

How to Install JSindo for Windows

Kiyoshi Yagi
kiyoshi.yagi@riken.jp

Theoretical Molecular Science Laboratory
RIKEN Pioneering Research Cluster

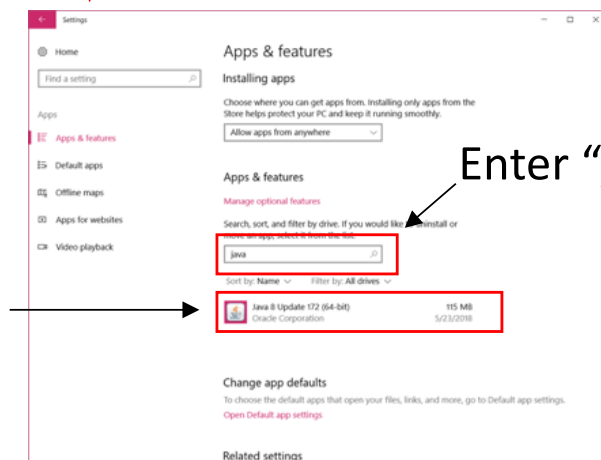
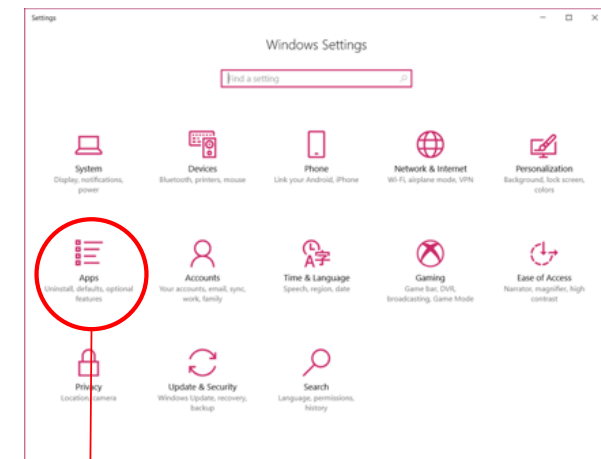
2018/06/01

1. Install Java

STEP1: Let's check if your PC has Java installed or not, and the version of Java if you have.

Open the “Windows Settings”, click “Apps”, and search for “java” in Apps & features.

- If you don't find anything, it means you don't have java in your PC. Goto **STEP2** to install.
- If your Java is Version 8, then you can skip the installation and go to Chap. 2.

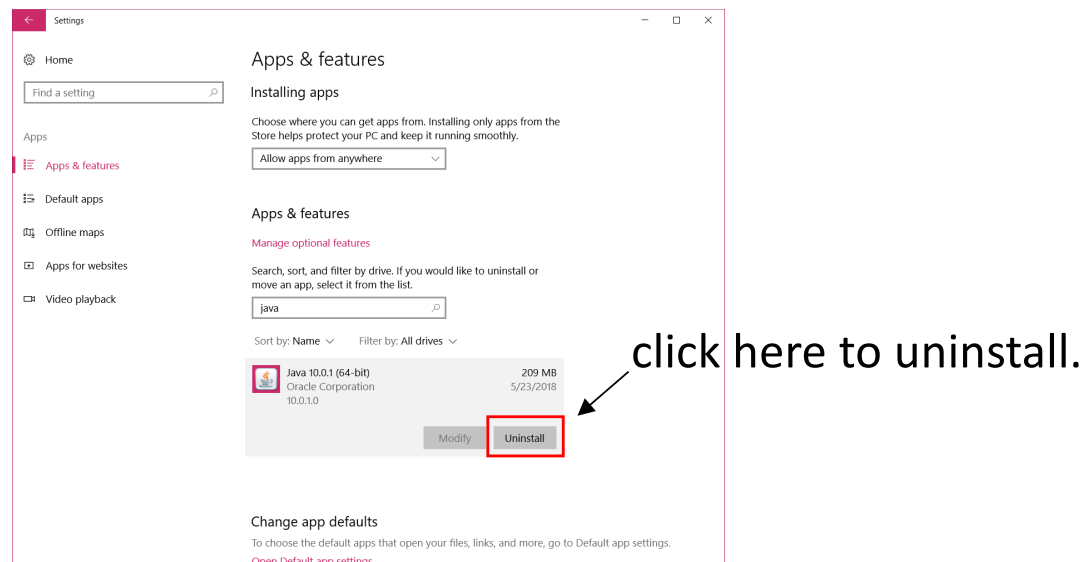


Version 8 Update 172!

Enter “java”.

