

Getting Started with IntelliJ

COMSM0086

Dr Simon Lock and Dr Sion Hannuna

Overview

- Downloading and installing IntelliJ IDE
- Dealing with User Agreement and Licensing
- Opening an existing template project
- Installing the Java Development Kit (JDK)
(including compiler, runtime and libraries)
- Running your first Java program !

Note: We will skim through these slides quite quickly now
Work through them at your own pace in practical session

Do you need to install ?

You don't HAVE to install IntelliJ on your laptop
The lab machines are fully installed and set up:

`/opt/idea/2024/bin/idea.sh`

License Server: <http://ls-jetbrains.bris.ac.uk:8080>

HOWEVER

Many people choose to work on their own machines
Just so they can work from home (and at any time)

If you do decide to install, here are a few tips...

Download from JetBrains

Make sure you get the download for your platform !
(Website *should* autodetect to the right version)

Windows macOS Linux



IntelliJ IDEA Ultimate

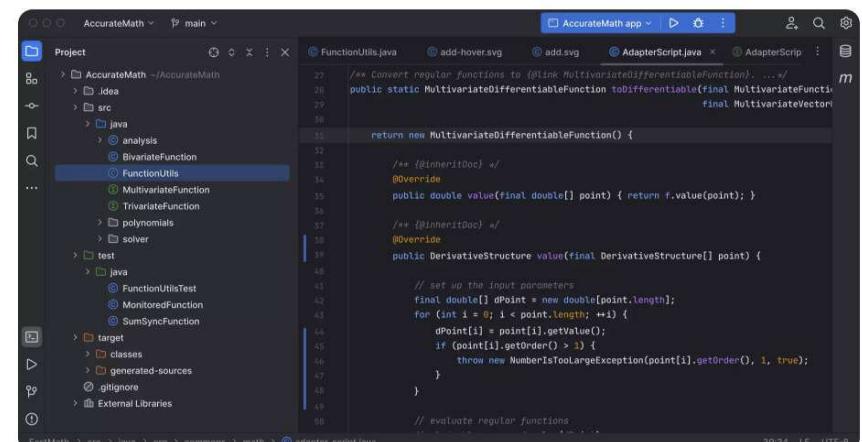
The Leading Java and Kotlin IDE

[Download](#)

.dmg ▾

Free 30-day trial

Select an installer for Intel or Apple Silicon



```
/*
 * Convert regular functions to {@Link MultivariateDifferentiableFunction} ...
 */
public static MultivariateDifferentiableFunction toDifferentiable(
    final MultivariateFunction f,
    final MultivariateVector point
) {
    return new MultivariateDifferentiableFunction() {
        /**
         * @inheritDoc
         */
        public double value(final double[] point) { return f.value(point); }

        /**
         * @inheritDoc
         */
        public DerivativeStructure value(final DerivativeStructure[] point) {
            // set up the input parameters
            final double[] dPoint = new double[point.length];
            for (int i = 0; i < point.length; ++i) {
                dPoint[i] = point[i].getValue();
                if (point[i].getOrder() > 1) {
                    throw new NumberIsTooLargeException(point[i].getOrder(), 1, true);
                }
            }
            // evaluate regular functions
        }
    };
}
```

Version: 2023.3.2

Build: 233.13135.103

20 December 2023

[System requirements](#)

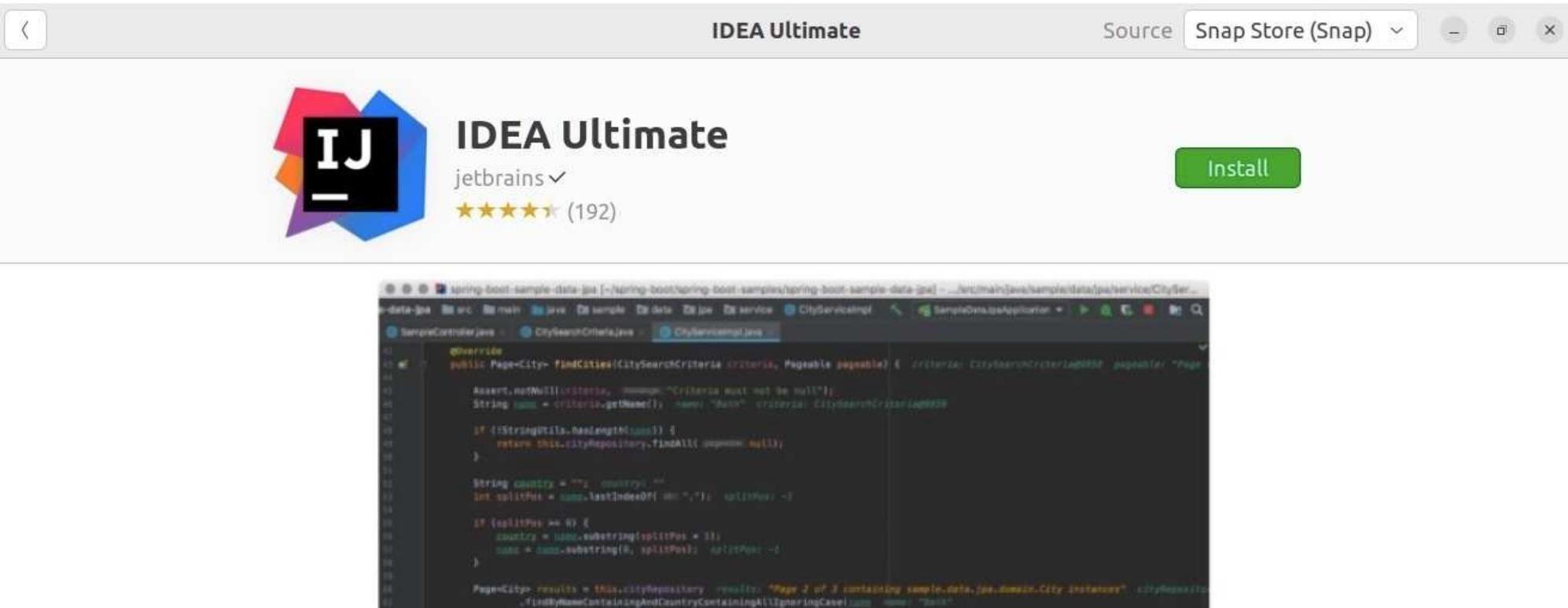
[Installation instructions](#)

[Other versions](#)

[Third-party software](#)

Alternative Approach

IntelliJ might be available through your platform's
Package Manager (if it has one !)



