
ENVIRONNEMENT DE TRAVAIL • SANS INSTALLATION

Travailler dans Google Colab

Brancher Google Colaboratory sur votre Google Drive, ouvrir les calepins du cours dans le navigateur et remettre votre travail, sans rien installer sur l'ordinateur.

Pour travailler sur les ordinateurs du cégep, sur un ordinateur emprunté, ou en attendant que l'installation fonctionne. Comptez cinq minutes.

Procédure en PDF

Pourquoi Google Colab ?

Google Colaboratory (« Colab ») est un Jupyter qui tourne dans le navigateur : Google vous prête un ordinateur dans le nuage, et vos calepins `.ipynb` s'ouvrent, s'exécutent et se sauvegardent dans votre **Google Drive**. Il n'y a rien à installer, et les librairies du cours (`numpy` , `matplotlib`) sont déjà là. Il suffit d'un compte Google : celui du cégep (le même identifiant que pour Microsoft) ou un compte personnel.

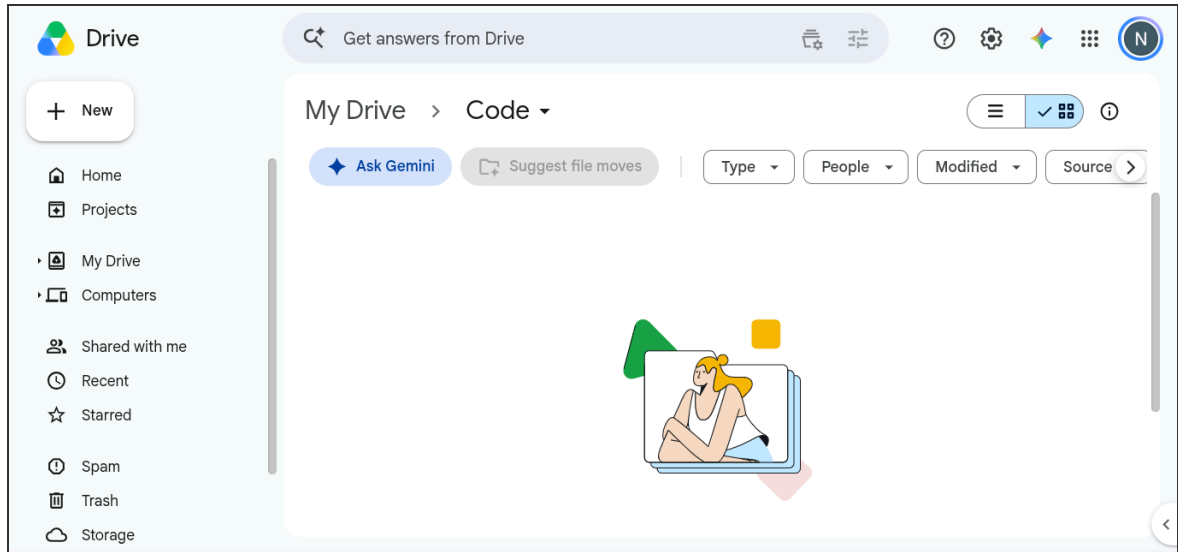
L'installation de Visual Studio Code sur votre ordinateur reste la voie principale du cours : c'est là que le mode strict vérifie vos types, ce que Colab ne fait pas. Colab est la voie de secours et la voie mobile : n'importe quel ordinateur avec un navigateur devient votre poste de travail.

Brancher Colaboratory sur votre Google Drive

À faire une seule fois par compte Google. Ensuite, un double-clic sur un fichier `.ipynb` dans le Drive l'ouvre directement dans Colab. Les captures montrent Google Drive en anglais ; le libellé français est donné à chaque étape.

- 1 Ouvrir drive.google.com, se connecter avec son compte Google, puis créer un dossier pour le cours avec **Nouveau** (New) > **Nouveau dossier** (New folder) et l'ouvrir.

