

Contents

X

1. 7161 27日

- · 게임이름
- · 71/2 LHB
- 거리소사법

2. Menu Scene

- Main Menu
- Button

3. Game Scene

- Player
- Platform
- · Level Generator
- Kill Zone
- Finish Line
- Game Over
- Score
- · Message Output
- Audio Manager
- Sound

7106 274

710님 17 - 게임 제목

THE DEATH RACE

710님 17 - 710님 내용

모두가 쇎로 감영된 학교에서 홀로 감영되지 않은 한 학생이 탈출하는 게임



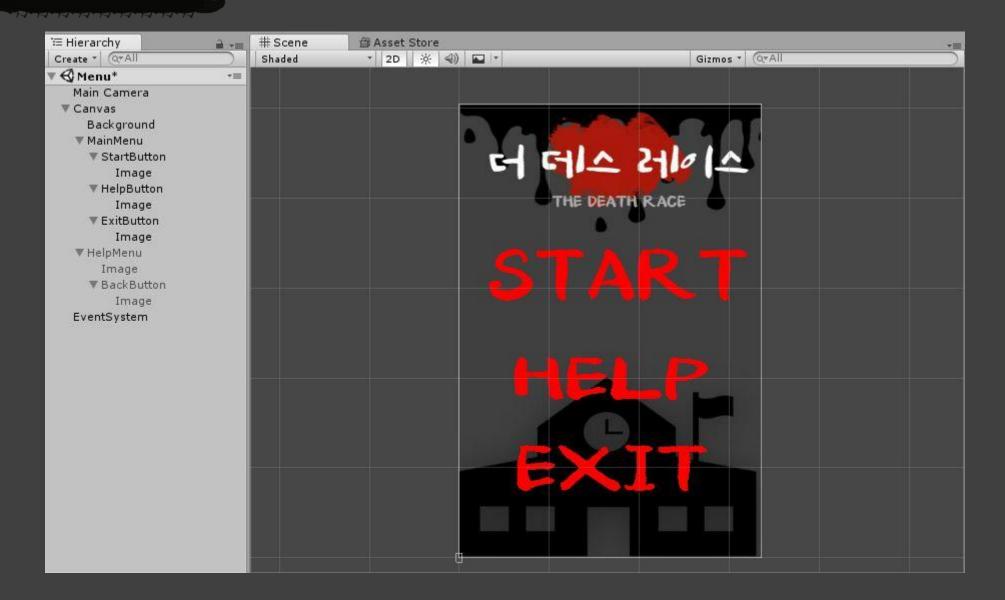
710님 27H - 게임 소사법

수인공을 좌우시하기로 조용하여 아라이에서 올라오는 참비의 손길로부터 도마차 위로 내247는 7세일

2

Menu Scene

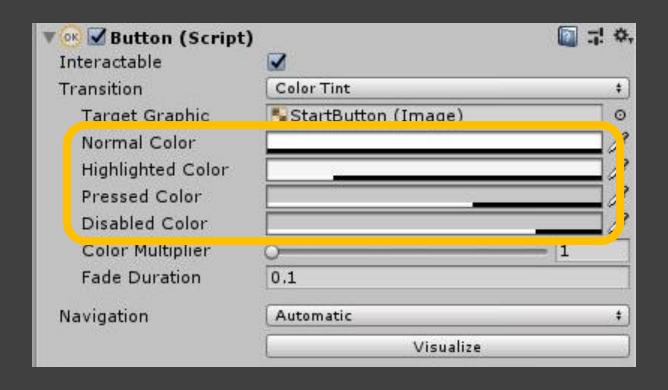
Menu Scene



Menu Scene - Main Menu

```
⊟using System.Collections;
       using System.Collections.Generic;
       using UnityEngine;
       using UnityEngine.SceneManagement;
      □public class MainMenu : MonoBehaviour {
            public void PlayGame()
                SceneManager.LoadScene(SceneManager.GetActiveScene().buildIndex + 1);
13
14
            public void QuitGame()
                Debug.Log("QUIT!");
                Application.Quit();
19
20
21
22
```

