

Demo 解説

白山雄翔 (WHITE WATER)

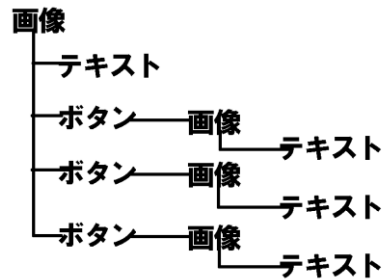
UI System Scene解説

- ・このシーンではGameObjectの親子関係の構築(特にUI)に力を入れました。
- ・まず起動したらこのような画面が出るとおもいます



左側のMenuと書かれているUIは親として背景の画像をセットしてMenuと書かれているテキストのUIとボタンの部分は背景画像の子としてセットすることでメニューのUIを実現しています。

ちなみにボタンのUIも親子関係を利用して背景画像の子としてテキストUIをセットして実現しています。



←図に書くとこのような構造になっています。

UI System Scene解説

- ・この親子関係の座標、回転、スケール値はこのように計算してます。↓



pPos=親の座標, pRot=親の回転度, pScl=親のスケール値,
lPos=ローカル座標, lRot=ローカル回転度, lScl=ローカルスケール値
だとすると

現在の座標 = pPos + lPos,

現在の回転度 = pRot + lRot,

現在のスケール値 = pScl + lScl,

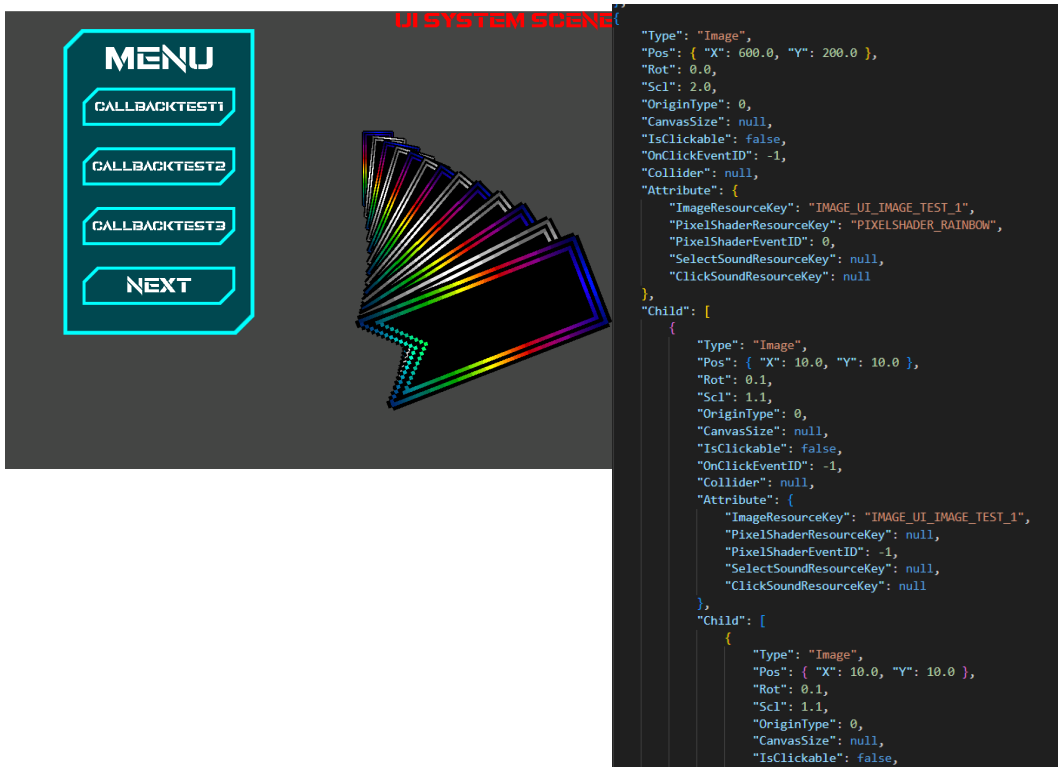
で計算をしています。

計算自体は簡単ですがこれをいざ実装しようとするとき少し親子関係の処理を工夫しないといけないので詳しくはGameObject2Dをご覧ください。

それと親と子の管理仕方にも力を入れてますので是非見てみてください。

UI System Scene解説

この右側の少しずつ回転している画像UIは親子関係を分かりやすくしたものです。



このシーンのUIは全てjsonで管理しています。上の”Pos”,”Rot”,”Sc1”と書いてある場所には直接参照する値をセットしているのですが、その次から値が小さくなっているのに実際の画面では少しずつ移動しながら回転、スケーリングされてると思います。

これはさっき解説した親子関係の計算によって実現しています。

それと親子関係とは関係ないですがjsonから各UIに簡単にピクセルシェーダーを適用したりできるように設計してるので詳しくはPixelShaderEventManagerクラスを見てください。

Inventory System Scene解説

- ・このシーンでは他のプロジェクトにすぐに移植できるインベントリとして使うことに特化したコンテナクラスを紹介してます。



ImGui側の操作方法を解説します。

まずDefaultPosというのがImGuiのウィンドウをデフォルトの座標にも戻すボタンです。

そしてここからが本題なのですがSlotというのが左側に描画されてるインベントリと同期していてDeleteを押すとそのアイテムスロットの中身が消えます。

そして下にあるAddItemIDに追加したいアイテムのIDを入れてAddItemというボタンを押すと自動で空きスロットを検索してアイテムを格納します。

空きスロットの検索処理なども工夫していますのでInventoryクラスを見てみて下さい。