Hextron - Group 3

Generated by Doxygen 1.8.13

Contents

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

Chapter 2

Class Index

2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Chapter 3

File Index

3.1 File List

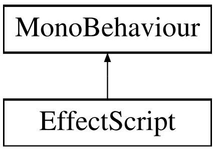
Here is a list of all documented files with brief descriptions:

Chapter 4

Class Documentation

4.1 EffectScript Class Reference

Inheritance diagram for EffectScript:



Static Public Member Functions

- static void playDestroy ()

    This function is called from TrapezoidCollision and plays the destroy block sound provided.

Private Member Functions

- void Start ()

    The Start() function is used to obtain the AudioSource component of the gameObject that contains this script.

Static Private Attributes

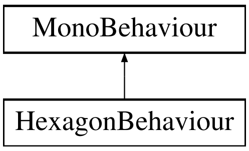
- static AudioSource **audioSrc**

The documentation for this class was generated from the following file:

- scripts/EffectScript.cs

4.2 HexagonBehaviour Class Reference

Inheritance diagram for HexagonBehaviour:



Public Attributes

- float **transitionSpeed** = 0.3f
- float **degree** = 0f
- bool **timeStay** = false
- float **hexagonHeightPixels** = 480f
- bool **pause** = false

Static Public Attributes

- static GameObject **blackHexagon** = null
- static Vector3 **areaZero**

Private Member Functions

- void Start ()

  The purpose of this function is to initialize the hexagon game object. It finds the GameObject and binds it to blackHexagon. An area defined as areaZero is used so that TrapezoidCollision can reference it for determine which position of the grid a trapezoid should go to.
- void Update ()

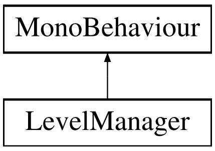
  The purpose of this function is to implement the rotation functionality of the hexagon. It checks if the gmae is paused, and if it isn't then it checks if the user pressed either the left or right arrow key. On the key press, it rotates the hexagon 60 degrees to the direction of the key. It also creates pausing of the game by preventing rotation of the hexagon and stopping all translations.

The documentation for this class was generated from the following file:

- scripts/HexagonBehaviour.cs

4.3 LevelManager Class Reference

Inheritance diagram for LevelManager:



Public Member Functions

- void LoadLevel (string name)

  The purpose of this function is to load an inputted level. This is determined by button presses for reaching different menus.
- void QuitRequest ()

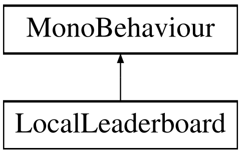
  The purpose of this function is to handle the quit function when it is requested.This only works in an executable file, and has been verified multiple times.

The documentation for this class was generated from the following file:

- scripts/LevelManager.cs

4.4 LocalLeaderboard Class Reference

Inheritance diagram for LocalLeaderboard:



Private Member Functions

- void Start ()

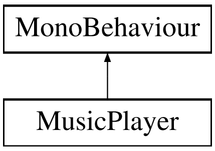
  The Start() function is used to update the highscores in the event of a new highscore.
- void **Update** ()

The documentation for this class was generated from the following file:

- scripts/LocalLeaderboard.cs

4.5 MusicPlayer Class Reference

Inheritance diagram for MusicPlayer:



Private Member Functions

- void Awake ()

  This function loads and plays the music file.

Static Private Attributes

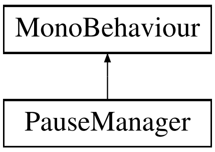
- static MusicPlayer **instance** = null

The documentation for this class was generated from the following file:

- scripts/MusicPlayer.cs

4.6 PauseManager Class Reference

Inheritance diagram for PauseManager:



Static Public Attributes

- static GameObject **canvas**
- static Text **text**
- static GameObject **camera**

Private Member Functions

- void Start ()

  The purpose of this function is to initialize the gameObjects within the context of this class. Canvas is static such that TrapezoidCollisions can refer to it for parenting when trapezoids have nothing underneath them after destroying.
- void Update ()

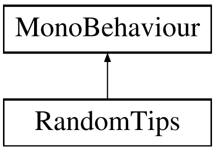
  The purpose of this function is always ensure the camera is in the middle of the canvas for proper positioning.

