# Install Development and Build Environments on Windows and Mac OS Machines

# Summary for Installing Development and Build Environments on Windows and Mac OS Machines

Manual Installation for Build and Deploy Machines versus pre-installed images is intentional; purpose explained during class.

1. Windows and MAC Machines are going to serve as:
   - **Development** Machine
   - **Build** Machine

2. **Development** Environment:
   - Java
   - Postman
   - Eclipse STS (IDE for Java/Spring Development)
   - Maven
   - Git Bash

3. **Build Machine** Environment:
   - (maven and java already installed via install for Development Environment)
   - (git bash client already installed via install for Development Environment)
   - docker
   - kubectl
   - (potentially add Jenkins)

4. **Deploy** Environment:
   - OpenShift
   - ibmcloud; Kubernetes in ibmcloud

There are other options for the **Build** Machine (especially when using CI/CD Pipelines that are run on Build Machines); a few of them are:

- Separate Machine, container, etc. running Jenkins and with all the needed build environment s/w
- AWS CodeBuild Service - we select an existing Image that has the build environment and we add our script that does the building
- OpenShift Build Images - uses Containers for the Build Machine w/ Images that have the build env s/w and the scripts for building

# Create Folders on Windows and Mac OS Machines

**Create following folders:**

Windows :

1. Open Command Windows or PowerShell
2. mkdir C:\projects\library
3. mkdir C:\logs
4. mkdir C:\dev

Mac OS:

1. Open Terminal
2. mkdir ~/projects
3. mkdir ~/projects/library
4. mkdir ~/logs
5. mkdir ~/dev

# Install 7zip on Windows Machine

**Install 7zip :**

Windows :

1. Google Chrome - https://www.7-zip.org/download.html
2. Click on Download link for your machine to download the 7zip install wizard (.exe)
3. Run the downloaded exe - e.g. **7z1900-x64.exe** - and accept all defaults

- IMPORTANT: Do NOT use the Windows built-in zip utility nor winzip to extract zip files

Mac OS:

- **Do not need for Mac OS; Mac OS's built-in tool works fine**

# Install Java SE Development Kit on Windows Machine

Notes:
We're installing Oracle's JDK and not OpenJDK - for some discussion on this, see https://www.baeldung.com/oracle-jdk-vs-openjdk

## Install Java SE Development Kit

### Windows

1. Open a browser and go to https://www.oracle.com/java/technologies/javase-jdk14-downloads.html
2. Click on the Windows x64 Installer (jdk-14_windows-x64_bin.exe)
3. Check the box to accept the license
4. Click on the green "**Download jdk-14_windows-x64_bin.exe**" button to download jdk-14_windows-x64_bin.exe
5. Double-click **jdk-14_windows-x64_bin.exe**
6. Windows x64 Installer starts
7. Click on **Next** on the first window of the installer
8. On the 2nd window of the installer, click on **'Change...**' to change the destination folder
9. Change the destination folder to **C:\dev\java\jdk-14**
10. Click OK
11. After changing the destination folder, click **Next** on the 2nd window if the installer
12. The next window shows the status of the installation
13. On the window that says **'Java(TM) SE Development Kit 14 (64-bit Successfully Installed**', click on **Close**
14. Java 14 is now installed
15. Add JAVA_HOME=C:\dev\java\jdk-14 as a System Variable
16. Add %JAVA_HOME%\bin to the Path System Variable
17. Make sure java installed and it's in the Path System Variable:
    a. Open a new Command Prompt window
    b. java -version

For screen shots, go to the Appendix - Screen Shots - Install Java SE Development Kit on Windows Machine

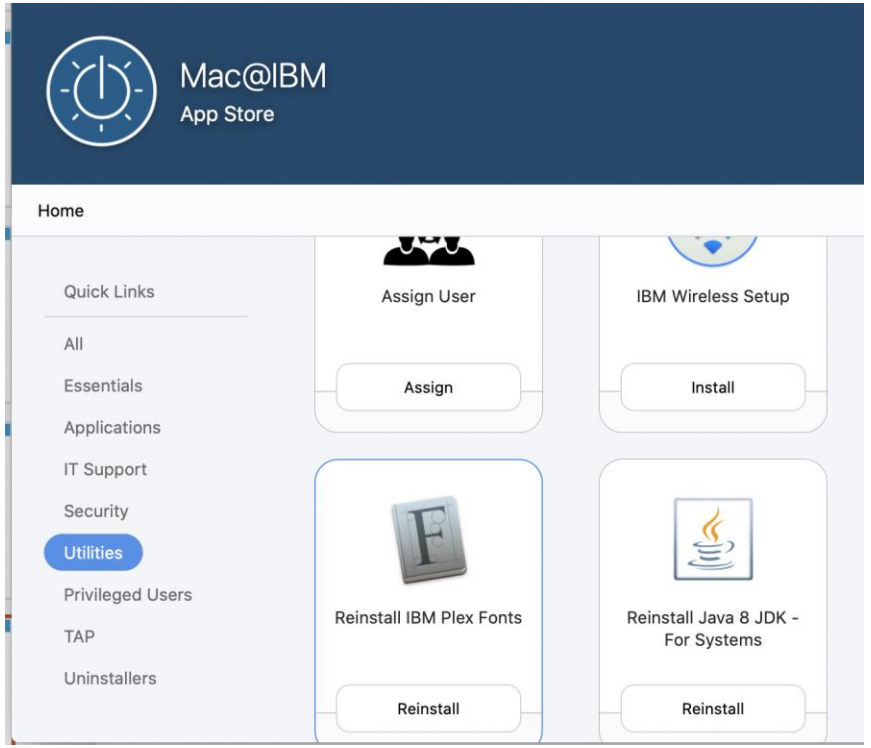
# Install Java SE Development Kit on Mac OS Machine

**Install Java SE Development Kit**

Mac OS :

1. Use the Mac@IBM store to install the JDK

Notes:
We're installing Oracle's JDK and not OpenJDK - for some discussion on this, see https://www.baeldung.com/oracle-jdk-vs-openjdk



# Install Maven on Windows and Mac OS Machines

**Install and Configure Maven**

Windows

1. Google Chrome - https://maven.apache.org/download.cgi
2. Download the binary zip (NOT the src!): click on apache-maven-3.6.3-bin.zip (NOTE: this was the latest version as of 01/15/2020)
3. Use 7-zip to extract the zip file (do NOT use windows built-in extract/unzip utility!) to **C:\dev** (change the default destination to c:\dev)
4. Add MAVEN_HOME System Environment variable set to: C:\dev\apache-maven-3.6.3 (NOTE: this was the latest version as of 01/15/2020)
5. Add %MAVEN_HOME%\bin to Path System Variable (and move it up in the Path)

Mac OS

1. Google Chrome - https://maven.apache.org/download.cgi & download the apache-maven…-bin.tar.gz (**e,g.** apache-maven-3.6.3-bin.tar.gz)
2. tar -xvf apache-maven-3.6.3-bin.tar.gz
3. In your shell's (bash, zsh) config file (bashrc, bash profile, etc) Set M2_HOME to where apache maven was installed and add M2
    - export M2_HOME=/users/tomsmith/apache-maven-3.6.3
    - export PATH=$PATH:$M2_HOME/bin
4. mvn -version

Both Mac OS and Windows :

1. Open a command window and create local maven repo (ignore the BUILD FAILURE): **mvn install -DperformRelease=true -DcreateChecksum=true**

# Install Git on Windows and Mac OS Machines

**Install Git Bash client**

Windows:

1. Google Chrome - https://git-scm.com/downloads and click on 'Windows' - it downloads **Git-2.25.0-64-bit.exe**
2. Run **Git-2.25.0-64-bit.exe**
3. Accept all the defaults  except ... change the Change editor to what you want - e.g. Notepad++ - on the 'Choosing the default editor using by Git' window and click Next

Mac OS :

1. No need to install git, it is pre-installed on the mac

Configure (both Windows and Mac OS):

1. See the slide deck, **git-clone-token-setup-v1.0.pptx**
2. In Git Bash, run git config with your name and email:
   - git config --global user.name "Your Name"
   - git config --global user.email your-ibm-email@us.ibm.com
   - git log --pretty=full

# Install Spring Tool Suite (STS) - Eclipse IDE on Windows and Mac OS Machines

**Install Spring Tool Suite** - it's the Eclipse IDE with features that support Spring development

For screen shots, go to the Appendix - Screen Shots - Install Spring Tool Suite (STS) - Eclipse IDE on Windows Machine

## Windows

- Using a Browser, go to https://spring.io/tools

1. Click on Windows 64-BIT
   - This will download **spring-tool-suite-4-4.6.0.RELEASE-e4.15.0-win32.win32.x86_64.self-extracting.jar**
2. Wait for spring-tool-suite-4-4.6.0.RELEASE-e4.15.0-win32.win32.x86_64.self-extracting.jar to finish downloading
   - You have to wait until C:\...\Downloads**Unconfirmed 357291.crdownload** goes away
3. Open a Windows Command Prompt window
4. cd to your where the jar file was downloaded - e.g. C:\Users\BILLLEONARD\Downloads
5. Run the following command: **java -jar spring-tool-suite-4-4.6.0.RELEASE-e4.15.0-win32.win32.x86_64.self-extracting.jar**
6. After "java -jar spring-tool-suite-4-4.6.0.RELEASE-e4.15.0-win32.win32.x86_64.self-extracting.jar" runs, the folder, sts-4.6.0.RELEASE, is created in the folder where you ran the command in - e.g. C:\Users\BILLLEONARD\Downloads\sts-4.6.0.RELEASE
7. Move the sts-4.6.0.RELEASE folder out of the Downloads folder and into C:\dev; result is: C:\dev\sts-4.6.0.RELEASE folder
8. Open STS by double-clicking on C:\dev\sts-4.6.0.RELEASE\SpringToolSuite4.exe
   - Note: it's easier if you create a desktop shortcut for this
9. STS starts; use the default directory for the workspace and click on '**Launch**'

