bpy_struct

subclasses — [AOV](), [AOVs](), [ActionChannelbag](), [ActionChannelbagFCurves](), [ActionChannelbagGroups](),
[ActionChannelbags](), [ActionFCurves](), [ActionGroup](), [ActionGroups](), [ActionLayer](), [ActionLayers](),
[ActionPoseMarkers](), [ActionSlot](), [ActionSlots](), [ActionStrip](), [ActionStrips](), [Addon](), [AddonPreferences](),
[Addons](), [AnimData](), [AnimDataDrivers](), [AnimViz](), [AnimVizMotionPaths](), [AnyType](), [Area](), [AreaSpaces](),
[ArmatureBones](), [ArmatureConstraintTargets](), [ArmatureEditBones](), [AssetCatalogPath](),
[AssetLibraryCollection](), [AssetLibraryReference](), [AssetMetaData](), [AssetRepresentation](), [AssetShelf](),
[AssetTag](), [AssetTags](), [AssetWeakReference](), [Attribute](), [AttributeGroupCurves](),
[AttributeGroupGreasePencil](), [AttributeGroupGreasePencilDrawing](), [AttributeGroupMesh](),
[AttributeGroupPointCloud](), [BakeSettings](), [BezierSplinePoint](), [BlendData](), [BlendDataActions](),
[BlendDataArmatures](), [BlendDataBrushes](), [BlendDataCacheFiles](), [BlendDataCameras](),
[BlendDataCollections](), [BlendDataCurves](), [BlendDataFonts](), [BlendDataGreasePencils](),
[BlendDataGreasePencilsV3](), [BlendDataHairCurves](), [BlendDataImages](), [BlendDataLattices](),
[BlendDataLibraries](), [BlendDataLights](), [BlendDataLineStyles](), [BlendDataMasks](), [BlendDataMaterials](),
[BlendDataMeshes](), [BlendDataMetaBalls](), [BlendDataMovieClips](), [BlendDataNodeTrees](), [BlendDataObjects](),
[BlendDataPaintCurves](), [BlendDataPalettes](), [BlendDataParticles](), [BlendDataPointClouds](),
[BlendDataProbes](), [BlendDataScenes](), [BlendDataScreens](), [BlendDataSounds](), [BlendDataSpeakers](),
[BlendDataTexts](), [BlendDataTextures](), [BlendDataVolumes](), [BlendDataWindowManagers](),
[BlendDataWorkSpaces](), [BlendDataWorlds](), [BlendImportContext](), [BlendImportContextItem](),
[BlendImportContextItems](), [BlendImportContextLibraries](), [BlendImportContextLibrary](), [BlenderRNA](),
[BoidRule](), [BoidSettings](), [BoidState](), [Bone](), [BoneCollection](), [BoneCollectionMemberships](),
[BoneCollections](), [BoneColor](), [BoolAttributeValue](), [BrushCapabilities](), [BrushCapabilitiesImagePaint](),
[BrushCapabilitiesSculpt](), [BrushCapabilitiesVertexPaint](), [BrushCapabilitiesWeightPaint](),
[BrushCurvesSculptSettings](), [BrushGpencilSettings](), [ByteColorAttributeValue](),
[ByteIntAttributeValue](), [CacheFileLayer](), [CacheFileLayers](), [CacheObjectPath](), [CacheObjectPaths](),
[CameraBackgroundImage](), [CameraBackgroundImages](), [CameraDOFSettings](), [CameraStereoData](),
[ChannelDriverVariables](), [ChildParticle](), [ClothCollisionSettings](), [ClothSettings](),
[ClothSolverResult](), [CollectionChild](), [CollectionChildren](), [CollectionExport](),
[CollectionLightLinking](), [CollectionObject](), [CollectionObjects](), [CollisionSettings](),
[ColorManagedDisplaySettings](), [ColorManagedInputColorspaceSettings](),
[ColorManagedSequencerColorspaceSettings](), [ColorManagedViewSettings](), [ColorMapping](), [ColorRamp](),
[ColorRampElement](), [ColorRampElements](), [CompositorNodeOutputFileFileSlots](),
[CompositorNodeOutputFileLayerSlots](), [ConsoleLine](), [Constraint](), [ConstraintTarget](),
[ConstraintTargetBone](), [Context](), [CryptomatteEntry](), [CurveMap](), [CurveMapPoint](), [CurveMapPoints](),
[CurveMapping](), [CurvePaintSettings](), [CurvePoint](), [CurveProfile](), [CurveProfilePoint](),
[CurveProfilePoints](), [CurveSlice](), [CurveSplines](), [Depsgraph](), [DepsgraphObjectInstance](),
[DepsgraphUpdate](), [DisplaySafeAreas](), [DopeSheet](), [Driver](), [DriverTarget](), [DriverVariable](),
[DynamicPaintBrushSettings](), [DynamicPaintCanvasSettings](), [DynamicPaintSurface](),
[DynamicPaintSurfaces](), [EQCurveMappingData](), [EditBone](), [EffectorWeights](), [EnumPropertyItem](), [Event](),
[FCurve](), [FCurveKeyframePoints](), [FCurveModifiers](), [FCurveSample](), [FFmpegSettings](), [FModifier](),
[FModifierEnvelopeControlPoint](), [FModifierEnvelopeControlPoints](), [FieldSettings](),
[FileAssetSelectIDFilter](), [FileBrowserFSMenuEntry](), [FileHandler](), [FileSelectEntry](),
[FileSelectIDFilter](), [FileSelectParams](), [Float2AttributeValue](), [Float4x4AttributeValue](),
[FloatAttributeValue](), [FloatColorAttributeValue](), [FloatVectorAttributeValue](),
[FloatVectorValueReadOnly](), [FluidDomainSettings](), [FluidEffectorSettings](), [FluidFlowSettings](),
[ForeachGeometryElementGenerationItem](), [ForeachGeometryElementInputItem](),
[ForeachGeometryElementMainItem](), [FreestyleLineSet](), [FreestyleModuleSettings](), [FreestyleModules](),
[FreestyleSettings](), [Function](), [GPencilFrame](), [GPencilFrames](), [GPencilInterpolateSettings](),
[GPencilLayer](), [GPencilSculptGuide](), [GPencilSculptSettings](), [GPencilStroke](), [GPencilStrokePoint](),
[Gizmo](), [GizmoGroup](), [GizmoGroupProperties](), [GizmoProperties](), [Gizmos](),
[GreasePencilDashModifierSegment](), [GreasePencilDrawing](), [GreasePencilFrame](), [GreasePencilFrames](),
[GreasePencilLayer](), [GreasePencilLayerGroup](), [GreasePencilLayerMask](), [GreasePencilLayerMasks](),
[GreasePencilLayers](), [GreasePencilTimeModifierSegment](), [GreasePencilv3LayerGroup](),
[GreasePencilv3Layers](), [Header](), [Histogram](), [ID](), [IDMaterials](), [IDOverrideLibrary](),

