

WA2271 Continuous Integration with Jenkins-CI

Classroom Setup Guide

Web Age Solutions

Table of Contents

Part 1 - Software Provided.....	3
Part 2 - Software for Setup # 1: VM_WA2271.....	3
Part 3 - Minimum Hardware Requirements for Setup # 1.....	3
Part 4 - Minimum Software Requirements for Setup # 1.....	3
Part 5 - Instructions for Setup # 1.....	4
Part 6 - Software for Setup # 2: WA2271_REL_9_0.....	5
Part 7 - Software Provided.....	6
Part 8 - Minimum Software Requirements for Setup # 2.....	6
Part 9 - Instructions for Setup # 2.....	7
Part 10 - Installing JDK 11.0.8 - 64 bit.....	7
Part 11 - Installing Eclipse-IDE-2020-03.....	12
Part 12 - Installing Maven 3.6.3.....	15
Part 13 - Install Jenkins Jenkins-2.443-Windows.....	16
Part 14 - Installing Git 2.26.....	26
Part 15 - Installing Node.js 10.16.0.....	29
Part 16 - Verification for Setups # 1 and # 2.....	31
Part 17 - Summary.....	31

Part 1 - Software Provided

List of ZIP files required for this course and used in next steps on this document:

WA2271_REL_9_0.zip

VM_WA2271.zip

Send an email to **support@webagesolutions.com** in order to obtain a copy of the software for this course if you haven't receive it yet.

All other software listed under Minimum Software Requirements is either commercially licensed software that you must provide or software that is freely available off the Internet.

Part 2 - Software for Setup # 1: VM_WA2271

- **VM_WA2271.zip**

Part 3 - Minimum Hardware Requirements for Setup # 1

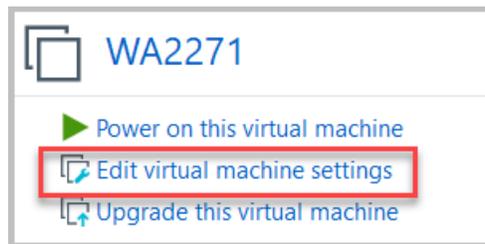
- 64-bit x86 CPU that supports hardware virtualization (Intel-VT or AMD-V).
- Hardware virtualization enabled in the BIOS.
- 12 GB RAM
- 20 GB in the hard disk
- Internet access

Part 4 - Minimum Software Requirements for Setup # 1

- VMWare player

Part 5 - Instructions for Setup # 1

- __1. Make sure you have VMWare player or VMWare workstation installed in your computer.
- __2. Extract the **VM_WA2271.zip** file in your hard disk, it will extract all the files from all the zips.
- __3. Open VMWare.
- __4. Click Open a Virtual Machine.
- __5. Select the **VM_WA2271** VM image.
- __6. Click **edit virtual machine settings**



- __7. Increase the memory to 6 GB RAM and 2 cores.

Device	Summary
 Memory	6 GB
 Processors	2

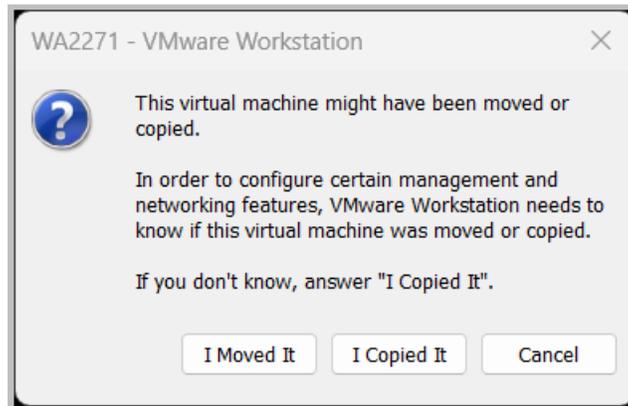
- __8. Click OK.
- __9. Open the folder where the zip was extracted.
- __10. Right click on the *.vmx file and select **Send to** → **Desktop (create a shortcut)** to create a shortcut in the desktop and then rename the shortcut as:

- **VM_WA2271**

Make sure to rename the shortcut because this name is used during the labs.

- __11. Using the shortcut in the desktop, start the VM.

__12. Click **I moved it**.



__13. Login using **wasadmin/wasadmin** as credentials.

__14. Make sure the VM has internet access.

__15. Close the VM.

Part 6 - Software for Setup # 2: WA2271_REL_9_0

From here to the rest of the document you will find specific instructions to create a new Windows VM, name the VM as follows:

- **WA2271_REL_9_0**

You can opt to install the software directly on the computer, if you do that then make sure there is no software installed other than windows in that computer.

If you are not creating a VM then you need to notify the instructor that the software will be installed directly in the computer for the students.

Part 7 - Software Provided

You will receive the following file:

WA2271_REL_9_0.zip

Part 8 - Minimum Software Requirements for Setup # 2

- Windows 8.1
- Chrome latest
- Firefox latest
- Adobe Acrobat Reader
- Zip extraction utility
- JDK 11.0.8 - 64 bits *
- Eclipse IDE-2020-03 *
- Maven 3.6.3 *
- Git 2.26.0 *
- Jenkins-2.443 *
- Node 10.16.0 *

* - indicates software provided as part of the courseware.

