

# **KOMPUTASI MULTIMEDIA**

## **MODUL 6 USING CAMERAS**

**Oleh:**

**KINANTI PERMATA PUTRI      NIM. 1841720022**

**TI-3D**

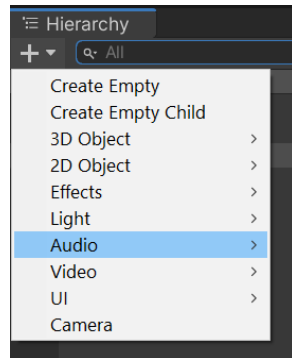


**PROGRAM STUDI TEKNIK INFORMATIKA  
JURUSAN TEKNOLOGI INFORMASI  
POLITEKNIK NEGERI MALANG  
MARET 2021**

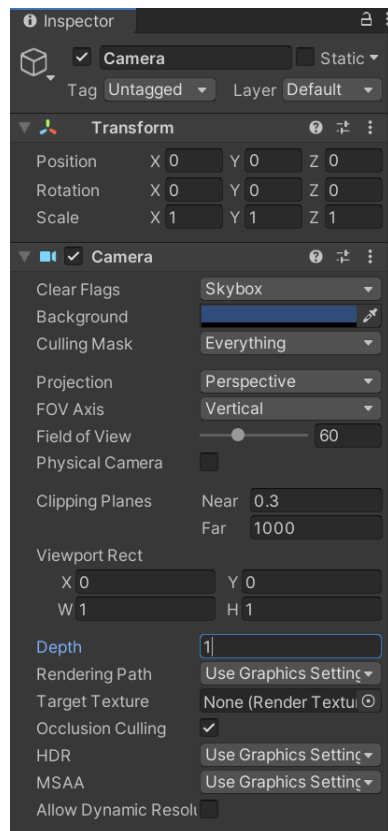
## A. Praktikum

### 1. Membuat tampilan Camera dari Arah Sudut Pandang lain di dalam Layar

- 1) Buatlah Project 3D baru pada Unity.
- 2) Masukkan paket BasicScene kedalam Assets.
- 3) Pada Project View, buka BasicScene kemudian akan muncul karakter animasi dan beberapa objek benda yang telah dibuat.
- 4) Tambahkan new Camera ke dalam scene melalui menu Create yang berada diatas Hierarchy, Pilih Create → Camera.



- 5) Pilih Camera pada Hierarchy yang telah dibuat, dari Inspector rubahlah “Depth = 1” seperti gambar di bawah ini.



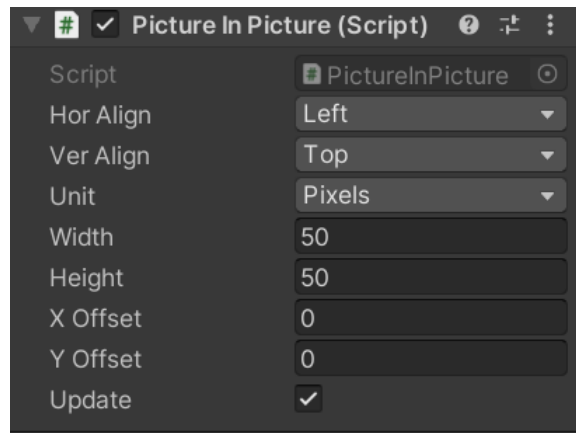
- 6) Pada Project View buatlah script C# baru dengan code seperti dibawah ini, dan berilah nama PictureInPicture.

```
PictureInPicture.cs X
Assets > Script > PictureInPicture.cs > PictureInPicture > AdjustCamera()
1  using UnityEngine;
   0 references
2  public class PictureInPicture: MonoBehaviour {
   5 references | 2 references | 1 reference | 1 reference
3      public enum hAlignment{left, center, right};
   5 references | 2 references | 1 reference | 1 reference
4      public enum vAlignment{top, middle, bottom};
   1 reference
5      public hAlignment horAlign = hAlignment.left;
   1 reference
6      public vAlignment verAlign = vAlignment.top;
   3 references | 1 reference | 1 reference
7      public enum UnitsIn{pixels, screen_percentage};
   1 reference
8      public UnitsIn unit = UnitsIn.pixels;
   2 references
9      public int width = 50;
   2 references
10     public int height= 50;
   2 references
11     public int xOffset = 0;
   2 references
12     public int yOffset = 0;
   1 reference
13     public bool update = true;
   5 references | 5 references | 4 references | 4 references
14     private int hsize, vsize, hloc, vloc;
15
   0 references
16     void Start (){
17         AdjustCamera ();
18     }
19
   0 references
20     void Update (){
21         if(update)
22             AdjustCamera ();
23     }
24
```

