

.Utilisation de Pycharm

Paramétrer et utiliser Pycharm (illustrations réalisées sur Pycharm 2022.1.3)

PyCharm est un environnement de développement intégré (IDE - Integrated development environment) utilisé pour programmer en Python.

Il permet l'analyse de code et contient un débogueur graphique.

Développé par l'entreprise tchèque *JetBrains*, c'est un logiciel multi-plateforme qui fonctionne sous Windows, Mac OS X et Linux.

PyCharm inclut toute une liste d'outils qui transforment *PyCharm* en un des meilleurs environnements de programmation pour Python mais qui en plus facilite le travail des développeurs. *PyCharm* offre la saisie de code intelligente, des inspections de code, la mise en évidence des erreurs à la volée et des correctifs rapides, en plus de riches capacités de navigation.

Ce tutoriel présente une procédure pour installer *Python* et *Pycharm*, quelques paramètres basiques permettant de travailler de façon efficace avec *Pycharm* et un court tutoriel sur la façon d'utiliser cet outil. Malheureusement, ce tutoriel est basé sur l'environnement Windows. Nous n'avons pas encore de version pour *MacOs*.

Table des matières

Installation de <i>Anaconda</i>	3
Téléchargement de Anaconda.....	3
Installation de Anaconda.....	Erreur ! Signet non défini.
Téléchargement et Installation de <i>Pycharm</i>	3
Téléchargement de <i>Pycharm</i>	5
Paramétrage de <i>Pycharm</i>	6
Paramétrage au niveau global	6
Paramétrage au niveau d'un projet.....	9
Utilisations de base de <i>Pycharm</i>	10
Création d'un projet	10
Ouverture d'un projet	12
Création d'un nouveau script.....	13
Ouverture d'un script existant	14

Installation de *Anaconda*

Pour installer *Python*, nous conseillons d'utiliser *Anaconda* cela évitera la difficulté d'installer un environnement *Python* complet.

Téléchargement de *Anaconda*

Anaconda est une plateforme d'environnement et de distribution des langages de programmation R et Python. Cette plateforme est utilisée pour le calcul scientifique, l'analyse prédictive et le traitement des données à grande échelle. Anaconda existe en version gratuite.

A son installation, Anaconda installe Python ainsi qu'une multitude de packages (numpy, spyder, PyQt, etc.).

La procédure que nous recommandons pour installer *Anaconda* sur son ordinateur personnel est la suivante :

1. supprimer toute version antérieure de *Python* sur son ordinateur
2. télécharger Anaconda sur le site : <https://www.anaconda.com/download/>
Vous pouvez ne pas saisir votre adresse mail en cochant l'option « Skip registration »

3. Vous obtenez un fichier exécutable qu'il suffit de lancer

Provide email to download Distribution

Email Address:

☐ Agree to receive communication from Anaconda regarding relevant content, products, and services. I understand that I can revoke this consent [here](#) at any time.

By continuing, I agree to Anaconda's [Privacy Policy](#) and [Terms of Service](#).

