

Déroulement d'un TP

1. Récupération des fichiers/documents associés (pages 1 à 8)
 2. Execution du "Jupyter Notebook" (pages 9 à 10)
 3. Execution d'un programme Python (pages 11 à 12)
-

1. Récupération des fichiers/documents associés (pages 1 à 8)

a) Aller sur le site de Mr Mustafa et la page associée au cours:

<https://lipn.univ-paris13.fr/~mustafa/Teaching/USPN/Prog2-2026s/>

ou recherche google : Mustafa P13

The screenshot shows a web browser window with the following sections:

- Ressources**:
 - Jupyter Notebooks: [UGE-IGM / courspython](#).
 - Livre: [Cours de Python](#), Patrick Fuchs et Pierre Poulaing (PDF).
 - Visualizer code en Python: [Python Tutor](#).
- CM**:
 - Conseils pour réussir ce module**
 - CM 1.** [Slides](#), variable types, conditionnelles, commentaires.
Lecture : [variables_conditionnelles](#)
 - John Tate**:
Il est vrai que je ne suis pas très doué pour résoudre des problèmes. Par exemple, je n'aurais jamais été bon aux Olympiades de mathématiques. I.e., la rapidité compte et je ne suis certainement pas un travailleur rapide. C'est d'ailleurs une chose agréable en mathématiques : le temps que l'on met n'a pas d'importance, si le résultat final est un beau théorème. La vitesse est un avantage, mais elle n'est pas essentielle.
- TD**: (empty box)
- Contrôle continu**: (empty box)
- TP**:
 - TP 1** (seance 1). [Files](#): `trie_sol.py`, `trie.py`, `Strie.py`.
 - Jupyter Notebook Exercises: [Lec1-valeurs-variables-types.ipynb](#) (PDF, HTML).
- Python**:
 - [How to install Python3 modules on LINUX.](#)

Pour le TP1, il nous faut récupérer/télécharger différents fichiers (*.py et *.ipynb)

b) récupérer/télécharger les fichiers associés au TP1

Tout d'abord, nous allons préparer le terrain en créant une arborescence dédiée au cours:
répertoire du cours : `~/ProgImp2`
répertoire du TP1 : `~/ProgImp2/TP1`
répertoire du TP2 : `~/ProgImp2/TP2`
...

On propose deux méthodes.

Méthode 1: axée "ligne de commande"

Méthode 2: avec l'interface graphique

Méthode 1: axée "ligne de commande" (1/2)

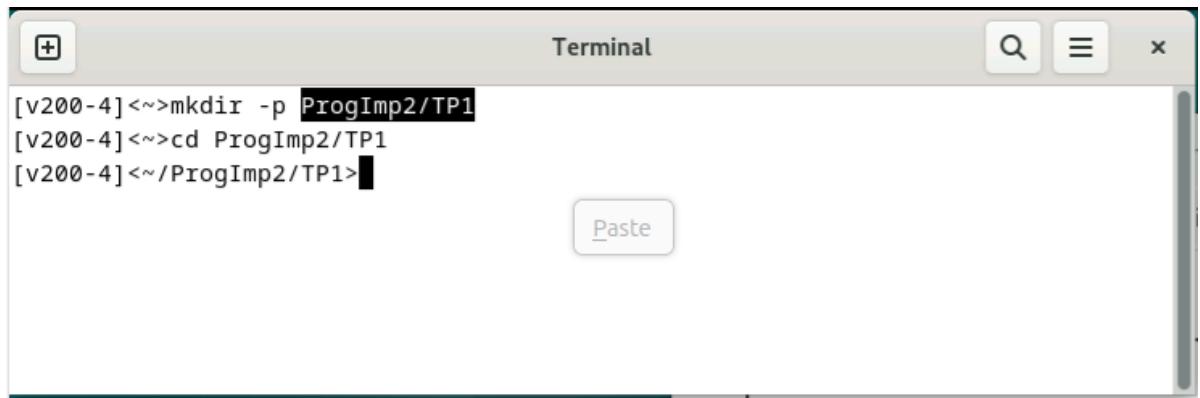
Dans un terminal, on execute la commande (Fig.2)

```
mkdir -p ~/ProgImp2/TP1
```

puis on se positionne dans le répertoire créé

```
cd ~/ProgImp2/TP1
```

Fig. 2



```
[v200-4] <~>mkdir -p ProgImp2/TP1
[v200-4] <~>cd ProgImp2/TP1
[v200-4] <~/ProgImp2/TP1> [
```

Ensuite, pour récupérer l'un des fichiers, on se positionne dessus (sur la page web) et on clique sur le bouton droit pour faire apparaître le menu contextuel, puis dans le menu, on va sur "Copy Link" (Fig.3) puis on clique sur le bouton gauche (le lien a été copié en mémoire)

Fig. 3

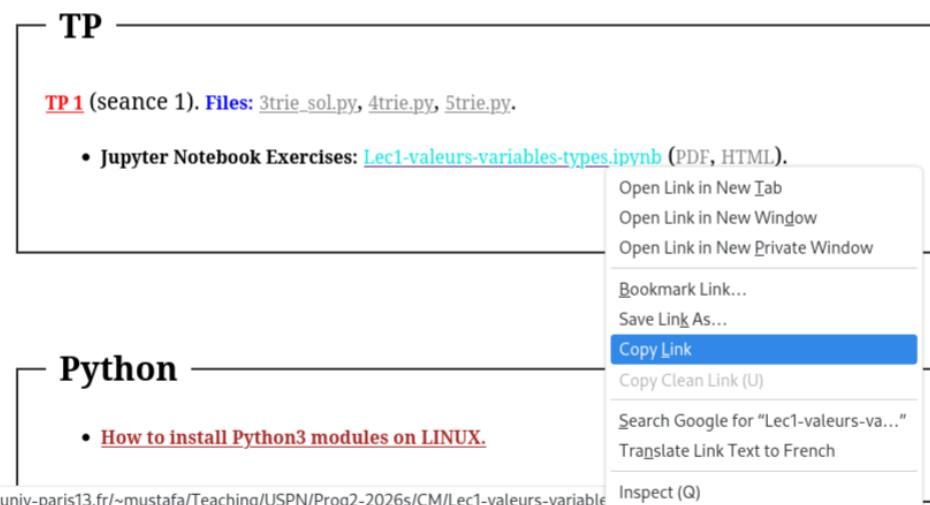
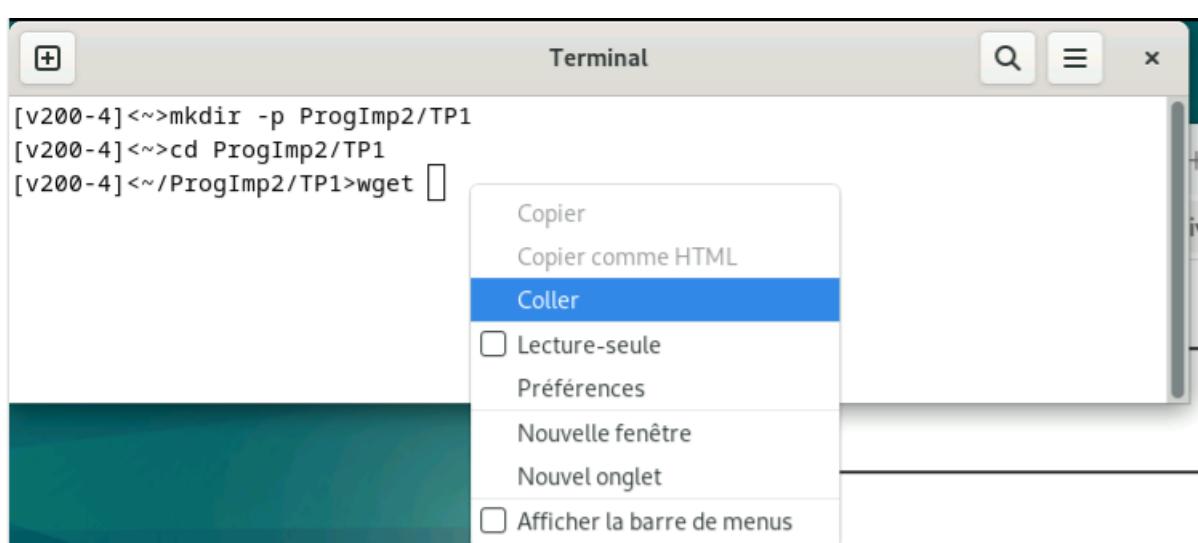
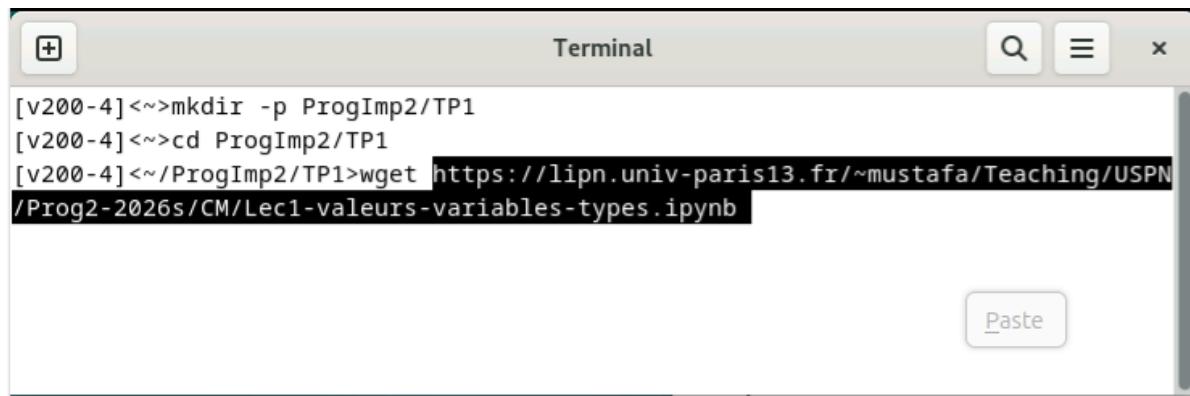


