

**NAME**

xcb\_set\_pointer\_mapping -

**SYNOPSIS**

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

**Request function**

```
xcb_set_pointer_mapping_cookie_t xcb_set_pointer_mapping(xcb_connection_t *conn,
    uint8_t map_len, const uint8_t *map);
```

**Reply datastructure**

```
typedef struct xcb_set_pointer_mapping_reply_t {
    uint8_t response_type;
    uint8_t status;
    uint16_t sequence;
    uint32_t length;
} xcb_set_pointer_mapping_reply_t;
```

**Reply function**

```
xcb_set_pointer_mapping_reply_t *xcb_set_pointer_mapping_reply(xcb_connection_t *conn,
    xcb_set_pointer_mapping_cookie_t cookie, xcb_generic_error_t **e);
```

**REQUEST ARGUMENTS**

*conn*           The XCB connection to X11.

*map\_len*        TODO: NOT YET DOCUMENTED.

*map*            TODO: NOT YET DOCUMENTED.

**REPLY FIELDS**

*response\_type* The type of this reply, in this case *XCB\_SET\_POINTER\_MAPPING*. This field is also present in the *xcb\_generic\_reply\_t* and can be used to tell replies apart from each other.

*sequence*       The sequence number of the last request processed by the X11 server.

*length*         The length of the reply, in words (a word is 4 bytes).

*status*         One of the following values:

*XCB\_MAPPING\_STATUS\_SUCCESS*

TODO: NOT YET DOCUMENTED.

*XCB\_MAPPING\_STATUS\_BUSY*

TODO: NOT YET DOCUMENTED.

*XCB\_MAPPING\_STATUS\_FAILURE*

TODO: NOT YET DOCUMENTED.

TODO: NOT YET DOCUMENTED.

## DESCRIPTION

## RETURN VALUE

Returns an *xcb\_set\_pointer\_mapping\_cookie\_t*. Errors have to be handled when calling the reply function *xcb\_set\_pointer\_mapping\_reply*.

If you want to handle errors in the event loop instead, use *xcb\_set\_pointer\_mapping\_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](mailto:xcb@lists.freedesktop.org) for corrections and improvements.