## Mac OS

1. Go to https://spring.io/tools and download STS 4
2. It will download a .dmg file in the Downloads folder
3. Click on the file and Open it
4. Move the file to the Applications folder

# Create Swagger Account

https://app.swaggerhub.com/home

# Install Postman on Windows and Mac OS Machines

**Install Postman**

Windows :

1. Google Chrome - https://www.getpostman.com/downloads/
2. Hover over 'Download' and click on "Windows 64-bit" to download - e.g. Postman-win64-7.14.0-Setup.exe
3. Run it - e.g. Postman-win64-7.14.0-Setup.exe

Mac OS :

1. Google Chrome - https://www.getpostman.com/downloads/
2. Click on macOS to download Postman-osx-7.14.0.zip
3. The zip file is automatically unzipped into a Postman application in the Downloads folder
4. Move the Postman app to the applications folder

# Install MongoDB Server and MongoDB Compass on Windows and Mac OS Machines

**Install MongoDb Server and MongoDB Compass (single install) on Windows**

- This installs a local MongoDB Server which is helpful for local testing, but we'll be using a remote MongoDB Server
- However, it also installs MongoDB Compass (web-based UI for interacting with MongoDB)

Windows

1. Google Chrome - https://www.mongodb.com/download-center
2. Click on **Server**
3. Click on **Download** button
4. Double-click on **mongodb-win32-x86_64-2012plus-4.2.1-signed.msi** in the Downloads folder
   - Use the latest version; the above is just an example
5. The install wizard runs - accept all the defaults (don't change anything)
6. The installation of MongoDB ends with starting MongoDB Compass

Mac OS

1. Goto https://docs.mongodb.com/manual/tutorial/install-mongodb-on-os-x/ and follow the steps to install MongoDB Server
2. Goto https://www.mongodb.com/download-center/compass and click on Download
3. It will download a .dmg file on. The Downloads folder
4. Click on the file and Open it
5. Move the file to the Applications folder

# Install docker and docker CLI, kubectl CLI, and ibmcloud CLI on your build machine (Windows and Mac OS Machines)

Install docker, kubectl, ibmcloud CLIs on windows or mac (your build machine):

Windows:

1. Open Windows PowerShell as Administrator and run following:
   - [Net.ServicePointManager]::SecurityProtocol = "Tls12"; iex(New-Object Net.WebClient).DownloadString('https://ibm.biz/idt-win-installer')
   - This will take some time
2. Alternative (if above has any issues):
   - Set-ExecutionPolicy Unrestricted; iex(New-Object Net.WebClient).DownloadString('http://ibm.biz/idt-win-installer')
     - Answer "Y" to the 'Execution Policy Change" question

Mac OS:

1. Open a Terminal
2. curl -sL https://ibm.biz/idt-installer | bash

Installs the following:

1. The base IBM Cloud CLI (ibmcloud)
2. Homebrew (Mac only)
3. Git
4. Docker
5. Helm
6. kubectl
7. curl (Linux™ only)
8. IBM Cloud Developer Tools plug-in
9. IBM Cloud Functions plug-in
10. IBM Cloud Object Storage plug-in
11. IBM Cloud Container Registry plug-in (ibmcloud cr)
12. IBM Cloud Kubernetes Service plug-in (ibmcloud ks)

Maven and jdk are needed for a java build machine; these were installed on the previous slides for the install for the Development environment

# Set up Deployment Environments

# Create ibmcloud account and an apiKey for your account

→ To see ibmcloud's services, click on Catalog on the ibmcloud Dashboard

Login/create free ibmcloud account associated with your ibm email:

1. Connect to IBM Network (e.g. IBM VPN)
2. Go to https://cloud.ibm.com/login
3. Create a free ibm cloud account if you don't have one (it creates an ibm cloud **lite** account)
4. Potential issues (work with ibm help and ibmcloud help if issues):
    - Get rid of trial accounts
    - Some had an issue where they had to contact the IBM help desk to connect their IBM email id to ibmcloud account

Create longterm apiKey credential for your ibmcloud user account:

1. Create a long-term credential - an apiKey - to use for authentication when using ibmcloud CLI - https://cloud.ibm.com/iam/apikeys
2. Click on "Create an IBM Cloud API key" - store key somewhere on your windows or mac
    - My api key as of 01/28/2020 6:04am CST is: HusilqYAaPcdnyBwrGQRTWUPUzJZfltAOPpj71Hvl-M
    - We use this in ibmcloud login: **ibmcloud login** --apikey HusilqYAaPcdnyBwrGQRTWUPUzJZfltAOPpj71Hvl-M -r us-south

We'll wait until class to do the following:

1. Create a Kubernetes Cluster in ibmcloud using ibmcloud's IKS service
    - Note: we're not going to create an OpenShift cluster on ibmcloud

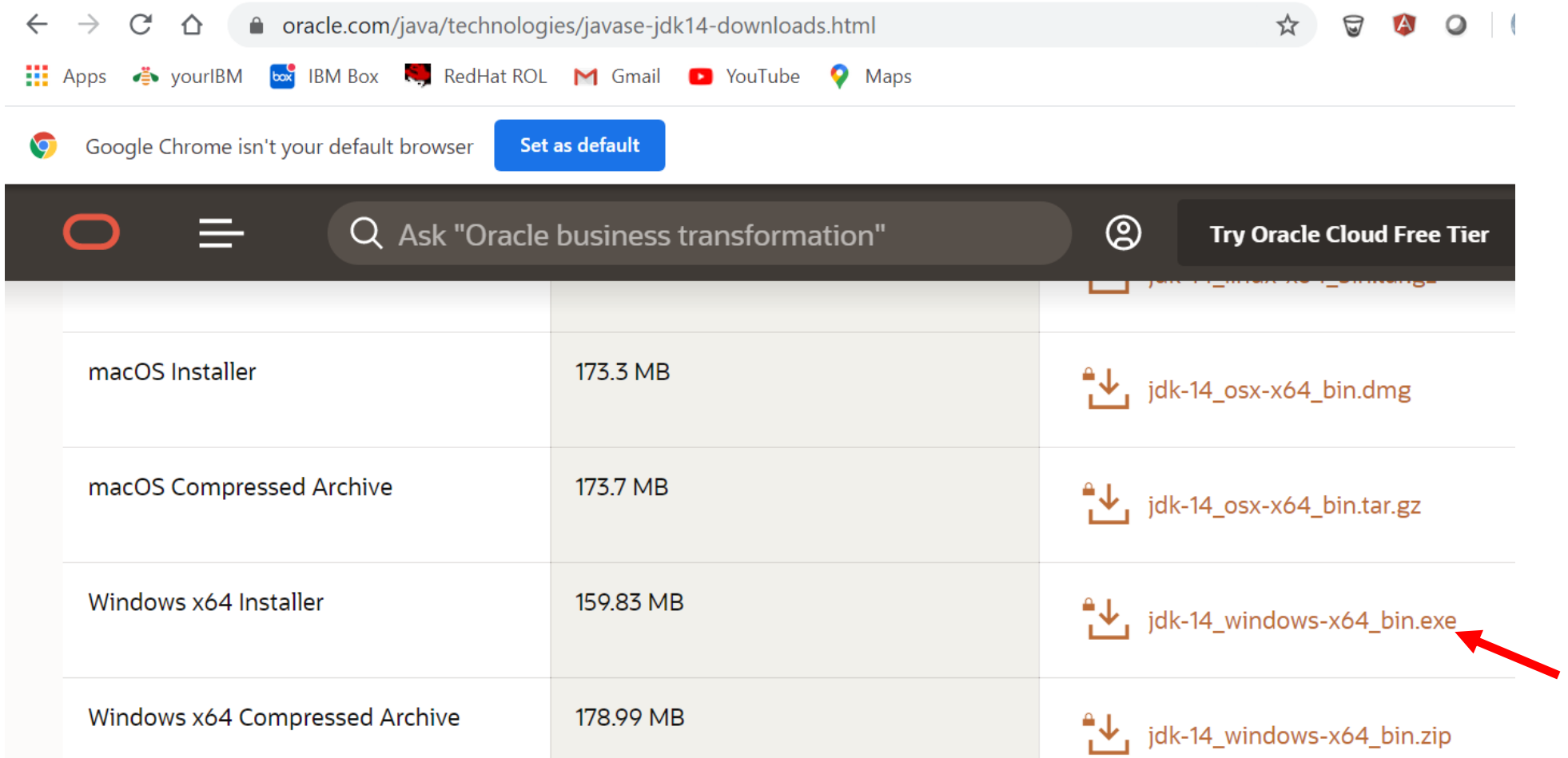
2. Create an OpenShift cluster in OpenShift's environment
    - https://www.openshift.com/products/online/

