



APRENDERAPROGRAMAR.COM

RESPONSEXML: MÉTODO
DE XMLHTTPREQUEST.
RECUPERAR DATOS DESDE
ARCHIVO XML CON AJAX.
EJEMPLO (CU01210F)

Sección: Cursos

Categoría: Tutorial básico del programador web: Ajax desde cero

Fecha revisión: 2031

Resumen: Entrega nº10 del Tutorial básico “Ajax desde cero”.

Autor: Alex Rodríguez

FORMATOS PARA INTERCAMBIOS DE DATOS: XML

La tecnología web se sustenta en el intercambio continuo de datos entre usuario y navegador, entre navegador y sistema operativo, entre navegador y servidor, etc. Es de especial relevancia el intercambio de datos del navegador con el servidor pues es lo que nos permite “traer la información desde internet a nuestro computador”.



Los datos en el servidor se pueden mantener de diversas maneras. Podrían mantenerse en ficheros o también en base de datos. Un ejemplo para mantener datos sería simplemente almacenarlos en un fichero txt alojado en el servidor. Un ejemplo de contenido podría ser este:

<p>País: España Capital: Madrid Texto capital: Madrid es un municipio y una ciudad de España. La localidad, con categoría histórica de villa, es la capital del país y de la Comunidad de Madrid. También conocida como la Villa y Corte, es la ciudad más poblada de España, con 3.165.235 habitantes. Como capital del Estado, Madrid alberga las sedes del gobierno, las Cortes Generales, ministerios, instituciones y organismos asociados, así como la residencia oficial de los reyes de España y del presidente del Gobierno. En el plano económico, es la cuarta ciudad más rica de Europa, tras Londres, París y Moscú. Ciudades más importantes: Madrid, Barcelona, Valencia, Sevilla, Zaragoza, Málaga, Murcia.</p> <p>País: México Capital: México D.F. Texto capital: La Ciudad de México, Distrito Federal o, en su forma abreviada, México, D.F., es la capital y sede de los poderes federales de los Estados Unidos Mexicanos. Se trata de una de las 32 entidades federativas que forman parte de México; como tal, no es parte integral de ninguna de las otras 31 (denominados estados), pero a la vez pertenece a toda la unión. Es el núcleo urbano más grande del país, y también el principal centro político, académico, económico, de moda, financiero, empresarial y cultural. Catalogada como ciudad global, es uno de los centros financieros y culturales más importantes de América y del mundo. Su población es de alrededor de 8,8 millones de habitantes. Ciudades más importantes: México D.F., Ecatepec, Guadalajara, Puebla, Juárez, Tijuana, León, Zapopan.</p> <p>País: Argentina Capital: Buenos Aires Texto Capital: Buenos Aires, formalmente Ciudad Autónoma de Buenos Aires (CABA) —también llamada Capital Federal por ser sede del gobierno federal, es una de las veinticuatro entidades federales y capital de la República Argentina. Está situada en la región centro-este del país. La población de la ciudad se estima en 2.890.151 habitantes, y la de su aglomeración urbana, el Aglomerado Gran Buenos Aires, en 14.391.538 habitantes, siendo la mayor área urbana del país, la segunda de Sudamérica, Hispanoamérica y del hemisferio sur, y una de las 20 mayores ciudades del mundo. Ciudades más importantes: Buenos Aires, Córdoba, Rosario, La Plata, Mar del Plata, San Miguel de Tucumán, Ciudad de Salta.</p> <p>País: Colombia Capital: Bogotá Texto Capital: Bogotá, oficialmente Bogotá, Distrito Capital, abreviado Bogotá, D. C. es la capital de la República de Colombia y del departamento de Cundinamarca. Está administrada como Distrito Capital, y goza de autonomía de gestión. Está ubicada en el centro de Colombia, en la región natural conocida como la sabana de Bogotá. Según las últimas estimaciones, Bogotá tiene una población de 7.878.783 habitantes. Tiene una longitud de 33 km de sur a norte, y 16 km de oriente a occidente. Como capital, alberga los organismos de mayor jerarquía de la rama ejecutiva, legislativa y judicial. Ciudades más importantes: Bogotá, Medellín, Cali, Barranquilla, Cartagena, Cúcuta, Soledad, Ibagué.</p>

Mantener los datos como hemos mostrado (“en bruto”) es posible, y podemos crear aplicaciones que funcionen utilizando estos datos, pero no utiliza estándares, y esto dificulta la programación.

