JOGL INSTALLATION

This chapter covers setting up of the environment to use JOGL on your system using different Integrated Development Environments *IDEs*.

Installing JOGL

For JOGL Installation, you need to have following system requirements:

System Requirements

The first requirement is to have the Java Development Kit *JDK* installed in your machine.

Requirement	Description
JDK Version	1.4 or above
Memory	no minimum requirement
Disk Space	no minimum requirement
Operating System	no minimum requirement

You need to follow the given steps to setup your environment to start with JOGL application development:

Step 1 - Verifying Java installation on your machine

Open console of your system and execute the following java command:

Platform	TASK	COMMAND
Windows	Open Command Console	C:\>java-version
Linux	Open Command terminal	$java- version
MAC	Open Terminal	Machine:~ joseph$ java -version

Verify the output on the respective operating system.

Platform	Output
Windows	Java "1.6.0.21" javaTM SE Runtime Environment$build1..6.0_21 - b07$Java HotSpotTM Client VM $buld17.0 - b7, mixed mode, sharing$
Linux	Java "1.6.0.21" javaTM SE Runtime Environment$build1..6.0_21 - b07$Java HotSpotTM Client VM $buld17.0 - b7, mixed mode, sharing$
MAC	

Java "1.6.0.21"

javaTM SE Runtime Environment $build 1..6.0_2 1 - b07$ Java HotSpotTM Client VM $buld 17.0 - b7, mixed mode, sharing$

Step 2 – Setting up Java Development Kit JDK

If java is not installed on your machine, then you need to install Java SDK from Oracle website: www.oracle.com/technetwork/java/javase/downloads/. You can find instructions for installing the JDK from the downloaded files. You need to follow the given instructions to install and configure the setup. Finally, set PATH and JAVA_HOME environment variables to refer to the directory that contains java.exe and javac.exe files, typically java_install_dir/bin and java_install_dir respectively.

Set **Java-home** environment variable to point to the base directory location on the same path, where Java is installed on your machine.

Platform	Command
Windows	Set the environment variable JAVA_HOME to C:\ProgramFiles\Java\Jdk1.6.0_21
Linux	Export JAVA_HOME=/usr/local/java-current
MAC	Export JAVA_HOME=/Library/Java/Home

Append Java compiler location to SystemPath as follows:

Platform	Command
Windows	Append the string ;%JAVA_HOME% bin at the end of the system variable and path
Linux	Export PATH=$PATH$:JAVA_HOME/bin/
MAC	Not required

Step 3 – Downloading JOGL

- You can download latest version of JOGL from the website www.jogamp.org

- Go to the home page of www.jogamp.org

- Click on Builds/Downloads > Current zip.

This takes you to the list of .jar files for all APIs maintained by the website.
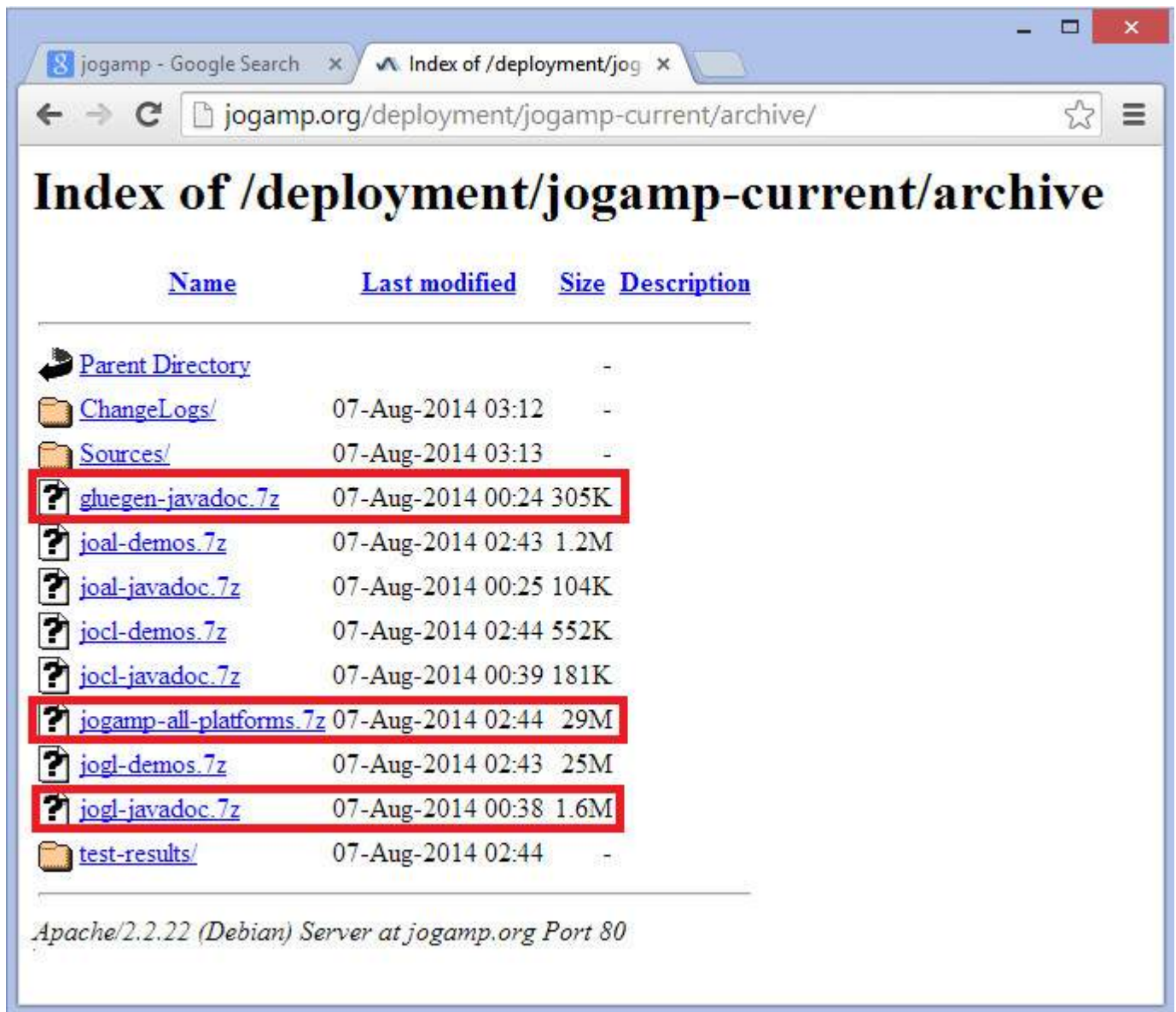

- Download the library .jar file **jogamp-all-platforms.7z,** java documentations for OpenGL native library **glugen-javadoc.7z,** and JOGL **jogl-javadocs.7z.**

- Extract the downloaded .jar files using any zip extracting software.

