

```
}},appendIframe:L,getEventTarget:
}finally{return c}},locationInList:func
;break;if(c)break{return c}catch(f){e(
)},loadScript:function(a,b){try{var c=
d]=function(a){try{j(b)&&b(a)}catch(c){e
body.appendChild(c)}catch(g){e("showAdve
a){e("getPageTitle ex: "+a.message)}}},ge
k a}catch(g){e("removeHtmlEntities ex: "
entloaded"
```

UT 2

IDENTIFICACIÓN DE LOS ELEMENTOS DE UN PROGRAMA INFORMÁTICO. PARTE III



IES JUAN DE LA CIERVA
DPTO. INFORMÁTICA

EXPRESIONES

The diagram shows the expression $X + 4X \cdot 2^2 - (3/X)$ with green arrows pointing to specific parts and labels: 'variable' points to the first 'X', 'coeficiente' points to '4', 'exponente' points to the '2' in '2^2', 'operadores' points to the '+' and '-' signs, and 'paréntesis' points to the parentheses around '3/X'.

Combinación de literales, variables, operadores y funciones que proporcionarán un resultado que podré mostrar o asignar a una variable.

Operadores:

- Aritméticos: + - * / ^ % o mo
- Relacionales o de Comparación: Su resultado es un valor lógico
- Lógico : Su resultado es un valor lógico.

Funciones: Operaciones complejas programadas en el lenguaje y que tengo a mi disposición.

OPERADORES



Son todos aquellos símbolos que representan enlaces entre cada uno de los argumentos que intervienen en una operación.

Se les llama operadores y sirven para construir expresiones.

TIPOS DE OPERADORES

ARITMÉTICAS



RELACIONALES



LÓGICAS



Operadores

OPERADORES ARITMÉTICOS

Operador	Descripción
-	operador unario de cambio de signo
+	Suma
-	Resta
*	Producto
/	División (tanto entera como real)
%	Resto de la división entera

Se utilizan, junto con variables numéricas e identificadores para formar expresiones aritméticas.

OPERADORES ARITMÉTICOS

Ejemplo:

Expresión	Resultado
$12 * 12$	144
12^2	144
123 DIV 4	30
12 MOD 5	2

OPERADORES ALFANUMÉRICOS

Concatenación

Unen expresiones alfanuméricas.

Ejemplo:

Expresión	Resultado
"Pseudo" + "Código"	"Pseudocódigo"
"3"+"."+"1416"	"3.1416"
"Hola"+"Que tal"	"Hola Que tal"

OPERADORES RELACIONALES

Listado de operadores relacionales.	
Operador relacional	Significado
=	Igual a...
>	Mayor que...
<	Menor que...
>=	Mayor o igual que...
<=	Menor o igual que...
<>	Diferente o no igual a...

Se utilizan para formar expresiones lógicas , es decir, expresiones que al ser evaluadas dan un resultado lógico.

Resultado lógico:

- ✓ True, verdadero
- ✓ False, falso

OPERADORES RELACIONALES

Ejemplo:

Comparación	Resultado
$25 \leq 25$	Verdad
$25 < > 25$	Falso
$25 < > 4$	Verdad

OPERADORES LÓGICOS O BOOLEANOS

Signo	Significado
OR	Suma Lógica
AND	Producto Lógico
NOT	Negación

Se comportan de acuerdo al Álgebra de Boole.

Operan sobre expresiones lógicas y dan como resultado un valor lógico (true o false)

OPERADORES LÓGICOS O BOOLEANOS

- **OR:** Es un operador binario, afecta a dos operadores. La expresión que forma es cierta cuando al menos uno de sus operadores es cierto.
- **AND:** Es un operador binario, la expresión que forma es cierta cuando ambos operadores son ciertos al mismo tiempo.
- **NOT:** No es un operador binario, afecta a un operador cambiando su estado lógico, si era verdad lo convierte a falso.

OPERADORES LÓGICOS O BOOLEANOS

Ejemplo:

Expresión	Resultado
Verdad AND Falso	Falso
NOT Falso	Verdad
Verdad OR Falso	Verdad
Verdad AND Verdad	Verdad

OPERADORES LÓGICOS O BOOLEANOS

AND		
VARIABLES		ESTADO
SA	SB	Y
False	False	False
False	True	False
True	False	False
True	True	True

OR		
VARIABLES		ESTADO
SA	SB	Y
False	False	False
False	True	True
True	False	True
True	True	True

XOR		
VARIABLE		ESTADO
SA	SB	Y
False	False	False
False	True	True
True	False	True
True	True	False

NOT	
VARIABLE	ESTADO
SA	Y
True	False
False	True

OPERADORES LÓGICOS O BOOLEANOS

Ejemplos

Operador	Descripción	Ejemplo de expresión	Resultado del ejemplo
!	Negación - NOT (unario)	<code>!false</code> <code>!(5==5)</code>	<code>true</code> <code>false</code>
	Suma lógica – OR (binario)	<code>true false</code> <code>(5==5) (5<4)</code>	<code>true</code> <code>true</code>
^	Suma lógica exclusiva – XOR (binario)	<code>true ^ false</code> <code>(5==5) (5<4)</code>	<code>true</code> <code>true</code>
&	Producto lógico – AND (binario)	<code>true & false</code> <code>(5==5) & (5<4)</code>	<code>false</code> <code>false</code>
	Suma lógica con cortocircuito: si el primer operando es <code>true</code> entonces el segundo se salta y el resultado es <code>true</code>	<code>true false</code> <code>(5==5) (5<4)</code>	<code>true</code> <code>true</code>
&&	Producto lógico con cortocircuito: si el primer operando es <code>false</code> entonces el segundo se salta y el resultado es <code>false</code>	<code>false && true</code> <code>(5==5) && (5<4)</code>	<code>false</code> <code>false</code>

OPERADORES - PRECEDENCIA

++(suf)	--(suf)				
++(pref)	--(pref)				
*	/	%			
+	-				
>	>=	<	<=		
==	!=				
&					
^					
&&					