

# Java Application: CheckPlainTrim

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

First version: Monday, 25 September 2017

Document revised: Monday, 25 September 2017

Copyright © 2017 by Keith Fenske. Apache License or GNU General Public License.

## Description

CheckPlainTrim is a Java 1.4 application to check if files are in plain text and do not have trailing spaces or tabs (white space) at the end of lines. How clean are your source, text files, and XML documents? The following are accepted:

- horizontal tab (0x09)
- line feed (0x0A)
- carriage return (0x0D)
- blank space (0x20)
- printable US-ASCII text (0x21 to 0x7E)

By default, files are assumed to be in the local character set (text encoding). You may select another encoding or raw 8-bit data bytes. See the TrimFile Java application to detect and remove trailing white space, in various character sets.

## Apache License or GNU General Public License

CheckPlainTrim 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. CheckPlainTrim 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 CheckPlainTrim 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)  
CheckPlainTrim2.class (22 KB, executable program)  
CheckPlainTrim2.doc (32 KB, this documentation in Microsoft Word format)  
CheckPlainTrim2.gif (16 KB, sample program image)  
CheckPlainTrim2.ico (87 KB, icon for Windows)  
CheckPlainTrim2.jar (13 KB, archive file with same class files inside)  
CheckPlainTrim2.java (63 KB, source code)  
CheckPlainTrim2.manifest (1 KB, main class manifest for archive file)  
CheckPlainTrim2.pdf (69 KB, this documentation in Adobe Acrobat format)  
CheckPlainTrim2User.class (1 KB, helper class for main program)  
GnuPublicLicense3.txt (35 KB, legal notice)  
RunJavaPrograms.pdf (60 KB, more notes about running Java)  
ShowCharSets2.java (3 KB, program to generate Java character set names)  
ShowCharSets2.txt (19 KB, sample file with Java character set names)

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 CheckPlainTrim2
```

The program name “CheckPlainTrim2” 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 CheckPlainTrim2” 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: CheckPlainTrim2
```

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 . CheckPlainTrim2
```

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 and folder names. If no file or folder names 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 CheckPlainTrim2 -?
```

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 file or folder names are given on the command line, then this program runs as a console application without a graphical interface. 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 CheckPlainTrim2 -s d:\temp >report.txt
```

The console application will return an exit status of 1 for success, -1 for failure, and 0 for unknown. The graphical interface can be very slow when the output text area gets too big, which will happen if thousands of files are reported.

---

file: CheckPlainTrim2.doc 2024-12-12