Nous avons vu comment représenter des nombres en informatique, nous allons maintenant voir comment sont représenté **les caractères**.

## Une image contenant texte, homme, personne, vieux Description générée automatiquementRepères Historiques

* **1854** Georges Boule logiciel britannique introduit le calcul booléen qui se révèlera très utile en informatique.
* **1963** Le code ASCII (American Standard Code For Information Interchange) permet d'uniformiser le codage informatique et facilitent les échanges d'informations. Au début il permet uniquement de représenter les caractères principaux de la langue anglaise ainsi que des caractères non imprimables comme le retour à la ligne.
* **1986** La norme ISO 8859-1 (ou Latin-1) est une extension de la norme ASCII, elle est formée de 191 caractères de l'alphabet latin, tous codés sur uniquement un octet. Elle permet de représenter la très grande majorité des caractères de nombreuses langues occidentales, des extensions existent pour d'autres alphabets comme le cyrillique ou le polonais.
* **1988** Le standard Unicode permet quant à lui de représenter tous les caractères du monde entier. Les caractères sont représentés selon la norme U+xxxx où xxxx est un nombre composé de 4 à 6 caractères hexadécimaux. L'interopérabilité entre différents logiciels devient possible, ce standard code actuellement 137 929 caractères.
* **1996** La norme UTF-8 possède un encodage de taille variable pour limiter le coût en mémoire. Les caractères de la table ASCII reste encodé sur 7 bits, tandis que les autres caractères se représentaient sur 2 3 ou 4 octets (95% des sites web).

## Le codage ASCII

Le codage ascii est une norme qui permet d'écrire en anglais, elle définit seulement 128 caractères numérotés de 0 à 127. Sept bits suffisent donc pour représenter un caractère ascii. Les ordinateurs travaillent sur 8 bits, depuis les années 70, le 8ième bits est 0 ou souvent utilisé comme bit de parité.

*Source : article ASCII sur Wikipedia*

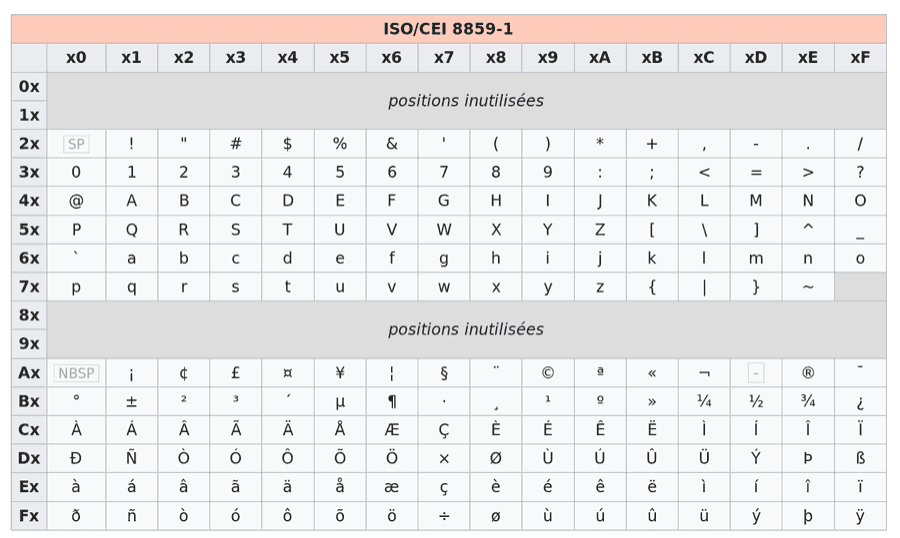
En base 16 :

Une image contenant table

Description générée automatiquement

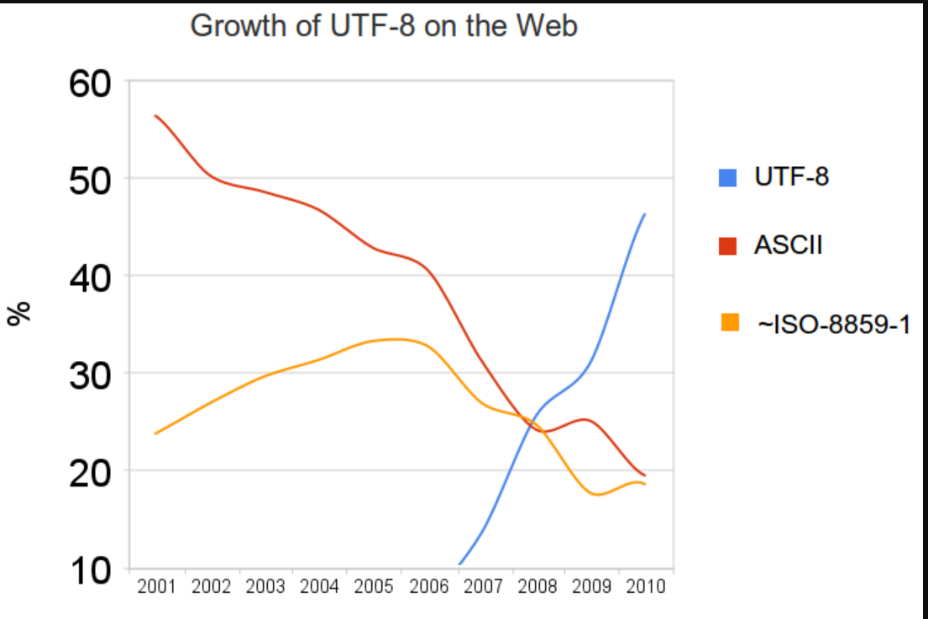
## Le codage ASCII étendu

Pour ajouter les caractères accentués, une autre norme s'impose, la norme ISO-8859-1 ou aussi un nommé ISO-Latin1. Elle utilise les 128 caractères disponible dans le codage ASCII.



