

Run/Debug Configuration: JSTestDriver

To get access to this dialog box, [enable](#) the *JSTestDriver* [bundled plugin](#).

In this dialog box, create a configuration to be used for running JavaScript unit tests in the browser against a JSTestDriver server. Configurations of this type enable running unit tests based on the [JSTestDriver Assertion](#), [Jasmine](#), and [QUnit](#) frameworks.

This section provides descriptions of the [configuration-specific items](#), as well as the [toolbar](#) and [options](#) that are common for all run/debug configurations.

The dialog box consists of the following tabs and areas:

- [Configuration Tab](#)
- [Debug Tab](#)
- [Coverage Tab](#)
- [Toolbar](#)
- [Common Options](#)

Configuration Tab

In this tab, specify the tests to run, the configuration files to use, and the server to run the tests against.

Test

In this area, tell IntelliJ IDEA where to find the tests to execute and how to get [test runner configuration files](#) that define which test files to load and in which order. The main approaches are:

- Specify the location of one or several previously created configuration files.
- Point at the target test file, test case, or test method, and then specify the location of the corresponding configuration file.

The way to find tests and configuration files is defined in the **Test** drop-down list. This choice determines the set of other controls in the area.

Item	Description	Available for
Test	<p>In this drop-down list, specify how IntelliJ IDEA will get test runner configuration files.</p> <ul style="list-style-type: none">■ All configuration files in directory: select this option to use all the test runner configuration files in a specific folder.■ Configuration file: select this option to use a specific test runner configuration file.■ JavaScript test file: select this option to have tests from a specific file executed using one of the previously defined configuration files.■ Case: select this option to run a specific test case using one of the previously defined configuration files.■ Method: select this option to run a specific test method using one of the previously defined configuration files.	
Directory	<p>In this text box, specify the folder to look for test runner configuration files in. Type the path manually or click the Browse button  and select the required folder in the dialog box, that opens.</p>	All configuration files in directory

Item	Description	Available for
Matched configuration files	This read-only field shows a list of all the *.jstd and JSTestDriver.conf test runner configuration files detected in the specified folder .	All configuration files in directory
Configuration file	In this text box, specify the test runner configuration file to use. Type the path manually or click the Browse button  and select the required file in the dialog box that opens.	Configuration file
JS test file	In this text box, specify the JavaScript files with tests to be executed. Type the path manually or click the Browse button  and select the required file in the dialog box, that opens.	JavaScript test file Case Method
Case	In this text box, type the name of the target case from the specified JavaScript file .	Case Method
Method	In this text box, type the name of the target method from the specified test case within the specified JavaScript file .	Method

Server

In this area, appoint the test server to run tests against.

Item	Description
At address	Choose this option to have the test execution handles by a remote test server. In the text box, specify the URL address to access the server through.
Running in IDE	Choose this option to have test execution handles through the JSTestDriver server that comes bundled with IntelliJ IDEA and can be launched from it.
Test Connection	Click this button to check that the specified test server is accessible. <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <p>The server must be running. Start the server from IntelliJ IDEA or manually, according to the server-specific instructions.</p> </div>

Debug Tab

In this tab, appoint the browser to debug the unit test in when two or more [browsers are captured](#) simultaneously.

Item	Description
Debug	From this drop-down list, choose the browser to debug the specified tests in when two browsers are captured at a time. The available options are: <ul style="list-style-type: none"> ■ Chrome ■ Firefox

Coverage Tab

In this tab, specify the files that you do not want to be involved in coverage analysis.

Item	Description
Exclude file path	<p>In this area, create a list of files to be skipped during coverage analysis.</p> <ul style="list-style-type: none">■ To add an item to the list, click the Add button  and choose the required file in the dialog that opens.■ To delete a file from the list so coverage is measured for it, select the file and click the Remove button .