Part 9 - Instructions for Setup # 2

__1. Make sure the account that you are using to install the software has administrative privileges and the student using this machine will have the same rights.

__2. Extract the **ZIP** file directly to **C:**

__3. Review that the following folders were created:

- **C:\LabFiles**
- **C:\Software\apache-maven-3.6.3**
- **C:\Software\Jenkins-2.443**

__4. Review that the following files were created:

- **C:\Software\Eclipse-IDE-2020-03-eclipse-inst-win64.exe**
- **C:\Software\Git-2.26.0-64-bit.exe**
- **C:\Software\jdk-11.0.8_windows-x64_bin.exe**
- **C:\Software\node-v10.16.0-x64.msi**

__5. Create the **C:\Workspace** folder.

__6. Make sure **C:\LabFiles** and **C:\Software** folders are not ready only.

__7. Continue the installation for Setup # 2 from here to the end of this document.

Part 10 - Installing JDK 11.0.8 - 64 bit

__1. Make sure there is no previous Java version already installed on the system. You can check this by using the Windows “Add/Remove Programs” utility. For the best compatibility with the labs it is suggested that all previous versions of Java be uninstalled

before proceeding with these instructions. If this is an issue, please contact the setup support person for the course.

__2. From the **C:\Software** directory run the following file:

```
jdk-11.0.8_windows-x64_bin.exe
```

Note: If using prompted by a security prompt allow the installation to continue.

__3. When the Welcome page of the setup appears, press the **Next** button.

__4. Leave the defaults for installation location and options, and press the **Next** button.

The installation will begin installing files. Wait until the software is completely installed.

__5. Click **Close**.

Set the Environment variables

__1. Open a Command Prompt. You can do this with '**Start → Programs → Accessories → Command Prompt**'.

__2. Use the 'cd' command to attempt to switch to the following directory. This will verify the presence of a directory used later so make sure you do not get any errors about not being able to “find the path specified”.

```
cd C:\Progra~1\Java\jdk-11.0.8
```

Note: The installation directory may be slightly different depending your operating system. You may need to use the following directory instead of the one listed above:

```
C:\Progra~2\Java\jdk-11.0.8
```

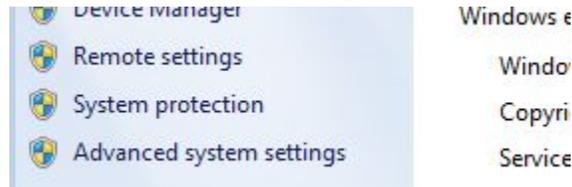
Some of the remaining steps will use the slightly different directory.

__3. Make sure you can reach the java folder and remember the value entered because you will use this value in the following steps.

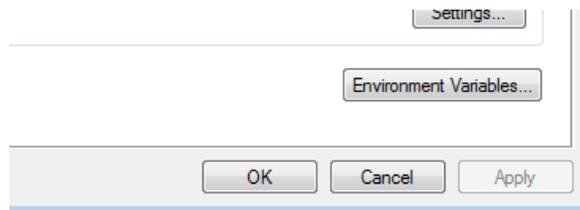
__4. Close the command prompt window.

__5. Open the system properties. The way to do this varies slightly by Windows version. The easiest way is often to open a File Explorer window, right click on the 'This PC' shortcut and select '**Properties**'

__6. Click on **Advanced system settings**.



__7. The system will display the **System Properties** dialog. Select the **Advanced** tab and click **Environment Variables**.



__8. Under the **System Variables** list, click the **New** button.

__9. As Variable name enter:

JAVA_HOME

__10. As Variable value enter the following. This should be the value you verified before.

C:\Progra~1\Java\jdk-11.0.8

__11. Click **OK** to create the variable.

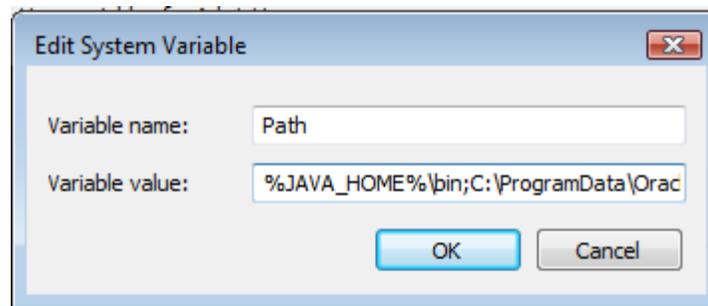


__12. From the *System Variables* list, select **Path** and click **Edit**.

__13. At the beginning of the line enter the following. Make sure to include the semi-colon on the end:

```
%JAVA_HOME%\bin;
```

__14. Click **OK**.



__15. Click **OK** to close the *Environment Variables* window.

__16. Click **OK** to close the *System Properties* window.

Verification

__1. Open a Windows command prompt. You can do this by selecting '**Start -> Run**', entering '**cmd**', and then pressing the **OK** button. Make sure it is a new command prompt and not one open previously.

__2. Enter the following command:

```
echo %PATH%
```

Make sure you see the Java 'bin' directory listed at the beginning.

