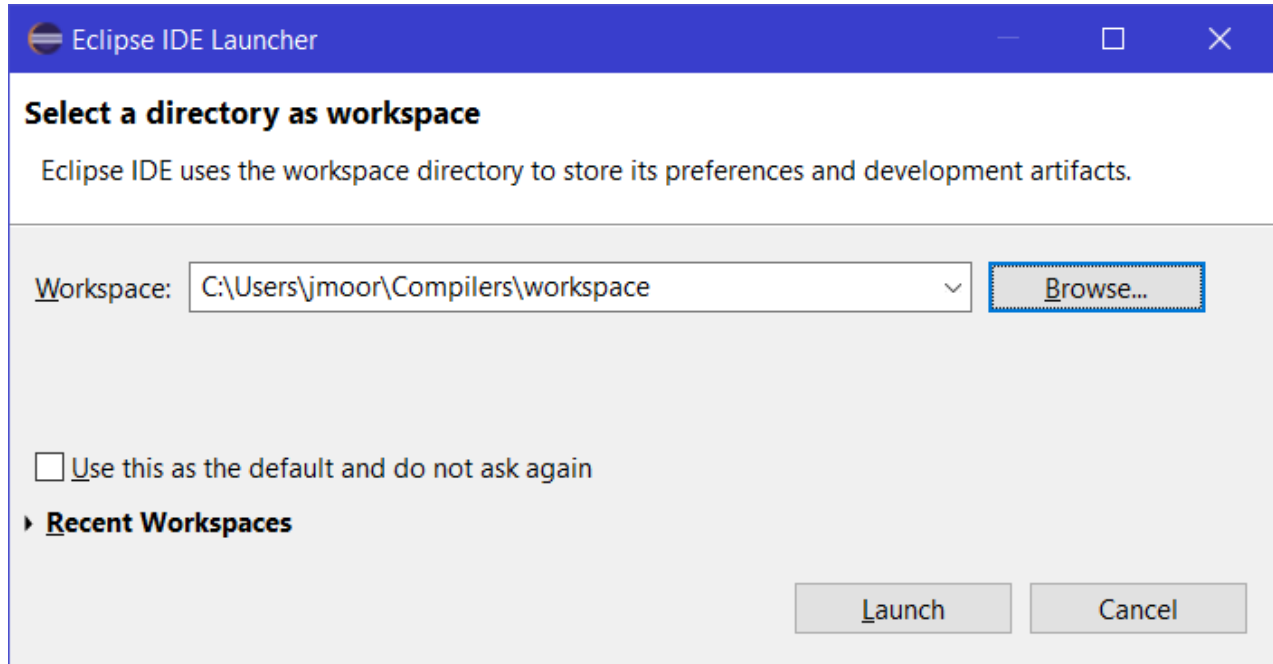


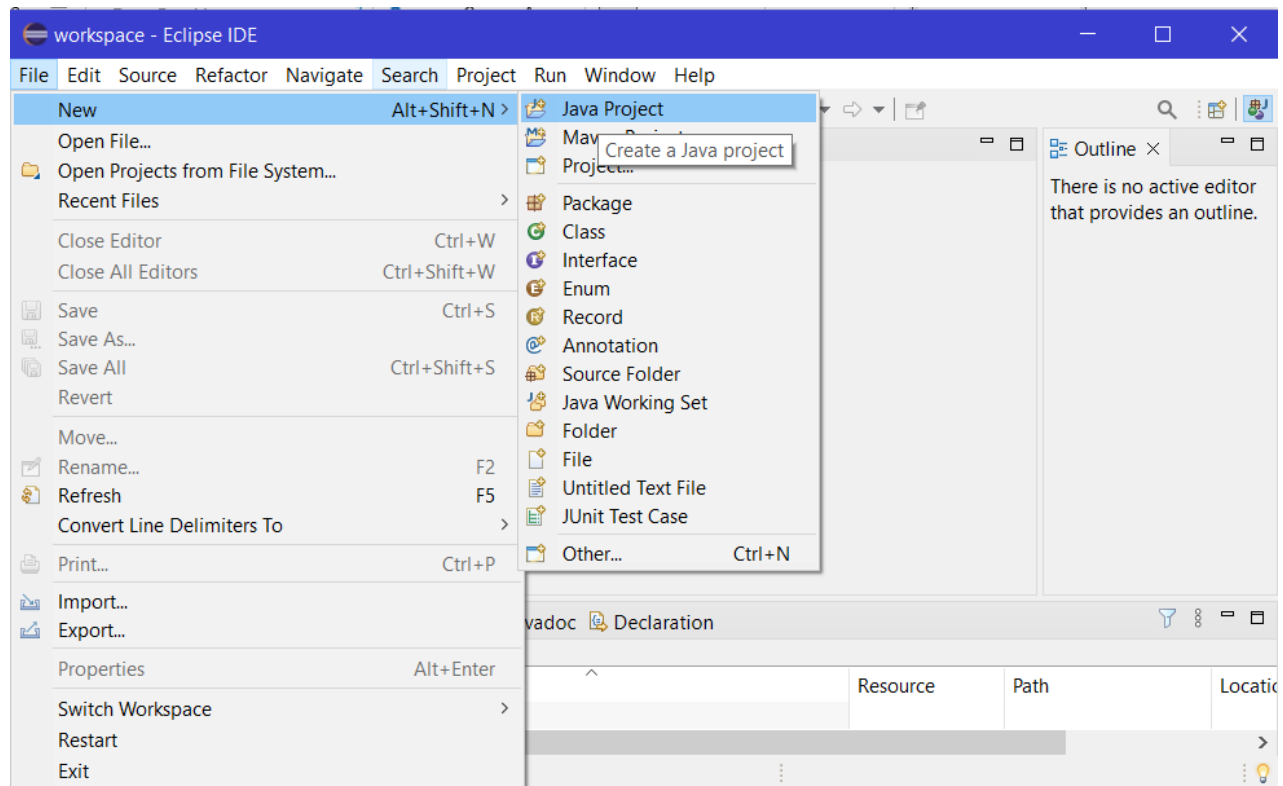
Tips on Creating a Multi-Module Project Using Eclipse

This handout illustrates *one* way to create a multi-module Java project for your compiler using Eclipse. It creates the Compiler (`edu.citadel.compiler`) and CVM (`edu.citadel.cvm`) modules as described in Appendix A of the book, but other modules are created similarly.

1. Create a new workspace at your preferred location. This example uses the name “workspace”.



2. Each module for your compiler will be packaged as a separate Eclipse project in the workspace.
Select File - > New -> Java Project



3. Name the project (module) `edu.citadel.compiler` and click Finish.

New Java Project

Create a Java Project

Create a Java project in the workspace or in an external location.

Project name:

☒ Use default location

Location: [Browse...](#)

JRE

☒ Use an execution environment JRE: [Configure JREs...](#)

☐ Use a project specific JRE:

☐ Use default JRE 'graalvm-ce-java17-22.3.2' and workspace compiler preferences

Project layout

☐ Use project folder as root for sources and class files

☒ Create separate folders for sources and class files [Configure default...](#)

Working sets

☐ Add project to working sets [New...](#)

Working sets: [Select...](#)

