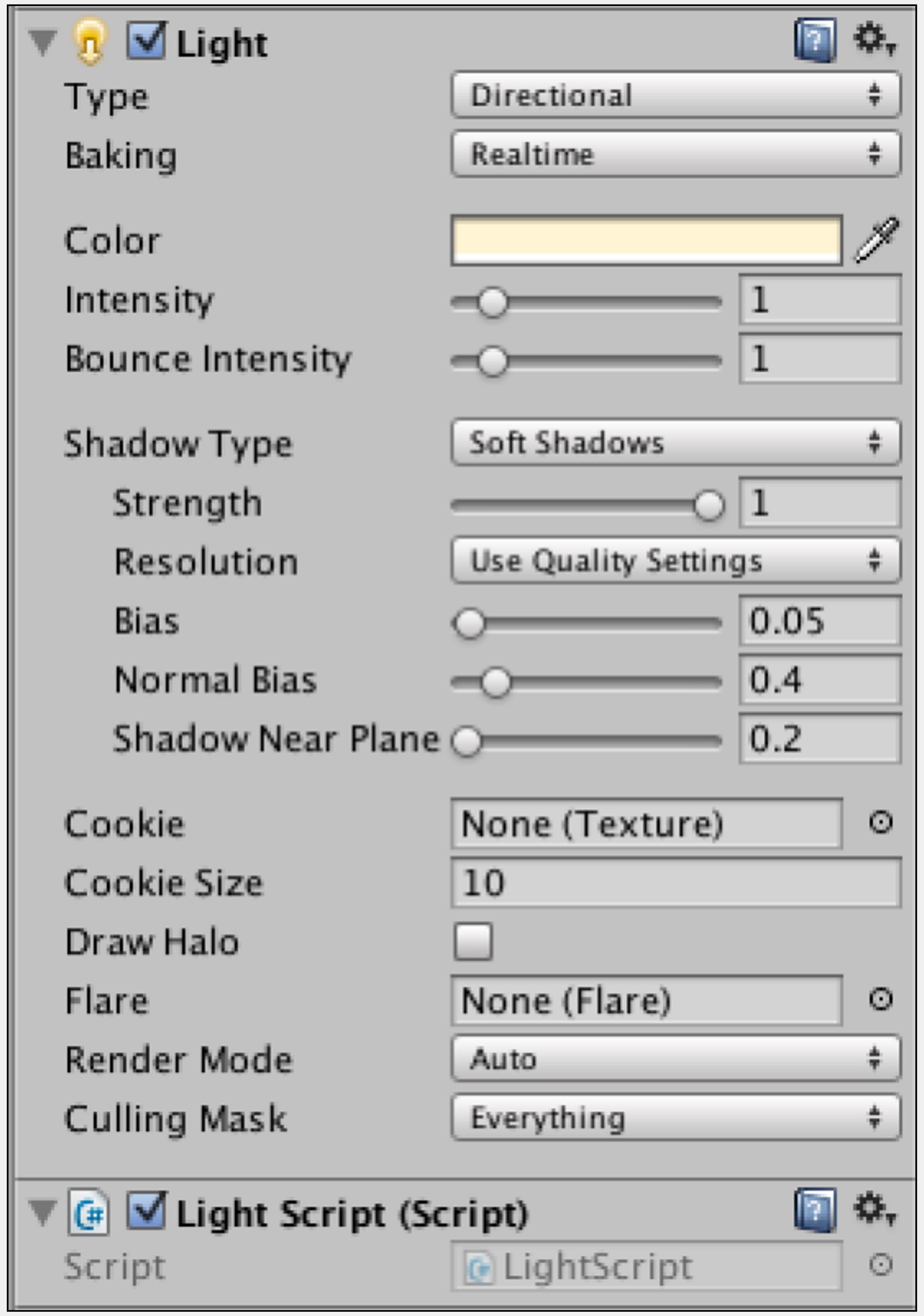
Scripting and UI

Jack Miller and Mitchell Talyat

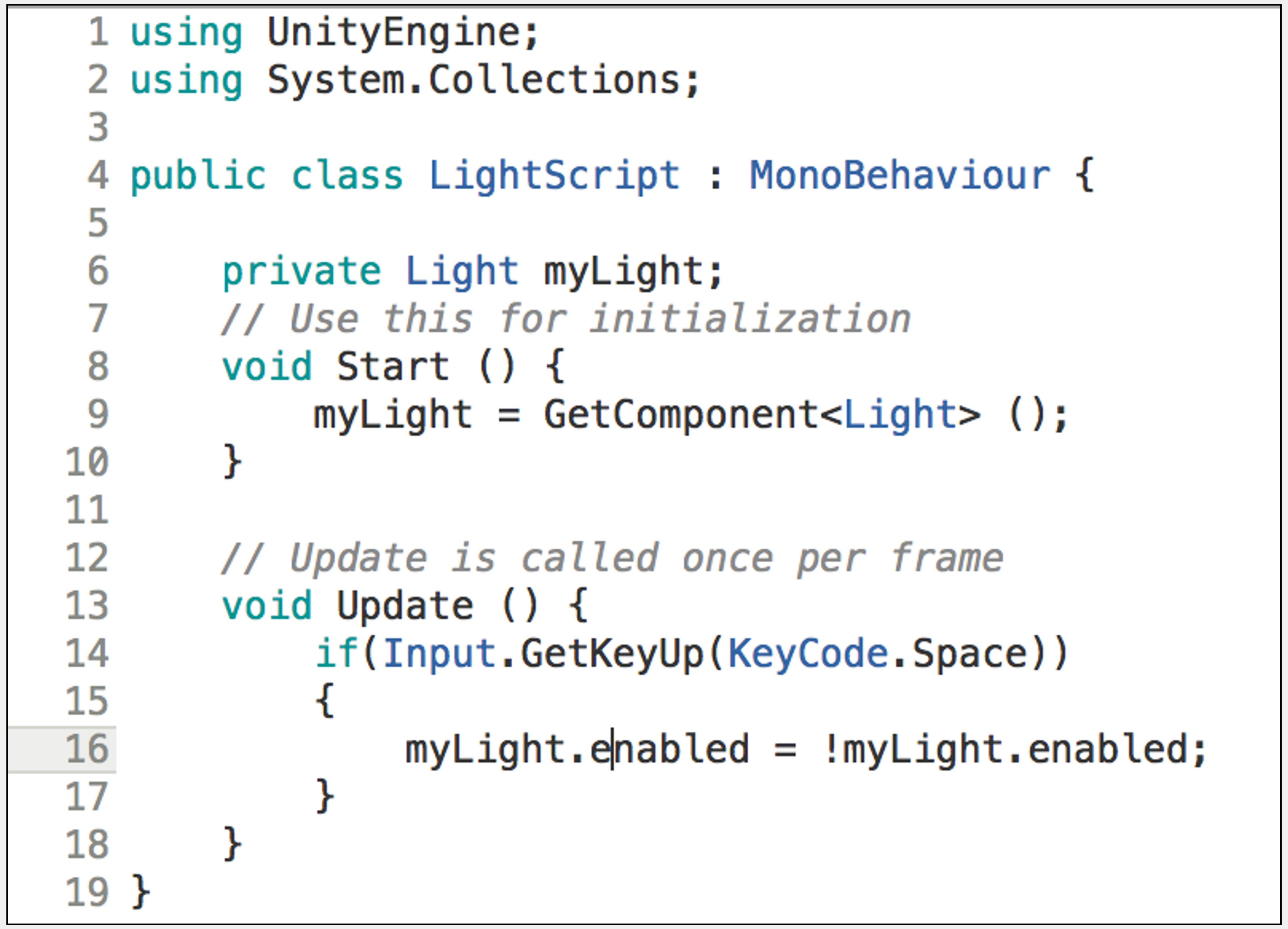
Day 1 Review

- Game Engines
- Unity Interface
- Cameras, Lights, and Objects
- Scripting in C#

