

Peter Mileff PhD

# Programming of Graphics

**Introduction to LWJGL**

University of Miskolc  
Department of Information Technology

# LWJGL Overview

## ● Lightweight Java Game Library

<https://www.lwjgl.org>

## ● What is LWJGL?

- Java library that enables cross-platform access to popular native APIs. Useful
  - for graphics (OpenGL)
  - for audio (OpenAL)
  - for parallel computing (OpenCL)

● It is a high performance, user-friendly wrapper library

● It is designed to create 2D and 3D graphics applications coded in Java

# LWJGL Overview

- LWJGL is an enabling technology and provides low-level access.
- It is cross platform
- **It is not a framework and does not provide higher-level utilities than what the native libraries expose**
  - **It is not a real game engine**
- **LWJGL is open source**

# Installation

## ● System Requirements


- Java SE Development Kit: version 1.8 or above
- Video Card that supports at least OpenGL version 1.1

## ● Configure Environment


- **PATH** and **JAVA\_HOME** environment variables should be set:
  - PATH: typically java\_install\_dir/bin
    - E.g.: C:\Program Files\Java\jdk1.8.0\_144\bin\
  - JAVA\_HOME: java\_install\_dir
    - E.g.: c:\Program Files\Java\jdk1.8.0\_144

## Download LWJGL 3

All official releases are available for download from GitHub:

 [Download from Github](#)

You can customize your LWJGL build or use the latest stable & nightly releases:

 [Customize LWJGL 3](#)

Or you can browse and download individual LWJGL artifacts:

 [Browse LWJGL files](#)

# LWJGL modules

- LWJGL is organized in modules.
- Each module is packaged as a collection of artifacts
  - where each artifact has the same name but a different classifier.
- The artifact name for the LWJGL core module is just **lwjgl**.
- The artifact name for a binding module is  
**lwjgl-<binding>**
- For example the OpenGL module is  
**Lwjgl-opengl**
- You only need to download the modules that are needed for your project.

## Release

Latest official release

3.1.3 build 16

Sep 21, 2017, 23:37:42 UTC

## Stable

Beta quality, verified to work

3.1.4 SNAPSHOT build 1

Sep 26, 2017, 19:10:21 UTC

## Nightly

Bleeding edge, possibly broken

3.1.4 SNAPSHOT build 4

Oct 4, 2017, 09:05:59 UTC




### Mode

- ZIP Bundle
- Maven
- Gradle
- Ivy

### Options

- Show descriptions
- Include source
- Include JavaDoc

### Natives

-  Windows
-  macOS
-  Linux

### Presets

- None
- Custom
- Everything
- Getting Started
- Minimal OpenGL
- Minimal OpenGL ES
- Minimal Vulkan

