
Shotgun

Release v0.2.2 - c3086e2

Apr 29, 2017

Contents

1	Overview	3
2	All Keymaps	5
3	Indices and tables	183

Welcome, this is the documentation for Shotgun version v0.2.2 - c3086e2. Shotgun is a custom keymap for [Blender](#) designed to standardize the keymap and make the most commonly used operators accessible.

Here is a birds eye view of the basics of Shotgun. Because Shotgun is designed to augment Blender's existing keymap most of the hotkeys will be the same as what you're already used to. For a more detailed reference of every hotkey see the list of all keymaps.

Navigation

All navigation in Shotgun is done with the mouse. Orbiting is done with the middle mouse, panning is the action mouse (right mouse if using a standard installation) and zooming the scroll wheel.

Interaction

There are a few basic hotkeys that are standardized across areas:

- *Ctrl-A*: is select all/toggle select
- *Double Click*: select group (what determines a group depends on the specific area)
- *Alt-Left Mouse*: border select (shift to extend)
- *Alt-Right Mouse*: lasso select (shift to deselect)
- *Alt-Right Click*: move cursor (in the dope sheet and graph editor it changes the current frame)
- *Ctrl-Alt-Click*: Add new point (e.g. extrude vertex)

Note: When UV editing border select is *Ctrl-Alt-Left Mouse*

CHAPTER 2

All Keymaps

This is the complete reference of every hotkey. For a more birds eye view of how it works see the overview.

Note: If a keymap area is not listed below it means that area uses the default Blender keymap.

3D View

Quick Reference

Hotkey	Operator
<i>Ctrl-ACTIONMOUSE</i>	<code>bpy.ops.view3d.cursor3d()</code>
<i>RIGHTMOUSE</i>	<code>bpy.ops.view3d.move()</code>
<i>Alt-EVT_TWEAK_A</i>	<code>bpy.ops.view3d.select_lasso()</code>
<i>Shift-Alt-EVT_TWEAK_A</i>	<code>bpy.ops.view3d.select_lasso()</code>
<i>A</i>	<code>bpy.ops.transform.skin_resize()</code>
<i>SELECTMOUSE</i>	<code>bpy.ops.object.select_grouped()</code>
<i>Alt-EVT_TWEAK_S</i>	<code>bpy.ops.view3d.select_border()</code>
<i>Shift-Alt-EVT_TWEAK_S</i>	<code>bpy.ops.view3d.select_border()</code>
<i>Any-LEFTMOUSE</i>	<code>bpy.ops.view3d.manipulator()</code>
<i>ACTIONMOUSE</i>	<code>bpy.ops.view3d.cursor3d()</code>
<i>MIDDLEMOUSE</i>	<code>bpy.ops.view3d.rotate()</code>
<i>Shift-MIDDLEMOUSE</i>	<code>bpy.ops.view3d.move()</code>
<i>Ctrl-MIDDLEMOUSE</i>	<code>bpy.ops.view3d.zoom()</code>
<i>Ctrl-Shift-MIDDLEMOUSE</i>	<code>bpy.ops.view3d.dolly()</code>
<i>Ctrl-NUMPAD_PERIOD</i>	<code>bpy.ops.view3d.view_selected()</code>
<i>NUMPAD_PERIOD</i>	<code>bpy.ops.view3d.view_selected()</code>
<i>Shift-NUMPAD_PERIOD</i>	<code>bpy.ops.view3d.view_lock_to_active()</code>
Continued on next page	

Table 2.1 – continued from previous page

Hotkey	Operator
<i>Alt-NUMPAD_PERIOD</i>	<code>bpy.ops.view3d.view_lock_clear()</code>
<i>Shift-F</i>	<code>bpy.ops.view3d.navigate()</code>
<i>Any-TIMER1</i>	<code>bpy.ops.view3d.smoothview()</code>
<i>TRACKPADPAN</i>	<code>bpy.ops.view3d.rotate()</code>
<i>MOUSEROTATE</i>	<code>bpy.ops.view3d.rotate()</code>
<i>Shift-TRACKPADPAN</i>	<code>bpy.ops.view3d.move()</code>
<i>TRACKPADZOOM</i>	<code>bpy.ops.view3d.zoom()</code>
<i>Ctrl-TRACKPADPAN</i>	<code>bpy.ops.view3d.zoom()</code>
<i>NUMPAD_PLUS</i>	<code>bpy.ops.view3d.zoom()</code>
<i>NUMPAD_MINUS</i>	<code>bpy.ops.view3d.zoom()</code>
<i>Ctrl-EQUAL</i>	<code>bpy.ops.view3d.zoom()</code>
<i>Ctrl-MINUS</i>	<code>bpy.ops.view3d.zoom()</code>
<i>WHEELINMOUSE</i>	<code>bpy.ops.view3d.zoom()</code>
<i>WHEELOUTMOUSE</i>	<code>bpy.ops.view3d.zoom()</code>
<i>Shift-NUMPAD_PLUS</i>	<code>bpy.ops.view3d.dolly()</code>
<i>Shift-NUMPAD_MINUS</i>	<code>bpy.ops.view3d.dolly()</code>
<i>Ctrl-Shift-EQUAL</i>	<code>bpy.ops.view3d.dolly()</code>
<i>Ctrl-Shift-MINUS</i>	<code>bpy.ops.view3d.dolly()</code>
<i>Shift-NUMPAD_ENTER</i>	<code>bpy.ops.view3d.zoom_camera_1_to_1()</code>
<i>HOME</i>	<code>bpy.ops.view3d.view_center_camera()</code>
<i>HOME</i>	<code>bpy.ops.view3d.view_center_lock()</code>
<i>Alt-HOME</i>	<code>bpy.ops.view3d.view_center_cursor()</code>
<i>Alt-F</i>	<code>bpy.ops.view3d.view_center_pick()</code>
<i>HOME</i>	<code>bpy.ops.view3d.view_all()</code>
<i>Ctrl-HOME</i>	<code>bpy.ops.view3d.view_all()</code>
<i>Shift-C</i>	<code>bpy.ops.view3d.view_all()</code>
<i>NUMPAD_0</i>	<code>bpy.ops.view3d.viewnumpad()</code>
<i>NUMPAD_1</i>	<code>bpy.ops.view3d.viewnumpad()</code>
<i>NUMPAD_2</i>	<code>bpy.ops.view3d.view_orbit()</code>
<i>NUMPAD_3</i>	<code>bpy.ops.view3d.viewnumpad()</code>
<i>NUMPAD_4</i>	<code>bpy.ops.view3d.view_orbit()</code>
<i>NUMPAD_5</i>	<code>bpy.ops.view3d.view_persportho()</code>
<i>NUMPAD_6</i>	<code>bpy.ops.view3d.view_orbit()</code>
<i>NUMPAD_7</i>	<code>bpy.ops.view3d.viewnumpad()</code>
<i>NUMPAD_8</i>	<code>bpy.ops.view3d.view_orbit()</code>
<i>Ctrl-NUMPAD_1</i>	<code>bpy.ops.view3d.viewnumpad()</code>
<i>Ctrl-NUMPAD_3</i>	<code>bpy.ops.view3d.viewnumpad()</code>
<i>Ctrl-NUMPAD_7</i>	<code>bpy.ops.view3d.viewnumpad()</code>
<i>Ctrl-NUMPAD_2</i>	<code>bpy.ops.view3d.view_pan()</code>
<i>Ctrl-NUMPAD_4</i>	<code>bpy.ops.view3d.view_pan()</code>
<i>Ctrl-NUMPAD_6</i>	<code>bpy.ops.view3d.view_pan()</code>
<i>Ctrl-NUMPAD_8</i>	<code>bpy.ops.view3d.view_pan()</code>
<i>Shift-NUMPAD_4</i>	<code>bpy.ops.view3d.view_roll()</code>
<i>Shift-NUMPAD_6</i>	<code>bpy.ops.view3d.view_roll()</code>
<i>NUMPAD_9</i>	<code>bpy.ops.view3d.view_orbit()</code>
<i>Ctrl-WHEELUPMOUSE</i>	<code>bpy.ops.view3d.view_pan()</code>
<i>Ctrl-WHEELDOWNMOUSE</i>	<code>bpy.ops.view3d.view_pan()</code>
<i>Shift-WHEELUPMOUSE</i>	<code>bpy.ops.view3d.view_pan()</code>
<i>Shift-WHEELDOWNMOUSE</i>	<code>bpy.ops.view3d.view_pan()</code>

Continued on next page

Table 2.1 – continued from previous page

Hotkey	Operator
<i>Ctrl-Alt-WHEELUPMOUSE</i>	<code>bpy.ops.view3d.view_orbit()</code>
<i>Ctrl-Alt-WHEELDOWNMOUSE</i>	<code>bpy.ops.view3d.view_orbit()</code>
<i>Shift-Alt-WHEELUPMOUSE</i>	<code>bpy.ops.view3d.view_orbit()</code>
<i>Shift-Alt-WHEELDOWNMOUSE</i>	<code>bpy.ops.view3d.view_orbit()</code>
<i>Ctrl-Shift-WHEELUPMOUSE</i>	<code>bpy.ops.view3d.view_roll()</code>
<i>Ctrl-Shift-WHEELDOWNMOUSE</i>	<code>bpy.ops.view3d.view_roll()</code>
<i>Shift-NUMPAD_1</i>	<code>bpy.ops.view3d.viewnumpad()</code>
<i>Shift-NUMPAD_3</i>	<code>bpy.ops.view3d.viewnumpad()</code>
<i>Shift-NUMPAD_7</i>	<code>bpy.ops.view3d.viewnumpad()</code>
<i>Ctrl-Shift-NUMPAD_1</i>	<code>bpy.ops.view3d.viewnumpad()</code>
<i>Ctrl-Shift-NUMPAD_3</i>	<code>bpy.ops.view3d.viewnumpad()</code>
<i>Ctrl-Shift-NUMPAD_7</i>	<code>bpy.ops.view3d.viewnumpad()</code>
<i>NUMPAD_SLASH</i>	<code>bpy.ops.view3d.localview()</code>
<i>NDOF_MOTION</i>	<code>bpy.ops.view3d.ndof_orbit_zoom()</code>
<i>Ctrl-NDOF_MOTION</i>	<code>bpy.ops.view3d.ndof_orbit()</code>
<i>Shift-NDOF_MOTION</i>	<code>bpy.ops.view3d.ndof_pan()</code>
<i>Ctrl-Shift-NDOF_MOTION</i>	<code>bpy.ops.view3d.ndof_all()</code>
<i>NDOF_BUTTON_FIT</i>	<code>bpy.ops.view3d.view_selected()</code>
<i>NDOF_BUTTON_ROLL_CCW</i>	<code>bpy.ops.view3d.view_roll()</code>
<i>NDOF_BUTTON_ROLL_CCW</i>	<code>bpy.ops.view3d.view_roll()</code>
<i>NDOF_BUTTON_FRONT</i>	<code>bpy.ops.view3d.viewnumpad()</code>
<i>NDOF_BUTTON_BACK</i>	<code>bpy.ops.view3d.viewnumpad()</code>
<i>NDOF_BUTTON_LEFT</i>	<code>bpy.ops.view3d.viewnumpad()</code>
<i>NDOF_BUTTON_RIGHT</i>	<code>bpy.ops.view3d.viewnumpad()</code>
<i>NDOF_BUTTON_TOP</i>	<code>bpy.ops.view3d.viewnumpad()</code>
<i>NDOF_BUTTON_BOTTOM</i>	<code>bpy.ops.view3d.viewnumpad()</code>
<i>Shift-NDOF_BUTTON_FRONT</i>	<code>bpy.ops.view3d.viewnumpad()</code>
<i>Shift-NDOF_BUTTON_RIGHT</i>	<code>bpy.ops.view3d.viewnumpad()</code>
<i>Shift-NDOF_BUTTON_TOP</i>	<code>bpy.ops.view3d.viewnumpad()</code>
<i>ACCENT_GRAVE</i>	<code>bpy.ops.view3d.layers()</code>
<i>Any-1</i>	<code>bpy.ops.view3d.layers()</code>
<i>Any-2</i>	<code>bpy.ops.view3d.layers()</code>
<i>Any-3</i>	<code>bpy.ops.view3d.layers()</code>
<i>Any-4</i>	<code>bpy.ops.view3d.layers()</code>
<i>Any-5</i>	<code>bpy.ops.view3d.layers()</code>
<i>Any-6</i>	<code>bpy.ops.view3d.layers()</code>
<i>Any-7</i>	<code>bpy.ops.view3d.layers()</code>
<i>Any-8</i>	<code>bpy.ops.view3d.layers()</code>
<i>Any-9</i>	<code>bpy.ops.view3d.layers()</code>
<i>Any-0</i>	<code>bpy.ops.view3d.layers()</code>
<i>Z</i>	<code>bpy.ops.wm.context_toggle_enum()</code>
<i>Alt-Z</i>	<code>bpy.ops.wm.context_toggle_enum()</code>
<i>Shift-Z</i>	<code>bpy.ops.view3d.toggle_render()</code>
<i>SELECTMOUSE</i>	<code>bpy.ops.view3d.select()</code>
<i>Shift-SELECTMOUSE</i>	<code>bpy.ops.view3d.select()</code>
<i>Ctrl-SELECTMOUSE</i>	<code>bpy.ops.view3d.select()</code>
<i>Alt-SELECTMOUSE</i>	<code>bpy.ops.view3d.select()</code>
<i>Ctrl-Shift-SELECTMOUSE</i>	<code>bpy.ops.view3d.select()</code>
<i>Ctrl-Alt-SELECTMOUSE</i>	<code>bpy.ops.view3d.select()</code>

Continued on next page

Table 2.1 – continued from previous page

Hotkey	Operator
<i>Shift-Alt-SELECTMOUSE</i>	<code>bpy.ops.view3d.select()</code>
<i>Ctrl-Shift-Alt-SELECTMOUSE</i>	<code>bpy.ops.view3d.select()</code>
<i>B</i>	<code>bpy.ops.view3d.select_border()</code>
<i>C</i>	<code>bpy.ops.view3d.select_circle()</code>
<i>Alt-B</i>	<code>bpy.ops.view3d.clip_border()</code>
<i>Shift-B</i>	<code>bpy.ops.view3d.zoom_border()</code>
<i>Shift-B</i>	<code>bpy.ops.view3d.render_border()</code>
<i>Ctrl-B</i>	<code>bpy.ops.view3d.render_border()</code>
<i>Ctrl-Alt-B</i>	<code>bpy.ops.view3d.clear_render_border()</code>
<i>Ctrl-Alt-NUMPAD_0</i>	<code>bpy.ops.view3d.camera_to_view()</code>
<i>Ctrl-NUMPAD_0</i>	<code>bpy.ops.view3d.object_as_camera()</code>
<i>Shift-S</i>	<code>bpy.ops.wm.call_menu()</code>
<i>Ctrl-C</i>	<code>bpy.ops.view3d.copybuffer()</code>
<i>Ctrl-V</i>	<code>bpy.ops.view3d.pastebuffer()</code>
<i>,</i>	<code>bpy.ops.wm.context_set_enum()</code>
<i>Ctrl-,</i>	<code>bpy.ops.wm.context_set_enum()</code>
<i>Alt-,</i>	<code>bpy.ops.wm.context_toggle()</code>
<i>Ctrl-SPACE</i>	<code>bpy.ops.wm.context_toggle()</code>
<i>.</i>	<code>bpy.ops.wm.context_set_enum()</code>
<i>Ctrl-.</i>	<code>bpy.ops.wm.context_set_enum()</code>
<i>Alt-.</i>	<code>bpy.ops.wm.context_set_enum()</code>
<i>G</i>	<code>bpy.ops.transform.translate()</code>
<i>EVT_TWEAK_S</i>	<code>bpy.ops.transform.translate()</code>
<i>R</i>	<code>bpy.ops.transform.rotate()</code>
<i>S</i>	<code>bpy.ops.transform.resize()</code>
<i>Shift-W</i>	<code>bpy.ops.transform.bend()</code>
<i>Shift-Alt-S</i>	<code>bpy.ops.transform.tosphere()</code>
<i>Ctrl-Shift-Alt-S</i>	<code>bpy.ops.transform.shear()</code>
<i>Alt-SPACE</i>	<code>bpy.ops.transform.select_orientation()</code>
<i>Ctrl-Alt-SPACE</i>	<code>bpy.ops.transform.create_orientation()</code>
<i>Ctrl-M</i>	<code>bpy.ops.transform.mirror()</code>
<i>Shift-Tab</i>	<code>bpy.ops.wm.context_toggle()</code>
<i>Ctrl-Shift-Tab</i>	<code>bpy.ops.wm.context_menu_enum()</code>
<i>Shift-T</i>	<code>bpy.ops.transform.translate()</code>
<i>Shift-Alt-T</i>	<code>bpy.ops.transform.resize()</code>
<i>Ctrl-A</i>	<code>bpy.ops.transform.skin_resize()</code>

Detailed Reference

Note: Hotkeys marked with the “(default)” prefix are inherited from the default blender keymap

Ctrl-ACTIONMOUSE → view3d.cursor3d : **MOUSE** → PRESS

Set 3D Cursor

`bpy.ops.view3d.cursor3d()`

RIGHTMOUSE → view3d.move : **MOUSE** → PRESS

Move View

`bpy.ops.view3d.move()`

Alt-EVT_TWEAK_A → view3d.select_lasso : **TWEAK** → ANY

Lasso Select

bpy.ops.view3d.select_lasso(path=[], deselect=False, extend=True)

Properties:	Values:
Deselect	False

Shift-Alt-EVT_TWEAK_A → view3d.select_lasso : **TWEAK** → ANY

Lasso Select

bpy.ops.view3d.select_lasso(path=[], deselect=False, extend=True)

Properties:	Values:
Deselect	True

A → transform.skin_resize : **KEYBOARD** → PRESS

Skin Resize

bpy.ops.transform.skin_resize(value=(1, 1, 1), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), release_confirm=False)

SELECTMOUSE → object.select_grouped : **MOUSE** → DOUBLE_CLICK

Select Grouped

bpy.ops.object.select_grouped(extend=False, type='CHILDREN_RECURSIVE')

Properties:	Values:
Type	GROUP

Alt-EVT_TWEAK_S → view3d.select_border : **TWEAK** → ANY

Border Select

bpy.ops.view3d.select_border(gesture_mode=0, xmin=0, xmax=0, ymin=0, ymax=0, extend=True)

Properties:	Values:
Extend	False

Shift-Alt-EVT_TWEAK_S → view3d.select_border : **TWEAK** → ANY

Border Select

bpy.ops.view3d.select_border(gesture_mode=0, xmin=0, xmax=0, ymin=0, ymax=0, extend=True)

Properties:	Values:
Extend	True

(default) Any-LEFTMOUSE → view3d.manipulator : **MOUSE** → PRESS

3D Manipulator

bpy.ops.view3d.manipulator(constraint_axis=(False, False, False), constraint_orientation='GLOBAL', release_confirm=False)

Properties:	Values:
Confirm on Release	True

(default) ACTIONMOUSE → view3d.cursor3d : **MOUSE** → PRESS

Set 3D Cursor

bpy.ops.view3d.cursor3d()

(default) MIDDLEMOUSE → view3d.rotate : **MOUSE** → PRESS

Rotate View

bpy.ops.view3d.rotate()

(default) Shift-MIDDLEMOUSE → view3d.move : **MOUSE** → PRESS

Move View

bpy.ops.view3d.move()

(default) Ctrl-MIDDLEMOUSE → view3d.zoom : **MOUSE** → PRESS

Zoom View

bpy.ops.view3d.zoom(delta=0, mx=0, my=0)

(default) Ctrl-Shift-MIDDLEMOUSE → view3d.dolly : **MOUSE** → PRESS

Dolly View

bpy.ops.view3d.dolly(delta=0, mx=0, my=0)

(default) Ctrl-NUMPAD_PERIOD → view3d.view_selected : **KEYBOARD** → PRESS

View Selected

bpy.ops.view3d.view_selected(use_all_regions=False)

Properties:	Values:
All Regions	True

(default) NUMPAD_PERIOD → view3d.view_selected : **KEYBOARD** → PRESS

View Selected

bpy.ops.view3d.view_selected(use_all_regions=False)

Properties:	Values:
All Regions	False

(default) Shift-NUMPAD_PERIOD → view3d.view_lock_to_active : **KEYBOARD** → PRESS

View Lock to Active

bpy.ops.view3d.view_lock_to_active()

(default) Alt-NUMPAD_PERIOD → view3d.view_lock_clear : **KEYBOARD** → PRESS

View Lock Clear

bpy.ops.view3d.view_lock_clear()

(default) Shift-F → view3d.navigate : **KEYBOARD** → PRESS

View Navigation

bpy.ops.view3d.navigate()

(default) Any-TIMER1 → view3d.smoothview : **TIMER** → ANY

Smooth View

bpy.ops.view3d.smoothview()

(default) TRACKPADPAN → view3d.rotate : **MOUSE** → ANY

Rotate View

bpy.ops.view3d.rotate()

(default) MOUSEROTATE → view3d.rotate : **MOUSE** → ANY

Rotate View

bpy.ops.view3d.rotate()

(default) Shift-TRACKPADPAN → view3d.move : **MOUSE** → ANY

Move View

bpy.ops.view3d.move()

(default) TRACKPADZOOM → view3d.zoom : **MOUSE** → ANY

Zoom View

bpy.ops.view3d.zoom(delta=0, mx=0, my=0)

(default) Ctrl-TRACKPADPAN → view3d.zoom : **MOUSE** → ANY

Zoom View

bpy.ops.view3d.zoom(delta=0, mx=0, my=0)

(default) NUMPAD_PLUS → view3d.zoom : **KEYBOARD** → PRESS

Zoom View

bpy.ops.view3d.zoom(delta=0, mx=0, my=0)

Properties:	Values:
Delta	1

(default) NUMPAD_MINUS → view3d.zoom : **KEYBOARD** → PRESS

Zoom View

bpy.ops.view3d.zoom(delta=0, mx=0, my=0)

Properties:	Values:
Delta	-1

(default) Ctrl-EQUAL → view3d.zoom : **KEYBOARD** → PRESS

Zoom View

bpy.ops.view3d.zoom(delta=0, mx=0, my=0)

Properties:	Values:
Delta	1

(default) Ctrl-MINUS → view3d.zoom : **KEYBOARD** → PRESS

Zoom View

bpy.ops.view3d.zoom(delta=0, mx=0, my=0)

Properties:	Values:
Delta	-1

(default) WHEELINMOUSE → view3d.zoom : **MOUSE** → PRESS

Zoom View

bpy.ops.view3d.zoom(delta=0, mx=0, my=0)

Properties:	Values:
Delta	1

(default) WHEELOUTMOUSE → view3d.zoom : **MOUSE** → PRESS

Zoom View

bpy.ops.view3d.zoom(delta=0, mx=0, my=0)

Properties:	Values:
Delta	-1

(default) Shift-NUMPAD_PLUS → view3d.dolly : **KEYBOARD** → PRESS

Dolly View

bpy.ops.view3d.dolly(delta=0, mx=0, my=0)

Properties:	Values:
Delta	1

(default) Shift-NUMPAD_MINUS → view3d.dolly : **KEYBOARD** → PRESS

Dolly View

bpy.ops.view3d.dolly(delta=0, mx=0, my=0)

Properties:	Values:
Delta	-1

(default) Ctrl-Shift-EQUAL → view3d.dolly : **KEYBOARD** → PRESS

Dolly View

bpy.ops.view3d.dolly(delta=0, mx=0, my=0)

Properties:	Values:
Delta	1

(default) Ctrl-Shift-MINUS → view3d.dolly : **KEYBOARD** → PRESS

Dolly View

bpy.ops.view3d.dolly(delta=0, mx=0, my=0)

Properties:	Values:
Delta	-1

(default) Shift-NUMPAD_ENTER → view3d.zoom_camera_1_to_1 : **KEYBOARD** → PRESS

Zoom Camera 1:1

bpy.ops.view3d.zoom_camera_1_to_1()

(default) HOME → view3d.view_center_camera : **KEYBOARD** → PRESS

View Camera Center

bpy.ops.view3d.view_center_camera()

(default) HOME → view3d.view_center_lock : **KEYBOARD** → PRESS

View Lock Center

bpy.ops.view3d.view_center_lock()

(default) Alt-HOME → view3d.view_center_cursor : **KEYBOARD** → PRESS

Center View to Cursor

bpy.ops.view3d.view_center_cursor()

(default) Alt-F → view3d.view_center_pick : **KEYBOARD** → PRESS

Center View to Mouse

bpy.ops.view3d.view_center_pick()

(default) HOME → view3d.view_all : **KEYBOARD** → PRESS

View All

bpy.ops.view3d.view_all(use_all_regions=False, center=False)

Properties:	Values:
Center	False

(default) Ctrl-HOME → view3d.view_all : **KEYBOARD** → PRESS

View All

bpy.ops.view3d.view_all(use_all_regions=False, center=False)

Properties:	Values:
All Regions	True
Center	False

(default) Shift-C → view3d.view_all : **KEYBOARD** → PRESS

View All

bpy.ops.view3d.view_all(use_all_regions=False, center=False)

Properties:	Values:
Center	True

(default) NUMPAD_0 → view3d.viewnumpad : **KEYBOARD** → PRESS

View Numpad

bpy.ops.view3d.viewnumpad(type='LEFT', align_active=False)

Properties:	Values:
View	CAMERA

(default) NUMPAD_1 → view3d.viewnumpad : **KEYBOARD** → PRESS

View Numpad

bpy.ops.view3d.viewnumpad(type='LEFT', align_active=False)

Properties:	Values:
View	FRONT

(default) NUMPAD_2 → view3d.view_orbit : **KEYBOARD** → PRESS

View Orbit

bpy.ops.view3d.view_orbit(angle=0, type='ORBITLEFT')

Properties:	Values:
Orbit	ORBITDOWN

(default) NUMPAD_3 → view3d.viewnumpad : **KEYBOARD** → PRESS

View Numpad

bpy.ops.view3d.viewnumpad(type='LEFT', align_active=False)

Properties:	Values:
View	RIGHT

(default) NUMPAD_4 → view3d.view_orbit : **KEYBOARD** → PRESS

View Orbit

bpy.ops.view3d.view_orbit(angle=0, type='ORBITLEFT')

Properties:	Values:
Orbit	ORBITLEFT

(default) NUMPAD_5 → view3d.view_persportho : **KEYBOARD** → PRESS

View Persp/Ortho

bpy.ops.view3d.view_persportho()

(default) NUMPAD_6 → view3d.view_orbit : **KEYBOARD** → PRESS

View Orbit

bpy.ops.view3d.view_orbit(angle=0, type='ORBITLEFT')

Properties:	Values:
Orbit	ORBITRIGHT

(default) NUMPAD_7 → view3d.viewnumpad : **KEYBOARD** → PRESS

View Numpad

bpy.ops.view3d.viewnumpad(type='LEFT', align_active=False)

Properties:	Values:
View	TOP

(default) NUMPAD_8 → view3d.view_orbit : **KEYBOARD** → PRESS

View Orbit

bpy.ops.view3d.view_orbit(angle=0, type='ORBITLEFT')

Properties:	Values:
Orbit	ORBITUP

(default) Ctrl1-NUMPAD_1 → view3d.viewnumpad : **KEYBOARD** → PRESS

View Numpad

bpy.ops.view3d.viewnumpad(type='LEFT', align_active=False)

Properties:	Values:
View	BACK

(default) Ctrl1-NUMPAD_3 → view3d.viewnumpad : **KEYBOARD** → PRESS

View Numpad

bpy.ops.view3d.viewnumpad(type='LEFT', align_active=False)

Properties:	Values:
View	LEFT

(default) Ctrl1-NUMPAD_7 → view3d.viewnumpad : **KEYBOARD** → PRESS

View Numpad

bpy.ops.view3d.viewnumpad(type='LEFT', align_active=False)

Properties:	Values:
View	BOTTOM

(default) Ctrl1-NUMPAD_2 → view3d.view_pan : **KEYBOARD** → PRESS

View Pan

bpy.ops.view3d.view_pan(type='PANLEFT')

Properties:	Values:
Pan	PANDOWN

(default) Ctrl1-NUMPAD_4 → view3d.view_pan : **KEYBOARD** → PRESS

View Pan

bpy.ops.view3d.view_pan(type='PANLEFT')

Properties:	Values:
Pan	PANLEFT

(default) Ctrl1-NUMPAD_6 → view3d.view_pan : **KEYBOARD** → PRESS

View Pan

bpy.ops.view3d.view_pan(type='PANLEFT')

Properties:	Values:
Pan	PANRIGHT

(default) Ctrl1-NUMPAD_8 → view3d.view_pan : **KEYBOARD** → PRESS

View Pan

bpy.ops.view3d.view_pan(type='PANLEFT')

Properties:	Values:
Pan	PANUP

(default) Shift-NUMPAD_4 → view3d.view_roll : **KEYBOARD** → PRESS

View Roll

bpy.ops.view3d.view_roll(angle=0, type='ANGLE')

Properties:	Values:
Roll Angle Source	LEFT

(default) Shift-NUMPAD_6 → view3d.view_roll : **KEYBOARD** → PRESS

View Roll

bpy.ops.view3d.view_roll(angle=0, type='ANGLE')

Properties:	Values:
Roll Angle Source	RIGHT

(default) NUMPAD_9 → view3d.view_orbit : **KEYBOARD** → PRESS

View Orbit

bpy.ops.view3d.view_orbit(angle=0, type='ORBITLEFT')

Properties:	Values:
Orbit	ORBITRIGHT
Roll	3.1415927410125732

(default) Ctrl-WHEELUPMOUSE → view3d.view_pan : **MOUSE** → PRESS

View Pan

bpy.ops.view3d.view_pan(type='PANLEFT')

Properties:	Values:
Pan	PANRIGHT

(default) Ctrl-WHEELDOWNMOUSE → view3d.view_pan : **MOUSE** → PRESS

View Pan

bpy.ops.view3d.view_pan(type='PANLEFT')

Properties:	Values:
Pan	PANLEFT

(default) Shift-WHEELUPMOUSE → view3d.view_pan : **MOUSE** → PRESS

View Pan

bpy.ops.view3d.view_pan(type='PANLEFT')

Properties:	Values:
Pan	PANUP

(default) Shift-WHEELDOWNMOUSE → view3d.view_pan : **MOUSE** → PRESS

View Pan

bpy.ops.view3d.view_pan(type='PANLEFT')

Properties:	Values:
Pan	PANDOWN

(default) Ctrl-Alt-WHEELUPMOUSE → view3d.view_orbit : **MOUSE** → PRESS

View Orbit

bpy.ops.view3d.view_orbit(angle=0, type='ORBITLEFT')

Properties:	Values:
Orbit	ORBITLEFT

(default) Ctrl-Alt-WHEELDOWNMOUSE → view3d.view_orbit : **MOUSE** → PRESS

View Orbit

bpy.ops.view3d.view_orbit(angle=0, type='ORBITLEFT')

Properties:	Values:
Orbit	ORBITRIGHT

(default) Shift-Alt-WHEELUPMOUSE → view3d.view_orbit : **MOUSE** → PRESS

View Orbit

bpy.ops.view3d.view_orbit(angle=0, type='ORBITLEFT')

Properties:	Values:
Orbit	ORBITUP

(default) Shift-Alt-WHEELDOWNMOUSE → view3d.view_orbit : **MOUSE** → PRESS

View Orbit

bpy.ops.view3d.view_orbit(angle=0, type='ORBITLEFT')

Properties:	Values:
Orbit	ORBITDOWN

(default) Ctrl-Shift-WHEELUPMOUSE → view3d.view_roll : **MOUSE** → PRESS

View Roll

bpy.ops.view3d.view_roll(angle=0, type='ANGLE')

Properties:	Values:
Roll Angle Source	LEFT

(default) Ctrl-Shift-WHEELDOWNMOUSE → view3d.view_roll : **MOUSE** → PRESS

View Roll

bpy.ops.view3d.view_roll(angle=0, type='ANGLE')

Properties:	Values:
Roll Angle Source	RIGHT

(default) Shift-NUMPAD_1 → view3d.viewnumpad : **KEYBOARD** → PRESS

View Numpad

bpy.ops.view3d.viewnumpad(type='LEFT', align_active=False)

Properties:	Values:
View	FRONT
Align Active	True

(default) Shift-NUMPAD_3 → view3d.viewnumpad : **KEYBOARD** → PRESS

View Numpad

bpy.ops.view3d.viewnumpad(type='LEFT', align_active=False)

Properties:	Values:
View	RIGHT
Align Active	True

(default) Shift-NUMPAD_7 → view3d.viewnumpad : **KEYBOARD** → PRESS

View Numpad

bpy.ops.view3d.viewnumpad(type='LEFT', align_active=False)

Properties:	Values:
View	TOP
Align Active	True

(default) Ctrl-Shift-NUMPAD_1 → view3d.viewnumpad : **KEYBOARD** → PRESS
View Numpad

bpy.ops.view3d.viewnumpad(type='LEFT', align_active=False)

Properties:	Values:
View	BACK
Align Active	True

(default) Ctrl-Shift-NUMPAD_3 → view3d.viewnumpad : **KEYBOARD** → PRESS
View Numpad

bpy.ops.view3d.viewnumpad(type='LEFT', align_active=False)

Properties:	Values:
View	LEFT
Align Active	True

(default) Ctrl-Shift-NUMPAD_7 → view3d.viewnumpad : **KEYBOARD** → PRESS
View Numpad

bpy.ops.view3d.viewnumpad(type='LEFT', align_active=False)

Properties:	Values:
View	BOTTOM
Align Active	True

(default) NUMPAD_SLASH → view3d.localview : **KEYBOARD** → PRESS
Local View

bpy.ops.view3d.localview()

(default) NDOF_MOTION → view3d.ndof_orbit_zoom : **NDOF** → ANY
NDOF Orbit View with Zoom

bpy.ops.view3d.ndof_orbit_zoom()

(default) Ctrl-NDOF_MOTION → view3d.ndof_orbit : **NDOF** → ANY
NDOF Orbit View

bpy.ops.view3d.ndof_orbit()

(default) Shift-NDOF_MOTION → view3d.ndof_pan : **NDOF** → ANY
NDOF Pan View

bpy.ops.view3d.ndof_pan()

(default) Ctrl-Shift-NDOF_MOTION → view3d.ndof_all : **NDOF** → ANY
NDOF Move View

bpy.ops.view3d.ndof_all()

(default) NDOF_BUTTON_FIT → view3d.view_selected : **NDOF** → PRESS
View Selected

bpy.ops.view3d.view_selected(use_all_regions=False)

Properties:	Values:
All Regions	False

(default) NDOF_BUTTON_ROLL_CCW → view3d.view_roll : **NDOF** → PRESS

View Roll

bpy.ops.view3d.view_roll(angle=0, type='ANGLE')

Properties:	Values:
Roll Angle Source	LEFT

(default) NDOF_BUTTON_ROLL_CCW → view3d.view_roll : **NDOF** → PRESS

View Roll

bpy.ops.view3d.view_roll(angle=0, type='ANGLE')

Properties:	Values:
Roll Angle Source	RIGHT

(default) NDOF_BUTTON_FRONT → view3d.viewnumpad : **NDOF** → PRESS

View Numpad

bpy.ops.view3d.viewnumpad(type='LEFT', align_active=False)

Properties:	Values:
View	FRONT

(default) NDOF_BUTTON_BACK → view3d.viewnumpad : **NDOF** → PRESS

View Numpad

bpy.ops.view3d.viewnumpad(type='LEFT', align_active=False)

Properties:	Values:
View	BACK

(default) NDOF_BUTTON_LEFT → view3d.viewnumpad : **NDOF** → PRESS

View Numpad

bpy.ops.view3d.viewnumpad(type='LEFT', align_active=False)

Properties:	Values:
View	LEFT

(default) NDOF_BUTTON_RIGHT → view3d.viewnumpad : **NDOF** → PRESS

View Numpad

bpy.ops.view3d.viewnumpad(type='LEFT', align_active=False)

Properties:	Values:
View	RIGHT

(default) NDOF_BUTTON_TOP → view3d.viewnumpad : **NDOF** → PRESS

View Numpad

bpy.ops.view3d.viewnumpad(type='LEFT', align_active=False)

Properties:	Values:
View	TOP

(default) NDOF_BUTTON_BOTTOM → view3d.viewnumpad : **NDOF** → PRESS

View Numpad

bpy.ops.view3d.viewnumpad(type='LEFT', align_active=False)

Properties:	Values:
View	BOTTOM

(default) Shift-NDOF_BUTTON_FRONT → view3d.viewnumpad : **NDOF** → PRESS

View Numpad

bpy.ops.view3d.viewnumpad(type='LEFT', align_active=False)

Properties:	Values:
View	FRONT
Align Active	True

(default) Shift-NDOF_BUTTON_RIGHT → view3d.viewnumpad : **NDOF** → PRESS

View Numpad

bpy.ops.view3d.viewnumpad(type='LEFT', align_active=False)

Properties:	Values:
View	RIGHT
Align Active	True

(default) Shift-NDOF_BUTTON_TOP → view3d.viewnumpad : **NDOF** → PRESS

View Numpad

bpy.ops.view3d.viewnumpad(type='LEFT', align_active=False)

Properties:	Values:
View	TOP
Align Active	True

(default) ACCENT_GRAVE → view3d.layers : **KEYBOARD** → PRESS

Layers

bpy.ops.view3d.layers(nr=1, extend=False, toggle=True)

Properties:	Values:
Number	0

(default) Any-1 → view3d.layers : **KEYBOARD** → PRESS

Layers

bpy.ops.view3d.layers(nr=1, extend=False, toggle=True)

Properties:	Values:
Number	1

(default) Any-2 → view3d.layers : **KEYBOARD** → PRESS

Layers

bpy.ops.view3d.layers(nr=1, extend=False, toggle=True)

Properties:	Values:
Number	2

(default) Any-3 → view3d.layers : **KEYBOARD** → PRESS

Layers

bpy.ops.view3d.layers(nr=1, extend=False, toggle=True)

Properties:	Values:
Number	3

(default) Any-4 → view3d.layers : **KEYBOARD** → PRESS

Layers

bpy.ops.view3d.layers(nr=1, extend=False, toggle=True)

Properties:	Values:
Number	4

(default) Any-5 → view3d.layers : **KEYBOARD** → PRESS
Layers

bpy.ops.view3d.layers(nr=1, extend=False, toggle=True)

Properties:	Values:
Number	5

(default) Any-6 → view3d.layers : **KEYBOARD** → PRESS
Layers

bpy.ops.view3d.layers(nr=1, extend=False, toggle=True)

Properties:	Values:
Number	6

(default) Any-7 → view3d.layers : **KEYBOARD** → PRESS
Layers

bpy.ops.view3d.layers(nr=1, extend=False, toggle=True)

Properties:	Values:
Number	7

(default) Any-8 → view3d.layers : **KEYBOARD** → PRESS
Layers

bpy.ops.view3d.layers(nr=1, extend=False, toggle=True)

Properties:	Values:
Number	8

(default) Any-9 → view3d.layers : **KEYBOARD** → PRESS
Layers

bpy.ops.view3d.layers(nr=1, extend=False, toggle=True)

Properties:	Values:
Number	9

(default) Any-0 → view3d.layers : **KEYBOARD** → PRESS
Layers

bpy.ops.view3d.layers(nr=1, extend=False, toggle=True)

Properties:	Values:
Number	10

(default) Z → wm.context_toggle_enum : **KEYBOARD** → PRESS
Context Toggle Values

bpy.ops.wm.context_toggle_enum(data_path="", value_1="", value_2="")

Properties:	Values:
Context Attributes	space_data.viewport_shade
Value	SOLID
Value	WIREFRAME

(default) Alt-Z → wm.context_toggle_enum : **KEYBOARD** → PRESS
Context Toggle Values

bpy.ops.wm.context_toggle_enum(data_path="", value_1="", value_2="")

Properties:	Values:
Context Attributes	space_data.viewport_shade
Value	SOLID
Value	TEXTURED

(default) Shift-Z → view3d.toggle_render : **KEYBOARD** → PRESS

Toggle Rendered Shading

bpy.ops.view3d.toggle_render()

(default) SELECTMOUSE → view3d.select : **MOUSE** → PRESS

Activate/Select

bpy.ops.view3d.select(extend=False, deselect=False, toggle=False, center=False, enumerate=False, object=False, location=(0, 0))

Properties:	Values:
Extend	False
Deselect	False
Toggle Selection	False
Center	False
Object	False
Enumerate	False

(default) Shift-SELECTMOUSE → view3d.select : **MOUSE** → PRESS

Activate/Select

bpy.ops.view3d.select(extend=False, deselect=False, toggle=False, center=False, enumerate=False, object=False, location=(0, 0))

Properties:	Values:
Extend	False
Deselect	False
Toggle Selection	True
Center	False
Object	False
Enumerate	False

(default) Ctrl-SELECTMOUSE → view3d.select : **MOUSE** → PRESS

Activate/Select

bpy.ops.view3d.select(extend=False, deselect=False, toggle=False, center=False, enumerate=False, object=False, location=(0, 0))

Properties:	Values:
Extend	False
Deselect	False
Toggle Selection	False
Center	True
Object	True
Enumerate	False

(default) Alt-SELECTMOUSE → view3d.select : **MOUSE** → PRESS

Activate/Select

bpy.ops.view3d.select(extend=False, deselect=False, toggle=False, center=False, enumerate=False, object=False, location=(0, 0))

Properties:	Values:
Extend	False
Deselect	False
Toggle Selection	False
Center	False
Object	False
Enumerate	True

(default) Ctrl-Shift-SELECTMOUSE → view3d.select : **MOUSE** → PRESS

Activate/Select

bpy.ops.view3d.select(extend=False, deselect=False, toggle=False, center=False, enumerate=False, object=False, location=(0, 0))

Properties:	Values:
Extend	True
Deselect	False
Toggle Selection	True
Center	True
Object	False
Enumerate	False

(default) Ctrl-Alt-SELECTMOUSE → view3d.select : **MOUSE** → PRESS

Activate/Select

bpy.ops.view3d.select(extend=False, deselect=False, toggle=False, center=False, enumerate=False, object=False, location=(0, 0))

Properties:	Values:
Extend	False
Deselect	False
Toggle Selection	False
Center	True
Object	False
Enumerate	True

(default) Shift-Alt-SELECTMOUSE → view3d.select : **MOUSE** → PRESS

Activate/Select

bpy.ops.view3d.select(extend=False, deselect=False, toggle=False, center=False, enumerate=False, object=False, location=(0, 0))

Properties:	Values:
Extend	False
Deselect	False
Toggle Selection	True
Center	False
Object	False
Enumerate	True

(default) Ctrl-Shift-Alt-SELECTMOUSE → view3d.select : **MOUSE** → PRESS

Activate/Select

bpy.ops.view3d.select(extend=False, deselect=False, toggle=False, center=False, enumerate=False, object=False, location=(0, 0))

Properties:	Values:
Extend	False
Deselect	False
Toggle Selection	True
Center	True
Object	False
Enumerate	True

(default) B → view3d.select_border : **KEYBOARD** → PRESS

Border Select

bpy.ops.view3d.select_border(gesture_mode=0, xmin=0, xmax=0, ymin=0, ymax=0, extend=True)

(default) C → view3d.select_circle : **KEYBOARD** → PRESS

Circle Select

bpy.ops.view3d.select_circle(x=0, y=0, radius=1, gesture_mode=0)

(default) Alt-B → view3d.clip_border : **KEYBOARD** → PRESS

Clipping Border

bpy.ops.view3d.clip_border(xmin=0, xmax=0, ymin=0, ymax=0)

(default) Shift-B → view3d.zoom_border : **KEYBOARD** → PRESS

Zoom to Border

bpy.ops.view3d.zoom_border(gesture_mode=0, xmin=0, xmax=0, ymin=0, ymax=0)

(default) Shift-B → view3d.render_border : **KEYBOARD** → PRESS

Set Render Border

bpy.ops.view3d.render_border(xmin=0, xmax=0, ymin=0, ymax=0, camera_only=False)

Properties:	Values:
Camera Only	True

(default) Ctrl-B → view3d.render_border : **KEYBOARD** → PRESS

Set Render Border

bpy.ops.view3d.render_border(xmin=0, xmax=0, ymin=0, ymax=0, camera_only=False)

Properties:	Values:
Camera Only	False

(default) Ctrl-Alt-B → view3d.clear_render_border : **KEYBOARD** → PRESS

Clear Render Border

bpy.ops.view3d.clear_render_border()

(default) Ctrl-Alt-NUMPAD_0 → view3d.camera_to_view : **KEYBOARD** → PRESS

Align Camera To View

bpy.ops.view3d.camera_to_view()

(default) Ctrl-NUMPAD_0 → view3d.object_as_camera : **KEYBOARD** → PRESS

Set Active Object as Camera

bpy.ops.view3d.object_as_camera()

(default) Shift-S → wm.call_menu : **KEYBOARD** → PRESS

Call Menu

bpy.ops.wm.call_menu(name='')

Properties:	Values:
Name	VIEW3D_MT_snap

(default) Ctrl-C → view3d.copybuffer : **KEYBOARD** → PRESS
Copy Selection to Buffer

bpy.ops.view3d.copybuffer()

(default) Ctrl-V → view3d.pastebuffer : **KEYBOARD** → PRESS
Paste Selection from Buffer

bpy.ops.view3d.pastebuffer(autoselect=True, active_layer=True)

(default) , → wm.context_set_enum : **KEYBOARD** → PRESS
Context Set Enum

bpy.ops.wm.context_set_enum(data_path="", value="")

Properties:	Values:
Context Attributes	space_data.pivot_point
Value	BOUNDING_BOX_CENTER

(default) Ctrl-, → wm.context_set_enum : **KEYBOARD** → PRESS
Context Set Enum

bpy.ops.wm.context_set_enum(data_path="", value="")

Properties:	Values:
Context Attributes	space_data.pivot_point
Value	MEDIAN_POINT

(default) Alt-, → wm.context_toggle : **KEYBOARD** → PRESS
Context Toggle

bpy.ops.wm.context_toggle(data_path="")

Properties:	Values:
Context Attributes	space_data.use_pivot_point_align

(default) Ctrl-SPACE → wm.context_toggle : **KEYBOARD** → PRESS
Context Toggle

bpy.ops.wm.context_toggle(data_path="")

Properties:	Values:
Context Attributes	space_data.show_manipulator

(default) . → wm.context_set_enum : **KEYBOARD** → PRESS
Context Set Enum

bpy.ops.wm.context_set_enum(data_path="", value="")

Properties:	Values:
Context Attributes	space_data.pivot_point
Value	CURSOR

(default) Ctrl-. → wm.context_set_enum : **KEYBOARD** → PRESS
Context Set Enum

bpy.ops.wm.context_set_enum(data_path="", value="")

Properties:	Values:
Context Attributes	space_data.pivot_point
Value	INDIVIDUAL_ORIGINS

(default) Alt-. → wm.context_set_enum : **KEYBOARD** → PRESS

Context Set Enum

bpy.ops.wm.context_set_enum(data_path="", value="")

Properties:	Values:
Context Attributes	space_data.pivot_point
Value	ACTIVE_ELEMENT

(default) G → transform.translate : **KEYBOARD** → PRESS

Translate

bpy.ops.transform.translate(value=(0, 0, 0), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpencil_strokes=False, texture_space=False, remove_on_cancel=False, release_confirm=False)

(default) EVT_TWEAK_S → transform.translate : **TWEAK** → ANY

Translate

bpy.ops.transform.translate(value=(0, 0, 0), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpencil_strokes=False, texture_space=False, remove_on_cancel=False, release_confirm=False)

(default) R → transform.rotate : **KEYBOARD** → PRESS

Rotate

bpy.ops.transform.rotate(value=0, axis=(0, 0, 0), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpencil_strokes=False, release_confirm=False)

(default) S → transform.resize : **KEYBOARD** → PRESS

Resize

bpy.ops.transform.resize(value=(1, 1, 1), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpencil_strokes=False, texture_space=False, remove_on_cancel=False, release_confirm=False)

(default) Shift-W → transform.bend : **KEYBOARD** → PRESS

Bend

bpy.ops.transform.bend(value=(0,), mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpencil_strokes=False, release_confirm=False)

(default) Shift-Alt-S → transform.tosphere : **KEYBOARD** → PRESS

To Sphere

bpy.ops.transform.tosphere(value=0, mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpencil_strokes=False, release_confirm=False)

(default) Ctrl-Shift-Alt-S → transform.shear : **KEYBOARD** → PRESS

Shear

```
bpy.ops.transform.shear(value=0, mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpencil_strokes=False, release_confirm=False)
```

(default) Alt-Space → transform.select_orientation : **KEYBOARD** → PRESS

Select Orientation

```
bpy.ops.transform.select_orientation(orientation='GLOBAL')
```

(default) Ctrl-Alt-Space → transform.create_orientation : **KEYBOARD** → PRESS

Create Orientation

```
bpy.ops.transform.create_orientation(name='', use_view=False, use=False, overwrite=False)
```

Properties:	Values:
Use after creation	True

(default) Ctrl-M → transform.mirror : **KEYBOARD** → PRESS

Mirror

```
bpy.ops.transform.mirror(constraint_axis=(False, False, False), constraint_orientation='GLOBAL', proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, gpencil_strokes=False, release_confirm=False)
```

(default) Shift-Tab → wm.context_toggle : **KEYBOARD** → PRESS

Context Toggle

```
bpy.ops.wm.context_toggle(data_path='')
```

Properties:	Values:
Context Attributes	tool_settings.use_snap

(default) Ctrl-Shift-Tab → wm.context_menu_enum : **KEYBOARD** → PRESS

Context Enum Menu

```
bpy.ops.wm.context_menu_enum(data_path='')
```

Properties:	Values:
Context Attributes	tool_settings.snap_element

(default) Shift-T → transform.translate : **KEYBOARD** → PRESS

Translate

```
bpy.ops.transform.translate(value=(0, 0, 0), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpencil_strokes=False, texture_space=False, remove_on_cancel=False, release_confirm=False)
```

Properties:	Values:
Edit Texture Space	True

(default) Shift-Alt-T → transform.resize : **KEYBOARD** → PRESS

Resize

```
bpy.ops.transform.resize(value=(1, 1, 1), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpencil_strokes=False, texture_space=False, remove_on_cancel=False, release_confirm=False)
```

Properties:	Values:
Edit Texture Space	True

(default) Ctrl-A → transform.skin_resize : **KEYBOARD** → PRESS

Skin Resize

```
bpy.ops.transform.skin_resize(value=(1, 1, 1), constraint_axis=(False, False, False),
constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', propor-
tional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0,
0), snap_align=False, snap_normal=(0, 0, 0), release_confirm=False)
```

Animation

Quick Reference

Hotkey	Operator
<i>Ctrl-ACTIONMOUSE</i>	<code>bpy.ops.anim.change_frame()</code>
<i>ACTIONMOUSE</i>	<code>bpy.ops.anim.change_frame()</code>
<i>Ctrl-T</i>	<code>bpy.ops.wm.context_toggle()</code>
<i>P</i>	<code>bpy.ops.anim.previewrange_set()</code>
<i>Alt-P</i>	<code>bpy.ops.anim.previewrange_clear()</code>

Detailed Reference

Note: Hotkeys marked with the “(default)” prefix are inherited from the default blender keymap

Ctrl-ACTIONMOUSE → anim.change_frame : **MOUSE** → PRESS

Change Frame

```
bpy.ops.anim.change_frame(frame=0, snap=False)
```

(default) ACTIONMOUSE → anim.change_frame : **MOUSE** → PRESS

Change Frame

```
bpy.ops.anim.change_frame(frame=0, snap=False)
```

(default) Ctrl-T → wm.context_toggle : **KEYBOARD** → PRESS

Context Toggle

```
bpy.ops.wm.context_toggle(data_path='')
```

Properties:	Values:
Context Attributes	space_data.show_seconds

(default) P → anim.previewrange_set : **KEYBOARD** → PRESS

Set Preview Range

```
bpy.ops.anim.previewrange_set(xmin=0, xmax=0, ymin=0, ymax=0)
```

(default) Alt-P → anim.previewrange_clear : **KEYBOARD** → PRESS

Clear Preview Range

```
bpy.ops.anim.previewrange_clear()
```

Animation Channels

Quick Reference

Hotkey	Operator
<i>Ctrl-A</i>	<code>bpy.ops.anim.channels_select_all_toggle()</code>
<i>LEFTMOUSE</i>	<code>bpy.ops.anim.channels_click()</code>
<i>Shift-LEFTMOUSE</i>	<code>bpy.ops.anim.channels_click()</code>
<i>Ctrl-Shift-LEFTMOUSE</i>	<code>bpy.ops.anim.channels_click()</code>
<i>Ctrl-LEFTMOUSE</i>	<code>bpy.ops.anim.channels_rename()</code>
<i>LEFTMOUSE</i>	<code>bpy.ops.anim.channels_rename()</code>
<i>LEFTMOUSE</i>	<code>bpy.ops.anim.channel_select_keys()</code>
<i>Shift-LEFTMOUSE</i>	<code>bpy.ops.anim.channel_select_keys()</code>
<i>Ctrl-F</i>	<code>bpy.ops.anim.channels_find()</code>
<i>A</i>	<code>bpy.ops.anim.channels_select_all_toggle()</code>
<i>Ctrl-I</i>	<code>bpy.ops.anim.channels_select_all_toggle()</code>
<i>B</i>	<code>bpy.ops.anim.channels_select_border()</code>
<i>EVT_TWEAK_L</i>	<code>bpy.ops.anim.channels_select_border()</code>
<i>X</i>	<code>bpy.ops.anim.channels_delete()</code>
<i>DEL</i>	<code>bpy.ops.anim.channels_delete()</code>
<i>Shift-W</i>	<code>bpy.ops.anim.channels_setting_toggle()</code>
<i>Ctrl-Shift-W</i>	<code>bpy.ops.anim.channels_setting_enable()</code>
<i>Alt-W</i>	<code>bpy.ops.anim.channels_setting_disable()</code>
<i>Tab</i>	<code>bpy.ops.anim.channels_editable_toggle()</code>
<i>NUMPAD_PLUS</i>	<code>bpy.ops.anim.channels_expand()</code>
<i>NUMPAD_MINUS</i>	<code>bpy.ops.anim.channels_collapse()</code>
<i>Ctrl-NUMPAD_PLUS</i>	<code>bpy.ops.anim.channels_expand()</code>
<i>Ctrl-NUMPAD_MINUS</i>	<code>bpy.ops.anim.channels_collapse()</code>
<i>PAGE_UP</i>	<code>bpy.ops.anim.channels_move()</code>
<i>PAGE_DOWN</i>	<code>bpy.ops.anim.channels_move()</code>
<i>Shift-PAGE_UP</i>	<code>bpy.ops.anim.channels_move()</code>
<i>Shift-PAGE_DOWN</i>	<code>bpy.ops.anim.channels_move()</code>
<i>Ctrl-G</i>	<code>bpy.ops.anim.channels_group()</code>
<i>Alt-G</i>	<code>bpy.ops.anim.channels_ungroup()</code>

Detailed Reference

Note: Hotkeys marked with the “(default)” prefix are inherited from the default blender keymap

