Setting Up Eclipse Development Environment

| [Setting up Eclipse](#_heading=h.2et92p0)  [Windows Setup](#_heading=h.tyjcwt)  [Step 1: Download SE Development Kit and Install](#_heading=h.3dy6vkm)  [Step 2: Download Eclipse (Windows)](#_heading=h.1t3h5sf)  [Step 3: Setup Environment Variables](#_heading=h.4d34og8)  [Step 4: Set Eclipse Windows](#_heading=h.2s8eyo1)  [Mac Set Up](#_heading=h.17dp8vu)  [Step 1: Download Java SE Development Kit](#_heading=h.3rdcrjn)  [Step 2: Set Up Eclipse](#_heading=h.26in1rg)  [Creating a Java Project and Program in Eclipse (Windows or Mac](#_heading=h.lnxbz9)  [Step 1: In Eclipse Create a Java Project](#_heading=h.44sinio)  [Step 2: Create, Compile and Run a Java Class File](#_heading=h.z337ya)  [Step 3: Github Desktop: Version and Backup](#_heading=h.3wsts4cm6oj)  [Troubleshooting Eclipse](#_heading=h.2xcytpi)  [Issue #1: Exit code 13 error](#_heading=h.1ci93xb)  [Issue #2: Project has a red !](#_heading=h.2bn6wsx) |
| --- |

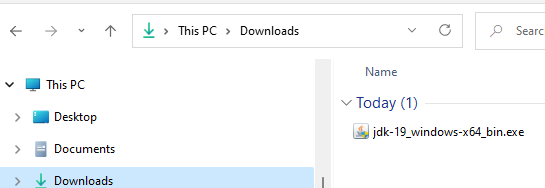
# Setting up Eclipse

* You should first have your [git repository set up](https://docs.google.com/document/d/1mqMr_xE4cPg7IMeNPt0dBCIb7HCUp41SqKb_H-I45Sw/edit?tab=t.0).
* I am providing the JDK and Eclipse versions from Fall 2024 since changes won’t affect what we are doing in this class.
* Take your time reading instructions so you do not need to do it over.
* Find a partner to set up together if you can.
* See windows set up below or go to [Mac Set Up](#_heading=h.17dp8vu)

## **Windows Setup**

### Step 1: Download SE Development Kit and Install

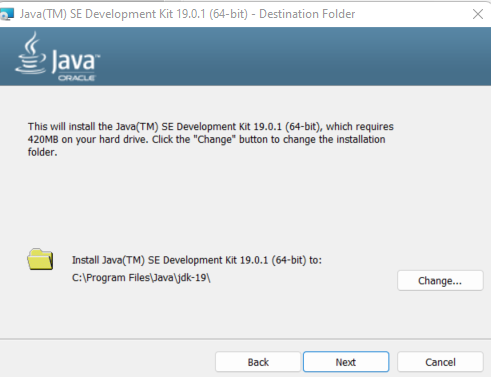
1. [Download this JDK-22 file](https://drive.google.com/file/d/1vP1itH-tBwCPGa4INFuVmEHXpFdSYnt-/view?usp=drive_link)
2. Once the download completes, start the installation by double clicking on the downloaded .exe in the browser or in the directory it is saved in.

****

1. You will be asked if you want to allow this app to make changes to your device - click **Yes**. Note: The installation can take a bit to get started – it can be slow - so be patient.
2. You will see the **Welcome** dialog – click **Next** to start download



1. The next dialog offers some options – click **Next**.
   * Notice that the JDK will be installed in the **Program Files/ Java** directory
   * This is where it should be installed so don’t change the location



1. When installing the status window may or may not display. – be patient, it takes time to get started. When the window shows successfully installed click close.



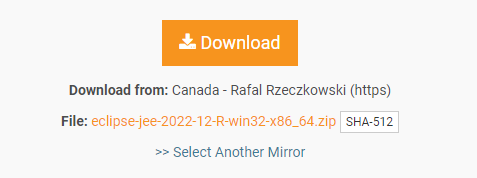
### Step 2: Download Eclipse (Windows)

Your Workspace files will be stored in git and not in the Eclipse Folder

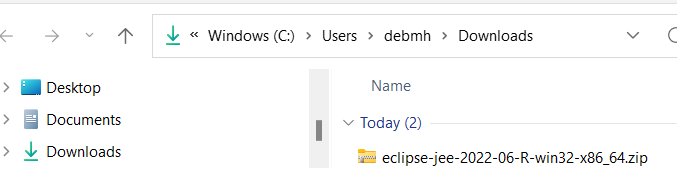
1. Install [Eclipse IDE 2024-06 R Packages](https://www.eclipse.org/downloads/packages/release/2024-06/r)
2. Scroll down to **Eclipse IDE for Enterprise Java and Web Developers** (DO NOT use the installer)



1. You should end up on a page with a big download button - click that button!



1. Windows The **Eclipse zip** file will download into the **default download location** (the “download” folder in your “user” account) unless you have set up your browser to save to a different location.

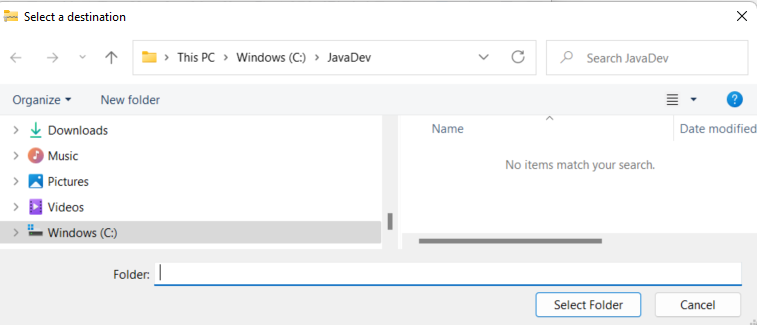


1. Eclipse is a bit different since you don’t actually “install” it. Instead, because it is a zip file you will **extract** it to a directory of your choice.

