CS321: Advanced Programming Techniques

Lab 01
**Due Date: January 21st**

# Lab 01: Setup and Hello World

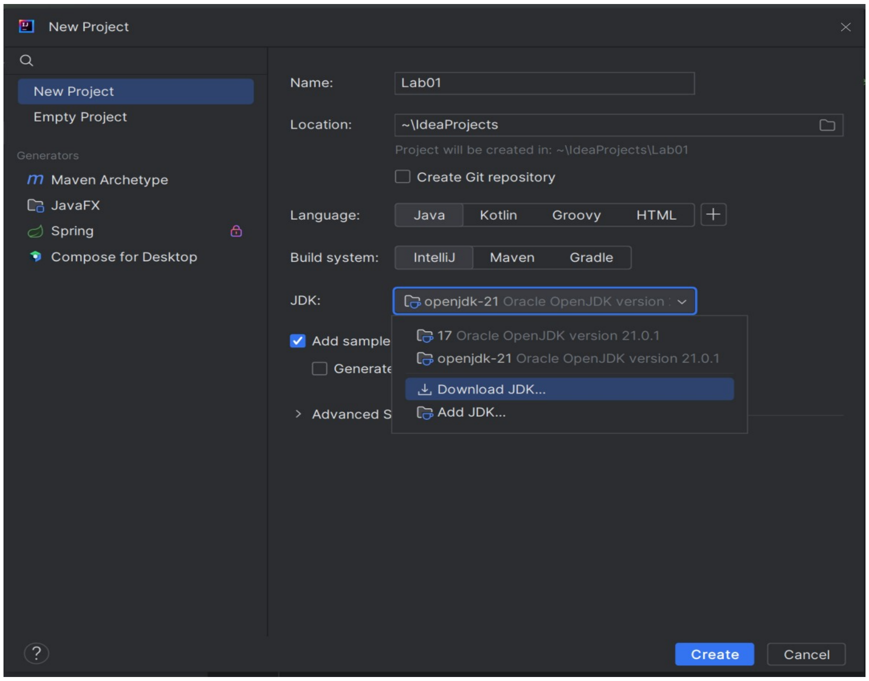
In this lab, you will only need to setup and run and submit a custom “Hello World” program on your own machine.

## Setup:

First, install the IntelliJ IDEA Community Edition from here: [https://www.jetbrains.com/idea/download/](https://www.jetbrains.com/idea/download/)

Next, you will need to create a new project:

- Open your new IDE.
- In the top-left corner click **File -> New -> Project…**
- Set the project SDK by clicking the dropdown menu and selecting **Download JDK**



New Project

Name: Lab01

Location: ~\IdeaProjects

Project will be created in: ~\IdeaProjects\Lab01

□ Create Git repository

Language: Java | Kotlin | Groovy | HTML

Build system: IntelliJ | Maven | Gradle

JDK: openjdk-21 Oracle OpenJDK version

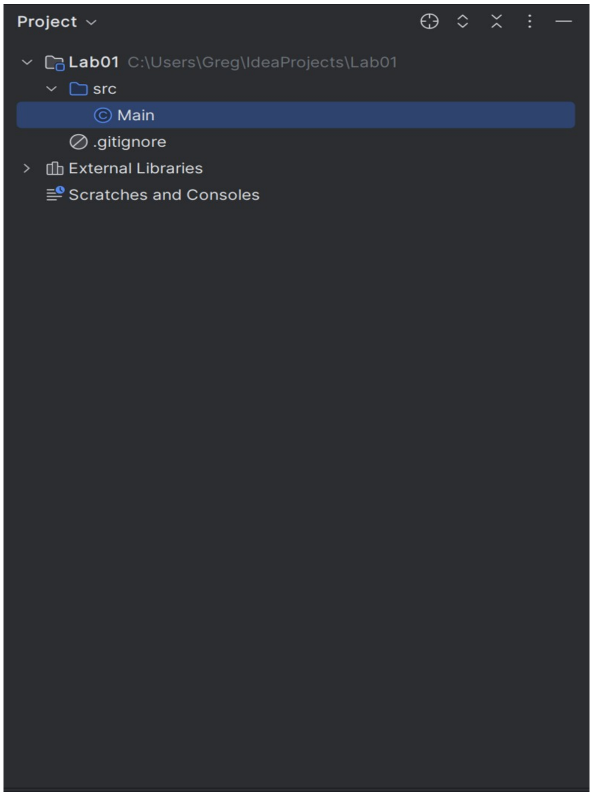
☑ Add sample

□ Generate

- In the window that pops open, select the Oracle OpenJDK version 21 (it may be near the bottom of the list)
- Name your project according to the guidelines for the course:
  - **First four letters** of your last name
  - **First three letters** of your first name
  - **Lab** if this is a Lab, **Assign** if this is an assignment

- ○ **XX** signifying the number of the work.
  - ○ **Example:** MierGre_Lab_01
- ➢ After this, your project will be created.

Now you have a project folder setup in your new IDE and your sidebar should look similar to this:



The **src** folder is where your code goes.

Start by making a new package in that folder by **right-clicking** and selecting **New -> Package**. Give the package the name **Lab**.

**Right-click** on the new package and select **New -> Java Class**. Give the new class a name.

A new file will be created in the folder with a template class inside.