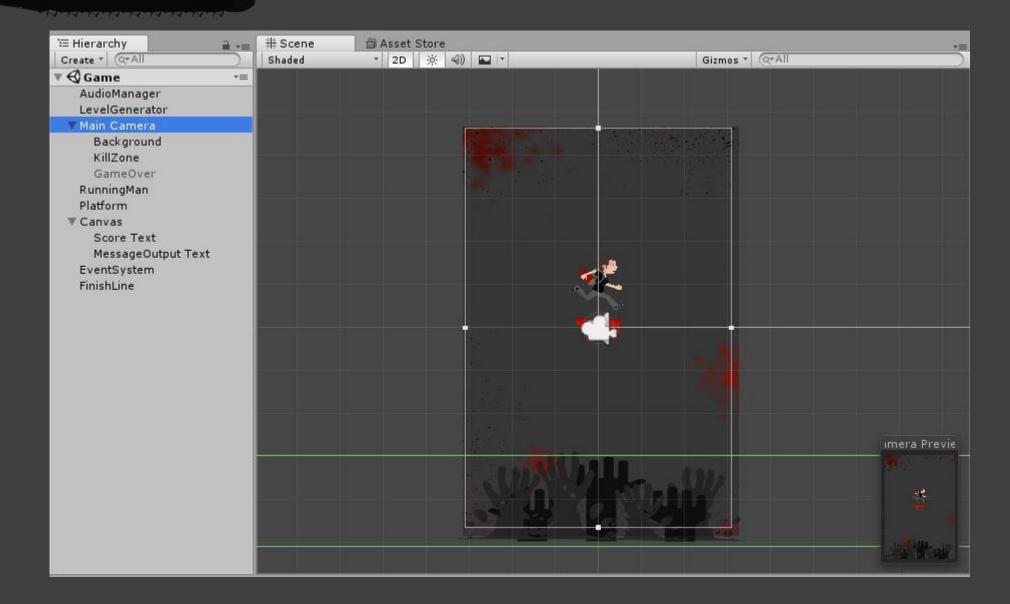
Menu Scene - Button 叶十十立工



3

Game Scene

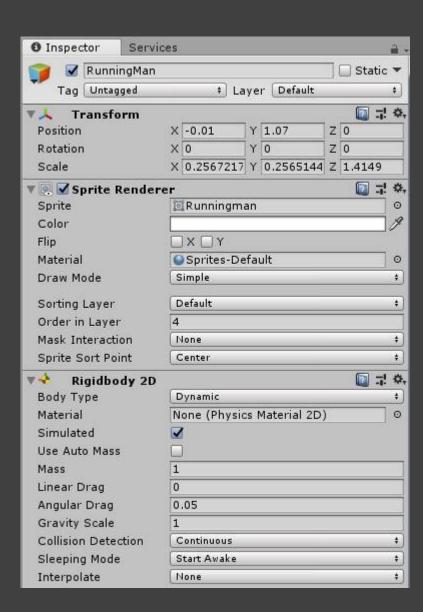
Game Scene

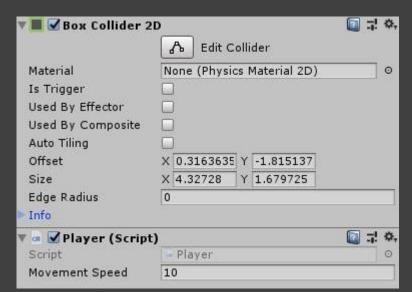


Game Scene - Player (Running Man)



A H H H H H H H





Game Scene - Player (Running Man)

```
private void MoveClamp()
72
73
74
               Vector2 left = Camera.main.YiewportToWorldPoint(new Yector2(0, 0));
75
               Vector2 right = Camera.main.ViewportToWorldPoint(new Vector2(1, 1));
76
77
78
               Vector2 player = transform.position;
79
               player.x = Mathf.Clamp(player.x, left.x + 0.8F, right.x - 0.8F);
80
81
82
               player.y = Mathf.Clamp(player.y, left.y + 1, right.y - 1);
83
84
               transform.position = player;
85
```

And and and and and and and

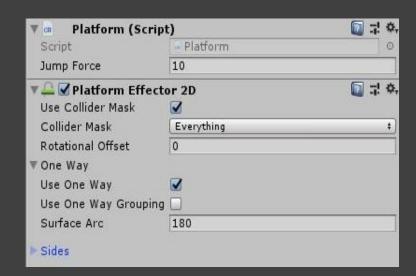
```
private void OnTriggerEnter2D(Collider2D collision)
                if (GameManager.currentState.Equals(GameState.Gameover)) return;
 88
 89
                switch(collision.tag)
 90
 91
                    case "KillZone":
                        GameManager.currentState = GameState.Gameover;
 93
                        SetGaravityScale(0);
 94
                        break;
                    case "FinalPlatform":
                        GameManager.currentState = GameState.Clear;
97
                        print("Clear Level.");
                        break;
99
100
101
```