Install as normal (for your platform) !

User Agreement

First time you run IntelliJ, you'll see User Agreement
Tick the box and click continue if you are happy !

JETBRAINS USER AGREEMENT

Version 1.4, effective as of September 22, 2021

IMPORTANT! READ CAREFULLY:

THIS IS A LEGAL AGREEMENT. BY CLICKING ON THE "I AGREE" (OR SIMILAR) BUTTON THAT IS PRESENTED TO YOU AT THE TIME OF YOUR FIRST USE OF THE JETBRAINS SOFTWARE, SUPPORT, OR PRODUCTS, YOU BECOME A PARTY TO THIS AGREEMENT, YOU DECLARE YOU HAVE THE LEGAL CAPACITY TO ENTER INTO SUCH AGREEMENT, AND YOU CONSENT TO BE BOUND BY ALL THE TERMS AND CONDITIONS SET FORTH BELOW.

1. PARTIES

1.1. "JetBrains" or "we" means JetBrains s.r.o., having its principal place of business at Na Hrebenech II 1718/10, Prague, 14000, Czech Republic, registered in the Commercial Register maintained by the Municipal Court of Prague, Section C, File 86211, ID No. 265 02 275

I confirm that I have read and accept the terms of this User Agreement

Exit

Continue

Register for an Educational License

<https://jetbrains.com/community/education/#students>

Apply with	University email address ISIC/ITIC membership Official document
Status:	<input checked="" type="radio"/> I'm a student <input type="radio"/> I'm a teacher
Level of study	Undergraduate ▾
Is Computer Science or Engineering your major field of study?	
	<input checked="" type="radio"/> Yes <input type="radio"/> No
Email address:	University email address, e.g. js@mit.edu

I certify that the university email address provided above is valid and belongs to me.

License Settings

"Log In to JetBrains" if you have an educational license
"Start Trial" if you haven't yet registered with JetBrains

The screenshot shows the "License Settings" screen of the IntelliJ IDEA application. On the left, there's a sidebar with icons for "IntelliJ IDEA Activate", "AI Assistant Activate to enable", and "Code With Me Activate to enable". Below that is a "Log in..." button and "Proxy settings". On the right, the main area has a title bar with "Activate IntelliJ IDEA" (radio button selected), "Start trial" (radio button unselected), and a "View plans and pricing" link. Below this is a "Get license from:" section with three options: "JetBrains Account" (radio button selected), "Activation code" (radio button unselected), and "License server" (radio button unselected). There are "Log In to JetBrains Account..." and "Register..." buttons. At the bottom right is an "Exit" button.

IntelliJ IDEA
Activate

AI Assistant
Activate to enable

Code With Me
Activate to enable

Log in...

Proxy settings

Activate IntelliJ IDEA Start trial [View plans and pricing ↗](#)

Get license from:

JetBrains Account Activation code License server

[Log In to JetBrains Account...](#) [Register...](#)

Exit

Final Step for Educational License

Make sure you click the "Activate" button !

 **IntelliJ IDEA**
Active until October 1, 2024

 **AI Assistant**
Activate to enable

 **Code With Me**
Active until October 1, 2024

Plugins

 Simon Lock

Proxy settings Activate plugin to enable paid functions Continue

Activate IntelliJ IDEA Start trial [View plans and pricing ↗](#)

Get license from:

JetBrains Account Activation code License server

Active license: Licensed to Simon Lock, For educational use only
Subscription is active until 01/10/2024

