



APRENDERAPROGRAMAR.COM

REDIRECCIONAR Y
RECARGAR WEBS CON
JAVASCRIPT.
WINDOW.LOCATION.
HREF, HOSTNAME,
ASSIGN, RELOAD,
REPLACE. (CU01171E)

Sección: Cursos

Categoría: Tutorial básico del programador web: JavaScript desde cero

Fecha revisión: 2029

Resumen: Entrega nº71 del Tutorial básico "JavaScript desde cero".

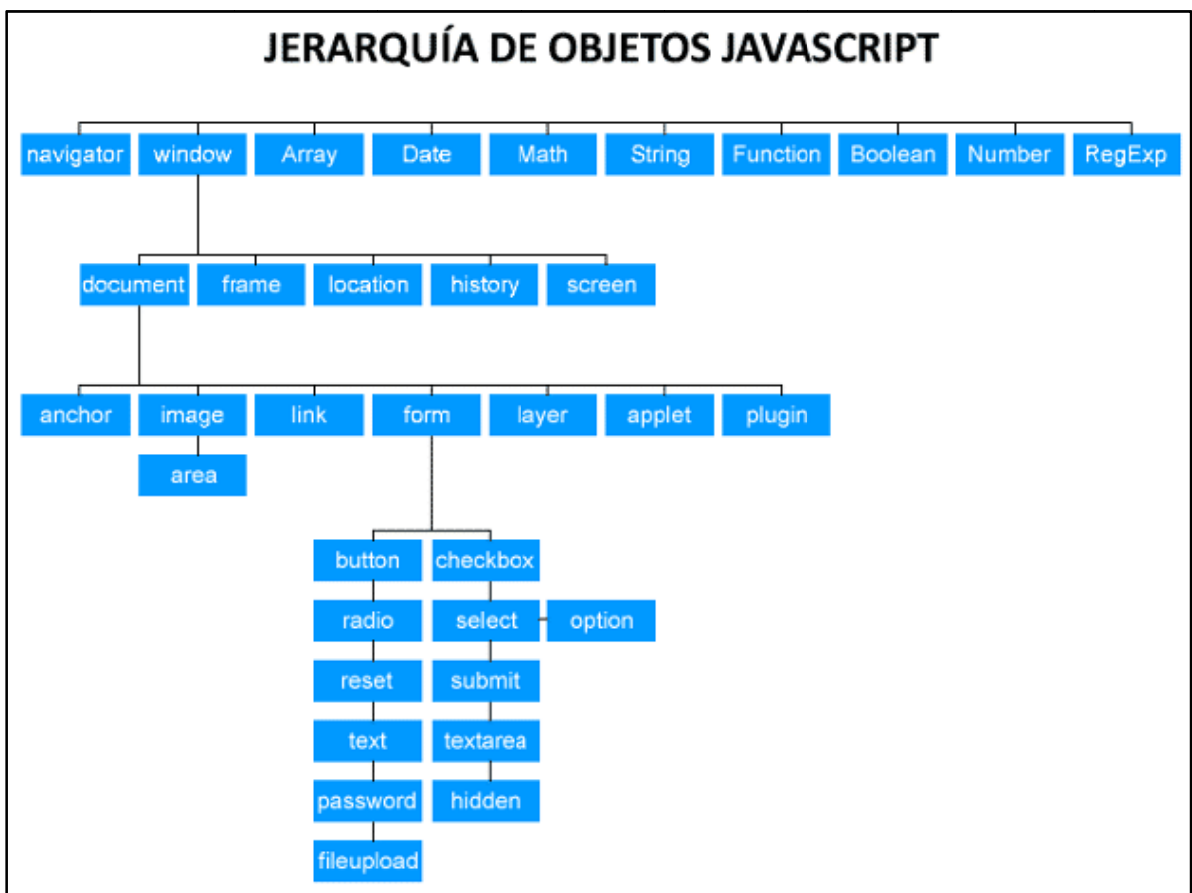
Autor: César Krall

WINDOW.LOCATION

Dentro de la jerarquía de objetos JavaScript existe un objeto denominado location, que es una propiedad de window. A su vez, se dice que existen objetos de tipo Location, siendo el objeto window.location un ejemplo de este tipo de objetos. Con este objeto podemos extraer información de la url (por ejemplo parámetros recibidos), redireccionar a otra página web, ó refrescar la web actual.



