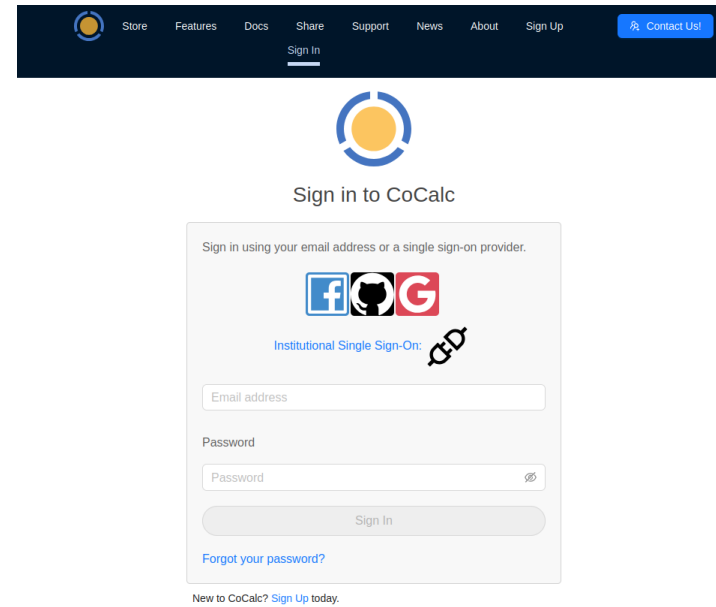
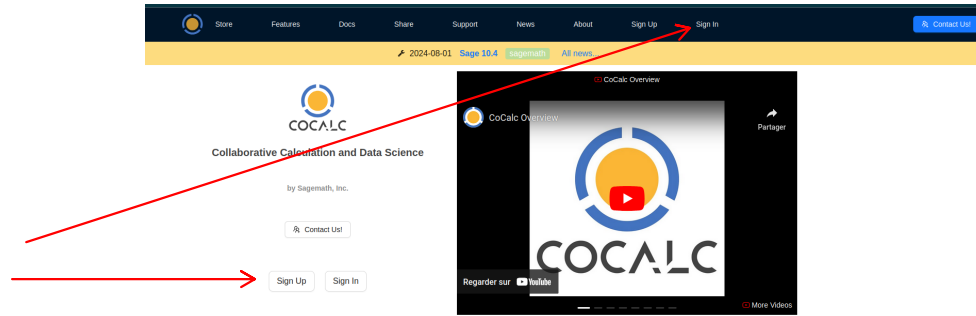


## Comment de connecter à Cocalc

Adresse du site: <https://cocalc.com>

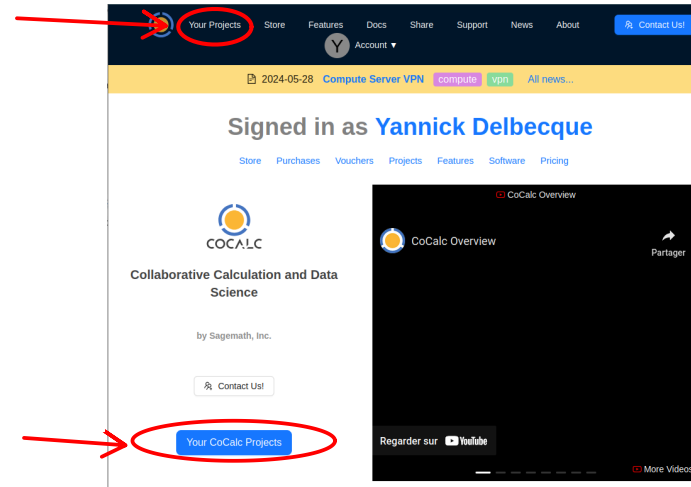
Pour se connecter, faire "sign in"



## Lancer le projet

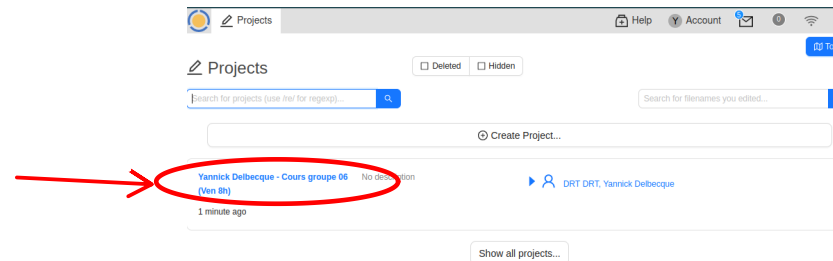
Un "projet" est un ordinateur virtuel avec un espace pour des fichiers et où sont installés plusieurs logiciels scientifiques.

