

Type booléen et opérateurs booléens

"Définition 1 : Type booléen"

Le type **booléen** ne prend que deux valeurs `True` et `False`. Son nom est un hommage à [George Boole](#) qui est le premier mathématicien à formaliser des calculs logiques au dix neuvième siècle

"Définition 2 : opérateur booléen"

Une fonction qui ne prend que des paramètres booléens et qui renvoie un booléen, est appelée **opérateur booléen**.

Si la fonction prend n paramètres et que chaque paramètre peut prendre deux valeurs cela donne 2^n listes distinctes de valeurs de paramètres.

Un **opérateur booléen** peut donc entièrement déterminé par un tableau où on fait apparaître une colonne par paramètre et une colonne pour la valeur renvoyée. Un tel tableau est appelé **table de vérité**. Si l'opérateur a n paramètres, sa table de vérité aura 2^n lignes.

"Exemple"

Voici la table de vérité d'un opérateur booléen f avec deux paramètres, on a 2^2 lignes.

a	b	f(a, b)
False	False	True
False	True	True
True	False	True
True	True	False

"Propriété 1 : Opérateurs booléens de base"

"Opérateur Non"

C'est un opérateur booléen *unaire* (un paramètre donc 2^1 lignes).

Table de vérité :

a	non (a)
False	True
True	False

En Python :

```
>>> not True
False
```

"Opérateur Ou"

C'est un opérateur booléen *binnaire* (deux paramètres donc 2^2 lignes).

Table de vérité :

a	b	ou(a, b)
False	False	False
False	True	True
True	False	True
True	True	True

En Python :

```
>>> True or False
True
```

"Opérateur Et"

C'est un opérateur booléen *binaire* (deux paramètres donc 2^2 lignes).

Table de vérité :

a	b	et(a, b)
False	False	False
False	True	False
True	False	False
True	True	True

En Python :

```
>>> True and False
False
```

Exercice :

Carnet Capytale : <https://capytale2.ac-paris.fr/web/c/aa70-4126072>