

Tutoriel pour travailler de chez soi

Ce petit tutoriel indique comment travailler de chez soi dans des conditions similaires aux TPs, pour Python ainsi que pour le shell. Il donne aussi une méthode basique permettant de récupérer les fichiers se trouvant sur le répertoire personnel du compte Université (sous Linux).

1 JupyterHub

Le service

<https://jupyterhub.ijclab.in2p3.fr/>

utilisé dans le cours d'introduction à la programmation impérative peut servir pour travailler à distance. Attention cependant :

- pour les TPs de Python, il ne faut pas choisir **Python** dans le menu **New** car cela crée un *notebook*. Il faut choisir **Text File** puis renommer le fichier créé pour lui donner un nom terminant par **.py**. On peut ensuite faire **New** puis **Terminal** et utiliser le terminal pour lancer le programme python avec la commande `python3 mon_fichier.py`
- pour les TPs de Shell, ouvrir un terminal avec **New** et **Terminal**. Seul les exercices nécessitant de lancer des applications graphiques ne fonctionneront pas.

Attention, les fichiers créés sur le compte JupyterHub ne sont pas directement visible dans vos comptes des salles machines et inversement. Il convient donc de les récupérer et les déposer au bons endroits.

2 Python

2.1 Sous Linux

Il suffit d'installer Python (version 3) sur votre distribution, en utilisant le système de paquets de cette dernière. Il n'est pas recommandé de passer par les sources ou par des dépôts externes. L'éditeur **geany** est normalement installable de la même façon. Une fois ce dernier installé, il faut installer aussi l'utilitaire **pycodestyle**. Il est ensuite possible de récupérer le fichier `filetypes.python` sur la page du cours. Il faut placer ce fichier dans le répertoire :

```
~/config/geany/filedefs
```

Que l'on aura précédemment créé avec la commande :

```
mkdir -p ~/config/geany/filedefs
```

Rappel sous Linux, les repertoires dont le nom commence par « . » sont « cachés » par défaut. Il est donc possible que le répertoire « `.config` » ne s'affiche pas dans votre interface graphique.

2.2 Sous Windows ou MacOS X

Sur ces deux plateformes, il est recommandé de télécharger Python depuis le site officiel :

<https://www.python.org>

Et de suivre les instructions d'installation. Après l'installation, vous disposerez en plus de l'interprète Python de l'interface graphique IDLE. Cette dernière propose :

- Un interprète Python exécutable dans une fenêtre graphique (sans terminal)
- Un éditeur de code Python intégré, permettant d'éditer des fichiers **.py**.

L'éditeur IDLE permet en particulier d'exécuter automatiquement le contenu d'un fichier en utilisant la touche F5, comme **geany**.

3 Shell

3.1 Linux et MacOS X

Sous Linux ou MacOS X, l'application **Terminal** permet de lancer un shell. Les commandes utilisées en TP (`mkdir`, `cat`, ...) de même que les motifs *globs* sont standard et devraient fonctionner sans problème.

3.2 Windows, solution légère

Sous Windows, une solution simple pour avoir un *shell* Unix est d'utiliser un programme permettant d'ouvrir un *shell* distant sur les machines de l'Université. Ce mode de fonctionnement requiert **une connexion Internet** pendant l'utilisation du programme.

Vous pouvez télécharger le programme PuTTY :

<https://www.putty.org/>

Dans la section **Download : Stable**, il est conseillé de télécharger simplement le binaire `putty.exe` pour la version de Windows que vous possédez. Vous pouvez alors lancer le programme sans installation.

Lors du lancement, dans le champ **Host**, il faut saisir l'adresse :

`tp-ssh1.pgip.universite-paris-saclay.fr`

Attention, ne pas mettre de `https://` devant. Optionnellement, dans la section *Appearance*, vous pouvez changer la taille de la police utilisée pour le terminal (plus grande par exemple).

Vous pouvez ensuite vous connecter. Lorsque le programme vous les demande, vous pouvez saisir vos identifiants de connexion des salles machines (login et mot de passe). Vous serez alors connecté dans un *shell distant* s'exécutant sur une machine d'enseignement. Le répertoire personnel sera le même que celui utilisé en TP.

3.3 Windows, solutions lourdes

Si vous voulez installer un shell Unix sur votre machine Windows, il y a deux alternatives, toutes deux relativement lourdes (espace disque, installation, ...). Elles sont réservées aux étudiant·e·s intéressés ou à ceux ne disposant pas d'une connexion Internet fiable. Elles sont déconseillées dans le cadre du cours d'Introduction à l'Informatique, les quelques commandes *shell* étudiées pouvant être facilement faites avec la solution légère ci-dessus.

3.3.1 CygWin

L'environnement `cygwin` est une collection d'utilitaires émulant un système Unix sous windows. L'installation de base inclut en particulier un *shell* Unix (chemins avec des `/`, permissions, ...).

<https://www.cygwin.com/>

Le site contient un guide d'installation. L'installation par défaut permet largement de faire les TPs Unix du cours d'Intro à l'Informatique. Il est déconseillé d'installer Python par ce biais, surtout s'il a déjà été installé avec l'installateur officiel (section précédente).

3.3.2 Windows Subsystem for Linux

Cette solution est disponible uniquement sur les versions récentes de Windows 10. Elle permet essentiellement d'installer une distribution Linux dans une machine virtuelle légère (*i.e.* un logiciel qui simule un vrai PC dans lequel est exécuté le système Linux invité).

<https://docs.microsoft.com/en-us/windows/wsl/install-win10>

Comme pour la solution précédente, celle ci est relativement lourde et comprend un certain nombre d'étapes d'installation et de configuration.

4 Récupération de fichiers

Il est possible de récupérer les fichiers de son répertoire personnel à distance, ou d'en déposer sur son compte. On peut utiliser pour cela l'utilitaire **FileZilla**. Sous Linux il est recommandé de l'installer depuis les paquets de sa distribution. Sous Windows ou MacOS X, le logiciel peut être téléchargé depuis le site officiel :

`https://filezilla-project.org/`

Il faut télécharger le client **FileZilla Client** pour votre système.

Lors de la connexion, utiliser les paramètres suivants :

Hôte/Host : `tp-ssh1.pgip.universite-paris-saclay.fr.fr`

Username/Identifiant : votre identifiant dans les salles machines (comme votre mail sans le `@universite-paris-saclay.fr`)

Password/Mot de passe : le mot de passe de votre compte

Port : 22

Lors de la première connexion FileZilla vous demande s'il doit stocker votre mot de passe (choisissez ce que vous voulez). Il se peut aussi que vous ayez le message :

La clé du serveur hôte est inconnue. ...

Approuver ce serveur et l'associer à la connexion ?

Cocher la case **Toujours faire confiance à cet hôte**, ... et cliquer sur OK.

Vous obtiendrez deux panneaux : celui de gauche avec les fichiers de votre ordinateur personnel, celui de droite avec les fichiers de votre compte informatique utilisé en TP. Vous pouvez copier les fichiers entre les deux panneaux.