**How to Install JDK 8 (on Windows, Mac OS & Ubuntu) and Get Started with Java Programming**

Java Development Kit (JDK), officially named "Java Platform Standard Edition (Java SE)", which is freely available from Sun Microsystems (now part of Oracle), is needed for writing Java programs. The mother site for JDK (Java SE) is <http://www.oracle.com/technetwork/java/index.html>.

**"JDK" or "JRE"?**

JRE (Java Runtime) is needed for *running* Java programs. JDK (Java Development Kit), which includes JRE plus the development tools (such as compiler and debugger), is need for *writing* as well as *running* Java programs. In other words, JRE is a subset of JDK. Since you are supposed to write Java Programs, you should install JDK, which includes JRE.

**JDK Versions**

The various JDK versions are:

1. JDK Alpha and Beta (1995): Sun announced Java in September 23, 1995.
2. JDK 1.0 (January 23, 1996): Originally called *Oak* (named after the oak tree outside James Gosling's office). Renamed to Java 1 in JDK 1.0.2.
3. JDK 1.1 (February 19, 1997): Introduced AWT event model, inner class, JavaBean, JDBC, and RMI.
4. J2SE 1.2 (JDK 1.2) (December 8, 1998): Re-branded as "Java 2" and renamed JDK to J2SE (Java 2 Standard Edition). Also released J2EE (Java 2 Enterprise Edition) and J2ME (Java 2 Micro Edition). Included JFC (Java Foundation Classes - Swing, Accessibility API, Java 2D, Pluggable Look and Feel and Drag and Drop). Introduced Collection Framework and JIT compiler.
5. J2SE 1.3 (JDK 1.3) (May 8, 2000): Introduced Hotspot JVM.
6. J2SE 1.4 (JDK 1.4) (February 6, 2002): Introduced assert, non-blocking IO (nio), logging API, image IO, Java webstart, regular expression support.
7. J2SE 5.0 (JDK 1.5) (September 30, 2004): Officially called 5.0 instead of 1.5. Introduced generics, autoboxing/unboxing, annotation, enum, varargs, for-each loop, static import.
8. Java SE 6 (JDK 1.6) (December 11, 2006): Renamed J2SE to Java SE (Java Standard Edition).
9. Java SE 7 (JDK 1.7) (July 28, 2011): First version after Oracle purchased Sun (called Oracle JDK).
10. Java SE 8 (JDK 1.8) (March 18, 2014): included support for Lambda expressions, default methods, and JavaScript runtime.
11. Java SE 9: Expected in July 2017.

**1.  How To Install JDK on Windows**

**Step 0: Un-Install Older Version(s) of JDK/JRE**

I recommend that you install only the *latest* JDK. Although you can install multiple versions of JDK/JRE concurrently, it is messy.

If you have previously installed older version(s) of JDK/JRE, un-install ALL of them. Goto "Control Panel" ⇒ "Programs" ⇒ "Programs and Features" ⇒ Un-install ALL programs begin with "Java", such as "Java SE Development Kit ...", "Java SE Runtime ...", "Java X Update ...", and etc.

**Step 1: Download JDK**

1. Goto Java SE download site @ <http://www.oracle.com/technetwork/java/javase/downloads/index.html>.
2. Under "Java Platform, Standard Edition" ⇒ "Java SE 8u{xx}", where {xx} is the latest update number ⇒ Click the "JDK Download" button.
3. Look for the latest "Java SE Development Kit 8u{xx}" ⇒ Check "Accept License Agreement".
4. Choose the JDK for your operating system, e.g., "Windows x64" (for 64-bit Windows OS) or "Windows x86" (for 32-bit Windows OS). You can check whether your Windows OS is 32-bit or 64-bit via "Control Panel" ⇒ (Optional) System and Security ⇒ System ⇒ Under "System Type".

**Step 2: Install JDK and JRE**

Run the downloaded installer (e.g., "jdk-8u{xx}-windows-x64.exe"), which installs both the JDK and JRE. By default, the JDK will be installed in directory "C:\Program Files\Java\jdk1.8.0\_xx", where xx denotes the upgrade number; and JRE in "C:\Program Files\Java\jre1.8.0\_*xx*".

