ActionConstraint(Constraint)

base classes — bpy_struct, Constraint

class bpy.types.**ActionConstraint(Constraint)**

Map an action to the transform axes of a bone

action

The constraining action

TYPE:

Action

action_slot

The slot identifies which sub-set of the Action is considered to be for this strip, and its name is used to find the right slot when assigning anothe Action

TYPE:

ActionSlot

action_slot_handle

A number that identifies which sub-set of the Action is considered to be for this Action Constraint

TYPE:

int in [-inf, inf], default 0

action_suitable_slots

The list of action slots suitable for this NLA strip

TYPE:

bpy_prop_collection of ActionSlot, (readonly)

eval_time

Interpolates between Action Start and End frames

TYPE:

float in [0, 1], default 0.0

frame_end

Last frame of the Action to use

TYPE:

int in [-1048574, 1048574], default 0

frame_start

First frame of the Action to use

TYPE:

int in [-1048574, 1048574], default 0

last_slot_identifier

The identifier of the most recently assigned action slot. The slot identifies which sub-set of the Action is considered to be for this constraint, an its identifier is used to find the right slot when assigning an Action.

TYPE:

string, default “”, (never None)

max

Maximum value for target channel range

TYPE:

float in [-1000, 1000], default 0.0

min

Minimum value for target channel range

TYPE:

float in [-1000, 1000], default 0.0

mix_mode

Specify how existing transformations and the action channels are combined

- `BEFORE_FULL` Before Original (Full) – Apply the action channels before the original transformation, as if applied to an imaginary pare in Full Inherit Scale mode. Will create shear when combining rotation and non-uniform scale..
- `BEFORE` Before Original (Aligned) – Apply the action channels before the original transformation, as if applied to an imaginary parent in Aligned Inherit Scale mode. This effectively uses Full for location and Split Channels for rotation and scale..
- `BEFORE_SPLIT` Before Original (Split Channels) – Apply the action channels before the original transformation, handling location, rotation and scale separately.
- `AFTER_FULL` After Original (Full) – Apply the action channels after the original transformation, as if applied to an imaginary child in Fu Inherit Scale mode. Will create shear when combining rotation and non-uniform scale..
- `AFTER` After Original (Aligned) – Apply the action channels after the original transformation, as if applied to an imaginary child in Aligne Inherit Scale mode. This effectively uses Full for location and Split Channels for rotation and scale..
- `AFTER_SPLIT` After Original (Split Channels) – Apply the action channels after the original transformation, handling location, rotation and scale separately.

TYPE:

enum in ['BEFORE_FULL', 'BEFORE', 'BEFORE_SPLIT', 'AFTER_FULL', 'AFTER', 'AFTER_SPLIT'], default 'AFTER_FULI

subtarget

Armature bone, mesh or lattice vertex group, …

TYPE:

string, default "", (never None)

target

Target object

TYPE:

`Object`

transform_channel

Transformation channel from the target that is used to key the Action

TYPE:

enum in ['LOCATION_X', 'LOCATION_Y', 'LOCATION_Z', 'ROTATION_X', 'ROTATION_Y', 'ROTATION_Z', 'SCALE_X', 'SCALE_Y', 'SCALE_Z'], default 'ROTATION_X'

use_bone_object_action

Bones only: apply the object's transformation channels of the action to the constrained bone, instead of bone's channels

TYPE:

boolean, default False

use_eval_time

use_eval_time

Interpolate between Action Start and End frames, with the Evaluation Time slider instead of the Target object/bone

TYPE:

boolean, default False

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`
- `Constraint.name`
- `Constraint.type`
- `Constraint.is_override_data`
- `Constraint.owner_space`
- `Constraint.target_space`
- `Constraint.space_object`
- `Constraint.space_subtarget`
- `Constraint.mute`
- `Constraint.enabled`
- `Constraint.show_expanded`
- `Constraint.is_valid`
- `Constraint.active`
- `Constraint.influence`
- `Constraint.error_location`
- `Constraint.error_rotation`

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`
- `Constraint.bl_rna_get_subclass`
- `Constraint.bl_rna_get_subclass_py`