

LAPORAN TUGAS KOMPUTASI MULTIMEDIA

Modul 10
Lanjutan Lights and Effects

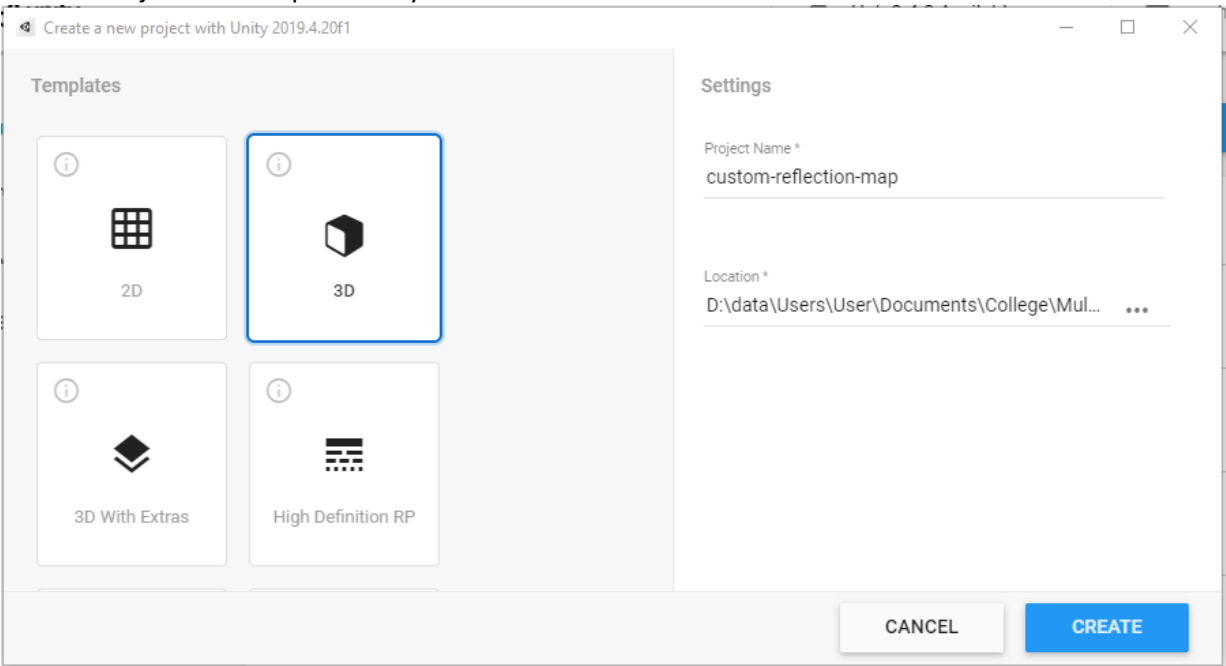
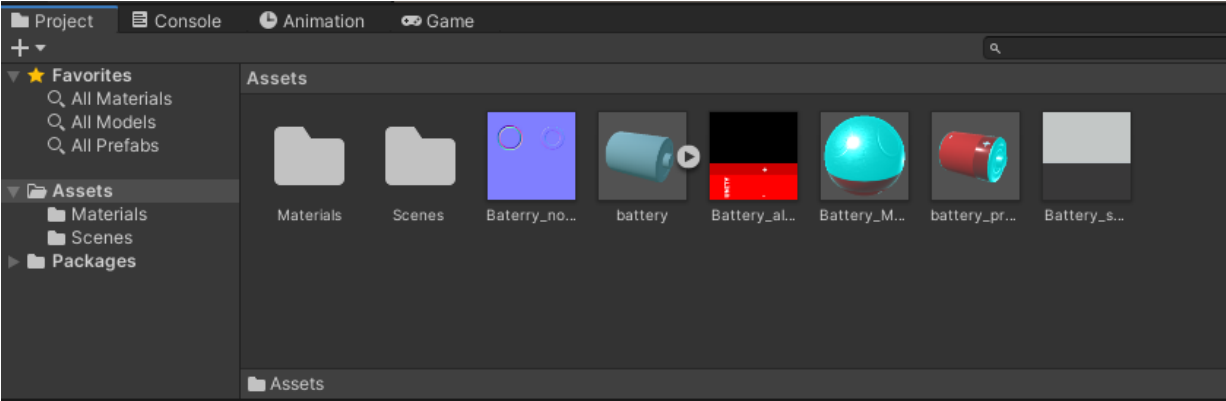
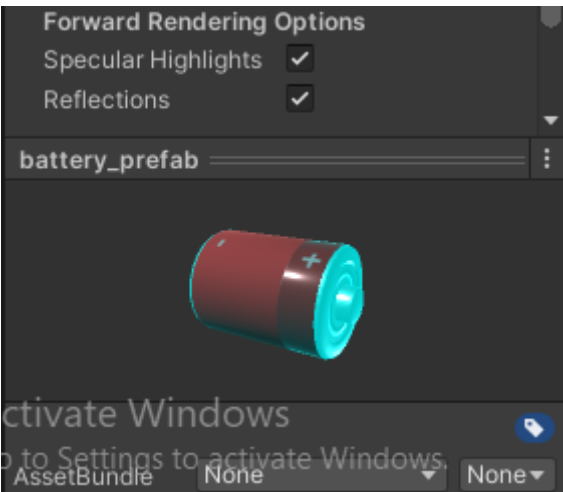


Disusun Oleh :

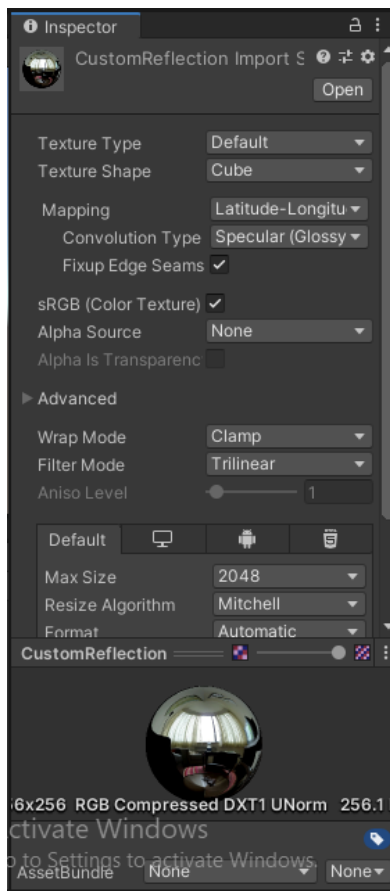
Nama : Hunayn Risatayn
NIM : 1841720148
Kelas : TI-3E

Program Studi D-IV Teknik Informatika
Jurusan Teknologi Informasi
Politeknik Negeri Malang
April 2021

