

My Project

Generated by Doxygen 1.8.6

Thu Apr 23 2015 01:56:52

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	BombScript Class Reference	5
3.1.1	Member Function Documentation	5
3.1.1.1	OnTriggerEnter	5
3.1.1.2	Start	6
3.1.1.3	Update	6
3.1.2	Member Data Documentation	6
3.1.2.1	BombEnemy	6
3.1.2.2	EnemySpeed	6
3.1.2.3	ExplosionSD	6
3.2	Bullets Class Reference	6
3.2.1	Member Function Documentation	7
3.2.1.1	OnTriggerEnter	7
3.2.1.2	Update	7
3.2.2	Member Data Documentation	7
3.2.2.1	Bulletspeed	7
3.2.2.2	Dmg	7
3.3	CoinScript Class Reference	7
3.4	CrateScript Class Reference	8
3.4.1	Member Data Documentation	8
3.4.1.1	EnemySpeed	8
3.5	DestroyByTime Class Reference	8
3.5.1	Member Function Documentation	8
3.5.1.1	Start	8
3.5.2	Member Data Documentation	9
3.5.2.1	lifetime	9

3.6	GameCamera Class Reference	9
3.6.1	Member Function Documentation	9
3.6.1.1	IncrementTowards	9
3.6.1.2	SetTarget	10
3.6.1.3	Update	10
3.6.2	Member Data Documentation	10
3.6.2.1	target	10
3.6.2.2	trackSpeed	10
3.7	GameController Class Reference	10
3.7.1	Member Function Documentation	11
3.7.1.1	Start	11
3.7.1.2	Update	11
3.7.2	Member Data Documentation	11
3.7.2.1	hazard	11
3.7.2.2	hazardCount	11
3.7.2.3	Object1	11
3.7.2.4	Object2	11
3.7.2.5	Object3	12
3.7.2.6	spawnValues	12
3.7.2.7	spawnWait	12
3.7.2.8	startWait	12
3.7.2.9	waveWait	12
3.8	GameManager Class Reference	12
3.8.1	Member Function Documentation	13
3.8.1.1	SpawnPlayer	13
3.8.1.2	Start	13
3.8.2	Member Data Documentation	13
3.8.2.1	cam	13
3.8.2.2	player	13
3.9	Health Class Reference	13
3.9.1	Member Function Documentation	14
3.9.1.1	isDead	14
3.9.1.2	ReceiveDmg	14
3.9.2	Member Data Documentation	14
3.9.2.1	currentHealth	14
3.10	MedPack Class Reference	14
3.10.1	Member Function Documentation	15
3.10.1.1	OnTriggerEnter	15
3.10.1.2	Update	15
3.10.2	Member Data Documentation	15

3.10.2.1	HealAMT	15
3.10.2.2	MedSpeed	15
3.11	ObjectBehavior2 Class Reference	15
3.11.1	Member Data Documentation	15
3.11.1.1	speed	15
3.12	PlayerController Class Reference	16
3.12.1	Member Function Documentation	17
3.12.1.1	Start	17
3.12.1.2	Update	17
3.12.2	Member Data Documentation	17
3.12.2.1	acceleration	17
3.12.2.2	amountToMove	17
3.12.2.3	currentSpeed	17
3.12.2.4	DeathSD	17
3.12.2.5	gravity	17
3.12.2.6	jumpHeight	17
3.12.2.7	JumpSD	17
3.12.2.8	Player	17
3.12.2.9	playerPhysics	17
3.12.2.10	RollSD	18
3.12.2.11	ShotSD	18
3.12.2.12	speed	18
3.12.2.13	targetSpeed	18
3.13	PlayerPhysics Class Reference	18
3.13.1	Member Function Documentation	19
3.13.1.1	Move	19
3.13.1.2	Start	20
3.13.2	Member Data Documentation	20
3.13.2.1	collisionMask	20
3.13.2.2	grounded	20
3.13.2.3	movementStopped	20
3.14	SpawnFloor Class Reference	20
3.14.1	Member Data Documentation	21
3.14.1.1	floor	21
3.14.1.2	spawnValues	21
3.14.1.3	spawnWait	21
3.14.1.4	startWait	21
3.15	SpikeObstacle Class Reference	21
3.16	TestColliderandHealth Class Reference	21
3.16.1	Member Function Documentation	22

