

## 10.1.12 Editors - Shader Editor - Header - Add Menu - Converter

Detailed table of content.....	1
Add menu - Converter.....	4
Blackbody.....	4
Clamp.....	4
Color Ramp.....	5
Combine HSV.....	7
Combine RGB.....	7
Combine XYZ.....	7
Map Range.....	8
Math.....	9
RGB to BW.....	10
Separate HSV.....	10
Separate RGB.....	10
Separate XYZ.....	11
Shader To RGB.....	11
Vector Math.....	12
Wavelength.....	13

## Detailed table of content

### Detailed table of content

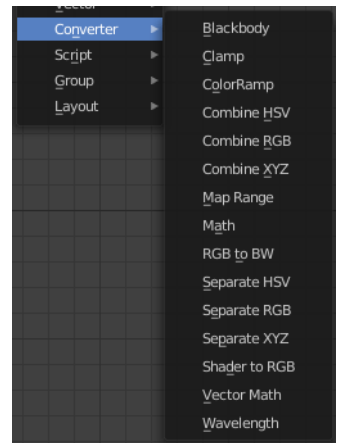
Detailed table of content.....	1
Add menu - Converter.....	4
Blackbody.....	4
Inputs.....	4
Temperature.....	4
Outputs.....	4
Color.....	4
Clamp.....	4
Inputs.....	4
Value.....	4
Min.....	4
Max.....	4
Properties.....	5
Clamp Type.....	5
Min Max.....	5
Range.....	5
Outputs.....	5
Result.....	5
Color Ramp.....	5
Inputs.....	5
Factor.....	5
Properties.....	5
Color Ramp.....	5
Controls.....	5
+.....	5
-.....	5

Tools menu.....	5
Flip Color Ramp.....	5
Distribute Stops from Left.....	5
Distribute Stops Evenly.....	6
Eyedropper (pipette icon) E.....	6
Reset Color Ramp.....	6
Color Mode.....	6
RGB.....	6
HSV/HSL.....	6
Interpolation.....	6
Ease.....	6
Cardinal.....	6
Linear.....	6
B-Spline.....	6
Constant.....	6
Color Ramp.....	6
Active Color Stop elements.....	6
Choose active color stop.....	6
Pos.....	6
Outputs.....	7
Image.....	7
Alpha.....	7
Combine HSV.....	7
Input.....	7
H, S and V.....	7
Output.....	7
Color.....	7
Combine RGB.....	7
Input.....	7
R, G and B.....	7
Output.....	7
Color.....	7
Combine XYZ.....	7
Input.....	8
X Y and Z.....	8
Output.....	8
Color.....	8
Map Range.....	8
Inputs.....	8
Value.....	8
From Min.....	8
From Max.....	8
To Min.....	8
To Max.....	8
Properties.....	8
Interpolation Type.....	8
Linear.....	8
Stepped Linear.....	8
Smooth Step.....	8
Smoother Step.....	9
Clamp.....	9
Outputs.....	9
Result.....	9

Math.....	9
Inputs.....	9
Value.....	9
Value.....	9
Properties.....	9
Operation.....	9
Clamp.....	9
Outputs.....	10
Value.....	10
RGB to BW.....	10
Inputs.....	10
Image.....	10
Outputs.....	10
Value.....	10
Separate HSV.....	10
Input.....	10
Color.....	10
Output.....	10
H , S and V.....	10
Separate RGB.....	10
Input.....	11
Image.....	11
Output.....	11
R, G and B.....	11
Separate XYZ.....	11
Input.....	11
Vector.....	11
Output.....	11
X, Y and Z.....	11
Shader To RGB.....	11
Inputs.....	12
Shader.....	12
Outputs.....	12
Color.....	12
Alpha.....	12
Vector Math.....	12
Inputs.....	12
Vector.....	12
Vector.....	12
Scale.....	12
Properties.....	12
Operation.....	12
Outputs.....	12
Vector.....	13
Value.....	13
Wavelength.....	13
Inputs.....	13
Wavelength.....	13
Outputs.....	13
Color.....	13

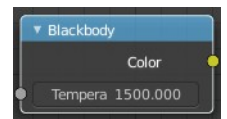
## Add menu - Converter

Here you find mainly nodes to convert data to another data.



