

## XAJAX: una librería de AJAX para PHP (parte 1)

AJAX es el acrónimo de Asynchronous Javascript And XML o Javascript y XML asíncronos dicho en nuestro idioma. Se trata de una combinación de tecnologías que se ha popularizado muchísimo desde hace unos años con la irrupción de la Web 2.0 y lo podemos encontrar en infinidad de páginas y servicios web como Gmail, Google Maps, Hotmail, Flickr y un largo etcétera.

La clave de AJAX está en que es asíncrono, es decir, permite que la página del cliente realice peticiones al servidor sin recargar la página. Sólo se actualizan las partes de la página que nos interesan pero sin tener que recargar la página entera, lo que proporciona una sensación de navegación mucho más fluida.

Para introducirnos en el mundo de AJAX vamos a utilizar una librería llamada XAJAX que nos permite aprovechar las capacidades de AJAX en nuestras páginas PHP. Podéis descargar esta librería en la página del proyecto <http://xajaxproject.org/>

A continuación, unos de códigos de ejemplo que se presentarán en diferentes capítulos para que sean más fáciles de leer.

También se puede descargar la versión PDF con todos los ejemplos aquí. Y un Zip con los códigos fuente.

### Ejemplo 1. Introducción a la librería.

Comencemos con un simple 'Hola, mundo' para ir calentando. En esta ocasión lo ponemos todo en el mismo archivo PHP llamado 'holaAjax.php'.

#### 'holaAjax.php'

```
<?php
require ('xajax/xajax.inc.php');
$xajax = new xajax();

function cambia_texto($mensaje){
    $respuesta = new xajaxResponse();
    $respuesta->addAssign("mensaje", "innerHTML", $mensaje);
    return $respuesta;
}

$xajax->registerFunction("cambia_texto");
$xajax->processRequests();
?>
<html>
<head>
<?php
```

```
$xajax->printJavascript("xajax/");
?>
</head>
<body>
<input type="button" onclick="xajax_cambia_texto('Hola Ajax');" value="Pulsa" />
<div id="mensaje"></div>
</body>
</html>
```

### Explicación:

Lo primero que debemos hacer es importar la librería de xajax y crear una instancia de xajax:

```
require ('xajax/xajax.inc.php');
xajax = new xajax();
```

A continuación definimos las funciones que se van a utilizar. En esta ocasión sólo tenemos la función 'cambia\_texto'. Dentro de esta función, creamos una instancia de respuesta y en esa respuesta asignamos un valor a una capa llamada 'mensaje' que se encuentra en el body de la página mediante el método addAssign().

Una vez definidas las funciones, las registramos en xajax y le indicamos a xajax que gestione las peticiones:

```
$xajax->registerFunction("cambia_texto");
$xajax->processRequests();
```

Es importante poner dentro de las etiquetas head de la página la instrucción que genera el código Javascript necesario para ejecutar nuestras funciones.

```
$xajax->printJavascript("xajax/");
```

Una vez hecho esto podemos asignar estas funciones a los elementos de nuestra página, normalmente a los componentes de formulario pero añadiendo el prefijo 'xajax\_' antes del nombre de nuestra función.

Este ejemplo no hace nada especial que no podamos hacer con Javascript, así que pasaremos a algo más complejo incluyendo llamadas a una base de datos.

Descargar archivo PHP.

## **XAJAX: una librería de AJAX para PHP (parte 2)**

## Ejemplo 2. Un formulario dinámico con un número indeterminado de campos.

En este ejemplo presentamos un formulario en el que no se sabe a priori el número de campos de entrada que se van a necesitar por lo que se generan dinámicamente añadiéndolos según los pide el usuario sin recargar el formulario entero.

El ejemplo simula un formulario de matrícula de una academia de idiomas en la que un alumno se puede matricular en varios idiomas y diferentes cursos para cada idioma. Como no se conoce el número de cursos en los que se va a matricular, cada vez que pulsamos el botón 'Añadir idioma' nos aparecen dos campos nuevos para seleccionar idioma y curso sin perder la información ya introducida.

Aquí mostramos cómo modificar el contenido de una página sin recargarla a la vez que mostramos cómo acceder a la información de un formulario usando Ajax.

