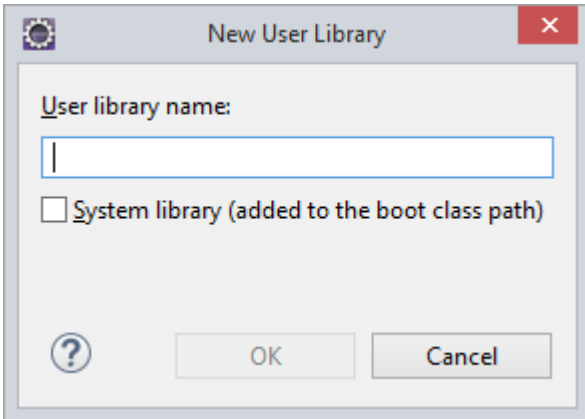


Hier unten ist das einfachste Verfahren, um JOGL in Eclipse einzubinden (**2 kurzen Schritte**) und anschließend ein OpenGL-fähiges Fenster zu erstellen (**5 kurzen Schritte**) – der unten gegebene Code dient als eine Vorlage, die anschließend mit weiteren Einzelheiten ergänzt wird:

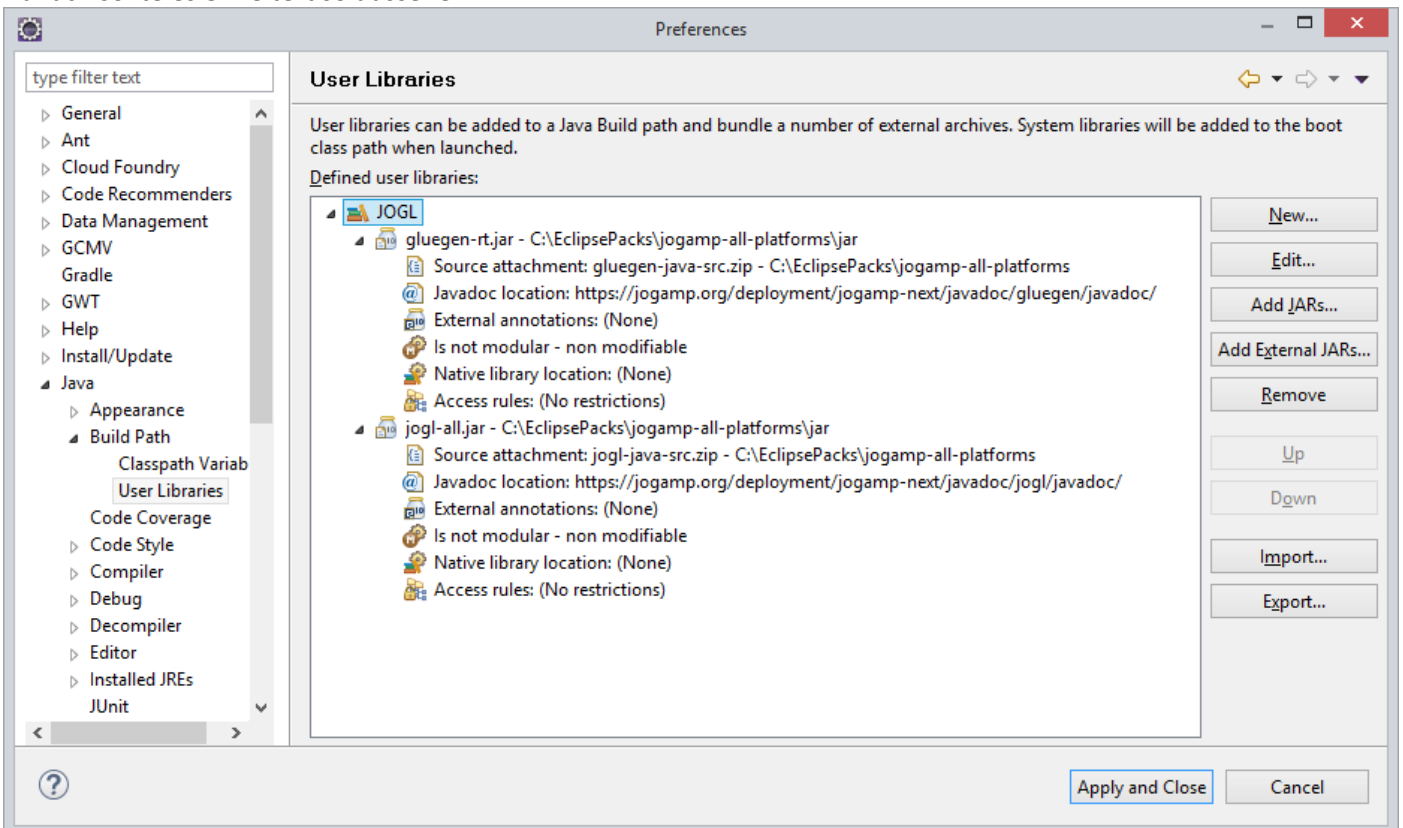
-2. Unter „Windows -> Preferences -> Java -> Java Build Path -> User Libraries“ auf „New“ klicken - es erscheint folgender Dialog:



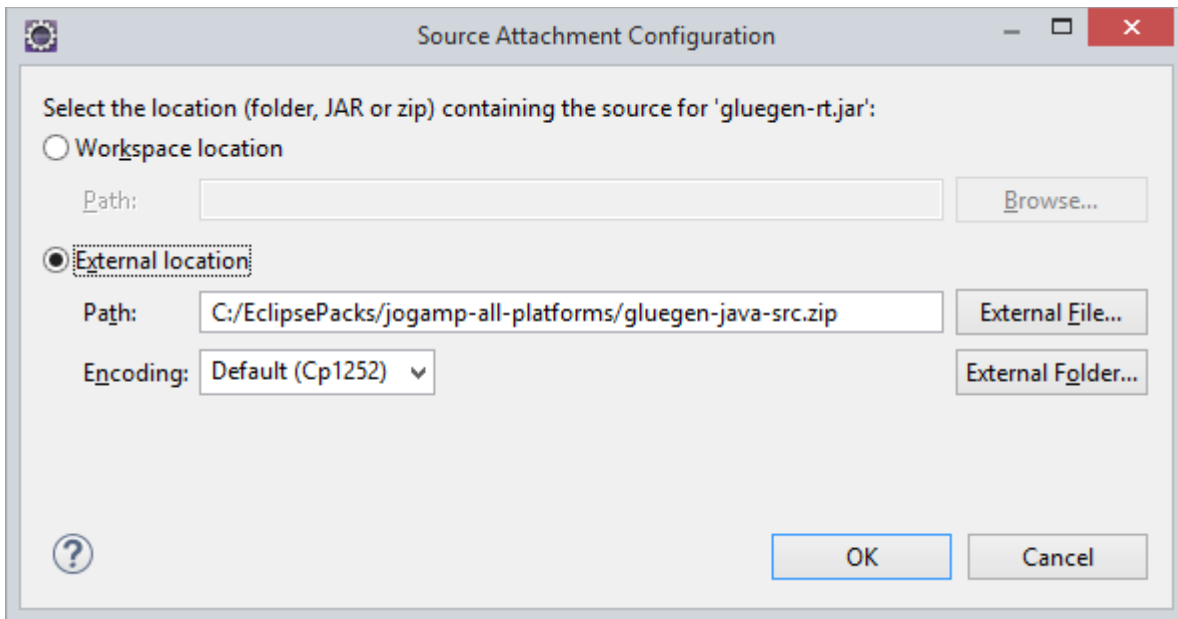
Geben Sie im Textfeld einen gewünschten Namen für die neue Bibliothek an. Zu empfehlen wäre „JOGL“. Dann klicken Sie auf „OK“.

Wählen Sie „JOGL“ und klicken Sie auf „Add External JARs...“ – dann finden Sie den entpackten Ordner mit bereits heruntergeladenen JOGL-Bibliotheken und fügen Sie die Dateien „gluegen-rt.jar“ und „jogl.all.jar“ aus dem „jogamp-all-platforms\jar“-Ordner hinzu.