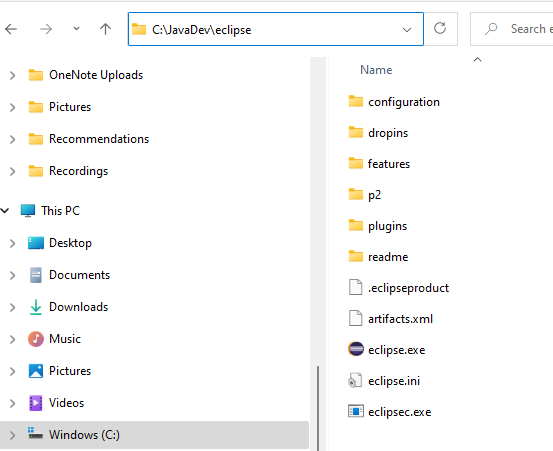


1. The “**Select a Destination and Extract Files**” dialog will be displayed. In the dialog, select where to place Eclipse. Click Browse and read below. Select show extracted files when complete. The unzip will start after you click extract– this could take a while depending on your system.
   * On my personal systems, my preference is to create a "**JavaDev**" directory on my C drive and unzip Eclipse there - **C:\JavaDev. DO NOT PUT IT IN YOU GIT REPOSITORY>**
   * When you start to do more development, having a development directory on your machine helps keep all development tools in one place.
   * **NOTE: Placing the eclipse file too deep into the file structure causes an error.**

You can click new folder to create a JavaDev folder



1. When the unzip completes, go to the location you selected and view all the files
   * On my system that location is **C:\JavaDev\eclipse**
   * Note: if you ever need to uninstall Eclipse, just delete the eclipse directory.
   * Note, when reinstalling or updating eclipse your workspace will be backed up in your git folder!

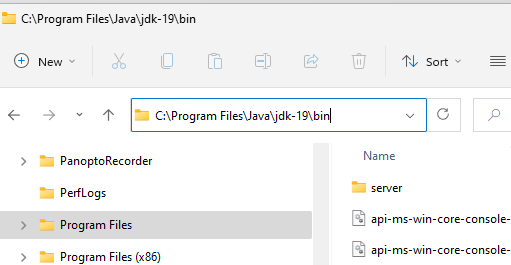


### 

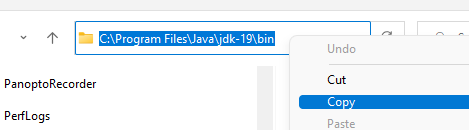
### Step 3: Setup Environment Variables

Remember to go slow and pay attention to details.

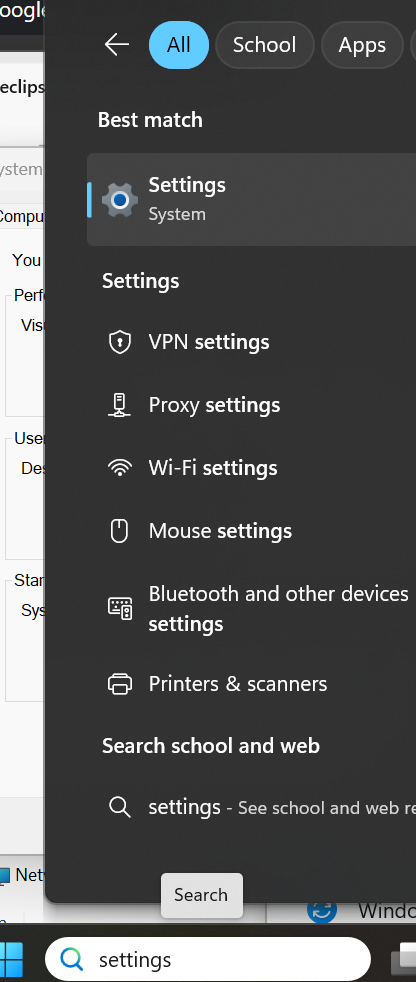
1. Find where the JDK is on your hard drive. It should be in the **Program Files** directory in the **Java** directory: **C:\Program Files\Java\jdk-22**
2. Go one more level down to the **bin** directory



1. In the File Explorer window, click in the little window where the path is shown which will highlight the path.
   * Make sure the path includes the **bin** directory: **C:\Program Files\Java\jdk-22\bin**
   * Copy the path to the **bin** directory (since it's highlighted use CTRL-C to save it in the buffeR or right click to copy.)



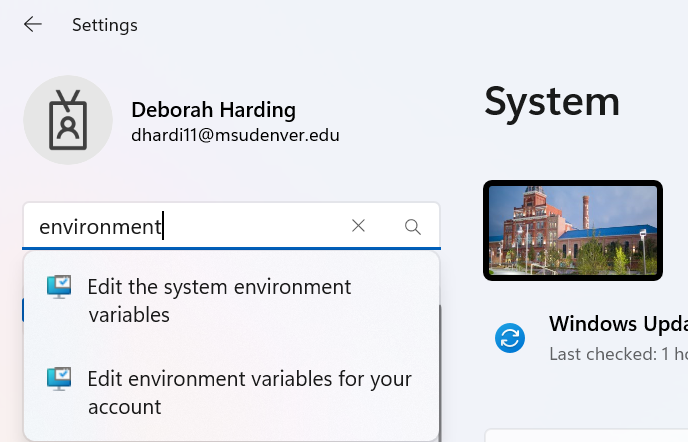
1. Go to settings. You can Search for settings



1. A new window will appear showing your system information. On the right, scroll down to **Related Settings** and click “**Advanced system settings**”. Might be different based on the OS version so see below for windows 10 and windows 11.
   * windows 10



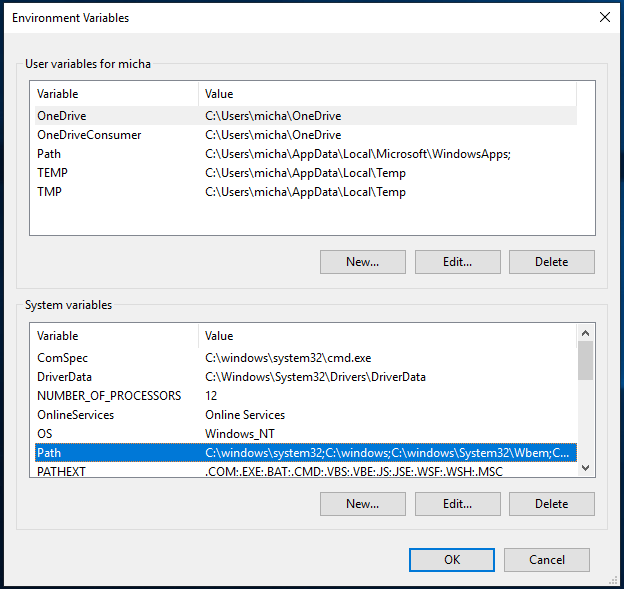
* + Note if you have Windows 11, I searched for environment and selected Edit the system environment variables.



