LAB1 – JAVA TOOLS AND IDE

COMP-2021: Object-Oriented Programming

Preparation

- Install JDK 11 and register an Oracle Account.

  (Please refer to page 4)

https://www.oracle.com/cn/java/technologies/javase/javase-jdk11-downloads.html

- Download TOOLBOX and install Intellij IDEA 2021.2

  (Please refer to page 12)

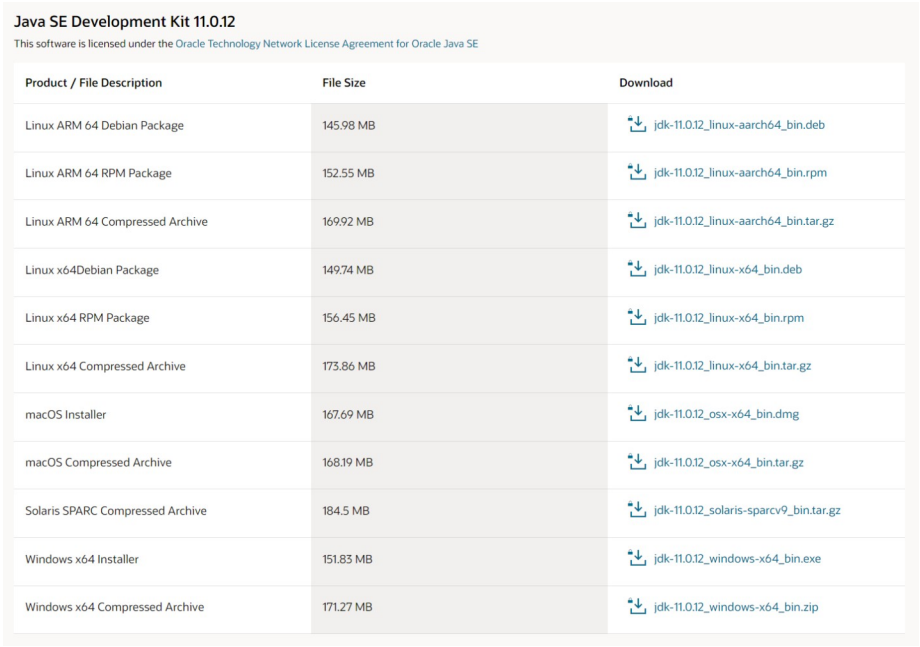
https://www.jetbrains.com/help/idea/2021.2/installation-guide.html#toolbox

Objectives

- To get started with IDE - Integrated Development Environment
  - To download and install JDK 11 and IntelliJ IDEA 2021.2
- To create your first Java program
  - To practice three steps (Edit-Compile-Run) of programming in JAVA
  - To get familiar with the basic command line arguments of Java applications
  - To utilize the debugger

Download JDK 11

- Go to https://www.oracle.com/cn/java/technologies/javase/javase-jdk11-downloads.html



Java SE Development Kit 11.0.12

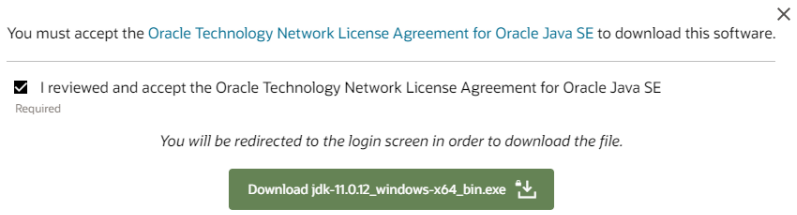
This software is licensed under the Oracle Technology Network License Agreement for Oracle Java SE

