

# Jak na PGRF

## - návod na používání Git, OpenGL, OpenCV a vytváření JAR v předmětech PGRF na FIM.UHK.CZ

Verze 20210921, uloženo 21.09.2021

### Obsah

Obsah.....	1
A. Poznámka.....	1
B. Vytvoření projektu z klonů na GitLab pomocí Eclipse/TortoiseGit.....	1
C. Návod na instalaci knihovny LWJGL v prostředí Eclipse .....	9
D. Návod na instalaci LWJGL a zprovoznění ukázek v prostředí IntelliJ IDEA .....	18
E. Návod na vytvoření spustitelného JAR souboru v prostředí IntelliJ IDEA při použití knihovny LWJGL .....	23
F. Návod na instalaci OpenCV a zprovoznění ukázek v prostředí IntelliJ IDEA.....	27
G. Užitečné odkazy .....	29

### A. Poznámka

Následující návod vznikl na základě požadavků a potřeb výuky v předmětech PGRF na FIM UHK. Jednotlivé kapitoly vznikaly a byly aktualizovány postupně, a proto nejsou všechna pojmenování projektů, balíčků, knihoven a dalších souborů vždy konzistentní. Rozdíly v pojmenování jsou především na obrázcích, které nejsou vždy znovu nasnímány pro novou verzi knihoven zapouzdřujících OpenGL či OpenCV. Některá označení se proto mohou v jednotlivých kapitolách rozcházet, což by nemělo být omezující pro pochopení popsaných postupů.

### B. Vytvoření projektu z klonů na GitLab pomocí Eclipse/TortoiseGit

Následující návod popisuje způsob vytvoření projektu v Eclipse pro potřeby předmětu PGRF na FIM UHK. Projekt využívá zdrojů z repozitářů na GitLab. Je třeba mít nainstalovány programy Git (<https://git-scm.com/>) a volitelně TortoiseGit (<https://tortoisegit.org/>).

Při použití prostředí IntelliJ IDEA lze provést klonování při vytváření projektu, viz „Návod na instalaci LWJGL a zprovoznění ukázek v prostředí IntelliJ IDEA“

1. Vytvořte adresář nového projektu, např. u:\java\PGRF3\_samples.
2. Pro projekt potřebujeme následující projekty dostupné na:

- lwjgl-samples:

<https://gitlab.com/Bruno.Jezek/lwjgl-samples.git>

- transforms:

<https://gitlab.com/honza.vanek/transforms.git>

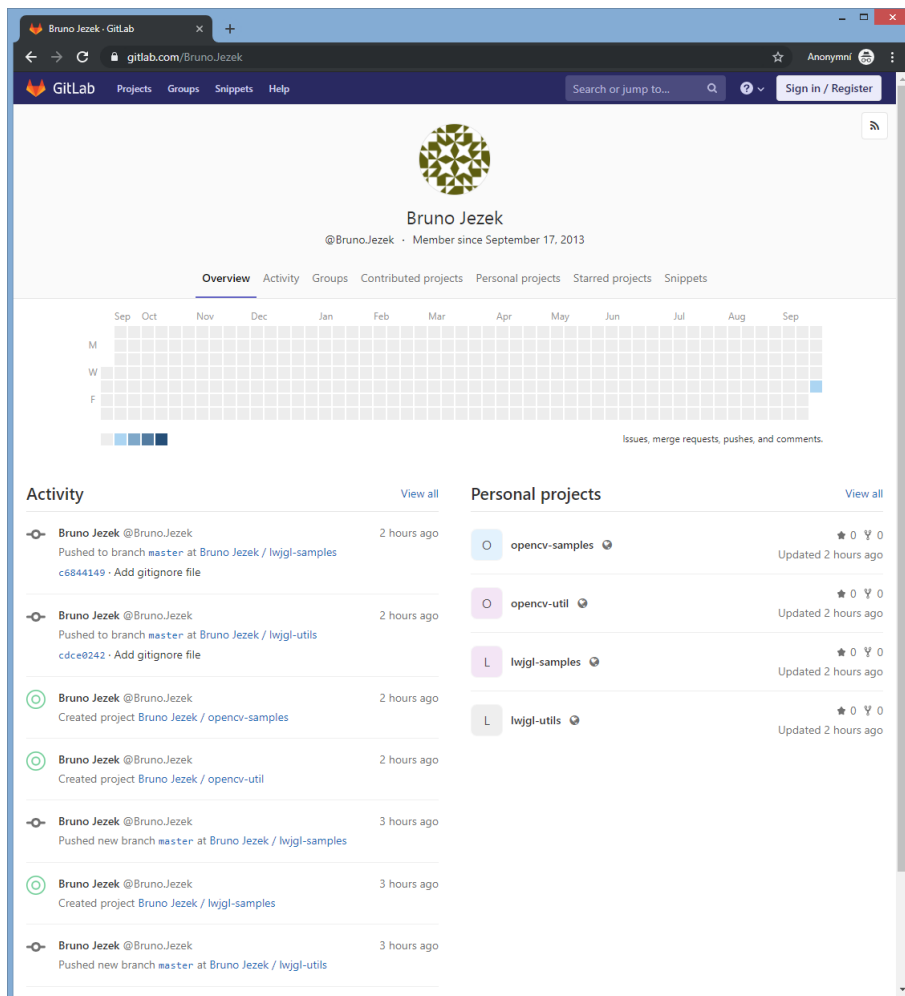
- lwjgl-utils:

<https://gitlab.com/Bruno.Jezek/lwjgl-utils.git>

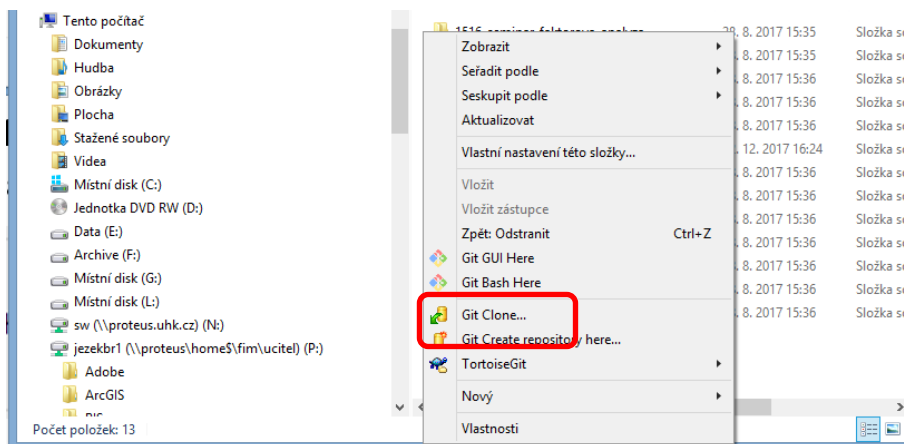
- případně opencv-utils a opencv-samples:

<https://gitlab.com/Bruno.Jezek/opencv-util.git>

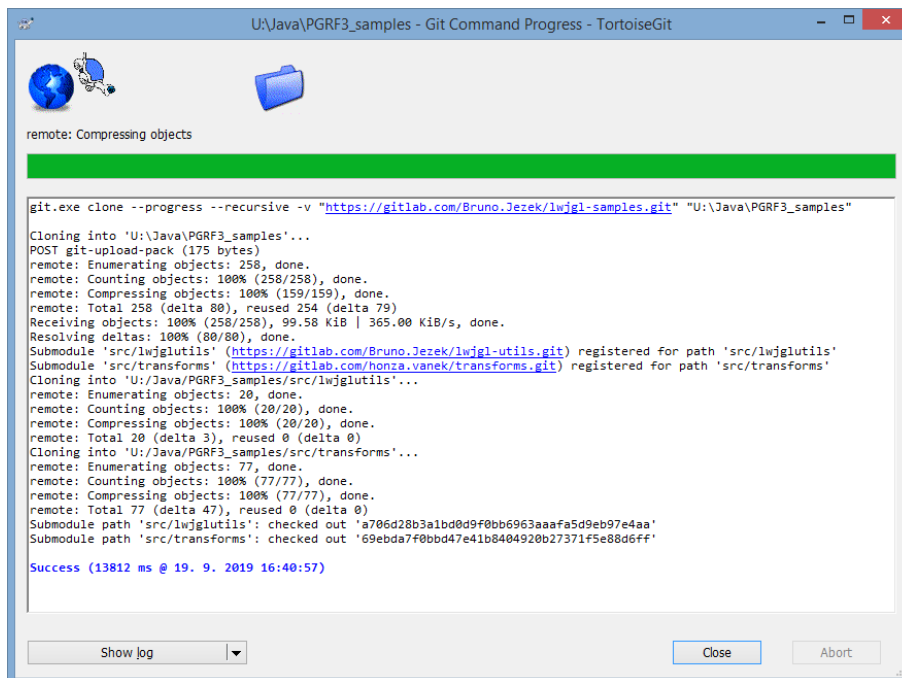
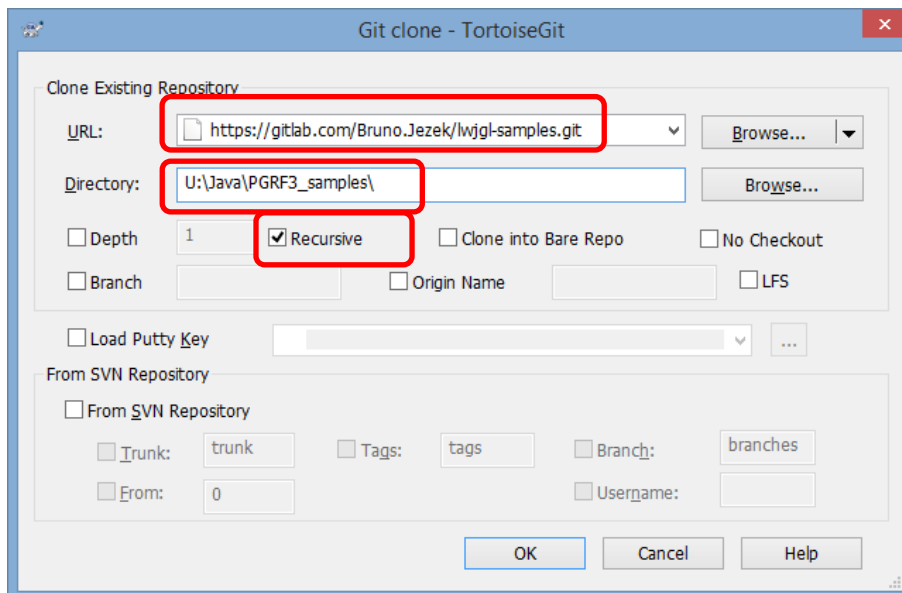
<https://gitlab.com/Bruno.Jezek/opencv-samples.git>



3. Do adresáře workspace Eclipse vytvořte prázdný adresář budoucího projektu (např. u:\java\PGRF3\_samples) a pomocí TortoiseGit funkce Git Clone...



Pro klon projektu <https://gitlab.com/Bruno.Jezek/lwjgl-samples.git> nastavte cestu do adresáře u:\java\PGRF3\_samples



Využijte možnosti rekurzivního klonování všech vnořených modulů, abyste nemuseli klonovat i další adresáře (moduly) vnořené v projektu (transforms a lwjgl-utils).

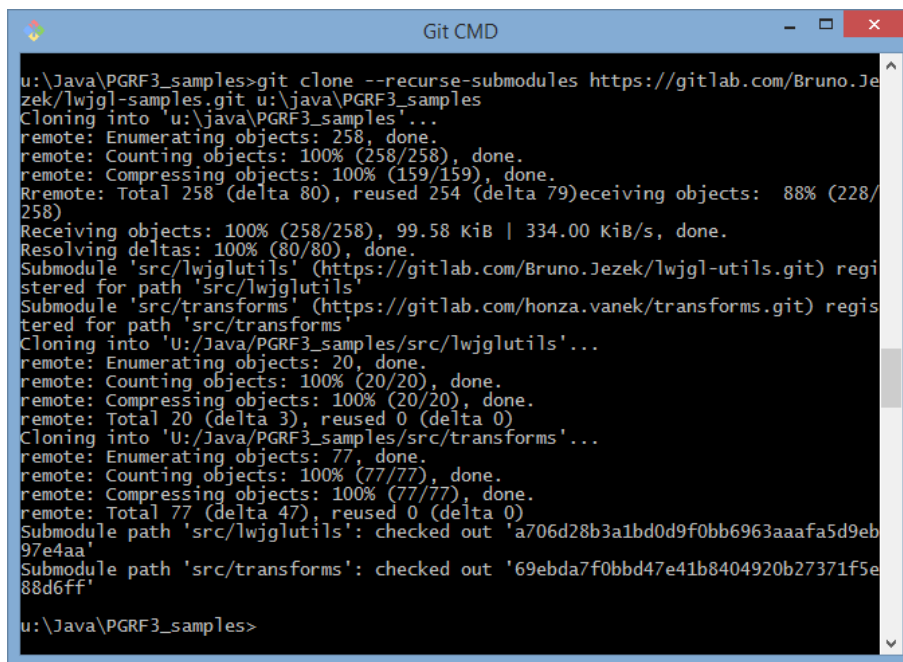
4. Pokud nemáte nainstalován TortoiseGIT, lze použít i konzolové rozhraní Gitu, git-cmd.exe resp. git-bash.exe. V adresáři s Java projekty vytvořit klon repozitáře pomocí příkazů git clone. Např. Klon repozitáře glsl-samples, ale bez modulů transforms a oglutils do zvoleného adresáře:

```
git clone https://gitlab.com/Bruno.Jezek/lwjgl-samples.git u:\java\PGRF3_samples
```

Klon celého repozitáře glsl-samples včetně transforms a lwjgl-utils do zvoleného adresáře s volbou rekurzivního volání:

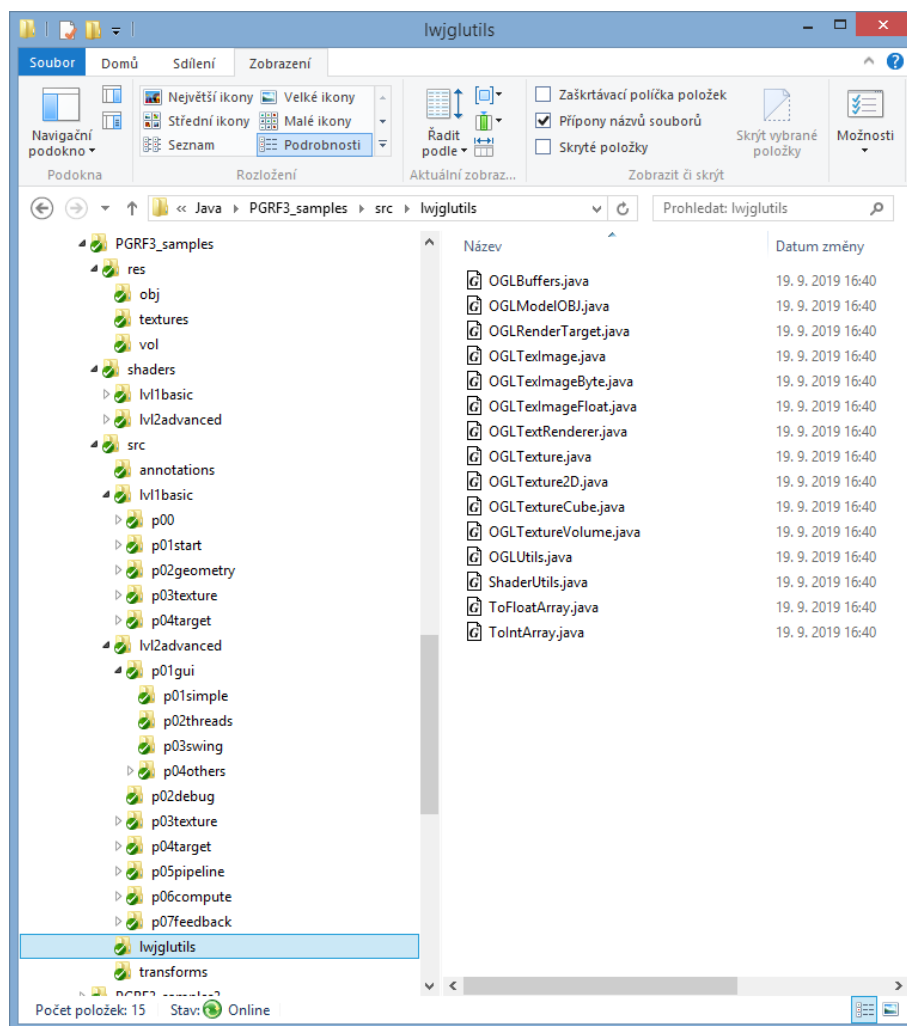
```
git clone --recurse-submodules https://gitlab.com/Bruno.Jezek/lwjgl-samples.git  
u:\java\PGRF3_samples
```

Rekurzivní volání funguje pro verze gitu 2.13 a výše. Starší verze mají parametr: --recursive. Viz: <https://stackoverflow.com/questions/3796927/how-to-git-clone-including-submodules>

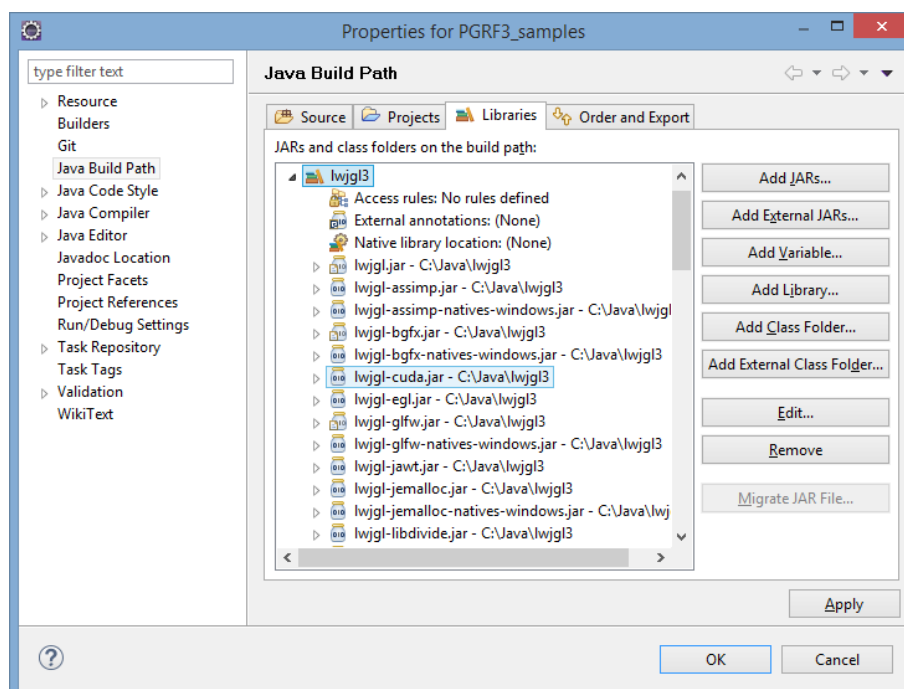
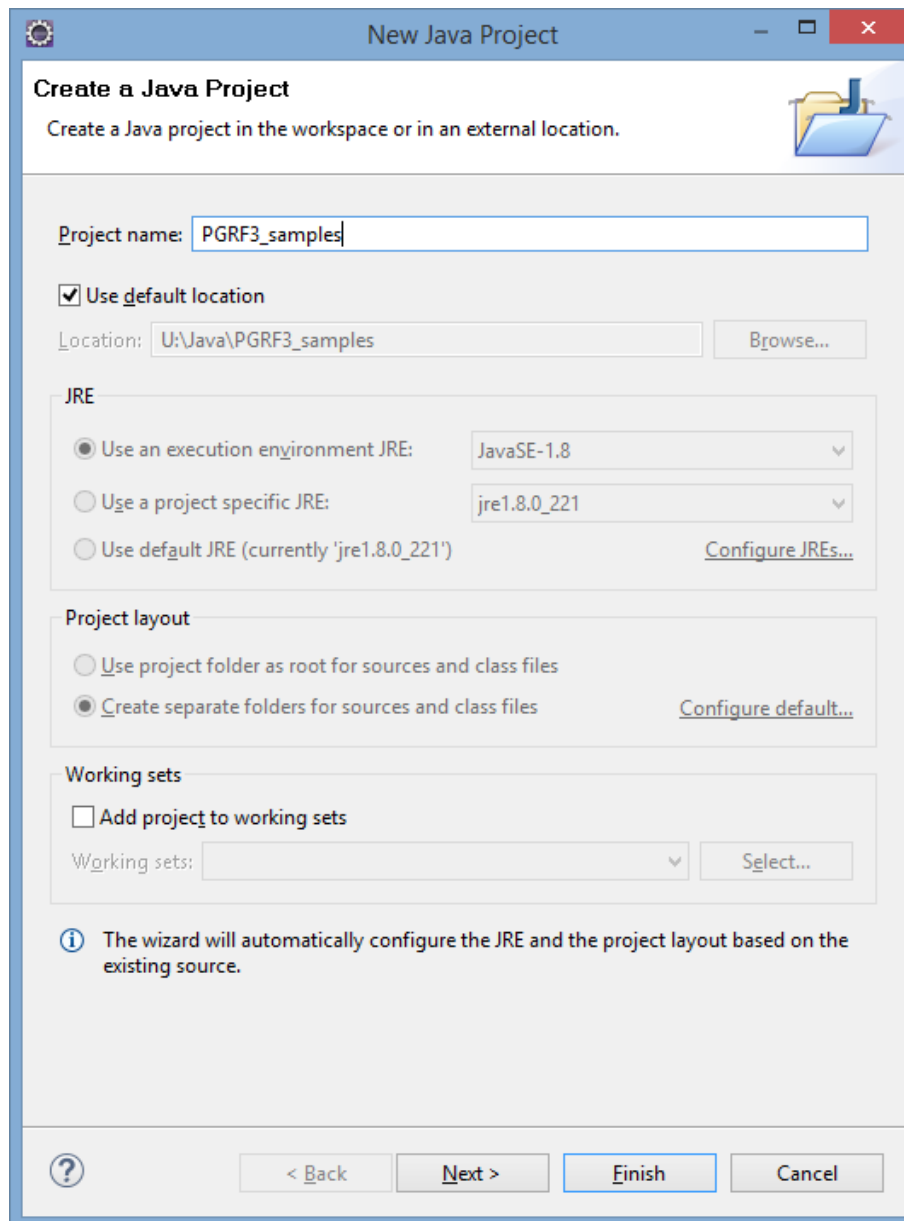


```
Git CMD  
u:\Java\PGRF3_samples>git clone --recurse-submodules https://gitlab.com/Bruno.Jezek/lwjgl-samples.git u:\java\PGRF3_samples  
Cloning into 'u:\java\PGRF3_samples'...  
remote: Enumerating objects: 258, done.  
remote: Counting objects: 100% (258/258), done.  
remote: Compressing objects: 100% (159/159), done.  
remote: Total 258 (delta 80), reused 254 (delta 79)  
Receiving objects: 88% (228/258)  
Receiving objects: 100% (258/258), 99.58 KiB | 334.00 KiB/s, done.  
Resolving deltas: 100% (80/80), done.  
Submodule 'src/lwjglutils' (https://gitlab.com/Bruno.Jezek/lwjgl-utils.git) registered for path 'src/lwjglutils'  
Submodule 'src/transforms' (https://gitlab.com/honza.vanek/transforms.git) registered for path 'src/transforms'  
Cloning into 'U:/Java/PGRF3_samples/src/lwjglutils'...  
remote: Enumerating objects: 20, done.  
remote: Counting objects: 100% (20/20), done.  
remote: Compressing objects: 100% (20/20), done.  
remote: Total 20 (delta 3), reused 0 (delta 0)  
Cloning into 'U:/Java/PGRF3_samples/src/transforms'...  
remote: Enumerating objects: 77, done.  
remote: Counting objects: 100% (77/77), done.  
remote: Compressing objects: 100% (77/77), done.  
remote: Total 77 (delta 47), reused 0 (delta 0)  
Submodule path 'src/lwjglutils': checked out 'a706d28b3a1bd0d9f0bb6963aaafa5d9eb97e4aa'  
Submodule path 'src/transforms': checked out '69ebda7f0bbd47e41b8404920b27371f5e88d6ff'  
u:\Java\PGRF3_samples>
```

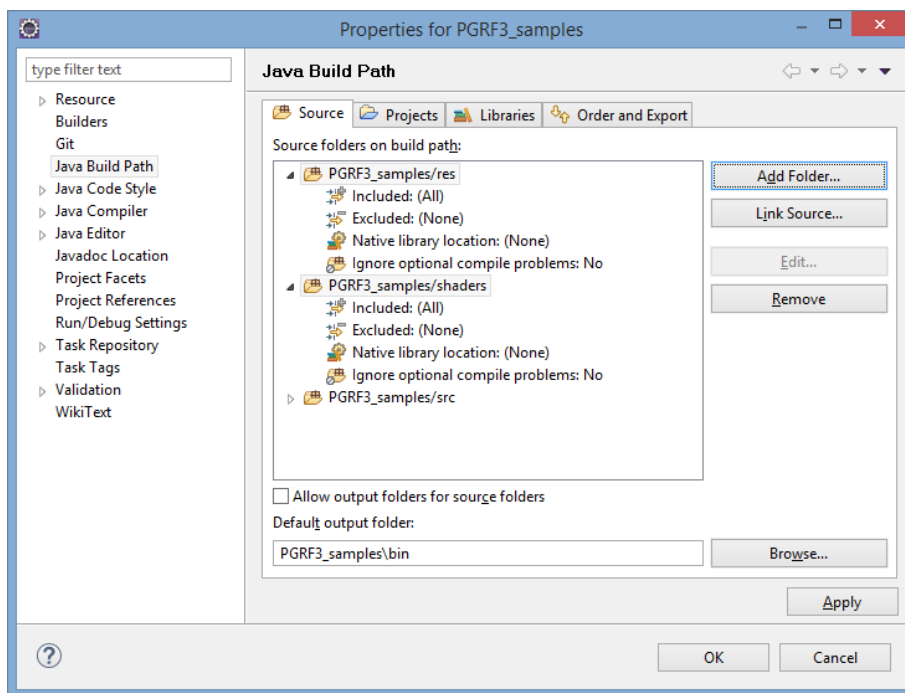
5. Výsledná adresářová struktura bude mít podobu Java projektu s adresáři **src**, **res** a případně **shaders**.



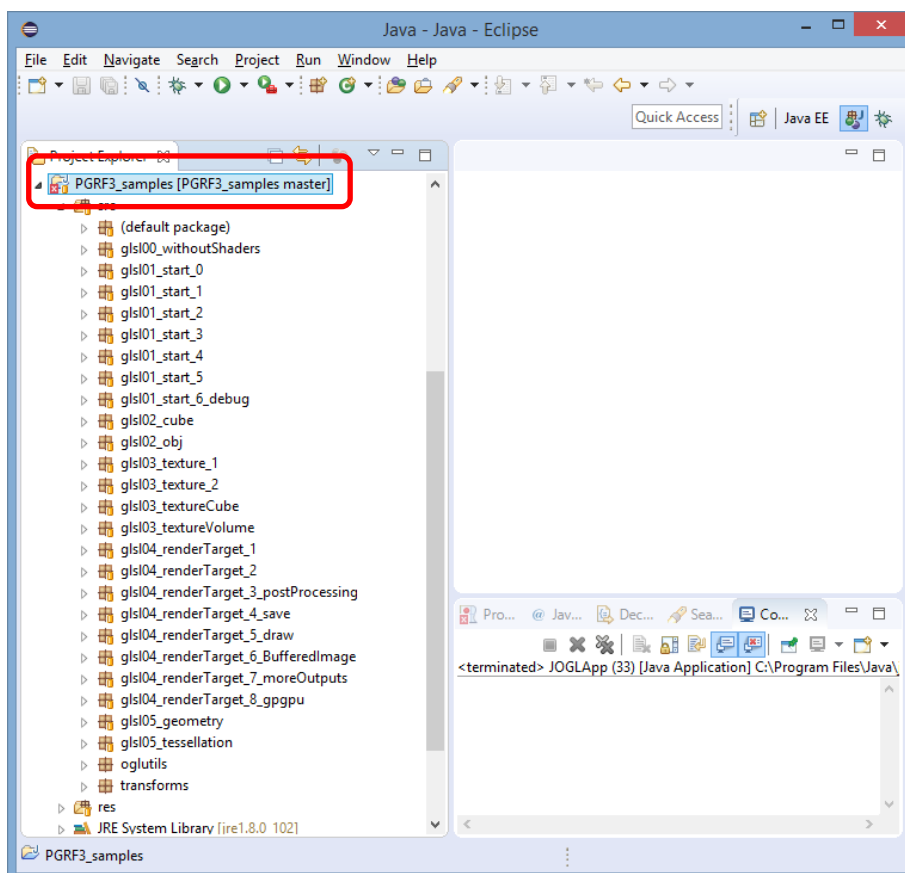
6. Vytvořte projekt v Eclipse, přidejte uživatelské knihovny LWJGL (viz návod „Návod na instalaci knihovny LWJGL v prostředí Eclipse“).

