



Eusko Jaurlaritzaren Informatika Elkarte
Sociedad Informática del Gobierno Vasco

DJ Java Decompilar:

Manual de usuario

Fecha: 28/09/2007

Referencia:

EJIE S.A.
Mediterráneo, 3
Tel. 945 01 73 00*
Fax. 945 01 73 01
01010 Vitoria-Gasteiz
Posta-kutxatila / Apartado: 809
01080 Vitoria-Gasteiz
www.ejie.es

Control de documentación

Título de documento: DJ JAVA DECOMPILER

Histórico de versiones

Código:

Versión: 1.0

Fecha: 28/09/2007

Resumen de cambios:

Cambios producidos desde la última versión

Primera versión.

Control de difusión

Responsable: Ander Martínez

Aprobado por: Ander Martínez

Firma:

Fecha:

Distribución:

Referencias de archivo

Autor: Consultoría de áreas de conocimiento

Nombre archivo: DJ Java Decompiler. Manual de usuario v1.0.doc

Localización:

Contenido

	Capítulo/sección	Página
1	Introducción	4
2	Conceptos básicos	4
3	Funciones elementales	4
3.1	Inicio de la aplicación	4
3.2	Operativa básica	4
4	Funciones avanzadas	6
4.1	Barras de herramientas	6
4.2	Barra de menús	6
5	Utilidad practica	15
6	Anexo 1: Decompilar un archivo .class	15

1 Introducción

El presente documento describe cuáles son las tareas básicas que se pueden ejecutar en la explotación de la herramienta DJ Java Decompiler.

2 Conceptos básicos

La aplicación DJ Java Decompiler es un decompilador para clases compiladas en Java, esto es, dado un fichero compilado binario .class es capaz de obtener el código fuente que lo generó.

DJ Java Decompiler es una herramienta “freeware” que se instala como una aplicación Windows stand-alone: no requiere una máquina virtual Java instalada.

Para obtener información adicional sobre el producto acceder a su página web:

<http://members.fortunecity.com/neshkov/dj.html#>

3 Funciones elementales

3.1 Inicio de la aplicación

La aplicación puede iniciarse de dos formas distintas:

1. Ejecutando el acceso directo existente en el escritorio (Inicio→ DJ Java Decompiler 3.7)
2. Haciendo doble clic sobre el fichero a descompilar (.class), se nos mostrará el programa con el fichero descompilado directamente, puesto que el instalador habrá realizado la asignación correspondiente.

3.2 Operativa básica

Dada su sencillez no resultará en ningún momento complicado su uso ya que posee una interfaz con botones de acceso directo a las diferentes funcionalidades que se ofrecen.

