NAME

curl_multi_cleanup − close down a multi session

SYNOPSIS

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

CURLMcode curl_multi_cleanup(CURLM *multi_handle);
```

DESCRIPTION

This function is the opposite of *curl_multi_init(3)*. Cleans up and removes a whole multi stack. It does not free or touch any individual easy handles in any way − they still need to be closed individually, using the usual *curl_easy_cleanup(3)* way. The order of cleaning up should be:

1 − *curl_multi_remove_handle(3)* before any easy handles are cleaned up

2 − *curl_easy_cleanup(3)* can now be called independently since the easy handle is no longer connected to the multi handle

3 − *curl_multi_cleanup(3)* should be called when all easy handles are removed

When this function is called, remaining entries in the connection pool held by the multi handle are shut down, which might trigger calls to the *CURLMOPT_SOCKETFUNCTION(3)* callback.

Passing in a NULL pointer in *multi_handle* makes this function return CURLM_BAD_HANDLE immediately with no other action.

Any use of the **multi_handle** after this function has been called and have returned, is illegal.

PROTOCOLS

This functionality affects all supported protocols

EXAMPLE

```
int main(void)
{
  CURLM *multi = curl_multi_init();

  /* when the multi transfer is done ... */

  /* remove all easy handles, then: */
  curl_multi_cleanup(multi);
}
```

AVAILABILITY

Added in curl 7.9.6

RETURN VALUE

This function returns a CURLMcode indicating success or error.

CURLM_OK (0) means everything was OK, non−zero means an error occurred, see *libcurl−errors(3)*.

SEE ALSO

curl_easy_cleanup(3), **curl_easy_init**(3), **curl_multi_get_handles**(3), **curl_multi_init**(3)