Geometry Operators

bpy.ops.geometry.**attribute_add(*, name='', domain='POINT', data_type='FLOAT')**

Add attribute to geometry

PARAMETERS:

- **name** (*string, (optional, never None)*) – Name, Name of new attribute
- **domain** (enum in Attribute Domain Items, (optional)) – Domain, Type of element that attribute is stored on
- **data_type** (enum in Attribute Type Items, (optional)) – Data Type, Type of data stored in attribute

bpy.ops.geometry.**attribute_convert(*, mode='GENERIC', domain='POINT', data_type='FLOAT')**

Change how the attribute is stored

PARAMETERS:

- **mode** (*enum in ['GENERIC', 'VERTEX_GROUP'], (optional)*) – Mode
- **domain** (enum in Attribute Domain Items, (optional)) – Domain, Which geometry element to move the attribute to
- **data_type** (enum in Attribute Type Items, (optional)) – Data Type

bpy.ops.geometry.**attribute_remove()**

Remove attribute from geometry

bpy.ops.geometry.**color_attribute_add(*, name='', domain='POINT', data_type='FLOAT_COLOR', color=(0.0, 0.0, 0.0, 1.0))**

Add color attribute to geometry

PARAMETERS:

- **name** (*string, (optional, never None)*) – Name, Name of new color attribute
- **domain** (enum in Color Attribute Domain Items, (optional)) – Domain, Type of element that attribute is stored on
- **data_type** (enum in Color Attribute Type Items, (optional)) – Data Type, Type of data stored in attribute
- **color** (*float array of 4 items in [0, inf], (optional)*) – Color, Default fill color

bpy.ops.geometry.**color_attribute_convert(*, domain='POINT', data_type='FLOAT_COLOR')**

Change how the color attribute is stored

PARAMETERS:

- **domain** (enum in Color Attribute Domain Items, (optional)) – Domain, Type of element that attribute is stored on
- **data_type** (enum in Color Attribute Type Items, (optional)) – Data Type, Type of data stored in attribute

bpy.ops.geometry.**color_attribute_duplicate()**

Duplicate color attribute

bpy.ops.geometry.**color_attribute_remove()**

Remove color attribute from geometry

bpy.ops.geometry.**color_attribute_render_set(*, name='Color')**

Set default color attribute used for rendering

PARAMETERS:

name (*string, (optional, never None)*) – Name, Name of color attribute

bpy.ops.geometry.**execute_node_group(*, asset_library_type='LOCAL', asset_library_identifier='', relative_asset_identifier='', name='', session_uid=0, mouse_position=(0, 0), region_size=(0, 0), cursor_position=(0.0, 0.0, 0.0), cursor_rotation=(0.0, 0.0, 0.0, 0.0), viewport_projection_matrix=(0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0), viewport_view_matrix=(0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0), viewport_is_perspective=False)**

Execute a node group on geometry

Execute a node group on geometry

PARAMETERS:

- **asset_library_type** (enum in Asset Library Type Items, (optional)) – Asset Library Type
- **asset_library_identifier** (*string, (optional, never None)*) – Asset Library Identifier
- **relative_asset_identifier** (*string, (optional, never None)*) – Relative Asset Identifier
- **name** (*string, (optional, never None)*) – Name, Name of the data-block to use by the operator
- **session_uid** (*int in [-inf, inf], (optional)*) – Session UID, Session UID of the data-block to use by the operator
- **mouse_position** (*int array of 2 items in [-inf, inf], (optional)*) – Mouse Position, Mouse coordinates in region space
- **region_size** (*int array of 2 items in [0, inf], (optional)*) – Region Size
- **cursor_position** (*float array of 3 items in [1.17549e-38, inf], (optional)*) – 3D Cursor Position
- **cursor_rotation** (*float array of 4 items in [1.17549e-38, inf], (optional)*) – 3D Cursor Rotation
- **viewport_projection_matrix** (*float array of 16 items in [1.17549e-38, inf], (optional)*) – Viewport Projection Transform
- **viewport_view_matrix** (*float array of 16 items in [1.17549e-38, inf], (optional)*) – Viewport View Transform
- **viewport_is_perspective** (*boolean, (optional)*) – Viewport Is Perspective

bpy.ops.geometry.**geometry_randomization(*, value=False)**

Toggle geometry randomization for debugging purposes

PARAMETERS:

value (*boolean, (optional)*) – Value, Randomize the order of geometry elements (e.g. vertices or edges) after some operations where there a no guarantees about the order. This avoids accidentally depending on something that may change in the future