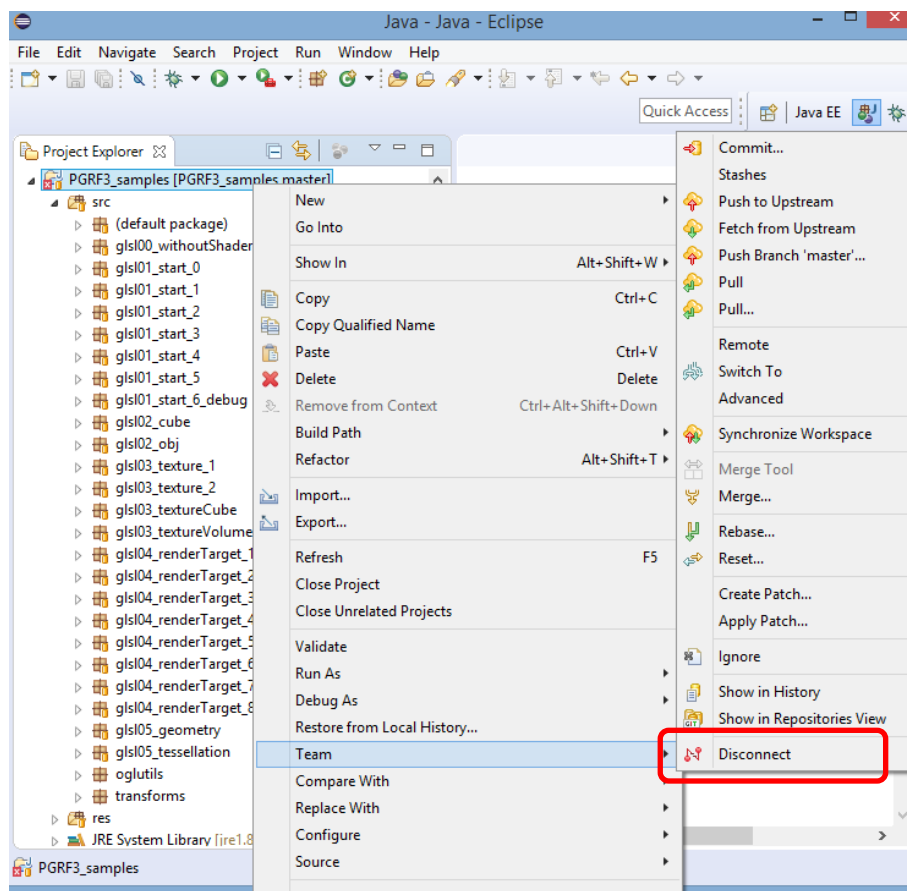


7. Pokud projekt obsahuje kromě src ještě další adresáře, přidejte je do projektu, např. pro PGRF3\_samples přidejte adresáře **shaders** a **res**.



8. V některých verzích Eclipse může být problém u projektu s gitovými adresáři a projekt nepůjde přeložit. Je třeba provést disconnect v kontextovém menu projektu (**Team-Disconnect**).





Projekt přeložte a vyzkoušejte jednotlivé ukázky.





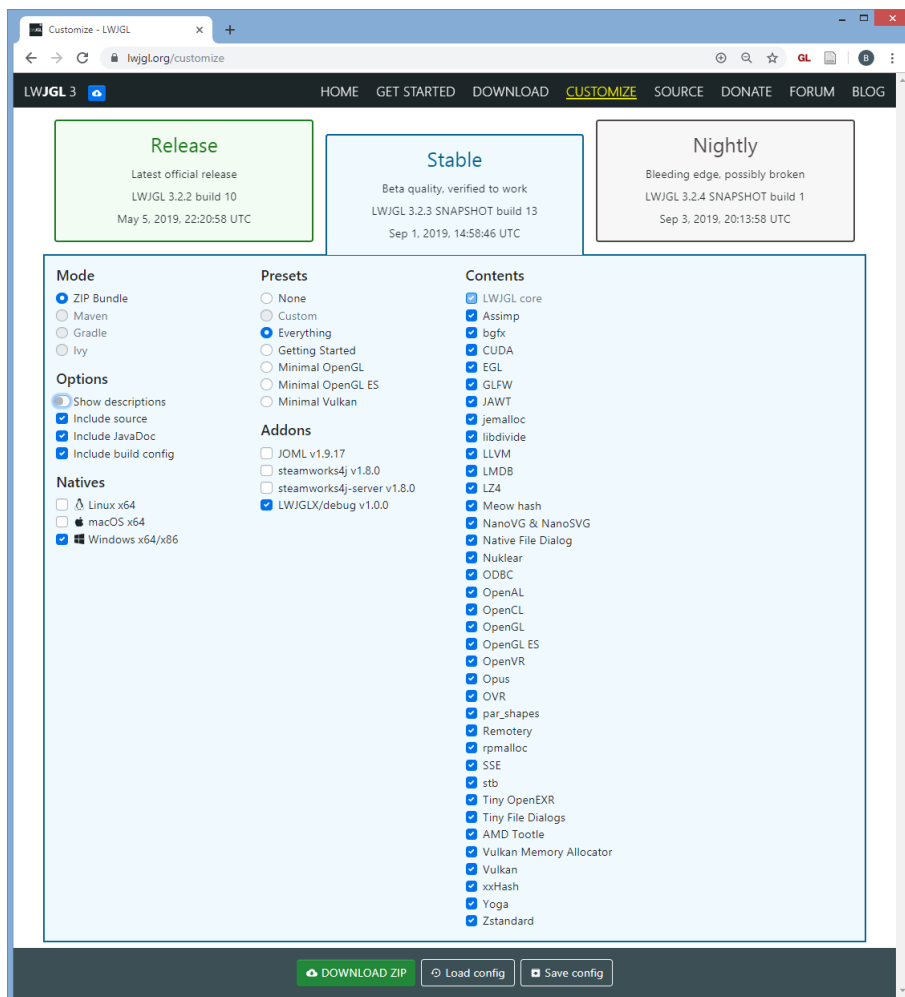
## C. Návod na instalaci knihovny LWJGL v prostředí Eclipse

1. Pokud ještě nemáte nainstalováno, stáhněte si a nainstalujte aktuální verzi JDK (JRE) a Eclipse

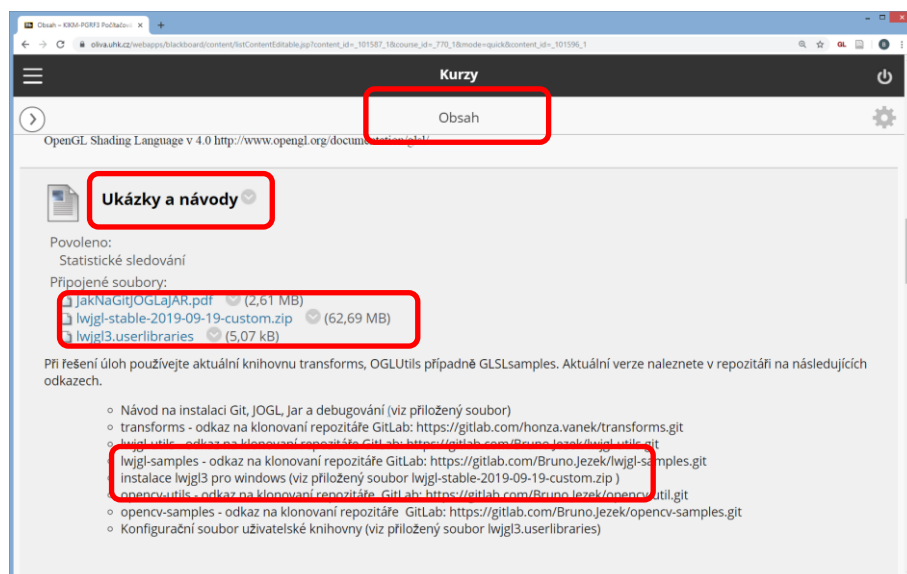
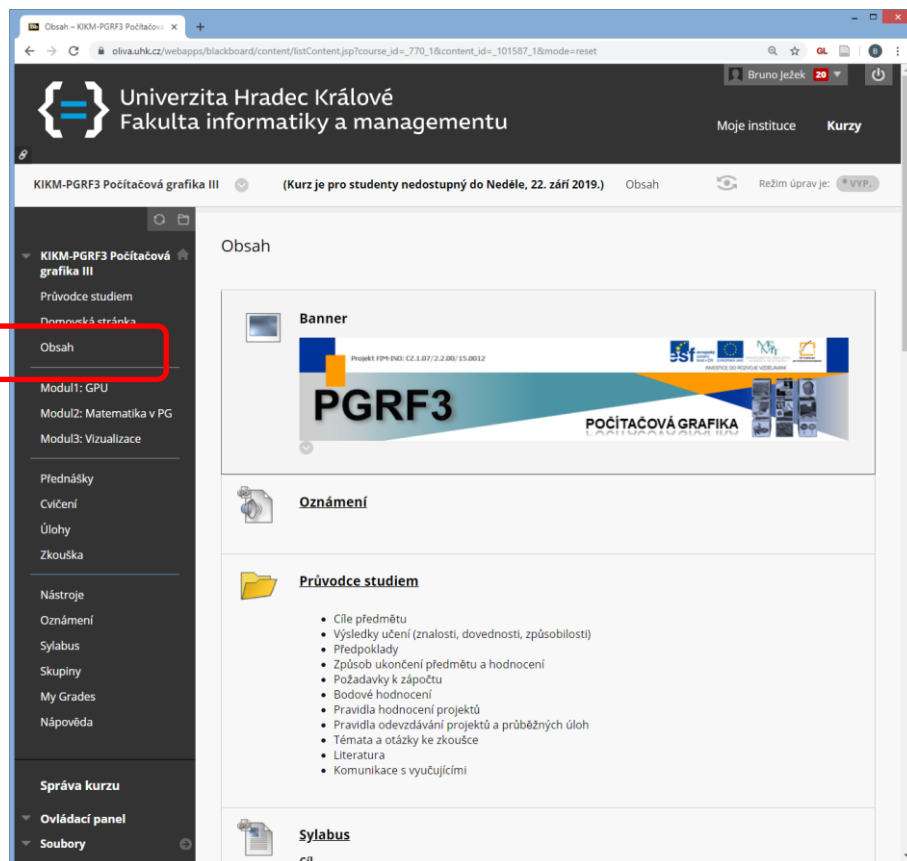
- JDK (JRE) <http://www.oracle.com/>, <http://www.java.com>
- Eclipse <http://www.eclipse.org/>

2. Stáhněte si aktuální verzi knihovny a dokumentaci LWJGL 3 (<https://www.lwjgl.org>)

Download aktuální stabilní verze je možný z oficiálních stránek projektu (<https://www.lwjgl.org/download>), z gitového repozitáře (<https://github.com/LWJGL/lwjgl3/releases>) nebo pomocí konfiguratoru (<https://www.lwjgl.org/customize>)

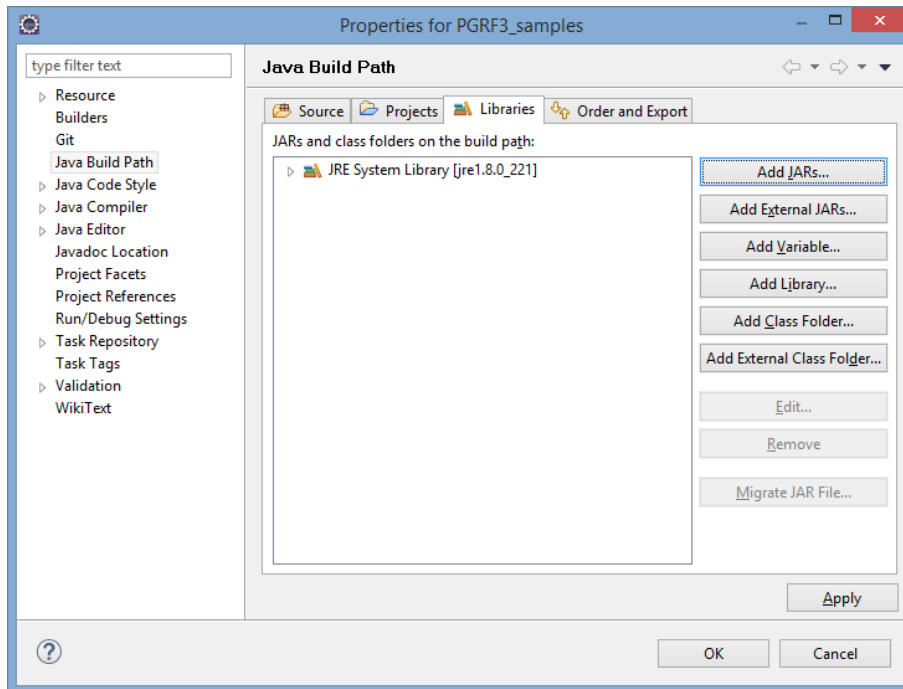


Lze použít i odkazy na kopie souborů v BB <https://oliva.uhk.cz/> v kurzech PGRF2 a PGRF3, které obsahují funkční stabilní build pro Windows. Tyto odkazy nemusí ale obsahovat poslední aktuální verzi. Např. [lwjgl-stable-2019-09-19-custom.zip](#)

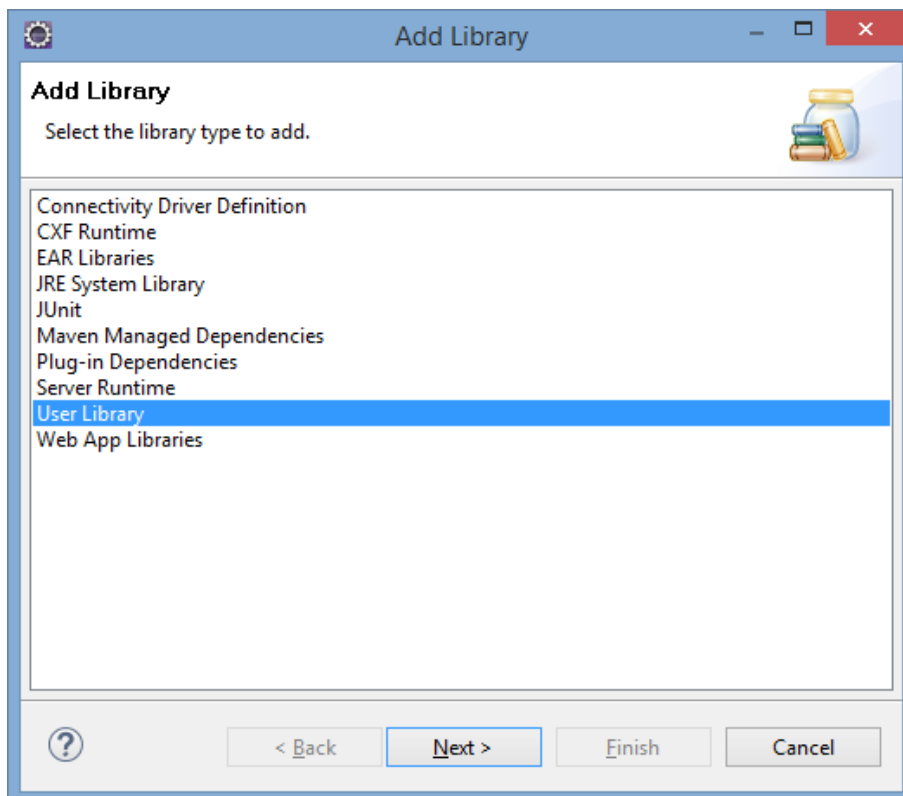


Pro instalaci lze použít i soubory v adresáři <c:\java\lwjgl3> dostupné na lokálních discích na učebnách J budovy FIM UHK.

