

Advanced: Build and run using IntelliJ IDEA

In “Step 2: Setup the build system: JDK and Gradle”, IntelliJ was configured to use Gradle as tool for launching JabRef. It is also possible to use IntelliJ’s internal build and run system to launch JabRef. Due to [IDEA-119280](#), it is a bit more work.

- 1 Navigate to **File > Settings... > Build, Execution, Deployment > Build Tools > Gradle**.
- 2 Change the setting “Build and run using:” to “IntelliJ IDEA”.
- 3 Navigate to **File > Settings... > Build, Execution, Deployment > Compiler**.
- 4 Uncheck `Clear output directory on rebuild`.
- 5 Navigate to **File > Settings... > Build, Execution, Deployment > Compiler > Java Compiler**.
- 6 Uncheck `--Use 'release' option for cross-compilation`.
- 7 Click “OK” to store the preferences and close the dialog.
- 8 **Build > Build Project** (Ctrl+F9)
- 9 Open the project view (Alt+1, on macOS cmd+1)
- 10 Copy all build resources to the folder of the build classes
 - a Navigate to the folder `build/resources/main`
 - b Right click -> “Open In” -> “Explorer (Finder on macOS)”
 - c Navigate into directory “main”
 - d Select the folder `out/production/classes`
 - e Right click -> “Open In” -> “Explorer (Finder on macOS)”
 - f Navigate into directory “classes”
 - g Now you have two Explorer windows opened. Copy all files and directories from the first one to the second one.
- 11 Locate the class `Launcher` (e.g., by ctrl+N and then typing `Launcher`). Press Enter to jump to that class.

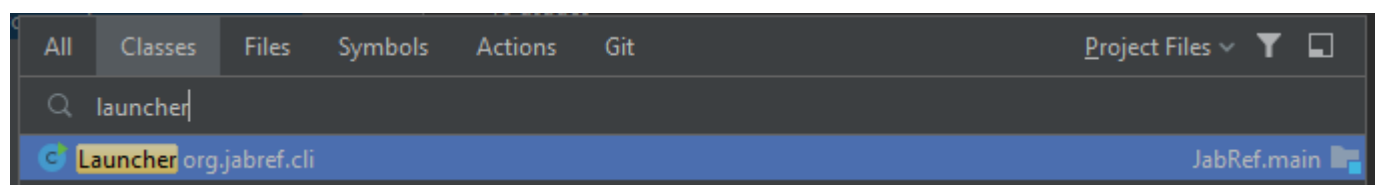
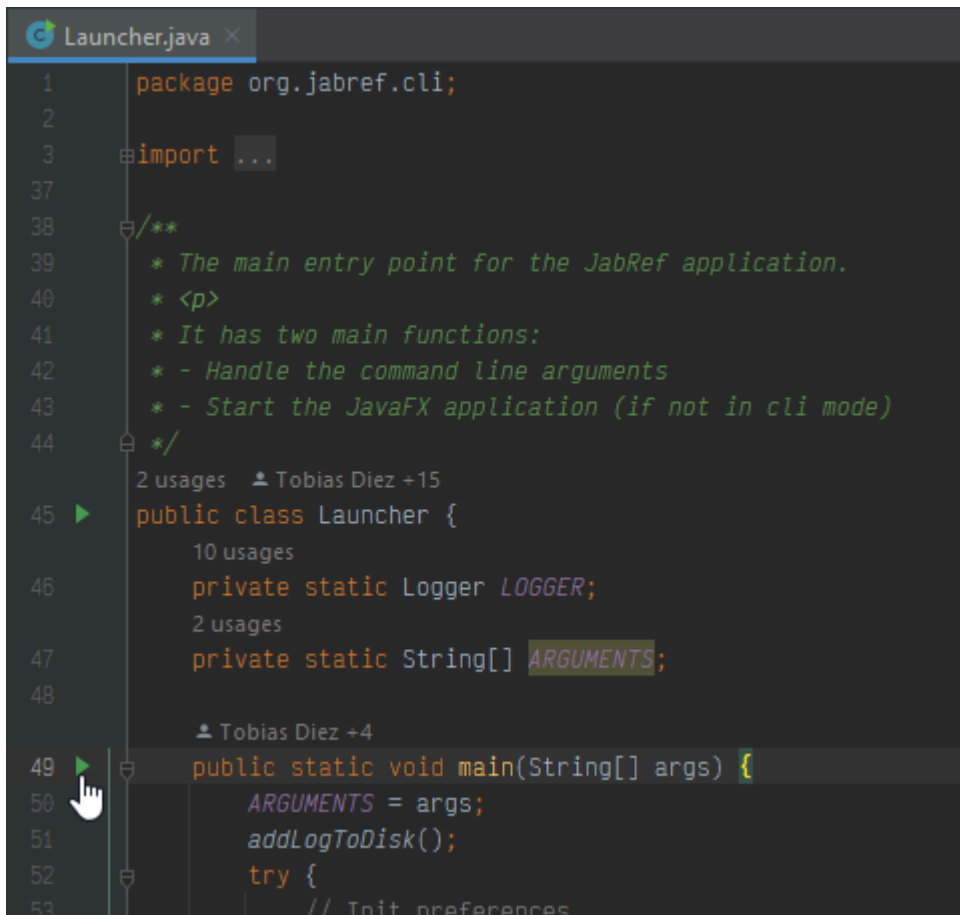


