

# Tutoriel d'utilisation d'OCaml avec JupyterHub

Le but de ce tutoriel est de présenter l'utilisation de la plateforme JupyterHub pour la programmation OCaml.

## 1 Avantages et inconvénients

### avantages

- accessible en ligne dans le navigateur
- ne nécessite aucune installation
- authentification avec les identifiants @université-paris-saclay.fr

### inconvénients

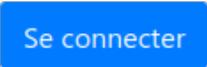
- requiert une connexion Internet
- le service peut être surchargé, en panne ou en maintenance (rare mais c'est déjà arrivé)
- support minimal (mais utilisable) pour OCaml : un éditeur de texte avec coloration (mais pas d'affichage des types ni d'autocomplétion), un terminal
- espace de stockage séparé de celui des salles machines du 336, penser à copier les fichiers ou les télécharger.

## 2 Connexion et démarrage de l'instance

1. Se connecter sur

`https://jupyterhub.ijclab.in2p3.fr/`

2. Cliquer sur le bouton



3. Choisir l'établissement Université Paris-Saclay (et éventuellement cocher les cases pour se souvenir du choix).
4. S'authentifier avec les identifiants Paris-Saclay si cela vous est demandé. Si vous vous êtes déjà authentifié dans le navigateur Web (accès récent à eCampus, au mail, ...) il se peut que la connexion se fasse automatiquement.
5. Cliquer sur le bouton