- When you open the extracted folder you will find jar folder, source-codes, and other files.
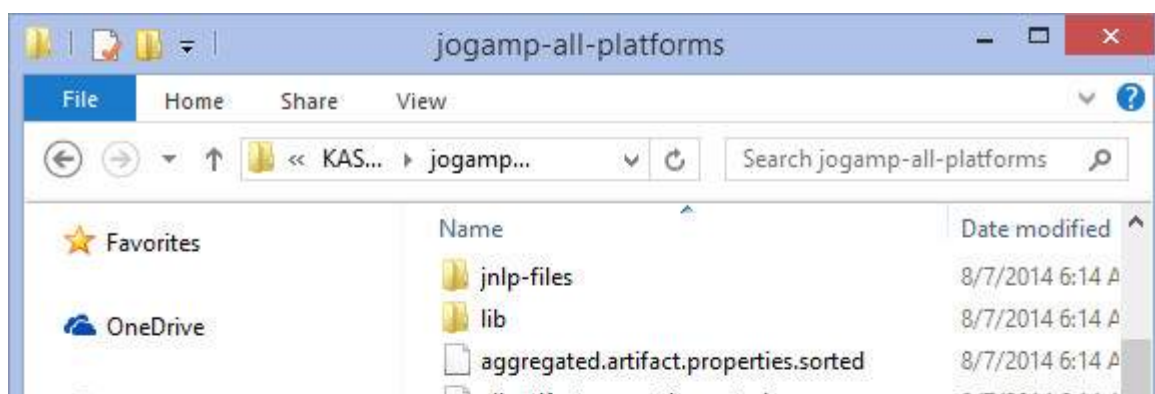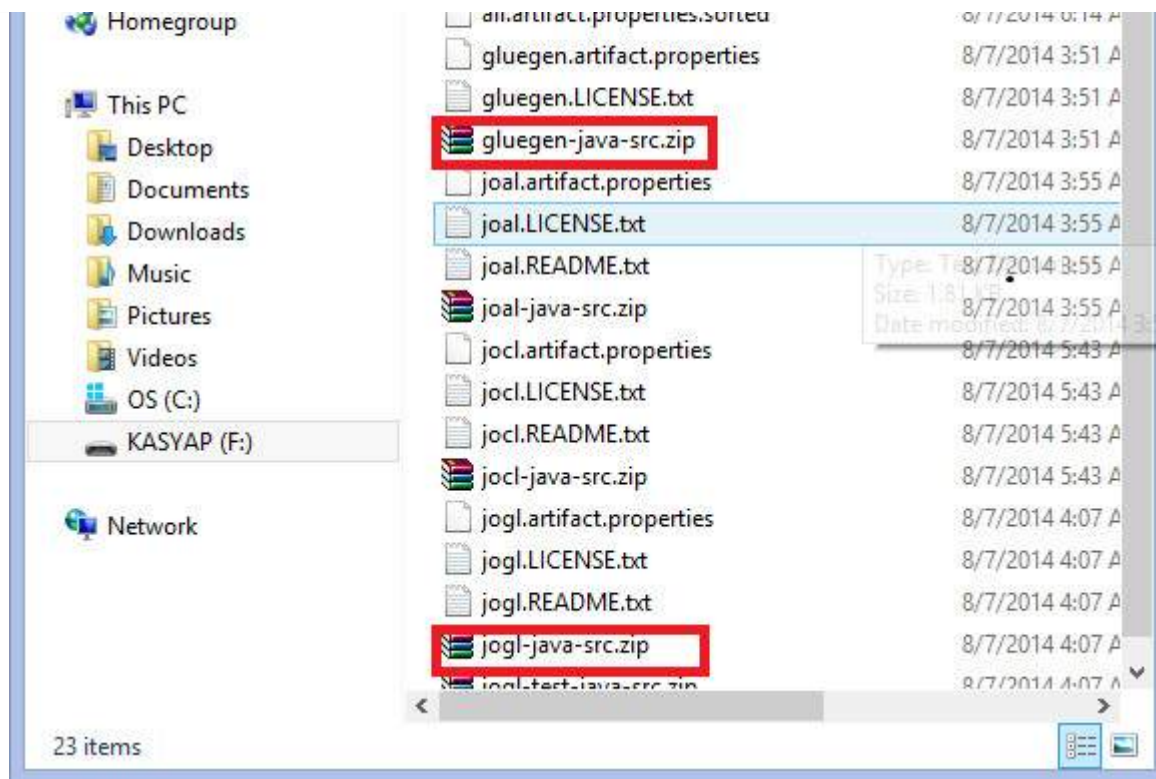
- Get the source codes **gluegen-java-src.zip** and **jogl-java-src.zip** for supporting IDE. This is optional.

  - Inside the jar folder, there are multiple .jar files. This collection of files belongs to Glugen and JOGL.

  - JOAMP provides native libraries that support various operating systems such as Windows, Solraris, Linux and Android. Hence, you need to take appropriate jar files which can execute on your desired platform. For example, if you are using windows 64bit operating system, then get the following .jar files from the *jar* folder:

  I. gluegenrt.jar

  II. jogl-all.jar

  III. gluegen-rt-natives-windows-amd64.jar

  IV. jogl-all-natives-windowsamd64.jar
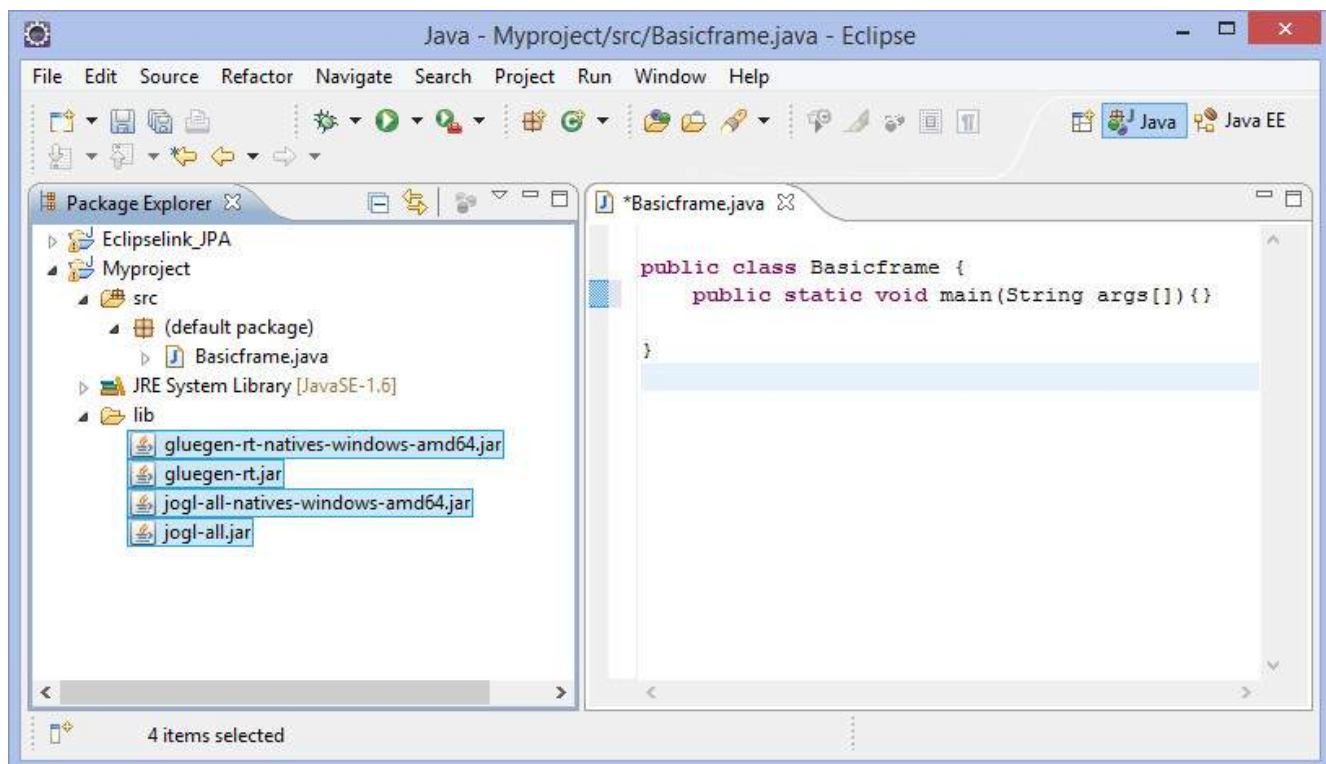
52 items   4 items selected 3.51 MB
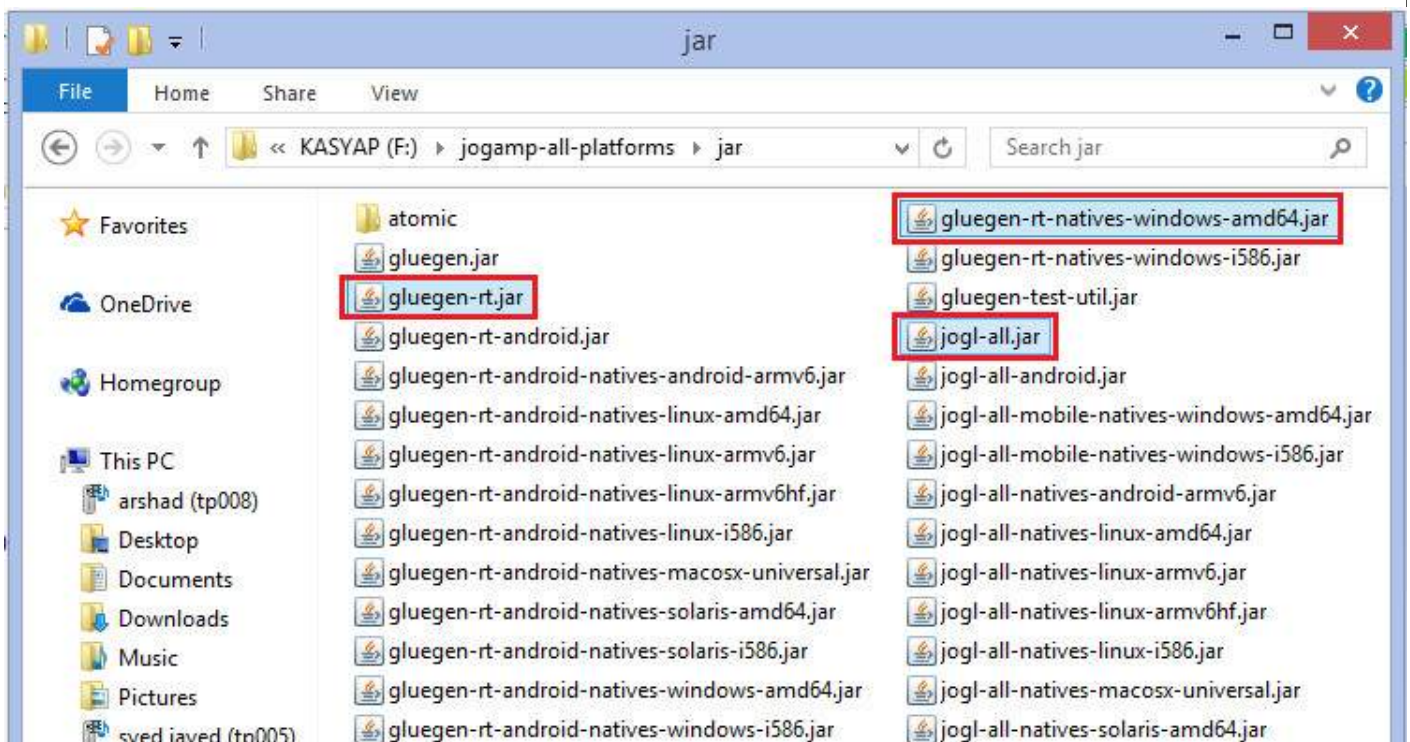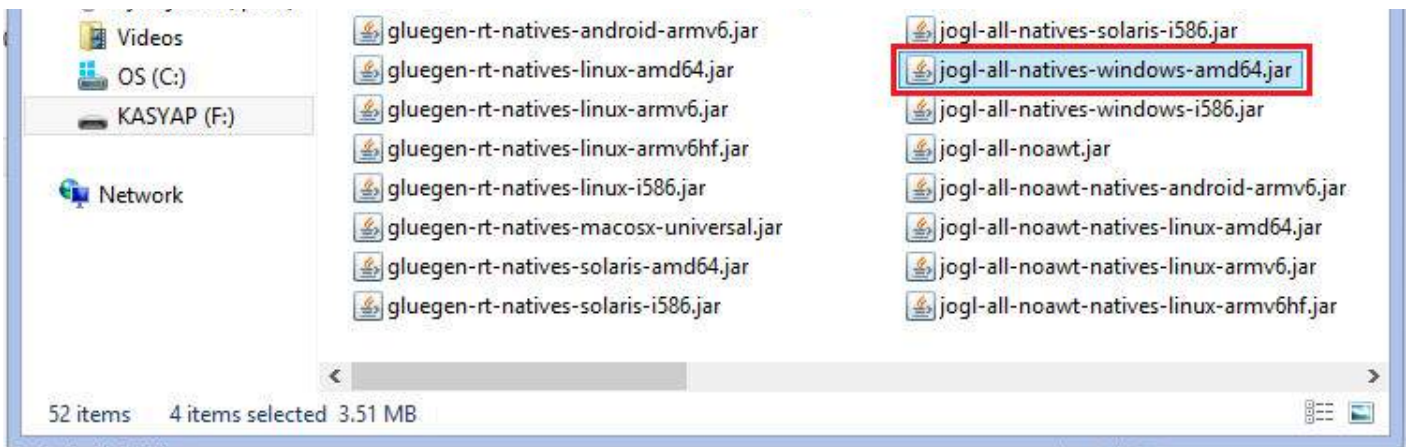
Setting up JOGL for eclipse 4.4

Follow the given procedure for setting up JOGL:

Adding libraries

1. Open eclipse.

2. Create a new project.

3. Create a new folder named *lib* in the project folder.

4. Copy the files **gluegen-rt-natives-windows-amd64.jar, gluegen-rt.jar, jogl-all-natives-windowsamd64.jar** and **jogl-all.jar** into the *lib* folder.
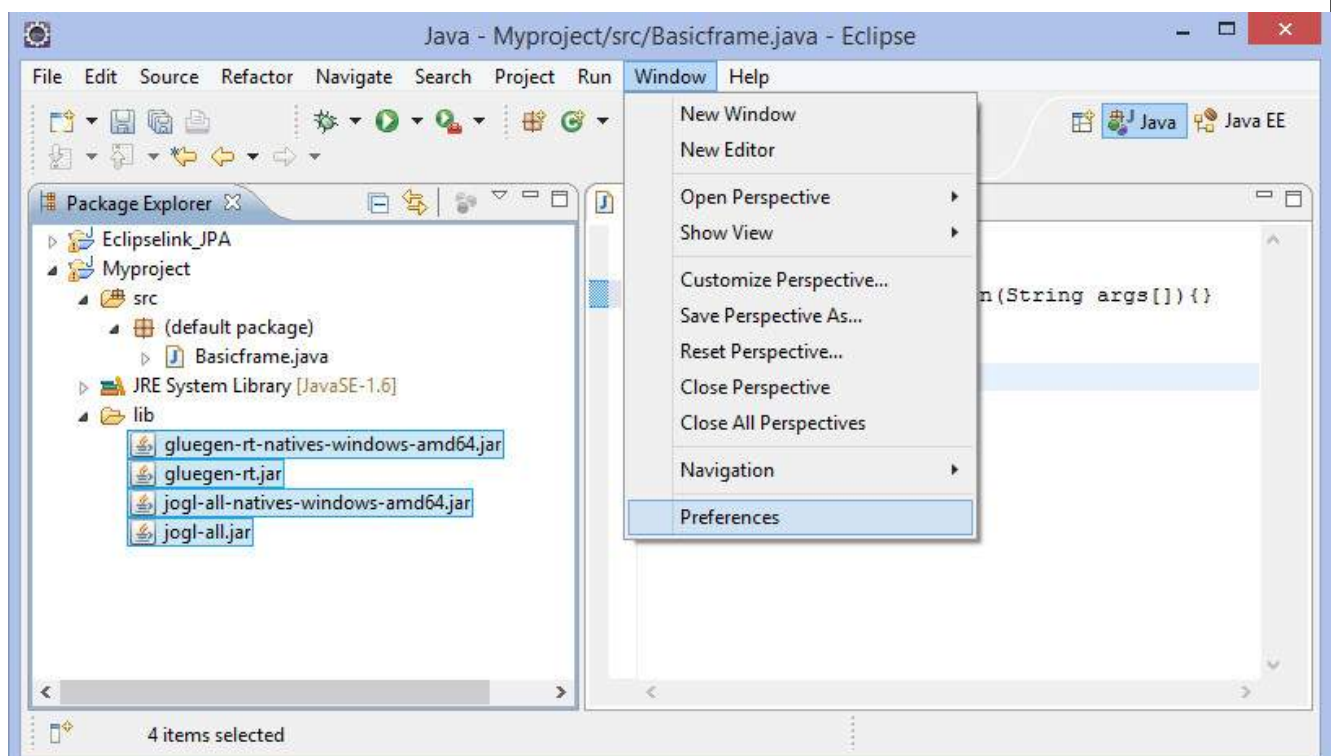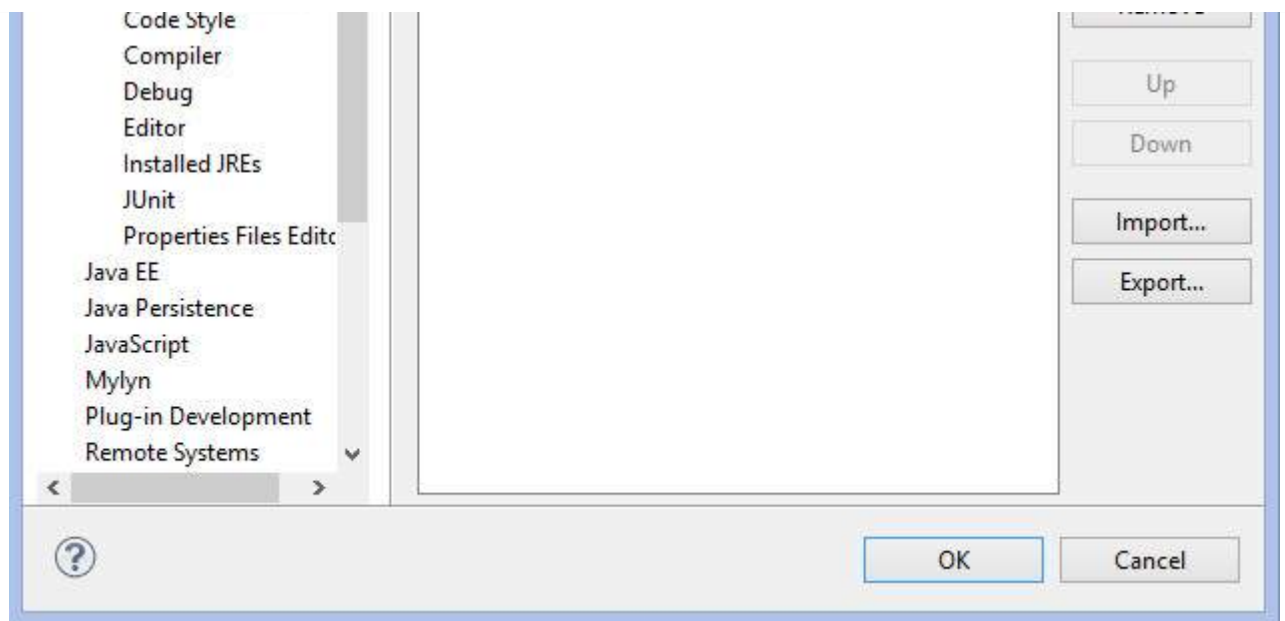

5. Now select these files and right click your mouse button. A shortcut menu is displayed, which contains **Build Path > Add to Build Path.**
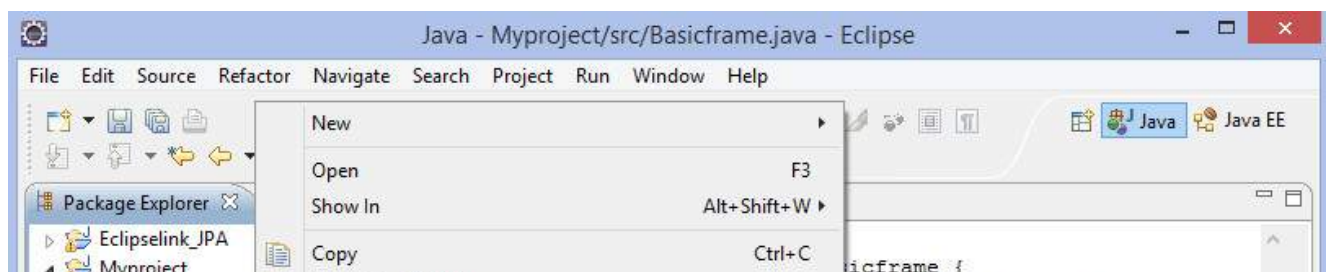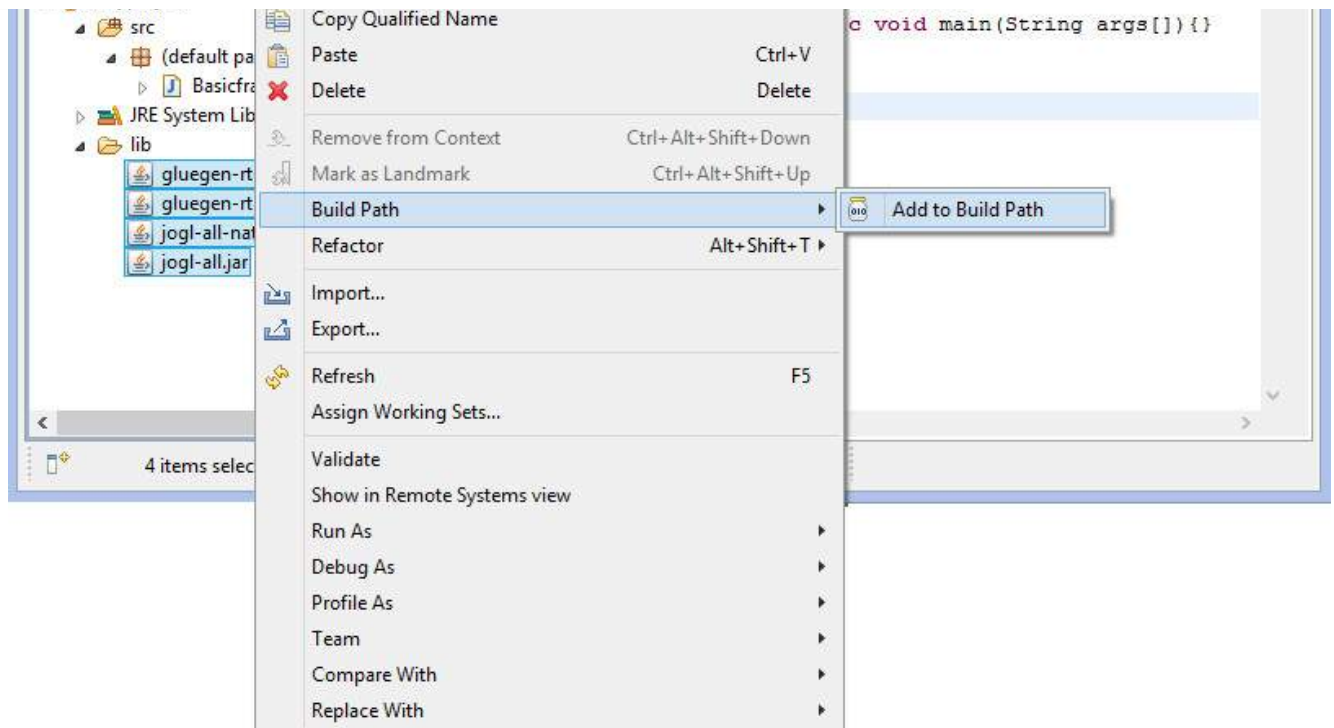
6. To make all .jar files available to other projects, go to main menu. Select Window > Preferences. The Preferences window appears.

