#Development Environment

The ability to develop code quickly is crucial. Having the right tool helps you not only with speed, but with formatting and correctness as well. A **development environment** is a unique set of tools and features that a developer can use to write software. Some IDEs (Independent Developer Environments) are better for certain languages/projects than others. We've already talked about the built-in IDEs here on coding rooms, but let's go a little further.

#Editors

One of the most crucial tools for software development is your editing environment. An editor is where you write your code and sometimes where you run your code.

Developers rely on editors for their helpful features, including:

- **Debugging:** Helps you discover bugs and errors by stepping through code, line by line. Some editors have debugging capabilities, or can be customized with add-ons to debug specific programming languages.
- **Syntax highlighting:** Adds colours and text formatting to code, making it easier to read. Most editors allow customized syntax highlighting, which is imperative if you plan to write code quickly.
- **Extensions and integrations**: Add specialized features that provide access to other tools that aren't built into the base editor. For example, many developers also need a way to document their code, to explain how it works, or to install a spell check extension to check for typos. Most of these additions are intended for use within a specific editor, and most editors come with a way to search for available extensions.
- **Customization**: Most editors are customizable, which allows developers to create their own unique development environments. Many editors also allow developers to create their own extensions.

#Java Development Environments

The tool that we use to compile a Java source file into a Java class file is called a **compiler**. Most programmers use an **Integrated Development Environment** (IDE) that has the compiler built-in and helps them write, compile, run, and debug their programs. Here are some IDEs that can be used as Java Development Environments (JDEs).

#Eclipse

Eclipse (https://www.eclipse.org/downloads/packages/installer) is what many professional Java programmers use. It is especially popular among banking companies, such as RBC and TD.

InfoWarningTip

We will be officially using Eclipse for Academy, and will be setting it up in the next lesson. If you have experience with another IDE and would prefer to use it, you may do so. However, the instructions throughout Academy's Java development phase will not be written for other IDEs, and you must assume all of the difficulties/risks related to using another IDE on your own. Using another editor may also make it a bit harder to get along at your future job placement–we've chosen Eclipse for a reason, and we recommend you follow along.

#IntelliJ

IntelliJ (https://www.jetbrains.com/idea/) is a free Java IDE from JetBrains that many java professionals use. It is a little easier to configure than Eclipse. Here is a guide on how to set up IntelliJ: https://www.jetbrains.com/help/idea/install-and-set-up-product.html.

#Spring Tool Suite (STS) IDE

Sprint Tool Suite is a special Eclipse-based IDE used for creating *Spring* enterprise applications. We will use this in the second month of our bootcamp for building some Spring web-apps. We will walk through setting this IDE up together when we need it.

#Netbeans

Netbeans (https://netbeans.org/) is one of the original Java IDEs. Here is a tutorial on how to set it up: https://netbeans.org/kb/docs/java/quickstart.html.

#Alternative IDE: VS Code

Visual Studio Code, also commonly referred to as VS Code, is a source-code editor made by Microsoft to support software development in various languages, of which Java is just one. It typically needs a bit more customization to set it up for Java programming. Here is a link to download VS Code: https://code.visualstudio.com/.

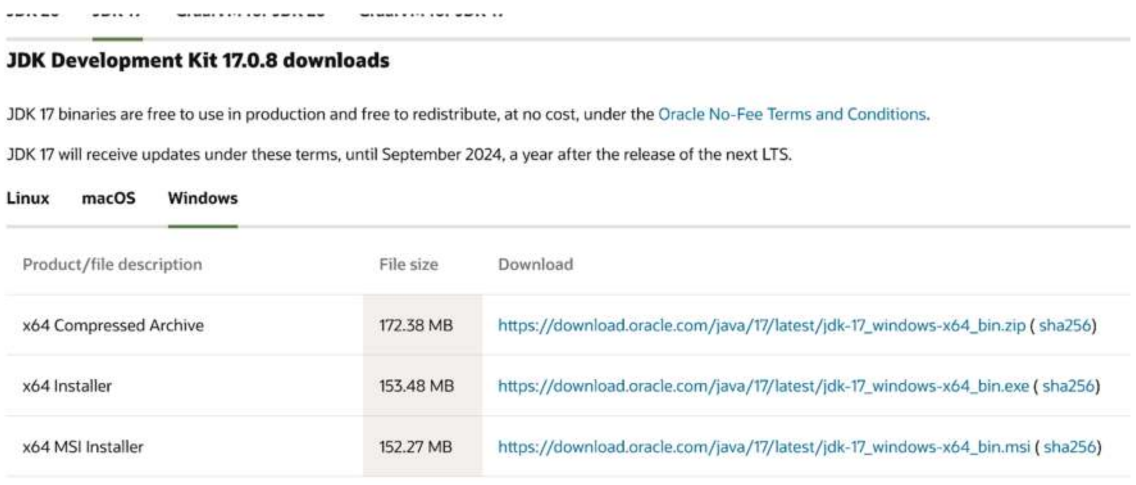
What is Eclipse?

Eclipse is an integrated development environment (IDE) for developing applications using the Java programming language and other programming languages such as C/C++, Python, PERL, Ruby etc. Because it's commonly used in the types of jobs that Academy is preparing you for, it will be our IDE of choice throughout the Java portions of this bootcamp.

#Setting up Eclipse IDE

1. Check if Java is pre-installed on your system:
   a. Go to **Start -> Control Panel -> Add/Remove** programs, and check the list to see if Java is installed

2. If Java is not installed, download the latest version from this link. Download the installer for **Windows (x64 Installer)**. If Java is installed, you can skip to Step 7 below. Otherwise, continue down this list in order.


JDK Development Kit 17.0.8 downloads

JDK 17 binaries are free to use in production and free to redistribute, at no cost, under the Oracle No-Fee Terms and Conditions.

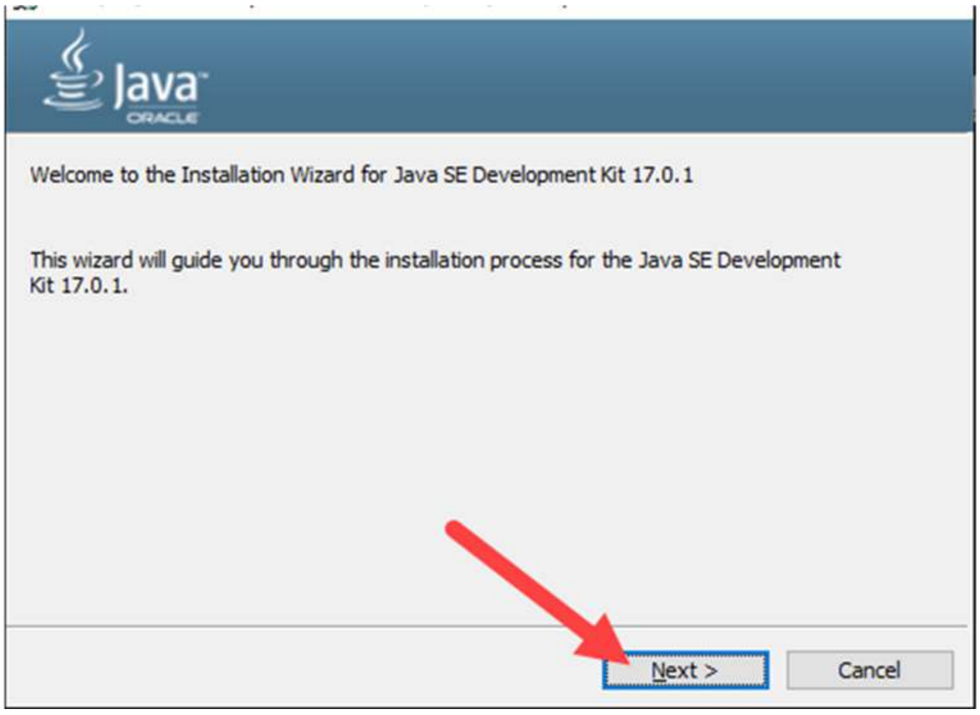
JDK 17 will receive updates under these terms, until September 2024, a year after the release of the next LTS.