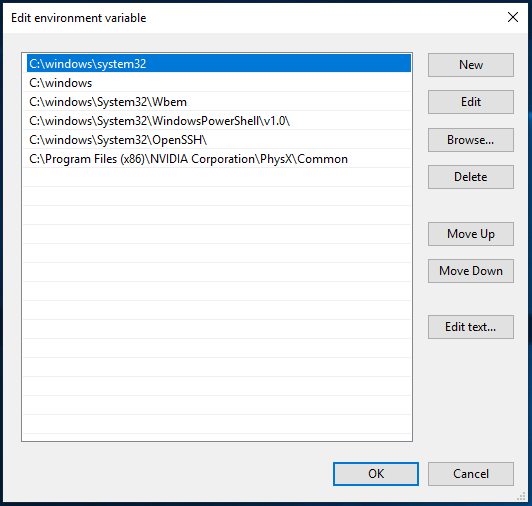
1. In the System Properties window, click the “**Environment Variables…**” button



1. The **Environment Variables** window will display. In the lower window - "System Variables" - find "**Path**" and single click on it to highlight



1. Once "Path" is highlighted, click **Edit** and this window will appear.



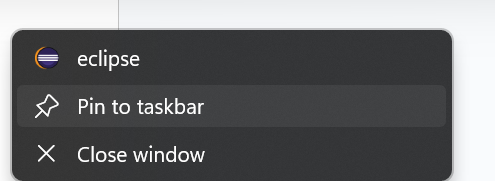
1. Click **New**. Paste the path for the**JDK bin** directory into the little window adding it to your path.
   * Click on the little window
   * Use CTRL-V to paste the path saved in the buffer into the little window
   * Close all those windows by clicking OK. You can close this window.



### 

### Step 4: Set Eclipse Windows

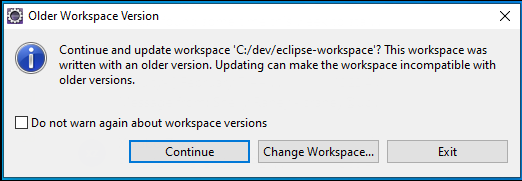
1. Go to the directory where you extracted Eclipse (on my system that would be **c:\JavaDev\eclipse**).
2. Double click the **eclipse.exe** file
   * If you don’t see the **.exe** file extension – double click the file with the round purple icon
   * To see file extensions in file explorer, click “**View**” then check “**File name extensions**”
   * You can right click eclipse and pin to taskbar to make it easier to launch



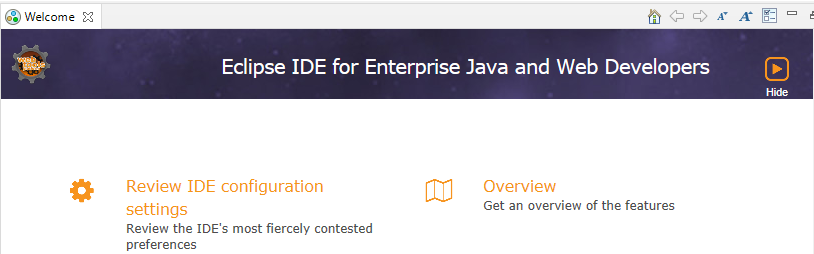
1. Eclipse will ask where you want your **workspace**. A **workspace** is where the .java and .class files for your class work and assignments will be stored. You will put this in the EclipseWorkSpaceFolder you created in your git repo ([Part 3: Git Repos Organize in Folders](https://docs.google.com/document/d/1mqMr_xE4cPg7IMeNPt0dBCIb7HCUp41SqKb_H-I45Sw/edit?tab=t.0#heading=h.xrdyourjd4ap)) so you can version the files using git and back up on the GitHub server. Click use this as the default.



1. Click launch **Note: if you’re updating the version of Eclipse on your machine you will see the following dialog**
   * Note: **Before you update Eclipse** to a newer version make a backup copy of your current workspace in case you run into issues.



1. The Welcome screen for Eclipse will appear. Click the **Hide** icon on the right side.



1. You are now set up to create, compile and run Java programs! Go to [Creating a Java Program in Eclipse (Windows and Mac)](#_heading=h.lnxbz9)

