Cocos2d-xで作る物理演算ゲーム Physics Editor編2

= 2016年10月22日 =

ここまでのソースプログラムはここ

http://monolizm.com/sab/src/AngryChicken25.zip

GETだぜ!

今回は 外部ツール(PhysicsEditor)で 対象物を複雑なものに します。



個別の剛体形状は physics editorでできるようになった。



しかし対象物を配置する方法がない。このまま ではすべてプログラムで地味に配置するという ことになる。

そんなことプログラマはやりたくない。 時間の無駄。 なので何かツールを使おう。。。

というわけで、cocos studuio を使ってしまおう!



Obiects		o ×	startNode.cs ×							1				Ý
▼ Basic Ob	jects		8											
	† † †	•	5											
TileMap	Particle	Audio												
$\stackrel{\uparrow}{\leftarrow} \stackrel{\uparrow}{=} \rightarrow$			100											
Node	Sprite		50											
∀ Widgets														
Button	$\mathbf{\bigvee}$													
Button	CheckBox	Image	8											
Resources														
	e2		8-											
	inode.csd		-150 -2(
			8 -		200	150 100	,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,	יידידין		150	200		200	11111
			Animation 🗆 🛪	Output	-200	-150 -100	-50		50 10	N 150	200	250	300	330 ⁻

cocos2d-xのためのGUI開発ツールです。画像素材のドラックアンドドロップによるSpriteを配置や、タイムラインでのコマアニメーション作成等、非常にスピーディーな開発を行うことが出来ます。

これで構想は整ったので やってみよう!

やることは大きく4つ。

Physics editor2で対象物の剛体作成し出力
 cocos studio2で対象物の配置し出力
 上記2つの出力データをxcodeに登録
 ソースコード修正

1. Physics editor 2 で対象物の 岡体作成

*ライセンスの関係があるので、 すべてのパソコンい入れるの無理 なので、こちらでやっておきました。 (ちなみに1年1ライセンス(PC2台まで)で2199円也



← これがその出力ファイル。
これを以下のフォルダにコピー

/Resource/pe/

cocos studio2で対象物の配置



cocostudio 2 が起動したら以下の設定。 iPhone5sにしとこう。そして横向きをクリック



リソースを登録

• • •	📄 アング	リーチキン	
$\langle \rangle$			Q. 検索
よく使う項目	名前	> 変更日	サイズ
🗏 マイファイル	AngryChicken	今日 14:59	
	▼ 📄 リソース	今日 15:16	
	- block1.png	今日 11:10	10 KB
(@) AirDrop	block2.png	今日 11:11	31 KB
·A. アプリケーション	chicken.plist	今日 11:22	8 KB
デスクトップ	\star star.png	今日 11:05	10 КВ
四 書類			
🔮 ダウンロード			
デバイス			
BOOTCAMP			
 ③ リモートディスク			
タグ			
● レッド			
😑 オレンジ			

上記ファイル3つを、cocos studio 2の Resourceペインにドラッグアンドドロップ



各画像を選択してドラッグアンドロップして、右のように配置しよう!(とりあえず同じものにしてね)



下は地面があるので40ピクセルぐらいうかせる

重要!名前を変更する。

プログラムから配置したオブジェクトを読み込 むために、名前を変えておく。 配置した画像をクリックすると、右ペインに プロパティがでるので最上部の名前を以下の ようにする。



最終的にはこんな感じになればOK







3. 上記2つのxcodeに出力データを登録



*フォルダ設定を「参照」にしてあるの で、自動的に以下のモノが見えている はず。

4. ソースコード修正

_createBlocksメソッドを以下のように変更。*コメント化してあるので、 ごっそり差し替えてください。

void MainGame::_createBlocks()

Node* node = CSLoader::getInstance()->createNode("MainScene.csb"); this->addChild(node);

auto starPb = PEShapeCache::getInstance()->getPhysicsBodyByName("star"); starPb->setMass(200.0); // 重さを指定(ここが無いと後で飛ばせなくなる) starPb->setContactTestBitmask(true); Sprite* star = (Sprite*)node->getChildByName("star1"); star->setPhysicsBody(starPb);

```
~~~~~省略~~~~~~
~~~~~省略~~~~~~
```

*/

auto blockPb4 = PEShapeCache::getInstance()->getPhysicsBodyByName("block2"); blockPb4->setMass(200.0); // 重さを指定(こうか悪いと後で飛ばせなくなる) blockPb4->setContactTestBitmask(true); auto block4 = (Sprite*)node->getChildByName("block2_3"); block4->setPhysicsBody(blockPb4);

> – もしかしたら5か所すべてで、ここが「_」(アンダーバーでなく、「-」 (ハイフン)になってるかも。その場合はアンダーバーになおしてくだ さい。具体的にはcocos studioで設定した名前がきます。



// cocos studioで出力したデータを取得し、描画ツリーに登録している。 Node* node = CSLoader::getInstance()->createNode("MainScene.csb"); this->addChild(node); // 描画ツリーに登録

// ファイルを読み込みルートノード取得

まずは作ったデータを描画ツリーに登録。これでcocosstudioで作った画像は表示はされる。

//peで作成した剛体情報と、cocos studio2で配置した画像を紐づける
//PEShapeCasheにはchicken.plistを登録済み(前回を参照)
auto starPb = PEShapeCache::getInstance()->getPhysicsBodyByName("star"); // star剛体を取得
starPb->setMass(200.0); // 重さを指定(ここが無いと後で飛ばせなくなる)
starPb->setContactTestBitmask(true);
Sprite* star = (Sprite*)node->getChildByName("star1"); //描画ツリーから、紐づけるspriteを取得(cocos stuido2で配置した画像を取得)
star->setPhysicsBody(starPb); // 剛体とspriteを紐づけ

剛体の画像の紐づけ 表示された画像を物理空間の影響を与えるために、PEで作った剛体と紐づける。 基本は上記処理を配置されたオブジェクトすべてに対して行う。

実行してみよう!





いろいろ変えてみよう!

cocos studioに戻って、配置を変えてみよう!

また、新しいオブジェクトを追加するのも簡単。 ただその場合、プログラムで剛体の紐づけ部 分の追加修正が必要になります。

まとめ

Physics editorで個々の剛体を作成し、配置は cocos studio2を使うことによってステージ作成 が簡単にできるようになる!

超悲報



cocosがまた開発IDEを変えま した。新しい環境はその名も cocos createor。unityライクに なり、言語もluaらしいです。こ の開発者殺し♪

とりあえずこのプレゼンはC++ の勉強、2Dの概念的な部分 は学べますし当面は今の環境 のままでいきます。ただ・・・状 況をみて、もしかしたら途中で 移行するかもです。

次回は物理演算Chipmunk Physics editor編「敵」を出す



ここまでのソースプログラムはここ

http://monolizm.com/sab/src/AngryChicken26.zip

ご清聴ありがとうございました。