

Quản lý phiên bản

Họ và tên: Hoàng Đình Hữu

MSSV: 20521384. Lớp: IE221.N23.CNCL

Lecturer: Nguyễn Thanh Sơn

Cách quản lý và sử dụng các phiên bản của hệ thống và thư viện trong python:

- I. Cách quản lý và sử dụng các phiên bản của hệ thống trong python:
 - Để quản lý các phiên bản của hệ thống trong python dễ dàng thì ta nên cài một công cụ quản lý các phiên bản của python như pyenv và venv.
 - Việc sử dụng các phiên bản hệ thống và thư viện trong python phụ thuộc khá nhiều vào thư viện mà dự án hay đồ án đó sử dụng, vì đối với mỗi phiên bản thì sẽ có nhiều khác biệt về các cấu trúc lệnh hay các thư viện sẽ có các phiên bản khác nhau. Vì vậy, ta nên thường xuyên theo dõi và cập nhật các thay đổi của phiên bản Python để có thể sử dụng nó một cách hiệu quả hơn.
 - Để kiểm tra phiên bản python đang sử dụng: ta có thể mở command prompt và gõ dòng lệnh “python -V” hoặc nhập trực tiếp trên google colab “!python --version”.
 - Link hướng dẫn cài đặt và sử dụng: [pyenv for Windows | pyenv-win](#)
- II. Cách quản lý và sử dụng các phiên bản của thư viện trong python:
 - Các thư viện trong python thay đổi và cập nhật theo các phiên bản khác nhau và tùy theo phiên bản Python đó còn hỗ trợ cho các phiên bản cũ hay không. Việc sử dụng thư viện sẽ trở nên khá khó khăn nếu thư viện đó không tồn tại tùy theo phiên bản python hoặc cú pháp thay đổi.
 - Để hiểu rõ phiên bản thư viện đang sử dụng:
 - Với VisualStudioCode, Google Colab: Ví dụ kiểm tra phiên bản của thư viện numpy

```
import numpy
print(numpy.version.version)
```

- Với command prompt: thì ta sử dụng lệnh “pip list”, ta có thể thấy tất cả các phiên bản của thư viện mà ta đã cài trong máy.
- Việc sử dụng phiên bản nào của thư viện sẽ phụ thuộc vào nhiều yếu tố. Ví dụ như tham khảo một bài code từ năm 2017 thì hên xui nó sẽ chạy được

và lỗi sẽ đa phần do việc phiên bản python không hỗ trợ phiên bản thư viện cũ nên ta cần phải xem lại là cần phải cài lại phiên bản nào của python hoặc phiên bản thư viện để phù hợp với bài code đó.

- Nếu thư viện chưa được cài thì gõ lệnh: `pip install "tên thư viện"` vào terminal trên VSCode hoặc `!pip install` đối với GGColab.
- Để cài các phiên bản của thư viện: có 2 cách một là cài lại thư viện theo phiên bản đó, hai là cài lại phiên bản của python nếu phiên bản hiện tại không hỗ trợ thư viện đó (chỉ dùng cách 2 nếu cách 1 không xài được)
 - Với Visual Studio Code: `pip install --force-reinstall numpy==version`
 - Với Google Colab:

```
!pip install numpy version==0.1.0

Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Requirement already satisfied: numpy in /usr/local/lib/python3.8/dist-packages (1.22.4)
Collecting version==0.1.0
  Downloading version-0.1.0.tar.gz (1.9 kB)
  Preparing metadata (setup.py) ... done
Building wheels for collected packages: version
  Building wheel for version (setup.py) ... done
  Created wheel for version: filename=version-0.1.0-py3-none-any.whl size=2307 sha256=17c519e74b06494ba9a8d0130fccf508220bdf6e9ddd1ef47dcd7dc702742b3
  Stored in directory: /root/.cache/pip/wheels/99/a4/8e/f4331a24d58242c3b151bf1ac9a6d74303f1b0b7726157d74f
Successfully built version
Installing collected packages: version
Successfully installed version-0.1.0
```

- Khuyến khích việc sử dụng các phiên bản mới nhất của thư viện và phiên bản của Python để tránh các lỗi từ các thành phần lỗi thời và có trải nghiệm tốt hơn.