

Chapitre – Données structurées et leur traitement



Repères historiques

Le métier Jacquard

En 1725, Basile Bouchon, un Lyonnais, met au point le premier système de programmation d'un métier à tisser grâce à un ruban perforé¹. En 1728, Jean-Baptiste Falcon, son assistant, remplace le ruban par une série de cartes perforées reliées entre elles¹.

Repères historiques

Le métier Jacquard

Joseph Marie Jacquard intègre les découvertes passées dans son métier à tisser qui fut adopté dans le monde entier à partir de 1801.

En 1834, Charles Babbage utilise les cartes du métier Jacquard pour donner des instructions et des données à sa machine analytique, l'ancêtre des ordinateurs.



1930: utilisation des cartes perforées, premier support de stockage de données;

1956: invention du disque dur permettant de stocker de plus grandes quantités de données, avec un accès de plus en plus rapide ;

1970: invention du modèle relationnel (E. L. Codd) pour la structuration et l'indexation des bases de données ;

1979: création du premier tableur, VisiCalc ;

2009: Open Government Initiative du président Obama ;

2013: charte du G8 pour l'ouverture des données publiques.

LES DONNEES

Les données constituent la matière première de toute activité numérique. Afin de permettre leur réutilisation, il est nécessaire de les conserver de manière persistante. Les structurer correctement garantit que l'on puisse les exploiter facilement pour produire de l'information.

I. Les données et l'information

1- Que sont les données?

Une donnée est en informatique la représentation d'une information, formatée de manière adaptée à sa nature et pour être exploitée par des applications. Toutes les informations humaines se codent en binaire ; bien entendu ce n'est pas l'objet réel, ce n'est que son reflet numérique.

RAPPEL : Pour connaître le type de donnée à laquelle on a affaire:
l'instruction `type()`-->

I. Les données et l'information

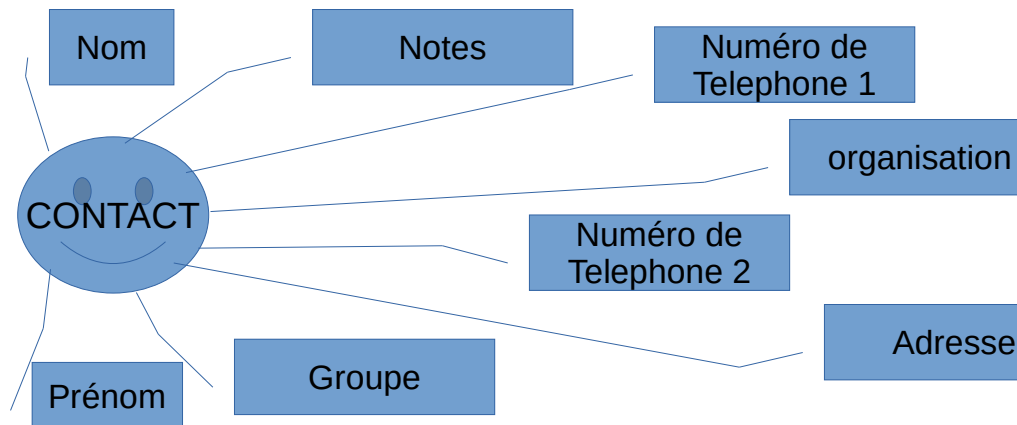
1- Que sont les données?

Comme les données peuvent être des objets relativement complexes, on utilise des **descripteurs** qui sont utiles pour décrire cet objet.

Petite application : prenons une donnée comme un contact (dans votre téléphone), combien de descripteurs pensez-vous devoir utiliser au minimum, lesquels?.

I. Les données et l'information

1- Que sont les données?



un certain nombre de descripteurs qui vont permettre à des programmes de pouvoir travailler avec, retrouver des informations.... Ces descripteurs se doivent donc d'être rangés toujours de la même façon!!! Pas comme votre chambre!

I. Les données et l'information

1- Que sont les données?

Une **collection** regroupe des objets (élémentaires ou complexes) partageant le même descripteur (par exemple, la collection des vidéos d'un Youtubeur aura un descripteur commun: leur auteur).

