

Utilisation des notebooks pour les exercices en ligne

Python 3

Apprendre à programmer en Python avec Pyzo et Jupyter Notebook

Introduction

Jupyter Notebook est une application web qui fournit deux fonctionnalités complémentaires :

- une interface qui permet d'exécuter (entre autres) du code *Python 3* ;
- un langage simple qui permet d'intégrer dans un même document du texte, des formules mathématiques, des graphiques, des images et des vidéos.

On peut ainsi créer des documents multimédias interactifs (appelés *notebooks*) qui permettent d'exécuter des codes à volonté, ce qui offre une grande valeur pédagogique. *Jupyter* permet de stocker et de partager ces documents, simples fichiers textuels, reconnaissables par leur extension `.ipynb`.

Note importante : Vous pouvez vous entraîner à l'utilisation de *Jupyter Notebook* en ligne sur le **site Jupyter** en cliquant sur le bouton “Try it in your browser”.

L'archive `notebooks.zip`

Outre cette documentation et un fichier texte contenant des indications sur les évolutions des versions des exercices (`VERSION.txt`), le lecteur dispose d'une archive contenant les 28 exercices du livre et plus d'une centaine d'exercices supplémentaires. Tous les exercices sont présentés sous forme de notebook, avec leur correction.

L'archive contient un répertoire `Notebooks`, dans lequel chaque chapitre (numéro NN) a son propre sous-répertoire `chapitre_NN`. On y trouve :

- un notebook `chapNN_exos.ipynb`, contenant les énoncés des exercices supplémentaires ;
- un notebook `livre_chapNN.ipynb`, contenant les énoncés des exercices du livre ;
- éventuellement plusieurs autres fichiers de données nécessaires à la réalisation des exercices supplémentaires ou à la présentation des énoncés.

Le répertoire “Notebooks” contient aussi :

- un notebook `navigation.ipynb`, qui permet de passer directement d'un notebook à un autre ;
- un répertoire `Corrections`, dans lequel on trouve les solutions de tous les exercices ;
- un répertoire `Sources`, contenant les principaux fichiers scripts de l'ouvrage.

Installation

L'annexe B du livre décrit en détail l'installation de *Pyzo*, de *miniconda* et de *Jupyter Notebook*.

```
Python 3.6.1 |Continuum Analytics, Inc.| (default, May 11 2017, 13:09:58) on linux (64 bits).
This is the Pyzo interpreter. Failed to integrate event loop for PYQT4: No module named 'PyQt4'
Type 'help' for help, type '?' for a list of *magic* commands.

>>> ?
List of *magic* commands:
?          - show this message
?X or X?   - show docstring of X
??X or X?? - help(X)
cd         - show current directory
cd X       - change directory
ls         - list current directory
who        - list variables in current workspace
whos       - list variables plus their class and representation
timeit X   - times execution of command X
open X     - open file X or the Python module that defines object X
run X      - run file X
pip        - manage packages using pip
conda      - manage packages using conda
db X       - debug commands
cls        - clear screen
notebook   - launch the Jupyter notebook
```

Figure 1: Liste des commandes *magiques*

Utilisation de *Jupyter Notebook*

Comment appeler *Jupyter Notebook* ?

La fenêtre **Shells** de *Pyzo* est un interpréteur Python amélioré qui fournit un jeu de commandes dites *magiques* (voir les figures 1 et 2). La dernière de ces commandes permet de lancer *Jupyter Notebook*. L'annexe B du livre vous indique comment sécuriser cet accès.

L'interface de *Jupyter Notebook*

Votre navigateur s'affiche et présente, dans l'onglet *Files*, la liste des fichiers du répertoire de lancement. Notons que l'onglet *Running* affiche la liste des notebooks actifs.

Note importante : Les notebooks actifs devront être soigneusement sauvegardés (clic sur l'icône de disquette ou **Ctrl + s**) et fermés (menu **File/Close and Halt**) avant de quitter la session — cela assure que le processus Python lié à l'ouverture du notebook est bien terminé. L'opération de sauvegarde n'est pas immédiate, il faut bien attendre sa fin.

Les notebooks sont repérés par l'icône d'un petit carnet (l'icône est verte si le notebook est ouvert et actif). En haut et à droite, le bouton *New* permet de lancer un nouveau notebook avec l'interpréteur Python 3.

L'interface d'un notebook

Un notebook est constitué d'une succession de cellules caractérisées par leur **type** et leur **mode**.

Une cellule est du **type markdown** ou du **type code**. Les cellules peuvent être dans le **mode commande** ou le **mode édition**. Un liseré de couleur repère la cellule sélectionnée : bleu si la cellule est en mode commande et vert si elle est en mode édition.

- On utilise le **mode édition (liseré vert)** pour écrire du code ou du texte. On sélectionne une cellule de type code par un simple clic et une cellule de type markdown par un double clic. Une icône de crayon apparaît à droite de la barre de menu.
- On utilise le **mode commande (liseré bleu)** pour gérer les cellules : créer, couper, copier, coller, naviguer, exécuter, interrompre. On passe en mode commande en cliquant en dehors d'une cellule ou en appuyant sur la touche **Échap**.