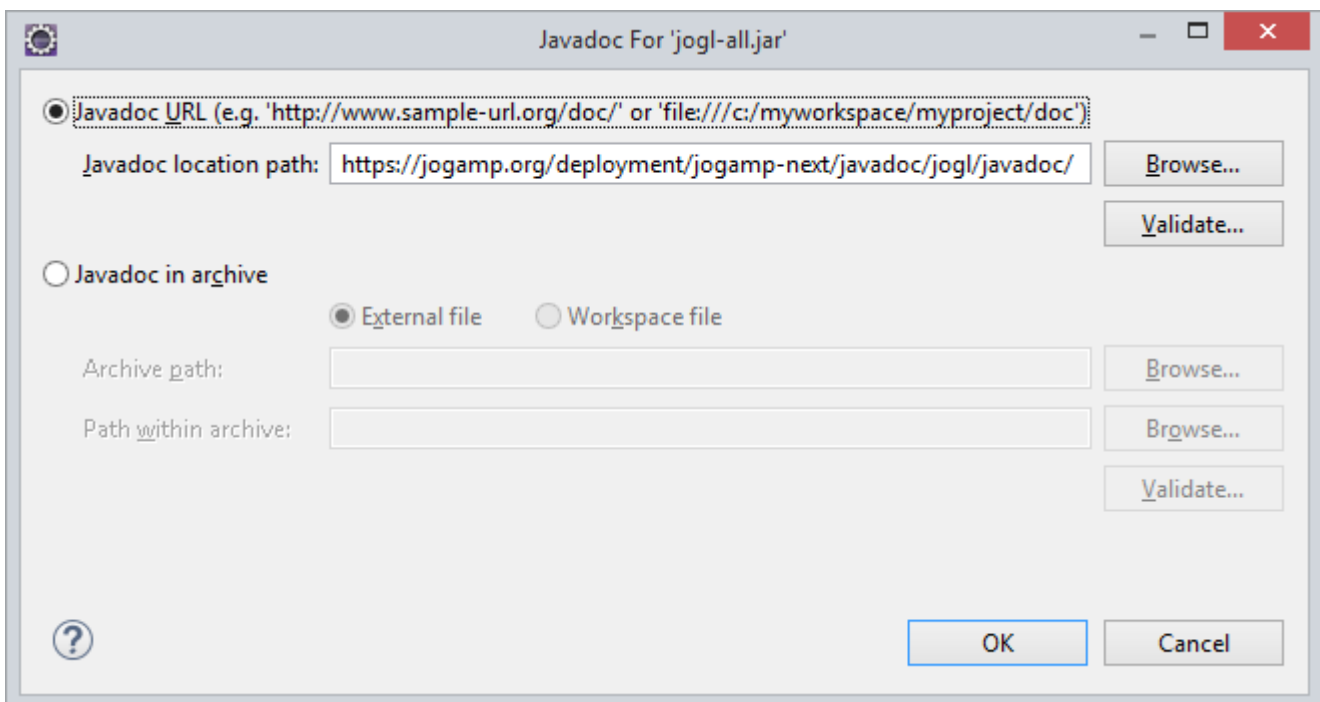
Danach sollte es erweitert so aussehen:



-1. Nun müssen nur noch „Source attachment:“ und „Javadoc location:“ jeweils angegeben werden. Wählen Sie für „gluegen-rt.jar“ als „Source attachment:“ die Datei „gluegen-java-src.zip“ aus dem „jogamp-all-platforms“-Ordner und für „jogl.all.jar“ als „Source attachment:“ die Datei „jogl-java-src.zip“ welche sich ebenfalls in „jogamp-all-platforms“ befindet:

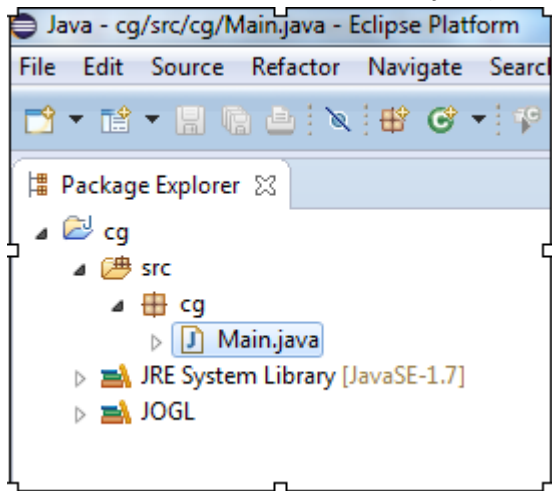


Wählen sie nun noch (für beide Einträge) die entsprechende „Javadoc location“:



URLs bekommt ihr unter dem „Builds / Downloads“-Menüpunkt [auf der offiziellen Webseite](#) von JOGL (kopiert einfach die Links unter „API Docs“ jeweils von GlueGen und JOGL). Anschließend „OK“ klicken.