Ctrl-A → `anim.channels_select_all_toggle` : **KEYBOARD** → PRESS

Select All

`bpy.ops.anim.channels_select_all_toggle(invert=False)`

(default) LEFTMOUSE → `anim.channels_click` : **MOUSE** → PRESS

Mouse Click on Channels

`bpy.ops.anim.channels_click(extend=False, children_only=False)`

(default) Shift-LEFTMOUSE → `anim.channels_click` : **MOUSE** → PRESS

Mouse Click on Channels

bpy.ops.anim.channels_click(extend=False, children_only=False)

Properties:	Values:
Extend Select	True

(default) Ctrl-Shift-LEFTMOUSE → anim.channels_click : **MOUSE** → PRESS

Mouse Click on Channels

bpy.ops.anim.channels_click(extend=False, children_only=False)

Properties:	Values:
Select Children Only	True

(default) Ctrl-LEFTMOUSE → anim.channels_rename : **MOUSE** → PRESS

Rename Channels

bpy.ops.anim.channels_rename()

(default) LEFTMOUSE → anim.channels_rename : **MOUSE** → DOUBLE_CLICK

Rename Channels

bpy.ops.anim.channels_rename()

(default) LEFTMOUSE → anim.channel_select_keys : **MOUSE** → DOUBLE_CLICK

Select Channel keyframes

bpy.ops.anim.channel_select_keys(extend=False)

(default) Shift-LEFTMOUSE → anim.channel_select_keys : **MOUSE** → DOUBLE_CLICK

Select Channel keyframes

bpy.ops.anim.channel_select_keys(extend=False)

Properties:	Values:
Extend	True

(default) Ctrl-F → anim.channels_find : **KEYBOARD** → PRESS

Find Channels

bpy.ops.anim.channels_find(query="Query")

(default) A → anim.channels_select_all_toggle : **KEYBOARD** → PRESS

Select All

bpy.ops.anim.channels_select_all_toggle(invert=False)

(default) Ctrl-I → anim.channels_select_all_toggle : **KEYBOARD** → PRESS

Select All

bpy.ops.anim.channels_select_all_toggle(invert=False)

Properties:	Values:
Invert	True

(default) B → anim.channels_select_border : **KEYBOARD** → PRESS

Border Select

bpy.ops.anim.channels_select_border(gesture_mode=0, xmin=0, xmax=0, ymin=0, ymax=0, extend=True)

(default) EVT_TWEAK_L → anim.channels_select_border : **TWEAK** → ANY

Border Select

bpy.ops.anim.channels_select_border(gesture_mode=0, xmin=0, xmax=0, ymin=0, ymax=0, extend=True)

(default) **X** → anim.channels_delete : **KEYBOARD** → PRESS

Delete Channels

bpy.ops.anim.channels_delete()

(default) **DEL** → anim.channels_delete : **KEYBOARD** → PRESS

Delete Channels

bpy.ops.anim.channels_delete()

(default) **Shift-W** → anim.channels_setting_toggle : **KEYBOARD** → PRESS

Toggle Channel Setting

bpy.ops.anim.channels_setting_toggle(mode='TOGGLE', type='PROTECT')

(default) **Ctrl-Shift-W** → anim.channels_setting_enable : **KEYBOARD** → PRESS

Enable Channel Setting

bpy.ops.anim.channels_setting_enable(mode='ENABLE', type='PROTECT')

(default) **Alt-W** → anim.channels_setting_disable : **KEYBOARD** → PRESS

Disable Channel Setting

bpy.ops.anim.channels_setting_disable(mode='DISABLE', type='PROTECT')

(default) **Tab** → anim.channels_editable_toggle : **KEYBOARD** → PRESS

Toggle Channel Editability

bpy.ops.anim.channels_editable_toggle(mode='TOGGLE', type='PROTECT')

(default) **NUMPAD_PLUS** → anim.channels_expand : **KEYBOARD** → PRESS

Expand Channels

bpy.ops.anim.channels_expand(all=True)

(default) **NUMPAD_MINUS** → anim.channels_collapse : **KEYBOARD** → PRESS

Collapse Channels

bpy.ops.anim.channels_collapse(all=True)

(default) **Ctrl-NUMPAD_PLUS** → anim.channels_expand : **KEYBOARD** → PRESS

Expand Channels

bpy.ops.anim.channels_expand(all=True)

Properties:	Values:
All	False

(default) **Ctrl-NUMPAD_MINUS** → anim.channels_collapse : **KEYBOARD** → PRESS

Collapse Channels

bpy.ops.anim.channels_collapse(all=True)

Properties:	Values:
All	False

(default) **PAGE_UP** → anim.channels_move : **KEYBOARD** → PRESS

Move Channels

bpy.ops.anim.channels_move(direction='DOWN')

Properties:	Values:
Direction	UP

(default) PAGE_DOWN → anim.channels_move : **KEYBOARD** → PRESS

Move Channels

bpy.ops.anim.channels_move(direction='DOWN')

Properties:	Values:
Direction	DOWN

(default) Shift-PAGE_UP → anim.channels_move : **KEYBOARD** → PRESS

Move Channels

bpy.ops.anim.channels_move(direction='DOWN')

Properties:	Values:
Direction	TOP

(default) Shift-PAGE_DOWN → anim.channels_move : **KEYBOARD** → PRESS

Move Channels

bpy.ops.anim.channels_move(direction='DOWN')

Properties:	Values:
Direction	BOTTOM

(default) Ctrl-G → anim.channels_group : **KEYBOARD** → PRESS

Group Channels

bpy.ops.anim.channels_group(name="New Group")

(default) Alt-G → anim.channels_ungroup : **KEYBOARD** → PRESS

Ungroup Channels

bpy.ops.anim.channels_ungroup()

Armature

Quick Reference

Hotkey	Operator
<i>Ctrl-A</i>	<code>bpy.ops.armature.select_all()</code>
<i>Ctrl-Alt-SELECTMOUSE</i>	<code>bpy.ops.armature.click_extrude()</code>
<i>X</i>	<code>bpy.ops.sketch.delete()</code>
<i>DEL</i>	<code>bpy.ops.sketch.delete()</code>
<i>RIGHTMOUSE</i>	<code>bpy.ops.sketch.finish_stroke()</code>
<i>ESC</i>	<code>bpy.ops.sketch.cancel_stroke()</code>
<i>Shift-LEFTMOUSE</i>	<code>bpy.ops.sketch.gesture()</code>
<i>LEFTMOUSE</i>	<code>bpy.ops.sketch.draw_stroke()</code>
<i>Ctrl-LEFTMOUSE</i>	<code>bpy.ops.sketch.draw_stroke()</code>
<i>MOUSEMOVE</i>	<code>bpy.ops.sketch.draw_preview()</code>
<i>Ctrl-MOUSEMOVE</i>	<code>bpy.ops.sketch.draw_preview()</code>
<i>H</i>	<code>bpy.ops.armature.hide()</code>
<i>Shift-H</i>	<code>bpy.ops.armature.hide()</code>
<i>Alt-H</i>	<code>bpy.ops.armature.reveal()</code>
<i>Ctrl-Alt-A</i>	<code>bpy.ops.armature.align()</code>
<i>Ctrl-N</i>	<code>bpy.ops.armature.calculate_roll()</code>
Continued on next page	

Table 2.2 – continued from previous page

Hotkey	Operator
<i>Alt-R</i>	<code>bpy.ops.armature.roll_clear()</code>
<i>Alt-F</i>	<code>bpy.ops.armature.switch_direction()</code>
<i>Shift-A</i>	<code>bpy.ops.armature.bone_primitive_add()</code>
<i>Ctrl-P</i>	<code>bpy.ops.armature.parent_set()</code>
<i>Alt-P</i>	<code>bpy.ops.armature.parent_clear()</code>
<i>A</i>	<code>bpy.ops.armature.select_all()</code>
<i>Ctrl-I</i>	<code>bpy.ops.armature.select_all()</code>
<i>Ctrl-Shift-M</i>	<code>bpy.ops.armature.select_mirror()</code>
<i>LEFT_BRACKET</i>	<code>bpy.ops.armature.select_hierarchy()</code>
<i>Shift-LEFT_BRACKET</i>	<code>bpy.ops.armature.select_hierarchy()</code>
<i>RIGHT_BRACKET</i>	<code>bpy.ops.armature.select_hierarchy()</code>
<i>Shift-RIGHT_BRACKET</i>	<code>bpy.ops.armature.select_hierarchy()</code>
<i>Ctrl-NUMPAD_PLUS</i>	<code>bpy.ops.armature.select_more()</code>
<i>Ctrl-NUMPAD_MINUS</i>	<code>bpy.ops.armature.select_less()</code>
<i>Shift-G</i>	<code>bpy.ops.armature.select_similar()</code>
<i>L</i>	<code>bpy.ops.armature.select_linked()</code>
<i>Ctrl-SELECTMOUSE</i>	<code>bpy.ops.armature.shortest_path_pick()</code>
<i>X</i>	<code>bpy.ops.wm.call_menu()</code>
<i>DEL</i>	<code>bpy.ops.wm.call_menu()</code>
<i>Ctrl-X</i>	<code>bpy.ops.armature.dissolve()</code>
<i>Shift-D</i>	<code>bpy.ops.armature.duplicate_move()</code>
<i>E</i>	<code>bpy.ops.armature.extrude_move()</code>
<i>Shift-E</i>	<code>bpy.ops.armature.extrude_forked()</code>
<i>Ctrl-ACTIONMOUSE</i>	<code>bpy.ops.armature.click_extrude()</code>
<i>F</i>	<code>bpy.ops.armature.fill()</code>
<i>Alt-M</i>	<code>bpy.ops.armature.merge()</code>
<i>Y</i>	<code>bpy.ops.armature.split()</code>
<i>P</i>	<code>bpy.ops.armature.separate()</code>
<i>Shift-W</i>	<code>bpy.ops.wm.call_menu()</code>
<i>Ctrl-Shift-W</i>	<code>bpy.ops.wm.call_menu()</code>
<i>Alt-W</i>	<code>bpy.ops.wm.call_menu()</code>
<i>Ctrl-ACCENT_GRAVE</i>	<code>bpy.ops.armature.layers_show_all()</code>
<i>Shift-M</i>	<code>bpy.ops.armature.armature_layers()</code>
<i>M</i>	<code>bpy.ops.armature.bone_layers()</code>
<i>Ctrl-Alt-S</i>	<code>bpy.ops.transform.transform()</code>
<i>Alt-S</i>	<code>bpy.ops.transform.transform()</code>
<i>Ctrl-R</i>	<code>bpy.ops.transform.transform()</code>
<i>W</i>	<code>bpy.ops.wm.call_menu()</code>

Detailed Reference

Note: Hotkeys marked with the “(default)” prefix are inherited from the default blender keymap

Ctrl-A → `armature.select_all` : **KEYBOARD** → PRESS

(De)select All

`bpy.ops.armature.select_all(action='TOGGLE')`

Properties:	Values:
Action	TOGGLE

Ctrl-Alt-SELECTMOUSE → `armature.click_extrude` : **MOUSE** → PRESS

Click-Extrude

`bpy.ops.armature.click_extrude()`

(default) X → `sketch.delete` : **KEYBOARD** → PRESS

Delete

`bpy.ops.sketch.delete()`

(default) DEL → `sketch.delete` : **KEYBOARD** → PRESS

Delete

`bpy.ops.sketch.delete()`

(default) RIGHTMOUSE → `sketch.finish_stroke` : **MOUSE** → PRESS

End Stroke

`bpy.ops.sketch.finish_stroke()`

(default) ESC → `sketch.cancel_stroke` : **KEYBOARD** → PRESS

Cancel Stroke

`bpy.ops.sketch.cancel_stroke()`

(default) Shift-LEFTMOUSE → `sketch.gesture` : **MOUSE** → PRESS

Gesture

`bpy.ops.sketch.gesture(snap=False)`

(default) LEFTMOUSE → `sketch.draw_stroke` : **MOUSE** → PRESS

Draw Stroke

`bpy.ops.sketch.draw_stroke(snap=False)`

(default) Ctrl-LEFTMOUSE → `sketch.draw_stroke` : **MOUSE** → PRESS

Draw Stroke

`bpy.ops.sketch.draw_stroke(snap=False)`

Properties:	Values:
Snap	True

(default) MOUSEMOVE → `sketch.draw_preview` : **MOUSE** → ANY

Draw Preview

`bpy.ops.sketch.draw_preview(snap=False)`

(default) Ctrl-MOUSEMOVE → `sketch.draw_preview` : **MOUSE** → ANY

Draw Preview

`bpy.ops.sketch.draw_preview(snap=False)`

Properties:	Values:
Snap	True

(default) H → `armature.hide` : **KEYBOARD** → PRESS

Hide Selected Bones

`bpy.ops.armature.hide(unselected=False)`

Properties:	Values:
Unselected	False

(default) Shift-H → armature.hide : **KEYBOARD** → PRESS

Hide Selected Bones

bpy.ops.armature.hide(unselected=False)

Properties:	Values:
Unselected	True

(default) Alt-H → armature.reveal : **KEYBOARD** → PRESS

Reveal Bones

bpy.ops.armature.reveal()

(default) Ctrl-Alt-A → armature.align : **KEYBOARD** → PRESS

Align Bones

bpy.ops.armature.align()

(default) Ctrl-N → armature.calculate_roll : **KEYBOARD** → PRESS

Recalculate Roll

bpy.ops.armature.calculate_roll(type='POS_X', axis_flip=False, axis_only=False)

(default) Alt-R → armature.roll_clear : **KEYBOARD** → PRESS

Clear Roll

bpy.ops.armature.roll_clear(roll=0)

(default) Alt-F → armature.switch_direction : **KEYBOARD** → PRESS

Switch Direction

bpy.ops.armature.switch_direction()

(default) Shift-A → armature.bone_primitive_add : **KEYBOARD** → PRESS

Add Bone

bpy.ops.armature.bone_primitive_add(name="Bone")

(default) Ctrl-P → armature.parent_set : **KEYBOARD** → PRESS

Make Parent

bpy.ops.armature.parent_set(type='CONNECTED')

(default) Alt-P → armature.parent_clear : **KEYBOARD** → PRESS

Clear Parent

bpy.ops.armature.parent_clear(type='CLEAR')

(default) A → armature.select_all : **KEYBOARD** → PRESS

(De)select All

bpy.ops.armature.select_all(action='TOGGLE')

Properties:	Values:
Action	TOGGLE

(default) Ctrl-I → armature.select_all : **KEYBOARD** → PRESS

(De)select All

bpy.ops.armature.select_all(action='TOGGLE')

Properties:	Values:
Action	INVERT

(default) Ctrl-Shift-M → armature.select_mirror : **KEYBOARD** → PRESS

Flip Active/Selected Bone

bpy.ops.armature.select_mirror(only_active=False, extend=False)

Properties:	Values:
Extend	False

(default) LEFT_BRACKET → armature.select_hierarchy : **KEYBOARD** → PRESS

Select Hierarchy

bpy.ops.armature.select_hierarchy(direction='PARENT', extend=False)

Properties:	Values:
Direction	PARENT
Extend	False

(default) Shift-LEFT_BRACKET → armature.select_hierarchy : **KEYBOARD** → PRESS

Select Hierarchy

bpy.ops.armature.select_hierarchy(direction='PARENT', extend=False)

Properties:	Values:
Direction	PARENT
Extend	True

(default) RIGHT_BRACKET → armature.select_hierarchy : **KEYBOARD** → PRESS

Select Hierarchy

bpy.ops.armature.select_hierarchy(direction='PARENT', extend=False)

Properties:	Values:
Direction	CHILD
Extend	False

(default) Shift-RIGHT_BRACKET → armature.select_hierarchy : **KEYBOARD** → PRESS

Select Hierarchy

bpy.ops.armature.select_hierarchy(direction='PARENT', extend=False)

Properties:	Values:
Direction	CHILD
Extend	True

(default) Ctrl-NUMPAD_PLUS → armature.select_more : **KEYBOARD** → PRESS

Select More

bpy.ops.armature.select_more()

(default) Ctrl-NUMPAD_MINUS → armature.select_less : **KEYBOARD** → PRESS

Select Less

bpy.ops.armature.select_less()

(default) Shift-G → armature.select_similar : **KEYBOARD** → PRESS

Select Similar

bpy.ops.armature.select_similar(type='LENGTH', threshold=0.1)

(default) L → armature.select_linked : **KEYBOARD** → PRESS

Select Connected

bpy.ops.armature.select_linked(extend=False)

(default) Ctrl-SELECTMOUSE → armature.shortest_path_pick : **MOUSE** → PRESS

Pick Shortest Path

bpy.ops.armature.shortest_path_pick()

(default) X → wm.call_menu : **KEYBOARD** → PRESS

Call Menu

bpy.ops.wm.call_menu(name='')

Properties:	Values:
Name	VIEW3D_MT_edit_armature_delete

(default) DEL → wm.call_menu : **KEYBOARD** → PRESS

Call Menu

bpy.ops.wm.call_menu(name='')

Properties:	Values:
Name	VIEW3D_MT_edit_armature_delete

(default) Ctrl-X → armature.dissolve : **KEYBOARD** → PRESS

Dissolve Selected Bone(s)

bpy.ops.armature.dissolve()

(default) Shift-D → armature.duplicate_move : **KEYBOARD** → PRESS

Duplicate

bpy.ops.armature.duplicate_move(ARMATURE_OT_duplicate={}, TRANSFORM_OT_translate={"value":(0, 0, 0), "constraint_axis":(False, False, False), "constraint_orientation":'GLOBAL', "mirror":False, "proportional":'DISABLED', "proportional_edit_falloff":'SMOOTH', "proportional_size":1, "snap":False, "snap_target":'CLOSEST', "snap_point":(0, 0, 0), "snap_align":False, "snap_normal":(0, 0, 0), "gpencil_strokes":False, "texture_space":False, "remove_on_cancel":False, "release_confirm":False})

Properties:	Values:
Duplicate Selected Bone(s)	N/A
Translate	N/A

(default) E → armature.extrude_move : **KEYBOARD** → PRESS

Extrude

bpy.ops.armature.extrude_move(ARMATURE_OT_extrude={"forked":False}, TRANSFORM_OT_translate={"value":(0, 0, 0), "constraint_axis":(False, False, False), "constraint_orientation":'GLOBAL', "mirror":False, "proportional":'DISABLED', "proportional_edit_falloff":'SMOOTH', "proportional_size":1, "snap":False, "snap_target":'CLOSEST', "snap_point":(0, 0, 0), "snap_align":False, "snap_normal":(0, 0, 0), "gpencil_strokes":False, "texture_space":False, "remove_on_cancel":False, "release_confirm":False})

Properties:	Values:
Extrude	N/A
Translate	N/A

(default) Shift-E → armature.extrude_forked : **KEYBOARD** → PRESS

Extrude Forked

bpy.ops.armature.extrude_forked(ARMATURE_OT_extrude={"forked":False}, TRANSFORM_OT_translate={"value":(0, 0, 0), "constraint_axis":(False, False, False), "constraint_orientation":'GLOBAL', "mirror":False, "proportional":'DISABLED', "proportional_edit_falloff":'SMOOTH', "proportional_size":1, "snap":False, "snap_target":'CLOSEST', "snap_point":(0, 0, 0), "snap_align":False, "snap_normal":(0, 0, 0), "gpencil_strokes":False, "texture_space":False, "remove_on_cancel":False, "release_confirm":False})

Properties:	Values:
Extrude	N/A
Translate	N/A

(default) Ctrl-ACTIONMOUSE \rightarrow `armature.click_extrude` : **MOUSE** \rightarrow **CLICK**
Click-Extrude

```
bpy.ops.armature.click_extrude()
```

(default) $\mathbf{F} \rightarrow \text{armature.fill} : \mathbf{KEYBOARD} \rightarrow \text{PRESS}$
 Fill Between Joints

```
bpy.ops.armature.fill()
```

(default) Alt-M \rightarrow `armature.merge` : **KEYBOARD** \rightarrow PRESS
Merge Bones

```
bpy.ops.armature.merge(type='WITHIN_CHAIN')
```

(default) $\mathbf{Y} \rightarrow \text{armature.split} : \mathbf{KEYBOARD} \rightarrow \text{PRESS}$
Split

bpy.ops.armature.split()

(default) P → **armature.separate** : **KEYBOARD** → PRESS
Separate Bones

bpy.ops.armature.separate()

(default) Shift-W → wm.call_menu : **KEYBOARD** → PRESS
Call Menu

```
bpy.ops.wm.call_menu(name='')
```

Properties:	Values:
Name	VIEW3D_MT_bone_options_toggle

(default) Ctrl-Shift-W → wm.call_menu : **KEYBOARD** → PRESS
Call Menu

```
bpy.ops.wm.call_menu(name='')
```

Properties:	Values:
Name	VIEW3D_MT_bone_options_enable

(default) Alt-W → wm.call_menu : **KEYBOARD** → PRESS
Call Menu

```
bpy.ops.wm.call_menu(name='')
```

Properties:	Values:
Name	VIEW3D_MT_bone_options_disable

(default) Ctrl-ACCENT_GRAVE → `armature.layers_show_all` : **KEYBOARD** → PRESS
Show All Layers

```
bpy.ops.armature.layers_show_all(all=True)
```

(default) Shift-M → `armature.armature_layers` : **KEYBOARD** → PRESS
Change Armature Layers

[illegible]

(default) $\mathbf{M} \rightarrow \text{armature.bone_layers} : \mathbf{KEYBOARD} \rightarrow \text{PRESS}$

Change Bone Layers

[illegible]

(default) Ctrl-Alt-S \rightarrow transform.transform : **KEYBOARD** \rightarrow PRESS

Transform

```
bpy.ops.transform.transform(mode='TRANSLATION', value=(0, 0, 0, 0), axis=(0, 0, 0), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpencil_strokes=False, release_confirm=False)
```

Properties:	Values:
Mode	BONE_SIZE

(default) Alt-S \rightarrow transform.transform : **KEYBOARD** \rightarrow PRESS

Transform

```
bpy.ops.transform.transform(mode='TRANSLATION', value=(0, 0, 0, 0), axis=(0, 0, 0), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpencil_strokes=False, release_confirm=False)
```

Properties:	Values:
Mode	BONE_ENVELOPE

(default) Ctrl-R \rightarrow transform.transform : **KEYBOARD** \rightarrow PRESS

Transform

```
bpy.ops.transform.transform(mode='TRANSLATION', value=(0, 0, 0, 0), axis=(0, 0, 0), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpencil_strokes=False, release_confirm=False)
```

Properties:	Values:
Mode	BONE_ROLL

(default) W → wm.call menu : **KEYBOARD** → PRESS

Call Menu

```
bpy.ops.wm.call_menu(name='')
```

Properties:	Values:
Name	VIEW3D_MT_armature_specials

Clip Editor

Quick Reference

Hotkey	Operator
<i>ACTIONMOUSE</i>	<code>bpy.ops.clip.view_pan()</code>
Continued on next page	

Table 2.3 – continued from previous page

Hotkey	Operator
<i>Ctrl-A</i>	<code>bpy.ops.clip.select_all()</code>
<i>Alt-EVT_TWEAK_A</i>	<code>bpy.ops.clip.select_lasso()</code>
<i>Shift-Alt-EVT_TWEAK_A</i>	<code>bpy.ops.clip.select_lasso()</code>
<i>Alt-EVT_TWEAK_S</i>	<code>bpy.ops.clip.select_border()</code>
<i>Shift-Alt-EVT_TWEAK_S</i>	<code>bpy.ops.clip.select_border()</code>
<i>Ctrl-ACTIONMOUSE</i>	<code>bpy.ops.clip.cursor_set()</code>
<i>MIDDLEMOUSE</i>	<code>bpy.ops.clip.view_pan()</code>
<i>Shift-MIDDLEMOUSE</i>	<code>bpy.ops.clip.view_pan()</code>
<i>TRACKPADPAN</i>	<code>bpy.ops.clip.view_pan()</code>
<i>Ctrl-MIDDLEMOUSE</i>	<code>bpy.ops.clip.view_zoom()</code>
<i>TRACKPADZOOM</i>	<code>bpy.ops.clip.view_zoom()</code>
<i>Ctrl-TRACKPADPAN</i>	<code>bpy.ops.clip.view_zoom()</code>
<i>WHEELINMOUSE</i>	<code>bpy.ops.clip.view_zoom_in()</code>
<i>WHEELOUTMOUSE</i>	<code>bpy.ops.clip.view_zoom_out()</code>
<i>NUMPAD_PLUS</i>	<code>bpy.ops.clip.view_zoom_in()</code>
<i>NUMPAD_MINUS</i>	<code>bpy.ops.clip.view_zoom_out()</code>
<i>Ctrl-NUMPAD_8</i>	<code>bpy.ops.clip.view_zoom_ratio()</code>
<i>Ctrl-NUMPAD_4</i>	<code>bpy.ops.clip.view_zoom_ratio()</code>
<i>Ctrl-NUMPAD_2</i>	<code>bpy.ops.clip.view_zoom_ratio()</code>
<i>Shift-NUMPAD_8</i>	<code>bpy.ops.clip.view_zoom_ratio()</code>
<i>Shift-NUMPAD_4</i>	<code>bpy.ops.clip.view_zoom_ratio()</code>
<i>Shift-NUMPAD_2</i>	<code>bpy.ops.clip.view_zoom_ratio()</code>
<i>NUMPAD_1</i>	<code>bpy.ops.clip.view_zoom_ratio()</code>
<i>NUMPAD_2</i>	<code>bpy.ops.clip.view_zoom_ratio()</code>
<i>NUMPAD_4</i>	<code>bpy.ops.clip.view_zoom_ratio()</code>
<i>NUMPAD_8</i>	<code>bpy.ops.clip.view_zoom_ratio()</code>
<i>HOME</i>	<code>bpy.ops.clip.view_all()</code>
<i>F</i>	<code>bpy.ops.clip.view_all()</code>
<i>NUMPAD_PERIOD</i>	<code>bpy.ops.clip.view_selected()</code>
<i>NDOF_BUTTON_FIT</i>	<code>bpy.ops.clip.view_all()</code>
<i>NDOF_MOTION</i>	<code>bpy.ops.clip.view_ndof()</code>
<i>Ctrl-Shift-LEFT_ARROW</i>	<code>bpy.ops.clip.frame_jump()</code>
<i>Ctrl-Shift-RIGHT_ARROW</i>	<code>bpy.ops.clip.frame_jump()</code>
<i>Shift-Alt-LEFT_ARROW</i>	<code>bpy.ops.clip.frame_jump()</code>
<i>Shift-Alt-RIGHT_ARROW</i>	<code>bpy.ops.clip.frame_jump()</code>
<i>LEFTMOUSE</i>	<code>bpy.ops.clip.change_frame()</code>
<i>SELECTMOUSE</i>	<code>bpy.ops.clip.select()</code>
<i>Shift-SELECTMOUSE</i>	<code>bpy.ops.clip.select()</code>
<i>A</i>	<code>bpy.ops.clip.select_all()</code>
<i>Ctrl-I</i>	<code>bpy.ops.clip.select_all()</code>
<i>B</i>	<code>bpy.ops.clip.select_border()</code>
<i>C</i>	<code>bpy.ops.clip.select_circle()</code>
<i>Shift-G</i>	<code>bpy.ops.wm.call_menu()</code>
<i>Ctrl-Alt-EVT_TWEAK_A</i>	<code>bpy.ops.clip.select_lasso()</code>
<i>Ctrl-Shift-Alt-EVT_TWEAK_A</i>	<code>bpy.ops.clip.select_lasso()</code>
<i>Ctrl-LEFTMOUSE</i>	<code>bpy.ops.clip.add_marker_slide()</code>
<i>Shift-DEL</i>	<code>bpy.ops.clip.delete_marker()</code>
<i>Shift-X</i>	<code>bpy.ops.clip.delete_marker()</code>
<i>LEFTMOUSE</i>	<code>bpy.ops.clip.slide_marker()</code>

Continued on next page

Table 2.3 – continued from previous page

Hotkey	Operator
<i>Shift-D</i>	<code>bpy.ops.clip.disable_markers()</code>
<i>DEL</i>	<code>bpy.ops.clip.delete_track()</code>
<i>X</i>	<code>bpy.ops.clip.delete_track()</code>
<i>Ctrl-L</i>	<code>bpy.ops.clip.lock_tracks()</code>
<i>Alt-L</i>	<code>bpy.ops.clip.lock_tracks()</code>
<i>H</i>	<code>bpy.ops.clip.hide_tracks()</code>
<i>Shift-H</i>	<code>bpy.ops.clip.hide_tracks()</code>
<i>Alt-H</i>	<code>bpy.ops.clip.hide_tracks_clear()</code>
<i>ACTIONMOUSE</i>	<code>bpy.ops.clip.slide_plane_marker()</code>
<i>I</i>	<code>bpy.ops.clip.keyframe_insert()</code>
<i>Alt-I</i>	<code>bpy.ops.clip.keyframe_delete()</code>
<i>Ctrl-J</i>	<code>bpy.ops.clip.join_tracks()</code>
<i>W</i>	<code>bpy.ops.wm.call_menu()</code>
<i>L</i>	<code>bpy.ops.wm.context_toggle()</code>
<i>Alt-D</i>	<code>bpy.ops.wm.context_toggle()</code>
<i>Alt-S</i>	<code>bpy.ops.wm.context_toggle()</code>
<i>M</i>	<code>bpy.ops.wm.context_toggle()</code>
<i>G</i>	<code>bpy.ops.transform.translate()</code>
<i>EVT_TWEAK_S</i>	<code>bpy.ops.transform.translate()</code>
<i>S</i>	<code>bpy.ops.transform.resize()</code>
<i>R</i>	<code>bpy.ops.transform.rotate()</code>
<i>Alt-T</i>	<code>bpy.ops.clip.clear_track_path()</code>
<i>Shift-T</i>	<code>bpy.ops.clip.clear_track_path()</code>
<i>Shift-Alt-T</i>	<code>bpy.ops.clip.clear_track_path()</code>
<i>ACTIONMOUSE</i>	<code>bpy.ops.clip.cursor_set()</code>
<i>,</i>	<code>bpy.ops.wm.context_set_enum()</code>
<i>Ctrl-,</i>	<code>bpy.ops.wm.context_set_enum()</code>
<i>.</i>	<code>bpy.ops.wm.context_set_enum()</code>
<i>Ctrl-.</i>	<code>bpy.ops.wm.context_set_enum()</code>
<i>Ctrl-C</i>	<code>bpy.ops.clip.copy_tracks()</code>
<i>Ctrl-V</i>	<code>bpy.ops.clip.paste_tracks()</code>

Detailed Reference

Note: Hotkeys marked with the “(default)” prefix are inherited from the default blender keymap

ACTIONMOUSE → clip.view_pan : **MOUSE** → PRESS

View Pan

`bpy.ops.clip.view_pan(offset=(0, 0))`

Ctrl-A → clip.select_all : **KEYBOARD** → PRESS

(De)select All

`bpy.ops.clip.select_all(action='TOGGLE')`

Properties:	Values:
Action	TOGGLE

Alt-EVT_TWEAK_A → clip.select_lasso : **TWEAK** → ANY

Lasso Select

bpy.ops.clip.select_lasso(path=[], deselect=False, extend=True)

Properties:	Values:
Deselect	False

Shift-Alt-EVT_TWEAK_A → clip.select_lasso : **TWEAK** → ANY

Lasso Select

bpy.ops.clip.select_lasso(path=[], deselect=False, extend=True)

Properties:	Values:
Deselect	True

Alt-EVT_TWEAK_S → clip.select_border : **TWEAK** → ANY

Border Select

bpy.ops.clip.select_border(gesture_mode=0, xmin=0, xmax=0, ymin=0, ymax=0, extend=True)

Properties:	Values:
Extend	False

Shift-Alt-EVT_TWEAK_S → clip.select_border : **TWEAK** → ANY

Border Select

bpy.ops.clip.select_border(gesture_mode=0, xmin=0, xmax=0, ymin=0, ymax=0, extend=True)

Properties:	Values:
Extend	True

Ctrl-ACTIONMOUSE → clip.cursor_set : **MOUSE** → PRESS

Set 2D Cursor

bpy.ops.clip.cursor_set(location=(0, 0))

(default) MIDDLEMOUSE → clip.view_pan : **MOUSE** → PRESS

View Pan

bpy.ops.clip.view_pan(offset=(0, 0))

(default) Shift-MIDDLEMOUSE → clip.view_pan : **MOUSE** → PRESS

View Pan

bpy.ops.clip.view_pan(offset=(0, 0))

(default) TRACKPADPAN → clip.view_pan : **MOUSE** → ANY

View Pan

bpy.ops.clip.view_pan(offset=(0, 0))

(default) Ctrl-MIDDLEMOUSE → clip.view_zoom : **MOUSE** → PRESS

View Zoom

bpy.ops.clip.view_zoom(factor=0)

(default) TRACKPADZOOM → clip.view_zoom : **MOUSE** → ANY

View Zoom

bpy.ops.clip.view_zoom(factor=0)

(default) Ctrl-TRACKPADPAN → clip.view_zoom : **MOUSE** → ANY

View Zoom

bpy.ops.clip.view_zoom(factor=0)

(default) **WHEELINMOUSE** → clip.view_zoom_in : **MOUSE** → PRESS

View Zoom In

bpy.ops.clip.view_zoom_in(location=(0, 0))

(default) **WHEELOUTMOUSE** → clip.view_zoom_out : **MOUSE** → PRESS

View Zoom Out

bpy.ops.clip.view_zoom_out(location=(0, 0))

(default) **NUMPAD_PLUS** → clip.view_zoom_in : **KEYBOARD** → PRESS

View Zoom In

bpy.ops.clip.view_zoom_in(location=(0, 0))

(default) **NUMPAD_MINUS** → clip.view_zoom_out : **KEYBOARD** → PRESS

View Zoom Out

bpy.ops.clip.view_zoom_out(location=(0, 0))

(default) **Ctrl-NUMPAD_8** → clip.view_zoom_ratio : **KEYBOARD** → PRESS

View Zoom Ratio

bpy.ops.clip.view_zoom_ratio(ratio=0)

Properties:	Values:
Ratio	8.0

(default) **Ctrl-NUMPAD_4** → clip.view_zoom_ratio : **KEYBOARD** → PRESS

View Zoom Ratio

bpy.ops.clip.view_zoom_ratio(ratio=0)

Properties:	Values:
Ratio	4.0

(default) **Ctrl-NUMPAD_2** → clip.view_zoom_ratio : **KEYBOARD** → PRESS

View Zoom Ratio

bpy.ops.clip.view_zoom_ratio(ratio=0)

Properties:	Values:
Ratio	2.0

(default) **Shift-NUMPAD_8** → clip.view_zoom_ratio : **KEYBOARD** → PRESS

View Zoom Ratio

bpy.ops.clip.view_zoom_ratio(ratio=0)

Properties:	Values:
Ratio	8.0

(default) **Shift-NUMPAD_4** → clip.view_zoom_ratio : **KEYBOARD** → PRESS

View Zoom Ratio

bpy.ops.clip.view_zoom_ratio(ratio=0)

Properties:	Values:
Ratio	4.0

(default) **Shift-NUMPAD_2** → clip.view_zoom_ratio : **KEYBOARD** → PRESS

View Zoom Ratio

bpy.ops.clip.view_zoom_ratio(ratio=0)

Properties:	Values:
Ratio	2.0

(default) NUMPAD_1 → clip.view_zoom_ratio : **KEYBOARD** → PRESS

View Zoom Ratio

bpy.ops.clip.view_zoom_ratio(ratio=0)

Properties:	Values:
Ratio	1.0

(default) NUMPAD_2 → clip.view_zoom_ratio : **KEYBOARD** → PRESS

View Zoom Ratio

bpy.ops.clip.view_zoom_ratio(ratio=0)

Properties:	Values:
Ratio	0.5

(default) NUMPAD_4 → clip.view_zoom_ratio : **KEYBOARD** → PRESS

View Zoom Ratio

bpy.ops.clip.view_zoom_ratio(ratio=0)

Properties:	Values:
Ratio	0.25

(default) NUMPAD_8 → clip.view_zoom_ratio : **KEYBOARD** → PRESS

View Zoom Ratio

bpy.ops.clip.view_zoom_ratio(ratio=0)

Properties:	Values:
Ratio	0.125

(default) HOME → clip.view_all : **KEYBOARD** → PRESS

View All

bpy.ops.clip.view_all(fit_view=False)

(default) F → clip.view_all : **KEYBOARD** → PRESS

View All

bpy.ops.clip.view_all(fit_view=False)

Properties:	Values:
Fit View	True

(default) NUMPAD_PERIOD → clip.view_selected : **KEYBOARD** → PRESS

View Selected

bpy.ops.clip.view_selected()

(default) NDOF_BUTTON_FIT → clip.view_all : **NDOF** → PRESS

View All

bpy.ops.clip.view_all(fit_view=False)

(default) NDOF_MOTION → clip.view_ndof : **NDOF** → ANY

NDOF Pan/Zoom

bpy.ops.clip.view_ndof()

(default) Ctrl-Shift-LEFT_ARROW → clip.frame_jump : **KEYBOARD** → PRESS

Jump to Frame

bpy.ops.clip.frame_jump(position='PATHSTART')

Properties:	Values:
Position	PATHSTART

(default) Ctrl-Shift-RIGHT_ARROW → clip.frame_jump : **KEYBOARD** → PRESS

Jump to Frame

bpy.ops.clip.frame_jump(position='PATHSTART')

Properties:	Values:
Position	PATHEND

(default) Shift-Alt-LEFT_ARROW → clip.frame_jump : **KEYBOARD** → PRESS

Jump to Frame

bpy.ops.clip.frame_jump(position='PATHSTART')

Properties:	Values:
Position	FAILEDPREV

(default) Shift-Alt-RIGHT_ARROW → clip.frame_jump : **KEYBOARD** → PRESS

Jump to Frame

bpy.ops.clip.frame_jump(position='PATHSTART')

Properties:	Values:
Position	PATHSTART

(default) LEFTMOUSE → clip.change_frame : **MOUSE** → PRESS

Change Frame

bpy.ops.clip.change_frame(frame=0)

(default) SELECTMOUSE → clip.select : **MOUSE** → PRESS

Select

bpy.ops.clip.select(extend=False, location=(0, 0))

Properties:	Values:
Extend	False

(default) Shift-SELECTMOUSE → clip.select : **MOUSE** → PRESS

Select

bpy.ops.clip.select(extend=False, location=(0, 0))

Properties:	Values:
Extend	True

(default) A → clip.select_all : **KEYBOARD** → PRESS

(De)select All

bpy.ops.clip.select_all(action='TOGGLE')

Properties:	Values:
Action	TOGGLE

(default) Ctrl-I → clip.select_all : **KEYBOARD** → PRESS

(De)select All

bpy.ops.clip.select_all(action='TOGGLE')

Properties:	Values:
Action	INVERT

(default) B → clip.select_border : **KEYBOARD** → PRESS

Border Select

bpy.ops.clip.select_border(gesture_mode=0, xmin=0, xmax=0, ymin=0, ymax=0, extend=True)

(default) C → clip.select_circle : **KEYBOARD** → PRESS

Circle Select

bpy.ops.clip.select_circle(x=0, y=0, radius=1, gesture_mode=0)

(default) Shift-G → wm.call_menu : **KEYBOARD** → PRESS

Call Menu

bpy.ops.wm.call_menu(name='')

Properties:	Values:
Name	CLIP_MT_select_grouped

(default) Ctrl-Alt-EVT_TWEAK_A → clip.select_lasso : **TWEAK** → ANY

Lasso Select

bpy.ops.clip.select_lasso(path=[], deselect=False, extend=True)

Properties:	Values:
Deselect	False

(default) Ctrl-Shift-Alt-EVT_TWEAK_A → clip.select_lasso : **TWEAK** → ANY

Lasso Select

bpy.ops.clip.select_lasso(path=[], deselect=False, extend=True)

Properties:	Values:
Deselect	True

(default) Ctrl-LEFTMOUSE → clip.add_marker_slide : **MOUSE** → PRESS

Add Marker and Slide

bpy.ops.clip.add_marker_slide(CLIP_OT_add_marker={"location":(0, 0)}, TRANSFORM_OT_translate={"value":(0, 0, 0), "constraint_axis":(False, False, False), "constraint_orientation":'GLOBAL', "mirror":False, "proportional":'DISABLED', "proportional_edit_falloff":'SMOOTH', "proportional_size":1, "snap":False, "snap_target":'CLOSEST', "snap_point":(0, 0, 0), "snap_align":False, "snap_normal":(0, 0, 0), "gpencil_strokes":False, "texture_space":False, "remove_on_cancel":False, "release_confirm":False})

Properties:	Values:
Add Marker	N/A
Translate	N/A

(default) Shift-DEL → clip.delete_marker : **KEYBOARD** → PRESS

Delete Marker

bpy.ops.clip.delete_marker()

(default) Shift-X → clip.delete_marker : **KEYBOARD** → PRESS

Delete Marker

bpy.ops.clip.delete_marker()

(default) LEFTMOUSE → clip.slide_marker : **MOUSE** → PRESS

Slide Marker

bpy.ops.clip.slide_marker(offset=(0, 0))

(default) Shift-D → clip.disable_markers : **KEYBOARD** → PRESS
Disable Markers

bpy.ops.clip.disable_markers(action='DISABLE')

Properties:	Values:
Action	TOGGLE

(default) DEL → clip.delete_track : **KEYBOARD** → PRESS
Delete Track

bpy.ops.clip.delete_track()

(default) X → clip.delete_track : **KEYBOARD** → PRESS
Delete Track

bpy.ops.clip.delete_track()

(default) Ctrl-L → clip.lock_tracks : **KEYBOARD** → PRESS
Lock Tracks

bpy.ops.clip.lock_tracks(action='LOCK')

Properties:	Values:
Action	LOCK

(default) Alt-L → clip.lock_tracks : **KEYBOARD** → PRESS
Lock Tracks

bpy.ops.clip.lock_tracks(action='LOCK')

Properties:	Values:
Action	UNLOCK

(default) H → clip.hide_tracks : **KEYBOARD** → PRESS
Hide Tracks

bpy.ops.clip.hide_tracks(unselected=False)

Properties:	Values:
Unselected	False

(default) Shift-H → clip.hide_tracks : **KEYBOARD** → PRESS
Hide Tracks

bpy.ops.clip.hide_tracks(unselected=False)

Properties:	Values:
Unselected	True

(default) Alt-H → clip.hide_tracks_clear : **KEYBOARD** → PRESS
Hide Tracks Clear

bpy.ops.clip.hide_tracks_clear()

(default) ACTIONMOUSE → clip.slide_plane_marker : **MOUSE** → PRESS
Slide Plane Marker

bpy.ops.clip.slide_plane_marker()

(default) I → clip.keyframe_insert : **KEYBOARD** → PRESS
Insert keyframe

bpy.ops.clip.keyframe_insert()

(default) Alt-I → clip.keyframe_delete : **KEYBOARD** → PRESS

Delete keyframe

bpy.ops.clip.keyframe_delete()

(default) Ctrl-J → clip.join_tracks : **KEYBOARD** → PRESS

Join Tracks

bpy.ops.clip.join_tracks()

(default) W → wm.call_menu : **KEYBOARD** → PRESS

Call Menu

bpy.ops.wm.call_menu(name='')

Properties:	Values:
Name	CLIP_MT_tracking_specials

(default) L → wm.context_toggle : **KEYBOARD** → PRESS

Context Toggle

bpy.ops.wm.context_toggle(data_path='')

Properties:	Values:
Context Attributes	space_data.lock_selection

(default) Alt-D → wm.context_toggle : **KEYBOARD** → PRESS

Context Toggle

bpy.ops.wm.context_toggle(data_path='')

Properties:	Values:
Context Attributes	space_data.show_disabled

(default) Alt-S → wm.context_toggle : **KEYBOARD** → PRESS

Context Toggle

bpy.ops.wm.context_toggle(data_path='')

Properties:	Values:
Context Attributes	space_data.show_marker_search

(default) M → wm.context_toggle : **KEYBOARD** → PRESS

Context Toggle

bpy.ops.wm.context_toggle(data_path='')

Properties:	Values:
Context Attributes	space_data.use_mute_footage

(default) G → transform.translate : **KEYBOARD** → PRESS

Translate

bpy.ops.transform.translate(value=(0, 0, 0), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpencil_strokes=False, texture_space=False, remove_on_cancel=False, release_confirm=False)

(default) EVT_TWEAK_S → transform.translate : **TWEAK** → ANY

Translate

bpy.ops.transform.translate(value=(0, 0, 0), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0,

0, 0), snap_align=False, snap_normal=(0, 0, 0), gpencil_strokes=False, texture_space=False, remove_on_cancel=False, release_confirm=False)

(default) S → transform.resize : **KEYBOARD** → PRESS

Resize

```
bpy.ops.transform.resize(value=(1, 1, 1), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpencil_strokes=False, texture_space=False, remove_on_cancel=False, release_confirm=False)
```

(default) R → transform.rotate : **KEYBOARD** → PRESS

Rotate

```
bpy.ops.transform.rotate(value=0, axis=(0, 0, 0), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpencil_strokes=False, release_confirm=False)
```

(default) Alt-T → clip.clear_track_path : **KEYBOARD** → PRESS

Clear Track Path

```
bpy.ops.clip.clear_track_path(action='REMAINED', clear_active=False)
```

Properties:	Values:
Action	REMAINED
Clear Active	False

(default) Shift-T → clip.clear_track_path : **KEYBOARD** → PRESS

Clear Track Path

```
bpy.ops.clip.clear_track_path(action='REMAINED', clear_active=False)
```

Properties:	Values:
Action	UPTO
Clear Active	False

(default) Shift-Alt-T → clip.clear_track_path : **KEYBOARD** → PRESS

Clear Track Path

```
bpy.ops.clip.clear_track_path(action='REMAINED', clear_active=False)
```

Properties:	Values:
Action	ALL
Clear Active	False

(default) ACTIONMOUSE → clip.cursor_set : **MOUSE** → PRESS

Set 2D Cursor

```
bpy.ops.clip.cursor_set(location=(0, 0))
```

(default) , → wm.context_set_enum : **KEYBOARD** → PRESS

Context Set Enum

```
bpy.ops.wm.context_set_enum(data_path="", value="")
```

Properties:	Values:
Context Attributes	space_data.pivot_point
Value	BOUNDING_BOX_CENTER

(default) Ctrl-, → wm.context_set_enum : **KEYBOARD** → PRESS

Context Set Enum

```
bpy.ops.wm.context_set_enum(data_path="", value="")
```

Properties:	Values:
Context Attributes	space_data.pivot_point
Value	MEDIAN_POINT

(default) **.** → **wm.context_set_enum** : **KEYBOARD** → PRESS

Context Set Enum

```
bpy.ops.wm.context_set_enum(data_path="", value="")
```

Properties:	Values:
Context Attributes	space_data.pivot_point
Value	CURSOR

(default) **Ctrl-** → **wm.context_set_enum** : **KEYBOARD** → PRESS

Context Set Enum

```
bpy.ops.wm.context_set_enum(data_path="", value="")
```

Properties:	Values:
Context Attributes	space_data.pivot_point
Value	INDIVIDUAL_ORIGINS

(default) **Ctrl-C** → **clip.copy_tracks** : **KEYBOARD** → PRESS

Copy Tracks

```
bpy.ops.clip.copy_tracks()
```

(default) **Ctrl-V** → **clip.paste_tracks** : **KEYBOARD** → PRESS

Paste Tracks

```
bpy.ops.clip.paste_tracks()
```

Clip Graph Editor

Quick Reference

Hotkey	Operator
<i>Ctrl-ACTIONMOUSE</i>	<code>bpy.ops.clip.change_frame()</code>
<i>Alt-EVT_TWEAK_S</i>	<code>bpy.ops.clip.graph_select_border()</code>
<i>Shift-Alt-EVT_TWEAK_S</i>	<code>bpy.ops.clip.graph_select_border()</code>
<i>Ctrl-A</i>	<code>bpy.ops.clip.graph_select_all_markers()</code>
<i>ACTIONMOUSE</i>	<code>bpy.ops.clip.change_frame()</code>
<i>SELECTMOUSE</i>	<code>bpy.ops.clip.graph_select()</code>
<i>Shift-SELECTMOUSE</i>	<code>bpy.ops.clip.graph_select()</code>
<i>A</i>	<code>bpy.ops.clip.graph_select_all_markers()</code>
<i>Ctrl-I</i>	<code>bpy.ops.clip.graph_select_all_markers()</code>
<i>B</i>	<code>bpy.ops.clip.graph_select_border()</code>
<i>DEL</i>	<code>bpy.ops.clip.graph_delete_curve()</code>
<i>X</i>	<code>bpy.ops.clip.graph_delete_curve()</code>
<i>Shift-DEL</i>	<code>bpy.ops.clip.graph_delete_knot()</code>
<i>Shift-X</i>	<code>bpy.ops.clip.graph_delete_knot()</code>
<i>HOME</i>	<code>bpy.ops.clip.graph_view_all()</code>
<i>NDOF_BUTTON_FIT</i>	<code>bpy.ops.clip.graph_view_all()</code>
<i>NUMPAD_PERIOD</i>	<code>bpy.ops.clip.graph_center_current_frame()</code>
<i>L</i>	<code>bpy.ops.wm.context_toggle()</code>
<i>Alt-T</i>	<code>bpy.ops.clip.clear_track_path()</code>
<i>Shift-T</i>	<code>bpy.ops.clip.clear_track_path()</code>
<i>Shift-Alt-T</i>	<code>bpy.ops.clip.clear_track_path()</code>
<i>Shift-D</i>	<code>bpy.ops.clip.graph_disable_markers()</code>
<i>G</i>	<code>bpy.ops.transform.translate()</code>
<i>EVT_TWEAK_S</i>	<code>bpy.ops.transform.translate()</code>
<i>S</i>	<code>bpy.ops.transform.resize()</code>
<i>R</i>	<code>bpy.ops.transform.rotate()</code>

Detailed Reference

Note: Hotkeys marked with the “(default)” prefix are inherited from the default blender keymap

Ctrl-ACTIONMOUSE → clip.change_frame : **MOUSE** → PRESS

Change Frame

`bpy.ops.clip.change_frame(frame=0)`

Alt-EVT_TWEAK_S → clip.graph_select_border : **TWEAK** → ANY

Border Select

`bpy.ops.clip.graph_select_border(gesture_mode=0, xmin=0, xmax=0, ymin=0, ymax=0, extend=True)`

Properties:	Values:
Extend	False

Shift-Alt-EVT_TWEAK_S → clip.graph_select_border : **TWEAK** → ANY

Border Select

bpy.ops.clip.graph_select_border(gesture_mode=0, xmin=0, xmax=0, ymin=0, ymax=0, extend=True)

Properties:	Values:
Extend	True

Ctrl-A → clip.graph_select_all_markers : **KEYBOARD** → PRESS
(De)select All Markers

bpy.ops.clip.graph_select_all_markers(action='TOGGLE')

Properties:	Values:
Action	TOGGLE

(default) ACTIONMOUSE → clip.change_frame : **MOUSE** → PRESS
Change Frame

bpy.ops.clip.change_frame(frame=0)

(default) SELECTMOUSE → clip.graph_select : **MOUSE** → PRESS
Select

bpy.ops.clip.graph_select(location=(0, 0), extend=False)

Properties:	Values:
Extend	False

(default) Shift-SELECTMOUSE → clip.graph_select : **MOUSE** → PRESS
Select

bpy.ops.clip.graph_select(location=(0, 0), extend=False)

Properties:	Values:
Extend	True

(default) A → clip.graph_select_all_markers : **KEYBOARD** → PRESS
(De)select All Markers

bpy.ops.clip.graph_select_all_markers(action='TOGGLE')

Properties:	Values:
Action	TOGGLE

(default) Ctrl-I → clip.graph_select_all_markers : **KEYBOARD** → PRESS
(De)select All Markers

bpy.ops.clip.graph_select_all_markers(action='TOGGLE')

Properties:	Values:
Action	INVERT

(default) B → clip.graph_select_border : **KEYBOARD** → PRESS
Border Select

bpy.ops.clip.graph_select_border(gesture_mode=0, xmin=0, xmax=0, ymin=0, ymax=0, extend=True)

(default) DEL → clip.graph_delete_curve : **KEYBOARD** → PRESS
Delete Curve

bpy.ops.clip.graph_delete_curve()

(default) X → clip.graph_delete_curve : **KEYBOARD** → PRESS
Delete Curve

bpy.ops.clip.graph_delete_curve()

(default) Shift-DEL → clip.graph_delete_knot : **KEYBOARD** → PRESS
Delete Knot

bpy.ops.clip.graph_delete_knot()

(default) Shift-X → clip.graph_delete_knot : **KEYBOARD** → PRESS
Delete Knot

bpy.ops.clip.graph_delete_knot()

(default) HOME → clip.graph_view_all : **KEYBOARD** → PRESS
View All

bpy.ops.clip.graph_view_all()

(default) NDOF_BUTTON_FIT → clip.graph_view_all : **NDOF** → PRESS
View All

bpy.ops.clip.graph_view_all()

(default) NUMPAD_PERIOD → clip.graph_center_current_frame : **KEYBOARD** → PRESS
Center Current Frame

bpy.ops.clip.graph_center_current_frame()

(default) L → wm.context_toggle : **KEYBOARD** → PRESS
Context Toggle

bpy.ops.wm.context_toggle(data_path='')

Properties:	Values:
Context Attributes	space_data.lock_time_cursor

(default) Alt-T → clip.clear_track_path : **KEYBOARD** → PRESS
Clear Track Path

bpy.ops.clip.clear_track_path(action='REMAINED', clear_active=False)

Properties:	Values:
Action	REMAINED
Clear Active	True

(default) Shift-T → clip.clear_track_path : **KEYBOARD** → PRESS
Clear Track Path

bpy.ops.clip.clear_track_path(action='REMAINED', clear_active=False)

Properties:	Values:
Action	UPTO
Clear Active	True

(default) Shift-Alt-T → clip.clear_track_path : **KEYBOARD** → PRESS
Clear Track Path

bpy.ops.clip.clear_track_path(action='REMAINED', clear_active=False)

Properties:	Values:
Action	ALL
Clear Active	True

(default) Shift-D → clip.graph_disable_markers : **KEYBOARD** → PRESS
Disable Markers

bpy.ops.clip.graph_disable_markers(action='DISABLE')

Properties:	Values:
Action	TOGGLE

(default) G → transform.translate : **KEYBOARD** → PRESS

Translate

```
bpy.ops.transform.translate(value=(0, 0, 0), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpencil_strokes=False, texture_space=False, remove_on_cancel=False, release_confirm=False)
```

(default) EVT_TWEAK_S → transform.translate : **TWEAK** → ANY

Translate

```
bpy.ops.transform.translate(value=(0, 0, 0), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpencil_strokes=False, texture_space=False, remove_on_cancel=False, release_confirm=False)
```

(default) S → transform.resize : **KEYBOARD** → PRESS

Resize

```
bpy.ops.transform.resize(value=(1, 1, 1), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpencil_strokes=False, texture_space=False, remove_on_cancel=False, release_confirm=False)
```

(default) R → transform.rotate : **KEYBOARD** → PRESS

Rotate

```
bpy.ops.transform.rotate(value=0, axis=(0, 0, 0), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpencil_strokes=False, release_confirm=False)
```

