VS Code and Java Installation Tutorial

Contents

V 1.0

2022-2023

Prerequisite

VS Code provides essential language features such as code completion, refactoring, linting, formatting, and code snippets along with convenient debugging and unit test support.

Windows OS

- Windows 7 or later (Windows 7, Windows 8, Windows 8.1, Windows 10)
- Administrator Control (You have the permission & credentials (Username/Password) to install software)

Mac OS

- Administrator Control (You have the permission & credentials (Username/Password) to install software)

New Install Visual Studio Code for Java

If there is your first time to use VS Code, to help you set up quickly, we recommend you use the Coding Pack for Java, which is the bundle of VS Code, the Java Development Kit (JDK), and a collection of suggested extensions by Microsoft. The Coding Pack can also be used to fix an existing development environment.

Choose one of the Coding Pack based on your operation system.

- Install the Coding Pack for Java – Windows OS (https://aka.ms/vscode-java-installer-win)
- Install the Coding Pack for Java - macOS (https://aka.ms/vscode-java-installer-mac)

Note: The Coding Pack for Java is only available for Windows and macOS. For other operating systems, you will need to manually install a JDK, VS Code, and Java extensions.

Install support for Java in existing VS Code

If you have already installed VS Code and want to add Java support to it, we recommend using the Extension Pack for Java, a collection of extensions suggested by Microsoft. Open the link (Extension Pack for Java) and click the **install** button.
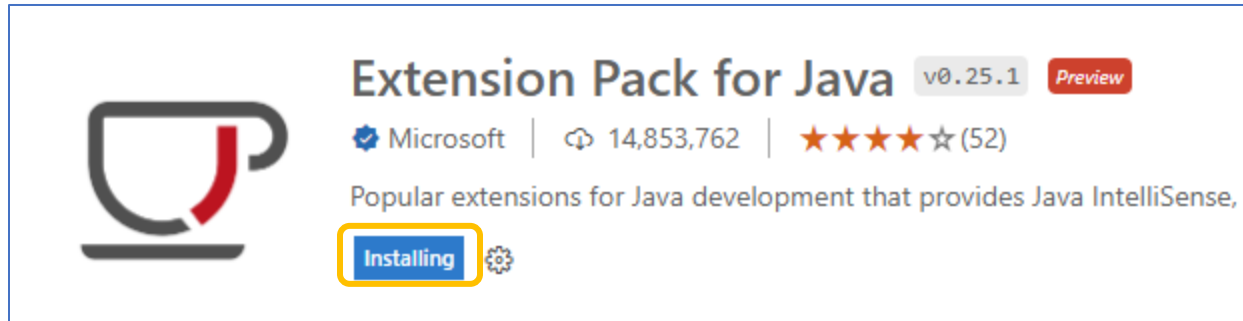
Figure 1. Install VS Code Java Extension

Install a Java Development Kit (JDK)

To use Java within Visual Studio Code, you need to install a Java Development Kit (**JDK**) on your computer.

A Java Development Kit (JDK) is a software development environment used for developing Java applications. In order to run Java within Visual Studio Code, you need to install a JDK. The Extension Pack for Java supports Java version 1.5 or above.

