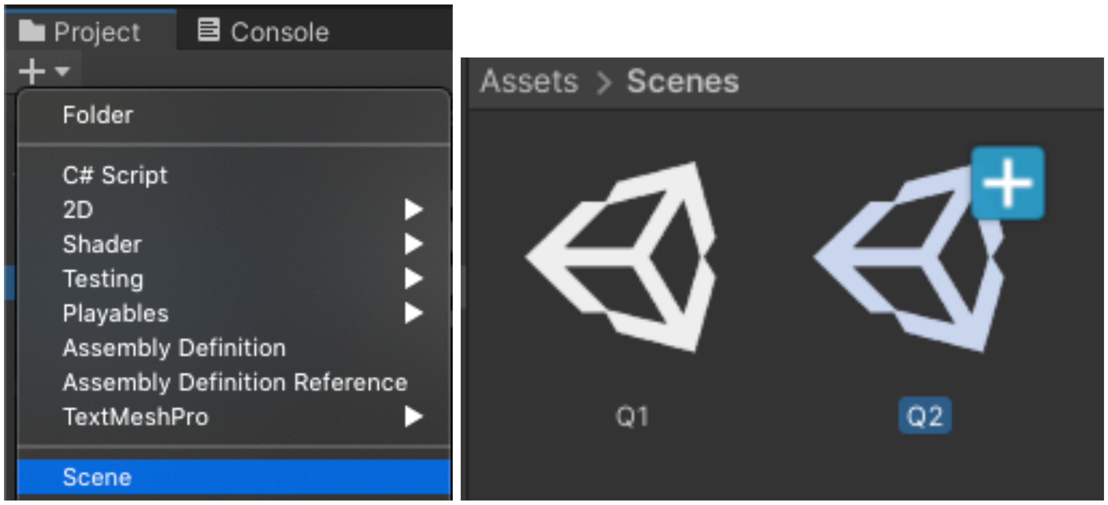
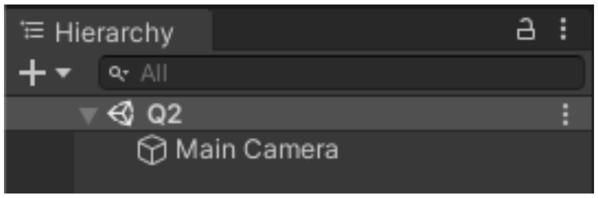
Quest 2 - Steps

1) Create Scene

Using the same Project as before, create Q2 in the Scenes folder for this exercise.

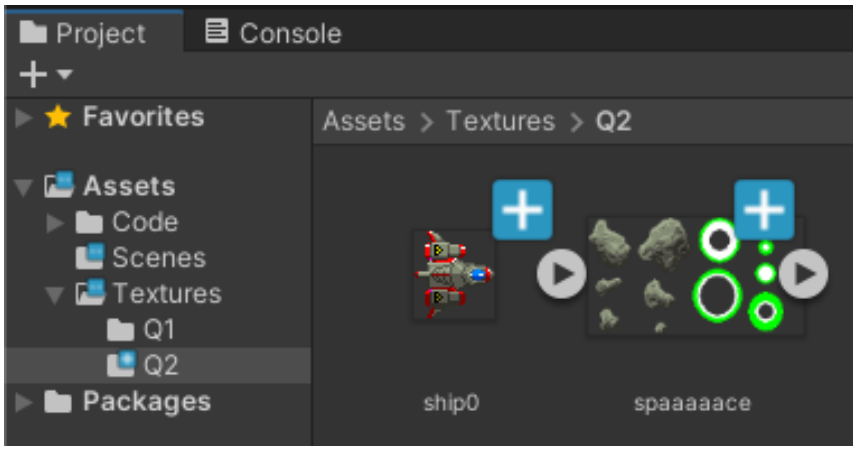


Double-click Q2 and check the Hierarchy to make sure you are working on the expected scene.



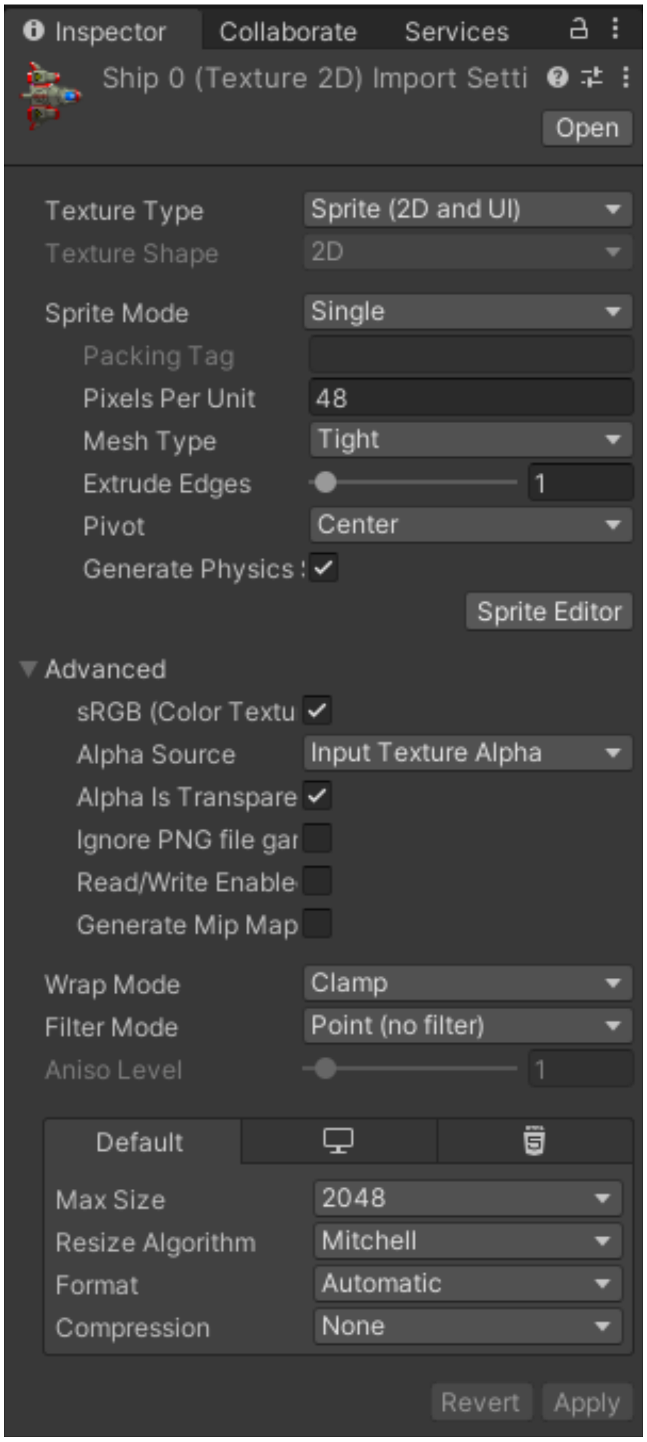
2) Setup Folders and Import Files

Import the supplied Q2 graphics into the /Assets/Textures/Q2/ folder.



