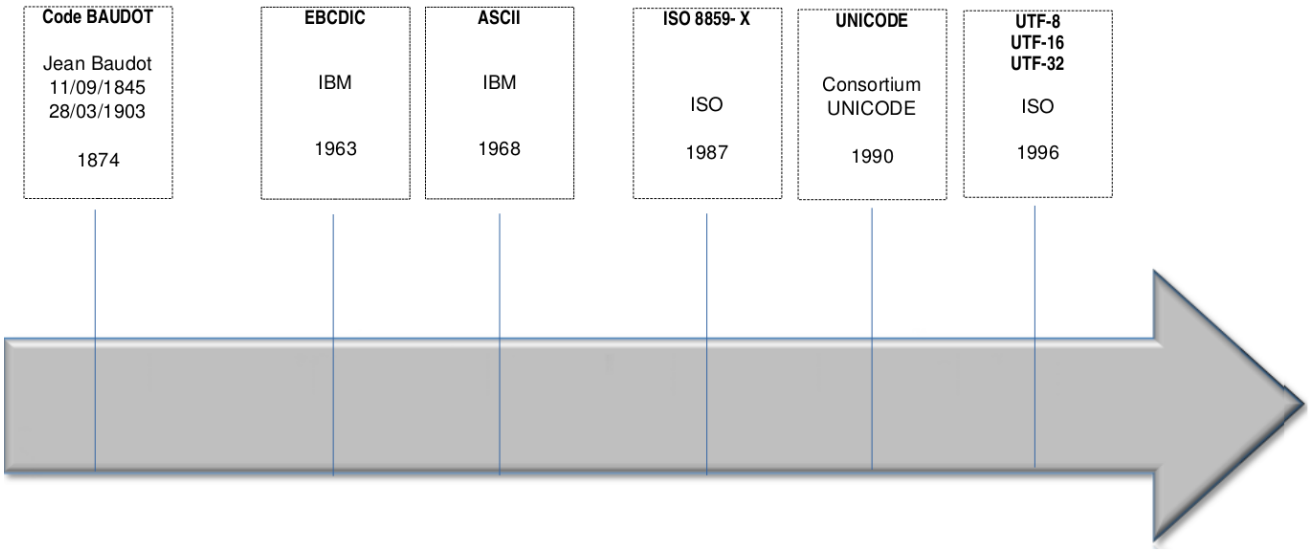
Pour être fonctionnelle, la représentation des caractères dans un système informatique doit respecter quelques principes :
– Associer un nombre unique à chaque caractère ;
– Tous les systèmes informatiques doivent utiliser le même encodage pour pouvoir échanger ;
– Être la plus compacte que possible pour économiser l'espace mémoire et le volume de données échangé lors des communications.

Voici, sans être exhaustif, quelques codes utilisés :



1. Le code ASCII (American Standart Code for Information Interchange) :

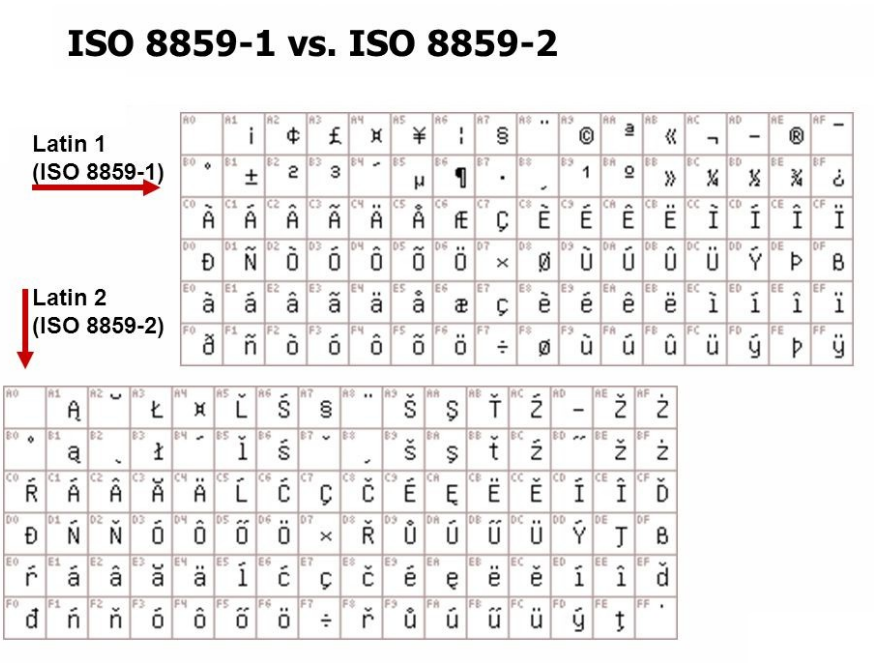
Ce codage utilise 7 bits. Il définit un jeu de 128 caractères (imprimable ou non).

