

Documentation pour l'installation de l'environnement Python avec Github Desktop

pour le projet d'automatisation

**Stage de conception et développement d'outils automatisés pour la
réalisation de rapports d'analyses bibliométriques**

Tuteur :

Lokman SBOUI

Encadrement :

Judith BOISSONNEAULT

Sonia SECK

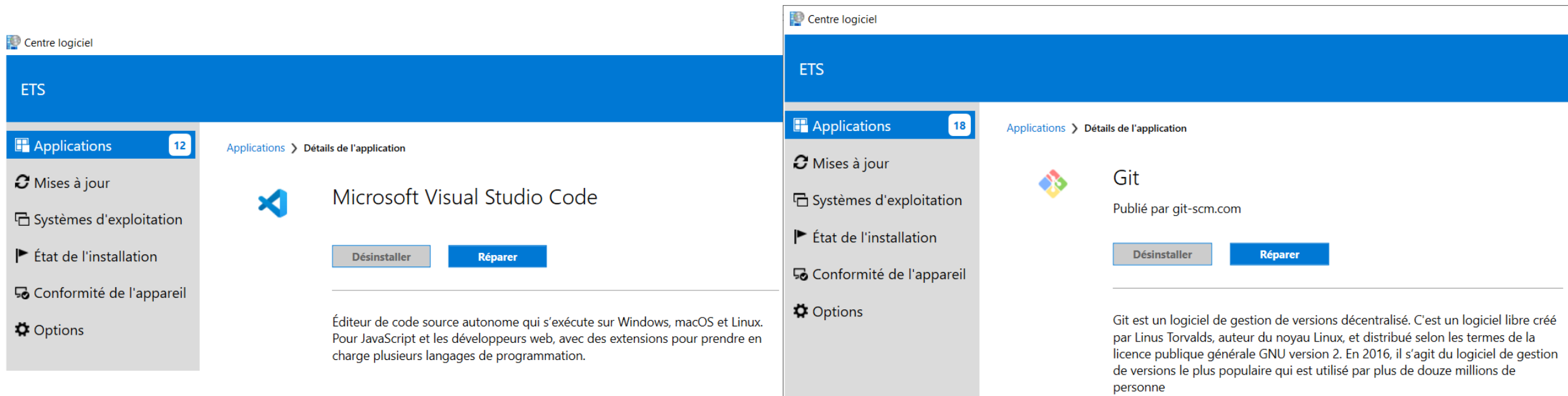
Vanessa AYOTTE

Plan

1. Installation de Git, Visual Studio Code et Python
2. Git: clone du dépôt distant
3. Configuration de Visual Studio Code
4. Installation des bibliothèques nécessaires
5. Lancement du programme principal
6. A l'avenir: Mettre à jour le repository du projet

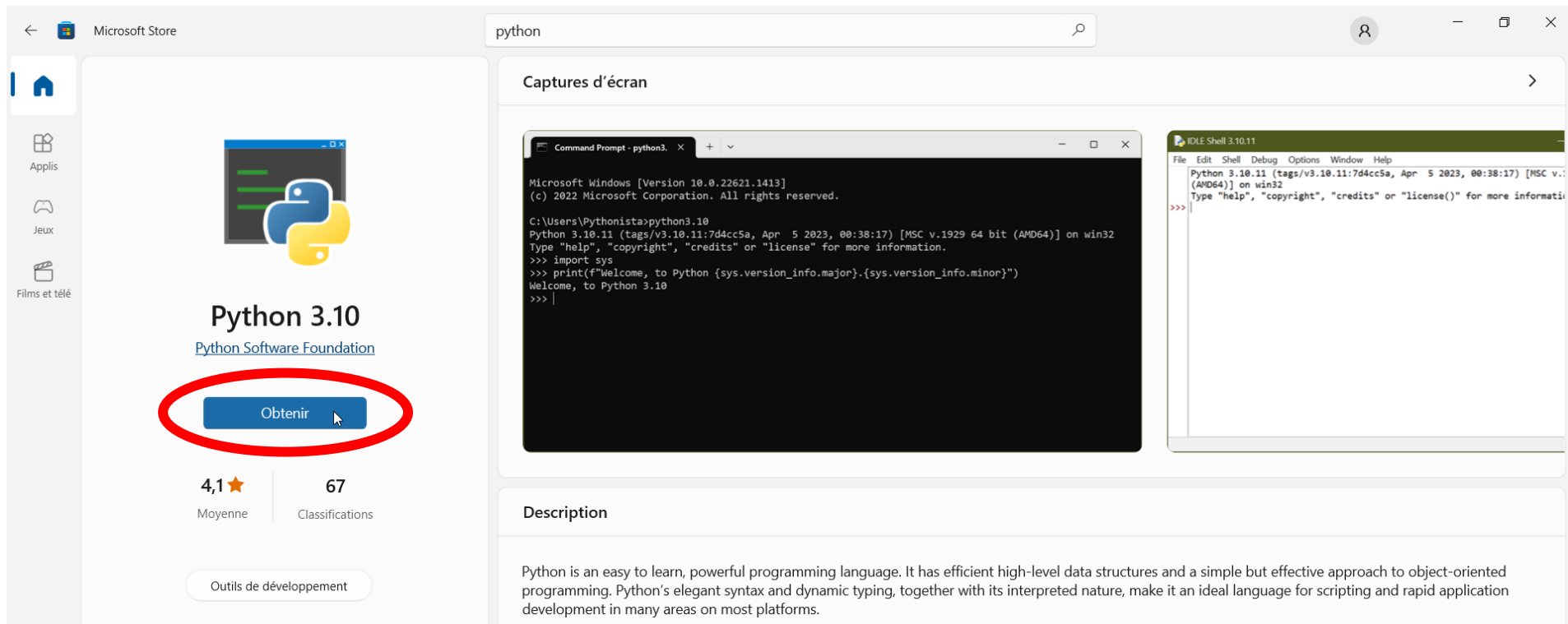
1. Installation de Git, Visual Studio Code et Python

- Pour Git et Visual Studio Code, vous pouvez les installer via le Centre Logiciel:



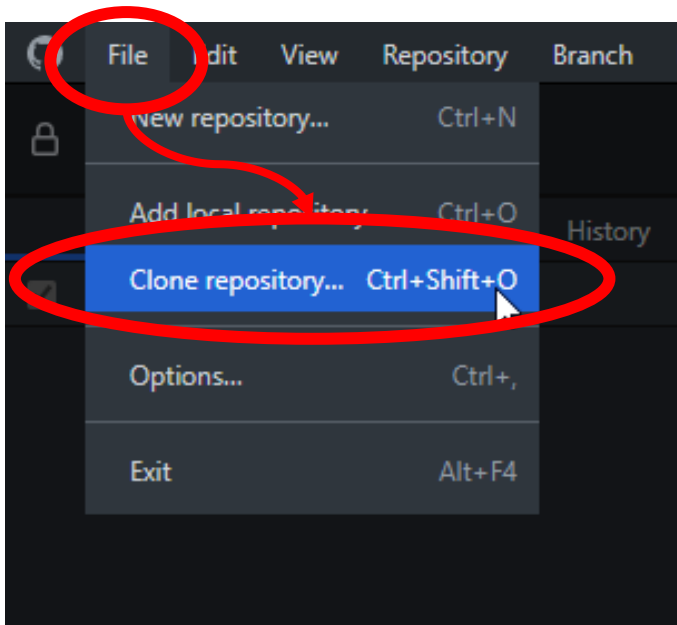
1. Installation de Git, Visual Studio Code et Python

- Pour Python, vous pouvez l'installer via Microsoft Store (installer de préférence la dernière version):



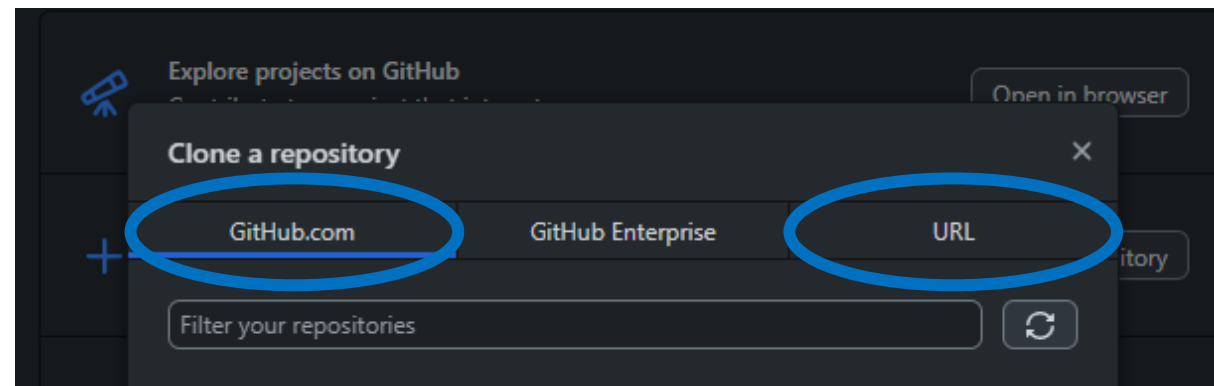
2. Clone du dépôt distant

- Commençons par le commencement, nous devons cloner le dépôt distant sur notre PC donc en « local » :