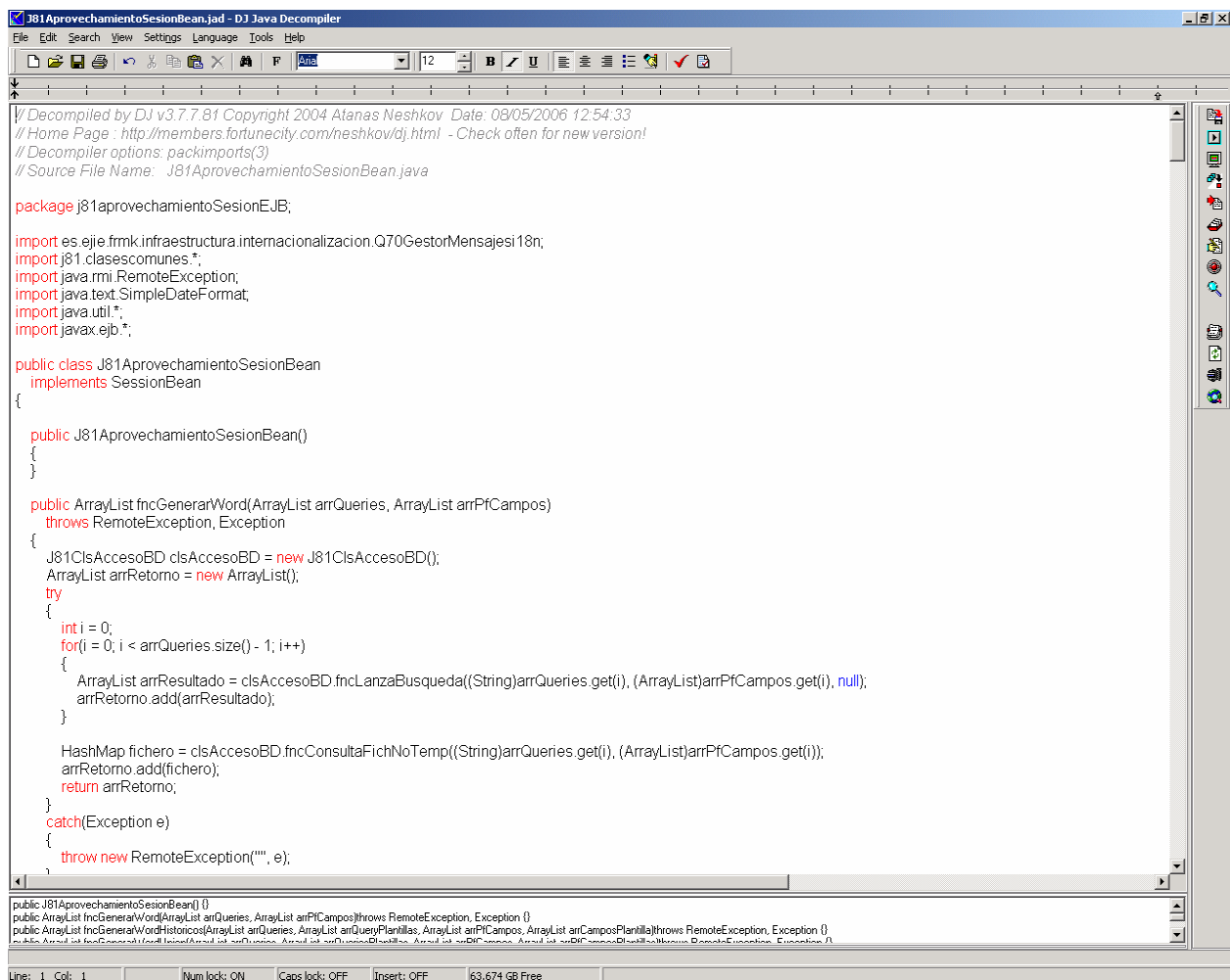
Al tratarse de un editor de texto, puede utilizarse para abrir otros tipos de ficheros cuyo contenido sea texto plano, resultando una herramienta multidisciplinar.

Para descompilar un archivo compilado binario .class tan solo debemos dirigirnos a el menú File→Open, y elegir el .class que queramos descompilar; la aplicación lo procesará y nos devolverá por pantalla el código fuente correspondiente a lo compilado. Hay que tener en cuenta que si a la hora de descompilar un fichero, a éste se le ofuscó el código fuente o se le aplicaron técnicas anti-decompilacion es posible que el código fuente que obtengamos sea prácticamente ilegible.

Una vez con el código fuente generado cargado en el editor, la aplicación funciona como un editor de texto, con todas las capacidades más comunes de este tipo de aplicaciones, con algunos añadidos en relación con los archivos de código fuente Java que se pueden encontrar en el menú View.

En el menú Settings se pueden configurar ciertos aspectos del programa, así como del motor de descompilación, para que nos genere más o menos información al descompilar.