IDOverrideLibraryProperties, IDOverrideLibraryProperty, IDOverrideLibraryPropertyOperation, IDOverrideLibraryPropertyOperations, IDPropertyWrapPtr, IKParam, ImageFormatSettings, ImagePackedFile, ImagePreview, ImageUser, IndexSwitchItem, Int2AttributeValue, IntAttributeValue, KeyConfig, KeyConfigPreferences, KeyConfigurations, KeyMap, KeyMapItem, KeyMapItems, KeyMaps, Keyframe, KeyingSet, KeyingSetInfo, KeyingSetPath, KeyingSetPaths, KeyingSets, KeyingSetsAll, LatticePoint, LayerCollection, LayerObjects, LayoutPanelState, LibraryWeakReference, Lightgroup, Lightgroups, LineStyleAlphaModifiers, LineStyleColorModifiers, LineStyleGeometryModifiers, LineStyleModifier, LineStyleTextureSlots LineStyleThicknessModifiers, Linesets, LoopColors, Macro, MaskLayer, MaskLayers, MaskParent, MaskSpline, MaskSplinePoint, MaskSplinePointUW, MaskSplinePoints, MaskSplines, MaterialGPencilStyle, MaterialLineArt, MaterialSlot, Menu, MeshEdge, MeshEdges, MeshLoop, MeshLoopColor, MeshLoopColorLayer, MeshLoopTriangle, MeshLoopTriangles, MeshLoops, MeshNormalValue, MeshPolygon, MeshPolygons, MeshSkinVertex, MeshSkinVertexLayer, MeshStatVis, MeshUVLoop, MeshUVLoopLayer, MeshVertex, MeshVertices, MetaBallElements, MetaElement, Modifier, MotionPath, MotionPathVert, MovieClipProxy, MovieClipScopes, MovieClipUser, MovieReconstructedCamera, MovieTracking, MovieTrackingCamera, MovieTrackingDopesheet, MovieTrackingMarker, MovieTrackingMarkers, MovieTrackingObject, MovieTrackingObjectPlaneTracks, MovieTrackingObjectTracks, MovieTrackingObjects, MovieTrackingPlaneMarker, MovieTrackingPlaneMarkers, MovieTrackingPlaneTrack, MovieTrackingPlaneTracks, MovieTrackingReconstructedCameras, MovieTrackingReconstruction, MovieTrackingSettings, MovieTrackingStabilization, MovieTrackingTrack, MovieTrackingTracks, NlaStrip, NlaStripFCurves, NlaStrips, NlaTrack, NlaTracks, Node, NodeEnumItem, NodeGeometryBakeItem, NodeGeometryBakeItems, NodeGeometryCaptureAttributeItem, NodeGeometryCaptureAttributeItems, NodeGeometryForeachGeometryElementGenerationItems, NodeGeometryForeachGeometryElementInputItems, NodeGeometryForeachGeometryElementMainItems, NodeGeometryRepeatOutputItems, NodeGeometrySimulationOutputItems, NodeIndexSwitchItems, NodeInputs, NodeInstanceHash, NodeInternalSocketTemplate, NodeLink, NodeLinks, NodeMenuSwitchItems, NodeOutputFileSlotFile, NodeOutputFileSlotLayer, NodeOutputs, NodeSocket NodeTreeInterface, NodeTreeInterfaceItem, NodeTreePath, Nodes, NodesModifierBake, NodesModifierBakeDataBlocks, NodesModifierBakes, NodesModifierDataBlock, NodesModifierPanel, NodesModifierPanels, NodesModifierWarning, ObjectBase, ObjectConstraints, ObjectDisplay, ObjectLightLinking, ObjectLineArt, ObjectModifiers, ObjectShaderFx, Operator, OperatorMacro, OperatorOptions, OperatorProperties, PackedFile, Paint, PaintModeSettings, PaletteColor, PaletteColors, Panel, Particle, ParticleBrush, ParticleDupliWeight, ParticleEdit, ParticleHairKey, ParticleKey, ParticleSettingsTextureSlots, ParticleSystem, ParticleSystems, ParticleTarget, PathCompare, PathCompareCollection, Point, PointCache, PointCacheItem, PointCaches, Pose, PoseBone, PoseBoneConstraints, Preferences, PreferencesApps, PreferencesEdit PreferencesExperimental, PreferencesExtensions, PreferencesFilePaths, PreferencesInput, PreferencesKeymap, PreferencesSystem, PreferencesView, PrimitiveBoolean, PrimitiveFloat, PrimitiveInt, PrimitiveString, Property, PropertyGroup, PropertyGroupItem, QuaternionAttributeValue, RaytraceEEVEE, ReadOnlyInteger, Region, RegionView3D, RenderEngine, RenderLayer, RenderPass, RenderPasses, RenderResult, RenderSettings, RenderSlot, RenderSlots, RenderView, RenderViews, RepeatItem, RetimingKey, RetimingKeys, RigidBodyConstraint, RigidBodyObject, RigidBodyWorld, SPHFluidSettings, SceneDisplay, SceneEEVEE, SceneGpencil, SceneHydra, SceneObjects, SceneRenderView, Scopes, ScriptDirectory, ScriptDirectoryCollection, SequenceEditor, SequenceTimelineChannel, SequencerCacheOverlay, SequencerPreviewOverlay, SequencerTimelineOverlay, SequencerToolSettings, ShaderFx, ShapeKey, ShapeKeyBezierPoint, ShapeKeyCurvePoint, ShapeKeyPoint, Short2AttributeValue, SimulationStateItem, SoftBodySettings Space, SpaceImageOverlay, SpaceNodeEditorPath, SpaceNodeOverlay, SpaceUVEditor, Spline, SplineBezierPoints, SplinePoint, SplinePoints, SpreadsheetColumn, SpreadsheetColumnID, SpreadsheetRowFilter, Stereo3dDisplay, Stereo3dFormat, StringAttributeValue, Strip, StripColorBalanceData, StripCrop, StripElement, StripElements, StripModifier, StripModifiers, StripProxy, StripTransform, StripsMeta, StripsTopLevel, Struct, StudioLight, StudioLights, TexMapping, TexPaintSlot, TextBox, TextCharacterFormat, TextLine, TextureSlot, Theme, ThemeAssetShelf, ThemeBoneColorSet, ThemeClipEditor, ThemeCollectionColor, ThemeConsole, ThemeDopeSheet, ThemeFileBrowser, ThemeFontStyle, ThemeGradientColors, ThemeGraphEditor,

