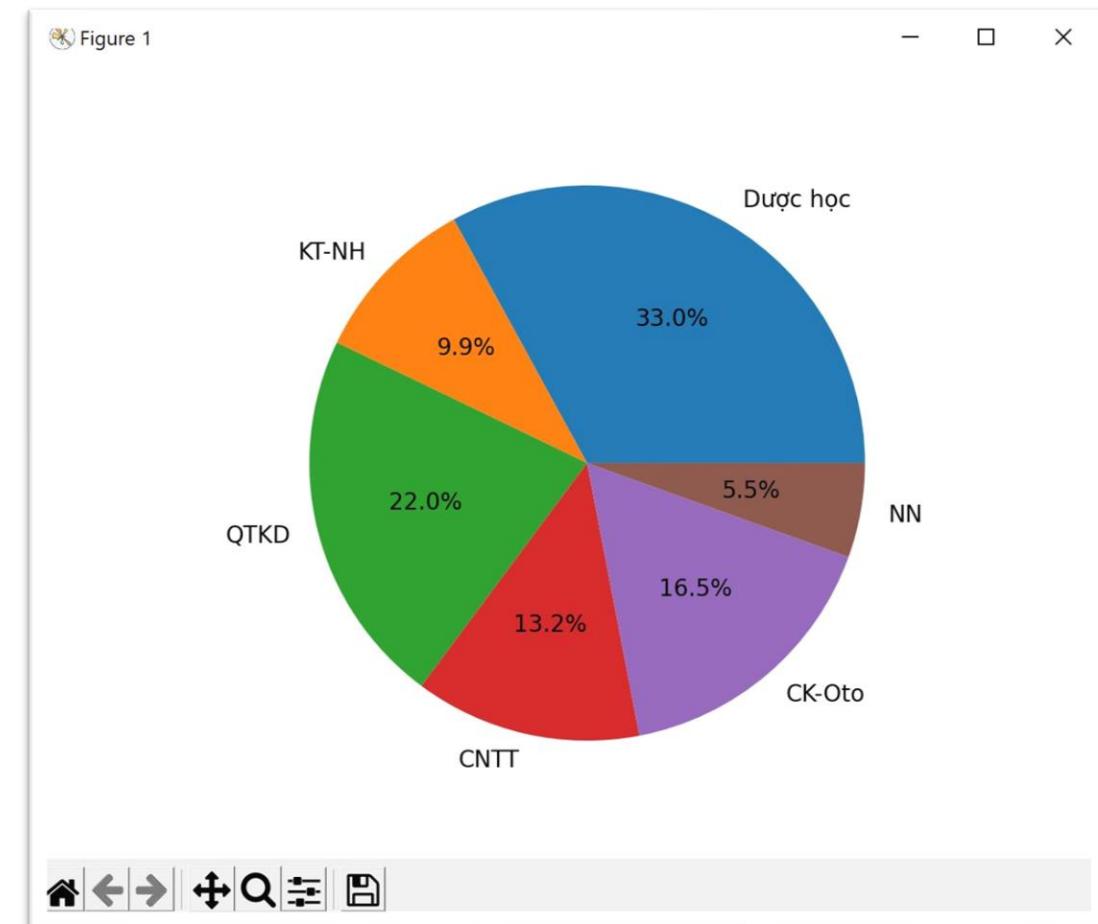
3.

# Biểu đồ bánh (pie chart)



# Làm rõ tương quan chi tiết / tổng thể

```
import matplotlib.pyplot as plt  
D = { 'Dược học': 3000,  
      'KT-NH': 900,  
      'QTKD': 2000,  
      'CNTT': 1200,  
      'CK-Oto': 1500,  
      'NN': 500 }  
  
plt.pie(D.values(), labels=D.keys(),  
        autopct='%1.1f%%')  
plt.axis('equal') #trục x = trục y  
plt.show()
```