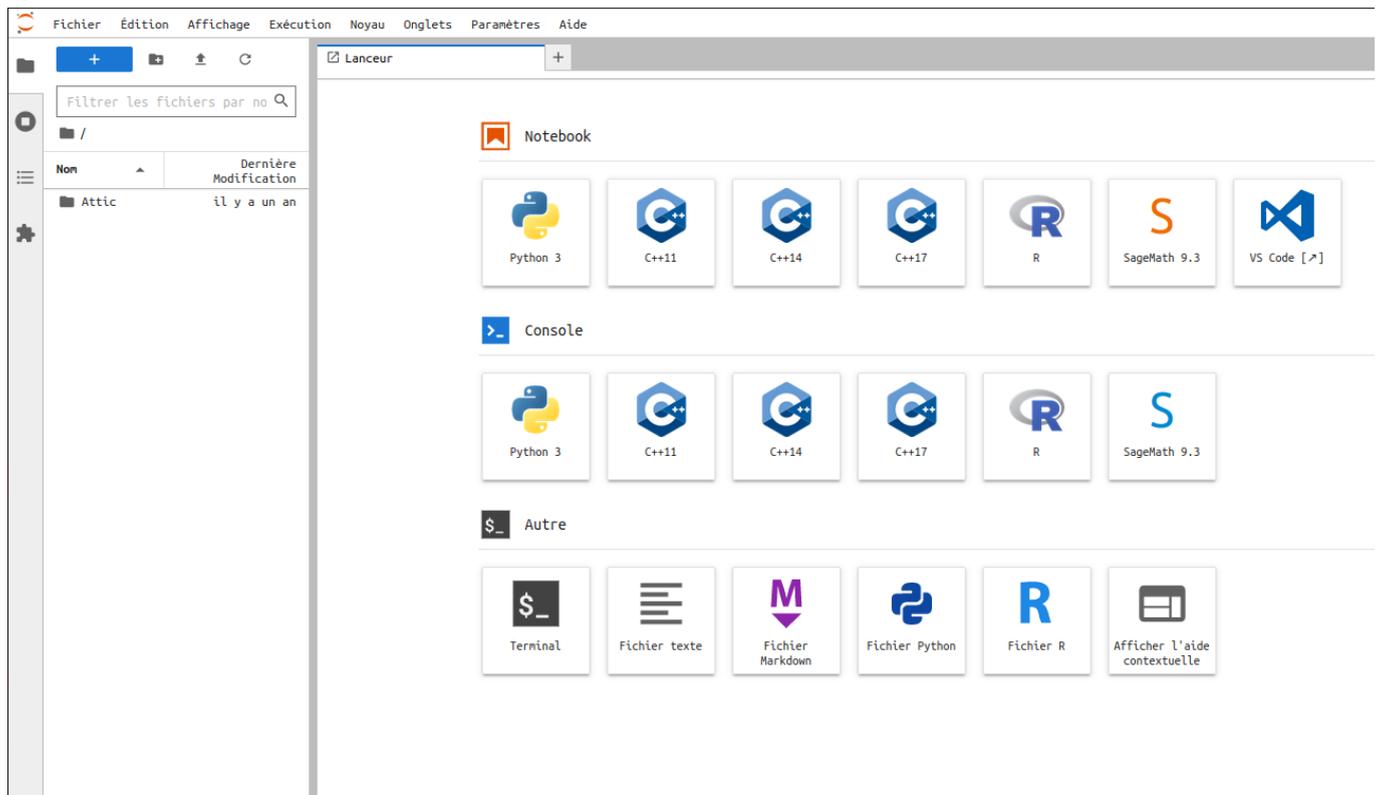


et attendre le démarrage du serveur

6. Attendre le démarrage du serveur et l'affichage de l'espace de travail

## 3 Utilisation d'OCaml sous JupyterLab

Une fois l'espace JupyterLab démarré, on se retrouve avec l'interface suivante :



Explorateur de fichiers



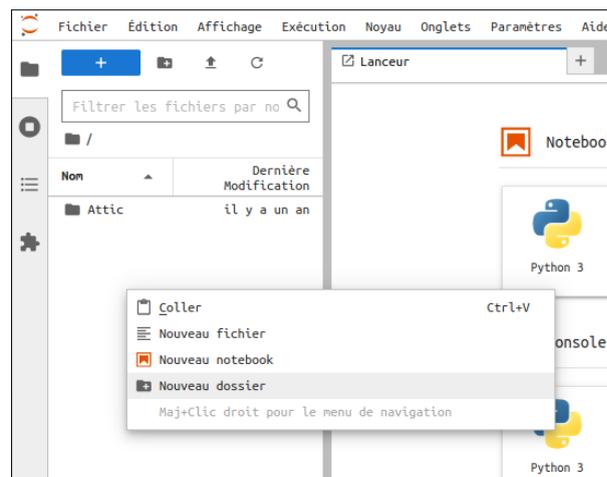
Lanceur de programmes

On se limitera à l'utilisation de l'explorateur de fichier, de l'éditeur de texte et du terminal. Les autres outils et options de configurations sont à utiliser avec précaution.

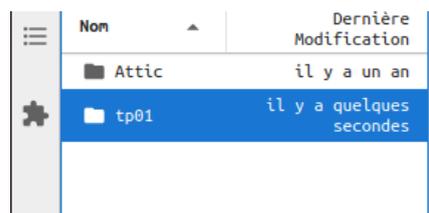
### 3.1 Création d'un nouveau répertoire de travail

Cette opération est à faire par exemple au début d'un TP.

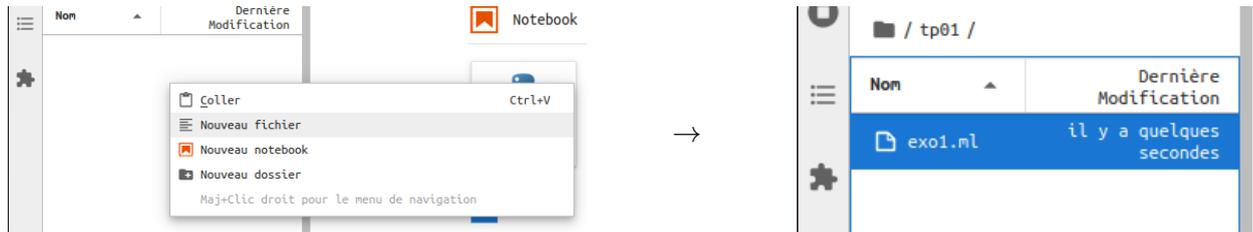
1. Cliquer bouton droit dans l'explorateur de fichier et choisir **Nouveau dossier** :



