

1 Environnements

1.1 Pycharm

Pycharm est un environnement de développement intégré (IDE en anglais, pour *integrated development environment*) pour la programmation en *Python*. Il est gratuit sous sa forme *community edition* et payant (sauf sous certaines conditions, comme être étudiant) sous sa forme *professional edition*. Ce logiciel est facile à utiliser grâce notamment à son interface ergonomique. Il contient les meilleurs éléments des IDEs modernes avec la complétion de code, la coloration des mots clés, l'analyse du code en temps réel pour trouver certaines erreurs de syntaxe, un débogueur, un profileur, etc.

1.1.1 Installation

L'installation de PyCharm se fait assez facilement. Il s'uffit en premier temps d'aller sur le site de PyCharm et d'installer la version *community* ou *professional*. Ces versions sont disponibles au lien suivant : <https://www.jetbrains.com/pycharm/download/#section=windows>. Une fois l'installation lancée, nous recommandons d'activer les options

- ☒ Create Desktop Shortcut/64-bit launcher ;
- ☒ Update context menu/Add "Open Folder as Project" ;
- ☒ Create Associations/.py ;

puisque celles-ci facilitent grandement l'ouverture de projet dans l'explorateur de fichiers.

1.1.2 Création d'un nouveau projet

Il existe essentiellement deux façons de créer un projet avec Pycharm. La première consiste à cliquer sur "new project" à l'ouverture de l'application ou dans le menu "File" si l'application est déjà ouverte sur un autre projet. Une fois fait, il faut entrer les paramètres du nouveau projet. Avec l'aide de la figure 1.1, on voit qu'il faut préalablement indiquer la localisation du projet ainsi que le nom de celui-ci. Il est recommandé de localiser ce nouveau projet à l'intérieur du dossier github associé à ce même projet. Pour ce qui est de l'environnement virtuel, celui-ci permet d'avoir un interpréteur propre au projet. En effet, différents projets vont demander différents pré-requis en termes de version de python ou de version de différents modules. Il est donc important d'avoir un environnement virtuel différent pour chacun des projets. Pour des projets plus avancés, il est même requis de faire plusieurs environnements virtuels pour un seul projet. Bref, pour cet environnement virtuel, il faut lui donner un nom qui est normalement tout simplement "venv". Finalement, il faut choisir la version de python qui sera utilisée dans le projet. Si c'est le premier projet et qu'aucun interpréteur python n'a encore été installé, il faut préalablement à cette étape aller installer la version de python voulue à <https://www.python.org/downloads/>.

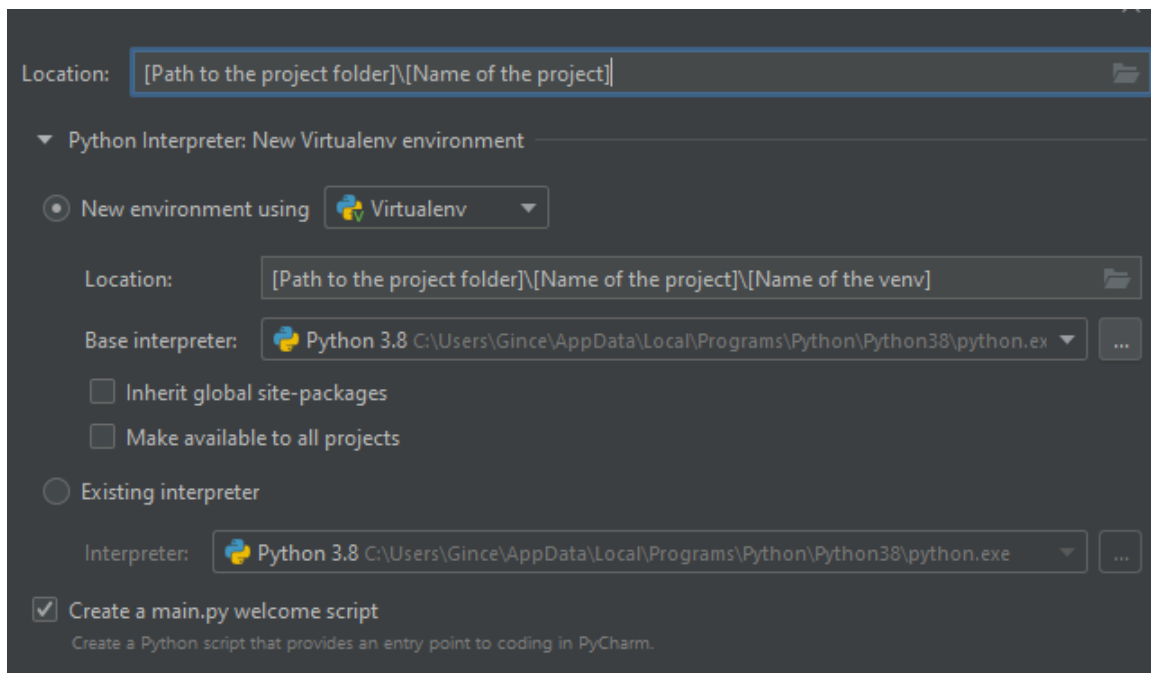


FIGURE 1.1: Fenêtre de création de projet avec PyCharm.

