

TP1

Il y a de nombreux objectifs pratiques à atteindre dans ce premier TP, qui serviront de base à tous les suivants : bien valider la manipulation de vos fichiers dans l'environnement de l'IUT, avoir une idée de ce qu'est un IDE, écrire et tester vos premières instructions et vos premiers programmes, mettre en pratique les premières notions du cours et des TD, rédiger un compte rendu selon le modèle proposé.

Choix de l'IDE:

- **Visual-Studio Code**, qui est un « éditeur de code extensible » (et non un Integrated Development Environment IDE, même si la limite entre les deux devient ténue). Il est spécialisé pour les programmeurs et dispose d'un mécanisme de plugins ajoutant à l'éditeur des fonctionnalités spécifiques à des langages par exemple. Nous utiliserons cette année le plugin « Python » de Microsoft (disponible sur Windows, Linux et Mac OS).

Remarques & travaux préliminaires

Organisation de vos fichiers :

Vous trouverez la documentation de connexion aux divers composants du réseau du département à l'emplacement suivant : `pedago-iut/INFO/Public/INFO-ROOT`

Vous pouvez travailler « en local » sur les machines virtuelles (c'est souvent plus rapide) mais vous devrez systématiquement sauvegarder vos fichiers dans l'un de vos espaces (répertoire personnel, cloud ... à vous de choisir).

Entrenez-vous bien à manipuler vos fichiers : une perte de données ne sera pas acceptée comme une excuse valable !

Rapport de TP

Ce rapport reprendra la forme du modèle qui vous est fourni dans :
`pedago-iut/INFO/Public/ENSEIGNEMENT/BUT1/R1_01`

Le fichier est nommé `TPXX_NOM_PRENOM_GXX.pdf`

Vous devrez respecter cette convention de nommage, la forme et toutes les consignes présentes dans ce modèle.

Vous êtes libres de la méthode et de l'outils de traitement de texte à utiliser. **Mettez bien à profit cette séance pour vous entraîner et tester différentes solutions :**

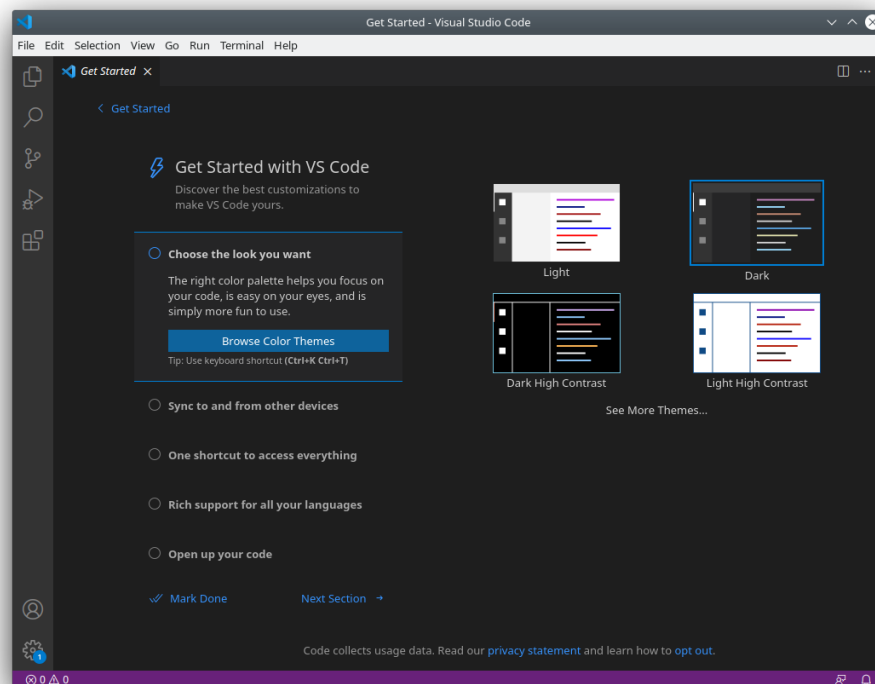
- Comment accéder à vos fichiers depuis n'importe quelle VM (voir documentation fournie dans INFO-ROOT)
- Comment prendre des captures d'écran (partielles selon sélection, par fenêtre, ou entière ?)
- Quelle méthode utiliser pour rédiger votre rapport ?

