LAB 3 (PART 1)

Objetive

* Configure working environment with Eclipse and Java, Lightweight Java Game Library (LWJGL)[[1]](#footnote-0);
* Create a simple windows;
* Create simple geometric figure with a list of data.

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

Install and configure working environment

**Folder Download ->**

1. Eclipce;
2. Libraryes: lwjgl-2.9.1, slick-util,

**Ajustments:**

* **Install** Eclipse - with the basic configurations;
* **Create** a folder with name **lib;**
* Inside of **lib** create 2 more folders : **jar** and **native**
* **Extract lwjgl-2.9.1** libraries , from ..**\lwjgl-2.9.1\jar**,
* **Copy** : **lwjgl**, **slick-util** and **lwjgl\_util** in folder **..\lib\jar**.
* From folder ..**\lwjgl-2.9.1\native**, **copy** the files that corresponds with your operating system(in our case windows) to folder .**.\lib\native**.

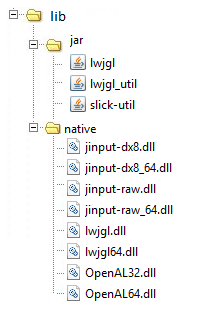


Figure1

* Start Eclipse and create new project: - with name **GameEngine**;
* Open the folder where the project is located (..\eclipse-workspace\ OpenGL\_JavaProject\GameEngine) and copy there the **lib** folder you created before;

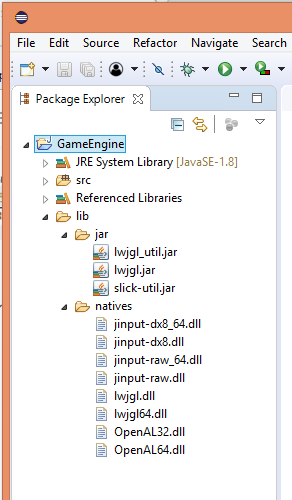


Figure 2

* Add the libraries to the project: BildPath -> Configure Bild Path… -> Libraries -> Add JARs…

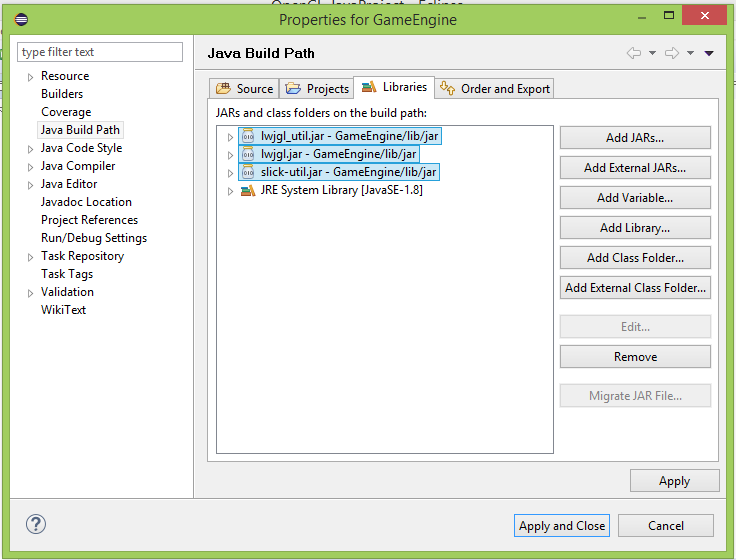


Figure 3

* Add the Native Library: JRE System Library -> Native Library location -> Edit -> Workspace… -> GameEngine/lib/natives.
* Save all changes.

Creating a simple windows

* Create a new package: renderEngine

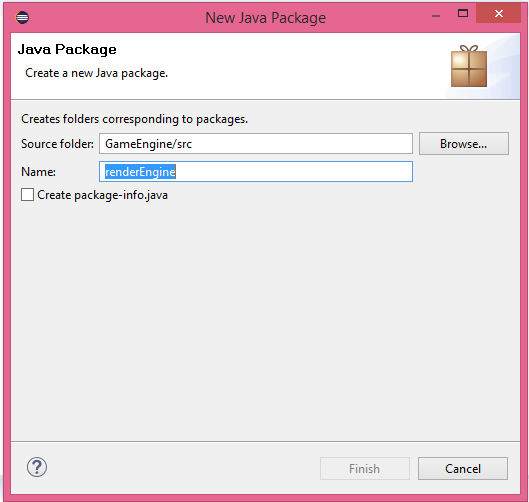


Figure 4

* Create a new class: DisplayManager
* Thes class is going to control the display, we'll create three methods for that:
* **public** **static** **void** createDisplay(){};
* **public** **static** **void** updateDisplay(){};
* **public** **static** **void** closeDisplay(){};
* The method createDisplay():

Display.*setDisplayMode*(**new** DisplayMode(***WIDTH***,***HEIGHT***));

* The parameters for its height and width will be set as constants;

**private** **static** **final** **int** ***WIDTH*** = 1280;

**private** **static** **final** **int** ***HEIGHT*** = 720;

* The function receive like paramethers the shape of pixels and some attributes. The attributes contains the version of used OpenGl libraries - in this case 3.2:

ContextAttribs attribs = **new** ContextAttribs(3,2)

.withForwardCompatible(**true**)

.withProfileCore(**true**);

Display.*create*(**new** PixelFormat(), attribs);

* To set the name of the windows use the function:

Display.*setTitle*("OpenGl First Display");

* To indicate OpenGL where to open the windows, and which part of all world will be dispayed we use the function glViewport. Parameters of the Viewport are: lower left corner and width and the height of the viewport until desired top right corner - in this case all over the windows:

GL11.*glViewport*(0, 0, ***WIDTH***, ***HEIGHT***);

* To close the windows, use destroy() function:

Display.*destroy*();

* To update the window on each frame, we have to synchronize first with Display.sync(frequency) and then actualize:

Display.*sync*(***FPS\_CAP***); //frame frequency

Display.*update*();

* To test the windows, create a package engineTester with main class MainGameLoop.
* This MainGameLoop class contains the main function and a cycle that works until the window is closed

**package** engineTester;

**import** org.lwjgl.opengl.Display;

**import** renderEngine.DisplayManager;

**public** **class** MainGameLoop {

**public** **static** **void** main(String[] args) {

DisplayManager.*createDisplay*();

**while**(!Display.*isCloseRequested*()) {

// game logic

DisplayManager.*updateDisplay*();

}

DisplayManager.*closeDisplay*();

}

}

1. <https://www.lwjgl.org/> [↑](#footnote-ref-0)