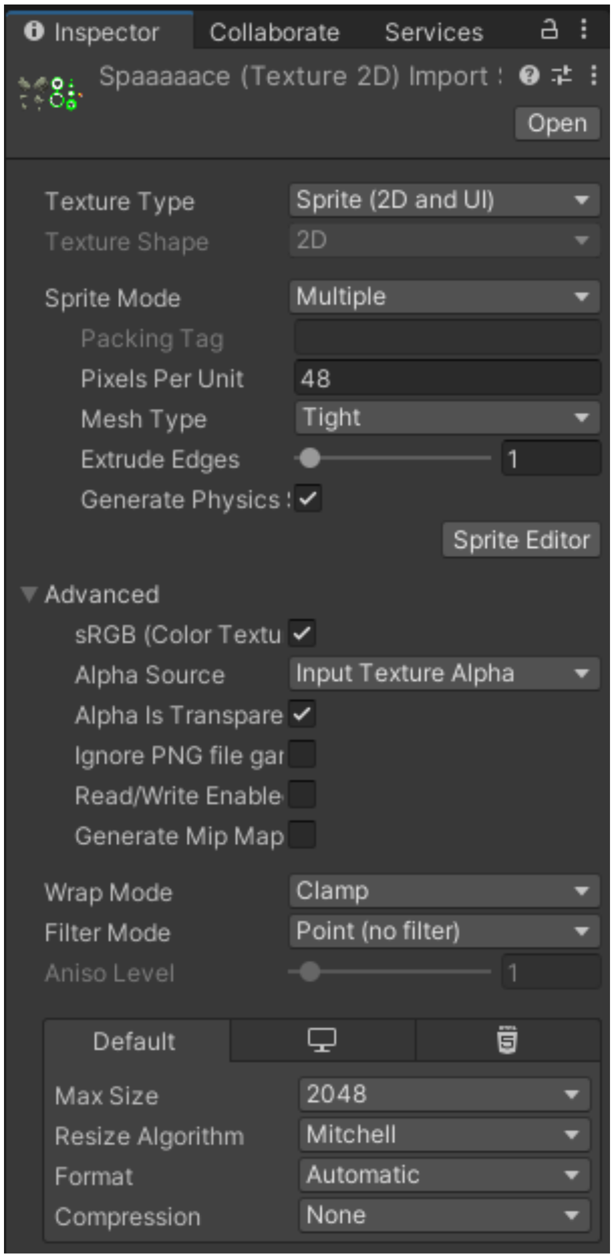
3) Import Settings for Ship Sprite

Select the ship0 sprite in the Project tab and configure it in the Inspector tab. Change the PPU to 48 so that the graphic is properly sized. The Filter Mode should be Point because the graphics are pixel art. Set Compression to None to maximize graphics quality.



Texture Type	Sprite (2D and UI)
Texture Shape	2D
Sprite Mode	Single
Packing Tag	
Pixels Per Unit	48
Mesh Type	Tight
Extrude Edges	1
Pivot	Center
Generate Physics :	✓
sRGB (Color Textu	✓
Alpha Source	Input Texture Alpha
Alpha Is Transpare	✓
Ignore PNG file gar	
Read/Write Enable	
Generate Mip Map	
Wrap Mode	Clamp
Filter Mode	Point (no filter)
Aniso Level	1
Max Size	2048
Resize Algorithm	Mitchell
Format	Automatic
Compression	None

4) Import Settings for Obstacles and Goals Sprites



Texture Type: Sprite (2D and UI)
Texture Shape: 2D
Sprite Mode: Multiple
Packing Tag:
Pixels Per Unit: 48
Mesh Type: Tight
Extrude Edges: 1
Generate Physics : ✓
Advanced
sRGB (Color Textu ✓
Alpha Source: Input Texture Alpha
Alpha Is Transpare ✓
Ignore PNG file gar
Read/Write Enable
Generate Mip Map
Wrap Mode: Clamp
Filter Mode: Point (no filter)
Aniso Level: 1
Max Size: 2048
Resize Algorithm: Mitchell
Format: Automatic
Compression: None

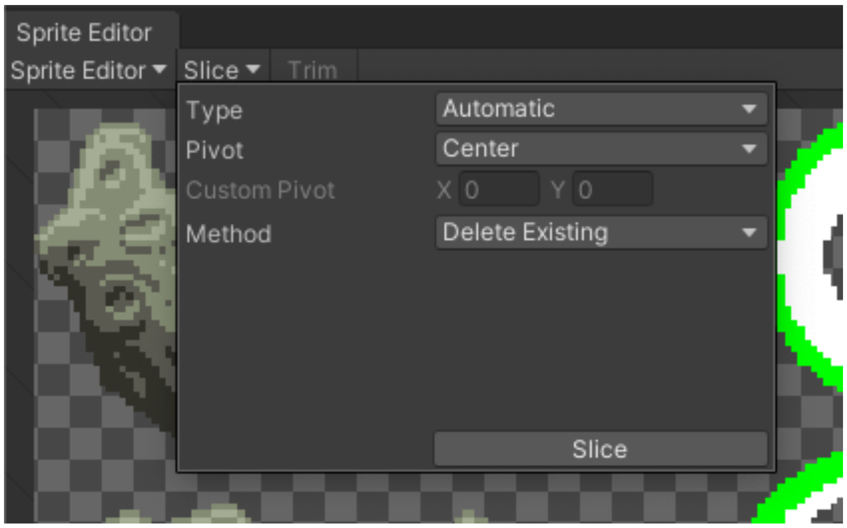
5) Multi-slice Sprites for Obstacles and Goals

Check all the settings. Be sure Sprite Mode is set to "Multiple."