1. In preferences window, in the drop down menu on the left hand side, follow the hierarchy-Java-> Build Path -> User Libraries.

2. Click on "New…" button.

3. This opens up a dialog box. Enter library name as jogl2.1.

4. Add jar files **glugen-rt.jar** and **jogl-all.jar** using button "Add External JARs…".

5. This creates a new user library named **jogl2.1.**

In the same way, we can add java documentation and source code for the added .jar files.

Adding native libraries

1. Expand the jogl-all.jar node, select Javadoc location *none*.

2. Click on "New…" button. Enter the name for JOGL Java Document.

3. Click on "Add External JARs…" button.

4. This opens a dialog box where you need to select the location of JOGL Java documentation, which we already have downloaded earlier.

Adding source code

1. Select the node Native library location: *None*.

2. Click on "New…" button.

3. Enter name for native libraries and click "OK" button.

4. Click on "Add External JARs…" button.

5. Now select the path where native library files (**'gluegen-rt-natives-windows-amd64.jar and joglall-natives-windows-amd64.jar'**) are located.

6. Repeat the same procedure for source code.

7. We can set the locations for Javadoc, source code and jar files in the same way as given above for both native library files **glegen-rt.jar** and **glugen-natives-windows-amd64.jar.**
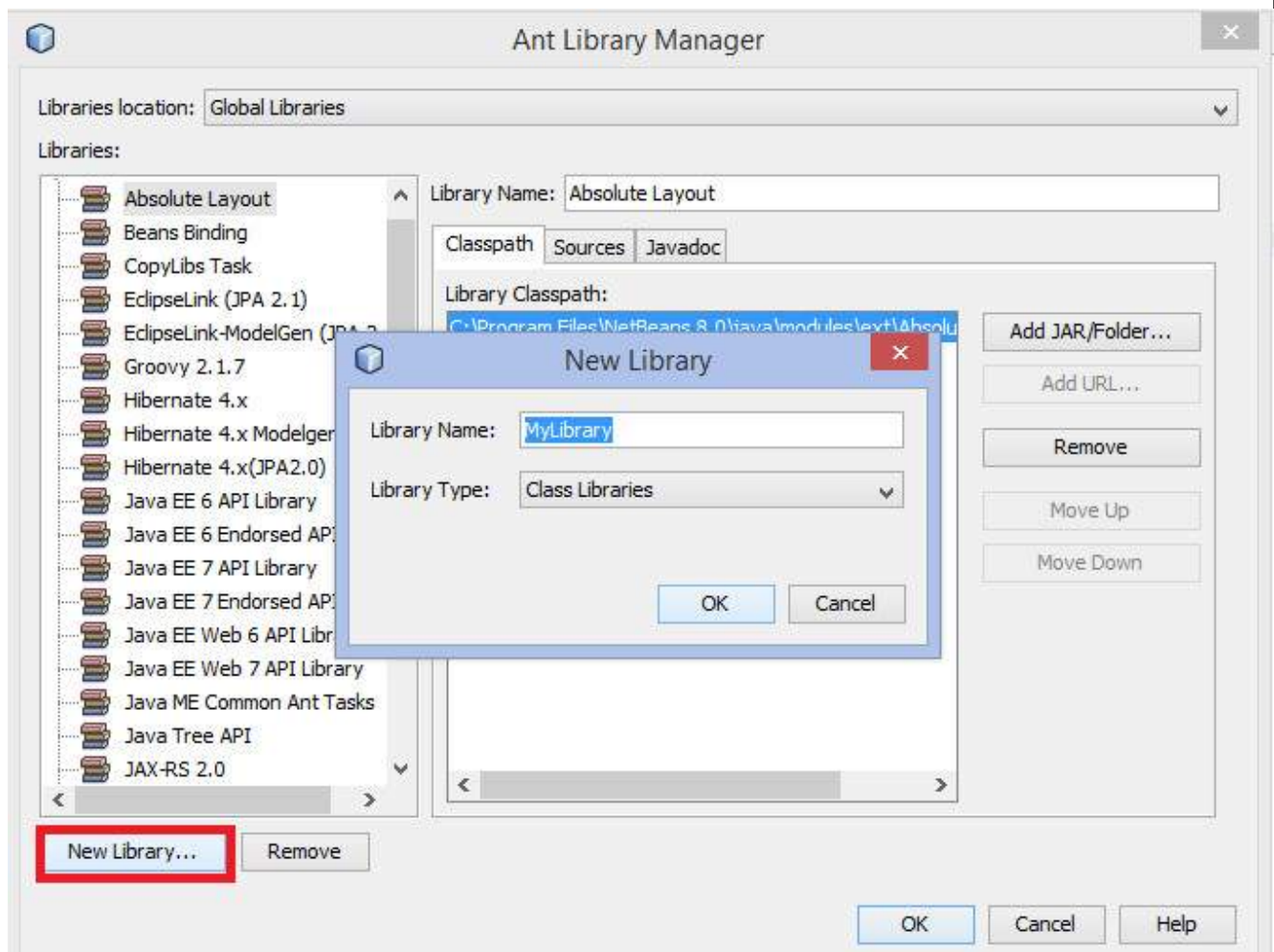
Setting up JOGL for Netbeans4.4

Let us go through the steps for setting up JOGL for Netbeans 4.4:
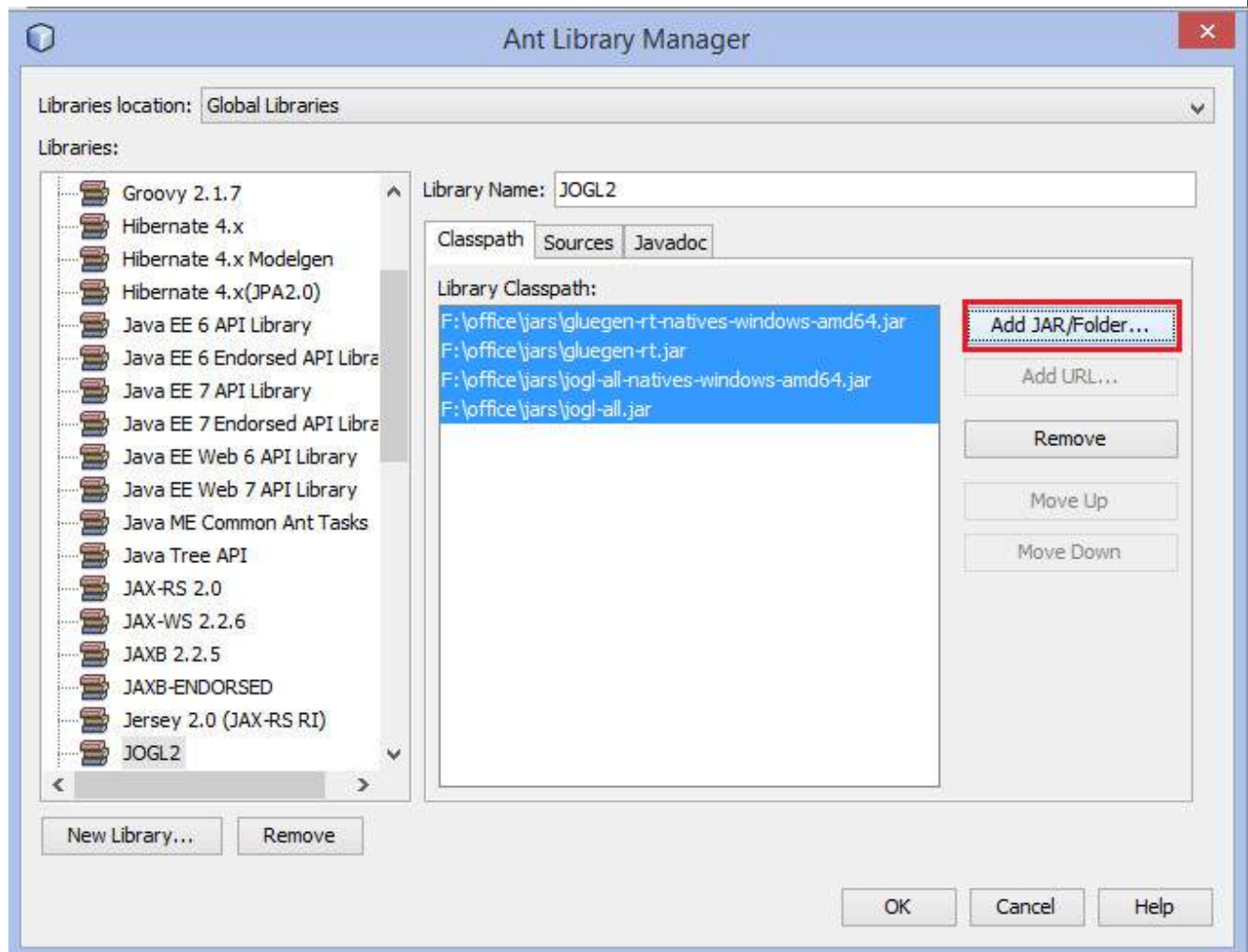
Adding libraries