Après s'être connecté, ouvrir la liste de vos projets avec un des deux liens encadrés



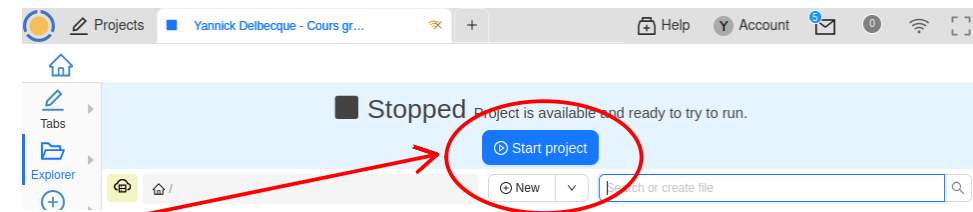
Cela affiche la liste de vos projets.

Ouvrir le projet associé au cours qui a été créé pour vous.



## Relancer un projet

Pour préserver les ressources du système, cocalc arrête automatiquement les projets inactifs. Si vous voyez apparaître "Start projet", simplement relancer le projet pour pouvoir continuer à travailler.



Cliquez ici pour relancer le projet

## Liste des répertoires

Une fois le projet lancé, vous verrez un navigateur de fichiers.

La liste des répertoire de votre projet s'allongera pendant la session. Des répertoires seront automatiquement ajoutés à chaque cours.

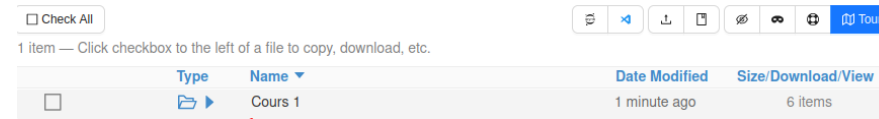
## Fichiers dans un répertoire

En ouvrant un répertoire, on voit la liste des fichiers distribués pour le cours.

Dans l'exemple à droite, il y a trois types de fichiers:

- (a) une feuille de calcul Jupyter (.ipynb);
- (b) un fichier texte contenant le code Python d'un programme (.py);
- (c) un fichier permettant d'ouvrir un terminal pour exécuter des commandes (.term).

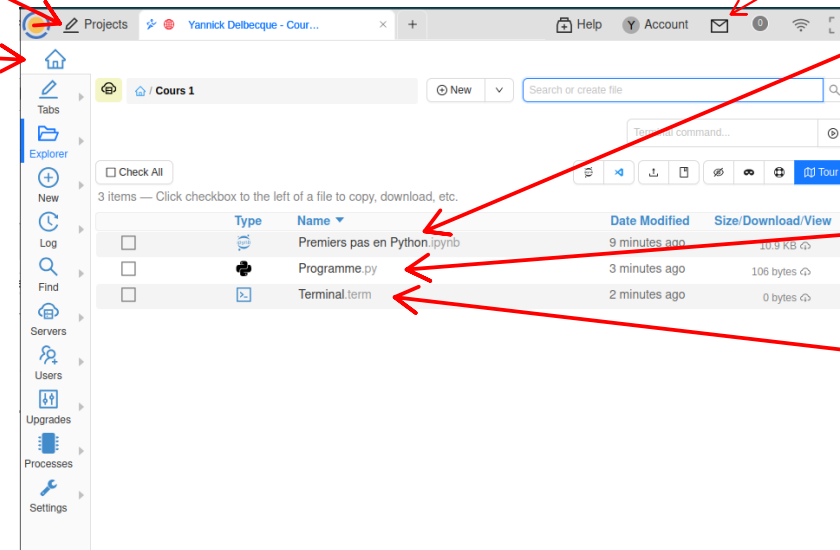
On ouvre un fichier en cliquant.



Répertoire du cours 1

Liste des projets et projets ouverts

Retour au répertoire principal du projet



Messagerie Cocalc

Feuille de calcul Jupyter

Code python

Terminal

## Créations de répertoires et de fichiers

On peut créer des nouveaux répertoires et des fichiers avec le menu déroulant à droite du bouton "New".  
Le répertoire ou fichier est créé dans le répertoire courant.

