

Load Image at Runtime in Unity

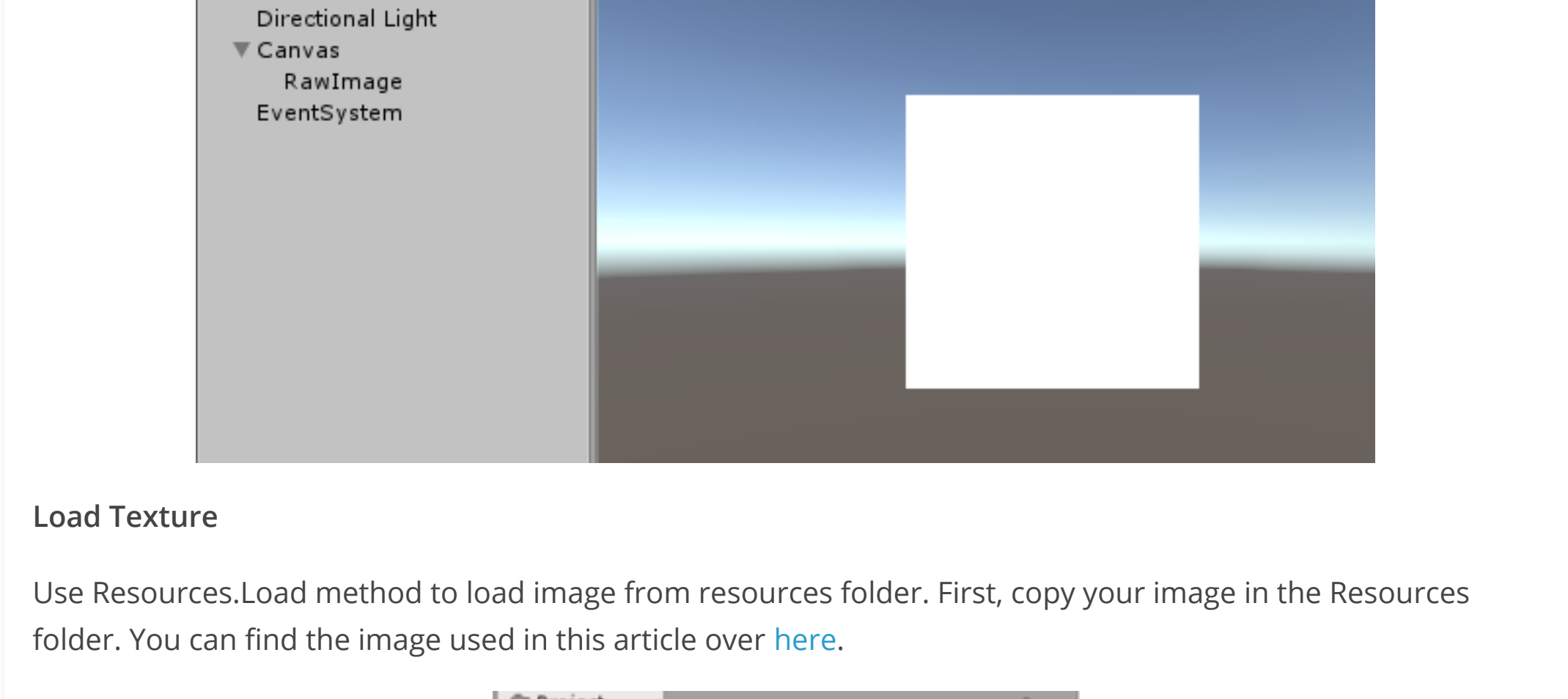
July 8, 2017 Gyanendu Shekhar 2 Comments Unity

In this article, I will explain how to load image at runtime in unity scene. I will cover following ways by which you can add image at runtime in unity.

- Load image from Resource folder
- Load image from any location of local computer
- Load image from a byte array
- Load image from an URL