### Addons

- JOML v1.9.4
- LWJGLX/debug v1.0.0
- steamworks4j v1.6.2

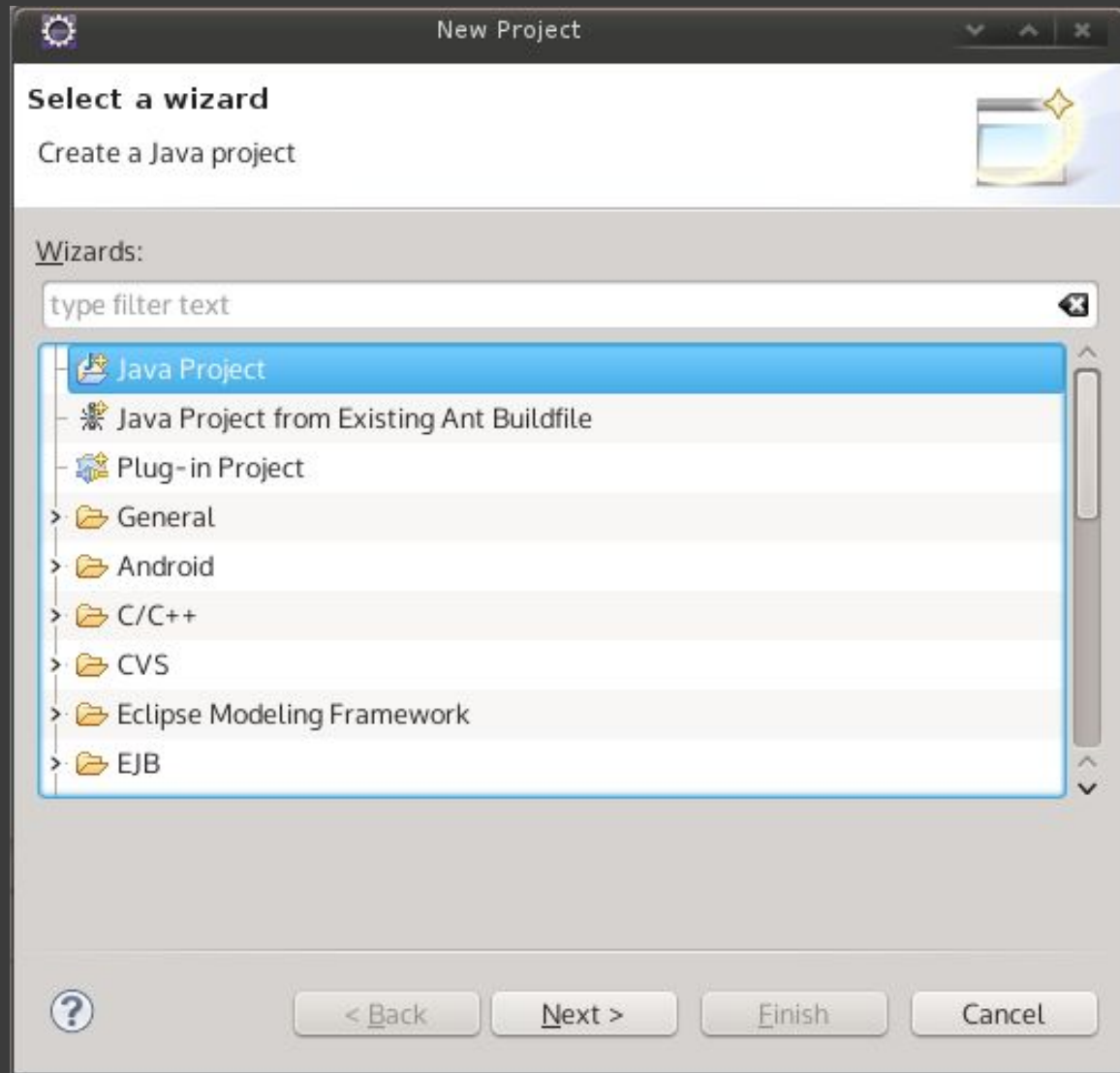
### Contents

- LWJGL core
- Assimp bindings
- bgfx bindings
- EGL bindings
- GLFW bindings
- JAWT bindings
- jemalloc bindings
- LMDB bindings
- NanoVG & NanoSVG bindings
- Native File Dialog bindings
- Nuklear bindings
- OpenAL bindings
- OpenCL bindings
- OpenGL bindings
- OpenGL ES bindings
- OpenVR bindings
- OVR bindings
- par\_shapes bindings
- rpmalloc bindings
- SSE bindings
- stb bindings
- Tiny OpenEXR bindings
- Tiny File Dialogs bindings
- Vulkan bindings
- xxHash bindings
- Yoga bindings

# Setup LWJGL in Eclipse...



# Create New Project



# Create New Project

New Java Project

**Create a Java Project**

Create a Java project in the workspace or in an external location.

Project name:

Use default location

Location:

JRE

Use an execution environment JRE:

Use a project specific JRE:

Use default JRE (currently 'jre1.8.0\_144') [Configure JREs...](#)

Project layout

Use project folder as root for sources and class files

Create separate folders for sources and class files [Configure default...](#)