Fig. 4



Méthode 1: axée "ligne de commande" (2/2)

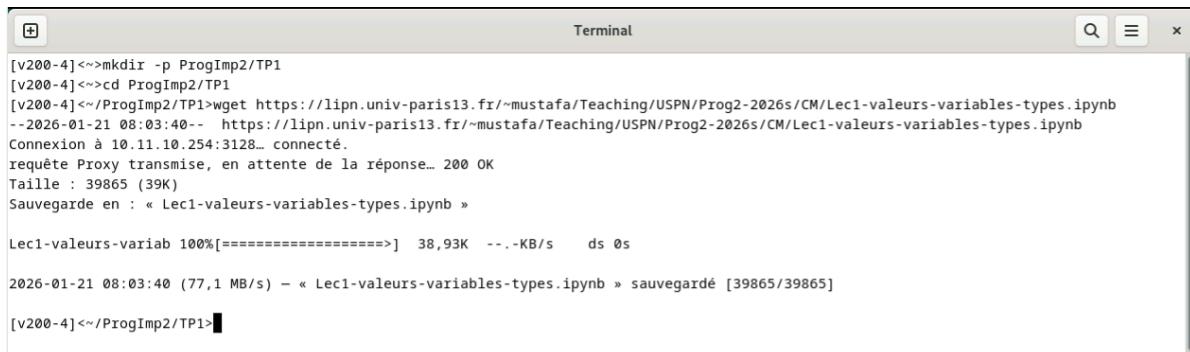
Fig. 5



```
[v200-4] <~>mkdir -p ProgImp2/TP1
[v200-4] <~>cd ProgImp2/TP1
[v200-4] <~/ProgImp2/TP1>wget https://lipn.univ-paris13.fr/~mustafa/Teaching/USPN/Prog2-2026s/CM/Lec1-valeurs-variables-types.ipynb
```

Ensuite, on execute la commande en appuyant sur "entrée/return" (résultat en Fig. 6)

Fig. 6



```
[v200-4] <~>mkdir -p ProgImp2/TP1
[v200-4] <~>cd ProgImp2/TP1
[v200-4] <~/ProgImp2/TP1>wget https://lipn.univ-paris13.fr/~mustafa/Teaching/USPN/Prog2-2026s/CM/Lec1-valeurs-variables-types.ipynb
--2026-01-21 08:03:40-- https://lipn.univ-paris13.fr/~mustafa/Teaching/USPN/Prog2-2026s/CM/Lec1-valeurs-variables-types.ipynb
Connexion à 10.11.10.254:3128... connecté.
requête Proxy transmise, en attente de la réponse... 200 OK
Taille : 39865 (39K)
Sauvegarde en : « Lec1-valeurs-variables-types.ipynb »

Lec1-valeurs-variab 100%[=====] 38,93K  --.-KB/s   0s
2026-01-21 08:03:40 (77,1 MB/s) - « Lec1-valeurs-variables-types.ipynb » sauvegardé [39865/39865]

[v200-4] <~/ProgImp2/TP1>
```

On effectue ces opérations sur chacun des fichiers à télécharger (copie lien + commande wget)

Méthode 2: avec l'interface graphique (1/5)

Pour créer le répertoire `~/ProgImp2/TP1` , on utilise l'application "Fichiers" (Fig. 7)

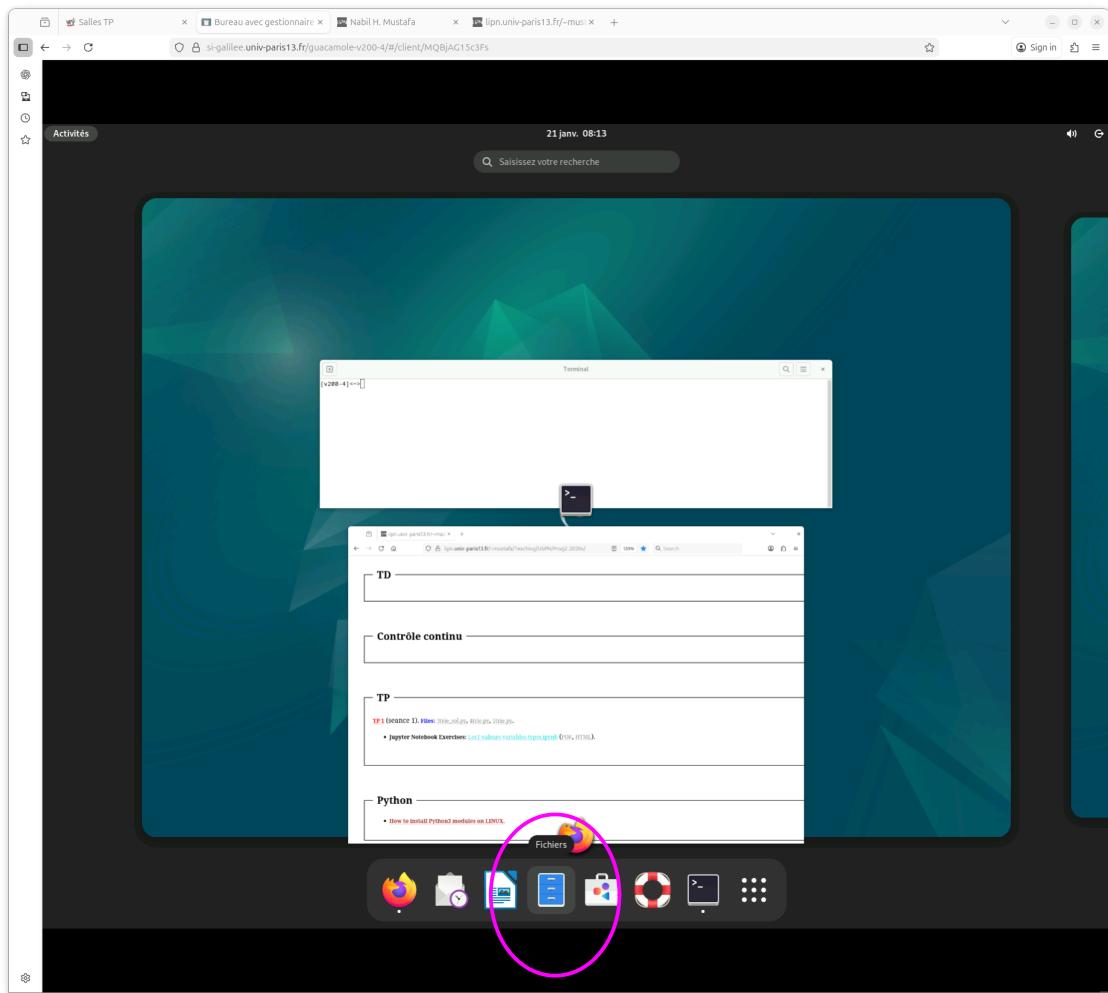


Fig. 7

Application "Fichiers" ouverte (Fig. 8)

