Test, Verification and Validation of Software

Ana Cristina Ramada Paiva

Preparation Of Tools For Practical Class

Mutation Testing

Hugo Almeida - up202103394

Orlando Macedo - up202103400

FEUP
Universidade do Porto

12-11-2021

Contents

1 Introduction

The next practical class will explore Mutation Testing methodologies. For that, we will use the **Java** programming language, and a compilation automation tool named **Maven**. As far as the IDE is concerned, you can use whatever you are most familiar with. However, because **IntelliJ** is being used in other subjects as well, we recommend that you also use it for the next practical class.

The installation of the tools will be described for **Windows**, **macOS** and **Linux**, so choose the section that interests you. Within the section itself, we will exemplify firstly the installation of the Java tools, and then the configuration that allows working with Maven.

When you finished this, you should go to the Practice section in order to test your installation.

2 Windows installation

2.1 Installing Java

For the Java installation, we will just redirect to a page with the detailed installation. Oracle does a pretty good job of explaining how to install their finest asset.

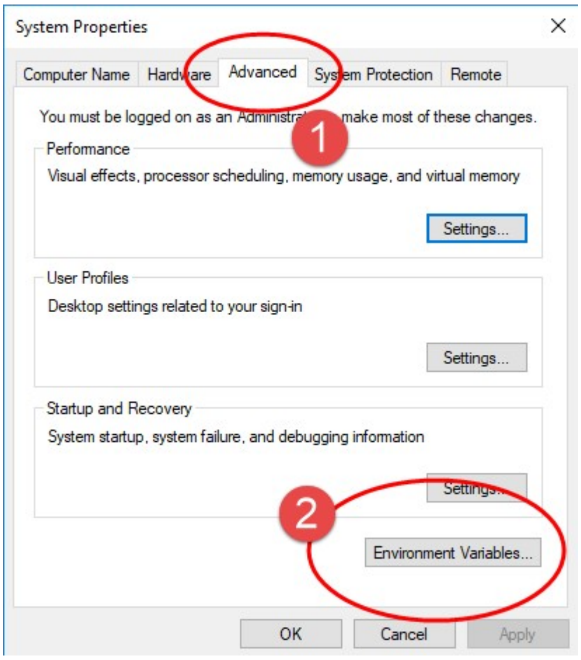
Link: https://www.java.com/en/download/help/windows_manual_download.html

2.2 Working with Environment Variables

As we will probably have to touch on the subject of creating or changing environment variables, below are the steps to do it.

1. To create/update a environment variable the first thing you have to do is navigate to **view advanced system settings**. This can be achieved by pressing the *windows key* and then searching for view advanced system settings.

2. You should see something like this:



You must be logged on as an Administra make most of these changes.

Performance

Visual effects, processor scheduling, memory usage, and virtual memory

User Profiles

Desktop settings related to your sign-in

Startup and Recovery

System startup, system failure, and debugging information

Figure 1: System Properties

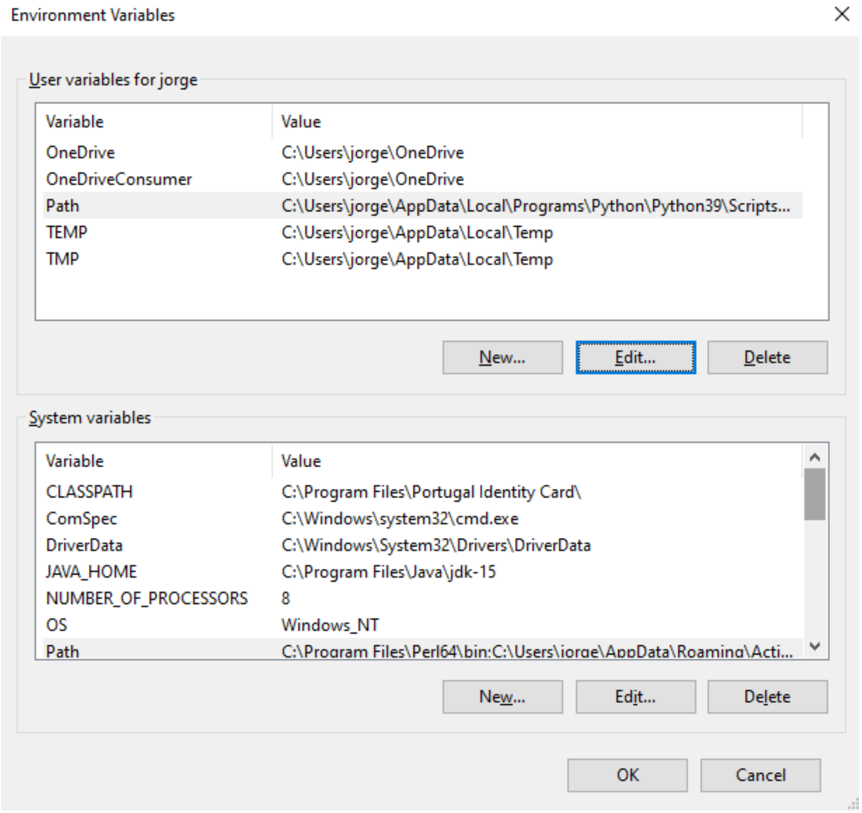
3. Select the **Advanced** tab, if you're not there already, and click in **Environment Variables**.