Enabling and Disabling Components



☑ Light	
Type	Directional
Baking	Realtime
Color	
Intensity	1
Bounce Intensity	1
Shadow Type	Soft Shadows
Strength	1
Resolution	Use Quality Settings
Bias	0.05
Normal Bias	0.4
Shadow Near Plane	0.2
Cookie	None (Texture)
Cookie Size	10
Draw Halo	☐
Flare	None (Flare)
Render Mode	Auto
Culling Mask	Everything
☑ Light Script (Script)	
Script	LightScript

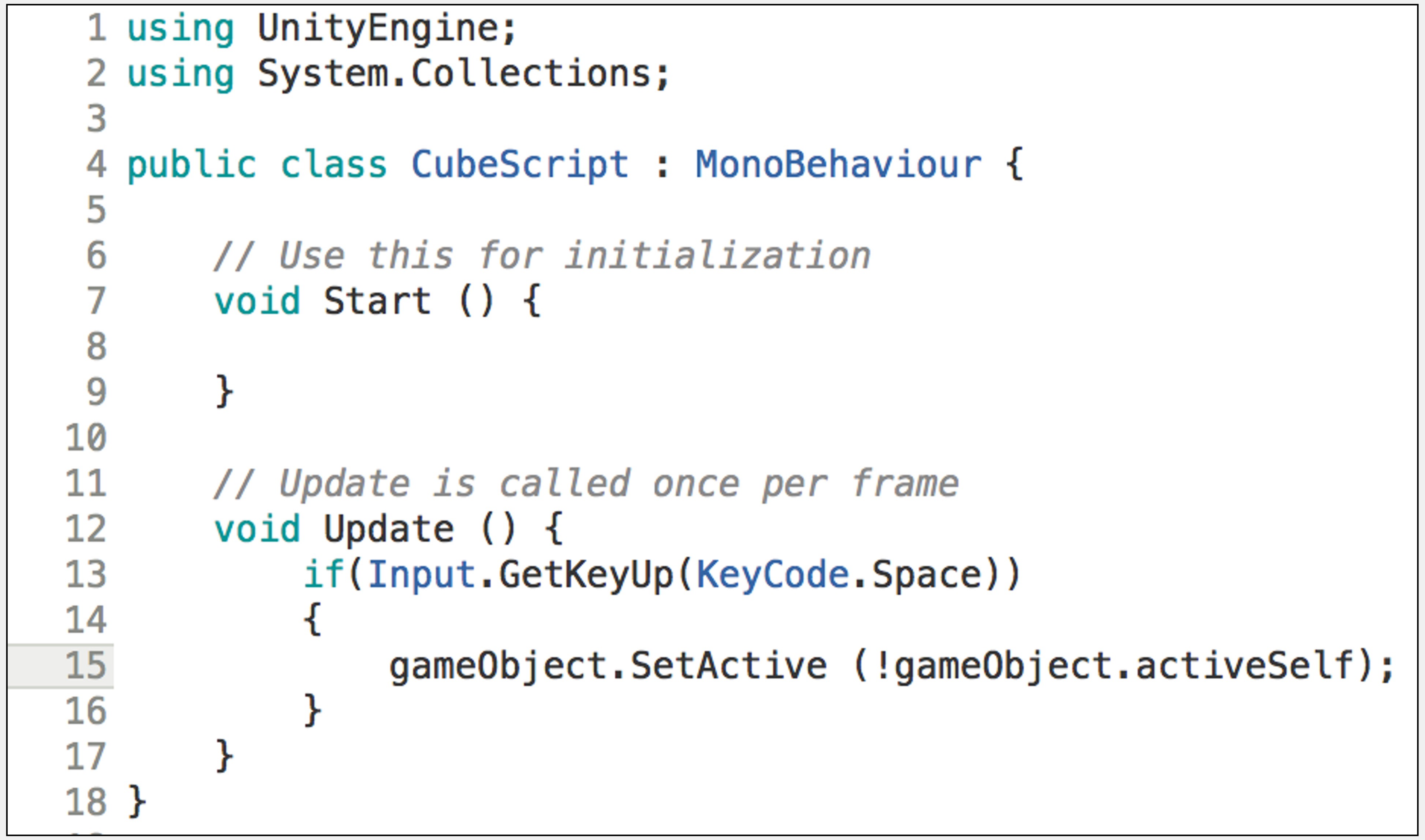


```
using UnityEngine;
using System.Collections;

public class LightScript : MonoBehaviour {

    private Light myLight;
    // Use this for initialization
    void Start () {
        myLight = GetComponent<Light> ();
    }

    // Update is called once per frame
    void Update () {
        if(Input.GetKeyUp(KeyCode.Space))
        {
            myLight.enabled = !myLight.enabled;
        }
    }
}
```

Activating Game Objects

- Making a GameObject inactive will disable every component and turn off any attached renderers, colliders, rigid bodies, scripts, etc...

- Any scripts that you have attached to the GameObject will no longer have Update() called

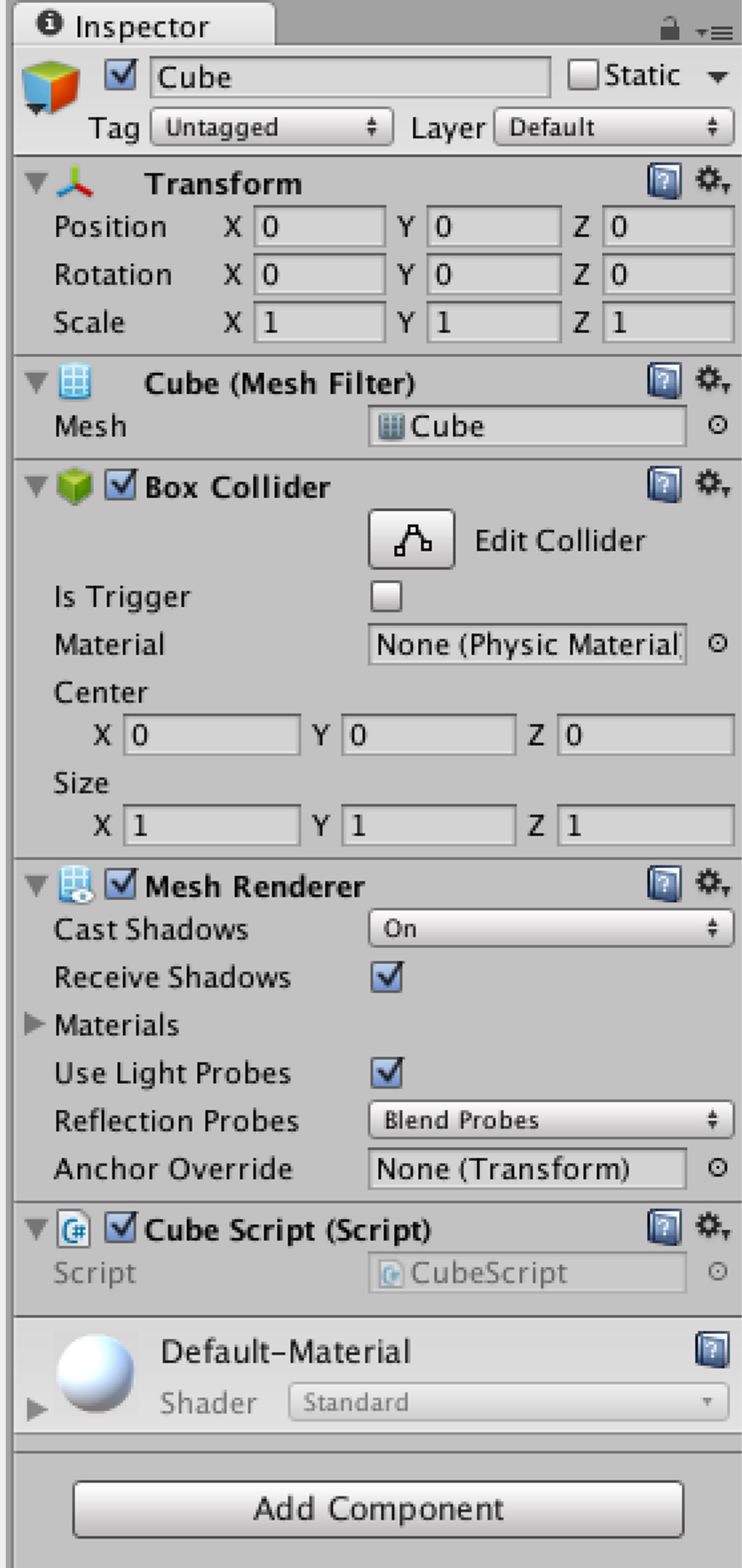


```
using UnityEngine;
using System.Collections;

public class CubeScript : MonoBehaviour {

    // Use this for initialization
    void Start () {

    }

    // Update is called once per frame
    void Update () {
        if(Input.GetKeyUp(KeyCode.Space))
        {
            gameObject.SetActive (!gameObject.activeSelf);
        }
    }
}
```