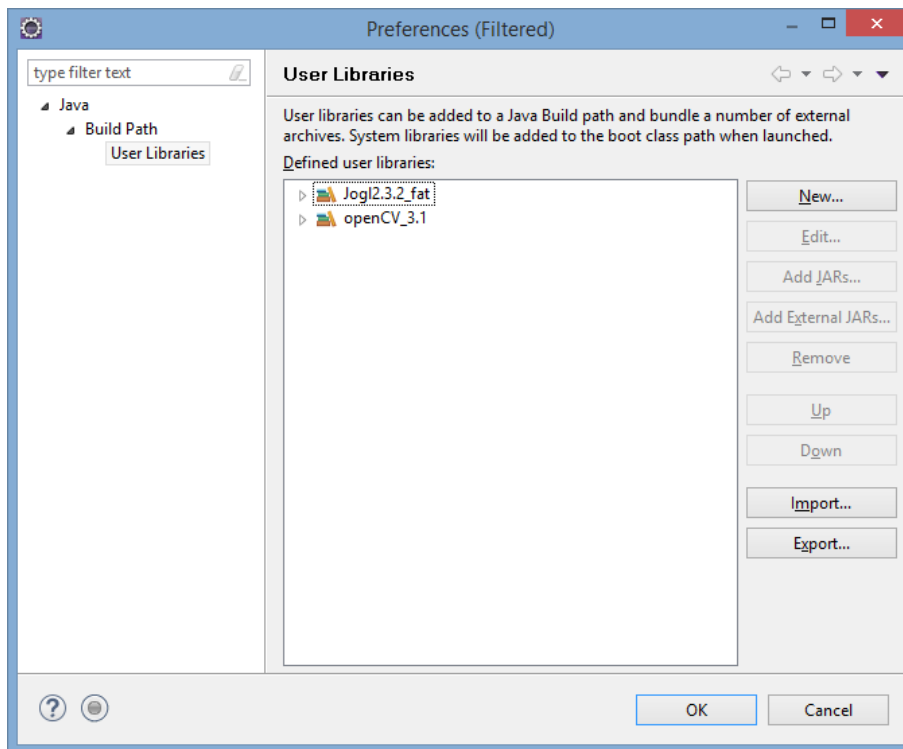
3. Soubor **lwjgl....zip** rozbalte do vhodného adresáře, například **c:\java\lwjgl3**.
4. V prostředí Eclipse vyberte projekt, ve kterém chcete pracovat s knihovnou LWJGL, a klikněte na **Properties – Java Built Path – Libraries**



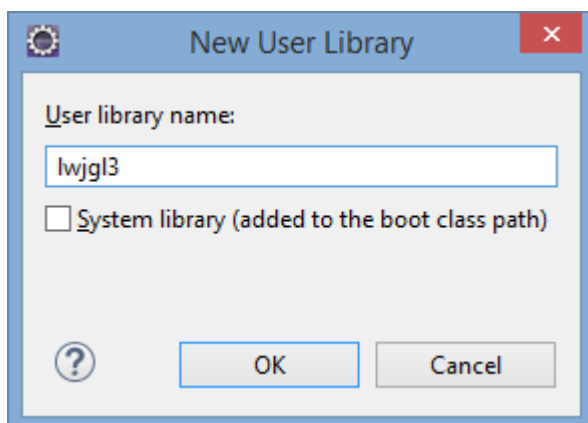
5. Zvolte **Add Library ... – User Library – Next**

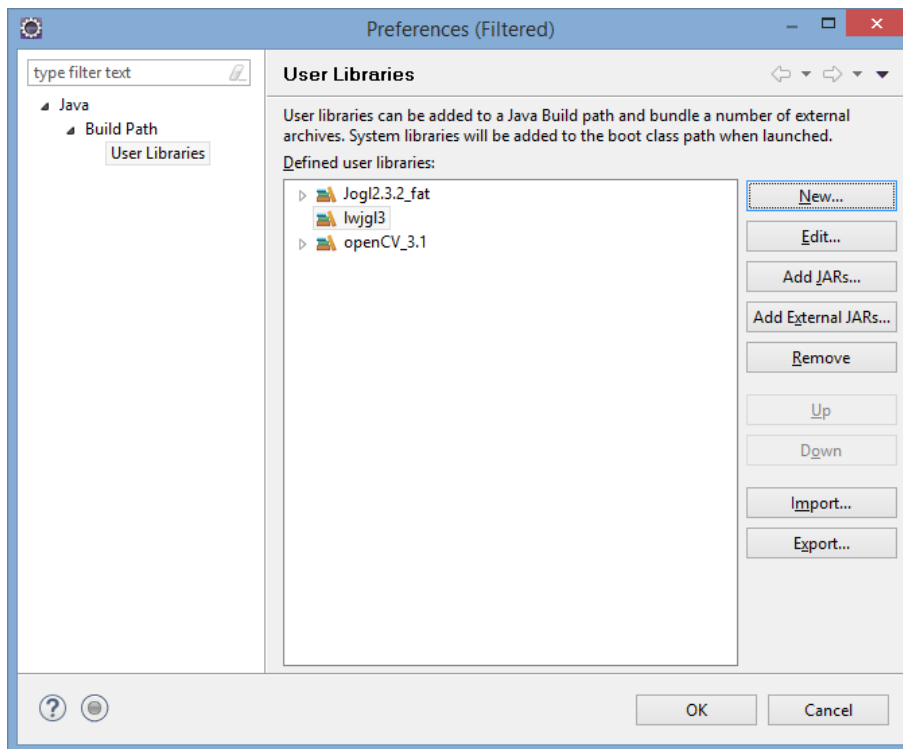


## 6. Zvolte **User Libraries...**

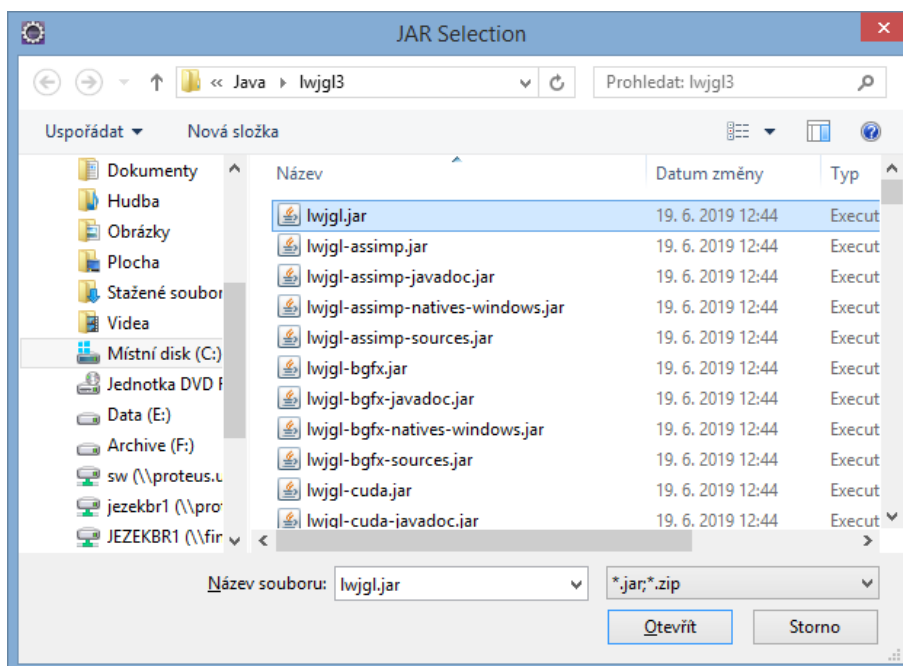


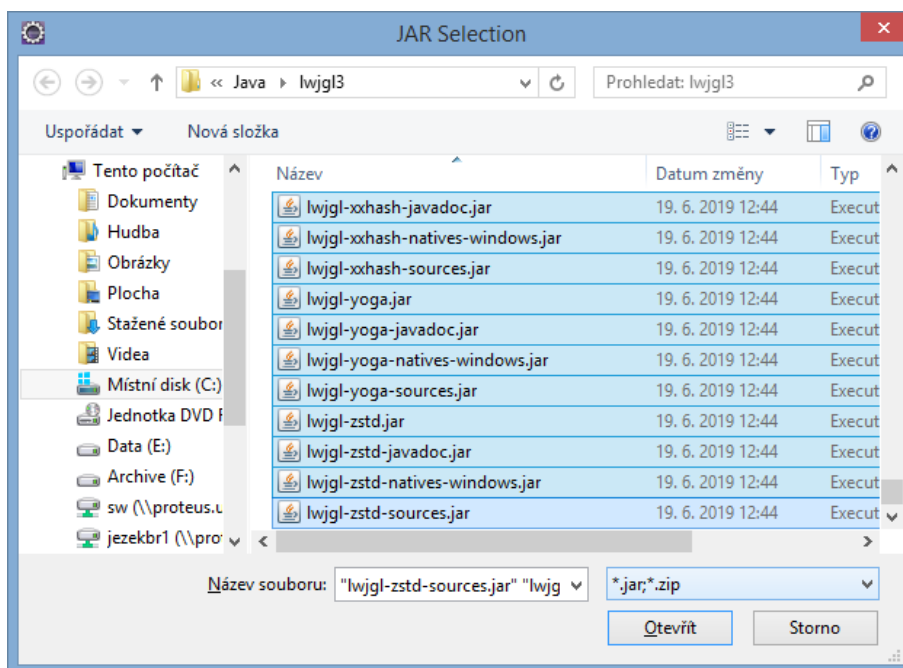
## 7. Vytvořte novou knihovnu **New...** , např. pojmenovanou **lwjgl3**



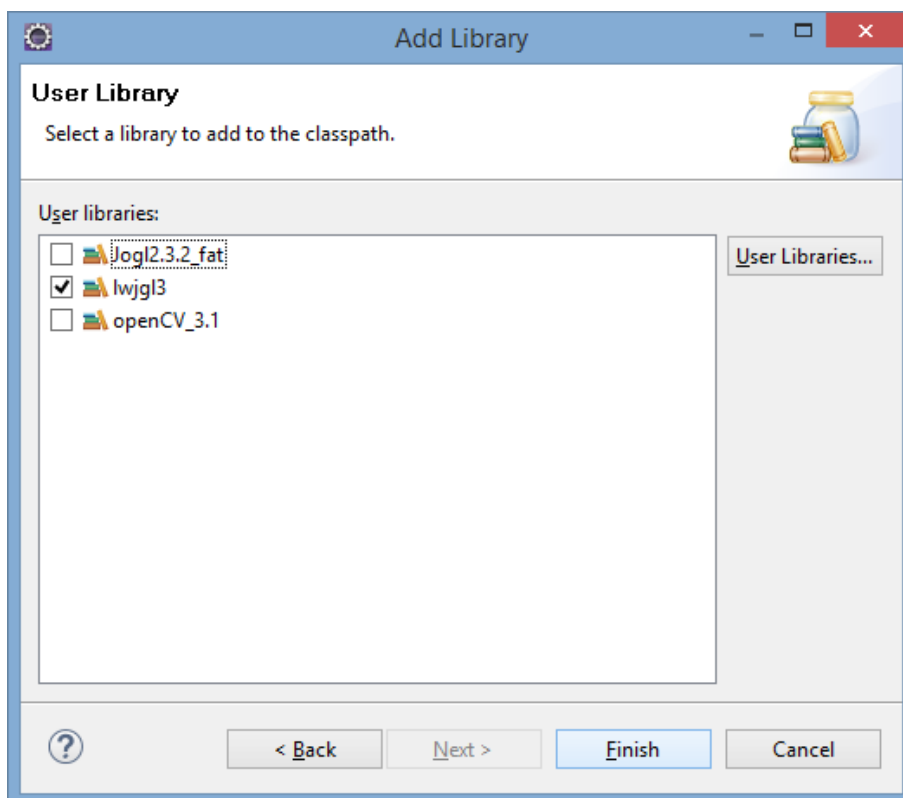


8. Přidejte všechny jar soubory pomocí **Add External JARs...** z vytvořeného adresáře, např. `c:\java\lwjgl3`

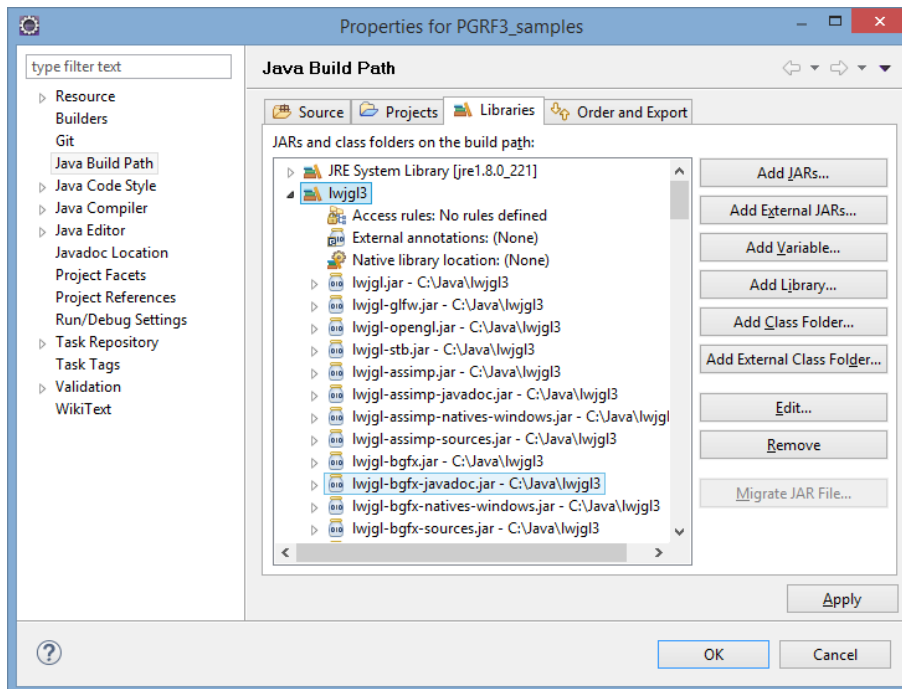




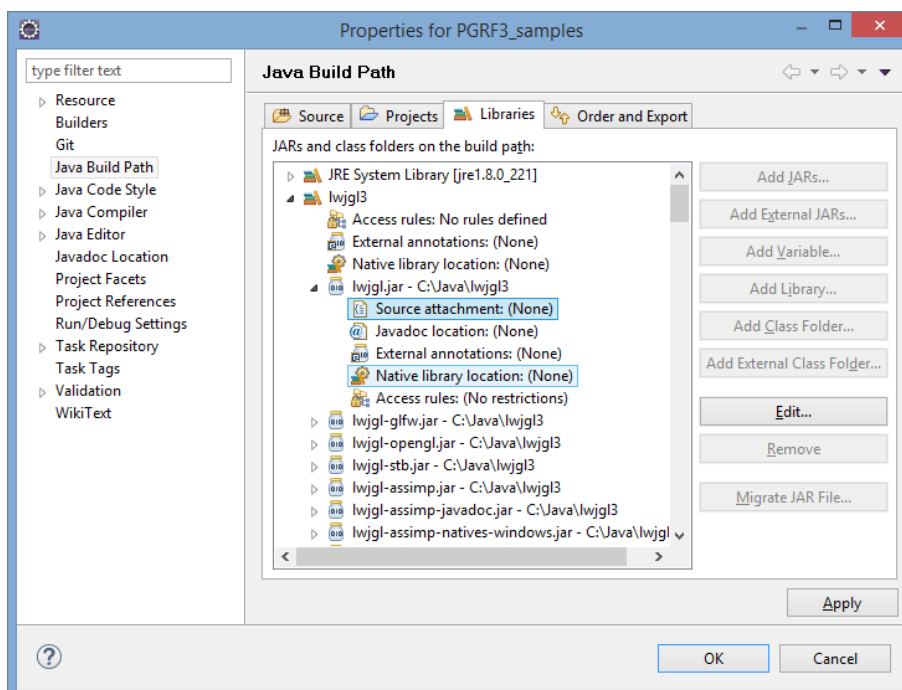
Přidejte celou uživatelskou knihovnu do projektu