Linux macOS Windows

Product/file description	File size	Download
x64 Compressed Archive	172.38 MB	https://download.oracle.com/java/17/latest/jdk-17_windows-x64_bin.zip (sha256)
x64 Installer	153.48 MB	https://download.oracle.com/java/17/latest/jdk-17_windows-x64_bin.exe (sha256)
x64 MSI Installer	152.27 MB	https://download.oracle.com/java/17/latest/jdk-17_windows-x64_bin.msi (sha256)

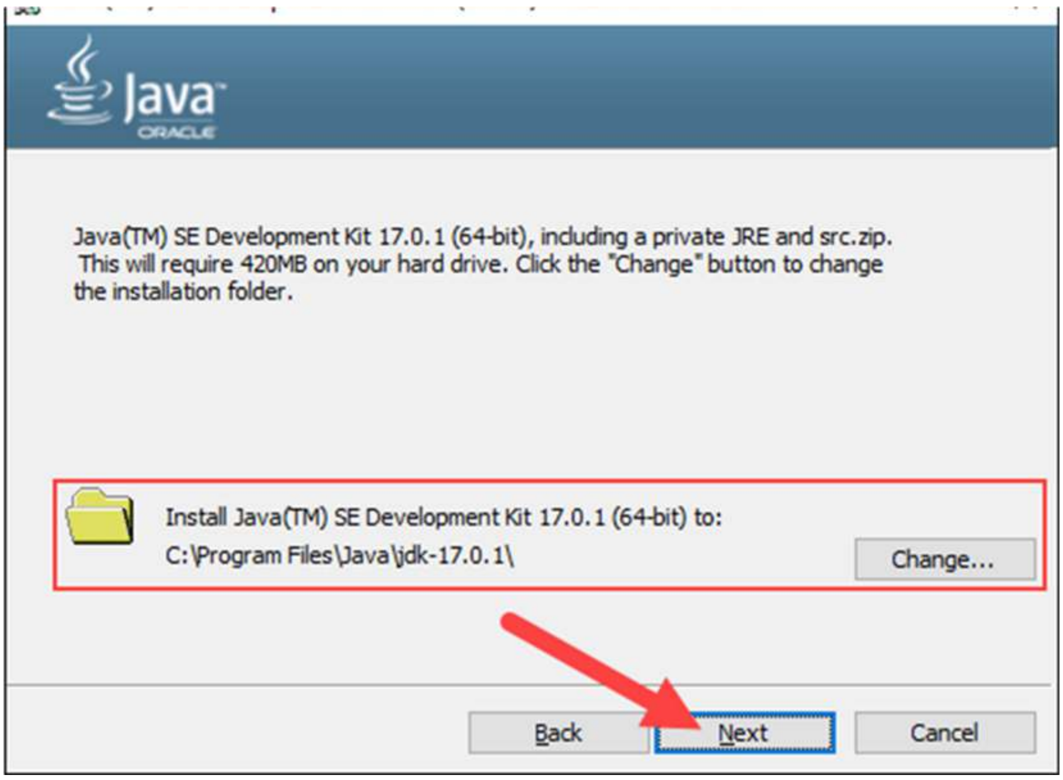
3. Run the installation file downloaded

4. Click **Next** to proceed to the next step


Welcome to the Installation Wizard for Java SE Development Kit 17.0.1

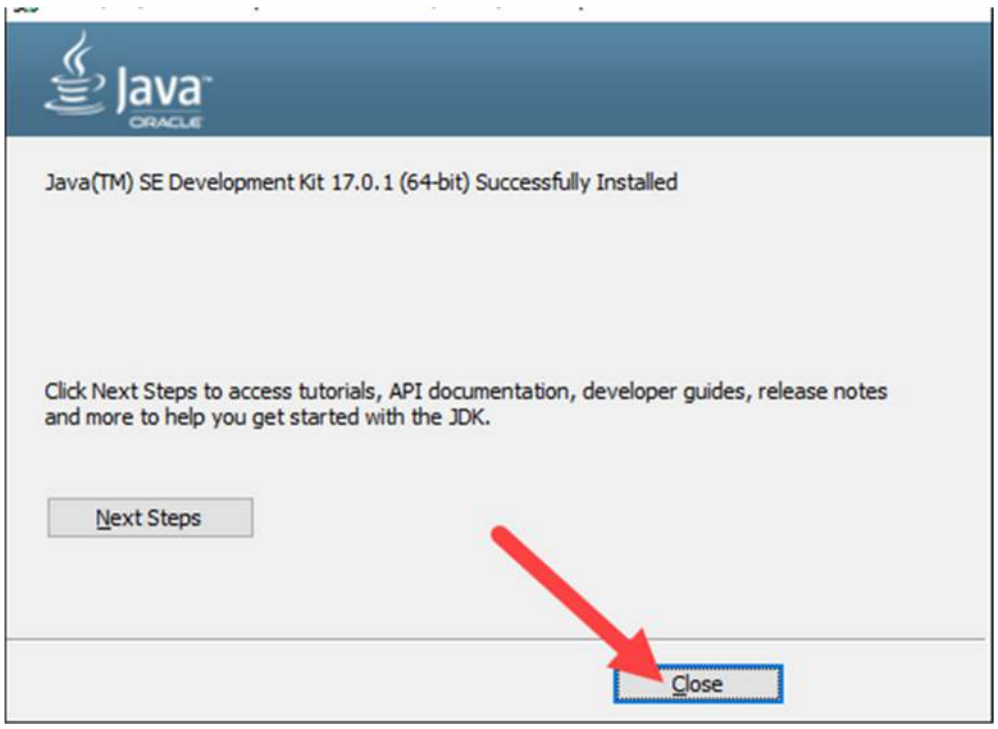
This wizard will guide you through the installation process for the Java SE Development Kit 17.0.1.

5. Choose the installation folder, and click on **Next**


Java(TM) SE Development Kit 17.0.1 (64-bit), including a private JRE and src.zip. This will require 420MB on your hard drive. Click the "Change" button to change the installation folder.

Install Java(TM) SE Development Kit 17.0.1 (64-bit) to:
C:\Program Files\Java\jdk-17.0.1\

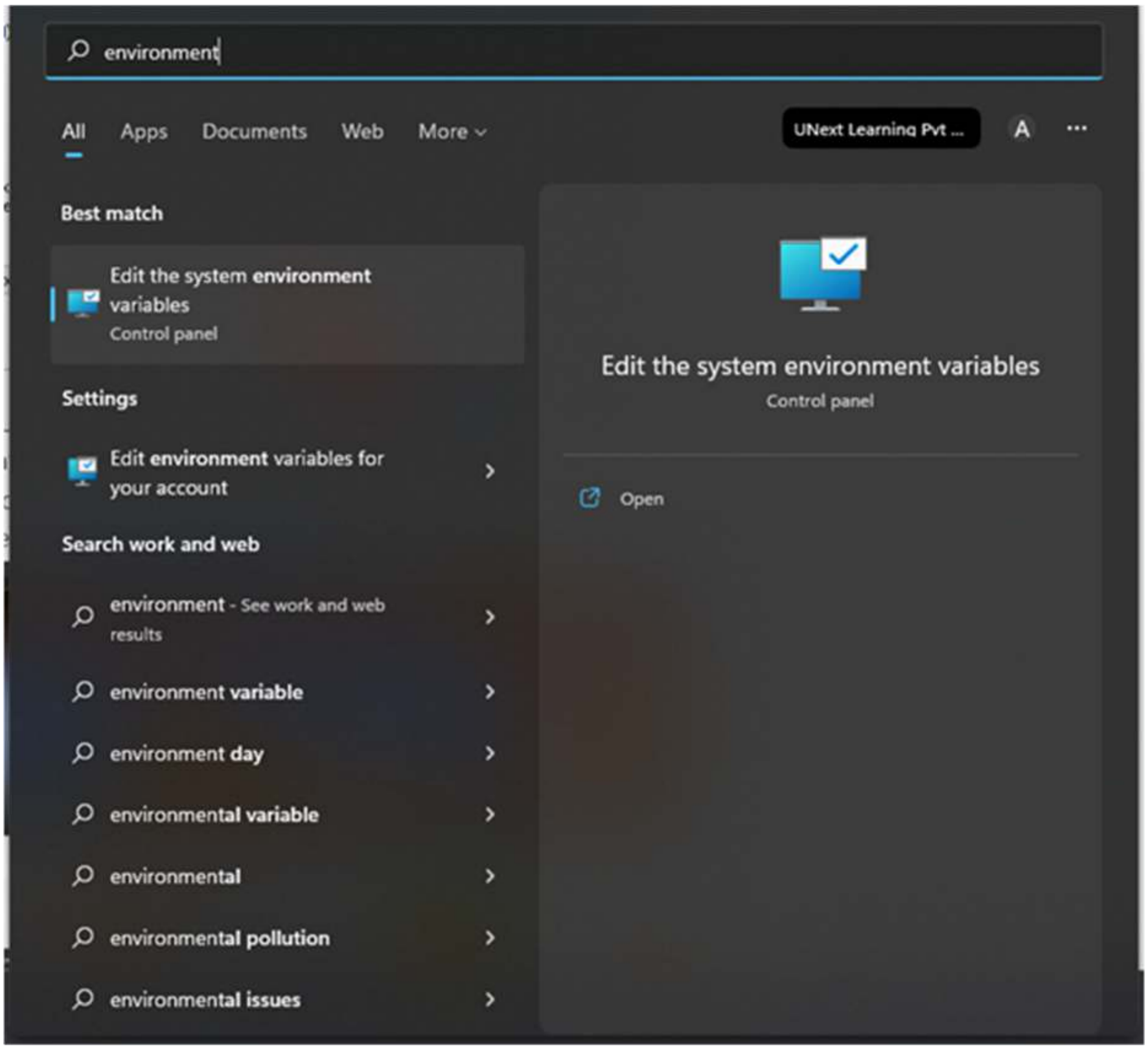
6. Once the installation is finished, click **Close**