If your Java is a newer one (version 9 and later), it is unfortunately **NOT** compatible with Java3D library, which JSindo use for visualization. In this case, uninstall Java and re-install version 8.

To uninstall java, click the program and then click a “uninstall” button.



Unfortunately, your Java is Version 10.0.1...

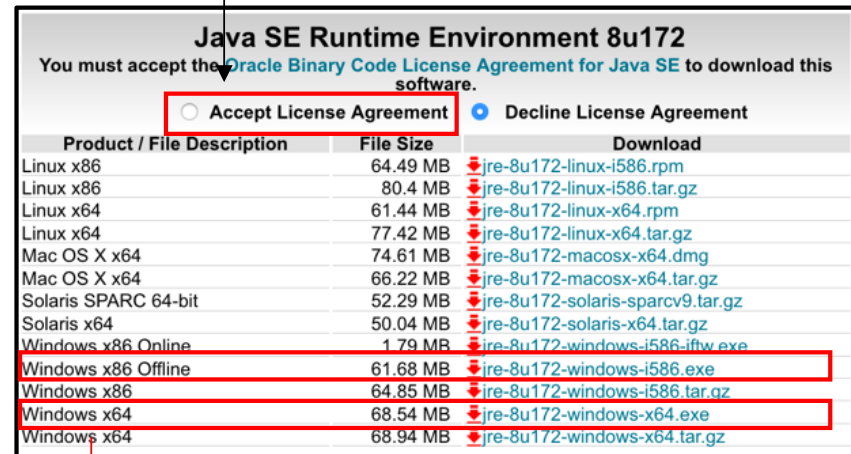
STEP2: Install Java8.

Search “Java SE download” in Google and goto the following website.

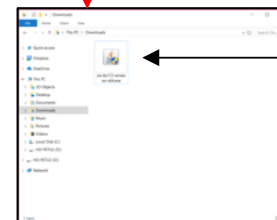
<http://www.oracle.com/technetwork/java/javase/downloads/index.html>



accept



download “i586” (32-bit) or
“x64” (64-bit).



Double click, follow the instruction, and you’re done.

You may do STEP1 to double check you’ve got the right version installed.

2. Download Java3D

JSindo uses Java3D for visualization. A stable version, 1.6.0, is available from JogAmp. Goto <http://jogamp.org>



Main Page

Welcome to the [JogAmp](#) wiki. It documents JOGL, JOCL and JOAL, the cross-platform bindings to the OpenGL, OpenCL, and OpenAL APIs.

click here

Getting Started

- [Downloading and installing](#)
- [Versioning and Releases](#)
- [Setting up a JogAmp project in your favorite IDE](#)
- [Source Code Repositories](#)
- [Tracking Reports](#)
- [Build and Test Server](#)

Community

- [Contribute to JogAmp](#)
- [Build JogAmp](#)
- [Maintainer and Contacts](#)
- [Report a bug](#)
 - [Bugzilla](#)
- [Ask a question in the forum](#)
- [JogAmp IRC](#)

Downloading and installing JOGL

Before you can build a project that uses JOGL [in your IDE](#) or [on the command line](#), you'll need to download and install the JOGL JAR files and native JARs or native library files (.dll/.so/.jnilib files).

You have a choice of JOGL versions to download. The [latest stable version](#) is the safest, but lags behind in features. The [latest automatic build](#) contains all checked-in code, but may be failing some tests.

Contents [hide]

- 1 [Downloading the latest stable version](#)
 - 1.1 [Using the 7z jogamp-all-platforms archive](#)
- 2 [Downloading the latest aggregated autobuild](#)
- 3 [Downloading the latest automatic build](#)
 - 3.1 [Native JARs vs. native library files](#)
 - 3.2 [Unzipping the files](#)
- 4 [Maven](#)
- 5 [More information](#)

Downloading the latest stable version

Go to [this page](#) and download the all-in-one 7z archive file:

[jogamp-all-platforms.7z](#)

click here and download
jogamp-all-platforms.7z

Go back to the Main page and scroll down

Main Page

Welcome to the [JogAmp](#) wiki. It documents JOGL, JOCL and JOAL, the cross-platform bindings to the OpenGL, OpenCL, and OpenAL APIs.

