

NAME

xcb_list_extensions -

SYNOPSIS

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

Request function

```
xcb_list_extensions_cookie_t xcb_list_extensions(xcb_connection_t *conn,
```

Reply datastructure

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

Reply function

```
xcb_list_extensions_reply_t *xcb_list_extensions_reply(xcb_connection_t *conn,
    xcb_list_extensions_cookie_t cookie, xcb_generic_error_t **e);
```

Reply accessors

```
int xcb_list_extensions_names_length(const xcb_list_extensions_reply_t *reply);
```

```
xcb_str_iterator_t xcb_list_extensions_names_iterator(const xcb_list_extensions_reply_t *reply);
```

REQUEST ARGUMENTS

conn The XCB connection to X11.

REPLY FIELDS

response_type The type of this reply, in this case *XCB_LIST_EXTENSIONS*. 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).

names_len TODO: NOT YET DOCUMENTED.

DESCRIPTION**RETURN VALUE**

Returns an *xcb_list_extensions_cookie_t*. Errors have to be handled when calling the reply function *xcb_list_extensions_reply*.

If you want to handle errors in the event loop instead, use *xcb_list_extensions_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.