



APRENDERAPROGRAMAR.COM

XMLHTTPREQUEST AJAX:
PROPIEDADES STATUS,
ONREADYSTATECHANGE,
READYSTATE,
RESPONSETEXT O XML
(CU01207F)

Sección: Cursos

Categoría: Tutorial básico del programador web: Ajax desde cero

Fecha revisión: 2031

Resumen: Entrega nº7 del Tutorial básico "Ajax desde cero".

Autor: Alex Rodríguez

OBJETO XMLHTTPREQUEST: PROPIEDADES

El objeto predefinido XMLHttpRequest es fundamental en la forma de comunicación o tecnología que denominamos Ajax para intercambiar información en segundo plano con un servidor. En la anterior entrega del curso, hemos visto un ejemplo de código con Ajax.



Revisa el código de la anterior entrega y mantenlo "delante de ti", pues va a resultar necesario para entender lo que vamos a explicar a continuación. Ahora vamos a centrarnos en entender un fragmento concreto de código que es el siguiente:

```
<script>
function mostrarSugerencia(str) {
var xmlhttp;
var contenidosRecibidos = new Array();
var nodoMostrarResultados = document.getElementById('listaCiudades');
var contenidosAMostrar = "";

if (str.length==0) { document.getElementById("txtInformacion").innerHTML="";
nodoMostrarResultados.innerHTML = ""; return; }

xmlhttp=new XMLHttpRequest();
xmlhttp.onreadystatechange = function() {
if (xmlhttp.readyState==4 && xmlhttp.status==200) {
contenidosRecibidos = xmlhttp.responseText.split(",");
document.getElementById("txtInformacion").innerHTML=contenidosRecibidos[0];
for (var i=1; i<contenidosRecibidos.length;i++) {
contenidosAMostrar = contenidosAMostrar+'<div id="ciudades'+i+" "> <a
href="http://aprenderaprogramar.com">' + contenidosRecibidos[i]+ '</a></div>';
}
nodoMostrarResultados.innerHTML = contenidosAMostrar;
}
}

xmlhttp.open("GET"," datosCU01206F.php?pais="+str);
xmlhttp.send();
}

</script>
```

En este fragmento de código tenemos en primer lugar la declaración de una función que se llama mostrarSugerencia y que recibe un parámetro denominado str. En la función se declara una variable (objeto) denominado xmlhttp. También se declara una variable denominada contenidosRecibidos como array, y otra variable para contener un nodo del DOM que obtenemos rescatándolo a través de su id (listaCiudades). Igualmente se declara una variable denominada contenidosAMostrar y se inicializa como cadena vacía.

La línea con el condicional `if (str.length==0)` sirve para detectar si por algún motivo como pudiera ser un error se invoca la función `mostrarSugerencia` pero el parámetro pasado está vacío. En este caso los nodos donde debe mostrarse la información recuperada se establecen con contenido vacío y se aborta la ejecución de la función ya que al llegar a la sentencia `return`; se termina la ejecución sin devolverse ni ejecutarse nada más.

En la línea `xmlhttp=new XMLHttpRequest();` se instancia un objeto predefinido: el objeto `XMLHttpRequest`. Podemos considerarlo un objeto predefinido JavaScript (aunque esto no es formalmente cierto, de momento podemos considerarlo así).

Este objeto al igual que otros objetos predefinidos tiene constructor para su inicialización, propiedades y métodos.

El nombre de este objeto resulta un tanto confuso porque incluye dentro de él las siglas XML cuando realmente no es un objeto relacionado estrictamente con XML en la actualidad. El nombre de este objeto se debe a causas históricas: en sus orígenes se concibió con ciertas expectativas que después se han ido ampliando, por ello no prestaremos demasiada atención al nombre en sí, sino a las posibilidades que nos brinda.

Crear un objeto `XMLHttpRequest` equivale a “abrir un canal de comunicación con el servidor”. Normalmente solo crearemos un objeto de este tipo de forma simultánea (por ejemplo dentro de una función de respuesta). En teoría es posible crear varios objetos de este tipo dentro de una función, pero los navegadores o servidores limitan el número de conexiones aceptadas para evitar sobrecargas. Ten en cuenta que si no se controla el número de objetos de este tipo con acción simultánea la web se podría bloquear por sobrecarga.

OBJETO XMLHTTPREQUEST: PROPIEDADES

El objeto predefinido `XMLHttpRequest` es el objeto clave de Ajax, ya que nos proporciona la interfaz o herramientas necesarias para hacer las operaciones y comprobaciones necesarias para la comunicación en segundo plano con el servidor.