Activate **Cancel** **Refresh license list**

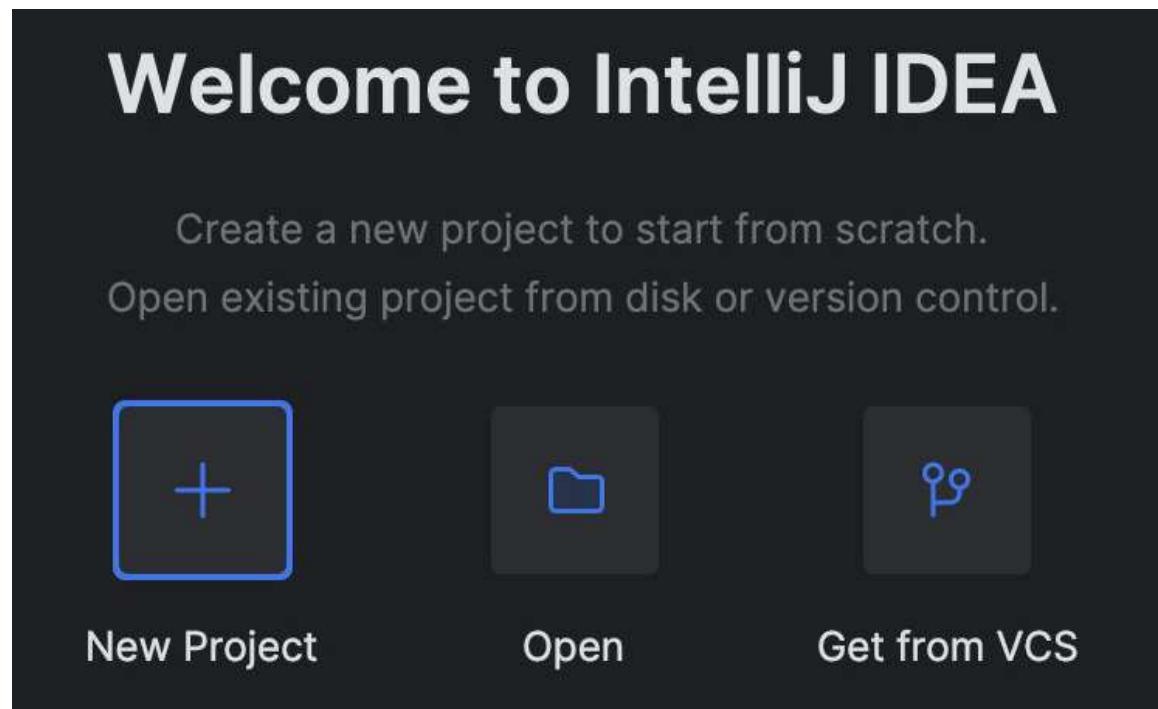


And that should provide you with
the same software as on the lab machines

Once you have the IDE installed...
How do we go about compiling and running code ?

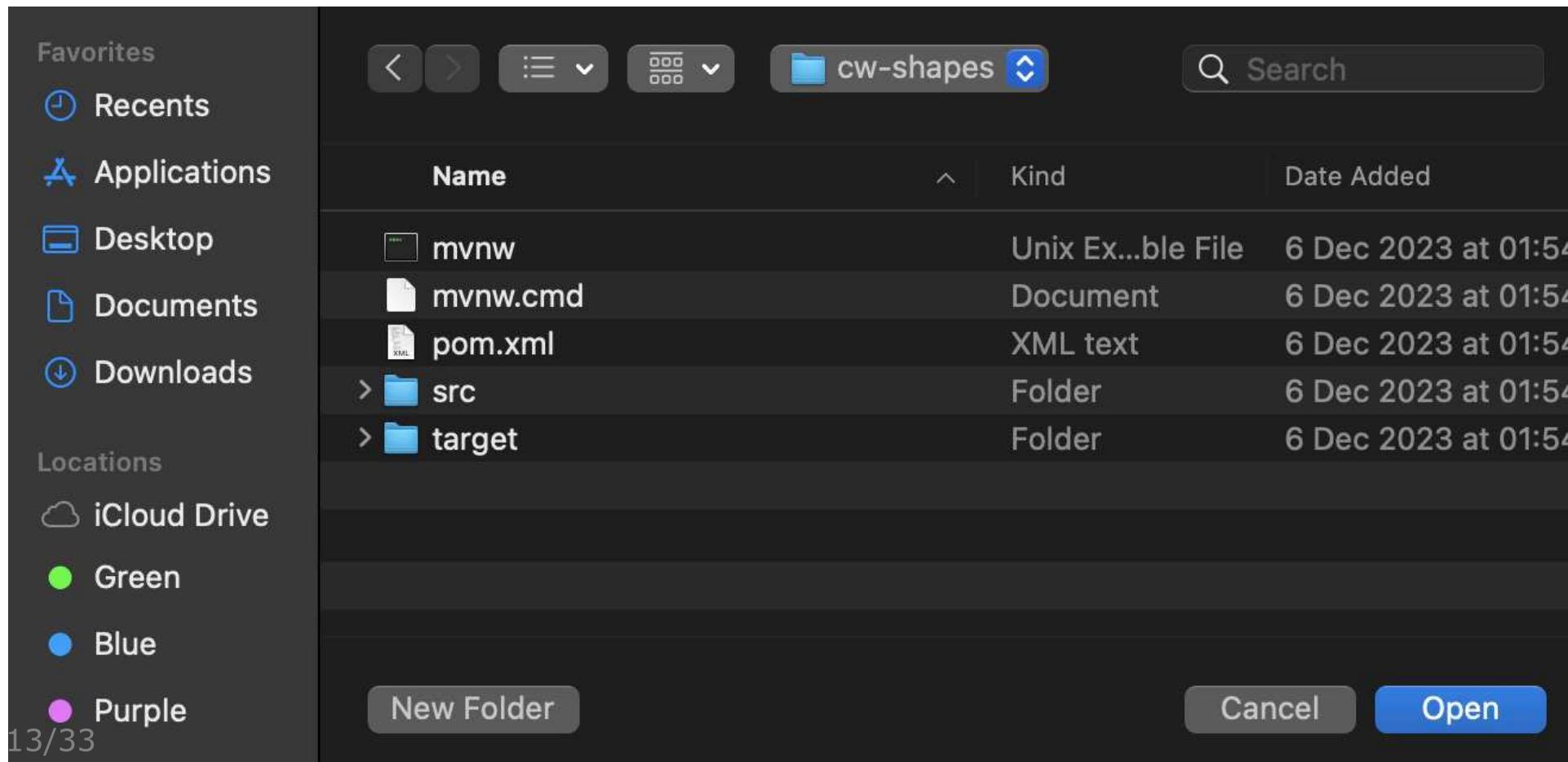
Welcome Screen

Most of the time on this unit you'll need "Open"
This is because we provide blank template projects
(Although you do get to practice creating a new project)



Typical Project Template

Find and open the FOLDER containing the pom.xml file



Do You Trust Us ?

Be careful with projects from other sources !



Trust and Open Project 'cw-shapes'?

IntelliJ IDEA provides features that may execute potentially malicious code from this folder.

If you don't trust the source, preview the project in the safe mode to only browse its code.

Trust projects in ~/Development/Weekly Workbooks/01 Introduction to OOP/IntelliJ Template

?