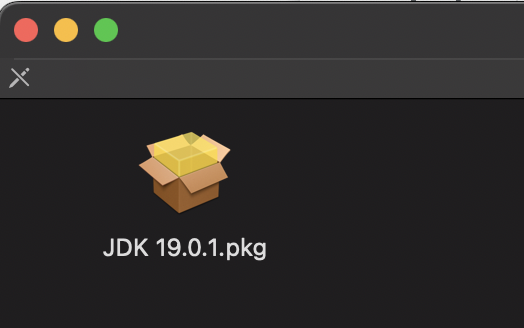
## Mac Set Up

### Step 1: Download Java SE Development Kit

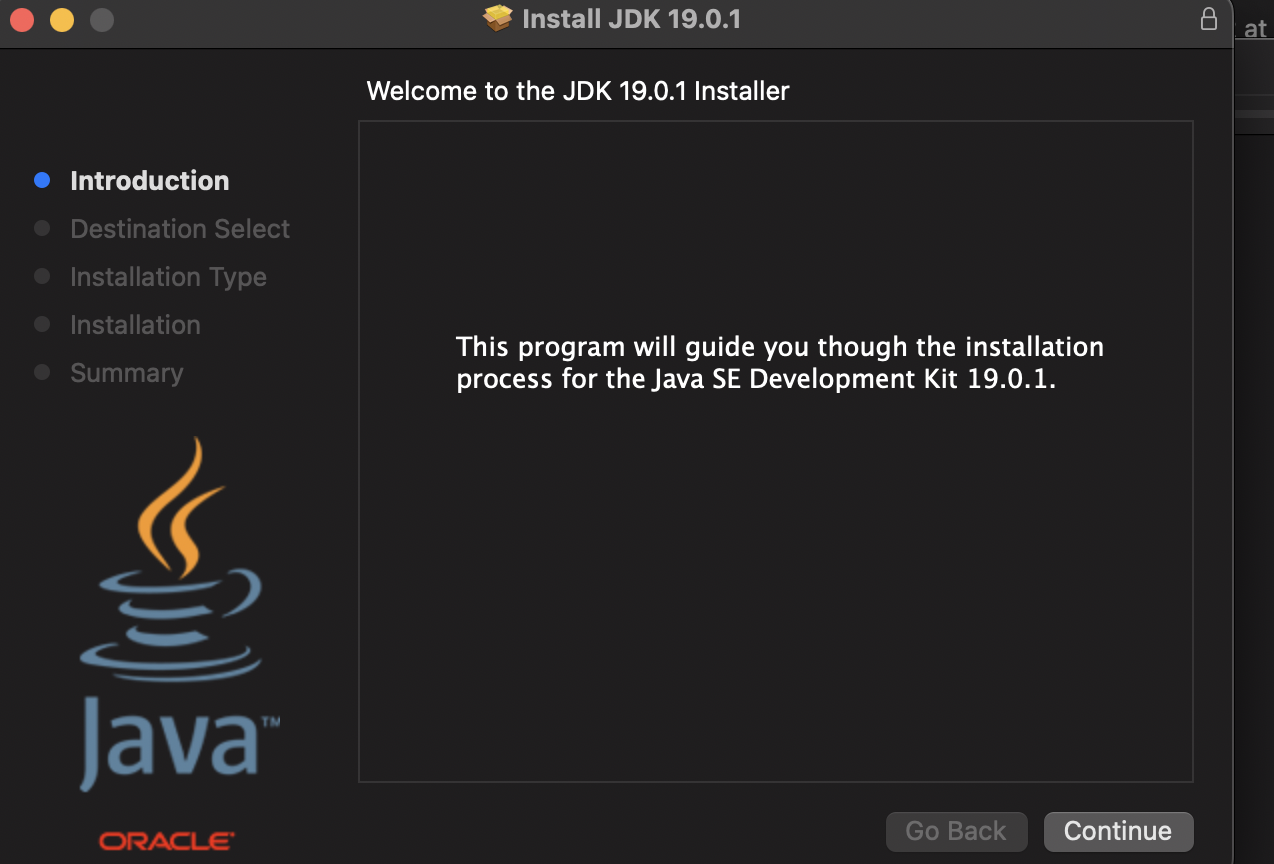
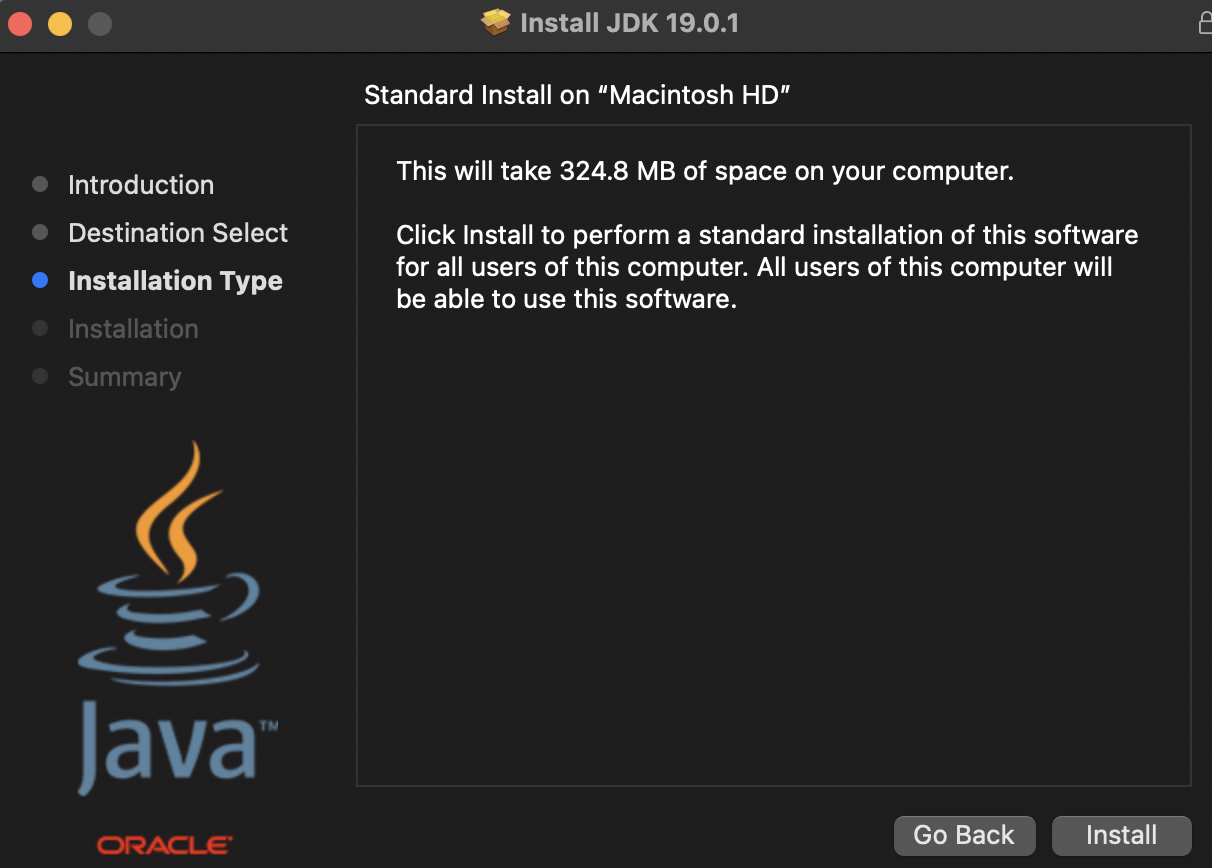
1. First you need to find out if your Mac is using X64 (x86-64) architecture or ARM 64 (M1 or M2).
   * Click apple and select about this Mac. If the chip is M1 or M2 you are using ARM 64.



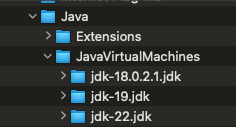
1. Here are the versions I will be using for you to download.
   * if using M1 or M2 chip download the [ARM 64 download this file](https://drive.google.com/file/d/1hmnYDoXS1CeAaJLRx3lsnGco3SKVCkLy/view?usp=drive_link)
   * otherwise download this file [jdk-22\_macos-x64\_bin.dmg](https://drive.google.com/file/d/1ze2QNxNhoAvfE8KLg9HsqNq__MLWfYCQ/view?usp=drive_link)
2. Double click the JKD pkg



1. The install window will appear. Click continue and then install. Enter password to install the software. Then click close

1. Here is where the JDK is located. You may have more than one installed. This will be addressed later.



### Step 2: Set Up Eclipse

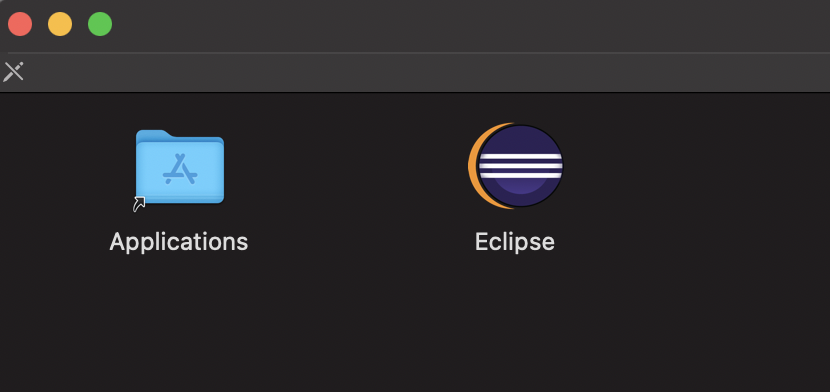
Your Workspace files will be stored in git and not in the Eclipse Folder