### 'matriculaidiomas.php'

```
<?php
require('xajax/xajax.inc.php');
$xajax = new xajax();
$xajax->setCharEncoding('latin1');

function addAsignaturas($formDatos){
    $respuesta = new xajaxResponse();
    //contador de idiomas matriculados
    $countIdioma = $formDatos['countIdioma'];
    $idiomas = array("Ingles", "Italiano", "Frances", "Chino", "Ruso");
    $cursos = array("1","2","3","4","5");
    $formIdiomas = "";

    //procesamos las anteriores asignaturas y mostramos los valores seleccionados
    for($i=1; $i<$countIdioma; $i++){
        $formIdiomas .= "<div>";
        $formIdiomas .= "Idioma $i: <select name='idioma_{$i}' id='idioma_{$i}'>";
        for ($j=0; $j<count($idiomas); $j++){
            $formIdiomas .= "<option value='".$idiomas[$j]."'";
            if (strcasecmp($idiomas[$j], $formDatos["idioma_{$i}"]) == 0){
                $formIdiomas .= " selected='selected' ";
            }
            $formIdiomas .= ">".$idiomas[$j]."</option>";
        }
        $formIdiomas .= "</select>";

        $formIdiomas .= " Curso: <select name='curso_{$i}' id='curso_{$i}'>";
        for ($j=0; $j<count($cursos); $j++){
            $formIdiomas .= "<option value='".$cursos[$j]."'";
            if (strcasecmp($cursos[$j], $formDatos["curso_{$i}"]) == 0){
                $formIdiomas .= " selected='selected' ";
            }
            $formIdiomas .= ">".$cursos[$j]."</option>";
        }
        $formIdiomas .= "</select>";
        $formIdiomas .= "</div>";
    }

    //añadimos uno nuevo en blanco
    $formIdiomas .= "
    <div>
    Idioma $countIdioma:
    <select name='idioma_{$countIdioma}' id='idioma_{$countIdioma}'>";
    for ($i=0; $i<count($idiomas); $i++){
        $formIdiomas .= "<option value='".$idiomas[$i]."'>".$idiomas[$i]."</option>";
    }
    $formIdiomas .= "</select>";
}
```

```

Curso :
<select name='curso_$$countIdioma' id='curso_$$countIdioma'>;
for ($i=0; $i<count($cursos); $i++){
    $formIdiomas .= "<option value='".$cursos[$i]."'>".$cursos[$i]."</option>";
}
$formIdiomas .= "</div>";
//incrementamos el contador de asignaturas en uno
$respuesta->addAssign("countIdioma", "value", ($countIdioma+1));
//añadimos los campos de formulario para introducir un idioma
$respuesta->addAssign("divIdiomas", "innerHTML", $formIdiomas);

return $respuesta;
}
}
$ajax->registerFunction('addAsignaturas');
$ajax->processRequests();
?>
<html>
<head>
<?php
$ajax->printJavascript('ajax/');
?>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
</head>
<body>
<h2>Registro de alumnos</h2>
<form name="formSocios" id="formSocios" method="post" action="#">
<div>
<table>
<tr><td>Nombre: </td><td><input type="text" name="textNombre" id="textNombre"
value="Pepe" /></td></tr>
<tr><td>Edad: </td><td><input type="text" name="textEdad" id="textEdad" value="20"
/></td></tr>
<tr><td>Sexo: </td><td><input type="radio" name="radioSexo" id="radioSexo"
value="hombre" checked="checked" /> Hombre
<input type="radio" name="radioSexo" id="radioSexo" value="mujer" checked="checked" />
Mujer </td></tr>
<tr><td>Ciudad: </td><td><select name="selectCiudad" id="selectCiudad">
<option value="Alicante">Alicante</option>
<option value="Castellón">Castellón</option>
<option value="Valencia">Valencia</option>
</select>
</td></tr>
<tr><td colspan="2">
<h3>Idiomas en los que se matricula</h3>
<div id="divIdiomas"><div><br /><br />
</div></tr>
<tr><td colspan="2">
<input type="button" name="buttonIdioma" id="buttonIdioma" value="Añadir idioma"
onclick="xajax_addAsignaturas(xajax.getFormValues('formSocios'))" />
<input type="hidden" name="countIdioma" id="countIdioma" value="1" />
</td></tr>
</table>
</div>
</form>
</body>
</html>

