



JOGL

java input output package

tutorialspoint

SIMPLY EASY LEARNING

www.tutorialspoint.com



<https://www.facebook.com/tutorialspointindia>



<https://twitter.com/tutorialspoint>

About the Tutorial

Java binding for **OpenGL** (JOGL) is an open source library for binding OpenGL graphics in Java. This tutorial provides a basic understanding of JOGL library and its features. It also explains how to develop 2D and 3D graphics applications using JOGL.

Audience

This tutorial is designed for all enthusiastic students and professionals in the domain of web development who want to learn how to integrate OpenGL bindings in their Java applications.

Prerequisites

You need to have a basic understanding of Java programming with exposure to concepts such as AWT and Swings. In addition, it is required that you have an awareness of OpenGL graphics.

Copyright & Disclaimer

© Copyright 2015 by Tutorials Point (I) Pvt. Ltd.

All the content and graphics published in this e-book are the property of Tutorials Point (I) Pvt. Ltd. The user of this e-book is prohibited to reuse, retain, copy, distribute or republish any contents or a part of contents of this e-book in any manner without written consent of the publisher.

We strive to update the contents of our website and tutorials as timely and as precisely as possible, however, the contents may contain inaccuracies or errors. Tutorials Point (I) Pvt. Ltd. provides no guarantee regarding the accuracy, timeliness or completeness of our website or its contents including this tutorial. If you discover any errors on our website or in this tutorial, please notify us at contact@tutorialspoint.com

Table of Contents

About the Tutorial	i
Audience.....	i
Prerequisites.....	i
Copyright & Disclaimer	i
Table of Contents.....	ii
1. OVERVIEW	1
What is OpenGL?	1
Java Binding for OpenGL API.....	2
Why JOGL?.....	3
History of JOGL	3
2. INSTALLATION.....	4
Installing JOGL	4
Setting up JOGL for Eclipse 4.4	9
Setting up JOGL for NetBeans 4.4.....	13
3. BASIC TEMPLATE.....	19
GLEventListener Interface	19
GLAutoDrawable Interface.....	20
GLCanvas Class.....	20
GLCapabilities Class	22
GLProfile Class	23
Basic Template using Canvas with AWT	25
Using Canvas with Swing.....	29
GLJPanel Class.....	31
Using GLJPanel with Swing Window	33

4. GRAPHICAL SHAPES	36
Drawing Objects.....	36
The Display() Method	36
Drawing a Line	37
Drawing Shapes using GL_Lines	40
Using Parameters of glBegin() for More Shapes	50
5. TRANSFORMATION OF OBJECTS	59
Moving an Object on the Window	59
Applying Color to an Object	61
Applying Color to a Polygon	64
Scaling	67
Rotation	70
Lighting	74
Applying Light to a Rotating Polygon	76
6. 3D GRAPHICS	81
Drawing a 3D Line	81
Drawing a 3D Cube.....	103
Applying Texture to the Cube	108
Appendix	114

1. OVERVIEW

This chapter introduces OpenGL, its functions, the OpenGL bindings in java (GL4java, LWJGL, JOGL), and the advantages of JOGL over other OpenGL bindings.

Java binding for OpenGL (JOGL) is the recent binding for OpenGL graphics API in Java. It is a wrapper library, which can access OpenGL API, and it is designed to create 2D and 3D graphics applications coded in Java. JOGL is an open-source library initially developed by former MIT graduate students Ken Russell and Chris Kline. Later, it was adopted by the gaming group at Sun Microsystems, and now it is maintained by Java on Graphics Audio and Processing (JOGAMP). JOGL functions on various operating systems such as Windows, Solaris, Mac OS X, and Linux (on x86).

What is OpenGL?

OpenGL stands for Open Graphics Library, which is a collection of commands to create 2D and 3D graphics. With OpenGL, you can create complicated 3D shapes using very basic primitives such as points, lines, polygons, bitmaps, and images.

Here are a few features of OpenGL:

- It can work on multiple platforms.
- It has bindings in several languages such as C++, Python, etc.
- It can render 2D and 3D vector graphics.
- It interacts with Graphical Processing Unit (GPU) for achieving speedy and high quality rendering. Rendering is the process of creating an image from a 2D or 3D model.
- It is an industry standard API for writing 3D Graphics applications. For example, games, screensavers, etc.
- It contains around 150 commands, which programmers can use to specify objects and operations to develop applications.
- It contains OpenGL Utility Library (GLU) that provides various modeling features, such as quadric surfaces and NURBS curves. GLU is a standard component of OpenGL.

- The design of OpenGL is focused on efficiency, effectiveness, and its implementation on multiple platforms using multiple languages. To maintain simplicity of an OpenGL API, windowing tasks are not included. Therefore, OpenGL depends on other programming languages for windowing tasks.

Java Binding for OpenGL API

It is a Java Specification Request (JSR) API specification, which allows to use OpenGL on Java platform.

Specifications	Details
JSR 231	This Java binding package supports Java SE platform.
JSR 239	This Java binding package supports Java ME platform.

There are various OpenGL bindings in Java. They are discussed below.

GL4java

It is known as OpenGL for Java technology. It has links to OpenGL 1.3 and to nearly all vendor extensions. Also, it can be used with Abstract Window Toolkit (AWT) and Swings. It is a game focused OpenGL binding, which is a single window that displays full screen applications.

LWJGL

- Light Weight Java Game Library (LWJGL), uses OpenGL 1.5 and works with latest version of java.
- It can use full screen capabilities of JSE 1.4. It has limited support for AWT / Swings.
- It is suitable for lightweight devices such as mobile phones, embedded devices, etc.

JOGL

- JOGL focuses only on 2D and 3D Rendering. The interfaces dealing with sound and input-output are not included in JOGL.



- It includes Graphics Utility Library (GLU), GL Utility toolkit (GLUT), and its own API - Native Windowing Toolkit (NEWT).

Why JOGL?