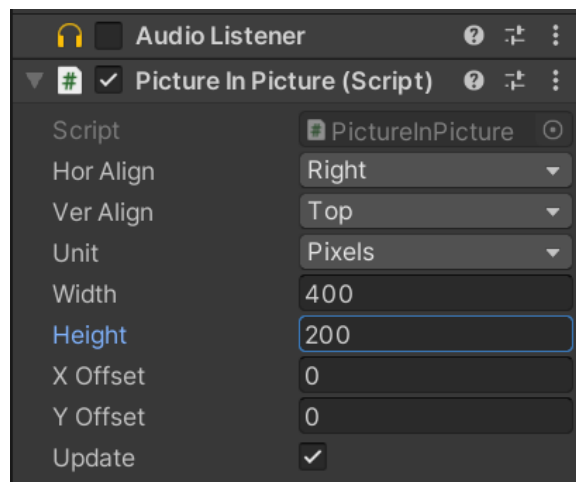
Assets > Script > PictureInPicture.cs > PictureInPicture

```
25     void AdjustCamera(){
26         int sw = Screen.width;
27         int sh = Screen.height;
28         float swPercent = sw * 0.01f;
29         float shPercent = sh * 0.01f;
30         float xOffPercent = xOffset * swPercent;
31         float yOffPercent = yOffset * shPercent;
32         int xOff;
33         int yOff;
34
35         if(unit == UnitsIn.screen_percentage){
36             hsize = width * (int)swPercent;
37             vsize = height * (int)shPercent;
38             xOff = (int)xOffPercent;
39             yOff = (int)yOffPercent;
40         } else {
41             hsize = width;
42             vsize = height;
43             xOff = xOffset;
44             yOff = yOffset;
45         }
46
47         switch (horAlign) {
48             case hAlignment.left:
49                 hloc = xOff;
50                 break;
51
52             case hAlignment.right:
53                 int justifiedRight = (sw - hsize);
54                 hloc = (justifiedRight - xOff);
55                 break;
56
57             case hAlignment.center:
58                 float justifiedCenter = (sw * 0.5f) - (hsize * 0.5f);
59                 hloc = (int)(justifiedCenter - xOff);
60                 break;
61         }
62
63         switch (verAlign) {
64             case vAlignment.top:
65                 int justifiedTop = sh - vsize;
66                 vloc = (justifiedTop - (yOff));
67                 break;
68
69             case vAlignment.bottom:
70                 vloc = yOff;
71                 break;
72
73             case vAlignment.middle:
74                 float justifiedMiddle = (sh * 0.5f) - (vsize * 0.5f);
75                 vloc = (int)(justifiedMiddle - yOff);
76                 break;
77         }
78         GetComponent<Camera>().pixelRect = new Rect(hloc,vloc,hsize,vsize);
79     }
80 }
```

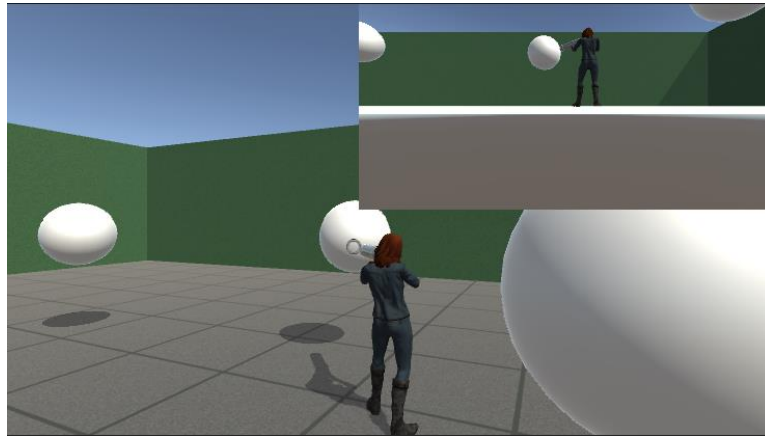
- 7) Setelah menyimpan script C# tersebut, drag script tersebut menuju Camera yang ada pada Hierarchy.



