


























VOP

🔍 表格视图

≡ 字段名	□ 标题	📄 I	≡ 概括	≡ 案例日期
	AAA---B 站 落于 ivi 整理		B 站 落于 ivi 整理	2307_00
	Absolute 绝对值	⊕	将负数, 转换为 相对的 正数	2307_24
	Add 相加	+	将输入端各数据 相加	2307_23
	AddConstant 加上常量	+ 1	在节点内 加上指定数值, 无需再创建一个节点输入数值	2307_23
	Align 对齐	📐	将以指定轴向参考 旋转至目标方向 与物体点位置相乘 可驱动	2308_01
	Anti-AliasedNoise 抗锯齿噪波	📶	可输入四维向量 更适合用来驱动动画	2307_27
	Bind/Bind Export 导入导出节点	🗨️	导入 导出属性 同节点不同预设	2307_26
	Ceiling 向上取整	📏	小数点后有数值 直接向上进 1	2307_26
	Clamp 钳制	📏	将输入数据的最小最大值 钳制为指定范围	2307_24
	ColorCorrection 色彩校正	🌈	控制颜色的 色相 饱和度 明度等	2307_27
	Complement 补数	1-X	1 以内通过该节点 获得补数 可以用来反转遮罩	2307_25
	Constant 常量		输出各类类型的数据 如整数 向量 浮点等---中间输入端 直接创建	2307_23
	Cosine 余弦函数	📈	基于输入数值 以余弦曲线变化	2307_24
	Cross 叉乘	📏	输入两个向量 将输出同时垂直二者的 向量	2307_27
	CrossProduct 点乘	📏	输入两个向量 彼此平行为 1, 相对为-1, 垂直为 0	2307_27
	DegreesToRadians 度数转弧度	📐	将度数 转换为弧度 常用与矩阵或函数等运算	2307_28
	DisplaceAlongNormal 法线位移	📐	将输入点 以法线方向位移	2307_25
	Distance 距离	📏	计算两个三维向量 或四维向量 之间的距离	2307_26
	Divide 相除	÷	对输入端数据 相除	2307_23
	DivideConstant 除以常量	÷ 1	在节点内 除以指定数值, 无需再创建一个节点输入数值	2307_23
	EulerToQuaternion 欧拉到四元数	📐	输出四元数 同样可用于四元数旋转	2307_28
	FitRange 范围映射	📏	先将数据钳制在指定范围内 在映射至目标范围	2307_24
	FloatToVector 浮点到向量	📏	将三个浮点数 转换为一个向量 作为位置 方向 颜色等	2307_23
	Floor 向下取整	📏	小数点后有数值 舍弃 向下区	2307_26
	GaussianRandom 高斯随机		基于输入数据 输出正负随机值 范围通过滑块控制	2307_27
	GetVectorComponent 提取向量分量	📏	提取三维向量中的 其中一个分量	2307_24
	HueShift 色相偏移	🌈	读取输入颜色在色环上的位置 滑块可驱动偏移	2307_27
	IfBlock 判定语句	📏	1 端口获取是或否 当成立时 执行下面输入 以及循环内部的内容	2307_31
	ImportPointAttribute 导入点属性		从 指定模型中 获取属性 并指定点编号	2307_26
	InGroup 导入组	📏	输入点或基元 如果在指定组中 则范围 1 可用作判定	2307_31
	IntegerToFloat 整数到浮点	2.0	将整数 转换为浮点 通常用来打破整数运算 避免结果依旧是整数	2307_29
	Invert 逆矩阵	📏	将 3x3 或 4x4 的矩阵逆向化 可将矩阵变换归位	2308_02
	Length 长度	📏	计算三维或四维向量的长度	2307_25
	Logarithm 指数函数	📈	基于输入数值 以指数曲线变化	2307_24
	LookAt 查看	📏	使用输入向量 进行跟踪方向 输出矩阵 与点相乘驱动	2308_01
	MakeTransform 转换矩阵	📏	4x4 矩阵变换的打包节点 无需串联不同矩阵进行控制	2307_31
	Matrix3ToQuaternion 矩阵 3 转四元数	📏	将 3x3 矩阵 转换为四元数	2308_01
	Maximum 最大值	📏	如输入数据与 0 比较 最低结果 不会小于 0	2307_25
	Minimum 最小值	📏	如输入数据与 0 比较 最低结果 不会大于 0	2307_25
	Mix 混合	📏	输入 AB 数据 使用第三个属性 控制混合二者的阈值	2307_25
	Multiply 相乘	×	对输入端数据 相乘	2307_23
	MultiplyAddConstant 乘后再加	++	可通过一个节点实现乘加乘 + 可用来控制偏移*用来控制强度	2307_24
	MultiplyConstant 乘以常量	× 1	在节点内 乘以指定数值, 无需再创建一个节点输入数值	2307_23
	Negate 取反	⊖	将输入的数值 反转 负变正 正变负	2307_23
	Normalize 规格化	0 1	将向量的长度 统一为 1 方向不变	2308_02
	Orient 定向	📐	可用来排除移动 至保留矩阵的旋转信息	2308_02
	Parameter 参数	📏	将当前输入数据 调用至外部控制 控制中键快速创建	2307_25
	Power 次方	X ^y	小于 1 将逐渐衰减 反则递增, 可用于提高对比度	2307_24
	Quaternion Distance 四元数距离		测量两个四元数距离 可作为阈值 使其二者最近的区域 指定效果	2307_29
	Quaternion 四元数	📏	可作为基础四维矩阵---若作为旋转 向量作为朝向 分量作为旋转角度	2307_28

