**NAME**
     XIBarrierReleasePointer, XIBarrierReleasePointers - allow pointer movement across a pointer barrier

**SYNOPSIS**
     #include <X11/extensions/XInput2.h>

     int XIBarrierReleasePointer( Display *display,
                         int deviceid,
                         PointerBarrier barrier,
                         BarrierEventID eventid);

     int XIBarrierReleasePointers( Display *display,
                          XIBarrierReleasePointerInfo *barriers,
                          int num_barriers);

     display
          Specifies the connection to the X server.

     deviceid
          Specifies the device allowed to move across the barrier.

     barrier
          Specifies the barrier.

     eventid
          Specifies the unique event ID for this barrier event sequence.

     barriers
          List of deviceid, barrier, eventid triplets.

     num_barriers
          Number of elements in barriers.

**DESCRIPTION**
          If a pointer is constrained by a pointer barrier, release the pointer
          from the barrier and allow it to cross the barrier with the next motion.
          If the pointer moves away from the barrier, this request does nothing.

          This request only releases the pointer for the given barrier event
          sequence. If the pointer moves away from or through the barrier, a

XI_BarrierLeave event is generated. Future pointer movement constrained
by this barrier will assign a new eventid and require a new
XIBarrierReleasePointer request.

If the pointer is not currently blocked by the barrier or the barrier
event sequence has ended, this request does nothing.

XIBarrierReleasePointer and XIBarrierReleasePointers can generate a
BadDevice or BadValue error.

## DIAGNOSTICS

BadDevice

An invalid device was specified. The device does not
exist or is not a master pointer device.

BadValue

A value for a PointerBarrier argument does not name a defined
pointer barrier.

## SEE ALSO

XFixesCreatePointerBarrier(3)