Expression をやってみよう vol.2「+-×÷」

エクスプレッション難易度 レベル 1: ★☆☆☆☆

はじめに

押忍!セガ 廣田です!

前回はイコールのエクスプレッションをやりました。「数値を同じにする」というやつですね。 イコールだけでも色々な事ができましたが、今回は数値を足し引きして、もっとたくさんの事をできるよう にしましょう。計算が苦手なアーティストの方のために、細かく説明しますので安心して下さい。

自動ドアを作る ~ 反対の動きをさせる ~



まずは簡単な所から。自動ドアのような左右開きの扉を作ってみましょう。

両方の扉に別々にアニメーションを付けても良いのですが、開くタイミングなどを調整し始めると手間が かかって大変です。

こういう時は、片側の動きを真似してくれると楽ですね。

では、片方の扉の真似をするために、前回のイコールでつなぐ所から始めましょう。 覚えてますか?

『わすれた!**』**

では軽く前回の復習。

2つの扉のプロパティエディタを表示して、パラメーターをドラッグ&ドロップです。

サンプルデータ: Sample1_Base.scn

手順:

まず扉の移動軸を確認しましょう(R 扉を選択して移動させてみて下さい)。X 軸で移動する扉ですね。 LR の扉のキネマティクスプロパティを表示します。(扉を選んで Ctrl+K。ロックを忘れずに) R 扉の位置 X パラメーターを、L 扉の位置 X にドラッグ&ドロップします。

	R		
Door : L : Local Transform		Door : R : Local Transform	
◆ Local transform SRT スケール オプション 位置制限 回転制限		SRT スケール オプション 位置制限 回転制限	
ピボット ニュートラルボーズ		ピボット ニュートラルボーズ	
D Y		O Y	NO NO
0 Z		0 z (1	
			1
			2
□ オ-ダ (XYZ 🕥		□ オ-ダ (XYZ •••	1
位置			/
			-
	~		-

どうでしょう?Rの扉を移動させてみてください。

『これはちょっと・・・。一緒についてきちゃって開かない・・・。』

値を同じにしたのだから、こうなるは当然と言えば当然です。

では、エクスプレッションエディタを使って、きちんと動くようにして行きましょう。

エクスプレッションエディタの使い方

まずは、エクスプレッションを書くための、エディタの見方から。

(エクスプレッションエディタを閉じてしまった場合は、パラメーターの「=」アイコンをクリックしてください。 再度エクスプレッションエディタが開きます)

Expression Editor	_ _ X				
ファイル、編集、ビュー、関数、オブジ	ェクト」パラメータ 削除 確認 道用				
Door.L.kine.local.posx	前次 現在の値: 0.000 ?				
エクスプレションを編集中: Door.L.kine.local.posx 有効なエクスプレッション: Door.R.kine.local.posx メッセージエリア					
Door.R.kine.local.posx	ここにエクスプレッションを書く				
・白いエリアにエクスプレッションを書きます。 ・灰色の所は Softimage からのメッセージが表示される所です。 ・右上に「適用」ボタンがあります。エクスプレッションを書き終わったら、このボタンを押すと					
Softimage にエクスプレッションの決定を伝えることができます。					

エクスプレッションの見方

次は、エクスプレッションの見方です。今はこんな風に書かれています。

Door.L.kine.global.rotx

どうでしょう? ながめてみると、見たことあるキーワードが並んでいるような気がします。

『なんとなく分かるような、分からないような・・・』

ドットの部分で区切ってみると、オブジェクトの名前とパラメーターが並んでいます。 エクスプローラーと見比べてみましょう。



エクスプローラーの階層を下がる毎に、ドットで区切られていますね。

エクスプレッションは文字で書かないといけないので、パラメーター階層をなんとか文字で表現しないといけません。そこで「階層1つ下がるごとに、ドットで区切って書こう」というルールになっています。

慣れるまではエクスプローラーのイメージと合わせるのは大変ですが、すぐに慣れると思います。

そして、階層の書き方以外に、まだルールがあります。

なんと1行で書かなければいけません!見やすいからと言って、途中で改行はできません。 日本語も禁止です。半角のみ。

エクスプレッションは Softimage の前身の、Softimage3D 時代からある古い機能なので、色々 と制限があります。今は、より高機能なスクリプトオペレーターや、ノードを組み替えるだけで 機能を作れる先進的な ICE と言った、素晴らしい仕組みが用意されています。