Submit >

Skip registration

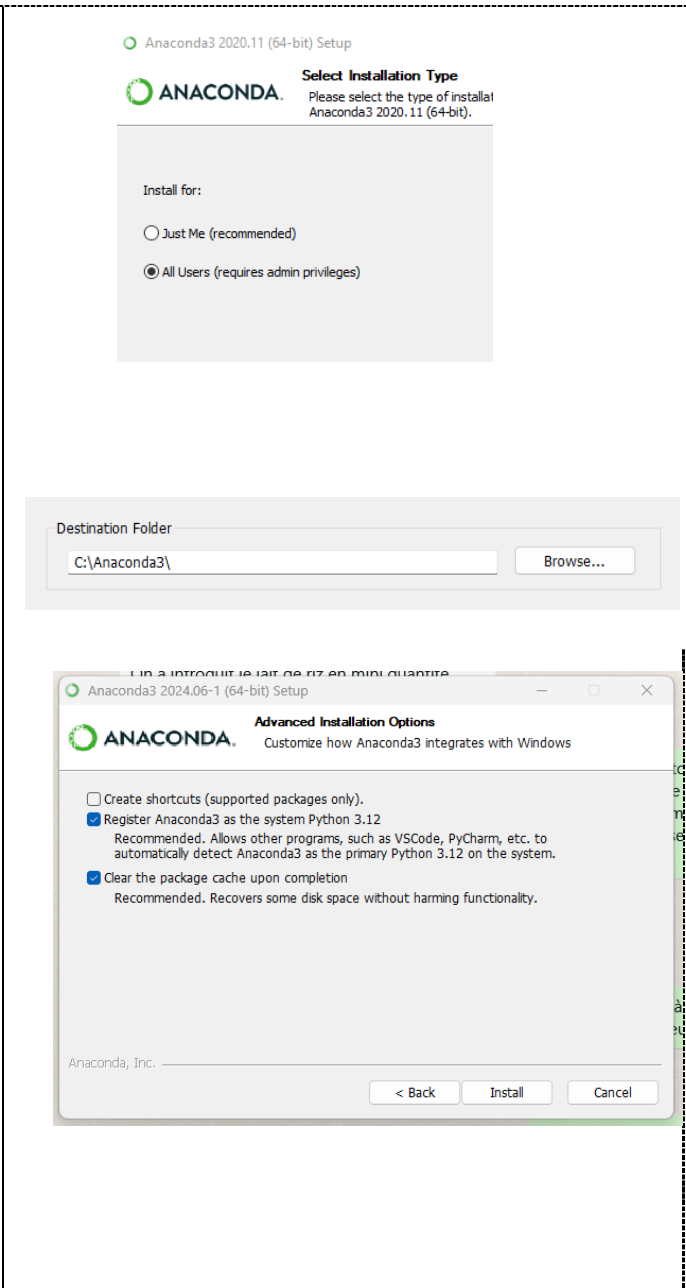
: Anaconda3-2023.07-2-Windows-x86_64.exe

4. Gardez les options par défaut et demander de préférence à ce qu'il soit installé **pour tous les utilisateurs**.

5. l'installer par exemple directement sous C:\Anaconda3
et **surtout pas sur C:\ProgramData** comme il pourrait le proposer.

- Lors de l'installation de Anaconda, laisser les options recommandées par défaut :

- L'installation stagne un peu à un moment donné, mais inutile de s'inquiéter, elle se poursuit un peu plus tard.



Téléchargement et Installation de *Pycharm*

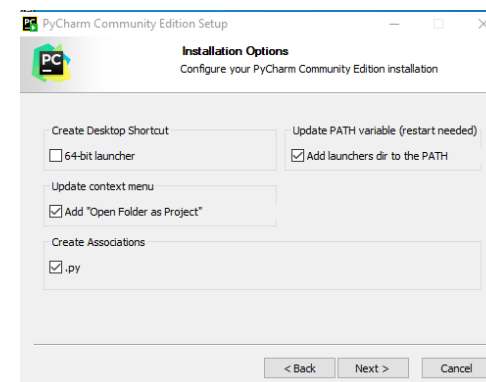
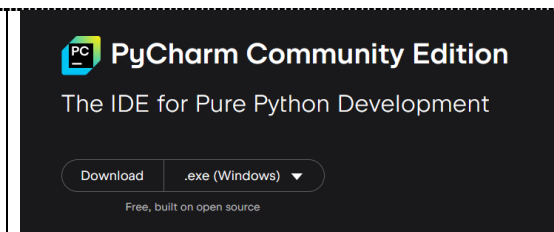
Téléchargement de *Pycharm*

1. télécharger **Pycharm Community** : <https://www.jetbrains.com/pycharm/>

Attention : descendre en bas de l'écran pour bien télécharger *Pycharm Community Edition* et non *Pycharm Professional*

2. lancer l'installation de *Pycharm*

3. cocher les options indiquées dans la capture à droite



Dans les sections suivantes nous expliquons comment définir l'interpréteur *Python* de *Anaconda* en tant qu'interpréteur par défaut et nous décrivons quelques paramètres de base intéressants à mettre en place sur *Pycharm*.