Ya por ultimo en el menú Tools disponemos de utilidades para compilar el archivo de código fuente cargado en el editor, gestionar paquetes JAR, Applets, etc.; en este menú se incluye una capacidad de gran utilidad, como es la posibilidad de descompilar varios ficheros .class de java a la vez, para cuando deseemos descompilar un proyecto entero con cientos de ficheros de una sola vez.



```

J81AprovechamientoSesionBean.jad - DJ Java Decompiler
File Edit Search View Settings Language Tools Help
// Decompiled by DJ v3.7.7.81 Copyright 2004 Atanas Neshkov Date: 08/05/2006 12:54:33
// Home Page : http://members.fortunecity.com/neshkov/dj.html - Check often for new version!
// Decompiler options: packimports(3)
// Source File Name: J81AprovechamientoSesionBean.java

package j81aprovechamientoSesionEJB;

import es.ejje.frmk.infraestructura.internacionalizacion.Q70GestorMensajesi18n;
import j81.clasescomunes.*;
import java.rmi.RemoteException;
import java.text.SimpleDateFormat;
import java.util.*;
import javax.ejb.*;

public class J81AprovechamientoSesionBean
    implements SessionBean
{
    public J81AprovechamientoSesionBean()
    {
    }

    public ArrayList fncGenerarWord(ArrayList arrQueries, ArrayList arrPfcCampos)
        throws RemoteException, Exception
    {
        J81ClsAccesoBD clsAccesoBD = new J81ClsAccesoBD();
        ArrayList arrRetorno = new ArrayList();
        try
        {
            int i = 0;
            for(i = 0; i < arrQueries.size() - 1; i++)
            {
                ArrayList arrResultado = clsAccesoBD.fncLanzaBusqueda((String)arrQueries.get(i), (ArrayList)arrPfcCampos.get(i), null);
                arrRetorno.add(arrResultado);
            }

            HashMap fichero = clsAccesoBD.fncConsultaFichNoTemp((String)arrQueries.get(i), (ArrayList)arrPfcCampos.get(i));
            arrRetorno.add(fichero);
            return arrRetorno;
        }
        catch(Exception e)
        {
            throw new RemoteException("", e);
        }
    }
}

public J81AprovechamientoSesionBean() {}
public ArrayList fncGenerarWord(ArrayList arrQueries, ArrayList arrPfcCampos) throws RemoteException, Exception {}
public ArrayList fncGenerarWordHistoricos(ArrayList arrQueries, ArrayList arrQueryPlantillas, ArrayList arrPfcCampos, ArrayList arrCamposPlantilla) throws RemoteException, Exception {}
public ArrayList fncGenerarWordHistoricos(ArrayList arrQueries, ArrayList arrQueryPlantillas, ArrayList arrPfcCampos, ArrayList arrCamposPlantilla, ArrayList arrCamposPlantilla) throws RemoteException, Exception {}

Line: 1 Col: 1 Num lock: ON Caps lock: OFF Insert: OFF 63,674 GB Free
  
```

4 Funciones avanzadas

4.1 Barras de herramientas

La barra de herramientas horizontal nos dará la posibilidad de realizar las instrucciones básicas como la gran mayoría de programas. En la imagen podremos apreciar las funciones a realizar.



Desde esta barra podremos crear un archivo nuevo, abrir uno existente, guardarlo e imprimirlo. Después se encuentran las opciones de copiar, pegar y eliminar texto.

A continuación, se encuentra el icono de búsqueda, para encontrar la palabra deseada en el código fuente. Seguido encontramos todas las opciones de fuente disponibles, pero para poder modificarlas debemos desactivar la opción **Syntax Colorizing**, último elemento de la barra. Una vez desactivada esta opción podremos cambiar la fuente y el color, pero el programa no nos ayudará coloreando las letras de diferentes categorías.

La barra de herramientas vertical dispone de opciones más específicas de la aplicación.



- **Compile:** Nos abrirá otra ventana con la que nos dará las opciones pertinentes para compilar un archivo .class.
- **Run:** Lanzará el archivo .class seleccionado. Habrá que añadirle las opciones necesarias.
- **Appletviewer:** Un applet es un código JAVA que carece de un método main, por eso se utiliza principalmente para el trabajo de páginas Web, ya que es un pequeño programa que es utilizado en una página HTML y representado por una pequeña pantalla gráfica dentro de ésta. Permite su visualización.
- **JAR:** Generará un archivo .JAR basándose en el código fuente seleccionado.
- **Native-to-ASCII Converter:** Permitirá convertir un archivo con código nativo a código unicode.
- **Applet Tag Editor:** Permitirá crear applets rápidamente y podrá corregir archivos HTML desde esta herramienta.
- **Find target:** Nos enviara a la carpeta contenedora del archivo .class.
- **Find files or folders:** Nos enviara directamente a la página de búsqueda de archivos de Windows.
- **Toggle to Bytecode View:** Aparecerá el bytecode del archivo .class.
- **Refresh:** Actualizara la página mostrada.