Toolbar

Item	Shortcut	Description
	Alt+Insert	Click this button to add new configuration to the list.
	Alt+Delete	Click this button to remove the selected configuration from the list.
	Ctrl+D	Click this button to create a copy of the selected configuration.
	Edit defaults	Click this button to edit the default configuration templates. The defaults are used for the newly created configurations.
 or	Alt+Up or Alt+Down	<p>Use these buttons to move the selected configuration or group of configurations (folder) up and down in the list.</p> <p>The order of configurations or folders in the list defines the order, in which configurations appear in the Run/Debug drop-down list on the main toolbar.</p>
		<p>Use this button to create a new folder.</p> <p>If one or more run/debug configurations have the focus, then the selected run/debug configurations are automatically moved to the newly created folder. If only a category has the focus, then an empty folder is created.</p> <p>Move run/debug configurations to a folder using drag-and-drop, or  buttons.</p>

Common options

Item	Description
Name	In this text box, specify the name of the current run/debug configuration. This field does not appear for the default run/debug configurations.
Defaults	This node in the left-hand pane of the dialog box contains the default run/debug configuration settings. Select the desired configuration to change its default settings in the right-hand pane. The defaults are applied to all newly created run/debug configurations.

Item	Description
Share	<p>Select this check box to make the run/debug configuration available to other team members.</p> <p>If the directory-based project format is used, the settings for a run/debug configuration are stored in a separate xml file in the <code>.idea\runConfigurations</code> folder if the run/debug configuration is shared and in the <code>.idea\workspace.xml</code> file otherwise.</p> <p>If the file-based format is used, the settings are stored in the <code>.ipr</code> file for shared configurations or in the <code>.iws</code> file for the ones that are not shared.</p> <p>This check box is not available when editing the run/debug configuration defaults.</p>
Single instance only	<p>If this check box is selected, this run/debug configuration cannot be launched more than once.</p> <p>Every time a new run/debug configuration is launched, IntelliJ IDEA checks the presence of the other instances of the same run/debug configuration, and displays a confirmation dialog box. If you click OK in the confirmation dialog box, the first instance of the runner will be stopped, and the next one will take its place.</p> <p>This make sense, when usage of certain resources can cause conflicts, or when launching two run/debug configurations of the same type consumes too much of the CPU and memory resources.</p> <p>If this check box is not selected, it is possible to launch as many instances of the runner as required. So doing, each runner will start in its own tab of the Run tool window.</p>

Item	Description		
Before launch	Specify which tasks should be carried out before starting the run/debug configuration. The specified tasks are performed in the order that they appear in the list.		
	Item	Keyboard shortcut	Description
	+	Alt+Insert	<p>Click this icon to add a task to the list. Select the task to be added:</p> <ul style="list-style-type: none"> ■ Run External tool. Select this option to run an application which is external to IntelliJ IDEA. In the dialog that opens, select the application or applications that should be run. If the necessary application is not defined in IntelliJ IDEA yet, add its definition. For more information, see Configuring Third-Party Tools and External Tools. ■ Build Artifacts. Select this option to have an artifact or artifacts built. In the dialog that opens, select the artifact or artifacts that should be built. See also, Configuring Artifacts. ■ Run Ant target. Select this option to have an Ant target run. In the dialog that opens, select the target to be run. For more information, see Ant. ■ Generate CoffeeScript Source Maps. Select this option to have the source maps for your CoffeeScript sources generated. In the dialog that opens, specify where your CoffeeScript source files are located. For more information, see CoffeeScript Support. ■ Run Maven Goal. Select this option to have a Maven goal run. In the dialog that opens, select the goal to be run. For more information, see Maven. ■ Run Remote External tool: Add a remote SSH external tool. Refer to the section Remote SSH External Tools for details.
	-	Alt+Delete	Click this icon to remove the selected task from the list.
		Enter	Click this icon to edit the selected task. Make the necessary changes in the dialog that opens.
	Alt+Up	Click this icon to move the selected task one line up in the list.	

Item	Item	Keyboard shortcut	Description
		Alt+Down	Click this icon to move the selected task one line down in the list.
	Show this page		Select this check box to have the run/debug configuration settings shown prior to actually starting the run/debug configuration.

See Also

Procedures:

- [Running JavaScript Unit Tests in Browser](#)
- [Monitoring Code Coverage for JavaScript](#)

Web Resources:

- [Developer Community](#) 