

PYTHON

Présenté par :
Xavier TABUTEAU

Les fichiers

Les fichiers

- **Le module OS**

Le module OS contient de nombreuses fonctions intéressantes pour l'accès au système d'exploitation.

```
>>> import os
>>> os.getcwd()
>>> os.chdir("C:\\Users\\User\\Documents")
>>> os.path.dirname(__file__)
```

- **Les fichiers**

Avec Python, l'accès aux fichiers est assuré par l'intermédiaire d'un objet-interface particulier, que l'on appelle objet-fichier. On crée cet objet à l'aide de la fonction intégrée `open()`. Celle-ci renvoie un objet doté de méthodes spécifiques, qui vous permettrons de lire et écrire dans le fichier.

Ecriture

```
>>> objetfichier = open('Monfichier','a')
>>> objetfichier.write('Bonjour, fichier !')
>>> objetfichier.write("Quel beau temps!")
>>> objetfichier.close()
```

Lecture

```
>>> of = open('Monfichier', 'r')
>>> t = of.read()
>>> print(t)
Bonjour, fichier !Quel beau temps !
>>> of.close()
```

Les fichiers

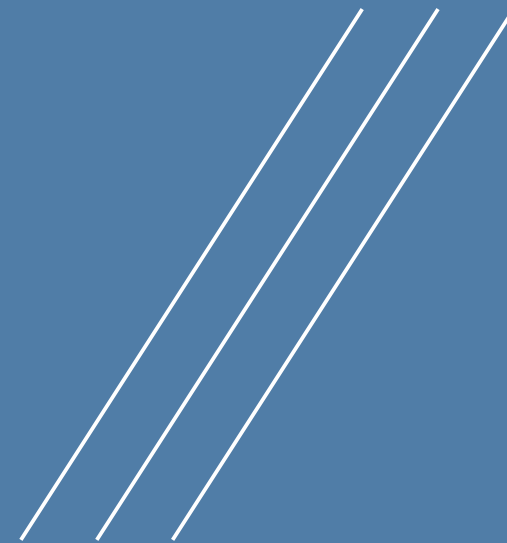
Les fichiers

La fonction `open()` attend deux arguments minimums, qui doivent tous deux être des chaînes de caractères. Le premier argument est le nom du fichier à ouvrir, et le second est le mode d'ouverture.

- 'a' : indique qu'il faut ouvrir ce fichier en mode « ajout » (append), ce qui signifie que les données à enregistrer doivent être ajoutées à la fin du fichier, à la suite de celles qui s'y trouvent éventuellement déjà.
- 'w' : utilisé aussi pour l'écriture mais lorsqu'on utilise ce mode « écriture » (write), Python crée toujours un nouveau fichier (vide), et l'écriture des données commence à partir du début de ce nouveau fichier. S'il existe déjà un fichier de même nom, celui-ci est effacé au préalable.
 - La méthode `write()` réalise l'écriture proprement dite. Les données à écrire doivent être fournies en argument. Chaque nouvel appel de `write()` (en mode a) continue l'écriture à la suite de ce qui est déjà enregistré.
- 'r' : Ouverture en lecture.
 - La méthode `read()` (en mode r) permet de lire le contenu d'un fichier dans son ensemble.
 - La méthode `readline()` (en mode r) lit qu'une ligne par appel de cette instruction.
 - La méthode `readlines()` (en mode r) lit les lignes individuellement dans une liste.
 - La méthode `seek(x)` (en mode r) permet de remplacer le curseur à la position x voulue.
 - La méthode `close()` referme le fichier dans n'importe qu'elle mode.
- 'r+' : c'est un mode lecture et écriture. Le fichier doit exister au préalable sinon une exception se produit. Le curseur est placé en début de fichier et le fichier n'est pas effacé.

`module_os.py`
`fichier_texte.py`
`ex_fichier`
`ex_multi_fichiers`
`ex_fichier_mini_bdd`

PYTHON



Présenté par
Xavier TABUTEAU