



JavaScript: Un Curso Rápido

Parte IV: Análisis de XML

Traducción de Dr. Roberto Solís Robles

Customized Java EE Training: <http://courses.coreservlets.com/>

Servlets, JSP, JSF 2.0, Struts, Ajax, GWT 2.0, Spring, Hibernate, SOAP & RESTful Web Services, Java 6.

Developed and taught by well-known author and developer. At public venues or onsite at *your* location.

Temas

- **Motivación**
- **Obteniendo un documento**
 - Via Ajax
 - La página XHTML actual en forma de DOM
 - De un string (para pruebas interactivas)
- **Principales clases relacionadas con XML**
 - Document
 - Element
 - Node



Intro

Customized Java EE Training: <http://courses.coreservlets.com/>

Servlets, JSP, JSF 2.0, Struts, Ajax, GWT 2.0, Spring, Hibernate, SOAP & RESTful Web Services, Java 6.

Developed and taught by well-known author and developer. At public venues or onsite at *your* location.

Para qué conocer técnicas de XML en JavaScript?

- **Razón principal**

- Para extraer información de un XML enviado por el servidor en respuesta a una solicitud Ajax.
 - Más adelante discutiremos los pros y cons de enviar XHTML, JSON y XML

- **Razón secundaria**

- El navegador representa internamente una página en forma de DOM (incluso si la página se escribió en HTML 4 o viola la sintaxis de XML).
- Por tanto, los métodos relacionados con XML pueden también ser usados para extraer información sobre la página actual

- **Nota**

- Si ya conoce el API de DOM de Java u otro lenguaje, el API de JavaScript es muy similar

Visión General del API

- **API es muy similar a la API DOM de Java**
 - Si conoce las clases y métodos de Java org.w3c.dom, son casi las mismas en JavaScript
- **Clase Document**
 - Representa al documento de alto nivel
 - También una versión especializada que representa la página XHTML
- **Clase Element**
 - Representa un elemento XML/XHTML
 - Hereda método de la clase Node mas extras
- **Clase Node**
 - Representa un node en el árbol XML
 - Element es el tipo principal de nodo, pero hay también nodos de texto, nodos CDATA, y otros
 - La mayoría de los métodos de Element se heredan de aquí

Temas más Importantes

- **Métodos y variables muy importantes**
 - Como tratar la respuesta de Ajax como XML
 - `var xmlDoc = response.responseXML;`
 - Algunos navegadores lo dejan tomar `response.responseText` y entonces lo tratan como un documento XML, pero esto no es portable.
 - Como obtener un arreglo de subelementos
 - `xmlDoc.getElementsByTagName(...)`
 - Como obtener un atributo de un elemento
 - `algunElemento.getAttribute(...)`
 - Como obtener el contenido de un elemento
 - `algunElemento.firstChild.nodeValue`

Puede hacer muchas aplicaciones Ajax (si es que usan XML!) usando solo estas cuatro técnicas!

Temas de Segundo Nivel de Importancia

- **Temas de importancia media**
 - Elemento de nivel superior
 - `xmlDoc.documentElement` (y `...nodeName`)
 - Nombre de la etiqueta principal de un elemento
 - `algunElemento.nodeName`
 - Todos los atributos de un elemento
 - `algunElemento.attributes`
 - Nodos hijo de un elemento
 - `algunElemento.childNodes`

Esta sección cubre muchos temas, y a menos que tenga experiencia con XML/DOM, no podrá seguirlos todos. Pero, para la mayoría de las aplicaciones Ajax, el contenido de estas dos diapositivas es por mucho el mas importante de conocer.