La seconde façon de créer un projet est la suivante. Il faut préalablement créer un dossier contenant préféablement les fichiers de github associé au projet. Ensuite, il faut faire un click droit sur le dossier et appuyer sur "Open Folder as a PyCharm Project". Si cette dernière option n'est pas disponible c'est sûrement parce que l'option *Update context menu/Add "Open Folder as Project"* n'a pas été coché à l'installation de l'application ou bien que l'ordinateur a besoin d'être redémarré. Dans le cas, ou ce n'est pas possible d'ouvrir le projet de cette façon, il suffit d'ouvrir l'application et aller dans le menu "File/Open" afin d'arriver au même résultat. Par la suite, afin d'ajuster les paramètres de l'interpréteur et de créer l'environnement virtuel, il faut aller dans le menu "File/Settings/Project :Python/Python Interpreter". Il faut ensuite, appuyer sur l'icône de paramètres à droite de "Python Interpreter" et appuyer sur "Add". Finalement, il faut créer l'environnement virtuel en allant dans le sous menu "Virtual Environment". Il sera donc possible d'entrer les mêmes valeurs qu'indiqué à la figure 1.1.

1.1.3 Création d'un environnement virtuel

Dans plusieurs cas, vous aurez un projet déjà existant et vous devrez vous préparer à l'utiliser. La première chose à faire est d'ouvrir le dossier content les fichiers du projet avec PyCharm. Pour ce faire, si vous avez activé l'option *Update context menu/Add "Open Folder as Project"* lors de l'installation de PyCharm, vous pouvez faire click droit sur le dossier directement et cliquez ensuite sur "Open Folder as Project". Si vous n'avez pas cette option, vous pouvez ouvrir PyCharm et ensuite ouvrir un nouveau projet en sélectionnant ce dossier. Vous avez maintenant le projet ouvert avec PyCharm.

Il est maintenant temps de créer votre environnement virtuel. Il y a plusieurs façons d'y arriver, la façon PyCharm et la façon en lignes de commandes. Pour la façon PyCharm, c'est très simple. Il suffit d'aller dans les *settings* du projet, ensuite aller dans *Project :[Nom du projet]/Python Interpreter*. À ce moment vous verrez que vous pouvez choisir l'interpréteur de votre projet, mais

vous voulez installer un nouvel environnement virtuel, alors vous aller cliquer sur *Add*. À ce moment, vous aller voir une interface très semblable à la figure 1.1, remplissez les entrées comme décrit dans cette section. Vous avez maintenant un nouvel environnement virtuel, il est temps d’aller l’activer. Pour ce faire, vous aller dans *Terminal* et vous ajouté un nouvel onglet nommé *Command prompt* ce qui va ouvrir un terminal avec votre environnement virtuel activé. Pour vous assurer que cela a bien été fait, vous allez voir ”([Nom de votre environnement virtuel])” en avant de votre *path* courant. Si vous n’arrivez pas à suivre les étapes d’activations de votre environnement virtuel, veuillez suivre les étapes en terminal. Une fois activer, vous allez vouloir installer les ”requirements” du projet, s’il y en a. pour ce faire, vous allez dans votre *Command prompt* et vous lancer la commande *pip install -r [path]/requirements.txt*. De même façon, si vous voulez installer de nouveaux modules python, vous lancer la commande *pip install [nom du module]*.

Pour la version terminale, vous commencer par ouvrir votre projet comme mentionner précédemment. Ensuite, vous ouvrez le terminal et vous lancer la commande suivante : *python -m venv [path du nouvel environnement]/[nom du nouvel environnement virtuel]* ou bien *[path vers le python que l’on veut ”copier”]/python -m venv [path du nouvel environnement]/[nom du nouvel environnement virtuel]*. Ensuite, pour l’activer, vous lancez la commande *[path du venv]/Scripts/activate*. Une fois activer, vous allez vouloir installer les ”requirements” du projet, s’il y en a. pour ce faire, vous lancer la commande *pip install -r [path]/requirements.txt*. De mêmes façons, si vous voulez installer de nouveaux modules python, vous lancer la commande *pip install [nom du module]*.

1.2 Anaconda

Anaconda est non seulement un interpréteur pour permettre à l’ordinateur d’exécuter le code écrit, mais c’est aussi un *package* contenant une foule d’outils très utiles pour les scientifiques. La distribution *Anaconda* inclue notamment *Jupyter* et un interface convivial pour gérer les modules installés ainsi que les environnement virtuels (*conda*). Il est vraiment facile à installer et utiliser, soit en ligne de commande ou par *Anaconda Navigator*, soit par son interface graphique. Pour plus d’informations ou pour installation, visitez le site web d’*Anaconda*.