NAME

CURLOPT_POST − request a HTTP POST

SYNOPSIS

#include <curl/curl.h>

CURLcode curl_easy_setopt(CURL *handle, CURLOPT_POST, long post);

DESCRIPTION

A parameter set to 1 tells libcurl to do a regular HTTP post. This will also make the library use a "Content-Type: application/x-www-form-urlencoded" header. (This is by far the most commonly used POST method).

Use one of *CURLOPT_POSTFIELDS(3)* or *CURLOPT_COPYPOSTFIELDS(3)* options to specify what data to post and *CURLOPT_POSTFIELDSIZE(3)* or *CURLOPT_POSTFIELDSIZE_LARGE(3)* to set the data size.

Optionally, you can provide data to POST using the *CURLOPT_READFUNCTION(3)* and *CURLOPT_READDATA(3)* options but then you must make sure to not set *CURLOPT_POSTFIELDS(3)* to anything but NULL. When providing data with a callback, you must transmit it using chunked transfer-encoding or you must set the size of the data with the *CURLOPT_POSTFIELDSIZE(3)* or *CURLOPT_POSTFIELDSIZE_LARGE(3)* options. To enable chunked encoding, you simply pass in the appropriate Transfer-Encoding header, see the post-callback.c example.

You can override the default POST Content-Type: header by setting your own with *CURLOPT_HTTPHEADER(3)*.

Using POST with HTTP 1.1 implies the use of a "Expect: 100-continue" header. You can disable this header with *CURLOPT_HTTPHEADER(3)* as usual.

If you use POST to a HTTP 1.1 server, you can send data without knowing the size before starting the POST if you use chunked encoding. You enable this by adding a header like "Transfer-Encoding: chunked" with *CURLOPT_HTTPHEADER(3)*. With HTTP 1.0 or without chunked transfer, you must specify the size in the request.

When setting *CURLOPT_POST(3)* to 1, it will automatically set *CURLOPT_NOBODY(3)* to 0.

If you issue a POST request and then want to make a HEAD or GET using the same re-used handle, you must explicitly set the new request type using *CURLOPT_NOBODY(3)* or *CURLOPT_HTTPGET(3)* or similar.

DEFAULT

0, disabled

PROTOCOLS

HTTP

EXAMPLE

TODO

AVAILABILITY

Along with HTTP

RETURN VALUE

Returns CURLE_OK if HTTP is supported, and CURLE_UNKNOWN_OPTION if not.

SEE ALSO

CURLOPT_POSTFIELDS(3), **CURLOPT_HTTPPOST**(3),