JDK Installation

It is recommended to install the specific jdk version used for this hands-on lab (openjdk 20.0.2).

For a Mac, if your chip is Apple, use this installer (To find which chip your Mac uses, click the Apple icon at the top left of the screen, then click 'About this Mac' and the display will indicate which chip is being used.

macOS Arm 64 DMG Installer
https://download.oracle.com/java/20/archive/jdk-20.0.2_macos-aarch64_bin.dmg (sha256)

If your chip is Intel, use this installer:

macOS x64 DMG Installer
https://download.oracle.com/java/20/archive/jdk-20.0.2_macos-x64_bin.dmg (sha256)

Download the appropriate installer. Go to the downloads folder and doubleclick the dmg file, then doubleclick the pkg file and follow the instructions to install the jdk.

The jdk will be installed to:
/Library/Java/JavaVirtualMachines

IntelliJ IDEA Installation

This Lab requires an IDE, and in the course we are using IntelliJ IDEA. You will need to download and install IntelliJ IDEA. Please note that the Ultimate edition has features that the Community edition of IntelliJ IDEA does not have. It is recommended to use the Ultimate edition. The Ultimate edition is not free however, you can download and use a 30 day trial of the Ultimate edition.

Make sure to be on the correct tab for your OS. And again, for a Mac, select the dmg for the chip your Mac is using (Apple or Intel).
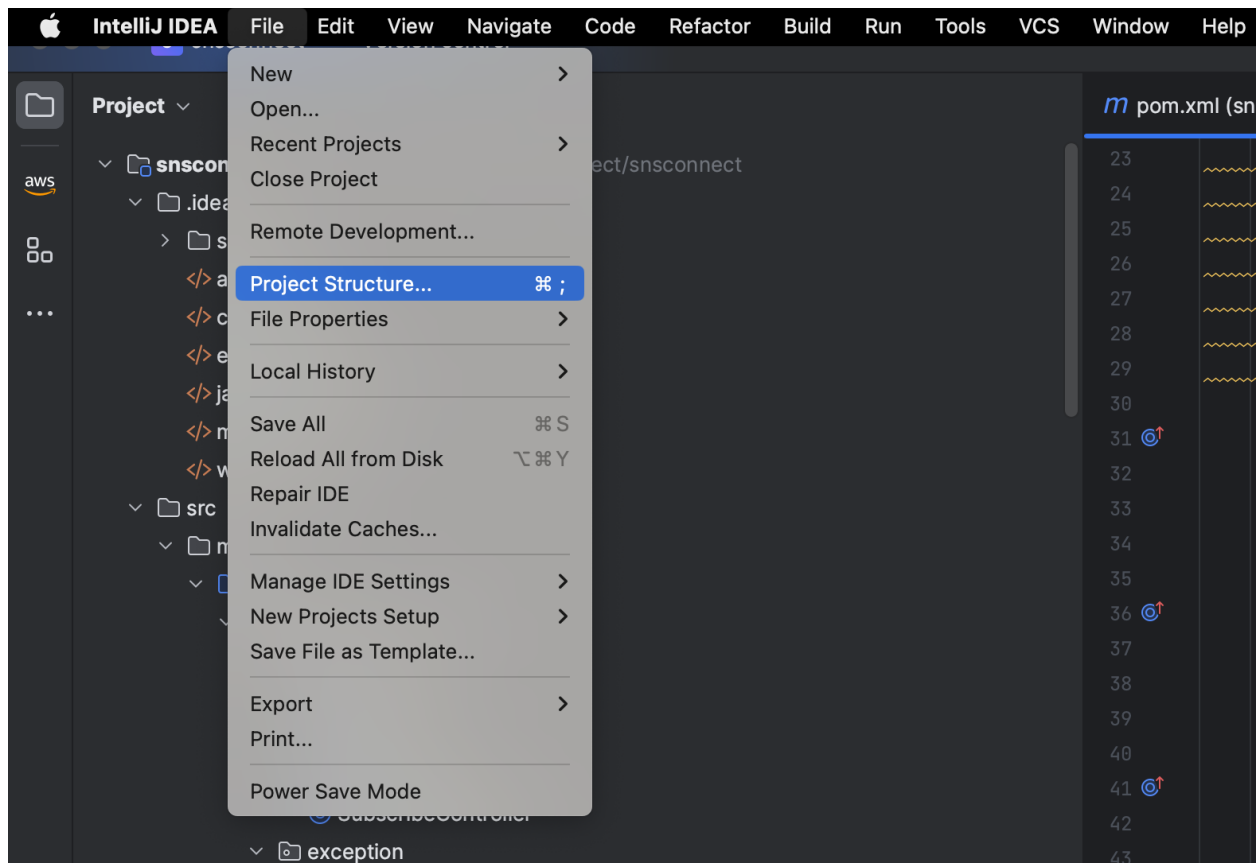
Download for the Ultimate (Professional) Edition:

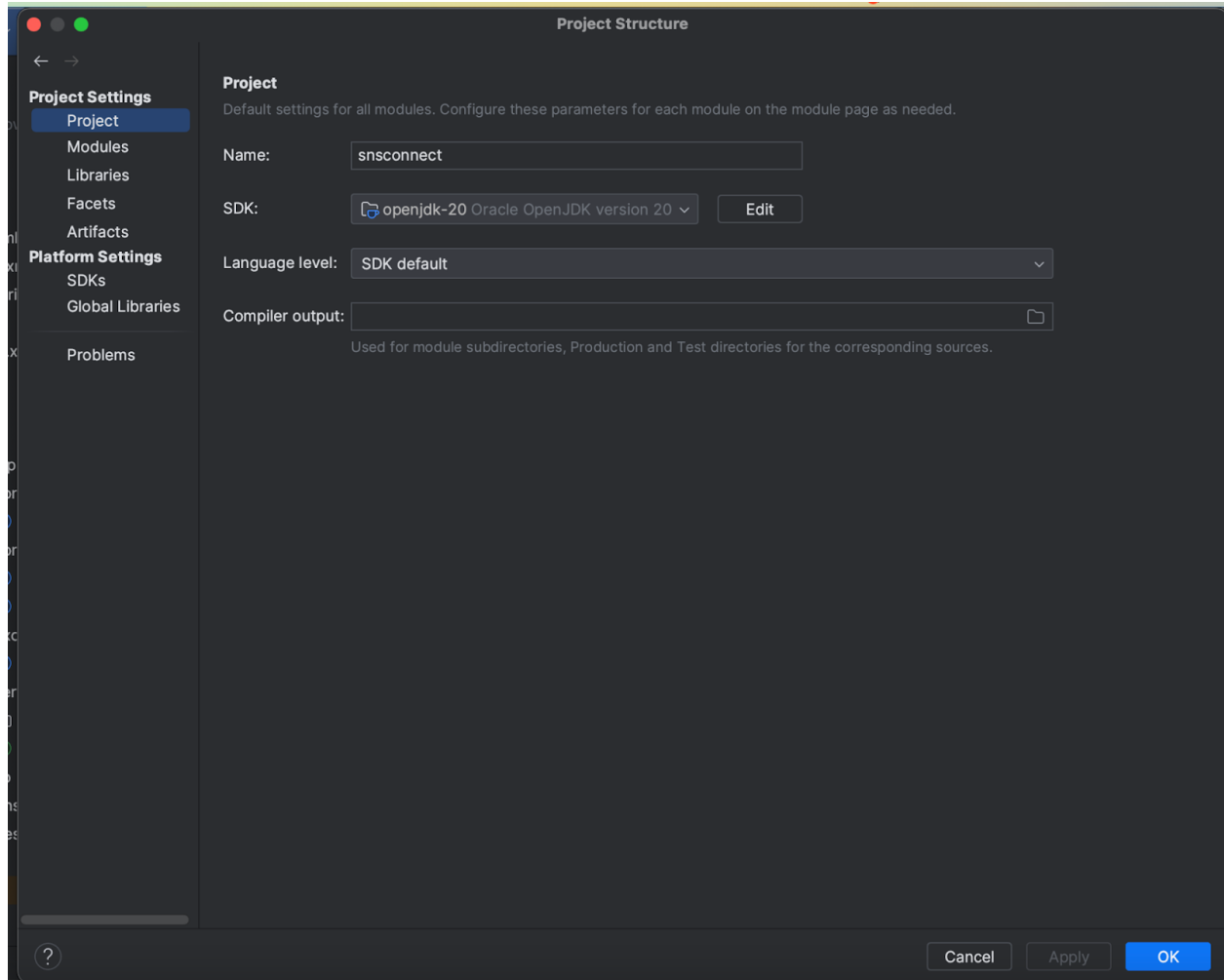https://www.jetbrains.com/idea/download/?section=mac