On peut constater que cette table ne prend pas en charge le caractère euro, une nouvelle norme adapté a été introduite : l'ISO-8859-15. Il existe d'autres variantes adapté à d'autres langues.

## Le codage UNICODE UTF-8

L'encodage en UTF-8 est en train de s'imposer face aux autres encodage en raison de son universalité.

*"*[*UnicodeGrow2b*](https://commons.wikimedia.org/wiki/File:UnicodeGrow2b.png#mediaviewer/File:UnicodeGrow2b.png)*" by*[*Krauss*](https://commons.wikimedia.org/wiki/User:Krauss)*- Licensed under*[*CC BY-SA 4.0*](http://creativecommons.org/licenses/by-sa/4.0)*via [Wikimedia Commons](https://commons.wikimedia.org/wiki/" \t "not_lyceum).*

La table complète des caractères unicode peut être consultée sur le site [unicode-table.com](http://unicode-table.com/fr/)

Par exemple si on cherche dans cette table la lettre  é , on trouve le code: **U+00E9** appelé point code, en binaire cela correspond au nombre décimal 233, et au nombre binaire 1110 1001.

Ce nombre binaire sera représenté en UTF-8 sur deux octets sous la forme: 110x\_xxxx 10xx\_xxxx. Les chiffres binaires du point de code sont rangés de droite à gauche aux positions 'x' de ce schéma.

On obtient donc 1100\_0011 1010\_1001 pour le caractère  é .

Ce qui représente en hexadécimal: C3 A9

## Encodage en python

En python 3 tous les fichiers doivent être codés en utf-8, et les chaînes de caractères string sont encodées en utf 8. Contrairement au bytes qui ne doivent contenir que des caractères ascii (nom des variables …).

La fonction **ord()** de python retourne le code ASCII d'un caractère en base 10.

*>>>* ***ord('a')***

On peut convertir en hexadécimal :

*>>>* ***hex(ord('a'))***

La fonction **chr()** retourne le caractère correspondant à un entier:

>>> ***chr(0x26)***

On peut aussi utiliser la saisis direct par le code ASCII hexadécimal du caractère avec la notation **\xhh**, où hh est le code hexadécimal.

>>> ***print('\x43e\x63i es\x74…')***

**Remarque :** Cette technique de saisie des caractères à l'aide du caractère **\** est appelée **caractère échappé** (ce nom vient du caractère **\** qu'on appelle également **caractère d'échappement**). D'autres raccourcis de caractères échappés sont également disponibles pour saisir les caractères spéciaux tableau ci-dessous donne les plus utilisés.

\b suppression

\f nouvelle page

\n nouvelle ligne

\r retour chariot

\t tabulation horizontale

\v tabulation verticale

\\ pour afficher le caractère \

## Encodage en HTML

Pour obtenir un affichage correct des caractères spéciaux en acheter ml il faut déclarer l'encodage utf 8 dans le fichier d'en-tête <head> acheter html en utilisant la balise <meta>

Exemple : <meta charset="utf-8">

Il est également possible d'afficher des caractères par leur code html, ces codes sont également présents sur la page [unicode-table.com](http://unicode-table.com/fr/).

Par exemple le caractère  À   U+00C0 peut être codé en html par &#192; , ou encore en notation hexadécimale: &#xc0;

## Exercices

### Questionnaire

#### Combien de caractères différents peut-on coder sur 7 bits ?

#### Pouvez-vous imaginer une des limites du code ASCII ?

#### Quel est l'intérêt d'avoir ajouté le 8ième bits de codage dans la norme Latin-1

#### Dans le traitement de texte de votre choix, saisir Alt+65. Quels caractères de la table UTF-8 obtenez vous ? Justifier que le caractère A soit codé par la valeur 41 en UTF-8.

#### À quel caractère de la table UTF 8 correspond les saisies Alt+192, Alt+0201 et Alt+0200 ?

### Codage Ascii

#### Donner le codage ASCII des 2 chaînes de caractères python ci-dessous.

'bonjour tout le monde'

'''programmer en Python'''

#### Écrire une fonction printASCII(s) qui affiche à l'écran les codes ascii au format hexadécimal d'une chaîne de caractères. Utiliser cette fonction pour vérifier la réponse à la question précédente.

### Chiffrement de texte

#### L'algorithme **rot13** est un algorithme très simple de chiffrement qui consiste décaler de 13 caractères chaque lettre d'un texte. Par exemple ***python*** est transformé en ***clguba***. Écrire en python une fonction **rot13** en supposant que la chaîne passer en argument ne contient que des caractères entre a et z minuscule et éventuellement des espaces. Votre fonction ne doit décaler que les lettres de l'alphabet, elle ne touche donc pas aux espaces. Elle renvoie en sortie une chaîne de caractères.

### Codage UTF-8

Sachant que le point de code du symbole é est 233, donner la séquence de points de code du mot ***élégance***, puis les octets en binaire correspondant à l'encodage UTF-8 de ce mot.