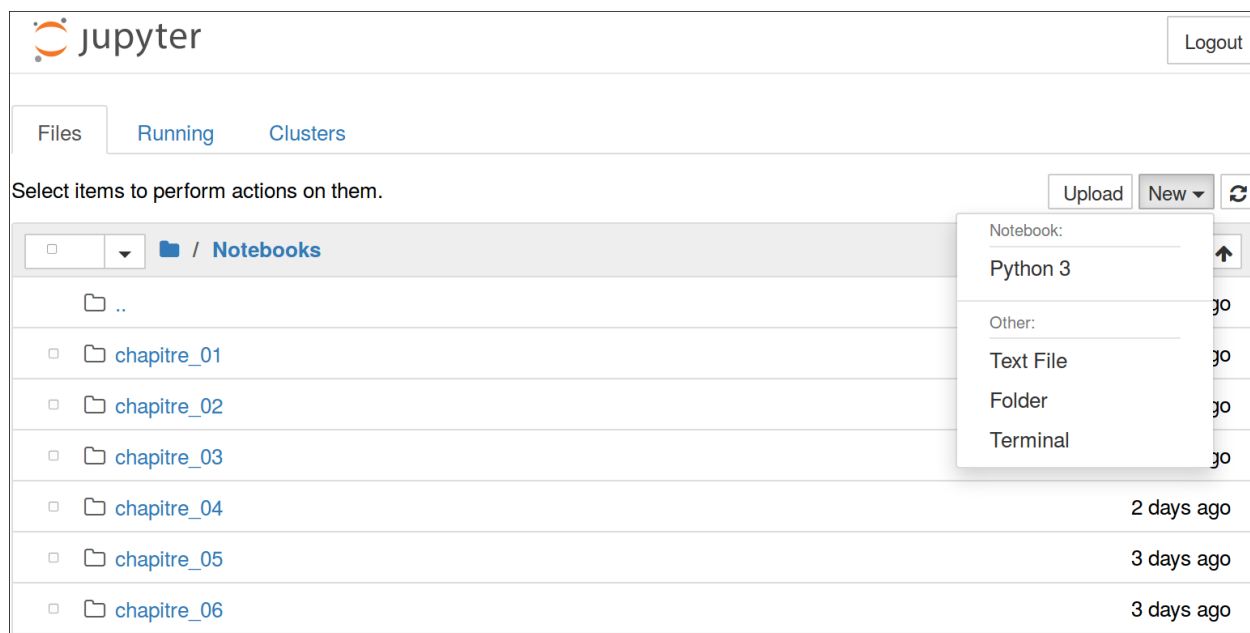


Figure 2: Interface principale de *Jupyter* et demande d'ouverture d'un nouveau notebook

Le mode édition (vert)

Ce mode est principalement utilisé pour produire du texte *enrichi* grâce au **language markdown**.

Le mode commande (bleu)

Les cellules pour écrire du code sont repérables par le prompt `In []`, le numéro entre crochet est un compteur incrémenté à chaque sollicitation de l'interpréteur. Les cellules reçoivent du code Python 3 (comme c'est rappelé en haut et à droite de la fenêtre) et sont interprétées par l'une des commandes suivantes :

- **Ctrl + Enter** : exécute la cellule ;
- **Shift + Enter** : exécute la cellule et sélectionne la cellule suivante. L'appui répété de cette touche permet ainsi d'exécuter pas à pas toutes les cellules du notebook ;
- **Alt + Enter** : exécute la cellule et insère une nouvelle cellule juste en dessous ;
- **Ctrl + s** : sauvegarde le notebook.

L'aide en ligne

Chaque notebook affiche une barre de menu qui comporte un bouton **Help**. On y trouve principalement :

- **User Interface Tour** : survol graphique des éléments de l'interface ;
- **Keyboard Shortcuts** : tableau des raccourcis des modes commande et édition ;
- **Notebook Help** : didacticiels et exemples ;
- **Markdown** : le langage markdown ;
- Des liens vers les documentations Python, IPython, NumPy, SciPy, Matplotlib, SimPy, pandas.

Comment faire et contrôler les exercices

L'utilisateur décompresse l'archive `notebooks.zip` dans le répertoire de son choix, dans lequel l'arborescence décrite plus haut sera déployée dans un répertoire **Notebooks**.

Chapitre 2 : "La calculatrice Python"

« *Le plus long des voyages commence par le premier pas.* »

Lao Tseu

CE QU'IL FAUT RETENIR

- Python comporte deux modes d'exécution : le *mode interprété* et le *mode script*.
- Les données peuvent être nommées *via* des variables (identificateurs), les mots réservés de Python ne doivent pas être utilisés comme nom de variable.
- Python comporte deux types de données entiers : `int` et `bool`, et deux types de données flottants : `float` et `complex`.

Figure 3: Interface d'un notebook d'exercices

exo02_19

Affecter les variables `temps` et `distance` par les valeurs 6.892 s et 19.7 m.

Calculer et afficher la valeur de la vitesse ($vitesse = \frac{distance}{temps}$).

Améliorer l'affichage en imposant un chiffre après le point décimal, sans oublier l'unité !

```
In [19]: # exo02_19
```

Figure 4: Exemple d'exercice dans un notebook

À partir du Jupyter Notebook démarré, l'utilisateur navigue jusqu'au répertoire **Notebooks** précédemment décompressé et ouvre les exercices du chapitre désiré.

Tous les notebooks des exercices supplémentaires sont organisés de la même façon :

- un titre (par exemple « Chapitre 3 : "Contrôle du flux d'instructions" ») suivi d'une citation ;
- une rubrique "Ce qu'il faut retenir" rappelant les points clés du chapitre ;
- une série d'exercices numérotés comprenant une cellule pour l'énoncé et une ou plusieurs cellules de code pour que l'utilisateur fasse l'exercice ; chaque exercice est noté de zéro à deux étoiles suivant son niveau de difficulté.

Les solutions des exercices sont regroupées dans le répertoire **Corrections** qui contient deux notebooks par chapitre, un pour les exercices du livre et un pour les exercices supplémentaires.