

Somos su empresa de Soporte a Desarrollo Informático

Ese apoyo que siempre quiso tener

- Desarrollo de componentes y proyectos a medida.
- Auditoría de código y recomendaciones de mejora.
- Arranque de proyectos basados en nuevas tecnologías.
- Curso de Formación
 - Dirección de Proyectos Informáticos.
 - Gestión eficaz del Tiempo.
 - Arquitecturas de desarrollo Web: Web, J2EE, SOA, WebServices, BPM, etc.
 - Java/ J2EE a todos los niveles: JSPs, Servlets, EJBs, JMS, JNI, etc.
 - Análisis y diseño orientado a objeto.
 - UML y patrones de diseño.
 - Buenas prácticas en el desarrollo de aplicaciones
 - Técnicas avanzadas: Lucene, Hibernate, Spring, JSF, Struts, etc.

*Nuestra mejor referencia son los conocimientos que
compartimos en nuestro web*

www.adictosaltrabajo.com

Decenas de entidades cuentan ya con nosotros

Para más información visítenos en www.autentia.com

Tel. 91 675 33 06 - info@autentia.com

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

Inicio

Quienes somos

Tutoriales

Formación

Comparador de salarios

Comentar libro

Charlas

Más

Estas en: Inicio » Tutoriales » Maven Assembly Plugin: empaquetando aplicaciones con Maven para la ejecución de procesos batch.

Ultimas Noticias

- » Formación por calendario
- » Historia de la Informática. Monográfico. La Impresora
- » El Agile Open Spain 2009 ha terminado ¿y ahora qué?
- » "Informática profesional: Las reglas no escritas para triunfar en la empresa." Pincha para comprar..
- » Comentando el libro: "No he venido aquí a hacer amigos, Desventuras de un consultor IT"
- » Autentia patrocina el primer Agile Open Spain
- » Nuevas funcionalidades en la web de www.adictosaltrabajo.com

+Noticias Destacadas

- » "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
- » Autentia cumple 6 años

+Comentarios Cómic

+Enlaces

Catálogo de servicios Autentia (PDF 6,2MB)

En formato comic...

Tutorial desarrollado por

Jose Manuel Sánchez Suárez

Consultor tecnológico de desarrollo de proyectos informáticos. Diseñador de Adictos Al Trabajo 2.0

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: [Maven-assembly-plugin-batchProcess.pdf](#)

Acceso de usuarios registrados:

E-mail:

Contraseña:

[Deseo registrarme](#)

[He olvidado mis datos de acceso](#)

Registra tu empresa:

Descubre las ventajas de registrar tu empresa en AdictosAlTrabajo...

[Registrar mi empresa](#)

[Listado de empresas ya registradas](#)

Fecha de creación del tutorial: 2009-11-12

Maven Assembly Plugin: empaquetando aplicaciones con Maven para la ejecución de procesos batch.

0. Índice de contenidos.

- 1. Introducción.
- 2. Entorno.
- 3. Configuración del módulo batch.
- 4. El resultado del empaquetado.
- 5. Los scripts de los procesos batch.
- 6. Conclusiones.

1. Introducción

Ya hemos escrito en adictos sobre [Empaquetamiento de aplicaciones de escritorio \(standalone\) con Maven](#) y, en este tutorial, vamos a hablar sobre el mismo plugin pero orientado al empaquetamiento de un módulo de nuestra aplicación para la ejecución de procesos batch, o la ejecución de procesos puntuales, de migración de datos, que requieren tener montado todo el entorno de la capa de servicios de nuestra aplicación.

Estamos hablando de aplicaciones JEE, lo suficientemente bien modularizadas y la necesidad consiste en ejecutar una aplicación standalone, que tenga accesibles los mismos servicios que tendría el módulo web.

Para dar soporte a esos procesos batch o de migración, vamos a crear un módulo específico dentro de nuestra estructura de aplicación, al que llamaremos **app-batch**, y que, como el resto de módulos, dependerá de un módulo parent.

2. Entorno.

El tutorial está escrito usando el siguiente entorno:

- Hardware: Portátil Asus G1 (Core 2 Duo a 2.1 GHz, 2048 MB RAM, 120 GB HD).
- Sistema operativo: Windows Vista Ultimate.
- JDK 1.5.0_15
- Eclipse 3.5, con IAM.