Recordemos que de forma aproximada puede representarse la jerarquía de objetos JavaScript de la siguiente manera.



PROPIEDAD WINDOW.LOCATION

La propiedad window.location devuelve un objeto de tipo Location con información relacionada con la URL del documento actual.

Vamos a citar aquí algunas de las **propiedades** de los objetos Location. Tener en cuenta que algunas de ellas no son soportadas por algunos navegadores:

PROPIEDAD	UTILIDAD	EJEMPLOS aprenderaprogramar.com
href	Devuelve un string con la url completa. window.location es equivalente a document.location pero recomendamos usar window.location.	<pre> alert ('url actual: '+window.location.href); alert ('url actual: '+location.href); // Devuelve la url actual completa </pre>
protocol	Devuelve el protocolo de la URL, normalmente http ó https. Si trabajamos en local será file:	<pre> alert ('protocolo actual: '+window.location.protocol); //Devuelve por ejemplo http </pre>
host	Devuelve el nombre del servidor, normalmente como nombre de dominio, y si existe un puerto especificado también, separado con .:	<pre> alert ('host actual: '+window.location.host); //Devuelve por ejemplo http://aprenderaprogramar.com </pre>
hostname	Devuelve el dominio o servidor.	<pre> alert ('el hostname actual: '+window.location.hostname); //Devuelve por ejemplo http://aprenderaprogramar.com </pre>
pathname	Devuelve la ruta excluido el nombre de dominio o servidor.	<pre> alert ('El pathname actual: '+window.location.pathname); //Por ejemplo devuelve /ejemplo1.html </pre>
search	Devuelve la cadena que representa los parámetros que existen en la url.	<pre> alert ('Los parámetros en url: '+window.location.search); // Por ejemplo: Los parámetros en url: ?name=pepe&apellido1=perez&pais=ecuador </pre>
hash	Devuelve # seguido de un texto si en la url existe un #seguido de un texto (ancla de localización del fragmento html).	<pre> alert ('Valor localizador #: '+ window.location.hash); // Por ejemplo: #final </pre>
origin	Devuelve una cadena que representa la raíz del sitio web	<pre> alert ('Valor origin: '+ window.location.origin); //Devuelve por ejemplo http://aprenderaprogramar.com </pre>

Vamos a citar aquí algunos de los **métodos** de los objetos Location. Tener en cuenta que algunas de ellas no son soportadas por algunos navegadores:

MÉTODO	UTILIDAD	EJEMPLOS aprenderaprogramar.com
assign (urlDestino)	Asigna una url a la ventana actual, lo que genera que se cargue la urlDestino. El usuario puede volver atrás con el botón back del navegador.	<pre> alert ('Transfiriendo a aprenderaprogramar: '); window.location.assign('http://aprenderaprogramar.com'); //Da lugar a que se cargue la url indicada con posibilidad de volver atrás con el botón back </pre>

MÉTODO	UTILIDAD	EJEMPLOS aprenderaprogramar.com
reload (opcSiDesdeServidor)	Da lugar a la recarga de la url actual. Si se indica como parámetro true, la recarga se hará desde el servidor (refresco completo). Si no se especifica o es false la recarga puede ser desde caché.	alert ('Recargando la página... '); window.location.reload(true); //Se recarga la página desde el servidor (refresco completo)
replace (urlDestino)	Asigna una url a la ventana actual, lo que genera que se cargue la urlDestino. El usuario no puede volver atrás con el botón back del navegador.	alert ('Reemplazando la página... '); window.location.replace('http://aprenderaprogramar.com'); //Da lugar a que se cargue la url indicada sin posibilidad de volver atrás con el botón back
toString()	Devuelve una cadena representativa de la url (window.location.toString() devuelve lo mismo que window.location.href)	alert ('Url actual: '+window.location.toString()); //Devuelve la url actual

EJEMPLO DE USO DE WINDOW.LOCATION

Supongamos una url que se construye según este patrón: `http://aprenderaprogramar.com/index.php?option=compra&moneda=euro&producto=libro1`, donde el parámetro option puede ser compra, venta o intercambio. El parámetro moneda puede ser pesoMexicano, euro, pesoArgentino, pesoColombiano, bolivar, sol, pesoChileno. Y el parámetro producto puede ser cualquier valor.