Module

☒ Create module-info.java file

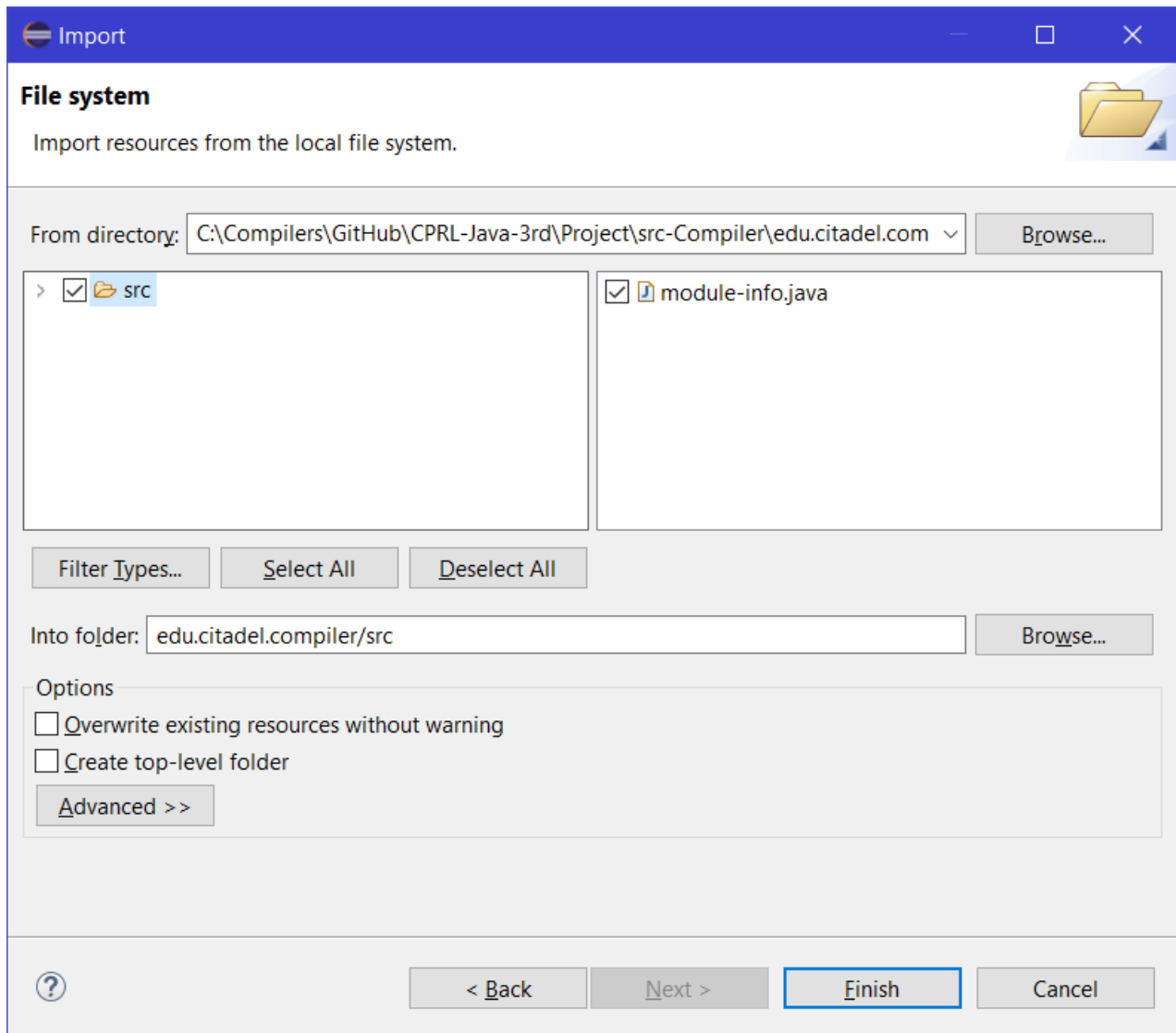
Module name:

☒ Generate comments

module name will be "edu.citadel.compiler" (if no module is specified, then project name will be used as module name)

