

tuto_Markdown

September 13, 2018

1 Une brève introduction aux Notebooks Jupyter pour Python3.

Ceci est un notebook Jupyter. **Python** est un langage de programmation, et un **notebook Jupyter** est un document interactif qui contient du texte (comme ce que vous êtes en train de lire maintenant) et du code Python. Les notebooks Jupyter seront utilisés tout au long du tutoriel, et ceci est une brève introduction à cette interface.

1.1 Table des matières

1. Cellules d'un notebook
2. Syntaxe Markdown
3. Sauver, charger, imprimer

1.2 Cellules d'un notebook

Cliquez dans la cellule ci-dessous, dans la partie à droite du prompt `In [2]` : Une bordure devrait apparaître autour de la cellule, avec une barre verte sur le côté, indiquant que la cellule est active. Puis appuyez sur la touche *shift-Enter* (tenir enfoncée la touche *shift* tout en appuyant sur la touche *Enter*) du clavier pour **évaluer** la cellule.

```
In [ ]: 2+2
```

Si un message comme `Out [1]: 4` apparaît, alors tout marche correctement !

Les notebooks Jupyter sont constitués de **cellules**. Il y a deux types principaux de cellules : les **cellules Markdown** and **cellules de Code**. Le contenu textuel sera contenu dans des cellules Markdown comme celle-ci. Le contenu de programmation sera contenu dans des cellules de Code, comme la cellule `2+2` ci-dessus.

Si une cellule a une bordure verte, ceci veut dire que vous êtes en train de l'éditer. Si elle a une bordure bleue, c'est qu'elle est sélectionnée (et peut être coupée-collée, etc.) mais pas éditée. Dans les autres cas elle n'est pas active. Pour éditer une cellule de Code, cliquez n'importe où dans sa zone d'évaluation. Par exemple, vous pouvez cliquer la cellule `2+2`, changer les nombres en `2+3` et la re-évaluer avec *shift-Enter*.

Pour éditer une cellule Markdown comme celle-ci, double-cliquez n'importe où dans le corps de la cellule. Vous pouvez écrire du Markdown comme n'importe quel document text, pour la plus grande partie. Quand vous avez fini d'éditer la cellule Markdown, appuyez sur *shift-Enter*.

Pour convertir une cellule de Code à Markdown ou inversement, vous devriez voir un menu déroulant avec "Markdown" ou "Code" visible, juste en haut du notebook. Vous devez juste sélectionner le type de cellule choisi dans le menu déroulant. Comme exercice, changez la cellule ci-dessous en Markdown, et écrivez une phrase ou deux. Evaluer avec *shift-Enter*.

```
In [ ]: # Convertissez cette cellule en cellule Markdown, et écrivez quelque chose
```

Maintenant, convertissez la cellule ci-dessous en cellule Code, et utilisez la pour calculer $999 * 999$.

(Je suis une cellule markdown. Convertissez moi en cellule Code et calculez $999 * 999$)

Souvent vous voudrez **insérer une nouvelle cellule** au-dessus ou au-dessous d'une cellule donnée. Pour cela, vous devez sélectionner la cellule en cliquant en marge sur sa gauche, de façon à obtenir une bordure bleue.

Une fois la bordure bleue visible, vous pouvez ajouter une nouvelle cellule **au-dessus** en appuyant sur la touche **a** (comme above). Vous pouvez ajouter une nouvelle cellule **au-dessous** en appuyant sur la touche **b** (comme below). Essayez d'ajouter une nouvelle cellule au-dessus puis au-dessous de la cellule Markdown ci-dessous. Les cellules nouvellement créées sont probablement des cellules Code par défaut.

Au lieu d'appuyer sur les touches **a** ou **b**, vous pouvez aussi appuyer sur la touche **x**. Ceci détruira la cellule ; soyez donc prudent et sauvegardez souvent.

(Je suis une cellule Markdown. Ajoutez une nouvelle cellule au-dessus et au-dessous de moi.)

Une petite note : il y a quelques **cellules invisibles** dans ce notebook, et dans d'autres notebooks de ce tutoriel. En fait, une de ces cellules furtives se trouve juste au-dessous ! C'est une cellule Markdown, comme vous pouvez le voir en double-cliquant dessus. Son contenu est ``, c'est un bout de code HTML qui permet d'activer les liens dans la table des matières. Ne modifiez pas ces cellules Markdown, à moins que vous ne vouliez changer les liens de la table des matières.

1.3 Markdown de base pour les notebooks Jupyter

Dans ce tutoriel vous allez apprendre à coder en Python. Au fur et à mesure de votre apprentissage, vous expérimenterez les cellules Code, et écrirez les réponses aux questions et d'autres notes dans les cellules Markdown. Markdown est un langage d'édition de texte, une sorte d'extension de HTML. Voici un petit guide pour écrire en Markdown.

La plupart du temps, vous pouvez écrire du Markdown à peu près comme n'importe quel texte. Une petite différence est que pour commencer une nouvelle ligne, vous devez insérer une ligne vide supplémentaire dans le Markdown. Double-cliquez cette cellule pour voir comment le nouveau paragraphe ci-dessous est créé.

Markdown possède de nombreuses possibilités : on peut utiliser différentes polices de caractères, des listes, des en-têtes, des liens, etc..

1.3.1 Polices de caractères

Pour mettre du texte en *italique* ou en **gras**, vous entourrez le texte par des astérisques, simples ou doubles. Double-cliquez dans cette cellule pour voir comment c'est fait. Puis, créez une nouvelle cellule ci-dessous pour expérimenter.