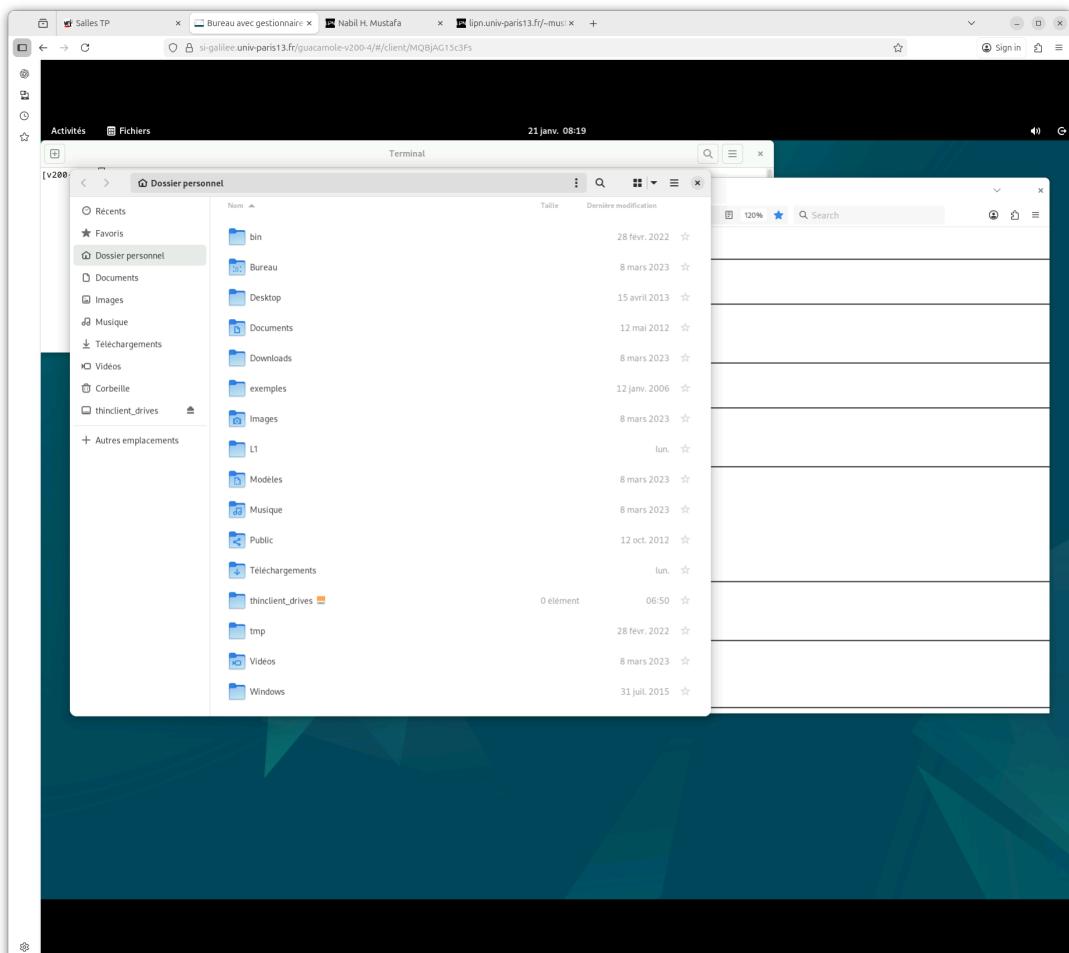
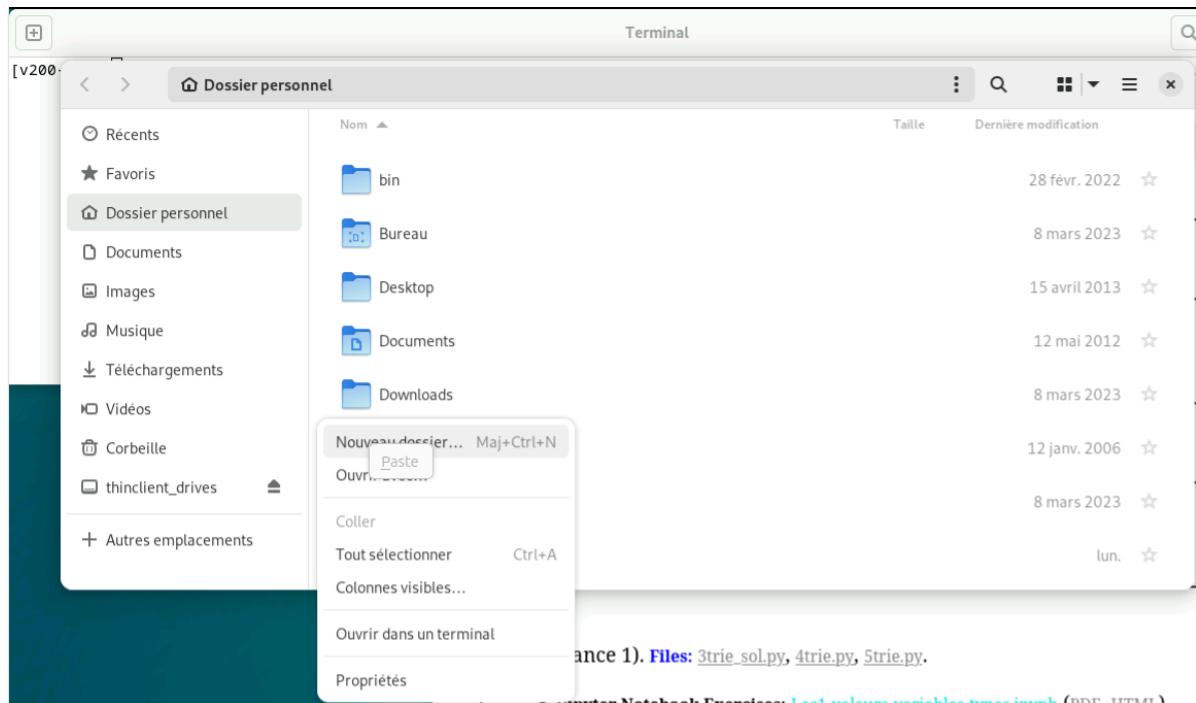


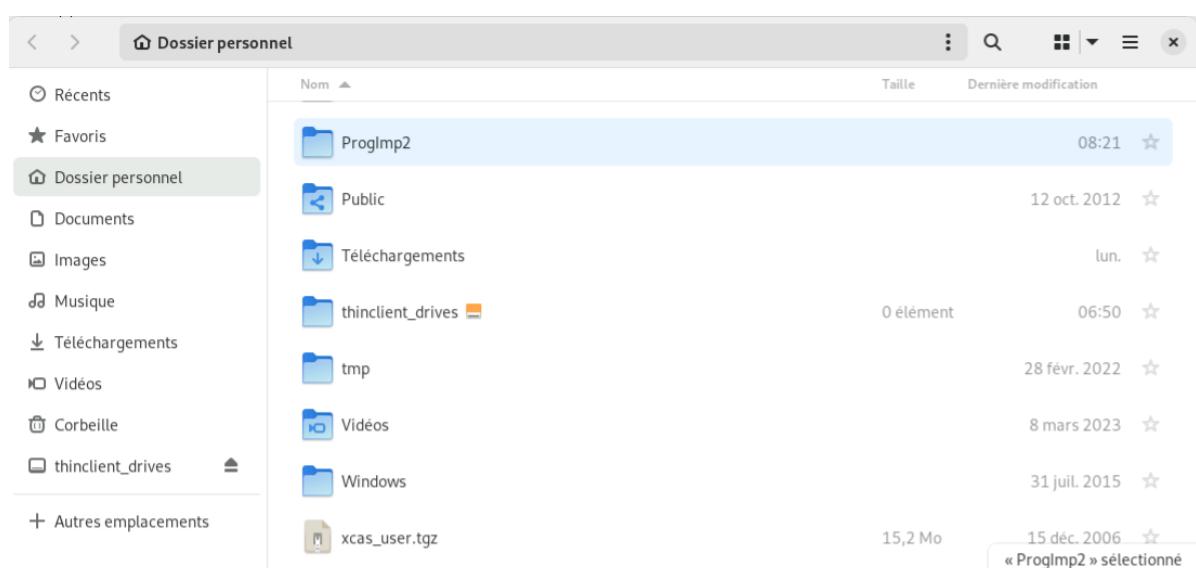
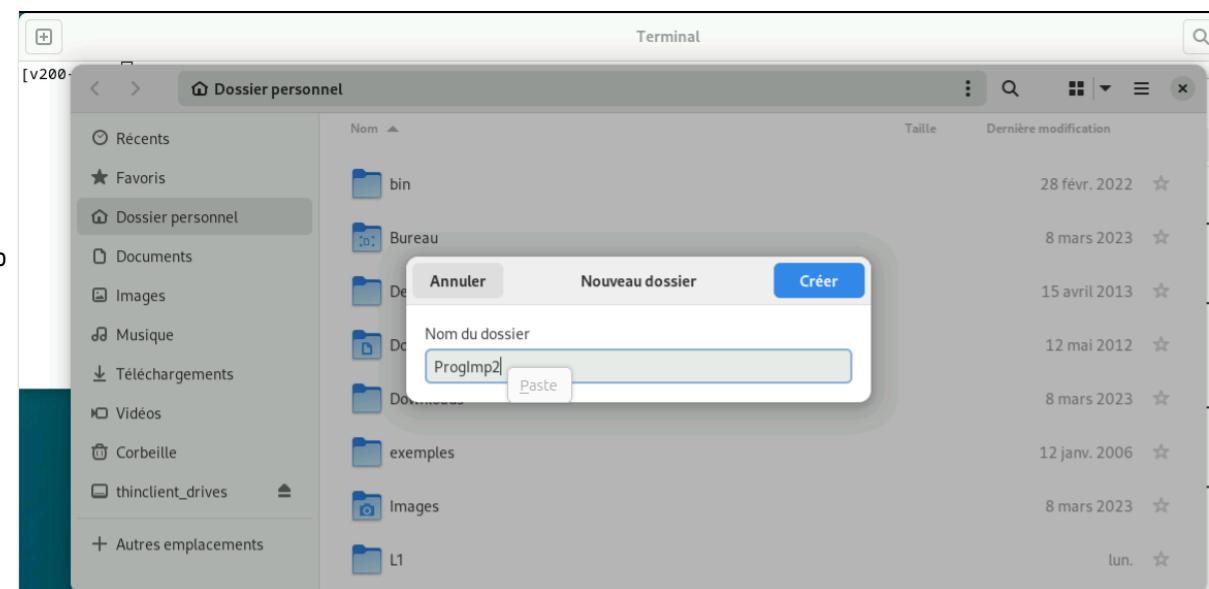
Fig. 8