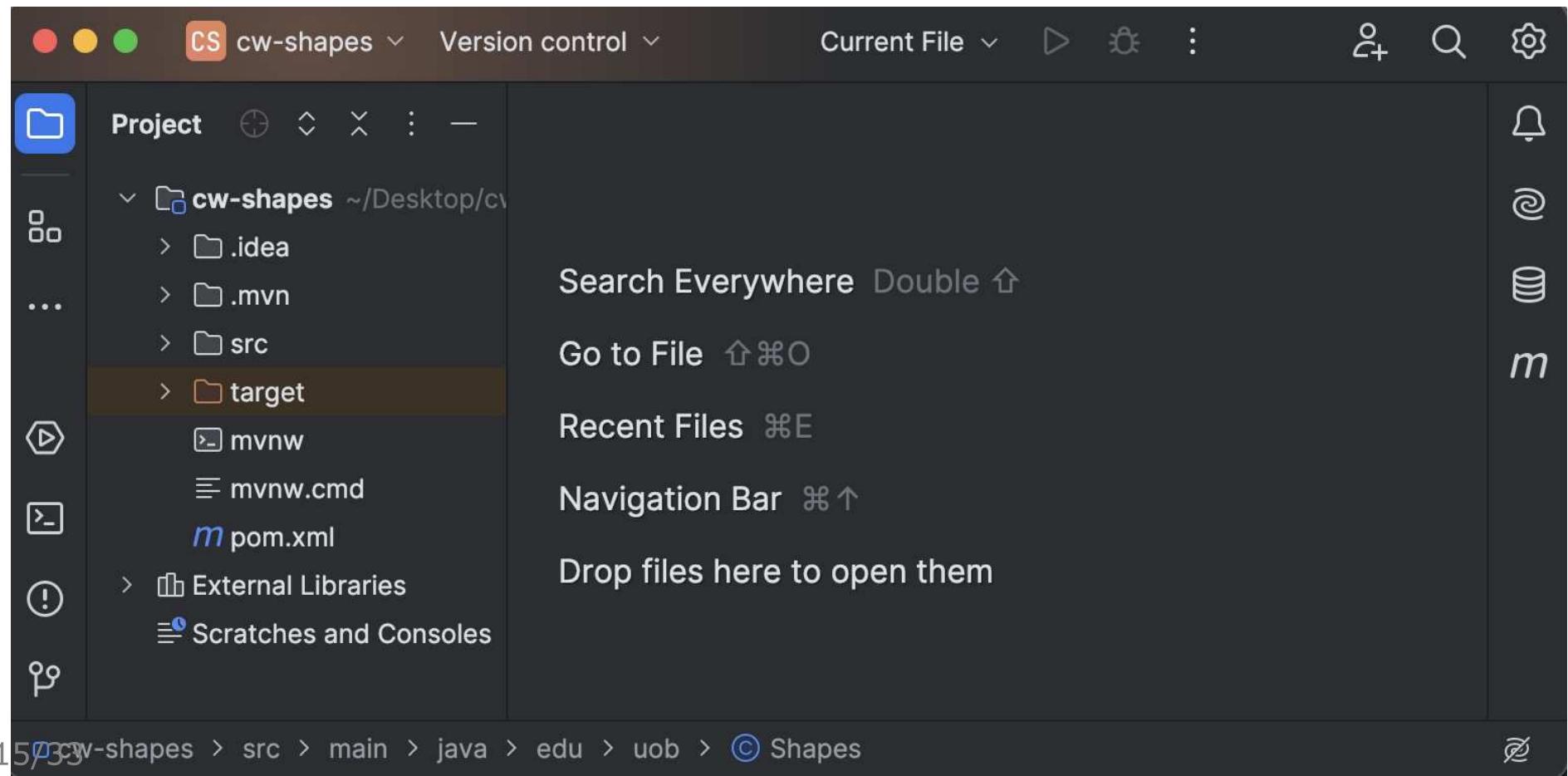
Don't Open

Preview in Safe Mode

Trust Project

Project Structure

A successfully opened project looks something like this:



Open the Main Class

Let's explore the project view to find the main class

In this project, the main class is a file called 'Shapes'

The screenshot shows a Java project named "cw-shapes" with the following structure:

- Project: cw-shapes (~/Desktop/cw-shapes)
- src folder
 - main folder
 - java folder
 - edu.uob folder
 - Circle.java
 - Shapes.java
 - Triangle.java

The "Shapes.java" file is open in the editor. The code is as follows:

```
1 package edu.uob;
2
3 public class Shapes {
4
5     // TODO use this class as then entry point
6     public static void main(String[] args) {
7
8
9     }
10}
```

"Project JDK is not defined"

IntelliJ is just an IDE - it has no built-in compiler !
Lab machines have Java Development Kit installed
If working on your own laptop, YOU must install it

The screenshot shows the IntelliJ IDEA interface. On the left is the Project tool window with several Java files listed: Circle, Colour, MultiVariantShape, Rectangle, Shapes (which is selected and highlighted in blue), Triangle, TriangleVariant, and TwoDimensionalShape. In the center is the code editor showing a Java file named Shapes.java. The code contains the following:

```
1 package edu.uob;
2
3 public class Shapes {
4
5     // TODO use this class as then ...
6     public static void main(String[] args) {
7         System.out.println("Hello, World!");
8     }
9 }
```

