

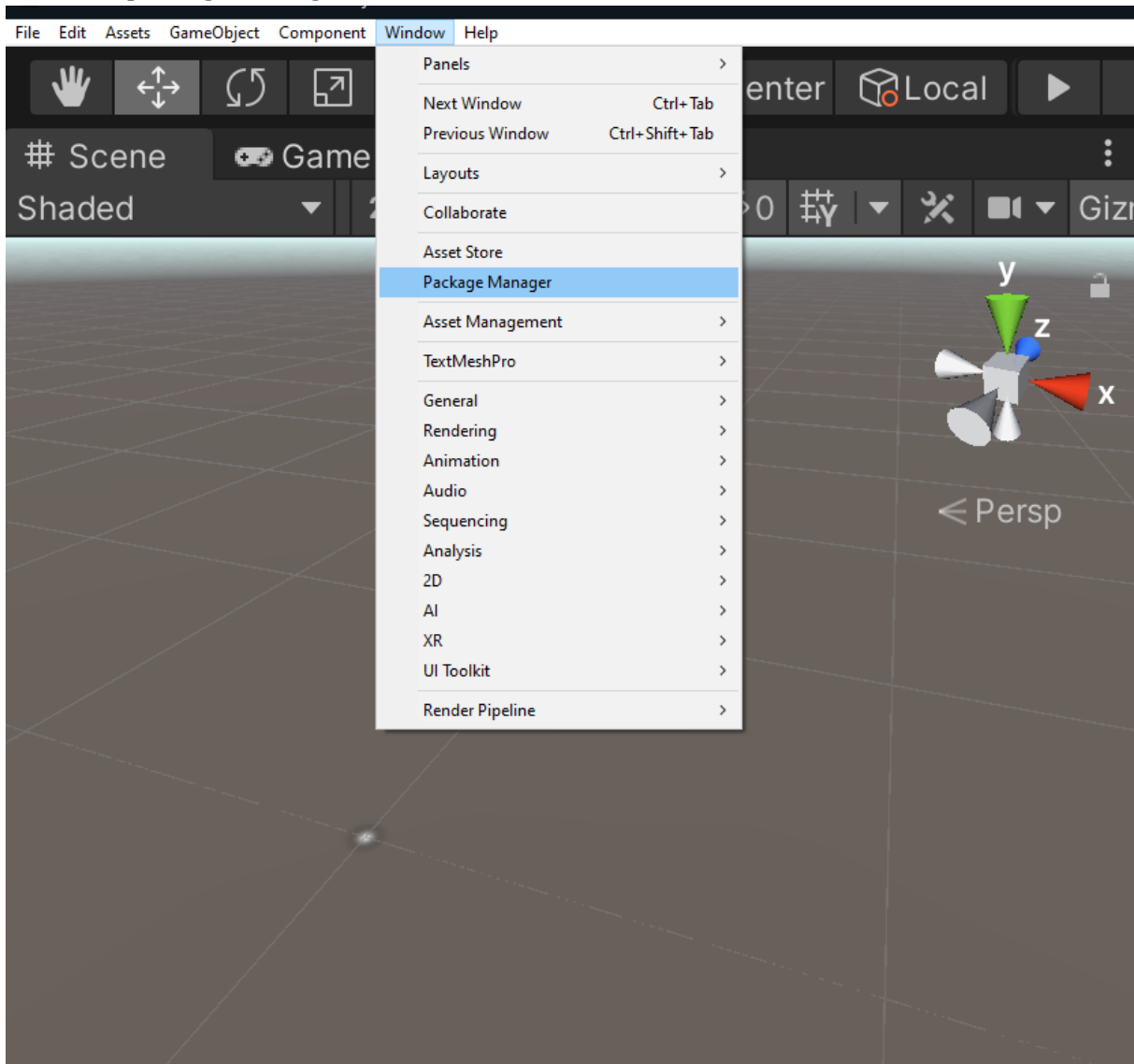
DAY 2 AR Foundation

PART 1 – SETUP

Create a new project with version 2020.1

Include android or ios build support for the version

window>package manager



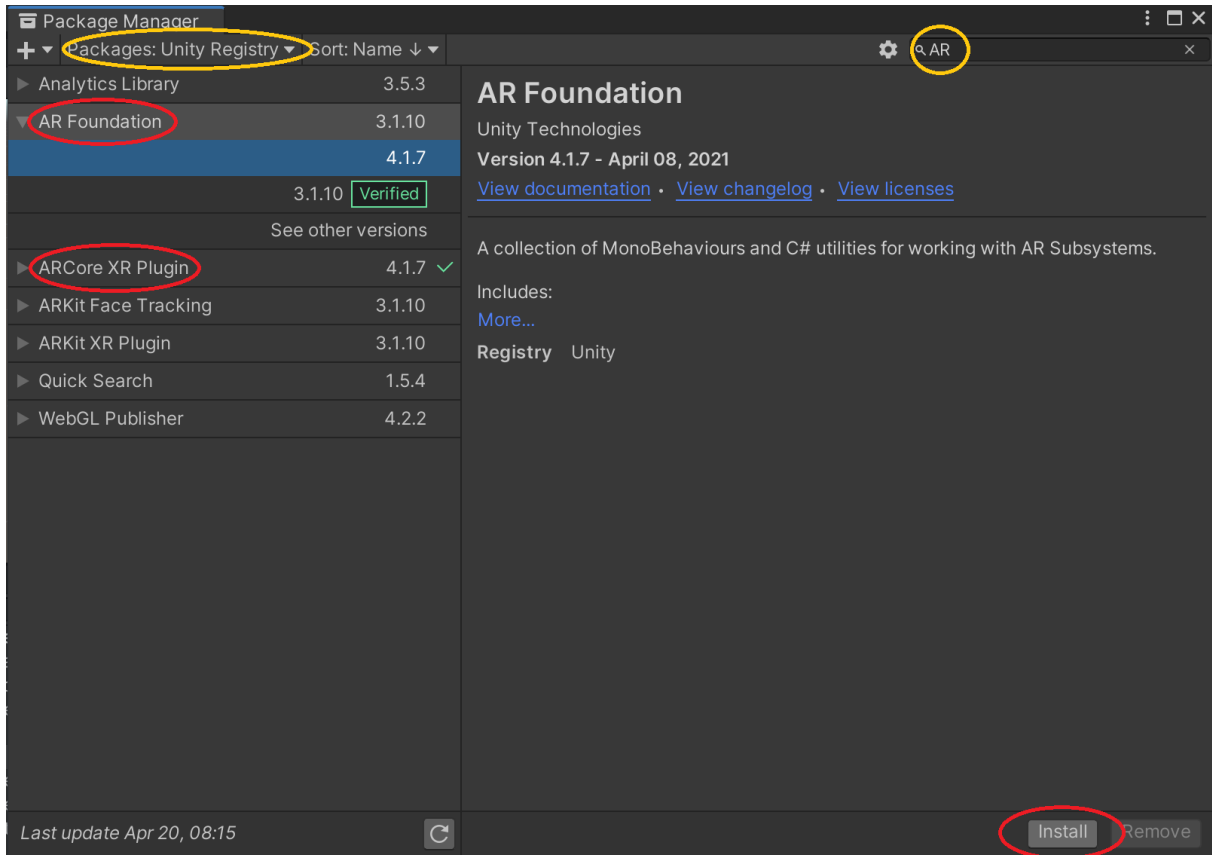
Select Packages: Unity Registry and install the following packages:

AR foundation 4.1.7

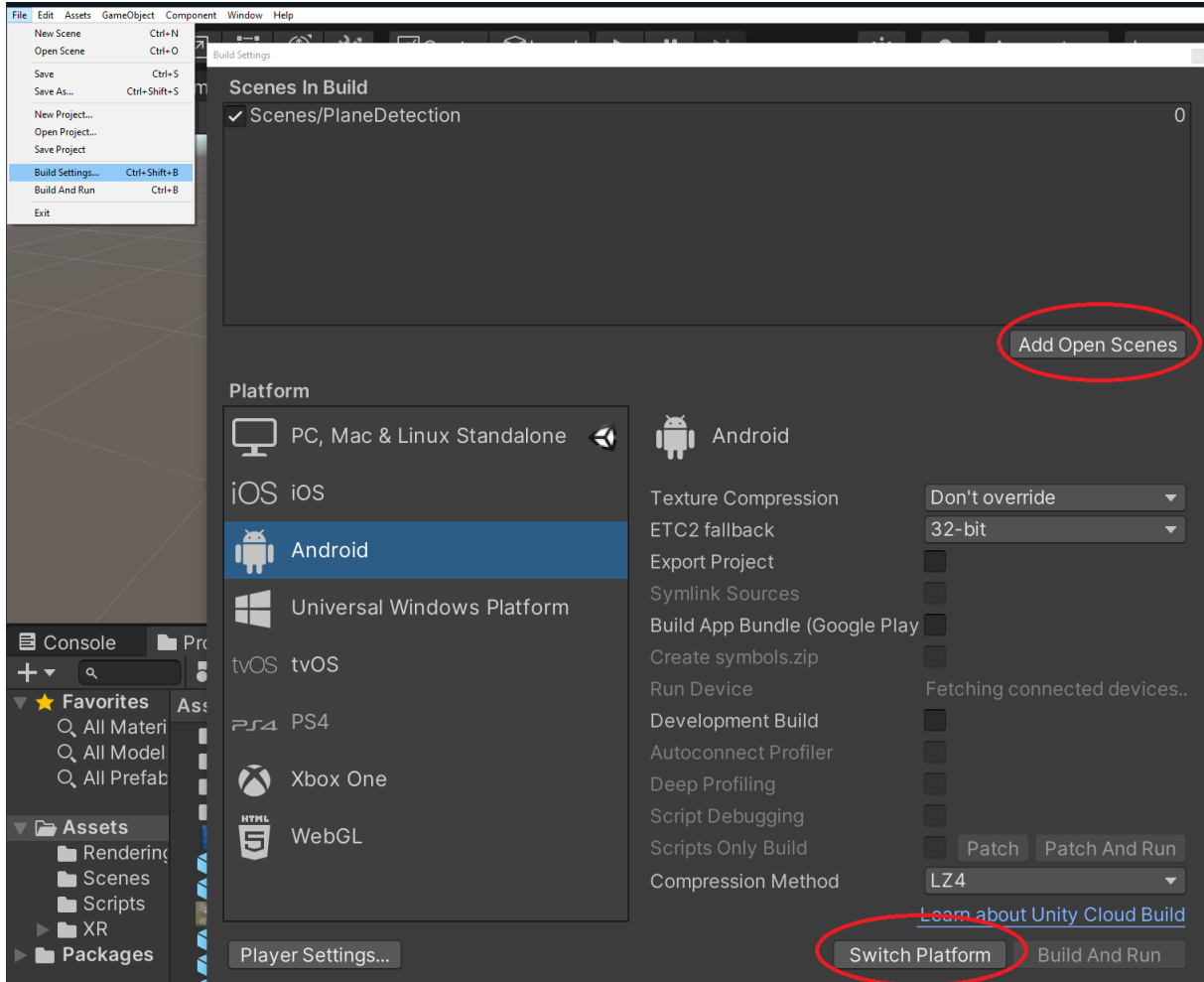
ARCore 4.1.7 (Android)

ARKit XR Plugin 4.1.7 (IOS)

