

Dernière mise à jour	Informatique	Denis DEFAUCHY
29/03/2021	Bases de la programmation	2 - Logiciel

Informatique

2

Logiciel

Cours

Dernière mise à jour	Informatique	Denis DEFAUCHY
29/03/2021	Bases de la programmation	2 - Logiciel


Logiciel	3
1.I. Manipulation d'un système d'exploitation : Windows et dossiers	3
1.II. Manipulation d'un environnement de développement – Python/Pyzo	4
1.II.1 Définition	4
1.II.2 Windows et Mac	4
1.II.3 Installation logicielle : « Pyzo » + « Miniconda »	5
1.II.3.a Installation	5
1.II.3.b Gestion des bugs Windows	7
1.II.3.c Désinstallation Windows	7
1.II.3.d Gestion des bugs Apple	8
1.II.4 Premiers pas avec Python sous « Pyzo »	9
1.II.4.a Lancement	9
1.II.4.b Premiers calculs	10
1.II.4.c Outils à connaître	12
1.III. Google Colab	13
1.III.1 Connexion	13
1.III.2 Premiers pas	14
1.III.3 Google drive	15
1.III.3.a Connexion	15
1.III.3.b Utilisation de Colab	16
1.III.3.b.i Lieu des fichiers de code	16
1.III.3.b.ii Chemins des fichiers utilisés par le code	16
1.III.4 Avantages et inconvénients	18
1.III.4.a Avantages	18
1.III.4.b Inconvénients	18
1.III.5 Utilisation mobile	18

Dernière mise à jour	Informatique	Denis DEFAUCHY
29/03/2021	Bases de la programmation	2 - Logiciel

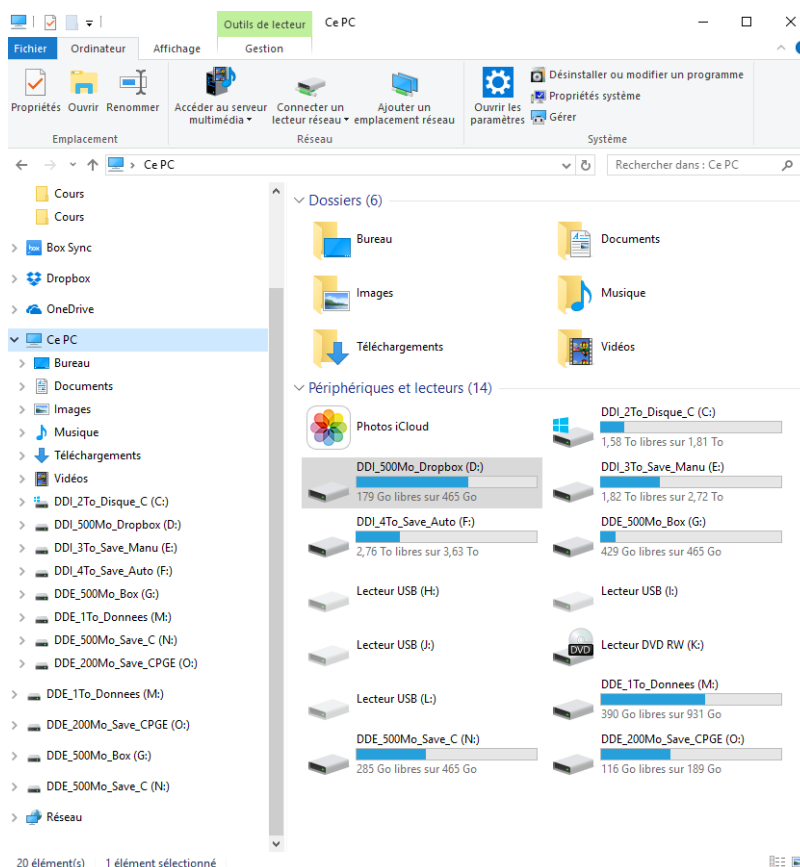
Logiciel

1.1. Manipulation d'un système d'exploitation : Windows et dossiers


Ce court paragraphe a pour but de vous montrer comment gérer la manipulation des fichiers sous Windows, mais je pense que vous maîtrisez tous déjà parfaitement cela.


Dans Windows, ouvrons donc un explorateur de fichier à l'aide de l'icône  probablement présente sur le bureau. Sinon, ouvrez le poste de travail !

A quelques détails près, cela ressemble à ça :



Sous la ligne « Ce PC », au lycée, et lorsque vous êtes logués avec votre session (prénom.nom) pour l'identifiant, et votre date de naissance sans / du type 01012000 pour mot de passe (avant de le changer), vous verrez apparaître un dossier à votre nom. C'est ici que vous stockerez vos documents que vous retrouverez d'un ordinateur à l'autre.

Je n'ai pas grand-chose à rajouter, si ce n'est qu'il peut être utile de savoir copier/coller l'adresse d'un répertoire, par exemple, je vais dans  > Ce PC > Images > Captures d'écran. Il suffit de cliquer à droite du dernier nom de dossier, ici « Captures d'écran », et on alors le chemin absolu qui apparaît :

 C:\Users\denis\OneDrive\Images\Captures d'écran

Dernière mise à jour	Informatique	Denis DEFAUCHY
29/03/2021	Bases de la programmation	2 - Logiciel

1.II. Manipulation d'un environnement de développement – Python/Pyzo

1.II.1 Définition

Un environnement de développement est un ensemble d'outils qui permettent d'éditer un code dans un langage de programmation donné, de le compiler, de le déboguer et de l'exécuter pour en afficher les résultats éventuels.

1.II.2 Windows et Mac

Vous êtes sous Windows, bien heureux soyez-vous !

Vous êtes sous mac ? Vous pourrez faire 99% de l'année avec Pyzo pour mac. Mais le 1% restant vous fera perdre des heures cruciales.

