

Curso de Programación Web – Javascript

Documentación: Escritorio/Material Web Nivel I/librosweb.es/ introduccion_javascript_2caras.pdf , introduccion_ajax_2caras.pdf



| Item | Descripción |
|---|--|
| <script> </script> | Apertura y cierre de código javascript. Atributos: type , src |
| var | La palabra reservada var declara una variable local. Ejemplo: var i=0 |
| if(condicion) { ... } | Se emplea para tomar decisiones en función de una condición |
| if(condicion) {...} else { ... } | Si la condición se cumple (es decir, si su valor es true) se ejecutan todas las instrucciones que se encuentran dentro del if(). Si la condición no se cumple (es decir, si su valor es false) se ejecutan todas las instrucciones contenidas en else { }. |
| Operaciones Relacionales | mayor que (>), menor que (<), mayor o igual (>=), menor o igual (<=), igual que (==) y distinto de (!=). |
| + | Se emplea para concatenar varias cadenas de texto o sumar elementos |
| alert() | La instrucción alert() es una de las utilidades que incluye JavaScript y permite mostrar un mensaje en la pantalla del usuario. |
| function nombre_funcion() {...} | Las funciones en JavaScript se definen mediante la palabra reservada function, seguida del nombre de la función. |
| Funciones Anónimas | Las funciones anónimas son ideales para los casos en los que se necesita definir funciones sencillas que sólo se utilizan una vez y para las que no es necesario crear una función tradicional con nombre. Ejemplo: var miFuncion = function(a, b) { return a+b; } |
| while(condicion) { ... } | La estructura while permite crear bucles que se ejecutan ninguna o más veces, dependiendo de la condición indicada |
| do { ... } while(condicion); | El bucle de tipo do...while es muy similar al bucle while, salvo que en este caso siempre se ejecutan las instrucciones del bucle al menos la primera vez. |
| for(inicializacion; condicion; actualizacion) {...} | La idea del funcionamiento de un bucle for es la siguiente: <i>"mientras la condición indicada se siga cumpliendo, repite la ejecución de las instrucciones definidas dentro del for. Además, después de cada repetición, actualiza el valor de las variables que se utilizan en la condición"</i> . |
| switch(variable) { case valor_1:...break; } | La estructura switch está especialmente diseñada para manejar de forma sencilla múltiples condiciones sobre la misma variable. |

Curso de Programación Web – Javascript

Documentación: Escritorio/Material Web Nivel I/librosweb.es/ introduccion_javascript_2caras.pdf , introduccion_ajax_2caras.pdf



| Item | Descripción |
|---------------------------|--|
| isNaN() | isNaN(), permite proteger a la aplicación de posibles valores numéricos no definidos. Ejemplo: 0 / 0 |
| length | En una cadena de caracteres calcula la longitud de una cadena de texto, en un vector calcula el número de elementos de un array. |
| split(separador) | convierte una cadena de texto en un array de cadenas de texto. La función parte la cadena de texto determinando sus trozos a partir del carácter separador indicado |
| substring(inicio, final), | extrae una porción de una cadena de texto. El segundo parámetro es opcional. Si sólo se indica el parámetro inicio, la función devuelve la parte de la cadena original correspondiente desde esa posición hasta el final |
| toLowerCase() | transforma todos los caracteres de la cadena a sus correspondientes caracteres en minúsculas |
| toUpperCase(), | transforma todos los caracteres de la cadena a sus correspondientes caracteres en mayúsculas |
| concat() | En vectores se emplea para concatenar los elementos de varios arrays, en cadenas de caracteres concatena al igual que (+) |
| toFixed(digitos) | devuelve el número original con tantos decimales como los indicados por el parámetro digitos y realiza los redondeos necesarios |
| parseInt() | Trata el dato enviado como parámetro como un entero |
| parseFloat() | Trata el dato enviado como parámetro como un flotante |
| getElementsByTagName() | La función getElementsByTagName(nombreEtiqueta) obtiene todos los elementos de la página XHTML cuya etiqueta sea igual que el parámetro que se le pasa a la función. |
| getElementsByTagName() | La función getElementsByTagName() buscan los elementos cuyo atributo name sea igual al parámetro proporcionado |
| getElementById() | La función getElementById() devuelve el elemento XHTML cuyo atributo id coincide con el parámetro indicado en la función. |
| | |

Curso de Programación Web – Javascript

Documentación: Escritorio/Material Web Nivel I/librosweb.es/ introduccion_javascript_2caras.pdf , introduccion_ajax_2caras.pdf



| Item | Descripción |
|-----------------------|---|
| XMLHttpRequest | Todas las aplicaciones realizadas con técnicas de AJAX deben instanciar en primer lugar el objeto XMLHttpRequest, que es el objeto clave que permite realizar comunicaciones con el servidor en segundo plano, sin necesidad de recargar las páginas. |
| readyState | Propiedad. Valor numérico (1,2,3 o 4) que almacena el estado de la petición |
| responseText | El contenido de la respuesta del servidor en forma de cadena de texto |
| status | El código de estado HTTP devuelto por el servidor (200 para una respuesta correcta, 404 para "No encontrado", 500 para un error de servidor, etc.) |
| onreadystatechange | Responsable de manejar los eventos que se producen. Se invoca cada vez que se produce un cambio en el estado de la petición HTTP. Normalmente es una referencia a una función JavaScript |
| open("metodo", "url") | Establece los parámetros de la petición que se realiza al servidor. Los parámetros necesarios son el método HTTP empleado y la URL destino (puede indicarse de forma absoluta o relativa) |
| send(contenido) | Realiza la petición HTTP al servidor |
| | |
| | |
| | |
| | |
| | |
| | |