For developers new to Java or new to VS Code, Open **menu/View/Command Palette/** (or press Ctrl+Shift+P) and type "**install new**" to select the **Java: Install New JDK** command to install JDK.
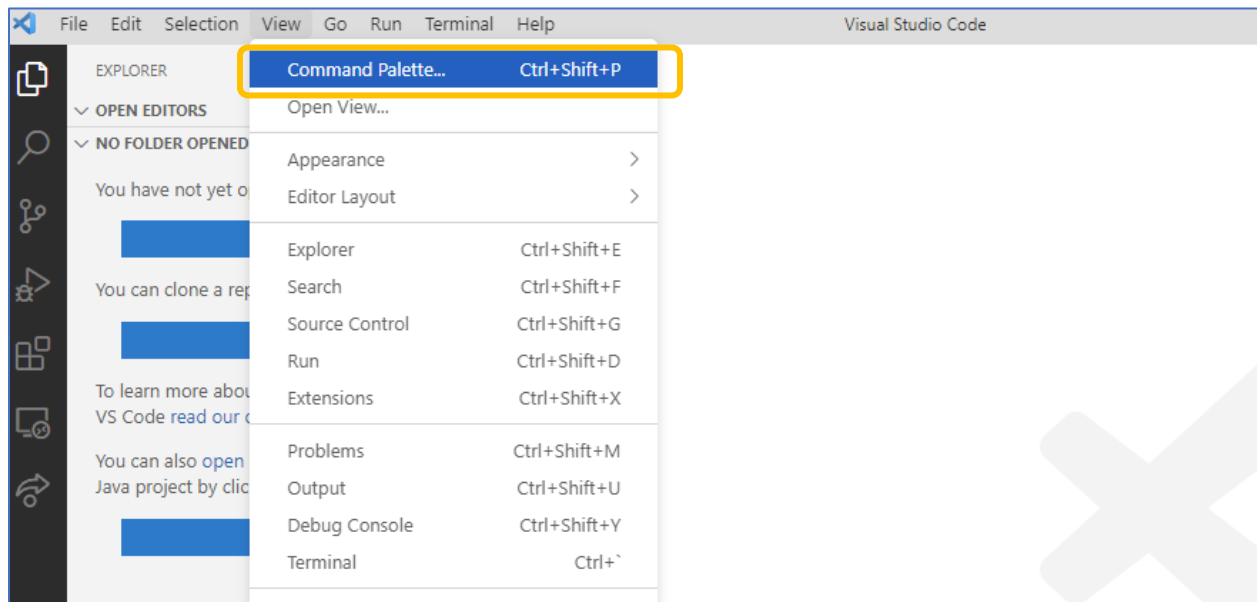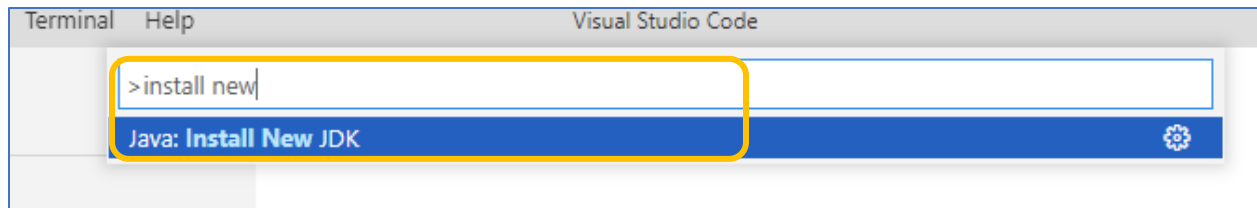

Figure 2. Control Palette in VS Code

Figure 3. Type to find JDK version in VS Code

In the prompt window, you may see a tutorial to install JDK via VS Code.

Figure 4. Select JDK version and Download the setup file.

Select Java version that meets your need and click **Download** button. A JDK installation file will be downloaded. For example, ***OpenJDK17U-jdk_XXXX_XXXXXX.msi***. The filename might be different, for example, ***OpenJDK17U-jdk_x64_windows_hotspot_17.0.4.1_1.msi*** (because I choose JDK version 17 and 64-bit windows version).