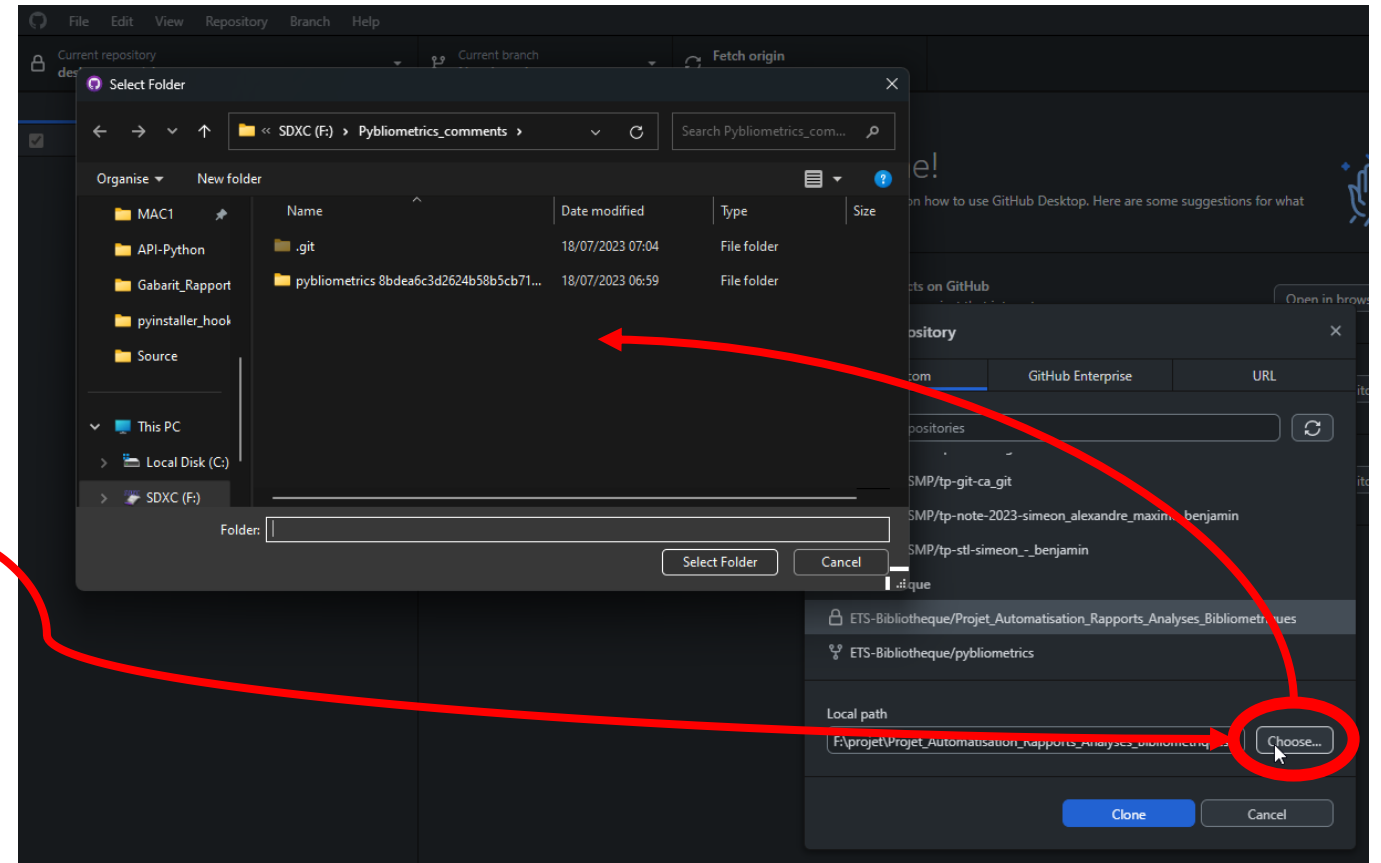
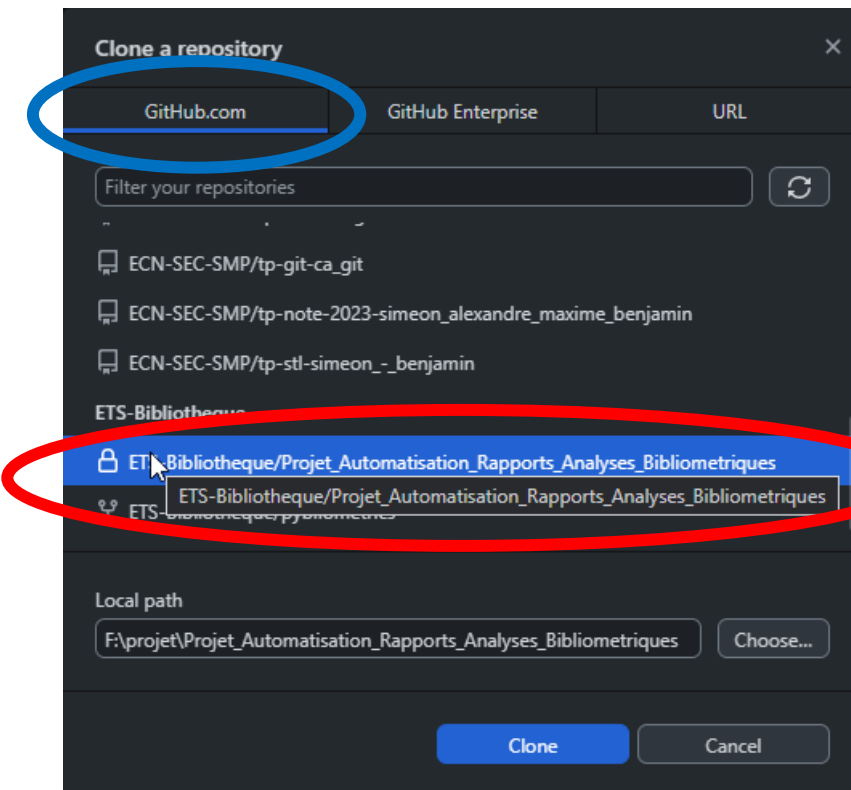
Vous avez alors 2 choix pour cloner le dépôt/repository via les onglets:

- [« GitHub.com »: avec authentification à votre compte GitHub](#)
- [« URL »: avec l'URL du dépôt distant](#)



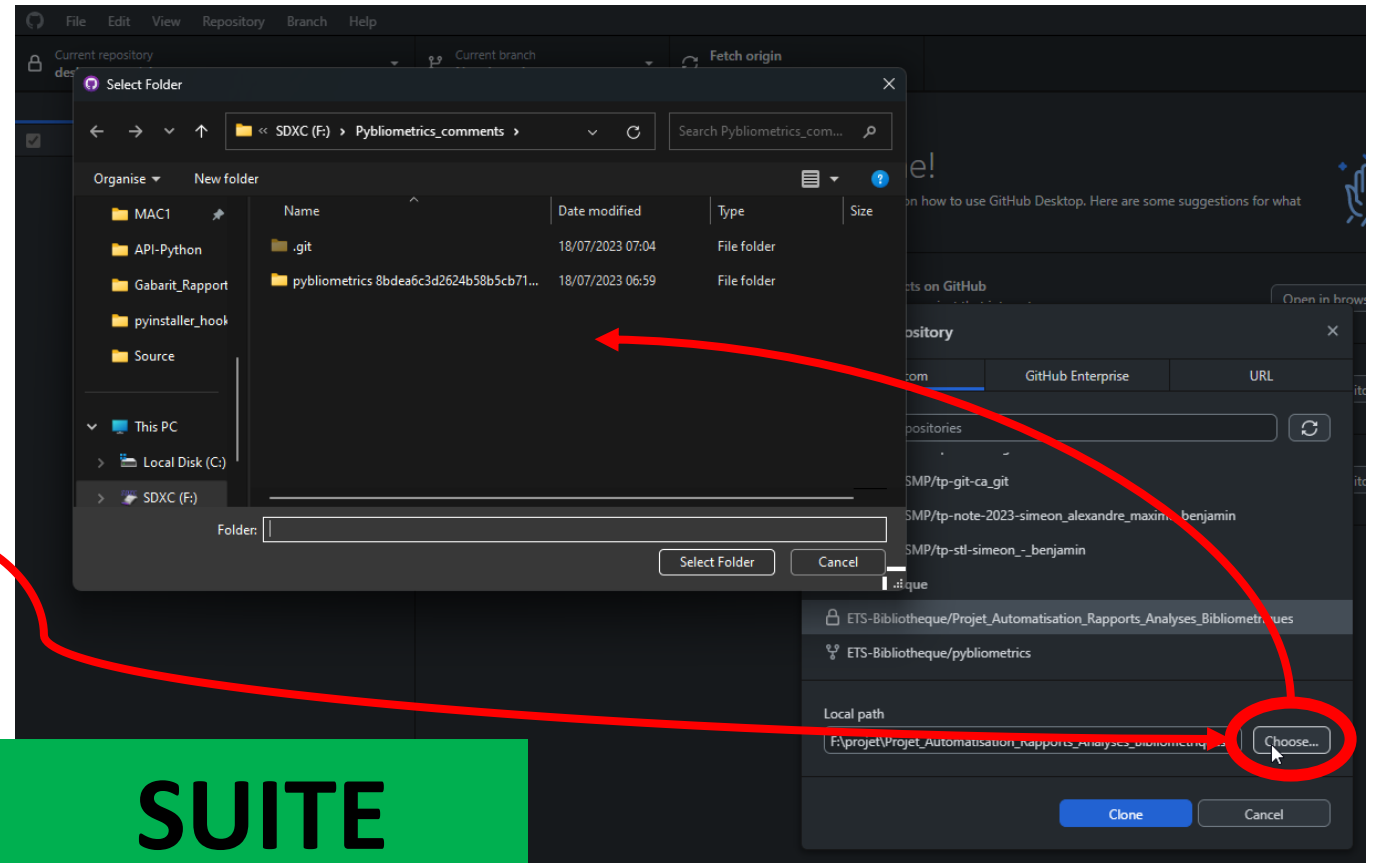
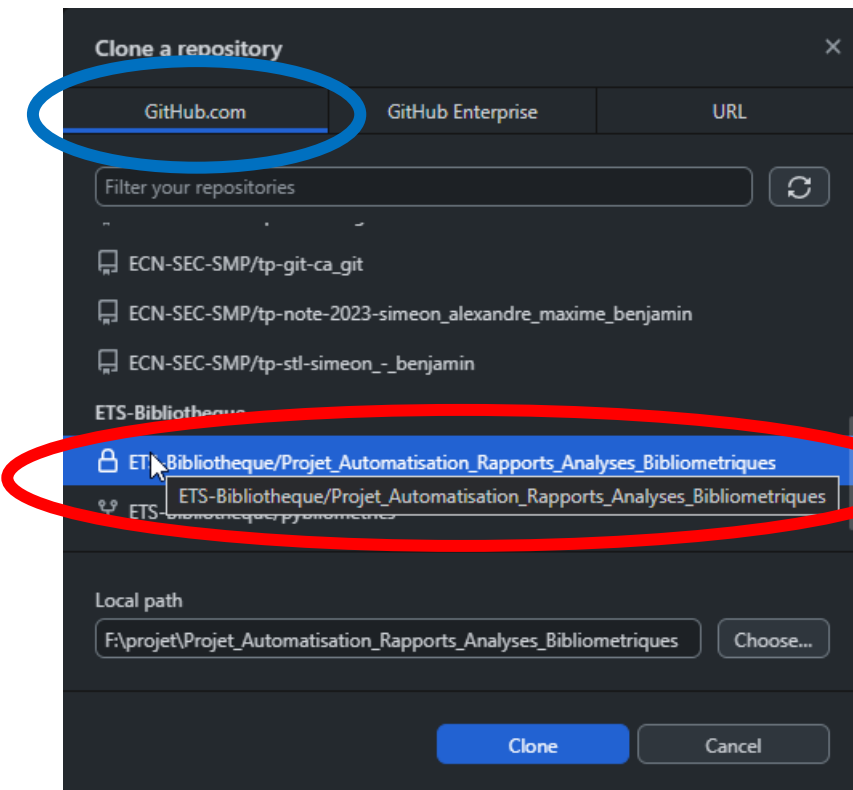
2. Clone du dépôt distant/ GitHub.com

