

**NAME**

xcb\_get\_pointer\_mapping -

**SYNOPSIS**

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

**Request function**

```
xcb_get_pointer_mapping_cookie_t xcb_get_pointer_mapping(xcb_connection_t *conn,
```

**Reply datastructure**

```
typedef struct xcb_get_pointer_mapping_reply_t {
    uint8_t response_type;
    uint8_t map_len;
    uint16_t sequence;
    uint32_t length;
    uint8_t pad0[24];
} xcb_get_pointer_mapping_reply_t;
```

**Reply function**

```
xcb_get_pointer_mapping_reply_t *xcb_get_pointer_mapping_reply(xcb_connection_t *conn,
    xcb_get_pointer_mapping_cookie_t cookie, xcb_generic_error_t **e);
```

**Reply accessors**

```
uint8_t *xcb_get_pointer_mapping_map(const xcb_get_pointer_mapping_request_t *reply);
```

```
int xcb_get_pointer_mapping_map_length(const xcb_get_pointer_mapping_reply_t *reply);
```

```
xcb_generic_iterator_t xcb_get_pointer_mapping_map_end(const xcb_get_pointer_mapping_reply_t
    *reply);
```

**REQUEST ARGUMENTS**

*conn*            The XCB connection to X11.

**REPLY FIELDS**

*response\_type*    The type of this reply, in this case `XCB_GET_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).

*map\_len*        TODO: NOT YET DOCUMENTED.

## **DESCRIPTION**

### **RETURN VALUE**

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

If you want to handle errors in the event loop instead, use *xcb\_get\_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.