La carte NUCLEO-WB55 a été préparée et configurée par votre enseignant.

L’ensemble logiciel nécessaire à la mise en œuvre de la carte a également été installé sur votre ordinateur.

Ci-après un extrait d’une documentation de ST Microlectronics pour la configuration du terminal de communication (Putty) et de l’environnement de programmation Pyscripter.

## Utilisation de l’environnement de programmation

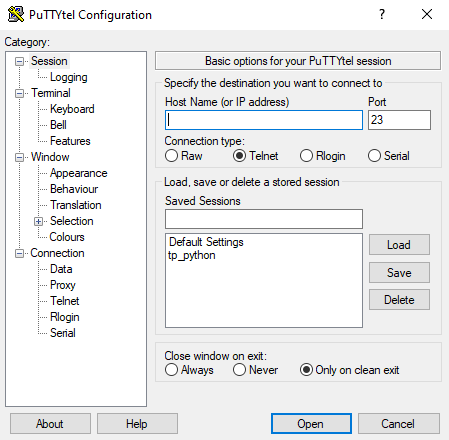
Ces manipulations devront être maîtrisées par les étudiants afin qu'ils puissent communiquer avec l'interpréteur et éditer des scripts MicroPython sur le kit NUCLEO-WB55.

### Configuration du terminal de communication

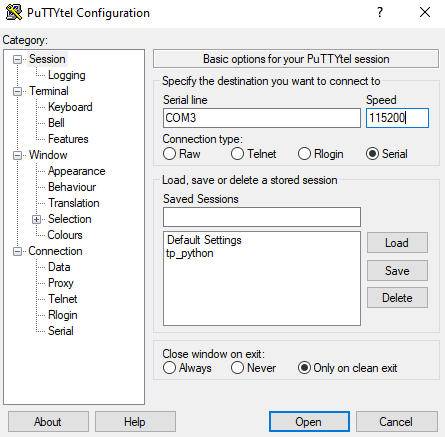
Maintenant que MicroPython est présent sur le kit NUCLEO-WB55, nous voudrions tester la communication avec Python. Le test consiste à envoyer une commande python et vérifier que l'exécution s’effectue.

La communication s’effectue par USB par le biais d’un port série, nous avons donc besoin d’un logiciel capable d’envoyer les commandes python sous forme de texte à la carte et de recevoir le résultat de l'exécution.

* Pour cela, lancez l’utilitaire Puttytel avec le raccourci placé sur le bureau :



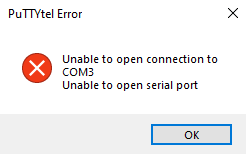
* Sélectionnez l’option “Serial” :



* Configurez les champs “Serial Line” en COM3 et “Speed” à 115200. Puis cliquer sur Open.

*Nota : à ce stade vous pouvez sauvegarder cette configuration en lui donnant un nom, par exemple on voit plus haut un exemple nommé tp\_python. Par la suite il suffira de double-cliquer sur le nom pour ouvrir la communication avec la carte STM32WB55.*

Si, après avoir cliqué sur Open, le message suivant apparaît :

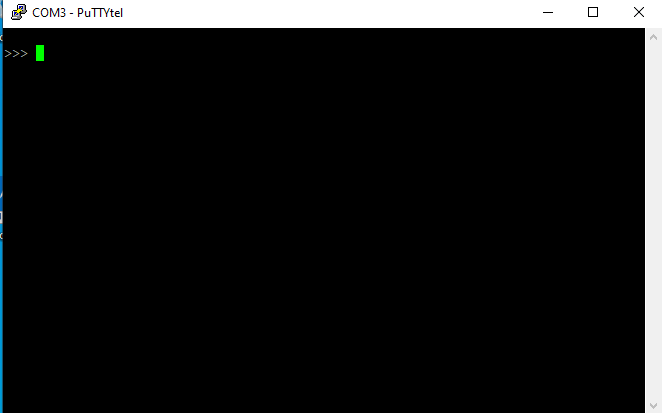


* Vérifiez que le kit est bien énuméré sur le bon port COM.