```

Descargar archivo php

**Explicación:**

En el formulario HTML vemos que el botón para añadir idiomas incluye un código xajax en el evento onclick que pasa todas las variables del formulario a la función addAsignaturas:

```

<input type="button" name="buttonIdioma" id="buttonIdioma" value="Añadir idioma"
onclick="xajax_addAsignaturas(xajax.getFormValues('formSocios'))" />

```

Para leer las variables del formulario simplemente debemos tratarlas como un array cualquiera:

```
$formDatos['textNombre']
```

Dentro de la función addAsignaturas hay dos partes principales: una que regenera las asignaturas ya seleccionadas y una segunda parte que presenta los campos de formulario necesarios para introducir una nueva asignatura. Se ha hecho mediante inputs tipo select para que sea más cómodo para el usuario, aunque a su vez se hace más complicado para el programador ya que hay que hacer muchos bucles for y comparaciones con cada valor para marcar la opción seleccionada.

## XAJAX: una librería de AJAX para PHP (parte 3)

### Ejemplo 3. Un formulario con sugerencias para autocompletado

En este ejemplo se muestra un ejemplo de autocompletado de campos de formulario utilizando Xajax.

En el formulario se le pide al usuario que introduzca el nombre de un país en un campo de texto pero para evitar errores le ofrecemos una lista de sugerencias a medida que va escribiendo. Los nombres de esta lista se leen de una base de datos de países con lo que incluimos acceso a base de datos en este ejemplo para hacerlo más realista. Si hace clic en una de las sugerencias, se le asigna el valor de la sugerencia seleccionada al campo de texto. Cuanto menos escriba el usuario, menos errores cometerá.

Además, también aplicamos validación del campo. Sólo daremos por bueno el nombre si coincide con alguno de la base de datos aunque la comparación la hacemos sólo en minúsculas.

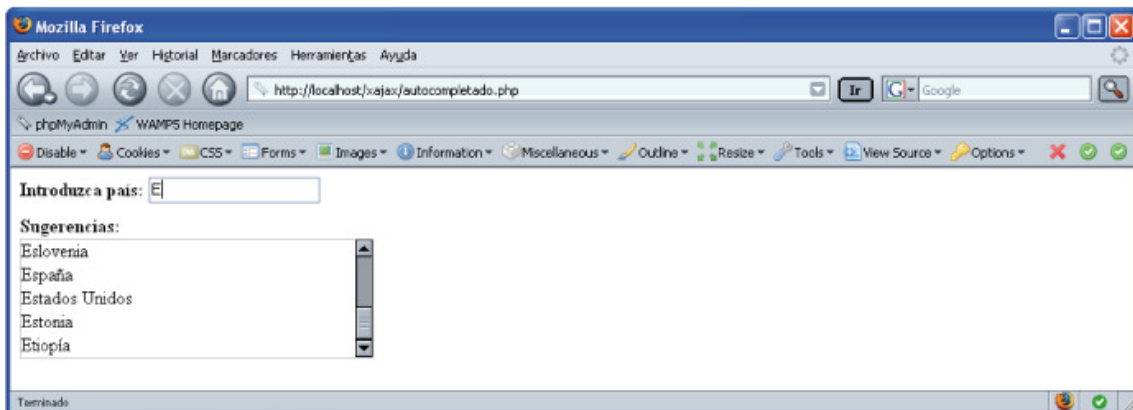


Figura 1. Captura de pantalla de la aplicación.