1. Go to: [Eclipse IDE 2024-06 R Packages](https://www.eclipse.org/downloads/packages/release/2024-06/r)
2. Scroll down to **Eclipse IDE for Enterprise Java and Web Developers** (DO NOT use the installer)
3. On the right-hand side, click “Mac OS X86\_64 or if new core processor AArch64”)

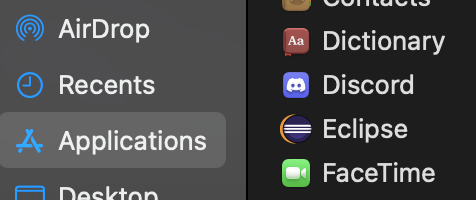




1. You should end up on a page with a big download button - click that button!
2. Go to the download location. Double click dmg file.
3. Drag eclipse into your application folder.

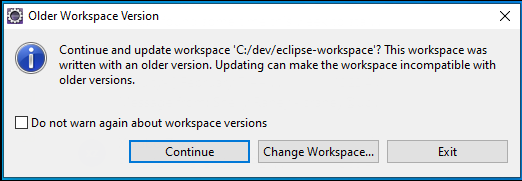


1. You can go to your applications folder to open Eclipse



**Note: if you’re updating the version of Eclipse on your machine you will see the following dialog**

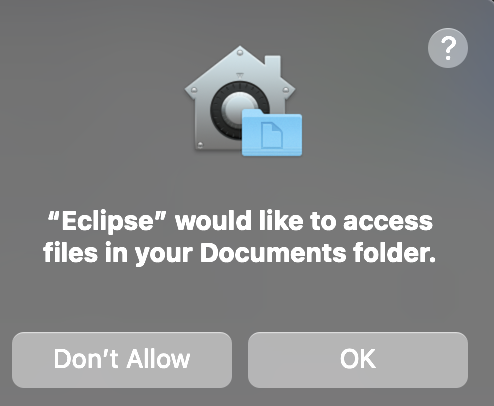
* + **Before you update Eclipse** to a newer version make a backup copy of your current workspace in case you run into issues.



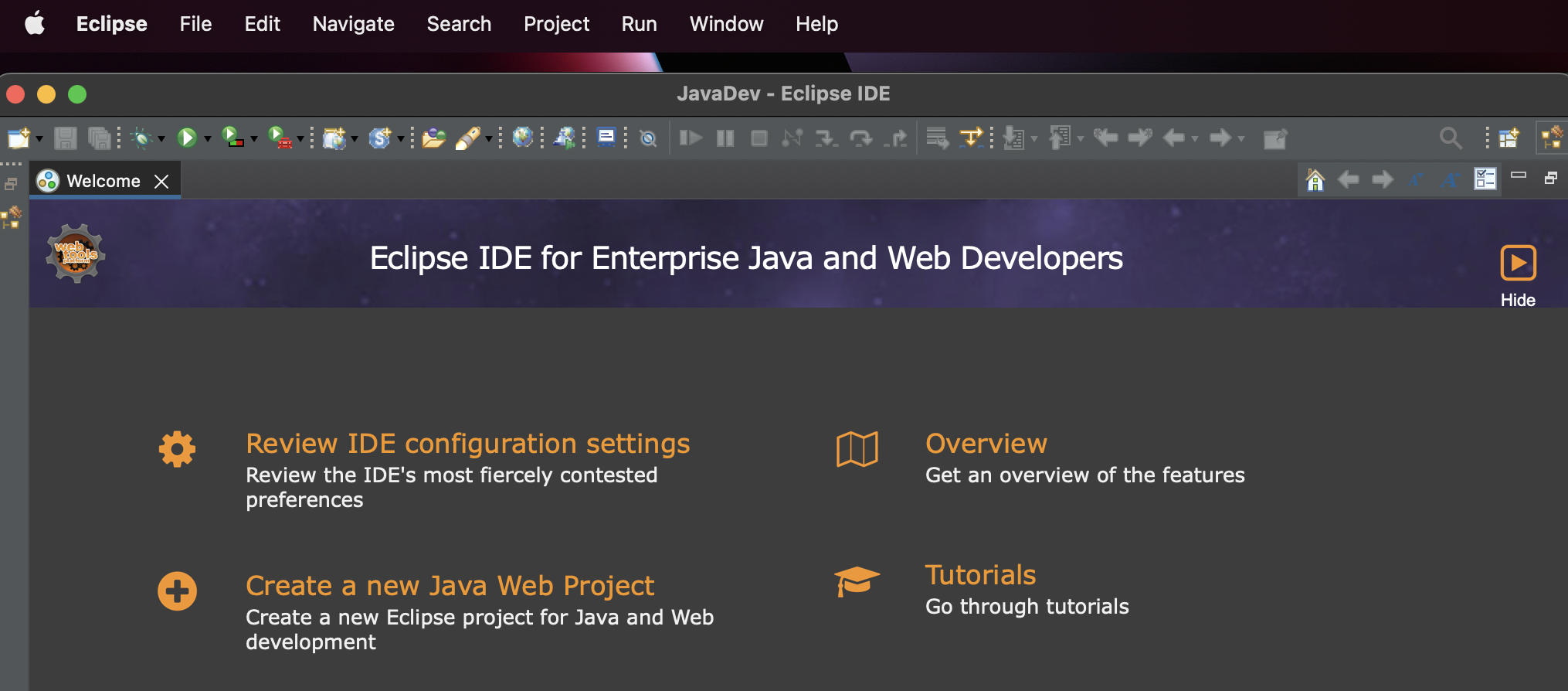
1. Eclipse will ask where you want your **workspace**. A **workspace** is where the .java and .class files for your class work and assignments will be stored. You will put this in the EclipseWorkSpaceFolder you created when setting up git repo: [Part 3: Organize in Folders](https://docs.google.com/document/d/1mqMr_xE4cPg7IMeNPt0dBCIb7HCUp41SqKb_H-I45Sw/edit?tab=t.0#heading=h.xrdyourjd4ap). Click use this as the default.