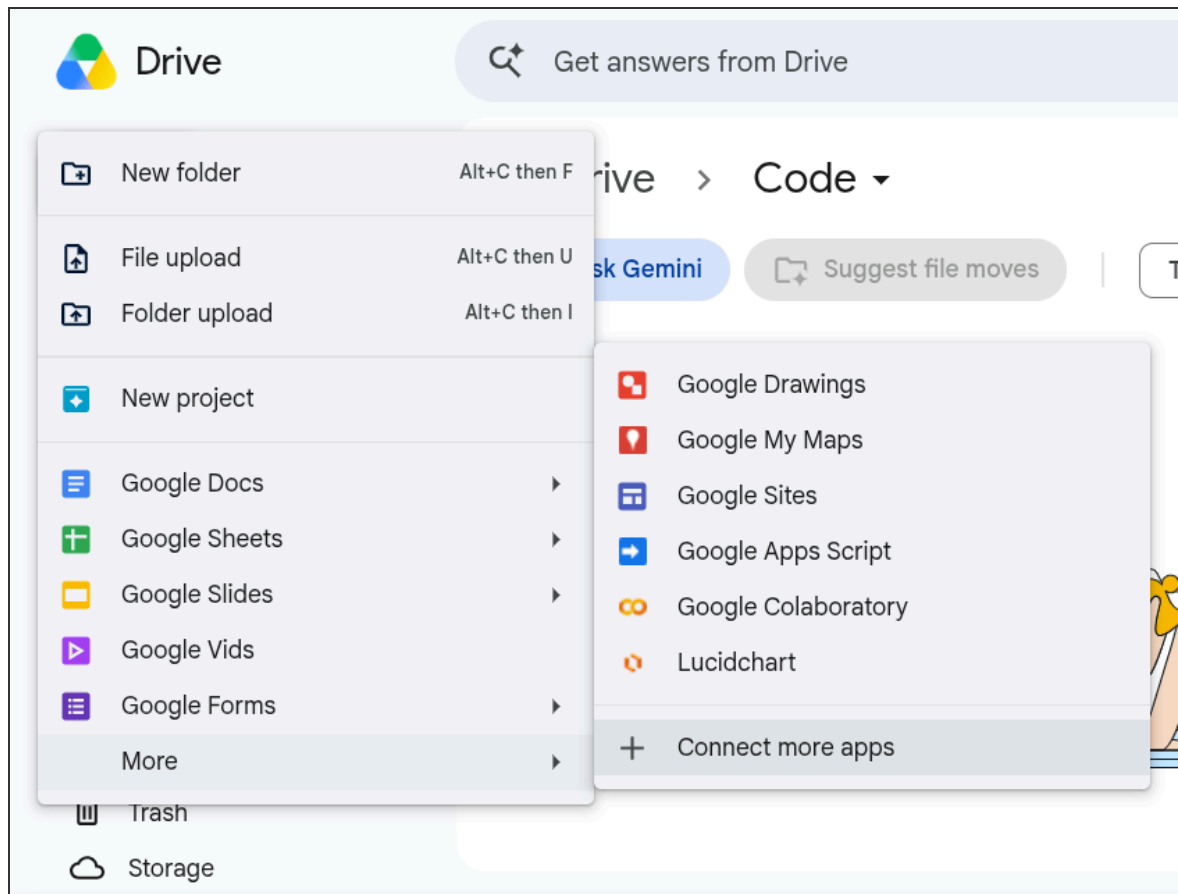
Le nom du dossier est libre : « Programmation », « 420-SN1 »... Tout le travail du cours ira dans ce dossier, calepins et fichiers de données.



Le dossier du cours, vide pour l'instant, dans Mon Drive.

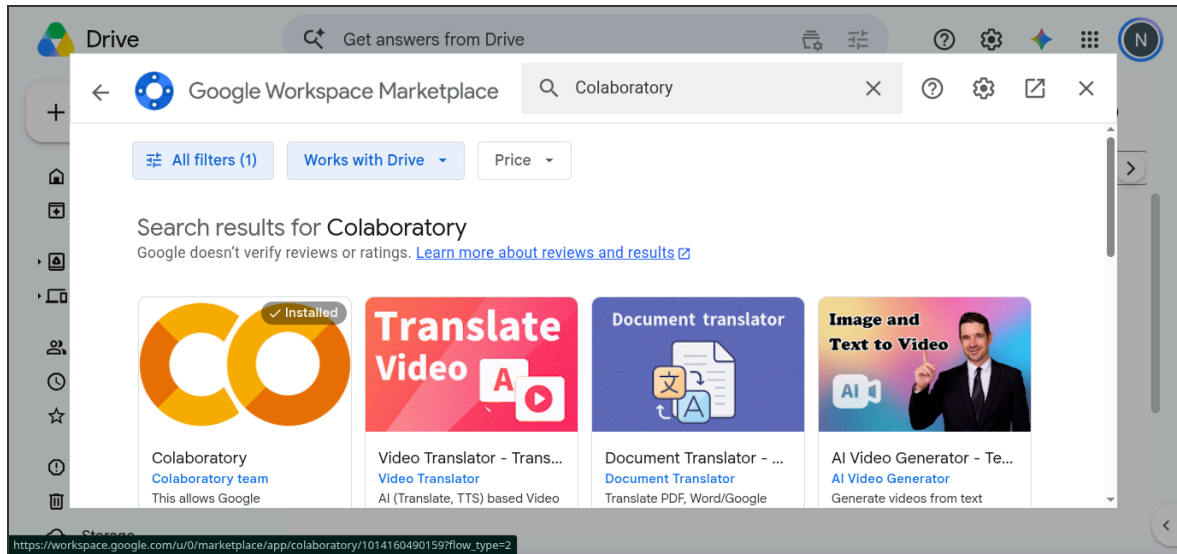
2 Cliquer sur **Nouveau** (New), puis **Plus** (More), puis **Connecter d'autres applications** (Connect more apps).

