

# JAVASCRIPT OPERATORS



## JavaScript Operators

Assignment Operators

Logical Operators

Ternary Operators

Unary Operators

Relational Operators

String Operators

Arithmetic Operators

Comparison Operators

Bitwise Operators

Comma Operators

Relational Operators

String Operators

## Arithmetic Opeartors

Operator	Description	Example
+	Addition	10+20 = 30
-	Subtraction	20-10 = 10
*	Multiplication	10*20 = 200
/	Division	20/10 = 2
%	Modulus (Remander)	20%10 = 0
++	Increment	var a=10; a++; Now a = 11
--	Decrement	var a=10; a--; Now a = 9

## Comparision Opeartors

Operator	Description	Example
==	Is equal to	10==20 = false
===	Identical (equal and of same type)	10===20 = false
!=	Not equal to	10!=20 = true
>=	Not identical	20!=20 = false
>	Greater than	20>10 = true
>=	Greater than or equal to	20>=10 = true
<	Less than	20<10 = false
<=	Less than or equal to	20<=10 = false

## Bitwise Opeartors

Operator	Description	Example
&	Bitwise AND	(10)&20 & (20)&(33) = false
	Bitwise OR	(10) 20  (20) (33) = false
^	Bitwise XOR	(10)^20 ^ (20)^33 = false
~	Bitwise NOT	~10 = -11
<<	Bitwise Left Shift	10<<2 = 40
>>	Bitwise Right Shift	10>>2 = 2
>>>	Bitwise Right Shift with Zero	10>>>2 = 2

## Logical Opeartors

Operator	Description	Example
&&	Logical AND	(10==20 && 20==33) = false
	Logical OR	(10==20    20==33) = false
!	Logical Not	!(10==20) = true

## Assignment Opeartors

Operator	Description	Example
=	Assign	10=10 = 20
+=	Add and assign	var a=10; a+=20; Now a = 30
-=	Subtract and assign	var a=20; a-=10; Now a = 10
*=	Multiply and assign	var a=10; a*=20; Now a = 200
/=	Divide and assign	var a=10; a/=2; Now a = 5
%=	Modulus and assign	var a=10; a%=2; Now a = 0

## Ternary Opeartors

The ternary operator has three operands. It is the simplified operator of if/else.

Eg: let result = (PMarks > 39) ?"Pass" : "Fail";

Name	Description	Syntax
<b>Ternary Operator(?:)</b>	It is like the short form of the if-else condition.	Y = ? A : B If the condition is true then Y = A otherwise Y = B

## Comma Opeartors

It mainly evaluates its operands from left to right sequentially and returns the value of the rightmost operand.

Name	Description	Syntax
<b>comma operator(,)</b>	When a comma operator is placed in an expression, it executes each expression and returns the rightmost expression.	Expression1, Expression2, Expression3, ...so on

## Unary Opeartors

A unary operation is an operation with only one operand.

Name	Description	Syntax
JavaScript typeof	It returns the operand type. The possible types that exist in javascript are undefined, Object, boolean, number, string, symbol, and function.	typeof variable
Delete	This operator is more specifically used to delete JavaScript object properties.	delete object // or delete object.property // or delete object['property']

Relational Opeartors

They are used to compare its operands and determine the relationship between them. They return a Boolean value (true or false) based on the comparison result.

Name	Description	Syntax
in	The in operator returns true if the specified property is in the specified object.	propertyName in objectName
instanceof	The instanceof operator returns true if the specified object is of the specified object type.	objectName instanceof objectType

String Opeartors

They are used to join strings or combine strings with other data types.

Name	Description	Syntax
concatenation operator (+)	It concatenates two string values together, returning another string that is the union of the two operand strings.	str1 + str2

BigInt Opeartors

Name	Description	Syntax
BigInt	It returns a new BigInt object that represents the original value.	BigInt(value);