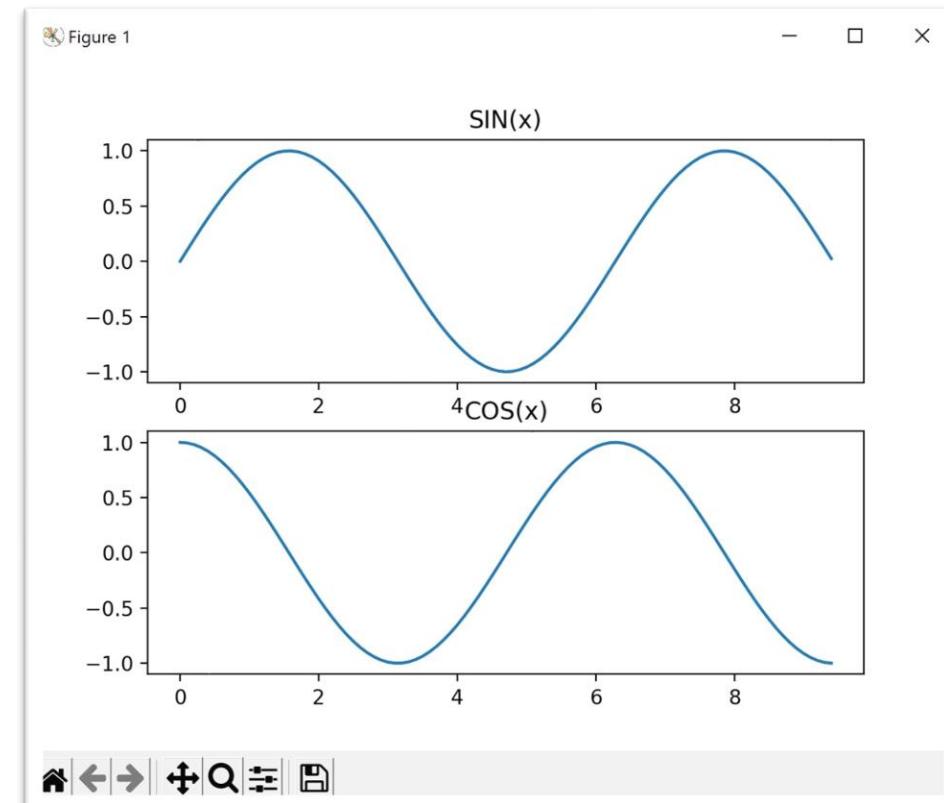
4.

# Một số chức năng hữu ích



# Chia thành các biểu đồ con

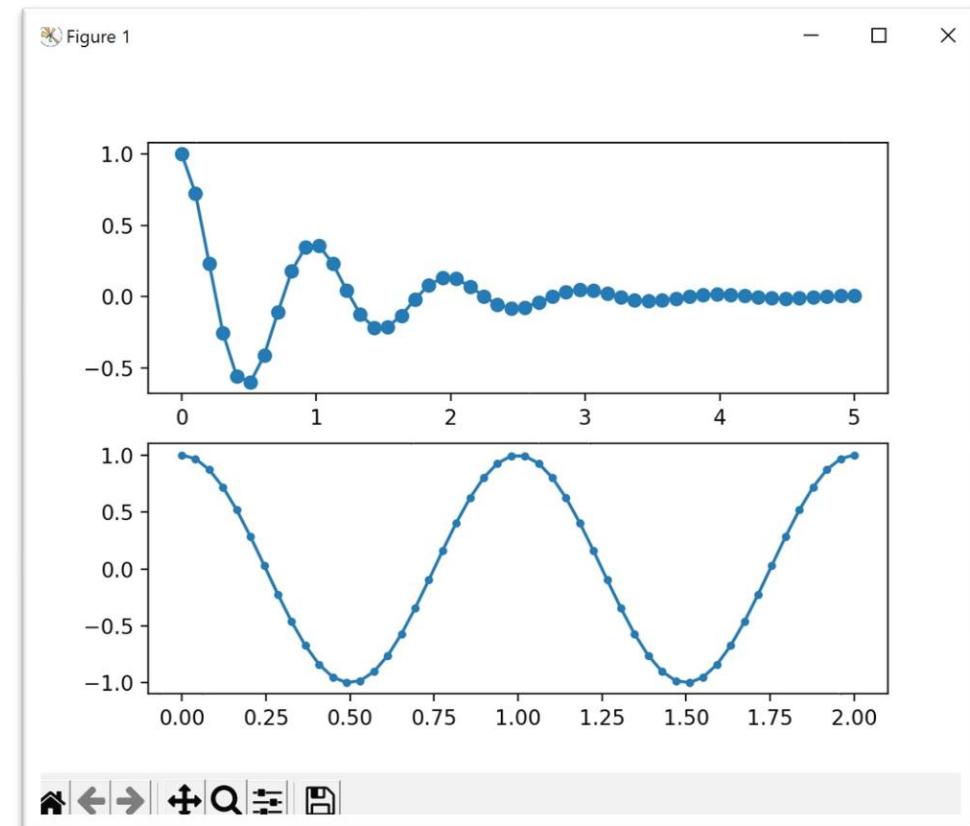
```
import numpy as np
import matplotlib.pyplot as plt
x = np.arange(0, 3 * np.pi, 0.1)
y_sin = np.sin(x)
y_cos = np.cos(x)
plt.subplot(2, 1, 1) # biểu đồ 1
plt.plot(x, y_sin)
plt.title('SIN(x)')
plt.subplot(2, 1, 2) # biểu đồ 2
plt.plot(x, y_cos)
plt.title('COS(x)')
plt.show()
```





