

# Enseigner avec un calepin Jupyter

Pacôme DELVA

SYRTE, Observatoire de Paris-PSL, Sorbonne Université, CNRS, LNE

Café Capsule

Paris, mercredi 19 octobre 2022



# Pourquoi enseigner avec Jupyter ?

Votre cours contient des données, des équations, des simulations, des modèles, des algorithmes ... alors les calepins Jupyter sont des outils qui peuvent enrichir votre cours avec plusieurs buts :

- Améliorer l'**engagement** et la **motivation** des étudiant·es
- Favoriser leur **participation**
- Accéder à une meilleure **compréhension** du cours
- De l'**entraînement** supplémentaire, avec un possible retour de l'enseignant·e
- Des compétences très utiles pour leur **carrière**

## Ce que nous dit la recherche



### DEUX MOTEURS DE L'APPRENTISSAGE

#### Motivation

Pour motiver les étudiants, il importe de créer les conditions favorables à l'apprentissage (Viau, 2009).

#### Engagement

L'engagement d'un étudiant est élevé s'il a une perception de contrôle sur les activités d'apprentissage et une perception positive de sa compétence, c'est-à-dire que les activités d'apprentissage offrent un défi à la hauteur de ses habiletés (Viau, 2009).

Denis Bédard, Le Tableau vol 9-4, 2020

# C'est quoi Jupyter ?

- **Jupyter** permet de réaliser des **calepins** (ou *notebooks*) contenant à la fois du texte narratif, du code informatique, des équations, des graphiques, des vidéos, des images, etc...
- Démonstration : *Mon premier graphique*
- Ces calepins sont souvent utilisés en sciences pour **explorer et analyser des données**, mais d'autres domaines sont possibles (écriture, musique, ingénierie, etc...)
- Ils sont utilisés dans des **situations diverses** avec de **nombreuses possibilités d'activités** : voir le site Teaching and Learning with Jupyter.
- Fonctionne avec **plusieurs langages de programmation** : Python 3 et R pour le moment (mais c'est possible avec plus de 40 langages)

# Pourquoi Python ?

*Python est un langage portable, extensible, gratuit, qui permet différentes approches de la programmation. Python est développé depuis 1989 par Guido van Rossum et de nombreux contributeurs bénévoles.*

- Python est **portable** : on peut l'installer sur beaucoup de systèmes d'exploitation différents
- Python est **gratuit** : cependant on peut l'utiliser dans des projets commerciaux
- Python convient pour de petits scripts (=programmes) aussi bien que pour des programmes très complexes
- La syntaxe de Python est **simple** et utilise des types de données évoluées → programmes compacts et lisibles
- Python est **extensible** : facilement interfaçable avec des programmes d'autres langages (C, Fortran, ...)
- De **nombreuses bibliothèques** de programmes ou **modules** sont développés pour différents domaines (plusieurs milliers !)

## Modélisation Numérique en Physique (LU2PY222)

- UE de L2 de 6 ECTS, avec de petits groupes de 12 étudiant·es, environ 200 étudiant·es sur l'année
- Objectifs principaux : science des données, modélisation numérique et gestion de projet
- Des étudiant·es avec des **niveaux initiaux très différent** en informatique
- Utilisation des calepins Jupyter :
  - **Cours** inversés
  - **Exercices** corrigés
  - **Compte-rendu** de mini-projets (retour de l'enseignant·e)
  - Compte-rendu de projet (en binome sur un sujet libre, 4 semaines)

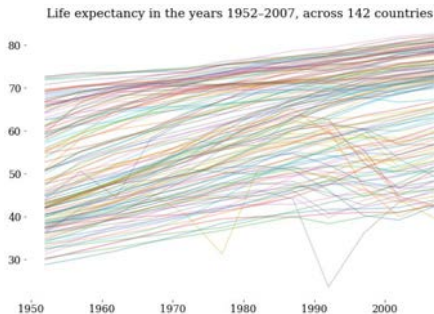
<https://phys-mod.github.io>

## Physique Expérimentale 2 (LU2PY215)

- UE de L2 de 3 ECTS, avec de petits groupes de 12 étudiant·es, environ 150 étudiant·es au S4
- Objectifs principaux : statistiques et probabilités, analyse de données et expérimentation
- Utilisation des calepins Jupyter :
  - **Compte-rendu** de travaux pratiques (retour de l'enseignant·e)