ThemeImageEditor, ThemeInfo, ThemeNLAEditor, ThemeNodeEditor, ThemeOutliner, ThemePanelColors, ThemePreferences, ThemeProperties, ThemeSequenceEditor, ThemeSpaceGeneric, ThemeSpaceGradient ThemeSpaceListGeneric, ThemeSpreadsheet, ThemeStatusBar, ThemeStripColor, ThemeStyle, ThemeTextEditor, ThemeTopBar, ThemeUserInterface, ThemeView3D, ThemeWidgetColors, ThemeWidgetStateColors, TimelineMarker, TimelineMarkers, Timer, ToolSettings, TransformOrientation, TransformOrientationSlot, UDIMTile, UDIMTiles, UILayout, UIList, UIPieMenu, UIPopover, UIPopupMenu, USDHook, UVLoopLayers, UVProjector, UnifiedPaintSettings, UnitSettings, UnknownType, UserAssetLibrary, UserExtensionRepo, UserExtensionRepoCollection, UserSolidLight, UvSculpt, VertexGroup, VertexGroupElement, VertexGroups, View2D, View3DCursor, View3DOverlay, View3DShading, ViewLayer, ViewLayerEEVEE, ViewLayers, ViewerPath, ViewerPathElem VolumeDisplay, VolumeGrid, VolumeGrids, VolumeRender, WalkNavigation, Window, WorkSpaceTool, WorldLighting, WorldMistSettings, XrActionMap, XrActionMapBinding, XrActionMapBindings, XrActionMapItem, XrActionMapItems, XrActionMaps, XrComponentPath, XrComponentPaths, XrEventData, XrSessionSettings, XrSessionState, XrUserPath, XrUserPaths, wmOwnerID, wmOwnerIDs, wmTools