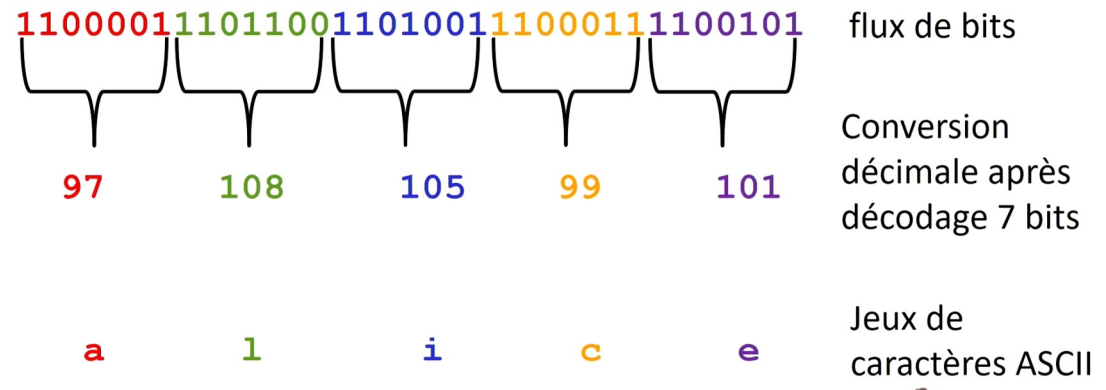
1. Les données et l'information

2- Les formats de données

une information est toujours codée sous forme d'une suite de bits (dans les machines, il n'y a que des bits, soit des séries de 0 et de 1)

Exemple : un codage possible des caractères : le code ASCII-American Standard Code for Information Interchange

Chaque octet commence par 0 dans ce code, on ne représente pas le premier bit de l'octet...

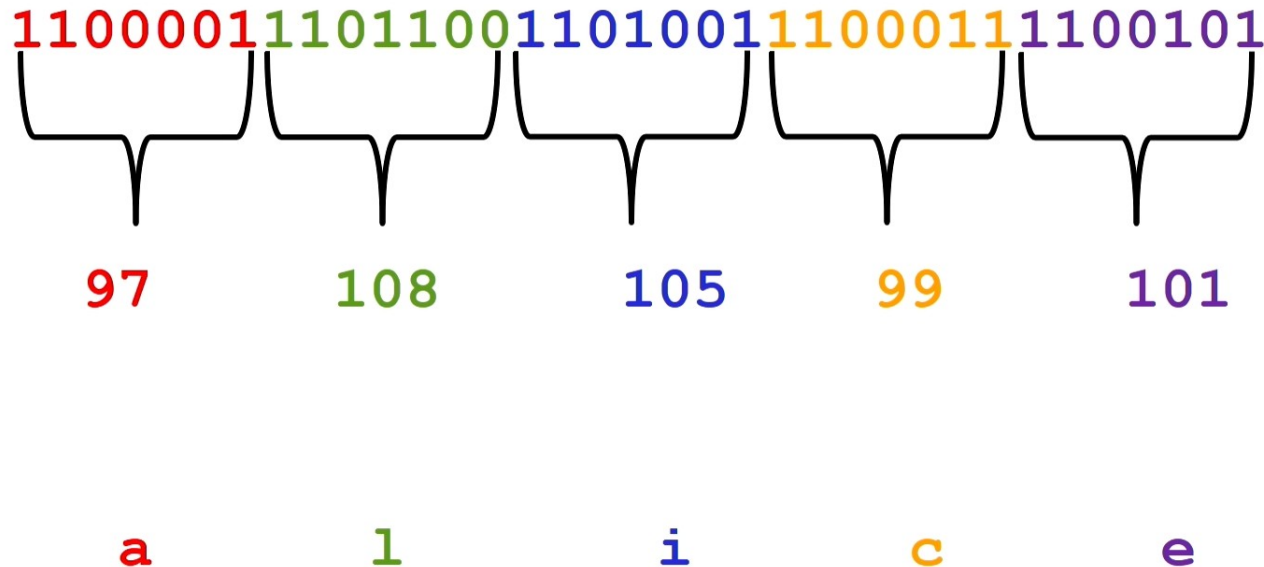


I. Les données et l'information

2- Les formats de données

le code ASCII-

Chaque octet commence par 0 dans ce code, on ne représente pas le premier bit de l'octet...