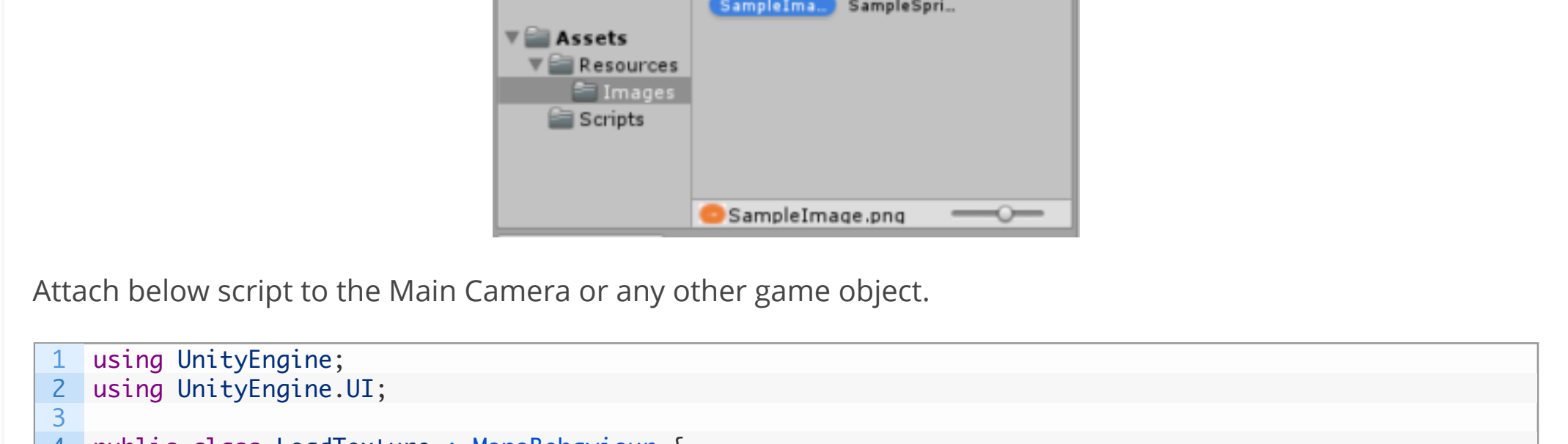
Load Image from Resource folder

Let's prepare a simple scene in unity for loading images at runtime. I have added RawImage game object in the scene.



Load Texture

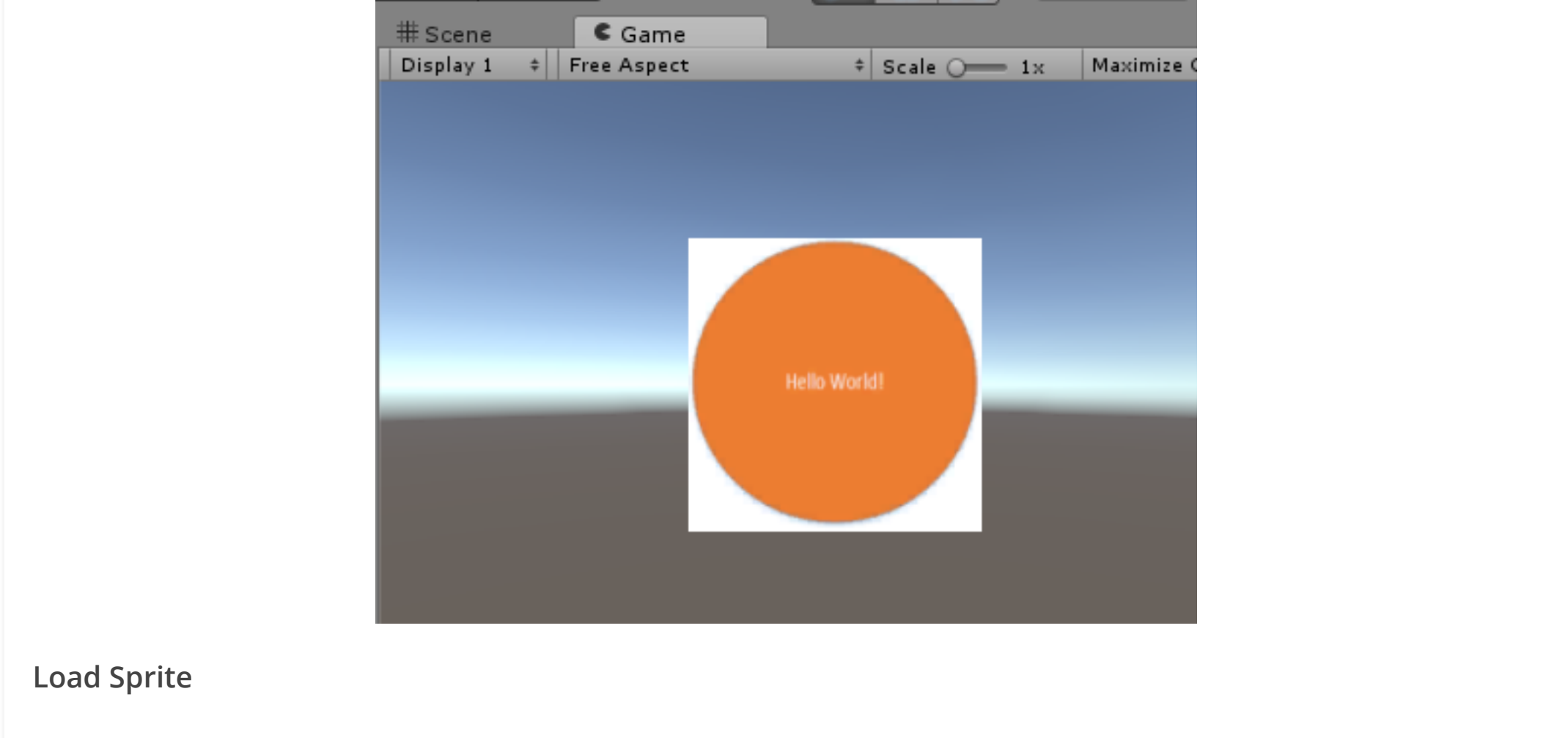
Use Resources.Load method to load image from resources folder. First, copy your image in the Resources folder. You can find the image used in this article over [here](#).