Paramétrage de *Pycharm*

La documentation complète est disponible sur le site de *Pycharm* :

<https://www.jetbrains.com/help/pycharm/guided-tour-around-the-user-interface.html>

<https://www.jetbrains.com/help/pycharm/basic-tutorials.html>

Les préférences, que nous détaillons plus loin, peuvent être configurées à deux niveaux :

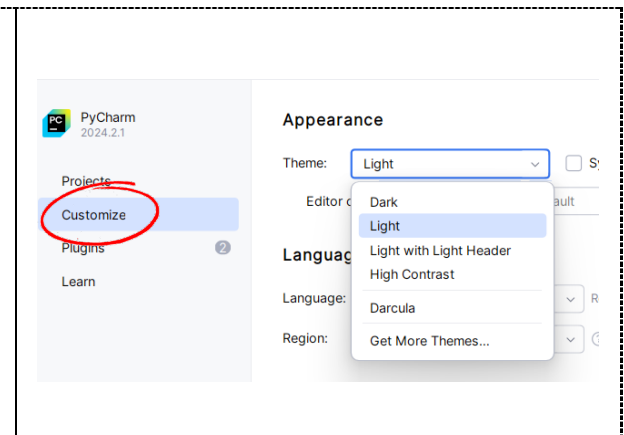
- Au niveau d'un projet (dossier de scripts *Python*)
- Au niveau global applicables à tous les nouveaux projets

Nous vous proposons dans les sections suivantes quelques paramétrages simples de *Pycharm* permettant de faciliter votre travail par la suite. Vous trouverez une documentation complète sur le site de *Pycharm* : <https://www.jetbrains.com/help/pycharm/configuring-project-and-ide-settings.html>

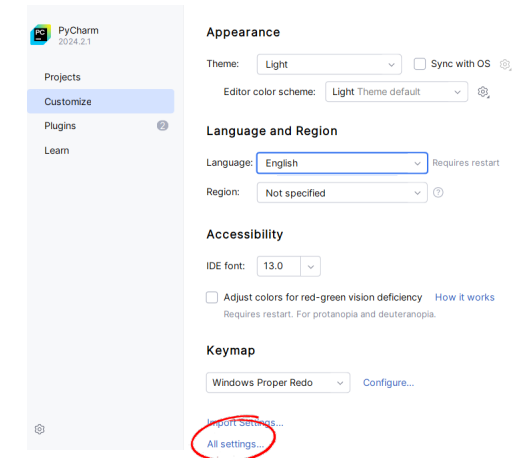
Au moment du 1^{er} lancement de *Pycharm* vous pouvez télécharger le package avec la langue Française, sinon on verra comment l'ajouter ultérieurement.

Paramétrage au niveau global

- Les paramètres peuvent être définis dès le démarrage de *Pycharm* quand il n'y a pas de projets ouverts, à partir de l'onglet *Customize*
- Vous pouvez en premier lieu choisir le thème d'affichage

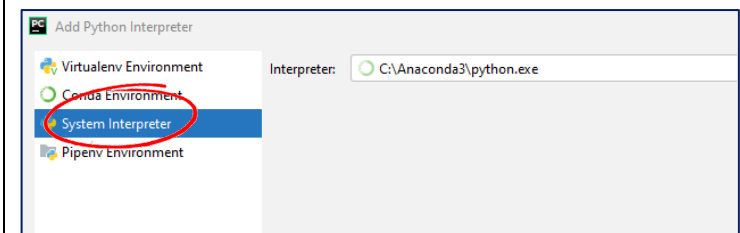
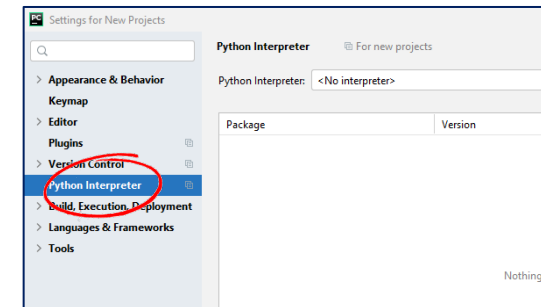


