

# ShaderNodeTexSky(ShaderNode)

base classes — [bpy\\_struct](#), [Node](#), [NodeInternal](#), [ShaderNode](#)

**class** bpy.types.**ShaderNodeTexSky**(**ShaderNode**)

Generate a procedural sky texture

## air\_density

Density of air molecules. • 0 - No air. • 1 - Clear day atmosphere. • 2 - Highly polluted day

### TYPE:

float in [0, 10], default 1.0

## altitude

Height from sea level

### TYPE:

float in [0, 60000], default 0.0

## color\_mapping

Color mapping settings

### TYPE:

[ColorMapping](#), (readonly, never None)

## dust\_density

Density of dust molecules and water droplets. • 0 - No dust. • 1 - Clear day atmosphere. • 5 - City like atmosphere. • 10 - Hazy day

### TYPE:

float in [0, 10], default 1.0

## ground\_albedo

Ground color that is subtly reflected in the sky

### TYPE:

float in [0, 1], default 0.0

## ozone\_density

Density of ozone layer. • 0 - No ozone. • 1 - Clear day atmosphere. • 2 - City like atmosphere

### TYPE:

float in [0, 10], default 1.0

## sky\_type

Which sky model should be used

- `PREETHAM` Preetham – Preetham 1999.
- `HOSEK_WILKIE` Hosek / Wilkie – Hosek / Wilkie 2012.
- `NISHITA` Nishita – Nishita 1993 improved.

### TYPE:

enum in ['PREETHAM', 'HOSEK\_WILKIE', 'NISHITA'], default 'PREETHAM'

## sun\_direction

Direction from where the sun is shining

--- ---

**TYPE:**

`mathutils.Vector` of 3 items in  $[-\text{inf}, \text{inf}]$ , default (0.0, 0.0, 1.0)

**sun\_disc**

Include the sun itself in the output

**TYPE:**

boolean, default True

**sun\_elevation**

Sun angle from horizon

**TYPE:**

float in  $[-\text{inf}, \text{inf}]$ , default 1.5708

**sun\_intensity**

Strength of sun

**TYPE:**

float in  $[0, 1000]$ , default 1.0

**sun\_rotation**

Rotation of sun around zenith

**TYPE:**

float in  $[-\text{inf}, \text{inf}]$ , default 0.0

**sun\_size**

Size of sun disc

**TYPE:**

float in  $[0, 1.5708]$ , default 0.00951204

**texture\_mapping**

Texture coordinate mapping settings

**TYPE:**

`TexMapping`, (readonly, never None)

**turbidity**

Atmospheric turbidity

**TYPE:**

float in  $[1, 10]$ , default 0.0

**classmethod is\_registered\_node\_type()**

True if a registered node type

**RETURNS:**

Result

**RETURN TYPE:**

boolean

**classmethod input\_template(index)**

Input socket template

**PARAMETERS:**

**index** (*int* in  $[0, \text{inf})$ ) – Index

`RNA (url in [0, inf]) - RNA`

**RETURNS:**

result

**RETURN TYPE:**

`NodeInternalSocketTemplate`

**classmethod** `output_template(index)`

Output socket template

**PARAMETERS:**

**index** (*int* in  $[0, inf]$ ) – Index

**RETURNS:**

result

**RETURN TYPE:**

`NodeInternalSocketTemplate`

**classmethod** `bl_ma_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_ma_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`
- `Node.type`
- `Node.location`
- `Node.location_absolute`
- `Node.width`
- `Node.height`
- `Node.dimensions`
- `Node.name`
- `Node.label`
- `Node.inputs`
- `Node.outputs`
- `Node.internal_links`
- `Node.parent`
- `Node.warning_propagation`
- `Node.use custom color`
- `Node.select`
- `Node.show_options`
- `Node.show_preview`
- `Node.hide`
- `Node.mute`
- `Node.show_texture`
- `Node.bl_idname`
- `Node.bl_label`
- `Node.bl_description`
- `Node.bl_icon`
- `Node.bl_static_type`
- `Node.bl_width_default`
- `Node.bl_width_min`
- `Node.bl_width_max`
- `Node.bl height default`

- `Node.color`
- `Node.bl_height_min`
- `Node.color_tag`
- `Node.bl_height_max`

## 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`
- `Node.socket_value_update`
- `Node.is_registered_node_type`
- `Node.poll`
- `Node.poll_instance`
- `Node.update`
- `Node.insert_link`
- `Node.init`
- `Node.copy`
- `Node.free`
- `Node.draw_buttons`
- `Node.draw_buttons_ext`
- `Node.draw_label`
- `Node.debug_zone_body_lazy_function_graph`
- `Node.debug_zone_lazy_function_graph`
- `Node.poll`
- `Node.bl_rna_get_subclass`
- `Node.bl_rna_get_subclass_py`
- `NodeInternal.poll`
- `NodeInternal.poll_instance`
- `NodeInternal.update`
- `NodeInternal.draw_buttons`
- `NodeInternal.draw_buttons_ext`
- `NodeInternal.bl_rna_get_subclass`
- `NodeInternal.bl_rna_get_subclass_py`
- `ShaderNode.poll`
- `ShaderNode.bl_rna_get_subclass`
- `ShaderNode.bl_rna_get_subclass_py`