

**WA2874 Microservices
Development Bootcamp with
immersive project (using Spring
Boot and Docker)**

Classroom Setup Guide

Web Age Solutions Inc.

Table of Contents

Part 1 - Minimum Hardware Requirements to host the VMs.....	3
Part 2 - Minimum Software Requirements.....	3
Part 3 - Software Provided.....	3
Part 4 - Setup # 1: PROJECT_VM_MSD_2.0_2020-10-01.....	4
Part 5 - Setup # 2: VM_WA2874_REL_1_0.....	5
Part 6 - Create a Docker Account and Obtain an Access Token [This part needs to be done by the Students or by the setup person].....	6
Part 7 - Setup # 3. WA2874_REL_2_0.....	8
Part 8 - Minimum Hardware Requirements.....	8
Part 9 - Minimum Software Requirements.....	8
Part 10 - Software Provided.....	9
Part 11 - Instructions.....	9
Part 12 - Privileges to Users.....	10
Part 13 - Installing JDK 11.0.8 - 64 bit.....	13
Part 14 - Installing Eclipse-IDE-2020-03.....	17
Part 15 - Installing Maven 3.6.3.....	20
Part 16 - Installing Git 2.26.....	21
Part 17 - Installing Postman 7.21.....	23
Part 18 - Installing Gradle 6.6.1.....	24
Part 19 - Verification.....	27
Part 20 - Summary.....	27

Part 1 - Minimum Hardware Requirements to host the VMs

- 64-bit x86 CPU that supports hardware virtualization (Intel-VT or AMD-V)
- Hardware virtualization enabled in the BIOS
- 2 cores
- 12 GB RAM, 16 GB recommended
- 80 GB Hard Disk

Part 2 - Minimum Software Requirements

- Windows 8.1-64 bits or later
- 7-zip
- VMWare Workstation Player 14 or later

Part 3 - Software Provided

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

- **WA2874_REL_2_0**
- **VM_WA2874_REL_1_0**
- **PROJECT_VM_2.0_2020-10-01**

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 4 - Setup # 1: PROJECT_VM_MSD_2.0_2020-10-01

__1. Download the following files:

- **PROJECT_VM_2.0_2020-10-01.7z.001**
- **PROJECT_VM_2.0_2020-10-01.7z.002**
- **PROJECT_VM_2.0_2020-10-01.7z.003**
- **PROJECT_VM_2.0_2020-10-01.7z.004**

__2. Open the folder where you downloaded the files and then right click the **PROJECT_VM_2.0_2020-10-01.7z.001** file and select **7-Zip → Extract here**.

__3. Open the **PROJECT_VM_2.0_2020-10-01** folder.

__4. Right click **PROJECT_VM_2.0_2020-10-01.vmx** and select **Send to → Desktop (create a shortcut)**

__5. On the desktop, rename the create shortcut to **PROJECT_VM_2.0_2020-10-01**

__6. Double click on the shortcut named as **PROJECT_VM_2.0_2020-10-01**. This will start the VM.

__7. Select **I Moved It** if prompt.

__8. Login using **osboxes.org** for user and password.

__9. Open the version file in the desktop and verify version is:

- **PROJECT_VM_2.0_2020-10-01**

__10. Close the VM.

<p>Note. This VM will be used for the students to work on a project after the Labs.</p>
--

Part 5 - Setup # 2: VM_WA2874_REL_1_0

__ 1. Download the following files:

- `VM_WA2874_REL_1_0.7z.001`
- `VM_WA2874_REL_1_0.7z.002`

__ 2. Open the folder where you downloaded the files and then right click the `VM_WA2874_REL_1_0.7z.001` file and select **7-Zip** → **Extract here**.

__ 3. This will extract all the compressed files from the 2 files (001 and 002) in the folder where you are located. After extraction you will see a new folder called `VM_WA2874_REL_1_0`

__ 4. Open the `VM_WA2874_REL_1_0` folder.

__ 5. Right click `VM_WA2874_REL_1_0.vmx` and select **Send to** → **Desktop (create a shortcut)**

__ 6. On the desktop, rename the create shortcut to `VM_WA2874_REL_1_0`

__ 7. Double click on the shortcut named as `VM_WA2874_REL_1_0`. This will start the VM.

__ 8. Select **I Moved It** if prompt.

__ 9. Login using **wasadmin** for user and password.

__ 10. Open a terminal and enter the following command:

```
sudo -i
```

[Enter **wasadmin** as password]

__ 11. Make sure you have internet by entering:

```
ping www.google.com
```

__ 12. Press **Ctrl + C** to stop the ping.

Important: If you cannot reach <code>www.google.com</code> means you don't have internet, check that the Network Adapter in the VM is configured as NAT.