Working sets

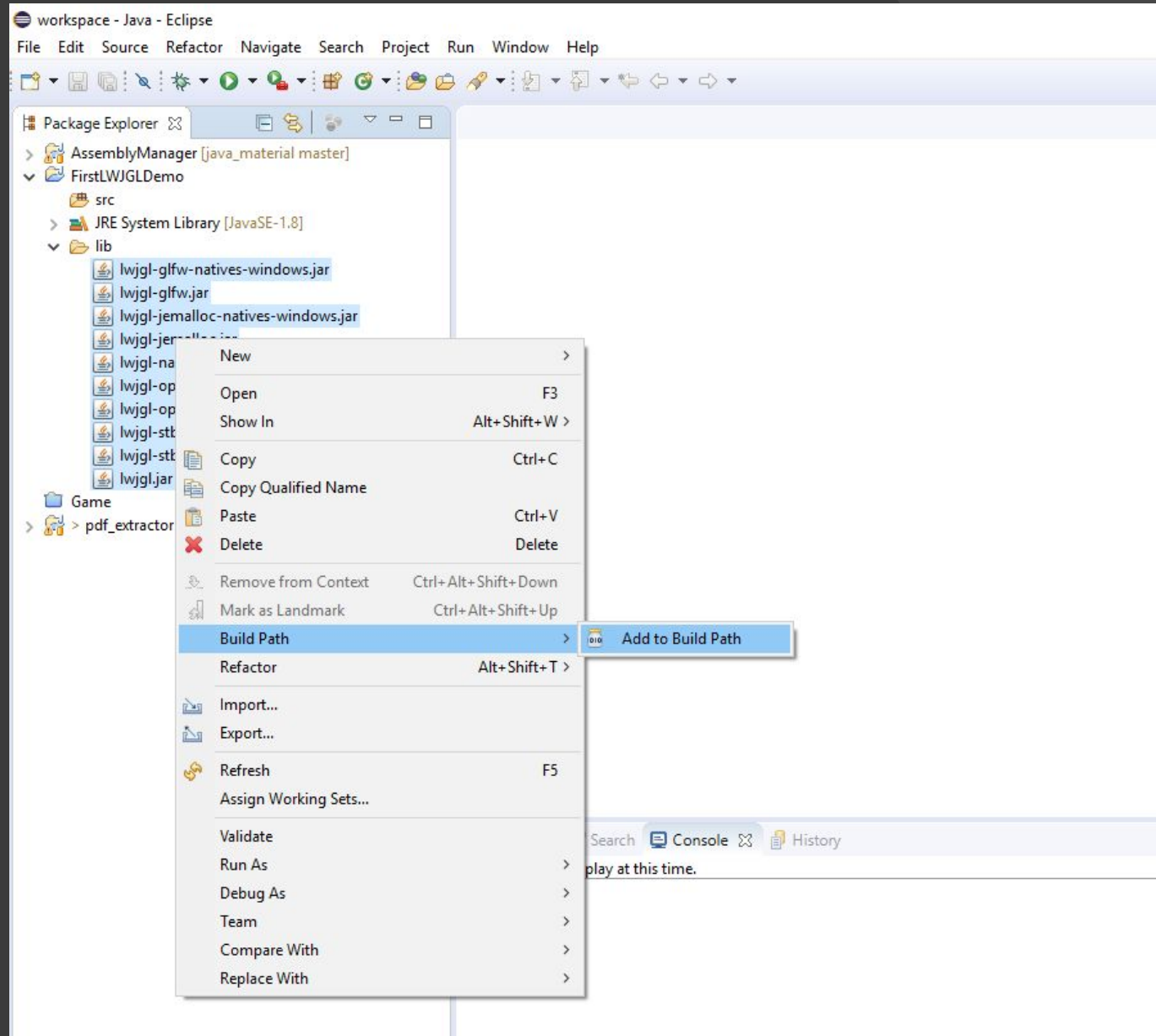
Add project to working sets

Working sets:

# LWJGL external libraries

- Create a new folder named **lib** in the project folder
- Copy the (proper) files into the lib folder:
  - Example in case of Windows 64 bit:
    - lwjgl-glfw.jar
    - lwjgl-jemalloc-natives-windows.jar
    - lwjgl-jemalloc.jar
    - lwjgl-natives-windows.jar
    - lwjgl-opengl-natives-windows.jar
    - lwjgl-opengl.jar
    - lwjgl-stb-natives-windows.jar
    - lwjgl-stb.jar
    - lwjgl.jar

# Add libraries to build path



# Jogl in Eclipse

- **To make all .jar files available to other projects**
  - Go to main menu, select Window > Preferences
- **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. E.g.: LWJGL3
  - Add jar files using button “Add External JARs...”.
  - It creates a new user library named LWJGL3

