

MODUL 5 – Using Cameras

A. TUJUAN

- Mahasiswa dapat membuat tampilan camera dari arah sudut pandang lain di dalam layar.
- Mahasiswa dapat membuat dan melakukan perpindahan beberapa sudut pandang camera.
- Mahasiswa dapat membuat hasil tangkapan (*capture*) layar yang berada di dalam frame.

B. PETUNJUK

1. Awali setiap kegiatan praktikum dengan berdoa
2. Baca dan pahami tujuan, dasar teori, dan latihan-latihan praktikum dengan baik
3. Kerjakan tugas-tugas praktikum dengan baik, sabar dan jujur
4. Tanyakan kepada dosen apabila ada hal-hal yang kurang jelas

C. ALOKASI WAKTU: 6 jam pelajaran

D. DASAR TEORI

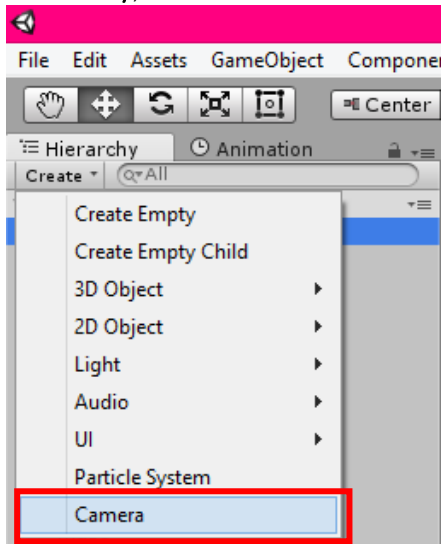
Penggunaan camera pada Unity ini sangat penting sekali terutama pada objek 3D. Hal tersebut dikarenakan adanya perbedaan antara objek 2D yang hanya bisa dilihat dari satu sudut pandang saja, sedangkan objek 3D dapat dilihat dari segala sudut pandang. Dari segi posisi juga terdapat perbedaan antara objek 2D dan 3D, yaitu jika pada objek 3D terdapat koordinat sumbu “z” yang sangat berpengaruh pada posisi objek yang kita buat. Pada minggu ini akan mempelajari beberapa hal untuk membuat dan menentukan tata letak sudut pandang camera dengan fitur-fitur camera yang ada pada Unity, antara lain:

- Tampilan camera dari arah sudut pandang lain di dalam layar utama, dimana di dalam satu layar tampilan game terdapat layar camera yang mempunyai sudut pandang yang berbeda dengan camera utama.
- Melakukan perpindahan beberapa sudut pandang camera dalam suatu game, dimana pengguna dapat merubah layar tampilan secara keseluruhan dengan sudut pandang camera yang berbeda.
- Hasil tangkapan (*capture*) layar yang berada di dalam frame, dimana hal ini memungkinkan pengguna untuk menangkap gambar dalam suatu game yang hasil tangkapan tersebut berasal dari sudut pandang camera utama.