Console

Quick Reference

Hotkey	Operator
<i>Tab</i>	<code>bpy.ops.console.autocomplete()</code>
<i>Ctrl-LEFT_ARROW</i>	<code>bpy.ops.console.move()</code>
<i>Ctrl-RIGHT_ARROW</i>	<code>bpy.ops.console.move()</code>
<i>HOME</i>	<code>bpy.ops.console.move()</code>
<i>END</i>	<code>bpy.ops.console.move()</code>
<i>Ctrl-WHEELUPMOUSE</i>	<code>bpy.ops.wm.context_cycle_int()</code>
<i>Ctrl-WHEELDOWNMOUSE</i>	<code>bpy.ops.wm.context_cycle_int()</code>
<i>Ctrl-NUMPAD_PLUS</i>	<code>bpy.ops.wm.context_cycle_int()</code>
<i>Ctrl-NUMPAD_MINUS</i>	<code>bpy.ops.wm.context_cycle_int()</code>
<i>LEFT_ARROW</i>	<code>bpy.ops.console.move()</code>
<i>RIGHT_ARROW</i>	<code>bpy.ops.console.move()</code>

Continued on next page

Table 2.4 – continued from previous page

Hotkey	Operator
<i>UP_ARROW</i>	<code>bpy.ops.console.history_cycle()</code>
<i>DOWN_ARROW</i>	<code>bpy.ops.console.history_cycle()</code>
<i>DEL</i>	<code>bpy.ops.console.delete()</code>
<i>BACK_SPACE</i>	<code>bpy.ops.console.delete()</code>
<i>Shift-BACK_SPACE</i>	<code>bpy.ops.console.delete()</code>
<i>Ctrl-DEL</i>	<code>bpy.ops.console.delete()</code>
<i>Ctrl-BACK_SPACE</i>	<code>bpy.ops.console.delete()</code>
<i>Shift-RET</i>	<code>bpy.ops.console.clear_line()</code>
<i>Shift-NUMPAD_ENTER</i>	<code>bpy.ops.console.clear_line()</code>
<i>RET</i>	<code>bpy.ops.console.execute()</code>
<i>NUMPAD_ENTER</i>	<code>bpy.ops.console.execute()</code>
<i>Ctrl-SPACE</i>	<code>bpy.ops.console.autocomplete()</code>
<i>Ctrl-Shift-C</i>	<code>bpy.ops.console.copy_as_script()</code>
<i>Ctrl-C</i>	<code>bpy.ops.console.copy()</code>
<i>Ctrl-V</i>	<code>bpy.ops.console.paste()</code>
<i>LEFTMOUSE</i>	<code>bpy.ops.console.select_set()</code>
<i>LEFTMOUSE</i>	<code>bpy.ops.console.select_word()</code>
<i>Ctrl-Tab</i>	<code>bpy.ops.console.insert()</code>
<i>Tab</i>	<code>bpy.ops.console.indent()</code>
<i>Shift-Tab</i>	<code>bpy.ops.console.unindent()</code>
<i>Any-TEXTINPUT</i>	<code>bpy.ops.console.insert()</code>

Detailed Reference

Note: Hotkeys marked with the “(default)” prefix are inherited from the default blender keymap

Tab → `console.autocomplete` : **KEYBOARD** → PRESS

Console Autocomplete

`bpy.ops.console.autocomplete()`

(default) Ctrl-LEFT_ARROW → `console.move` : **KEYBOARD** → PRESS

Move Cursor

`bpy.ops.console.move(type='LINE_BEGIN')`

Properties:	Values:
Type	PREVIOUS_WORD

(default) Ctrl-RIGHT_ARROW → `console.move` : **KEYBOARD** → PRESS

Move Cursor

`bpy.ops.console.move(type='LINE_BEGIN')`

Properties:	Values:
Type	NEXT_WORD

(default) HOME → `console.move` : **KEYBOARD** → PRESS

Move Cursor

`bpy.ops.console.move(type='LINE_BEGIN')`

Properties:	Values:
Type	LINE_BEGIN

(default) END → console.move : **KEYBOARD** → PRESS

Move Cursor

bpy.ops.console.move(type='LINE_BEGIN')

Properties:	Values:
Type	LINE_END

(default) Ctrl-WHEELUPMOUSE → wm.context_cycle_int : **MOUSE** → PRESS

Context Int Cycle

bpy.ops.wm.context_cycle_int(data_path="", reverse=False, wrap=False)

Properties:	Values:
Context Attributes	space_data.font_size
Reverse	False

(default) Ctrl-WHEELDOWNMOUSE → wm.context_cycle_int : **MOUSE** → PRESS

Context Int Cycle

bpy.ops.wm.context_cycle_int(data_path="", reverse=False, wrap=False)

Properties:	Values:
Context Attributes	space_data.font_size
Reverse	True

(default) Ctrl-NUMPAD_PLUS → wm.context_cycle_int : **KEYBOARD** → PRESS

Context Int Cycle

bpy.ops.wm.context_cycle_int(data_path="", reverse=False, wrap=False)

Properties:	Values:
Context Attributes	space_data.font_size
Reverse	False

(default) Ctrl-NUMPAD_MINUS → wm.context_cycle_int : **KEYBOARD** → PRESS

Context Int Cycle

bpy.ops.wm.context_cycle_int(data_path="", reverse=False, wrap=False)

Properties:	Values:
Context Attributes	space_data.font_size
Reverse	True

(default) LEFT_ARROW → console.move : **KEYBOARD** → PRESS

Move Cursor

bpy.ops.console.move(type='LINE_BEGIN')

Properties:	Values:
Type	PREVIOUS_CHARACTER

(default) RIGHT_ARROW → console.move : **KEYBOARD** → PRESS

Move Cursor

bpy.ops.console.move(type='LINE_BEGIN')

Properties:	Values:
Type	NEXT_CHARACTER

(default) UP_ARROW → console.history_cycle : **KEYBOARD** → PRESS

History Cycle

bpy.ops.console.history_cycle(reverse=False)

Properties:	Values:
Reverse	True

(default) **DOWN_ARROW** → console.history_cycle : **KEYBOARD** → PRESS

History Cycle

bpy.ops.console.history_cycle(reverse=False)

Properties:	Values:
Reverse	False

(default) **DEL** → console.delete : **KEYBOARD** → PRESS

Delete

bpy.ops.console.delete(type='NEXT_CHARACTER')

Properties:	Values:
Type	NEXT_CHARACTER

(default) **BACK_SPACE** → console.delete : **KEYBOARD** → PRESS

Delete

bpy.ops.console.delete(type='NEXT_CHARACTER')

Properties:	Values:
Type	PREVIOUS_CHARACTER

(default) **Shift-BACK_SPACE** → console.delete : **KEYBOARD** → PRESS

Delete

bpy.ops.console.delete(type='NEXT_CHARACTER')

Properties:	Values:
Type	PREVIOUS_CHARACTER

(default) **Ctrl-DEL** → console.delete : **KEYBOARD** → PRESS

Delete

bpy.ops.console.delete(type='NEXT_CHARACTER')

Properties:	Values:
Type	NEXT_WORD

(default) **Ctrl-BACK_SPACE** → console.delete : **KEYBOARD** → PRESS

Delete

bpy.ops.console.delete(type='NEXT_CHARACTER')

Properties:	Values:
Type	PREVIOUS_WORD

(default) **Shift-RET** → console.clear_line : **KEYBOARD** → PRESS

Clear Line

bpy.ops.console.clear_line()

(default) **Shift-NUMPAD_ENTER** → console.clear_line : **KEYBOARD** → PRESS

Clear Line

bpy.ops.console.clear_line()

(default) **RET** → console.execute : **KEYBOARD** → PRESS

Console Execute

bpy.ops.console.execute(interactive=False)

Properties:	Values:
interactive	True

(default) NUMPAD_ENTER → console.execute : **KEYBOARD** → PRESS

Console Execute

bpy.ops.console.execute(interactive=False)

Properties:	Values:
interactive	True

(default) Ctrl-SPACE → console.autocomplete : **KEYBOARD** → PRESS

Console Autocomplete

bpy.ops.console.autocomplete()

(default) Ctrl-Shift-C → console.copy_as_script : **KEYBOARD** → PRESS

Copy to Clipboard (as script)

bpy.ops.console.copy_as_script()

(default) Ctrl-C → console.copy : **KEYBOARD** → PRESS

Copy to Clipboard

bpy.ops.console.copy()

(default) Ctrl-V → console.paste : **KEYBOARD** → PRESS

Paste from Clipboard

bpy.ops.console.paste()

(default) LEFTMOUSE → console.select_set : **MOUSE** → PRESS

Set Selection

bpy.ops.console.select_set()

(default) LEFTMOUSE → console.select_word : **MOUSE** → DOUBLE_CLICK

Select Word

bpy.ops.console.select_word()

(default) Ctrl-Tab → console.insert : **KEYBOARD** → PRESS

Insert

bpy.ops.console.insert(text="")

Properties:	Values:
Text	\t

(default) Tab → console.indent : **KEYBOARD** → PRESS

Indent

bpy.ops.console.indent()

(default) Shift-Tab → console.unindent : **KEYBOARD** → PRESS

Unindent

bpy.ops.console.unindent()

(default) Any-TEXTINPUT → console.insert : **TEXTINPUT** → ANY

Insert

bpy.ops.console.insert(text="")

Curve

Quick Reference

Hotkey	Operator
<i>Ctrl-A</i>	<code>bpy.ops.curve.select_all()</code>
<i>Ctrl-Alt-SELECTMOUSE</i>	<code>bpy.ops.curve.vertex_add()</code>
<i>Shift-A</i>	<code>bpy.ops.wm.call_menu()</code>
<i>V</i>	<code>bpy.ops.curve.handle_type_set()</code>
<i>Ctrl-ACTIONMOUSE</i>	<code>bpy.ops.curve.vertex_add()</code>
<i>Shift-ACTIONMOUSE</i>	<code>bpy.ops.curve.draw()</code>
<i>A</i>	<code>bpy.ops.curve.select_all()</code>
<i>Ctrl-I</i>	<code>bpy.ops.curve.select_all()</code>
<i>Shift-R</i>	<code>bpy.ops.curve.select_row()</code>
<i>Ctrl-NUMPAD_PLUS</i>	<code>bpy.ops.curve.select_more()</code>
<i>Ctrl-NUMPAD_MINUS</i>	<code>bpy.ops.curve.select_less()</code>
<i>Ctrl-L</i>	<code>bpy.ops.curve.select_linked()</code>
<i>Shift-G</i>	<code>bpy.ops.curve.select_similar()</code>
<i>L</i>	<code>bpy.ops.curve.select_linked_pick()</code>
<i>Shift-L</i>	<code>bpy.ops.curve.select_linked_pick()</code>
<i>Ctrl-SELECTMOUSE</i>	<code>bpy.ops.curve.shortest_path_pick()</code>
<i>P</i>	<code>bpy.ops.curve.separate()</code>
<i>Y</i>	<code>bpy.ops.curve.split()</code>
<i>E</i>	<code>bpy.ops.curve.extrude_move()</code>
<i>Shift-D</i>	<code>bpy.ops.curve.duplicate_move()</code>
<i>F</i>	<code>bpy.ops.curve.make_segment()</code>
<i>Alt-C</i>	<code>bpy.ops.curve.cyclic_toggle()</code>
<i>X</i>	<code>bpy.ops.wm.call_menu()</code>
<i>DEL</i>	<code>bpy.ops.wm.call_menu()</code>
<i>Ctrl-X</i>	<code>bpy.ops.curve.dissolve_verts()</code>
<i>Ctrl-DEL</i>	<code>bpy.ops.curve.dissolve_verts()</code>
<i>Alt-T</i>	<code>bpy.ops.curve.tilt_clear()</code>
<i>Ctrl-T</i>	<code>bpy.ops.transform.tilt()</code>
<i>Alt-S</i>	<code>bpy.ops.transform.transform()</code>
<i>Alt-H</i>	<code>bpy.ops.curve.reveal()</code>
<i>H</i>	<code>bpy.ops.curve.hide()</code>
<i>Shift-H</i>	<code>bpy.ops.curve.hide()</code>
<i>Ctrl-N</i>	<code>bpy.ops.curve.normals_make_consistent()</code>
<i>Ctrl-P</i>	<code>bpy.ops.object.vertex_parent_set()</code>
<i>W</i>	<code>bpy.ops.wm.call_menu()</code>
<i>Ctrl-H</i>	<code>bpy.ops.wm.call_menu()</code>
<i>Shift-O</i>	<code>bpy.ops.wm.context_cycle_enum()</code>
<i>O</i>	<code>bpy.ops.wm.context_toggle_enum()</code>
<i>Alt-O</i>	<code>bpy.ops.wm.context_toggle_enum()</code>

Detailed Reference

Note: Hotkeys marked with the “(default)” prefix are inherited from the default blender keymap

Ctrl-A → curve.select_all : **KEYBOARD** → PRESS

(De)select All

bpy.ops.curve.select_all(action='TOGGLE')

Properties:	Values:
Action	TOGGLE

Ctrl-Alt-SELECTMOUSE → curve.vertex_add : **MOUSE** → CLICK

Add Vertex

bpy.ops.curve.vertex_add(location=(0, 0, 0))

(default) Shift-A → wm.call_menu : **KEYBOARD** → PRESS

Call Menu

bpy.ops.wm.call_menu(name='')

Properties:	Values:
Name	INFO_MT_edit_curve_add

(default) V → curve.handle_type_set : **KEYBOARD** → PRESS

Set Handle Type

bpy.ops.curve.handle_type_set(type='AUTOMATIC')

(default) Ctrl-ACTIONMOUSE → curve.vertex_add : **MOUSE** → CLICK

Add Vertex

bpy.ops.curve.vertex_add(location=(0, 0, 0))

(default) Shift-ACTIONMOUSE → curve.draw : **MOUSE** → PRESS

Draw Curve

bpy.ops.curve.draw(error_threshold=0, fit_method='REFIT', corner_angle=1.22173, use_cyclic=True, stroke=[], wait_for_input=True)

Properties:	Values:
Wait for Input	False

(default) A → curve.select_all : **KEYBOARD** → PRESS

(De)select All

bpy.ops.curve.select_all(action='TOGGLE')

Properties:	Values:
Action	TOGGLE

(default) Ctrl-I → curve.select_all : **KEYBOARD** → PRESS

(De)select All

bpy.ops.curve.select_all(action='TOGGLE')

Properties:	Values:
Action	INVERT

(default) Shift-R → curve.select_row : **KEYBOARD** → PRESS

Select Control Point Row

bpy.ops.curve.select_row()

(default) Ctrl-NUMPAD_PLUS → curve.select_more : **KEYBOARD** → PRESS

Select More

bpy.ops.curve.select_more()

(default) **Ctrl-NUMPAD_MINUS** → curve.select_less : **KEYBOARD** → PRESS

Select Less

bpy.ops.curve.select_less()

(default) **Ctrl-L** → curve.select_linked : **KEYBOARD** → PRESS

Select Linked All

bpy.ops.curve.select_linked()

(default) **Shift-G** → curve.select_similar : **KEYBOARD** → PRESS

Select Similar

bpy.ops.curve.select_similar(type='WEIGHT', compare='EQUAL', threshold=0.1)

(default) **L** → curve.select_linked_pick : **KEYBOARD** → PRESS

Select Linked

bpy.ops.curve.select_linked_pick(deselect=False)

Properties:	Values:
Deselect	False

(default) **Shift-L** → curve.select_linked_pick : **KEYBOARD** → PRESS

Select Linked

bpy.ops.curve.select_linked_pick(deselect=False)

Properties:	Values:
Deselect	True

(default) **Ctrl-SELECTMOUSE** → curve.shortest_path_pick : **MOUSE** → CLICK

Pick Shortest Path

bpy.ops.curve.shortest_path_pick()

(default) **P** → curve.separate : **KEYBOARD** → PRESS

Separate

bpy.ops.curve.separate()

(default) **Y** → curve.split : **KEYBOARD** → PRESS

Split

bpy.ops.curve.split()

(default) **E** → curve.extrude_move : **KEYBOARD** → PRESS

Extrude Curve and Move

bpy.ops.curve.extrude_move(CURVE_OT_extrude={"mode":'TRANSLATION'},
FORM_OT_translate={"value":(0, 0, 0), "constraint_axis":(False, False, False), "con-
straint_orientation":'GLOBAL', "mirror":False, "proportional":'DISABLED', "propor-
tional_edit_falloff":'SMOOTH', "proportional_size":1, "snap":False, "snap_target":'CLOSEST',
"snap_point":(0, 0, 0), "snap_align":False, "snap_normal":(0, 0, 0), "gpencil_strokes":False, "tex-
ture_space":False, "remove_on_cancel":False, "release_confirm":False})

Properties:	Values:
Extrude	N/A
Translate	N/A

(default) **Shift-D** → curve.duplicate_move : **KEYBOARD** → PRESS

Add Duplicate


```
bpy.ops.curve.duplicate_move(CURVE_OT_duplicate={}, TRANSFORM_OT_translate={"value":(0, 0, 0), "constraint_axis":(False, False, False), "constraint_orientation":'GLOBAL', "mirror":False, "proportional":'DISABLED', "proportional_edit_falloff":'SMOOTH', "proportional_size":1, "snap":False, "snap_target":'CLOSEST', "snap_point":(0, 0, 0), "snap_align":False, "snap_normal":(0, 0, 0), "gpencil_strokes":False, "texture_space":False, "remove_on_cancel":False, "release_confirm":False})
```

Properties:	Values:
Duplicate Curve	N/A
Translate	N/A

(default) F → curve.make_segment : **KEYBOARD** → PRESS

Make Segment

```
bpy.ops.curve.make_segment()
```

(default) Alt-C → curve.cyclic_toggle : **KEYBOARD** → PRESS

Toggle Cyclic

```
bpy.ops.curve.cyclic_toggle(direction='CYCLIC_U')
```

(default) X → wm.call_menu : **KEYBOARD** → PRESS

Call Menu

```
bpy.ops.wm.call_menu(name='')
```

Properties:	Values:
Name	VIEW3D_MT_edit_curve_delete

(default) DEL → wm.call_menu : **KEYBOARD** → PRESS

Call Menu

```
bpy.ops.wm.call_menu(name='')
```

Properties:	Values:
Name	VIEW3D_MT_edit_curve_delete

(default) Ctrl-X → curve.dissolve_verts : **KEYBOARD** → PRESS

Dissolve Vertices

```
bpy.ops.curve.dissolve_verts()
```

(default) Ctrl-DEL → curve.dissolve_verts : **KEYBOARD** → PRESS

Dissolve Vertices

```
bpy.ops.curve.dissolve_verts()
```

(default) Alt-T → curve.tilt_clear : **KEYBOARD** → PRESS

Clear Tilt

```
bpy.ops.curve.tilt_clear()
```

(default) Ctrl-T → transform.tilt : **KEYBOARD** → PRESS

Tilt

```
bpy.ops.transform.tilt(value=0, mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), release_confirm=False)
```

(default) Alt-S → transform.transform : **KEYBOARD** → PRESS

Transform

```
bpy.ops.transform.transform(mode='TRANSLATION', value=(0, 0, 0, 0), axis=(0, 0, 0), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False,
```

```
snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpen-
cil_strokes=False, release_confirm=False)
```

Properties:	Values:
Mode	CURVE_SHRINKFATTEN

(default) Alt-H → curve.reveal : **KEYBOARD** → PRESS

Reveal Hidden

```
bpy.ops.curve.reveal()
```

(default) H → curve.hide : **KEYBOARD** → PRESS

Hide Selected

```
bpy.ops.curve.hide(unselected=False)
```

Properties:	Values:
Unselected	False

(default) Shift-H → curve.hide : **KEYBOARD** → PRESS

Hide Selected

```
bpy.ops.curve.hide(unselected=False)
```

Properties:	Values:
Unselected	True

(default) Ctrl-N → curve.normals_make_consistent : **KEYBOARD** → PRESS

Recalc Normals

```
bpy.ops.curve.normals_make_consistent(calc_length=False)
```

(default) Ctrl-P → object.vertex_parent_set : **KEYBOARD** → PRESS

Make Vertex Parent

```
bpy.ops.object.vertex_parent_set()
```

(default) W → wm.call_menu : **KEYBOARD** → PRESS

Call Menu

```
bpy.ops.wm.call_menu(name='')
```

Properties:	Values:
Name	VIEW3D_MT_edit_curve_specials

(default) Ctrl-H → wm.call_menu : **KEYBOARD** → PRESS

Call Menu

```
bpy.ops.wm.call_menu(name='')
```

Properties:	Values:
Name	VIEW3D_MT_hook

(default) Shift-O → wm.context_cycle_enum : **KEYBOARD** → PRESS

Context Enum Cycle

```
bpy.ops.wm.context_cycle_enum(data_path='', reverse=False, wrap=False)
```

Properties:	Values:
Context Attributes	tool_settings.proportional_edit_falloff
Wrap	True

(default) O → wm.context_toggle_enum : **KEYBOARD** → PRESS

Context Toggle Values

```
bpy.ops.wm.context_toggle_enum(data_path='', value_1='', value_2='')
```

Properties:	Values:
Context Attributes	tool_settings.proportional_edit
Value	DISABLED
Value	ENABLED

(default) **Alt-O** → `wm.context_toggle_enum` : **KEYBOARD** → PRESS

Context Toggle Values

`bpy.ops.wm.context_toggle_enum(data_path="", value_1="", value_2="")`

Properties:	Values:
Context Attributes	tool_settings.proportional_edit
Value	DISABLED
Value	CONNECTED

Dopesheet

Quick Reference

Hotkey	Operator
<i>Ctrl-A</i>	<code>bpy.ops.action.select_all_toggle()</code>
<i>SELECTMOUSE</i>	<code>bpy.ops.action.clickselect()</code>
<i>Alt-SELECTMOUSE</i>	<code>bpy.ops.action.clickselect()</code>
<i>Shift-SELECTMOUSE</i>	<code>bpy.ops.action.clickselect()</code>
<i>Shift-Alt-SELECTMOUSE</i>	<code>bpy.ops.action.clickselect()</code>
<i>Ctrl-Alt-SELECTMOUSE</i>	<code>bpy.ops.action.clickselect()</code>
<i>Ctrl-Shift-Alt-SELECTMOUSE</i>	<code>bpy.ops.action.clickselect()</code>
<i>Ctrl-SELECTMOUSE</i>	<code>bpy.ops.action.select_leftright()</code>
<i>Ctrl-Shift-SELECTMOUSE</i>	<code>bpy.ops.action.select_leftright()</code>
<i>LEFT_BRACKET</i>	<code>bpy.ops.action.select_leftright()</code>
<i>RIGHT_BRACKET</i>	<code>bpy.ops.action.select_leftright()</code>
<i>A</i>	<code>bpy.ops.action.select_all_toggle()</code>
<i>Ctrl-I</i>	<code>bpy.ops.action.select_all_toggle()</code>
<i>B</i>	<code>bpy.ops.action.select_border()</code>
<i>Alt-B</i>	<code>bpy.ops.action.select_border()</code>
<i>Ctrl-EVT_TWEAK_A</i>	<code>bpy.ops.action.select_lasso()</code>
<i>Ctrl-Shift-EVT_TWEAK_A</i>	<code>bpy.ops.action.select_lasso()</code>
<i>C</i>	<code>bpy.ops.action.select_circle()</code>
<i>K</i>	<code>bpy.ops.action.select_column()</code>
<i>Ctrl-K</i>	<code>bpy.ops.action.select_column()</code>
<i>Shift-K</i>	<code>bpy.ops.action.select_column()</code>
<i>Alt-K</i>	<code>bpy.ops.action.select_column()</code>
<i>Ctrl-NUMPAD_PLUS</i>	<code>bpy.ops.action.select_more()</code>
<i>Ctrl-NUMPAD_MINUS</i>	<code>bpy.ops.action.select_less()</code>
<i>L</i>	<code>bpy.ops.action.select_linked()</code>
<i>Ctrl-G</i>	<code>bpy.ops.action.frame_jump()</code>
<i>Shift-S</i>	<code>bpy.ops.action.snap()</code>
<i>Shift-M</i>	<code>bpy.ops.action.mirror()</code>
<i>V</i>	<code>bpy.ops.action.handle_type()</code>
<i>T</i>	<code>bpy.ops.action.interpolation_type()</code>
<i>Shift-E</i>	<code>bpy.ops.action.extrapolation_type()</code>
Continued on next page	

Table 2.6 – continued from previous page

Hotkey	Operator
<i>R</i>	<code>bpy.ops.action.keyframe_type()</code>
<i>Shift-O</i>	<code>bpy.ops.action.sample()</code>
<i>X</i>	<code>bpy.ops.wm.call_menu()</code>
<i>DEL</i>	<code>bpy.ops.wm.call_menu()</code>
<i>Shift-D</i>	<code>bpy.ops.action.duplicate_move()</code>
<i>I</i>	<code>bpy.ops.action.keyframe_insert()</code>
<i>Ctrl-C</i>	<code>bpy.ops.action.copy()</code>
<i>Ctrl-V</i>	<code>bpy.ops.action.paste()</code>
<i>Ctrl-Shift-V</i>	<code>bpy.ops.action.paste()</code>
<i>Ctrl-Alt-P</i>	<code>bpy.ops.action.previewrange_set()</code>
<i>HOME</i>	<code>bpy.ops.action.view_all()</code>
<i>NDOF_BUTTON_FIT</i>	<code>bpy.ops.action.view_all()</code>
<i>NUMPAD_PERIOD</i>	<code>bpy.ops.action.view_selected()</code>
<i>NUMPAD_0</i>	<code>bpy.ops.action.view_frame()</code>
<i>Tab</i>	<code>bpy.ops.anim.channels_editable_toggle()</code>
<i>Ctrl-F</i>	<code>bpy.ops.anim.channels_find()</code>
<i>G</i>	<code>bpy.ops.transform.transform()</code>
<i>EVT_TWEAK_S</i>	<code>bpy.ops.transform.transform()</code>
<i>E</i>	<code>bpy.ops.transform.transform()</code>
<i>S</i>	<code>bpy.ops.transform.transform()</code>
<i>Shift-T</i>	<code>bpy.ops.transform.transform()</code>
<i>O</i>	<code>bpy.ops.wm.context_toggle()</code>
<i>M</i>	<code>bpy.ops.marker.add()</code>
<i>Ctrl-M</i>	<code>bpy.ops.marker.rename()</code>

Detailed Reference

Note: Hotkeys marked with the “(default)” prefix are inherited from the default blender keymap

Ctrl-A → `action.select_all_toggle` : **KEYBOARD** → PRESS

Select All

`bpy.ops.action.select_all_toggle(invert=False)`

Properties:	Values:
Invert	False

(default) SELECTMOUSE → `action.clickselect` : **MOUSE** → PRESS

Mouse Select Keys

`bpy.ops.action.clickselect(extend=False, column=False, channel=False)`

Properties:	Values:
Extend Select	False
Column Select	False
Only Channel	False

(default) Alt-SELECTMOUSE → `action.clickselect` : **MOUSE** → PRESS

Mouse Select Keys

`bpy.ops.action.clickselect(extend=False, column=False, channel=False)`

Properties:	Values:
Extend Select	False
Column Select	True
Only Channel	False

(default) Shift-SELECTMOUSE → action.clickselect : **MOUSE** → PRESS

Mouse Select Keys

bpy.ops.action.clickselect(extend=False, column=False, channel=False)

Properties:	Values:
Extend Select	True
Column Select	False
Only Channel	False

(default) Shift-Alt-SELECTMOUSE → action.clickselect : **MOUSE** → PRESS

Mouse Select Keys

bpy.ops.action.clickselect(extend=False, column=False, channel=False)

Properties:	Values:
Extend Select	True
Column Select	True
Only Channel	False

(default) Ctrl-Alt-SELECTMOUSE → action.clickselect : **MOUSE** → PRESS

Mouse Select Keys

bpy.ops.action.clickselect(extend=False, column=False, channel=False)

Properties:	Values:
Extend Select	False
Column Select	False
Only Channel	True

(default) Ctrl-Shift-Alt-SELECTMOUSE → action.clickselect : **MOUSE** → PRESS

Mouse Select Keys

bpy.ops.action.clickselect(extend=False, column=False, channel=False)

Properties:	Values:
Extend Select	True
Column Select	False
Only Channel	True

(default) Ctrl-SELECTMOUSE → action.select_leftright : **MOUSE** → PRESS

Select Left/Right

bpy.ops.action.select_leftright(mode='CHECK', extend=False)

Properties:	Values:
Extend Select	False
Mode	CHECK

(default) Ctrl-Shift-SELECTMOUSE → action.select_leftright : **MOUSE** → PRESS

Select Left/Right

bpy.ops.action.select_leftright(mode='CHECK', extend=False)

Properties:	Values:
Extend Select	True
Mode	CHECK

(default) **LEFT_BRACKET** → action.select_leftright : **KEYBOARD** → PRESS

Select Left/Right

bpy.ops.action.select_leftright(mode='CHECK', extend=False)

Properties:	Values:
Extend Select	False
Mode	LEFT

(default) **RIGHT_BRACKET** → action.select_leftright : **KEYBOARD** → PRESS

Select Left/Right

bpy.ops.action.select_leftright(mode='CHECK', extend=False)

Properties:	Values:
Extend Select	False
Mode	RIGHT

(default) **A** → action.select_all_toggle : **KEYBOARD** → PRESS

Select All

bpy.ops.action.select_all_toggle(invert=False)

Properties:	Values:
Invert	False

(default) **Ctrl-I** → action.select_all_toggle : **KEYBOARD** → PRESS

Select All

bpy.ops.action.select_all_toggle(invert=False)

Properties:	Values:
Invert	True

(default) **B** → action.select_border : **KEYBOARD** → PRESS

Border Select

bpy.ops.action.select_border(gesture_mode=0, xmin=0, xmax=0, ymin=0, ymax=0, extend=True, axis_range=False)

Properties:	Values:
Axis Range	False

(default) **Alt-B** → action.select_border : **KEYBOARD** → PRESS

Border Select

bpy.ops.action.select_border(gesture_mode=0, xmin=0, xmax=0, ymin=0, ymax=0, extend=True, axis_range=False)

Properties:	Values:
Axis Range	True

(default) **Ctrl-EVT_TWEAK_A** → action.select_lasso : **TWEAK** → ANY

Lasso Select

bpy.ops.action.select_lasso(path=[], deselect=False, extend=True)

Properties:	Values:
Deselect	False

(default) **Ctrl-Shift-EVT_TWEAK_A** → action.select_lasso : **TWEAK** → ANY

Lasso Select

bpy.ops.action.select_lasso(path=[], deselect=False, extend=True)

Properties:	Values:
Deselect	True

(default) C → action.select_circle : **KEYBOARD** → PRESS

Circle Select

bpy.ops.action.select_circle(x=0, y=0, radius=1, gesture_mode=0)

(default) K → action.select_column : **KEYBOARD** → PRESS

Select All

bpy.ops.action.select_column(mode='KEYS')

Properties:	Values:
Mode	KEYS

(default) Ctrl-K → action.select_column : **KEYBOARD** → PRESS

Select All

bpy.ops.action.select_column(mode='KEYS')

Properties:	Values:
Mode	CFRA

(default) Shift-K → action.select_column : **KEYBOARD** → PRESS

Select All

bpy.ops.action.select_column(mode='KEYS')

Properties:	Values:
Mode	MARKERS_COLUMN

(default) Alt-K → action.select_column : **KEYBOARD** → PRESS

Select All

bpy.ops.action.select_column(mode='KEYS')

Properties:	Values:
Mode	MARKERS_BETWEEN

(default) Ctrl-NUMPAD_PLUS → action.select_more : **KEYBOARD** → PRESS

Select More

bpy.ops.action.select_more()

(default) Ctrl-NUMPAD_MINUS → action.select_less : **KEYBOARD** → PRESS

Select Less

bpy.ops.action.select_less()

(default) L → action.select_linked : **KEYBOARD** → PRESS

Select Linked

bpy.ops.action.select_linked()

(default) Ctrl-G → action.frame_jump : **KEYBOARD** → PRESS

Jump to Keyframes

bpy.ops.action.frame_jump()

(default) Shift-S → action.snap : **KEYBOARD** → PRESS

Snap Keys

bpy.ops.action.snap(type='CFRA')

(default) Shift-M → action.mirror : **KEYBOARD** → PRESS

Mirror Keys

```
bpy.ops.action.mirror(type='CFRA')
```

(default) V → action.handle_type : **KEYBOARD** → PRESS

Set Keyframe Handle Type

```
bpy.ops.action.handle_type(type='FREE')
```

(default) T → action.interpolation_type : **KEYBOARD** → PRESS

Set Keyframe Interpolation

```
bpy.ops.action.interpolation_type(type='CONSTANT')
```

(default) Shift-E → action.extrapolation_type : **KEYBOARD** → PRESS

Set Keyframe Extrapolation

```
bpy.ops.action.extrapolation_type(type='CONSTANT')
```

(default) R → action.keyframe_type : **KEYBOARD** → PRESS

Set Keyframe Type

```
bpy.ops.action.keyframe_type(type='KEYFRAME')
```

(default) Shift-O → action.sample : **KEYBOARD** → PRESS

Sample Keyframes

```
bpy.ops.action.sample()
```

(default) X → wm.call_menu : **KEYBOARD** → PRESS

Call Menu

```
bpy.ops.wm.call_menu(name='')
```

Properties:	Values:
Name	DOPE SHEET_MT_delete

(default) DEL → wm.call_menu : **KEYBOARD** → PRESS

Call Menu

```
bpy.ops.wm.call_menu(name='')
```

Properties:	Values:
Name	DOPE SHEET_MT_delete

(default) Shift-D → action.duplicate_move : **KEYBOARD** → PRESS

Duplicate

```
bpy.ops.action.duplicate_move(ACTION_OT_duplicate={}, TRANSFORM_OT_transform={"mode":'TRANSLATION',
"value":(0, 0, 0), "axis":(0, 0, 0), "constraint_axis":(False, False, False), "constraint_orientation":'GLOBAL',
"mirror":False, "proportional":'DISABLED', "proportional_edit_falloff":'SMOOTH', "proportional_size":1,
"snap":False, "snap_target":'CLOSEST', "snap_point":(0, 0, 0), "snap_align":False, "snap_normal":(0, 0, 0),
"gpencil_strokes":False, "release_confirm":False})
```

Properties:	Values:
Duplicate Keyframes	N/A
Transform	N/A

(default) I → action.keyframe_insert : **KEYBOARD** → PRESS

Insert Keyframes

```
bpy.ops.action.keyframe_insert(type='ALL')
```


(default) Ctrl-C → action.copy : **KEYBOARD** → PRESS

Copy Keyframes

bpy.ops.action.copy()

(default) Ctrl-V → action.paste : **KEYBOARD** → PRESS

Paste Keyframes

bpy.ops.action.paste(offset='START', merge='MIX', flipped=False)

(default) Ctrl-Shift-V → action.paste : **KEYBOARD** → PRESS

Paste Keyframes

bpy.ops.action.paste(offset='START', merge='MIX', flipped=False)

Properties:	Values:
Flipped	True

(default) Ctrl-Alt-P → action.previewrange_set : **KEYBOARD** → PRESS

Auto-Set Preview Range

bpy.ops.action.previewrange_set()

(default) HOME → action.view_all : **KEYBOARD** → PRESS

View All

bpy.ops.action.view_all()

(default) NDOF_BUTTON_FIT → action.view_all : **NDOF** → PRESS

View All

bpy.ops.action.view_all()

(default) NUMPAD_PERIOD → action.view_selected : **KEYBOARD** → PRESS

View Selected

bpy.ops.action.view_selected()

(default) NUMPAD_0 → action.view_frame : **KEYBOARD** → PRESS

View Frame

bpy.ops.action.view_frame()

(default) Tab → anim.channels_editable_toggle : **KEYBOARD** → PRESS

Toggle Channel Editability

bpy.ops.anim.channels_editable_toggle(mode='TOGGLE', type='PROTECT')

(default) Ctrl-F → anim.channels_find : **KEYBOARD** → PRESS

Find Channels

bpy.ops.anim.channels_find(query="Query")

(default) G → transform.transform : **KEYBOARD** → PRESS

Transform

bpy.ops.transform.transform(mode='TRANSLATION', value=(0, 0, 0, 0), axis=(0, 0, 0), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpencil_strokes=False, release_confirm=False)

Properties:	Values:
Mode	TIME_TRANSLATE

(default) EVT_TWEAK_S → transform.transform : **TWEAK** → ANY

Transform

```
bpy.ops.transform.transform(mode='TRANSLATION', value=(0, 0, 0, 0), axis=(0, 0, 0), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpen-  
cil_strokes=False, release_confirm=False)
```

Properties:	Values:
Mode	TIME_TRANSLATE

(default) E → transform.transform : **KEYBOARD** → PRESS

Transform

```
bpy.ops.transform.transform(mode='TRANSLATION', value=(0, 0, 0, 0), axis=(0, 0, 0), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpen-  
cil_strokes=False, release_confirm=False)
```

Properties:	Values:
Mode	TIME_EXTEND

(default) S → transform.transform : **KEYBOARD** → PRESS

Transform

```
bpy.ops.transform.transform(mode='TRANSLATION', value=(0, 0, 0, 0), axis=(0, 0, 0), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpen-  
cil_strokes=False, release_confirm=False)
```

Properties:	Values:
Mode	TIME_SCALE

(default) Shift-T → transform.transform : **KEYBOARD** → PRESS

Transform

```
bpy.ops.transform.transform(mode='TRANSLATION', value=(0, 0, 0, 0), axis=(0, 0, 0), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpen-  
cil_strokes=False, release_confirm=False)
```

Properties:	Values:
Mode	TIME_SLIDE

(default) O → wm.context_toggle : **KEYBOARD** → PRESS

Context Toggle

```
bpy.ops.wm.context_toggle(data_path='')
```

Properties:	Values:
Context Attributes	tool_settings.use_proportional_action

(default) M → marker.add : **KEYBOARD** → PRESS

Add Time Marker

```
bpy.ops.marker.add()
```

(default) **Ctrl-M** → marker.rename : **KEYBOARD** → PRESS

Rename Marker

```
bpy.ops.marker.rename(name="RenamedMarker")
```

Face Mask

Quick Reference

Hotkey	Operator
<i>Ctrl-A</i>	<code>bpy.ops.paint.face_select_all()</code>
<i>A</i>	<code>bpy.ops.paint.face_select_all()</code>
<i>Ctrl-I</i>	<code>bpy.ops.paint.face_select_all()</code>
<i>H</i>	<code>bpy.ops.paint.face_select_hide()</code>
<i>Shift-H</i>	<code>bpy.ops.paint.face_select_hide()</code>
<i>Alt-H</i>	<code>bpy.ops.paint.face_select_reveal()</code>
<i>Ctrl-L</i>	<code>bpy.ops.paint.face_select_linked()</code>
<i>L</i>	<code>bpy.ops.paint.face_select_linked_pick()</code>
<i>Shift-L</i>	<code>bpy.ops.paint.face_select_linked_pick()</code>

Detailed Reference

Note: Hotkeys marked with the “(default)” prefix are inherited from the default blender keymap

Ctrl-A → paint.face_select_all : **KEYBOARD** → PRESS

(De)select All

```
bpy.ops.paint.face_select_all(action='TOGGLE')
```

Properties:	Values:
Action	TOGGLE

(default) **A** → paint.face_select_all : **KEYBOARD** → PRESS

(De)select All

```
bpy.ops.paint.face_select_all(action='TOGGLE')
```

Properties:	Values:
Action	TOGGLE

(default) **Ctrl-I** → paint.face_select_all : **KEYBOARD** → PRESS

(De)select All

```
bpy.ops.paint.face_select_all(action='TOGGLE')
```

Properties:	Values:
Action	INVERT

(default) **H** → paint.face_select_hide : **KEYBOARD** → PRESS

Face Select Hide

```
bpy.ops.paint.face_select_hide(unselected=False)
```

Properties:	Values:
Unselected	False

(default) Shift-H → paint.face_select_hide : **KEYBOARD** → PRESS

Face Select Hide

bpy.ops.paint.face_select_hide(unselected=False)

Properties:	Values:
Unselected	True

(default) Alt-H → paint.face_select_reveal : **KEYBOARD** → PRESS

Face Select Reveal

bpy.ops.paint.face_select_reveal(unselected=False)

(default) Ctrl-L → paint.face_select_linked : **KEYBOARD** → PRESS

Select Linked

bpy.ops.paint.face_select_linked()

(default) L → paint.face_select_linked_pick : **KEYBOARD** → PRESS

Select Linked Pick

bpy.ops.paint.face_select_linked_pick(deselect=False)

Properties:	Values:
Deselect	False

(default) Shift-L → paint.face_select_linked_pick : **KEYBOARD** → PRESS

Select Linked Pick

bpy.ops.paint.face_select_linked_pick(deselect=False)

Properties:	Values:
Deselect	True

File Browser Main

Quick Reference

Hotkey	Operator
<i>Ctrl-A</i>	<code>bpy.ops.file.select_all_toggle()</code>
<i>LEFTMOUSE</i>	<code>bpy.ops.file.execute()</code>
<i>NUMPAD_PERIOD</i>	<code>bpy.ops.file.refresh()</code>
<i>LEFTMOUSE</i>	<code>bpy.ops.file.select()</code>
<i>Shift-LEFTMOUSE</i>	<code>bpy.ops.file.select()</code>
<i>Ctrl-Shift-LEFTMOUSE</i>	<code>bpy.ops.file.select()</code>
<i>RIGHTMOUSE</i>	<code>bpy.ops.file.select()</code>
<i>Shift-RIGHTMOUSE</i>	<code>bpy.ops.file.select()</code>
<i>Alt-RIGHTMOUSE</i>	<code>bpy.ops.file.select()</code>
<i>UP_ARROW</i>	<code>bpy.ops.file.select_walk()</code>
<i>Shift-UP_ARROW</i>	<code>bpy.ops.file.select_walk()</code>
<i>Ctrl-Shift-UP_ARROW</i>	<code>bpy.ops.file.select_walk()</code>
<i>DOWN_ARROW</i>	<code>bpy.ops.file.select_walk()</code>
<i>Shift-DOWN_ARROW</i>	<code>bpy.ops.file.select_walk()</code>
<i>Ctrl-Shift-DOWN_ARROW</i>	<code>bpy.ops.file.select_walk()</code>
<i>LEFT_ARROW</i>	<code>bpy.ops.file.select_walk()</code>
Continued on next page	

Table 2.7 – continued from previous page

Hotkey	Operator
<i>Shift-LEFT_ARROW</i>	<code>bpy.ops.file.select_walk()</code>
<i>Ctrl-Shift-LEFT_ARROW</i>	<code>bpy.ops.file.select_walk()</code>
<i>RIGHT_ARROW</i>	<code>bpy.ops.file.select_walk()</code>
<i>Shift-RIGHT_ARROW</i>	<code>bpy.ops.file.select_walk()</code>
<i>Ctrl-Shift-RIGHT_ARROW</i>	<code>bpy.ops.file.select_walk()</code>
<i>BUTTON4MOUSE</i>	<code>bpy.ops.file.previous()</code>
<i>BUTTON5MOUSE</i>	<code>bpy.ops.file.next()</code>
<i>A</i>	<code>bpy.ops.file.select_all_toggle()</code>
<i>B</i>	<code>bpy.ops.file.select_border()</code>
<i>EVT_TWEAK_L</i>	<code>bpy.ops.file.select_border()</code>
<i>Ctrl-LEFTMOUSE</i>	<code>bpy.ops.file.rename()</code>
<i>Any-MOUSEMOVE</i>	<code>bpy.ops.file.highlight()</code>
<i>NUMPAD_PLUS</i>	<code>bpy.ops.file.filenum()</code>
<i>Shift-NUMPAD_PLUS</i>	<code>bpy.ops.file.filenum()</code>
<i>Ctrl-NUMPAD_PLUS</i>	<code>bpy.ops.file.filenum()</code>
<i>NUMPAD_MINUS</i>	<code>bpy.ops.file.filenum()</code>
<i>Shift-NUMPAD_MINUS</i>	<code>bpy.ops.file.filenum()</code>
<i>Ctrl-NUMPAD_MINUS</i>	<code>bpy.ops.file.filenum()</code>

Detailed Reference

Note: Hotkeys marked with the “(default)” prefix are inherited from the default blender keymap

Ctrl-A → `file.select_all_toggle` : **KEYBOARD** → PRESS

(De)select All Files

`bpy.ops.file.select_all_toggle()`

(default) LEFTMOUSE → `file.execute` : **MOUSE** → DOUBLE_CLICK

Execute File Window

`bpy.ops.file.execute(need_active=False)`

Properties:	Values:
Need Active	True

(default) NUMPAD_PERIOD → `file.refresh` : **KEYBOARD** → PRESS

Refresh Filelist

`bpy.ops.file.refresh()`

(default) LEFTMOUSE → `file.select` : **MOUSE** → CLICK

Activate/Select File

`bpy.ops.file.select(extend=False, fill=False, open=True)`

(default) Shift-LEFTMOUSE → `file.select` : **MOUSE** → CLICK

Activate/Select File

`bpy.ops.file.select(extend=False, fill=False, open=True)`

Properties:	Values:
Extend	True

(default) Ctrl-Shift-LEFTMOUSE → file.select : **MOUSE** → CLICK

Activate/Select File

bpy.ops.file.select(extend=False, fill=False, open=True)

Properties:	Values:
Extend	True
Fill	True

(default) RIGHTMOUSE → file.select : **MOUSE** → CLICK

Activate/Select File

bpy.ops.file.select(extend=False, fill=False, open=True)

Properties:	Values:
Open	False

(default) Shift-RIGHTMOUSE → file.select : **MOUSE** → CLICK

Activate/Select File

bpy.ops.file.select(extend=False, fill=False, open=True)

Properties:	Values:
Extend	True
Open	False

(default) Alt-RIGHTMOUSE → file.select : **MOUSE** → CLICK

Activate/Select File

bpy.ops.file.select(extend=False, fill=False, open=True)

Properties:	Values:
Extend	True
Fill	True
Open	False

(default) UP_ARROW → file.select_walk : **KEYBOARD** → PRESS

Walk Select/Deselect File

bpy.ops.file.select_walk(direction='UP', extend=False, fill=False)

Properties:	Values:
Walk Direction	UP

(default) Shift-UP_ARROW → file.select_walk : **KEYBOARD** → PRESS

Walk Select/Deselect File

bpy.ops.file.select_walk(direction='UP', extend=False, fill=False)

Properties:	Values:
Walk Direction	UP
Extend	True

(default) Ctrl-Shift-UP_ARROW → file.select_walk : **KEYBOARD** → PRESS

Walk Select/Deselect File

bpy.ops.file.select_walk(direction='UP', extend=False, fill=False)

Properties:	Values:
Walk Direction	UP
Extend	True
Fill	True

(default) DOWN_ARROW → file.select_walk : **KEYBOARD** → PRESS

Walk Select/Deselect File

bpy.ops.file.select_walk(direction='UP', extend=False, fill=False)

Properties:	Values:
Walk Direction	DOWN

(default) Shift-DOWN_ARROW → file.select_walk : **KEYBOARD** → PRESS

Walk Select/Deselect File

bpy.ops.file.select_walk(direction='UP', extend=False, fill=False)

Properties:	Values:
Walk Direction	DOWN
Extend	True

(default) Ctrl-Shift-DOWN_ARROW → file.select_walk : **KEYBOARD** → PRESS

Walk Select/Deselect File

bpy.ops.file.select_walk(direction='UP', extend=False, fill=False)

Properties:	Values:
Walk Direction	DOWN
Extend	True
Fill	True

(default) LEFT_ARROW → file.select_walk : **KEYBOARD** → PRESS

Walk Select/Deselect File

bpy.ops.file.select_walk(direction='UP', extend=False, fill=False)

Properties:	Values:
Walk Direction	LEFT

(default) Shift-LEFT_ARROW → file.select_walk : **KEYBOARD** → PRESS

Walk Select/Deselect File

bpy.ops.file.select_walk(direction='UP', extend=False, fill=False)

Properties:	Values:
Walk Direction	LEFT
Extend	True

(default) Ctrl-Shift-LEFT_ARROW → file.select_walk : **KEYBOARD** → PRESS

Walk Select/Deselect File

bpy.ops.file.select_walk(direction='UP', extend=False, fill=False)

Properties:	Values:
Walk Direction	LEFT
Extend	True
Fill	True

(default) RIGHT_ARROW → file.select_walk : **KEYBOARD** → PRESS

Walk Select/Deselect File

bpy.ops.file.select_walk(direction='UP', extend=False, fill=False)

Properties:	Values:
Walk Direction	RIGHT

(default) Shift-RIGHT_ARROW → file.select_walk : **KEYBOARD** → PRESS

Walk Select/Deselect File

bpy.ops.file.select_walk(direction='UP', extend=False, fill=False)

Properties:	Values:
Walk Direction	RIGHT
Extend	True

(default) Ctrl-Shift-RIGHT_ARROW → file.select_walk : **KEYBOARD** → PRESS

Walk Select/Deselect File

bpy.ops.file.select_walk(direction='UP', extend=False, fill=False)

Properties:	Values:
Walk Direction	RIGHT
Extend	True
Fill	True

(default) BUTTON4MOUSE → file.previous : **MOUSE** → CLICK

Previous Folder

bpy.ops.file.previous()

(default) BUTTON5MOUSE → file.next : **MOUSE** → CLICK

Next Folder

bpy.ops.file.next()

(default) A → file.select_all_toggle : **KEYBOARD** → PRESS

(De)select All Files

bpy.ops.file.select_all_toggle()

(default) B → file.select_border : **KEYBOARD** → PRESS

Activate/Select File

bpy.ops.file.select_border(gesture_mode=0, xmin=0, xmax=0, ymin=0, ymax=0, extend=True)

(default) EVT_TWEAK_L → file.select_border : **TWEAK** → ANY

Activate/Select File

bpy.ops.file.select_border(gesture_mode=0, xmin=0, xmax=0, ymin=0, ymax=0, extend=True)

(default) Ctrl-LEFTMOUSE → file.rename : **MOUSE** → PRESS

Rename File or Directory

bpy.ops.file.rename()

(default) Any-MOUSEMOVE → file.highlight : **MOUSE** → ANY

Highlight File

bpy.ops.file.highlight()

(default) NUMPAD_PLUS → file.filenum : **KEYBOARD** → PRESS

Increment Number in Filename

bpy.ops.file.filenum(increment=1)

Properties:	Values:
Increment	1

(default) Shift-NUMPAD_PLUS → file.filenum : **KEYBOARD** → PRESS

Increment Number in Filename

bpy.ops.file.filenum(increment=1)

Properties:	Values:
Increment	10

(default) **Ctrl-NUMPAD_PLUS** → file.filename : **KEYBOARD** → PRESS

Increment Number in Filename

bpy.ops.file.filename(increment=1)

Properties:	Values:
Increment	100

(default) **NUMPAD_MINUS** → file.filename : **KEYBOARD** → PRESS

Increment Number in Filename

bpy.ops.file.filename(increment=1)

Properties:	Values:
Increment	-1

(default) **Shift-NUMPAD_MINUS** → file.filename : **KEYBOARD** → PRESS

Increment Number in Filename

bpy.ops.file.filename(increment=1)

Properties:	Values:
Increment	-10

(default) **Ctrl-NUMPAD_MINUS** → file.filename : **KEYBOARD** → PRESS

Increment Number in Filename

bpy.ops.file.filename(increment=1)

Properties:	Values:
Increment	-100

Graph Editor

Quick Reference

Hotkey	Operator
<i>Ctrl-ACTIONMOUSE</i>	<code>bpy.ops.graph.cursor_set()</code>
<i>Ctrl-A</i>	<code>bpy.ops.graph.select_all_toggle()</code>
<i>Alt-EVT_TWEAK_A</i>	<code>bpy.ops.graph.select_lasso()</code>
<i>Shift-Alt-EVT_TWEAK_A</i>	<code>bpy.ops.graph.select_lasso()</code>
<i>Alt-EVT_TWEAK_S</i>	<code>bpy.ops.graph.select_border()</code>
<i>Shift-Alt-EVT_TWEAK_S</i>	<code>bpy.ops.graph.select_border()</code>
<i>SELECTMOUSE</i>	<code>bpy.ops.graph.select_linked()</code>
<i>Ctrl-Alt-SELECTMOUSE</i>	<code>bpy.ops.graph.click_insert()</code>
<i>Ctrl-H</i>	<code>bpy.ops.wm.context_toggle()</code>
<i>ACTIONMOUSE</i>	<code>bpy.ops.graph.cursor_set()</code>
<i>SELECTMOUSE</i>	<code>bpy.ops.graph.clickselect()</code>
<i>Alt-SELECTMOUSE</i>	<code>bpy.ops.graph.clickselect()</code>
<i>Shift-SELECTMOUSE</i>	<code>bpy.ops.graph.clickselect()</code>
<i>Shift-Alt-SELECTMOUSE</i>	<code>bpy.ops.graph.clickselect()</code>
<i>Ctrl-Alt-SELECTMOUSE</i>	<code>bpy.ops.graph.clickselect()</code>
<i>Ctrl-Shift-Alt-SELECTMOUSE</i>	<code>bpy.ops.graph.clickselect()</code>
<i>Ctrl-SELECTMOUSE</i>	<code>bpy.ops.graph.select_leftright()</code>
<i>Ctrl-Shift-SELECTMOUSE</i>	<code>bpy.ops.graph.select_leftright()</code>