# Motivation et engagement

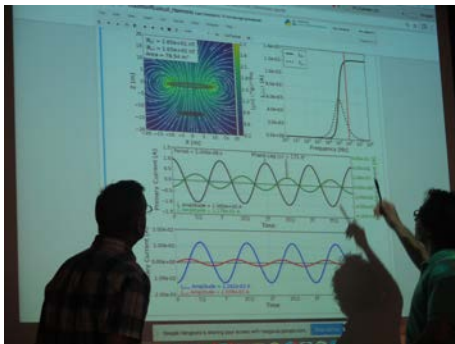
Travailler avec des données réelles et leurs représentations engagent les étudiant-es dans une **réflexion ancrée dans le réel**. Cela favorise la motivation et l'engagement des étudiant-es, crée des liens et des ponts entre différentes disciplines, ou avec leur culture générale.



from [https://nbviewer.org/github/engineersCode/EngComp2\\_takeoff/blob/master/notebooks\\_en/4\\_Life\\_Expectancy.ipynb](https://nbviewer.org/github/engineersCode/EngComp2_takeoff/blob/master/notebooks_en/4_Life_Expectancy.ipynb)

# Participation

Le calepin peut être utilisé comme une illustration du cours de façon **interactive**, en montrant en direct les effets de différents paramètres sur un modèle.



Dr. Douglas Oldenburg en discussion avec des étudiant·es, en cours de géophysique. from <https://jupyter4edu.github.io/jupyter-edu-book>



# Améliorer la compréhension du cours

L'utilisation de calepins permet de diviser la construction du savoir en petits éléments, pour permettre aux étudiant·es d'aller à leur rythme et vérifier qu'ils et elles ont bien compris chaque étape, avec une difficulté/complexité croissante.

## Exemple

L'équation de Navier-Stokes en 12 étapes par Lorena Barba :

<https://jose.theoj.org/papers/10.21105/jose.00021>

Design pattern :

- 1 Décomposez en petites étapes
- 2 Assembler les petites étapes en étapes plus importantes
- 3 Ajoutez un récit
- 4 Lien vers la documentation
- 5 Intercaler des exercices faciles
- 6 Épicez avec des questions plus difficile
- 7 Publiez en ligne

Travailler avec les calepins permet de développer une méthode pour analyser, comprendre et résoudre des problèmes.

- 1 **Décomposition** : décomposer les données, les processus ou les problèmes en parties plus petites et gérables
- 2 **Reconnaissance de motifs** : observer des modèles, des tendances et des régularités dans les données
- 3 **Abstraction** : identifier les principes généraux qui génèrent ces motifs
- 4 **Conception d'algorithmes** : développement des algorithmes étape par étape pour résoudre ce problème et des problèmes similaires

# Utilisations possibles des calepins Jupyter (1/2)

- **Calepin de cours** : il est facile de changer un polycopié en calepins, puis d'ajouter des illustrations statiques ou interactives, ou du code. On peut faire un calepin avec un index qui mène vers les différents calepins de cours (chapitres). Il est possible ensuite de l'éditer sur papier ou en ligne.
- **Ajouter des exercices** : on peut ajouter des exercices simples tout au long du calepin, ce qui permet à l'étudiant de vérifier si il a bien compris une notion donnée. Il peut répondre directement dans le calepin.
- Calepin d'**exercices dirigés** : un énoncé de TD avec des exercices à faire directement dans le calepin, sous chaque question
- **Prise de note** : pour prendre des notes de cours, où l'on peut mettre directement des bouts de code illustratif

## Utilisations possibles des calepins Jupyter (2/2)

- **Calepin de simulation** : pour manipuler un modèle et ses différentes représentations
- **Compte-rendu de TP** : dans un calepin on peut inclure le schéma de l'expérience, structurer le rapport en différentes parties et sous-parties, lire les données, faire un fit, convertir le calepin en HTML ou pdf
- **Multimedia** : on peut inclure dans le calepin des vidéos (de cours, pour illustrer un expérience, faire un point de méthode), mais aussi du son, des liens vers d'autres ressources, etc...
- **Présentation** : il est possible d'utiliser un calepin pour une présentation en cours (fullscreen), ou bien le transformer en *slides* (reveal.js ou Rise)
- **Coder en direct** : dans un calepin permet d'aller plus lentement et de bien détailler chaque étape d'un algorithme

# Un exemple de classe inversée (LU2PY222)



La taxonomie des objectifs éducationnels selon Bloom.  
(image Wikipedia)

- 1 **Connaissance** : calepin de cours à faire en autonomie ; préparer des questions pour la séance en présence
- 2 **Compréhension** : questions simples à faire dans le calepin de cours ; quizz d'auto-positionnement après le cours
- 3 **Application/analyse** : calepin avec exercices dirigés, guidés par l'enseignant·e ; correction disponible
- 4 **Analyse/synthèse** : résolution de problème avec production d'un calepin de compte-rendu (difficulté croissante)
- 5 **Synthèse/évaluation** : projet de fin de semestre avec production d'un calepin de compte-rendu

23 *motifs pédagogiques* sont proposées sur le site jupyter4edu.

