TECHNISCHE UNIVERSITÄT DARMSTADT

Informatik

HCI Lab

IVAR: Lab 2

Introduction to unity roll-a-ball

31.10.2023

Labs

17.10	~~Website (hugo) + unity setup~~
24.10	~~Reverse classroom topics~~
31.10	**Introduction to Unity (roll-a-ball)**
	Roll-a-ball in VR
07.11	VR parkour
14.11	Pitch your locomotion and interaction idea
21.11	Reverse classroom 1
27.11	Reverse classroom 2

Reverse classroom topics

First name	Choice
Sebastian Rudolf	Animations in Unity3D
Lea Sophie	How To Grab Objects With Hands In VR - Oculus Interaction SDK
Michael	How to jump in Unity (with or without physics)
Elif	How to move objects in Unity (3 methods)
Frank	How to setup Meta Avatar in Unity - VR Tutorial
Nadine	How to use Cameras in Unity: Cinemachine Overview and Brain Explained!
Chen	Introduction to VR in Unity - PART 9 : CLIMBING
Hanjo	Inverse kinematics
Karolis	Meta Quest Passthrough Tutorial in Unity - PART 2 : Styling
Alexander	Tunnelling Vignette against Motion Sickness
Jonas	Unity Shader Graph (Trails)
Austin	Unity VR Game Basics - PART 7 - Continuous Movement
Luis	Writing Your First Shader In Godot
Dillon	Writing Your First Shader In Godot
Flavian	?

Pick up your Meta Quest

31.10 Tue. 14-16h
01.11 Wed. 9-12h, 13-16h
02.11 Thur. 9-12h, 13-16h

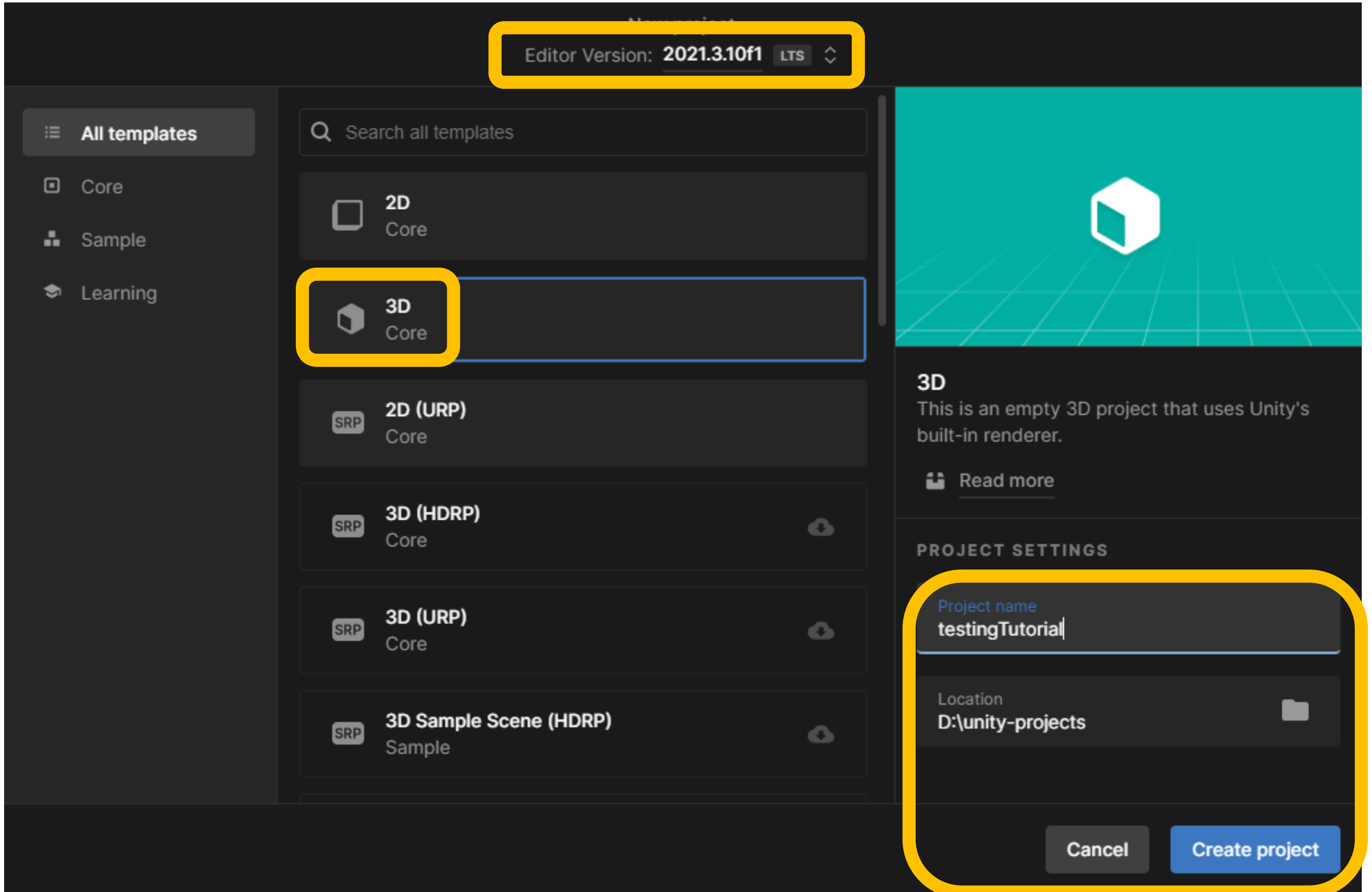
Come to **A307, S2|02**, to pick up your Quest 2!

If you could not make it, please contact
wen-jie.tseng@tu-darmstadt.de or
willich@tk.tu-darmstadt.de

basics of Unity

Create a new 3D project from UnityHub

Go to **Projects** tab > press **NEW** to create a new project



Editor Version: 2021.3.10f1 LTS

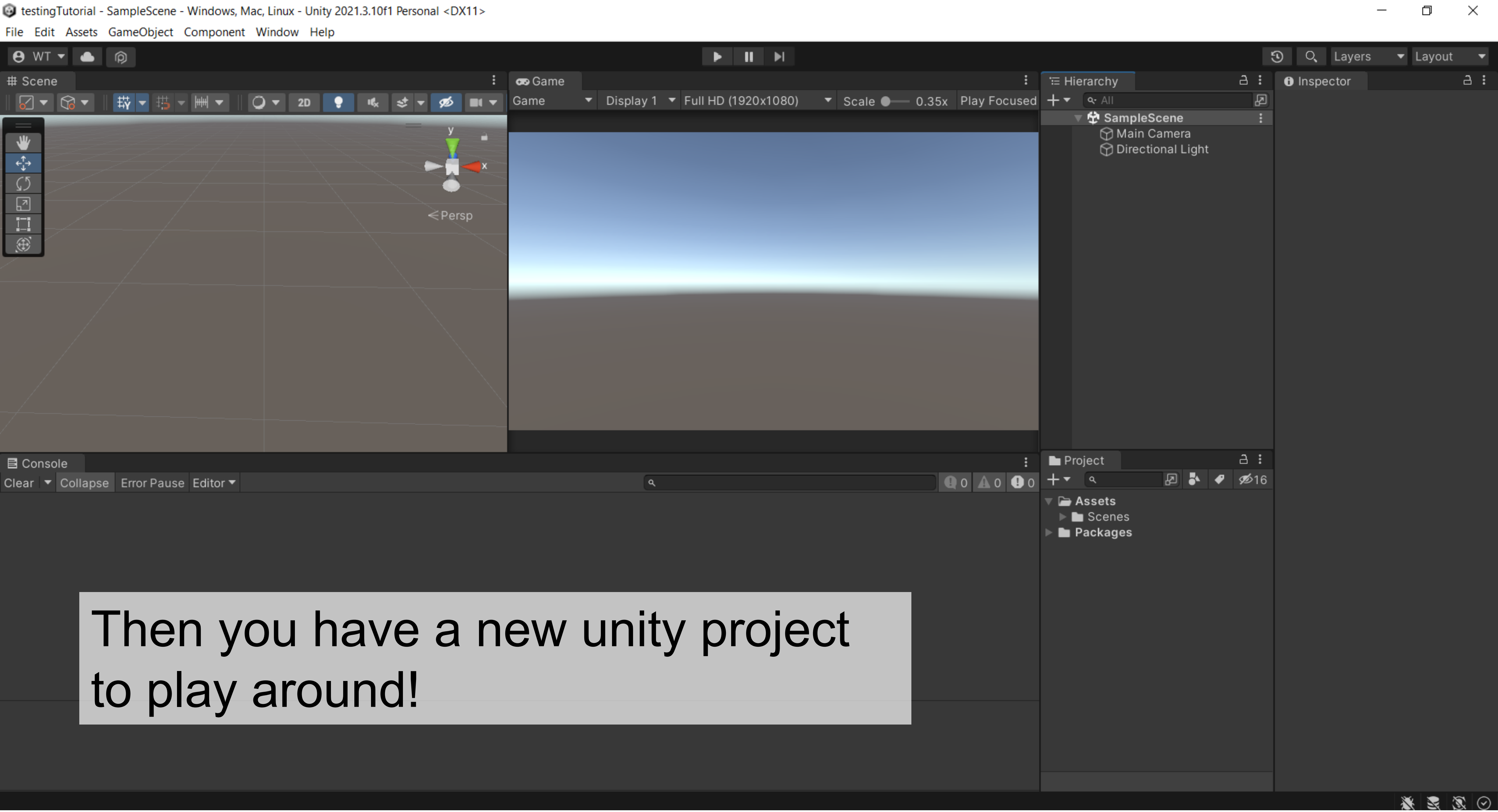
3D

This is an empty 3D project that uses Unity's built-in renderer.

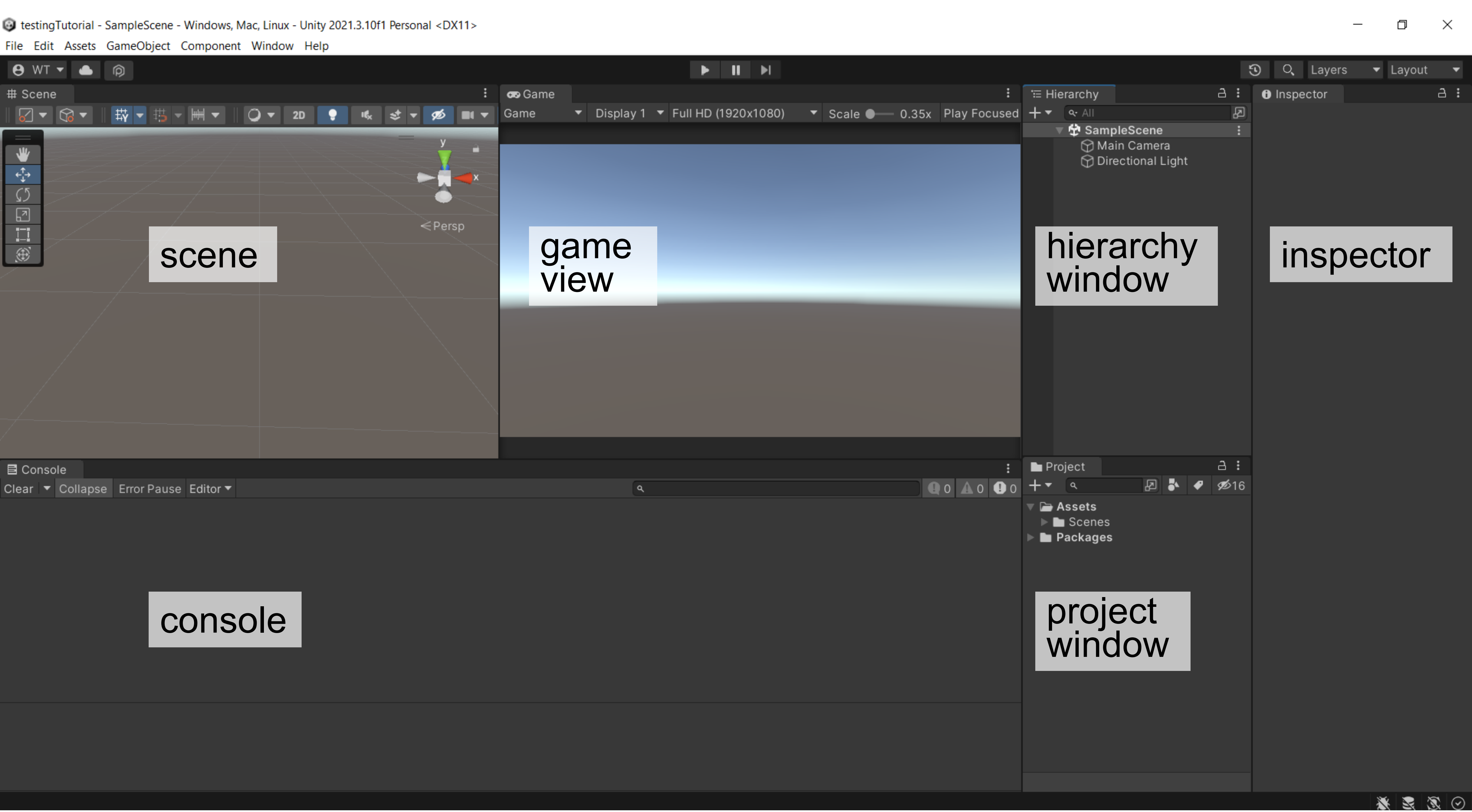
PROJECT SETTINGS

Project name
testingTutorial

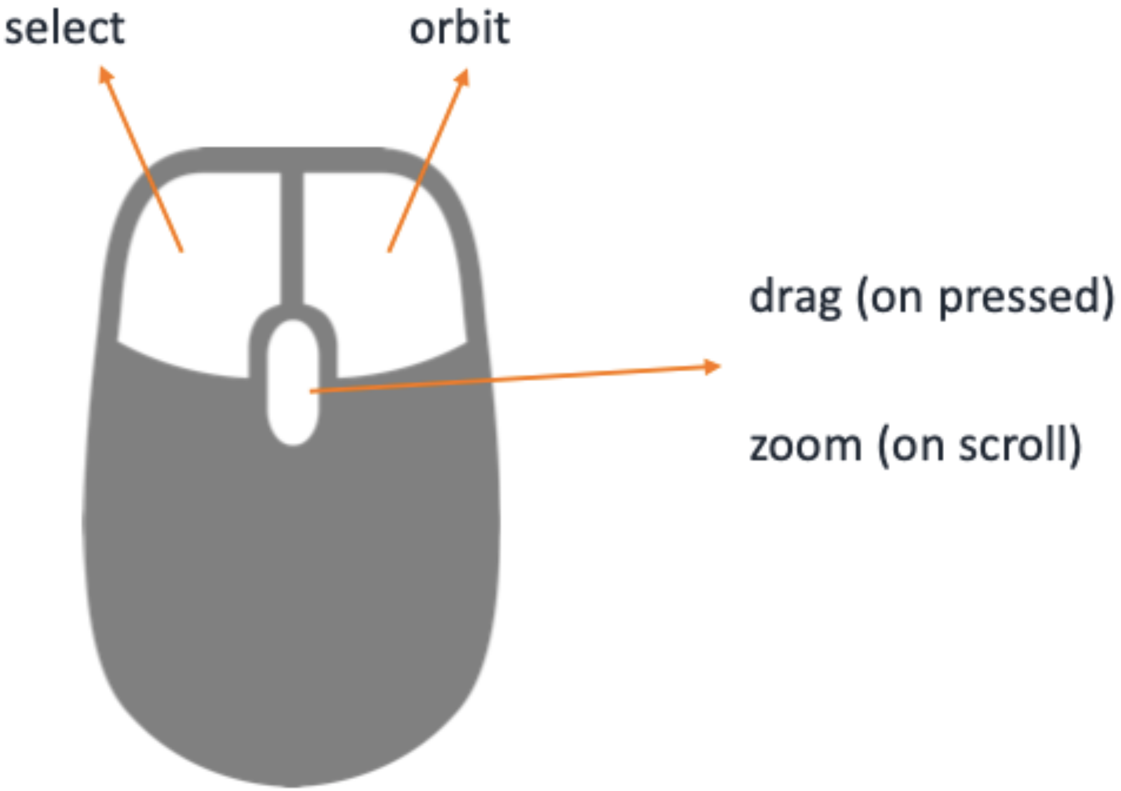
Location
D:\unity-projects



Then you have a new unity project to play around!


testingTutorial - SampleScene - Windows, Mac, Linux - Unity 2021.3.10f1 Personal <DX11>
File Edit Assets GameObject Component Window Help
WT
Layers
Layout
Scene
2D
Persp
Game
Game
Display 1
Full HD (1920x1080)
Scale 0.35x
Play Focused
Hierarchy
All
SampleScene
Main Camera
Directional Light
Inspector
Console
Clear
Collapse
Error Pause
Editor
0
0
0
Project
16
Assets
Scenes
Packages
scene
game view
hierarchy window
inspector
console
project window

Scene navigation



select

orbit

drag (on pressed)

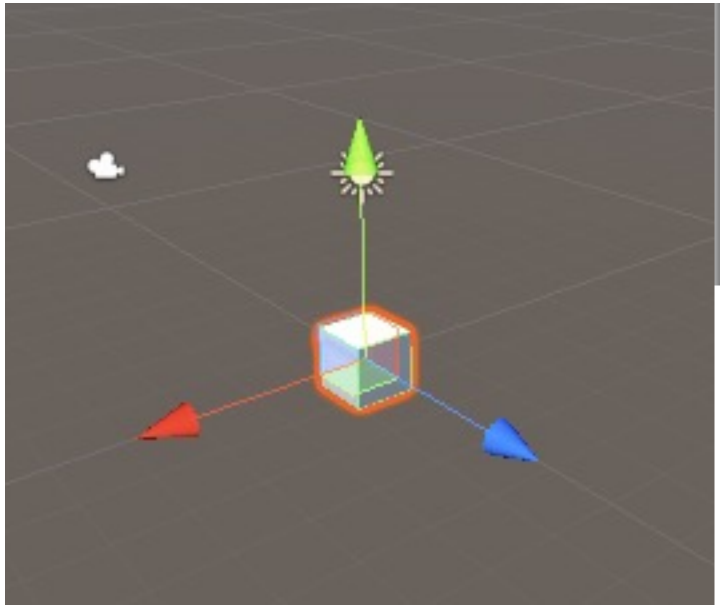
zoom (on scroll)

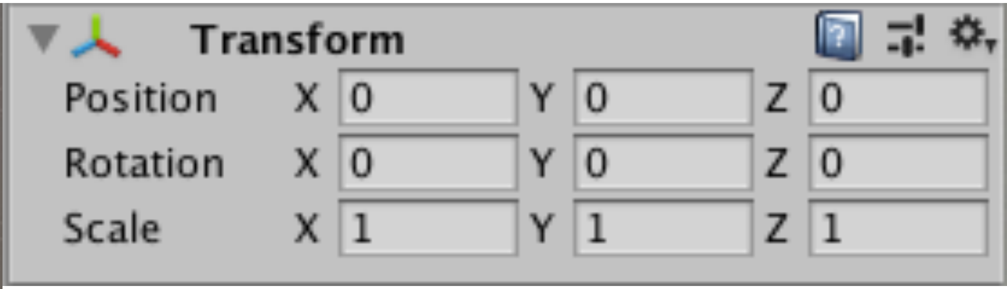
Flythrough mode

Right click + WASD EQ

Transform

- The GameObject has Components that determine their behavior
- All objects have the **Transform** component to determine the object's
  - Position
  - Rotation
  - Scale



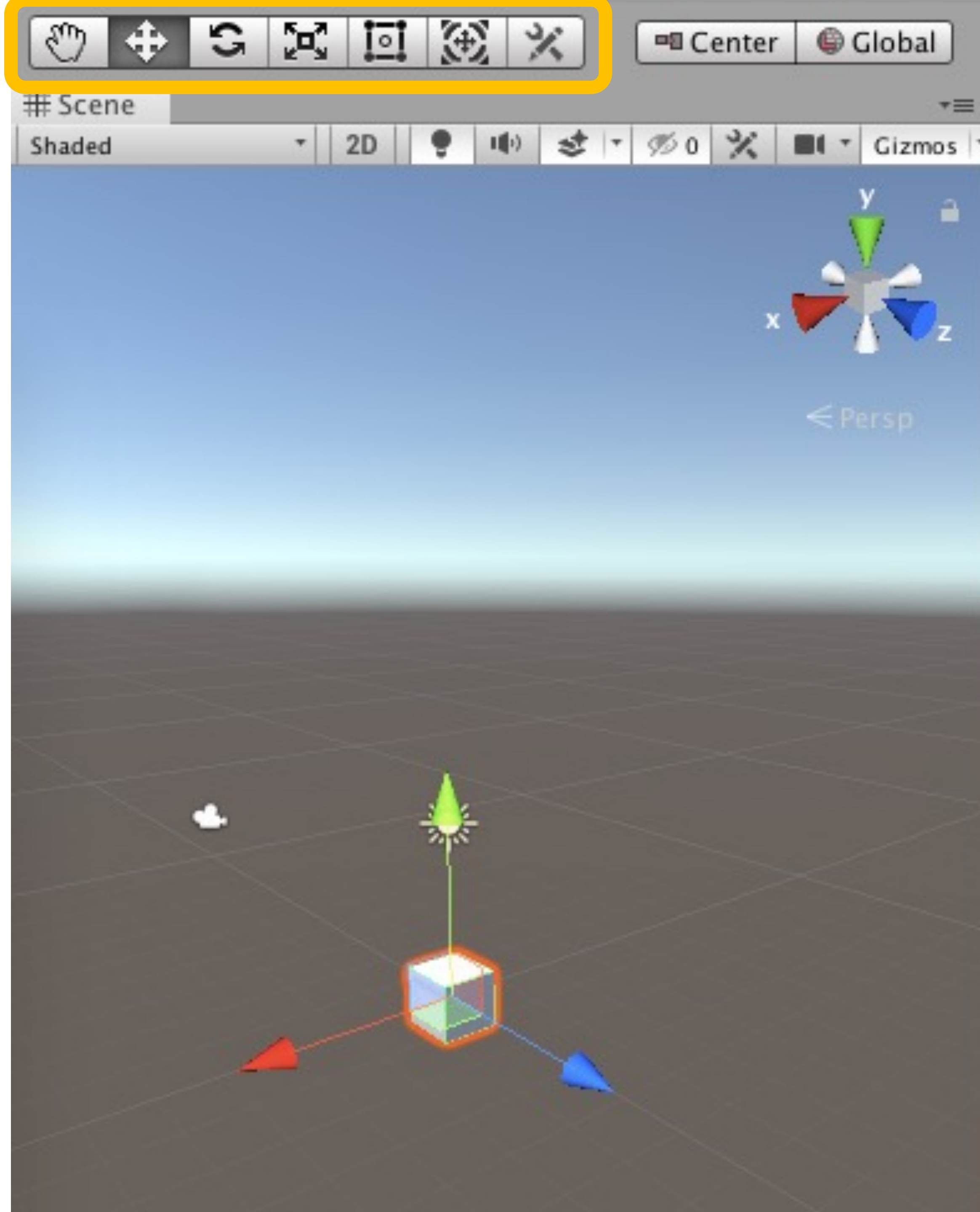


Transform

Position	X	0	Y	0	Z	0
Rotation	X	0	Y	0	Z	0
Scale	X	1	Y	1	Z	1

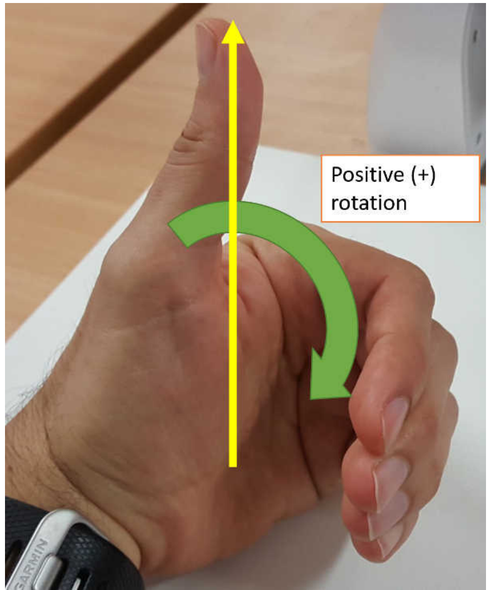
Coordinate system

- Change position, rotation, scales in the scene view

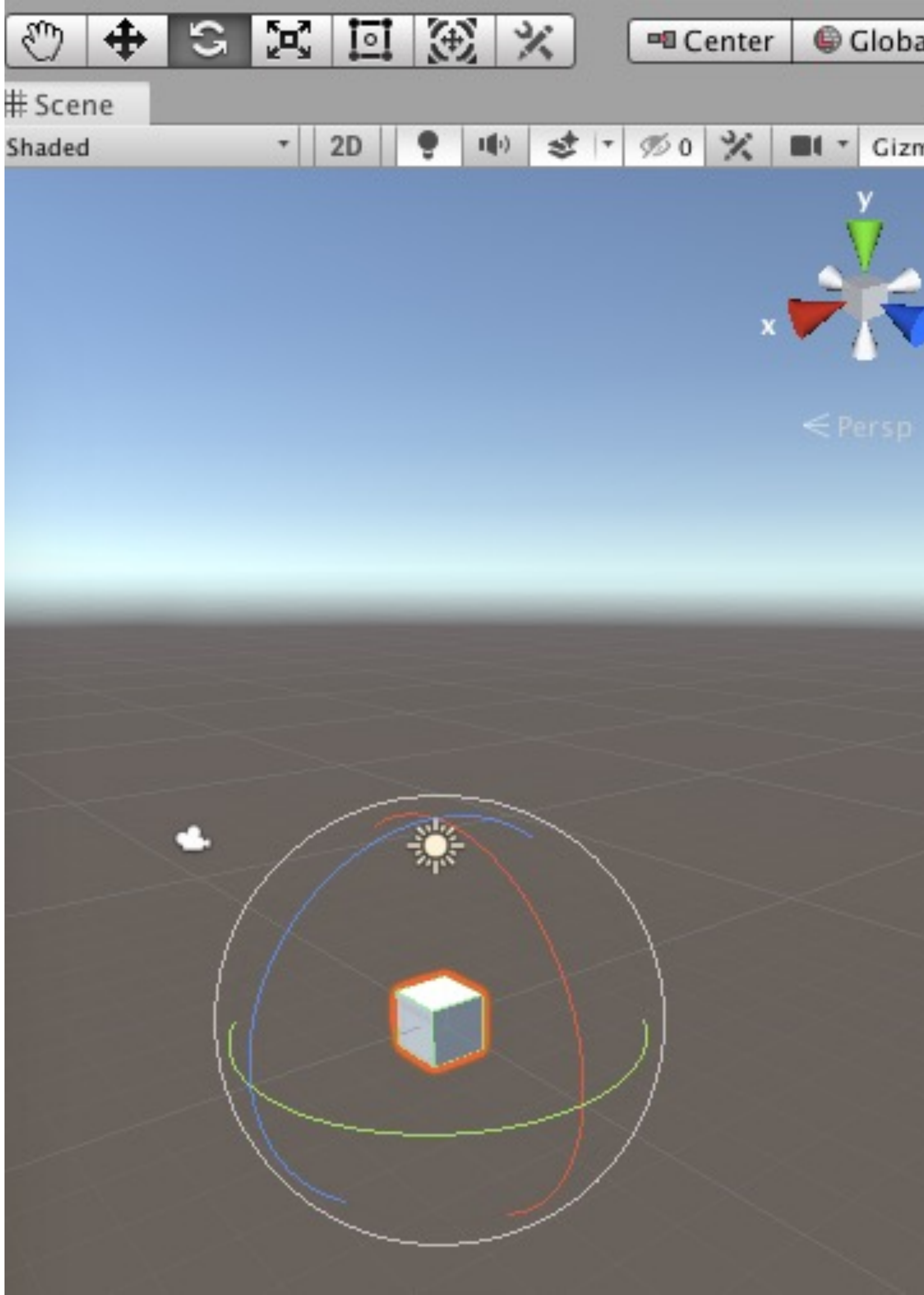


Rotation

- Can be expressed in EulerAngles (x, y, z)
- Positive and negative rotation using left hand coordinate
- Rotates the GameObject’s local axis

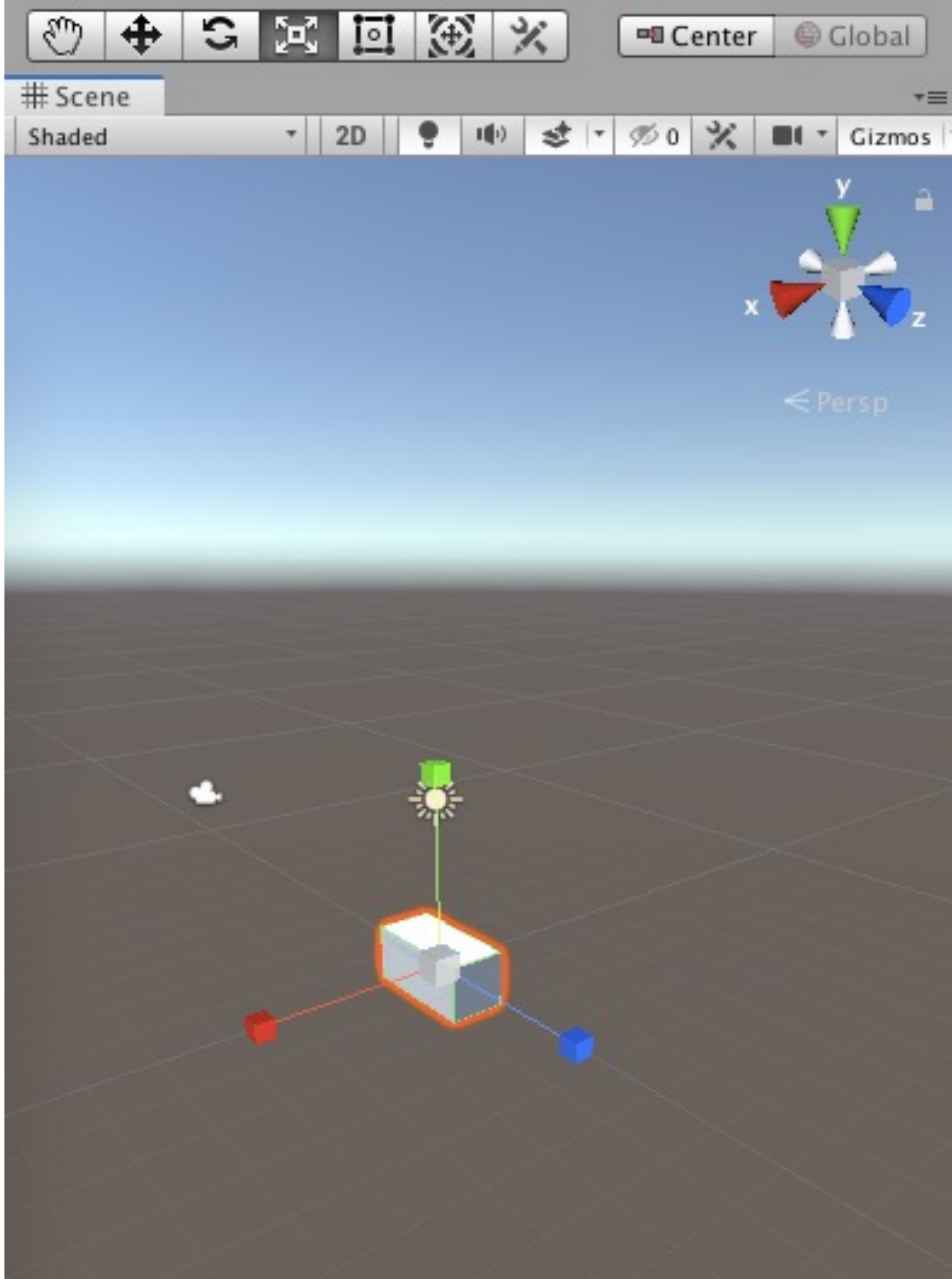


Positive (+) rotation

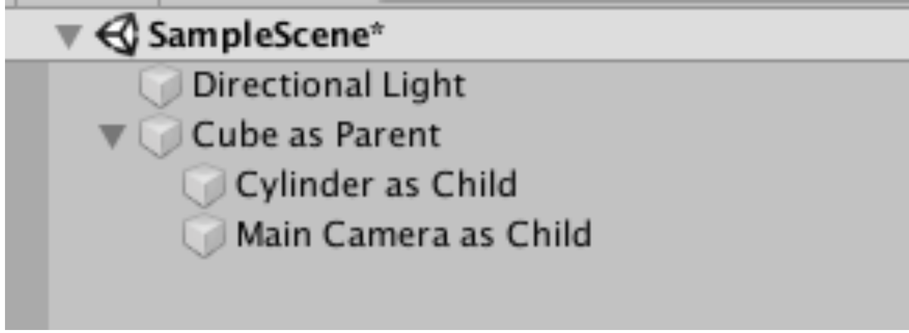


Scale

- Scale > 1: increase the size of an object
- Scale < 1: reduce the size of an object



Parent – Child



- SampleScene*
  - Directional Light
  - Cube as Parent
    - Cylinder as Child
    - Main Camera as Child

- The transform of the child is relative to the parent’s

- For example:
  - Character carries the Camera
  - GameObject made of many GameObjects
  - Grouping things together

- To remove this relationship, just drag child out of the hierarchy of parent.