3.16.1.1	Update	22
3.16.2	Member Data Documentation	22
3.16.2.1	NewHealth	22
3.16.2.2	PreviousHealth	22
Index		23

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

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

MonoBehaviour	
BombScript	5
Bullets	6
CoinScript	7
CrateScript	8
DestroyByTime	8
GameCamera	9
GameController	10
GameManager	12
Health	13
PlayerController	16
MedPack	14
ObjectBehavior2	15
PlayerPhysics	18
SpawnFloor	20
SpikeObstacle	21
TestColliderandHealth	21

Chapter 2

Class Index

2.1 Class List

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

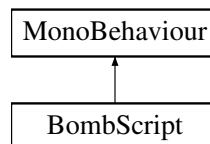
BombScript	5
Bullets	6
CoinScript	7
CrateScript	8
DestroyByTime	8
GameCamera	9
GameController	10
GameManager	12
Health	13
MedPack	14
ObjectBehavior2	15
PlayerController	16
PlayerPhysics	18
SpawnFloor	20
SpikeObstacle	21
TestColliderandHealth	21

Chapter 3

Class Documentation

3.1 BombScript Class Reference

Inheritance diagram for BombScript:



Public Member Functions

- void **Start** ()
Stuff to do at the start, gets attached animator and audio source
- void **Update** ()
Updates per frame, moves object left per frame, checks if dead, if true, starts death animation& sound then destroys object
- void **OnTriggerEnter** (Collider col)

Public Attributes

- Animator **BombEnemy**
The bomb enemy.
- AudioSource **ExplosionSD**
The explosion S.
- float **EnemySpeed** = 0.1f
The enemy speed.

3.1.1 Member Function Documentation

3.1.1.1 void BombScript.OnTriggerEnter (Collider col) [inline]

Raises the trigger enter event.

Parameters

<i>col</i>	Col collision parameter to detect collisions
<i>return</i>	Tag "Player" collision causes Health call, Dmg received & object destruct

3.1.1.2 void BombScript.Start () [inline]

Stuff to do at the start, gets attached animator and audio source

3.1.1.3 void BombScript.Update () [inline]

Updates per frame, moves object left per frame, checks if dead, if true, starts death animation& sound then destroys object

3.1.2 Member Data Documentation

3.1.2.1 Animator BombScript.BombEnemy

The bomb enemy.

3.1.2.2 float BombScript.EnemySpeed = 0.1f

The enemy speed.

3.1.2.3 AudioSource BombScript.ExplosionSD

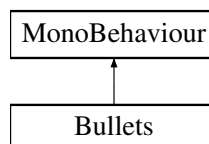
The explosion S.

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

- BombScript.cs

3.2 Bullets Class Reference

Inheritance diagram for Bullets:



Public Member Functions

- void [Update](#) ()
Update called once per frame, moves object right, destroys object after certain time
- void [OnTriggerEnter](#) (Collider Hit)
Collider trigger, on hit do something

Public Attributes

- float `Bulletspeed` = 5
The bulletspeed.
- float `Dmg` = 5
The damage.

3.2.1 Member Function Documentation

3.2.1.1 void Bullets.OnTriggerEnter (Collider *Hit*) [*inline*]

Collider trigger, on hit do something

Parameters

<i>Hit</i>	Hit collision detect
<i>Return</i>	> If hit is obstacle, destroy object and this, if hit is enemy, enemy is damaged

3.2.1.2 void Bullets.Update () [*inline*]

Update called once per frame, moves object right, destroys object after certain time

3.2.2 Member Data Documentation

3.2.2.1 float Bullets.Bulletspeed = 5

The bulletspeed.

3.2.2.2 float Bullets.Dmg = 5

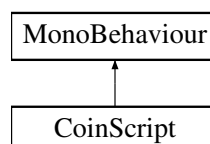
The damage.

