

Game Tutorial

FlappyBird



Oluwaseye Ayinla

Certified Developer

Download Link



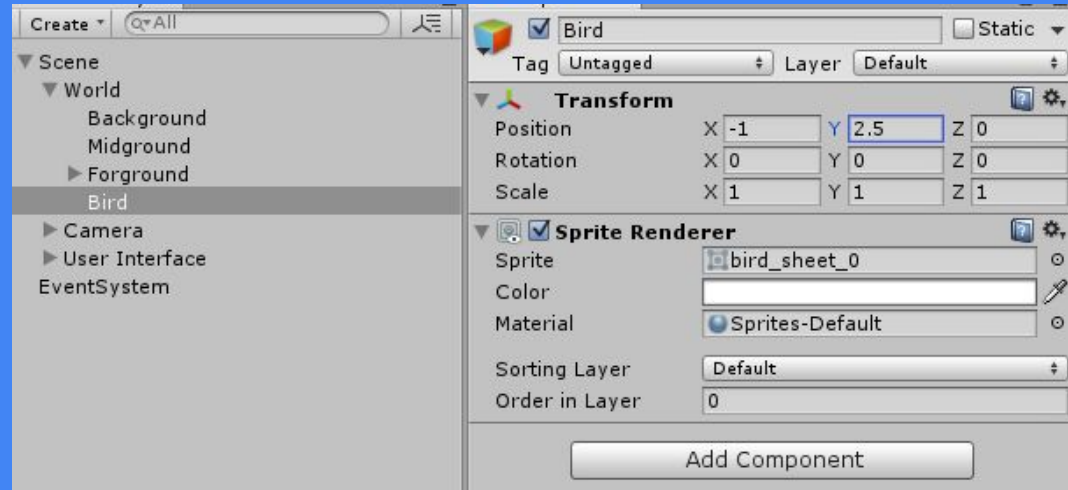
<https://github.com/oluwaseyeayinla/GGJ17-Flappy-Bird-Tutorial>

Setting up the bird

Drag the 'bird_sheet_0' into the World gameobject

Rename the new object to Bird

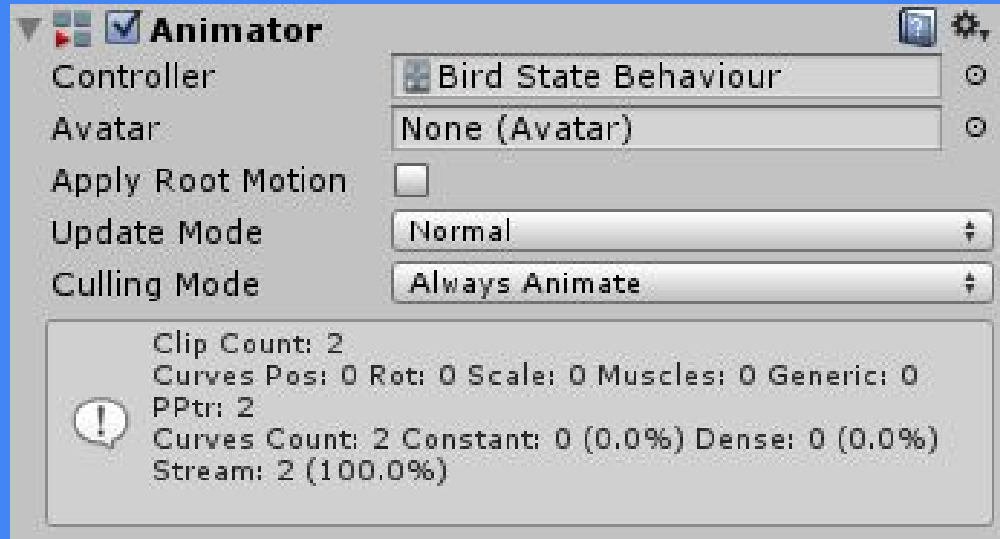
Position the transform of the Bird gameobject as in the picture



Bird animation

Add an animator component to the bird gameobject

Find the Bird State Behaviour and assign it as the controller and test out the flapping animation