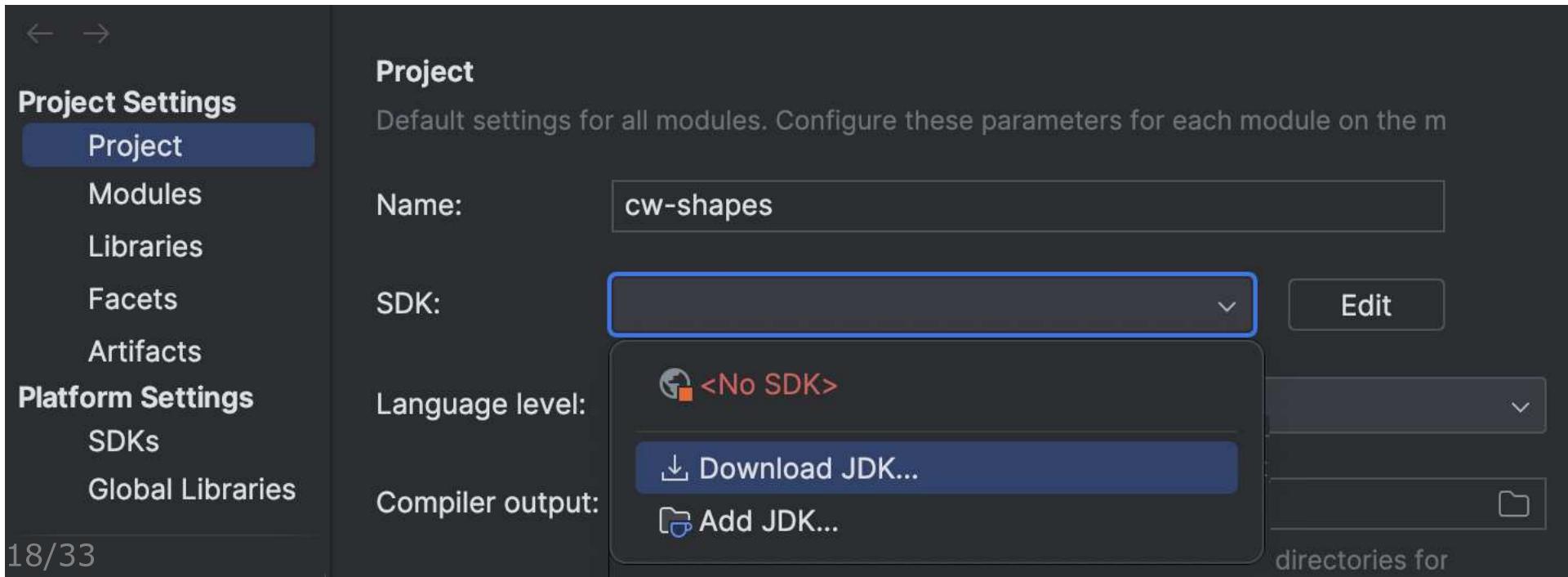
A yellow warning icon with the text "Project JDK is not defined" is displayed above the code editor. To the right of the warning are two buttons: "Setup SDK" and a dropdown menu. The status bar at the bottom shows "File | Open | Help | Exit".

Project Settings (File > Project Structure !!!)

Need to select a JDK to use to compile & run project

Could use an existing JDK (if you have one installed)

Or download a new one from list of those available...



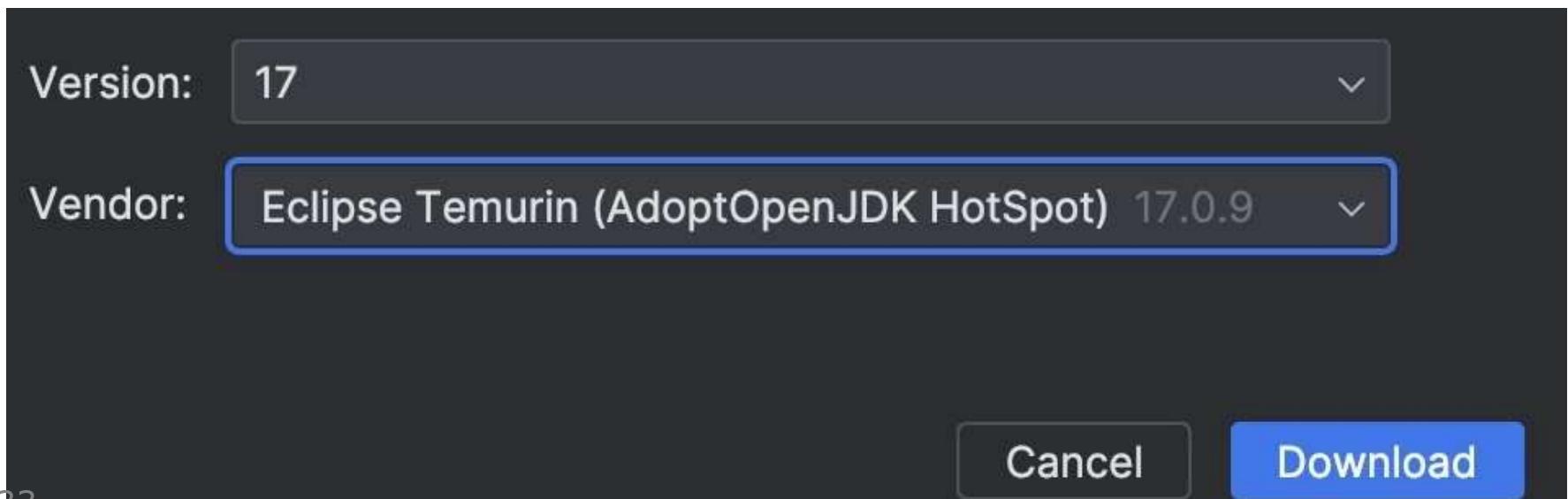
Download JDK

Select JDK to automatically download & install

Lowest Common Denominator: Lab has Java 17

(That's where we are going to mark your code !)