1.3.2 Listes

Il est souvent utile de créer des listes. Les deux sortes de liste qu'on utilisera sont les listes numérotées ou non-numérotées. En Markdown c'est facile ; pour une liste numérotée, démarrez chaque nouvelle ligne par un numéro. Double-cliquez cette cellule pour voir. 1. Première ligne. 2. Deuxième ligne. 3. Troisième ligne.

Pour une liste non-numérotée, commencez chaque nouvelle ligne par un tiret – (quelques autres symboles marchent aussi). Double-cliquez pour voir.

- Première ligne.
- Deuxième ligne.
- Troisième ligne.

Maintenant, retour au texte normal.

Vous pouvez imbriquer vos listes. Mais les gens utilisent trop souvent les listes imbriquées dans leurs écrits et présentations. Les listes sont utiles mais on doit utiliser les paragraphes plus souvent. Si vous voulez imbriquer des listes, double-cliquez cette cellule pour voir son code Markdown, et les indentations utilisées.

1. Première ligne.
 - Première sous-ligne
 - Deuxième sous-ligne
2. Deuxième ligne.
3. Troisième ligne.

Maintenant, retour au texte normal.

1.3.3 Titres

Il y a différents types de titres, c'est-à-dire des titres, sous-titres, qui seront typiquement rendus en caractères gras de différentes tailles. Pour un titre de haut niveau, utilisez un simple hashtag #. Au niveau suivant, utilisez un double hashtag ##, et ainsi de suite. Double-cliquez pour voir. There are different levels of headings, i.e., headings, subheadings, etc., which typically render as boldface text of different sizes. To make a top-level heading, a single hashtag # is used. The next level heading is created with a double hashtag ##, and so on. Double-click this cell to see how it was done.

2 Titre de haut niveau.

2.1 Titre au niveau suivant.

2.1.1 Titre de troisième niveau.

Titre de quatrième niveau.

2.1.2 Liens

Enfin, nous discutons comment insérer des liens internet dans votre code Markdown. Par exemple, voici un lien vers la page du livre [An Illustrated Theory of Numbers](#). Le texte affiché doit être entouré de crochets et suivi de l'URL (adresse web) entre parenthèses. Double-cliquez pour voir.

2.1.3 Conclusion

On peut faire beaucoup d'autres choses en Markdown. En fait Markdown est une extension de HTML, donc on peut mettre n'importe quel contenu web dans une cellule Markdown. Une rapide recherche Google permet de trouver beaucoup de tutoriels Markdown, mais on doit préciser Markdown and Jupyter pour avoir une version plus particulièrement adaptée à Jupyter. On peut ainsi incorporer dans son texte des formules mathématiques

$$e^x = \sum_{i=0}^{\infty} \frac{1}{i!} x^i$$

ou du code Python

```
import abc
def f(x):
    """a docstring"""
    return x**2
```

ou encore des tableaux

First Header	Second Header
Content Cell	Content Cell
Content Cell	Content Cell

There is much much more you can do with Markdown. In fact, Markdown is a superset of HTML, so you can put just about any basic web content in a Markdown cell. A quick google search will turn up lots of Markdown guides, and it's best to search for Markdown and Jupyter to find a guide for Jupyter's particular flavor of Markdown (which differs a bit from WordPress markdown, for example).

2.2 Sauvegarder, charger, et imprimer des notebooks

Vous pouvez sauvegarder, charger et imprimer des notebooks Jupyter. En fait, c'est probablement un bon moment de sauvegarder ce notebook depuis que vous avez comencé à éditer le Markdown. Pour cela, aller dans le menu "File... Save and Checkpoint".

Le format standard des fichiers Jupyter est `.ipynb` qui signifie "iPython Notebook", car Jupyter s'appelait iPython. Si vous allez dans le menu "File... Download as...", alors "IPython Notebook" sera le premier choix. Vous pourrez toujours recharger votre notebook et l'éditer, si vous le sauvegardez en iPython Notebook.

Vous pouvez aussi le sauvegarder au format HTML. HTML est le code utilisé par les navigateurs web pour rendre les pages web, et donc c'est un bon format pour partager avec n'importe qui ayant un navigateur. Aussi, votre navigateur peut probablement imprimer les fichiers `.html` avec une bonne qualité ; c'est donc un bon format pour imprimer. Vous pouvez aller directement dans le menu "File... Print preview" pour voir à quoi ressemblera le notebook une fois imprimé.

Pour ouvrir un notebook Python, vous pouvez aller à "File... Open" ou retourner à votre onglet Jupyter original (s'il est toujours là) et parcourir les répertoires pour trouver des fichiers `.ipynb`. Mais Jupyter doit être d'abord démarré avant de charger des notebooks Python dans le

navigateur.

Un autre outil à connaître est [GitHub](#). GitHub est un site web qui permet aux utilisateurs de partager et développer toutes sortes de code. Il est utilisé pour le développement collaboratif de software, et a beaucoup de propriétés intéressantes comme le “versioning” (garder la trace des différentes versions du code au cours du développement). Vous pouvez utiliser librement GitHub pour stocker vos notebooks. GitHub demande un peu de pratique, mais ça en mérite la peine sur le long terme.

Les tutoriels notebooks sont stockés sur GitHub pour plusieurs raisons. D’abord, ils peuvent être facilement édités et améliorés. Ensuite, ils sont librement et publiquement accessibles. Allez sur [la page GitHub](#) pour voir les notebooks de ce cours. Enfin, même s’ils sont stockés au format `.ipynb`, n’importe qui peut voir les notebooks Python sur le web avec le [Jupyter Notebook Viewer](#). Par exemple vous pouvez utiliser le Jupyter Notebook Viewer pour voir [ce notebook en ligne](#).

Maintenant, vous en savez assez pour commencer à programmer en Python ! Téléchargez les notebooks et **bonne programmation !**