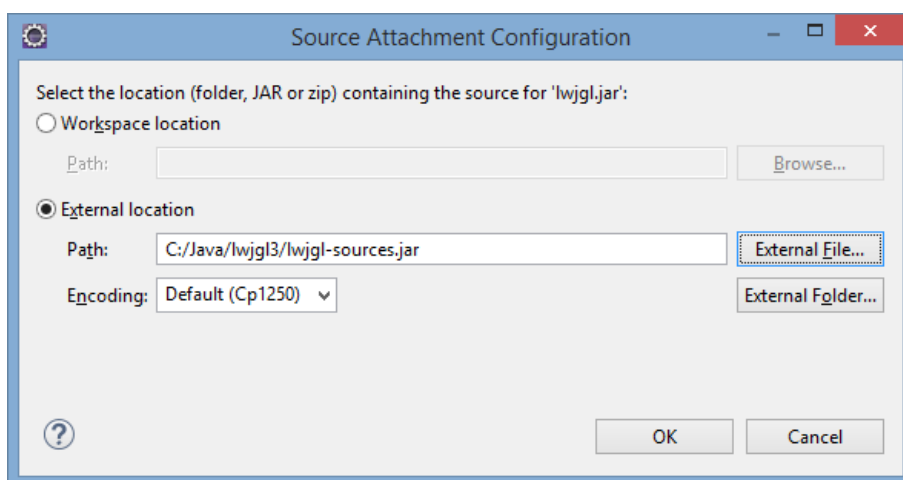
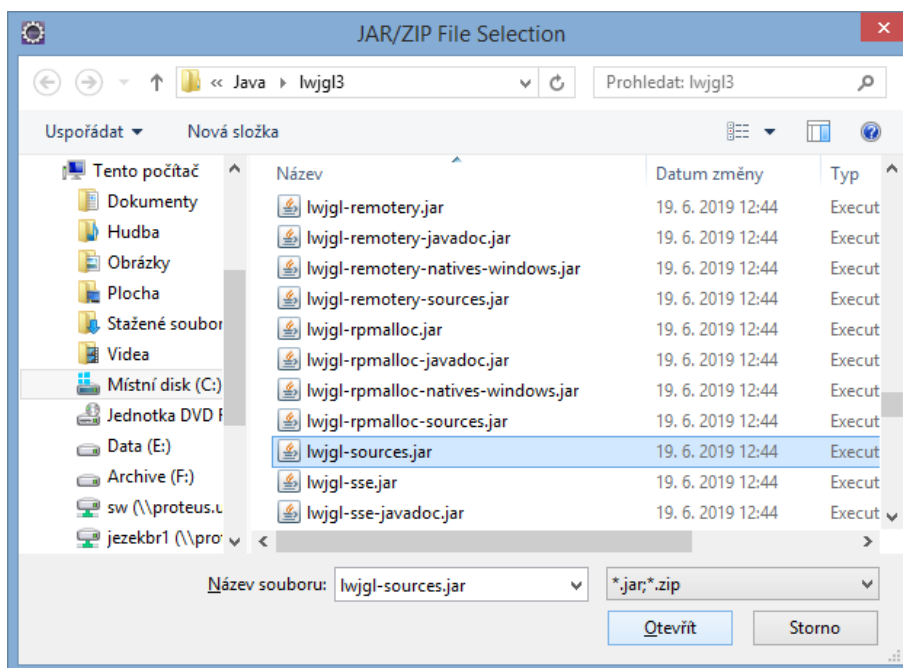


Nastavení knihoven projektu bude mít následující podobu

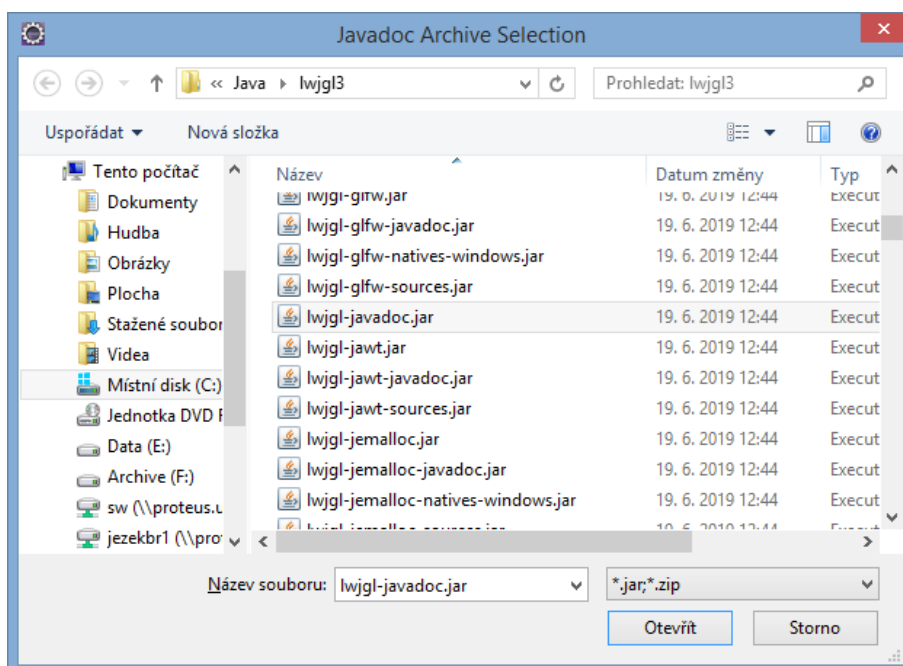


9. (Nepovinné) pro snadnější programování lze přiřadit zdrojové soubory k jednotlivým jar souborům. Rozbalením jar knihoven a kliknutím na položku **Source attachment** můžete tlačítkem **Edit...** přiřadit zdrojové soubory např (C:/java/lwjgl3/lwjgl-sources.jar)

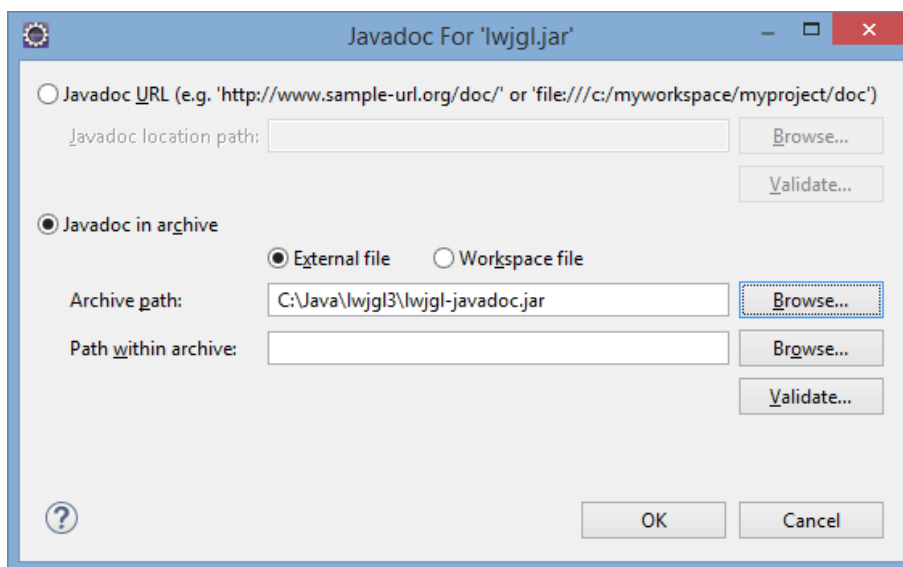




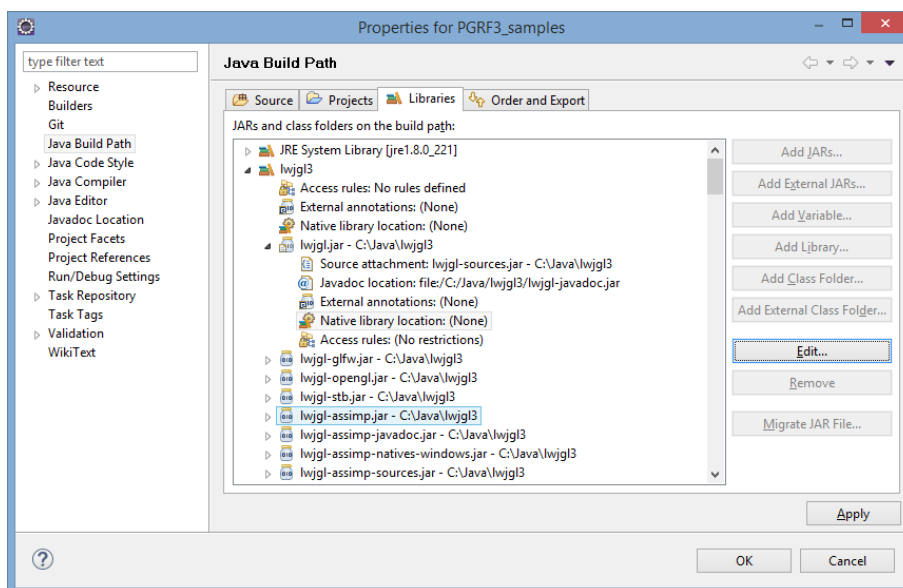
10. (Nepovinné) Výběrem položky **Java doc location** lze přiřadit dokumentaci k jednotlivým knihovnám







11. Výsledné nastavení knihoven projektu bude vypadat následovně



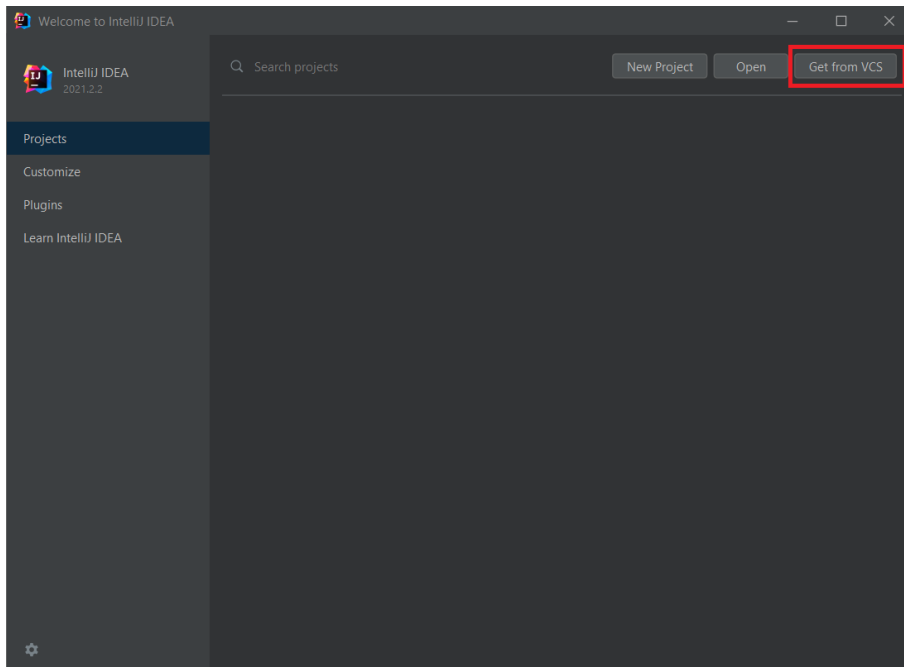
Pokud použijete stejně pojmenovanou strukturu adresářů, můžete použít import připravené uživatelské knihovny z externího souboru pomocí **Properties – Java Built Path – Libraries – Add Library - User Library – Next – User Libraries – Import** a soubor \*.userlibraries, např. lwjgl3.userlibraries. Tento soubor je ke stažení i v kurzech BB.

## D. Návod na instalaci LWJGL a zprovoznění ukázek v prostředí IntelliJ IDEA

1. **Stáhněte IntelliJ IDEA** – <https://www.jetbrains.com/idea/download/>
2. **Stáhněte Javu** (OpenJDK) – <https://openjdk.java.net>
3. **Stažení LWJGL:** Proveďte krok 2 předcházejícího návodu (Návod na instalaci knihovny LWJGL v prostředí Eclipse)

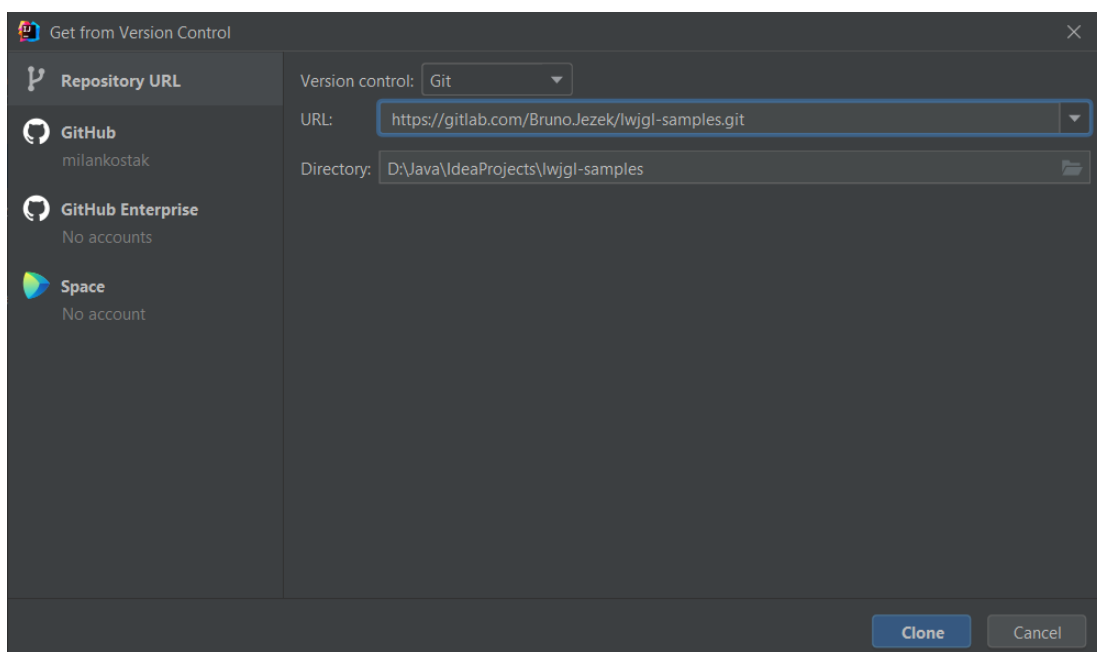
Pro vyhnutí se potenciálním problémům je vhodné při použití konfigurátoru stáhnout variantu „Everything“. Minimální dostačující varianta pro by měla obsahovat core, GLFW, Native File Dialog, Nuklear, OpenGL, stb a Tiny File Dialogs (na vlastní nebezpečí!).