4. From here you can create environment variables or update them. We will need to create a variable named MAVEN_HOME, and probably, another one named JAVA_HOME.

2.3 Installing Maven

Maven doesn't really have a *.exe* to ease our life. But either way the installation is pretty easy, just follow thoroughly the steps bellow and all will be fine.

1. First we need to check if the JAVA_HOME environment variable is configured. For that purpose we can follow the steps above described.
2. Once in the **Environment Variables** window verify if the **JAVA_HOME** variable is configured.



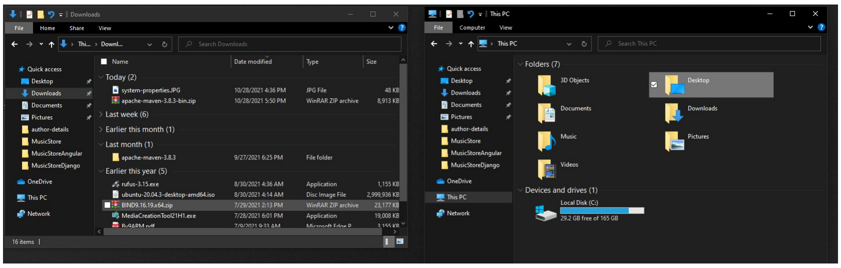
Variable	Value
OneDrive	C:\Users\jorge\OneDrive
OneDriveConsumer	C:\Users\jorge\OneDrive
Path	C:\Users\jorge\AppData\Local\Programs\Python\Python39\Scripts...
TEMP	C:\Users\jorge\AppData\Local\Temp
TMP	C:\Users\jorge\AppData\Local\Temp

Variable	Value
CLASSPATH	C:\Program Files\Portugal Identity Card\
ComSpec	C:\Windows\system32\cmd.exe
DriverData	C:\Windows\System32\Drivers\DriverData
JAVA_HOME	C:\Program Files\Java\jdk-15
NUMBER_OF_PROCESSORS	8
OS	Windows_NT
Path	C:\Program Files\Perl64\bin;C:\Users\jorge\AppData\Roaming\Acti...

Figure 2: JAVA_HOME Environment Variable

3. As it's possible to see, in this case the variable already exists and it's pointing to *C:\Program Files\Java\jdk-15*, which means that java is installed in the *Program Files* directory.
4. If the variable isn't available, let's create it.
   (a) If you don't know where Java is installed, do the following command on cmd c:\> for %i in (java.exe) do echo. %~$PATH:i
   (b) Now click **New** on the **System Variables** section, **not** the **User Variables** section.
   (c) And fill the form with the following values
      - Variable name : JAVA_HOME
      - Variable value : **C:\Program Files\Java\jdk-15** or whatever was the result of the execution of a)
      - Click OK

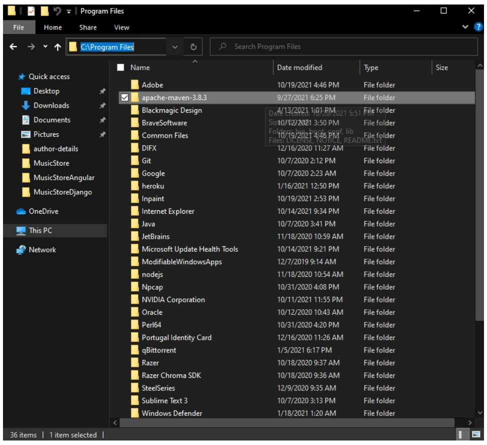
5. Having java configured, it's time to download **Maven**. You can do that at this link. Download the zip binary here.
6. After that, we can unzip the archive and move the unzipped folder to *Program Files*.
7. Don't worry, I will walk you through this process:
   (a) With the unzipped folder, you will open 2 file explorer.



