

Tutorial para la creación de un JAR y posteriormente convertirlo en un Archivo .EXE



Domínguez Geniz Amalio Javier
Todos los Derechos Reservados. 2009
Tecnológico de Estudios Superiores de Chalco

ajdgeniz@hotmail.com

Gracias por Descargar otro de mis manuales publicados en mygnet, en el presente manual mostraré como hacer un archivo jar y después convertirlo a un archivo ejecutable en Windows con extensión .exe.

Empezaremos creando una aplicación de consola y después una con interfaz gráfica, a manera de ejemplo escribiremos el clásico Hola Mundo, para iniciar abra el bloc de notas de Windows o cualquier otro editor de texto como el edit de MS-DOS y escriba el siguiente código fuente:

```
class HolaMundo{  
  
public static void main(String []args) {  
System.out.println("Hola Mundo");  
}  
  
}
```

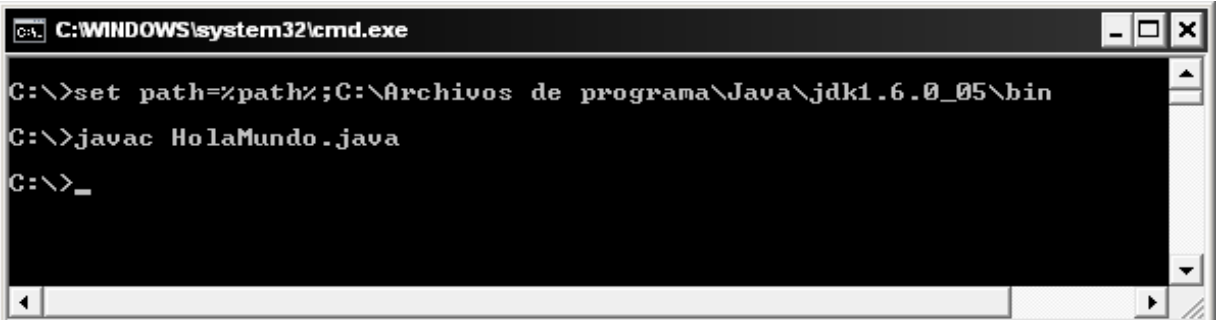
A continuacion guarde este archivo con el nombre `HolaMundo.java` en la carpeta que guste, el siguiente paso es compilarlo, para ello abra el simbolo del sistema y establezca la ruta de donde se encuentra el JDK en una variable path, por ejemplo:

set path=%path%;C:\Archivos de programa\Java\jdk1.6.0_05\bin



```
C:\WINDOWS\system32\cmd.exe  
C:\>set path=%path%;C:\Archivos de programa\Java\jdk1.6.0_05\bin  
C:\>_
```

Una vez establecida la ruta con ayuda del comando CD diríjase a donde guardo el archivo y escriba el comando **javac HolaMundo.java**, reemplace el archivo `HolaMundo.java` por el nombre de su archivo y recuerde que el compilador distingue letras mayúsculas y minúsculas:

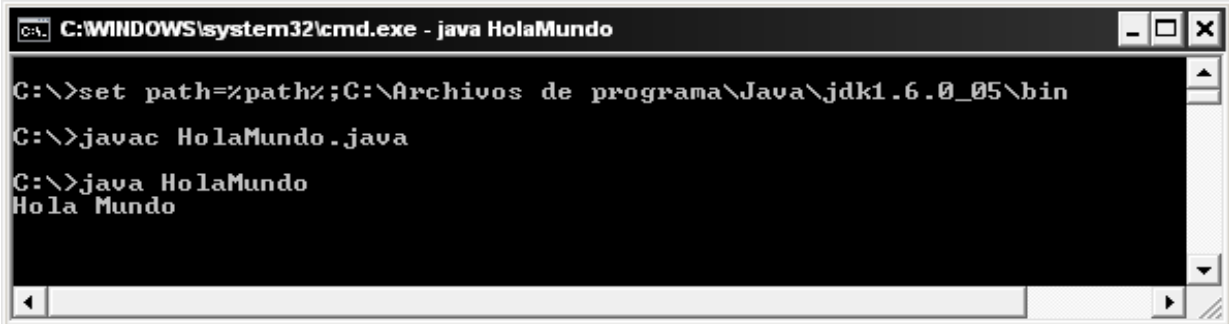


```
C:\WINDOWS\system32\cmd.exe  
C:\>set path=%path%;C:\Archivos de programa\Java\jdk1.6.0_05\bin  
C:\>javac HolaMundo.java  
C:\>_
```