- It provides full access to the OpenGL APIs (version 1.0, 4.3, ES 1, ES 2 and ES 3) as well as nearly all the vendor extensions. Hence, all the features in OpenGL are included in JOGL.
- JOGL integrates with the AWT, Swing, and Standard Widget Toolkit (SWT). It also includes its own Native Windowing Toolkit (NEWT). Hence, it provides complete support for windowing.

History of JOGL

- 1992 - Silicon Graphics Inc. released the first OpenGL specification.
- 2003 - Java.net website was launched with new features and JOGL was published for the first time on the same website.
- 2010 - Since year 2010, it has been independent open source project under BSD license, which is a liberal license for computer software.

2. 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 on 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" java(TM) SE Runtime Environment(build 1..6.0_21-b07)Java HotSpot(TM) Client VM(build 17.0-b7, mixed mode, sharing)
Linux	Java "1.6.0.21" java(TM) SE Runtime Environment(build 1..6.0_21-b07)Java HotSpot(TM) Client VM(build 17.0-b7, mixed mode, sharing)
MAC	Java "1.6.0.21" java(TM) SE Runtime Environment(build 1..6.0_21-b07)Java HotSpot(TM) Client VM (build 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 the 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 System Path as follows:

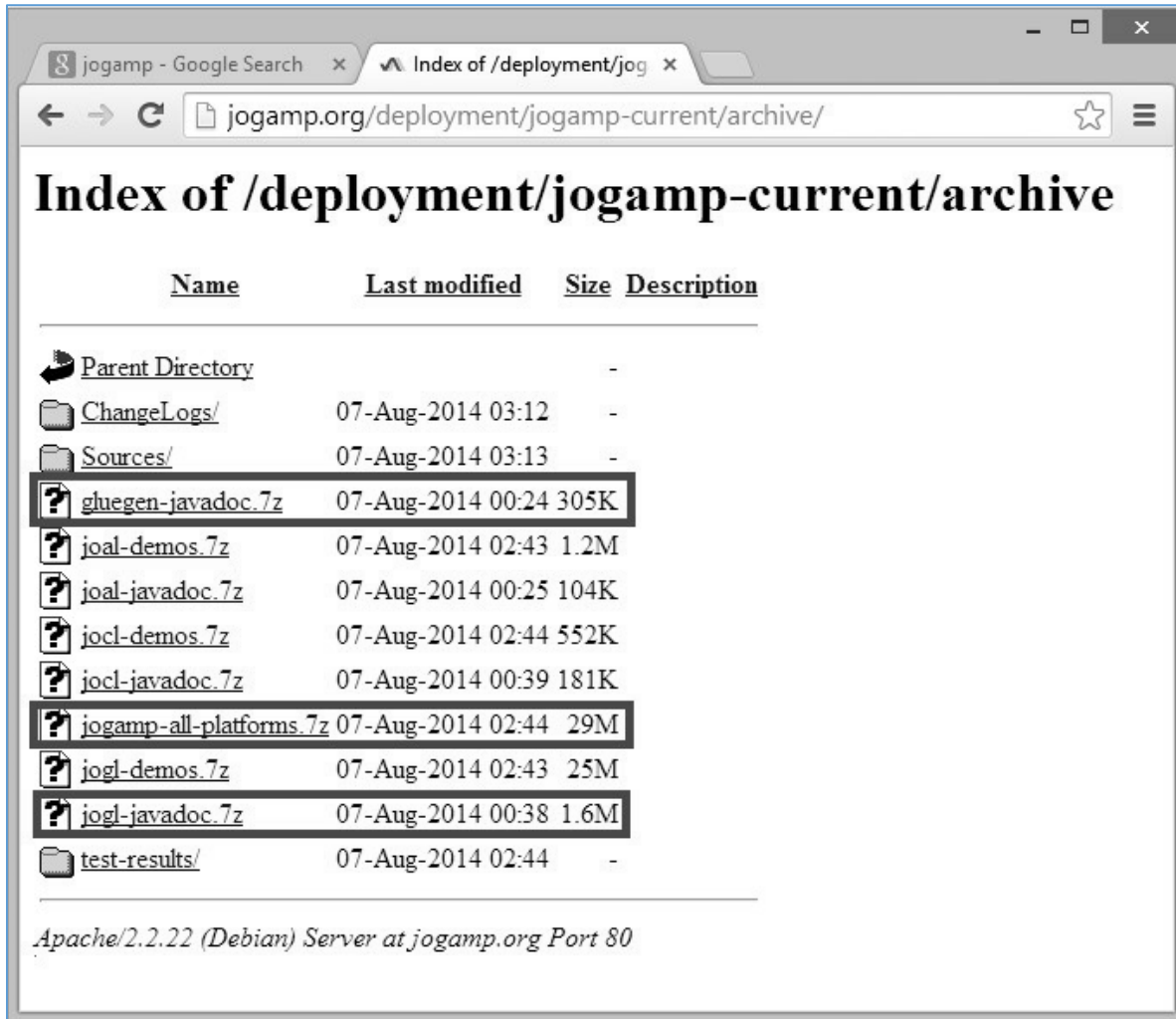
Platform	Command
Windows	Append the string ;%JAVA_HOME% bin at the end of the system variable and path
Linux	Export PATH=\$PATH:\$JAVA_HOME/bin/
MAC	Not required

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.