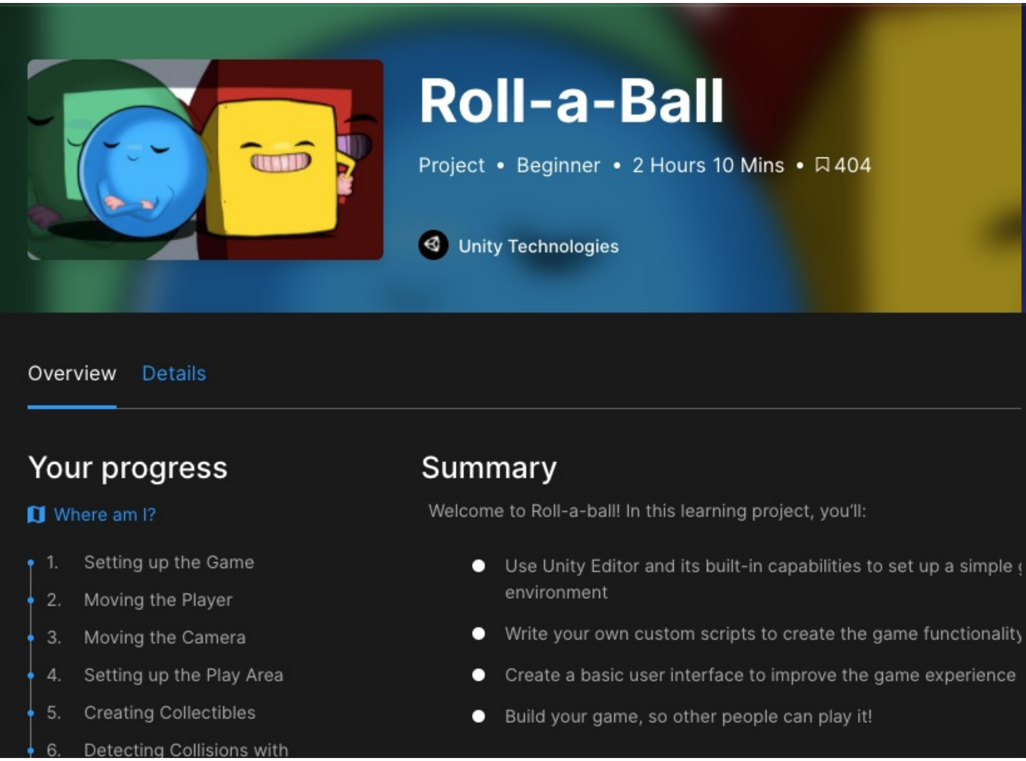
unity roll a ball

Online resources

- There are many materials ranging from the beginner to advance on
  - Unity projects (https://learn.unity.com/projects)
  - Unity tutorial (https://learn.unity.com/tutorials)
  - **GitHub, YouTube**, etc.

Some projects to start with

- Roll-a-Ball (https://learn.unity.com/project/roll-a-ball?uv=2019.4)



Roll-a-Ball

Project • Beginner • 2 Hours 10 Mins • 404

Unity Technologies

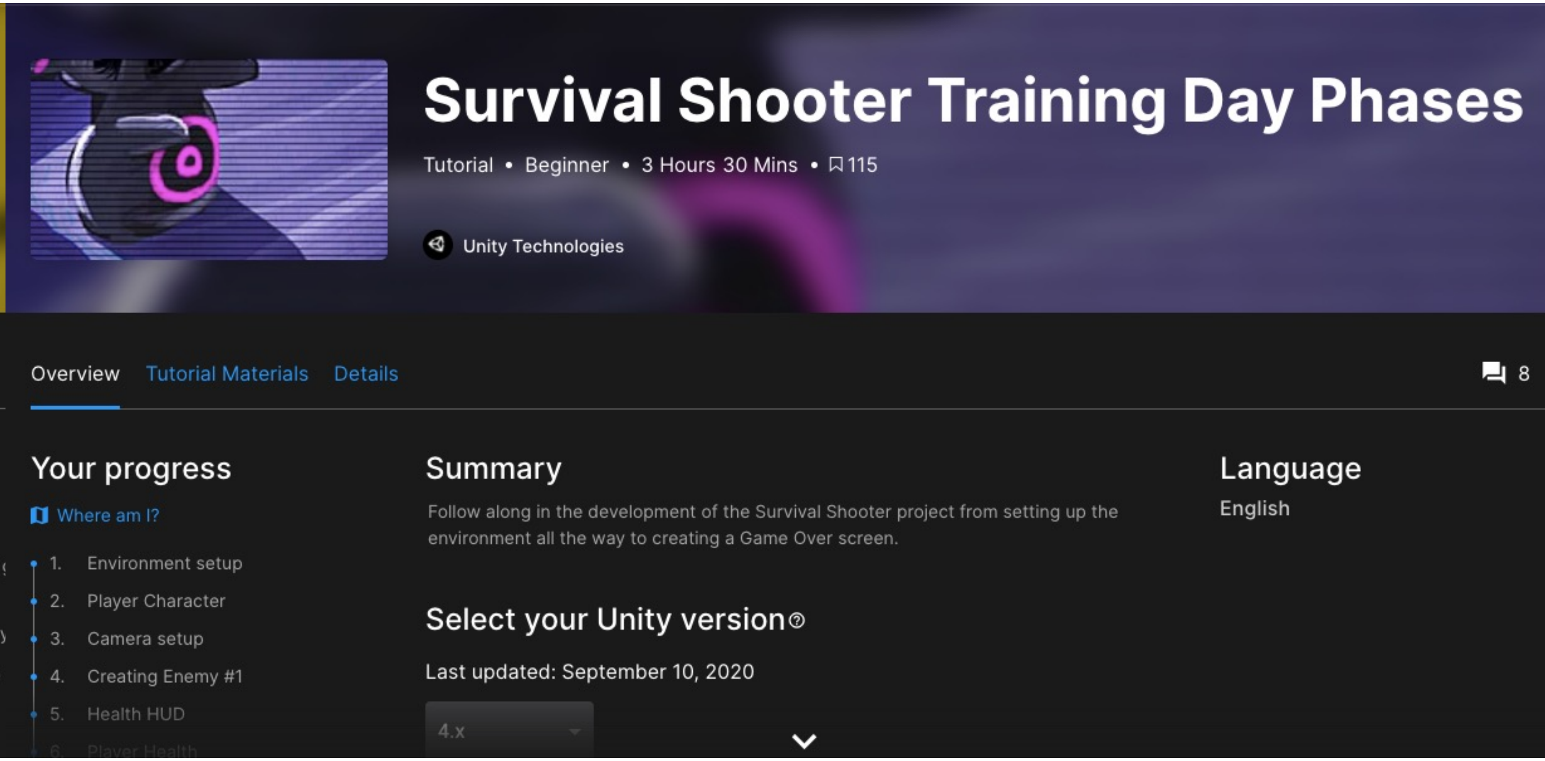
Your progress

1. Setting up the Game
2. Moving the Player
3. Moving the Camera
4. Setting up the Play Area
5. Creating Collectibles
6. Detecting Collisions with

Summary

Welcome to Roll-a-ball! In this learning project, you'll:

- Use Unity Editor and its built-in capabilities to set up a simple g environment
- Write your own custom scripts to create the game functionality
- Create a basic user interface to improve the game experience
- Build your game, so other people can play it!



Survival Shooter Training Day Phases

Tutorial • Beginner • 3 Hours 30 Mins • 115

Unity Technologies

Your progress

1. Environment setup
2. Player Character
3. Camera setup
4. Creating Enemy #1
5. Health HUD
6. Player Health

Summary

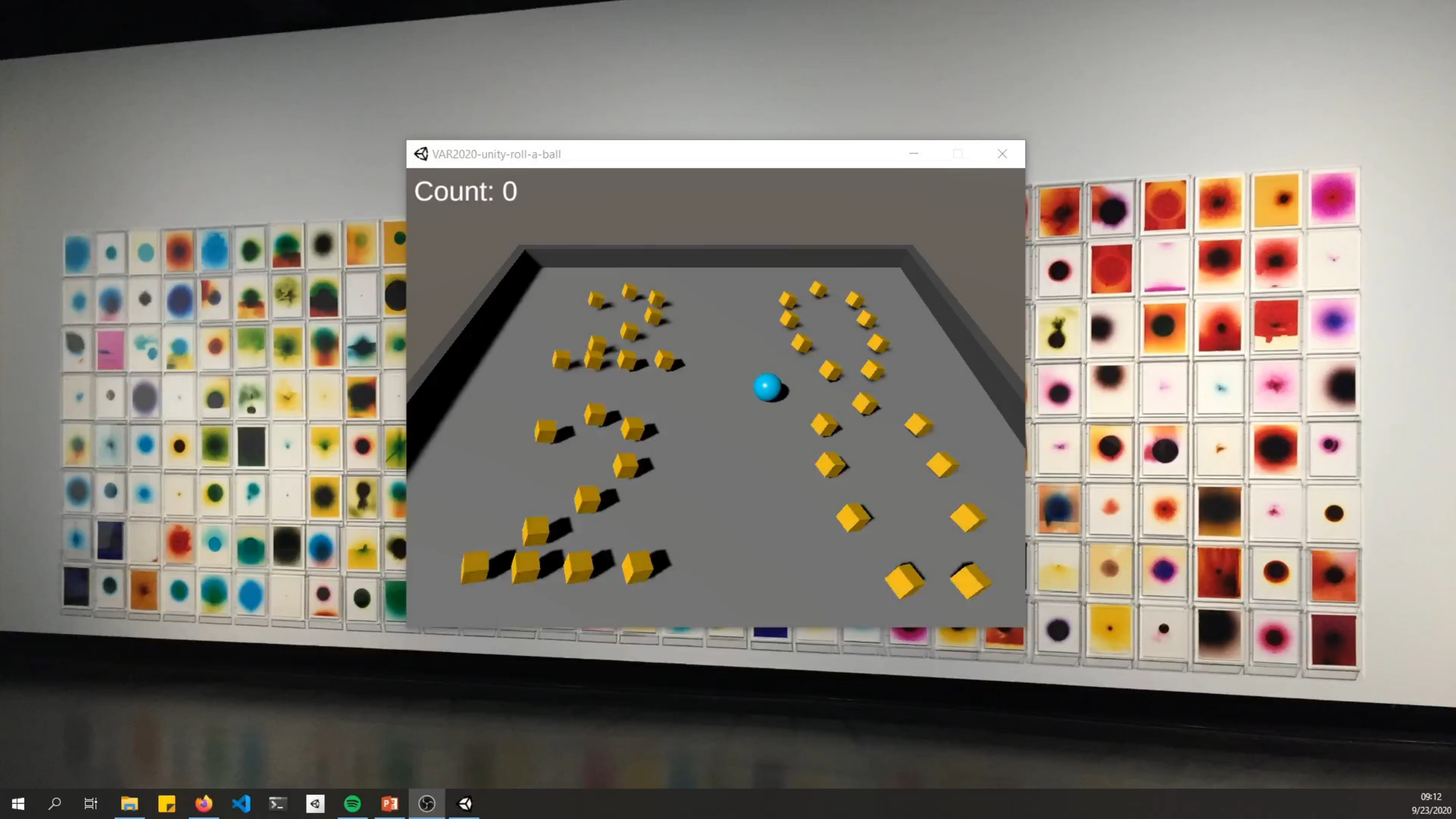
Follow along in the development of the Survival Shooter project from setting up the environment all the way to creating a Game Over screen.

Language

English

Select your Unity version

Last updated: September 10, 2020



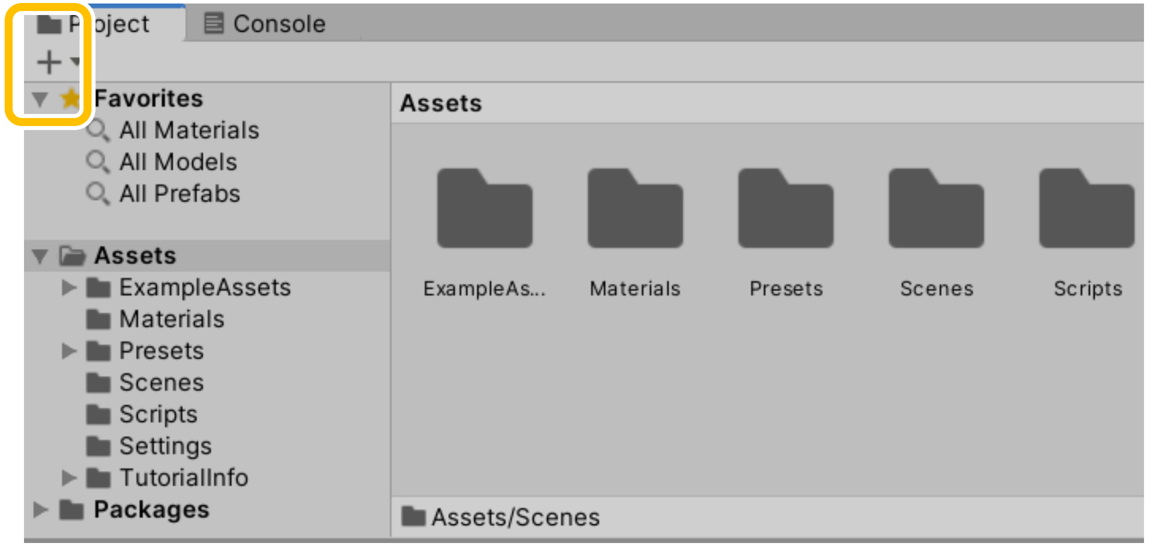
Count: 0

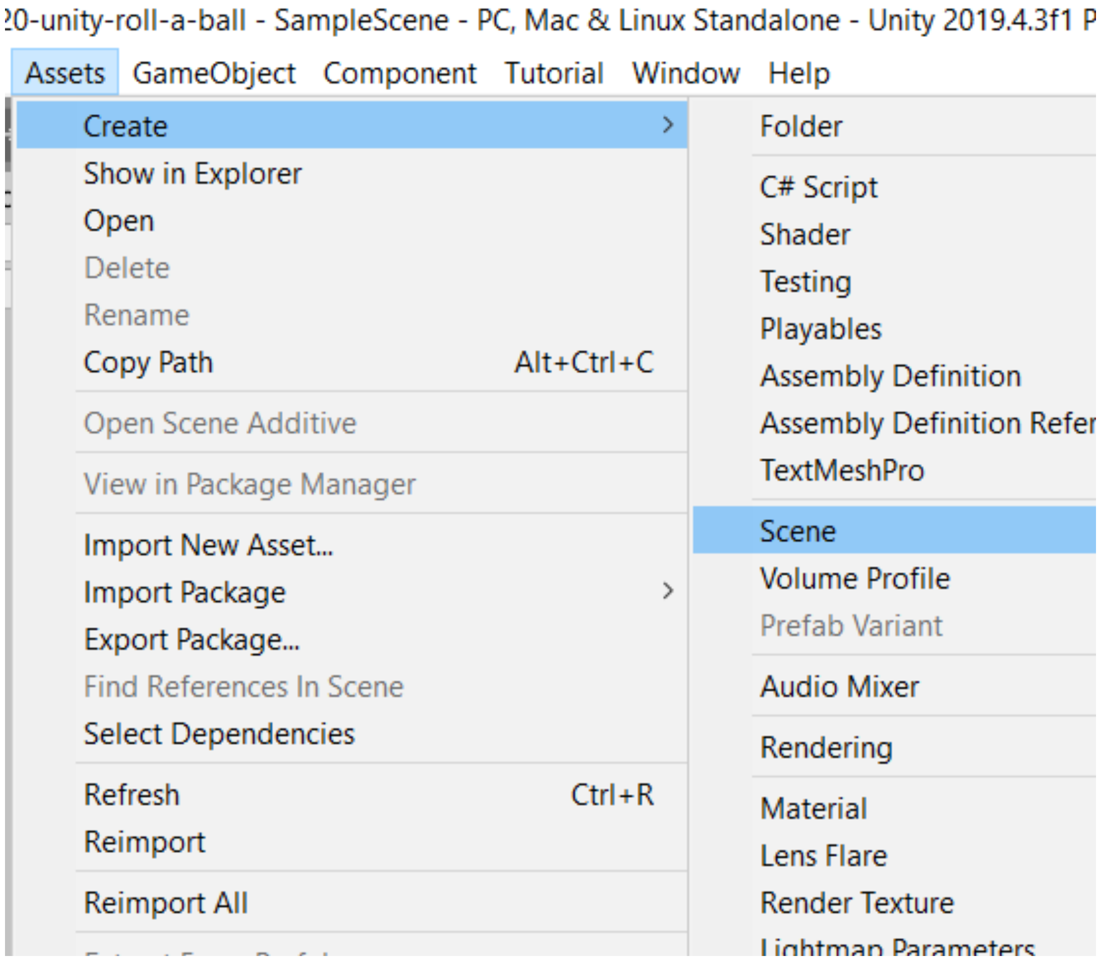
The goal of the game:

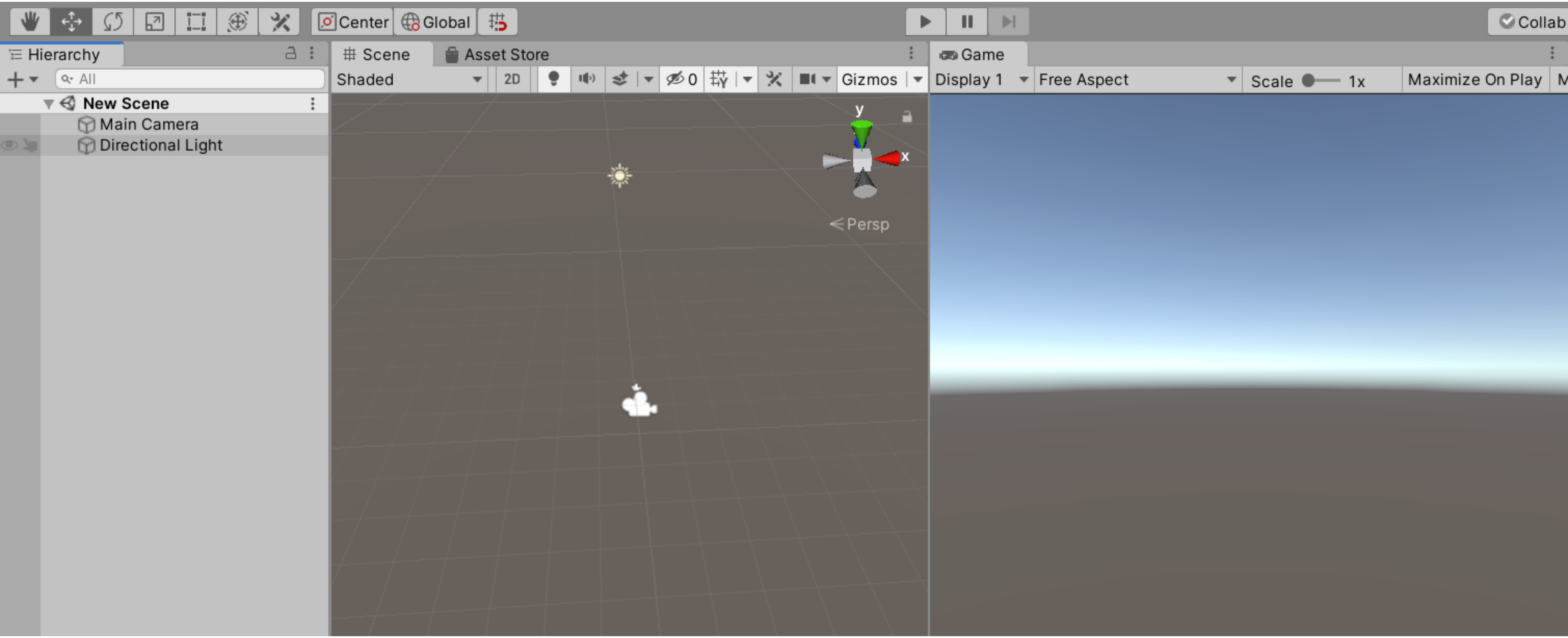
- when Player hits a Pick-up, the Pick-up disappears and increase the score.
- if score > X, win.

create a new Scene

- Assets > create > scene
- Use project window

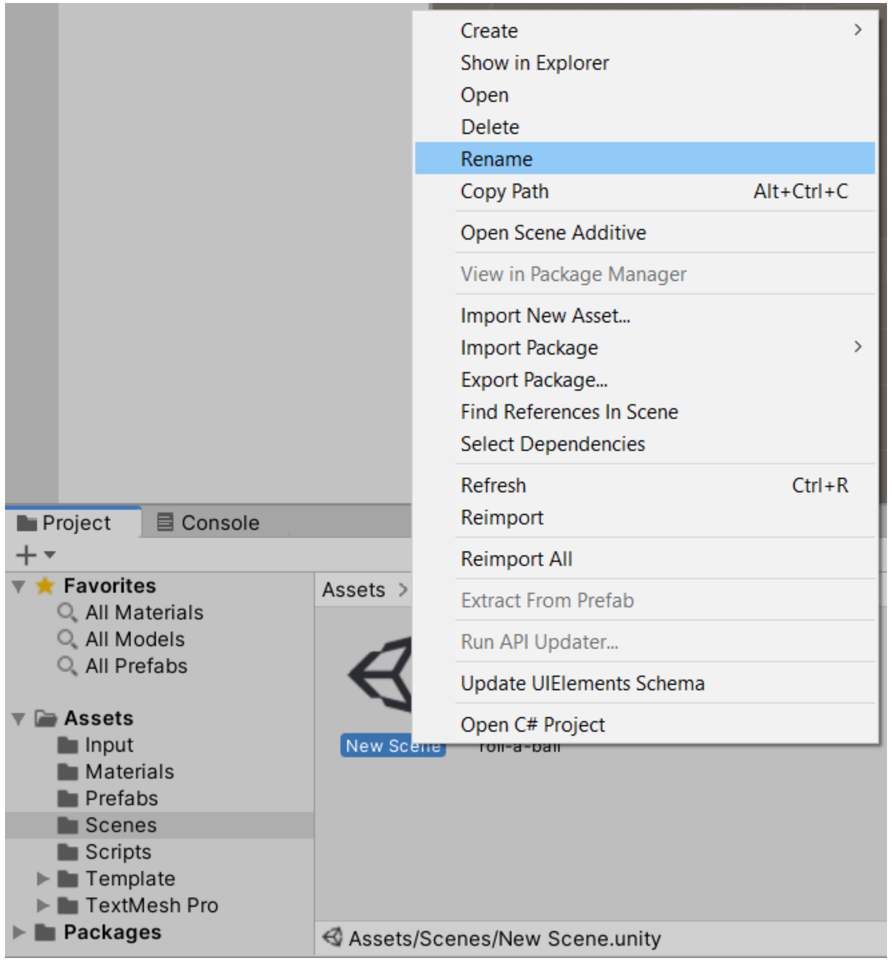





Center
Global
Collab
Hierarchy
All
New Scene
Main Camera
Directional Light
Scene
Asset Store
Shaded
2D
0
Gizmos
y
x
Persp
Game
Display 1
Free Aspect
Scale
1x
Maximize On Play
M

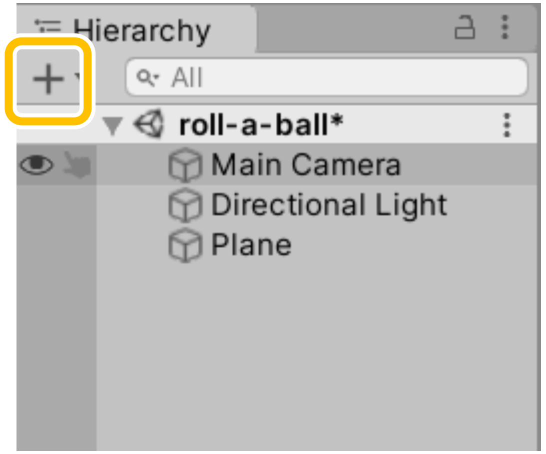
rename New Scene

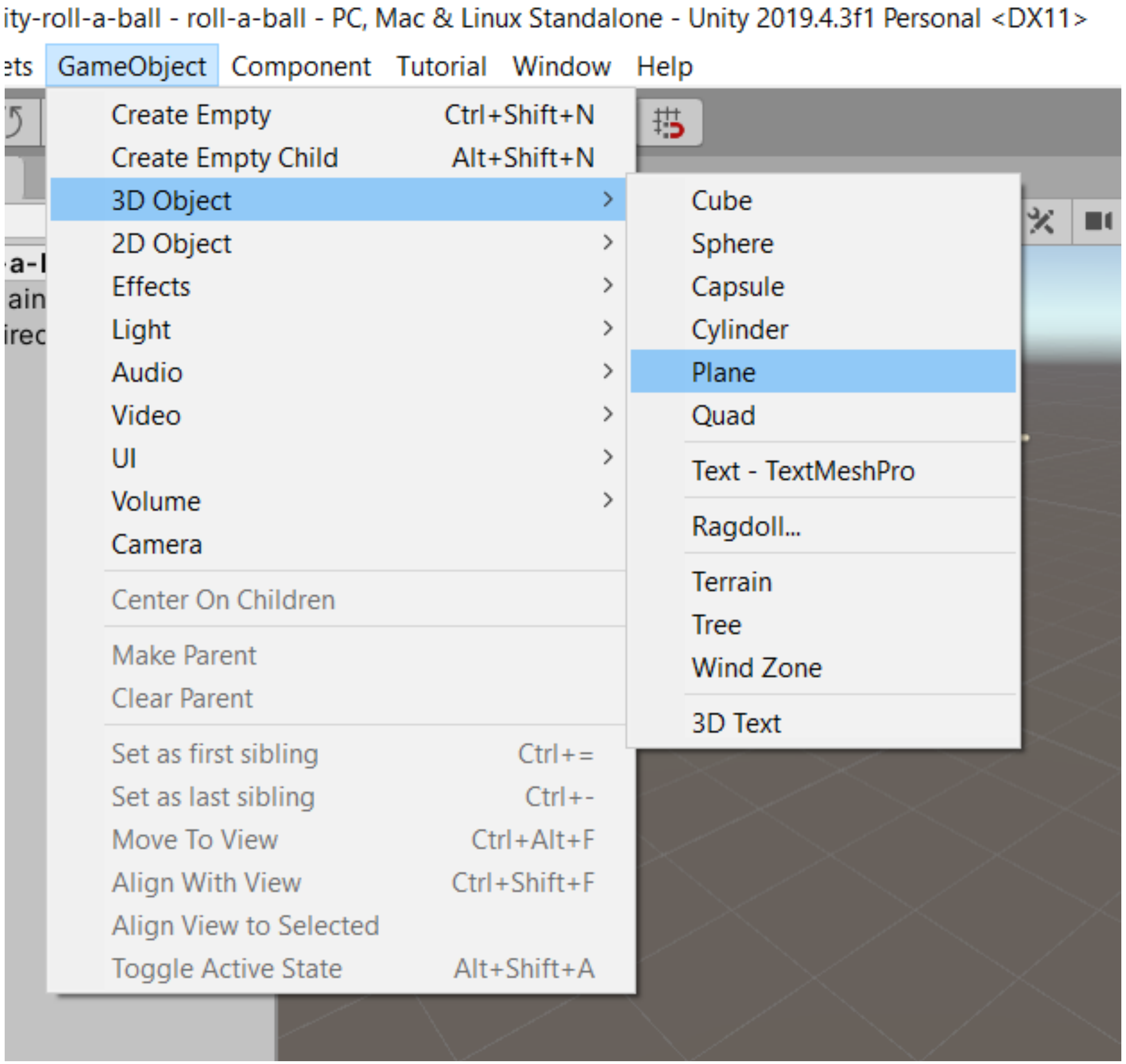
- right click the Scene you created in the Project panel
- select the Rename