Java(TM) SE Development Kit 17.0.1 (64-bit) Successfully Installed

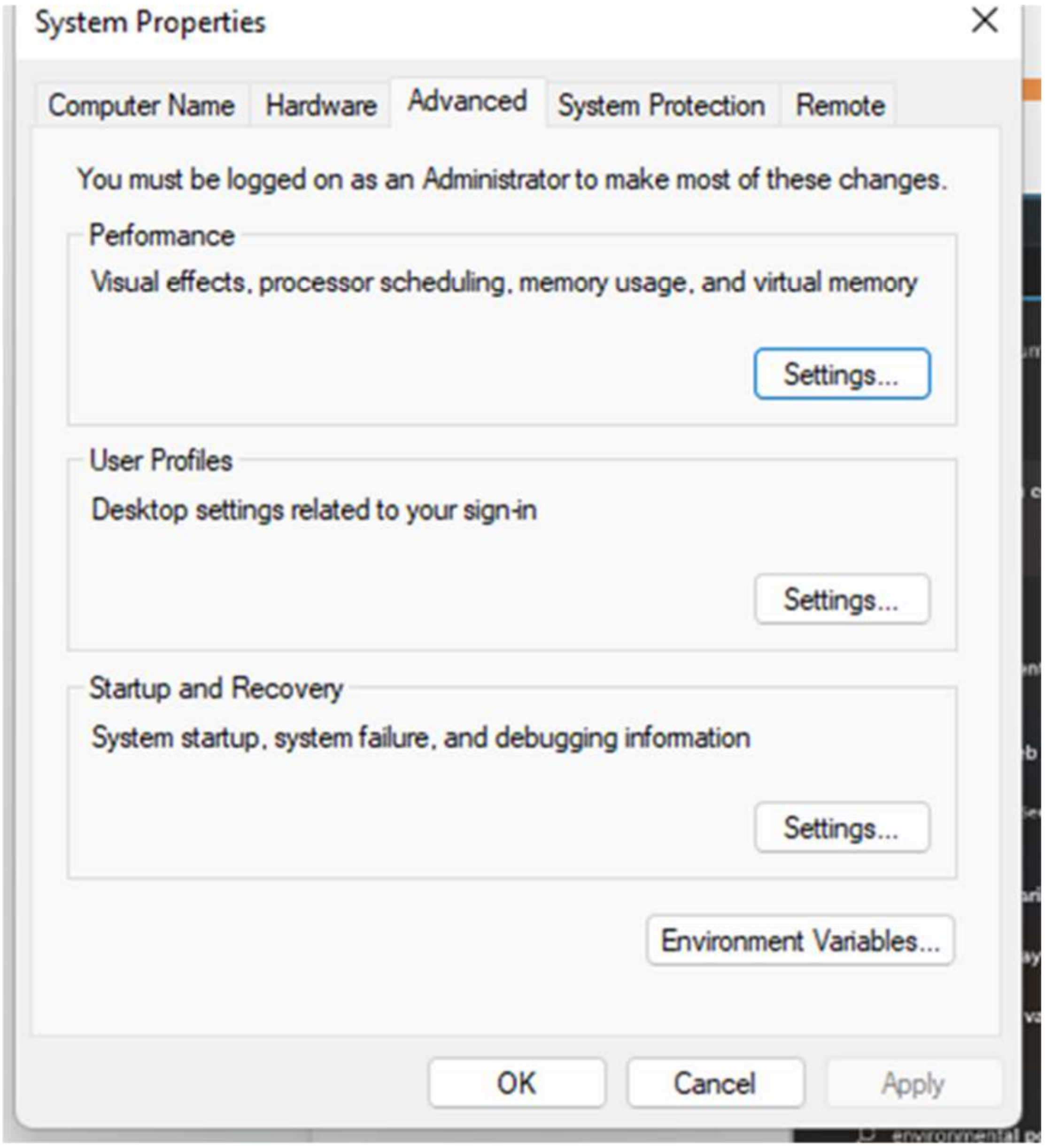
Click Next Steps to access tutorials, API documentation, developer guides, release notes and more to help you get started with the JDK.

7. Add Java's installation path to the system path

a. Go to the start menu and search for an environment variable, and click on **Edit the system environment variable**



b. Click on the **Environment Variables** Button


System Properties

Computer Name | Hardware | Advanced | System Protection | Remote

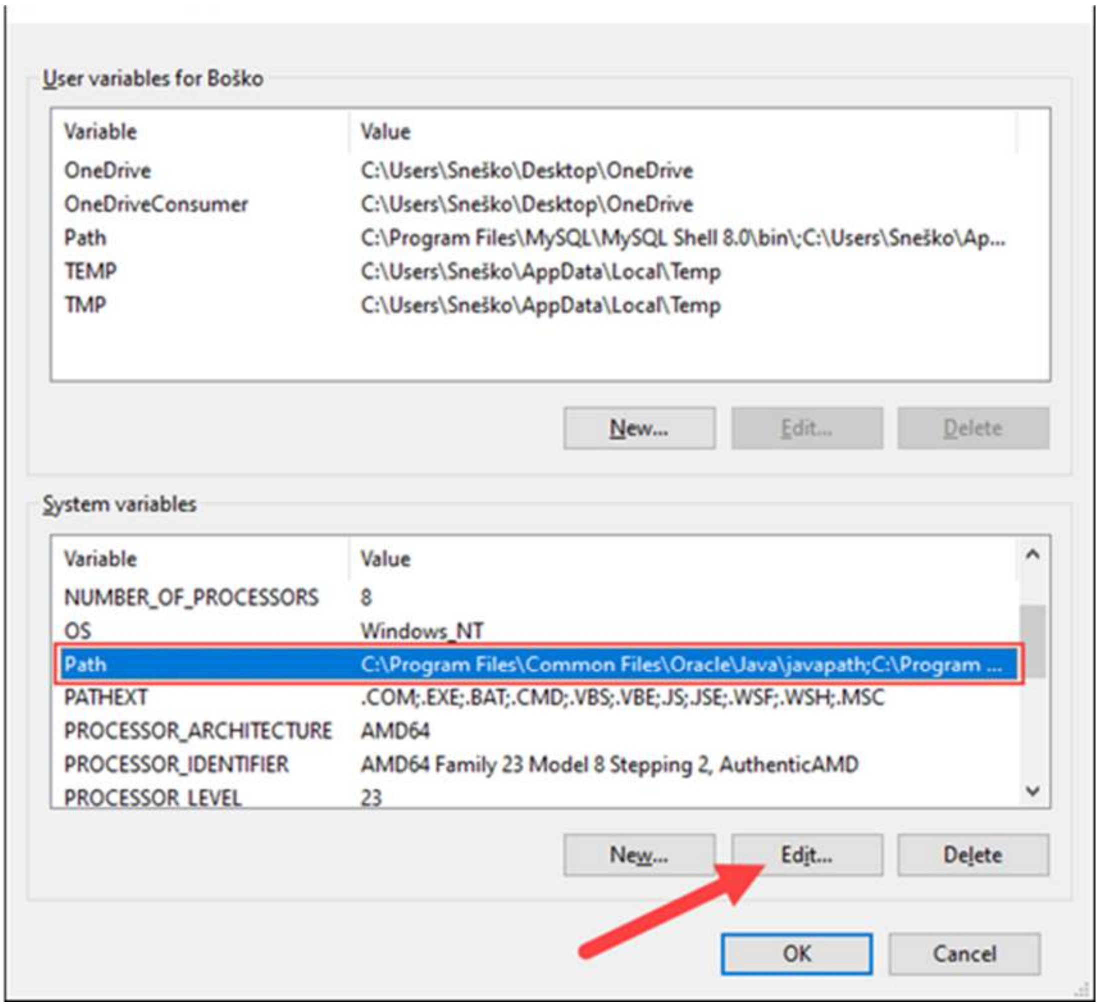
You must be logged on as an Administrator to make most of these changes.