Resumen de Clases Principales

- **Clase Document**
 - Propiedades
 - documentElement
 - Métodos
 - getElementsByTagName, getElementById (solo XHTML)
- **Clase Element**
 - Métodos
 - getAttribute, getElementsByTagName, hasAttribute
- **Node**
 - Propiedades
 - attributes, childNodes, firstChild, lastChild, nextSibling, nodeName, nodeType, nodeValue, parentNode, previousSibling
 - Métodos
 - hasAttributes, hasChildNodes, normalize



La Clase Document

Customized Java EE Training: <http://courses.coreservlets.com/>

Servlets, JSP, JSF 2.0, Struts, Ajax, GWT 2.0, Spring, Hibernate, SOAP & RESTful Web Services, Java 6.

Developed and taught by well-known author and developer. At public venues or onsite at *your* location.

Obteniendo Document: La Vida Real

- **Obteniendo un documento XML general**
 - Se obtiene un documento XML como resultado de una solicitud Ajax
 - `var xmlDocument = request.responseXML;`
- **Obteniendo un doc XML que representa una página XHTML**
 - Use la variable interna “document”
 - Muchas características especiales se aplican a esta versión especializada
 - Método getElementById, propiedad innerHTML para elementos, propiedades de formas, anclas e imagenes, insensibilidad a mayusculas/minusculas de getElementsByTagName
 - Estas no son generales a XML, y no aplican en un doc XML que viene de la red en respuesta a una solicitud Ajax
 - http://www.w3schools.com/html/dom/dom_obj_document.asp

Obteniendo Document: Practica y Aprendizaje

- **Firefox**

```
var xmlString = "<blah>...</blah>";  
var parser = new DOMParser();  
var xmlDocument =  
    parser.parseFromString(xmlString, "application/xml");
```

- **Internet Explorer**

```
var xmlDocument =  
    new ActiveXObject("Microsoft.XMLDOM");  
xmlDocument.async = false;  
xmlDocument.loadXML(xmlString);  
return(xmlDocument);
```

- **Determinando cual es cual**

- Verifique si (typeof DOMParser != "undefined")
 - True: use enfoque Firefox
 - False: use enfoque IE
- Advertencia: esto es solo para la practica