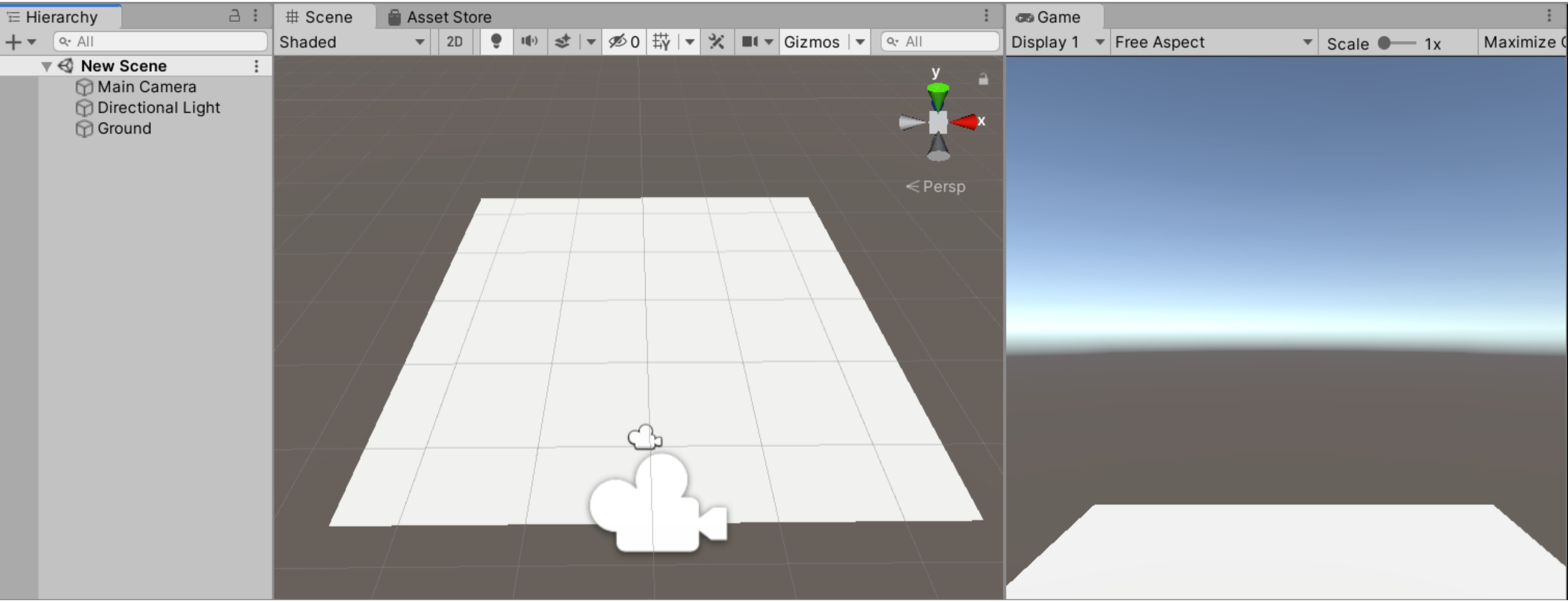


create a ground

- GameObject > 3D Object > Plane



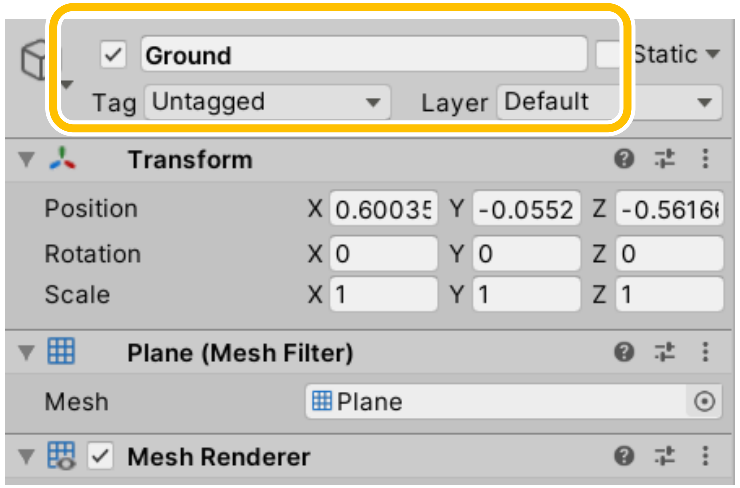



Hierarchy
All
New Scene
Main Camera
Directional Light
Ground
Scene
Asset Store
Shaded
2D
0
Gizmos
All
y
x
< Persp
Game
Display 1
Free Aspect
Scale
1x
Maximize

rename the Plane

- right click it in the hierarchy /

  in the inspector



☑ Ground

Static

Tag Untagged

Layer Default

Transform

Position X 0.60035 Y -0.0552 Z -0.56166

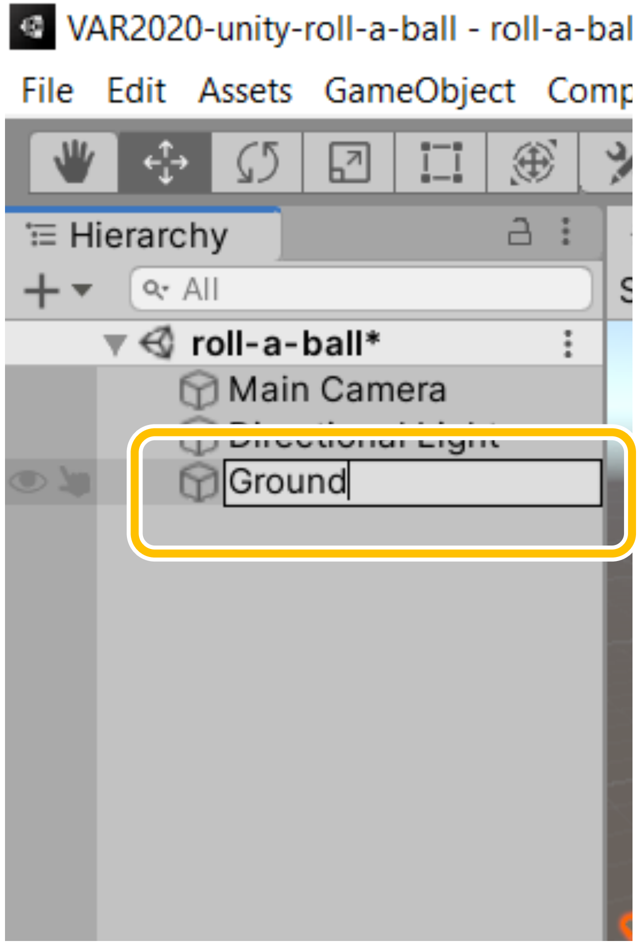
Rotation X 0 Y 0 Z 0

Scale X 1 Y 1 Z 1

Plane (Mesh Filter)

Mesh Plane

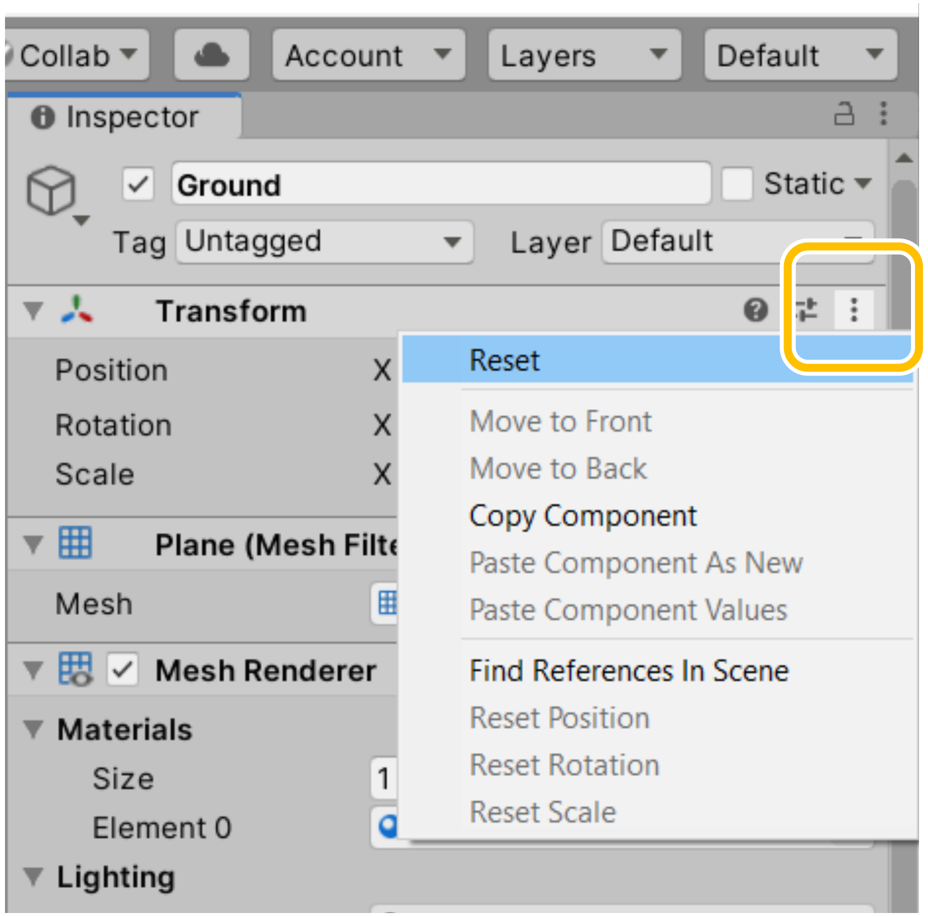
☑ Mesh Renderer

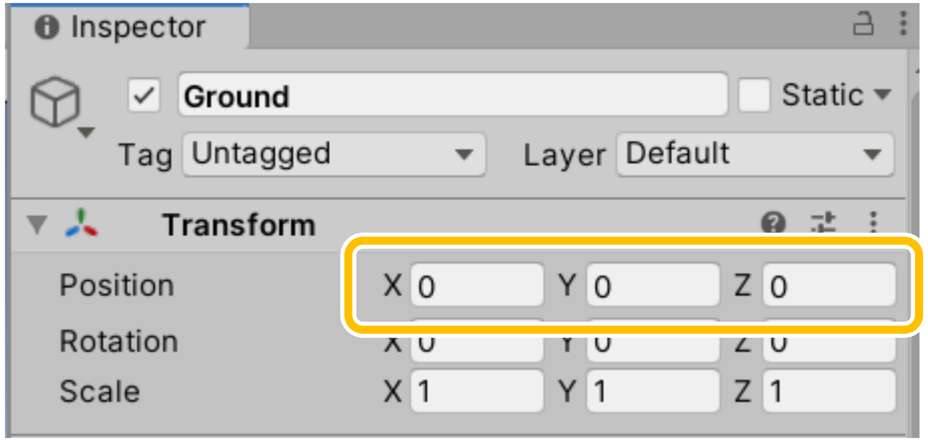


Ground

set Ground at (0,0,0)

- Inspector > Transform




Inspector

Ground ☑ □ Static

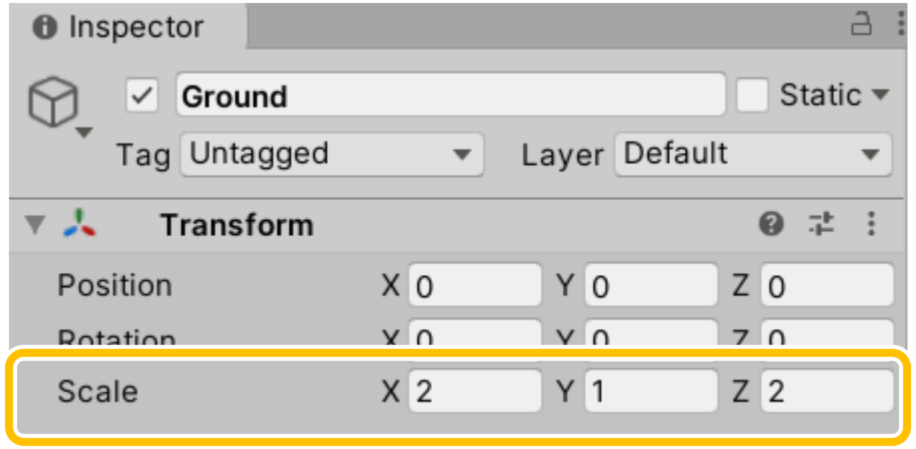
Tag Untagged Layer Default

Transform

Position	X 0	Y 0	Z 0
Rotation	X 0	Y 0	Z 0
Scale	X 1	Y 1	Z 1

Change scale to (2,1,2)

- Use inspector



Inspector

☑ Ground □ Static

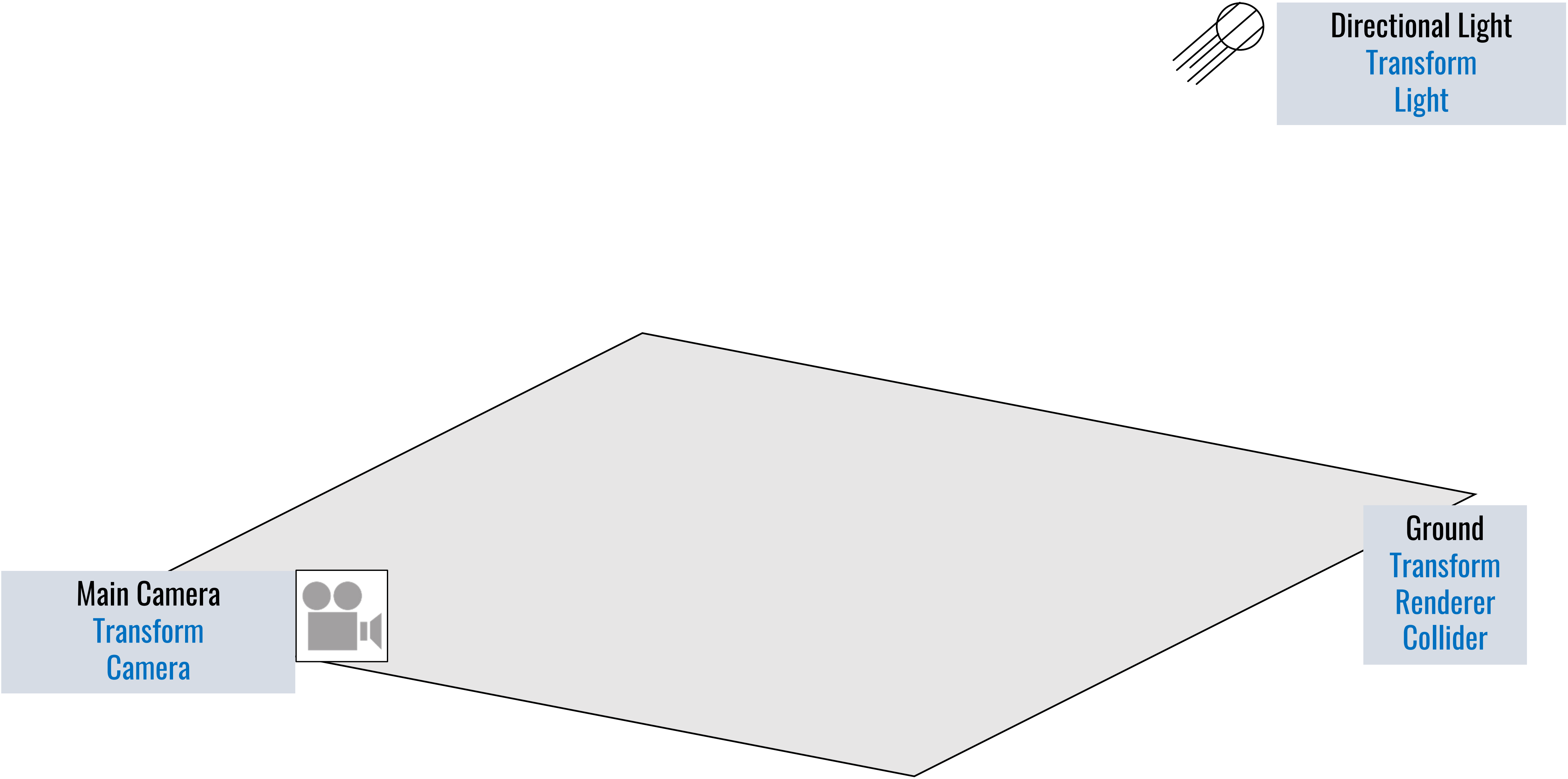
Tag Untagged Layer Default

Transform

Position X 0 Y 0 Z 0

Rotation X 0 Y 0 Z 0

Scale X 2 Y 1 Z 2



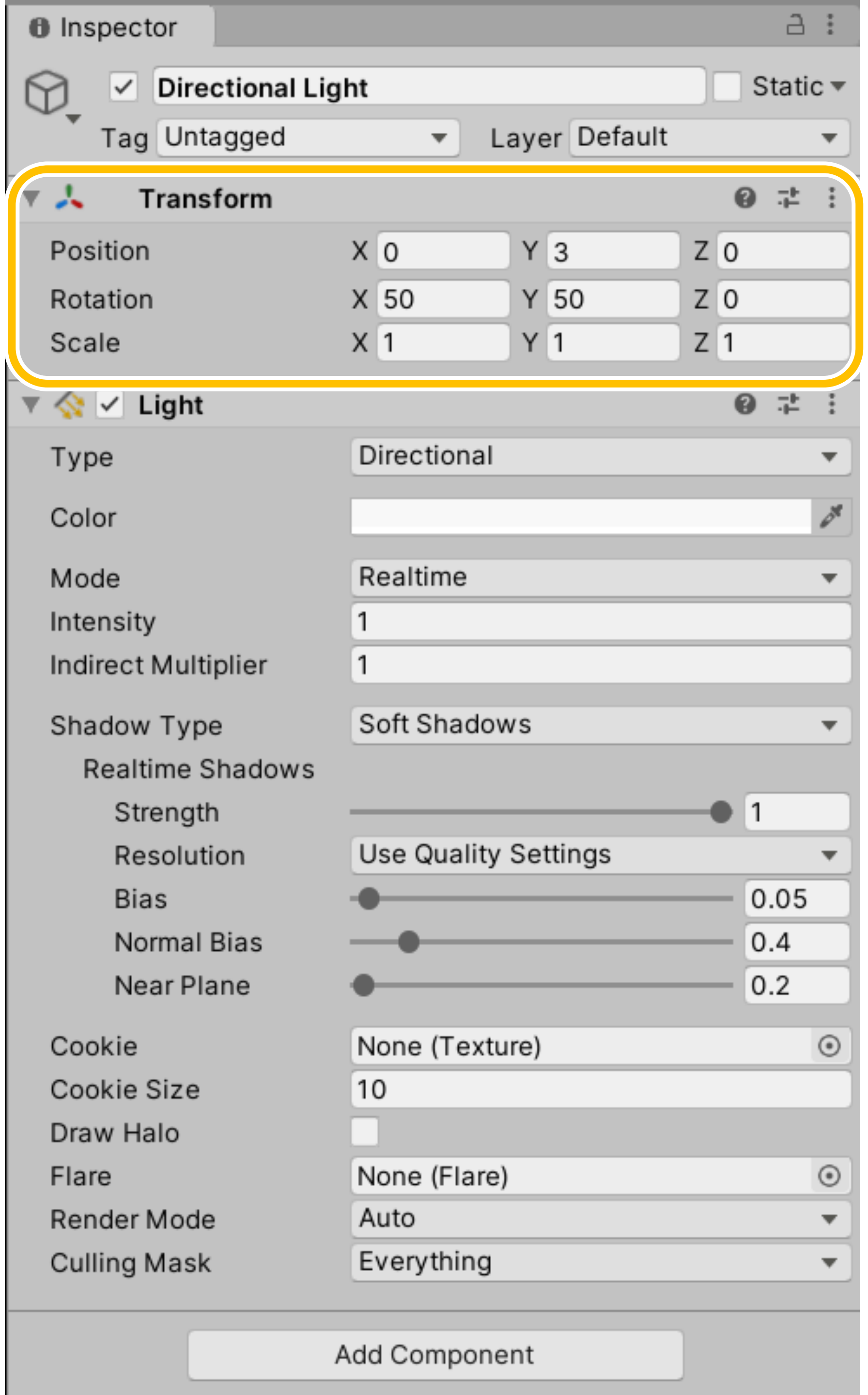
Directional Light
Transform
Light

Main Camera
Transform
Camera

Ground
Transform
Renderer
Collider

In the Directional Light

- Adjust Transform



Inspector

☑ Directional Light □ Static

Tag Untagged Layer Default

Transform

Position X 0 Y 3 Z 0

Rotation X 50 Y 50 Z 0

Scale X 1 Y 1 Z 1

☑ Light

Type Directional

Color

Mode Realtime

Intensity 1

Indirect Multiplier 1

Shadow Type Soft Shadows

Realtime Shadows

Strength 1

Resolution Use Quality Settings

Bias 0.05

Normal Bias 0.4

Near Plane 0.2

Cookie None (Texture)

Cookie Size 10

Draw Halo □

Flare None (Flare)

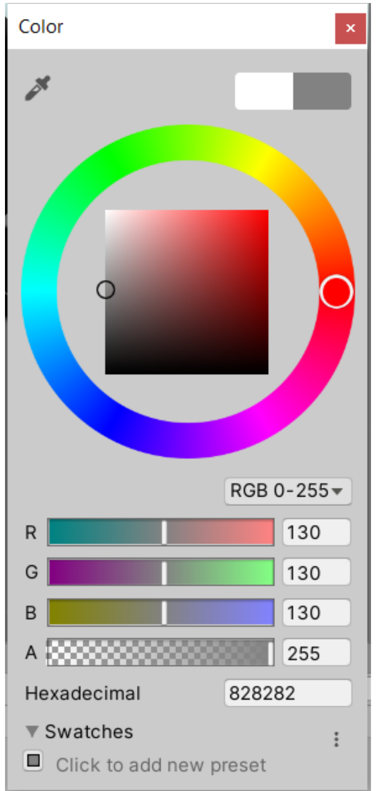
Render Mode Auto

Culling Mask Everything

Add Component

In the Directional Light

- Adjust Transform
- Change the color of light to white



Color

RGB 0-255

R 130

G 130

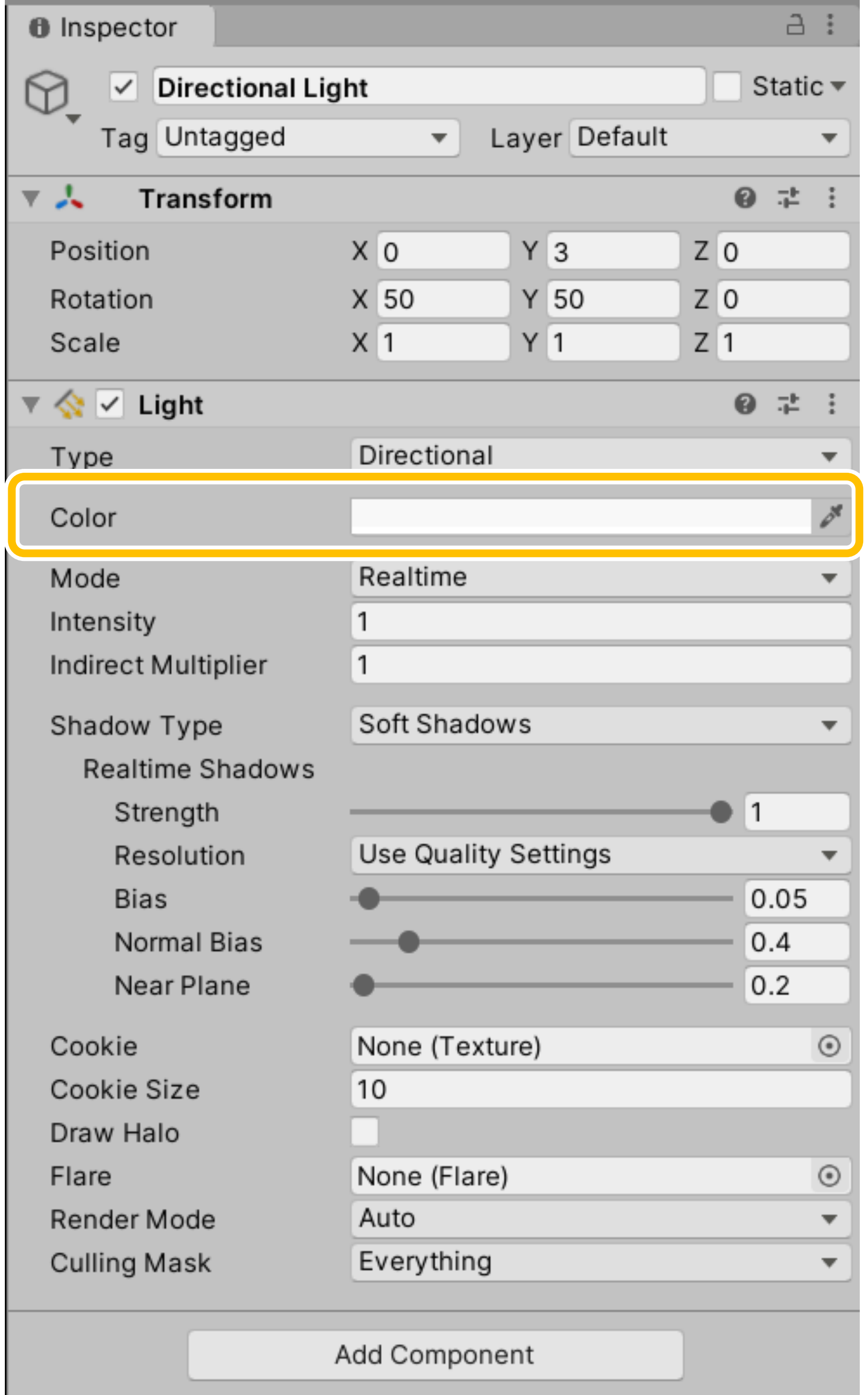
B 130

A 255

Hexadecimal 828282

Swatches

Click to add new preset



Inspector

☑ Directional Light □ Static

Tag Untagged Layer Default

Transform

Position X 0 Y 3 Z 0

Rotation X 50 Y 50 Z 0

Scale X 1 Y 1 Z 1

☑ Light

Type Directional

Color

Mode Realtime

Intensity 1

Indirect Multiplier 1

Shadow Type Soft Shadows

Realtime Shadows

Strength 1

Resolution Use Quality Settings

Bias 0.05

Normal Bias 0.4

Near Plane 0.2

Cookie None (Texture)

Cookie Size 10

Draw Halo □

Flare None (Flare)

