

Les fichiers

Pour manipuler les fichiers en Python Il y a un type spécial et on crée un objet de ce type de la façon suivante :

```
f = open(<ref> [, mode=<opt>, encoding=<encodage>])
```

Les différents modes possibles:

- "r" ouverture en mode lecture. C'est le mode par défaut (si on indique le mode d'ouverture).
- "w" ouverture en mode écriture.
- "a" ouverture en mode écriture à la fin du fichier.

L'encodage peut-être modifier (par défaut celui du shell dans lequel est lancé l'interpréteur python) ex : `encoding='utf-8'`

```
f = open('mon_fichier', encoding='utf-8')
```

Si le fichier 'mon_fichier' n'existe pas ou si l'ouverture ne peut se faire open lève une exception de type IOError.

```
try:
    f = open('mon_fichier')
except IOError:
    print("Erreur ouverture fichier")
```

Mais Pour l'ouverture en écriture si le fichier n'existe pas, il est créé

```
f = open('mon_fichier', mode='w')
```

Si le fichier est ouvert en mode 'r' ou 'w' l'objet de type file f pointe alors au début du fichier. Pour l'ouverture avec le mode 'a' f pointe alors à la fin du fichier

```
f = open("fichier.txt")
```

Lecture

Pour lire, il y a les méthodes : `read()`:

```
f.read(n)
```

On lit les n char(octets) à partir de la position du curseur et le place ensuite n char(octet) plus loin

readline() :

```
f.readline()
```

On lit une ligne de la position du curseur jusqu'à la fin de la ligne (\n)

readlines()

```
for ligne in readlines():  
    print(ligne)
```

La fin de la boucle le curseur est placé a la fin du fichier

Écriture

```
f = open("fichier.txt", "w")  
  
f.write("blabla")    # ecrit la chaine après la position du curseur et la modifie  
f.write(str())
```

Curseur

seek:

La methode seek() permet changer la position du curseur

```
seek(<position en nombre d'octet> [, SEEK_* ])
```

Change la position du curseur en lui donnant un nombre d'octet. Le déplacement est calculer relativement par rapport a la contante

Contantes

SEEK_SET or 0 : Début du fichier (par default); la position doit être 0 ou positive

SEEK_CUR or 1 : Position courrante du curseur; position positive ou négative

SEEK_END or 2 : Fin du fichier; position généralement négative

retourne la nouvelle postition du curseur

tell:

La methode tell() permet connaitre la position du curseur

ATTENTION

Ne pas oublier de fermer le fichier

f.close()