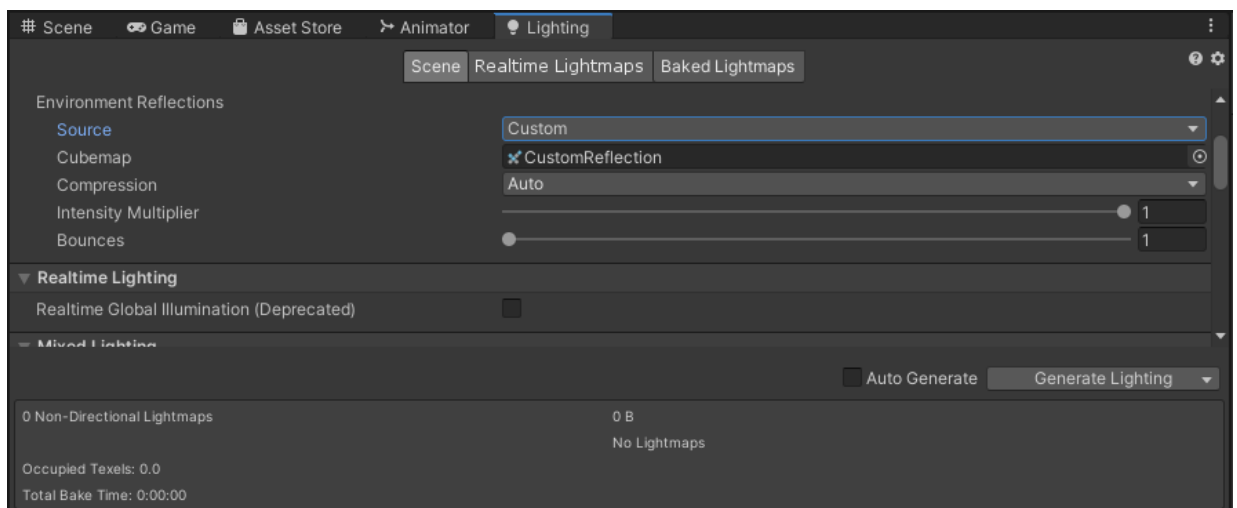
Praktikum – Bagian 1: Menambahkan Reflection Map secara Custom pada Scene

Langkah	Keterangan
1	<p>Buatlah Project 3D baru pada Unity.</p>  <p>Import batteryPrefab.unitypackage kedalam project baru. Pilih object battery_prefab dari folder Asset pada Project View</p>  <p>Perhatikan pada Inspector, expand komponen Material. Karena Map Specular, material telah dapat menampilkan kesan reflective.</p> 

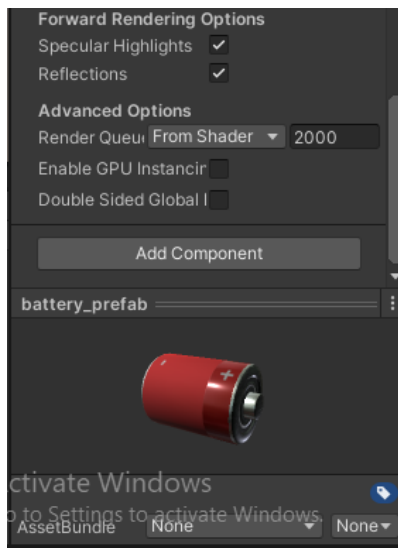
Import image CustomReflection.jpg. Dari Inspector View, pastikan texture typenya ke Default, Texture Shape = Cube, Mappingnya ke Latitude – Longitude Layout (Cylindrical), Convolution Type = Specular (Glossy Reflection), centang Fixup Edge Seams. Kemudian ubah Filter Modenya ke Trilinear dan klik tombol Apply.



Ubah Reflection Skybox dengan reflection map yang baru saja anda buat. Buka jendela Lighting dari Window | Lighting. Pilih Section Scene dan ubah Reflection Source menjadi Custom. Selanjutnya Drag atau pilih CustomReflection yang anda buat kedalam CubeMap.



Perhatikan perbedaan lighting saat menggunakan skybox dan menggunakan reflection map yang baru anda buat.

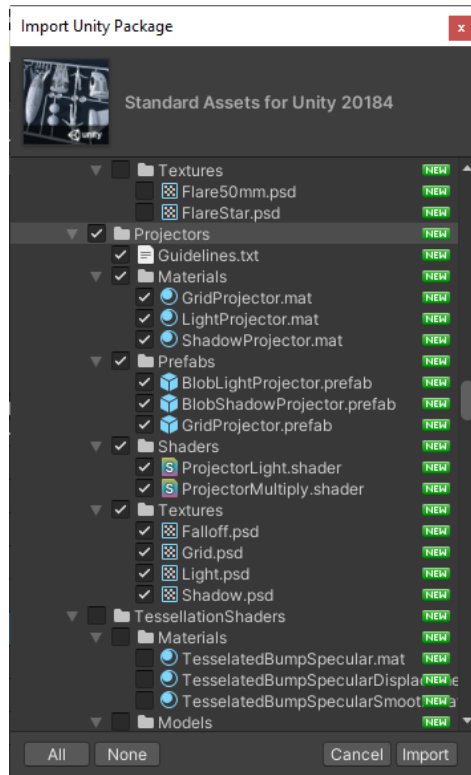


Membuat Laser pointer menggunakan Projector dan Line Renderer

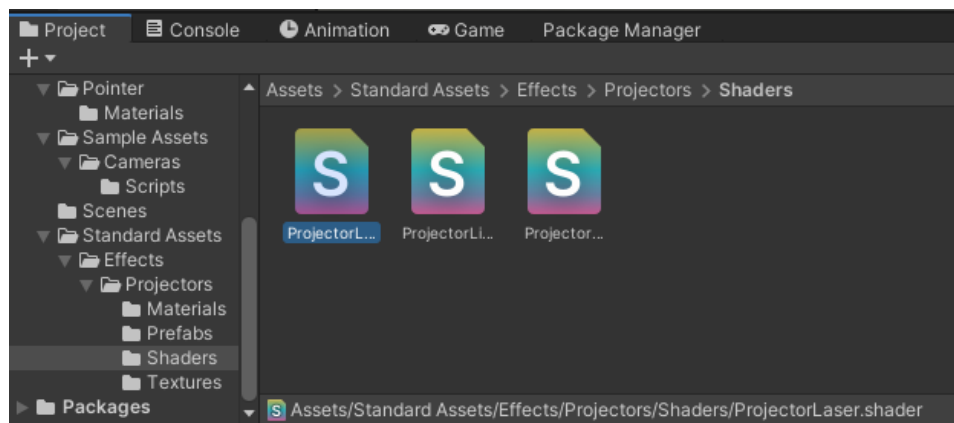
Import BasicScene.unitypackage kedalam project baru. Buka scene yang bernama BasicScene. Ini adalah scene dasar berisi character yang targetnya dikendalikan menggunakan mouse.