Figure: IntelliJ search for class “Launcher”

- 1 Click on the green play button next to the `main` method to create a Launch configuration. IntelliJ will fail in launching.



```
1 package org.jabref.cli;
2
3 import ...
4
37
38 /**
39  * The main entry point for the JabRef application.
40  * <p>
41  * It has two main functions:
42  * - Handle the command line arguments
43  * - Start the JavaFX application (if not in cli mode)
44  */
45 2 usages Tobias Diez +15
46 public class Launcher {
47     10 usages
48     private static Logger LOGGER;
49     2 usages
50     private static String[] ARGUMENTS;
51
52     Tobias Diez +4
53     public static void main(String[] args) {
54         ARGUMENTS = args;
55         addLogToDisk();
56         try {
57             // Init preferences
```

Figure: However on green play

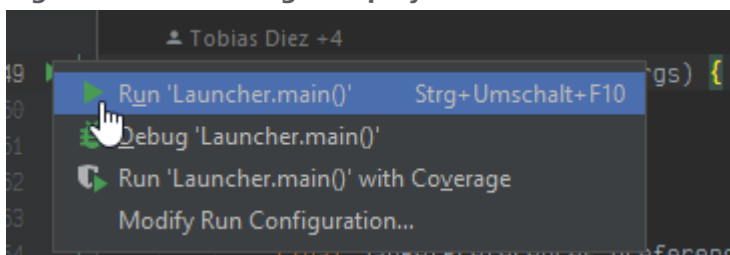


Figure: Run JabRef via launcher

- 1 On the top right of the IntelliJ window, next to the newly created launch configuration, click on the drop down
- 2 Click on “Edit Configurations...”
- 3 On the right, click on “Modify options”
- 4 Ensure that “Use classpath of module” is checked
- 5 Select “Add VM options”
- 6 In the newly appearing field for VM options, insert:

```
--add-exports=javafx.controls/com.sun.javafx.scene.control=org.jabref
--add-opens=org.controlsfx.controls/org.controlsfx.control.textfield=org.jabref
--add-exports=org.controlsfx.controls/impl.org.controlsfx.skin=org.jabref
--add-exports javafx.controls/com.sun.javafx.scene.control=org.jabref
--add-exports org.controlsfx.controls/impl.org.controlsfx.skin=org.jabref
--add-exports javafx.graphics/com.sun.javafx.scene=org.controlsfx.controls
--add-opens javafx.graphics/javafx.scene=org.controlsfx.controls
```

```
--add-exports javafx.graphics/com.sun.javafx.scene.traversal=org.controlsfx.controls
--add-exports javafx.graphics/com.sun.javafx.css=org.controlsfx.controls
--add-exports javafx.controls/com.sun.javafx.scene.control.behavior=org.controlsfx.controls
--add-exports javafx.controls/com.sun.javafx.scene.control=org.controlsfx.controls
--add-opens=javafx.controls/javafx.scene.control.skin=org.controlsfx.controls
--add-exports javafx.controls/com.sun.javafx.scene.control.inputmap=org.controlsfx.controls
--add-exports javafx.base/com.sun.javafx.event=org.controlsfx.controls
--add-exports javafx.base/com.sun.javafx.collections=org.controlsfx.controls
--add-exports javafx.base/com.sun.javafx.runtime=org.controlsfx.controls
--add-exports javafx.web/com.sun.webkit=org.controlsfx.controls
--add-exports javafx.graphics/com.sun.javafx.css=org.controlsfx.controls
--add-reads org.jabref=org.fxmisc.flowless
--add-reads org.jabref=org.apache.commons.csv
```

7 Click “Apply”

8 Click “Run”. You can also click on the debug symbol next to it to enable stopping at breakpoints.

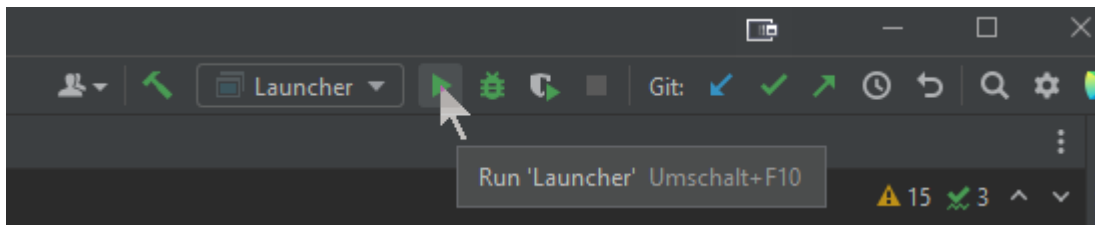


Figure: Launch menu contains “Launcher”