__ 13. Enter the following command:

```
~/pull-images.sh
```

- __14. Wait for it to complete.
- __15. Type **exit** many times to close the Terminal.

Part 6 - Create a Docker Account and Obtain an Access Token [This part needs to be done by the Students or by the setup person]

In the next steps you will create a Docker account and obtain an access token. With the free Docker account, you can pull up to 200 Docker images every 6 hours. Without the account, you will most likely run into Docker pull rate-limiting issue where anonymous users get restricted to 100 Docker image pulls every 6 hours and that can cause issues especially when multiple anonymous users connect from the same network.

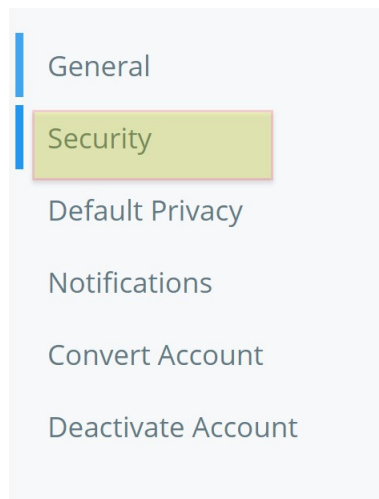
- __1. Open a browser and navigate to the following URL:

`https://hub.docker.com/`

- __2. Click **Sign Up** and create a new account.
- __3. Sign in with your newly created account.
- __4. Click on your username in the top right corner and select Account Settings.

Alternatively, click navigate to <https://hub.docker.com/settings/general> to access Account Settings.

- __5. Select **Security**.



- __ 6. Click **New Access Token**.
- __ 7. Add the token description, such as *docker & Kubernetes labs*.
- __ 8. Keep Access permissions as default (**Read, Write, Delete**).
- __ 9. Click **Generate**.

Notice it shows the following information on the page:

To use the access token from your Docker CLI client:

1. Run `docker login -u [redacted]`
2. At the password prompt, enter the personal access token.



- __ 10. Make a note of both the steps displayed on the page.
- __ 11. Start the virtual machine called **VM_WA2874_REL_1_0**.
- __ 12. Open a new terminal window.
- __ 13. Switch to the root user. [enter the sudo password]

```
sudo -i
```

- __ 14. Verify you are logged in as root.

```
whoami
```

It should show root.

- __ 15. Run the following command to log into Docker.

```
docker login -u {your-docker-id} -p {your-access-token}
```

- __ 16. Close the Terminal.
- __ 17. Close the VM.

Part 7 - Setup # 3. WA2874_REL_2_0

From here to the rest of the document you will find specific instructions to create a new Windows VM.

Call it as WA2874_REL_2_0

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 student.

Part 8 - Minimum Hardware Requirements

- 64 bits OS
- 2 cores
- 4 GB RAM minimum; 8 GB RAM recommended
- 20 GB in the hard disk
- Internet access.

Part 9 - Minimum Software Requirements

- Windows 8.1
- Firefox latest
- Chrome 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 *
- Postman 7.21.0 *
- Gradle 6.6.1 *
- * - indicates software provided as part of the courseware.

Part 10 - Software Provided

You will receive the following file:

- **WA2874_REL_2_0.zip**

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 11 - Instructions

__1. Extract the **.zip** file to **C:**

__2. Review that the following folders were created:

- **C:\LabFiles**
- **C:\Software\apache-maven-3.6.3**
- **C:\Software\gradle-6.6.1**