3. Add OpenShift CLI (oc) to Windows and Mac machines

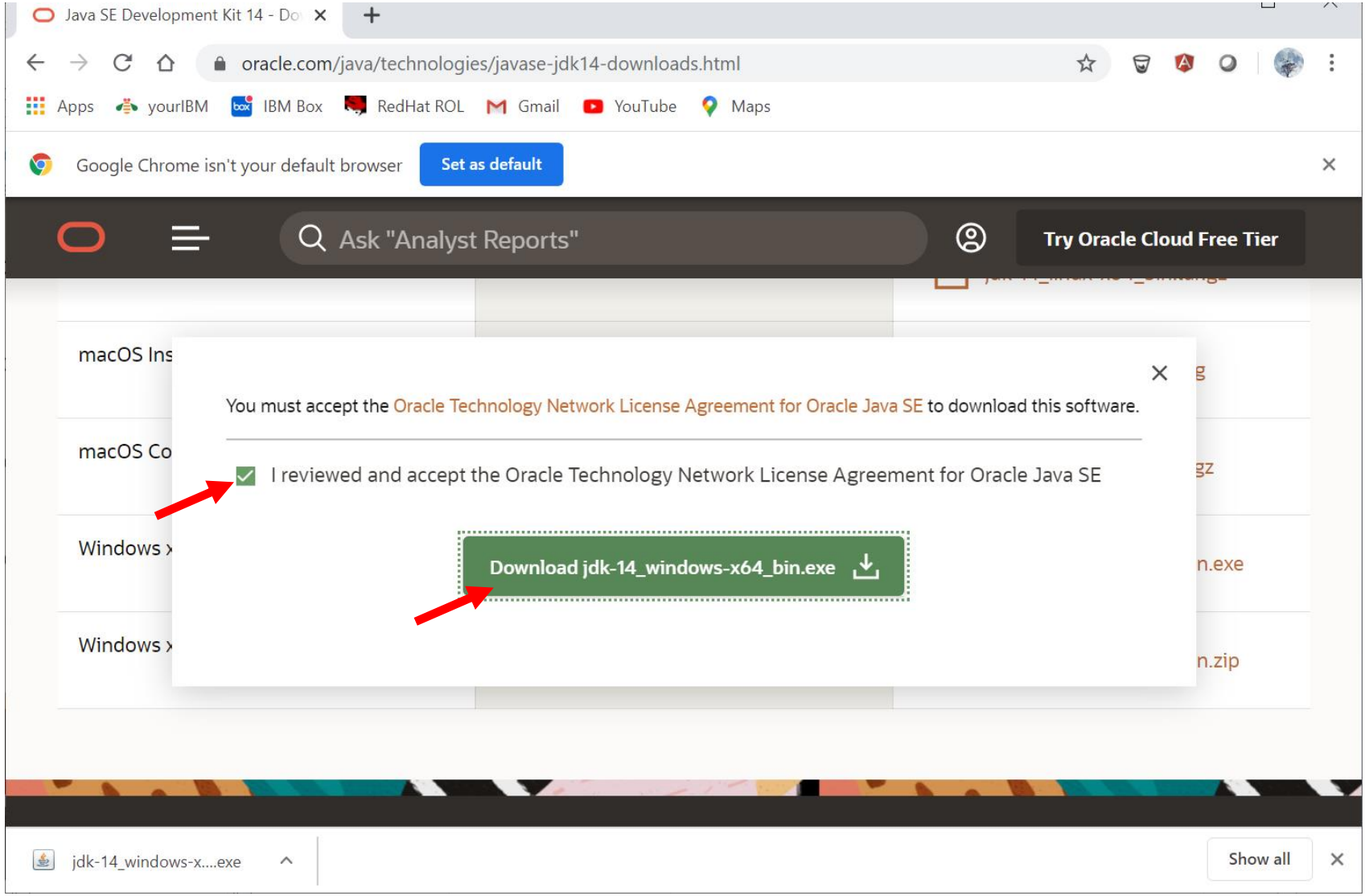
# Appendix

# Screen Shots - Install Java SE Development Kit on Windows Machine

1. Open a browser and go to https://www.oracle.com/java/technologies/javase-jdk14-downloads.html
2. Click on the Windows x64 Installer (jdk-14_windows-x64_bin.exe)

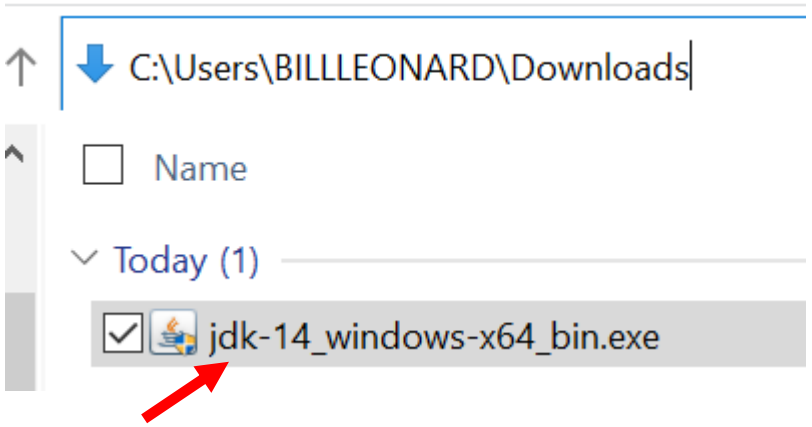


1. Check the box to accept the license
2. Click on the green "Download jdk-14_windows-x64_bin.exe" button to download jdk-14_windows-x64_bin.exe