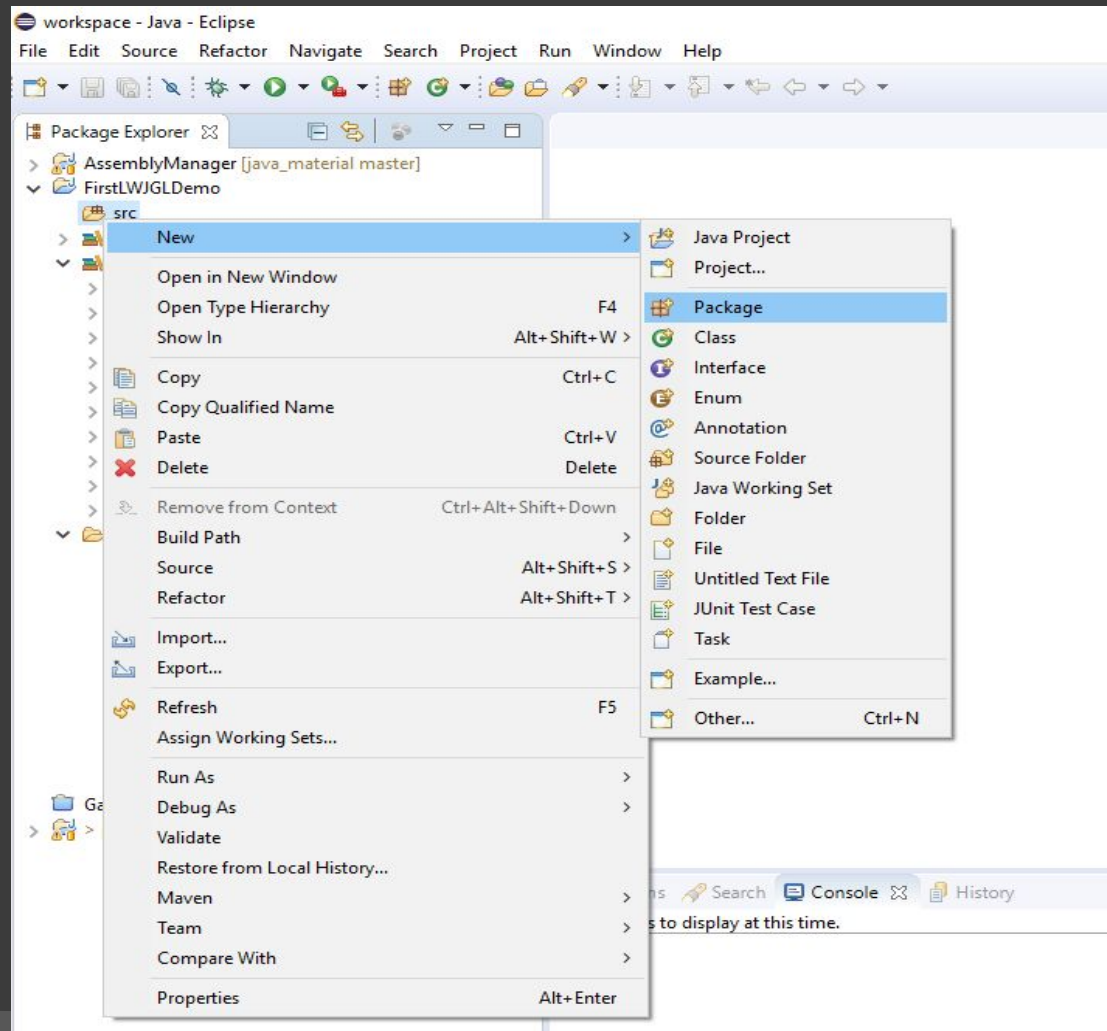
# LWJGL in Eclipse

- **We can start developing OpenGL based applications !**

**Create a Hello World LWJGL  
application...**

# Step 1

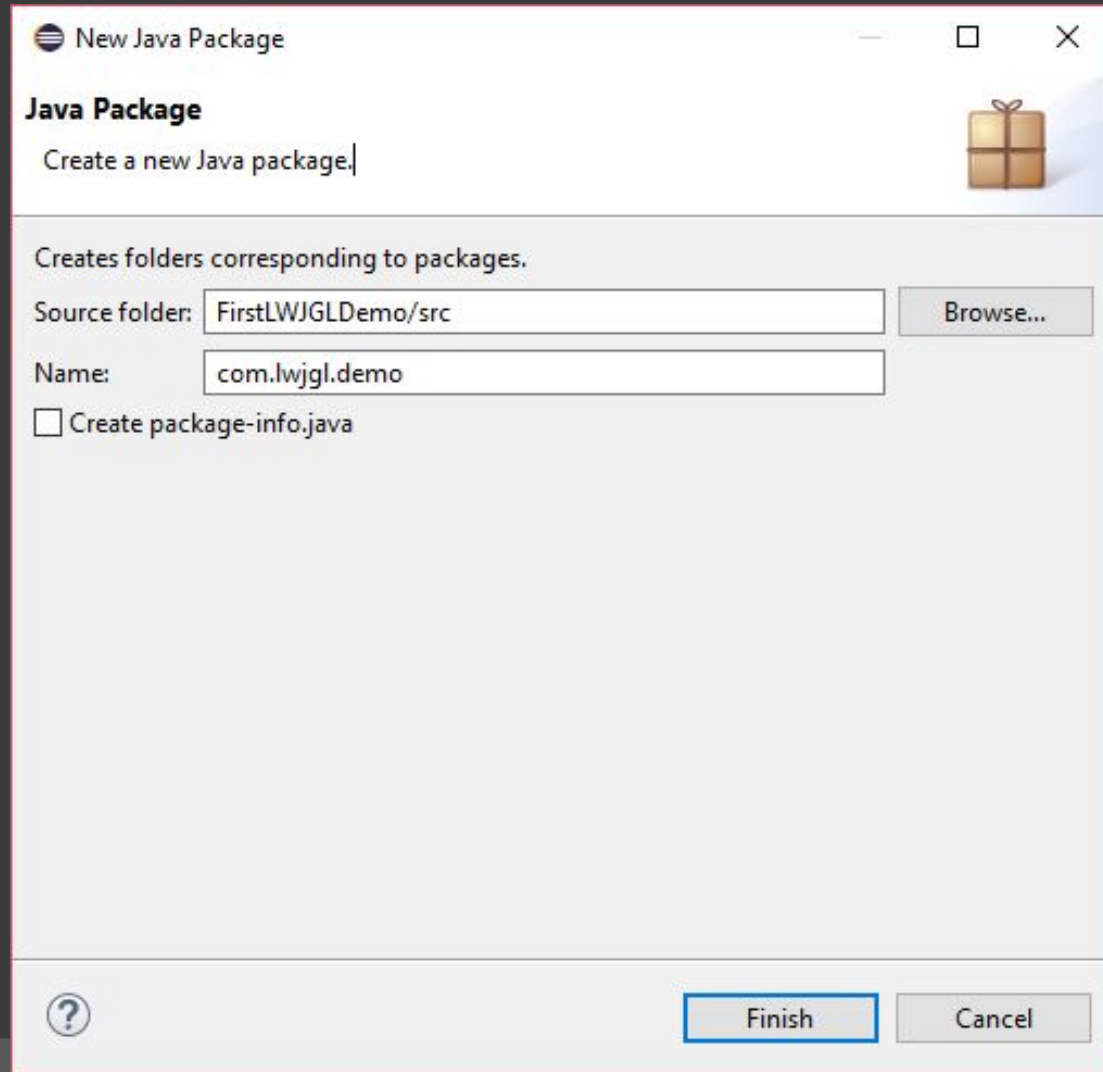
## 1) Create a new package





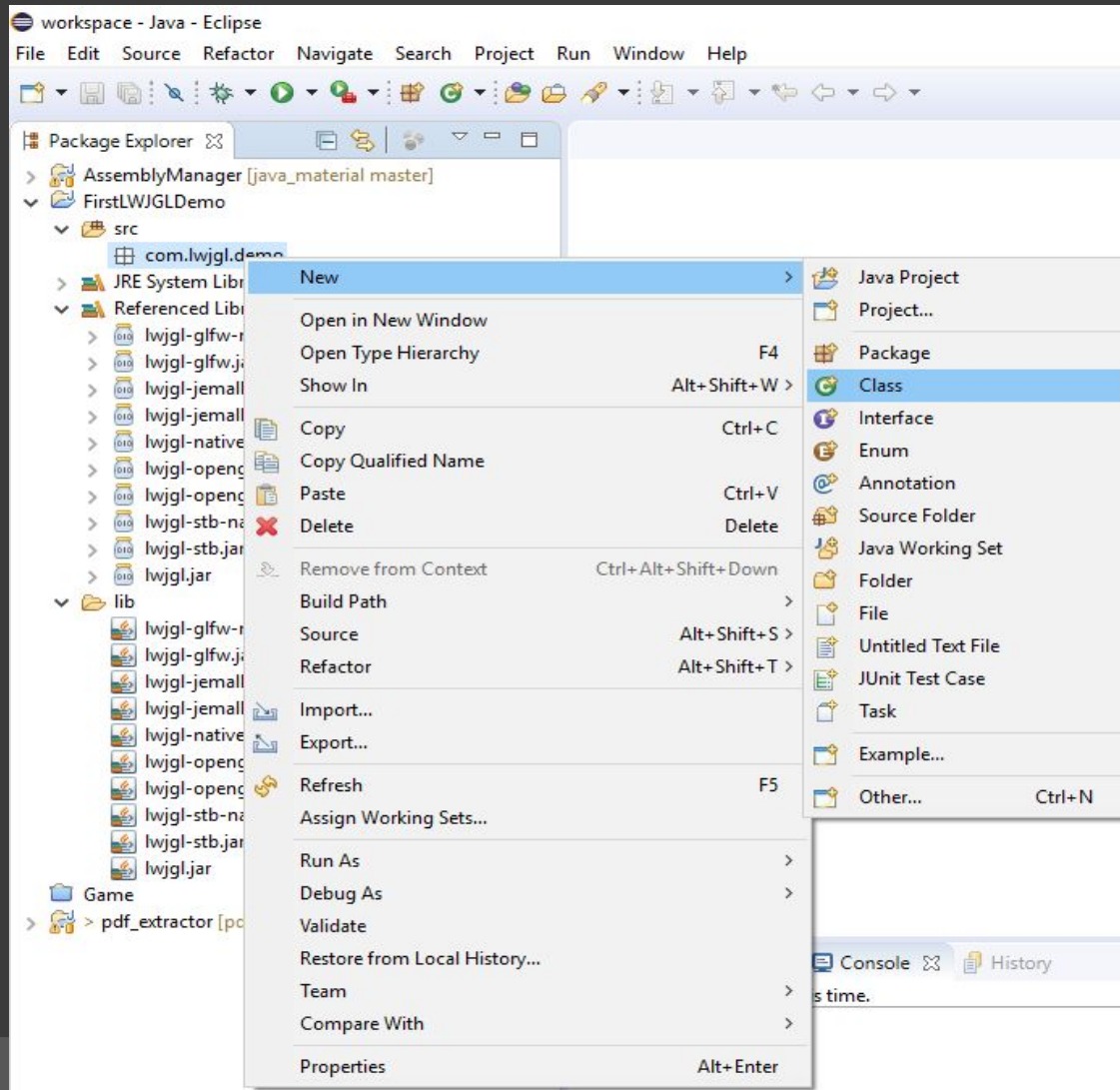
# Step 2

- E.g.: com.lwjgl.demo



# Step 3

- Create a new class



# Step 4

- E.g.: LWJGLHelloWorld

**New Java Class**

**Java Class**  
Create a new Java class.

Source folder:

Package:

Enclosing type:

Name:

Modifiers:  public  package  private  protected  
 abstract  final  static

Superclass:

Interfaces:

Which method stubs would you like to create?

public static void main(String[] args)  
 Constructors from superclass  
 Inherited abstract methods

Do you want to add comments? (Configure templates and default value [here](#))  
 Generate comments

# Step 5

- **Get sample your first sample code from LWJGL website:**

<https://www.lwjgl.org/guide>

- **Copy the code into your class, replace class name and run**

GAME OVER