Exceptionnellement, à la fin de cette première séance, vous devrez remettre à votre encadrant un rapport de TP, en format PDF (placé par exemple dans son casier numérique ou à l'emplacement qu'il vous indiquera). Cela permettra de valider votre travail. Par la suite, ces rapports devront être fait pour vous permettre de faciliter vos révisions et identifier vos problèmes. Ils n'ont pas vocation à être rendu à chaque séance.

Prise en main et configuration de Visual-Studio Code

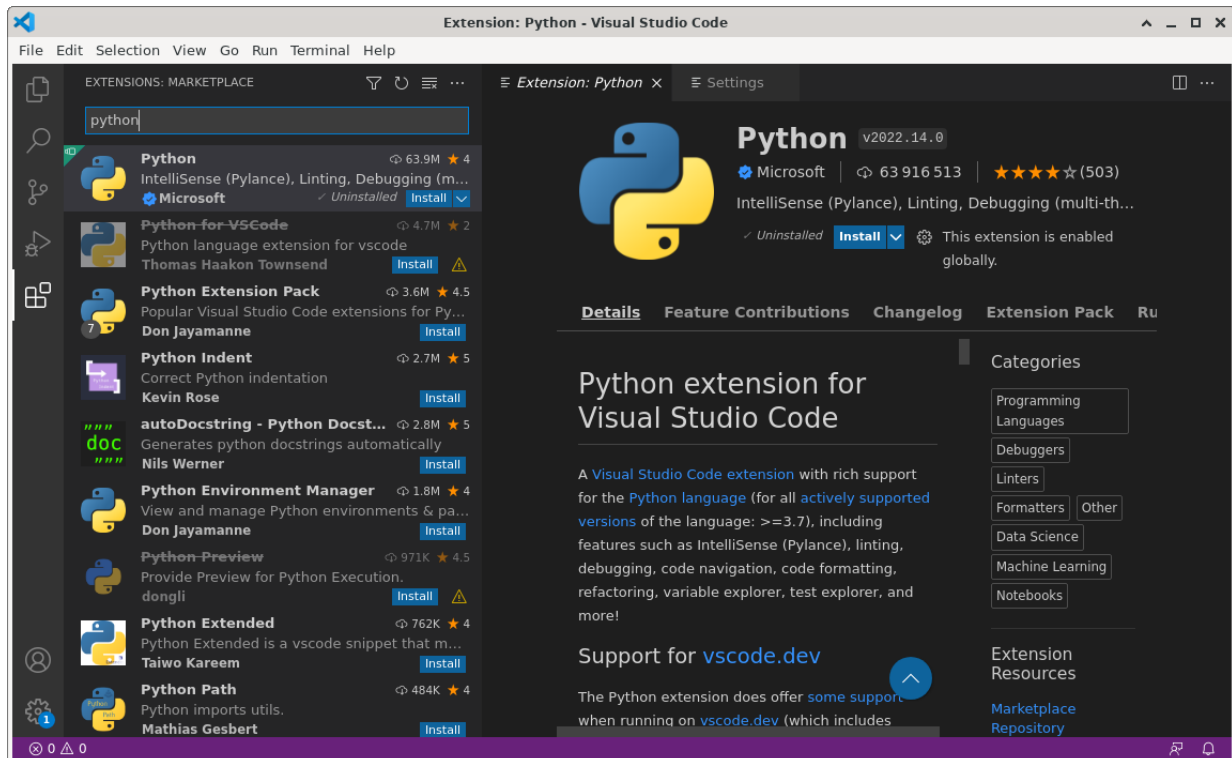
Vous êtes libres dans votre choix d'OS / de VM ...