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

User Profiles
Desktop settings related to your sign-in

Startup and Recovery
System startup, system failure, and debugging information

c. Under the system variable category, select the **Path** and click **Edit**

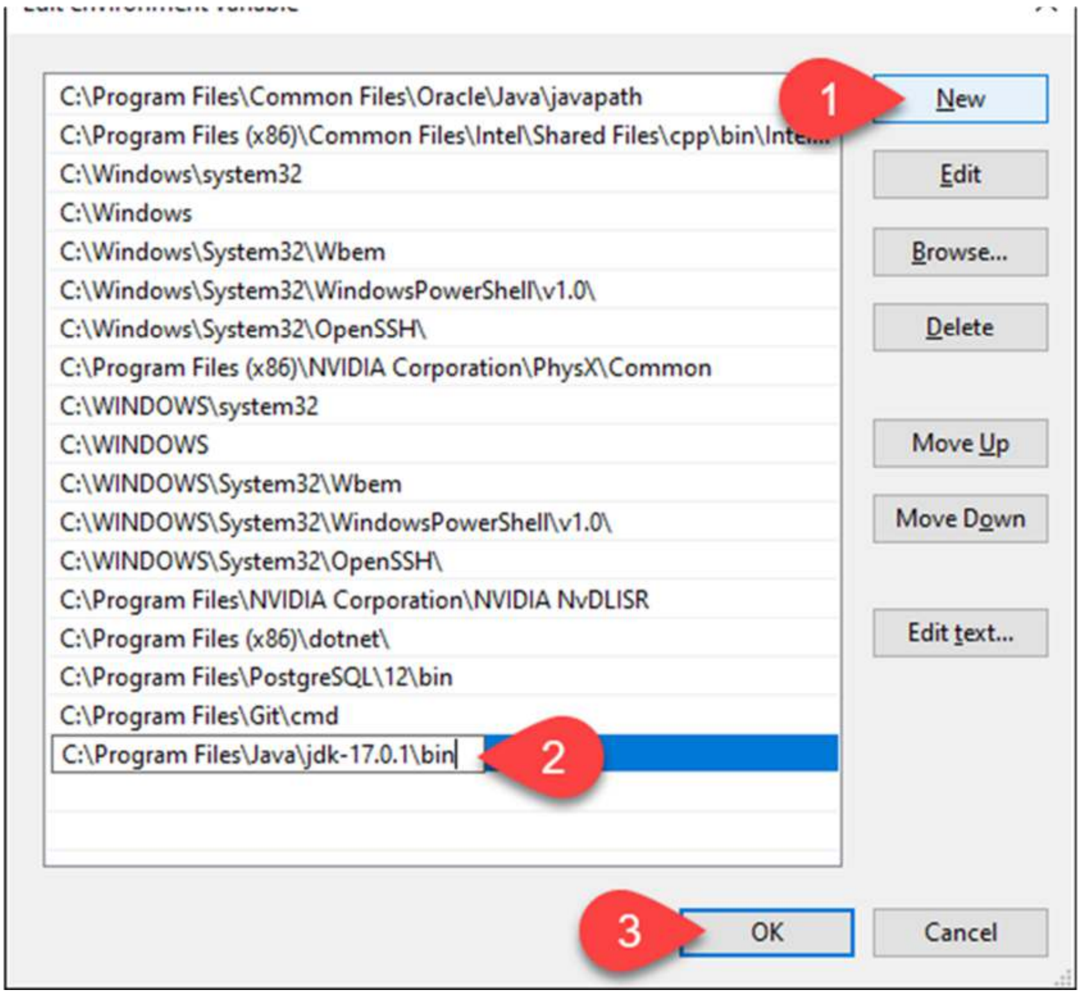

User variables for Boško

Variable	Value
OneDrive	C:\Users\Sneško\Desktop\OneDrive
OneDriveConsumer	C:\Users\Sneško\Desktop\OneDrive
Path	C:\Program Files\MySQL\MySQL Shell 8.0\bin\;C:\Users\Sneško\Ap...
TEMP	C:\Users\Sneško\AppData\Local\Temp
TMP	C:\Users\Sneško\AppData\Local\Temp

System variables

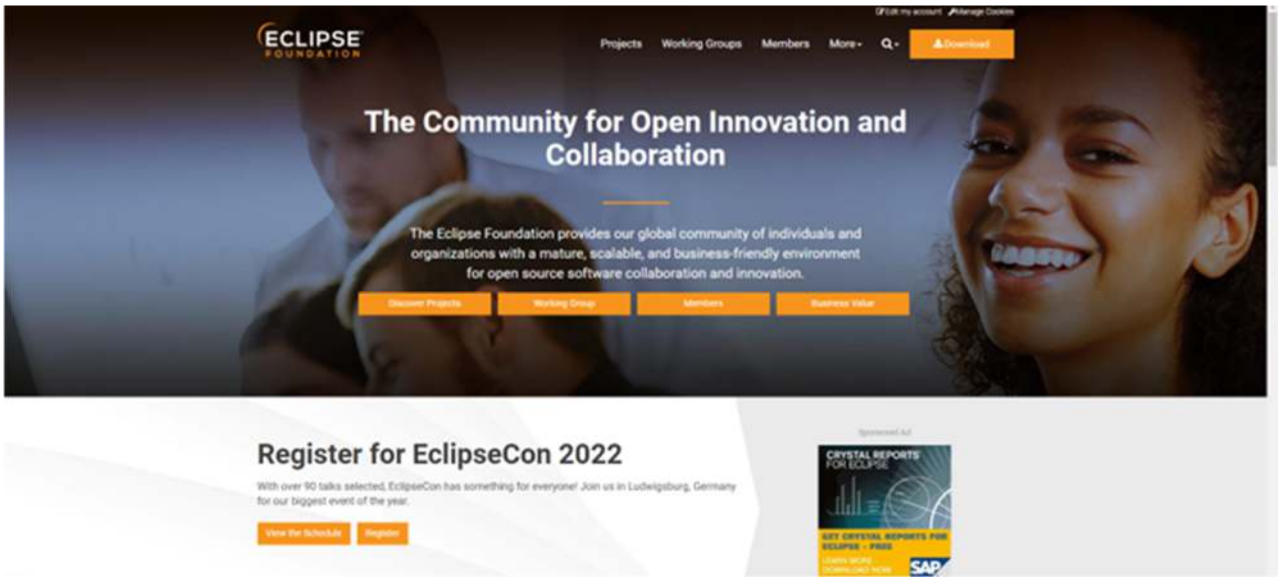
Variable	Value
NUMBER_OF_PROCESSORS	8
OS	Windows_NT
Path	C:\Program Files\Common Files\Oracle\Java\javapath;C:\Program ...
PATHEXT	.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH;.MSC
PROCESSOR_ARCHITECTURE	AMD64
PROCESSOR_IDENTIFIER	AMD64 Family 23 Model 8 Stepping 2, AuthenticAMD
PROCESSOR_LEVEL	23

d. Click the **New** button to enter the path of the java bin directory. The default path is usually: C:\Program Files\Java\jdk-17.0.1\bin. Once you've done this, click **OK**



```
C:\Program Files\Common Files\Oracle\Java\javapath
C:\Program Files (x86)\Common Files\Intel\Shared Files\cpp\bin\Inte...
C:\Windows\system32
C:\Windows
C:\Windows\System32\Wbem
C:\Windows\System32\WindowsPowerShell\v1.0\
C:\Windows\System32\OpenSSH\
C:\Program Files (x86)\NVIDIA Corporation\PhysX\Common
C:\WINDOWS\system32
C:\WINDOWS
C:\WINDOWS\System32\Wbem
C:\WINDOWS\System32\WindowsPowerShell\v1.0\
C:\WINDOWS\System32\OpenSSH\
C:\Program Files\NVIDIA Corporation\NVIDIA NvDLISR
C:\Program Files (x86)\dotnet\
C:\Program Files\PostgreSQL\12\bin
C:\Program Files\Git\cmd
C:\Program Files\Java\jdk-17.0.1\bin
```

8. Navigate to https://www.eclipse.org/



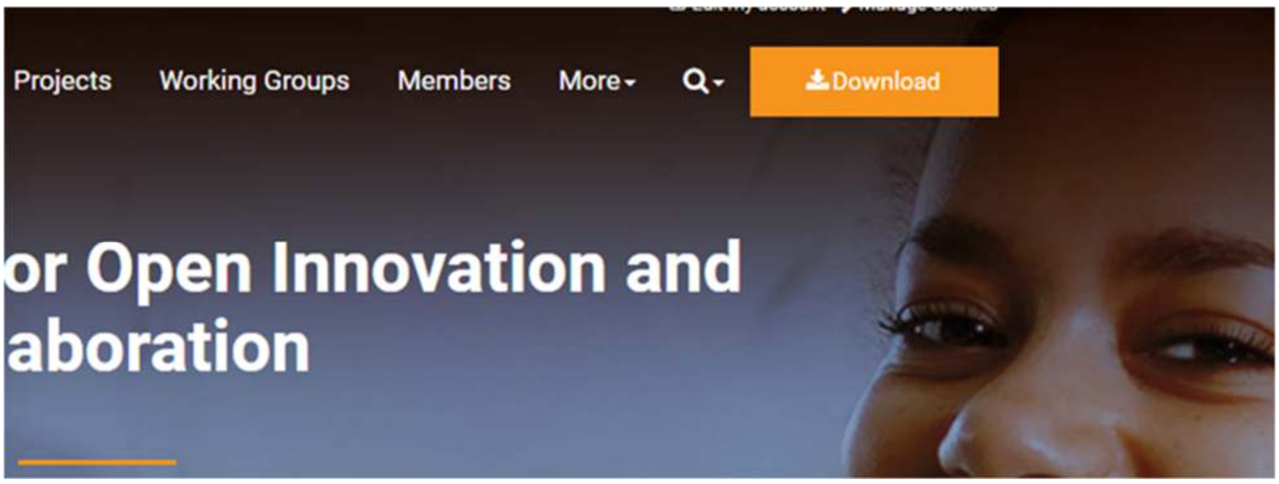
The Community for Open Innovation and Collaboration

The Eclipse Foundation provides our global community of individuals and organizations with a mature, scalable, and business-friendly environment for open source software collaboration and innovation.