Bird physics & controller

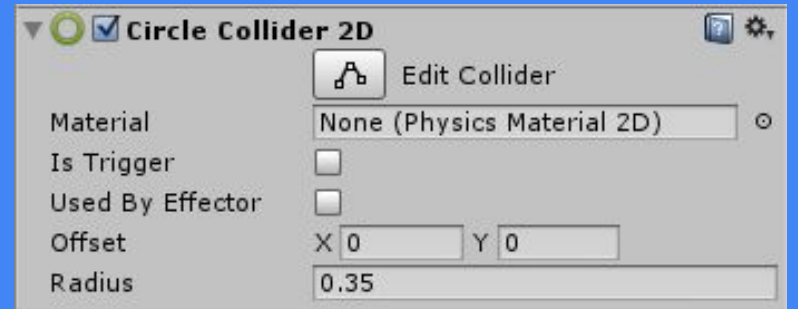
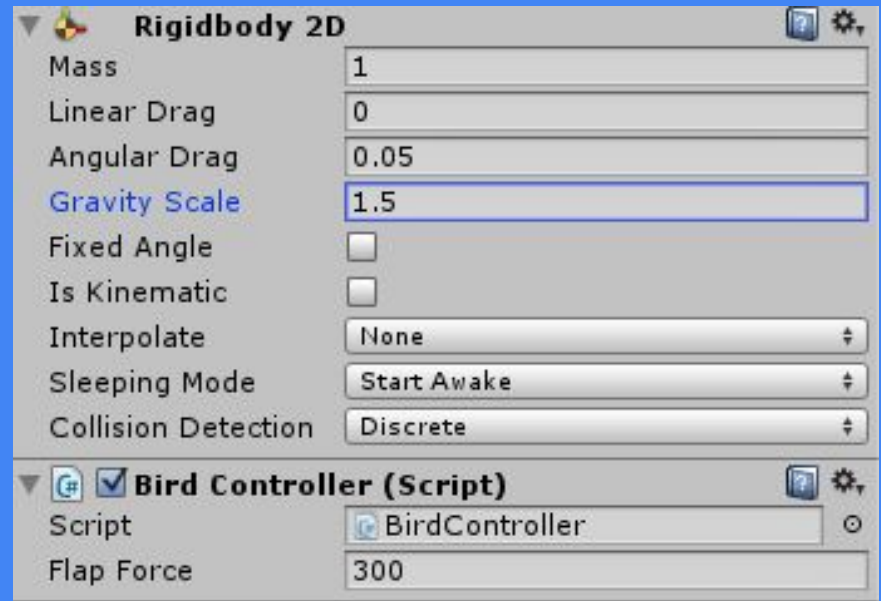
Add a rigidbody2D component to the bird gameobject and test