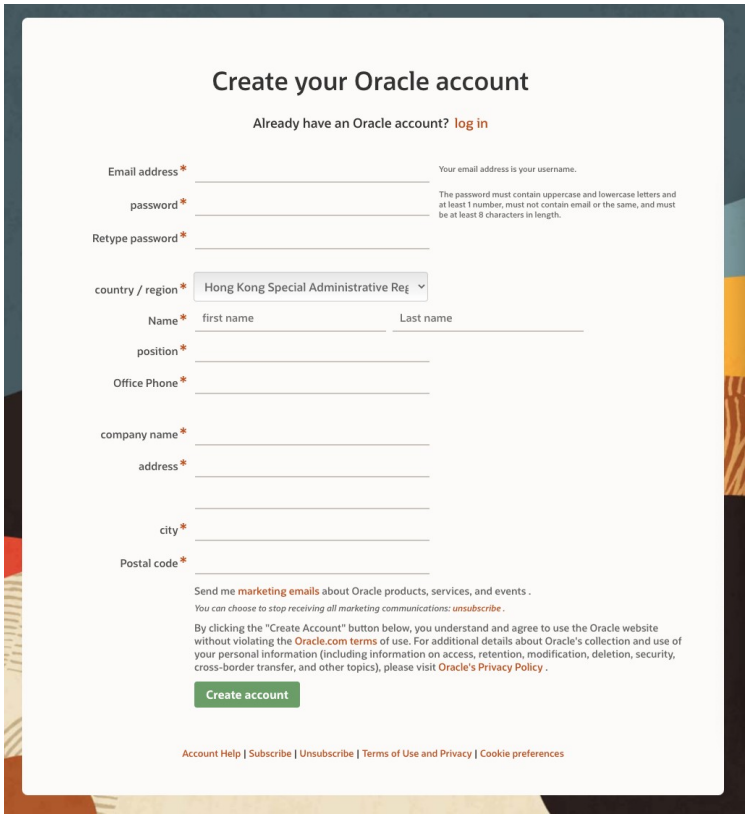
Product / File Description	File Size	Download
Linux ARM 64 Debian Package	145.98 MB	jdk-11.0.12_linux-aarch64_bin.deb
Linux ARM 64 RPM Package	152.55 MB	jdk-11.0.12_linux-aarch64_bin.rpm
Linux ARM 64 Compressed Archive	169.92 MB	jdk-11.0.12_linux-aarch64_bin.tar.gz
Linux x64Debian Package	149.74 MB	jdk-11.0.12_linux-x64_bin.deb
Linux x64 RPM Package	156.45 MB	jdk-11.0.12_linux-x64_bin.rpm
Linux x64 Compressed Archive	173.86 MB	jdk-11.0.12_linux-x64_bin.tar.gz
macOS Installer	167.69 MB	jdk-11.0.12_osx-x64_bin.dmg
macOS Compressed Archive	168.19 MB	jdk-11.0.12_osx-x64_bin.tar.gz
Solaris SPARC Compressed Archive	184.5 MB	jdk-11.0.12_solaris-sparcv9_bin.tar.gz
Windows x64 Installer	151.83 MB	jdk-11.0.12_windows-x64_bin.exe
Windows x64 Compressed Archive	171.27 MB	jdk-11.0.12_windows-x64_bin.zip

Download file according to your own Operating system.

Download JDK 11

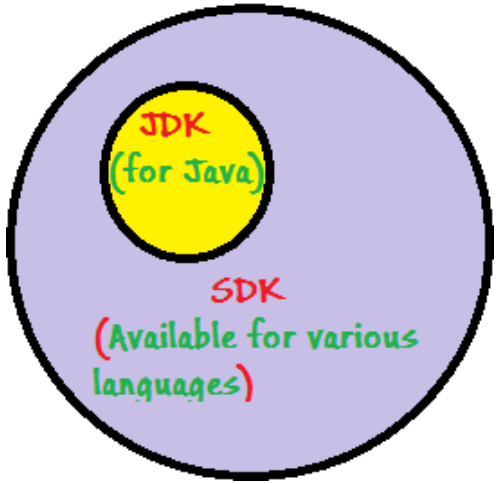
To download this file, you need to register an Oracle Account.





JDK and SDK

- SDK – Software Development Kit
  - enables one to write the code with more more ease, e ffectiveness and efficiency.
  - SDKs come for various languages.



The SDK for Java is called as JDK, the Java Development Kit. So by saying SDK for Java you are actually referring to the JDK.

Open the Command Prompt (a.k.a. Terminal)

- Windows
Pressing `WIN+R` and typing `cmd` on the pop-up window

-Mac
Select Applications from the left side.
Click the arrow to expand the Utilities folder.
Double-click Terminal

Check if Java works properly

Add the JDK bin directory to the system Path, e.g.,

- Windows

```
set PATH=%PATH%;C:\Program Files\Java\jdk-11.0.12\bin
```

- Mac OS

```
vi     ~/.bash_profile
export PATH=$PATH:/usr/java/jdk-11.0.12/bin
source ~/.bash_profile
```

* The JDK directory may be different on different machines.

Check if Java works properly on the machine by typing `java -version`. If the Java environment has been properly configured, you should see the version of Java.