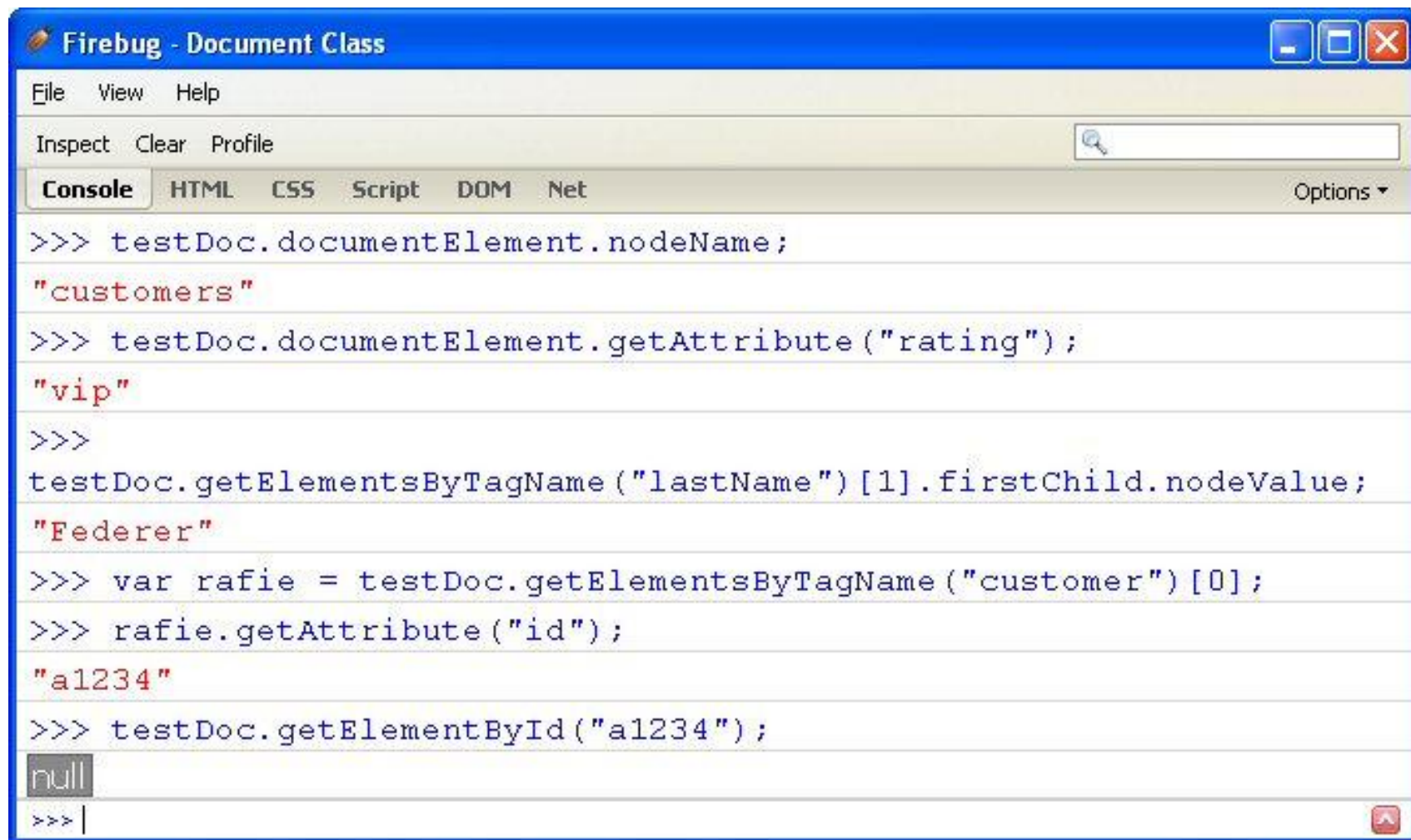
La clase Document

- **Propiedad documentElement**
 - El Element raíz del documento
- **Método getElementById**
 - Regresa el Element con el ID especificado.
 - Para documentos XHTML solamente!
 - Se refiere al atributo que el DTD define como un “atributo de id”, no necesariamente llamado “id”. No hace concordancia con atributos llamados “id” en los documentos regulares XML.
 - Debería ser sensitivo a mayusculas/minusculas, pero IE no lo es
- **getElementsByTagName**
 - Regresa un arreglo de Elements que tienen un nombre de etiqueta dado
 - Puede usar “*” para todos los Elements en el documento
 - No soportado en IE 5
 - Sensitivo a mayus/minus para documentos regulares XML
 - No sensitivo para documentos XHTML

La Clase Document : Ejemplos (Codigo)

```
function getXmlDoc(xmlString) {  
    var parser = new DOMParser();  
    var xmlDocument =  
        parser.parseFromString(xmlString, "application/xml");  
    return(xmlDocument);  
}  
  
var test =  
    "<customers rating='vip'>" +  
        "<customer id='a1234'>" +  
            "<firstName>Rafael</firstName>" +  
            "<lastName>Nadal</lastName>" +  
        "</customer>" +  
        "<customer id='a1235'>" +  
            "<firstName>Roger</firstName>" +  
            "<lastName>Federer</lastName>" +  
        "</customer>" +  
    "</customers>";  
  
var testDoc = getXmlDoc(test);
```

La Clase Document : Ejemplos (Resultados)