Méthode 2: avec l'interface graphique (2/5)

Avec cette application, on va tout d'abord créer le répertoire ProgImp2 dans le "Dossier personnel" en cliquant avec le bouton droit dans une "zone libre" de la partie droite de l'application (Fig. 9) pour faire apparaître le menu contextuel, puis "Nouveau dossier..."



On tape : ProgImp2 (Fig. 10) puis bouton créer. On obtient alors la Fig. 11



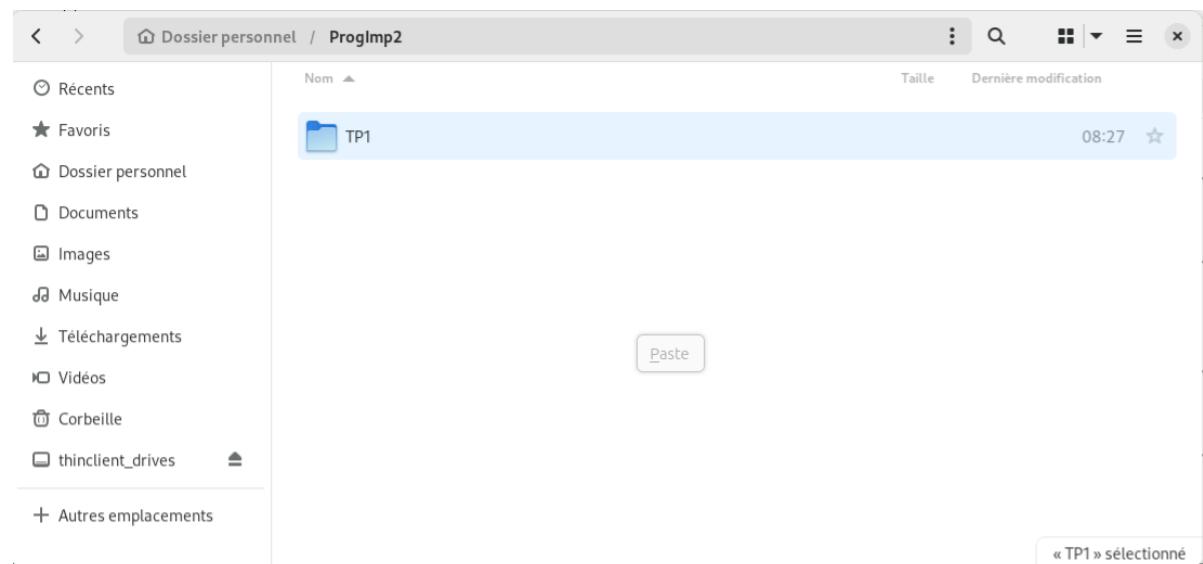
Méthode 2: avec l'interface graphique (3/5)

Pour créer le répertoire TP1 dans ProgImp2, on se déplace dans ce dernier en double cliquant sur le répertoire ProgImp2 que l'on vient de créer (résultat en Fig. 12) puis on crée le répertoire TP1 (bouton droit, "Nouveau dossier") pour obtenir la Fig. 13

Fig. 12



Fig. 13



Méthode 2: avec l'interface graphique (4/5)

Une fois le répertoire `~/ProgImp2/TP1` créé, on peut récupérer les fichiers de la page web.

Pour cela, bouton droit (menu contextuel) sur le fichier à récupérer puis "Save Link As ..." (Fig. 14)

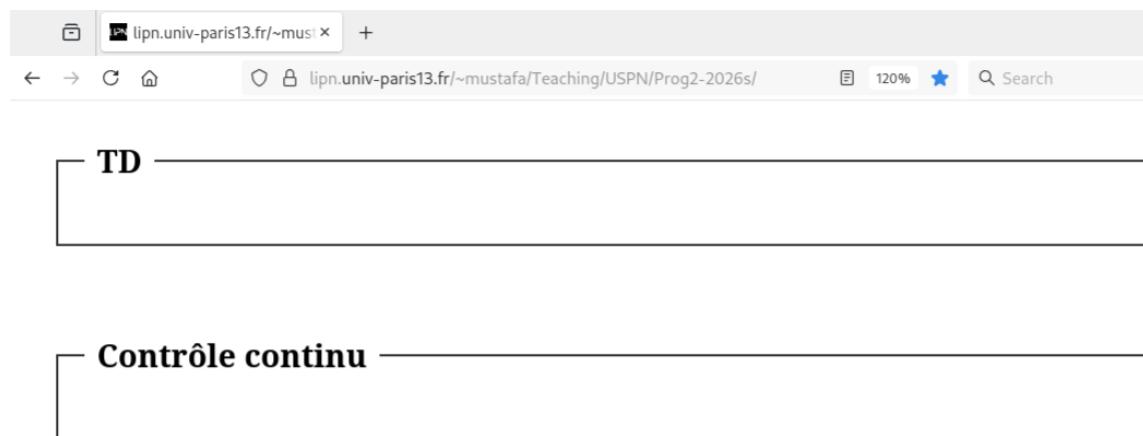
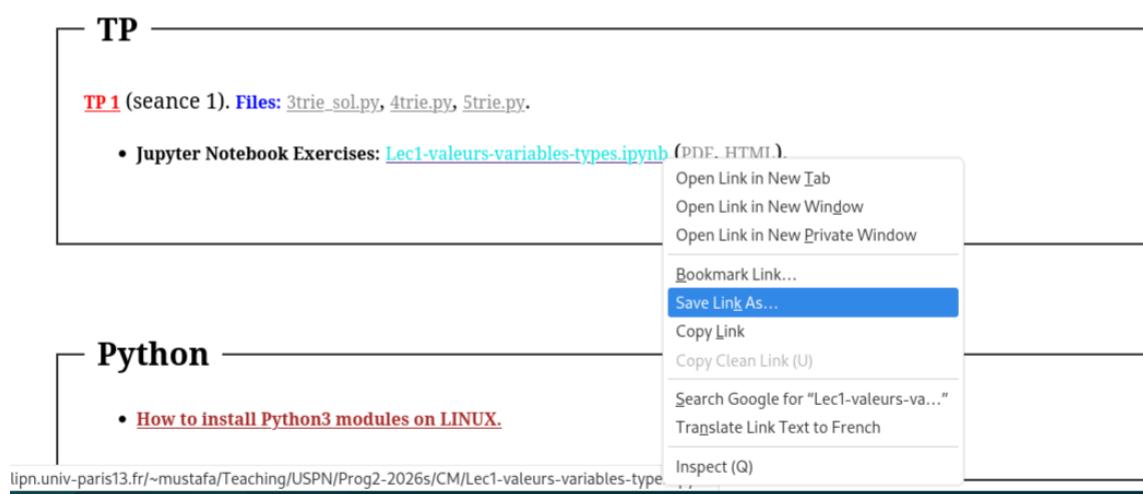


Fig. 14



Puis on se déplace dans le répertoire `~/ProgImp2/TP1` et en clique sur le bouton "Enregistrer" (Fig. 15)