Répertoire courant

Bouton "New"



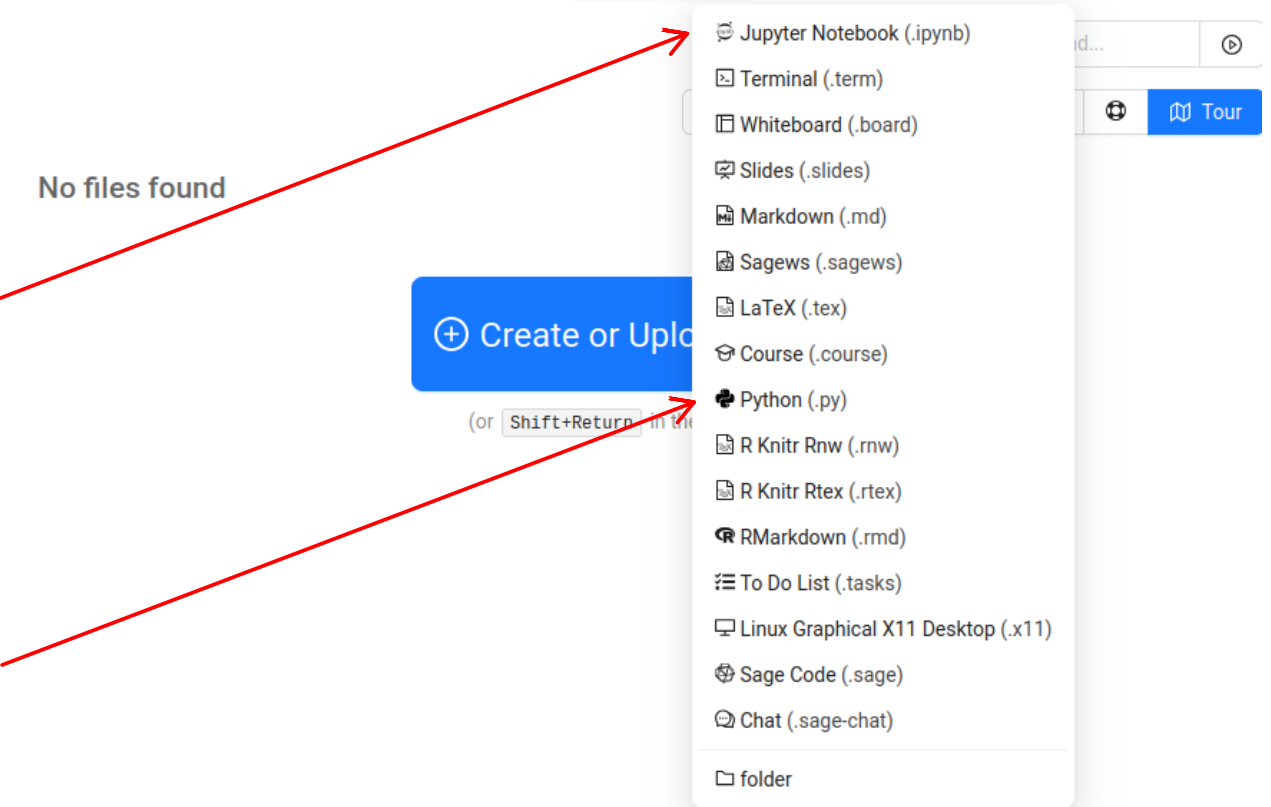
Vous pourrez ensuite choisir un nom pour le fichier ou répertoire que vous voulez créer.

No files found

## Type de fichiers

Jupyter notebook (.ipynb): feuille de calcul comme celles distribuées pour les notes de cours.  
Si vous créer une feuille de calcul, choisissez ensuite le noyau (kernel) qui exécutera les commandes entrée: Python ou Sagemath.

Code source python (.py): pour faire un programme python.



## Ouverture d'une feuille de calcul jupyter

Les feuilles de calculs Jupyter permettent de mettre des boîtes où entrer des commandes Python dans un texte.

Les notes de cours seront distribuées sous forme de feuille de calcul à compléter.