Continued on next page

Table 2.8 – continued from previous page

Hotkey	Operator
<i>LEFT_BRACKET</i>	<code>bpy.ops.graph.select_leftright()</code>
<i>RIGHT_BRACKET</i>	<code>bpy.ops.graph.select_leftright()</code>
<i>A</i>	<code>bpy.ops.graph.select_all_toggle()</code>
<i>Ctrl-I</i>	<code>bpy.ops.graph.select_all_toggle()</code>
<i>B</i>	<code>bpy.ops.graph.select_border()</code>
<i>Alt-B</i>	<code>bpy.ops.graph.select_border()</code>
<i>Ctrl-B</i>	<code>bpy.ops.graph.select_border()</code>
<i>Ctrl-Alt-B</i>	<code>bpy.ops.graph.select_border()</code>
<i>Ctrl-EVT_TWEAK_A</i>	<code>bpy.ops.graph.select_lasso()</code>
<i>Ctrl-Shift-EVT_TWEAK_A</i>	<code>bpy.ops.graph.select_lasso()</code>
<i>C</i>	<code>bpy.ops.graph.select_circle()</code>
<i>K</i>	<code>bpy.ops.graph.select_column()</code>
<i>Ctrl-K</i>	<code>bpy.ops.graph.select_column()</code>
<i>Shift-K</i>	<code>bpy.ops.graph.select_column()</code>
<i>Alt-K</i>	<code>bpy.ops.graph.select_column()</code>
<i>Ctrl-NUMPAD_PLUS</i>	<code>bpy.ops.graph.select_more()</code>
<i>Ctrl-NUMPAD_MINUS</i>	<code>bpy.ops.graph.select_less()</code>
<i>L</i>	<code>bpy.ops.graph.select_linked()</code>
<i>Ctrl-G</i>	<code>bpy.ops.graph.frame_jump()</code>
<i>Shift-S</i>	<code>bpy.ops.graph.snap()</code>
<i>Shift-M</i>	<code>bpy.ops.graph.mirror()</code>
<i>V</i>	<code>bpy.ops.graph.handle_type()</code>
<i>T</i>	<code>bpy.ops.graph.interpolation_type()</code>
<i>Ctrl-E</i>	<code>bpy.ops.graph.easing_type()</code>
<i>Alt-O</i>	<code>bpy.ops.graph.smooth()</code>
<i>Shift-O</i>	<code>bpy.ops.graph.sample()</code>
<i>Alt-C</i>	<code>bpy.ops.graph.bake()</code>
<i>X</i>	<code>bpy.ops.wm.call_menu()</code>
<i>DEL</i>	<code>bpy.ops.wm.call_menu()</code>
<i>Shift-D</i>	<code>bpy.ops.graph.duplicate_move()</code>
<i>I</i>	<code>bpy.ops.graph.keyframe_insert()</code>
<i>Ctrl-Shift-ACTIONMOUSE</i>	<code>bpy.ops.graph.click_insert()</code>
<i>Ctrl-C</i>	<code>bpy.ops.graph.copy()</code>
<i>Ctrl-V</i>	<code>bpy.ops.graph.paste()</code>
<i>Ctrl-Shift-V</i>	<code>bpy.ops.graph.paste()</code>
<i>Ctrl-Alt-P</i>	<code>bpy.ops.graph.previewrange_set()</code>
<i>HOME</i>	<code>bpy.ops.graph.view_all()</code>
<i>NDOF_BUTTON_FIT</i>	<code>bpy.ops.graph.view_all()</code>
<i>NUMPAD_PERIOD</i>	<code>bpy.ops.graph.view_selected()</code>
<i>NUMPAD_0</i>	<code>bpy.ops.graph.view_frame()</code>
<i>Ctrl-Shift-M</i>	<code>bpy.ops.graph.fmodifier_add()</code>
<i>Tab</i>	<code>bpy.ops.anim.channels_editable_toggle()</code>
<i>G</i>	<code>bpy.ops.transform.translate()</code>
<i>EVT_TWEAK_S</i>	<code>bpy.ops.transform.translate()</code>
<i>E</i>	<code>bpy.ops.transform.transform()</code>
<i>R</i>	<code>bpy.ops.transform.rotate()</code>
<i>S</i>	<code>bpy.ops.transform.resize()</code>
<i>O</i>	<code>bpy.ops.wm.context_toggle()</code>
<i>,</i>	<code>bpy.ops.wm.context_set_enum()</code>

Continued on next page

Table 2.8 – continued from previous page

Hotkey	Operator
.	<code>bpy.ops.wm.context_set_enum()</code>
<i>Ctrl</i> ..	<code>bpy.ops.wm.context_set_enum()</code>
<i>M</i>	<code>bpy.ops.marker.add()</code>
<i>Ctrl-M</i>	<code>bpy.ops.marker.rename()</code>

Detailed Reference

Note: Hotkeys marked with the “(default)” prefix are inherited from the default blender keymap

Ctrl-ACTIONMOUSE → `graph.cursor_set` : **MOUSE** → PRESS

Set Cursor

`bpy.ops.graph.cursor_set(frame=0, value=0)`

Ctrl-A → `graph.select_all_toggle` : **KEYBOARD** → PRESS

Select All

`bpy.ops.graph.select_all_toggle(invert=False)`

Properties:	Values:
Invert	False

Alt-EVT_TWEAK_A → `graph.select_lasso` : **TWEAK** → ANY

Lasso Select

`bpy.ops.graph.select_lasso(path=[], deselect=False, extend=True)`

Properties:	Values:
Deselect	False

Shift-Alt-EVT_TWEAK_A → `graph.select_lasso` : **TWEAK** → ANY

Lasso Select

`bpy.ops.graph.select_lasso(path=[], deselect=False, extend=True)`

Properties:	Values:
Deselect	True

Alt-EVT_TWEAK_S → `graph.select_border` : **TWEAK** → ANY

Border Select

`bpy.ops.graph.select_border(gesture_mode=0, xmin=0, xmax=0, ymin=0, ymax=0, extend=True, axis_range=False, include_handles=False)`

Properties:	Values:
Extend	False

Shift-Alt-EVT_TWEAK_S → `graph.select_border` : **TWEAK** → ANY

Border Select

`bpy.ops.graph.select_border(gesture_mode=0, xmin=0, xmax=0, ymin=0, ymax=0, extend=True, axis_range=False, include_handles=False)`

Properties:	Values:
Extend	True

SELECTMOUSE → graph.select_linked : **MOUSE** → DOUBLE_CLICK
Select Linked

bpy.ops.graph.select_linked()

Ctrl-Alt-SELECTMOUSE → graph.click_insert : **MOUSE** → CLICK
Click-Insert Keyframes

bpy.ops.graph.click_insert(frame=1, value=1, extend=False)

(default) Ctrl-H → wm.context_toggle : **KEYBOARD** → PRESS
Context Toggle

bpy.ops.wm.context_toggle(data_path='')

Properties:	Values:
Context Attributes	space_data.show_handles

(default) ACTIONMOUSE → graph.cursor_set : **MOUSE** → PRESS
Set Cursor

bpy.ops.graph.cursor_set(frame=0, value=0)

(default) SELECTMOUSE → graph.clickselect : **MOUSE** → PRESS
Mouse Select Keys

bpy.ops.graph.clickselect(extend=False, column=False, curves=False)

Properties:	Values:
Extend Select	False
Only Curves	False
Column Select	False

(default) Alt-SELECTMOUSE → graph.clickselect : **MOUSE** → PRESS
Mouse Select Keys

bpy.ops.graph.clickselect(extend=False, column=False, curves=False)

Properties:	Values:
Extend Select	False
Only Curves	False
Column Select	True

(default) Shift-SELECTMOUSE → graph.clickselect : **MOUSE** → PRESS
Mouse Select Keys

bpy.ops.graph.clickselect(extend=False, column=False, curves=False)

Properties:	Values:
Extend Select	True
Only Curves	False
Column Select	False

(default) Shift-Alt-SELECTMOUSE → graph.clickselect : **MOUSE** → PRESS
Mouse Select Keys

bpy.ops.graph.clickselect(extend=False, column=False, curves=False)

Properties:	Values:
Extend Select	True
Only Curves	False
Column Select	True

(default) Ctrl-Alt-SELECTMOUSE → graph.clickselect : **MOUSE** → PRESS

Mouse Select Keys

bpy.ops.graph.clickselect(extend=False, column=False, curves=False)

Properties:	Values:
Extend Select	False
Only Curves	True
Column Select	False

(default) Ctrl-Shift-Alt-SELECTMOUSE → graph.clickselect : **MOUSE** → PRESS

Mouse Select Keys

bpy.ops.graph.clickselect(extend=False, column=False, curves=False)

Properties:	Values:
Extend Select	True
Only Curves	True
Column Select	False

(default) Ctrl-SELECTMOUSE → graph.select_leftright : **MOUSE** → PRESS

Select Left/Right

bpy.ops.graph.select_leftright(mode='CHECK', extend=False)

Properties:	Values:
Extend Select	False
Mode	CHECK

(default) Ctrl-Shift-SELECTMOUSE → graph.select_leftright : **MOUSE** → PRESS

Select Left/Right

bpy.ops.graph.select_leftright(mode='CHECK', extend=False)

Properties:	Values:
Extend Select	True
Mode	CHECK

(default) LEFT_BRACKET → graph.select_leftright : **KEYBOARD** → PRESS

Select Left/Right

bpy.ops.graph.select_leftright(mode='CHECK', extend=False)

Properties:	Values:
Extend Select	False
Mode	LEFT

(default) RIGHT_BRACKET → graph.select_leftright : **KEYBOARD** → PRESS

Select Left/Right

bpy.ops.graph.select_leftright(mode='CHECK', extend=False)

Properties:	Values:
Extend Select	False
Mode	RIGHT

(default) A → graph.select_all_toggle : **KEYBOARD** → PRESS

Select All

bpy.ops.graph.select_all_toggle(invert=False)

Properties:	Values:
Invert	False

(default) **Ctrl-I** → graph.select_all_toggle : **KEYBOARD** → PRESS

Select All

bpy.ops.graph.select_all_toggle(invert=False)

Properties:	Values:
Invert	True

(default) **B** → graph.select_border : **KEYBOARD** → PRESS

Border Select

bpy.ops.graph.select_border(gesture_mode=0, xmin=0, xmax=0, ymin=0, ymax=0, extend=True, axis_range=False, include_handles=False)

Properties:	Values:
Axis Range	False
Include Handles	False

(default) **Alt-B** → graph.select_border : **KEYBOARD** → PRESS

Border Select

bpy.ops.graph.select_border(gesture_mode=0, xmin=0, xmax=0, ymin=0, ymax=0, extend=True, axis_range=False, include_handles=False)

Properties:	Values:
Axis Range	True
Include Handles	False

(default) **Ctrl-B** → graph.select_border : **KEYBOARD** → PRESS

Border Select

bpy.ops.graph.select_border(gesture_mode=0, xmin=0, xmax=0, ymin=0, ymax=0, extend=True, axis_range=False, include_handles=False)

Properties:	Values:
Axis Range	False
Include Handles	True

(default) **Ctrl-Alt-B** → graph.select_border : **KEYBOARD** → PRESS

Border Select

bpy.ops.graph.select_border(gesture_mode=0, xmin=0, xmax=0, ymin=0, ymax=0, extend=True, axis_range=False, include_handles=False)

Properties:	Values:
Axis Range	True
Include Handles	True

(default) **Ctrl-EVT_TWEAK_A** → graph.select_lasso : **TWEAK** → ANY

Lasso Select

bpy.ops.graph.select_lasso(path=[], deselect=False, extend=True)

Properties:	Values:
Deselect	False

(default) **Ctrl-Shift-EVT_TWEAK_A** → graph.select_lasso : **TWEAK** → ANY

Lasso Select

bpy.ops.graph.select_lasso(path=[], deselect=False, extend=True)

Properties:	Values:
Deselect	True

(default) C → graph.select_circle : **KEYBOARD** → PRESS

Circle Select

bpy.ops.graph.select_circle(x=0, y=0, radius=1, gesture_mode=0)

(default) K → graph.select_column : **KEYBOARD** → PRESS

Select All

bpy.ops.graph.select_column(mode='KEYS')

Properties:	Values:
Mode	KEYS

(default) Ctrl-K → graph.select_column : **KEYBOARD** → PRESS

Select All

bpy.ops.graph.select_column(mode='KEYS')

Properties:	Values:
Mode	CFRA

(default) Shift-K → graph.select_column : **KEYBOARD** → PRESS

Select All

bpy.ops.graph.select_column(mode='KEYS')

Properties:	Values:
Mode	MARKERS_COLUMN

(default) Alt-K → graph.select_column : **KEYBOARD** → PRESS

Select All

bpy.ops.graph.select_column(mode='KEYS')

Properties:	Values:
Mode	MARKERS_BETWEEN

(default) Ctrl-NUMPAD_PLUS → graph.select_more : **KEYBOARD** → PRESS

Select More

bpy.ops.graph.select_more()

(default) Ctrl-NUMPAD_MINUS → graph.select_less : **KEYBOARD** → PRESS

Select Less

bpy.ops.graph.select_less()

(default) L → graph.select_linked : **KEYBOARD** → PRESS

Select Linked

bpy.ops.graph.select_linked()

(default) Ctrl-G → graph.frame_jump : **KEYBOARD** → PRESS

Jump to Keyframes

bpy.ops.graph.frame_jump()

(default) Shift-S → graph.snap : **KEYBOARD** → PRESS

Snap Keys

bpy.ops.graph.snap(type='CFRA')

(default) Shift-M → graph.mirror : **KEYBOARD** → PRESS

Mirror Keys

bpy.ops.graph.mirror(type='CFRA')

(default) V → graph.handle_type : **KEYBOARD** → PRESS
Set Keyframe Handle Type

```
bpy.ops.graph.handle_type(type='FREE')
```

(default) T → graph.interpolation_type : **KEYBOARD** → PRESS
Set Keyframe Interpolation

```
bpy.ops.graph.interpolation_type(type='CONSTANT')
```

(default) Ctrl-E → graph.easing_type : **KEYBOARD** → PRESS
Set Keyframe Easing Type

```
bpy.ops.graph.easing_type(type='AUTO')
```

(default) Alt-O → graph.smooth : **KEYBOARD** → PRESS
Smooth Keys

```
bpy.ops.graph.smooth()
```

(default) Shift-O → graph.sample : **KEYBOARD** → PRESS
Sample Keyframes

```
bpy.ops.graph.sample()
```

(default) Alt-C → graph.bake : **KEYBOARD** → PRESS
Bake Curve

```
bpy.ops.graph.bake()
```

(default) X → wm.call_menu : **KEYBOARD** → PRESS
Call Menu

```
bpy.ops.wm.call_menu(name='')
```

Properties:	Values:
Name	GRAPH_MT_delete

(default) DEL → wm.call_menu : **KEYBOARD** → PRESS
Call Menu

```
bpy.ops.wm.call_menu(name='')
```

Properties:	Values:
Name	GRAPH_MT_delete

(default) Shift-D → graph.duplicate_move : **KEYBOARD** → PRESS
Duplicate

```
bpy.ops.graph.duplicate_move(GRAPH_OT_duplicate={"mode":'TRANSLATION'}, TRANS-  
FORM_OT_transform={"mode":'TRANSLATION', "value":(0, 0, 0, 0), "axis":(0, 0, 0), "con-  
straint_axis":(False, False, False), "constraint_orientation":'GLOBAL', "mirror":False, "propor-  
tional":'DISABLED', "proportional_edit_falloff":'SMOOTH', "proportional_size":1, "snap":False,  
"snap_target":'CLOSEST', "snap_point":(0, 0, 0), "snap_align":False, "snap_normal":(0, 0, 0), "gpen-  
cil_strokes":False, "release_confirm":False})
```

Properties:	Values:
Duplicate Keyframes	N/A
Transform	N/A

(default) I → graph.keyframe_insert : **KEYBOARD** → PRESS
Insert Keyframes

```
bpy.ops.graph.keyframe_insert(type='ALL')
```


(default) Ctrl-Shift-ACTIONMOUSE → graph.click_insert : **MOUSE** → CLICK

Click-Insert Keyframes

bpy.ops.graph.click_insert(frame=1, value=1, extend=False)

Properties:	Values:
Extend	True

(default) Ctrl-C → graph.copy : **KEYBOARD** → PRESS

Copy Keyframes

bpy.ops.graph.copy()

(default) Ctrl-V → graph.paste : **KEYBOARD** → PRESS

Paste Keyframes

bpy.ops.graph.paste(offset='START', merge='MIX', flipped=False)

(default) Ctrl-Shift-V → graph.paste : **KEYBOARD** → PRESS

Paste Keyframes

bpy.ops.graph.paste(offset='START', merge='MIX', flipped=False)

Properties:	Values:
Flipped	True

(default) Ctrl-Alt-P → graph.previewrange_set : **KEYBOARD** → PRESS

Auto-Set Preview Range

bpy.ops.graph.previewrange_set()

(default) HOME → graph.view_all : **KEYBOARD** → PRESS

View All

bpy.ops.graph.view_all(include_handles=True)

(default) NDOF_BUTTON_FIT → graph.view_all : **NDOF** → PRESS

View All

bpy.ops.graph.view_all(include_handles=True)

(default) NUMPAD_PERIOD → graph.view_selected : **KEYBOARD** → PRESS

View Selected

bpy.ops.graph.view_selected(include_handles=True)

(default) NUMPAD_0 → graph.view_frame : **KEYBOARD** → PRESS

View Frame

bpy.ops.graph.view_frame()

(default) Ctrl-Shift-M → graph.fmodifier_add : **KEYBOARD** → PRESS

Add F-Curve Modifier

bpy.ops.graph.fmodifier_add(type='NULL', only_active=True)

Properties:	Values:
Only Active	False

(default) Tab → anim.channels_editable_toggle : **KEYBOARD** → PRESS

Toggle Channel Editability

bpy.ops.anim.channels_editable_toggle(mode='TOGGLE', type='PROTECT')

(default) G → transform.translate : **KEYBOARD** → PRESS

Translate

```
bpy.ops.transform.translate(value=(0, 0, 0), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpencil_strokes=False, texture_space=False, remove_on_cancel=False, release_confirm=False)
```

(default) EVT_TWEAK_S → transform.translate : **TWEAK** → ANY

Translate

```
bpy.ops.transform.translate(value=(0, 0, 0), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpencil_strokes=False, texture_space=False, remove_on_cancel=False, release_confirm=False)
```

(default) E → transform.transform : **KEYBOARD** → PRESS

Transform

```
bpy.ops.transform.transform(mode='TRANSLATION', value=(0, 0, 0, 0), axis=(0, 0, 0), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpencil_strokes=False, release_confirm=False)
```

Properties:	Values:
Mode	TIME_EXTEND

(default) R → transform.rotate : **KEYBOARD** → PRESS

Rotate

```
bpy.ops.transform.rotate(value=0, axis=(0, 0, 0), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpencil_strokes=False, release_confirm=False)
```

(default) S → transform.resize : **KEYBOARD** → PRESS

Resize

```
bpy.ops.transform.resize(value=(1, 1, 1), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpencil_strokes=False, texture_space=False, remove_on_cancel=False, release_confirm=False)
```

(default) O → wm.context_toggle : **KEYBOARD** → PRESS

Context Toggle

```
bpy.ops.wm.context_toggle(data_path='')
```

Properties:	Values:
Context Attributes	tool_settings.use_proportional_fcurve

(default) , → wm.context_set_enum : **KEYBOARD** → PRESS

Context Set Enum

```
bpy.ops.wm.context_set_enum(data_path='', value='')
```

Properties:	Values:
Context Attributes	space_data.pivot_point
Value	BOUNDING_BOX_CENTER

(default) `.` → `wm.context_set_enum : KEYBOARD` → PRESS
Context Set Enum

`bpy.ops.wm.context_set_enum(data_path="", value="")`

Properties:	Values:
Context Attributes	space_data.pivot_point
Value	CURSOR

(default) `Ctrl-` → `wm.context_set_enum : KEYBOARD` → PRESS
Context Set Enum

`bpy.ops.wm.context_set_enum(data_path="", value="")`

Properties:	Values:
Context Attributes	space_data.pivot_point
Value	INDIVIDUAL_ORIGINS

(default) `M` → `marker.add : KEYBOARD` → PRESS
Add Time Marker

`bpy.ops.marker.add()`

(default) `Ctrl-M` → `marker.rename : KEYBOARD` → PRESS
Rename Marker

`bpy.ops.marker.rename(name="RenamedMarker")`

Image

Quick Reference

Hotkey	Operator
<i>Alt-F</i>	<code>bpy.ops.image.view_all()</code>
<i>ACTIONMOUSE</i>	<code>bpy.ops.image.view_pan()</code>
<i>SELECTMOUSE</i>	<code>bpy.ops.image.sample()</code>
<i>HOME</i>	<code>bpy.ops.image.view_all()</code>
<i>Shift-HOME</i>	<code>bpy.ops.image.view_all()</code>
<i>NUMPAD_PERIOD</i>	<code>bpy.ops.image.view_selected()</code>
<i>MIDDLEMOUSE</i>	<code>bpy.ops.image.view_pan()</code>
<i>Shift-MIDDLEMOUSE</i>	<code>bpy.ops.image.view_pan()</code>
<i>TRACKPADPAN</i>	<code>bpy.ops.image.view_pan()</code>
<i>NDOF_BUTTON_FIT</i>	<code>bpy.ops.image.view_all()</code>
<i>NDOF_MOTION</i>	<code>bpy.ops.image.view_ndof()</code>
<i>WHEELINMOUSE</i>	<code>bpy.ops.image.view_zoom_in()</code>
<i>WHEELOUTMOUSE</i>	<code>bpy.ops.image.view_zoom_out()</code>
<i>NUMPAD_PLUS</i>	<code>bpy.ops.image.view_zoom_in()</code>
<i>NUMPAD_MINUS</i>	<code>bpy.ops.image.view_zoom_out()</code>
<i>Ctrl-MIDDLEMOUSE</i>	<code>bpy.ops.image.view_zoom()</code>
<i>TRACKPADZOOM</i>	<code>bpy.ops.image.view_zoom()</code>
Continued on next page	

Table 2.9 – continued from previous page

Hotkey	Operator
<i>Ctrl-TRACKPADPAN</i>	<code>bpy.ops.image.view_zoom()</code>
<i>Shift-B</i>	<code>bpy.ops.image.view_zoom_border()</code>
<i>Ctrl-NUMPAD_8</i>	<code>bpy.ops.image.view_zoom_ratio()</code>
<i>Ctrl-NUMPAD_4</i>	<code>bpy.ops.image.view_zoom_ratio()</code>
<i>Ctrl-NUMPAD_2</i>	<code>bpy.ops.image.view_zoom_ratio()</code>
<i>Shift-NUMPAD_8</i>	<code>bpy.ops.image.view_zoom_ratio()</code>
<i>Shift-NUMPAD_4</i>	<code>bpy.ops.image.view_zoom_ratio()</code>
<i>Shift-NUMPAD_2</i>	<code>bpy.ops.image.view_zoom_ratio()</code>
<i>NUMPAD_1</i>	<code>bpy.ops.image.view_zoom_ratio()</code>
<i>NUMPAD_2</i>	<code>bpy.ops.image.view_zoom_ratio()</code>
<i>NUMPAD_4</i>	<code>bpy.ops.image.view_zoom_ratio()</code>
<i>NUMPAD_8</i>	<code>bpy.ops.image.view_zoom_ratio()</code>
<i>LEFTMOUSE</i>	<code>bpy.ops.image.change_frame()</code>
<i>ACTIONMOUSE</i>	<code>bpy.ops.image.sample()</code>
<i>Ctrl-ACTIONMOUSE</i>	<code>bpy.ops.image.curves_point_set()</code>
<i>Shift-ACTIONMOUSE</i>	<code>bpy.ops.image.curves_point_set()</code>
<i>Tab</i>	<code>bpy.ops.object.mode_set()</code>
<i>1</i>	<code>bpy.ops.wm.context_set_int()</code>
<i>2</i>	<code>bpy.ops.wm.context_set_int()</code>
<i>3</i>	<code>bpy.ops.wm.context_set_int()</code>
<i>4</i>	<code>bpy.ops.wm.context_set_int()</code>
<i>5</i>	<code>bpy.ops.wm.context_set_int()</code>
<i>6</i>	<code>bpy.ops.wm.context_set_int()</code>
<i>7</i>	<code>bpy.ops.wm.context_set_int()</code>
<i>8</i>	<code>bpy.ops.wm.context_set_int()</code>
<i>,</i>	<code>bpy.ops.wm.context_set_enum()</code>
<i>Ctrl-,</i>	<code>bpy.ops.wm.context_set_enum()</code>
<i>.</i>	<code>bpy.ops.wm.context_set_enum()</code>
<i>Ctrl-B</i>	<code>bpy.ops.image.render_border()</code>
<i>Ctrl-Alt-B</i>	<code>bpy.ops.image.clear_render_border()</code>

Detailed Reference

Note: Hotkeys marked with the “(default)” prefix are inherited from the default blender keymap

Alt-F → `image.view_all` : **KEYBOARD** → PRESS

View All

`bpy.ops.image.view_all(fit_view=False)`

Properties:	Values:
Fit View	True

ACTIONMOUSE → `image.view_pan` : **MOUSE** → PRESS

View Pan

`bpy.ops.image.view_pan(offset=(0, 0))`

SELECTMOUSE → `image.sample` : **MOUSE** → PRESS

Sample Color

`bpy.ops.image.sample()`

(default) HOME → image.view_all : **KEYBOARD** → PRESS

View All

bpy.ops.image.view_all(fit_view=False)

(default) Shift-HOME → image.view_all : **KEYBOARD** → PRESS

View All

bpy.ops.image.view_all(fit_view=False)

Properties:	Values:
Fit View	True

(default) NUMPAD_PERIOD → image.view_selected : **KEYBOARD** → PRESS

View Center

bpy.ops.image.view_selected()

(default) MIDDLEMOUSE → image.view_pan : **MOUSE** → PRESS

View Pan

bpy.ops.image.view_pan(offset=(0, 0))

(default) Shift-MIDDLEMOUSE → image.view_pan : **MOUSE** → PRESS

View Pan

bpy.ops.image.view_pan(offset=(0, 0))

(default) TRACKPADPAN → image.view_pan : **MOUSE** → ANY

View Pan

bpy.ops.image.view_pan(offset=(0, 0))

(default) NDOF_BUTTON_FIT → image.view_all : **NDOF** → PRESS

View All

bpy.ops.image.view_all(fit_view=False)

(default) NDOF_MOTION → image.view_ndof : **NDOF** → ANY

NDOF Pan/Zoom

bpy.ops.image.view_ndof()

(default) WHEELINMOUSE → image.view_zoom_in : **MOUSE** → PRESS

View Zoom In

bpy.ops.image.view_zoom_in(location=(0, 0))

(default) WHEELOUTMOUSE → image.view_zoom_out : **MOUSE** → PRESS

View Zoom Out

bpy.ops.image.view_zoom_out(location=(0, 0))

(default) NUMPAD_PLUS → image.view_zoom_in : **KEYBOARD** → PRESS

View Zoom In

bpy.ops.image.view_zoom_in(location=(0, 0))

(default) NUMPAD_MINUS → image.view_zoom_out : **KEYBOARD** → PRESS

View Zoom Out

bpy.ops.image.view_zoom_out(location=(0, 0))

(default) Ctrl1-MIDDLEMOUSE → image.view_zoom : **MOUSE** → PRESS

View Zoom

bpy.ops.image.view_zoom(factor=0)

(default) **TRACKPADZOOM** → image.view_zoom : **MOUSE** → ANY

View Zoom

bpy.ops.image.view_zoom(factor=0)

(default) **Ctrl-TRACKPADPAN** → image.view_zoom : **MOUSE** → ANY

View Zoom

bpy.ops.image.view_zoom(factor=0)

(default) **Shift-B** → image.view_zoom_border : **KEYBOARD** → PRESS

Zoom to Border

bpy.ops.image.view_zoom_border(gesture_mode=0, xmin=0, xmax=0, ymin=0, ymax=0)

(default) **Ctrl-NUMPAD_8** → image.view_zoom_ratio : **KEYBOARD** → PRESS

View Zoom Ratio

bpy.ops.image.view_zoom_ratio(ratio=0)

Properties:	Values:
Ratio	8.0

(default) **Ctrl-NUMPAD_4** → image.view_zoom_ratio : **KEYBOARD** → PRESS

View Zoom Ratio

bpy.ops.image.view_zoom_ratio(ratio=0)

Properties:	Values:
Ratio	4.0

(default) **Ctrl-NUMPAD_2** → image.view_zoom_ratio : **KEYBOARD** → PRESS

View Zoom Ratio

bpy.ops.image.view_zoom_ratio(ratio=0)

Properties:	Values:
Ratio	2.0

(default) **Shift-NUMPAD_8** → image.view_zoom_ratio : **KEYBOARD** → PRESS

View Zoom Ratio

bpy.ops.image.view_zoom_ratio(ratio=0)

Properties:	Values:
Ratio	8.0

(default) **Shift-NUMPAD_4** → image.view_zoom_ratio : **KEYBOARD** → PRESS

View Zoom Ratio

bpy.ops.image.view_zoom_ratio(ratio=0)

Properties:	Values:
Ratio	4.0

(default) **Shift-NUMPAD_2** → image.view_zoom_ratio : **KEYBOARD** → PRESS

View Zoom Ratio

bpy.ops.image.view_zoom_ratio(ratio=0)

Properties:	Values:
Ratio	2.0

(default) **NUMPAD_1** → image.view_zoom_ratio : **KEYBOARD** → PRESS

View Zoom Ratio

bpy.ops.image.view_zoom_ratio(ratio=0)

Properties:	Values:
Ratio	1.0

(default) NUMPAD_2 → image.view_zoom_ratio : **KEYBOARD** → PRESS
View Zoom Ratio

bpy.ops.image.view_zoom_ratio(ratio=0)

Properties:	Values:
Ratio	0.5

(default) NUMPAD_4 → image.view_zoom_ratio : **KEYBOARD** → PRESS
View Zoom Ratio

bpy.ops.image.view_zoom_ratio(ratio=0)

Properties:	Values:
Ratio	0.25

(default) NUMPAD_8 → image.view_zoom_ratio : **KEYBOARD** → PRESS
View Zoom Ratio

bpy.ops.image.view_zoom_ratio(ratio=0)

Properties:	Values:
Ratio	0.125

(default) LEFTMOUSE → image.change_frame : **MOUSE** → PRESS
Change Frame

bpy.ops.image.change_frame(frame=0)

(default) ACTIONMOUSE → image.sample : **MOUSE** → PRESS
Sample Color

bpy.ops.image.sample()

(default) Ctrl-ACTIONMOUSE → image.curves_point_set : **MOUSE** → PRESS
Set Curves Point

bpy.ops.image.curves_point_set(point='BLACK_POINT')

Properties:	Values:
Point	BLACK_POINT

(default) Shift-ACTIONMOUSE → image.curves_point_set : **MOUSE** → PRESS
Set Curves Point

bpy.ops.image.curves_point_set(point='BLACK_POINT')

Properties:	Values:
Point	WHITE_POINT

(default) Tab → object.mode_set : **KEYBOARD** → PRESS
Set Object Mode

bpy.ops.object.mode_set(mode='OBJECT', toggle=False)

Properties:	Values:
Mode	EDIT
Toggle	True

(default) 1 → wm.context_set_int : **KEYBOARD** → PRESS

Context Set

bpy.ops.wm.context_set_int(data_path="", value=0, relative=False)

Properties:	Values:
Context Attributes	space_data.image.render_slots.active_index
Value	0

(default) 2 → wm.context_set_int : **KEYBOARD** → PRESS

Context Set

bpy.ops.wm.context_set_int(data_path="", value=0, relative=False)

Properties:	Values:
Context Attributes	space_data.image.render_slots.active_index
Value	1

(default) 3 → wm.context_set_int : **KEYBOARD** → PRESS

Context Set

bpy.ops.wm.context_set_int(data_path="", value=0, relative=False)

Properties:	Values:
Context Attributes	space_data.image.render_slots.active_index
Value	2

(default) 4 → wm.context_set_int : **KEYBOARD** → PRESS

Context Set

bpy.ops.wm.context_set_int(data_path="", value=0, relative=False)

Properties:	Values:
Context Attributes	space_data.image.render_slots.active_index
Value	3

(default) 5 → wm.context_set_int : **KEYBOARD** → PRESS

Context Set

bpy.ops.wm.context_set_int(data_path="", value=0, relative=False)

Properties:	Values:
Context Attributes	space_data.image.render_slots.active_index
Value	4

(default) 6 → wm.context_set_int : **KEYBOARD** → PRESS

Context Set

bpy.ops.wm.context_set_int(data_path="", value=0, relative=False)

Properties:	Values:
Context Attributes	space_data.image.render_slots.active_index
Value	5

(default) 7 → wm.context_set_int : **KEYBOARD** → PRESS

Context Set

bpy.ops.wm.context_set_int(data_path="", value=0, relative=False)

Properties:	Values:
Context Attributes	space_data.image.render_slots.active_index
Value	6

(default) 8 → `wm.context_set_int : KEYBOARD` → PRESS

Context Set

`bpy.ops.wm.context_set_int(data_path="", value=0, relative=False)`

Properties:	Values:
Context Attributes	<code>space_data.image.render_slots.active_index</code>
Value	7

(default) , → `wm.context_set_enum : KEYBOARD` → PRESS

Context Set Enum

`bpy.ops.wm.context_set_enum(data_path="", value="")`

Properties:	Values:
Context Attributes	<code>space_data.pivot_point</code>
Value	CENTER

(default) Ctrl- → `wm.context_set_enum : KEYBOARD` → PRESS

Context Set Enum

`bpy.ops.wm.context_set_enum(data_path="", value="")`

Properties:	Values:
Context Attributes	<code>space_data.pivot_point</code>
Value	MEDIAN

(default) . → `wm.context_set_enum : KEYBOARD` → PRESS

Context Set Enum

`bpy.ops.wm.context_set_enum(data_path="", value="")`

Properties:	Values:
Context Attributes	<code>space_data.pivot_point</code>
Value	CURSOR

(default) Ctrl-B → `image.render_border : KEYBOARD` → PRESS

Render Border

`bpy.ops.image.render_border(xmin=0, xmax=0, ymin=0, ymax=0)`

(default) Ctrl-Alt-B → `image.clear_render_border : KEYBOARD` → PRESS

Clear Render Border

`bpy.ops.image.clear_render_border()`

Info

Quick Reference

Hotkey	Operator
<code>Ctrl-A</code>	<code>bpy.ops.info.select_all_toggle()</code>
<code>SELECTMOUSE</code>	<code>bpy.ops.info.select_pick()</code>
<code>A</code>	<code>bpy.ops.info.select_all_toggle()</code>
<code>B</code>	<code>bpy.ops.info.select_border()</code>
<code>R</code>	<code>bpy.ops.info.report_replay()</code>
<code>X</code>	<code>bpy.ops.info.report_delete()</code>
<code>DEL</code>	<code>bpy.ops.info.report_delete()</code>
<code>Ctrl-C</code>	<code>bpy.ops.info.report_copy()</code>

Detailed Reference

Note: Hotkeys marked with the “(default)” prefix are inherited from the default blender keymap

Ctrl-A → info.select_all_toggle : **KEYBOARD** → PRESS

(De)select All

bpy.ops.info.select_all_toggle()

(default) SELECTMOUSE → info.select_pick : **MOUSE** → PRESS

Select Report

bpy.ops.info.select_pick(report_index=0)

(default) A → info.select_all_toggle : **KEYBOARD** → PRESS

(De)select All

bpy.ops.info.select_all_toggle()

(default) B → info.select_border : **KEYBOARD** → PRESS

Border Select

bpy.ops.info.select_border(gesture_mode=0, xmin=0, xmax=0, ymin=0, ymax=0, extend=True)

(default) R → info.report_replay : **KEYBOARD** → PRESS

Replay Operators

bpy.ops.info.report_replay()

(default) X → info.report_delete : **KEYBOARD** → PRESS

Delete Reports

bpy.ops.info.report_delete()

(default) DEL → info.report_delete : **KEYBOARD** → PRESS

Delete Reports

bpy.ops.info.report_delete()

(default) Ctrl-C → info.report_copy : **KEYBOARD** → PRESS

Copy Reports to Clipboard

bpy.ops.info.report_copy()

Lattice

Quick Reference

Hotkey	Operator
<i>Ctrl-A</i>	<code>bpy.ops.lattice.select_all()</code>
<i>A</i>	<code>bpy.ops.lattice.select_all()</code>
<i>Ctrl-I</i>	<code>bpy.ops.lattice.select_all()</code>
<i>Ctrl-NUMPAD_PLUS</i>	<code>bpy.ops.lattice.select_more()</code>
<i>Ctrl-NUMPAD_MINUS</i>	<code>bpy.ops.lattice.select_less()</code>
<i>Ctrl-P</i>	<code>bpy.ops.object.vertex_parent_set()</code>
<i>Ctrl-F</i>	<code>bpy.ops.lattice.flip()</code>
<i>Ctrl-H</i>	<code>bpy.ops.wm.call_menu()</code>
<i>Shift-O</i>	<code>bpy.ops.wm.context_cycle_enum()</code>
<i>O</i>	<code>bpy.ops.wm.context_toggle_enum()</code>

Detailed Reference

Note: Hotkeys marked with the “(default)” prefix are inherited from the default blender keymap

Ctrl-A → `lattice.select_all` : **KEYBOARD** → PRESS

(De)select All

`bpy.ops.lattice.select_all(action='TOGGLE')`

Properties:	Values:
Action	TOGGLE

(default) A → `lattice.select_all` : **KEYBOARD** → PRESS

(De)select All

`bpy.ops.lattice.select_all(action='TOGGLE')`

Properties:	Values:
Action	TOGGLE

(default) Ctrl-I → `lattice.select_all` : **KEYBOARD** → PRESS

(De)select All

`bpy.ops.lattice.select_all(action='TOGGLE')`

Properties:	Values:
Action	INVERT

(default) Ctrl-NUMPAD_PLUS → `lattice.select_more` : **KEYBOARD** → PRESS

Select More

`bpy.ops.lattice.select_more()`

(default) Ctrl-NUMPAD_MINUS → `lattice.select_less` : **KEYBOARD** → PRESS

Select Less

`bpy.ops.lattice.select_less()`

(default) Ctrl-P → `object.vertex_parent_set` : **KEYBOARD** → PRESS

Make Vertex Parent

```
bpy.ops.object.vertex_parent_set()
```

(default) Ctrl-F → **lattice.flip : KEYBOARD** → PRESS

Flip (Distortion Free)

```
bpy.ops.lattice.flip(axis='U')
```

(default) Ctrl-H → **wm.call_menu : KEYBOARD** → PRESS

Call Menu

```
bpy.ops.wm.call_menu(name='')
```

Properties:	Values:
Name	VIEW3D_MT_hook

(default) Shift-O → **wm.context_cycle_enum : KEYBOARD** → PRESS

Context Enum Cycle

```
bpy.ops.wm.context_cycle_enum(data_path='', reverse=False, wrap=False)
```

Properties:	Values:
Context Attributes	tool_settings.proportional_edit_falloff
Wrap	True

(default) O → **wm.context_toggle_enum : KEYBOARD** → PRESS

Context Toggle Values

```
bpy.ops.wm.context_toggle_enum(data_path='', value_1='', value_2='')
```

Properties:	Values:
Context Attributes	tool_settings.proportional_edit
Value	DISABLED
Value	ENABLED

Markers

Quick Reference

Hotkey	Operator
<i>Alt-SELECTMOUSE</i>	<code>bpy.ops.marker.select()</code>
<i>Alt-EVT_TWEAK_S</i>	<code>bpy.ops.marker.move()</code>
<i>Ctrl-A</i>	<code>bpy.ops.marker.select_all()</code>
<i>M</i>	<code>bpy.ops.marker.add()</code>
<i>EVT_TWEAK_S</i>	<code>bpy.ops.marker.move()</code>
<i>Shift-D</i>	<code>bpy.ops.marker.duplicate()</code>
<i>SELECTMOUSE</i>	<code>bpy.ops.marker.select()</code>
<i>Shift-SELECTMOUSE</i>	<code>bpy.ops.marker.select()</code>
<i>Ctrl-SELECTMOUSE</i>	<code>bpy.ops.marker.select()</code>
<i>Ctrl-Shift-SELECTMOUSE</i>	<code>bpy.ops.marker.select()</code>
<i>B</i>	<code>bpy.ops.marker.select_border()</code>
<i>A</i>	<code>bpy.ops.marker.select_all()</code>
<i>X</i>	<code>bpy.ops.marker.delete()</code>
<i>DEL</i>	<code>bpy.ops.marker.delete()</code>
<i>Ctrl-M</i>	<code>bpy.ops.marker.rename()</code>
<i>G</i>	<code>bpy.ops.marker.move()</code>
<i>Ctrl-B</i>	<code>bpy.ops.marker.camera_bind()</code>

Detailed Reference

Note: Hotkeys marked with the “(default)” prefix are inherited from the default blender keymap

Alt-SELECTMOUSE → marker.select : **MOUSE** → PRESS

Select Time Marker

bpy.ops.marker.select(extend=False, camera=False)

Alt-EVT_TWEAK_S → marker.move : **TWEAK** → ANY

Move Time Marker

bpy.ops.marker.move(frames=0)

Ctrl-A → marker.select_all : **KEYBOARD** → PRESS

(De)select all Markers

bpy.ops.marker.select_all(action='TOGGLE')

(default) M → marker.add : **KEYBOARD** → PRESS

Add Time Marker

bpy.ops.marker.add()

(default) EVT_TWEAK_S → marker.move : **TWEAK** → ANY

Move Time Marker

bpy.ops.marker.move(frames=0)

(default) Shift-D → marker.duplicate : **KEYBOARD** → PRESS

Duplicate Time Marker

bpy.ops.marker.duplicate(frames=0)

(default) SELECTMOUSE → marker.select : **MOUSE** → PRESS

Select Time Marker

bpy.ops.marker.select(extend=False, camera=False)

(default) Shift-SELECTMOUSE → marker.select : **MOUSE** → PRESS

Select Time Marker

bpy.ops.marker.select(extend=False, camera=False)

Properties:	Values:
Extend	True

(default) Ctrl-SELECTMOUSE → marker.select : **MOUSE** → PRESS

Select Time Marker

bpy.ops.marker.select(extend=False, camera=False)

Properties:	Values:
Extend	False
Camera	True

(default) Ctrl-Shift-SELECTMOUSE → marker.select : **MOUSE** → PRESS

Select Time Marker

bpy.ops.marker.select(extend=False, camera=False)

Properties:	Values:
Extend	True
Camera	True

(default) B → marker.select_border : **KEYBOARD** → PRESS

Marker Border Select

bpy.ops.marker.select_border(gesture_mode=0, xmin=0, xmax=0, ymin=0, ymax=0, extend=True)

(default) A → marker.select_all : **KEYBOARD** → PRESS

(De)select all Markers

bpy.ops.marker.select_all(action='TOGGLE')

(default) X → marker.delete : **KEYBOARD** → PRESS

Delete Markers

bpy.ops.marker.delete()

(default) DEL → marker.delete : **KEYBOARD** → PRESS

Delete Markers

bpy.ops.marker.delete()

(default) Ctrl-M → marker.rename : **KEYBOARD** → PRESS

Rename Marker

bpy.ops.marker.rename(name="RenamedMarker")

(default) G → marker.move : **KEYBOARD** → PRESS

Move Time Marker

bpy.ops.marker.move(frames=0)

(default) Ctrl-B → marker.camera_bind : **KEYBOARD** → PRESS

Bind Camera to Markers

bpy.ops.marker.camera_bind()

Mask Editing

Quick Reference

Hotkey	Operator
<i>Ctrl-A</i>	<code>bpy.ops.mask.select_all()</code>
<i>Ctrl-Alt-SELECTMOUSE</i>	<code>bpy.ops.mask.add_vertex_slide()</code>
<i>Alt-EVT_TWEAK_A</i>	<code>bpy.ops.mask.select_lasso()</code>
<i>Shift-Alt-EVT_TWEAK_A</i>	<code>bpy.ops.mask.select_lasso()</code>
<i>Alt-EVT_TWEAK_S</i>	<code>bpy.ops.mask.select_border()</code>
<i>Shift-Alt-EVT_TWEAK_S</i>	<code>bpy.ops.mask.select_border()</code>
<i>SELECTMOUSE</i>	<code>bpy.ops.mask.slide_point()</code>
<i>Ctrl-Shift-SELECTMOUSE</i>	<code>bpy.ops.mask.add_feather_vertex_slide()</code>
<i>Ctrl-ACTIONMOUSE</i>	<code>bpy.ops.uv.cursor_set()</code>
<i>Alt-N</i>	<code>bpy.ops.mask.new()</code>
<i>Shift-A</i>	<code>bpy.ops.wm.call_menu()</code>
<i>Shift-O</i>	<code>bpy.ops.wm.context_cycle_enum()</code>
<i>O</i>	<code>bpy.ops.wm.context_toggle()</code>

Continued on next page

Table 2.10 – continued from previous page

Hotkey	Operator
<i>Ctrl</i> -ACTIONMOUSE	bpy.ops.mask.add_vertex_slide()
<i>Shift</i> -ACTIONMOUSE	bpy.ops.mask.add_feather_vertex_slide()
<i>X</i>	bpy.ops.mask.delete()
<i>DEL</i>	bpy.ops.mask.delete()
<i>SELECT</i> MOUSE	bpy.ops.mask.select()
<i>Shift</i> -SELECTMOUSE	bpy.ops.mask.select()
<i>A</i>	bpy.ops.mask.select_all()
<i>Ctrl</i> - <i>I</i>	bpy.ops.mask.select_all()
<i>Ctrl</i> - <i>L</i>	bpy.ops.mask.select_linked()
<i>L</i>	bpy.ops.mask.select_linked_pick()
<i>Shift</i> - <i>L</i>	bpy.ops.mask.select_linked_pick()
<i>B</i>	bpy.ops.mask.select_border()
<i>C</i>	bpy.ops.mask.select_circle()
<i>Ctrl</i> - <i>Alt</i> -EVT_TWEAK_A	bpy.ops.mask.select_lasso()
<i>Ctrl</i> - <i>Shift</i> - <i>Alt</i> -EVT_TWEAK_A	bpy.ops.mask.select_lasso()
<i>Ctrl</i> -NUMPAD_PLUS	bpy.ops.mask.select_more()
<i>Ctrl</i> -NUMPAD_MINUS	bpy.ops.mask.select_less()
<i>Alt</i> - <i>H</i>	bpy.ops.mask.hide_view_clear()
<i>H</i>	bpy.ops.mask.hide_view_set()
<i>Shift</i> - <i>H</i>	bpy.ops.mask.hide_view_set()
<i>Ctrl</i> -SELECTMOUSE	bpy.ops.clip.select()
<i>Alt</i> - <i>C</i>	bpy.ops.mask.cyclic_toggle()
ACTIONMOUSE	bpy.ops.mask.slide_spline_curvature()
<i>V</i>	bpy.ops.mask.handle_type_set()
<i>Ctrl</i> - <i>N</i>	bpy.ops.mask.normals_make_consistent()
<i>Ctrl</i> - <i>P</i>	bpy.ops.mask.parent_set()
<i>Alt</i> - <i>P</i>	bpy.ops.mask.parent_clear()
<i>I</i>	bpy.ops.mask.shape_key_insert()
<i>Alt</i> - <i>I</i>	bpy.ops.mask.shape_key_clear()
<i>Shift</i> - <i>D</i>	bpy.ops.mask.duplicate_move()
<i>Ctrl</i> - <i>C</i>	bpy.ops.mask.copy_splines()
<i>Ctrl</i> - <i>V</i>	bpy.ops.mask.paste_splines()
<i>G</i>	bpy.ops.transform.translate()
EVT_TWEAK_S	bpy.ops.transform.translate()
<i>S</i>	bpy.ops.transform.resize()
<i>R</i>	bpy.ops.transform.rotate()
<i>Alt</i> - <i>S</i>	bpy.ops.transform.transform()

Detailed Reference

Note: Hotkeys marked with the “(default)” prefix are inherited from the default blender keymap

Ctrl-A → mask.select_all : **KEYBOARD** → PRESS

(De)select All

bpy.ops.mask.select_all(action='TOGGLE')

Properties:	Values:
Action	TOGGLE

Ctrl-Alt-SELECTMOUSE → mask.add_vertex_slide : **MOUSE** → PRESS

Add Vertex and Slide

```
bpy.ops.mask.add_vertex_slide(MASK_OT_add_vertex={"location":(0, 0)},
MASK_OT_slide_point={"slide_feather":False, "is_new_point":False})
```

Properties:	Values:
Add Vertex	N/A
Slide Point	N/A

Alt-EVT_TWEAK_A → mask.select_lasso : **TWEAK** → ANY

Lasso Select

```
bpy.ops.mask.select_lasso(path=[], deselect=False, extend=True)
```

Properties:	Values:
Deselect	False

Shift-Alt-EVT_TWEAK_A → mask.select_lasso : **TWEAK** → ANY

Lasso Select

```
bpy.ops.mask.select_lasso(path=[], deselect=False, extend=True)
```

Properties:	Values:
Deselect	True

Alt-EVT_TWEAK_S → mask.select_border : **TWEAK** → ANY

Border Select

```
bpy.ops.mask.select_border(gesture_mode=0, xmin=0, xmax=0, ymin=0, ymax=0, extend=True)
```

Properties:	Values:
Extend	False

Shift-Alt-EVT_TWEAK_S → mask.select_border : **TWEAK** → ANY

Border Select

```
bpy.ops.mask.select_border(gesture_mode=0, xmin=0, xmax=0, ymin=0, ymax=0, extend=True)
```

Properties:	Values:
Extend	True

SELECTMOUSE → mask.slide_point : **MOUSE** → PRESS

Slide Point

```
bpy.ops.mask.slide_point(slide_feather=False, is_new_point=False)
```

Ctrl-Shift-SELECTMOUSE → mask.add_feather_vertex_slide : **MOUSE** → PRESS

Add Feather Vertex and Slide

```
bpy.ops.mask.add_feather_vertex_slide(MASK_OT_add_feather_vertex={"location":(0, 0)},
MASK_OT_slide_point={"slide_feather":False, "is_new_point":False})
```

Properties:	Values:
Add Feather Vertex	N/A
Slide Point	N/A

Ctrl-ACTIONMOUSE → uv.cursor_set : **MOUSE** → PRESS

Set 2D Cursor

```
bpy.ops.uv.cursor_set(location=(0, 0))
```

(default) Alt-N → mask.new : **KEYBOARD** → PRESS

New Mask


```
bpy.ops.mask.new(name='')
```

(default) Shift-A → **wm.call_menu : KEYBOARD** → PRESS

Call Menu

```
bpy.ops.wm.call_menu(name='')
```

Properties:	Values:
Name	MASK_MT_add

(default) Shift-O → **wm.context_cycle_enum : KEYBOARD** → PRESS

Context Enum Cycle

```
bpy.ops.wm.context_cycle_enum(data_path='', reverse=False, wrap=False)
```

Properties:	Values:
Context Attributes	tool_settings.proportional_edit_falloff
Wrap	True

(default) O → **wm.context_toggle : KEYBOARD** → PRESS

Context Toggle

```
bpy.ops.wm.context_toggle(data_path='')
```

Properties:	Values:
Context Attributes	tool_settings.use_proportional_edit_mask

(default) Ctrl-ACTIONMOUSE → **mask.add_vertex_slide : MOUSE** → PRESS

Add Vertex and Slide

```
bpy.ops.mask.add_vertex_slide(MASK_OT_add_vertex={"location":(0, 0)},
MASK_OT_slide_point={"slide_feather":False, "is_new_point":False})
```

Properties:	Values:
Add Vertex	N/A
Slide Point	N/A

(default) Shift-ACTIONMOUSE → **mask.add_feather_vertex_slide : MOUSE** → PRESS

Add Feather Vertex and Slide

```
bpy.ops.mask.add_feather_vertex_slide(MASK_OT_add_feather_vertex={"location":(0, 0)},
MASK_OT_slide_point={"slide_feather":False, "is_new_point":False})
```

Properties:	Values:
Add Feather Vertex	N/A
Slide Point	N/A

(default) X → **mask.delete : KEYBOARD** → PRESS

Delete

```
bpy.ops.mask.delete()
```

(default) DEL → **mask.delete : KEYBOARD** → PRESS

Delete

```
bpy.ops.mask.delete()
```

(default) SELECTMOUSE → **mask.select : MOUSE** → PRESS

Select

```
bpy.ops.mask.select(extend=False, deselect=False, toggle=False, location=(0, 0))
```

Properties:	Values:
Extend	False
Deselect	False
Toggle Selection	False

(default) Shift-SELECTMOUSE → mask.select : **MOUSE** → PRESS

Select

bpy.ops.mask.select(extend=False, deselect=False, toggle=False, location=(0, 0))

Properties:	Values:
Extend	False
Deselect	False
Toggle Selection	True

(default) A → mask.select_all : **KEYBOARD** → PRESS

(De)select All

bpy.ops.mask.select_all(action='TOGGLE')

Properties:	Values:
Action	TOGGLE

(default) Ctrl-I → mask.select_all : **KEYBOARD** → PRESS

(De)select All

bpy.ops.mask.select_all(action='TOGGLE')

Properties:	Values:
Action	INVERT

(default) Ctrl-L → mask.select_linked : **KEYBOARD** → PRESS

Select Linked All

bpy.ops.mask.select_linked()

(default) L → mask.select_linked_pick : **KEYBOARD** → PRESS

Select Linked

bpy.ops.mask.select_linked_pick(deselect=False)

Properties:	Values:
Deselect	False

(default) Shift-L → mask.select_linked_pick : **KEYBOARD** → PRESS

Select Linked

bpy.ops.mask.select_linked_pick(deselect=False)

Properties:	Values:
Deselect	True

(default) B → mask.select_border : **KEYBOARD** → PRESS

Border Select

bpy.ops.mask.select_border(gesture_mode=0, xmin=0, xmax=0, ymin=0, ymax=0, extend=True)

(default) C → mask.select_circle : **KEYBOARD** → PRESS

Circle Select

bpy.ops.mask.select_circle(x=0, y=0, radius=1, gesture_mode=0)

(default) Ctrl-Alt-EVT_TWEAK_A → mask.select_lasso : **TWEAK** → ANY

Lasso Select

bpy.ops.mask.select_lasso(path=[], deselect=False, extend=True)

Properties:	Values:
Deselect	False

(default) Ctrl-Shift-Alt-EVT_TWEAK_A → mask.select_lasso : **TWEAK** → ANY

Lasso Select

bpy.ops.mask.select_lasso(path=[], deselect=False, extend=True)

Properties:	Values:
Deselect	True

(default) Ctrl-NUMPAD_PLUS → mask.select_more : **KEYBOARD** → PRESS

Select More

bpy.ops.mask.select_more()

(default) Ctrl-NUMPAD_MINUS → mask.select_less : **KEYBOARD** → PRESS

Select Less

bpy.ops.mask.select_less()

(default) Alt-H → mask.hide_view_clear : **KEYBOARD** → PRESS

Clear Restrict View

bpy.ops.mask.hide_view_clear()

(default) H → mask.hide_view_set : **KEYBOARD** → PRESS

Set Restrict View

bpy.ops.mask.hide_view_set(unselected=False)

Properties:	Values:
Unselected	False

(default) Shift-H → mask.hide_view_set : **KEYBOARD** → PRESS

Set Restrict View

bpy.ops.mask.hide_view_set(unselected=False)

Properties:	Values:
Unselected	True

(default) Ctrl-SELECTMOUSE → clip.select : **MOUSE** → PRESS

Select

bpy.ops.clip.select(extend=False, location=(0, 0))

Properties:	Values:
Extend	False

(default) Alt-C → mask.cyclic_toggle : **KEYBOARD** → PRESS

Toggle Cyclic

bpy.ops.mask.cyclic_toggle()

(default) ACTIONMOUSE → mask.slide_spline_curvature : **MOUSE** → PRESS

Slide Spline Curvature

bpy.ops.mask.slide_spline_curvature()

(default) V → mask.handle_type_set : **KEYBOARD** → PRESS

Set Handle Type

bpy.ops.mask.handle_type_set(type='AUTO')

(default) **Ctrl-N** → mask.normals_make_consistent : **KEYBOARD** → PRESS

Recalc Normals

bpy.ops.mask.normals_make_consistent()

(default) **Ctrl-P** → mask.parent_set : **KEYBOARD** → PRESS

Make Parent

bpy.ops.mask.parent_set()

(default) **Alt-P** → mask.parent_clear : **KEYBOARD** → PRESS

Clear Parent

bpy.ops.mask.parent_clear()

(default) **I** → mask.shape_key_insert : **KEYBOARD** → PRESS

Insert Shape Key

bpy.ops.mask.shape_key_insert()

(default) **Alt-I** → mask.shape_key_clear : **KEYBOARD** → PRESS

Clear Shape Key

bpy.ops.mask.shape_key_clear()