## Shift-Enter

- **Description** : le lecteur lit le calepin en tapant SHIFT+ENTER pour aller de cellule en cellule.
- Exemple, Objectif d'apprentissage, Audience, Format, Caractéristiques
- **Piège** : peu interactif. Un·e étudiant·e peut finir beaucoup plus vite que les autres. Il est possible de diviser le cours en plusieurs calepins, avec un calepin index.

## Remplir les blancs

- **Description** : écrire un code en laissant des blancs, à remplir par les étudiant·es ou bien par l'enseignant·e en direct pendant une présentation.
- **Exemple** : écrire une boucle, en laissant juste l'actualisation de la variable d'itération dans la boucle à remplir plus tard.
- Objectif d'apprentissage, Audience, Format, Caractéristiques
- **Piège** : peut être trop simple et peu engageant.

## Modifier le code

- **Description** : un code plus ou moins long est donné dans un calepin. Il est demandé de modifier certaines choses dans le code, avec des modifications plus ou moins grandes (*tweak*, *twiddle* and *frob*).
- **Exemple** : un modèle numérique avec résolution d'équations différentielles est donné. *tweak* : changer un des paramètres du modèle pour voir les effets. *twiddle* : modifier le graphique pour ajouter une courbe. *frob* : changer la méthode de résolution.
- Objectif d'apprentissage, Audience, Format, Caractéristiques, Piège



# Quelques retours d'étudiant·es sur les calepins Jupyter (1/2)

- “j’ai découvert les Jupyter Notebook, et je trouve ça très **pratique**”
- “J’ai découvert les Jupiter notebook et **j’aime vraiment cela**”
- “Les calepins sont très bien faits et permettent bien d’**apprendre seul**”
- “Les vidéos de cours et les calepins d’exercice restent **accessibles à tout moment**, ce qui permet de revenir sur un outil/une fonction peu utilisée ou que l’on aurait oublié.”
- “(...) L’utilisation de Jupyterhub est également un avantage puisque nous pouvons à la fois programmer en parallèle et directement appliquer le cours pour voir ce que nous devons trouver. Cela nous permet de savoir où on va et à quoi ça sert. Cette **synergie pédagogique** est plutôt puissante.”

# Quelques retours d'étudiant·es sur les calepins Jupyter (2/2)

- “Le calepin d'utilisation du module matplotlib a été particulièrement utile car nous sommes souvent amenés à représenter des courbes et de mettre en forme des graphiques afin qu'ils soient facilement compréhensibles. De plus, tous les calepins d'exercices nous permettent de **mettre en pratique les fonctions et méthodes vues dans les cours**, ce qui, d'après moi, est essentiel.”
- “Les notebooks sont pratiques pour l'enseignement car on peut exécuter directement les cellules, c'est plus **sympa** pour comprendre. Mais personnellement j'utilisais spyder pour programmer puis jupyter notebook pour **rédigier**”

# Publier les calepins Jupyter sur internet : GitHub Pages

- 1 Créer un Livre Jupyter avec Jupyter Book
- 2 Le mettre sur GitHub et configurer GitHub Actions
- 3 Pour modifier le site internet, il suffit de synchroniser vos calepins avec GitHub, tout est automatique



The screenshot shows the homepage of a Jupyter Book titled "Modélisation Numérique en Physique". The page features a header with navigation icons, a search bar, and a table of contents. The main content area includes a welcome message, a quote by George E. P. Box, and a paragraph about mental models. The left sidebar contains a navigation menu with categories like "Informations générales", "Python base", "Python intermédiaire", "Modélisation numérique", "Exercices", "Mini-projets", and "Description du projet".

**MNP**

Search this book...

Informations générales  
Python base  
Python intermédiaire  
Modélisation numérique  
Exercices  
Mini-projets  
Description du projet

## Modélisation Numérique en Physique

Made with [Jupyter](#) | [Deploy](#) | [License](#) | [Say Thanks](#)

Bienvenue dans le cours de Modélisation Numérique en Physique, un cours de la deuxième année de la licence de physique de Sorbonne Université.