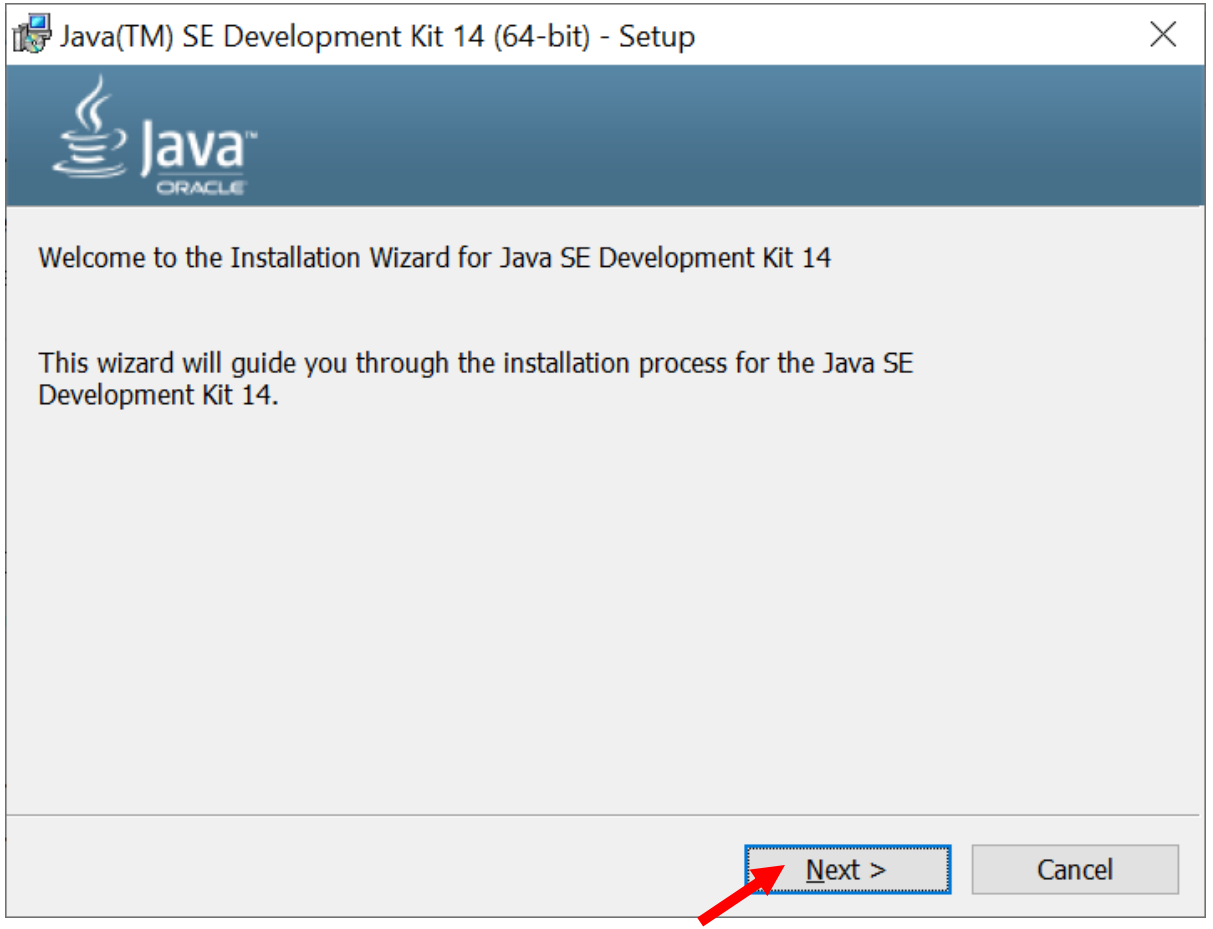


# Screen Shots - Install Java SE Development Kit on Windows Machine - Slide 3 of 8

**Double-click** jdk-14_windows-x64_bin.exe

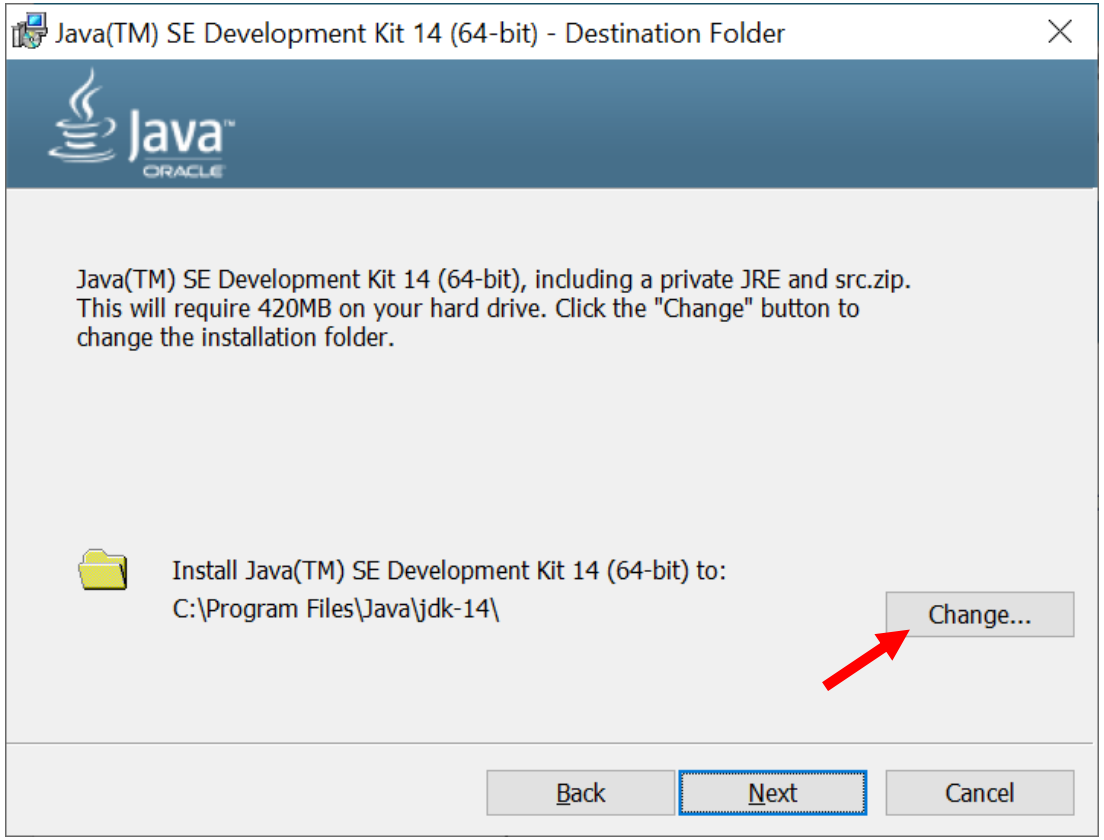



- Windows x64 Installer starts
- Click on Next on the first window

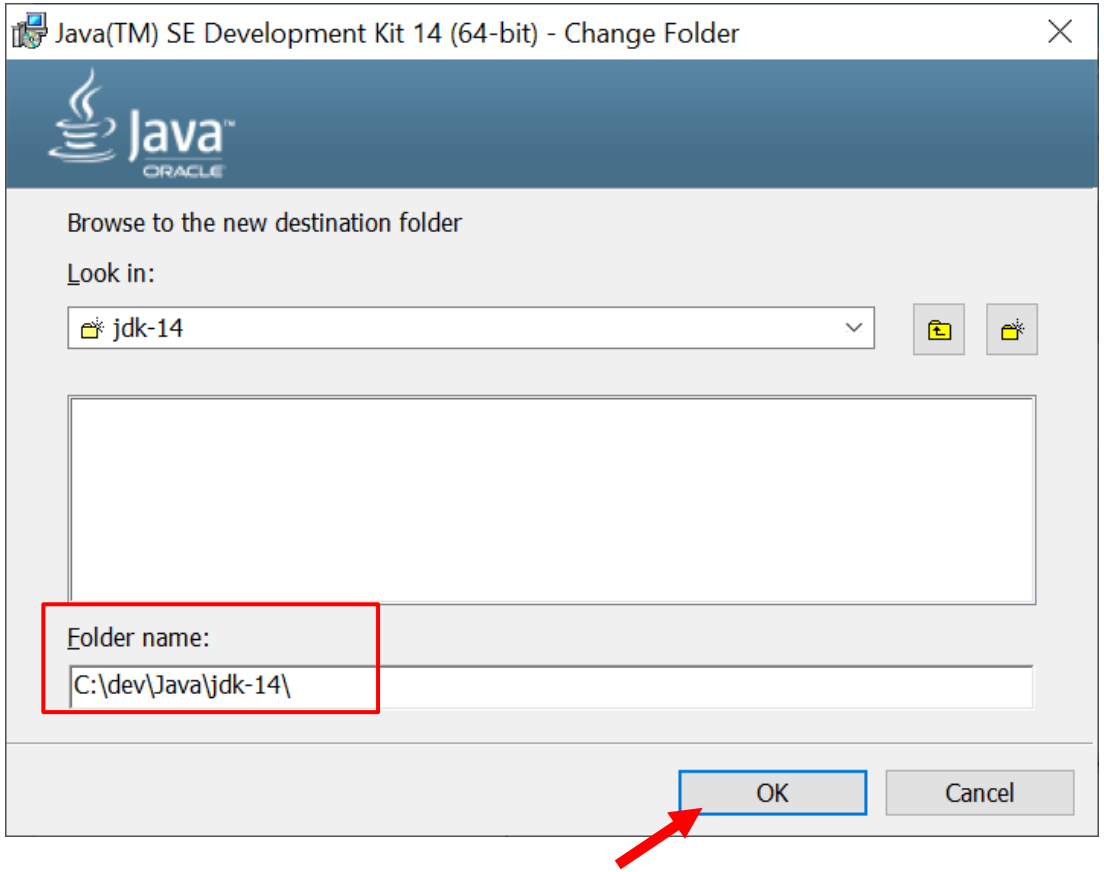


# Screen Shots - Install Java SE Development Kit on Windows Machine - Slide 4 of 8

On the 2nd window of the installer, click on 'Change'...'
to change the destination folder



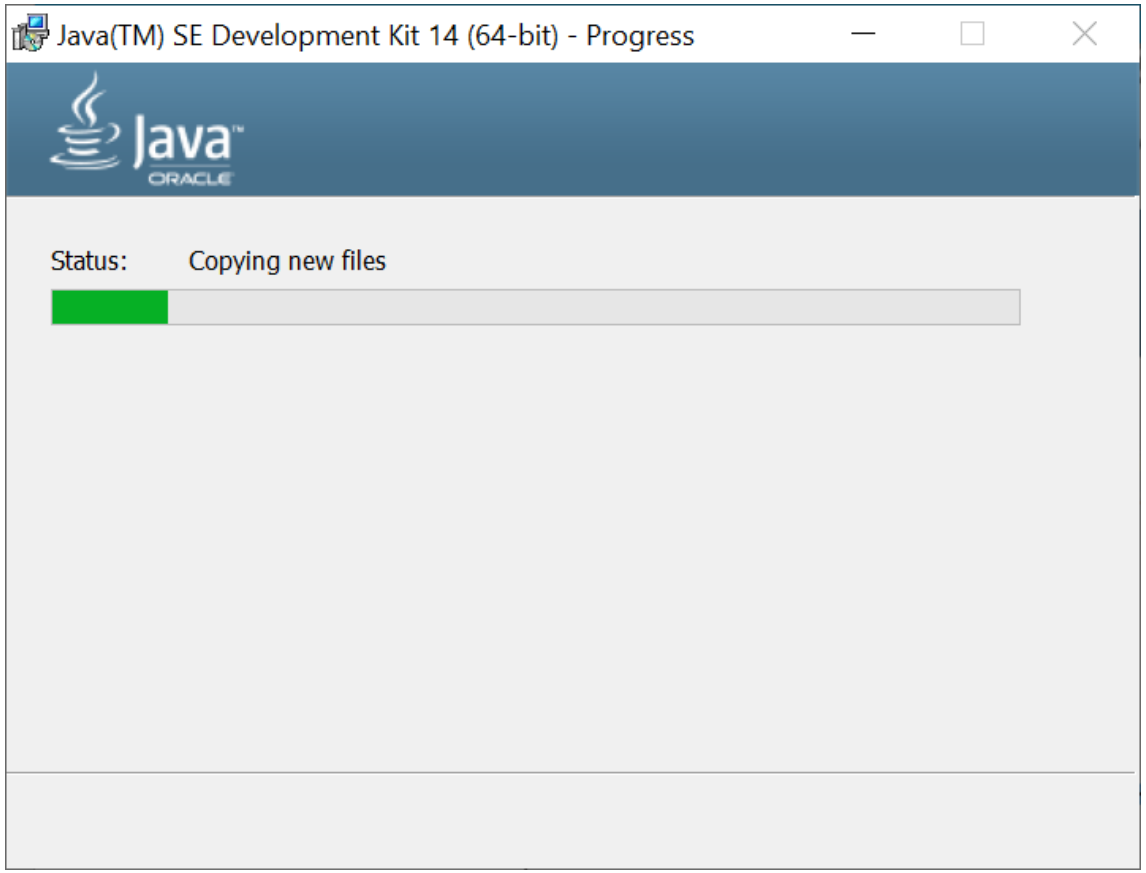

- Change the destination folder to C:\dev\java\jdk-14
- Click OK



# Screen Shots - Install Java SE Development Kit on Windows Machine - Slide 5 of 8

After changing the destination folder, click Next on the 2nd window if the installer