(default) **Shift-D** → mask.duplicate_move : **KEYBOARD** → PRESS

Add Duplicate

bpy.ops.mask.duplicate_move(MASK_OT_duplicate={}, TRANSFORM_OT_translate={"value":(0, 0, 0), "constraint_axis":(False, False, False), "constraint_orientation":'GLOBAL', "mirror":False, "proportional":'DISABLED', "proportional_edit_falloff":'SMOOTH', "proportional_size":1, "snap":False, "snap_target":'CLOSEST', "snap_point":(0, 0, 0), "snap_align":False, "snap_normal":(0, 0, 0), "gpencil_strokes":False, "texture_space":False, "remove_on_cancel":False, "release_confirm":False})

Properties:	Values:
Duplicate Mask	N/A
Translate	N/A

(default) **Ctrl-C** → mask.copy_splines : **KEYBOARD** → PRESS

Copy Splines

bpy.ops.mask.copy_splines()

(default) **Ctrl-V** → mask.paste_splines : **KEYBOARD** → PRESS

Paste Splines

bpy.ops.mask.paste_splines()

(default) **G** → transform.translate : **KEYBOARD** → PRESS

Translate

bpy.ops.transform.translate(value=(0, 0, 0), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpencil_strokes=False, texture_space=False, remove_on_cancel=False, release_confirm=False)

(default) **EVT_TWEAK_S** → transform.translate : **TWEAK** → ANY

Translate

bpy.ops.transform.translate(value=(0, 0, 0), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0,

0, 0), snap_align=False, snap_normal=(0, 0, 0), gpencil_strokes=False, texture_space=False, remove_on_cancel=False, release_confirm=False)

(default) S → transform.resize : **KEYBOARD** → PRESS

Resize

```
bpy.ops.transform.resize(value=(1, 1, 1), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpencil_strokes=False, texture_space=False, remove_on_cancel=False, release_confirm=False)
```

(default) R → transform.rotate : **KEYBOARD** → PRESS

Rotate

```
bpy.ops.transform.rotate(value=0, axis=(0, 0, 0), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpencil_strokes=False, release_confirm=False)
```

(default) Alt-S → transform.transform : **KEYBOARD** → PRESS

Transform

```
bpy.ops.transform.transform(mode='TRANSLATION', value=(0, 0, 0, 0), axis=(0, 0, 0), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpencil_strokes=False, release_confirm=False)
```

Properties:	Values:
Mode	MASK_SHRINKFATTEN

Mesh

Quick Reference

Hotkey	Operator
<i>Ctrl-A</i>	<code>bpy.ops.mesh.select_all()</code>
<i>Ctrl-Alt-SELECTMOUSE</i>	<code>bpy.ops.mesh.dupli_extrude_cursor()</code>
<i>Ctrl-Shift-Alt-SELECTMOUSE</i>	<code>bpy.ops.mesh.dupli_extrude_cursor()</code>
<i>SELECTMOUSE</i>	<code>bpy.ops.mesh.select_linked()</code>
<i>Shift-SELECTMOUSE</i>	<code>bpy.ops.mesh.select_linked()</code>
<i>Ctrl-R</i>	<code>bpy.ops.mesh.loopcut_slide()</code>
<i>Ctrl-Shift-R</i>	<code>bpy.ops.mesh.offset_edge_loops_slide()</code>
<i>I</i>	<code>bpy.ops.mesh.inset()</code>
<i>Alt-P</i>	<code>bpy.ops.mesh.poke()</code>
<i>Ctrl-B</i>	<code>bpy.ops.mesh.bevel()</code>
<i>Ctrl-Shift-B</i>	<code>bpy.ops.mesh.bevel()</code>
<i>Alt-SELECTMOUSE</i>	<code>bpy.ops.mesh.loop_select()</code>
<i>Shift-Alt-SELECTMOUSE</i>	<code>bpy.ops.mesh.loop_select()</code>
<i>Ctrl-Alt-SELECTMOUSE</i>	<code>bpy.ops.mesh.edgering_select()</code>
<i>Ctrl-Shift-Alt-SELECTMOUSE</i>	<code>bpy.ops.mesh.edgering_select()</code>
<i>Ctrl-SELECTMOUSE</i>	<code>bpy.ops.mesh.shortest_path_pick()</code>

Continued on next page

Table 2.11 – continued from previous page

Hotkey	Operator
<i>Ctrl-Shift-SELECTMOUSE</i>	<code>bpy.ops.mesh.shortest_path_pick()</code>
<i>A</i>	<code>bpy.ops.mesh.select_all()</code>
<i>Ctrl-I</i>	<code>bpy.ops.mesh.select_all()</code>
<i>Ctrl-NUMPAD_PLUS</i>	<code>bpy.ops.mesh.select_more()</code>
<i>Ctrl-NUMPAD_MINUS</i>	<code>bpy.ops.mesh.select_less()</code>
<i>Ctrl-Shift-NUMPAD_PLUS</i>	<code>bpy.ops.mesh.select_next_item()</code>
<i>Ctrl-Shift-NUMPAD_MINUS</i>	<code>bpy.ops.mesh.select_prev_item()</code>
<i>Ctrl-Shift-Alt-M</i>	<code>bpy.ops.mesh.select_non_manifold()</code>
<i>Ctrl-L</i>	<code>bpy.ops.mesh.select_linked()</code>
<i>L</i>	<code>bpy.ops.mesh.select_linked_pick()</code>
<i>Shift-L</i>	<code>bpy.ops.mesh.select_linked_pick()</code>
<i>Ctrl-Shift-Alt-F</i>	<code>bpy.ops.mesh.faces_select_linked_flat()</code>
<i>Shift-G</i>	<code>bpy.ops.wm.call_menu()</code>
<i>Ctrl-Tab</i>	<code>bpy.ops.wm.call_menu()</code>
<i>H</i>	<code>bpy.ops.mesh.hide()</code>
<i>Shift-H</i>	<code>bpy.ops.mesh.hide()</code>
<i>Alt-H</i>	<code>bpy.ops.mesh.reveal()</code>
<i>Ctrl-N</i>	<code>bpy.ops.mesh.normals_make_consistent()</code>
<i>Ctrl-Shift-N</i>	<code>bpy.ops.mesh.normals_make_consistent()</code>
<i>E</i>	<code>bpy.ops.view3d.edit_mesh_extrude_move_normal()</code>
<i>Alt-E</i>	<code>bpy.ops.wm.call_menu()</code>
<i>Shift-E</i>	<code>bpy.ops.transform.edge_crease()</code>
<i>Alt-R</i>	<code>bpy.ops.mesh.spin()</code>
<i>Alt-F</i>	<code>bpy.ops.mesh.fill()</code>
<i>Shift-Alt-F</i>	<code>bpy.ops.mesh.beautify_fill()</code>
<i>Ctrl-T</i>	<code>bpy.ops.mesh.quads_convert_to_tris()</code>
<i>Ctrl-Shift-T</i>	<code>bpy.ops.mesh.quads_convert_to_tris()</code>
<i>Alt-J</i>	<code>bpy.ops.mesh.tris_convert_to_quads()</code>
<i>V</i>	<code>bpy.ops.mesh.rip_move()</code>
<i>Alt-V</i>	<code>bpy.ops.mesh.rip_move_fill()</code>
<i>Alt-D</i>	<code>bpy.ops.mesh.rip_edge_move()</code>
<i>Alt-M</i>	<code>bpy.ops.mesh.merge()</code>
<i>Alt-S</i>	<code>bpy.ops.transform.shrink_fatten()</code>
<i>F</i>	<code>bpy.ops.mesh.edge_face_add()</code>
<i>Shift-D</i>	<code>bpy.ops.mesh.duplicate_move()</code>
<i>Shift-A</i>	<code>bpy.ops.wm.call_menu()</code>
<i>P</i>	<code>bpy.ops.mesh.separate()</code>
<i>Y</i>	<code>bpy.ops.mesh.split()</code>
<i>J</i>	<code>bpy.ops.mesh.vert_connect_path()</code>
<i>Shift-V</i>	<code>bpy.ops.transform.vert_slide()</code>
<i>Ctrl-ACTIONMOUSE</i>	<code>bpy.ops.mesh.dupli_extrude_cursor()</code>
<i>Ctrl-Shift-ACTIONMOUSE</i>	<code>bpy.ops.mesh.dupli_extrude_cursor()</code>
<i>X</i>	<code>bpy.ops.wm.call_menu()</code>
<i>DEL</i>	<code>bpy.ops.wm.call_menu()</code>
<i>Ctrl-X</i>	<code>bpy.ops.mesh.dissolve_mode()</code>
<i>Ctrl-DEL</i>	<code>bpy.ops.mesh.dissolve_mode()</code>
<i>K</i>	<code>bpy.ops.mesh.knife_tool()</code>
<i>Shift-K</i>	<code>bpy.ops.mesh.knife_tool()</code>
<i>Ctrl-P</i>	<code>bpy.ops.object.vertex_parent_set()</code>

Continued on next page

Table 2.11 – continued from previous page

Hotkey	Operator
<i>W</i>	<code>bpy.ops.wm.call_menu()</code>
<i>Ctrl-F</i>	<code>bpy.ops.wm.call_menu()</code>
<i>Ctrl-E</i>	<code>bpy.ops.wm.call_menu()</code>
<i>Ctrl-V</i>	<code>bpy.ops.wm.call_menu()</code>
<i>Ctrl-H</i>	<code>bpy.ops.wm.call_menu()</code>
<i>U</i>	<code>bpy.ops.wm.call_menu()</code>
<i>Ctrl-G</i>	<code>bpy.ops.wm.call_menu()</code>
<i>Ctrl-0</i>	<code>bpy.ops.object.subdivision_set()</code>
<i>Ctrl-1</i>	<code>bpy.ops.object.subdivision_set()</code>
<i>Ctrl-2</i>	<code>bpy.ops.object.subdivision_set()</code>
<i>Ctrl-3</i>	<code>bpy.ops.object.subdivision_set()</code>
<i>Ctrl-4</i>	<code>bpy.ops.object.subdivision_set()</code>
<i>Ctrl-5</i>	<code>bpy.ops.object.subdivision_set()</code>
<i>Shift-O</i>	<code>bpy.ops.wm.context_cycle_enum()</code>
<i>O</i>	<code>bpy.ops.wm.context_toggle_enum()</code>
<i>Alt-O</i>	<code>bpy.ops.wm.context_toggle_enum()</code>

Detailed Reference

Note: Hotkeys marked with the “(default)” prefix are inherited from the default blender keymap

Ctrl-A → `mesh.select_all` : **KEYBOARD** → PRESS
(De)select All

`bpy.ops.mesh.select_all(action='TOGGLE')`

Properties:	Values:
Action	TOGGLE

Ctrl-Alt-SELECTMOUSE → `mesh.dupli_extrude_cursor` : **MOUSE** → PRESS
Duplicate or Extrude to Cursor

`bpy.ops.mesh.dupli_extrude_cursor(rotate_source=True)`

Properties:	Values:
Rotate Source	True

Ctrl-Shift-Alt-SELECTMOUSE → `mesh.dupli_extrude_cursor` : **MOUSE** → PRESS
Duplicate or Extrude to Cursor

`bpy.ops.mesh.dupli_extrude_cursor(rotate_source=True)`

Properties:	Values:
Rotate Source	False

SELECTMOUSE → `mesh.select_linked` : **MOUSE** → DOUBLE_CLICK
Select Linked All

`bpy.ops.mesh.select_linked(delimit={'SEAM'})`

Shift-SELECTMOUSE → `mesh.select_linked` : **MOUSE** → DOUBLE_CLICK
Select Linked All

`bpy.ops.mesh.select_linked(delimit={'SEAM'})`

(default) Ctrl-R → mesh.loopcut_slide : **KEYBOARD** → PRESS

Loop Cut and Slide

```
bpy.ops.mesh.loopcut_slide(MESH_OT_loopcut={"number_cuts":1, "smoothness":0,
"falloff":'INVERSE_SQUARE', "edge_index":-1, "mesh_select_mode_init":(False, False, False)}, TRANS-
FORM_OT_edge_slide={"value":0, "single_side":False, "use_even":False, "flipped":False, "use_clamp":True,
"mirror":False, "snap":False, "snap_target":'CLOSEST', "snap_point":(0, 0, 0), "snap_align":False,
"snap_normal":(0, 0, 0), "correct_uv":False, "release_confirm":False})
```

Properties:	Values:
Loop Cut	N/A
Edge Slide	N/A

(default) Ctrl-Shift-R → mesh.offset_edge_loops_slide : **KEYBOARD** → PRESS

Offset Edge Slide

```
bpy.ops.mesh.offset_edge_loops_slide(MESH_OT_offset_edge_loops={"use_cap_endpoint":False}, TRANS-
FORM_OT_edge_slide={"value":0, "single_side":False, "use_even":False, "flipped":False, "use_clamp":True,
"mirror":False, "snap":False, "snap_target":'CLOSEST', "snap_point":(0, 0, 0), "snap_align":False,
"snap_normal":(0, 0, 0), "correct_uv":False, "release_confirm":False})
```

Properties:	Values:
Offset Edge Loop	N/A
Edge Slide	N/A

(default) I → mesh.inset : **KEYBOARD** → PRESS

Inset Faces

```
bpy.ops.mesh.inset(use_boundary=True, use_even_offset=True, use_relative_offset=False,
use_edge_rail=False, thickness=0.01, depth=0, use_outset=False, use_select_inset=False,
use_individual=False, use_interpolate=True)
```

(default) Alt-P → mesh.poke : **KEYBOARD** → PRESS

Poke Faces

```
bpy.ops.mesh.poke(offset=0, use_relative_offset=False, center_mode='MEAN_WEIGHTED')
```

(default) Ctrl-B → mesh.bevel : **KEYBOARD** → PRESS

Bevel

```
bpy.ops.mesh.bevel(offset_type='OFFSET', offset=0, segments=1, profile=0.5, vertex_only=False,
clamp_overlap=False, loop_slide=True, material=-1)
```

Properties:	Values:
Vertex Only	False

(default) Ctrl-Shift-B → mesh.bevel : **KEYBOARD** → PRESS

Bevel

```
bpy.ops.mesh.bevel(offset_type='OFFSET', offset=0, segments=1, profile=0.5, vertex_only=False,
clamp_overlap=False, loop_slide=True, material=-1)
```

Properties:	Values:
Vertex Only	True

(default) Alt-SELECTMOUSE → mesh.loop_select : **MOUSE** → PRESS

Loop Select

```
bpy.ops.mesh.loop_select(extend=False, deselect=False, toggle=False, ring=False)
```


Properties:	Values:
Extend Select	False
Deselect	False
Toggle Select	False

(default) Shift-Alt-SELECTMOUSE → mesh.loop_select : **MOUSE** → PRESS

Loop Select

bpy.ops.mesh.loop_select(extend=False, deselect=False, toggle=False, ring=False)

Properties:	Values:
Extend Select	False
Deselect	False
Toggle Select	True

(default) Ctrl-Alt-SELECTMOUSE → mesh.edgering_select : **MOUSE** → PRESS

Edge Ring Select

bpy.ops.mesh.edgering_select(extend=False, deselect=False, toggle=False, ring=True)

Properties:	Values:
Extend	False
Deselect	False
Toggle Select	False

(default) Ctrl-Shift-Alt-SELECTMOUSE → mesh.edgering_select : **MOUSE** → PRESS

Edge Ring Select

bpy.ops.mesh.edgering_select(extend=False, deselect=False, toggle=False, ring=True)

Properties:	Values:
Extend	False
Deselect	False
Toggle Select	True

(default) Ctrl-SELECTMOUSE → mesh.shortest_path_pick : **MOUSE** → PRESS

Pick Shortest Path

bpy.ops.mesh.shortest_path_pick(use_face_step=False, use_topology_distance=False, use_fill=False, nth=1, skip=1, offset=0, index=-1)

Properties:	Values:
Fill Region	False

(default) Ctrl-Shift-SELECTMOUSE → mesh.shortest_path_pick : **MOUSE** → PRESS

Pick Shortest Path

bpy.ops.mesh.shortest_path_pick(use_face_step=False, use_topology_distance=False, use_fill=False, nth=1, skip=1, offset=0, index=-1)

Properties:	Values:
Fill Region	True

(default) A → mesh.select_all : **KEYBOARD** → PRESS

(De)select All

bpy.ops.mesh.select_all(action='TOGGLE')

Properties:	Values:
Action	TOGGLE

(default) Ctrl-I → mesh.select_all : **KEYBOARD** → PRESS

(De)select All

```
bpy.ops.mesh.select_all(action='TOGGLE')
```

Properties:	Values:
Action	INVERT

(default) Ctrl-NUMPAD_PLUS → mesh.select_more : **KEYBOARD** → PRESS
Select More

```
bpy.ops.mesh.select_more(use_face_step=True)
```

(default) Ctrl-NUMPAD_MINUS → mesh.select_less : **KEYBOARD** → PRESS
Select Less

```
bpy.ops.mesh.select_less(use_face_step=True)
```

(default) Ctrl-Shift-NUMPAD_PLUS → mesh.select_next_item : **KEYBOARD** → PRESS
Select Next Element

```
bpy.ops.mesh.select_next_item()
```

(default) Ctrl-Shift-NUMPAD_MINUS → mesh.select_prev_item : **KEYBOARD** → PRESS
Select Previous Element

```
bpy.ops.mesh.select_prev_item()
```

(default) Ctrl-Shift-Alt-M → mesh.select_non_manifold : **KEYBOARD** → PRESS
Select Non Manifold

```
bpy.ops.mesh.select_non_manifold(extend=True, use_wire=True, use_boundary=True, use_multi_face=True, use_non_contiguous=True, use_verts=True)
```

(default) Ctrl-L → mesh.select_linked : **KEYBOARD** → PRESS
Select Linked All

```
bpy.ops.mesh.select_linked(delimit={'SEAM'})
```

(default) L → mesh.select_linked_pick : **KEYBOARD** → PRESS
Select Linked

```
bpy.ops.mesh.select_linked_pick(deselect=False, delimit={'SEAM'}, index=-1)
```

Properties:	Values:
Deselect	False

(default) Shift-L → mesh.select_linked_pick : **KEYBOARD** → PRESS
Select Linked

```
bpy.ops.mesh.select_linked_pick(deselect=False, delimit={'SEAM'}, index=-1)
```

Properties:	Values:
Deselect	True

(default) Ctrl-Shift-Alt-F → mesh.faces_select_linked_flat : **KEYBOARD** → PRESS
Select Linked Flat Faces

```
bpy.ops.mesh.faces_select_linked_flat(sharpness=0.0174533)
```

(default) Shift-G → wm.call_menu : **KEYBOARD** → PRESS
Call Menu

```
bpy.ops.wm.call_menu(name='')
```

Properties:	Values:
Name	VIEW3D_MT_edit_mesh_select_similar

(default) Ctrl-Tab → wm.call_menu : **KEYBOARD** → PRESS

Call Menu

bpy.ops.wm.call_menu(name='')

Properties:	Values:
Name	VIEW3D_MT_edit_mesh_select_mode

(default) H → mesh.hide : **KEYBOARD** → PRESS

Hide Selection

bpy.ops.mesh.hide(unselected=False)

Properties:	Values:
Unselected	False

(default) Shift-H → mesh.hide : **KEYBOARD** → PRESS

Hide Selection

bpy.ops.mesh.hide(unselected=False)

Properties:	Values:
Unselected	True

(default) Alt-H → mesh.reveal : **KEYBOARD** → PRESS

Reveal Hidden

bpy.ops.mesh.reveal()

(default) Ctrl-N → mesh.normals_make_consistent : **KEYBOARD** → PRESS

Make Normals Consistent

bpy.ops.mesh.normals_make_consistent(inside=False)

Properties:	Values:
Inside	False

(default) Ctrl-Shift-N → mesh.normals_make_consistent : **KEYBOARD** → PRESS

Make Normals Consistent

bpy.ops.mesh.normals_make_consistent(inside=False)

Properties:	Values:
Inside	True

(default) E → view3d.edit_mesh_extrude_move_normal : **KEYBOARD** → PRESS

Extrude and Move on Normals

bpy.ops.view3d.edit_mesh_extrude_move_normal()

(default) Alt-E → wm.call_menu : **KEYBOARD** → PRESS

Call Menu

bpy.ops.wm.call_menu(name='')

Properties:	Values:
Name	VIEW3D_MT_edit_mesh_extrude

(default) Shift-E → transform.edge_crease : **KEYBOARD** → PRESS

Edge Crease

bpy.ops.transform.edge_crease(value=0, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), release_confirm=False)

(default) Alt-R → mesh.spin : **KEYBOARD** → PRESS

Spin

```
bpy.ops.mesh.spin(steps=9, dupli=False, angle=1.5708, center=(0, 0, 0), axis=(0, 0, 0))
```

(default) Alt-F → mesh.fill : **KEYBOARD** → PRESS

Fill

```
bpy.ops.mesh.fill(use_beauty=True)
```

(default) Shift-Alt-F → mesh.beautify_fill : **KEYBOARD** → PRESS

Beautify Faces

```
bpy.ops.mesh.beautify_fill(angle_limit=3.14159)
```

(default) Ctrl-T → mesh.quads_convert_to_tris : **KEYBOARD** → PRESS

Triangulate Faces

```
bpy.ops.mesh.quads_convert_to_tris(quad_method='BEAUTY', ngon_method='BEAUTY')
```

Properties:	Values:
Quad Method	BEAUTY
Polygon Method	BEAUTY

(default) Ctrl-Shift-T → mesh.quads_convert_to_tris : **KEYBOARD** → PRESS

Triangulate Faces

```
bpy.ops.mesh.quads_convert_to_tris(quad_method='BEAUTY', ngon_method='BEAUTY')
```

Properties:	Values:
Quad Method	FIXED
Polygon Method	CLIP

(default) Alt-J → mesh.tris_convert_to_quads : **KEYBOARD** → PRESS

Tris to Quads

```
bpy.ops.mesh.tris_convert_to_quads(face_threshold=0.698132, shape_threshold=0.698132, uvs=False, vcols=False, seam=False, sharp=False, materials=False)
```

(default) V → mesh.rip_move : **KEYBOARD** → PRESS

Rip

```
bpy.ops.mesh.rip_move(MESH_OT_rip={"mirror":False, "proportional":'DISABLED', "proportional_edit_falloff":'SMOOTH', "proportional_size":1, "release_confirm":False, "use_fill":False}, TRANSFORM_OT_translate={"value":(0, 0, 0), "constraint_axis":(False, False, False), "constraint_orientation":'GLOBAL', "mirror":False, "proportional":'DISABLED', "proportional_edit_falloff":'SMOOTH', "proportional_size":1, "snap":False, "snap_target":'CLOSEST', "snap_point":(0, 0, 0), "snap_align":False, "snap_normal":(0, 0, 0), "gpencil_strokes":False, "texture_space":False, "remove_on_cancel":False, "release_confirm":False})
```

Properties:	Values:
Rip	N/A
Translate	N/A

(default) Alt-V → mesh.rip_move_fill : **KEYBOARD** → PRESS

Rip Fill

```
bpy.ops.mesh.rip_move_fill(MESH_OT_rip={"mirror":False, "proportional":'DISABLED', "proportional_edit_falloff":'SMOOTH', "proportional_size":1, "release_confirm":False, "use_fill":False}, TRANSFORM_OT_translate={"value":(0, 0, 0), "constraint_axis":(False, False, False), "constraint_orientation":'GLOBAL', "mirror":False, "proportional":'DISABLED', "proportional_edit_falloff":'SMOOTH', "proportional_size":1, "snap":False, "snap_target":'CLOSEST',
```

“snap_point”:(0, 0, 0), “snap_align”:False, “snap_normal”:(0, 0, 0), “gpencil_strokes”:False, “texture_space”:False, “remove_on_cancel”:False, “release_confirm”:False})

Properties:	Values:
Rip	N/A
Translate	N/A

(default) Alt-D → mesh.rip_edge_move : **KEYBOARD** → PRESS

Extend Vertices

```
bpy.ops.mesh.rip_edge_move(MESH_OT_rip_edge={"mirror":False, "proportional":'DISABLED',
"proportional_edit_falloff":'SMOOTH', "proportional_size":1, "release_confirm":False},
TRANSFORM_OT_translate={"value":(0, 0, 0), "constraint_axis":(False, False, False),
"constraint_orientation":'GLOBAL', "mirror":False, "proportional":'DISABLED', "proportional_edit_falloff":'SMOOTH', "proportional_size":1, "snap":False, "snap_target":'CLOSEST',
"snap_point":(0, 0, 0), "snap_align":False, "snap_normal":(0, 0, 0), "gpencil_strokes":False, "texture_space":False, "remove_on_cancel":False, "release_confirm":False})
```

Properties:	Values:
Extend Vertices	N/A
Translate	N/A

(default) Alt-M → mesh.merge : **KEYBOARD** → PRESS

Merge

```
bpy.ops.mesh.merge(type='CENTER', uvs=False)
```

(default) Alt-S → transform.shrink_fatten : **KEYBOARD** → PRESS

Shrink/Fatten

```
bpy.ops.transform.shrink_fatten(value=0, use_even_offset=True, mirror=False, proportional='DISABLED',
proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST',
snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), release_confirm=False)
```

(default) F → mesh.edge_face_add : **KEYBOARD** → PRESS

Make Edge/Face

```
bpy.ops.mesh.edge_face_add()
```

(default) Shift-D → mesh.duplicate_move : **KEYBOARD** → PRESS

Add Duplicate

```
bpy.ops.mesh.duplicate_move(MESH_OT_duplicate={"mode":1}, TRANSFORM_OT_translate={"value":(0, 0, 0), "constraint_axis":(False, False, False), "constraint_orientation":'GLOBAL', "mirror":False, "proportional":'DISABLED', "proportional_edit_falloff":'SMOOTH', "proportional_size":1, "snap":False, "snap_target":'CLOSEST', "snap_point":(0, 0, 0), "snap_align":False, "snap_normal":(0, 0, 0), "gpencil_strokes":False, "texture_space":False, "remove_on_cancel":False, "release_confirm":False})
```

Properties:	Values:
Duplicate	N/A
Translate	N/A

(default) Shift-A → wm.call_menu : **KEYBOARD** → PRESS

Call Menu

```
bpy.ops.wm.call_menu(name='')
```

Properties:	Values:
Name	INFO_MT_mesh_add

(default) P → mesh.separate : **KEYBOARD** → PRESS

Separate

bpy.ops.mesh.separate(type='SELECTED')

(default) Y → mesh.split : **KEYBOARD** → PRESS
Split

bpy.ops.mesh.split()

(default) J → mesh.vert_connect_path : **KEYBOARD** → PRESS
Vertex Connect Path

bpy.ops.mesh.vert_connect_path()

(default) Shift-V → transform.vert_slide : **KEYBOARD** → PRESS
Vertex Slide

bpy.ops.transform.vert_slide(value=0, use_even=False, flipped=False, use_clamp=True, mirror=False, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), correct_uv=False, release_confirm=False)

(default) Ctrl-ACTIONMOUSE → mesh.dupli_extrude_cursor : **MOUSE** → CLICK
Duplicate or Extrude to Cursor

bpy.ops.mesh.dupli_extrude_cursor(rotate_source=True)

Properties:	Values:
Rotate Source	True

(default) Ctrl-Shift-ACTIONMOUSE → mesh.dupli_extrude_cursor : **MOUSE** → CLICK
Duplicate or Extrude to Cursor

bpy.ops.mesh.dupli_extrude_cursor(rotate_source=True)

Properties:	Values:
Rotate Source	False

(default) X → wm.call_menu : **KEYBOARD** → PRESS
Call Menu

bpy.ops.wm.call_menu(name='')

Properties:	Values:
Name	VIEW3D_MT_edit_mesh_delete

(default) DEL → wm.call_menu : **KEYBOARD** → PRESS
Call Menu

bpy.ops.wm.call_menu(name='')

Properties:	Values:
Name	VIEW3D_MT_edit_mesh_delete

(default) Ctrl-X → mesh.dissolve_mode : **KEYBOARD** → PRESS
Dissolve Selection

bpy.ops.mesh.dissolve_mode(use_verts=False, use_face_split=False, use_boundary_tear=False)

(default) Ctrl-DEL → mesh.dissolve_mode : **KEYBOARD** → PRESS
Dissolve Selection

bpy.ops.mesh.dissolve_mode(use_verts=False, use_face_split=False, use_boundary_tear=False)

(default) K → mesh.knife_tool : **KEYBOARD** → PRESS
Knife Topology Tool

bpy.ops.mesh.knife_tool(use_occlude_geometry=True, only_selected=False)

Properties:	Values:
Occlude Geometry	True
Only Selected	False

(default) Shift-K → mesh.knife_tool : **KEYBOARD** → PRESS

Knife Topology Tool

bpy.ops.mesh.knife_tool(use_occlude_geometry=True, only_selected=False)

Properties:	Values:
Occlude Geometry	False
Only Selected	True

(default) Ctrl-P → object.vertex_parent_set : **KEYBOARD** → PRESS

Make Vertex Parent

bpy.ops.object.vertex_parent_set()

(default) W → wm.call_menu : **KEYBOARD** → PRESS

Call Menu

bpy.ops.wm.call_menu(name='')

Properties:	Values:
Name	VIEW3D_MT_edit_mesh_specials

(default) Ctrl-F → wm.call_menu : **KEYBOARD** → PRESS

Call Menu

bpy.ops.wm.call_menu(name='')

Properties:	Values:
Name	VIEW3D_MT_edit_mesh_faces

(default) Ctrl-E → wm.call_menu : **KEYBOARD** → PRESS

Call Menu

bpy.ops.wm.call_menu(name='')

Properties:	Values:
Name	VIEW3D_MT_edit_mesh_edges

(default) Ctrl-V → wm.call_menu : **KEYBOARD** → PRESS

Call Menu

bpy.ops.wm.call_menu(name='')

Properties:	Values:
Name	VIEW3D_MT_edit_mesh_vertices

(default) Ctrl-H → wm.call_menu : **KEYBOARD** → PRESS

Call Menu

bpy.ops.wm.call_menu(name='')

Properties:	Values:
Name	VIEW3D_MT_hook

(default) U → wm.call_menu : **KEYBOARD** → PRESS

Call Menu

bpy.ops.wm.call_menu(name='')

Properties:	Values:
Name	VIEW3D_MT_uv_map

(default) Ctrl-G → wm.call_menu : **KEYBOARD** → PRESS

Call Menu

```
bpy.ops.wm.call_menu(name='')
```

Properties:	Values:
Name	VIEW3D_MT_vertex_group

(default) Ctrl-0 → object.subdivision_set : **KEYBOARD** → PRESS

Subdivision Set

```
bpy.ops.object.subdivision_set(level=1, relative=False)
```

Properties:	Values:
Level	0

(default) Ctrl-1 → object.subdivision_set : **KEYBOARD** → PRESS

Subdivision Set

```
bpy.ops.object.subdivision_set(level=1, relative=False)
```

Properties:	Values:
Level	1

(default) Ctrl-2 → object.subdivision_set : **KEYBOARD** → PRESS

Subdivision Set

```
bpy.ops.object.subdivision_set(level=1, relative=False)
```

Properties:	Values:
Level	2

(default) Ctrl-3 → object.subdivision_set : **KEYBOARD** → PRESS

Subdivision Set

```
bpy.ops.object.subdivision_set(level=1, relative=False)
```

Properties:	Values:
Level	3

(default) Ctrl-4 → object.subdivision_set : **KEYBOARD** → PRESS

Subdivision Set

```
bpy.ops.object.subdivision_set(level=1, relative=False)
```

Properties:	Values:
Level	4

(default) Ctrl-5 → object.subdivision_set : **KEYBOARD** → PRESS

Subdivision Set

```
bpy.ops.object.subdivision_set(level=1, relative=False)
```

Properties:	Values:
Level	5

(default) Shift-O → wm.context_cycle_enum : **KEYBOARD** → PRESS

Context Enum Cycle

```
bpy.ops.wm.context_cycle_enum(data_path='', reverse=False, wrap=False)
```

Properties:	Values:
Context Attributes	tool_settings.proportional_edit_falloff
Wrap	True

(default) O → `wm.context_toggle_enum` : **KEYBOARD** → PRESS

Context Toggle Values

`bpy.ops.wm.context_toggle_enum(data_path="", value_1="", value_2="")`

Properties:	Values:
Context Attributes	<code>tool_settings.proportional_edit</code>
Value	DISABLED
Value	ENABLED

(default) Alt-O → `wm.context_toggle_enum` : **KEYBOARD** → PRESS

Context Toggle Values

`bpy.ops.wm.context_toggle_enum(data_path="", value_1="", value_2="")`

Properties:	Values:
Context Attributes	<code>tool_settings.proportional_edit</code>
Value	DISABLED
Value	CONNECTED

Metaball

Quick Reference

Hotkey	Operator
<i>Ctrl-A</i>	<code>bpy.ops.mball.select_all()</code>
<i>Shift-A</i>	<code>bpy.ops.object.metaball_add()</code>
<i>Alt-H</i>	<code>bpy.ops.mball.reveal_metaelems()</code>
<i>H</i>	<code>bpy.ops.mball.hide_metaelems()</code>
<i>Shift-H</i>	<code>bpy.ops.mball.hide_metaelems()</code>
<i>X</i>	<code>bpy.ops.mball.delete_metaelems()</code>
<i>DEL</i>	<code>bpy.ops.mball.delete_metaelems()</code>
<i>Shift-D</i>	<code>bpy.ops.mball.duplicate_move()</code>
<i>A</i>	<code>bpy.ops.mball.select_all()</code>
<i>Ctrl-I</i>	<code>bpy.ops.mball.select_all()</code>
<i>Shift-G</i>	<code>bpy.ops.mball.select_similar()</code>
<i>Shift-O</i>	<code>bpy.ops.wm.context_cycle_enum()</code>
<i>O</i>	<code>bpy.ops.wm.context_toggle_enum()</code>
<i>Alt-O</i>	<code>bpy.ops.wm.context_toggle_enum()</code>

Detailed Reference

Note: Hotkeys marked with the “(default)” prefix are inherited from the default blender keymap

Ctrl-A → `mball.select_all` : **KEYBOARD** → PRESS

(De)select All

`bpy.ops.mball.select_all(action='TOGGLE')`

Properties:	Values:
Action	TOGGLE

Properties:	Values:
Action	INVERT

(default) Shift-G → `mball.select_similar` : **KEYBOARD** → PRESS

Select Similar

`bpy.ops.mball.select_similar(type='TYPE', threshold=0.1)`

(default) Shift-O → `wm.context_cycle_enum` : **KEYBOARD** → PRESS

Context Enum Cycle

`bpy.ops.wm.context_cycle_enum(data_path='', reverse=False, wrap=False)`

Properties:	Values:
Context Attributes	<code>tool_settings.proportional_edit_falloff</code>
Wrap	True

(default) O → `wm.context_toggle_enum` : **KEYBOARD** → PRESS

Context Toggle Values

`bpy.ops.wm.context_toggle_enum(data_path='', value_1='', value_2='')`

Properties:	Values:
Context Attributes	<code>tool_settings.proportional_edit</code>
Value	DISABLED
Value	ENABLED

(default) Alt-O → `wm.context_toggle_enum` : **KEYBOARD** → PRESS

Context Toggle Values

`bpy.ops.wm.context_toggle_enum(data_path='', value_1='', value_2='')`

Properties:	Values:
Context Attributes	<code>tool_settings.proportional_edit</code>
Value	DISABLED
Value	CONNECTED

NLA Editor

Quick Reference

Hotkey	Operator
<i>Ctrl-A</i>	<code>bpy.ops.nla.select_all_toggle()</code>
<i>A</i>	<code>bpy.ops.nla.apply_scale()</code>
<i>SELECTMOUSE</i>	<code>bpy.ops.nla.click_select()</code>
<i>Shift-SELECTMOUSE</i>	<code>bpy.ops.nla.click_select()</code>
<i>Ctrl-SELECTMOUSE</i>	<code>bpy.ops.nla.select_leftright()</code>
<i>Ctrl-Shift-SELECTMOUSE</i>	<code>bpy.ops.nla.select_leftright()</code>
<i>LEFT_BRACKET</i>	<code>bpy.ops.nla.select_leftright()</code>
<i>RIGHT_BRACKET</i>	<code>bpy.ops.nla.select_leftright()</code>
<i>A</i>	<code>bpy.ops.nla.select_all_toggle()</code>
<i>Ctrl-I</i>	<code>bpy.ops.nla.select_all_toggle()</code>
<i>B</i>	<code>bpy.ops.nla.select_border()</code>
<i>Alt-B</i>	<code>bpy.ops.nla.select_border()</code>
<i>Ctrl-Alt-P</i>	<code>bpy.ops.nla.previewrange_set()</code>

Continued on next page

Table 2.12 – continued from previous page

Hotkey	Operator
<i>HOME</i>	<code>bpy.ops.nla.view_all()</code>
<i>NDOF_BUTTON_FIT</i>	<code>bpy.ops.nla.view_all()</code>
<i>NUMPAD_PERIOD</i>	<code>bpy.ops.nla.view_selected()</code>
<i>NUMPAD_0</i>	<code>bpy.ops.nla.view_frame()</code>
<i>Shift-A</i>	<code>bpy.ops.nla.actionclip_add()</code>
<i>Shift-T</i>	<code>bpy.ops.nla.transition_add()</code>
<i>Shift-K</i>	<code>bpy.ops.nla.soundclip_add()</code>
<i>Shift-G</i>	<code>bpy.ops.nla.meta_add()</code>
<i>Alt-G</i>	<code>bpy.ops.nla.meta_remove()</code>
<i>Shift-D</i>	<code>bpy.ops.nla.duplicate()</code>
<i>Alt-D</i>	<code>bpy.ops.nla.duplicate()</code>
<i>U</i>	<code>bpy.ops.nla.make_single_user()</code>
<i>X</i>	<code>bpy.ops.nla.delete()</code>
<i>DEL</i>	<code>bpy.ops.nla.delete()</code>
<i>Y</i>	<code>bpy.ops.nla.split()</code>
<i>H</i>	<code>bpy.ops.nla.mute_toggle()</code>
<i>Alt-F</i>	<code>bpy.ops.nla.swap()</code>
<i>PAGE_UP</i>	<code>bpy.ops.nla.move_up()</code>
<i>PAGE_DOWN</i>	<code>bpy.ops.nla.move_down()</code>
<i>Ctrl-A</i>	<code>bpy.ops.nla.apply_scale()</code>
<i>Alt-S</i>	<code>bpy.ops.nla.clear_scale()</code>
<i>Shift-S</i>	<code>bpy.ops.nla.snap()</code>
<i>Ctrl-Shift-M</i>	<code>bpy.ops.nla.fmodifier_add()</code>
<i>G</i>	<code>bpy.ops.transform.transform()</code>
<i>EVT_TWEAK_S</i>	<code>bpy.ops.transform.transform()</code>
<i>E</i>	<code>bpy.ops.transform.transform()</code>
<i>S</i>	<code>bpy.ops.transform.transform()</code>
<i>M</i>	<code>bpy.ops.marker.add()</code>
<i>Ctrl-M</i>	<code>bpy.ops.marker.rename()</code>

Detailed Reference

Note: Hotkeys marked with the “(default)” prefix are inherited from the default blender keymap

Ctrl-A → `nla.select_all_toggle` : **KEYBOARD** → PRESS

(De)select All

`bpy.ops.nla.select_all_toggle(invert=False)`

Properties:	Values:
Invert	False

A → `nla.apply_scale` : **KEYBOARD** → PRESS

Apply Scale

`bpy.ops.nla.apply_scale()`

(default) SELECTMOUSE → `nla.click_select` : **MOUSE** → PRESS

Mouse Select

`bpy.ops.nla.click_select(extend=False)`

Properties:	Values:
Extend Select	False

(default) Shift-SELECTMOUSE → nla.click_select : **MOUSE** → PRESS

Mouse Select

bpy.ops.nla.click_select(extend=False)

Properties:	Values:
Extend Select	True

(default) Ctrl-SELECTMOUSE → nla.select_leftright : **MOUSE** → PRESS

Select Left/Right

bpy.ops.nla.select_leftright(mode='CHECK', extend=False)

Properties:	Values:
Extend Select	False
Mode	CHECK

(default) Ctrl-Shift-SELECTMOUSE → nla.select_leftright : **MOUSE** → PRESS

Select Left/Right

bpy.ops.nla.select_leftright(mode='CHECK', extend=False)

Properties:	Values:
Extend Select	True
Mode	CHECK

(default) LEFT_BRACKET → nla.select_leftright : **KEYBOARD** → PRESS

Select Left/Right

bpy.ops.nla.select_leftright(mode='CHECK', extend=False)

Properties:	Values:
Extend Select	False
Mode	LEFT

(default) RIGHT_BRACKET → nla.select_leftright : **KEYBOARD** → PRESS

Select Left/Right

bpy.ops.nla.select_leftright(mode='CHECK', extend=False)

Properties:	Values:
Extend Select	False
Mode	RIGHT

(default) A → nla.select_all_toggle : **KEYBOARD** → PRESS

(De)select All

bpy.ops.nla.select_all_toggle(invert=False)

Properties:	Values:
Invert	False

(default) Ctrl-I → nla.select_all_toggle : **KEYBOARD** → PRESS

(De)select All

bpy.ops.nla.select_all_toggle(invert=False)

Properties:	Values:
Invert	True

(default) B → nla.select_border : **KEYBOARD** → PRESS

Border Select

```
bpy.ops.nla.select_border(gesture_mode=0, xmin=0, xmax=0, ymin=0, ymax=0, extend=True, axis_range=False)
```

Properties:	Values:
Axis Range	False

(default) Alt-B → nla.select_border : **KEYBOARD** → PRESS

Border Select

```
bpy.ops.nla.select_border(gesture_mode=0, xmin=0, xmax=0, ymin=0, ymax=0, extend=True, axis_range=False)
```

Properties:	Values:
Axis Range	True

(default) Ctrl-Alt-P → nla.previewrange_set : **KEYBOARD** → PRESS

Auto-Set Preview Range

```
bpy.ops.nla.previewrange_set()
```

(default) HOME → nla.view_all : **KEYBOARD** → PRESS

View All

```
bpy.ops.nla.view_all()
```

(default) NDOF_BUTTON_FIT → nla.view_all : **NDOF** → PRESS

View All

```
bpy.ops.nla.view_all()
```

(default) NUMPAD_PERIOD → nla.view_selected : **KEYBOARD** → PRESS

View Selected

```
bpy.ops.nla.view_selected()
```

(default) NUMPAD_0 → nla.view_frame : **KEYBOARD** → PRESS

View Frame

```
bpy.ops.nla.view_frame()
```

(default) Shift-A → nla.actionclip_add : **KEYBOARD** → PRESS

Add Action Strip

```
bpy.ops.nla.actionclip_add(action='<UNKNOWN ENUM>')
```

(default) Shift-T → nla.transition_add : **KEYBOARD** → PRESS

Add Transition

```
bpy.ops.nla.transition_add()
```

(default) Shift-K → nla.soundclip_add : **KEYBOARD** → PRESS

Add Sound Clip

```
bpy.ops.nla.soundclip_add()
```

(default) Shift-G → nla.meta_add : **KEYBOARD** → PRESS

Add Meta-Strips

```
bpy.ops.nla.meta_add()
```

(default) Alt-G → nla.meta_remove : **KEYBOARD** → PRESS

Remove Meta-Strips

bpy.ops.nla.meta_remove()

(default) Shift-D → nla.duplicate : **KEYBOARD** → PRESS

Duplicate Strips

bpy.ops.nla.duplicate(linked=False, mode='TRANSLATION')

Properties:	Values:
Linked	False

(default) Alt-D → nla.duplicate : **KEYBOARD** → PRESS

Duplicate Strips

bpy.ops.nla.duplicate(linked=False, mode='TRANSLATION')

Properties:	Values:
Linked	True

(default) U → nla.make_single_user : **KEYBOARD** → PRESS

Make Single User

bpy.ops.nla.make_single_user()

(default) X → nla.delete : **KEYBOARD** → PRESS

Delete Strips

bpy.ops.nla.delete()

(default) DEL → nla.delete : **KEYBOARD** → PRESS

Delete Strips

bpy.ops.nla.delete()

(default) Y → nla.split : **KEYBOARD** → PRESS

Split Strips

bpy.ops.nla.split()

(default) H → nla.mute_toggle : **KEYBOARD** → PRESS

Toggle Muting

bpy.ops.nla.mute_toggle()

(default) Alt-F → nla.swap : **KEYBOARD** → PRESS

Swap Strips

bpy.ops.nla.swap()

(default) PAGE_UP → nla.move_up : **KEYBOARD** → PRESS

Move Strips Up

bpy.ops.nla.move_up()

(default) PAGE_DOWN → nla.move_down : **KEYBOARD** → PRESS

Move Strips Down

bpy.ops.nla.move_down()

(default) Ctrl-A → nla.apply_scale : **KEYBOARD** → PRESS

Apply Scale

bpy.ops.nla.apply_scale()

(default) Alt-S → nla.clear_scale : **KEYBOARD** → PRESS

Clear Scale

bpy.ops.nla.clear_scale()

(default) Shift-S → nla.snap : **KEYBOARD** → PRESS

Snap Strips

```
bpy.ops.nla.snap(type='CFRA')
```

(default) Ctrl-Shift-M → nla.fmodifier_add : **KEYBOARD** → PRESS

Add F-Modifier

```
bpy.ops.nla.fmodifier_add(type='NULL', only_active=True)
```

(default) G → transform.transform : **KEYBOARD** → PRESS

Transform

```
bpy.ops.transform.transform(mode='TRANSLATION', value=(0, 0, 0, 0), axis=(0, 0, 0), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpen-
cil_strokes=False, release_confirm=False)
```

Properties:	Values:
Mode	TRANSLATION

(default) EVT_TWEAK_S → transform.transform : **TWEAK** → ANY

Transform

```
bpy.ops.transform.transform(mode='TRANSLATION', value=(0, 0, 0, 0), axis=(0, 0, 0), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpen-
cil_strokes=False, release_confirm=False)
```

Properties:	Values:
Mode	TRANSLATION

(default) E → transform.transform : **KEYBOARD** → PRESS

Transform

```
bpy.ops.transform.transform(mode='TRANSLATION', value=(0, 0, 0, 0), axis=(0, 0, 0), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpen-
cil_strokes=False, release_confirm=False)
```

Properties:	Values:
Mode	TIME_EXTEND

(default) S → transform.transform : **KEYBOARD** → PRESS

Transform

```
bpy.ops.transform.transform(mode='TRANSLATION', value=(0, 0, 0, 0), axis=(0, 0, 0), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpen-
cil_strokes=False, release_confirm=False)
```

Properties:	Values:
Mode	TIME_SCALE

(default) M → marker.add : **KEYBOARD** → PRESS

Add Time Marker

```
bpy.ops.marker.add()
```


(default) **Ctrl-M** → marker.rename : **KEYBOARD** → PRESS

Rename Marker

```
bpy.ops.marker.rename(name="RenamedMarker")
```

Node Editor

Quick Reference

Hotkey	Operator
<i>Alt-EVT_TWEAK_A</i>	<code>bpy.ops.node.select_lasso()</code>
<i>Shift-Alt-EVT_TWEAK_A</i>	<code>bpy.ops.node.select_lasso()</code>
<i>Alt-SELECTMOUSE</i>	<code>bpy.ops.node.backimage_sample()</code>
<i>Ctrl-A</i>	<code>bpy.ops.node.select_all()</code>
<i>Tab</i>	<code>bpy.ops.node.add_search()</code>
<i>ACTIONMOUSE</i>	<code>bpy.ops.node.select()</code>
<i>SELECTMOUSE</i>	<code>bpy.ops.node.select()</code>
<i>Ctrl-ACTIONMOUSE</i>	<code>bpy.ops.node.select()</code>
<i>Ctrl-SELECTMOUSE</i>	<code>bpy.ops.node.select()</code>
<i>Alt-ACTIONMOUSE</i>	<code>bpy.ops.node.select()</code>
<i>Alt-SELECTMOUSE</i>	<code>bpy.ops.node.select()</code>
<i>Ctrl-Alt-ACTIONMOUSE</i>	<code>bpy.ops.node.select()</code>
<i>Ctrl-Alt-SELECTMOUSE</i>	<code>bpy.ops.node.select()</code>
<i>Shift-ACTIONMOUSE</i>	<code>bpy.ops.node.select()</code>
<i>Shift-SELECTMOUSE</i>	<code>bpy.ops.node.select()</code>
<i>Ctrl-Shift-ACTIONMOUSE</i>	<code>bpy.ops.node.select()</code>
<i>Ctrl-Shift-SELECTMOUSE</i>	<code>bpy.ops.node.select()</code>
<i>Shift-Alt-ACTIONMOUSE</i>	<code>bpy.ops.node.select()</code>
<i>Shift-Alt-SELECTMOUSE</i>	<code>bpy.ops.node.select()</code>
<i>Ctrl-Shift-Alt-ACTIONMOUSE</i>	<code>bpy.ops.node.select()</code>
<i>Ctrl-Shift-Alt-SELECTMOUSE</i>	<code>bpy.ops.node.select()</code>
<i>EVT_TWEAK_S</i>	<code>bpy.ops.node.select_border()</code>
<i>Ctrl-Alt-EVT_TWEAK_A</i>	<code>bpy.ops.node.select_lasso()</code>
<i>Ctrl-Shift-Alt-EVT_TWEAK_A</i>	<code>bpy.ops.node.select_lasso()</code>
<i>C</i>	<code>bpy.ops.node.select_circle()</code>
<i>LEFTMOUSE</i>	<code>bpy.ops.node.link()</code>
<i>Ctrl-LEFTMOUSE</i>	<code>bpy.ops.node.link()</code>
<i>LEFTMOUSE</i>	<code>bpy.ops.node.resize()</code>
<i>Shift-LEFTMOUSE</i>	<code>bpy.ops.node.add_reroute()</code>
<i>Ctrl-LEFTMOUSE</i>	<code>bpy.ops.node.links_cut()</code>
<i>Ctrl-Shift-LEFTMOUSE</i>	<code>bpy.ops.node.select_link_viewer()</code>
<i>Alt-MIDDLEMOUSE</i>	<code>bpy.ops.node.backimage_move()</code>
<i>V</i>	<code>bpy.ops.node.backimage_zoom()</code>
<i>Alt-V</i>	<code>bpy.ops.node.backimage_zoom()</code>
<i>Alt-HOME</i>	<code>bpy.ops.node.backimage_fit()</code>
<i>Alt-ACTIONMOUSE</i>	<code>bpy.ops.node.backimage_sample()</code>
<i>F</i>	<code>bpy.ops.node.link_make()</code>
<i>Shift-F</i>	<code>bpy.ops.node.link_make()</code>
<i>Shift-A</i>	<code>bpy.ops.wm.call_menu()</code>

Continued on next page

Table 2.13 – continued from previous page

Hotkey	Operator
<i>Shift-D</i>	<code>bpy.ops.node.duplicate_move()</code>
<i>Ctrl-Shift-D</i>	<code>bpy.ops.node.duplicate_move_keep_inputs()</code>
<i>Ctrl-P</i>	<code>bpy.ops.node.parent_set()</code>
<i>Alt-P</i>	<code>bpy.ops.node.detach()</code>
<i>Ctrl-J</i>	<code>bpy.ops.node.join()</code>
<i>H</i>	<code>bpy.ops.node.hide_toggle()</code>
<i>M</i>	<code>bpy.ops.node.mute_toggle()</code>
<i>Shift-H</i>	<code>bpy.ops.node.preview_toggle()</code>
<i>Ctrl-H</i>	<code>bpy.ops.node.hide_socket_toggle()</code>
<i>HOME</i>	<code>bpy.ops.node.view_all()</code>
<i>NDOF_BUTTON_FIT</i>	<code>bpy.ops.node.view_all()</code>
<i>NUMPAD_PERIOD</i>	<code>bpy.ops.node.view_selected()</code>
<i>B</i>	<code>bpy.ops.node.select_border()</code>
<i>X</i>	<code>bpy.ops.node.delete()</code>
<i>DEL</i>	<code>bpy.ops.node.delete()</code>
<i>Ctrl-X</i>	<code>bpy.ops.node.delete_reconnect()</code>
<i>A</i>	<code>bpy.ops.node.select_all()</code>
<i>Ctrl-I</i>	<code>bpy.ops.node.select_all()</code>
<i>Shift-L</i>	<code>bpy.ops.node.select_linked_to()</code>
<i>L</i>	<code>bpy.ops.node.select_linked_from()</code>
<i>Shift-G</i>	<code>bpy.ops.node.select_grouped()</code>
<i>Ctrl-Shift-G</i>	<code>bpy.ops.node.select_grouped()</code>
<i>Shift-RIGHT_BRACKET</i>	<code>bpy.ops.node.select_same_type_step()</code>
<i>Shift-LEFT_BRACKET</i>	<code>bpy.ops.node.select_same_type_step()</code>
<i>Ctrl-F</i>	<code>bpy.ops.node.find_node()</code>
<i>Ctrl-G</i>	<code>bpy.ops.node.group_make()</code>
<i>Alt-G</i>	<code>bpy.ops.node.group_ungroup()</code>
<i>P</i>	<code>bpy.ops.node.group_separate()</code>
<i>Tab</i>	<code>bpy.ops.node.group_edit()</code>
<i>Ctrl-Tab</i>	<code>bpy.ops.node.group_edit()</code>
<i>Ctrl-R</i>	<code>bpy.ops.node.read_renderlayers()</code>
<i>Shift-R</i>	<code>bpy.ops.node.read_fullsamplelayers()</code>
<i>Z</i>	<code>bpy.ops.node.render_changed()</code>
<i>Ctrl-C</i>	<code>bpy.ops.node.clipboard_copy()</code>
<i>Ctrl-V</i>	<code>bpy.ops.node.clipboard_paste()</code>
<i>Ctrl-B</i>	<code>bpy.ops.node.viewer_border()</code>
<i>Ctrl-Alt-B</i>	<code>bpy.ops.node.clear_viewer_border()</code>
<i>G</i>	<code>bpy.ops.node.translate_attach()</code>
<i>EVT_TWEAK_A</i>	<code>bpy.ops.node.translate_attach()</code>
<i>EVT_TWEAK_S</i>	<code>bpy.ops.node.translate_attach()</code>
<i>G</i>	<code>bpy.ops.transform.translate()</code>
<i>EVT_TWEAK_A</i>	<code>bpy.ops.transform.translate()</code>
<i>EVT_TWEAK_S</i>	<code>bpy.ops.transform.translate()</code>
<i>R</i>	<code>bpy.ops.transform.rotate()</code>
<i>S</i>	<code>bpy.ops.transform.resize()</code>
<i>Alt-D</i>	<code>bpy.ops.node.move_detach_links()</code>
<i>Alt-EVT_TWEAK_A</i>	<code>bpy.ops.node.move_detach_links_release()</code>
<i>Alt-EVT_TWEAK_S</i>	<code>bpy.ops.node.move_detach_links()</code>
<i>Shift-Tab</i>	<code>bpy.ops.wm.context_toggle()</code>

Continued on next page

Table 2.13 – continued from previous page

Hotkey	Operator
<i>Ctrl-Shift-Tab</i>	<code>bpy.ops.wm.context_menu_enum()</code>

Detailed Reference

Note: Hotkeys marked with the “(default)” prefix are inherited from the default blender keymap