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

- Bullets.cs

3.3 CoinScript Class Reference

Inheritance diagram for CoinScript:

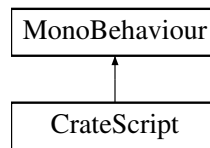


The documentation for this class was generated from the following file:

- CoinScript.cs

3.4 CrateScript Class Reference

Inheritance diagram for CrateScript:



Public Attributes

- float `EnemySpeed` = 0.1f
Speed of GameObject.

3.4.1 Member Data Documentation

3.4.1.1 float `CrateScript.EnemySpeed` = 0.1f

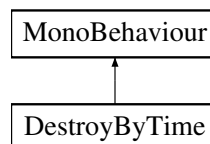
Speed of GameObject.

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

- `CrateScript.cs`

3.5 DestroyByTime Class Reference

Inheritance diagram for DestroyByTime:



Public Member Functions

- void `Start` ()
Inititation, destroy object at lifetime

Public Attributes

- float `lifetime`
The lifetime of an object before being destroyed

3.5.1 Member Function Documentation

3.5.1.1 void `DestroyByTime.Start` () [inline]

Inititation, destroy object at lifetime

3.5.2 Member Data Documentation

3.5.2.1 float DestroyByTime.lifetime

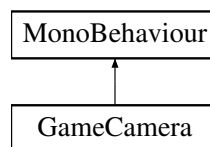
The lifetime of an object before being destroyed

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

- DestroyByTime.cs

3.6 GameCamera Class Reference

Inheritance diagram for GameCamera:



Public Member Functions

- void [SetTarget](#) (Transform t)
Sets the target to aim for
- void [Update](#) ()
Update per frame, transforms object to target vector
- float [IncrementTowards](#) (float n, float [target](#), float a)
Increments param n towards target by speed

Public Attributes

- Transform [target](#)
Vector target to aim for.
- float [trackSpeed](#) = 10
The track speed.

3.6.1 Member Function Documentation

3.6.1.1 float GameCamera.IncrementTowards (float n, float [target](#), float a) [inline]

Increments param n towards target by speed

Returns

The towards.

Parameters

<i>n</i>	First vector, current position vector of cam
<i>target</i>	Second vector, target vector for cam
<i>a</i>	The alpha component.

3.6.1.2 void GameCamera.SetTarget (Transform *t*) [inline]

Sets the target to aim for

Parameters

<i>t</i>	vector to aim for
----------	-------------------

3.6.1.3 void GameCamera.Update () [inline]

Update per frame, transforms object to target vector

3.6.2 Member Data Documentation

3.6.2.1 Transform GameCamera.target

Vector target to aim for.

3.6.2.2 float GameCamera.trackSpeed = 10

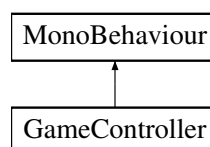
The track speed.

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

- GameCamera.cs

3.7 GameController Class Reference

Inheritance diagram for GameController:



Public Member Functions

- void [Update](#) ()

Update per frame, randomizer for spawning between objects

- void [Start](#) ()

initializer to start time for waves, sets hazards to spawn, spawn time, wave time and position to spawn

Public Attributes

- GameObject [Object1](#)
1st Placement for gameObject spawning.
- GameObject [Object2](#)
2nd Placement for gameObject spawning.
- GameObject [Object3](#)
3rd Placement for gameObject spawning.
- GameObject [hazard](#)
Hazard spot for randomizing between 3 objects.
- Vector3 [spawnValues](#)
The spawn values.
- int [hazardCount](#)
The hazard count.
- float [spawnWait](#)
The spawn wait.
- float [startWait](#)
The start wait.
- float [waveWait](#)
The wave wait.
- int [choice](#)
Float Randomizer for objects summary>

3.7.1 Member Function Documentation

3.7.1.1 void GameController.Start () [inline]

initializer to start time for waves, sets hazards to spawn, spawn time, wave time and position to spawn

3.7.1.2 void GameController.Update () [inline]

Update per frame, randomizer for spawning between objects

3.7.2 Member Data Documentation

3.7.2.1 GameObject GameController.hazard

Hazard spot for randomizing between 3 objects.