The screenshot shows the Firebug console window titled "Firebug - Document Class". The console has tabs for Console, HTML, CSS, Script, DOM, and Net. The Console tab is active, showing a series of JavaScript commands and their results. The commands are: 1. `testDoc.documentElement.nodeName;` with result `"customers"`. 2. `testDoc.documentElement.getAttribute("rating");` with result `"vip"`. 3. `testDoc.getElementsByTagName("lastName")[1].firstChild.nodeValue;` with result `"Federer"`. 4. `var rafie = testDoc.getElementsByTagName("customer")[0];` followed by `rafie.getAttribute("id");` with result `"a1234"`. 5. `testDoc.getElementById("a1234");` with result `null`. The console input field at the bottom shows `>>> |`.

```
Firebug - Document Class
File View Help
Inspect Clear Profile
Console HTML CSS Script DOM Net Options
>>> testDoc.documentElement.nodeName;
"customers"
>>> testDoc.documentElement.getAttribute("rating");
"vip"
>>>
testDoc.getElementsByTagName("lastName")[1].firstChild.nodeValue;
"Federer"
>>> var rafie = testDoc.getElementsByTagName("customer")[0];
>>> rafie.getAttribute("id");
"a1234"
>>> testDoc.getElementById("a1234");
null
>>> |
```




Nodos y Elementos

Customized Java EE Training: <http://courses.coreservlets.com/>

Servlets, JSP, JSF 2.0, Struts, Ajax, GWT 2.0, Spring, Hibernate, SOAP & RESTful Web Services, Java 6.

Developed and taught by well-known author and developer. At public venues or onsite at *your* location.

La Clase Element

- **getAttribute**
 - Obtiene el valor de un atributo especificado.
 - Por ejemplo, si el elemento se refiere a `<foo bar="a" baz="b">...</foo>`,
`element.getAttribute("baz")` returns "b"
- **getElementsByTagName**
 - Regresa un arreglo de subelementos que tienen un nombre de etiqueta especificado
 - Subelementos pueden estar anidados arbitrariamente
- **hasAttribute**
 - Prueba si el elemento tiene un atributo de cierto nombre
- **También hereda de la clase Node**
 - Todos los Elements son Nodes, pero no vice versa

La Clase Node : Propiedades

- **attributes**
 - Un arreglo de los atributos (llame a nodeName sobre cada uno para obtener sus nombres)
- **childNodes**
 - Un arreglo de nodos hijos directos. Longitud cero si no tiene hijos.
- **firstChild, lastChild, parentNode**
 - Nodos hijos especificos. Nodo padre(null para elemento superior).
- **nextSibling, previousSibling**
 - Hijos del nodo padre relacionados
- **nodeName**
 - Para nodos Element, el nombre del elemento XML
- **nodeType**
 - Node.ELEMENT_NODE, Node.TEXT_NODE, Node.ATTRIBUTE_NODE, Node.CDATA_SECTION_NODE, y otras opciones. **Falla en algunas versiones de IE!**
- **nodeValue**
 - Para nodos Text, el contenido. Llame a normalize primero.

La clase Node : Métodos

- **hasAttributes**
 - Tiene este Node atributos?
- **hasChildNodes**
 - Tiene este Node hijos?
- **normalize**
 - Fusiona nodos de texto multilinea.
 - Importante si el elemento tiene un contenido que abarca multiples lineas o tiene espacio en blanco extra.
 - Puede llamarlo sobre el elemento raiz solo una vez.
 - `xmlDoc.documentElement.normalize();`

Resumen

- **Como tratar la respuesta Ajax como XML**
 - `var xmlDoc = response.responseXML;`
 - Para practica y aprendizaje, puede hacer un string que contenga etiquetas XML y convertirlo en un documento XML usando funciones específicas de Firefox o IE.
- **Como obtener un arreglo de subelementos**
 - `xmlDoc.getElementsByTagName(...)`
- **Como obtener un atributo de un elemento**
 - `algunElemento.getAttribute(...)`
- **Como obtener el contenido de un elemento**
 - `algunElemento.firstChild.nodeValue`



Preguntas?

Customized Java EE Training: <http://courses.coreservlets.com/>

Servlets, JSP, JSF 2.0, Struts, Ajax, GWT 2.0, Spring, Hibernate, SOAP & RESTful Web Services, Java 6.

Developed and taught by well-known author and developer. At public venues or onsite at *your* location.