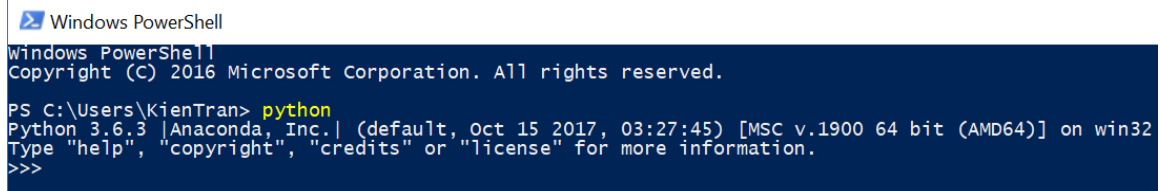


# Python

## Cài đặt Jupyter Notebook, Python và các thư viện

- Download và cài đặt [Anaconda](#). Anaconda đã bao gồm luôn cho bạn Jupyter Notebook (trước đây gọi là IPython Notebook), Python và các thư viện cần thiết. **Lưu ý: chọn bản (i) ứng với hệ điều hành của bạn, và (ii) Python 3.x (chứ không phải 2.x); khi cài đặt, nhớ check dòng “Add Anaconda to my PATH environment variable”.**
- Test cài đặt (mình đang giả định bạn dùng Windows): mở PowerShell (dùng chức năng search của Windows và search với từ khóa PowerShell) và gõ `python`; nếu cài đặt OK thì bạn sẽ vào cửa sổ command line của Python và kết quả sẽ có dạng như ở hình bên dưới:



```
Windows PowerShell
Copyright (C) 2016 Microsoft Corporation. All rights reserved.

PS C:\Users\KienTran> python
Python 3.6.3 [Anaconda, Inc.] (default, Oct 15 2017, 03:27:45) [MSC v.1900 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
```

- Ở đây, bạn có thể thực thi các câu lệnh của Python; ví dụ, bạn gõ `1 + 1` (rồi Enter). Để thoát ra khỏi cửa sổ command line của Python, bạn gõ `quit()`. Sau khi thoát thì các câu lệnh đã gõ sẽ không được lưu lại; do đó, mình thường chỉ dùng cửa sổ command line của Python để test nhanh các câu lệnh. Trong các bài tập môn học, bạn sẽ code trên Jupyter Notebook (sẽ nói ở dưới), chứ không dùng cửa sổ command line này.

## Làm quen với Jupyter Notebook, Python và các thư viện

- Mình có đính kèm 2 file ví dụ về Jupyter Notebook là `Python.ipynb` và `Markdown.ipynb`. Bạn mở PowerShell, gõ `cd "<đường dẫn đến thư mục chứa 2 file ipynb>"`, và gõ `jupyter notebook`. Trang chủ của notebook sẽ được mở ra ở web browser của bạn. Ở đây, bạn sẽ thấy hai file ipynb ở trên. Bạn có thể mở file ipynb bằng cách click vào file này.
- Bạn hãy mở file `Python.ipynb`. Notebook ứng với file này sẽ được mở ra ở một tab mới.
  - Một notebook sẽ được cấu thành từ các **cell**. Bạn có thể dùng phím mũi tên lên/xuống để di chuyển giữa các cell. Có hai loại cell là **markdown cell** và **code cell**. Markdown cell cho phép soạn thảo và hiển thị văn bản bằng cách viết code theo cú pháp của markdown rồi thực thi. Còn code cell cho phép bạn viết và thực thi Python code. Dấu hiệu để nhận biết code cell là ở bên trái có `In [...]`; còn markdown cell thì không có.
  - Khi bạn gõ từ bàn phím vào notebook, có hai chế độ là **edit mode** và **command mode**. Với một cell, bạn sẽ ở một trong hai chế độ này: nếu ở edit mode thì bạn sẽ thấy đường viền của cell có màu xanh lá và con trỏ nhấp nháy; ngược lại, bạn sẽ thấy đường viền của cell có màu xanh dương và không có con trỏ nhấp nháy. Để chuyển từ command mode sang edit mode, bạn có thể ấn Enter; để chuyển ngược lại, bạn có thể ấn Esc. Ở edit mode, bạn có thể soạn thảo một cách bình thường (ví dụ, bạn gõ phím `b` thì sẽ hiện ra chữ `b`). Ở command mode, khi bạn gõ bàn phím thì các phím được gõ sẽ được ánh xạ sang các câu lệnh (ví dụ, bạn gõ phím `b` thì sẽ thực hiện câu lệnh chèn thêm một cell ở dưới cell hiện tại). Bạn có thể xem danh sách ánh xạ từ phím sang câu lệnh bằng cách chọn Help – Keyboard Shortcuts (hoặc ấn phím `h` khi đang ở command mode).

