

¿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)



Powered by **autentia** Hosting patrocinado por **enREDados**

[Inicio](#) [Quienes somos](#) [Tutoriales](#) [Formación](#) [Empleo](#) [Colabora](#) [Comunidad](#) [Libro de Visitas](#) [Comic](#)

NUEVO ¿Quieres saber cuánto ganas en relación al mercado? pincha aquí...

[Ver cursos que ofrece Autentia](#)

[Descargar comics en PDF y alta resolución](#)



[NUEVO!] 2008-02-26



2008-02-24



2008-02-19



2008-02-17

Estamos escribiendo un libro sobre la profesión informática y estas viñetas formarán parte de él. Puedes opinar en la sección [comic](#).

Tutorial desarrollado por



Daniel Hernandez del Peso

Consultor tecnológico de desarrollo de proyectos informáticos. Constructor de [AdictosAlTrabajo.com](#) es el Web de difusión de conocimiento de [Autentia](#). Adictos Al Trabajo 2.0

Ingeniero en Informática

Puedes encontrarme en [Autentia](#)

Somos expertos en Java/J2EE

Catálogo de servicios de Autentia

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

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



[Catálogo de cursos](#)

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

Fecha de creación del tutorial: 2008-03-03

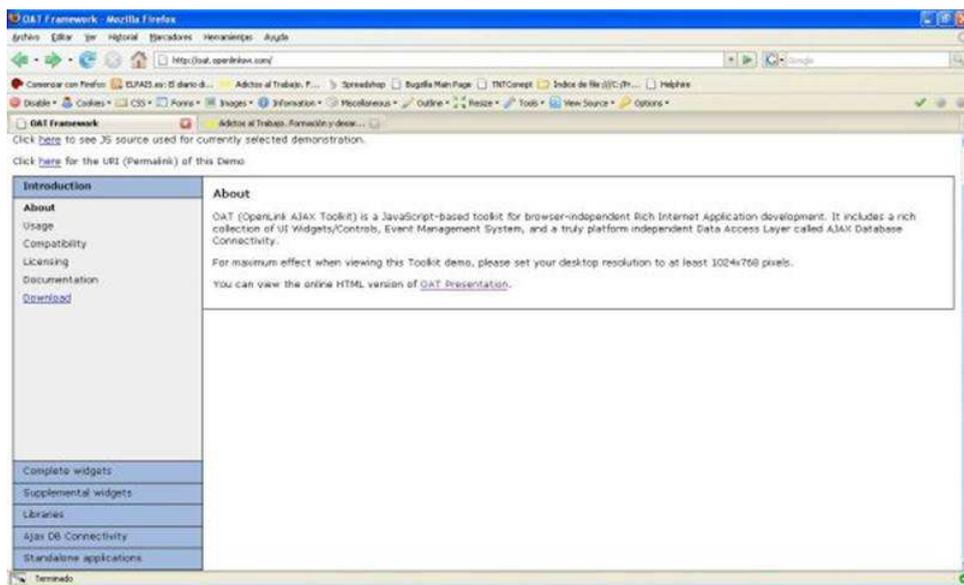
Tablas dinámicas "online"

Introducción

A raíz del [comparador de salarios](#) y de las gráficas que usamos para la explotación de los datos, nos dimos cuenta (ayudados por los comentarios de muchos de vosotros) de que las gráficas no ayudaban en un primer análisis de los datos, pero que para dar el siguiente paso, necesitábamos algo distinto, que permitiera combinar a nuestro antojo cualquiera de los datos entre sí. La solución fue una [tabla dinámica en Excel](#). Pero queríamos una solución online, así que rebuscamos en Internet para encontrar un producto que satisficiera nuestras necesidades. Hasta que encontramos el producto que hoy os presentamos: el [OpenLink AJAX Toolkit \(OAT\)](#)

Cómo empezar

Lo primero es obtener el framework. Para ello, visitamos su [página oficial](#)



En esta página hay tres cosas importantes: el enlace de descarga, la información de licencia (os recomiendo leer siempre las [licencias](#) de los productos que uséis, os podéis evitar muchos problemas innecesarios.) y los "widgets", donde encontraréis ejemplos de todo lo que podéis hacer con OAT.