__3. Enter the following command:

```
java -version
```

Make sure you see the response shown below.

```
C:\Users\wasadmin>java -version
java version "11.0.8" 2020-07-14 LTS
Java(TM) SE Runtime Environment 18.9 (build 11.0.8+10-LTS)
Java HotSpot(TM) 64-Bit Server VM 18.9 (build 11.0.8+10-LTS, mixed mode)
```

Troubleshooting: If you get an error message means that your Environment variable was incorrectly entered, go back and fix the values.

__4. Enter the following command:

```
javac
```

Verify that you get the options to run the Java compiler:

```
C:\>javac
Usage: javac <options> <source files>
where possible options include:
  @<filename>           Read options and filenames from file
  -Akey[=value]         Options to pass to annotation proces
  --add-modules <module>(,<module>)*
                        Root modules to resolve in addition to the initial modules,
                        on the module path if <module> is ALL-MODULE-PATH.
  boot-class-path <path>  bootclasspath <path>
```

__5. Enter the following command:

```
java -XshowSettings:all 2>&1 | findstr /c:"sun.arch.data.model"
```

Verify that it displays the correct value of 64. This indicates that it is 64-bit java that is installed.

__6. Close the command prompt window and any extra windows that are open.

You have completed Java installation.

Part 11 - Installing Eclipse-IDE-2020-03

__1. Open C:\Software\

__2. Double click in this file to install eclipse:

Eclipse-IDE-2020-03-eclipse-inst-win64.exe

__3. Select **Eclipse IDE for Enterprise Java and Web Developers**.



__4. Enter the **Installation folder** as C:\Software



Note that Java version may vary.

__5. Click **Install**.

__6. Click **Accept Now**.

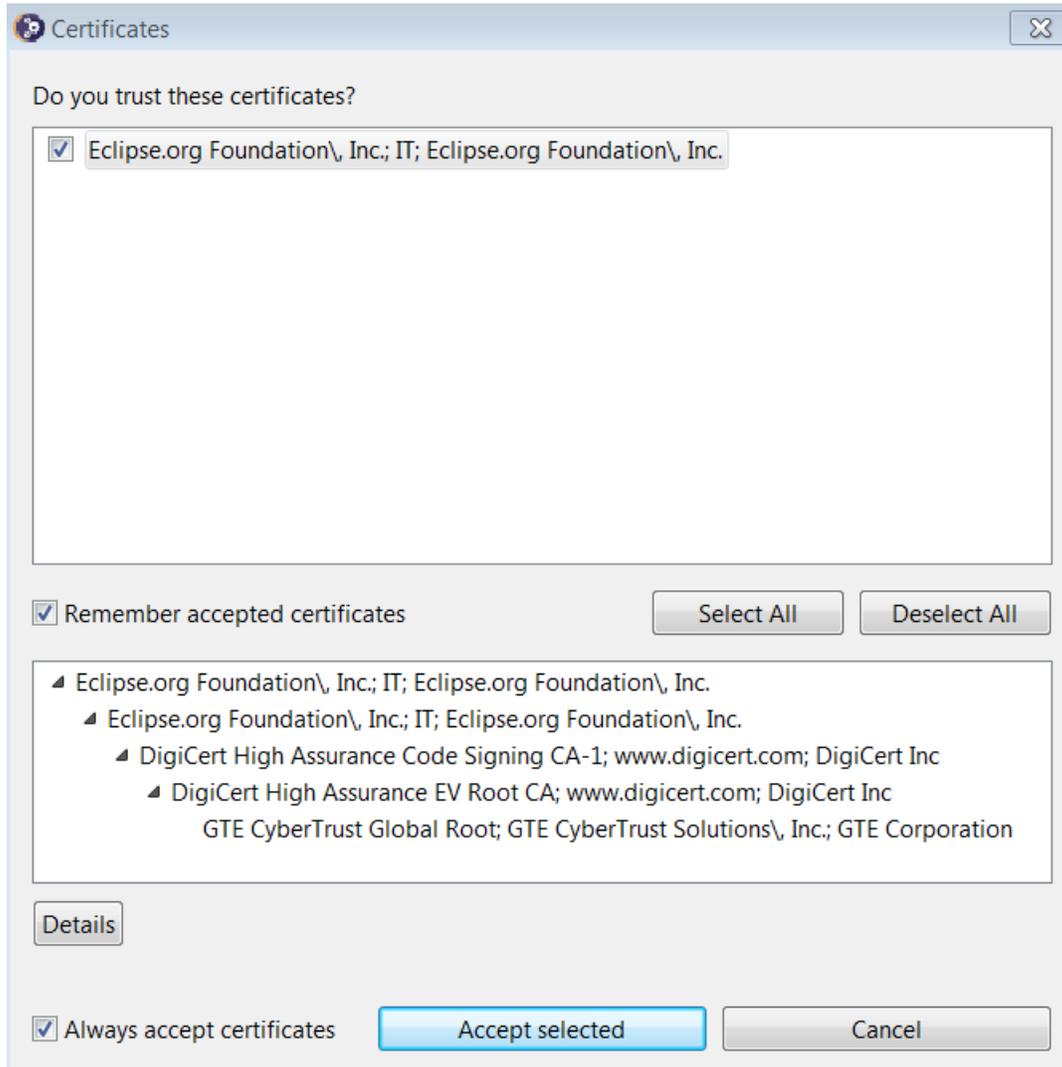
The installation will start and may take a while.

