

情報デザイン応用演習I 4.Canvas応用

目次

1. 初めに

i. CanvasFirstStep

2. Canvas応用

i. 今日の狙い

ii. 物理的な動き

iii. オブジェクト指向

3. 演習

i. では今日も演習に取り組みましょう。

初めに

「ECMAScript入門」「Canvas入門」やっていない人は、今日の内容全くわかりません。
必ずJavascriptFirstStep,CanvasFirstStepやってからこちらに取り掛かりましょう。

CanvasFirstStep

軽くおさらいしてみましょう。

一定時間ごとに適当に (ランダム) に描画させることにより、アニメーションを作成するという方法を取りました。

Canvas応用

今日の狙い

前に描画した情報を元に、次の情報を更新しながら書き換え、「動き」のあるアニメーションを作成します。

物理的な動き

`draw()`が呼び出されたタイミングで描画されるのは前と変わりませんが、ひとつ前で、どこに描画したか、を変数で覚えておいて、次の位置を計算しながら描画するとアニメーションとなります。

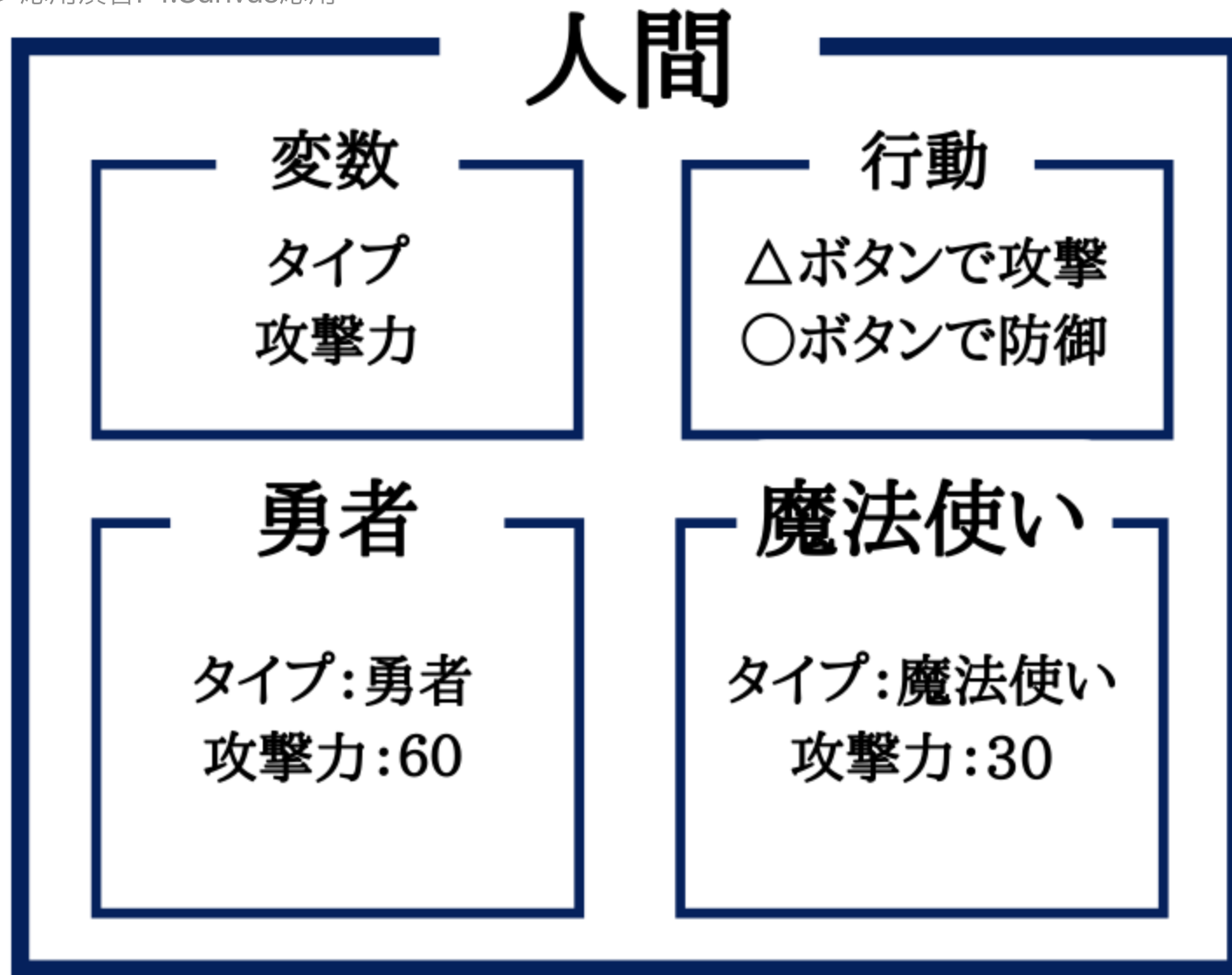
オブジェクト指向

円を一つだけ描画するなら、前回と同じような書き方で良いのですが、ややこしくなってくるので、オブジェクト指向という考え方を導入します。

これは、現在多くのプログラミング言語で採用されている考え方です。

この2,30年**オブジェクト指向**の考え方が良いとされてきましたが、最近では**関数型**の方が良いという動きもあります。

どちらを選ぶ？ 関数型プログラミングとオブジェクト指向プログラミング(OOP)



クラス・インスタンス・プロパティ・メソッド

- クラス：同じ特徴や性質を持つモノをまとめたオブジェクト。抽象化された概念
- インスタンス：クラスによって、作られたオブジェクト。実体化されたもの
- プロパティ：オブジェクトが持っている特徴
- メソッド：オブジェクトが起こすアクションのこと

今日やることは

- クラス：パーティクル(粒子)
- インスタンス：実体化されたパーティクル(粒子)
- プロパティ：色・速度・位置
- メソッド：draw(描く), update(位置情報を更新)

粒子でできちゃえば、線でも三角形でも同じようにプログラムすることができます。

パーティクルプラグイン

AfterEffectsやBlender等でパーティクルと呼ばれるプラグインがあります。ちょっとみてみましょう。

これを、プログラミングでどのように記述するか、が今日の内容となります。

- **【プラグイン無し】 簡単！パーティクルで紙吹雪を作ろう！【AfterEffects チュートリアル】**

演習

では今日も演習に取り組みましょう。

今日の教材は

- Canvas SecondStep

です。

(時間なかったら、ちょっと古い書き方のJavaScriptのままかも...)

ひとつずつ、理解しながら進めましょう。