The next window shows the status of the installation

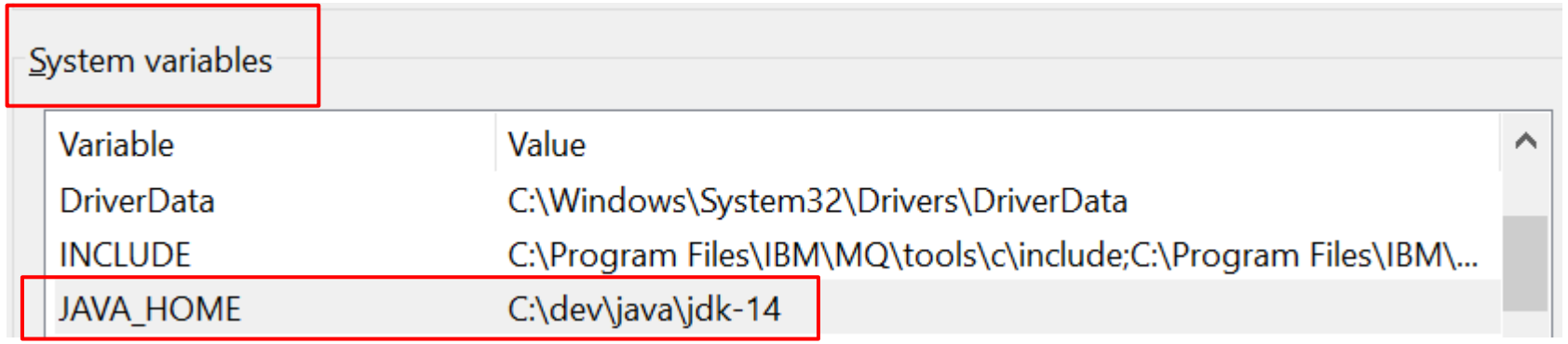


# Screen Shots - Install Java SE Development Kit on Windows Machine - Slide 6 of 8

- On the window that says 'Java(TM) SE Development Kit 14 (64-bit Successfully Installed', click on Close
- Java 14 is now installed

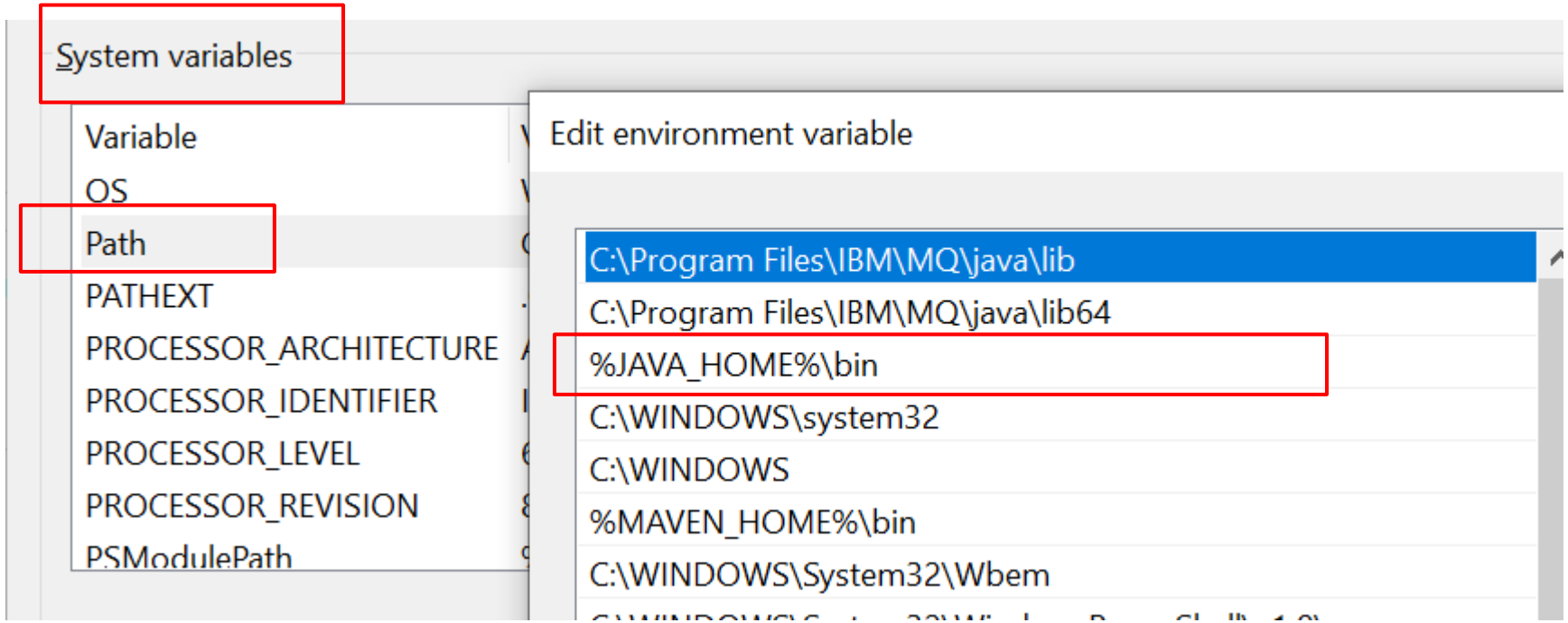
# Screen Shots - Install Java SE Development Kit on Windows Machine - Slide 7 of 8

Add JAVA_HOME=C:\dev\java\jdk-14 as a System Variable




---

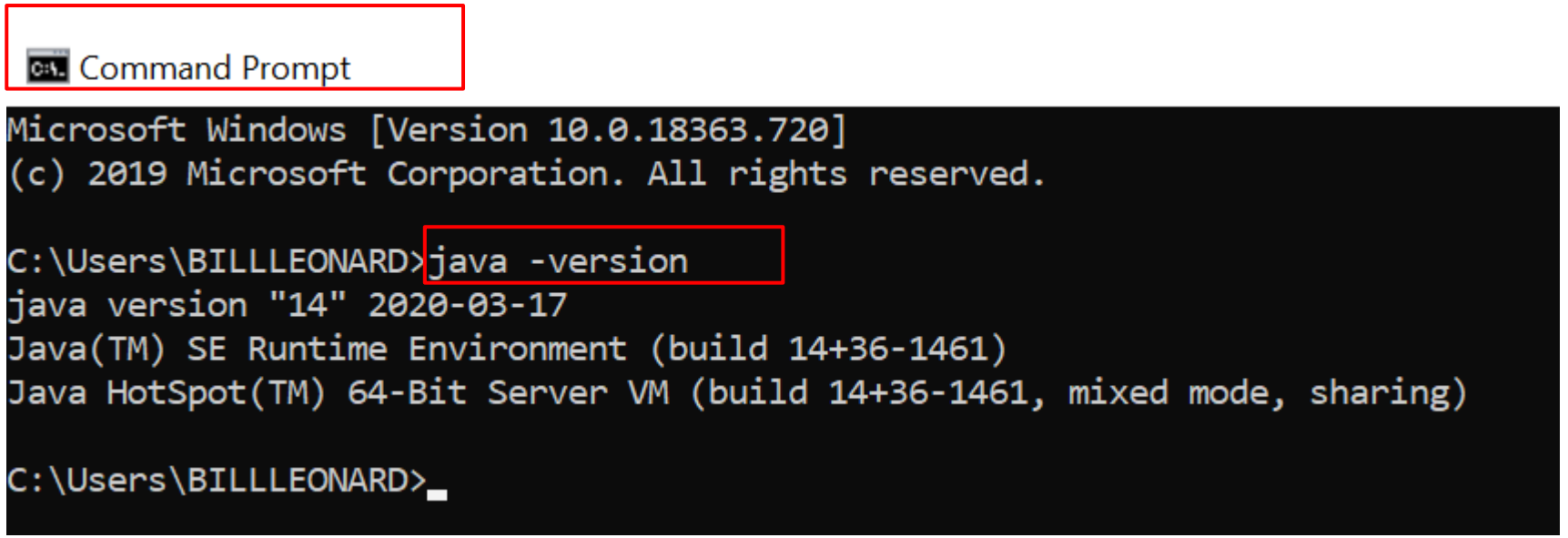
Add %JAVA_HOME%\bin to the **Path** System Variable



# Screen Shots - Install Java SE Development Kit on Windows Machine - Slide 8 of 8

Make sure java installed and it's in the Path System Variable:

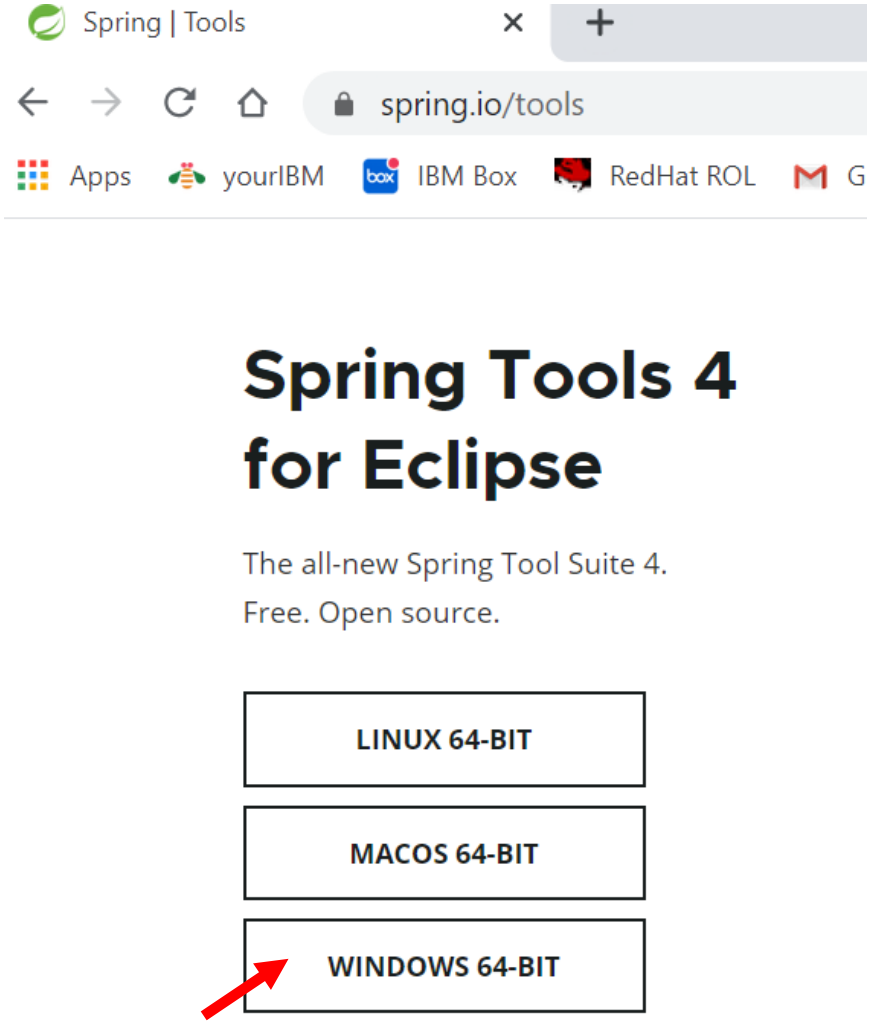
- Open a new Command Prompt window
- java -version



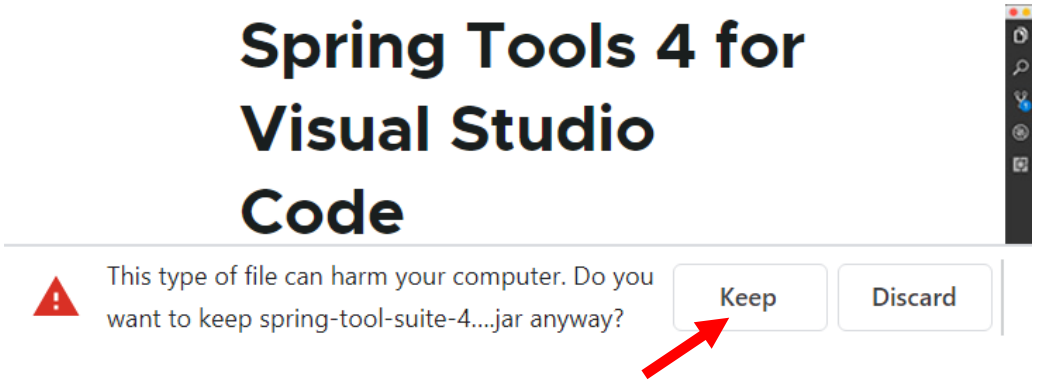
# Screen Shots - Install Spring Tool Suite (STS) - Eclipse IDE on Windows Machine

# Screen Shots - Install Spring Tool Suite (STS) - Eclipse IDE on Windows Machine - Slide 1 of 6

- Using a Browser, go to https://spring.io/tools
- Click on Windows 64-BIT

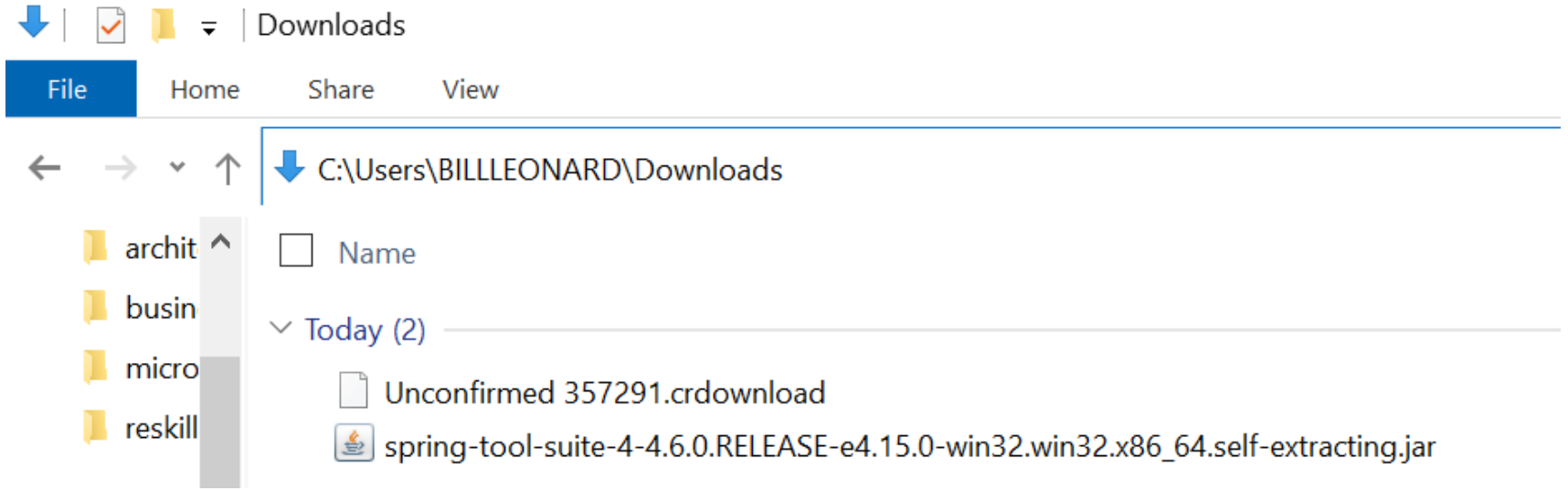



- Google Chrome Browser shows the downloaded file to the bottom left of the browser
- Google Chrome gave me a warning because the file is a jar file
- Click on Keep if you get this warning



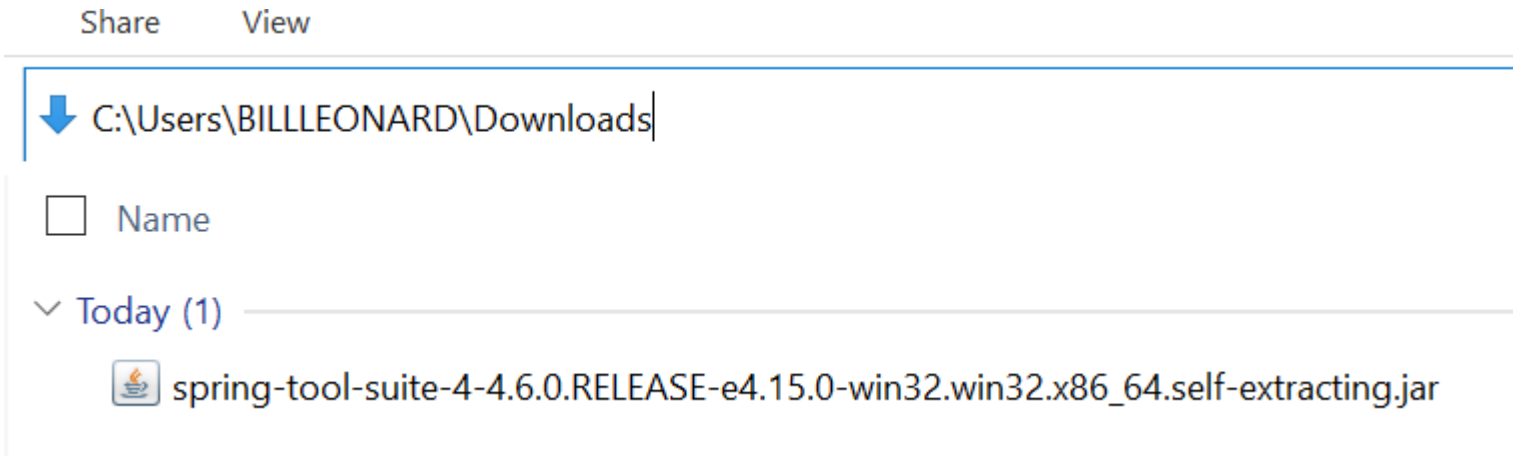
# Screen Shots - Install Spring Tool Suite (STS) - Eclipse IDE on Windows Machine - Slide 2 of 6

- Wait for **spring-tool-suite-4-4.6.0.RELEASE-e4.15.0-win32.win32.x86_64.self-extracting.jar** to finish downloading
  - You have to wait until C:\Users\BILLLEONARD\Downloads**Unconfirmed 357291.crdownload** goes away
    - Don't be in the C:\Users\BILLLEONARD\Downloads folder



