

❄ Chapitre 2 ❄

Les booléens

I. Pourquoi des booléens ?

Ce type de variable est né de la nécessité d'avoir un type de variable ne prenant que deux valeurs : VRAI ou FAUX (0 ou 1) afin de répondre à des conditions d'exécution de code.

🍃 Exemple 1:

Obtention de la mention « Bien » au Baccalauréat : selon la note moyenne N obtenue, quelle(s) condition(s) doivent être validées ? D'autres conditions étaient-elles envisageables ?

II. Un peu d'histoire

L'algèbre de Boole est due au logicien, mathématicien et philosophe britannique George Boole (1815-1864). En 1847 puis 1854, ce scientifique publie les bases de ce qu'on appelle aujourd'hui l'algèbre de Boole. Son intérêt est de traduire des idées et des concepts en équations, leur appliquer certaines lois et retraduire le résultat en termes logiques, n'acceptant que deux valeurs numériques : 0 et 1 (Faux ou Vrai). L'importance de son travail est dans de nombreux domaines (théorie des probabilités, circuits téléphoniques, hydrauliques...), dont l'informatique.

❄ Définition 1:

Un booléen est un type de variable (type bool en Python) à deux états : Vrai ou Faux, représentés respectivement par 1 et 0.

III. Booléens et opérateurs de comparaison

Un booléen peut être le résultat d'une comparaison, il s'agit alors d'un test. Si le test est valide, le résultat est alors Vrai (True sous Python). Dans le cas contraire, le résultat est Faux (False sous Python).

🍃 Exemple 2:

```
1 N = 13
2 print(N > 12)
```

L'interpréteur affiche True

```
1 N = 13
2 print(N < 12)
```

L'interpréteur affiche False

Les opérateurs de comparaison que nous utiliserons cette année sont :

Opérateur	Signification
<	strictement inférieur
<=	inférieur ou égal
>	strictement supérieur
>=	supérieur ou égal
==	égal
!=	différent

IV. Opérateurs booléens et tables de vérité

1. Opérateur « and »

L'opérateur « and » permet de tester si deux booléens sont vrais : « a and b » a pour résultat True si et seulement si a est vrai et b est vrai. Sa table de vérité est donnée par :

⚠ Remarque :

- Mathématiquement le résultat est donné par $a \times b$, c'est-à-dire le produit.
- Cette opération booléenne est appelée « conjonction »

🍃 Exemple 3:

- La variable booléenne AmpouleOK représente le fait qu'une ampoule est en état de fonctionnement.
- La variable booléenne InterrupteurOn est vraie si l'interrupteur contrôlant cette ampoule est en position « on ».
- La variable booléenne LumiereAllumee représente le fait que la lumière est allumée.

Alors :

a	b	a and b
0	0	0
0	1	0
1	0	0
1	1	1

$$\text{LumiereAllumee} = \text{AmpouleOK} \text{ And } \text{InterrupteurOn}$$

Ainsi, on peut affirmer que la lumière est allumée (LumiereAllumee est vraie) si et seulement si l'ampoule est en état de fonctionnement et si l'interrupteur reste en position « on » (AmpouleOK vrai et InterrupteurOn vrai).

2. Opérateur « or »

L'opérateur « or » permet de tester si l'un des deux booléens au moins est vrai : « a or b » a pour résultat True si et seulement si au moins un des booléens a ou b est vrai. Sa table de vérité est donnée par :

a	b	a or b
0	0	0
0	1	1
1	0	1
1	1	1

Remarque :

Cette opération booléenne est appelée « disjonction »

Exemple 4:

- La variable booléenne AmpouleHS représente le fait qu'une ampoule n'est pas en état de fonctionnement.
- La variable booléenne InterrupteurOff est vraie si l'interrupteur contrôlant cette ampoule est en position « off ».
- La variable booléenne LumiereEteinte représente le fait que la lumière est éteinte.

Alors :

$$\text{LumiereEteinte} = \text{AmpouleHS} \text{ or } \text{InterrupteurOff}$$

Ainsi, on peut affirmer que la lumière est éteinte (LumiereEteinte est vraie) si et seulement si l'ampoule n'est pas en état de fonctionnement ou si l'interrupteur est en position « off » (AmpouleHS vrai ou InterrupteurOff vrai).

3. Opérateur « not »

L'opérateur « not » retourne le booléen opposé. Sa table de vérité est donnée par :

a	not a
0	1
1	0

Remarque :

Cette opération booléenne est appelée « négation »

Exemple 5:

- La variable booléenne EauSolide représente le fait que l'eau est sous forme de glace.
- La variable booléenne TemperaturePositive est vraie si la température de l'eau est positive.

Alors :

$$\text{EauSolide} = \text{not } \text{TemperaturePositive}$$

Ainsi, si TemperaturePositive est vraie avant cette affectation, alors EauSolide vaudra faux après cette affectation. Réciproquement, si TemperaturePositive est fautive avant cette affectation, alors EauSolide vaudra vrai après cette affectation.

4. Opérateur « xor »

L'opérateur « xor » aussi appelé « ou exclusif » permet de tester si seulement l'un des deux booléens est vrai : « a xor b » a pour résultat True si et seulement si un des booléens a ou b est vrai. Sa table de vérité est donnée par :

a	b	a xor b
0	0	0
0	1	1
1	0	1
1	1	0

Exemple 6:

Deux interrupteurs va-et-vient contrôlent la lumière d'une pièce, pour qu'elle soit allumée, les deux doivent être dans la même position. La variable booléenne VaEtVient1_ON représente le fait que l'interrupteur va-et-vient n°1 est en position « on ». La variable booléenne VaEtVient2_ON représente le fait que l'interrupteur va-et-vient n°2 est en position « off ». EtatLumiere est vraie si la lumière est allumée. Alors :

$$\text{EtatLumiere} = \text{VaEtVient1_ON} \text{ xor } \text{VaEtVient2_ON}$$

Ainsi, si EtatLumiere est vraie si et seulement si les deux interrupteurs sont dans la même position.