	QuaternionInvert 四元数反转		反转当前的四元数 同获得相反的旋转	2307_29
	QuaternionMultiply 四元数相乘		使用四元数旋转后 配合该节点 以旋转后的矩阵坐标 二次旋转	2307_28
	QuaternionToMatrix3 四元数转矩阵 3		将四元数 转换为 3x3 矩阵	2308_02
	RadiansToDegrees 弧度转角度		将弧度 转换为度数 常用于弧度的运算 转换为角度 方便运算	2307_28
	RampParameter 渐变映射		在上级 将数据以坡度 或颜色滑块 进行映射	2307_24
	Random 随机		输出随机属性 可设置常数 或三维数据---直接输出四元数 作为旋转	2307_26
	RelativeToBoundingBox 相对边界		获得相对于几何体边框的相对位置 取分量 可看到渐变	2307_25
	Rotate 旋转矩阵		指定旋转轴向 另一个分量为弧度旋转角度 与单位相乘驱动	2307_30
	RotateByQuaternion 四元数旋转		输入用来旋转的四元数 以及需要旋转的点 进行驱动	2308_01
	RoundToInteger 四舍五入			2307_22
	Scale 缩放矩阵		可以使用三个轴 控制缩放矩阵 与单位相乘驱动	2307_30
	SetVectorComponent 设置向量分量		为向量中的其中一个分量 赋值	2307_23
	Sign 正负性		输入小于 0 返回-1, 大于等于 0 返回 1, 只有-1 1	2307_27
	Sine 正弦函数		基于输入数值 逐渐 0-1-0- -1 循环	2307_24
	SphericalLinearInterp 四元数插值		将在两个四元数驱动的方向中, 进行混合	2307_29
	Subtract 减去		第一个输入 减去其他输入の数値	2307_23
	Subtract 减去常量		在节点内 减去指定数值, 无需再创建一个节点输入数值	2307_23
	Switch 切换节点		根据整数 切换不同的 对应输入端	2307_23
	Tangent 切线函数		基于输入数值 以切线曲线变化	2307_24
	TransformMatrix 变换矩阵		通过移动 旋转 缩放 来变换输入的位置 法线 向量	2307_27
	Translate 移动矩阵		通过控制三个轴 来移动一个矩阵 与点位置相乘 驱动	2307_30
	TrigonometricFunctions 三角函数		包含各种三角函数预设 并可直接控制强度 偏移等	2307_24
	TurbulentNoise 湍流噪波		可计算输出常量 1 或向量 3 的噪波---OPN 模式为正负噪波	2307_23
	VectorToFloat 向量到浮点		将一个三维向量 解压为三个分量	2307_23
	VectorToMatrix3 向量到矩阵 3		将三个向量 合并为 3x3 的矩阵 一般作为旋转矩阵使用	2308_02
	VectorToQuaternion 向量到四元数		输出围绕该轴的 四元数	2307_28
	VectorToVector4 向量到四维向量		可用于代表朝向的向量 而剩下的分量 用来以该轴 进行旋转	2307_27