KeyMapItem(bpy_struct)

base class — `bpy_struct`

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

Item in a Key Map

active

Activate or deactivate item

TYPE:

boolean, default False

alt

Alt key pressed, -1 for any state

TYPE:

int in [-1, 1], default 0

alt_ui

Alt key pressed

TYPE:

boolean, default False

any

Any modifier keys pressed

TYPE:

boolean, default False

ctrl

Control key pressed, -1 for any state

TYPE:

int in [-1, 1], default 0

ctrl_ui

Control key pressed

TYPE:

boolean, default False

direction

The direction (only applies to drag events)

TYPE:

enum in Event Direction Items, default 'ANY'

id

ID of the item

TYPE:

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

idname

Identifier of operator to call on input event

TYPE:

string, default "", (never None)

is_user_defined

Is this keymap item user defined (doesn't just replace a builtin item)

TYPE:

boolean, default False, (readonly)

is_user_modified

Is this keymap item modified by the user

TYPE:

boolean, default False, (readonly)

key_modifier

Regular key pressed as a modifier

TYPE:

enum in Event Type Items, default 'NONE'

map_type

Type of event mapping

TYPE:

enum in ['KEYBOARD', 'MOUSE', 'NDOF', 'TEXTINPUT', 'TIMER'], default 'KEYBOARD'

name

Name of operator (translated) to call on input event

TYPE:

string, default "", (readonly, never None)

oskey

Operating system key pressed, -1 for any state

TYPE:

int in [-1, 1], default 0

oskey_ui

Operating system key pressed

TYPE:

boolean, default False

properties

Properties to set when the operator is called

TYPE:

`OperatorProperties`, (readonly)

propvalue

The value this event translates to in a modal keymap

TYPE:

enum in Keymap Propvalue Items, default 'NONE'

repeat

Active on key-repeat events (when a key is held)

> **TYPE:**
> boolean, default False

shift

Shift key pressed, -1 for any state

> **TYPE:**
> int in [-1, 1], default 0

shift_ui

Shift key pressed

> **TYPE:**
> boolean, default False

show_expanded

Show key map event and property details in the user interface

> **TYPE:**
> boolean, default False

type

Type of event

> **TYPE:**
> enum in Event Type Items, default 'NONE'

value

> **TYPE:**
> enum in Event Value Items, default 'NOTHING'

compare(item)

compare

> **PARAMETERS:**
> **item** (`KeyMapItem`) – Item
>
> **RETURNS:**
> Comparison result
>
> **RETURN TYPE:**
> boolean

to_string(*, compact=False)

to_string

> **PARAMETERS:**
> **compact** (*boolean, (optional)*) – Compact
>
> **RETURNS:**
> result
>
> **RETURN TYPE:**
> string, (never None)

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

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

- `KeyConfigurations.find_item_from_operator`
- `KeyMap.keymap_items`
- `KeyMap.restore_item_to_default`
- `KeyMapItem.compare`
- `KeyMapItems.find_from_operator`
- `KeyMapItems.from_id`
- `KeyMapItems.match_event`
- `KeyMapItems.new`
- `KeyMapItems.new_from_item`
- `KeyMapItems.new_from_item`
- `KeyMapItems.new_modal`
- `KeyMapItems.remove`
- `UILayout.template_event_from_keymap_item`
- `UILayout.template_keymap_item_properties`

KeyMap(bpy_struct)

Made with Furo