

Xdotool - Launching Terminal Apps & Keys

xdotool key " "	
Aacute	
alt	
ampersand	(&)
apostrophe	
asciicircum	(^)
asciitilde	(~)
asterisk	(*)
at	(@)
backslash	(\)
BackSpace	
bar	()
braceleft	{ }
braceright	{ }
bracketleft	[]
bracketright	[]
colon	
comma	
ctrl	
del	
dollar	
Down	
eac	
equal	
Escape	
exclam	(!)
grave	(`)
F1 ... F12	
greater	
Left	
less	
meta	
minus	
numersign	(#)
parenleft	
parenright	
percent	
period	
plus	
question	
quotedbl	
quoteleft	
quoteright	
Return	(keyup Return)
Right	
semicolon	
shift	
slash	(/)
space	
super	
Tab	
underscore	
Up	

The xdotool is a utility used from the terminal or in a script to manually perform keyboard input. The commands can also be used to manually perform actions on windows and perform mouse inputs.

The syntax for xdotool depends on the command being used. To send keystrokes to a window you use the 'key' command. The syntax is as follows: xdotool [options] [keys]. There are three options available for "options":

- 1. --window window_id - specified keys for the keystrokes are sent to the window_id application. To get *window_id* from *window-title*: xdotool search --name "window-title", or xdotool getactivewindow.
- 2. --clearmodifiers - all modifiers are cleared, such as CAPS LOCK, NUM LOCK, shift held down, a mouse button held down, etc.
- 3. --delay milliseconds - sets the delay between each keystroke being sent, the default is 12 ms.

The [keys], which are small and cap letters and the special keys in the box on the left, specify which keys are being sent to the specified window. The keys are based on the X Keysym strings. Each of these characters can be used separately and some together. A space is placed between individual keystrokes and simultaneous keystrokes are connected with a plus (+) sign. For example, to perform a single keystroke of a SPACE and then an 'x' would be: 'space x'. When simultaneous keys are pressed, such as CTRL and 'x', the keystroke would be: 'ctrl+x'.

```
shutdown.sh
#!/bin/bash
xdotool key "super+t" &
sleep 1
xdotool key type "sudo shutdown -h o" &
sleep 1
xdotool key "Return" &
sleep 1
xdotool key type "password" &
sleep 1
xdotool key "Return" &
fi
```

Xdotool can be used in scripts to carry out commands that require typing in a terminal. For example, the spell helper **ispell** launches in the terminal by simply typing in *ispell* and clicking *enter (Return)* on the keyboard. The shell script **ispell.sh**, shown in the box on the right, put in *~/bin*, made excutable and put in the menus or linked to the desktop will perform the same function with a single click. Here, *xdotool key "super+t"* launches the terminal (my keybinding to launch terminal), *xdotool key type "ispell"* types "ispell" into the terminal and *xdotool key "Return"* effectively strikes the *enter* or *return* key. The application *inxi* can be launched with a similar script with *xdotool key type "inxi -F"*. It would appear this formula for creating shell launchers will work for just about any app launched in the terminal,

```
ispell.sh
#!/bin/bash
xdotool key "super+t" &
sleep 1
xdotool key type "ispell" &
sleep 1
xdotool key "Return" &
fi
```

which includes most linux apps. For a slightly more complicated example, see the launch script to create a *shutdown* button in the box on the left.

A second recipe for putting terminal apps on a menu is to use a tiny shell script **tlaunch.sh** shown in the first box on the right. If tlaunch.sh is stored in *~/bin*, then to put *ispell* in menus, for example, you can also use:

```
exec=lxterminal -e ~/bin/tlaunch.sh "ispell"
```

directly in the menu's config file, or *.xml* file, or *.desktop* file.

Other xdotool tools:

- 1. xdotool getactivewindow [gets active window id]
- 2. xdotool search --name "dog house" [lists all open windows with "dog house" in their title]
- 3. xdotool windowactive 37748739 [puts focus and brings to the foreground this window]
- 4. xwininfo -id 37748739 [gets information, in particular, geometry, for window with id 37748739]

Some Mouse & Window Controls	
xdotool type " "	
xdotool key type " "	(alternate type)
xdotool click "1"	(left click)
xdotool click "2"	(middle click)
xdotool click "3"	(right click)
xdotool mousedown "1"	(left button held down)
xdotool mousedown "2"	(middle button held down)
xdotool mousedown "3"	(right button held down)
xdotool mouseup "1-3"	(left-middle-right button held up)

```
tlaunch.sh
#!/bin/bash
$@
/bin/bash
```