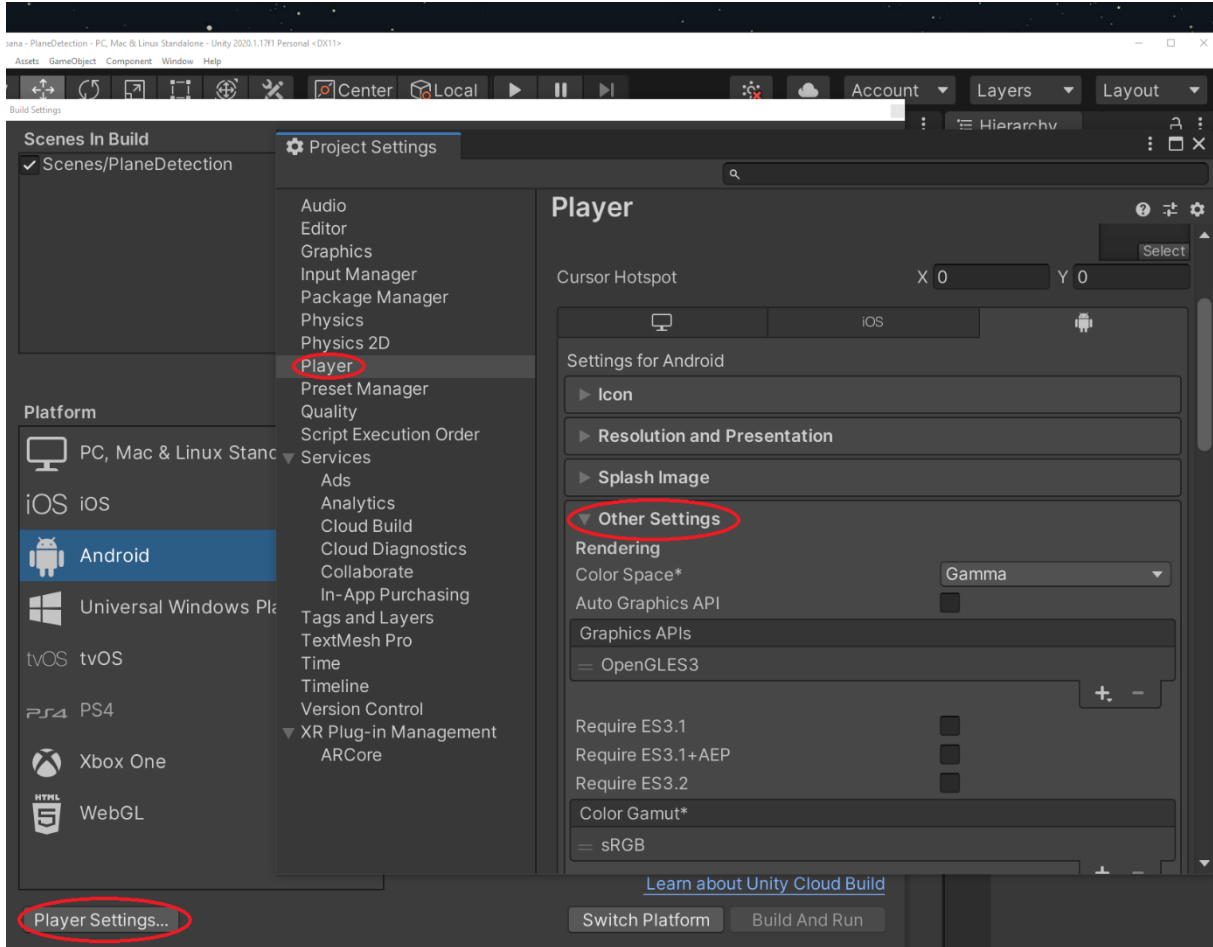
If it doesn't show newer versions, select the „See other versions“ option



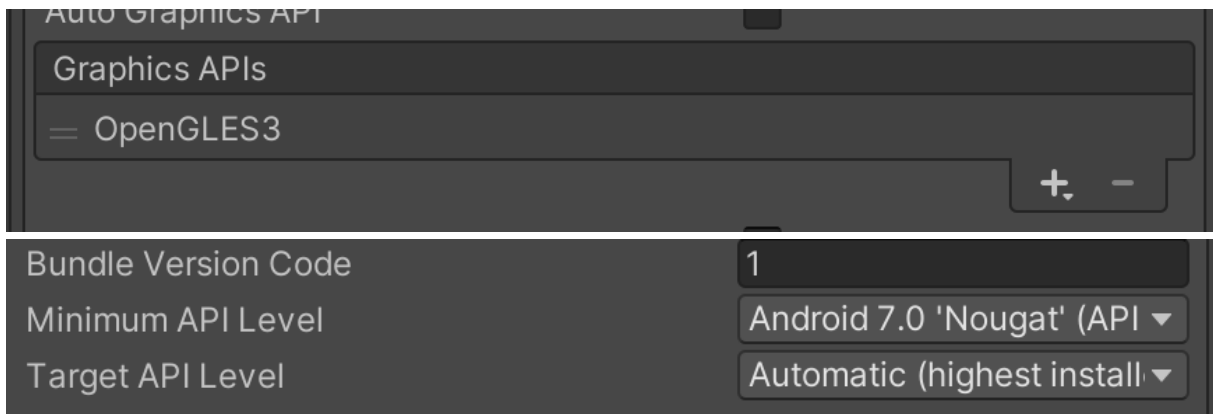
- Open up the build settings
- Change the platform to android
- Add the current scene to open scenes



Under Player settings, go to Other settings



Remove Vulkan and set the minimum API level to Android 7.0

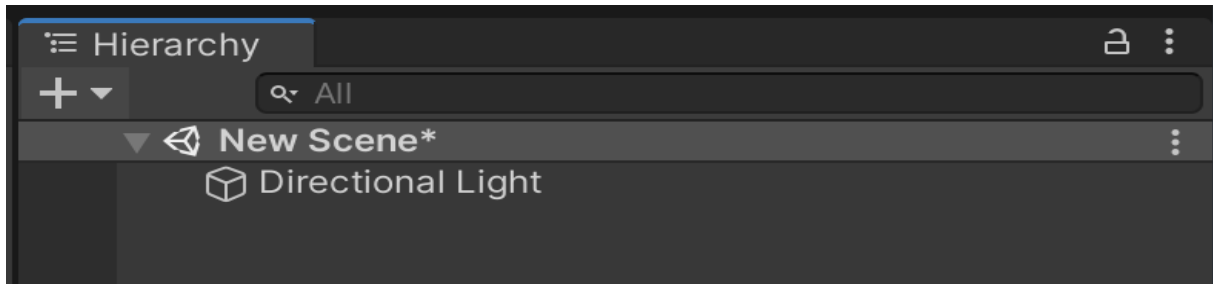
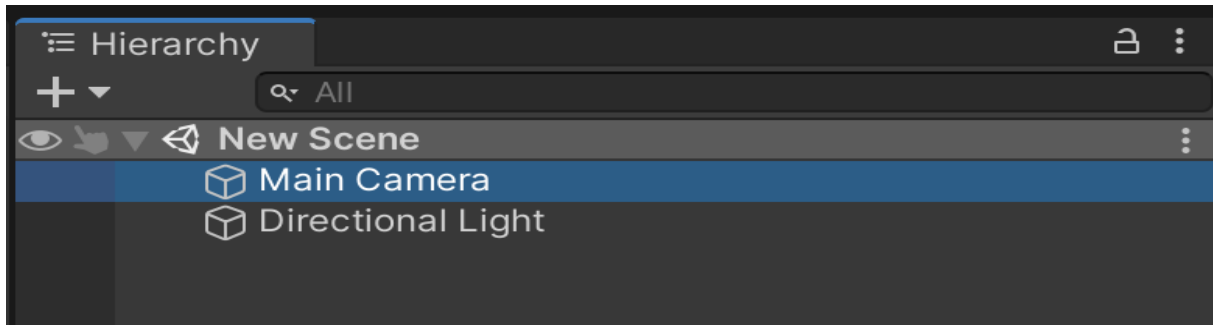


XR Plug-in Management>check ARCore (NOT UNDER PLAYER SETTINGS)

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

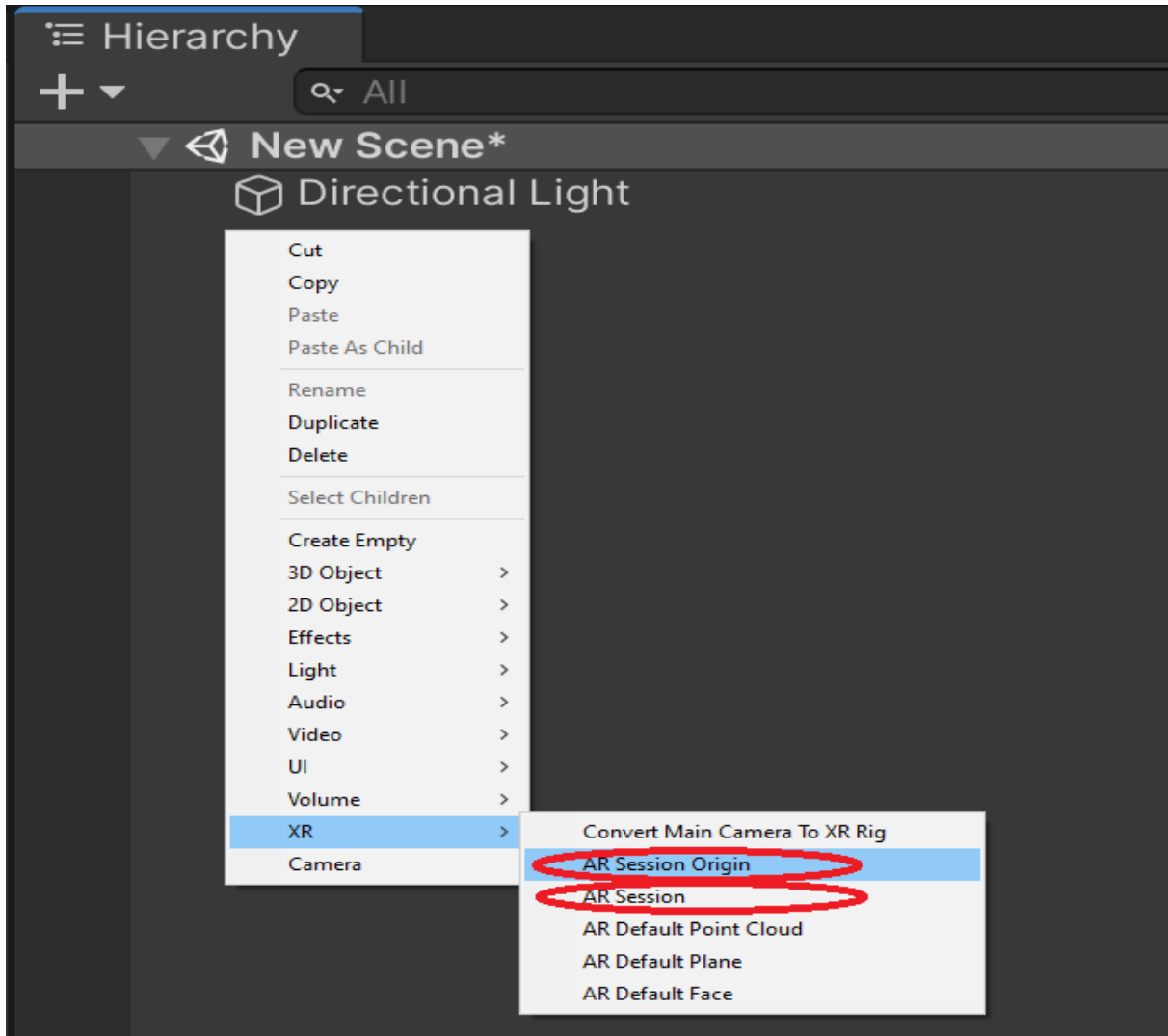
Bundle Version Code	1
Minimum API Level	Android 7.0 'Nougat' (API ▼)
Target API Level	Automatic (highest install) ▼