Import package Effects dengan menu navigasi Assets | Import Package | Effects. Jika ingin mengimport yang dibutuhkan untuk project ini saja dan tidak semua effect, hilangkan semua centang pada jendela Importing packages dengan meng-klik tombol None, kemudian centang hanya folder Projectors saja. Kemudian klik Import :



Dari View Inspector, cari shader ProjectorLight (didalam folder Assets | Standard Assets | Effects | Projectors | Shaders). Duplicate file shader tersebut dan ubah namanya menjadi ProjectorLaser



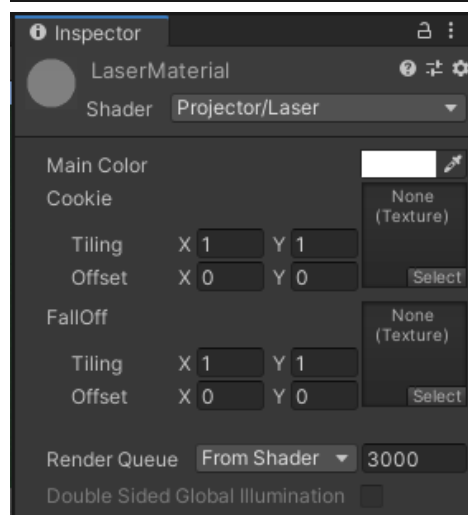
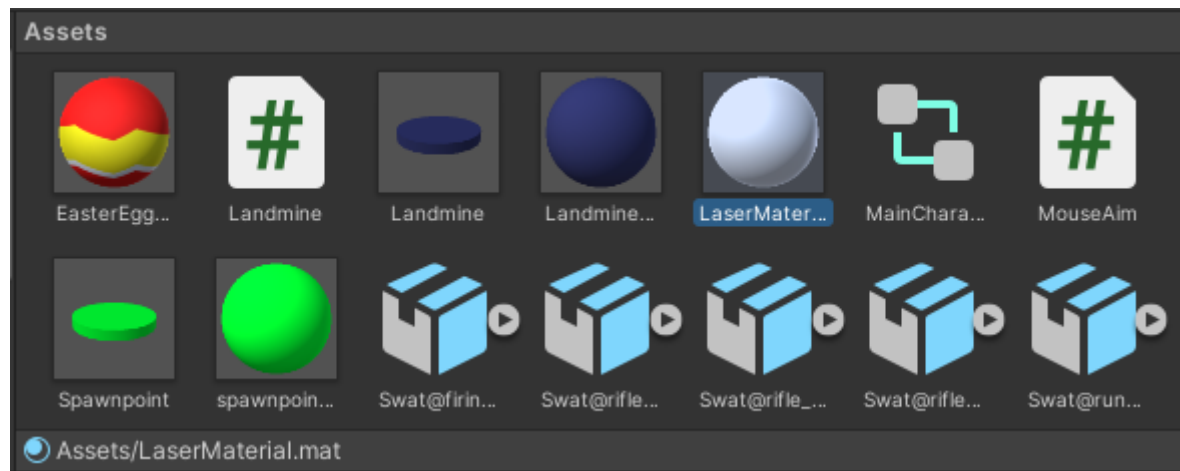
Buka code dari ProjectorLaser, pada baris pertama, ubah Shader "Projector/Light" menjadi Shader "Projector/Laser". Kemudian cari baris kode berisi – Blend DstColor One dan ubah menjadi Blend One One, Simpan dan tutup filenya. (Meng-edit shader menjadi Laser dilakukan untuk mengubah tipe Blend menjadi Additive.)

```

4  Shader "Projector/Laser" {
5      Properties {
6          _Color ("Main Color", Color) = (1,1,1,1)
7          _ShadowTex ("Cookie", 2D) = "" {}
8          _FalloffTex ("Falloff", 2D) = "" {}
9      }
10
11     Subshader {
12         Tags {"Queue"="Transparent"}
13         Pass {
14             ZWrite Off
15             ColorMask RGB
16             Blend One One
17             Offset -1, -1
18         }
19     }
20 }

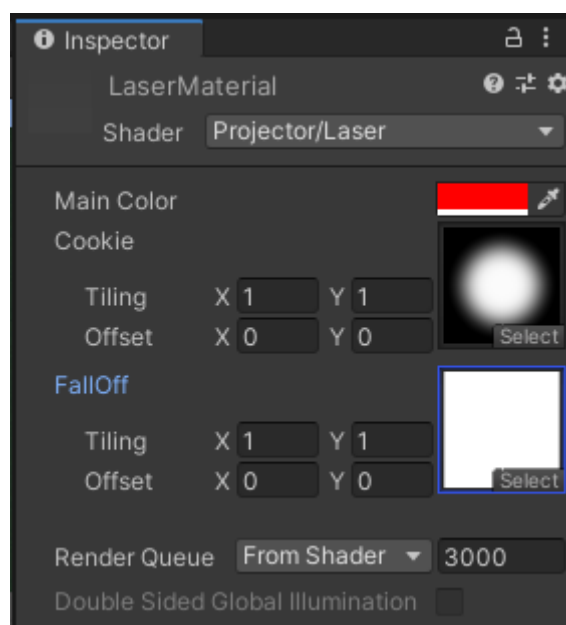
```

Selanjutnya untuk memperbaiki shader, dibutuhkan material. Dari Project View, gunakan menu dropdown Create untuk membuat Material baru. Berikan nama LaserMaterial. Pilih LaserMaterial dari Project view, dari Inspector view, ubah shadernya menjadi Projector/Laser.

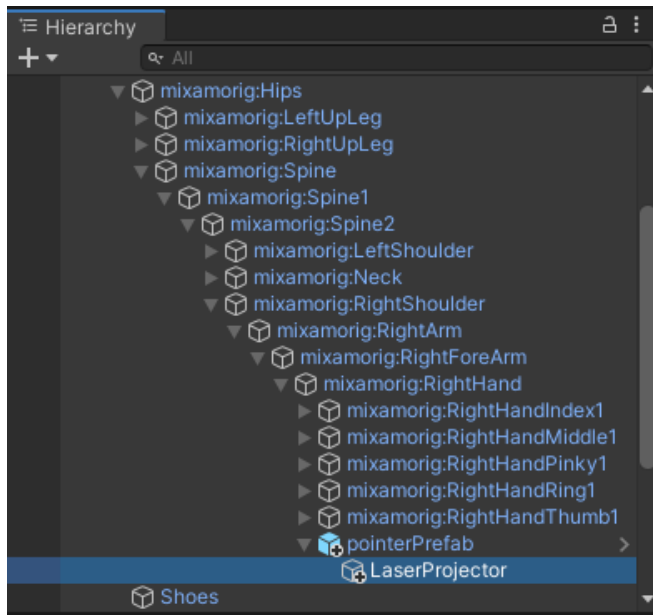


Dari Project view, cari texture Falloff, buka ke dalam image editor, pixel kolom pertama dan kolom terakhir buat sebagai warna hitam, kolom yang lain dibuat sebagai warna putih. Simpan filenya dan kembali lagi ke Unity.

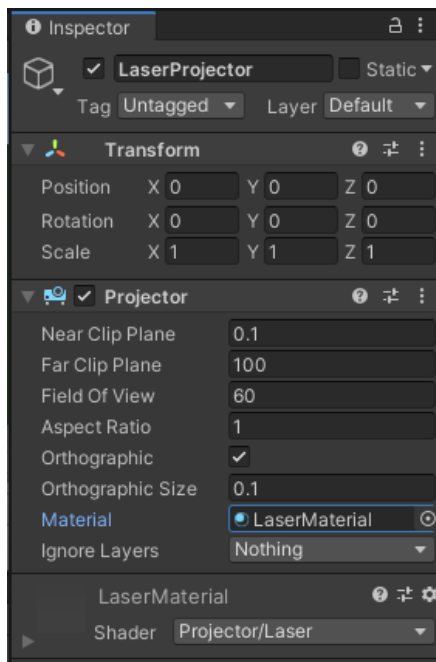
Ubah Main Color dari LaserMaterial menjadi red (RGB: 255,0,0). Kemudian dari slot texture, pilih texture Light sebagai Cookie dan texture Falloff menjadi Falloff.



