**MavenWrapperDownloader.java**

|  |
| --- |
| /\* |
|  | \* Copyright 2007-present the original author or authors. |
|  | \* |
|  | \* Licensed under the Apache License, Version 2.0 (the "License"); |
|  | \* you may not use this file except in compliance with the License. |
|  | \* You may obtain a copy of the License at |
|  | \* |
|  | \* https://www.apache.org/licenses/LICENSE-2.0 |
|  | \* |
|  | \* Unless required by applicable law or agreed to in writing, software |
|  | \* distributed under the License is distributed on an "AS IS" BASIS, |
|  | \* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. |
|  | \* See the License for the specific language governing permissions and |
|  | \* limitations under the License. |
|  | \*/ |
|  | import java.net.\*; |
|  | import java.io.\*; |
|  | import java.nio.channels.\*; |
|  | import java.util.Properties; |
|  |  |
|  | public class MavenWrapperDownloader { |
|  |  |
|  | private static final String WRAPPER\_VERSION = "0.5.6"; |
|  | /\*\* |
|  | \* Default URL to download the maven-wrapper.jar from, if no 'downloadUrl' is provided. |
|  | \*/ |
|  | private static final String DEFAULT\_DOWNLOAD\_URL = "https://repo.maven.apache.org/maven2/io/takari/maven-wrapper/" |
|  | + WRAPPER\_VERSION + "/maven-wrapper-" + WRAPPER\_VERSION + ".jar"; |
|  |  |
|  | /\*\* |
|  | \* Path to the maven-wrapper.properties file, which might contain a downloadUrl property to |
|  | \* use instead of the default one. |
|  | \*/ |
|  | private static final String MAVEN\_WRAPPER\_PROPERTIES\_PATH = |
|  | ".mvn/wrapper/maven-wrapper.properties"; |
|  |  |
|  | /\*\* |
|  | \* Path where the maven-wrapper.jar will be saved to. |
|  | \*/ |
|  | private static final String MAVEN\_WRAPPER\_JAR\_PATH = |
|  | ".mvn/wrapper/maven-wrapper.jar"; |
|  |  |
|  | /\*\* |
|  | \* Name of the property which should be used to override the default download url for the wrapper. |
|  | \*/ |
|  | private static final String PROPERTY\_NAME\_WRAPPER\_URL = "wrapperUrl"; |
|  |  |
|  | public static void main(String args[]) { |
|  | System.out.println("- Downloader started"); |
|  | File baseDirectory = new File(args[0]); |
|  | System.out.println("- Using base directory: " + baseDirectory.getAbsolutePath()); |
|  |  |
|  | // If the maven-wrapper.properties exists, read it and check if it contains a custom |
|  | // wrapperUrl parameter. |
|  | File mavenWrapperPropertyFile = new File(baseDirectory, MAVEN\_WRAPPER\_PROPERTIES\_PATH); |
|  | String url = DEFAULT\_DOWNLOAD\_URL; |
|  | if(mavenWrapperPropertyFile.exists()) { |
|  | FileInputStream mavenWrapperPropertyFileInputStream = null; |
|  | try { |
|  | mavenWrapperPropertyFileInputStream = new FileInputStream(mavenWrapperPropertyFile); |
|  | Properties mavenWrapperProperties = new Properties(); |
|  | mavenWrapperProperties.load(mavenWrapperPropertyFileInputStream); |
|  | url = mavenWrapperProperties.getProperty(PROPERTY\_NAME\_WRAPPER\_URL, url); |
|  | } catch (IOException e) { |
|  | System.out.println("- ERROR loading '" + MAVEN\_WRAPPER\_PROPERTIES\_PATH + "'"); |
|  | } finally { |
|  | try { |
|  | if(mavenWrapperPropertyFileInputStream != null) { |
|  | mavenWrapperPropertyFileInputStream.close(); |
|  | } |
|  | } catch (IOException e) { |
|  | // Ignore ... |
|  | } |
|  | } |
|  | } |