3. Configuración del módulo batch.

He aquí el pom.xml del módulo app-batch, está comentado:

Web

www.adictosaltrabajo.com

Últimos tutoriales

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

```
view plain print ?

01. ...
02.     <parent>
03.         <groupId>com.autentia.training.maven</groupId>
04.         <artifactId>app-parent</artifactId>
05.         <version>1.0-SNAPSHOT</version>
06.     </parent>
07.
08.     <groupId>com.autentia.training.maven</groupId>
09.     <artifactId>app-batch</artifactId>
10.     <version>1.0-SNAPSHOT</version>
11.     <packaging>jar</packaging>
12.
13. ...
14.
15.     <dependencies>
16.         <!-- incluimos una dependencia del módulo core de nuestra aplicación,
17.              aquél que contiene la capa de servicios, de manera que cuando empaquetemos el módulo
18.              batch arrastre dicha dependencia, así como las transitivas del mismo -->
19.         <dependency>
20.             <groupId>com.autentia.training.maven</groupId>
21.             <artifactId>app-core</artifactId>
22.             <version>1.0-SNAPSHOT</version>
23.         </dependency>
24.         <!-- si el módulo core tiene acceso a base de datos, lo normal es que el driver esté
25.              incluido a nivel de servidor de aplicaciones, aquí estamos hablando de un proceso standalone
26.              con lo que necesitaremos incluirlo en el empaquetado, o documentarlo para que añada en función
27.              del entorno -->
28.         <dependency>
29.             <groupId>mysql</groupId>
30.             <artifactId>mysql-connector-java</artifactId>
31.             <version>5.0.5</version>
32.         </dependency>
33.     </dependencies>
34.
35.     <build>
36.         <plugins>
37.             <!-- configuración del plugin de empaquetado -->
38.             <plugin>
39.                 <artifactId>maven-assembly-plugin</artifactId>
40.                 <configuration>
41.                     <descriptors>
42.                         <!--
43.                          ruta al descriptor del empaquetado: por convención se incluye en el directorio assembly -->
44.                         <descriptor>src/main/assembly/installer.xml</descriptor>
45.                     </descriptors>
46.                     </configuration>
47.                     <!--
48.                      enganchamos la ejecución del plugin al ciclo de vida de maven, en concreto a la fase de instalación
49.                      de este modo al instalar el artefacto generará nuestro distribuible del módulo batch -
50. -->
51.                     <executions>
52.                         <execution>
53.                             <id>package-native</id>
54.                             <phase>install</phase>
55.                             <goals>
56.                                 <goal>attached</goal>
57.                             </goals>
58.                         </execution>
59.                     </executions>
60.                 </plugin>
61.             </plugins>
62.         </build>
```

A continuación el contenido del descriptor de empaquetado src/main/assembly/installer.xml:

```
view plain print ?

01. [?xml version="1.0" encoding="UTF-8" ?]    <!--
02.   cambiad los caracteres [ y ] por < y > respectivamente -->
03. <assembly>
04.     <id>installer</id>
05.     <formats>
06.         <!-- formato de salida del empaquetado -->
07.         <format>zip</format>
08.     </formats>
09.     <!--
10.      no queremos que el empaquetado tenga como raíz un directorio con el nombre del módulo batch -->
11.     <includeBaseDirectory>false</includeBaseDirectory>
12.     <!-- las dependencias del módulo (las librerías) se incluirán en un directorio /lib-->
13.     <dependencySets>
14.         <dependencySet>
15.             <outputDirectory>/lib</outputDirectory>
16.             <scope>runtime</scope>
17.         </dependencySet>
18.     </dependencySets>
19.     <!-- además el contenido del zip incluirá los siguientes directorios -->
20.     <fileSets>
21.         <!-- irá al directorio lib el contenido del directorio target del módulo batch -->
22.         <fileSet>
23.             <directory>target</directory>
24.             <outputDirectory>lib</outputDirectory>
```

Cómo conseguir que 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

2009-09-23
EJB 3.0 y pruebas unitarias con Maven, JUnit 4 y Apache Open EJB.

2009-09-21
Primeros pasos con JBoss Seam

2009-09-21
Integración con sistemas de Bug Tracking desde NetBeans 6.7

2009-09-14
EJB 3.0 y pruebas de persistencia con Maven, JUnit 4 y Embedded JBoss sobre Java 6.

2009-09-12
Instalación de Liferay en Tomcat existente

2009-09-11
Release Bugzilla Maven Plugin

2009-09-11
Enlazar Bugzilla con MavenChangesPlugin

2009-09-08
Sobre las reglas de codificación o... ¿de dónde salen esos caracteres "raros"?