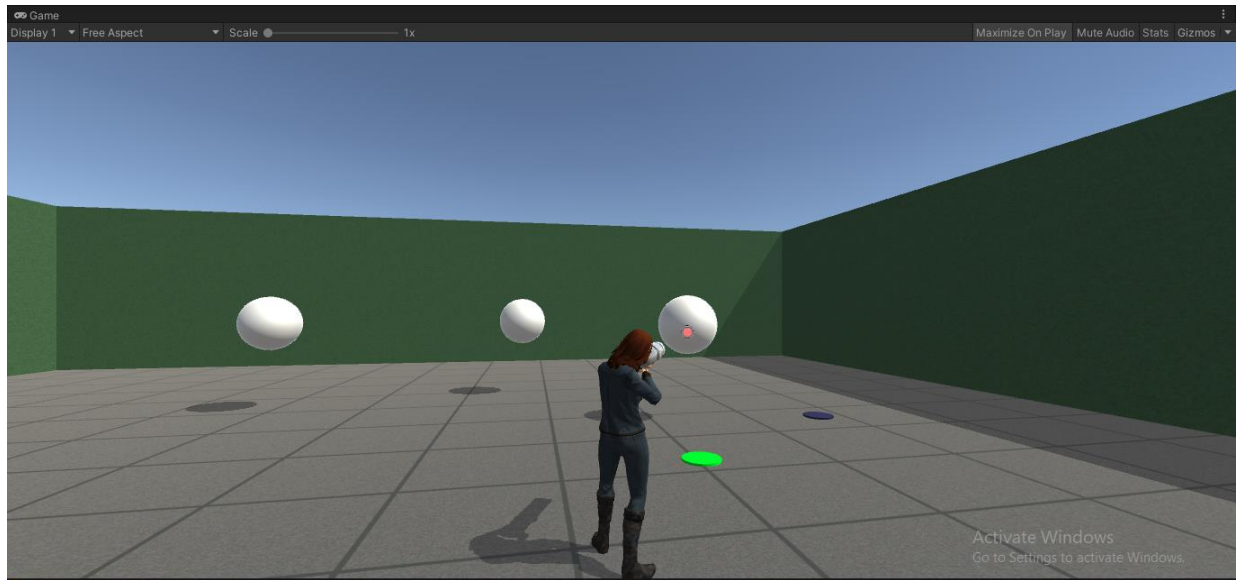
Dari Hierarchy view, Cari dan pilih object pointerPrefab (MsLaser | mixamorig:Hips | mixamorig:Spine | mixamorig:Spine1 | mixamorig:Spine2 | mixamorig:RightShoulder | mixamorig:RightArm | mixamorig:RightForeArm | mixamorig:RightHand | pointerPrefab). Kemudian, dari menu drop-down Create, pilih Create Empty Child. Ubah nama new child dari pointerPrefab sebagai LaserProjector.



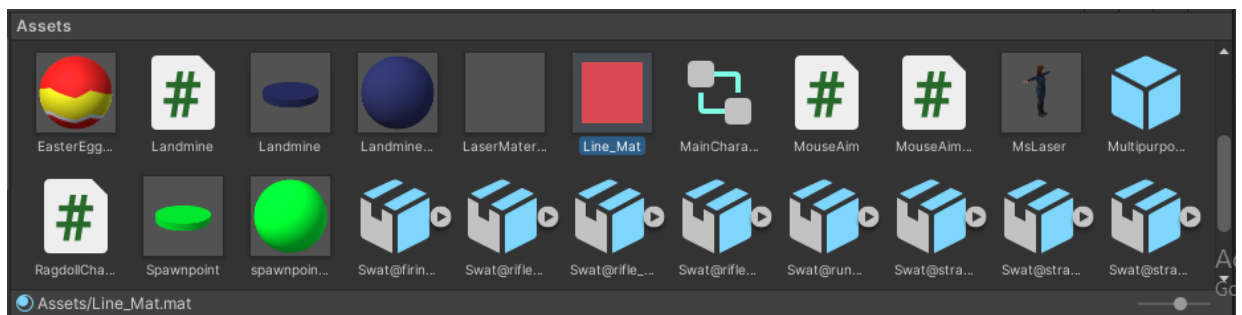
Select object ProjectorLaser. Kemudian dari Inspector View, klik Add Component, kemudian pilih Effect | Projector. Kemudian dari komponen Projector, set option Orthographic sebagai True dan set Orthographic Size menjadi 0.1. Kemudian pilih LaserMaterial dari slot Material.



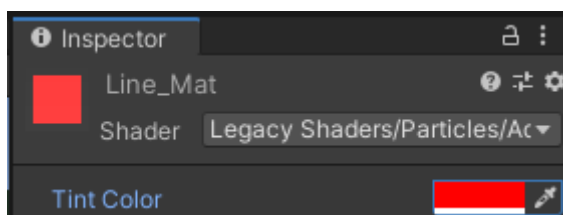
Coba jalankan scene, jika sudah benar maka akan terlihat laser yang berwarna merah menyasar ke target.



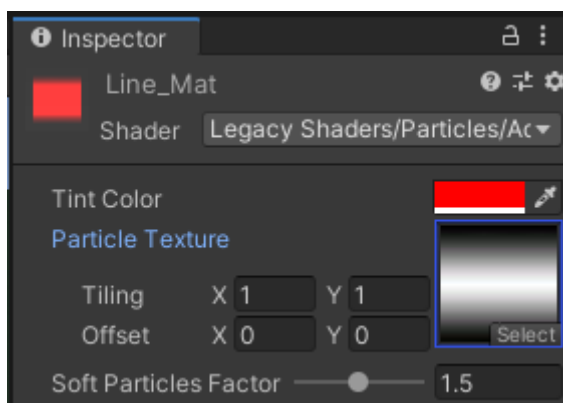
Selanjutnya akan dibuat material untuk komponen Line Renderer. Dari Project View, gunakan menu drop-down Create untuk menambahkan Material baru. Beri nama Line-Mat.



Dari menu Inspector, ubah shader dari Line_Mat menjadi Particles/Additive. Set Tint Colornya menjadi red (RGB: 255,0,0)

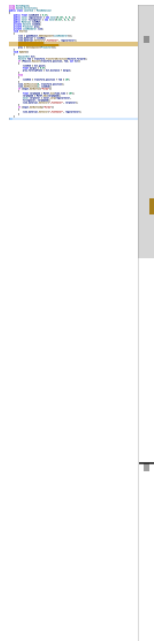


Import file image LineTexture. Set sebagai Particle Texture dari Line_Mat. Set Particles Factor menjadi 1.5.