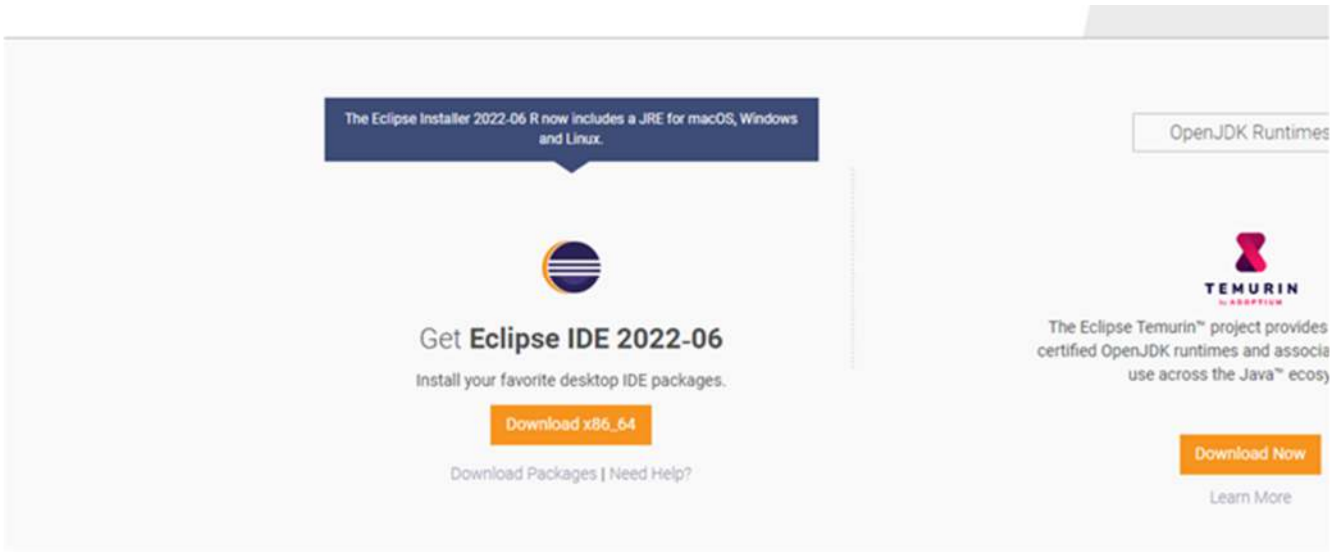
Register for EclipseCon 2022

With over 90 talks selected, EclipseCon has something for everyone! Join us in Ludwigsburg, Germany for our biggest event of the year.

9. Click on the **Download** Button

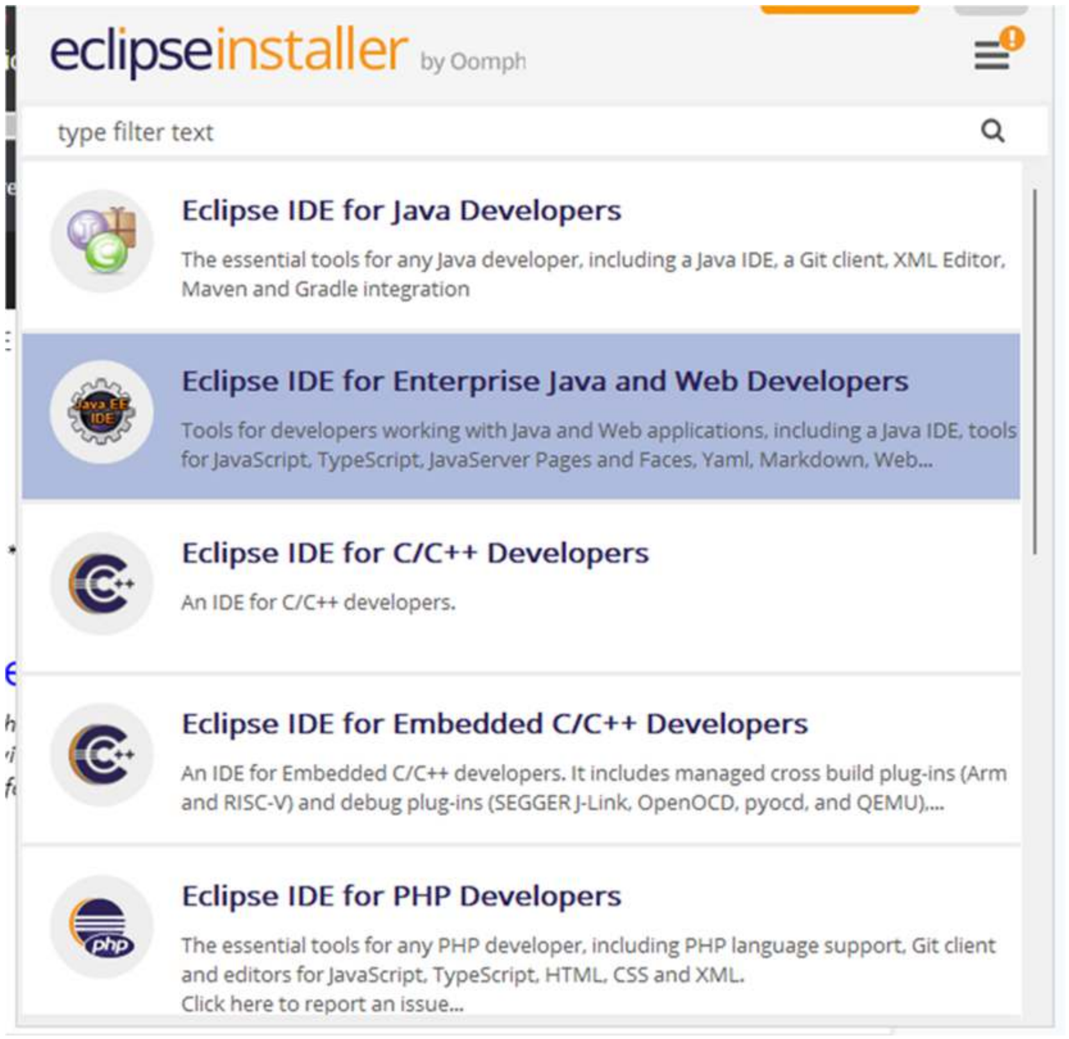


10. Click on the **Download** button


The Eclipse Installer 2022-06 R now includes a JRE for macOS, Windows and Linux.
Get Eclipse IDE 2022-06
Install your favorite desktop IDE packages.
Download x86_64
Download Packages | Need Help?
OpenJDK Runtimes
TEMURIN
The Eclipse Temurin™ project provides
certified OpenJDK runtimes and associa
use across the Java™ ecosy
Download Now
Learn More

11. Click on the downloaded file to start the installation process

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



Eclipse IDE for Java Developers

The essential tools for any Java developer, including a Java IDE, a Git client, XML Editor, Maven and Gradle integration

Eclipse IDE for Enterprise Java and Web Developers

Tools for developers working with Java and Web applications, including a Java IDE, tools for JavaScript, TypeScript, JavaServer Pages and Faces, Yaml, Markdown, Web...

Eclipse IDE for C/C++ Developers

An IDE for C/C++ developers.

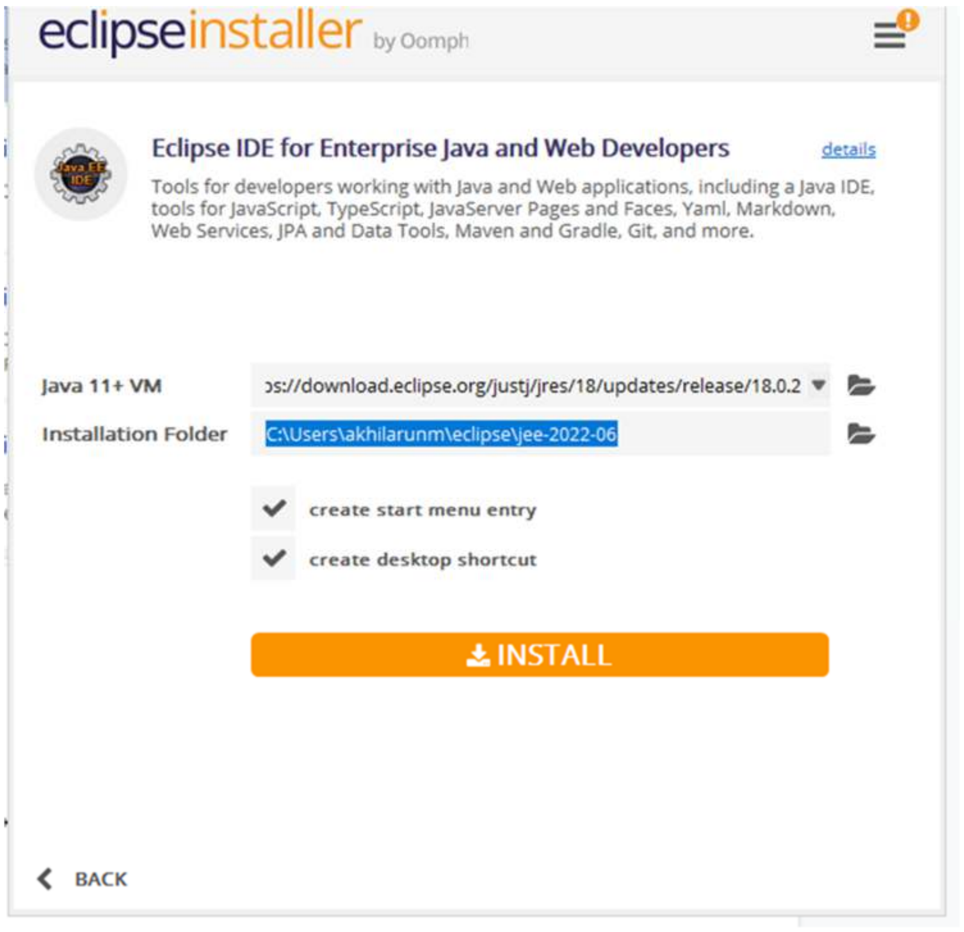
Eclipse IDE for Embedded C/C++ Developers

An IDE for Embedded C/C++ developers. It includes managed cross build plug-ins (Arm and RISC-V) and debug plug-ins (SEGGER J-Link, OpenOCD, pyocd, and QEMU),...

