Gizmo(bpy_struct)

base class — `bpy_struct`

class bpy.types.**Gizmo(bpy_struct)**

Collection of gizmos

alpha

TYPE:

float in [0, 1], default 0.0

alpha_highlight

TYPE:

float in [0, 1], default 0.0

bl_idname

TYPE:

string, default "", (never None)

color

TYPE:

`mathutils.Color` of 3 items in [0, inf], default (0.0, 0.0, 0.0)

color_highlight

TYPE:

`mathutils.Color` of 3 items in [0, inf], default (0.0, 0.0, 0.0)

group

Gizmo group this gizmo is a member of

TYPE:

`GizmoGroup` , (readonly)

hide

TYPE:

boolean, default False

hide_keymap

Ignore the key-map for this gizmo

TYPE:

boolean, default False

hide_select

TYPE:

boolean, default False

is_highlight

TYPE:

boolean, default False, (readonly)

is_modal

TYPE:

boolean, default False, (readonly)

line_width

TYPE:

float in [0, inf], default 0.0

matrix_basis

TYPE:

`mathutils.Matrix` of 4 * 4 items in [-inf, inf], default ((0.0, 0.0, 0.0, 0.0), (0.0, 0.0, 0.0, 0.0), (0.0, 0.0, 0.0, 0.0), (0.0, 0.0, 0, 0.0))

matrix_offset

TYPE:

`mathutils.Matrix` of 4 * 4 items in [-inf, inf], default ((0.0, 0.0, 0.0, 0.0), (0.0, 0.0, 0.0, 0.0), (0.0, 0.0, 0.0, 0.0), (0.0, 0.0, 0, 0.0))

matrix_space

TYPE:

`mathutils.Matrix` of 4 * 4 items in [-inf, inf], default ((0.0, 0.0, 0.0, 0.0), (0.0, 0.0, 0.0, 0.0), (0.0, 0.0, 0.0, 0.0), (0.0, 0.0, 0, 0.0))

matrix_world

TYPE:

`mathutils.Matrix` of 4 * 4 items in [-inf, inf], default ((0.0, 0.0, 0.0, 0.0), (0.0, 0.0, 0.0, 0.0), (0.0, 0.0, 0.0, 0.0), (0.0, 0.0, 0, 0.0)), (readonly)

properties

TYPE:

`GizmoProperties`, (readonly, never None)

scale_basis

TYPE:

float in [0, inf], default 0.0

select

TYPE:

boolean, default False

select_bias

Depth bias used for selection

TYPE:

float in [-inf, inf], default 0.0

use_draw_hover

TYPE:

boolean, default False

use_draw_modal

Show while dragging

TYPE:

boolean, default False

use_draw_offset_scale

use_draw_offset_scale

Scale the offset matrix (use to apply screen-space offset)

TYPE:

boolean, default False

use_draw_scale

Use scale when calculating the matrix

TYPE:

boolean, default True

use_draw_value

Show an indicator for the current value while dragging

TYPE:

boolean, default False

use_event_handle_all

When highlighted, do not pass events through to be handled by other keymaps

TYPE:

boolean, default False

use_grab_cursor

TYPE:

boolean, default False

use_operator_tool_properties

Merge active tool properties on activation (does not overwrite existing)

TYPE:

boolean, default False

use_select_background

Don't write into the depth buffer

TYPE:

boolean, default False

use_tooltip

Use tooltips when hovering over this gizmo

TYPE:

boolean, default True

draw(context)

draw_select(context, *, select_id=0)

test_select(context, location)

PARAMETERS:

location (*int array of 2 items in [-inf, inf], (never None)*) – Location, Region coordinates

RETURNS:

Use -1 to skip this gizmo

RETURN TYPE:

int in [-1, inf]

int in [-1, inf]

modal(context, event, tweak)

>   **PARAMETERS:**
>
> >   **tweak** (*enum set in {'PRECISE', 'SNAP'}*) – Tweak
>
>   **RETURNS:**
>
> >   result
>
>   **RETURN TYPE:**
>
> >   enum set in Operator Return Items

setup()

invoke(context, event)

>   **RETURNS:**
>
> >   result
>
>   **RETURN TYPE:**
>
> >   enum set in Operator Return Items

exit(context, cancel)

>   **PARAMETERS:**
>
> >   **cancel** (*boolean*) – Cancel, otherwise confirm

select_refresh()

draw_preset_box(matrix, *, select_id=-1)

>   Draw a box
>
>   **PARAMETERS:**
>
> - **matrix** (`mathutils.Matrix` of 4 * 4 items in [-inf, inf]) – The matrix to transform
> - **select_id** (*int in [-1, inf], (optional)*) – ID to use when gizmo is selectable. Use -1 when not selecting.

draw_preset_arrow(matrix, *, axis='POS_Z', select_id=-1)

