

**NAME**

CURLOPT\_NOBODY – do the download request without getting the body

**SYNOPSIS**

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

```
CURLcode curl_easy_setopt(CURL *handle, CURLOPT_NOBODY, long opt);
```

**DESCRIPTION**

A long parameter set to 1 tells libcurl to not include the body-part in the output when doing what would otherwise be a download. For HTTP(S), this makes libcurl do a HEAD request. For most other protocols it means not asking to transfer the body data.

For HTTP operations when *CURLOPT\_NOBODY(3)* has been set, disabling this option (with 0) makes it a GET again – only if the method is still set to be HEAD. The proper way to get back to a GET request is to set *CURLOPT\_HTTPGET(3)* and for other methods, use the POST or UPLOAD options.

Enabling *CURLOPT\_NOBODY(3)* means asking for a download without a body.

If you do a transfer with HTTP that involves a method other than HEAD, you get a body (unless the resource and server sends a zero byte body for the specific URL you request).

**DEFAULT**

0, the body is transferred

**PROTOCOLS**

This functionality affects all supported protocols

**EXAMPLE**

```
int main(void)
{
    CURL *curl = curl_easy_init();
    if(curl) {
        CURLcode result;
        curl_easy_setopt(curl, CURLOPT_URL, "https://example.com");

        /* get us the resource without a body - use HEAD */
        curl_easy_setopt(curl, CURLOPT_NOBODY, 1L);

        /* Perform the request */
        result = curl_easy_perform(curl);
        curl_easy_cleanup(curl);
    }
}
```

**AVAILABILITY**

Added in curl 7.1

**RETURN VALUE**

*curl\_easy\_setopt(3)* returns a CURLcode indicating success or error.

CURLE\_OK (0) means everything was OK, non-zero means an error occurred, see *libcurl-errors(3)*.

**SEE ALSO**

**CURLOPT\_HTTPGET(3), CURLOPT\_MIMEPOST(3), CURLOPT\_POSTFIELDS(3), CURLOPT\_REQUEST\_TARGET(3), CURLOPT\_UPLOAD(3)**