

Opérateurs logiques en C#

5 JUILLET

Concepteur Développeur d'Applications

Auteur : Aurélien BOUDIER



Les opérations logiques ?

Les opérations logiques font partie de l'algèbre booléenne, qui est souvent utilisé en informatique, en sciences et en électronique. Voici une brève introduction.

En algèbre booléenne, la valeur des variables ne peut être que vraie ou fausse, également notée 1 et 0 respectivement. Contrairement à l'algèbre élémentaire, où les principales opérations sont l'addition, la soustraction, etc, les principales opérations de l'algèbre booléenne sont la conjonction (ET), la disjonction (OU) et la négation (NON).

Le ET logique

Supposons que nous avons deux variables : x et y : **En C #, l'opérateur ET logique est indiqué par &&.**

Nous pouvons définir une expression booléenne comme suit : **z = x && y**

Dans cette expression, z est vrai si x et y sont vrais, sinon z sera faux.

Exemple concret en programmation :

Imaginez que vous développez une application de gestion de prêt. Le prestataire n'offre des prêts qu'aux candidats âgés de plus de 18 ans et citoyens du pays donné. Dans cet exemple, nous avons deux variables :

x = demandeur âgé de plus de 18 ans

y = demande d'être citoyen

z = est éligible pour demander un prêt = x && y

Si les deux x et y sont vrais, le demandeur est admissible à demander un prêt.

Grace aux instructions conditionnelles, on pourra vérifier si l'expression ci-dessus est évaluée à vrai ou faux, puis on pourra modifier le flux de votre application en fonction.

Le OU logique

En C #, le OU logique est indiqué par deux lignes verticales (||).

Considérant l'expression suivante : **z = x || y**

z sera vrai, si x ou y est vrai.

Exemple concret en programmation :

Imaginez que vous créez un logiciel pour un recruteur. Pour une offre d'emploi, les candidats peuvent postuler s'ils ont un diplôme en informatique, ou plus de 5 ans d'expérience dans le domaine.

Vous pouvez modéliser cela à l'aide d'une expression booléenne comme suit :

x = le candidat a un diplôme en informatique

y = le candidat a plus de 5 ans d'expérience

z = la candidature est éligible = x || y

Si x ou y est vrai, z sera vrai.

Donc, contrairement au ET logique, où les deux variables doivent être vraies, avec le OU logique, si au moins une des deux est vrai, le résultat sera vrai.

Le NON logique

L'opérateur NOT en C # est indiqué par un point d'exclamation (!) Il est l'inverse la valeur d'une variable ou d'une expression donnée : $y = !x$

Donc, ici, si x est vrai, y sera faux, et si x est faux, y sera vrai.

Un codage propre et compréhensible

Dans tous les exemples précédents, on a utilisé les variables x et y, principalement pour relier la programmation à l'algèbre booléen.

Mais en ce qui concerne le codage, on doit éviter d'utiliser des noms de variables tels que x, y, z car ils ne donnent pas explicitement l'utilité de la variable dans le programme. Cela peut poser des problèmes aux autres développeurs qui lisent notre code (ou même à nous-même). Au lieu de cela, utilisons des noms de variables avec du sens.

Il est souvent recommandé de préfixer les noms booléens avec IS ou HAS (si possible).

Crédits :

Action	Auteur	Descriptif	Date
Rédaction du document	Aurélien BOUDIER	Rédaction du document sur les opérateurs logiques en C#	05/07/20