__3. Review that the following files were created:

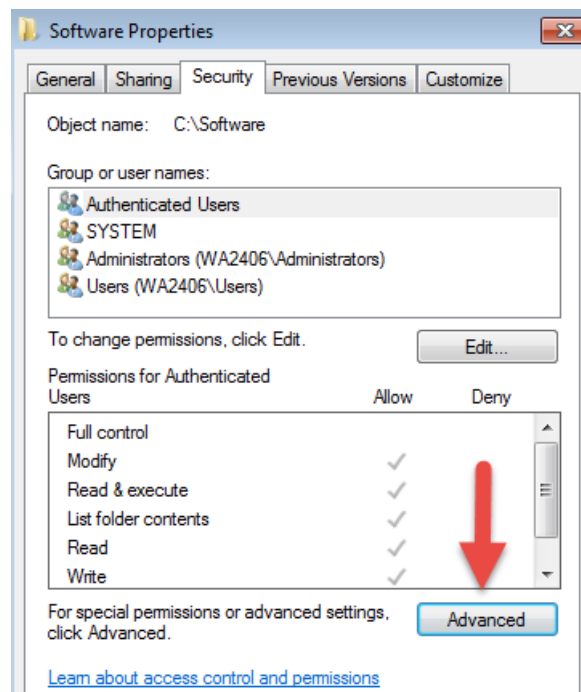
- **C:\Software\Eclipse-IDE-2020-03-eclipse-inst-win64.exe**
- **C:\Software\Git-2.26.0-64-bit.exe**
- **C:\Software\jdk-11.0.8_windows-x64_bin.exe**
- **C:\Software\Postman-win64-7.21.0-Setup.exe**

Part 12 - Privileges to Users

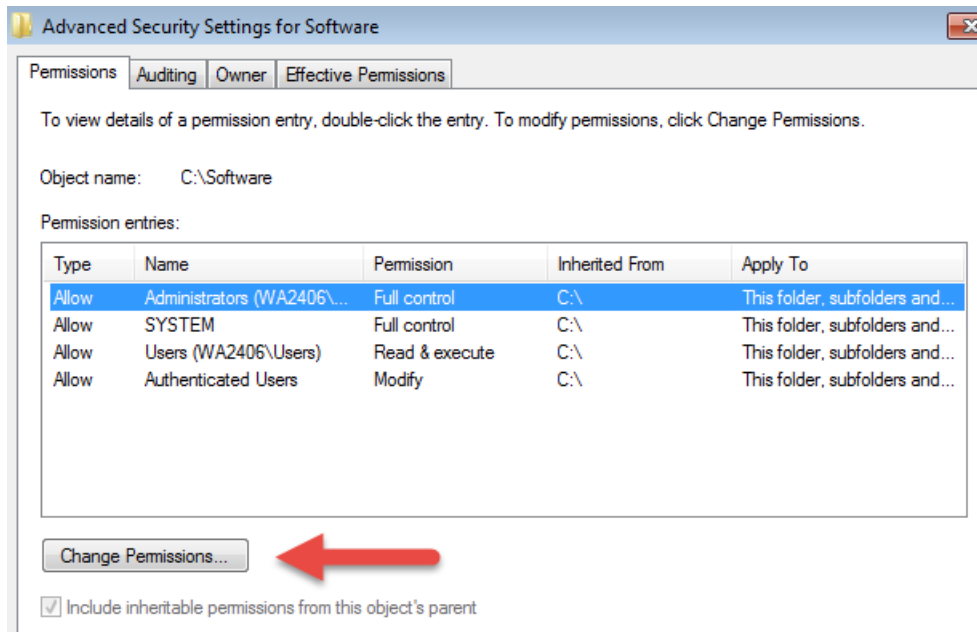
During the labs students will require privileges on several directories. They will need to be able to write, read, and modify files in these directories for the labs. Although the directions below are provided as an example, this may differ depending on Windows version. The important thing is to make sure students have full permissions on the folders below or they may not be able to accomplish some labs.

- **C:\Software**
- **C:\LabFiles**
- **C:\Workspace**

- __ 1. Create the user that the students will use during the class.
- __ 2. Give the student user account administrative rights. During the labs the student requires this to install software.
- __ 3. Open Windows Explorer.
- __ 4. Right click **C:\Software** and select **Properties**.
- __ 5. Click the **Security** tab.
- __ 6. Click **Advanced**.