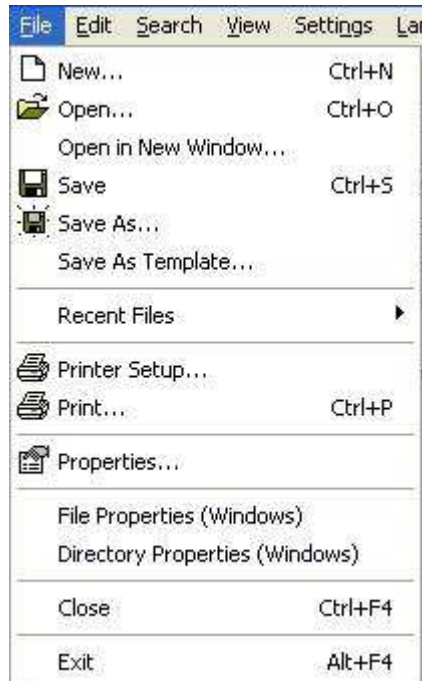
4.2 Barra de menús

La barra de menús presenta el siguiente aspecto:

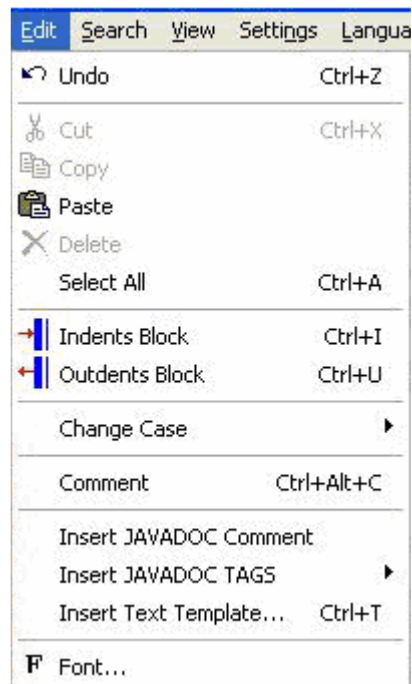


La primera sección que encontraremos será el menú **File**. En ella encontraremos todas las opciones más

comunes de un programa, como son: Nuevo, abrir, guardar, imprimir e incluso podremos ver las propiedades del archivo .class que esté abierto en ese momento. Esta última opción la activaremos presionando sobre **File Properties** y presionando sobre **Directory Properties**. Aparecerán las propiedades de la carpeta donde se encuentre el archivo .class que estemos utilizando. En la imagen podremos observar todas las opciones.



El menú **Edit** mostrará todas las opciones de edición como pueden ser: Deshacer, cortar, pegar y cambiar las opciones de la fuente. También podremos añadirle o restarle una sangría para organizar el documento a gusto del usuario. Y desde este mismo menú podremos programar la aplicación para que todas las letras las escriba, o bien, todas en mayúsculas o todas en minúsculas, incluso solo la primera letra de cada palabra.



El menú **Search** nos dará todas las facilidades posibles para encontrar una palabra o una orden en concreto y poder incluso reemplazarla, también podremos buscar una lista expresamente o encontrar el cierre y la apertura de la llave.



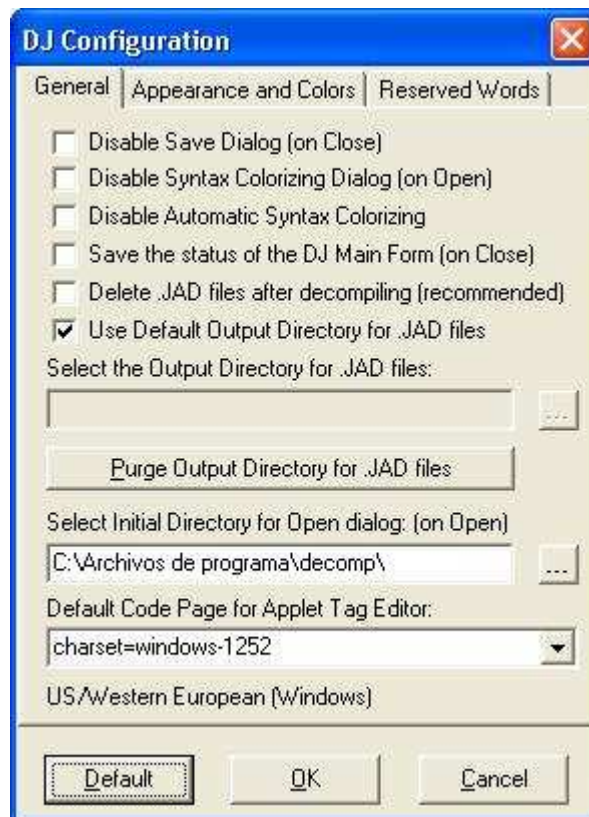
El menú **View** nos mostrará opciones tales como actualizar el archivo, verlo en bytecode, tendremos la opción de visualizar todos los archivos .class abiertos hasta el momento. Podremos ver una lista con los paquetes y clases diferentes. Visualizaremos los modificadores de acceso que se usan en la clase y cuáles están activados, incluso podremos ver los diferentes campos del documento.