1. Click ok to allow eclipse access.



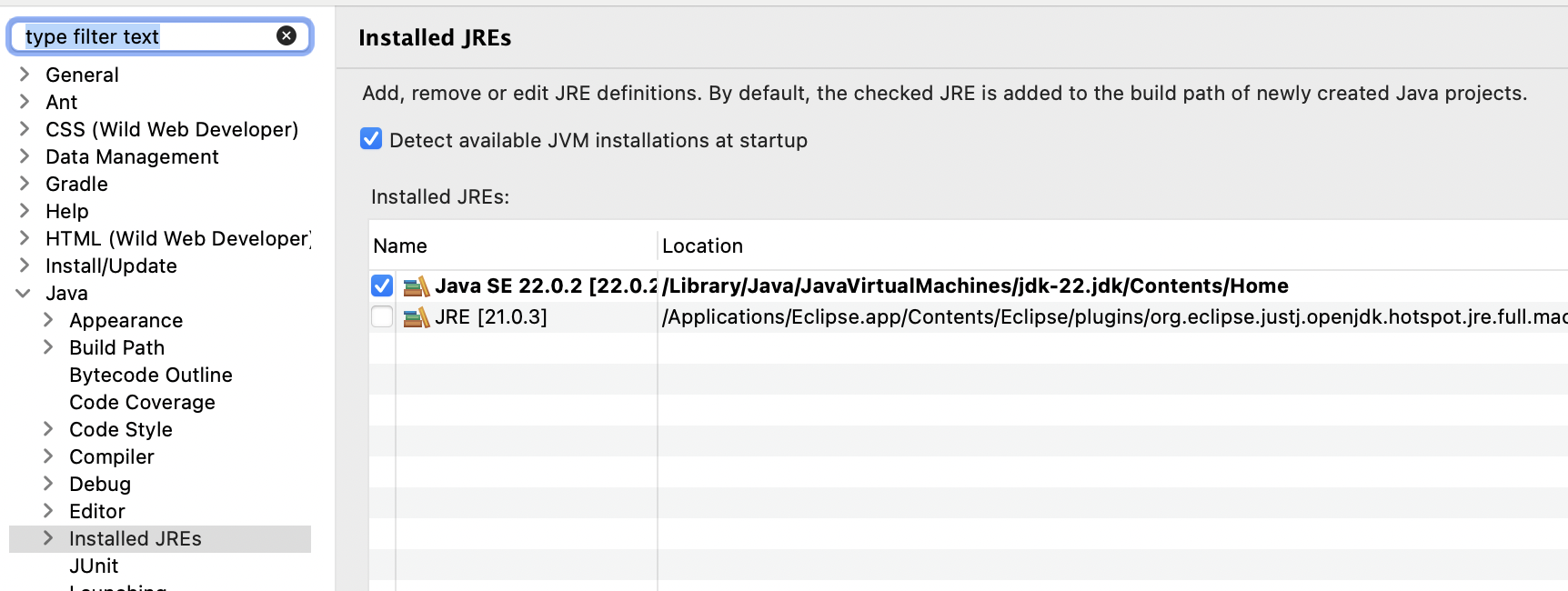
1. The Welcome screen for Eclipse will appear. Click the **Hide** icon on the right side.



1. Set up JRE. Go to Eclipse menu, select Preferences and navigate to Java -> Installed JREs



1. Scroll down to Java->Installed JREs. Select Java SE22. Click apply and close. THIS STEP IS A PLACE MANY DO NOT SELECT SE - you do not want JRE.



1. Now that the Java SE is set go to [Creating a Java Program in Eclipse (Windows and Mac)](#_heading=h.lnxbz9) below.

# Creating a Java Project and Program in Eclipse (Windows or Mac

Here are the steps for creating a simple Java program to get started

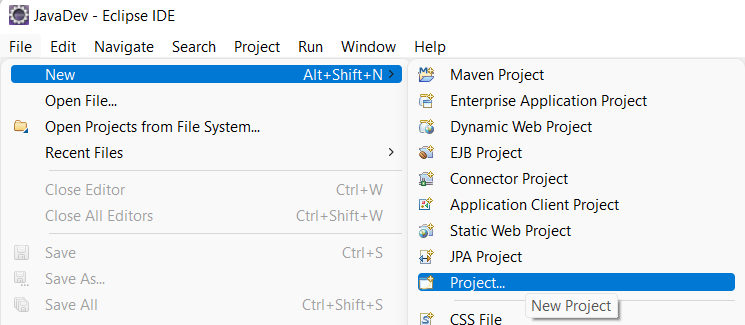
* Quick look at programming from a general view
* Everything will be covered in more detail in later lectures

## Step 1: In Eclipse Create a Java Project

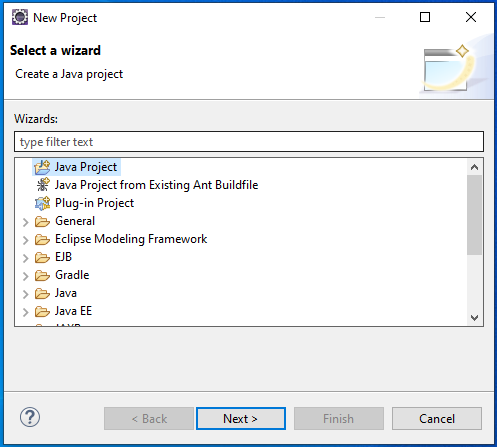
Take your time!

It is important to organize your work. One way is to create Java Project folders based on the module.

1. The first time you create a project you will see a slightly different menu than on future creations.
2. On the top menu, select **File->New-> Project** to display the New project dialog



1. In the New Project dialog, select **Java Project** then click **Next**.



1. Set the following in the **Create Java Project** dialog and then click **finish**. You will use this same process to create future projects for other class assignments. You may need to configure JREs. DEB WILL ADD INFO.

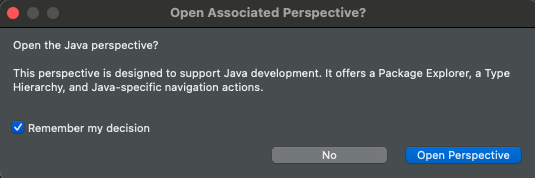


1. If you see the following dialog – click “OK”

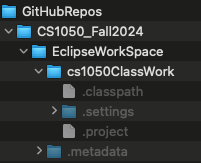
Graphical user interface, text, application

Description automatically generated

1. You may also see a dialog that asks you about the Java perspective. Click Remember by decision and click “Open Perspective”



1. If Welcome appears close it
2. If you go to your local repository you will see the project was created in the EclipseWorkSpace



**IMPORTANT NOTE:** before proceeding please read:

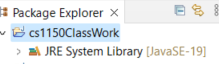
* + If there is a red **!** next to the project name in Eclipse, your environment is **NOT** set up properly. See the [troubleshooting section](#_heading=h.2bn6wsx) at the end of this document for simple steps to fix this **before you create a Java class**. If you don’t resolve this issue, you will be unable to get your code to work. After you fix this go to the next step to create a java class.