- Avec « GitHub.com »: Sélectionnez le dépôt **puis** choisissez l'endroit où vous voulez le mettre sur votre machine



2. Clone du dépôt distant/ GitHub.com

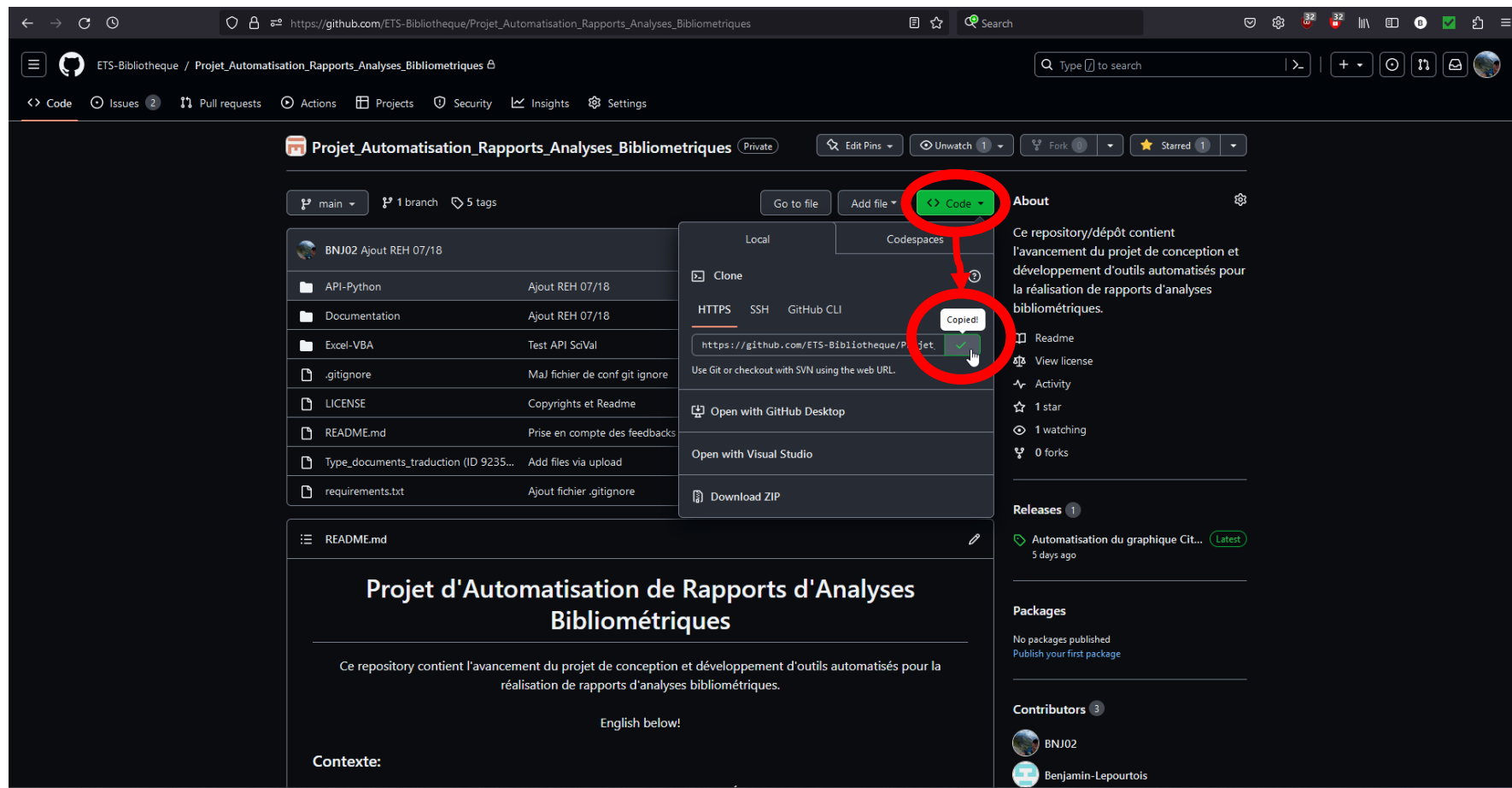
- Avec « GitHub.com »: Sélectionnez le dépôt **puis** choisissez l'endroit où vous voulez le mettre sur votre machine



SUITE
slide 10

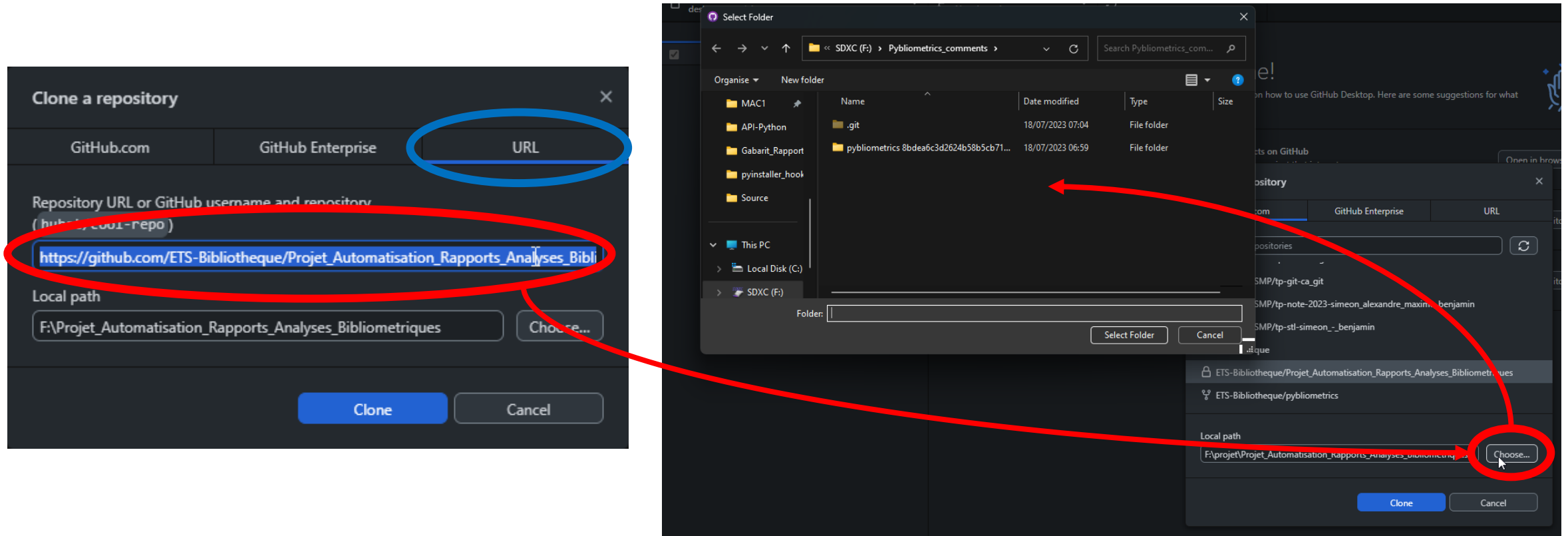
2. Clone du dépôt distant/ URL

- Avec « URL »: Rendez-vous sur [Github](https://github.com/ETS-Bibliotheque/Projet_Automatisation_Rapports_Analyses_Bibliometriques) et en particulier sur le « repository » du projet, vous pouvez alors copier l'adresse « https » en cliquant sur « < > Code » puis sur l'icône presse-papier:



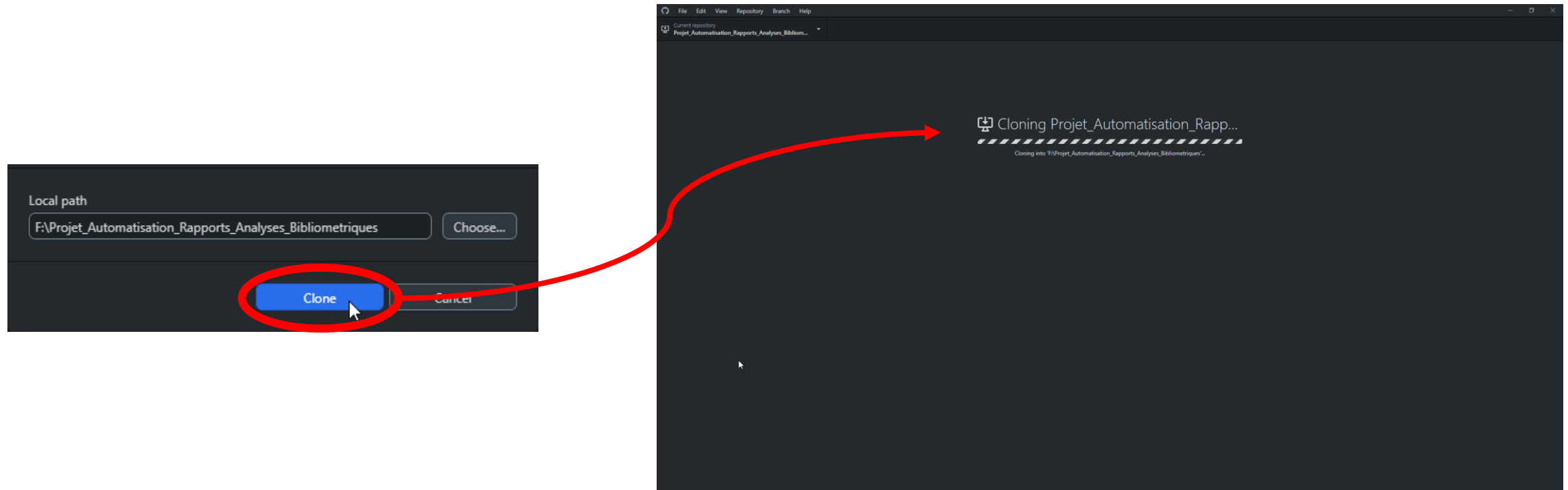
2. Clone du dépôt distant/ URL

- Avec « URL »: Collez l'URL maintenant le chemin d'accès: <Clique droit> dans la barre puis « Copier » OU <Clique gauche> puis <Ctrl+C>



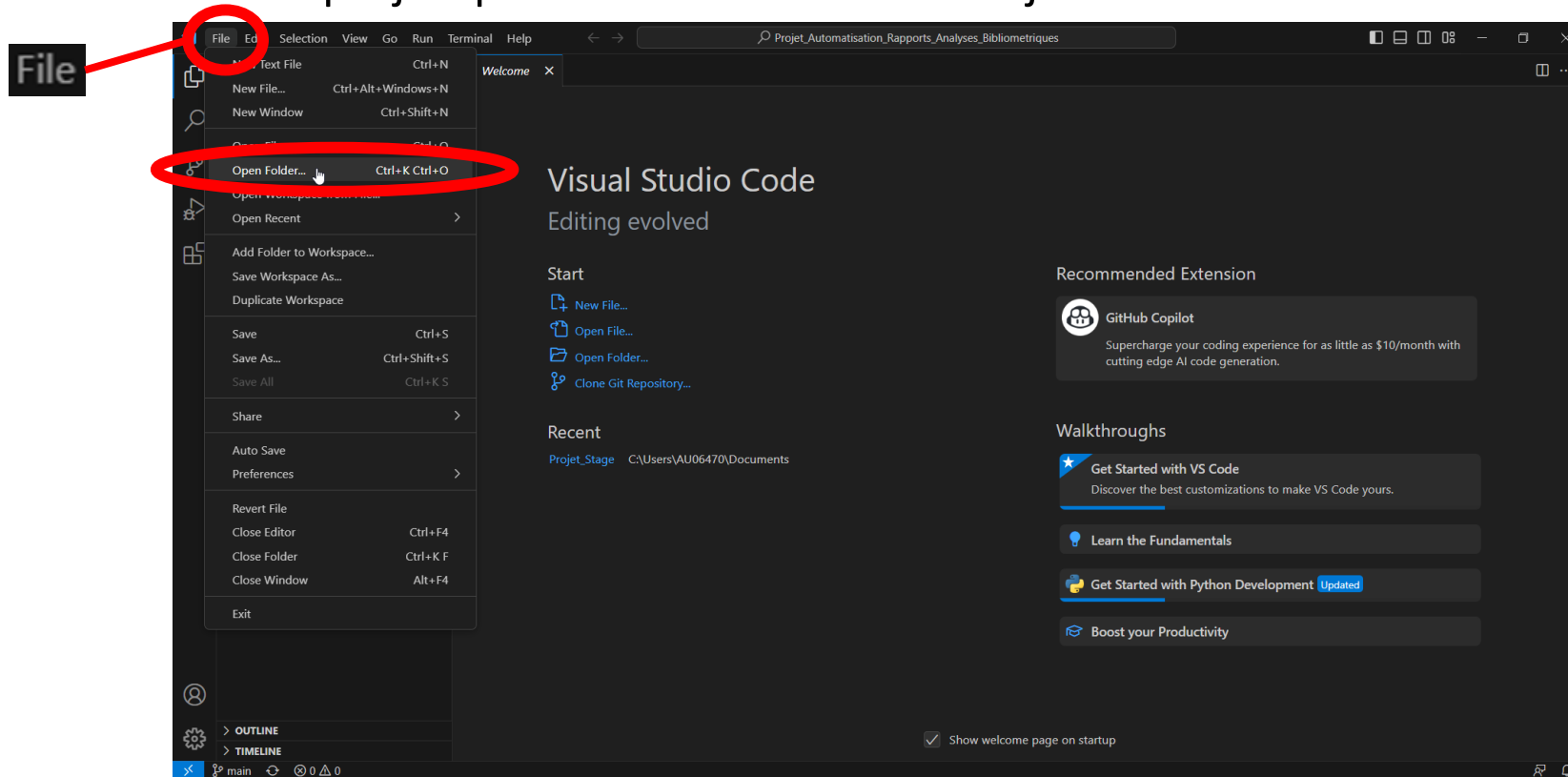
2. Clone du dépôt distant

- Avec « URL »: Commençons par le commencement, nous devons cloner le dépôt distant sur notre PC donc en « local » :



