

THÈME 10: LECTURE/ECRITURE DANS UN FICHER

Introduction aux fichiers

- Jusqu'à présent, nous avons utilisé `input()` et `print()` pour lire les entrées du programme, et afficher les résultats obtenus.
- Parfois, les données d'entrée sont stockées dans un fichier et on aimerait pouvoir y accéder directement, sans les saisir manuellement au clavier.
- Il est aussi souvent utile de sauvegarder nos résultats dans des fichiers afin de pouvoir y accéder plus tard. Exemple: Sauvegarde d'une partie dans un jeu vidéo.
- En python, il est très facile de lire et d'écrire des données dans des fichiers.

Accès à un fichier texte

- Avant de commencer la lecture d'un fichier, il faut d'abord **l'ouvrir**. Ouvrir un fichier veut simplement dire que l'on crée une variable qui permet de le manipuler.
- La fonction **open()** est utilisée pour ouvrir un fichier. Par exemple, pour ouvrir le fichier appelé « data.txt », il suffit de faire:

```
>>> f=open('data.txt')
```

- Par défaut, open() ouvre un fichier en mode « **lecture** », c'est-à-dire qu'on ne peut pas modifier son contenu.
- Si on tente d'ouvrir un fichier inexistant en mode « lecture », on reçoit une erreur:

```
>>> f = open('toto') # le fichier 'toto' n'existe pas
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
IOError: [Errno 2] No such file or directory: 'toto'
```

Lecture d'un fichier texte

- Une fois notre fichier texte ouvert, il existe plusieurs manières de lire son contenu:

- Lire en une fois tout le contenu dans une chaîne de caractères:

```
texte = f.read() # la chaîne texte contient tout le texte du fichier
```

- Lire en une fois toutes les lignes du fichier dans une liste de chaînes de caractères:

```
lignes = f.readlines() # lignes est une liste qui contient les lignes du fichier: lignes[0] contient la première ligne, etc.
```

- Lire le fichier ligne par ligne dans une boucle **for**:

```
for ligne in f:  
    print(ligne) # affiche une ligne du fichier
```

Écriture dans un fichier texte

- Pour pouvoir écrire dans un fichier, il faut l'ouvrir en mode écriture:
`f = open('fichier.txt', 'w')`
- Si le fichier ouvert en mode « écriture » n'existe pas, il sera créé. S'il existe déjà, tout son contenu sera effacé.
- La fonction permettant d'écrire dans un fichier texte est **write()**.
`f.write('ce texte sera écrit dans le fichier')`
- Contrairement à `print()`, la fonction `write()` ne saute pas de ligne automatiquement. Pour sauter de ligne dans le fichier, il faut écrire un saut de ligne manuellement:
`f.write('\n')` # ceci permet de sauter la ligne
- L'argument passé à `write()` doit obligatoirement être une chaîne de caractères. Pour écrire un entier ou un autre type, il faut le convertir en chaîne de caractères en utilisant **str()**.

Fin de la manipulation

- Une fois la lecture/écriture terminée, il faut *fermer* le fichier en utilisant la fonction `close()`.

```
f = open('fichier.txt')
for ligne in f:
    print('une ligne lue :', ligne)
f.close()
```

Exemple de lecture

- Soit le fichier « nombres.txt » ci-contre qui contient des entiers (un par ligne).
- On veut calculer la somme de ces entiers:

```
fichier = open('nombres.txt')
somme = 0
for nombre in fichier:
    # nombre est une chaine, ne pas
    # oublier de la convertir en entier !
    somme = somme + int(nombre)
print(somme)
fichier.close()
```

nombres.txt
15
18
30
55
16
3
12
13

- Ces slides ont été réalisés par:
 - Amir Charif
 - Lydie Du Bousquet
 - Aurélie Lagoutte
 - Julie Peyre
- Leur contenu est placé sous les termes de la licence **Creative Commons CC BY-NC-SA**