* On lab computers,

1. Double click y:\AppsMenu.bat
2.  => Programming / Apps Development => Oracle JDK (path set already)

Create a Program

1. Create a `Main.java` file

- Windows
Create a `Main.java` file by typing following command in the Command Prompt

```
notepad Main.java
```

- Mac

Click on Applications and then TextEdit.
Click on TextEdit on the menu bar and select Preferences.
Select Plain Text.
Save as `Main.java`

2. Write the Java program in the text editor and save.

```
public class Main {
    public static void main(String[] args) {
        System.out.println("Hello World!");
    }
}
```

Compile the Program

Compile the program by

```
javac Main.java
```

`Main.java` is the source file we want to compile.
`javac` is the Java compiler. It takes the source file as the input and translates the Java code into instructions that the Java virtual machine can understand. The instructions are known as bytecode. During the compilation a file named `Main.class` is generated in the same directory.

*Navigate to the previous location before invoking javac to compile the program.

*Type `java` to see more usages of java command
*Type `javac` to see more usages of javac command

Step 3 Run the Program

- Run the program by typing

```
java Main
```

  You should see the output says `Hello World!`

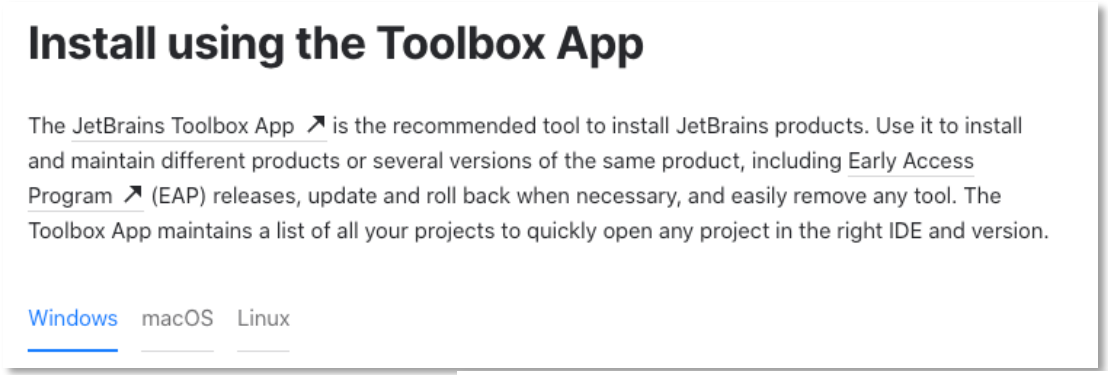
- The `java` command starts a Java application. It does so by starting a Java runtime environment, loading the specified class, and calling that class's main method.

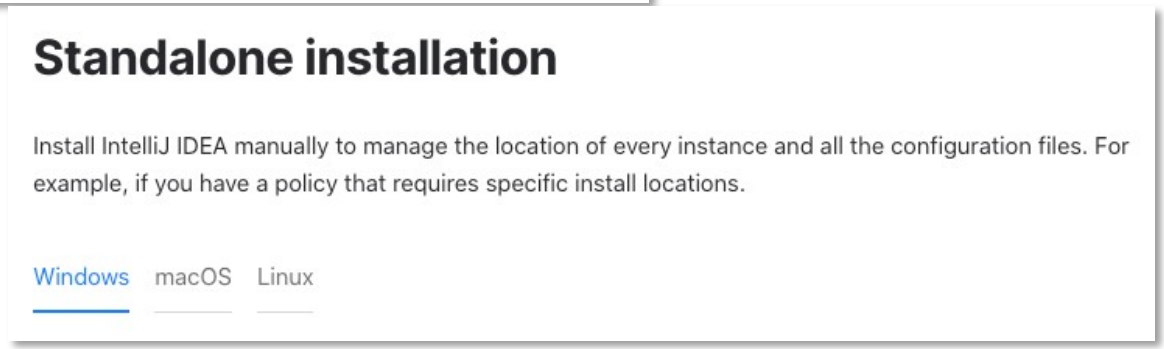
Exercise

Modify the given program to print your name and birth date.

Download and install Intellij IDEA 2021.2

- Go to https://www.jetbrains.com/help/idea/2021.2/installation-guide.html#toolbox
- Install according to the guide.





JVM, JRE, and JDK

- JVM – Java Virtual machine
  - responsible for executing the java program
  - also known as interpreter
- JRE – Java Runtime Environment
  - an installation package which provides environment to **only run** (not develop) the java program
  - consists of the JVM, core classes, and supporting files
  - is used by end users
- JDK – Java Development Kit
  - provides the environment to **develop and execute(run)** the Java program
  - includes JRE and Development Tools
  - is used by Java Developers.

```
JRE = JVM + Library Classes
JDK = JRE + Development Tools
```

* For more details, refer to https://www.geeksforgeeks.org/differences-jdk-jre-jvm

Getting Started with the IntelliJ IDEA

- Create a project
  - Setup JDK and SDK
  - Edit a run/debug configuration
- Debug a program