Si **Google Colaboratory** figure déjà dans la liste du sous-menu Plus, l'application est déjà branchée sur ce compte : passez directement à l'étape 8.



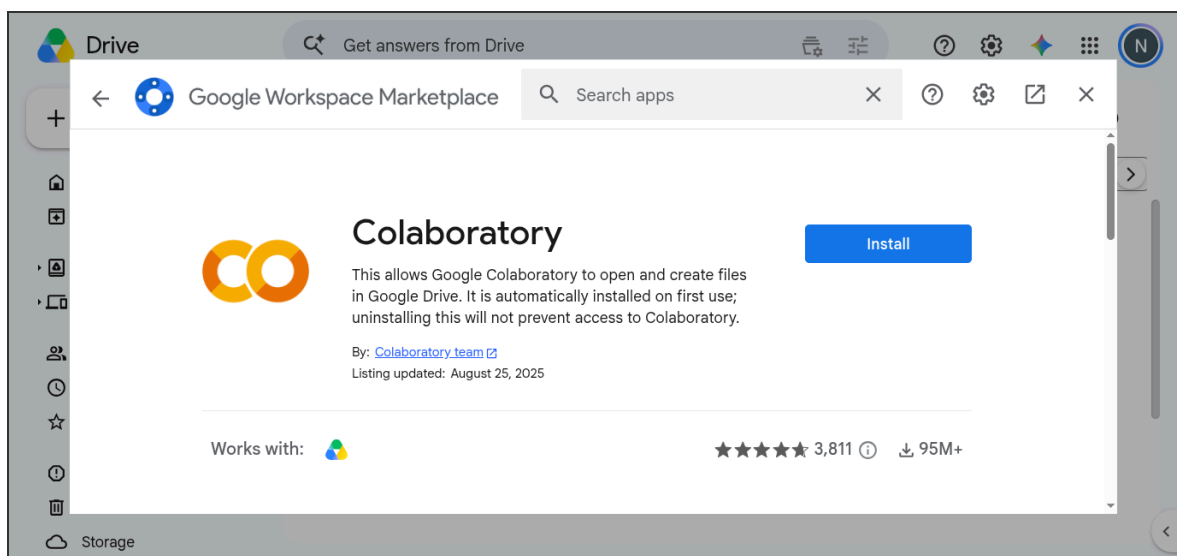
3 Dans la fenêtre **Google Workspace Marketplace**, taper **Colaboratory** dans la barre de recherche et appuyer sur Entrée.

L'application se reconnaît à son logo : deux anneaux orange formant « CO ». Elle est publiée par « Colaboratory team ».