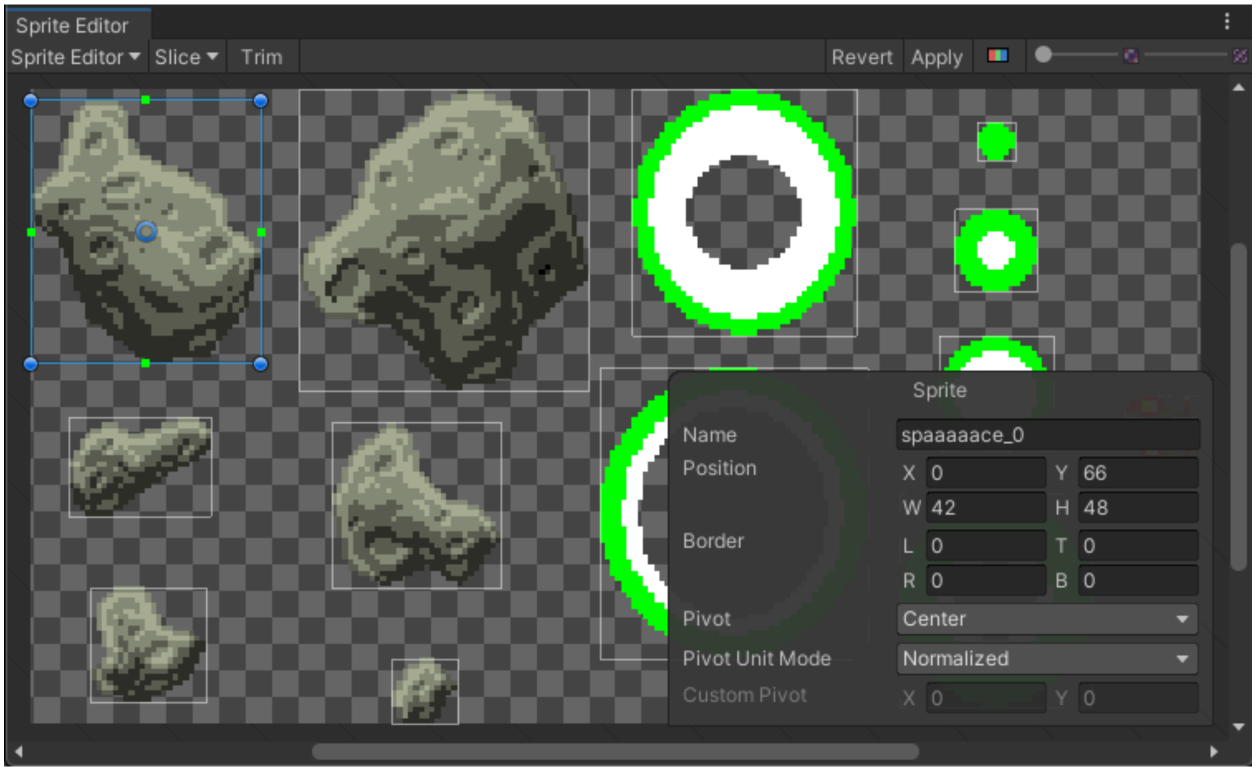
Click "Sprite Editor" in the Inspector tab to open the Sprite Editor tool.

Click "Slice," set Type to Automatic, and Click the Splice button.

If you click an individual graphic, you will notice it is sliced as its own sprite.



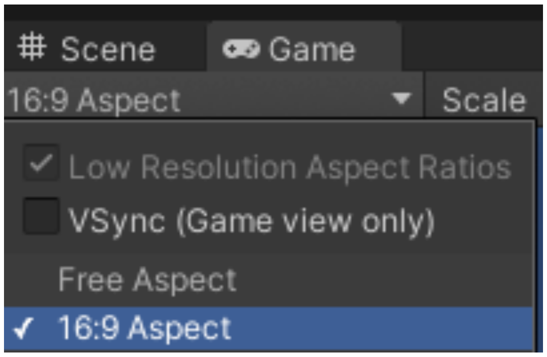
Type: Automatic
Pivot: Center
Custom Pivot: X 0 Y 0
Method: Delete Existing



Name: spaaaaace_0
Position: X 0 Y 66 W 42 H 48
Border: L 0 T 0 R 0 B 0
Pivot: Center
Pivot Unit Mode: Normalized
Custom Pivot: X 0 Y 0

6) Set your game’s Aspect Ratio to 16:9

This helps ensure consistent screen boundaries across testing devices.



Scene | Game

16:9 Aspect ▾ | Scale

✓ Low Resolution Aspect Ratios

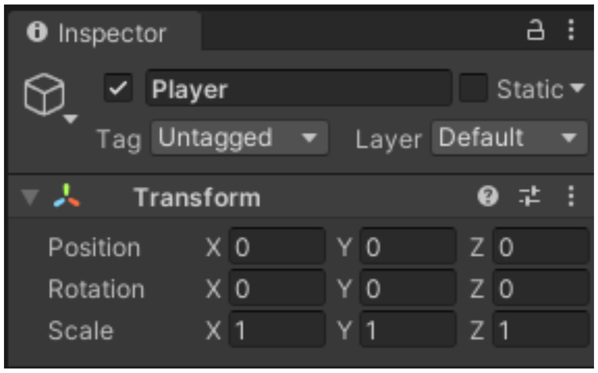
☐ VSync (Game view only)

Free Aspect

✓ 16:9 Aspect

7) Create a Player GameObject and Components

The order components are added is important. The collider knows what shape to take based on the sprite that is assigned in the Sprite Renderer.