Una vez que hemos leído la licencia, descargamos el producto (cuando se hizo este tutorial, la última versión era la 2.7, de diciembre de 2007) y lo descomprimos

Catálogo de servicios Autentia (PDF 6,2MB)



[En formato comic...](#)

Google

Web
 [www.adictosaltrabajo.com](#)

Últimos tutoriales

2008-03-03
[Tablas dinámicas online](#)

2008-02-29
[Generación automática de gráficas en un web](#)

2008-02-28
[Manual de instalación de OpenCms 7](#)

2008-02-28
[Creación de un proyecto en SourceForge.net](#)

2008-02-22
[Lucene: Analyzers, stemming y búsqueda de documentos similares.](#)

2008-02-22
[Crear un logger utilizado a través de aspectos con Spring AOP.](#)

2008-02-20
[Primeros pasos con PostgreSQL en Debian](#)

2008-02-17
[Cómo realizar pruebas unitarias con Spring y JUnit4 utilizando Gienah](#)

2008-02-15
[Creación de una aplicación con Spring e Hibernate desde 0](#)

2008-02-07
[Slimming básico de JBoss](#)

Últimas ofertas de empleo

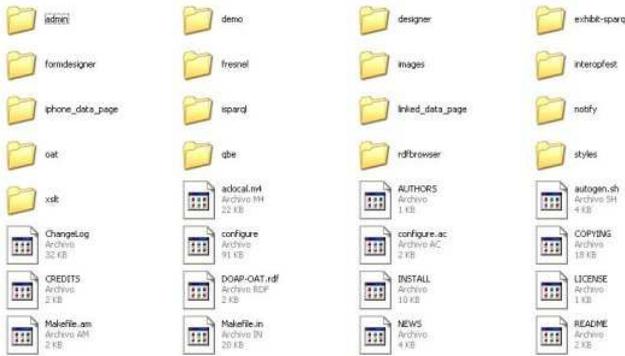
2008-02-06
[T. Información - Analista / Programador - MADRID.](#)

2008-02-04
[T. Información - Becario - MADRID.](#)

2008-01-28
[T. Información - Becario - MADRID.](#)

2008-01-25
[Otras Sin catalogar - MURCIA.](#)

2008-01-24
[T. Información - Analista / Programador - MADRID.](#)



Anuncios Google

[Hibernate Java](#)

[Hibernate ORM](#)

[Java Downloads](#)

[PDF Applet](#)

Dentro de todas estas carpetas, la más importante es la carpeta "oat", que es la que contiene todos los scripts necesarios para hacer funcionar los widgets. Además, dentro de esa carpeta hay otra, "docs", con documentación de todos los objetos y scripts. De hecho, vamos a tener esa documentación muy presente a la hora de crear nuestra tabla pivotante.

El otro elemento que tendremos en cuenta para crear nuestra tabla es el fichero "demo.js" que se encuentra en la carpeta "demo". Esta demo es la misma que se puede ver cuando se accede a la página oficial de OAT.

Ya lo tenemos todo... vamos a empezar con el ejemplo

Ejemplo de tabla pivotante

Lo primero que necesitamos es una página web normal y corriente... y la documentación

Como nos dice en la documentación (en el apartado de "Basic Operation"), agregamos los elementos que queremos cargar. Para nuestra tabla pivotante vamos a usar dos: la propia tabla y un elemento que nos permitirá calcular agregaciones de distinto tipo para cada dato (media, mediana, desviación típica, valor máximo, valor mínimo...)

El HTML queda, pues, inicialmente, así:

```

view plain print ?
01. <html>
02. <head>
03. <meta http-equiv="content-type" content="text/html; charset=windows-1250">
04. <script type="text/javascript">
05.     var featureList = ["pivot", "statistics"];
06. </script>
07. <script type="text/javascript" src="oat/loader.js"></script>
08. <title></title>
09. </head>
10. <body>
11. </body>
12. </html>
    
```

El script "loader.js" se encarga de detectar que características queremos usar y generar dinámicamente todas las inclusiones de scripts necesarias para que funcione

ahora hay que crear la tabla pivotante. Mirando la documentación, vamos a ver como se declara una tabla pivotante. Según la documentación, vemos que la llamada javascript para crear una tabla es la siguiente

```

view plain print ?
01. OAT.Pivot(div, chartDiv, filterDiv, headerRow, dataRows, headerRowIndex,
02.     headerColIndexes, filterIndexes, dataColumnIndex, optObj)
    
```

