NAME
>   libcurl-easy – easy interface overview

DESCRIPTION
>   When using libcurl's "easy" interface you init your session and get a handle (often referred to as an "easy handle"), which you use as input to the easy interface functions you use. Use *curl_easy_init(3)* to get the handle.
>
>   You continue by setting all the options you want in the upcoming transfer, the most important among them is the URL itself (you can't transfer anything without a specified URL as you may have figured out yourself). You might want to set some callbacks as well that will be called from the library when data is available etc. *curl_easy_setopt(3)* is used for all this.
>
>   *CURLOPT_URL(3)* is only option you really must set, as otherwise there can be no transfer. Another commonly used option is *CURLOPT_VERBOSE(3)* that will help you see what libcurl is doing under the hood, very useful when debugging for example. The *curl_easy_setopt(3)* man page has a full index of the over 200 available options.
>
>   If you at any point would like to blank all previously set options for a single easy handle, you can call *curl_easy_reset(3)* and you can also make a clone of an easy handle (with all its set options) using *curl_easy_duphandle(3)*.
>
>   When all is setup, you tell libcurl to perform the transfer using *curl_easy_perform(3)*. It will then do the entire operation and won't return until it is done (successfully or not).
>
>   After the transfer has been made, you can set new options and make another transfer, or if you're done, cleanup the session by calling *curl_easy_cleanup(3)*. If you want persistent connections, you don't cleanup immediately, but instead run ahead and perform other transfers using the same easy handle.

SEE ALSO
>   **curl_easy_init**(3),**curl_easy_cleanup**(3),**curl_easy_setopt**(3),   **libcurl-errors**(3),   **libcurl-multi**(3), **libcurl**(3)