



CHAPITRE 7 : LES FICHIERS

I. Introduction :

Nous avons déjà eu l'occasion d'étudier les « entrées-sorties conversationnelles » (*Clavier et Ecran*), c'est-à-dire les fonctions permettant d'échanger des informations entre le programme et l'utilisateur. Dans ce chapitre on va étudier les fonctions permettant au programme d'échanger des informations avec des **fichiers**.



II. Notion de fichier :

Les données stockées en mémoire en utilisant les variables, sont perdues dès la sortie du programme. Les fichiers stockés sur un support physique (disque dur, clé USB, CD, DVD, carte mémoire SD...), constituent par contre des moyens de conservation à long terme des données produites par un programme.

III. Types de fichiers selon le format du contenu :

- **Les fichiers textes** : l'information est stockée sous forme de caractères lisibles par un éditeur de texte (principalement des lettres et des chiffres). Ils se manipulent ligne par ligne (ou caractère par caractère).
- **les fichiers binaires** : leur contenu est uniquement constitué de 0 et 1 (Fichiers multimédias : images, sons, vidéos).

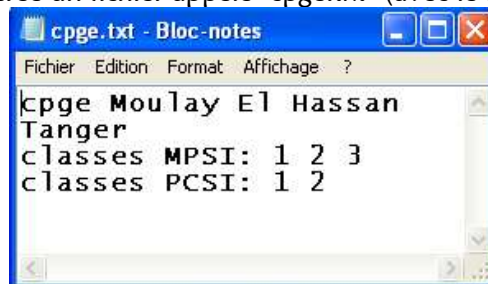
IV. MANIPULATION DES FICHIERS TEXTES:

Les principales opérations possibles avec les fichiers:

- Ouvrir un fichier
- Créer un fichier
- Ecrire dans un fichier
- Lire un fichier
- Fermer un fichier

IV-1. Ouvrir un fichier pour lire le contenu :

Imaginons que vous ayez créé un fichier appelé "cpge.txt" (avec le bloc note par exemple).



Pour lire ce fichier, il faut d'abord l'ouvrir ; voici la syntaxe :



```
fichier = open("cpge.txt","r")
```

- **"cpge.txt"** : est une chaîne de caractères représentant le chemin du fichier à ouvrir. Si votre fichier est dans même dossier que votre programme, le chemin donné ici est suffisant. Sinon, mettez le chemin absolu, par exemple : "C:/cours/cpge.txt"
- **"r"** : le "r" signifie **"read"** .C'est une chaîne de caractères représentant l'ouverture en lecture ; le fichier doit exister, le pointeur est positionné au début du fichier. Sur ce fichier seules les opérations de lecture sont permises.

On peut ensuite lire le contenu du fichier de différentes manières :

chaîne = fichier.read() lit le fichier en intégralité et renvoie une chaîne de caractères.	p= open("cpge.txt","r") a=p.read() print(a)	le script affiche : cpge Moulay El Hassan Tanger classes MPSI: 1 2 3 classes PCSI: 1 2
chaîne = fichier.read(n) lit n caractères du fichier à partir de la position courante et renvoie une chaîne de caractères.	p= open("cpge.txt","r") a=p.read(6) print(a)	le script affiche : cpge M
ligne = fichier.readline() lit une seule ligne à partir de la position courante y compris le '\n' qui est le caractère de fin de ligne ; retourne une chaîne vide quand la fin du fichier est atteinte ;	p= open("cpge.txt","r") a=p.readline() print(a) print(len(a))	le script affiche : cpge Moulay El Hassan 22
liste = fichier.readlines() retourne une liste de toutes les lignes du fichier.	p= open("cpge.txt","r") a=p.readlines() print(a) print(len(a))	le script affiche : ['cpge Moulay El Hassan\n', 'Tanger\n', 'classes MPSI: 1 2 3\n', 'classes PCSI: 1 2'] 4

Après l'utilisation du fichier, il faut le fermer. Voici la syntaxe :

```
fichier.close()
```

IV-2. Ecrire dans le fichier :

Pour écrire dans un fichier, il faut également l'ouvrir ; il faut toutefois savoir si on veut écraser le fichier précédent (**"w"** comme "write") ou ajouter à la fin du fichier (**"a"** comme "append"). Dans les deux cas, si le fichier n'existe pas il sera créé.

Ensuite il suffit d'appliquer la méthode suivante pour écrire une ligne dans le fichier (avec ajout automatique d'un caractère de fin de ligne) :

```
fichier.write("mon texte ici")
```



Exemple :

```
p= open("cpge.txt","w")  
p.write("la programmation\n")  
p.write("python")  
p.close()
```

Dans le fichier cpge.txt on aura le contenu suivant :



Remarque :

On peut utiliser la méthode suivante pour écrire dans un fichier une liste de lignes :

fichier.writelines(liste)

Exemple :

```
p= open("cpge.txt","w")  
l=["cpge de tanger\n","Maroc\n"]  
p.writelines(l)  
p.close()
```

Dans le fichier cpge.txt on aura le contenu suivant :