- Les autres paramètres se définissent en cliquant sur **All Settings**



Définition de l'interpréteur

- Menu *Python Interpreter*
 - Aller dans le champ *Python Interpreter*
 - Si l'installation de *Anaconda* s'est déroulée correctement, son interpréteur devrait apparaître automatiquement. Sinon :
 - Cliquer sur la roue crantée sur *Add Interpreter / Add Local Interpreter*
 - Sélectionner **System Interpreter** (et surtout pas *Virtualenv Environment*)
 - Ouvrir la fenêtre de navigation à partir du bouton : à partir de  et sélectionner l'exécutable *python.exe* dans le dossier *Anaconda*
Par exemple : *C:\Anaconda\python.exe*
- On peut de la même façon configurer l'interpréteur pour la console :
 - *Build, Execution, Development / Console / Python Console*



Entête des scripts

Pour que tout nouveau script python débute par un entête de la forme suivante :

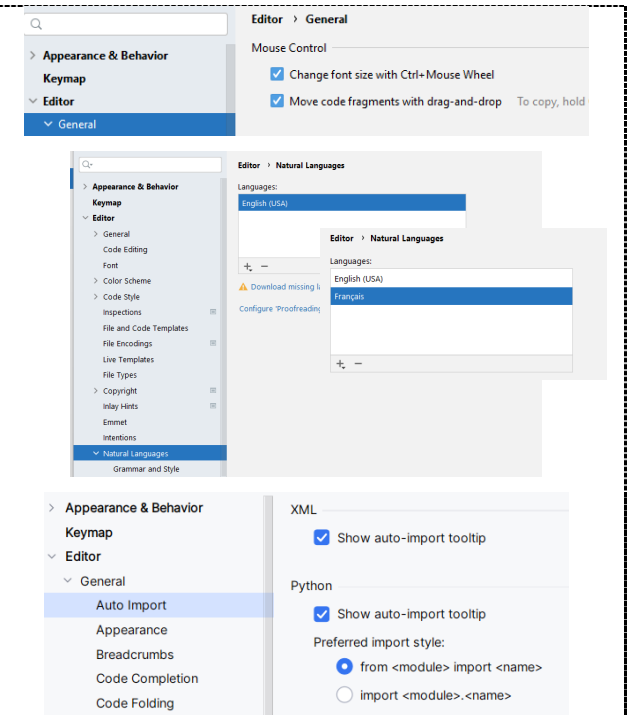
```
# -*- coding:utf-8 -*-  
  
__projet__ = "mon_projet"  
__nom_fichier__ = "mon_script"  
__author__ = "Prénom Nom"  
__date__ = "date de création"
```

```
# -*- coding:utf-8 -*-  
  
__projet__ = "${PROJECT_NAME}"  
__nom_fichier__ = "${NAME}"  
__author__ = "mon_Prénom mon_Nom"  
__date__ = "${MONTH_NAME_FULL} ${YEAR}"
```

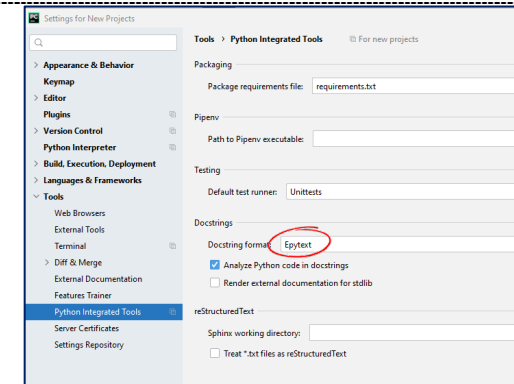
Dans *Editor / File and Code Templates / Python Script* taper le texte à droite :

Autres paramétrages divers

- **Zoomer** avec la molette de la souris
 - Dans le menu *Editor / General* / cocher la case « *Change font size with Ctrl-Mouse...* »
- Intégrer un **dictionnaire français**
 - Dans le menu *Editor / Natural Languages / Spelling*
 - A l'aide du bouton "+", ajouter le français comme langue utilisée dans l'éditeur.
- Dans *General / Auto Import* autoriser l'auto import en cochant l'option *Show auto-import tooltip*