3.7.2.2 int GameController.hazardCount

The hazard count.

3.7.2.3 GameObject GameController.Object1

1st Placement for gameObject spawning.

3.7.2.4 GameObject GameController.Object2

2nd Placement for gameObject spawning.

3.7.2.5 `GameObject GameController.Object3`

3rd Placement for gameObject spawning.

3.7.2.6 `Vector3 GameController.spawnValues`

The spawn values.

3.7.2.7 `float GameController.spawnWait`

The spawn wait.

3.7.2.8 `float GameController.startWait`

The start wait.

3.7.2.9 `float GameController.waveWait`

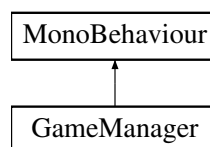
The wave wait.

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

- `GameController.cs`

3.8 GameManager Class Reference

Inheritance diagram for GameManager:



Public Member Functions

- void `Start ()`
At start, get attached gameObject and spawn player
- void `SpawnPlayer ()`
Spawns the player and sets camera on player

Public Attributes

- `GameObject player`
GameObject spot to place player.
- `GameCamera cam`
GameObject camera.

3.8.1 Member Function Documentation

3.8.1.1 void GameManager.SpawnPlayer () [inline]

Spawns the player and sets camera on player

3.8.1.2 void GameManager.Start () [inline]

At start, get attached gameobject and spawn player

3.8.2 Member Data Documentation

3.8.2.1 GameCamera GameManager.cam

Gameobject camera.

3.8.2.2 GameObject GameManager.player

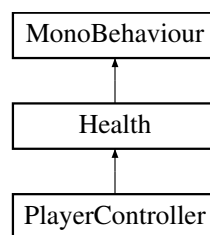
Gameobject spot to place player.

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

- GameManager.cs

3.9 Health Class Reference

Inheritance diagram for Health:



Public Member Functions

- void [ReceiveDmg](#) (float dmg)
Damage function, subtracts dmg from current health ///
- bool [isDead](#) ()
Checks if the object is dead depending on current health ///

Public Attributes

- float [currentHealth](#)
The current health.

3.9.1 Member Function Documentation

3.9.1.1 `bool Health.isDead () [inline]`

Checks if the object is dead depending on current health ///

Returns

`true`, if dead was dead, `false` target is still alive.

3.9.1.2 `void Health.ReceiveDmg (float dmg) [inline]`

Damage function, subtracts dmg from current health ///

Parameters

<i>dmg</i>	Damage to take
------------	----------------

3.9.2 Member Data Documentation

3.9.2.1 `float Health.currentHealth`

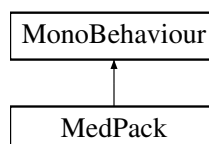
The current health.

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

- Health.cs

3.10 MedPack Class Reference

Inheritance diagram for MedPack:



Public Member Functions

- void `Update` ()
Updates per frame, moves object left
- void `OnTriggerEnter` (Collider col)
Object trigger on collision

Public Attributes

- float `MedSpeed` = 0.1f
The medpack speed.
- float `HealAMT` = 2
The total heal amount.

3.10.1 Member Function Documentation

3.10.1.1 void MedPack.OnTriggerEnter (Collider col) [inline]

Object trigger on collision

Parameters

<i>col</i>	Collider col
<i>Return</i>	If col == player, player gets health and this is destroyed

3.10.1.2 void MedPack.Update () [inline]

Updates per frame, moves object left

3.10.2 Member Data Documentation

3.10.2.1 float MedPack.HealAMT = 2

The total heal amount.

3.10.2.2 float MedPack.MedSpeed = 0.1f

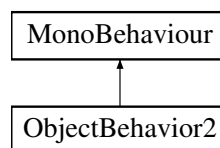
The medpack speed.

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

- MedPack.cs

3.11 ObjectBehavior2 Class Reference

Inheritance diagram for ObjectBehavior2:



Public Attributes

- float **speed** = -20f
The speed of GameObject.