Eclipse IDE for PHP Developers

The essential tools for any PHP developer, including PHP language support, Git client and editors for JavaScript, TypeScript, HTML, CSS and XML.
Click here to report an issue...

13. Click **INSTALL**


eclipseinstaller by Oomph

Eclipse IDE for Enterprise Java and Web Developers

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

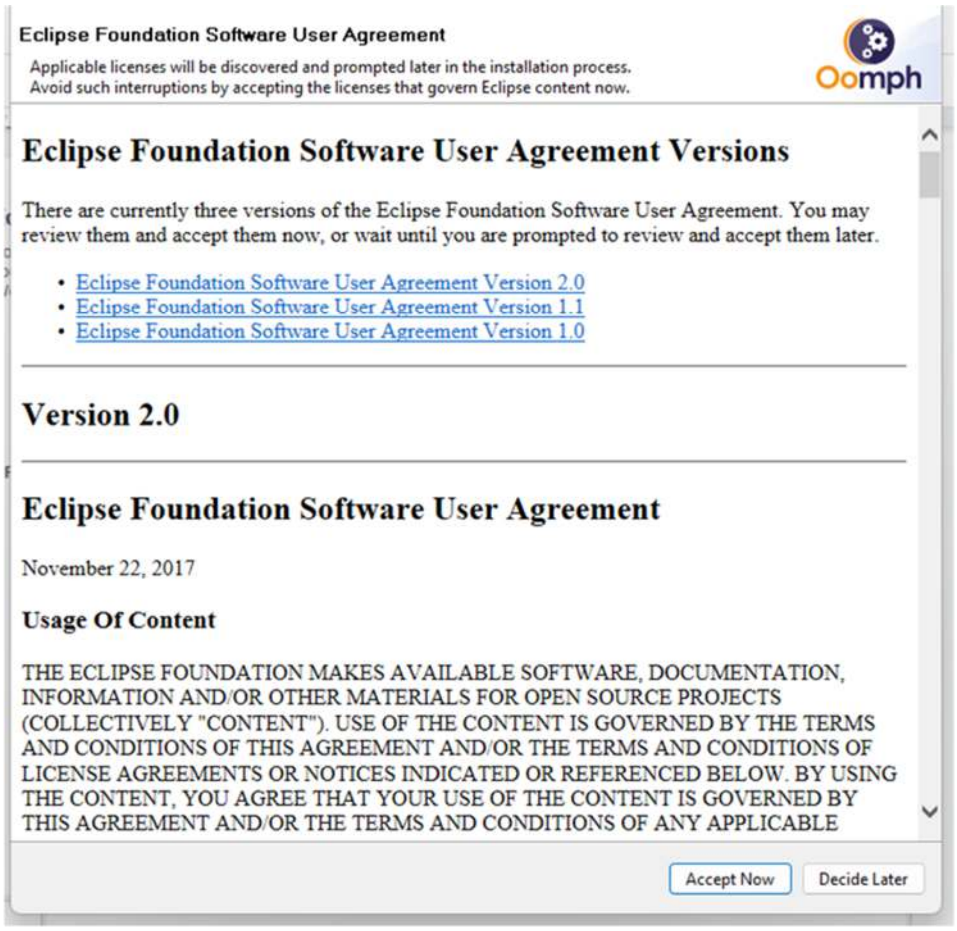
Java 11+ VM: ›s://download.eclipse.org/justj/jres/18/updates/release/18.0.2

Installation Folder: C:\Users\akhilarunm\eclipse\jee-2022-06

✓ create start menu entry

✓ create desktop shortcut

14. Click **Accept Now**



Eclipse Foundation Software User Agreement

Applicable licenses will be discovered and prompted later in the installation process.
Avoid such interruptions by accepting the licenses that govern Eclipse content now.

Eclipse Foundation Software User Agreement Versions

There are currently three versions of the Eclipse Foundation Software User Agreement. You may review them and accept them now, or wait until you are prompted to review and accept them later.

- Eclipse Foundation Software User Agreement Version 2.0
- Eclipse Foundation Software User Agreement Version 1.1
- Eclipse Foundation Software User Agreement Version 1.0

Version 2.0

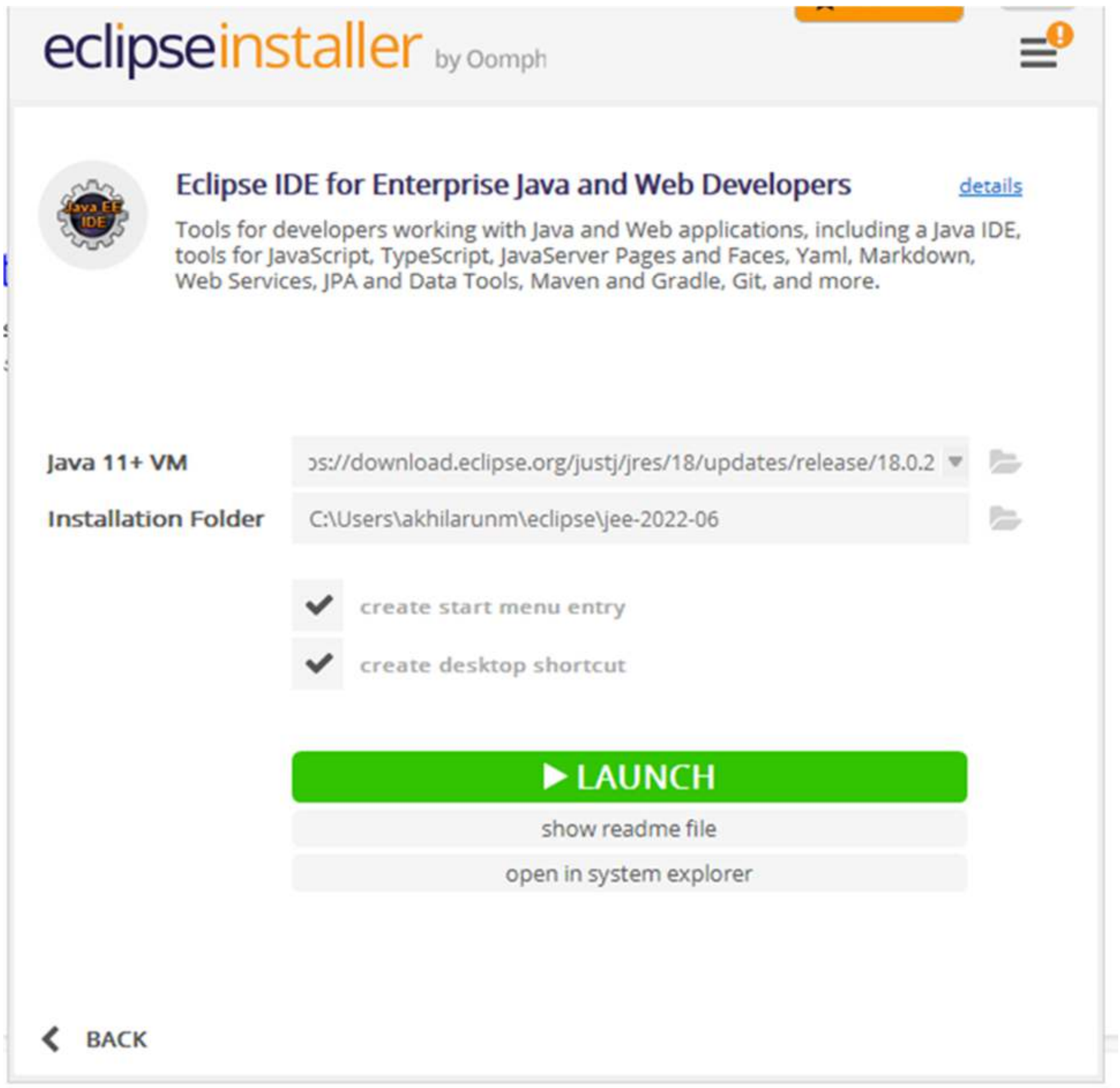
Eclipse Foundation Software User Agreement

