Setup instructions for using LibGDX

Jeremy Swartwood

Created 02/23/2018

Last updated 02/28/2018

# Table of contents

[Table of contents 1](#_Toc507601798)

[I. Download, Install, AND CONFIGURE Java SDK: 1](#_Toc507601799)

[I.1 Download Java JDK 8: 1](#_Toc507601800)

[I.2 Install the JDK. 2](#_Toc507601801)

[II. Download, Install, AND CONFIGURE Git for windows: 2](#_Toc507601802)

[III. Download, install, and configure github for windows 2](#_Toc507601803)

[IV. Download, Install, AND CONFIGURE Android Studio: 2](#_Toc507601804)

[V. Download, Install, AND CONFIGURE LibGDX: 3](#_Toc507601805)

[Download Steps: 3](#_Toc507601806)

[Installation Steps: 3](#_Toc507601807)

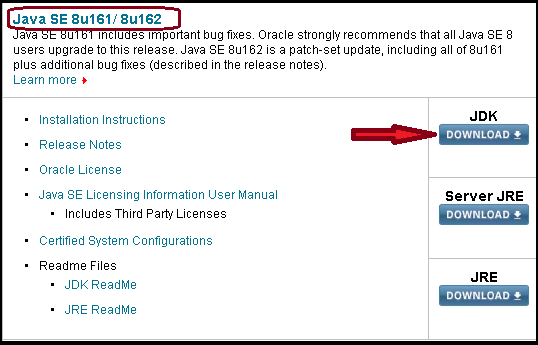
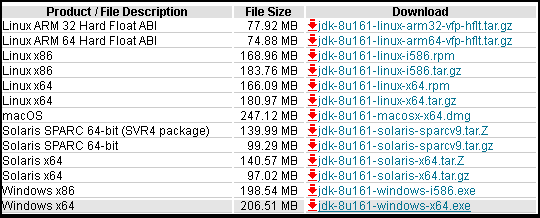
[Create empty LibGDX project: 3](#_Toc507601808)

[Setup the Desktop mode so you can run it locally: 5](#_Toc507601809)

[Errors and Resolutions: 7](#_Toc507601810)

# I. Download, Install, AND CONFIGURE Java SDK:

## I.1 Download Java JDK 8:

1. Go to the java downloads: <http://www.oracle.com/technetwork/java/javase/downloads/index.html>
2. Click Download  
   
3. Scroll down to 8u161 or 8u162 and click I accept  
   
4. Click on the link for the version for your system  
   

## I.2 Install the JDK.

# II. Download, Install, AND CONFIGURE Git for windows:

1. Download Git for Windows: <http://git-scm.com/download/win>
2. During installation, choose: Use Git bash only
3. Use OpenSSH (no need to configure other tools)
4. Checkout Windows-style, commit unix-style line endings. (Save from issues with platform specific repositories)

# III. Download, install, and configure github for windows

(This is **optional**, and only if you want a GUI to access github beyond git’s gui)

1. Download from here: <https://windows.github.com/>
2. **(Optional)** On GITHUB, you should setup an SSH key to access your data without using a password
   1. Generate an SSH key
      1. <https://help.github.com/articles/generating-ssh-keys/>
   2. Setup and use the SSH key
      1. Upper right of the webpage, click the COG to open settings.
      2. Click on the SSH Keys tab.
         1. Click Add SSH Key
         2. Enter a name, and paste the SSH key.
3. Install Github for Windows, and use the SSH key instead of a username and password.
   1. I just use my username and password because I’m lazy.

# IV. Download, Install, AND CONFIGURE Android Studio:

1. Download Android Studio Bundle for Windows: <http://developer.android.com/sdk/index.html#top>
2. Open and run the installer: android-studio-bundle\*.exe
   1. Select all components and click next
   2. Pick the appropriate drives/locations for files: Changed to D: drive
   3. Accept all other defaults
3. Open Android Studio:
   1. Select the JDK location (***NOTE:*** *Since JDK 8 is out, it will complain about 7 not being up to* date)
   2. Select Standard installation.
   3. Accept and next to update SDK/API/etc.
4. Add Packages:
   1. <http://developer.android.com/sdk/installing/adding-packages.html>
   2. Click Configure -> SDK Manager
      1. Tools:
         1. Andoird SDK Tools
         2. Android SDK Platform-Tools
         3. Android SDK Build-Tools
      2. Android 5.1 (Or latest version)
         1. I just left all checked.
      3. Any previous android version you want.
      4. Extras
         1. Android Support Repository
         2. Android Support Library
         3. Google USB Driver
         4. Google Repository
         5. Google Play services
5. Configure GIT settings:
   1. (NOTE: This may no longer be needed, I have not tested this on a fresh install)
   2. Configure -> Settings
   3. Version Control -> Git
   4. Enter the path to your installation of GIT
      1. D:\Program Files\Git\bin\git.exe
   5. Enter your Github Token