Lancez VSCode :



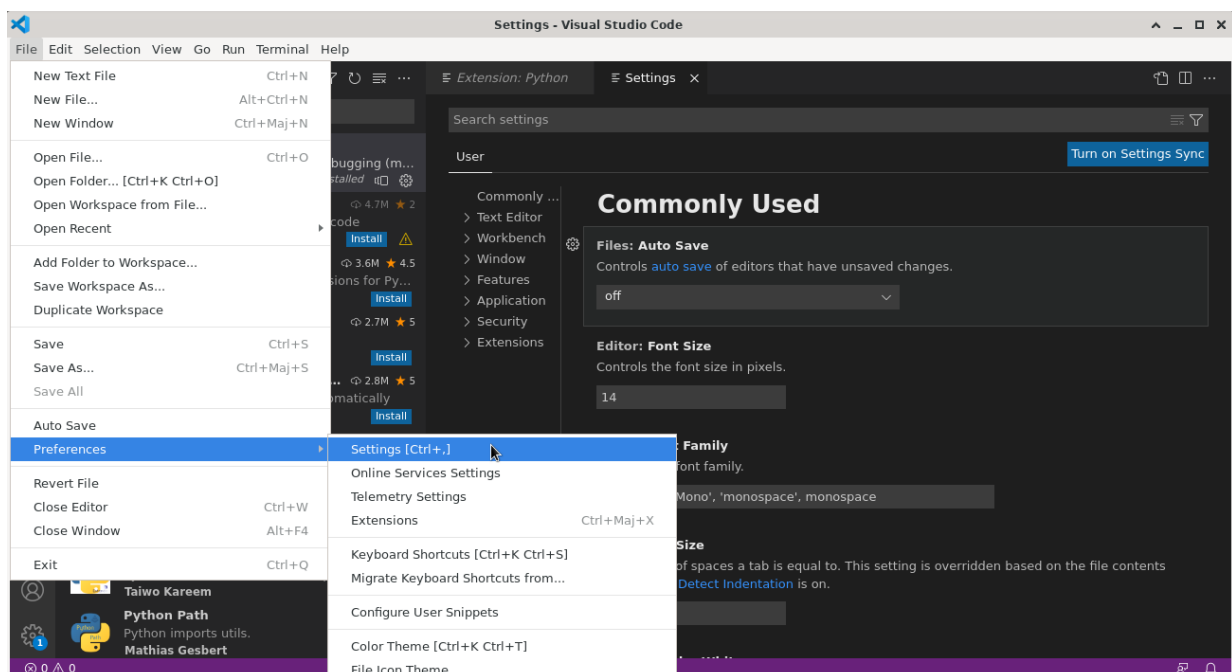
Nous devons maintenant installer et configurer le plugin Python. Il est possible que cette étape soit à recommencer à chaque séance (selon les salles). C'est très rapide et nous permettra d'avoir toujours une version « à jour » du plugin.

Cliquer sur l'icône « Extensions » et entrez python dans la barre de recherche :