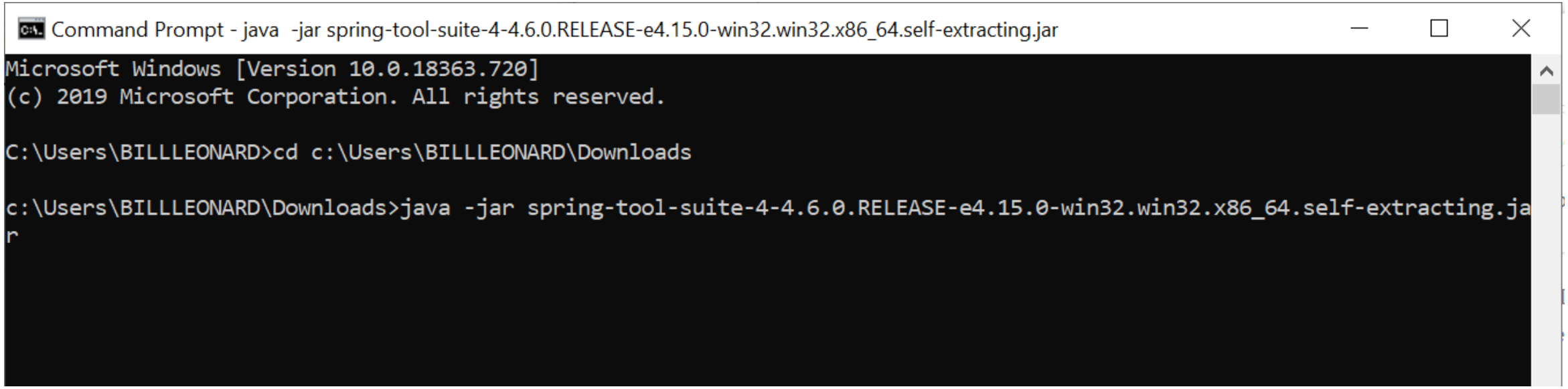

---

After **Unconfirmed 357291.crdownload** goes away, **spring-tool-suite-4-4.6.0.RELEASE-e4.15.0-win32.win32.x86_64.self-extracting.jar** has finished downloading



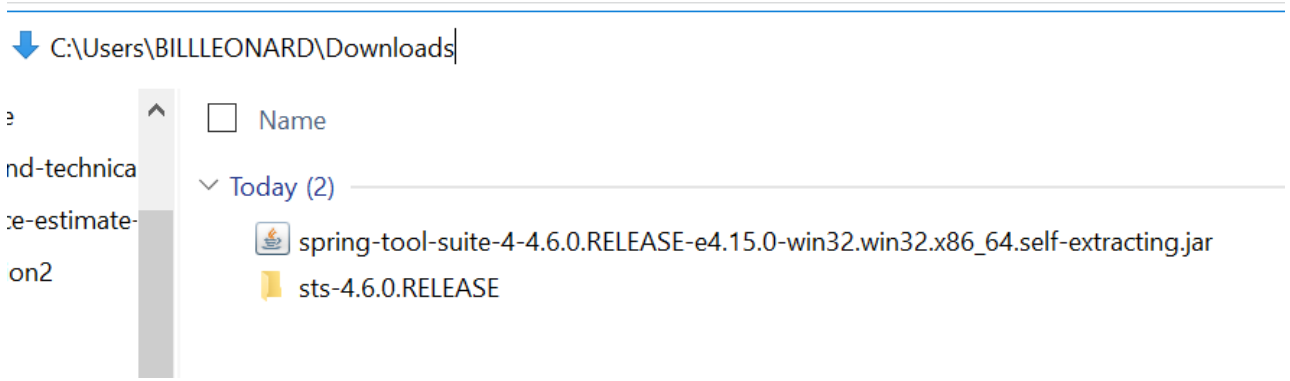
# Screen Shots - Install Spring Tool Suite (STS) - Eclipse IDE on Windows Machine - Slide 3 of 6

- Open a Windows Command Prompt window
- cd C:\Users\BILLLEONARD\Downloads
- **java -jar spring-tool-suite-4-4.6.0.RELEASE-e4.15.0-win32.win32.x86_64.self-extracting.jar**

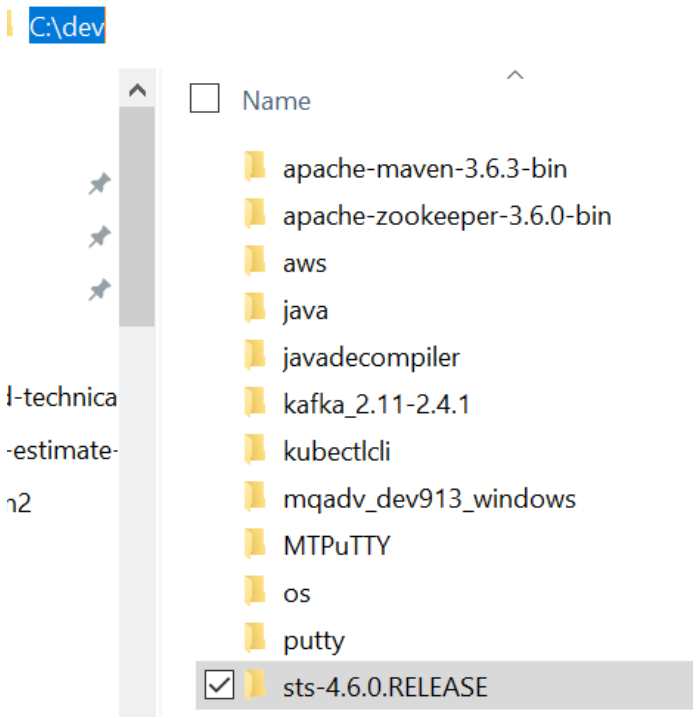


# Screen Shots - Install Spring Tool Suite (STS) - Eclipse IDE on Windows Machine - Slide 4 of 6

After "java -jar spring-tool-suite-4-4.6.0.RELEASE-e4.15.0-win32.win32.x86_64.self-extracting.jar" runs, the folder, sts-4.6.0.RELEASE, is created (C:\Users\BILLLEONARD\Downloads\sts-4.6.0.RELEASE)

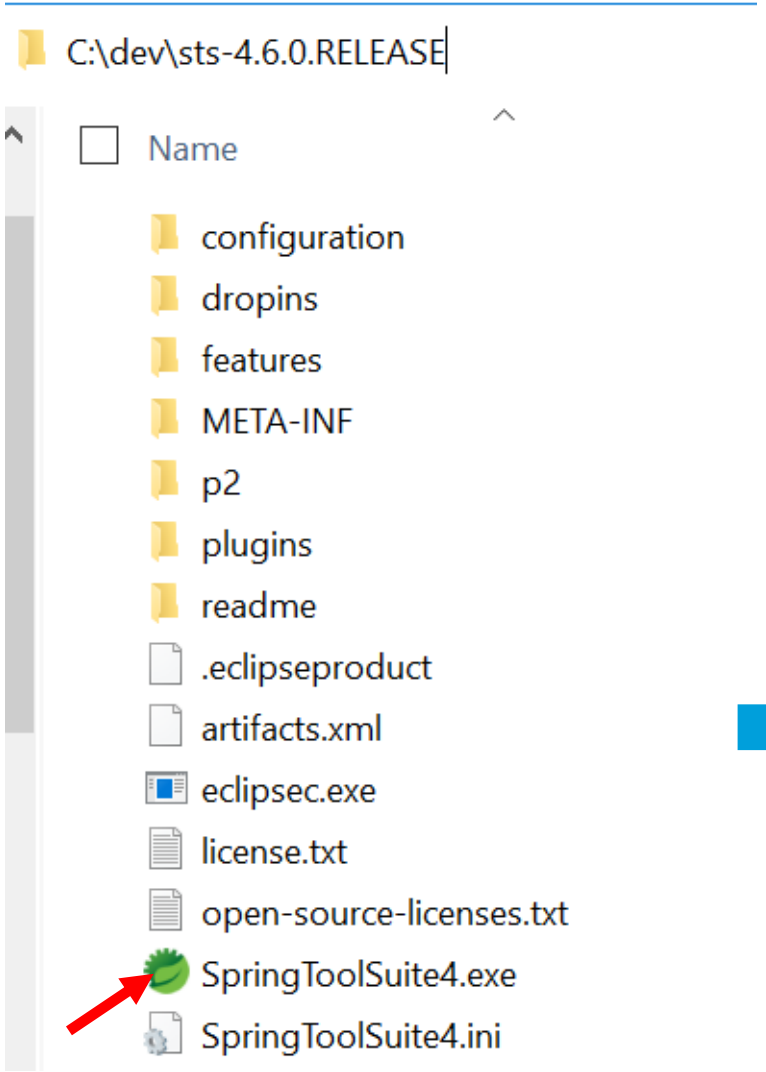



Move sts-4.6.0.RELEASE folder to C:\dev (C:\dev\sts-4.6.0.RELEASE)