3.11.1 Member Data Documentation

3.11.1.1 float ObjectBehavior2.speed = -20f

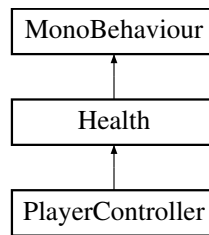
The speed of GameObject.

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

- ObjectBehavior2.cs

3.12 PlayerController Class Reference

Inheritance diagram for PlayerController:



Public Member Functions

- void [Start](#) ()
Gets player physics, animator and sets audio sources to listen
- void [Update](#) ()
Update per frame. check if player is dead, sets player speed, checks for user input

Public Attributes

- float [gravity](#) = 20
The gravity.
- float [speed](#) = 8
The speed.
- float [acceleration](#) = 30
The acceleration.
- float [jumpHeight](#) = 12
The height of the jump.
- float [currentSpeed](#)
The current speed.
- float [targetSpeed](#)
The target speed.
- Vector2 [amountToMove](#)
The amount to move.
- AudioSource [ShotSD](#)
The shot Sound.
- AudioSource [RollSD](#)
The roll Sound.
- AudioSource [JumpSD](#)
The jump Sound.
- AudioSource [DeathSD](#)
The death Sound.
- [PlayerPhysics](#) [playerPhysics](#)
Gets player physics script.
- Animator [Player](#)
Gets player animator.

3.12.1 Member Function Documentation

3.12.1.1 void PlayerController.Start () [inline]

Gets player physics, animator and sets audio sources to listen

3.12.1.2 void PlayerController.Update () [inline]

Update per frame. check if player is dead, sets player speed, checks for user input

3.12.2 Member Data Documentation

3.12.2.1 float PlayerController.acceleration = 30

The acceleration.

3.12.2.2 Vector2 PlayerController.amountToMove

The amount to move.

3.12.2.3 float PlayerController.currentSpeed

The current speed.

3.12.2.4 AudioSource PlayerController.DeathSD

The death Sound.

3.12.2.5 float PlayerController.gravity = 20

The gravity.

3.12.2.6 float PlayerController.jumpHeight = 12

The height of the jump.

3.12.2.7 AudioSource PlayerController.JumpSD

The jump Sound.

3.12.2.8 Animator PlayerController.Player

Gets player animator.

3.12.2.9 PlayerPhysics PlayerController.playerPhysics

Gets player physics script.

3.12.2.10 AudioSource PlayerController.RollSD

The roll Sound.

3.12.2.11 AudioSource PlayerController.ShotSD

The shot Sound.

3.12.2.12 float PlayerController.speed = 8

The speed.

3.12.2.13 float PlayerController.targetSpeed

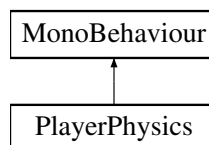
The target speed.

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

- PlayerController.cs

3.13 PlayerPhysics Class Reference

Inheritance diagram for PlayerPhysics:



Public Member Functions

- void **Start** ()
Initialization, gets collider of player and its values(size and center)
- void **Move** (Vector2 moveAmount)
Move the specified moveAmount.
- void **SetCollider** (Vector3 size, Vector3 centre)
- void **ResetCollider** ()

Public Attributes

- LayerMask **collisionMask**
Set the layer for collisionmask, player will collide with this layer
- bool **grounded**
Boolean to check if object is in contact with object from bottom
- bool **movementStopped**
Boolean to check player is no longer moving

3.13.1 Member Function Documentation

3.13.1.1 void PlayerPhysics.Move (*Vector2 moveAmount*) [inline]

Move the specified moveAmount.

Parameters

<i>moveAmount</i>	Move amount.
-------------------	--------------

3.13.1.2 void PlayerPhysics.Start () [inline]

Initialization, gets collider of player and its values(size and center)

3.13.2 Member Data Documentation

3.13.2.1 LayerMask PlayerPhysics.collisionMask

Set the layer for collisionmask, player will collide with this layer

3.13.2.2 bool PlayerPhysics.grounded

Boolean to check if object is in contact with object from bottom

3.13.2.3 bool PlayerPhysics.movementStopped

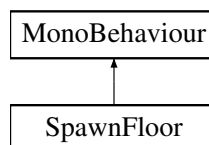
Boolean to check player is no longer moving

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

- PlayerPhysics.cs

3.14 SpawnFloor Class Reference

Inheritance diagram for SpawnFloor:



Public Attributes

- GameObject [floor](#)
The GameObject that acts as the main floor of the game.
- Vector3 [spawnValues](#)
The spawn position values.
- float [spawnWait](#)
The delay between object spawn times.
- float [startWait](#)
The delay between the start of the game and the start of the floor spawning.

