

Webinaire 03 et 4 juin 2020

JupyterLab dans Lycée connecté

Documents de formation disponibles sur
<https://github.com/luciobx/webinaire>

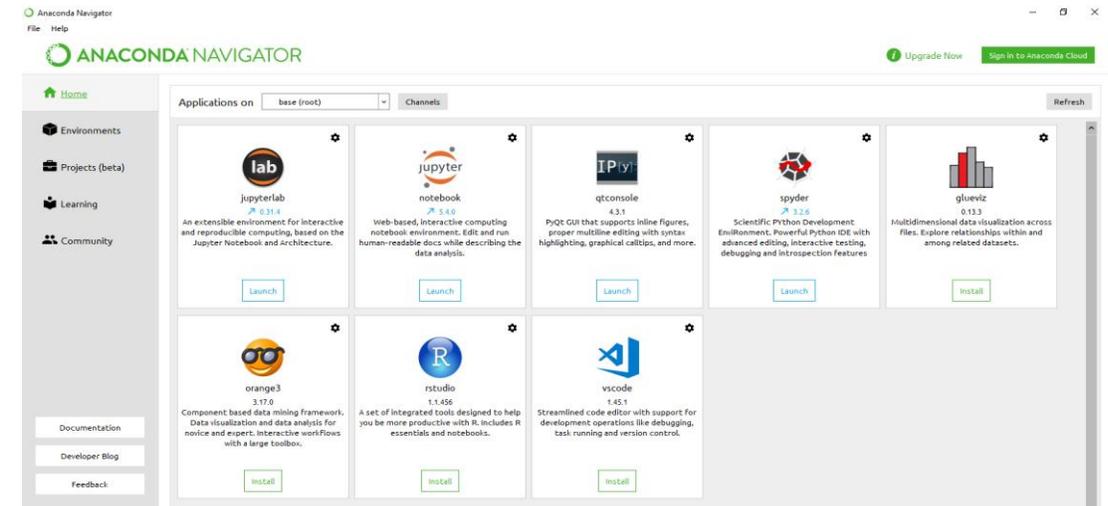
• Présentation



- Le projet Jupyter pour Julia, Python et R, trois langages utilisés dans la science des données (data science)
- Des fichiers bloc-notes d'extension .ipynb (iPYthon Notebook).
- Notebooks, Blocs-Note, Cahier Jupyter
- Du texte, des images, des formules mathématiques et du code informatique exécutable.
- Manipulables interactivement via un navigateur

• Différence entre le cahier Jupyter et le laboratoire Jupyter

- Jupiter Notebook
- Projet historique
- Installation en local sur une machine personnelle (pip install notebook)
- Disponible avec la distribution Anaconda



• Différence entre le cahier Jupyter et le laboratoire Jupyter

- Jupiter Notebook
- Projet historique
- Existence d'une version multi-utilisateur conçue pour les entreprises, les salles de classe et les laboratoires de recherche
- Un moyen simple de partager des cahiers Jupyter (nbviewer)



• Différence entre le cahier Jupyter et le laboratoire Jupyter

- JupyterLab est l'interface utilisateur Web de nouvelle génération pour le projet Jupyter.
- JupyterLab remplacera à terme le portable Jupyter classique.
- Installable en local (pip install jupyterlab)

- Binder (<https://mybinder.org/>)
- Colaboratory (<https://colab.research.google.com/notebooks/welcome.ipynb>)
- Azure Notebooks (<https://notebooks.azure.com/#>)

- Dans l'ENT Lycée Connecté.



Binder



Colaboratory



Azure

• **Différence entre le cahier Jupyter et le laboratoire Jupyter**

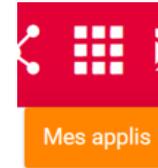
- Des différences de comportements
- Les modules installés dans Python ne sont pas les mêmes
- Le navigateur se comporte différemment en local et sur un site en ligne
- Multitudes de versions des environnements Jupiter

- Se connecter à l'environnement JupyterLab

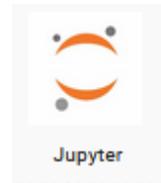
- Lycée Connecté <https://lyceeconnecte.fr/>