Best to choose: Eclipse Temurin (AdoptOpenJDK)

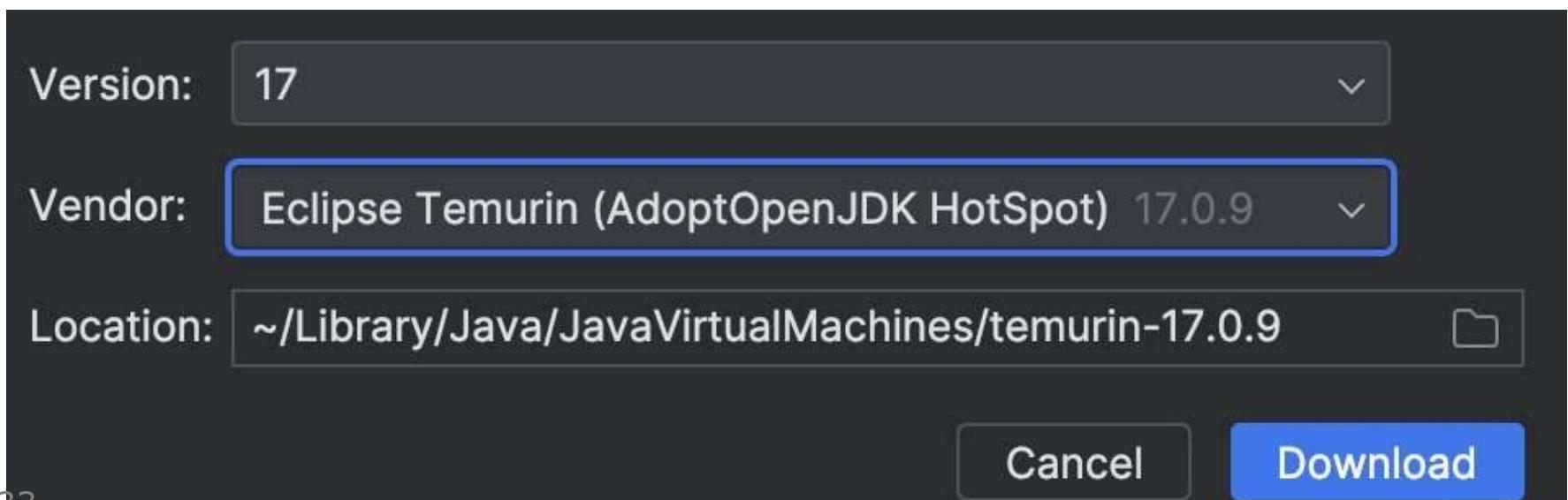


Installation Location

Keep a note of location where IntelliJ will install JDK
You'll need this later (to compile on command line)

On Mac OSX the location will be something like:

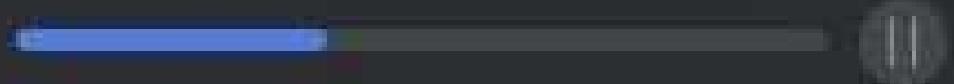
`~/Library/Java/JavaVirtualMachines/temurin-17.0.9`



Be Patient !

It takes a while for the JDK to download and install
Don't worry, only happens once (when you first install)
Keep an eye on how things are going on progress bar:

Indexing JDK 'temurin-17'



Using an Existing JDK

If you already have a JDK installed, you can use that
This will save you having multiple versions installed
It *might* be listed in dropdown, if not click "Add JDK"

← →

Project Settings

Project (selected)

Modules

Libraries

Facets

Artifacts

Platform Settings

SDKs

Global Libraries

Project

Default settings for all modules. Configure these parameters for each module on the module page.

Name: cw-shapes

SDK: (dropdown menu open)

<No SDK>

Download JDK...

Add JDK...

Compiler output:

directories for

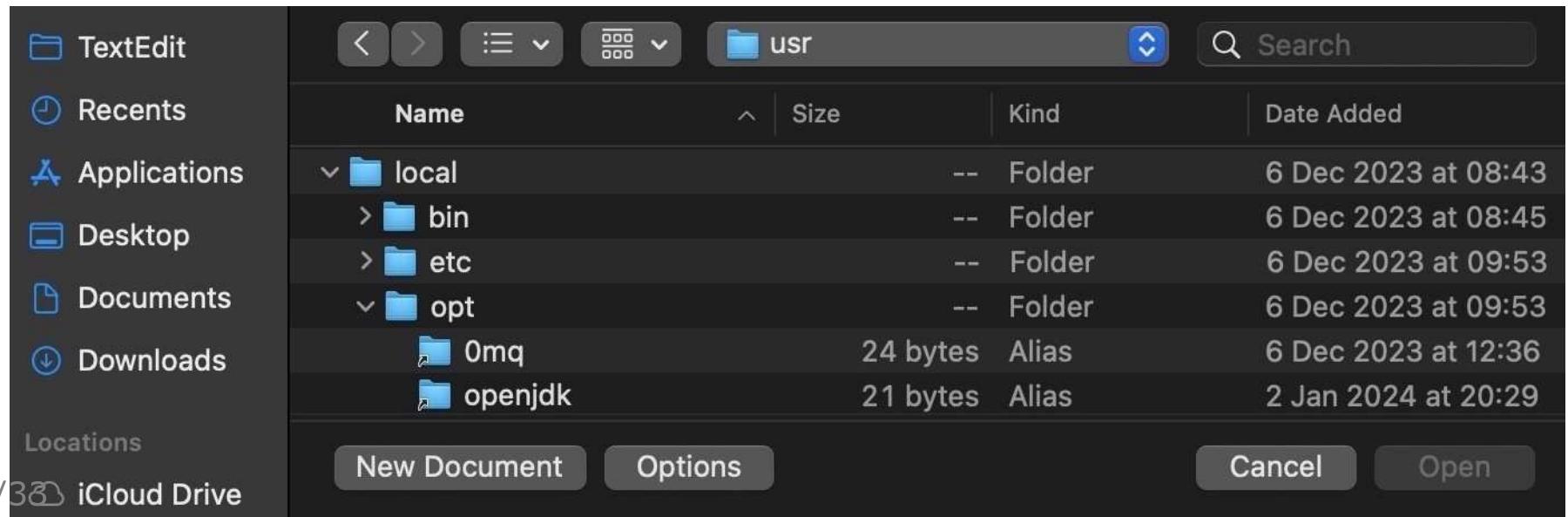
Hunting down Existing JDK

IntelliJ is clever and will usually find installed JDKs

I installed JDK using 'Homebrew' package manager

Homebrew is good at hiding installed software ;o)

So I needed to manually hunt around to find the JDK



JDK Version

The version I have installed is more recent than 17
However, we can limit language features used to 17
Prevents use of newer features that won't work in lab

Project

Default settings for all modules. Configure these parameters for each module on the module page as needed.