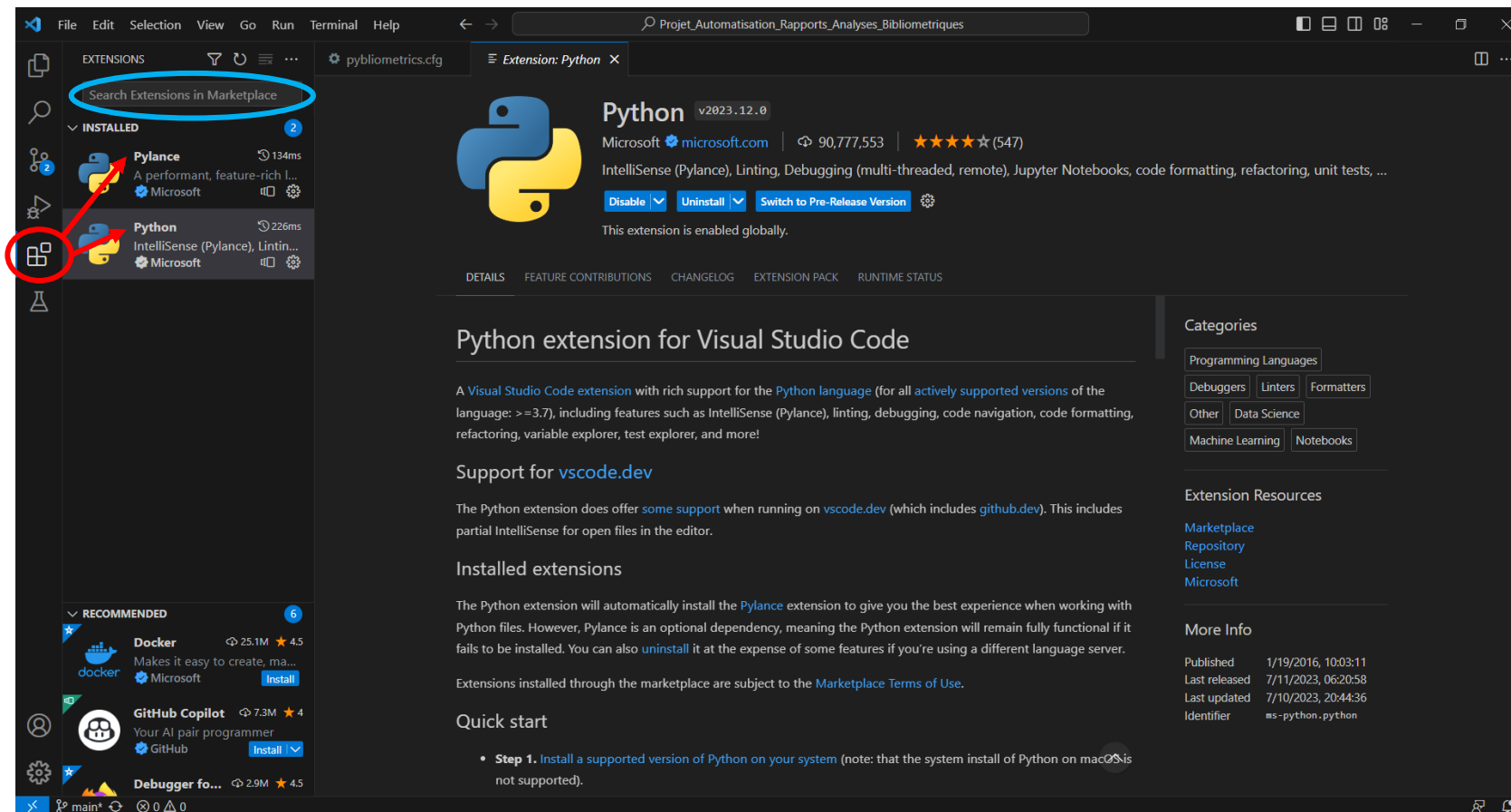
3. Configuration de Visual Studio Code

- Lancez Visual Studio Code via la barre de recherche Windows.
- Ouvrez le dossier du projet que nous venons de cloner juste avant.



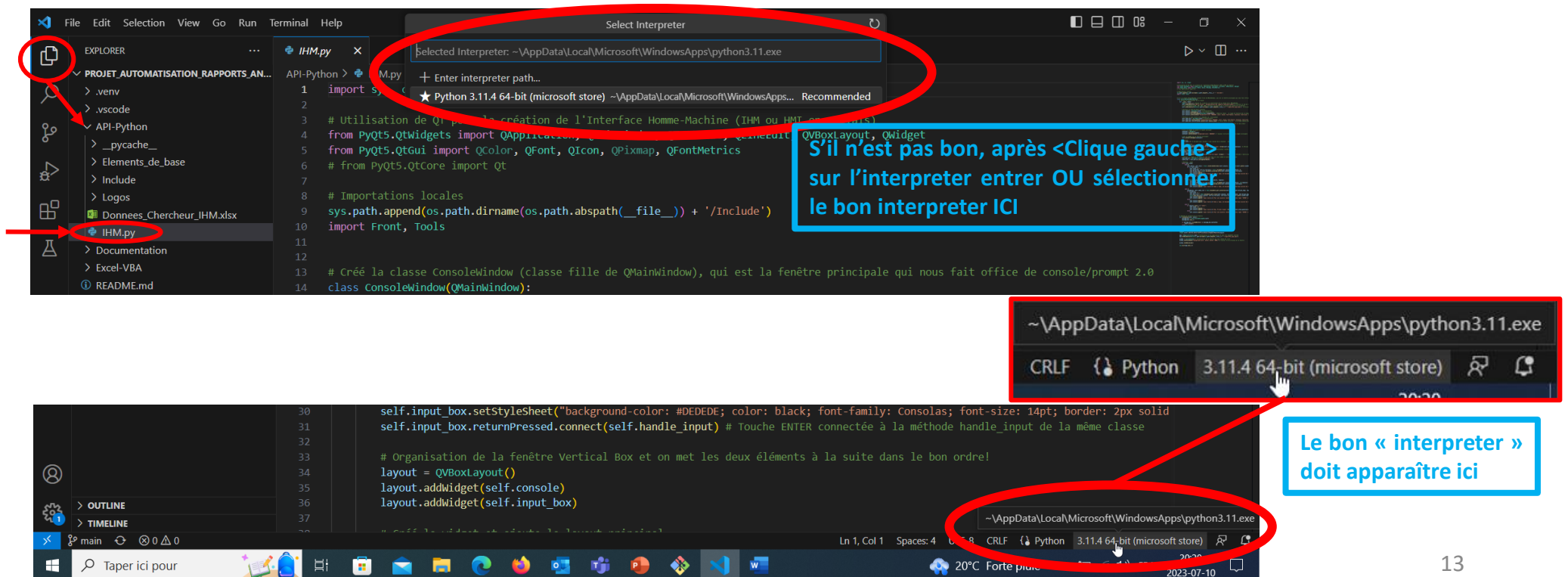
3. Configuration de Visual Studio Code

- Dans l'onglet « Extensions » (onglet « View » > Extensions), installez « Pylance » et « Python » à l'aide de la barre de recherche de l'onglet, comme ci-dessous:



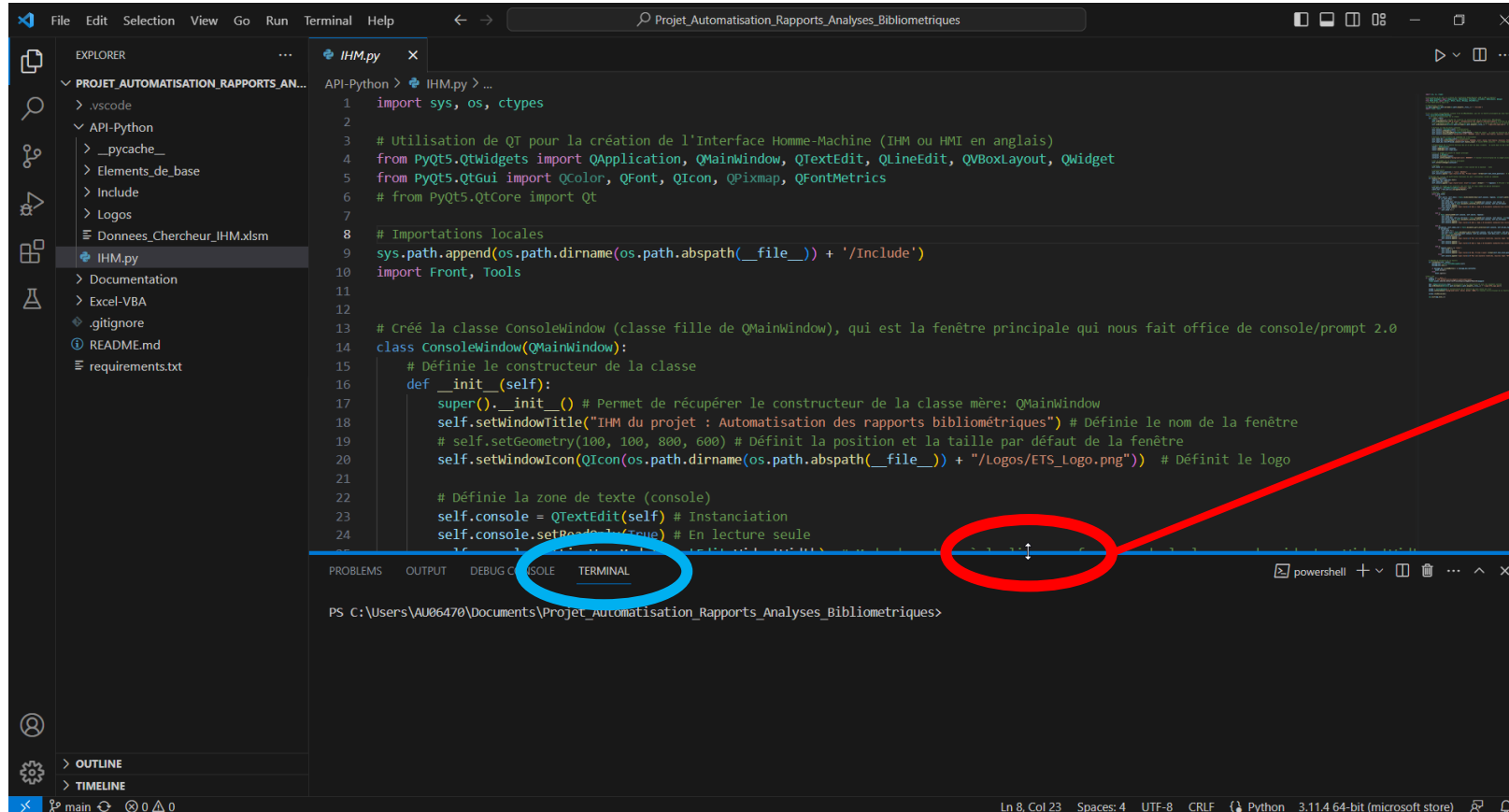
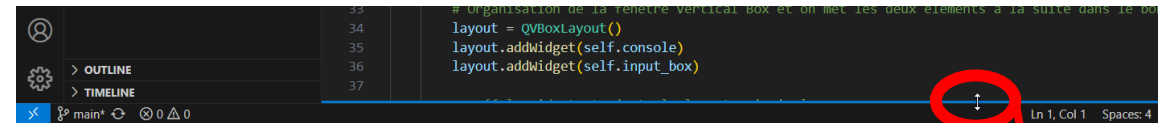
3. Configuration de Visual Studio Code

- Cliquez sur « IHM.py » dans l'arborescence du projet (**DANS** API-Python > IHM.py)
- Vérifiez ou Sélectionnez le bon « interpreter » Python. Nous voulons celui que nous avons installé avec Microsoft Store donc avec la notation « (microsoft store) ». Sinon choisir le bon:



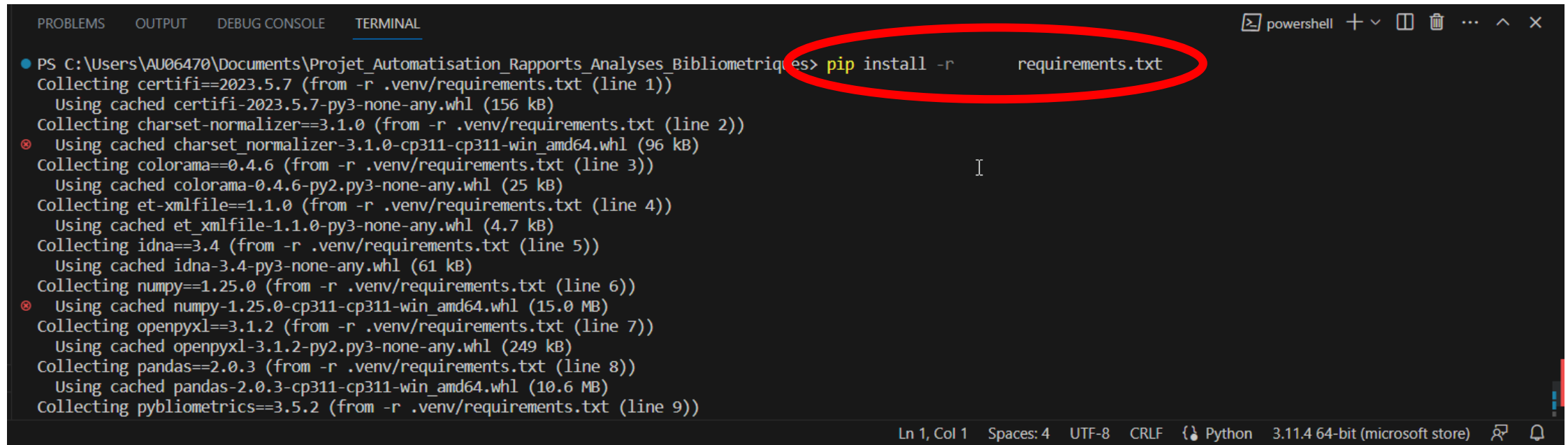
4. Installation des bibliothèques nécessaires

- Tirez vers le haut la page du bas comme ceci:
- Assurez-vous d'être bien dans l'onglet « TERMINAL »



4. Installation des bibliothèques nécessaires

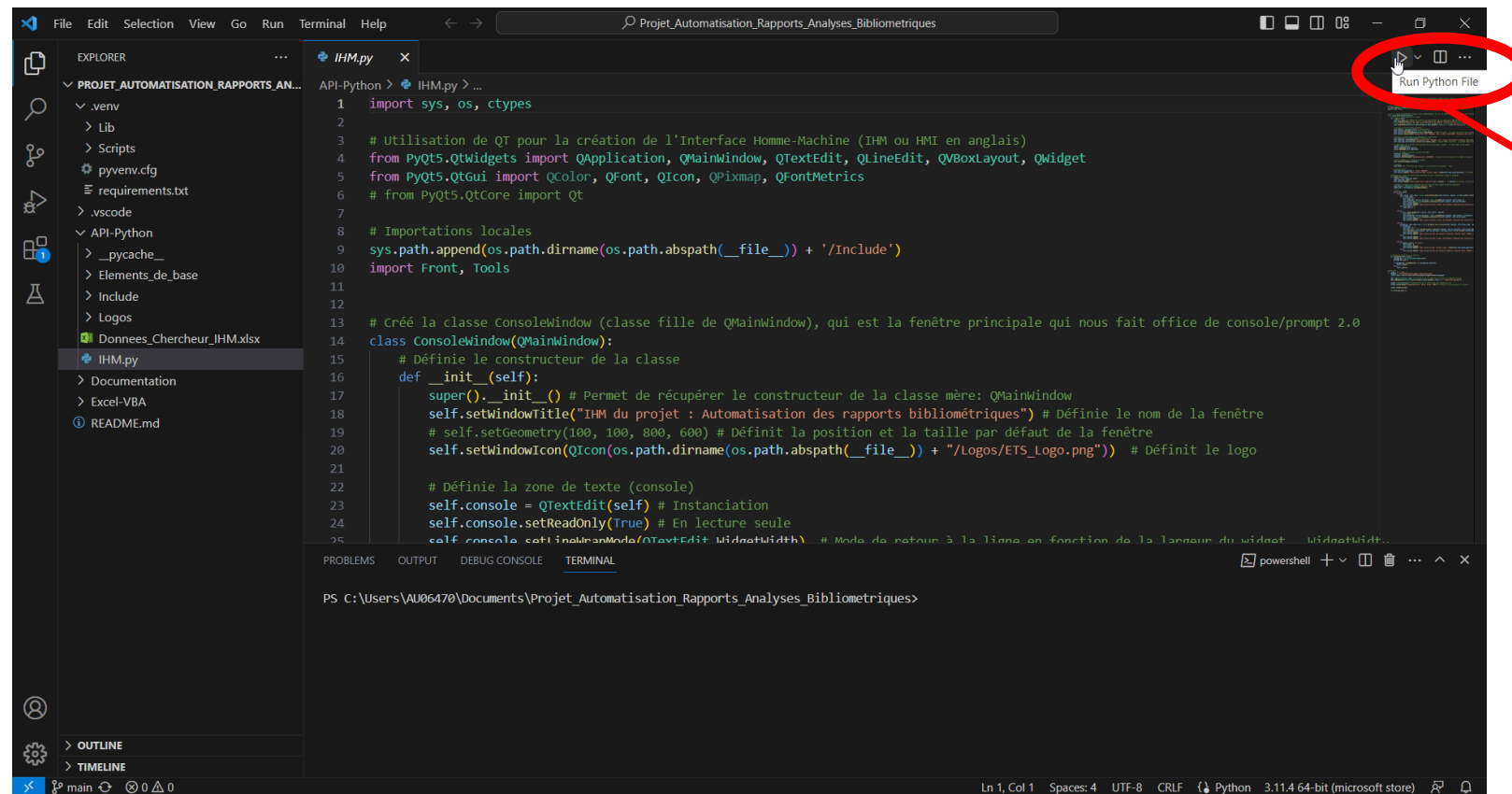
- Exécutez la commande suivante « `pip install -r requirements.txt` », toutes les bibliothèques devraient s'installer:



```
PS C:\Users\AU06470\Documents\Projet_Automatisation_Rapports_Analyses_Bibliometriques> pip install -r requirements.txt
Collecting certifi==2023.5.7 (from -r .env\requirements.txt (line 1))
  Using cached certifi-2023.5.7-py3-none-any.whl (156 kB)
Collecting charset-normalizer==3.1.0 (from -r .env\requirements.txt (line 2))
  Using cached charset_normalizer-3.1.0-cp311-cp311-win_amd64.whl (96 kB)
Collecting colorama==0.4.6 (from -r .env\requirements.txt (line 3))
  Using cached colorama-0.4.6-py2.py3-none-any.whl (25 kB)
Collecting et-xmlfile==1.1.0 (from -r .env\requirements.txt (line 4))
  Using cached et_xmlfile-1.1.0-py3-none-any.whl (4.7 kB)
Collecting idna==3.4 (from -r .env\requirements.txt (line 5))
  Using cached idna-3.4-py3-none-any.whl (61 kB)
Collecting numpy==1.25.0 (from -r .env\requirements.txt (line 6))
  Using cached numpy-1.25.0-cp311-cp311-win_amd64.whl (15.0 MB)
Collecting openpyxl==3.1.2 (from -r .env\requirements.txt (line 7))
  Using cached openpyxl-3.1.2-py2.py3-none-any.whl (249 kB)
Collecting pandas==2.0.3 (from -r .env\requirements.txt (line 8))
  Using cached pandas-2.0.3-cp311-cp311-win_amd64.whl (10.6 MB)
Collecting pybliometrics==3.5.2 (from -r .env\requirements.txt (line 9))
```

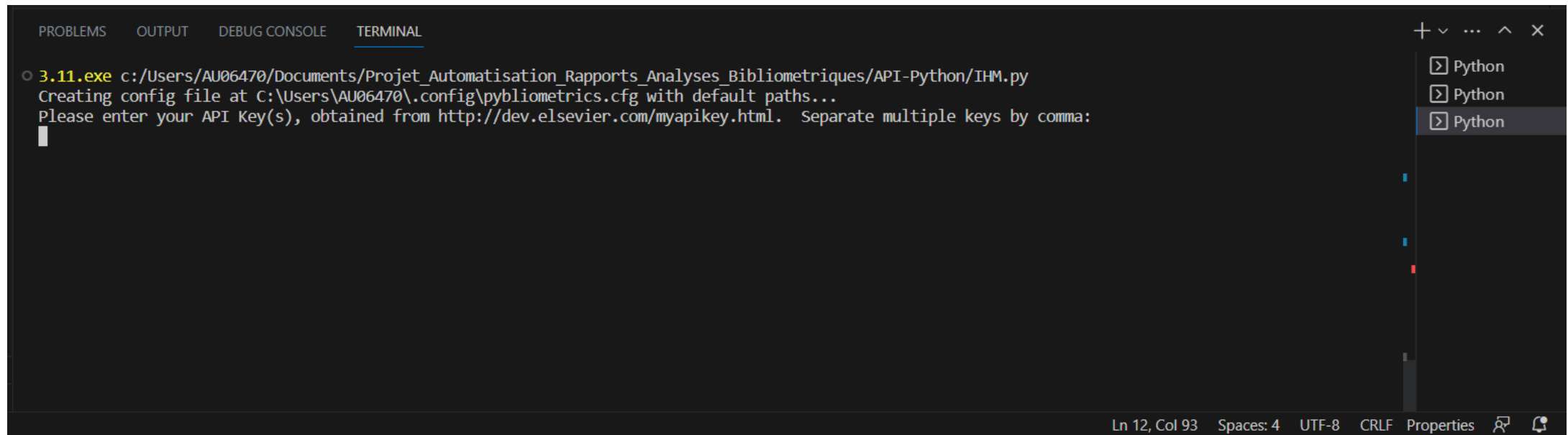
5. Lancement du programme principal

- Cliquez sur l'icône « RUN » (en vous assurant d'être sur le fichier IHM.py)