Un conseil donc : Utilisez l'utilitaire Boot camp, et installez une version de Windows sur votre mac. Il faut une image iso de Windows, et une licence. Cela vous permettra au passage d'utiliser Solidworks en SI.

Dernière mise à jour	Informatique	Denis DEFAUCHY
29/03/2021	Bases de la programmation	2 - Logiciel

1.II.3 Installation logicielle : « Pyzo » + « Miniconda »

1.II.3.a Installation

Partons du principe que vous disposez d'un ordinateur avec une version récente de Windows installée.

Bien qu'ils soient déjà à votre disposition en salle d'informatique, je souhaite ici vous expliquer comment avoir ces logiciels à votre disposition chez vous.

Il existe plusieurs possibilités pour programmer en Python. Je vous propose de vous rendre sur le site ci-dessous :

<https://pyzo.org/start.html>

Téléchargez et installez dans le répertoire par défaut « Pyzo », logiciel qui permettra de taper du code dans une interface appropriée.

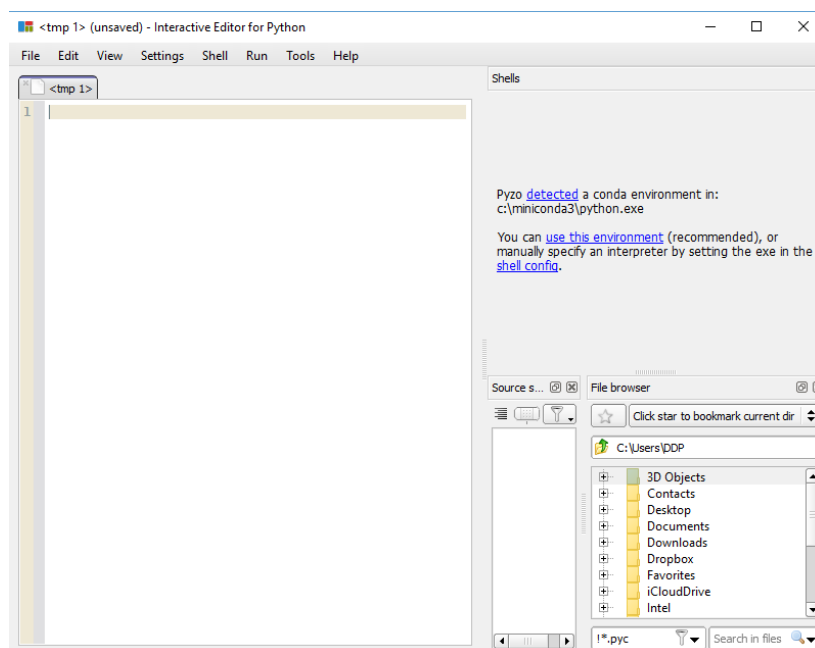
Au choix, téléchargez et installez dans le répertoire par défaut :

- « Anaconda » : contient tous les packages dont nous aurons besoin, mais prend de la place
- « Miniconda » : nécessitera l'installation supplémentaire de quelques modules

Si vous avez un système d'exploitation 64 bits, je vous recommande fortement d'installer la version pour 64, ce qui vous permettra d'utiliser plus que 4 Go de RAM si nécessaire...

Nous utilisons Python 3 et non Python 2 (mais si vous suivez les directives, vous aurez bien Python 3 !)

Lorsque tout est installé, lancez « Pyzo » (icône sur le bureau), vous devriez voir apparaître ceci (avec anaconda ou miniconda):



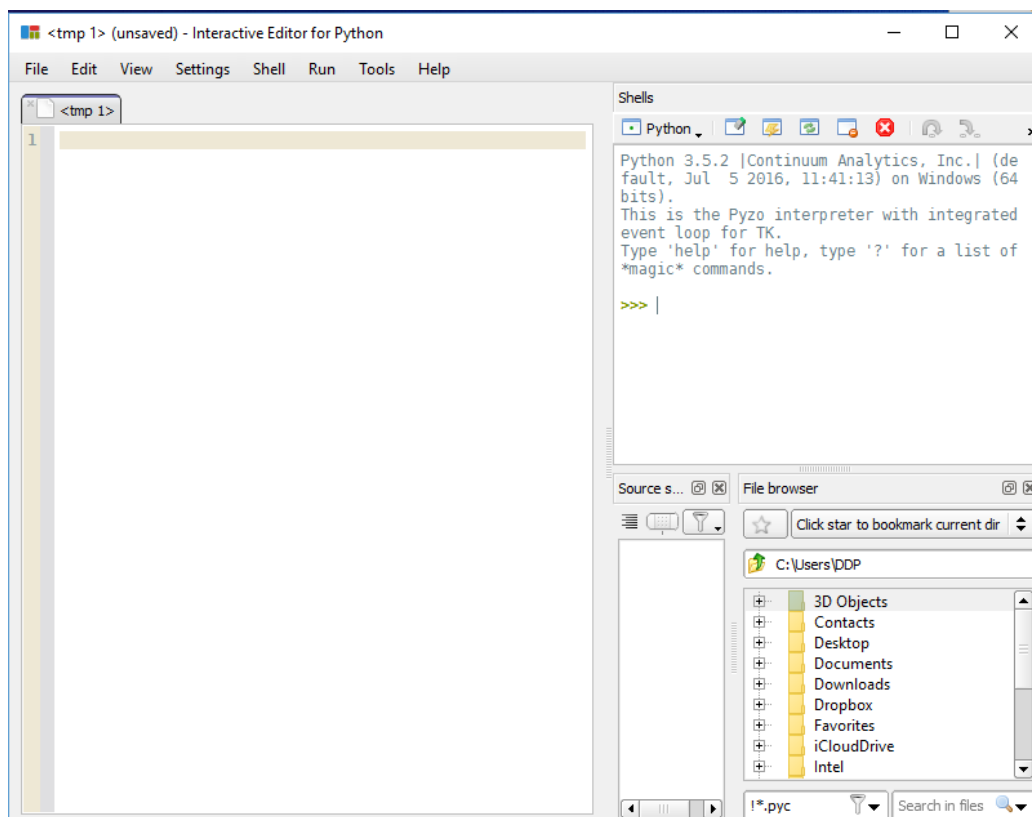
Dernière mise à jour	Informatique	Denis DEFAUCHY
29/03/2021	Bases de la programmation	2 - Logiciel