## 'autocompletado.php':

```
<?php
require ('xajax/xajax.inc.php');
$xajax = new xajax();
$xajax->setCharEncoding('utf-8');

function autocompleta($input){
    $respuesta = new xajaxResponse();
    $output = "";
    $db = mysql_connect("localhost", "root", "");
    mysql_select_db("test");
    mysql_query("SET NAMES 'utf8'");
    //aquellos paises que empiecen como el valor del input
    $query = "SELECT nombre FROM pais WHERE nombre LIKE '".$input."%";
    $res = mysql_query($query);

    //si no hay texto, se deja la capa en blanco y salimos
    if ($input == ""){
        $respuesta->addAssign("divSugerencias", "innerHTML", "");
        return $respuesta;
    }

    //numero de resultados coincidentes
    $num = mysql_num_rows($res);

    if ($num == 0){ //si no hay coincidencias se avisa
        $output = "<font color='red'>Alerta: No existe ningun pais con ese
nombre</font>";
    }
    else if ($num == 1){ //si solo hay una coincidencia
        $row = mysql_fetch_row($res);
        if (strcasecmp($input, $row[0]) == 0){ //si los nombres coinciden, exito
            $output = "<font color='green'><b>OK</b></font>";
        }
        else{ //muestra el select con la unica opcion
            $output = "
            <div><b>Sugerencias:</b></div>
            <div id='divLista'>
            <table name='tablePais' id='tablePais'>
                <tr>
                    <td
onClick=\"xajax_seleccionaPais('".$row[0]."');xajax_autocompleta('".$row[0]."')\">".$row
[0]."</td>
                </tr>
            </div>
            </table>>";
        }
    }
    else{ //si hay varias coincidencias se envia un listado
        $output .= "<div><b>Sugerencias:</b></div>
        <div id='divLista'>
        <table name='tablePais' id='tablePais' cellpadding='0' cellspacing='0'>;
        while ($row = mysql_fetch_row($res)){
            $output .= "<tr><td
onClick=\"xajax_seleccionaPais('".$row[0]."');xajax_autocompleta('".$row[0]."')\">".$row
```

```

[0]."</td></tr>";
    }
    $output .= "</div></table>";
}

//asignamos es contenido de la carpeta sugerencias
$respuesta->addAssign("divSugerencias", "innerHTML", $output);
return $respuesta;
}

function seleccionaPais($pais){
    $respuesta = new xajaxResponse();
    $respuesta->addAssign("textPais", "value", $pais);
    return $respuesta;
}

$xajax->registerFunction('autocompleta');
$xajax->registerFunction('seleccionaPais');
$xajax->processRequests();

?>
<html>

<head>
<?php
$xajax->printJavascript('xajax/');
?>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<style type="text/css">
#divLista{
    width:300px;height:100px;overflow:auto;border:solid 1px #ccc;
}
</style>
</head>

<body>
    <form action="#" name="formPais" id="formPais">
        <b>Introduzca pa&iacute;s:</b> <input type="text" name="textPais" id="textPais"
onkeyup="xajax_autocompleta(this.value)" />
        <div id="divSugerencias" style="margin-top:10px;"></div>
    </form>
</body>
</html>

```

Descargar archivo PHP.

Descargar archivo SQL con la base de datos de pa&iacute;ses.

### Explicaci&acirc;n:

En este caso tenemos dos funciones diferentes:

- autocompleta: con cada tecla que se pulsa en la caja de texto del formulario se consulta a la base de datos pasando el valor de esta caja de texto a la funci&acirc;n. Se busca en la base de datos

todos aquellos países cuyo nombre comience por el texto insertado en la caja de texto. Dentro de la función se pueden dar varios casos:

- Que el valor sea nulo: se borra la capa de sugerencias.
  - Que no haya ninguna coincidencia: se avisa en la capa de sugerencias que no hay ningún país con ese nombre.
  - Que sólo haya una coincidencia: si el nombre de la sugerencia es igual al valor de la caja de texto se muestra el mensaje OK.
  - Que hayan varias coincidencias: se muestra una tabla que lista los países con nombre parecido. Si se hace clic en una celda se llama a la función `seleccionaPais` y se actualiza el contenido de la capa de sugerencias.
- `seleccionaPais`: asigna a la caja de texto el valor seleccionado en la tabla de sugerencias.

Ambas funciones se registran mediante y se ordena que `xajax` procese las peticiones:

```
$xajax->registerFunction('autocompleta');  
$xajax->registerFunction('seleccionaPais');  
$xajax->processRequests();
```

En este caso, la base de datos que se ha utilizado es MySQL y la librería de funciones para mysql que viene con las distribuciones de PHP.

También hemos definido el juego de caracteres que se utilizará en la aplicación:

```
$xajax->setCharEncoding('utf-8');
```

En la base de datos:

```
mysql_query("SET NAMES 'utf8'");
```

Y también en el head de la página:

```
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
```

Si no nos aseguramos de que se corresponden perfectamente los juegos de caracteres puede ser que nombres de países como España o Etiopía nunca coincidan por llevar acentos o eñes.

He querido hacer un ejemplo muy sencillo para que sea fácilmente entendible. Ya sé que hay muchas páginas donde ofrecen sistemas de autocompletado mucho más elaborados, pero el código sería mucho mayor. No es tan vistoso como lo que se suele utilizar pero no lo he hecho con el fin de usarlo directamente en una página web sino con motivos instructivos para iniciarnos en el mundo de Ajax.