The documentation for this class was generated from the following file:

- scripts/PauseManager.cs

4.7 RandomTips Class Reference

Inheritance diagram for RandomTips:



Private Member Functions

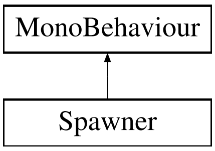
- void **Start** ()
- void **Update** ()

The documentation for this class was generated from the following file:

- scripts/RandomTips.cs

4.8 Spawner Class Reference

Inheritance diagram for Spawner:



Public Attributes

- GameObject [ ] **coloredTrapezoids**
- GameObject **blackTrapezoid**
- GameObject **rainbowTrapezoid**
- float **spawnSpeed**
- int **countForBlack** = 0
- int **countForRainbow** = 0

Static Public Attributes

- static float **angle** = 0

Private Member Functions

- void Start ()

  The Start() function is used to initialize all the variables and GameObjects needed to spawn a trapezoid.
- void spawnOne ()

  This function randomly generates one trapezoid. It randomly chooses one of 6 places to spawn and it also randomly chooses the colour between the four colours.
- void increaseRate ()

  This function increases the rate of spawning after a given period of time and reinvokes the spawning function given a new speed to implement.

Private Attributes

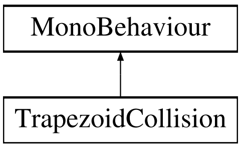
- GameObject **hexagon**
- GameObject **canvas**

The documentation for this class was generated from the following file:

- scripts/Spawner.cs

4.9 TrapezoidCollision Class Reference

Inheritance diagram for TrapezoidCollision:



Public Member Functions

- GameObject topObject (GameObject trapezoid)

  This function determines the top object inside of a stack. This is used for when a large stack is rotated into a currently falling trapezoid.
- void **destroyAllAroundHexagon** (int side, int row)
- void **destroyAllAroundRegular** (int side, int row)

Public Attributes

- float **hexagonHeightPixels** = 480f

Static Public Attributes

- static int **score** = 0
- static GameObject [,] **grid** = new GameObject[6,8]
- static bool [,] **visited** = new bool[6, 8]
- static ArrayList **connectedGameObjectsRow** = new ArrayList()
- static ArrayList **connectedGameObjectsSide** = new ArrayList()
- static int **enteringSide** = 0
- static int **enteringRow** = 0

Private Member Functions

- void OnCollisionEnter2D (Collision2D coll)

  The purpose of this function is to detect a collision between two different rigidbodies with colliders inside of the child rectangles for the particular trapezoid. This detects whether a collision with the black hexagon or any of the child trapezoids take place. If the gameObject that is hit is the black hexagon, the gameObject that this script is referring to will become a child of the hexagon. If not, then it will become a child of whichever trapezoid it hits.
- void addSelfAndChildren (GameObject trapezoidObject, int side, int row)

  This function adds the trapezoid that collides and recursively calls each of its children to add to the grid system as well.
- void floodfill (int side, int row, string rootObjectType)

  This function is a recursive algorithm that destroys blocks that are within a connection of three or greater. After all checks are completed, the game objects are destroyed.
- void ContinueFlowOfGame ()

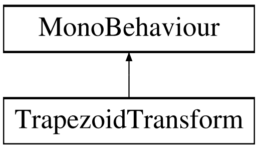
  This function continues the game after a floodfill is performed. It checks if a gameObject on the grid has nothing beneath it. If it doesn't, then the object and its children are removed from the grid, parented to the hexagon (so that it can rotate with the hexagon), and move towards the hexagon to move down however many rows necessary. Game over conditions are also checked!

The documentation for this class was generated from the following file:

- scripts/TrapezoidCollision.cs

4.10 TrapezoidTransform Class Reference

Inheritance diagram for TrapezoidTransform:



Public Member Functions

- void **LeftSideStick** (GameObject leftSide)
- void **RightSideStick** (GameObject rightSide)
- void **definingTrapezoid** (string rectangleName, string leftSideName, string rightSideName)

Public Attributes

- int **side** = 0
- int **row** = 0
- bool **move** = true
- bool **gravityFall** = false
- float **pixelsPerUnit** = 100f
- float **rectangleHeightPixels** = 271f
- float **rectangleWidthPixels** = 373f
- float **hexagonHeightPixels** = 480f
- float **scalePerUnit** = 0
- float **halfHexagonWorldUnit** = 0
- float **hexagonSideWorldUnit** = 0
- float **rectangleScaleOnSide** = 0
- string **objectType**

Private Member Functions

- void Start ()

  This is the intial setup the script does. One time run.
- void Update ()

  This runs every frame.