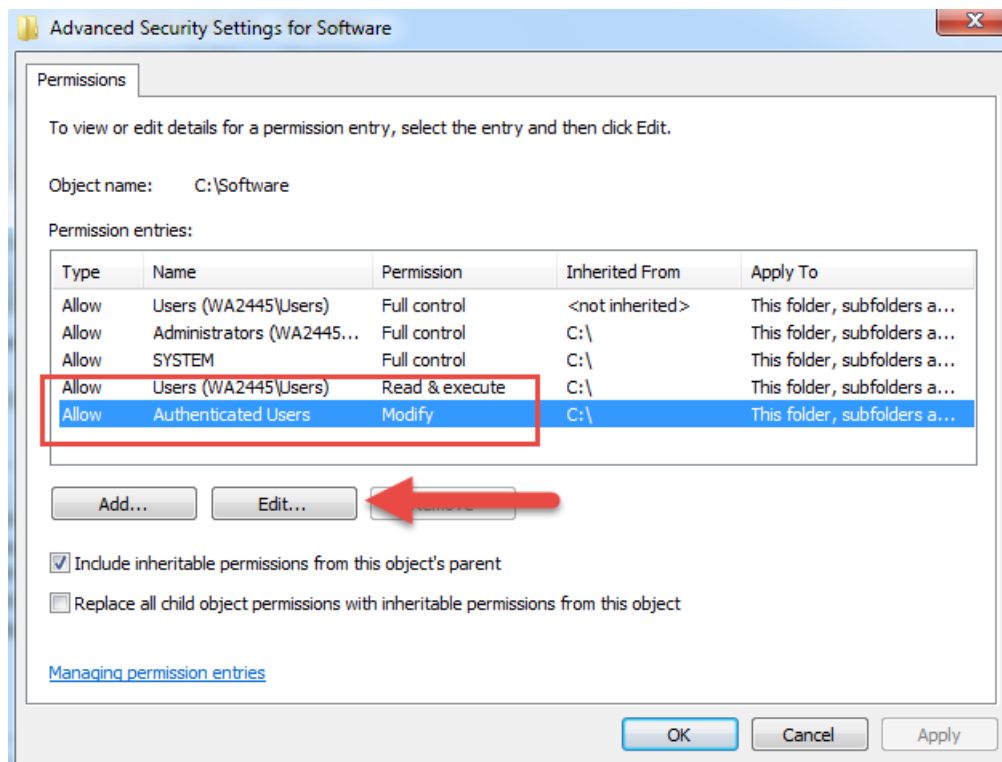


- __ 7. Click **Change Permissions**.

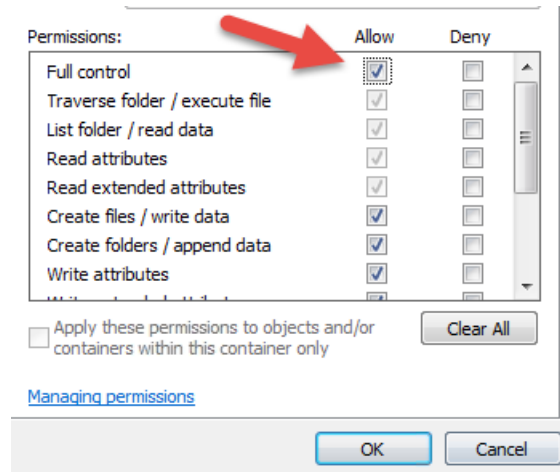


8. Select the **Authenticated Users** entry and click **Edit...**

Note. If you don't have a domain in your environment the select **Users** instead Authenticated Users.



9. Check the **Allow** checkbox to the right of **Full Control** and click **OK**.



__10. Select the checkbox for **Replace all child permissions with inheritable permissions from this objects** and then click **OK**.



- Include inheritable permissions from this object's parent
- Replace all child object permissions with inheritable permissions from this object

__11. A *Windows Security* dialog will open. Click **Yes**.

__12. Wait until Windows finishes updating security.

__13. Back in the *Advanced Security Settings* dialog, click **OK**.

__14. Back in the *Properties* dialog, click **OK**.

__15. Do the same steps to the **C:\LabFiles** folder.

__16. Create a folder called **C:\Workspace** if its not already created.

__17. Do the same steps to the **C:\Workspace** folder.

Part 13 - 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

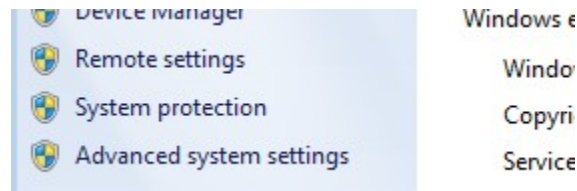
Some of the remaining steps will use the slightly different directory.
--