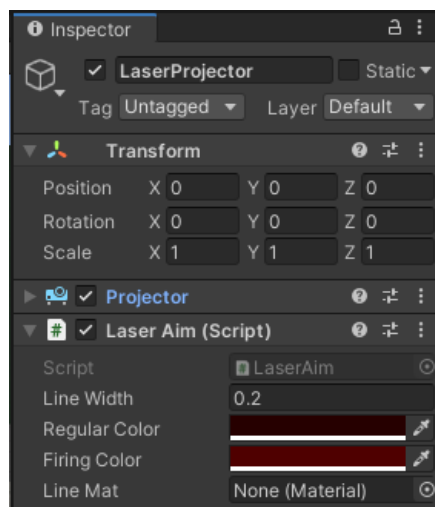


Gunakan menu drop-down Create dari Project view untuk menambahkan script C# yang dinamakan LaserAim. Buka di script editor. Tambahkan script berikut:

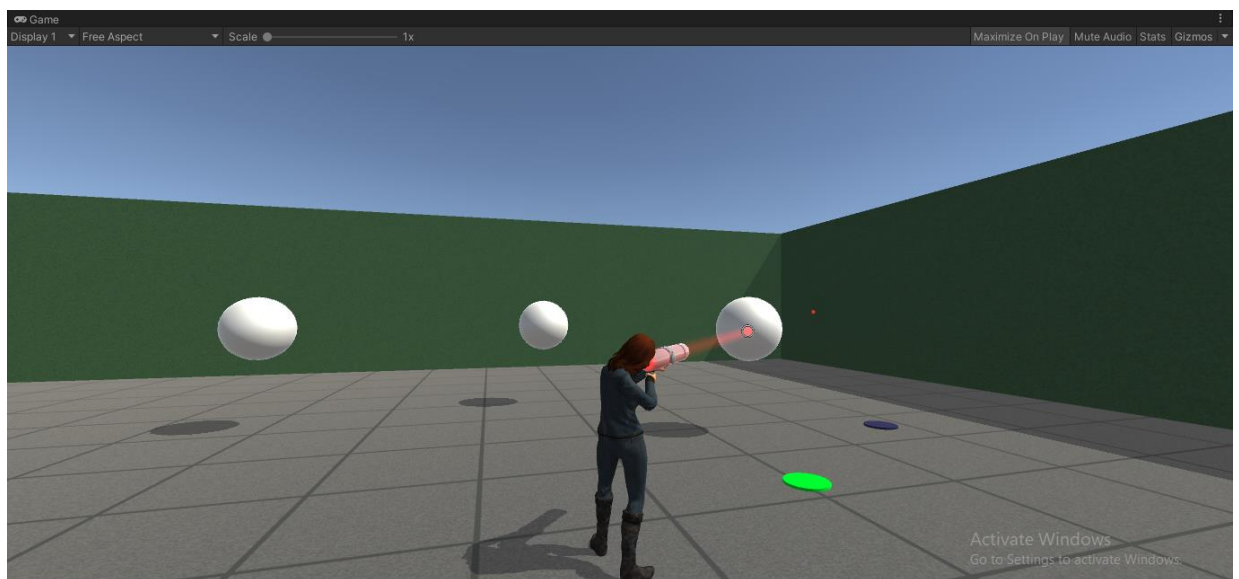
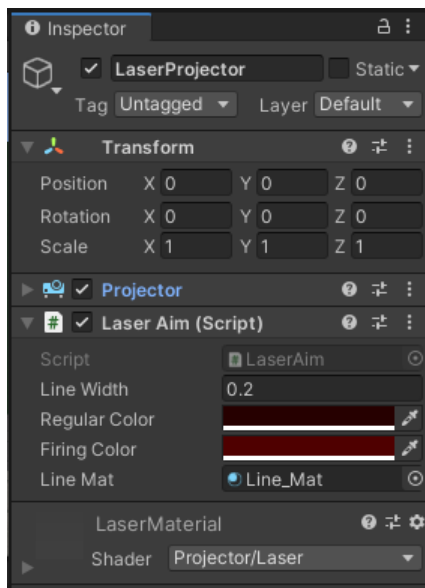
```
Assets > Scripts > LaserAim.cs > LaserAim
1  using UnityEngine;
2  using System.Collections;
0 references
3  public class LaserAim : MonoBehaviour
4  {
    2 references
5      public float lineWidth = 0.2f;
    3 references
6      public Color regularColor = new Color(0.15f, 0, 0, 1);
    1 reference
7      public Color firingColor = new Color(0.31f, 0, 0, 1);
    1 reference
8      public Material lineMat;
    3 references
9      private Vector3 lineEnd;
    2 references
10     private Projector proj;
    9 references
11     private LineRenderer line;
    0 references
12     void Start()
13     {
14         line = gameObject.AddComponent<LineRenderer>();
15         line.material = lineMat;
16         line.material.SetColor("_TintColor", regularColor);
17         line.SetVertexCount(2);
18         line.SetWidth(lineWidth, lineWidth);
19         proj = GetComponent<Projector>();
20     }
```



Atach script yang baru dibuat ke object LaserProjector.



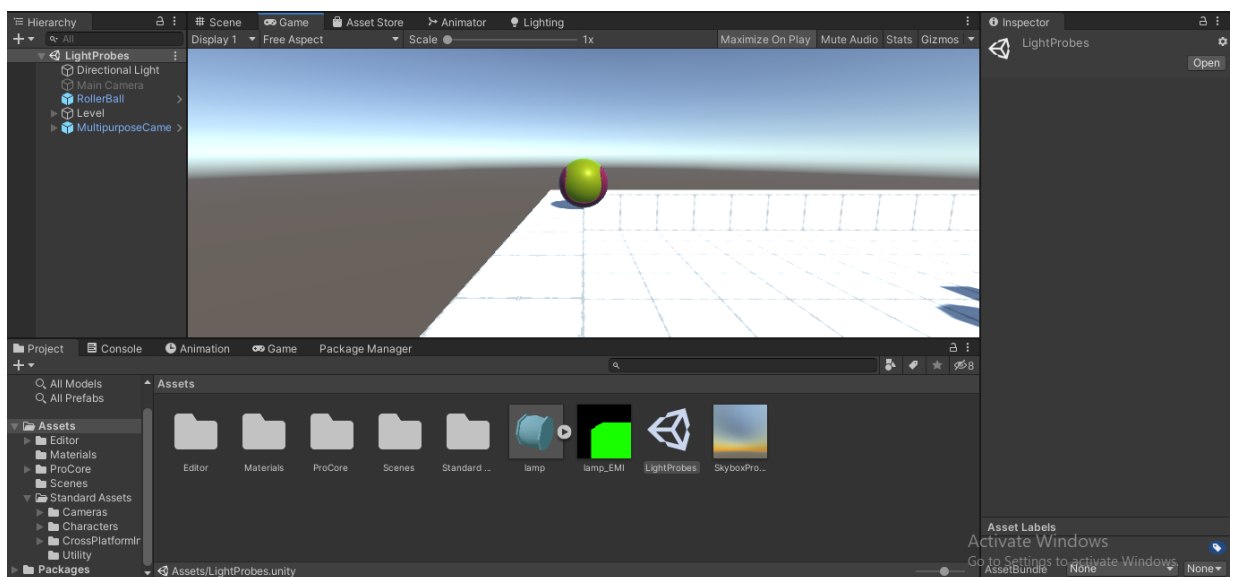
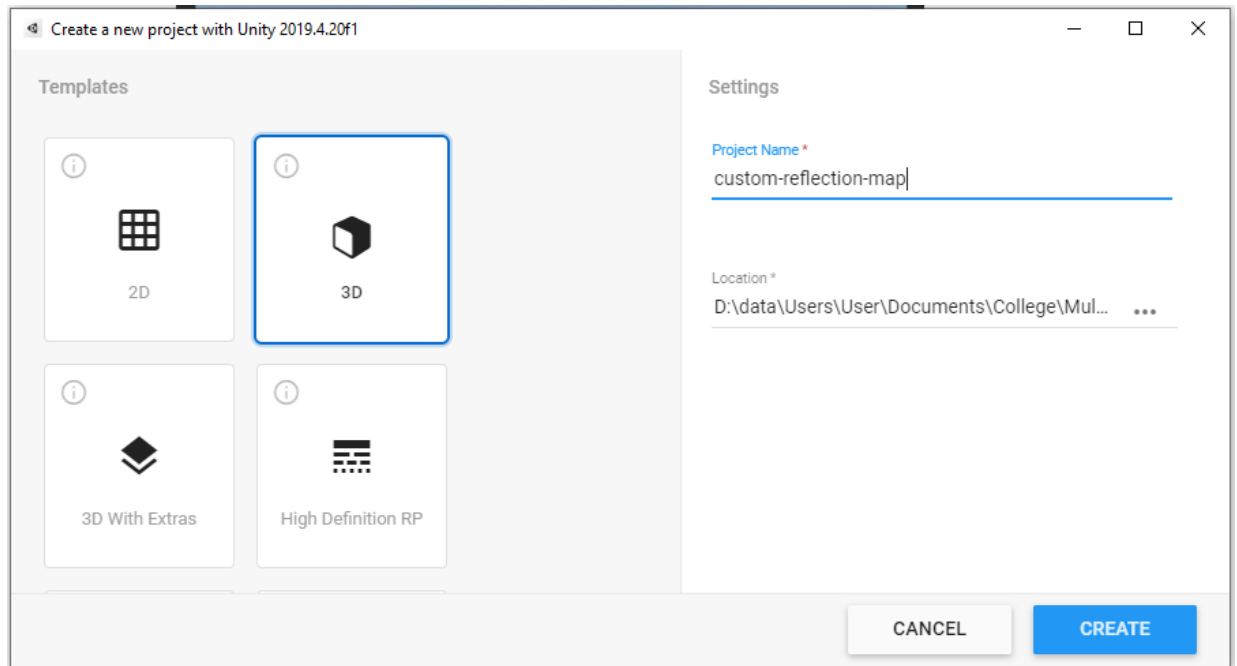
Pilih GameObject LaserProjector, dari Inspector view, pada komponen Laser Aim isikan Line_Mat pada slot Line Material. Jalankan scene, tekan Ctrl untuk menembak target dan perhatikan perbedaan laser beam saat menembak dan saat tidak menembak.



