

Java Application: DumpFile

Written by: Keith Fenske, <http://kwfenske.github.io/>

First version: Thursday, 8 February 2007

Document revised: Friday, 7 March 2008

Copyright © 2007 by Keith Fenske. Apache License or GNU General Public License.

Description

DumpFile is a Java 1.4 application to dump the contents of files in hexadecimal and as 8-bit text bytes. For example, using eight input bytes per line of the dump output:

```
Dumping file: C:\MSDOS.SYS
00000000  3B 53 59 53 0D 0A 5B 50  |;SYS..[P|
00000008  61 74 68 73 5D 0D 0A 57  |aths]..w|
00000010  69 6E 44 69 72 3D 43 3A  |inDir=C:|
      ^.^.^
000006E8  73 0D 0A 0D 0A          |s....  |
1,773 bytes dumped.
```

Choose your options; then click on the “Open Files” button and select one or more files to be dumped. Output will be shown in a scrolling text area, and this output can be saved into a text file with the “Save Output As” button or by copying and pasting from the text area.

Apache License or GNU General Public License

DumpFile is free software and has been released under the terms and conditions of the Apache License (version 2.0 or later) and/or the GNU General Public License (GPL, version 2 or later). This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY, without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the license(s) for more details. You should have received a copy of the licenses along with this program. If not, see the <http://www.apache.org/licenses/> and <http://www.gnu.org/licenses/> web pages.

Installation

You must have the Java run-time environment (JRE) installed on your computer. DumpFile was developed with Java 1.4 and should run on later versions. It may also run on earlier versions, but this has not been tested. You can download the JRE from Oracle (formerly Sun Microsystems):

JRE for end users: <http://www.java.com/getjava/>

SDK for programmers: <http://www.oracle.com/technetwork/java/>

IDE for programmers: <http://www.netbeans.org/>

Once Java is installed, you need to put the program files for DumpFile into a folder (directory) on your hard drive. The name of the folder and the location are your choice, except it is easier if the name does not include spaces. Assume that files will go into a “C:\Java” folder. Then create the folder and unpack the Java *.class files into this folder (if you received the program as a ZIP file). The files look something like this:

- ApacheLicense20.txt (12 KB, legal notice)
- DumpFile5.class (12 KB, executable program)
- DumpFile5.doc (32 KB, this documentation in Microsoft Word format)
- DumpFile5.gif (19 KB, sample program image)
- DumpFile5.jar (8 KB, archive file with same class files inside)
- DumpFile5.java (37 KB, source code)
- DumpFile5.manifest (1 KB, main class manifest for archive file)
- DumpFile5.pdf (79 KB, this documentation in Adobe Acrobat format)
- DumpFile5User.class (1 KB, helper class for main program)
- GnuPublicLicense3.txt (35 KB, legal notice)
- RunJavaPrograms.pdf (60 KB, more notes about running Java)

To run the program on Windows, start a DOS command prompt, which is Start button, Programs, Accessories, Command Prompt on Windows XP/Vista/7. Change to the folder with the program files and run the program with a “java” command:

```
c:
cd \java
java DumpFile5
```

The program name “DumpFile5” must appear exactly as shown; uppercase and lowercase letters are different in Java names. Some systems (Macintosh) will run a main “class” file by clicking on the class file name while viewing a directory in the file browser (Mac Finder). Many systems will run a “jar” file by clicking (or double clicking) on the jar file name (Windows Explorer). The command line is the only guaranteed way of running a Java program. Should you find this program to be popular, you can create a Start menu item or desktop shortcut on Windows XP/Vista/7 with a target of “java.exe DumpFile5” starting in the “C:\Java” folder.

One complication may arise when trying to run this program. Java looks for an environment variable called CLASSPATH. If it finds this variable, then that is a list of folders where it looks for *.class files. It won’t look anywhere else, not even in the current directory, unless the path contains “.” as one of the choices. The symptom is an error message that says:

Exception in thread "main" java.lang.NoClassDefFoundError: DumpFile5

To find out if your system has a CLASSPATH variable defined, type the following command in a DOS window:

```
set CLASSPATH
```

To temporarily change the CLASSPATH variable to the current directory, use the following command line:

```
java -cp . DumpFile5
```

To permanently change the CLASSPATH, you must find where it is being set. This is in Control Panel, System, Advanced, Environment Variables on Windows XP/Vista/7.

Removal or Uninstall

To remove this program from your computer, delete the installation files listed above. If the folder that contained the files is now empty, you may also delete the folder ... if you created the folder, of course, not the system. If you created desktop shortcuts or Start menu items, then delete those too. There are no hidden configuration or preference files, and no information is stored in the Windows system registry. You don't need an "uninstall" program.

Graphical Versus Console Application

The Java command line may contain options or file names. If no file names are given on the command line, then this program runs as a graphical or "GUI" application with the usual dialog boxes and windows. If one or more file names are given on the command line, then this program runs as a console application without a graphical interface. See the "-?" option for a help summary:

```
java DumpFile5 -?
```

The dump is written on standard output, and may be redirected with the ">" or ">" operators. (Standard error may be redirected with the ">" operator.) An example command line is:

```
java DumpFile5 -w16 c:\msdos.sys >d:\temp\filedump.txt
```

The graphical interface can be very slow when the output text area gets too big. The Java Swing components JScrollPane and JTextArea get themselves in knots trying to display more than a megabyte of scrolling output text. The console application is much faster and uses a constant amount of virtual memory (under 6 MB), instead of a growing amount that consumes the entire Java heap and eventually generates "out of memory" errors. The fastest way to use this program

is from the command line with standard output redirected to a file. Then open the output file in your favorite word processor or plain text editor.

Restrictions and Limitations

Input bytes are shown only as 7-bit or 8-bit ASCII text. A period (“.”) is substituted for unprintable characters. It would be possible to display pairs of bytes as simple 16-bit Unicode text. However, none of the double-byte shifting character sets common in Chinese, Japanese, or Korean are practical because multi-byte character sequences will most likely be broken across lines in the dump output. Extended Unicode characters can not be supported for the same reason, where multiple 16-bit values are combined into additional character codes.

file: DumpFile5.doc 2021-10-25