|  | System.out.println("- Downloading from: " + url); |
|  |  |
|  | File outputFile = new File(baseDirectory.getAbsolutePath(), MAVEN\_WRAPPER\_JAR\_PATH); |
|  | if(!outputFile.getParentFile().exists()) { |
|  | if(!outputFile.getParentFile().mkdirs()) { |
|  | System.out.println( |
|  | "- ERROR creating output directory '" + outputFile.getParentFile().getAbsolutePath() + "'"); |
|  | } |
|  | } |
|  | System.out.println("- Downloading to: " + outputFile.getAbsolutePath()); |
|  | try { |
|  | downloadFileFromURL(url, outputFile); |
|  | System.out.println("Done"); |
|  | System.exit(0); |
|  | } catch (Throwable e) { |
|  | System.out.println("- Error downloading"); |
|  | e.printStackTrace(); |
|  | System.exit(1); |
|  | } |
|  | } |
|  |  |
|  | private static void downloadFileFromURL(String urlString, File destination) throws Exception { |
|  | if (System.getenv("MVNW\_USERNAME") != null && System.getenv("MVNW\_PASSWORD") != null) { |
|  | String username = System.getenv("MVNW\_USERNAME"); |
|  | char[] password = System.getenv("MVNW\_PASSWORD").toCharArray(); |
|  | Authenticator.setDefault(new Authenticator() { |
|  | @Override |
|  | protected PasswordAuthentication getPasswordAuthentication() { |
|  | return new PasswordAuthentication(username, password); |
|  | } |
|  | }); |
|  | } |
|  | URL website = new URL(urlString); |
|  | ReadableByteChannel rbc; |
|  | rbc = Channels.newChannel(website.openStream()); |
|  | FileOutputStream fos = new FileOutputStream(destination); |
|  | fos.getChannel().transferFrom(rbc, 0, Long.MAX\_VALUE); |
|  | fos.close(); |
|  | rbc.close(); |
|  | } |
|  |  |
|  | } |

## ****SpringBootAwsApplicationTests.java****

|  |
| --- |
| package com.SpringBoot.SpringBoot\_aws; |
|  |  |
|  | import org.junit.jupiter.api.Test; |
|  | import org.springframework.boot.test.context.SpringBootTest; |
|  |  |
|  | @SpringBootTest |
|  | class SpringBootAwsApplicationTests { |
|  |  |
|  | @Test |
|  | void contextLoads() { |
|  | } |
|  |  |
|  | } |

## ****Application.properties****

server.port=8085

|  |
| --- |
| ****mvnw**** #!/bin/sh |
|  | # ---------------------------------------------------------------------------- |
|  | # Licensed to the Apache Software Foundation (ASF) under one |
|  | # or more contributor license agreements. See the NOTICE file |
|  | # distributed with this work for additional information |
|  | # regarding copyright ownership. The ASF licenses this file |
|  | # to you under the Apache License, Version 2.0 (the |
|  | # "License"); you may not use this file except in compliance |
|  | # with the License. You may obtain a copy of the License at |
|  | # |
|  | # https://www.apache.org/licenses/LICENSE-2.0 |
|  | # |
|  | # Unless required by applicable law or agreed to in writing, |
|  | # software distributed under the License is distributed on an |
|  | # "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY |
|  | # KIND, either express or implied. See the License for the |
|  | # specific language governing permissions and limitations |
|  | # under the License. |
|  | # ---------------------------------------------------------------------------- |
|  |  |
|  | # ---------------------------------------------------------------------------- |
|  | # Maven Start Up Batch script |
|  | # |
|  | # Required ENV vars: |
|  | # ------------------ |
