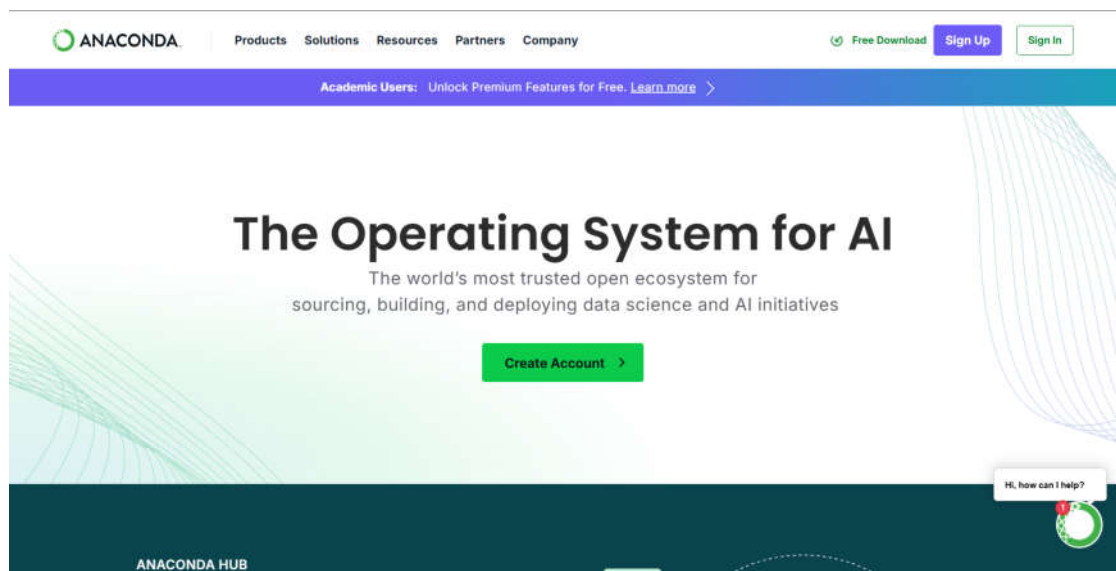


Day 1 - Cài đặt môi trường cho lập trình Python

Đặng-Nhã Nguyễn, Đoàn Ngọc Cường



1 Anaconda là gì và tại sao cần Anaconda?

1.1 Anaconda là gì?

Anaconda là một bộ công cụ giúp chúng ta dễ dàng cài đặt và quản lý Python cùng với các thư viện quan trọng để làm việc với dữ liệu, trí tuệ nhân tạo (AI), và khoa học máy tính. Nó bao gồm trình quản lý gói conda, Python, và nhiều công cụ hỗ trợ như Jupyter Notebook, Spyder.

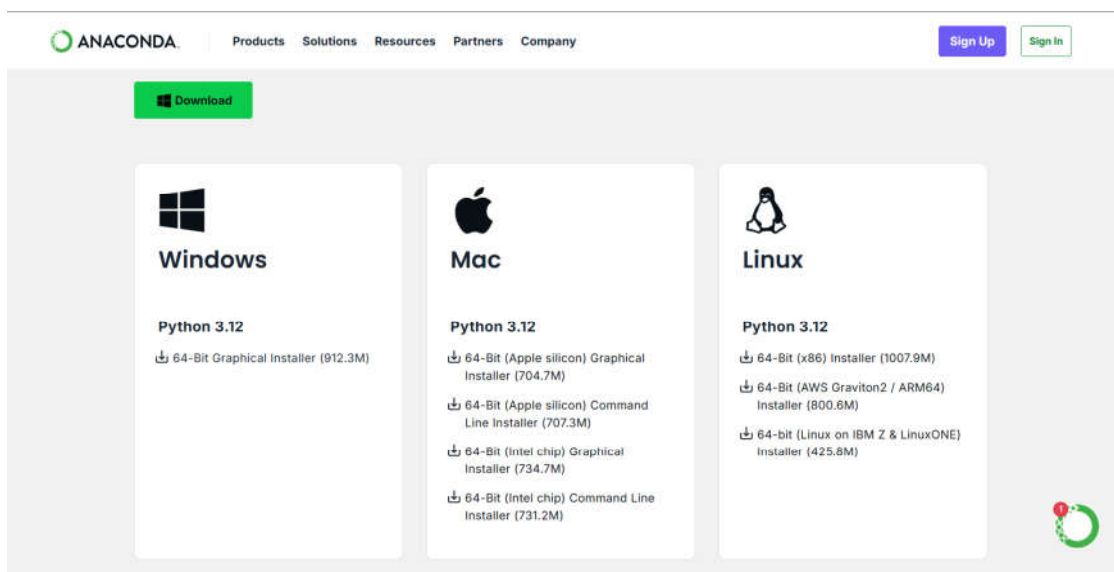
1.2 Lợi ích khi sử dụng Anaconda:

- **Quản lý môi trường dễ dàng:** Bạn có thể tạo, kích hoạt hoặc xóa các môi trường riêng biệt mà không ảnh hưởng đến môi trường chính của hệ thống.
- **Cài đặt thư viện nhanh chóng:** Anaconda đi kèm với nhiều thư viện phổ biến như numpy, pandas, matplotlib, v.v., giúp tiết kiệm thời gian cài đặt.
- **Hỗ trợ đa nền tảng:** Windows, macOS và Linux.

2 Cài đặt Anaconda

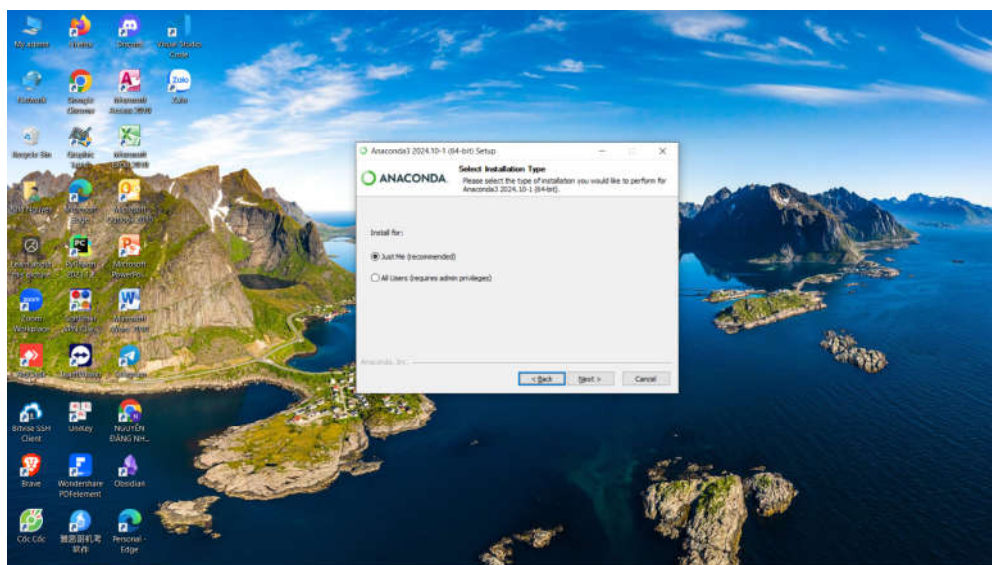
Bước 1: Tải và cài đặt Anaconda

- Truy cập trang web chính thức của Anaconda tại: <https://www.anaconda.com/download>
- Tải xuống phiên bản phù hợp với hệ điều hành của bạn (window, macOS hoặc Linux).