__7. In the Certificates page, click **Select All**.

__8. You may see a pop-up message about the installation is taking longer. Just close that pop-up.

__9. Check the boxes for **Remember accepted certificates**.

__10. Check the boxes for **Always accept certificates**.



__11. Click **Accept selected** to continue with the installation.

__12. Finally it will finish installing eclipse. Click **Launch**.



Eclipse IDE for Enterprise Java Developers

[details](#)

Tools for developers working with Java and Web applications, including a Java IDE, tools for Web Services, JPA and Data Tools, JavaServer Pages and Faces, Mylyn, Maven and Gradle, Git, and more.

Java 1.8+ VM	C:\Progra~1\Java\jdk-11.0.8 (Current)	
Installation Folder	C:\Software	
	<input checked="" type="checkbox"/> create start menu entry	
	<input checked="" type="checkbox"/> create desktop shortcut	
<div style="text-align: center;"></div>		

Eclipse will start.



__13. Change the workspace directory to **C:\Workspace** and click **Launch**.

__14. Eclipse will open showing the Welcome page. Close the page by clicking on the X in the tab.

__15. From the menu, select **File > Exit** to close Eclipse.

You have completed Eclipse installation.

Part 12 - Installing Maven 3.6.3

The following steps are based on Windows 7, other Windows versions instructions may vary.

__1. In the Windows Start Menu, right-click on the **Computer** link in the right-hand side of the Start panel, and then select **Properties**.

__2. Click on **Advanced system settings**.

__3. The system will display the **System Properties** dialog. Select the **Advanced** tab and click **Environment Variables**.

__4. Verify there is a **JAVA_HOME** variable.

__5. In the 'System Variables' panel, locate the entry for '**Path**' and double-click on it.

__6. Add the following to the **end** of the **Variable Value** field (including the semi-colon)

```
;C:\Software\apache-maven-3.6.3\bin
```

__7. Click **OK** on the variable editor dialog.

__8. Click **OK** on the **Environment Variables** dialog.

__9. Click **OK** in the **System Properties** dialog.

__10. Open a command prompt window.

__11. In the command window, type:

```
mvn -version
```

__12. Verify the version is 3.6.3 as shown below:

```
C:\Users\wasadmin>mvn -version
Apache Maven 3.6.3 (cecedd343002696d0abb50b321
Maven home: C:\Software\apache-maven-3.6.3\bin
Java version: 1.8.0_45, vendor: Oracle Corpora
Default locale: en_US, platform encoding: Cp12
OS name: "windows 7", version: "6.1", arch: "x
```

__13. Close all.

Part 13 - Install Jenkins Jenkins-2.443-Windows

__1. Open a command prompt window and ensure that the Java JDK is installed.

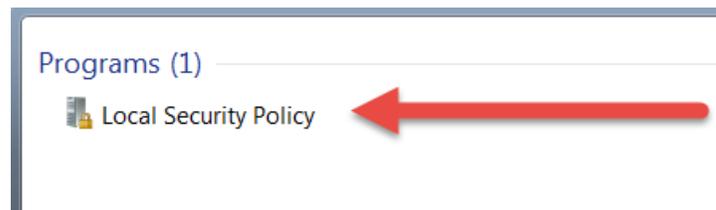
```
java -version
```

__2. If you are using a computer where possibly Jenkins was used before then delete the following folders in case they exist:

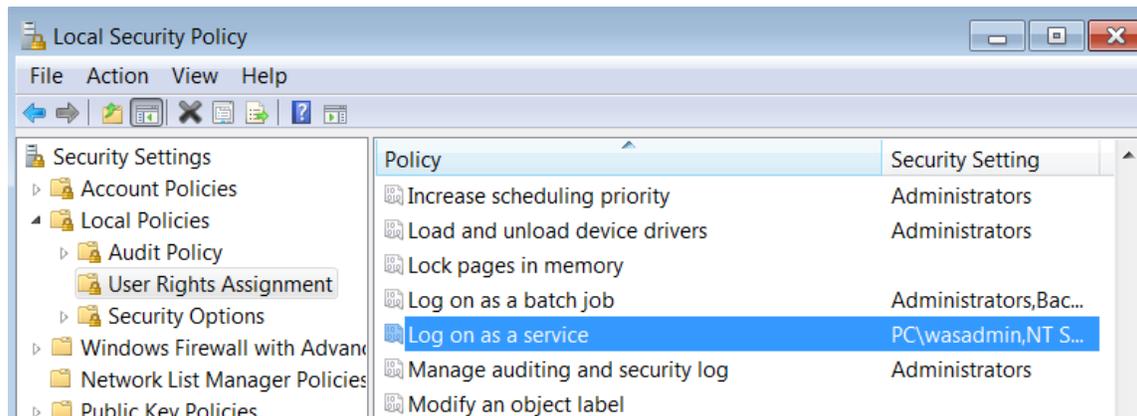
```
C:\Program Files\Jenkins  
C:\Program Files(x86)\Jenkins  
C:\Users\\.m2
```

__3. Make Chrome your default browser and make sure the latest Chrome version is installed.

__4. Before installing Jenkins you need to make sure that the user using Jenkins have the privileges to start a Service. To do that, from the start menu, start typing **Local Security** and the **Local Security Policy** will show up. Click on it. On **Windows 8/10** search for **secpol.msc** and click on it.



