

Fiche de TP numéro 1 - Découverte de l'environnement et Fonctions

Étape 1 - La première chose à faire, c'est de positionner le proxy pour pouvoir accéder à Internet. Allez dans les paramètres du système (tout en haut à droite de votre écran). Sélectionnez *Réseau*, puis *Serveur mandataire* puis positionnez le proxy pour les protocoles http, https et ftp : `cache-etu.univ-artois.fr` avec le port **3128**.

Pour réaliser les travaux pratiques, vous utiliserez le logiciel Visual Studio Code (VS Code).

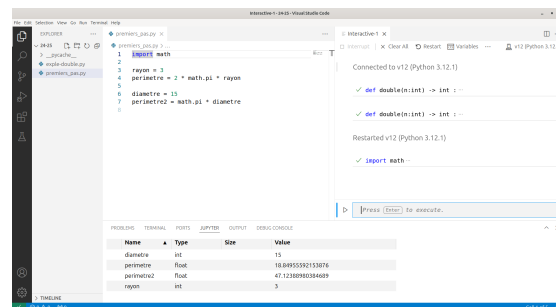
Étape 2 - Dans votre répertoire personnel, créez un répertoire `algo1`, et un sous-répertoire `tp1`. Lancez maintenant VS Code.

Ceux qui ont déjà programmé en Python et qui connaissent VS Code peuvent aller directement aux exercices.

Ceux qui ont déjà recodé et testé avec la fenêtre interactive tous les exemples du cours avec VS Code peuvent aussi aller directement aux exercices.

Ceux qui ont déjà programmé en Python et qui ne connaissent pas VS Code doivent lire la partie suivante avant de démarrer les exercices.

Et tous les autres suivent le guide !



Visual Studio Code (VS Code) est un environnement de programmation puissant qui permet de travailler dans beaucoup de langages. Vous l'utiliserez pour vos exercices en Python. Il est composé de plusieurs sections importantes :



Exploreur de fichiers : à gauche, il vous permettra de naviguer entre vos différents fichiers


Éditeur : en haut, partie gauche. C'est là où vous saisirez vos programmes. L'éditeur vous permettra de les sauvegarder dans des fichiers ;

Fenêtre interactive : en haut, partie droite. C'est ici que vous testerez votre code.

et bien d'autres éléments que vous découvrirez au fur et à mesure.

Étape 3 - Vous allez maintenant installer votre poste de travail.

1. Ouvrez VS Code (icône )
2. Vous préférez travailler sur un fond blanc ? sur un fond noir ? Votre confort visuel est important. Cliquez sur l'icône . Choisissez Themes/Color Theme, puis choisissez l'un des deux thèmes Light (Visual Studio) ou Dark (Visual Studio). Il y a beaucoup d'autres thèmes ? Oui, c'est vrai, mais vous les testerez en dehors des séances de TP.

3. Il faut maintenant ajouter les outils dont vous avez besoin. Installez l'extension **Python** :
 - Cliquez sur l'icône des extensions sur la barre latérale gauche  (ou utilisez le raccourci `Ctrl+Shift+X`).
 - Dans la barre de recherche des extensions, tapez **Python** et installez l'extension officielle de Microsoft.
 - Choisissez l'interpréteur Python, et "install release version",
 - Installez également python debugger
4. Cherchez maintenant l'extension pour **Jupyter** et installez-la. Cette extension va nous permettre de travailler en mode interactif.
5. Pour utiliser Jupyter, il faut en plus que Jupyter Notebook soit installé sur votre machine. Dans la barre de menu, en haut de VS Code, choisissez `Terminal/New Terminal`. Vous pouvez maintenant l'installer via la commande suivante dans le terminal : `pip install notebook`.
6. Sélection de l'interprète Python :
 - Une fois l'extension installée, ouvrez votre répertoire `algo1/tp1` (menu `File/Open Folder`).
 - Ouvrez un fichier `.py` dans l'éditeur (ou créez-en un nouveau). (Enregistrez votre fichier en utilisant `Ctrl+S`. Dans la boîte de dialogue d'enregistrement, enregistrez le et nommez-le `intro.py`)
 - En bas à droite de VS Code, un message devrait apparaître vous demandant de sélectionner un interpréteur Python. Cliquez dessus.
 - Si vous ne voyez pas ce message, vous pouvez aussi manuellement choisir l'interpréteur en appuyant sur `Ctrl+Shift+P` pour ouvrir la palette de commandes, puis tapez `Python: Select Interpreter`.
 - Dans la liste qui apparaît, sélectionnez la version recommandée de Python déjà installée sur votre machine.

Étape 4 - VS Code vous permet d'utiliser des fichiers Python en mode interactif, ce qui est très pratique pour tester et expérimenter avec votre code. Voici comment utiliser un fichier Python dans le mode interactif avec Jupyter :

1. Ouvrez le fichier Python que vous souhaitez utiliser dans VS Code.
2. Pour exécuter le fichier Python dans le mode interactif :
 - Cliquez avec le bouton droit sur le fichier ouvert dans l'éditeur et sélectionnez `Run Current File in Interactive Window`.
 - Vous pouvez également ouvrir la palette de commandes en appuyant sur `Ctrl+Shift+P` (ou `Cmd+Shift+P` sur Mac), tapez `Run File in Interactive Window` et appuyez sur `Entrée`.
3. Utilisez le mode interactif :
 - Le code de votre fichier Python sera exécuté dans une fenêtre interactive où vous pourrez voir les résultats de chaque cellule de code.
 - Vous pouvez exécuter des portions spécifiques de votre code en sélectionnant les lignes souhaitées et en appuyant sur `Shift+Enter`.