### Blackbody

The Blackbody node converts a blackbody temperature to RGB value. This can be useful for materials that emit light at natural occurring frequencies.



### Inputs

#### *Temperature*

The temperature in Kelvin.

### Outputs

#### *Color*

RGB color output.

### Clamp

The Clamp node clamps a value between a minimum and a maximum.

### Inputs

#### *Value*

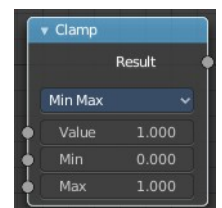
The input value to be clamped.

#### *Min*

The minimum value.

#### *Max*

The maximum value.



## Properties

### *Clamp Type*

#### Min Max

Clamp values using Min and Max values.

#### Range

Clamp values between Min and Max range.



## Outputs

### *Result*

The input value after clamping.

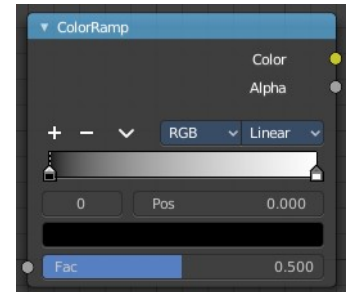
## Color Ramp

The Color Ramp Node is used for mapping values to colors with the use of a gradient.

## Inputs

### *Factor*

The Factor input is used as an index for the color ramp.



## Properties

### *Color Ramp*

Color Ramps enables the user to specify a range of colors based on color stops. The color between the color stops gets interpolated.

## Controls

+

Add a stop to your color ramp. The stop will be added after the selected one, in the middle to the next one.

-

Deletes the selected color stop from the list.

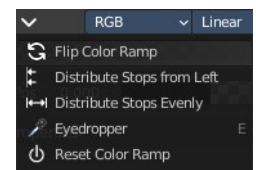
### *Tools menu*

#### Flip Color Ramp

Flips the gradient, inverting the values of the color ramp.

#### Distribute Stops from Left

Rearrange the stops so that every step has the same space to the right.



## Distribute Stops Evenly

Space between all neighboring stops becomes equal.

## Eyedropper (pipette icon) E

An Eyedropper to sample a color or gradient from the interface to be used in the color ramp.

## Reset Color Ramp

Resets the color ramp to its default state.

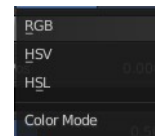
## Color Mode

### RGB

Blends color by mixing each color channel and combining.

### HSV/HSL

Blends colors by first converting to HSV or HSL, mixing, then combining again. This has the advantage of maintaining saturation between different hues, where RGB would de-saturate, this allows for a richer gradient.



## Interpolation

### Ease

Uses an Ease Interpolation for the color stops.

### Cardinal

Uses a Cardinal Interpolation for the color stops.

### Linear

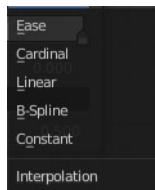
Uses a Linear Interpolation for the color stops.

### B-Spline

Uses a B-Spline Interpolation for the color stops.

### Constant

Uses a Constant Interpolation for the color stops.



## Color Ramp

The color band. A click at one of the color stops makes it the active one. You can move the color stops by clicking at them and dragging them around.



## Active Color Stop elements

Adjust the active color stop.



## Choose active color stop

Choose the color stop by index.

## Pos

The position of the active color stop. The range goes from 0.000 to 1.000

## Outputs

### *Image*

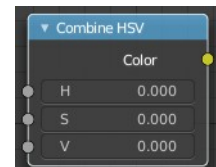
Standard image output.