Game Scene - Platform



THE HARMAN

1 Inspector Service	es:	+≡
Platform	□ Static	
Tag Untagged	Layer Default	+
▼▲ Transform	<u> </u>	20
Position	X 0 Y 0 Z 0	~
Rotation	X 0 Y 0 Z 0	
Scale	X 0.4721169 Y 0.6143647 Z 1	
▼ 💽 🗹 Sprite Rendere		*
Sprite	Platform	0
Color		1
Flip	$\square \times \square Y$	
Material	Sprites-Default	0
Draw Mode	Simple	+
Sorting Layer	Default	
Order in Layer	0	
Mask Interaction	None	÷
Sprite Sort Point	Center	+
▼ 🚵 🗹 Edge Collider 2	D = 1	ď
Luge comuer 2	A Edit Collider	2000
Material	None (Physics Material 2D)	0
Is Trigger		
Used By Effector	Z	
Offset	X 0 Y 0	
Edge Radius	0	
▼ Points		
Size	2	
Element 0	X -1.050326 Y 0.3635605	
Element 1	X 1.073687 Y 0.3635605	
▶ Info		



Game Scene - Platform

```
⊟using System.Collections;
       using System.Collections.Generic;
       using UnityEngine;
      public class Platform : MonoBehaviour
           public float jumpForce = 5f;
           public void SetActivation(bool activation)
10
               gameObject.SetActive(activation);
12
13
           public void SetPosition(Vector3 position)
15
16
               transform.position = position;
18
           void OnCollisionEnter2D(Collision2D collision)
19
20
21
               if (GameManager.currentState.Equals(GameState.Gameover)) return;
22
23
               if (collision.relativeVelocity.y <= Of )
24
25
                   Rigidbody2D rb = collision.collider.GetComponent<Rigidbody2D>();
26
27
                    if (rb != null)
28
29
                       Vector2 velocity = rb.velocity;
30
                       velocity.y = jumpForce;
31
                       rb.velocity = velocity;
32
33
34
35
36
```

Game Scene - Level Generator

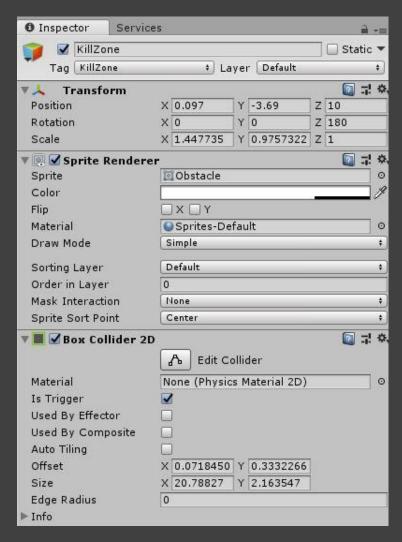
```
Flusing System.Collections:
       using System.Collections.Generic;
      using UnityEngine;
      Epublic class LevelGenerator: MonoBehaviour
6
7
8
           public GameObject platformPrefab;
9
           public GameObject finishLine;
10
           public int numberOfBlocks = 50;
           public float blockHeight = 1f;
12
           public int numberOfPlatformsInBlock = 3;
13
14
           public float levelWidth = 2.5f;
15
16
           public float offset = 0.25f;
17
18
           private List<Platform> platformList = new List<Platform>();
19
20
           public void Start()
22
23
               int maxCount = numberOfBlocks * numberOfPlatformsInBlock;
24
               Transform plaformsParent = new GameObject("Plaforms").transform;
25
               for(int i=0; i < maxCount; i++)</pre>
26
27
                   Platform platformTemp = Instantiate(platformPrefab, plaformsParent).GetComponent<Platform>();
28
                   platformList.Add(platformTemp);
29
                   platformTemp.SetActivation(false);
30
31
               print(plaformsParent.childCount.ToString());
32
```

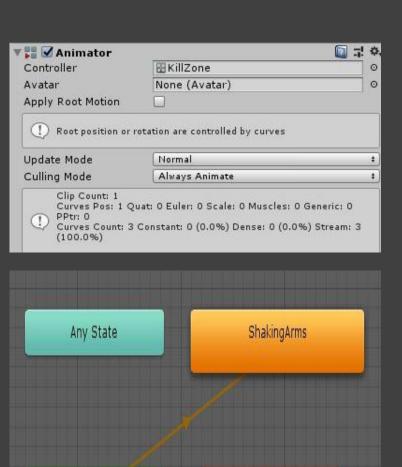
A H H H H H H H H

```
34
            public void LevelStart(int level)
35
36
               float blockHeightTemp = blockHeight + (level - 1) * 0.25f;
37
38
39
               for (int i = 0; i < numberOfBlocks; i++)
40
                    float minY = i * blockHeightTemp;
43
                    float maxY = (i + 1) * blockHeightTemp;
44
                    for (int j = 0; j < numberOfPlatformsInBlock; j++)</pre>
45
                       Vector3 spawnPosition = new Vector3();
48
                        spawnPosition.y = Random.Range(minY, maxY);
49
                        spawnPosition.x = Random.Range(-levelWidth, levelWidth);
50
                        int index = i * i;
51
                        platformList[index].SetPosition(spawnPosition);
52
                        platformList[index].SetActivation(true);
53
54
55
               float finishLineY = blockHeightTemp * numberOfBlocks;
56
                finishLine.transform.position = new Vector2(0, finishLineY);
57
58
```