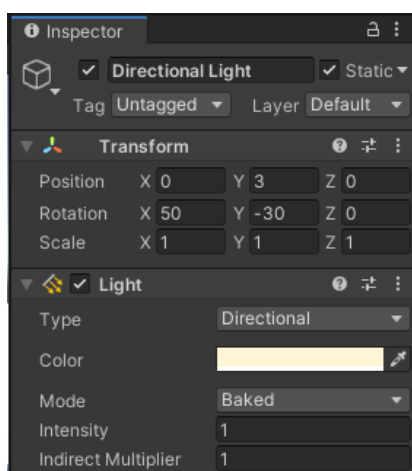
Untuk mengubah ketebalan dari beam (line), dapat diubah dari Line Width pada komponen Laser Aim menjadi lebih kecil, dan Orthographic size dari komponen Projector menjadi lebih kecil.

Lighting pada scene sederhana menggunakan Lightmaps dan Light Probes

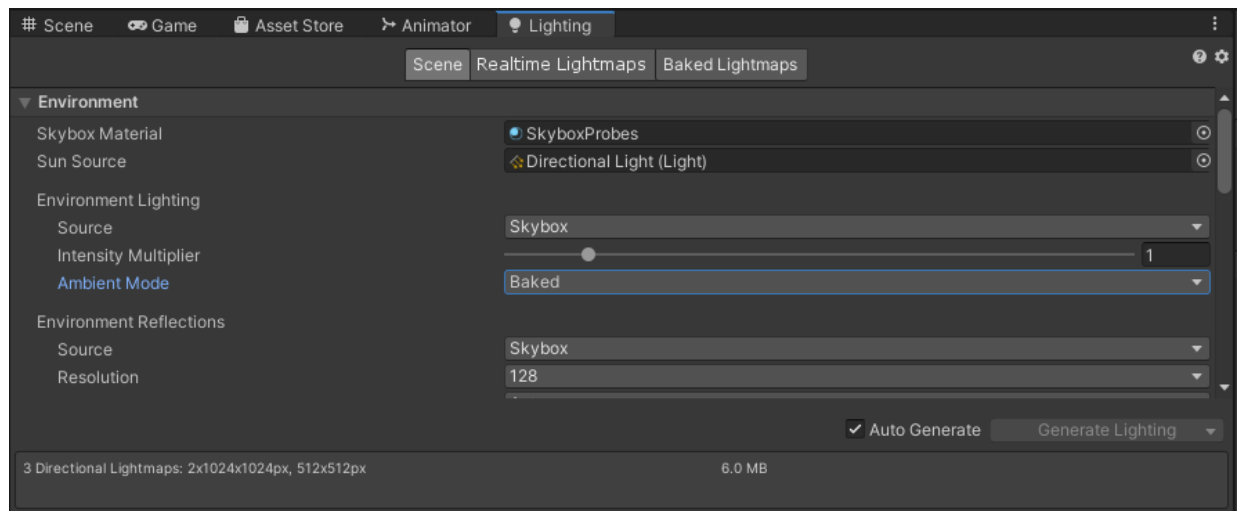
Import LightProbes.unitypackage kedalam project baru. Buka scene yang bernama LightProbes. Scene tersebut berisi basic environment dan game Rollerball sederhana.



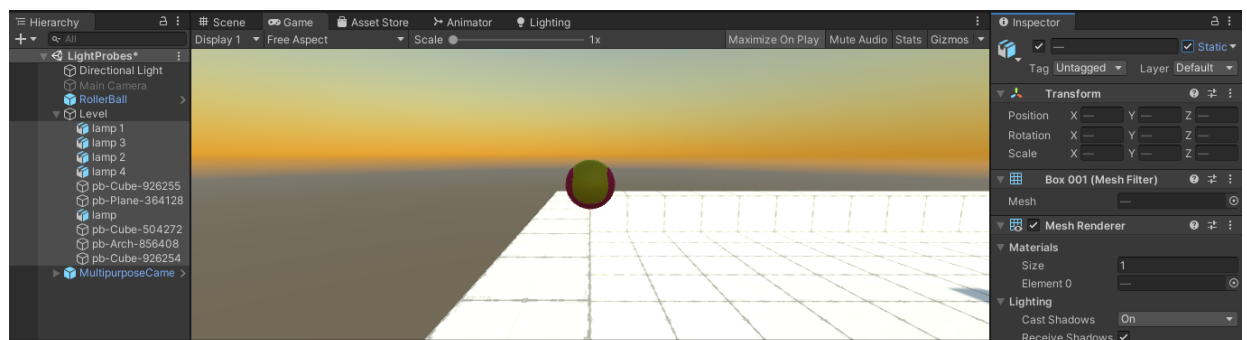
Untuk tahap awal, dimulai dengan set up lighting. Dari hierarchy view, pilih Directional Light. Kemudian, dari Inspector view, set Baking menjadi Baked. Masih di Inspector view, centang static pada bagian kanan atas view, seperti terlihat pada gambar berikut.