5. Lancement du programme principal

- La bibliothèque\API vous demande une clef API, rentrez alors la clef qui vous a été fournie:

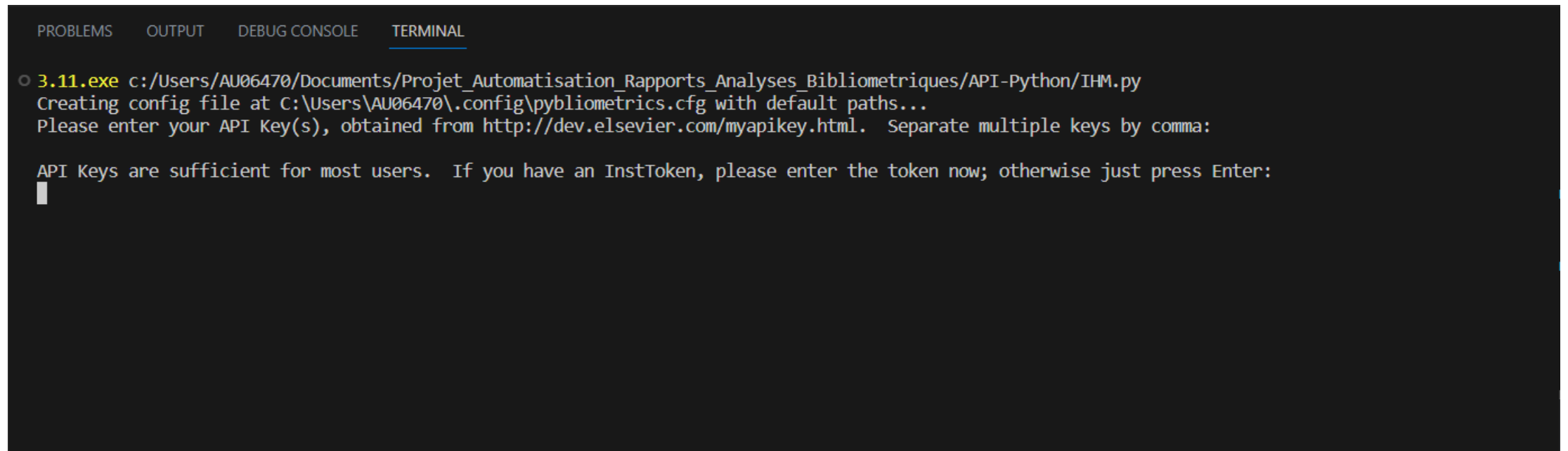


```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
3.11.exe c:/Users/AU06470/Documents/Projet_Automatisation_Rapports_Analyses_Bibliometriques/API-Python/IHM.py
Creating config file at C:\Users\AU06470\.config\pybliometrics.cfg with default paths...
Please enter your API Key(s), obtained from http://dev.elsevier.com/myapikey.html. Separate multiple keys by comma:

```

5. Lancement du programme principal

- La bibliothèque\API vous demande alors un token institutionnel, rentrez alors le token qui vous a été fourni:



```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL

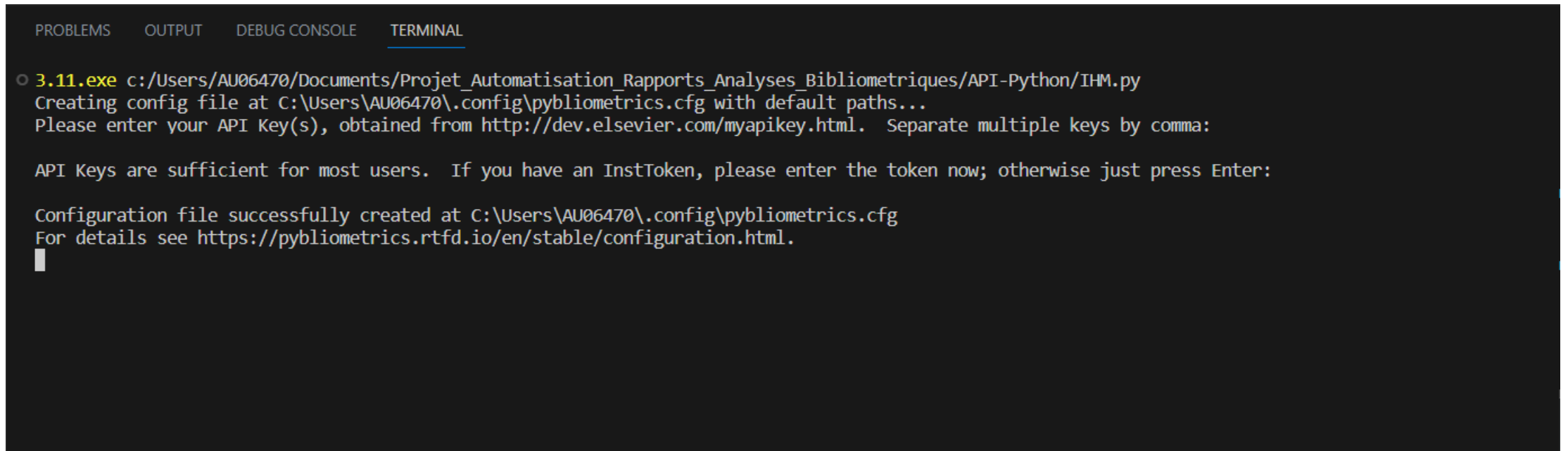
○ 3.11.exe c:/Users/AU06470/Documents/Projet_Automatisation_Rapports_Analyses_Bibliometriques/API-Python/IHM.py
Creating config file at C:\Users\AU06470\.config\pybliometrics.cfg with default paths...
Please enter your API Key(s), obtained from http://dev.elsevier.com/myapikey.html. Separate multiple keys by comma:

API Keys are sufficient for most users. If you have an InstToken, please enter the token now; otherwise just press Enter:
█
```

5. Lancement du programme principal

- Normalement, un fichier de configuration se crée avec les données que vous venez de rentrer.

/!\ Je vous conseille de noter l'emplacement de ce fichier de configuration si vous souhaitez ajouter des clefs API /!\



```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL

○ 3.11.exe c:/Users/AU06470/Documents/Projet_Automatisation_Rapports_Analyses_Bibliometriques/API-Python/IHM.py
Creating config file at C:\Users\AU06470\.config\pybliometrics.cfg with default paths...
Please enter your API Key(s), obtained from http://dev.elsevier.com/myapikey.html. Separate multiple keys by comma:

API Keys are sufficient for most users. If you have an InstToken, please enter the token now; otherwise just press Enter:

Configuration file successfully created at C:\Users\AU06470\.config\pybliometrics.cfg
For details see https://pybliometrics.rtf.d.io/en/stable/configuration.html.
█
```

5. Lancement du programme principal

- L'interface devrait s'afficher automatiquement sur votre écran comme ceci:

IHM du projet : Automatisation des rapports bibliométriques

• Nom, Prénom:
▶ SBOUI, Lokman

Chercheur.s trouvé.s:

	Nom	Prénom	Affiliation	Nb docs	Ville	Pays	Domaine(s) de recherche
Index							
0	Sboui	Lokman	École de Technologie Supérieure	34	Montreal	Canada	COMP (41); ENGI (40); MATH (7)

Résumé du chercheur sélectionné:
Sboui L. from Department of Systems Engineering in Canada, published 34 document(s) since 2011 which were cited by 343 author(s) in 438 document(s) as of 2023-07-10

Nb de documents en fonction de leur type:

	Type de documents	Nombre
Index		
0	Conference Paper	22
1	Article	12

Total: 34

• Quel.s type.s de documents souhaitez-vous exclure de la liste (index avec virgule comme séparateur)?
▶

Votre sélection: Conference Paper, Article

Tableau pour le graphique des citations:

Année	2011	2012	2013	2014	2015	2016	2017	2018	2019	2020	2021	2022	2023
Citations	0	0	6	14	31	36	54	52	66	54	55	51	19
Documents	1	0	4	3	6	6	4	2	3	1	1	3	0

• Pour une nouvelle recherche, veuillez taper "OTHER":



6. A l'avenir: Mettre à jour le repository du projet

- Supprimez les changements apportés au projet (lors de l'utilisation)
- **/!\ Si c'est des fichiers que vous voulez garder, veuillez les exporter hors du dossier du projet /!**

<Clique droit>

Confirm discard all changes

Are you sure you want to discard all changes to:

- API-Python\Donnees_Chercheur_IHM.xlsm
- API-Python\Rapport.docx

Changes can be restored by retrieving them from the Recycle Bin.

Discard all changes Cancel

Error

Failed to discard changes to Recycle Bin.

Common reasons are:

- The Recycle Bin is configured to delete items immediately.
- Restricted access to move the file(s).

These changes will be unrecoverable from the Recycle Bin.

