

¿Qué ofrece Autentia Real Business Solutions S.L?

Somos su empresa de **Soporte a Desarrollo Informático**.
 Ese apoyo que siempre quiso tener...

1. Desarrollo de componentes y proyectos a medida



2. Auditoría de código y recomendaciones de mejora

3. Arranque de proyectos basados en nuevas tecnologías

1. Definición de frameworks corporativos.
2. Transferencia de conocimiento de nuevas arquitecturas.
3. Soporte al arranque de proyectos.
4. Auditoría preventiva periódica de calidad.
5. Revisión previa a la certificación de proyectos.
6. Extensión de capacidad de equipos de calidad.
7. Identificación de problemas en producción.



4. Cursos de formación (impartidos por desarrolladores en activo)

Spring MVC, JSF-PrimeFaces /RichFaces,
 HTML5, CSS3, JavaScript-jQuery

Gestor portales (Liferay)
 Gestor de contenidos (Alfresco)
 Aplicaciones híbridas

Tareas programadas (Quartz)
 Gestor documental (Alfresco)
 Inversión de control (Spring)

Control de autenticación y
 acceso (Spring Security)
 UDDI
 Web Services
 Rest Services
 Social SSO
 SSO (Cas)

JPA-Hibernate, MyBatis
 Motor de búsqueda empresarial (Solr)
 ETL (Talend)

Dirección de Proyectos Informáticos.
 Metodologías ágiles
 Patrones de diseño
 TDD

BPM (jBPM o Bonita)
 Generación de informes (JasperReport)
 ESB (Open ESB)



Empieza el jaleo: publicamos el libro
INFORMÁTICA PROFESIONAL
LAS REGLAS NO ESCRITAS PARA TRIUNFAR EN LA
EMPRESA

-¿Crees que la informática es sólo programar?
-¿Sabrías organizar eficientemente un equipo?
-¿Quién te resuelve las dudas sobre esta
profesión?..... o te siembra más?

Hosting patrocinado por **enREDados**



Inicio **Quienes somos** **Tutoriales** **Formación** **Comparador de salarios** **Comentar libro** **Charlas** **Más**

Estas en: **Inicio** **Tutoriales** AjaxSingle: el partialSubmit de RichFaces.

Últimas Noticias

- » Nuestros tutoriales alcanzan la cifra de 10 millones de visitas!!!
- » Publicado el primer libro de TDD en castellano de Carlos Blé.
- » Comentando: Esta no es mi empresa: El desapego de los profesionales del siglo XXI de Ignacio Muro Benayas.
- » Impresiones y video de asistentes al primer coding-dojó de agilismo.es
- » Tarta v. 7º aniversario de AdictosAlTrabajo.com

+Noticias Destacadas

- » Impresiones y video de asistentes al primer coding-dojó de agilismo.es
- » "Informática profesional: Las reglas no escritas para triunfar en la empresa." Pincha para comprar..
- » Autentia patrocina el primer Agile Open Spain
- » Lanzamiento del nuevo Web de Autentia

+Comentarios Cómic

+Enlaces

Catálogo de servicios Autentia (PDF 6,2MB)



En formato comic...

Acceso de usuarios registrados:

E-mail:

Contraseña:

[Deseo registrarme](#)

[He olvidado mis datos de acceso](#)

Tutorial desarrollado por



Jose Manuel Sánchez Suárez

Consultor tecnológico de desarrollo de proyectos informáticos.

Puedes encontrarme en [Autentia](#)

Somos expertos en Java/J2EE

Catálogo de servicios de Autentia

[Descargar \(6,2 MB\)](#)

[Descargar en versión comic \(17 MB\)](#)

[AdictosAlTrabajo.com](#) es el Web de difusión de conocimiento de [Autentia](#).



[Catálogo de cursos](#)

Descargar este documento en formato PDF: [richFacesAjaxSingle.pdf](#)

Fecha de creación del tutorial: **2010-02-02**

AjaxSingle: el partialSubmit de RichFaces.

0. Índice de contenidos.

- 1. Introducción.
- 2. Entorno.
- 3. Inyectando ajax a componentes JSF.
- 4. Nuestro propio partialSubmit con el soporte de Ajax4JSF y facelets.
- 5. Referencias.
- 6. Conclusiones.