In the main scene remove the camera

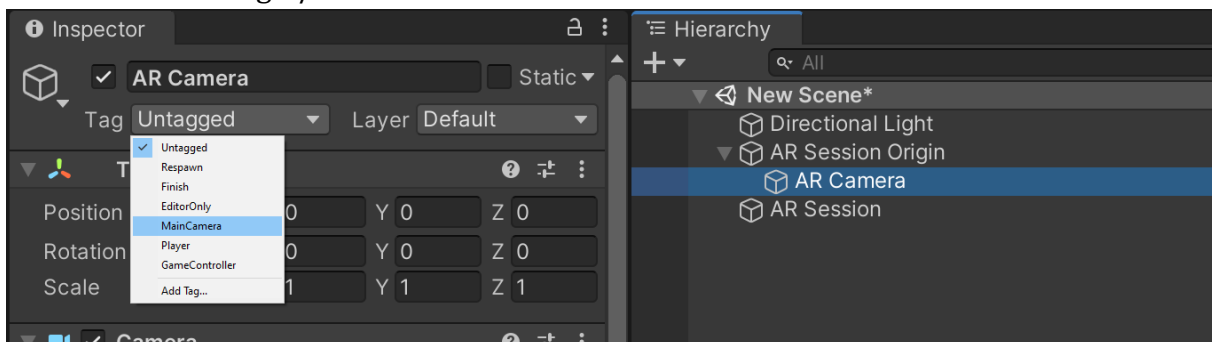


add>XR>AR Session Origin (that adds the camera)

Also, add AR Session

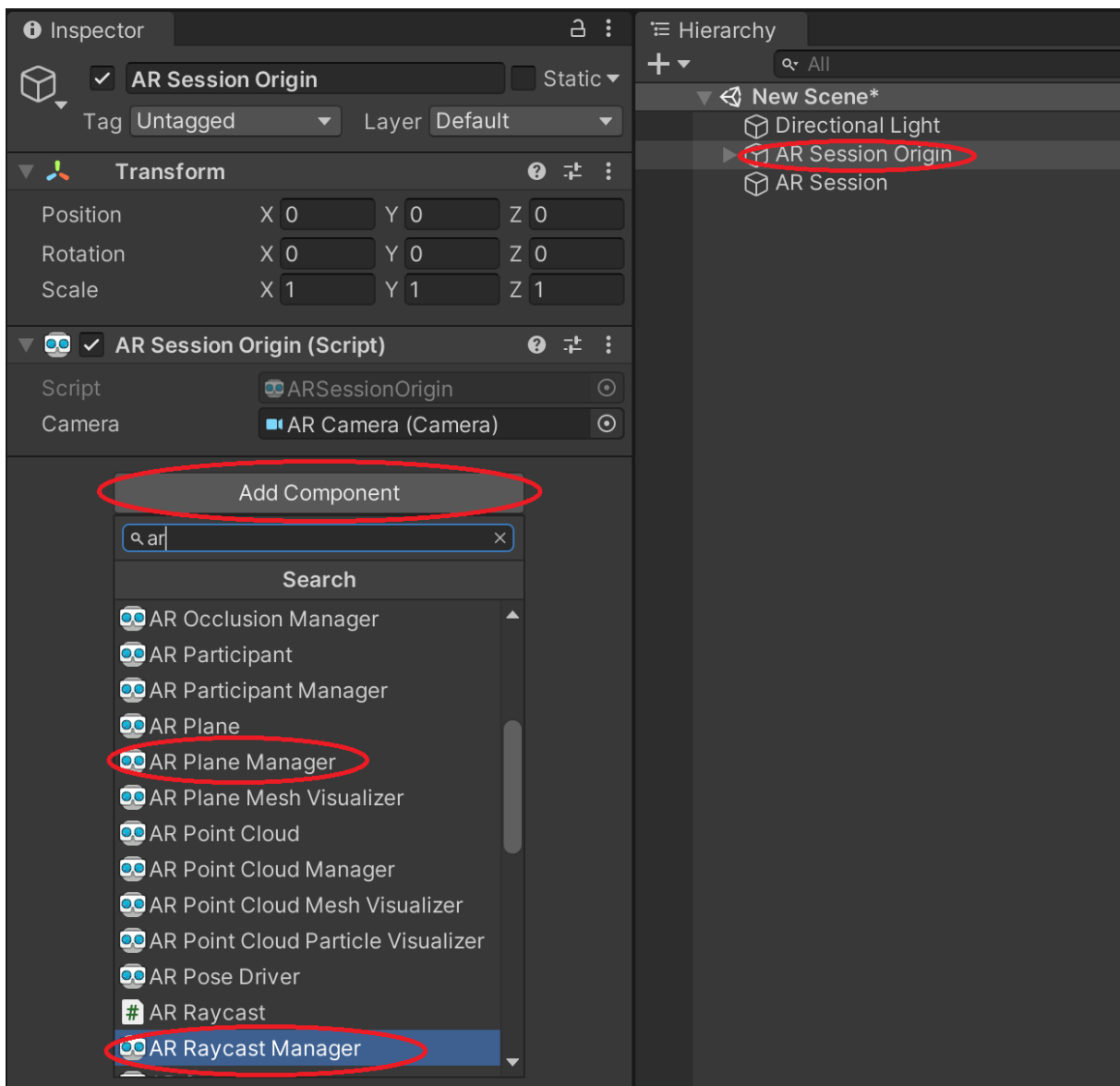


Set AR Session Origin/AR Camera to the main camera

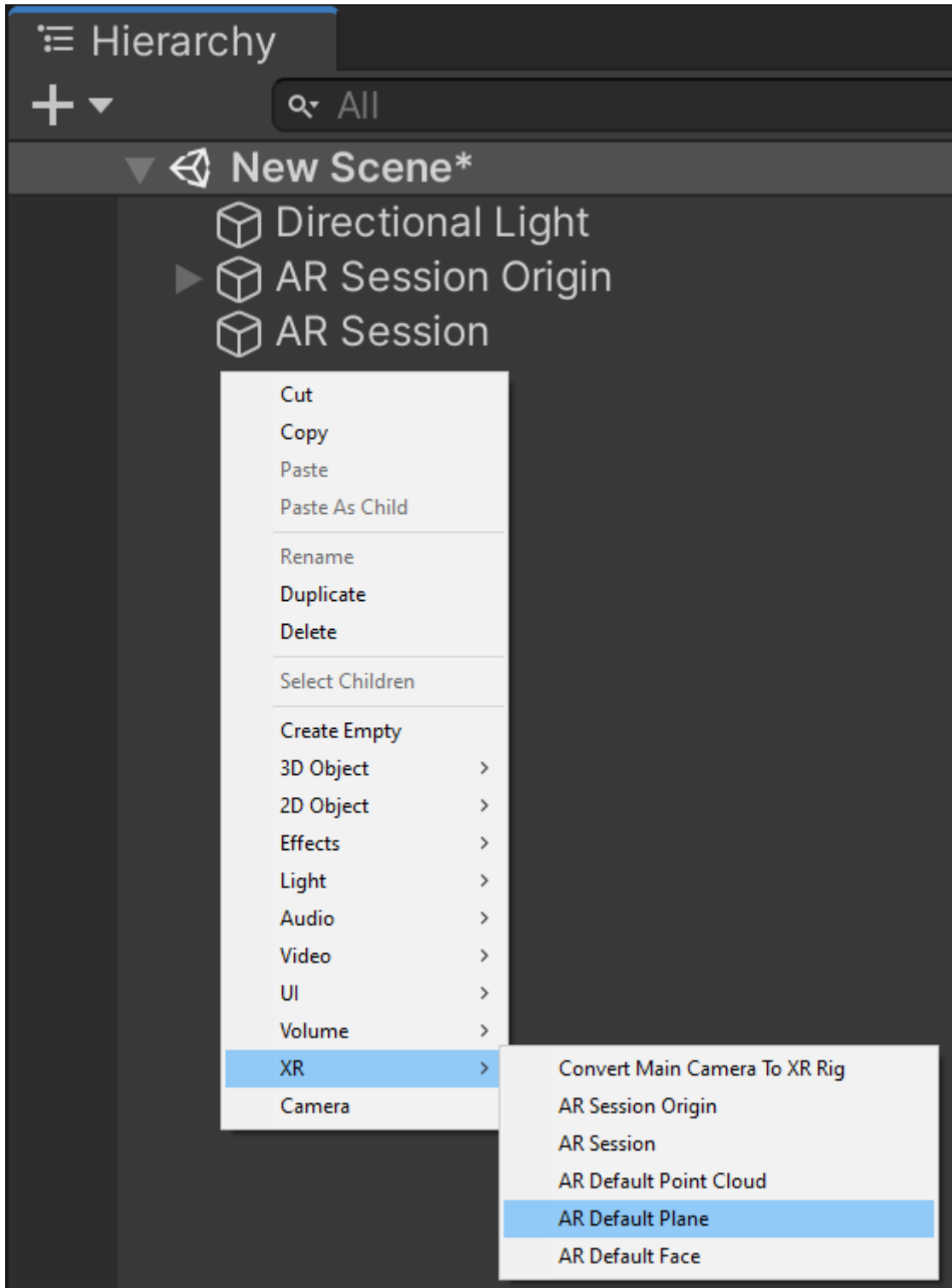


- Select AR session origin
- Click Add Component
- Add ar raycast manager
- Add a plane manager

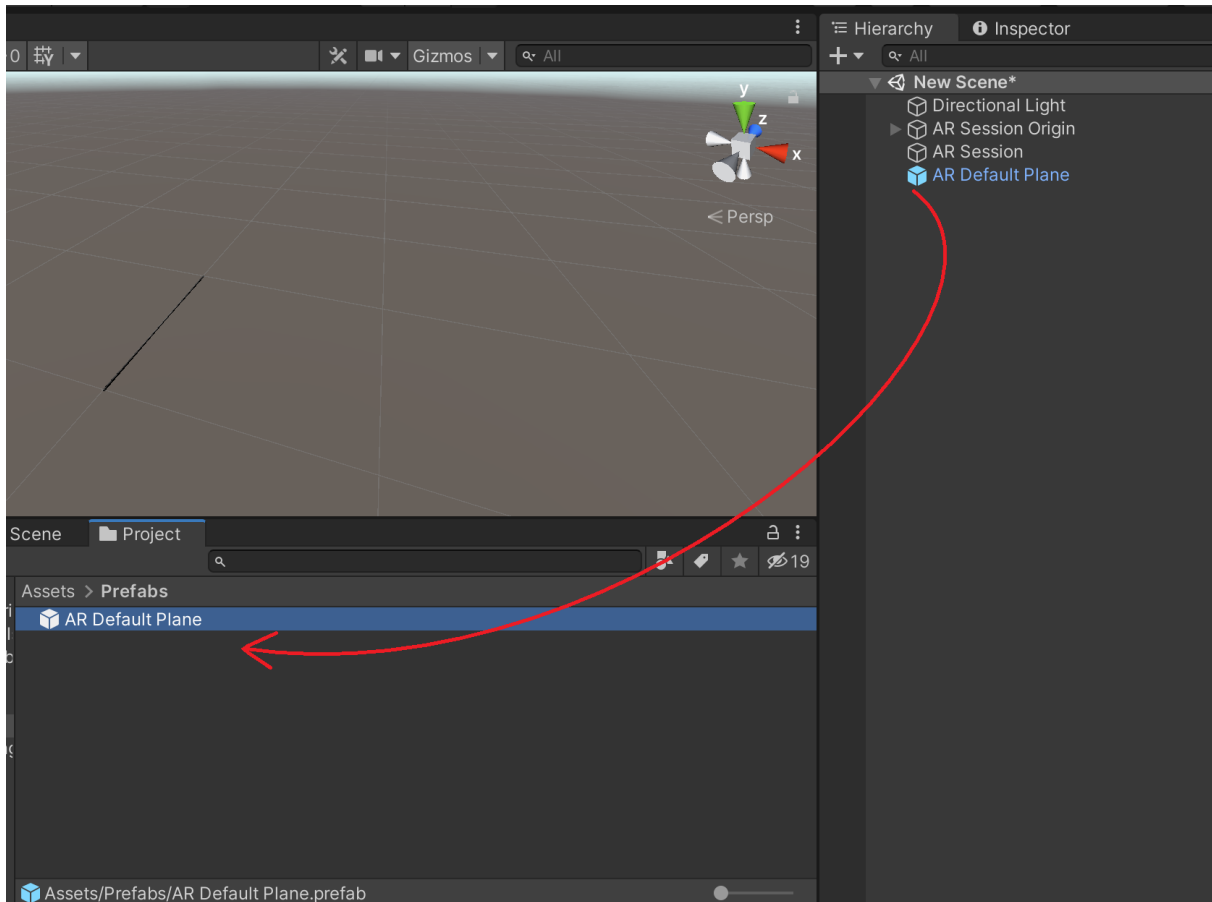
The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.