Virtual Reality Applications Center

Getting a Component

- GetComponent<Type>()
- Allows you access to any Component in the object
- You can access Parent and Children too



Inspector

Cube — 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

Cube (Mesh Filter)
Mesh: Cube

Box Collider
Edit Collider
Is Trigger: ☐
Material: None (Physic Material)
Center
X 0 Y 0 Z 0
Size
X 1 Y 1 Z 1

Mesh Renderer
Cast Shadows: On
Receive Shadows: ✓
Materials
Use Light Probes: ✓
Reflection Probes: Blend Probes
Anchor Override: None (Transform)

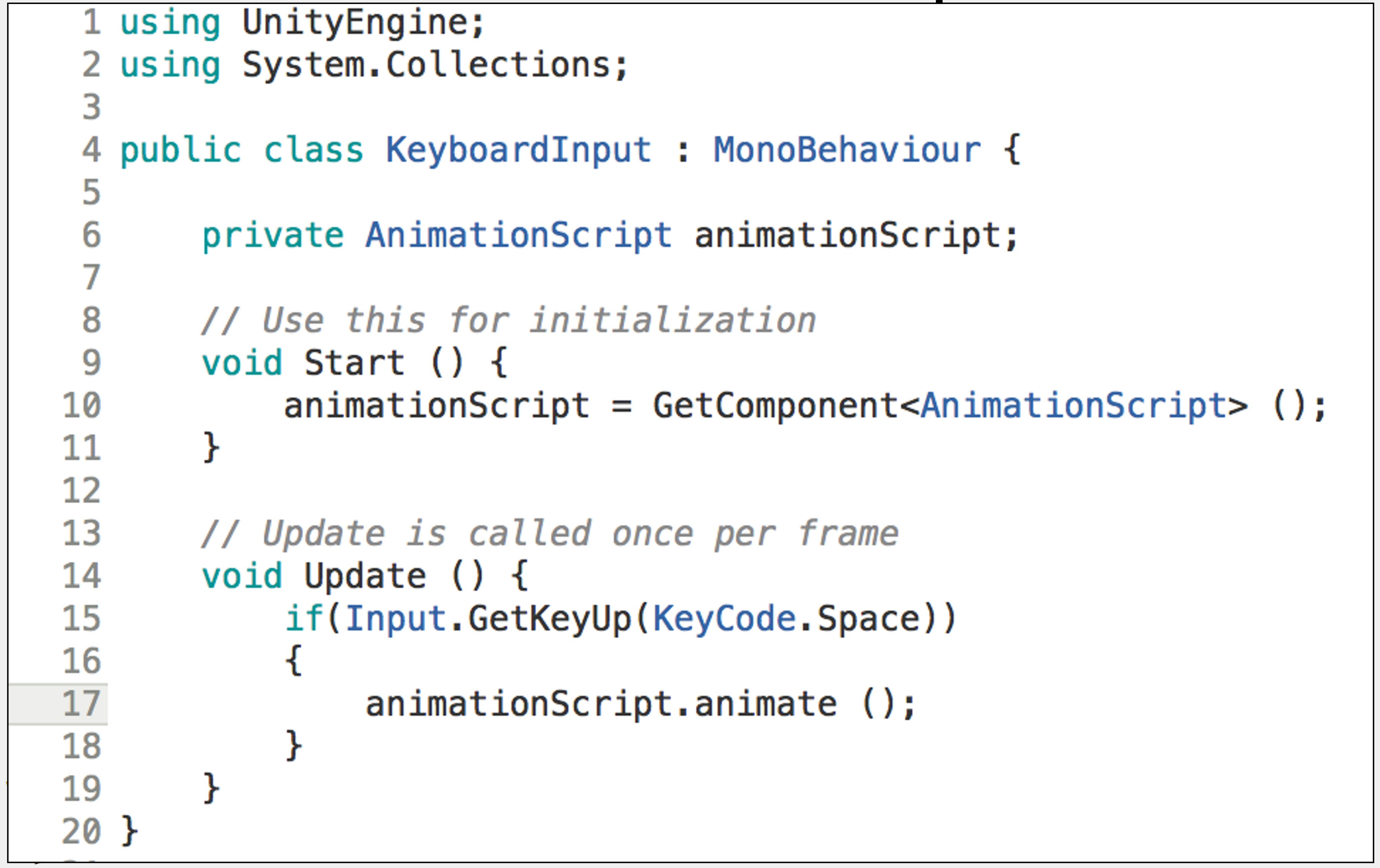
Cube Script (Script)
Script: CubeScript

Default-Material
Shader: Standard

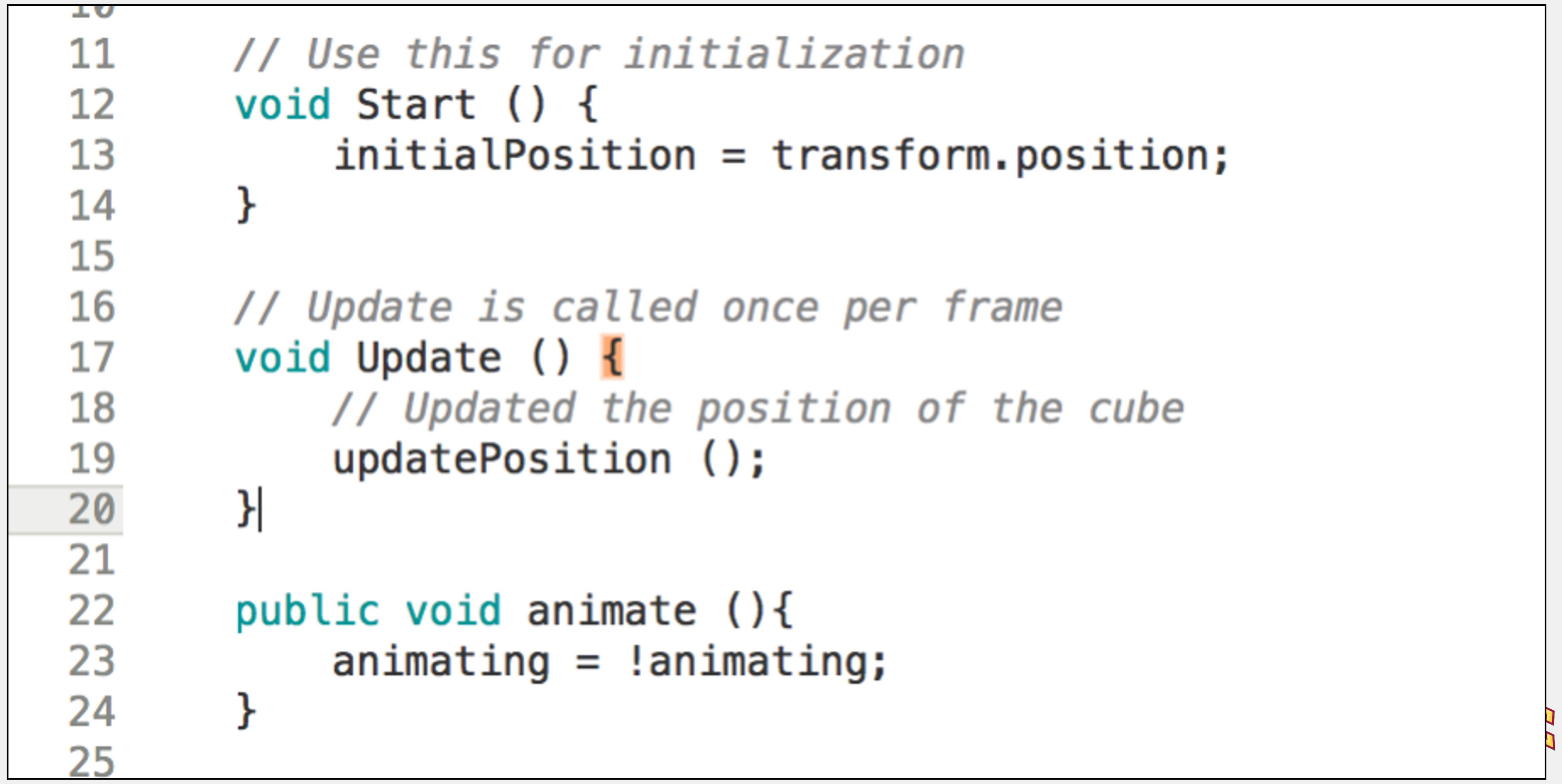
Add Component

Calling Other Scripts

- Scripts are GameComponents, so you can use GetComponent<Type>()or FindObjectOfType<Type>() to obtain a reference to other scripts



```
using UnityEngine;