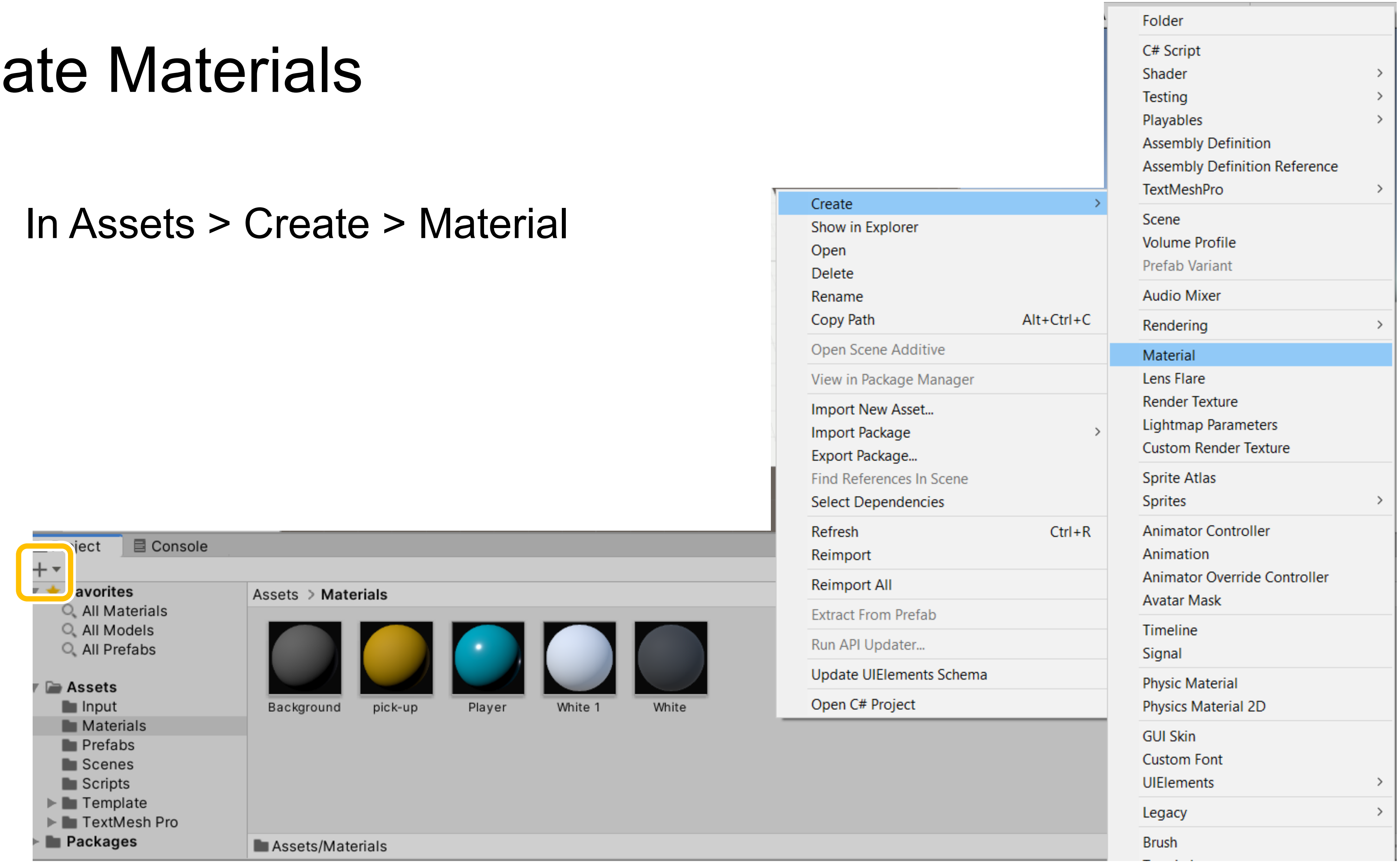
Render Mode Auto

Culling Mask Everything

Add Component

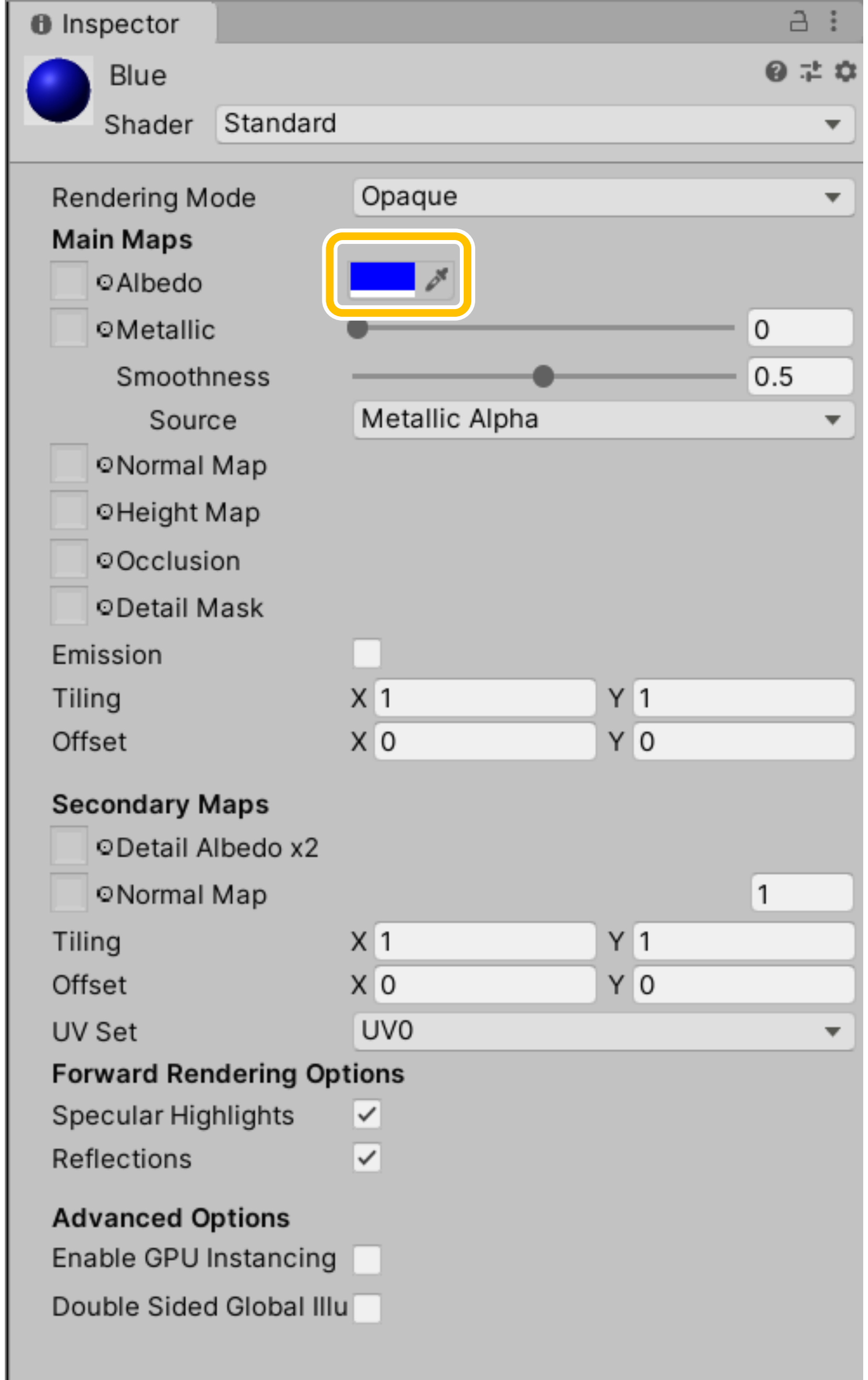
create Materials

- In Assets > Create > Material



change Color

- Select the Material
- Change color in the Inspector



Inspector

Blue

Shader: Standard

Rendering Mode: Opaque

Main Maps

Albedo

Metallic: 0

Smoothness: 0.5

Source: Metallic Alpha

Normal Map

Height Map

Occlusion

Detail Mask

Emission □

Tiling: X 1 Y 1

Offset: X 0 Y 0

Secondary Maps

Detail Albedo x2

Normal Map: 1

Tiling: X 1 Y 1

Offset: X 0 Y 0

UV Set: UV0

Forward Rendering Options

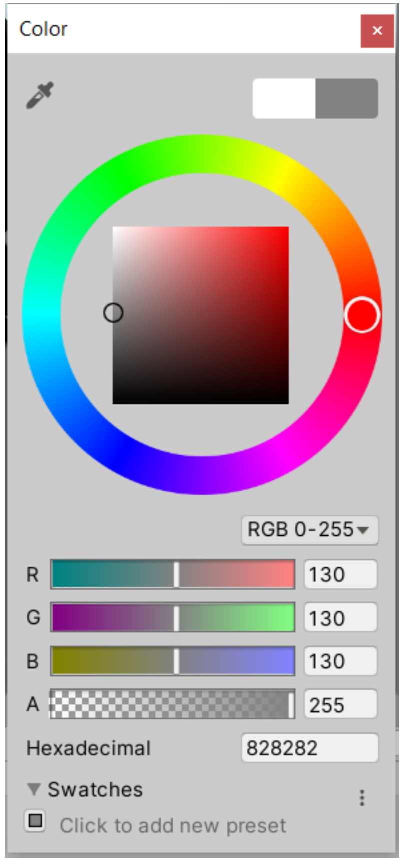
Specular Highlights ☑

Reflections ☑

Advanced Options

Enable GPU Instancing □

Double Sided Global Illu □



Color

RGB 0-255

R: 130

G: 130

B: 130

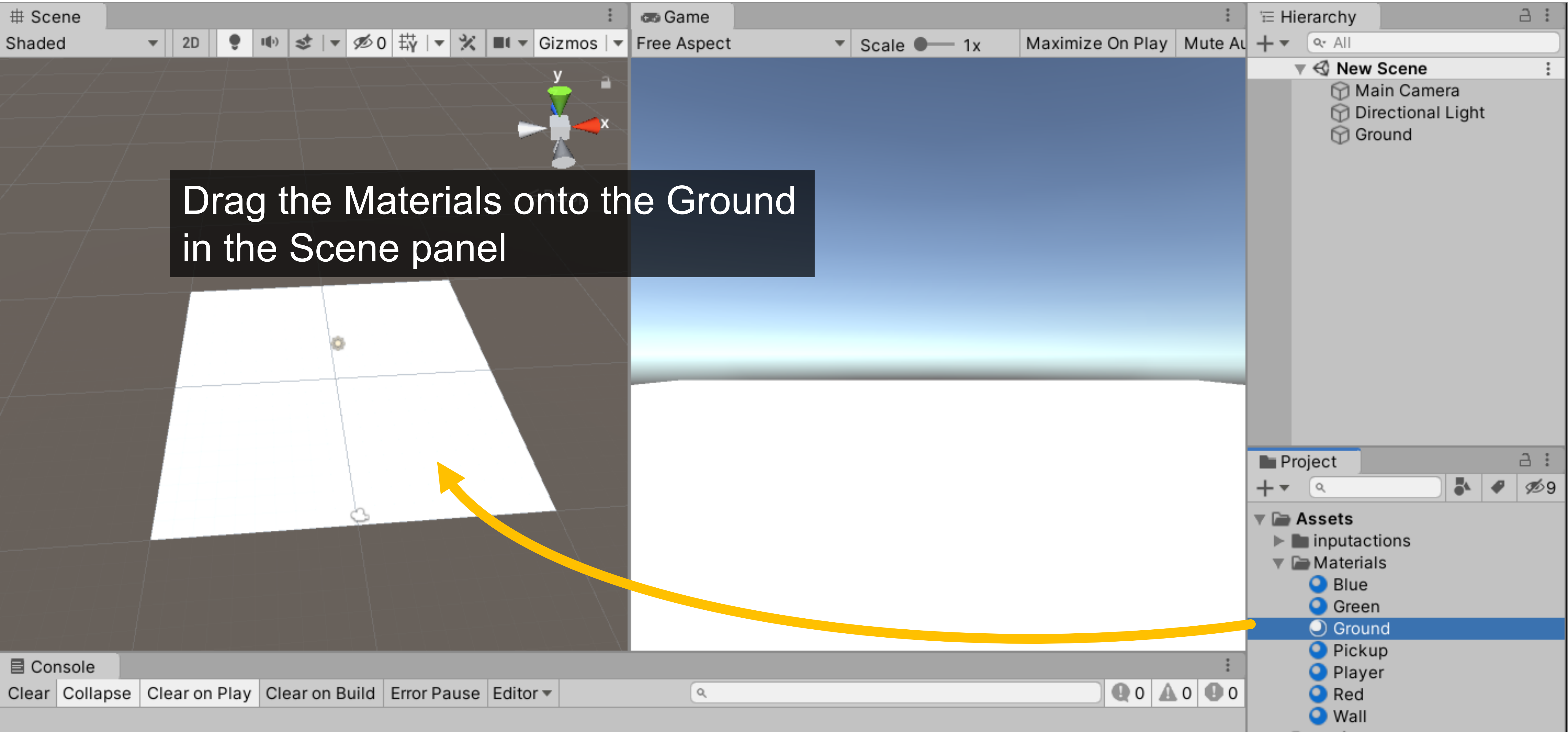
A: 255

Hexadecimal: 828282

Swatches

Click to add new preset

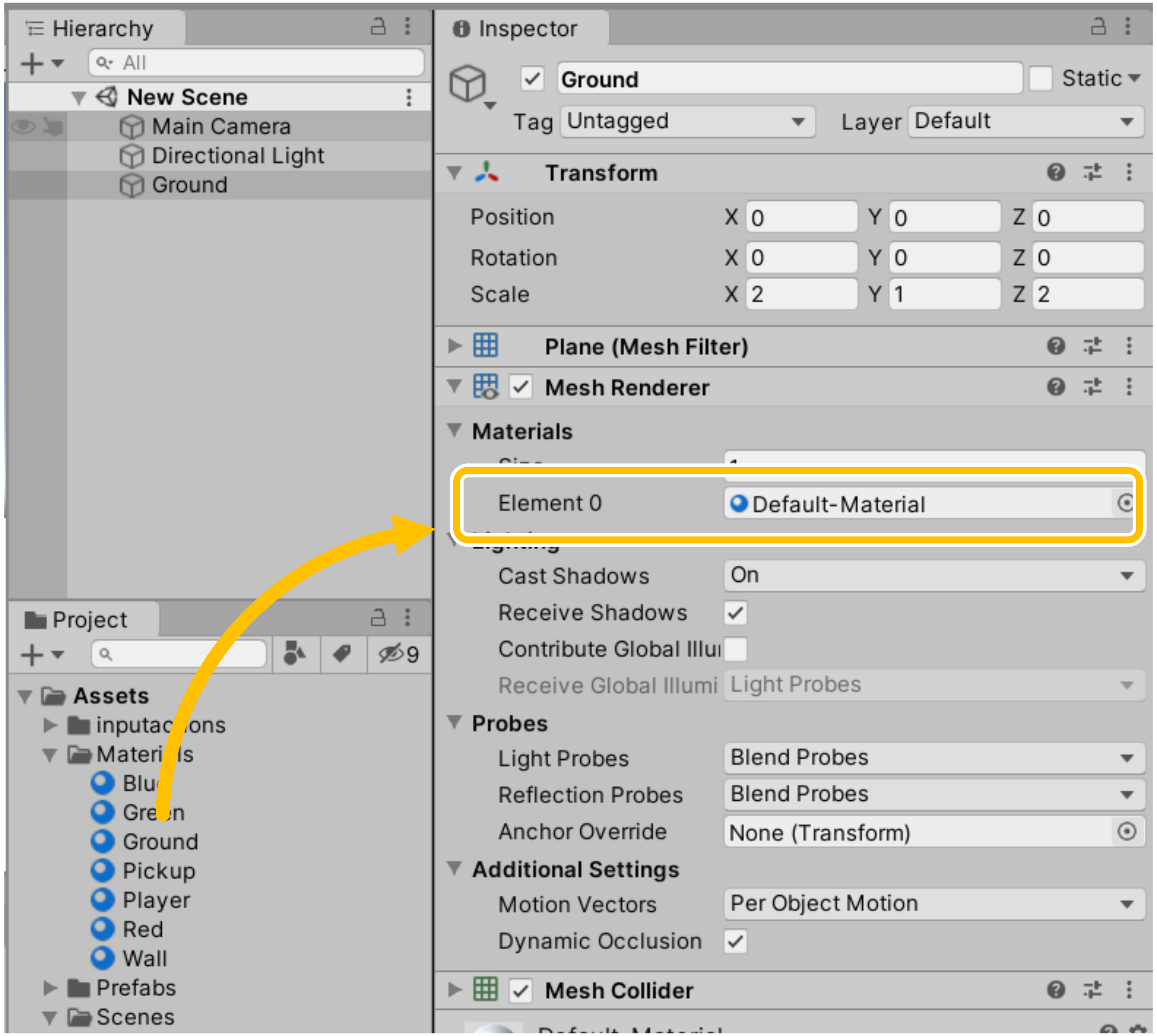
change Color



Drag the Materials onto the Ground in the Scene panel

change Color

- You can also drag into the inspector of the gameobject.



☑ Ground □ Static

Tag Untagged Layer Default

Transform

Position X 0 Y 0 Z 0

Rotation X 0 Y 0 Z 0

Scale X 2 Y 1 Z 2

Plane (Mesh Filter)

☑ Mesh Renderer

Materials

Size 1

Element 0 Default-Material

Lighting

Cast Shadows On

Receive Shadows ☑

Contribute Global Illu □

Receive Global Illumi Light Probes

Probes

Light Probes Blend Probes

Reflection Probes Blend Probes

Anchor Override None (Transform)

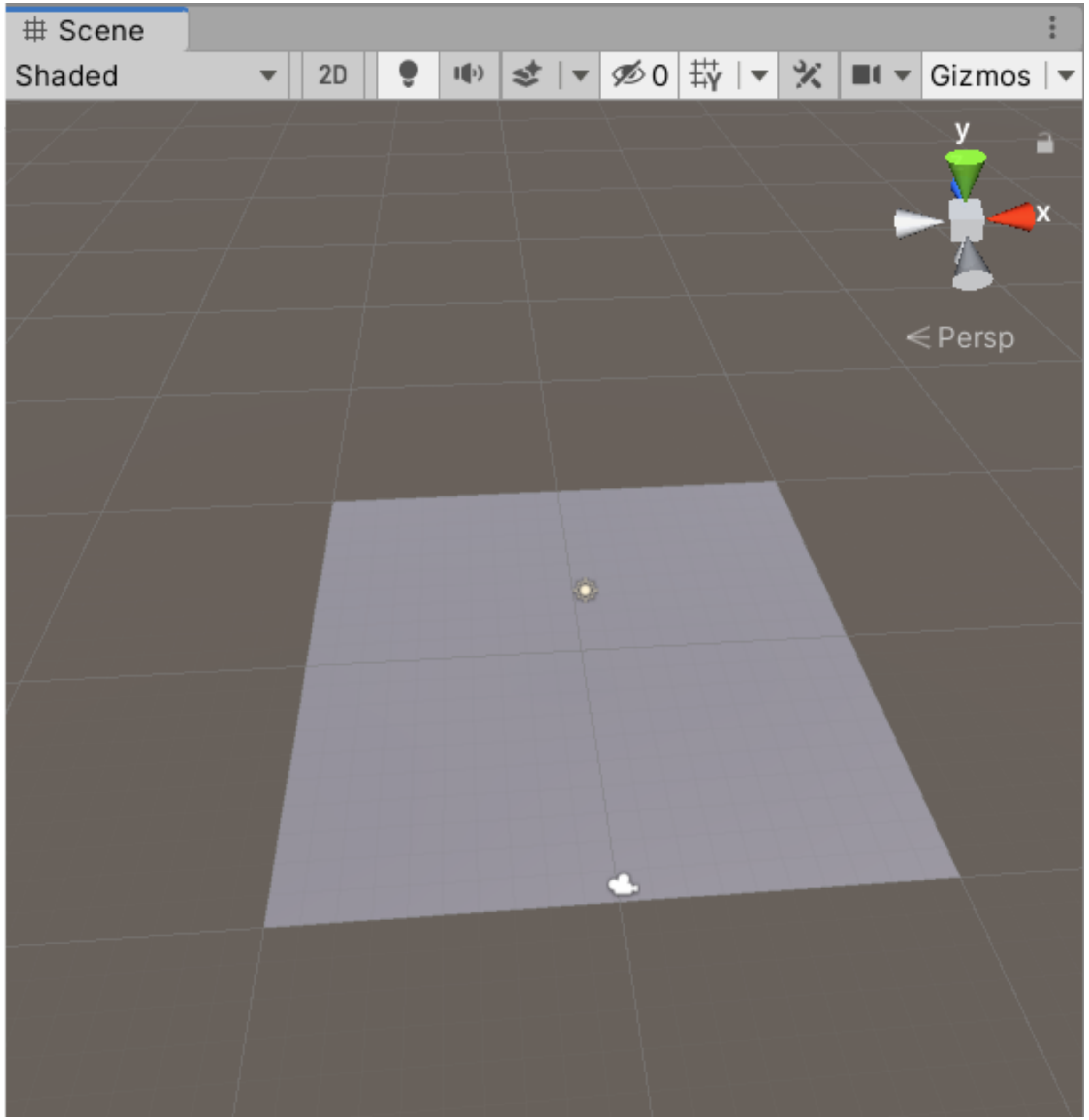
Additional Settings

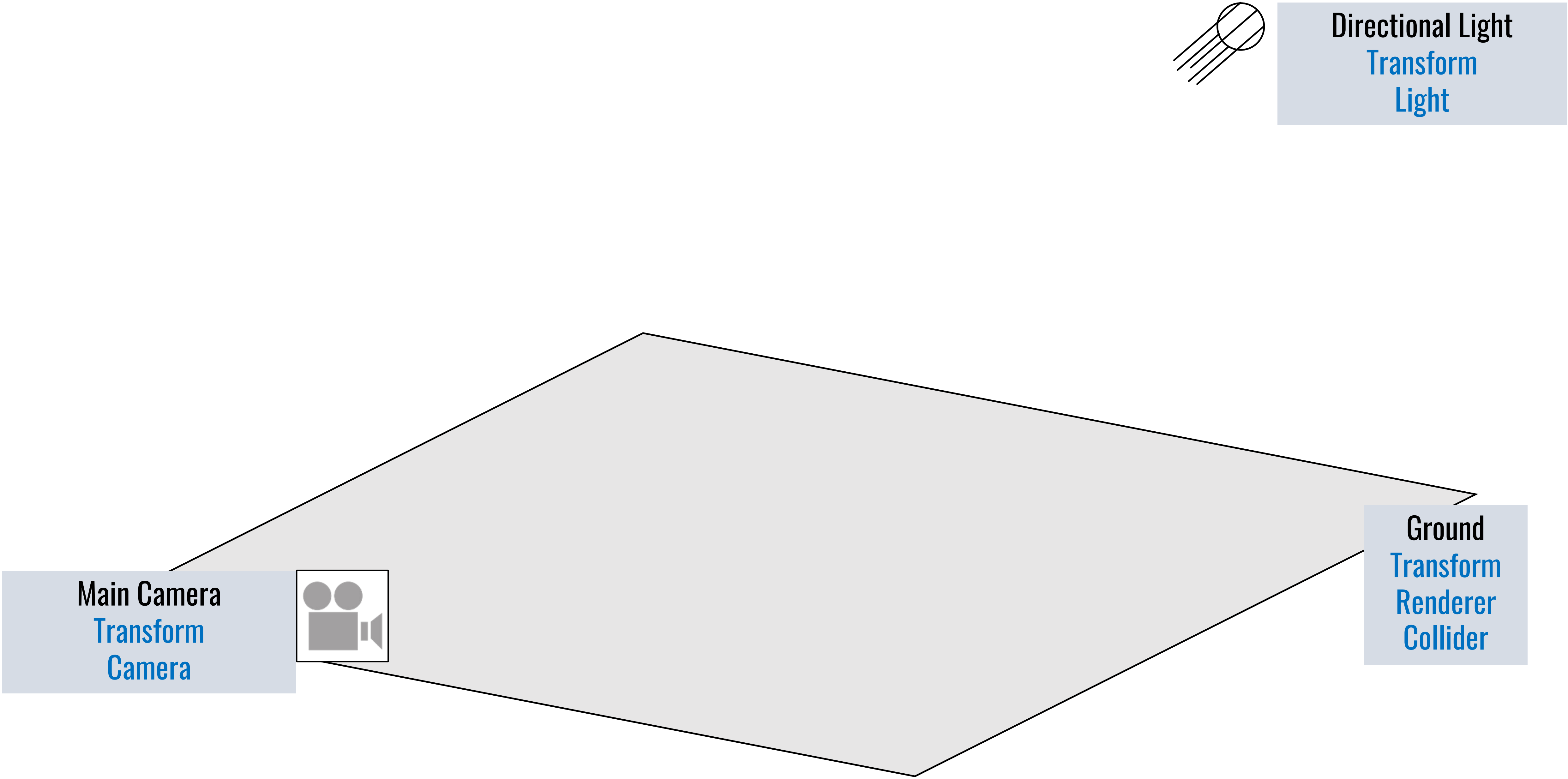
Motion Vectors Per Object Motion

Dynamic Occlusion ☑

☑ Mesh Collider

Default-Material

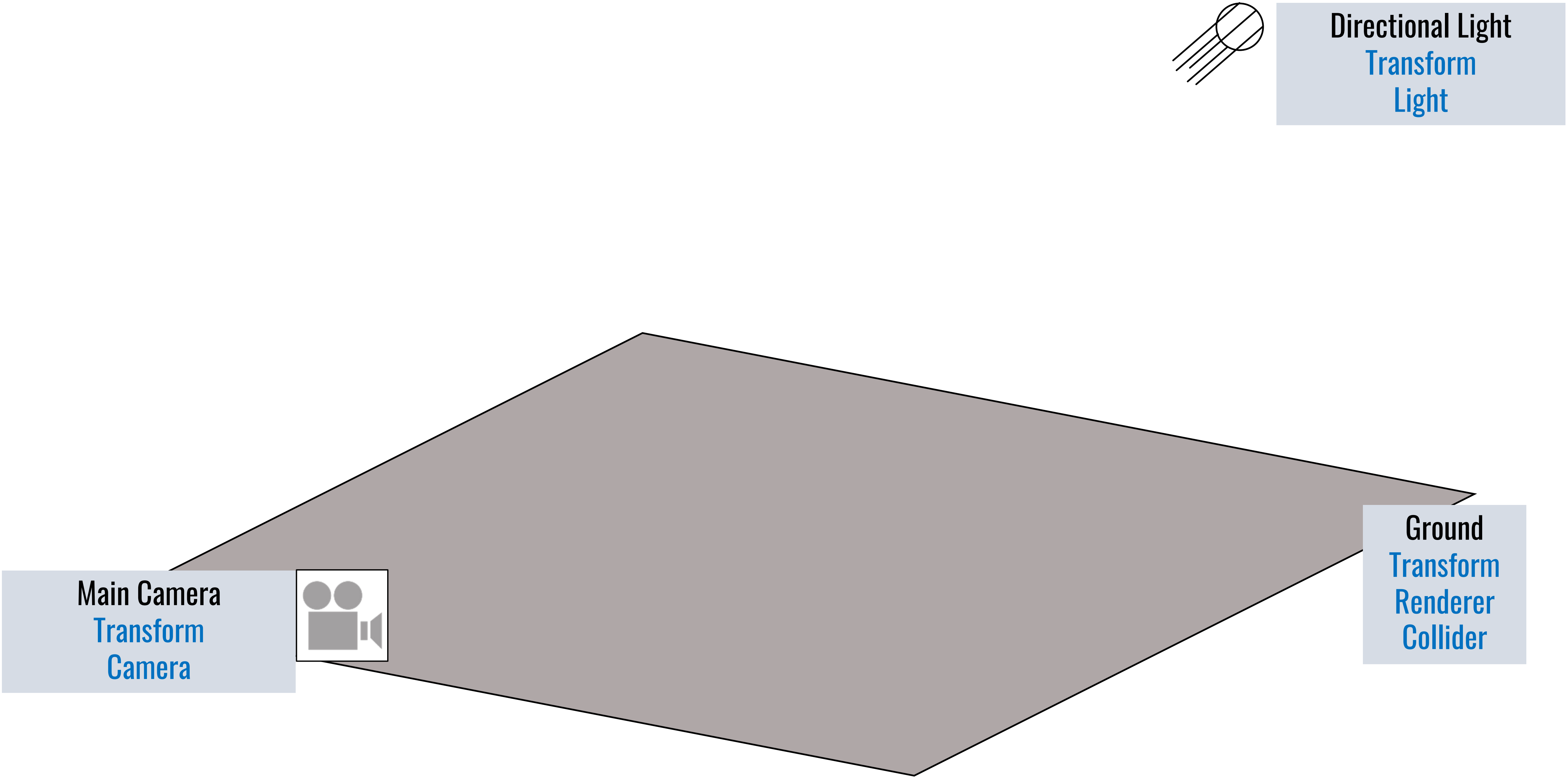

Scene
Shaded
2D
0
Gizmos
y
x
< Persp



Directional Light
Transform
Light

Ground
Transform
Renderer
Collider

Main Camera
Transform
Camera



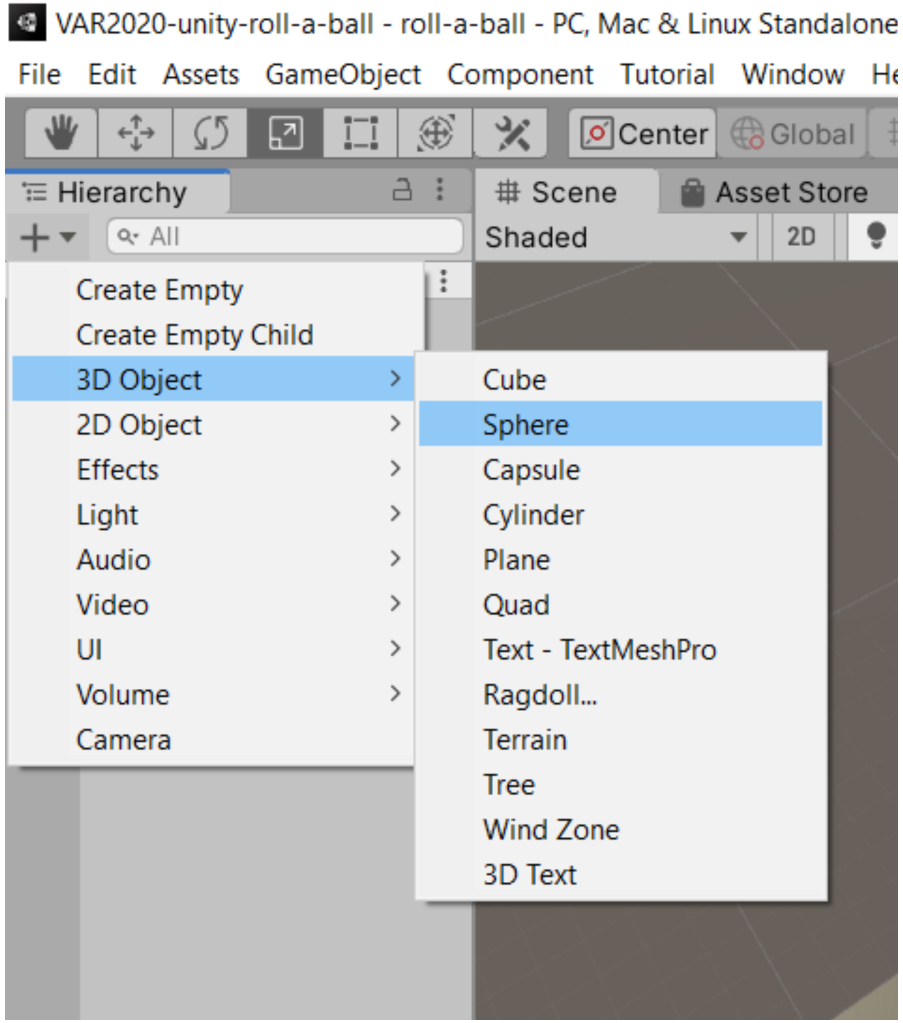
Directional Light
Transform
Light

Main Camera
Transform
Camera

Ground
Transform
Renderer
Collider

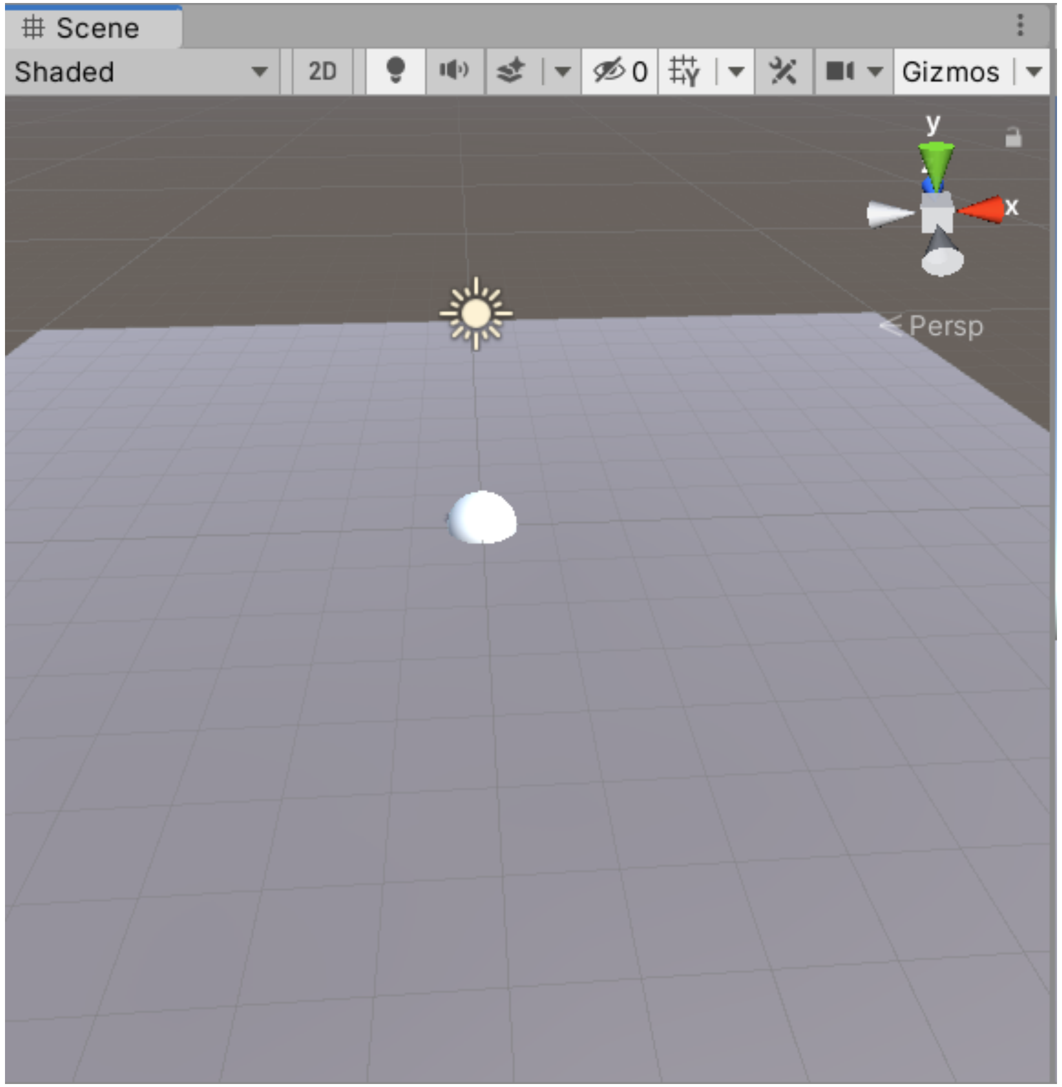
Create a sphere

- Reset to (0, 0, 0)
- Name it as “Player”