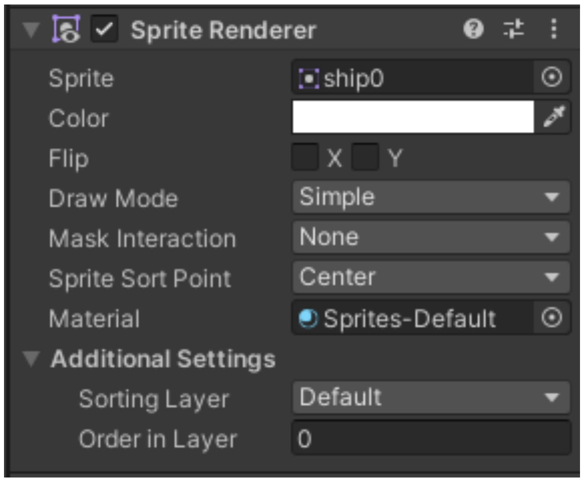
Inspector

✓ Player ☐ Static

Tag Untagged · Layer Default

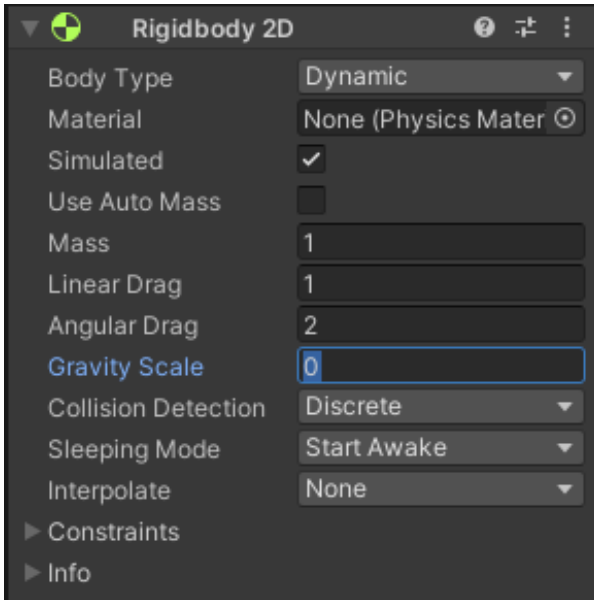
Transform

Position	X 0	Y 0	Z 0
Rotation	X 0	Y 0	Z 0
Scale	X 1	Y 1	Z 1



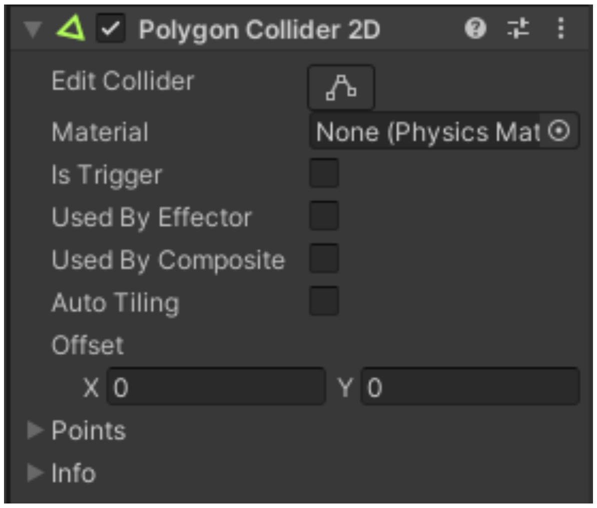
✓ **Sprite Renderer**

Sprite	ship0
Color	⬜
Flip	☐ X ☐ Y
Draw Mode	Simple
Mask Interaction	None
Sprite Sort Point	Center
Material	Sprites-Default
Additional Settings	
Sorting Layer	Default
Order in Layer	0



Rigidbody 2D

Body Type	Dynamic
Material	None (Physics Mater
Simulated	✓
Use Auto Mass	☐
Mass	1
Linear Drag	1
Angular Drag	2
Gravity Scale	0
Collision Detection	Discrete
Sleeping Mode	Start Awake
Interpolate	None
▶ Constraints	
▶ Info	

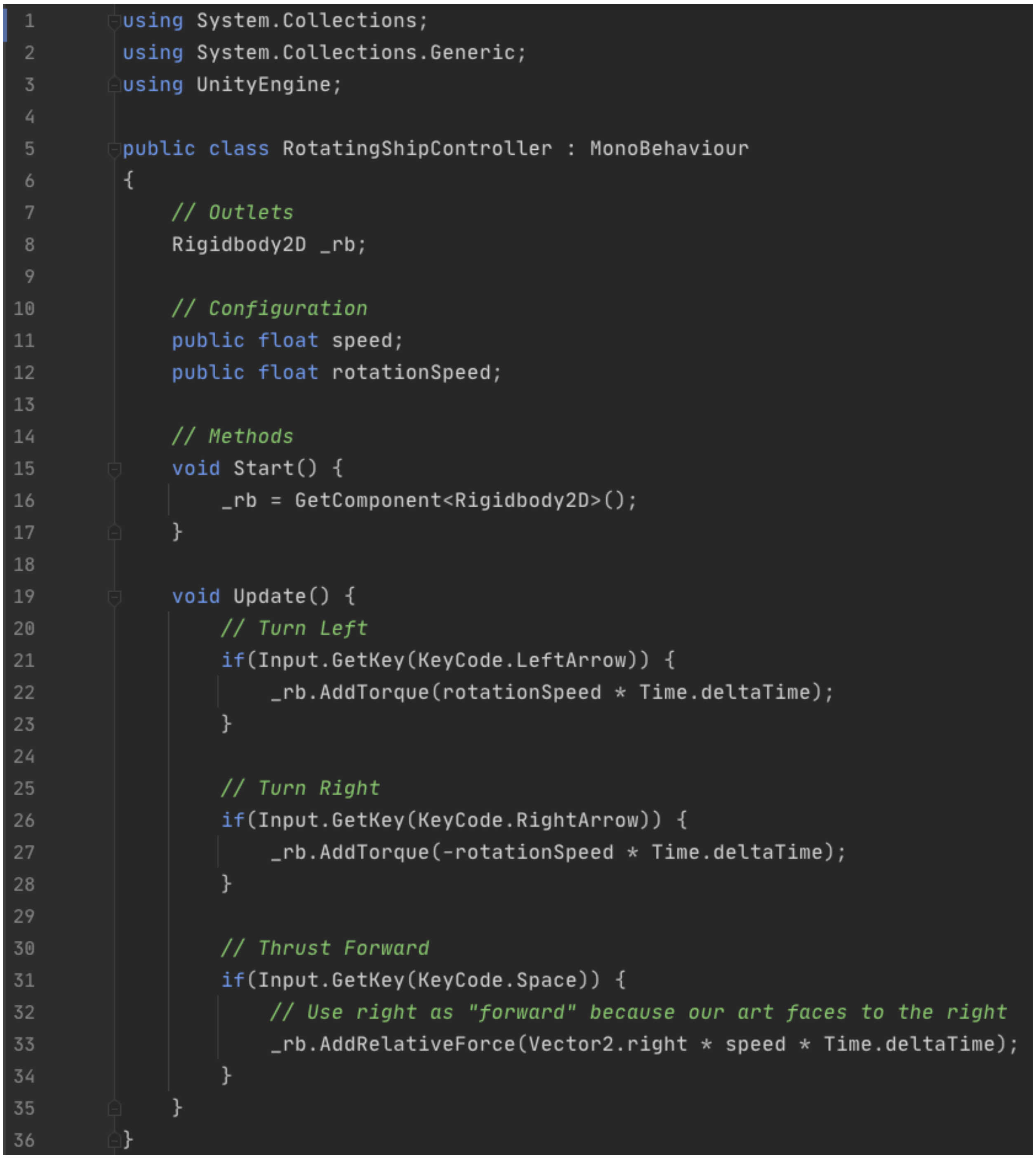


✓ **Polygon Collider 2D**

Edit Collider	
Material	None (Physics Mat
Is Trigger	☐
Used By Effector	☐
Used By Composite	☐
Auto Tiling	☐
Offset	X 0 Y 0
▶ Points	
▶ Info	