Getting Started with the IntelliJ IDEA

Open the IntelliJ IDEA:

* On lab computers,

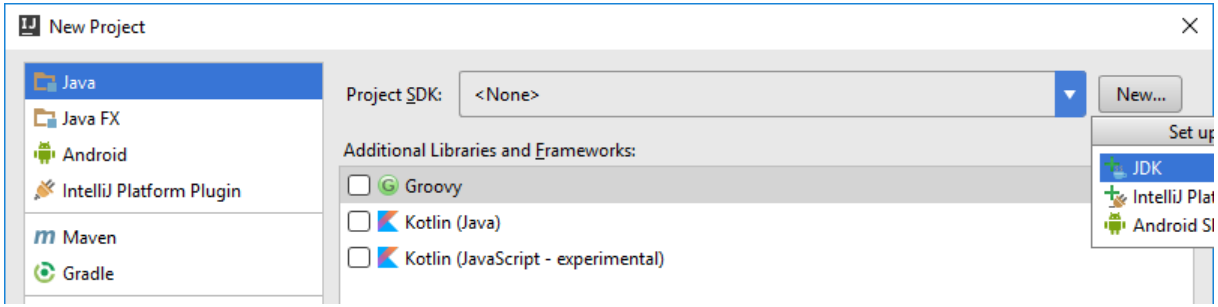
1. Double click y:\AppsMenu.bat
2.  => Programming / Apps Development => IntelliJ IDEA)

If it is the first time opening the IntelliJ, click on `Don't send` | `Evaluate for free`

Create a project

Click `Create New Project` on the Welcome screen.

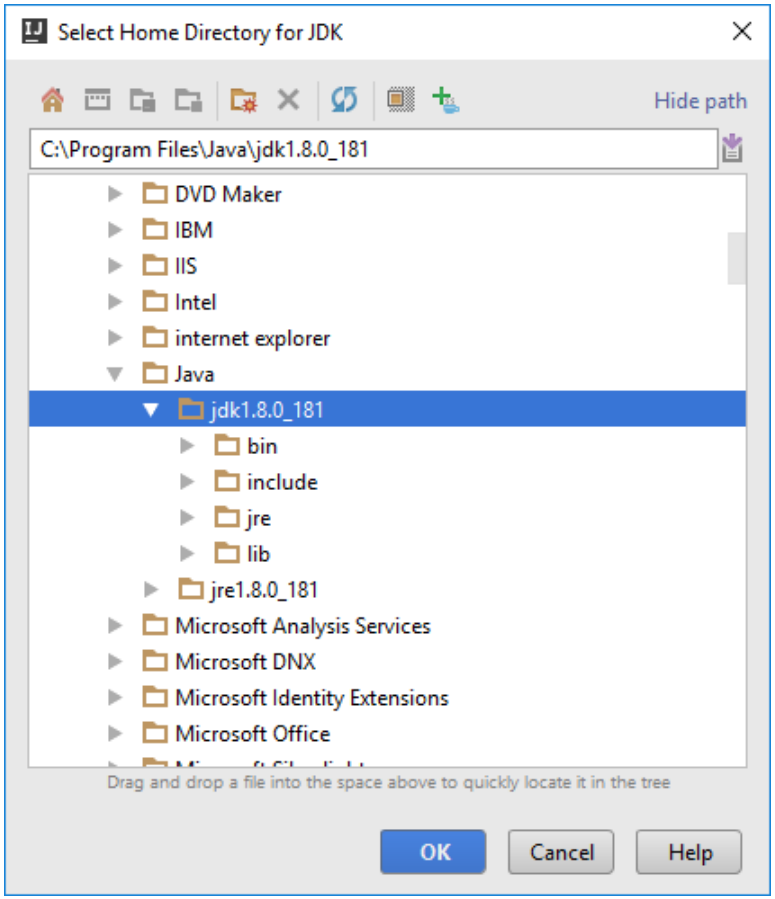
(Optional) Create a project



- In the left-hand side panel, select `Java`.
- In the Project SDK field, specify the JDK for your project by one of the option:
  - Select the JDK from the `Project SDK` list.
  - Click `New` to setup a JDK when the JDK is already available on your computer but not show on the list.