Si nuestra aplicación no genere errores entonces obtendrá un archivo con extensiones .CLASS que son los bytecodes que serán ejecutados, escriba el siguiente comando para ver su aplicación:

Java HolaMundo

Y Recuerde no colocar la extensión java ni class



```
C:\WINDOWS\system32\cmd.exe - java HolaMundo
C:\>set path=%path%;C:\Archivos de programa\Java\jdk1.6.0_05\bin
C:\>javac HolaMundo.java
C:\>java HolaMundo
Hola Mundo
```

Pues bien nuestro programa esta ahora listo para ser empaquetado en un fichero jar (Java Archive) que es básicamente un fichero comprimido en ZIP, para ello utilizaremos el comando jar con ciertos parámetros que nos permitirán agregar archivos, ejemplo:

jar cvfm myjar.jar manifest.mf HolaMundo.class otras_carpetas



```
C:\WINDOWS\system32\cmd.exe
Hola Mundo
C:\>jar cvfm myjar.jar manifest.mf HolaMundo.class otras_carpetas
```

Donde:

- c:** Indica q se va a crear un archivo jar
- v:** Genera la salida
- f:** Especificar el nombre del archivo
- m:** incluir un archivo manifest.mf

y:

myjar.jar: Nombre de nuestro archivo con extensión.jar

manifest.mf: Archivo manifiesto con extensión mf

HolaMundo.class: Archivos compilados, puede agregar mas de uno separados por un espacio: archivo1.class archivo2.class archivon.class

Otras_carpetas: indica que puede agregar carpetas opcionales

Una vez sabiendo todo esto solo nos falta crear el archivo manifest.mf que contiene parámetros necesarios para la ejecución de la aplicación, a continuación veremos como hacer un sencillo archivo manifest:

```
Manifest-Version: 1.0
Created-By: Geniz
Main-Class: HolaMundo
Class-Path: lib/substance.jar lib/sqljdbc.jar lib/mysql-connector-java-5.0.4bin.jar
X-COMMENT: Main-Class will be added automatically by build
```

Donde:

Manifest-Version: Indica la versión del archivo

Created-By: Autor del archivo

Main-Class: Especifica la clase principal que será la primera en ejecutarse


Class-Path: Indica librerías extras que ocupara la aplicación, por ejemplo conectores JDBC par alas bases de datos o librerías de terceros.

X-COMMENT: Aquí podrá colocar comentarios extras.

Cree su archivo manifiesto y guárdelo como manifest.mf, recuerde colocarlo en la misma carpeta donde se encuentren el resto de los archivos, no será necesario agregar todos los parámetros solo bastará con los siguientes:

```
Manifest-Version: 1.0
Created-By: Geniz
Main-Class: HolaMundo
X-COMMENT: Este es mi primer archivo Manifiesto
```

Una vez que todos nuestros archivos están en el mismo directorio teclee el comando en la consola del símbolo del sistema:



```
C:\WINDOWS\system32\cmd.exe

C:\>jar cvfm myjar.jar manifest.mf HolaMundo.class
manifest agregado
agregando: HolaMundo.class <entrada = 568> <salida = 356> <desinflado 37%>
C:\>
```

Para ejecutarlo escriba: **java -jar myjar.jar**

A screenshot of a Windows command prompt window. The title bar reads "C:\WINDOWS\system32\cmd.exe - java -jar myjar.jar". The command prompt shows the following text:

```
C:\>jar cvfm myjar.jar manifest.mf HolaMundo.class
manifest agregado
agregando: HolaMundo.class <entrada = 568> <salida = 356> <desinflado 37%>

C:\>java -jar myjar.jar
Hola Mundo
-
```

Diríjase (ya no desde símbolo del sistema) al directorio donde se encuentre su archivo jar y haga doble clic sobre el, notará que no pasa nada eso se debe a que nuestra aplicación es una consola sin interfaz grafica, para ejecutarlo crearemos un archivo de ejecución por lotes (CMD), de manera muy sencilla abra el editor de texto de sus preferencia y escriba la orden que utilizo para ejecutar el jar desde el símbolo del sistema:

```
java -jar myjar.jar
```

