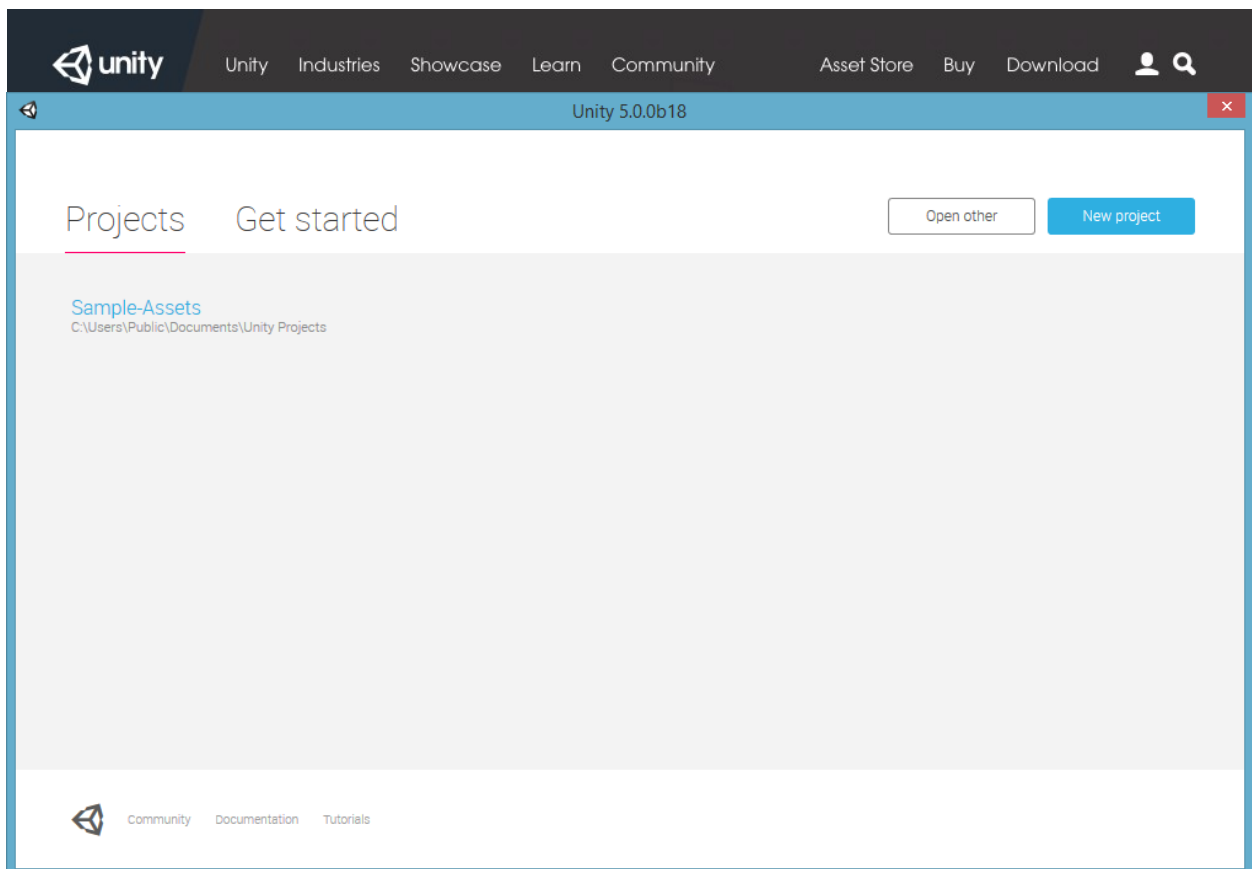
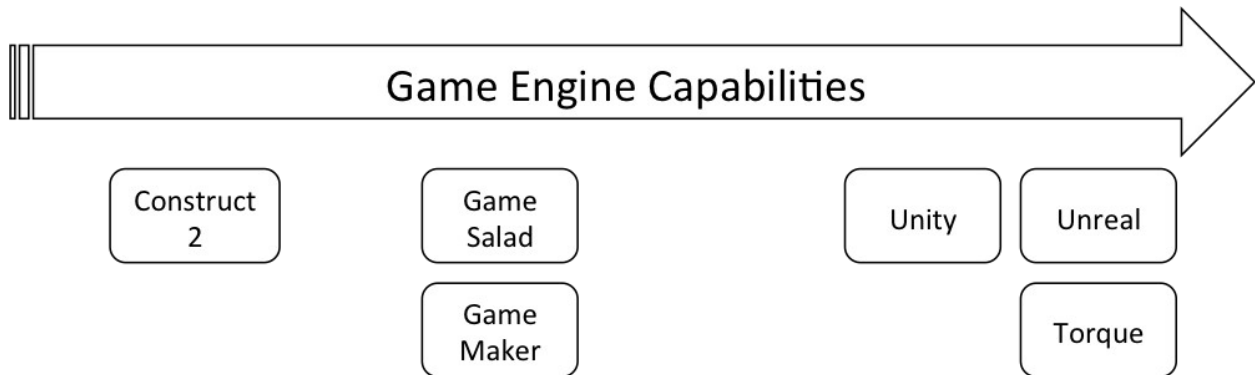
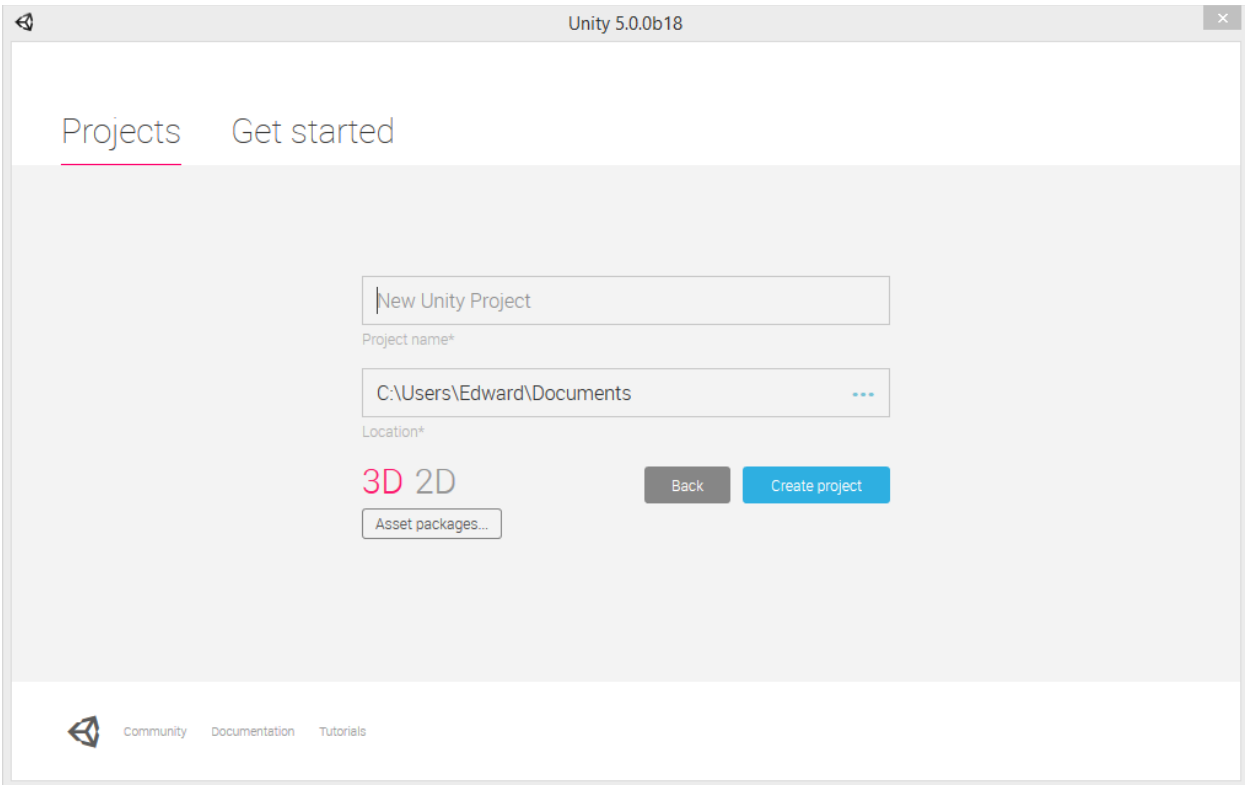
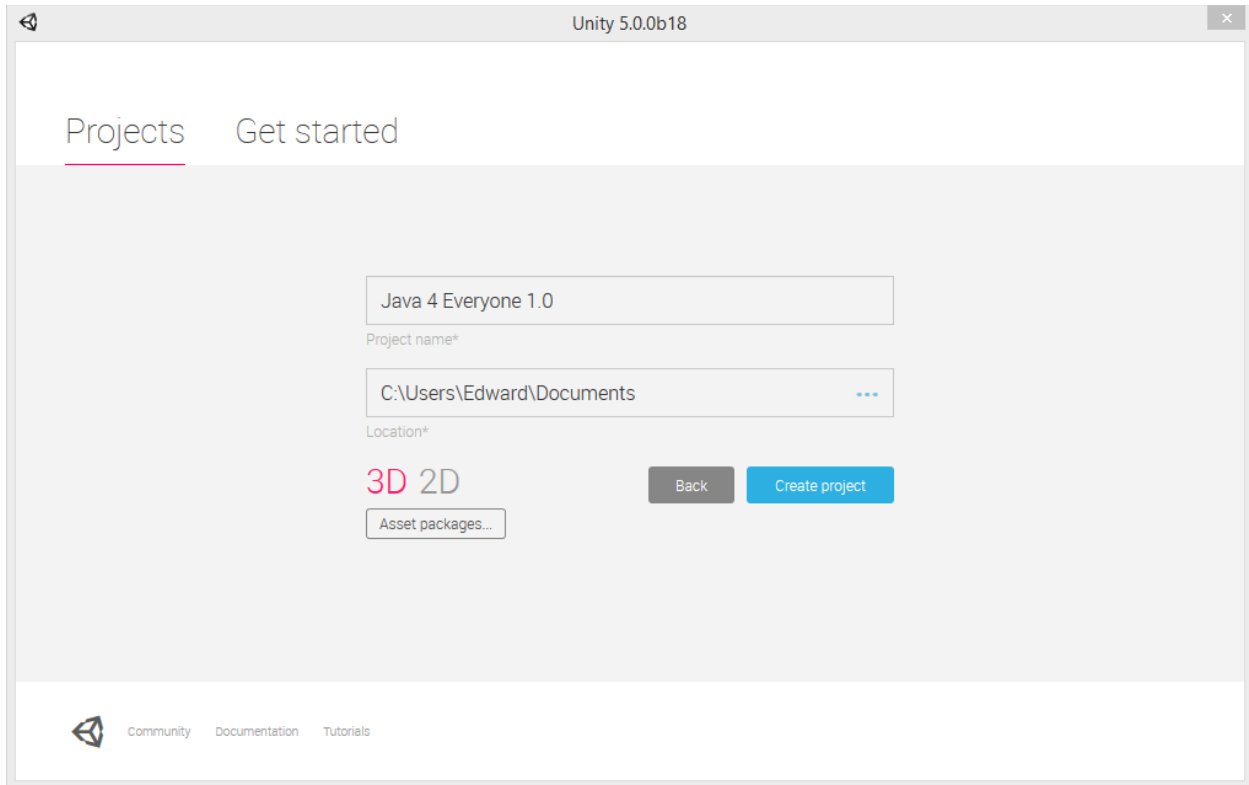


Chapter 1: Getting Jiggy with the Unity Interface

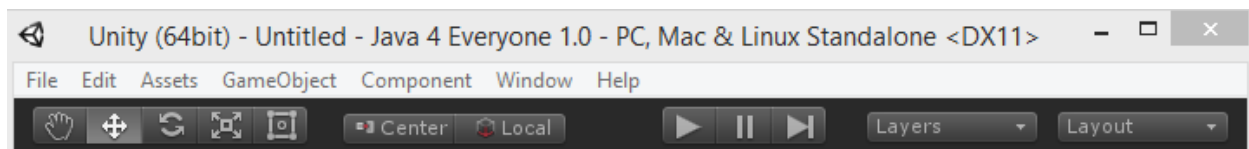


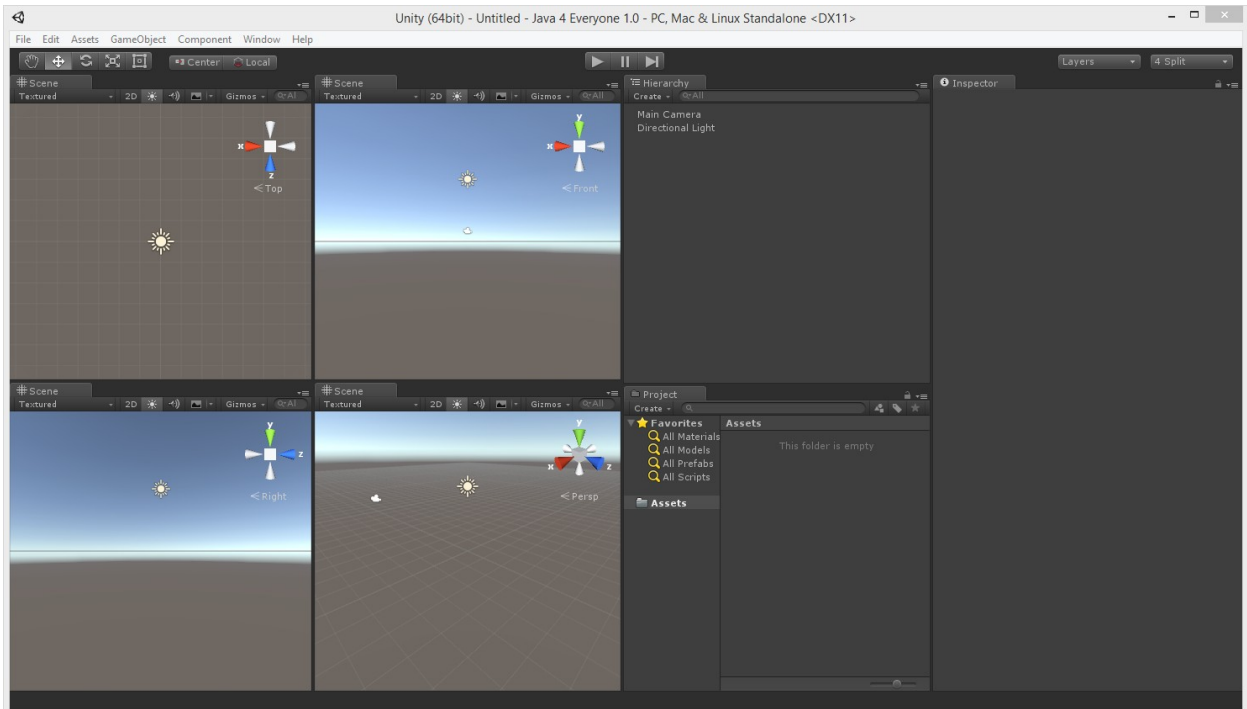
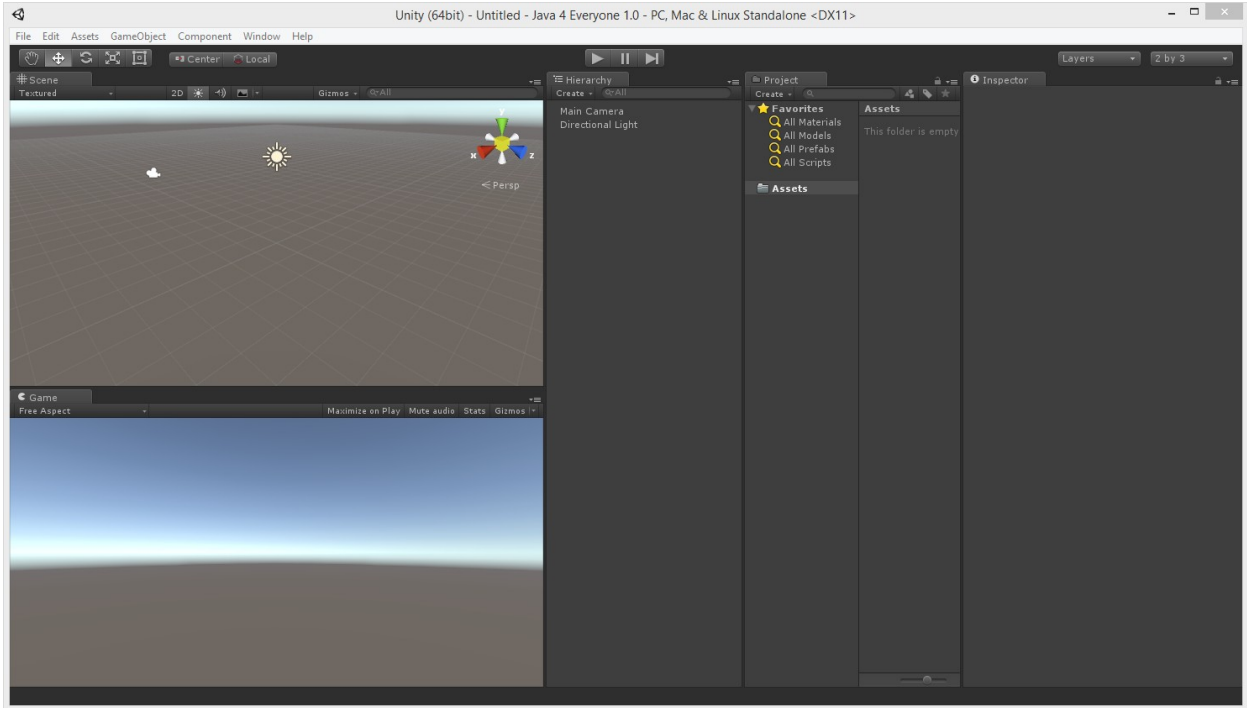


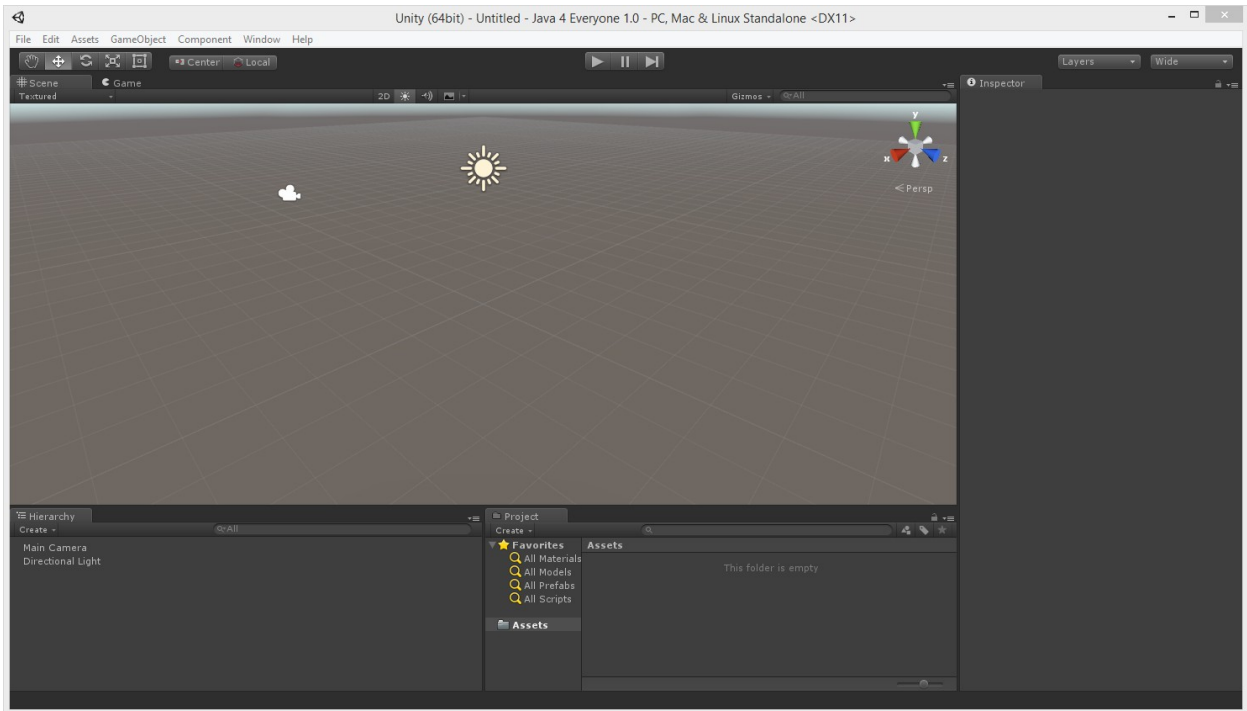
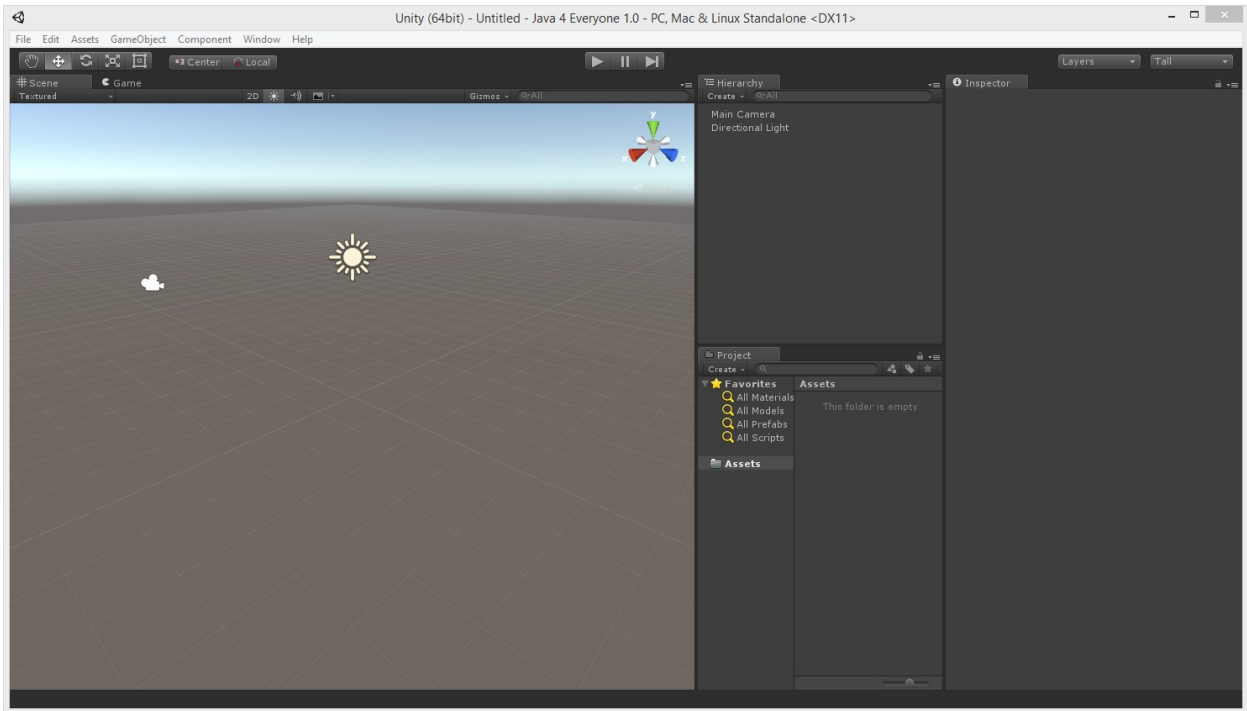