# V. Download, Install, AND CONFIGURE LibGDX:

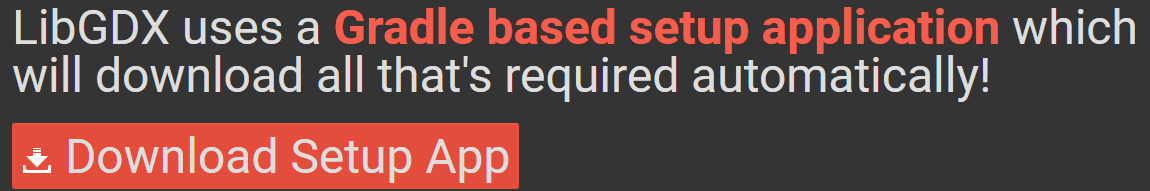
This is the framework that is similar to the old C# XNA Game Studio’s library.

* <http://libgdx.badlogicgames.com/documentation.html>

Someone else’s instructions:

* <https://github.com/libgdx/libgdx/wiki/Project-setup,-running-&-debugging>

## Download Steps:

1. Download LibGDX: <http://libgdx.badlogicgames.com/download.html>
   1. 
2. Unblock the file (if using windows)
   1. Right-click and chose properties
   2. Chose unblock, click apply then ok to close the properties.

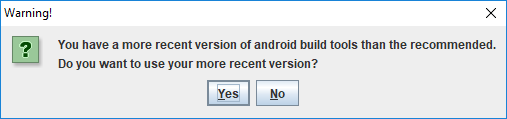
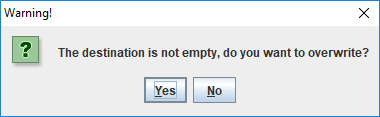
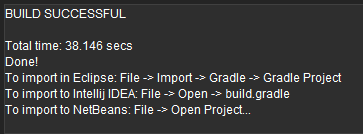
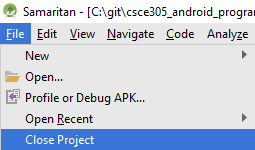
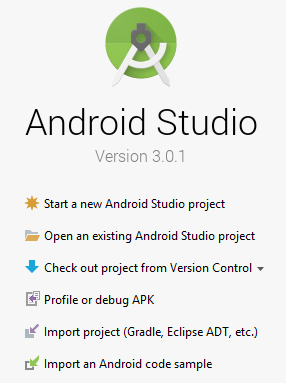
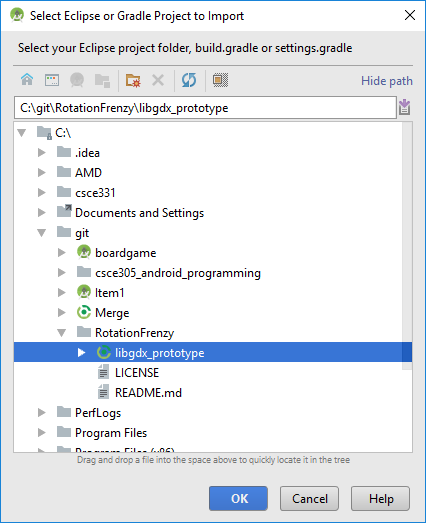
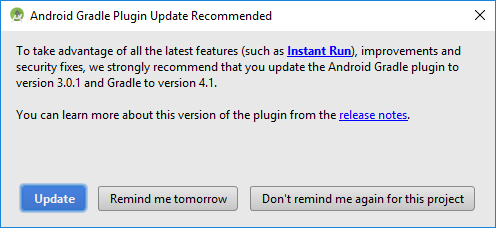
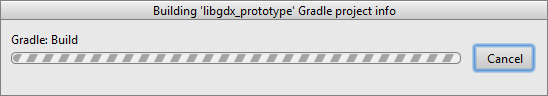
## Installation Steps:

Technically there are no installation steps because it is just a library that uses a build tool called Gradle to download the required library files. All the downloaded file does is creates an empty android studio project with gradle build options.

