

**NAME**

XSetInputFocus, XGetInputFocus - control input focus

**SYNTAX**

```
int XSetInputFocus(Display *display, Window focus, int revert_to, Time time);
```

```
int XGetInputFocus(Display *display, Window *focus_return, int *revert_to_return);
```

**ARGUMENTS**

*display* Specifies the connection to the X server.

*focus* Specifies the window, **PointerRoot**, or **None**.

*focus\_return* Returns the focus window, **PointerRoot**, or **None**.

*revert\_to* Specifies where the input focus reverts to if the window becomes not viewable. You can pass **RevertToParent**, **RevertToPointerRoot**, or **RevertToNone**.

*revert\_to\_return* Returns the current focus state (**RevertToParent**, **RevertToPointerRoot**, or **RevertToNone**).

*time* Specifies the time. You can pass either a timestamp or **CurrentTime**.

**DESCRIPTION**

The **XSetInputFocus** function changes the input focus and the last-focus-change time. It has no effect if the specified time is earlier than the current last-focus-change time or is later than the current X server time. Otherwise, the last-focus-change time is set to the specified time (**CurrentTime** is replaced by the current X server time). **XSetInputFocus** causes the X server to generate **FocusIn** and **FocusOut** events.

Depending on the focus argument, the following occurs:

- ⊕ If focus is **None**, all keyboard events are discarded until a new focus window is set, and the *revert\_to* argument is ignored.
- ⊕ If focus is a window, it becomes the keyboard's focus window. If a generated keyboard event would normally be reported to this window or one of its inferiors, the event is reported as usual. Otherwise, the event is reported relative to the focus window.

- ⊕ If focus is **PointerRoot**, the focus window is dynamically taken to be the root window of whatever screen the pointer is on at each keyboard event. In this case, the `revert_to` argument is ignored.

The specified focus window must be viewable at the time **XSetInputFocus** is called, or a **BadMatch** error results. If the focus window later becomes not viewable, the X server evaluates the `revert_to` argument to determine the new focus window as follows:

- ⊕ If `revert_to` is **RevertToParent**, the focus reverts to the parent (or the closest viewable ancestor), and the new `revert_to` value is taken to be **RevertToNone**.
- ⊕ If `revert_to` is **RevertToPointerRoot** or **RevertToNone**, the focus reverts to **PointerRoot** or **None**, respectively. When the focus reverts, the X server generates **FocusIn** and **FocusOut** events, but the last-focus-change time is not affected.

**XSetInputFocus** can generate **BadMatch**, **BadValue**, and **BadWindow** errors.

The **XGetInputFocus** function returns the focus window and the current focus state.

## DIAGNOSTICS

**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.

## SEE ALSO

**XWarpPointer(3)**

*Xlib - C Language X Interface*