Attach below script to the Main Camera or any other game object.

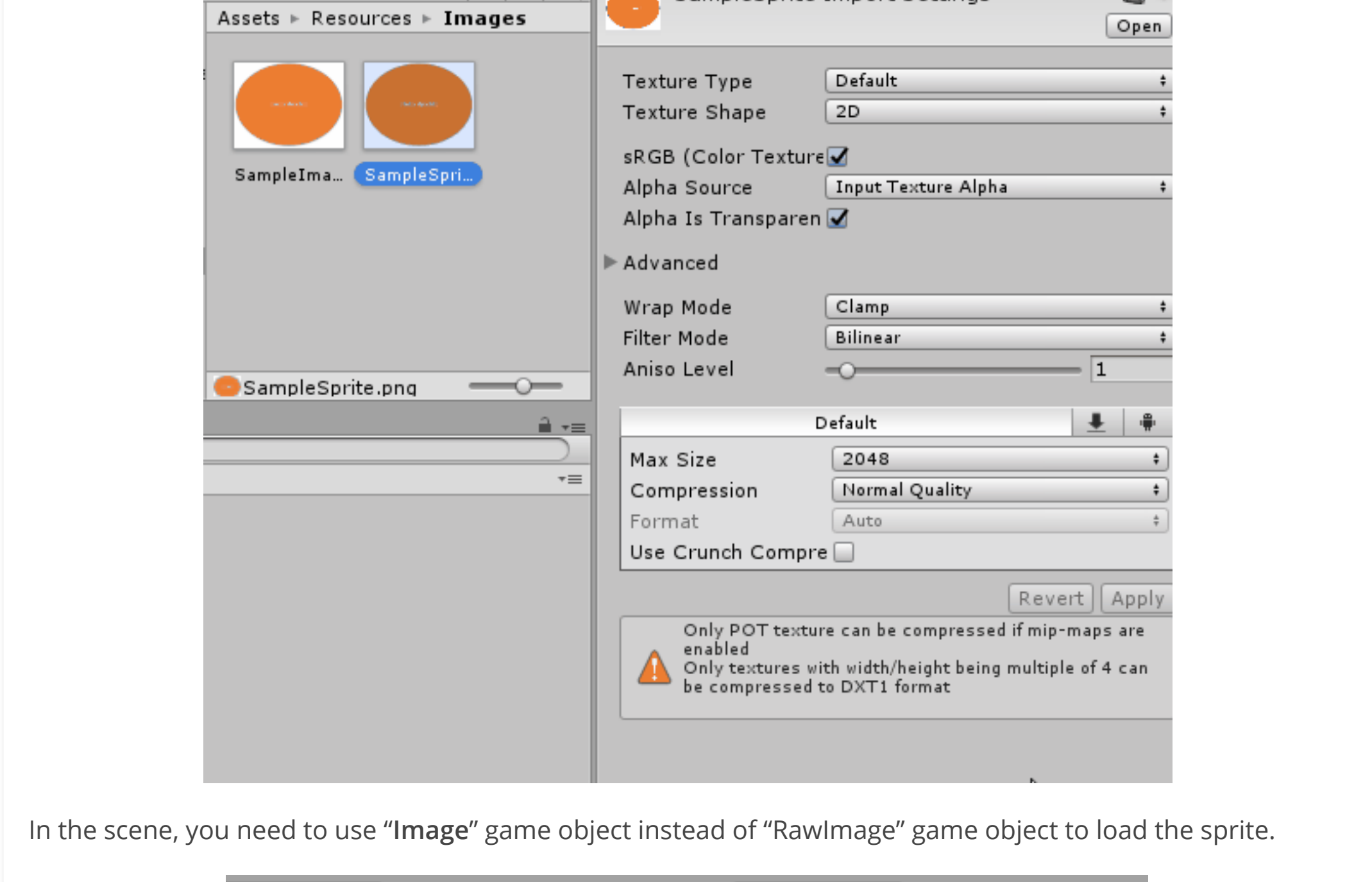
```
1 using UnityEngine;
2 using UnityEngine.UI;
3
4 public class LoadTexture : MonoBehaviour {
5     Texture2D myTexture;
6
7     // Use this for initialization
8     void Start () {
9         // Load texture from resource folder
10        myTexture = Resources.Load ("Images/SampleImage") as Texture2D;
11
12        GameObject rawImage = GameObject.Find ("RawImage");
13        rawImage.GetComponent<RawImage> ().texture = myTexture;
14    }
15 }
```