### *Alpha*

---

## Combine HSV

Combine the single HSV channels into a single image.



## Input

### *H, S and V*

The Hue, Saturation and Value channels of an image.

## Output

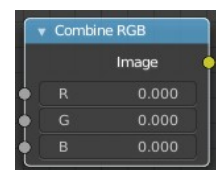
### *Color*

Color output.

---

## Combine RGB

Combine the single RGB channels into a single image.



## Input

### *R, G and B*

The red, green and blue channels of an image.

## Output

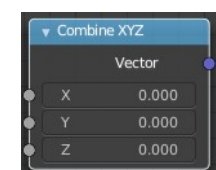
### *Color*

Color output.

---

## Combine XYZ

Same as with Combine RGB node. It combines color values. But instead combining rgb values, which are in the range of 0 to 255, it uses values in the range from 0 to 1.



## Input

### *X Y and Z*

X, Y and Z values.

## Output

### *Color*

Color output.

## Map Range

The Map Range node remaps a value from a range to a target range.

## Inputs

### *Value*

The input value to be remapped.

### *From Min*

The lower bound of the range to remap from.

### *From Max*

The higher bound of the range to remap from.

### *To Min*

The lower bound of the target range.

### *To Max*

The higher bound of the target range.

## Properties

### *Interpolation Type*

#### **Linear**

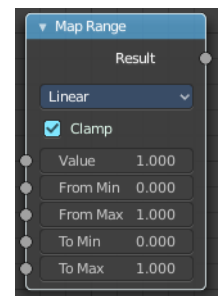
Linear interpolation between From Min and From Max values.

#### **Stepped Linear**

Stepped linear interpolation between From Min and From Max values.

#### **Smooth Step**

Smooth Hermite edge interpolation between From Min and From Max values.





## Smoother Step

Smoother Hermite edge interpolation between From Min and From Max values.

## Clamp

If enabled, the output is clamped to the target range.

## Outputs

### Result

The input value after remapping.

## Math

The Math Node performs math operations.

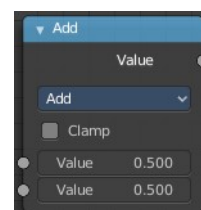
## Inputs

### Value

First numerical value. The trigonometric functions accept values in radians.

### Value

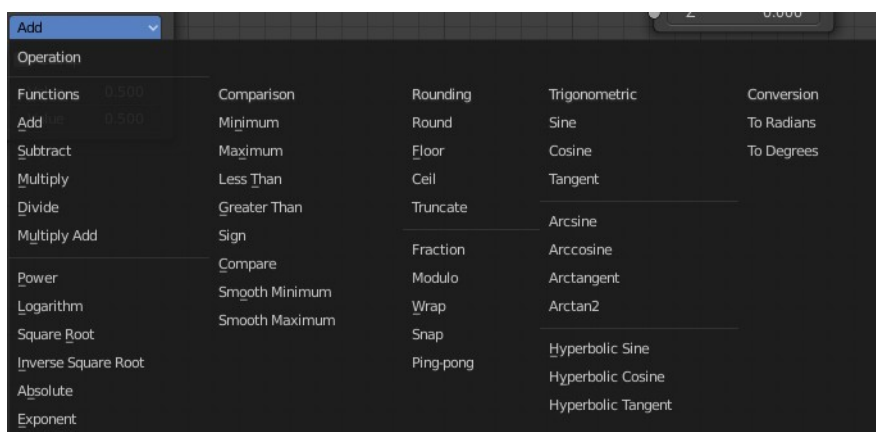
Second numerical value. This value is not used in functions that accept only one parameter like the trigonometric functions, Round and Absolute.



## Properties

### Operation

Here you can choose what mathematical operation to perform.



## Clamp

Limits the output to the range (0 to 1). See clamp.

