

# Using RESTClient for Firefox

## The REST Client Firefox Add-On

The REST Client Add-On for Firefox is very useful for testing syntax. Download it here: <https://addons.mozilla.org/en-US/firefox/addon/restclient/>

- [Using the RESTClient plugin to GET and item](#)
- [Creating a basic New Item from RESTClient](#)
- [Creating a more advanced New Item with RESTClient \(using more fields\) Example](#)
  - [Full Example\("parentId":"4f2add32e4b009ef29abd063",](#)

### Using the RESTClient plugin to GET and item



Note: Parameters for other REST clients should be similar to these instructions, but may break the URL down into HOST and GET parts.

In HTTPClient it is very simple to query an item for JSON or XML. Set the "Method" to GET, pointed at the URL of the item we want to query. Click "Headers" menu item then "Custom Header". Set the Name = "Accept" and it's Value = "application/json" for JSON, "application/xml" for XML, or "application/text" for the normal text that you would see in a web browser. That's it! Push "Send" and you will be able to see your results in raw and copyable form in the "Response Body" tab or formatted in a "Formatted JSON", "Formatted XML", or "Formatted HTML" tab.

You will also be returned with the "Response Header" in it's own tab with useful information like the Status Code.

The screenshot shows the RESTClient interface with the following details:

- Request** section: Method is set to GET, and the URL is `https://www.sciencebase.gov/catalog/item/559fa651e4b0b94a6401984b`. A red "SEND" button is visible.
- Headers** section: A custom header is added with the name "Accept" and the value "application/json". This header entry is highlighted with a red box.
- Body** section: The request body is currently empty, with the placeholder text "Request Body".

### Creating a basic New Item from RESTClient

To create a new item using the RESTClient for Firefox we first set the "Method" to "POST" and the URL to point at "www.sciencebase.gov/catalog/item/". The "Content-Type" value must now be set to "application/json" since we are sending it JSON. We put our JSON request into the "Request Body", here we will use the minimum requirements to create a new item, setting just the parentId and the title:

```
{ "parentId": "4f28888de4b050c1ade5f38e", "title": "Test Item Title" }
```

**[ - ] Request**


Method  URL  ★ ▼ SEND


**Headers** Remove All

Content-Type: application/json ×

**Body**

```
{ "parentId": "4f28888de4b050c1ade5f38e", "title": "Test Item Title" }
```

 Note: If you haven't used JSON before it is basically just a map with String keys and values that can be Strings or collections of Strings. See <http://www.json.org/> for more info.

 Note: You must be logged in for this to be successful! You will get a Status Code of 200 even if you are not logged in, but the item WILL NOT be created and it will return Formatted HTML (in the form of a login page) instead of the Formatted JSON of your new item. The reason we use this plugin is because you can sign into usgs in a different tab and then use RESTClient because it will use the saved cookies you created.

### Creating a more advanced New Item with RESTClient (using more fields) Example

Adding more fields to an item can be as simple as adding key/value string pairs, but the best way to figure out the formatting is to do a GET on an item that has the fields you are looking for (sometimes you may have to create a new item with these fields and do a get on it).

Nested Strings can be sent like this:

```
{ "key": { "subkey": "value", "subkey2": "value2" } }
```

Nested Arrays of Strings can be sent like this:

```
{ "key": [ "value1", "value2", "value3" ] }
```

Nested Arrays of key/value pairs can be sent like this:

```
{ "key": [ { "subkey1": "value", "subkey2": "value" }, { "subkey1.2": "value", "subkey2.2": "value" } ] }
```

Full Example{"parentId":"4f2add32e4b009ef29abd063",

```
"title":"Test Item Title",
"identifiers":[{"type":"Type","scheme":"Scheme","key":"Key1"}, {"type":"Type2","scheme":"Scheme2","key":"Key21"}],
"subTitle":"SubTitle1",
"alternateTitles":["Alternate Titles","Alternate Titles 2","Alternate Titles 3"],
"narratives":[{"type":"Type","content":"Content"}, {"type":"Type2","content":"Content2"}],
"summary":"Summary1",
"body":"Body",
"purpose":"Purpose1",
"rights":"Rights1",
"provenance":"Provenance1",
"materialRequestInstructions":"Material Request Instructions",
"edition":"Edition1",
"tableOfContents":"Table of contents",
"contacts":[{"type":"type","name":"name","highlighted":false}, {"type":"Type2","name":"Name2","highlighted":false}],
"webLinks":[{"type":"download","uri":"http://URI.com/","rel":"related","title":"Title","hidden":false}, {"type":"browseImage","uri":"http://URI2.com/","rel":"Related","title":"Title2","hidden":false}],
"tags":[{"name":"Tag1"}, {"name":"Tag2"}],
"dates":[{"type":"Date","dateString":"2012-01-01"}, {"type":"Date2","dateString":"2012"}]}
```