

Minifying JavaScript

The term *minification* or *compression* in the context of JavaScript means removing all unnecessary characters, such as *spaces*, *new lines*, *comments* without affecting the functionality of the source code.

These characters facilitate working with the code at the development and debugging stage by improving the code readability. However at the production stage these characters become extraneous: being insignificant for code execution, they increase the size of code to be transferred. Therefore it is considered good practice to remove them before deployment.

IntelliJ IDEA supports integration with the following JavaScript minification tools:

- [YUI Compressor](#) 
- [UglifyJS](#) 
- [Closure Compiler](#) 

In IntelliJ IDEA, minifier configurations are called *File Watchers*. For each supported minifier, IntelliJ IDEA provides a predefined *File Watcher* template. Predefined *File Watcher* templates are available at the IntelliJ IDEA level. To run a minifier against your project files, you need to create a project-specific *File Watcher* based on the relevant template, at least, specify the path to the minifier to use on your machine.

On this page:

- [Installing and configuring the Closure Compiler minification tool](#)
- [Installing and configuring the YUI Compressor JS or UglifyJS minification tool](#)
- [Creating a File Watcher](#)
- [Minifying the code](#)

Installing and configuring the Closure Compiler minification tool

1. Download and install the [Node.js](#)  framework because the JavaScript minifier is started through *Node.js*.
2. Install and enable the *NodeJS* repository plugin.

The plugin is not bundled with IntelliJ IDEA, but it is available from the [JetBrains plugin repository](#) . Once enabled, the plugin is available at the IDE level, that is, you can use it in all your IntelliJ IDEA projects. See [Installing, Updating and Uninstalling Repository Plugins](#) and [Enabling and Disabling Plugins](#) for details.

3. Install and enable the *File Watchers* repository plugin.

The plugin is not bundled with IntelliJ IDEA, but it is available from the [IntelliJ IDEA plugin repository](#) . Once enabled, the plugin is available at the IDE level, that is, you can use it in all your IntelliJ IDEA projects. See [Installing, Updating and Uninstalling Repository Plugins](#) and [Enabling and Disabling Plugins](#) for details.

4. Download the compiled tool at <https://closure-compiler.googlecode.com/files/compiler-latest.zip> , or at <http://dl.google.com/closure-compiler/compiler-latest.zip> .
5. Extract the archive to the folder under the *Node.js*. This ensures that *Node.js*, which is required for starting the tool, will be specified in the path to it.

Installing and configuring the YUI Compressor JS or UglifyJS minification tool

1. Download and install [Node.js](#). The framework is required for two reasons:
 - The JavaScript minifier is started through *Node.js*.
 - *NPM*, which is a part of the framework, is also the easiest way to download the JavaScript minifier.

For details on using *Node.js* in IntelliJ IDEA, see the section [Node.js](#)

Alternatively, you can define [Node.js](#) as an external tool, as described in the section [Configuring third-party tools](#). This approach is helpful, when you need facilities that are missing in the plugin, for example, the possibility to pass certain parameters as wildcards.

2. If you are going to use the command line mode, make sure the following paths are added to the PATH variable:

1. The path to the parent folder of the *Node.js* executable file.
2. The path to the *npm* folder.

This enables you to launch the JavaScript minifier and *npm* from any folder.

3. Install and enable the *NodeJS* repository plugin.

The plugin is not bundled with IntelliJ IDEA, but it is available from the [JetBrains plugin repository](#). Once enabled, the plugin is available at the IDE level, that is, you can use it in all your IntelliJ IDEA projects. See [Installing, Updating and Uninstalling Repository Plugins](#) and [Enabling and Disabling Plugins](#) for details.

4. Install and enable the *File Watchers* repository plugin.

The plugin is not bundled with IntelliJ IDEA, but it is available from the [IntelliJ IDEA plugin repository](#). Once enabled, the plugin is available at the IDE level, that is, you can use it in all your IntelliJ IDEA projects. See [Installing, Updating and Uninstalling Repository Plugins](#) and [Enabling and Disabling Plugins](#) for details.

5. Download and install the JavaScript minification tool. The easiest way is to use the *Node Package Manager (npm)*, which is a part of [Node.js](#).

1. Switch to the directory where the *Node Package Manager (npm)* is stored or define a path variable for it so it is available from any folder.
2. Depending on the tool you want to use, type one of the following commands at the command line prompt:
 - To have the *YUI Compressor JS* installed, type:

```
npm install yuicompressor
```

- To have the *UglifyJS* installed, type:

```
npm install uglify-js
```

If you use the *Node Package Manager (npm)*, the tools are installed under *Node.js* so *Node.js*, which is required for starting the tool, will be specified in the path to it.

Creating a File Watcher

IntelliJ IDEA provides a common procedure and user interface for creating *File Watchers* of all types. The only difference is in the predefined templates you choose in each case.

1. To start creating a *File Watcher*, open the **Project Settings** by choosing **File | Settings** on the main menu, and then click **File Watchers**. The [File Watchers page](#) that opens, shows the list of *File Watchers* that are already configured in the project.
2. Click the **Add** button **+** or press **Alt+Insert**. Depending on the tool you are going to use, choose the appropriate predefined template from the pop-up list:
 - **Closure Compiler**
 - **UglifyJS**
 - **YUI Compressor JS**
3. In the **Program** text box, specify the path to the compiler executable file or archive depending on the chosen predefined template:
 - `compiler.jar` for **Closure Compiler**
 - `yuicompressor-<version>.jar` for **YUI Compressor JS**
 - `uglifyjs.cmd` for **UglifyJS**If you installed the tool through the *Node Package Manager*, IntelliJ IDEA locates the required file itself and fills in the field automatically. Otherwise, type the path manually or click the **Browse** button  and choose the file location in the dialog box that opens.
4. Proceed as described on page [Using File Watchers](#).

Minifying the code

When a *Minification File Watcher* is enabled (see [Enabling and disabling File Watchers](#)), minification starts automatically as soon as a file to which transpilation is applicable is changed or saved, see [Configuring the behaviour of the File Watcher](#).

IntelliJ IDEA creates a separate file with the generated output. The file has the name of the source *JavaScript* file and the extension `min.js`. The location of the generated file is defined in the **Output paths to refresh** text box of the [New Watcher dialog](#). However, in the **Project Tree**, it is shown under the source *JavaScript* file which is now displayed as a node.

See Also

Procedures:

- [Using File Watchers](#)

Language and Framework-Specific Guidelines:

- [JavaScript-Specific Guidelines](#)

Reference:

- [File Watchers](#)
- [New Watcher Dialog](#)

External Links:

- <https://developers.google.com/closure/compiler/> 
- <http://yui.github.com/yuicompressor/> 
- <https://github.com/mishoo/UglifyJS> 

Web Resources:

- [Developer Community](#) 