Add the bird controller script to the bird and set its flap force to 300

Then set its gravity scale to 1.5

Test out the flap

Add a circle collider 2d to the bird gameobject and set the radius to 0.35



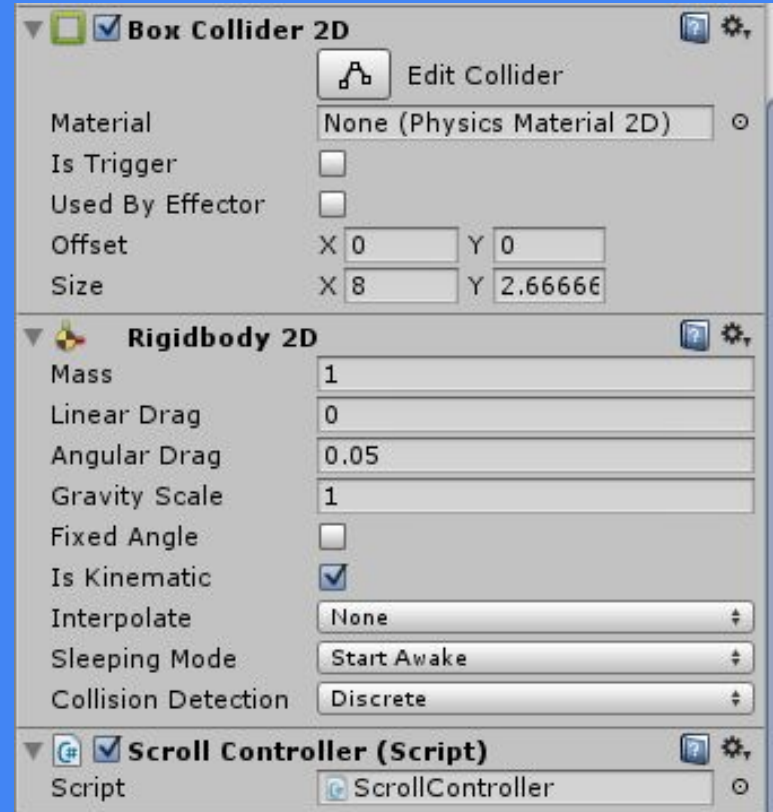
Scrolling the ground

Add a box collider 2d to the ground gameobject in the foreground and test out the collision

Add a rigidbody 2D to the ground and set it to kinematic

Attach the scroll controller to the ground gameobject and test

Fix gap by duplicating the ground and shifting the x axis of the duplicated ground game object by 8 units; which is the same length as the size of the box collider attached to it.



Scrolling the ground contd.

Add the repeating controller script to both ground objects and watch the magic happen

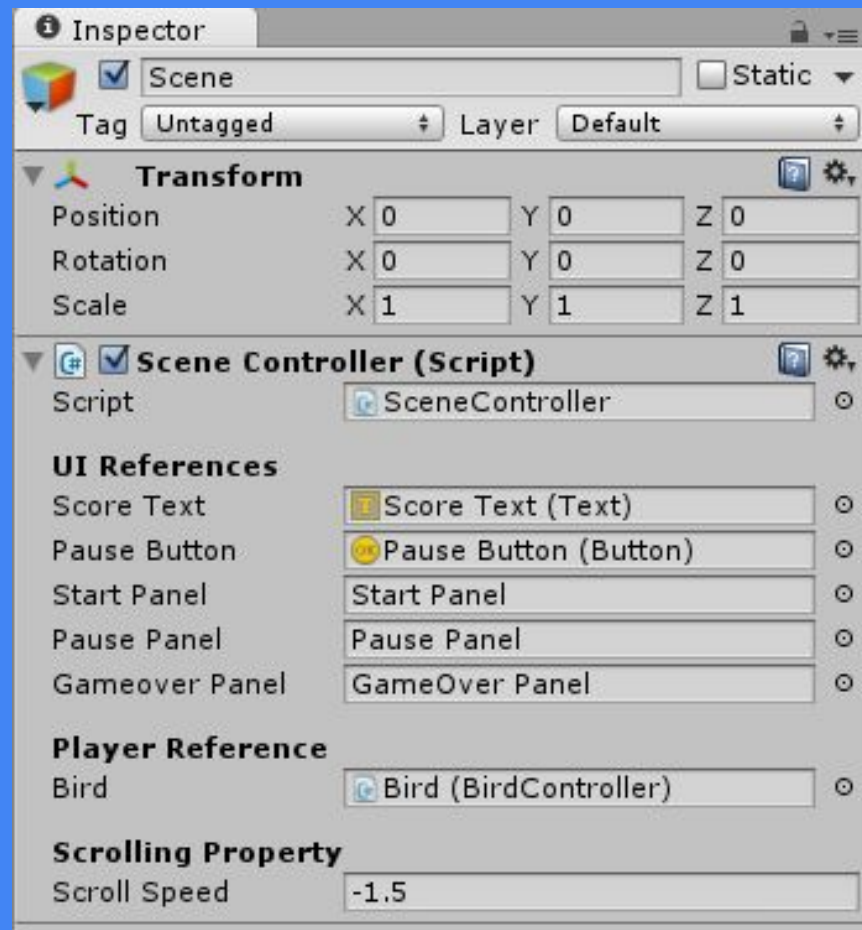
Add a physics material of solid ground in the material section of the box colliders of both ground objects



