Shared function

InterpolateBetween

Interpolates a 3D Vector between a source value and a target value using either linear interpolation or any other easing function. It can also be used to interpolate 2D vectors or scalars by only setting some of the x, y, z values and putting 0 to the others.

Syntax

```
float float float interpolateBetween ( float x1, float y1, float z1, float x2, float y2, float z2, float fProgress, string st
rEasingType, [ float fEasingPeriod, float fEasingAmplitude, float fEasingOvershoot ] )
```

Required Arguments

- **x1, y1, z1:** 3D coordinates of source vector/value
- **x2, y2, z2:** 3D coordinates of target vector/value
- **fProgress:** float between 0 and 1 indicating the interpolation progress (0 at the beginning of the interpolation, 1 at the end). If it is higher than 1, it will start from the beginning.
- **strEasingType:** the easing function to use for the interpolation

Optional Arguments

- **fEasingPeriod:** the period of the easing function (only some easing functions use this parameter)
- **fEasingAmplitude:** the amplitude of the easing function (only some easing functions use this parameter)
- **fEasingOvershoot:** the overshoot of the easing function (only some easing functions use this parameter)

Returns

Returns *x, y, z* the interpolated 3D vector/value if successful, *false* otherwise (error in parameters). As mentioned before, interpolateBetween can be used on 2D vectors or scalars in which case only some (x, y or just x) of the returned values are to be used (cf. alpha interpolation in marker example or size interpolation in window example).