2009-08-28
Cómo hacer deploy del site de Maven en SourceForge

2009-08-26
Ordenación por cantidades en informe cruzado

2009-08-20
Selenium IDE -Incorporando while en los test

```
23.         <!-- a exclusión del empaquetado del javadoc y los fuentes -->
24.         <excludes>
25.             <exclude>app-batch-*-source.jar</exclude>
26.             <exclude>app-batch-*-javadoc.jar</exclude>
27.         </excludes>
28.         <!-- se incluirá la propia librería del módulo batch -->
29.         <includes>
30.             <include>app-batch-*.jar</include>
31.         </includes>
32.     </fileSet>
33.     <!-- se creará un directorio doc en el que se incluirá el site del módulo batch -->
34.     <fileSet>
35.         <directory>target/site</directory>
36.         <outputDirectory>doc</outputDirectory>
37.         <includes>
38.             <include>**/*</include>
39.         </includes>
40.     </fileSet>
41.     <!--
42. los ficheros de configuración de la aplicación deberán estar disponibles también en el entorno del
43. módulo batch, aunque quizás no se requiera la configuración de todas sus propiedades -->
44.     <fileSet>
45.         <directory>src/main/config</directory>
46.         <outputDirectory>/config</outputDirectory>
47.         <includes>
48.             <include>**/*</include>
49.         </includes>
50.     </fileSet>
51.     <!--
52. para la ejecución de los procesos batch haremos uso de unos ejecutables que invoquen a los métodos m
53. de las clases que implementen los procesos -->
54.     <fileSet>
55.         <directory>src/main/bin</directory>
56.         <outputDirectory>/bin</outputDirectory>
57.         <lineEnding>dos</lineEnding>
58.         <includes>
59.             <include>**/*.cmd</include>
60.         </includes>
61.     </fileSet>
62.     <!--
63. al distribuir multiplataforma debemos tener en cuenta que el entorno puede ser windows o unix.
64. Los scripts para ambos serán empaquetados bajo el directorio bin. -->
65.     <fileSet>
66.         <directory>src/main/bin</directory>
67.         <outputDirectory>/bin</outputDirectory>
68.         <lineEnding>unix</lineEnding>
69.         <fileMode>0755</fileMode>
70.         <includes>
71.             <include>**/*.sh</include>
72.         </includes>
73.     </fileSet>
74.     <!-- crearemos un directorio de salida para los logs de los procesos -->
75.     <fileSet>
76.         <directory>src/main/logs</directory>
77.         <outputDirectory>/logs</outputDirectory>
78.     </fileSet>
79. </fileSets>
80. </assembly>
```

Con todo lo configurado necesitaremos mantener una estructura de directorios en el módulo batch similar a la que sigue:



Nuestra arquitectura de servicios está basada en Spring y JPA, de ahí la existencia de los ficheros de configuración de ambos frameworks.

4. El resultado del empaquetado.

Si invocamos al goal install de maven, en el directorio target del módulo batch encontraremos un fichero zip, con el nombre del módulo más el sufijo del identificador del assembly, en nuestro caso: app-batch-1.0-SNAPSHOT-installer.zip.

Dicho fichero comprimido es nuestra empaquetación del módulo batch, con todo lo necesario para ejecutar los procesos que hayamos programado. No tenemos más que descomprimir el zip en la máquina en la que queremos ejecutarlo, configurar los ficheros de entorno (acceso a la base de datos, propiedades individuales de la aplicación que ya tendremos configuradas en el módulo web,...) e invocar a los ejecutables de cada uno de los procesos.

5. Los scripts de los procesos batch.

Los ejecutables de los procesos batch simplemente invocarán a los métodos main de las clases que los implementen, pasándoles los parámetros que requieran por línea de comandos.

En este punto tenemos que resolver dos cuestiones:

- la carga dentro del classpath de ejecución de la máquina virtual de las librerías que han sido empaquetadas, para que todas estén disponibles en tiempo de ejecución,
- el paso de parámetros desde el script de ejecución de los procesos al método main de la clase que lo implementa

Estas cuestiones tienen diferentes soluciones en función del entorno.

