# JAVASCRIPT

JavaScript is the scripting language of the Web.

JavaScript is used in millions of Web pages to add functionality, validate forms, detect browsers, and much more.

## Introduction to JavaScript

JavaScript is used in millions of Web pages to improve the design, validate forms, detect browsers, create cookies, and much more.

JavaScript is the most popular scripting language on the Internet, and works in all major browsers, such as Internet Explorer, Mozilla Firefox, and Opera.

## What is JavaScript?

JavaScript was designed to add interactivity to HTML pages

JavaScript is a scripting language

A scripting language is a lightweight programming language

JavaScript is usually embedded directly into HTML pages

JavaScript is an interpreted language (means that scripts execute without preliminary compilation) Everyone can use JavaScript without purchasing a license

Java and JavaScript are two completely different languages in both concept and design!

Java (developed by Sun Microsystems) is a powerful and much more complex programming language - in the same category as C and C++.

## What can a JavaScript Do ?

**JavaScript gives HTML designers a programming tool -** HTML authors are normally not programmers, but JavaScript is a scripting language with a very simple syntax! Almost anyone can put small "snippets" of code into their HTML pages

**JavaScript can put dynamic text into an HTML page -** A JavaScript statement like this: document.write("<h1>" + name + "</h1>") can write a variable text into an HTML page

**JavaScript can react to events -** A JavaScript can be set to execute when something happens, like when a page has finished loading or when a user clicks on an HTML element

**JavaScript can read and write HTML elements -** A JavaScript can read and change the content of an HTML element

**JavaScript can be used to validate data -** A JavaScript can be used to validate form data before it is submitted to a server. This saves the server from extra processing

**JavaScript can be used to detect the visitor's browser** - A JavaScript can be used to detect the visitor's browser, and - depending on the browser - load another page specifically designed for that browser

**JavaScript can be used to create cookies** - A JavaScript can be used to store and retrieve information on the visitor's computer.

## JavaScript Variables

Variables are "containers" for storing information.

JavaScript variables are used to hold values or expressions.

A variable can have a short name, like x, or a more descriptive name, like carname.

Rules for JavaScript variable names:

Variable names are case sensitive (y and Y are two different variables)
Variable names must begin with a letter or the underscore character

**Note:** Because JavaScript is case-sensitive, variable names are case-sensitive.

***Example***

A variable's value can change during the execution of a script. You can refer to a variable by its name to display or change its value.

```
<html>
<body>
<script type="text/javascript">
var firstname;
firstname="Welcome";
document.write(firstname);
document.write("<br />");
firstname="XYZ";
document.write(firstname);
</script>

<p>The script above declares a variable,
assigns a value to it, displays the value, change the value,
and displays the value again.</p>

</body>
</html>
```

**Output :**

```
Welcome
XYZ
```

The script above declares a variable, assigns a value to it, displays the value, change the value, and displays the value again.

### *Declaring (Creating) JavaScript Variables*

Creating variables in JavaScript is most often referred to as "declaring" variables.

You can declare JavaScript variables with the **var statement**:

```
var x;
var carname;
```

After the declaration shown above, the variables are empty (they have no values yet).

However, you can also assign values to the variables when you declare them:

```
var x=5;
var carname="Scorpio";
```

After the execution of the statements above, the variable **x** will hold the value **5**, and **carname** will hold the value **Scorpio**.

**Note:** When you assign a text value to a variable, use quotes around the value.

### *Assigning Values to Undeclared JavaScript Variables*

If you assign values to variables that have not yet been declared, the variables will automatically be declared.

These statements:

```
x=5;
carname="Scorpio";
```

have the same effect as:

```
var x=5;
var carname="Scorpio";
```

### *Redeclaring JavaScript Variables*

If you redeclare a JavaScript variable, it will not lose its original value.

```
var x=5;
var x;
```

After the execution of the statements above, the variable x will still have the value of 5. The value of x is not reset (or cleared) when you redeclare it.

## DataTypes

Numbers - are values that can be processed and calculated. You don't enclose them in quotation marks. The numbers can be either positive or negative.
Strings - are a series of letters and numbers enclosed in quotation marks. JavaScript uses the string literally; it doesn't process it. You'll use strings for text you want displayed or values you want passed along.
Boolean (**true**/**false**) - lets you evaluate whether a condition meets or does not meet specified criteria.
Null - is an empty value. **null** is not the same as 0 **--** 0 is a real, calculable number, whereas **null** is the **absence of any value**.