Dans la fenêtre à droite, normalement, Pyzo a reconnu un « conda environment », cliquez sur les mots bleus « use this environment » :

```
Pyzo detected a conda environment in:
c:\miniconda3\python.exe

You can use this environment (recommended), or
manually specify an interpreter by setting the exe in the
shell config.
```

Vous voyez alors apparaître ceci :



Si vous avez chargé « Miniconda », dans la fenêtre à droite, après les sigles >>>, tapez d’abord **conda install numpy** afin d’installer le module « numpy », pressez « Entrée », attendez, puis lors du message « Proceed ([y]/n) », écrire « y » puis pressez « Entrée » et attendez la fin de l’installation. Ensuite, essayez de taper « import numpy » et vérifiez qu’il n’y a pas d’erreurs. S’il y en a, mis à part chercher longuement en ligne, je vous conseille de revenir en arrière et de désinstaller miniconda pour une version plus à jour, voire rétrogradée, voir même de faire la même chose pour pyzo. J’ai partagé les liens deux pages plus haut !

Procédez de même avec la phrase : **conda install pyqt matplotlib pandas sympy scipy**

Attention : selon ce que vous avez téléchargé (regular, miniconda, anaconda), il se peut que des modules soient déjà présents, vous en serez avertis.

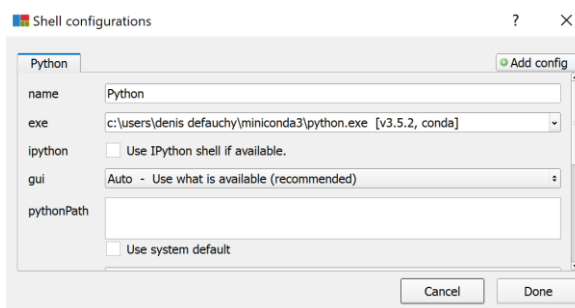
Vous êtes prêts à programmer en Python et à faire du calcul scientifique. Autrement dit, vous avez le piano, reste à apprendre à lire des partitions puis à jouer.

Dernière mise à jour	Informatique	Denis DEFAUCHY
29/03/2021	Bases de la programmation	2 - Logiciel

1.II.3.b Gestion des bugs Windows

Je complète ce paragraphe au fil du temps, selon les problèmes rencontrés par les élèves, et vous propose des pistes de résolution de ceux-ci :

- Parmi vous, certains auront des problèmes avec certains modules, en particulier « numpy ». Si tel est le cas, le mieux est d'aller chercher les dernières versions, voir de plus anciennes, de « miniconda », voire de « pyzo », afin de régler ces problèmes. Alors voici où les trouver :
 - o Pyzo : <https://github.com/pyzo/pyzo/releases>
 - o Miniconda : <https://repo.anaconda.com/miniconda/>
 - o Une solution peut consister à installer Pyzo en 64 bits (si vous avez un processeur 64 et un système 64) et Miniconda en 32 ! (valable en fin 2018)
- En cas d'erreur PermissionError 13 lors de l'installation des modules, essayez ce qui suit : [LIEN](#)
- En cas d'erreur CondaHTTPError: HTTP 000 CONNECTION FAILED lors de l'installation des modules
 - o Tenter de désactiver le pare-feu Windows le temps de l'installation des modules
 - o Essayer de mettre à jour pip en tapant « pip install --upgrade pip » puis installer les modules, soit avec la commande normale « conda install ... », soit en écrivant à la place « pip install ... »
- Après une réinstallation, il se peut que vous n'ayez plus accès à la console (une erreur mentionnant le mot Kernel de mémoire). Pyzo essaye d'ouvrir l'ancienne version de python qui a été désinstallée. Cliquer sur « Python » dans Shells, puis « Edit shell configurations... ». Il faut alors aller chercher le bon chemin dans « exe » via le menu déroulant, et relancer le shell ;)

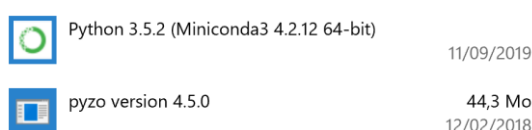


1.II.3.c Désinstallation Windows

Attention, ne jamais désinstaller un programme en supprimant son répertoire, mais utiliser l'outil Windows prévu à cet effet.

Faites un clic droit sur le menu démarrer, et aller dans « Applications et fonctionnalités ».

Trouvez alors vos logiciels préférés :



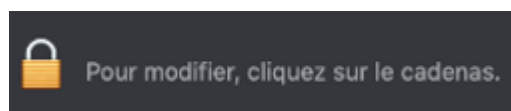
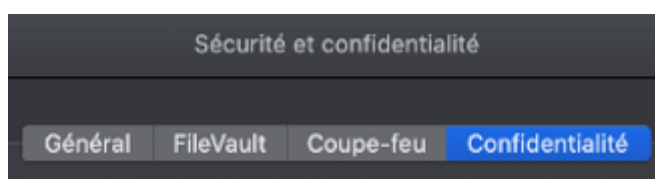
Cliquez dessus, et désinstallez-les !