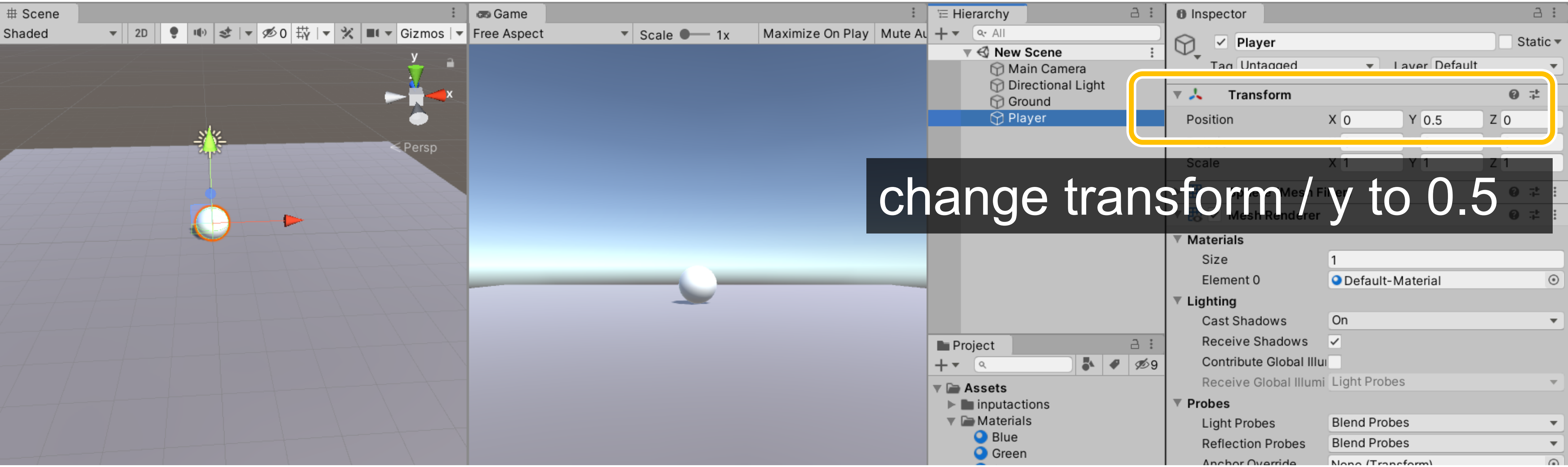


Create a sphere

- The player is in the Plane…



adjust Transform



change transform / y to 0.5

Player

Tag Untagged

Layer Default

Transform

Position X 0 Y 0.5 Z 0

Scale X 1 Y 1 Z 1

Materials

Size 1

Element 0 Default-Material

Lighting

Cast Shadows On

Receive Shadows ☑

Contribute Global Illui □

Receive Global Illumi Light Probes

Probes

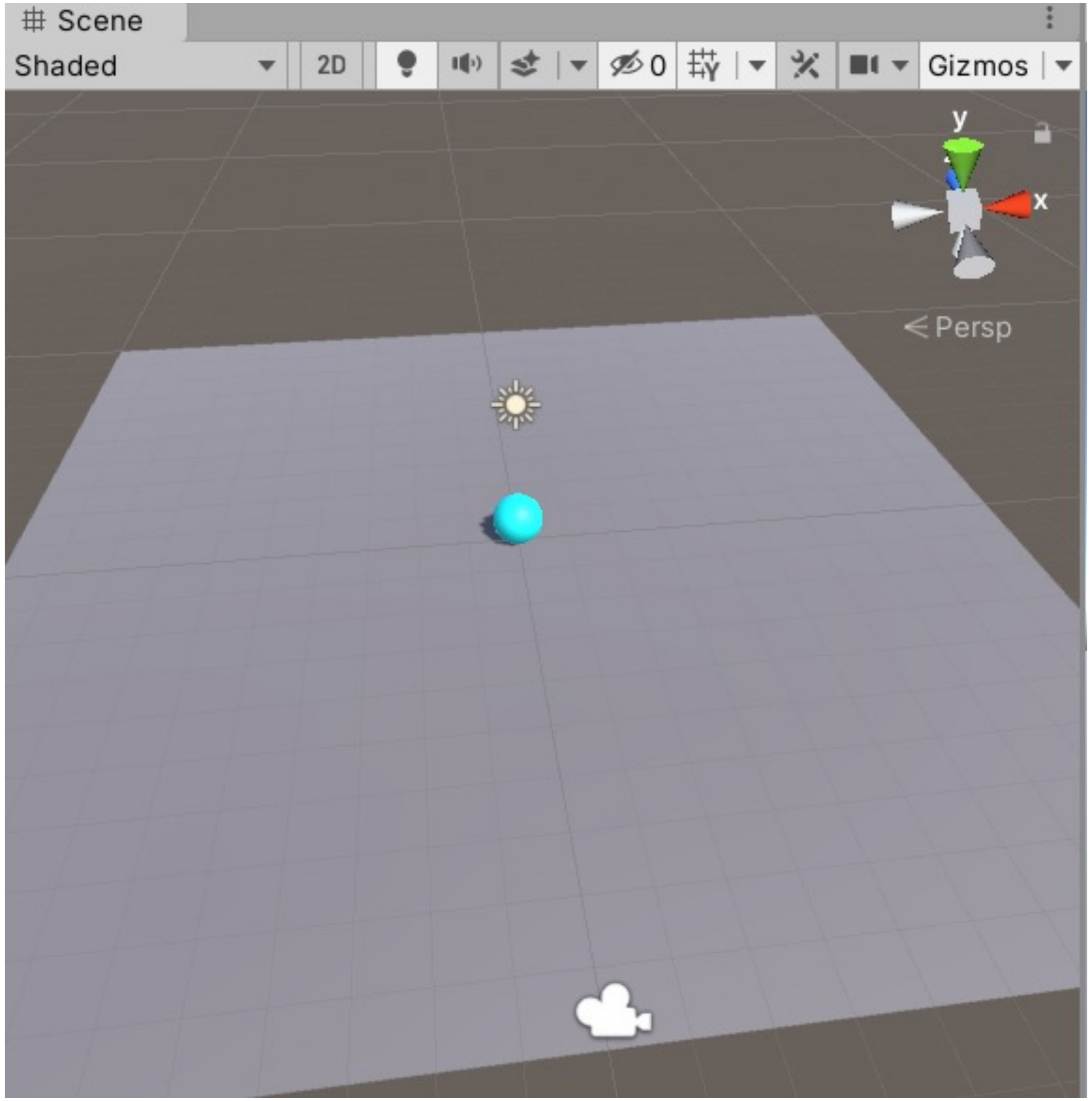
Light Probes Blend Probes

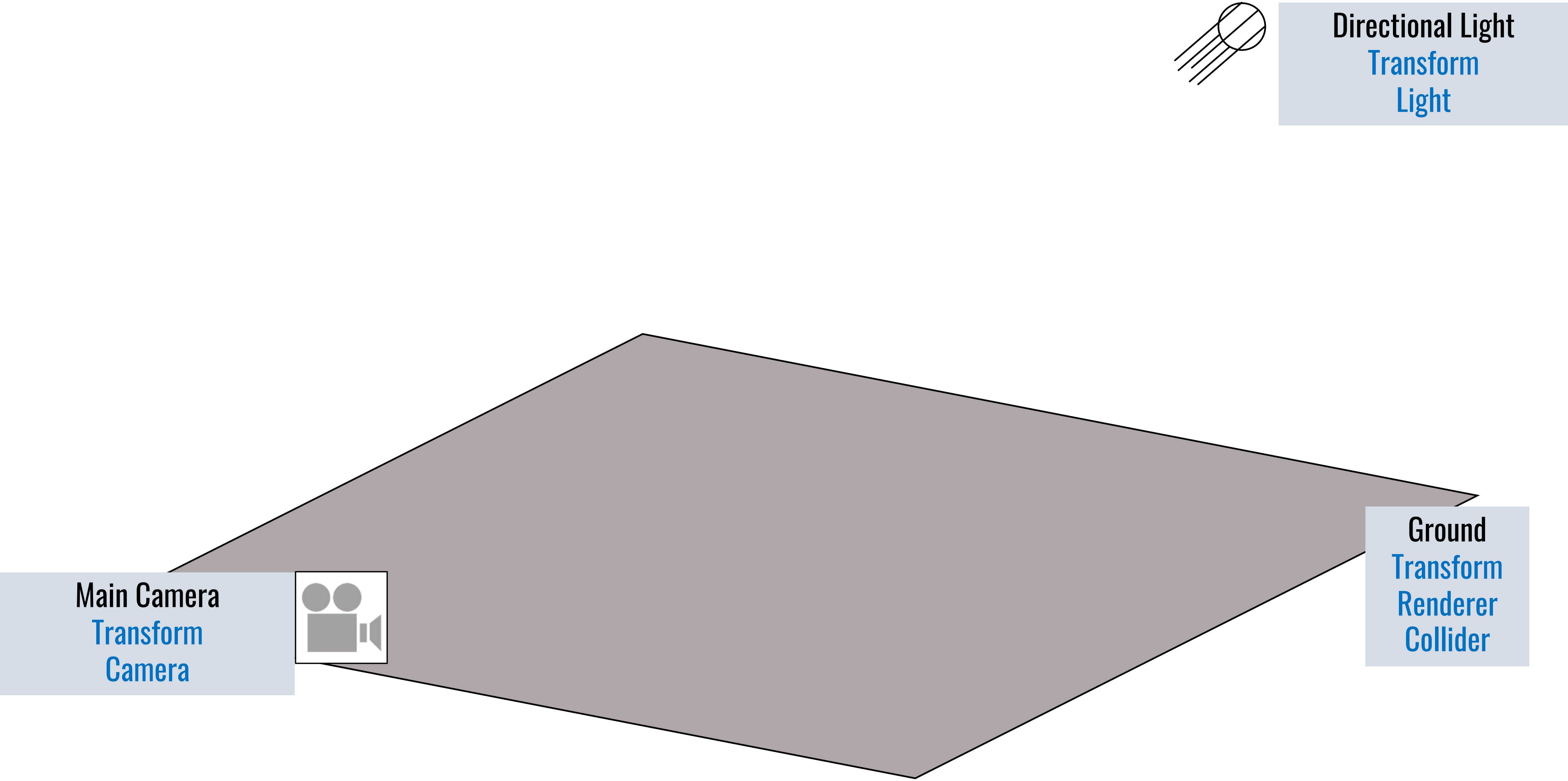
Reflection Probes Blend Probes

Anchor Override None (Transform)

adjust Transform

- Create a new color for Player

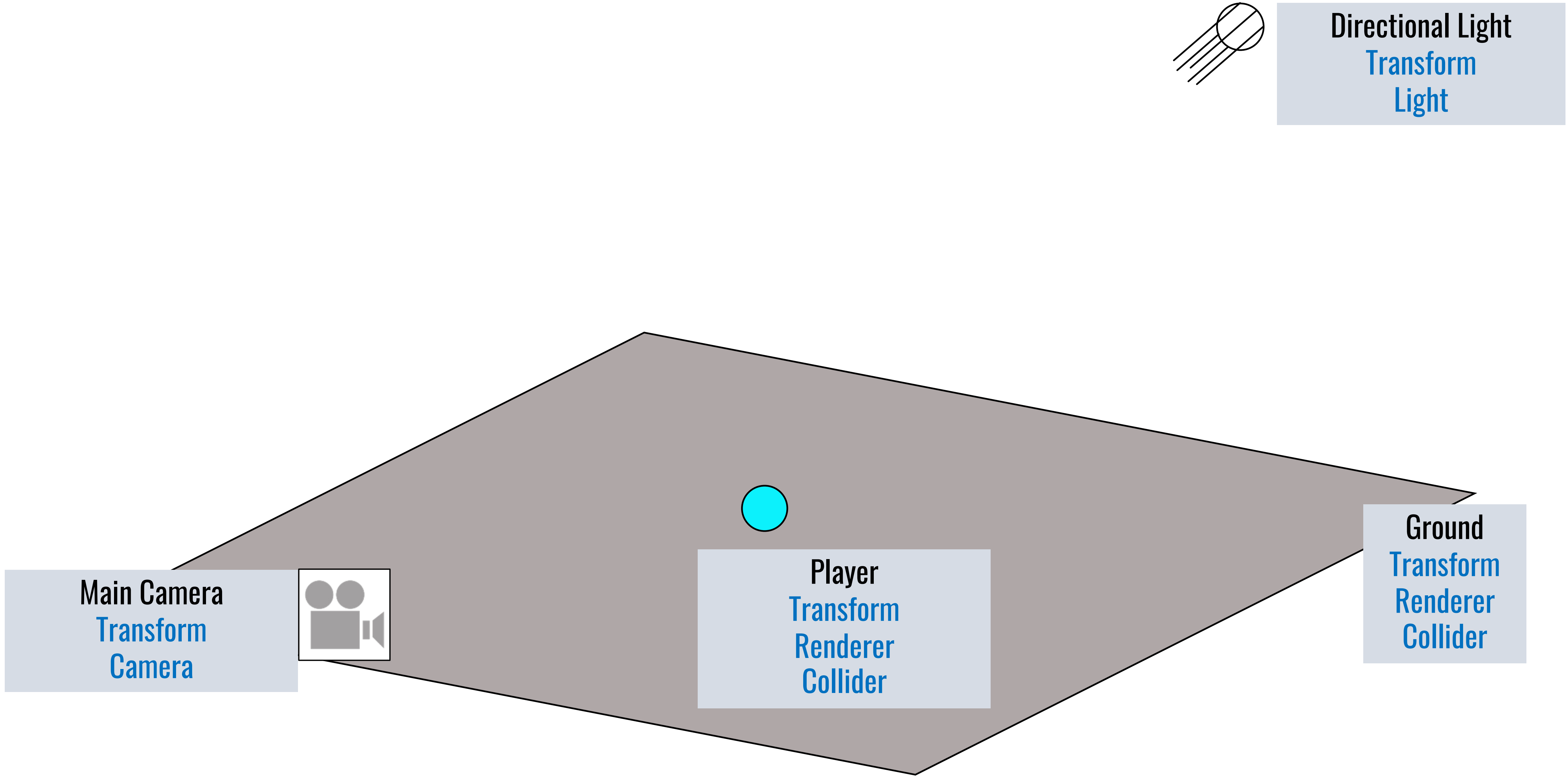




Directional Light
Transform
Light

Main Camera
Transform
Camera

Ground
Transform
Renderer
Collider

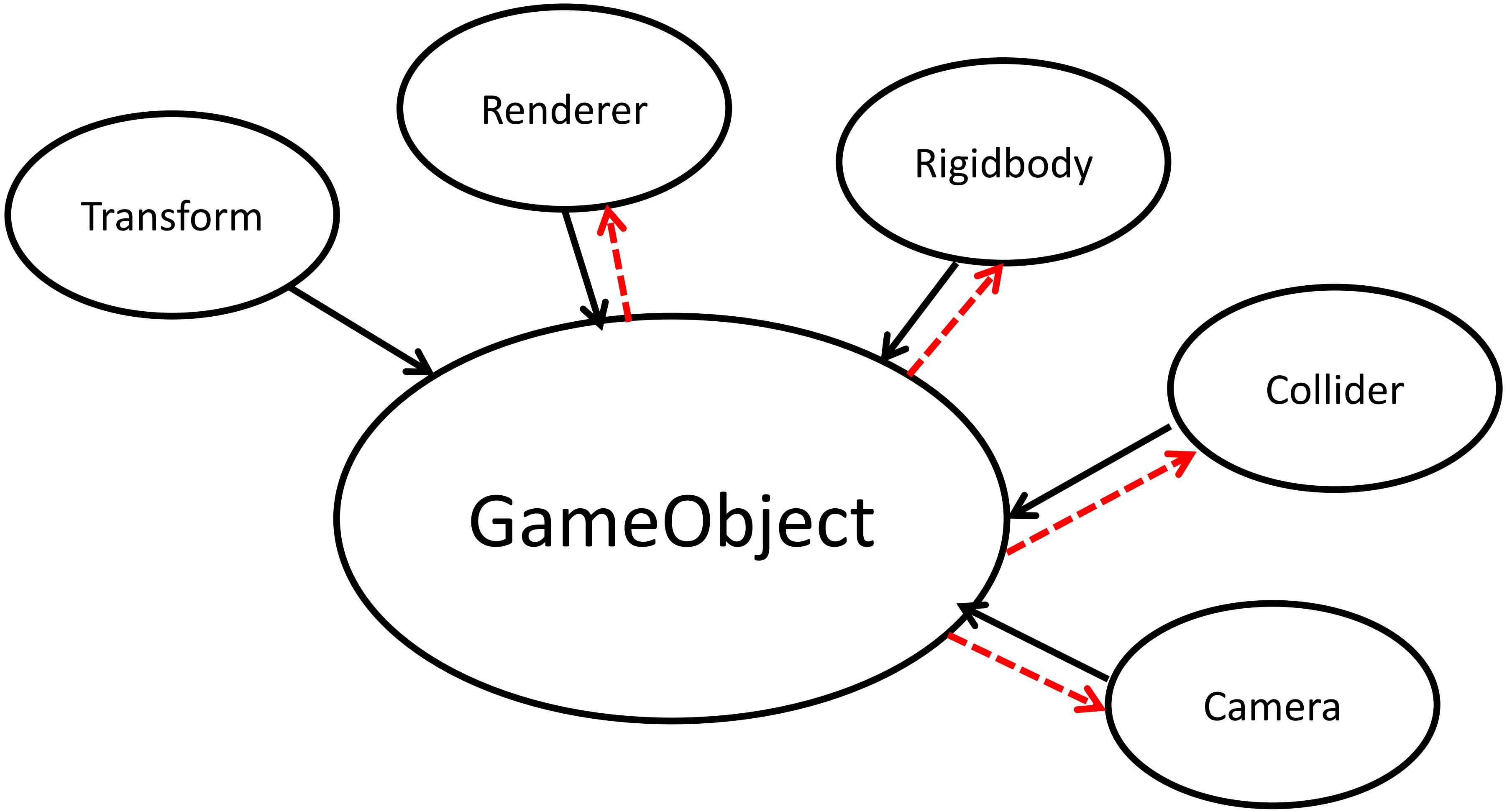


Directional Light
Transform
Light

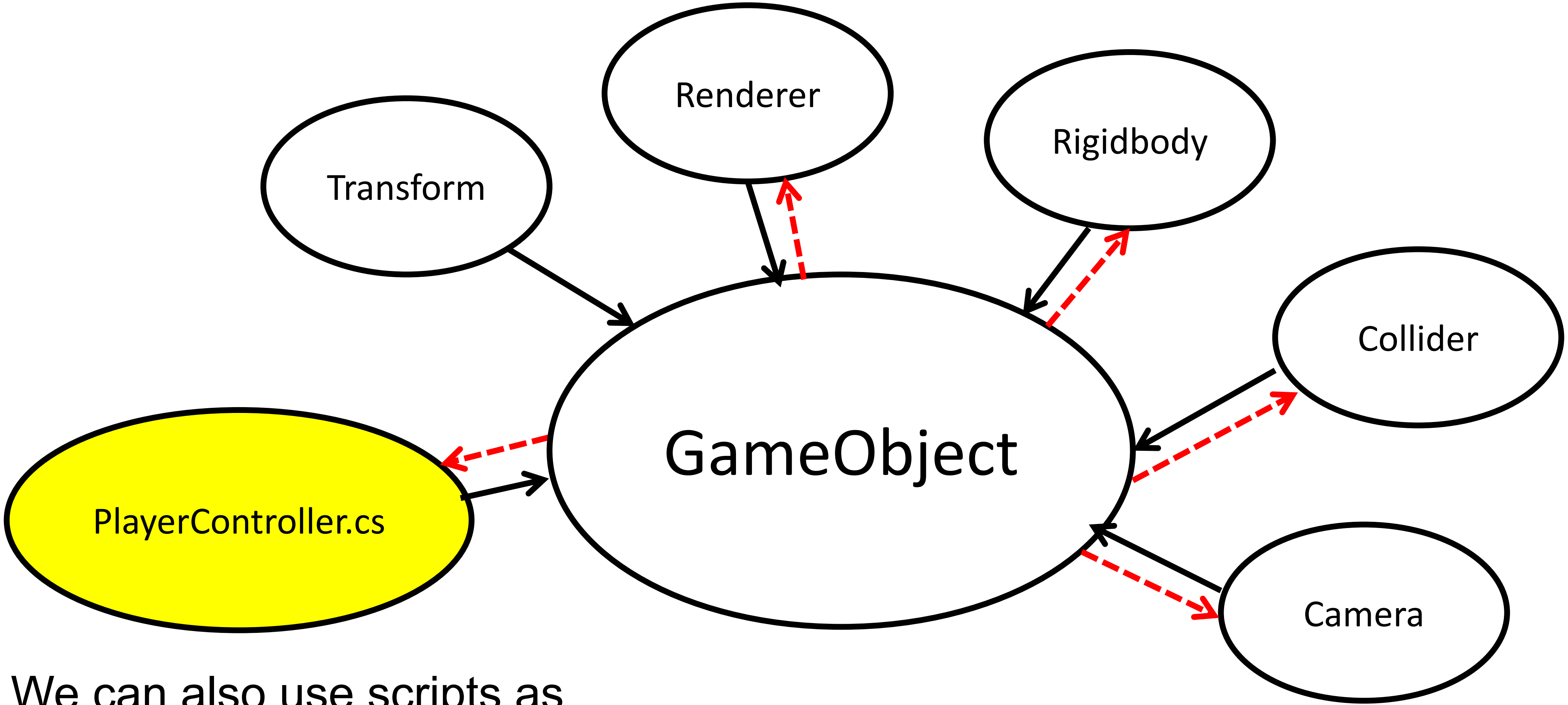
Main Camera
Transform
Camera

Player
Transform
Renderer
Collider

Ground
Transform
Renderer
Collider



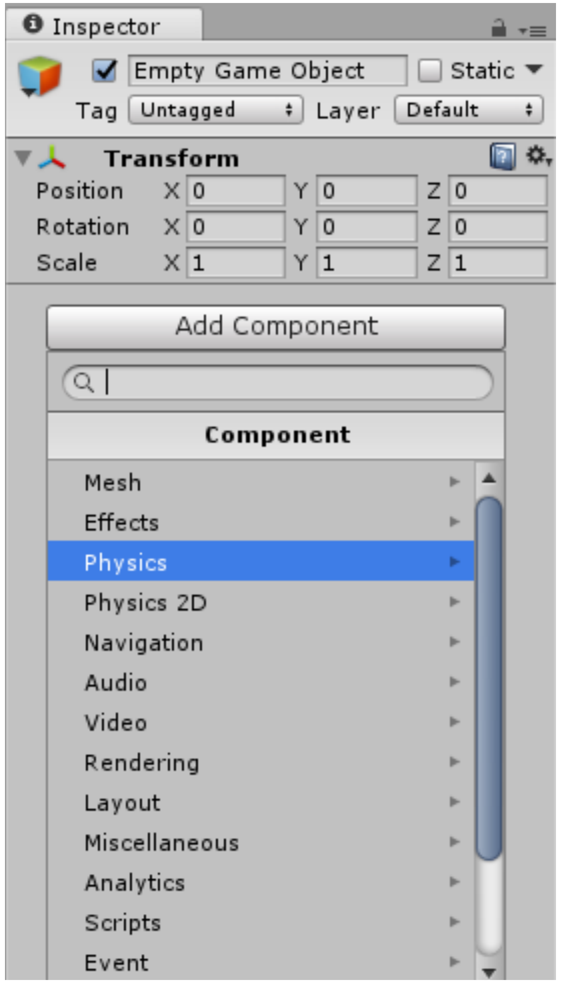
A GameObject has many components.
They all can be attached or removed in Unity or by script.



We can also use scripts as a Component to control the GameObject.