Alternativamente los datos pueden encapsularse en estructuras de datos bajo un lenguaje de programación, por ejemplo:

```
$pais[0]="spain";
$pais[1]="mexico";
$pais[2]="argentina";
$pais[3]="colombia";

$capital[0]="Madrid";
$textoCapital[0]=" Madrid es un municipio y una ciudad de España. La localidad, con categoría histórica de...";
$ciudad[0][0]="Madrid"; $ciudad[0][1]="Barcelona"; $ciudad[0][2]="Valencia"; $ciudad[0][3]="Sevilla";
$ciudad[0][4]="Zaragoza"; $ciudad[0][5]="Málaga"; $ciudad[0][6]="Murcia";

$capital[1]="México";
$textoCapital[1]="La Ciudad de México, Distrito Federal o, en su forma abreviada, México, D.F., es la...";
$ciudad[1][0]="México D.F."; $ciudad[1][1]="Ecatepec"; $ciudad[1][2]="Guadalajara"; $ciudad[1][3]="Puebla";
$ciudad[1][4]="Juárez"; $ciudad[1][5]="Tijuana"; $ciudad[1][6]="León"; $ciudad[1][7]="Zapopan";

$capital[2]="Argentina";
$textoCapital[2]=" Buenos Aires, formalmente Ciudad Autónoma de Buenos Aires (CABA) —también llamada...";
$ciudad[2][0]="Buenos Aires"; $ciudad[2][1]="Córdoba"; $ciudad[2][2]="Rosario"; $ciudad[2][3]="La Plata";
$ciudad[2][4]="Mar del Plata"; $ciudad[2][5]="San Miguel de Tucumán"; $ciudad[2][6]="Ciudad de Salta";

$capital[3]="Colombia";
$textoCapital[3]=" Bogotá, oficialmente Bogotá, Distrito Capital, abreviado Bogotá, D. C. es la capital de...";
$ciudad[3][0]="Bogotá"; $ciudad[3][1]="Medellín"; $ciudad[3][2]="Cali"; $ciudad[3][3]="Barranquilla";
$ciudad[3][4]="Cartagena"; $ciudad[3][5]="Cúcuta"; $ciudad[3][6]="Soledad"; $ciudad[3][7]="Ibagué";
```

Esto es lo que hicimos en un ejemplo de código anterior. No obstante, esto sigue planteando problemas y no es realmente una estandarización de datos. Este formato es dependiente del lenguaje, en este caso PHP, y una adecuada estandarización no debe ser dependiente del lenguaje. Con la estandarización conseguimos independencia de los datos: que puedan ser usados por cualquier tipo de aplicación en cualquier tipo de lenguaje.

Para solucionar esto surgen formas de estandarización universal entre los que podemos citar los formatos de fichero xml y json, así como las bases de datos.

FORMATO DE DATOS XML

XML (Extensible Markup Language) es un lenguaje de etiquetas, es decir, cada paquete de información está delimitado por dos etiquetas como se hace también en el lenguaje HTML, pero XML separa el contenido de la presentación. No vamos a entrar a explicar XML, simplemente vamos a verlo desde el punto de vista práctico, y con los ejemplos que presentaremos será suficiente. Puedes leer más sobre XML en aprenderaprogramar.com.

La información anterior vamos a guardarla en un fichero xml con la siguiente estructura (archivo al que vamos a denominar listadoPaises.xml):

```
<?xml version="1.0" encoding="UTF-8"?>
<listadoPaises>
<pais>
  <nombre>España</nombre>
  <capital>Madrid</capital>
  <textoCapital>Madrid es un municipio y una ciudad de España. La localidad, con categoría...</textoCapital>
  <ciudadImportante>Madrid</ciudadImportante>
  <ciudadImportante>Barcelona</ciudadImportante>
  <ciudadImportante>Valencia</ciudadImportante>
  <ciudadImportante>Sevilla</ciudadImportante>
  <ciudadImportante>Zaragoza</ciudadImportante>
  <ciudadImportante>Málaga</ciudadImportante>
  <ciudadImportante>Murcia</ciudadImportante>
</pais>
<pais>
  <nombre>México</nombre>
  <capital>México D.F.</capital>
  <textoCapital>La Ciudad de México, Distrito Federal o, en su forma abreviada, México, D.F., es... </textoCapital>
  <ciudadImportante>México D.F.</ciudadImportante>
  <ciudadImportante>Ecatepec</ciudadImportante>
  <ciudadImportante>Guadalajara</ciudadImportante>
  <ciudadImportante>Puebla</ciudadImportante>
  <ciudadImportante>Juárez</ciudadImportante>
  <ciudadImportante>Tijuana</ciudadImportante>
  <ciudadImportante>León</ciudadImportante>
  <ciudadImportante>Zapopan</ciudadImportante>
</pais>
<pais>
  <nombre>Argentina</nombre>
  <capital>Buenos Aires</capital>
  <textoCapital>Buenos Aires, formalmente Ciudad Autónoma de Buenos Aires (CABA) —también... </textoCapital>
  <ciudadImportante>Buenos Aires</ciudadImportante>
  <ciudadImportante>Córdoba</ciudadImportante>
  <ciudadImportante>Rosario</ciudadImportante>
  <ciudadImportante>La Plata</ciudadImportante>
  <ciudadImportante>Mar del Plata</ciudadImportante>
  <ciudadImportante>San Miguel de Tucumán</ciudadImportante>
  <ciudadImportante>Ciudad de Salta</ciudadImportante>
</pais>
<pais>
  <nombre>Colombia</nombre>
  <capital>Bogotá</capital>
  <textoCapital>Bogotá, oficialmente Bogotá, Distrito Capital, abreviado Bogotá, D. C. es la... </textoCapital>
  <ciudadImportante>Bogotá</ciudadImportante>
  <ciudadImportante>Medellín</ciudadImportante>
  <ciudadImportante>Cali</ciudadImportante>
  <ciudadImportante>Barranquilla</ciudadImportante>
  <ciudadImportante>Cartagena</ciudadImportante>
  <ciudadImportante>Cúcuta</ciudadImportante>
  <ciudadImportante>Soledad</ciudadImportante>
  <ciudadImportante>Ibagué</ciudadImportante>
</pais>
</listadoPaises>
```

