

# DESARROLLO WEB CLIENTE WEB (DWC)



CURSO 2017-2018

11 / 03 / 2018

---

## PRÁCTICA TEMA 8

# COMUNICACIÓN

## ASÍNCRONA JQUERY

Por:  
José Luis Ferrete Olarte

**Licencia Creative Commons BY-NC-SA**

Los contenidos de este documento se publican bajo licencia [CC BY-NC-SA](https://creativecommons.org/licenses/by-nc-sa/2.5/es/legalcode.es): Creative Commons Reconocimiento – NoComercial – CompartirIgual.

**Cedemos los siguientes derechos sobre la obra:**

Derecho de reproducción, distribución y comunicación pública sobre la obra.

Derecho a incorporarla en una o más obras conjuntas o bases de datos y para su reproducción en tanto que incorporada a dichas obras conjuntas o bases de datos.

Derecho para efectuar cualquier transformación sobre la obra y crear y reproducir obras derivadas.

Derecho de distribución y comunicación pública de copias o grabaciones de la obra, como incorporada a obras conjuntas o bases de datos.

Derecho de distribución y comunicación pública de copias o grabaciones de la obra, por medio de una obra derivada.

**Siempre que lo hagas con estas condiciones:**

Reconocer la autoría, especificando la firma y el autor que lo publica (*"D. José Luis Ferrete Olarte"*). Si es en formato digital, debes añadir un enlace al contenido original.

Compartir bajo la misma licencia. Si reproduces o remezclas esta obra, sólo puedes distribuir la obra generada bajo una licencia como ésta (con las mismas condiciones). Hemos realizado este trabajo con fines educativos y queremos que, si nos utilizas como base para tus creaciones, también contribuyas a la comunidad difundiéndolas con licencias libres. Así, aprenderemos todos juntos.

No se permite un uso comercial de la obra original, ni de las posibles obras derivadas.

Ante cualquier duda sobre las condiciones de cesión de derechos, consultar: <https://creativecommons.org/licenses/by-nc-sa/2.5/es/legalcode.es>

## INDICE

I.	Introducción .....	3
II.	Tareas a realizar .....	4
A.	Petición AJAX y procesamiento de respuesta XML .....	4
B.	Petición AJAX y procesamiento de respuesta HTML .....	5
C.	Petición AJAX y procesamiento de respuesta JSON .....	6
III.	Bibliografía .....	8

## I. Introducción

Para la correcta consecución de esta práctica deberemos acometer tres tareas que pasamos a detallar a continuación:

En la **primera** de ellas debemos implementar una aplicación que lea un fichero de datos y lo procese usando la biblioteca jQuery para añadir una tabla con esta información en una página web. No tenemos que preocuparnos por el estilo porque vendrá dado por la página dónde se incruste. Podemos añadir la tabla directamente al cuerpo del documento HTML.

Como especificaciones adicionales, se nos indica que:

- El fichero con los datos de los alumnos está disponible en la plataforma con el nombre de `alumnos.xml`.
- Si cargamos los datos mediante el método GET, debemos ejecutar el archivo a través de un servidor.

Para la **segunda** de las tareas, debemos crear una función que permita cargar un fragmento de código HTML utilizando la biblioteca jQuery, que será generado mediante AJAX utilizando una plantilla, y que lo insertemos en el cuerpo de la página. Además, el archivo HTML con los datos de los alumnos está disponible en la plataforma con el nombre de `alumnos.html`.

Por último, para la **tercera**, crearemos una función que permita cargar un fichero de datos y procesarlo para agregar una tabla con esa información a una página web utilizando la biblioteca jQuery.

No tenemos que preocuparnos por el estilo porque vendrá dado por la página dónde se incruste. Podemos añadir la tabla directamente al cuerpo del documento HTML. El archivo con los datos de los alumnos está disponible en la plataforma con el nombre de `alumnos.json`.