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

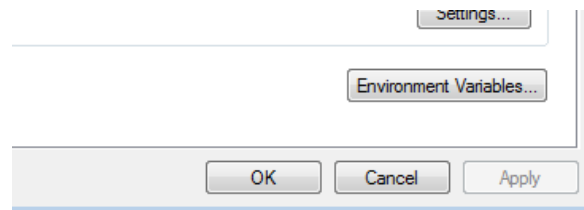
__4. Close the command prompt window.

__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.

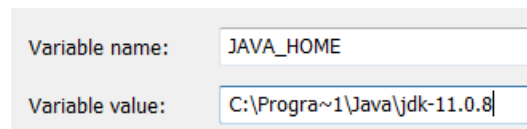
__9. As Variable name enter:

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.

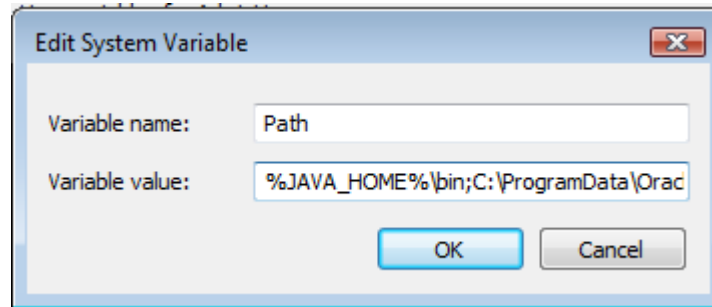


__12. From the *System Variables* list, select **Path** and click **Edit**.

__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 14 - Installing Eclipse-IDE-2020-03

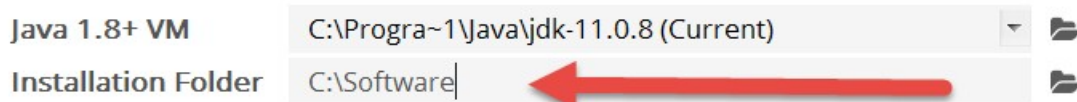
1. Open C:\Software\
2. Double click in this file to install eclipse:

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.