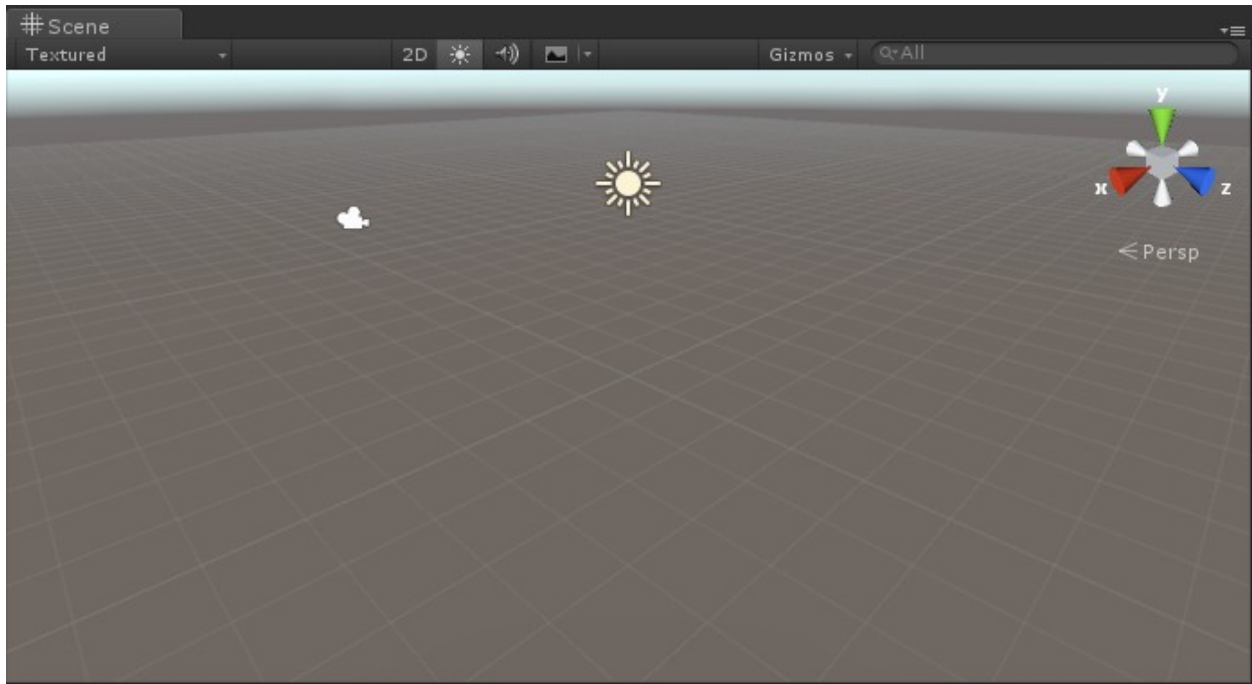
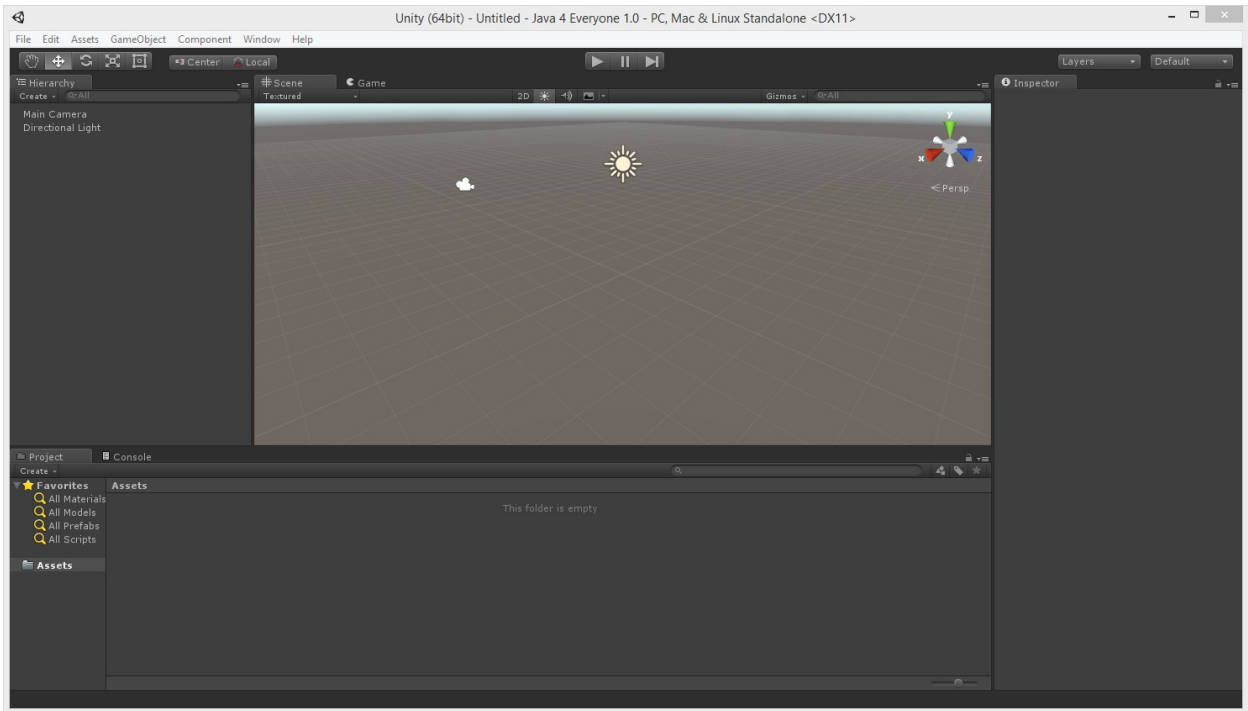
Documents ▸ Java 4 Everyone 1.0

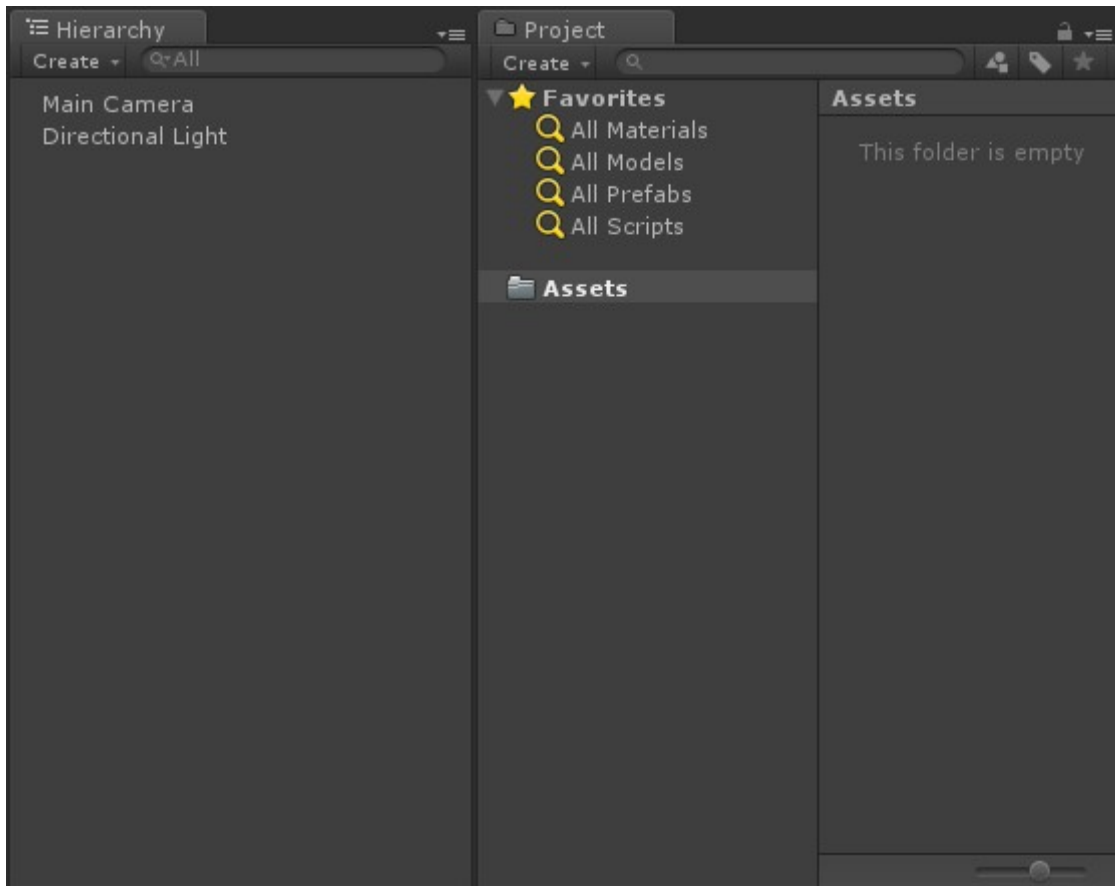
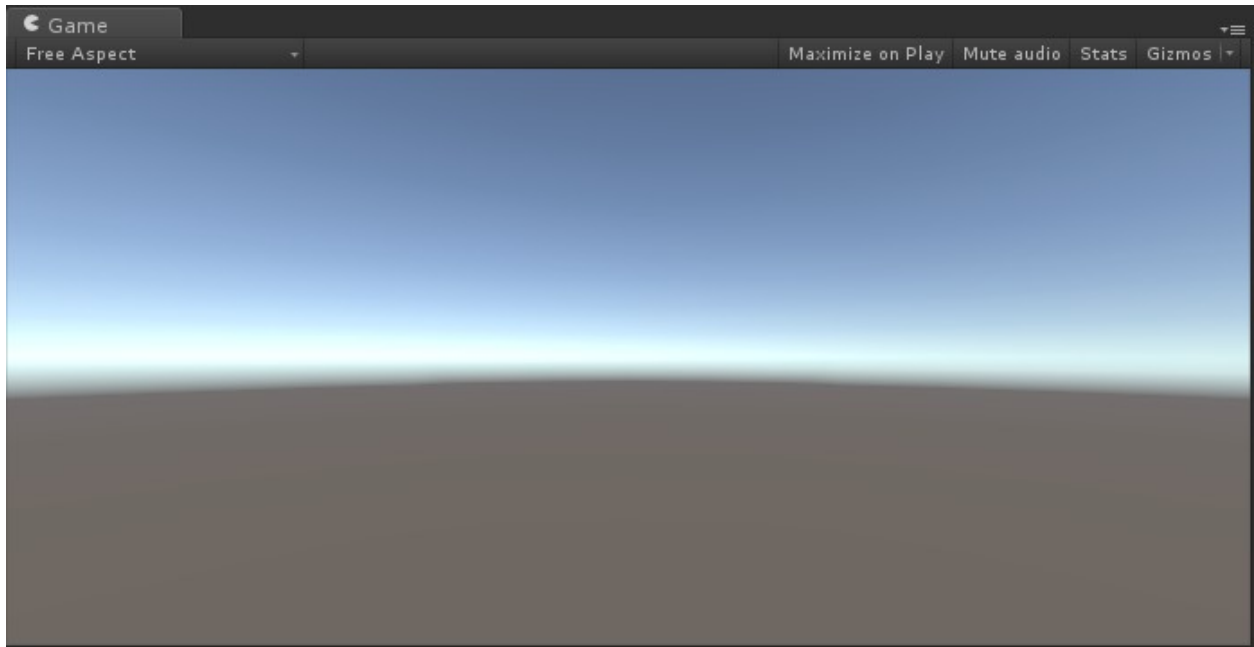
Name	Date modified	Type
Assets	1/27/2015 2:03 PM	File folder
Library	1/27/2015 2:03 PM	File folder
ProjectSettings	1/27/2015 2:03 PM	File folder
Temp	1/27/2015 2:03 PM	File folder

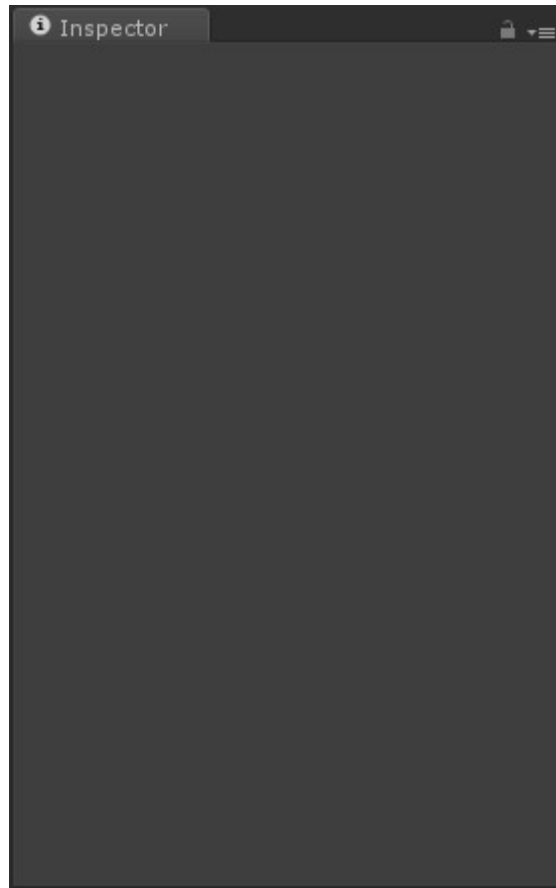


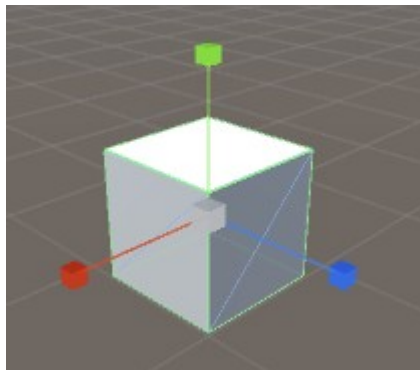
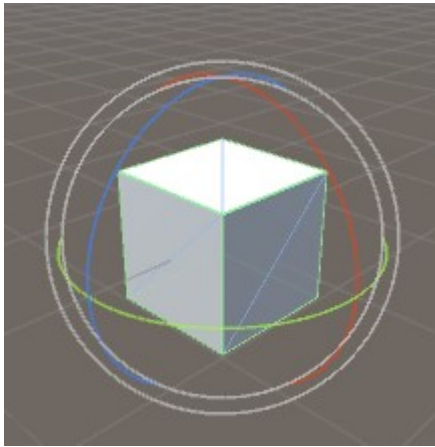
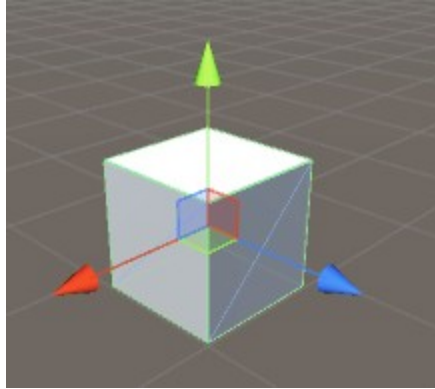


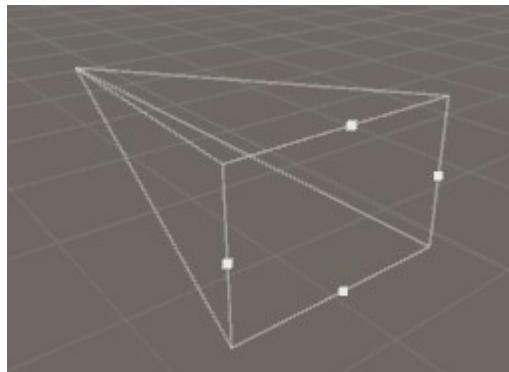
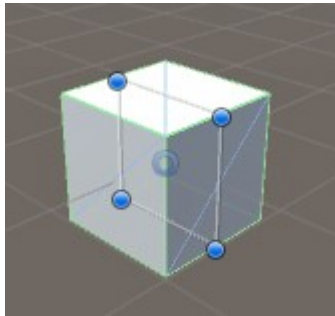




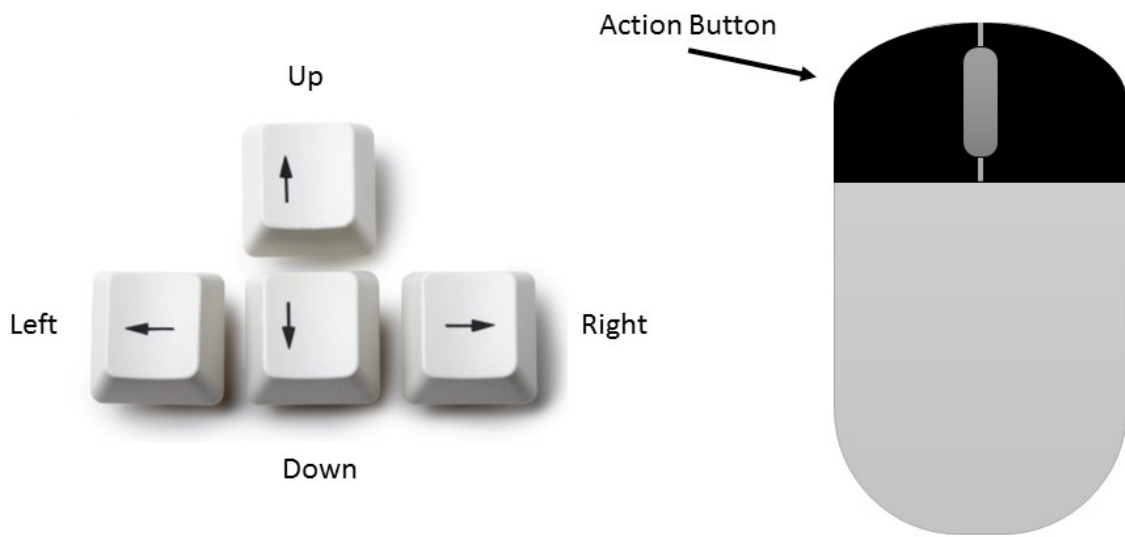




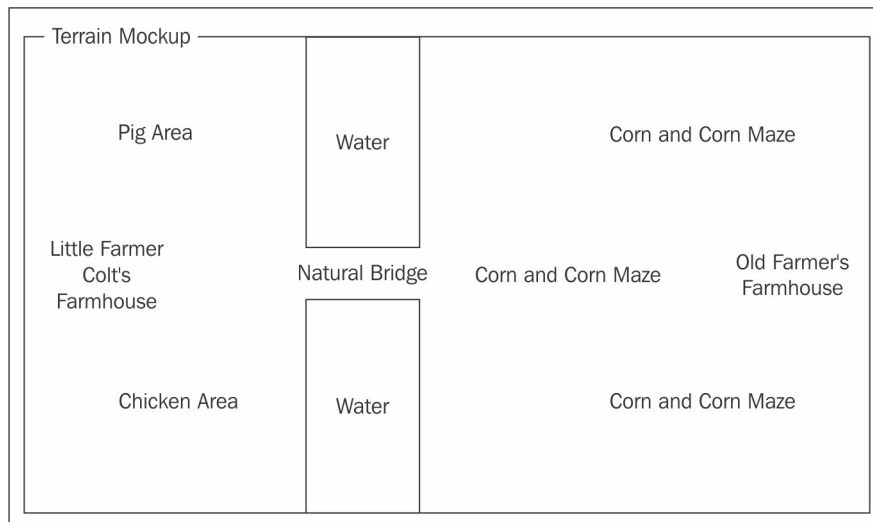


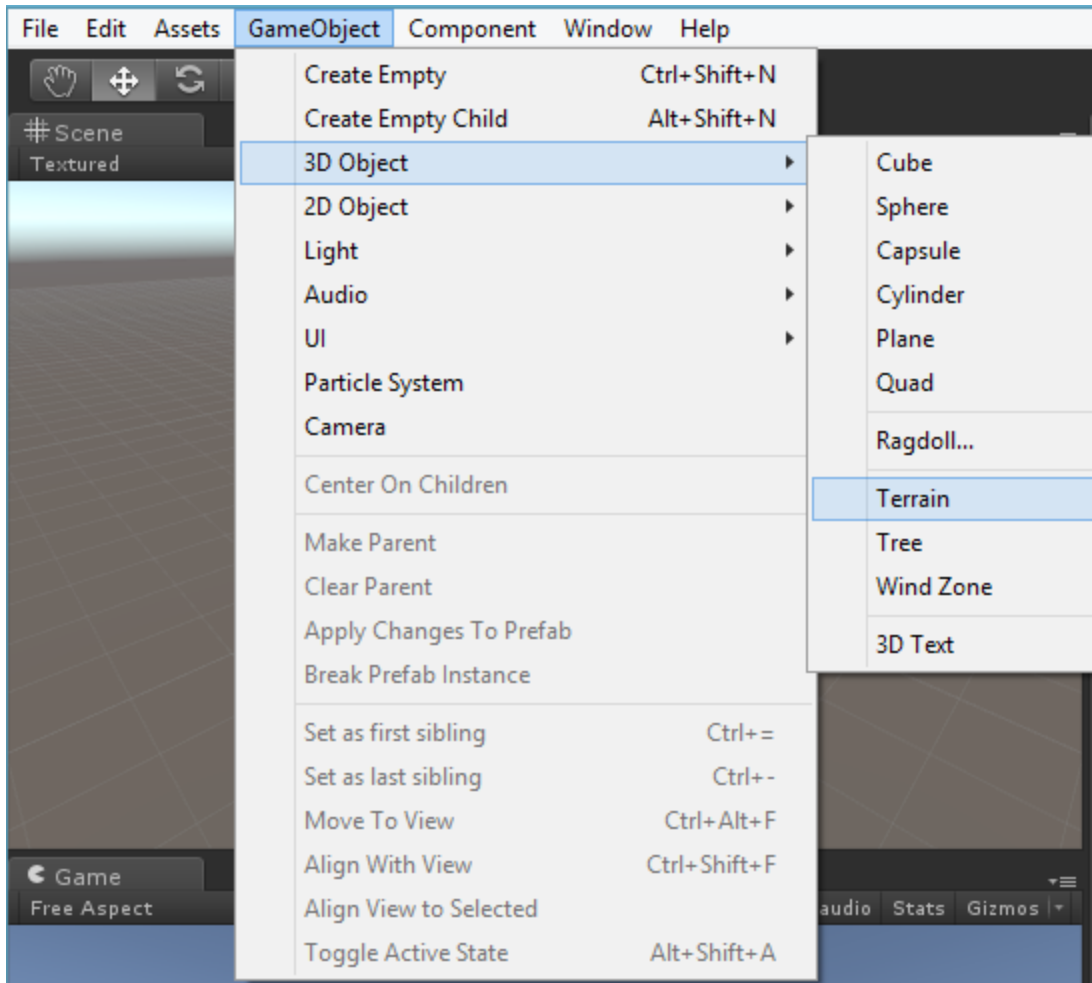


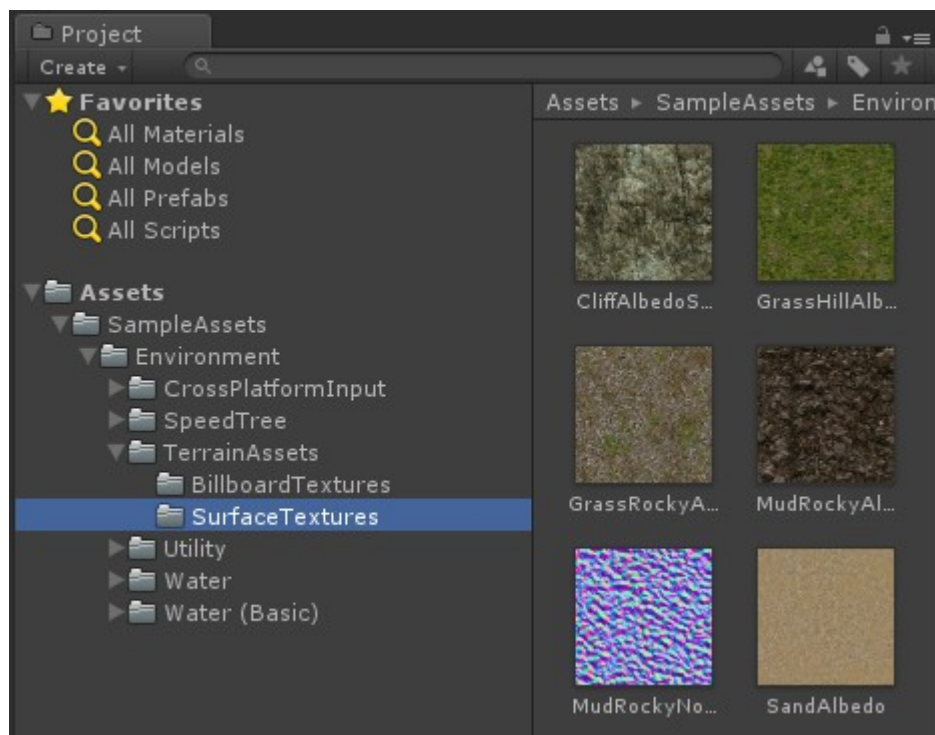
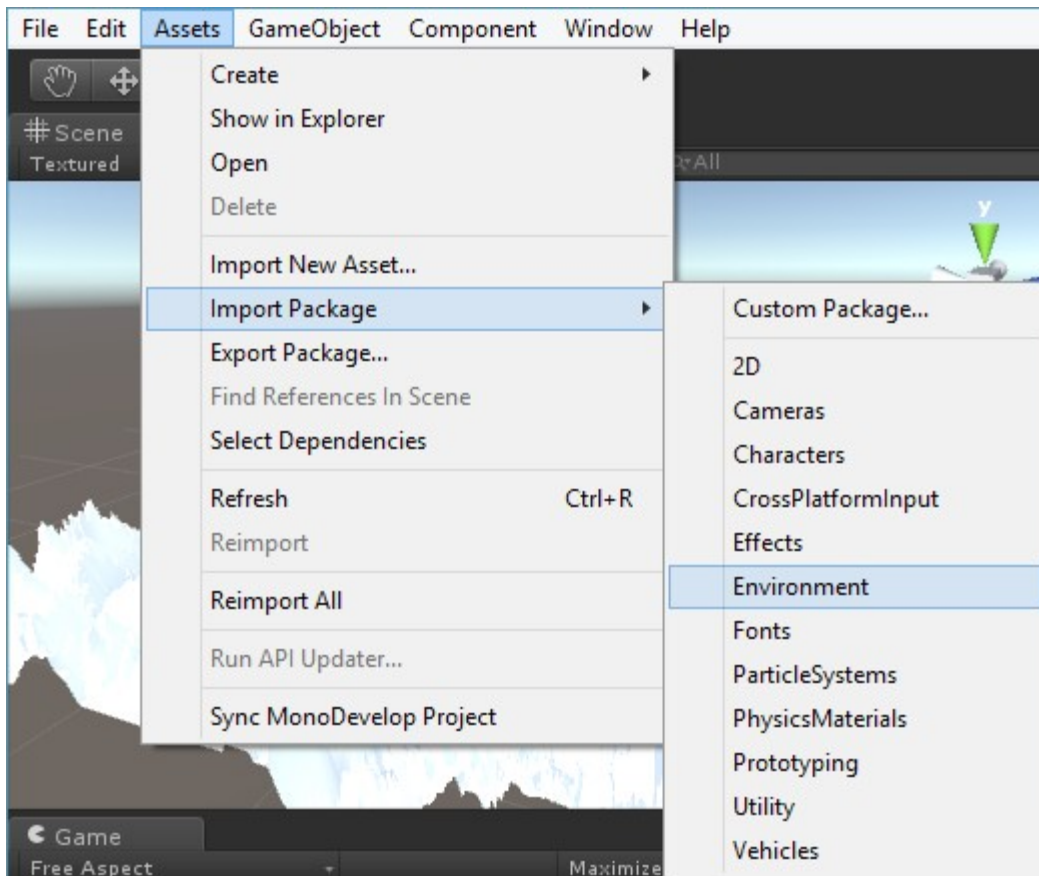
Chapter 2: Creating the Game Environment