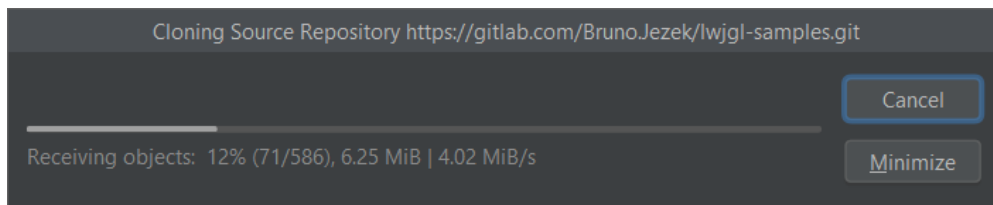
4. **Vytvořte nový projekt** pomocí tlačítka „Get from VCS“



Na následující obrazovce vložte URL repositáře. Složka na lokálním disku bude automaticky vyplněna podle názvu projektu, lze ji však změnit. Tlačítkem „Clone“ bude spuštěno klonování projektu. Rekursivní klonování submodulů bude provedeno automaticky.

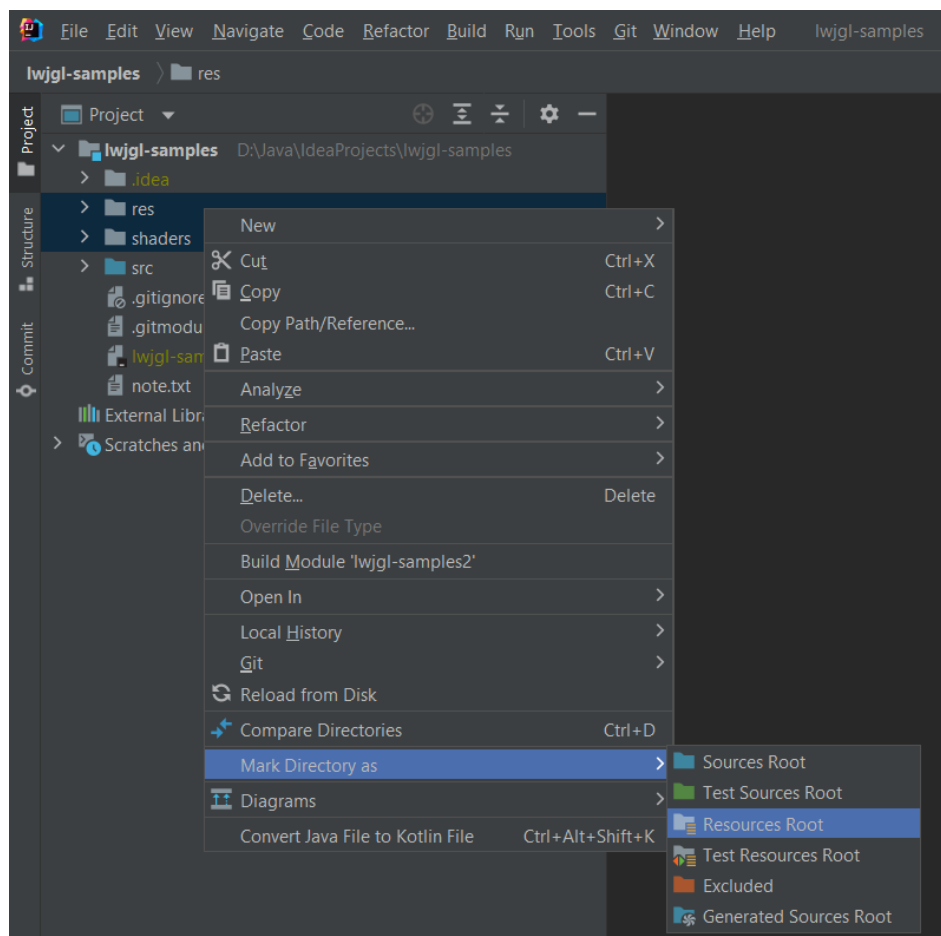
<https://gitlab.com/Bruno.Jezek/lwjgl-samples.git>



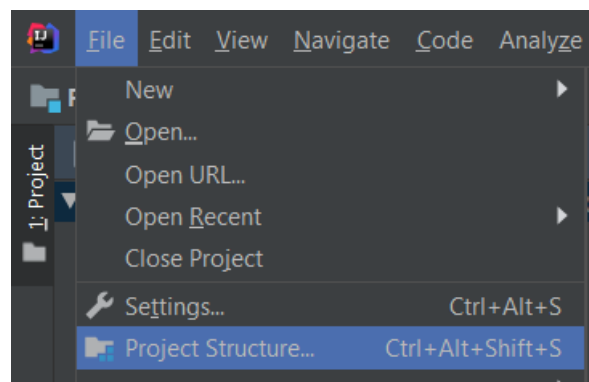


5. **Klonovaný projekt bude automaticky otevřen.**

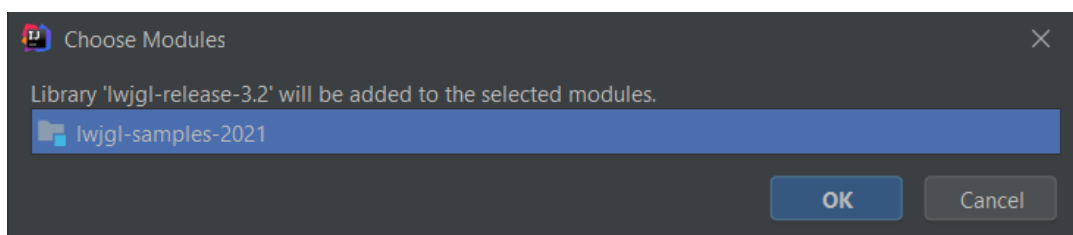
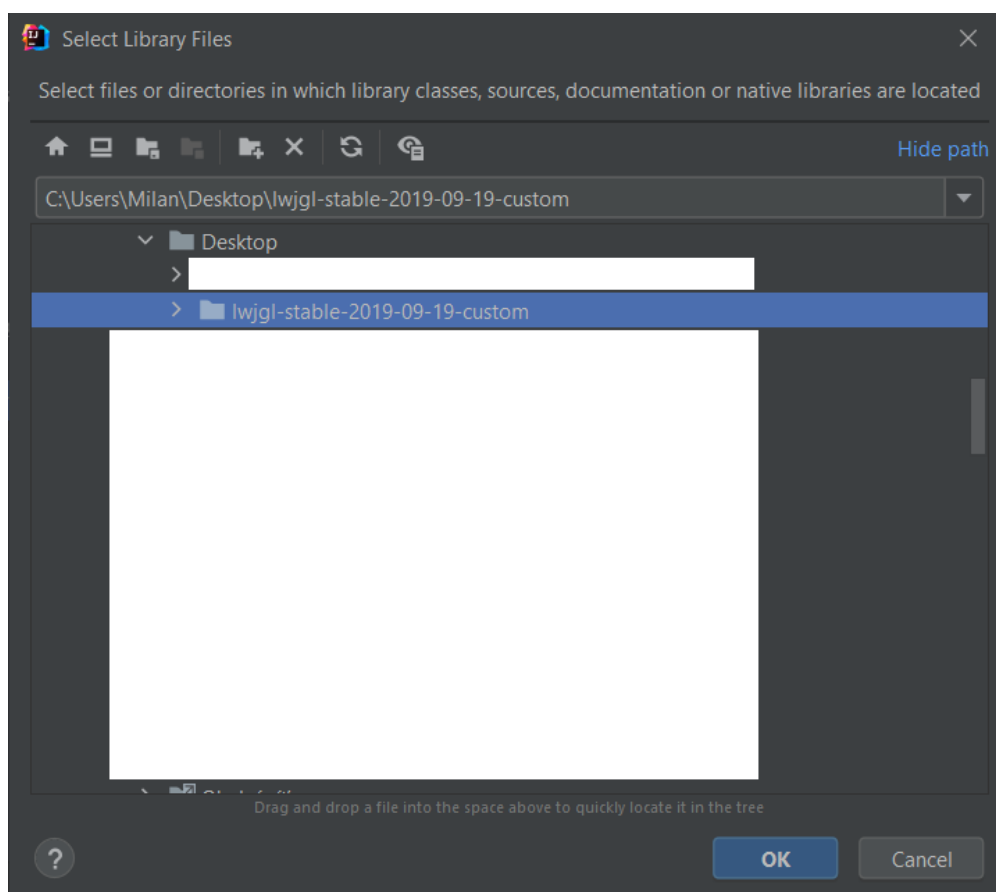
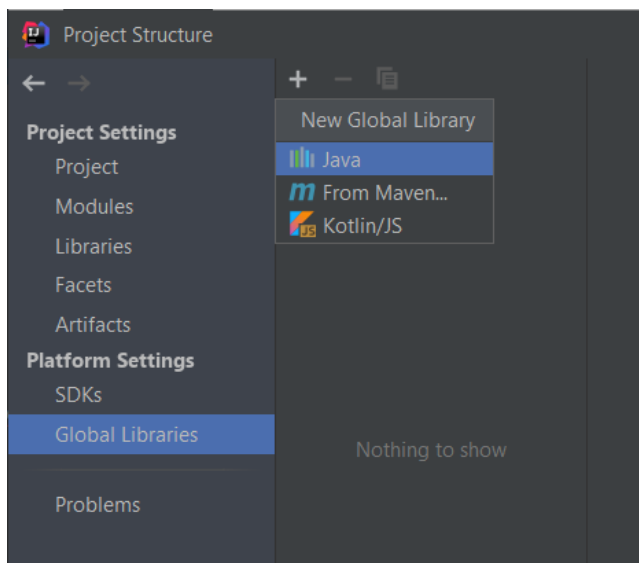
6. **Označení složek projektu.** Aby IDE správně dokázalo najít obrázky a shadery, které jsou nutné ke spuštění projektů, je nutné takové složky označit jako „Resources root“. Složky „res“ a „shaders“ se označí, klikne se pravým tlačítkem myši, následuje volba „Mark Directory as“ a možnost „Resources root“. Pokud by při jednom z přechodů kroků, kdy se automaticky detekovala složka se zdrojovými kódy, došlo k problému, tak stejným způsobem by se složka „src“ označila jako „Sources Root“.



7. **Posledním krokem je nastavení knihovny LWJGL do projektu.** V horním menu zvolte možnost „File“ a položku „Project Structure“.

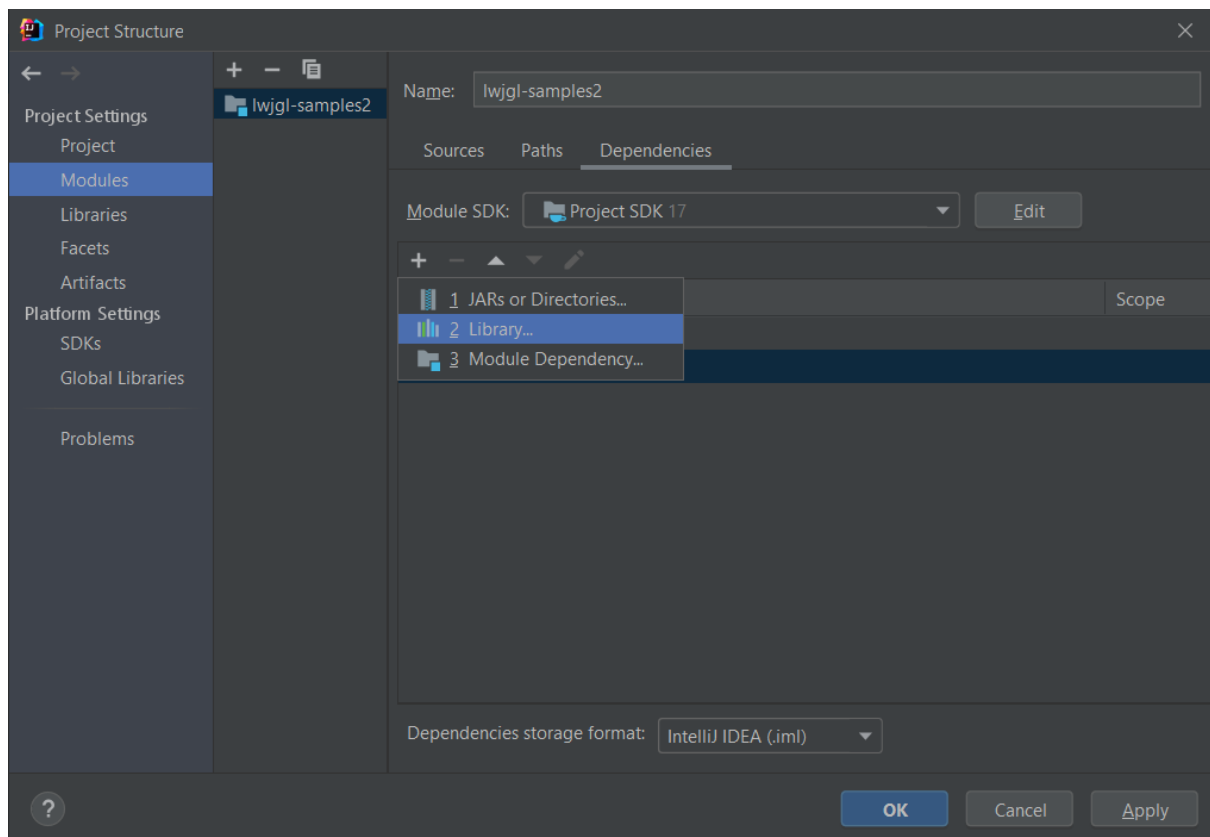


8. Po budoucí znovupoužití knihovny je vhodné LWJGL vložit jako globálního knihovnu. V nově otevřeném okně vyberte vlevo možnost „Global Libraries“. Tlačítkem plus a volbou „Java“ otevřete okno pro výběr knihoven. Jděte do složky se staženým a rozbaleným LWJGL a vyberte všechny podsložky. Volbu potvrďte tlačítkem OK. IDE vám rovnou nabídne soubor knihoven přidat do projektu.

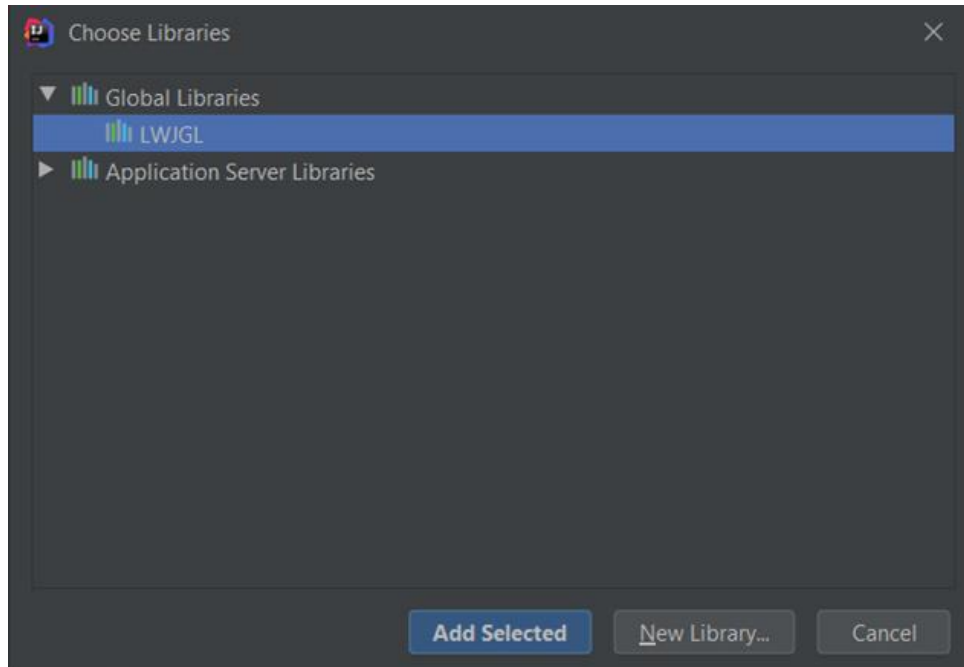


Soubor knihoven můžete nyní přejmenovat.