3.14.1 Member Data Documentation

3.14.1.1 GameObject SpawnFloor.floor

The GameObject that acts as the main floor of the game.

3.14.1.2 Vector3 SpawnFloor.spawnValues

The spawn position values.

3.14.1.3 float SpawnFloor.spawnWait

The delay between object spawn times.

3.14.1.4 float SpawnFloor.startWait

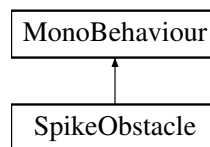
The delay between the start of the game and the start of the floor spawning.

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

- SpawnFloor.cs

3.15 SpikeObstacle Class Reference

Inheritance diagram for SpikeObstacle:



Public Attributes

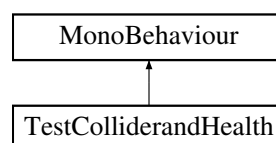
- float **ObstacleSpeed** = 10

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

- SpikeObstacle.cs

3.16 TestColliderandHealth Class Reference

Inheritance diagram for TestColliderandHealth:



Public Member Functions

- void `Update` ()
Update per frame to move object left to collide with player

Public Attributes

- float `PreviousHealth` = 0
Holder float to check previous health.
- float `NewHealth` = 0
Holder to check new health.

3.16.1 Member Function Documentation

3.16.1.1 void `TestColliderandHealth.Update` () [`inline`]

Update per frame to move object left to collide with player

3.16.2 Member Data Documentation

3.16.2.1 float `TestColliderandHealth.NewHealth` = 0

Holder to check new health.

3.16.2.2 float `TestColliderandHealth.PreviousHealth` = 0

Holder float to check previous health.

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

- `TestColliderandHealth.cs`

Index

- acceleration
 - PlayerController, [17](#)
- amountToMove
 - PlayerController, [17](#)
- BombEnemy
 - BombScript, [6](#)
- BombScript, [5](#)
 - BombEnemy, [6](#)
 - EnemySpeed, [6](#)
 - ExplosionSD, [6](#)
 - OnTriggerEnter, [5](#)
 - Start, [6](#)
 - Update, [6](#)
- Bullets, [6](#)
 - Bulletspeed, [7](#)
 - Dmg, [7](#)
 - OnTriggerEnter, [7](#)
 - Update, [7](#)
- Bulletspeed
 - Bullets, [7](#)
- cam
 - GameManager, [13](#)
- CoinScript, [7](#)
- collisionMask
 - PlayerPhysics, [20](#)
- CrateScript, [8](#)
 - EnemySpeed, [8](#)
- currentHealth
 - Health, [14](#)
- currentSpeed
 - PlayerController, [17](#)
- DeathSD
 - PlayerController, [17](#)
- DestroyByTime, [8](#)
 - lifetime, [9](#)
 - Start, [8](#)
- Dmg
 - Bullets, [7](#)
- EnemySpeed
 - BombScript, [6](#)
 - CrateScript, [8](#)
- ExplosionSD
 - BombScript, [6](#)
- floor
 - SpawnFloor, [21](#)
- GameCamera, [9](#)
 - IncrementTowards, [9](#)
 - SetTarget, [10](#)
 - target, [10](#)
 - trackSpeed, [10](#)
 - Update, [10](#)
- GameController, [10](#)
 - hazard, [11](#)
 - hazardCount, [11](#)
 - Object1, [11](#)
 - Object2, [11](#)
 - Object3, [11](#)
 - spawnValues, [12](#)
 - spawnWait, [12](#)
 - Start, [11](#)
 - startWait, [12](#)
 - Update, [11](#)
 - waveWait, [12](#)
- GameManager, [12](#)
 - cam, [13](#)
 - player, [13](#)
 - SpawnPlayer, [13](#)
 - Start, [13](#)
- gravity
 - PlayerController, [17](#)
- grounded
 - PlayerPhysics, [20](#)
- hazard
 - GameController, [11](#)
- hazardCount
 - GameController, [11](#)
- HealAMT
 - MedPack, [15](#)
- Health, [13](#)
 - currentHealth, [14](#)
 - isDead, [14](#)
 - ReceiveDmg, [14](#)
- IncrementTowards
 - GameCamera, [9](#)
- isDead
 - Health, [14](#)
- jumpHeight
 - PlayerController, [17](#)
- JumpSD
 - PlayerController, [17](#)
- lifetime