Sinon, une nouvelle fenêtre s’affiche :



* Appuyez sur CTRL+C pour faire apparaître l'interpréteur de commande python :



* Vous pouvez maintenant exécuter des commandes Python .

##### Exemple : print(“Hello World”)



Nous avons maintenant terminé la programmation du firmware MicroPython.

Notre kit de développement NUCLEO-WB55 est désormais prêt à être utilisé avec Python.

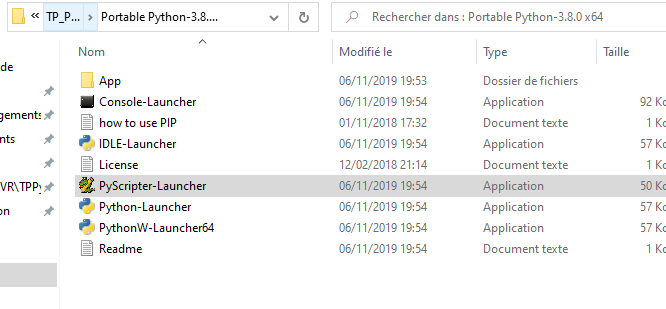
* Gardez la fenêtre Putty ouverte, elle vous sera utile pour voir l'exécution des scripts.

**NOTE** : L'autocompletion (appui sur TAB) est disponible lors de la communication série.

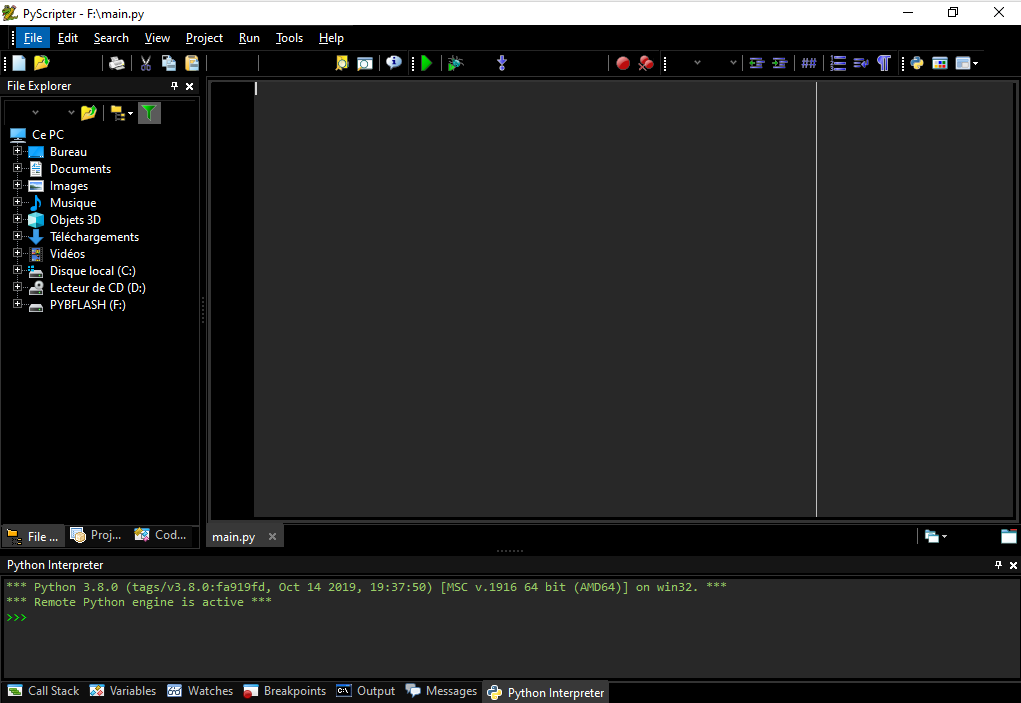
### Utilisation de l’éditeur de script python

Nous allons voir, dans cette partie, comment utiliser un environnement de développement MicoPython. Cet environnement a déjà été installé par votre professeur.