El significado de los parámetros, traduciendo la documentación es el siguiente:

- div
Nombre (id) de la capa en que se pintará la tabla
- chartDiv
Nombre (id) de la capa en que se pintará los gráficos asociados a la tabla. En este ejemplo no se van a usar
- filterDiv
Nombre (id) de la capa en que colocarán las opciones de filtrado de la tabla
- headerRow
Array con todos los nombres de los campos que se van mostrar en la tabla
- dataRows
Tabla de datos. Es un array de arrays. Cada array debe tener el mismo número de campos que el "headerRow", y en el mismo orden
- headerRowIndex
Array con los índices de la lista de campos que se pintarán en el eje Y de la tabla
- headerColIndexes
Array con los índices de la lista de campos que se pintarán en el eje X de la tabla
- filterIndexes
Array con los índices de la lista de campos que se usarán como filtro
- dataColumnIndex
posición de la lista en que se encuentra el campo que contiene el valor, es decir el dato que se pintará para cada combinación de X e Y
- optObj
Se trata de un objeto con opciones de configuración para determinadas propiedades de la tabla. su descripción viene en la documentación, así que no me extenderé en ello

O sea, que para seguir adelante hay que modificar el html, crear la tabla y las capas para albergar cada elemento. En nuestro caso, ya hemos dicho que no vamos a mostrar los gráficos, así que solo crearemos capas para la tabla y los filtros. Además, crearemos los arrays con los campos y con los datos. El nuevo HTML es el que sigue:

```

view plain print ?
01. <html>
02. <head>
03. <meta http-equiv="content-type" content="text/html; charset=windows-1250" >
04. <script type="text/javascript">
05.     var featureList = ["pivot","statistics"];
06. </script>
07. <script type="text/javascript" src="oat/loader.js"></script>
08.
09. <script type="text/javascript">
10.
11.     var cabecera = new Array("campo1", "campo2", "campo3", "campo4", "campo5", "valor");
12.
13.     var datos = new Array();
14.     var datos[0] = new Array("rojo", "1", "perro", "manzana", "juan", "5.5");
15.     var datos[1] = new Array("azul", "2", "gato", "pera", "pedro", "8");
16.     var datos[2] = new Array("verde", "3", "rata", "ciruela", "antonio", "6");
17.     var datos[3] = new Array("amarillo", "4", "ardilla", "mandarina", "miguel", "4.7");
18.
19.     var pTable = new OAT.Pivot('tabla', '', 'filtros', cabecera, datos,
20.                               [0,1], [2,3], [4], 5, null)
21. </script>
22.
23. <title></title>
24. </head>
25. <body>
26.     <div id="filtros"></div>
27.     <div id="tabla"></div>
28.
29. </body>
30. </html>