Dernière mise à jour	Informatique	Denis DEFAUCHY
29/03/2021	Bases de la programmation	2 - Logiciel

1.II.3.d Gestion des bugs Apple

Bien que je ne maîtrise pas ce système d'exploitation, quand des élèves me remontent des difficultés rencontrées et la solution associée, je les ajoute ici (sans les avoir vérifiées) :

- Message « Impossible d'ouvrir « pyzo » car le développeur ne peut pas être vérifié » : quitter l'application, aller dans les réglages « Préférences système », « Sécurité et confidentialité », cliquer sur le cadenas en bas, entrer votre code/empreinte..., relancez Pyzo et ça fonctionne. Après cette première ouverture, remettre le cadenas (Merci Chanelle S.).

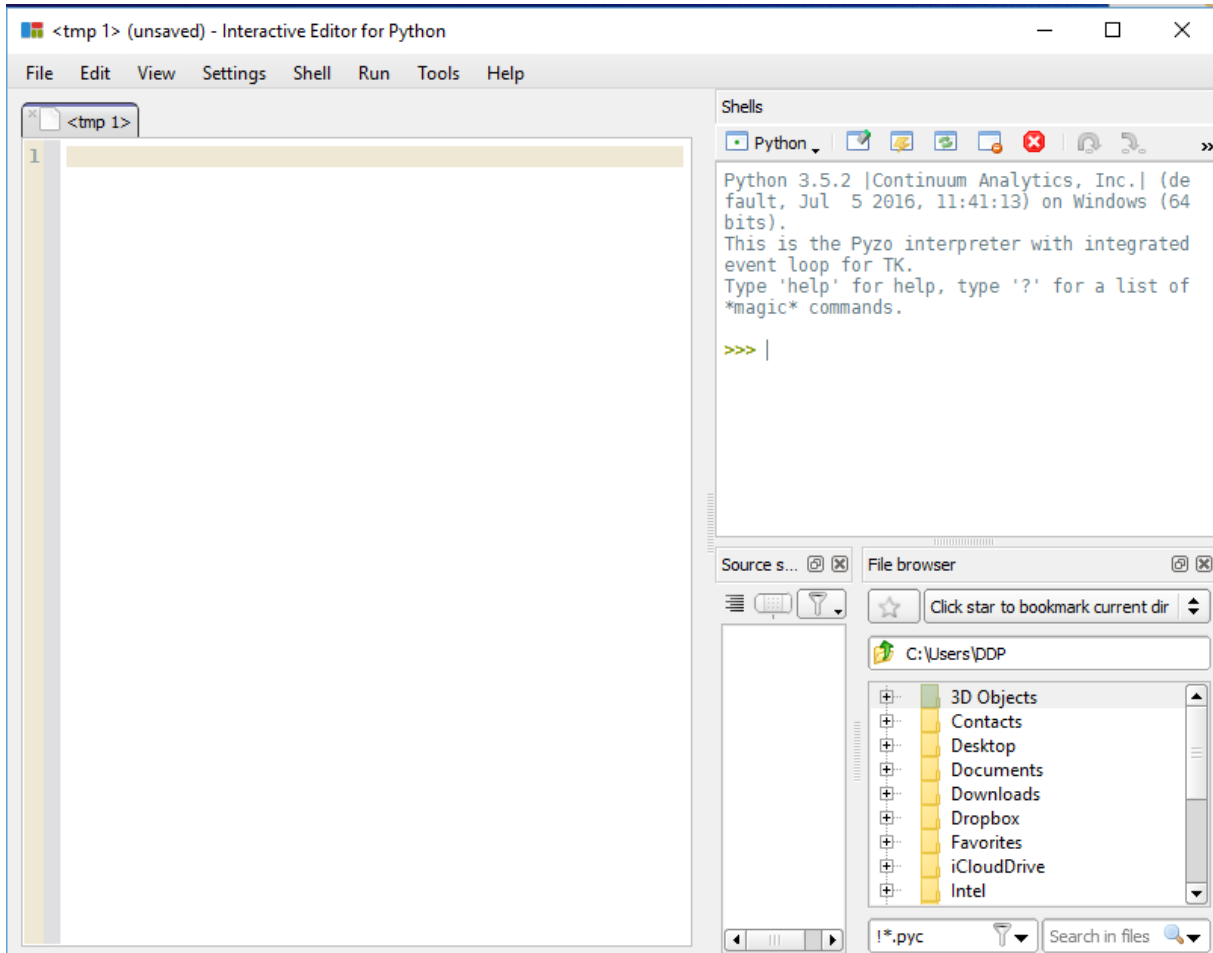


Dernière mise à jour	Informatique	Denis DEFAUCHY
29/03/2021	Bases de la programmation	2 - Logiciel

1.II.4 Premiers pas avec Python sous « Pyzo »

1.II.4.a Lancement

Revenons sur l’affichage de Pyzo :



La partie gauche est un fichier appelé « tmp 1 » dans lequel vous allez pouvoir inscrire du code en vue de l’exécuter intégralement dès que nécessaire. C’est un peu comme écrire une partition. L’écriture de la partition ne fait pas de musique !

A droite, dans le cadre du haut, on voit ce que l’on appelle la « console » ou « Shells », c’est-à-dire la zone où les calculs se font. C’est l’instrument de musique. Tout ce qui y est écrit sera exécuté dès l’appuie sur « Entrée ». Toutefois, à la différence d’un instrument de musique, ne sortira que ce que le programmeur a demandé, et heureusement, car s’il y a des milliers de calculs par secondes, on ne veut que le résultat...

Vous pouvez réorganiser les fenêtres à convenance en cliquant sur leur nom, en laissant appuyé le bouton gauche et en la déplaçant. De même, à l’interface entre fenêtres, vous verrez apparaître une double flèche qui permettra de redimensionner chaque fenêtre.