Name:

cw-shapes

SDK:

21 version 21.0.1

Edit

Language level:

17 - Sealed types, always-strict floating-point semantics

REMEMBER

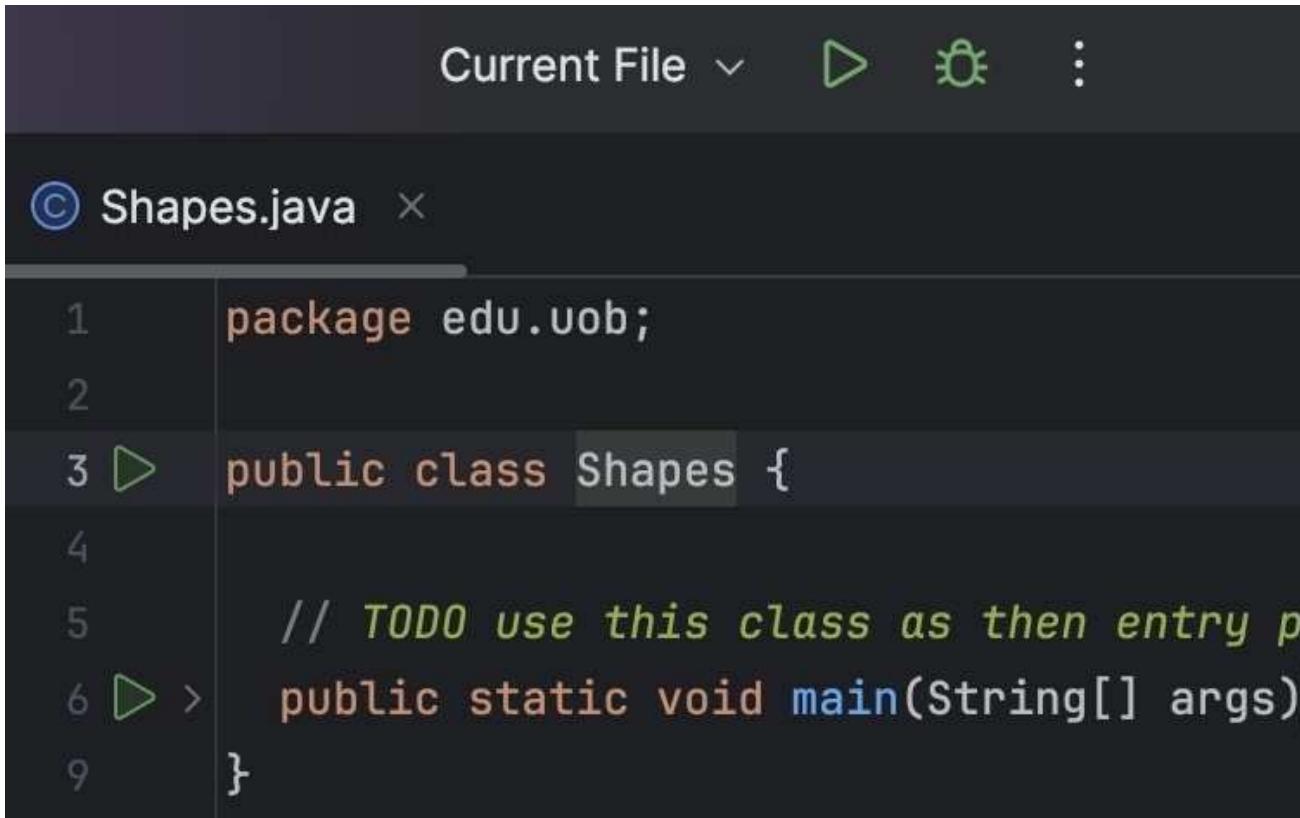
Labs machines ALREADY have a JDK installed
(so you won't need to follow the previous steps)
You only need to install JDK on your OWN laptop

Back to the Project...

Ready to Run !

With the main class open in the editing panel...

You should now see the green "run" button at the top



A screenshot of a Java code editor showing the file `Shapes.java`. The code contains a single class definition:1 package edu.uob;
2
3 ▶ public class Shapes {
4
5 // TODO use this class as then entry p
6 ▶ > public static void main(String[] args)
7
8 }
9The editor interface includes a toolbar with icons for current file, run, and settings. The run icon is highlighted in green, indicating the file is ready to be executed.

Success !

Might take a while for IntelliJ to compile & run the code

It has to build a lot of files the first time around

If everything worked OK, you should see...



A screenshot of an IntelliJ terminal window. The window has a dark background and light-colored text. At the top, there are several icons: a green circular arrow, a square, a camera, a timer, and a vertical ellipsis. Below the icons, the text "Hello world!" is displayed, followed by "Process finished with exit code 0". On the left side of the window, there is a vertical toolbar with several icons: a double arrow pointing up, a double arrow pointing down, a horizontal line with arrows at both ends, a downward arrow inside a square, a clipboard icon, and a trash can icon. The bottom left corner of the window shows the number "27/33".