|  | # JAVA\_HOME - location of a JDK home dir |
|  | # |
|  | # Optional ENV vars |
|  | # ----------------- |
|  | # M2\_HOME - location of maven2's installed home dir |
|  | # MAVEN\_OPTS - parameters passed to the Java VM when running Maven |
|  | # e.g. to debug Maven itself, use |
|  | # set MAVEN\_OPTS=-Xdebug -Xrunjdwp:transport=dt\_socket,server=y,suspend=y,address=8000 |
|  | # MAVEN\_SKIP\_RC - flag to disable loading of mavenrc files |
|  | # ---------------------------------------------------------------------------- |
|  |  |
|  | if [ -z "$MAVEN\_SKIP\_RC" ] ; then |
|  |  |
|  | if [ -f /etc/mavenrc ] ; then |
|  | . /etc/mavenrc |
|  | fi |
|  |  |
|  | if [ -f "$HOME/.mavenrc" ] ; then |
|  | . "$HOME/.mavenrc" |
|  | fi |
|  |  |
|  | fi |
|  |  |
|  | # OS specific support. $var \_must\_ be set to either true or false. |
|  | cygwin=false; |
|  | darwin=false; |
|  | mingw=false |
|  | case "`uname`" in |
|  | CYGWIN\*) cygwin=true ;; |
|  | MINGW\*) mingw=true;; |
|  | Darwin\*) darwin=true |
|  | # Use /usr/libexec/java\_home if available, otherwise fall back to /Library/Java/Home |
|  | # See https://developer.apple.com/library/mac/qa/qa1170/\_index.html |
|  | if [ -z "$JAVA\_HOME" ]; then |
|  | if [ -x "/usr/libexec/java\_home" ]; then |
|  | export JAVA\_HOME="`/usr/libexec/java\_home`" |
|  | else |
|  | export JAVA\_HOME="/Library/Java/Home" |
|  | fi |
|  | fi |
|  | ;; |
|  | esac |
|  |  |
|  | if [ -z "$JAVA\_HOME" ] ; then |
|  | if [ -r /etc/gentoo-release ] ; then |
|  | JAVA\_HOME=`java-config --jre-home` |
|  | fi |
|  | fi |
|  |  |
|  | if [ -z "$M2\_HOME" ] ; then |
|  | ## resolve links - $0 may be a link to maven's home |
|  | PRG="$0" |
|  |  |
|  | # need this for relative symlinks |
|  | while [ -h "$PRG" ] ; do |
|  | ls=`ls -ld "$PRG"` |
|  | link=`expr "$ls" : '.\*-> \(.\*\)$'` |
|  | if expr "$link" : '/.\*' > /dev/null; then |
|  | PRG="$link" |
|  | else |
|  | PRG="`dirname "$PRG"`/$link" |
|  | fi |
|  | done |
|  |  |
|  | saveddir=`pwd` |
|  |  |
|  | M2\_HOME=`dirname "$PRG"`/.. |
|  |  |
|  | # make it fully qualified |
|  | M2\_HOME=`cd "$M2\_HOME" && pwd` |
|  |  |
|  | cd "$saveddir" |
|  | # echo Using m2 at $M2\_HOME |
|  | fi |
|  |  |
|  | # For Cygwin, ensure paths are in UNIX format before anything is touched |
|  | if $cygwin ; then |
|  | [ -n "$M2\_HOME" ] && |
|  | M2\_HOME=`cygpath --unix "$M2\_HOME"` |
|  | [ -n "$JAVA\_HOME" ] && |
|  | JAVA\_HOME=`cygpath --unix "$JAVA\_HOME"` |
|  | [ -n "$CLASSPATH" ] && |
|  | CLASSPATH=`cygpath --path --unix "$CLASSPATH"` |
|  | fi |
|  |  |
|  | # For Mingw, ensure paths are in UNIX format before anything is touched |
|  | if $mingw ; then |
|  | [ -n "$M2\_HOME" ] && |
|  | M2\_HOME="`(cd "$M2\_HOME"; pwd)`" |
|  | [ -n "$JAVA\_HOME" ] && |
|  | JAVA\_HOME="`(cd "$JAVA\_HOME"; pwd)`" |
|  | fi |
|  |  |
|  | if [ -z "$JAVA\_HOME" ]; then |
|  | javaExecutable="`which javac`" |
|  | if [ -n "$javaExecutable" ] && ! [ "`expr \"$javaExecutable\" : '\([^ ]\*\)'`" = "no" ]; then |
|  | # readlink(1) is not available as standard on Solaris 10. |
|  | readLink=`which readlink` |
|  | if [ ! `expr "$readLink" : '\([^ ]\*\)'` = "no" ]; then |
|  | if $darwin ; then |
|  | javaHome="`dirname \"$javaExecutable\"`" |
|  | javaExecutable="`cd \"$javaHome\" && pwd -P`/javac" |
|  | else |