Dernière mise à jour	Informatique	Denis DEFAUCHY
29/03/2021	Bases de la programmation	2 - Logiciel

1.II.4.b Premiers calculs

Lorsque l'on écrit 10+10 dans la console, et que l'on tape entrée, on obtient le résultat :


```
>>> 10+10
20
```

Remarque : il vous sera souvent très utile de pouvoir réécrire une ligne qui a été précédemment exécutée sans avoir besoin d'utiliser la souris pour faire copier/coller. Pour cela, rien de plus simple. Dans la console, sur une nouvelle ligne vierge, il suffit de taper sur la **flèche du haut** autant de fois qu'il faut remonter les lignes. Exemple : tapez 5+5 puis « Entrée », puis 10+10 et « Entrée », puis 20+20 et « Entrée » :

```
>>> 5+5
10
>>> 10+10
20
>>> 20+20
40
```

Pour pouvoir exécuter la ligne 5+5, il suffit de cliquer 3 fois sur la flèche vers le haut puis « Entrée ».

Intéressons-nous maintenant au fichier à gauche. Créons un fichier personnel dans lequel nous allons entrer du code. Faire « Fichier », « Save as... » puis enregistrer le fichier avec un nom personnel (par exemple « Essai_1 ») et dans un répertoire personnel de l'ordinateur, pour moi ce sera sur le bureau.

Il peut être utile d'avoir accès à plusieurs fichiers python dans un même répertoire de travail. Dans la partie en bas à droite, nommée « File browser », cliquez sur le symbole  et allez ouvrir le dossier dans lequel vous avez mis votre fichier. Vous devriez alors le voir apparaître en dessous.

On voit maintenant dans la partie gauche notre fichier, vierge, dans lequel je vous invite à écrire 10+10 puis taper Entrée.

```
1 10+10
2
```

Il ne se passe rien, c'est normal, on a en fait préparé une ligne de code à exécuter lorsqu'on le souhaitera. Pour ce faire, il suffit de taper la touche « F5 » qui

- Enregistre le fichier
- L'exécute dans la console

Dans la console, on voit alors :

```
>>> (executing line 1 of "Essai_1.py")
>>>
```

L'exécution a eu lieu, une nouvelle ligne permet d'écrire du code, il n'y a donc pas eu d'erreurs. L'ordinateur nous informe qu'il a exécuté le fichier mais on ne voit pas le résultat de notre calcul. En effet, la console et le fichier ne se comportent pas de la même façon. La console exécute et donne le résultat du code tapé directement alors que l'exécution d'un fichier exécute chacune de ses lignes sans renvoyer de résultat tant qu'il n'est pas demandé, et heureusement.

Dernière mise à jour	Informatique	Denis DEFAUCHY
29/03/2021	Bases de la programmation	2 - Logiciel

Imaginons qu'un calcul prenne un milliard de calculs intermédiaires, heureusement que ce milliard de calculs n'apparaît pas.

Pour afficher le résultat d'un calcul, il faut le demander avec la commande « print() ».

Entrez dans le fichier le code suivant :

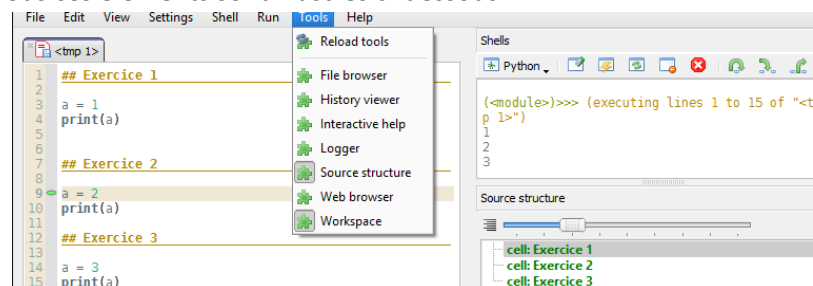
```
1 print(10+10)
2
```

Puis exécutez le fichier avec la touche « F5 » ou « CTRL+E », vous verrez alors apparaître le résultat :

```
>>> (executing line 1 of "Essai_1.py")
20
```

Remarques :

- Sur certains claviers, F5 est accessible moyennant l'appui simultané sur la touche « fn » à priori en bas à gauche du clavier
- Vous pouvez changer la taille du texte du logiciel en faisant la touche CTRL et + ou -
- Il est possible de n'exécuter qu'une partie d'un fichier, pour cela il faut sélectionner le code en question et faire Alt + Entrée.
- On vide la console (mais pas les variables en mémoire) en écrivant cls dans la console, puis touche Entrée
- Pour réinitialiser la console, Ctrl+1 😊 Sinon, crois rouge puis Python – New shell – Create...
- Il est possible de créer des « cell » (cellules), c'est-à-dire des parties de programme, par exemple des exercices, que l'on peut exécuter séparément en plaçant le curseur dans la partie concernée, puis en faisant « Ctrl+Entrée ». On peut par ailleurs simplement naviguer d'une « cell » à l'autre et donc la sélectionner en affichant les « cell » existantes en faisant « Tools » puis cliquer sur « Source structure » qui affiche la fenêtre « Source structure » avec les « cell » existantes. Tous ces éléments sont illustrés ci-dessous :



Lorsqu'il y a une erreur dans un code, empêchant son exécution, un message apparaît dans la console en rouge. L'erreur est indiquée : N° ligne + position à l'aide d'un ^.

```
a = 1
b = 2
c = a + 2b
```

File "<tmp 2>", line 6
 c = a + 2b
 ^
 SyntaxError: invalid syntax

Attention, il faut retenir une chose importante : **Un chapeau en début de ligne correspond généralement à une erreur à la ligne précédente. Par exemple, une parenthèse oubliée :**

```
a = 1
b = (a + 1
print(a)
```

File "<tmp 1>", line 3
 print(a)
 ^
 SyntaxError: invalid syntax

Dernière mise à jour	Informatique	Denis DEFAUCHY
29/03/2021	Bases de la programmation	2 - Logiciel

1.II.4.c Outils à connaître

Pour effectuer des additions, soustractions, multiplications, et divisions, il suffit d'utiliser les symboles + - * / disponibles au niveau du clavier numérique.

