Install JDK 17 for Windows

This is the Windows installation video - skip this if you are on a Mac or running Linux.

Right - so we've established that you want to install Java 17, or to be specific, JDK version 17.

This video will show you how to install on Windows 11. Note that if you are running on Windows 10, the installation will be pretty much identical. Just follow the steps you see in this video if you are running Windows 10.

The only potential gotcha is if you are running a 32 bit version of Windows 10 (these days most computers are running the 64 bit edition). In these rare cases you will find you need to download an older version of the JDK.

Install JDK 17 for Windows

If you have a relatively new PC from the past 4-5 years (or even a bit longer) then all should work.

Windows 11 only comes in a 64 bit version.

So if you are running Windows 10 or Windows 11, just follow along and assume your computer is running a 64 bit edition of Windows and it all should work fine.

Install JDK 17 for Windows

Great, so we now have the JDK installed, in the next video, it's time to download and install IntelliJ, which is the program we will be using in the course, that works with the JDK we just installed, to let you write and execute Java programs.

Confirming installation and intro to JShell

Now that we have Java installed on our machines, I want to introduce you to the software tool we'll be using to begin to explore the Java programming language. JShell.

Firstly, let's open a command prompt on Windows, or a Terminal in Mac or Linux, to check our Java version, to make sure we are ready.

Confirming installation and intro to JShell

On Windows, you can press and hold the Windows key located next to the space bar, and then press the R key. This will take you to the 'Run' window, where you can type in cmd and hit the enter key.

On a Mac, you can press the Command key and spacebar at the same time, to get to Spotlight, and then type in Terminal. This should take you to the built in terminal on your Mac.

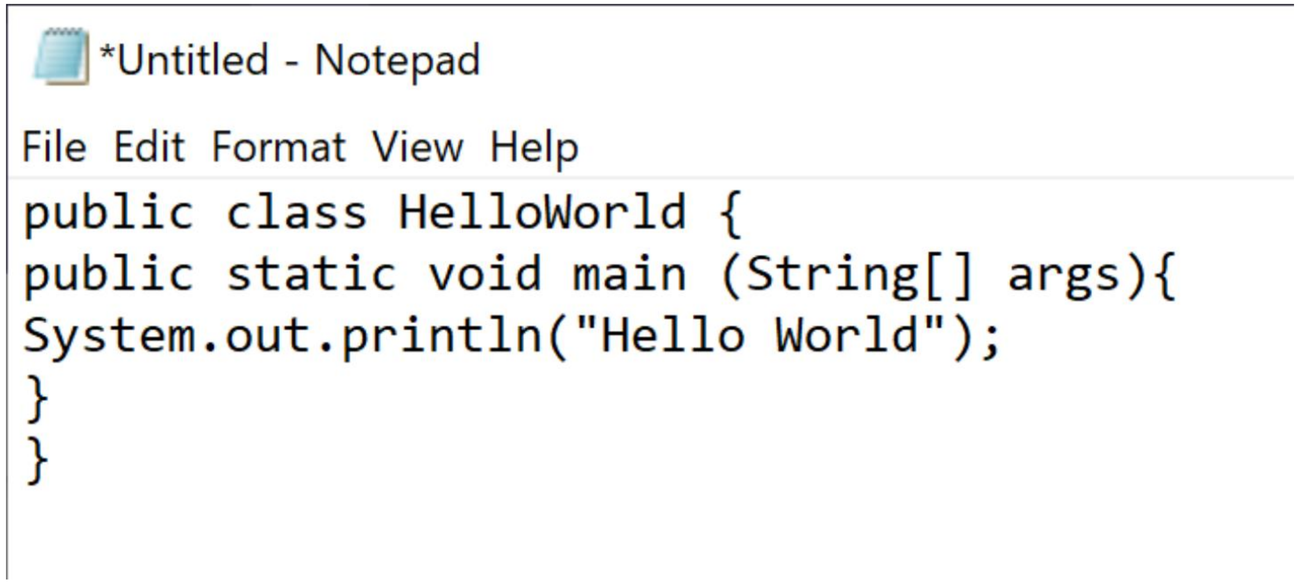
For Ubuntu Linux, click on 'show applications', in the bottom left of your screen. Then in the search bar, you can type in Terminal and hit enter.

Java Version

```
C:\Users\timbuchalka>java -version
java version "17.0.4.1" 2022-08-18 LTS
Java(TM) SE Runtime Environment (build 17.0.4.1+1-LTS-2)
Java HotSpot(TM) 64-Bit Server VM (build 17.0.4.1+1-LTS-2, mixed mode, sharing)
```

Code in Notepad

You could use a simple text editor to write Java code.



IntelliJ IDE

Most likely, you'll be using an integrated development environment, or IDE, to develop your Java code.

HelloWorld.java

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

JShell

JShell became a standard component of the Java Developers Kit in Java 9.

It is what is known as a Read-Eval-Print-Loop interactive program (or REPL for short) which means it does pretty much just that:

- it **reads** the command or code segment we type in.
- it **evaluates** and executes the code, and often allows short cuts to be used.
- it **prints** out the results of the evaluation or execution, without making the developer write code to output the results.
- Lastly, it loops right back for more input (more code segments or commands).

JShell runs in a terminal (or on the command line for Windows) and is useful for quickly trying out new ideas.

JShell

JShell does not replace the need for an IDE.

It's just a handy tool to quickly get started with Java. We will be transitioning to an IDE later in the course.

Where's the Java code?

Why did we introduce you to JShell, and not write any Java code? I had a good reason for doing this.

Our first Java code is going to be the very traditional program for a beginner, and we'll do that in the next video.

I wanted to get all the JShell commands and usage out of the way, so in the next video, we can concentrate on just Java code. Remember later in the course we will swing over to an IDE instead of JShell.