using System.Collections;

public class KeyboardInput : MonoBehaviour {

    private AnimationScript animationScript;

    // Use this for initialization
    void Start () {
        animationScript = GetComponent<AnimationScript> ();
    }

    // Update is called once per frame
    void Update () {
        if(Input.GetKeyUp(KeyCode.Space))
        {
            animationScript.animate ();
        }
    }
}
```



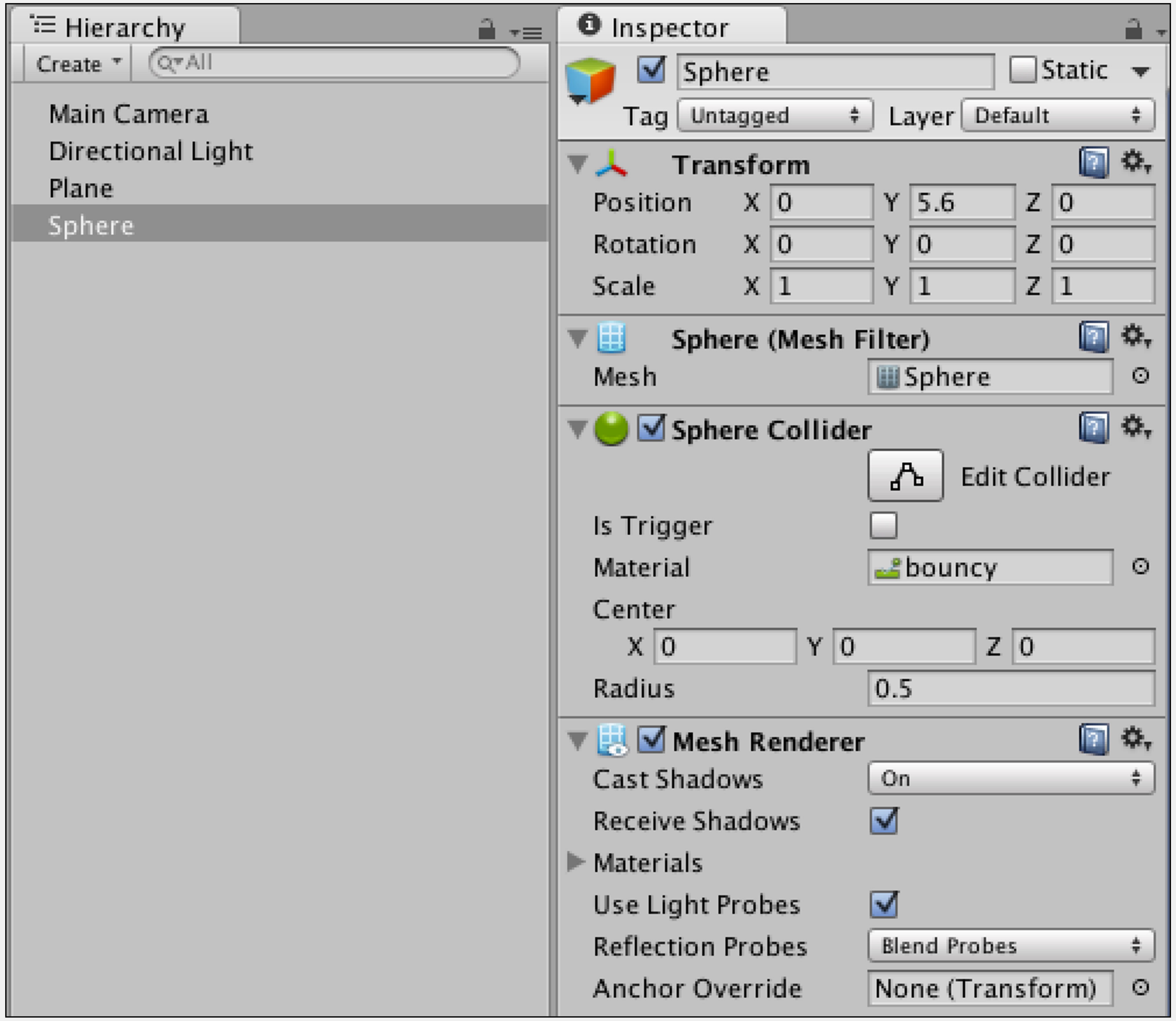
```
    // Use this for initialization
    void Start () {
        initialPosition = transform.position;
    }

    // Update is called once per frame
    void Update () {
        // Updated the position of the cube
        updatePosition ();
    }

    public void animate (){
        animating = !animating;
    }

```

Particle Systems

- Uses a large number of small objects to mimic "fuzzy" phenomena
- Fire, Smoke, Rain, Snow, Clouds, etc.