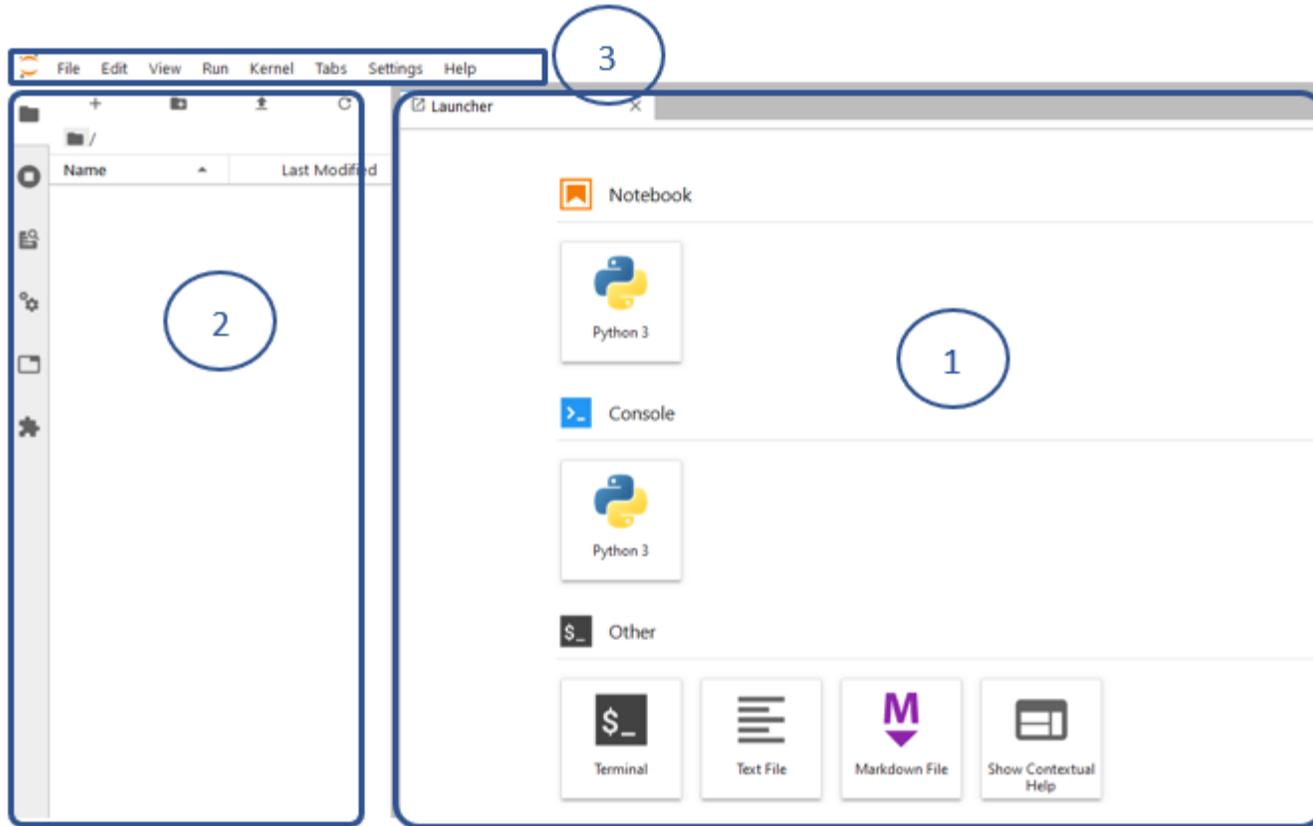
- Retrouver JupyterLab dans « Mes applis »



- Cliquer

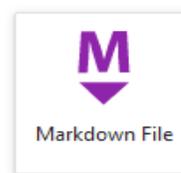
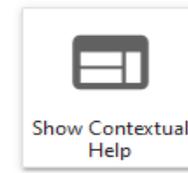
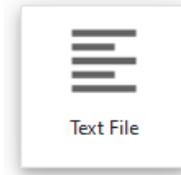
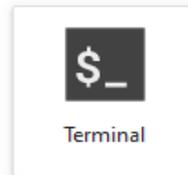
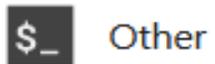
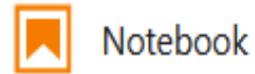


• L'environnement



- Une zone de travail contenant des onglets de documents et d'activités
- Une barre latérale gauche repliable
- Une barre de menus

• Trois domaines du lanceur

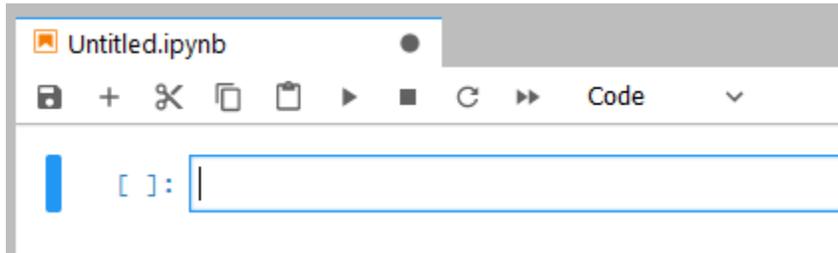


- Un outil d'écriture et d'exécution de Notebook
- L'accès à une console Python
- Des outils variés