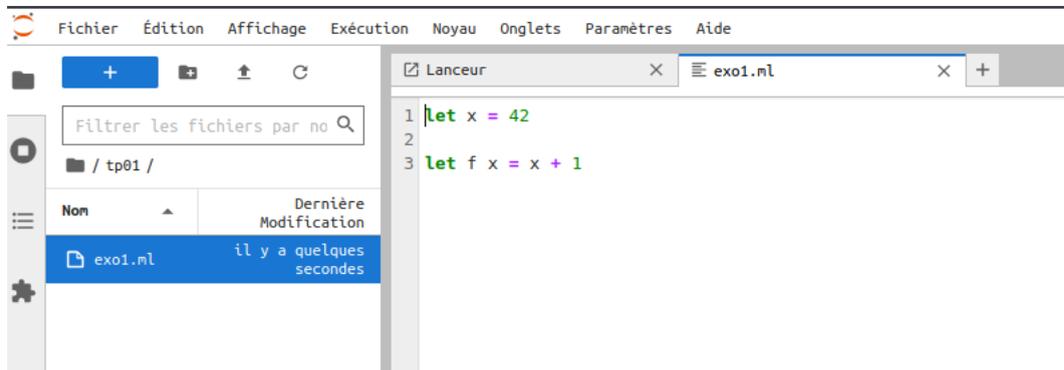
2. Choisir un nom pour le dossier (de préférence sans espace ni accents afin de faciliter plus tard la saisie dan le terminal) :



3. Cliquer sur le dossier, puis cliquer bouton droit et choisir Nouveau fichier. Choisir un nom de fichier et modifier l'extension en .ml :



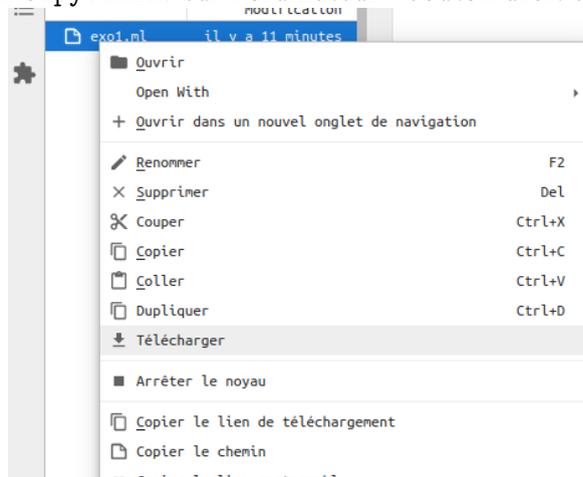
4. Cliquer sur le fichier nouvellement créé, un éditeur de texte s'ouvre dans un nouvel onglet, vous pouvez maintenant éditer le fichier :



### 3.2 Dépôt et récupération d'un fichier existant

Il est souvent utile de déposer ou récupérer un fichier déjà écrit depuis ou vers son ordinateur (code à trous, fichier déjà travaillé en TP, ...). On utilise pour cela l'explorateur de fichiers.

- téléchargement d'un fichier JupyterLab sur l'ordinateur : bouton droit sur le fichier puis, télécharger.



- dépôt d'un fichier existant : se placer dans le répertoire voulu puis cliquer sur l'icône 

### 3.3 Navigation dans l'arborescence

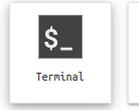
Il est possible de remonter dans l'arborescence de répertoire en cliquant sur les éléments voulus dans le chemin affiché :



### 3.4 Utilisation du terminal

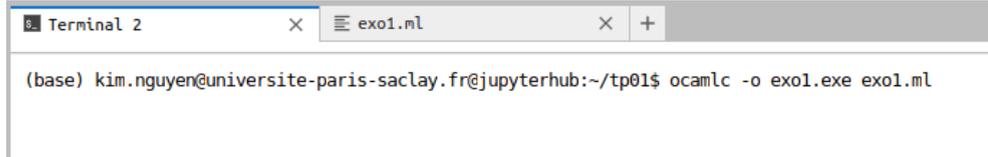
Il est possible de démarrer un terminal depuis le lanceur :

1. (Si le lanceur n'est pas disponible vous pouvez en ouvrir un avec **Ctrl-Shift-L** ou **Fichier** → **Nouveau lanceur...**



2. Cliquer sur l'icône de terminal et attendre le lancement dans un nouvel onglet

3. Entrer les commandes comme en TP

A screenshot of a terminal window. The title bar shows "Terminal 2" and "exo1.ml". The terminal content shows the prompt "(base) kim.nguyen@universite-paris-saclay.fr@jupyterhub:~/tp01\$" followed by the command "ocamlc -o exo1.exe exo1.ml".

```
(base) kim.nguyen@universite-paris-saclay.fr@jupyterhub:~/tp01$ ocamlc -o exo1.exe exo1.ml
```

4. Vous pouvez maintenant aller et venir entre les deux onglets (édition de code et compilation dans le terminal). Ne pas oublier de sauvegarder le fichier avant de compiler (**Ctrl-S** ou **Fichier** → **Sauvegarder**).