Java Platform, Standard Edition Installation Guide
Contents   Previous   Next

---

15 JDK Installation for Microsoft Windows

This page describes how to install and uninstall JDK 8 for Windows.

JDK 8

The page has these topics:

- System Requirements

- Installation Instructions Notation

- Installation Instructions

- Uninstalling the JDK

- Installed Directory Tree

- Installation Troubleshooting

See "JDK 8 and JRE 8 Installation Start Here" for general information about installing JDK 8 and JRE 8.

System Requirements

See `http://www.oracle.com/technetwork/java/javase/certconfig-2095354.html` for information about supported platforms, operating systems, and browsers.

See "Windows System Requirements for JDK and JRE" for minimum processor, disk space, and memory requirements.

Note:

- The JDK and JRE have a version string that enables you to determine the version number. See `http://www.oracle.com/technetwork/java/javase/jdk8-naming-2157130.html` for information about Java SE 8 version numbers.

- The JDK has the option of installing the public JRE. For more information about JRE installation, see "JRE Installation for Microsoft Windows".

- If you have any difficulties, see "Installation Troubleshooting" or submit a bug report at `http://bugreport.java.com/bugreport/`.

Installation Instructions Notation

For any text in this document that contains the following notation, you must substitute the appropriate update version number:

`version`

For example, if you were downloading the JDK installer for 32-bit systems for update 1.8.0_01, the

file name: `jdk-8version-windows-i586.exe` becomes `jdk-8u1-windows-i586.exe`.

Similarly, if you were <u>downloading the JDK installer for 64-bit systems</u> for update 1.8.0_01, the file name `jdk-8u`<mark>`<version>`</mark>`-windows-x64.exe` becomes `jdk-8u1-windows-x64.exe`.

Installation Instructions

In these instructions, you <u>run the self-installing executable file to unpack and install the JDK</u>. As part of the JDK, this installation includes an option to include the public Java Runtime Environment. (The JDK also contains a private JRE for use only by its tools; see <u>"Private Versus Public JRE"</u> for more information.)

<u>Install the JDK</u> by doing the following:

- <u>"Downloading the Installer"</u>

- <u>"Running the JDK Installer"</u>

- <u>"Installing the JDK Silently"</u>

- <u>"Updating the PATH Environment Variable"</u>

- <u>"Starting to Use the JDK"</u>

Downloading the Installer

If you <u>save the self-installing</u> <mark>executable file</mark> <u>to disk</u> without running it from the download page at the web site, <mark>note</mark> the file size specified on the download page. After the download has completed, <u>verify that you have downloaded the complete file</u>.

Running the JDK Installer

You <u>must have</u> <mark>administrative permissions</mark> <u>in order to install the JDK</u> on Microsoft Windows.

The file `jdk-8version-windows-i586.exe` is the JDK installer for 32-bit systems. The file `jdk-8version-windows-x64.exe` is the JDK installer for 64-bit systems. <u>If you downloaded either file</u> instead of running it directly from the web site, <u>double-click the installer's icon. Then, follow the instructions the installer provides</u>. When finished with the installation, you can delete the downloaded file to recover disk space.

Installers for JDK 7u6 and later install the JavaFX SDK and integrate it into the JDK installation directory. Installers for JDK 7u2 to 7u5 install the JDK first, then start the JavaFX SDK installer, which installs JavaFX SDK in the default directory `C:\Program Files\Oracle\JavaFX 2.0 SDK` or `C:\Program Files (x86)\Oracle\JavaFX 2.0 SDK` on 64-bit operating systems. If you want to install the JavaFX SDK (version 2.0.2) with JDK 7u1 or earlier, see `http://docs.oracle.com/javafx/2/installation/jfxpub-installation.htm` for more information.

Java Start Menu

<u>Starting with</u> <mark>JDK 7u40 release,</mark> <mark>Java menu items</mark> <u>are added to</u> the <mark>Windows Start Menu</mark> to <u>provide easy access to</u> <mark>Java resources.</mark>

During JDK install, a <mark>Java Development Kit</mark> <u>folder is created in the Windows Start Menu</u>, which contains the following items:

- **Reference Documentation:** Opens the Online API documentation web page.

- **Java Mission Control:** Opens the Java Mission Control profiling and diagnostics tools suite.

During JDK install and uninstall processes, the appropriate start menu items are updated to be associated with the latest JDK version on the system

---

Note:

Java Mission Control is a commercial feature available to users with a Java SE Advanced license.

Installing the JDK Silently

Instead of double-clicking or opening the installer, you can perform a silent, non-interactive, JDK installation by using the command-line arguments. The following table lists example installation scenarios and the commands required to perform them. The notation `jdk` stands for the downloaded installer file base name, such as `jdk-8u05-windows-i586`.

Installation Scenario	Command
Install the public JRE in silent mode	`jdk.exe /s`
Install development tools and source code in silent mode but not the public JRE	`jdk.exe /s ADDLOCAL="ToolsFeature,SourceFeature"`
Install development tools, source code, and the public JRE in silent mode	`jdk.exe /s ADDLOCAL="ToolsFeature,SourceFeature,PublicjreFeature"`
Install the public JRE in the specified directory `C:\test\` in silent mode	`jdk.exe /s /INSTALLDIRPUBJRE=C:\test\`

Updating the PATH Environment Variable

If you do not set the `PATH` variable, you need to specify the full path to the executable file every time you run it, such as:

```
C:\> "C:\Program Files\Java\jdk1.8.0\bin\javac" MyClass.java
```

It is useful to set the `PATH` variable permanently so it will persist after rebooting.