Il est bon de savoir calculer une puissance :

Pour calculer 2^3 , il faut écrire `2**3` :

```
>>> 2**3
8
```

Pour toutes les autres fonctions mathématiques (cos, log, exp, racine (sqrt)...), nous verrons qu'il est nécessaire d'importer des librairies car python ne sait pas, à priori, ce que veut dire cos par exemple.

Il est très utile de connaître la commande du « modulo », qui s'écrit $x[n]$ en mathématiques, et `x%n` sous Python. Il donne le reste dans la division euclidienne de x par n . Plus généralement, on a :

$$\begin{cases} a = bq + r \\ q = a//b \\ r = a\%b \end{cases}$$

La valeur absolue s'obtient avec la commande `abs(x)`. La fonction `abs` n'a pas à être importée.

Enfin, il est bien de connaître les raccourcis suivants :

<code>a = a + 1</code>	<code>a += 1</code>
<code>a = a - 1</code>	<code>a -= 1</code>
<code>a = a * 1</code>	<code>a *= 1</code>
<code>a = a / 1</code>	<code>a /= 1</code>
Une nouvelle variable <code>a</code> est créée, de même nom que l'ancienne	La variable <code>a</code> est modifiée

Attention : Même ci pour des réels, il n'y a pas de différence, nous verrons dans le paragraphe sur les listes que le choix de « `a +=` » qui modifie `a` ou « `a = a +` » qui crée une nouvelle variable `a` peut avoir une influence non négligeable (modification d'une liste dans une liste).

Dernière mise à jour	Informatique	Denis DEFAUCHY
29/03/2021	Bases de la programmation	2 - Logiciel

1.III. Google Colab

Google Colab est à Pyzo ce que Word est à un éditeur de texte en txt. Cela veut le coup d'essayer, et d'aller plus loin, je me limiterai à un tuto de prise en main.

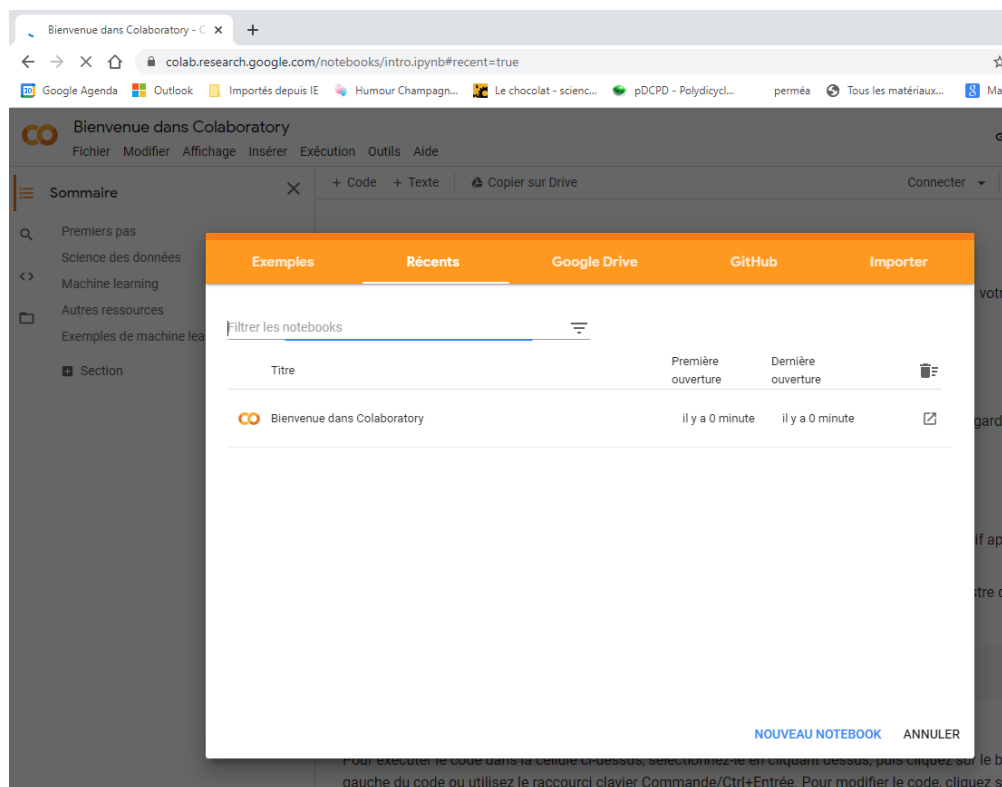
1.III.1 Connexion

Ce que je vais décrire ci-dessous pourrait ne pas correspondre directement à ce que vous verrez, cela dépend des mises à jour qui peuvent intervenir. Je sais que vous vous adapterez.

Après vous être connectés à votre compte Google depuis le navigateur, rendez-vous sur ce lien :

<https://colab.research.google.com/>

Vous verrez alors cela :

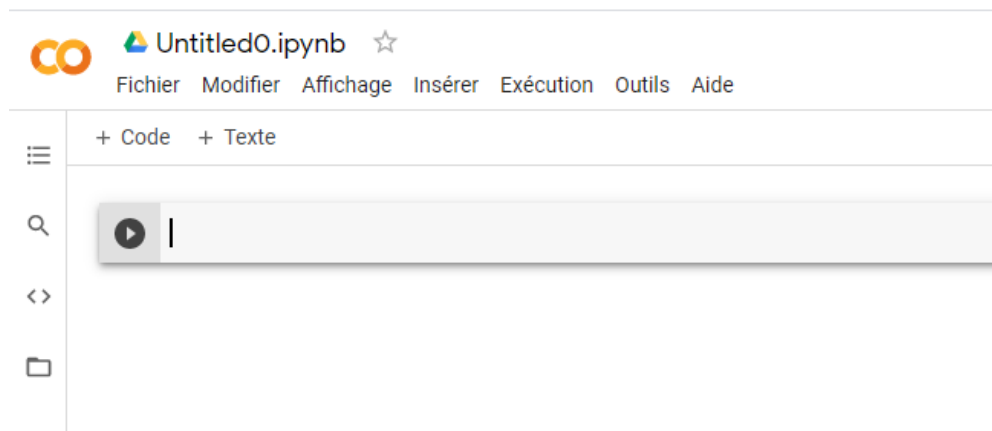


Cliquez sur « Nouveau Notebook », en bas à droite.