1. Introducción

Después de haber visto una breve [introducción a RichFaces](#) y conocer el soporte que proporciona para Ajax de la mano de [Ajax4JSF](#), en este tutorial vamos a examinar la posibilidad de hacer uso del atributo ajaxSingle que funciona como el atributo partialSubmit de ICEfaces, de modo que, asignado a un componente, se submite el valor del mismo al controlador iniciando un ciclo de vida JSF completo, que se aplicará únicamente a dicho componente.

La diferencia entre ICEfaces y RichFaces, es que el primero extiende los componentes de la implementación básica de JSF, de modo que tiene su propio <ice:inputText, sin embargo RichFaces no, con lo que dicho atributo no se puede asignar directamente al componente, hay que hacer uso de un segundo componente no visual que imprime el soporte para Ajax.

En la redacción de este tutorial se da por hecho que el lector tiene experiencia en JSF y tiene nociones de RichFaces e ICEfaces.

2. Entorno.

El tutorial está escrito usando el siguiente entorno:

- Hardware: Portátil MacBook Pro 17' (2.93 GHz Intel Core 2 Duo, 4GB DDR3 SDRAM).
- Sistema Operativo: Mac OS X Snow Leopard 10.6.1
- RichFaces 3.3.2.GA
- Maven 2.2.1.
- Eclipse 3.5: Ganymede, con IAM (plugin para Maven).
- Apache Tomcat 6.0.20 con la jdk 1.5.

3. Inyectando ajax a componentes JSF.

Sobre la base del ejemplo visto en la [introducción a RichFaces](#) vamos a añadir al formulario algún componente adicional, inyectando sobre los mismos la funcionalidad ajax que comentamos:

```
01 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
02 <html xmlns="http://www.w3.org/1999/xhtml">
```

Registra tu empresa:

Descubre las ventajas de registrar tu empresa en [AdictosAlTrabajo...](#)

[Registrar mi empresa](#)

[Listado de empresas ya registradas](#)



Web

[www.adictosaltrabajo.com](#)

Últimos tutoriales

2010-02-02
[AjaxSingle: el partialSubmit de RichFaces.](#)

2010-02-01
[Introducción a RichFaces.](#)

2010-01-29
[Transformación de mensajes en SOA con OpenESB](#)

2010-01-26
[JMeter. Uso de funciones.](#)

2010-01-18
[Autenticando los usuarios de Sonar contra un LDAP](#)

2010-01-18
[Introducción a jQuery UI.](#)

```

03     xmlns:f="http://java.sun.com/jsf/core"
04     xmlns:h="http://java.sun.com/jsf/html"
05     xmlns:ui="http://java.sun.com/jsf/facelets"
06     xmlns:a4j="https://ajax4jsf.dev.java.net/ajax"
07     xmlns:c="http://java.sun.com/jsp/jstl/core"
08     xmlns:rich="http://richfaces.org/rich"
09     >
10     <f:view>
11     <rich:panel>
12         <f:facet name="header">
13             <h:outputText value="RichFaces ajaxSingle Demo" />
14         </f:facet>
15
16         <a4j:form>
17             <h:panelGrid columns="2">
18                 <h:outputLabel value="Nombre" for="name"/>
19                 <h:inputText id="name" value="#{bean.name}" required="true">
20                     <a4j:support ajaxSingle="true" event="onblur" reRender="outputContainer"/>
21                 </h:inputText>
22                 <h:outputLabel value="Apellidos" for="surname"/>
23                 <h:inputText id="surname" value="#{bean.surname}" required="true">
24                     <a4j:support ajaxSingle="true" event="onblur" reRender="outputContainer"/>
25                 </h:inputText>
26                 <h:outputLabel value="NIF" for="nif"/>
27                 <h:inputText id="nif" value="#{bean.nif}">
28                     <a4j:support ajaxSingle="true" event="onblur" reRender="outputContainer"/>
29                 </h:inputText>
30                 <h:outputLabel value="Fecha de nacimiento" for="birthDay"/>
31                 <rich:calendar id="birthDay" datePattern="d/M/yyyy HH:mm" value="#{bean.birthDay}">
32                     <a4j:support ajaxSingle="true" event="onchanged" reRender="outputContainer"/>
33                 </rich:calendar>
34                 <h:panelGroup />
35                 <h:commandButton action="none" value="Asignar" />
36             </h:panelGrid>
37
38             <rich:messages>
39
40             <rich:spacer height="10px"/>
41
42             <rich:separator height="5px"/>
43             <rich:spacer height="20px"/>
44             <h:panelGrid columns="2" id="outputContainer">
45                 <h:outputLabel value="Nombre: " /><h:outputText value="#{bean.name}"/>
46                 <h:outputLabel value="Apellidos: " /><h:outputText value="#{bean.surname}"/>
47                 <h:outputLabel value="NIF: " /><h:outputText value="#{bean.nif}"/>
48                 <h:outputLabel value="Fecha de nacimiento: " /><h:outputText value="#{bean.birthDay}"/>
49             </h:panelGrid>
50
51         </a4j:form>
52     </rich:panel>
53 </f:view>
54 </html>
55 </rich:messages>
56
57