Y supongamos que queremos rescatar el valor que tenga el parámetro moneda usando JavaScript. Podemos hacerlo de esta manera. Escribe este código y guárdalo con nombre de archivo ejemplo1.html:

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
<html><head><title>Ejemplo aprenderaprogramar.com</title><meta charset="utf-8">
<script type="text/javascript">
function ejemplo() {
    msg = "";
    var moneda = extraerParametroUrl('moneda');
    if (moneda) {alert ('El valor del parámetro moneda en la url es: '+moneda)}
    else {alert ('No se recibe moneda en la url!'); }
}
function extraerParametroUrl( nombreParametro ){
    var regexS = "[\\?&]" + nombreParametro + "=[^&#]*";
    var regex = new RegExp ( regexS );
    var tmpURL = window.location.href;
    var results = regex.exec( tmpURL );
    if( results == null ) { return ""; } else {return results[1];}
}
</script>
</head>
<body><div id="cabecera"><h2>Cursos aprenderaprogramar.com</h2><h3>Ejemplos JavaScript</h3></div>
<div style="color:blue; margin:20px;" id="pulsador" onclick="ejemplo()"><a href="#"> Probar</a> </div>
</body></html>
    
```

A continuación visualiza el archivo en tu navegador. El resultado esperado es que al pulsar el enlace "Probar" se muestre: << No se recibe moneda en la url >>

Si añadimos el parámetro al final de la ruta, por ejemplo: .../ ejemplo1.html?moneda=pesoMexicano

El resultado esperado es que al pulsar el enlace "Probar" se muestre: <<El valor del parámetro moneda en la url es: pesoMexicano>>

Este código hace uso de expresiones regulares JavaScript. Si no recuerdas cómo funcionan éstas, lee los anteriores apartados del curso que hemos dedicado a expresiones regulares.

```
Hemos definido var regexS = "[\\?&]" + nombreParametro + "=(^[&#]*)";
```

Esto representa una expresión regular JavaScript que comienza con ? ó &, viene seguido de nombreParametro, seguido de un igual, y de cualesquiera caracteres hasta alcanzar el final de la cadena ó un símbolo & ó un símbolo #, que serían las situaciones que nos indicarían que el valor del parámetro ya ha sido extraído completamente. Recordar que dentro de corchetes el símbolo ^ funciona como negación. Al ejecutar exec sobre una cadena (en este caso la URL) el índice [0] del resultado contiene el valor completo, por ejemplo <<?moneda= pesoMexicano >>, mientras que el siguiente índice contiene el match correspondiente al primer paréntesis, en este caso sería << pesoMexicano >>. El índice [2] contendría el match correspondiente al segundo paréntesis, pero aquí sólo tenemos un paréntesis con lo cual el índice [2] no tiene valor definido.

Como conclusión de este ejemplo, vemos cómo podemos rescatar parámetros de una url enviados por el método GET haciendo uso de JavaScript de varias maneras: hemos visto métodos específicos y esta otra manera mostrada en este ejemplo, que al ser más generalista asegura un buen funcionamiento en todos los navegadores.

EJERCICIO

Crea una página web que al ser invocada muestre un mensaje "Hemos cambiado de ubicación esta página. En breves momentos será redireccionado..." junto a una cuenta atrás que muestre 5, 4, 3, 2, 1 (correspondiente a 5 segundos). Tras transcurrir 5 segundos y mostrarse la cuenta atrás, el usuario debe ser redireccionada a la url <http://aprenderaprogramar.com>. Para realizar este ejercicio debes usar la propiedad window.location junto con funciones que permitan el retardo en la ejecución vistas en entregas anteriores del curso.

Para comprobar si tus respuestas y código son correctos puedes consultar en los foros aprenderaprogramar.com.

Próxima entrega: CU01172E

Acceso al curso completo en aprenderaprogramar.com -- > Cursos, o en la dirección siguiente:
http://aprenderaprogramar.com/index.php?option=com_content&view=category&id=78&Itemid=206