Move mouse to rotate character









Inspector

Terrain 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

Terrain

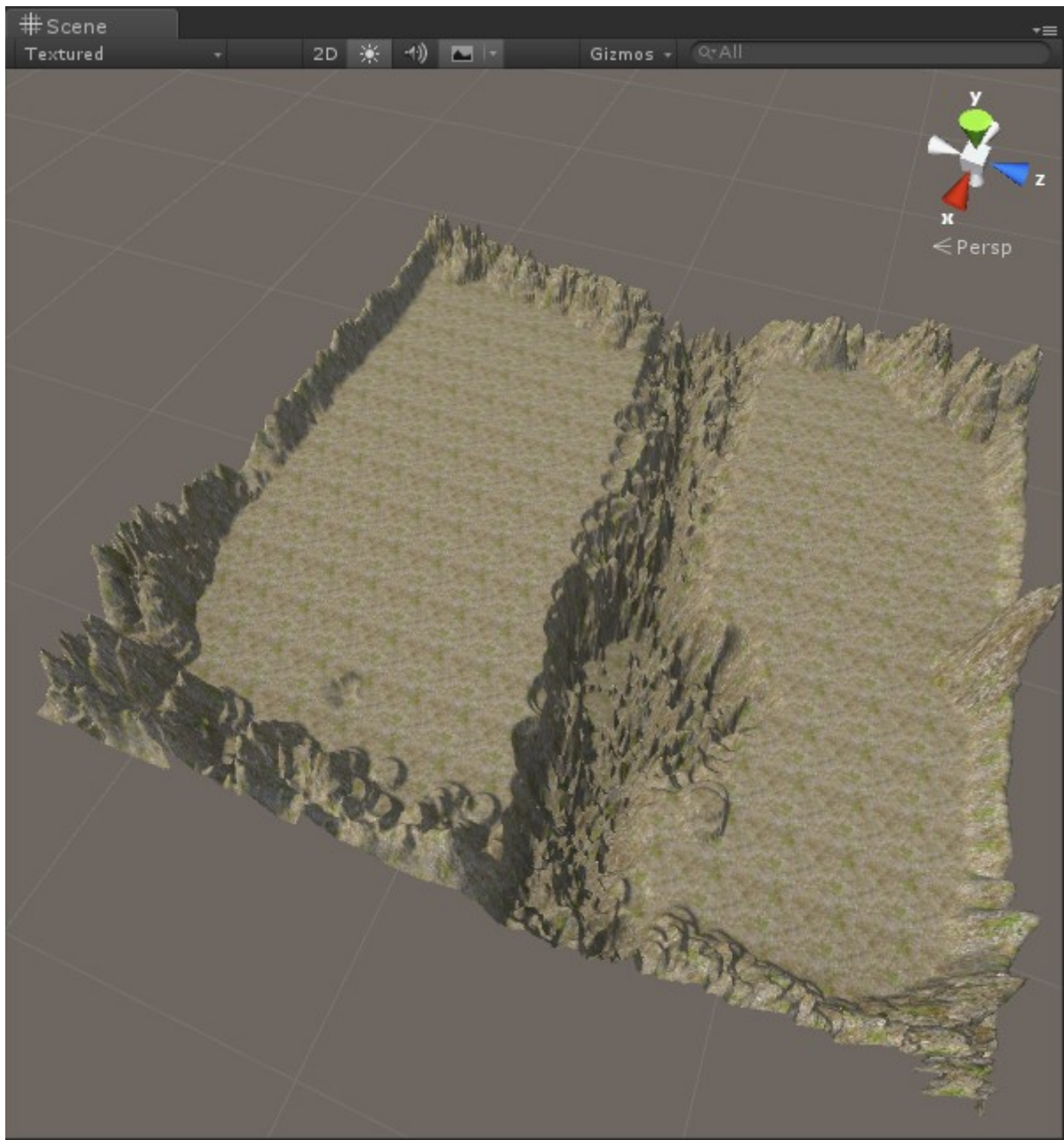
Paint Texture
Select a texture below, then click to paint

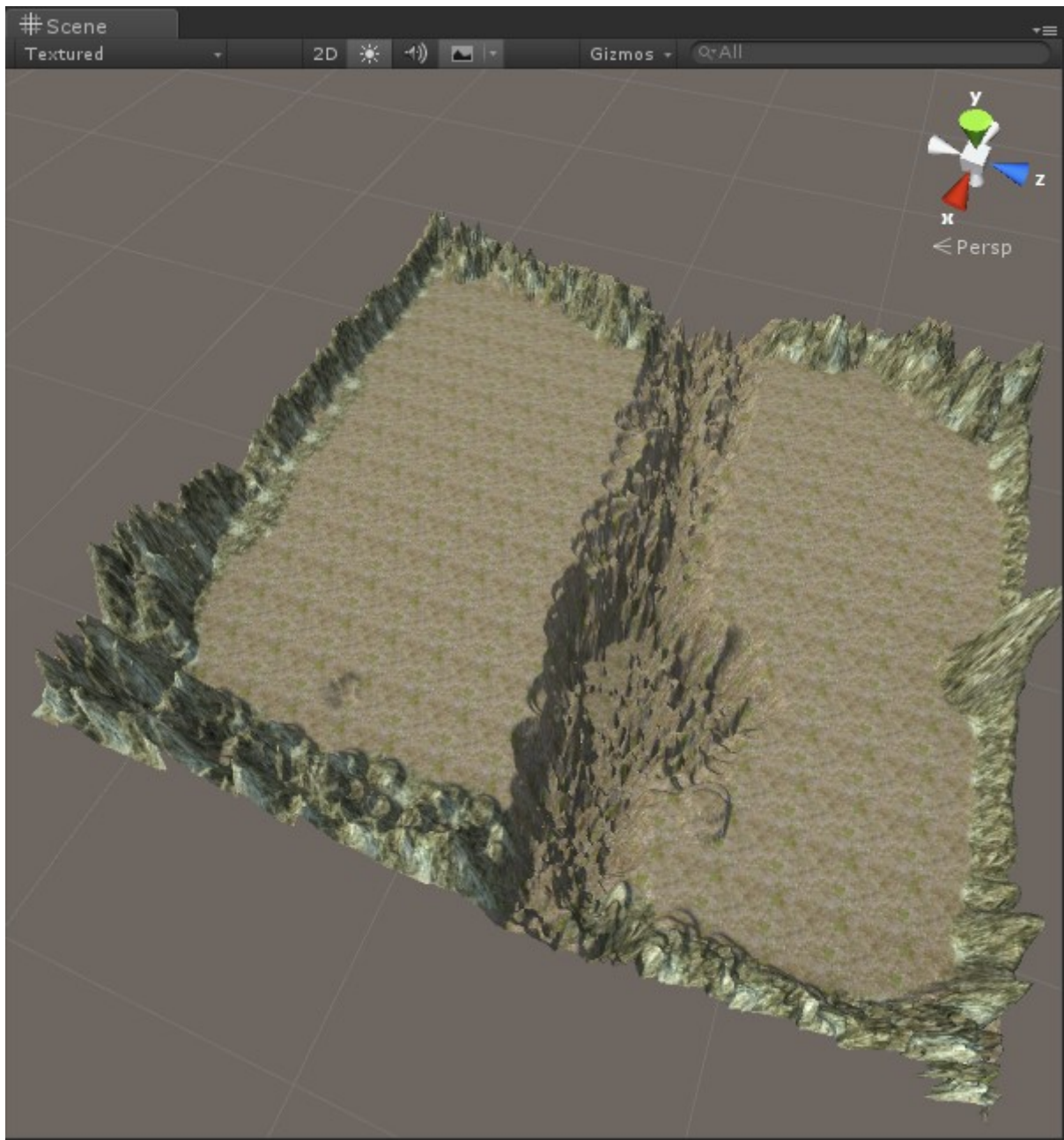
Brushes

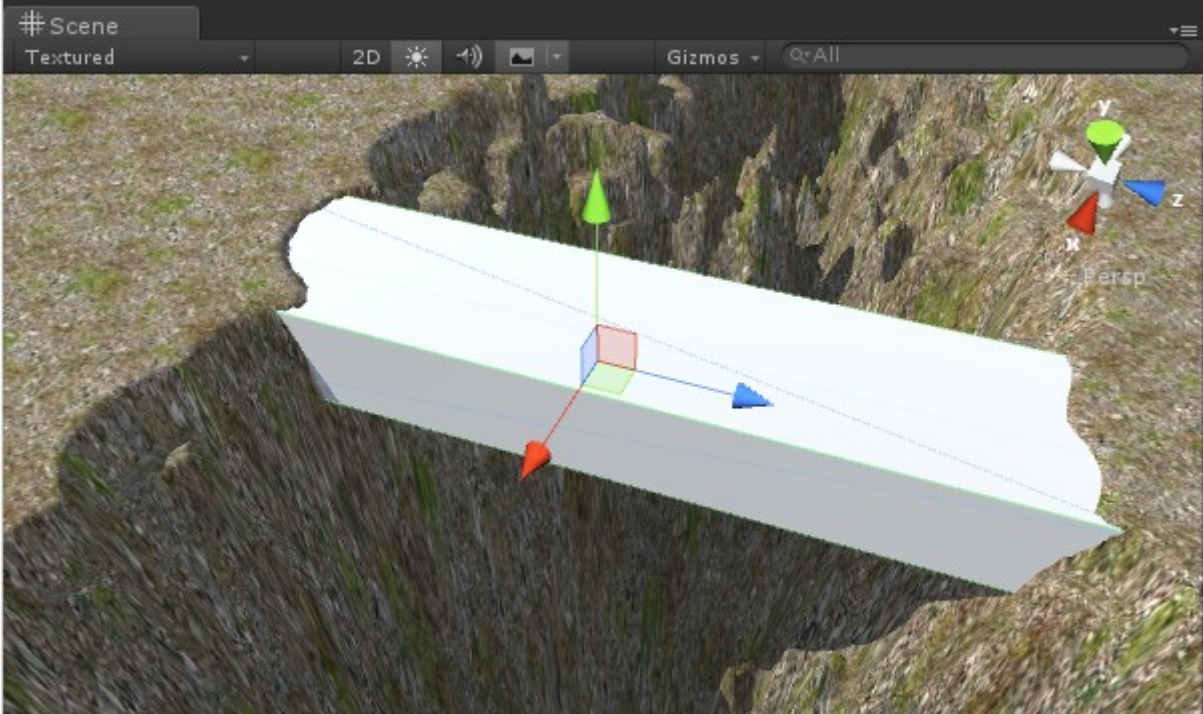
Textures

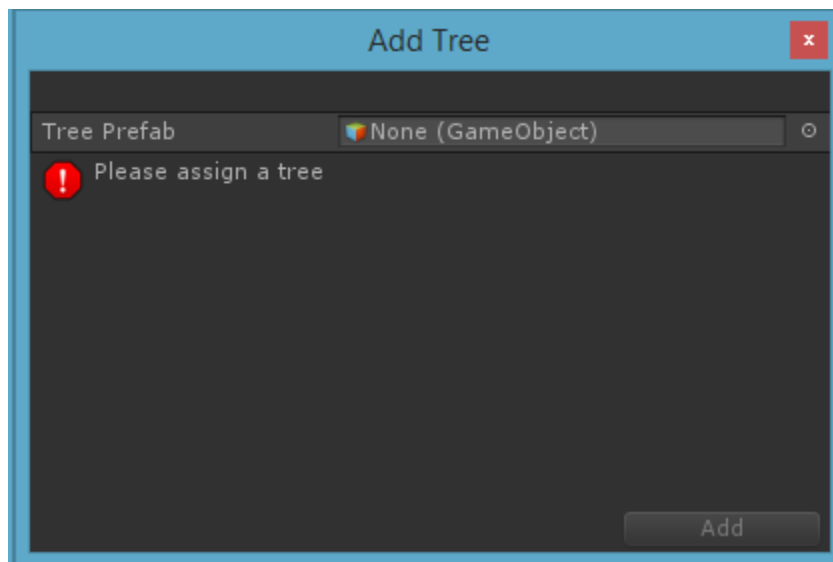
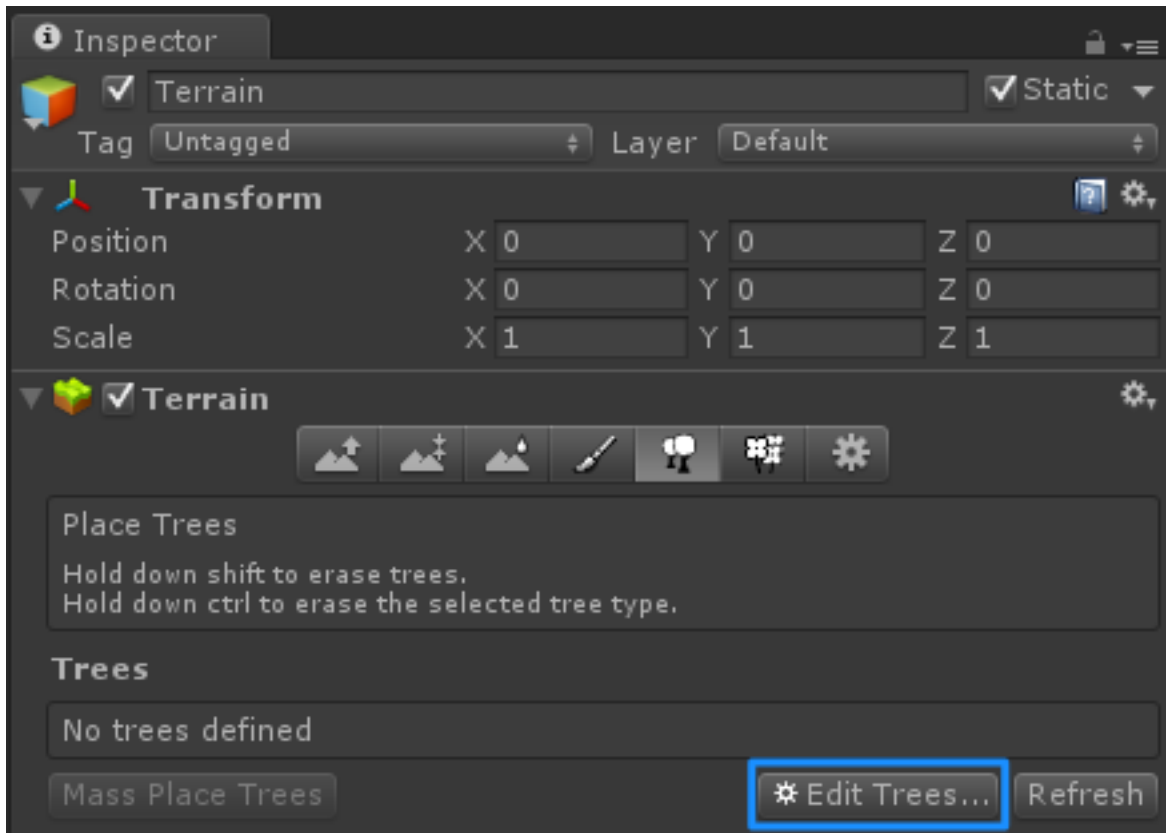
No terrain textures defined.

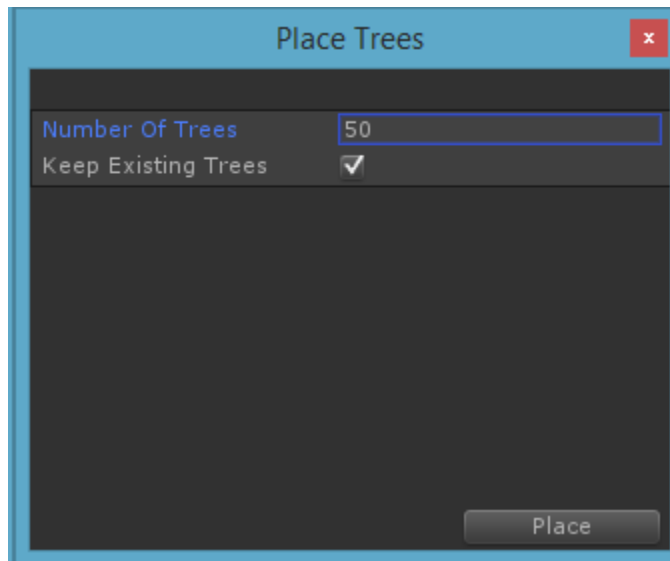
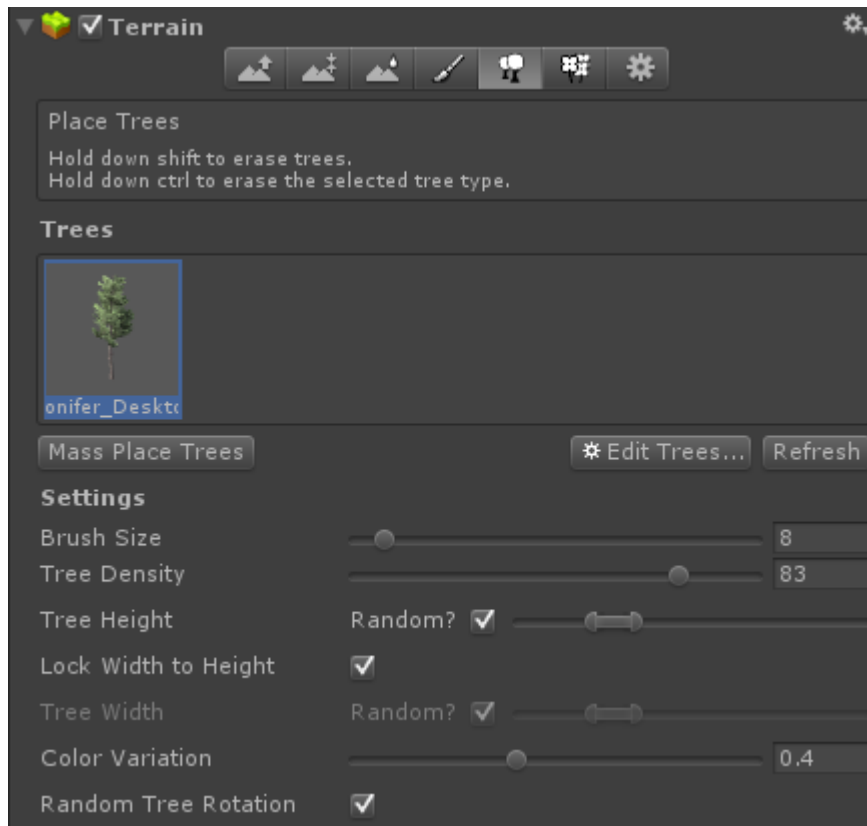
[* Edit Textures...](#)

