# Introduction to Java: Eclipse Setup Guide

**Step-by-step instructions for installing JDK and Eclipse, creating a project, and using the built-in terminal.** Note – the versions may vary – as long as your JDK works!

---

## 1. Install the Java Development Kit (JDK)

Eclipse runs on the Java platform, so you need a **Java Development Kit (JDK)** before installing the IDE. A JDK **includes** the runtime (JRE) plus tools and source code.

1. **Download the JDK** – Visit the Eclipse Temurin downloads page and select the long-term support installation, we suggest **JDK 21 (LTS)** for **Windows x64** (choose the `.msi` installer for easiest installation). If you already have JDK 25 (LTS) installed, that's also fine, but see the Java 25 note below about Eclipse support.)
2. **Run the installer** – After the download completes, double-click the `.msi` file. Accept the license agreement and use the default installation path unless you have a specific reason to change it.
3. **Verify the installation** – Open a Command Prompt (press `Win + R`, type `cmd`, press **Enter**) and run `java -version`. You should see the current version information.

*💡 Oracle vs Temurin distribution: If you need a completely free, production-ready JDK with multi-year LTS updates, Temurin is generally the go-to choice.*

*If you require Oracle's commercial support guarantees (SLA), or depend on their specific certification and branding, Oracle JDK with a support subscription may be preferable.*

---

## 2. Install the Eclipse IDE

1. **Download the installer** – Go to the official Eclipse Installer download page and choose the **Windows 64-bit** installer.
2. **Run the installer** – When the installer opens, pick **"Eclipse IDE for Java Developers"** from the list of packages (this package includes everything you need for Java programming).


*Figure 1. The Eclipse Installer lists available packages; choose "Eclipse IDE for Java Developers."*

3. **Choose an installation folder** – The default location is usually fine. Click **Install** to begin. The installer will download and install the IDE automatically.
4. **Select your Eclipse workspace** 🏠

| |
|---|
| On first launch, Eclipse will ask for a **workspace directory**. This is where Eclipse stores its own settings, metadata, and preferences.<br>**Always** choose a folder that **only** holds these Eclipse files! |

Choose a location (e.g., `C:\Users\YourName\eclipse-workspace`) and click **Launch**

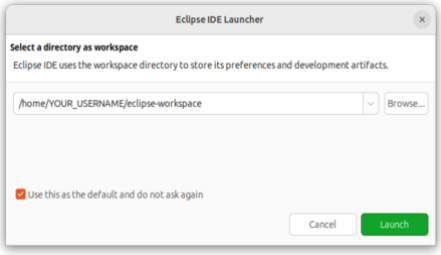



*Figure 2. Eclipse prompts for a workspace folder on first launch.*

🎯 *Tip: Check the "Use this as the default and do not ask again" box if you don't want to choose the workspace every time.*

## Java 25 in Eclipse

The current Eclipse release (2025-09 / 4.37) supports Java up to **Java 24** out of the box. If you installed **JDK 25 (LTS)** on your machine, Eclipse may only show JavaSE-24 as an option when you create a new project.

If you really want Eclipse to recognize **Java 25** explicitly (showing `JavaSE-25` in the execution environment list), you can install the official Java 25 support patch:

1. In Eclipse, go to **Help → Install New Software…**
2. In the **Work with** box, paste this URL:
`https://download.eclipse.org/jdt/updates/4.37-P-builds/`
3. Press Enter, select **Java 25 Support for Eclipse 2025-09 (4.37)** (or similar), and complete the installation.
4. Restart Eclipse. When you create a new Java project, `JavaSE-25` should now appear as an execution environment.