『まだ覚えることがあるのか・・・。』

いきなり難しいことを一気に覚える必要はありません、少しずつ進んで行けば大丈夫です。 しかし、エクスプレッションはたった一行でも色々な事が手軽にできる強力さを持っています。 気を取り直して、式を作っていきましょう。

反対方向ということで、マイナスを掛けます。 掛けるは「*」記号を使います。(×と書いてもダメですよ。)

『も一、色々ルールありすぎ!』

では、ついに式を書く時がやってまいりました! エクスプレッションエディタの白いエリアに、下のように入力してみましょう。

Door.L.kine.local.posx * -1

「*」や「-1」の前後のスペースは入れても入れなくても構いません。無ければ Softimage が 勝手に入れてくれます。

書き終わったら「適用」ボタンを押します。

エクスプレッションを決定する時は、必ず「適用」ボタンを押さなければいけません。書いただけでは Softimage には伝わりません。

メッセージエリアに「有効なエクスプレッション」と表示されたら成功です。

どうでしょう?R扉を動かすと、L扉が反対方向に動くようになりましたか?

『動いたけど、何でマイナスを掛けるの?』

さて、なぜマイナスを掛けると逆に動くかですが、ちょっと数の仕組みを考えてみましょう。

ひとまず数直線上で数の動きを見てみます。 0が基準です。とりあえず右方向に進むことにしましょう。

3を考えます。3は目盛り3個分です。0の位置から行くと、3の位置に来ました。

引き算を考えてみます。

3-2 はどうでしょう?

「マイナスは逆方向に進む」と考えると良いです。

0から右方向に3進んで、逆方向に2行きます。答えは1ですね。



では、3-4 はどうでしょう? 0 から右方向に3 進んで、4 戻ります。 なんと、0 から左にはみ出してしまいました! この、はみ出して到着した場所を、0 から逆に1 つ行った場所ですから、マイナス1と名前を 付けましょう。



次は、「掛け算」を考えてみます。

3×1はどうでしょう?

目盛り3個分を、1つと読みます。3ですね。これは最初と同じ場所です。



では、3×(-1)はどうでしょう? これは目盛り3個分を、0から逆方向に1つと読みます。 答えは -3 になりました。 なので、数値にマイナスを掛けると逆方向に動く事になります。 同じように考えて、3×(-2)は目盛り3個分を逆方向に2つで-6となります。



ここからは発展です。「マイナス×マイナス」を考えてみます。 (-3)×(-1)はいくつでしょう?

『むむー!?』

ついにやってきてしまいました。「マイナス×マイナス」。

マイナス×マイナスはプラスとテスト前に暗記して、そのまま数学を嫌いになってしまった方はいませんか?

ここで、もう一度やってみましょう!

ちょっとずつ整理しながらやれば大丈夫です。「-3」と「*-1」に分けて考えましょう。 -3 は一つ前でやりました。3×(-1)の結果です。 なので、-3 を 3×(-1) に戻して置き換えてみると・・・・

$$(-3) \times (-1)$$

 $3 \times (-1) \times (-1)$

3×(-1)×(-1)と書けます。

さっき見た形に近くなってきました。

同じように読むと、「目盛り3個分を0から逆方向に1個を、逆方向に1個」と読めるようになり ました。

では、いくつでしょう?

こたえは+3 です。 逆の逆は元の方向です。なので+3。裏の裏は表みたいなものです。



どうでしょう?

マイナス×マイナスはプラスになりました。

計算した結果たどりついた目盛り上のマイナスという名前の位置と、逆に行くというのを一緒 に考えてしまうと、よく分からなくなります。

「マイナスは単に逆方向へ行く」という意味で、気楽に考えて構いません。

『か・・・、簡単すぎるぜ!』

時計を作ってみよう!