- 8) Pilih Camera lalu pada Inspector hilangkan centang pada komponen Audio Listener dan rubah beberapa parameter yang ada pada Script PictureInPicture seperti gambar dibawah ini:

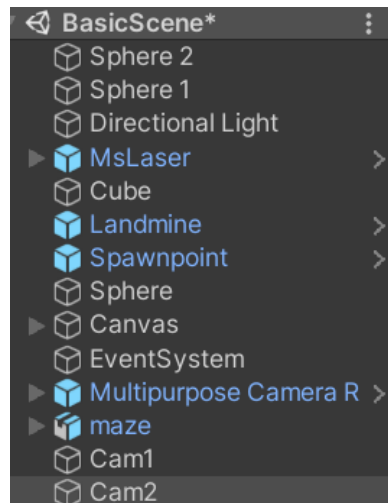


- 9) Silahkan jalankan program, maka pada layar tampilan akan muncul camera dari arah sudut pandang berbeda yang terletak pada kanan atas layar.



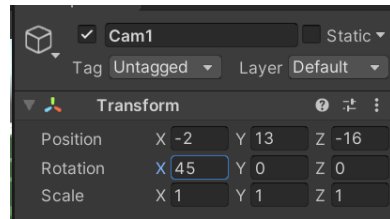
## 2. Membuat dan Melakukan Perpindahan Beberapa Sudut Pandang Camera

- 1) Buatlah Project 3D baru pada Unity 3D.
- 2) Masukkan paket BasicScene kedalam Assets.
- 3) Pada Project View, buka BasicScene kemudian akan muncul karakter animasi dan beberapa objek benda yang telah dibuat.
- 4) Tambahkan 2 new Camera ke dalam scene melalui menu Create yang berada diatas Hierarchy, Pilih Create à Camera. Kemudian beri nama cam1 dan cam2.

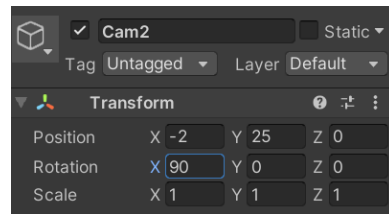


- 5) Pada Inspector, rubahlah posisi camera pada cam1 dan cam2 seperti dibawah ini.

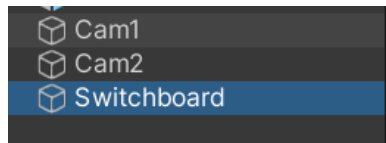
Cam1:



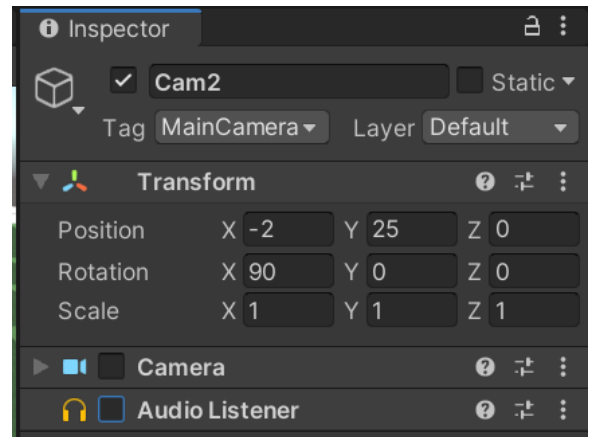
Cam2:



- 6) Buatlah GameObject baru menggunakan menu Create pada Hierarchy. Pilih Create → Create Empty. Kemudian ganti nama menjadi Switchboard.



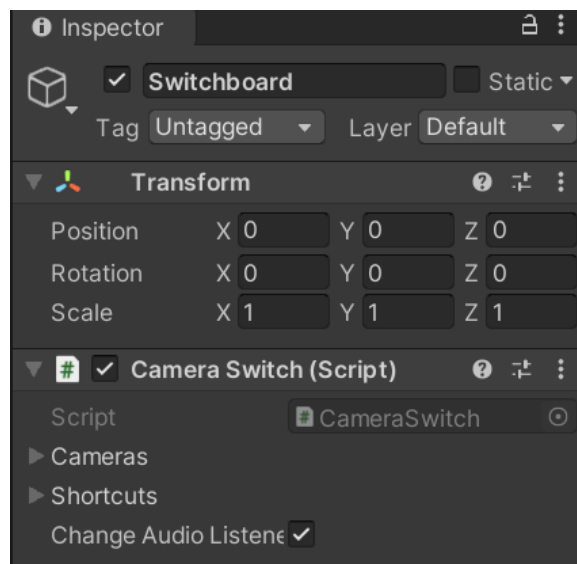
- 7) Dari Inspector pada cam1 dan cam2, hilangkan tanda centang pada komponen Camera dan Audio Listener. Serta atur kolom Tag menjadi MainCamera.