flux de bits

Conversion
décimale après
décodage 7 bits

Jeux de
caractères ASCII

Tableau des codes ASCII

En premier-->		000	001	010	011	100	101	110	111
En second		0	1	2	3	4	5	6	7
0000	0	NUL	DLE	SP	0	@	P	`	p
0001	1	SOH	DC1		1	A	Q	a	q
0010	2	STX	DC2	"	2	B	R	b	r
0011	3	ETX	DC3	#	3	C	S	c	s
0100	4	ECT	DC4	\$	4	D	T	d	t
0101	5	ENQ	NAK	%	5	E	U	e	u
0110	6	ACK	SYN	&	6	F	V	f	v
0111	7	BEL	ETB	'	7	G	W	g	w
1000	8	BS	CAN	(	8	H	X	h	x
1001	9	HT	EM	)	9	I	Y	i	y
1010	10	LF	SUB	*	:	J	Z	j	z
1011	11	VT	ESC	+	;	K	[	k	{
1100	12	FF	FS	,	<	L	\	l	!
1101	13	CR	GS	-	=	M	]	m	}
1110	14	SO	RS	.	>	N	'	n	~
1111	15	SI	US	/	?	O	-	o	DEL

COMMANDES

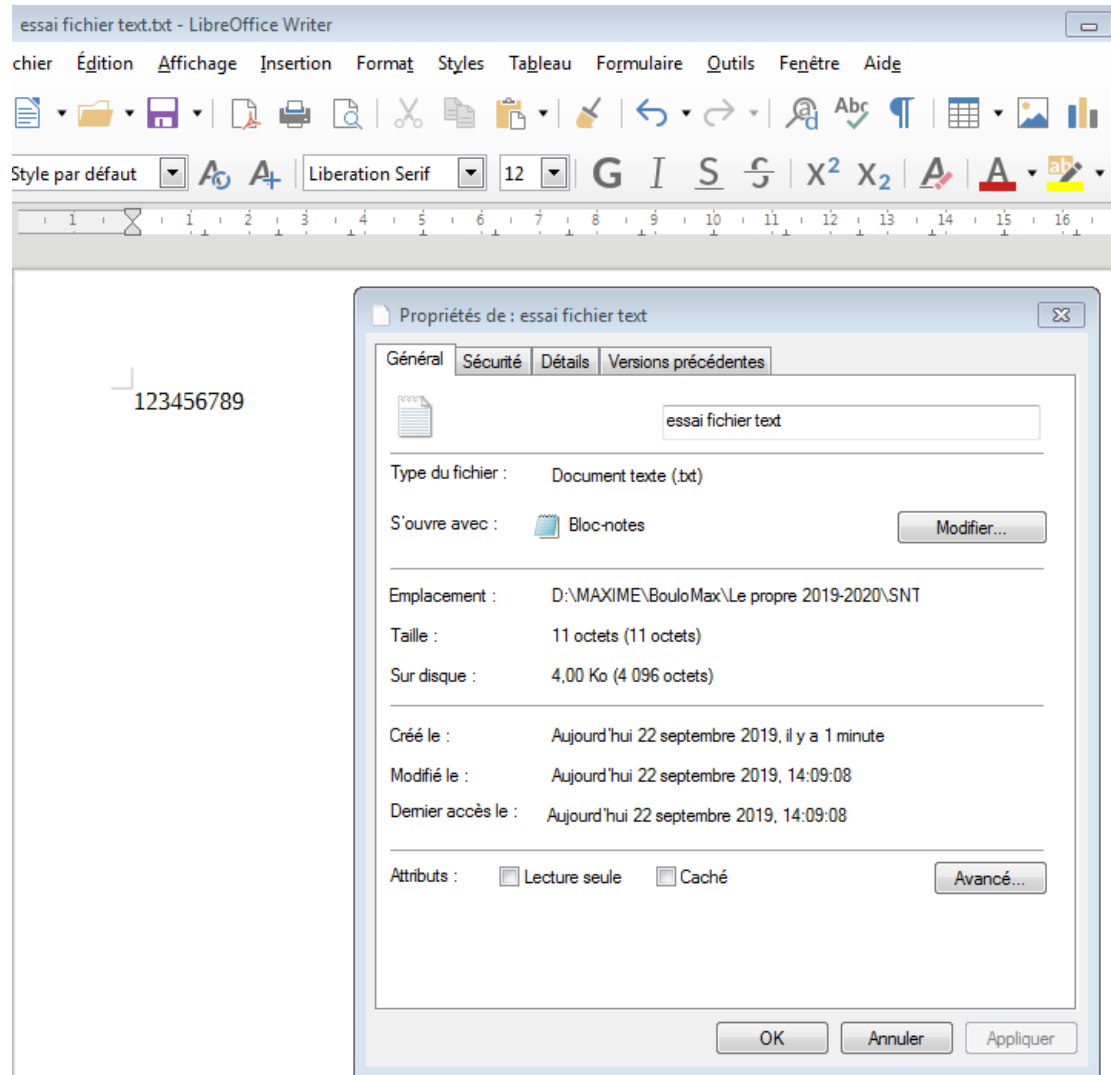
CARACTÈRES

01000001

1. Les données et l'information

2- Les formats de données

**Exemple pour
un texte en
format txt
ascii :**

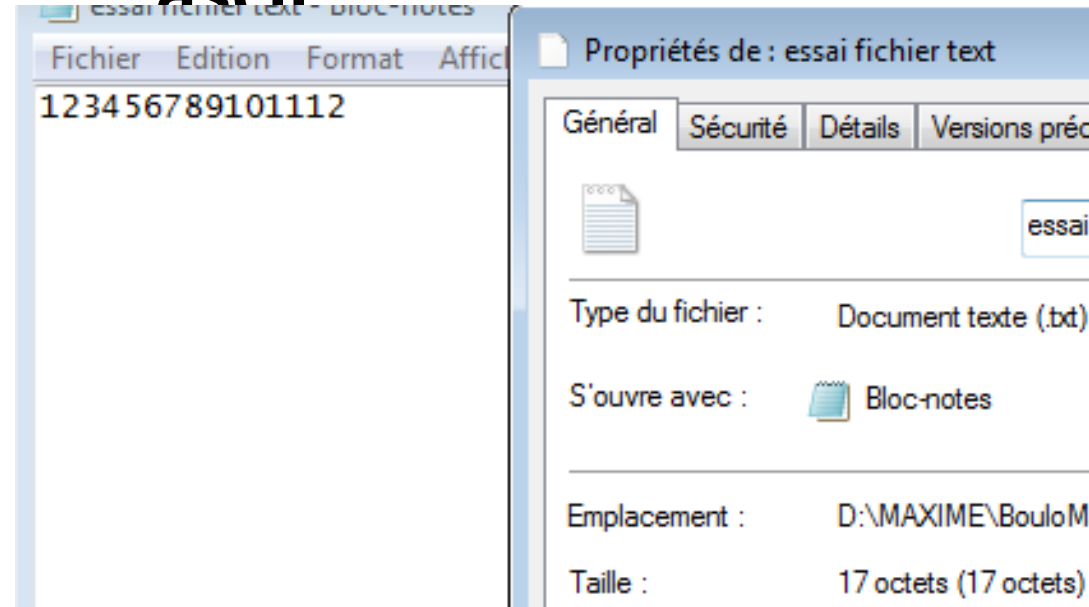


1. Les données et l'information

2- Les formats de données

Dans un texte le premier octet donne la longueur du texte , le dernier est 00000000 indique la fin du texte. Un texte de n caractères occupe donc une place de $n+2$ octets

Exemple pour un texte en format txt ascii :



I. Les données et l'information

2- Les formats de données

ACTIVITE 02-01 LE codage des nombres....

1- Les trois principaux codages utilisés pour coder les nombres entiers.

A- Comment décomposer un nombre en base décimal

B- décomposons un nombre en base binaire

C- décomposons un nombre en base hexadécimal

2- traduire pour communiquer/ Comment convertir d'une base à l'autre

A- Entre le décimal et le binaire

1- Passer du décimal au binaire2- Passer du binaire au décimale

B- Entre le décimal et l'hexadécimal.