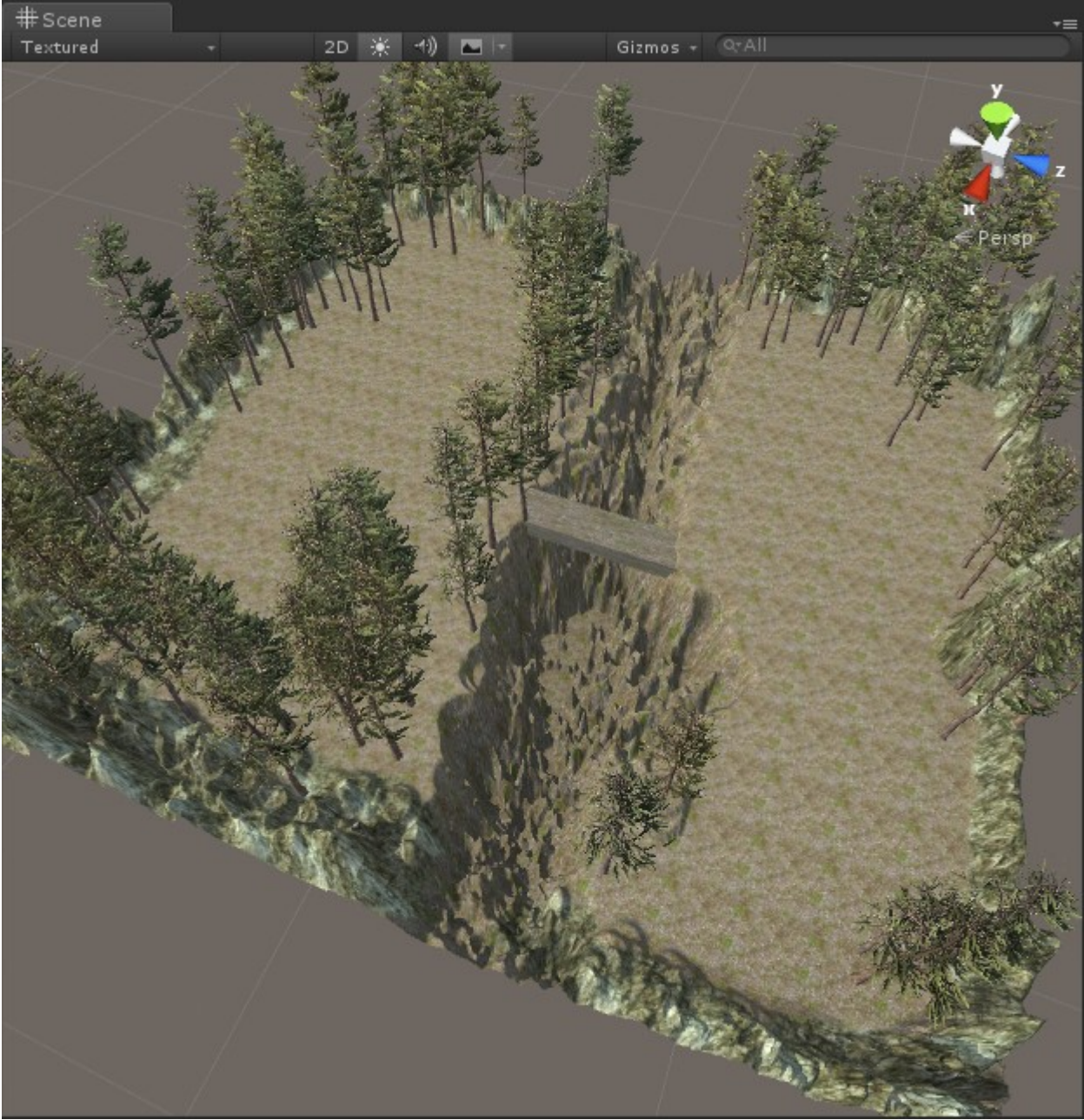




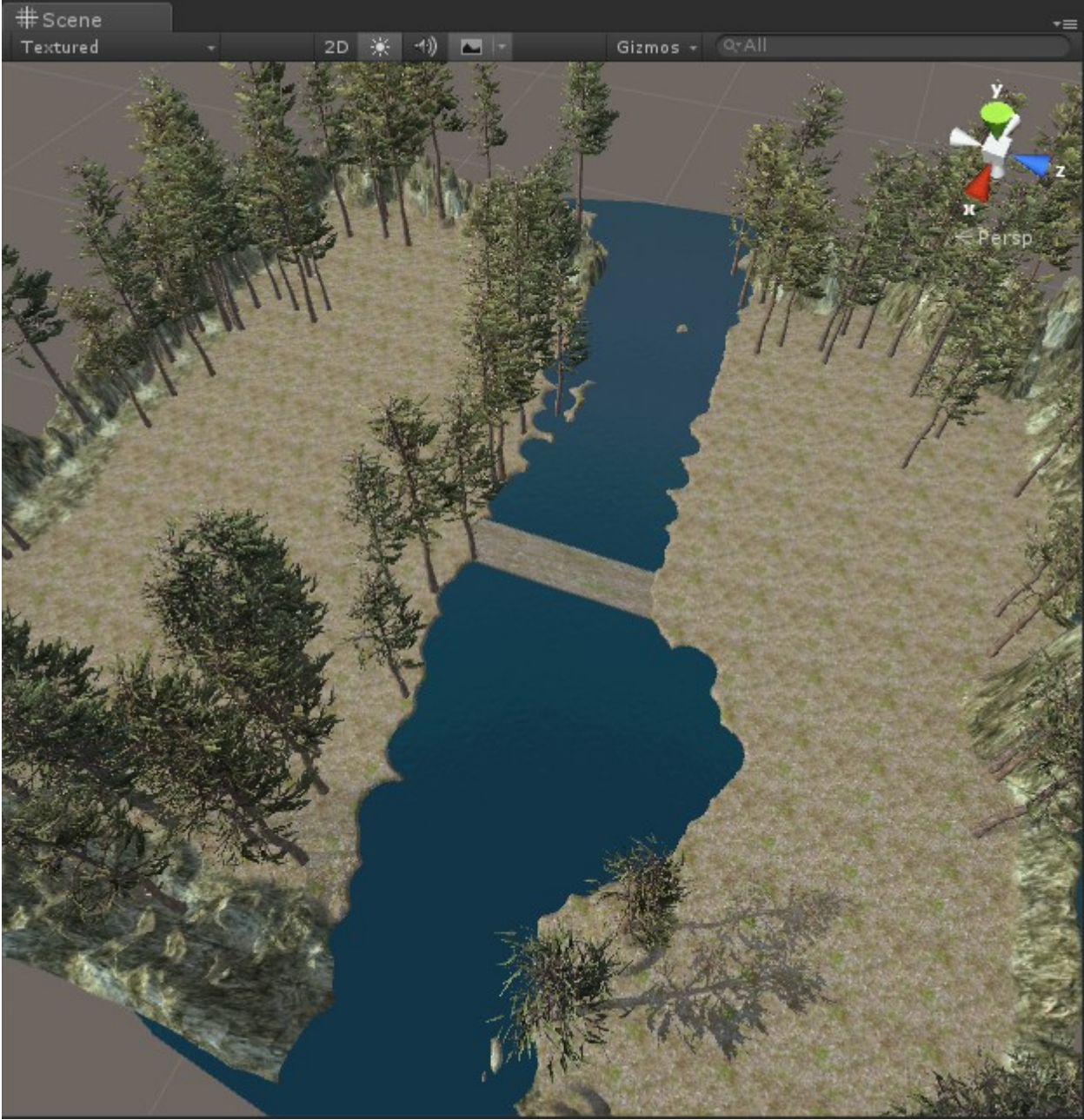


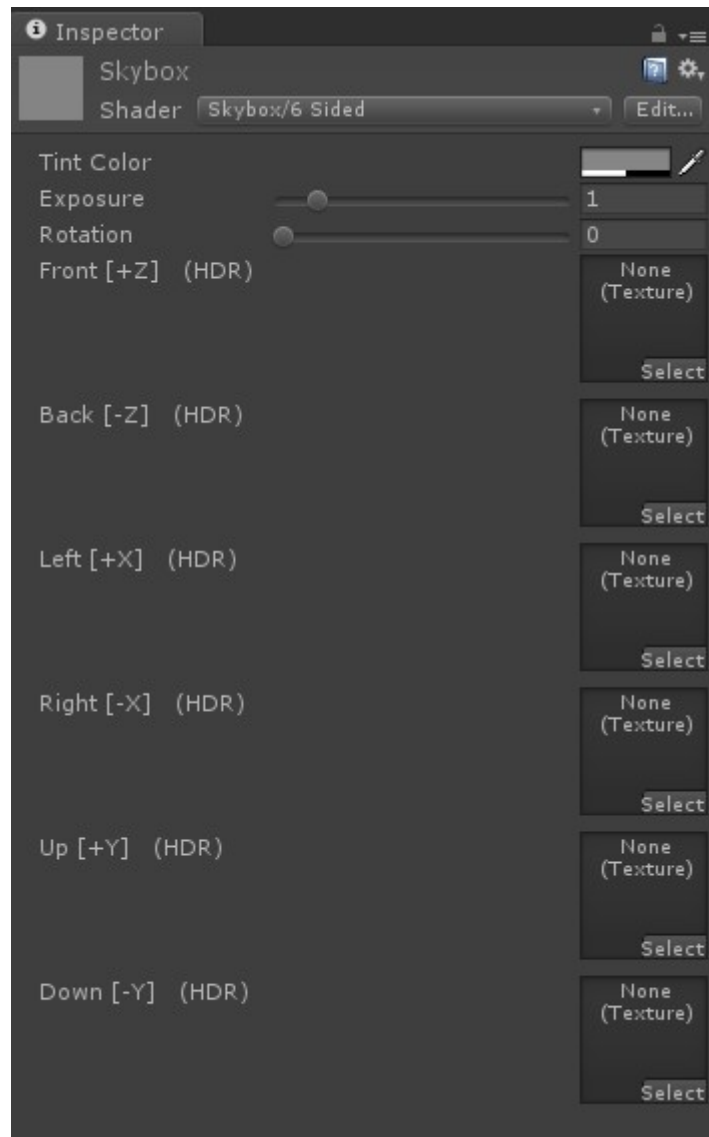


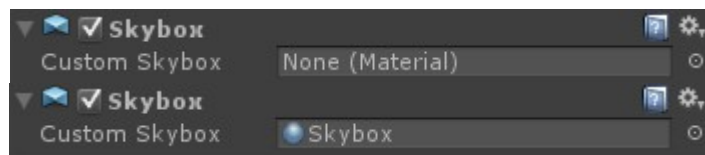
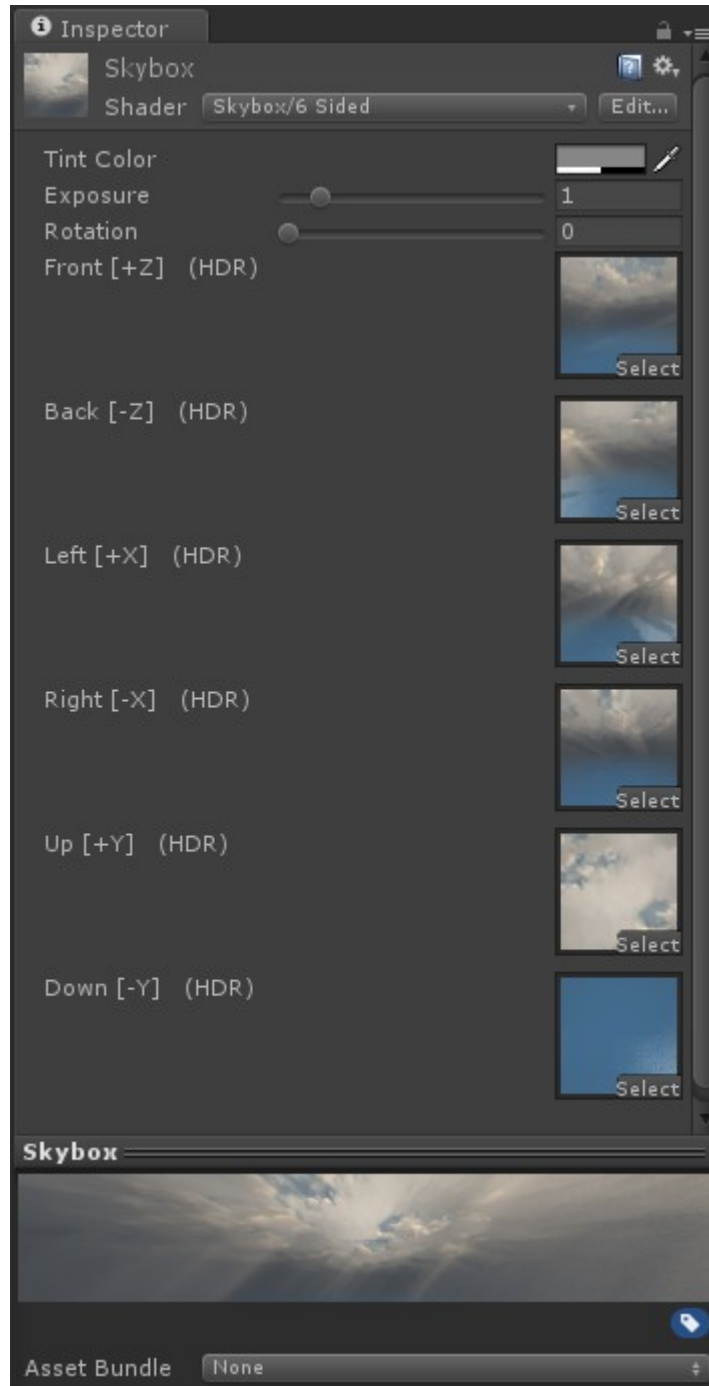




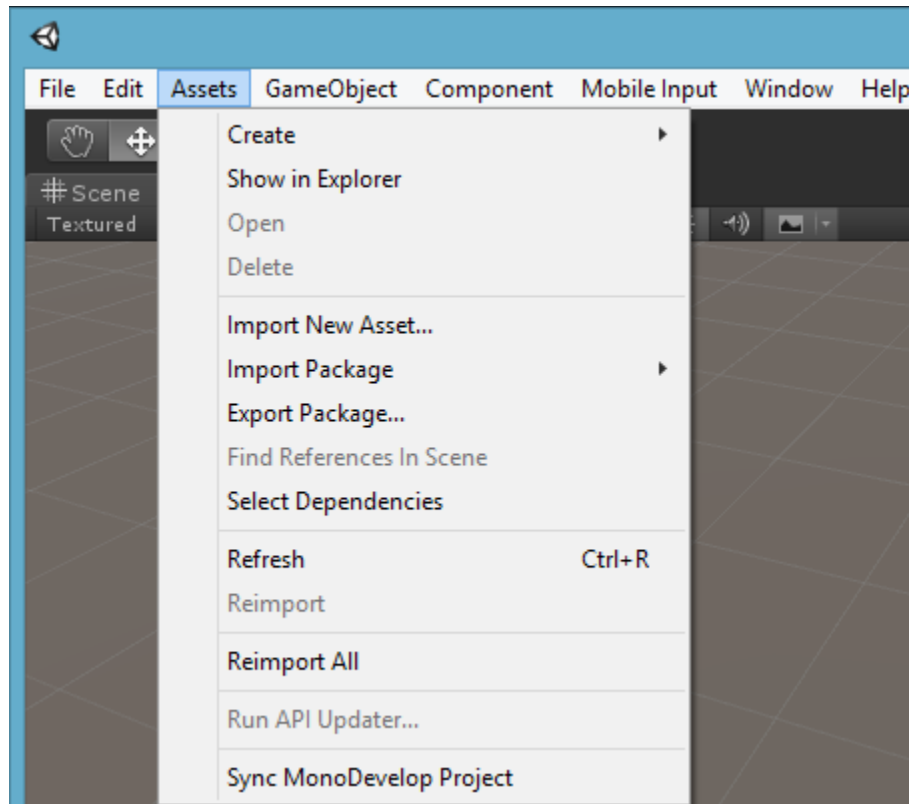


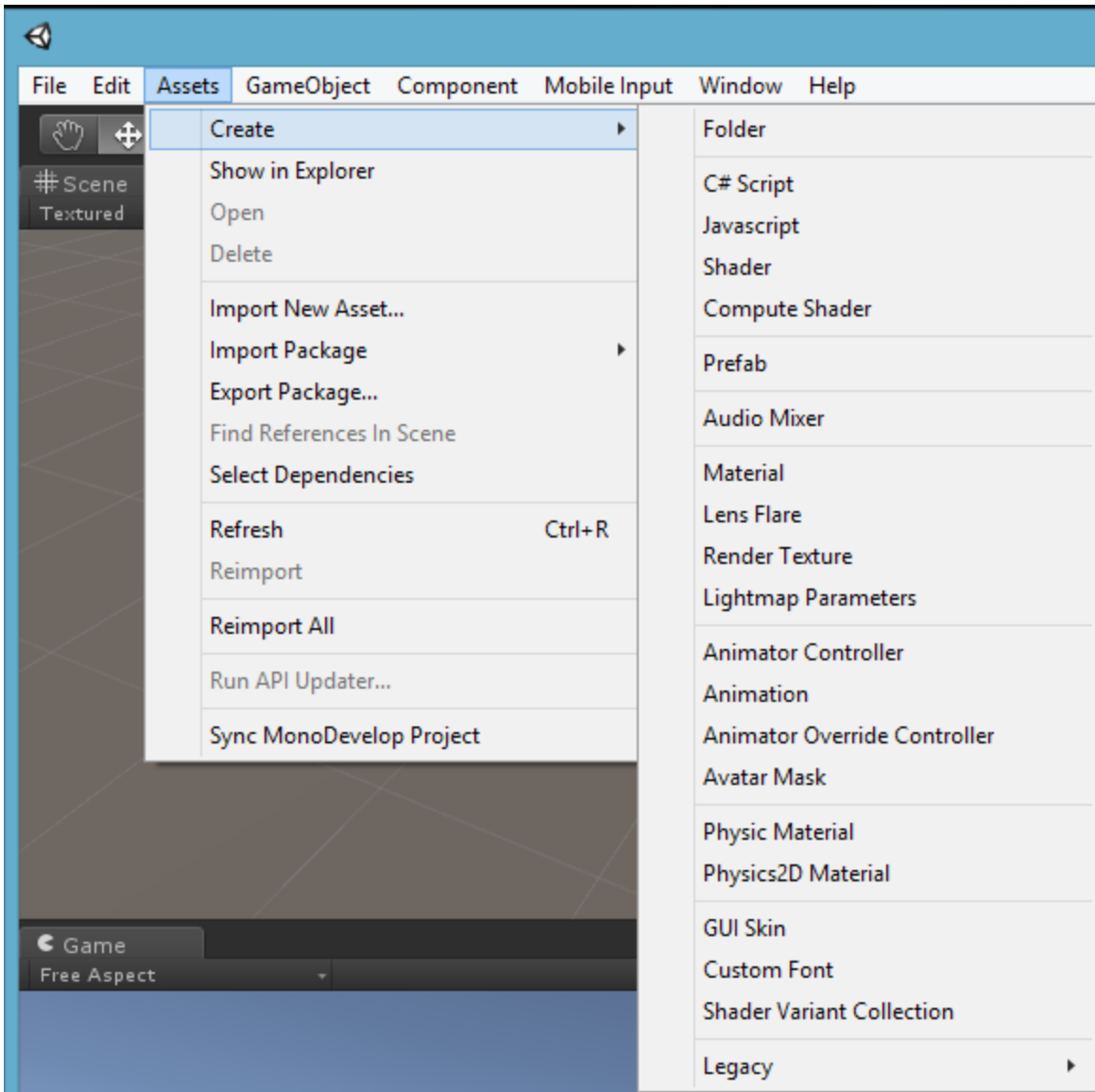


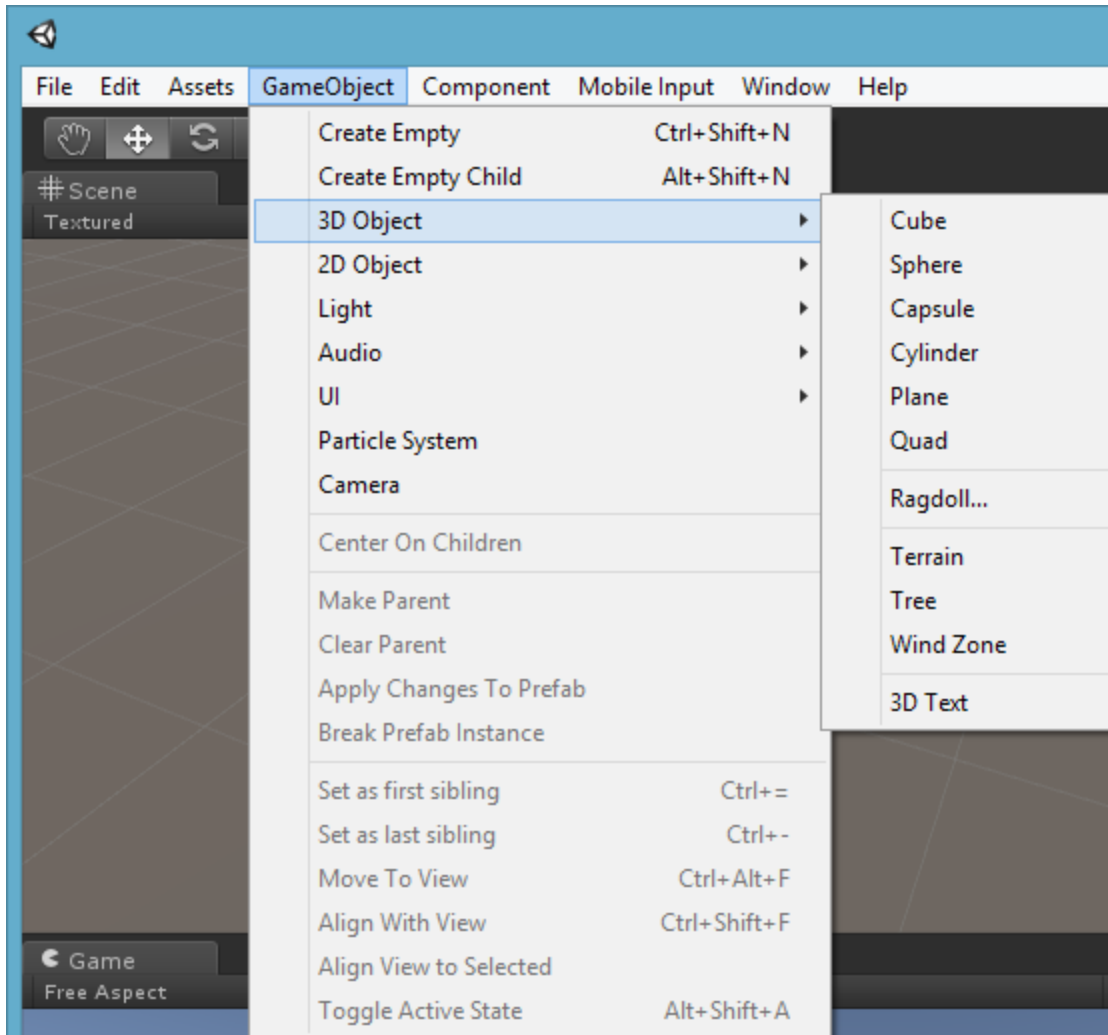


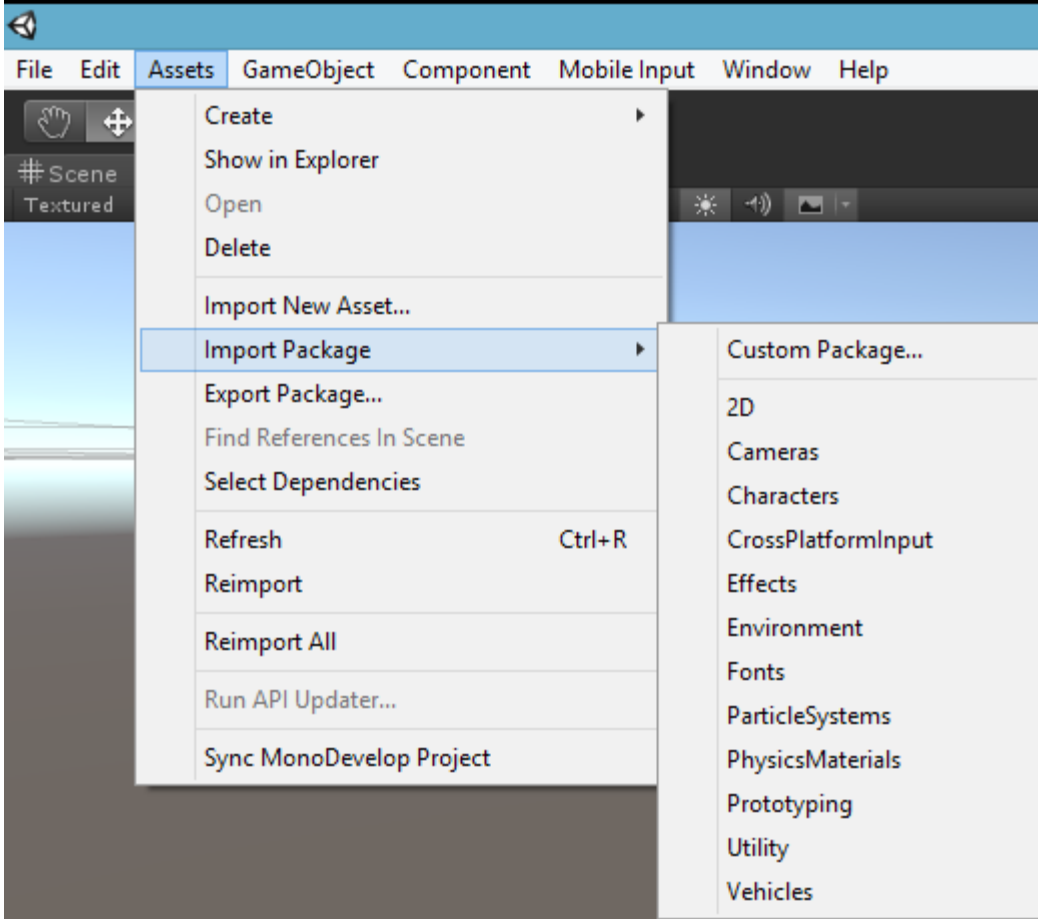


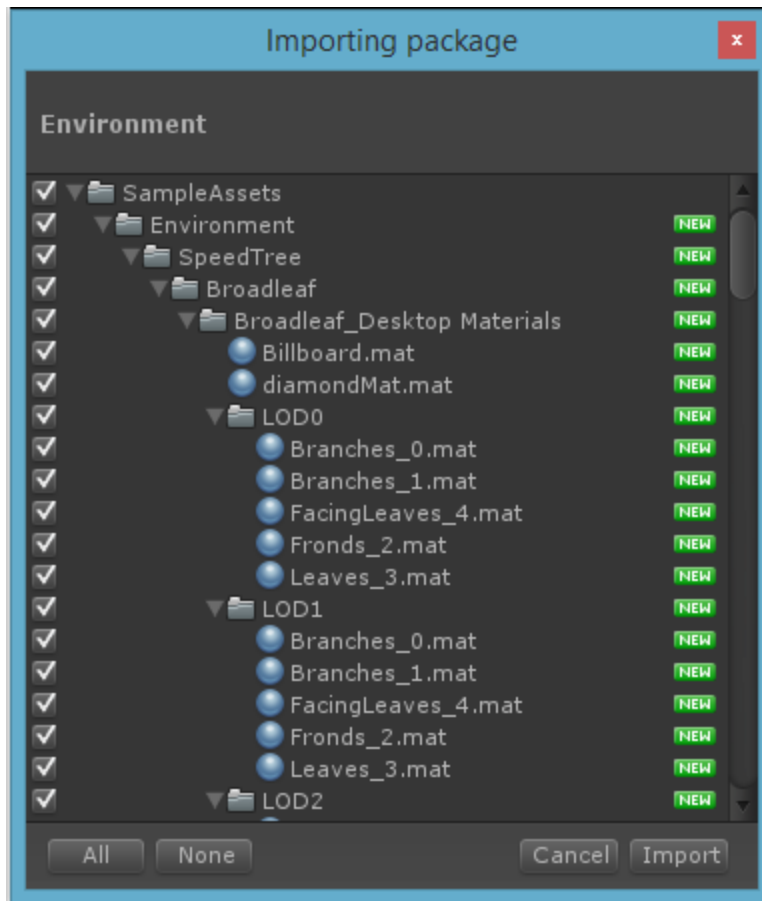
Chapter 3: Working with Assets



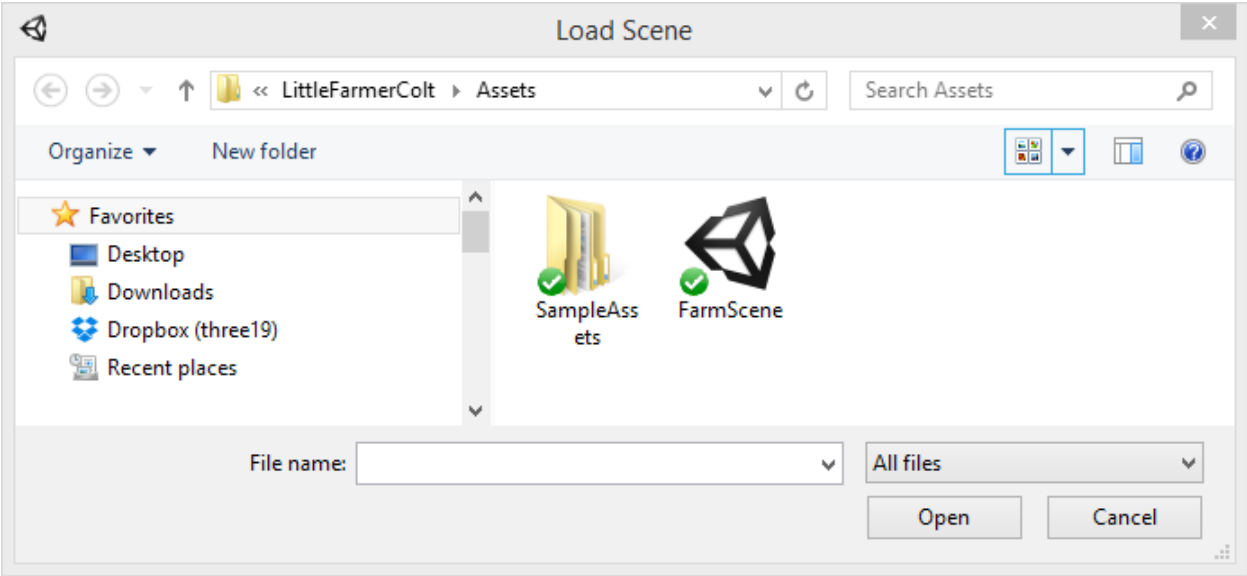








- 🏠 Home
- ▶ 3D Models
- ▶ Animation
- ▶ Applications
- ▶ Audio
- ▶ Complete Projects
- ▶ Editor Extensions
- ▶ Particle Systems
- ▶ Scripting
- ▶ Services
- ▶ Shaders
- ▶ Textures & Materials
- ▶ Unity Essentials












Asset Store

Language: English Edward D Lavieri Jr

Search Results

Search Asset Store

- Home
- 3D Models
- Animation
- Applications
- Audio
- Complete Projects
- Editor Extensions
- Particle Systems
- Scripting
- Services
- Shaders
- Textures & Materials
- Unity Essentials

 Mining Pickaxe 3D Models/Props/Tools Triix3D Not enough ratings \$2	 Pickaxe 3D Models/Props/Tools Triix3D Not enough ratings \$2	 Pickaxe 3D Models/Props/Tools Unity Technologies ★★★★★ (35) Free
 Pickaxe 3D Models/Props/Weapo... artware Not enough ratings \$5	 Excavation Tool SFX (Dig... Audio/Sound FX/Foley WelchCompositions Not enough ratings \$15	 Minion 3D Models/Characters/H... Claudiu Not enough ratings \$10
 Power-Ups & Pick-Ups (P... Textures & Materials/Icon... Playelectric ★★★★★ (4) \$20	 High Quality Weapons T... 3D Models/Props/Weapo... SummumArt Not enough ratings \$10	 Power-Ups & Pick-Ups (P... Textures & Materials/Icon... Playelectric ★★★★★ (4) \$10

Pickaxe

[Add to Wish List](#)

Category: 3D Models/Props/Tools
Publisher: Unity Technologies
Rating: ★★★★★ (135)
Price: Free

[Download](#) [Share](#) [Twitter](#) [Facebook](#) [StumbleUpon](#)

A pickaxe is a simple hand tool with a hard head attached to a handle. Useful in rural settings such as farms, mines and archaeological sites.
Public Domain.