Command Line

Although we will be using IntelliJ *most* of the time
It is useful to also be able to use the command line

Coursework will be marked on the command line
It's essential that you check your code runs there !

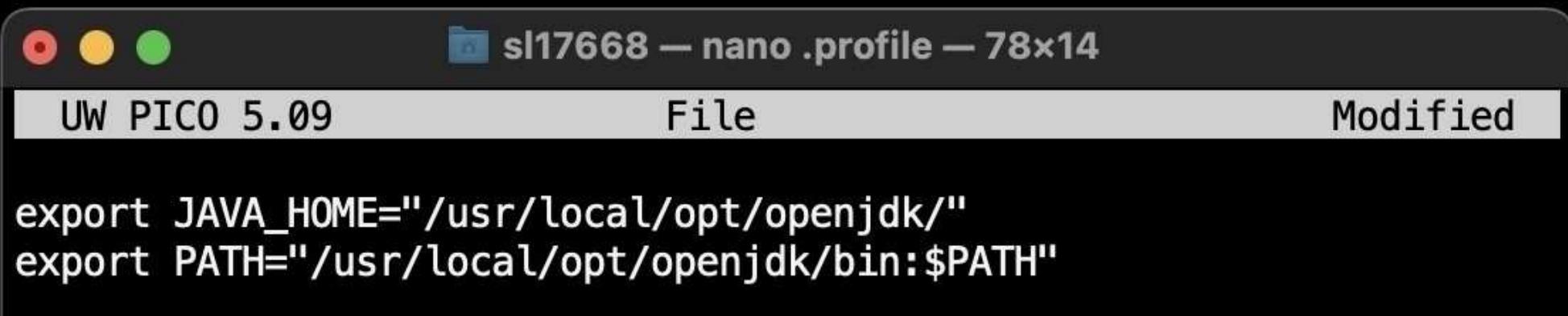
In order to be able to compile and run your code
You must tell command line where to find the JDK...

OSX and Linux Environment Variables

Add two environment variables to your shell config file
(`~/.profile` on OSX or `~/.bashrc` on linux)

`JAVA_HOME` must point to your installed JDK folder

You must also prepend the JDK `*bin*` folder to `$PATH`



A screenshot of a macOS terminal window titled "sl17668 – nano .profile – 78x14". The window has the standard OS X title bar with red, yellow, and green buttons. The main area shows the following text in white on a black background:

```
export JAVA_HOME="/usr/local/opt/openjdk/"
export PATH="/usr/local/opt/openjdk/bin:$PATH"
```

Please note that:

`/usr/local/opt/openjdk/`

Is the location of MY installation of the JDK

YOURS is going to be somewhere different !
(wherever IntelliJ said it was going to put it)

Windows Environment Variables

Environment Variables in Windows are set differently

Using a graphical interface in "System Preferences"

See separate guide to setting up JDK on windows

User variables

Variable	Value
Path	C:\Program Files (x86)\Common Files\Oracle\Java\javapath;C:\Wi...
TEMP	C:\Users\ AppData\Local\Temp
TMP	C:\Users\ AppData\Local\Temp

New...

Edit...

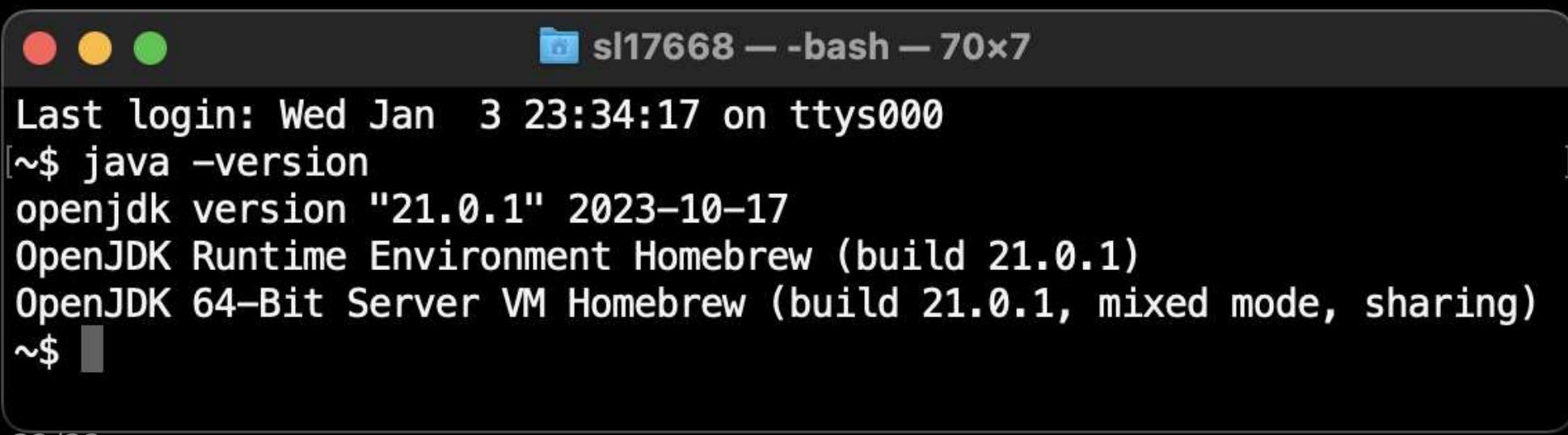
Delete

Testing Your Environment

Open a fresh terminal or command prompt and type:

```
java -version
```

If everything worked, you'll see something like:



```
Last login: Wed Jan  3 23:34:17 on ttys000
[~$ java -version
openjdk version "21.0.1" 2023-10-17
OpenJDK Runtime Environment Homebrew (build 21.0.1)
OpenJDK 64-Bit Server VM Homebrew (build 21.0.1, mixed mode, sharing)
~$ ]
```

And now to work !