Colliders

- Allows physical interaction between objects
- Colliders react with other colliders
- Can also be used for selecting objects



Main Camera
Directional Light
Plane
Sphere

Sphere — Static
Tag Untagged Layer Default

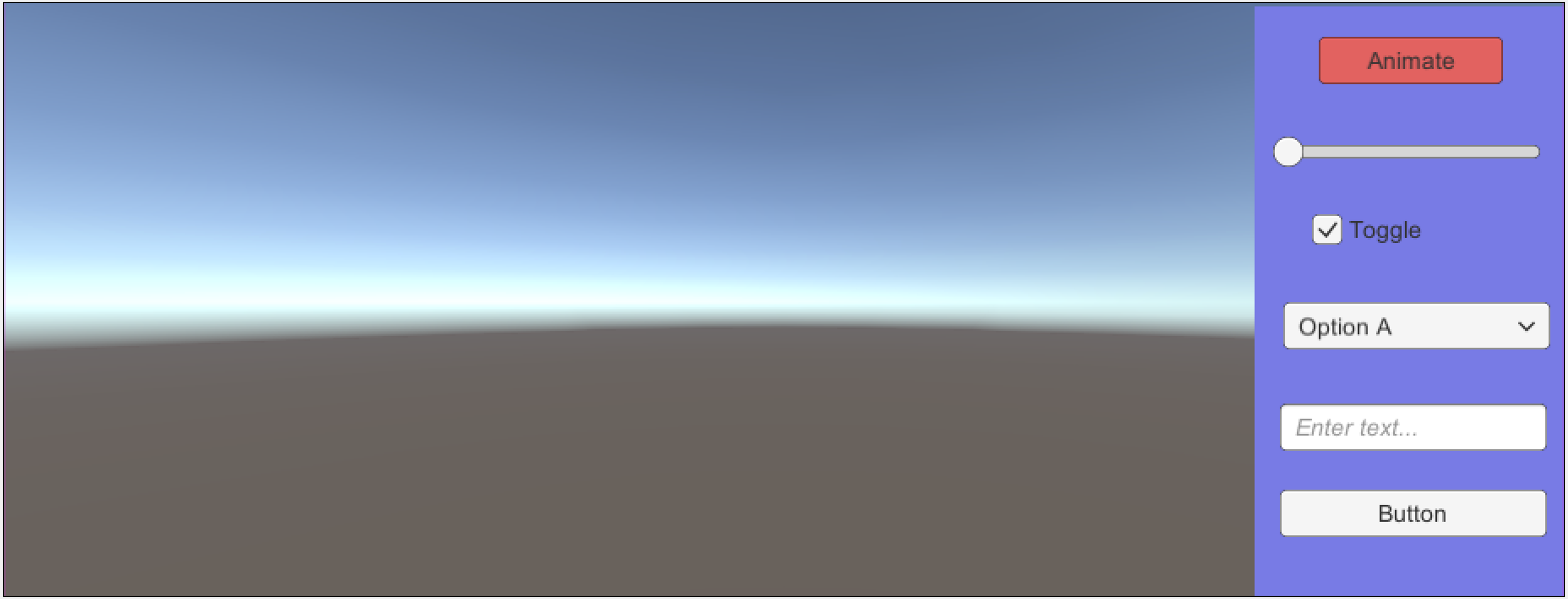
Transform
Position X 0 Y 5.6 Z 0
Rotation X 0 Y 0 Z 0
Scale X 1 Y 1 Z 1

Sphere (Mesh Filter)
Mesh Sphere

Sphere Collider
Edit Collider
Is Trigger ☐
Material bouncy
Center
X 0 Y 0 Z 0
Radius 0.5

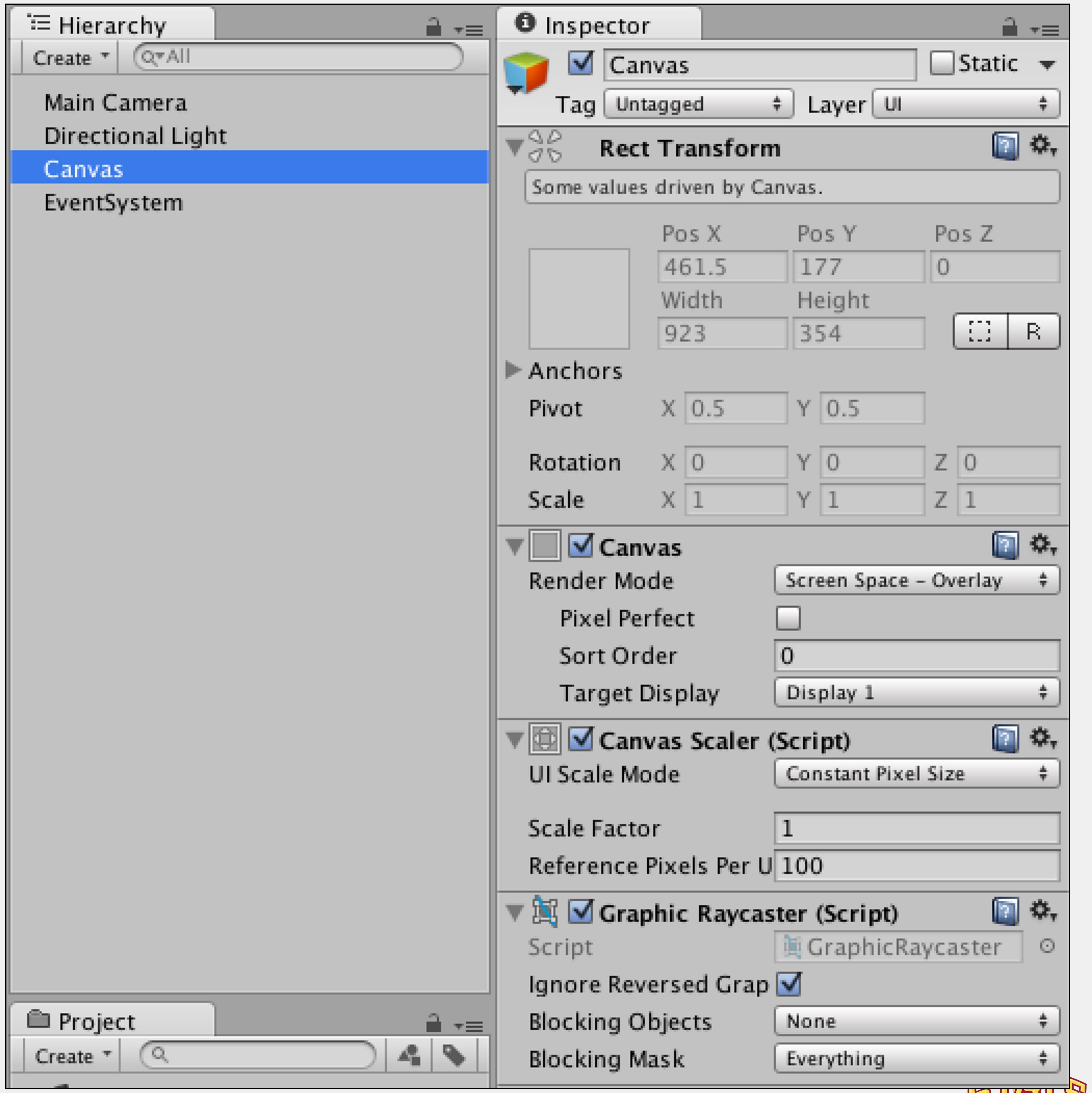
Mesh Renderer
Cast Shadows On
Receive Shadows ✓
Materials
Use Light Probes ✓
Reflection Probes Blend Probes
Anchor Override None (Transform)

Unity User Interfaces



UI Canvas

- Everything UI starts with the Canvas
- Canvas is a GameObject
- All UI elements must be children of a canvas



Hierarchy
Create ▾ | Q▾All

- Main Camera
- Directional Light
- Canvas
- EventSystem

Project
Create ▾

Inspector

Canvas ☑ | Static ☐
Tag: Untagged | Layer: UI

Rect Transform
Some values driven by Canvas.