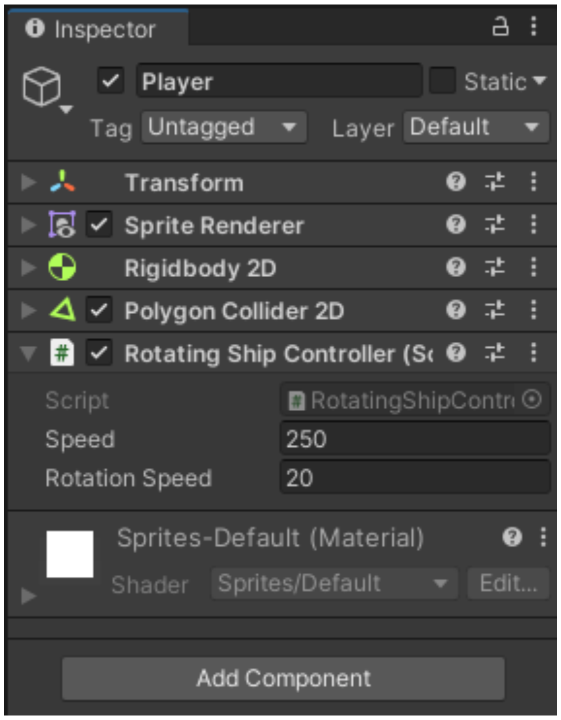
8) Create a Player Script

Create RotatingShipController.cs in /Assets/Code/Q2/



```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class RotatingShipController : MonoBehaviour
{
    // Outlets
    Rigidbody2D _rb;

    // Configuration
    public float speed;
    public float rotationSpeed;

    // Methods
    void Start() {
        _rb = GetComponent<Rigidbody2D>();
    }

    void Update() {
        // Turn Left
        if(Input.GetKey(KeyCode.LeftArrow)) {
            _rb.AddTorque(rotationSpeed * Time.deltaTime);
        }

        // Turn Right
        if(Input.GetKey(KeyCode.RightArrow)) {
            _rb.AddTorque(-rotationSpeed * Time.deltaTime);
        }

        // Thrust Forward
        if(Input.GetKey(KeyCode.Space)) {
            // Use right as "forward" because our art faces to the right
            _rb.AddRelativeForce(Vector2.right * speed * Time.deltaTime);
        }
    }
}
```

9) Attach and Configure the Script


Inspector

Player | Static
Tag Untagged | Layer Default

Transform
Sprite Renderer
Rigidbody 2D
Polygon Collider 2D
Rotating Ship Controller (Sc

Script	RotatingShipContr
Speed	250
Rotation Speed	20

Sprites-Default (Material)
Shader Sprites/Default | Edit...

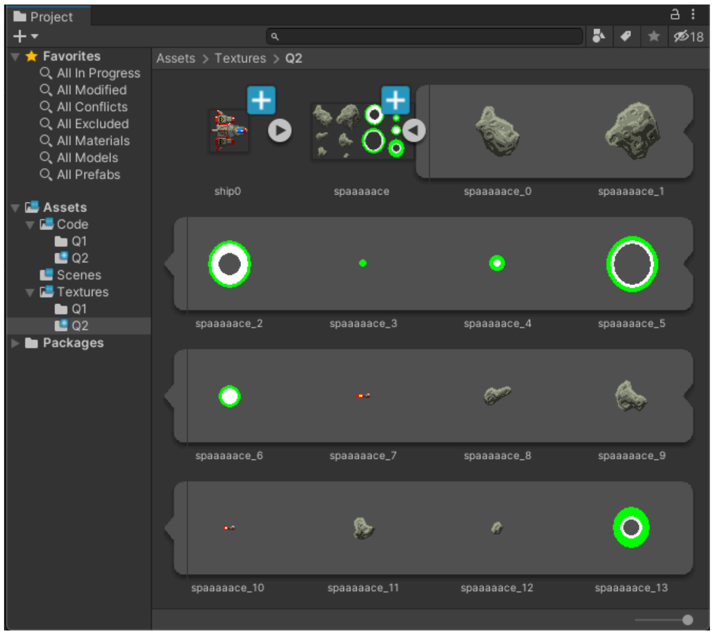
Add Component

10) PlayTest for “Game Feel”

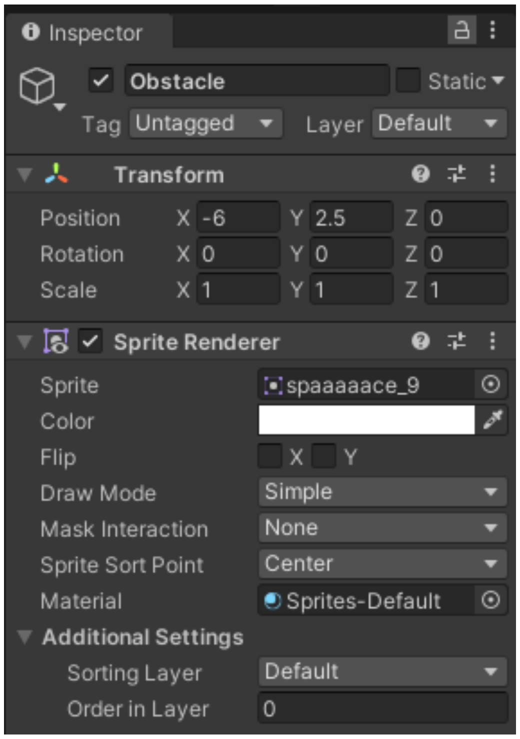
Adjust Rigidbody 2D’s Linear Drag and Angular Drag as well as PlayerController’s Speed and Rotation Speed for an appropriate “Game Feel.”

11) Create an Obstacle GameObject

Sprite sheets can be expanded to reveal individual sprites.