```

El resultado es que, conforme vamos introduciendo valor a los campos del formulario, el mismo se va submitiendo al servidor mediante una petición en segundo plano a través de Ajax, pasando por todo el ciclo de vida de JSF, únicamente para el componente que se submitte.

RichFaces ajaxSingle Demo

Nombre	<input type="text" value="Pablo"/>
Apellidos	<input type="text" value="Sánchez"/>
NIF	<input type="text" value="777"/>
Fecha de nacimiento	<input type="text" value="27/11/2008 12:00"/>
<input type="button" value="Asignar"/>	

Nombre:	Pablo
Apellidos:	Sánchez
NIF:	777
Fecha de nacimiento:	Thu Nov 27 12:00:00 CET 2008

Tenemos en la parte inferior una serie de campos de salida que muestran los valores que submitimos y podemos comprobar que los mismos cambian conforme vamos introduciendo datos en los campos de entrada del formulario.

Con todo ello podríamos realizar validaciones en el servidor con la apariencia de validaciones en el cliente (como ocurre con los campos nombre y apellidos), puesto que no se recarga toda la página.

4. Nuestro propio partialSubmit con el soporte de Ajax4JSF y facelets.

Vamos a reducir el código de nuestras páginas haciendo uso del soporte de facelets para la generación de nuestra propia tag library a través de la [creación de componentes por composición](#).

Ya hemos comprobado que funciona la submisión en segundo plano y la recarga, con lo que nos vamos a olvidar de la parte inferior con los campos de salida y, como consecuencia, del atributo reRender.

Bajo WEB-INF/facelets/tags crearemos un fichero inputText.xhtml con el siguiente código:

```

01 <?xml version="1.0" encoding="UTF-8"?>
02
03 <jsp:root xmlns:jsp="http://java.sun.com/JSP/Page"
04     xmlns:ui="http://java.sun.com/jsf/facelets"
05     xmlns:h="http://java.sun.com/jsf/html"
06     xmlns:f="http://java.sun.com/jsf/core"
07     xmlns:c="http://java.sun.com/jstl/core"
08     xmlns:a4j="https://ajax4jsf.dev.java.net/ajax"
09     xmlns:rich="http://richfaces.org/rich">
10
11     <ui:composition>
12         <!-- The id attribute is optional. empty by default -->

```

2010-01-18
[jQuery: cómo crear nuestros propios plugins.](#)

2010-01-18
[Cómo consumir un servicio web RESTful con el soporte de Ajax y JSON de jQuery.](#)

2010-01-18
[Introducción a jQuery.](#)

2010-01-17
[Introducción a Tapestry 5](#)

2010-01-14
[JMeter. Gestión de usuarios](#)

2010-01-14
[Patrón Visitor con commons-collections y sus Closures](#)

2010-01-12
[Creación de servicios web Restful, con soporte a persistencia, en NetBeans.](#)

2010-01-11
[JMeter y JSF. Extracción del parámetro ViewState](#)

2010-01-07
[Importar el correo de Microsoft Outlook al cliente de correo de Mac OS.](#)

2010-01-07
[Monitor de Hudson para Eclipse.](#)

2010-01-07
[Patrones de diseño de XML Schema](#)

2010-01-04
[Procesador Inteligente de Eventos \(IEP\) con OpenESB](#)

2010-01-04
[PHP Vs Java](#)

2009-12-29
[Tutorial de BPEL con OpenESB \(II\)](#)

2009-12-29
[Tutorial de BPEL con OpenESB \(I\)](#)

2009-12-28
[Pruebas funcionales de servicios web con soapUI](#)

2009-12-28
[SoapUI: jugando con web services](#)

2009-12-17
[¿Qué son el cloud computing y google app engine?](#)