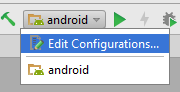
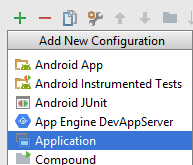
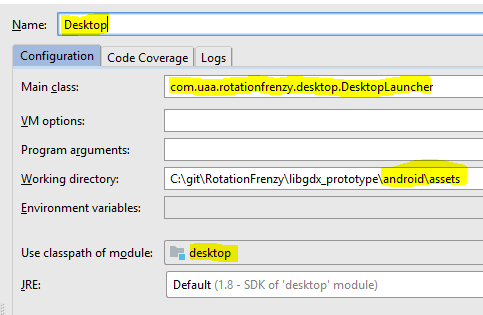
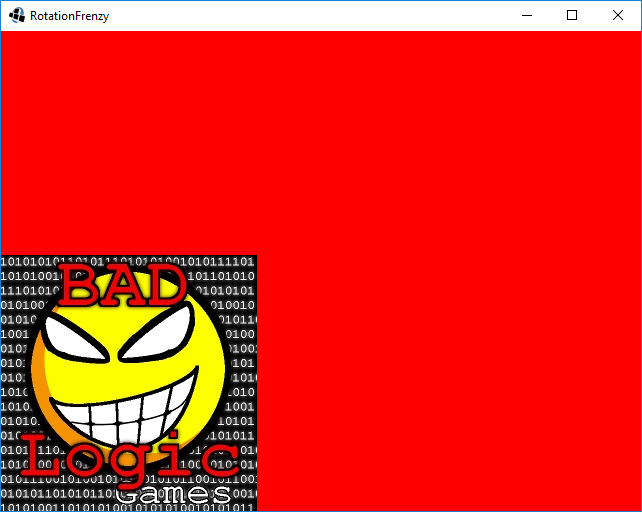
## Create empty LibGDX project:

1. Run the downloaded app (gdx-setup.jar)
2. Enter the following Information:

|  |  |
| --- | --- |
| **Name**: | Name for your game or project |
| **Package**: | Package name, generally com.company.application |
| **Game Class**: | Root class name, generally CamelCase |
| **Destination**: | Where do you want the project to be created? |
| **Android SDK**: | I had to open Android Studio and go to Settings and search for SDK to find my location. |
| **Sub** **Projects**: | Desktop, Android, iOS and Html are the defaults, change what you want. |
| **Extensions**: | Box2d is the default, change what you want to add/remove. |

1. Click **Generate**.
2. Possible warning:
   1. 
      1. You have a more recent version of build tools than is recommended. Do you want to use your more recent version?
      2. Click Yes.
   2. 
      1. The destination is not empty, do you want to overwrite?
      2. ***NOTE:*** *This doesn’t remove existing files, but if the same filenames exist, they will be overwritten.*
      3. Click Yes if you have nothing in the folder you care about.
3. Wait until you see Build Successful.  
   
4. You can now close the LibGDX project creation window.
5. Similar to what the success image says, here is where we switch to our IDE (Android Studio).
6. Open Android Studio.
7. If you have a current project open, close that project.  
   
8. Select Import project (Gradle, Eclipse, ADT, etc)  
   
9. Select the folder where you created the LibGDX project and click Ok.  
   
10. If you get the following message do NOT upgrade, click Don’t remind me again for this project. (See errors and resolutions for why)  
    
11. Wait for the Gradle Build process to complete and Android Studio to finish importing.  
    
12. Project should be open.

## Setup the Desktop mode so you can run it locally:

1. In the open Android Studio go to the Android button next to the Green Play button.
   1. Click the Down Arrow and then click Edit Configurations…  
      
2. Click the green + to add a new configuration, and select Application.  
   
3. Enter the following information:  
   
   1. **Name:** Desktop
   2. **Main Class:** Select the DesktopLauncher
   3. **Working Directory:** Add android\assets
   4. **Use classpath of module:** select the desktop folder.
4. **Apply** and **save** the information, and **close** the configuration window.
5. **Run** the application by selecting it from the Run drop down list.
6. You should get the following screen from a fresh install:  
   

# How to deploy to HTML:

<https://github.com/libgdx/libgdx/wiki/Deploying-your-application#deploy-to-htmljs>

# Errors and Resolutions:

|  |  |
| --- | --- |
| **Error** | A problem occurred evaluating project ‘:html’  > Plugin with id ‘jetty’ not found. |
| **Image** |  |
| **Description** | When trying to build the project, the following error is displayed. |
| **Cause** | Apparently Grade 4.1 no longer supports jetty |
| **Resolution** | <https://stackoverflow.com/questions/46975883/error2-0-plugin-with-id-jetty-not-found>   1. Downgrade Gradle to 3.3 from 4.1   Find gradle folder inside your project, Open gradle-wrapper.properties and change distributionUrl for 3.3  distributionUrl=https\://services.gradle.org/distributions/gradle-3.3-all.zip   1. Downgrade Android-gradle-plugin to 2.3.3 from 3.0.0   Open root build.gradle file and find artifact and change version  classpath 'com.android.tools.build:gradle:2.3.3'   1. Comment/delete google() from repo list 2. Build project again 3. DO NOT upgrade gradle after the build when prompted, click Don’t remind me. |

|  |  |
| --- | --- |
| **Error** | Couldn’t load file: badlogic.jpg |
| **Image** |  |
| **Description** | When running the desktop mode, the above error appears indicating that LWJGL could not load the file badlogic.jpg |
| **Cause** | The desktop run configuration is missing the assets folder location |
| **Resolution** | Open the configurations and add the assets folder. See the section [above](#_Setup_the_Desktop) on how to setup desktop mode. |