

Introducción a Javascript

ARQUITECTURA DE LA INFORMACIÓN

¿Qué es Javascript?

JavaScript es un lenguaje de programación interpretado (no necesita compilarse)

Se utiliza principalmente para **crear páginas web dinámicas**.

Lenguaje orientado a objetos, pero utilizado habitualmente en navegadores web, similares en sintaxis a Java y C



Acciones dinámicas

- Efectos en Textos
- Animaciones
- Acciones que se activan a pulsar un botón
- Autocompletar en caja de texto
- Validaciones en formularios

Ejemplos de código

Muestra alerta al dar clic:

https://www.w3schools.com/js/tryit.asp?filename=tryjs_alert

Muestra mensaje al dar clic:

[https://www.w3schools.com/jsref/tryit.asp?filename=tryjsref onclick](https://www.w3schools.com/jsref/tryit.asp?filename=tryjsref_onclick)

Valida campo en formulario:

[https://www.w3schools.com/js/tryit.asp?filename=tryjs_validation js](https://www.w3schools.com/js/tryit.asp?filename=tryjs_validation_js)

Autocompletar:

[https://www.w3schools.com/howto/tryit.asp?filename=tryhow js autocomplete](https://www.w3schools.com/howto/tryit.asp?filename=tryhow_js_autocomplete)

Variables

Se utiliza la palabra reservada **var** para definir variables, no es necesario definir el tipo de dato.

Tipos de Variables

Numéricas

```
var iva = 16;           // variable tipo entero
var total = 234.65;    // variable tipo decimal
```

Cadenas de texto

```
var mensaje = "Bienvenido a nuestro sitio web";
var letraSeleccionada = 'c';
```

Arrays → Colección de variables

```
var dias = ["Lunes", "Martes", "Miércoles", "Jueves", "Viernes",
"Sábado", "Domingo"];
var diaSeleccionado = dias[0];    // diaSeleccionado = "Lunes"
var otroDia = dias[5];           // otroDia = "Sábado"
```

Booleanos

```
var clienteRegistrado = false;
var ivaIncluido = true;
```

```
1 // JavaScript File
2 var n=1; // variable global
3
4 var suma = function (m){
5     var cuadrado = m*m; //Variable local
6     n+= cuadrado;
7 }
8 suma (2);
9 n;
```

Operadores

Permiten manipular variables

Asignación

```
var iva = 16;
```

Incremento y decremento (solo para var numéricas)

```
var numero = 5;  
++numero;
```

```
var num2 = 7;  
--num2;
```

Matemáticos

suma (+), resta (-), multiplicación (*) y división (/)

Lógicos

Negación (!), AND (&&), OR (||)

Relacionales

Mayor que (>), Menor que (<)

Sentencias de control

Estructura if...else

```
if(condicion) {  
    ...  
}  
  
else {  
    ...  
}  
  
var nombre = "";  
  
if(nombre == "") {  
    alert("Aún no nos has dicho tu nombre");  
}  
else {  
    alert("Hemos guardado tu nombre");  
}
```

Estructura for

```
for(inicializacion  
; condicion;  
actualizacion) {  
    ...  
}  
  
var mensaje = "Hola, estoy dentro de un bucle";  
  
for(var i = 0; i < 5; i++) {  
    alert(mensaje);  
}
```

Estructura for...in

```
for(indice in  
array) {  
    ...  
}
```

```
var dias = ["Lunes", "Martes", "Miércoles", "Jueves", "Viernes", "Sábado", "Domingo"];  
var i;  
for(i in dias) {  
    alert(dias[i]);  
}
```

Otras sentencias de control

switch, while, do...while, return, break, continue

Funciones

Sirven para realizar tareas que se puedan repetir. Se definen con la palabra reservada **function**

Ejemplo:

```
function nombre_funcion() {  
    ...//Instrucciones a ejecutar  
}
```

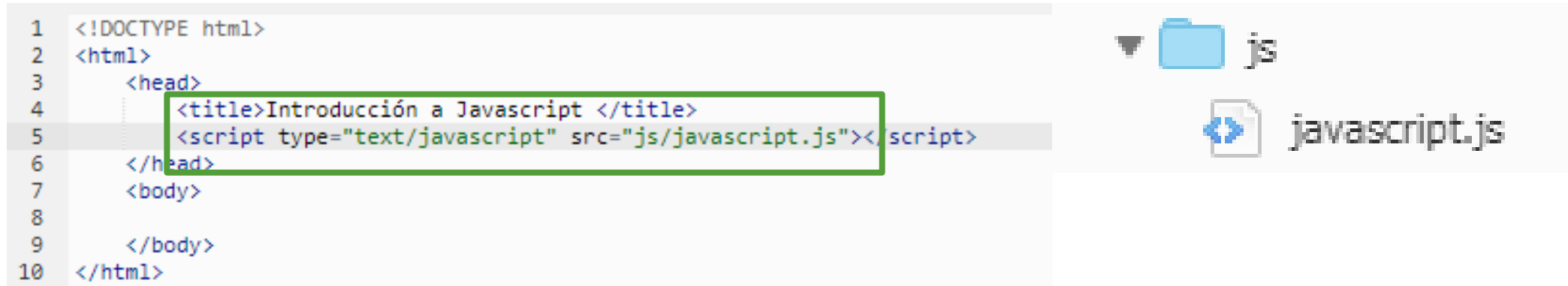
Argumentos y valores de retorno

```
function nombre_funcion(parameters) {  
    ...//Instrucciones a ejecutar  
  
    return valorDeRetorno;  
}
```

```
function myFunction(a, b) {  
    return a * b;  
}
```

