Window(bpy_struct)

base class — `bpy_struct`

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

Open window

height

Window height

TYPE:

int in [0, 32767], default 0, (readonly)

modal_operators

A list of currently running modal operators

TYPE:

`bpy_prop_collection` of `Operator`, (readonly)

parent

Active workspace and scene follow this window

TYPE:

`Window`, (readonly)

scene

Active scene to be edited in the window

TYPE:

`Scene`, (never None)

screen

Active workspace screen showing in the window

TYPE:

`Screen`, (never None)

stereo_3d_display

Settings for stereo 3D display

TYPE:

`Stereo3dDisplay`, (readonly, never None)

view_layer

The active workspace view layer showing in the window

TYPE:

`ViewLayer`, (never None)

width

Window width

TYPE:

int in [0, 32767], default 0, (readonly)

workspace

Active workspace showing in the window

TYPE:
    `WorkSpace`, (never None)

x

Horizontal location of the window

TYPE:
    int in [-32768, 32767], default 0, (readonly)

y

Vertical location of the window

TYPE:
    int in [-32768, 32767], default 0, (readonly)

cursor_warp(x, y)

Set the cursor position

cursor_set(cursor)

Set the cursor

PARAMETERS:
    cursor (enum in Window Cursor Items) – cursor

cursor_modal_set(cursor)

Set the cursor, so the previous cursor can be restored

PARAMETERS:
    cursor (enum in Window Cursor Items) – cursor

cursor_modal_restore()

Restore the previous cursor after calling `cursor_modal_set`

event_simulate(type, value, *, unicode='', x=0, y=0, shift=False, ctrl=False, alt=False, oskey=False)

event_simulate

PARAMETERS:
- **type** (enum in Event Type Items) – Type
- **value** (enum in Event Value Items) – Value
- **shift** (*boolean, (optional)*) – Shift
- **ctrl** (*boolean, (optional)*) – Ctrl
- **alt** (*boolean, (optional)*) – Alt
- **oskey** (*boolean, (optional)*) – OS Key

RETURNS:
    Item, Added key map item

RETURN TYPE:
    `Event`

classmethod bl_rna_get_subclass(id, default=None)

PARAMETERS:
    id (*str*) – The RNA type identifier.

RETURNS:
    The RNA type or default when not found

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

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.is_property_overridable_library`
- `bpy_struct.is_property_readonly`
- `bpy_struct.is_property_set`
- `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`
- `bpy_struct.property_unset`
- `bpy_struct.type_recast`
- `bpy_struct.values`

References

- `Context.window`
- `Window.parent`
- `WindowManager.event_timer_add`
- `WindowManager.windows`