• Un premier bloc-notes

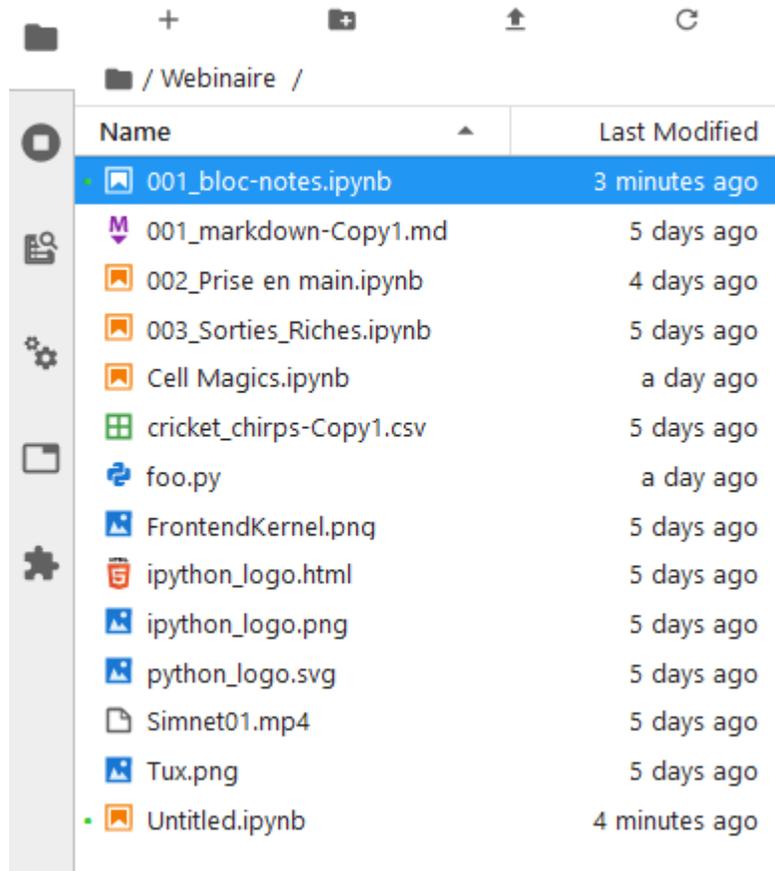


 Notebook



- Le seul langage installé est Python 3
- Le code est exécuté cellule par cellule.

• Organiser ses blocs-notes



Name	Last Modified
001_bloc-notes.ipynb	3 minutes ago
001_markdown-Copy1.md	5 days ago
002_Prise en main.ipynb	4 days ago
003_Sorties_Riches.ipynb	5 days ago
Cell Magics.ipynb	a day ago
cricket_chirps-Copy1.csv	5 days ago
foo.py	a day ago
FrontendKernel.png	5 days ago
ipython_logo.html	5 days ago
ipython_logo.png	5 days ago
python_logo.svg	5 days ago
Simnet01.mp4	5 days ago
Tux.png	5 days ago
Untitled.ipynb	4 minutes ago



- Déposer des fichiers par glisser déposer
- Créer des dossiers
- Editer des fichiers .txt ou .CSV

• Les cellules de code



```
[1]: a = 10
```

```
[2]: print(a)
```

```
10
```

```
[3]: for i in range(5):  
      print(a + i)
```

```
10  
11  
12  
13  
14
```

```
File "<ipython-input-2-83bcdcc5e1fc>", line 1  
  chaine = 'un caractère ou une chiane de caractère  
SyntaxError: EOL while scanning string literal
```

```
[ ]:
```

- Coloration syntaxique
- Les résultats ou les messages d'erreur sont affichés en tant que sortie de la cellule
- Exécutée dans l'ordre par l'utilisateur

• Fonctionnalités sur les cellules



Par défaut la cellule est en code

...

```
[ ]: print(a)
```

...

- Glisser et déposer les cellules pour réorganiser votre bloc-notes.
- Glisser les cellules entre les blocs-notes pour copier rapidement leur contenu
- Réduire et développer le code et la sortie à l'aide du menu Affichage ou du bouton de réduction bleu à gauche de chaque cellule.