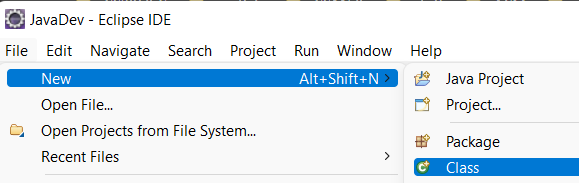
## Step 2: Create, Compile and Run a Java Class File

You will create a new java class in the project folder you want. You must do the following every time you want to create a new java source file.

1. Make sure you have selected the project folder you want to create it in, selected.

****

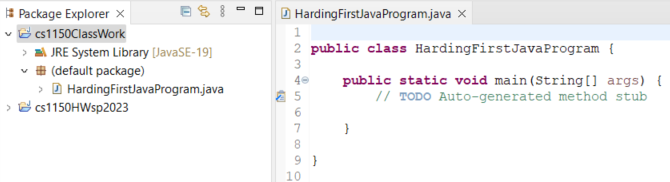
1. Go to **File->New->Class** (or **File->New->Other** then pick **Class** in the dialog)



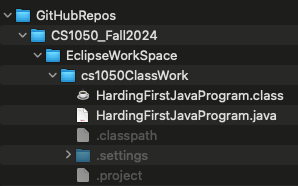
1. Set the following (see screen below)
   * Source Folder – should be filled in with the name you gave your project
   * **Package should be empty**
   * Name - name for file starts with capital letter
   * Check: **public static void main(String[] args)**
   * Uncheck: Inherited abstract methods
   * Check Generate Comments
   * Click Finish



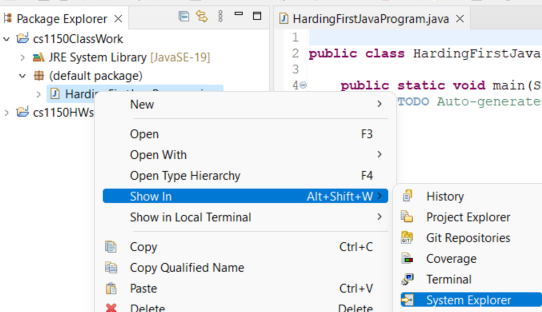
1. In Eclipse, on the left side you will see in the Package Explorer panel and your code in the middle.



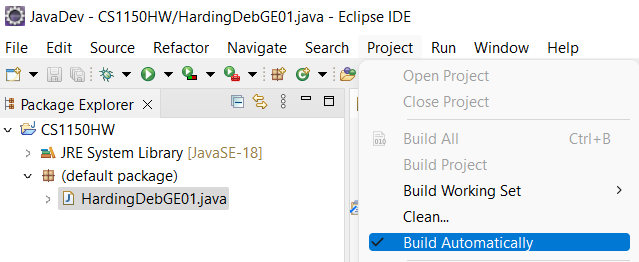
1. You will repeat this step #3 (creating a java class) each time you want to create a new file.
2. Go to your project located in you local repository



Another way to get to the directory is to right click the file then go to Show In -> System Explorer



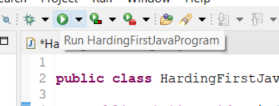
1. Eclipse automatically builds the .class file In Eclipse, if you go under Projects, you will see “**Build Automatically**” is selected. This is how the .class file was created automatically for you.



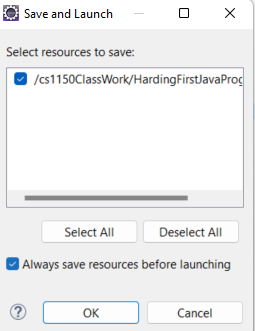
1. Add code to your first file. **System.out.print("Hello Java");**

| public class HardingHello {  /\*\*  \* @param args  \*/  public static void main(String[] args) {  // Print hello  System.out.print("Hello");  }  } |
| --- |

1. To run your program Select "Run" from the top menu



1. If the **Save and Launch** dialog appears, select “Always save resources before launching” so you do not need to see this every time.



1. You should see the output in the **console window** on bottom of the screen

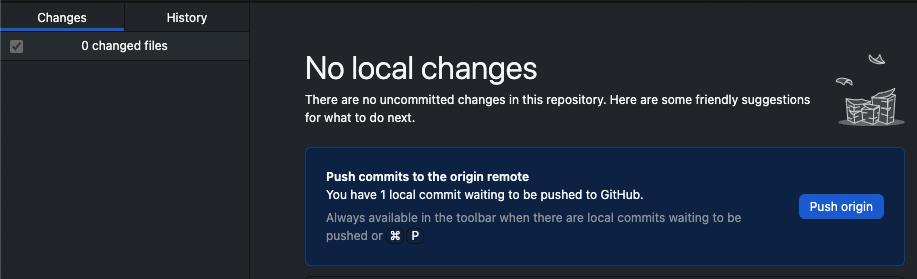


1. Celebrate!

## Step 3: Github Desktop: Version and Backup

1. Go to your github desktop and you should see the following. This is showing your new .java file. You need to version the file locally by committing. Put



1. Next you will push the update to the github server. 

## Troubleshooting Eclipse

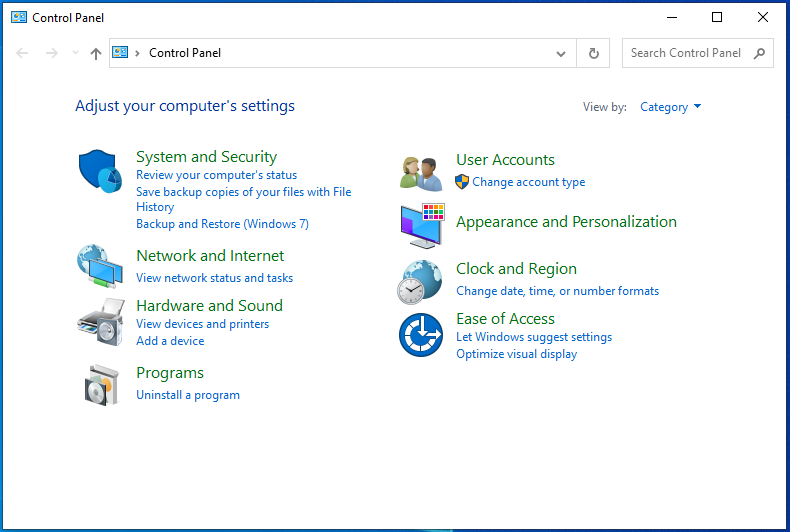
This section covers two possible issues that you might run into