To set the `PATH` variable permanently, add the full path of the `jdk1.8.0\bin` directory to the `PATH` variable. Typically, this full path looks something like `C:\Program Files\Java\jdk1.8.0\bin`. Set the `PATH` variable as follows on Microsoft Windows:

1. Click **Start**, then **Control Panel**, then **System**.

2. Click **Advanced**, then **Environment Variables**.

3. Add the location of the bin folder of the JDK installation to the `PATH` variable in **System Variables**. The following is a typical value for the `PATH` variable:

```
C:\WINDOWS\system32;C:\WINDOWS;C:\Program Files\Java\jdk1.8.0\bin
```

Note:

- The `PATH` environment variable is a series of directories separated by semicolons (`;`) and is not case-sensitive. Microsoft Windows looks for programs in the `PATH` directories in order, from left to right.

- You should only have one `bin` directory for a JDK in the path at a time. Those following the first instance are ignored.

- If you are not sure where to add the JDK path, append it.

- The new path takes effect in each new command window you open after setting the `PATH` variable.

Starting to Use the JDK

Use the **Java** item in the Windows **Start** menu to get access to essential Java information and functions, including help, API documentation, the Java Control Panel, checking for updates, and Java Mission Control.

If you are new to developing and running programs in the Java programming language, see `http://docs.oracle.com/javase/tutorial` for some guidance. Note especially the tutorial trails under the heading *Trails Covering the Basics*.

You can also download the JDK documentation from `http://www.oracle.com/technetwork/java/javase/downloads/index-jsp-138363.html` page.

Uninstalling the JDK

To uninstall the JDK, use the "Add/Remove Programs" utility in the Microsoft Windows Control Panel.

Installed Directory Tree

See `http://docs.oracle.com/javase/8/docs/technotes/tools/windows/jdkfiles.html` for a description of the directory structure of the JDK. (Note that the file structure of the JRE is identical to that of the JDK's `jre` directory.)

Installation Troubleshooting

Below are some tips for working around problems that are sometimes seen during or following an installation:

- ["Corrupt Cabinet File"](#)

- ["System Error During Decompression"](#)

- ["Program Cannot Be Run in DOS Mode"](#)

- ["Private Versus Public JRE"](#)

- ["Source Files in Notepad"](#)

- ["Characters That Are Not Part of the System Code Page"](#)

For more troubleshooting information, see `http://docs.oracle.com/javase/8/docs/technotes/guides/tsgdesktop/index.html`.

Corrupt Cabinet File

If you see the error message "corrupt cabinet file," then the file you have downloaded is corrupted. Check the file size against the expected file size listed in these instructions. If sizes do not match, try downloading the bundle again. (A cabinet file contains compressed application, data, resource, and DLL files.)

System Error During Decompression

If you see the error message "system error during decompression," then you might not have enough space on the disk that contains your `TEMP` directory.

Program Cannot Be Run in DOS Mode

If you see the error message "This program cannot be run in DOS mode," then do the following:

1. Open the MS-DOS shell or Command Prompt window.

2. Right-click the title bar.

3. Select **Properties**.

4. Choose the **Program** tab.

5. Click the **Advanced** button.

6. Ensure that the item "Prevent MS-DOS-based programs from detecting Windows" is not selected.

7. Select **OK**.

8. Select **OK** again.

9. Exit the MS-DOS shell.

10. Restart your computer.

Private Versus Public JRE

Installing the JDK also installs a private JRE and optionally a public copy. The private JRE is required to run the tools included with the JDK. It has no registry settings and is contained entirely in a `jre` directory (typically at `C:\Program Files\jdk1.8.0\jre`) whose location is known only to the JDK. On the other hand, the public JRE can be used by other Java applications, is contained outside the JDK (typically at `C:\Program Files\Java\jre1.8.0`), is registered with the Windows registry (at `HKEY_LOCAL_MACHINE\SOFTWARE\JavaSoft`), can be removed using Add/Remove Programs, might be registered with browsers, and might have the `java.exe` file copied to the Windows system directory (which would make it the default system Java platform).

Source Files in Notepad

In Microsoft Windows, when you create a new file in Microsoft Notepad and then save it for the first time, Notepad usually adds the `.txt` extension to the file name. Therefore, a file you name `Test.java` is saved as `Test.java.txt`. It is important to note that you cannot see the `.txt` extension unless you turn on the viewing of file extensions (in Microsoft Windows Explorer, unselect "Hide file extensions for known file types" under Folder Options). To prevent the `.txt` extension, enclose the file name in quotation marks, such as `"Test.java"`, when typing it into the Save As dialog box.

On the other hand, Microsoft WordPad does not add the `.txt` extension if you specify another extension. However, you must save the file as "Text Document".

Characters That Are Not Part of the System Code Page

It is possible to name directories using characters that are not part of the system locale's code page. If such a directory is part of the installation path, then generic error 1722 occurs, and installation is not completed. Error 1722 is a Windows Installer error code. It indicates that the installation process has failed. The exact reason for this error is not known at this time.

To prevent this problem, ensure that the user and system locales are identical, and that the installation path contains only characters that are part of the system locale's code page. User and system locales can be set in the Regional Options or Regional Settings control panel.

The associated bug number is [4895647](#).

[Contents](#)   [Previous](#)   [Next](#)