(Optional) Setup the JDK

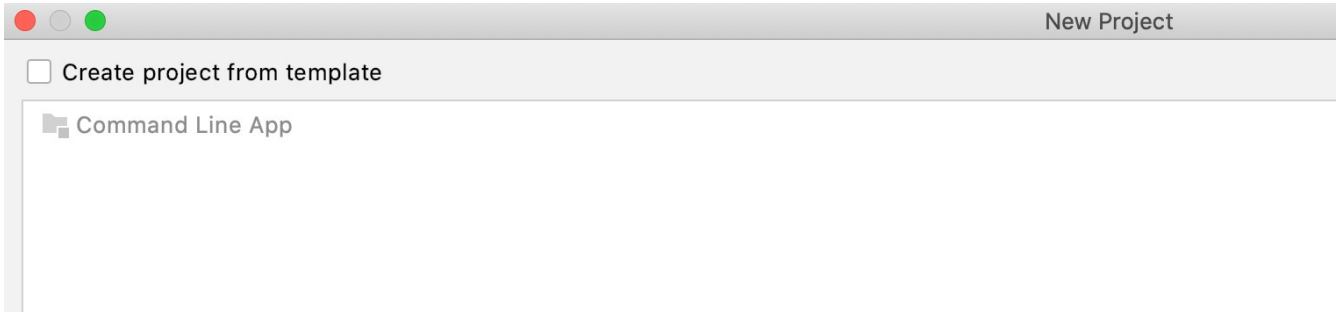
In the dialog that opens, select the JDK installation directory. Note that the JDK installation directory on a lab computer is where you installed the JDK.
For example: `C:\Program Files\Java\jdk-11.0.12`



* On lab computers,
Y:\APPs\OracleJDK\jdk-11.0.12

(Continue) Create a project

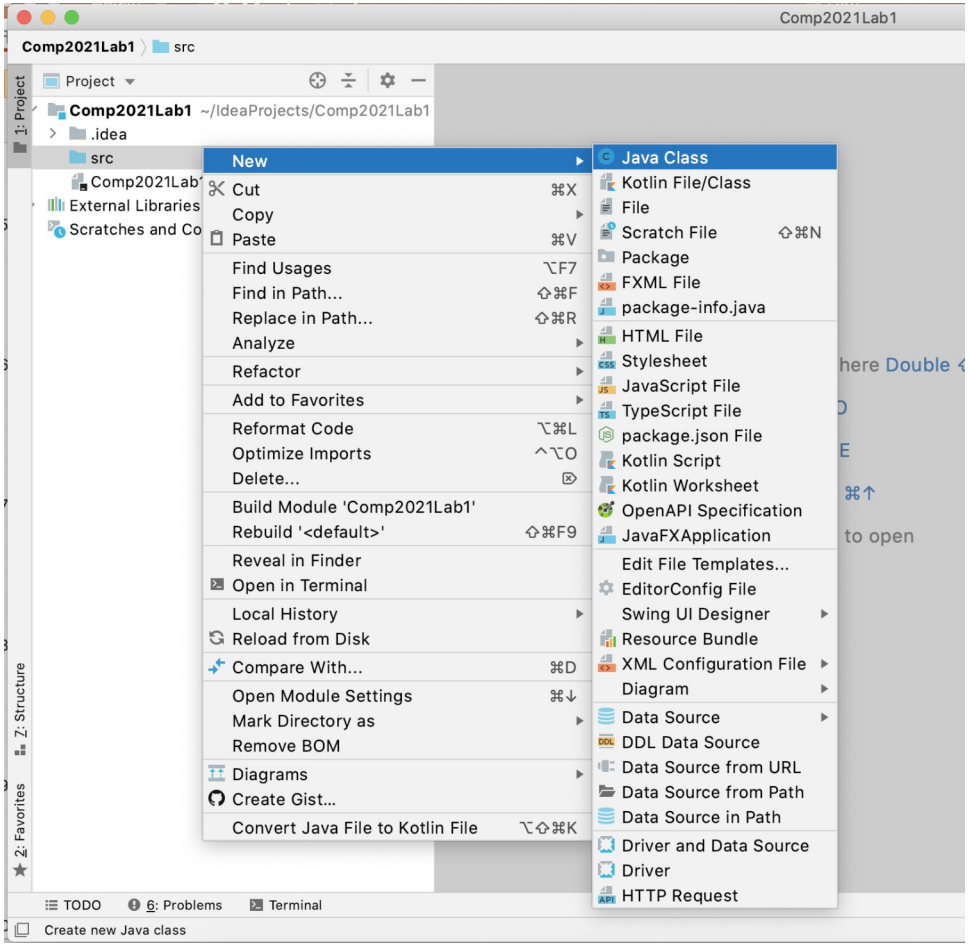
- The IntelliJ IDEA provides basic project templates. To create an empty project, select nothing and click on `Next`.



