

Lesson 2 プログラムの構成要素

こんにちは！Lesson2では基本的なプログラムの構成要素を使ってみます。
これまでよりかなりプログラムっぽくなると思います。

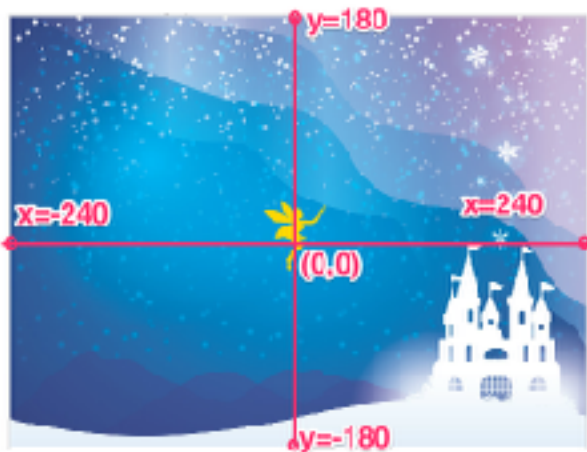
サンプルとして、ちょっとロマンチックな
ものを用意しました。

「氷を降らす妖精」
妖精を動かすと、妖精さんからキラキラ
と氷が舞い降ります。



ステージの座標

ステージにスプライトを配置するにはステージの座標を知る必要があります。
横の位置を示すのがx座標、縦の位置を示すのがy座標です。



中心が(0,0) で、
x座標は-240（左端）から240（右端）
y座標は-180（上端）から180（下端）
です。

妖精さんの位置は
スクリプトエリアの右上端、

マウスの位置は
ステージの右下端に
あります。今は妖精さんは
(4,4)にいますね。



妖精さんを別の場所に移動させるには、「動き」のブロックを使います。



演習：妖精さんを (-55, 20) に動かしてみましょう。

スプライトの向きと動き

今度は妖精さんを動かしてみましょう。
スプライトを動かすには2種類の方法があります。

(1) 向きを決めて動かす

コスチュームでスプライトの絵を描いた時点で、スプライトは右を向いているという設定になっています。（左を向いている絵でもプログラムでは右を向いている設定になる。）

この状態で **10 歩動かす** を実行すると、右向きの状態で10歩分移動します。これが「向きを決めて動かす」方法です。

なお、10歩動かすといってもアニメーション風に10歩動かすのではなく、10歩分の大きさで移動するので、パッと位置が変わるイメージです。10歩歩かせたい時の話はこの次の「繰り返し」にてやりましょう。

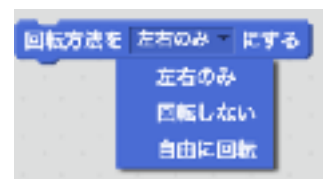
向きを変えるには **90度 度に向ける** または **90度 度に向ける** を使うと向きが変わります。

向きの変え方は右の図を参考にしてください。最初に右向きということは90度を向いている状態です。たとえばここで「0度に向ける」を実行してみると妖精さんが上を向いた状態になります。



演習：妖精さんを左向きにして移動させてみてください。

なお、今回の場合あんまり妖精さんが上や下を向くのはかっこよくないので、左右しか向かないように設定したいとします。その場合は一度「回転方法を「左右のみ」にする」を実行すると、90度か-90度しか向けなくなります。



(2) x座標またはy座標（または両方）の移動を指示する

向きに関係なく動かしたい時は、**x座標を 10 ずつ変える**（もしくはy座標を(10)ずつ変える）を実行するとx座標またはy座標の移動距離を指定して動かすことができます。