El menú **Settings** nos mostrará las propiedades de nuestro propio equipo. También podremos ocultar y mostrar el reloj y desactivar o activar la opción de colorear la fuente. También se podrá cambiar la configuración de los dos menús (superior y derecho) añadiendo o quitando las opciones deseadas.

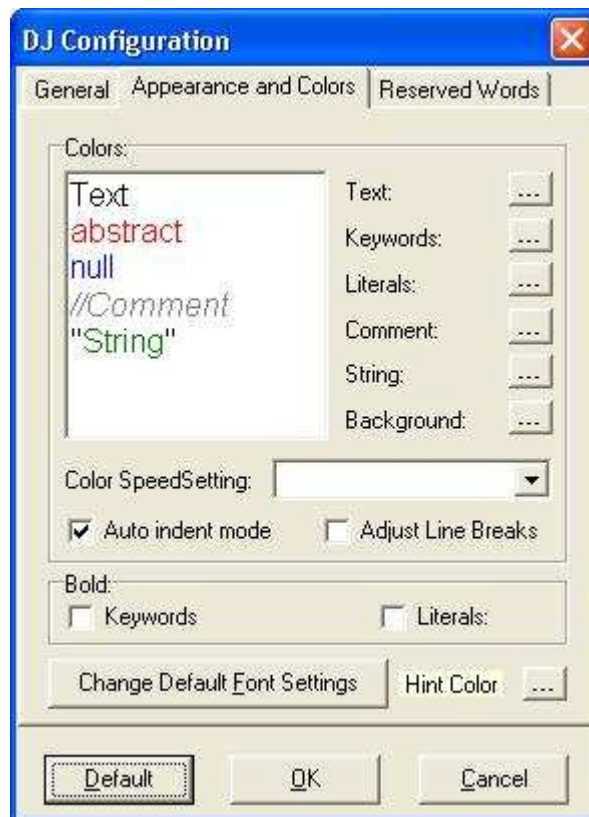


Cuando presionemos sobre **Configuration**, nos aparecerá la siguiente ventana con todas las opciones disponibles en el programa, separadas en 3 pestañas.

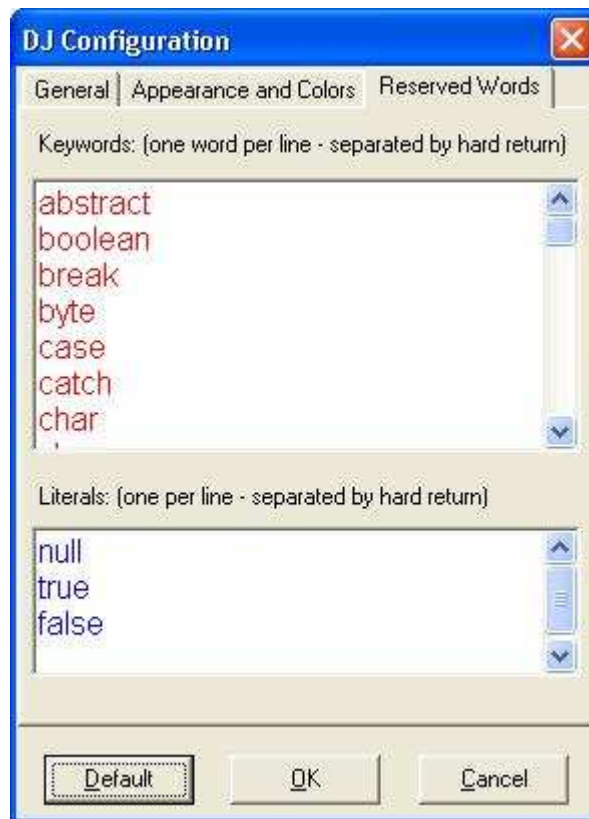


Esta primera pestaña nos mostrará todas las opciones que podremos seleccionar al iniciar y finalizar la aplicación. Podremos insertar incluso un directorio que se abrirá cuando se inicie la aplicación.