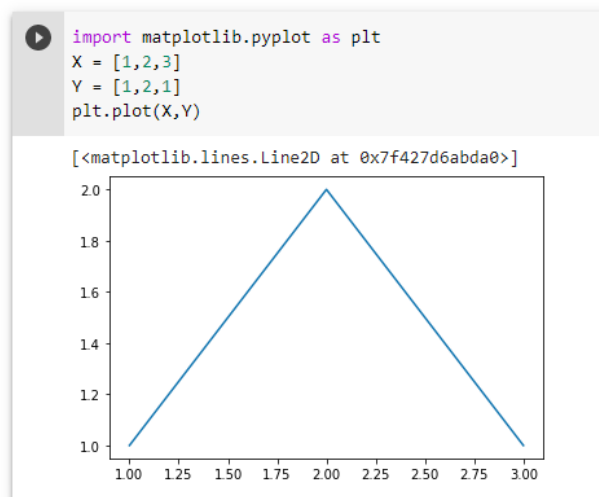
Dernière mise à jour	Informatique	Denis DEFAUCHY
29/03/2021	Bases de la programmation	2 - Logiciel

1.III.2 Premiers pas

Vous voilà déjà prêts à coder :

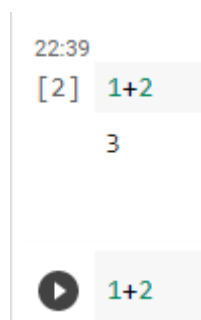


Taper du code dans la fenêtre puis « Lecture » :



Oui, pas de « plt.show() », j'en parlerai après !

Je vous suggère de faire « Affichage » puis « Historique d'exécution » afin d'obtenir une console à droite :



Dernière mise à jour	Informatique	Denis DEFAUCHY
29/03/2021	Bases de la programmation	2 - Logiciel

1.III.3 Google drive

1.III.3.a Connexion

Il reste à activer Google Drive pour rendre les choses plus faciles à gérer. Cliquer sur la 4° icône à gauche « Dossier » puis sur la 3° :



Autorisez alors l'accès à votre « Google Drive » afin de pouvoir stocker vos codes, mais aussi des photos ou fichiers textes sur lesquels travailler.

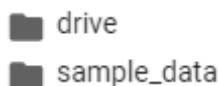
Autoriser ce notebook à accéder à vos fichiers Google Drive ?

La connexion à Google Drive autorisera le code exécuté dans ce notebook à modifier les fichiers contenus dans vos dossiers Google Drive.

NON, MERCI

SE CONNECTER À GOOGLE DRIVE

Si vous avez bien un compte Google drive avec votre compte Google utilisé pour Google Colab, apparaît alors le dossier « Drive » :



Sinon, il se peut qu'il faille réaliser une procédure indiquée pour connecter un autre Drive (exécution d'un code « `from google.colab import drive` » + procédure d'acceptation). Il faudra à la fin de la procédure cliquer que « Actualiser » pour que « drive » apparaisse.



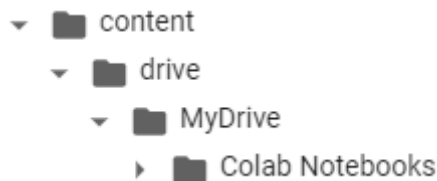
Si vous avez par mégarde cliqué sur cela :



Il suffit de revenir dans le dossier « content ».

Dernière mise à jour	Informatique	Denis DEFAUCHY
29/03/2021	Bases de la programmation	2 - Logiciel

Allez ensuite dans « drive » puis « my drive ».

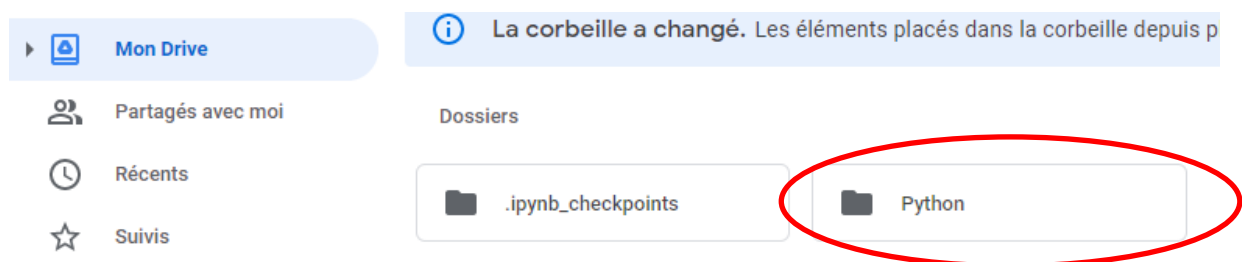


Faites les 3 petits points à droite de « My Drive » puis créer un « Nouveau dossier », appelé par exemple « Python ».



Allez dans votre Google drive ici : https://www.google.com/intl/fr_tg/drive/

Vous y verrez ce dossier :



Revenez dans Colab. Vous avez donc maintenant accès à un dossier python qui est dans votre Google Drive 😊.