☐ Do not show this message again

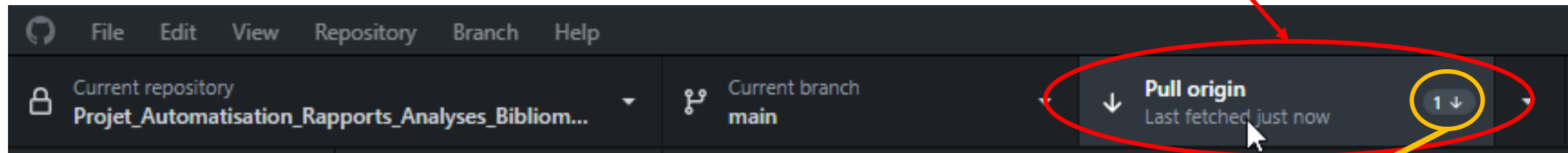
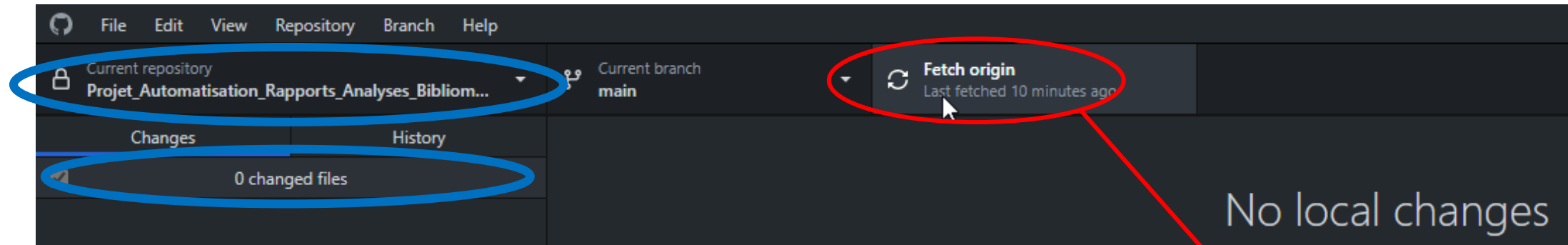
This will discard changes and they will be unrecoverable.

Permanently discard changes Cancel



6. A l'avenir: Mettre à jour le repository du projet

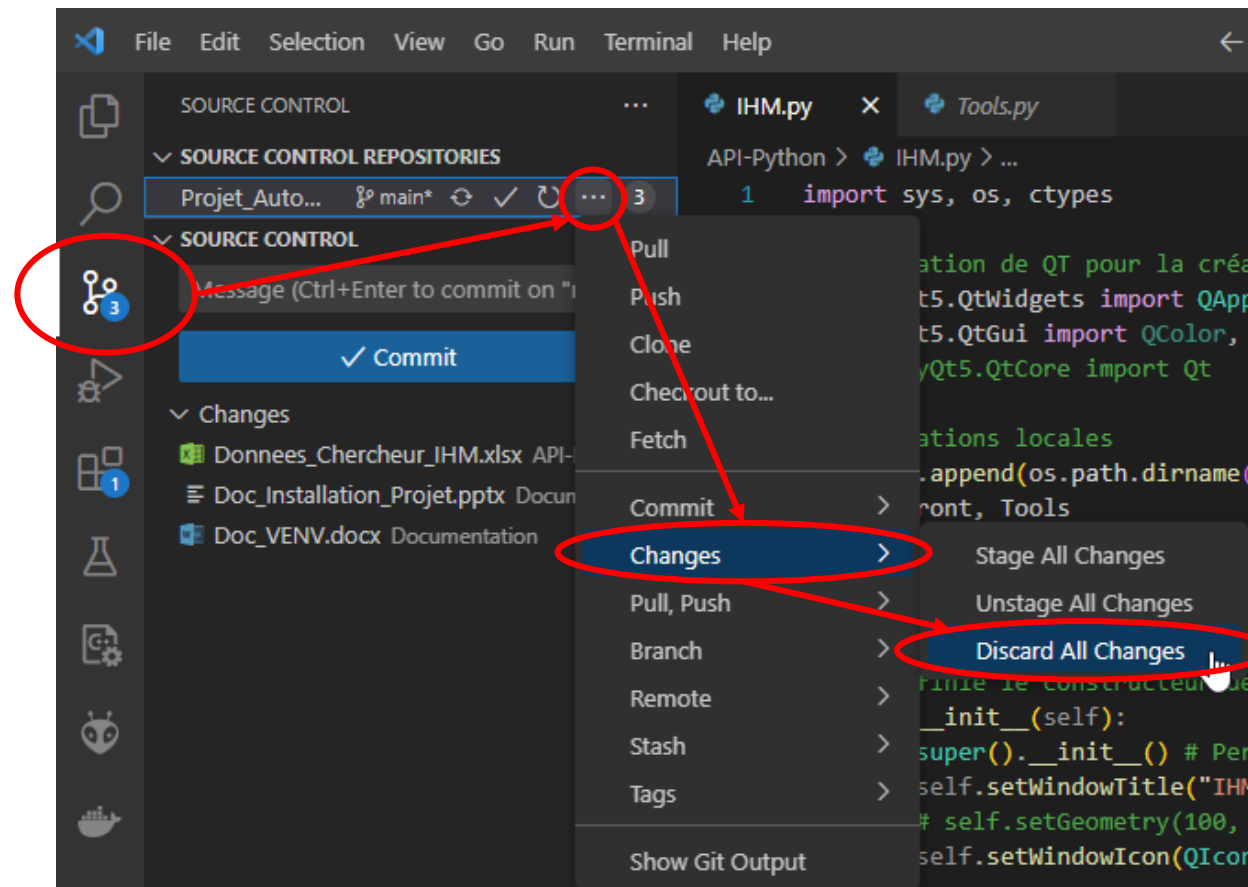
- Tirez les changements du dernier commit sur le dépôt distant, en s'assurant d'être bien connecté à votre compte Github:



Nb de commits de différence
entre les dépôts local et distant

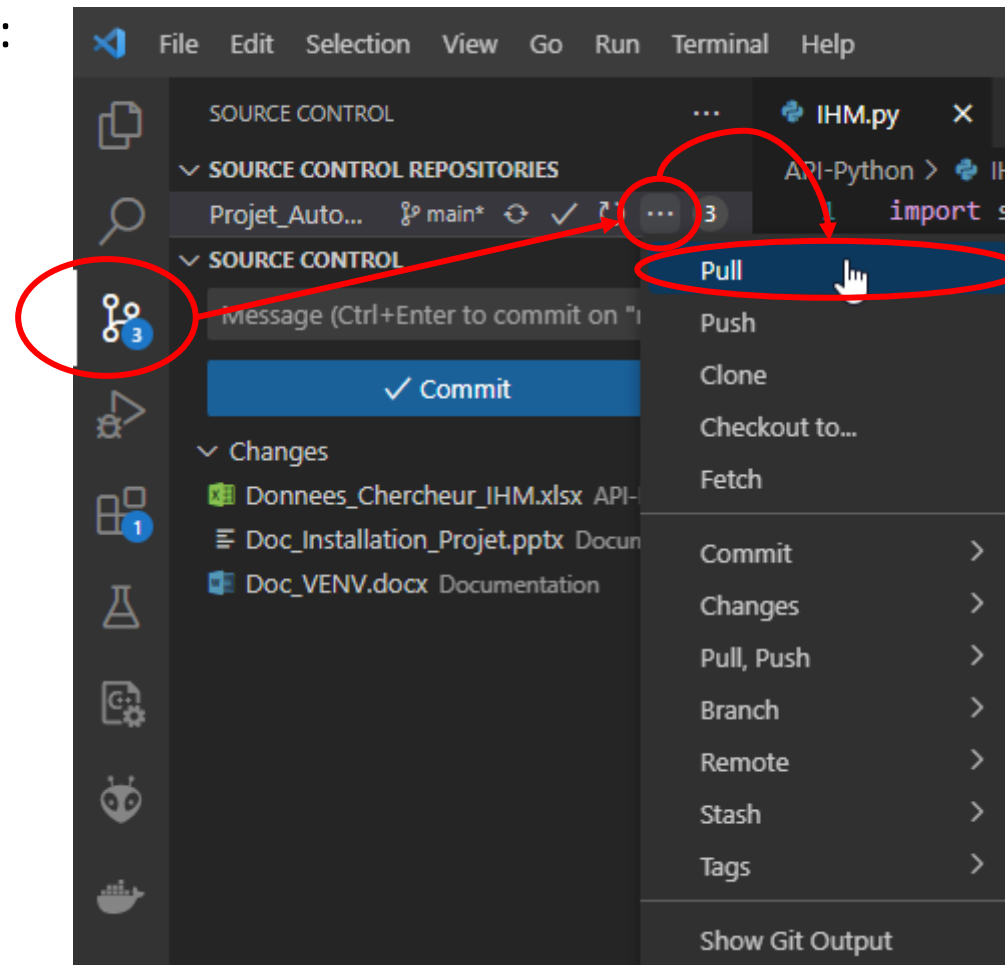
6. A l'avenir: Mettre à jour le repository du projet

- Supprimez les changements apportés au projet (lors de l'utilisation)
- **/!\ Si c'est des fichiers que vous voulez garder, veuillez les exporter hors du dossier du projet /!**



6. A l'avenir: Mettre à jour le repository du projet

- Tirez les changements du dernier commit sur le dépôt distant, en s'assurant d'être bien connecté à votre compte Github:



Pour me contacter:

Mail: benjamin.lepourtois@gmail.com

LinkedIn: <https://www.linkedin.com/in/benjamin-lepourtois-b09564232/>

GitHub : <https://github.com/BNJ02>