L'environnement disponible est “Portable Python-3.8.0 x64” :



* Lancez l’outil “PyScript-Launcher” :



Cet outil vous permettra de débugger et d'exécuter vos scripts python.

**Remarque importante :**

**La bibliothèque “pyb”, c’est à dire le groupement des fonctions micropython d’accès aux périphériques du microcontrôleur, n’est pas disponible lors du débogage des scripts sous Windows.**

**Attention aussi à n’utiliser que les bibliothèques Python 3 portées sous micropython. Par exemple la bibliothèque “numpy” n’est pas encore implémentée officiellement dans MicroPython, il sera donc impossible d’effectuer un ‘import numpy” dans un script MicroPyhon à vocation d’être téléchargé dans la NUCLEO WB55.**

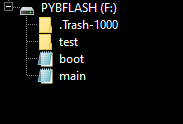
Voici la liste des bibliothèques utilisables avec MicroPython :

<http://docs.micropython.org/en/latest/library/index.html#micropython-specific-libraries>

L’éditeur de script python (PyScript) est donc utile pour le test d’algorithme ou la complétion automatique des méthodes python.

#### Ecrire et exécuter un script MicroPython

* Revenez à l’éditeur de script PyScript-Launcher, puis développez le lecteur PYBFLASH :



Nous voyons ici 2 fichiers intéressants, boot.py et main.py.

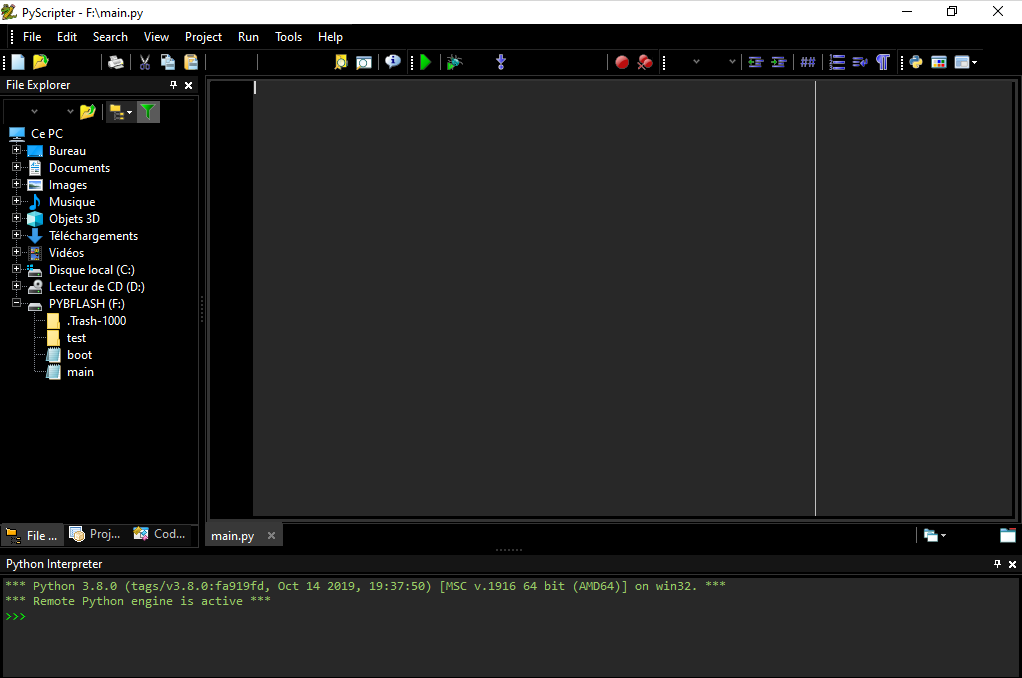
Le script boot.py, à modifier par les utilisateurs avancés, permet d’initialiser MicroPython.

