

JSON + JQUERY + AJAX + PHP

Fuente: <http://www.bloogie.es>

Contenido

JSON + JQUERY + AJAX + PHP.....	1
Introducción.....	1
El archivo php y json.....	1
El código ajax con jquery.....	2
1. El HTML.....	2
2. El código ajax con jQuery.....	3
3. El archivo casas.php.....	3

Introducción

En un archivo php realizaremos las acciones a llevar a cabo en el servidor como consultas a la base de datos y devolveremos los resultados utilizando json como lenguaje de intercambio de datos.

Json son las siglas de **JavaScript Object Notation** y es un formato muy ligero de intercambio de datos de uso muy simple, motivo por el que se ha hecho muy popular su uso en aplicaciones ajax. Como formato de intercambio de datos tiene sus ventajas e inconvenientes respecto al uso de XML pero en eso no nos vamos a meter en este artículo. Para ofrecer los resultados en formato json utilizando php es necesario instalar la extensión php json, desde la versión php 5.2.0 json se incluye por defecto en cualquier instalación de php.

El archivo php y json

El archivo php que usaremos para procesar la información en el servidor devolverá los datos en formato json. En el código php de este archivo tomamos los valores de los parámetros de la url para realizar la acción necesaria por GET y devolvemos los datos en formato json usando **json_encode**.

```
<?php
$dato = $_GET['parametro'];
$dato2 = $_GET['parametro2'];
if($dato == "valor") {
    process($dato);
}
function process($dato) {
    $resultado = hacer algo con el dato;
    if($resultado_satisfactorio) {
        $jsondata['mensaje'] = "mensaje de exito";
        $jsondata['resultado'] = "resultado";
    } else {
        $jsondata['mensaje'] = "mensaje de error";
    } echo json_encode($jsondata);
}
?>
```

El código ajax con jquery

El código javascript para realizar una llamada ajax con jquery es realmente sencillo. En el ejemplo que vemos a continuación se especifica que se realizará la solicitud ajax al hacer clic en el elemento con id="id" y sólo se activará una vez que el documento está listo en el lado del cliente. Al hacer clic en dicho elemento se pone en marcha la solicitud al servidor a través del archivo php especificado en el parametro "url"; las variables según las cuales realizaremos una acción u otra en el archivo php que nos devuelve la información desde el servidor se especifican en el parámetro "data". Si hay éxito en la solicitud se llama a la función function(data) especificada en "success". Nota que establecemos el tipo de datos como json y el tipo de solicitud en GET. (código probado con jquery versión 1.4.2).

```
$(document).ready(function(){
$("#id").click(function(){
$.ajax({
  data: "parametro1=valor1&parametro2=valor2",
  type: "GET",
  dataType: "json",
  url: "url_archivo_php",
  success: function(data){ alguna_funcion_a_realizar_segun_resultados; }
});
});
});
```

En nuestro ejemplo pusimos dos datos en formato json, mensaje y resultados, que procesaremos con jquery. Para obtener el valor de estos datos por separado usamos las variables data.mensaje y data.resultado en el código jquery.

Sencillo, ¿verdad?

Respecto a la interacción con bases de datos, pongamos un ejemplo.

Supongamos que tiene una tabla en tu base de datos llamada casas en la que almacenas información sobre viviendas, cada una de ellas identificada con un número llamado id. Los campos de esta tabla son:

- id: número identificador de la casa
- m2: metros cuadrados de la vivienda
- hab: número de habitaciones
- direccion: dirección donde está ubicada la casa

Siguamos haciendo suposiciones y supongamos que quieres obtener, via ajax, los datos de una vivienda. Utilizaremos para ello el id identificador de cada casa.

1. El HTML

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

```

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="es-es" lang="es-es" >
  <head>
    <script type="text/javascript" src="/js/jquery.js"></script>
    <script type="text/javascript" src="/js/ajax.js"></script>
  </head>
  <body>
    <a href="#" id="myid">Detalles casa 2</a>
    <div class="info"></div>
  </body>
</html>

```

2. El código ajax con jQuery

Un ejemplo sería el siguiente que hemos incluido el archivo ajax.js llamado en el código html puesto unas líneas más arriba.

```

function results(data) {
  $("#div.info").show();
  $("#div.info").append("ID: "+data.id);
  $("#div.info").append("M2: "+data.m2+"");
  $("#div.info").append("Habitaciones: "+data.hab);
  $("#div.info").append("Lacalización: "+data.direccion);
}
$(document).ready(function(){
  $("#myid").click(function(){
    $.ajax({
      data: "id=2",
      type: "GET",
      dataType: "json",
      url: "http://www.miweb.com/casas.php",
      success: function(data){
        results(data);
      }
    });
  });
});

```

La llamada ajax se activa al hacer clic en el elemento de id="myid". El archivo de procesamiento en el servidor es llamado casas.php y el parámetro pasado es el id de la casa, en el ejemplo 2.

En caso de éxito, se mostrarán los datos en un div de clase "info" que inicialmente podemos tener escondido (p.e. style="display: none").

Nota que los datos json obtenidos son id, m2, hab y direccion, accesibles de forma individual a través de data.id, data.m2, data.hab y data.direccion.

3. El archivo casas.php

El archivo casas.php recibirá los datos pasados por la llamada ajax desde jQuery, este caso la id=2. Realizará la consulta a la tabla "casas" y devolverá la información relativa a la vivienda seleccionada en formato de datos json. Un ejemplo podría ser el siguiente:

```

<?php
(int) $id = $_GET['id'];
if($id > 0) {
    process($id);
}

function process($id) {
    $db = mysql_connect('localhost', 'mysql_user', 'mysql_password');
    if (!$db) {
        die('Could not connect: ' . mysql_error());
    }
    if (!mysql_select_db('nombre_base_de_datos')) {
        die('Could not select database: ' . mysql_error());
    }
    $result = mysql_query("SELECT id,m2,hab,direccion FROM casas WHERE id=".$id);
    if (!$result) {
        die('Could not query: ' . mysql_error());
    }

    $row = mysql_fetch_object($result);
    $jsondata['id'] = $row->id;
    $jsondata['m2'] = $row->m2;
    $jsondata['hab'] = $row->hab;
    $jsondata['direccion'] = $row->direccion;

    echo json_encode($jsondata);
    mysql_close($link);
}
?>

```