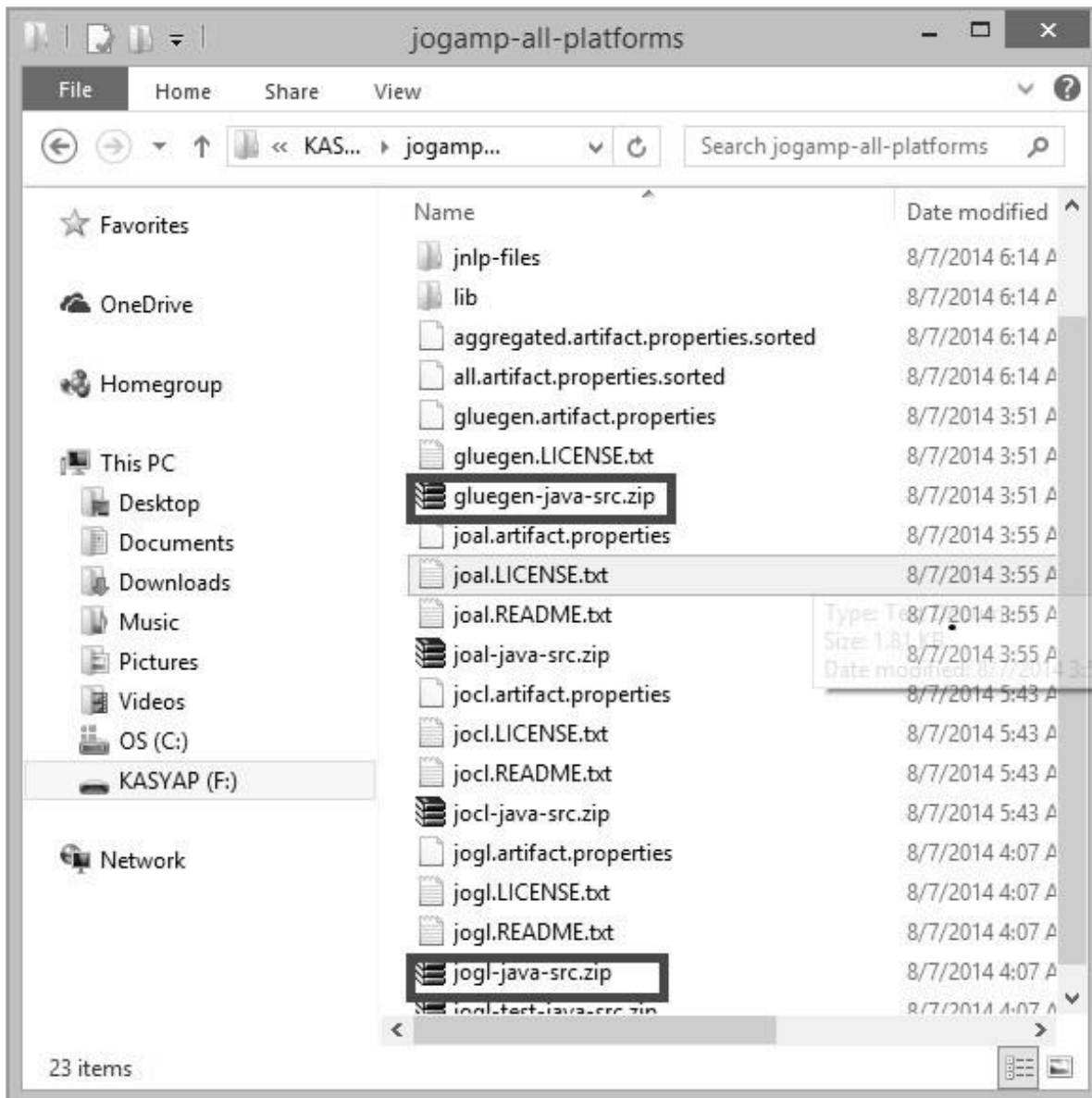
Index of /deployment/jogamp-current/archive

<u>Name</u>	<u>Last modified</u>	<u>Size</u>	<u>Description</u>
Parent Directory	-	-	-
ChangeLogs/	07-Aug-2014 03:12	-	-
Sources/	07-Aug-2014 03:13	-	-
gluegen-javadoc.7z	07-Aug-2014 00:24	305K	
joal-demos.7z	07-Aug-2014 02:43	1.2M	
joal-javadoc.7z	07-Aug-2014 00:25	104K	
jocl-demos.7z	07-Aug-2014 02:44	552K	
jocl-javadoc.7z	07-Aug-2014 00:39	181K	
jogamp-all-platforms.7z	07-Aug-2014 02:44	29M	
jogl-demos.7z	07-Aug-2014 02:43	25M	
jogl-javadoc.7z	07-Aug-2014 00:38	1.6M	
test-results/	07-Aug-2014 02:44	-	-

Apache/2.2.22 (Debian) Server at jogamp.org Port 80

- Download the library .jar file jogamp-all-platforms.7z, java documentations for OpenGL native library gluegen-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.