Game Scene - Kill Zone

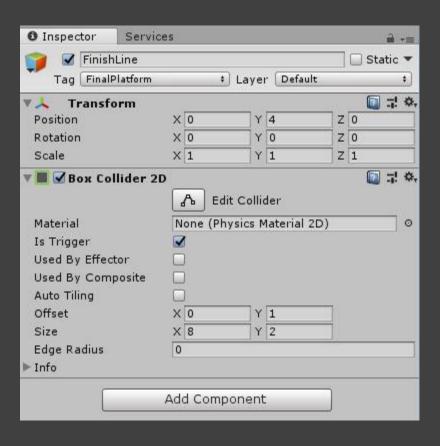






Game Scene - Finish Line

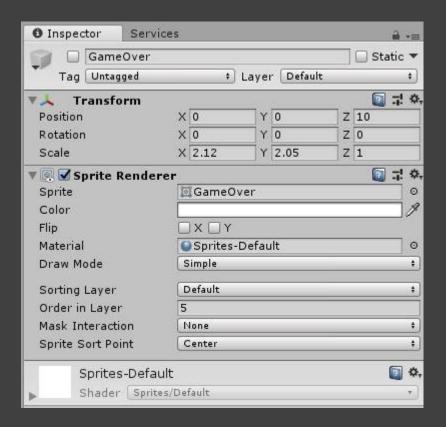
The character and and and and



Game Scene - Game Over



AMEND AND AND AND

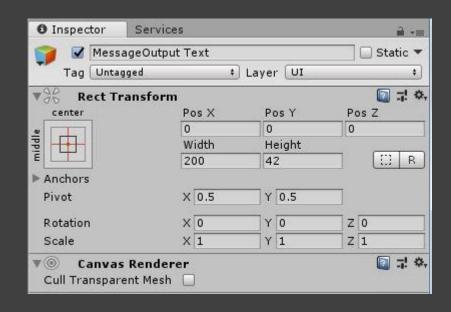


Game Scene - Score

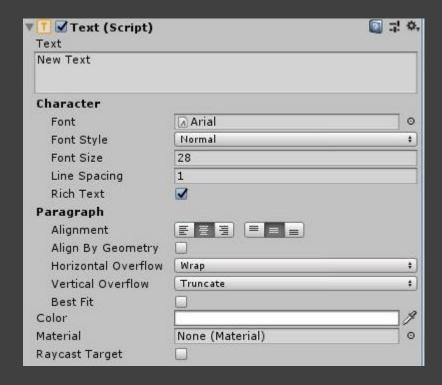
A HA HA HA HA HA HA

```
⊟using UnityEngine;
       using UnityEngine.UI;
      □public class Score : MonoBehaviour {
           public Transform player;
           public Text scoreText;
9
           public int lastScore = 0;
           public int currentSocre = 0;
           public void SetLastScore()
12
13
               lastScore = currentSocre;
               currentSocre = 0;
15
16
           // Update is called once per frame
           void Update () {
18
               currentSocre = lastScore + (int)player.position.y * 10;
19
               scoreText.text = currentSocre.ToString();
20
21
```

Game Scene - Message Output



A HA HA HA HA HA HA



Game Scene - Audio Manager

And and and and and and and

```
⊟using UnityEngine.Audio;
2
       using UnityEngine;
      ■public class AudioManager: MonoBehaviour {
           public Sound[] sounds;
 6
           // Use this for initialization
 8
           void Awake () {
               foreach (Sound's in sounds)
12
                   s.source = gameObject.AddComponent<AudioSource>();
13
                   s.source.clip = s.clip;
                   s.source.volume = s.vloume;
15
                   s.source.pitch = s.pitch;
17
18
19
           // Update is called once per frame
20
           void Update () {
23
```

Game Scene - Sound

```
⊟using UnityEngine.Audio;
       using UnityEngine;
       [System.Serializable]
     Epublic class Sound {
           public string name;
           public AudioClip clip;
10
           [Range(Of, 1f)]
           public float vloume;
13
           [Range(.1f, 3f)]
14
           public float pitch;
15
16
           [HideInInspector]
           public AudioSource source;
18
19
```