1- Passer du décimal à l'hexadécimal.

2- Passer de l'hexadécimal au décimal

II. Les données sont structurées

1- Les tables

La structure de **table** permet de présenter une collection: les objets en ligne, les descripteurs en colonne et les données à l'intersection. Les données sont alors dites structurées.

Pour assurer la persistance des données, ces dernières sont stockées dans des fichiers. Le **format CSV** (Comma Separated Values, les données avec des séparateurs) est un format de fichier simple permettant d'enregistrer une table.

II. Les c

1- Les t

id	serie	numero	titre	editeur	dessinateur	scenariste
1	Thorgal	4	La galère noire	Le Lombard	Rosinki	Van Hamme
2	Blueberry	1	Fort Navajo	Dargaud	Giraud	Charlier
3	Lanfeust de Troy	1	L'ivoire du Magohamoth	Soleil	Tarquin	Arleston
4	Aldebaran	1	La catastrophe	Dargaud	Léo	Léo
5	Astérix	5	Le tour de gaule	Dargaud	Uderzo	Goscinny
6	Gaston Lauffer	12	Le gang des gaffeurs	Dupuis	Franquin	Franquin

Exemple de champ

Exemple de ch

id	serie	numero	titre	editeur	des
1	Thorgal	4	La galère noire	Le Lombard	Ros
2	Blueberry	1	Fort Navajo	Dargaud	Gir
3	Lanfeust de Troy	1	L'ivoire du Magohamoth	Soleil	Tar
4	Aldebaran	1	La catastrophe	Dargaud	Léo

25	Lucky Luke	3	Dalton City	Dargaud	Morris	Goscinny
26	Lucky Luke	11	Le cavalier blanc	Dargaud	Morris	Goscinny
27	Astérix	2	La serpe d'or	Dargaud	Uderzo	Goscinny
28	Astérix	7	Le combat des chefs	Dargaud	Uderzo	Goscinny
29	Astérix	1	Astérix le Gaulois	Dargaud	Uderzo	Goscinny

II. Les données sont structurées

2- Les métadonnées

À toute donnée ou tout fichier de données sont associées des métadonnées qui permettent d'en décrire le contenu.

Les métadonnées varient selon le type de fichier (date et coordonnées de géolocalisation d'une photographie, auteur et titre d'un fichier texte, etc.)

II. Les données sont structurées

3- Les bases de données

Une base de données (ou database, ou BDD par commodité) regroupe ainsi plusieurs collections de données reliées entre elles. Les informations y sont organisées afin d'être facilement consultables, gérables et mises à jour. Elles sont indexées afin de pouvoir facilement trouver les informations recherchées à l'aide d'un logiciel informatique.)

II. Les données sont structurées

4- Enregistrer, accéder et modifier des données

Activité 02-02 partie 1

II.4- Enregistrer, accéder et modifier des données

Partie 1: créer un fichier de données, le modifier, le lire: la BASE!

1- Les instructions de base

a- conseil de sioux: créer un répertoire courant de travail à accès facile pour le fichier à travailler:

`import os` #on importe le module nécessaire pour travailler les fichiers et dossiers(=répertoires)

`os.chdir("C:/tests python")` # indique le répertoire existant de travail courant 'tests pythons' à la racine de C:

`print(os.getcwd())` # pour vérifier qu'on a bien changé le répertoire de travail courant...

II.4- Enregistrer, accéder et modifier des données

Partie 1: créer un fichier de données, le modifier, le lire: la BASE!

1- Les instructions de base

b- Pour **nommer un fichier** dans python, il faut décrire le **chemin absolu** pour s'y rendre: exemple
`C:/mon_dossier/prog_python/mon_fichier.txt`

c- Il faut toujours ouvrir le fichier selon un mode d'ouverture particulier(lecture 'r', écriture 'w', ajout 'a'):
`mon_fichier = open("fichier.txt", "r")` #on affecte le fichier ouvert à une variable mon_fichier pour la LIRE facilement

Rq: 'r': le fichier doit être existant, 'w' le contenu du fichier est écrasé, 'a' le contenu n'est pas écrasé, on peut le modifier. Dans les cas de 'w' et 'a', si le fichier n'existe pas, il est créé.