Guárdelo con el nombre que desee y póngale la extensión .CMD, listo ahora haga doble clic y se mostrará nuestro primer programa en ejecución:

NOTA: Si la pantalla aparece y desaparece demasiado rápido añada el siguiente código, esto la detendrá y se cerrara hasta que Ud. Presione una tecla.

```
class HolaMundo{
public static void main(String []args) throws Exception
{
System.out.println("Hola Mundo");
    System.in.read();
}
}
```

Otra forma, ponga este codigo debajo de System.out.println:

```
try{
System.in.read();
}
catch (Exception e){}
```

Ahora utilizando los mismos comandos pero con una aplicación grafica crearemos un archivo jar y en seguida lo convertiremos a uno con estension .EXE, a manera de ejemplo utilizaremos otro clasico que es el convertidor de numeros decimales a binarios:

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
public class binario extends JFrame implements ActionListener {
private JLabel eti1, eti2; JTextField num, bin; JButton calcular;

public binario() {
super("Conversion a numeros Binarios");
Container contenedor=getContentPane();
contenedor.setLayout(new FlowLayout());
contenedor.setBackground(Color.green);

eti1=new JLabel("Escriba numero a convertir");
contenedor.add(eti1);
num=new JTextField(10);
contenedor.add(num);
eti2=new JLabel("En Binario es:");
contenedor.add(eti2);
bin=new JTextField(10);
contenedor.add(bin);
calcular=new JButton("convertir");
contenedor.add(calcular);
calcular.addActionListener(this);

setSize(200,300);
setVisible(true);
}
public static void main(String args[])
{
binario aplicacion=new binario();
aplicacion.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
}
int num1=0;
int a=0;
String num2="";