⋮ ↓ Scroll down

Related Projects

Java3D

- Overview
- **Downloading and installing**
- Tutorial
- API Documentation
- FAQ

click here

Ji Gong

- Overview
- Motivation
- FAQ

Page **Discussion**

Downloading and installing Java3D

Downloading the latest stable version

Go to [this page](#) and download the 7z archive file:

[jogamp-java3d.7z](#)

Do the same for JogAmp as it is indicated [here](#).

click here and download
jogamp-java3d.7z

Unarchive the two files you've just downloaded. 7z files can be unarchived using, for example, "7Z Opener"



7Z Opener
Tiny Opener

You will find jar files in jogamp-all-platforms/jar and in jogamp-java3d. The following jar files are needed for JSindo:

```
jogamp-all-platforms/jar/  
  gluegen-rt.jar  
  gluegen.jar  
  gluegen-rt-natives-windows-xxx.jar  
  jogl-all.jar  
  jogl-all-natives-windows-xxx.jar
```

```
jogamp-java3d/  
  j3dcore.jar  
  j3dutils.jar  
  vecmath.jar
```

where **xxx** = i586 (32-bit) or amd64 (64-bit).

3. Download JAMA

JAMA is a linear algebra library for JAVA. We use it for matrix multiplications, diagonalization, and so on. It can be downloaded from,

<https://math.nist.gov/javanumerics/jama/>

JAMA : A Java Matrix Package

[\[Background \]](#) [\[The Package \]](#) [\[Request for Comments \]](#) [\[Authors \]](#) [\[Related Links & Libraries \]](#)

Background

JAMA is a basic linear algebra package for Java. It provides user-level classes for constructing and manipulating real, dense matrices. It is meant to provide sufficient functionality for routine problems, packaged in a way that is natural and understandable to non-experts. It is intended to serve as *the* standard matrix class for Java, and will be proposed as such to the [Java Grande Forum](#) and then to [Sun](#). A straightforward public-domain reference implementation has been developed by the [MathWorks](#) and [NIST](#) as a strawman for such a class. We are releasing this version in order to obtain public comment. There is no guarantee that future versions of JAMA will be compatible with this one.

⋮ ↓ Scroll down

The Package

Version 1.0.3 (November 9, 2012)

- [Documentation](#)
- [Example](#)
- Source [[Jama-1.0.3.zip](#)] [[Jama-1.0.3.tar.gz](#)]
- Jar file [[Jama-1.0.3.jar](#)]
- [ChangeLog](#)

→ click here and download a jarfile.

4. Copy jar files

Check whether your java is 32-bit or 64-bit. In Apps & features (see Chap. 1), you will find “Java8 Update xxx **(64-bit)**” for 64-bit, and just “Java8 Update xxx” for 32-bit. [It doesn’t explicitly write 32-bit.]

Now, we will copy the jar files to an extension folder, which is located at

32-bit: c:\Program Files (x86)\Java\jre1.8.x_xxx\lib\ext

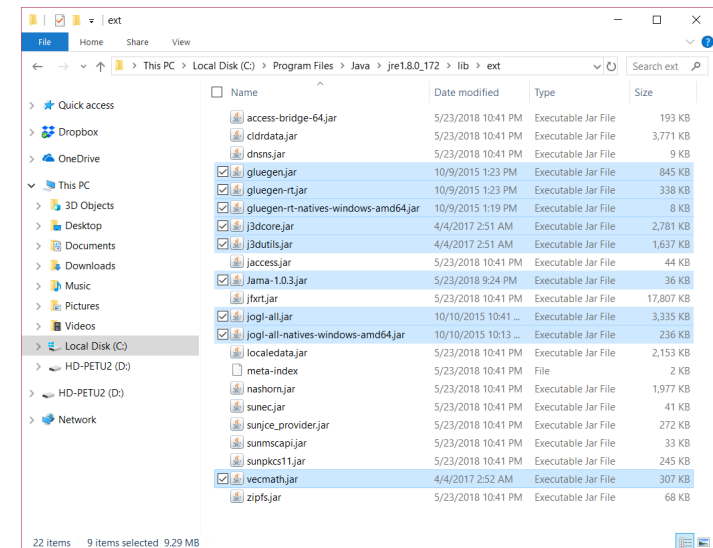
64-bit: c:\Program Files\Java\jre1.8.x_xxx\lib\ext

Copy the following jar files in this folder,

```
gluegen-rt.jar
gluegen.jar
gluegen-rt-natives-windows-xxx.jar
jogl-all.jar
jogl-all-natives-windows-xxx.jar
```

```
j3dcore.jar
j3dutils.jar
vecmath.jar
```

```
Jama-1.0.3.jar
```



where **xxx** = i586 (32-bit) or amd64 (64-bit).

