NAME

CURLOPT_ACCEPT_ENCODING − enables automatic decompression of HTTP downloads

SYNOPSIS

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

CURLcode curl_easy_setopt(CURL *handle, CURLOPT_ACCEPT_ENCODING, char *enc);
```

DESCRIPTION

Pass a char * argument specifying what encoding you'd like.

Sets the contents of the Accept-Encoding: header sent in a HTTP request, and enables decoding of a response when a Content-Encoding: header is received. Three encodings are supported: *identity*, which does nothing, *deflate* which requests the server to compress its response using the zlib algorithm, and *gzip* which requests the gzip algorithm.

If a zero-length string is set like "", then an Accept-Encoding: header containing all built-in supported encodings is sent.

You can also opt to just include the Accept-Encoding: header in your request with *CURLOPT_HTTPHEADER(3)* but then there will be no automatic decompressing when receiving data.

This is a request, not an order; the server may or may not do it. This option must be set (to any non-NULL value) or else any unsolicited encoding done by the server is ignored. See the special file lib/README.encoding for further details.

DEFAULT

NULL

PROTOCOLS

HTTP

EXAMPLE

```
CURL *curl = curl_easy_init();
if(curl) {
  curl_easy_setopt(curl, CURLOPT_URL, "http://example.com");

  /* enable all supported built-in compressions */
  curl_easy_setopt(curl, CURLOPT_ACCEPT_ENCODING, "");

  /* Perform the request */
  curl_easy_perform(curl);
}
```

AVAILABILITY

This option was called CURLOPT_ENCODING before 7.21.6

RETURN VALUE

Returns CURLE_OK if the option is supported, CURLE_UNKNOWN_OPTION if not, or CURLE_OUT_OF_MEMORY if there was insufficient heap space.

SEE ALSO

CURLOPT_TRANSFER_ENCODING(3), **CURLOPT_HTTPHEADER**(3), **CURLOPT_HTTP_CONTENT_DECODING**(3),