November 22, 2017

Usage Of Content

THE ECLIPSE FOUNDATION MAKES AVAILABLE SOFTWARE, DOCUMENTATION, INFORMATION AND/OR OTHER MATERIALS FOR OPEN SOURCE PROJECTS (COLLECTIVELY "CONTENT"). USE OF THE CONTENT IS GOVERNED BY THE TERMS AND CONDITIONS OF THIS AGREEMENT AND/OR THE TERMS AND CONDITIONS OF LICENSE AGREEMENTS OR NOTICES INDICATED OR REFERENCED BELOW. BY USING THE CONTENT, YOU AGREE THAT YOUR USE OF THE CONTENT IS GOVERNED BY THIS AGREEMENT AND/OR THE TERMS AND CONDITIONS OF ANY APPLICABLE

14. Once the installation is complete, click on **LAUNCH**



Eclipse IDE for Enterprise Java and Web Developers

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

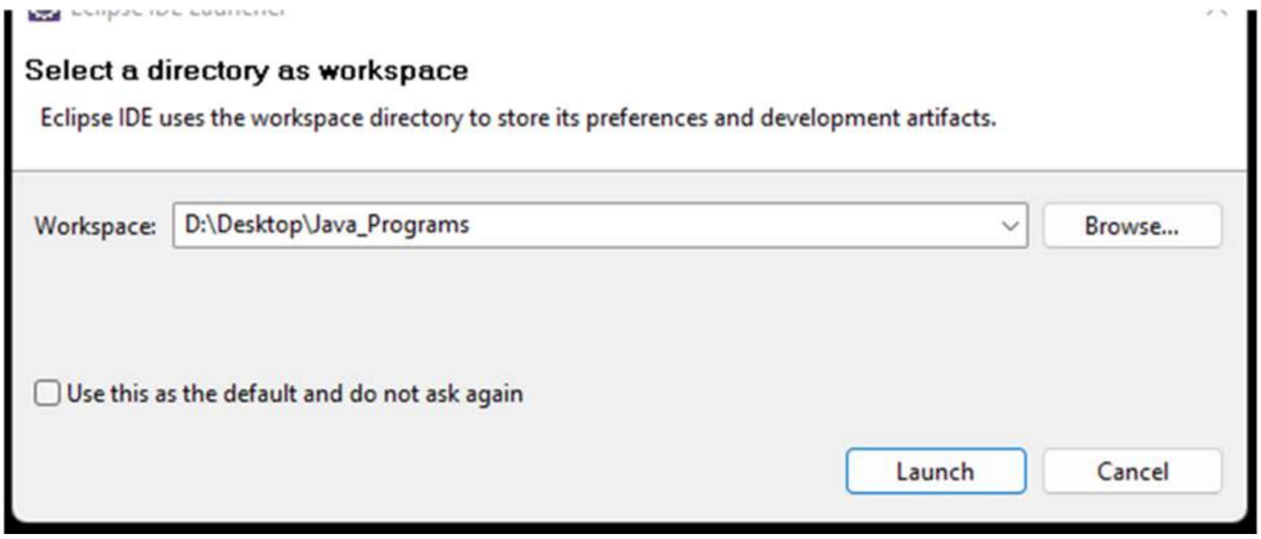
Java 11+ VM: ps://download.eclipse.org/justj/jres/18/updates/release/18.0.2

Installation Folder: C:\Users\akhilarunm\eclipse\jee-2022-06

✓ create start menu entry

✓ create desktop shortcut

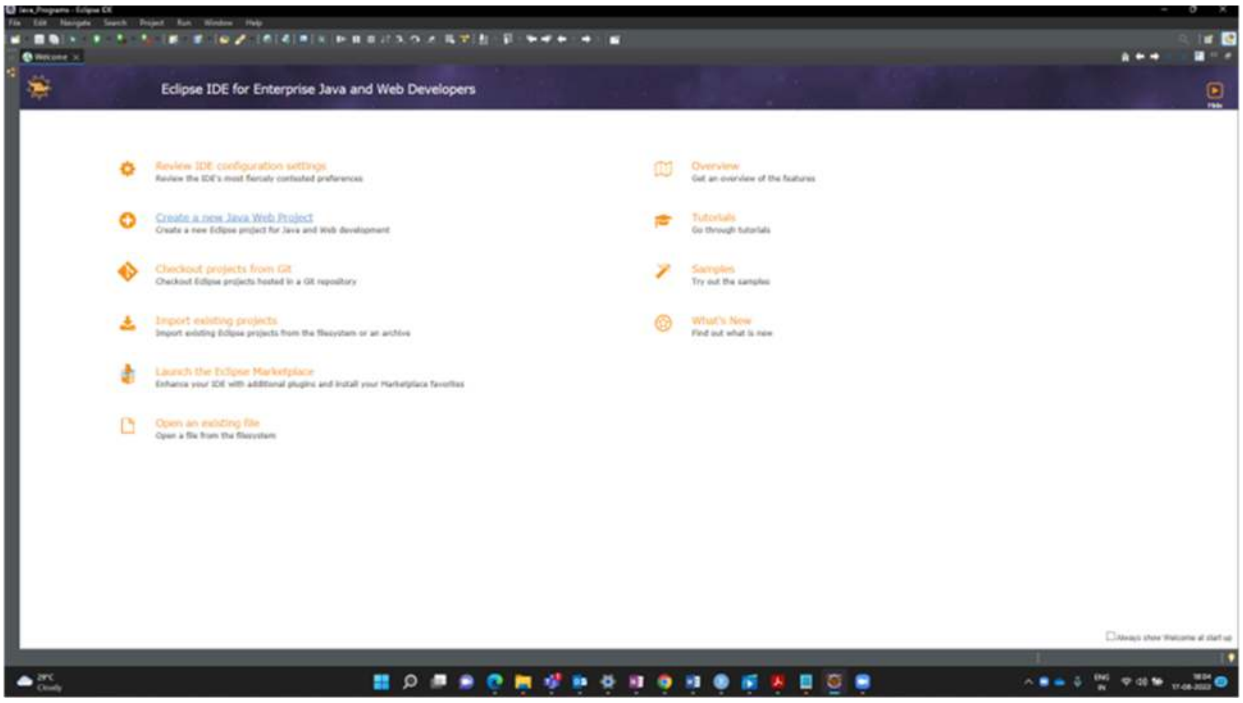
16. Set up the workspace path where you will work (preferably on the desktop), and click **LAUNCH**


Select a directory as workspace

Eclipse IDE uses the workspace directory to store its preferences and development artifacts.

Workspace: D:\Desktop\Java_Programs

17. You will view the welcome page


Eclipse IDE for Enterprise Java and Web Developers

Review IDE configuration settings
Review the IDE's most fiercely contested preferences

Create a new Java Web Project
Create a new Eclipse project for Java and Web development

Checkout projects from Git
Checkout Eclipse projects hosted in a Git repository

Import existing projects
Import existing Eclipse projects from the filesystem or an archive

Launch the Eclipse Marketplace
Enhance your IDE with additional plugins and install your Marketplace favorites

Open an existing file
Open a file from the filesystem

Overview
Get an overview of the features

Tutorials
Go through tutorials

Samples
Try out the samples

What's New
Find out what is new

InfoWarningTip
Eclipse is now successfully installed on your Windows PC. If you've closed it accidentally, you should be able to find it by opening your Start menu.

#Hello World in Eclipse

Watch the video below and carefully follow the steps for setting up your Eclipse workspace and file structure for the Java section of Academy. Our own instructor, James, does a good job walking you through it, but you can reach out to your team if you struggle with any of the steps. If you're having trouble reading the text on the screen, hover over the video and click on the settings cog, then set the video quality to 1080p.

InfoWarningTip

The first time you run a file in Eclipse, you may get a popup that asks if you want to save the file before running it. Say yes, and you'll be on your way.

InfoWarningTip

Despite the video mentioning that you may have other projects within the same larger workspace, you may be using a different and more appropriate **editor** to open and work on certain projects. For example, while working on frontend projects, you may use Visual Studio Code rather than Eclipse.