# Hello World:

You’re all set to write your first Java Program now!

In this lab, you need to write a main function within the class you created above. This main function can do anything you want. Get creative and experiment with Java!

For those unsure of where to start, keep in mind that the basic syntax is nearly exactly the same as C++.

Your submission needs to run without errors or warnings and must output something through `System.out`:

- System.out.println
- System.out.print
- System.out.printf

To run, click the **play symbol** next to your **main** method.

Before you are ready to submit, you will need to build some documentation for the code you wrote:

- Add a JavaDoc-compatible comment for the main method.
- Select **Tools -> Generate JavaDoc…**
- In the new window, set the output directory to a new directory called **docs** inside of your project folder. It should reside at the same level as the **src** and **bin/out** directories.

When you’re finished experimenting and would like to submit your code do the following:

- Go over the submission guidelines on the next page to make sure that you are following them. Marks will be deducted if they are not followed.
- Update your JavaDocs!
- **Find your project folder through any file explorer, and compress it into a Zip.**
- Keep the project name as the name of the Zip file.
- Send me the project by email or Moodle (if it’s setup).

# Submission Guidelines:

These guidelines will be used for all assignments and labs.

- All submitted projects must be formatted as follows:
  - **First four letters** of your last name
  - **First three letters** of your first name
  - **Lab** if this is a Lab, **Assign** if this is an assignment
  - **XX** signifying the number of the work.
  - **Example:** MierGre_Lab_01

- Use the default formatting your IDE provides or follow the Java Code Style: https://www.oracle.com/java/technologies/javase/codeconventions-contents.html

- Use JavaDoc-compatible comments and always provide a **docs** folder with your submission:
  - Styling guide: https://www.oracle.com/technical-resources/articles/java/javadoc-tool.html
  - Select **Tools -> Generate JavaDoc…**
  - In the new window, set the output directory to a new directory called **docs** inside of your project folder.
  - It should reside at the same level as the **src** and **bin** directories.

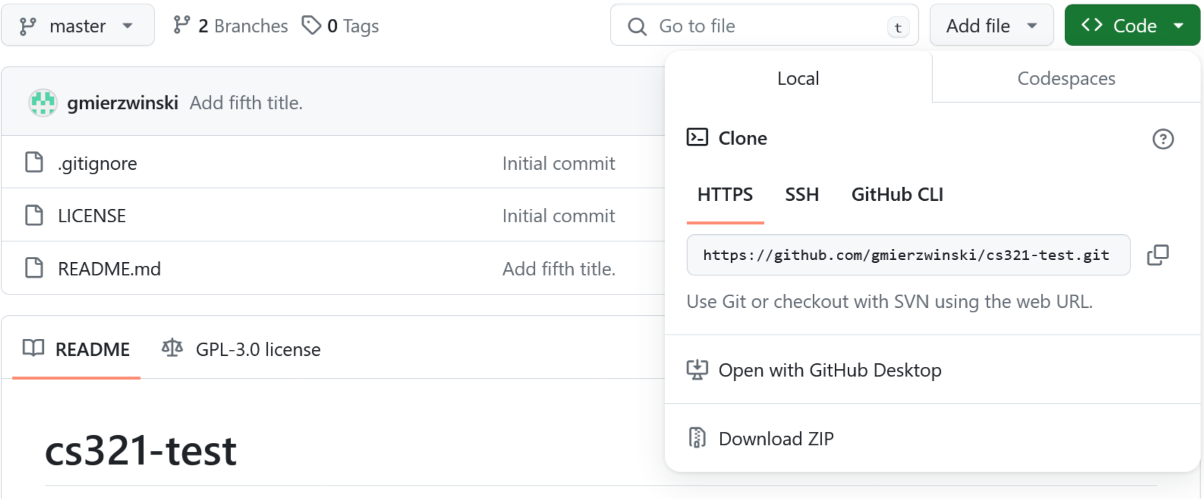
- The file that contains the main function must also contain a block comment like this at the very top:

```
/**
 * Name: Gregory Mierzwinski
 * Date: January 18th, 2022
 * Description: …
 */
```

- Other files must also contain a description such as this at the top, but they can refer back to the main description if there is nothing to describe.

- Submission exporting guidelines using a Zip file:
  - **Find your project folder through any file explorer, and compress it into a Zip.**
  - Keep the project name as the name of the Zip file.
  - Send me the project to my email or Moodle (if it's setup).

- Submission exporting guidelines using Github:
  - Install Git
  - Click **VCS -> Create Git Repository..**
  - Select the folder that contains your project and click OK
  - Click **Git -> Github -> Share Project on Github**
  - Keep the project name as the name of the repository, **keep it set as private**.
  - Login to Github through the same window.
  - After pressing **Share** a page will open up with the new repository.
  - Download the zip of the repository from the **Code** dropdown:



```
https://github.com/gmierzwinski/cs321-test.git
```

  - Send that zip to me by email

- If anything is unclear, please don't hesitate to ask questions.

## Grading Criteria:

| | |
|---|---|
| **Comments, Formatting, & Readability** | **5 Marks** |
| **Submission Guidelines** | **5 Marks** |
| **Program** | **5 Marks** |
| **Total** | **15 Marks** |