Dealing with the UI

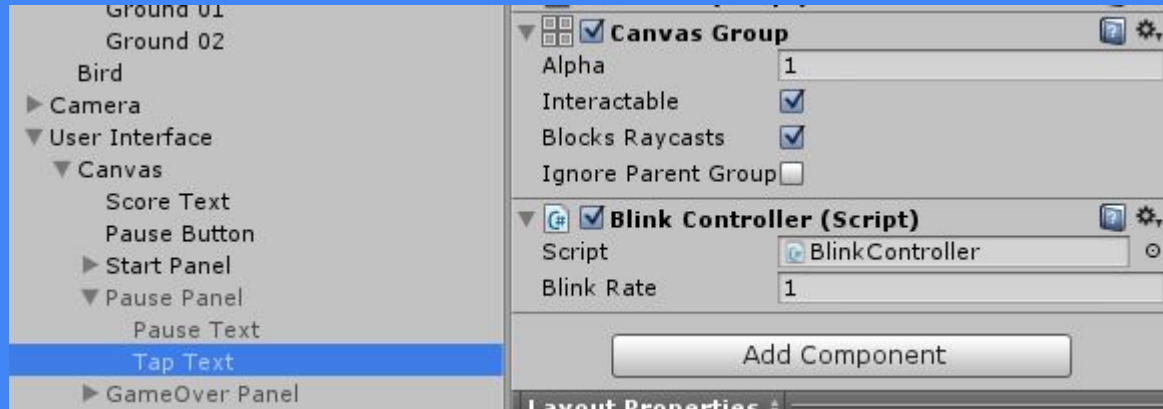
Add the scene controller to the scene gameobject and assign all the references seen in the inspector

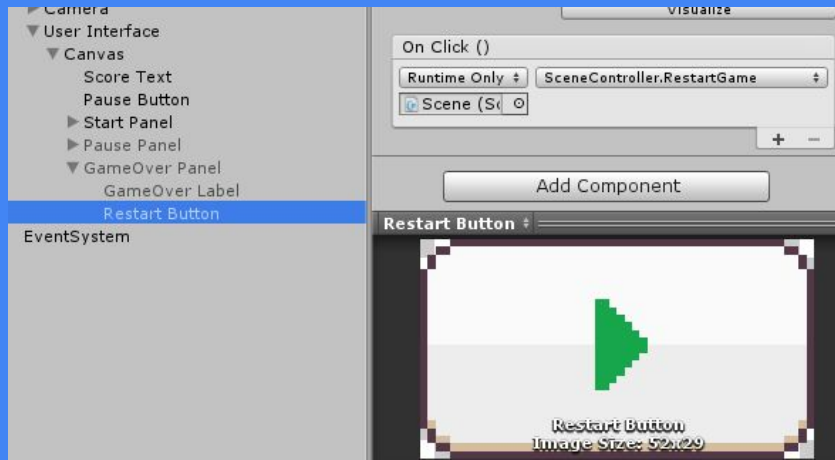
Set the scroll speed to -1.5



Dealing with the UI contd.

Attach the blink controller to the tap text of the pause panel and test out game

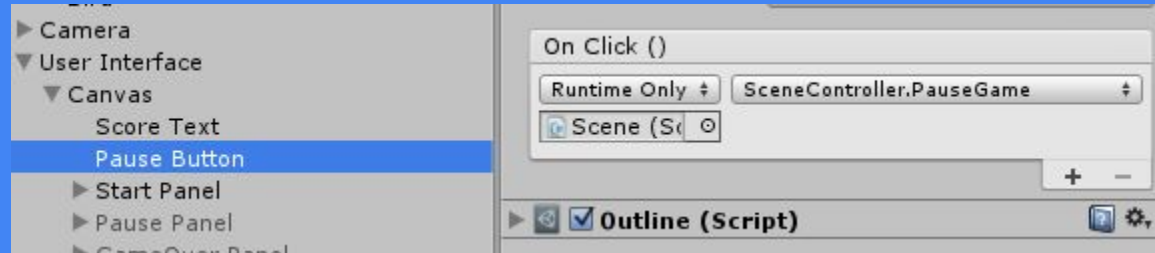




Set the OnClick button function of the restart button to restart game in scene controller