Create an Obstacle GameObject



Inspector

Obstacle — Static — Tag: Untagged — Layer: Default

Transform

	X	Y	Z
Position	-6	2.5	0
Rotation	0	0	0
Scale	1	1	1

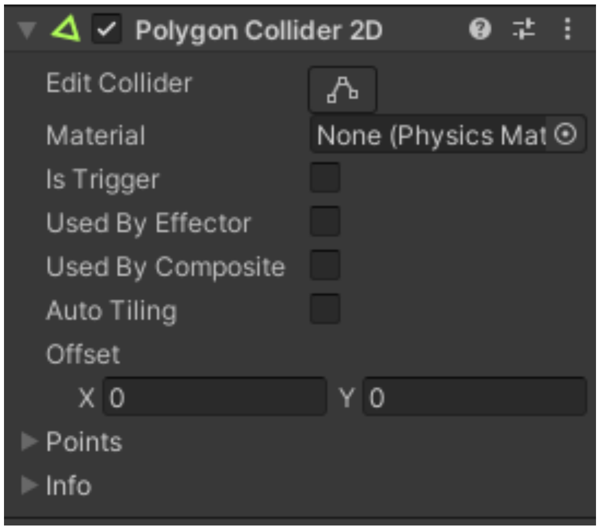
Sprite Renderer

Sprite	spaaaaace_9
Color	
Flip	X Y
Draw Mode	Simple
Mask Interaction	None
Sprite Sort Point	Center
Material	Sprites-Default

Additional Settings

Sorting Layer	Default
Order in Layer	0

Set the Sprite before adding the PolygonCollider, so the engine knows what shape to generate.

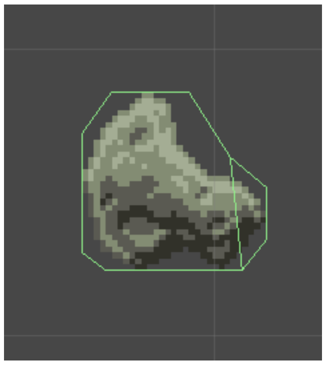


Polygon Collider 2D

Edit Collider	
Material	None (Physics Mat
Is Trigger	
Used By Effector	
Used By Composite	
Auto Tiling	
Offset	X 0 Y 0

Points

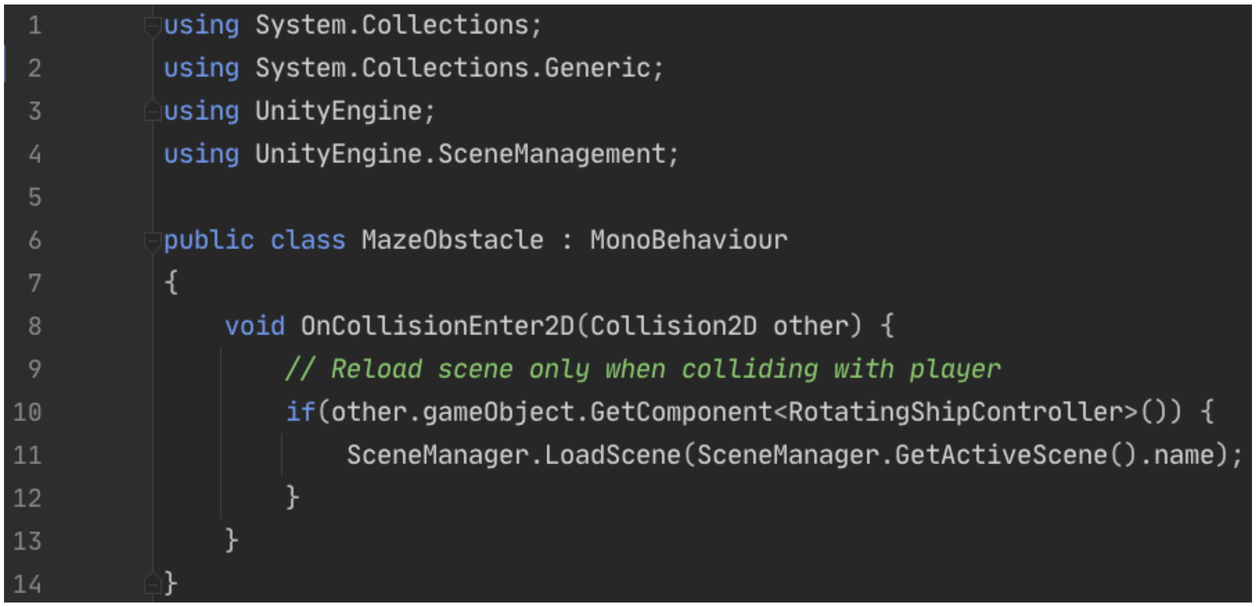
Info



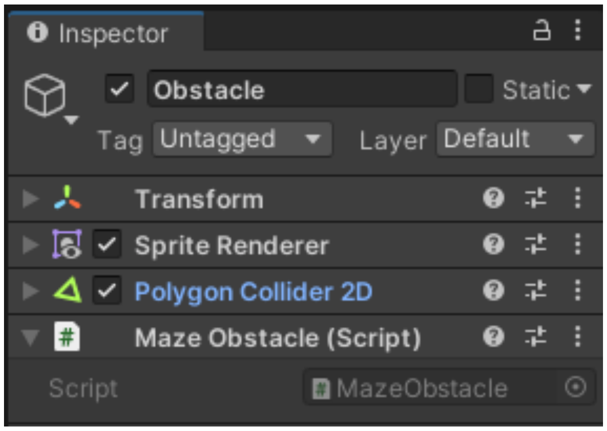
12) Create a Maze Obstacle script

Create MazeObstacle.cs in /Assets/Code/Q2/ and attach it to the Obstacle object.

Our Obstacle objects will check for Collision events with the Player and reload the Scene.