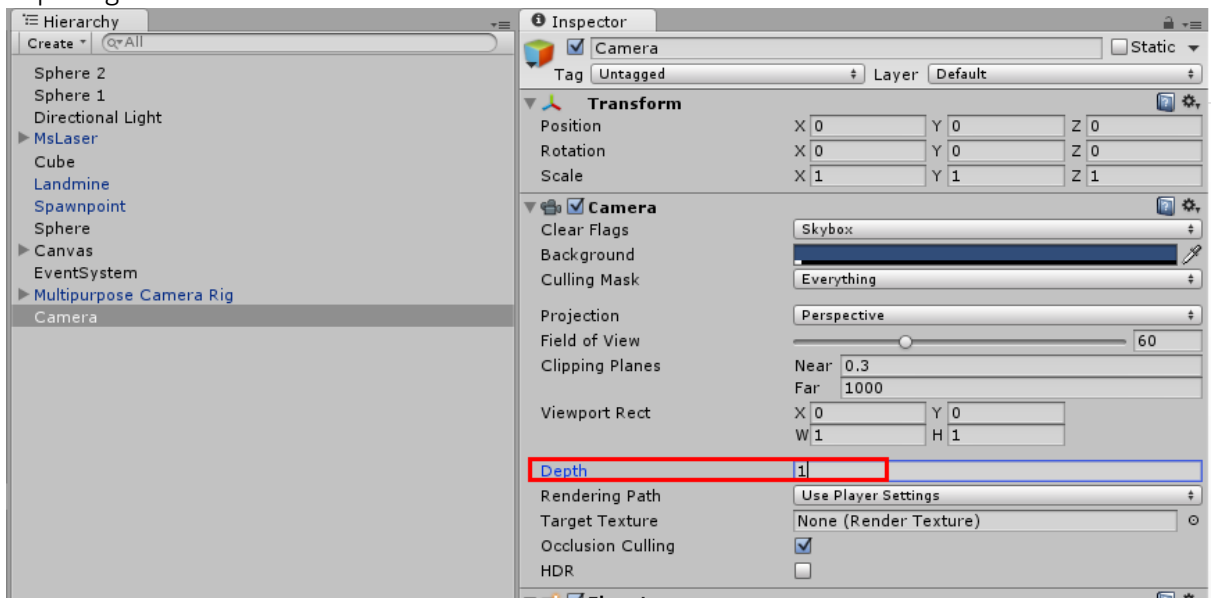
E. LATIHAN PRAKTIKUM

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*.

```

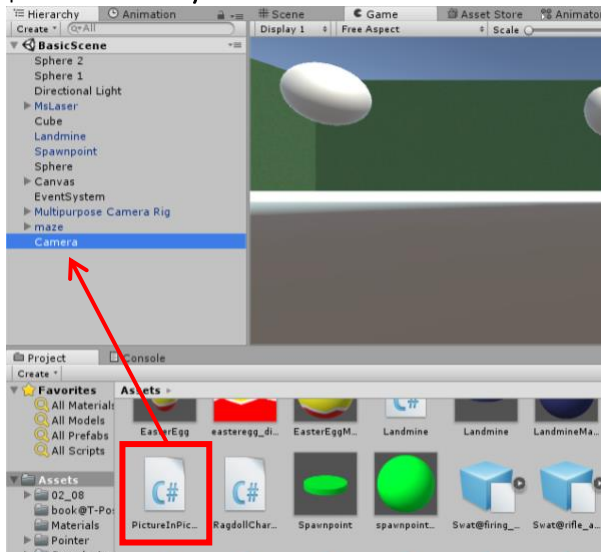
using UnityEngine;

public class PictureInPicture: MonoBehaviour {
    public enum hAlignment{left, center, right};
    public enum vAlignment{top, middle, bottom};
    public hAlignment horAlign = hAlignment.left;
    public vAlignment verAlign = vAlignment.top;
    public enum UnitsIn{pixels, screen_percentage};
    public UnitsIn unit = UnitsIn.pixels;
    public int width = 50;
    public int height= 50;
    public int xOffset = 0;
    public int yOffset = 0;
    public bool update = true;
    private int hsize, vsize, hloc, vloc;

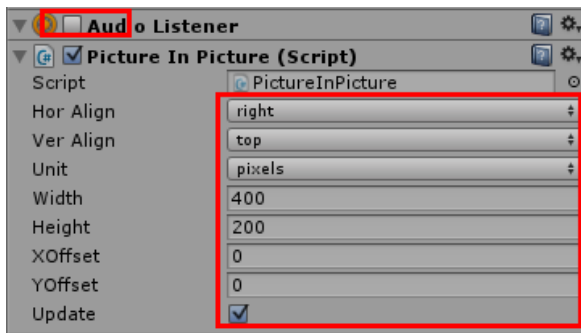
    void Start (){
        AdjustCamera ();
    }
    void Update (){
        if(update)
            AdjustCamera ();
    }
    void AdjustCamera(){
        int sw = Screen.width;
        int sh = Screen.height;
        float swPercent = sw * 0.01f;
        float shPercent = sh * 0.01f;
        float xOffPercent = xOffset * swPercent;
        float yOffPercent = yOffset * shPercent;
        int xOff;
        int yOff;
        if(unit == UnitsIn.screen_percentage){
            hsize = width * (int)swPercent;
            vsize = height * (int)shPercent;
            xOff = (int)xOffPercent;
            yOff = (int)yOffPercent;
        } else {
            hsize = width;
            vsize = height;
            xOff = xOffset;
            yOff = yOffset;
        }
        switch (horAlign) {
            case hAlignment.left:
                hloc = xOff;
                break;
            case hAlignment.right:
                int justifiedRight = (sw - hsize);
                hloc = (justifiedRight - xOff);
                break;
            case hAlignment.center:
                float justifiedCenter = (sw * 0.5f) - (hsize * 0.5f);
                hloc = (int)(justifiedCenter - xOff);
                break;
        }
        switch (verAlign) {
            case vAlignment.top:
                int justifiedTop = sh - vsize;
                vloc = (justifiedTop - (yOff));
                break;
            case vAlignment.bottom:
                vloc = yOff;
                break;
            case vAlignment.middle:
                float justifiedMiddle = (sh * 0.5f) - (vsize * 0.5f);
                vloc = (int)(justifiedMiddle - yOff);
                break;
        }
        GetComponent<Camera>().pixelRect = new Rect(hloc,vloc,hsize,vsize);
    }
}