public void actionPerformed(ActionEvent evento)
{
num1=Integer.parseInt(num.getText());
if(evento.getSource()==calcular){
while(num1!=0){
a=num1 % 2;
num1=num1/2;
num2=a+num2;
bin.setText(num2);
}}}}}
```

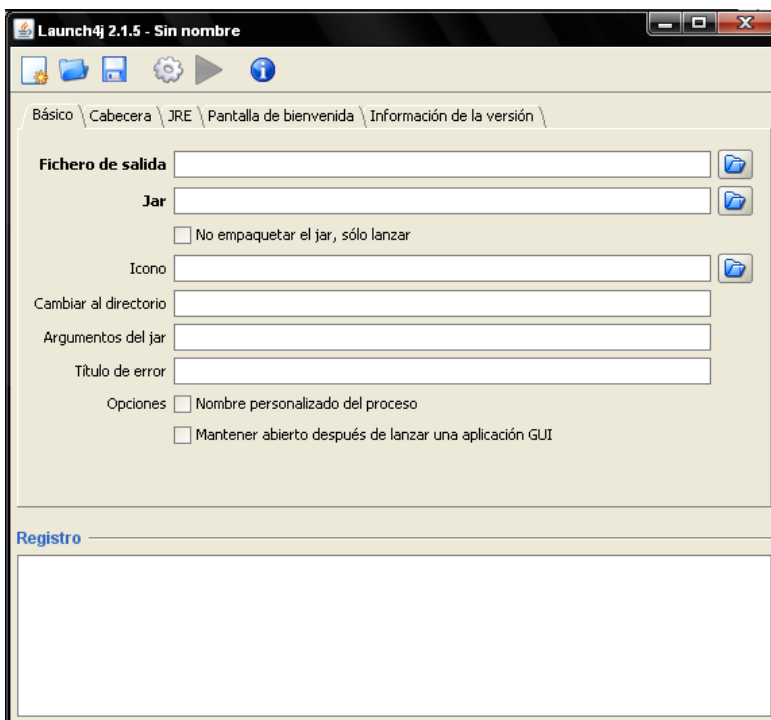
Siga los procesos ya antes descritos para compilar y crear el archivo jar



Ahora busque el archivo donde lo guardo y haga doble clic sobre el, notara que nuestra aplicación se ejecuta directamente, ya no es necesario crear un archivo por lotes (CMD), ahora crearemos nuestro archivo exe a partir del jar antes creado.

CREANDO EL ARCHIVO EXE

Para crear un archivo necesitaremos un software (Open Source) escrito en java llamado Launch4j el cual podemos descargar desde el sitio: <http://launch4j.sourceforge.net>, lo encontrara comprimido en formato zip, solo descomprimalo, y ejecute el archivo launch4j.exe y se abrirá la siguiente pantalla:



Ahora haga clic sobre la **ficha basico** y escriba los siguientes parametros:

Fichero de salida 

→ Coloque la ruta de salida y el nombre del archivo, con la extension.EXE

Jar 

→ Aquí debera colocar la localizacion del archivo jar

Icono 

→ Coloque la ruta y el nombre de un icono para su aplicación.

Registro

Compilando recursos
Enlazando
Empaquetando
Creado con éxito C:\Binario.exe

→ Muestra los registros de si se a creado el archivo o si ha fallado.

Esto es todo lo que debera escribir en la ficha basico, las opciones no mencionadas pueden ser dejadas en blanco en esta y en el resto de las demas fichas.

NOTA: NO sera necesario agregar nada en la **ficha cabecera**.


Ficha JRE

En esta ficha configurará la maquina virtual de java que sera la encargada de ejecutar nuestras aplicaciones, aunque estas tenga la extension exe, sera necesario contar con una maquina virtual de java para ejecutarla, en la siguiente pantalla especificara la localizacion de esta y la version minima y maxima con la que debera contar, tenga cuidado al escribir la ruta del JRE como la del ejemplo, buscandola siempre dentro del JDK.

| | |
|-------------------------------|---|
| Ruta del JRE | <input type="text" value="C:\Archivos de programa\Java\jdk1.6.0_05\jre"/> |
| Versión mínima del JRE | <input type="text" value="1.5.0"/> |
| Versión máxima del JRE | <input type="text" value="1.6.0"/> |

Ficha Pantalla de Bienvenida



Solo nos permite mostrar una imagen BMP al inicializarse la aplicación, es solo una pantalla de presentacion.

| | | |
|-------------------------------------|---|---|
| <input checked="" type="checkbox"/> | Activar pantalla de bienvenida | |
| Imagen | <input type="text" value="F:\Mis documentos\Mis imágenes\ENTRADA.bmp"/> |  |
| Esperar la ventana | <input checked="" type="checkbox"/> | Cerrar la pantalla de bienvenida cuando aparezca una ventana de la aplicación |
| Tiempo de espera [s] | <input type="text" value="60"/> | |
| <input checked="" type="checkbox"/> | Señal de error asociada al tiempo de espera | |

Ficha Informacion de la Versión

Estos parametros son opcionales y solo serviran para ir versionando nuestras aplicaciones , no es necesaria.

| | | |
|-------------------------------------|---|--|
| <input checked="" type="checkbox"/> | Añade información sobre la versión | |
| Versión del fichero | <input type="text" value="1.0.0.0"/> | Forma libre <input type="text" value="1.0.0.0"/> |
| Descripción del fichero | <input type="text" value="Numeros Binarios"/> | |
| Copyright | <input type="text" value="Geniz 2009"/> | |
| Información adicional | | |
| Versión del producto | <input type="text" value="1.0.0.0"/> | Forma libre <input type="text" value="1.0.0.0"/> |
| Nombre del producto | <input type="text" value="Binarios"/> | |
| Nombre de la organización | <input type="text" value="Geniz SWF"/> | |
| Nombre interno | <input type="text" value="Binarios"/> | |
| Nombre original del fichero | <input type="text" value="binario.exe"/> | |

Una vez que ha escrito todos los parametros correspondientes, haga clic en el boton “Construir el empaquetador”  y vea la pantalla de registro para ver si ocurrio error alguno, de lo contrario haga clic en “comprobar el empaquetador”  y podrá ver en ejecucion a su aplicación nativa de windows (EXE), ahora solo busque el archivo exe donde lo alojo y ahora ya podra utilizarlo.

Listo!!! ahora ya sabe como constriuir archivos JAR y despues transformarlos en EXE, en la siguiente edicion de este manual mostrare como hacer a partir del exe, crearemos un archivo instalador.

NOTA: Podra guardar todas sus configuraciones en un archivo XML , solo haga clic en el boton: 