Run the application.



Load Sprite

First convert the image into sprite in unity editor, then you can load that at runtime just like textures.



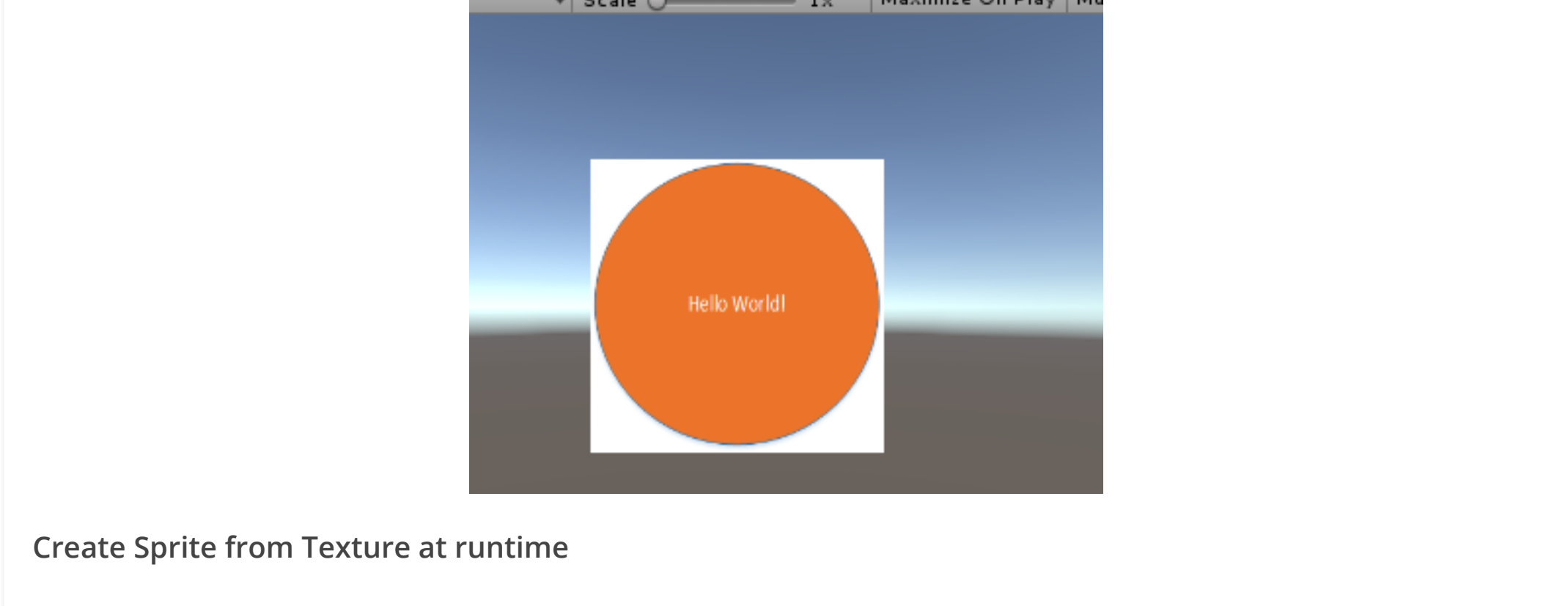
In the scene, you need to use "Image" game object instead of "RawImage" game object to load the sprite.