## Outputs

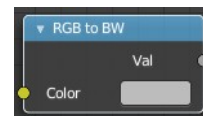
### *Value*

Numerical value output.

---

## RGB to BW

The RGB to BW Node converts an RGB color image to a gray-scale image based at its luminance.



## Inputs

### *Image*

Color image input.

## Outputs

### *Value*

Gray-scale value output.

---

## Separate HSV

Combine the single RGB channels into a single image.



## Input

### *Color*

Color input.

## Output

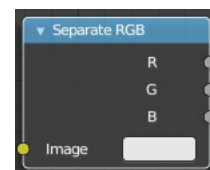
### *H , S and V*

The Hue, Saturation and Value channels of an image.

---

## Separate RGB

Combine the single RGB channels into a single image.



## Input

### *Image*

The image input.

## Output

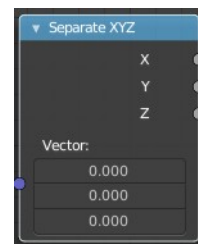
### *R, G and B*

The red, green and blue channels of an image.

---

## Separate XYZ

Same as with Combine RGB node. It separates color values. But instead separating rgb values, which are in the range of 0 to 255, it uses a vector with the values in the range from 0 to 1.



## Input

### *Vector*

The Input vector.

## Output

### *X, Y and Z*

The output vectors for X, Y and Z

---

## Shader To RGB

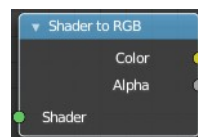
### *Eevee Only*

The Shader to RGB node is typically used for non-photo realistic rendering, to apply additional effects on the output of BSDFs. For example, a color ramp on the output of a diffuse BSDF can be used to create a flexible toon shader.

Using this conversion breaks the PBR pipeline and thus makes the result unpredictable when used in combination with effects such as ambient occlusion, contact shadows, soft shadows and screen space refraction.

Some effects require multiple samples to converge, and applying arbitrary changes to noisy input may not convert to a smooth result.

Warning! If a Shader to RGB node is used, any upstream BSDF will be invisible to the following effects: Screen Space Reflection, Subsurface Scattering



## Inputs

### *Shader*

Any shader such as a BSDF or Emission node can be linked here.

## Outputs

### *Color*

Surface color computed from BSDFs and lighting.

### *Alpha*

Alpha transparency from any Transparent BSDFs in the input.

---

## Vector Math

The Vector Math node performs the selected math operation on the input vectors.

## Inputs

The inputs of the node are dynamic. Some inputs are only available in certain operations. For instance, the Scale input is only available in the Scale operator.

### *Vector*

Input vector A.

### *Vector*

Input vector B.

### *Scale*

Input Scale.

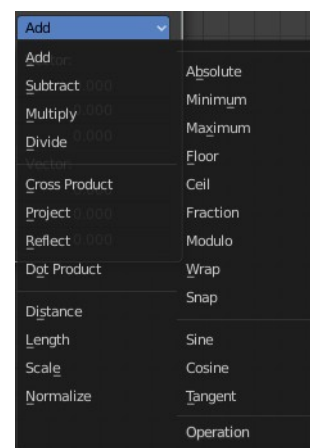
## Properties

### *Operation*

The vector math operator to be applied on the input vectors.

## Outputs

The output of the node is dynamic. It is either a vector or a scalar depending on the



operator. For instance, the Length operator have a scalar output while the Add operator have a vector output.

### ***Vector***

Output vector.

### ***Value***

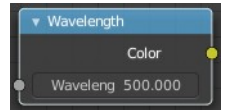
Output value.

---

## **Wavelength**

### **Cycles Only**

The Wavelength node converts a wavelength value to an RGB value. This can be used to achieve a specific color on the light spectrum.



### **Inputs**

#### ***Wavelength***

The color wavelength from 380 to 780 nanometers.

### **Outputs**

#### ***Color***

RGB color output.