* **Exit code 13 error**
* **Project has a red !**

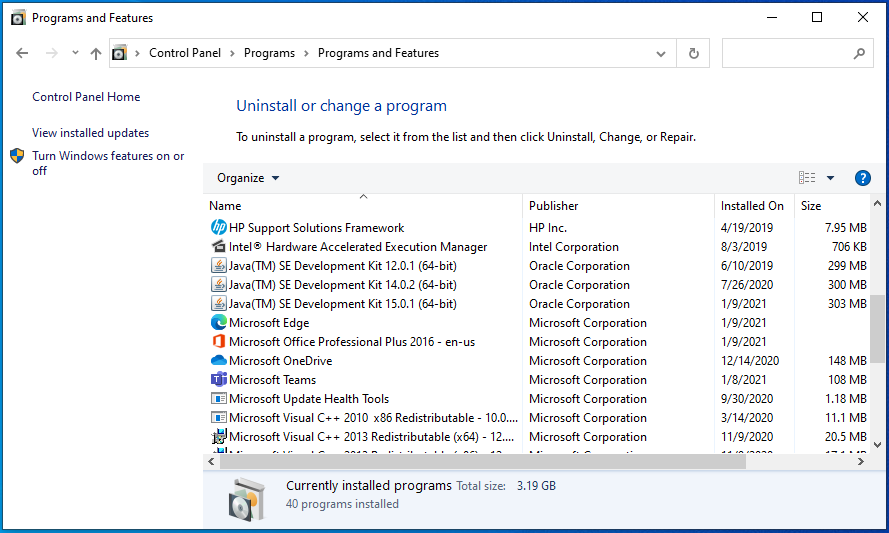
### Issue #1: Exit code 13 error

This error occurs when there is more than one version of the JRE on your system.

1. Note that you have 2 directories on your C drive:
   * + Program Files
     + Program Files (x86)
2. You correctly installed the JDK into the Program Files
   * + The Program Files directory is the correct location for the JDK so don’t mess with it.
3. Exit code 13 occurs when there is a java directory in Program Files (X86)
   * + If you have a **Java** directory in Program Files (x86), Eclipse will use it instead of the one you just installed in Program Files.
     + You need to **UNINSTALL** the java directory in Program Files (X86)
     + Uninstall is different from deleting, so DO NOT delete the directory.
4. To fix this issue:
   * + **Uninstall** the JRE in the Program Files (X86) directory
     + The new JDK that you just installed already contains the JRE.
     + Open the Control Panel
     + Select under Programs “Uninstall a program”



1. You next screen will appear with a list of all the programs installed on your system
   * + Locate the JRE
     + I don’t have the JRE on my system so the screen grab only shows the JDK’s so I can’t show what this looks like.
     + Once you locate the JRE, select it, then select uninstall



* You should now be able to start Eclipse.

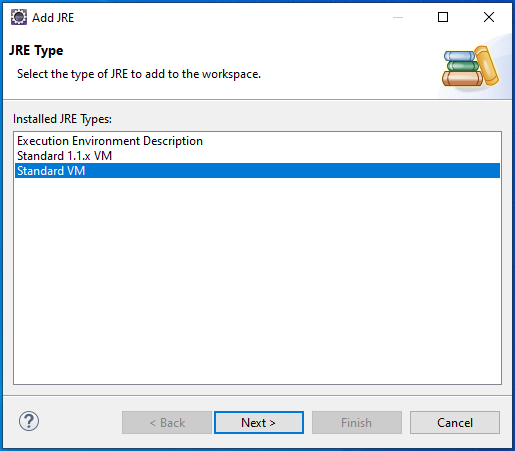
## 

### Issue #2: Project has a red !

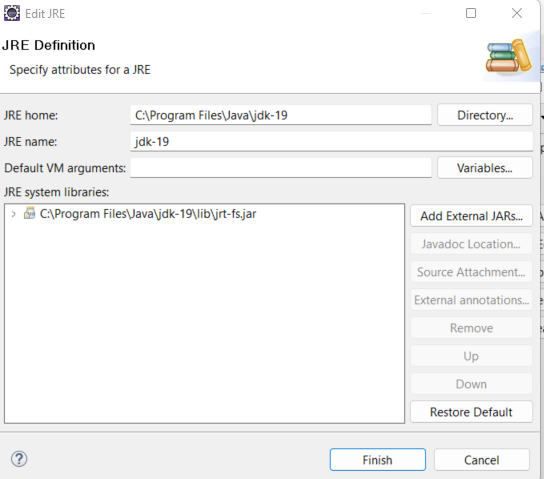
1. A project with a red **!** indicates that Eclipse needs the new JDK.
2. Eclipse will not work properly until you fix this issue!
3. In Eclipse, on the top menu, go to **Window -> Preferences**
4. In the preference dialog, click arrow **>** next to **Java** to open Java, then select **Installed JRE’s**



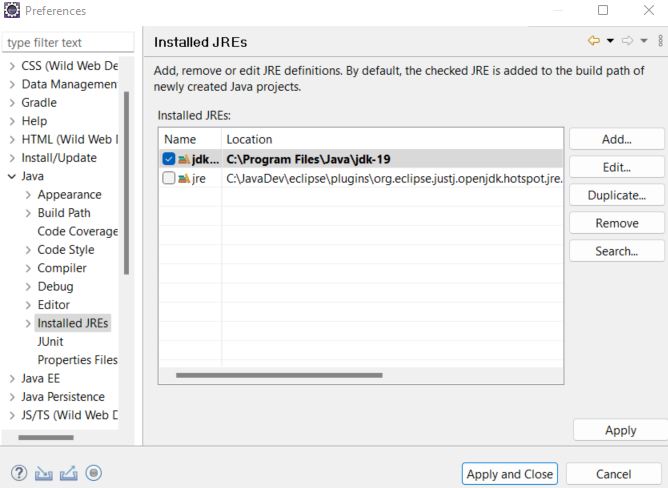
1. Click **Add**. In Add JRE window, select **Standard VM** then next



1. In the updated **Add JRE** screen, to the right of JRE Home, click **Directory...**
2. Browse to C:\Program Files\Java\jdk-22
3. Highlight jdk-22 and click **Select Folder**
4. The Add JRE dialog will now be filled in, click **Finish. In image below it should be jdk-22 instead of 19.**



1. You will now see in Installed JREs dialog a new entry with **jdk-22**
2. Select the box to the **left** of the jdk-22



1. Click **Apply and Close** button
2. The project should no longer have the red ! Return to [Create a Java Class](#_heading=h.z337ya) If you still have an issue, send me an email to get help.

