NAME

ne_buffer_destroy, ne_buffer_finish **-** destroy a buffer object

SYNOPSIS

#include <ne_string.h>

void ne_buffer_destroy(ne_buffer ***_buf_);**

char *ne_buffer_finish(ne_buffer ***_buf_);**

DESCRIPTION

ne_buffer_destroy frees all memory associated with the buffer. **ne_buffer_finish** frees the buffer structure, but not the actual string stored in the buffer, which is returned and must be **free**()d by the caller.

Any use of the buffer object after calling either of these functions gives undefined behaviour.

RETURN VALUE

ne_buffer_finish returns the **malloc**-allocated string stored in the buffer.

EXAMPLES

An example use of **ne_buffer_finish**; the **duplicate** function returns a string made up of _n_ copies of _str_:

```
static char *duplicate(int n, const char *str)
{
  ne_buffer *buf = ne_buffer_create();
  while (n--) {
    ne_buffer_zappend(buf, str);
  }
  return ne_buffer_finish(buf);
}
```

SEE ALSO

ne_buffer, ne_buffer_create, ne_buffer_zappend

AUTHOR

Joe Orton <neon@lists.manyfish.co.uk>
Author.

COPYRIGHT