Play test when done

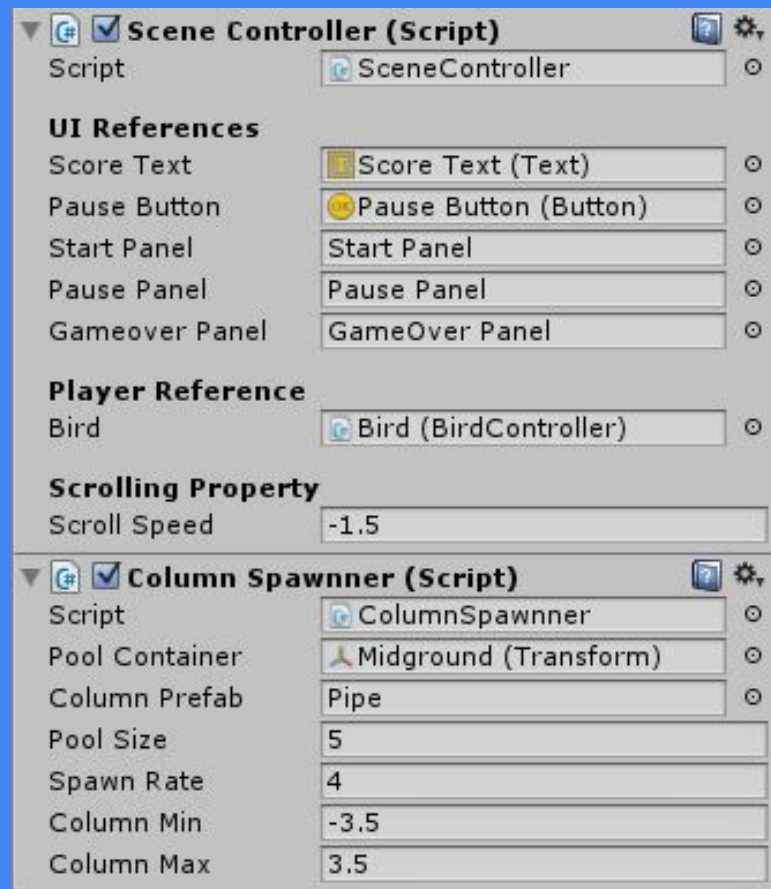
Set the OnClick button function of the pause button to pause game in scene controller



Spawn pipes

Attach the column spanner to the scene gameobject and set the column -3.5

Also assign the references seen in the inspector

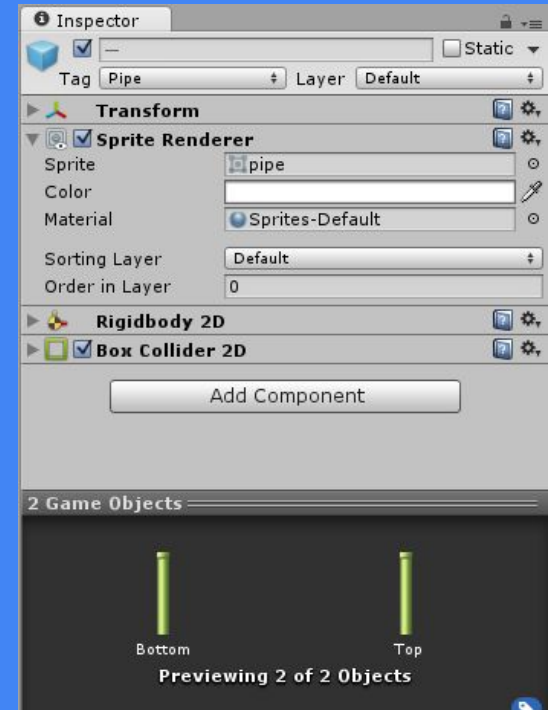
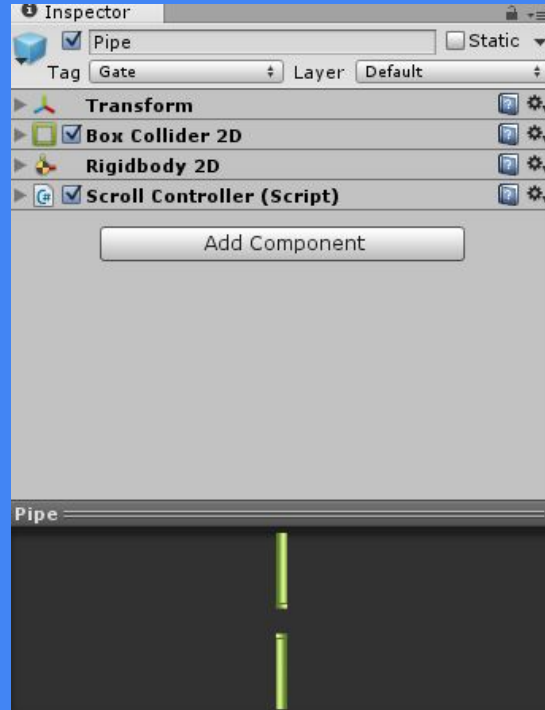
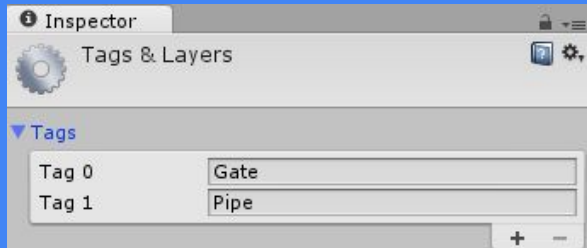