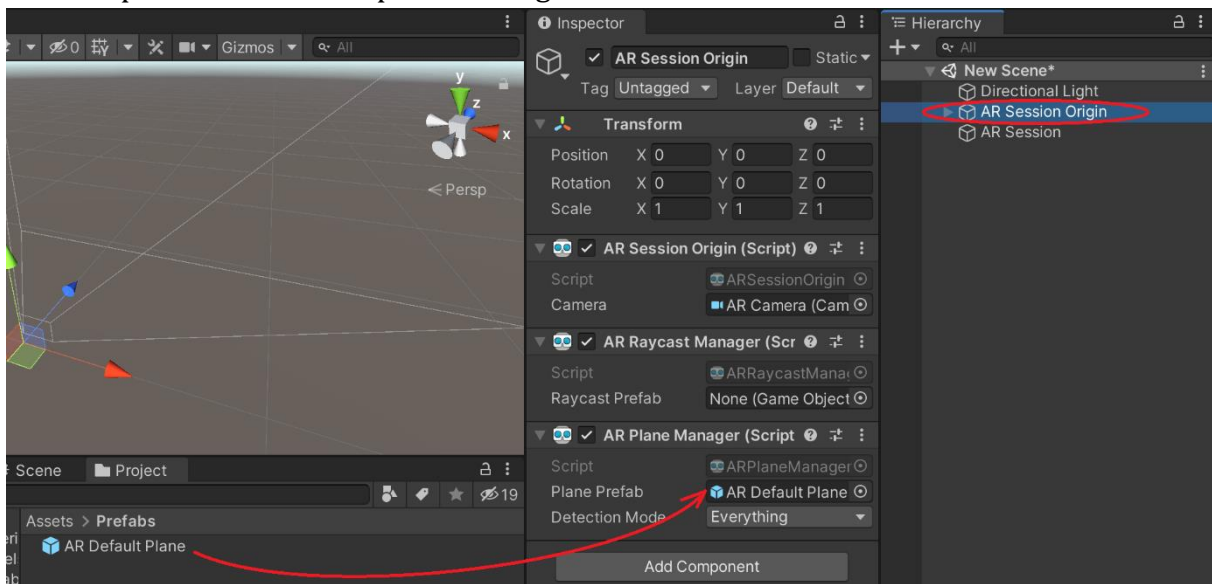
Greate AR default plane



Click and drag the AR Default Plane into the Project window
Then delete the AR Default Plane from the Scene

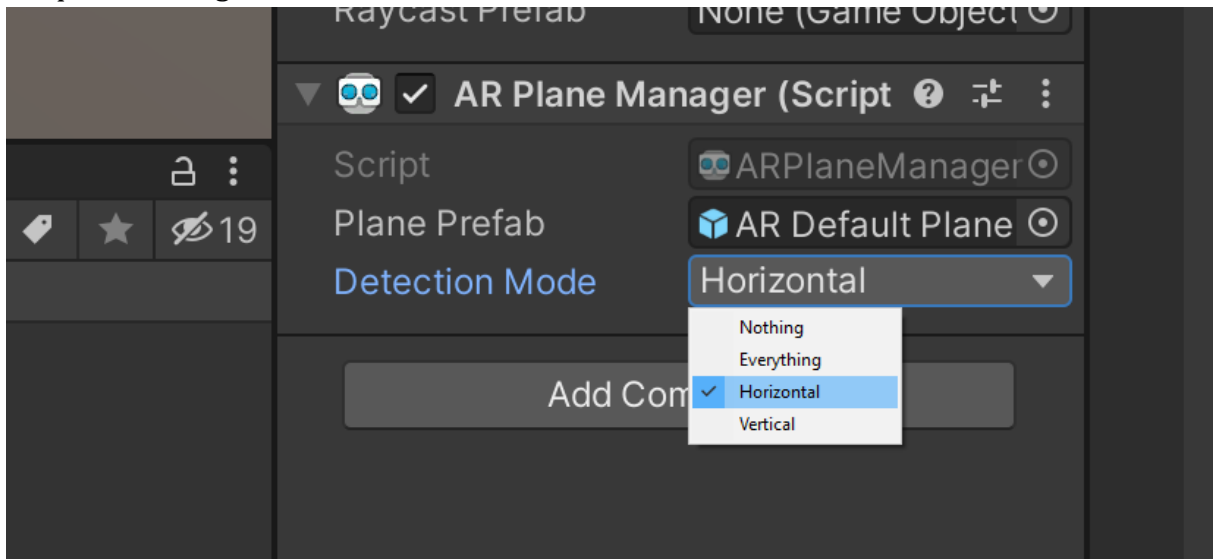


Add the prefab into the AR plane manager



The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Set plane manager to horizontal