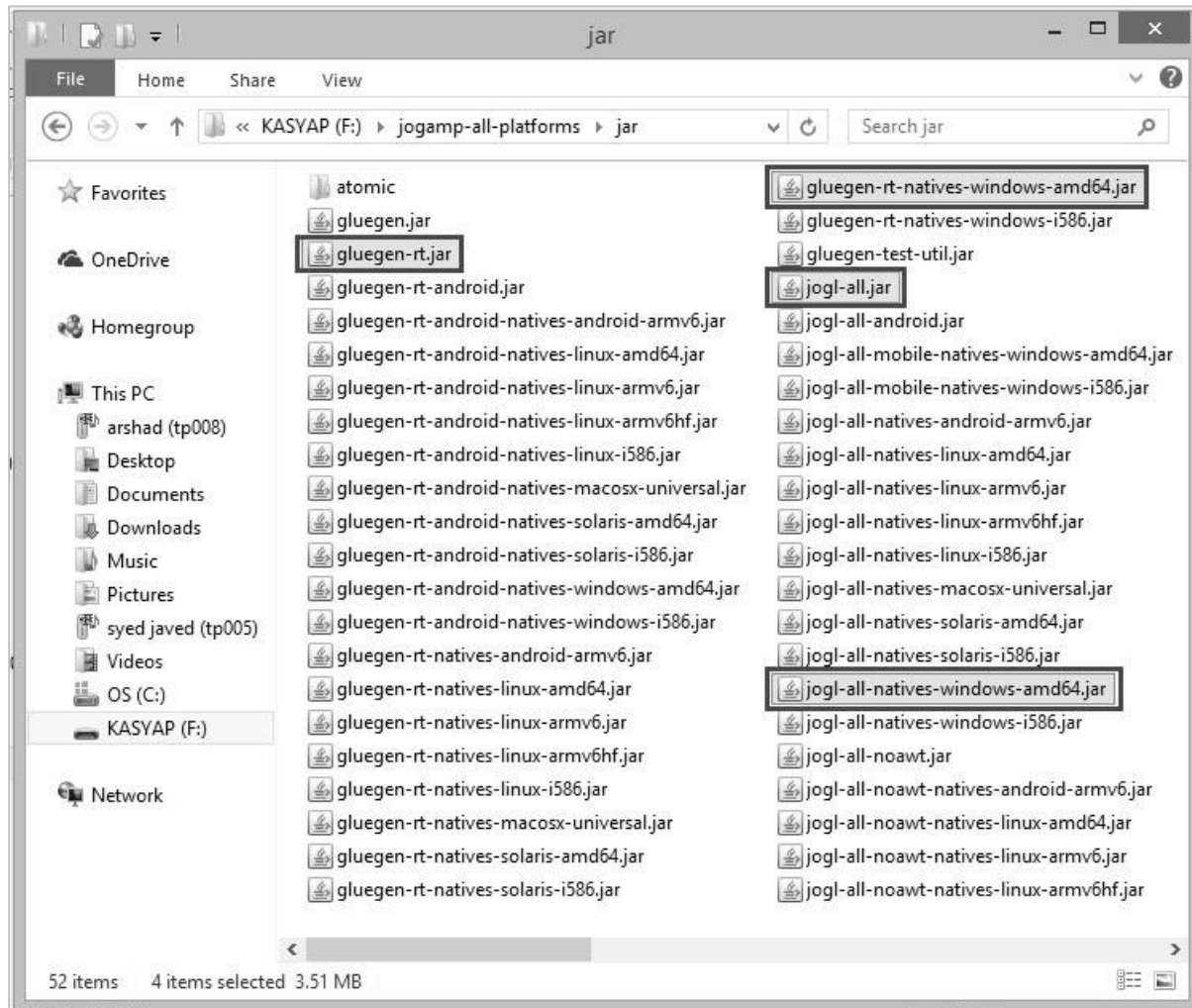


Get the source codes gluegen-java-src.zip and jogl-java-src.zip for supporting IDE. This is optional.

- 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, Solaris, 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 64-bit operating system, then get the following .jar files from the jar folder:

- gluegen-rt.jar
- jogl-all.jar
- gluegen-rt-natives-windows-amd64.jar
- jogl-all-natives-windowsamd64.jar



Setting up JOGL for Eclipse 4.4

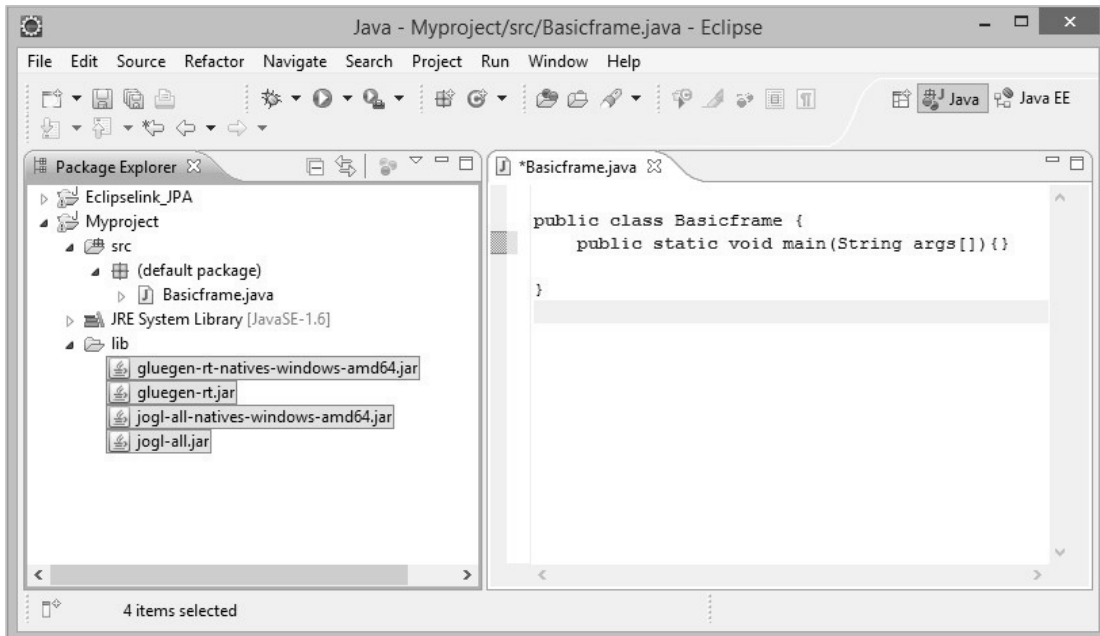
Follow the given procedure for setting up JOGL:

Adding Libraries

1. Open Eclipse.

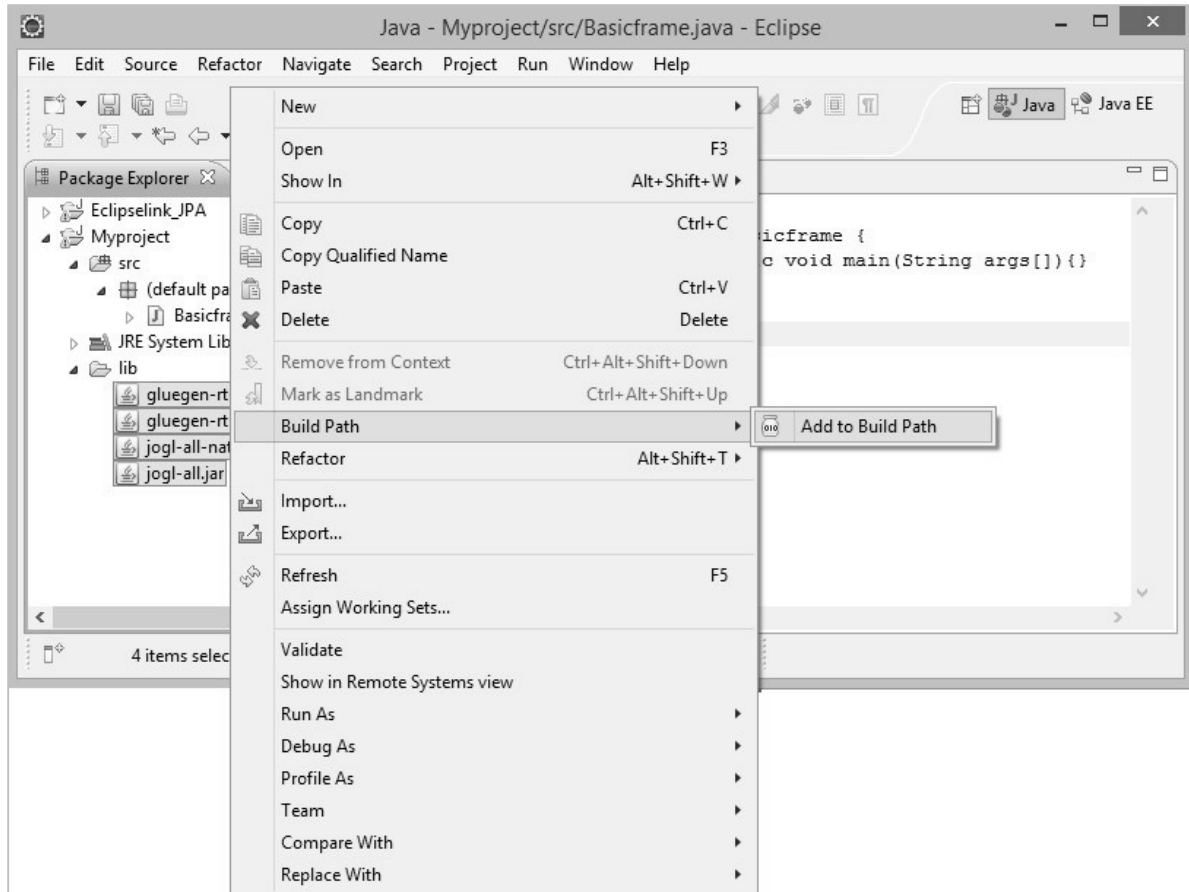


2. Create a new project.
3. Create a new folder named *lib* in the project folder.
4. Copy the files **gluegen-rt-natives-windows-amd64.jar**, **gluegen-rt.jar**, **jogl-all-natives-windows-amd64.jar** and **jogl-all.jar** into the *lib* folder.



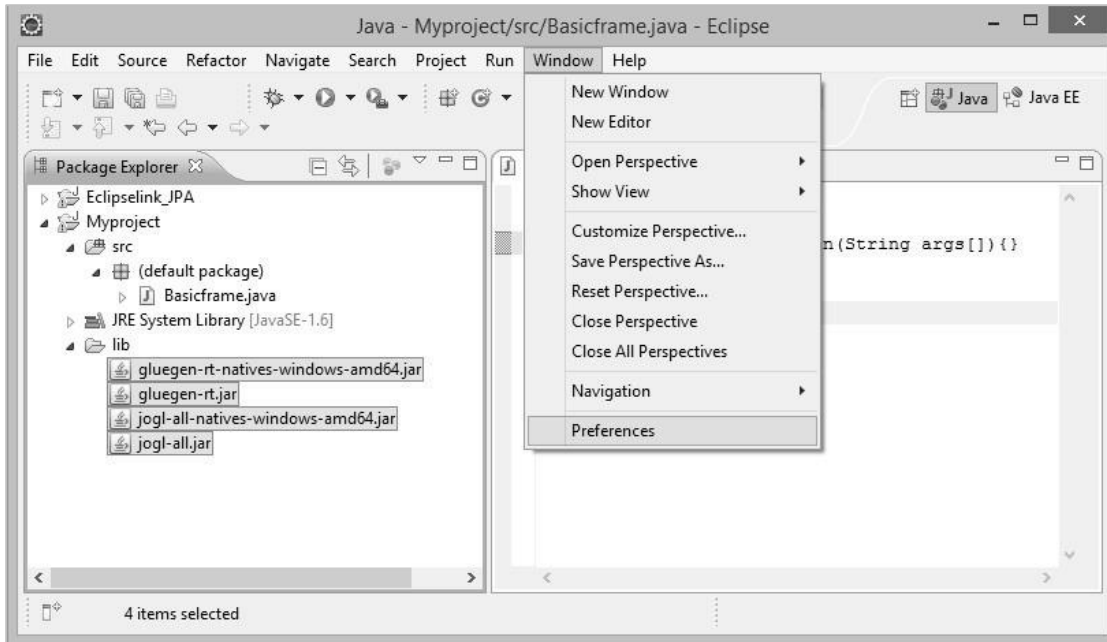
5. Now select these files and right click your mouse button. A shortcut menu is displayed, which contains **Build Path > Add to Build Path**.