# Chia thành các biểu đồ con

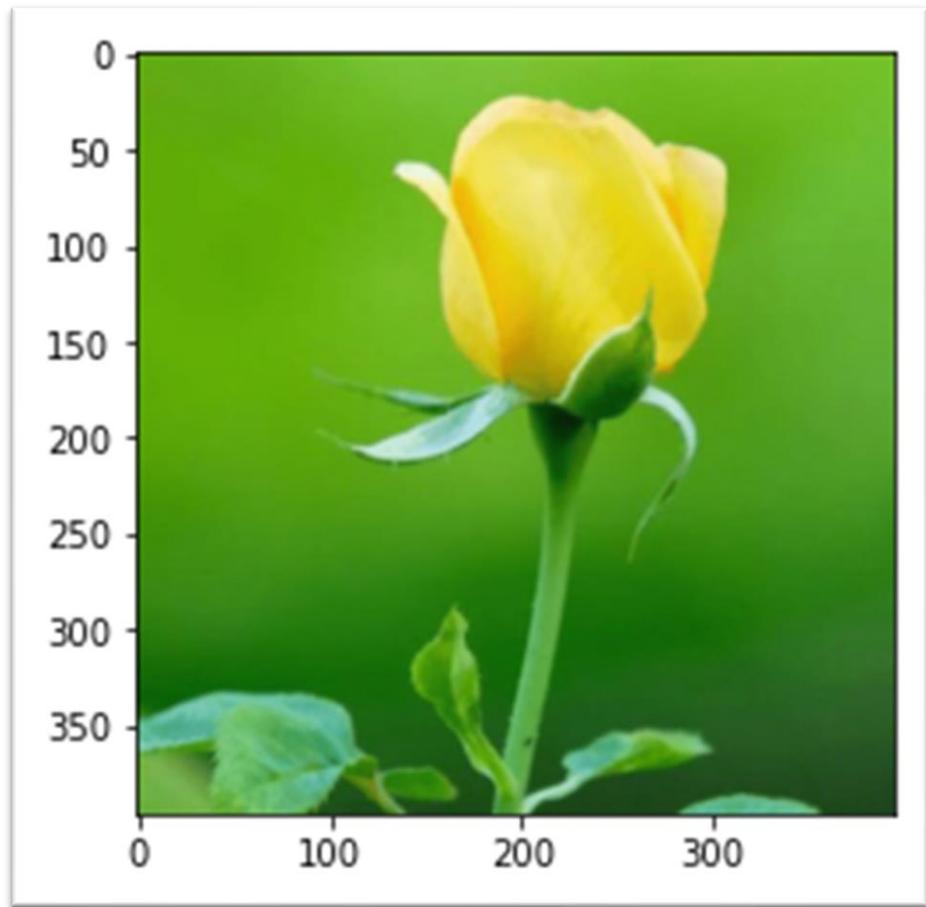
```
import numpy as np  
import matplotlib.pyplot as plt  
  
x1 = np.linspace(0.0, 5.0)  
x2 = np.linspace(0.0, 2.0)  
y1 = np.cos(2* np.pi* x1)* np.exp(-x1)  
y2 = np.cos(2* np.pi* x2)  
plt.subplot(2, 1, 1)  
plt.plot(x1, y1, 'o-')  
plt.subplot(2, 1, 2)  
plt.plot(x2, y2, '.-')  
plt.show()
```