```

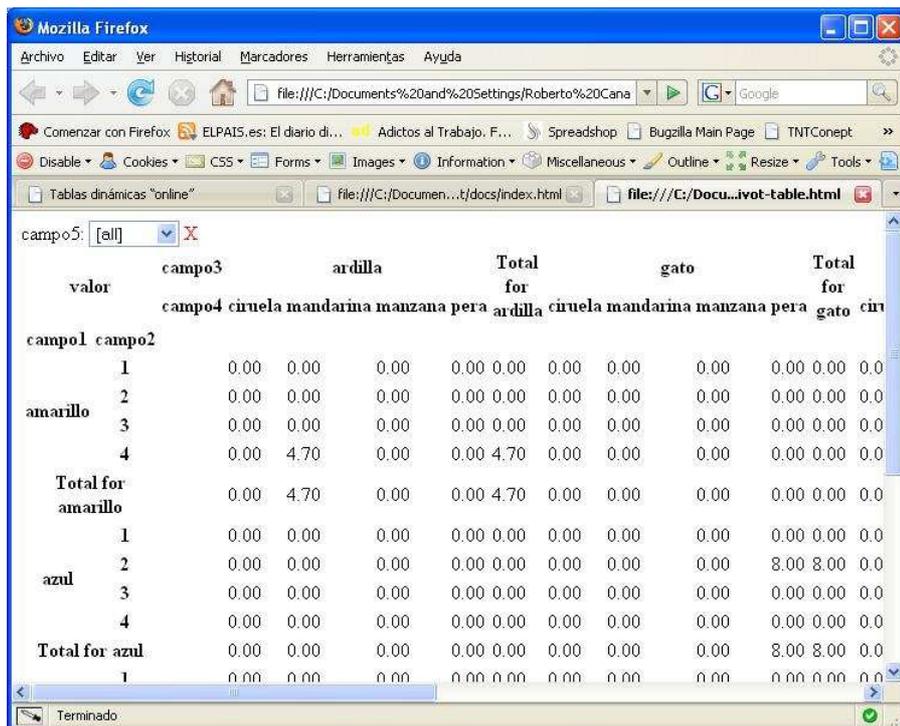
Pero aún no se ve nada... hay que decir al script que pinte la tabla. Para ello, vamos a hacer uso de otra recomendación que hace la documentación de OAT y usar la función init, que se llamará automáticamente cuando se pinte la página. este es el resultado de modificar el HTML

```

view plain print ?
01. <html>
02. <head>
03. <meta http-equiv="content-type" content="text/html; charset=windows-1250" >
04. <script type="text/javascript">
05.     var featureList = ["pivot","statistics"];
06. </script>
07. <script type="text/javascript" src="oat/loader.js"></script>
08.
09. <script type="text/javascript">
10.
11.     var cabecera = new Array('campo1', 'campo2', 'campo3', 'campo4', 'campo5', 'valor');
12.
13.     var datos = new Array();
14.     datos[0] = new Array('rojo', '1', 'perro', 'manzana', 'juan', '5.5');
15.     datos[1] = new Array('azul', '2', 'gato', 'pera', 'pedro', '8');
16.     datos[2] = new Array('verde', '3', 'rata', 'ciruela', 'antonio', '6');
17.     datos[3] = new Array('amarillo', '4', 'ardilla', 'mandarina', 'miguel', '4.7');
18.
19.     function init() {
20.         var pTable = new OAT.Pivot('tabla', '', 'filtros', cabecera, datos,
21.                                   [0,1], [2,3], [4], 5, null);
22.
23.         pTable.go();
24.     }
25. </script>
26.
27. <title></title>
28. </head>
29. <body>
30.     <div id="filtros"></div>
31.     <div id="tabla"></div>
32.
33. </body>
34. </html>

```

Y una vez hecho esto, ya tenemos funcionando la tabla:



Aunque el resultado no es demasiado atractivo todavía... Para solucionarlo, en la documentación de OAT para el componente "pivot.js" encontramos los estilos (clases CSS) que se aplican a la tabla, y que podemos definir a nuestro gusto.

Para este ejemplo, vamos a hacerlo más fácil... vamos a coger los estilos que se aplican en la demo. Los podemos encontrar en la distribución que nos hemos descargado, en la carpeta "styles/pivot.css"

Copiamos los estilos en nuestro ejemplo... y vemos cómo mejora el resultado

campo5:

valor		campo3	ardilla				Total for ardilla	gato				Total for gato	
campo1	campo2	campo4	ciruela	mandarina	manzana	pera		ciruela	mandarina	manzana	pera		ciruela
amarillo	1		0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	2		0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	3		0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	4		0.00	4.70	0.00	0.00	4.70	0.00	0.00	0.00	0.00	0.00	0.00
Total for amarillo			0.00	4.70	0.00	0.00	4.70	0.00	0.00	0.00	0.00	0.00	0.00
azul	1		0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	2		0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	8.00	8.00	0.00
	3		0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	4		0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Total for azul			0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	8.00	8.00	0.00
rojo	1		0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	2		0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	3		0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	4		0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Una vez que tenemos esto... vamos a ver como funciona:

- Si cambiamos el valor del combo, la tabla muestra solo los datos para el valor del combo seleccionado.
- Si pinchamos y arrastramos el nombre de una fila o columna, podemos colocarla donde queramos, incluso sacarla de la tabla y ponerla como filtro.
- Si pinchamos en la "X" roja al lado de los filtros, el campo pasa a la tabla.
- Si pinchamos en los nombres de las columnas (sin arrastrar), aparecen opciones de ordenación.
- Si pinchamos en el nombre del dato que se muestra en el cuerpo de la tabla ("valor" en nuestro caso) aparecen opciones para pintar los datos en un formato u otro (como números enteros, como moneda...) y otras.

