

API Jitterbit Variables

Introduction

This page covers Jitterbit variables that are available for Jitterbit Custom APIs, organized by informational variables that you get (read), and settings variables that you set (write):

Informational

- `jitterbit.api.request.body` and `jitterbit.api.request.body.*`
- `jitterbit.api.request.enum.body`
- `jitterbit.api.request.enum.headers`
- `jitterbit.api.request.enum.mvparameters`
- `jitterbit.api.request.enum.parameters`
- `jitterbit.api.request.headers.*`
- `jitterbit.api.request.headers.fulluri`
- `jitterbit.api.request.method`
- `jitterbit.api.request.mvparameters.*`
- `jitterbit.api.request.parameters.*`

Settings

- `jitterbit.api.response`
- `jitterbit.api.response.headers.*`
- `jitterbit.api.response.status_code`

Informational

`jitterbit.api.request.body` and `jitterbit.api.request.body.*`

Variable	<code>jitterbit.api.request.body</code> and <code>jitterbit.api.request.body.*</code>
Data Type	String
Description	<p>Looks at the payload/payloads submitted to the API. Note that for the majority of the APIs, you would expect only one plain payload and, as such, <code>jitterbit.api.request.body</code> is the variable to use (also known as <code>content-type:text/plain</code>).</p> <p>If you expect multiple payloads to be submitted at once, using the URL-encoded form (also known as <code>content-type:application/x-www-form-urlencoded</code>), as in the case of an API being used as the backend of a submission form (see http://www.w3.org/TR/html401/interact/forms.html), then you should be using <code>jitterbit.api.request.body.*</code>. As with <code>jitterbit.api.request.parameters.*</code>, <code>jitterbit.api.request.body.name</code> will be equal to <code>EStore</code> if the value of the form's field "name" was entered as <code>EStore</code>.</p>

`jitterbit.api.request.enum.body`

Variable	<code>jitterbit.api.request.enum.body</code>
Data Type	String
Description	Variable array used to dynamically iterate through all of the submitted parts of the payload/body (versus checking a specific part as with <code>jitterbit.api.request.body.*</code>). The usage is the same as with the <code>jitterbit.api.request.enum.parameters</code> .

`jitterbit.api.request.enum.headers`

Variable	<code>jitterbit.api.request.enum.headers</code>
Data Type	String

Related Articles

- [API Jitterbit Variables](#)
- [Global Variable Target](#)
- [Hosted Endpoint Jitterbit Variables](#)
- [Operation Jitterbit Variables](#)
- [Scripting Jitterbit Variables](#)
- [SFDC Jitterbit Variables](#)
- [Source Jitterbit Variables](#)
- [Specifying Source and Target Fields Dynamically](#)
- [Target Jitterbit Variables](#)
- [Text Jitterbit Variables](#)
- [Transformation Jitterbit Variables](#)
- [Using Global Variables](#)
- [Web Service Jitterbit Variables](#)

Related Topics

Content by label

There is no content with the specified labels

Last updated: Jul 17, 2019

Description	Variable array used to dynamically iterate through all of the request headers (versus checking specific headers as with <code>jitterbit.api.request.headers.*</code>). The usage is the same as with the <code>jitterbit.api.request.enum.parameters</code> and <code>jitterbit.api.request.enum.body</code> .
--------------------	---

`jitterbit.api.request.enum.mvparameters`

Variable	<code>jitterbit.api.request.enum.mvparameters</code>
Data Type	String
Description	Variable array used to dynamically iterate through all of the multi-value parameters (as opposed to checking each parameter specifically as <code>jitterbit.api.request.mvparameters.ProdID</code>).

`jitterbit.api.request.enum.parameters`

Variable	<code>jitterbit.api.request.enum.parameters</code>
Data Type	String
Description	<p>Variable array used to dynamically iterate through all of the submitted parameters (as opposed to checking each parameter specifically as <code>jitterbit.api.request.parameters.name</code>).</p> <p>This sample script appends all of the provided parameters to a new variable for later display back to the user:</p> <pre><trans> \$output = "URL Parameters:
\r\n"; enum = \$jitterbit.api.request.enum.parameters; i = 0; while(i<length(enum), name = enum[i]; \$output = \$output + "\$" + name + ": " + Get(name) + "
\r\n"; i = i+1;); if(i==0, \$output = \$output + "(none)
\r\n"); </trans></pre>

`jitterbit.api.request.headers.*`

Variable	<code>jitterbit.api.request.headers.*</code>
Data Type	String
Description	Variable used to look at the request headers submitted to the API; for example, <code>\$jitterbit.api.request.headers.x_forwarded_for</code> is the public IP of the box /user accessing the URL.

`jitterbit.api.request.headers.fulluri`

Variable	<code>jitterbit.api.request.headers.fulluri</code>
Data Type	String
Description	The URL that was called to trigger the Jitterbit OData or Custom API.

`jitterbit.api.request.method`

Variable	<code>jitterbit.api.request.method</code>
-----------------	---

Data Type	String
Description	The request method that was used to call the API.

jitterbit.api.request.mvparameters.*

Variable	<code>jitterbit.api.request.mvparameters.*</code>
Data Type	String
Description	Looks at the multi-values of the parameter submitted to the API directly via the URL and returns the values as an array with a space between each value. For example, if the URL is <code>https://jitterbitxx.na.jitterbit.org/dev/ProductAPIResponse?ProdID=abc&ProdID=abc1&ProdID=abc2</code> , then <code>jitterbit.api.request.mvparameters.ProdID</code> will be <code>abc abc1 abc2</code> .

jitterbit.api.request.parameters.*

Variable	<code>jitterbit.api.request.parameters.*</code>
Data Type	String
Description	Looks at the parameters submitted to the API directly via the URL; for example, <code>jitterbit.api.request.parameters.name</code> will be equal to <code>EStore</code> if the URL requested had <code>&name=EStore</code> . <div style="border: 1px solid #ccc; padding: 5px; margin: 5px 0;">NOTE: Multi-value URL parameters will return a string delimited by <code> </code> (3 pipes). To return multi-value URL parameters as an array, use the <code>jitterbit.api.request.mvparameters.*</code> variable instead.</div> For example, if the URL is <code>https://jitterbitxx.na.jitterbit.org/dev/ProductAPIResponse?ProdID=abc&ProdID=abc1&ProdID=abc2</code> , then <code>jitterbit.api.request.parameters.ProdID</code> will be <code>abc abc1 abc2</code> .

Settings

jitterbit.api.response

Variable	<code>jitterbit.api.response</code>
Data Type	String
Description	This variable needs to be set if your API is set to use a variable as the response instead of using the final target.

jitterbit.api.response.headers.*

Variable	<code>jitterbit.api.response.headers.*</code>
Data Type	String
Description	Used to set the response headers of the API. For example, set <code>jitterbit.api.response.headers.access_control_allow_origin="*" to override default CORS behavior and allow the API to be accessed by any domain in a cross-site manner.</code>

jitterbit.api.response.status_code

Variable	<code>jitterbit.api.response.status_code</code>
Data Type	String

Description	Provides the ability to override HTTP response code for custom APIs via Jitterbit script variable. Set the <code>jitterbit.api.response.status_code</code> variable in the script that is executed by a Custom API. This allows project authors to set a specific HTTP error code (along with actual payload information) versus relying on the system to return codes 200 or 500 based on default behavior.
--------------------	--