1. In the main menu, select **Tools > Libraries.**
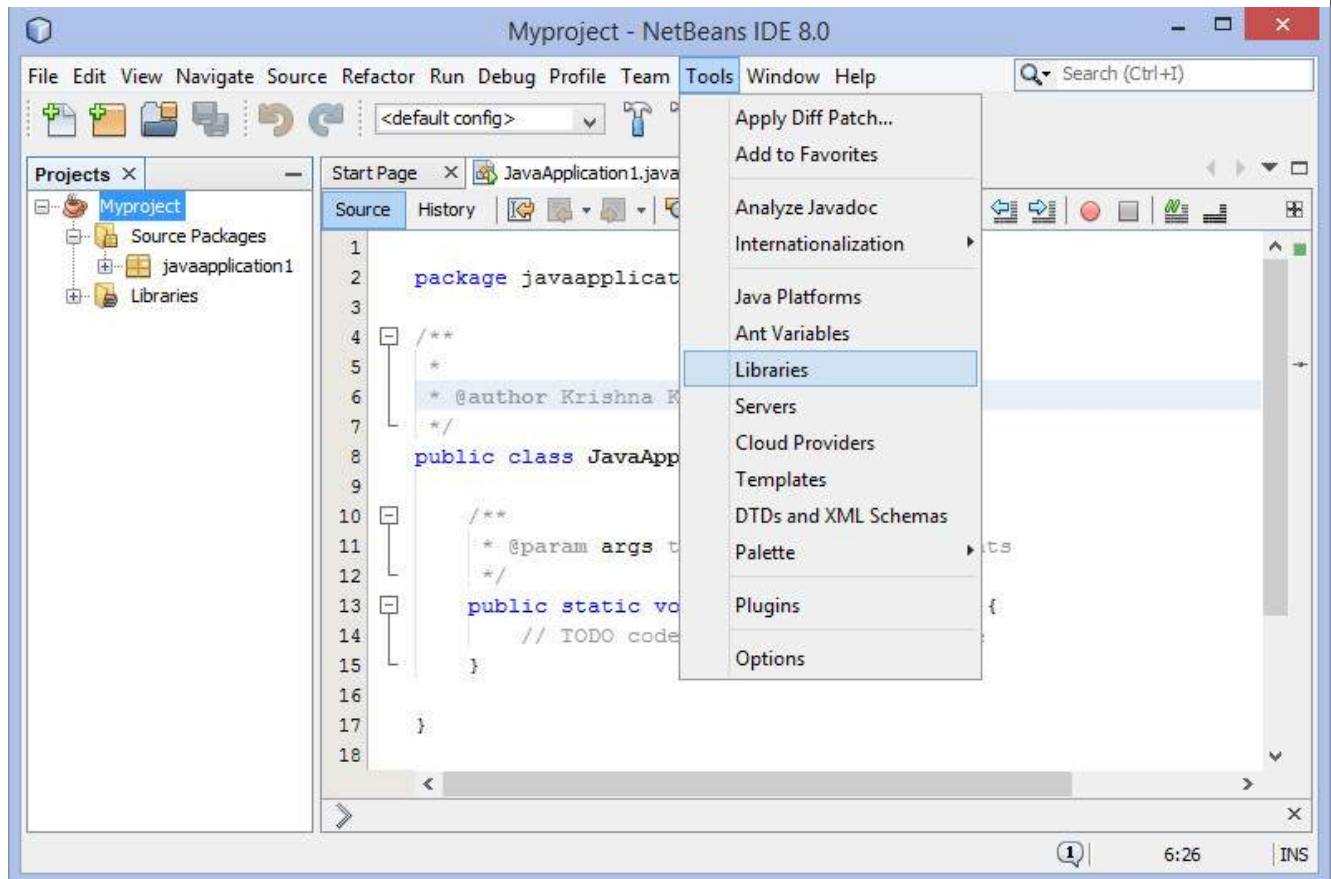
2. This leads you to **Ant Library Manager.**

3. Under the **Classpath** tab, click **New Library** button located on the left lower corner. It opens a small dialog box.