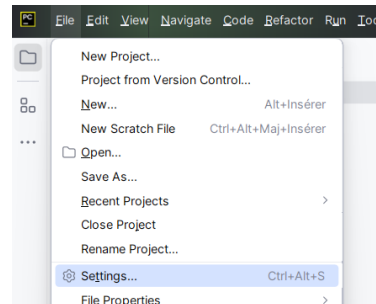
- Dans *Tools / Python Integrated Tools* adopter la syntaxe *Epytext* pour les docstrings



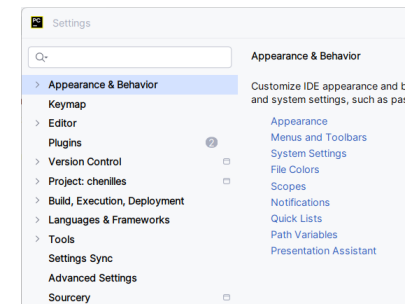
Paramétrage au niveau d'un projet

Les paramètres peuvent être également définis au niveau d'un projet.

- dans *File / Settings* on retrouve les paramètres que l'on souhaite définir au niveau du projet ouvert



- au moment de la création d'un projet on peut également définir l'interpréteur utilisé (voir \$ ci-dessous *Création d'un projet*)

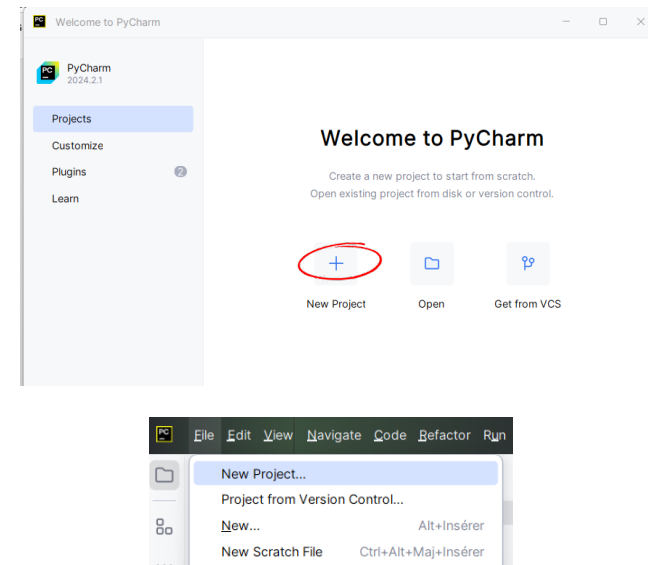


Utilisations de base de *Pycharm*

Création d'un projet

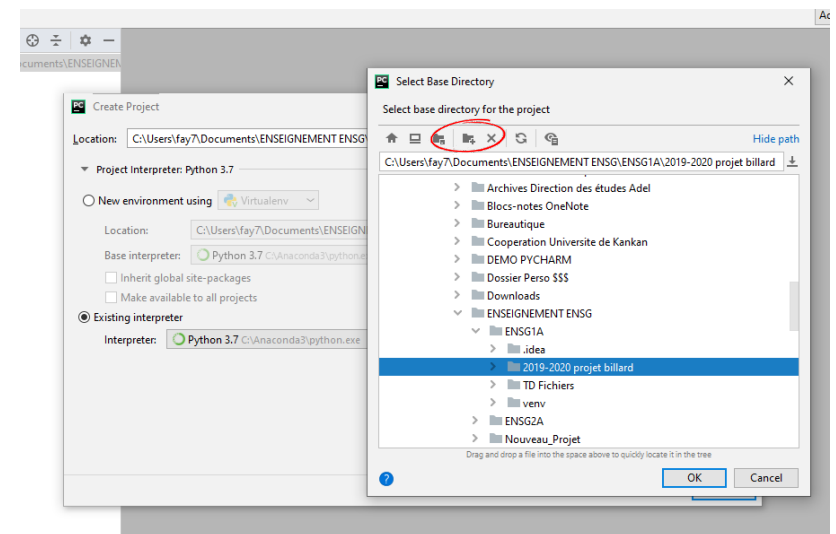
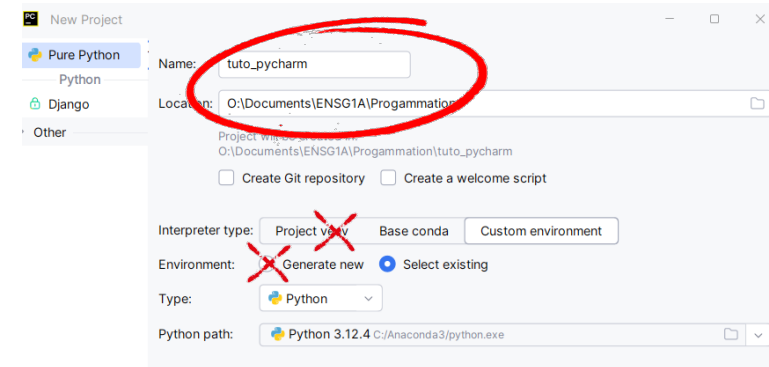
Chaque projet sur lequel vous allez travailler correspond à un dossier. Ce dossier peut être créé au moment de la création du projet.