II.4- Enregistrer, accéder et modifier des données

Partie 1: créer un fichier de données, le modifier, le lire: la BASE!

On peut ensuite affecter le contenu de notre fichier à une variable, la modifier, l'afficher....

`contenu = mon_fichier.read()` # affecte à la variable
contenu le contenu du fichier

`print(contenu)` # affiche le contenu du fichier(en fait de la variable mais c'est la même chose ici...)

`mon_fichier.close()` # pour fermer le fichier et **le rendre disponible aux autres applications(indispensable!!!).**

II.4- Enregistrer, accéder et modifier des données

Partie 1: créer un fichier de données, le modifier, le lire: la BASE!

2- Exercice d'application:

En utilisant l'interface python:

- Dans un dossier “mes fichiers python” situé à la racine de votre clef, créer un fichier .txt vide
- afficher son contenu
- le fermer(vous essaieriez ensuite sans le fermer pour constater le conflit...).
- ouvrez le avec un traitement de texte... et inscrivez y votre NOM et votre prénom et un mot secret, enregistrez, fermez le.
- afficher son contenu

III. Les opérations sur les données

Partie 2: opérations sur les données

1, Quelques opérations à effectuer sur les données.

1- Un exemple: créer un fichier texte liste de course....

Ressources:

`liste.write(variable à ajouter)` #ajoute la variable dans le fichier 'liste', ici ce sera une chaîne de caractères(str)

`liste.write('\n')` #va à la ligne

`print('{}, c'est à présent ton essai n{}'.format(NOM, i))` #

permet d'afficher **JEAN RENE, c'est à présent ton essai n°5**

#si NOM= JEAN RENE et i = 5....

III. Les opérations sur les données

1- Un exemple: créer un fichier texte liste de course....

2- Exercice d'application:En utilisant l'interface python:

exemple de rendu au sein du fichier quel que soit le niveau:

Ma liste de course

1- œufs

2- viande

3- un arbuste pour le jardin

III. Les opérations sur les données

1- Un exemple: créer un fichier texte liste de course....

Niveau 1:

- Créer une programme python qui vous permettra de dresser une liste de course lisible enregistré dans un fichier 'listedecourse.txt'
- ouvrir le fichier 'liste de course'

Niveau 2:

- Créer une programme python qui demande à l'utilisateur:
 - combien de produits il désire mettre sur sa liste de course
 - quels sont ces produits
 - et que ce programme crée un fichier 'liste de course.txt' dans lequel on trouve la liste.
- ouvrir le fichier 'liste de course'

III. Les opérations sur les données

1- Un exemple: créer un fichier texte liste de course....

Niveau ultime:

- Créer une programme python qui demande à l'utilisateur:
 - combien de produits il désire
 - quels sont ces produits
 - et que ce programme crée un fichier 'listedecourse.txt' dans lequel on trouve la liste.

+ le programme doit dresser la liste afin que les produits soit numérotés alors que l'utilisateur n'a entré que le nombre de produits total et les nom de chacun.'

III. Les opérations sur les données

2- Créer et modifier des listes/tables: (HProg)

1-Très utile pour stocker les bases de données, on peut créer des tables, les variables vont donc être définies par des descripteurs. Le plus compatible est fichier de type csv.

Ressource:

Liste=[variable1, variable2, variable3] # on affecte à Liste une liste de trois variables.

Si on a une chaîne de caractères de type chaîne='NOM, PRENOM, AGE'

Liste=chaîne.split(',') # permet de transformer la chaîne en liste de 3 variables séparées par des virgules.

III. Les opérations sur les données

2- Créer et modifier des listes/tables: (HPorg)

1-Très utile pour stocker les bases de données, on peut créer des tables, les variables vont donc être définies par des descripteurs. Le plus compatible est fichier de type csv.

Ressource:

Variable=m[j][i] #ressort l'élément i dans la liste j de la liste de liste m.

Grrrrrr.insert(i, x) #Insère x en i dans la liste Grrrrrr, ajoute un élément dans la liste à une position précise(décale le reste!!!)

