

# Fonaments de la programació

Operadors en JavaScript

`{.js}`  
**JavaScript**



**UNIÓ EUROPEA**  
Fons Europeu de  
Desenvolupament Regional

# Taula d'operadors en JavaScript



ARITMÈTICS	RELACIONALS	LÒGICS	STRING	D'ASSIGNACIÓ	CONDICIONAL
(1)	(2)	(3)	(4)	(5)	(6)
+	==	&&	+	=	? :
-	===		+=	+=	
*	!=	!		-=	
**	!==			*=	
/	>			/=	
%	<			%=	
++	>=			**=	
--	<=				

# Operadors aritmètics

Els operadors aritmètics s'utilitzen per realitzar operacions matemàtiques comunes.  
El més important a l'hora d'utilitzar-los és tenir en compte l'ordre de precedència d'aquests.

**Utilitzar parèntesis per a canviar l'ordre d'execució**

	Operador	Descripció
1	++ --	(Operadors unitaris ) increment i decrement
2	**	eleva a una potència
3	* / %	multiplicar / dividir
4	+ -	suma resta

Les operacions al mateix nivell (com + i -) s'executen en l'ordre en què es troben

# Operadors unitaris

Aquests operadors afecten un sol operant.

- L'operador **++** (operador increment) incrementa en 1 el valor de la variable.
- L'operador **--** (operador decrement) resta 1 al valor de la variable.

Exemple:

```
let i = 5;  
i++;    // i vale 6  
i--;    // i vale 5
```

# Operadors relacionals o de comparació

Els operadors relacionals **comparen dos operands** i donen com a resultat de la comparació **true** o **false**.



Operador	Descripció
==	igual valor
===	igual valor i igual tipus
!=	diferent
!==	diferent valor o tipus
>	més gran que
<	més petit que
>=	més gran o igual que
<=	més petit o igual que

Els operadors de comparació es poden utilitzar en declaracions condicionals per comparar valors i prendre mesures en funció del resultat.

```
if (edat < 18) missatge = "menor d'edat";
```

# Concatenar (unir) cadenes de text

L'operador + també es pot utilitzar per afegir (concatenar) cadenes de text.  
Exemple:

```
let nom = "Pepito";  
let cognom = "Pérez";  
let nom_cognom = nom + " " + cognom;
```

L'operador d'assignació += també es pot utilitzar per afegir (concatenar) cadenes:

```
let nom = "Pepito";  
nom += " Pérez";
```



Si sumeu dos nombres, es retornarà la suma,  
però si afegiu un nombre i una cadena es tornarà una cadena

# Operadors lògics

Ordre de precedència: **!** **&&** **||**

Els operadors lògics s'utilitzen amb operands de tipus **boolean**.

S'utilitzen per a construir expressions lògiques, el resultat de les quals és de tipus **true** o **false**.

Els operadors lògics a JavaScript són:

- &&**      AND.      El resultat és **true** si els dos operands són vertaders.  
El resultat és **false** en cas contrari.  
Si el primer operant és fals no s'avalua el segon, ja que el resultat serà fals.
- ||**      OR.      El resultat és **false** si els dos operands són falsos.  
Si un d'ells és veritader el resultat és **true**.  
Si el primer operant és veritader no s'avalua el segon.
- !**      NOT.      S'aplica sobre un sol operant.  
Canvia el valor de l'operand de **true** a **false** i viceversa.

# Operadors d'assignació

S'utilitzen per a assignar el valor d'una expressió a una variable.

En general:

variable **operador** = expressió

equivale a escriure

variable = variable **operador** expressió

**Exemple:**

```
x += 3;
```

equivale a escriure

```
x = x + 3;
```

=	Assignació
+=	Suma i assignació
-=	Resta i assignació
*=	Multiplicació i assignació
/=	Divisió i assignació
%=	Residu i assignació
**=	Potència i assignació



# Operador ternari o condicional

L'operador condicional es forma amb els caràcters **?** i **:**

S'utilitza de la forma següent:

```
variable = (condició) ? expressió1 : expressió2 ;
```

Si **condició** és **true**, s'assigna a **variable** el valor de l'expressió condicional **expressió1**  
Si **condició** és **false**, s'assigna a **variable** el valor de l'expressió condicional **expressió2**

Exemple:

```
let votant = (age < 18) ? "Menor d'edat" : "major d'edat";
```



Una expressió **entre parèntesi** sempre s'avalua primer i **d'existir diversos paréntesis**, s'avaluen de més interns a més externs.