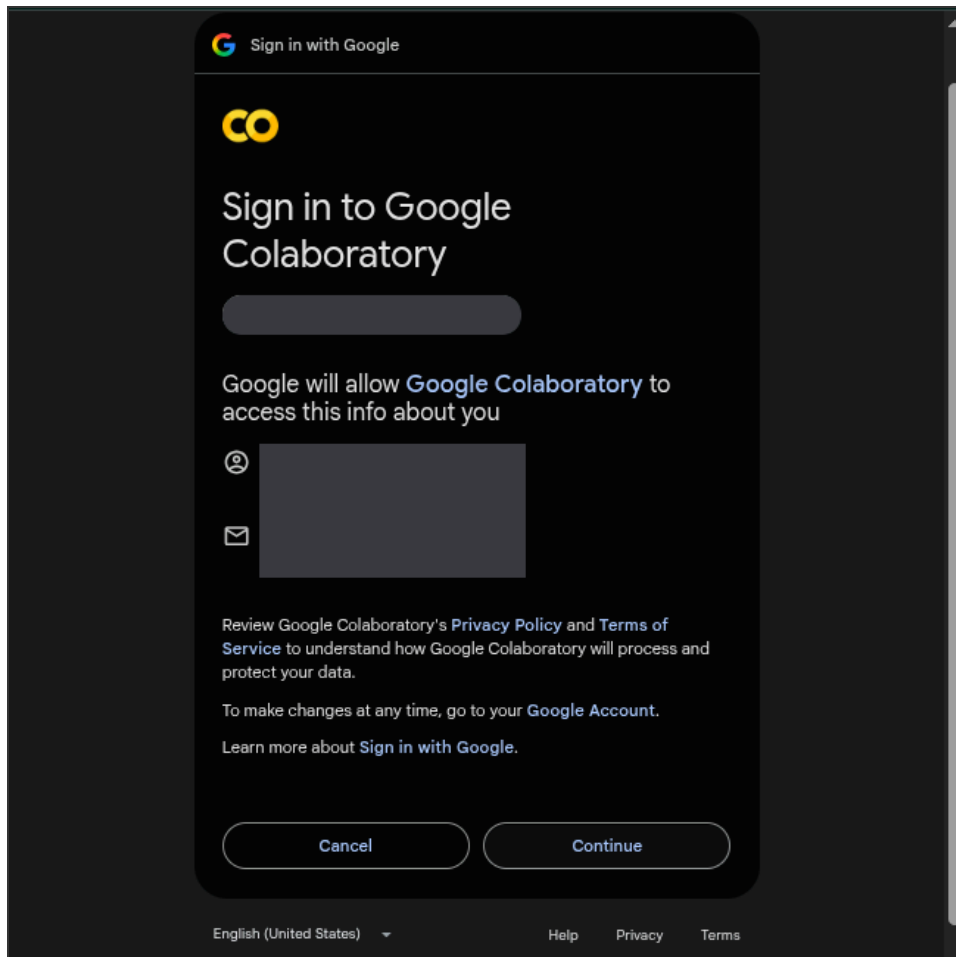
4 Cliquer sur la tuile **Colaboratory**, puis sur le bouton bleu **Installer** (Install).

La fiche précise que l'installation permet à Colaboratory d'ouvrir et de créer des fichiers dans Google Drive. C'est exactement ce qu'on veut.



- 5 Cliquer sur **Continuer** (Continue) pour autoriser l'installation, puis, dans la petite fenêtre qui s'ouvre, choisir son compte Google et cliquer de nouveau sur **Continuer**.

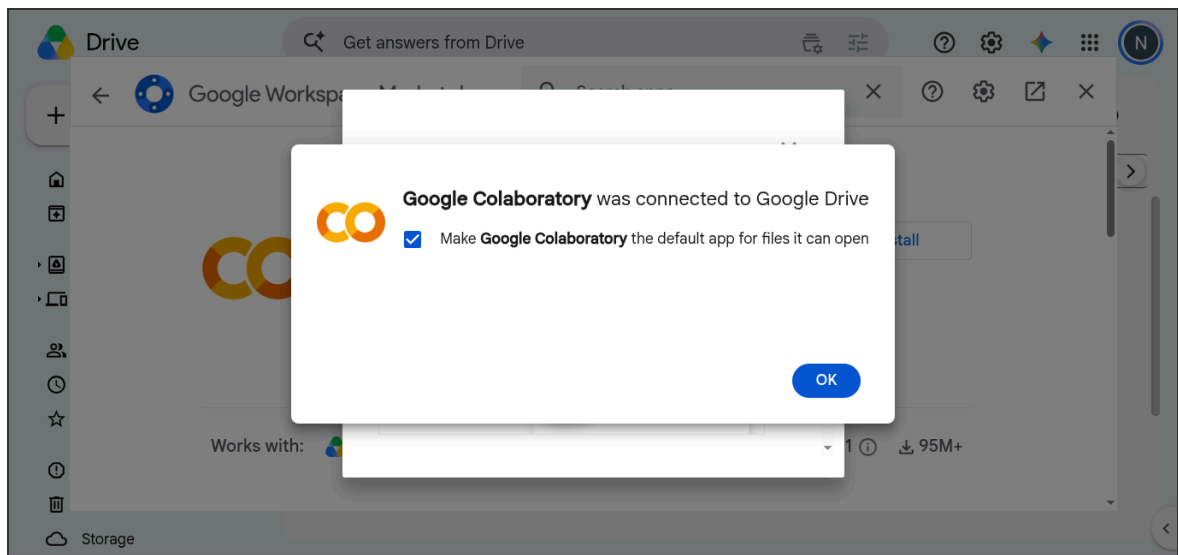
Google indique que Colaboratory verra votre nom et votre adresse courriel : c'est normal, c'est ce qui lui permet d'enregistrer vos calepins dans votre Drive. Si la petite fenêtre n'apparaît pas, vérifiez que le navigateur ne bloque pas les fenêtres surgissantes.