Components

- Add Component:



Inspector

☑ Empty Game Object □ Static

Tag Untagged Layer Default

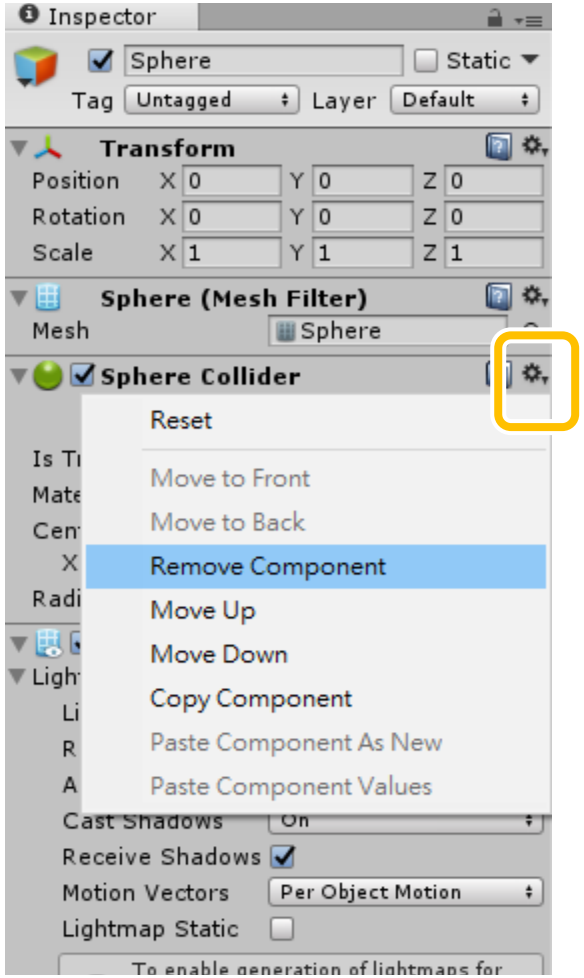
Transform

Position X 0 Y 0 Z 0

Rotation X 0 Y 0 Z 0

Scale X 1 Y 1 Z 1

- Remove Component:



Inspector

☑ Sphere □ Static

Tag Untagged Layer Default

Transform

Position X 0 Y 0 Z 0

Rotation X 0 Y 0 Z 0

Scale X 1 Y 1 Z 1

Sphere (Mesh Filter)

Mesh Sphere

☑ Sphere Collider

Cast Shadows On

Receive Shadows ☑

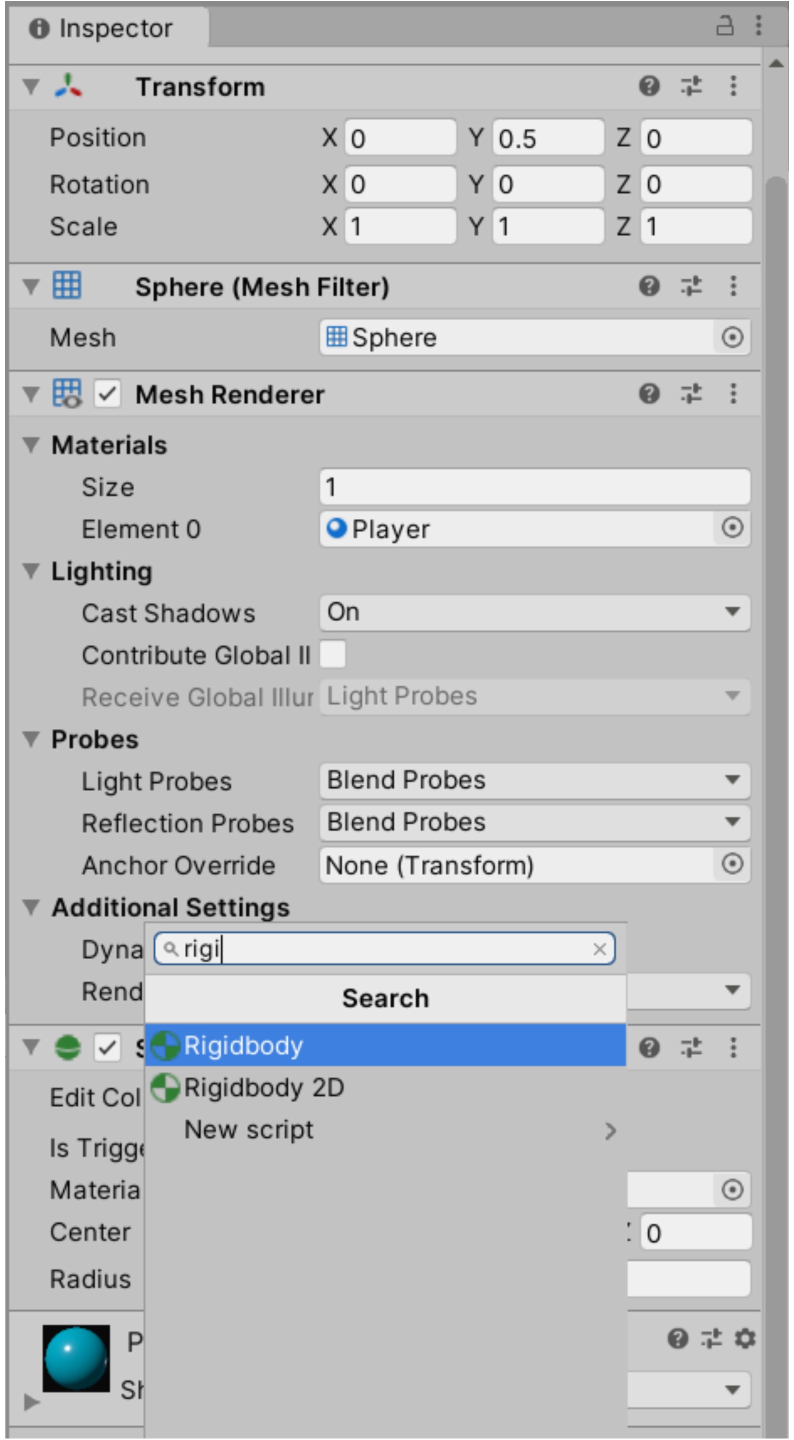
Motion Vectors Per Object Motion

Lightmap Static □

To enable generation of lightmaps for

Add Component

- Select Rigidbody:
  - physics
  - detecting collision or trigger



Inspector

Transform

Position X 0 Y 0.5 Z 0

Rotation X 0 Y 0 Z 0

Scale X 1 Y 1 Z 1

Sphere (Mesh Filter)

Mesh: Sphere

☑ Mesh Renderer

Materials

Size: 1

Element 0: Player

Lighting

Cast Shadows: On

Contribute Global Il: □

Receive Global Illur: Light Probes

Probes

Light Probes: Blend Probes

Reflection Probes: Blend Probes

Anchor Override: None (Transform)

Additional Settings

Dyna

Rend

rigi

Search

Rigidbody

Rigidbody 2D

New script

☑ S

Edit Col

Is Trigge

Materia

Center: 0

Radius

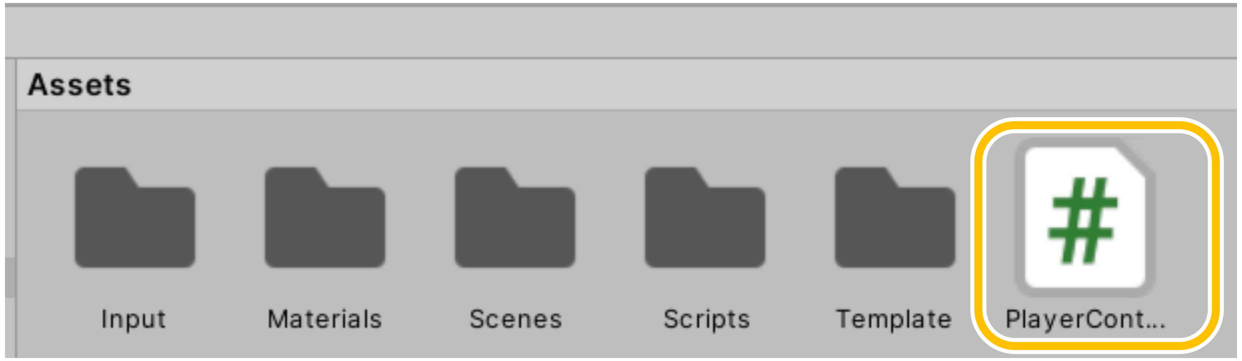
P

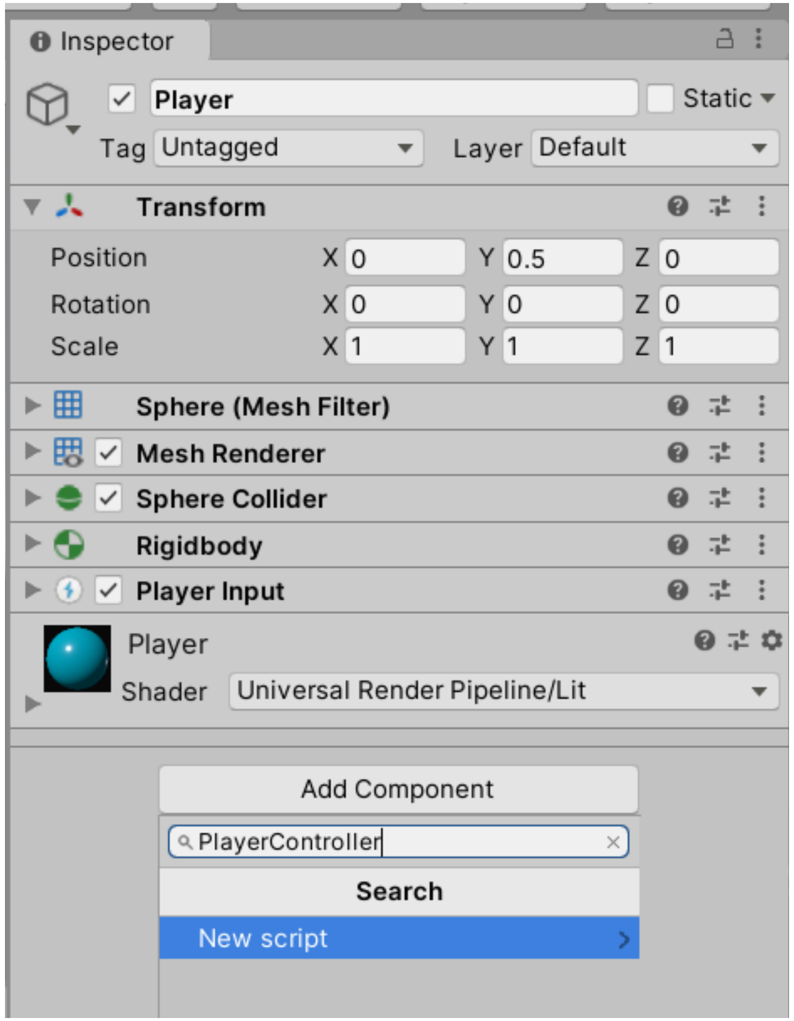
Sh

Create a new script PlayerController.cs

- Add Component > type PlayerController

  > New script





☑ Player □ Static

Tag Untagged Layer Default

Transform

Position	X 0	Y 0.5	Z 0
Rotation	X 0	Y 0	Z 0
Scale	X 1	Y 1	Z 1

Sphere (Mesh Filter)

☑ Mesh Renderer

☑ Sphere Collider

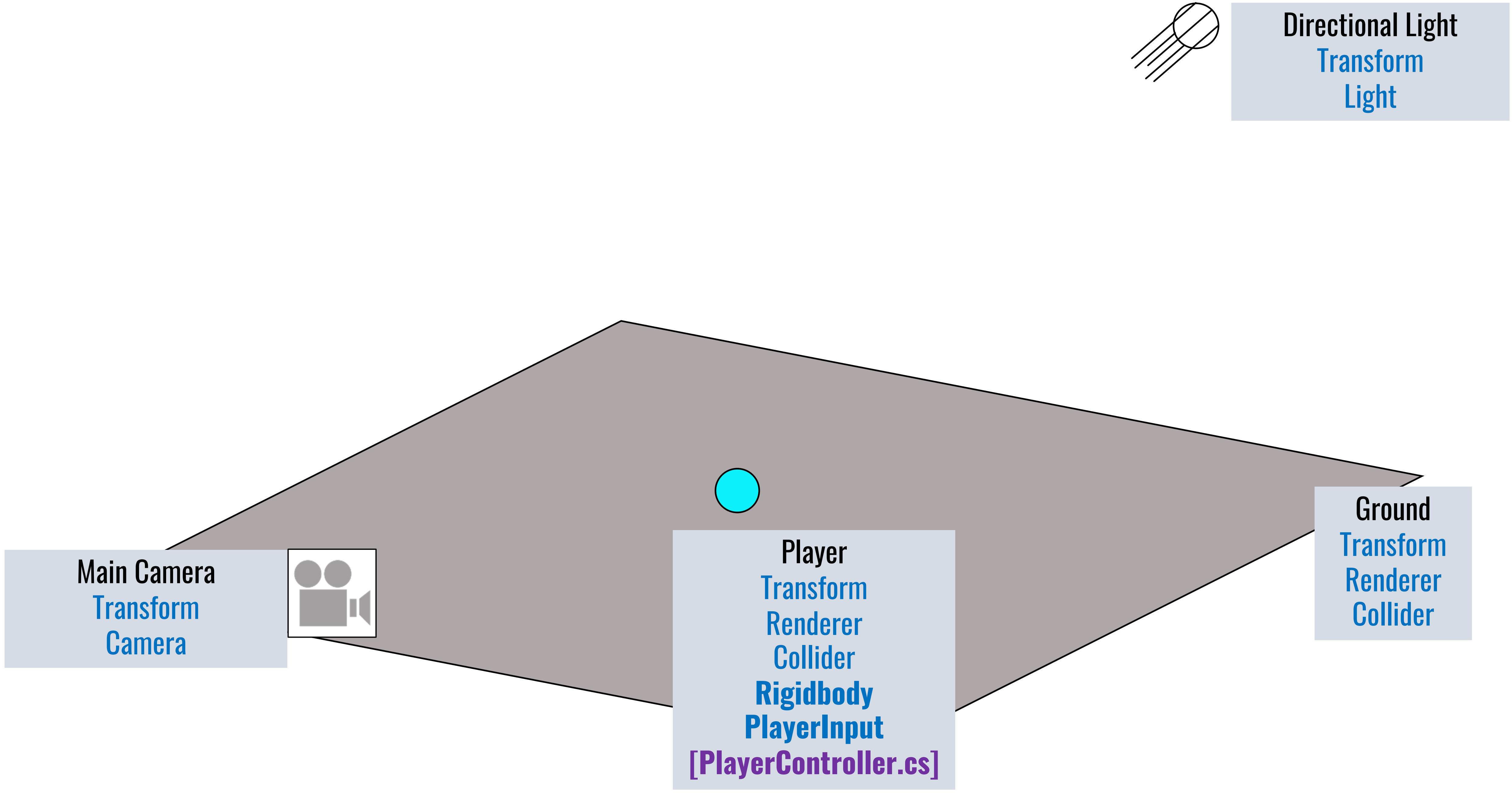
Rigidbody

☑ Player Input

Player

Shader Universal Render Pipeline/Lit

PlayerController



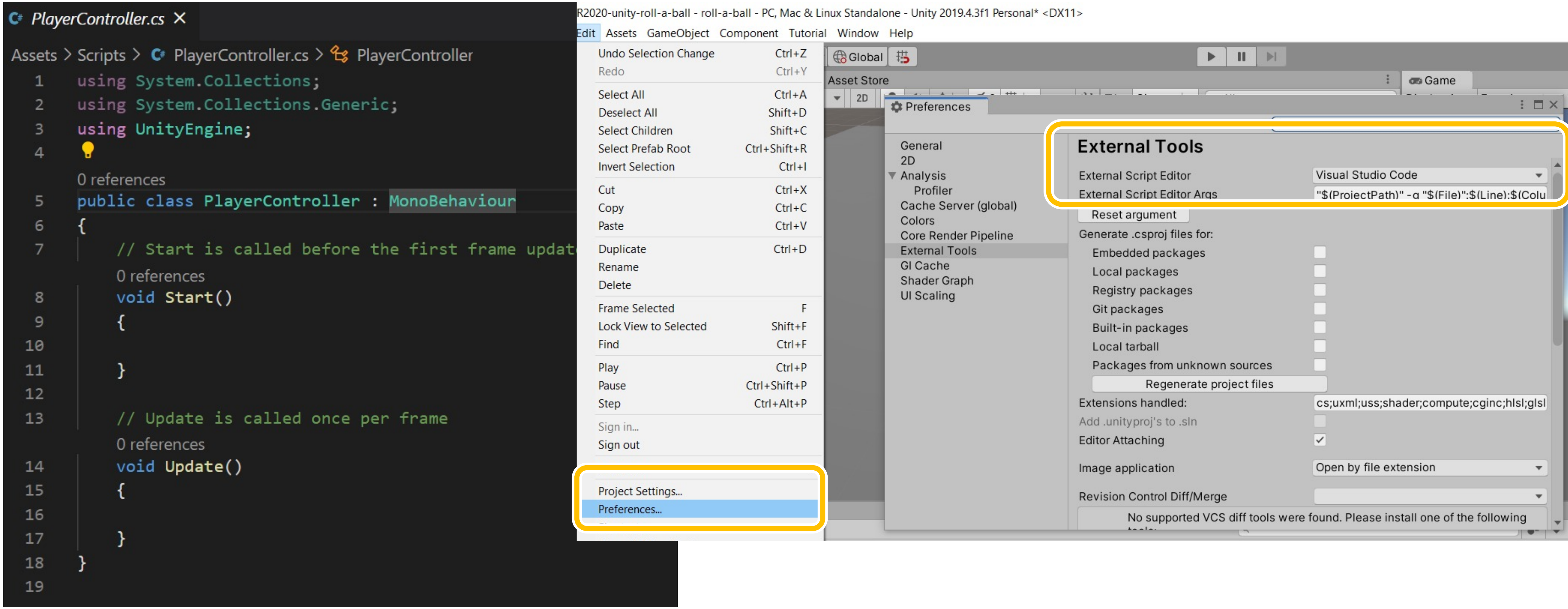
Directional Light
Transform
Light

Main Camera
Transform
Camera

Player
Transform
Renderer
Collider
Rigidbody
PlayerInput
[PlayerController.cs]

Ground
Transform
Renderer
Collider

Double click PlayerController.cs



PlayerController.cs ×

Assets > Scripts > PlayerController.cs > PlayerController

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

0 references
public class PlayerController : MonoBehaviour
{
    // Start is called before the first frame updat
    0 references
    void Start()
    {

    }

    // Update is called once per frame
    0 references
    void Update()
    {

    }
}
```

R2020-unity-roll-a-ball - roll-a-ball - PC, Mac & Linux Standalone - Unity 2019.4.3f1 Personal* <DX11>

External Tools

External Script Editor: Visual Studio Code

External Script Editor Args: "$(ProjectPath)" -g "$(File)":$(Line):$(Colu

Generate .csproj files for:

- Embedded packages □
- Local packages □
- Registry packages □
- Git packages □
- Built-in packages □
- Local tarball □
- Packages from unknown sources □

Extensions handled: cs;uxml;uss;shader;compute;cginc;hlsl;glsl

Add .unityproj's to .sln □

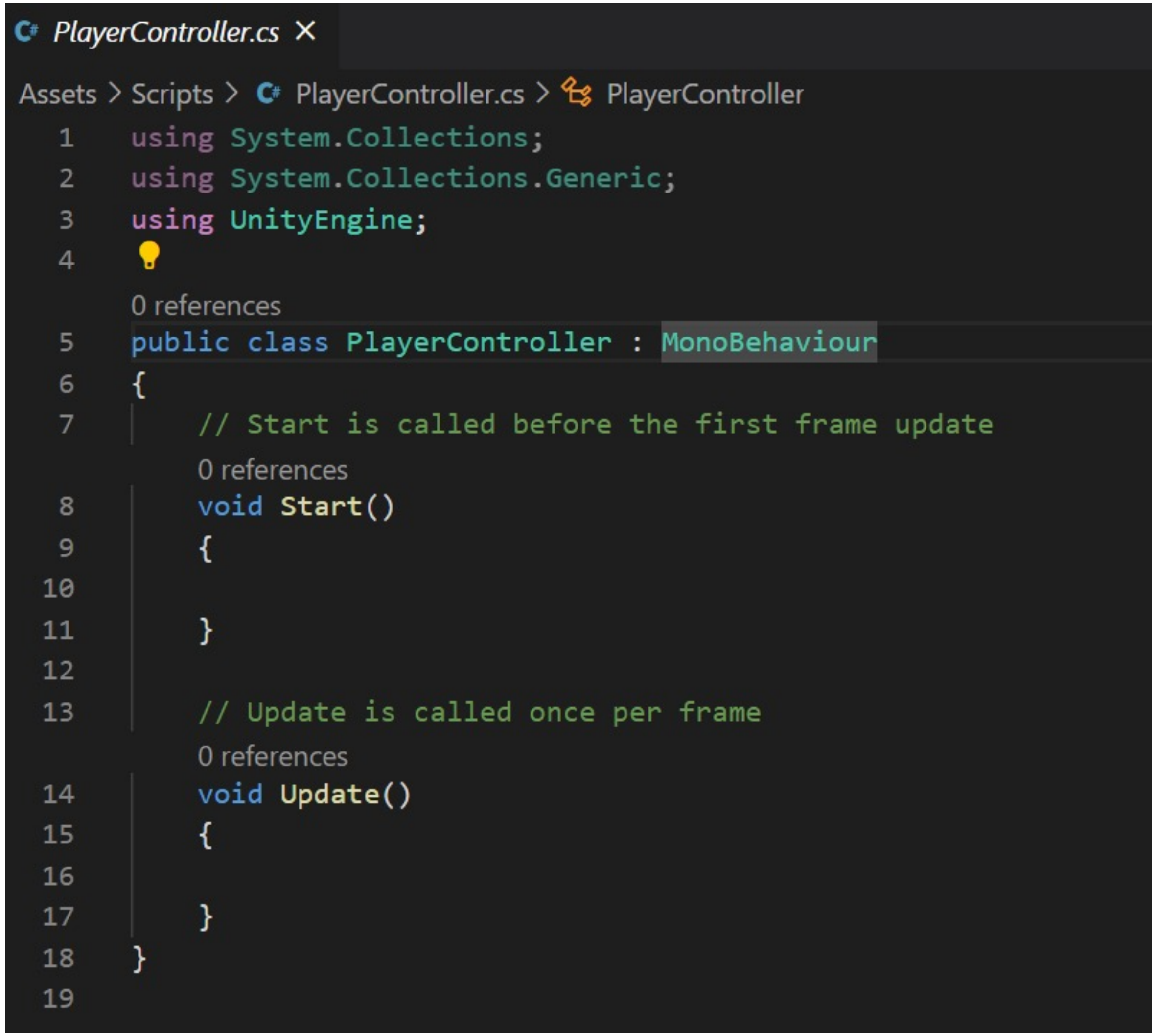
Editor Attaching ☑

Image application: Open by file extension

Revision Control Diff/Merge:

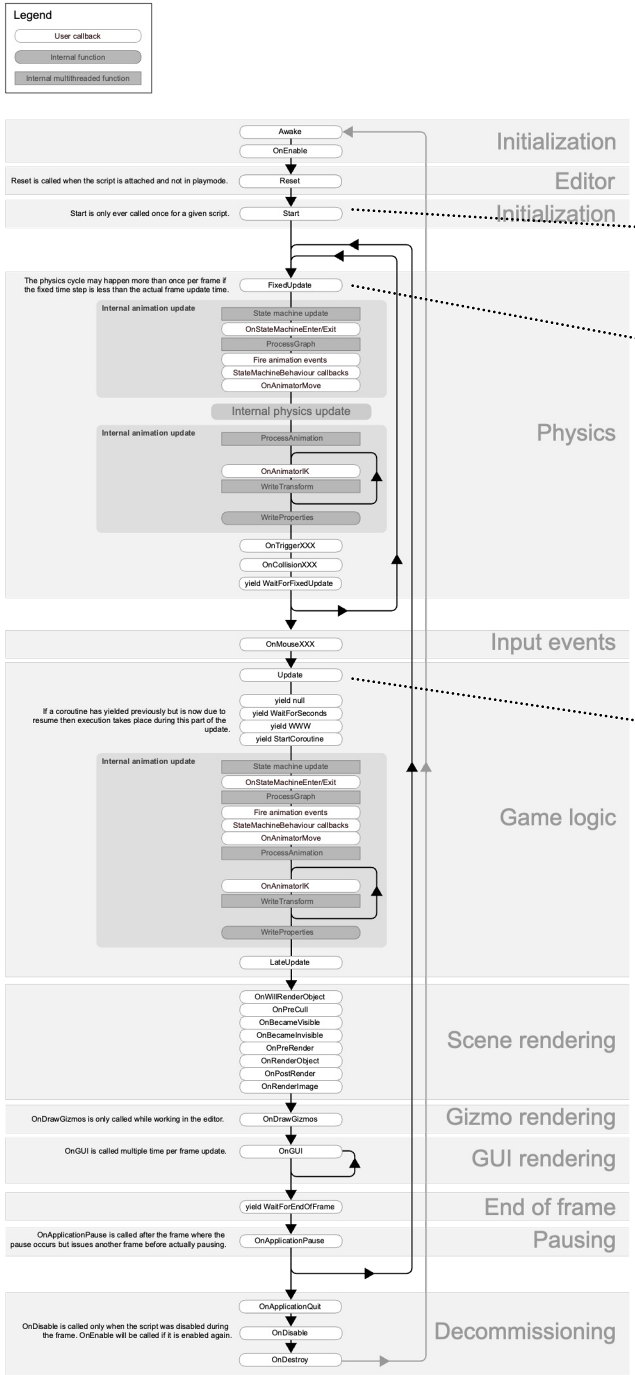
No supported VCS diff tools were found. Please install one of the following tools:

Double click PlayerController.cs



```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class PlayerController : MonoBehaviour
{
    // Start is called before the first frame update
    void Start()
    {

    }

    // Update is called once per frame
    void Update()
    {

    }
}

```

Unity Lifecycle



Legend

User callback

Internal function

Internal multithreaded function

Initialization

Awake

OnEnable

Editor

Reset is called when the script is attached and not in playmode.

Reset

Initialization

Start is only ever called once for a given script.

Start

Physics

The physics cycle may happen more than once per frame if the fixed time step is less than the actual frame update time.

FixedUpdate

Internal animation update

State machine update

OnStateMachineEnter/Exit

ProcessGraph

Fire animation events

StateMachineBehaviour callbacks

OnAnimatorMove

Internal physics update

Internal animation update

ProcessAnimation

OnAnimatorIK

WriteTransform

WriteProperties

OnTriggerXXX

OnCollisionXXX

yield WaitForFixedUpdate

Input events

OnMouseXXX

Game logic

Update

If a coroutine has yielded previously but is now due to resume then execution takes place during this part of the update.

yield null

yield WaitForSeconds

yield WWW

yield StartCoroutine

Internal animation update

State machine update

OnStateMachineEnter/Exit

ProcessGraph

Fire animation events

StateMachineBehaviour callbacks

OnAnimatorMove

ProcessAnimation

OnAnimatorIK

WriteTransform

WriteProperties

LateUpdate

Scene rendering

OnWillRenderObject

OnPreCull

OnBecameVisible

OnBecameInvisible

OnPreRender

OnRenderObject

OnPostRender

OnRenderImage

Gizmo rendering

OnDrawGizmos is only called while working in the editor.

OnDrawGizmos

GUI rendering

OnGUI is called multiple time per frame update.

OnGUI

End of frame

yield WaitForEndOfFrame

Pausing

OnApplicationPause is called after the frame where the pause occurs but issues another frame before actually pausing.

OnApplicationPause

Decommissioning

OnApplicationQuit

OnDisable is called only when the script was disabled during the frame. OnEnable will be called if it is enabled again.

OnDisable

OnDestroy

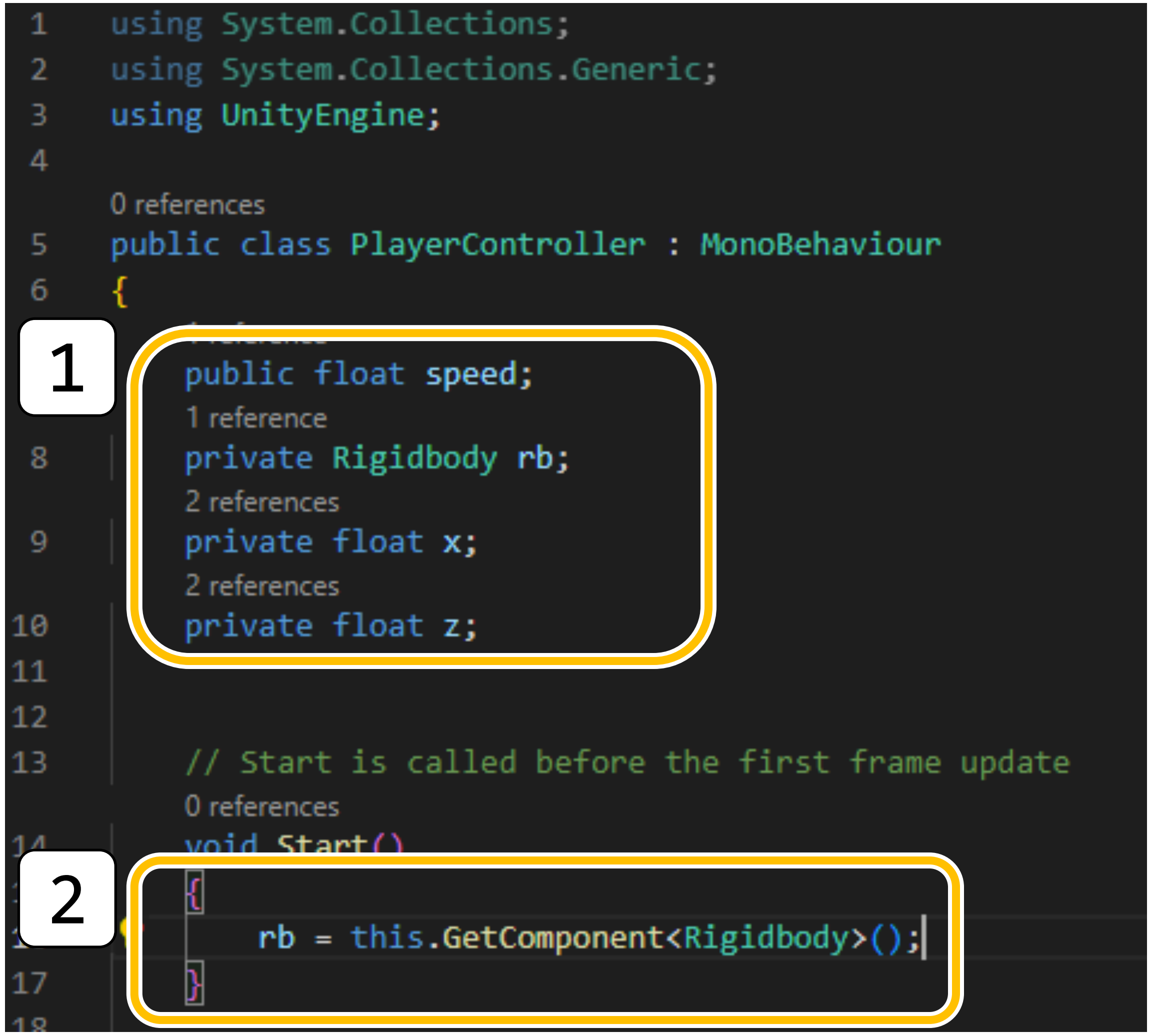
Start: called once at the beginning

FixedUpdate: for physics (e.g., Rigidbody)

Update: we usually put everything here except physics

PlayerController.cs

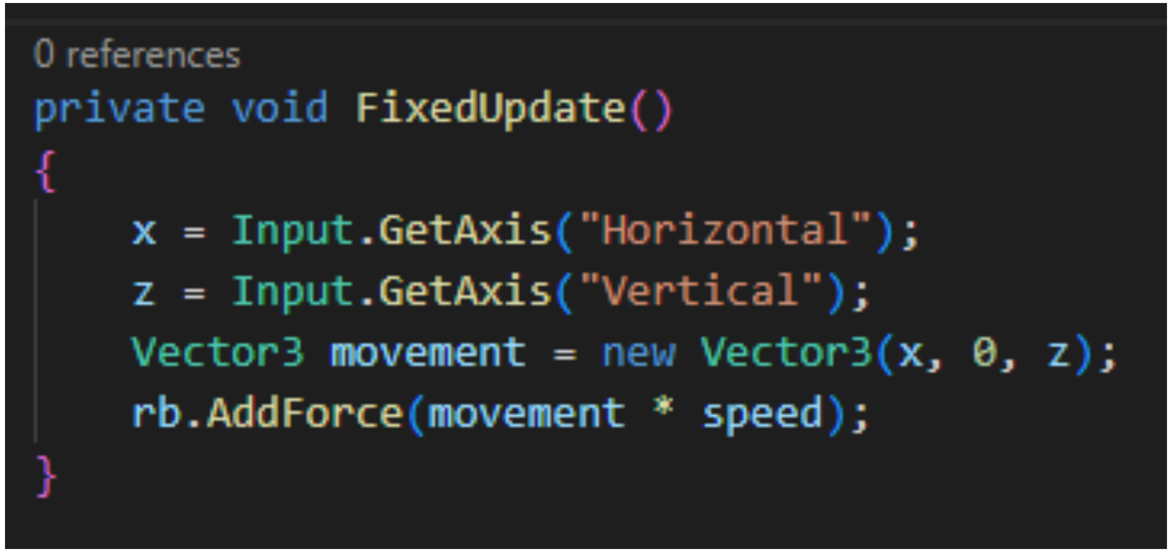
1. global variables
2. reference the rigidbody in the script



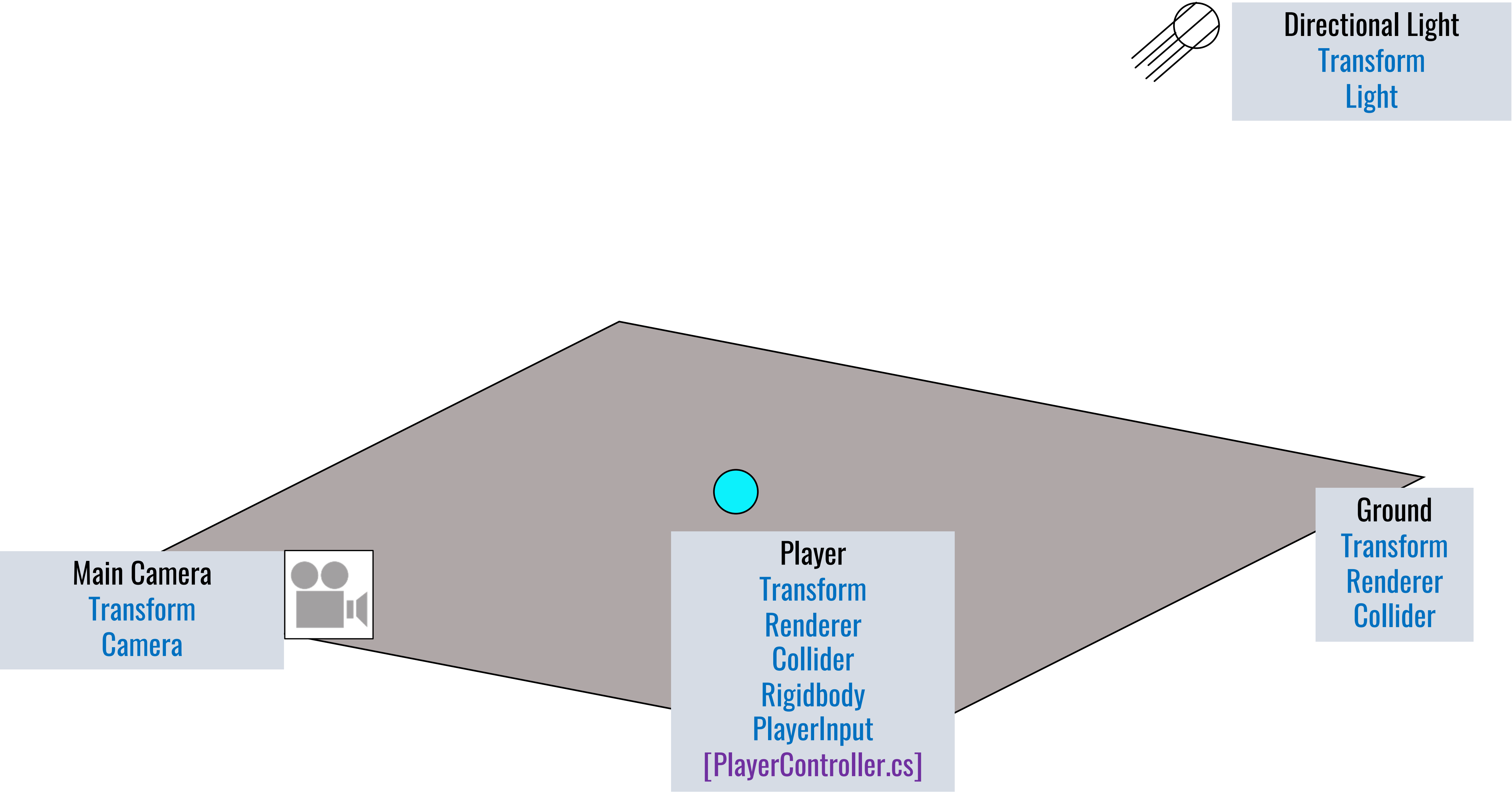
```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