Considera el intercambio de mercancías por barco a nivel mundial. ¿Cómo se realiza? A través de contenedores, que son un estándar (medidas, dimensiones, materiales, etc.). Como todos los productores, transportistas, intermediarios y destinatarios se han puesto de acuerdo para que el intercambio se haga usando contenedores la operación de mercancías resulta más fácil. Nada impide que se envíe una mercancía por barco dentro de una gigantesca olla de barro, pero esto dificultaría la operación ya que no cumple los estándares y los sistemas y máquinas no están preparados para operar con ollas de barro, pero sí con contenedores.



Para el intercambio de datos informáticos se han definido formatos estandarizados que facilitan el almacenamiento y recuperación de la información de una forma predeterminada. Ahora estamos estudiando XML, más adelante veremos el formato JSON (JavaScript Object Notation).

Como puedes observar, el fichero XML consta de una primera línea donde declaramos la versión de XML que vamos a emplear, y de una serie de datos delimitados por etiquetas.

XML es un formato estandarizado para almacenamiento de información. Esto supone que todos los sistemas informáticos están preparados para manejar información en este formato, y al igual que manejar las mercancías en contenedores tiene ventajas para el transporte por barco, usar XML u otras formas normalizadas tiene ventajas en el intercambio de datos en computación.

RECUPERAR UN DOCUMENTO XML CON EL MÉTODO RESPONSEXML

El objeto predefinido XMLHttpRequest nos provee de distintos métodos, uno de los cuales es responseXML. Este método nos permite recuperar un documento con estructura DOM a partir de un archivo xml cuya ruta habremos invocado con el método open.

Ejemplo: invocamos xmlhttp.open("GET", "listadoPaises.xml");

Recuperamos el documento usando este método: var xmlDoc = xmlhttp.responseXML;

A partir de aquí podemos trabajar sobre el documento usando los métodos JavaScript para operar con el DOM. Por ejemplo podremos obtener var listaPaises = xmlDoc.getElementsByTagName('pais'); Suponemos que ya conoces estos métodos ya que se explican en el curso JavaScript de

aprenderaprogramar.com, que indicamos forma parte de los conocimientos previos necesarios para seguir este curso de Ajax.