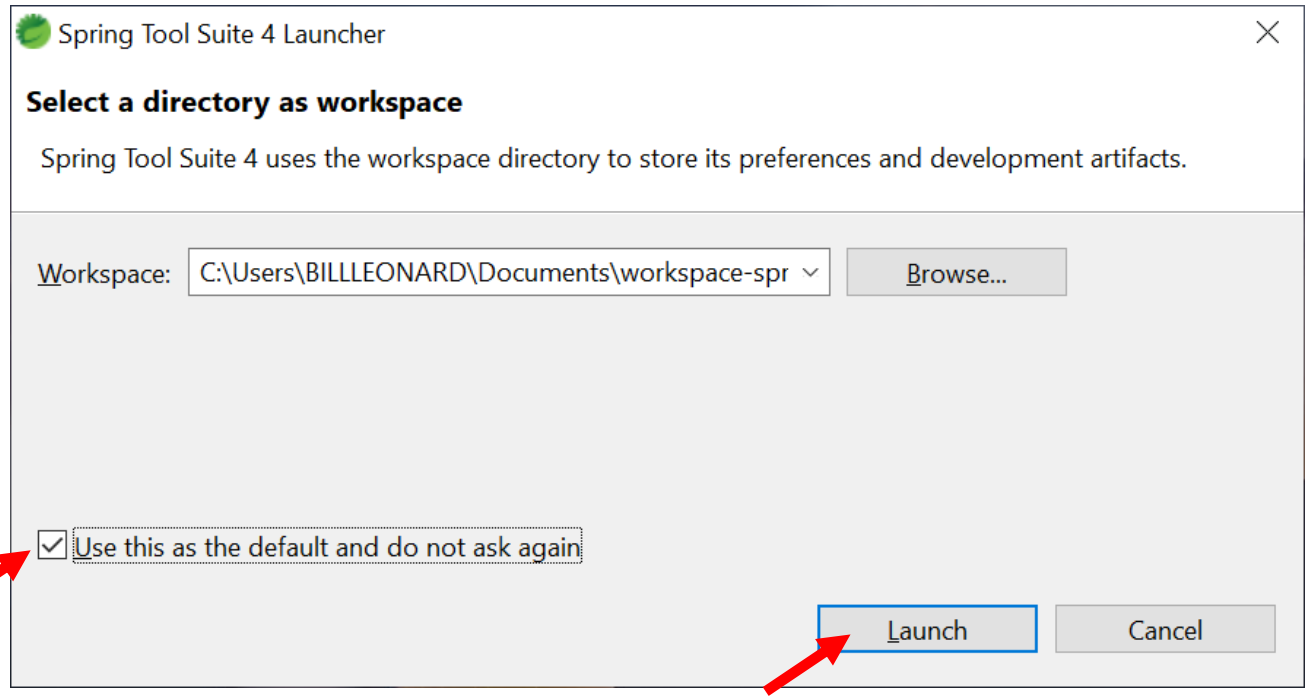


# Screen Shots - Install Spring Tool Suite (STS) - Eclipse IDE on Windows Machine - Slide 5 of 6

- Open STS by double-clicking on C:\dev\sts-4.6.0.RELEASE\SpringToolSuite4.exe
- Note: it's easier if you  create a desktop shortcut for this

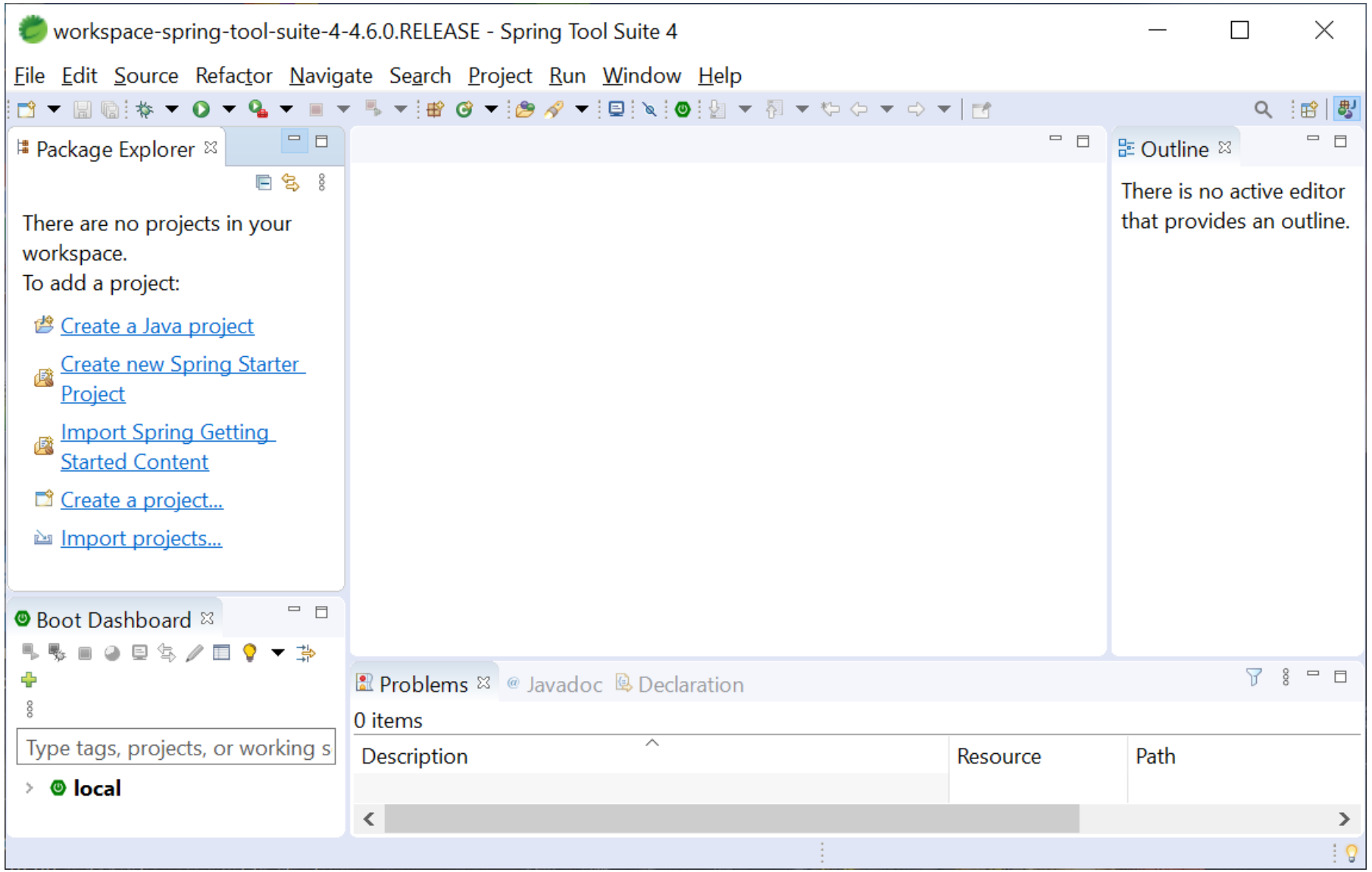



- Use the default directory for the workspace and click on 'Launch'



# Screen Shots - Install Spring Tool Suite (STS) - Eclipse IDE on Windows Machine - Slide 6 of 6

STS after it's Opened:



# Wait until the class to do the following

# Create a Kubernetes Cluster in ibmcloud using IKS Service ; create a docker registry namespace using ibmcloud's CR service

Please do not do this prior to the class.
We'll work through this **in the class.**

- We could create an OpenShift cluster on ibmcloud, but we're not. We'll use the OpenShift cloud-based environment for that (see "Set up Deployment Environments - OpenShift" slides).
- OpenShift is a Kubernetes cluster; therefore, anything that you can do in Kubernetes, you can do the same in OpenShift.

Create K8S cluster using ibmcloud IKS via ibmcloud dashboard:

- You can use the ibmcloud CLI to send IKS service requests but this shows using the dashboard

1. Log into IBM Cloud w/ your IBM intranet ID (make sure you don't switch to the 1999532 account)
2. Go to the Catalog Service: https://cloud.ibm.com/catalog
3. Click on "Kubernetes Service" to get to the Kubernetes Service
4. On the "Kubernetes Service" page, click on "Create" button on the top right
5. On the "Create a new cluster" page, Click on "Free"
6. On the same 'Create a new cluster" page, change the cluster name to something more meaningful if you want
7. Click on "Create Cluster"
8. It shows you your specific Kubernetes Cluster - in my case, I named it "**billscluster**"
9. The instructions for the rest of the tasks are on this cluster's page under the 'Access' tab (they're not exact - follow the instructions in here instead)

Check access to k8s dashboard:

1. Click on "IBM Cloud" - then select "Kubernetes" to get to the "Clusters" page
2. Click on your Kubernetes cluster on the "Clusters" page
3. In your Kubernetes cluster page, click on "Kubernetes dashboard"

# Create a docker registry namespace using ibmcloud's CR service

Please do not do this prior to the class.
We'll work through this **in the class.**

Create a namespace in IBM Cloud Container Registry (it's a Docker Registry)

- The docker image needs to be in a Docker Registry in order for Kubernetes Deployment Controller to pull it and use docker to create a docker container from it

1. On your windows or Mac machine, open a PowerShell (Windows) or Terminal (Mac) and:
    a. ibmcloud login …
    b. ibmcloud cr namespace-add **billscrns ← change 'billscrns' to anything you want**
    c. ibmcloud cr region-set us-south

2. Go to the Container Registry service's web console and make sure that you see the namespace created:
   https://cloud.ibm.com/kubernetes/registry/main/namespaces
    - Can do this step via the ibmcloud CLI, too - just want to show that we can use the CLI or the web UI for ibmcloud services

https://www.openshift.com/products/online/

Please do not do this prior to the class.
We'll work through this **in the class.**