Funciones de agregación

Ya tenemos lo más importante funcionando, y ahora vamos a usar el otro componente que hemos instanciado (statistics). Para ello vamos a modificar la función "init", que queda así:

```

view plain print ?
01. function init() {
02.     var pTable = new OAT.Pivot('tabla', '', 'filtros', cabecera, datos,
03.                               [0,1], [2,3], [4], 5, null);
04.
05.     var aggRef = function() {
06.         pivot.options.agg = parseInt($v("pivot_agg"));
07.         pivot.go();
08.     }
09.
10.     /* create agg function list */
11.     OAT.Dom.clear("pivot_agg");
12.     for (var i=0;i<OAT.Statistics.list.length;i++) {
13.         var item = OAT.Statistics.list[i];
14.         OAT.Dom.option(item.shortDesc,i, "pivot_agg");
15.         if (pivot.options.agg == i) { $("pivot_agg").selectedIndex = i; }
16.     }
17.
18.     OAT.Dom.attach("pivot_agg","change",aggRef);
19.
20.     pTable.go();
21. }

```

Vemos varios cambios en la función.

Primero, se ha definido una función, "aggRef". Esta función se encarga de repintar la tabla cada vez que se invoque, usando la función de agregación seleccionada.

Después, un bucle que se encarga de rellenar un combo (un control "select" de HTML) con los distintas funciones disponibles.

Por último, se ha añadido una llamada a la función "attach", que asocia a un evento ("change") de un control HTML ("pivot_agg") una función ("aggRef") que se llamará cuando se produzca dicho evento.

Vamos a aprovechar también para añadir algunos datos más... y el HTML resultante será:

```

view plain print ?
01. <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" >
02. <html>
03. <head>
04. <meta http-equiv="content-type" content="text/html; charset=windows-1250" >
05. <script type="text/javascript">
06.     var featureList = ["pivot","statistics"];
07. </script>
08. <script type="text/javascript" src="oat/loader.js"></script>
09.
10. <script type="text/javascript">
11.
12.     var cabecera = new Array('campo1', 'campo2', 'campo3', 'campo4', 'campo5', 'valor');
13.
14.     var datos = new Array();
15.     datos[0] = new Array('rojo', '1', 'perro', 'manzana', 'juan', '5.5');
16.     datos[1] = new Array('azul', '2', 'gato', 'pera', 'pedro', '8');
17.     datos[2] = new Array('verde', '3', 'rata', 'ciruela', 'antonio', '6');
18.     datos[3] = new Array('amarillo', '4', 'ardilla', 'mandarina', 'miguel', '4.7');
19.     datos[4] = new Array('amarillo', '4', 'ardilla', 'mandarina', 'miguel', '5.2');
20.     datos[5] = new Array('amarillo', '4', 'ardilla', 'mandarina', 'miguel', '4.9');
21.     datos[6] = new Array('amarillo', '4', 'ardilla', 'mandarina', 'miguel', '3.1');
22.
23.     function init() {
24.         var pTable = new OAT.Pivot('tabla', '', 'filtros', cabecera, datos,
25.             [0,1], [2,3], [4], 5, null);
26.
27.         var aggRef = function() {
28.             pTable.options.agg = parseInt($("#pivot_agg"));
29.             pTable.go();
30.         }
31.
32.         /* create agg function list */
33.         OAT.Dom.clear("pivot_agg");
34.         for (var i=0;i<OAT.Statistics.list.length;i++) {
35.             var item = OAT.Statistics.list[i];
36.             OAT.Dom.option(item.shortDesc,i,"pivot_agg");
37.             if (pTable.options.agg == 1) { $("#pivot_agg").selectedIndex = i; }
38.         }
39.         OAT.Dom.attach("pivot_agg","change",aggRef);
40.
41.     }
42.     pTable.go();
43. </script>
44.
45. <link href="styles/pivot.css" rel="stylesheet">
46.
47. <title></title>
48. </head>
49. <body>
50.     <select id="pivot_agg"></select>
51.     <div id="filtros"></div>
52.     <div id="tabla"></div>
53.
54. </body>
55. </html>