# Nạp ảnh (png\_jpg)

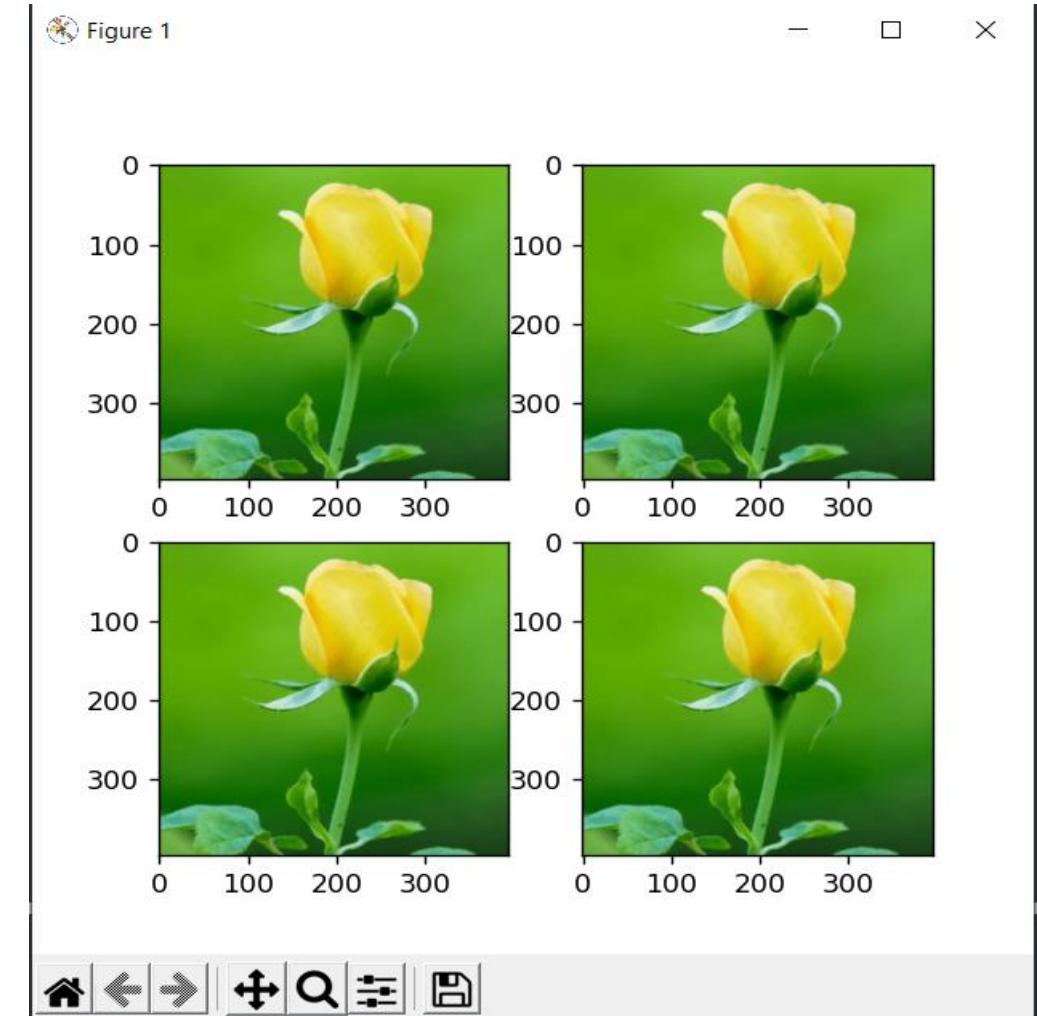
```
import matplotlib.pyplot as plt  
import matplotlib.image as mpimg  
  
image = mpimg.imread("hoa hồng.jpg")  
plt.imshow(image)
```





# Nạp ảnh (png\_jpg)

```
import matplotlib.pyplot as plt  
import matplotlib.image as mpimg  
  
image = mpimg.imread("hoa hồng.jpg")  
fig, axs = plt.subplots(2, 2,  
figsize=(5, 5))  
axs[0, 0].imshow(image)  
axs[1, 0].imshow(image)  
axs[0, 1].imshow(image)  
axs[1, 1].imshow(image)  
plt.show()
```





# Lưu biểu đồ ra file

```
import matplotlib.pyplot as plt
import matplotlib.image as mpimg

image = mpimg.imread("hoa hồng.jpg")
fig, axs = plt.subplots(2, 2, figsize=(5, 5))
axs[0, 0].imshow(image)
axs[1, 0].imshow(image)
axs[0, 1].imshow(image)
axs[1, 1].imshow(image)

plt.savefig('1.jpg')
plt.savefig('1.pdf')
```





**THANK  
YOU**

---

*THS. VƯƠNG XUÂN CHÍ*  
**VXCHI@NTT.EDU.VN**

**0903 270 567**