

Running Unit Tests on Karma

JavaScript unit tests are run in a browser against a test server which actually handles the testing process.

This server allows you to capture a browser to run tests in, loads the test targets to the captured browser, controls the testing process, and exchanges data between the browser and IntelliJ IDEA, so you can view test results without leaving the IDE.

The test server does not necessarily have to be on your machine, it can be launched right from IntelliJ IDEA through the [Cucumber.js](#) test runner.

Karma executes unit tests according to a `karma.conf.js` configuration file which is generated semi-automatically in the interactive mode.

On this page:

- [Prerequisites](#)
- [Generating a Karma configuration file](#)
- [Creating a Karma run configuration](#)
- [Launching unit tests](#)

Prerequisites

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

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 transpiledLanguage toolType and *npm* from any folder.

3. Make sure the *Karma* test runner is installed as described in [Preparing to Use Karma Test Runner](#).
4. 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.

5. Install and enable the *Karma* 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.

Generating a Karma configuration file

Karma executes unit tests according to a `karma.conf.js` configuration file which is generated semi-automatically in the interactive mode. The instruction below ensures successful creation of a consistent configuration file `karma.conf.js` which in its turn ensures successful execution of the tests in your project. For more details, see <http://karma-runner.github.io/0.10/config/configuration-file.html>.

1. In the command line mode, switch to your project directory.
2. Type the following command at the command line prompt:

```
karma init
```

If *Karma* does not start, check the installation: the parent folder or the *Karma* executable file should be specified in the `PATH` variable.

3. Answer the questions to specify the following basic settings:
 - The testing framework to use.
 - The browsers to be captured automatically.
 - The patterns that define the location of test files to be involved in testing or excluded from it, for example, `src/*.js` and `test/*.js`. For more details, see <http://karma-runner.github.io/0.10/config/files.html>.

Creating a Karma run configuration

1. Open the [Run/Debug Configuration](#) dialog box by doing one of the following:
 - On the main menu, choose **Run | Edit Configurations**.
 - Press `RunConfiguration` and choose **Edit Configuration** on the context menu.
 - Open the test file in the editor, and then choose **Create <file name>** on the context menu.
 - Select the test file in the **Project** tool window, and then choose **Create <file name>** on the context menu of the selection.
2. Click the **Add** button **+** on the toolbar and select the **Karma** configuration type.
3. In the dialog box that opens, specify the location of the *Node.js* and *Karma* executable files and the path to the `karma.conf.js` configuration file.
4. Apply the changes and close the dialog box.

Launching unit tests

1. To launch the tests according to a run configuration, select the **Karma** run/debug configuration from the list on the main toolbar. Then click the **Run** button **▶** to the right of the list.
2. The *Karma* test server starts automatically without any steps from your side. View and analyze messages from the test server in the **Karma Server** tab of the **Run** tool window.
3. Monitor test execution in the **Test Runner** tab of the **Run** tool window as described in [Monitoring and Managing Tests](#).

See Also

Procedures:

- [Preparing to Use Karma Test Runner](#)
- [Testing Frameworks](#)
- [Running Unit Test on JSTestDriver](#)
- [Running, Debugging and Testing](#)

- [Testing](#)

Reference:

- [Run/Debug Configuration: Karma](#)

Web Resources:

- [Developer Community](#) 