**Data Types**

| TYPE | EXAMPLE |
|---|---|
| **Numbers** | Any number, such as 17, 21, or 54e7 |
| **Strings** | "Greetings!" or "Fun" |
| **Boolean** | Either true or false |
| **Null** | A special keyword for exactly that – the null value (that is, nothing) |

## *JavaScript Arithmetic*

As with algebra, you can do arithmetic operations with JavaScript variables:

```
y=x-5;
z=y+5;
```

## JavaScript Operators

The operator = is used to assign values.

The operator + is used to add values.

The assignment operator = is used to assign values to JavaScript variables.

The arithmetic operator + is used to add values together.

```
y=5;
z=2;
x=y+z;
```

The value of x, after the execution of the statements above is 7.

### ***JavaScript Arithmetic Operators***

Arithmetic operators are used to perform arithmetic between variables and/or values.

Given that **y=5**, the table below explains the arithmetic operators:

| Operator | Description | Example | Result |
|---|---|---|---|
| + | Addition | x=y+2 | x=7 |
| - | Subtraction | x=y-2 | x=3 |
| * | Multiplication | x=y*2 | x=10 |
| / | Division | x=y/2 | x=2.5 |
| % | Modulus (division remainder) | x=y%2 | x=1 |
| ++ | Increment | x=++y | x=6 |
| -- | Decrement | x=--y | x=4 |

### ***JavaScript Assignment Operators***

Assignment operators are used to assign values to JavaScript variables.

Given that **x=10** and **y=5**, the table below explains the assignment operators:

| Operator | Example | Same As | Result |
|---|---|---|---|
| = | x=y |  | x=5 |
| += | x+=y | x=x+y | x=15 |
| -= | x-=y | x=x-y | x=5 |
| *= | x*=y | x=x*y | x=50 |
| /= | x/=y | x=x/y | x=2 |
| %= | x%=y | x=x%y | x=0 |

### ***The + Operator Used on Strings***

The + operator can also be used to add string variables or text values together.

To add two or more string variables together, use the + operator.

```
txt1="What a very";
txt2="nice day";
txt3=txt1+txt2;
```

After the execution of the statements above, the variable txt3 contains "What a verynice day".

To add a space between the two strings, insert a space into one of the strings:

```
txt1="What a very ";
txt2="nice day";
txt3=txt1+txt2;
```

or insert a space into the expression:

```
txt1="What a very";
txt2="nice day";
txt3=txt1+" "+txt2;
```

After the execution of the statements above, the variable txt3 contains:

"What a very nice day"

***Adding Strings and Numbers***

Look at these examples:

```
x=5+5;
document.write(x);

x="5"+"5";
document.write(x);

x=5+"5";
document.write(x);

x="5"+5;
document.write(x);
```

The rule is:

**If you add a number and a string, the result will be a string.**

### JavaScript Comparison and Logical Operators

**Comparison and Logical operators are used to test for true or false.**

## ***Comparison Operators***

Comparison operators are used in logical statements to determine equality or difference between variables or values.

Given that **x=5**, the table below explains the comparison operators:

| Operator | Description | Example |
|---|---|---|
| == | is equal to | x==8 is false |
| === | is exactly equal to (value and type) | x===5 is true<br>x==="5" is false |
| != | is not equal | x!=8 is true |
| > | is greater than | x>8 is false |
| < | is less than | x<8 is true |
| >= | is greater than or equal to | x>=8 is false |
| <= | is less than or equal to | x<=8 is true |

### *How Can it be Used*

Comparison operators can be used in conditional statements to compare values and take action depending on the result:

```
if (age<18) document.write("Too young");
```

You will learn more about the use of conditional statements in the next chapter of this tutorial.

## ***Logical Operators***

Logical operators are used to determine the logic between variables or values.

Given that **x=6 and y=3**, the table below explains the logical operators:

| Operator | Description | Example |
|---|---|---|
| && | and | (x < 10 && y > 1) is true |
| \|\| | or | (x==5 \|\| y==5) is false |
| ! | not | !(x==y) is true |

## ***Conditional Operator***

JavaScript also contains a conditional operator that assigns a value to a variable based on some condition.

**Syntax**

```
variablename=(condition)?value1:value2
```

**Example**

```
greeting=(visitor=="PRES")?"Dear President ":"Dear ";
```

If the variable **visitor** has the value of "PRES", then the variable **greeting** will be assigned the value "Dear President " else it will be assigned "Dear".