Extrait de la table ASCII

Dec	Hex	Oct	Binary	Char	Dec	Hex	Oct	Binary	Char	Dec	Hex	Oct	Binary	Char	Dec	Hex	Oct	Binary	Char
0	00	000	0000000	NUL (null character)	32	20	040	0100000	space	64	40	100	1000000	@	96	60	140	1100000	`
1	01	001	0000001	SOH (start of header)	33	21	041	0100001	!	65	41	101	1000001	A	97	61	141	1100001	a
2	02	002	0000010	STX (start of text)	34	22	042	0100010	"	66	42	102	1000010	B	98	62	142	1100010	b
3	03	003	0000011	ETX (end of text)	35	23	043	0100011	#	67	43	103	1000011	C	99	63	143	1100011	c
4	04	004	0000100	EOT (end of transmission)	36	24	044	0100100	$	68	44	104	1000100	D	100	64	144	1100100	d
5	05	005	0000101	ENQ (enquiry)	37	25	045	0100101	%	69	45	105	1000101	E	101	65	145	1100101	e
6	06	006	0000110	ACK (acknowledge)	38	26	046	0100110	&	70	46	106	1000110	F	102	66	146	1100110	f
7	07	007	0000111	BEL (bell (ring))	39	27	047	0100111	'	71	47	107	1000111	G	103	67	147	1100111	g

2. Normes ISO 8859-n

La liste des caractères imprimables de la table ASCII est insuffisante pour transmettre des caractères dans des langues autres que l'anglais.
Pour remédier à ce problème, la norme ISO 8859 est apparue. On utilise 8 bits pour représenter un caractère soit 256 caractères différents.
Cela reste cependant bien insuffisant pour représenter tous les caractères utilisés rien que dans les langues latines. La norme prévoit donc plusieurs tables notées ISO 8859-n où n représente le numéro de la table. Ces tables sont conçues pour être indépendantes et compatibles entre elles. Ces tables sont également compatibles avec le code ASCII.
Il est par contre impossible de « piocher » les caractères dans différentes tables.



3. Unicode

Pour remplacer l'utilisation des tables de code, l'ISO a défini un jeu universel de caractères (UCS : Universal Character Set).
À chaque caractère est associé :
– Un nom unique ;
– Un point de code (nombre) ;

Les 256 premiers points de code sont compatibles avec la norme ISO-8859-1.

https://unicode-table.com/fr/

Unicode se contente de recenser, nommer les caractères et leur attribuer un numéro. Mais il ne dit pas comment ils doivent être codés en informatique.

Depuis sa version 3, Python utilise Unicode pour le codage des caractères.

Plusieurs codages des caractères Unicode existent :

Nombre d'octets(s) utilisé(s) selon le format d'encodage

Numéro UNICODE	UTF-8	UTF-16	UTF-32
U+0000 à U+007F	1	2	4
U+0080 à U+07FF	2	2	4
U+0800 à U+FFFF	3	2	4
U+10000 à U+10FFFF	4	4	4

Le plus couramment utilisé, notamment pour les pages Web, est UTF-8.

UTF-8 (UCS (Universal Character Set) Transformation Format 8 bits) est un format de longueur variable, défini pour les caractères Unicode. Chaque caractère est codé sur une suite de un à quatre octets. UTF-8 a été conçu pour assurer une bonne compatibilité avec les logiciels prévus pour traiter des caractères d'un seul octet. Les protocoles de communication d'Internet échangeant du texte doivent supporter UTF-8.

Description

Unicode attribue un numéro à chaque caractère. Les caractères de numéro 0 à 127 sont codés sur un octet dont le bit de poids fort est toujours nul. Les caractères de numéro supérieur à 127 sont codés sur plusieurs octets. Dans ce cas, les bits de poids fort du premier octet forment une suite de 1 de longueur égale au nombre d'octets utilisés pour coder le caractère, les octets suivants ayant 10 comme bits de poids fort.

Ce principe pourrait être étendu jusqu'à six octets pour un caractère, mais UTF-8 pose la limite à quatre. Ce principe permet également d'utiliser plus d'octets que nécessaire pour coder un caractère, mais UTF-8 l'interdit.

Définition du nombre d'octet utilisé - Représentation binaire	Signification
0xxxxxxx	1 octet codant 1 à 7 bits
110xxxxx 10xxxxxx	2 octets codant 8 à 11 bits
1110xxxx 10xxxxxx 10xxxxxx	3 octets codant 12 à 16 bits
11110xxx 10xxxxxx 10xxxxxx 10xxxxxx	4 octets codant 17 à 21 bits

Dans toute chaîne de caractères UTF-8, on remarque que :

– Tout octet de bit de poids fort nul code un caractère US-ASCII sur un octet ;

– Tout octet de bits de poids fort valant 11 est le premier octet d'un caractère codé sur plusieurs octets ;

– Tout octet de bits de poids fort valant 10 est à l'intérieur d'un caractère codé sur plusieurs octets.