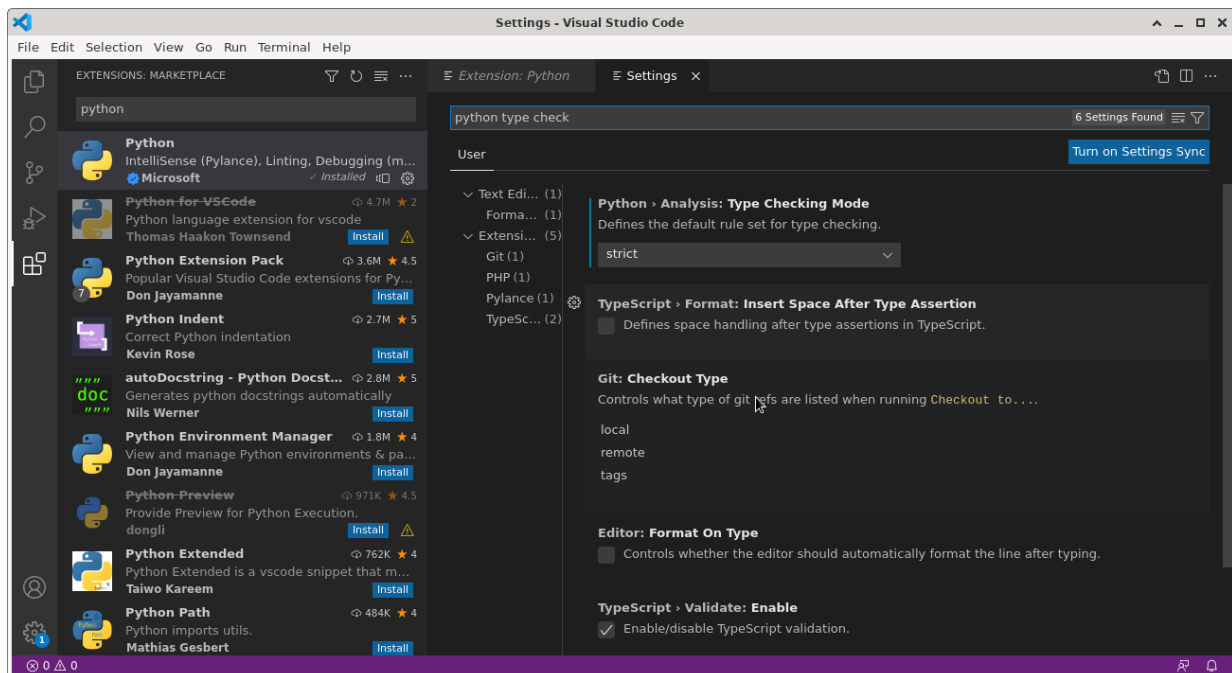


Le plugin à installer (bouton « install ») est présenté sur l'écran ci-dessus.

Il nous faut maintenant le configurer pour permettre la vérification des types :



Dans la barre de recherche des réglages, tapez « python type check » puis réglez cette préférence sur « strict » comme cela est illustré ci-dessous :



Premier programme

Créez un dossier sur le bureau (nommé ALGO par exemple). Dans ce dossier, créez un nouveau dossier nommé TP1. N'hésitez pas à bien vous familiariser avec l'interface utilisateur du système.

Ouvrez ensuite ce dossier dans VSCode. Vous retrouverez votre arborescence dans la barre latérale.

Grace au bouton droit de la souris, vous pouvez aussi créer dossiers et fichiers directement depuis la barre latérale :

- Créez un nouveau fichier dans TP1, que vous nommerez `premier.py`
- Tapez votre premier programme (notez qu'il y a 2 _ avant et après name et main) :

```
if __name__ == "__main__" :  
    print ("Bonjour !")
```

Pour l'exécuter dans VSCode :

- Cliquez sur le triangle « play » en haut à droite : un terminal va s'ouvrir et exécuter votre programme ;
- Notez que la ligne de commande permettant l'exécution de votre programme est affichée dans le terminal. Vous pouvez ré-utiliser cette ligne en dehors de VSCode !

Tentez de générer des erreurs :

- En enlevant le ':' de la première ligne par exemple : que se passe-t-il ? commentez !
- En enlevant le i de print, que se passe-t-il ? Quelle différence entre les messages d'erreur ?
- En remplaçant `__main__` par `_main_`, que se passe-t-il ? Commentez !

Vérifiez la bonne vérification des types, avec par exemple :

```
import math

if __name__ == "__main__" :
    var_ent : int
    chaine : str

    chaine = float(input("entrez un entier :"))
    var_ent = math.sin(0.5)
```

Vous devez obtenir des erreurs (bien) documentées par VSCode.

Pensez à la rédaction de votre rapport, qui devra présenter vos résultats et commentaires précédents (y compris les erreurs que vous venez de générer et d'analyser) dans la partie « jeux d'essai ».

Sauvegardez bien vos fichiers de travail dans votre espace si ce n'est pas déjà fait.

Partie 2 :

Chaque programme sera placé dans un fichier spécifique dans le répertoire TP1. Essayez de refaire ces programmes sans la correction ...

- Testez le programme de calcul de la surface d'un cercle. Ajoutez-lui le calcul et l'affichage du périmètre de ce cercle.
- Programmez et testez tous les programmes vus en cours
 - o Pour chaque programme, votre compte rendu devra présenter un jeu d'essais « bien choisi » permettant d'illustrer son bon fonctionnement (et les bugs éventuels que vous détecterez !)

Votre rapport doit être totalement rédigé et rendu en fin de séance !