

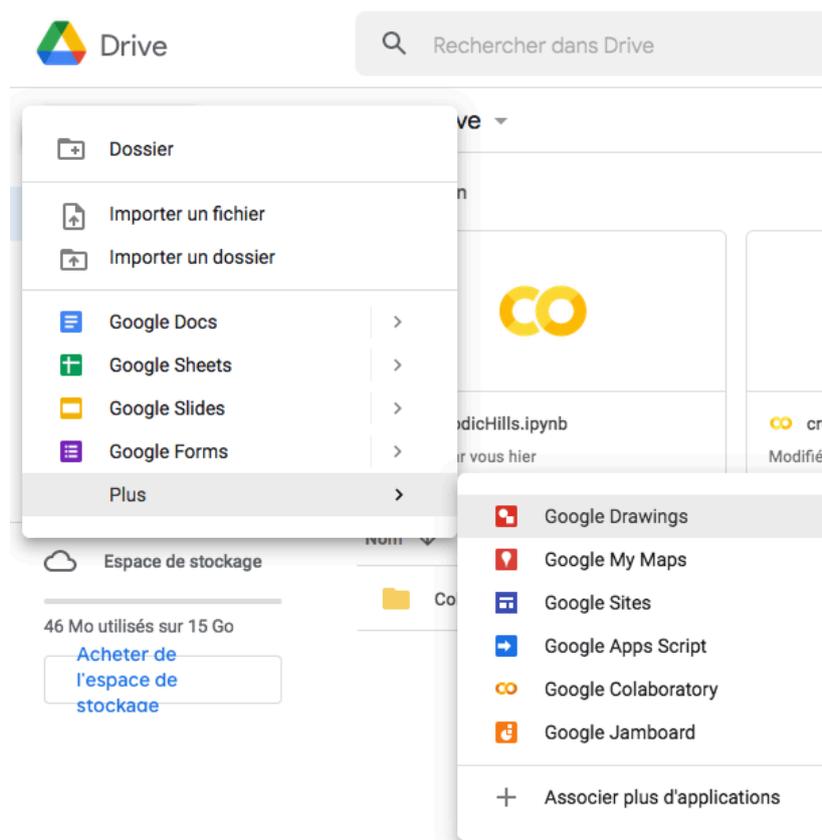
Mettre en ligne ses programmes Python et ses « Jupyter notebooks »

Olivier THUAL, Toulouse INP
Projet INP'CLUSION, 20 novembre 2021

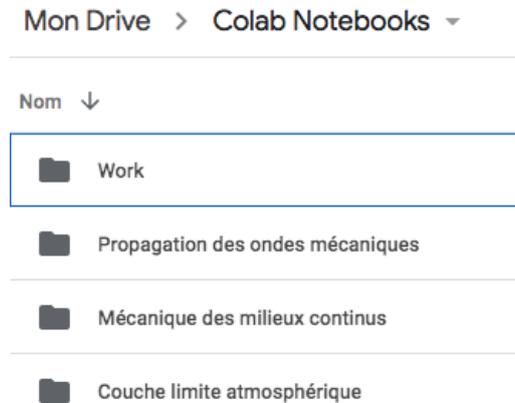
Vous souhaitez mettre en ligne des programmes Python pour que vos étudiants puissent les faire exécuter en les modifiant : ce tutoriel est pour vous. Grâce aux « Jupyter notebooks », vous pourrez ajouter du texte, des équations, des images ou des animations. Deux méthodes sont proposées ici : Colab, qui présente l'inconvénient de nécessiter des comptes Google, y compris pour les étudiants, ou (GitHub + Binder), qui est complètement libre mais un peu plus compliqué à mettre en place.

Google Colab :

Il vous faut tout d'abord disposer d'un compte Google. Personnellement, j'ai créé un compte spécifique pour tous mes Jupyter notebooks d'enseignement. Allez ensuite sur le « Google Drive » associé à ce compte : <https://drive.google.com>. Dans cet espace, créez un fichier « Google Colaboratory », en activant « Associer plus d'application » si ce choix n'est pas proposé dans la liste.



Les fichiers seront stockés dans un répertoire « Colab Notebooks ». Vous pouvez y créer des sous-répertoires les organiser :



Ces fichiers sont des « Jupyter notebooks » d'extension .ipynb que vous pouvez télécharger, par exemple pour une sauvegarde. Vous pouvez aussi téléverser de tels fichiers après les avoir créés en local, par exemple avec l'application Jupyter de l'interface Anaconda. À partir du Google Drive, vous pouvez partager un lien en permettant à tous les utilisateurs le possédant d'y accéder :



Obtenir le lien

Avec ce lien, **tous les internautes** peuvent consulter l'élément

[Modifier](#)

[Copier le lien](#)

À titre d'exemple, voici un de mes liens publics pour le cours « Propagation des ondes mécaniques » que j'enseigne en en 2A à Toulouse :

<https://colab.research.google.com/drive/13SdbSEyUr1LChHYkNXX937dJnKSyyTk1?usp=sharing>

Avec ce lien, vous disposez d'une copie de mon Jupyter notebook sur votre compte Google et pouvez, à l'instar des étudiants, modifier le programme comme vous le souhaitez, sans affecter mon original.