Étape 5 - Pour exécuter votre code dans le mode interactif avec un raccourci clavier personnalisé :

- Ouvrez les raccourcis clavier en appuyant sur `Ctrl+Shift+P` (ou `Cmd+Shift+P` sur Mac) et tapez `Preferences: Open Keyboard Shortcuts`.
- Recherchez la commande `Python: Run File in Interactive Window`.
- Configurez un raccourci clavier personnalisé pour cette commande en cliquant sur l'icône de modification à côté de la commande, par exemple `Ctrl+Alt+R`.
- Vous pouvez ensuite utiliser ce raccourci pour exécuter rapidement votre code dans le mode interactif.

Étape 6 - Testez les expressions Python suivantes en les saisissant dans le prompt. Que se passe-t-il ?

```
3/2
3//2
(5//2)+8
```

Étape 7 - Saisissez maintenant l'expression `(5+3)*2` dans l'éditeur (le fichier que vous avez créé).

Pour exécuter ce qui est dans l'éditeur, il faut

- Cliquez avec le bouton droit sur le fichier ouvert dans l'éditeur et sélectionnez `Run Current File in Interactive Window`.

Que se passe-t-il ? Pourquoi ?

Étape 8 - Vous allez transformer votre programme pour qu'il contienne maintenant une instruction :

```
a = (5+3)*2
```

Sauvegardez votre programme (`Ctrl+S`). Exécutez-le. Que se passe-t-il ? Pourquoi ?


Saisissez maintenant dans le prompt l'expression :

```
>>> a
```

Que se passe-t-il ? Pourquoi ?


Étape 9 - On va regarder en détail ce qu'il vient de se passer. Cliquez sur l'icône de poubelle dans l'onglet du terminal (ou utilisez le raccourci `Ctrl+Shift+``) pour fermer le terminal actuel, puis ouvrez un nouveau terminal en cliquant sur `Terminal` dans la barre de menu en haut, puis sur `New Terminal`.

Ensuite, ouvrez le panneau des variables de débogage :

- Cliquez sur l'icône  dans la barre latérale gauche pour ouvrir le panneau de débogage.

Étape 10 - Vous allez maintenant exécuter votre programme en mode débogage.

Pour lancer le débogage :

- Cliquez sur le petit rond rouge à côté de la première ligne du fichier. Cliquez ensuite sur le bouton "Run and Debug" pour démarrer le débogage.
- Assurez-vous que le panneau des variables est visible. Vous pouvez le trouver sous l'onglet *Variables* dans le panneau de débogage sur la gauche.
- Utilisez les boutons de débogage pour avancer dans le code. Vous pourrez cliquer sur  (ou appuyez sur `F11`) pour entrer dans les fonctions appelées et observer leur exécution ligne par ligne (lorsque vous aurez écrit des fonctions).

Assurez-vous de bien comprendre ce qui se passe à chaque étape et observez les valeurs des variables dans le panneau de débogage à gauche.

Exercices

Pour chaque fonction, vérifiez que vous avez bien utilisé des noms de variables et constantes qui suivent les recommandations données en cours. Écrivez la spécification de la fonction. N'oubliez pas de bien tester votre fonction avant de passer à la suivante. Vous écrirez toutes ces fonctions dans un seul et même fichier `tp1.py`.

Exercice 1 : Écrire et spécifier une fonction `celsius_to_fahr` qui reçoit un nombre flottant `c` en argument, représentant une température en Celsius, et retourne la valeur correspondante en Fahrenheit. Utiliser la formule :

$$C = \frac{F - 32}{1.8}$$

Exemples d'exécution :

```
>>> celsius_to_fahr(100.0)
212.0
>>> celsius_to_fahr(30.0)
86.0
```

Exercice 2 : Écrire et spécifier une nouvelle fonction qui reçoit en argument trois nombres flottants hauteur, largeur_inf et largeur_sup en argument, représentant un trapèze. La fonction doit retourner l'aire du trapèze. La formule pour calculer l'aire du trapèze est :

$$A = \frac{1}{2}(x_1 + x_2)h$$

Exemple d'exécution :

```
>>> aire_trapeze(5.0, 10.0, 7.0)
42.4
```

Exercice 3 : Écrire et spécifier une fonction pour chacun des calculs ci-dessus :

- Aire du cercle : $A = \pi r^2$
- Aire de l'ellipse : $A = \pi r_1 r_2$
- Aire du triangle équilatéral : $A = \frac{c^2 \sqrt{3}}{4}$
- Volume du cône : $V = \frac{\pi r^2 h}{3}$
- Volume de la sphère : $V = \frac{4\pi r^3}{3}$
- Aire d'un triangle arbitraire : $A = \frac{1}{2}ab \sin(C)$

Exercice 4 : Écrire et spécifier une fonction `etoile_six_branches` qui retourne l'aire d'une étoile constituée de deux triangles équilatéraux. Vous penserez à réutiliser du travail déjà fait...

Exercice 5 : C'est l'été et vous avez décidé de repeindre votre chambre, les quatre murs et le plafond. Vous connaissez les dimensions de la pièce (largeur, longueur, hauteur), combien couvre 1 litre de peinture, et la contenance (en litres) des pots de peinture proposés. Écrire et spécifier une fonction qui retourne le nombre de pots de peinture à acheter. Proposez une décomposition en plusieurs petites fonctions.