- On the next page, specify the project name (e.g. Comp2021Lab1). If necessary, change the project location.
- Click `Finish`. The IntelliJ IDEA creates the project and prepare the development kits automatically. When the creation process is complete, the structure of your new project is shown in the `Project` tool window.

Create a Java class

Right click on `src` folder of your project, then click on `New` | `Java Class`. Then give a name to the class.



Example

The source code shown below is a Java class named `Main`. The class contains two methods named `main` and `getMaxElement` respectively. The `getMaxElement` method* takes an integer array as input and is supposed to output the maximum value in the input array.

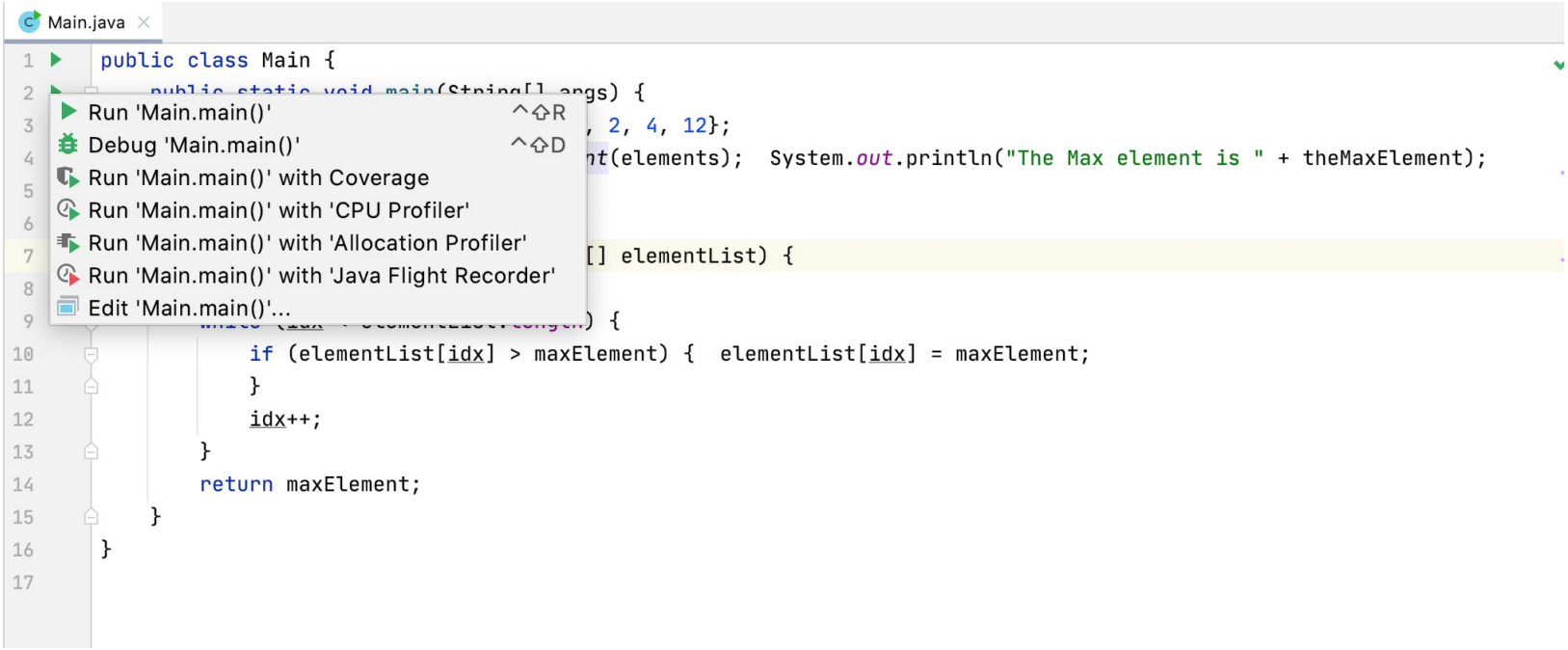
```
public class Main {
    public static void main(String[] args) {
        int[] elements = {5, 3, 7, 8, 9, 2, 4, 12};
        int theMaxElement = getMaxElement(elements);
        System.out.println("The Max element is " +
        theMaxElement);
    }

    public static int getMaxElement(int[] elementList) {
        int maxElement = 0, idx = 0;
        while (idx < elementList.length) {
            if (elementList[idx] > maxElement) {
                elementList[idx] = maxElement;
            }
            idx++;
        }
        return maxElement;
    }
}
```

* There is a seeded bug in the `getMaxElement` method for you to practice the debugtool.