OpenJDK17U-jdk
_x64_windows_h
otspot_17.0.4.1_
1.msi
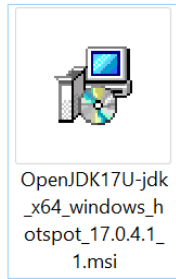
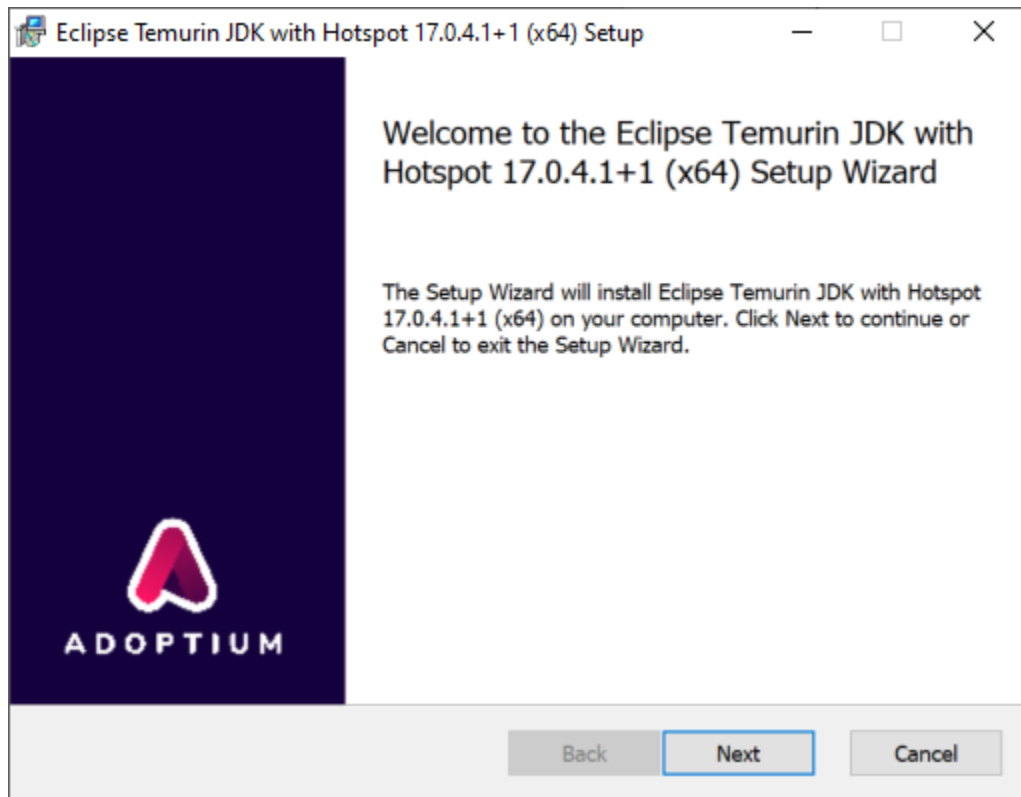Run the downloaded installation file and follow its instructions.