Name	Date modified	Type	Size
Today (2)			
system-properties.JPG	10/28/2021 4:36 PM	JPG File	48 KB
apache-maven-3.8.3-bin.zip	10/28/2021 5:50 PM	WinRAR ZIP archive	8,913 KB
Last week (6)			
Earlier this month (1)			
Last month (1)			
apache-maven-3.8.3	9/27/2021 6:25 PM	File folder	
Earlier this year (5)			
rufus-3.15.exe	8/30/2021 4:36 AM	Application	1,155 KB
ubuntu-20.04.3-desktop-amd64.iso	8/30/2021 4:14 AM	Disc Image File	2,999,936 KB
BIND9.16.19.x64.zip	7/29/2021 2:13 PM	WinRAR ZIP archive	23,177 KB
MediaCreationTool21H1.exe	7/28/2021 6:01 PM	Application	19,008 KB
By9ARM.pdf	7/9/2021 9:33 AM	Microsoft Edge P	3,155 KB

Figure 3: 2 File Explorer

   (b) After, you can just navigate to your Local Disk and click in the **Program Files** folder.
   (c) Lastly, just move (or copy) the apache-maven-*** folder to Program Files. At the end you should have maven in the Program Files folder, similar to the figure 4.



Name	Date modified	Type	Size
Adobe	10/19/2021 4:46 PM	File folder	
☑ apache-maven-3.8.3	9/27/2021 6:25 PM	File folder	
Blackmagic Design	4/13/2021 1:01 PM	File folder	
BraveSoftware	10/12/2021 3:50 PM	File folder	
Common Files	10/19/2021 4:46 PM	File folder	
DIFX	12/16/2020 11:27 AM	File folder	
Git	10/7/2020 2:12 PM	File folder	
Google	10/7/2020 2:23 AM	File folder	
heroku	1/16/2021 12:50 PM	File folder	
Inpaint	10/19/2021 2:53 PM	File folder	
Internet Explorer	10/14/2021 9:34 PM	File folder	
Java	10/7/2020 3:41 PM	File folder	
JetBrains	11/18/2020 10:59 AM	File folder	
Microsoft Update Health Tools	10/14/2021 9:21 PM	File folder	
ModifiableWindowsApps	12/7/2019 9:14 AM	File folder	
nodejs	11/18/2020 10:54 AM	File folder	
Npcap	10/31/2020 4:08 PM	File folder	
NVIDIA Corporation	10/11/2021 11:55 PM	File folder	
Oracle	10/12/2020 10:43 AM	File folder	
Perl64	10/31/2020 4:20 PM	File folder	
Portugal Identity Card	12/16/2020 11:26 AM	File folder	
qBittorrent	1/5/2021 6:17 PM	File folder	
Razer	10/18/2020 9:37 AM	File folder	
Razer Chroma SDK	10/18/2020 9:36 AM	File folder	
SteelSeries	12/9/2020 9:35 AM	File folder	
Sublime Text 3	10/7/2020 3:13 PM	File folder	
Windows Defender	1/18/2021 1:20 AM	File folder	

Figure 4: Maven in Program Files

8. We will need to go once again to the **Environment Variables** and add the MAVEN_HOME variable.
9. In the **System Variables** section click **New**. Put the following values on the form shown.
   - Variable name : MAVEN_HOME
   - Variable value : **C:\Program Files\apache-maven-3.8.3**
   - Be careful that the version of maven might differ.
   - Click OK.
10. The last step is to add the bin directory to PATH.
11. Search **Path** or **PATH** in your System Variables.

12. When you find it, click **Edit**. After the window is shown, click **New** and paste the following text %MAVEN_HOME%\bin.
13. Well, you can finally check maven on cmd or powershell. Just open your favourite and execute c:\>mvn -version.
14. Congratulations, you just installed Maven! Easy right?