Pokud musíte LWJGL do projektu přidat ručně, tak ve stejném okně vyberte vlevo možnost „Modules“ a nahore pak možnost „Dependencies“. Vpravo pak stiskněte tlačítko „+“ a vyberte možnost „Library“.

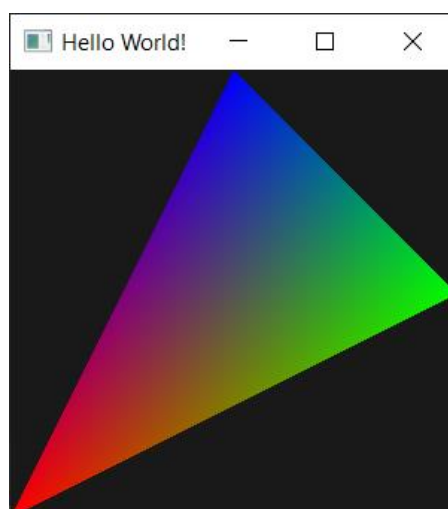
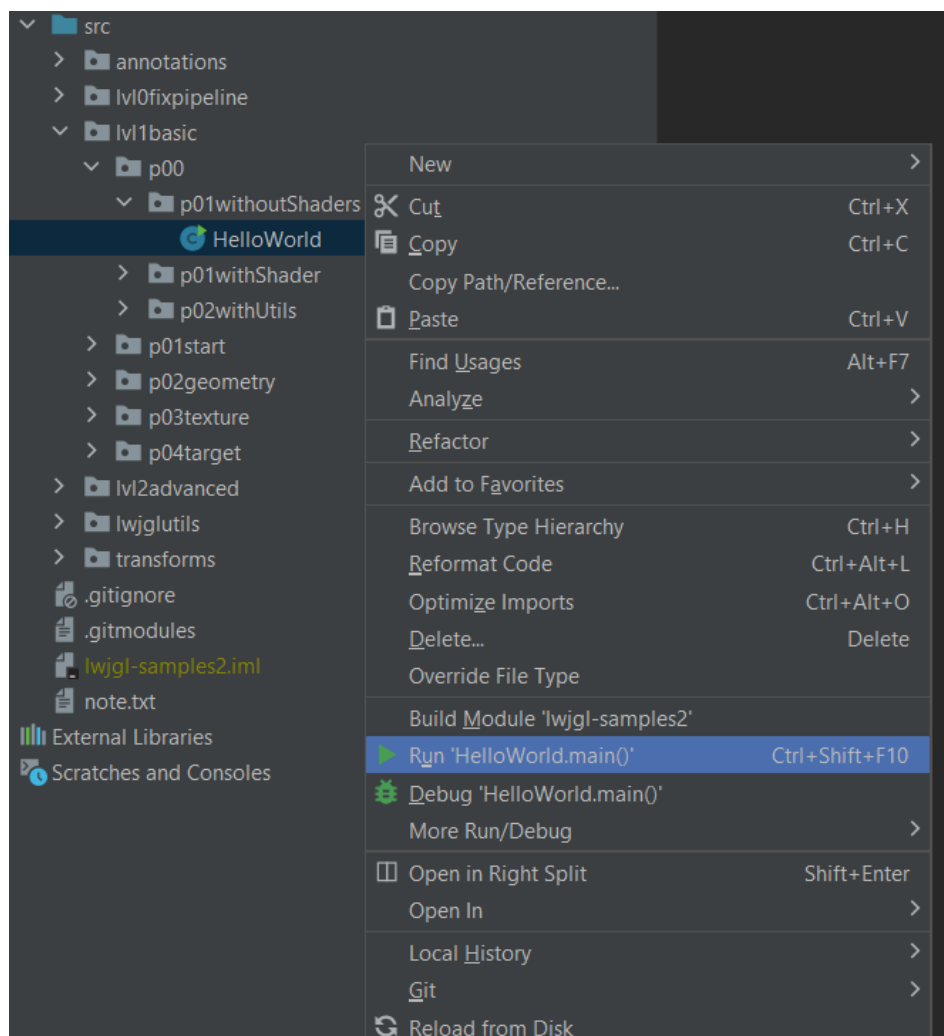
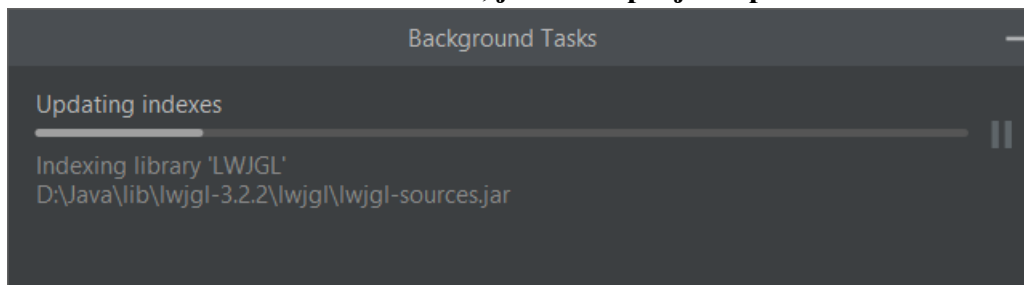


Vyberte soubor knihoven „LWJGL“ (případně podle jména, které jste mu dali) a zvolte „Add Selected“.



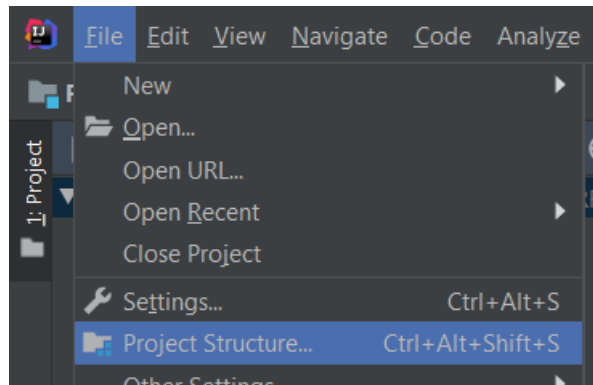
Celé okno s nastavením pak zavřete tlačítkem „OK“ vpravo dole.

## 9. Poté, co IDE dokončí indexování knihoven, je možné projekt spustit.

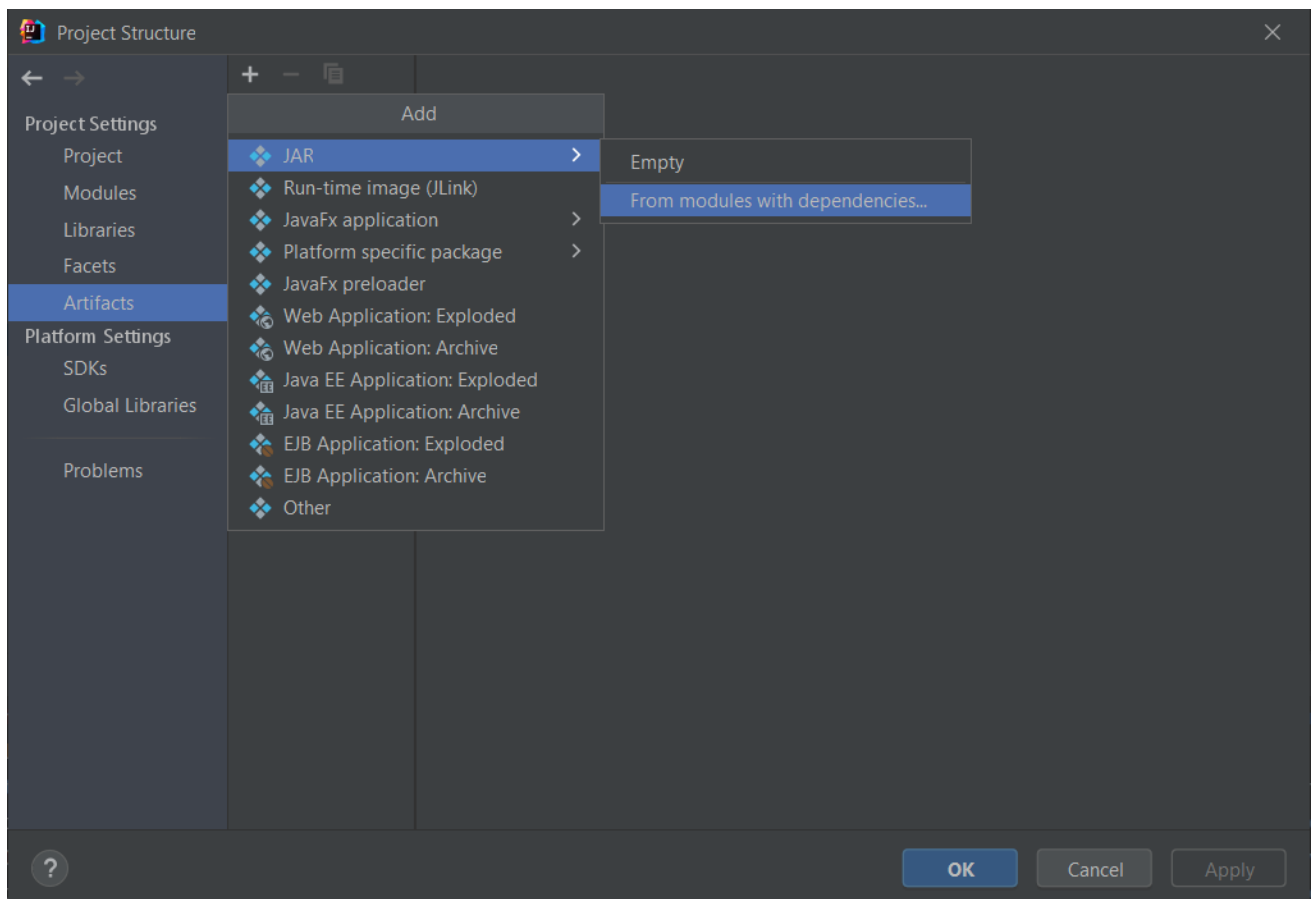


## E. Návod na vytvoření spustitelného JAR souboru v prostředí IntelliJ IDEA při použití knihovny LWJGL

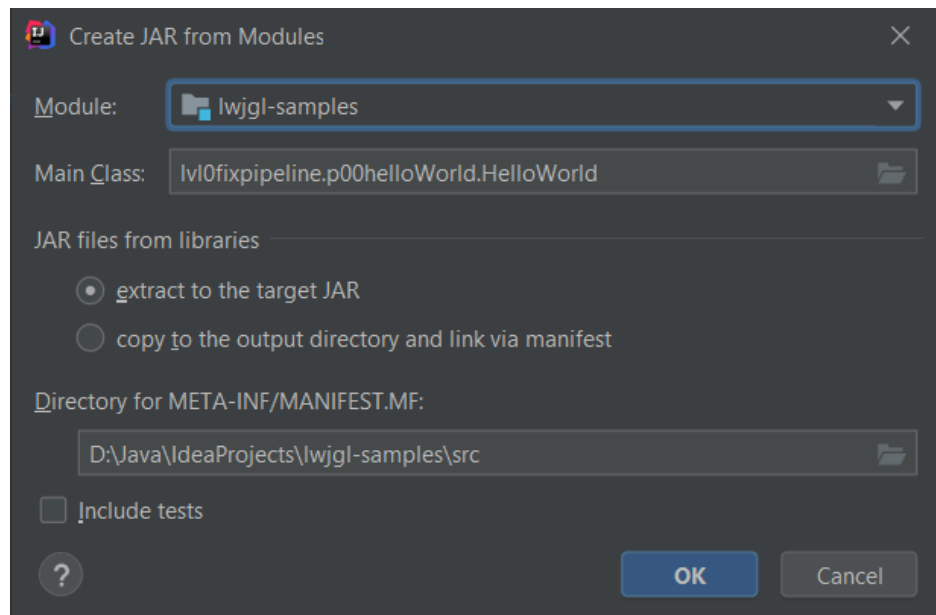
1. V menu vybrat možnost File > Project Structure.



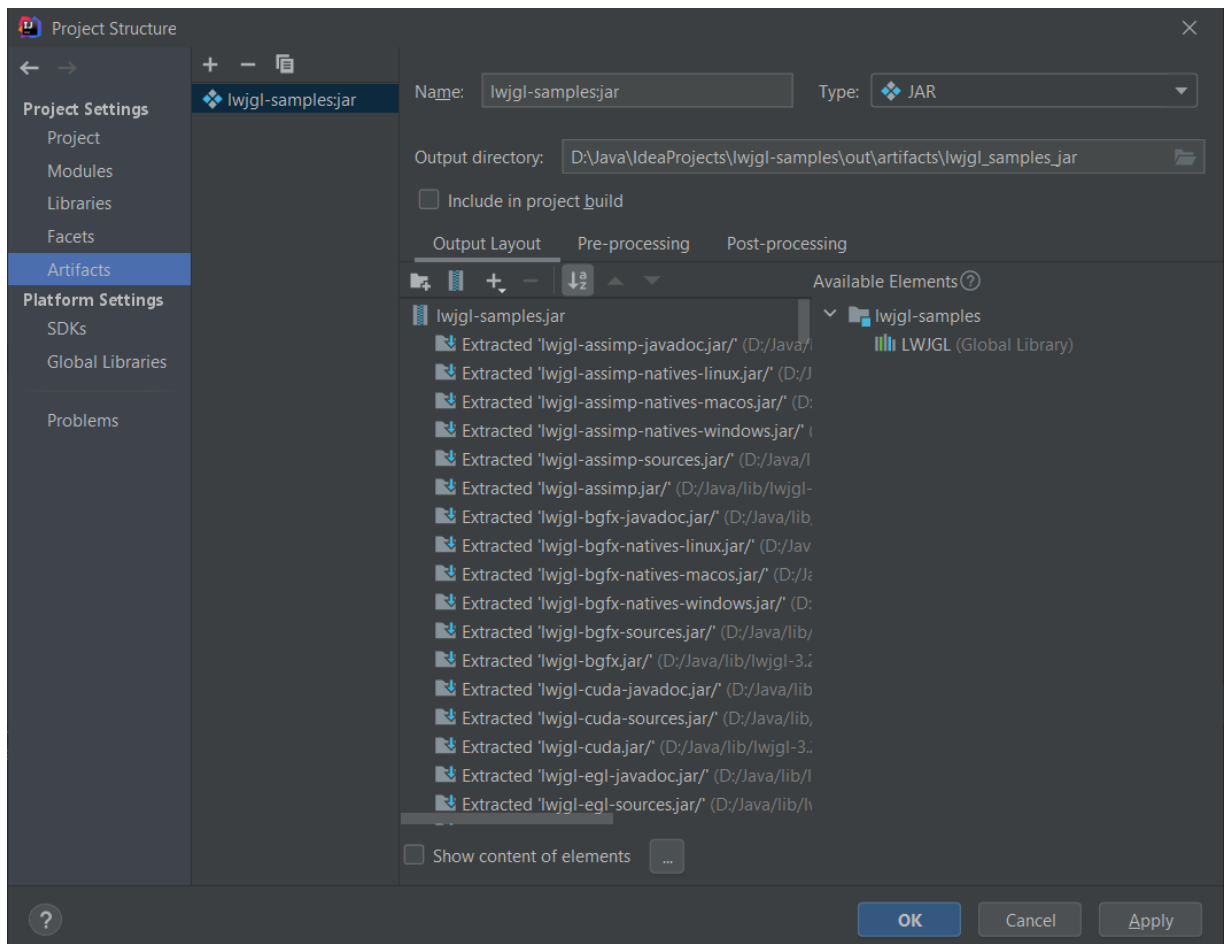
2. Artifacts > + > JAR > From modules with dependencies...