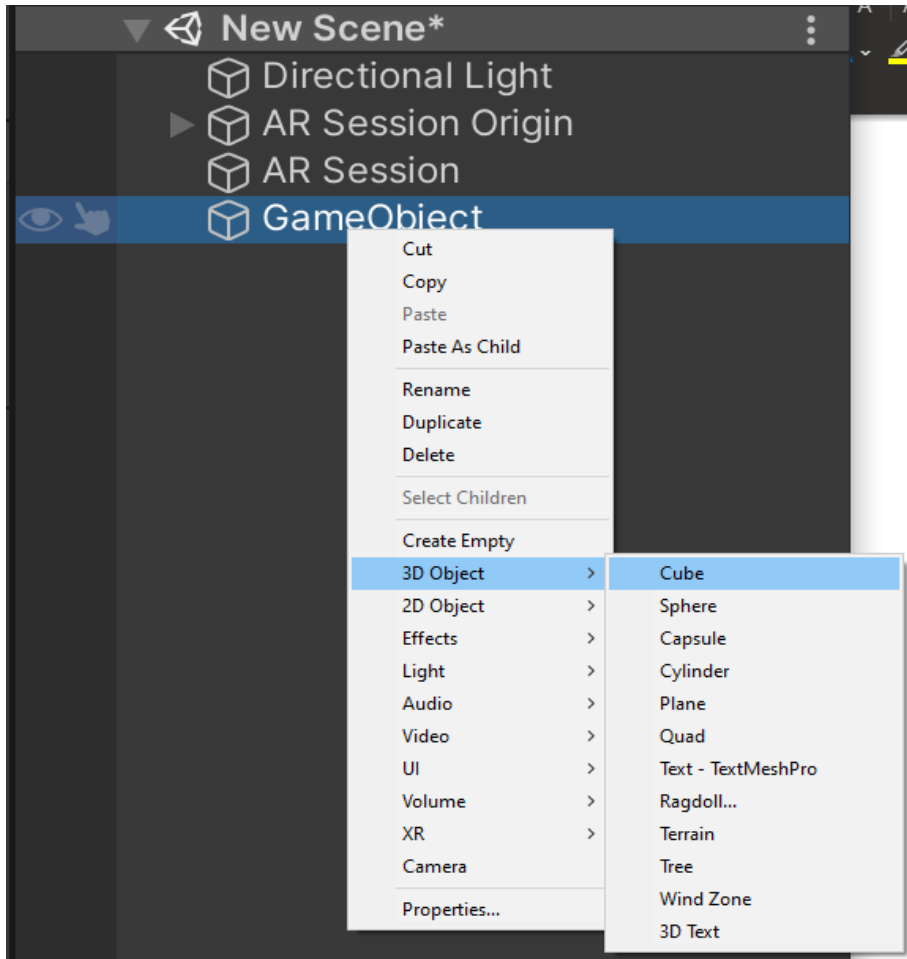


BUILD AND RUN THE CODE

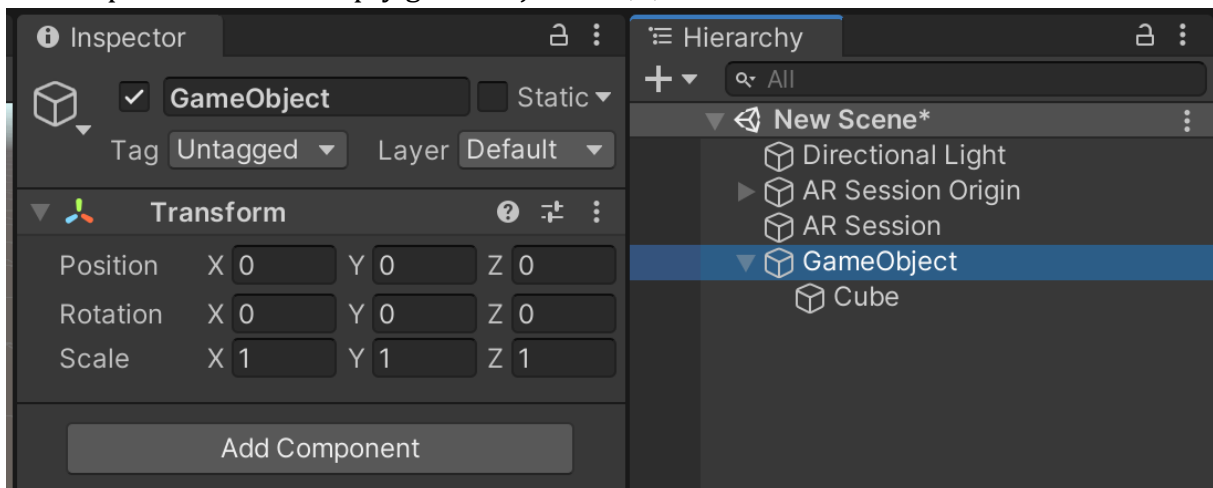
PART 2 – INTERACTION IN AR

Create an Empty object

Select it and create a Cube object inside the empty „GameObject“

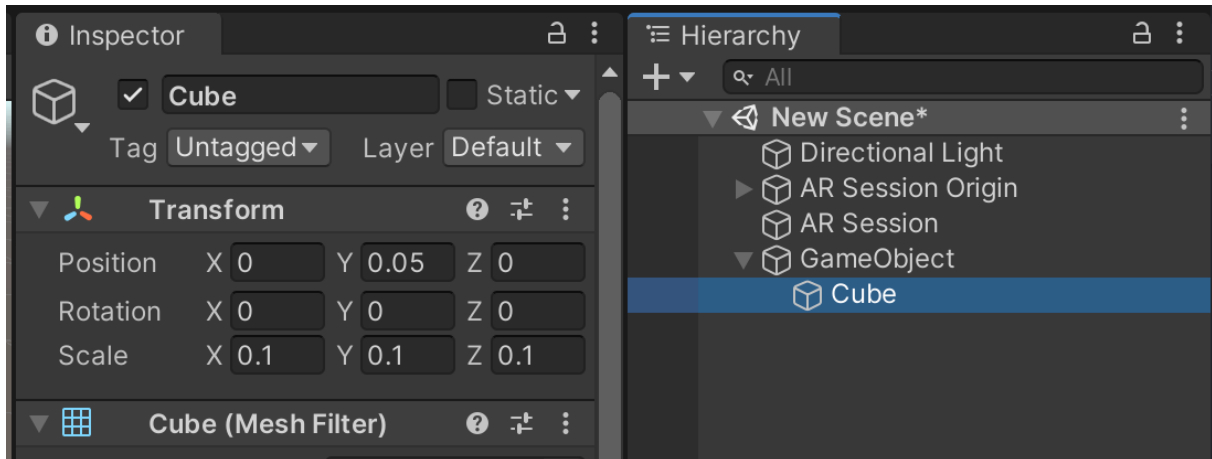


Set the position of the empty game object to 0,0,0

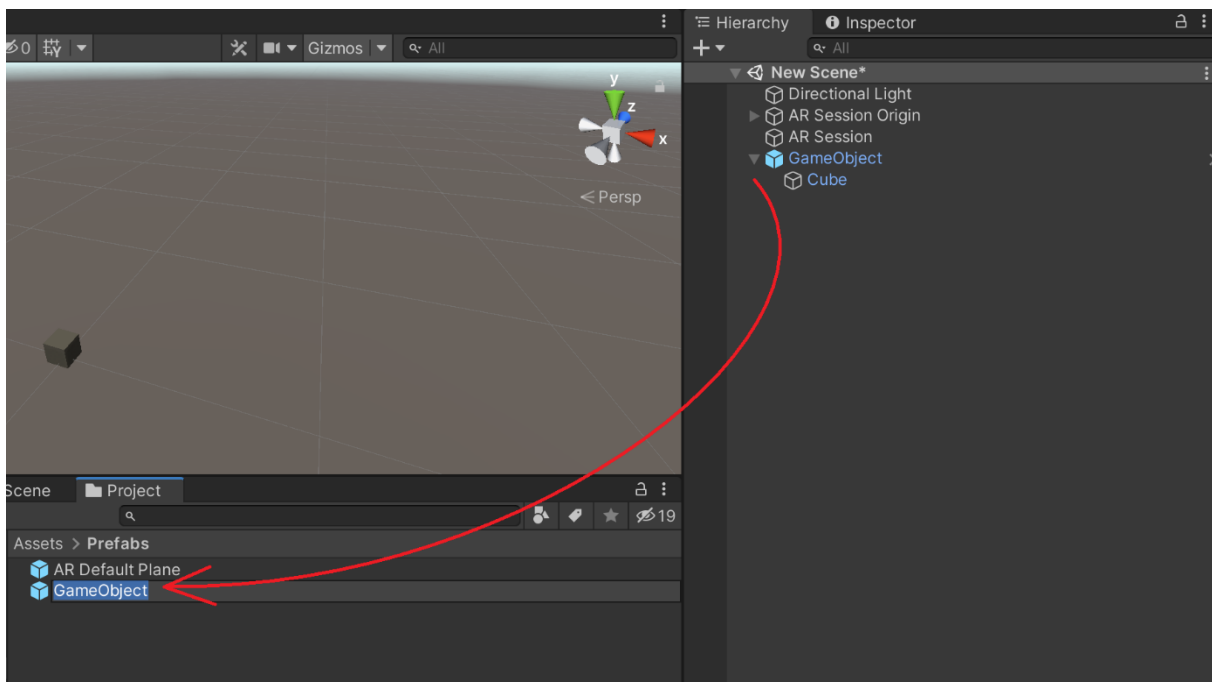


The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

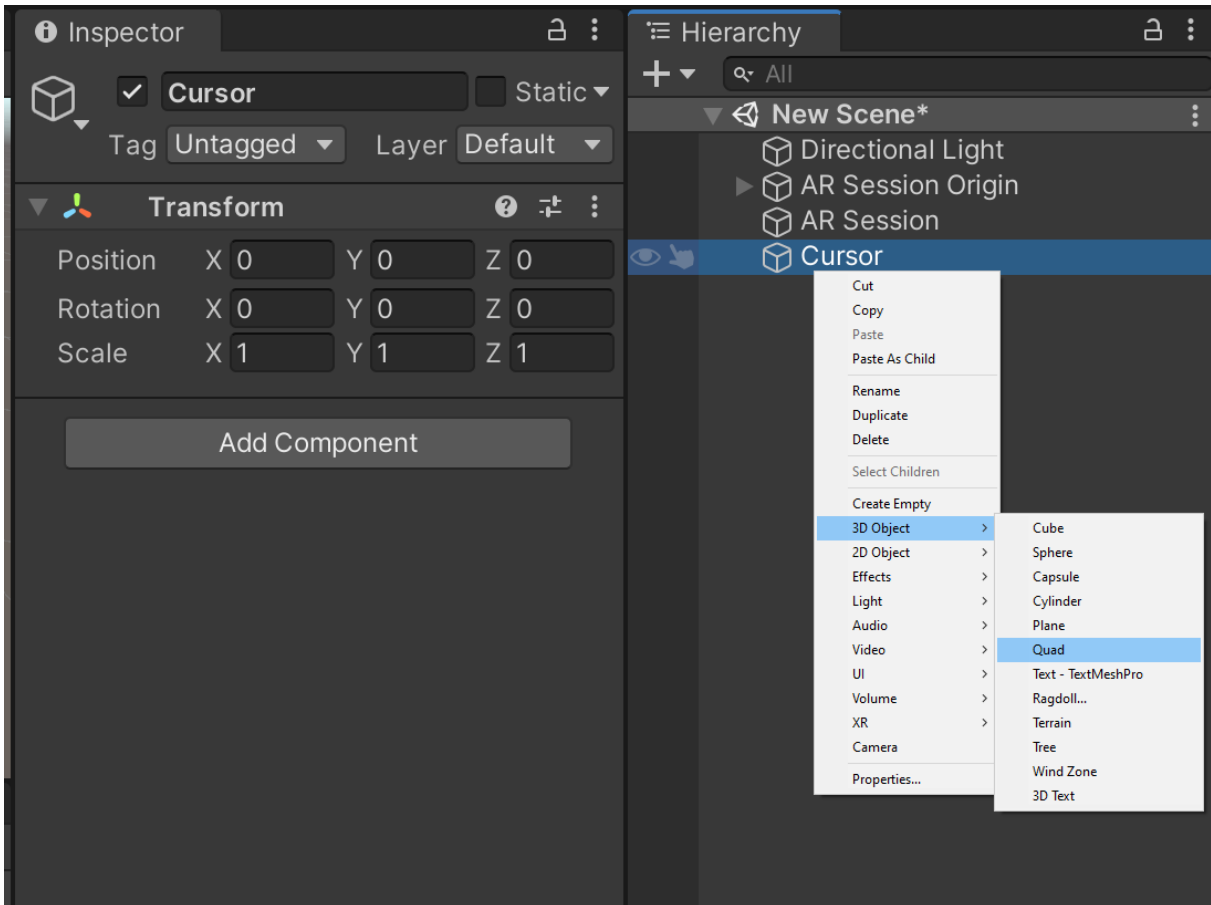
Scale the cube to 0.1 and set the y position to 0.05



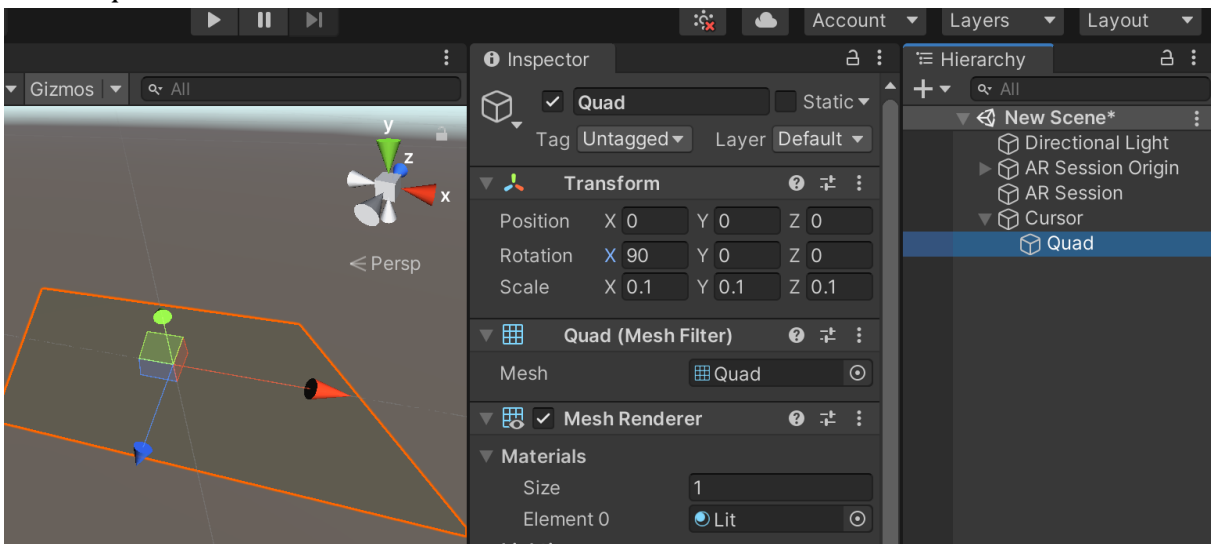
Save it as a prefab just like with the AR Default Plane
Then delete it from the Scene



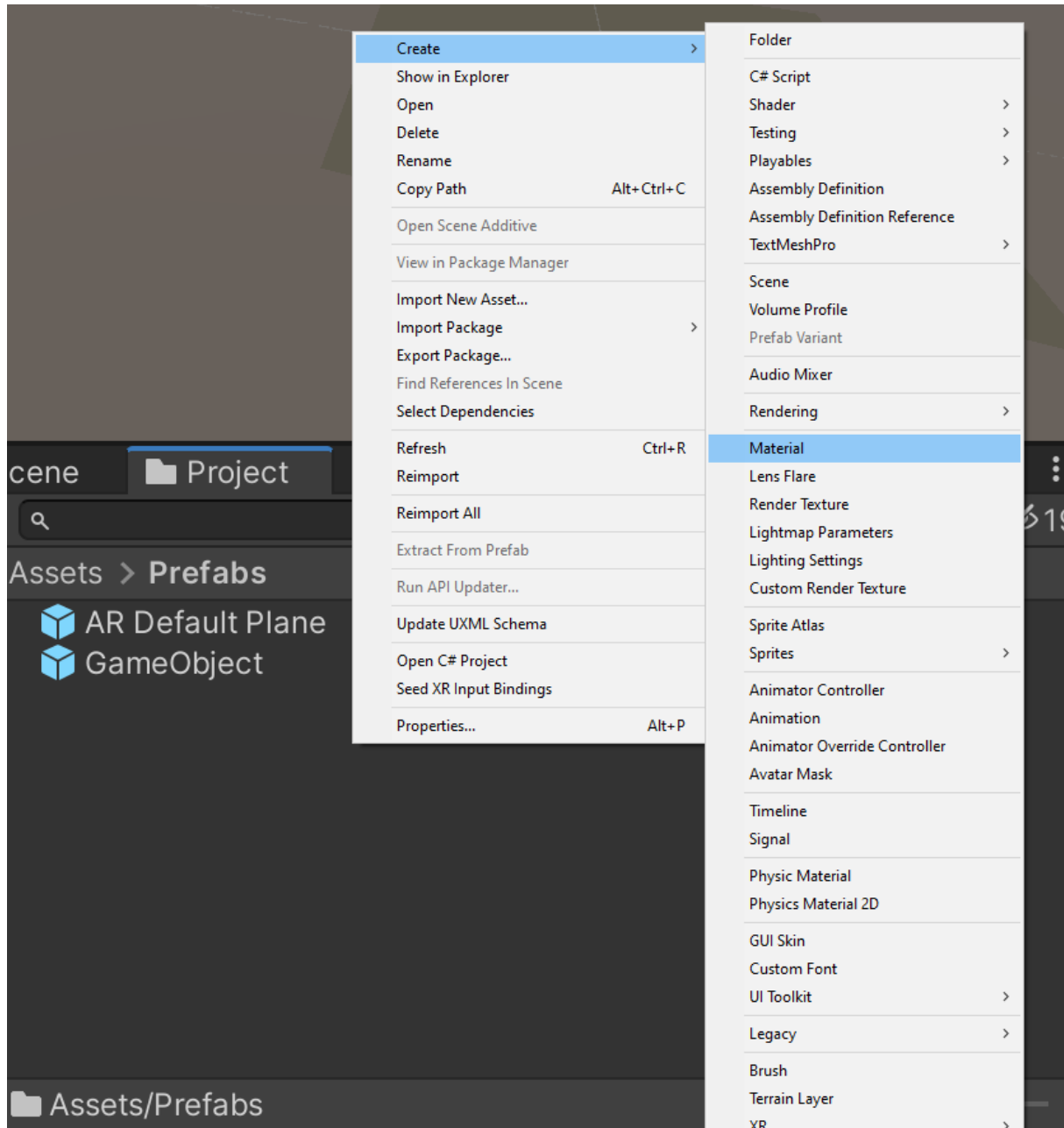
Create an empty gameobject and name it Cursor and put a Quad in it



