

**NAME**

curl\_multi\_add\_handle - add an easy handle to a multi session

**SYNOPSIS**

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

```
CURLMcode curl_multi_add_handle(CURLM *multi_handle, CURL *easy_handle);
```

**DESCRIPTION**

Adds a standard easy handle to the multi stack. This function call will make this *multi\_handle* control the specified *easy\_handle*. Furthermore, libcurl now initiates the connection associated with the specified *easy\_handle*.

When an easy handle has been added to a multi stack, you can not and you must not use *curl\_easy\_perform(3)* on that handle!

If the easy handle is not set to use a shared (CURLOPT\_SHARE) or global DNS cache (CURLOPT\_DNS\_USE\_GLOBAL\_CACHE), it will be made to use the DNS cache that is shared between all easy handles within the multi handle when *curl\_multi\_add\_handle(3)* is called.

The easy handle will remain added until you remove it again with *curl\_multi\_remove\_handle(3)*. You should remove the easy handle from the multi stack before you terminate first the easy handle and then the multi handle:

1 - *curl\_multi\_remove\_handle(3)*

2 - *curl\_easy\_cleanup(3)*

3 - *curl\_multi\_cleanup(3)*

**RETURN VALUE**

CURLMcode type, general libcurl multi interface error code.

**SEE ALSO**

**curl\_multi\_cleanup(3), curl\_multi\_init(3)**