Pos X	Pos Y	Pos Z
461.5	177	0
Width	**Height**	
923	354	

Anchors

	X	Y	Z
Pivot	0.5	0.5	
Rotation	0	0	0
Scale	1	1	1

Canvas ☑

Render Mode	Screen Space – Overlay
Pixel Perfect	☐
Sort Order	0
Target Display	Display 1

Canvas Scaler (Script) ☑

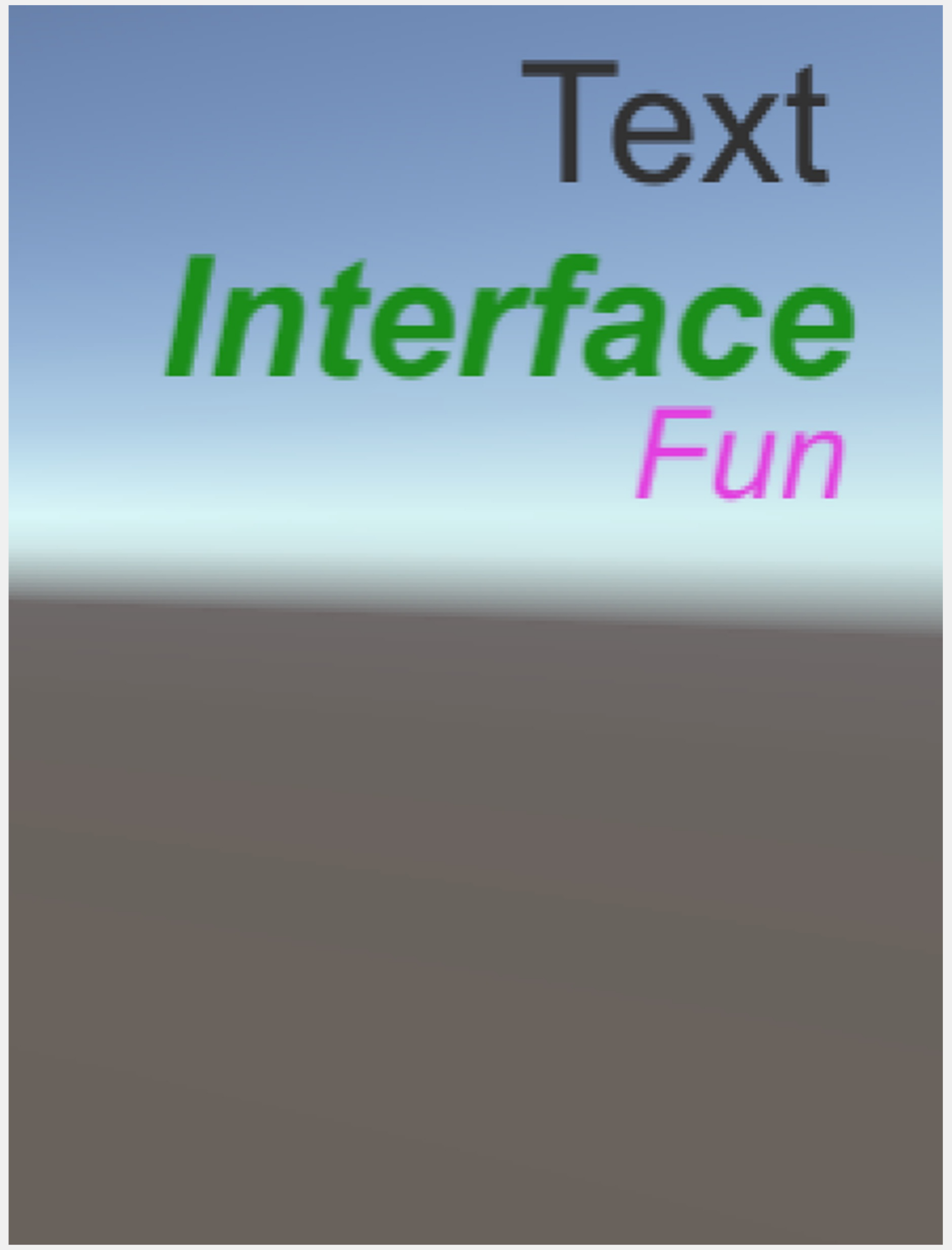
UI Scale Mode	Constant Pixel Size
Scale Factor	1
Reference Pixels Per U	100

Graphic Raycaster (Script) ☑

Script	GraphicRaycaster
Ignore Reversed Grap	☑
Blocking Objects	None
Blocking Mask	Everything

UI Text

- Whenever you need text
- Text properties can be set in the Inspector
- Can be changed during runtime through scripting