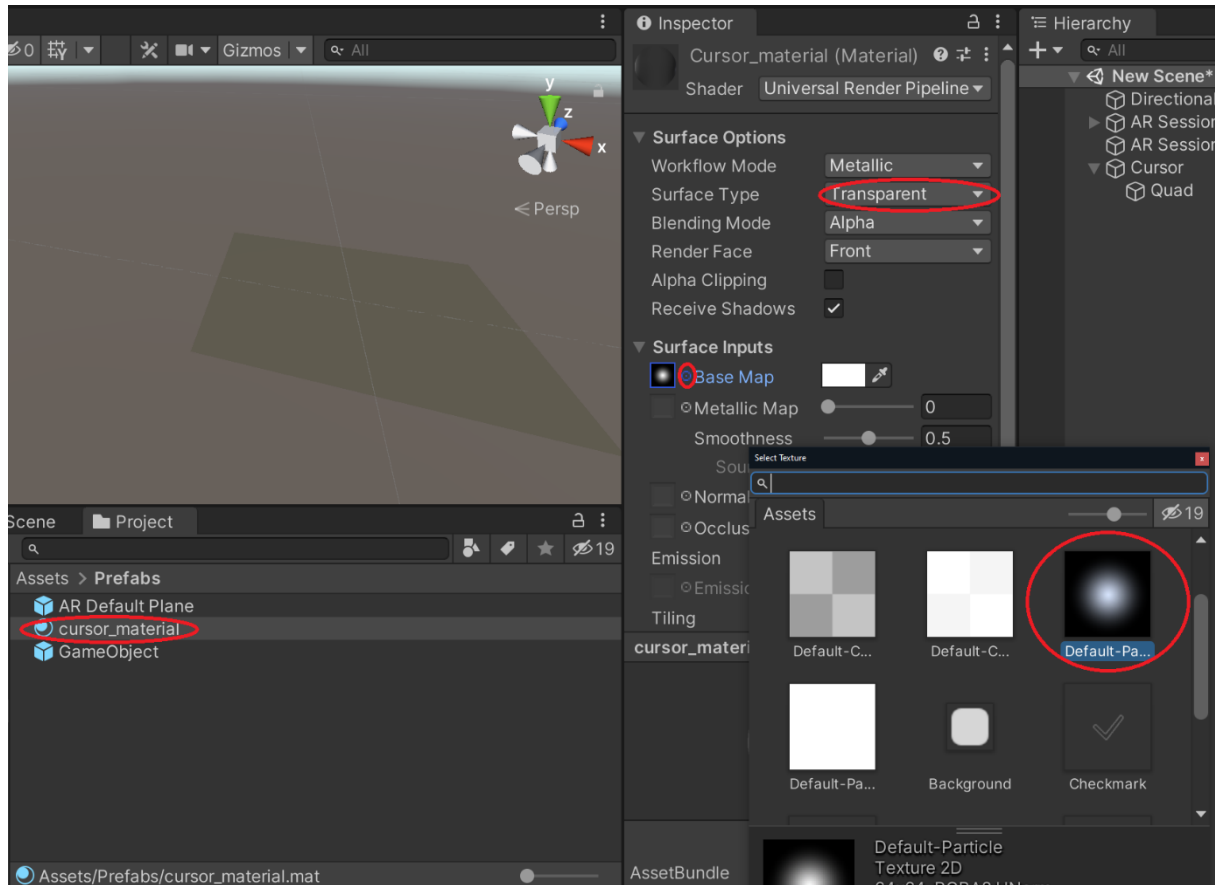
Rotate quad 90 in the x-axis and scale it down to 0.1



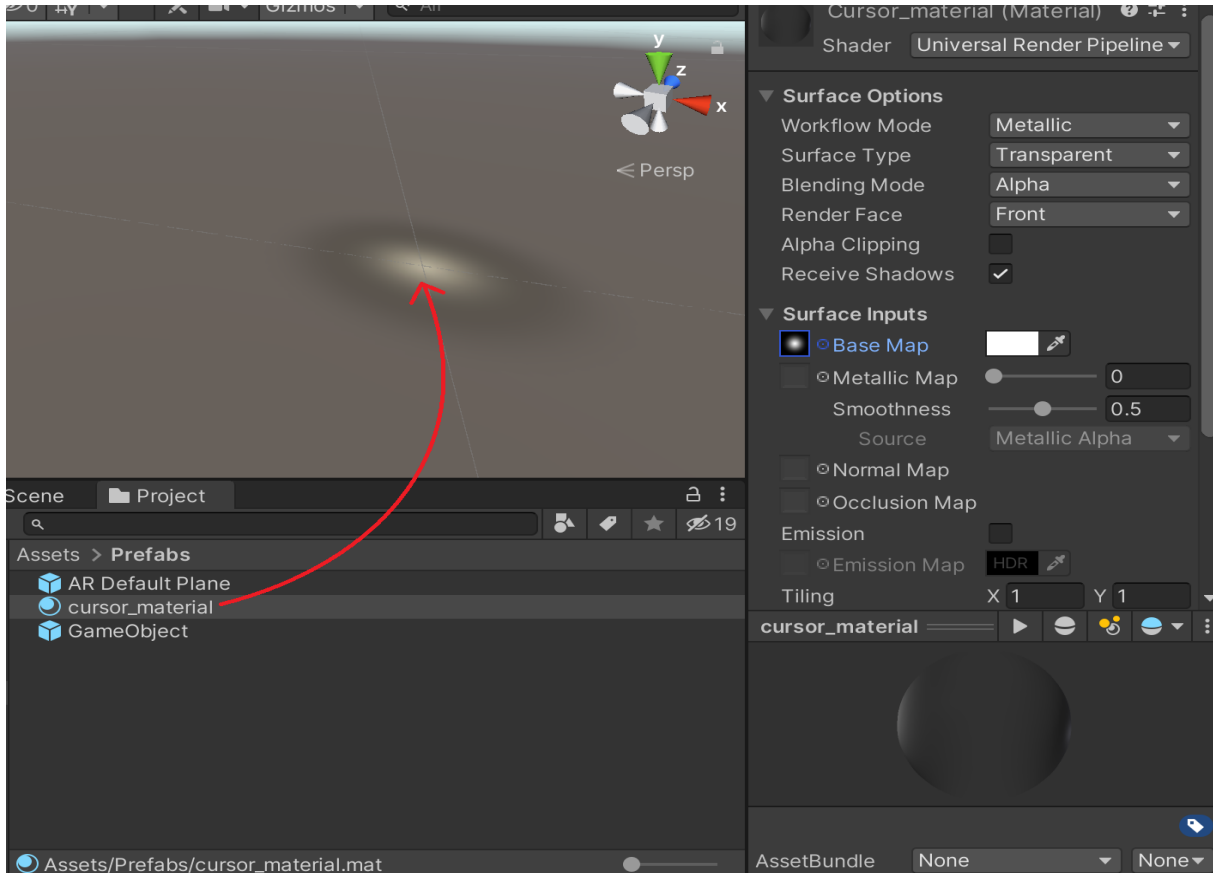
Create a new material in your project window and name it to cursor material



Select the material, set it Transparent, and give it a texture



Then drag and drop the material onto the Quad in the Scene



Create another empty object in the Scene and name it ObjectPlacer

Write IndicatorPlacer code and attach it to an empty ObjectPlacer

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.XR.ARFoundation;
using UnityEngine.XR.ARSubsystems;

public class IndicatorPlacer : MonoBehaviour
{
    public GameObject indicator;
    private ARSessionOrigin arOrigin;
    private ARRaycastManager aRRaycastManager;
    private bool planeDetected = false;

    // Start is called before the first frame update
    void Start()
    {
        aRRaycastManager = FindObjectOfType<ARRaycastManager>();
    }

    // Update is called once per frame
    void Update()
    {
        UpdatePose();
    }

    private void UpdatePose()
    {
        //Physics.Raycast() this exists in unity to cast a ray from any given point in any direction
        //but arfoundation has a build in implementation to cast from the screen
        List<ARRaycastHit> hits = new List<ARRaycastHit>();
        aRRaycastManager.Raycast(new Vector2(Screen.width / 2, Screen.height / 2), hits, TrackableType.Planes);

        if (hits.Count > 0)
        {
            indicator.SetActive(true);
            indicator.transform.position = hits[0].pose.position;
            indicator.transform.rotation = hits[0].pose.rotation;
        }