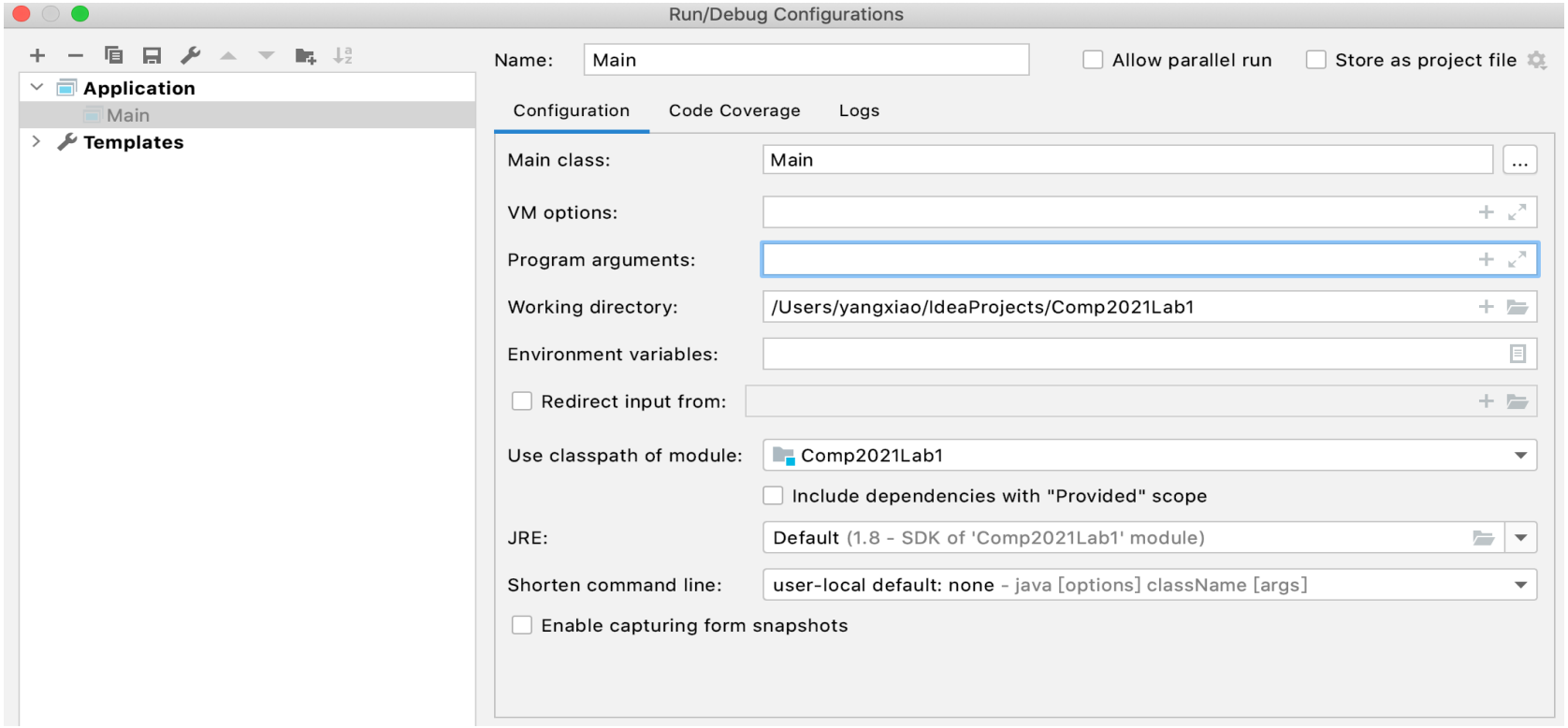
Run the program

- Click on the ▶ button on the left gutter area where the `main` function is declared. Then IntelliJ will automatically create a default Run/Debug Configuration and execute the Java application.



(Optional) Edit the run/debug configuration

- Open the Run/Debug Configuration dialog: Choose `Run` | `Edit Configurations` from the main menu.
- Select `Application` and setup your `main class` and `program arguments`.