1. Nachdem ihr ein neues Java-Projekt (mit JOGL-Unterstützung) erstellt, wird euer Projekt wie folgt aussehen:



2. Leert ihr den Inhalt von der "Main" Klasse und fügt den folgenden Code ein:

```
package cg;

import javax.swing.JFrame;

//import com.jogamp.opengl.GL2;
import com.jogamp.opengl.GLAutoDrawable;
import com.jogamp.opengl.GLCapabilities;
import com.jogamp.opengl.GLEventListener;
import com.jogamp.opengl.GLProfile;
import com.jogamp.opengl.awt.GLCanvas;

public class Main implements GLEventListener {

    @Override
    public void dispose(GLAutoDrawable arg0) {
        //body
    }

    @Override
    public void reshape(GLAutoDrawable arg0, int arg1, int arg2, int arg3, int arg4) {
        //body
    }

    @Override
    public void init(GLAutoDrawable arg0) {
        //body
    }

    @Override
    public void display(GLAutoDrawable arg0) {
        //body
    }

    public static void main(String[] args) {

        //instantiating OpenGL functionality
        final GLProfile profile = GLProfile.get(GLProfile.GL2);
        GLCapabilities capabilities = new GLCapabilities(profile);

        //creating frame
        final GLCanvas glcanvas = new GLCanvas(capabilities);

        //setting frame size inputs
        final int width = 800;
        final int height = 600;

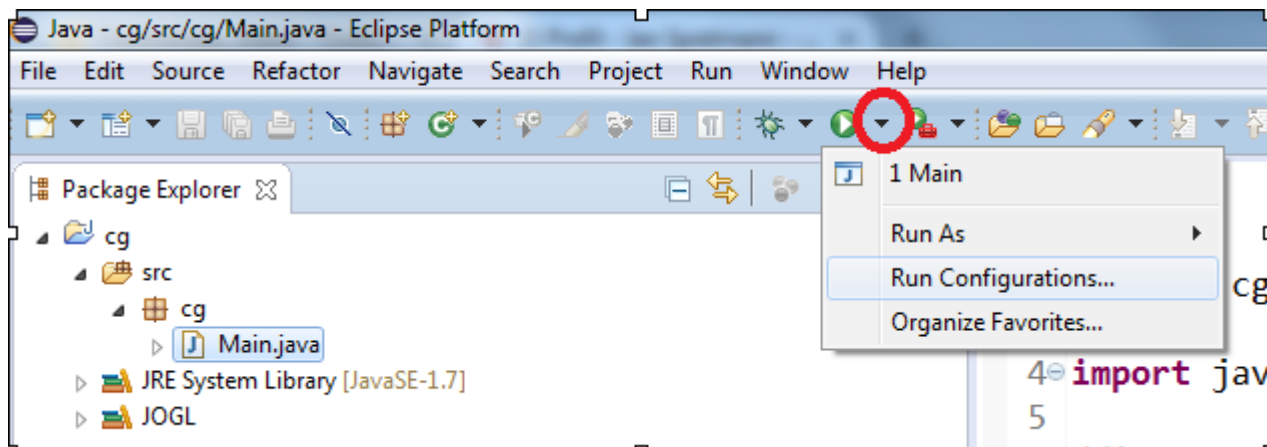
        Main m = new Main();
        glcanvas.addGLEventListener(m);

        //creating frame
        final JFrame frame = new JFrame ("OpenGL Fenster");

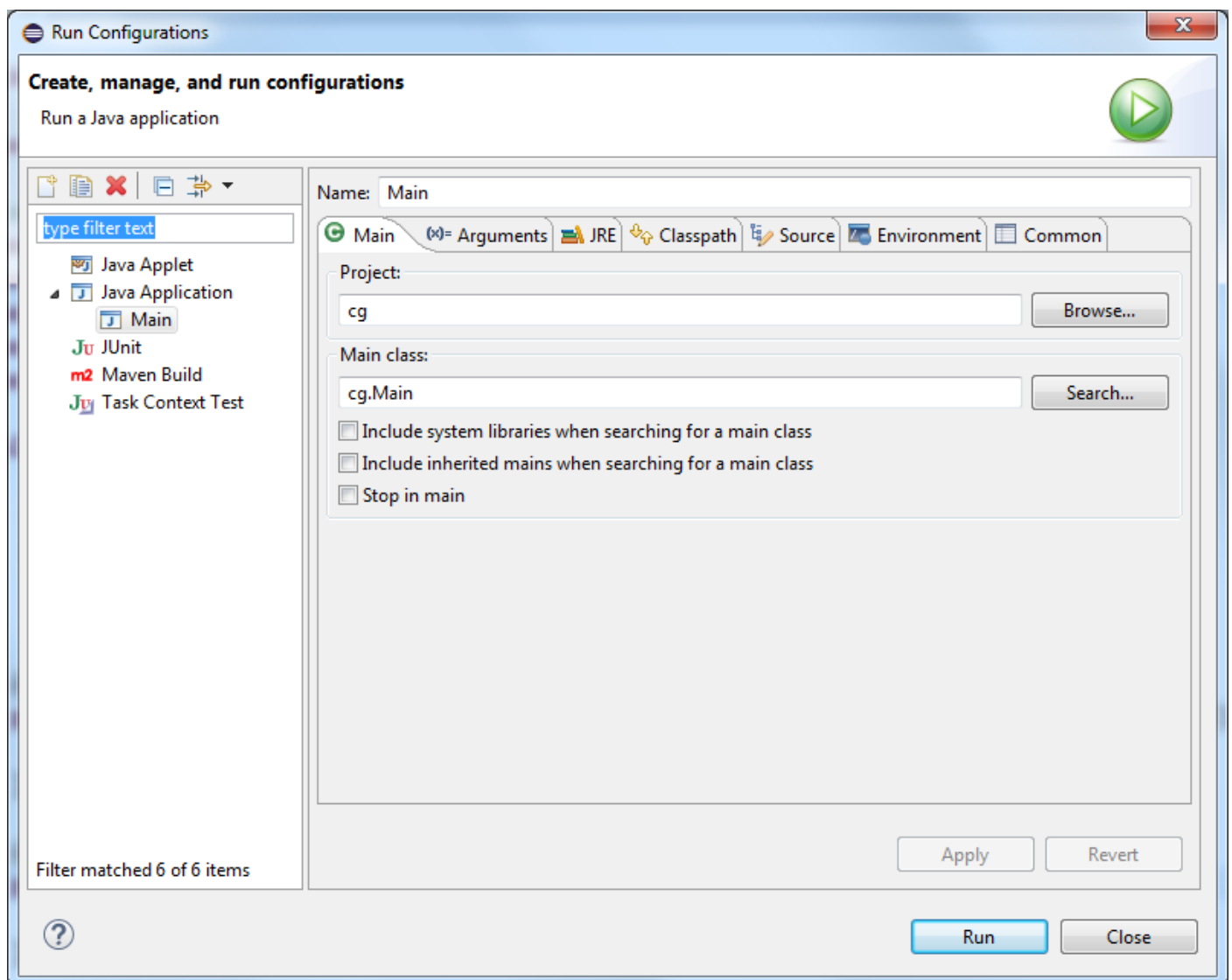
        //adding canvas to frame
        frame.getContentPane().add(glcanvas);

        //setting visibility and sizes to frame:
        frame.setSize(width, height);
        frame.setVisible(true);
    }
}
```

3. Drückt ihr auf dem nach unten zeigenden Pfeil rechts von der "Run"-Taste und wählt "Run Configurations..." aus:



4. Stellt sicher, dass das aufgepoppte Fenster so (wie unten) aussieht und drückt anschließend auf "Run":



5. Das schwarze Fenster mit dem Titel "OpenGL Fenster" wird erstellt und durchs Drücken auf "X" geschlossen:

