Java Application: FileChecksum

Written by: Keith Fenske, http://www.psc-consulting.ca/fenske/
First version: Thursday, 30 October 2008
Document revised: Saturday, 13 February 2010
Copyright © 2008 by Keith Fenske. Released under the GNU General Public License (GPL).

Description

FileChecksum is a Java 1.4 application to compute common checksums for files: CRC32, MD5, and SHA1. Checksums are small hexadecimal "signatures" for testing whether or not files have been copied correctly, such as over a network. One person sends a file along with the checksum computed on the original computer. A second person calculates a similar checksum for the received file, and if the two checksums agree, then the received file is assumed to be correct. MD5 is more reliable than and preferred over the older and simpler CRC32. Many web sites provide MD5 signatures for their downloads; use this program to verify files that you download.

See the CompareFolders application for comparing two folders to determine if all files and subfolders are identical. See the FindDupFiles application to look for duplicate files based on MD5 checksums.

GNU General Public License (GPL)

FileChecksum is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License or (at your option) any later version. 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 GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program. If not, see the http://www.gnu.org/licenses/ web page.

Installation

You must have the Java run-time environment (JRE) installed on your computer. FileChecksum 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. For Macintosh computers, the version of Java is determined by your version of MacOS. For Windows, Linux, and Solaris, you can download the JRE from Sun Microsystems:

Sun Java
    JRE for end users: http://www.java.com/getjava/
    SDK for programmers: http://developers.sun.com/downloads/
    IDE for programmers: http://www.netbeans.org/

Once Java is installed, you need to put the program files for FileChecksum 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:

```
FileChecksum3.au  (3 KB, barking dog sound)
FileChecksum3.class  (18 KB, executable program)
FileChecksum3.doc  (33 KB, this documentation in Microsoft Word format)
FileChecksum3.gif  (14 KB, sample program image)
FileChecksum3.ico  (7 KB, icon for Windows)
FileChecksum3.jar  (11 KB, archive file with same class files inside)
FileChecksum3.java  (55 KB, source code)
FileChecksum3.manifest  (1 KB, main class manifest for archive file)
FileChecksum3.pdf  (72 KB, this documentation in Adobe Acrobat format)
FileChecksum3User.class  (1 KB, helper class for main program)
GnuPublicLicense3.txt  (35 KB, legal notice)
RunJavaPrograms.pdf  (88 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 2000/XP. Change to the folder with the program files and run the program with a "java" command:

```
c:
cd \java
java  FileChecksum3
```

The program name "FileChecksum3" 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 2000/XP with a target of "java.exe FileChecksum3" 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: FileChecksum3
```

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  .  FileChecksum3
```

To permanently change the CLASSPATH, you must find where it is being set. This may be in an old AUTOEXEC.* file in the root directory of your system disk (usually the C:\ folder), or it may be in Control Panel, System, Advanced, Environment Variables on Windows 2000/XP.

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 parameters for a file name and optional checksums. If no parameters are given on the command line, then this program runs as a graphical or "GUI" application with the usual dialog boxes and windows. See the "-?" option for a help summary:

```
java  FileChecksum3  -?
```

The command line has more options than are visible in the graphical interface. An option such as -u14 or -u16 is recommended because the default Java font is too small. If parameters are given on the command line, then this program runs as a console application without a graphical interface. The first parameter must be a file name. Checksums are calculated for that file. A generated report is written on standard output, and may be redirected with the ">" or "1>" operators. (Standard error may be redirected with the "2>" operator.) An example command line is:

```
java  FileChecksum3  README.TXT  >output.txt
```

This will calculate checksums for a file named "README.TXT". Standard output will be written to a file named "output.txt". The second and following parameters, if given, must be hexadecimal checksums. For example:

    java FileChecksum3 README.TXT d36952838c47c701745293e1a16333f3

Second and following parameters are compared against the generated checksums (CRC32, MD5, SHA1). If each parameter matches a checksum, then the result is considered successful. The console application will return an exit status of 1 for success, –1 for failure, and 0 for unknown.