Le compte affiché ici a été masqué : vous verrez le vôtre.

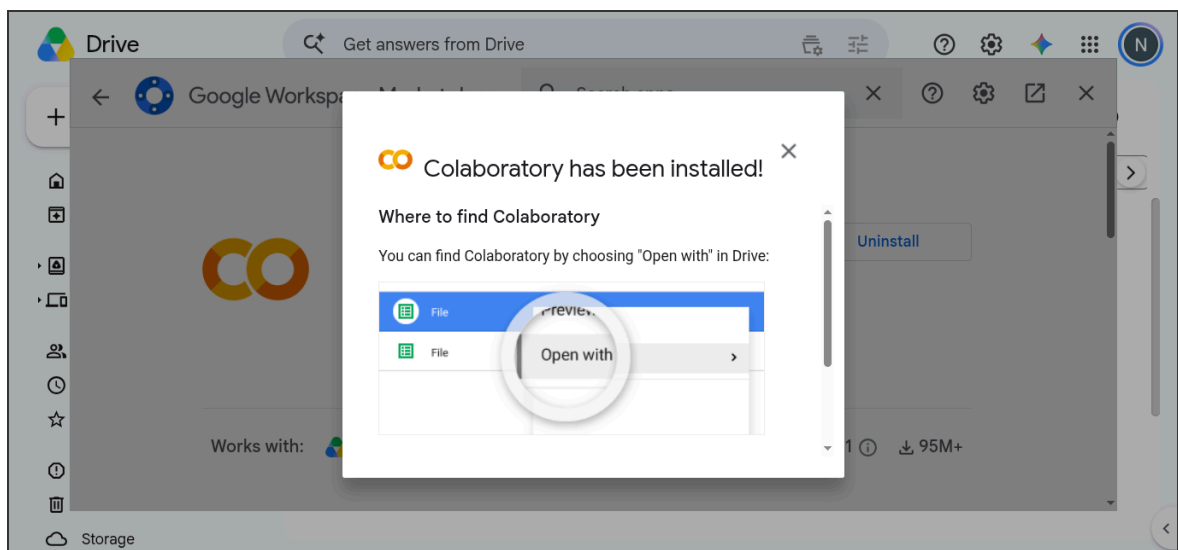
- 6 Le message « Google Colaboratory a été connecté à Google Drive » apparaît. Laisser cochée la case **Définir Google Colaboratory comme application par défaut** (Make Google Colaboratory the default app for files it can open), puis cliquer sur **OK**.

Grâce à cette case, un double-clic sur un fichier `.ipynb` dans votre Drive l'ouvrira directement dans Colab, sans passer par un menu.