__5. Expand **Local Policies > User Rights Assignment** and double click on **Log on as a service** on the right panel.

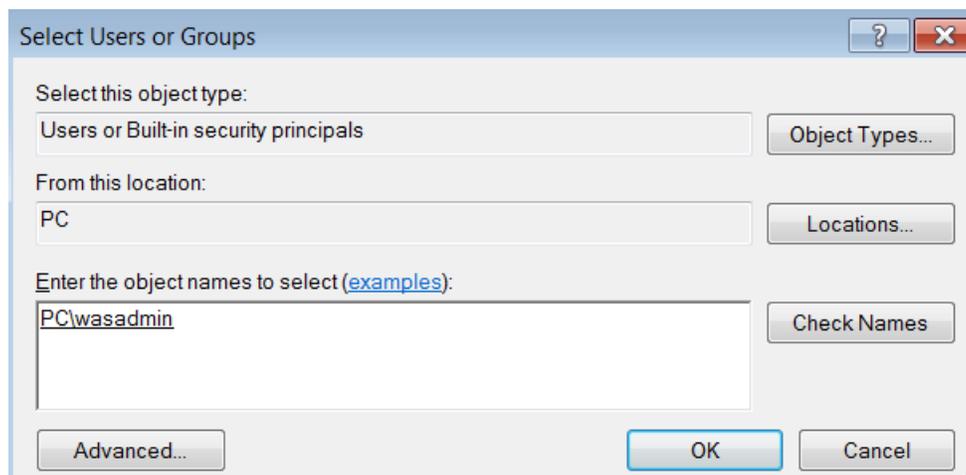


__6. Click **Add User or Group**.

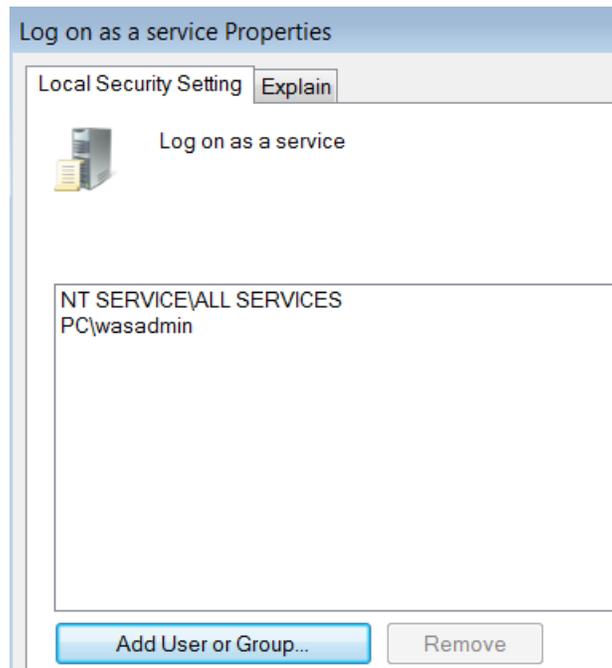
__7. Enter the name of the user that the students will use to do the Labs.

__8. Click **Check Names** to verify is the correct user.

__9. Click **OK**.



__10. You will see your user listed in this example 'wasadmin'. Click **OK**.



__ 11. In Windows Explorer, navigate to:

C:\Software\Jenkins-2.443

__ 12. Double-click on:

jenkins.msi

The installer will show the initial dialog.

__ 13. Click **Next**.

__ 14. On the **Destination Folder** panel, leave the defaults and click **Next**.

__ 15. Select **Run service as local or domain user** and enter your **user / password** and click **Test Credentials**. Make sure the test passes.

Service Logon Credentials **Jenkins** 

Enter service credentials for the service.

Jenkins 2.249.1 installs and runs as an independent Windows service. To operate in this manner, you must supply the user account credentials for Jenkins 2.249.1 to run successfully.

Logon Type:

Run service as LocalSystem (not recommended)

Run service as local or domain user:

Account:

Password:



__16. Click **Next**.

__17. Click **Test Port** and make sure it works fine.

Port Number (1-65535):



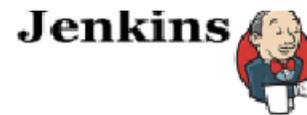
__18. Click **Next**.

__19. Leave the default **Java home directory (JDK or JRE)** and click **Next**.

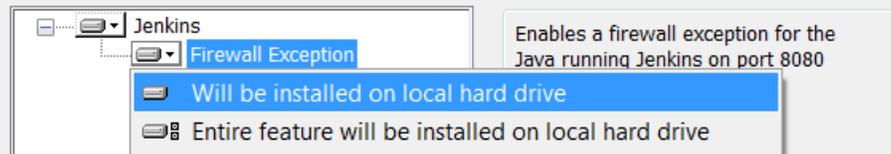
__20. Click on **Firewall Exception** and select **Will be installed on local hard drive**.