```

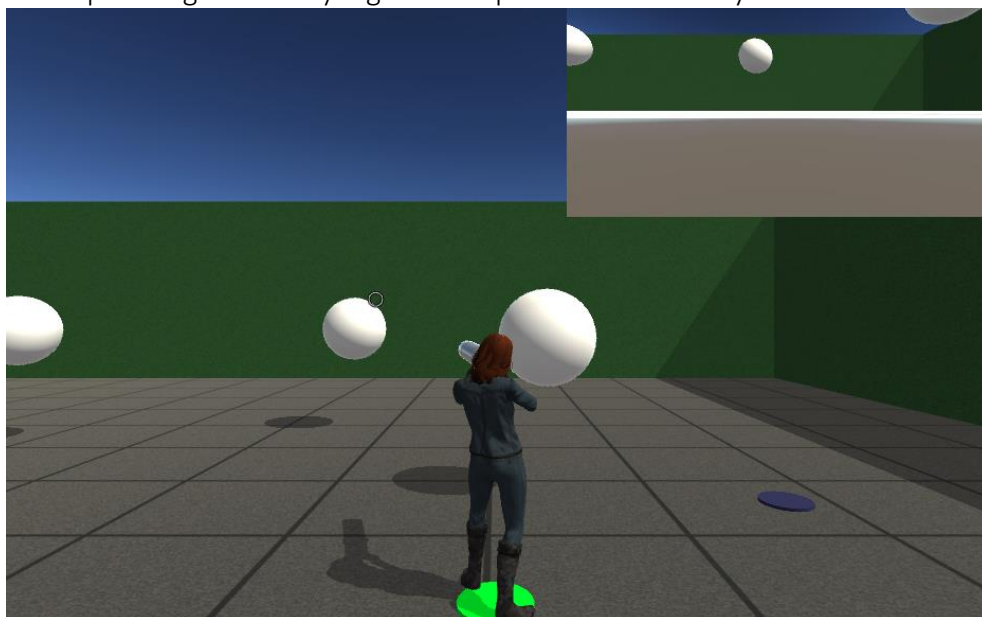
7. Setelah menyimpan script C# tersebut, drag script tersebut menuju **Camera** yang ada pada **Hierarchy**.