2009-12-14
[JavaBean Datasource Ireport](#)

2009-12-11

```

13 <c:if test="#{empty id}">
14 <c:set var="id" value="" />
15 </c:if>
16 <!-- The label attribute is optional. Generate it if it is missing. -->
17 <c:if test="#{empty label}">
18 <c:set var="label" value="#{entity.class.simpleName}.#{field}" />
19 </c:if>
20 <!-- The required attribute is optional. false by default -->
21 <c:if test="#{empty required}">
22 <c:set var="required" value="false" />
23 </c:if>
24 <!-- The partialSubmit attribute is optional. false by default -->
25 <c:if test="#{empty partialSubmit}">
26 <c:set var="partialSubmit" value="false" />
27 </c:if>
28 <!-- The reRender attribute is optional. empty by default -->
29 <c:if test="#{empty reRender}">
30 <c:set var="reRender" value="" />
31 </c:if>
32
33 <!-- Label -->
34 <h:panelGroup>
35 <h:outputLabel id="labelFor#{id}" for="#{id}"
36 value="#{label}:" />
37 <h:outputText styleClass="required" rendered="#{required}">*</h:outputText>
38 </h:panelGroup>
39
40 <!-- input component -->
41 <h:panelGroup>
42 <h:inputText id="#{id}" value="#{value}"
43 required="#{required}"
44 <c:if test="#{partialSubmit}">
45 <a4j:support ajaxSingle="true" event="onblur" />
46 </c:if>
47 <ui:insert />
48 </h:inputText>
49 </h:panelGroup>
50
51 <!-- message -->
52 <h:panelGroup>
53 <rich:message id="messageFor#{id}" for="#{id}"
54 fatalLabelClass="fatal" errorLabelClass="error"
55 warnLabelClass="warn" infoLabelClass="info" />
56 </h:panelGroup>
57
58 </ui:composition>
59
60 </jsp:root>

```

Encapsula un componente de etiqueta con la descripción del campo que informa, además, de la obligatoriedad del mismo, el propio componente de entrada, y un componente que muestra los errores asociados a este.

Generaremos nuestra propia librería de tags, creando un fichero autentia.taglib.xml en el mismo directorio con el siguiente contenido:

```

01 <?xml version="1.0"?>
02 <!DOCTYPE facelet-taglib PUBLIC "-//Sun Microsystems, Inc.//DTD
03 Facelet Taglib 1.0//EN" "facelet-taglib_1_0.dtd">
04 <facelet-taglib>
05 <namespace>http://www.autentia.com/richFaces/facelets</namespace>
06 <tag>
07 <tag-name>inputText</tag-name>
08 <source>inputText.xhtml</source>
09 </tag>
10 </facelet-taglib>

```

Registramos la librería en el web.xml con el siguiente parámetro de contexto:

```

1 <context-param>
2 <param-name>facelets.LIBRARIES</param-name>
3 <param-value>/WEB-INF/facelets/tags/autentia.taglib.xml</param-value>
4 </context-param>

```

Y ya podemos hacer uso de nuestros tags en las páginas que lo requieran, el código de nuestra página quedaría como sigue

```

01 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-
02 transitional.dtd">
03 <html xmlns="http://www.w3.org/1999/xhtml"
04 xmlns:f="http://java.sun.com/jsf/core"
05 xmlns:h="http://java.sun.com/jsf/html"
06 xmlns:ui="http://java.sun.com/jsf/facelets"
07 xmlns:a4j="https://ajax4jsf.dev.java.net/ajax"
08 xmlns:c="http://java.sun.com/jsp/jstl/core"
09 xmlns:rich="http://richfaces.org/rich"
10 xmlns:tnt="http://www.autentia.com/richFaces/facelets">
11 <f:view>
12 <rich:panel>
13 <f:facet name="header">
14 <h:outputText value="RichFaces ajaxSingle Demo" />
15 </f:facet>
16
17 <a4j:form>
18 <h:panelGrid columns="3">
19
20 <tnt:inputText id="name" label="Nombre" value="#{bean.name}" required="true"
21 partialSubmit="true"/>
22 <tnt:inputText id="surname" label="Apellidos" value="#{bean.surname}" required="true"
23 partialSubmit="true"/>
24 <tnt:inputText id="nif" label="NIF" value="#{bean.nif}"
25 partialSubmit="true"/>
26
27 <h:outputLabel value="Fecha de nacimiento" for="birthDay"/>
28 <rich:calendar id="birthDay" datePattern="d/M/yyyy HH:mm" value="#{bean.birthDay}">
29 <a4j:support ajaxSingle="true" event="onchanged" reRender="outputContainer"/>
30 </rich:calendar>
31 <rich:messages for="birthDay" />
32
33 <h:panelGroup />
34 <h:commandButton action="none" value="Asignar" />
35 </h:panelGroup />
36
37 </h:panelGrid>
38
39 </a4j:form>
40
41 </rich:panel>
42 </f:view>
43 </html>