Accept the defaults and follow the screen instructions to install JDK and JRE.

Check the JDK installed directory by inspecting these folders using File Explorer. Take note of your JDK installed directory, in particular, the upgrade number, which you will need in the next step.

I shall refer to the JDK installed directory as <JAVA\_HOME>, hereafter, in this article.

**Step 3: Include JDK's "bin" Directory in the PATH**

Windows Shell searches the current directory and the directories listed in the PATH *environment variable* (*system variable*) for executable programs. JDK's programs (such as Java compiler javac.exe and Java runtime java.exe) reside in directory "<JAVA\_HOME>\bin" (where <JAVA\_HOME> denotes the JDK installed directory). You need to include "<JAVA\_HOME>\bin" in the PATH to run the JDK programs.

To edit the PATH environment variable in Windows 7/8/10:

1. Launch "Control Panel" ⇒ (Optional) System and Security ⇒ System ⇒ Click "Advanced system settings" on the left pane.
2. Switch to "Advanced" tab ⇒ Push "Environment Variables" button.
3. Under "System Variables" (the bottom pane), scroll down to select "Path" ⇒ Click "Edit...".
4. **For Windows 10 (newer releases)**:  
   You shall see a TABLE listing all the existing PATH entries (if not, goto next step). Click "New" ⇒ Enter the JDK's binary directory "c:\Program Files\Java\jdk1.8.0\_xx\bin" (Replace xx with your installation number!!!) ⇒ Select "Move Up" to move this entry all the way to the TOP.  
     
   **Prior to Windows 10**:  
   **(CAUTION: Read this paragraph 3 times before doing this step! Don't push "Apply" or "OK" until you are 101% sure. There is no UNDO!!!)**  
   (To be SAFE, copy the content of the "Variable value" to Notepad before changing it!!!)  
   In "Variable value" field, INSERT "c:\Program Files\Java\jdk1.8.0\_xx\bin" (Replace xx with your installation number!!!) IN FRONT of all the existing directories, followed by a semi-colon (;) which separates the JDK's binary directory from the rest of the existing directories. DO NOT DELETE any existing entries; otherwise, some existing applications may not run.
5. Variable name : **PATH**

Variable value : **c:\Program Files\Java\jdk1.8.0\_xx\bin;[*exiting entries...*]**

My Notes: Starting from JDK 1.8, the installation created a directory "c:\ProgramData\Oracle\Java\javapath" and added to the PATH. It contains only JRE executables (java.exe, javaw.exe, javaws.exe) but NOT the JDK executables (e.g., javac.exe).

**Step 4: Verify the JDK Installation**

Launch a CMD shell via one of the following means:

1. Click "Search" button ⇒ Enter "cmd" ⇒ Choose "Command Prompt", or
2. right-click "Start" button ⇒ run... ⇒ enter "cmd", or
3. (Prior to Windows 10) click "Start" button ⇒ All Programs ⇒ Accessories (or Windows System) ⇒ Command Prompt, or
4. (Windows 10) click "Start" button ⇒ Windows System ⇒ Command Prompt

Issue the following commands to verify your JDK installation:

1. Issue "path" command to list the contents of the PATH environment variable. Check to make sure that your <JAVA\_HOME>\bin is listed in the PATH.  
   Don't type prompt>, which denotes the command prompt!!! Key in the command (highlighted) only.
2. // Display the PATH entries
3. prompt> **path**

PATH=**c:\Program Files\Java\jdk1.8.0\_xx\bin**;[*other entries...*]

1. Issue the following commands to verify that JDK/JRE are properly installed and display their version:
2. // Display the JRE version
3. prompt> **java -version**
4. java version "1.8.0\_xx"
5. Java(TM) SE Runtime Environment (build 1.8.0\_xx-b13)
6. Java HotSpot(TM) 64-Bit Server VM (build 25.5-b02, mixed mode)
8. // Display the JDK version
9. prompt> **javac -version**