III. Les opérations sur les données

2- Créer et modifier des listes/tables: (HPorg)

Pour créer un fichier csv, c'est comme pour un txt mais il faut réfléchir à la structure:

- a. lors de la lecture, chaque ligne pourra être transformée une liste par python à condition de séparer les éléments par un caractère précis(une virgule, un point virgule, une tabulation...), on se servira de la fonction `split(« ? » )`
- b. penser, si l'on veut représenter un tableau, à renseigner la première ligne comme intitulés de colonnes.
- c. Il faut que chaque ligne possède le même nombre d'éléments et qu'ils soient rangés dans le même ordre que les intitulés.

III. Les opérations sur les données

2- Créer et modifier des listes/tables: (HProg)

Un exemple:

`f=open('test.csv', 'w')` # lien vers un fichier on l'affecte la variable f en écriture, on écrase le contenu

`f.write('intVar1,intVar2,intVar3\n')` #on crée les intitulés et on va à la ligne

`f.close()` #on ferme dans la config 'w'

`f=open('test.csv', 'a')` # lien vers un fichier on l'affecte la variable f en ajout, on écrase pas

`f.write(var1+', '+var2+', '+var3+'\n')` #on ajoute à la liste 3 valeurs de variables et on va à la ligne

`f.close()` #on ferme dans la config 'a'

III. Les opérations sur les données

2- Créer et modifier des listes/tables: (Hporg)

Exercice d'application:

Niveau 1:

Créer à présent un programme permettant de créer, en interaction avec un utilisateur, une base de donnée concernant ce client.

Le client devra renseigner son nom, prénom, son sexe(M ou F), son age, son adresse mail, sa pointure de chaussure. Le fichier se nommera basededonneesclients.csv.

III. Les opérations sur les données

2- Créer et modifier des listes/tables: (Hporg)

Exercice d'application:

Niveau 2:

Créer à présent un programme permettant de créer, en interaction avec X(à demander) utilisateurs successifs, une base de donnée clients.

Chaque client devra renseigner son nom, prénom, son sexe(M ou F), son age, son adresse mail, sa pointure de chaussure. Le fichier se nommera basededonneesclients.csv.'

III. Les opérations sur les données

3- Rechercher, filtrer, trier ou calculer sur une ou plusieurs tables des listes/tables:

Objectifs du projet: lire un fichier csv afin de trier son contenu et d'utiliser une partie des données qu'il comporte.

III. Les opérations sur les données

3-

Ressources : (tout n'est pas forcément utile...)

list.append(x) #Ajoute un élément à la fin de la liste 'list'

list.insert(i, x) #Insère un élément à la position indiquée. Le premier argument est la position de l'élément courant avant lequel l'insertion doit s'effectuer

list.remove(x) #Supprime de la liste le premier élément dont la valeur est égale à x

list.clear() #Supprime tous les éléments de la liste

list.index(x[, start[, end]]) #Renvoie la position du premier élément de la liste dont la valeur égale x

list.count(x) #Renvoie le nombre d'éléments ayant la valeur x dans la liste.

III. Les opérations sur les données

2- Exercice d'application:

Nous travaillerons sur le fichier nat2018.csv qui recense tous les prénoms nés en France, par année depuis 1900.

Votre programme python devra demander le prénom que vous recherchez et afficher les lignes correspondantes au prénom ainsi que la somme de toutes les naissances sous ce prénom depuis 1900.

III. Les opérations sur les données

Aide Pas à pas:

a- On l'invente pas, ce fichier nécessite un traducteur, codec, commencez le programme ainsi:

```
import codecs # le fichier nécessite des traducteurs, il est codé en UTF-8
```

```
monfichier=codecs.open('K:/nat2018.csv','r','utf-8') #on affecte la variable 'monfichier' du fichier à travailler
```

b- Il va falloir séparer cette immense chaîne de caractères en listes: d'abord une liste par ligne, puis pour chaque ligne, la transformer en liste qui comprend autant de variables que de colonnes.

c- Il va falloir faire autant de boucles qu'il y a de lignes(moins l'entête des intitulés)

d- Une dernière fonctionne très utile : `mot.upper()` permet de mettre en majuscule une chaîne de caractère (ici mot).

IV- Le stockage des données.

1, Les supports

BLA BLA BLA