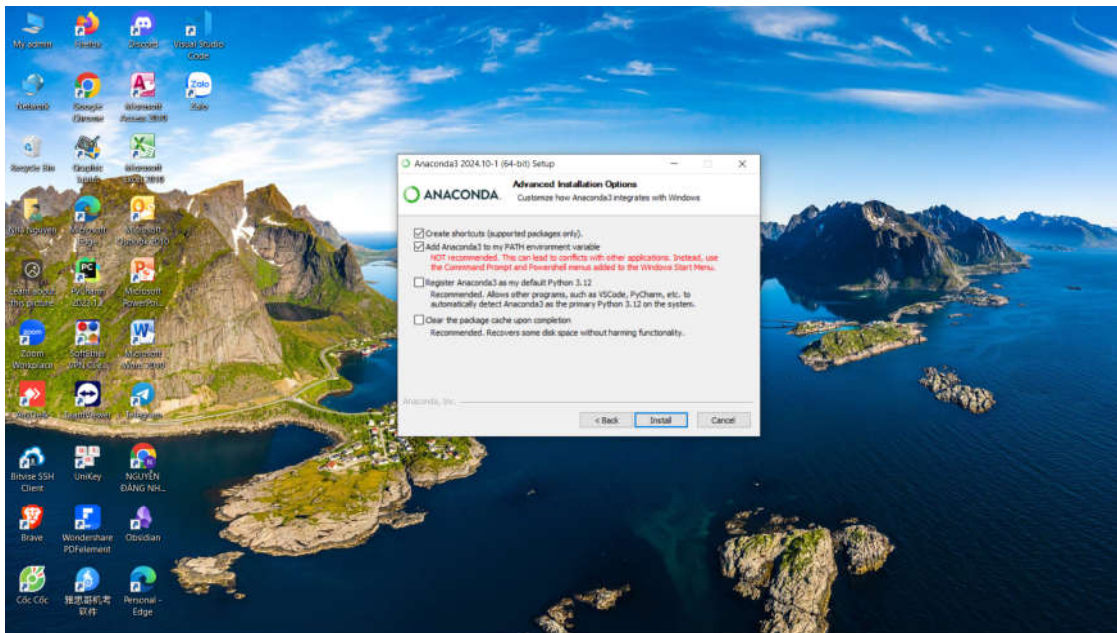


- Chạy file .exe vừa tải về để bắt đầu quá trình cài đặt.

Bước 2: Lưu ý trong quá trình cài đặt



- Chọn loại cài đặt, gồm có 2 loại: **Just me** và **All Users**.
 - **Just me:** Cài đặt Anaconda chỉ dành cho tài khoản người dùng Microsoft hiện tại. Tức là khi login vào tài khoản khác sẽ không có phần mềm này.
 - **All Users:** Cài đặt Anaconda cho tất cả người dùng trên máy tính. Nghĩa là cho dù dùng bất kỳ account nào đăng nhập cũng có thể sử dụng anaconda.
 - **Khuyến nghị:** Trên máy tính cá nhân, mọi người có thể chọn bất kỳ tùy chọn nào trong hai tùy chọn trên. Ở đây mình sẽ chọn "Just Me" để tiến hành cài đặt mà không gặp trở ngại (ví dụ cần quyền admin).
- Chọn nơi lưu trữ: Đảm bảo mọi người nhớ đường dẫn cài đặt để dễ dàng quản lý sau này khi gặp lỗi.
- Tick vào ô "Add Anaconda3 to my PATH environment variable": Điều này giúp chúng ta có thể chạy Anaconda trực tiếp từ terminal bằng lệnh 'conda'.



- Tiếp tục các bước còn lại cho tới khi cài đặt thành công.

Bước 3: Kiểm tra cài đặt

Mở terminal hoặc command prompt, gõ lệnh:

```
1 conda --version
```

Nếu hiển thị phiên bản của Anaconda, bạn đã thêm PATH thành công.

Lưu ý: Nếu vẫn gặp lỗi, bạn có thể cần khởi động lại máy để cập nhật các thay đổi.

