My Project

Generated by Doxygen 1.8.8

Thu Dec 4 2014 21:34:39

Contents

1 Hierarchical Index **1**

1.1 Class Hierarchy 1

2 Class Index **3**

2.1 Class List 3

3 Class Documentation **5**

3.1 box Class Reference 5

3.1.1 Member Data Documentation 5

3.1.1.1 finish 5

3.2 buttonPlay Class Reference 5

3.3 collision Class Reference 6

3.4 gravityShift Class Reference 6

3.5 LevelSelect Class Reference 6

3.5.1 Detailed Description 7

3.6 RobotControllerSript Class Reference 7

3.6.1 Detailed Description 7

3.7 RobotLEFT Class Reference 7

3.7.1 Detailed Description 8

3.8 RobotUP Class Reference 8

3.8.1 Detailed Description 8

3.9 trigger Class Reference 9

3.10 UIManager Class Reference 9

3.10.1 Detailed Description 10

3.11 win Class Reference 10

Index **11**

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

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

MonoBehaviour

- box . . . 5
- buttonPlay . . . 5
- collision . . . 6
- gravityShift . . . 6
- LevelSelect . . . 6
- RobotControllerSript . . . 7
- RobotLEFT . . . 7
- RobotUP . . . 8
- trigger . . . 9
- UIManager . . . 9
- win . . . 10

Chapter 2

Class Index

2.1 Class List

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

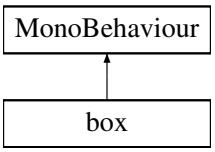
box 5
buttonPlay 5
collision 6
gravityShift 6
LevelSelect
    Level select: searches PlayerPrefs for key to unlock levels, if found, it will destroy the lock for the level 6
RobotControllerSript
    Robot controller sript. Used to Control player and change gravity in scene. Also used to detect collision on player 7
RobotLEFT
    RobotLEFT: This is used to start a level with the gravity oriented so the player falls left. Otherwise, it is the same script as RobotControllerScript 7
RobotUP
    RobotUP: This is used to start a level with the gravity inverted so the player falls upwards. Otherwise, it is the same script as RobotControllerScript 8
trigger 9
UIManager
    UI Manager: Creates functions to be called when a button is clicked in the main menu 9
win 10

Chapter 3

Class Documentation

3.1 box Class Reference

Inheritance diagram for box:



Public Attributes

- bool finish = false

  box: collision detection for the box.

3.1.1 Member Data Documentation

3.1.1.1 bool box.finish = false

box: collision detection for the box.

Parameters

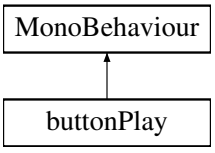
finish	Determines if the box has opened the door yet.

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

- box.cs

3.2 buttonPlay Class Reference

Inheritance diagram for buttonPlay:



Public Attributes

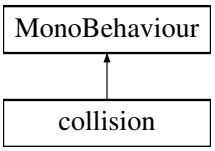
- GameObject **Play**
- GameObject **Levels**
- GameObject **Credits**

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

- buttonPlay.cs

3.3 collision Class Reference

Inheritance diagram for collision:

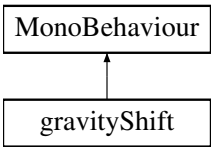


The documentation for this class was generated from the following file:

- collision.cs

3.4 gravityShift Class Reference

Inheritance diagram for gravityShift:



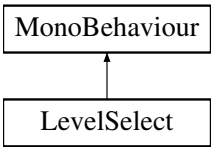
The documentation for this class was generated from the following file:

- gravityShift.cs

3.5 LevelSelect Class Reference

Level select: searches PlayerPrefs for key to unlock levels, if found, it will destroy the lock for the level.

Inheritance diagram for LevelSelect:



3.5.1 Detailed Description

Level select: searches PlayerPrefs for key to unlock levels, if found, it will destroy the lock for the level.

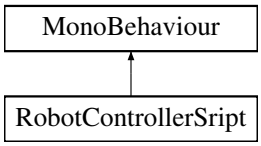
The documentation for this class was generated from the following file:

- LevelSelect.cs

3.6 RobotControllerSript Class Reference

Robot controller sript. Used to Control player and change gravity in scene. Also used to detect collision on player.