```

Contract-First web services
con Visual Studio 2008

2009-12-09
Integrando Sonar con Hudson

2009-12-09
Apache + Tomcat: Balanceo
de carga y alta disponibilidad

2009-12-08
MySQL: Replicación de bases
de datos en MySQL

2009-12-07
Analizando la calidad del
código Java con Sonar

2009-12-03
Instalar OpenESB 2.1 e
Introducción

2009-11-25
Tutorial de Google Forms

2009-11-23
Alfresco - Modificando y
eliminando contenido desde
nuestras aplicaciones Java

2009-11-23
Alfresco - Añadiendo
contenido desde nuestras
aplicaciones Java

2009-11-23
Haciendo funcionar Google
Chromium Operating System
sobre Windows XP

2009-11-23
Redimensionar el tamaño de
una partición de VirtualBox

2009-11-20
El Arte de las Presentaciones.
Siguiendo la Senda Zen.

2009-11-18
Tutorial basico de google
wave bots

2009-11-13
Introducción a Escritorios
Animado (Winamp y
MilkWave)

2009-11-12
Maven Assembly Plugin:
empaquetando aplicaciones
con Maven para la ejecución
de procesos batch.

2009-11-12
Tutorial básico sobre Google
Application Engine

2009-11-11
Instalación de Glassfish 2.1

2009-11-26
Rational Software Architect y
Rational RequisitePro

2009-10-27
Cómo conseguir que

Bastante menos código para crear los campos del formulario. Deberíamos continuar con el calendario y con cada componente de formulario que vayamos utilizando.

Indirectamente estamos encapsulando la librería de componentes con la que realmente trabajamos.

5. Referencias.

- <http://www.jboss.org/richfaces/docs.html>
- <http://livedemo.exadel.com/richfaces-demo/richfaces/support.jsf?c=support&tab=usage>

6. Conclusiones.

El soporte de ajax no es nativo en los componentes de RichFaces, se debe añadir un componente de Ajax4JSF, pero nos da más control sobre qué partes de la página deben repintarse. Son dos puntos de vista distintos, tener el control o no tenerlo, puesto que con ICEfaces, precisamente de lo que te despreocupas es de eso mismo.

Con facelets podemos encapsular parte del problema y reducir el código de nuestras páginas xhtml. Es imprescindible el uso de un sistema de plantillas, usemos la tecnología que usemos para la vista.

Un saludo.

Jose

jmsanchez@autentia.com

¿Qué te ha parecido el tutorial? Déjanos saber tu opinión y ivota!

Muy malo Malo Regular Bueno Muy bueno

Votar

(Sólo para usuarios registrados)

» [Regístrate y accede a esta y otras ventajas](#) «

Anímate y coméntanos lo que pienses sobre este tutorial

Puedes opinar o comentar cualquier sugerencia que quieras comunicarnos sobre este tutorial; con tu ayuda, podemos ofrecerte un mejor servicio.

Enviar comentario

(Sólo para usuarios registrados)

» [Regístrate y accede a esta y otras ventajas](#) «

Autor

Mensaje de usuario registrado

- Puedes inscribirte en nuestro servicio de notificaciones haciendo clic [aquí](#).
- Puedes firmar en nuestro libro de visitas haciendo clic [aquí](#).
- Puedes asociarte al grupo AdictosAITrabajo en XING haciendo clic [aquí](#).
- Añadir a favoritos Technorati.  [ADD THIS BLOG TO MY FAVORITES](#)



Esta obra está licenciada bajo [licencia Creative Commons de Reconocimiento-No comercial-Sin obras derivadas 2.5](#)

Recuerda

Autentia te regala la mayoría del conocimiento aquí compartido ([Ver todos los tutoriales](#)). Somos expertos en: J2EE, Struts, JSF, C++, OOP, UML, UP, Patrones de diseño ... y muchas otras cosas.

