

### ➤ Ouverture et fermeture

*open(file, mode)* Ouvre le fichier dans un des modes suivant :

r	(read) lecture seule
w	(write) écriture
a	(append) ajout

**Exemple** : `fichier = open("essai.txt", "r")`

Attention :

- Les modes "w" et "a" ne permettent pas de lire le contenu du fichier.
- Le mode "r" ne permet pas de modifier le fichier.
- Si le fichier n'existe pas, le mode "r" renvoie une erreur alors que les modes.
- Si le fichier n'existe pas, les modes "w" et "a" créent le fichier sur le disque.
- Si le fichier existe déjà, le mode "w" va l'effacer et en créer un nouveau.

*close()* Ferme le fichier.  
**Exemple** : `fichier.close()`

Remarque : Pour gérer plus facilement les fichiers sur le disque dur, il existe des modules complémentaires nommés `os` et `os.path`. Le plus simple est de commencer par sauvegarder le programme en cours pour connaître le répertoire courant utilisé !

Les fichiers sont des objets, c'est-à-dire que l'on peut leur appliquer une méthode : il suffit d'indiquer le nom du fichier et la méthode à lui appliquer, séparés par un point comme dans : `fichier.close()`

### ➤ Lecture et écriture

*read()* Lit tout le contenu du fichier et le renvoie dans une chaîne de caractères ou uniquement les n premiers caractères si on précise une valeur n entière.  
**Exemple** : `a = fichier.read(10)`  
 Remarque : Si on enchaîne plusieurs « *read()* », à chaque fois on reprend la lecture à l'endroit où s'est arrêté le « *read()* » précédent

*readline()* Renvoie le contenu de la ligne suivante complète du fichier.  
**Exemple** : `a = fichier.readline()`

*readlines()* Renvoie le contenu de chaque ligne du fichier dans une liste  
**Exemple** : `( "la première ligne", "la deuxième ligne", "la troisième ligne" )`

*write()* Permet d'écrire une chaîne de caractère dans le fichier à la suite de la précédente.  
**Exemple** : `fichier.write("bonjour")`

Attention : Il faut penser à gérer les retours à la ligne en lecture et en écriture...

En lecture on peut les supprimer en indiquant :

`fichier.readline().replace("\n", "")` : Remplace tous les sauts de ligne par un vide.

`fichier.readline().strip("\n")` : Supprime tous les sauts de ligne.

En écriture on doit les ajouter à la fin de chaque ligne en indiquant :

`fichier.write("bonjour.\n")` : Ecrit « bonjour. » avec un saut de ligne à la fin.