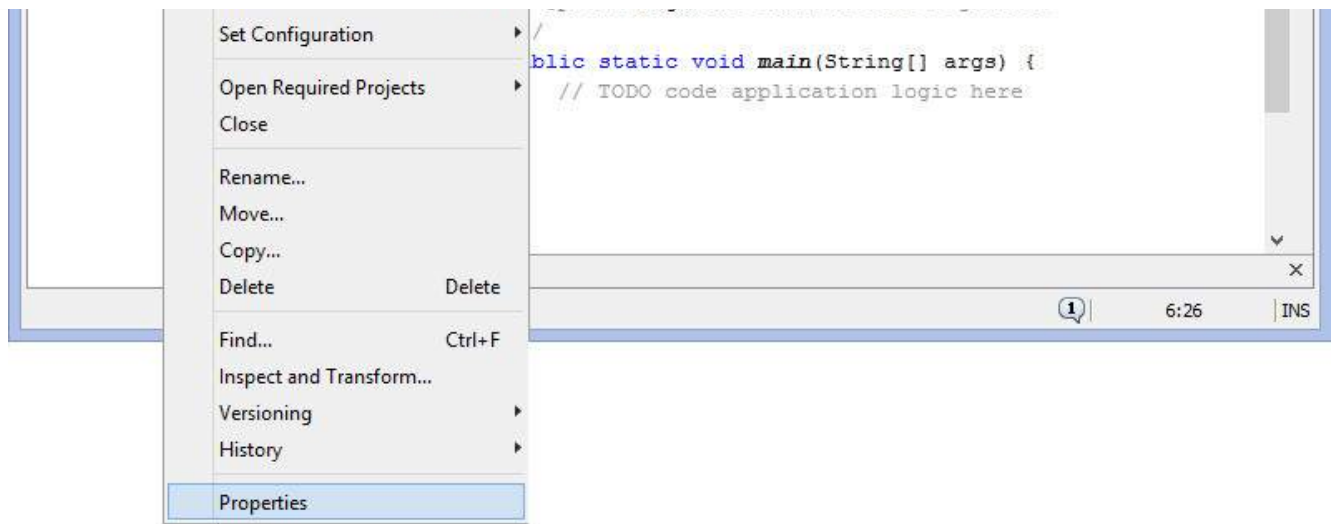4. Enter Library name as **JoGl2.0.**

5. Click on "OK" button.

6. Click on "Add JAR/Folder…" button.

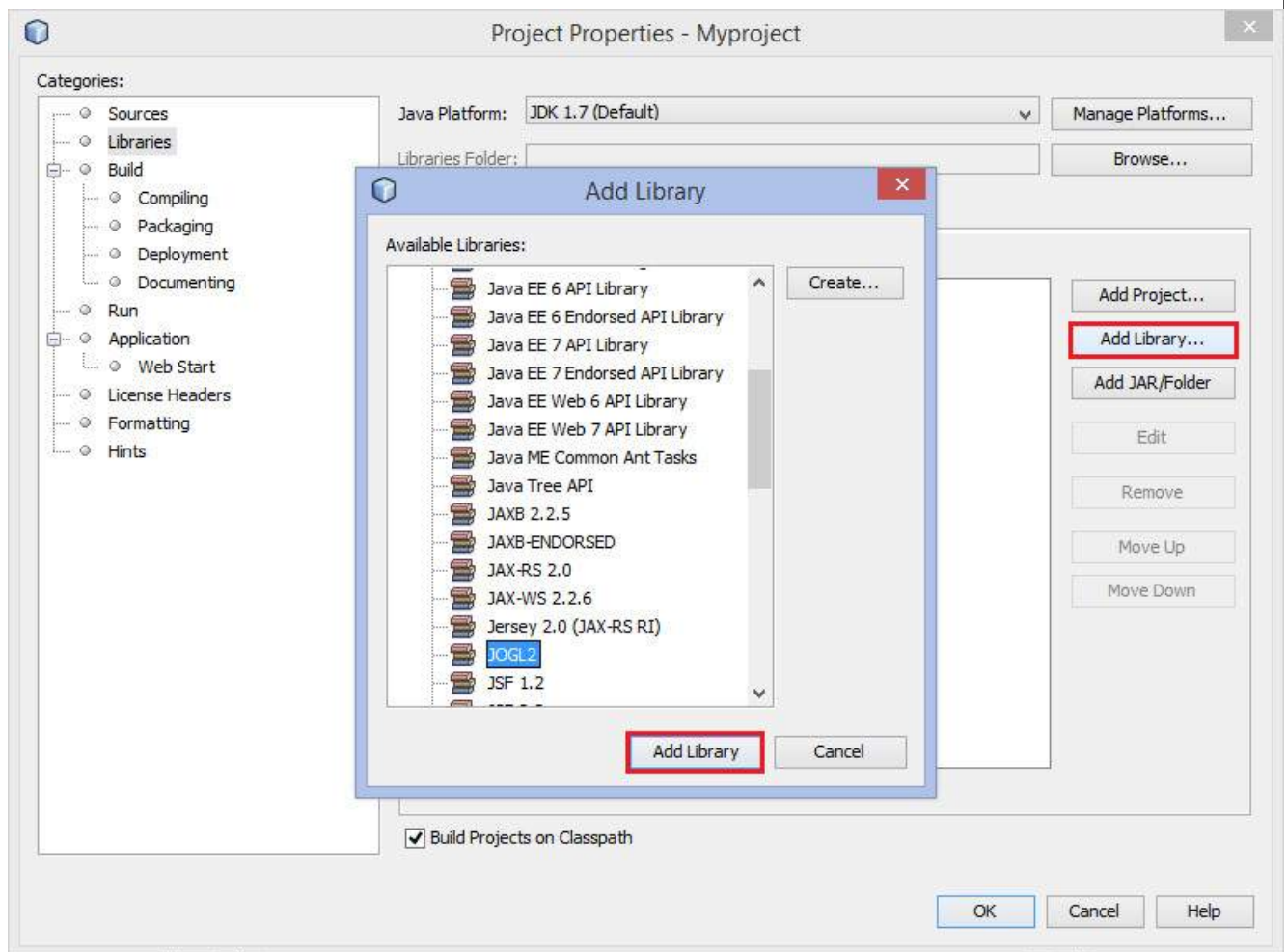7. Select the path where .jar files **jogl.all.jar** and **gluegen-rt.jar** are located.

To include JOGL library into each project, follow the steps given below:

1. Right-click on the **project name.** It shows a short-cut menu.
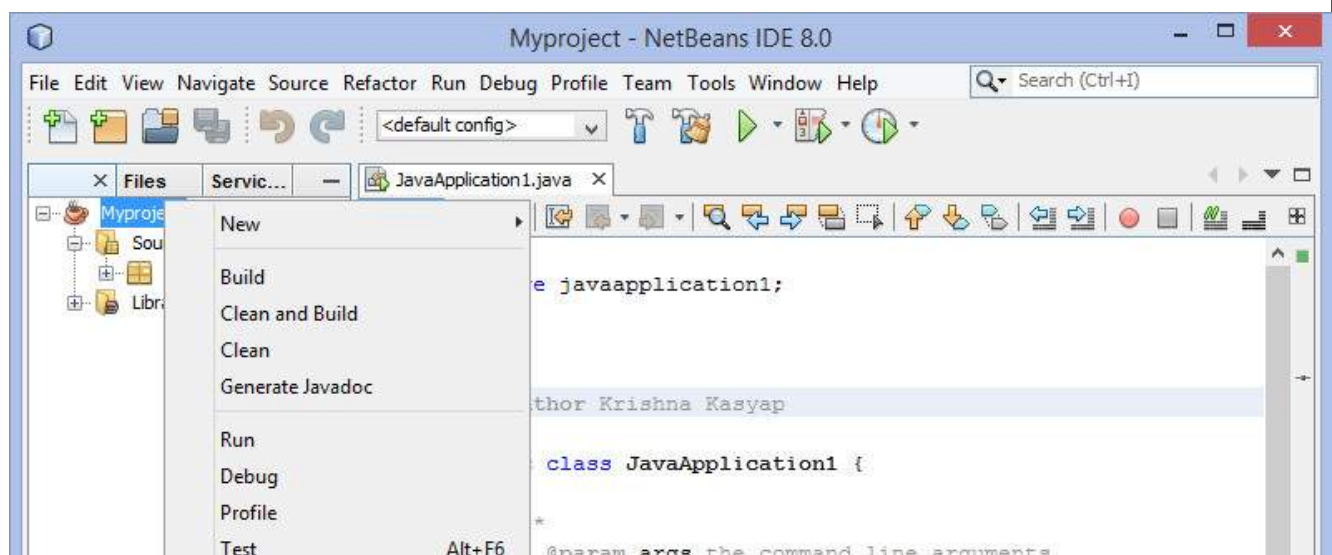
2. Select **Properties.** It opens a window named **Project properties.**