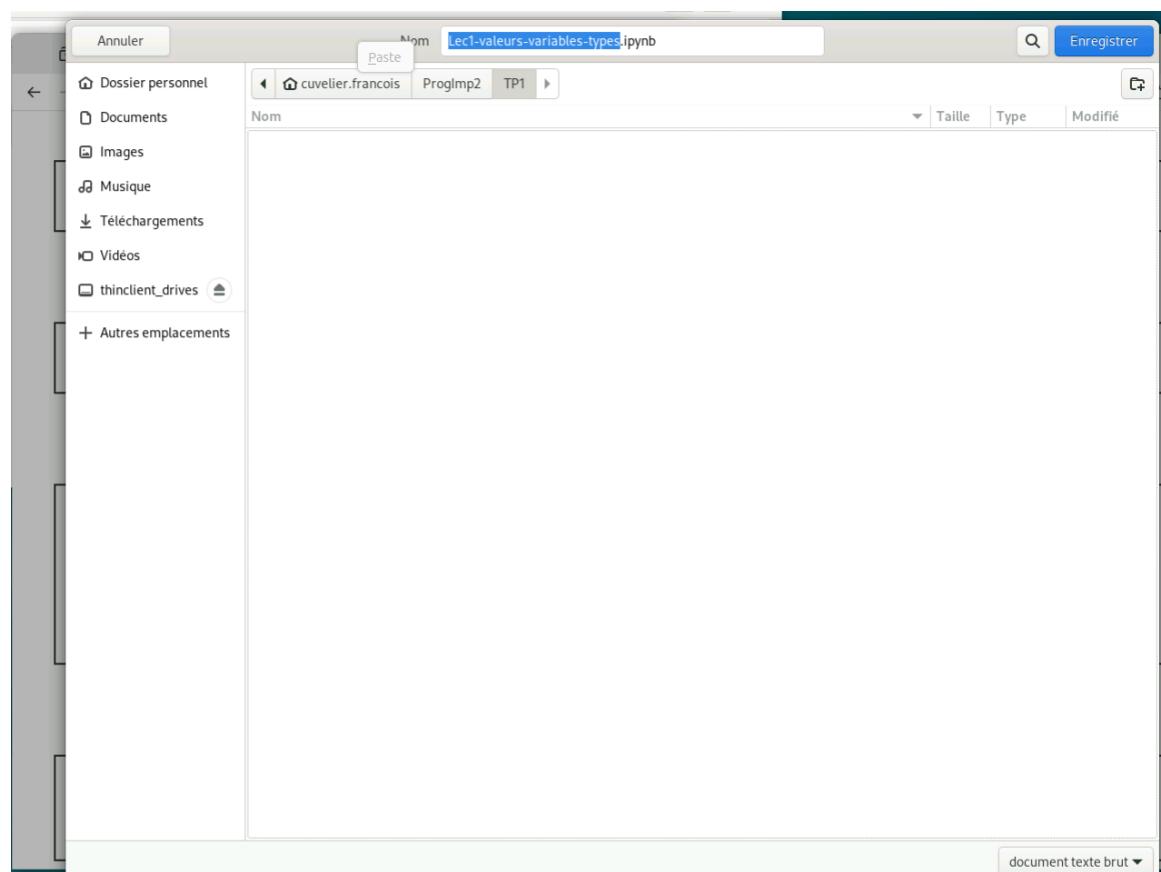
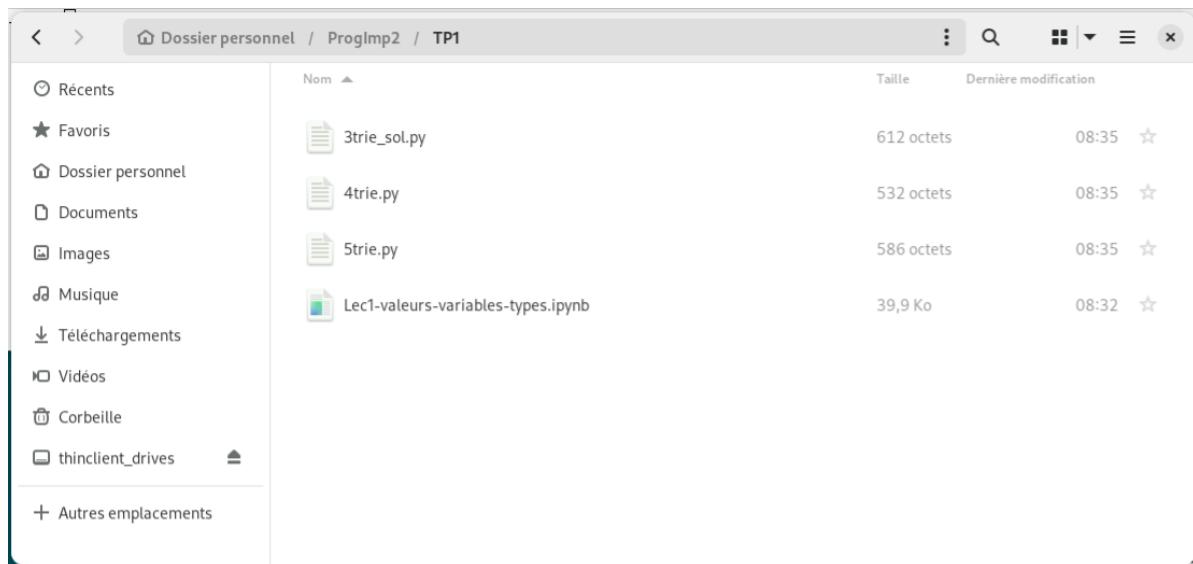


Fig. 15

Méthode 2: avec l'interface graphique (5/5)

On réitère sur l'ensemble des fichiers et au final on doit avoir 4 fichiers dans le répertoire ~/ProgImp2/TP1 (Fig. 16)

Fig. 16



2. Execution du "Jupyter Notebook"

Pour cet exemple (TP1), un Jupyter Notebook est disponible et il a été téléchargé dans le répertoire ~/ProgImp2/TP1

Le plus "simple" pour l'utiliser et d'utiliser l'application "Fichiers" (cliquez sur "Activités" en haut à gauche puis sur "Fichiers" (Fig. 17)

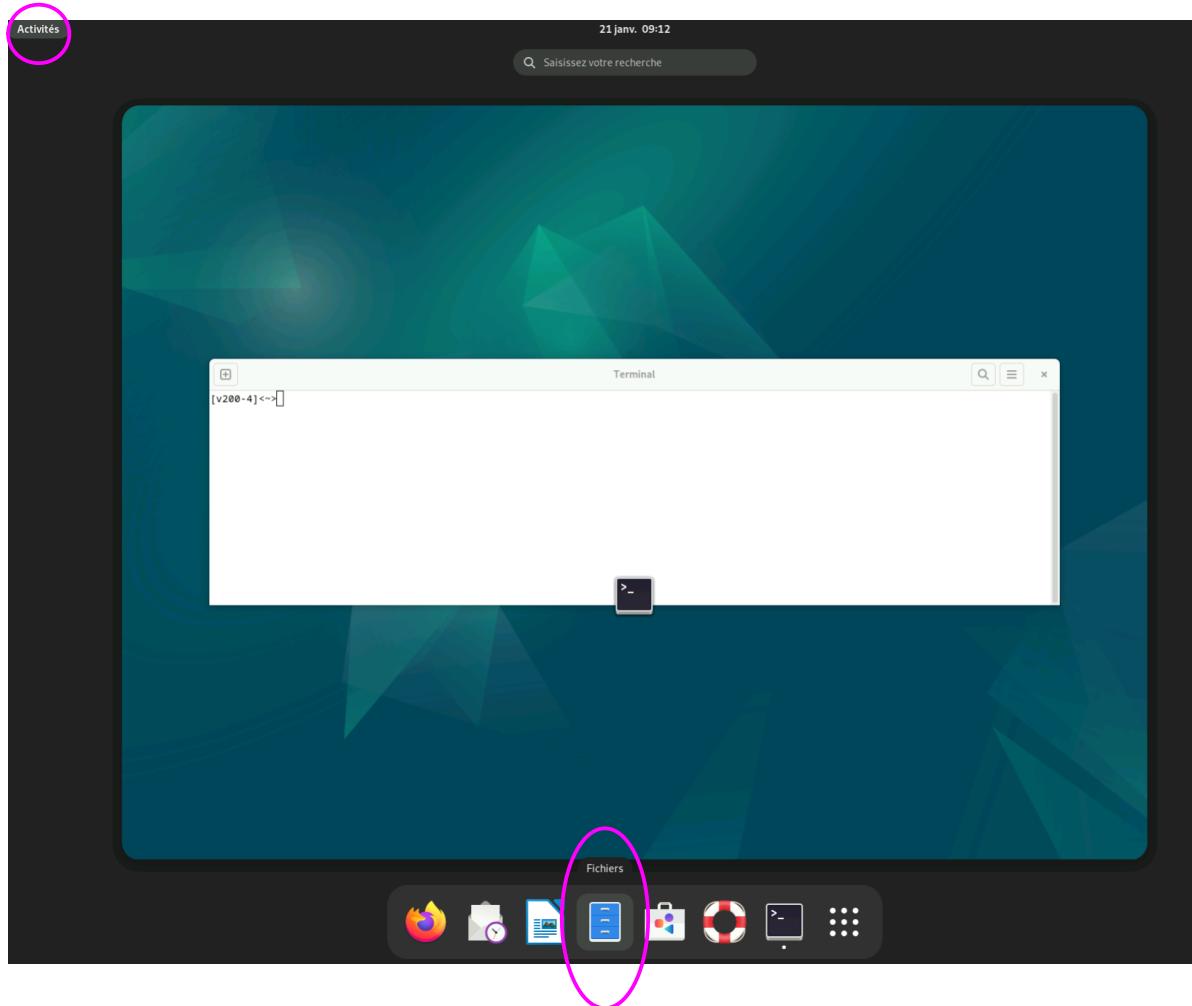


Fig. 17

Ensuite, on doit aller dans le répertoire ~/ProgImp2/TP1 où se situe le fichier. Pour cela on double clique sur ProgImp2 (Fig. 18) puis sur TP1 (Fig. 19)