Figure 5. JDK Installation

Note: To be set-up correctly, usually you must add the location of the installed JDK to the system path variable for your OS. For example, set install status for Add to PATH, Set JAVA_HOME variable as the steps in the screenshot.
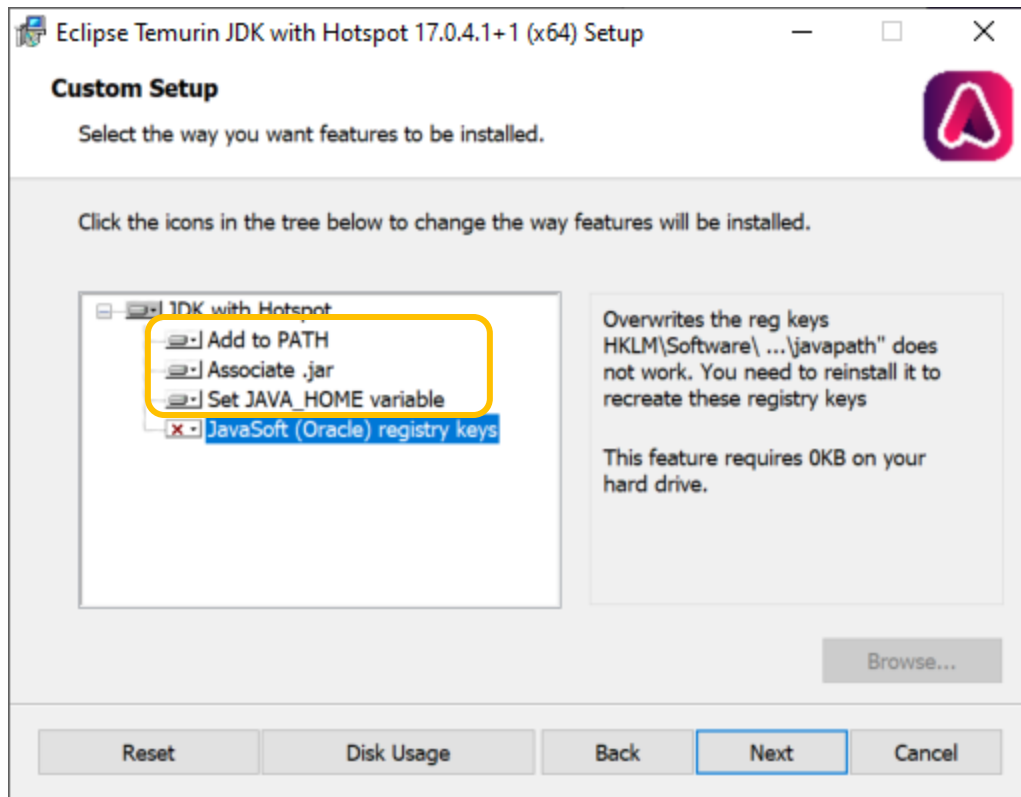
Figure 6. Set PATH Variable
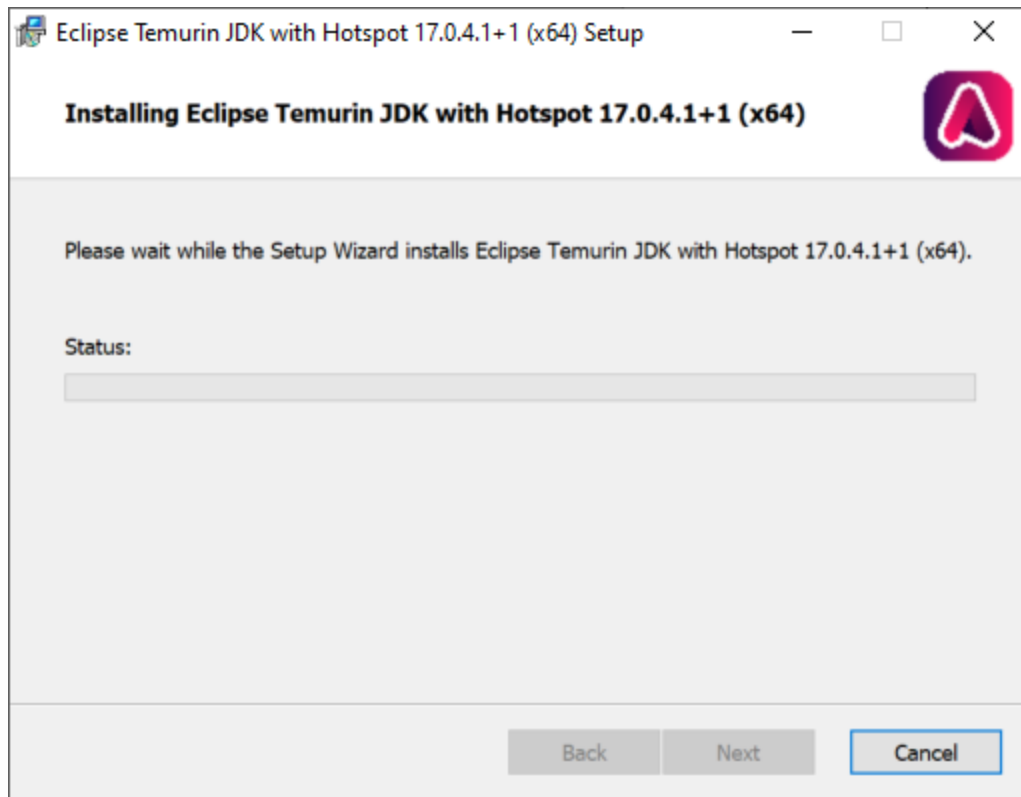
Enter your administrator credentials if prompts.

