Part 5: Unity Flappy Bird Tutorial

Contents

The User Interface

Unity uses a Canvas game object to display information to the user.

1. **Hierarchy** → **Right-click** → **Create** → **UI** → **Canvas** This also creates an **EventSystem** game object.
2. In the Inspector → Set the following properties:
    a. **Render Mode:** Screen Space – Camera
    b. Drag the **Main Camera** to **Render Camera.**
    c. **Order in Layer:** 10
    d. **UI Scale Mode:** Scale With Screen Size
    e. **Reference Resolution:** X: 900 Y: 1600
    f. **Match:** 0.5

Score Display

1. Right-click the **Canvas** game object → **Create** → **UI** → **Text**
2. Select the **Move** tool → move the text toward the top of the Scene
3. **Inspector** → Set the following properties

a. **Name:** TxtScore

b. **Text:** 0

c. **Font Size:** 150

d. **Paragraph Alignment:** Center Center

e. **Horizontal Overflow:** Overflow

f. **Vertical Overflow:** Overflow

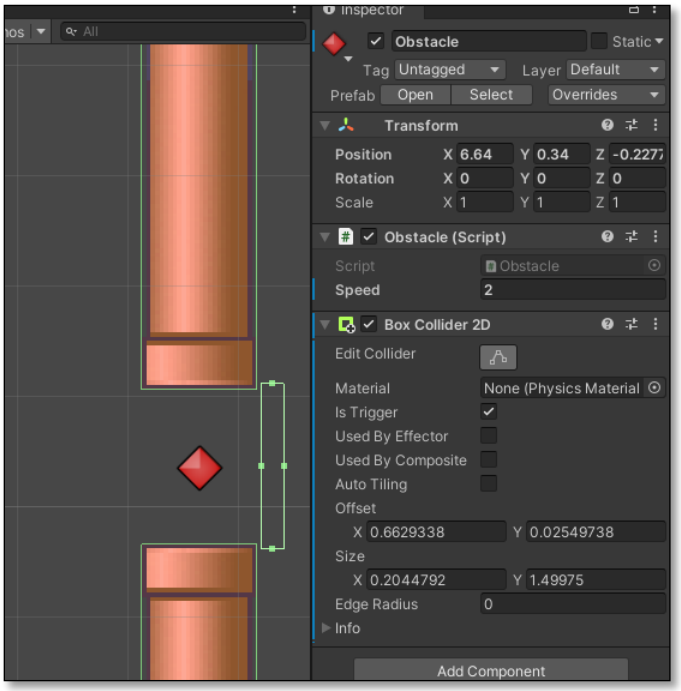
g. **Color:** White R:255 G:255 B:255

4. **Add Component → Shadow → Effect Distance → X:** 7 **Y:** -7

Add Collider to Obstacle

We want to keep track of each time the bird makes it successfully past the pipes. We will use a Box Collider 2D to Trigger a Score event. In the image below, notice the collider is just past the pipes.

1. In the **Hierarchy → Obstacle → Add Component → Box Collider 2D**
2. **Is Trigger:** Enabled
3. Click **Edit Collider** → Reshape the collider as shown below.

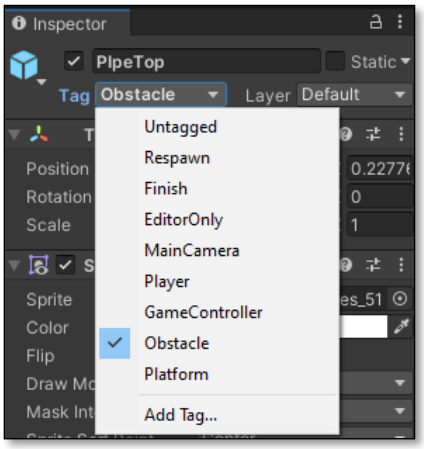


4. In the **Inspector** → **Overrides** → **Apply All**

Appy Tags

As part of our collision detection, we want to tag the Game Objects.