class bpy.types.**bpy_struct**

    built-in base class for all classes in bpy.types.

> **Note**
>
> Note that `bpy.types.bpy_struct` is not actually available from within Blender, it only exists for the purpose of documentation.

    as_pointer()

        Returns the memory address which holds a pointer to Blender's internal data

        RETURNS:

            int (memory address).

        RETURN TYPE:

            int

> **Note**
>
> This is intended only for advanced script writers who need to pass blender data to their own C/Python modules.

    driver_add(path, index=-1)

        Adds driver(s) to the given property

        PARAMETERS:

- **path** (*str*) – path to the property to drive, analogous to the fcurve's data path.
- **index** (*int*) – array index of the property drive. Defaults to -1 for all indices or a single channel if the property is not an array.

        RETURNS:

            The driver added or a list of drivers when index is -1.

        RETURN TYPE:

            `bpy.types.FCurve` | list[`bpy.types.FCurve`]

    driver_remove(path, index=-1)

        Remove driver(s) from the given property

        PARAMETERS:

- **path** (*str*) – path to the property to drive, analogous to the fcurve's data path.
- **index** (*int*) – array index of the property drive. Defaults to -1 for all indices or a single channel if the property is not an array.

        RETURNS:

            Success of driver removal.

        RETURN TYPE:

bool

get(key, default=None)

Returns the value of the custom property assigned to key or default when not found (matches Python's dictionary function of the same name).

PARAMETERS:

- **key** (*str*) – The key associated with the custom property.
- **default** (*Any*) – Optional argument for the value to return if *key* is not found.