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:

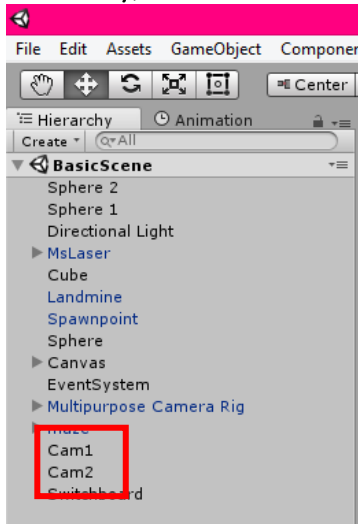


9. Silahkan jalankan program, maka pada layar tampilan akan muncul camera dari arah sudut pandang berbeda yang terletak pada kanan atas layar.

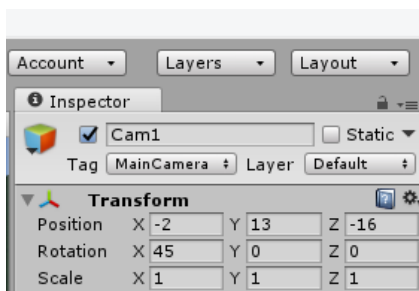


2. Membuat dan Melakukan Perpindahan Beberapa Sudut Pandang Camera

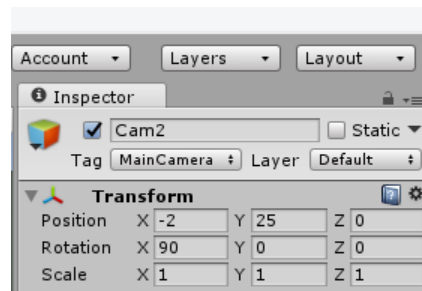
1. Buatlah Project 3D baru pada Unity 3D.
2. Masukkan paket *BasicScene* kedalam Assets.
3. Pada **Project View**, buka *BasicScene* kemudian akan muncul karakter animasi dan beberapa objek benda yang telah dibuat.
4. Tambahkan 2 new **Camera** ke dalam scene melalui menu **Create** yang berada diatas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



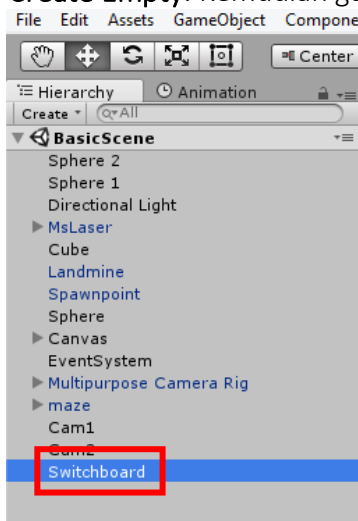
5. Pada **Inspector**, rubahlah posisi camera pada **cam1** dan **cam2** seperti dibawah ini.
Cam1:



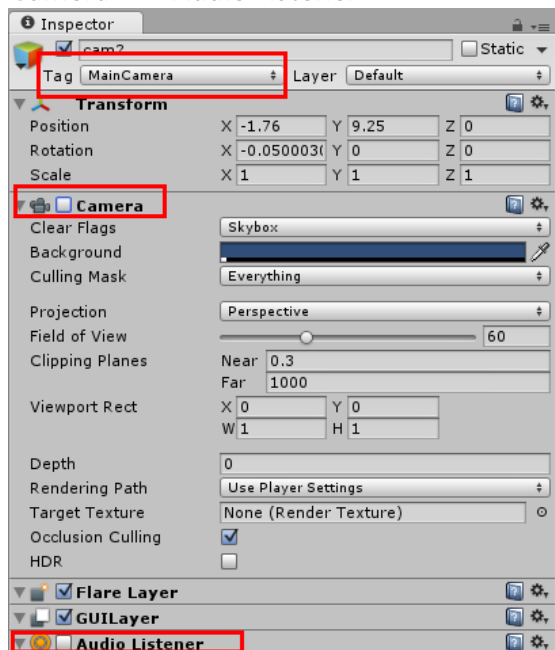
Cam2:



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**.

```
using UnityEngine;

public class CameraSwitch : MonoBehaviour {
    public GameObject[] cameras;
    public string[] shortcuts;
    public bool changeAudioListener = true;

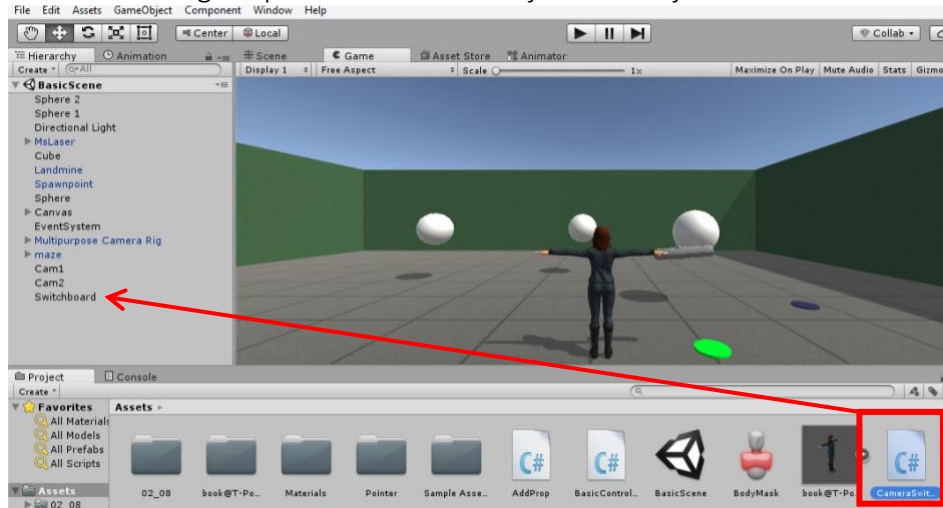
    void Update () {
        if (Input.anyKeyDown) {
            for (int i=0; i<cameras.Length; i++) {
                if (Input.GetKeyDown (shortcuts [i]))
                    SwitchCamera (i);
            }
        }
    }

    void SwitchCamera ( int index ) {
        for(int i = 0; i<cameras.Length; i++){
            if(i != index){
                cameras[i].GetComponent<Camera>().enabled = false;
                if(changeAudioListener)

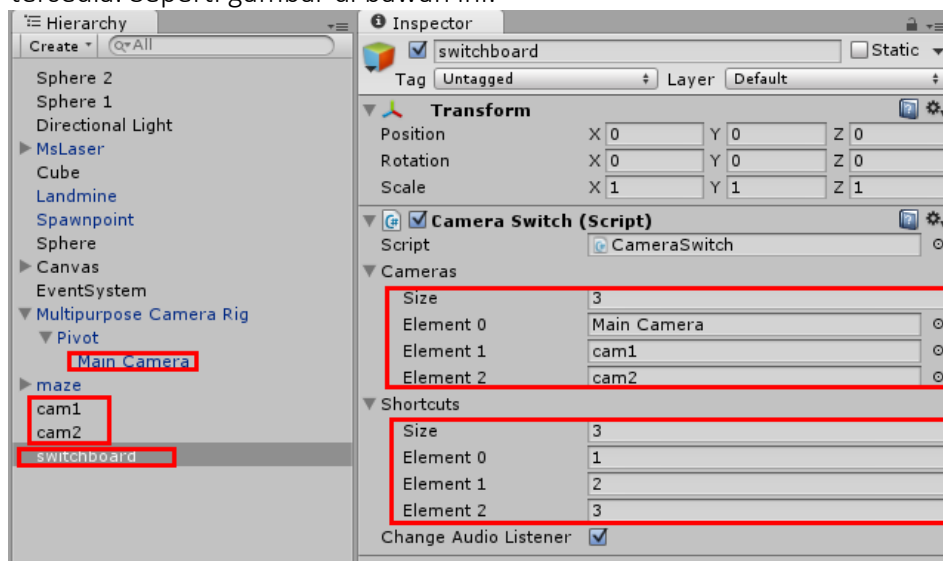
                cameras[i].GetComponent<AudioListener>().enabled = false;
            } else {
                cameras[i].GetComponent<Camera>().enabled = true;
                if(changeAudioListener)

                cameras[i].GetComponent<AudioListener>().enabled = true;
            }
        }
    }
}
```

