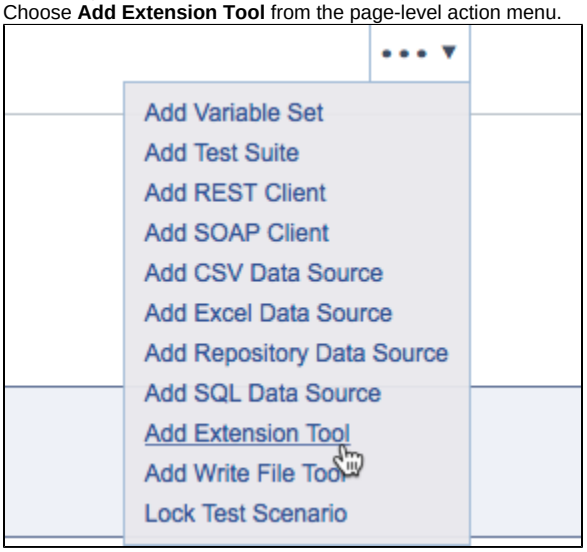
Scripting

In this section:

Add Extensions Tool

The Extensions Tool provides the primary mechanism for interfacing with the Extensibility API. It allows you to extend tests with actions that may not be directly supported by built-in features. The tool supports Java, JavaScript, Python/Jython, and Groovy for writing custom scripts that address unique system needs. It also enables you to perform complex validation or transformation operations on tool outputs, or execute any custom actions as part of test scenarios. You can also refer to the Extensibility API, found under the Help menu.

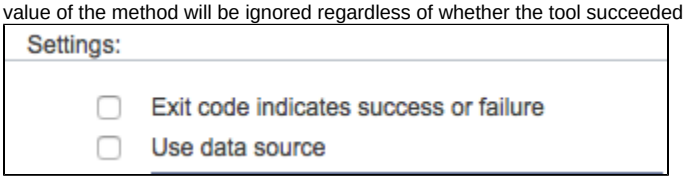
1. In the left pane, select the test artifact where you want the new extension tool added.
2. Choose **Add Extension Tool** from the page-level action menu.



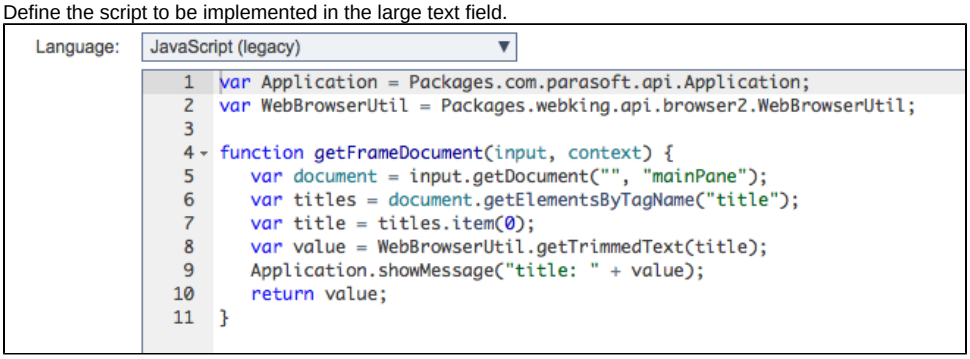
3. (Optional) Modify the name of the newly-created tool.
4. Configure the tool as described below.
5. Save the new tool configuration.

Configuring the Extension Tool

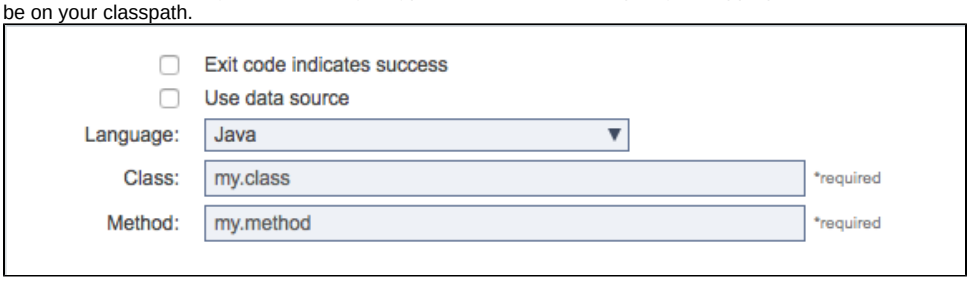
1. If a return value for this tool indicates the tool's success, enable the **Exit code indicates success** option in the settings. Otherwise, the return value of the method will be ignored regardless of whether the tool succeeded or failed.



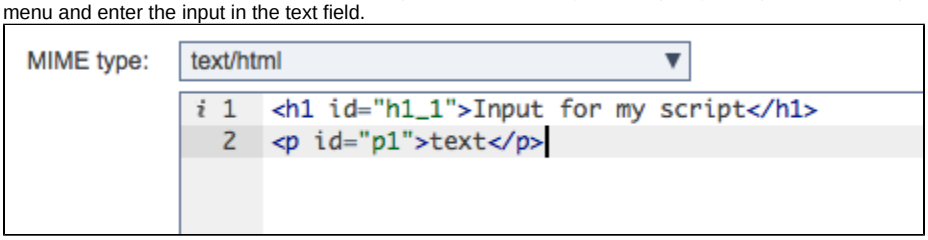
2. Enable the **Use data source** option to use the data source associated with the test suite.
3. Choose a scripting language from the **Language** drop-down menu that your method is or will be written in.
4. Define the script to be implemented in the large text field.



The form will check for syntax errors as you type. For Java methods, specify the appropriate class in the Class field. The class you choose must be on your classpath.



5. Specify a method you want to use.
6. If the extension tool is not attached to the output of another tool, you can specify an input for your script. Choose a MIME type from the drop-down menu and enter the input in the text field.



7. Click **Save** when finished.