

**NAME**

CURLMOPT\_PIPELINING – enable/disable HTTP pipelining

**SYNOPSIS**

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

```
CURLMcode curl_multi_setopt(CURLM *handle, CURLMOPT_PIPELINING, long bits);
```

**DESCRIPTION**

Set the **bits** parameter to 1 to make libcurl use HTTP pipelining for HTTP/1.1 transfers done using this multi handle, as far as possible. This means that if you add a second request that can use an already existing connection, the second request will be "piped" on the same connection rather than being executed in parallel.

When using pipelining, there are also several other related options that are interesting to tweak and adjust to alter how libcurl spreads out requests on different connections or not etc.

Starting in 7.43.0, the **bits** parameter's bit 1 also has a meaning and libcurl is now offering symbol names for the bits:

CURLPIPE\_NOHING

Default, which means doing no attempts at pipelining or multiplexing.

CURLPIPE\_HTTP1

If this bit is set, libcurl will try to pipeline HTTP/1.1 requests on connections that are already established and in use to hosts.

CURLPIPE\_MULTIPLEX

If this bit is set, libcurl will try to multiplex the new transfer over an existing connection if possible. This requires HTTP/2.

**DEFAULT**

0 (off)

**PROTOCOLS**

HTTP(S)

**EXAMPLE**

TODO

**AVAILABILITY**

Added in 7.16.0. Multiplex support bit added in 7.43.0.

**RETURN VALUE**

Returns CURLM\_OK if the option is supported, and CURLM\_UNKNOWN\_OPTION if not.

**SEE ALSO**

CURLMOPT\_MAX\_PIPELINE\_LENGTH(3), CURLMOPT\_PIPELINING\_SITE\_BL(3), CURLMOPT\_CONTENT\_LENGTH\_PENALTY\_SIZE(3), CURLMOPT\_CHUNK\_LENGTH\_PENALTY\_SIZE(3), CURLMOPT\_MAX\_HOST\_CONNECTIONS(3), CURLMOPT\_MAXCONNECTS(3), CURLMOPT\_MAX\_HOST\_CONNECTIONS(3),