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:

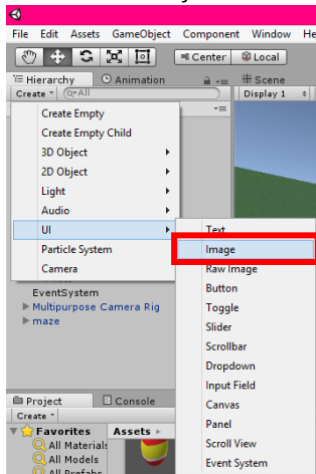


11. Jalankan program dan coba tekan angka 1, 2, dan 3 pada keyboard, maka akan terjadi perubahan pada sudut pandang camera utama.

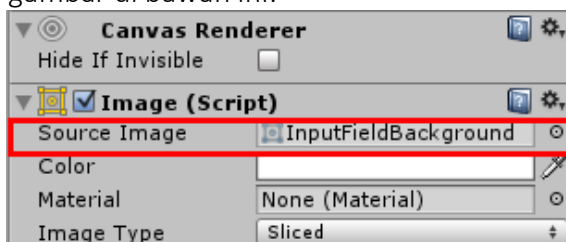
3. Membuat Hasil Tangkapan (*Capture*) Layar yang Berada di Dalam Frame

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**.

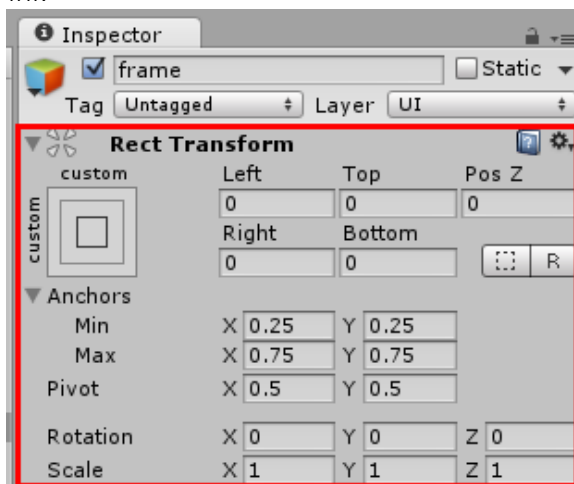
4. Buatlah **UI Image** dari menu **Create** yang ada pada **Hierarchy**, Pilih **Create → UI → Image**. Setelah itu akan muncul **Image** yang berada pada **Canvas** kemudian ganti nama tersebut menjadi **frame**.



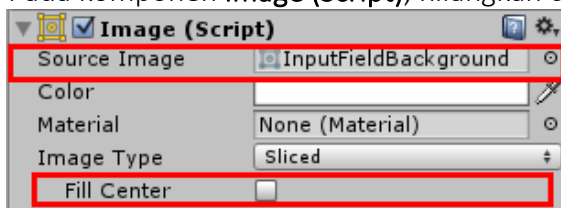
5. Pilih **frame** pada **Hierarchy**, kemudian pada komponen **Image (Script)** yang berada di **Inspector** atur kolom **Source Image** menjadi **InputFieldBackground**. Seperti pada gambar di bawah ini:



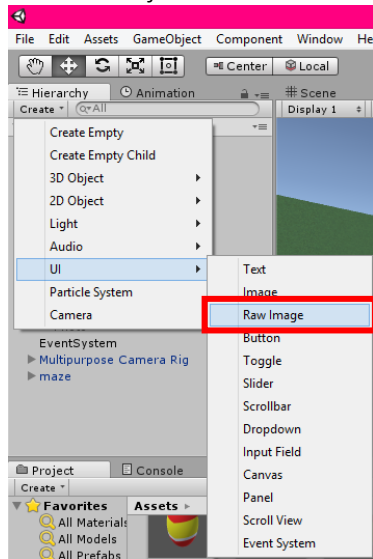
6. Pada **Inspector** milik **Frame**, Rubahlah **Rect Transform** sesuai dengan gambar dibawah ini:



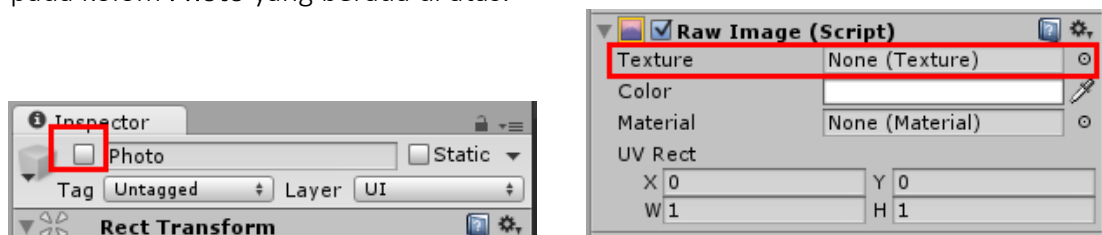
7. Pada komponen **Image (Script)**, hilangkan centang pada **Fill Center**.



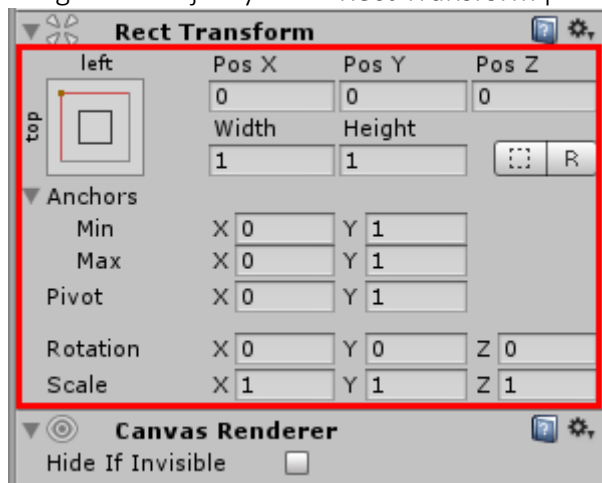
8. Buatlah UI Raw Image dari menu **Create** yang terdapat pada **Hierarchy**, Pilih **Create** → **UI** → **RawImage**. Setelah itu akan muncul pada **Canvas** GameObject kemudian ganti nama menjadi **Photo**.



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**.