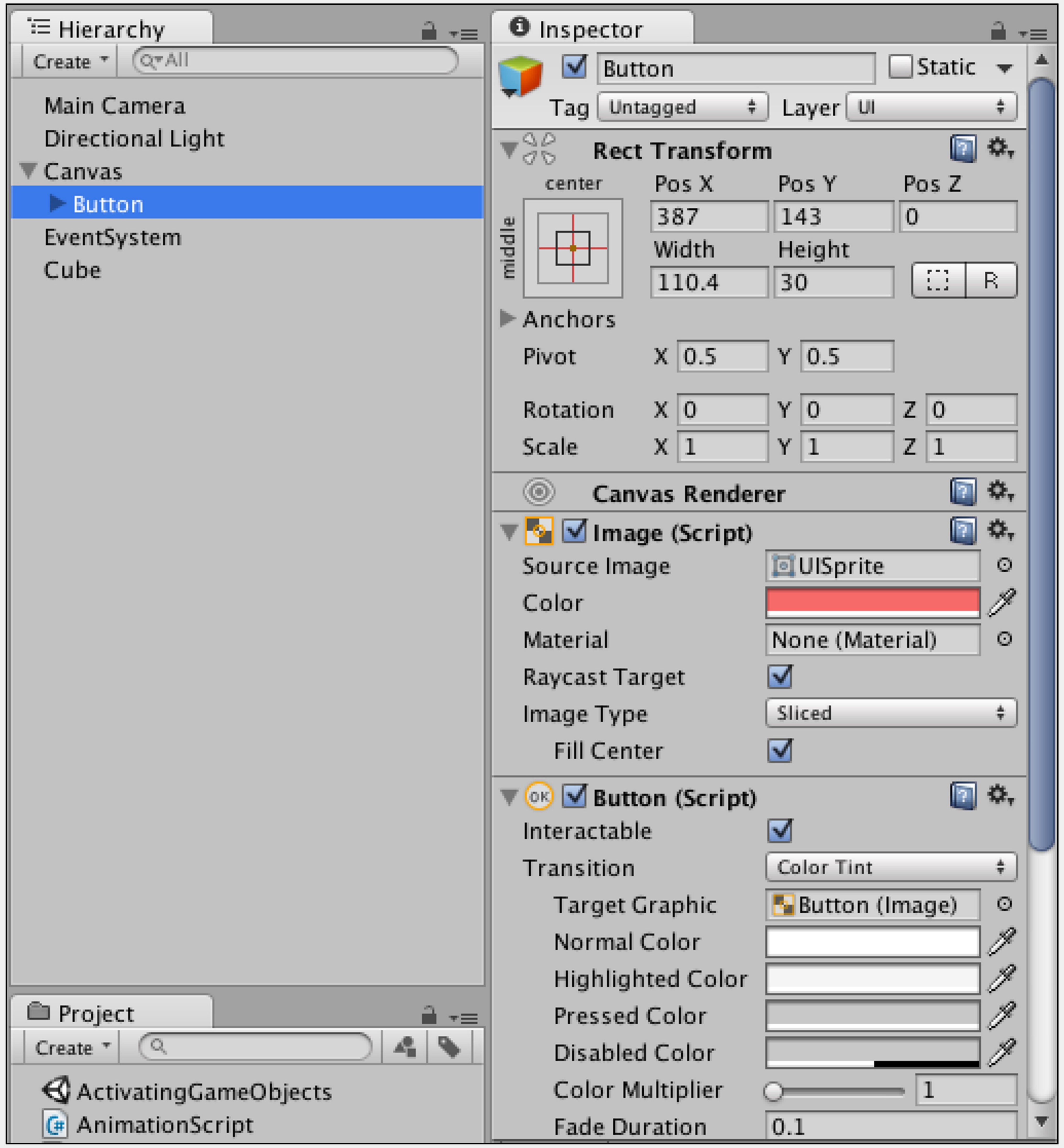


UI Image

- Can be used for almost anything, button, slider, etc.
- When importing an image, you must define what type of texture it is (Normal Map, Light Map, Sprite)
- For UI, we want a Sprite

UI Button

- Button is a GameObject that must be a child of a canvas
- Many different options for styling



Button
Static
Tag: Untagged
Layer: UI

Rect Transform
center
middle
Pos X: 387
Pos Y: 143
Pos Z: 0
Width: 110.4
Height: 30
Anchors
Pivot: X 0.5, Y 0.5
Rotation: X 0, Y 0, Z 0
Scale: X 1, Y 1, Z 1

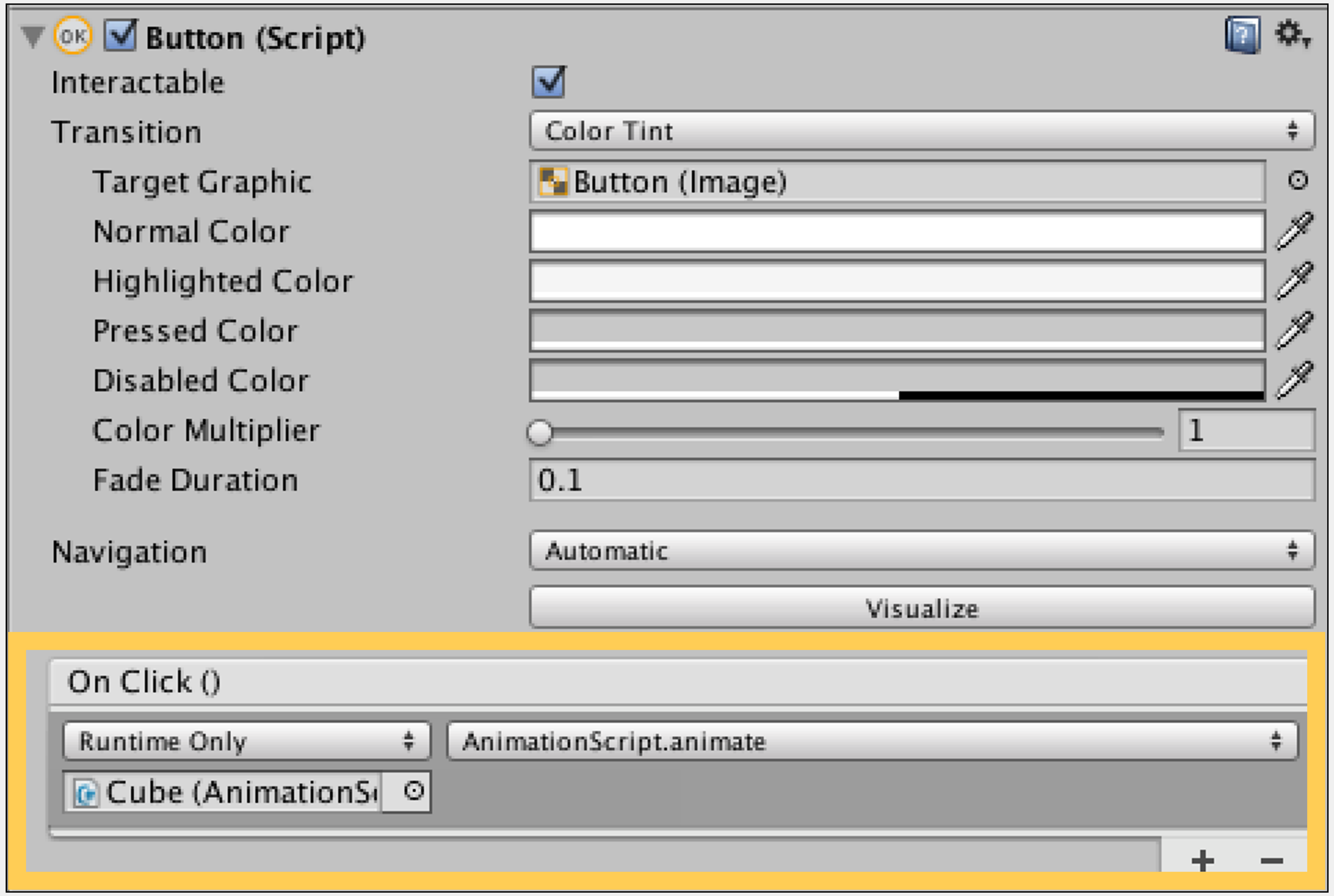
Canvas Renderer

Image (Script)
Source Image: UISprite
Color
Material: None (Material)
Raycast Target: ✓
Image Type: Sliced
Fill Center: ✓

Button (Script)
Interactable: ✓
Transition: Color Tint
Target Graphic: Button (Image)
Normal Color
Highlighted Color
Pressed Color
Disabled Color
Color Multiplier: 1
Fade Duration: 0.1

On Click()

- You can hook up a button to an action through the Inspector
- Chose your GameObject
- Choose your Component
- Choose your Method



Button (Script)

Interactable ✓
Transition: Color Tint
Target Graphic: Button (Image)
Normal Color
Highlighted Color
Pressed Color
Disabled Color
Color Multiplier: 1
Fade Duration: 0.1
Navigation: Automatic
Visualize

