

# Configuring Compiler Settings

---

This section describes how to gain access to the compiler settings, modify the list of recognized resources, exclude certain paths from compilation, select the desired compiler, and configure annotation processing.

The issues of configuring compiler output on the project and module levels are discussed in the sections [Configuring Project Compiler Output](#) and [Configuring Module Compiler Output](#).

## To configure compiler settings

1. [Open the Project Settings](#).
2. In the [Compiler](#) page, configure common settings, for example:
  - In the **Resource patterns** field, update the regular expression that describes extensions of the files to be recognized as resources. Separate individual patterns with semicolons (;). Refer to the [list of wildcard characters](#), and examples.
  - If necessary, choose to perform compilation as an external process.
3. On the [Excludes](#) page, specify the files and folders that should be excluded from compilation. Click **+**, and choose the desired directory from the [Select Path](#) dialog box.

Please note the following:

- If an excluded path is a dependency of the source code being compiled, this path will be included in compilation and processed by the compiler as required.
  - If some file contains errors and fails to compile, but it is not important for the current project state, or if you want to skip some files and not include them into the output directory, you can exclude such files from compilation.
4. Select the specific compiler. For example, in the [Java Compiler](#) page, select the desired compiler from the **Use compiler** drop-down list. Specify compiler-specific settings.
  5. In the [Annotation Processors](#) page, [configure annotation processing parameters](#).
  6. Apply changes and close the dialog.
  7. Define the paths, providing the proper order of module roots and libraries, as described in the section [Configuring Module Dependencies](#).

Take into account that compiler in IntelliJ IDEA works sequentially, that is, in accordance with [dependencies](#) and [processing order](#) defined for your project modules, which might be important for some projects.

## See Also

Concepts:

- [Compiler and Builder](#)

Reference:

- [Compiler](#)

Web Resources:

- [Developer Community](#) 