Alt-EVT_TWEAK_A → node.select_lasso : **TWEAK** → ANY

Lasso Select

`bpy.ops.node.select_lasso(path=[], deselect=False, extend=True)`

Shift-Alt-EVT_TWEAK_A → node.select_lasso : **TWEAK** → ANY

Lasso Select

`bpy.ops.node.select_lasso(path=[], deselect=False, extend=True)`

Properties:	Values:
Deselect	True

Alt-SELECTMOUSE → node.backimage_sample : **MOUSE** → PRESS

Backimage Sample

`bpy.ops.node.backimage_sample()`

Ctrl-A → node.select_all : **KEYBOARD** → PRESS

(De)select All

`bpy.ops.node.select_all(action='TOGGLE')`

Properties:	Values:
Action	TOGGLE

Tab → node.add_search : **KEYBOARD** → DOUBLE_CLICK

Search and Add Node

`bpy.ops.node.add_search(type='', settings=[], use_transform=False, node_item='0')`

Properties:	Values:
Use Transform	True

(default) ACTIONMOUSE → node.select : **MOUSE** → PRESS

Select

`bpy.ops.node.select(mouse_x=0, mouse_y=0, extend=False)`

Properties:	Values:
Extend	False

(default) SELECTMOUSE → node.select : **MOUSE** → PRESS

Select

`bpy.ops.node.select(mouse_x=0, mouse_y=0, extend=False)`

Properties:	Values:
Extend	False

(default) Ctrl-ACTIONMOUSE → node.select : **MOUSE** → PRESS

Select

```
bpy.ops.node.select(mouse_x=0, mouse_y=0, extend=False)
```

Properties:	Values:
Extend	False

(default) Ctrl-SELECTMOUSE → node.select : **MOUSE** → PRESS
Select

```
bpy.ops.node.select(mouse_x=0, mouse_y=0, extend=False)
```

Properties:	Values:
Extend	False

(default) Alt-ACTIONMOUSE → node.select : **MOUSE** → PRESS
Select

```
bpy.ops.node.select(mouse_x=0, mouse_y=0, extend=False)
```

Properties:	Values:
Extend	False

(default) Alt-SELECTMOUSE → node.select : **MOUSE** → PRESS
Select

```
bpy.ops.node.select(mouse_x=0, mouse_y=0, extend=False)
```

Properties:	Values:
Extend	False

(default) Ctrl-Alt-ACTIONMOUSE → node.select : **MOUSE** → PRESS
Select

```
bpy.ops.node.select(mouse_x=0, mouse_y=0, extend=False)
```

Properties:	Values:
Extend	False

(default) Ctrl-Alt-SELECTMOUSE → node.select : **MOUSE** → PRESS
Select

```
bpy.ops.node.select(mouse_x=0, mouse_y=0, extend=False)
```

Properties:	Values:
Extend	False

(default) Shift-ACTIONMOUSE → node.select : **MOUSE** → PRESS
Select

```
bpy.ops.node.select(mouse_x=0, mouse_y=0, extend=False)
```

Properties:	Values:
Extend	True

(default) Shift-SELECTMOUSE → node.select : **MOUSE** → PRESS
Select

```
bpy.ops.node.select(mouse_x=0, mouse_y=0, extend=False)
```

Properties:	Values:
Extend	True

(default) Ctrl-Shift-ACTIONMOUSE → node.select : **MOUSE** → PRESS
Select

```
bpy.ops.node.select(mouse_x=0, mouse_y=0, extend=False)
```

Properties:	Values:
Extend	True

(default) Ctrl-Shift-SELECTMOUSE → node.select : **MOUSE** → PRESS

Select

bpy.ops.node.select(mouse_x=0, mouse_y=0, extend=False)

Properties:	Values:
Extend	True

(default) Shift-Alt-ACTIONMOUSE → node.select : **MOUSE** → PRESS

Select

bpy.ops.node.select(mouse_x=0, mouse_y=0, extend=False)

Properties:	Values:
Extend	True

(default) Shift-Alt-SELECTMOUSE → node.select : **MOUSE** → PRESS

Select

bpy.ops.node.select(mouse_x=0, mouse_y=0, extend=False)

Properties:	Values:
Extend	True

(default) Ctrl-Shift-Alt-ACTIONMOUSE → node.select : **MOUSE** → PRESS

Select

bpy.ops.node.select(mouse_x=0, mouse_y=0, extend=False)

Properties:	Values:
Extend	True

(default) Ctrl-Shift-Alt-SELECTMOUSE → node.select : **MOUSE** → PRESS

Select

bpy.ops.node.select(mouse_x=0, mouse_y=0, extend=False)

Properties:	Values:
Extend	True

(default) EVT_TWEAK_S → node.select_border : **TWEAK** → ANY

Border Select

bpy.ops.node.select_border(gesture_mode=0, xmin=0, xmax=0, ymin=0, ymax=0, extend=True, tweak=False)

Properties:	Values:
Tweak	True

(default) Ctrl-Alt-EVT_TWEAK_A → node.select_lasso : **TWEAK** → ANY

Lasso Select

bpy.ops.node.select_lasso(path=[], deselect=False, extend=True)

Properties:	Values:
Deselect	False

(default) Ctrl-Shift-Alt-EVT_TWEAK_A → node.select_lasso : **TWEAK** → ANY

Lasso Select

bpy.ops.node.select_lasso(path=[], deselect=False, extend=True)

Properties:	Values:
Deselect	True

(default) C → node.select_circle : **KEYBOARD** → PRESS

Circle Select

bpy.ops.node.select_circle(x=0, y=0, radius=1, gesture_mode=0)

(default) LEFTMOUSE → node.link : **MOUSE** → PRESS

Link Nodes

bpy.ops.node.link(detach=False)

Properties:	Values:
Detach	False

(default) Ctrl-LEFTMOUSE → node.link : **MOUSE** → PRESS

Link Nodes

bpy.ops.node.link(detach=False)

Properties:	Values:
Detach	True

(default) LEFTMOUSE → node.resize : **MOUSE** → PRESS

Resize Node

bpy.ops.node.resize()

(default) Shift-LEFTMOUSE → node.add_reroute : **MOUSE** → PRESS

Add Reroute

bpy.ops.node.add_reroute(path=[], cursor=6)

(default) Ctrl-LEFTMOUSE → node.links_cut : **MOUSE** → PRESS

Cut Links

bpy.ops.node.links_cut(path=[], cursor=9)

(default) Ctrl-Shift-LEFTMOUSE → node.select_link_viewer : **MOUSE** → PRESS

Link Viewer

bpy.ops.node.select_link_viewer(NODE_OT_select={"mouse_x":0, "mouse_y":0, "extend":False},
NODE_OT_link_viewer={})

Properties:	Values:
Select	N/A
Link to Viewer Node	N/A

(default) Alt-MIDDLEMOUSE → node.backimage_move : **MOUSE** → PRESS

Background Image Move

bpy.ops.node.backimage_move()

(default) V → node.backimage_zoom : **KEYBOARD** → PRESS

Background Image Zoom

bpy.ops.node.backimage_zoom(factor=1.2)

Properties:	Values:
Factor	0.833329975605011

(default) Alt-V → node.backimage_zoom : **KEYBOARD** → PRESS

Background Image Zoom

bpy.ops.node.backimage_zoom(factor=1.2)

Properties:	Values:
Factor	1.2000000476837158

(default) Alt-HOME → node.backimage_fit : **KEYBOARD** → PRESS

Background Image Fit

bpy.ops.node.backimage_fit()

(default) Alt-ACTIONMOUSE → node.backimage_sample : **MOUSE** → PRESS

Backimage Sample

bpy.ops.node.backimage_sample()

(default) F → node.link_make : **KEYBOARD** → PRESS

Make Links

bpy.ops.node.link_make(replace=False)

Properties:	Values:
Replace	False

(default) Shift-F → node.link_make : **KEYBOARD** → PRESS

Make Links

bpy.ops.node.link_make(replace=False)

Properties:	Values:
Replace	True

(default) Shift-A → wm.call_menu : **KEYBOARD** → PRESS

Call Menu

bpy.ops.wm.call_menu(name='')

Properties:	Values:
Name	NODE_MT_add

(default) Shift-D → node.duplicate_move : **KEYBOARD** → PRESS

Duplicate

bpy.ops.node.duplicate_move(NODE_OT_duplicate={"keep_inputs":False}, NODE_OT_translate_attach={"TRANSFORM_OT_translate":{"value":(0, 0, 0), "constraint_axis":(False, False, False), "constraint_orientation":'GLOBAL', "mirror":False, "proportional":'DISABLED', "proportional_edit_falloff":'SMOOTH', "proportional_size":1, "snap":False, "snap_target":'CLOSEST', "snap_point":(0, 0, 0), "snap_align":False, "snap_normal":(0, 0, 0), "gpencil_strokes":False, "texture_space":False, "remove_on_cancel":False, "release_confirm":False}, "NODE_OT_attach":{ }, "NODE_OT_insert_offset":{ }})

Properties:	Values:
Duplicate Nodes	N/A
Move and Attach	N/A

(default) Ctrl-Shift-D → node.duplicate_move_keep_inputs : **KEYBOARD** → PRESS

Duplicate

bpy.ops.node.duplicate_move_keep_inputs(NODE_OT_duplicate={"keep_inputs":False}, NODE_OT_translate_attach={"TRANSFORM_OT_translate":{"value":(0, 0, 0), "constraint_axis":(False, False, False), "constraint_orientation":'GLOBAL', "mirror":False, "proportional":'DISABLED', "proportional_edit_falloff":'SMOOTH', "proportional_size":1, "snap":False, "snap_target":'CLOSEST', "snap_point":(0, 0, 0), "snap_align":False, "snap_normal":(0, 0, 0), "gpencil_strokes":False, "texture_space":False, "remove_on_cancel":False, "release_confirm":False}, "NODE_OT_attach":{ }, "NODE_OT_insert_offset":{ }})

Properties:	Values:
Duplicate Nodes	N/A
Move and Attach	N/A

(default) Ctrl-P → node.parent_set : **KEYBOARD** → PRESS

Make Parent

bpy.ops.node.parent_set()

(default) Alt-P → node.detach : **KEYBOARD** → PRESS

Detach Nodes

bpy.ops.node.detach()

(default) Ctrl-J → node.join : **KEYBOARD** → PRESS

Join Nodes

bpy.ops.node.join()

(default) H → node.hide_toggle : **KEYBOARD** → PRESS

Hide

bpy.ops.node.hide_toggle()

(default) M → node.mute_toggle : **KEYBOARD** → PRESS

Toggle Node Mute

bpy.ops.node.mute_toggle()

(default) Shift-H → node.preview_toggle : **KEYBOARD** → PRESS

Toggle Node Preview

bpy.ops.node.preview_toggle()

(default) Ctrl-H → node.hide_socket_toggle : **KEYBOARD** → PRESS

Toggle Hidden Node Sockets

bpy.ops.node.hide_socket_toggle()

(default) HOME → node.view_all : **KEYBOARD** → PRESS

View All

bpy.ops.node.view_all()

(default) NDOF_BUTTON_FIT → node.view_all : **NDOF** → PRESS

View All

bpy.ops.node.view_all()

(default) NUMPAD_PERIOD → node.view_selected : **KEYBOARD** → PRESS

View Selected

bpy.ops.node.view_selected()

(default) B → node.select_border : **KEYBOARD** → PRESS

Border Select

bpy.ops.node.select_border(gesture_mode=0, xmin=0, xmax=0, ymin=0, ymax=0, extend=True, tweak=False)

Properties:	Values:
Tweak	False

(default) X → node.delete : **KEYBOARD** → PRESS

Delete

bpy.ops.node.delete()

(default) DEL → node.delete : **KEYBOARD** → PRESS

Delete

bpy.ops.node.delete()

(default) Ctrl-X → node.delete_reconnect : **KEYBOARD** → PRESS

Delete with Reconnect

bpy.ops.node.delete_reconnect()

(default) A → node.select_all : **KEYBOARD** → PRESS

(De)select All

bpy.ops.node.select_all(action='TOGGLE')

Properties:	Values:
Action	TOGGLE

(default) Ctrl-I → node.select_all : **KEYBOARD** → PRESS

(De)select All

bpy.ops.node.select_all(action='TOGGLE')

Properties:	Values:
Action	INVERT

(default) Shift-L → node.select_linked_to : **KEYBOARD** → PRESS

Select Linked To

bpy.ops.node.select_linked_to()

(default) L → node.select_linked_from : **KEYBOARD** → PRESS

Select Linked From

bpy.ops.node.select_linked_from()

(default) Shift-G → node.select_grouped : **KEYBOARD** → PRESS

Select Grouped

bpy.ops.node.select_grouped(extend=False, type='TYPE')

Properties:	Values:
Extend	False

(default) Ctrl-Shift-G → node.select_grouped : **KEYBOARD** → PRESS

Select Grouped

bpy.ops.node.select_grouped(extend=False, type='TYPE')

Properties:	Values:
Extend	True

(default) Shift-RIGHT_BRACKET → node.select_same_type_step : **KEYBOARD** → PRESS

Activate Same Type Next/Prev

bpy.ops.node.select_same_type_step(prev=False)

Properties:	Values:
Previous	False

(default) Shift-LEFT_BRACKET → node.select_same_type_step : **KEYBOARD** → PRESS

Activate Same Type Next/Prev

bpy.ops.node.select_same_type_step(prev=False)

Properties:	Values:
Previous	True

(default) Ctrl-F → node.find_node : **KEYBOARD** → PRESS

Find Node

bpy.ops.node.find_node(prev=False)

(default) Ctrl-G → node.group_make : **KEYBOARD** → PRESS
Make Group

bpy.ops.node.group_make()

(default) Alt-G → node.group_ungroup : **KEYBOARD** → PRESS
Ungroup

bpy.ops.node.group_ungroup()

(default) P → node.group_separate : **KEYBOARD** → PRESS
Separate

bpy.ops.node.group_separate(type='COPY')

(default) Tab → node.group_edit : **KEYBOARD** → PRESS
Edit Group

bpy.ops.node.group_edit(exit=False)

Properties:	Values:
Exit	False

(default) Ctrl-Tab → node.group_edit : **KEYBOARD** → PRESS
Edit Group

bpy.ops.node.group_edit(exit=False)

Properties:	Values:
Exit	True

(default) Ctrl-R → node.read_renderlayers : **KEYBOARD** → PRESS
Read Render Layers

bpy.ops.node.read_renderlayers()

(default) Shift-R → node.read_fullsamplelayers : **KEYBOARD** → PRESS
Read Full Sample Layers

bpy.ops.node.read_fullsamplelayers()

(default) Z → node.render_changed : **KEYBOARD** → PRESS
Render Changed Layer

bpy.ops.node.render_changed()

(default) Ctrl-C → node.clipboard_copy : **KEYBOARD** → PRESS
Copy to Clipboard

bpy.ops.node.clipboard_copy()

(default) Ctrl-V → node.clipboard_paste : **KEYBOARD** → PRESS
Paste from Clipboard

bpy.ops.node.clipboard_paste()

(default) Ctrl-B → node.viewer_border : **KEYBOARD** → PRESS
Viewer Border

bpy.ops.node.viewer_border(gesture_mode=0, xmin=0, xmax=0, ymin=0, ymax=0, extend=True)

(default) Ctrl-Alt-B → node.clear_viewer_border : **KEYBOARD** → PRESS
Clear Viewer Border

bpy.ops.node.clear_viewer_border()

(default) G → node.translate_attach : **KEYBOARD** → PRESS

Move and Attach

```
bpy.ops.node.translate_attach(TRANSFORM_OT_translate={"value":(0, 0, 0), "constraint_axis":(False, False, False), "constraint_orientation":'GLOBAL', "mirror":False, "proportional":'DISABLED', "proportional_edit_falloff":'SMOOTH', "proportional_size":1, "snap":False, "snap_target":'CLOSEST', "snap_point":(0, 0, 0), "snap_align":False, "snap_normal":(0, 0, 0), "gpencil_strokes":False, "texture_space":False, "remove_on_cancel":False, "release_confirm":False}, NODE_OT_attach={}, NODE_OT_insert_offset={})
```

Properties:	Values:
Translate	N/A
Attach Nodes	N/A
Insert Offset	N/A

(default) EVT_TWEAK_A → node.translate_attach : **TWEAK** → ANY

Move and Attach

```
bpy.ops.node.translate_attach(TRANSFORM_OT_translate={"value":(0, 0, 0), "constraint_axis":(False, False, False), "constraint_orientation":'GLOBAL', "mirror":False, "proportional":'DISABLED', "proportional_edit_falloff":'SMOOTH', "proportional_size":1, "snap":False, "snap_target":'CLOSEST', "snap_point":(0, 0, 0), "snap_align":False, "snap_normal":(0, 0, 0), "gpencil_strokes":False, "texture_space":False, "remove_on_cancel":False, "release_confirm":False}, NODE_OT_attach={}, NODE_OT_insert_offset={})
```

Properties:	Values:
Translate	N/A
Attach Nodes	N/A
Insert Offset	N/A

(default) EVT_TWEAK_S → node.translate_attach : **TWEAK** → ANY

Move and Attach

```
bpy.ops.node.translate_attach(TRANSFORM_OT_translate={"value":(0, 0, 0), "constraint_axis":(False, False, False), "constraint_orientation":'GLOBAL', "mirror":False, "proportional":'DISABLED', "proportional_edit_falloff":'SMOOTH', "proportional_size":1, "snap":False, "snap_target":'CLOSEST', "snap_point":(0, 0, 0), "snap_align":False, "snap_normal":(0, 0, 0), "gpencil_strokes":False, "texture_space":False, "remove_on_cancel":False, "release_confirm":False}, NODE_OT_attach={}, NODE_OT_insert_offset={})
```

Properties:	Values:
Translate	N/A
Attach Nodes	N/A
Insert Offset	N/A

(default) G → transform.translate : **KEYBOARD** → PRESS

Translate

```
bpy.ops.transform.translate(value=(0, 0, 0), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpencil_strokes=False, texture_space=False, remove_on_cancel=False, release_confirm=False)
```

Properties:	Values:
Confirm on Release	True

(default) EVT_TWEAK_A → transform.translate : **TWEAK** → ANY

Translate

```
bpy.ops.transform.translate(value=(0, 0, 0), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpencil_strokes=False, texture_space=False, remove_on_cancel=False, release_confirm=False)
```

Properties:	Values:
Confirm on Release	True

(default) EVT_TWEAK_S → transform.translate : **TWEAK** → ANY

Translate

```
bpy.ops.transform.translate(value=(0, 0, 0), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpencil_strokes=False, texture_space=False, remove_on_cancel=False, release_confirm=False)
```

Properties:	Values:
Confirm on Release	True

(default) R → transform.rotate : **KEYBOARD** → PRESS

Rotate

```
bpy.ops.transform.rotate(value=0, axis=(0, 0, 0), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpencil_strokes=False, release_confirm=False)
```

(default) S → transform.resize : **KEYBOARD** → PRESS

Resize

```
bpy.ops.transform.resize(value=(1, 1, 1), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpencil_strokes=False, texture_space=False, remove_on_cancel=False, release_confirm=False)
```

(default) Alt-D → node.move_detach_links : **KEYBOARD** → PRESS

Detach

```
bpy.ops.node.move_detach_links(NODE_OT_links_detach={}, TRANSFORM_OT_translate={"value":(0, 0, 0), "constraint_axis":(False, False, False), "constraint_orientation":'GLOBAL', "mirror":False, "proportional":'DISABLED', "proportional_edit_falloff":'SMOOTH', "proportional_size":1, "snap":False, "snap_target":'CLOSEST', "snap_point":(0, 0, 0), "snap_align":False, "snap_normal":(0, 0, 0), "gpencil_strokes":False, "texture_space":False, "remove_on_cancel":False, "release_confirm":False}, NODE_OT_insert_offset={})
```

Properties:	Values:
Detach Links	N/A
Translate	N/A
Insert Offset	N/A

(default) Alt-EVT_TWEAK_A → node.move_detach_links_release : **TWEAK** → ANY

Detach

```
bpy.ops.node.move_detach_links_release(NODE_OT_links_detach={}, NODE_OT_translate_attach={"TRANSFORM_OT_translate":(0, 0, 0), "constraint_axis":(False, False, False), "constraint_orientation":'GLOBAL', "mirror":False, "proportional":'DISABLED', "proportional_edit_falloff":'SMOOTH', "proportional_size":1, "snap":False, "snap_target":'CLOSEST', "snap_point":(0, 0, 0), "snap_align":False, "snap_normal":(0, 0, 0),
```

“gpencil_strokes”:False, “texture_space”:False, “remove_on_cancel”:False, “release_confirm”:False},
 “NODE_OT_attach”:{}, “NODE_OT_insert_offset”:{}})

Properties:	Values:
Detach Links	N/A
Move and Attach	N/A

(default) Alt-EVT-TWEAK-S → node.move_detach_links : **TWEAK** → ANY

Detach

bpy.ops.node.move_detach_links(NODE_OT_links_detach={}, TRANSFORM_OT_translate={"value":(0, 0, 0), “constraint_axis”:(False, False, False), “constraint_orientation”:’GLOBAL’, “mirror”:False, “proportional”:'DISABLED', “proportional_edit_falloff”:'SMOOTH', “proportional_size”:1, “snap”:False, “snap_target”:'CLOSEST', “snap_point”:(0, 0, 0), “snap_align”:False, “snap_normal”:(0, 0, 0), “gpencil_strokes”:False, “texture_space”:False, “remove_on_cancel”:False, “release_confirm”:False}, NODE_OT_insert_offset={})

Properties:	Values:
Detach Links	N/A
Translate	N/A
Insert Offset	N/A

(default) Shift-Tab → wm.context_toggle : **KEYBOARD** → PRESS

Context Toggle

bpy.ops.wm.context_toggle(data_path=’')

Properties:	Values:
Context Attributes	tool_settings.use_snap

(default) Ctrl-Shift-Tab → wm.context_menu_enum : **KEYBOARD** → PRESS

Context Enum Menu

bpy.ops.wm.context_menu_enum(data_path=’')

Properties:	Values:
Context Attributes	tool_settings.snap_node_element

Object Mode

Quick Reference

Hotkey	Operator
<i>Ctrl-A</i>	<code>bpy.ops.object.select_all()</code>
<i>A</i>	<code>bpy.ops.wm.call_menu()</code>
<i>Shift-O</i>	<code>bpy.ops.wm.context_cycle_enum()</code>
<i>O</i>	<code>bpy.ops.wm.context_toggle()</code>
<i>P</i>	<code>bpy.ops.view3d.game_start()</code>
<i>A</i>	<code>bpy.ops.object.select_all()</code>
<i>Ctrl-I</i>	<code>bpy.ops.object.select_all()</code>
<i>Ctrl-NUMPAD_PLUS</i>	<code>bpy.ops.object.select_more()</code>
<i>Ctrl-NUMPAD_MINUS</i>	<code>bpy.ops.object.select_less()</code>
<i>Shift-L</i>	<code>bpy.ops.object.select_linked()</code>
<i>Shift-G</i>	<code>bpy.ops.object.select_grouped()</code>
<i>Ctrl-Shift-M</i>	<code>bpy.ops.object.select_mirror()</code>
Continued on next page	

Table 2.14 – continued from previous page

Hotkey	Operator
<i>LEFT_BRACKET</i>	<code>bpy.ops.object.select_hierarchy()</code>
<i>Shift-LEFT_BRACKET</i>	<code>bpy.ops.object.select_hierarchy()</code>
<i>RIGHT_BRACKET</i>	<code>bpy.ops.object.select_hierarchy()</code>
<i>Shift-RIGHT_BRACKET</i>	<code>bpy.ops.object.select_hierarchy()</code>
<i>Ctrl-P</i>	<code>bpy.ops.object.parent_set()</code>
<i>Ctrl-Shift-P</i>	<code>bpy.ops.object.parent_no_inverse_set()</code>
<i>Alt-P</i>	<code>bpy.ops.object.parent_clear()</code>
<i>Ctrl-T</i>	<code>bpy.ops.object.track_set()</code>
<i>Alt-T</i>	<code>bpy.ops.object.track_clear()</code>
<i>Ctrl-Shift-C</i>	<code>bpy.ops.object.constraint_add_with_targets()</code>
<i>Ctrl-Alt-C</i>	<code>bpy.ops.object.constraints_clear()</code>
<i>Alt-G</i>	<code>bpy.ops.object.location_clear()</code>
<i>Alt-R</i>	<code>bpy.ops.object.rotation_clear()</code>
<i>Alt-S</i>	<code>bpy.ops.object.scale_clear()</code>
<i>Shift-Alt-G</i>	<code>bpy.ops.object.location_clear()</code>
<i>Shift-Alt-R</i>	<code>bpy.ops.object.rotation_clear()</code>
<i>Shift-Alt-S</i>	<code>bpy.ops.object.scale_clear()</code>
<i>Alt-O</i>	<code>bpy.ops.object.origin_clear()</code>
<i>Alt-H</i>	<code>bpy.ops.object.hide_view_clear()</code>
<i>H</i>	<code>bpy.ops.object.hide_view_set()</code>
<i>Shift-H</i>	<code>bpy.ops.object.hide_view_set()</code>
<i>Ctrl-Alt-H</i>	<code>bpy.ops.object.hide_render_clear()</code>
<i>Ctrl-H</i>	<code>bpy.ops.object.hide_render_set()</code>
<i>M</i>	<code>bpy.ops.object.move_to_layer()</code>
<i>X</i>	<code>bpy.ops.object.delete()</code>
<i>Shift-X</i>	<code>bpy.ops.object.delete()</code>
<i>DEL</i>	<code>bpy.ops.object.delete()</code>
<i>Shift-DEL</i>	<code>bpy.ops.object.delete()</code>
<i>Shift-A</i>	<code>bpy.ops.wm.call_menu()</code>
<i>Ctrl-Shift-A</i>	<code>bpy.ops.object.duplicates_make_real()</code>
<i>Ctrl-A</i>	<code>bpy.ops.wm.call_menu()</code>
<i>U</i>	<code>bpy.ops.wm.call_menu()</code>
<i>Ctrl-L</i>	<code>bpy.ops.wm.call_menu()</code>
<i>Shift-D</i>	<code>bpy.ops.object.duplicate_move()</code>
<i>Alt-D</i>	<code>bpy.ops.object.duplicate_move_linked()</code>
<i>Ctrl-J</i>	<code>bpy.ops.object.join()</code>
<i>Alt-C</i>	<code>bpy.ops.object.convert()</code>
<i>Ctrl-Alt-P</i>	<code>bpy.ops.object.proxy_make()</code>
<i>L</i>	<code>bpy.ops.object.make_local()</code>
<i>I</i>	<code>bpy.ops.anim.keyframe_insert_menu()</code>
<i>Alt-I</i>	<code>bpy.ops.anim.keyframe_delete_v3d()</code>
<i>Ctrl-Shift-Alt-I</i>	<code>bpy.ops.anim.keying_set_active_set()</code>
<i>Ctrl-G</i>	<code>bpy.ops.group.create()</code>
<i>Ctrl-Alt-G</i>	<code>bpy.ops.group.objects_remove()</code>
<i>Ctrl-Shift-Alt-G</i>	<code>bpy.ops.group.objects_remove_all()</code>
<i>Ctrl-Shift-G</i>	<code>bpy.ops.group.objects_add_active()</code>
<i>Shift-Alt-G</i>	<code>bpy.ops.group.objects_remove_active()</code>
<i>W</i>	<code>bpy.ops.wm.call_menu()</code>
<i>Ctrl-Shift-T</i>	<code>bpy.ops.object.data_transfer()</code>

Continued on next page

Table 2.14 – continued from previous page

Hotkey	Operator
<i>Ctrl-0</i>	<code>bpy.ops.object.subdivision_set()</code>
<i>Ctrl-1</i>	<code>bpy.ops.object.subdivision_set()</code>
<i>Ctrl-2</i>	<code>bpy.ops.object.subdivision_set()</code>
<i>Ctrl-3</i>	<code>bpy.ops.object.subdivision_set()</code>
<i>Ctrl-4</i>	<code>bpy.ops.object.subdivision_set()</code>
<i>Ctrl-5</i>	<code>bpy.ops.object.subdivision_set()</code>

Detailed Reference

Note: Hotkeys marked with the “(default)” prefix are inherited from the default blender keymap

Ctrl-A → `object.select_all` : **KEYBOARD** → PRESS

(De)select All

`bpy.ops.object.select_all(action='TOGGLE')`

Properties:	Values:
Action	TOGGLE

A → `wm.call_menu` : **KEYBOARD** → PRESS

Call Menu

`bpy.ops.wm.call_menu(name='')`

Properties:	Values:
Name	VIEW3D_MT_object_apply

(default) Shift-O → `wm.context_cycle_enum` : **KEYBOARD** → PRESS

Context Enum Cycle

`bpy.ops.wm.context_cycle_enum(data_path='', reverse=False, wrap=False)`

Properties:	Values:
Context Attributes	<code>tool_settings.proportional_edit_falloff</code>
Wrap	True

(default) O → `wm.context_toggle` : **KEYBOARD** → PRESS

Context Toggle

`bpy.ops.wm.context_toggle(data_path='')`

Properties:	Values:
Context Attributes	<code>tool_settings.use_proportional_edit_objects</code>

(default) P → `view3d.game_start` : **KEYBOARD** → PRESS

Start Game Engine

`bpy.ops.view3d.game_start()`

(default) A → `object.select_all` : **KEYBOARD** → PRESS

(De)select All

`bpy.ops.object.select_all(action='TOGGLE')`

Properties:	Values:
Action	TOGGLE

(default) **Ctrl-I** → object.select_all : **KEYBOARD** → PRESS

(De)select All

bpy.ops.object.select_all(action='TOGGLE')

Properties:	Values:
Action	INVERT

(default) **Ctrl-NUMPAD_PLUS** → object.select_more : **KEYBOARD** → PRESS

Select More

bpy.ops.object.select_more()

(default) **Ctrl-NUMPAD_MINUS** → object.select_less : **KEYBOARD** → PRESS

Select Less

bpy.ops.object.select_less()

(default) **Shift-L** → object.select_linked : **KEYBOARD** → PRESS

Select Linked

bpy.ops.object.select_linked(extend=False, type='OBDATA')

(default) **Shift-G** → object.select_grouped : **KEYBOARD** → PRESS

Select Grouped

bpy.ops.object.select_grouped(extend=False, type='CHILDREN_RECURSIVE')

(default) **Ctrl-Shift-M** → object.select_mirror : **KEYBOARD** → PRESS

Select Mirror

bpy.ops.object.select_mirror(extend=False)

(default) **LEFT_BRACKET** → object.select_hierarchy : **KEYBOARD** → PRESS

Select Hierarchy

bpy.ops.object.select_hierarchy(direction='PARENT', extend=False)

Properties:	Values:
Direction	PARENT
Extend	False

(default) **Shift-LEFT_BRACKET** → object.select_hierarchy : **KEYBOARD** → PRESS

Select Hierarchy

bpy.ops.object.select_hierarchy(direction='PARENT', extend=False)

Properties:	Values:
Direction	PARENT
Extend	True

(default) **RIGHT_BRACKET** → object.select_hierarchy : **KEYBOARD** → PRESS

Select Hierarchy

bpy.ops.object.select_hierarchy(direction='PARENT', extend=False)

Properties:	Values:
Direction	CHILD
Extend	False

(default) **Shift-RIGHT_BRACKET** → object.select_hierarchy : **KEYBOARD** → PRESS

Select Hierarchy

bpy.ops.object.select_hierarchy(direction='PARENT', extend=False)

Properties:	Values:
Direction	CHILD
Extend	True

(default) Ctrl-P → object.parent_set : **KEYBOARD** → PRESS

Make Parent

bpy.ops.object.parent_set(type='OBJECT', xmirror=False, keep_transform=False)

(default) Ctrl-Shift-P → object.parent_no_inverse_set : **KEYBOARD** → PRESS

Make Parent without Inverse

bpy.ops.object.parent_no_inverse_set()

(default) Alt-P → object.parent_clear : **KEYBOARD** → PRESS

Clear Parent

bpy.ops.object.parent_clear(type='CLEAR')

(default) Ctrl-T → object.track_set : **KEYBOARD** → PRESS

Make Track

bpy.ops.object.track_set(type='DAMPTRACK')

(default) Alt-T → object.track_clear : **KEYBOARD** → PRESS

Clear Track

bpy.ops.object.track_clear(type='CLEAR')

(default) Ctrl-Shift-C → object.constraint_add_with_targets : **KEYBOARD** → PRESS

Add Constraint (with Targets)

bpy.ops.object.constraint_add_with_targets(type='<UNKNOWN ENUM>')

(default) Ctrl-Alt-C → object.constraints_clear : **KEYBOARD** → PRESS

Clear Object Constraints

bpy.ops.object.constraints_clear()

(default) Alt-G → object.location_clear : **KEYBOARD** → PRESS

Clear Location

bpy.ops.object.location_clear(clear_delta=False)

Properties:	Values:
Clear Delta	False

(default) Alt-R → object.rotation_clear : **KEYBOARD** → PRESS

Clear Rotation

bpy.ops.object.rotation_clear(clear_delta=False)

Properties:	Values:
Clear Delta	False

(default) Alt-S → object.scale_clear : **KEYBOARD** → PRESS

Clear Scale

bpy.ops.object.scale_clear(clear_delta=False)

Properties:	Values:
Clear Delta	False

(default) Shift-Alt-G → object.location_clear : **KEYBOARD** → PRESS

Clear Location

```
bpy.ops.object.location_clear(clear_delta=False)
```

Properties:	Values:
Clear Delta	True

(default) Shift-Alt-R → object.rotation_clear : **KEYBOARD** → PRESS
Clear Rotation

```
bpy.ops.object.rotation_clear(clear_delta=False)
```

Properties:	Values:
Clear Delta	True

(default) Shift-Alt-S → object.scale_clear : **KEYBOARD** → PRESS
Clear Scale

```
bpy.ops.object.scale_clear(clear_delta=False)
```

Properties:	Values:
Clear Delta	True

(default) Alt-O → object.origin_clear : **KEYBOARD** → PRESS
Clear Origin

```
bpy.ops.object.origin_clear()
```

(default) Alt-H → object.hide_view_clear : **KEYBOARD** → PRESS
Clear Restrict View

```
bpy.ops.object.hide_view_clear()
```

(default) H → object.hide_view_set : **KEYBOARD** → PRESS
Set Restrict View

```
bpy.ops.object.hide_view_set(unselected=False)
```

Properties:	Values:
Unselected	False

(default) Shift-H → object.hide_view_set : **KEYBOARD** → PRESS
Set Restrict View

```
bpy.ops.object.hide_view_set(unselected=False)
```

Properties:	Values:
Unselected	True

(default) Ctrl-Alt-H → object.hide_render_clear : **KEYBOARD** → PRESS
Clear Restrict Render

```
bpy.ops.object.hide_render_clear()
```

(default) Ctrl-H → object.hide_render_set : **KEYBOARD** → PRESS
Set Restrict Render

```
bpy.ops.object.hide_render_set(unselected=False)
```

(default) M → object.move_to_layer : **KEYBOARD** → PRESS
Move to Layer

```
bpy.ops.object.move_to_layer(layers=(False, False, False, False, False, False, False, False, False, False, False, False, False, False, False, False, False, False, False, False))
```

(default) X → object.delete : **KEYBOARD** → PRESS
Delete

bpy.ops.object.delete(use_global=False)

Properties:	Values:
Delete Globally	False

(default) Shift-X → object.delete : **KEYBOARD** → PRESS

Delete

bpy.ops.object.delete(use_global=False)

Properties:	Values:
Delete Globally	True

(default) DEL → object.delete : **KEYBOARD** → PRESS

Delete

bpy.ops.object.delete(use_global=False)

Properties:	Values:
Delete Globally	False

(default) Shift-DEL → object.delete : **KEYBOARD** → PRESS

Delete

bpy.ops.object.delete(use_global=False)

Properties:	Values:
Delete Globally	True

(default) Shift-A → wm.call_menu : **KEYBOARD** → PRESS

Call Menu

bpy.ops.wm.call_menu(name='')

Properties:	Values:
Name	INFO_MT_add

(default) Ctrl-Shift-A → object.duplicates_make_real : **KEYBOARD** → PRESS

Make Duplicates Real

bpy.ops.object.duplicates_make_real(use_base_parent=False, use_hierarchy=False)

(default) Ctrl-A → wm.call_menu : **KEYBOARD** → PRESS

Call Menu

bpy.ops.wm.call_menu(name='')

Properties:	Values:
Name	VIEW3D_MT_object_apply

(default) U → wm.call_menu : **KEYBOARD** → PRESS

Call Menu

bpy.ops.wm.call_menu(name='')

Properties:	Values:
Name	VIEW3D_MT_make_single_user

(default) Ctrl-L → wm.call_menu : **KEYBOARD** → PRESS

Call Menu

bpy.ops.wm.call_menu(name='')

Properties:	Values:
Name	VIEW3D_MT_make_links

(default) Shift-D → object.duplicate_move : **KEYBOARD** → PRESS

Duplicate Objects

```
bpy.ops.object.duplicate_move(OBJECT_OT_duplicate={"linked":False, "mode":'TRANSLATION'},
TRANSFORM_OT_translate={"value":(0, 0, 0), "constraint_axis":(False, False, False),
"constraint_orientation":'GLOBAL', "mirror":False, "proportional":'DISABLED', "proportional_edit_falloff":'SMOOTH', "proportional_size":1, "snap":False, "snap_target":'CLOSEST',
"snap_point":(0, 0, 0), "snap_align":False, "snap_normal":(0, 0, 0), "gpencil_strokes":False, "texture_space":False, "remove_on_cancel":False, "release_confirm":False})
```

Properties:	Values:
Duplicate Objects	N/A
Translate	N/A

(default) Alt-D → object.duplicate_move_linked : **KEYBOARD** → PRESS

Duplicate Linked

```
bpy.ops.object.duplicate_move_linked(OBJECT_OT_duplicate={"linked":False, "mode":'TRANSLATION'},
TRANSFORM_OT_translate={"value":(0, 0, 0), "constraint_axis":(False, False, False),
"constraint_orientation":'GLOBAL', "mirror":False, "proportional":'DISABLED', "proportional_edit_falloff":'SMOOTH', "proportional_size":1, "snap":False, "snap_target":'CLOSEST',
"snap_point":(0, 0, 0), "snap_align":False, "snap_normal":(0, 0, 0), "gpencil_strokes":False, "texture_space":False, "remove_on_cancel":False, "release_confirm":False})
```

Properties:	Values:
Duplicate Objects	N/A
Translate	N/A

(default) Ctrl-J → object.join : **KEYBOARD** → PRESS

Join

```
bpy.ops.object.join()
```

(default) Alt-C → object.convert : **KEYBOARD** → PRESS

Convert to

```
bpy.ops.object.convert(target='MESH', keep_original=False)
```

(default) Ctrl-Alt-P → object.proxy_make : **KEYBOARD** → PRESS

Make Proxy

```
bpy.ops.object.proxy_make(object='DEFAULT')
```

(default) L → object.make_local : **KEYBOARD** → PRESS

Make Local

```
bpy.ops.object.make_local(type='SELECT_OBJECT')
```

(default) I → anim.keyframe_insert_menu : **KEYBOARD** → PRESS

Insert Keyframe Menu

```
bpy.ops.anim.keyframe_insert_menu(type='DEFAULT', confirm_success=False, always_prompt=False)
```

(default) Alt-I → anim.keyframe_delete_v3d : **KEYBOARD** → PRESS

Delete Keyframe

```
bpy.ops.anim.keyframe_delete_v3d()
```

(default) Ctrl-Shift-Alt-I → anim.keying_set_active_set : **KEYBOARD** → PRESS

Set Active Keying Set

```
bpy.ops.anim.keying_set_active_set(type='DEFAULT')
```

(default) Ctrl-G → group.create : **KEYBOARD** → PRESS

Create New Group

```
bpy.ops.group.create(name="Group")
```

(default) Ctrl-Alt-G → group.objects_remove : **KEYBOARD** → PRESS

Remove From Group

```
bpy.ops.group.objects_remove(group='<UNKNOWN ENUM>')
```

(default) Ctrl-Shift-Alt-G → group.objects_remove_all : **KEYBOARD** → PRESS

Remove From All Groups

```
bpy.ops.group.objects_remove_all()
```

(default) Ctrl-Shift-G → group.objects_add_active : **KEYBOARD** → PRESS

Add Selected To Active Group

```
bpy.ops.group.objects_add_active(group='<UNKNOWN ENUM>')
```

(default) Shift-Alt-G → group.objects_remove_active : **KEYBOARD** → PRESS

Remove Selected From Active Group

```
bpy.ops.group.objects_remove_active(group='<UNKNOWN ENUM>')
```

(default) W → wm.call_menu : **KEYBOARD** → PRESS

Call Menu

```
bpy.ops.wm.call_menu(name="")
```

Properties:	Values:
Name	VIEW3D_MT_object_specials

(default) Ctrl-Shift-T → object.data_transfer : **KEYBOARD** → PRESS

Transfer Mesh Data

```
bpy.ops.object.data_transfer(use_reverse_transfer=False, use_freeze=False, data_type='<UNKNOWN ENUM>', use_create=True, vert_mapping='NEAREST', edge_mapping='NEAREST', loop_mapping='NEAREST_POLYNOR', poly_mapping='NEAREST', use_auto_transform=False, use_object_transform=True, use_max_distance=False, max_distance=1, ray_radius=0, islands_precision=0.1, layers_select_src='ACTIVE', layers_select_dst='ACTIVE', mix_mode='REPLACE', mix_factor=1)
```

(default) Ctrl-0 → object.subdivision_set : **KEYBOARD** → PRESS

Subdivision Set

```
bpy.ops.object.subdivision_set(level=1, relative=False)
```

Properties:	Values:
Level	0

(default) Ctrl-1 → object.subdivision_set : **KEYBOARD** → PRESS

Subdivision Set

```
bpy.ops.object.subdivision_set(level=1, relative=False)
```

Properties:	Values:
Level	1

(default) Ctrl-2 → object.subdivision_set : **KEYBOARD** → PRESS

Subdivision Set

```
bpy.ops.object.subdivision_set(level=1, relative=False)
```

Properties:	Values:
Level	2

(default) Ctrl-3 → object.subdivision_set : **KEYBOARD** → PRESS
Subdivision Set

bpy.ops.object.subdivision_set(level=1, relative=False)

Properties:	Values:
Level	3

(default) Ctrl-4 → object.subdivision_set : **KEYBOARD** → PRESS
Subdivision Set

bpy.ops.object.subdivision_set(level=1, relative=False)

Properties:	Values:
Level	4

(default) Ctrl-5 → object.subdivision_set : **KEYBOARD** → PRESS
Subdivision Set

bpy.ops.object.subdivision_set(level=1, relative=False)

Properties:	Values:
Level	5

Outliner

Quick Reference

Hotkey	Operator
<i>Ctrl-A</i>	<code>bpy.ops.outliner.selected_toggle()</code>
<i>LEFTMOUSE</i>	<code>bpy.ops.outliner.item_rename()</code>
<i>LEFTMOUSE</i>	<code>bpy.ops.outliner.item_activate()</code>
<i>Shift-LEFTMOUSE</i>	<code>bpy.ops.outliner.item_activate()</code>
<i>Ctrl-LEFTMOUSE</i>	<code>bpy.ops.outliner.item_activate()</code>
<i>Ctrl-Shift-LEFTMOUSE</i>	<code>bpy.ops.outliner.item_activate()</code>
<i>B</i>	<code>bpy.ops.outliner.select_border()</code>
<i>RET</i>	<code>bpy.ops.outliner.item_openclose()</code>
<i>Shift-RET</i>	<code>bpy.ops.outliner.item_openclose()</code>
<i>Ctrl-LEFTMOUSE</i>	<code>bpy.ops.outliner.item_rename()</code>
<i>RIGHTMOUSE</i>	<code>bpy.ops.outliner.operation()</code>
<i>HOME</i>	<code>bpy.ops.outliner.show_hierarchy()</code>
<i>.</i>	<code>bpy.ops.outliner.show_active()</code>
<i>NUMPAD_PERIOD</i>	<code>bpy.ops.outliner.show_active()</code>
<i>PAGE_DOWN</i>	<code>bpy.ops.outliner.scroll_page()</code>
<i>PAGE_UP</i>	<code>bpy.ops.outliner.scroll_page()</code>
<i>NUMPAD_PLUS</i>	<code>bpy.ops.outliner.show_one_level()</code>
<i>NUMPAD_MINUS</i>	<code>bpy.ops.outliner.show_one_level()</code>
<i>A</i>	<code>bpy.ops.outliner.selected_toggle()</code>
<i>Shift-A</i>	<code>bpy.ops.outliner.expanded_toggle()</code>
<i>R</i>	<code>bpy.ops.outliner.renderability_toggle()</code>
<i>S</i>	<code>bpy.ops.outliner.selectability_toggle()</code>
<i>V</i>	<code>bpy.ops.outliner.visibility_toggle()</code>
<i>K</i>	<code>bpy.ops.outliner.keyingset_add_selected()</code>
<i>Alt-K</i>	<code>bpy.ops.outliner.keyingset_remove_selected()</code>
<i>I</i>	<code>bpy.ops.anim.keyframe_insert()</code>
<i>Alt-I</i>	<code>bpy.ops.anim.keyframe_delete()</code>
<i>D</i>	<code>bpy.ops.outliner.drivers_add_selected()</code>
<i>Alt-D</i>	<code>bpy.ops.outliner.drivers_delete_selected()</code>

Detailed Reference

Note: Hotkeys marked with the “(default)” prefix are inherited from the default blender keymap

Ctrl-A → `outliner.selected_toggle` : **KEYBOARD** → PRESS

Toggle Selected

`bpy.ops.outliner.selected_toggle()`

(default) LEFTMOUSE → `outliner.item_rename` : **MOUSE** → DOUBLE_CLICK

Rename Item

`bpy.ops.outliner.item_rename()`

(default) LEFTMOUSE → `outliner.item_activate` : **MOUSE** → CLICK

Activate Item

bpy.ops.outliner.item_activate(extend=True, recursive=False)

Properties:	Values:
Recursive	False
Extend	False

(default) Shift-LEFTMOUSE → outliner.item_activate : **MOUSE** → CLICK

Activate Item

bpy.ops.outliner.item_activate(extend=True, recursive=False)

Properties:	Values:
Recursive	False
Extend	True

(default) Ctrl-LEFTMOUSE → outliner.item_activate : **MOUSE** → CLICK

Activate Item

bpy.ops.outliner.item_activate(extend=True, recursive=False)

Properties:	Values:
Recursive	True
Extend	False

(default) Ctrl-Shift-LEFTMOUSE → outliner.item_activate : **MOUSE** → CLICK

Activate Item

bpy.ops.outliner.item_activate(extend=True, recursive=False)

Properties:	Values:
Recursive	True
Extend	True

(default) B → outliner.select_border : **KEYBOARD** → PRESS

Border Select

bpy.ops.outliner.select_border(gesture_mode=0, xmin=0, xmax=0, ymin=0, ymax=0)

(default) RET → outliner.item_openclose : **KEYBOARD** → PRESS

Open/Close Item

bpy.ops.outliner.item_openclose(all=True)

Properties:	Values:
All	False

(default) Shift-RET → outliner.item_openclose : **KEYBOARD** → PRESS

Open/Close Item

bpy.ops.outliner.item_openclose(all=True)

Properties:	Values:
All	True

(default) Ctrl-LEFTMOUSE → outliner.item_rename : **MOUSE** → PRESS

Rename Item

bpy.ops.outliner.item_rename()

(default) RIGHTMOUSE → outliner.operation : **MOUSE** → PRESS

Execute Operation

bpy.ops.outliner.operation()

(default) HOME → `outliner.show_hierarchy` : **KEYBOARD** → PRESS
Show Hierarchy

`bpy.ops.outliner.show_hierarchy()`

(default) . → `outliner.show_active` : **KEYBOARD** → PRESS
Show Active

`bpy.ops.outliner.show_active()`

(default) NUMPAD_PERIOD → `outliner.show_active` : **KEYBOARD** → PRESS
Show Active

`bpy.ops.outliner.show_active()`

(default) PAGE_DOWN → `outliner.scroll_page` : **KEYBOARD** → PRESS
Scroll Page

`bpy.ops.outliner.scroll_page(up=False)`

Properties:	Values:
Up	False

(default) PAGE_UP → `outliner.scroll_page` : **KEYBOARD** → PRESS
Scroll Page

`bpy.ops.outliner.scroll_page(up=False)`

Properties:	Values:
Up	True

(default) NUMPAD_PLUS → `outliner.show_one_level` : **KEYBOARD** → PRESS
Show/Hide One Level

`bpy.ops.outliner.show_one_level(open=True)`

(default) NUMPAD_MINUS → `outliner.show_one_level` : **KEYBOARD** → PRESS
Show/Hide One Level

`bpy.ops.outliner.show_one_level(open=True)`

Properties:	Values:
Open	False

(default) A → `outliner.selected_toggle` : **KEYBOARD** → PRESS
Toggle Selected

`bpy.ops.outliner.selected_toggle()`

(default) Shift-A → `outliner.expanded_toggle` : **KEYBOARD** → PRESS
Expand/Collapse All

`bpy.ops.outliner.expanded_toggle()`

(default) R → `outliner.renderability_toggle` : **KEYBOARD** → PRESS
Toggle Renderability

`bpy.ops.outliner.renderability_toggle()`

(default) S → `outliner.selectability_toggle` : **KEYBOARD** → PRESS
Toggle Selectability

`bpy.ops.outliner.selectability_toggle()`

(default) V → `outliner.visibility_toggle` : **KEYBOARD** → PRESS

Toggle Visibility

`bpy.ops.outliner.visibility_toggle()`

(default) K → `outliner.keyingset_add_selected` : **KEYBOARD** → PRESS

Keying Set Add Selected

`bpy.ops.outliner.keyingset_add_selected()`

(default) Alt-K → `outliner.keyingset_remove_selected` : **KEYBOARD** → PRESS

Keying Set Remove Selected

`bpy.ops.outliner.keyingset_remove_selected()`

(default) I → `anim.keyframe_insert` : **KEYBOARD** → PRESS

Insert Keyframe

`bpy.ops.anim.keyframe_insert(type='DEFAULT', confirm_success=True)`

(default) Alt-I → `anim.keyframe_delete` : **KEYBOARD** → PRESS

Delete Keying-Set Keyframe

`bpy.ops.anim.keyframe_delete(type='DEFAULT', confirm_success=True)`

(default) D → `outliner.drivers_add_selected` : **KEYBOARD** → PRESS

Add Drivers for Selected

`bpy.ops.outliner.drivers_add_selected()`

(default) Alt-D → `outliner.drivers_delete_selected` : **KEYBOARD** → PRESS

Delete Drivers for Selected

`bpy.ops.outliner.drivers_delete_selected()`

Particle

Quick Reference

Hotkey	Operator
<i>Ctrl-A</i>	<code>bpy.ops.particle.select_all()</code>
<i>A</i>	<code>bpy.ops.particle.select_all()</code>
<i>Ctrl-I</i>	<code>bpy.ops.particle.select_all()</code>
<i>Ctrl-NUMPAD_PLUS</i>	<code>bpy.ops.particle.select_more()</code>
<i>Ctrl-NUMPAD_MINUS</i>	<code>bpy.ops.particle.select_less()</code>
<i>L</i>	<code>bpy.ops.particle.select_linked()</code>
<i>Shift-L</i>	<code>bpy.ops.particle.select_linked()</code>
<i>X</i>	<code>bpy.ops.particle.delete()</code>
<i>DEL</i>	<code>bpy.ops.particle.delete()</code>
<i>Alt-H</i>	<code>bpy.ops.particle.reveal()</code>
<i>H</i>	<code>bpy.ops.particle.hide()</code>
<i>Shift-H</i>	<code>bpy.ops.particle.hide()</code>
<i>Any-LEFTMOUSE</i>	<code>bpy.ops.view3d.manipulator()</code>
<i>LEFTMOUSE</i>	<code>bpy.ops.particle.brush_edit()</code>
<i>Shift-LEFTMOUSE</i>	<code>bpy.ops.particle.brush_edit()</code>
<i>F</i>	<code>bpy.ops.wm.radial_control()</code>
<i>Shift-F</i>	<code>bpy.ops.wm.radial_control()</code>
<i>W</i>	<code>bpy.ops.wm.call_menu()</code>
<i>Shift-K</i>	<code>bpy.ops.particle.weight_set()</code>
<i>Shift-O</i>	<code>bpy.ops.wm.context_cycle_enum()</code>
<i>O</i>	<code>bpy.ops.wm.context_toggle_enum()</code>

Detailed Reference

Note: Hotkeys marked with the “(default)” prefix are inherited from the default blender keymap

Ctrl-A → `particle.select_all` : **KEYBOARD** → PRESS

(De)select All

`bpy.ops.particle.select_all(action='TOGGLE')`

Properties:	Values:
Action	TOGGLE

(default) A → `particle.select_all` : **KEYBOARD** → PRESS

(De)select All

`bpy.ops.particle.select_all(action='TOGGLE')`

Properties:	Values:
Action	TOGGLE

(default) Ctrl-I → `particle.select_all` : **KEYBOARD** → PRESS

(De)select All

`bpy.ops.particle.select_all(action='TOGGLE')`

Properties:	Values:
Action	INVERT

(default) Ctrl-NUMPAD_PLUS → particle.select_more : **KEYBOARD** → PRESS
Select More

bpy.ops.particle.select_more()

(default) Ctrl-NUMPAD_MINUS → particle.select_less : **KEYBOARD** → PRESS
Select Less

bpy.ops.particle.select_less()

(default) L → particle.select_linked : **KEYBOARD** → PRESS
Select Linked

bpy.ops.particle.select_linked(deselect=False, location=(0, 0))

Properties:	Values:
Deselect	False

(default) Shift-L → particle.select_linked : **KEYBOARD** → PRESS
Select Linked

bpy.ops.particle.select_linked(deselect=False, location=(0, 0))

Properties:	Values:
Deselect	True

(default) X → particle.delete : **KEYBOARD** → PRESS
Delete

bpy.ops.particle.delete(type='PARTICLE')

(default) DEL → particle.delete : **KEYBOARD** → PRESS
Delete

bpy.ops.particle.delete(type='PARTICLE')

(default) Alt-H → particle.reveal : **KEYBOARD** → PRESS
Reveal

bpy.ops.particle.reveal()

(default) H → particle.hide : **KEYBOARD** → PRESS
Hide Selected

bpy.ops.particle.hide(unselected=False)

Properties:	Values:
Unselected	False

(default) Shift-H → particle.hide : **KEYBOARD** → PRESS
Hide Selected

bpy.ops.particle.hide(unselected=False)

Properties:	Values:
Unselected	True

(default) Any-LEFTMOUSE → view3d.manipulator : **MOUSE** → PRESS
3D Manipulator

bpy.ops.view3d.manipulator(constraint_axis=(False, False, False), constraint_orientation='GLOBAL', release_confirm=False)

Properties:	Values:
Confirm on Release	True

(default) LEFTMOUSE → particle.brush_edit : **MOUSE** → PRESS

Brush Edit

bpy.ops.particle.brush_edit(stroke=[])

(default) Shift-LEFTMOUSE → particle.brush_edit : **MOUSE** → PRESS

Brush Edit

bpy.ops.particle.brush_edit(stroke=[])