0 references
public class PlayerController : MonoBehaviour
{
    1 reference
    public float speed;
    1 reference
    private Rigidbody rb;
    2 references
    private float x;
    2 references
    private float z;


    // Start is called before the first frame update
    0 references
    void Start()
    {
        rb = this.GetComponent<Rigidbody>();
    }
```

PlayerController.cs

- FixedUpdate(): move the Player using .AddForce



```
0 references
private void FixedUpdate()
{
    x = Input.GetAxis("Horizontal");
    z = Input.GetAxis("Vertical");
    Vector3 movement = new Vector3(x, 0, z);
    rb.AddForce(movement * speed);
}
```

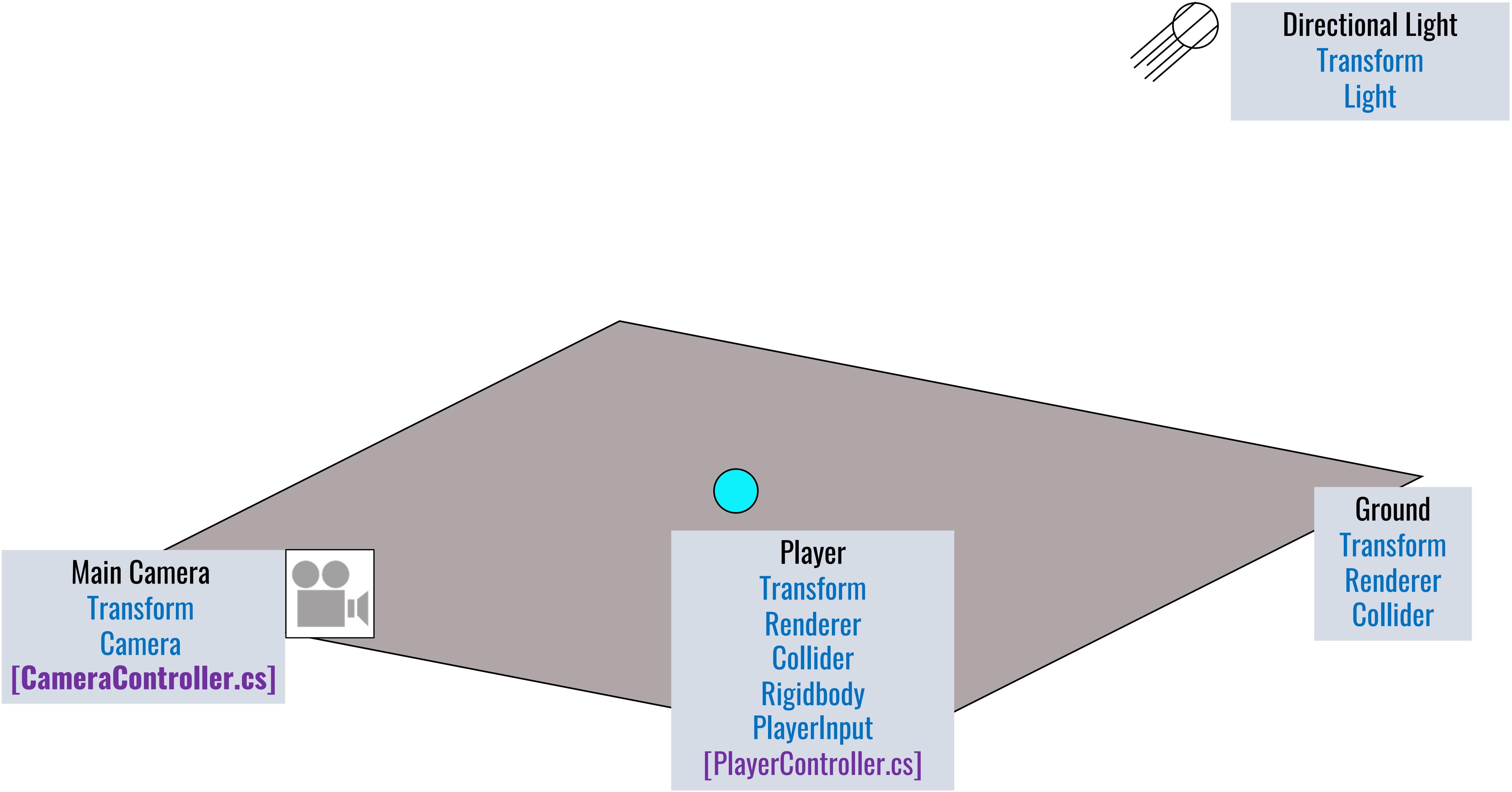


Directional Light
Transform
Light

Main Camera
Transform
Camera

Player
Transform
Renderer
Collider
Rigidbody
PlayerInput
[PlayerController.cs]

Ground
Transform
Renderer
Collider

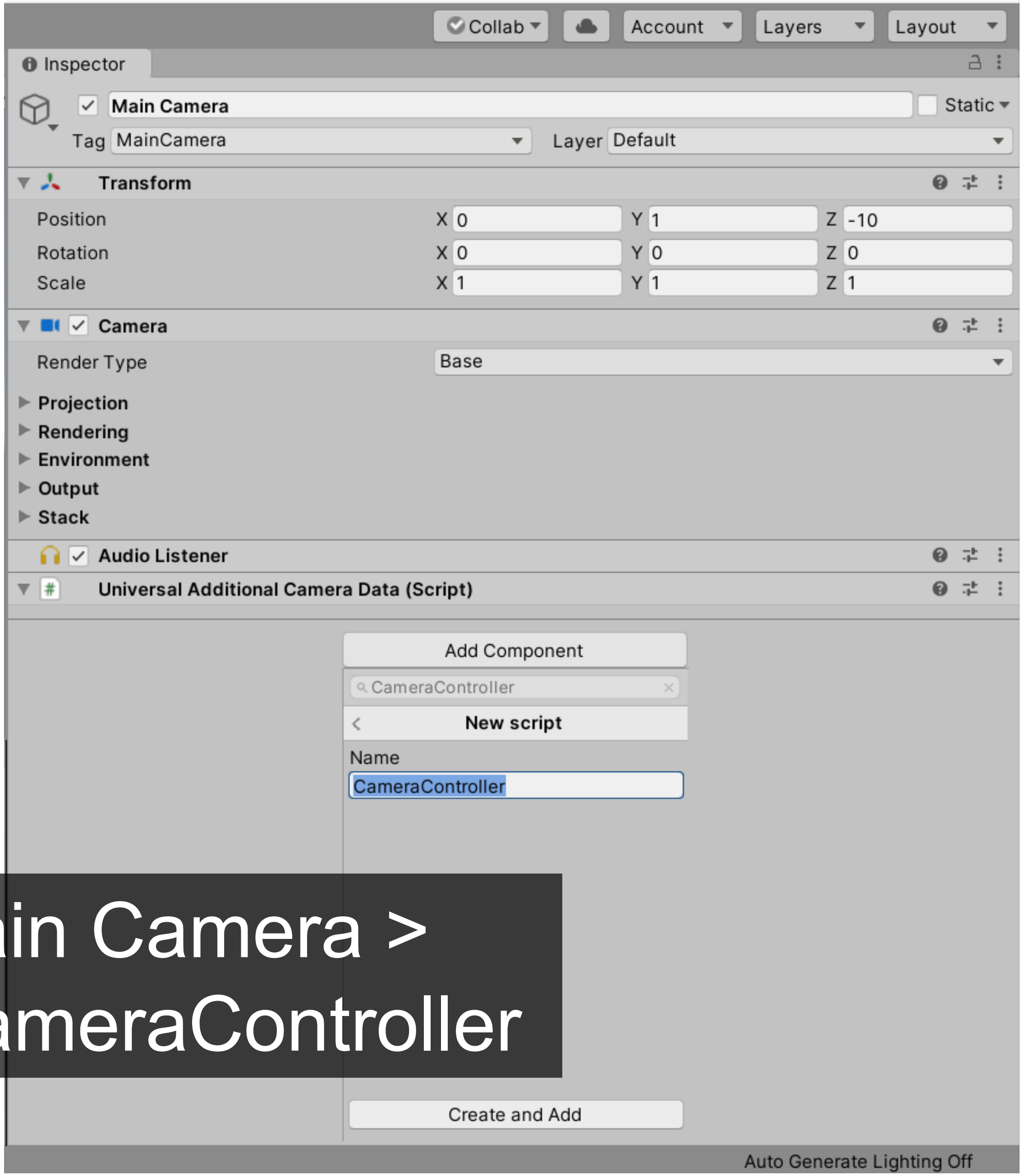


Directional Light
Transform
Light

Main Camera
Transform
Camera
[CameraController.cs]

Player
Transform
Renderer
Collider
Rigidbody
PlayerInput
[PlayerController.cs]

Ground
Transform
Renderer
Collider



☑ Main Camera □ Static

Tag MainCamera Layer Default

Transform

Position	X 0	Y 1	Z -10
Rotation	X 0	Y 0	Z 0
Scale	X 1	Y 1	Z 1

☑ Camera

Render Type Base

- Projection
- Rendering
- Environment
- Output
- Stack

☑ Audio Listener

Universal Additional Camera Data (Script)

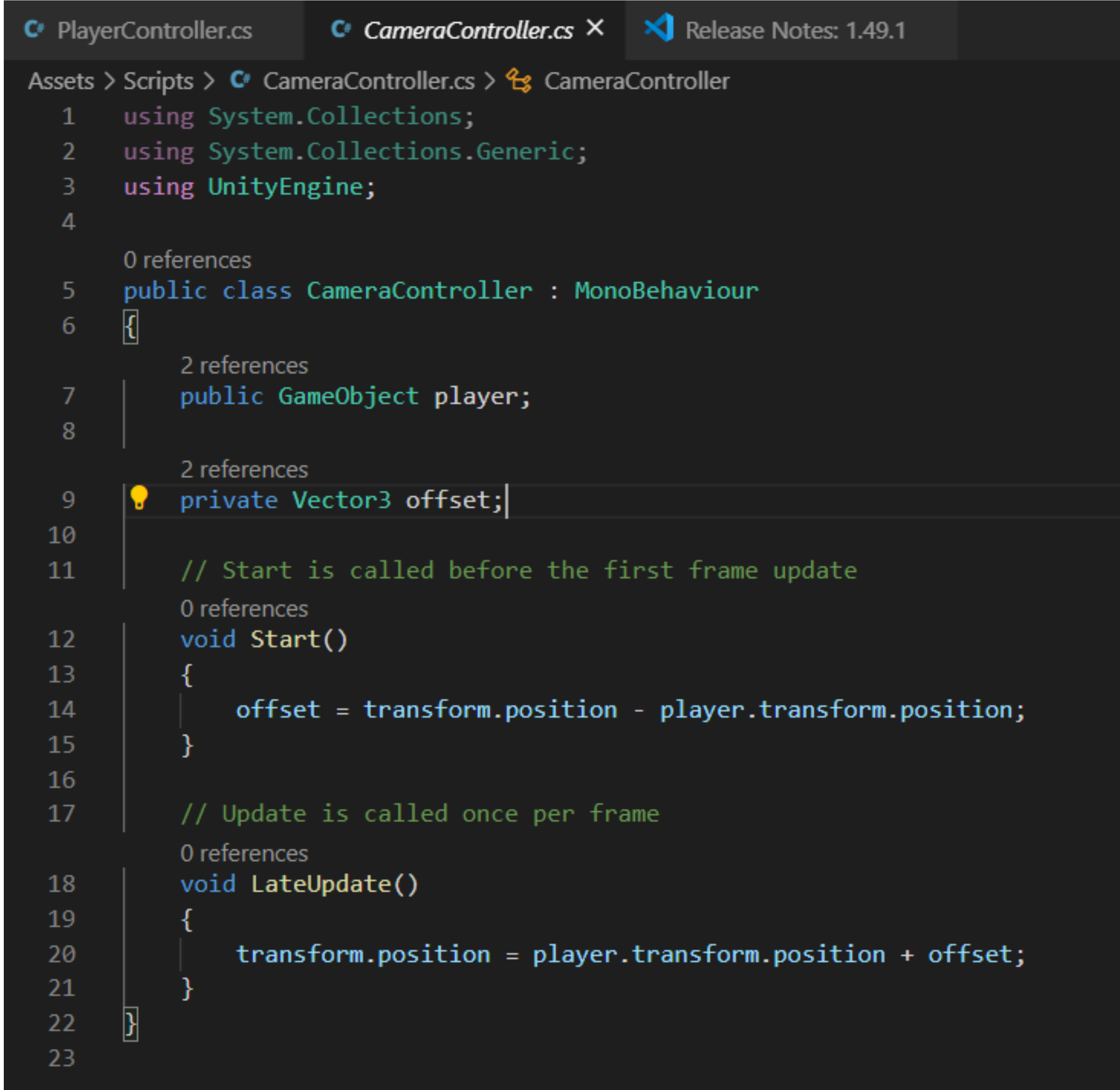
CameraController

Name

CameraController

select Main Camera > create CameraController

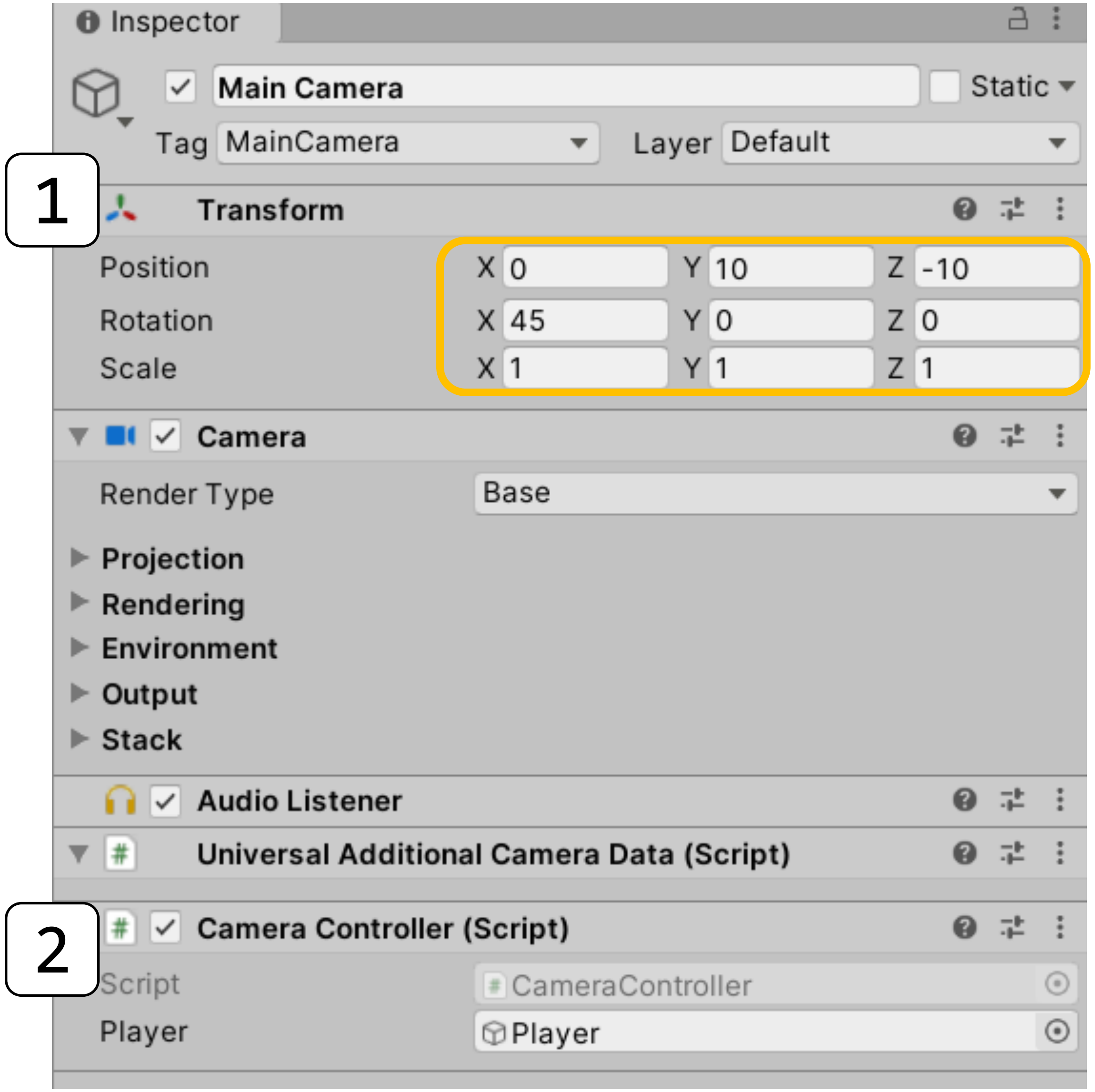
CameraController.cs



```csharp
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class CameraController : MonoBehaviour
{
    public GameObject player;

    private Vector3 offset;

    // Start is called before the first frame update
    void Start()
    {
        offset = transform.position - player.transform.position;
    }

