XGrabKeyboard, XUngrabKeyboard – grab the keyboard

int XGrabKeyboard(display, grab_window, owner_events, pointer_mode, keyboard_mode, time)
    Display *display;
    Window grab_window;
    Bool owner_events;
    int pointer_mode, keyboard_mode;
    Time time;
XUngrabKeyboard(display, time)
    Display *display;
    Time time;

Display *display; Specifies the connection to the X server.
grab_window Specifies the grab window.
keyboard_mode Specifies further processing of keyboard events. You can pass GrabModeSync or GrabModeAsync.
owner_events Specifies a Boolean value that indicates whether the keyboard events are to be reported as usual.
pointer_mode Specifies further processing of pointer events. You can pass GrabModeSync or GrabModeAsync.
time Specifies the time. You can pass either a timestamp or CurrentTime.

The XGrabKeyboard function actively grabs control of the keyboard and generates FocusIn and FocusOut events. Further key events are reported only to the grabbing client. XGrabKeyboard overrides any active keyboard grab by this client. If owner_events is False, all generated key events are reported with respect to grab_window. If owner_events is True and if a generated key event would normally be reported to this client, it is reported normally; otherwise, the event is reported with respect to the grab_window. Both KeyPress and KeyRelease events are always reported, independent of any event selection made by the client.

If the keyboard_mode argument is GrabModeAsync, keyboard event processing continues as usual. If the keyboard is currently frozen by this client, then processing of keyboard events is resumed. If the keyboard_mode argument is GrabModeSync, the state of the keyboard (as seen by client applications) appears to freeze, and the X server generates no further keyboard events until the grabbing client issues a releasing XAllowEvents call or until the keyboard grab is released. Actual keyboard changes are not lost while the keyboard is frozen; they are simply queued in the server for later processing.

If pointer_mode is GrabModeAsync, pointer event processing is unaffected by activation of the grab. If pointer_mode is GrabModeSync, the state of the pointer (as seen by client applications) appears to freeze, and the X server generates no further pointer events until the grabbing client issues a releasing XAllowEvents call or until the keyboard grab is released. Actual pointer changes are not lost while the pointer is frozen; they are simply queued in the server for later processing.

If the keyboard is actively grabbed by some other client, XGrabKeyboard fails and returns AlreadyGrabbed. If grab_window is not viewable, it fails and returns GrabNotViewable. If the keyboard is frozen by an active grab of another client, it fails and returns GrabFrozen. If the specified time is earlier than the last-keyboard-grab time or later than the current X server time, it fails and returns GrabInvalidTime. Otherwise, the last-keyboard-grab time is set to the specified time (CurrentTime is replaced by the current X server time).

XGrabKeyboard can generate BadValue and BadWindow errors.

The XUngrabKeyboard function releases the keyboard and any queued events if this client has it actively grabbed from either XGrabKeyboard or XGrabKey. XUngrabKeyboard does not release the keyboard and any queued events if the specified time is earlier than the last-keyboard-grab time or is later than the current X server time. It also generates FocusIn and FocusOut events. The X server automatically
performs an UngrabKeyboard request if the event window for an active keyboard grab becomes not viewable.

**BadValue** Some numeric value falls outside the range of values accepted by the request. Unless a specific range is specified for an argument, the full range defined by the argument’s type is accepted. Any argument defined as a set of alternatives can generate this error.  **BadWindow** A value for a Window argument does not name a defined Window.

XAllowEvents(3X11), XGrabButton(3X11), XGrabKey(3X11), XGrabPointer(3X11)

Xlib – C Language X Interface