(default) F → wm.radial_control : **KEYBOARD** → PRESS

Radial Control

bpy.ops.wm.radial_control(data_path_primary="", data_path_secondary="", use_secondary="", rotation_path="", color_path="", fill_color_path="", fill_color_override_path="", fill_color_override_test_path="", zoom_path="", image_id="", secondary_tex=False)

Properties:	Values:
Primary Data Path	tool_settings.particle_edit.brush.size

(default) Shift-F → wm.radial_control : **KEYBOARD** → PRESS

Radial Control

bpy.ops.wm.radial_control(data_path_primary="", data_path_secondary="", use_secondary="", rotation_path="", color_path="", fill_color_path="", fill_color_override_path="", fill_color_override_test_path="", zoom_path="", image_id="", secondary_tex=False)

Properties:	Values:
Primary Data Path	tool_settings.particle_edit.brush.strength

(default) W → wm.call_menu : **KEYBOARD** → PRESS

Call Menu

bpy.ops.wm.call_menu(name="")

Properties:	Values:
Name	VIEW3D_MT_particle_specials

(default) Shift-K → particle.weight_set : **KEYBOARD** → PRESS

Weight Set

bpy.ops.particle.weight_set(factor=1)

(default) Shift-O → wm.context_cycle_enum : **KEYBOARD** → PRESS

Context Enum Cycle

bpy.ops.wm.context_cycle_enum(data_path="", reverse=False, wrap=False)

Properties:	Values:
Context Attributes	tool_settings.proportional_edit_falloff
Wrap	True

(default) O → wm.context_toggle_enum : **KEYBOARD** → PRESS

Context Toggle Values

bpy.ops.wm.context_toggle_enum(data_path="", value_1="", value_2="")

Properties:	Values:
Context Attributes	tool_settings.proportional_edit
Value	DISABLED
Value	ENABLED

Pose

Quick Reference

Hotkey	Operator
<i>A</i>	<code>bpy.ops.wm.call_menu()</code>
<i>Ctrl-A</i>	<code>bpy.ops.pose.select_all()</code>
<i>Ctrl-P</i>	<code>bpy.ops.object.parent_set()</code>
<i>Shift-A</i>	<code>bpy.ops.wm.call_menu()</code>
<i>H</i>	<code>bpy.ops.pose.hide()</code>
<i>Shift-H</i>	<code>bpy.ops.pose.hide()</code>
<i>Alt-H</i>	<code>bpy.ops.pose.reveal()</code>
<i>Ctrl-A</i>	<code>bpy.ops.wm.call_menu()</code>
<i>Alt-R</i>	<code>bpy.ops.pose.rot_clear()</code>
<i>Alt-G</i>	<code>bpy.ops.pose.loc_clear()</code>
<i>Alt-S</i>	<code>bpy.ops.pose.scale_clear()</code>
<i>Alt-F</i>	<code>bpy.ops.pose.quaternions_flip()</code>
<i>Ctrl-R</i>	<code>bpy.ops.pose.rotation_mode_set()</code>
<i>Ctrl-C</i>	<code>bpy.ops.pose.copy()</code>
<i>Ctrl-V</i>	<code>bpy.ops.pose.paste()</code>
<i>Ctrl-Shift-V</i>	<code>bpy.ops.pose.paste()</code>
<i>A</i>	<code>bpy.ops.pose.select_all()</code>
<i>Ctrl-I</i>	<code>bpy.ops.pose.select_all()</code>
<i>Shift-P</i>	<code>bpy.ops.pose.select_parent()</code>
<i>LEFT_BRACKET</i>	<code>bpy.ops.pose.select_hierarchy()</code>
<i>Shift-LEFT_BRACKET</i>	<code>bpy.ops.pose.select_hierarchy()</code>
<i>RIGHT_BRACKET</i>	<code>bpy.ops.pose.select_hierarchy()</code>
<i>Shift-RIGHT_BRACKET</i>	<code>bpy.ops.pose.select_hierarchy()</code>
<i>L</i>	<code>bpy.ops.pose.select_linked()</code>
<i>Shift-G</i>	<code>bpy.ops.pose.select_grouped()</code>
<i>Ctrl-Shift-F</i>	<code>bpy.ops.pose.select_mirror()</code>
<i>Ctrl-Shift-C</i>	<code>bpy.ops.pose.constraint_add_with_targets()</code>
<i>Ctrl-Alt-C</i>	<code>bpy.ops.pose.constraints_clear()</code>
<i>Shift-I</i>	<code>bpy.ops.pose.ik_add()</code>
<i>Ctrl-Alt-I</i>	<code>bpy.ops.pose.ik_clear()</code>
<i>Ctrl-G</i>	<code>bpy.ops.wm.call_menu()</code>
<i>Shift-W</i>	<code>bpy.ops.wm.call_menu()</code>
<i>Ctrl-Shift-W</i>	<code>bpy.ops.wm.call_menu()</code>
<i>Alt-W</i>	<code>bpy.ops.wm.call_menu()</code>
<i>Ctrl-ACCENT_GRAVE</i>	<code>bpy.ops.armature.layers_show_all()</code>
<i>Shift-M</i>	<code>bpy.ops.armature.armature_layers()</code>
<i>M</i>	<code>bpy.ops.pose.bone_layers()</code>
<i>Ctrl-Alt-S</i>	<code>bpy.ops.transform.transform()</code>
<i>I</i>	<code>bpy.ops.anim.keyframe_insert_menu()</code>
<i>Alt-I</i>	<code>bpy.ops.anim.keyframe_delete_v3d()</code>
<i>Ctrl-Shift-Alt-I</i>	<code>bpy.ops.anim.keying_set_active_set()</code>
<i>Ctrl-L</i>	<code>bpy.ops.poselib.browse_interactive()</code>
<i>Shift-L</i>	<code>bpy.ops.poselib.pose_add()</code>
<i>Alt-L</i>	<code>bpy.ops.poselib.pose_remove()</code>
<i>Ctrl-Shift-L</i>	<code>bpy.ops.poselib.pose_rename()</code>

Continued on next page

Table 2.15 – continued from previous page

Hotkey	Operator
<i>Ctrl-E</i>	<code>bpy.ops.pose.push()</code>
<i>Alt-E</i>	<code>bpy.ops.pose.relax()</code>
<i>Shift-E</i>	<code>bpy.ops.pose.breakdown()</code>
<i>W</i>	<code>bpy.ops.wm.call_menu()</code>
<i>Alt-P</i>	<code>bpy.ops.wm.call_menu()</code>

Detailed Reference

Note: Hotkeys marked with the “(default)” prefix are inherited from the default blender keymap

A → `wm.call_menu` : **KEYBOARD** → PRESS

Call Menu

`bpy.ops.wm.call_menu(name='')`

Properties:	Values:
Name	VIEW3D_MT_pose_apply

Ctrl-A → `pose.select_all` : **KEYBOARD** → PRESS

(De)select All

`bpy.ops.pose.select_all(action='TOGGLE')`

Properties:	Values:
Action	TOGGLE

(default) Ctrl-P → `object.parent_set` : **KEYBOARD** → PRESS

Make Parent

`bpy.ops.object.parent_set(type='OBJECT', xmirror=False, keep_transform=False)`

(default) Shift-A → `wm.call_menu` : **KEYBOARD** → PRESS

Call Menu

`bpy.ops.wm.call_menu(name='')`

Properties:	Values:
Name	INFO_MT_add

(default) H → `pose.hide` : **KEYBOARD** → PRESS

Hide Selected

`bpy.ops.pose.hide(unselected=False)`

Properties:	Values:
Unselected	False

(default) Shift-H → `pose.hide` : **KEYBOARD** → PRESS

Hide Selected

`bpy.ops.pose.hide(unselected=False)`

Properties:	Values:
Unselected	True

(default) Alt-H → `pose.reveal` : **KEYBOARD** → PRESS

Reveal Selected

```
bpy.ops.pose.reveal()
```

(default) Ctrl-A → **wm.call_menu : KEYBOARD** → PRESS
Call Menu

```
bpy.ops.wm.call_menu(name='')
```

Properties:	Values:
Name	VIEW3D_MT_pose_apply

(default) Alt-R → **pose.rot_clear : KEYBOARD** → PRESS
Clear Pose Rotation

```
bpy.ops.pose.rot_clear()
```

(default) Alt-G → **pose.loc_clear : KEYBOARD** → PRESS
Clear Pose Location

```
bpy.ops.pose.loc_clear()
```

(default) Alt-S → **pose.scale_clear : KEYBOARD** → PRESS
Clear Pose Scale

```
bpy.ops.pose.scale_clear()
```

(default) Alt-F → **pose.quaternions_flip : KEYBOARD** → PRESS
Flip Quats

```
bpy.ops.pose.quaternions_flip()
```

(default) Ctrl-R → **pose.rotation_mode_set : KEYBOARD** → PRESS
Set Rotation Mode

```
bpy.ops.pose.rotation_mode_set(type='QUATERNION')
```

(default) Ctrl-C → **pose.copy : KEYBOARD** → PRESS
Copy Pose

```
bpy.ops.pose.copy()
```

(default) Ctrl-V → **pose.paste : KEYBOARD** → PRESS
Paste Pose

```
bpy.ops.pose.paste(flipped=False, selected_mask=False)
```

Properties:	Values:
Flipped on X-Axis	False

(default) Ctrl-Shift-V → **pose.paste : KEYBOARD** → PRESS
Paste Pose

```
bpy.ops.pose.paste(flipped=False, selected_mask=False)
```

Properties:	Values:
Flipped on X-Axis	True

(default) A → **pose.select_all : KEYBOARD** → PRESS
(De)select All

```
bpy.ops.pose.select_all(action='TOGGLE')
```

Properties:	Values:
Action	TOGGLE

(default) **Ctrl-I** → pose.select_all : **KEYBOARD** → PRESS

(De)select All

bpy.ops.pose.select_all(action='TOGGLE')

Properties:	Values:
Action	INVERT

(default) **Shift-P** → pose.select_parent : **KEYBOARD** → PRESS

Select Parent Bone

bpy.ops.pose.select_parent()

(default) **LEFT_BRACKET** → pose.select_hierarchy : **KEYBOARD** → PRESS

Select Hierarchy

bpy.ops.pose.select_hierarchy(direction='PARENT', extend=False)

Properties:	Values:
Direction	PARENT
Extend	False

(default) **Shift-LEFT_BRACKET** → pose.select_hierarchy : **KEYBOARD** → PRESS

Select Hierarchy

bpy.ops.pose.select_hierarchy(direction='PARENT', extend=False)

Properties:	Values:
Direction	PARENT
Extend	True

(default) **RIGHT_BRACKET** → pose.select_hierarchy : **KEYBOARD** → PRESS

Select Hierarchy

bpy.ops.pose.select_hierarchy(direction='PARENT', extend=False)

Properties:	Values:
Direction	CHILD
Extend	False

(default) **Shift-RIGHT_BRACKET** → pose.select_hierarchy : **KEYBOARD** → PRESS

Select Hierarchy

bpy.ops.pose.select_hierarchy(direction='PARENT', extend=False)

Properties:	Values:
Direction	CHILD
Extend	True

(default) **L** → pose.select_linked : **KEYBOARD** → PRESS

Select Connected

bpy.ops.pose.select_linked(extend=False)

(default) **Shift-G** → pose.select_grouped : **KEYBOARD** → PRESS

Select Grouped

bpy.ops.pose.select_grouped(extend=False, type='LAYER')

(default) **Ctrl-Shift-F** → pose.select_mirror : **KEYBOARD** → PRESS

Flip Active/Selected Bone

bpy.ops.pose.select_mirror(only_active=False, extend=False)

(default) **Shift-E** → pose.breakdown : **KEYBOARD** → PRESS

Pose Breakdowner

```
bpy.ops.pose.breakdown(prev_frame=0, next_frame=0, percentage=0.5)
```

(default) **W** → wm.call_menu : **KEYBOARD** → PRESS

Call Menu

```
bpy.ops.wm.call_menu(name='')
```

Properties:	Values:
Name	VIEW3D_MT_pose_specials

(default) **Alt-P** → wm.call_menu : **KEYBOARD** → PRESS

Call Menu

```
bpy.ops.wm.call_menu(name='')
```

Properties:	Values:
Name	VIEW3D_MT_pose_propagate

Sequencer

Quick Reference

Hotkey	Operator
<i>Ctrl-A</i>	<code>bpy.ops.sequencer.select_all()</code>
<i>A</i>	<code>bpy.ops.sequencer.select_all()</code>
<i>Ctrl-I</i>	<code>bpy.ops.sequencer.select_all()</code>
<i>K</i>	<code>bpy.ops.sequencer.cut()</code>
<i>Shift-K</i>	<code>bpy.ops.sequencer.cut()</code>
<i>H</i>	<code>bpy.ops.sequencer.mute()</code>
<i>Shift-H</i>	<code>bpy.ops.sequencer.mute()</code>
<i>Alt-H</i>	<code>bpy.ops.sequencer.unmute()</code>
<i>Shift-Alt-H</i>	<code>bpy.ops.sequencer.unmute()</code>
<i>Shift-L</i>	<code>bpy.ops.sequencer.lock()</code>
<i>Shift-Alt-L</i>	<code>bpy.ops.sequencer.unlock()</code>
<i>R</i>	<code>bpy.ops.sequencer.reassign_inputs()</code>
<i>Alt-R</i>	<code>bpy.ops.sequencer.reload()</code>
<i>Shift-Alt-R</i>	<code>bpy.ops.sequencer.reload()</code>
<i>Alt-O</i>	<code>bpy.ops.sequencer.offset_clear()</code>
<i>Shift-D</i>	<code>bpy.ops.sequencer.duplicate_move()</code>
<i>X</i>	<code>bpy.ops.sequencer.delete()</code>
<i>DEL</i>	<code>bpy.ops.sequencer.delete()</code>
<i>Ctrl-C</i>	<code>bpy.ops.sequencer.copy()</code>
<i>Ctrl-V</i>	<code>bpy.ops.sequencer.paste()</code>
<i>Y</i>	<code>bpy.ops.sequencer.images_separate()</code>
<i>Tab</i>	<code>bpy.ops.sequencer.meta_toggle()</code>
<i>Ctrl-G</i>	<code>bpy.ops.sequencer.meta_make()</code>
<i>Alt-G</i>	<code>bpy.ops.sequencer.meta_separate()</code>
<i>HOME</i>	<code>bpy.ops.sequencer.view_all()</code>
<i>NDOF_BUTTON_FIT</i>	<code>bpy.ops.sequencer.view_all()</code>
<i>NUMPAD_PERIOD</i>	<code>bpy.ops.sequencer.view_selected()</code>

Continued on next page

Table 2.16 – continued from previous page

Hotkey	Operator
<i>NUMPAD_0</i>	<code>bpy.ops.sequencer.view_frame()</code>
<i>PAGE_UP</i>	<code>bpy.ops.sequencer.strip_jump()</code>
<i>PAGE_DOWN</i>	<code>bpy.ops.sequencer.strip_jump()</code>
<i>Alt-PAGE_UP</i>	<code>bpy.ops.sequencer.strip_jump()</code>
<i>Alt-PAGE_DOWN</i>	<code>bpy.ops.sequencer.strip_jump()</code>
<i>Alt-LEFT_ARROW</i>	<code>bpy.ops.sequencer.swap()</code>
<i>Alt-RIGHT_ARROW</i>	<code>bpy.ops.sequencer.swap()</code>
<i>BACK_SPACE</i>	<code>bpy.ops.sequencer.gap_remove()</code>
<i>Shift-BACK_SPACE</i>	<code>bpy.ops.sequencer.gap_remove()</code>
<i>Shift-EQUAL</i>	<code>bpy.ops.sequencer.gap_insert()</code>
<i>Shift-S</i>	<code>bpy.ops.sequencer.snap()</code>
<i>Alt-S</i>	<code>bpy.ops.sequencer.swap_inputs()</code>
<i>1</i>	<code>bpy.ops.sequencer.cut_multicam()</code>
<i>2</i>	<code>bpy.ops.sequencer.cut_multicam()</code>
<i>3</i>	<code>bpy.ops.sequencer.cut_multicam()</code>
<i>4</i>	<code>bpy.ops.sequencer.cut_multicam()</code>
<i>5</i>	<code>bpy.ops.sequencer.cut_multicam()</code>
<i>6</i>	<code>bpy.ops.sequencer.cut_multicam()</code>
<i>7</i>	<code>bpy.ops.sequencer.cut_multicam()</code>
<i>8</i>	<code>bpy.ops.sequencer.cut_multicam()</code>
<i>9</i>	<code>bpy.ops.sequencer.cut_multicam()</code>
<i>0</i>	<code>bpy.ops.sequencer.cut_multicam()</code>
<i>SELECTMOUSE</i>	<code>bpy.ops.sequencer.select()</code>
<i>Shift-SELECTMOUSE</i>	<code>bpy.ops.sequencer.select()</code>
<i>Alt-SELECTMOUSE</i>	<code>bpy.ops.sequencer.select()</code>
<i>Shift-Alt-SELECTMOUSE</i>	<code>bpy.ops.sequencer.select()</code>
<i>Ctrl-SELECTMOUSE</i>	<code>bpy.ops.sequencer.select()</code>
<i>Ctrl-Shift-SELECTMOUSE</i>	<code>bpy.ops.sequencer.select()</code>
<i>Ctrl-NUMPAD_PLUS</i>	<code>bpy.ops.sequencer.select_more()</code>
<i>Ctrl-NUMPAD_MINUS</i>	<code>bpy.ops.sequencer.select_less()</code>
<i>L</i>	<code>bpy.ops.sequencer.select_linked_pick()</code>
<i>Shift-L</i>	<code>bpy.ops.sequencer.select_linked_pick()</code>
<i>Ctrl-L</i>	<code>bpy.ops.sequencer.select_linked()</code>
<i>B</i>	<code>bpy.ops.sequencer.select_border()</code>
<i>Shift-G</i>	<code>bpy.ops.sequencer.select_grouped()</code>
<i>Shift-A</i>	<code>bpy.ops.wm.call_menu()</code>
<i>C</i>	<code>bpy.ops.wm.call_menu()</code>
<i>S</i>	<code>bpy.ops.sequencer.slip()</code>
<i>O</i>	<code>bpy.ops.wm.context_set_int()</code>
<i>G</i>	<code>bpy.ops.transform.seq_slide()</code>
<i>EVT_TWEAK_S</i>	<code>bpy.ops.transform.seq_slide()</code>
<i>E</i>	<code>bpy.ops.transform.transform()</code>
<i>M</i>	<code>bpy.ops.marker.add()</code>
<i>Ctrl-M</i>	<code>bpy.ops.marker.rename()</code>

Detailed Reference

Note: Hotkeys marked with the “(default)” prefix are inherited from the default blender keymap

Ctrl-A → sequencer.select_all : **KEYBOARD** → PRESS
(De)select All

bpy.ops.sequencer.select_all(action='TOGGLE')

Properties:	Values:
Action	TOGGLE

(default) A → sequencer.select_all : **KEYBOARD** → PRESS
(De)select All

bpy.ops.sequencer.select_all(action='TOGGLE')

Properties:	Values:
Action	TOGGLE

(default) Ctrl-I → sequencer.select_all : **KEYBOARD** → PRESS
(De)select All

bpy.ops.sequencer.select_all(action='TOGGLE')

Properties:	Values:
Action	INVERT

(default) K → sequencer.cut : **KEYBOARD** → PRESS
Cut Strips

bpy.ops.sequencer.cut(frame=0, type='SOFT', side='BOTH')

Properties:	Values:
Type	SOFT

(default) Shift-K → sequencer.cut : **KEYBOARD** → PRESS
Cut Strips

bpy.ops.sequencer.cut(frame=0, type='SOFT', side='BOTH')

Properties:	Values:
Type	HARD

(default) H → sequencer.mute : **KEYBOARD** → PRESS
Mute Strips

bpy.ops.sequencer.mute(unselected=False)

Properties:	Values:
Unselected	False

(default) Shift-H → sequencer.mute : **KEYBOARD** → PRESS
Mute Strips

bpy.ops.sequencer.mute(unselected=False)

Properties:	Values:
Unselected	True

(default) Alt-H → sequencer.unmute : **KEYBOARD** → PRESS
Un-Mute Strips

bpy.ops.sequencer.unmute(unselected=False)

Properties:	Values:
Unselected	False

(default) Shift-Alt-H → sequencer.unmute : **KEYBOARD** → PRESS

Un-Mute Strips

bpy.ops.sequencer.unmute(unselected=False)

Properties:	Values:
Unselected	True

(default) Shift-L → sequencer.lock : **KEYBOARD** → PRESS

Lock Strips

bpy.ops.sequencer.lock()

(default) Shift-Alt-L → sequencer.unlock : **KEYBOARD** → PRESS

UnLock Strips

bpy.ops.sequencer.unlock()

(default) R → sequencer.reassign_inputs : **KEYBOARD** → PRESS

Reassign Inputs

bpy.ops.sequencer.reassign_inputs()

(default) Alt-R → sequencer.reload : **KEYBOARD** → PRESS

Reload Strips

bpy.ops.sequencer.reload(adjust_length=False)

(default) Shift-Alt-R → sequencer.reload : **KEYBOARD** → PRESS

Reload Strips

bpy.ops.sequencer.reload(adjust_length=False)

Properties:	Values:
Adjust Length	True

(default) Alt-O → sequencer.offset_clear : **KEYBOARD** → PRESS

Clear Strip Offset

bpy.ops.sequencer.offset_clear()

(default) Shift-D → sequencer.duplicate_move : **KEYBOARD** → PRESS

Duplicate Strips

bpy.ops.sequencer.duplicate_move(SEQUENCER_OT_duplicate={"mode":'TRANSLATION'}, TRANSFORM_OT_seq_slide={"value":(0, 0), "snap":False, "snap_target":'CLOSEST', "snap_point":(0, 0, 0), "snap_align":False, "snap_normal":(0, 0, 0), "release_confirm":False})

Properties:	Values:
Duplicate Strips	N/A
Sequence Slide	N/A

(default) X → sequencer.delete : **KEYBOARD** → PRESS

Erase Strips

bpy.ops.sequencer.delete()

(default) DEL → sequencer.delete : **KEYBOARD** → PRESS

Erase Strips

bpy.ops.sequencer.delete()

(default) Ctrl-C → sequencer.copy : **KEYBOARD** → PRESS

Copy

bpy.ops.sequencer.copy()

(default) Ctrl-V → sequencer.paste : **KEYBOARD** → PRESS
Paste

bpy.ops.sequencer.paste()

(default) Y → sequencer.images_separate : **KEYBOARD** → PRESS
Separate Images

bpy.ops.sequencer.images_separate(length=1)

(default) Tab → sequencer.meta_toggle : **KEYBOARD** → PRESS
Toggle Meta Strip

bpy.ops.sequencer.meta_toggle()

(default) Ctrl-G → sequencer.meta_make : **KEYBOARD** → PRESS
Make Meta Strip

bpy.ops.sequencer.meta_make()

(default) Alt-G → sequencer.meta_separate : **KEYBOARD** → PRESS
UnMeta Strip

bpy.ops.sequencer.meta_separate()

(default) HOME → sequencer.view_all : **KEYBOARD** → PRESS
View All

bpy.ops.sequencer.view_all()

(default) NDOF_BUTTON_FIT → sequencer.view_all : **NDOF** → PRESS
View All

bpy.ops.sequencer.view_all()

(default) NUMPAD_PERIOD → sequencer.view_selected : **KEYBOARD** → PRESS
View Selected

bpy.ops.sequencer.view_selected()

(default) NUMPAD_0 → sequencer.view_frame : **KEYBOARD** → PRESS
View Frame

bpy.ops.sequencer.view_frame()

(default) PAGE_UP → sequencer.strip_jump : **KEYBOARD** → PRESS
Jump to Strip

bpy.ops.sequencer.strip_jump(next=True, center=True)

Properties:	Values:
Next Strip	True
Use strip center	False

(default) PAGE_DOWN → sequencer.strip_jump : **KEYBOARD** → PRESS
Jump to Strip

bpy.ops.sequencer.strip_jump(next=True, center=True)

Properties:	Values:
Next Strip	False
Use strip center	False

(default) Alt-PAGE_UP → sequencer.strip_jump : **KEYBOARD** → PRESS
Jump to Strip

bpy.ops.sequencer.strip_jump(next=True, center=True)

Properties:	Values:
Next Strip	True
Use strip center	True

(default) Alt-PAGE_DOWN → sequencer.strip_jump : **KEYBOARD** → PRESS

Jump to Strip

bpy.ops.sequencer.strip_jump(next=True, center=True)

Properties:	Values:
Next Strip	False
Use strip center	True

(default) Alt-LEFT_ARROW → sequencer.swap : **KEYBOARD** → PRESS

Swap Strip

bpy.ops.sequencer.swap(side='RIGHT')

Properties:	Values:
Side	LEFT

(default) Alt-RIGHT_ARROW → sequencer.swap : **KEYBOARD** → PRESS

Swap Strip

bpy.ops.sequencer.swap(side='RIGHT')

Properties:	Values:
Side	RIGHT

(default) BACK_SPACE → sequencer.gap_remove : **KEYBOARD** → PRESS

Remove Gaps

bpy.ops.sequencer.gap_remove(all=False)

Properties:	Values:
All Gaps	False

(default) Shift-BACK_SPACE → sequencer.gap_remove : **KEYBOARD** → PRESS

Remove Gaps

bpy.ops.sequencer.gap_remove(all=False)

Properties:	Values:
All Gaps	True

(default) Shift-EQUAL → sequencer.gap_insert : **KEYBOARD** → PRESS

Insert Gaps

bpy.ops.sequencer.gap_insert(frames=10)

(default) Shift-S → sequencer.snap : **KEYBOARD** → PRESS

Snap Strips

bpy.ops.sequencer.snap(frame=0)

(default) Alt-S → sequencer.swap_inputs : **KEYBOARD** → PRESS

Swap Inputs

bpy.ops.sequencer.swap_inputs()

(default) 1 → sequencer.cut_multicam : **KEYBOARD** → PRESS

Cut multicam

bpy.ops.sequencer.cut_multicam(camera=1)

Properties:	Values:
Camera	1

(default) 2 → sequencer.cut_multicam : **KEYBOARD** → PRESS

Cut multicam

bpy.ops.sequencer.cut_multicam(camera=1)

Properties:	Values:
Camera	2

(default) 3 → sequencer.cut_multicam : **KEYBOARD** → PRESS

Cut multicam

bpy.ops.sequencer.cut_multicam(camera=1)

Properties:	Values:
Camera	3

(default) 4 → sequencer.cut_multicam : **KEYBOARD** → PRESS

Cut multicam

bpy.ops.sequencer.cut_multicam(camera=1)

Properties:	Values:
Camera	4

(default) 5 → sequencer.cut_multicam : **KEYBOARD** → PRESS

Cut multicam

bpy.ops.sequencer.cut_multicam(camera=1)

Properties:	Values:
Camera	5

(default) 6 → sequencer.cut_multicam : **KEYBOARD** → PRESS

Cut multicam

bpy.ops.sequencer.cut_multicam(camera=1)

Properties:	Values:
Camera	6

(default) 7 → sequencer.cut_multicam : **KEYBOARD** → PRESS

Cut multicam

bpy.ops.sequencer.cut_multicam(camera=1)

Properties:	Values:
Camera	7

(default) 8 → sequencer.cut_multicam : **KEYBOARD** → PRESS

Cut multicam

bpy.ops.sequencer.cut_multicam(camera=1)

Properties:	Values:
Camera	8

(default) 9 → sequencer.cut_multicam : **KEYBOARD** → PRESS

Cut multicam

bpy.ops.sequencer.cut_multicam(camera=1)

Properties:	Values:
Camera	9

(default) 0 → sequencer.cut_multicam : **KEYBOARD** → PRESS

Cut multicam

bpy.ops.sequencer.cut_multicam(camera=1)

Properties:	Values:
Camera	10

(default) SELECTMOUSE → sequencer.select : **MOUSE** → PRESS

Activate/Select

bpy.ops.sequencer.select(extend=False, linked_handle=False, left_right='NONE', linked_time=False)

Properties:	Values:
Extend	False
Linked Handle	False
Left/Right	NONE
Linked Time	False

(default) Shift-SELECTMOUSE → sequencer.select : **MOUSE** → PRESS

Activate/Select

bpy.ops.sequencer.select(extend=False, linked_handle=False, left_right='NONE', linked_time=False)

Properties:	Values:
Extend	True
Linked Handle	False
Left/Right	NONE
Linked Time	False

(default) Alt-SELECTMOUSE → sequencer.select : **MOUSE** → PRESS

Activate/Select

bpy.ops.sequencer.select(extend=False, linked_handle=False, left_right='NONE', linked_time=False)

Properties:	Values:
Extend	False
Linked Handle	True
Left/Right	NONE
Linked Time	False

(default) Shift-Alt-SELECTMOUSE → sequencer.select : **MOUSE** → PRESS

Activate/Select

bpy.ops.sequencer.select(extend=False, linked_handle=False, left_right='NONE', linked_time=False)

Properties:	Values:
Extend	True
Linked Handle	True
Left/Right	NONE
Linked Time	False

(default) Ctrl-SELECTMOUSE → sequencer.select : **MOUSE** → PRESS

Activate/Select

bpy.ops.sequencer.select(extend=False, linked_handle=False, left_right='NONE', linked_time=False)

Properties:	Values:
Extend	False
Linked Handle	False
Left/Right	MOUSE
Linked Time	True

(default) Ctrl-Shift-SELECTMOUSE → sequencer.select : **MOUSE** → PRESS

Activate/Select

bpy.ops.sequencer.select(extend=False, linked_handle=False, left_right='NONE', linked_time=False)

Properties:	Values:
Extend	True
Linked Handle	False
Left/Right	NONE
Linked Time	True

(default) Ctrl-NUMPAD_PLUS → sequencer.select_more : **KEYBOARD** → PRESS

Select More

bpy.ops.sequencer.select_more()

(default) Ctrl-NUMPAD_MINUS → sequencer.select_less : **KEYBOARD** → PRESS

Select Less

bpy.ops.sequencer.select_less()

(default) L → sequencer.select_linked_pick : **KEYBOARD** → PRESS

Select Pick Linked

bpy.ops.sequencer.select_linked_pick(extend=False)

Properties:	Values:
Extend	False

(default) Shift-L → sequencer.select_linked_pick : **KEYBOARD** → PRESS

Select Pick Linked

bpy.ops.sequencer.select_linked_pick(extend=False)

Properties:	Values:
Extend	True

(default) Ctrl-L → sequencer.select_linked : **KEYBOARD** → PRESS

Select Linked

bpy.ops.sequencer.select_linked()

(default) B → sequencer.select_border : **KEYBOARD** → PRESS

Border Select

bpy.ops.sequencer.select_border(gesture_mode=0, xmin=0, xmax=0, ymin=0, ymax=0, extend=True)

(default) Shift-G → sequencer.select_grouped : **KEYBOARD** → PRESS

Select Grouped

bpy.ops.sequencer.select_grouped(type='TYPE', extend=False, use_active_channel=False)

(default) Shift-A → wm.call_menu : **KEYBOARD** → PRESS

Call Menu

bpy.ops.wm.call_menu(name='')

Properties:	Values:
Name	SEQUENCER_MT_add

(default) C → wm.call_menu : **KEYBOARD** → PRESS

Call Menu

bpy.ops.wm.call_menu(name='')

Properties:	Values:
Name	SEQUENCER_MT_change

(default) S → sequencer.slip : **KEYBOARD** → PRESS

Trim Strips

bpy.ops.sequencer.slip(offset=0)

(default) O → wm.context_set_int : **KEYBOARD** → PRESS

Context Set

bpy.ops.wm.context_set_int(data_path="", value=0, relative=False)

Properties:	Values:
Context Attributes	scene.sequence_editor.overlay_frame
Value	0

(default) G → transform.seq_slide : **KEYBOARD** → PRESS

Sequence Slide

bpy.ops.transform.seq_slide(value=(0, 0), snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), release_confirm=False)

(default) EVT_TWEAK_S → transform.seq_slide : **TWEAK** → ANY

Sequence Slide

bpy.ops.transform.seq_slide(value=(0, 0), snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), release_confirm=False)

(default) E → transform.transform : **KEYBOARD** → PRESS

Transform

bpy.ops.transform.transform(mode='TRANSLATION', value=(0, 0, 0, 0), axis=(0, 0, 0), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpencil_strokes=False, release_confirm=False)

Properties:	Values:
Mode	TIME_EXTEND

(default) M → marker.add : **KEYBOARD** → PRESS

Add Time Marker

bpy.ops.marker.add()

(default) Ctrl-M → marker.rename : **KEYBOARD** → PRESS

Rename Marker

bpy.ops.marker.rename(name="RenamedMarker")

Timeline

Quick Reference

Hotkey	Operator
<i>LEFTMOUSE</i>	<code>bpy.ops.anim.change_frame()</code>
<i>S</i>	<code>bpy.ops.time.start_frame_set()</code>
<i>E</i>	<code>bpy.ops.time.end_frame_set()</code>
<i>HOME</i>	<code>bpy.ops.time.view_all()</code>
<i>NDOF_BUTTON_FIT</i>	<code>bpy.ops.time.view_all()</code>
<i>NUMPAD_0</i>	<code>bpy.ops.time.view_frame()</code>

Detailed Reference

Note: Hotkeys marked with the “(default)” prefix are inherited from the default blender keymap

LEFTMOUSE → `anim.change_frame` : **MOUSE** → PRESS

Change Frame

`bpy.ops.anim.change_frame(frame=0, snap=False)`

(default) S → `time.start_frame_set` : **KEYBOARD** → PRESS

Set Start Frame

`bpy.ops.time.start_frame_set()`

(default) E → `time.end_frame_set` : **KEYBOARD** → PRESS

Set End Frame

`bpy.ops.time.end_frame_set()`

(default) HOME → `time.view_all` : **KEYBOARD** → PRESS

View All

`bpy.ops.time.view_all()`

(default) NDOF_BUTTON_FIT → `time.view_all` : **NDOF** → PRESS

View All

`bpy.ops.time.view_all()`

(default) NUMPAD_0 → `time.view_frame` : **KEYBOARD** → PRESS

View Frame

`bpy.ops.time.view_frame()`

UV Editor

Quick Reference

Hotkey	Operator
<i>Alt-EVT_TWEAK_A</i>	<code>bpy.ops.uv.select_lasso()</code>
Continued on next page	

Table 2.17 – continued from previous page

Hotkey	Operator
<i>Shift-Alt-EVT_TWEAK_A</i>	<code>bpy.ops.uv.select_lasso()</code>
<i>Ctrl-Alt-EVT_TWEAK_S</i>	<code>bpy.ops.uv.select_border()</code>
<i>Ctrl-Shift-Alt-EVT_TWEAK_S</i>	<code>bpy.ops.uv.select_border()</code>
<i>Ctrl-A</i>	<code>bpy.ops.uv.select_all()</code>
<i>SELECTMOUSE</i>	<code>bpy.ops.uv.select_linked()</code>
<i>Ctrl-ACTIONMOUSE</i>	<code>bpy.ops.uv.cursor_set()</code>
<i>Q</i>	<code>bpy.ops.wm.context_toggle()</code>
<i>Ctrl-E</i>	<code>bpy.ops.uv.mark_seam()</code>
<i>SELECTMOUSE</i>	<code>bpy.ops.uv.select()</code>
<i>Shift-SELECTMOUSE</i>	<code>bpy.ops.uv.select()</code>
<i>Alt-SELECTMOUSE</i>	<code>bpy.ops.uv.select_loop()</code>
<i>Shift-Alt-SELECTMOUSE</i>	<code>bpy.ops.uv.select_loop()</code>
<i>Y</i>	<code>bpy.ops.uv.select_split()</code>
<i>B</i>	<code>bpy.ops.uv.select_border()</code>
<i>Ctrl-B</i>	<code>bpy.ops.uv.select_border()</code>
<i>C</i>	<code>bpy.ops.uv.circle_select()</code>
<i>Ctrl-EVT_TWEAK_A</i>	<code>bpy.ops.uv.select_lasso()</code>
<i>Ctrl-Shift-EVT_TWEAK_A</i>	<code>bpy.ops.uv.select_lasso()</code>
<i>Ctrl-L</i>	<code>bpy.ops.uv.select_linked()</code>
<i>L</i>	<code>bpy.ops.uv.select_linked_pick()</code>
<i>Ctrl-Shift-L</i>	<code>bpy.ops.uv.select_linked()</code>
<i>Shift-L</i>	<code>bpy.ops.uv.select_linked_pick()</code>
<i>Ctrl-NUMPAD_PLUS</i>	<code>bpy.ops.uv.select_more()</code>
<i>Ctrl-NUMPAD_MINUS</i>	<code>bpy.ops.uv.select_less()</code>
<i>A</i>	<code>bpy.ops.uv.select_all()</code>
<i>Ctrl-I</i>	<code>bpy.ops.uv.select_all()</code>
<i>Shift-P</i>	<code>bpy.ops.uv.select_pinned()</code>
<i>W</i>	<code>bpy.ops.wm.call_menu()</code>
<i>V</i>	<code>bpy.ops.uv.stitch()</code>
<i>P</i>	<code>bpy.ops.uv.pin()</code>
<i>Alt-P</i>	<code>bpy.ops.uv.pin()</code>
<i>E</i>	<code>bpy.ops.uv.unwrap()</code>
<i>Ctrl-V</i>	<code>bpy.ops.uv.minimize_stretch()</code>
<i>Ctrl-P</i>	<code>bpy.ops.uv.pack_islands()</code>
<i>Ctrl-A</i>	<code>bpy.ops.uv.average_islands_scale()</code>
<i>H</i>	<code>bpy.ops.uv.hide()</code>
<i>Shift-H</i>	<code>bpy.ops.uv.hide()</code>
<i>Alt-H</i>	<code>bpy.ops.uv.reveal()</code>
<i>Shift-ACTIONMOUSE</i>	<code>bpy.ops.uv.tile_set()</code>
<i>Shift-S</i>	<code>bpy.ops.wm.call_menu()</code>
<i>Ctrl-Tab</i>	<code>bpy.ops.wm.call_menu()</code>
<i>Shift-O</i>	<code>bpy.ops.wm.context_cycle_enum()</code>
<i>O</i>	<code>bpy.ops.wm.context_toggle_enum()</code>
<i>G</i>	<code>bpy.ops.transform.translate()</code>
<i>EVT_TWEAK_S</i>	<code>bpy.ops.transform.translate()</code>
<i>R</i>	<code>bpy.ops.transform.rotate()</code>
<i>S</i>	<code>bpy.ops.transform.resize()</code>
<i>Ctrl-Shift-Alt-S</i>	<code>bpy.ops.transform.shear()</code>
<i>Ctrl-M</i>	<code>bpy.ops.transform.mirror()</code>

Continued on next page

Table 2.17 – continued from previous page

Hotkey	Operator
<i>Shift-Tab</i>	<code>bpy.ops.wm.context_toggle()</code>
<i>Ctrl-Shift-Tab</i>	<code>bpy.ops.wm.context_menu_enum()</code>

Detailed Reference

Note: Hotkeys marked with the “(default)” prefix are inherited from the default blender keymap

Alt-EVT_TWEAK_A → `uv.select_lasso` : **TWEAK** → ANY

Lasso Select UV

`bpy.ops.uv.select_lasso(path=[], deselect=False, extend=True)`

Properties:	Values:
Deselect	False

Shift-Alt-EVT_TWEAK_A → `uv.select_lasso` : **TWEAK** → ANY

Lasso Select UV

`bpy.ops.uv.select_lasso(path=[], deselect=False, extend=True)`

Properties:	Values:
Deselect	True

Ctrl-Alt-EVT_TWEAK_S → `uv.select_border` : **TWEAK** → ANY

Border Select

`bpy.ops.uv.select_border(pinned=False, gesture_mode=0, xmin=0, xmax=0, ymin=0, ymax=0, extend=True)`

Properties:	Values:
Extend	False

Ctrl-Shift-Alt-EVT_TWEAK_S → `uv.select_border` : **TWEAK** → ANY

Border Select

`bpy.ops.uv.select_border(pinned=False, gesture_mode=0, xmin=0, xmax=0, ymin=0, ymax=0, extend=True)`

Properties:	Values:
Extend	True

Ctrl-A → `uv.select_all` : **KEYBOARD** → PRESS

(De)select All

`bpy.ops.uv.select_all(action='TOGGLE')`

Properties:	Values:
Action	TOGGLE

SELECTMOUSE → `uv.select_linked` : **MOUSE** → DOUBLE_CLICK

Select Linked

`bpy.ops.uv.select_linked(extend=False)`

Ctrl-ACTIONMOUSE → `uv.cursor_set` : **MOUSE** → PRESS

Set 2D Cursor

`bpy.ops.uv.cursor_set(location=(0, 0))`

(default) Q → `wm.context_toggle` : **KEYBOARD** → PRESS

Context Toggle

`bpy.ops.wm.context_toggle(data_path='')`

Properties:	Values:
Context Attributes	<code>tool_settings.use_uv_sculpt</code>

(default) Ctrl-E → `uv.mark_seam` : **KEYBOARD** → PRESS

Mark Seam

`bpy.ops.uv.mark_seam(clear=False)`

(default) SELECTMOUSE → `uv.select` : **MOUSE** → PRESS

Select

`bpy.ops.uv.select(extend=False, location=(0, 0))`

Properties:	Values:
Extend	False

(default) Shift-SELECTMOUSE → `uv.select` : **MOUSE** → PRESS

Select

`bpy.ops.uv.select(extend=False, location=(0, 0))`

Properties:	Values:
Extend	True

(default) Alt-SELECTMOUSE → `uv.select_loop` : **MOUSE** → PRESS

Loop Select

`bpy.ops.uv.select_loop(extend=False, location=(0, 0))`

Properties:	Values:
Extend	False

(default) Shift-Alt-SELECTMOUSE → `uv.select_loop` : **MOUSE** → PRESS

Loop Select

`bpy.ops.uv.select_loop(extend=False, location=(0, 0))`

Properties:	Values:
Extend	True

(default) Y → `uv.select_split` : **KEYBOARD** → PRESS

Select Split

`bpy.ops.uv.select_split()`

(default) B → `uv.select_border` : **KEYBOARD** → PRESS

Border Select

`bpy.ops.uv.select_border(pinned=False, gesture_mode=0, xmin=0, xmax=0, ymin=0, ymax=0, extend=True)`

Properties:	Values:
Pinned	False

(default) Ctrl-B → `uv.select_border` : **KEYBOARD** → PRESS

Border Select

`bpy.ops.uv.select_border(pinned=False, gesture_mode=0, xmin=0, xmax=0, ymin=0, ymax=0, extend=True)`

Properties:	Values:
Pinned	True

(default) C → uv.circle_select : **KEYBOARD** → PRESS

Circle Select

bpy.ops.uv.circle_select(x=0, y=0, radius=1, gesture_mode=0)

(default) Ctrl-EVT_TWEAK_A → uv.select_lasso : **TWEAK** → ANY

Lasso Select UV

bpy.ops.uv.select_lasso(path=[], deselect=False, extend=True)

Properties:	Values:
Deselect	False

(default) Ctrl-Shift-EVT_TWEAK_A → uv.select_lasso : **TWEAK** → ANY

Lasso Select UV

bpy.ops.uv.select_lasso(path=[], deselect=False, extend=True)

Properties:	Values:
Deselect	True

(default) Ctrl-L → uv.select_linked : **KEYBOARD** → PRESS

Select Linked

bpy.ops.uv.select_linked(extend=False)

Properties:	Values:
Extend	False

(default) L → uv.select_linked_pick : **KEYBOARD** → PRESS

Select Linked Pick

bpy.ops.uv.select_linked_pick(extend=False, location=(0, 0))

Properties:	Values:
Extend	False

(default) Ctrl-Shift-L → uv.select_linked : **KEYBOARD** → PRESS

Select Linked

bpy.ops.uv.select_linked(extend=False)

Properties:	Values:
Extend	True

(default) Shift-L → uv.select_linked_pick : **KEYBOARD** → PRESS

Select Linked Pick

bpy.ops.uv.select_linked_pick(extend=False, location=(0, 0))

Properties:	Values:
Extend	True

(default) Ctrl-NUMPAD_PLUS → uv.select_more : **KEYBOARD** → PRESS

Select More

bpy.ops.uv.select_more()

(default) Ctrl-NUMPAD_MINUS → uv.select_less : **KEYBOARD** → PRESS

Select Less

bpy.ops.uv.select_less()

(default) A → uv.select_all : **KEYBOARD** → PRESS

(De)select All

bpy.ops.uv.select_all(action='TOGGLE')

Properties:	Values:
Action	TOGGLE

(default) Ctrl-I → uv.select_all : **KEYBOARD** → PRESS

(De)select All

bpy.ops.uv.select_all(action='TOGGLE')

Properties:	Values:
Action	INVERT

(default) Shift-P → uv.select_pinned : **KEYBOARD** → PRESS

Selected Pinned

bpy.ops.uv.select_pinned()

(default) W → wm.call_menu : **KEYBOARD** → PRESS

Call Menu

bpy.ops.wm.call_menu(name='')

Properties:	Values:
Name	IMAGE_MT_uvseweldalign

(default) V → uv.stitch : **KEYBOARD** → PRESS

Stitch

bpy.ops.uv.stitch(use_limit=False, snap_islands=True, limit=0.01, static_island=0, midpoint_snap=False, clear_seams=True, mode='VERTEX', stored_mode='VERTEX', selection=[])

(default) P → uv.pin : **KEYBOARD** → PRESS

Pin

bpy.ops.uv.pin(clear=False)

Properties:	Values:
Clear	False

(default) Alt-P → uv.pin : **KEYBOARD** → PRESS

Pin

bpy.ops.uv.pin(clear=False)

Properties:	Values:
Clear	True

(default) E → uv.unwrap : **KEYBOARD** → PRESS

Unwrap

bpy.ops.uv.unwrap(method='ANGLE_BASED', fill_holes=True, correct_aspect=True, use_subsurf_data=False, margin=0.001)

(default) Ctrl-V → uv.minimize_stretch : **KEYBOARD** → PRESS

Minimize Stretch

bpy.ops.uv.minimize_stretch(fill_holes=True, blend=0, iterations=0)

(default) Ctrl-P → uv.pack_islands : **KEYBOARD** → PRESS

Pack Islands

bpy.ops.uv.pack_islands(rotate=True, margin=0.001)

(default) **Ctrl-A** → uv.average_islands_scale : **KEYBOARD** → PRESS
Average Islands Scale

bpy.ops.uv.average_islands_scale()

(default) **H** → uv.hide : **KEYBOARD** → PRESS
Hide Selected

bpy.ops.uv.hide(unselected=False)

Properties:	Values:
Unselected	False

(default) **Shift-H** → uv.hide : **KEYBOARD** → PRESS
Hide Selected

bpy.ops.uv.hide(unselected=False)

Properties:	Values:
Unselected	True

(default) **Alt-H** → uv.reveal : **KEYBOARD** → PRESS
Reveal Hidden

bpy.ops.uv.reveal()

(default) **Shift-ACTIONMOUSE** → uv.tile_set : **MOUSE** → PRESS
Set Tile

bpy.ops.uv.tile_set(tile=(0, 0))

(default) **Shift-S** → wm.call_menu : **KEYBOARD** → PRESS
Call Menu

bpy.ops.wm.call_menu(name='')

Properties:	Values:
Name	IMAGE_MT_uv_snap

(default) **Ctrl-Tab** → wm.call_menu : **KEYBOARD** → PRESS
Call Menu

bpy.ops.wm.call_menu(name='')

Properties:	Values:
Name	IMAGE_MT_uv_select_mode

(default) **Shift-O** → wm.context_cycle_enum : **KEYBOARD** → PRESS
Context Enum Cycle

bpy.ops.wm.context_cycle_enum(data_path='', reverse=False, wrap=False)

Properties:	Values:
Context Attributes	tool_settings.proportional_edit_falloff
Wrap	True

(default) **O** → wm.context_toggle_enum : **KEYBOARD** → PRESS
Context Toggle Values

bpy.ops.wm.context_toggle_enum(data_path='', value_1='', value_2='')

Properties:	Values:
Context Attributes	tool_settings.proportional_edit
Value	DISABLED
Value	ENABLED

(default) G → transform.translate : **KEYBOARD** → PRESS

Translate

```
bpy.ops.transform.translate(value=(0, 0, 0), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpencil_strokes=False, texture_space=False, remove_on_cancel=False, release_confirm=False)
```

(default) EVT_TWEAK_S → transform.translate : **TWEAK** → ANY

Translate

```
bpy.ops.transform.translate(value=(0, 0, 0), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpencil_strokes=False, texture_space=False, remove_on_cancel=False, release_confirm=False)
```

(default) R → transform.rotate : **KEYBOARD** → PRESS

Rotate

```
bpy.ops.transform.rotate(value=0, axis=(0, 0, 0), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpencil_strokes=False, release_confirm=False)
```

(default) S → transform.resize : **KEYBOARD** → PRESS

Resize

```
bpy.ops.transform.resize(value=(1, 1, 1), constraint_axis=(False, False, False), constraint_orientation='GLOBAL', mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpencil_strokes=False, texture_space=False, remove_on_cancel=False, release_confirm=False)
```

(default) Ctrl-Shift-Alt-S → transform.shear : **KEYBOARD** → PRESS

Shear

```
bpy.ops.transform.shear(value=0, mirror=False, proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, snap=False, snap_target='CLOSEST', snap_point=(0, 0, 0), snap_align=False, snap_normal=(0, 0, 0), gpencil_strokes=False, release_confirm=False)
```

(default) Ctrl-M → transform.mirror : **KEYBOARD** → PRESS

Mirror

```
bpy.ops.transform.mirror(constraint_axis=(False, False, False), constraint_orientation='GLOBAL', proportional='DISABLED', proportional_edit_falloff='SMOOTH', proportional_size=1, gpencil_strokes=False, release_confirm=False)
```

(default) Shift-Tab → wm.context_toggle : **KEYBOARD** → PRESS

Context Toggle

```
bpy.ops.wm.context_toggle(data_path='')
```

Properties:	Values:
Context Attributes	tool_settings.use_snap

(default) Ctrl-Shift-Tab → wm.context_menu_enum : **KEYBOARD** → PRESS

Context Enum Menu

```
bpy.ops.wm.context_menu_enum(data_path='')
```

Properties:	Values:
Context Attributes	tool_settings.snap_uv_element

View2D

Quick Reference

Hotkey	Operator
<i>MIDDLEMOUSE</i>	<code>bpy.ops.view2d.pan()</code>
<i>ACTIONMOUSE</i>	<code>bpy.ops.view2d.pan()</code>
<i>LEFTMOUSE</i>	<code>bpy.ops.view2d.scroller_activate()</code>
<i>MIDDLEMOUSE</i>	<code>bpy.ops.view2d.scroller_activate()</code>
<i>MIDDLEMOUSE</i>	<code>bpy.ops.view2d.pan()</code>
<i>Shift-MIDDLEMOUSE</i>	<code>bpy.ops.view2d.pan()</code>
<i>TRACKPADPAN</i>	<code>bpy.ops.view2d.pan()</code>
<i>Ctrl-WHEELDOWNMOUSE</i>	<code>bpy.ops.view2d.scroll_right()</code>
<i>Ctrl-WHEELUPMOUSE</i>	<code>bpy.ops.view2d.scroll_left()</code>
<i>Shift-WHEELDOWNMOUSE</i>	<code>bpy.ops.view2d.scroll_down()</code>
<i>Shift-WHEELUPMOUSE</i>	<code>bpy.ops.view2d.scroll_up()</code>
<i>NDOF_MOTION</i>	<code>bpy.ops.view2d.ndof()</code>
<i>WHEELOUTMOUSE</i>	<code>bpy.ops.view2d.zoom_out()</code>
<i>WHEELINMOUSE</i>	<code>bpy.ops.view2d.zoom_in()</code>
<i>NUMPAD_MINUS</i>	<code>bpy.ops.view2d.zoom_out()</code>
<i>NUMPAD_PLUS</i>	<code>bpy.ops.view2d.zoom_in()</code>
<i>Ctrl-TRACKPADPAN</i>	<code>bpy.ops.view2d.zoom()</code>
<i>Any-TIMER1</i>	<code>bpy.ops.view2d.smoothview()</code>
<i>WHEELDOWNMOUSE</i>	<code>bpy.ops.view2d.scroll_down()</code>
<i>WHEELUPMOUSE</i>	<code>bpy.ops.view2d.scroll_up()</code>
<i>WHEELDOWNMOUSE</i>	<code>bpy.ops.view2d.scroll_right()</code>
<i>WHEELUPMOUSE</i>	<code>bpy.ops.view2d.scroll_left()</code>
<i>Ctrl-MIDDLEMOUSE</i>	<code>bpy.ops.view2d.zoom()</code>
<i>TRACKPADZOOM</i>	<code>bpy.ops.view2d.zoom()</code>
<i>Shift-B</i>	<code>bpy.ops.view2d.zoom_border()</code>

Detailed Reference

Note: Hotkeys marked with the “(default)” prefix are inherited from the default blender keymap

MIDDLEMOUSE → view2d.pan : **MOUSE** → ANY

Pan View

`bpy.ops.view2d.pan(deltax=0, deltay=0)`

ACTIONMOUSE → view2d.pan : **MOUSE** → ANY

Pan View

`bpy.ops.view2d.pan(deltax=0, deltay=0)`

(default) LEFTMOUSE → view2d.scroller_activate : **MOUSE** → PRESS

Scroller Activate

```

bpy.ops.view2d.scroller_activate()
(default) MIDDLEMOUSE → view2d.scroller_activate : MOUSE → PRESS
    Scroller Activate
    bpy.ops.view2d.scroller_activate()
(default) MIDDLEMOUSE → view2d.pan : MOUSE → PRESS
    Pan View
    bpy.ops.view2d.pan(deltax=0, deltay=0)
(default) Shift-MIDDLEMOUSE → view2d.pan : MOUSE → PRESS
    Pan View
    bpy.ops.view2d.pan(deltax=0, deltay=0)
(default) TRACKPADPAN → view2d.pan : MOUSE → ANY
    Pan View
    bpy.ops.view2d.pan(deltax=0, deltay=0)
(default) Ctrl-WHEELDOWNMOUSE → view2d.scroll_right : MOUSE → PRESS
    Scroll Right
    bpy.ops.view2d.scroll_right(deltax=0, deltay=0)
(default) Ctrl-WHEELUPMOUSE → view2d.scroll_left : MOUSE → PRESS
    Scroll Left
    bpy.ops.view2d.scroll_left(deltax=0, deltay=0)
(default) Shift-WHEELDOWNMOUSE → view2d.scroll_down : MOUSE → PRESS
    Scroll Down
    bpy.ops.view2d.scroll_down(deltax=0, deltay=0, page=False)
(default) Shift-WHEELUPMOUSE → view2d.scroll_up : MOUSE → PRESS
    Scroll Up
    bpy.ops.view2d.scroll_up(deltax=0, deltay=0, page=False)
(default) NDOF_MOTION → view2d.ndof : NDOF → ANY
    NDOF Pan/Zoom
    bpy.ops.view2d.ndof()
(default) WHEELOUTMOUSE → view2d.zoom_out : MOUSE → PRESS
    Zoom Out
    bpy.ops.view2d.zoom_out(zoomfacx=0, zoomfacy=0)
(default) WHEELINMOUSE → view2d.zoom_in : MOUSE → PRESS
    Zoom In
    bpy.ops.view2d.zoom_in(zoomfacx=0, zoomfacy=0)
(default) NUMPAD_MINUS → view2d.zoom_out : KEYBOARD → PRESS
    Zoom Out
    bpy.ops.view2d.zoom_out(zoomfacx=0, zoomfacy=0)
(default) NUMPAD_PLUS → view2d.zoom_in : KEYBOARD → PRESS
    Zoom In
    bpy.ops.view2d.zoom_in(zoomfacx=0, zoomfacy=0)