Definir Javascript en un archivo externo

Las instrucciones JavaScript se pueden incluir en un archivo externo que los documentos HTML enlazan mediante etiquetas `<script></script>` en el `<head></head>` o en el `<body></body>`



Ventajas

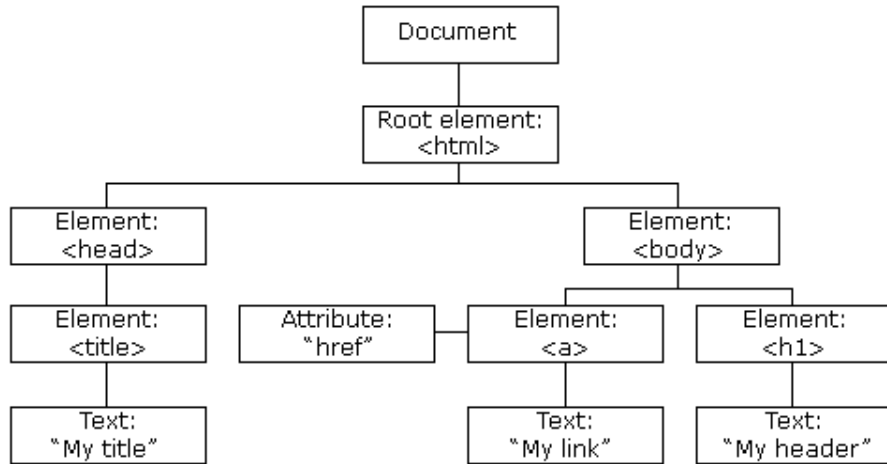
Separa HTML y código.

Facilita la lectura y el mantenimiento de HTML y JavaScript.

Los archivos JavaScript en caché pueden acelerar la carga de páginas

HTML DOM (Modelo de Objetos del Documento)

Cuando se carga una página web, el navegador crea el DOM



Con el DOM, JavaScript obtiene todo lo que necesita para crear HTML dinámico:

Puede cambiar todos los elementos HTML en la página

Puede cambiar todos los atributos HTML en la página

Puede cambiar todos los estilos CSS en la página

Puede eliminar elementos y atributos HTML existentes

Puede agregar nuevos elementos y atributos HTML

Puede reaccionar a todos los eventos HTML existentes en la página

Puede crear nuevos eventos HTML en la página

Métodos HTML DOM

Los métodos de HTML DOM son **acciones** que puede realizar (en Elementos HTML).

Una **propiedad** es un valor que puede obtener o establecer (como cambiar el contenido de un elemento HTML).

El método getElementById

La forma más común de acceder a un elemento HTML es usar el id del elemento.

var elem= document.getElementById("demo") <p id="demo">Contenido </p>

La propiedad innerHTML

La propiedad innerHTML se puede usar para obtener o cambiar cualquier elemento HTML, incluidos <html> y <body>.

```
<p id="demo"></p>

<script>
document.getElementById("demo").innerHTML = "Hello World!";
</script>
```

Métodos para buscar elementos HTML

<code>document.getElementById(<i>id</i>)</code>	Encuentra elementos por id
<code>document.getElementsByTagName(<i>name</i>)</code>	Encuentra elementos por etiqueta html
<code>document.getElementsByClassName(<i>name</i>)</code>	Encuentra elementos por clase

Cambio de elementos HTML

<code><i>element</i>.innerHTML = nuevo conteido</code>	Cambia el contenido del elemento HTML
<code><i>element</i>.attribute = nuevo valor</code>	Cambia el atributo de un elemento HTML
<code><i>element</i>.setAttribute(<i>attribute</i>, <i>value</i>)</code>	Cambia el valor de un atributo de un elemento HTML
<code><i>element</i>.style.property = nuevo stilo</code>	Cambia el estilo de un elemento HTML

Agregar y eliminar elementos HTML

<code>document.createElement(<i>element</i>)</code>	Crea un elemento HTML
<code>document.removeChild(<i>element</i>)</code>	Remueve un elemento HTML
<code>document.appendChild(<i>element</i>)</code>	Agrega un elemento HTML
<code>document.replaceChild(<i>element</i>)</code>	Remplaza un elemento html

Escribir en el documento HTML

<code>document.write(texto)</code>	Escribe cadenas en el cuerpo de un documento HTML
------------------------------------	---

Frameworks basados en Javascript

- ❑ Angular → Lanzado por Google y es open source
- ❑ React → es mas bien una librería, Facebook e Instagram están desarrolladas en React
- ❑ Vue → Basado en patrón de diseño MVC
- ❑ JQuery → Es un framework clásico
- ❑ Ember → Basado en patrón de diseño MVC
- ❑ Backbone → Reddedit esta basado en este framework
- ❑ Mercury → Relativamente nuevo