### Modèles et données

"Essentially, all models are wrong, but some are useful."  
*George E. P. Box, statistician, 1919–2013.*

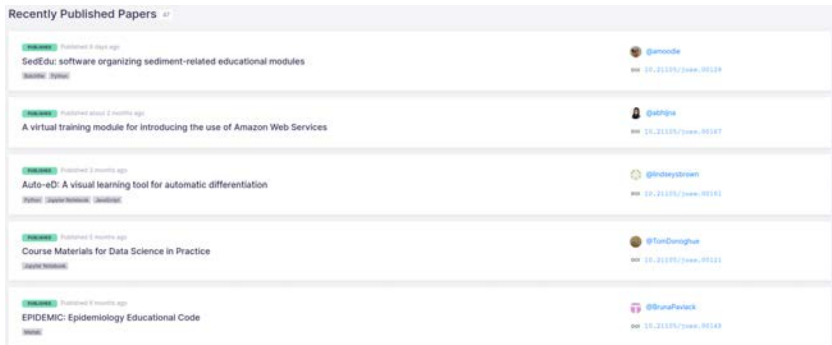
Les modèles sont des *constructions mentales* qui forment la base des raisonnements du physicien. Ils sont construits à partir des théories physiques pour décrire un phénomène ou une situation particulière.

Contents  
Modèles et données  
Résumé du cours  
Amphi de rentrée






<https://github.com/phys-mod/phys-mod.github.io>

# Publier les calepins Jupyter dans un journal : JOSE

- The Journal of Open Source Education
- Journal *peer-review* de publication de ressources *open source* pour l'enseignement
- Peut être un cours ou bien un programme pour l'enseignement, tant qu'il y a des fichiers sources (Jupyter notebooks ou documents LaTeX, Markdown, ReStructuredText, AsciiDoc, R Markdown ...)



Recently Published Papers

<p><b>SeoEdu</b> Published 3 days ago</p> <p>SeoEdu: software organizing sediment-related educational modules</p> <p><a href="#">Source</a> <a href="#">Details</a></p>	<p> @amooie</p> <p><a href="#">DOI: 10.21105/jose.00129</a></p>
<p><b>SeoEdu</b> Published about 2 months ago</p> <p>A virtual training module for introducing the use of Amazon Web Services</p>	<p> @stfipa</p> <p><a href="#">DOI: 10.21105/jose.00187</a></p>
<p><b>SeoEdu</b> Published 2 months ago</p> <p>Auto-eD: A visual learning tool for automatic differentiation</p> <p><a href="#">Python</a> <a href="#">Supplementary</a> <a href="#">Dataset</a></p>	<p> @lindseybrown</p> <p><a href="#">DOI: 10.21105/jose.00161</a></p>
<p><b>SeoEdu</b> Published 2 months ago</p> <p>Course Materials for Data Science in Practice</p> <p><a href="#">Jupyter Notebook</a></p>	<p> @TomDonoghue</p> <p><a href="#">DOI: 10.21105/jose.00121</a></p>
<p><b>SeoEdu</b> Published 2 months ago</p> <p>EPIDEMIC: Epidemiology Educational Code</p> <p><a href="#">Source</a></p>	<p> @BrunaFevick</p> <p><a href="#">DOI: 10.21105/jose.00143</a></p>

# Publier les calepins à Sorbonne Université : DocSU

## Portail gestion électronique des documents Sorbonne Université

The screenshot displays the DocSU portal interface. At the top, there is a navigation bar with links for 'Accueil', 'Fichiers partagés', 'Sites', 'Tâches', 'Personnes', and 'Événements'. The user 'Pacôme Delva' is logged in, and a search bar is available. The main header features the Sorbonne Université logo and the text 'Capsule : Appui à l'enseignement'. Below this, there are four content sections: 'Calendrier du site' listing events like 'Café CAPSULE jupyter', 'Mes discussions' with a 'Nouveaux sujet' button, 'Récemment modifiés par moi', and 'Pas de page configurée'.

Espace documentaire pour partager/écrire des ressources liées à la pédagogie et à l'enseignement au quotidien (lien vers le site).

## Tutoriel JupyterHub

Atelier de 13h45 à 14h30

salle 117 de l'Atrium

- Comment ajouter une activité JupyterHub à Moodle
- Comment ajouter des calepins à JupyterHub
- Utilisation de Jupyter avec Python
- Utilisation avancée : tutoriel sur nbgrader (en construction)