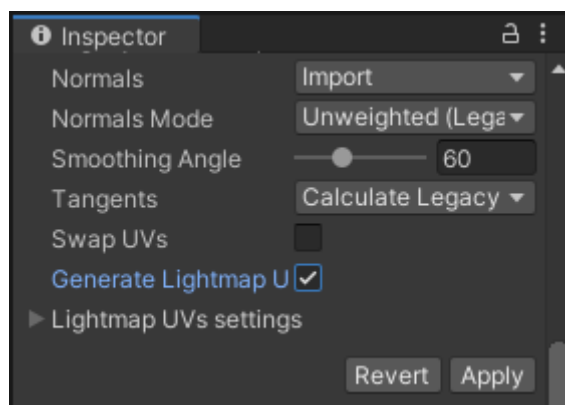
Langkah berikutnya dengan men-set Global Illumination untuk scene. Buka jendela Lighting (melalui Window | Lighting) dan pilih section scene. Kemudian dari subsection Environment Lighting, set SkyboxProbes (ada didalam Assets) di skybox, dan Directional Light di Sun. Selanjutnya, ubah opsi Ambient GI dari realtime menjadi Baked, seperti terlihat pada gambar berikut:



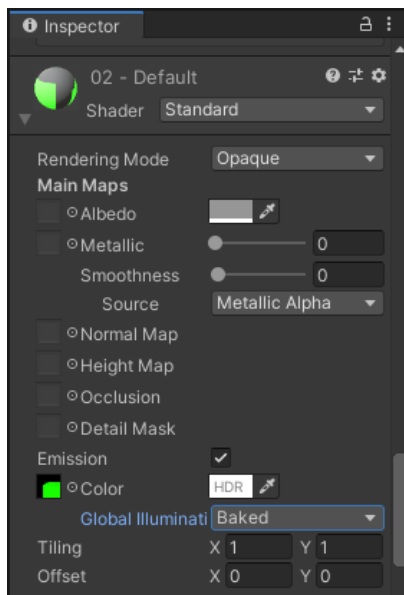
Lightmaps hanya dapat digunakan pada object statis saja. Dari Hierarchy view, expand game object Level untuk menampilkan daftar object-object childrenya. Kemudian pilih semua children dan set ssebagai static pada inspector view kanan atas.



3D Mesh yang diimport harus memiliki fitur Lightmap UV Coordinates. Dari Project view, cari dan pilih mesh lamp. Kemudian dari Inspector view, didalam seksi Model dari Import Setting, periksa opsi Generate Lightmaps UV, klik tombol Apply untuk menyimpan perubahan.

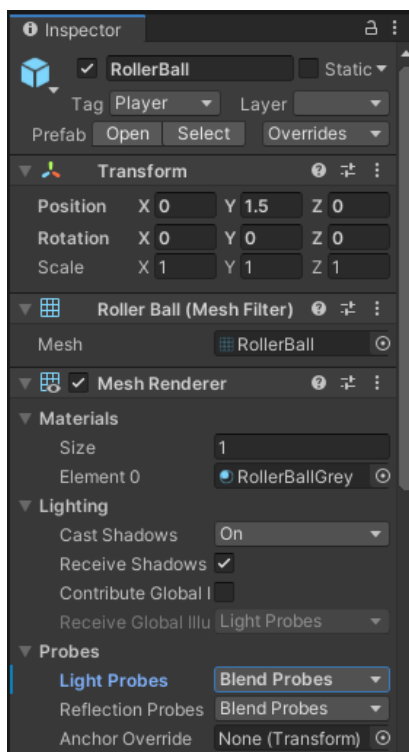


Pada bagian bawah (Shader), Ubah Emission menjadi lamp_EMI, dan Global Illuminationnya menjadi dari realtime menjadi Baked. Perhatikan perubahan yang terjadi dari lamp pada scene.

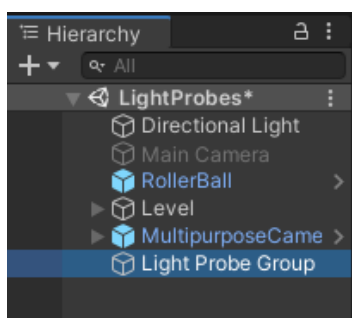


Buka Jendela Lighting. Secara default, Opsi Continues Baking selalu dicentang, hilangkan centangnya agar Lightmaps dapat di-bake sesuai kebutuhan. Klik Build dan tunggu Lightmaps sampai selesai di generate.

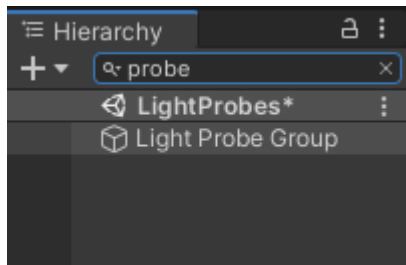
Dari Hierarchy view, pilih RollerBall. Kemudian dari Inspector view, cari komponen Mesh Renderer, Pada Light Probes, pastikan Blend Probes sudah terpilih:



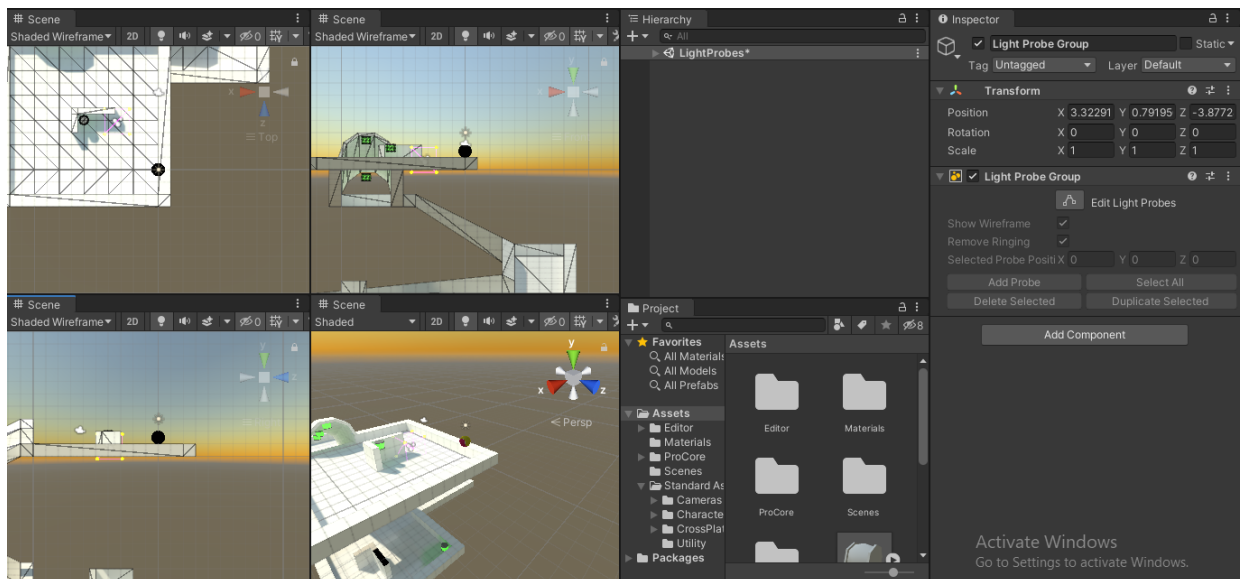
Selanjutnya akan dibuat Light Probes untuk scene. Dari Hierarchy view, klik menu Dropdown Create dan tambahkan Light Probe Group kedalam scene (Create | Light | Light Probe Group)