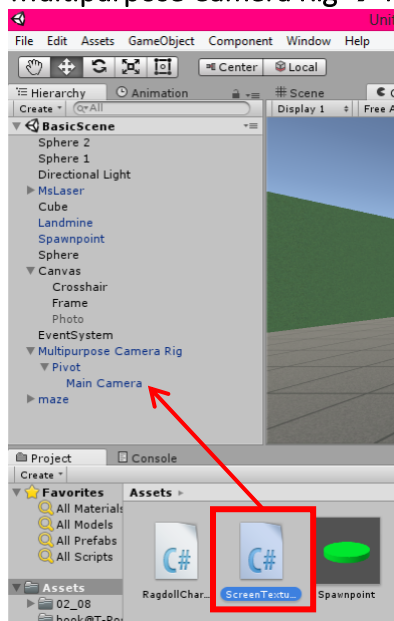
```
using UnityEngine;
using UnityEngine.UI;
using System.Collections;

public class ScreenTexture : MonoBehaviour {
    public GameObject photoGUI;
    public GameObject frameGUI;
    public float ratio = 0.25f;

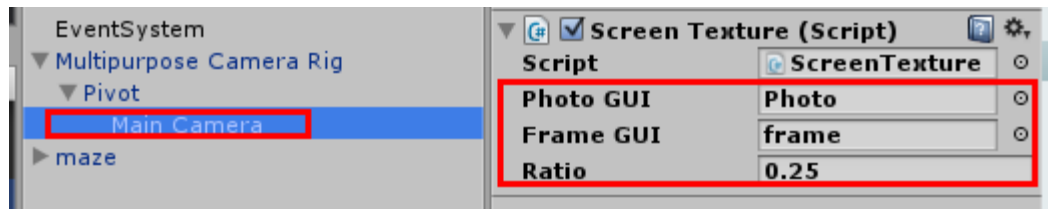
    void Update () {
        if (Input.GetKeyUp (KeyCode.Mouse0))
            StartCoroutine(CaptureScreen());
    }

    IEnumerator CaptureScreen () {
        photoGUI.SetActive (false);
        int sw = Screen.width;
        int sh = Screen.height;
        RectTransform frameTransform =
frameGUI.GetComponent<RectTransform> ();
        Rect framing = frameTransform.rect;
        Vector2 pivot = frameTransform.pivot;
        Vector2 origin = frameTransform.anchorMin;
        origin.x *= sw;
        origin.y *= sh;
        float xOffset = pivot.x * framing.width;
        origin.x += xOffset;
        float yOffset = pivot.y * framing.height;
        origin.y += yOffset;
        framing.x += origin.x;
        framing.y += origin.y;
        int textWidth = (int)framing.width;
        ...
    }
}
```

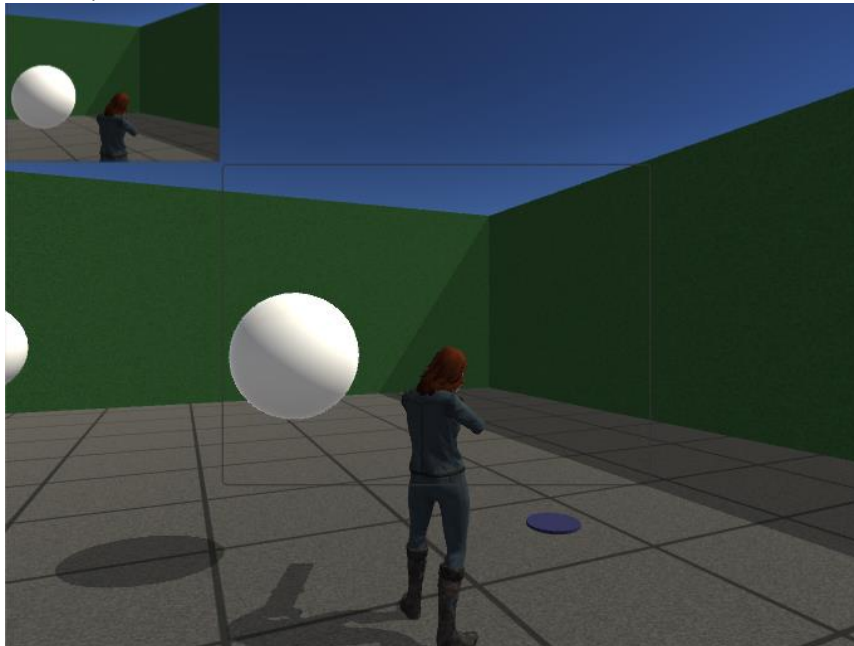
12. Setelah itu drag Script C# tersebut menuju **Main Camera** yang berada pada **Multipurpose Camera Rig → Pivot**.



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.



A. TUGAS PRAKTIKUM

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

--- SELAMAT BELAJAR ---