Run the installer and follow the Wizard steps.
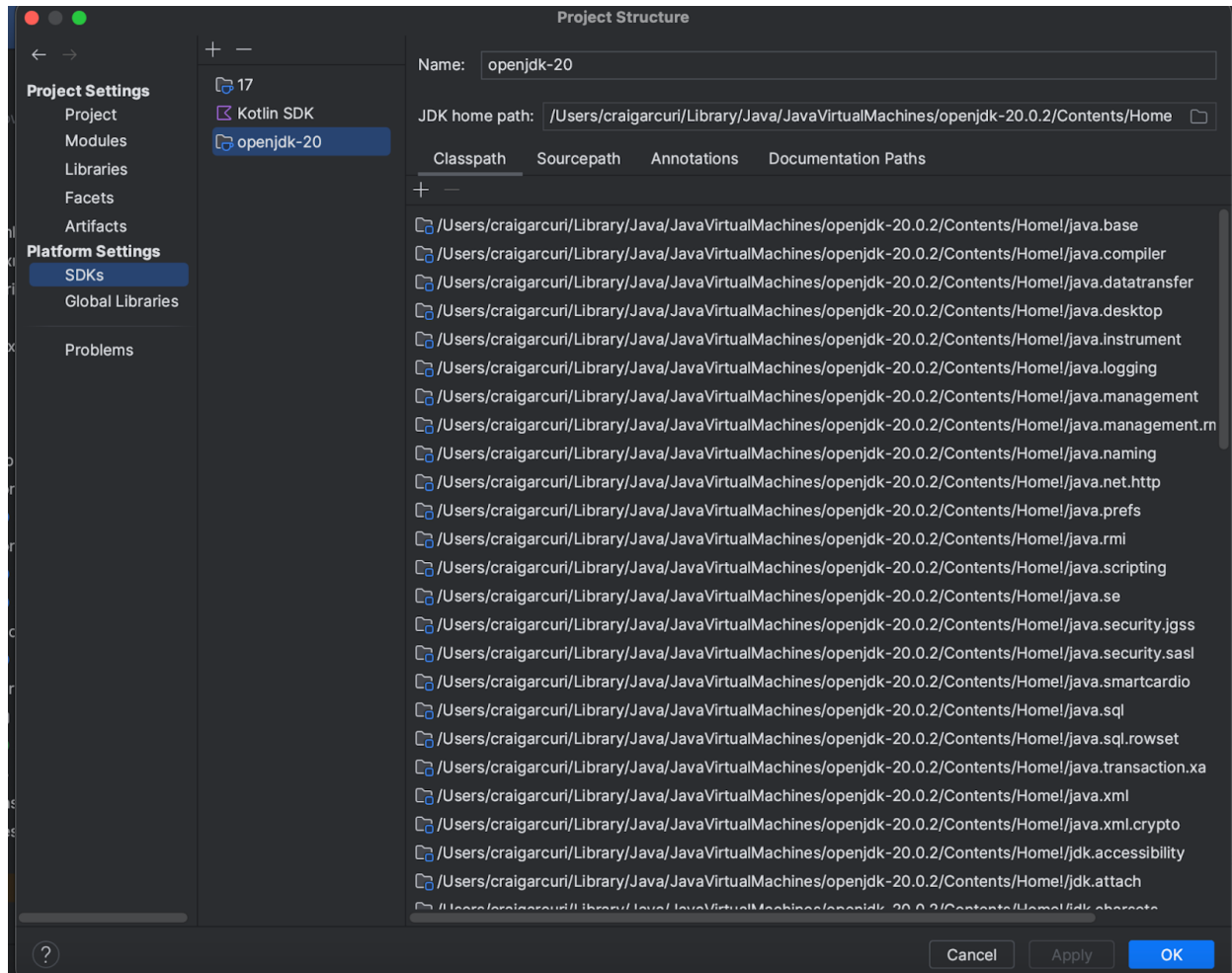
Set the jdk in IntelliJ IDEA