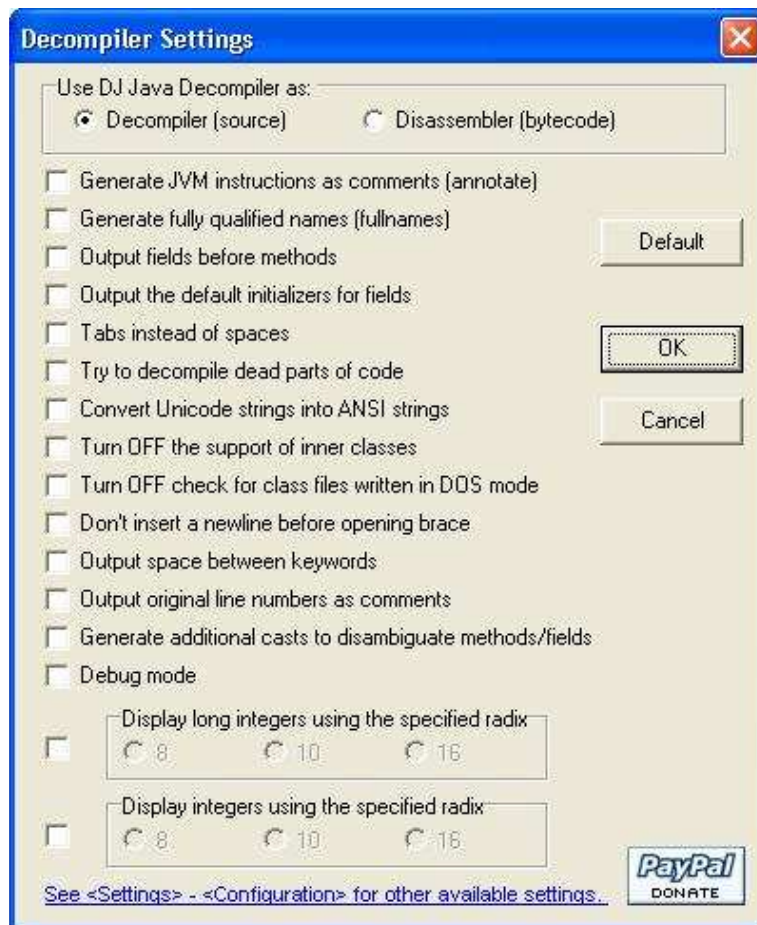
La segunda pestaña muestra las opciones de coloreado de la fuente, con diferentes opciones que se podrán modificar a gusto del usuario.



En esta última pestaña configuraremos palabras concretas para que cojan el color elegido.



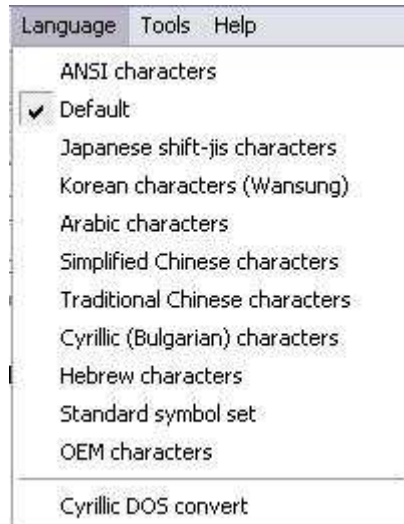
En el menú **Settings**, aparece la opción **Decompiler Settings**. Una vez se haga clic en esta opción, aparecerá la siguiente ventana en nuestra aplicación.



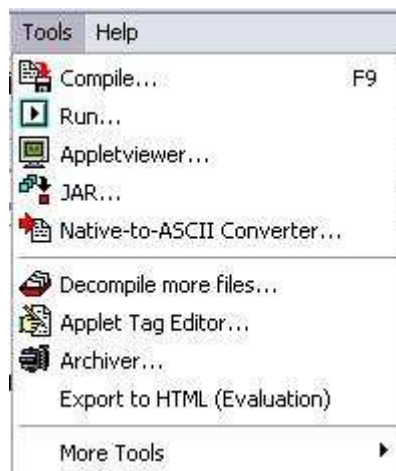
Esta ventana nos dará a elegir el modo en que queremos que actúe la aplicación, pudiendo elegir entre Decompilador (fuente) o Desensamblador (bytecode) . Ahora proseguiremos con la lista mostrada en esta ventana, observando todas las posibilidades.