SFC Paul Ray Smith Simulation & Training
Technology Center
Courtesy of ADL repository.



Version: 1.0 (Jun 03, 2011) Size: 138.8 kB

[Visit Publisher's Website](#)

Package Contents

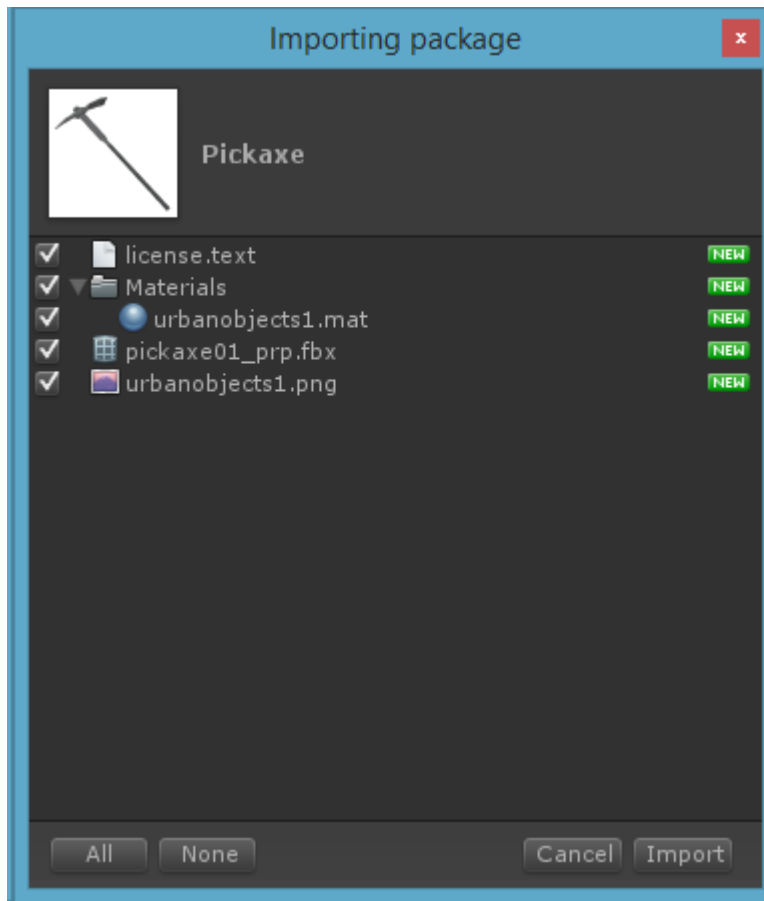
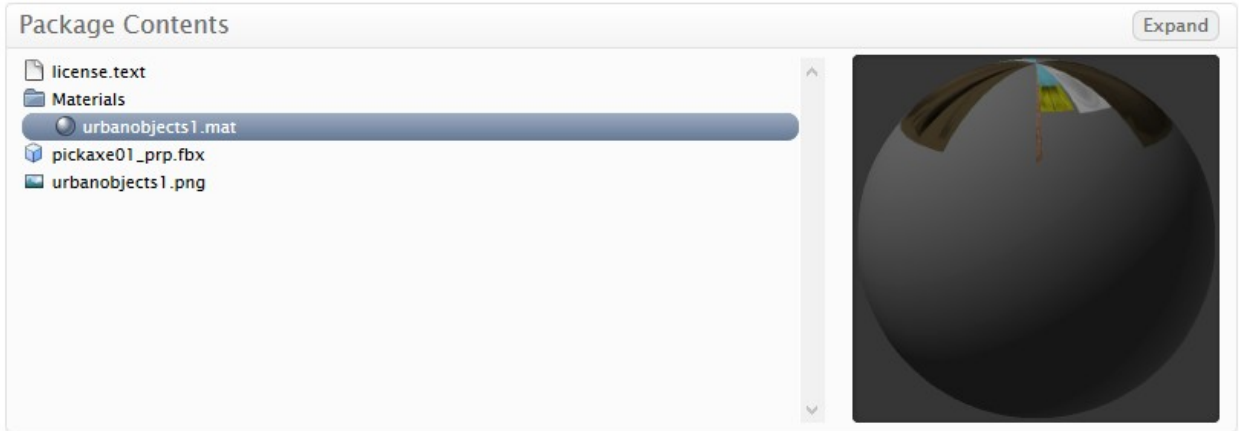
[Expand](#)

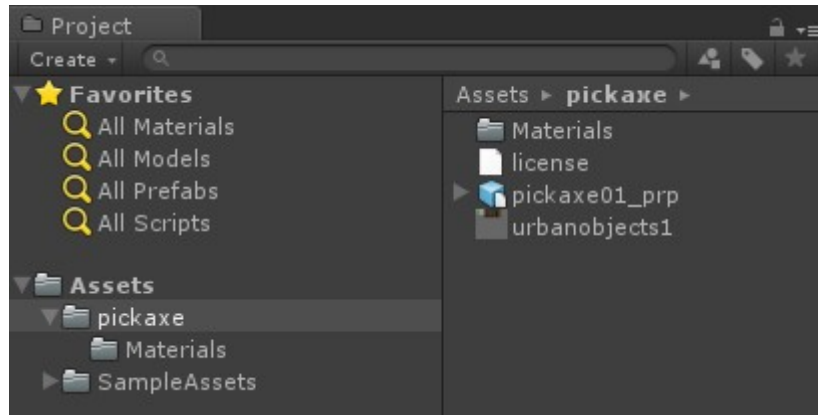
- license.text
- Materials
 - urbanobjects1.mat
- pickaxe01_prp.fbx
- urbanobjects1.png



Version: 1.0 (Jun 03, 2011) Size: 138.8 kB

[Visit Publisher's Website](#)





Chapter 4: Animating the Game Characters

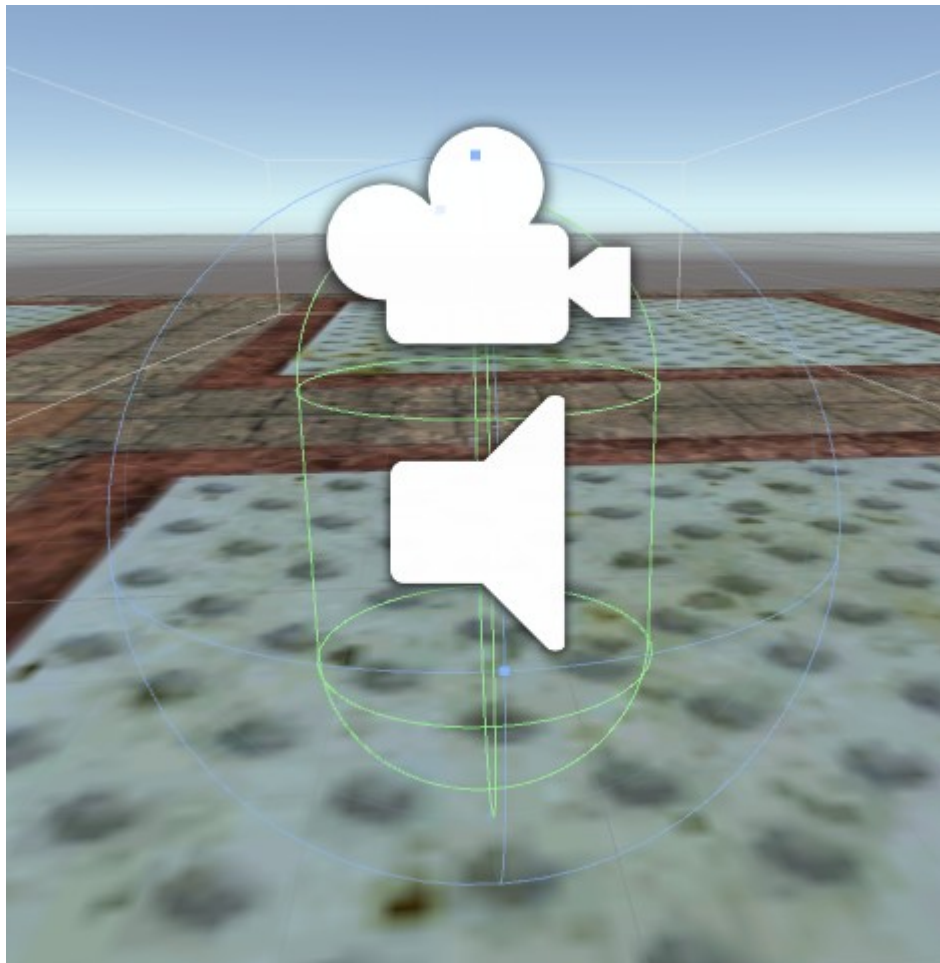
Player Perspectives



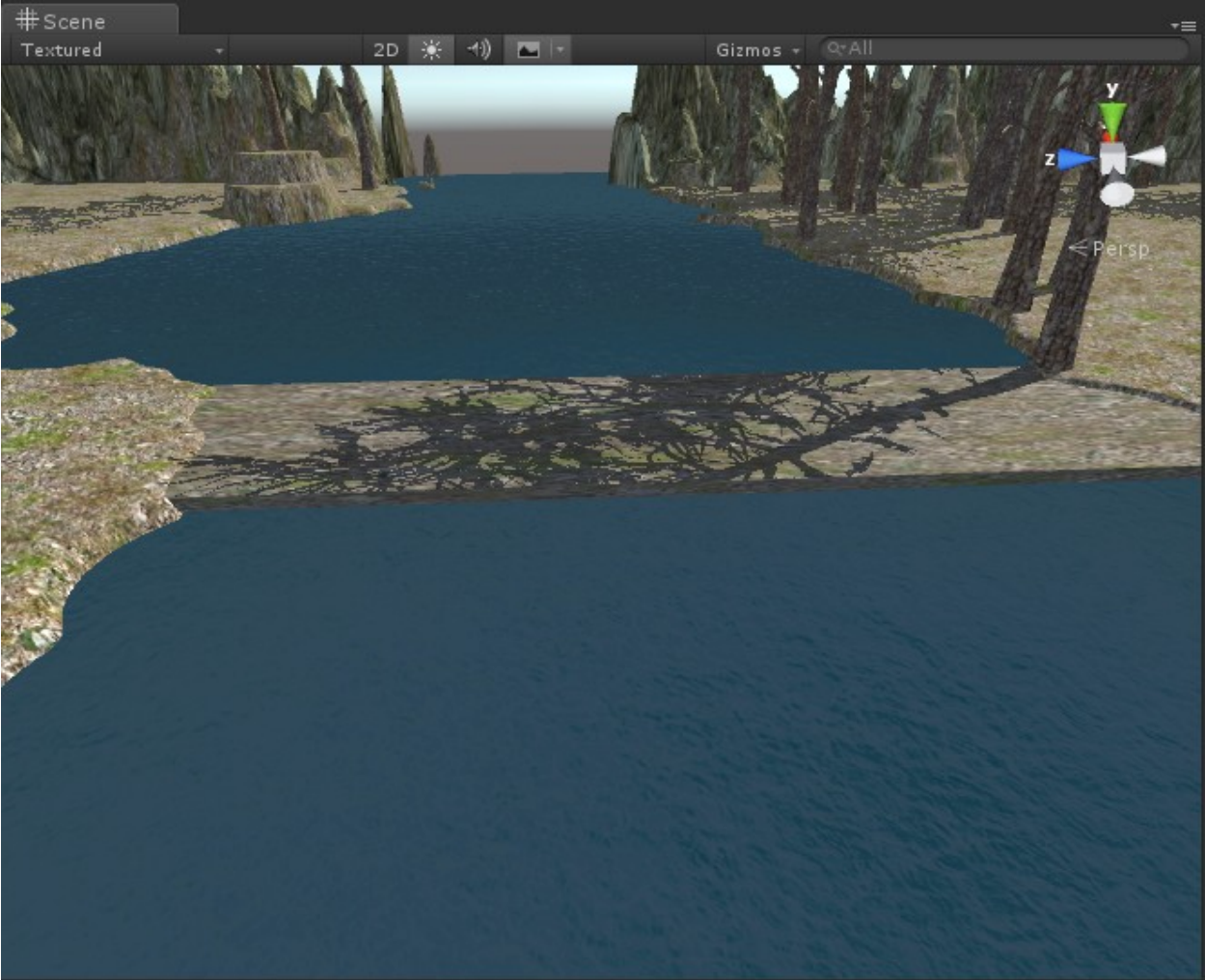
First Person

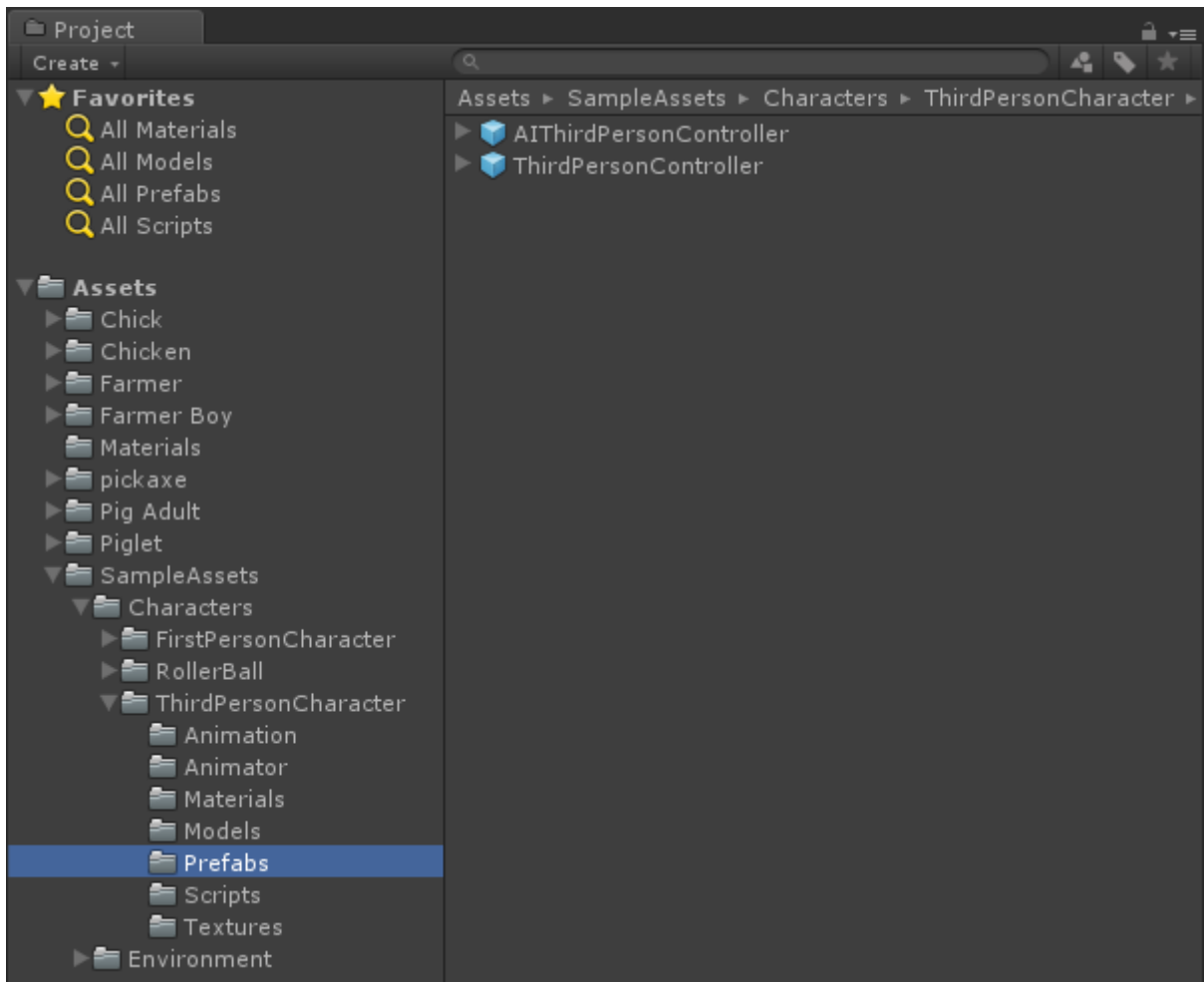


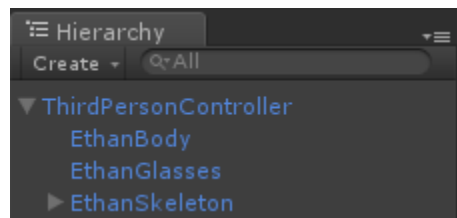
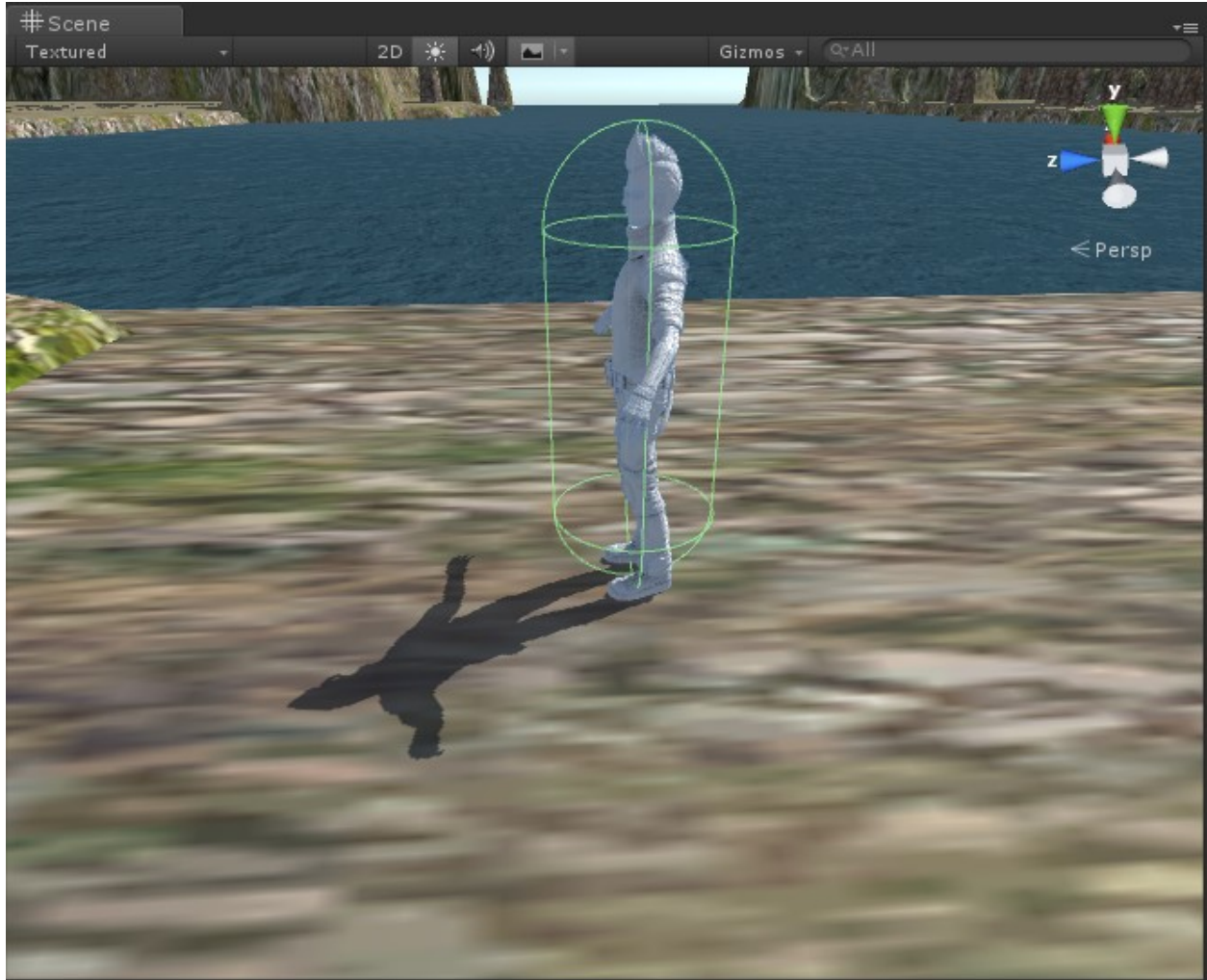
Third Person

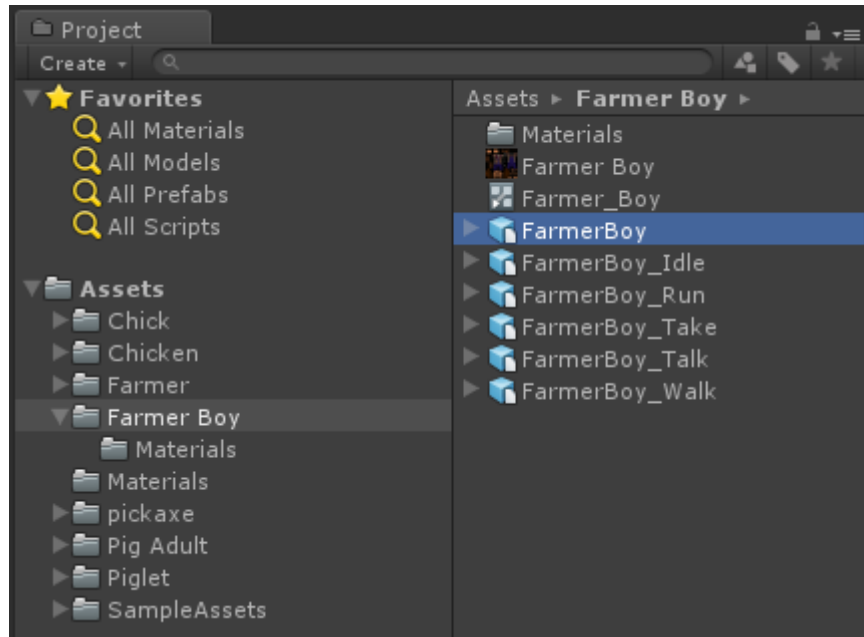




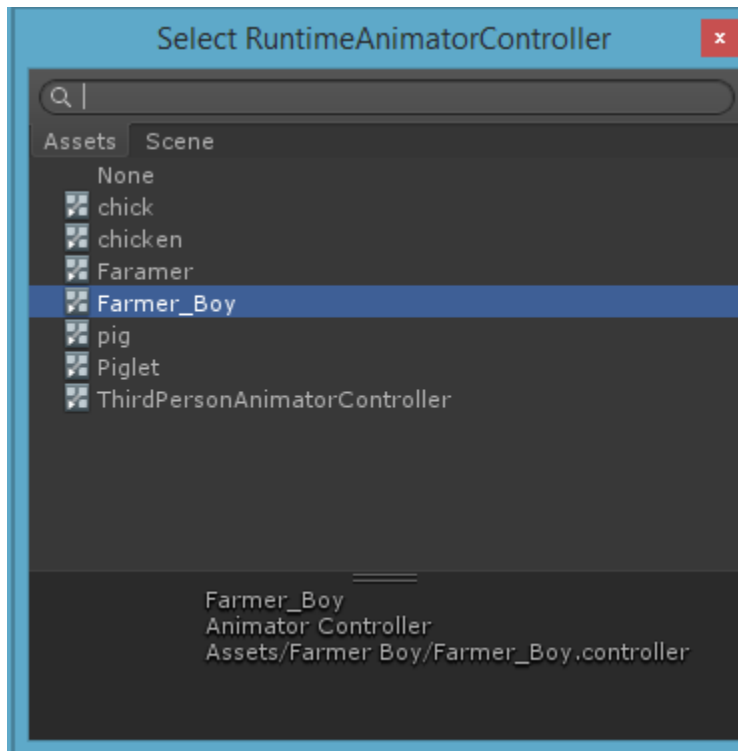
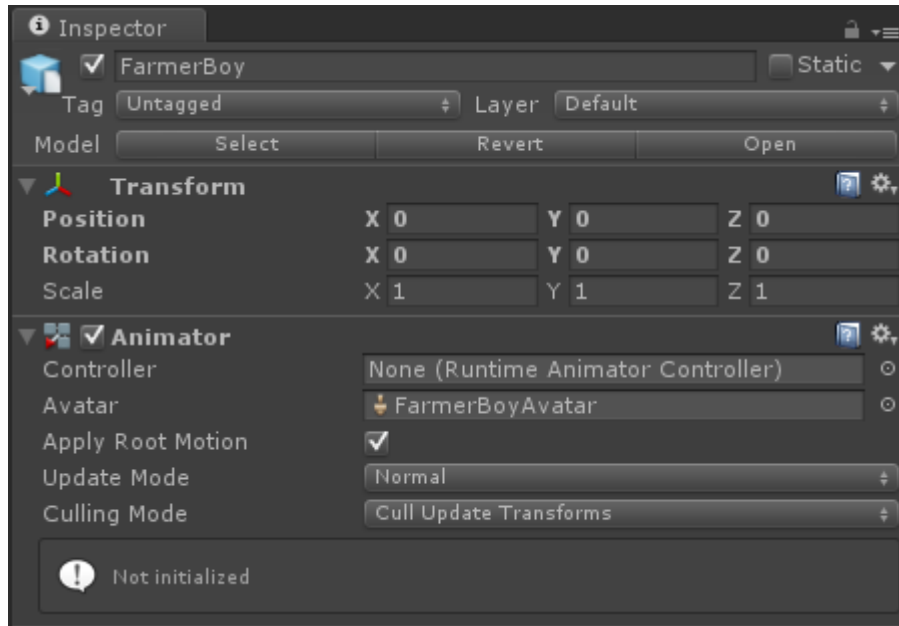