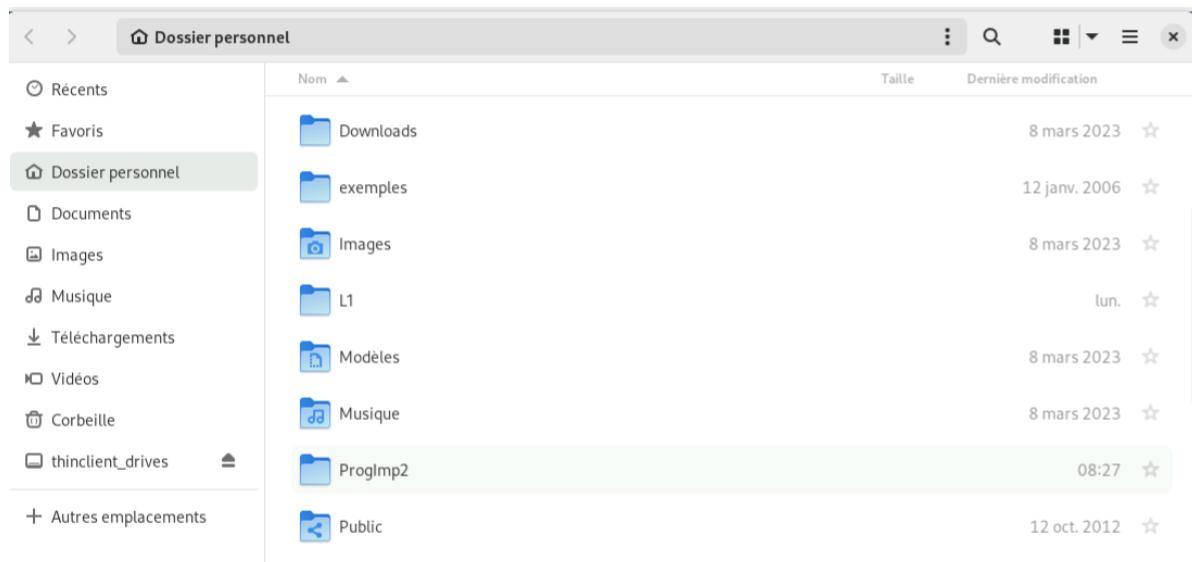
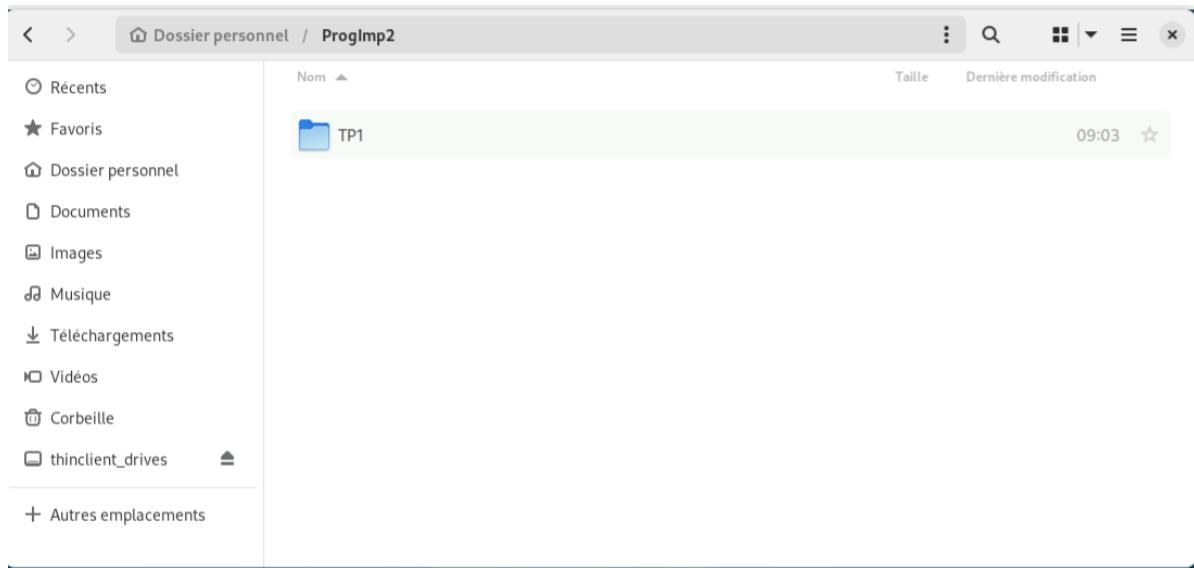