```

(default) **Ctrl-TRACKPADPAN** → view2d.zoom : **MOUSE** → ANY

Zoom 2D View

bpy.ops.view2d.zoom(deltax=0, deltax=0)

(default) **Any-TIMER1** → view2d.smoothview : **TIMER** → ANY

Smooth View 2D

bpy.ops.view2d.smoothview(gesture_mode=0, xmin=0, xmax=0, ymin=0, ymax=0)

(default) **WHEELDOWNMOUSE** → view2d.scroll_down : **MOUSE** → PRESS

Scroll Down

bpy.ops.view2d.scroll_down(deltax=0, deltax=0, page=False)

(default) **WHEELUPMOUSE** → view2d.scroll_up : **MOUSE** → PRESS

Scroll Up

bpy.ops.view2d.scroll_up(deltax=0, deltax=0, page=False)

(default) **WHEELDOWNMOUSE** → view2d.scroll_right : **MOUSE** → PRESS

Scroll Right

bpy.ops.view2d.scroll_right(deltax=0, deltax=0)

(default) **WHEELUPMOUSE** → view2d.scroll_left : **MOUSE** → PRESS

Scroll Left

bpy.ops.view2d.scroll_left(deltax=0, deltax=0)

(default) **Ctrl-MIDDLEMOUSE** → view2d.zoom : **MOUSE** → PRESS

Zoom 2D View

bpy.ops.view2d.zoom(deltax=0, deltax=0)

(default) **TRACKPADZOOM** → view2d.zoom : **MOUSE** → ANY

Zoom 2D View

bpy.ops.view2d.zoom(deltax=0, deltax=0)

(default) **Shift-B** → view2d.zoom_border : **KEYBOARD** → PRESS

Zoom to Border

bpy.ops.view2d.zoom_border(gesture_mode=0, xmin=0, xmax=0, ymin=0, ymax=0)

Weight Paint Vertex Selection

Quick Reference

Hotkey	Operator
<i>Ctrl-A</i>	<code>bpy.ops.paint.vert_select_all()</code>
<i>A</i>	<code>bpy.ops.paint.vert_select_all()</code>
<i>Ctrl-I</i>	<code>bpy.ops.paint.vert_select_all()</code>
<i>B</i>	<code>bpy.ops.view3d.select_border()</code>
<i>Ctrl-EVT_TWEAK_A</i>	<code>bpy.ops.view3d.select_lasso()</code>
<i>Ctrl-Shift-EVT_TWEAK_A</i>	<code>bpy.ops.view3d.select_lasso()</code>
<i>C</i>	<code>bpy.ops.view3d.select_circle()</code>

Detailed Reference

Note: Hotkeys marked with the “(default)” prefix are inherited from the default blender keymap

Ctrl-A → paint.vert_select_all : **KEYBOARD** → PRESS

(De)select All

bpy.ops.paint.vert_select_all(action='TOGGLE')

Properties:	Values:
Action	TOGGLE

(default) A → paint.vert_select_all : **KEYBOARD** → PRESS

(De)select All

bpy.ops.paint.vert_select_all(action='TOGGLE')

Properties:	Values:
Action	TOGGLE

(default) Ctrl-I → paint.vert_select_all : **KEYBOARD** → PRESS

(De)select All

bpy.ops.paint.vert_select_all(action='TOGGLE')

Properties:	Values:
Action	INVERT

(default) B → view3d.select_border : **KEYBOARD** → PRESS

Border Select

bpy.ops.view3d.select_border(gesture_mode=0, xmin=0, xmax=0, ymin=0, ymax=0, extend=True)

(default) Ctrl-EVT_TWEAK_A → view3d.select_lasso : **TWEAK** → ANY

Lasso Select

bpy.ops.view3d.select_lasso(path=[], deselect=False, extend=True)

Properties:	Values:
Deselect	False

(default) Ctrl-Shift-EVT_TWEAK_A → view3d.select_lasso : **TWEAK** → ANY

Lasso Select

bpy.ops.view3d.select_lasso(path=[], deselect=False, extend=True)

Properties:	Values:
Deselect	True

(default) C → view3d.select_circle : **KEYBOARD** → PRESS

Circle Select

bpy.ops.view3d.select_circle(x=0, y=0, radius=1, gesture_mode=0)

CHAPTER 3

Indices and tables

- `genindex`
- `modindex`
- `search`

Symbols

3dview (module), 5
3dview ., 24
3dview ., 24
3dview A, 9
3dview ACCENT_GRAVE, 19
3dview ACTIONMOUSE, 9
3dview Alt-., 24
3dview Alt-., 24
3dview Alt-B, 23
3dview Alt-EVT_TWEAK_A, 8
3dview Alt-EVT_TWEAK_S, 9
3dview Alt-F, 12
3dview Alt-HOME, 12
3dview Alt-NUMPAD_PERIOD, 10
3dview Alt-SELECTMOUSE, 21
3dview Alt-SPACE, 26
3dview Alt-Z, 20
3dview Any-0, 20
3dview Any-1, 19
3dview Any-2, 19
3dview Any-3, 19
3dview Any-4, 19
3dview Any-5, 20
3dview Any-6, 20
3dview Any-7, 20
3dview Any-8, 20
3dview Any-9, 20
3dview Any-LEFTMOUSE, 9
3dview Any-TIMER1, 10
3dview B, 23
3dview C, 23
3dview Ctrl-., 24
3dview Ctrl-., 24
3dview Ctrl-A, 26
3dview Ctrl-ACTIONMOUSE, 8
3dview Ctrl-Alt-B, 23
3dview Ctrl-Alt-NUMPAD_0, 23
3dview Ctrl-Alt-SELECTMOUSE, 22
3dview Ctrl-Alt-SPACE, 26
3dview Ctrl-Alt-WHEELDOWNMOUSE, 15
3dview Ctrl-Alt-WHEELUPMOUSE, 15
3dview Ctrl-B, 23
3dview Ctrl-C, 24
3dview Ctrl-EQUAL, 11
3dview Ctrl-HOME, 12
3dview Ctrl-M, 26
3dview Ctrl-MIDDLEMOUSE, 10
3dview Ctrl-MINUS, 11
3dview Ctrl-NDOF_MOTION, 17
3dview Ctrl-NUMPAD_0, 23
3dview Ctrl-NUMPAD_1, 14
3dview Ctrl-NUMPAD_2, 14
3dview Ctrl-NUMPAD_3, 14
3dview Ctrl-NUMPAD_4, 14
3dview Ctrl-NUMPAD_6, 14
3dview Ctrl-NUMPAD_7, 14
3dview Ctrl-NUMPAD_8, 14
3dview Ctrl-NUMPAD_PERIOD, 10
3dview Ctrl-SELECTMOUSE, 21
3dview Ctrl-SPACE, 24
3dview Ctrl-Shift-Alt-S, 25
3dview Ctrl-Shift-Alt-SELECTMOUSE, 22
3dview Ctrl-Shift-EQUAL, 12
3dview Ctrl-Shift-MIDDLEMOUSE, 10
3dview Ctrl-Shift-MINUS, 12
3dview Ctrl-Shift-NDOF_MOTION, 17
3dview Ctrl-Shift-NUMPAD_1, 17
3dview Ctrl-Shift-NUMPAD_3, 17
3dview Ctrl-Shift-NUMPAD_7, 17
3dview Ctrl-Shift-SELECTMOUSE, 22
3dview Ctrl-Shift-Tab, 26
3dview Ctrl-Shift-WHEELDOWNMOUSE, 16
3dview Ctrl-Shift-WHEELUPMOUSE, 16
3dview Ctrl-TRACKPADPAN, 11
3dview Ctrl-V, 24
3dview Ctrl-WHEELDOWNMOUSE, 15
3dview Ctrl-WHEELUPMOUSE, 15
3dview EVT_TWEAK_S, 25

- 3dview G, 25
- 3dview HOME, 12
- 3dview MIDDLEMOUSE, 9
- 3dview MOUSEROTATE, 10
- 3dview NDOF_BUTTON_BACK, 18
- 3dview NDOF_BUTTON_BOTTOM, 18
- 3dview NDOF_BUTTON_FIT, 17
- 3dview NDOF_BUTTON_FRONT, 18
- 3dview NDOF_BUTTON_LEFT, 18
- 3dview NDOF_BUTTON_RIGHT, 18
- 3dview NDOF_BUTTON_ROLL_CCW, 17, 18
- 3dview NDOF_BUTTON_TOP, 18
- 3dview NDOF_MOTION, 17
- 3dview NUMPAD_0, 13
- 3dview NUMPAD_1, 13
- 3dview NUMPAD_2, 13
- 3dview NUMPAD_3, 13
- 3dview NUMPAD_4, 13
- 3dview NUMPAD_5, 13
- 3dview NUMPAD_6, 13
- 3dview NUMPAD_7, 13
- 3dview NUMPAD_8, 14
- 3dview NUMPAD_9, 15
- 3dview NUMPAD_MINUS, 11
- 3dview NUMPAD_PERIOD, 10
- 3dview NUMPAD_PLUS, 11
- 3dview NUMPAD_SLASH, 17
- 3dview R, 25
- 3dview RIGHTMOUSE, 8
- 3dview S, 25
- 3dview SELECTMOUSE, 9, 21
- 3dview Shift-Alt-EVT_TWEAK_A, 9
- 3dview Shift-Alt-EVT_TWEAK_S, 9
- 3dview Shift-Alt-S, 25
- 3dview Shift-Alt-SELECTMOUSE, 22
- 3dview Shift-Alt-T, 26
- 3dview Shift-Alt-WHEELDOWNMOUSE, 16
- 3dview Shift-Alt-WHEELUPMOUSE, 16
- 3dview Shift-B, 23
- 3dview Shift-C, 12
- 3dview Shift-F, 10
- 3dview Shift-MIDDLEMOUSE, 10
- 3dview Shift-NDOF_BUTTON_FRONT, 18
- 3dview Shift-NDOF_BUTTON_RIGHT, 19
- 3dview Shift-NDOF_BUTTON_TOP, 19
- 3dview Shift-NDOF_MOTION, 17
- 3dview Shift-NUMPAD_1, 16
- 3dview Shift-NUMPAD_3, 16
- 3dview Shift-NUMPAD_4, 15
- 3dview Shift-NUMPAD_6, 15
- 3dview Shift-NUMPAD_7, 16
- 3dview Shift-NUMPAD_ENTER, 12
- 3dview Shift-NUMPAD_MINUS, 11
- 3dview Shift-NUMPAD_PERIOD, 10

- 3dview Shift-NUMPAD_PLUS, 11
- 3dview Shift-S, 23
- 3dview Shift-SELECTMOUSE, 21
- 3dview Shift-T, 26
- 3dview Shift-TRACKPADPAN, 10
- 3dview Shift-Tab, 26
- 3dview Shift-W, 25
- 3dview Shift-WHEELDOWNMOUSE, 15
- 3dview Shift-WHEELUPMOUSE, 15
- 3dview Shift-Z, 21
- 3dview TRACKPADPAN, 10
- 3dview TRACKPADZOOM, 10
- 3dview WHEELINMOUSE, 11
- 3dview WHEELOUTMOUSE, 11
- 3dview Z, 20

A

- animation (module), 27
- animation ACTIONMOUSE, 27
- animation Alt-P, 27
- animation Ctrl-ACTIONMOUSE, 27
- animation Ctrl-T, 27
- animation P, 27
- animationchannels (module), 28
- animationchannels A, 29
- animationchannels Alt-G, 31
- animationchannels Alt-W, 30
- animationchannels B, 29
- animationchannels Ctrl-A, 28
- animationchannels Ctrl-F, 29
- animationchannels Ctrl-G, 31
- animationchannels Ctrl-I, 29
- animationchannels Ctrl-LEFTMOUSE, 29
- animationchannels Ctrl-NUMPAD_MINUS, 30
- animationchannels Ctrl-NUMPAD_PLUS, 30
- animationchannels Ctrl-Shift-LEFTMOUSE, 29
- animationchannels Ctrl-Shift-W, 30
- animationchannels DEL, 30
- animationchannels EVT_TWEAK_L, 29
- animationchannels LEFTMOUSE, 28, 29
- animationchannels NUMPAD_MINUS, 30
- animationchannels NUMPAD_PLUS, 30
- animationchannels PAGE_DOWN, 30
- animationchannels PAGE_UP, 30
- animationchannels Shift-LEFTMOUSE, 28, 29
- animationchannels Shift-PAGE_DOWN, 31
- animationchannels Shift-PAGE_UP, 31
- animationchannels Shift-W, 30
- animationchannels Tab, 30
- animationchannels X, 29
- armature (module), 31
- armature A, 34
- armature Alt-F, 34
- armature Alt-H, 34

armature Alt-M, 37
 armature Alt-P, 34
 armature Alt-R, 34
 armature Alt-S, 38
 armature Alt-W, 37
 armature Ctrl-A, 32
 armature Ctrl-ACCENT_GRAVE, 37
 armature Ctrl-ACTIONMOUSE, 37
 armature Ctrl-Alt-A, 34
 armature Ctrl-Alt-S, 38
 armature Ctrl-Alt-SELECTMOUSE, 33
 armature Ctrl-I, 34
 armature Ctrl-LEFTMOUSE, 33
 armature Ctrl-MOUSEMOVE, 33
 armature Ctrl-N, 34
 armature Ctrl-NUMPAD_MINUS, 35
 armature Ctrl-NUMPAD_PLUS, 35
 armature Ctrl-P, 34
 armature Ctrl-R, 38
 armature Ctrl-SELECTMOUSE, 35
 armature Ctrl-Shift-M, 34
 armature Ctrl-Shift-W, 37
 armature Ctrl-X, 36
 armature DEL, 33, 36
 armature E, 36
 armature ESC, 33
 armature F, 37
 armature H, 33
 armature L, 35
 armature LEFT_BRACKET, 35
 armature LEFTMOUSE, 33
 armature M, 37
 armature MOUSEMOVE, 33
 armature P, 37
 armature RIGHT_BRACKET, 35
 armature RIGHTMOUSE, 33
 armature Shift-A, 34
 armature Shift-D, 36
 armature Shift-E, 36
 armature Shift-G, 35
 armature Shift-H, 33
 armature Shift-LEFT_BRACKET, 35
 armature Shift-LEFTMOUSE, 33
 armature Shift-M, 37
 armature Shift-RIGHT_BRACKET, 35
 armature Shift-W, 37
 armature W, 38
 armature X, 33, 36
 armature Y, 37

C

clipeditor (module), 38
 clipeditor ., 48
 clipeditor ., 49

clipeditor A, 44
 clipeditor ACTIONMOUSE, 40, 46, 48
 clipeditor Alt-D, 47
 clipeditor Alt-EVT_TWEAK_A, 40
 clipeditor Alt-EVT_TWEAK_S, 41
 clipeditor Alt-H, 46
 clipeditor Alt-I, 46
 clipeditor Alt-L, 46
 clipeditor Alt-S, 47
 clipeditor Alt-T, 48
 clipeditor B, 44
 clipeditor C, 45
 clipeditor Ctrl-., 48
 clipeditor Ctrl-., 49
 clipeditor Ctrl-A, 40
 clipeditor Ctrl-ACTIONMOUSE, 41
 clipeditor Ctrl-Alt-EVT_TWEAK_A, 45
 clipeditor Ctrl-C, 49
 clipeditor Ctrl-I, 44
 clipeditor Ctrl-J, 47
 clipeditor Ctrl-L, 46
 clipeditor Ctrl-LEFTMOUSE, 45
 clipeditor Ctrl-MIDDLEMOUSE, 41
 clipeditor Ctrl-NUMPAD_2, 42
 clipeditor Ctrl-NUMPAD_4, 42
 clipeditor Ctrl-NUMPAD_8, 42
 clipeditor Ctrl-Shift-Alt-EVT_TWEAK_A, 45
 clipeditor Ctrl-Shift-LEFT_ARROW, 43
 clipeditor Ctrl-Shift-RIGHT_ARROW, 44
 clipeditor Ctrl-TRACKPADPAN, 41
 clipeditor Ctrl-V, 49
 clipeditor DEL, 46
 clipeditor EVT_TWEAK_S, 47
 clipeditor F, 43
 clipeditor G, 47
 clipeditor H, 46
 clipeditor HOME, 43
 clipeditor I, 46
 clipeditor L, 47
 clipeditor LEFTMOUSE, 44, 45
 clipeditor M, 47
 clipeditor MIDDLEMOUSE, 41
 clipeditor NDOF_BUTTON_FIT, 43
 clipeditor NDOF_MOTION, 43
 clipeditor NUMPAD_1, 43
 clipeditor NUMPAD_2, 43
 clipeditor NUMPAD_4, 43
 clipeditor NUMPAD_8, 43
 clipeditor NUMPAD_MINUS, 42
 clipeditor NUMPAD_PERIOD, 43
 clipeditor NUMPAD_PLUS, 42
 clipeditor R, 48
 clipeditor S, 48
 clipeditor SELECTMOUSE, 44

clipeditor Shift-Alt-EVT_TWEAK_A, 41
clipeditor Shift-Alt-EVT_TWEAK_S, 41
clipeditor Shift-Alt-LEFT_ARROW, 44
clipeditor Shift-Alt-RIGHT_ARROW, 44
clipeditor Shift-Alt-T, 48
clipeditor Shift-D, 45
clipeditor Shift-DEL, 45
clipeditor Shift-G, 45
clipeditor Shift-H, 46
clipeditor Shift-MIDDLEMOUSE, 41
clipeditor Shift-NUMPAD_2, 42
clipeditor Shift-NUMPAD_4, 42
clipeditor Shift-NUMPAD_8, 42
clipeditor Shift-SELECTMOUSE, 44
clipeditor Shift-T, 48
clipeditor Shift-X, 45
clipeditor TRACKPADPAN, 41
clipeditor TRACKPADZOOM, 41
clipeditor W, 47
clipeditor WHEELINMOUSE, 41
clipeditor WHEELOUTMOUSE, 42
clipeditor X, 46
clipgrapheditor (module), 50
clipgrapheditor A, 51
clipgrapheditor ACTIONMOUSE, 51
clipgrapheditor Alt-EVT_TWEAK_S, 50
clipgrapheditor Alt-T, 52
clipgrapheditor B, 51
clipgrapheditor Ctrl-A, 51
clipgrapheditor Ctrl-ACTIONMOUSE, 50
clipgrapheditor Ctrl-I, 51
clipgrapheditor DEL, 51
clipgrapheditor EVT_TWEAK_S, 53
clipgrapheditor G, 53
clipgrapheditor HOME, 52
clipgrapheditor L, 52
clipgrapheditor NDOF_BUTTON_FIT, 52
clipgrapheditor NUMPAD_PERIOD, 52
clipgrapheditor R, 53
clipgrapheditor S, 53
clipgrapheditor SELECTMOUSE, 51
clipgrapheditor Shift-Alt-EVT_TWEAK_S, 50
clipgrapheditor Shift-Alt-T, 52
clipgrapheditor Shift-D, 52
clipgrapheditor Shift-DEL, 51
clipgrapheditor Shift-SELECTMOUSE, 51
clipgrapheditor Shift-T, 52
clipgrapheditor Shift-X, 52
clipgrapheditor X, 51
console (module), 53
console Any-TEXTINPUT, 57
console BACK_SPACE, 56
console Ctrl-BACK_SPACE, 56
console Ctrl-C, 57
console Ctrl-DEL, 56
console Ctrl-LEFT_ARROW, 54
console Ctrl-NUMPAD_MINUS, 55
console Ctrl-NUMPAD_PLUS, 55
console Ctrl-RIGHT_ARROW, 54
console Ctrl-Shift-C, 57
console Ctrl-SPACE, 57
console Ctrl-Tab, 57
console Ctrl-V, 57
console Ctrl-WHEELDOWNMOUSE, 55
console Ctrl-WHEELUPMOUSE, 55
console DEL, 56
console DOWN_ARROW, 56
console END, 55
console HOME, 54
console LEFT_ARROW, 55
console LEFTMOUSE, 57
console NUMPAD_ENTER, 57
console RET, 56
console RIGHT_ARROW, 55
console Shift-BACK_SPACE, 56
console Shift-NUMPAD_ENTER, 56
console Shift-RET, 56
console Shift-Tab, 57
console Tab, 54, 57
console UP_ARROW, 55
curve (module), 58
curve A, 59
curve Alt-C, 61
curve Alt-H, 62
curve Alt-O, 63
curve Alt-S, 61
curve Alt-T, 61
curve Ctrl-A, 58
curve Ctrl-ACTIONMOUSE, 59
curve Ctrl-Alt-SELECTMOUSE, 59
curve Ctrl-DEL, 61
curve Ctrl-H, 62
curve Ctrl-I, 59
curve Ctrl-L, 60
curve Ctrl-N, 62
curve Ctrl-NUMPAD_MINUS, 59
curve Ctrl-NUMPAD_PLUS, 59
curve Ctrl-P, 62
curve Ctrl-SELECTMOUSE, 60
curve Ctrl-T, 61
curve Ctrl-X, 61
curve DEL, 61
curve E, 60
curve F, 61
curve H, 62
curve L, 60
curve O, 62
curve P, 60

curve Shift-A, 59
 curve Shift-ACTIONMOUSE, 59
 curve Shift-D, 60
 curve Shift-G, 60
 curve Shift-H, 62
 curve Shift-L, 60
 curve Shift-O, 62
 curve Shift-R, 59
 curve V, 59
 curve W, 62
 curve X, 61
 curve Y, 60

D

dopesheet (module), 63
 dopesheet A, 66
 dopesheet Alt-B, 66
 dopesheet Alt-K, 67
 dopesheet Alt-SELECTMOUSE, 64
 dopesheet B, 66
 dopesheet C, 67
 dopesheet Ctrl-A, 64
 dopesheet Ctrl-Alt-P, 69
 dopesheet Ctrl-Alt-SELECTMOUSE, 65
 dopesheet Ctrl-C, 68
 dopesheet Ctrl-EVT_TWEAK_A, 66
 dopesheet Ctrl-F, 69
 dopesheet Ctrl-G, 67
 dopesheet Ctrl-I, 66
 dopesheet Ctrl-K, 67
 dopesheet Ctrl-M, 70
 dopesheet Ctrl-NUMPAD_MINUS, 67
 dopesheet Ctrl-NUMPAD_PLUS, 67
 dopesheet Ctrl-SELECTMOUSE, 65
 dopesheet Ctrl-Shift-Alt-SELECTMOUSE, 65
 dopesheet Ctrl-Shift-EVT_TWEAK_A, 66
 dopesheet Ctrl-Shift-SELECTMOUSE, 65
 dopesheet Ctrl-Shift-V, 69
 dopesheet Ctrl-V, 69
 dopesheet DEL, 68
 dopesheet E, 70
 dopesheet EVT_TWEAK_S, 69
 dopesheet G, 69
 dopesheet HOME, 69
 dopesheet I, 68
 dopesheet K, 67
 dopesheet L, 67
 dopesheet LEFT_BRACKET, 65
 dopesheet M, 70
 dopesheet NDOF_BUTTON_FIT, 69
 dopesheet NUMPAD_0, 69
 dopesheet NUMPAD_PERIOD, 69
 dopesheet O, 70
 dopesheet R, 68

dopesheet RIGHT_BRACKET, 66
 dopesheet S, 70
 dopesheet SELECTMOUSE, 64
 dopesheet Shift-Alt-SELECTMOUSE, 65
 dopesheet Shift-D, 68
 dopesheet Shift-E, 68
 dopesheet Shift-K, 67
 dopesheet Shift-M, 67
 dopesheet Shift-O, 68
 dopesheet Shift-S, 67
 dopesheet Shift-SELECTMOUSE, 65
 dopesheet Shift-T, 70
 dopesheet T, 68
 dopesheet Tab, 69
 dopesheet V, 68
 dopesheet X, 68

F

facemask (module), 71
 facemask A, 71
 facemask Alt-H, 72
 facemask Ctrl-A, 71
 facemask Ctrl-I, 71
 facemask Ctrl-L, 72
 facemask H, 71
 facemask L, 72
 facemask Shift-H, 71
 facemask Shift-L, 72
 filebrowsermain (module), 72
 filebrowsermain A, 76
 filebrowsermain Alt-RIGHTMOUSE, 74
 filebrowsermain Any-MOUSEMOVE, 76
 filebrowsermain B, 76
 filebrowsermain BUTTON4MOUSE, 76
 filebrowsermain BUTTON5MOUSE, 76
 filebrowsermain Ctrl-A, 73
 filebrowsermain Ctrl-LEFTMOUSE, 76
 filebrowsermain Ctrl-NUMPAD_MINUS, 77
 filebrowsermain Ctrl-NUMPAD_PLUS, 77
 filebrowsermain Ctrl-Shift-DOWN_ARROW, 75
 filebrowsermain Ctrl-Shift-LEFT_ARROW, 75
 filebrowsermain Ctrl-Shift-LEFTMOUSE, 73
 filebrowsermain Ctrl-Shift-RIGHT_ARROW, 76
 filebrowsermain Ctrl-Shift-UP_ARROW, 74
 filebrowsermain DOWN_ARROW, 74
 filebrowsermain EVT_TWEAK_L, 76
 filebrowsermain LEFT_ARROW, 75
 filebrowsermain LEFTMOUSE, 73
 filebrowsermain NUMPAD_MINUS, 77
 filebrowsermain NUMPAD_PERIOD, 73
 filebrowsermain NUMPAD_PLUS, 76
 filebrowsermain RIGHT_ARROW, 75
 filebrowsermain RIGHTMOUSE, 74
 filebrowsermain Shift-DOWN_ARROW, 75

filebrowsermain Shift-LEFT_ARROW, 75
filebrowsermain Shift-LEFTMOUSE, 73
filebrowsermain Shift-NUMPAD_MINUS, 77
filebrowsermain Shift-NUMPAD_PLUS, 76
filebrowsermain Shift-RIGHT_ARROW, 75
filebrowsermain Shift-RIGHTMOUSE, 74
filebrowsermain Shift-UP_ARROW, 74
filebrowsermain UP_ARROW, 74

G

grapheditor (module), 77
grapheditor ,, 86
grapheditor ., 87
grapheditor A, 81
grapheditor ACTIONMOUSE, 80
grapheditor Alt-B, 82
grapheditor Alt-C, 84
grapheditor Alt-EVT_TWEAK_A, 79
grapheditor Alt-EVT_TWEAK_S, 79
grapheditor Alt-K, 83
grapheditor Alt-O, 84
grapheditor Alt-SELECTMOUSE, 80
grapheditor B, 82
grapheditor C, 82
grapheditor Ctrl-., 87
grapheditor Ctrl-A, 79
grapheditor Ctrl-ACTIONMOUSE, 79
grapheditor Ctrl-Alt-B, 82
grapheditor Ctrl-Alt-P, 85
grapheditor Ctrl-Alt-SELECTMOUSE, 80
grapheditor Ctrl-B, 82
grapheditor Ctrl-C, 85
grapheditor Ctrl-E, 84
grapheditor Ctrl-EVT_TWEAK_A, 82
grapheditor Ctrl-G, 83
grapheditor Ctrl-H, 80
grapheditor Ctrl-I, 81
grapheditor Ctrl-K, 83
grapheditor Ctrl-M, 87
grapheditor Ctrl-NUMPAD_MINUS, 83
grapheditor Ctrl-NUMPAD_PLUS, 83
grapheditor Ctrl-SELECTMOUSE, 81
grapheditor Ctrl-Shift-ACTIONMOUSE, 84
grapheditor Ctrl-Shift-Alt-SELECTMOUSE, 81
grapheditor Ctrl-Shift-EVT_TWEAK_A, 82
grapheditor Ctrl-Shift-M, 85
grapheditor Ctrl-Shift-SELECTMOUSE, 81
grapheditor Ctrl-Shift-V, 85
grapheditor Ctrl-V, 85
grapheditor DEL, 84
grapheditor E, 86
grapheditor EVT_TWEAK_S, 86
grapheditor G, 85
grapheditor HOME, 85

grapheditor I, 84
grapheditor K, 83
grapheditor L, 83
grapheditor LEFT_BRACKET, 81
grapheditor M, 87
grapheditor NDOF_BUTTON_FIT, 85
grapheditor NUMPAD_0, 85
grapheditor NUMPAD_PERIOD, 85
grapheditor O, 86
grapheditor R, 86
grapheditor RIGHT_BRACKET, 81
grapheditor S, 86
grapheditor SELECTMOUSE, 79, 80
grapheditor Shift-Alt-EVT_TWEAK_A, 79
grapheditor Shift-Alt-EVT_TWEAK_S, 79
grapheditor Shift-Alt-SELECTMOUSE, 80
grapheditor Shift-D, 84
grapheditor Shift-K, 83
grapheditor Shift-M, 83
grapheditor Shift-O, 84
grapheditor Shift-S, 83
grapheditor Shift-SELECTMOUSE, 80
grapheditor T, 84
grapheditor Tab, 85
grapheditor V, 83
grapheditor X, 84

I

image (module), 87
image ,, 93
image ., 93
image 1, 91
image 2, 92
image 3, 92
image 4, 92
image 5, 92
image 6, 92
image 7, 92
image 8, 92
image ACTIONMOUSE, 88, 91
image Alt-F, 88
image Ctrl-., 93
image Ctrl-ACTIONMOUSE, 91
image Ctrl-Alt-B, 93
image Ctrl-B, 93
image Ctrl-MIDDLEMOUSE, 89
image Ctrl-NUMPAD_2, 90
image Ctrl-NUMPAD_4, 90
image Ctrl-NUMPAD_8, 90
image Ctrl-TRACKPADPAN, 90
image HOME, 89
image LEFTMOUSE, 91
image MIDDLEMOUSE, 89
image NDOF_BUTTON_FIT, 89

image NDOF_MOTION, 89
 image NUMPAD_1, 90
 image NUMPAD_2, 91
 image NUMPAD_4, 91
 image NUMPAD_8, 91
 image NUMPAD_MINUS, 89
 image NUMPAD_PERIOD, 89
 image NUMPAD_PLUS, 89
 image SELECTMOUSE, 88
 image Shift-ACTIONMOUSE, 91
 image Shift-B, 90
 image Shift-HOME, 89
 image Shift-MIDDLEMOUSE, 89
 image Shift-NUMPAD_2, 90
 image Shift-NUMPAD_4, 90
 image Shift-NUMPAD_8, 90
 image Tab, 91
 image TRACKPADPAN, 89
 image TRACKPADZOOM, 89
 image WHEELINMOUSE, 89
 image WHEELOUTMOUSE, 89
 info (module), 93
 info A, 94
 info B, 94
 info Ctrl-A, 94
 info Ctrl-C, 94
 info DEL, 94
 info R, 94
 info SELECTMOUSE, 94
 info X, 94

L

lattice (module), 95
 lattice A, 95
 lattice Ctrl-A, 95
 lattice Ctrl-F, 96
 lattice Ctrl-H, 96
 lattice Ctrl-I, 95
 lattice Ctrl-NUMPAD_MINUS, 95
 lattice Ctrl-NUMPAD_PLUS, 95
 lattice Ctrl-P, 95
 lattice O, 96
 lattice Shift-O, 96

M

markers (module), 96
 markers A, 98
 markers Alt-EVT_TWEAK_S, 97
 markers Alt-SELECTMOUSE, 97
 markers B, 98
 markers Ctrl-A, 97
 markers Ctrl-B, 98
 markers Ctrl-M, 98
 markers Ctrl-SELECTMOUSE, 97

markers Ctrl-Shift-SELECTMOUSE, 97
 markers DEL, 98
 markers EVT_TWEAK_S, 97
 markers G, 98
 markers M, 97
 markers SELECTMOUSE, 97
 markers Shift-D, 97
 markers Shift-SELECTMOUSE, 97
 markers X, 98
 maskediting (module), 98
 maskediting A, 102
 maskediting ACTIONMOUSE, 103
 maskediting Alt-C, 103
 maskediting Alt-EVT_TWEAK_A, 100
 maskediting Alt-EVT_TWEAK_S, 100
 maskediting Alt-H, 103
 maskediting Alt-I, 104
 maskediting Alt-N, 100
 maskediting Alt-P, 104
 maskediting Alt-S, 105
 maskediting B, 102
 maskediting C, 102
 maskediting Ctrl-A, 99
 maskediting Ctrl-ACTIONMOUSE, 100, 101
 maskediting Ctrl-Alt-EVT_TWEAK_A, 102
 maskediting Ctrl-Alt-SELECTMOUSE, 99
 maskediting Ctrl-C, 104
 maskediting Ctrl-I, 102
 maskediting Ctrl-L, 102
 maskediting Ctrl-N, 103
 maskediting Ctrl-NUMPAD_MINUS, 103
 maskediting Ctrl-NUMPAD_PLUS, 103
 maskediting Ctrl-P, 104
 maskediting Ctrl-SELECTMOUSE, 103
 maskediting Ctrl-Shift-Alt-EVT_TWEAK_A, 103
 maskediting Ctrl-Shift-SELECTMOUSE, 100
 maskediting Ctrl-V, 104
 maskediting DEL, 101
 maskediting EVT_TWEAK_S, 104
 maskediting G, 104
 maskediting H, 103
 maskediting I, 104
 maskediting L, 102
 maskediting O, 101
 maskediting R, 105
 maskediting S, 105
 maskediting SELECTMOUSE, 100, 101
 maskediting Shift-A, 101
 maskediting Shift-ACTIONMOUSE, 101
 maskediting Shift-Alt-EVT_TWEAK_A, 100
 maskediting Shift-Alt-EVT_TWEAK_S, 100
 maskediting Shift-D, 104
 maskediting Shift-H, 103
 maskediting Shift-L, 102

- maskediting Shift-O, 101
- maskediting Shift-SELECTMOUSE, 102
- maskediting V, 103
- maskediting X, 101
- mesh (module), 105
- mesh A, 109
- mesh Alt-D, 113
- mesh Alt-E, 111
- mesh Alt-F, 112
- mesh Alt-H, 111
- mesh Alt-J, 112
- mesh Alt-M, 113
- mesh Alt-O, 117
- mesh Alt-P, 108
- mesh Alt-R, 111
- mesh Alt-S, 113
- mesh Alt-SELECTMOUSE, 108
- mesh Alt-V, 112
- mesh Ctrl-0, 116
- mesh Ctrl-1, 116
- mesh Ctrl-2, 116
- mesh Ctrl-3, 116
- mesh Ctrl-4, 116
- mesh Ctrl-5, 116
- mesh Ctrl-A, 107
- mesh Ctrl-ACTIONMOUSE, 114
- mesh Ctrl-Alt-SELECTMOUSE, 107, 109
- mesh Ctrl-B, 108
- mesh Ctrl-DEL, 114
- mesh Ctrl-E, 115
- mesh Ctrl-F, 115
- mesh Ctrl-G, 115
- mesh Ctrl-H, 115
- mesh Ctrl-I, 109
- mesh Ctrl-L, 110
- mesh Ctrl-N, 111
- mesh Ctrl-NUMPAD_MINUS, 110
- mesh Ctrl-NUMPAD_PLUS, 110
- mesh Ctrl-P, 115
- mesh Ctrl-R, 107
- mesh Ctrl-SELECTMOUSE, 109
- mesh Ctrl-Shift-ACTIONMOUSE, 114
- mesh Ctrl-Shift-Alt-F, 110
- mesh Ctrl-Shift-Alt-M, 110
- mesh Ctrl-Shift-Alt-SELECTMOUSE, 107, 109
- mesh Ctrl-Shift-B, 108
- mesh Ctrl-Shift-N, 111
- mesh Ctrl-Shift-NUMPAD_MINUS, 110
- mesh Ctrl-Shift-NUMPAD_PLUS, 110
- mesh Ctrl-Shift-R, 108
- mesh Ctrl-Shift-SELECTMOUSE, 109
- mesh Ctrl-Shift-T, 112
- mesh Ctrl-T, 112
- mesh Ctrl-Tab, 110

- mesh Ctrl-V, 115
- mesh Ctrl-X, 114
- mesh DEL, 114
- mesh E, 111
- mesh F, 113
- mesh H, 111
- mesh I, 108
- mesh J, 114
- mesh K, 114
- mesh L, 110
- mesh O, 116
- mesh P, 113
- mesh SELECTMOUSE, 107
- mesh Shift-A, 113
- mesh Shift-Alt-F, 112
- mesh Shift-Alt-SELECTMOUSE, 109
- mesh Shift-D, 113
- mesh Shift-E, 111
- mesh Shift-G, 110
- mesh Shift-H, 111
- mesh Shift-K, 115
- mesh Shift-L, 110
- mesh Shift-O, 116
- mesh Shift-SELECTMOUSE, 107
- mesh Shift-V, 114
- mesh U, 115
- mesh V, 112
- mesh W, 115
- mesh X, 114
- mesh Y, 114
- metaball (module), 117
- metaball A, 118
- metaball Alt-H, 118
- metaball Alt-O, 119
- metaball Ctrl-A, 117
- metaball Ctrl-I, 118
- metaball DEL, 118
- metaball H, 118
- metaball O, 119
- metaball Shift-A, 117
- metaball Shift-D, 118
- metaball Shift-G, 119
- metaball Shift-H, 118
- metaball Shift-O, 119
- metaball X, 118

N

- nlaeditor (module), 119
- nlaeditor A, 120, 121
- nlaeditor Alt-B, 122
- nlaeditor Alt-D, 123
- nlaeditor Alt-F, 123
- nlaeditor Alt-G, 122
- nlaeditor Alt-S, 123

nlaeditor B, 121
 nlaeditor Ctrl-A, 120, 123
 nlaeditor Ctrl-Alt-P, 122
 nlaeditor Ctrl-I, 121
 nlaeditor Ctrl-M, 124
 nlaeditor Ctrl-SELECTMOUSE, 121
 nlaeditor Ctrl-Shift-M, 124
 nlaeditor Ctrl-Shift-SELECTMOUSE, 121
 nlaeditor DEL, 123
 nlaeditor E, 124
 nlaeditor EVT_TWEAK_S, 124
 nlaeditor G, 124
 nlaeditor H, 123
 nlaeditor HOME, 122
 nlaeditor LEFT_BRACKET, 121
 nlaeditor M, 124
 nlaeditor NDOF_BUTTON_FIT, 122
 nlaeditor NUMPAD_0, 122
 nlaeditor NUMPAD_PERIOD, 122
 nlaeditor PAGE_DOWN, 123
 nlaeditor PAGE_UP, 123
 nlaeditor RIGHT_BRACKET, 121
 nlaeditor S, 124
 nlaeditor SELECTMOUSE, 120
 nlaeditor Shift-A, 122
 nlaeditor Shift-D, 123
 nlaeditor Shift-G, 122
 nlaeditor Shift-K, 122
 nlaeditor Shift-S, 123
 nlaeditor Shift-SELECTMOUSE, 121
 nlaeditor Shift-T, 122
 nlaeditor U, 123
 nlaeditor X, 123
 nlaeditor Y, 123
 nodeeditor (module), 125
 nodeeditor A, 133
 nodeeditor ACTIONMOUSE, 127
 nodeeditor Alt-ACTIONMOUSE, 128, 131
 nodeeditor Alt-D, 136
 nodeeditor Alt-EVT_TWEAK_A, 127, 136
 nodeeditor Alt-EVT_TWEAK_S, 137
 nodeeditor Alt-G, 134
 nodeeditor Alt-HOME, 131
 nodeeditor Alt-MIDDLEMOUSE, 130
 nodeeditor Alt-P, 132
 nodeeditor Alt-SELECTMOUSE, 127, 128
 nodeeditor Alt-V, 130
 nodeeditor B, 132
 nodeeditor C, 130
 nodeeditor Ctrl-A, 127
 nodeeditor Ctrl-ACTIONMOUSE, 127
 nodeeditor Ctrl-Alt-ACTIONMOUSE, 128
 nodeeditor Ctrl-Alt-B, 134
 nodeeditor Ctrl-Alt-EVT_TWEAK_A, 129
 nodeeditor Ctrl-Alt-SELECTMOUSE, 128
 nodeeditor Ctrl-B, 134
 nodeeditor Ctrl-C, 134
 nodeeditor Ctrl-F, 133
 nodeeditor Ctrl-G, 134
 nodeeditor Ctrl-H, 132
 nodeeditor Ctrl-I, 133
 nodeeditor Ctrl-J, 132
 nodeeditor Ctrl-LEFTMOUSE, 130
 nodeeditor Ctrl-P, 131
 nodeeditor Ctrl-R, 134
 nodeeditor Ctrl-SELECTMOUSE, 128
 nodeeditor Ctrl-Shift-ACTIONMOUSE, 128
 nodeeditor Ctrl-Shift-Alt-ACTIONMOUSE, 129
 nodeeditor Ctrl-Shift-Alt-EVT_TWEAK_A, 129
 nodeeditor Ctrl-Shift-Alt-SELECTMOUSE, 129
 nodeeditor Ctrl-Shift-D, 131
 nodeeditor Ctrl-Shift-G, 133
 nodeeditor Ctrl-Shift-LEFTMOUSE, 130
 nodeeditor Ctrl-Shift-SELECTMOUSE, 129
 nodeeditor Ctrl-Shift-Tab, 137
 nodeeditor Ctrl-Tab, 134
 nodeeditor Ctrl-V, 134
 nodeeditor Ctrl-X, 132
 nodeeditor DEL, 132
 nodeeditor EVT_TWEAK_A, 135
 nodeeditor EVT_TWEAK_S, 129, 135, 136
 nodeeditor F, 131
 nodeeditor G, 134, 135
 nodeeditor H, 132
 nodeeditor HOME, 132
 nodeeditor L, 133
 nodeeditor LEFTMOUSE, 130
 nodeeditor M, 132
 nodeeditor NDOF_BUTTON_FIT, 132
 nodeeditor NUMPAD_PERIOD, 132
 nodeeditor P, 134
 nodeeditor R, 136
 nodeeditor S, 136
 nodeeditor SELECTMOUSE, 127
 nodeeditor Shift-A, 131
 nodeeditor Shift-ACTIONMOUSE, 128
 nodeeditor Shift-Alt-ACTIONMOUSE, 129
 nodeeditor Shift-Alt-EVT_TWEAK_A, 127
 nodeeditor Shift-Alt-SELECTMOUSE, 129
 nodeeditor Shift-D, 131
 nodeeditor Shift-F, 131
 nodeeditor Shift-G, 133
 nodeeditor Shift-H, 132
 nodeeditor Shift-L, 133
 nodeeditor Shift-LEFT_BRACKET, 133
 nodeeditor Shift-LEFTMOUSE, 130
 nodeeditor Shift-R, 134
 nodeeditor Shift-RIGHT_BRACKET, 133

nodeeditor Shift-SELECTMOUSE, 128
nodeeditor Shift-Tab, 137
nodeeditor Tab, 127, 134
nodeeditor V, 130
nodeeditor X, 132
nodeeditor Z, 134

O

objectmode (module), 137
objectmode A, 139
objectmode Alt-C, 144
objectmode Alt-D, 144
objectmode Alt-G, 141
objectmode Alt-H, 142
objectmode Alt-I, 144
objectmode Alt-O, 142
objectmode Alt-P, 141
objectmode Alt-R, 141
objectmode Alt-S, 141
objectmode Alt-T, 141
objectmode Ctrl-0, 145
objectmode Ctrl-1, 145
objectmode Ctrl-2, 145
objectmode Ctrl-3, 145
objectmode Ctrl-4, 146
objectmode Ctrl-5, 146
objectmode Ctrl-A, 139, 143
objectmode Ctrl-Alt-C, 141
objectmode Ctrl-Alt-G, 145
objectmode Ctrl-Alt-H, 142
objectmode Ctrl-Alt-P, 144
objectmode Ctrl-G, 144
objectmode Ctrl-H, 142
objectmode Ctrl-I, 139
objectmode Ctrl-J, 144
objectmode Ctrl-L, 143
objectmode Ctrl-NUMPAD_MINUS, 140
objectmode Ctrl-NUMPAD_PLUS, 140
objectmode Ctrl-P, 141
objectmode Ctrl-Shift-A, 143
objectmode Ctrl-Shift-Alt-G, 145
objectmode Ctrl-Shift-Alt-I, 144
objectmode Ctrl-Shift-C, 141
objectmode Ctrl-Shift-G, 145
objectmode Ctrl-Shift-M, 140
objectmode Ctrl-Shift-P, 141
objectmode Ctrl-Shift-T, 145
objectmode Ctrl-T, 141
objectmode DEL, 143
objectmode H, 142
objectmode I, 144
objectmode L, 144
objectmode LEFT_BRACKET, 140
objectmode M, 142
objectmode O, 139
objectmode P, 139
objectmode RIGHT_BRACKET, 140
objectmode Shift-A, 143
objectmode Shift-Alt-G, 141, 145
objectmode Shift-Alt-R, 142
objectmode Shift-Alt-S, 142
objectmode Shift-D, 143
objectmode Shift-DEL, 143
objectmode Shift-G, 140
objectmode Shift-H, 142
objectmode Shift-L, 140
objectmode Shift-LEFT_BRACKET, 140
objectmode Shift-O, 139
objectmode Shift-RIGHT_BRACKET, 140
objectmode Shift-X, 143
objectmode U, 143
objectmode W, 145
objectmode X, 142
outliner (module), 147
outliner ., 149
outliner A, 149
outliner Alt-D, 150
outliner Alt-I, 150
outliner Alt-K, 150
outliner B, 148
outliner Ctrl-A, 147
outliner Ctrl-LEFTMOUSE, 148
outliner Ctrl-Shift-LEFTMOUSE, 148
outliner D, 150
outliner HOME, 148
outliner I, 150
outliner K, 150
outliner LEFTMOUSE, 147
outliner NUMPAD_MINUS, 149
outliner NUMPAD_PERIOD, 149
outliner NUMPAD_PLUS, 149
outliner PAGE_DOWN, 149
outliner PAGE_UP, 149
outliner R, 149
outliner RET, 148
outliner RIGHTMOUSE, 148
outliner S, 149
outliner Shift-A, 149
outliner Shift-LEFTMOUSE, 148
outliner Shift-RET, 148
outliner V, 149

P

particle (module), 151
particle A, 151
particle Alt-H, 152
particle Any-LEFTMOUSE, 152
particle Ctrl-A, 151

particle Ctrl-I, 151
 particle Ctrl-NUMPAD_MINUS, 152
 particle Ctrl-NUMPAD_PLUS, 152
 particle DEL, 152
 particle F, 153
 particle H, 152
 particle L, 152
 particle LEFTMOUSE, 152
 particle O, 153
 particle Shift-F, 153
 particle Shift-H, 152
 particle Shift-K, 153
 particle Shift-L, 152
 particle Shift-LEFTMOUSE, 153
 particle Shift-O, 153
 particle W, 153
 particle X, 152
 pose (module), 154
 pose A, 155, 156
 pose Alt-E, 159
 pose Alt-F, 156
 pose Alt-G, 156
 pose Alt-H, 155
 pose Alt-I, 159
 pose Alt-L, 159
 pose Alt-P, 160
 pose Alt-R, 156
 pose Alt-S, 156
 pose Alt-W, 158
 pose Ctrl-A, 155, 156
 pose Ctrl-ACCENT_GRAVE, 158
 pose Ctrl-Alt-C, 158
 pose Ctrl-Alt-I, 158
 pose Ctrl-Alt-S, 159
 pose Ctrl-C, 156
 pose Ctrl-E, 159
 pose Ctrl-G, 158
 pose Ctrl-I, 156
 pose Ctrl-L, 159
 pose Ctrl-P, 155
 pose Ctrl-R, 156
 pose Ctrl-Shift-Alt-I, 159
 pose Ctrl-Shift-C, 157
 pose Ctrl-Shift-F, 157
 pose Ctrl-Shift-L, 159
 pose Ctrl-Shift-V, 156
 pose Ctrl-Shift-W, 158
 pose Ctrl-V, 156
 pose H, 155
 pose I, 159
 pose L, 157
 pose LEFT_BRACKET, 157
 pose M, 158
 pose RIGHT_BRACKET, 157

pose Shift-A, 155
 pose Shift-E, 159
 pose Shift-G, 157
 pose Shift-H, 155
 pose Shift-I, 158
 pose Shift-L, 159
 pose Shift-LEFT_BRACKET, 157
 pose Shift-M, 158
 pose Shift-P, 157
 pose Shift-RIGHT_BRACKET, 157
 pose Shift-W, 158
 pose W, 160

S

sequencer (module), 160
 sequencer 0, 167
 sequencer 1, 165
 sequencer 2, 166
 sequencer 3, 166
 sequencer 4, 166
 sequencer 5, 166
 sequencer 6, 166
 sequencer 7, 166
 sequencer 8, 166
 sequencer 9, 166
 sequencer A, 162
 sequencer Alt-G, 164
 sequencer Alt-H, 162
 sequencer Alt-LEFT_ARROW, 165
 sequencer Alt-O, 163
 sequencer Alt-PAGE_DOWN, 165
 sequencer Alt-PAGE_UP, 164
 sequencer Alt-R, 163
 sequencer Alt-RIGHT_ARROW, 165
 sequencer Alt-S, 165
 sequencer Alt-SELECTMOUSE, 167
 sequencer B, 168
 sequencer BACK_SPACE, 165
 sequencer C, 168
 sequencer Ctrl-A, 161
 sequencer Ctrl-C, 163
 sequencer Ctrl-G, 164
 sequencer Ctrl-I, 162
 sequencer Ctrl-L, 168
 sequencer Ctrl-M, 169
 sequencer Ctrl-NUMPAD_MINUS, 168
 sequencer Ctrl-NUMPAD_PLUS, 168
 sequencer Ctrl-SELECTMOUSE, 167
 sequencer Ctrl-Shift-SELECTMOUSE, 168
 sequencer Ctrl-V, 163
 sequencer DEL, 163
 sequencer E, 169
 sequencer EVT_TWEAK_S, 169
 sequencer G, 169

- sequencer H, 162
- sequencer HOME, 164
- sequencer K, 162
- sequencer L, 168
- sequencer M, 169
- sequencer NDOF_BUTTON_FIT, 164
- sequencer NUMPAD_0, 164
- sequencer NUMPAD_PERIOD, 164
- sequencer O, 169
- sequencer PAGE_DOWN, 164
- sequencer PAGE_UP, 164
- sequencer R, 163
- sequencer S, 169
- sequencer SELECTMOUSE, 167
- sequencer Shift-A, 168
- sequencer Shift-Alt-H, 162
- sequencer Shift-Alt-L, 163
- sequencer Shift-Alt-R, 163
- sequencer Shift-Alt-SELECTMOUSE, 167
- sequencer Shift-BACK_SPACE, 165
- sequencer Shift-D, 163
- sequencer Shift-EQUAL, 165
- sequencer Shift-G, 168
- sequencer Shift-H, 162
- sequencer Shift-K, 162
- sequencer Shift-L, 163, 168
- sequencer Shift-S, 165
- sequencer Shift-SELECTMOUSE, 167
- sequencer Tab, 164
- sequencer X, 163
- sequencer Y, 164

T

- timeline (module), 170
- timeline E, 170
- timeline HOME, 170
- timeline LEFTMOUSE, 170
- timeline NDOF_BUTTON_FIT, 170
- timeline NUMPAD_0, 170
- timeline S, 170

U

- uveditor (module), 170
- uveditor A, 174
- uveditor Alt-EVT_TWEAK_A, 172
- uveditor Alt-H, 176
- uveditor Alt-P, 175
- uveditor Alt-SELECTMOUSE, 173
- uveditor B, 173
- uveditor C, 173
- uveditor Ctrl-A, 172, 175
- uveditor Ctrl-ACTIONMOUSE, 172
- uveditor Ctrl-Alt-EVT_TWEAK_S, 172
- uveditor Ctrl-B, 173

- uveditor Ctrl-E, 173
- uveditor Ctrl-EVT_TWEAK_A, 174
- uveditor Ctrl-I, 175
- uveditor Ctrl-L, 174
- uveditor Ctrl-M, 177
- uveditor Ctrl-NUMPAD_MINUS, 174
- uveditor Ctrl-NUMPAD_PLUS, 174
- uveditor Ctrl-P, 175
- uveditor Ctrl-Shift-Alt-EVT_TWEAK_S, 172
- uveditor Ctrl-Shift-Alt-S, 177
- uveditor Ctrl-Shift-EVT_TWEAK_A, 174
- uveditor Ctrl-Shift-L, 174
- uveditor Ctrl-Shift-Tab, 177
- uveditor Ctrl-Tab, 176
- uveditor Ctrl-V, 175
- uveditor E, 175
- uveditor EVT_TWEAK_S, 177
- uveditor G, 176
- uveditor H, 176
- uveditor L, 174
- uveditor O, 176
- uveditor P, 175
- uveditor Q, 172
- uveditor R, 177
- uveditor S, 177
- uveditor SELECTMOUSE, 172, 173
- uveditor Shift-ACTIONMOUSE, 176
- uveditor Shift-Alt-EVT_TWEAK_A, 172
- uveditor Shift-Alt-SELECTMOUSE, 173
- uveditor Shift-H, 176
- uveditor Shift-L, 174
- uveditor Shift-O, 176
- uveditor Shift-P, 175
- uveditor Shift-S, 176
- uveditor Shift-SELECTMOUSE, 173
- uveditor Shift-Tab, 177
- uveditor V, 175
- uveditor W, 175
- uveditor Y, 173

V

- view2d (module), 178
- view2d ACTIONMOUSE, 178
- view2d Any-TIMER1, 180
- view2d Ctrl-MIDDLEMOUSE, 180
- view2d Ctrl-TRACKPADPAN, 179
- view2d Ctrl-WHEELDOWNMOUSE, 179
- view2d Ctrl-WHEELUPMOUSE, 179
- view2d LEFTMOUSE, 178
- view2d MIDDLEMOUSE, 178, 179
- view2d NDOF_MOTION, 179
- view2d NUMPAD_MINUS, 179
- view2d NUMPAD_PLUS, 179
- view2d Shift-B, 180

view2d Shift-MIDDLEMOUSE, [179](#)
view2d Shift-WHEELDOWNMOUSE, [179](#)
view2d Shift-WHEELUPMOUSE, [179](#)
view2d TRACKPADPAN, [179](#)
view2d TRACKPADZOOM, [180](#)
view2d WHEELDOWNMOUSE, [180](#)
view2d WHEELINMOUSE, [179](#)
view2d WHEELOUTMOUSE, [179](#)
view2d WHEELUPMOUSE, [180](#)

W

weightpaintvertexselection (module), [180](#)
weightpaintvertexselection A, [181](#)
weightpaintvertexselection B, [181](#)
weightpaintvertexselection C, [181](#)
weightpaintvertexselection Ctrl-A, [181](#)
weightpaintvertexselection Ctrl-EVT_TWEAK_A, [181](#)
weightpaintvertexselection Ctrl-I, [181](#)
weightpaintvertexselection Ctrl-Shift-EVT_TWEAK_A,
[181](#)