

# Base de données – manip1

## Objectifs de la manip :

- Création d'un dossier spécifique pour assurer une meilleure portabilité
- Utiliser la gestion de fichier en python pour permettre d'encoder des données.
- Accompagner chaque encodage d'un identifiant unique (avec un caractère dédié)
- Création de fichiers dérivés du fichier de base

## Notion utile et petits rappels :

- Création d'un dossier spécifique

Jusqu'à maintenant nous avons précisé le dossier dans lequel créer son fichier (en donnant tout le chemin, par exemple "c:\python27").

Pour améliorer la portabilité du code, on choisira de travailler dans le dossier courant.

Comment créer un dossier dans le dossier courant et y mettre un fichier ?

```
os.mkdir("dossier_perso")
os.chdir("dossier_perso")
my_data_file=open("data_base.txt","w")
```

Mini exemple complet :

```
# -*- coding: cp1252 -*-
import os

if not os.path.isdir("dossier_perso"): #si le dossier n'existe pas
    os.mkdir("dossier_perso") #on crée le dossier dossier_perso au sein du dossier courant
os.chdir("dossier_perso") #on change le dossier courant en dossier_perso (créé précédemment)
my_data_file = open("data_base.txt","w") #on ouvre/crée un fichier data_base.txt en écriture
my_data_file.write("bonjour")
my_data_file.close()
```

- Quelques rappels (syntaxe ci-dessous et explications orales)

## **import os**

### **os.chdir("c:\directory")**

---->directory est le chemin ou doit être créé le fichier

### **fichier=open("nom\_fichier","argument")**

---->nom\_fichier est le nom du fichier que l'on veut utiliser

----> l'argument peut être w=write, r=read, a=append

### **fichier.write("texte")** ou **fichier.write(variable)**

---->variable doit être un string

### **fichier.close()**

---->ferme le fichier

## Base de données, énoncé de la manip 1 :

Créez sur votre ordinateur un nouveau dossier avec votre nom suivi de database1.  
Ce dossier contiendra tout ce qui concerne cette manip (tous les points).

- 1.) Lancez python 2.7, créez un nouveau fichier nommé encodage1.py que vous sauvez dans le dossier que vous venez de créer.

Complétez votre programme pour que son exécution fasse apparaître :

Veillez introduire un ID (3 chiffres) :

Veillez introduire votre nom :

Veillez introduire votre prenom :

Veillez introduire votre age :

Il doit apparaître dans votre dossier courant un dossier « data » qui contient le fichier « data\_base.txt » avec les données qui viennent d'être encodées compte tenu de :

Encadrez l'ID (nommez le « my\_id ») des caractères # (exemple : #134#)

Les différents éléments sont suivis d'un passage à la ligne

A la fin de la série d'encodage un 2<sup>ème</sup> passage à la ligne apparaît

Plusieurs séries d'encodage peuvent être mises au sein du même fichier.

- 2.) Toujours dans votre dossier de cette manip, créez un fichier encodage2.py et complétez-le afin qu'il ajoute des fonctionnalités par rapport au point précédent :  
Au départ il doit apparaître :

Si vous désirez faire un encodage, tapez 'o' sinon tapez 'n'

Tant que l'utilisateur choisit oui ('o') un encodage tel qu'en 1.) doit apparaître.

Dés que l'utilisateur choisit non ('n') l'encodage se termine et le programme doit :

Lire tout le fichier data\_base.txt et l'afficher

Afficher un passage à la ligne

Lire tout le fichier data\_base.txt et afficher tous les ID (un ID par ligne).

On peut bien sûr exploiter la manière dont les données sont dans le fichier :

Exemple avec 1 record	Caractéristiques des données dans le fichier
#123# Jo fugtniol 65	Un record commence par l'ID mis entre # et #    Un record se termine par un saut de ligne

Le premier # d'un record indique donc le début de l'ID

Le # suivant indique la fin de l'ID

Quelques pistes :

Pour savoir si on est dans la partie de l'ID on peut utiliser un flag

Pour reconnaître le caractère '#' on peut mettre deux FOR en cascade,

```
current_id = ''
reading_an_id = False
for a_line in my_file:
    for a_char in a_line:
        if reading_an_id:
            if a_char == '#': .. #then finish reading
```

- 3.) Toujours dans votre dossier de cette manip, créez un fichier id\_et\_nom.py et complétez-le afin qu'il parcoure le fichier « data\_base.txt » et qu'il constitue, toujours dans le dossier data, un fichier « id\_et\_nom.txt » avec les ID et les noms.
- 4.) Créez un fichier encodage3.py qui est similaire au fichier encodage1.py mais en s'assurant de ne pas encoder deux records avec le même ID.