- DestroyByTime, [9](#)
- MedPack, [14](#)
 - HealAMT, [15](#)
 - MedSpeed, [15](#)
 - OnTriggerEnter, [15](#)
 - Update, [15](#)
- MedSpeed
 - MedPack, [15](#)
- Move
 - PlayerPhysics, [19](#)
- movementStopped
 - PlayerPhysics, [20](#)
- NewHealth
 - TestColliderandHealth, [22](#)
- Object1
 - GameController, [11](#)
- Object2
 - GameController, [11](#)
- Object3
 - GameController, [11](#)
- ObjectBehavior2, [15](#)
 - speed, [15](#)
- OnTriggerEnter
 - BombScript, [5](#)
 - Bullets, [7](#)
 - MedPack, [15](#)
- Player
 - PlayerController, [17](#)
- player
 - GameManager, [13](#)
- PlayerController, [16](#)
 - acceleration, [17](#)
 - amountToMove, [17](#)
 - currentSpeed, [17](#)
 - DeathSD, [17](#)
 - gravity, [17](#)
 - jumpHeight, [17](#)
 - JumpSD, [17](#)
 - Player, [17](#)
 - playerPhysics, [17](#)
 - RollSD, [17](#)
 - ShotSD, [18](#)
 - speed, [18](#)
 - Start, [17](#)
 - targetSpeed, [18](#)
 - Update, [17](#)
- PlayerPhysics, [18](#)
 - collisionMask, [20](#)
 - grounded, [20](#)
 - Move, [19](#)
 - movementStopped, [20](#)
 - Start, [20](#)
- playerPhysics
 - PlayerController, [17](#)
- PreviousHealth
 - TestColliderandHealth, [22](#)
- ReceiveDmg
 - Health, [14](#)
- RollSD
 - PlayerController, [17](#)
- SetTarget
 - GameCamera, [10](#)
- ShotSD
 - PlayerController, [18](#)
- SpawnFloor, [20](#)
 - floor, [21](#)
 - spawnValues, [21](#)
 - spawnWait, [21](#)
 - startWait, [21](#)
- SpawnPlayer
 - GameManager, [13](#)
- spawnValues
 - GameController, [12](#)
 - SpawnFloor, [21](#)
- spawnWait
 - GameController, [12](#)
 - SpawnFloor, [21](#)
- speed
 - ObjectBehavior2, [15](#)
 - PlayerController, [18](#)
- SpikeObstacle, [21](#)
- Start
 - BombScript, [6](#)
 - DestroyByTime, [8](#)
 - GameController, [11](#)
 - GameManager, [13](#)
 - PlayerController, [17](#)
 - PlayerPhysics, [20](#)
- startWait
 - GameController, [12](#)
 - SpawnFloor, [21](#)
- target
 - GameCamera, [10](#)
- targetSpeed
 - PlayerController, [18](#)
- TestColliderandHealth, [21](#)
 - NewHealth, [22](#)
 - PreviousHealth, [22](#)
 - Update, [22](#)
- trackSpeed
 - GameCamera, [10](#)
- Update
 - BombScript, [6](#)
 - Bullets, [7](#)
 - GameCamera, [10](#)
 - GameController, [11](#)
 - MedPack, [15](#)
 - PlayerController, [17](#)
 - TestColliderandHealth, [22](#)
- waveWait

GameController, [12](#)