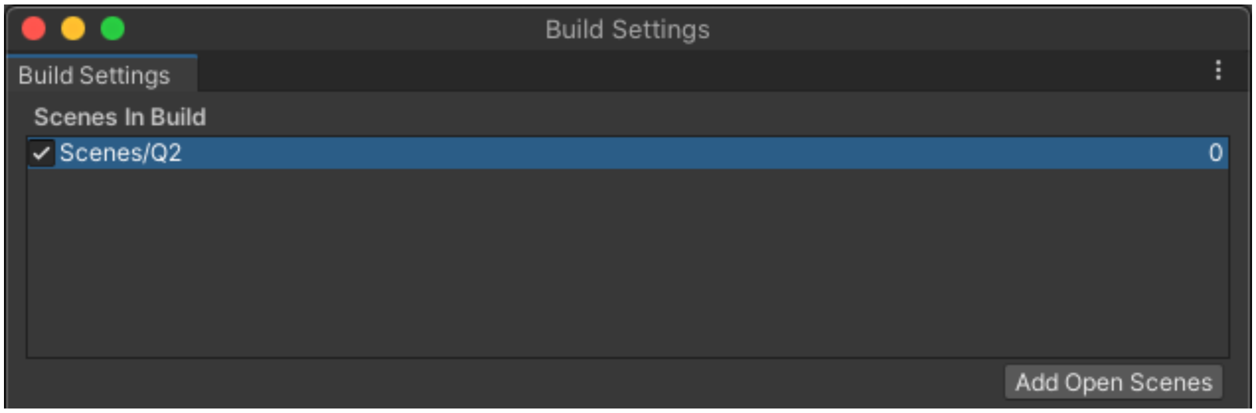
```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.SceneManagement;

public class MazeObstacle : MonoBehaviour
{
    void OnCollisionEnter2D(Collision2D other) {
        // Reload scene only when colliding with player
        if(other.gameObject.GetComponent<RotatingShipController>()) {
            SceneManager.LoadScene(SceneManager.GetActiveScene().name);
        }
    }
}
```



Inspector
✓ Obstacle Static
Tag Untagged Layer Default
Transform
✓ Sprite Renderer
✓ Polygon Collider 2D
Maze Obstacle (Script)
Script MazeObstacle

13) Scene Setup

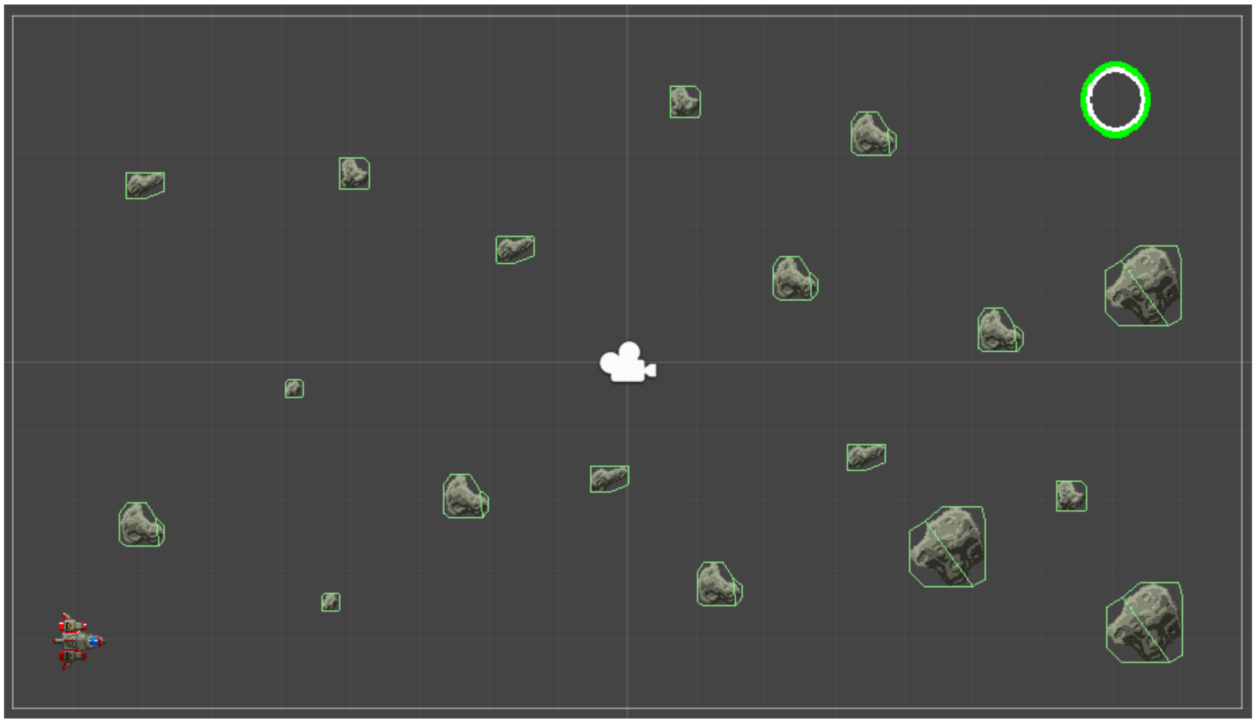
The SceneManager can only operate on scenes enabled in the Build Settings. Go to File->Build Settings. Ensure the current Scene is listed in “Scenes in Build” and is checked. You can drag in Scenes from the Projects tab.



Build Settings
Build Settings
Scenes In Build
✓ Scenes/Q2 0
Add Open Scenes

14) Level Design

Create a challenging Level Design in which the Player must navigate through Obstacles to reach a Goal. You do not need to script the Goal for this assignment. A Goal graphic is all that is necessary.You must ensure all the asteroids have a collider shape that matches their graphic.