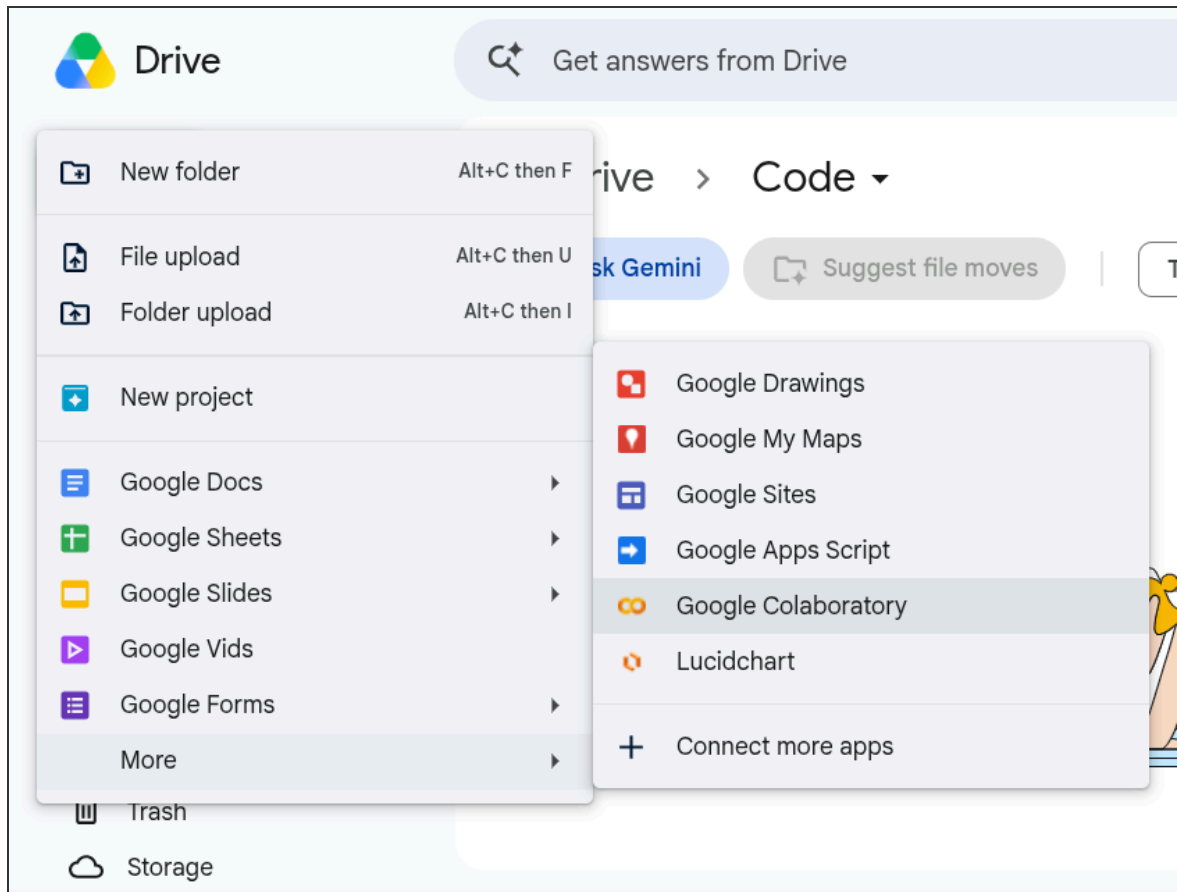
- 7 Fermer la fenêtre « Colaboratory has been installed » avec la croix, puis fermer le Marketplace de la même façon.

Cette fenêtre rappelle l'autre façon d'ouvrir un calepin : clic droit sur le fichier dans le Drive, **Ouvrir avec** (Open with) > **Google Colaboratory**. Utile si un jour le double-clic ouvre autre chose.



- 8 De retour dans le dossier du cours, cliquer sur **Nouveau** (New) > **Plus** (More) > **Google Colaboratory** : un calepin vide s'ouvre dans un nouvel onglet, déjà rangé dans le dossier.

Le calepin s'appelle `Untitled0.ipynb`. Cliquez sur ce nom, en haut à gauche de Colab, pour le renommer (par exemple `prise-en-main.ipynb`). Colaboratory est branché : c'est fait.



Un calepin du cours, du site à la remise

Le chemin complet d'un laboratoire : le télécharger depuis le site, le travailler dans Colab, le retrouver, puis le remettre.

- 9 Sur le [site du cours](#), ouvrir la page du laboratoire et cliquer sur **Télécharger le fichier source (.ipynb)** dans le bandeau du haut.

Le fichier `laboratoire-N.ipynb` arrive dans le dossier Téléchargements de l'ordinateur. Quand le laboratoire vient avec des fichiers de données (`.txt` , `.csv`) ou un `.zip` , téléchargez-les aussi et décompressez le zip.

- 10 Dans Google Drive, ouvrir le dossier du cours et y glisser le fichier `.ipynb` depuis le dossier Téléchargements, ou utiliser **Nouveau > Importer un fichier** (File upload).

Le calepin est maintenant dans votre Drive : c'est là qu'il vivra, et c'est là que Colab l'enregistrera à chaque modification.

- 11 Double-cliquer sur le fichier : il s'ouvre dans Colab. Cliquer dans une cellule de code et l'exécuter avec le bouton de lecture à sa gauche, ou avec `Ctrl` + `Entrée` .

La première exécution prend quelques secondes : Colab se connecte à l'ordinateur prêté (« Connecté », en haut à droite). Si le laboratoire lit un fichier de données, téléversez ce fichier dans le panneau **Fichiers** (l'icône de dossier dans la marge gauche de Colab) au début de chaque séance : ce panneau est vidé à la fin de la session, seul le calepin dans le Drive est conservé.

- 12 Rien à sauvegarder : Colab enregistre le calepin dans le Drive à mesure. `Ctrl` + `S` force l'enregistrement avant de fermer l'onglet.

Pour reprendre plus tard, depuis n'importe quel ordinateur : double-clic sur le fichier dans le Drive, ou colab.research.google.com > onglet **Google Drive**, qui liste vos calepins récents.