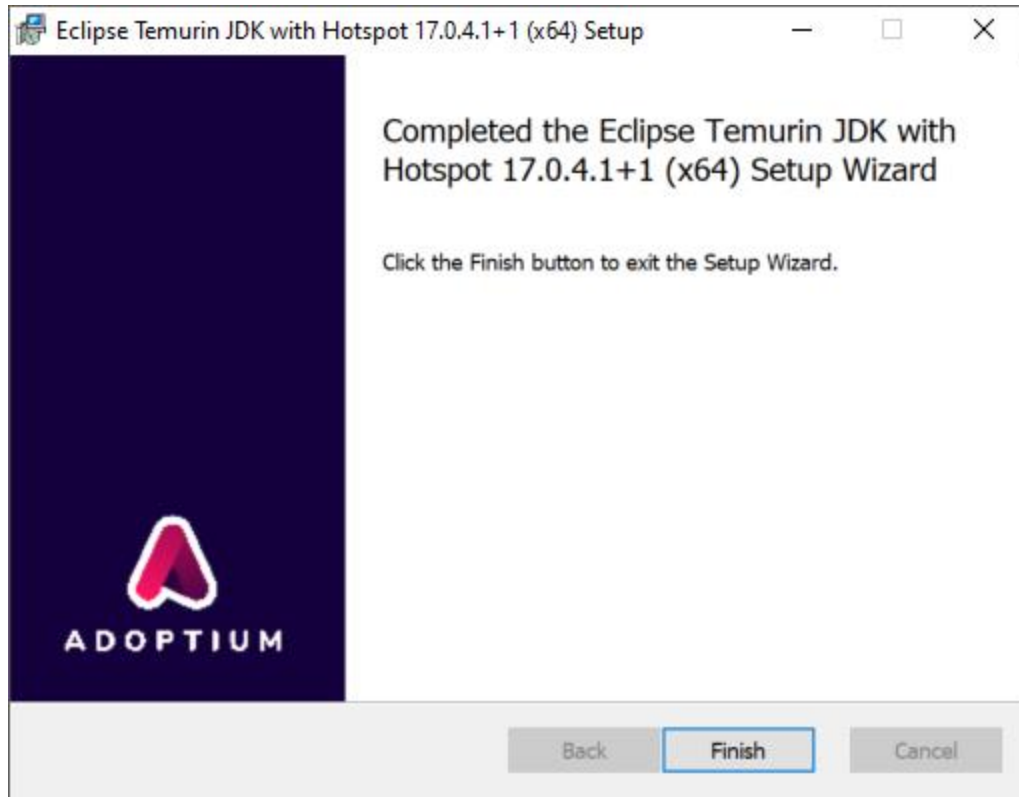*Figure 7. JDK Installation*

Figure 8. JDK Installation

After the installation is done, click Finish and re-open your VS code. You can start to write your Java Code!

(Optional step) Alternatively, there are many other sources you can consider installing the JDK from one of these sources:

- Amazon Corretto
- Azul Zulu
- Eclipse Adoptium's Temurin
- Microsoft Build of OpenJDK
- Oracle Java SE
- Red Hat build of OpenJDK
- SapMachine

You may choose to use them based on the requirement of your class or project.

Getting Started Hello World in Java

## Working with Java source files

You can use VS Code to read, write, run, and debug Java source file(s) without creating a project. VS Code for Java supports two modes, **lightweight** and **standard**.

- **Lightweight** mode is ideal for scenarios that only deal with source file(s).
- If you want to work with a full-scale project, **standard** mode will be required.

