

**NAME**

xcb\_input\_get\_device\_key\_mapping -

**SYNOPSIS**

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

**Request function**

```
xcb_input_get_device_key_mapping_cookie_t
xcb_input_get_device_key_mapping(xcb_connection_t *conn, uint8_t device_id,
xcb_input_key_code_t first_keycode, uint8_t count);
```

**Reply datastructure**

```
typedef struct xcb_input_get_device_key_mapping_reply_t {
    uint8_t response_type;
    uint8_t xi_reply_type;
    uint16_t sequence;
    uint32_t length;
    uint8_t keysyms_per_keycode;
    uint8_t pad0[23];
} xcb_input_get_device_key_mapping_reply_t;
```

**Reply function**

```
xcb_input_get_device_key_mapping_reply_t
*xcb_input_get_device_key_mapping_reply(xcb_connection_t *conn,
xcb_input_get_device_key_mapping_cookie_t cookie, xcb_generic_error_t **e);
```

**Reply accessors**

```
xcb_keysym_t *xcb_input_get_device_key_mapping_keysyms(const
xcb_input_get_device_key_mapping_request_t *reply);

int xcb_input_get_device_key_mapping_keysyms_length(const
xcb_input_get_device_key_mapping_reply_t *reply);

xcb_generic_iterator_t xcb_input_get_device_key_mapping_keysyms_end(const
xcb_input_get_device_key_mapping_reply_t *reply);
```

**REQUEST ARGUMENTS**

*conn*            The XCB connection to X11.

