rl_on_new_line()

The `rl_on_new_line()` function is used to inform the Readline library that the cursor is now logically on a new line. This is typically necessary when managing line redisplay manually.

It's included in the "readline/readline.h" system header.

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

int rl_on_new_line(void);
```

Returns 0 on success. In case of failure returns -1.

It's useful when asynchronous contexts display information on the terminal while the user is typing (e.g. A signal handler is triggered).

When building a shell, printing messages while the user is typing requires restoring the prompt afterward:

- Tell Readline the cursor is on a new line with "rl_on_new_line()"
- Rewrite the line with "rl_replace_line()"
- Re-render with "rl_redisplay()"

Example Usage

```
#include <readline/readline.h>
#include <stdio.h>

void notify(void)
{
    printf("\n[New Message]\n");

        // tell readline we're on a new line
    rl_on_new_line();
    // clear current input line
    rl_replace_line("", 0);
    // redisplay prompt
    rl_redisplay();
}
```