You can easily switch from lightweight mode to standard mode, when needed. To learn more, see Lightweight Mode.

Hello World

Let's create a new file in VS Code. Save and name it as **HelloWorld.java**
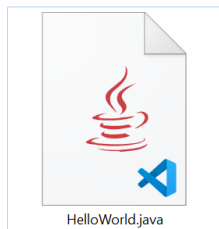
HelloWorld.java

Inside the Java file, you can enter the following Java code.

Note: the filename should be the same as your class name, in our example, both are **HelloWorld**.

```java
// HelloWorld.java
public class HelloWorld {
    public static void main(String []args) {
        System.out.println("Hello World"); // prints
Hello World
    }
 }
```

Figure 9. HelloWorld.java Source Code

In the open Java file, click **menu/Run/Start Debugging** (or press **F5)** to run your program it.
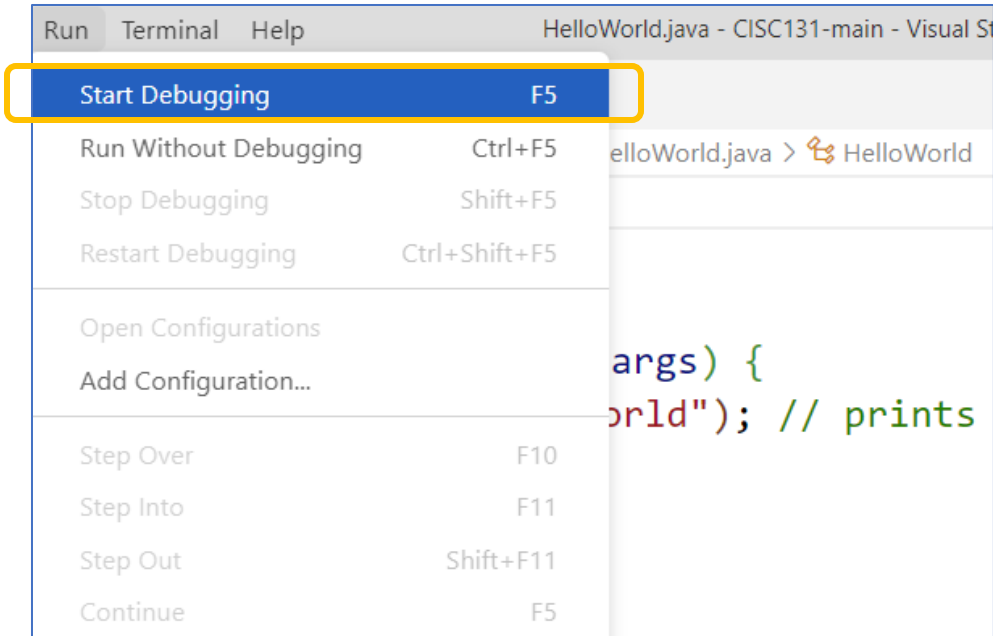
Figure 10. Run Java Code in Lightweight mode

You will see "**Hello World**" output to the Terminal panel.

Figure 11. Java Output on Terminal Panel

Compile and run Java code in Command line.

Open your command terminal (Windows prompt) and locate your java source code.

For example, my **HelloWorld.java** are located in **C:\Users\Professor Wang\Downloads**

Navigate your command prompt to the same location and Enter

`javac HelloWorld.java`

Figure 12. Locate and compile java file in Command Prompt

After finished the code above, if there is no syntax error, your JVM will compile the .java file into .class file which is the compiled machine code. In our example, it is **HelloWorld.class**

Figure 13. Using dir command to list the file in current folder

Last, because in the current folder, we have both source code (**HelloWorld.java**) and complied class file (**HelloWorld.class**). We can run the program using the following command:

`java HelloWorld`

Figure 14. Java Program Output in Command Prompt

You can also see the output Hello World message in your terminal.

Now you are ready to go! Have fun ! 😊

Reference

https://code.visualstudio.com/docs/languages/java