5.1. UNIX.

El contenido del ejecutable de los procesos tendrá el siguiente código:

2009-08-14
[Blender y JMonkeyEngine. Exportación de archivos Blender y uso de los mismos en JMonkeyEngine](#)

2009-08-14
[5º tutorial TNT Concept Versión 0.16.1 Gestión de informes, vacaciones y utilidades](#)

2009-08-14
[Joomla 1.5. Instalación y configuración](#)

2009-08-13
[Introducción a los diagramas EPC \(Event-Driven Process Chain\)](#)

2009-08-10
[Blender. Animaciones avanzadas y renderización](#)

2009-08-10
[Gestión de Calidad, tablón y seguimiento en TNT Concept Versión 0.16.1](#)

2009-08-10
[Cómo hacer una página web](#)

2009-08-06
[Tips And Tricks JUnit Spring](#)

2009-08-03
[Instalación de VirtualBox PUEL](#)

2009-08-03
[Gestión de contactos y pedidos en TNT Concept versión 0.16.1](#)

2009-08-03
[Comentando el libro: La estrategia del océano azul](#)

2009-07-30
[Funciones esenciales para crear un juego.](#)

2009-07-30
[2º tutorial TNT Concept versión 1.16.1](#)

2009-07-29
[Hibernate Search, Bridges, Analizadores y más](#)

2009-07-24
[Migración de EJB3 a JPA y Spring.](#)

2009-07-20
[Directorio de ejemplos de JMonkey Engine](#)

2009-07-19
[JSR-179 Location API para J2ME: Posicionamiento geográfico en nuestras aplicaciones.](#)

```
view plain print ?

01. #!/bin/sh
02.
03. PRGDIR=`dirname "$0"`
04.
05. # -- Prepare CLASSPATH -----
06. LOCALCLASSPATH=$CLASSPATH:$PRGDIR/../config
07. for i in $PRGDIR/../lib/*.jar
08. do
09.     LOCALCLASSPATH=$LOCALCLASSPATH:"$i"
10. done
11.
12. # -- Invocation to client -----
13. java -cp $LOCALCLASSPATH com.autentia.training.maven.client.TestClient $@
```

- la primera cuestión la resolvemos con un bucle, en el que vamos añadiendo a una variable local cada uno de los jar que distribuimos en el directorio lib, una línea más arriba ya hemos incluido el contenido del directorio config. Con -cp añadimos a la ejecución de la máquina virtual todo lo incluido en la variable local.
- con \$@ resolvemos la cuestión del paso de parámetros del script a la clase.

5.2. Windows.

El contenido del ejecutable de los procesos tendrá el siguiente código:

```
view plain print ?

01. @echo off
02. setlocal
03.
04. rem -- Check existence of JVM -----
05. java -version >nul 2>1
06. if errorlevel 1 goto :NO_JAVA
07.
08. rem -- Prepare CLASSPATH -----
09. call "%~dp0cp\atcp.cmd" %~dp0..\config
10. for %1 in ("%~dp0..\lib\*.jar") do call "%~dp0cp\atcp.cmd" %1
11.
12. rem -- Invocation to client -----
13. echo.
14. java -Xms256m -Xmx512m -XX:MaxPermSize=128m -
15. cp "%CLASSPATH%"; com.autentia.training.maven.client.TestClient %*
16. goto :END
17.
18. :NO_JAVA
19. echo Java Virtual Machine not found, please add JRE's bin directory to PATH environment variable
20. goto :END
21.
22. :END
endlocal
```

- para resolver la primera cuestión volvemos a necesitar un bucle, pero ahora nos tenemos que apoyar en un segundo script **atcp.cmd** que vaya concatenando a la variable %CLASSPATH% una a una, la librería que itera. Con -cp añadimos a la ejecución de la máquina virtual todo lo incluido en la variable %CLASSPATH%.
- con %* resolvemos la cuestión del paso de parámetros del script a la clase.

El contenido del script atcp.cmd, es el siguiente:

```
view plain print ?

01. @echo off
02.
03. set ARG=%1
04. shift
05.
06. :APPEND
07. set ARG=%ARG% %1
08. shift
09. if not "%1"==" " goto :APPEND
10.
11. @set CLASSPATH=%CLASSPATH%;%ARG%
```

6. Conclusiones.

Seguimos haciendo uso extensivo de maven para la gestión del ciclo de vida de nuestros proyectos y, aunque la primera intención pueda ser la de realizar este tipo de tareas manualmente, puesto que quizás sean puntuales o pensemos que total, seguro que tardo menos,... la experiencia nos dice que hacer las cosas de una manera controlada, a la larga, es mucho más beneficioso.