Bạn hãy xem thử phím tắt nào giúp chuyển từ code cell sang markdown cell và ngược lại? Việc sử dụng các phím tắt khi ở command mode giúp ta thực hiện các thao tác một cách nhanh chóng.

- Như vậy, để viết code ở một cell thì bạn cần phải ở edit mode; để thực thi code của cell đó thì bạn có thể ấn `Ctrl+Enter` (hoặc `Shift+Enter`). Với markdown cell thì bạn sẽ viết code theo cú pháp của markdown (bạn xem các cú pháp thông dụng của markdown ở file `Markdown.ipynb`), với code cell thì bạn sẽ viết code theo cú pháp của Python (sẽ được hướng dẫn bên dưới).
- Bây giờ, ở trang chủ của notebook (trang đầu tiên được mở ra khi bạn gõ `jupyter notebook`), bạn hãy chọn `New - Python 3` để mở ra một notebook mới. Ở phía trên bạn sẽ thấy dòng `Untitled`, bạn click vào, đặt tựa đề cho notebook của bạn là `LearningPython`, rồi `Ctrl+S` để lưu lại; bạn sẽ thấy trong thư mục hiện tại sẽ có file `LearningPython.ipynb`. Tiếp theo, bạn sẽ vừa đọc [tutorial về Python này](#) (chỉ cần đọc các phần về Python, Numpy, Matplotlib; bỏ qua phần về SciPy, trong phần về Python bỏ qua phần về classes), vừa gõ các câu lệnh vào notebook để chạy thử; tập sử dụng code cell và markdown cell, edit mode và command mode (bạn tham khảo cách mình làm trong file `Python.ipynb` ở trên).
- Tài liệu video (không bắt buộc): nếu bạn muốn xem thêm video demo về Jupyter Notebook thì bạn có thể xem [ở đây](#). Ngoài ra, tác giả cũng có [một series các video về Python](#) (mình thấy rất tốt), bạn có thể lưu lại và từ từ xem; nếu muốn thì trước mắt bạn có thể xem các video đầu mà ở tựa đề có ghi là “Python Tutorial for Beginners”.

## Python level 2

Sau khi thực hành với tutorial ở trên, hy vọng là làm bạn cảm thấy sự dễ dàng khi lập trình bằng Python (so với các ngôn ngữ low-level khác như C/C++). Tuy nhiên, tutorial ở trên là ở mức cơ bản (tạm gọi là level 1). Trong một số trường hợp, để có thể code đúng thì cần lên level 2, cần hiểu sâu hơn một ít về cách hoạt động bên dưới của Python. Để lên level 2, bạn sẽ đọc file “`PythonTutorial2.pdf`” đính kèm (nguồn: file này được lấy ra từ [file gốc này](#)). Các ý chính bạn nên nắm được sau khi đọc xong tutorial này:

- Nắm được cơ chế hoạt động ở bên dưới của phép gán (assignment) trong Python, nắm được khi nào thay đổi dữ liệu của biến này thì sẽ làm thay đổi dữ liệu của biến khác (để tránh chạy sai so với mong muốn).
- Nắm được cơ chế truyền các tham số của hàm trong Python, nắm được khi nào thay đổi dữ liệu của biến trong hàm thì sẽ làm thay đổi dữ liệu của biến ngoài hàm (để tránh chạy sai so với mong muốn).
- Nắm được sự khác biệt giữa slicing trong list và slicing trong Numpy array.