

DESARROLLO WEB CLIENTE (DWC)



CURSO 2017-2018

11 / 03 / 2018

PRÁCTICA TEMA 7

COMUNICACIÓN ASÍNCRONA CON JAVASCRIPT

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	6
III.	Bibliografía	7

I. Introducción

La realización de esta práctica consta de dos tareas a completar:

En la primera de estas tareas, nos pide que implementemos una aplicación que lea un fichero de datos y que lo procese para añadir una tabla con esa información a nuestra 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(`document.body`).

Adicionalmente, debemos tener en cuenta los siguientes factores:

- 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, se nos pide crear una función que permita cargar un fragmento de código HTML, que será generado mediante AJAX utilizando una plantilla, y que lo insertemos en el cuerpo de la página.

Estas son las especificaciones adicionales:

- El archivo HTML con los datos de los alumnos está disponible en la plataforma con el nombre de `alumnos.html` .
- Tenga en cuenta que hay que sobrescribir el tipo multimedia de la respuesta para que sea interpretado correctamente.

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>
    //Vamos a trabajar con navegadores modernos, por tanto, y como se dijo en
    clase, omitimos los métodos alternativos ActiveXObject
    httpRequest = new XMLHttpRequest();
    //Indicamos que cuando se realice la carga del httpRequest, se ejecute la
    función de procesar la respuesta
    httpRequest.onload = procesarRespuesta;
    //Abrimos la petición, indicando el método, el fichero y si va a ser
    asíncrono o no
    httpRequest.open('GET', 'alumnos.xml', true);
    httpRequest.send(null);

    //Funcion para procesar respuesta AJAX (Pág 16 de los apuntes de clase)
    function procesarRespuesta() {
      var elementos = httpRequest.responseXML;
      var item = procesarElemento(elementos);
      document.body.appendChild(item);
    }
    //Función para procesar la tabla que generaremos con los datos
    function procesarElemento(elementos) {
      //Para hacer el código mas comprensible vamos a ir creando los
      elementos para posteriormente anexarlos a donde corresponda
      //Generamos el elemento tabla
      var table = document.createElement('table');
      //Generamos el elemento fila (row)
      var row=document.createElement('tr');
      /*Generamos las 4 columnas que contiene nuestra tabla y modificamos
      su contenido con método textContent.
      Inicialmente habíamos usado el método innerText, pero no es compatible
      con algunos navegadores.*/
      var column1=document.createElement('th');
      column1.textContent="Nombre";
      var column2=document.createElement('th');
      column2.textContent="Apellido";
      var column3=document.createElement('th');
      column3.textContent="Nota";
      var column4=document.createElement('th');
      column4.textContent="Convocatoria";
```

```

        //Añadimos a la fila (row), las 4 columnas que lo componen como
elementos hijos
        row.appendChild(column1);
        row.appendChild(column2);
        row.appendChild(column3);
        row.appendChild(column4);
        //Añadimos a la tabla el elemento fila (row)
        table.appendChild(row);
        //Generamos una variable que almacena los elementos por nombre de
httpRequest.responseXML
        var alumnos = elementos.getElementsByTagName("alumno");
        //Iteramos los elementos para crear los tr y td. Una vez creados,
modificamos el contenido con textContent
        for (var i = 0; i < alumnos.length; i++) {
            var rows=document.createElement('tr');
            var nombre=document.createElement('td');
            var apellido=document.createElement('td');
            var nota=document.createElement('td');
            var convocatoria=document.createElement('td');
            /* Hemos de tener en cuenta que el tratamiento que se hace de
la respuesta XML es igual a lo que
            se puede hacer en HTML (navegación por el DOM), por lo que es
posible extraer la información de los atributos
            con el método getAttribute (para el caso de la convocatoria) y
del contenido textual con textContent */
            nombre.textContent =
alumnos[i].getElementsByTagName("nombre")[0].childNodes[0].nodeValue;
            apellido.textContent =
alumnos[i].getElementsByTagName("apellido")[0].childNodes[0].nodeValue;
            nota.textContent =
alumnos[i].getElementsByTagName("nota")[0].childNodes[0].nodeValue;
            convocatoria.textContent =
alumnos[i].getElementsByTagName("nota")[0].getAttribute("convocatoria");
            //Una vez generados los elementos, los añadimos
            rows.appendChild(nombre);
            rows.appendChild(apellido);
            rows.appendChild(nota);
            rows.appendChild(convocatoria);
            table.appendChild(rows);
        }
        return table;
    }
</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>
    //Vamos a trabajar con navegadores modernos, por tanto, y como se dijo en
    clase, omitimos los métodos alternativos XMLHttpRequest
    httpRequest = new XMLHttpRequest();
    //Indicamos que cuando se realice la carga del httpRequest, se ejecute la
    función de procesar la respuesta
    httpRequest.onload = procesarRespuesta;
    //Abrimos la petición, indicando el método, el fichero y si va a ser
    asíncrono o no
    httpRequest.open('GET', 'alumnos.html', true);
    //Como se indica en la práctica, debemos sobrescribir el tipo multimedia
    de la respuesta
    httpRequest.overrideMimeType('text/html');
    httpRequest.send(null);

    /*En los ejercicios de teoría, se adjuntaba un método para mostrar el
    progreso durante la carga, pero al no pedirse
    explícitamente en la práctica, hemos omitido ese fragmento de código*/

    //Función para procesar respuesta AJAX (Pág 14 de los apuntes de clase)
    function procesarRespuesta() {
      var respuesta = httpRequest.responseText;
      var contenedor = document.createElement('div');
      contenedor.innerHTML = respuesta;
      document.body.appendChild(contenedor);
    }
  </script>
</head>
<body>

</body>
</html>
```

III. Bibliografía

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

StackOverFlow. Difference between text content vs inner text. [En línea] [Citado el: 17 de 02 de 2018.] <https://stackoverflow.com/questions/35213147/difference-between-text-content-vs-inner-text>.

W3School.com. HTML DOM textContent Property. [En línea] [Citado el: 17 de 02 de 2018.] https://www.w3schools.com/jsref/prop_node_textcontent.asp.