> Note
>
> Only the `bpy.types.ID`, `bpy.types.Bone` and `bpy.types.PoseBone` classes support custom properties.

id_properties_clear()

RETURNS:

Remove the parent group for an RNA struct's custom IDProperties.

id_properties_ensure()

RETURNS:

the parent group for an RNA struct's custom IDProperties.

RETURN TYPE:

`idprop.types.IDPropertyGroup`

id_properties_ui(key)

RETURNS:

Return an object used to manage an IDProperty's UI data.

PARAMETERS:

key – String name of the property.

RETURN TYPE:

`bpy.types.IDPropertyUIManager`

is_property_hidden(property)

Check if a property is hidden.

RETURNS:

True when the property is hidden.

RETURN TYPE:

bool

is_property_overridable_library(property)

Check if a property is overridable.

RETURNS:

True when the property is overridable.

RETURN TYPE:

bool

is_property_readonly(property)

Check if a property is readonly.

RETURNS:

True when the property is readonly (not writable).

RETURN TYPE:

RETURN TYPE:

    bool

is_property_set(property, ghost=True)

    Check if a property is set, use for testing operator properties.

    PARAMETERS:

        ghost (*bool*) –

        Used for operators that re-run with previous settings. In this case the property is not marked as set, yet the value from the previous execution is used.

        In rare cases you may want to set this option to false.

    RETURNS:

        True when the property has been set.

    RETURN TYPE:

        bool

> **Note**
>
> Properties defined at run-time store the values of the properties as custom-properties.
>
> This method checks if the underlying data exists, causing the property to be considered *set*.
>
> A common pattern for operators is to calculate a value for the properties that have not had their values explicitly set by the caller (where the caller could be a key-binding, menu-items or Python script for example).
>
> In the case of executing operators multiple times, values are re-used from the previous execution.
>
> For example: subdividing a mesh with a smooth value of 1.0 will keep using that value on subsequent calls to subdivision, unless the operator is called with that property set to a different value.
>
> This behavior can be disabled using the `SKIP_SAVE` option when the property is declared (see: `bpy.props`).
>
> The `ghost` argument allows detecting how a value from a previous execution is handled.
>
> - When true: The property is considered unset even if the value from a previous call is used.
> - When false: The existence of any values causes `is_property_set` to return true.
>
> While this argument should typically be omitted, there are times when it's important to know if a value is anything besides the default.
>
> For example, the previous value may have been scaled by the scene's unit scale. In this case scaling the value multiple times would cause problems, so the `ghost` argument should be false.

items()

    Returns the items of this objects custom properties (matches Python's dictionary function of the same name).

    RETURNS:

        custom property key, value pairs.

    RETURN TYPE:

        `idprop.types.IDPropertyGroupViewItems`

> **Note**
>
> Only the `bpy.types.ID`, `bpy.types.Bone` and `bpy.types.PoseBone` classes support custom properties.

keyframe_delete(data_path, index=-1, frame=bpy.context.scene.frame_current, group='')

    Remove a keyframe from this properties fcurve.

    PARAMETERS:

        - **data_path** (*str*) – path to the property to remove a key, analogous to the fcurve's data path.

- **index** (*int*) – array index of the property to remove a key. Defaults to -1 removing all indices or a single channel if the property is not an array.
- **frame** (*float*) – The frame on which the keyframe is deleted, defaulting to the current frame.
- **group** (*str*) – The name of the group the F-Curve should be added to if it doesn't exist yet.

RETURNS:

Success of keyframe deletion.

RETURN TYPE:

bool

keyframe_insert(data_path, index=-1, frame=bpy.context.scene.frame_current, group='', options=set(), keytype='KEYFRAME')

Insert a keyframe on the property given, adding fcurves and animation data when necessary.