|  | javaExecutable="`readlink -f \"$javaExecutable\"`" |
|  | fi |
|  | javaHome="`dirname \"$javaExecutable\"`" |
|  | javaHome=`expr "$javaHome" : '\(.\*\)/bin'` |
|  | JAVA\_HOME="$javaHome" |
|  | export JAVA\_HOME |
|  | fi |
|  | fi |
|  | fi |
|  |  |
|  | if [ -z "$JAVACMD" ] ; then |
|  | if [ -n "$JAVA\_HOME" ] ; then |
|  | if [ -x "$JAVA\_HOME/jre/sh/java" ] ; then |
|  | # IBM's JDK on AIX uses strange locations for the executables |
|  | JAVACMD="$JAVA\_HOME/jre/sh/java" |
|  | else |
|  | JAVACMD="$JAVA\_HOME/bin/java" |
|  | fi |
|  | else |
|  | JAVACMD="`which java`" |
|  | fi |
|  | fi |
|  |  |
|  | if [ ! -x "$JAVACMD" ] ; then |
|  | echo "Error: JAVA\_HOME is not defined correctly." >&2 |
|  | echo " We cannot execute $JAVACMD" >&2 |
|  | exit 1 |
|  | fi |
|  |  |
|  | if [ -z "$JAVA\_HOME" ] ; then |
|  | echo "Warning: JAVA\_HOME environment variable is not set." |
|  | fi |
|  |  |
|  | CLASSWORLDS\_LAUNCHER=org.codehaus.plexus.classworlds.launcher.Launcher |
|  |  |
|  | # traverses directory structure from process work directory to filesystem root |
|  | # first directory with .mvn subdirectory is considered project base directory |
|  | find\_maven\_basedir() { |
|  |  |
|  | if [ -z "$1" ] |
|  | then |
|  | echo "Path not specified to find\_maven\_basedir" |
|  | return 1 |
|  | fi |
|  |  |
|  | basedir="$1" |
|  | wdir="$1" |
|  | while [ "$wdir" != '/' ] ; do |
|  | if [ -d "$wdir"/.mvn ] ; then |
|  | basedir=$wdir |
|  | break |
|  | fi |
|  | # workaround for JBEAP-8937 (on Solaris 10/Sparc) |
|  | if [ -d "${wdir}" ]; then |
|  | wdir=`cd "$wdir/.."; pwd` |
|  | fi |
|  | # end of workaround |
|  | done |
|  | echo "${basedir}" |
|  | } |
|  |  |
|  | # concatenates all lines of a file |
|  | concat\_lines() { |
|  | if [ -f "$1" ]; then |
|  | echo "$(tr -s '\n' ' ' < "$1")" |
|  | fi |
|  | } |
|  |  |
|  | BASE\_DIR=`find\_maven\_basedir "$(pwd)"` |
|  | if [ -z "$BASE\_DIR" ]; then |
|  | exit 1; |
|  | fi |
|  |  |
|  | ########################################################################################## |
|  | # Extension to allow automatically downloading the maven-wrapper.jar from Maven-central |
|  | # This allows using the maven wrapper in projects that prohibit checking in binary data. |
|  | ########################################################################################## |
|  | if [ -r "$BASE\_DIR/.mvn/wrapper/maven-wrapper.jar" ]; then |
|  | if [ "$MVNW\_VERBOSE" = true ]; then |
|  | echo "Found .mvn/wrapper/maven-wrapper.jar" |
|  | fi |
|  | else |
|  | if [ "$MVNW\_VERBOSE" = true ]; then |
|  | echo "Couldn't find .mvn/wrapper/maven-wrapper.jar, downloading it ..." |
|  | fi |
|  | if [ -n "$MVNW\_REPOURL" ]; then |
|  | jarUrl="$MVNW\_REPOURL/io/takari/maven-wrapper/0.5.6/maven-wrapper-0.5.6.jar" |
|  | else |
|  | jarUrl="https://repo.maven.apache.org/maven2/io/takari/maven-wrapper/0.5.6/maven-wrapper-0.5.6.jar" |
|  | fi |
|  | while IFS="=" read key value; do |
|  | case "$key" in (wrapperUrl) jarUrl="$value"; break ;; |
|  | esac |
|  | done < "$BASE\_DIR/.mvn/wrapper/maven-wrapper.properties" |
|  | if [ "$MVNW\_VERBOSE" = true ]; then |
|  | echo "Downloading from: $jarUrl" |
|  | fi |
|  | wrapperJarPath="$BASE\_DIR/.mvn/wrapper/maven-wrapper.jar" |
|  | if $cygwin; then |
|  | wrapperJarPath=`cygpath --path --windows "$wrapperJarPath"` |