Inheritance diagram for RobotControllerSript:



Public Attributes

- float **maxSpeed** = 10f
- bool **facingRight** = true
- Transform **groundCheck**
- LayerMask **whatIsGround**
- float **jumpForce** = 700f
- int **jumpdir**
- AudioClip **crush**
- bool **dead** = false

3.6.1 Detailed Description

Robot controller sript. Used to Control player and change gravity in scene. Also used to detect collision on player.

Parameters

grounded	Detects if the player is standing on something
facingRight	Detects which direction the player is standing
whatIsGround	Allows the developer to dictate what is consider ground for grounded variable.
damping	Buffer to tell how much to flip character when gravity shifts
jumpdir	Variable to track gravity orientation
dead	Tracks if character was killed

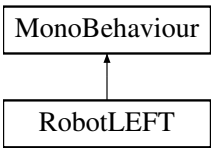
The documentation for this class was generated from the following file:

- RobotControllerSript.cs

3.7 RobotLEFT Class Reference

RobotLEFT: This is used to start a level with the gravity oriented so the player falls left. Otherwise, it is the same script as RobotControllerScript.

Inheritance diagram for RobotLEFT:



Public Attributes

- float **maxSpeed** = 10f
- Transform **groundCheck**
- LayerMask **whatIsGround**
- float **jumpForce** = 700f
- int **jumpdir**
- AudioClip **crush**

3.7.1 Detailed Description

RobotLEFT: This is used to start a level with the gravity oriented so the player falls left. Otherwise, it is the same script as RobotControllerScript.

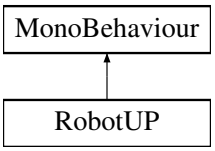
The documentation for this class was generated from the following file:

- RobotLEFT.cs

3.8 RobotUP Class Reference

RobotUP: This is used to start a level with the gravity inverted so the player falls upwards. Otherwise, it is the same script as RobotControllerScript.

Inheritance diagram for RobotUP:



Public Attributes

- float **maxSpeed** = 10f
- Transform **groundCheck**
- LayerMask **whatIsGround**
- float **jumpForce** = 700f
- int **jumpdir**
- AudioClip **crush**

3.8.1 Detailed Description

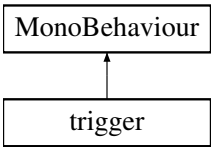
RobotUP: This is used to start a level with the gravity inverted so the player falls upwards. Otherwise, it is the same script as RobotControllerScript.

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

- RobotUP.cs

3.9 trigger Class Reference

Inheritance diagram for trigger:



Public Attributes

- GameObject **movingWall**
- string **doorOpen**
- GameObject **success**
- GameObject **elect1**
- GameObject **elect2**
- bool **isOpen** = false
- AudioClip **Zap**

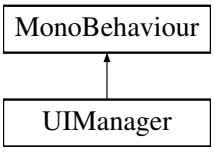
The documentation for this class was generated from the following file:

- trigger.cs

3.10 UIManager Class Reference

UI Manager: Creates functions to be called when a button is clicked in the main menu.

Inheritance diagram for UIManager:



Public Member Functions

- void **StartGame** ()
- void **Levels** ()
- void **Credits** ()
- void **TitleMenu** ()
- void **Mute** ()
- void **ErasePlayerPrefs** ()
- void **LL1** ()
- void **LL2** ()
- void **LL3** ()
- void **LL4** ()
- void **LL5** ()
- void **LL6** ()
- void **LL7** ()
- void **LL8** ()

- void **LL9** ()
- void **LL10** ()
- void **LL11** ()
- void **LL12** ()
- void **LL13** ()
- void **LL14** ()
- void **LL15** ()
- void **LL16** ()
- void **LL17** ()
- void **LL18** ()
- void **LL19** ()
- void **LL20** ()

3.10.1 Detailed Description

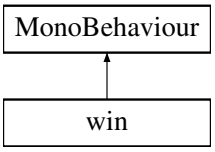
UI Manager: Creates functions to be called when a button is clicked in the main menu.

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

- UIManager.cs

3.11 win Class Reference

Inheritance diagram for win:



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

- win.cs

Index

box, 5
    finish, 5
buttonPlay, 5

collision, 6

finish
    box, 5

gravityShift, 6

LevelSelect, 6

RobotControllerSript, 7
RobotLEFT, 7
RobotUP, 8

trigger, 9

UIManager, 9

win, 10