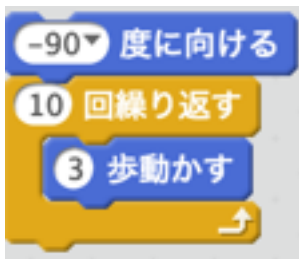
繰り返し

さて、妖精さんを動かせたとはいえアニメーションのように徐々に動かせていません。移動を何度もくりかえすと滑らかに動いているように見えます。

繰り返しは「制御」のブロックです。左右反対のコの字担っているブロックのコの中に命令を入れるとそれが何度も繰り返されるようになります。

繰り返しの条件はいくつかあって、

- ・ 回数指定（～回繰り返す）
 - ・ 指定しない（ずっと）
 - ・ 条件を満たさなくなるまで
- があります。



例として左のように「左に向けて3x10歩歩く」とやると、それなりにスムーズに左に進むようになります。

条件によって動きを変える

制御のブロックには繰り返しの他に右のブロックもあって、これは「もし」と「なら」の間にコの字の内側を実行する条件を書きます。



たとえば、妖精さんが左に移動して、端まで行ってしまったら反対側から現れる、ということを書きたい場合、下のように書きます。



今回は繰り返しを「ずっと」にしています。


わかりますでしょうか？条件は「x座標<-240」です。=でない理由は、10歩ずつ動かすので必ずしも-240ぴったりにはならないからです。


演習：妖精さんをいろいろ動かしてみましょう



参考：ハートのトレースは、ハートのスプライトのスクリプトで以下のように「消す」以降のブロックを「旗をクリックした時」にくっつけるとできます。

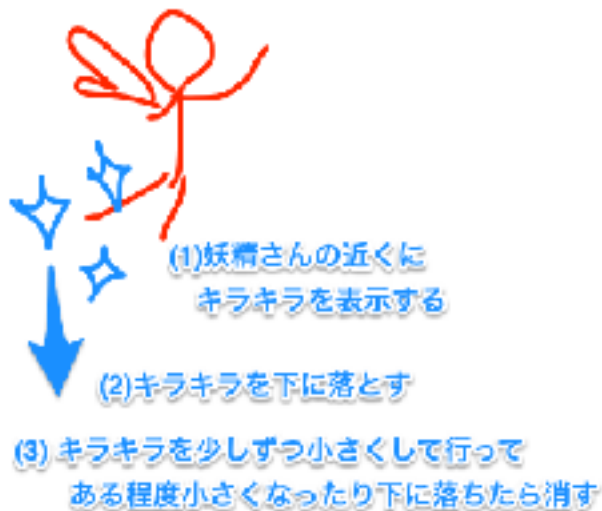
また、向きや位置を指定するのに、マウスや他のスプライトの位置を使うこともできます。

例えば  を使うと、マウスポインターのある方向へスプライトを向けさせることができます。

また、  を使うと、マウスポインターのある場所へスプライトを移動させることができます。

スプライトを複製する

今度は妖精さんの通った場所に、キラキラを降らせて見ましょう。



方針としてはこんな感じです。キラキラをたくさん表示するには、複製をする必要があります。複製の方法は2つあります。

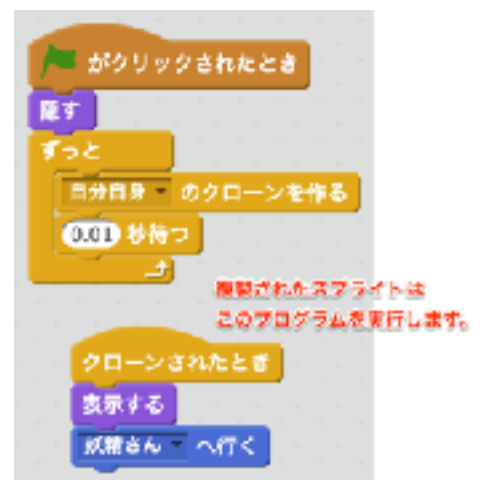
(1) スタンプを使う

- ・複製したあと位置や見た目などを変更する必要がない場合は、スタンプを使います。

(2) クローンを使う

- ・複製したあと位置や見た目などを変更する必要がある場合は、クローンを使います。

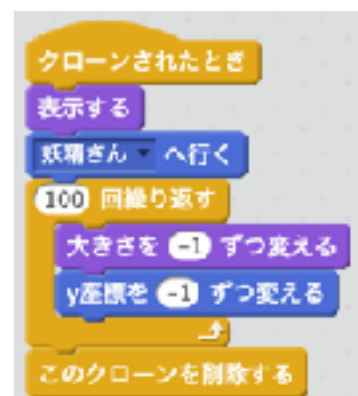
ここでは「キラキラを少しずつ小さくして下に落とす」という変更があるのでクローンを使います。まずは大きさや位置を変えずに、妖精さんの通った後にキラキラのクローンを残していくプログラムをみましょう。



まずは上のブロックから。ずっと繰り返して、キラキラのクローンを作っています。そして、クローンされたスプライトは下のブロックを実行します。ここでは妖精さんのいる位置に表示しています。

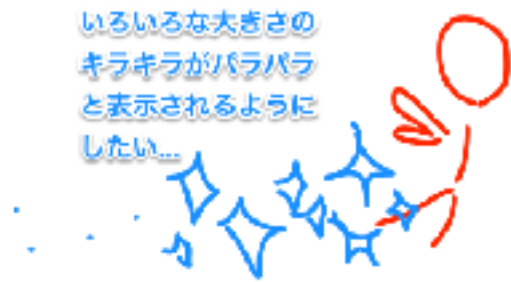
「キラキラを少しずつ小さくして下に落とす」はこんな風に描いて見ました。

ここで大事なのが、最後に「このクローンを削除する」をつけているところ。クローンをたくさん動かすのはプログラムの大変です。動作が重くなるのとクローンできる数に限りがあるので必要がなくなったらすぐに削除するようにするのが良いでしょう。



演習 実はこのままだとちょっとキラキラがイけて
いません。キラキラの大きさや位置をちょっとぶら
しながら表示すると、ちょっとだけ素敵になります。

下の2種類の演算ブロックをうまく使って、工夫を
してみましょう。



Lesson2まとめ

- ❖ 位置や向きを指定してスプライトを置いたり移動することができるようになりました
- ❖ 繰り返しや条件分岐が使えるようになりました
 - ・ 繰り返しや条件分岐はプログラミングを使いこなす上でとても重要なポイントです
- ❖ クローンを使った複製ができるようになりました

Lesson1と比べてかなりプログラムらしくなってきましたね。作成できる作品の幅もかなり広がったのではないかと思います。

そろそろ「こんなものを作って見たいなあ。どうやったら作れるだろう」ということを意識しながら取り組んでみると良いと思います。

それでは、Let's enjoy!