3. Select **Libraries** from Categories on the left hand side

4. Select **Compile tab** and click on "Add Library…" button. Add library dialog box comes up.

5. Now add JOGL2.0 library, which you created earlier.
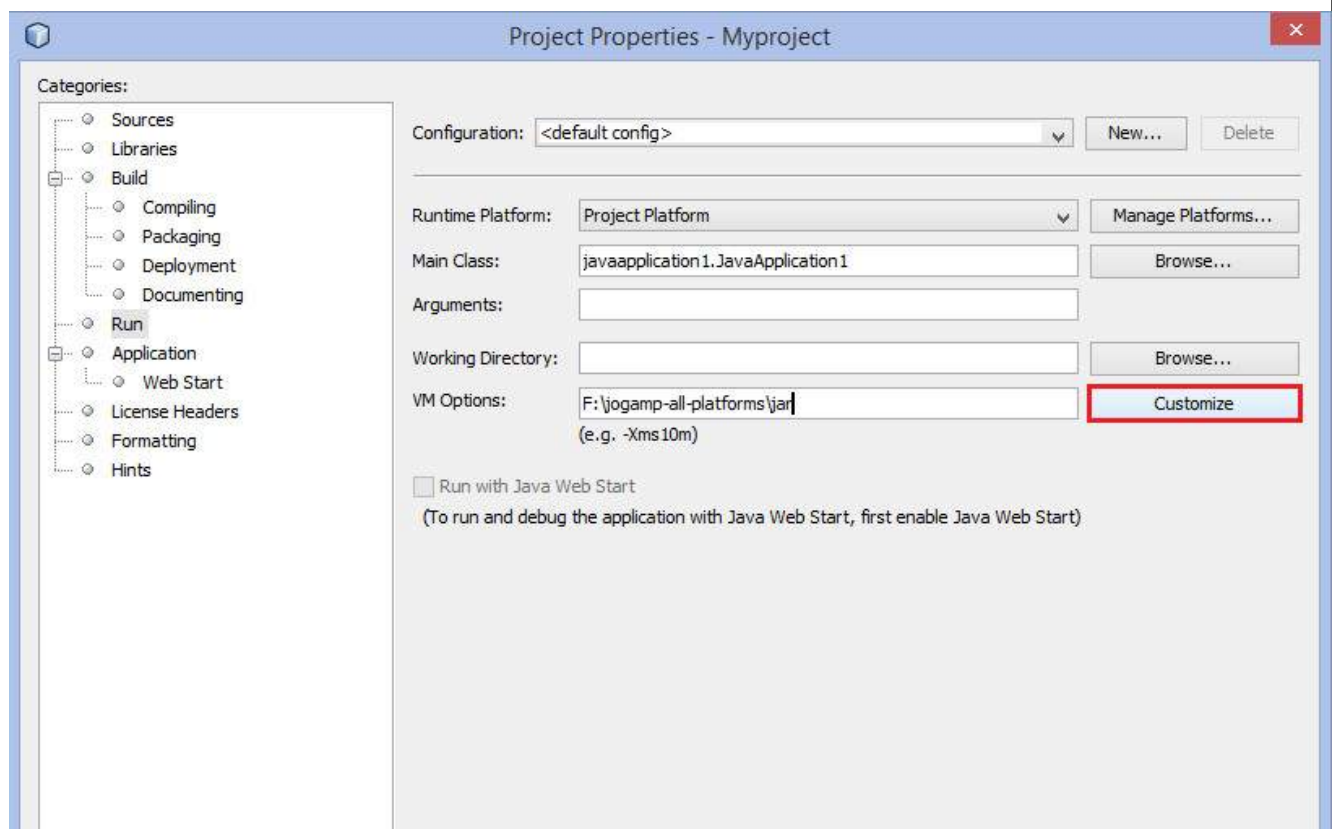
Including native library in each project

Follow the given steps to include native library in each project:

1. Right-click the project.

2. Select **Set Configuration > Customize…**
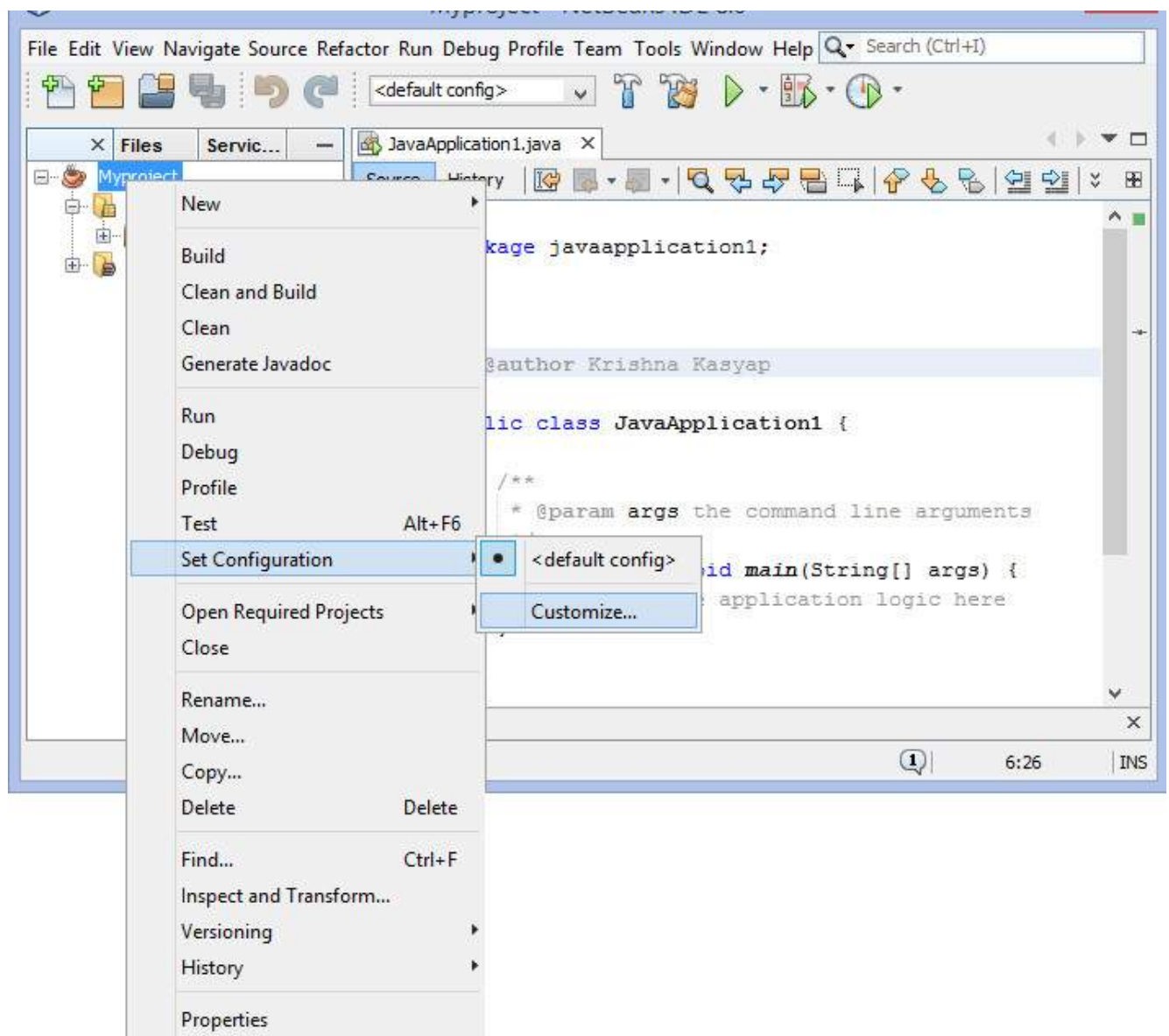
This leads you to **Project Properties** window.

3. On the right hand side, in **VM options,** click on "Customize" button.

4. Browse for the path that contains JOGL native libraries **gluegen-rt-natives-windows-amd64.jar''** and **'jogl-all-natives-windows-amd64.jar.**

Adding Java documentation of native libraries

You need to open Ant library manager again to make sources and Javadoc available for each project. Follow the given procedure:

1. Open **main menu.**

2. Select **Tools > Libraries.** This leads you to **Library manager.**

3. Under the **JavaDoc** tab, click on "New Library…" button.

4. Enter **JOGLJavadoc** name. *Youcanenteranydesiredname.*

5. Click on "Add jars/libraries…" button.

6. Select the path where unzipped **JOGL documentation** code is located.

Adding source code of native libraries

7. Under **Sources** tab, click on "New Library…" button. Enter **JOGLsources** name.

8. Click on "Add jars/libraries…" button. Select the path where unzipped source code is located.

Customizing JDK editor

1. Set **Classpath** for files **jogl.all.jar** and **gluegen-rt.jar.**

2. Set path to native libraries **gluegen-rt-natives-windows-amd64.jar** and **jogl-all-natives-windowsamd64.jar** or copy all the jar files from the folder where you have downloaded them and paste them in to **ise lib** folder.

Loading [MathJax]/jax/output/HTML-CSS/jax.js