**Important for CSSE 220:** You do **not** need Java 25 for this course. Using **JDK 21 (LTS)** is perfectly fine and recommended unless your instructor says otherwise

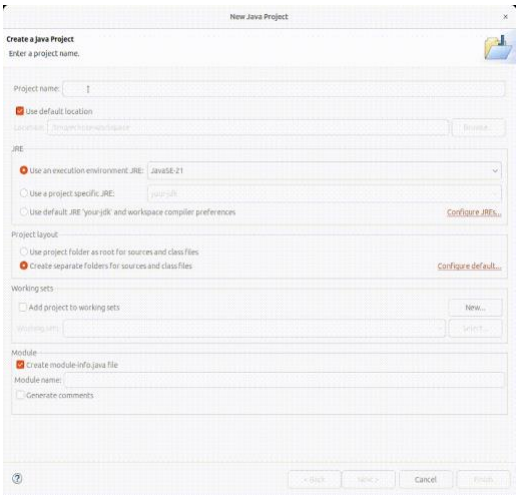
---

# 3. Create a Java Project, Package, and Class

Time to create your first project! 😊

**Before you begin:** Create a top-level **course folder** on your drive named, for example, `CSSE220`. Inside it, make a **subfolder** for this practice, `intro`.

**Now you are ready:**



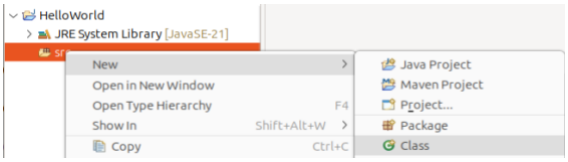

1. In Eclipse, from the top menu, go to **File** → **New** → **Java Project**.
2. The New Java Project wizard opens. Enter a project name**:** HelloWorld
3. **Uncheck Use default location**
4. Click **Browse…** and navigate to your course subfolder
5. Uncheck *Create module-info.java* (modules aren't needed for intro projects).
6. Click **Finish**.

## Create a package

In the *Project Explorer* on the left, right-click the `src` folder and select **New** → **Package**. Enter a package name, such as `introduction (lower case)`, and click **Finish**. Packages organize classes.

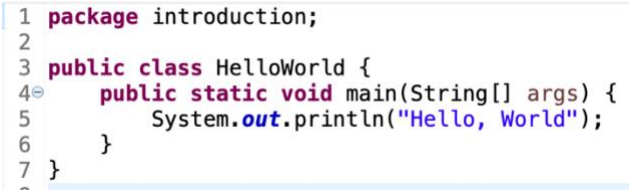
## Create a class

Right-click the package you just created and choose **New** → **Class**. Name the class (e.g., `HelloWorld`). Check the box to generate a `public static void main(String[] args)` method and click **Finish**.




When you click **Finish**, Eclipse will create your class and open it in the editor. Type:

```
System.out.println("Hello,world!");
```



```
1 package introduction;
2
3 public class HelloWorld {
4     public static void main(String[] args) {
5         System.out.println("Hello, World");
6     }
7 }
```

inside the main method, save the file, and click the **Run** button (the green play icon) to see your program output in the console.

## Lost or stuck? 😉 No worries!

1. In Eclipse's top menu, click **Help** 🕮 → Choose **Window** → **Tutorials** 🛠
2. Select the **Hello World** tutorial and work through the steps 👩‍💻👨‍💻

---

## 3. Eclipse Shortcuts Every Undergrad Should Know

- **Indent:** `Ctrl + I`
- **De-Indent:** `Shift + Tab`

- **Toggle line comment:** `Ctrl + /`
- **Block comment:** `Ctrl + Shift + /` (wraps selection with /* */)

- **Move line up/down:** `Alt + ↑` / `Alt + ↓`
- **Duplicate line:** `Ctrl + Alt + ↓`

- **sysout snippet:** type `sysout` → press `Ctrl + Space`
- **General auto-suggest:** `Ctrl + Space`

- **Maximize/restore current panel:** `Ctrl + M`

---

## 5. You're Ready to Code!

Congratulations 🥕 – you've set up your Java development environment! You installed the JDK and Eclipse, created your first project, package and class. Have fun and keep learning! 🚀