3. **Vybrat spouštěcí třídu** (Main Class). Nastavit cestu pro MANIFEST.MF. Pozor, že složka META-INF se v cílovém adresáři ještě nesmí nacházet.

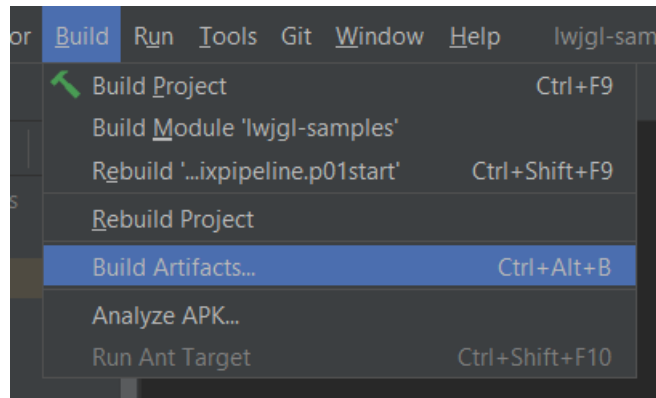


4. **Výsledek** vypadá zhruba takto:

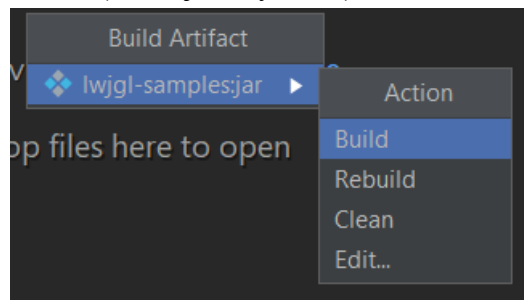




5. **V menu Build > Build Artifacts**



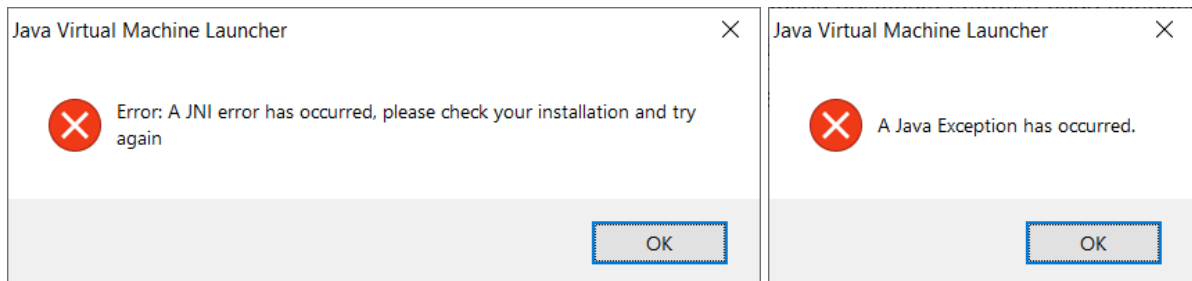
6. **V popup okně zvolit správný artefakt (může jich být více) a zvolit Build.**



7. **JAR soubor bude v cestě:** lwjgl-samples\out\artifacts\lwjgl\_samples.jar. Spustit je ho možné dvojitém poklikáním nebo pomocí příkazové řádky/terminálu.

Pozor na to, že JAR soubor nelze spustit pomocí nižší verze Javy, než ve které byl vytvořen. V takovém případě dostanete výjimku nebo chybové okno.

Chybové okno:



Výjimka (pokus o spuštění pomocí Javy 8 (52.0) aplikaci vytvořenou pomocí Javy 15 (59.0)):

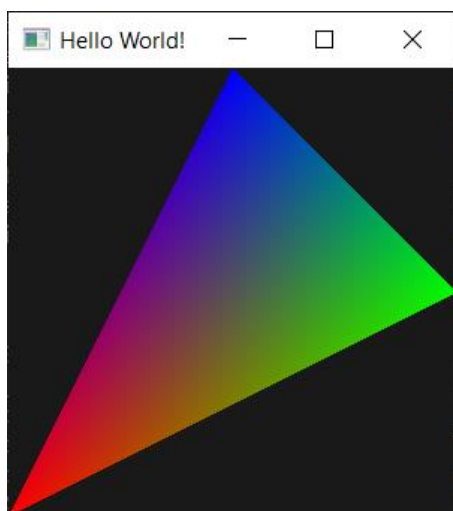
```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19041.804]
(c) 2020 Microsoft Corporation. Všechna práva vyhrazena.

D:\Java\IdeaProjects\lwjgl-samples\out\artifacts\lwjgl_samples.jar>java -jar lwjgl-samples.jar
Error: A JNI error has occurred, please check your installation and try again
Exception in thread "main" java.lang.UnsupportedClassVersionError: lvl0fixpipeline/p00helloWorld/HelloWorld has been compiled by a more recent version of the Java Runtime (class file version 59.0), this version of the Java Runtime only recognizes class file versions up to 52.0
    at java.lang.ClassLoader.defineClass1(Native Method)
    at java.lang.ClassLoader.defineClass(ClassLoader.java:763)
    at java.security.SecureClassLoader.defineClass(SecureClassLoader.java:142)
    at java.net.URLClassLoader.defineClass(URLClassLoader.java:468)
    at java.net.URLClassLoader.access$100(URLClassLoader.java:74)
    at java.net.URLClassLoader$1.run(URLClassLoader.java:369)
    at java.net.URLClassLoader$1.run(URLClassLoader.java:363)
    at java.security.AccessController.doPrivileged(Native Method)
    at java.net.URLClassLoader.findClass(URLClassLoader.java:362)
    at java.lang.ClassLoader.loadClass(ClassLoader.java:424)
    at sun.misc.Launcher$AppClassLoader.loadClass(Launcher.java:349)
    at java.lang.ClassLoader.loadClass(ClassLoader.java:357)
    at sun.launcher.LauncherHelper.checkAndLoadMain(LauncherHelper.java:495)
```

Cestu je v takovém případě nutné vynutit. Například:

"C:\Program Files\Java\jdk-15.0.2\bin\java.exe" -jar lwjgl-samples.jar

8. Správně spuštěná aplikace.



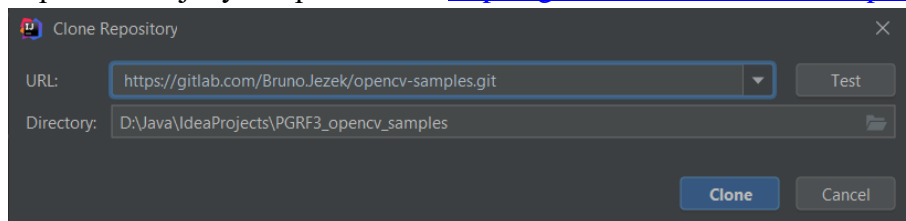
## F. Návod na instalaci OpenCV a zprovoznění ukázek v prostředí IntelliJ IDEA

1. **Stáhněte OpenCV** z následujícího webu: <https://opencv.org/releases/>, vyberte build pro vaši platformu
2. Stažený soubor rozbalte do požadované složky. Výsledek by měl vypadat takto:

» Tento počítač » Data (D:) » Java » lib » opencv-411 »

Název	Datum změny	Typ	Velikost
build	19.09.2019 12:58	Složka souborů	
sources	19.09.2019 12:59	Složka souborů	
LICENSE	26.07.2019 6:09	Textový dokument	3 kB
LICENSE_FFMPEG	26.07.2019 6:09	Textový dokument	28 kB
README.md	26.07.2019 6:09	Textový dokument	1 kB

3. **Proveďte kroky 1-6 z návodu** „Návod na instalaci LWJGL a zprovoznění ukázek v prostředí IntelliJ IDEA“. Liší se pouze zdrojovým repositářem - <https://gitlab.com/Bruno.Jezek/opencv-samples.git>



4. Stejným způsobem, jakým byla přidána knihovna LWJGL je **přidána i knihovna OpenCV**. Potřebný JAR soubor se nachází na adrese \build\java\opencv-420.jar
5. Dále je nutné upravit jednu třídu ručně. Konkrétně se jedná o opencvutils.VideoGrabber. V horní části této třídy se nachází tento řádek. Je nutné upravit cestu ke správnému souboru opencv\_videoio\_ffmpeg420 podle místa, kam jste OpenCV rozbalili.

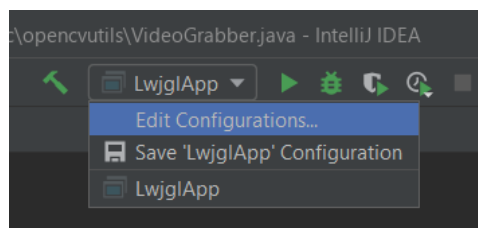
```
System.load("D:\\Java\\lib\\opencv-420\\build\\bin\\opencv_videoio_ffmpeg420_64.dll");
```

6. Nyní při spuštění nějaké ukázky bude docházet k následující výjimce:

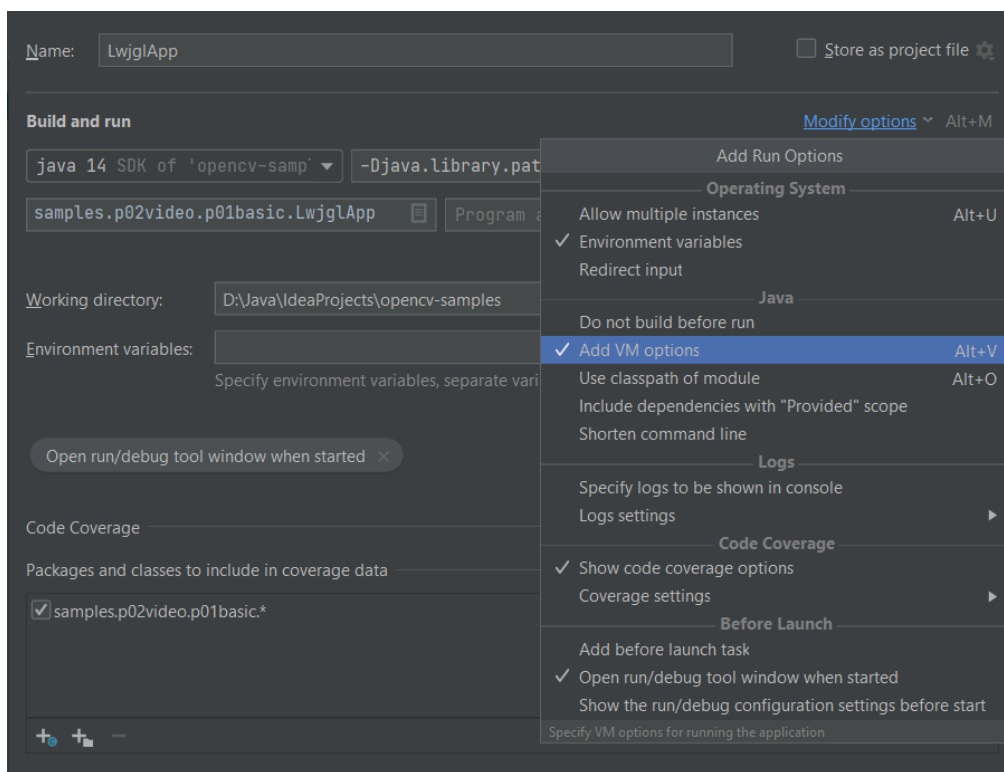
```
Exception in thread "main" java.lang.UnsatisfiedLinkError: no opencv_java420 in java.library.path
```

Problém se řeší přidáním parametru do „VM options“.

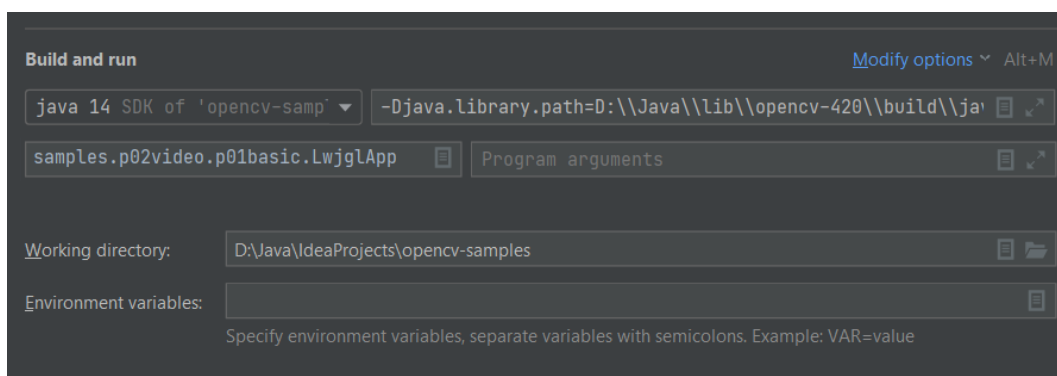
(-Djava.library.path=D:\\Java\\lib\\opencv-420\\build\\java\\x64\\). Znovu je nutné cestu upravit podle vaší instalace.



Zobrazení pole pro zadání parametrů:



Zadání parametru:



7. To je vše. OpenCV ukázky by nyní měly jít spustit.

## G. Užitečné odkazy

Setting up a JogAmp project in your favorite IDE

[http://jogamp.org/wiki/index.php/Setting\\_up\\_a\\_JogAmp\\_project\\_in\\_your\\_favorite\\_IDE#Compile\\_and\\_run\\_your\\_project\\_from\\_the\\_command\\_line](http://jogamp.org/wiki/index.php/Setting_up_a_JogAmp_project_in_your_favorite_IDE#Compile_and_run_your_project_from_the_command_line)

Tutorial: A cross-platform graphical application using Java, OpenGL, and Eclipse

<https://wadeawalker.wordpress.com/2010/10/09/tutorial-a-cross-platform-workbench-program-using-java-opengl-and-eclipse/>

OpenGL API Documentation <http://docs.gl/>

OpenGL 4.5 Reference Pages <https://www.opengl.org/sdk/docs/man/>

OpenGL 4.5 Core Profile Specification <https://www.opengl.org/registry/doc/glspec45.core.pdf>

OpenGL Shading Language 4.50 Specification <https://www.opengl.org/registry/doc/GLSLangSpec.4.50.pdf>