Custom Setup

Select the way you want features to be installed.



Click the icons in the tree below to change the way features will be installed.



- __21. Click **Next**.
- __22. On the **Ready to Install...** page, click **Install**.
- __23. Windows may show a security dialog. If it does, click **Yes**.
- __24. In the final dialog panel, click **Finish**.

The installer will open the default browser window to the Jenkins home page. The page shows the location of a file where you can find the initial administration password, and also a text box to enter the password into.

- __25. Open Mozilla and enter **localhost:8080**

Unlock Jenkins

To ensure Jenkins is securely set up by the administrator, a password has been written to the log (**not sure where to find it?**) and this file on the server:

```
C:\Users\wasadmin\AppData\Local\Jenkins.jenkins\secrets\initialAdminPassword
```

Please copy the password from either location and paste it below.

Administrator password

Continue

__26. Open the indicated file with an editor such as Notepad, and copy the password to the clipboard with Ctrl-C.

__27. Paste the password into the **Administrator Password** box and click **Continue**.

__28. If you are prompted to save the password just close that window.

__29. Click on **Install Suggested Plugins**.

Customize Jenkins

Plugins extend Jenkins with additional features to support many different needs.

Install suggested plugins

Install plugins the Jenkins community finds most useful.

Select plugins to install

Select and install plugins most suitable for your needs.

Getting Started will begin. Wait until is done.

Getting Started

✓ Folders Plugin	✓ OWASP Markup Formatter Plugin	✓ build timeout plugin	✓ Credentials Binding Plugin	** Jenkins C
✓ Timestamper	✓ Workspace Cleanup Plugin	✓ Ant Plugin	✓ Gradle Plugin	** Jenkins G
✓ Pipeline	⌛ GitHub Organization Folder Plugin	✓ Pipeline: Stage View Plugin	✓ Git plugin	** Pipeline:
⌛ Subversion Plug-in	⌛ SSH Slaves plugin	✓ Matrix Authorization	✓ PAM Authentication	Libraries

** Branch AP
** Pipeline:
** Durable T
** Pipeline:
Processes
** Pipeline:
** Pipeline:

In case a plugin failed to be installed, you can retry or click Continue.

__30. In the **Create First Admin User** screen. Enter the following fields:

Username: wasadmin

Password: wasadmin

Confirm Password: wasadmin

Full name: Administrator

E-mail address: wasadmin@wasadmin.com

__31. When the input looks like below, click **Save and Continue**.

Create First Admin User

Username:

Password:

Confirm password:

Full name:

E-mail address:

__32. Instance Configuration page will open, just click **Save and Finish**.

Instance Configuration

Jenkins URL:

The Jenkins URL is used to provide the root URL for absolute links to various Jenkins resources. That means this value is required for proper operation of many Jenkins features including email notifications, PR status updates, and the `BUILD_URL` environment variable provided to build steps.

The proposed default value shown is not saved yet and is generated from the current request, if possible. The best practice is to set this value to the URL that users are expected to use. This will avoid confusion when sharing or viewing links.

150.3

Not now

Save and Finish

__33. If you are prompted to save the password just close that window.

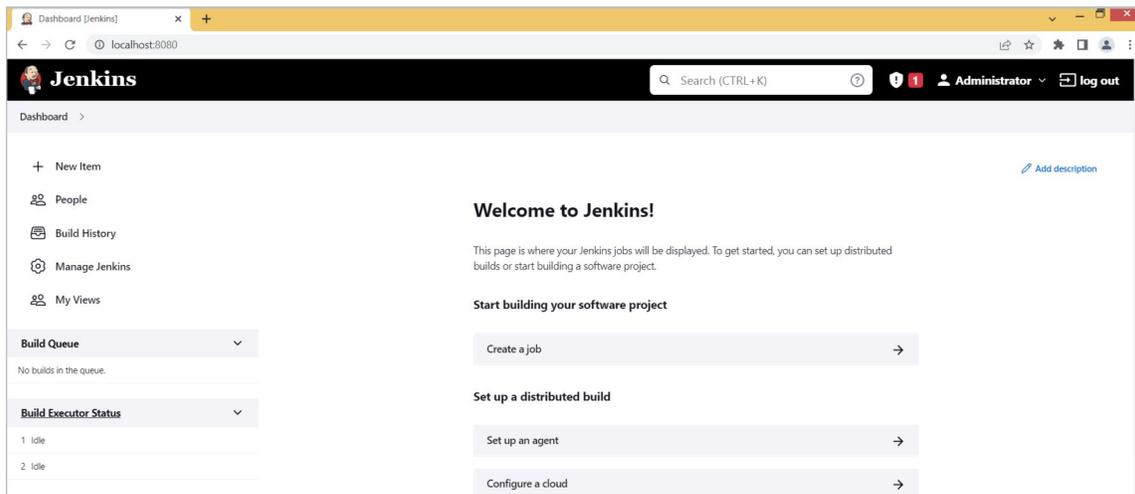
__34. You will see that Jenkins is ready. Click **Start using Jenkins**.