もうちょっと手のこんだことをやってみましょう。 今度は時計を作ってみます。 長針が1周すると1時間というアレです。(知ってますよね;) 長針をくるくる回すと、短針がいい具合の時間を示すようにしてみましょう。

時計の形を作る(短針と長針の長さを決める)

まずは、時計自体を作りましょう。短針と長針を準備します。 短針と長針が同じ長さだと区別がつかないので、エクスプレッションで長針を短針の 1.5 倍の長さにし てみます。

サンプルデータ: Sample2_Base.scn

手順 :

・長針を長くするにしても、どっち方向に大きくするか分からないので、最初に軸を確認しましょう。

・長針(LongHand)を選んだ時に出てくるスケール変換マニピュレータ(xキー)で確認します。向きはY 方向です。

・長針(LongHand)のYスケールを短針(ShortHand)と同じ値にして(もうできますよね?)、エクスプレッションを書いていきます。

Clock : ShortHand : Local Transform 💿 🕤 📘 🗙	Clock : LongHand : Local Transform 🛛 😨 🕤 💶 🗶			
Local Transform Expression	Local Transform Expression			
▼ Local Transform ?	▼ Local Transform ?			
SRT スケール オプション 位置制限 回転制限	SRT スケール オブション 位置制限 回転制限			
ピボット ニュートラルボーズ	ピボット ニュートラルボーズ			
_{こスケール} ShortHand(短針)	LongHand(長針)			
	0 X (1			
	• F Y (1.5			
<u>Clock.ShortHand.kine.local.scly</u> * 1.5				
│ 式の解説 :				
Clock.ShortHand.kine.local.scly * 1.5				
-14				
短計の人ケール値	× 1.5倍			
最後に適用ボタンを押すのを忘れずに!				

短針のY軸スケールに合わせて、長針の長さが変わるようになりました。 好きな長さにしてみてください。

時間の仕組みを作る(長針に合わせて、短針が動くようにチャレンジ!)

さて、これで時計の「形」としての準備はできました。

今度は時計の「仕組み」を作って行きましょう。

長針はそのまま回せば良いとして、問題は「長針が回った時、短針をどのくらい回すか」です。 時間の文字間の角度を測っても良いのですが、もうちょっとスマートにやりたいですね。 まず、文字盤をよく見てみましょう。丸いですね。

『そりゃそうだ。』

そう言わずに。大切なことなので。

丸い=円です。円の1周は 360°です。これは OK ですね。 もっと文字盤をよく見ましょう。1 から 12 までの表示がありますね。

つまり、360° が 12 分割されているということです。

これを何とかすれば、1時間分の角度が分かりそうです。 では、360°を12分割してみます。

 $360^{\circ} \div 12 = 30^{\circ}$

ということで、時間の数字は 30° ずつ並んでいるようです。 つまり、1時間経つと短針は 30° ずつ進むという事です。

『小学校でやったような・・。』

次に、長針が何時間進んでいるかを求めてみます。

「1時間経った」とは、どういうことでしょう?長針が1周回った=360°回ったと言うことですね。 つまり長針の今の回転量を360で割れば、何回まわっているか(=何時間経ったか)が分かりますね。 では、短針(ShortHand)に式を入れてみましょう。





長針を回すと、合わせて短針が動くようになったと思います。 どうでしょう。計算ぽくなってきましたね。

『なんかカッコイイ・・。』

完成シーン(Sample2_Finish.scn)では Null にエクスプレッションが入っていますので、長針・短針の ジオメトリを好きなデザインのものに差し替えることができます。

こう言った仕組みをひとつ作っておけば、時計を作る時は再利用すれば良いので安心ですね。

エクスプレッションを組む理由は、このような使いまわせる仕組みを作るという部分にもあります。

歯車 ~時計の応用で~



今度は歯車です。難しそうですね。でも、時計ができれば、同じ式で歯車も動かせます。 これも1つずつ考えてみましょう。

歯車は、1 つの歯が動くと、他の歯車も必ず一山分だけ動きます。 一山動いたら、隣の歯車が二山動いてしまうような事はありません。 つまり、隣が一山回ったら、自分はどのくらい回ればよいか?を考えます。

まず、時計と同じように、円が何分割されているかを考えましょう。

数えやすい所で、歯の山の数を数えてみます。

時計は親切に何分割か数字が書かれていましたが、これは自分で数えるしかありません。 サンプルでは、8枚と16枚なので、

8 枚歯があるギアは、一山動くと、360°÷ 8 枚で、45°

16 枚歯があるギアは、一山動くと、360° ÷ 16 枚で、22.5°

となります。

つまり、隣りの歯が一つ進むと、自分は「360÷自分の歯の数」だけ回ります。

では、早速エクスプレッションを設定していきましょう。



Expression をやってみよう 第2回「+-×÷」



どうでしょうか?時計の式とほとんど同じですね。

次回

いかがでしたか。つないだだけのパラメーター値に+-×÷することで、格段に出来ることが広がりましたね。

次回は、円周率をやってみましょう。