Open IntelliJ IDEA, click File → Project Structure

Then click the Edit button (IntelliJ comes with a default jdk, but you must have a jdk installed on your machine to develop in IntelliJ. We need to select the jdk we installed above):

Project Structure

Project Settings
Project
Modules
Libraries
Facets
Artifacts

Platform Settings
SDKs
Global Libraries

Problems

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

Name: snsconnect

SDK: openjdk-20 Oracle OpenJDK version 20   Edit

Language level: SDK default

Compiler output:

Used for module subdirectories, Production and Test directories for the corresponding sources.

Cancel   Apply   OK

When you click edit, you will have a list of available jdk's, and need to select openjdk 20, then click 'OK'.
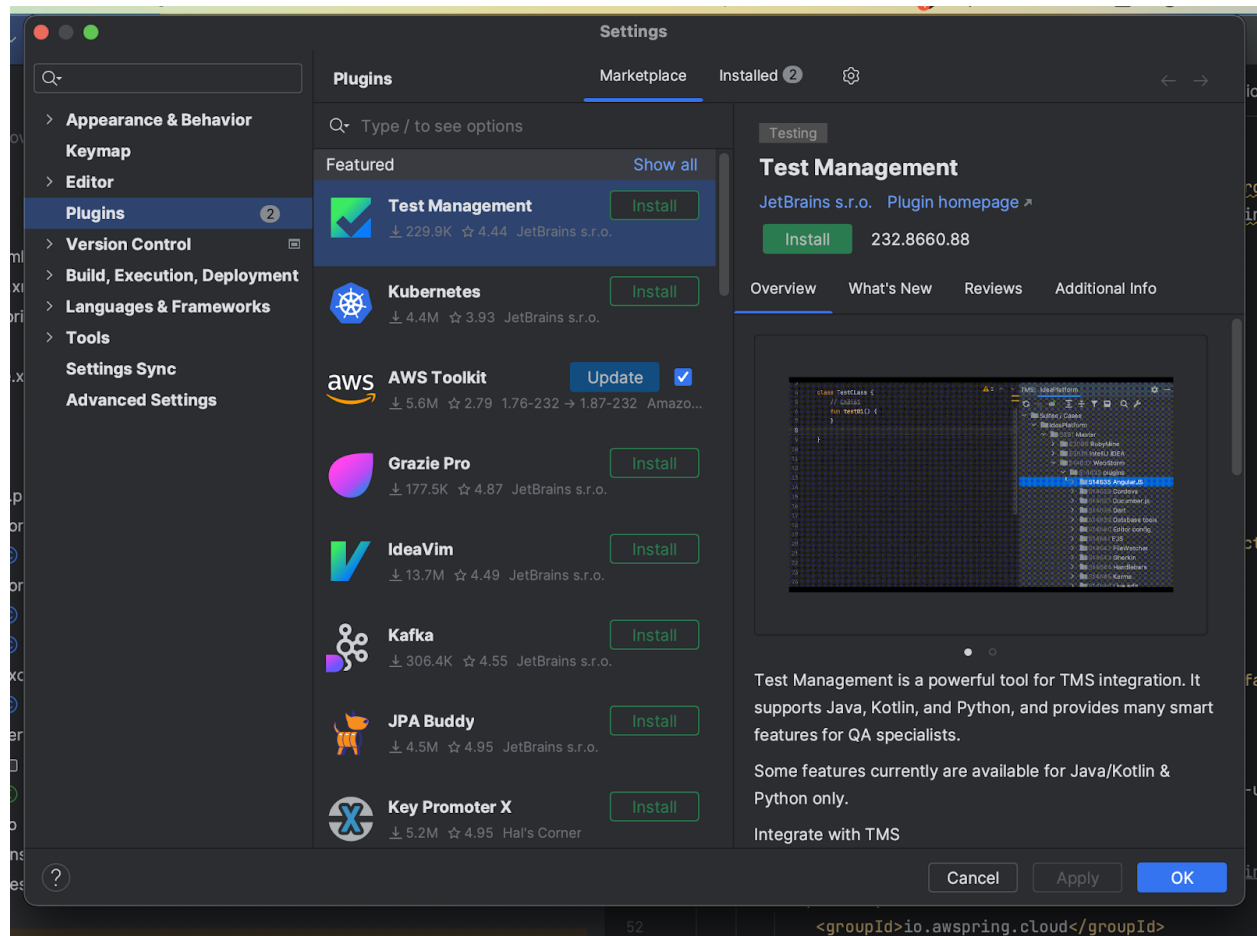