        else
        {
            indicator.SetActive(false);
        }
    }
}
```

Drag and drop the Cursor object in the scene into the „indicator“ field

Write TapSpawn code and attach it to an empty ObjectPlacer

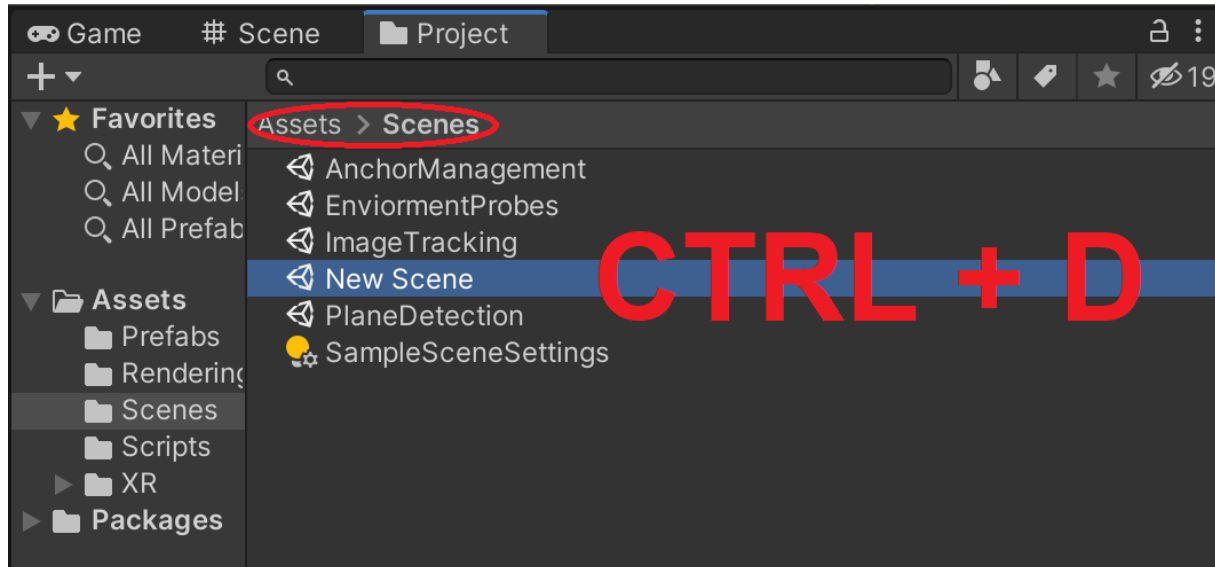
```
1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4
5  public class TapSpawn : MonoBehaviour
6  {
7      public GameObject objectToSpawn;
8      private IndicatorPlacer placementIndicator;
9
10     void Start()
11     {
12         placementIndicator = FindObjectOfType<IndicatorPlacer>();
13     }
14
15     void Update()
16     {
17         if (Input.touchCount > 0 && Input.touches[0].phase == TouchPhase.Began)
18         {
19             GameObject obj = Instantiate(objectToSpawn,
20                                         placementIndicator.indicator.transform.position,
21                                         placementIndicator.indicator.transform.rotation);
22         }
23     }
24 }
25
26
```

Drag and drop the GameObject prefab we created into the „objectToSpawn“ field

BUILD AND RUN THE CODE

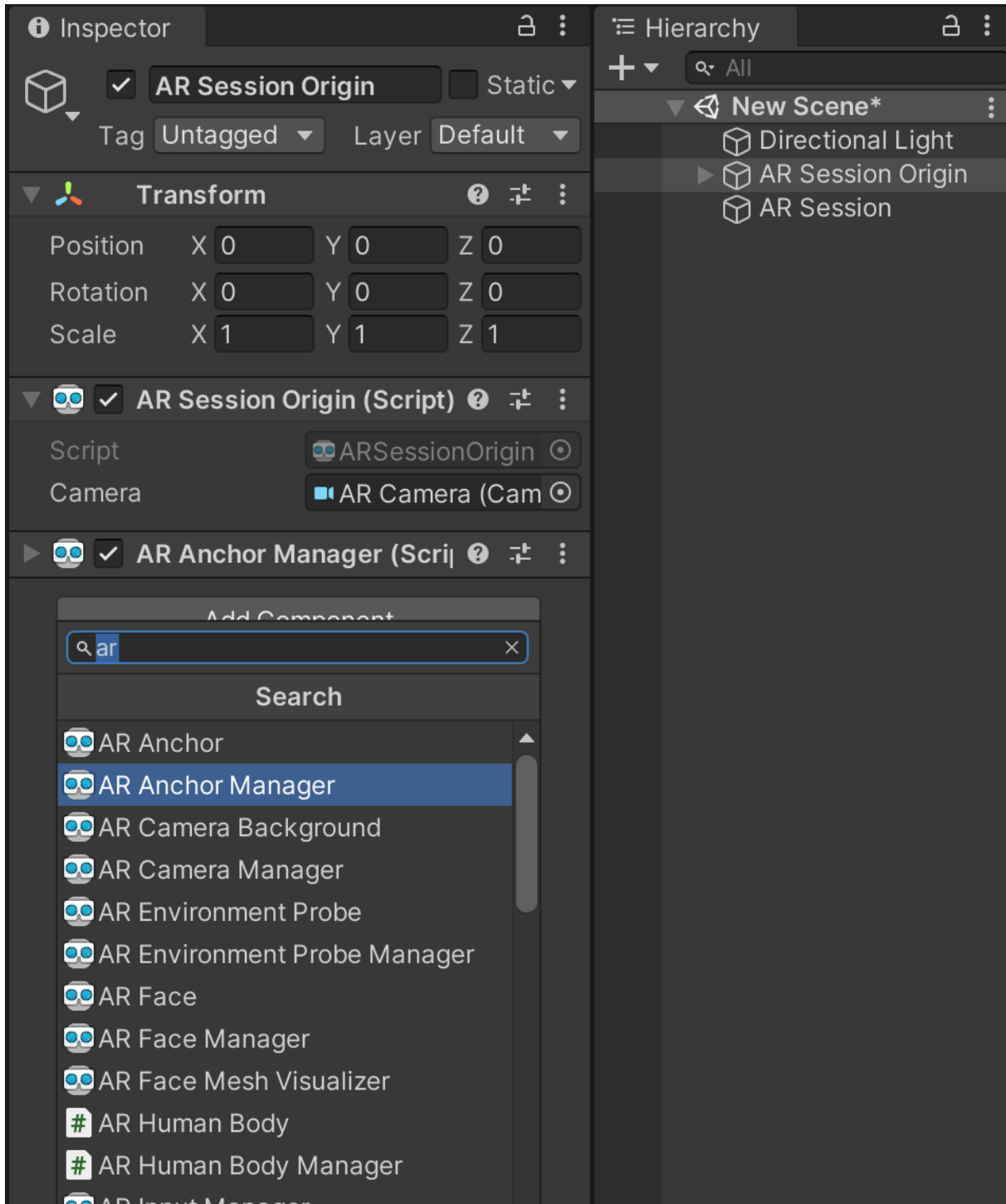
PART 3 – AR ANCHORS

Duplicate the current scene by selecting the Scene in the project window CTRL+D and double-click on it

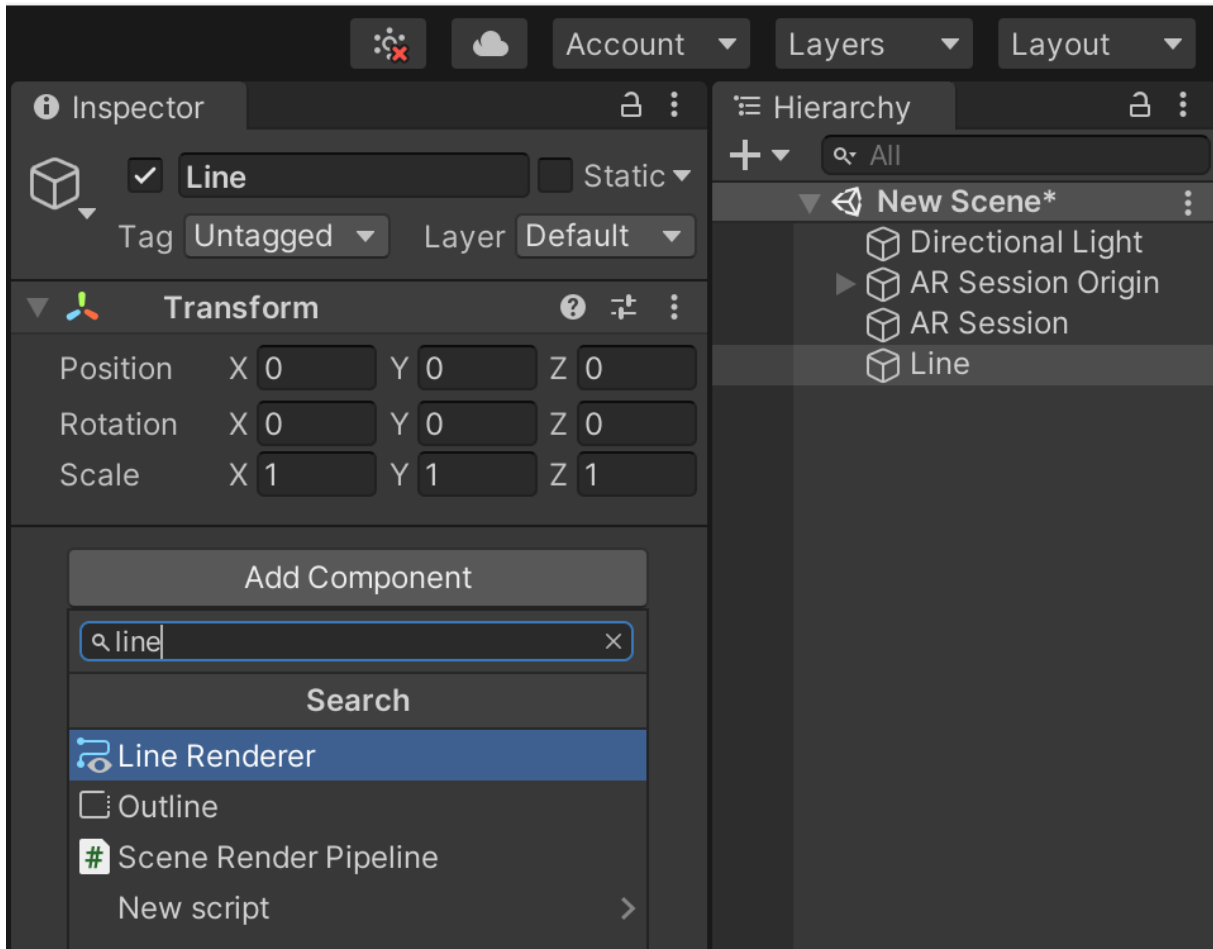


Delete the cursor from the scene and delete AR Plane manager + AR Raycast manager from the AR Session Origin (The ones that we had added previously)

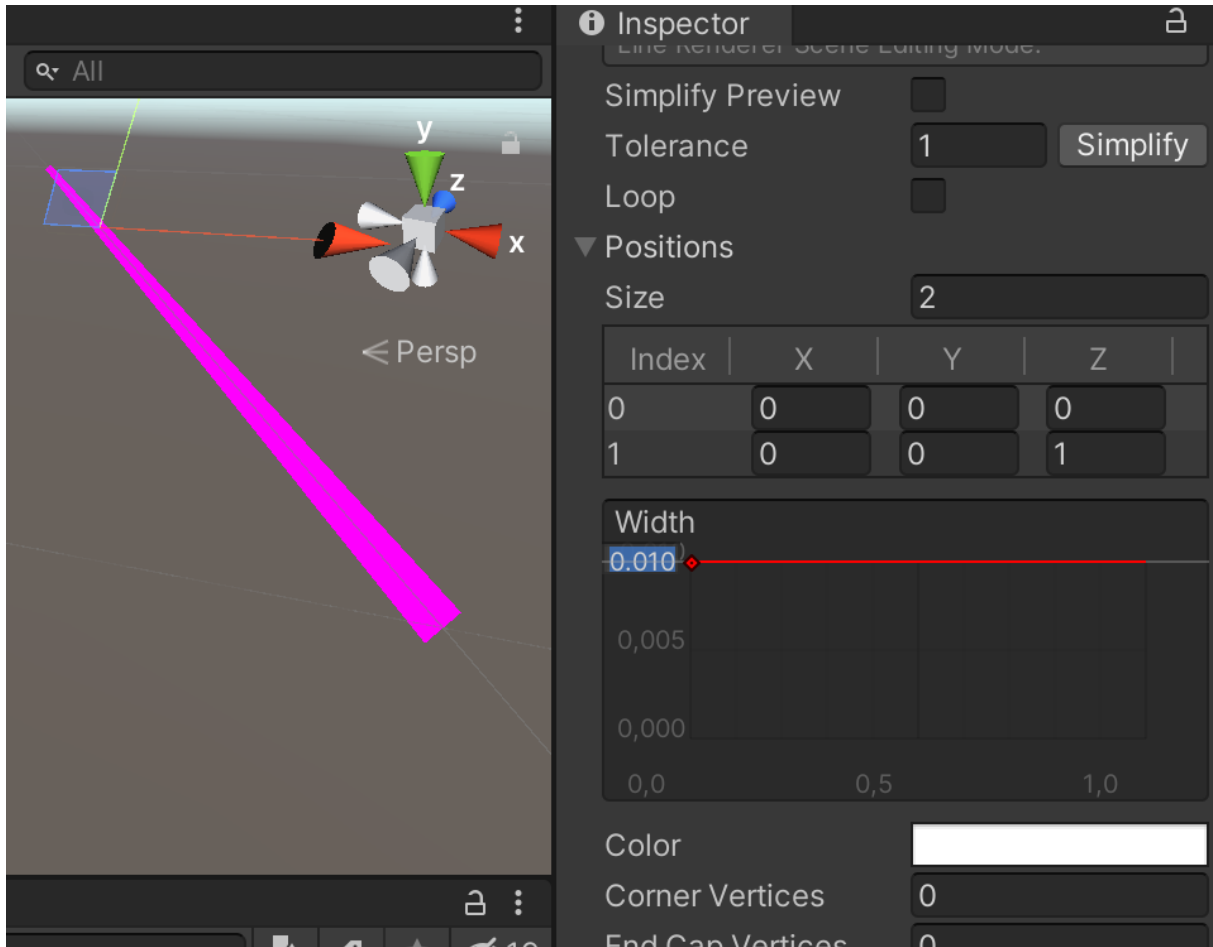
Select AR session origin
Add ar anchor manager



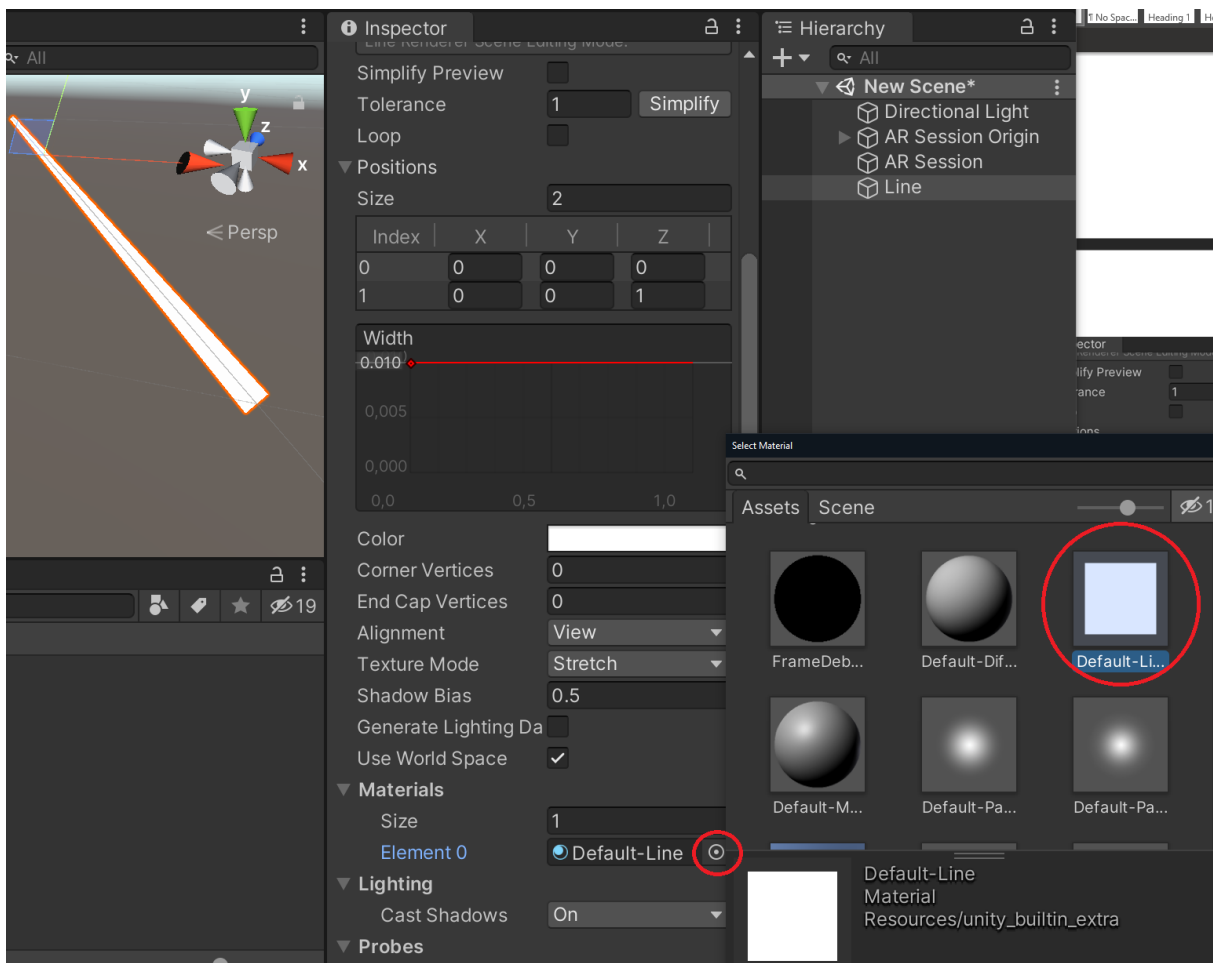
Create an empty and add a line renderer to it

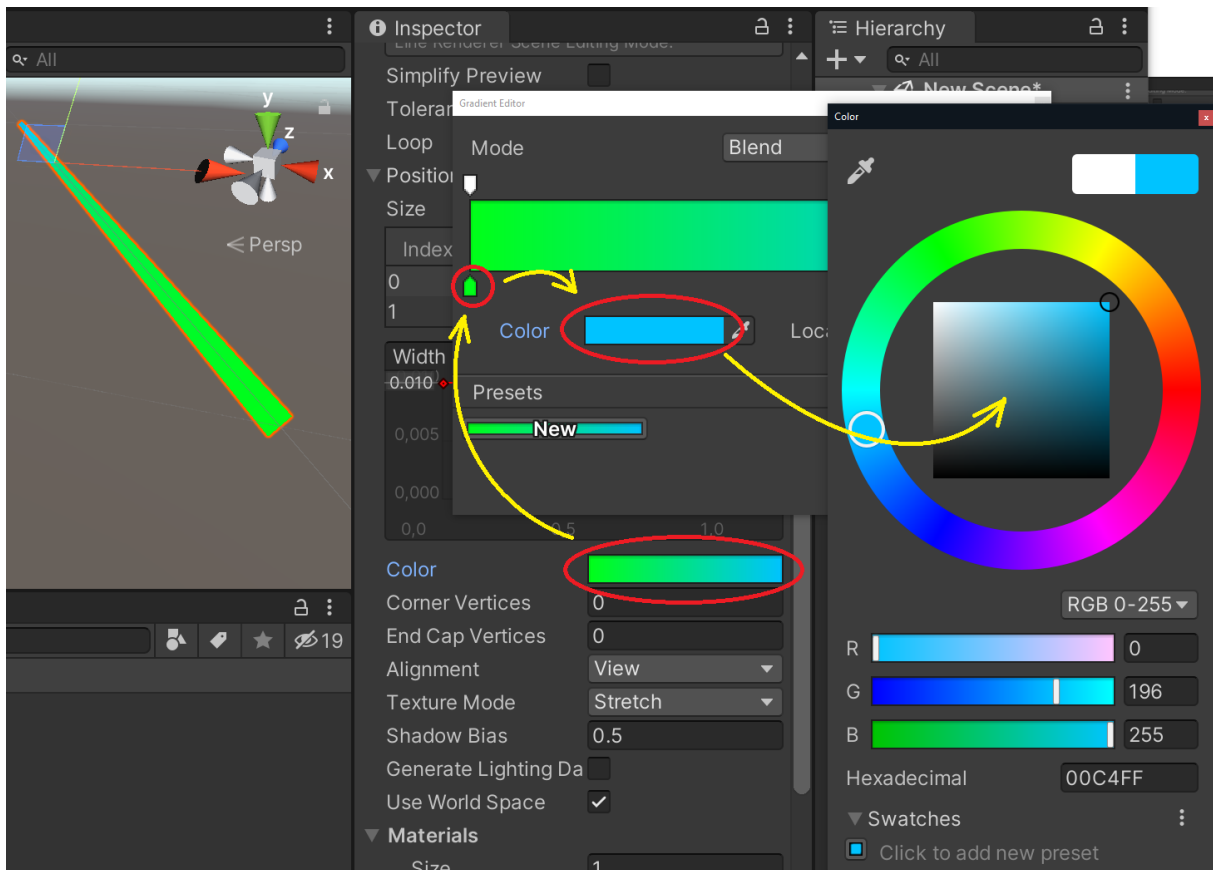


Change the width to 0.01

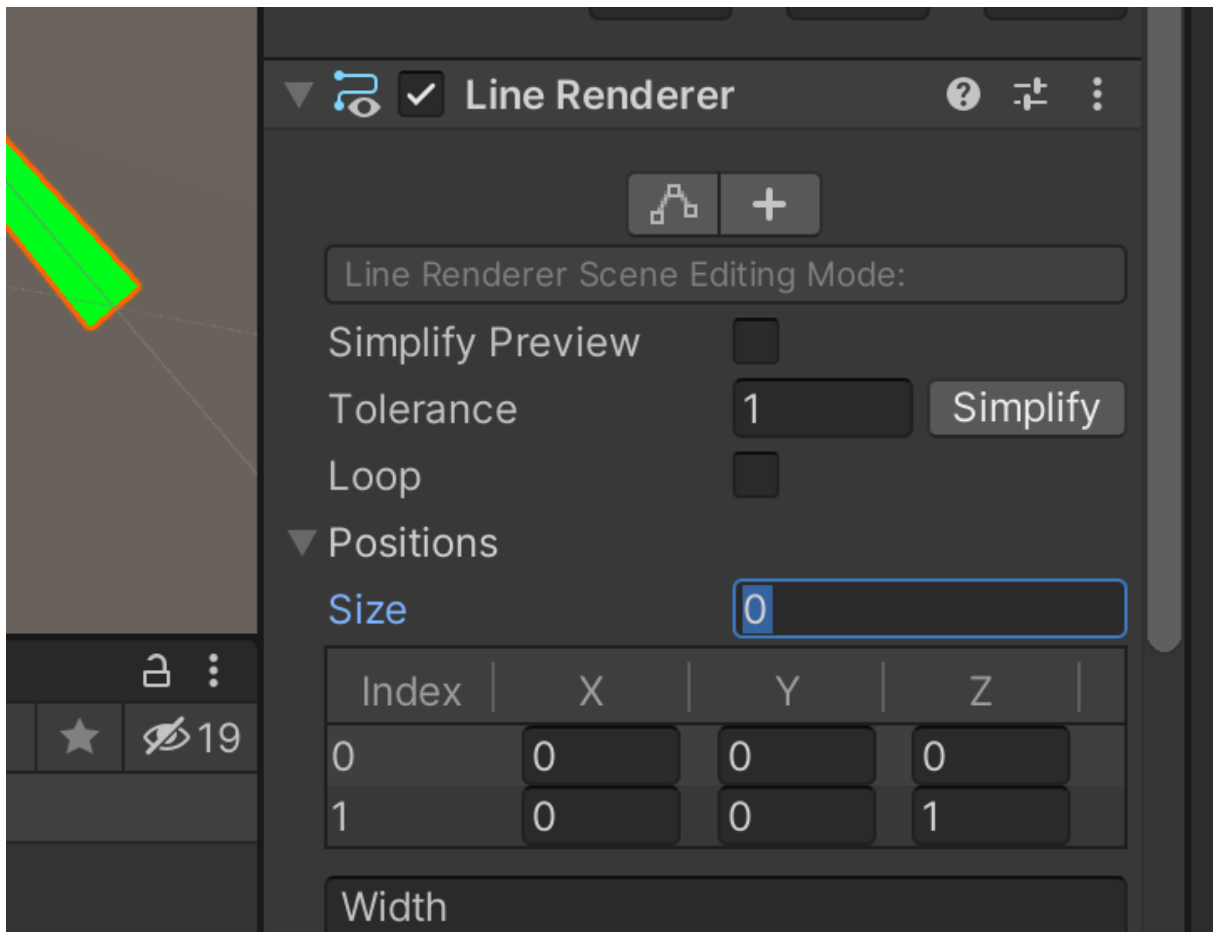


add material and color to the line





Set the number of points to 0



Save as a prefab just like with the previous times

Create a DrawManager script