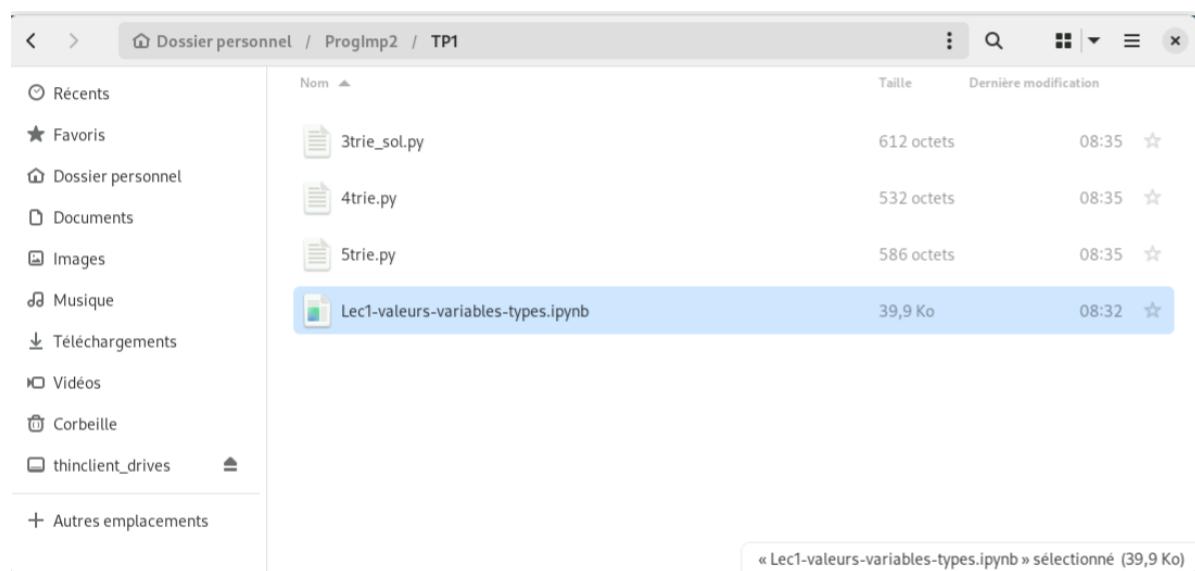
Fig. 18

Fig. 19



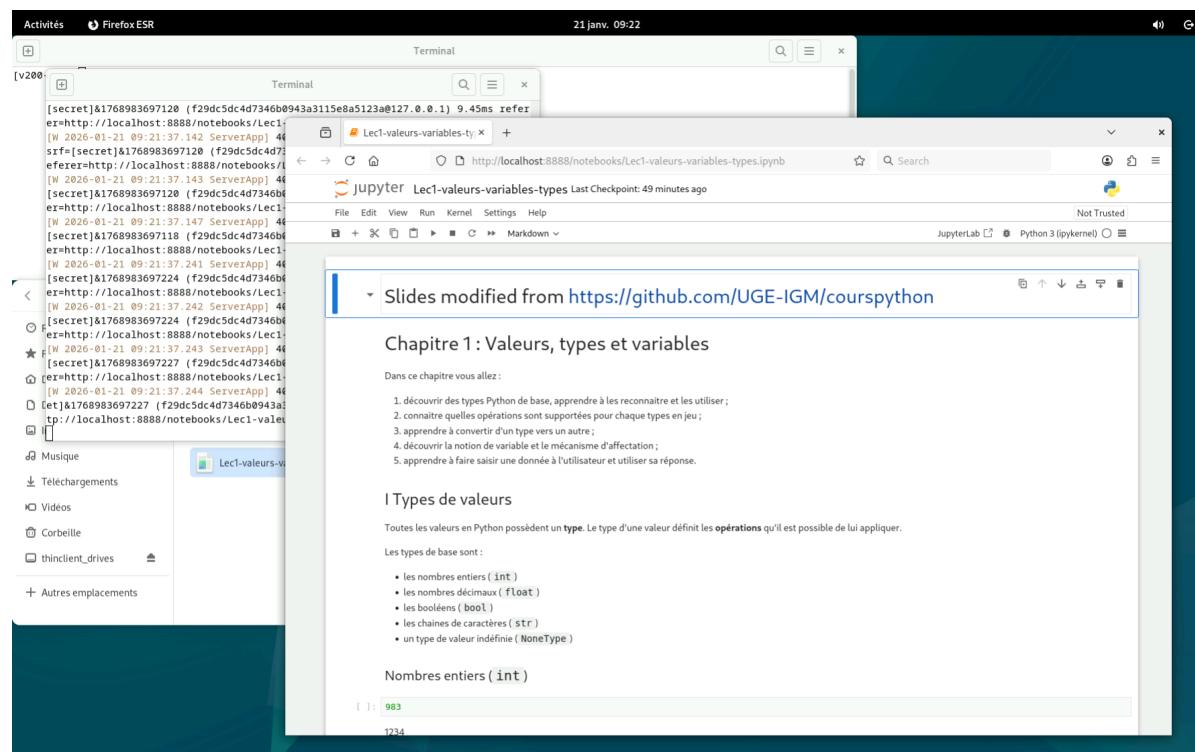
puis on "double clique" sur le fichier .ipynb ou "bouton droit/Ouvrir avec Jupyter Notebook" (Fig. 20)

Fig. 20



Cela ouvre un terminal et firefox avec le notebook prêt à être utilisé (Fig. 21)

Fig. 21



3. Execution d'un programme Python

Pour cet exemple (TP1), un programme complet est disponible et il a été téléchargé dans le répertoire `~/ProgImp2/TP1` et il se nomme `3trie_sol.py`

Pour l'executer, on va ouvrir un terminal (Fig. 22) et se placer dans le répertoire `~/ProgImp2/TP1` (Fig. 23)

