

**NAME**

xcb\_xkb\_get\_indicator\_map -

**SYNOPSIS**

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

**Request function**

```
xcb_xkb_get_indicator_map_cookie_t xcb_xkb_get_indicator_map(xcb_connection_t *conn,
    xcb_xkb_device_spec_t deviceSpec, uint32_t which);
```

**Reply datastructure**

```
typedef struct xcb_xkb_get_indicator_map_reply_t {
    uint8_t response_type;
    uint8_t deviceID;
    uint16_t sequence;
    uint32_t length;
    uint32_t which;
    uint32_t realIndicators;
    uint8_t nIndicators;
    uint8_t pad0[15];
} xcb_xkb_get_indicator_map_reply_t;
```

**Reply function**

```
xcb_xkb_get_indicator_map_reply_t *xcb_xkb_get_indicator_map_reply(xcb_connection_t *conn,
    xcb_xkb_get_indicator_map_cookie_t cookie, xcb_generic_error_t **e);
```

**Reply accessors**

```
xcb_xkb_indicator_map_t *xcb_xkb_get_indicator_map_maps(const
    xcb_xkb_get_indicator_map_request_t *reply);
```

```
int xcb_xkb_get_indicator_map_maps_length(const xcb_xkb_get_indicator_map_reply_t *reply);
```

```
xcb_xkb_indicator_map_iterator_t xcb_xkb_get_indicator_map_maps_iterator(const
    xcb_xkb_get_indicator_map_reply_t *reply);
```

**REQUEST ARGUMENTS**

*conn*            The XCB connection to X11.

*deviceSpec*    TODO: NOT YET DOCUMENTED.

*which*            TODO: NOT YET DOCUMENTED.

## REPLY FIELDS

*response\_type*   The type of this reply, in this case *XCB\_XKB\_GET\_INDICATOR\_MAP*. 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).

*deviceID*        TODO: NOT YET DOCUMENTED.

*which*            TODO: NOT YET DOCUMENTED.

*realIndicators*   TODO: NOT YET DOCUMENTED.

*nIndicators*     TODO: NOT YET DOCUMENTED.

## DESCRIPTION

### RETURN VALUE

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

If you want to handle errors in the event loop instead, use *xcb\_xkb\_get\_indicator\_map\_unchecked*. See **xcb-requests(3)** for details.

## ERRORS

This request does never generate any errors.

## SEE ALSO

### AUTHOR

Generated from xkb.xml. Contact [xcb@lists.freedesktop.org](mailto:xcb@lists.freedesktop.org) for corrections and improvements.