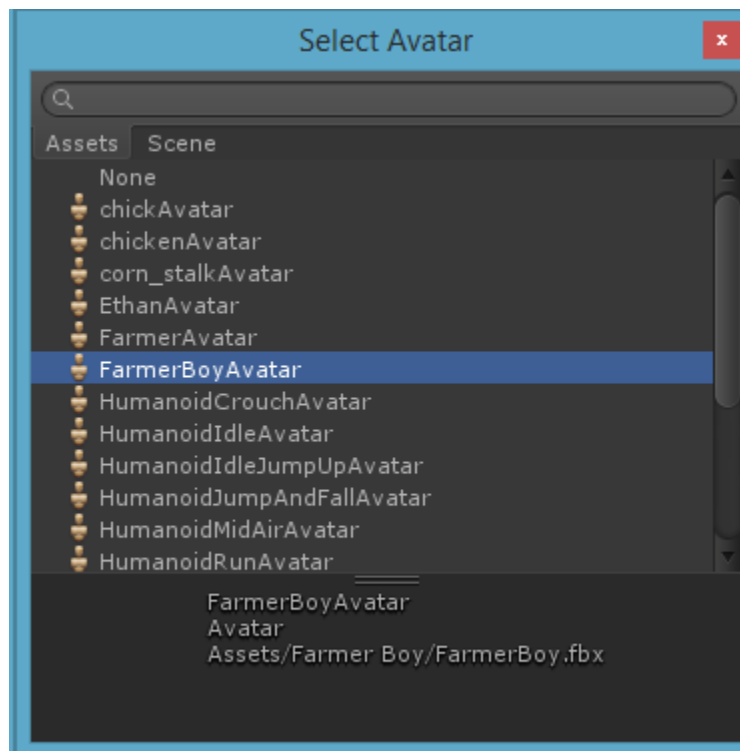
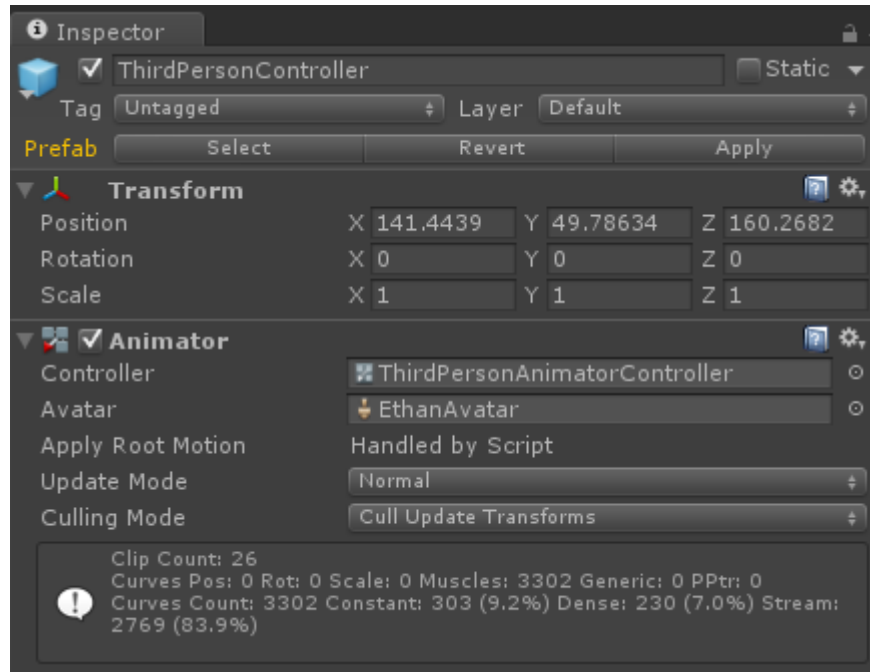


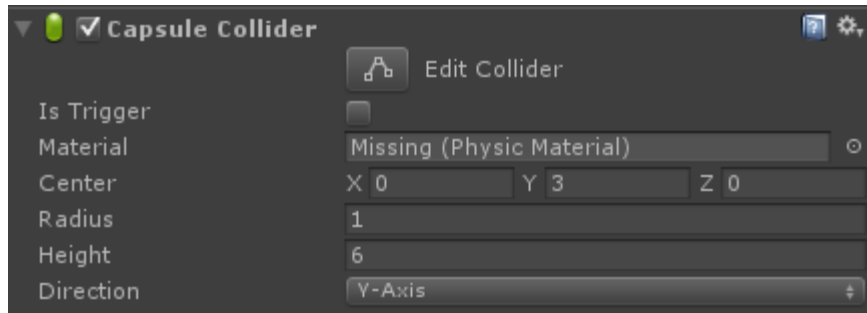


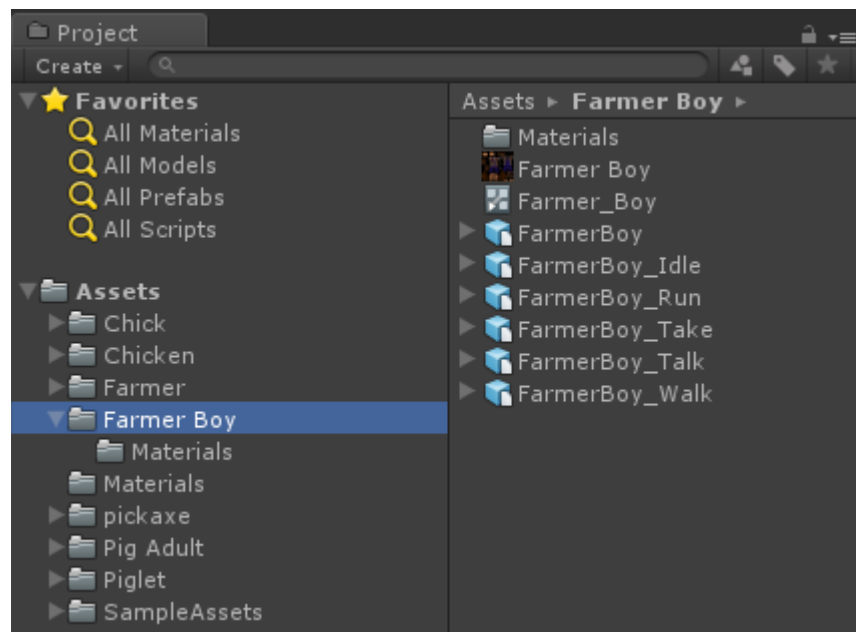


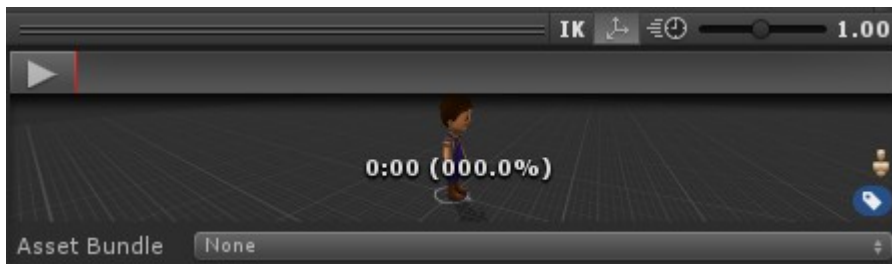
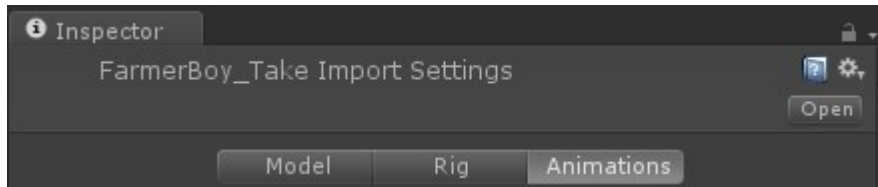


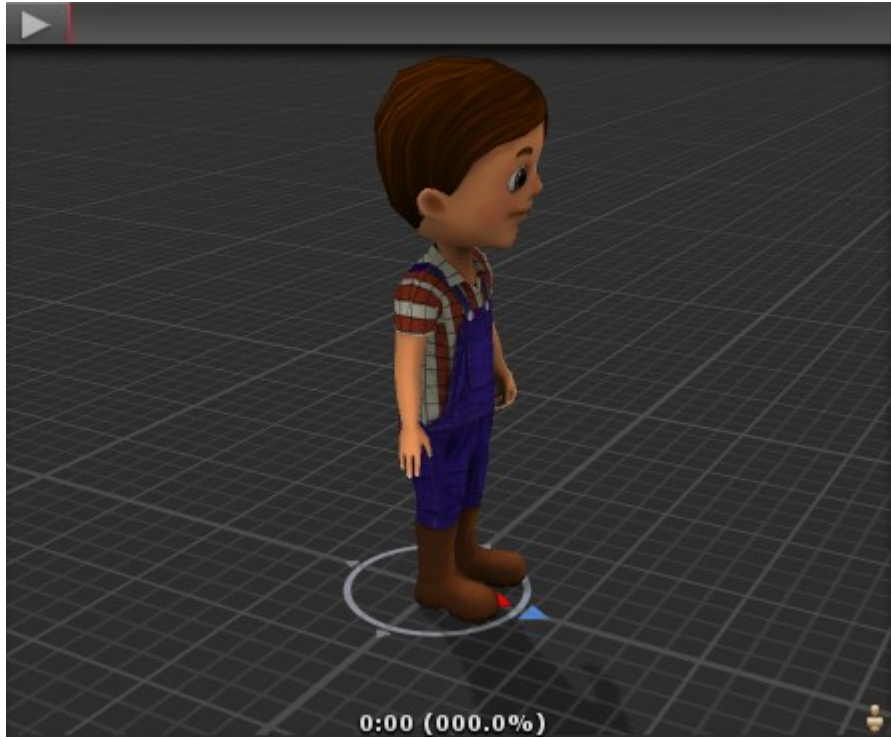


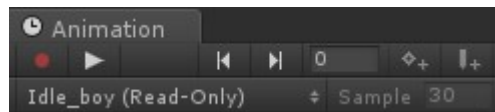
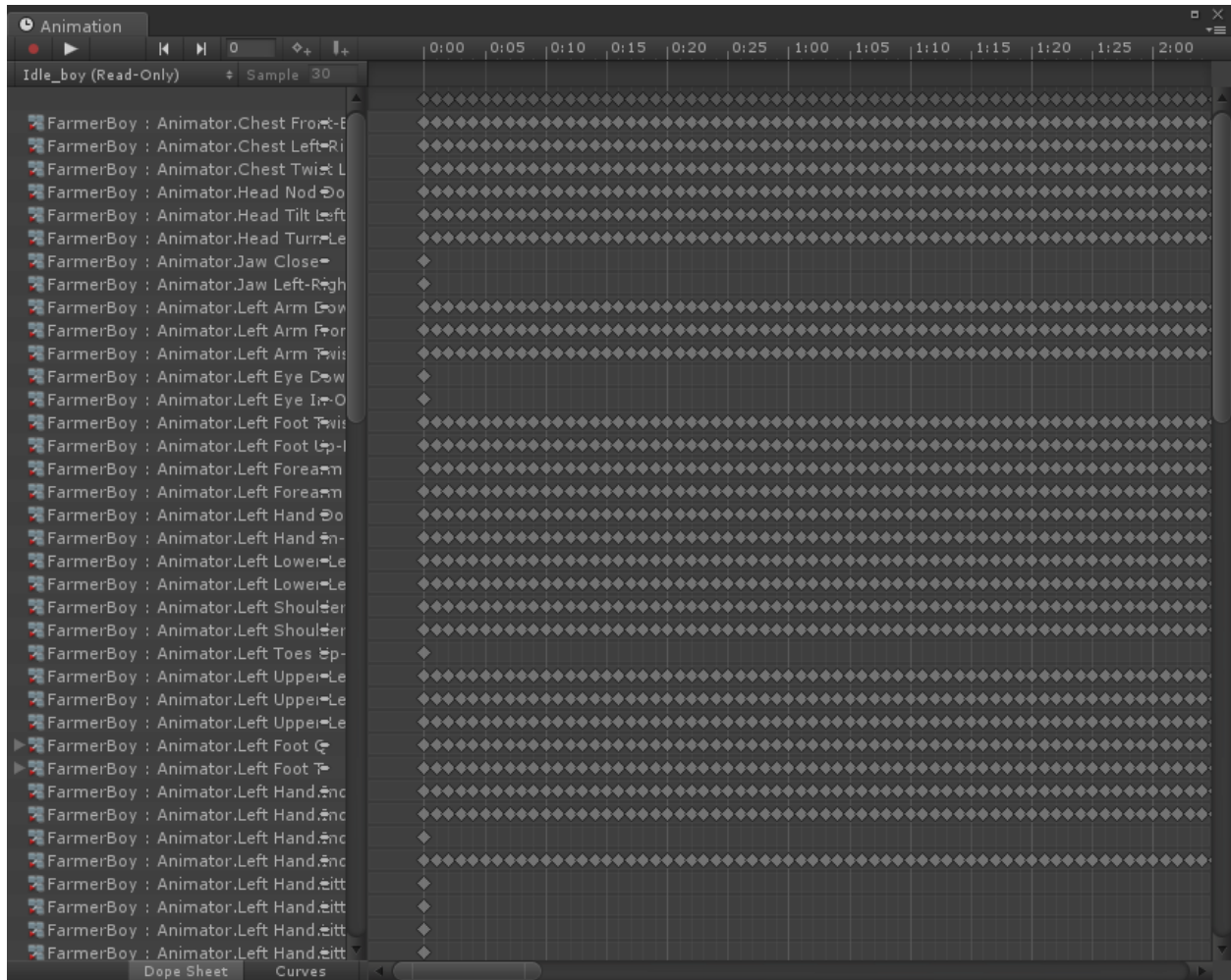




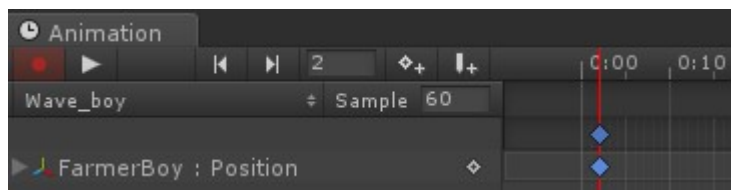
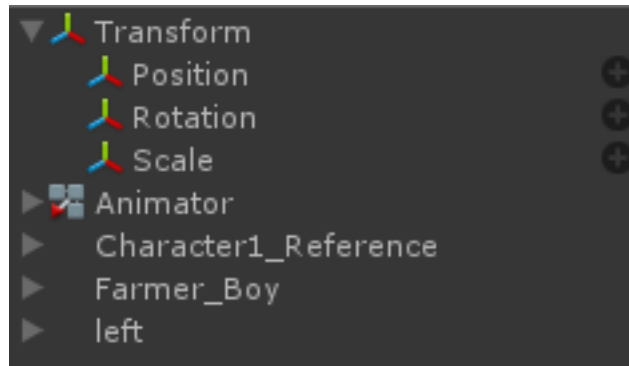
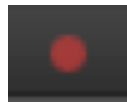
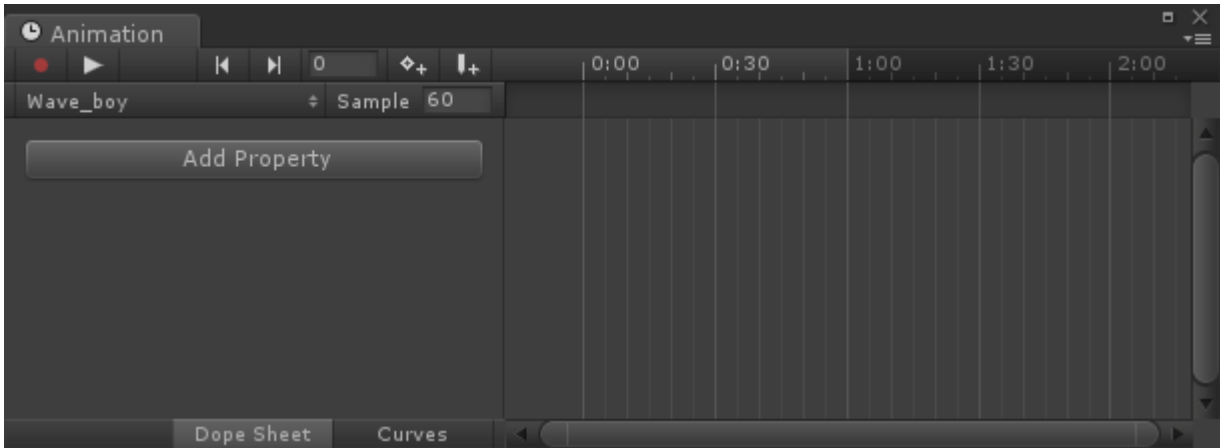


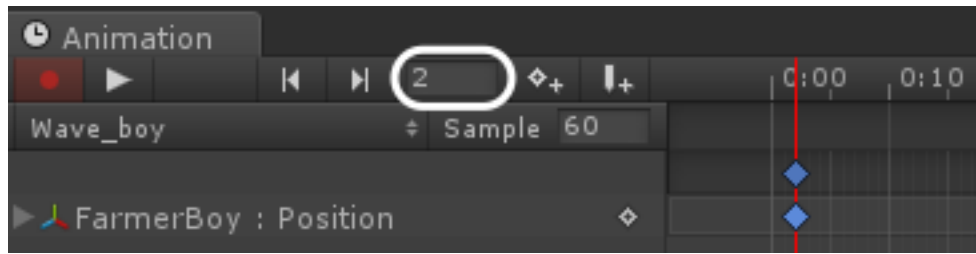






- Idle_boy (Read-Only)
- Running_boy (Read-Only)
- Talk_boy (Read-Only)
- Walk_boy (Read-Only)
- Take (Read-Only)
-
-





Chapter 5: Scripting the Game

Inspector

testScript Import Settings

Imported Object

testScript

```
using UnityEngine;
using System.Collections;

public class testScript : MonoBehaviour {

    // Use this for initialization
    void Start () {

    }

    // Update is called once per frame
    void Update () {

    }

}
```

```
1 using UnityEngine;
2 using System.Collections;
3
4 public class testScript : MonoBehaviour {
5
6     // Use this for initialization
7     void Start () {
8
9     }
10
11    // Update is called once per frame
12    void Update () {
13
14    }
15 }
```

```
1 using UnityEngine;
2 using System.Collections;
3
4 public class testScript : MonoBehaviour {
5
6     // Use this for initialization
7     void Start () {
8         print ("Hello Little Farmer Colt!");
9     }
10
11    // Update is called once per frame
12    void Update () {
13
14    }
15 }
```

Inspector

testScript Import Settings

Imported Object

testScript

```
using UnityEngine;
using System.Collections;

public class testScript : MonoBehaviour {

    // Use this for initialization
    void Start () {
        print ("Hello Little Farmer Colt!");
    }

    // Update is called once per frame
    void Update () {

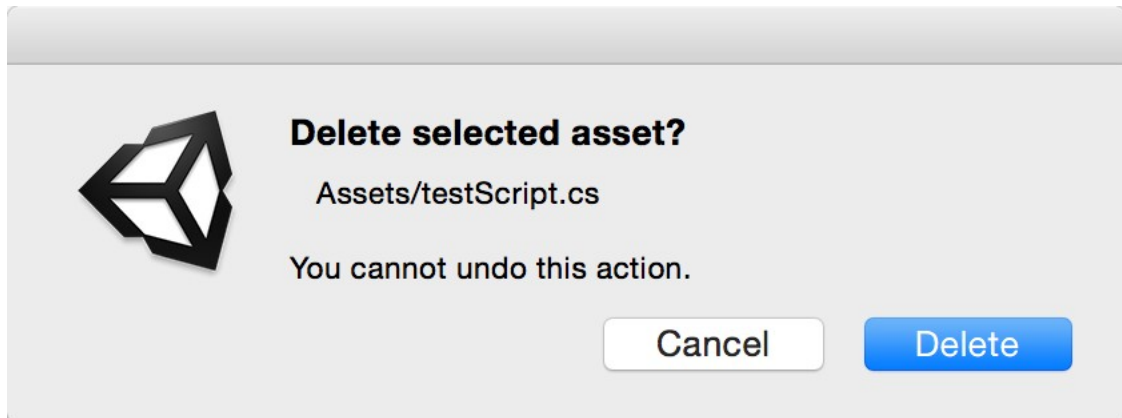
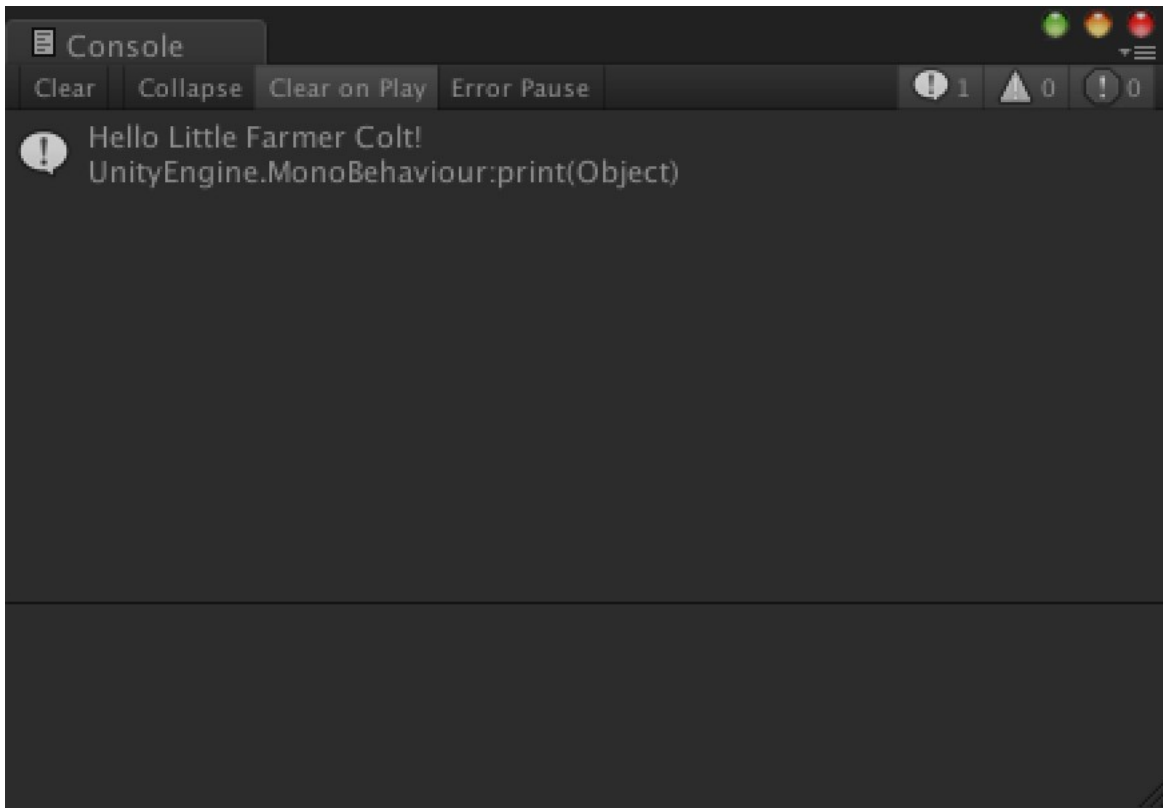
    }

}
```

Test Script (Script) testScript

Script testScript

Hello Little Farmer Colt!



Lens Flares

Category: Scripting/Effects
Publisher: Unity Technologies
Rating: ★★★★★ (319)
Price: Free

[Open in Unity](#) [Twitter](#) [Facebook](#) [Google+](#)

Requires Unity 3.1.0 or higher.

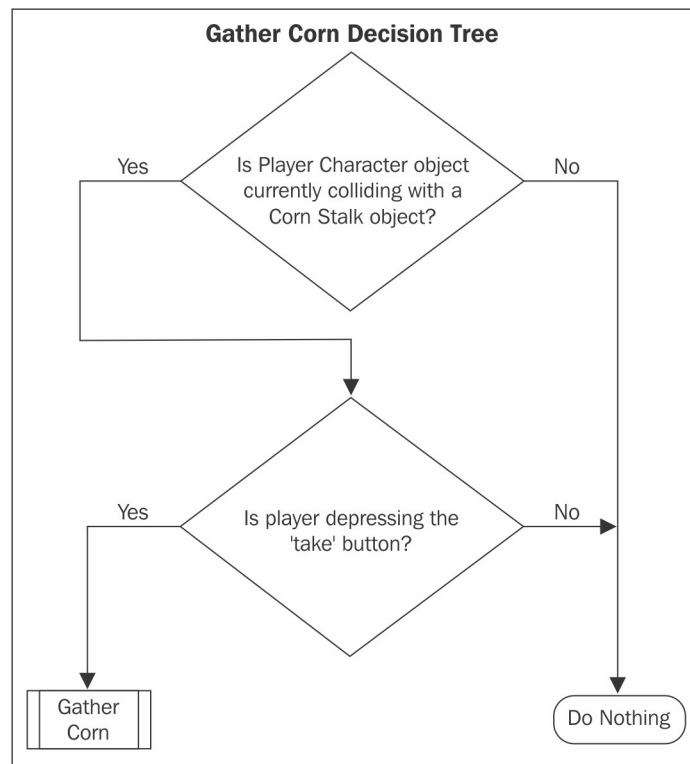
Lens Flares simulate the effect of lights refracting inside camera lens. They are used to represent really bright lights or, more subtly, just to add a bit more atmosphere to your scene.

This package includes sixteen lens flares for your Unity projects.

