Skip to content

Ui Operators

bpy.ops.ui.**assign_default_button()**

Set this property's current value as the new default

bpy.ops.ui.**button_execute(*, skip_depressed=False)**

Presses active button

PARAMETERS:

skip_depressed (*boolean, (optional)*) – Skip Depressed

bpy.ops.ui.**button_string_clear()**

Unsets the text of the active button

bpy.ops.ui.**copy_as_driver_button()**

Create a new driver with this property as input, and copy it to the internal clipboard. Use Paste Driver to add it to the target property, or Paste Driver Variables to extend an existing driver

bpy.ops.ui.**copy_data_path_button(*, full_path=False)**

Copy the RNA data path for this property to the clipboard

PARAMETERS:

full_path (*boolean, (optional)*) – full_path, Copy full data path

bpy.ops.ui.**copy_driver_to_selected_button(*, all=False)**

Copy the property's driver from the active item to the same property of all selected items, if the same property exists

PARAMETERS:

all (*boolean, (optional)*) – All, Copy to selected the drivers of all elements of the array

bpy.ops.ui.**copy_python_command_button()**

Copy the Python command matching this button

bpy.ops.ui.**copy_to_selected_button(*, all=True)**

Copy the property's value from the active item to the same property of all selected items if the same property exists

PARAMETERS:

all (*boolean, (optional)*) – All, Copy to selected all elements of the array

bpy.ops.ui.**drop_color(*, color=(0.0, 0.0, 0.0, 0.0), gamma=False, has_alpha=False)**

Drop colors to buttons

PARAMETERS:

- **color** (*float array of 4 items in [0, inf], (optional)*) – Color, Source color
- **gamma** (*boolean, (optional)*) – Gamma Corrected, The source color is gamma corrected
- **has_alpha** (*boolean, (optional)*) – Has Alpha, The source color contains an Alpha component

bpy.ops.ui.**drop_material(*, session_uid=0)**

Drag material to Material slots in Properties

PARAMETERS:

session_uid (*int in [-inf, inf], (optional)*) – Session UID, Session UID of the data-block to use by the operator

bpy.ops.ui.**drop_name(*, string='')**

Drop name to button

PARAMETERS:

string (*string, (optional, never None)*) – String, The string value to drop into the button

bpy.ops.ui.**editsource()**

Edit UI source code of the active button

bpy.ops.ui.**eyedropper_bone()**

Sample a bone from the 3D View or the Outliner to store in a property

bpy.ops.ui.**eyedropper_color(*, prop_data_path='')**

Sample a color from the Blender window to store in a property

PARAMETERS:

prop_data_path (*string, (optional, never None)*) – Data Path, Path of property to be set with the depth

bpy.ops.ui.**eyedropper_colorramp()**

Sample a color band

bpy.ops.ui.**eyedropper_colorramp_point()**

Point-sample a color band

bpy.ops.ui.**eyedropper_depth(*, prop_data_path='')**

Sample depth from the 3D view

PARAMETERS:

prop_data_path (*string, (optional, never None)*) – Data Path, Path of property to be set with the depth

bpy.ops.ui.**eyedropper_driver(*, mapping_type='SINGLE_MANY')**

Pick a property to use as a driver target

PARAMETERS:

mapping_type (*enum in ['SINGLE_MANY', 'DIRECT', 'MATCH', 'NONE_ALL', 'NONE_SINGLE'], (optional)*) –

Mapping Type, Method used to match target and driven properties

- `SINGLE_MANY` All from Target – Drive all components of this property using the target picked.
- `DIRECT` Single from Target – Drive this component of this property using the target picked.
- `MATCH` Match Indices – Create drivers for each pair of corresponding elements.
- `NONE_ALL` Manually Create Later – Create drivers for all properties without assigning any targets yet.
- `NONE_SINGLE` Manually Create Later (Single) – Create driver for this property only and without assigning any targets yet.

bpy.ops.ui.**eyedropper_grease_pencil_color(*, mode='MATERIAL', material_mode='STROKE')**

Sample a color from the Blender Window and create Grease Pencil material

PARAMETERS:

- **mode** (*enum in ['MATERIAL', 'PALETTE', 'BRUSH'], (optional)*) – Mode
- **material_mode** (*enum in ['STROKE', 'FILL', 'BOTH'], (optional)*) – Material Mode

bpy.ops.ui.**eyedropper_id()**

Sample a data-block from the 3D View to store in a property

bpy.ops.ui.**jump_to_target_button()**

Switch to the target object or bone

bpy.ops.ui.**list_start_filter()**

Start entering filter text for the list in focus

Start entering filter text for the list in focus

bpy.ops.ui.**override_idtemplate_clear()**

Delete the selected local override and relink its usages to the linked data-block if possible, else reset it and mark it as non editable

bpy.ops.ui.**override_idtemplate_make()**

Create a local override of the selected linked data-block, and its hierarchy of dependencies

bpy.ops.ui.**override_idtemplate_reset()**

Reset the selected local override to its linked reference values

bpy.ops.ui.**override_remove_button(*, all=True)**

Remove an override operation

PARAMETERS:

all (*boolean, (optional)*) – All, Reset to default values all elements of the array

bpy.ops.ui.**override_type_set_button(*, all=True, type='REPLACE')**

Create an override operation, or set the type of an existing one

PARAMETERS:

- **all** (*boolean, (optional)*) – All, Reset to default values all elements of the array
- **type** (*enum in ['NOOP', 'REPLACE', 'DIFFERENCE', 'FACTOR'], (optional)*) –

  Type, Type of override operation

  - `NOOP` NoOp – 'No-Operation', place holder preventing automatic override to ever affect the property.
  - `REPLACE` Replace – Completely replace value from linked data by local one.
  - `DIFFERENCE` Difference – Store difference to linked data value.
  - `FACTOR` Factor – Store factor to linked data value (useful e.g. for scale).

bpy.ops.ui.**reloadtranslation()**

Force a full reload of UI translation

bpy.ops.ui.**reset_default_button(*, all=True)**

Reset this property's value to its default value

PARAMETERS:

all (*boolean, (optional)*) – All, Reset to default values all elements of the array

bpy.ops.ui.**unset_property_button()**

Clear the property and use default or generated value in operators

bpy.ops.ui.**view_drop()**

Drag and drop onto a data-set or item within the data-set

bpy.ops.ui.**view_item_rename()**

Rename the active item in the data-set view

bpy.ops.ui.**view_scroll()**

Undocumented, consider contributing.

bpy.ops.ui.**view_start_filter()**

Start entering filter text for the data-set in focus