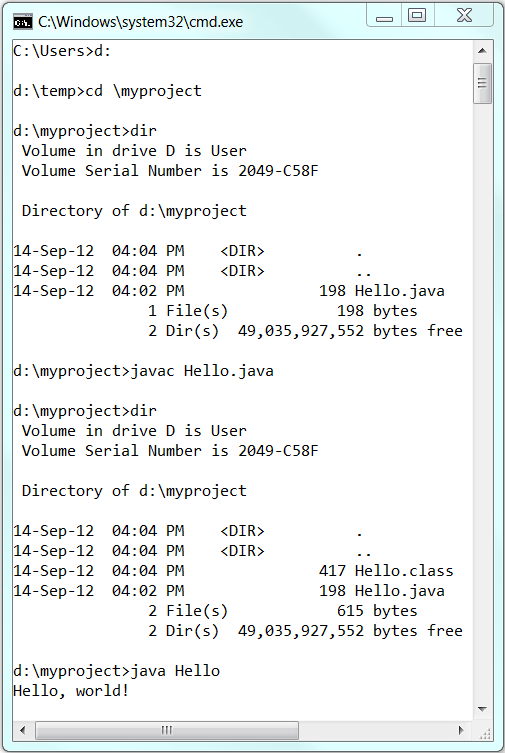
javac 1.8.0\_xx

**Step 5: Write a Hello-World Java Program**

1. Create a directory to keep your works, e.g., "d:\myProject", or "c:\myProject", or any directory of your choice. Do NOT save your works in "Desktop" or "Documents" as they are hard to locate. The directory name shall not contain *blank* or special characters. Use meaningful but short name as it is easier to type.
2. Launch a programming text editor (such as [TextPad](https://www3.ntu.edu.sg/home/ehchua/programming/howto/CMD_Survival.html#textpad), or [NotePad++](https://www3.ntu.edu.sg/home/ehchua/programming/howto/CMD_Survival.html#NotePadPP), or Sublime Text, or Atom). Begin with a *new file* and enter the following source code. Save the file as "Hello.java", under your work directory (e.g., d:\myProject).
3. /\*
4. \* First Java program to say Hello
5. \*/
6. public class Hello { // Save as "Hello.java" under "d:\myProject"
7. public static void main(String[] args) {
8. System.out.println("Hello, world!");
9. }

}

**Step 6: Compile and Run the Hello-World Java Program**



To compile the source code "Hello.java":

1. Start a CMD Shell (Click the "Start" button ⇒ "run..." ⇒ Enter "cmd").
2. Set the *Current Drive* to the drive where you saved your source file "Hello.java". For example, suppose that your source file is saved in drive "d", enter "**d:**" as follow:
3. prompt> **d:**

D:\xxx>

Don't enter prompt>, which denotes the command prompt.

1. Set the *Current Working Directory* to the directory that you saved your source file via the **cd** (*Change Directory*) command. For example, suppose that your source file is saved in directory "d:\myProject".
2. D:\xxx> **cd \myProject**

D:\myProject>

1. Issue a **dir** (*List Directory*) command to confirm that your source file is present in the *current directory*.
2. D:\myProject> **dir**
3. ......
4. xx-xxx-xx 06:25 PM 277 **Hello.java**

......

1. Invoke the JDK compiler "**javac**" to compile the source code "Hello.java".

D:\myProject> **javac Hello.java**

The compilation is successful if the command prompt returns. Otherwise, error messages would be shown. Correct the errors in your source file and re-compile. Check "[Common JDK Installation Errors](https://www3.ntu.edu.sg/home/ehchua/programming/howto/ErrorMessages.html#JDKErrors)", if you encounter problem compiling your program.

1. The output of the compilation is a Java class called "Hello.class". Issue a **dir** (*List Directory*) command again to check for the output.
2. D:\myProject> **dir**
3. ......
4. xx-xxx-xx 01:53 PM 416 **Hello.class**
5. xx-xxx-xx 06:25 PM 277 Hello.java

......

To run the program, invoke the Java Runtime "**java**":

D:\myProject> **java Hello**

Hello, world!