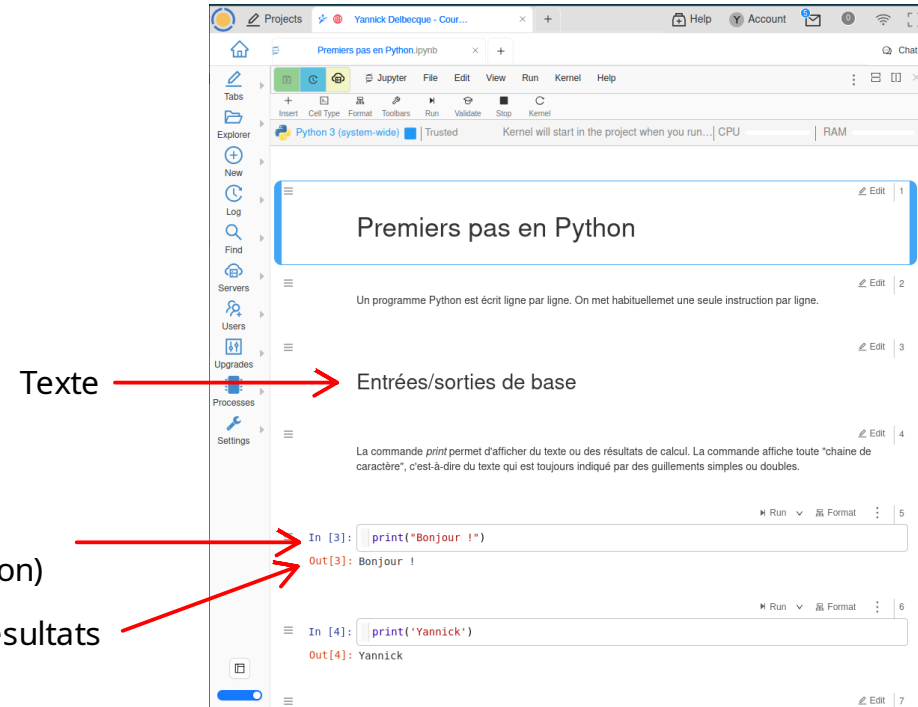
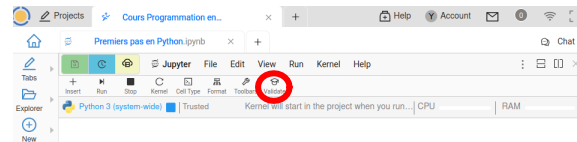
Note: les commandes entrées sont numérotées, par exemple: "In [3]"

La sortie correspondante est "Out [3]".

La numérotation reflète l'ordre d'exécution des commandes et non leur ordre d'apparition dans le fichier.

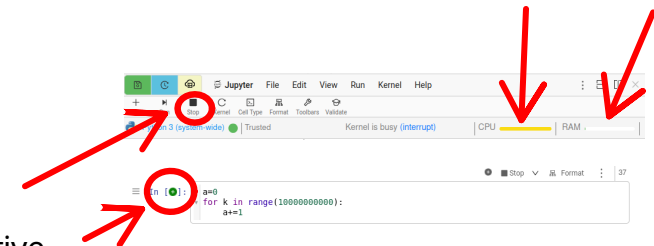
## Contrôle d'exécution des cellules

Le bouton "Validate" reprend l'interpréteur Python à zéro et exécute le code de toutes les cellules dans l'ordre.



## Utilisation CPU et RAM

Si l'exécution d'une cellule prend trop de temps, utilisez le bouton "stop" exécution active



## Éditeur de code

Les programmes et les textes sont ouverts dans un éditeur de texte (brut).

Le code python (fichiers se terminant par ".py") sera automatiquement reconnu et affiché avec rehaussement syntaxique en couleur.

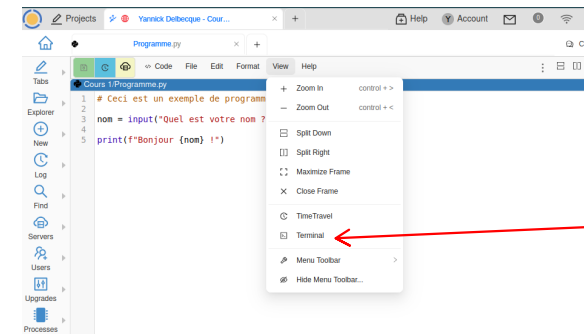
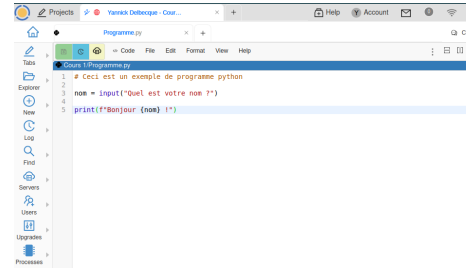
On peut ouvrir un terminal à côté de l'éditeur pour exécuter le code python qui est édité.  
-> Menu "View", item "Terminal".

Une fenêtre terminal sera ouverte à côté du code.