- On peut créer un projet au moment où on ouvre *Pycharm* quand il n'y a pas encore de projet défini
- On peut également créer un projet plus tard par le menu : *File / New Project*



Les étapes à suivre ensuite lors de la création d'un projet sont les suivantes :

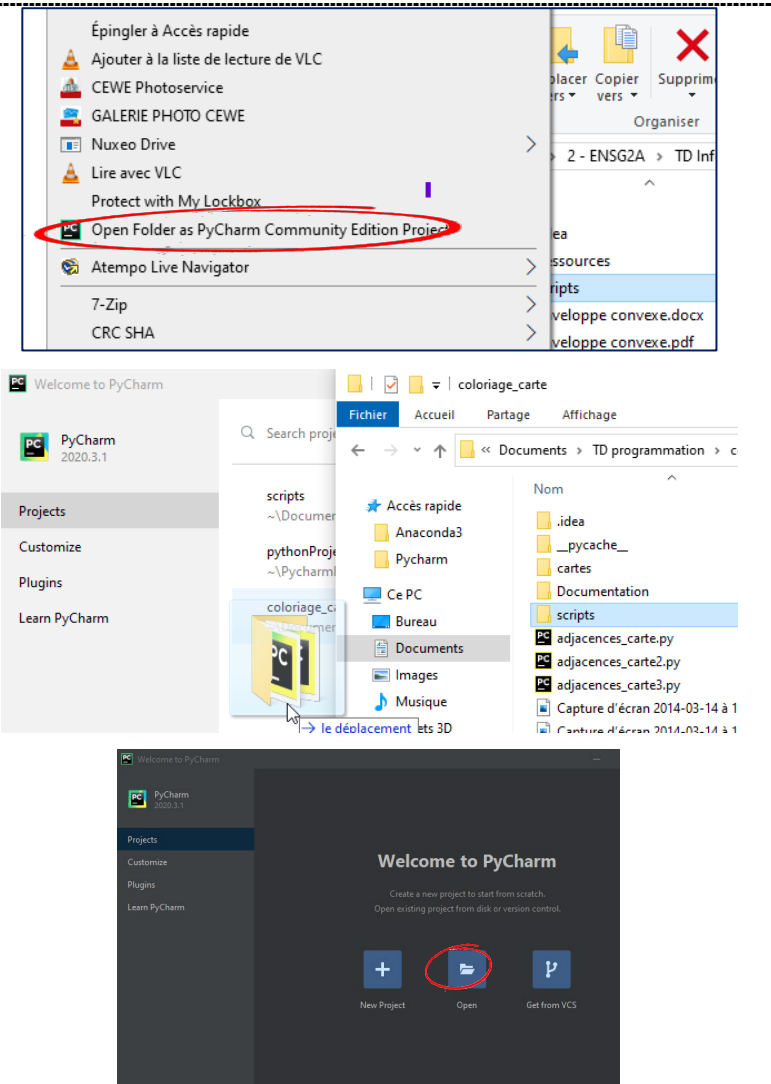
- Sélectionner le dossier et donnez un nom au projet.
 - Sur les postes de l'école, pensez à bien créer votre projet sur O : et non sur C :
 - Sélectionnez *Custom environment / Select existing*
 - Etre vigilant à **ne pas utiliser d'environnement virtuel**.
 - Les autres paramètres définis au niveau global sont automatiquement appliqués au projet. Vous pouvez les modifier par la suite.
-
- Avec l'explorateur de Pycharm, vous pouvez réaliser au cours de cette étape les opérations classiques de manipulation de dossiers : création, suppression, renommage...