>   Draw a box
>
>   **PARAMETERS:**
>
> - **matrix** (`mathutils.Matrix` of 4 * 4 items in [-inf, inf]) – The matrix to transform
> - **axis** (enum in Object Axis Items, (optional)) – Arrow Orientation
> - **select_id** (*int in [-1, inf], (optional)*) – ID to use when gizmo is selectable. Use -1 when not selecting.

draw_preset_circle(matrix, *, axis='POS_Z', select_id=-1)

>   Draw a box
>
>   **PARAMETERS:**
>
> - **matrix** (`mathutils.Matrix` of 4 * 4 items in [-inf, inf]) – The matrix to transform
> - **axis** (enum in Object Axis Items, (optional)) – Arrow Orientation
> - **select_id** (*int in [-1, inf], (optional)*) – ID to use when gizmo is selectable. Use -1 when not selecting.

target_set_prop(target, data, property, *, index=-1)

>   **PARAMETERS:**
>
> - **target** (*string, (never None)*) – Target property
> - **data** (`AnyType`, (never None)) – Data from which to take property
> - **property** (*string, (never None)*) – Identifier of property in data

target_set_operator(operator, *, index=0)

Operator to run when activating the gizmo (overrides property targets)

PARAMETERS:

- **operator** (*string, (never None)*) – Target operator
- **index** (*int in [0, 255], (optional)*) – Part index

RETURNS:

Operator properties to fill in

RETURN TYPE:

`OperatorProperties`

target_is_valid(property)

PARAMETERS:

property (*string, (never None)*) – Property identifier

RETURN TYPE:

boolean

draw_custom_shape(shape, *, matrix=None, select_id=None)

Draw a shape created form `Gizmo.draw_custom_shape`.

PARAMETERS:

- **shape** (*Any*) – The cached shape to draw.
- **matrix** (`mathutils.Matrix`) – 4x4 matrix, when not given `Gizmo.matrix_world` is used.
- **select_id** (*int*) – The selection id. Only use when drawing within `Gizmo.draw_select`.

static new_custom_shape(type, verts)

Create a new shape that can be passed to `Gizmo.draw_custom_shape`.

PARAMETERS:

- **type** (*str*) – The type of shape to create in (POINTS, LINES, TRIS, LINE_STRIP).
- **verts** (*Sequence[Sequence[float]]*) – Sequence of 2D or 3D coordinates.

RETURNS:

The newly created shape (the return type make change).

RETURN TYPE:

Any

classmethod bl_rna_get_subclass(id, default=None)

PARAMETERS:

id (*str*) – The RNA type identifier.

RETURNS:

The RNA type or default when not found.

RETURN TYPE:

`bpy.types.Struct` subclass

classmethod bl_rna_get_subclass_py(id, default=None)

PARAMETERS:

id (*str*) – The RNA type identifier.

RETURNS:

The class or default when not found.

RETURN TYPE:

type

target_get_range(target):

Get the range for this target property.

> **PARAMETERS:**
>> **target** – Target property name.
>
> **RETURNS:**
>> The range of this property (min, max).
>
> **RETURN TYPE:**
>> tuple[float, float]

target_get_value(target):

Get the value of this target property.

> **PARAMETERS:**
>> **target** (*str*) – Target property name.
>
> **RETURNS:**
>> The value of the target property as a value or array based on the target type.
>
> **RETURN TYPE:**
>> float | tuple[float, …]

target_set_handler(target, get, set, range=None):

Assigns callbacks to a gizmos property.

> **PARAMETERS:**
> - **target** (*str*) – Target property name.
> - **get** (*Callable[[], float | Sequence[float]]*) – Function that returns the value for this property (single value or sequence).
> - **set** (*Callable[[tuple[float, ...]], Any]*) – Function that takes a single value argument and applies it.
> - **range** (*callable*) – Function that returns a (min, max) tuple for gizmos that use a range. The returned value is not used.

target_set_value(target):

Set the value of this target property.

> **PARAMETERS:**
>> **target** (*str*) – Target property name.

Inherited Properties

- bpy_struct.id_data

Inherited Functions

- bpy_struct.as_pointer
- bpy_struct.driver_add
- bpy_struct.driver_remove
- bpy_struct.get
- bpy_struct.id_properties_clear
- bpy_struct.id_properties_ensure
- bpy_struct.id_properties_ui
- bpy_struct.is_property_hidden

- bpy_struct.items
- bpy_struct.keyframe_delete
- bpy_struct.keyframe_insert
- bpy_struct.keys
- bpy_struct.path_from_id
- bpy_struct.path_resolve
- bpy_struct.pop
- bpy_struct.property_overridable_library_set

References

- GizmoGroup.gizmos
- GizmoGroup.invoke_prepare
- Gizmos.new
- Gizmos.remove

Previous
GeometryNodeWarning(GeometryNode)

Report issue on this page

Copyright © Blender Authors
Made with Furo

Ne
GizmoGroup(bpy_stru