

REST Client Tool Window

Tools | Test RESTful Web Service
View | Tool Windows | REST Client

Use the **REST Client** tool window for [testing a RESTful Web Service](#). The tool window is intended for composing and submitting test requests to Web service methods based on the service API, as well as for viewing and analyzing server responses.

Please note the following:

- **View | Tool Windows | REST Client** - the tool window can be accessed this way after you have opened it using the **Tools | Test RESTful Web Service** command.
- The tool window is available only when the *REST Client* bundled plugin is enabled. The plugin is active by default. If not, activate it in the [Plugins](#) page of the [Settings](#) dialog box.

IntelliJ IDEA supports integration between the code and the contents of the **REST Client** tool window controls.

On this page:

- [Common Request Settings](#)
- [Request Tab](#)
- [Cookies Tab](#)
- [Response Tab](#)
- [Response Headers Tab](#)

Common Request Settings

In this area, choose the request method and specify the data to compose the [request URI](#) from.

Item	Description
HTTP method	In this drop-down list, specify the request method . The available options are: <ul style="list-style-type: none">■ GET■ POST■ PUT■ PATCH■ DELETE■ HEAD■ OPTIONS
Host/port	In this text box, type the URL address of the host where the target Web service is deployed and the port it listens to. By default, the port number is 80. If another port is used, specify it explicitly in the format <host URL>:<port number>.
Path	In this drop-down list, specify the relative path to the target method.

Item	Tooltip and shortcut	Description
	Submit Request	Click this button to submit the generated test request to the server. If a server is not trusted, IntelliJ IDEA shows a dialog box suggesting you to accept the server, or reject it. If you accept the server as trusted, IntelliJ IDEA writes its certificate to the trust store. On the next connect to the server, this dialog box will not be shown.
	Replay Recent Requests	Click this button to have a Recent Requests pop-up list displayed and select the relevant request. The fields are filled in with the settings of the selected request. Click the Submit Request button  .
	Export Request	Click this button to have the current request settings saved in an XML file so they are available in another IntelliJ IDEA session. In the dialog box that opens, specify the name of the file to save the settings in and its parent folder. When necessary, you can retrieve the saved settings and run the request again.
	Import Request	Click this button to have the settings of a previously saved request retrieved from an XML file. In the dialog box that opens, select the relevant XML file.
	Update resource paths from code	Click this button to synchronize the contents of the Path drop-down list with the @Path annotations.
	Configure HTTP Proxy	Click this button to specify proxy server settings in the dialog box that opens.
	Close	Close the REST Client tool window.
	Help	Show this page.

The server response code and the content length are shown in the upper-right corner of the **REST Client** tool window.

Request Tab

Use this tab to specify the parameters to be passed to the service in the generated test request either through the query string for **GET** requests or through the request body for other request types. Also configure interaction between the client side and the Web service by specifying the format of data that the service and the client accept.

Item	Description
------	-------------

Item	Description
Headers	<p>In this pane, specify the technical data included in the request header. These data are passed through header fields and define the format of the input parameters (accept field), the response format (content-type field), the caching mechanism (cache-control field), etc.</p> <p>To add a field to the list, click Add +, then specify the field name in the Name text box and the field value in the Value drop-down list.</p> <p>The set of fields and their values should comply with the Web service API. In other words, the specified input format should be exactly the one expected by the Web service as well as the expected response format should be exactly the one that the service returns.</p> <p>For accept, content-type, and some other fields IntelliJ IDEA provides a list of suggested values. Choose the relevant format type from the Value drop-down list.</p>
Request Parameters	<p>In this pane, specify the parameters to be passed to the target method through a query string inside the URL. This approach is used for requests of the type GET. By default, the pane shows an empty list with one line.</p> <ul style="list-style-type: none"> ■ To add a parameter, click Add +, then specify the name of the parameter in the Name text box and the value of the parameter in the Value drop-down list. ■ To delete a parameter from the list, select it and click Remove -. ■ To suppress sending the specified query string parameters and disable the controls in the Request Parameters pane, press the Don't send anything toggle button . <p>The set of parameters and their types should comply with the Web service API, in particular, they should be exactly the same as the input parameters of the target method.</p> <ul style="list-style-type: none"> ■ select this check box This may be helpful, for example, if you want to test passing parameters through other request methods in the request body but still preserve the data typed in the Query string parameters pane.
Request Body	<p>The pane is disabled when the GET request method is selected. In this pane, specify the input parameters to be passed to the target method inside a request message body.</p> <ul style="list-style-type: none"> ■ Empty: choose this option to send a request with an empty body. ■ Text: choose this option to send a request with a string of parameters with values. Specify the parameters in the text box next to the option. ■ File contents: choose this option to have the parameters inserted in the request body from a local text file. Specify the source text file in the File to send field. ■ File upload(multipart/form-data): choose this option if you want to pass a binary file as a parameter, which requires that the file be converted. Specify the source binary file in the File to send field.

Use this tab to create, save, edit, and remove cookies, both received through responses and created manually. The *name* and *value* of a cookie is automatically included in each request to the URL address that matches the *domain* and *path* specified for the cookie, provided that the *expiry date* has not been reached.

The tab shows a list of all currently available cookies that you received through responses or created manually. The cookies are shown in the order they were added to the list. When you click a cookie, its details become editable and are displayed in text boxes.

Item	Description
Name	In this fields, specify the name of the cookie to be included in the request.
Value	In this field, specify the value of the cookie to be included in requests.
Domain	In this field, specify the host and port the requests to which must be supplied with the <i>name</i> and <i>value</i> of the cookie.
Path	In this field, specify the path of the URL the requests to which must be supplied with the <i>name</i> and <i>value</i> of the cookie.
Expiry date	In this field, specify the expiry date of the cookie.
	Click this button to add a new row to the list and define a new coolie in it.
	Click this button to remove the selected cookie from the list.

Response Tab

Use this tab to view responses from the Web service. By default, responses are shown in the plain text form. Use the icons of the tab to have them displayed in the editor in the HTML, XML, and JSON formats.

Item	Tooltip	Description
	View as HTML	Click this button to open a new tab in the main editor window and display there the response as HTML.
	View as XML	Click this button to open a new tab in the main editor window and display there the server response as XML.
	View as JSON	Click this button to open a new tab in the main editor window and display there the server response as JSON.
	Open in browser	Click this button to view the response in your default Web browser.

Response Headers Tab

The tab shows the technical data provided in the [headers of Web service responses](#).

See Also

Procedures:

- [Testing RESTful Web Services](#)
- [RESTful WebServices](#)

Getting Started:

- [IntelliJ IDEA Tool Windows](#)

Web Resources:

- [Developer Community](#) 