Add 2 tags in the tags layer; Gate and Pipe

Assign the Gate tag to the Pipe (parent) gameobject and assign the Pipe tag to the Top and Bottom (children) gameobject and test the collision

Gate

Pipe



Polish

A few things are still wrong

- We don't want the bird to be flapping or flying even after game over
- There is no sound so the game isn't fun
- The bird appears behind the pipes
- And the bird is not rotating as you flap in the air.

Adding sound

```
// Update is called once per frame
void Update()
{
    if (SceneController.Instance.IsGameRunning && isAlive)
    {
        // Flap
        if (Input.GetKeyDown(KeyCode.Space) || Input.GetMouseButtonDown(0))
        {
            rigidbody2DComponent.velocity = Vector2.zero;
            rigidbody2DComponent.AddForce(Vector2.up * flapForce);
            AudioSource.PlayClipAtPoint(flapClip, Vector3.zero);
        }
    }
}
```

Don't forget to assign the proper sound files in the sounds folder to the bird controller inspector references

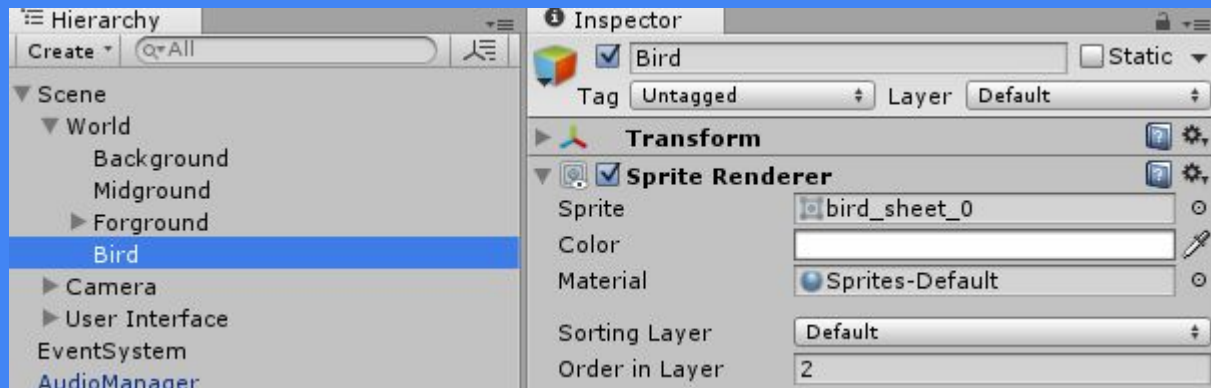
Uncomment the code in the update method of the bird controller to fix the flapping when dead

Also uncomment any AudioSource code you see elsewhere in the script and play test



Bring bird to the front of pipes

Set the order layer of the bird game object to be a value of 2 or higher



Rotate head of bird

Uncomment the fixed update code in the bird controller to perform the intended rotation

```
/// <summary>
/// When the bird goes up, it'll rotate up to 45 degrees. when it falls, rotation will be -90 degrees min
/// </summary>
void FixedUpdate()
{
    if (SceneController.Instance.IsGameRunning || SceneController.Instance.IsGameOver)
    {
        RotateBird();
    }
}
```


Thanks!

Contact me:

Github: [oluwaseyeayinla](#)

Facebook: [oluwaseyeayinla](#)

oluwaseyeayinla@gmail.com

www.about.me/oluwaseyeayinla