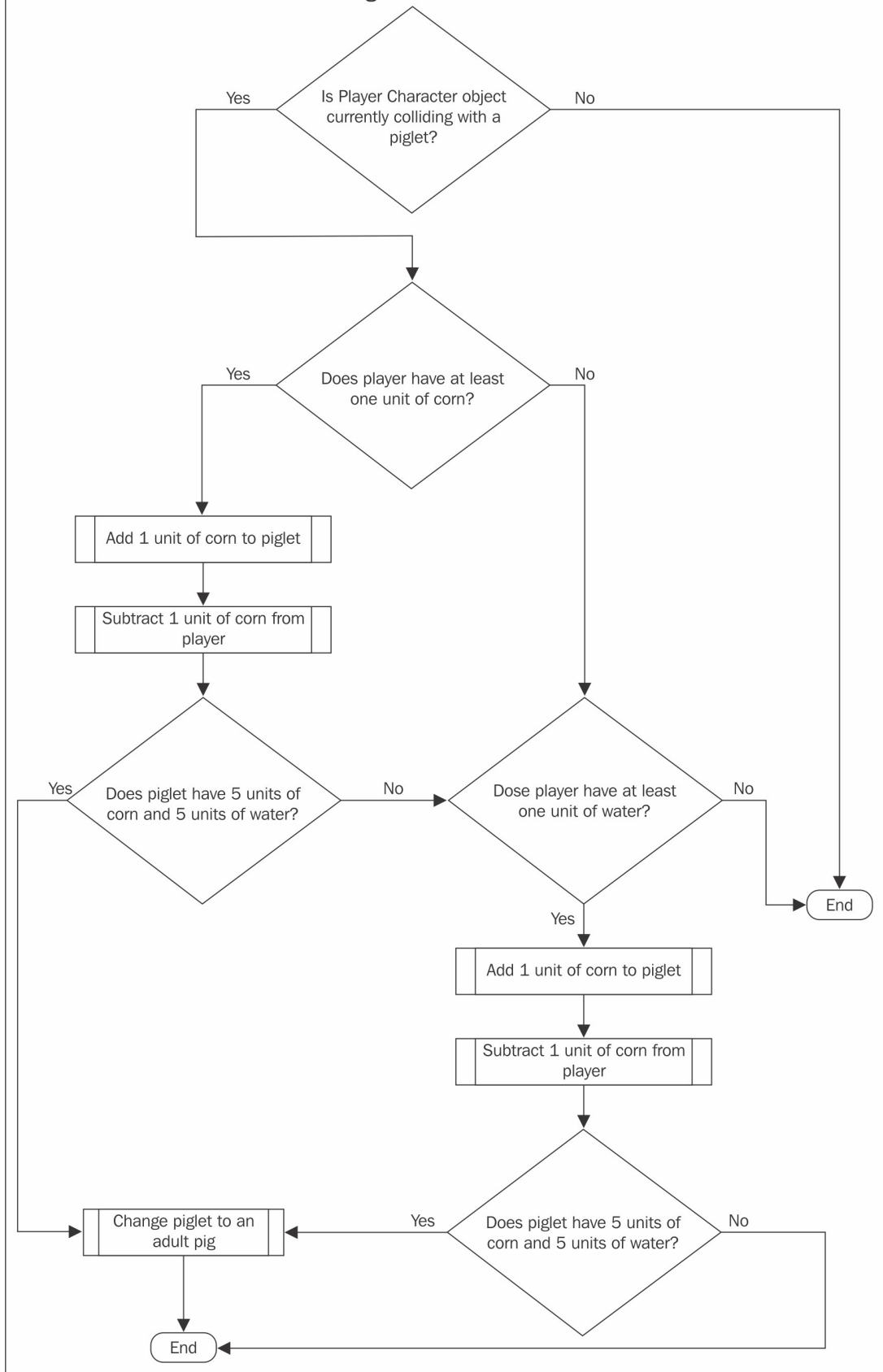


Version: 1.04 (Oct 19, 2010) Size: 1.9 MB

[Visit Publisher's Website](#)



Feed Piglet Decision Tree




```
Arrays.cs
Arrays ▶ Start ()
1 using UnityEngine;
2 using System.Collections;
3
4 public class Arrays : MonoBehaviour {
5
6     int[,] pigletArray = new int[10, 3];
7
8     void Start () {
9
10        for(int i = 0; i < 10; i++)
11        {
12            pigletArray[i, 0] = 1; // visible: 0=no; 1=yes
13            pigletArray[i, 1] = 0; // corn
14            pigletArray[i, 2] = 0; // water
15        }
16    }
17
18    void Update () {
19
20    }
21 }
```

	0	1	2
0			
1			
2			
3			
4			
5			
6			
7			
8			
9			

Console

Clear Collapse Clear on Play Error Pause

- ! Piglet 0 1 0 0
UnityEngine.Debug:Log(Object)
- ! Piglet 1 1 0 0
UnityEngine.Debug:Log(Object)
- ! Piglet 2 1 0 0
UnityEngine.Debug:Log(Object)
- ! Piglet 3 1 0 0
UnityEngine.Debug:Log(Object)
- ! Piglet 4 1 0 0
UnityEngine.Debug:Log(Object)
- ! Piglet 5 1 0 0
UnityEngine.Debug:Log(Object)
- ! Piglet 6 1 0 0
UnityEngine.Debug:Log(Object)
- ! Piglet 7 1 0 0
UnityEngine.Debug:Log(Object)
- ! Piglet 8 1 0 0
UnityEngine.Debug:Log(Object)
- ! Piglet 9 1 0 0
UnityEngine.Debug:Log(Object)

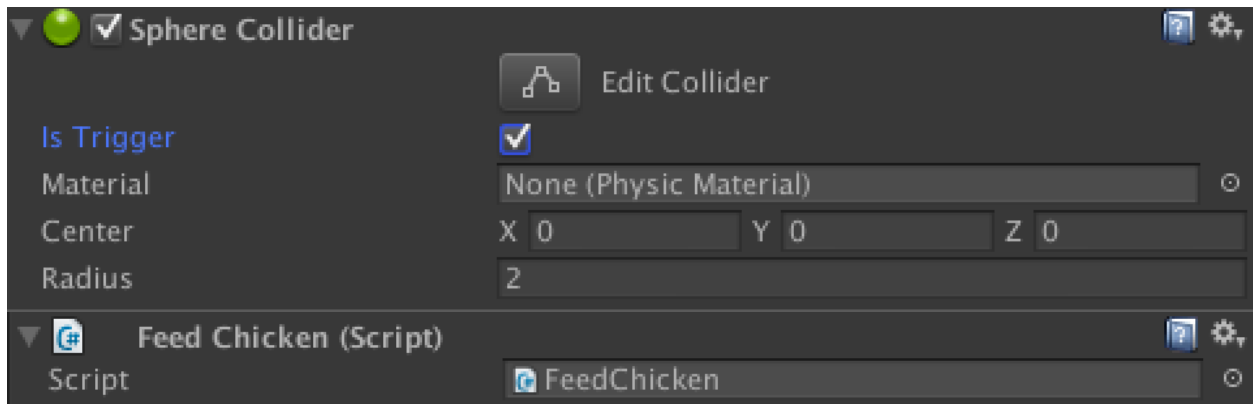
```
Arrays.cs
Arrays ▶ Start ()
1 using UnityEngine;
2 using System.Collections;
3
4 public class Arrays : MonoBehaviour {
5
6     int[,] pigletArray = new int[10, 3];
7     int[,] pigArray = new int[10, 3];
8
9     void Start () {
10
11         for(int i = 0; i < 10; i++)
12         {
13             // piglet Array initial values
14             pigletArray[i, 0] = 1; // visible: 0=no; 1=yes
15             pigletArray[i, 1] = 0; // corn
16             pigletArray[i, 2] = 0; // water
17
18             // pig Array initial values
19             pigArray[i, 0] = 0; // visible: 0=no; 1=yes
20             pigArray[i, 1] = 0; // corn
21             pigArray[i, 2] = 0; // water
22         }
23     }
24
25     void Update () {
26
27     }
28 }
```

```
Arrays.cs x
C Arrays ▶ No selection
1 using UnityEngine;
2 using System.Collections;
3
4 public class Arrays : MonoBehaviour {
5
6     int[,] pigletArray = new int[10, 3];
7     int[,] pigArray = new int[10, 3];
8     int[,] chickArray = new int[10, 3];
9     int[,] chickenArray = new int[10, 3];
10
11     void Start () {
12
13         for(int i = 0; i < 10; i++)
14         {
15             // piglet Array initial values
16             pigletArray[i, 0] = 1; // visible: 0=no; 1=yes
17             pigletArray[i, 1] = 0; // corn
18             pigletArray[i, 2] = 0; // water
19
20             // adult pig Array initial values
21             pigArray[i, 0] = 0; // visible: 0=no; 1=yes
22             pigArray[i, 1] = 0; // corn
23             pigArray[i, 2] = 0; // water
24
25             // baby chick Array initial values
26             chickArray[i, 0] = 1; // visible: 0=no; 1=yes
27             chickArray[i, 1] = 0; // corn
28             chickArray[i, 2] = 0; // water
29
30             // adult chicken Array initial values
31             chickenArray[i, 0] = 0; // visible: 0=no; 1=yes
32             chickenArray[i, 1] = 0; // corn
33             chickenArray[i, 2] = 0; // water
34         }
35     }
36     |
37     void Update () {
38
39     }
40 }
```

```
GameData.cs
No selection
1 using UnityEngine;
2 using System.Collections;
3
4 public class GameData : MonoBehaviour {
5
6     public int corn = 0;
7     public int water = 0;
8
9     void Start () {
10
11     }
12
13     void Update () {
14
15     }
16 }
```

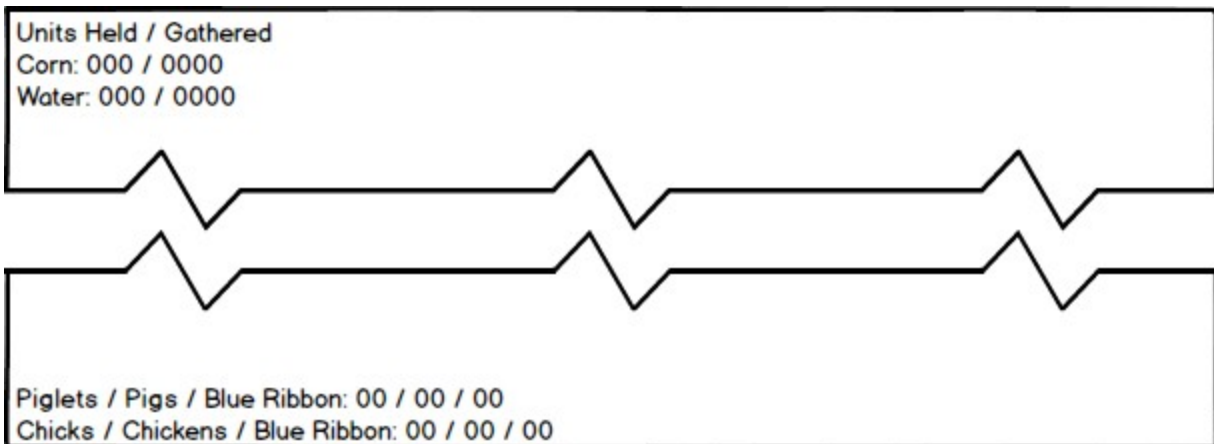
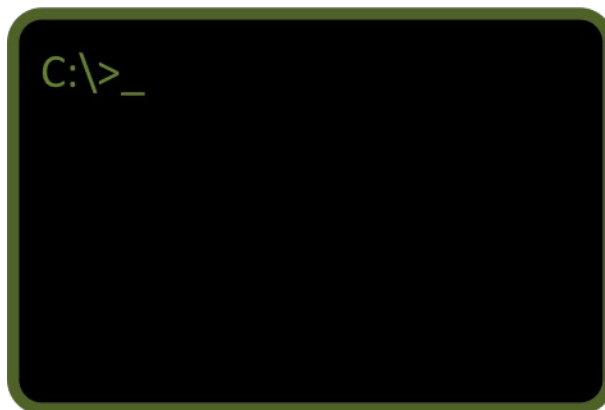
The screenshot shows the Unity Inspector window with the following configuration:

- Arrays (Script)**: Script: Arrays
- Game Data (Script)**: Script: GameData, Corn: 0, Water: 0
- Sphere Collider**: Edit Collider, Is Trigger: , Material: None (Physic Material), Center: X 0, Y 0, Z 0, Radius: 1.5

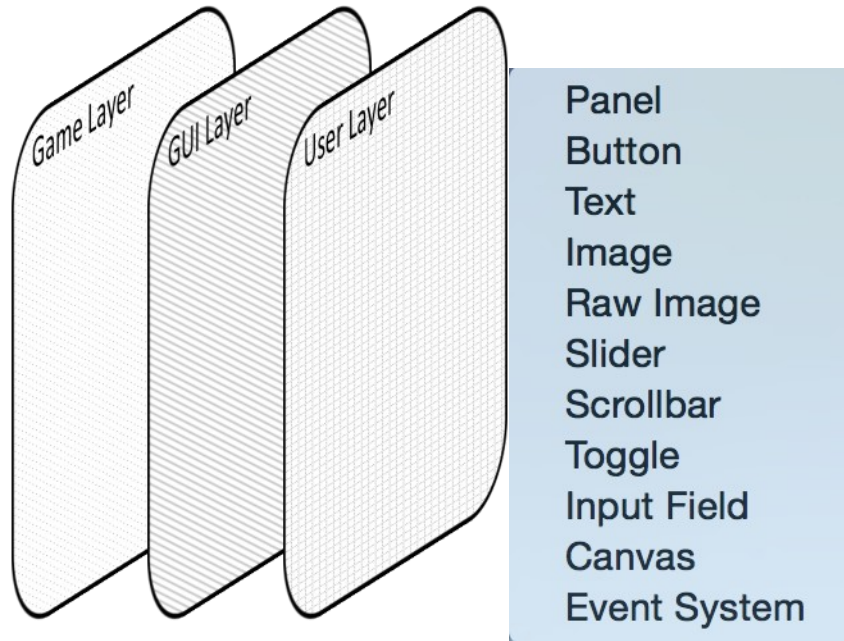


Chapter 6: Adding a Graphical User Interface

Early User Interface

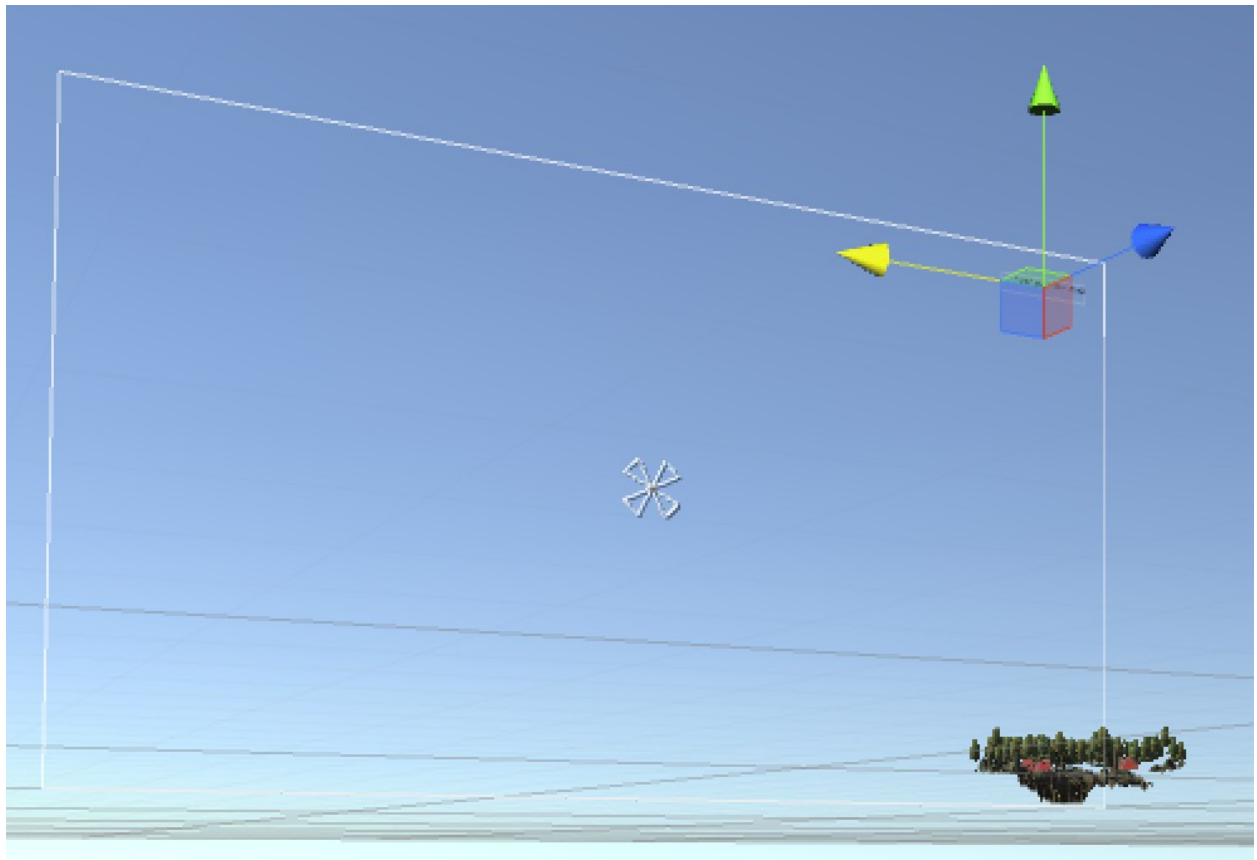


Game Layers



- ✓ Screen Space - Overlay
- Screen Space - Camera
- World Space

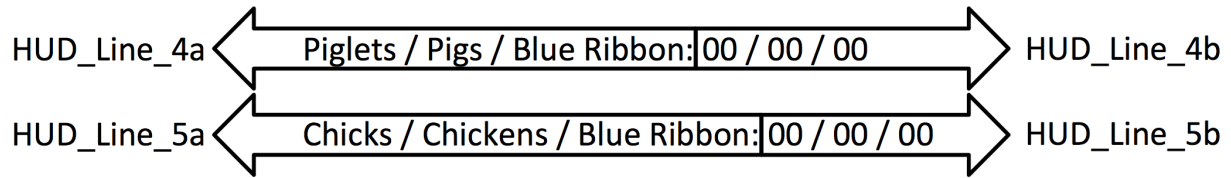




Units Held / Gathered

Corn: 000 / 0000

Water: 000 / 0000



- ▼ Canvas
 - HUD_Line_1
 - HUD_Line_2
 - HUD_Line_3
 - HUD_Line_4a
 - HUD_Line_4b
 - HUD_Line_5a
 - HUD_Line_5b

Piglets / Pigs / Blue Ribbon: 00 / 00 / 00
 Chicks / Chickens / Blue Ribbon: 00 / 00 / 00

Inspector

Tags & Layers

- ▶ Tags
- ▶ Sorting Layers
- ▼ Layers

Builtin Layer 0	Default
Builtin Layer 1	TransparentFX
Builtin Layer 2	Ignore Raycast
Builtin Layer 3	
Builtin Layer 4	Water
Builtin Layer 5	UI
Builtin Layer 6	
Builtin Layer 7	
User Layer 8	Mini-Map

- ✓ Default
- TransparentFX
- Ignore Raycast
- Water
- UI
- Mini-Map**
- Add Layer...

- Nothing
- Everything
- ✓ Default
- ✓ TransparentFX
- ✓ Ignore Raycast
- ✓ Water
- ✓ UI
- Mini-Map**



Save Scene

Save As:

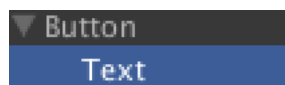
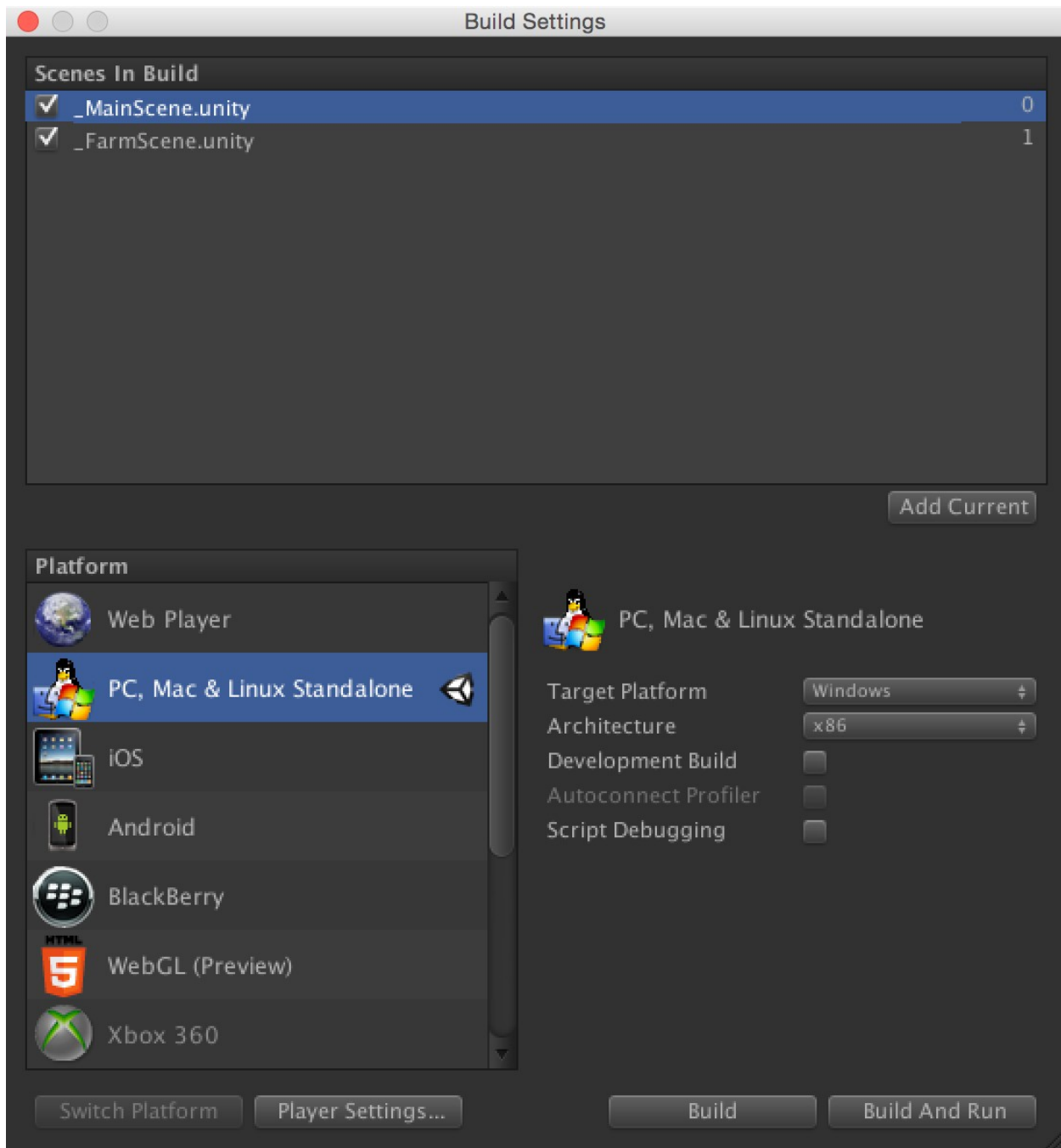


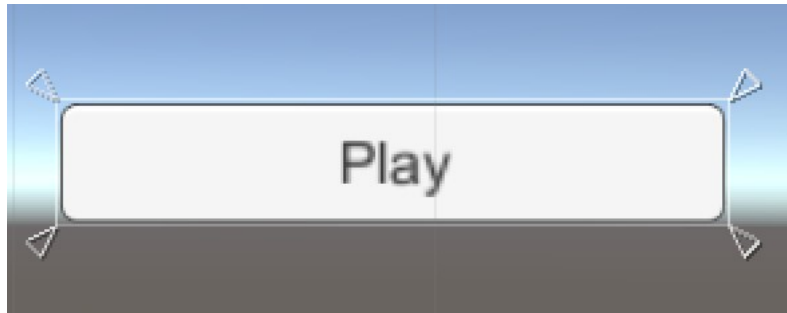
Tags:

Where:

Cancel

Save





Button (Script)

Interactable

Transition: Color Tint

Target Graphic: Play_Button (Image)

Normal Color: [Color Picker]

Highlighted Color: [Color Picker]

Pressed Color: [Color Picker]

Disabled Color: [Color Picker]

Color Multiplier: 1

Fade Duration: 0.1

Navigation: Automatic

Visualize

On Click ()

List is Empty

On Click ()