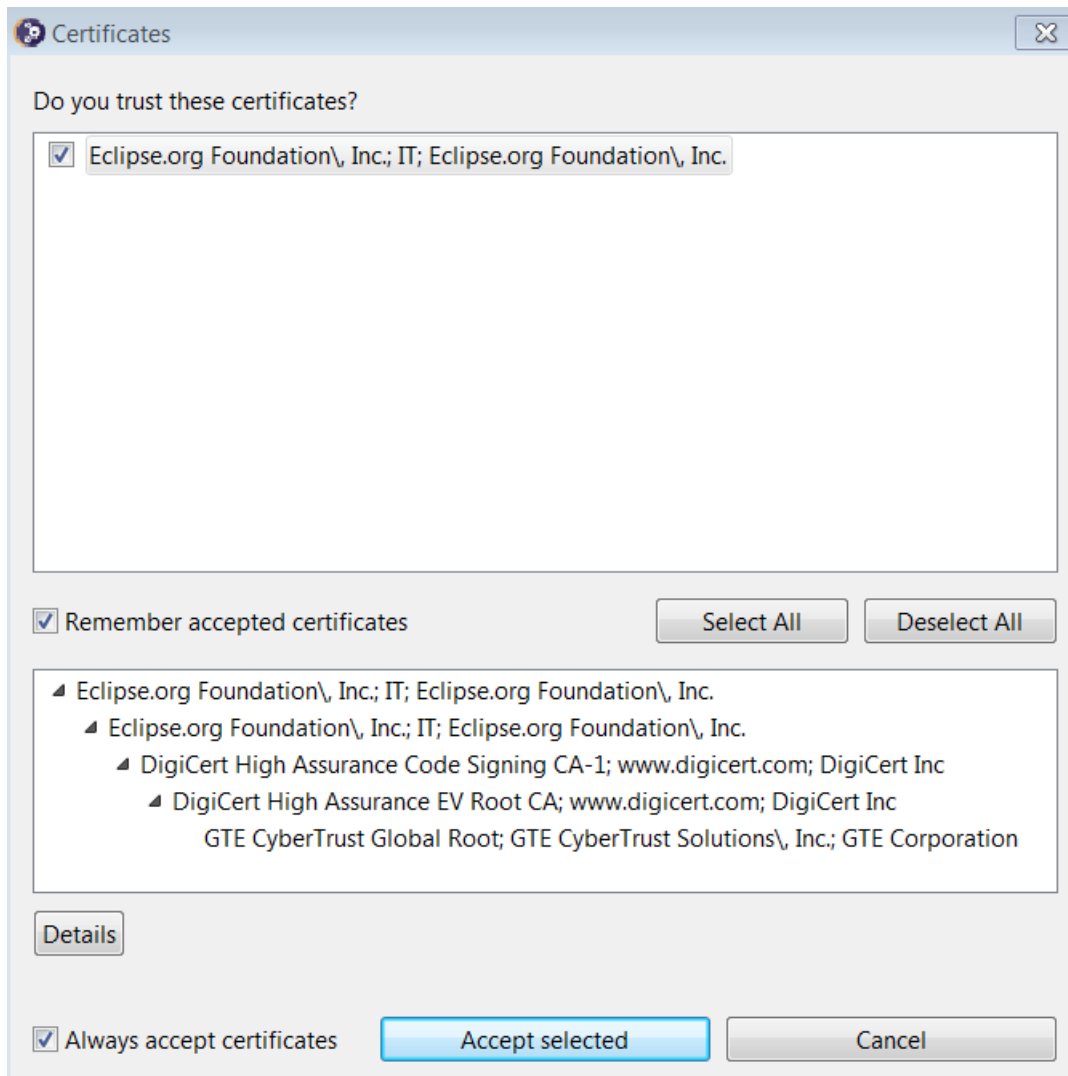
5. Click **Install**.
6. Click **Accept Now**.

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	
▶ LAUNCH		

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 15 - 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 16 - 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**.

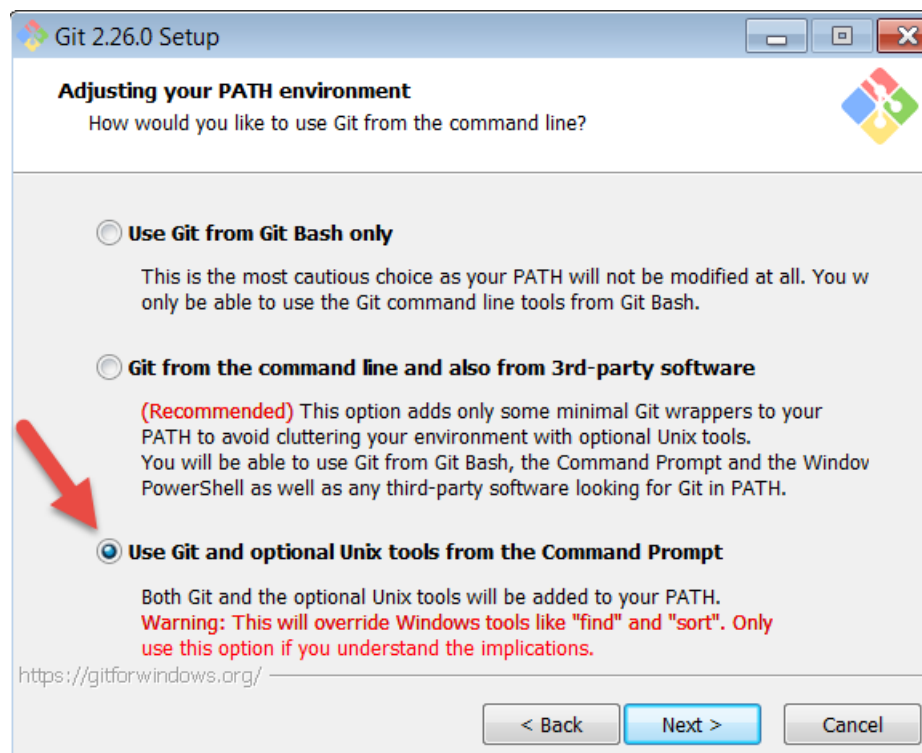
__4. In the Select Destination Location page, leave defaults and click **Next**.