- Genera instrucciones de la maquina virtual de Java como si fueran comentarios.
- Genera los nombres completamente cualificados (Nombre completo).
- Campos de salida antes de los métodos.
- Hará salir los iniciadores por defecto asignados a los campos.
- Pestañas en vez de espacios.
- Intentará decompilar las partes muertas de código.
- Convertir una secuencia Unicote en secuencia ANSI.
- Desactivar la ayuda en las clases internas.
- Desactivar la comprobación de las clases en modo DOS.
- No insertar una nueva línea antes de abrir el apoyo.
- Espacio de salida entre las palabras clave.
- Poner el número de líneas de salida como comentarios.
- Genera moldes adicionales para quitar ambigüedades de los métodos o en los campos.
- Modo de eliminación de errores.
- Mostrara los números enteros usando la raíz especificada.
- Mostrara los números usando la raíz especificada.

El menú **Language** nos permitirá cambiar el idioma.



La siguiente opción llamada **Tools** muestra todas las opciones expuestas en el menú derecho e incluye alguna más, pero para ello, deberíamos registrarnos.



El menú **Help** incluirá todas las ayudas proporcionadas por la propia aplicación.



Por ultimo comentar en la pantalla principal, en la parte inferior, una vez abierto un archivo .class, aparecerá un índice con todas las ordenes escritas. En la imagen podremos apreciarlo mejor.

```
public BitField(int mask) {}
public int getValue(int holder) {}
public short getShortValue(short holder) {}
public int getRawValue(int holder) {}
public short getShortRawValue(short holder) {}
public boolean isSet(int holder) {}
public boolean isAllSet(int holder) {}
public int setValue(int holder, int value) {}
public short setShortValue(short holder, short value) {}
public int clear(int holder) {}
public short clearShort(short holder) {}
public byte clearByte(byte holder) {}
public int set(int holder) {}
public short setShort(short holder) {}
public byte setByte(byte holder) {}
public int setBoolean(int holder, boolean flag) {}
public short setShortBoolean(short holder, boolean flag) {}
public byte setByteBoolean(byte holder, boolean flag) {}
```

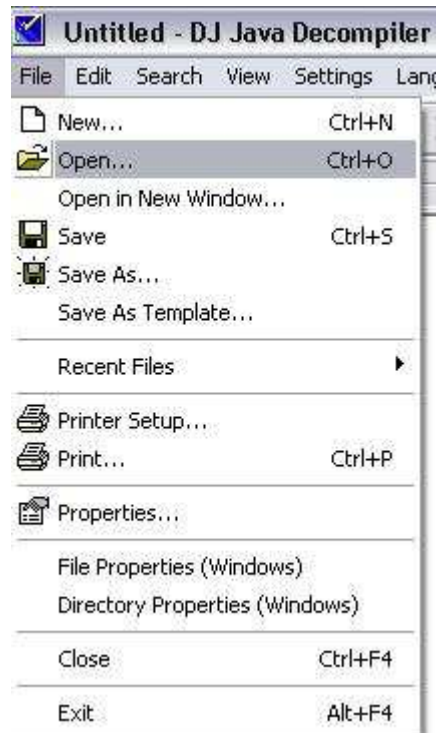
5 Utilidad practica

- DJ Java Decompiler es útil para la recuperación del código fuente perdido o destruido accidentalmente.
- Decompilar Java es una manera excelente de aprender Java y saber cómo trabaja la maquina virtual. Podremos ver las diferentes clases de Java y aprender del código fuente. Tiene un manejo muy sencillo y un intuitivo interfaz grafico que ayudara mucho a los usuarios que comiencen a utilizar Java.
- Corrige y Debuggea los archivos .class. Podremos utilizar DJ Java Decompiler cuando los desarrolladores tarden demasiado en responder una pregunta que necesita respuesta inmediata.
- DJ Java Decompiler es muy útil para explorar el código fuente de las clases Java que se podrán ejecutar.
- Podremos comprobar los resultados de la ofuscación.