Tened en cuenta que si lo hacemos bien, nuestros procesos tendrían que estar documentados en el site, y eso se distribuye también en el zip, con lo que tendríamos cubiertas nuestras necesidades a la hora de ejecutarlos.

En cuanto a los ejecutables, más simple en Unix que en Windows... :-D, pero en ambos entornos "estamos abiertos a extensiones y cerrados a modificaciones" puesto que la inclusión de una librería a nuestra arquitectura o un parámetro adicional a nuestros procesos no implicarán modificación alguna en los scripts.

Un saludo.

Jose

<mailto:jmsanchez@autentia.com>

2009-07-16
[Gestión de Usuarios en TNT Concept versión 0.16.1](#)

2009-07-16
[Continuación del Tutorial: JMonkeyEngine, Creación de nuestro primer juego.](#)

2009-07-16
[Como implementar el Scene Monitor para analizar las escenas en jMonkeyEngine](#)

2009-02-26
[Transformaciones de escena en JMonkeyEngine](#)

2009-07-15
[Detalles del juego de la moto en jMonekyEngine.](#)

2009-07-14
[JMonkeyEngine, Creación de nuestro primer juego.](#)

2009-07-13
[Ajax tests con Selenium: prototype.js e ICEfaces.](#)

2009-07-08
[AOP con AspectJ y Maven](#)

2009-07-07
[Instalación y configuración de Eclipse Galileo](#)

2009-07-07
[Iniciarse en el manejo de JME, Creación de un Cloth.](#)

2009-07-06
[Primeros pasos con Blender: Pintando nuestra mascota en 3D](#)

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](#)
[Java ORM](#)

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



(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.

(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 AdictosAlTrabajo en XING [haciendo clic aquí](#).
- Añadir a favoritos Technorati.



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

Tutoriales recomendados

Nombre	Resumen	Fecha	Visitas	Valoración	Votos	Pdf
Maven Assembly Plugin: empaquetando aplicaciones con Maven para la ejecución de procesos batch.	Examinamos cómo llevar a cabo el empaquetado de nuestras utilidades batch, basadas en una capa de servicios y cómo hacer la distribución independiente del entorno.	2009-11-12	15	-	-	

Cómo conseguir que Subversion avise a Hudson para lanzar una build	En este tutorial vamos a ver como configurar Subversion para que sea este el que avise a Hudson cada vez que hay un commit, y así se lance la build.	2009-10-27	503	-	-	
Cómo instalar Hudson en Apache Tomcat	Instalar Hudson en Apache Tomcat	2009-10-26	696	-	-	
EJB 3.0 y pruebas unitarias con Maven, JUnit 4 y Apache Open EJB.	Continuamos buscando una buena solución para llevar a cabo test de EJBs, ahora con OpenEJB.	2009-09-23	1102	-	-	
EJB 3.0 y pruebas de persistencia con Maven, JUnit 4 y Embedded JBoss sobre Java 6.	Damos continuidad al tutorial EJB 3.0 y pruebas unitarias con Maven, JUnit 4 y Embedded JBoss Java 6, probando el llevar a cabo un test de persistencia de un EJB de entidad, bajo soporte de EJB de servicio que implementa el patrón dao, en el mismo entorno	2009-09-14	1361	Muy bueno	1	
Release Bugzilla Maven Plugin	En este tutorial vamos a mostrar como automatizar un conjunto de acciones que hay que hacer siempre en los sistemas de gestión de incidencias, tales como dar de alta una nueva versión del producto, cerrar las incidencias que solucionan la nueva versión, et	2009-09-11	762	-	-	
Enlazar Bugzilla con MavenChangesPlugin	En este tutorial veremos como enlazar Bugzilla con MavenChangesPlugin	2009-09-11	628	-	-	
Sobre las reglas de codificación o... ¿de dónde salen esos caracteres "raros"?	En este tutorial vamos a tratar de dar algo de luz a los errores que tenemos habitualmente con la codificación de caracteres en aplicaciones en las que se ven implicados varios sistemas que intercambian o almacenan información.	2009-09-08	2219	Muy bueno	1	
Cómo hacer deploy del site de Maven en SourceForge	Este tutorial nos enseña un poco mas sobre Maven	2009-08-28	1145	-	-	
AOP con AspectJ y Maven	Programacion orientada a aspectos con AspectJ y Maven	2009-07-08	1818	-	-	

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.

- Anuncios Google
- Taller Informatico
- Tiendas Informatica
- Cursos Informática
- Lotus Notes Tutorial
- Manuales Anaya