GitHub + Binder :

Il vous faut tout d'abord vous créer un compte sur GitHub : <https://github.com>. Créez alors un « repository » (dépôt) et nommez-le.

Overview Repositories 4 Projects Packages

Find a repository... Type Language Sort New

Owner * Repository name *

thual / croix-saint-andre ✓

Great repository names are short and memorable. Need inspiration? How about 1

Description (optional)

Croix de Saint-André : émission d'ondes internes par un obstacle oscillant'

Public Anyone on the internet can see this repository. You choose who can commit.

Une fois le « repository » créé, cliquez sur « uploading an existing file » pour téléverser votre Jupyter notebook.

Quick setup — if you've done this kind of thing before

Set up in Desktop or HTTPS SSH https://github.com/thual

Get started by [creating a new file](#) or [uploading an existing file](#). We recommend

Cliquez sur « Commit ». Il faut ensuite ajouter un fichier texte « requirements.txt » en cliquant sur « Add file » / « Create new file » ou « Upload files » s'il existe sur votre poste de travail :

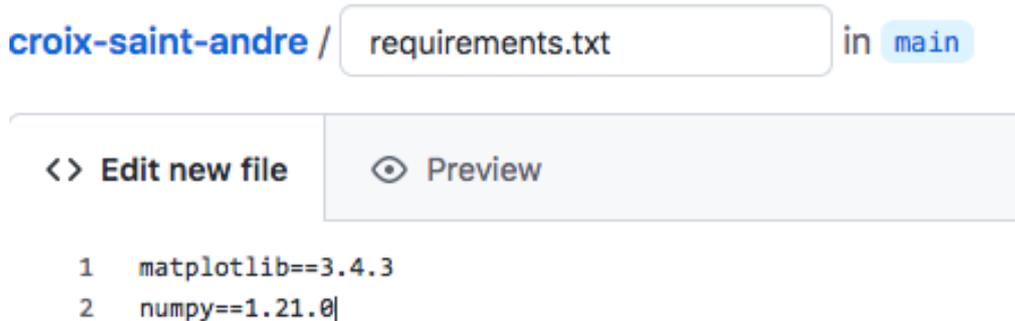
main 1 branch 0 tags Go to file Add file Code

thual Add files via upload 5f 1 commit

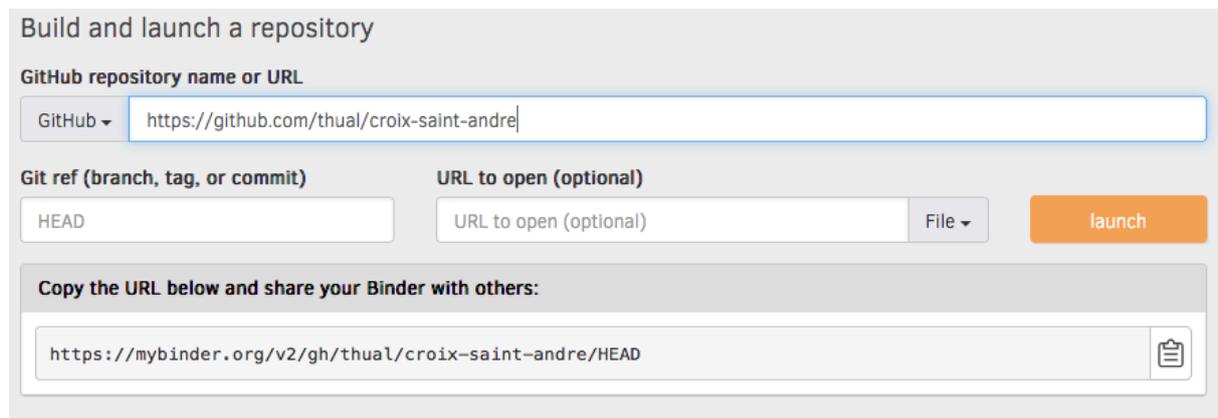
README.md	Create README.md	3 minutes ago
croixSaintAndrew.ipynb	Add files via upload	3 minutes ago

Dans ce fichier, il faut décrire les bibliothèques python dont a besoin le programme avec leurs numéros de version, comme par exemple :

```
matplotlib==3.4.3  
numpy==1.21.0
```