13 Pour la remise : d'abord **Exécution** (Runtime) > **Tout exécuter** (Run all), puis **Fichier** (File) > **Télécharger** (Download) > **Télécharger le fichier .ipynb**, et déposer ce fichier sur la plateforme de remise du cours.

« Tout exécuter » rejoue les cellules de haut en bas : les résultats et les graphiques sont enregistrés dans le fichier remis, et la prof les voit sans rien relancer. Un calepin remis sans résultats est un calepin dont le code n'a peut-être jamais tourné.

Prise en main : votre premier calepin (formatif)

Un exercice de cinq minutes pour vérifier que tout fonctionne, repris du matériel du département. Créez un calepin (étape 8), renommez-le `prise-en-main.ipynb`, puis ajoutez les blocs ci-dessous dans l'ordre avec les boutons **+ Code** et **+ Texte** du haut de la page. Exécutez chaque bloc de code au fur et à mesure.

BLOC TEXTE

```
# Introduction à Python avec Jupyter Notebook
```

```
## 1. Introduction
```

Ce calepin donne une introduction aux bases de Python.

```
## 2. Types de données
```

Python prend en charge plusieurs types de données de base : les entiers, les flottants, les chaînes de caractères et les listes.

BLOC CODE

```
nombreEntier = 10
print("Entier :", nombreEntier)
nombreFlottant = 3.14
print("Flottant :", nombreFlottant)
liste = [1, 2, 3, 4, 5]
print("Liste :", liste)
```

BLOC TEXTE

3. Installation de modules

Il est possible d'installer des modules avec une commande comme ``%pip install nom_du_module``.

Pour l'exercice, nous installons les deux modules ci-dessous.

- * ``numpy`` (Numerical Python), une bibliothèque de calcul numérique très puissante.
- * ``matplotlib``, pour tracer des graphiques.

BLOC CODE

```
%%capture --no-display

%pip install numpy
%pip install matplotlib
```

BLOC TEXTE

Nous pouvons maintenant utiliser ``numpy`` pour générer les données d'une fonction sinus, puis afficher le graphique avec ``matplotlib``.

BLOC CODE

```
import numpy as np

abscisses = np.linspace(0, 10, 100)
ordonnees = np.sin(abscisses)
print("Tableau NumPy x :", abscisses)
print("Tableau NumPy y :", ordonnees)
```

BLOC CODE

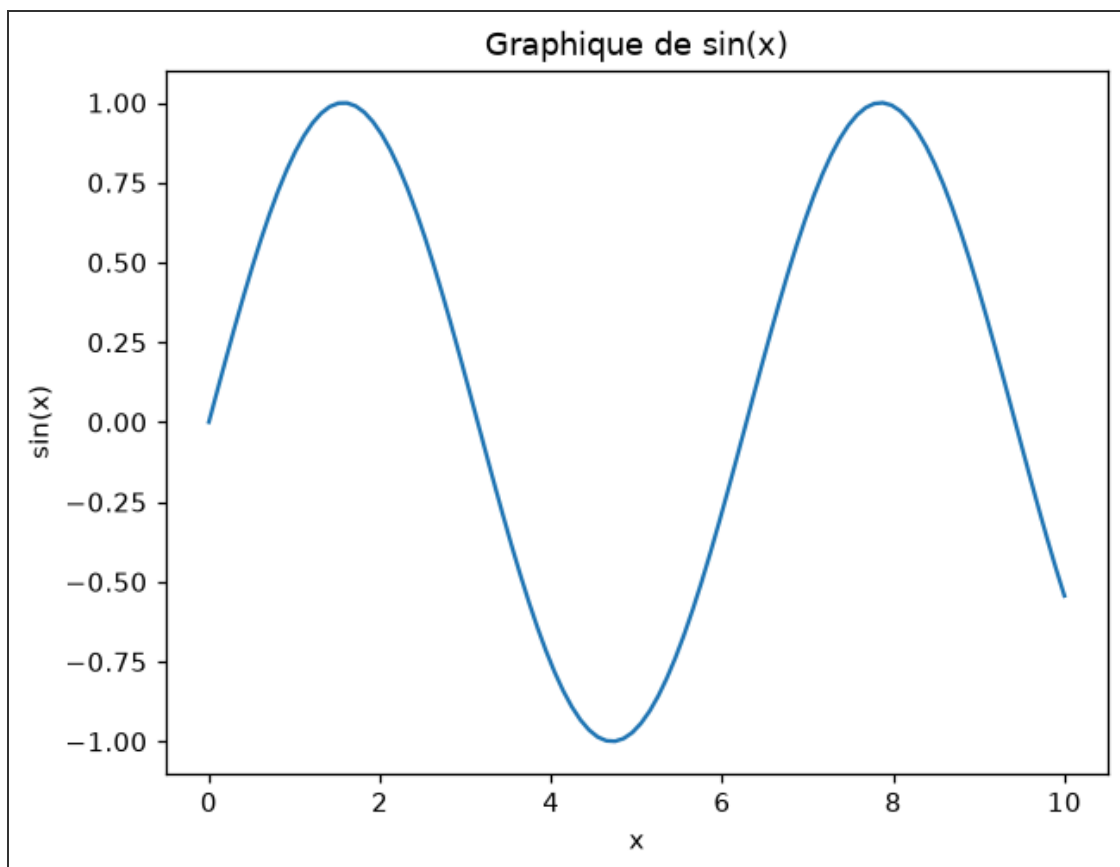
```
import matplotlib.pyplot as plt

plt.plot(abscisses, ordonnees)
plt.title("Graphique de sin(x)")
plt.xlabel("x")
plt.ylabel("sin(x)")
plt.show()
```