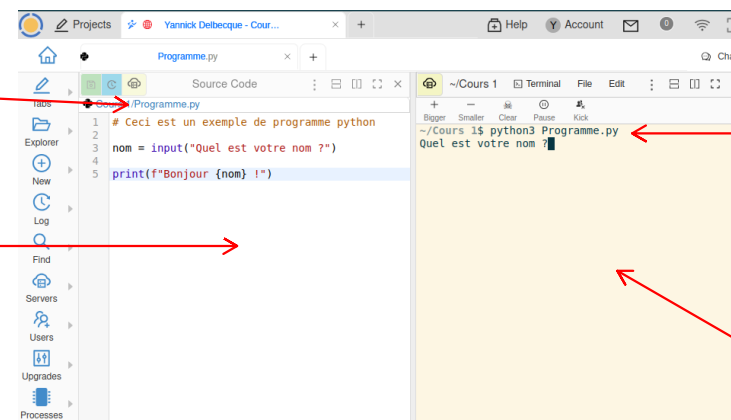
Pour exécuter votre programme, entrez "python3 " suivi du nom de votre programme.

Truc: dans un terminal, les flèches haut/bas font défiler l'historique des commandes entrées

Truc 2: la touche "tabulation" complète plusieurs entrées.



Ouvrir un terminal



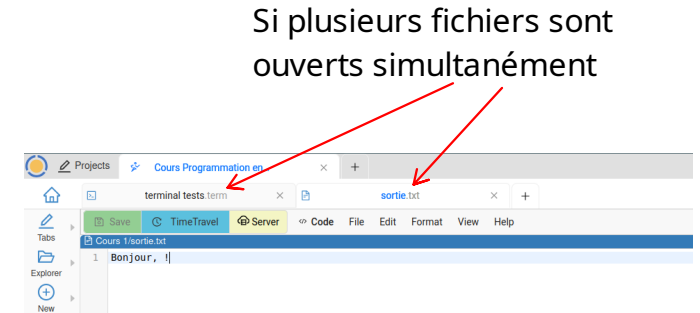
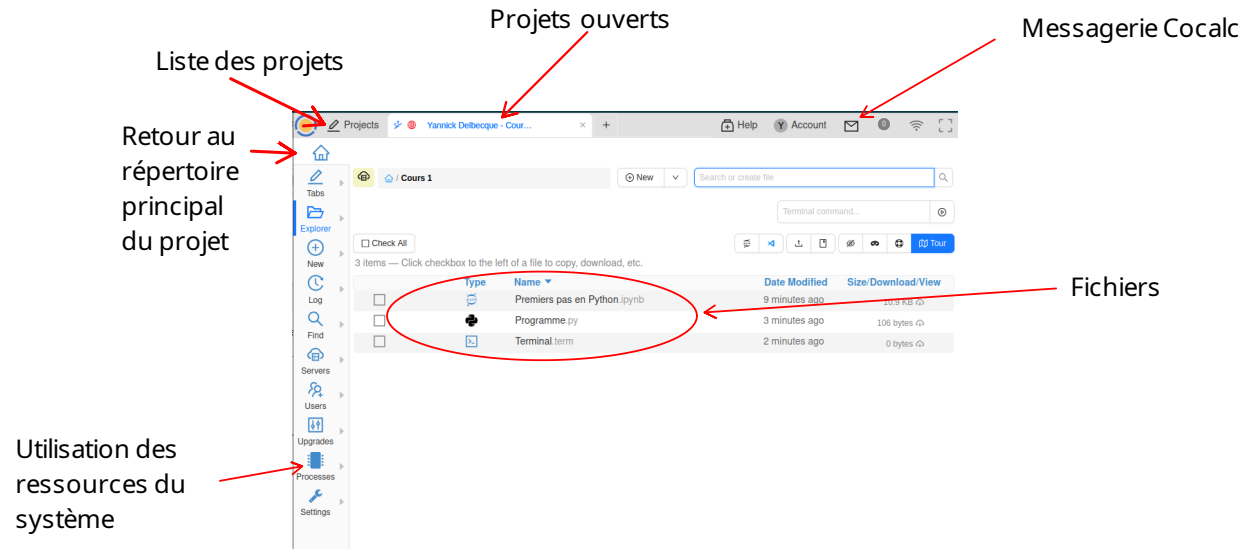
Nom du programme

Éditeur de code

Commande pour exécuter le programme "Programme.py"

Terminal

# Éléments importants de l'interface Cocalc



## Voir l'utilisation des ressources système

Cliquer ici pour voir l'utilisation des ressources du système

Utilisation du processeur

Process	Project	PID	CPU%	CPU Time	Memory
node --trace-warnings --trace-uncaught --optimize-for-s...	Project	8	0.5%	12.1s	94 MiB
sshd /usr/sbin/sshd -D -p 2222 -h /tmp/cocalc/ssh_hos...	SSH	212	0.0%	0.01s	9 MiB
bash	Programme	888	0.0%	0.04s	7 MiB
- bash	Terminal	960	0.0%	0.03s	7 MiB
python3.10		1023	0.0%	0.03s	11 MiB

Utilisation de la mémoire