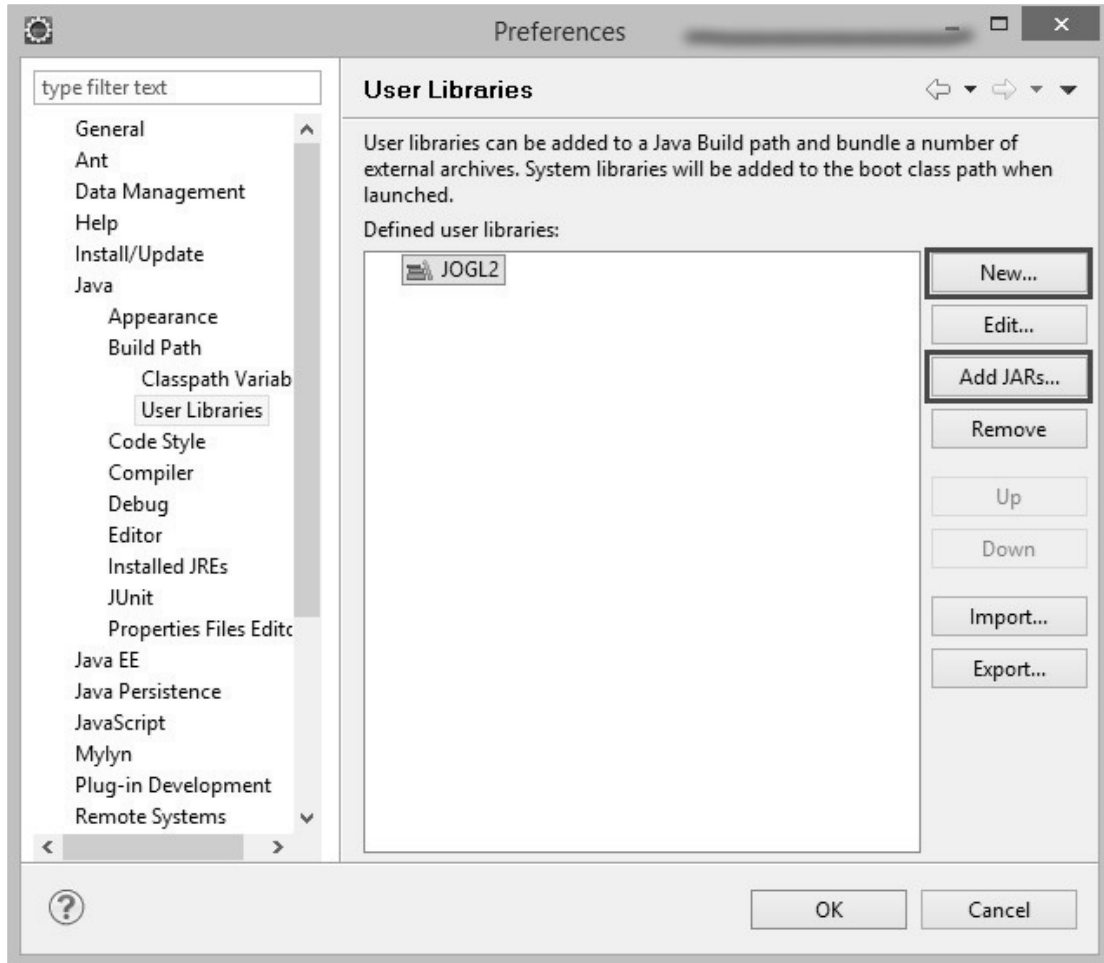
Java Binding for OpenGL (JOGL)



6. To make all .jar files available to other projects, go to main menu. Select Window > Preferences. The Preferences window appears.

Java Binding for OpenGL (JOGL)





- - In preferences window, in the drop down menu on the left hand side, follow the hierarchy- Java-> Build Path -> User Libraries.
 - Click on "New..." button.
 - It opens up a dialog box. Enter the library name as jogl2.1.
 - Add jar files **glugen-rt.jar** and **jogl-all.jar** using button "Add External JARs...".
 - It 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. It 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**.

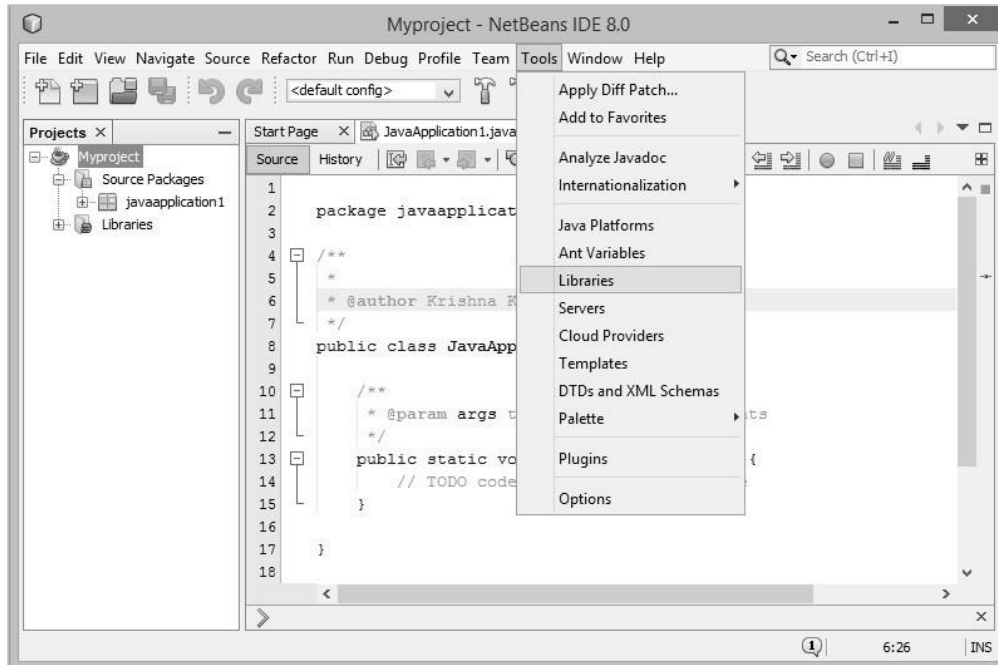
Setting up JOGL for NetBeans 4.4

- Let us go through the steps for setting up JOGL for NetBeans 4.4:

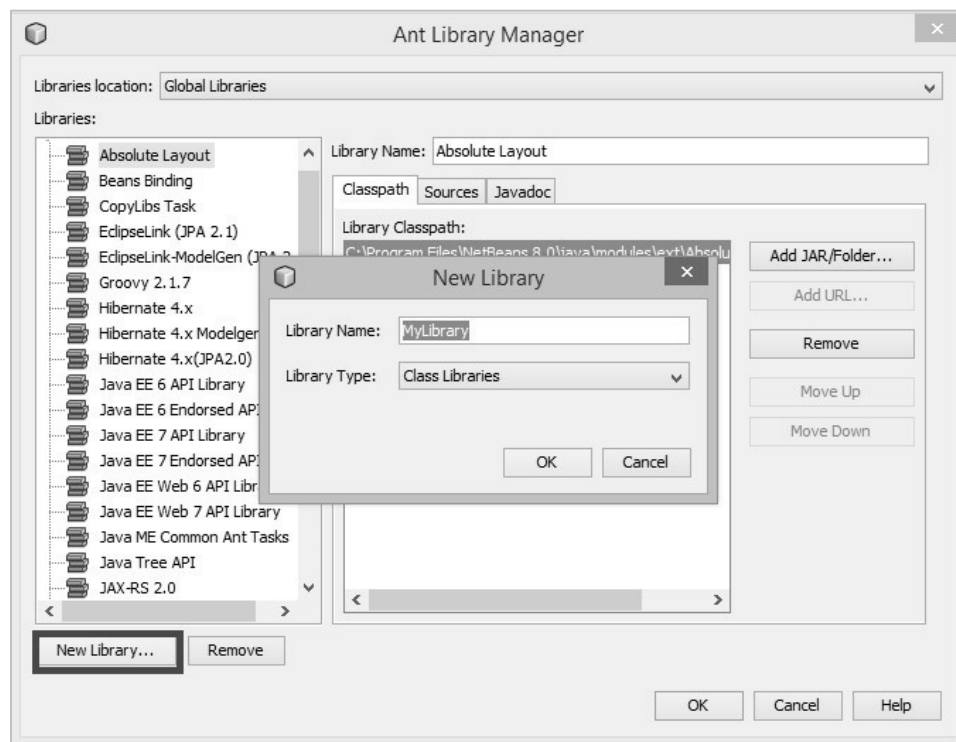
Adding Libraries

1. In the main menu, select **Tools > Libraries**.



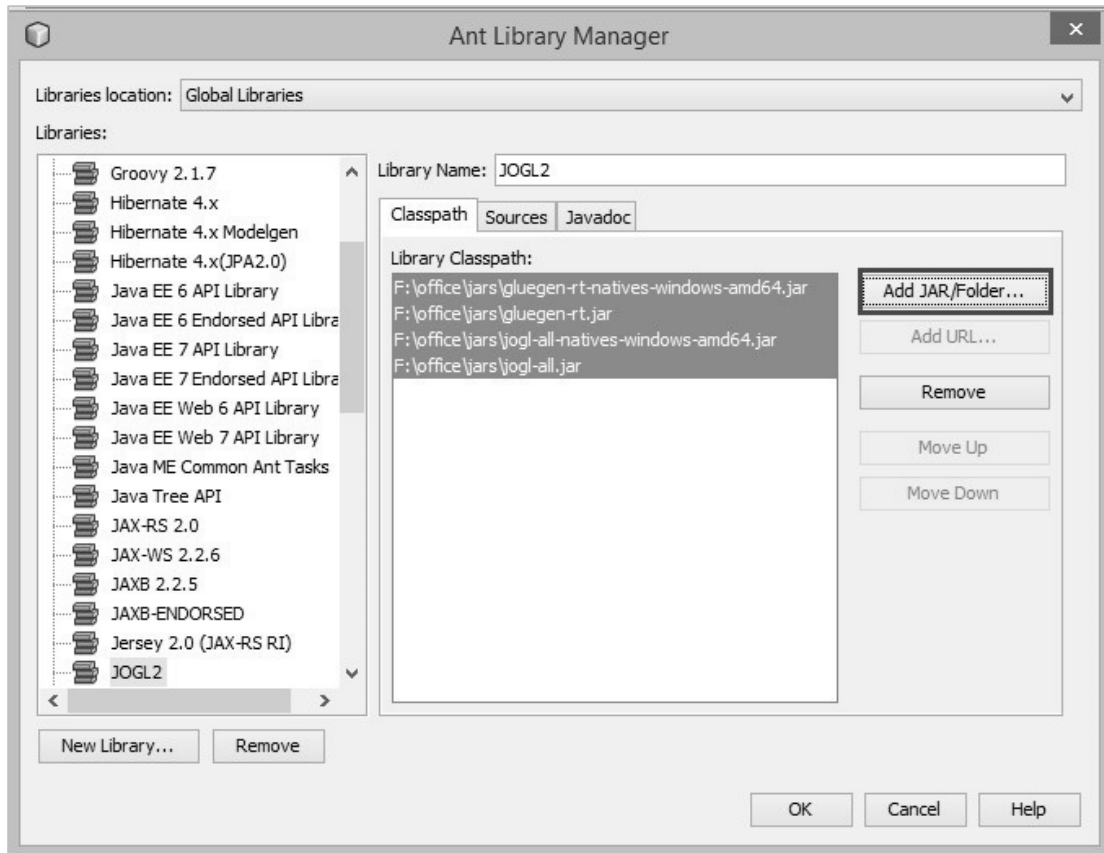


2. It leads you to Ant Library Manager.



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

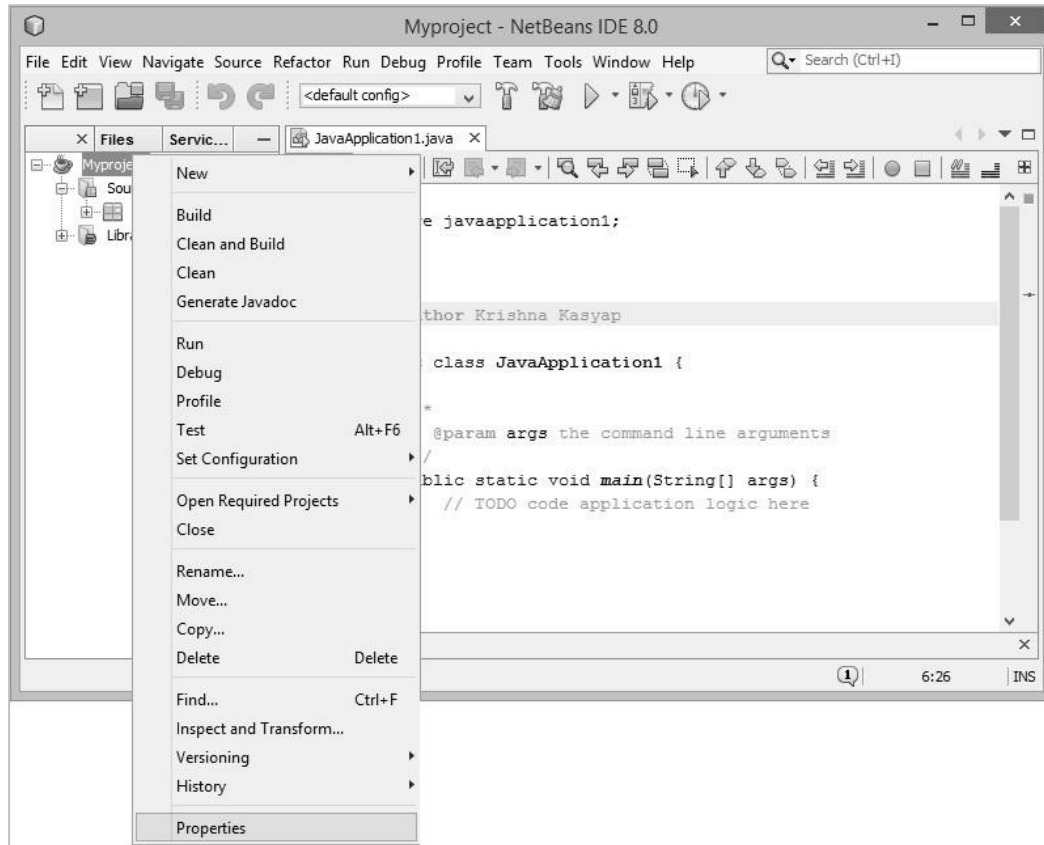
4. Enter Library name as **JoGI2.0**.
5. Click on "OK" button.



6. Click on "Add JAR/Folder..." button.
7. Select the path where .jar files **jogl.all.jar** and **gluegen-rt.jar** are located.