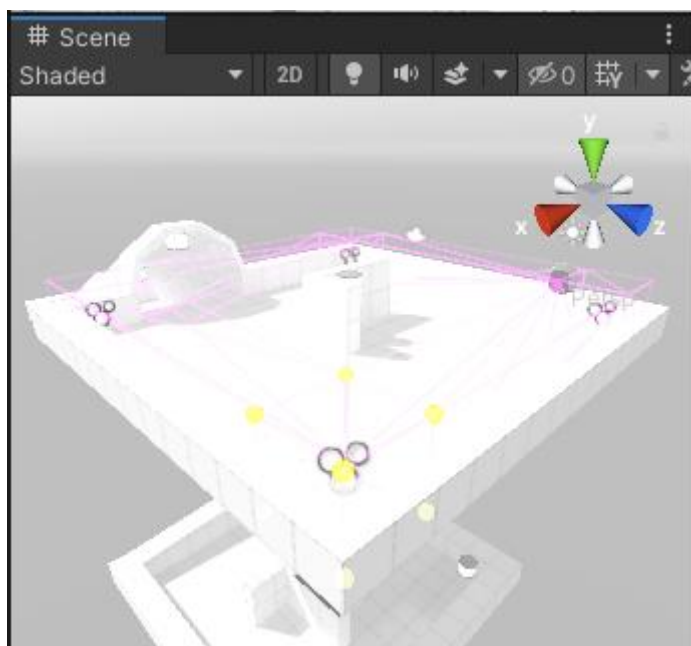
Untuk memanipulasi probes saja, ketikkan Probe kedalam field pencarian dari hierarchy view. Hal ini akan membuat Light Probe Group sebagai satu-satunya game object yang tampak dan dapat diedit di scene.



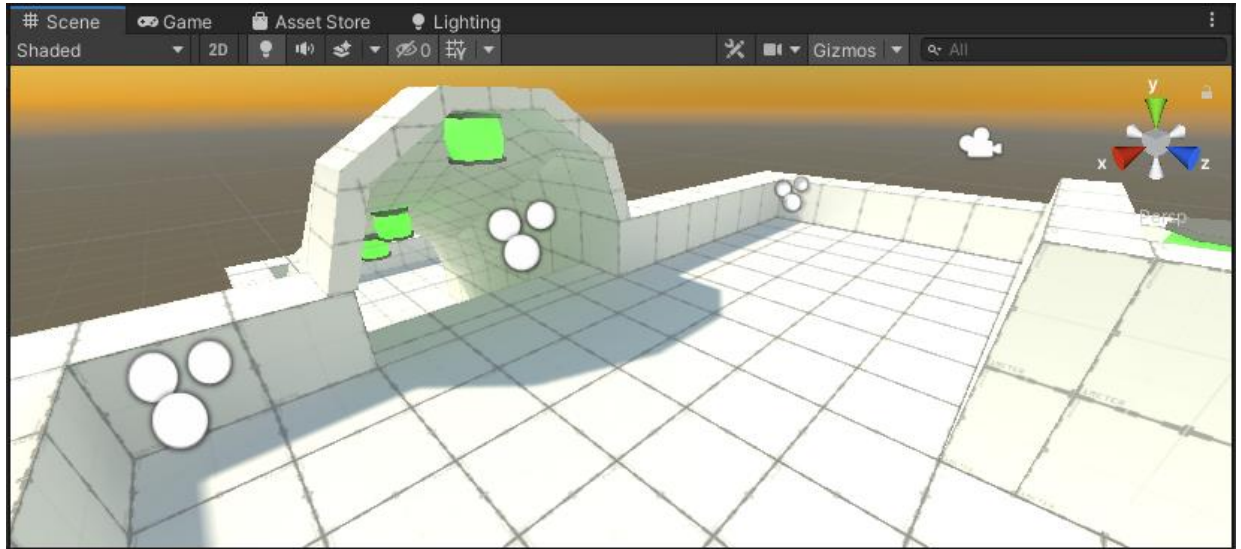
Ubah Layout viewport menjadi 4 bagian dengan menggunakan menu Window | Layouts | 4 Split. Set viewport sebagai Top, Front, Right, dan Persp. Pada bagian Top, Front, dan Right tampilannya diubah ke Wireframe. Hal ini dilakukan untuk mempermudah kita menempatkan Light Probes.



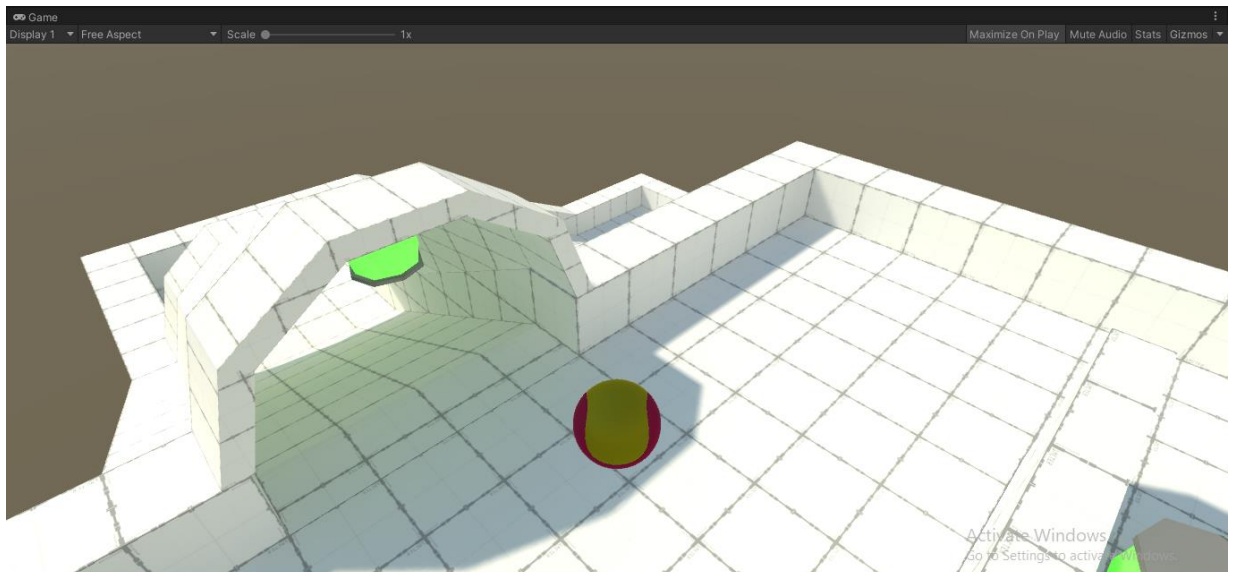
Tempatkan Light Probes yang sudah dibuat ke tiap sudut dari level top room. Dengan drag tiap Probe nya.



Pilih empat probes pada sisi kiri dari tunnel / terowongan. Duplikat keempat probe tersebut Dengan Ctrl + D, lalu geser probenya sedikit kesebelah kanan dari probe asli, setidaknya hingga tidak terkena shadow dari wall / dinding.



Test scenenya. Rollerball akan dikenai cahaya sesuai dengan Light Probes yang telah dibuat.



Link Github : https://github.com/hunaynr/komputasi_multimedia/tree/main/10thWeek

Link Youtube : <https://youtu.be/KvPLPjM2Lol>