3 Installation on MacOS

3.1 Installing Java

Since Java is no longer installed with *MacOS*, it is necessary to install it. For this class we will use the *Open JDK 11* version, which is an open source variant of the *JRE* and *JDK*.

To facilitate the installation we will use a Package Manager for MacOS, *Homebrew*. For your installation just use this command:

1. $ /bin/bash -c ”$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)”

With that done, we proceed to install Java:

1. $ brew tap AdoptOpenJDK/openjdk
2. $ brew install –cask adoptopenjdk11
3. Check if the installation succeeded with **$ /usr/libexec/java_home -V**. It should show a Java Virtual Machine with the version 11.

Finally, we'll set the required JAVA_HOME environment variable later on:

1. Edit the file */.zshrc* using a text editor of your choice. For Vim, you should use $ vim /.zshrc
2. Add the followin content to the file: export JAVA_HOME=$(/usr/libexec/java_home -v11)
3. Source the file with $ source /.zshrc
4. Make sure the variable is set with $ echo $JAVA_HOME

3.2 Installing Maven

To install Maven, use Homebrew again:

1. $ brew install maven
2. Verify if the installation was successful using **mvn -version**

4 Linux Installation (Ubuntu)

4.1 Installing Java

The easiest option to install *Java* is to use the version bundled with *Ubuntu*. By default, *Ubuntu 20.04* includes *Open JDK 11*, which is an open source variant of the *JRE* and *JDK*.

Follow the following steps to install **Java** on your machine.

1. $ sudo apt update
2. $ sudo apt install default-jre
3. Check if the installation succeeded with **java –version**
4. $ sudo apt install default-jdk
5. Once again, verify if the installation was successful using **javac –version**
6. Congratulations, you're done with java, lets go to the maven installation.

4.2 Installing Maven

Now that we have installed the Java JDK we are able to install Maven. A few things to keep in mind, Maven needs to have a JAVA_HOME environment variable that points to the location of the Java JDK. However, we can avoid this if Java itself can be found from the PATH variable.

Actually, it's better if we configure the JAVA_HOME environment variable, so that's what we will do.

If you want to go rogue, you can skip to step 11.

1. First you need to know where java was installed
2. For that purpose, follow the following steps
   - $ whereis java
   - $ ls -lah /usr/bin/java, this path might be different from yours. Adjust to the appropriate.
   - $ ls -lah /etc/alternatives/java
   - And voila, you found where java was hidden.
3. Now copy the path until the bin directory, to be like */usr/lib/jvm/java-11-openjdk-amd64*.
4. Next you need to write a new line to the **/etc/environment** file. For this you can use nano, vim or whatever. For simplicity purposes we will use gedit.
5. $ sudo gedit /etc/environment
6. Add a new line to the file with the following content
7. JAVA_HOME="/usr/lib/jvm/java-11-openjdk-amd64"
8. Keep in mind that the path to your java directory can be different from this. Use the one that was found previously.
9. Save and exit. After that use $ source /etc/environment. Now, you should be able to use $ echo $JAVA_HOME and see the java directory.
10. Now for the easy part, installing maven.
11. Download the bin at this link, you can download the tar or zip version. We will use tar.

12. Once the archive was downloaded, move it to the opt directory using $ sudo mv apache-maven-?? /opt/
13. Change directory $ cd /opt/
14. And extract the archive $ sudo tar -xzvf apache-maven-??-bin.tar.gz
15. You can remove the archive with $ sudo rm apache-maven-??-bin.tar.gz
16. The last thing to do is to add the bin directory of maven to the PATH variable.
17. Let's open the /.bashrc file and add at the bottom the following command **export PATH=/opt/apache-maven-***/bin:$PATH**. Don't forget to replace the ?? for the actual installed version of maven.
18. If you're with problems opening the file just use $ sudo gedit /.bashrc
19. Save and exit. After that, just source the bashrc file with $ source /.bashrc.
20. And finally! You should be able to execute $ mvn –version.
21. Congratulations, you did well today!

5 Practice

In the directory of this document you can find the project **mutation_testing-experiment** containing a simple example in order to test the use of the Mutation Testing tool **PITest.**

Please follow the steps done in the following demonstration.

At the end, you can check if the practical exercises for Friday are running as well. These are located on the folder **Exercises**.