

NAME

curl_easy_init – create an easy handle

SYNOPSIS

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

```
CURL *curl_easy_init();
```

DESCRIPTION

This function allocates and returns an easy handle. Such a handle is used as input to other functions in the easy interface. This call must have a corresponding call to *curl_easy_cleanup(3)* when the operation is complete.

The easy handle is used to hold and control a single network transfer. It is encouraged to reuse easy handles for repeated transfers.

An alternative way to get a new easy handle is to duplicate an already existing one with *curl_easy_duphandle(3)*, which has the upside that it gets all the options that were set in the source handle set in the new copy as well.

If you did not already call *curl_global_init(3)* before calling this function, *curl_easy_init(3)* does it automatically. This can be lethal in multi-threaded cases for platforms where *curl_global_init(3)* is not thread-safe, and it may then result in resource problems because there is no corresponding cleanup.

You are strongly advised to not allow this automatic behavior, by calling *curl_global_init(3)* yourself properly. See the description in *libcurl(3)* of global environment requirements for details of how to use this function.

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");
        result = curl_easy_perform(curl);
        curl_easy_cleanup(curl);
    }
}
```

AVAILABILITY

Added in curl 7.1

RETURN VALUE

If this function returns NULL, something went wrong and you cannot use the other curl functions.

SEE ALSO

curl_easy_cleanup(3), curl_easy_duphandle(3), curl_easy_perform(3), curl_easy_reset(3), curl_global_init(3), curl_multi_init(3)