Résultat attendu : le dernier bloc affiche la courbe ci-dessous. Si elle apparaît, votre environnement Colab fonctionne.

Dans Colab, `numpy` et `matplotlib` sont déjà installés : le bloc `%pip install` ne fait donc presque rien, mais c'est le geste à connaître pour tout autre module.



Régler Colab pour vraiment apprendre

Par défaut, Colab souffle le code avec l'autocomplétion et l'IA. En apprentissage, ça écrit l'exercice à votre place. Ouvrir **Outils** (Tools) > **Paramètres** (Settings), régler les deux onglets ci-dessous, puis cliquer sur **Enregistrer** (Save). Ces réglages sont gardés dans le navigateur : à faire une fois par ordinateur.

ONGLET « ASSISTANCE IA » (AI ASSISTANCE)

- ☐ **Afficher les complétions IA en ligne** (*Show AI-powered inline completions*)
Sinon Colab propose des bouts de code tout faits pendant que vous tapez.

- ☐ **Consentement aux fonctions d'IA générative** (*Consented to use generative AI features*)
Coupe l'assistant qui rédige le code à votre place.

- ☒ **Masquer les fonctions d'IA générative** (*Hide generative AI features*)
Fait disparaître les boutons d'IA de l'interface.

ONGLET « ÉDITEUR » (EDITOR)

- ☐ **Fermer automatiquement les parenthèses et les guillemets** (*Automatically close brackets and quotes*)
Évite les parenthèses en double qui apparaissent toutes seules.

- ☐ **La touche Entrée accepte les suggestions** (*Enter key accepts suggestions*)
Pour qu'Entrée fasse un vrai retour à la ligne.

- ☒ **Afficher les numéros de ligne** (*Show line numbers*)
Une erreur Python dit « ligne 7 » : vous la trouvez d'un coup d'oeil.

- ☒ **Afficher les guides d'indentation** (*Show indentation guides*)
Des traits verticaux montrent les blocs : un allié contre les erreurs d'indentation.

Quand Colab bloque

Une cellule tourne sans fin, une variable garde une vieille valeur, une erreur revient sans raison ? On repart à neuf, sans rien perdre du code :

- **Exécution** (Runtime) > **Déconnecter et supprimer l'environnement d'exécution** (Disconnect and delete runtime), puis confirmer.
- Cliquer sur **Connecter** en haut à droite. Si l'écran semble figé, recharger la page avec **Ctrl + R**.
- **Exécution** > **Tout exécuter** (Run all), ou **Ctrl + F9** : les cellules sont rejouées dans l'ordre, de haut en bas. Beaucoup de bogues viennent simplement de cellules lancées dans le désordre.

Un fichier téléversé hier a disparu ? Ce n'est pas un bogue : le panneau Fichiers de Colab est un brouillon effacé à chaque fin de session. Le calepin, lui, est dans le Drive et n'a rien perdu. Retéléversez le fichier de

données, ou rangez-le dans le dossier du cours du Drive et branchez le Drive au début du calepin avec `from google.colab import drive` puis `drive.mount("/content/drive")`.

C'est terminé

Colaboratory est branché sur votre Google Drive, et vous savez faire le chemin complet d'un laboratoire : le télécharger depuis le site, le déposer dans le dossier du cours, le travailler dans le navigateur et remettre le fichier `.ipynb` avec ses résultats. Sur votre propre ordinateur, revenez à l'[installation de Visual Studio Code](#) dès qu'elle fonctionne : c'est elle qui vérifie vos types en mode strict.