```
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4 using UnityEngine.XR.ARFoundation;
5 using UnityEngine.XR.ARSubsystems;
6
7 public class DrawManager : MonoBehaviour
8 {
9     private Camera arcamera;
10    public GameObject linePrefab;
11    GameObject line;
12    Vector3 prevPosition;
13
14    private void Start()
15    {
16        arcamera = Camera.main;
17    }
18
19    void Update()
20    {
21        if (Input.touchCount > 0)
22        {
23            Draw();
24        }
25    }
26
27    private void Draw()
28    {
29        Vector3 position = arcamera.ScreenToWorldPoint(new Vector3(Screen.width / 2, Screen.height / 2, 0.3f));
30        if (Input.touches[0].phase == TouchPhase.Began)
31        {
32            line = Instantiate(linePrefab, position, Quaternion.identity);
33            line.AddComponent<ARAnchor>();
34        }
35        else
36        {
37            AddPointToLine(position);
38        }
39    }
40
41    private void AddPointToLine(Vector3 position)
42    {
43        if (prevPosition == null)
44        {
45            prevPosition = position;
46            return;
47        }
48        if (Vector3.Distance(position, prevPosition) > 0.1) {
49            prevPosition = position;
50
51            LineRenderer renderer = line.GetComponent<LineRenderer>();
52            renderer.positionCount += 1;
53            renderer.SetPosition(renderer.positionCount - 1, position);
54        }
55    }
56 }
57
```

And add it to a new empty object in the scene

Drag and drop the line prefab we just created into the „linePrefab“ field

BUILD AND RUN THE CODE