5. Download and test JSindo

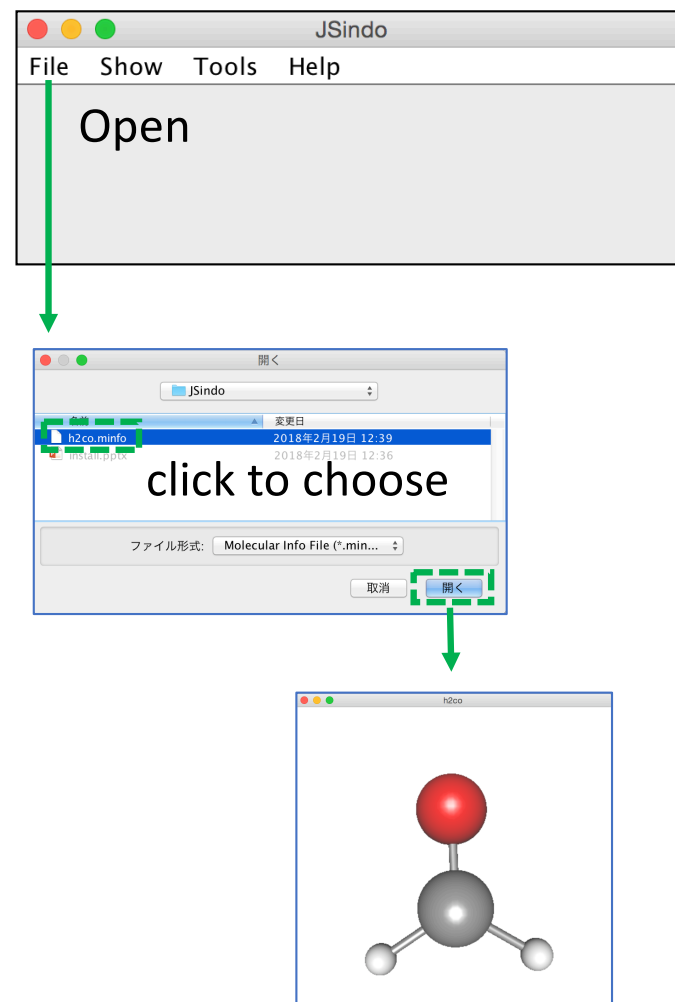
Download JSindo-4.x.jar and sample.zip (or sample.tar.gz) from our website:
<http://www.riken.jp/TMS2012/tms/en/research/software/sindo/index.html>

Now, double click JSindo-4.0.jar. You should see a control panel of JSindo. If you don't see the panel, review the installation of Java.

Let's open "sample/h2co.minfo" included in sample.zip. Double click sample.zip to unzip the file.

In JSindo control panel, click File -> Open, choose "h2co.minfo", and click Open. If you see formaldehyde, you're done with the first step!

If this step fails, it is highly likely that JogAmp/Java3D has a problem. Double check if the right jarfiles (xxx = amd64 or i586) are located in the folder.



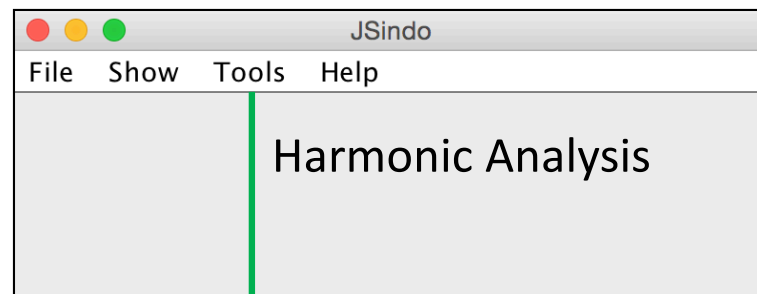
Finally, goto Tools -> Harmonic Analysis. This should create a panel of “Normal modes”.

If you don't see this panel, JAMA isn't working. Make sure the jarfile of JAMA is copied to the extension folder.

If the panel appears, you're all set! Congratulations!

Check on “show vibrational coordinates”, and choose a mode you want to see. Vibrational motion will be indicated by arrows. You can “Invert the arrows” by a check box, and change the magnitude using a slider.

Thanks for using JSindo!
Enjoy!

A screenshot of the "Normal modes (h2co)" window. It contains a table with 4 columns: Mode, Frequency (cm...), Reduced Mass (...), and Intensity (km m...). There are 6 rows of data. Below the table are two checkboxes: "Show vibrational coordinates." and "Invert the arrows.", and a slider bar.

Mode	Frequency (cm...)	Reduced Mass (...)	Intensity (km m...)
1	1196.9147	1.3615	7.0342
2	1266.7685	1.3335	9.3885
3	1540.1545	1.1550	10.7003
4	1752.9374	5.7700	67.7530
5	2973.6886	1.0439	66.6832
6	3047.6560	1.1221	88.4298