Jenkins is ready!

Your Jenkins setup is complete.

Start using Jenkins

Jenkins will open.



__35. Jenkins installation is complete. Close the browser.

__36. Edit the following file with notepad or similar text editor:

C:\Program Files\Jenkins\Jenkins.xml

__37. Find this line:

```
<arguments>-Xrs -Xmx256m -Dhudson.lifecycle=udson.lifecycle.WindowsServiceLifecycle -jar  
"C:\Program Files\Jenkins\jenkins.war" --httpPort=8080 --webroot="%LocalAppData%\Jenkins\  
war"</arguments>
```

__38. Insert the following text **before** the first "-Dhudson ...":

```
-Dhudson.plugins.git.GitSCM.ALLOW_LOCAL_CHECKOUT=true
```

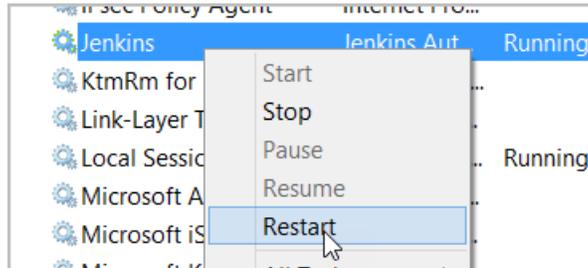
__39. The updated line will look like this:

```
<arguments>-Xrs -Xmx256m -Dhudson.plugins.git.GitSCM.ALLOW_LOCAL_CHECKOUT=true
-Dhudson.lifecycle=udson.lifecycle.WindowsServiceLifecycle -jar "C:\Program
Files\Jenkins\jenkins.war" --httpPort=8080 --webroot="%LocalAppData%\Jenkins\
war"</arguments>
```

__40. Save and close the file.

__41. Open Services.

__42. Restart Jenkins.



__43. Open Mozilla and go to Jenkins homepage:

http://localhost:8080/

__44. Login using **wasadmin** for user and password.

__45. Make sure login Jenkins is successful and you can see the home page.



__46. Close all.

Part 14 - Installing Git 2.26

IMPORTANT: Setup is easy but you need to make sure you do the change in Step 8.

__1. From the **C:\Software** directory run the following file:

Git-2.26.0-64-bit.exe

__2. You may need to allow the program to run.

__3. In the Information page, click **Next**.

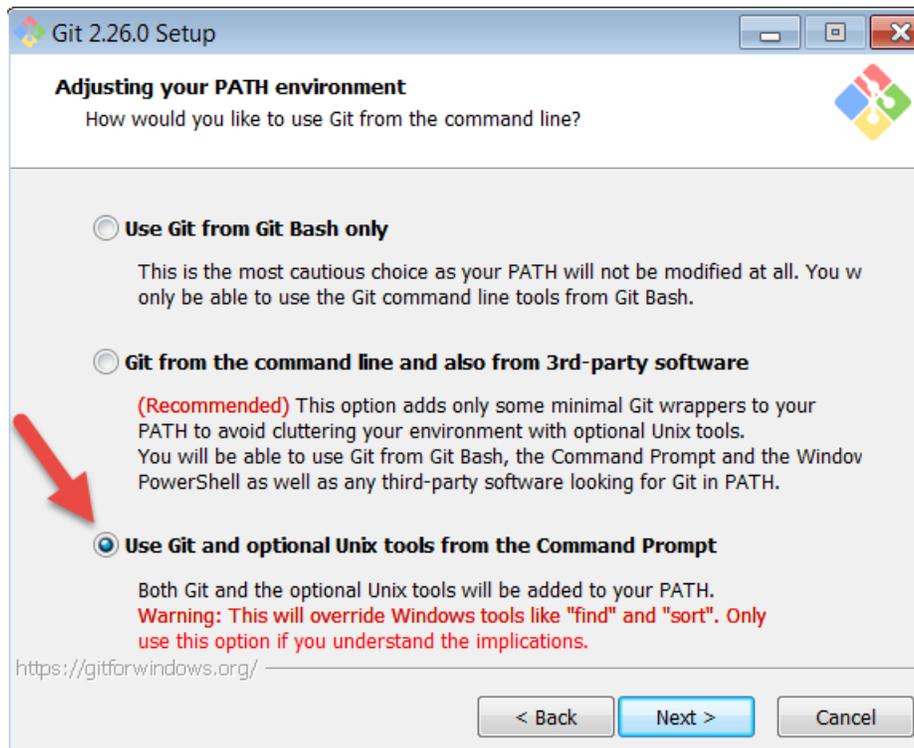
__4. In the Select Destination Location page, leave defaults and click **Next**.

__5. In the Select Components page, leave defaults and click **Next**.

__6. In the Select Start Menu Folder page, leave defaults and click **Next**.

__7. In the Choosing the default editor used by Git page, leave defaults and click **Next**.

__8. In the Adjusting your PATH environments page, select **Use Git and Optional Unix tools from the Windows Command Prompt**.



