

FetchRemote

This function allows you to post and receive data from HTTP servers. The calls are asynchronous so you do not get an immediate result from the call, instead a callback function you specify is called when the download completes.

In the case when the call fails, a string containing "ERROR" followed by an integer containing the error reason will be passed to the callback function. The reason for failure will be similar to errors found with websites - file not found, server not found and timeouts.

If you are using fetchRemote to connect to a PHP script, you can use `file_get_contents("php://input")` to read the **postData** sent from this function.

Note: Client side function only works with the server the player is connected to unless the domain has been accepted with `requestBrowserDomains`

Warning: function won't trigger inside another fetchRemote function

Syntax

```
bool fetchRemote ( string URL, [ string queueName = "default" ], [ int connectionAttempts = 10, int connectTimeout = 10000 ],
    function callbackFunction, [ string postData = "", bool postIsBinary = false ], [ arguments... ] )
```

Required Arguments

- **URL:** A full URL in the format `http://hostname/path/file.ext`. A port can be specified with a colon followed by a port number appended to the hostname.
- **callbackFunction:** This is the function that should receive the data returned from the remote server. The callback argument list should be:
 - **responseData** - A string containing the remote response or "ERROR" if there was a problem
 - **error** - A number containing the error number or zero if there was no error. A list of possible error values are:
 - **1-89:** See cURL website or its mirror at cURL errors
 - **400-599:** See HTTP status codes
 - **1002:** Download aborted
 - **1003:** Failed to initialize
 - **1004:** Unable to parse url
 - **1005:** Unable to resolve host name
 - **1006:** Destination IP not allowed
 - **1007:** File error
- **arguments...** - The arguments that were passed into fetchRemote

Optional Arguments

ADDED/UPDATED IN VERSION 1.5.3-9.11270 :

- **queueName:** Name of the queue to use. Any name can be used. If not set, the queue name is "default". Requests in the same queue are processed in order, one at a time.

- **connectionAttempts:** Number of times to retry if the remote host does not respond. *In the case of a non-responding remote server, each connection attempt will timeout after 10 seconds. Therefore, the default setting of 10 connection attempts means it will be 100 seconds before your script gets a callback about the error. Reducing this value to 2 for example, will decrease that period to 20 seconds*
- **connectTimeout:** Number of milliseconds each connection attempt will take before timing out
- **postData:** A string specifying any data you want to send to the remote HTTP server.
- **postIsBinary :** A boolean specifying if the data is text, or binary.
- **arguments:** Any arguments you may want to pass to the callback.

ADDED/UPDATED IN VERSION 1.5.4-9.11342 :

Syntax

```
bool fetchRemote ( string URL[, table options ], callback callbackFunction[, table callbackArguments ] )
```

Required Arguments

- **URL:** A full URL in the format *http://hostname/path/file.ext*. A port can be specified with a colon followed by a port number appended to the hostname.
- **callbackFunction:** This is the function that should receive the data returned from the remote server. The callback argument list should be:
 - **responseData** - A string containing the remote response
 - **responseInfo** - A table containing:
 - **success** - A boolean indicating if the request was successful.
 - **statusCode** - An integer status/error code
 - **headers** - A table containing the HTTP response headers
 - **arguments...** - The arguments that were passed into fetchRemote

Optional Arguments

- **options:** A table containing any request options:
 - **queueName:** Name of the queue to use. Any name can be used. If not set, the queue name is "default". Requests in the same queue are processed in order, one at a time.
 - **connectionAttempts:** Number of times to retry if the remote host does not respond. *(Defaults to 10)*
 - **connectTimeout:** Number of milliseconds each connection attempt will take before timing out. *(Defaults to 10000)*
 - **postData:** A string specifying any data you want to send to the remote HTTP server.
 - **postIsBinary :** A boolean specifying if the data is text, or binary. *(Defaults to false)*
 - **method:** A string specifying the request method. *(Defaults to GET or POST)*
 - **headers:** A table containing HTTP request headers. *e.g. { Pragma="no-cache" }*
 - **maxRedirects:** An integer limiting the number of HTTP redirections to automatically follow. *(Defaults to 8)*
 - **username:** A string specifying the username for protected pages.
 - **password:** A string specifying the password for protected pages.

ADDED/UPDATED IN VERSION 1.5.4-9.11413 :

- - **formFields:** A table containing form items to submit. (for POST method only) *e.g. { name="bob", email="" }*
- **arguments:** A table containing arguments you may want to pass to the callback.

Returns

ADDED/UPDATED IN VERSION 1.5.7-9.20307 :

Returns a **request** value which can be used with getRemoteRequestInfo or abortRemoteRequest