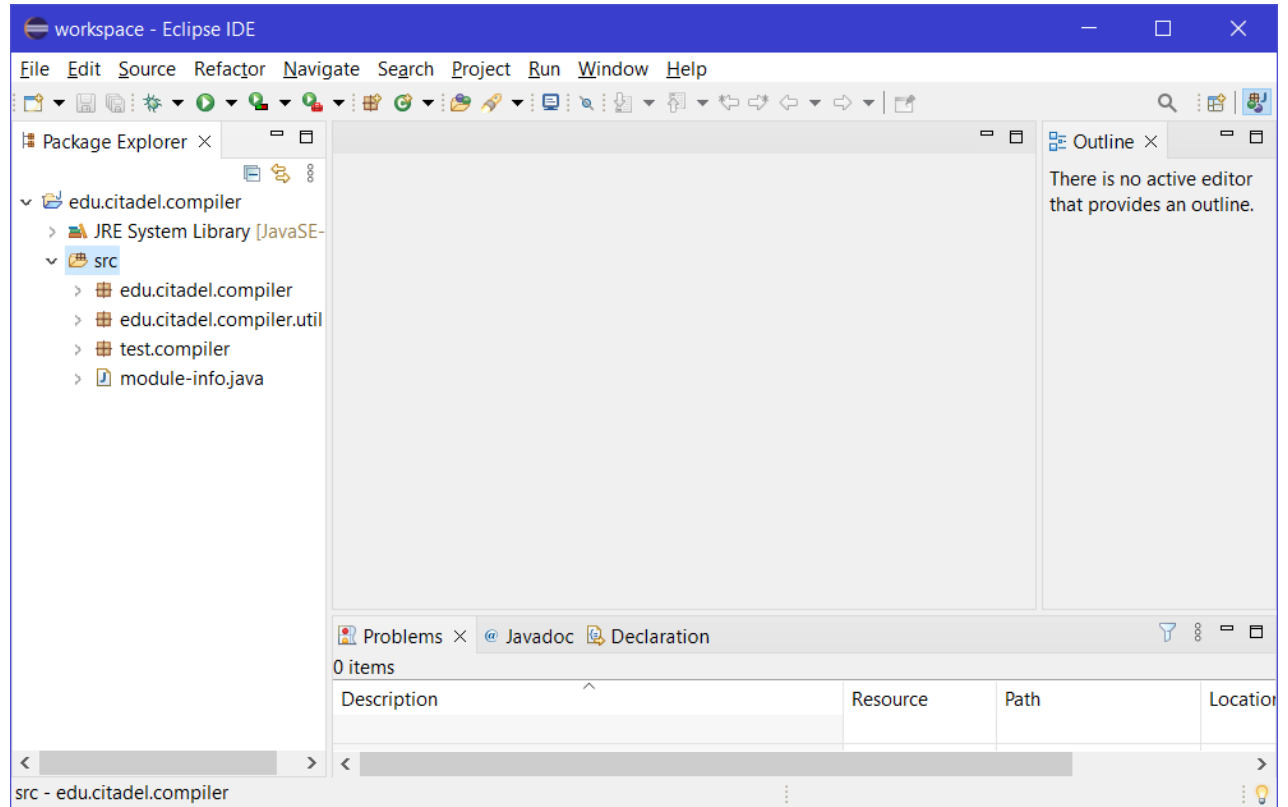
[?](#) [< Back](#) [Next >](#) [Finish](#) [Cancel](#)

- Once any module has been created, you can add new packages and Java source files to the module in the standard way, but all source code for this particular module is available in the GitHub repository as described in Appendix A. There are several ways to copy the code downloaded from GitHub into this module. The easiest way is to right click on the `src` folder, select `Import... -> General -> File System`, and browse to the `src` folder downloaded from GitHub.



Alternatively, instead of using Eclipse to import the source code, use a file explorer to copy the `src` folder for this module (previously downloaded from GitHub) to the `src` folder in the newly created Eclipse project. This will also replace the file named `module-info.java` created automatically by Eclipse. Back in Eclipse, right click on `src` and select `refresh` or simply press the `F5` key.

5. If you expand the `src` folder in Eclipse, it should look similar to the following.



6. Use steps 2-4 above, create a new module named `edu.citadel.cvm`. This module has a dependency on module `edu.citadel.compiler`, so after creating the module using these steps Eclipse indicates that errors exist. We need to tell Eclipse to look for missing files in the compiler project. Right click on the project and select Build Path -> Configure Build Path... Select the Projects tab, and add `edu.citadel.compiler` to the Modulepath. Click Apply and Close.