2.1 Không thể dùng lệnh 'conda' trong terminal (Phần này dành cho những người gặp lỗi)

Nếu bạn đã cài đặt anaconda nhưng quên tick vào ô "Add Anaconda3 to my PATH environment variable" trong quá trình cài đặt hoặc vì lí do nào đó dẫn tới việc không thể dùng lệnh 'conda -version' thì có thể thêm thủ công bằng các bước sau

Bước 1: Xác định đường dẫn thư mục Anaconda3

- **Windows:** Mặc định, Anaconda thường được cài vào:

- C:\Users\- hoặc C:\Anaconda3

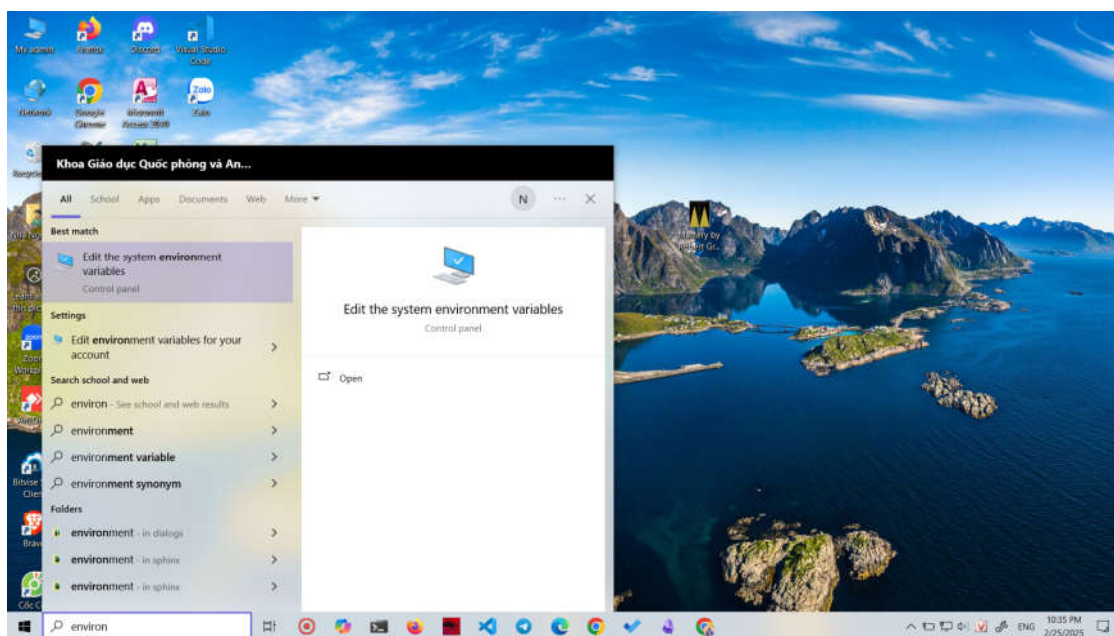
- **Mac/Linux:** Đường dẫn mặc định thường là:

- /Users/<Tên người dùng>/anaconda3
- hoặc /home/<Tên người dùng>/anaconda3

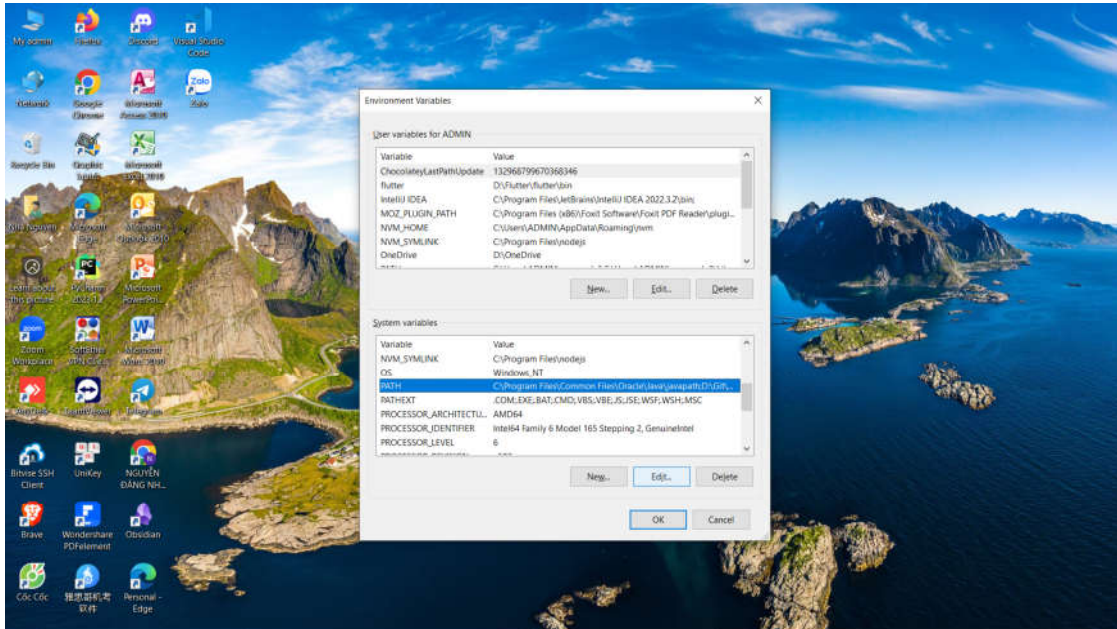
Bước 2: Thêm đường dẫn vào PATH

Cách thêm PATH trên Windows:

1. Mở **Start Menu** và tìm **Environment Variables** hoặc vào **Control Panel** → **System** → **Advanced System Settings** → **Environment Variables**.



2. Trong cửa sổ **System Properties**, chọn **Environment Variables**.
3. Ở phần **System variables**, tìm biến **Path** và nhấn **Edit**.



4. Nhấn **New** và thêm các đường dẫn sau:

- C:\Users\- C:\Users\- C:\Users\

5. Nhấn **OK** để lưu thay đổi.

Cách thêm PATH trên Mac/Linux:

1. Mở terminal.

2. Gõ lệnh:

```
1 nano ~/.bashrc # Hoặc ~/.zshrc nếu dùng zsh
2
```

3. Thêm dòng sau vào cuối file:

```
1 export PATH="/Users/<Tên người dùng>/anaconda3/bin:$PATH"
2
```

4. Lưu file và tải lại cấu hình:

```
1 source ~/.bashrc # Hoặc ~/.zshrc nếu dùng zsh
2
```

3. Khởi tạo môi trường với conda (sử dụng CMD)

Môi trường (environment) trong Conda giống như một hộp riêng biệt chứa các thư viện và phiên bản Python cụ thể để chạy code mà không ảnh hưởng đến hệ thống chung. Dưới đây là hướng dẫn cơ bản sử dụng conda trong CMD.

Bước 1: Kiểm tra phiên bản conda

Mở CMD và nhập lệnh kiểm tra:

```
1 conda --version
```

Nếu lệnh trả về phiên bản, bạn đã cài đặt thành công.

Bước 2: Liệt kê môi trường hiện có

Xem các môi trường ảo bằng lệnh:

```
1 conda env list
```

Bước 3: Tạo môi trường mới

Tạo môi trường với tên và phiên bản Python cụ thể:

```
1 conda create --name <ten_moi_truong> python=<phien_ban_python>
```

Ví dụ: `conda create -name myenv python=3.12`

Nhấn y để xác nhận khi được yêu cầu.

Bước 4: Kích hoạt và hủy kích hoạt môi trường

Kích hoạt môi trường

Dùng lệnh sau để kích hoạt môi trường:

```
1 conda activate <ten_moi_truong>
```

Hủy kích hoạt môi trường

Thoát môi trường khi hoàn tất:

```
1 conda deactivate
```

Bước 5: Cài đặt gói cần thiết

Khi môi trường được kích hoạt, bạn có thể cài đặt các gói:

```
1 conda install <ten_goi>
```

```
2 pip install <ten_goi>
```

Ưu tiên `conda install` để đảm bảo tương thích.