|  | fi |
|  |  |
|  | if command -v wget > /dev/null; then |
|  | if [ "$MVNW\_VERBOSE" = true ]; then |
|  | echo "Found wget ... using wget" |
|  | fi |
|  | if [ -z "$MVNW\_USERNAME" ] || [ -z "$MVNW\_PASSWORD" ]; then |
|  | wget "$jarUrl" -O "$wrapperJarPath" |
|  | else |
|  | wget --http-user=$MVNW\_USERNAME --http-password=$MVNW\_PASSWORD "$jarUrl" -O "$wrapperJarPath" |
|  | fi |
|  | elif command -v curl > /dev/null; then |
|  | if [ "$MVNW\_VERBOSE" = true ]; then |
|  | echo "Found curl ... using curl" |
|  | fi |
|  | if [ -z "$MVNW\_USERNAME" ] || [ -z "$MVNW\_PASSWORD" ]; then |
|  | curl -o "$wrapperJarPath" "$jarUrl" -f |
|  | else |
|  | curl --user $MVNW\_USERNAME:$MVNW\_PASSWORD -o "$wrapperJarPath" "$jarUrl" -f |
|  | fi |
|  |  |
|  | else |
|  | if [ "$MVNW\_VERBOSE" = true ]; then |
|  | echo "Falling back to using Java to download" |
|  | fi |
|  | javaClass="$BASE\_DIR/.mvn/wrapper/MavenWrapperDownloader.java" |
|  | # For Cygwin, switch paths to Windows format before running javac |
|  | if $cygwin; then |
|  | javaClass=`cygpath --path --windows "$javaClass"` |
|  | fi |
|  | if [ -e "$javaClass" ]; then |
|  | if [ ! -e "$BASE\_DIR/.mvn/wrapper/MavenWrapperDownloader.class" ]; then |
|  | if [ "$MVNW\_VERBOSE" = true ]; then |
|  | echo " - Compiling MavenWrapperDownloader.java ..." |
|  | fi |
|  | # Compiling the Java class |
|  | ("$JAVA\_HOME/bin/javac" "$javaClass") |
|  | fi |
|  | if [ -e "$BASE\_DIR/.mvn/wrapper/MavenWrapperDownloader.class" ]; then |
|  | # Running the downloader |
|  | if [ "$MVNW\_VERBOSE" = true ]; then |
|  | echo " - Running MavenWrapperDownloader.java ..." |
|  | fi |
|  | ("$JAVA\_HOME/bin/java" -cp .mvn/wrapper MavenWrapperDownloader "$MAVEN\_PROJECTBASEDIR") |
|  | fi |
|  | fi |
|  | fi |
|  | fi |
|  | ########################################################################################## |
|  | # End of extension |
|  | ########################################################################################## |
|  |  |
|  | export MAVEN\_PROJECTBASEDIR=${MAVEN\_BASEDIR:-"$BASE\_DIR"} |
|  | if [ "$MVNW\_VERBOSE" = true ]; then |
|  | echo $MAVEN\_PROJECTBASEDIR |
|  | fi |
|  | MAVEN\_OPTS="$(concat\_lines "$MAVEN\_PROJECTBASEDIR/.mvn/jvm.config") $MAVEN\_OPTS" |
|  |  |
|  | # For Cygwin, switch paths to Windows format before running java |
|  | if $cygwin; then |
|  | [ -n "$M2\_HOME" ] && |
|  | M2\_HOME=`cygpath --path --windows "$M2\_HOME"` |
|  | [ -n "$JAVA\_HOME" ] && |
|  | JAVA\_HOME=`cygpath --path --windows "$JAVA\_HOME"` |
|  | [ -n "$CLASSPATH" ] && |
|  | CLASSPATH=`cygpath --path --windows "$CLASSPATH"` |
|  | [ -n "$MAVEN\_PROJECTBASEDIR" ] && |
|  | MAVEN\_PROJECTBASEDIR=`cygpath --path --windows "$MAVEN\_PROJECTBASEDIR"` |
|  | fi |
|  |  |
|  | # Provide a "standardized" way to retrieve the CLI args that will |
|  | # work with both Windows and non-Windows executions. |
|  | MAVEN\_CMD\_LINE\_ARGS="$MAVEN\_CONFIG $@" |
|  | export MAVEN\_CMD\_LINE\_ARGS |
|  |  |
|  | WRAPPER\_LAUNCHER=org.apache.maven.wrapper.MavenWrapperMain |
|  |  |
|  | exec "$JAVACMD" \ |
|  | $MAVEN\_OPTS \ |
|  | -classpath "$MAVEN\_PROJECTBASEDIR/.mvn/wrapper/maven-wrapper.jar" \ |
|  | "-Dmaven.home=${M2\_HOME}" "-Dmaven.multiModuleProjectDirectory=${MAVEN\_PROJECTBASEDIR}" \ |
|  | ${WRAPPER\_LAUNCHER} $MAVEN\_CONFIG "$@" |