- __ 9. Make sure you select the 3rd option as shown above and then click **Next**.
- __ 10. In the Choosing HTTP transport backend page, leave defaults and click **Next**.
- __ 11. In the Configuring the terminal emulator to use with Git Bash page, leave defaults and click **Next**.
- __ 12. In the Configuring the line ending conversions page, leave defaults and click **Next**.
- __ 13. Finally, in the Configuring extra options page, leave defaults and click **Install**.

Configuring extra options

Which features would you like to enable?



Enable file system caching

File system data will be read in bulk and cached in memory for certain operations ("core.fscache" is set to "true"). This provides a significant performance boost.

Enable Git Credential Manager

The [Git Credential Manager for Windows](#) provides secure Git credential storage for Windows, most notably multi-factor authentication support for Visual Studio Team Services and GitHub. (requires .NET framework v4.5.1 or later).

Enable symbolic links

Enable [symbolic links](#) (requires the SeCreateSymbolicLink permission). Please note that existing repositories are unaffected by this setting.

<https://gitforwindows.org/>

< Back Install Cancel

__14. Check the box for **Launch Git Bash** and Click **Next**.

Completing the Git Setup Wizard

Setup has finished installing Git on your computer. The application may be launched by selecting the installed shortcuts.

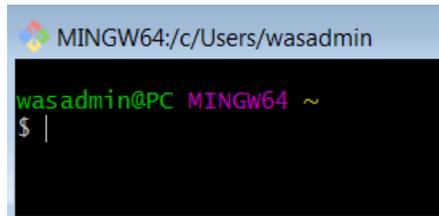
Click Finish to exit Setup.



- Launch Git Bash
- View Release Notes

Next >

__15. You will see a Terminal Git window like below.



__16. Close all.

You have completed GIT installation.

Part 15 - Installing Node.js 10.16.0

__1. Open **C:\Software**

__2. Double click on **node-v10.16.0-x64.msi** to begin installation.

__3. In the Welcome page, click **Next**.

__4. Check **I accept the terms...** and click **Next**.

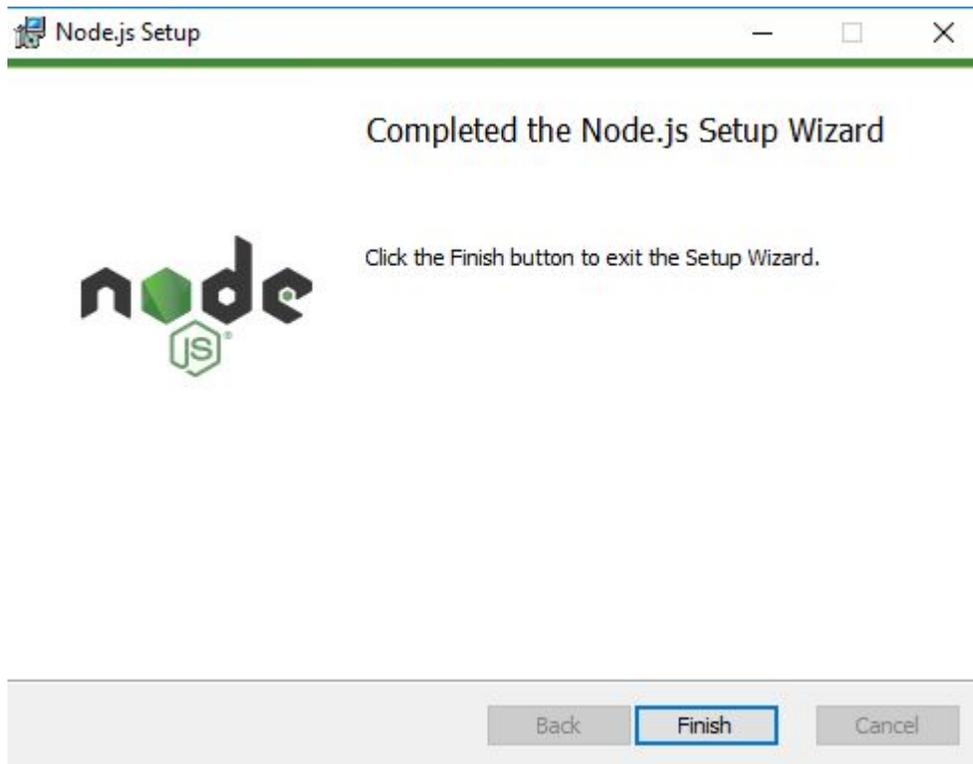
__5. Accept default destination folder and click **Next**.

__6. Accept default Custom Setup selection and click **Next**.

__7. Click **Install**.

__8. A popup window may open asking to install the software. Click Yes to continue.

__9. Click **Finish**.



Installation verification of Node.js

__1. Open a command prompt window.

__2. Enter the command:

```
node --version
```

__3. Make sure that you see 10.16.0 as the output.

__4. Close all.

Part 16 - Verification for Setups # 1 and # 2

__1. Make sure the following shortcuts are located in the desktop:

- **WA2271_REL_9_0**
- **VM_WA2271**

__2. Stop all VMs.

Part 17 - Summary

You have successfully installed the software for this course!

If you have any question please contact us by email at support@webagesolutions.com

From US and Canada call: 1-877-812-8887 ext. 26

International call: 416-406-3994 ext. 26