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

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

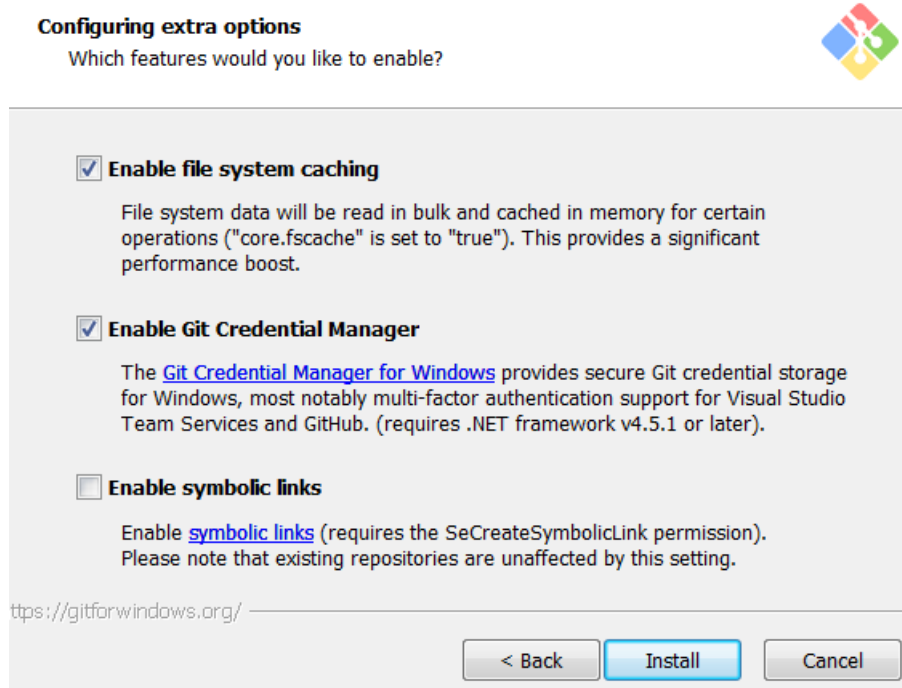
__7. In the Choosing the default editor used by Git page, leave defaults and click **Next**.

__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**.



- __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.

A screenshot of a terminal window. The title bar shows 'MINGW64:/c/Users/wasadmin'. The terminal content shows 'wasadmin@PC MINGW64 ~' followed by a '\$' prompt and a cursor.

__16. Close all.

You have completed GIT installation.

