\*\*\* this is an entirely new script \*\*\*

using UnityEditor;

using UnityEngine;

/// <summary> ##################################

///

/// NOTICE :

/// This script is just an editor extension to call and open the board layout custom

/// inspector onto a gui window. This makes it easier to see the custom inspector as

/// it is naturally going to be quite big.

///

/// DO NOT TOUCH UNLESS REQUIRED

///

/// </summary> ##################################

public class JSFBoardLayoutWeightsWindow : EditorWindow

{

static JSFBoardLayout bl;

static int pos = 0;

SerializedProperty colorWeights;

SerializedObject script;

public static void setDisplayType (int position)

{

pos = position;

EditorWindow bsc = EditorWindow.GetWindow (typeof (JSFBoardLayoutWeightsWindow),false, "Set Weights");

bsc.position = new Rect( Event.current.mousePosition.x + 150,Event.current.mousePosition.y + 250,

bsc.position.width,bsc.position.height);

}

void OnGUI () {

if( Selection.activeGameObject != null){

bl = Selection.activeGameObject.GetComponent<JSFBoardLayout>();

if (bl != null)

{

if(script == null) script = new SerializedObject(bl);

showWeights();

} else {

showErrorMsg(); // tells user to select the GameManger object

}

} else {

showErrorMsg(); // tells user to select the GameManger object

}

}

void showErrorMsg(){

EditorGUILayout.LabelField("\n\* Please select the object that contains the " +

"\"GameManager\" script.\nThen check back here again.", GUILayout.Height(45));

}

void showWeights(){

EditorGUILayout.LabelField("This weight is only for start-game layout~!\n" +

"For spawning weights, please go to the " +

"\n\"Normal Piece\" script in the " +

"PieceManager.", GUILayout.Height(45));

colorWeights = script.FindProperty("colorWeight").GetArrayElementAtIndex(pos);

EditorGUILayout.PropertyField(colorWeights,true);

// set the properties to be edited...

colorWeights.FindPropertyRelative("name").stringValue =

"Editing Board[" + pos%bl.gm.boardHeight + "," + ((bl.gm.boardWidth-1) - pos/bl.gm.boardHeight) + "]";

colorWeights.isExpanded = true;

colorWeights.FindPropertyRelative("weights").arraySize = bl.gm.NumOfActiveType;

colorWeights.FindPropertyRelative("weights").isExpanded = true;

script.ApplyModifiedProperties();

}

}