

NAME

curl_share_cleanup – close a shared object

SYNOPSIS

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

```
CURLSHcode curl_share_cleanup(CURLSH *share_handle);
```

DESCRIPTION

This function deletes a shared object. The share handle cannot be used anymore when this function has been called. The share fails the call if it is still being used in any easy handle.

Passing in a NULL pointer in *share_handle* makes this function return immediately with no action.

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

For applications that use a share in several threads, it is critical that the destruction of the share is only done when all other threads have stopped using it. While libcurl tracks how many easy handles are using a share, it can not observe how many pointers to the share the application has.

PROTOCOLS

This functionality affects all supported protocols

EXAMPLE

```
int main(void)
{
    CURLSHcode sh;
    CURLSH *share = curl_share_init();
    sh = curl_share_setopt(share, CURLSHOPT_SHARE, CURL_LOCK_DATA_CONNECT);
    /* use the share, then ... */
    curl_share_cleanup(share);
}
```

AVAILABILITY

Added in curl 7.10

RETURN VALUE

CURLSHE_OK (zero) means that the option was set properly, non-zero means an error occurred as *<curl/curl.h>* defines. See the *libcurl-errors(3)* man page for the full list with descriptions. If an error occurs, then the share object is not deleted.

SEE ALSO

curl_share_init(3), **curl_share_setopt(3)**