## ****mvnw.cmd****

|  |
| --- |
| @REM ---------------------------------------------------------------------------- |
|  | @REM Licensed to the Apache Software Foundation (ASF) under one |
|  | @REM or more contributor license agreements. See the NOTICE file |
|  | @REM distributed with this work for additional information |
|  | @REM regarding copyright ownership. The ASF licenses this file |
|  | @REM to you under the Apache License, Version 2.0 (the |
|  | @REM "License"); you may not use this file except in compliance |
|  | @REM with the License. You may obtain a copy of the License at |
|  | @REM |
|  | @REM https://www.apache.org/licenses/LICENSE-2.0 |
|  | @REM |
|  | @REM Unless required by applicable law or agreed to in writing, |
|  | @REM software distributed under the License is distributed on an |
|  | @REM "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY |
|  | @REM KIND, either express or implied. See the License for the |
|  | @REM specific language governing permissions and limitations |
|  | @REM under the License. |
|  | @REM ---------------------------------------------------------------------------- |
|  |  |
|  | @REM ---------------------------------------------------------------------------- |
|  | @REM Maven Start Up Batch script |
|  | @REM |
|  | @REM Required ENV vars: |
|  | @REM JAVA\_HOME - location of a JDK home dir |
|  | @REM |
|  | @REM Optional ENV vars |
|  | @REM M2\_HOME - location of maven2's installed home dir |
|  | @REM MAVEN\_BATCH\_ECHO - set to 'on' to enable the echoing of the batch commands |
|  | @REM MAVEN\_BATCH\_PAUSE - set to 'on' to wait for a keystroke before ending |
|  | @REM MAVEN\_OPTS - parameters passed to the Java VM when running Maven |
|  | @REM e.g. to debug Maven itself, use |
|  | @REM set MAVEN\_OPTS=-Xdebug -Xrunjdwp:transport=dt\_socket,server=y,suspend=y,address=8000 |
|  | @REM MAVEN\_SKIP\_RC - flag to disable loading of mavenrc files |
|  | @REM ---------------------------------------------------------------------------- |
|  |  |
|  | @REM Begin all REM lines with '@' in case MAVEN\_BATCH\_ECHO is 'on' |
|  | @echo off |
|  | @REM set title of command window |
|  | title %0 |
|  | @REM enable echoing by setting MAVEN\_BATCH\_ECHO to 'on' |
|  | @if "%MAVEN\_BATCH\_ECHO%" == "on" echo %MAVEN\_BATCH\_ECHO% |
|  |  |
|  | @REM set %HOME% to equivalent of $HOME |
|  | if "%HOME%" == "" (set "HOME=%HOMEDRIVE%%HOMEPATH%") |
|  |  |
|  | @REM Execute a user defined script before this one |
|  | if not "%MAVEN\_SKIP\_RC%" == "" goto skipRcPre |
|  | @REM check for pre script, once with legacy .bat ending and once with .cmd ending |
|  | if exist "%HOME%\mavenrc\_pre.bat" call "%HOME%\mavenrc\_pre.bat" |
|  | if exist "%HOME%\mavenrc\_pre.cmd" call "%HOME%\mavenrc\_pre.cmd" |
|  | :skipRcPre |
|  |  |
|  | @setlocal |
|  |  |
|  | set ERROR\_CODE=0 |
|  |  |
|  | @REM To isolate internal variables from possible post scripts, we use another setlocal |
|  | @setlocal |
|  |  |
|  | @REM ==== START VALIDATION ==== |
|  | if not "%JAVA\_HOME%" == "" goto OkJHome |
|  |  |
|  | echo. |
|  | echo Error: JAVA\_HOME not found in your environment. >&2 |
|  | echo Please set the JAVA\_HOME variable in your environment to match the >&2 |
|  | echo location of your Java installation. >&2 |
|  | echo. |
|  | goto error |