1.III.3.b Utilisation de Colab

1.III.3.b.i Lieu des fichiers de code

De base, quand vous faites « Fichier » puis « Enregistrer une copie dans Google Drive », les fichiers sont stockés dans un dossier du Drive nommé « Colabs Notebooks ». Il vous sera ensuite possible de les déplacer depuis Google Drive (dommage de ne pas pouvoir directement enregistrer où l'on veut depuis Colab). Vous pourrez ensuite les ouvrir via « Fichier » puis « Ouvrir le Notebook », puis en haut « Google Drive ». Il trouvera tous les fichiers disponibles (oui, pas d'arborescence de dossiers, dommage aussi...). Alors nommez correctement vos codes afin de les retrouver avec le filtre.

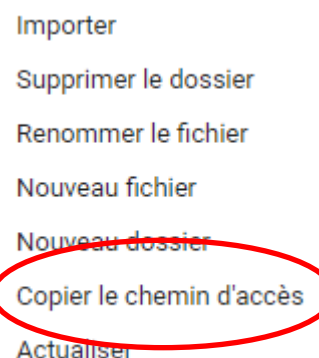
1.III.3.b.ii Chemins des fichiers utilisés par le code

Où que soit votre fichier Colab, le chemin de base (quand vous ne précisez rien) pour l'ouverture d'un fichier texte, d'une image, ou pour l'enregistrement, est le dossier « content » dans « Colab ».

Il faudra préciser le chemin de vos fichiers à partir de ce dossier. Pour cela, naviguez depuis Colab vers le dossier cible :



Utilisez les 3 petits points à droite pour « Copier le chemin d'accès » :



Dernière mise à jour	Informatique	Denis DEFAUCHY
29/03/2021	Bases de la programmation	2 - Logiciel

Il suffira ensuite de le coller sans rien changer là où c'est nécessaire, pour moi cela donne :

/content/drive/MyDrive/Python

Finalement, essayez par exemple ce code :

```
import matplotlib.pyplot as plt
import time

def Polygon(fig,Liste_Points):
    X = []
    Y = []
    for Point in Liste_Points:
        X.append(Point[0])
        Y.append(Point[1])
    plt.figure(fig)
    plt.fill(X,Y,'k')

from math import sqrt
import numpy as np

A = np.array([0,0])
B = np.array([1,0])
C = np.array([0.5,sqrt(3)/2])

def Sierpinski(fig,n,A,B,C):
    if n == 1:
        Polygon(fig,[A,B,C])
    else:
        Sierpinski(fig,n-1,A,(A+B)/2,(A+C)/2)
        Sierpinski(fig,n-1,B,(A+B)/2,(B+C)/2)
        Sierpinski(fig,n-1,C,(A+C)/2,(B+C)/2)

fig = 1
plt.figure(fig)
plt.show()

tic = time.perf_counter()
Sierpinski(fig,10,A,B,C)
toc = time.perf_counter()
Temps = toc - tic
print(Temps)

plt.savefig('/content/drive/MyDrive/Python/Sierpinski.jpg')
```

Vous verrez que l'image est sauvegardée dans votre dossier « Python » dans Google Drive.

Dernière mise à jour	Informatique	Denis DEFAUCHY
29/03/2021	Bases de la programmation	2 - Logiciel

1.III.4 Avantages et inconvénients

1.III.4.a Avantages

Stockage en ligne, modules à jour, ergonomie, accessible depuis un téléphone, une tablette, un ordinateur, permettant une belle organisation interactive... Mais surtout ?

Essayez le code de Sierpinski sur un ordinateur et sur Colab en enlevant le savefig pour éviter les problèmes de chemin non existant, vous verrez la différence de temps d'exécution d'un code en $O(3^n)$. Chez moi, 12 s sur Colab contre 124 s sur l'ordinateur... On profite de la puissance des ordinateurs de chez Google !

1.III.4.b Inconvénients

Je ne maîtrise pas encore cet outil, mais j'ai déjà pu expérimenter une différence importante entre l'utilisation de Pyzo (Python) et colab (iPython) avec matplotlib. Regardez plutôt :

Code	Pyzo	Colab
<i>D'abord :</i> <pre>import matplotlib.pyplot as plt X = [1,2,3,4,5] Y1 = [1,1,2,2,3] Y2 = [2,1,1,1,3] Y3 = [3,1,1,1,3] Y4 = [4,1,1,1,3] fig1 = 1 fig2 = 2</pre>		
<i>Puis, au choix :</i> <pre>def Affiche(fig,X,Y): plt.figure(fig) plt.plot(X,Y) plt.show()</pre>	Trace bien les 2 courbes sur chaque figure 😊	Trace 4 courbes sur 4 figures différentes 😞
<pre>def Affiche(fig,X,Y): plt.figure(fig) plt.plot(X,Y)</pre>	Ne trace rien 😞	Trace bien les 2 courbes sur chaque figure 😊
<pre>def Affiche(fig,X,Y): plt.figure(fig) plt.plot(X,Y) plt.figure(fig1) plt.show() plt.figure(fig2) plt.show()</pre>	Fonctionne correctement dans les deux cas 😊	
<i>Puis :</i> <pre>Affiche(fig1,X,Y1) Affiche(fig2,X,Y2) Affiche(fig1,X,Y3) Affiche(fig2,X,Y4)</pre>		

Pour en savoir un peu plus, cherchez du côté de « iPython » et lisez cette page ([LIEN](#)). Une solution à explorer serait le plt.draw() mais je m'arrête là.

1.III.5 Utilisation mobile

Il est possible d'utiliser Google Colab directement depuis un navigateur ([LIEN](#)), une appli android sur [Google play](#) (non essayée) et sur l'[App store](#) (fonctionne très bien – Inutile d'acheter la version PRO).