JS 06 : Opérateurs

> Dans tous les exemples de ce cours, on considère que la valeur initiale de `x` est égale à *11*.

Les opérateurs de calcul

Signe	Nom	Signification	Exemple	Résultat
`+`	Plus	Addition	x + 3	14
`-`	Moins	Soustraction	x - 3	8
`*`	Multiplié par	Multiplication	x*2	22
`/`	Divisé par	Division	x/2	5.5
`%`	Modulo	Reste de la division	x%5	1
`=`	Egal	Reçoit la valeur de droite	x = 5	5

> L'opérateur `+` sert aussi à concaténer des chaînes de caractères.

Les opérateurs de comparaison

+++ TODO : A supprimer, transféré dans JS - 06 - Les Conditions. (09/08/2019) +++

Signe	Nom	Exemple	Résultat
`==`	Egal	`x == 11`	`true`
`<`	Inférieur	`x < 11`	`false`
`<=`	Inférieur ou égal	`x <= 11`	`true`
`>`	Supérieur	`x > 11`	`false`
`>=`	Supérieur ou égal	`x > 11`	`true`
`!=`	Différent	`x != 11`	`false`

> **On confond souvent `=` avec `==`.** Le `=` est un opérateur d'affectation tandis que le `==` est un opérateur de comparaison. Cette confusion est une source classique d'erreur de programmation.

Les opérateurs associatifs

On appelle ainsi les opérateurs qui réalisent un calcul dans lequel une variable intervient des deux côtés du signe `=` (ce sont donc en quelque sorte des opérateurs d'attribution).

Dans les exemples suivants `x` vaut toujours *11* et `y` aura comme valeur *5*.

Signe	Description	Exemple	Signification	Résultat
+=	plus égal	x += y	x = x + y	16
-=	moins égal	x -= y	x = x - y	6
*=	multiplié égal	x *= y	x = x * y	55
/=	divisé égal	x /= y	x = x / y	2.2

Les opérateurs logiques

Aussi appelés opérateurs booléens, ils servent à vérifier deux ou plusieurs conditions.

Signe	Nom	Exemple	Signification
&&	et	(condition1) && (condition2)	condition1 et condition2
||	ou	(condition1) || (condition2)	condition1 ou condition2

Les opérateurs d'incrémentation

Ces opérateurs vont augmenter ou diminuer la valeur de la variable d'une unité. Ce qui sera fort utile, par exemple, pour mettre en place des boucles.

> Dans les exemples ci-dessous, `x` vaut 3.

Signe	Description	Exemple	Signification	Résultat
x++	incrémentation ()	y = x++	x++ est la même chose que x = x + 1	y vaut 4
x--	décrémentation	y = x--	x-- est la même chose que x = x - 1	y vaut 2

La priorité des opérateurs JavaScript

Les opérations s'effectuent dans l'ordre suivant de priorité (du degré de priorité le plus faible au degré de priorité le plus élevé).

Dans le cas d'opérateurs de priorité égale, de gauche à droite.

Opération	Opérateur
,	virgule ou séparateur de liste
= += -= *= /= %=	affectation
? :	opérateur conditionnel
\|\| | ou logique
&& | et logique
== != | égalité, différence
< <= >= > | relationnel
+ - | addition, soustraction
* / | multiplication, division
!- ++ -- | unaire
!- ++ -- | parenthèses

Exercice

Soit les variables suivantes :

- `a` qui contient la chaîne de caractères 100
- `b` = 100
- `c` qui contient la valeur 1,00
- `d` booléen qui vaut vrai

A réaliser :

- Affichez "Ceci est une chaîne de caractères :" et concaténez cette chaîne avec la variable `a` pour afficher "Ceci est une chaîne de caractères : *100*".
- Appliquez à `b` l'opérateur de décrémentation
- Ajoutez à `c` la valeur de `a`
- Inversez la valeur de `d`