- 8) Kemudian buatlah script C# dengan code seperti dibawah ini kemudian berilah nama CameraSwitch.

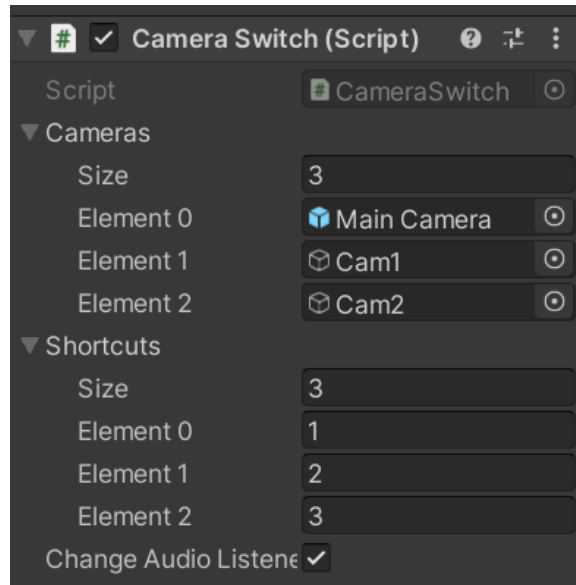
```
CameraSwitch.cs X
Assets > Script > CameraSwitch.cs > CameraSwitch > SwitchCamera(int index)
1 using UnityEngine;
2
3 public class CameraSwitch : MonoBehaviour {
4     public GameObject[] cameras;
5     public string[] shortcuts;
6     public bool changeAudioListener = true;
7
8     void Update () {
9
10        if (Input.anyKeyDown) {
11            for (int i=0; i<cameras.Length; i++) {
12                if (Input.GetKeyDown (shortcuts [i]))
13                    SwitchCamera (i);
14            }
15        }
16    }
17
18    void SwitchCamera ( int index ){
19        for(int i = 0; i<cameras.Length; i++){
20            if(i != index){
21                cameras[i].GetComponent<Camera>().enabled = false;
22                if(changeAudioListener)
23                    cameras[i].GetComponent<AudioListener>().enabled = false;
24            } else {
25                cameras[i].GetComponent<Camera>().enabled = true;
26                if(changeAudioListener)
27                    cameras[i].GetComponent<AudioListener>().enabled = true;
28            }
29        }
30    }
31 }
```

- 9) Setelah itu drag script C# tersebut menuju GameObject Switchboard.



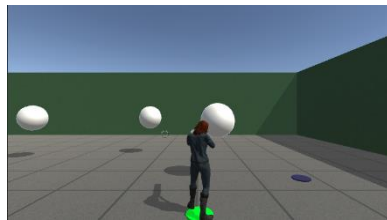


- 10) Pada Inspector, atur ukuran Cameras dan Shortcuts dengan “sizes = 3”. Kemudian, drag dan arahkan secara urut mulai dari Main Camera (berada di Multipurpose Camera Rig → Pivot), cam1, dan cam2 menuju ke setiap slots yang ada pada Cameras. Kemudian pada Shortcuts ketikkan angka 1, 2, dan 3 secara urut pada slots yang tersedia. Seperti gambar di bawah ini:



- 11) Jalankan program dan coba tekan angka 1, 2, dan 3 pada keyboard, maka akan terjadi perubahan pada sudut pandang camera utama.