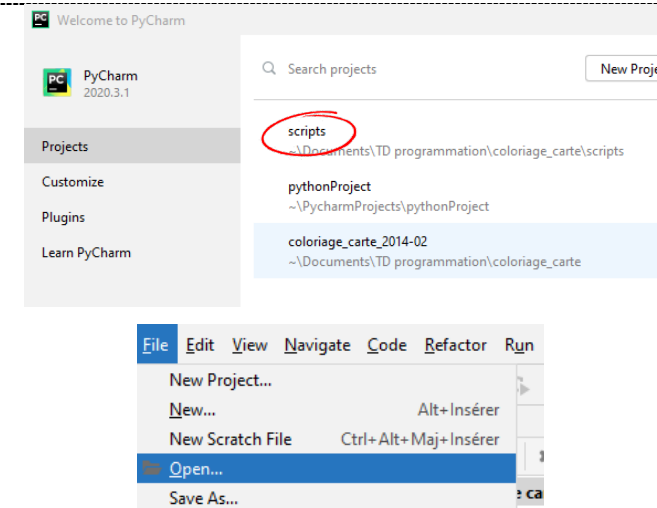
Ouverture d'un projet

On peut ouvrir un projet de nombreuses façons différentes.

- Avec le menu contextuel dans l'explorateur *Windows*
- Par Glisser / Déposer vers l'application *Pycharm* à partir de l'explorateur du système
- A l'ouverture de *Pycharm* avec la commande *Open*



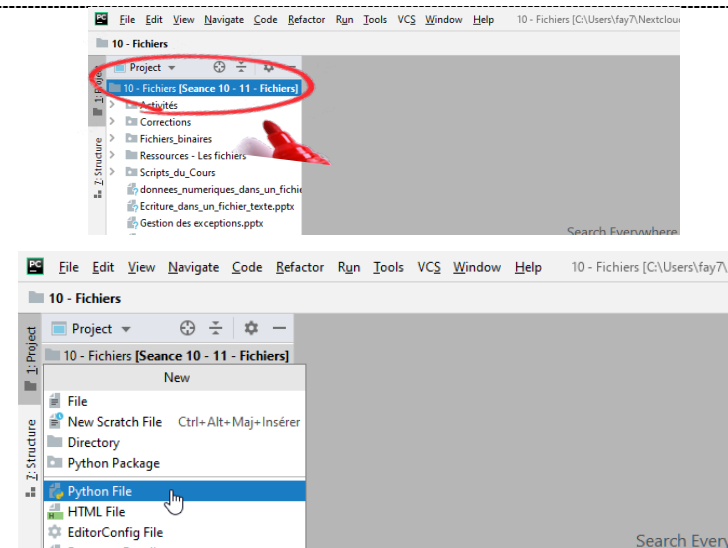
- En sélectionnant un des projets récents à l'ouverture de *Pycharm* ou ensuite dans le menu *File / Recent Projects*
- Par la commande *File / Open*



Création d'un nouveau script

1. Sélectionner le dossier de stockage du script dans l'onglet *Project*
2. Puis sélectionner la commande *File / New / Python File*

Ne pas créer de fichier dans le dossier Scratches and Consoles car cela crée des fichiers temporaires qui ne sont pas du tout stockés dans le dossier du projet.



Ouverture d'un script existant

- Double-cliquer sur le nom du fichier dans la liste des fichiers du projet