|  |  |
|  | :OkJHome |
|  | if exist "%JAVA\_HOME%\bin\java.exe" goto init |
|  |  |
|  | echo. |
|  | echo Error: JAVA\_HOME is set to an invalid directory. >&2 |
|  | echo JAVA\_HOME = "%JAVA\_HOME%" >&2 |
|  | echo Please set the JAVA\_HOME variable in your environment to match the >&2 |
|  | echo location of your Java installation. >&2 |
|  | echo. |
|  | goto error |
|  |  |
|  | @REM ==== END VALIDATION ==== |
|  |  |
|  | :init |
|  |  |
|  | @REM Find the project base dir, i.e. the directory that contains the folder ".mvn". |
|  | @REM Fallback to current working directory if not found. |
|  |  |
|  | set MAVEN\_PROJECTBASEDIR=%MAVEN\_BASEDIR% |
|  | IF NOT "%MAVEN\_PROJECTBASEDIR%"=="" goto endDetectBaseDir |
|  |  |
|  | set EXEC\_DIR=%CD% |
|  | set WDIR=%EXEC\_DIR% |
|  | :findBaseDir |
|  | IF EXIST "%WDIR%"\.mvn goto baseDirFound |
|  | cd .. |
|  | IF "%WDIR%"=="%CD%" goto baseDirNotFound |
|  | set WDIR=%CD% |
|  | goto findBaseDir |
|  |  |
|  | :baseDirFound |
|  | set MAVEN\_PROJECTBASEDIR=%WDIR% |
|  | cd "%EXEC\_DIR%" |
|  | goto endDetectBaseDir |
|  |  |
|  | :baseDirNotFound |
|  | set MAVEN\_PROJECTBASEDIR=%EXEC\_DIR% |
|  | cd "%EXEC\_DIR%" |
|  |  |
|  | :endDetectBaseDir |
|  |  |
|  | IF NOT EXIST "%MAVEN\_PROJECTBASEDIR%\.mvn\jvm.config" goto endReadAdditionalConfig |
|  |  |
|  | @setlocal EnableExtensions EnableDelayedExpansion |
|  | for /F "usebackq delims=" %%a in ("%MAVEN\_PROJECTBASEDIR%\.mvn\jvm.config") do set JVM\_CONFIG\_MAVEN\_PROPS=!JVM\_CONFIG\_MAVEN\_PROPS! %%a |
|  | @endlocal & set JVM\_CONFIG\_MAVEN\_PROPS=%JVM\_CONFIG\_MAVEN\_PROPS% |
|  |  |
|  | :endReadAdditionalConfig |
|  |  |
|  | SET MAVEN\_JAVA\_EXE="%JAVA\_HOME%\bin\java.exe" |
|  | set WRAPPER\_JAR="%MAVEN\_PROJECTBASEDIR%\.mvn\wrapper\maven-wrapper.jar" |
|  | set WRAPPER\_LAUNCHER=org.apache.maven.wrapper.MavenWrapperMain |
|  |  |
|  | set DOWNLOAD\_URL="https://repo.maven.apache.org/maven2/io/takari/maven-wrapper/0.5.6/maven-wrapper-0.5.6.jar" |
|  |  |
|  | FOR /F "tokens=1,2 delims==" %%A IN ("%MAVEN\_PROJECTBASEDIR%\.mvn\wrapper\maven-wrapper.properties") DO ( |
|  | IF "%%A"=="wrapperUrl" SET DOWNLOAD\_URL=%%B |
|  | ) |
|  |  |
|  | @REM Extension to allow automatically downloading the maven-wrapper.jar from Maven-central |
|  | @REM This allows using the maven wrapper in projects that prohibit checking in binary data. |
|  | if exist %WRAPPER\_JAR% ( |
|  | if "%MVNW\_VERBOSE%" == "true" ( |
|  | echo Found %WRAPPER\_JAR% |
|  | ) |
|  | ) else ( |
|  | if not "%MVNW\_REPOURL%" == "" ( |
|  | SET DOWNLOAD\_URL="%MVNW\_REPOURL%/io/takari/maven-wrapper/0.5.6/maven-wrapper-0.5.6.jar" |
|  | ) |
|  | if "%MVNW\_VERBOSE%" == "true" ( |
|  | echo Couldn't find %WRAPPER\_JAR%, downloading it ... |
|  | echo Downloading from: %DOWNLOAD\_URL% |
|  | ) |
|  |  |
|  | powershell -Command "&{"^ |
|  | "$webclient = new-object System.Net.WebClient;"^ |
|  | "if (-not ([string]::IsNullOrEmpty('%MVNW\_USERNAME%') -and [string]::IsNullOrEmpty('%MVNW\_PASSWORD%'))) {"^ |
|  | "$webclient.Credentials = new-object System.Net.NetworkCredential('%MVNW\_USERNAME%', '%MVNW\_PASSWORD%');"^ |
|  | "}"^ |
|  | "[Net.ServicePointManager]::SecurityProtocol = [Net.SecurityProtocolType]::Tls12; $webclient.DownloadFile('%DOWNLOAD\_URL%', '%WRAPPER\_JAR%')"^ |
|  | "}" |
|  | if "%MVNW\_VERBOSE%" == "true" ( |