1



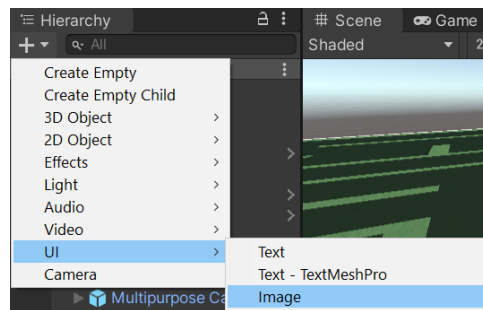
2



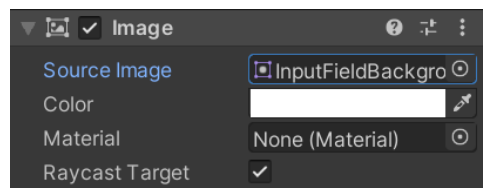
3



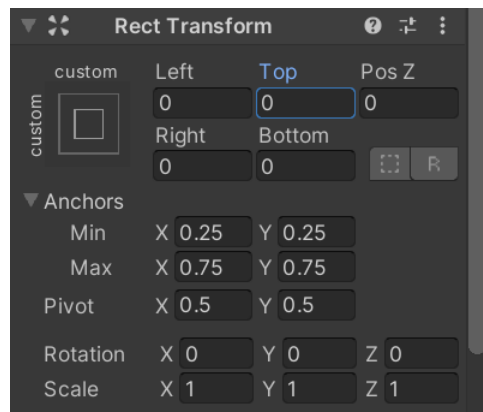
3. Membuat Hasil Tangkapan (Capture) Layar yang Berada di Dalam Frame
  - 1) Buatlah Project 3D baru pada Unity 3D.
  - 2) Masukkan paket BasicScene ke dalam Assets.
  - 3) Pada Project View, buka BasicScene kemudian akan muncul karakter animasi dan beberapa objek benda yang telah dibuat. Didalam paket tersebut juga sudah terdapat fitur Canvas untuk UI elements.
  - 4) Buatlah UI Image dari menu Create yang ada pada Hierarchy, Pilih Create → UI → Image. Setelah itu akan muncul Image yang berada pada Canvas kemudian ganti nama tersebut menjadi frame.



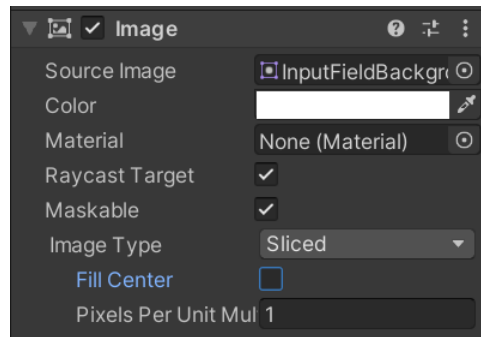
- 5) Pilih frame pada Hierarchy, kemudian pada komponen Image (Script) yang berada di Inspector atur kolom Source Image menjadi InputFieldBackground. Seperti pada gambar di bawah ini:



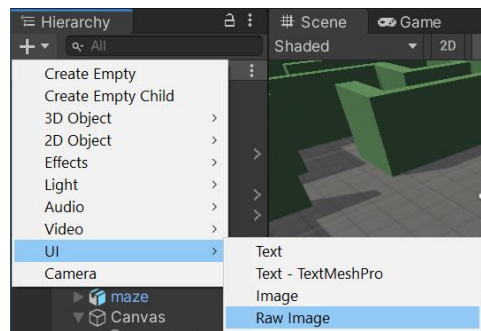
- 6) Pada Inspector milik Frame, Rubahlah Rect Transform sesuai dengan gambar dibawah ini:



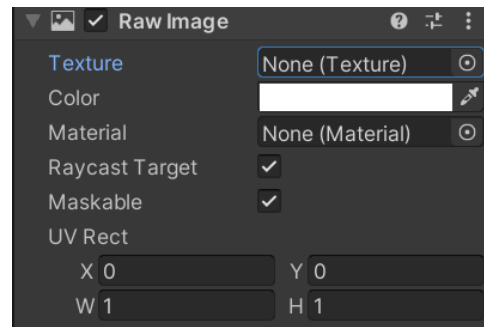
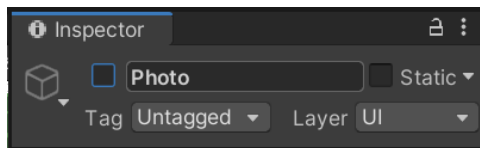
- 7) Pada komponen Image (Script), hilangkan centang pada Fill Center.