15) PlayTest

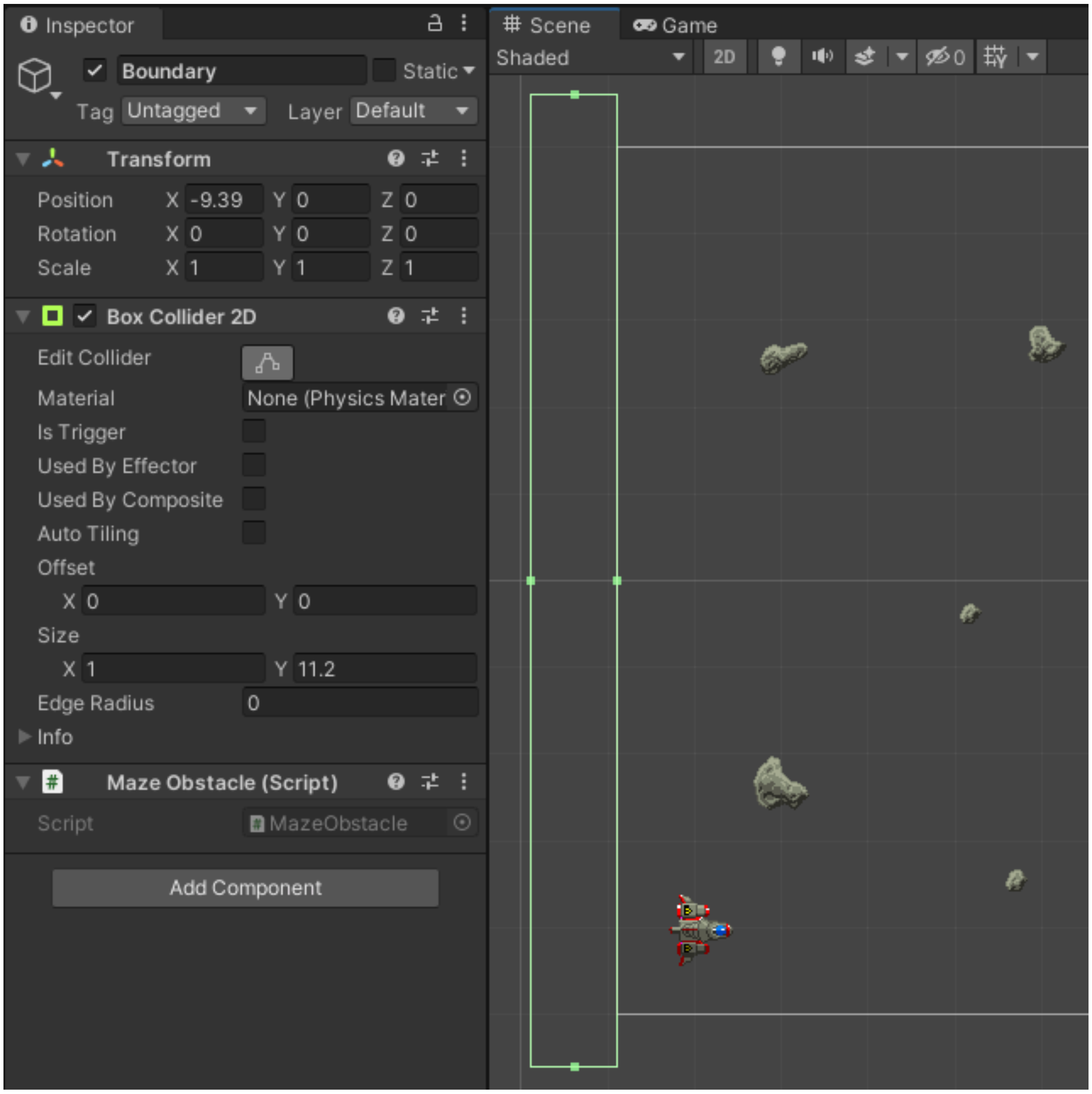
PlayTest your Level Design and adjust Level Layout and Player Physics until you are satisfied with the Game Feel.

16) Create Level Boundaries

Again, ensure your Game tab is set to an Aspect Ratio of 16:9.

Obstacles do not always have to have graphics.

Create Obstacles that represent the level boundaries by making a GameObject that has BoxColliders but no SpriteRenderer. Use the “Edit Collider” button to shape the Collider. DO NOT use the Transform->Scale to stretch it.



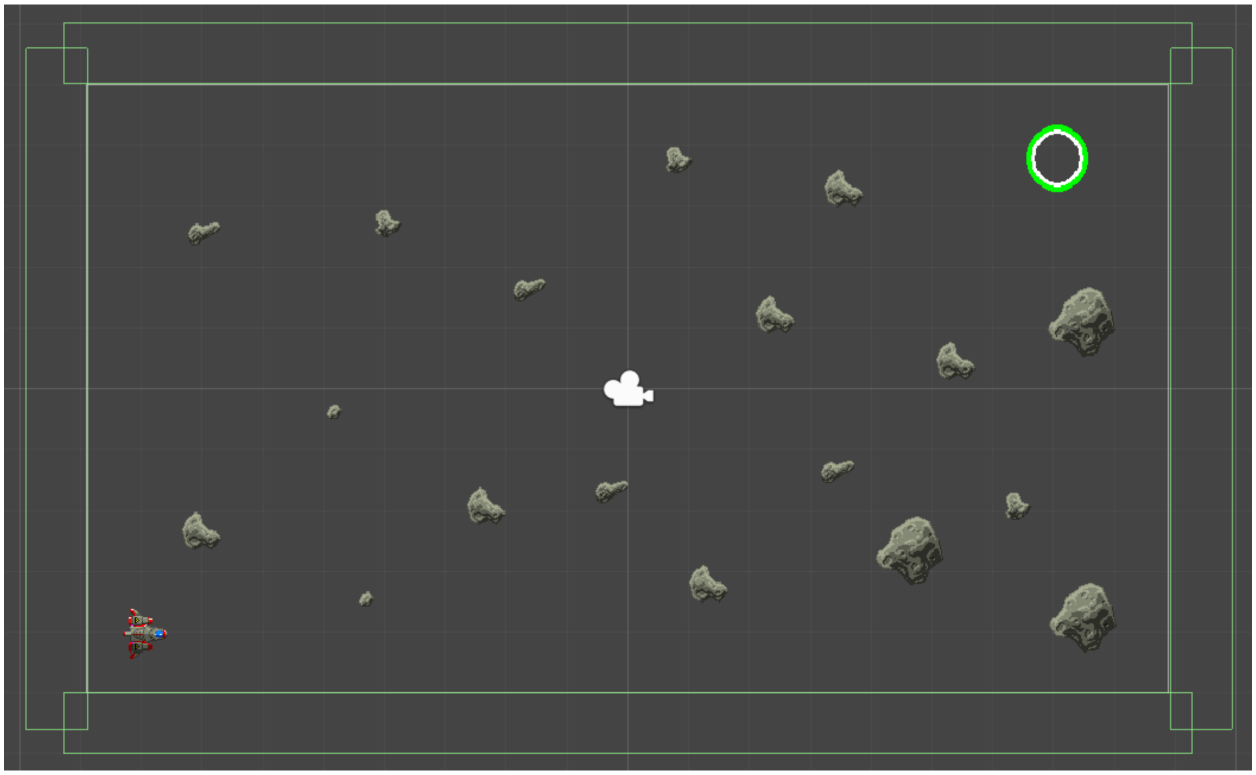
Boundary ✓ Static
Tag Untagged Layer Default

Transform
Position X -9.39 Y 0 Z 0
Rotation X 0 Y 0 Z 0
Scale X 1 Y 1 Z 1

Box Collider 2D ✓
Edit Collider
Material None (Physics Mater
Is Trigger
Used By Effector
Used By Composite
Auto Tiling
Offset
X 0 Y 0
Size
X 1 Y 11.2
Edge Radius 0
Info

Maze Obstacle (Script)
Script MazeObstacle

Finalize all four level boundaries. Ensure there are no gaps where the ship could fall out of the level.



17) PlayTest

Play your game again with the added difficulty and adjust balance for game feel.