

Forum:[Hack Forums](#)

Thread: [Need Help with my first Unity Game](#)

luuca - November 17, 2018, 7:59 am

Hello anons,

Over the last 2 days i tried to program my first Unity Game in C#.

I got one major problem.

If I jump on enemies it happens that either only I die.

me and the enemy dies.

Also the "You Won!"-Canvas i created doesnt appear if I hit the Exit.

Besides that i want to create a GUI and Multiplayer for the game so i can play with friends. If someone has a little knowledge of Unity and c# please leave a comment. ?

The Files can be found at:

[google.com/open?id=1bOR4Q63zP6NMgy6LNx4qm6z1S8VPVqrY](https://drive.google.com/open?id=1bOR4Q63zP6NMgy6LNx4qm6z1S8VPVqrY)[https://drive\[WARNING POSSIBLY INFECTED\]google.com/open?id=1bOR4Q63zP6NMgy6LNx4qm6z1S8VPVqrY](https://drive.google.com/open?id=1bOR4Q63zP6NMgy6LNx4qm6z1S8VPVqrY)

Thank you in advance!

lg luca

gigajew - November 17, 2018, 8:24 am

I suppose on the OnTouch event or whatever it is, you could have some code like (pseudocode)

```
if (player.Y <= monster.Height * 0.9) { // if players vertical position is higher than 0.9 of the
monsters height, you've successfully killed it
    monster.Die();
} else {
    player.Die();
}
```

luuca - November 17, 2018, 10:29 am

(Nov 17, 2018 09:24 AM)gigajew Wrote: [CITE](#) I suppose on the OnTouch event or whatever it is, you could have some code like (pseudocode)

```
if (player.Y <= monster.Height * 0.9) { // if players vertical position is higher than 0.9 of the
monsters height, you've successfully killed it
    monster.Die();
} else {
    player.Die();
}
```

Thank you for your answer i'll try to fix it.

But maybe you could look at the Source Code so you can see how i did it

Here are the two scripts in the game (c#):

PlayerController.cs

Code using System.Collections;

using System.Collections.Generic;

using UnityEngine;

using UnityEngine.SceneManagement;

```

public class PlayerController : MonoBehaviour {

    public SpriteRenderer sprite;
    public LayerMask groundLayer;
    public Transform groundCheck;
    public float groundCheckRadius;
    private bool groundCollision;
    private bool doubleJumped;
    public CanvasGroup canvasYouWin;
    private int lives = 3;

    void Start () {

    }

    void FixedUpdate () {
        groundCollision = Physics2D.OverlapCircle (groundCheck.position, groundCheckRadius,
        groundLayer);
    }

    void Update ()
    {
        var rigidBody = GetComponent<Rigidbody2D> ();
        var transform = GetComponent<Transform> ();
        if (Input.GetKey ("right")) {
            sprite.flipX = false;
            if (Input.GetKey ("left shift"))
                rigidBody.velocity = new Vector2 (10, rigidBody.velocity.y);
            else
                rigidBody.velocity = new Vector2 (5, rigidBody.velocity.y);
        }

        if (Input.GetKey ("left")) {
            sprite.flipX = true;
            if (Input.GetKey ("left shift"))
                rigidBody.velocity = new Vector2 (-10, rigidBody.velocity.y);
            else
                rigidBody.velocity = new Vector2 (-5, rigidBody.velocity.y);
        }

        if (groundCollision)
            doubleJumped = false;
        if (Input.GetKeyDown ("space") && groundCollision)
            rigidBody.velocity = new Vector2 (rigidBody.velocity.x, 10);
        if (Input.GetKeyDown ("space") && !groundCollision && !doubleJumped) {
            rigidBody.velocity = new Vector2 (rigidBody.velocity.x, 10);
            doubleJumped = true;
        }
        if (transform.position.y < -6) {
            lives -= 1;
        }
    }
}

```

```

    if (transform.position.x < 2)
        transform.position = new Vector2 (-5, 2);
    else
        transform.position = new Vector2 (2, 2);
}

if (lives == 0)
    SceneManager.LoadScene(SceneManager.GetActiveScene().name);
}

public void OnTriggerEnter2D (Collider2D other) {
    if (other.name == "EnemyDamage") {
        lives -= 1;
        transform.position = new Vector2 (2, 1);
    } else if (other.name == "exit") {
        canvasYouWin.alpha = 1;
    }
}
}
}

```

EnemyController.cs

Code using System.Collections;
using System.Collections.Generic;
using UnityEngine;

```

public class EnemyController : MonoBehaviour {

    public bool moveLeft;
    public LayerMask groundLayer;
    public Transform wallCheck;
    public float wallCheckRadius;
    private bool wallCollision;
    private bool notAtEdge;
    public Transform edgeCheck;

    void Start () {
        moveLeft = true;
    }

    void FixedUpdate () {
        wallCollision = Physics2D.OverlapCircle (wallCheck.position, wallCheckRadius, groundLayer);
        notAtEdge = Physics2D.OverlapCircle (edgeCheck.position, wallCheckRadius, groundLayer);
    }

    void Update ()
    {
        var rigidBody = GetComponent<Rigidbody2D> ();

        if (wallCollision || !notAtEdge) {
            moveLeft = !moveLeft;
        }

        if (moveLeft) {
            transform.localScale = new Vector2 (1, 1);

```

```

        rigidBody.velocity = new Vector2 (-5, rigidBody.velocity.y);
    } else {
        transform.localScale = new Vector2 (-1, 1);
        rigidBody.velocity = new Vector2 (5, rigidBody.velocity.y);
    }
}

void OnTriggerEnter2D (Collider2D other) {
    if (other.name == "HeadStomper") {
        Destroy (gameObject);
    }
}
}
}

```

0xBB - November 17, 2018, 4:28 pm

(Nov 17, 2018 09:24 AM)gigajew Wrote: [CITE](#) I suppose on the OnTouch event or whatever it is, you could have some code like (pseudocode)

```

if (player.Y <= monster.Height * 0.9) { // if players vertical position is higher than 0.9 of the
monsters height, you've successfully killed it
    monster.Die();
} else {
    player.Die();
}

```

That is very bad .. @op look up [raycast](#) in unity to detect where the objects intersect and apply your logic here.