1. **Hierarchy** → Select **Platform** → **Inspector**
2. Click **Tag** → **Add Tag** → Click the **+** sign → **New Tag Name**: Obstacle
3. Use the same method to add an **Obstacle** tag to **PipeTop** and **PipeBottom**.
4. Use the same method to add a **Score** tag to **Obstacle**.



Bird Script

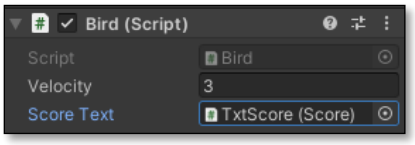
1. **Assets** → **Scripts** → Double-click **BirdScript**
2. Add the following code to the script.

```
using UnityEngine;

Unity Script (1 asset reference) | 0 references
public class Bird : MonoBehaviour
{
    // Variable to store the Bird vertical velocity
    public float velocity;

    // Create a reference to the Rigidbody2D class
    private Rigidbody2D rb;
    // Create reference to Score script object
    public Score scoreText;
```

3. Add a new method at the end of the script.

```
Unity Message | 0 references
32      private void OnTriggerEnter2D(Collider2D collision)
33      {
34          // If the Bird collides with the Obstacle
35          if (collision.CompareTag("Obstacle"))
36          {
37              // Call the ScoreUp method to display the new score
38              scoreText.ScoreUp();
39          }
40      }
41
        Unity Message | 0 references
42      private void OnCollisionEnter2D(Collision2D collision)
43      {
44          // If the bird collides with the Platform or the Obstacle
45          if (collision.gameObject.CompareTag("Platform") ||
46                  collision.gameObject.CompareTag("Obstacle"))
47          {
48              // Game over, freeze the game
49              Time.timeScale = 0;
50          }
51      }
52  }
```

4. Save the script.
5. **Hierarchy** → **Bird** → **Bird (Script)** → **Score Text**→ Click to the right on the circle → Select **TxtScore**



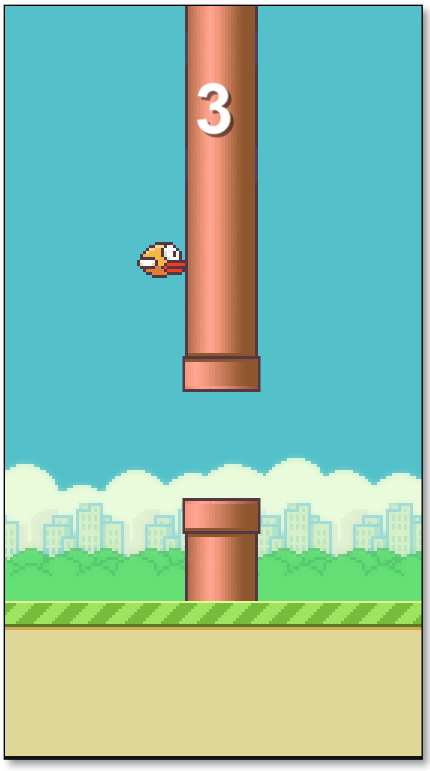
Score Script

1. In the **Scripts** folder → Create a new C# script named **Score**
2. Double-click to Edit the script.
3. Don't forget to add **using UnityEngine.UI** as shown.

```
using UnityEngine;
using UnityEngine.UI;
Unity Script | 1 reference
public class Score : MonoBehaviour
{
    // Create a variable to store the score
    static int score;
    // Start is called before the first frame update
    Unity Message | 0 references
    void Start()
    {
        // Set score to 0 at the start of the game
        score = 0;
    }

    // Update is called once per frame
    1 reference
    public void ScoreUp()
    {
        // Increment score each time the ScoreUp method is called
        score++;
        // Display new score on the Text property of the component
        GetComponent<Text>().text = score.ToString();
    }
}
```

4. Save the script.
5. **Hierarchy** → **TxtScore** → **Add Component** → Search for and add the **Score** script.

Play the game. The Score should update each time the bird successfully passes a pipe as show below. The game should freeze whenever the bird hits the Platform or a pipe.



Assignment Submission

A Unity project is at least 200 MB. That is too big to be submitted.

Please attach a screenshot of your project to the assignment in Blackboard.