|  | echo Finished downloading %WRAPPER\_JAR% |
|  | ) |
|  | ) |
|  | @REM End of extension |
|  |  |
|  | @REM Provide a "standardized" way to retrieve the CLI args that will |
|  | @REM work with both Windows and non-Windows executions. |
|  | set MAVEN\_CMD\_LINE\_ARGS=%\* |
|  |  |
|  | %MAVEN\_JAVA\_EXE% %JVM\_CONFIG\_MAVEN\_PROPS% %MAVEN\_OPTS% %MAVEN\_DEBUG\_OPTS% -classpath %WRAPPER\_JAR% "-Dmaven.multiModuleProjectDirectory=%MAVEN\_PROJECTBASEDIR%" %WRAPPER\_LAUNCHER% %MAVEN\_CONFIG% %\* |
|  | if ERRORLEVEL 1 goto error |
|  | goto end |
|  |  |
|  | :error |
|  | set ERROR\_CODE=1 |
|  |  |
|  | :end |
|  | @endlocal & set ERROR\_CODE=%ERROR\_CODE% |
|  |  |
|  | if not "%MAVEN\_SKIP\_RC%" == "" goto skipRcPost |
|  | @REM check for post script, once with legacy .bat ending and once with .cmd ending |
|  | if exist "%HOME%\mavenrc\_post.bat" call "%HOME%\mavenrc\_post.bat" |
|  | if exist "%HOME%\mavenrc\_post.cmd" call "%HOME%\mavenrc\_post.cmd" |
|  | :skipRcPost |
|  |  |
|  | @REM pause the script if MAVEN\_BATCH\_PAUSE is set to 'on' |
|  | if "%MAVEN\_BATCH\_PAUSE%" == "on" pause |
|  |  |
|  | if "%MAVEN\_TERMINATE\_CMD%" == "on" exit %ERROR\_CODE% |
|  |  |
|  | exit /B %ERROR\_CODE% |

## ****pom.xml****

|  |
| --- |
| <?xml version="1.0" encoding="UTF-8"?> |
|  | <project xmlns="http://maven.apache.org/POM/4.0.0" |
|  | xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" |
|  | xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 https://maven.apache.org/xsd/maven-4.0.0.xsd"> |
|  | <modelVersion>4.0.0</modelVersion> |
|  | <parent> |
|  | <groupId>org.springframework.boot</groupId> |
|  | <artifactId>spring-boot-starter-parent</artifactId> |
|  | <version>2.6.1</version> |
|  | <relativePath /> <!-- lookup parent from repository --> |
|  | </parent> |
|  | <groupId>com.SpringBoot</groupId> |
|  | <artifactId>SpringBoot\_aws</artifactId> |
|  | <version>0.0.1-SNAPSHOT</version> |
|  | <name>SpringBoot\_aws</name> |
|  | <description>Demo project for Spring Boot</description> |
|  | <properties> |
|  | <java.version>11</java.version> |
|  | </properties> |
|  | <dependencies> |
|  | <dependency> |
|  | <groupId>org.springframework.boot</groupId> |
|  | <artifactId>spring-boot-starter-thymeleaf</artifactId> |
|  | </dependency> |
|  | <dependency> |
|  | <groupId>org.springframework.boot</groupId> |
|  | <artifactId>spring-boot-starter-web</artifactId> |
|  | </dependency> |
|  |  |
|  | <dependency> |
|  | <groupId>org.springframework.boot</groupId> |
|  | <artifactId>spring-boot-devtools</artifactId> |
|  | <scope>runtime</scope> |
|  | <optional>true</optional> |
|  | </dependency> |
|  | <dependency> |
|  | <groupId>org.springframework.boot</groupId> |
|  | <artifactId>spring-boot-starter-test</artifactId> |
|  | <scope>test</scope> |
|  | </dependency> |
|  | </dependencies> |
|  |  |
|  | <build> |
|  | <plugins> |
|  | <plugin> |
|  | <groupId>org.springframework.boot</groupId> |
|  | <artifactId>spring-boot-maven-plugin</artifactId> |
|  | </plugin> |
|  | <plugin> |
|  | <groupId>org.apache.maven.plugins</groupId> |
|  | <artifactId>maven-compiler-plugin</artifactId> |
|  | <configuration> |
|  | <source>1.8</source> |
|  | <target>1.8</target> |
|  | </configuration> |
|  | </plugin> |
|  | </plugins> |
|  | </build> |
|  |  |
|  | </project> |