PARAMETERS:

- **data_path** (*str*) – path to the property to key, analogous to the fcurve's data path.
- **index** (*int*) – array index of the property to key. Defaults to -1 which will key all indices or a single channel if the property is not an array.
- **frame** (*float*) – The frame on which the keyframe is inserted, defaulting to the current frame.
- **group** (*str*) – The name of the group the F-Curve should be added to if it doesn't exist yet.
- **options** (*set[str]*) –

  Optional set of flags:

  - `INSERTKEY_NEEDED` Only insert keyframes where they're needed in the relevant F-Curves.
  - `INSERTKEY_VISUAL` Insert keyframes based on 'visual transforms'.
  - `INSERTKEY_XYZ_TO_RGB` This flag is no longer in use, and is here so that code that uses it doesn't break. The XYZ=RGB coloring is determined by the animation preferences.
  - `INSERTKEY_REPLACE` Only replace already existing keyframes.
  - `INSERTKEY_AVAILABLE` Only insert into already existing F-Curves.
  - `INSERTKEY_CYCLE_AWARE` Take cyclic extrapolation into account (Cycle-Aware Keying option).

- **keytype** (*str*) – Type of the key: 'KEYFRAME', 'BREAKDOWN', 'MOVING_HOLD', 'EXTREME', 'JITTER', or 'GENERATED'

RETURNS:

Success of keyframe insertion.

RETURN TYPE:

bool

This is the most simple example of inserting a keyframe from python.

```python
import bpy

obj = bpy.context.object

# set the keyframe at frame 1
obj.location = (3.0, 4.0, 10.0)
obj.keyframe_insert(data_path="location", frame=1)
```

Note that when keying data paths which contain nested properties this must be done from the `ID` subclass, in this case the `Armature` rather than the bone.

```python
import bpy
from bpy.props import (
    FloatProperty,
    PointerProperty,
)
```

```
# Define a nested property.
class MyPropGroup(bpy.types.PropertyGroup):
    nested: FloatProperty(name="Nested", default=0.0)


# Register it so its available for all bones.
bpy.utils.register_class(MyPropGroup)
bpy.types.Bone.my_prop = PointerProperty(
    type=MyPropGroup,
    name="MyProp",
)


# Get a bone.
obj = bpy.data.objects["Armature"]
arm = obj.data


# Set the keyframe at frame 1.
arm.bones["Bone"].my_prop.nested = 10
arm.keyframe_insert(
    data_path='bones["Bone"].my_prop.nested',
    frame=1,
    group="Nested Group",
)
```

keys()

Returns the keys of this objects custom properties (matches Python's dictionary function of the same name).

RETURNS:

custom property keys.

RETURN TYPE:

idprop.types.IDPropertyGroupViewKeys

> Note
>
> Only the `bpy.types.ID`, `bpy.types.Bone` and `bpy.types.PoseBone` classes support custom properties.

path_from_id(property='')

Returns the data path from the ID to this object (string).

PARAMETERS:

property (*str*) – Optional property name which can be used if the path is to a property of this object.

RETURNS:

The path from `bpy.types.bpy_struct.id_data` to this struct and property (when given).

RETURN TYPE:

str

path_resolve(path, coerce=True)

Returns the property from the path, raise an exception when not found.

PARAMETERS:

- **path** (*str*) – path which this property resolves.
- **coerce** (*bool*) – optional argument, when True, the property will be converted into its Python representation.

pop(key, default=None)

Remove and return the value of the custom property assigned to key or default when not found (matches Python's dictionary function of the same name).

> **PARAMETERS:**
> - **key** (*str*) – The key associated with the custom property.
> - **default** (*Any*) – Optional argument for the value to return if *key* is not found.

> > Note
> >
> > Only the `bpy.types.ID`, `bpy.types.Bone` and `bpy.types.PoseBone` classes support custom properties.

property_overridable_library_set(property, overridable)

Define a property as overridable or not (only for custom properties!).

> **RETURNS:**
> True when the overridable status of the property was successfully set.

> **RETURN TYPE:**
> bool

property_unset(property)

Unset a property, will use default value afterward.

type_recast()

Return a new instance, this is needed because types such as textures can be changed at runtime.

> **RETURNS:**
> a new instance of this object with the type initialized again.

> **RETURN TYPE:**
> `bpy.types.bpy_struct`

values()

Returns the values of this objects custom properties (matches Python's dictionary function of the same name).

> **RETURNS:**
> custom property values.

> **RETURN TYPE:**
> `idprop.types.IDPropertyGroupViewValues`

> > Note
> >
> > Only the `bpy.types.ID`, `bpy.types.Bone` and `bpy.types.PoseBone` classes support custom properties.

id_data

The `bpy.types.ID` object this datablock is from or None, (not available for all data types)