Propiedad	Descripción	Observaciones / Ejemplo
<code>readyState</code>	Contiene un valor numérico entero que representa la situación del intercambio de datos a través del objeto.	<p><code>if (xmlhttp.readyState==4)</code> indica que se ha recibido la información solicitada del servidor. Puede valer:</p> <ul style="list-style-type: none"> 0: no inicializado. Indica que no se ha abierto la conexión con el servidor (no se ha llamado a <code>open</code>) 1: conexión con servidor establecida. Indica que se ha abierto la conexión pero todavía no se ha enviado la petición (no se ha llamado a <code>send</code>). 2: recibida petición en servidor. Indica que el servidor ya ha recibido la petición (se ha llamado a <code>send</code>) 3: enviando información. Se está enviando la información por parte del servidor, todavía no se ha completado la recepción. 4: completado. Se ha recibido la información del servidor y está lista para operar con ella.

Propiedad	Descripción	Observaciones / Ejemplo
onreadystatechange	Sirve para definir una función que será llamada automáticamente cada vez que cambie la propiedad readyState del objeto.	Puede definirse directamente como función anónima: <code>xmlhttp.onreadystatechange = function() { ...}</code> O indicando el nombre de función a ejecutar: <code>xmlhttp.onreadystatechange = ejecutarRespuesta;</code>
responseText	Una vez completada la comunicación, responseText contiene la respuesta del servidor en forma de cadena de texto	<code>contenidosRecibidos = xmlhttp.responseText.split(",");</code> En este ejemplo la respuesta en forma de texto separado por comas se traslada a los elementos de un array.
responseXML	Una vez completada la comunicación, responseText contiene la respuesta del servidor en formato XML	Si el servidor no ha respondido en formato XML al tratar de recuperar responseXML obtendremos null. La respuesta es un documento o nodo con estructura DOM. Veremos ejemplos más adelante.
status	Código numérico entero enviado por el servidor que indica el tipo de respuesta dada a la petición. Puede tomar valores como: 200: respuesta correcta. 404: no encontrado. 500: error interno del servidor.	<code>if (xmlhttp.readyState==4 && xmlhttp.status==200)</code> El código anterior es un condicional asociado a si se ha completado el intercambio de información y si el servidor ha devuelto un código de respuesta correcta.
statusText	Equivalente a status pero en forma de cadena de texto "OK": respuesta correcta. "Not found": no encontrado.	<code>alert ('Respuesta server: ' + xmlhttp.statusText);</code>
multipart	Booleano con valor predeterminado false. En caso de valer true indica que se espera una respuesta formada por múltiples documentos XML.	De momento no lo estudiaremos
upload	Permite añadir event listeners de respuesta a los eventos de subida de información al servidor.	De momento no lo estudiaremos.
withCredentials	Booleano con valor predeterminado false. Indica cuando una petición de información a un servidor externo debe llevar información asociada como cookies o encabezados específicos.	No es de aplicación cuando la petición se realiza dentro del mismo servidor que aloja la página web. De momento no lo estudiaremos.
Otras propiedades	Por ejemplo channel	De momento no las estudiaremos.

RESUMEN

El objeto XMLHttpRequest es un objeto predefinido que nos provee de propiedades y métodos para manejar el intercambio de información en segundo plano con el servidor.

Con lo que hemos visto hasta ahora ya somos capaces de comprender algunas líneas del ejemplo con el que estamos trabajando. En concreto debemos ser capaces de comprender este código:

```
xmlhttp=new XMLHttpRequest();
xmlhttp.onreadystatechange = function() {
  if (xmlhttp.readyState==4 && xmlhttp.status==200) {
    contenidosRecibidos = xmlhttp.responseText.split(",");
  }
}
```

En las próximas entregas completaremos la información necesaria para entender el objeto XMLHttpRequest y la operación de Ajax.

EJERCICIO

Tomando como punto de partida el código de ejemplo Ajax con el que estamos trabajando:

a) Modifica el código para que cuando se produzca una transmisión de información se muestre en una tabla el valor de todas las propiedades del objeto XMLHttpRequest.

Para comprobar si tus respuestas son correctas puedes consultar en los foros aprenderaprogramar.com.

Próxima entrega: CU01208F

Acceso al curso completo en aprenderaprogramar.com -- > Cursos, o en la dirección siguiente:

http://aprenderaprogramar.com/index.php?option=com_content&view=category&id=83&Itemid=212