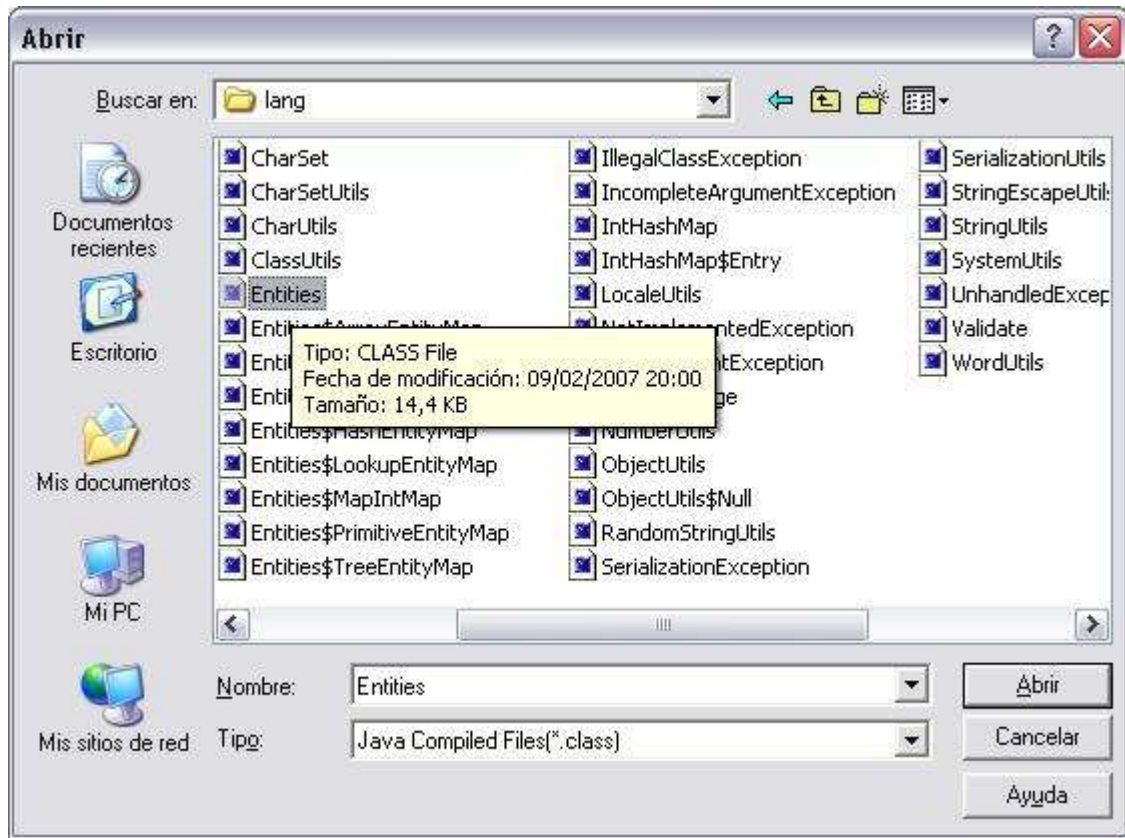
6 Anexo 1: Decompilar un archivo .class

Procederemos a realizar un ejercicio muy sencillo, que consiste en decompilar un archivo. Teniendo en cuenta la utilidad que tendrá esta aplicación será el proceso mas utilizado.

Comenzamos abriendo **DJ Java Decompiler**, aparecerá la pantalla blanco, nos situamos encima de **File** y desplegaremos su menú:



Presionaremos sobre **Open...** y nos aparecerá la siguiente ventana, que nos dará la posibilidad de elegir un archivo .class alojado en el disco local.



En la foto se puede apreciar la forma que tendrán los archivos .class, en este caso abriremos el archivo llamado **Entities**. Aparecerá el siguiente mensaje, preguntando si deseamos añadirle la ayuda de sintaxis que incorpora el programa. Para ello el programa necesitará unos segundos.



Y nos aparecerá de la siguiente manera en la pantalla principal en la aplicación, decompilado y listo para observar lo que necesitamos.

```
// Decompiled by DJ v3.7.7.81 Copyright 2004 Atanas Neshkov Date: 27/09/2007 17:28:17
// Home Page : http://members.fortunecity.com/neshkov/dj.html - Check often for new version!
// Decompiler options: packimports(3)
// Source File Name: Entities.java

package org.apache.commons.lang;

import java.io.*;
import java.util.*;

public BinaryEntityMap() {}
public BinaryEntityMap(int growBy) {}
private int binarySearch(int key) {}
public void add(String name, int value) {}
public String name(int value) {}
public ArrayEntityMap() {}
public ArrayEntityMap(int growBy) {}
public void add(String name, int value) {}
protected void ensureCapacity(int capacity) {}
public String name(int value) {}
public int value(String name) {}
LookupEntityMap() {}
public String name(int value) {}
private String[] lookupTable() {}
private void createLookupTable() {}
```

Como hemos explicado anteriormente en la parte superior aparecerá el código fuente del archivo abierto y en la zona inferior podremos ver un índice del archivo ejecutado, para mejor acceso.