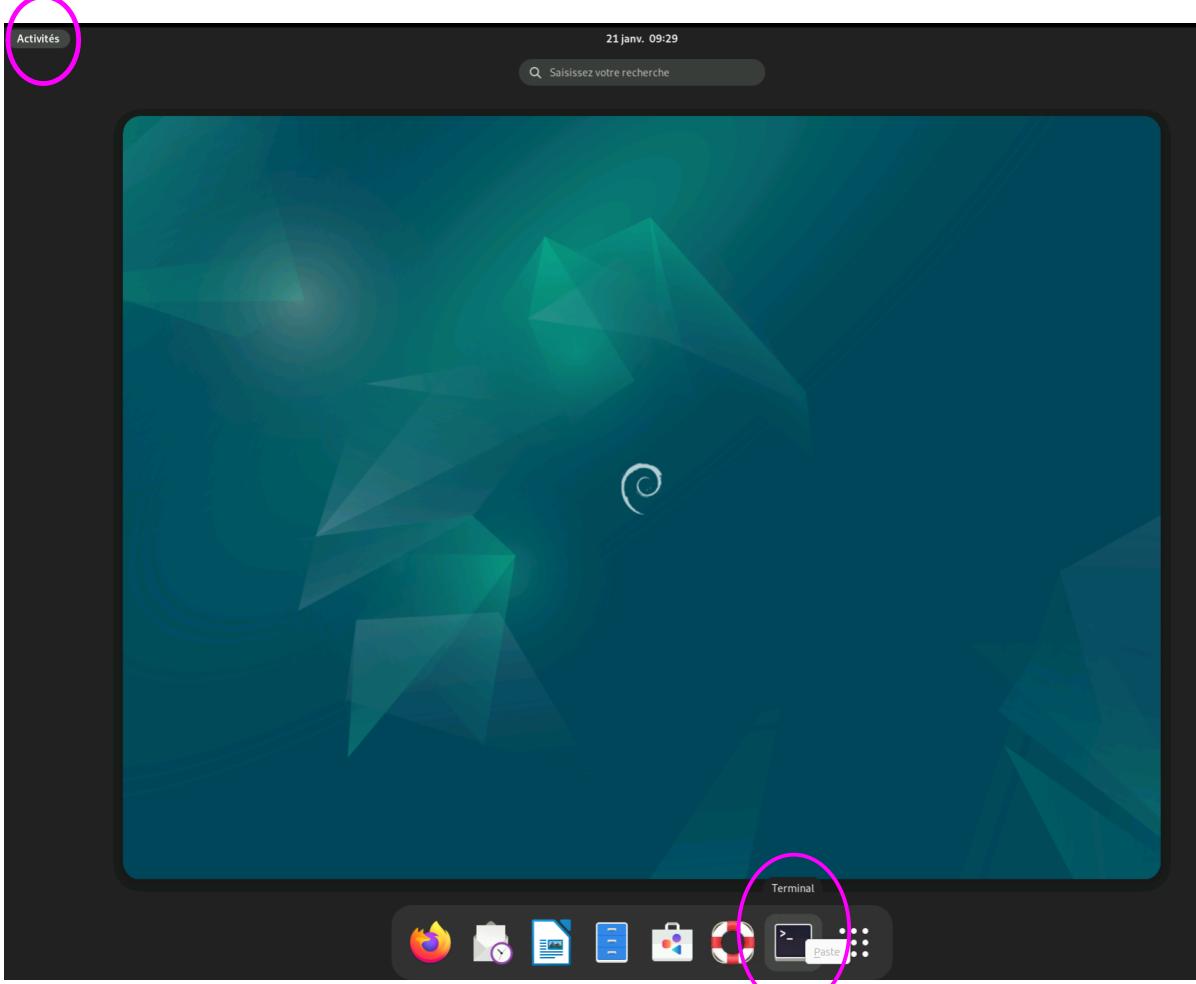


Fig. 22

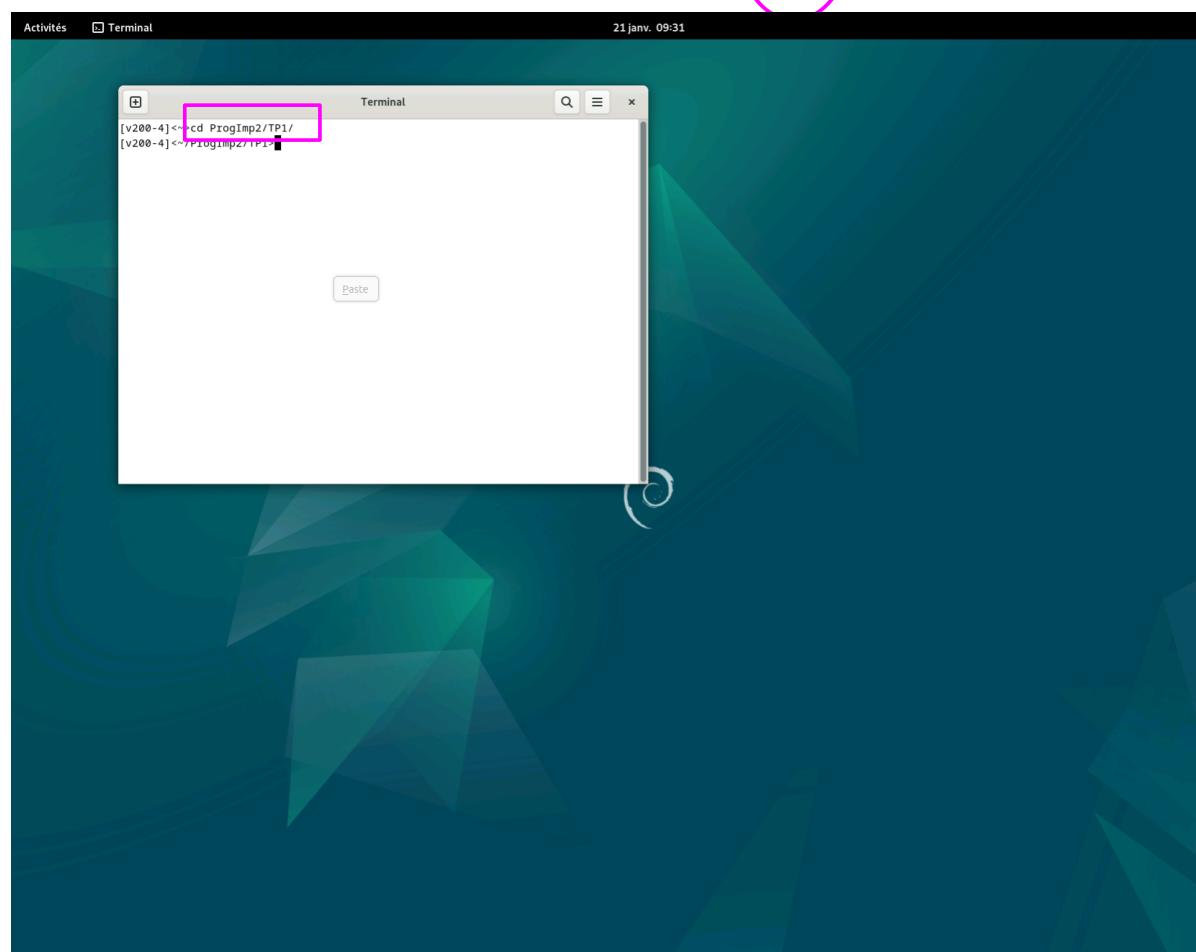
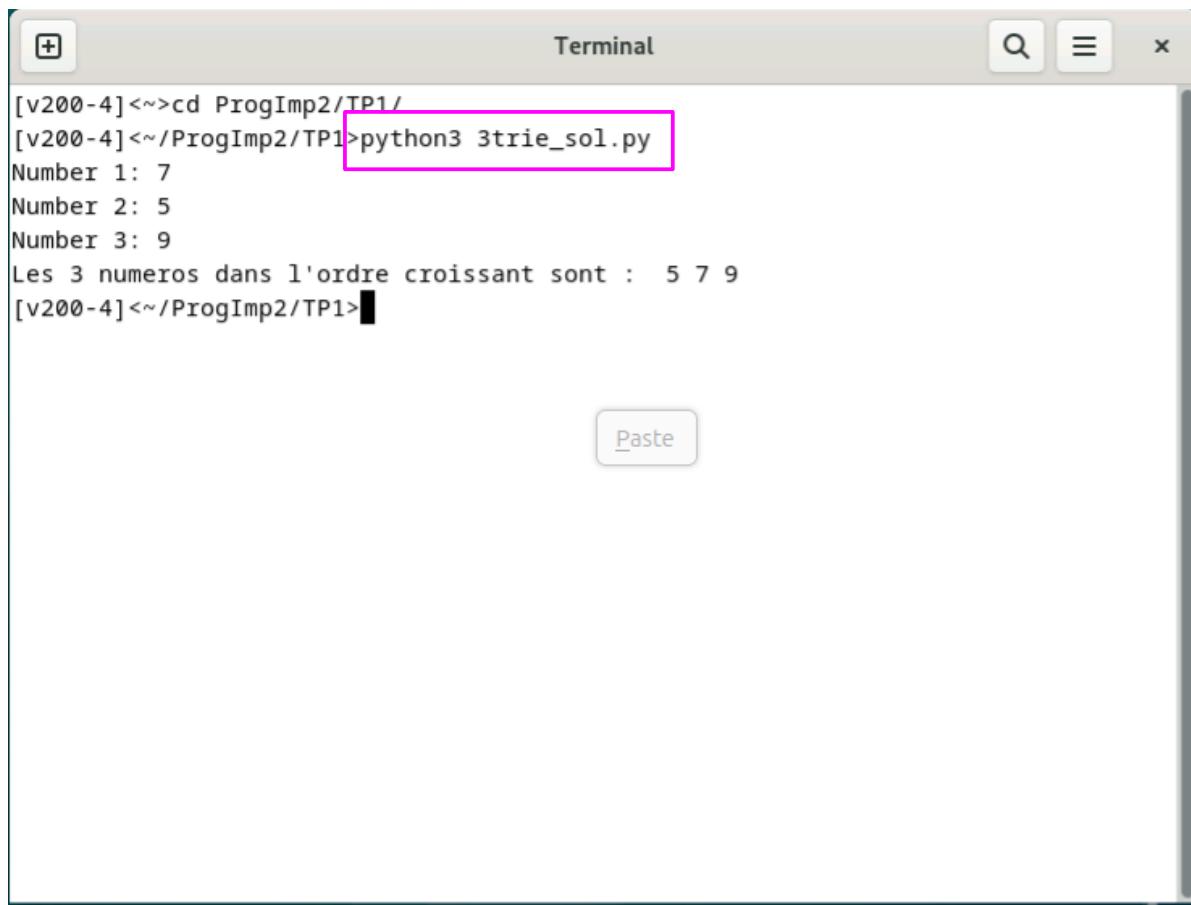


Fig. 23

Pour executer le programme, on tape dans le terminal

```
python3 3trie_sol.py
```

suivi de "Entrée/Return" (Fig. 24)



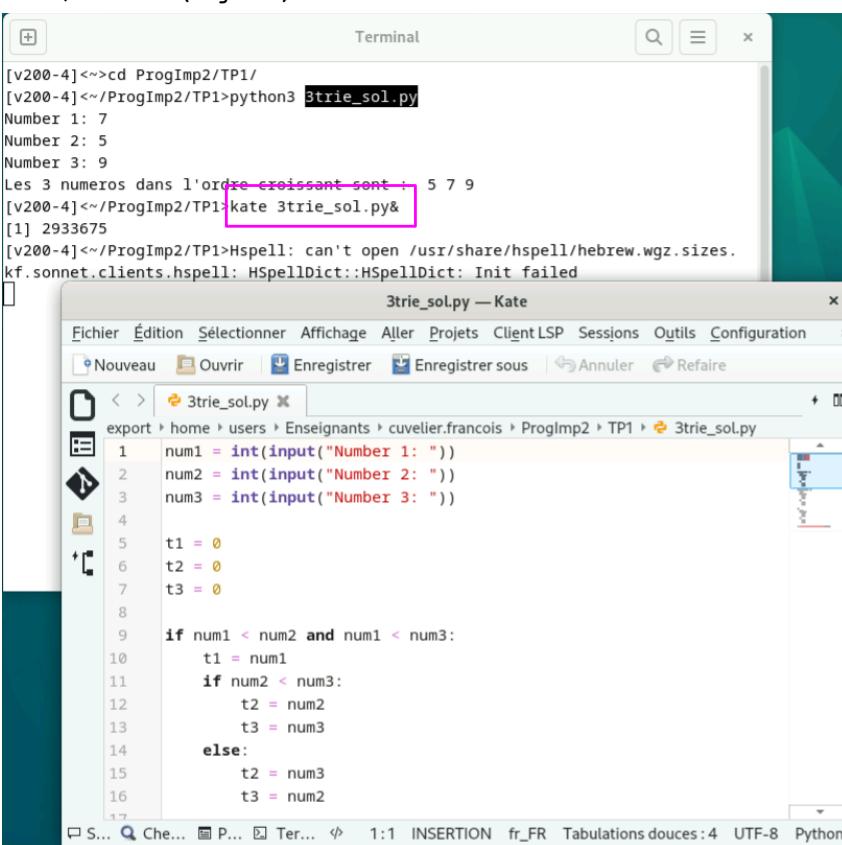
```
[v200-4] <~> cd ProgImp2/TP1/
[v200-4] <~/ProgImp2/TP1> python3 3trie_sol.py
Number 1: 7
Number 2: 5
Number 3: 9
Les 3 numeros dans l'ordre croissant sont : 5 7 9
[v200-4] <~/ProgImp2/TP1>
```

Fig. 24

Pour éditer le fichier, on peut, à partir du terminal, taper la commande

```
kate 3trie_sol.py&
```

suivi de "Entrée/Return" (Fig. 25)



```
[v200-4] <~> cd ProgImp2/TP1/
[v200-4] <~/ProgImp2/TP1> python3 3trie_sol.py
Number 1: 7
Number 2: 5
Number 3: 9
Les 3 numeros dans l'ordre croissant sont : 5 7 9
[v200-4] <~/ProgImp2/TP1> kate 3trie_sol.py&
[1] 2933675
[v200-4] <~/ProgImp2/TP1> Hspell: can't open /usr/share/hspell/hebrew.wgz.sizes.
kf.sonnet.clients.hspell: HSpellDict::HSpellDict: Init failed
```

3trie_sol.py — Kate

```
1 num1 = int(input("Number 1: "))
2 num2 = int(input("Number 2: "))
3 num3 = int(input("Number 3: "))

4
5 t1 = 0
6 t2 = 0
7 t3 = 0

8
9 if num1 < num2 and num1 < num3:
10     t1 = num1
11     if num2 < num3:
12         t2 = num2
13         t3 = num3
14     else:
15         t2 = num3
16         t3 = num2
```

Fig. 25