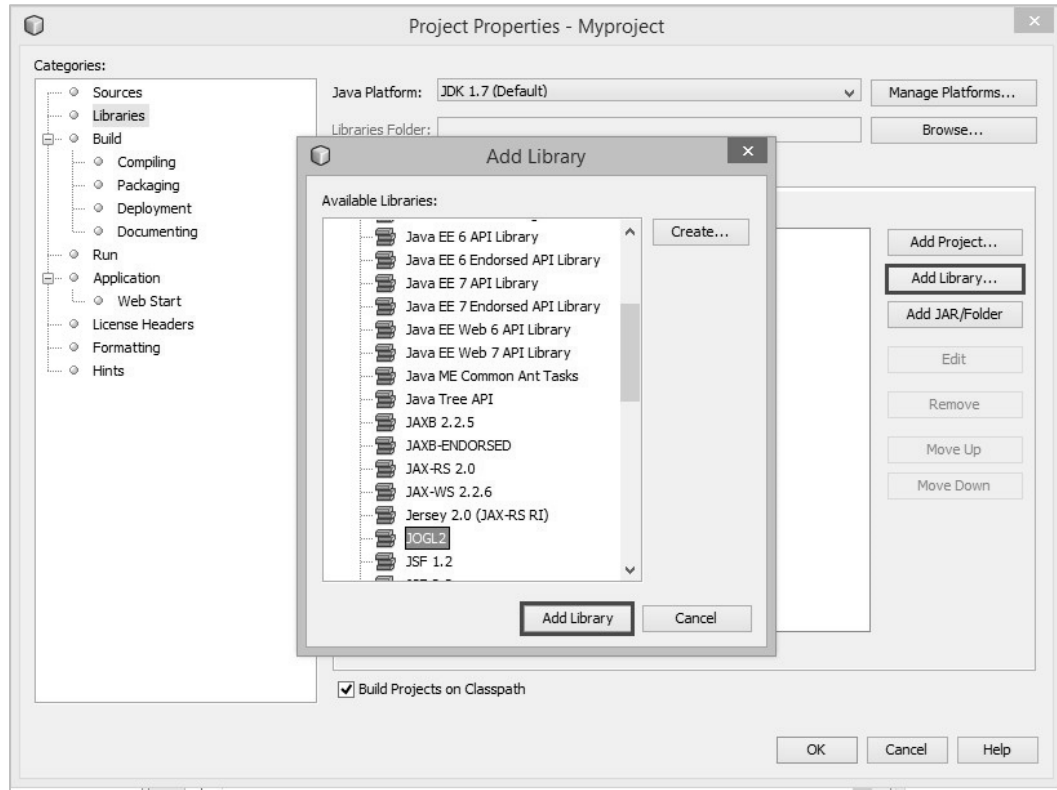
To include JOGL library into each project, follow the steps given below:

1. Right-click on the **project name**. It shows a short-cut menu.



2. Select **Properties**. It opens a window named **Project properties**.

Java Binding for OpenGL (JOGL)



End of ebook preview
If you liked what you saw...
Buy it from our store @ <https://store.tutorialspoint.com>