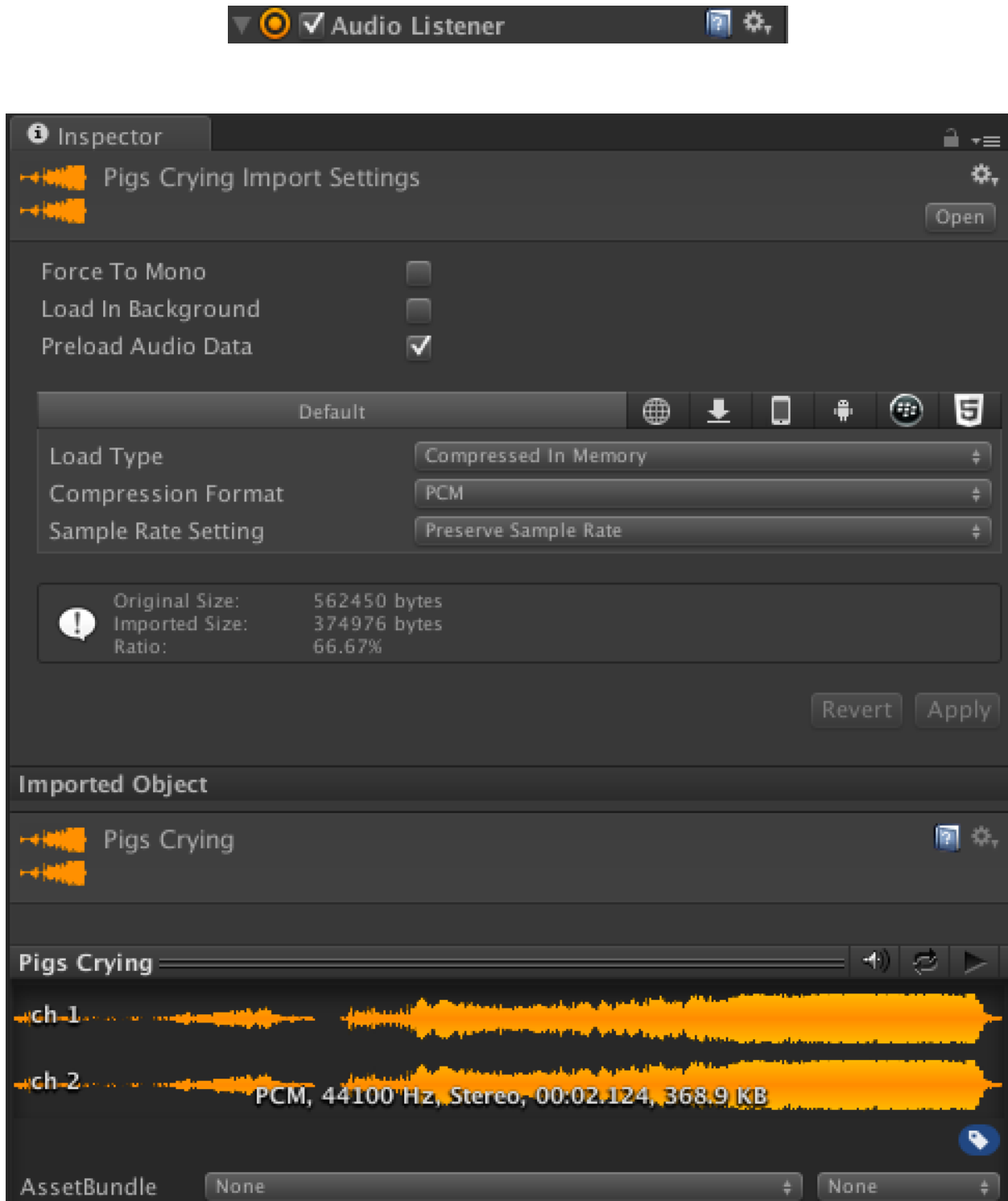
Runtime Only: [Dropdown]

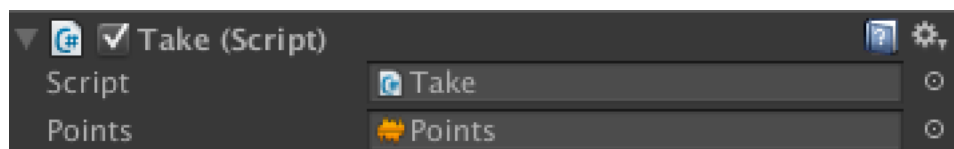
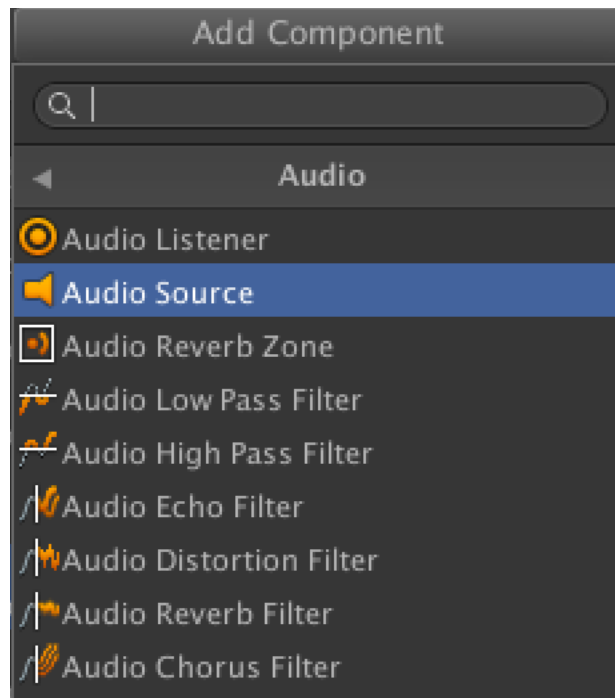
LoadSceneOnClick.LoadScene: [Dropdown]

Canvas (LoadScene): [Dropdown]

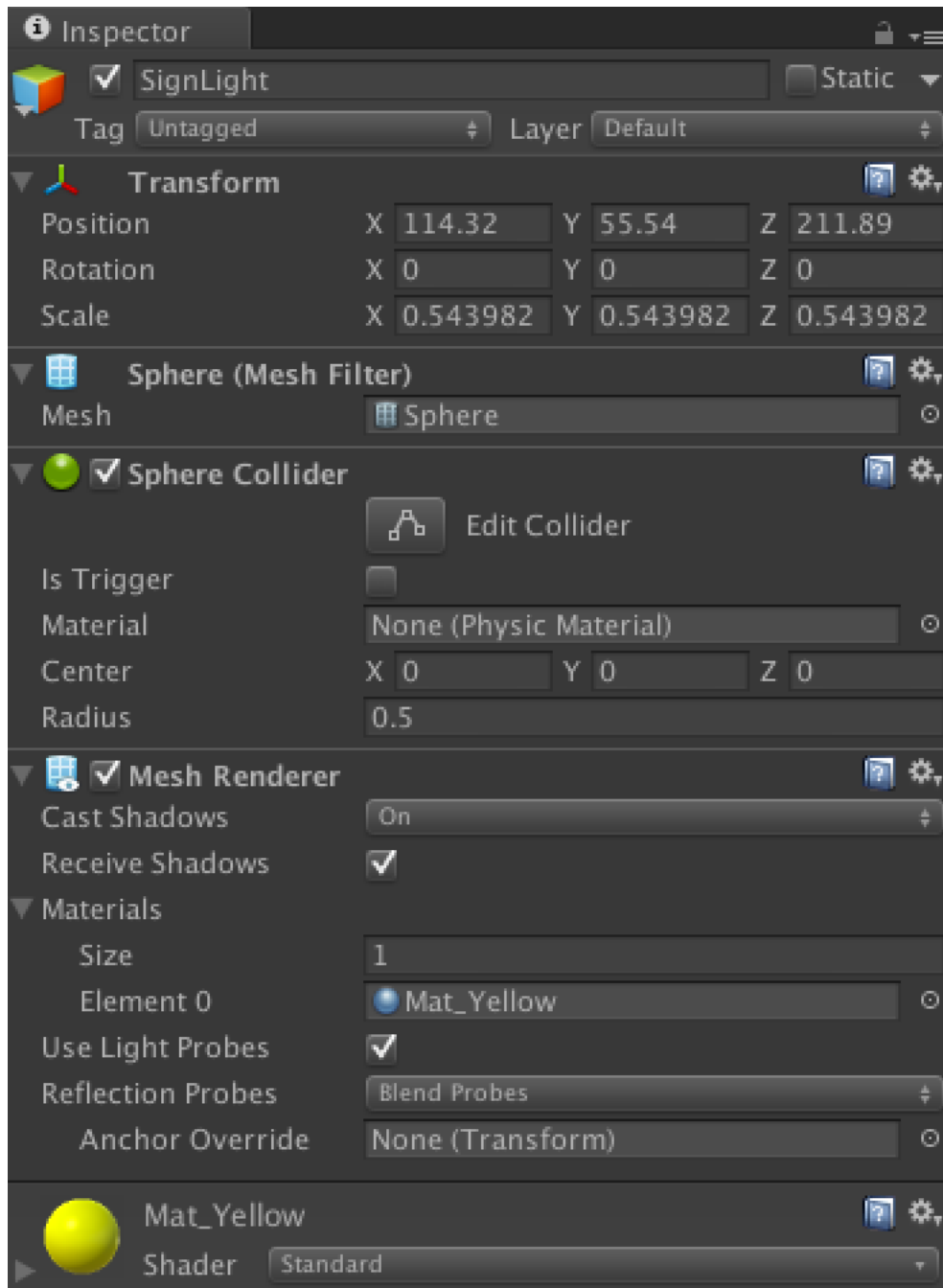
1

Chapter 7: Polishing and Optimizing the Game

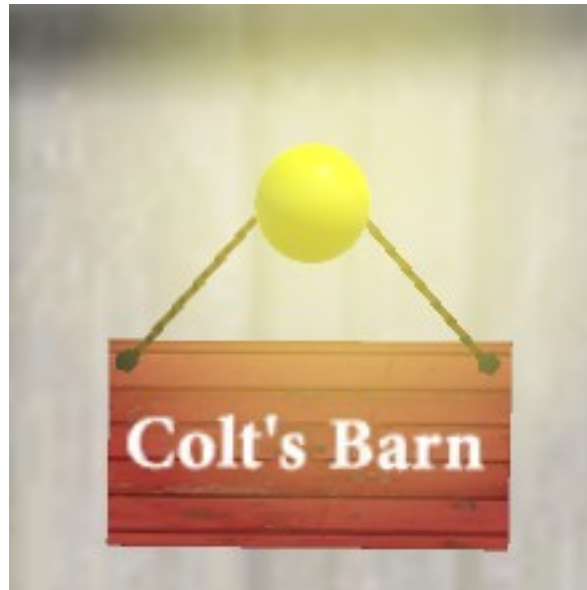








▼ SignLight
Spotlight



Terrain

Terrain Settings

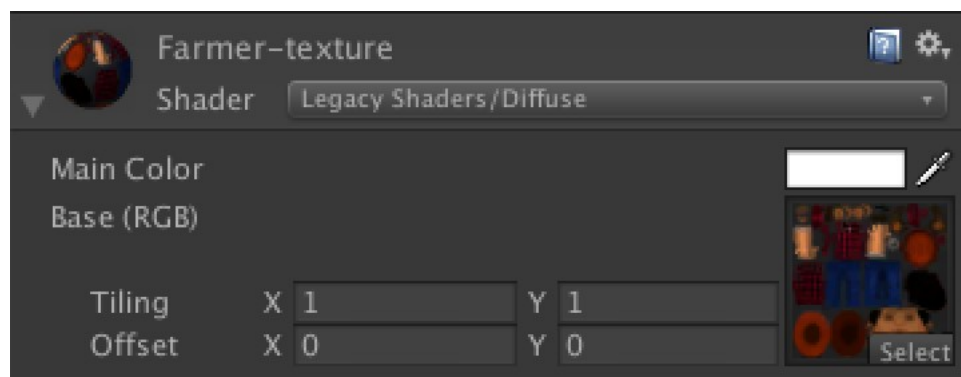
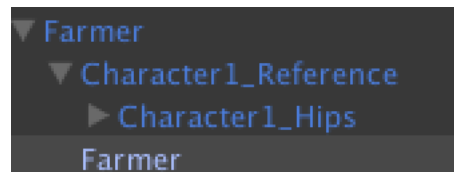
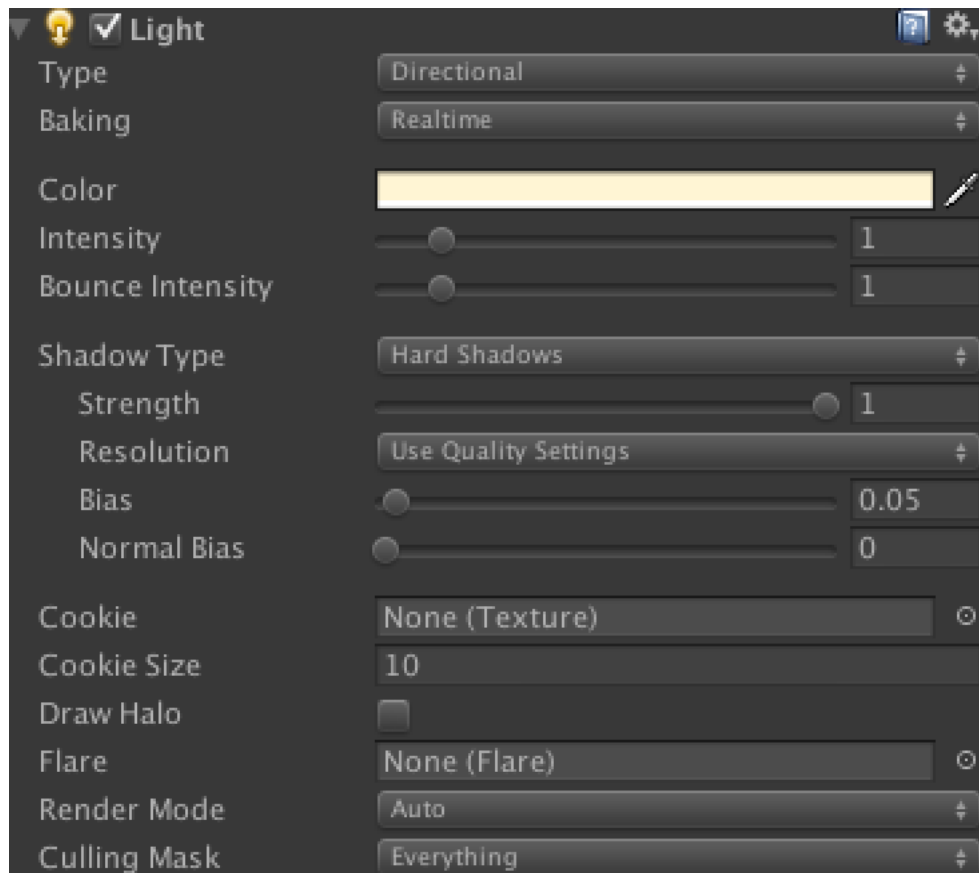
Base Terrain

Draw

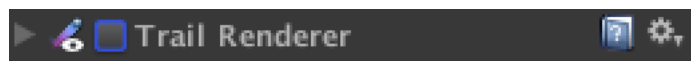
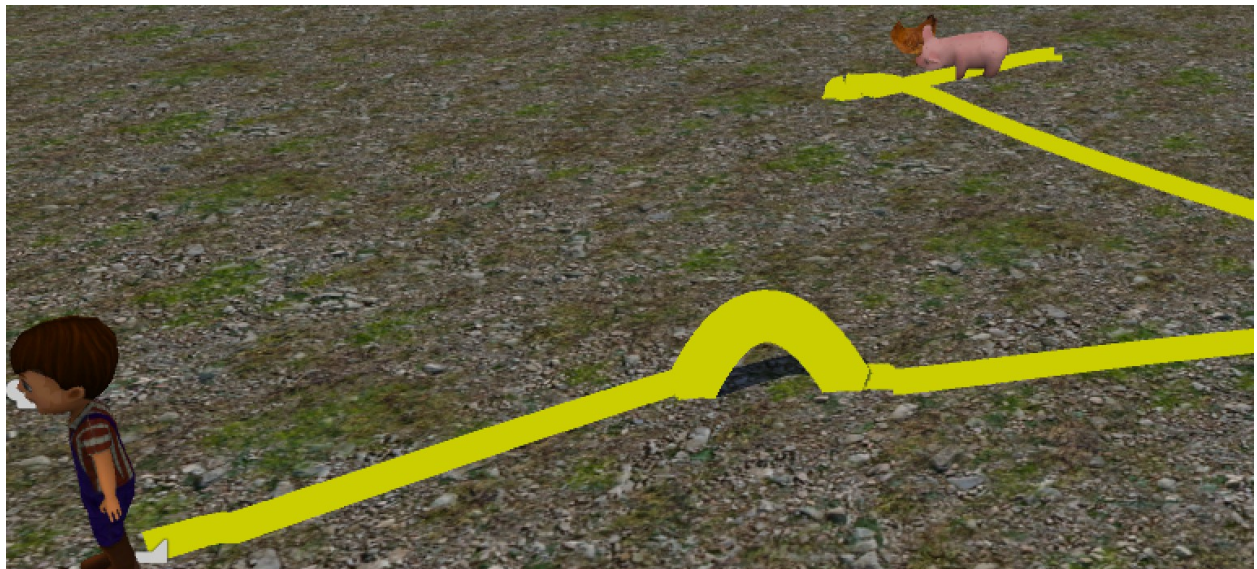
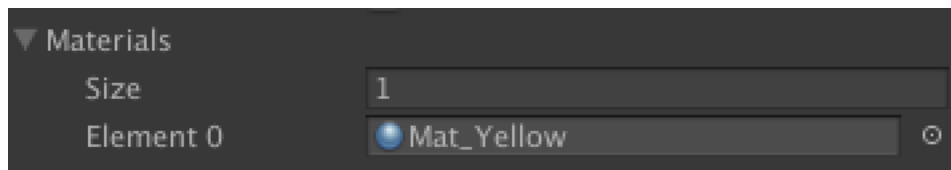
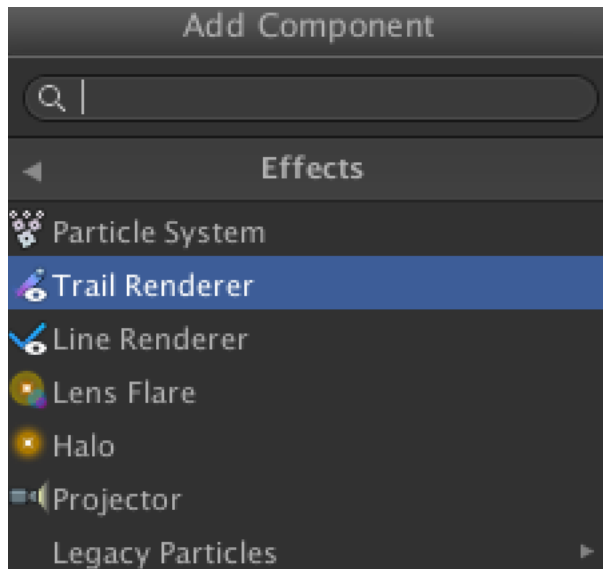
Pixel Error

Base Map Dist.

Cast Shadows









CPU Usage

- Rendering
- Scripts
- Physics
- GarbageCollector
- VSync
- GI
- Others



Audio

- Playing Sources
- Audio Voices
- Total Audio CPU
- Total Audio Memory



Rendering

- Batches
- SetPass Calls
- Triangles
- Vertices



Physics

- Active Rigidbodies
- Number of Contacts



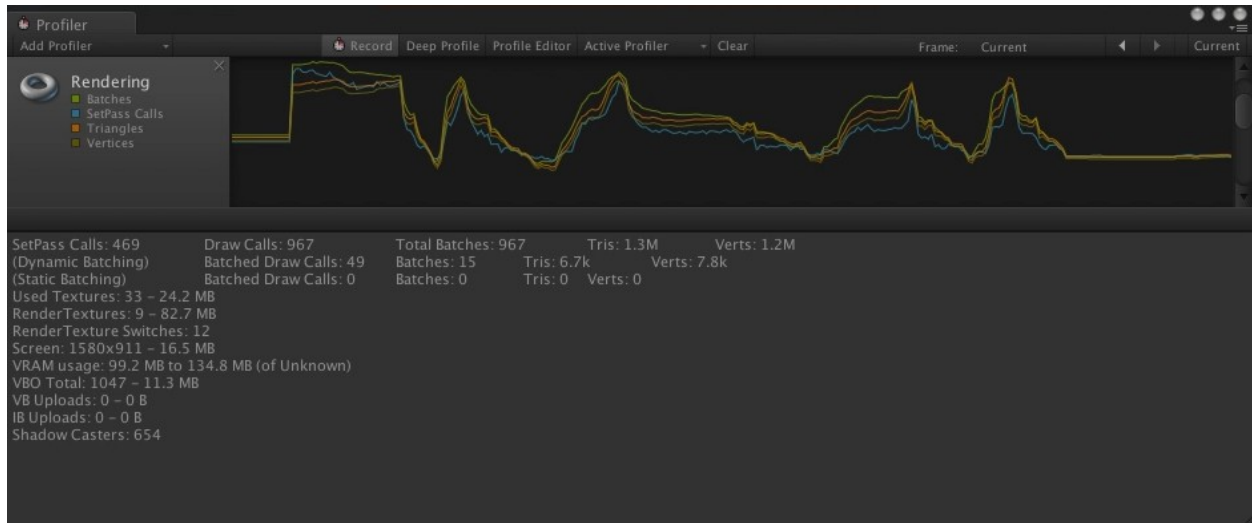
Memory

- Total Allocated
- Texture Memory
- Mesh Memory
- Material Count
- Object Count
- Total GC Allocated
- GC Allocated

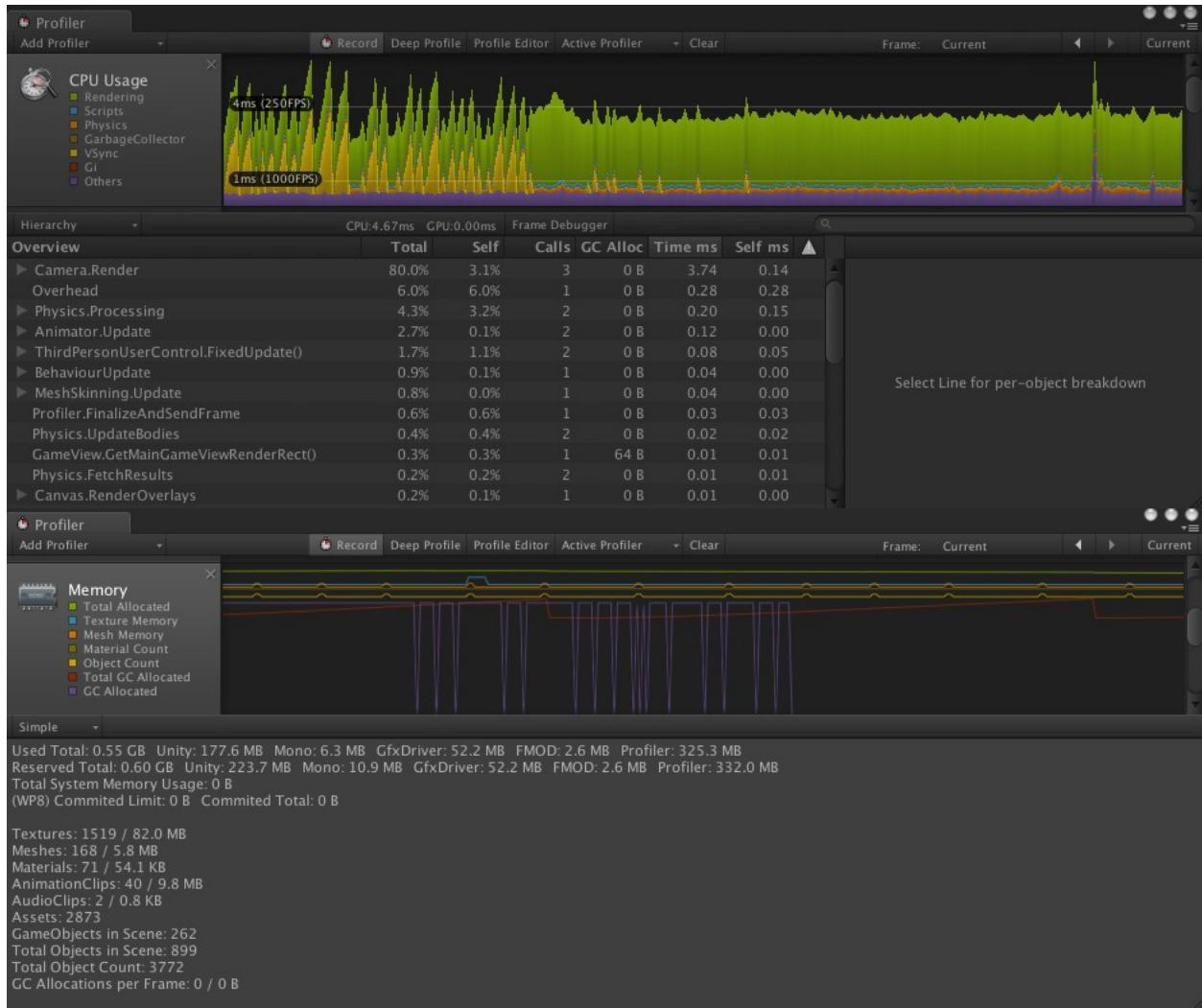


Physics (2D)

- Active Bodies
- Sleeping Bodies
- Dynamic Bodies
- Kinematic Bodies
- Discrete Bodies
- Continuous Bodies
- Joints
- Contacts







Chapter 8: What's Next?



Particle System ? ⚙️

Open Editor...

Particle System +

Duration	5.00
Looping	<input checked="" type="checkbox"/>
Prewarm	<input type="checkbox"/>
Start Delay	0
Start Lifetime	5 ▾
Start Speed	5 ▾
Start Size	1 ▾
Start Rotation	0 ▾
Start Color	▾
Gravity Modifier	0
Inherit Velocity	0
Simulation Space	Local ▾
Play On Awake	<input checked="" type="checkbox"/>
Max Particles	1000

- Emission
- Shape
- Velocity over Lifetime
- Limit Velocity over Lifetime
- Force over Lifetime
- Color over Lifetime
- Color by Speed
- Size over Lifetime
- Size by Speed
- Rotation over Lifetime
- Rotation by Speed
- External Forces
- Collision
- Sub Emitters
- Texture Sheet Animation
- Renderer

Resimulate Wireframe