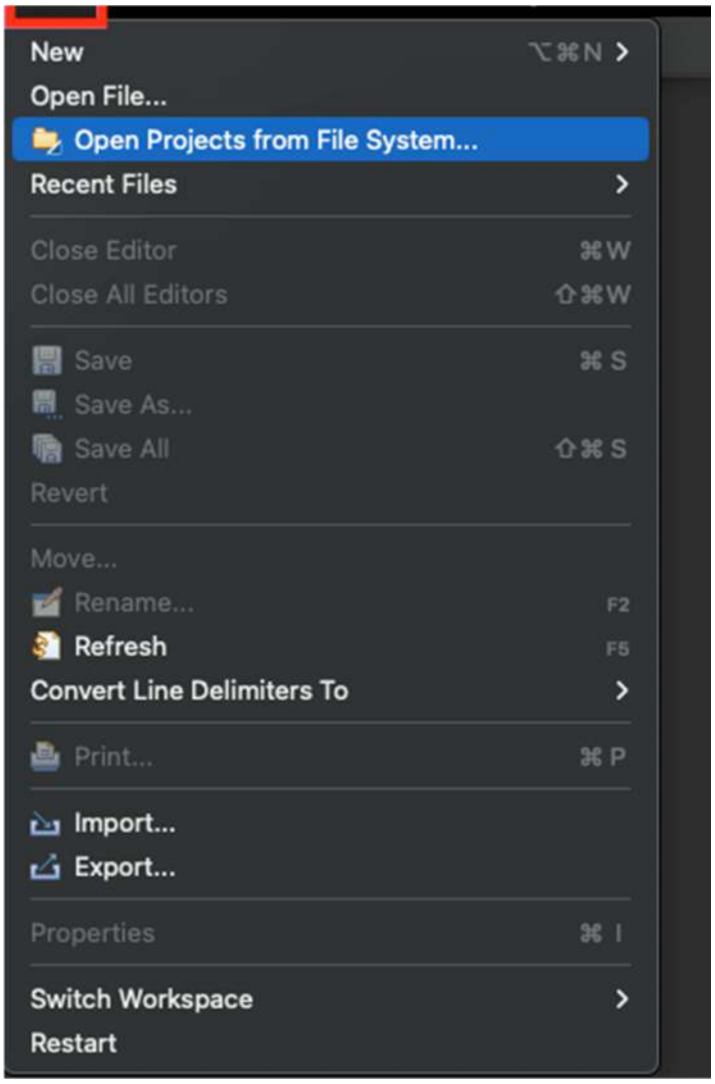
IDE Exploration

So far, you've learned a lot. Moreover, you've installed an IDE and executed your first program. Before we move on to learn more about Java, let's double check that you can create a few basic things with the **print function**.

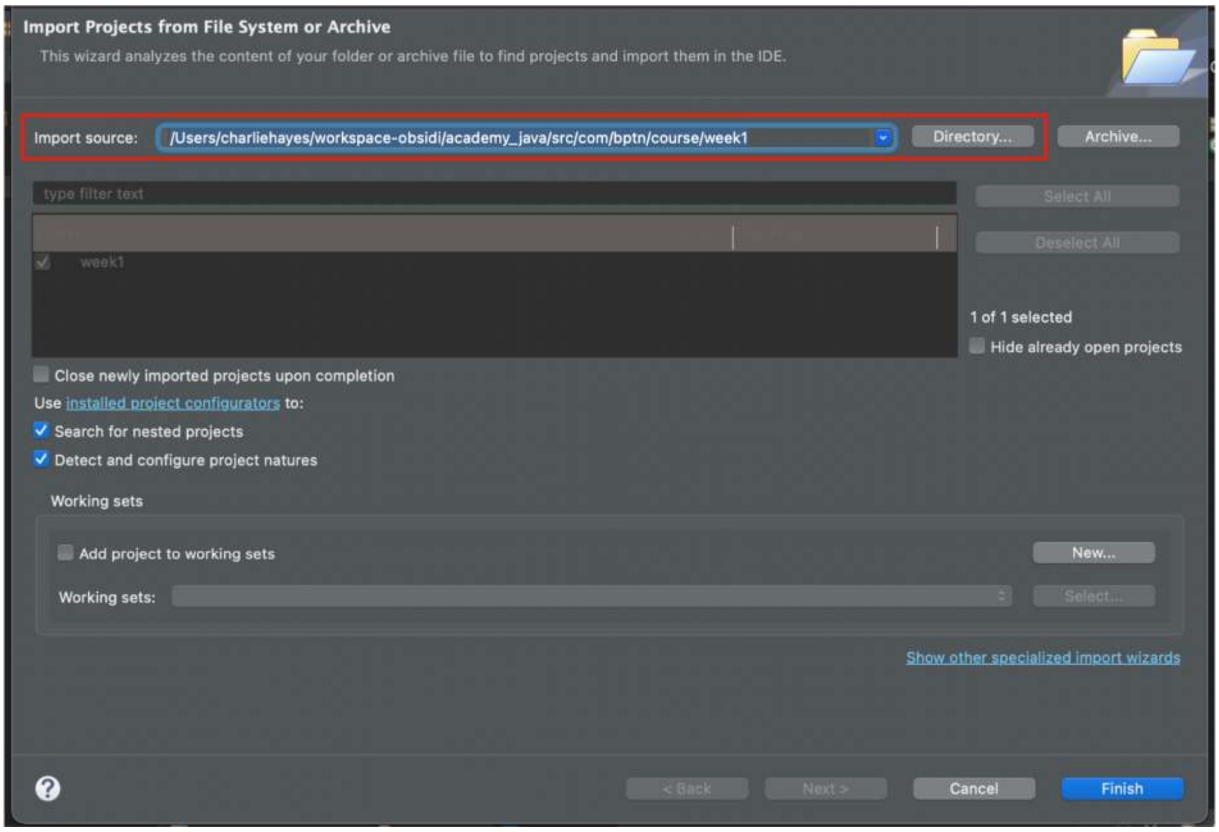
Your task is to create a new file in your IDE, inside the same week1 project that you've created, and print multiple things of your choice; this can be **any patterns, names, numbers, mathematical operations–whatever you like.**

#Steps:

1. **(If the project is not already open from our last lesson/opened automatically when you launch Eclipse then do the following steps, otherwise skip to step 4) -** Start your Eclipse IDE and go to **File -> Open Projects from File System**



2. In the Import Projects from File System or Archive dialog, click the Directory button and navigate to the directory where your Week1 project is located. Select the directory and click **OK.** Eclipse will search the selected directory for any existing Java projects. If it finds any, it will display them in the Projects list. Select the project you want to open and click **Finish**


Import Projects from File System or Archive

This wizard analyzes the content of your folder or archive file to find projects and import them in the IDE.

Import source: /Users/charliehayes/workspace-obsidi/academy_java/src/com/bptn/course/week1

☑ week1

1 of 1 selected

☐ Hide already open projects

☐ Close newly imported projects upon completion

Use installed project configurators to:

☑ Search for nested projects

☑ Detect and configure project natures

Working sets

☐ Add project to working sets

Working sets:

Eclipse may already know where your project is, or you can search through your computer by clicking on Directory...

3. Once you have imported the project, it will appear in the Package Explorer on the left side of the Eclipse window.

4. Create a new class called **VariableOperations** inside the Week1 package we created earlier today. If you don't remember how, check the video from Coding 1.1 and make yourself a cheatsheet of the important steps. You're going to be making a lot of classes throughout your time here, so it's best to get good now.

InfoWarningTip

Don't forget to check the box for `public static void main(String[] args)` when you make the class. This will save you some handwritten boilerplate, which is always a good idea. Errors in boilerplate can cause the file not to run, and be difficult to track down..

5. Declare two variables to store two numbers. Use the `int` data type to store these values, and name them whatever you like.

6. Assign values to the variables. These can be any whole numbers you like.

7. Use the basic arithmetic operators (`+`, `-`, `*`, and `/`) to perform a couple operations on the variables and store the results in new variables.

8. Print your original variables, as well as the results of the operations, to the console.

9. Create a new line after the print statement above. Store new values in the 2 variables you previously created, e.g., `var1 = 10` and `var2 = 20`.

10. Write another print statement to print your variables to show that they've changed.

11. While you're here, declare a few more variables and have them printed to the console as well. Maybe a `char` and a `String`–just play around a little, and make sure you output your new variables with a `print` statement, too.

12. Save the file and click run; when you run this program, it should print the results of your basic operations to the console. We should see:
    a. your original int variables
    b. the results of some math operations on them
    c. your reassigned variables
    d. and at least two more variables of other types

13. Upload a screenshot of your console output below:

Choose files or drag it here

Capture
.PNG

#From Here On: Use Eclipse.

The IDEs here on the Coding Rooms website are helpful, and we're going to need to see your answers here so we can test/grade them. However, from here on out, you should be doing your work inside of your Eclipse IDE. There are many reasons for this, but the biggest is simply that you won't be using Coding Rooms IDEs once you graduate from Academy. You should get used to working with real tools as soon as possible.

Luckily, Eclipse is more fully-featured and has a lot of cool, helpful tricks up its sleeve that will make the transition a pleasure. Believe us: while it may seem a bit scary at first, you'll adapt quickly, and it'll make life a whole lot easier.

Let's lay out the three top tips for using Eclipse before you continue on with your day:

1. As shown above, you should always create your new classes inside of appropriate folders/projects. If you follow the steps you saw here to create new classes for future lessons in this first week, you'll be just fine. We recommend creating new folders/packages for future weeks as you see fit–eventually, you'll be creating fully-fledged projects that require their own space, but we'll be giving special directions when we get there.
2. If we offer you an IDE that will check your code, make sure you name your new class in Eclipse the same way we've named what you see in Coding Rooms. If our class is "Fibonacci", yours should be, too. This will cut down on the errors you could get when copy/pasting the code into Coding Rooms for review.

3. If you get an error, either in your Eclipse IDE or when you copy/paste your answer into a Coding Room IDE, talk to your teammates about it. If nobody can help, Google the error and see what comes up. Learning how to find solutions when faced with errors is a huge part of becoming a developer. If your team (and the internet) can't help, reach out to your instructors and we'll see what we can do.