Bước 6: Quản lý môi trường

Liệt kê môi trường

Xem lại các môi trường:

```
1 conda env list
```

Xóa môi trường

Xóa môi trường không còn sử dụng:

```
1 conda remove --name <ten_moi_truong> --all
```

Việc sử dụng môi trường ảo giúp cô lập dự án, tránh xung đột và đảm bảo tính ổn định cho hệ thống.

4. Cài đặt Jupyter Notebook

Bước 1: Truy cập Anaconda Navigator

Mở **Anaconda Navigator**, tìm **Jupyter Notebook**, nhấn **Install** nếu chưa cài hoặc **Launch** để mở.

Bật nhanh Jupyter Notebook bằng CMD

Bước 1: Mở CMD tại thư mục dự án

Chuột phải vào thư mục, chọn **Open Terminal** hoặc **Open Command Prompt**.

Bước 2: Gõ lệnh mở Jupyter Notebook

```
1 jupyter notebook
```

Nếu gặp lỗi không nhận diện lệnh, thêm đường dẫn sau vào PATH:

- C:\Users\Username\anaconda3\Scripts

Bước 3: Mở tệp và bắt đầu làm việc

Trình duyệt sẽ mở giao diện Jupyter, chọn tệp .ipynb hoặc tạo tệp mới để lập trình.

- Chúc bạn thành công! -

BÀI TẬP

Yêu cầu cài đặt Anaconda (2 bài test)

1. **Test 1:** Mở Jupyter Notebook trên máy local, tạo một tệp notebook mới và chạy đoạn code:

```
1 print("Hello World")
```

2. **Test 2:**

- Tạo môi trường conda mới (ví dụ: `conda create -name myenv python=3.8`).
- Kích hoạt môi trường (`conda activate myenv`).
- Cài matplotlib:

```
1 pip install matplotlib
```

hoặc

```
1 conda install matplotlib
```

- Chạy thử đoạn code sau (có thể do ChatGPT gợi ý) để kiểm tra việc vẽ đồ thị:

```
1 import matplotlib.pyplot as plt
2
3 x = [1, 2, 3, 4, 5]
4 y = [i**2 for i in x]
5 plt.plot(x, y, marker='o')
6 plt.title("Matplotlib Test")
7 plt.xlabel("X")
8 plt.ylabel("Y")
9 plt.show()
```

- Đảm bảo code chạy thành công và hiển thị được đồ thị.

Ngoài Anaconda, còn công cụ nào khác?

Ngoài Anaconda, bạn có thể sử dụng các công cụ khác để tạo môi trường ảo:

- `python -m venv`: Một công cụ tích hợp sẵn trong Python để tạo môi trường ảo.
- `virtualenv`: Công cụ mạnh mẽ hơn `venv` cho phép tùy chỉnh và quản lý môi trường ảo một cách linh hoạt.

Ví dụ: Tạo môi trường ảo bằng `venv`

```
1 # Tạo môi trường ảo
2 python -m venv myenv
3
4 # Kích hoạt môi trường ảo
5 # Windows:
6 myenv\Scripts\activate.bat
```



```
7  
8 # macOS/Linux:  
9 source myenv/bin/activate
```

Lựa chọn nào phù hợp hơn?

- Nếu bạn cần quản lý nhiều dự án phức tạp hoặc dùng thư viện lớn như TensorFlow, PyTorch, Anaconda là sự lựa chọn tối ưu. Vì Anaconda tích hợp sẵn nhiều thư viện khoa học dữ liệu, hỗ trợ quản lý phụ thuộc tốt và cài đặt dễ dàng các gói lớn, đặc biệt với GPU.
- Nếu chỉ cần môi trường nhẹ cho một vài dự án nhỏ, `venv` hoặc `virtualenv` có thể đáp ứng tốt. Vì chúng tạo môi trường nhanh, gọn và chỉ cài đặt đúng các gói cần thiết mà không gây nặng máy.

- *Hết* -