On Click ()
Runtime Only | AnimationScript.animate
Cube (AnimationS

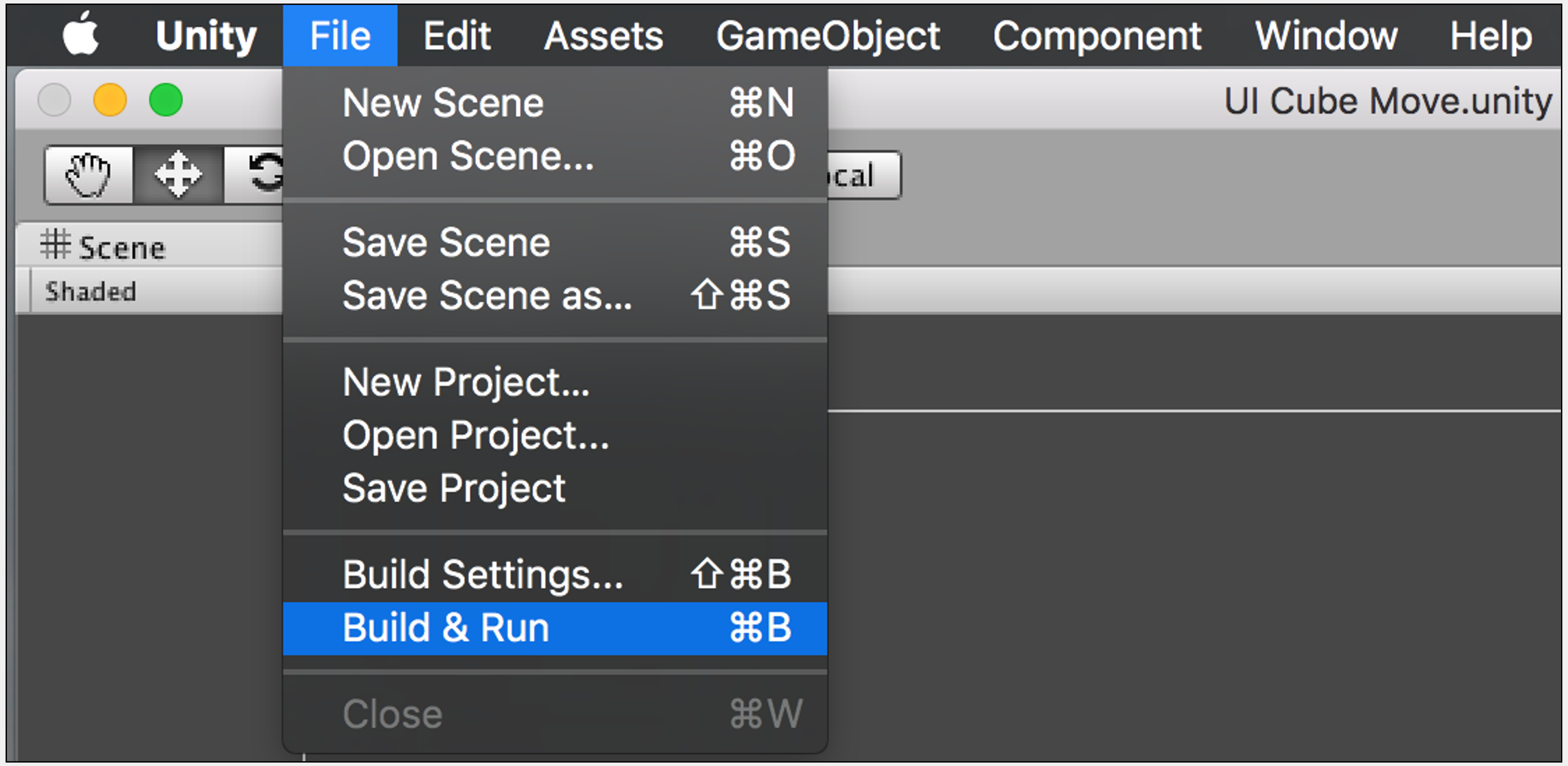
Activity

- Using the same scene
- Play around with the existing UI
- Add new UI elements and functionality

Creating an Executable

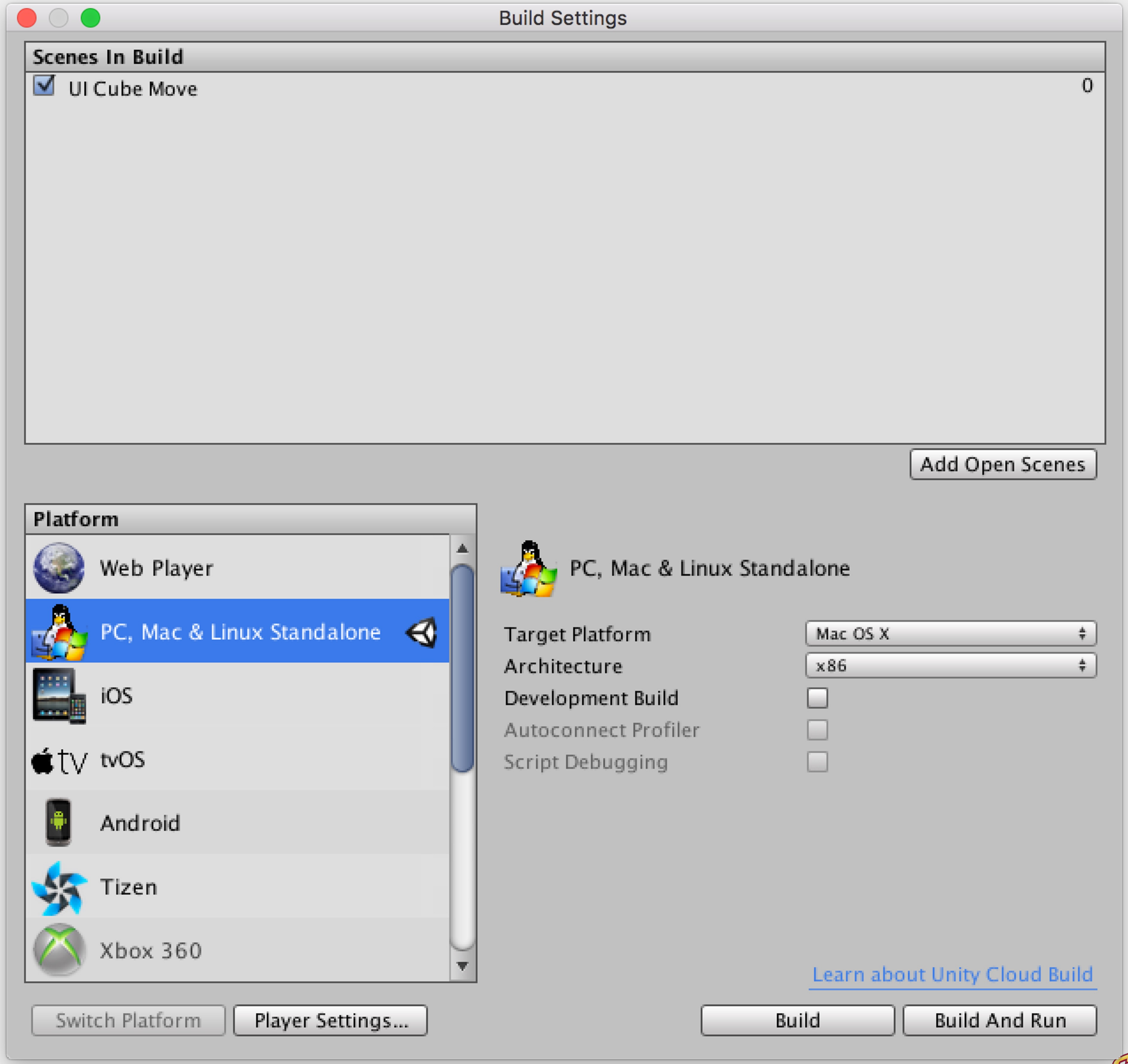
- What if I want to create a standalone app?

- Let's make an executable



Creating an Executable

- Add the desired scene
- Select your platform
- Build and Run!



Target Platform: Mac OS X
Architecture: x86
Development Build: ☐
Autoconnect Profiler: ☐
Script Debugging: ☐