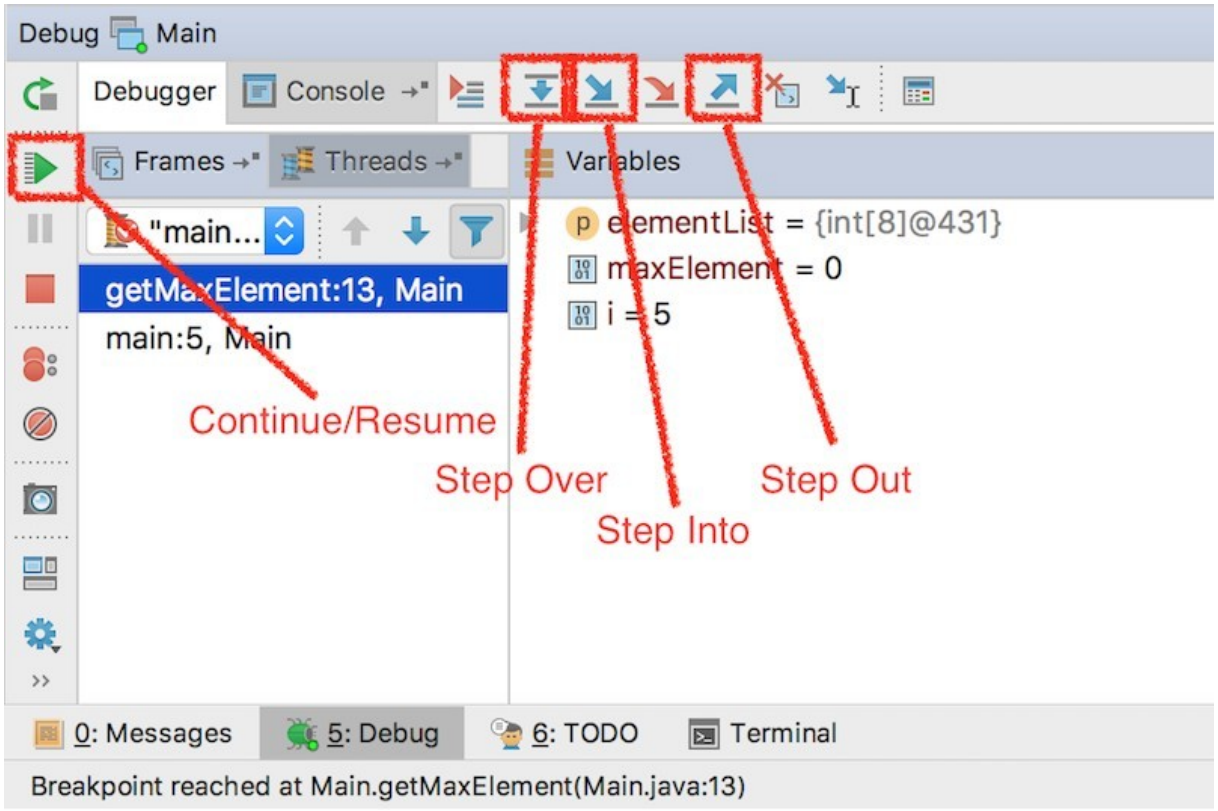
* Creating and Editing Run/Debug Configurations -https://www.jetbrains.com/help/idea/run-debug-configuration.html

Debug A Program

- Line Breakpoint.

  A line breakpoint is a breakpoint assigned to a specific line in the source code. If an application runs on debug mode, the program will suspend if it hit a breakpoint. During the suspension, we can check the program runtime states at the break point and control the execution using the following Commands:

Step Command	Description
Continue / Resume	Advance to the next breakpoint or to the end of the program, whichever comes first.
Step Over	Advance to the next line in the current method, stepping over any lines that invoke other methods.
Step Into	Advance to the next line in the program.
Step Out	Advance to the next line in the method that invoked the current method.



- For more information, please refer to document Stepping toolbar.

* Stepping toolbar - https://www.jetbrains.com/help/idea/debug-tool-window.html#steptoolbar.

Utilize the Debugger

Step 1 - Create line breakpoints in the editor

Click on the left gutter area at a line where you want to toggle a breakpoint. You will see  after click.

Step 2 - Start the debug session

Click  icon in the left gutter, and then choose  .

Step 3 - Control the execution

When a breakpoint is hit, the execution of the program is suspended. You can now

- Examine program states.
- Try out step commands.

Exercise

- Debug the given program and fix the bug to print the actual max element.

References

1. https://www.oracle.com/cn/java/technologies/javase/javase-jdk11-downloads.html
2. https://www.jetbrains.com/help/idea/2021.2/installation-guide.html#toolbox
3. https://docs.oracle.com/javase/tutorial/getStarted/cupojava/index.html
   https://www.jetbrains.com/help/idea/creating-and-running-your-first-java-application.html
4. https://www.jetbrains.com/help/idea/configuring-intellij-platform-plugin-sdk.html
5. https://www.geeksforgeeks.org/differences-jdk-jre-jvm/
   https://stackoverflow.com/a/36154746/6384650

* Oracle Java SE Development Kit Version 11 and IntelliJ IDEA Community Edition Version 2021.2 will be used in grading your assignments and project. Make sure you use the same versions of tools for your development in your local environment.