Private Attributes

- GameObject **rectangle**
- GameObject **leftSide**
- GameObject **rightSide**
- GameObject **hexagon**

4.10.1 Member Function Documentation

4.10.1.1 Start()

```
void TrapezoidTransform.Start ( )  [inline], [private]
```

This is the intial setup the script does. One time run.

The start method, which instantiates all trapezoid gameObjects with specific positions relative to the player.

4.10.1.2 Update()

```
void TrapezoidTransform.Update ( )  [inline], [private]
```

This runs every frame.

The update function defines the scaling factor that is required for every world unit of distance the rectangle is away from the player hexagon. It also defines the position of the two side triangles of yellow trapezoid to ensure that it they are always to the left and right of the rectangle.

The documentation for this class was generated from the following file:

- scripts/TrapezoidTransform.cs

Chapter 5

File Documentation

5.1 scripts/EffectScript.cs File Reference

The purpose of this file is to play sounds whenever a series of three or more objects is destroyed from a floodfill inside TrapezoidCollision.

Classes

- class EffectScript

5.1.1 Detailed Description

The purpose of this file is to play sounds whenever a series of three or more objects is destroyed from a floodfill inside TrapezoidCollision.

Author

Jason Li, Yousaf Shaheen, Scott Williams

Date

November 8th, 2017

5.2 scripts/HexagonBehaviour.cs File Reference

The purpose of this file is to implement the behaviour of the Hexagon, mostly the physical rotation of the Hexagon and pausing.

Classes

- class HexagonBehaviour

5.2.1 Detailed Description

The purpose of this file is to implement the behaviour of the Hexagon, mostly the physical rotation of the Hexagon and pausing.

Author

Jason Li, Yousaf Shaheen, Scott Williams

Date

November 8th, 2017

5.3 scripts/LevelManager.cs File Reference

The purpose of this file is to manage the loading of the game and transitioning between different game scenes.

Classes

- class LevelManager

5.3.1 Detailed Description

The purpose of this file is to manage the loading of the game and transitioning between different game scenes.

Author

Jason Li, Yousaf Shaheen, Scott Williams

Date

November 8th, 2017

5.4 scripts/LocalLeaderboard.cs File Reference

The purpose of this file is to update and store data to reflect highscores.

Classes

- class LocalLeaderboard

5.4.1 Detailed Description

The purpose of this file is to update and store data to reflect highscores.

Author

Jason Li, Yousaf Shaheen, Scott Williams

Date

December 6thth, 2017

5.5 scripts/MusicPlayer.cs File Reference

The purpose of this file is to manage the music and make sure the music is playing.

Classes

- class MusicPlayer

5.5.1 Detailed Description

The purpose of this file is to manage the music and make sure the music is playing.

Author

Jason Li, Yousaf Shaheen, Scott Williams

Date

November 8th, 2017

5.6 scripts/PauseManager.cs File Reference

The purpose of this file is to ensure that the canvas is always position towards the center of the camera and vice versa.

Classes

- class PauseManager

5.6.1 Detailed Description

The purpose of this file is to ensure that the canvas is always position towards the center of the camera and vice versa.

Author

Jason Li, Yousaf Shaheen, Scott Williams

Date

November 8th, 2017

5.7 scripts/Spawner.cs File Reference

The purpose of this file is to implement the spawning functionality for the trapezoids.

Classes

- class Spawner

5.7.1 Detailed Description

The purpose of this file is to implement the spawning functionality for the trapezoids.

Author

Jason Li, Yousaf Shaheen, Scott Williams

Date

November 8th, 2017

5.8 scripts/TrapezoidCollision.cs File Reference

The purpose of this file is to detect collisions between a gameObject and either the center black hexagon, or with a child of the hexagon (i.e. a trapezoid that is already on top of the hexagon).

Classes

- class TrapezoidCollision

5.8.1 Detailed Description

The purpose of this file is to detect collisions between a gameObject and either the center black hexagon, or with a child of the hexagon (i.e. a trapezoid that is already on top of the hexagon).

Author

Jason Li, Yousaf Shaheen, Scott Williams

Date

November 8th, 2017

5.9 scripts/TrapezoidTransform.cs File Reference

The purpose of this file is to define the behaviour for each color of trapezoid. This includes both the motion, scaling, and position of the trapezoids.

Classes

- class TrapezoidTransform

5.9.1 Detailed Description

The purpose of this file is to define the behaviour for each color of trapezoid. This includes both the motion, scaling, and position of the trapezoids.

Author

Jason Li, Yousaf Shaheen, Scott Williams

Date

October 26, 2017

Index