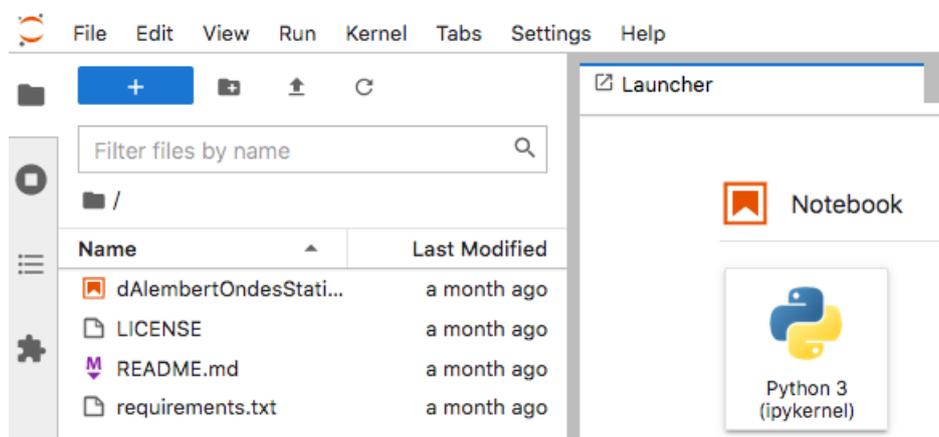


Il faut ensuite aller sur <https://mybinder.org>, y taper l'adresse du repository et appuyer sur launch :



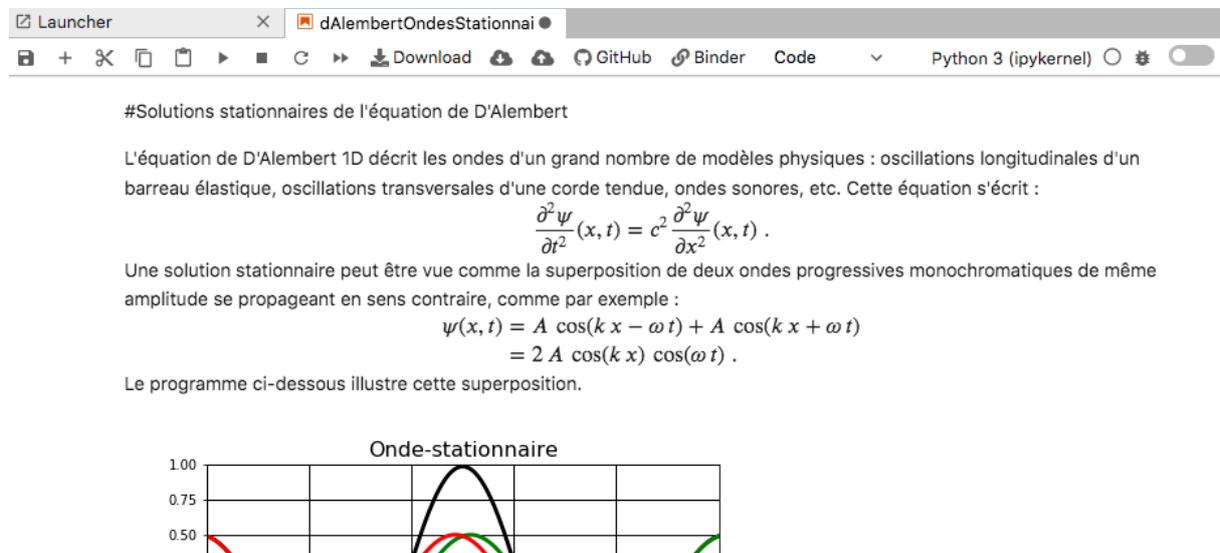
Il suffit ensuite de recopier le lien de partage comme par exemple celui-ci :

<https://mybinder.org/v2/gh/thual/propagation-ondes-mecaniques/HEAD>



Arrivés ici avec ce lien, les étudiants doivent double cliquer sur le Notebook (orange).

Ils entrent alors dans une copie de votre Jupyter Notebook :



#Solutions stationnaires de l'équation de D'Alembert

L'équation de D'Alembert 1D décrit les ondes d'un grand nombre de modèles physiques : oscillations longitudinales d'un barreau élastique, oscillations transversales d'une corde tendue, ondes sonores, etc. Cette équation s'écrit :

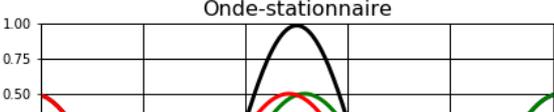
$$\frac{\partial^2 \psi}{\partial t^2}(x, t) = c^2 \frac{\partial^2 \psi}{\partial x^2}(x, t) .$$

Une solution stationnaire peut être vue comme la superposition de deux ondes progressives monochromatiques de même amplitude se propageant en sens contraire, comme par exemple :

$$\begin{aligned} \psi(x, t) &= A \cos(k x - \omega t) + A \cos(k x + \omega t) \\ &= 2 A \cos(k x) \cos(\omega t) . \end{aligned}$$

Le programme ci-dessous illustre cette superposition.

Onde-stationnaire



et peuvent modifier les lignes de votre code python :

Exécutez cette première partie du code pour définir les fonctions !

```
[ ]: import numpy as np
import matplotlib.pyplot as plt

def zfi(x, le=2):
    miss=le-len(str(x))
    a='0'*miss+str(x)
    return a
```

Pour exécuter les différentes portions du code, il faut cliquer sur Run :



Run the selected cells and advance

#Solutions stationnaires de l'équation de D'Alembert

Attention : si vous modifiez vos fichiers sur le repository GitHub, il vous faut refaire la procédure « launch » sur MyBinder. Le lien public reste le même.

Pour toute question et apport d'informations : olivier.thual@toulouse-inp.fr