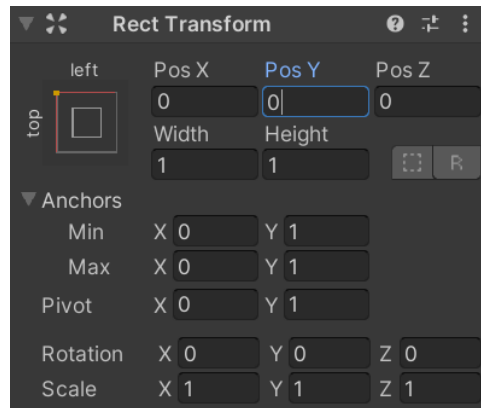
- 8) Buatlah UI Raw Image dari menu Create yang terdapat pada Hierarchy, Pilih Create → UI → RawImage. Setelah itu akan muncul pada Canvas GameObject kemudian ganti nama menjadi Photo.



- 9) Pada Hierarchy pilih Photo, kemudian pada Inspector carilah komponen Raw Image (Script) dan atur kolom Texture menjadi None (Texture). Setelah itu hilangkan centang pada kolom Photo yang berada di atas.



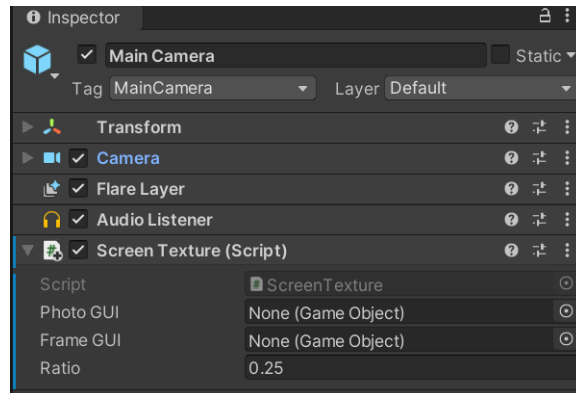
10) Langkah selanjutnya atur Rect Transform pada Photo sesuai gambar dibawah ini:



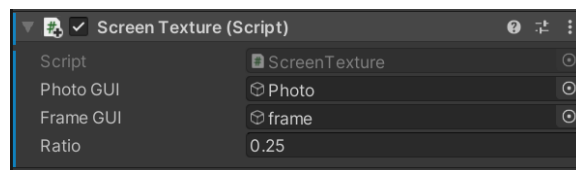
11) Kemudian buatlah script C# dengan code seperti dibawah ini kemudian berilah nama ScreenTexture.

```
ScreenTexture.cs X
Assets > Script > ScreenTexture.cs
1 using UnityEngine;
2 using UnityEngine.UI;
3 using System.Collections;
4 public class ScreenTexture : MonoBehaviour {
5     public GameObject photoGUI;
6     public GameObject frameGUI;
7     public float ratio = 0.25f;
8     void Update () {
9         if (Input.GetKeyUp (KeyCode.Mouse0))
10             StartCoroutine(CaptureScreen());
11     }
12     IEnumerator CaptureScreen () {
13         photoGUI.SetActive (false);
14         int sw = Screen.width;
15         int sh = Screen.height;
16         RectTransform frameTransform = frameGUI.GetComponent<RectTransform> ();
17         Rect framing = frameTransform.rect;
18         Vector2 pivot = frameTransform.pivot;
19         Vector2 origin = frameTransform.anchorMin;
20         origin.x *= sw;
21         origin.y *= sh;
22         float xOffset = pivot.x * framing.width;
23         origin.x += xOffset;
24         float yOffset = pivot.y * framing.height;
25         origin.y += yOffset;
26         framing.x += origin.x;
27         framing.y += origin.y;
28         int textWidth = (int)framing.width;
29         int textHeight = (int)framing.height;
30         Texture2D texture = new Texture2D(textWidth, textHeight);
31         yield return new WaitForEndOfFrame();
32         texture.ReadPixels(framing, 0, 0);
33         texture.Apply();
34         photoGUI.SetActive (true);
35         Vector3 photoScale = new Vector3 (framing.width * ratio, framing.height * ratio, 1);
36         photoGUI.GetComponent<RectTransform> ().localScale = photoScale;
37         photoGUI.GetComponent<RawImage>().texture = texture;
38     }
39 }
```