En el siguiente ejemplo usamos el archivo listadoPaises.xml que vimos anteriormente para recuperar información, de modo que según elija el usuario un país de un combobox recuperaremos dinámicamente las ciudades de ese país desde el fichero xml. Sube el fichero xml al servidor. Escribe el siguiente código y guárdalo como ejemploCU01210F_1.html y súbelo también al servidor y comprueba el resultado al visualizar este fichero html en tu navegador.

```
<!DOCTYPE html><html><head><title>Cursos aprende a programar</title><meta charset="utf-8">

<style type="text/css">
*{font-family:sans-serif;} a:link {text-decoration:none;} select{font-size:18px;}
div div {color: blue; background-color:#F1FEC6; font-size: 20px; float:left; border: solid; margin: 20px; padding:15px;}
</style>

<script>
function mostrarSugerencia(str) {
var paisElegido="";
if (str=='spain') {paisElegido='España';}
else if (str=='mexico') {paisElegido='México';}
else if (str=='argentina') {paisElegido='Argentina';}
else if (str=='colombia') {paisElegido='Colombia';}
else {paisElegido='none';}

var xmlhttp;
if (str.length==0 || paisElegido=='none') { document.getElementById("txtInformacion").innerHTML="no hay datos";
mostrarCiudades(); return; }

xmlhttp=new XMLHttpRequest();
xmlhttp.onreadystatechange = function() {
if (xmlhttp.readyState==4 && xmlhttp.status==200) {
    var xmlDoc = xmlhttp.responseXML;
    paisesRecibidos = obtenerArrayNodosPorTag(xmlDoc, 'pais');

    for (var i=0; i<paisesRecibidos.length;i++) {
        var nombrePais = obtenerArrayNodosPorTag(paisesRecibidos[i], 'nombre')[0].innerHTML;
        if (nombrePais==paisElegido) {
            document.getElementById("txtInformacion").innerHTML = 'El pais recibido por get en segundo plano es
'+nombrePais+ ' y tiene indice '+i;
            var ciudadesPais = obtenerArrayNodosPorTag(paisesRecibidos[i], 'ciudadImportante');
            mostrarCiudades(ciudadesPais);
        }
    }
}
}

xmlhttp.open("GET","listadoPaises.xml");
xmlhttp.send();
}

//Continua
```

```
function obtenerArrayNodosPorTag (nodoRaiz, nombreTag) {
var arrayNodos = nodoRaiz.getElementsByTagName(nombreTag);
return arrayNodos; }

function mostrarCiudades (arrayCiudades) {
var nodoMostrarResultados = document.getElementById('listaCiudades');
if (!arrayCiudades) {nodoMostrarResultados.innerHTML = ""; return}
var contenidosAMostrar = "";
for (var i=0; i<arrayCiudades.length;i++) {
    contenidosAMostrar = contenidosAMostrar+'<div id="ciudades'+i+">';
    contenidosAMostrar += '<a href="http://aprenderaprogramar.com">' + arrayCiudades[i].innerHTML+ '</a></div>';
}
if (contenidosAMostrar) {nodoMostrarResultados.innerHTML = contenidosAMostrar;}
}
</script></head>

<body style="margin:20px;"><h2>Elige un país:</h2>
<form action="">
<select onchange="mostrarSugerencia(this.value)">
<option value="none">Elige</option>
<option value="spain">España</option>
<option value="mexico">México</option>
<option value="argentina">Argentina</option>
<option value="colombia">Colombia</option>
</select>
</form>
<br/>
<p>Informacion sobre operacion en segundo plano con Ajax: <span style="color:brown;" id="txtInformacion"></span></p>
<div id="listaCiudades"> </div>
</body></html>
```

Si has seguido los pasos indicados, debes ser capaz de elegir opciones del combobox desplegable y visualizar resultados por pantalla. Por ejemplo, si eliges la opción “Argentina”, el resultado esperado será que por pantalla se visualice:

Informacion sobre operacion en segundo plano con Ajax: El pais recibido por get en segundo plano es argentina y tiene indice 2

Rosario	La Plata	Mar del Plata	San Miguel de Tucumán	Ciudad de Salta
---------	----------	---------------	-----------------------	-----------------

Elige un país:

Argentina ▾

Informacion sobre operacion en segundo plano con Ajax:
 El pais recibido por get en segundo plano es argentina y tiene indice 2

Rosario	La Plata	Mar del Plata
San Miguel de Tucumán	Ciudad de Salta	

Si cambias la selección del país por otro país, debes observar un refresco “casi instantáneo” de modo que se modifica el texto y las ciudades que se muestran, viéndose en cada caso las correspondientes al país seleccionado.

Cada vez que seleccionamos un país, hay una comunicación con el servidor en segundo plano y la información recibida del servidor la utilizamos para hacer cambios en la página web sin necesidad de hacer una recarga tradicional.

Este resultado es idéntico al que obtuvimos en una entrega anterior del curso. La diferencia está en que anteriormente teníamos los datos sin normalizar en un fichero php y ahora tenemos los datos normalizados en un fichero xml.

COMPRENDIENDO LOS CONCEPTOS

Intenta interpretar el código anterior. En la próxima entrega daremos una explicación sobre el mismo.

EJERCICIO

Considera el siguiente documento xml al que denominamos ejemplo.xml:

```
<?xml version="1.0"?>
<catalog>
  <book id="bk101">
    <author>Gambardella, Matthew</author>
    <title>XML Developer's Guide</title>
    <price>44.95</price>
    <publish_date>2000-10-01</publish_date>
  </book>
  <book id="bk102">
    <author>Ralls, Kim</author>
    <title>Midnight Rain</title>
    <price>5.95</price>
    <publish_date>2000-12-16</publish_date>
  </book>
  <book id="bk103">
    <author>Corets, Eva</author>
    <title>Maev Ascendant</title>
    <price>5.95</price>
    <publish_date>2000-11-17</publish_date>
  </book>
</catalog>
```

- a) Transforma el archivo xml para que id en lugar de ser un atributo de etiqueta sea una etiqueta
- b) Crea un archivo html que permita elegir el título (title) y dinámicamente recupere desde el archivo xml el contenido de todas las demás etiquetas incluido el id y los muestre por pantalla.

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

Próxima entrega: CU01211F

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