Installing plugins in IntelliJ IDEA

To install plugins, click IntelliJ IDEA (top left of the screen), then Settings:
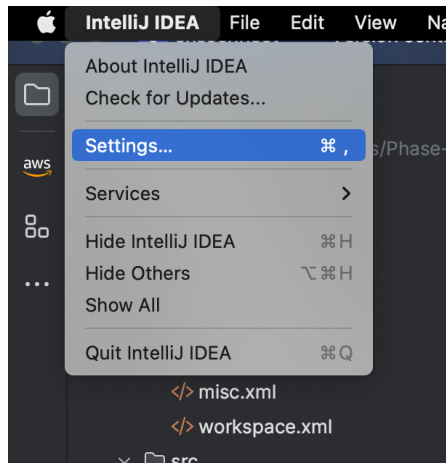
Select plugins and make sure you are on the 'Marketplace' tab:

The two plugins used in this lab are 'AWS Toolkit' and 'Lombok'. If you don't see them in the list, you can search for them (type the name of the plugin in the search bar).
Once located, just click 'Install' for each of them.

After the 'AWS Toolkit' plugin is installed, restart your IDE.

Once the IDE starts, in the bottom left of the IDE will be a new 'AWS Explorer' tool window:
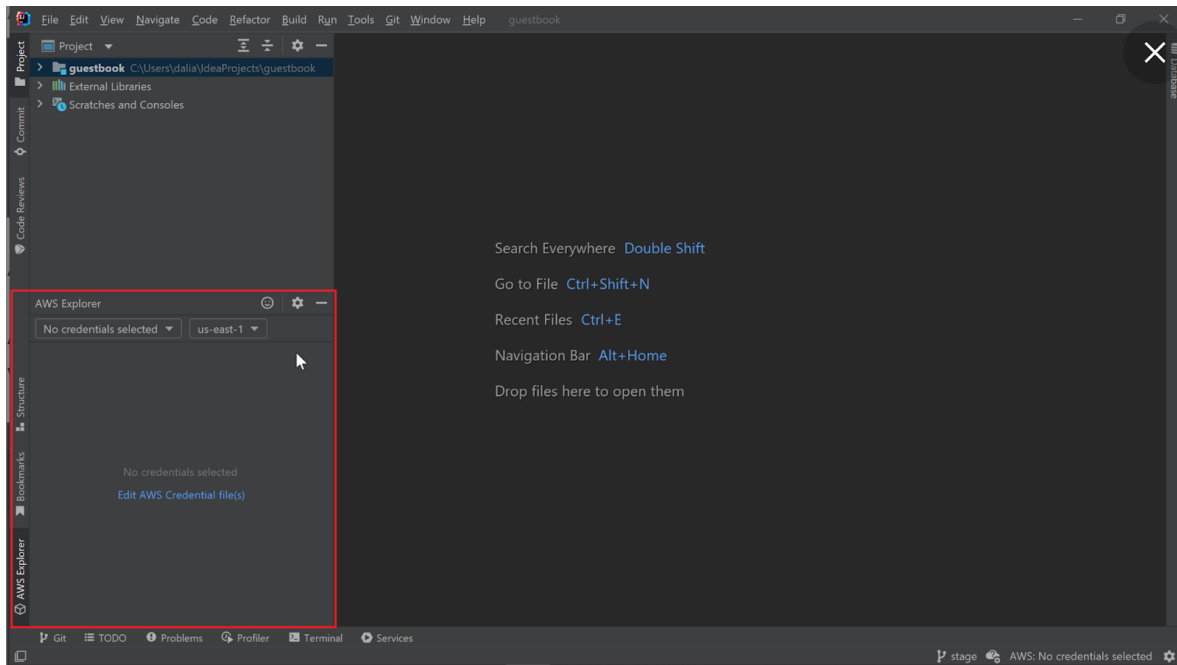
You need to connect the IDE to your AWS account.