Il permet notamment de choisir quel script sera exécuté après le démarrage de MicroPython, par défaut le script main.py.

Laissez la configuration initiale pour le moment.

Nous allons nous intéresser au script main.py.

* Double-cliquez sur celui-ci :



Exécuter le script

Déboguer le script

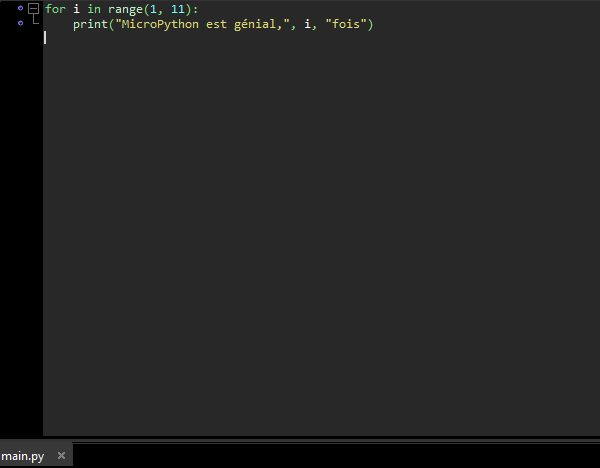
Ajouter des points d’arrêts

Script en cours d’édition

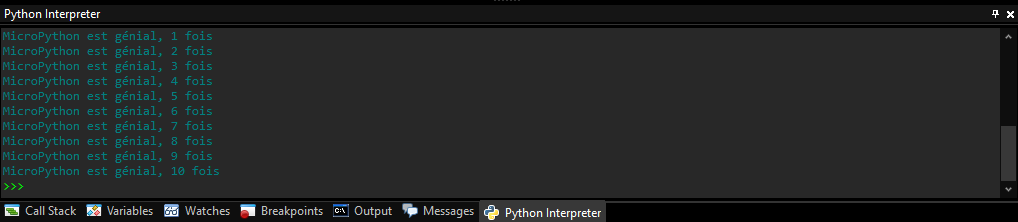
Console Python

Notre premier script va consister à afficher 10 fois le message “MicroPython est génial” avec le numéro du message.

* Ecrivez l’algorithme suivant dans l’éditeur de script :



* Exécutez et vérifiez le résultat de celui-ci en appuyant sur  ou CTRL+F9 :

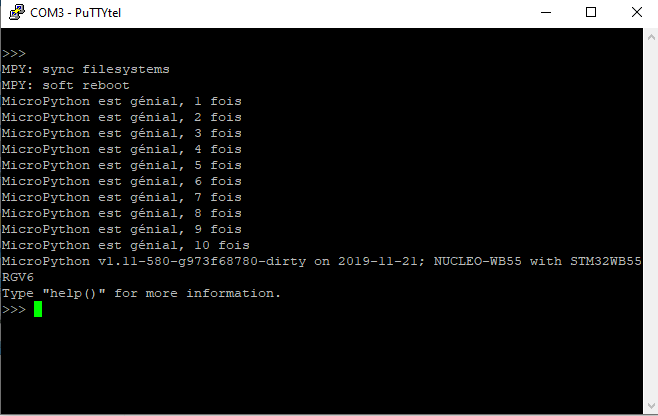


Le script fonctionne bien sous Windows, nous voulons maintenant l'exécuter sur la carte NUCLEO-WB55.

* Pour cela sauvegardez le script main.py (CTRL+S), puis revenir au terminal Putty.

Le script sera exécuté lors d’un redémarrage de MicroPython.

* Pour effectuer un redémarrage appuyez sur CTRL+C (pour afficher l'interpréteur “>>>”) puis CTRL+D (pour forcer le redémarrage) dans le terminal Putty :



Notre script s’est exécuté avec succès ! Nous voyons bien notre message s’afficher 10 fois dans le terminal.

Nous sommes désormais prêts à effectuer des commandes MicroPython nous permettant d’accéder aux périphériques du microcontrôleur STM32WB55.