```

Y este es el resultado final:

valor	campo3	ardilla	gato	perro	rata	TOTAL	campo3
campo4	mandarina	pera	manzana	ciruela			campo4
campo1	campo2						
amarillo	4					4	
azul	2					1	
rojo	1					1	
verde	3					1	
TOTAL						7	
campo1	campo2						

Si cambiáis el valor del combo superior, veréis que los datos cambian según la función que seleccionéis

Conclusiones

Como hemos visto, el resultado final es bastante útil para manejar datos con múltiples variables. Combinándolo con un lenguaje de programación como PHP o Java se pueden crear conjuntos de datos a partir de lo almacenado en una base de datos...

Pero como no se puede tener todo, este componente tiene para mí un pequeño defecto... y es que trabaja con todos los datos en cliente... es decir, que si queremos manejar 10000 datos con 10 variables cada uno... imaginaos el tamaño del código que se puede transmitir entre servidor y cliente. Por otro lado, el componente de estadísticas tiene pequeños fallos con la agregación de campos vacíos...

Y hasta aquí este tutorial, en que os hemos presentado otro producto gratuito (con licencia GPL). Solo hemos visto una pequeña parte de lo que se puede hacer con OAT. Si queréis saber más, tenéis todos los medios para seguir adelante y, si preferís que sean otros los que os propongan soluciones creativas a vuestros problemas, recordad que podéis contratar a **Autentia** para que os ayude en vuestros proyectos informáticos

- Puedes opinar sobre este tutorial haciendo clic aquí.
- Puedes firmar en nuestro libro de visitas haciendo clic aquí.
- Puedes asociarte al grupo AdictosAlTrabajo 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 ...

Autentia = Soporte a Desarrollo & Formación.

info@autentia.com



Servicio de notificaciones:

Si deseas que te enviemos un correo electrónico cuando introduzcamos nuevos tutoriales.

Formulario de suscripción a novedades:

E-mail

Tutoriales recomendados

Nombre	Resumen	Fecha	Visitas	pdf
Imagen de validación de formularios con PHP	Con este tutorial vamos a ver otra forma de validación de formularios con PHP mediante la generación de una imagen aleatoria, que NO guardaremos en disco.	2006-11-20	4211	pdf
Crear RSS con PHP	En este tutorial os mostramos cómo generar un archivo RSS con PHP	2006-04-04	5012	pdf
PHP Nuke en Windows 2000	Os mostramos como instalar paso a paso Php-Nuke en vuestro entorno Windows con MySQL	2003-10-20	21006	pdf
Manejar tablas de datos con JSF	En este tutorial os mostramos un ejemplo de utilización de la extension del componente DataTable, realizada por la implementación Tomahawk de MyFaces	2006-03-09	11194	pdf
Comparador de sueldos y tablas dinámicas/pivotables	En este documento mostramos los primeros resultados del comparador de sueldos de la web y su representación en hojas de cálculo dinámicas	2008-02-05	622	pdf
Creación de ejemplo en PHP	Os enseñamos a construir en PHP el ejemplo de como mostrar la frase del día	2003-06-01	14816	pdf
Instalación y primeros pasos con PHP Nuke 7.9	En este tutorial, se indicarán los pasos para descargar, instalar, configurar y poner en funcionamiento el gestor de contenidos PHP-Nuke, y se indicarán varias webs de referencia para consultar y aumentar los conocimientos sobre él.	2007-04-02	2325	pdf
Plugin PHPeclipse en Debian	Alejandro Pérez nos muestra como instalar el plugin de Eclipse: PHPeclipse, en GNU / Linux (Debian)	2004-03-23	54652	pdf
Apache, MySQL y PHP	Os mostramos como configurar Apache, MySQL y PHP en vuestra máquina	2003-12-27	37764	pdf
Generación automática de gráficas en un web	Daniel nos muestra cómo ha desarrollado las gráficas del comparador de sueldos de la web de Adictosaltrabajo.com	2008-02-29	122	pdf

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.