## II. Tareas a realizar

### A. Petición AJAX y procesamiento de respuesta XML

```
<!doctype html>
<html lang="ES">
<head>
  <meta charset="UTF-8">
  <title></title>
  <script src="https://code.jquery.com/jquery-3.1.1.min.js"></script>
  <script>
    // Como vemos en la página 2 de nuestros apuntes de clase
    // Invocamos al método Ajax del objeto Jquery
    $.ajax('alumnos.xml', {
      success: procesarRespuesta,
      error: procesarError
    });
    //Definimos las funciones
    function procesarRespuesta(datos, statusText, jqXHR) {
      //Definimos las funciones necesarias
      var alumnos =$(datos).find('alumno');
      var table = $('<table>');
      var                                     rowTitle                                     =
      $('<tr><th>Nombre</th><th>Apellidos</th><th>Nota</th><th>Convocatoria</th></tr>
      ');
      //Añadimos a la tabla la fila
      table.append(rowTitle);
      //Realizamos la búsqueda en el DOM y ejecutamos la función
      alumnos.each(listar);
      //Definimos qué hará nuestra función de búsqueda
      function listar() {
        var nombre = $(this).find('nombre').html();
        var apellido = $(this).find('apellido').html();
        var nota = $(this).find('nota').html();
        var convocatoria =$(this).find('nota').attr('convocatoria');

        var row = $('<tr>');

        row.html('<td>'+nombre+'</td><td>'+apellido+'</td><td>'+nota+'</td><td>'+
convocatoria+'</td>');
        table.append(row);

      }

      $('<body>').append(table);
```

```
    }  
    function procesarError(jqXHR, textStatus, error) {  
        console.log(error, textStatus);  
    }  
</script>  
  
</head>  
<body>  
  
</body>  
</html>
```

#### B. Petición AJAX y procesamiento de respuesta HTML

```
<!doctype html>  
<html lang="ES">  
<head>  
    <meta charset="UTF-8">  
    <title></title>  
    <script src="https://code.jquery.com/jquery-3.1.1.min.js"></script>  
</head>  
<body>  
    <script>  
        // Como vemos en la página 7 de nuestros apuntes de clase  
        // Invocamos al método Ajax  
        $.ajax({  
            url: 'alumnos.html',  
            dataType: 'html',  
            success: procesarRespuesta,  
            error: procesarError  
        });  
        //Definimos las funciones  
        function procesarRespuesta(datos, textStatus, jqXHR) {  
            //Definimos las funciones necesarias  
            $datos = $(datos);  
            //Agregamos los datos al body  
            $('body').append($datos);  
        }  
        //Si el procesamiento es erróneo, nos arroja el error por la consola  
        function procesarError(jqXHR, textStatus, error) {  
            console.log(error, textStatus);  
        }  
    </script>  
</body>  
</html>
```

## C. Petición AJAX y procesamiento de respuesta JSON

```
<!doctype html>
<html lang="ES">
<head>
  <meta charset="UTF-8">
  <title></title>
  <script src="https://code.jquery.com/jquery-3.1.1.min.js"></script>

</head>
<body>
  <script>
    // Como vemos en la página 8 de nuestros apuntes de clase
    // Invocamos al método Ajax
    $.ajax({
      url: 'alumnos.json',
      dataType: 'json',
      success: procesarRespuesta,
      error: procesarError
    });
    //Definimos las funciones
    function procesarRespuesta(datos, textStatus, jqXHR) {
      //Definimos las funciones necesarias
      var alumnos =datos.alumnos;
      var table = $('<table>');
      var                                     rowTitle                                     =
      $('<tr><th>Nombre</th><th>Apellidos</th><th>Nota</th><th>Convocatoria</th></tr>
      ');
      //Añadimos a la tabla la fila
      table.append(rowTitle);
      //Definimos el bucle que servirá para recorrer la array de datos
      for (var i=0; i<alumnos.length; i++){
        //Definimos las variables en la que almacenaremos los datos
        var nombre = alumnos[i].nombre;
        var apellido = alumnos[i].apellido;
        var nota = alumnos[i].nota;
        var convocatoria = alumnos[i].convocatoria;
        //Definimos la fila en la que insertaremos los datos recogidos
        var row = $('<tr>');
        //Añadimos los datos dentro de la fila, con sus correspondientes
        td

        row.html('<td>'+nombre+'</td><td>'+apellido+'</td><td>'+nota+'</td><td>'+
        convocatoria+'</td>');
        //Agregamos la fila a la tabla
        table.append(row);
      }
    }
  </script>
</body>
</html>
```

```
        //Agregamos la tabla al body
        $('body').append(table);

    }
    //En caso de que falle el procesamiento de datos, mostrará un error por
consola
    function procesarError(jqXHR, textStatus, error) {
        console.log(error, textStatus);
    }
</script>
</body>
</html>
```



### III. Bibliografía

**The jQuery Foundation.** .find(). [En línea] [Citado el: 22 de 02 de 2018.]  
<https://api.jquery.com/find/>.

**GrupoStudium. 2018.** Apuntes de la asignatura. Sevilla : s.n., 2018.