Link IntelliJ IDEA to your AWS Account

Much like working with the AWS CLI, IntelliJ IDEA uses a credential file to connect to your AWS account. If you've worked with the AWS CLI and have a credentials file on your machine, IntelliJ IDEA **may** detect the credential file and automatically connect to your account.

- ● Another way this can go is that the credential file may cause a conflict and prevent linking IntelliJ IDEA to your AWS account. In this case, you can either delete or rename the existing credential file, then you can proceed with the next steps.

If your machine has no credentials file, and there are no credentials to choose from in the dropdown, then you will see this:
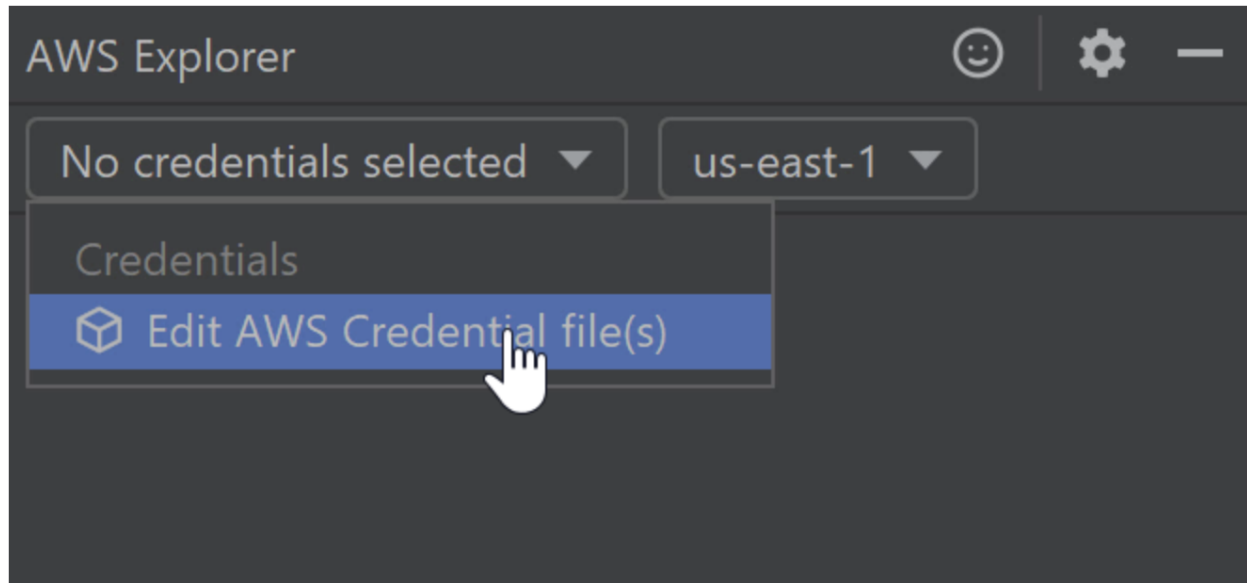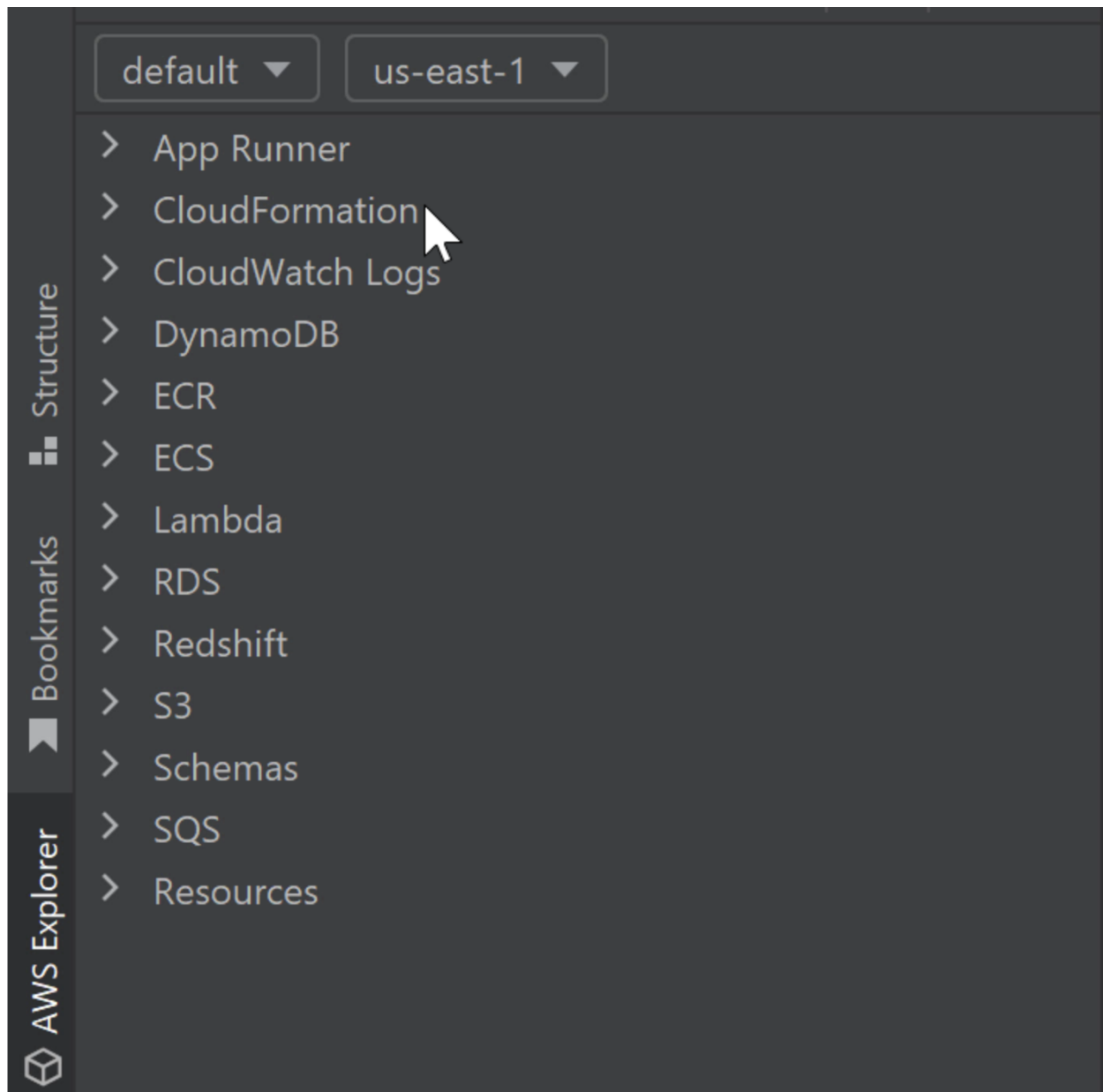
Click 'Edit AWS Credential file(s).

IntelliJ IDEA will prompt you to create a file. Click 'Create'.

A new configuration file is created containing comments at the top describing the contents of the files and fields where we can enter our aws_access_key_id and aws_secret_access_key.

For this to work, you have to have create a user in AWS IAM with Programmatic Access, and created access and secret access keys. Download the file with these keys and insert the keys into the above configuration file.

In IntelliJ IDEA, I'll click on the 'Save' blue button. Then, in the AWS Explorer window, I'll click on the credentials drop-down menu. I can now see the 'Profile:default' associated with my credentials. Once I select the default profile, IntelliJ IDEA will load a list of AWS services associated with my AWS account.

If you run into issues, consult the AWS Toolkit User Guide:

https://docs.aws.amazon.com/toolkit-for-jetbrains/latest/userguide/welcome.html