

NAME

xcb_alloc_color_cells -

SYNOPSIS

```
#include <xcb/xproto.h>
```

Request function

```
xcb_alloc_color_cells_cookie_t xcb_alloc_color_cells(xcb_connection_t *conn, uint8_t contiguous,
    xcb_colormap_t cmap, uint16_t colors, uint16_t planes);
```

Reply datastructure

```
typedef struct xcb_alloc_color_cells_reply_t {
    uint8_t response_type;
    uint8_t pad0;
    uint16_t sequence;
    uint32_t length;
    uint16_t pixels_len;
    uint16_t masks_len;
    uint8_t pad1[20];
} xcb_alloc_color_cells_reply_t;
```

Reply function

```
xcb_alloc_color_cells_reply_t *xcb_alloc_color_cells_reply(xcb_connection_t *conn,
    xcb_alloc_color_cells_cookie_t cookie, xcb_generic_error_t **e);
```

Reply accessors

```
uint32_t *xcb_alloc_color_cells_pixels(const xcb_alloc_color_cells_request_t *reply);
```

```
int xcb_alloc_color_cells_pixels_length(const xcb_alloc_color_cells_reply_t *reply);
```

```
xcb_generic_iterator_t xcb_alloc_color_cells_pixels_end(const xcb_alloc_color_cells_reply_t *reply);
```

```
uint32_t *xcb_alloc_color_cells_masks(const xcb_alloc_color_cells_request_t *reply);
```

```
int xcb_alloc_color_cells_masks_length(const xcb_alloc_color_cells_reply_t *reply);
```

```
xcb_generic_iterator_t xcb_alloc_color_cells_masks_end(const xcb_alloc_color_cells_reply_t *reply);
```

REQUEST ARGUMENTS

<i>conn</i>	The XCB connection to X11.
<i>contiguous</i>	TODO: NOT YET DOCUMENTED.
<i>cmap</i>	TODO: NOT YET DOCUMENTED.
<i>colors</i>	TODO: NOT YET DOCUMENTED.
<i>planes</i>	TODO: NOT YET DOCUMENTED.

REPLY FIELDS

<i>response_type</i>	The type of this reply, in this case <i>XCB_ALLOC_COLOR_CELLS</i> . This field is also present in the <i>xcb_generic_reply_t</i> and can be used to tell replies apart from each other.
<i>sequence</i>	The sequence number of the last request processed by the X11 server.
<i>length</i>	The length of the reply, in words (a word is 4 bytes).
<i>pixels_len</i>	TODO: NOT YET DOCUMENTED.
<i>masks_len</i>	TODO: NOT YET DOCUMENTED.

DESCRIPTION

RETURN VALUE

Returns an *xcb_alloc_color_cells_cookie_t*. Errors have to be handled when calling the reply function *xcb_alloc_color_cells_reply*.

If you want to handle errors in the event loop instead, use *xcb_alloc_color_cells_unchecked*. See **xcb-requests(3)** for details.

ERRORS

This request does never generate any errors.

SEE ALSO

AUTHOR

Generated from xproto.xml. Contact xcb@lists.freedesktop.org for corrections and improvements.