NSI · PREMIÈRE / COURS

Booléens et encodage du texte

Astrolab Cours · astrolabcours.fr

01 Le problème

Un ordinateur doit prendre des décisions (« si le mot de passe est bon et que le compte n'est pas bloqué ») et stocker du texte avec des accents, des emojis et des alphabets du monde entier, alors qu'il ne connaît que des 0 et des 1.

01 L'idée : l'interrupteur et le dictionnaire de codes

Un booléen, c'est un interrupteur : allumé ou éteint, vrai ou faux. Du texte, c'est un message chiffré avec un dictionnaire de codes : chaque caractère a un numéro. Si l'expéditeur et le lecteur n'utilisent pas le même dictionnaire, le message est déformé.

01 Les booléens

Un **booléen** vaut `True` ou `False`. Trois opérateurs de base :

a	b	a and b	a or b	not a
F	F	F	F	V
F	V	F	V	V
V	F	F	V	F
V	V	V	V	F

Le **ou exclusif** (XOR) est vrai quand exactement un des deux est vrai.

En Python, `and` et `or` sont **paresseux** : dans `a and b`, si `a` est faux, `b` n'est même pas évalué. C'est utile pour écrire `i < len(t) and t[i] > 0` sans risquer de sortir du tableau.

Encoder le texte

- **ASCII** (1963) : 128 caractères codés sur 7 bits, lettres anglaises sans accents. `A` = 65, `a` = 97, `0` = 48.
- **ISO-8859-1** (Latin-1) : 256 caractères sur 8 bits, avec les accents d'Europe de l'Ouest.
- **Unicode** attribue un numéro à chaque caractère de toutes les écritures (plus de 150 000). **UTF-8** les encode sur 1 à 4 octets ; les caractères ASCII gardent leur code sur 1 octet.

UTF-8 est aujourd'hui l'encodage de presque tout le web.

En Python : `ord("A")` donne 65, `chr(233)` donne `"é"`, et `"é".encode("utf-8")` donne deux octets.

© Astrolab Cours · astrolabcours.fr/nsi/premiere/booleens-et-texte/ · Tous droits réservés : usage personnel uniquement, republication et usage commercial interdits

Exemple guidé

Un site accepte la connexion si `mdp_ok and not bloque`. Puis il enregistre le prénom « Léa ».

mdp_ok	bloque	Connexion
True	False	True
True	True	False
False	False	False
False	True	False

1. Un seul cas sur quatre ouvre le compte : bon mot de passe **et** compte non bloqué.
2. En UTF-8, « L » et « a » prennent un octet chacun (codes 76 et 97, comme en ASCII). Le « é » (point de code U+00E9) en prend deux.

```
"Léa".encode("utf-8")   # b'L\xc3\xa9a'
len("Léa".encode("utf-8"))   # 4 octets pour 3 caractères
```

Retour au problème

Les décisions se ramènent à des opérations sur True et False, et chaque caractère à un nombre codé en binaire. UTF-8 garde un octet pour les lettres simples et en utilise jusqu'à quatre pour les accents, les autres alphabets et les emojis.

Les pièges

- `a or b and c` : `and` est prioritaire sur `or`. Mettre des parenthèses.
- Lire un fichier UTF-8 en Latin-1 : « é » devient « Ã© ».
- Un caractère n'occupe pas toujours un octet en UTF-8.

© Astrolab Cours · astrolabcours.fr/nsi/premiere/booleens-et-texte/ · Tous droits réservés : usage personnel uniquement, republication et usage commercial interdits