Add below given script to Main Camera or any other game object.

```
1 using UnityEngine;
2 using UnityEngine.UI;
3
4 public class LoadSprite : MonoBehaviour {
5     Sprite sprite;
6
7     // Use this for initialization
8     void Start () {
9         sprite = Resources.Load <Sprite>("Images/SampleSprite");
10
11        GameObject image = GameObject.Find ("Image");
12        image.GetComponent<Image>().sprite = sprite;
13    }
14 }
```

Run the application.



Create Sprite from Texture at runtime

You can also create sprite from texture at runtime.

```
1 // convert texture to sprite if required otherwise load sprite directly from resources folder
2 Texture2D myTexture = Resources.Load <Texture2D>("Images/SampleImage");
3 sprite = Sprite.Create (myTexture, new Rect (0, 0, myTexture.width, myTexture.height), new Vector2 (0.5f
```

Load image from any location of local computer

Use WWW class for loading image from any local directory. Make sure the file path is correct. The file path should be like "file:///yourPath".

```
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4 using UnityEngine.UI;
5
6 public class LoadLocalMachine : MonoBehaviour {
7
8     // Use this for initialization
9     IEnumerator Start () {
10        WWW www = new WWW ("file:///D:/SampleImage.png");
11        while (!www.isDone)
12            yield return null;
13        GameObject image = GameObject.Find ("RawImage");
14        image.GetComponent<RawImage>().texture = www.texture;
15    }
16 }
```

Run the application. You will get the same result as above.

Load Image from byte array

We can also load image using a byte array with LoadImage method of Texture2D class.

```
1 using UnityEngine;
2 using UnityEngine.UI;
3 using System.IO;
4
5 public class LoadImageDemo : MonoBehaviour {
6
7     // Use this for initialization
8     void Start () {
9         // read image and store in a byte array
10        byte[] byteArray = File.ReadAllBytes(@"D:\SampleImage.png");
11        //create a texture and load byte array to it
12        // Texture size does not matter
13        Texture2D sampleTexture = new Texture2D(2,2);
14        // the size of the texture will be replaced by image size
15        bool isLoaded = sampleTexture.LoadImage(byteArray);
16        // apply this texture as per requirement on image or material
17        GameObject image = GameObject.Find ("RawImage");
18        if (isLoaded)
19        {
20            image.GetComponent<RawImage>().texture = sampleTexture;
21        }
22    }
23 }
```

Load Image from an URL

Image can also be loaded from any URL using WWW class.

```
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4 using UnityEngine.UI;
5
6 public class LoadWWW : MonoBehaviour {
7
8     // Use this for initialization
9     IEnumerator Start () {
10        WWW www = new WWW ("http://gyanendushekar.com/wp-content/uploads/2017/07/SampleImage.png");
11        while (!www.isDone)
12            yield return null;
13        Debug.Log (www.texture.name);
14        GameObject rawImage = GameObject.Find ("RawImage");
15        rawImage.GetComponent<RawImage> ().texture = www.texture;
16    }
17 }
```

Run the application. You will get the same result as above.

Image can also be loaded from asset bundles. Please refer my article [Load Asset Bundle in Unity](#) for details.

Hope you enjoyed reading this article. Please post your comments for queries and feedback. Thanks for reading.

Share [Twitter](#) [Facebook](#) [LinkedIn](#) [Reddit](#)

Bio

Latest Posts

Gyanendu Shekhar

Gyanendu Shekhar is a technology enthusiast. He loves to learn new technologies. His area of interest includes Microsoft technologies, Augmented reality, Virtual reality, unity3d and android development.

← Getting started with Facebook Gameroom: Unity 5 Tutorial

Tuples in C# →

2 thoughts on “Load Image at Runtime in Unity”

PETER MAVRONICOLAS

April 27, 2018 at 12:55 pm

Permalink

Gyanendu,

Excellent tutorial! Are you able to run the same code on a 3D object such as a cube or do we have to use 'canvas' and 'raw image'?

Manoj balaji

June 7, 2018 at 1:04 pm

Permalink

i tried to get the png file as a texture from the asset folder and i want to apply the texture to material. but it's not working

Reply

Leave a Reply

Your email address will not be published. Required fields are marked *

Comment

Name *

Email *

Website

Post Comment