NAME

CURLOPT_POSTFIELDS − specify data to POST to server

SYNOPSIS

```
#include <curl/curl.h>

CURLcode curl_easy_setopt(CURL *handle, CURLOPT_POSTFIELDS, char *postdata);
```

DESCRIPTION

Pass a char * as parameter, pointing to the full data to send in a HTTP POST operation. You must make sure that the data is formatted the way you want the server to receive it. libcurl will not convert or encode it for you in any way. For example, the web server may assume that this data is url-encoded.

The data pointed to is NOT copied by the library: as a consequence, it must be preserved by the calling application until the associated transfer finishes. This behaviour can be changed (so libcurl does copy the data) by setting the *CURLOPT_COPYPOSTFIELDS(3)* option.

This POST is a normal application/x-www-form-urlencoded kind (and libcurl will set that Content-Type by default when this option is used), which is commonly used by HTML forms. Change Content-Type with *CURLOPT_HTTPHEADER(3)*.

You can use *curl_easy_escape(3)* to url-encode your data, if necessary. It returns a pointer to an encoded string that can be passed as *postdata*.

Using *CURLOPT_POSTFIELDS(3)* implies *CURLOPT_POST(3)*.

If *CURLOPT_POSTFIELDS(3)* is explicitly set to NULL then libcurl will get the POST data from the read callback. If you want to send a zero-byte POST set *CURLOPT_POSTFIELDS(3)* to an empty string, or set *CURLOPT_POST(3)* to 1 and *CURLOPT_POSTFIELDSIZE(3)* to 0.

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.

To make multipart/formdata posts (aka RFC2388-posts), check out the *CURLOPT_HTTPPOST(3)* option combined with *curl_formadd(3)*.

DEFAULT

NULL

PROTOCOLS

HTTP

EXAMPLE

```
CURL *curl = curl_easy_init();
if(curl) {
  const char *data = "data to send";

  curl_easy_setopt(curl, CURLOPT_URL, "http://example.com");

  /* size of the POST data */
  curl_easy_setopt(curl, CURLOPT_POSTFIELDSIZE, 12L);

  /* pass in a pointer to the data - libcurl will not copy */
  curl_easy_setopt(curl, CURLOPT_POSTFIELDS, data);

  curl_easy_perform(curl);
}
```

AVAILABILITY

Always

RETURN VALUE

Returns CURLE_OK

SEE ALSO

CURLOPT_POSTFIELDSIZE(3), **CURLOPT_READFUNCTION**(3),