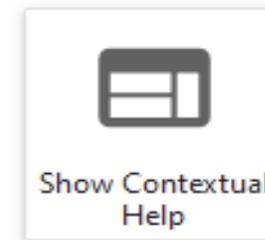
• Aide contextuelle



```
: chaîne = 'un caractère ou une chaîne de caractère'
type(chaine)
print(a)
print(a)
10
for i in range(10):
```

```
Init signature: type(self, /, *args, **kwargs)
Docstring:
type(object_or_name, bases, dict)
type(object) -> the object's type
type(name, bases, dict) -> a new type
Type:
Subclasses: ABCMeta, EnumMeta, _TemplateMetaclass, MetaHasDescriptors, _ProtocolMeta,
NamedTupleMeta, LexerMeta, StyleMeta, _NormalizerMeta, CachedMetaClass, ...
```

- les info-bulles (Shift Tab) donnent accès à l'aide contextuelle
- Cette zone peut être maintenue visible par l'outil zone d'aide contextuelle



• Cellule Markdown



En Markdown

- une liste
 - niveau 2
 - niveau3

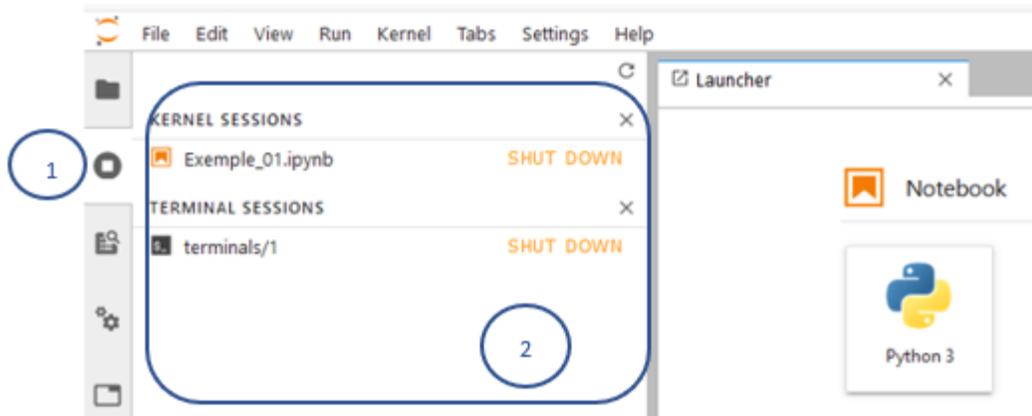
$$\frac{1}{T} \int_a^{a+T} f(t) dt$$

Voici le titre du tableau

Titre A1	Titre A2	Titre A3	Titre A4
Titre B1	Valeur B2	Valeur B3	Valeur B4

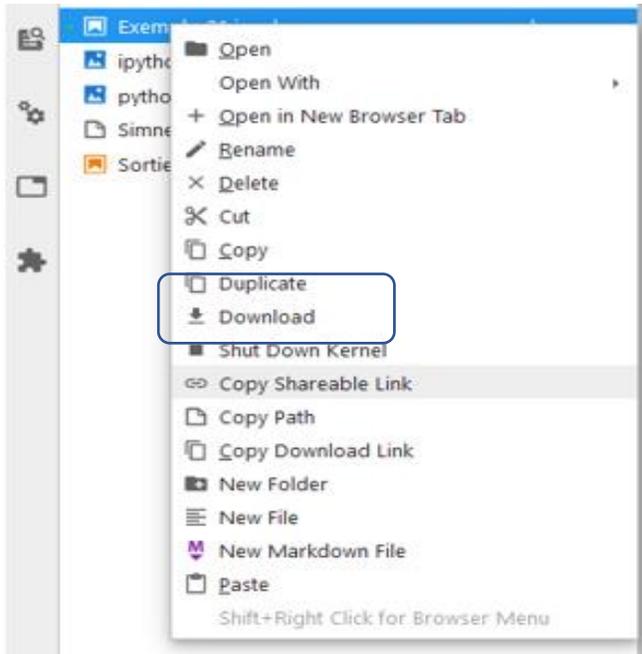
- Un moyen simple de spécifier les parties du texte à mettre en évidence (italique), en gras, les listes de formulaires, etc.
- Inclure les formules mathématiques de manière simple, en utilisant la notation LaTeX standard : $\$...\$$ pour les formules en ligne et $\$\$...\$\$$ pour les formules affichées.
- Inclure du HTML

• Gestion des noyaux et des terminaux



- Le panneau « En cours d'exécution » dans la barre latérale gauche affiche une liste de tous les noyaux et terminaux en cours d'exécution

• Partage de bloc-notes



- il est possible de télécharger le bloc-notes afin de le partager avec d'autres utilisateurs.
- Intégration dans Moodle à venir