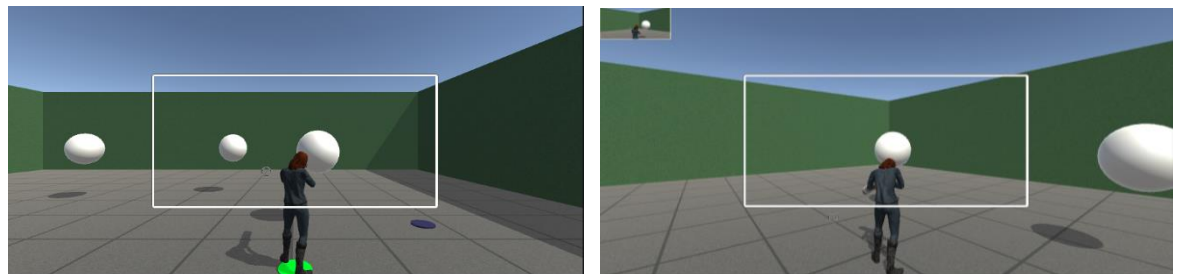
- 12) Setelah itu drag Script C# tersebut menuju Main Camera yang berada pada Multipurpose Camera Rig → Pivot.



- 13) Pada Inspector milik Main Camera, carilah komponen Screen Texture (Script) kemudian atur kolom Photo GUI dengan memilih Photo. Sama halnya dengan kolom Frame GUI dengan memilih Frame.



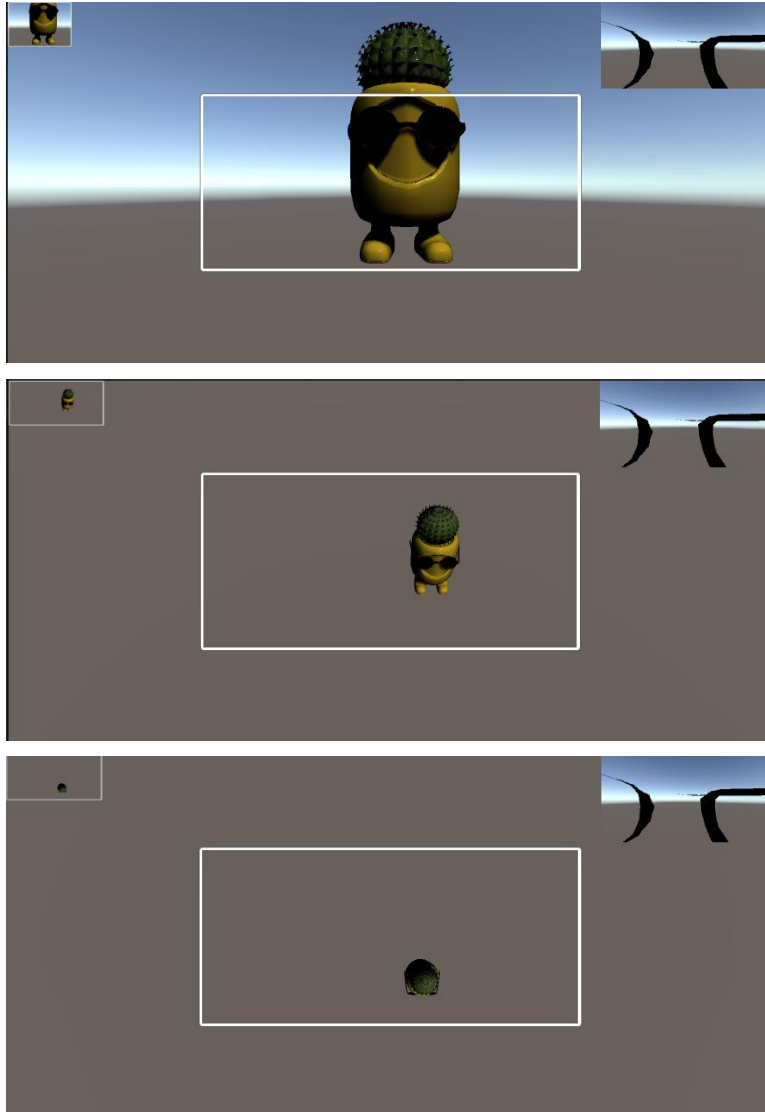
- 14) Jalankan program, maka anda akan bisa meng-capture layar yang berada dalam frame dengan cara menekan klik kiri pada mouse. Hasil capture akan tampil pada pojok kiri atas layar.



## B. TUGAS PRAKTIKUM

- 1) Buatlah sebuah project 3D yang didalamnya terdapat 3 Fitur Using Cameras pada modul diatas beserta laporan tugas praktikumnya.

Hasil:



Link Video Youtube: <https://youtu.be/mDGapJ0zaZo>

Link Github: [https://github.com/kinanpermata/TI-3D\\_17\\_Komputasi-Multimedia](https://github.com/kinanpermata/TI-3D_17_Komputasi-Multimedia)