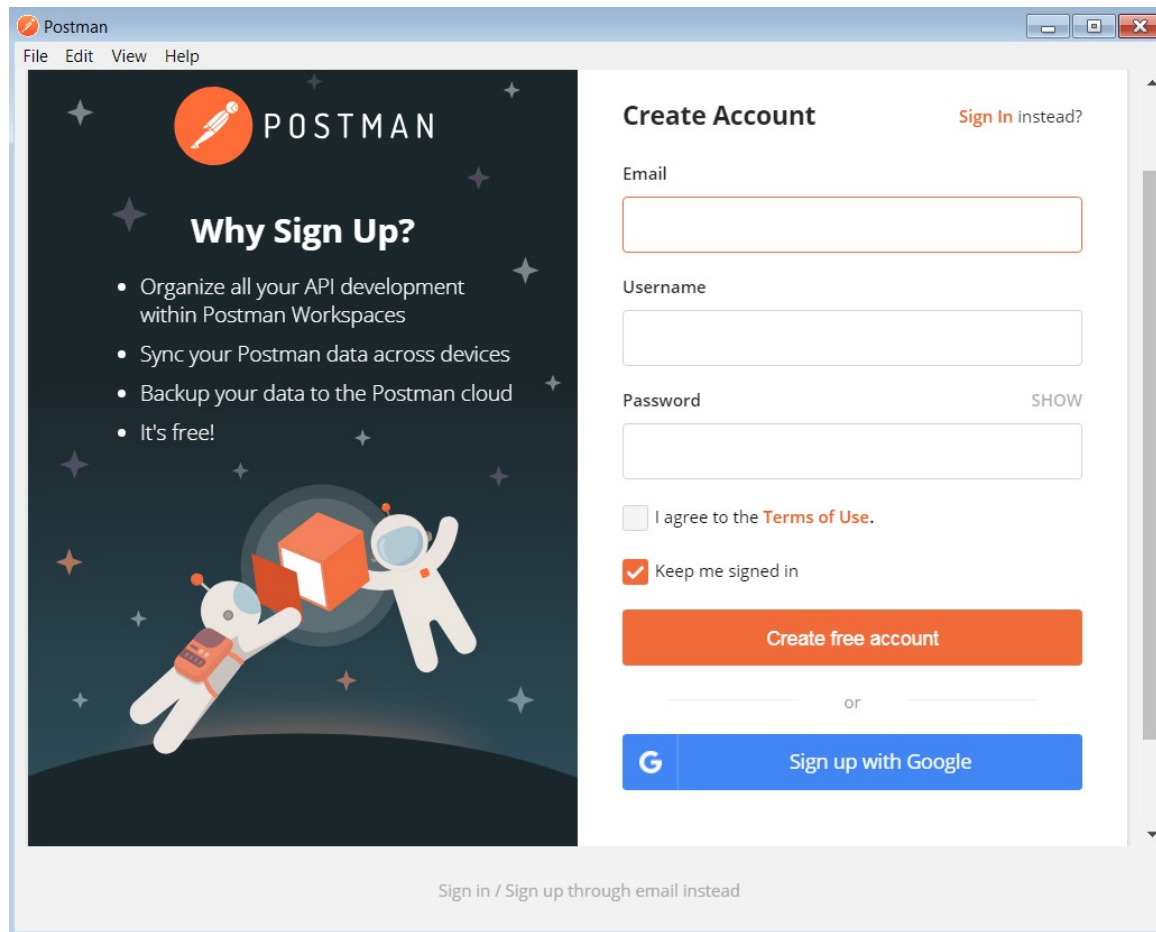
Part 17 - Installing Postman 7.21

__1. Open C:\Software\

__2. Double click this file begin installation:

Postman-win64-7.21.0-Setup.exe

Installation will start and after a while Postman will be launched.



__3. Close Postman.

Part 18 - Installing Gradle 6.6.1

__1. Open a Command Prompt.

__2. In the command prompt enter following command to verify Java is installed.

```
java -version
```

__3. Close the command prompt.

Next you will add Gradle to the environment variables.

__4. In the Windows Start Menu, right-click on the **Computer** link in the right-hand side of the Start panel, and then select **Properties**.

__ 5. Click on **Advanced system settings**.

__ 6. The system will display the **System Properties** dialog. Select the **Advanced** tab and click **Environment Variables**.

__ 7. Under the **System Variables** list, click the **New** button.

__ 8. As Variable name, enter:

```
GRADLE_HOME
```

__ 9. As Variable value, enter:

```
C:\Software\gradle-6.6.1
```

__ 10. Click **OK** to create the variable.

__ 11. From the **System Variables** list, select **Path** and click **Edit**.

__ 12. At the end of the line enter the following. Make sure to include the semi-colon at the beginning:

```
;%GRADLE_HOME%\bin
```

__ 13. Click **OK**.

__ 14. Click **OK** to close the **Environment Variables** window.

__ 15. Click **OK** to close the **System Properties** window.

Gradle verification

In this part you will verify Gradle is installed and added it to the path.

__ 16. Open a Command Prompt. You can do this with '**Start → Programs → Accessories → Command Prompt**'.

__ 17. Verify Gradle is installed.

```
gradle -version
```

__ 18. Verify **Gradle 6.6.1** is installed.

```
C:\Users\wasadmin>gradle -version

Welcome to Gradle 6.6.1!

Here are the highlights of this release:
- Experimental build configuration caching
- Built-in conventions for handling credentials
- Java compilation supports --release flag

For more details see https://docs.gradle.org/6.6.1/release-notes.html

-----
Gradle 6.6.1
-----
```

19. Close all.

Part 19 - Verification

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

- **WA2874_REL_2_0** (Windows VM)
- **VM_WA2874-REL_1_0** (Ubuntu VM)
- **PROJECT_VM_2.0_2020-10-01** (Ubuntu VM)

__2. Stop all VMs.

Part 20 - 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