¿Nos vas a tener en cuenta cuando necesites consultoría o formación en tu empresa?, ¿Vas a ser tan generoso con nosotros como lo tratamos de ser con vosotros?

Somos pocos, somos buenos, estamos motivados y nos gusta lo que hacemos

[Subversion avise a Hudson para lanzar una build](#)

2009-10-26
[Cómo instalar Hudson en Apache Tomcat](#)

2009-10-26
[Estilos externos en iReport](#)

2009-10-14
[Echo en MySQL](#)

2009-09-28
[iReport : Solución al problema de los parámetros dinámicos.](#)

2009-09-27
[Mavenizar Liferay SDK](#)

2009-09-27
[Crear un plugin con Liferay SDK](#)

Últimas ofertas de empleo

2009-07-31
[T. Información - Operador \(día / noche\) - BARCELONA.](#)

2009-06-25
[Atención a cliente - Call Center - BARCELONA.](#)

2009-06-19
[Otras - Ingeniería \(minas, puentes y puertos\) - VALENCIA.](#)

2009-06-17
[Comercial - Ventas - ALICANTE.](#)

2009-06-03
[Comercial - Ventas - VIZCAYA.](#)

Anuncios Google

[Tutoriales](#)

[Manuales](#)

[Mule Soa](#)

[Mule ESB Tutorial](#)

...

Autentia = Soporte a Desarrollo & Formación.

info@autentia.com

Gestión de contenidos

Tutoriales recomendados

Nombre	Resumen	Fecha	Visitas	Valoración	Votos	Pdf
AjaxSingle: el partialSubmit de RichFaces.	En este tutorial vamos a examinar la posibilidad de hacer uso del atributo ajaxSingle, en los componentes de RichFaces, que funciona como el atributo partialSubmit de ICEFaces.	2010-02-02	5	-	-	
Introducción a RichFaces.	RichFaces es una librería de componentes visuales para JSF con soporte para Ajax4JSF.	2010-02-01	101	-	-	
JMeter. Uso de funciones.	En este tutorial tratamos el uso de las funciones más habituales de la herramienta JMeter.	2010-01-26	280	-	-	
jQuery: cómo crear nuestros propios plugins.	En este tutorial vamos a ver cómo crear un plugin para jQuery sencillo, para sentar las bases, y otro un poco más elaborado.	2010-01-18	485	-	-	
Introducción a jQuery UI.	En los tutoriales anteriores hemos hecho una introducción a jQuery y hemos visto cómo crear plugins para jQuery. Ahora se trata de seguir ampliando conocimientos, haciendo uso de una librería que se basa en la misma.	2010-01-18	628	-	-	
Introducción a jQuery.	jQuery es una librería javascript ligera, rápida y concisa que simplifica el tratamiento de documentos HTML, el manejo de eventos, la creación de animaciones y las interacciones vía Ajax, para agilizar el desarrollo de aplicaciones web.	2010-01-18	647	Bueno	1	
Cómo consumir un servicio web RESTful con el soporte de Ajax y JSON de jQuery.	En este tutorial vamos a seguir analizando cómo explotar las funcionalidades de jQuery.	2010-01-18	427	-	-	
JMeter. Gestión de usuarios	En este tutorial tratamos la simulación de distintos usuarios, en la herramienta JMeter, mediante el archivo externo users.xml o mediante la función Counter.	2010-01-14	551	-	-	
JMeter y JSF. Extracción del parámetro ViewState	En este tutorial ofrecemos una solución a la parametrización del atributo ViewState, de JSF (Java Server Faces), cuando ejecutamos scripts de pruebas de carga mediante la herramienta JMeter.	2010-01-11	537	-	-	
Haciendo funcionar Google Chromium Operating System sobre Windows XP	Cómo instalar el nuevo SO de Google corriendo sobre XP	2009-11-23	1801	Muy bueno	1	

Nota:

Los tutoriales mostrados en este Web tienen como objetivo la difusión del conocimiento. Los contenidos y comentarios de los tutoriales son responsabilidad de sus respectivos autores. En algún caso se puede hacer referencia a marcas o nombres cuya propiedad y derechos es de sus respectivos dueños. Si algún afectado desea que incorporemos alguna reseña específica, no tiene más que solicitarlo. Si alguien encuentra algún problema con la información publicada en este Web, rogamos que informe al administrador rcanales@adictosaltrabajo.com para su resolución.