    // Update is called once per frame
    void LateUpdate()
    {
        transform.position = player.transform.position + offset;
    }
}

```

Inspector of Main Camera

1. Adjust the Transform for viewing angle of the game.
2. Drag the Player GameObject in the reference of CameraController.cs



Inspector

☑ Main Camera □ Static

Tag MainCamera Layer Default

Transform

Position X 0 Y 10 Z -10

Rotation X 45 Y 0 Z 0

Scale X 1 Y 1 Z 1

☑ Camera

Render Type Base

- Projection
- Rendering
- Environment
- Output
- Stack

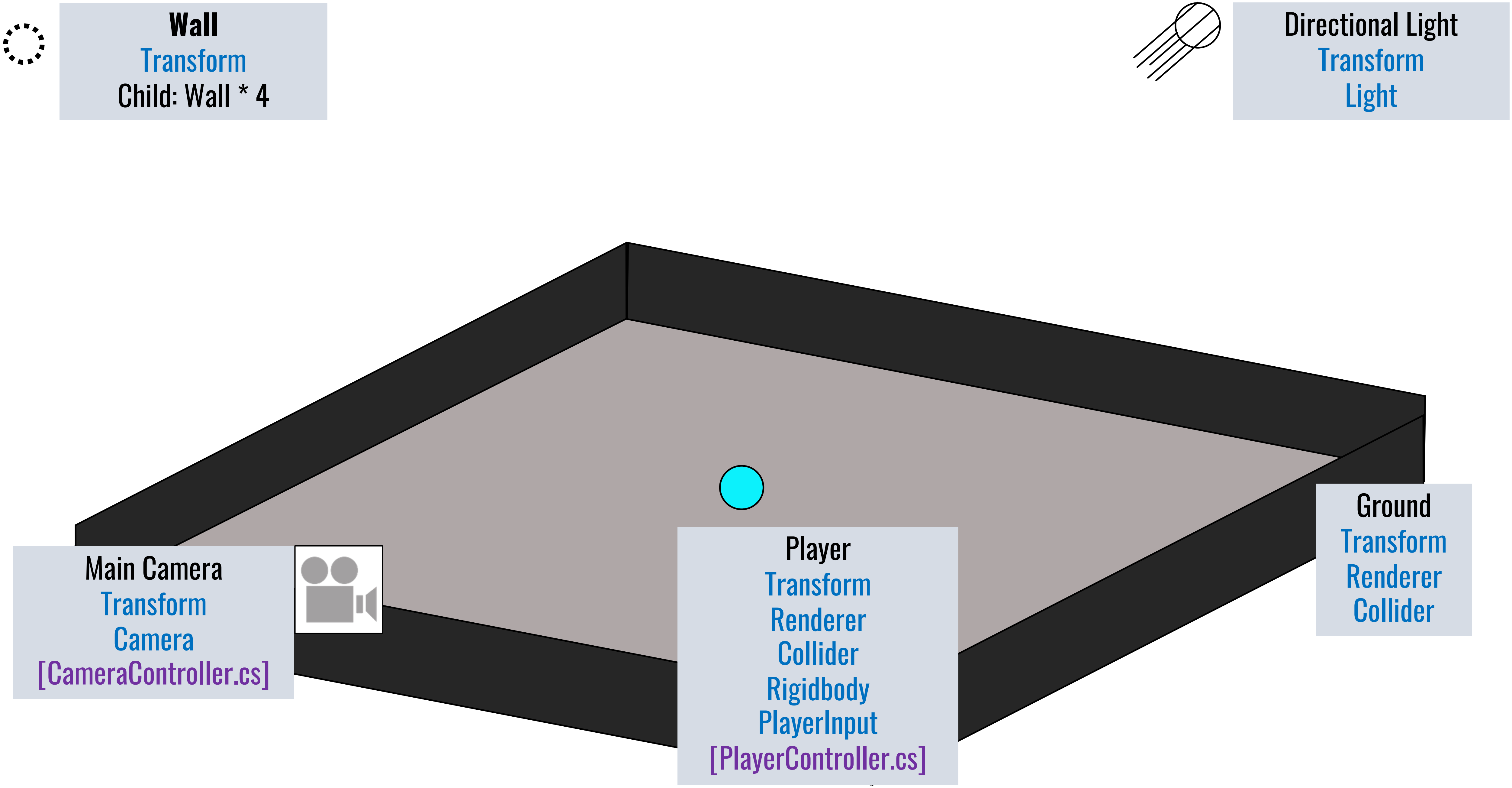
☑ Audio Listener

Universal Additional Camera Data (Script)

☑ Camera Controller (Script)

Script CameraController

Player Player



Wall
Transform
Child: Wall * 4

Directional Light
Transform
Light

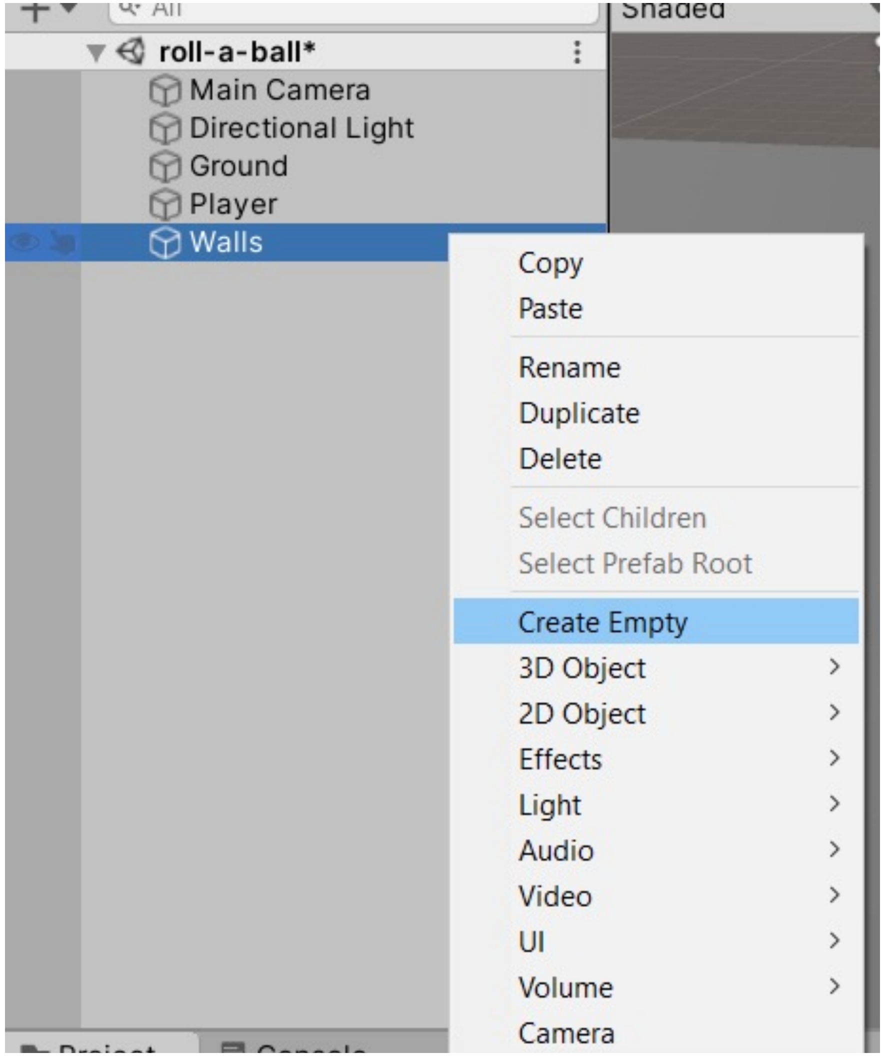
Main Camera
Transform
Camera
[CameraController.cs]

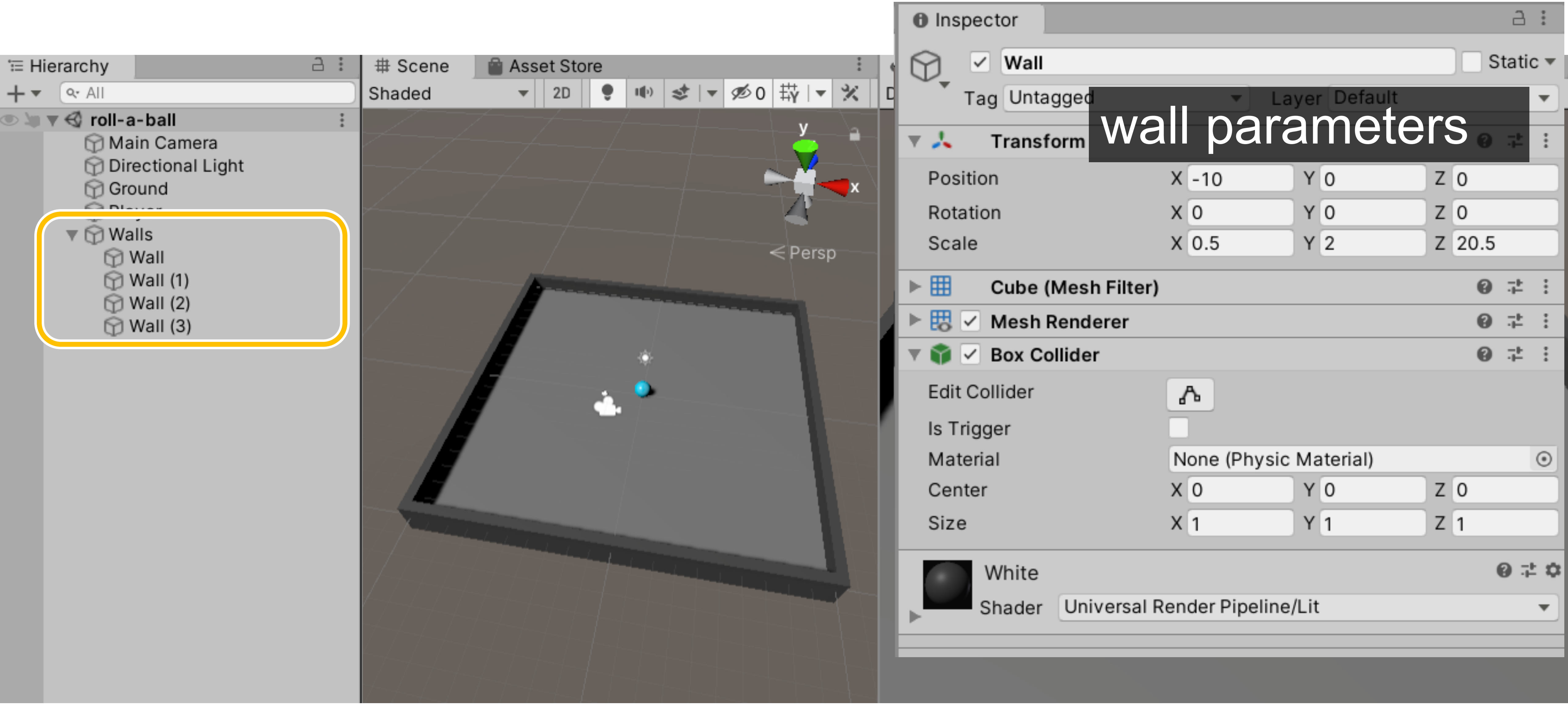
Player
Transform
Renderer
Collider
Rigidbody
PlayerInput
[PlayerController.cs]

Ground
Transform
Renderer
Collider

create Empty GameObject

- We use empty GO to collect things together (e.g., walls, environment).
- Therefore, you can manipulate them all at the same time.





wall parameters

☑ Wall □ Static

Tag Untagged Layer Default

Transform

Position	X	-10	Y	0	Z	0
Rotation	X	0	Y	0	Z	0
Scale	X	0.5	Y	2	Z	20.5

Cube (Mesh Filter)

☑ Mesh Renderer

☑ Box Collider

Edit Collider

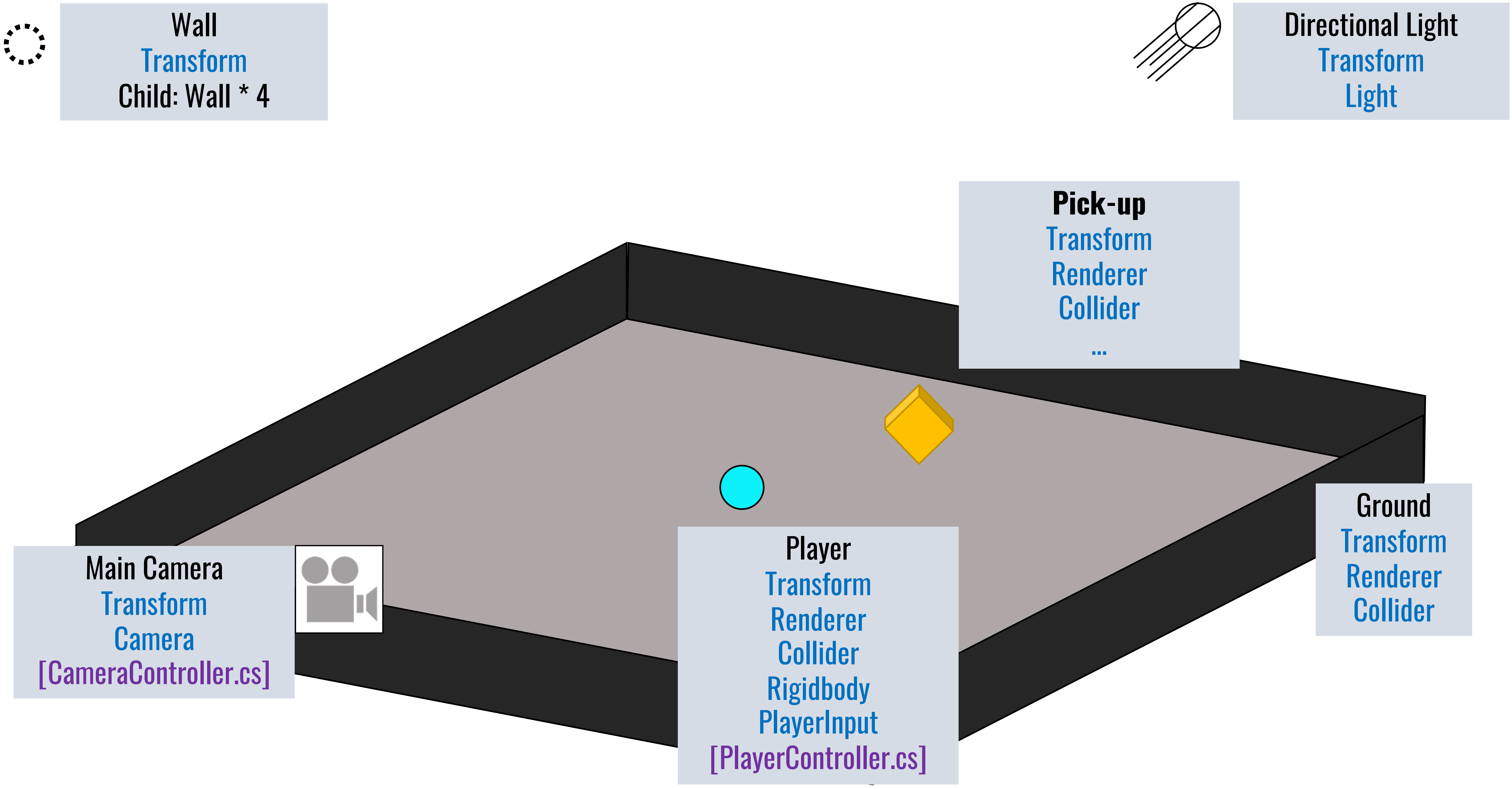
Is Trigger □

Material None (Physic Material)

Center	X	0	Y	0	Z	0
Size	X	1	Y	1	Z	1

White

Shader Universal Render Pipeline/Lit



Wall
Transform
Child: Wall * 4

Directional Light
Transform
Light

Pick-up
Transform
Renderer
Collider
...

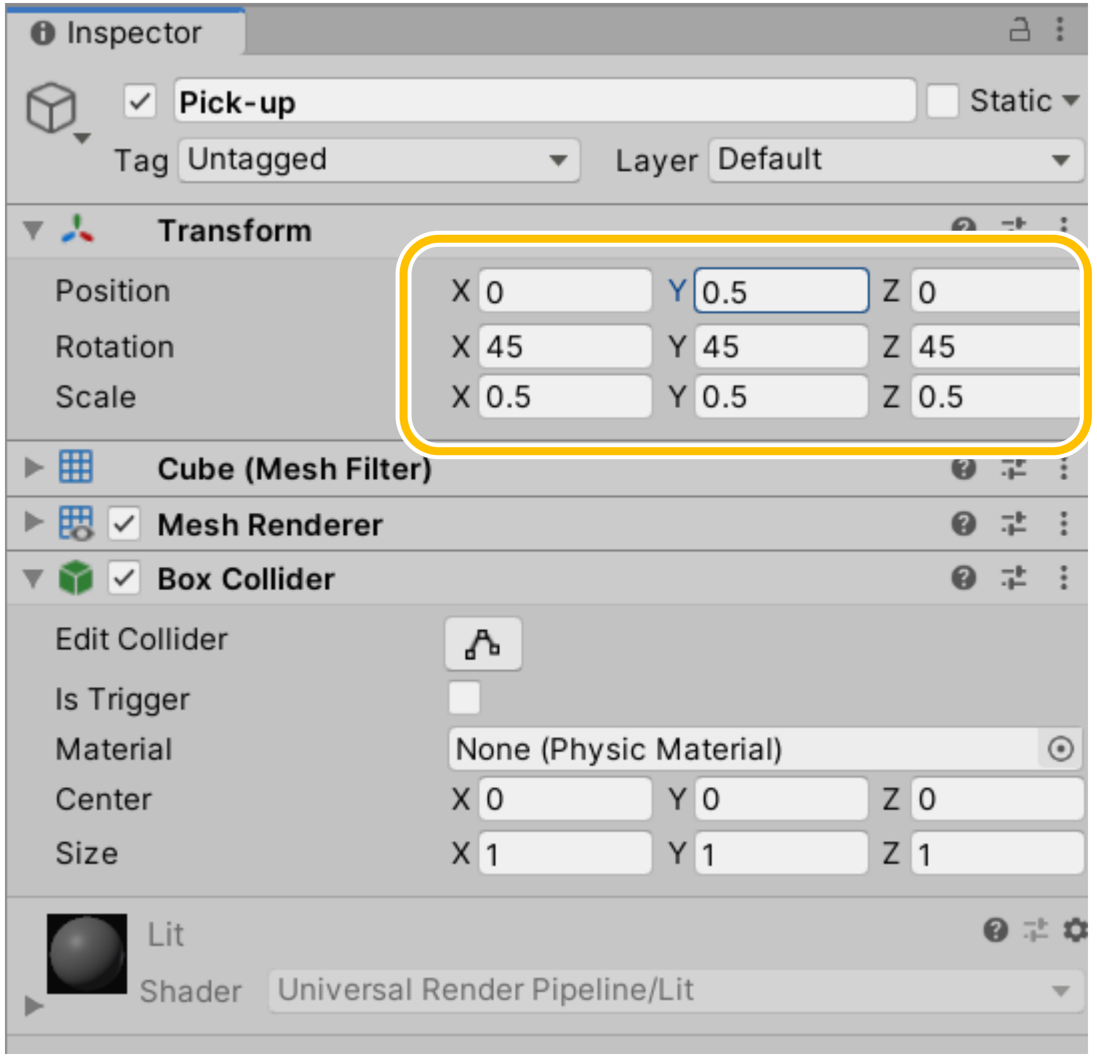
Ground
Transform
Renderer
Collider

Main Camera
Transform
Camera
[CameraController.cs]

Player
Transform
Renderer
Collider
Rigidbody
PlayerInput
[PlayerController.cs]

The pick-up

- Create a cube and name as pick-up
- Change Transform
- Create a material for the pick-up



Inspector

☑ Pick-up □ Static

Tag Untagged Layer Default

Transform

Position X 0 Y 0.5 Z 0

Rotation X 45 Y 45 Z 45

Scale X 0.5 Y 0.5 Z 0.5

Cube (Mesh Filter)

☑ Mesh Renderer

☑ Box Collider

Edit Collider

Is Trigger □

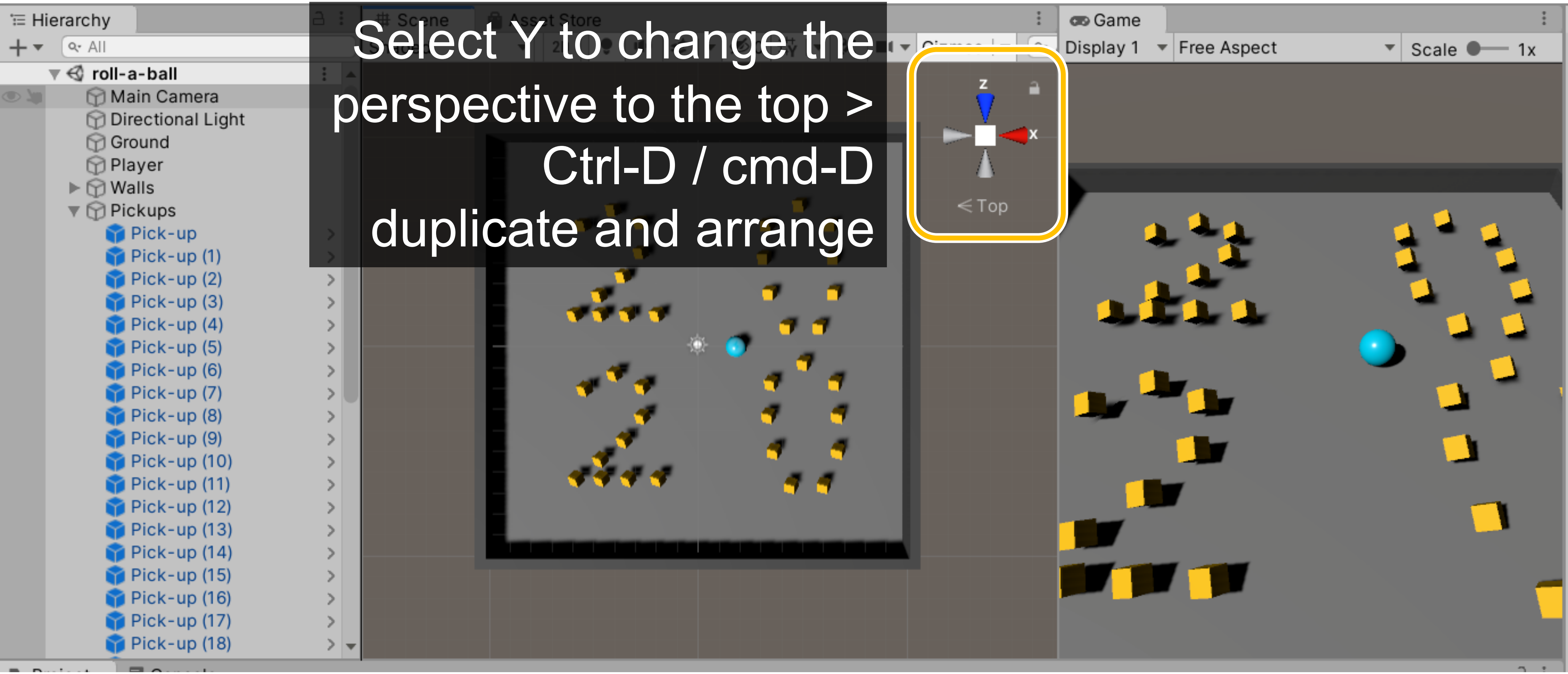
Material None (Physic Material)

Center X 0 Y 0 Z 0

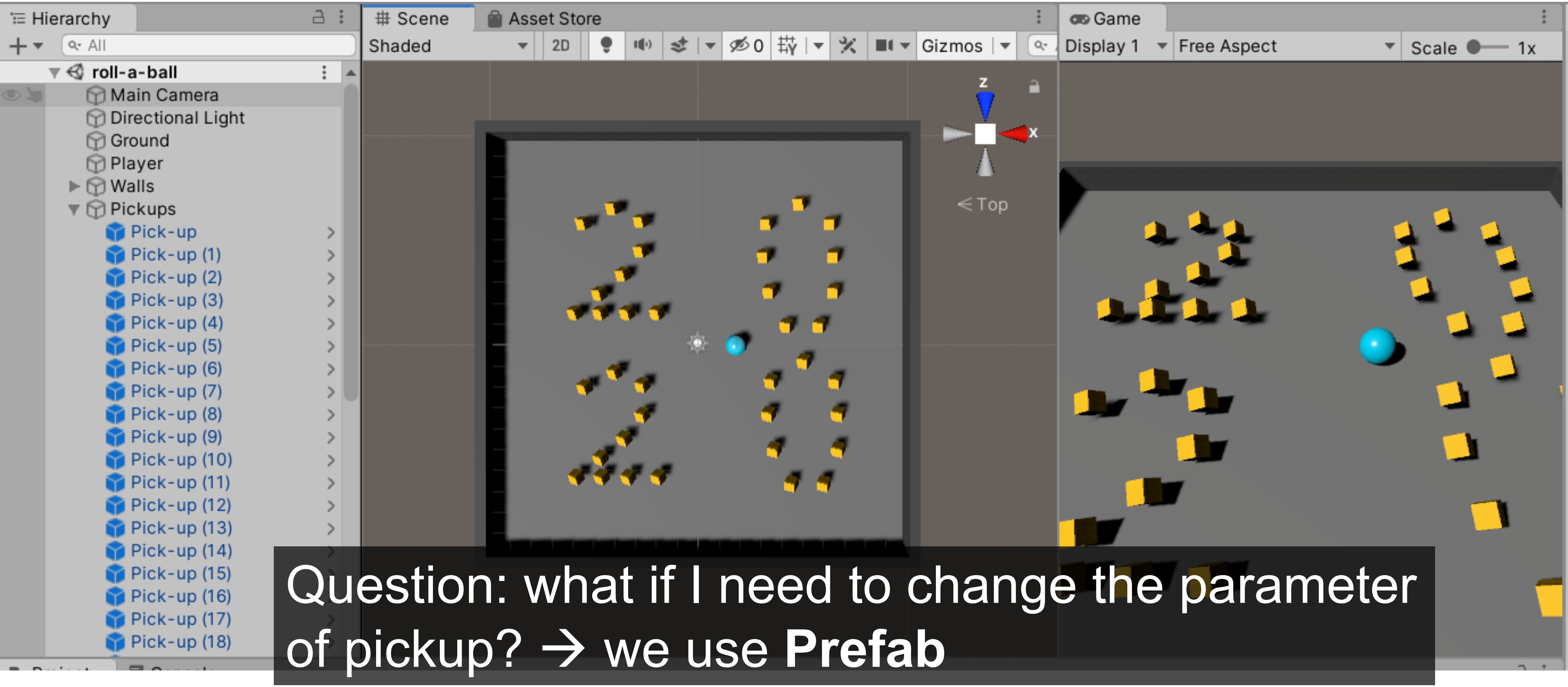
Size X 1 Y 1 Z 1

Lit

Shader Universal Render Pipeline/Lit

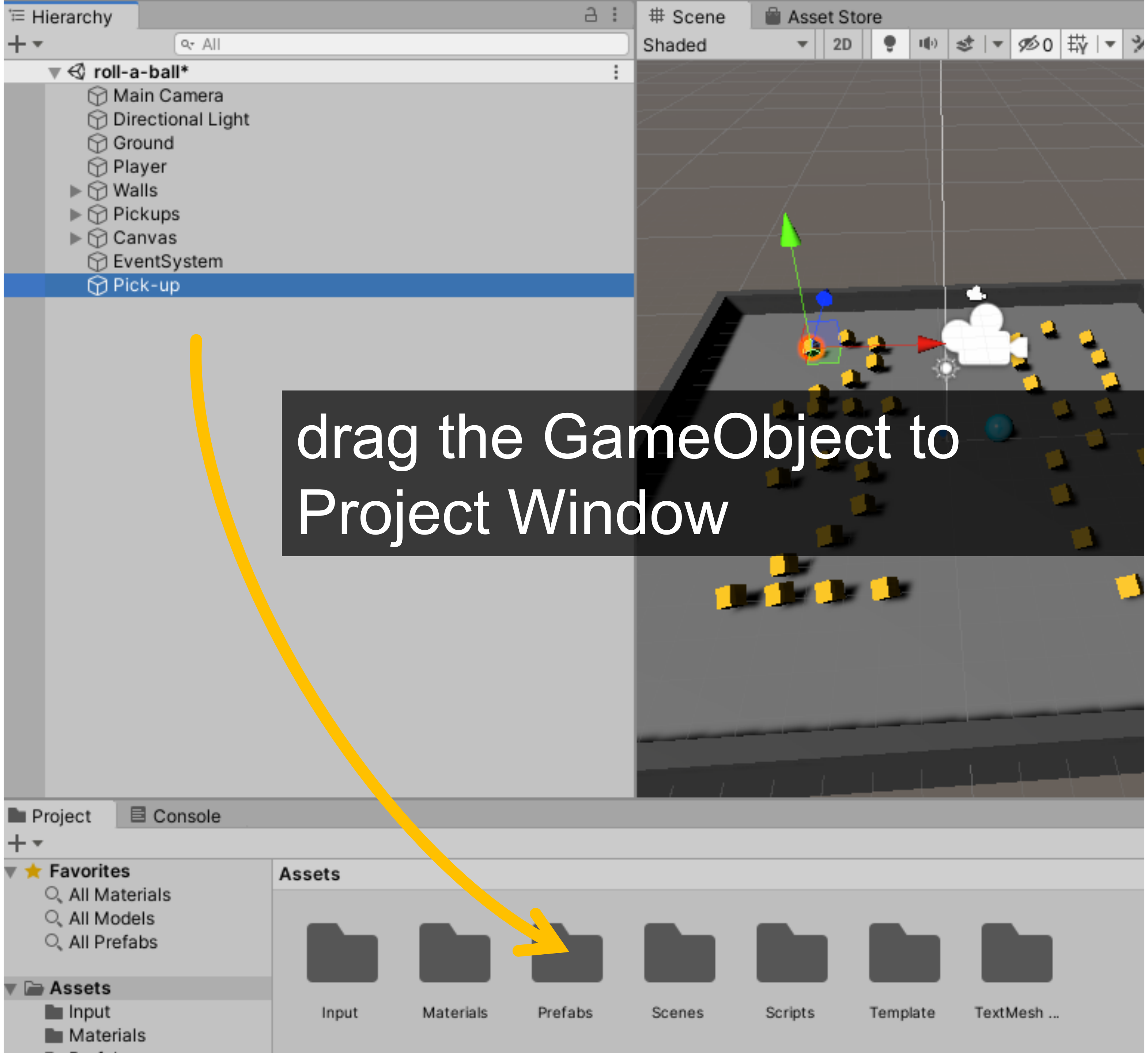


Select Y to change the perspective to the top > Ctrl-D / cmd-D duplicate and arrange



Question: what if I need to change the parameter of pickup? → we use **Prefab**

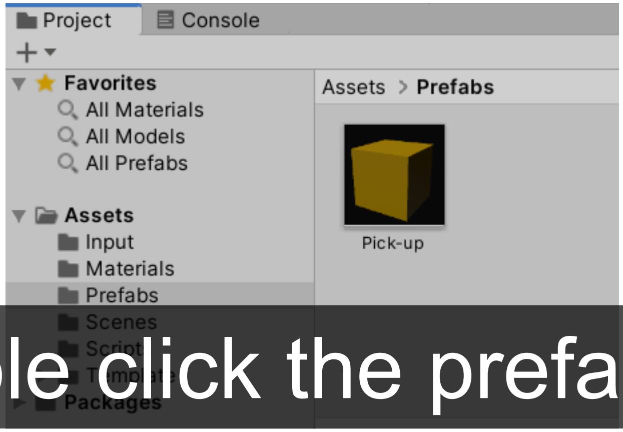
create Prefab

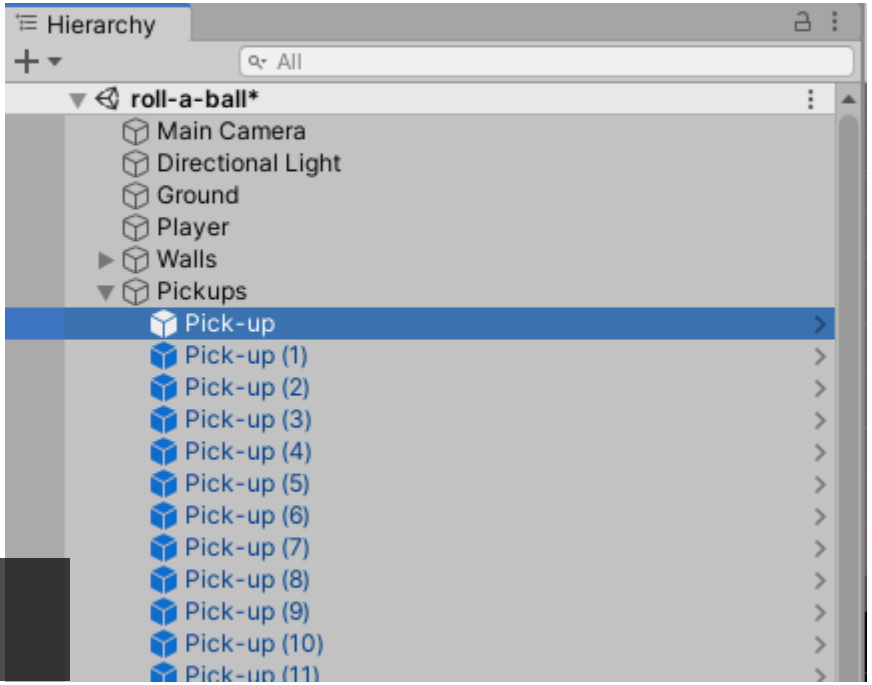


drag the GameObject to Project Window

create Prefab

- The GameObject in hierarchy turns blue.

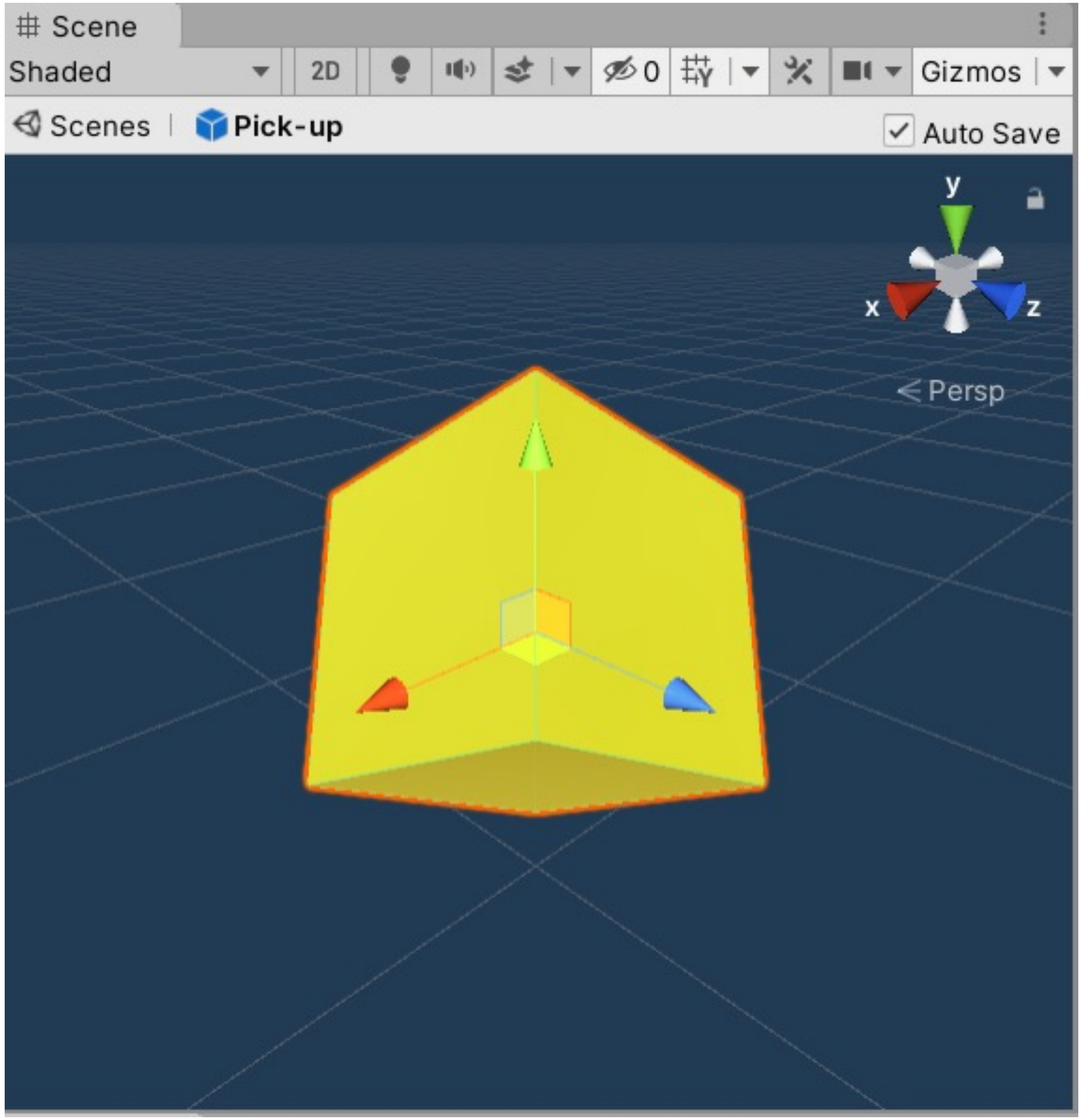




double click the prefab

Prefab editing mode

- The changes you made in this mode will be passing to all the prefab gameobjects in the scene.

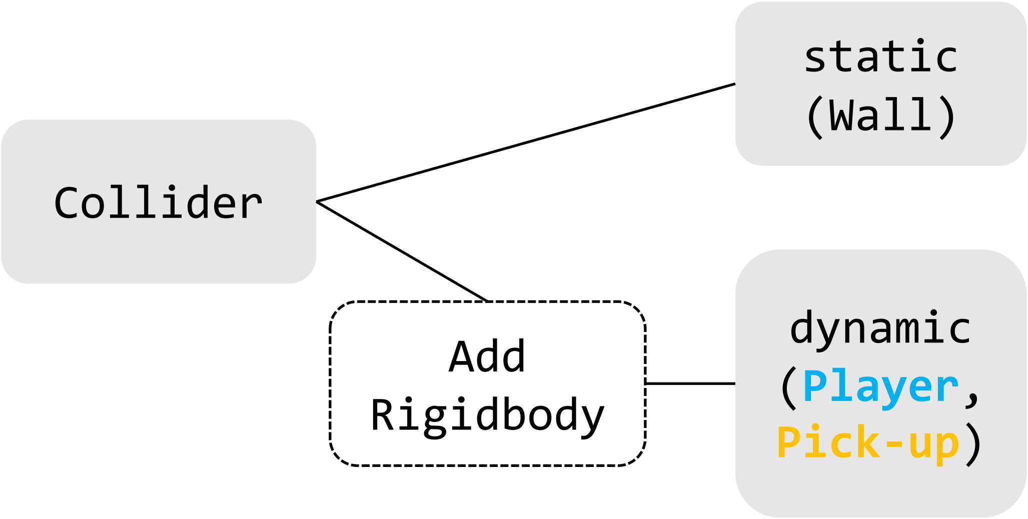


Our goal in this game:

- **when Player hits Pick-up, the Pick-up disappears** and increase the score.
- if score > X, win.

Let’s have a look in our game

- Unity Colliders



detect collision:

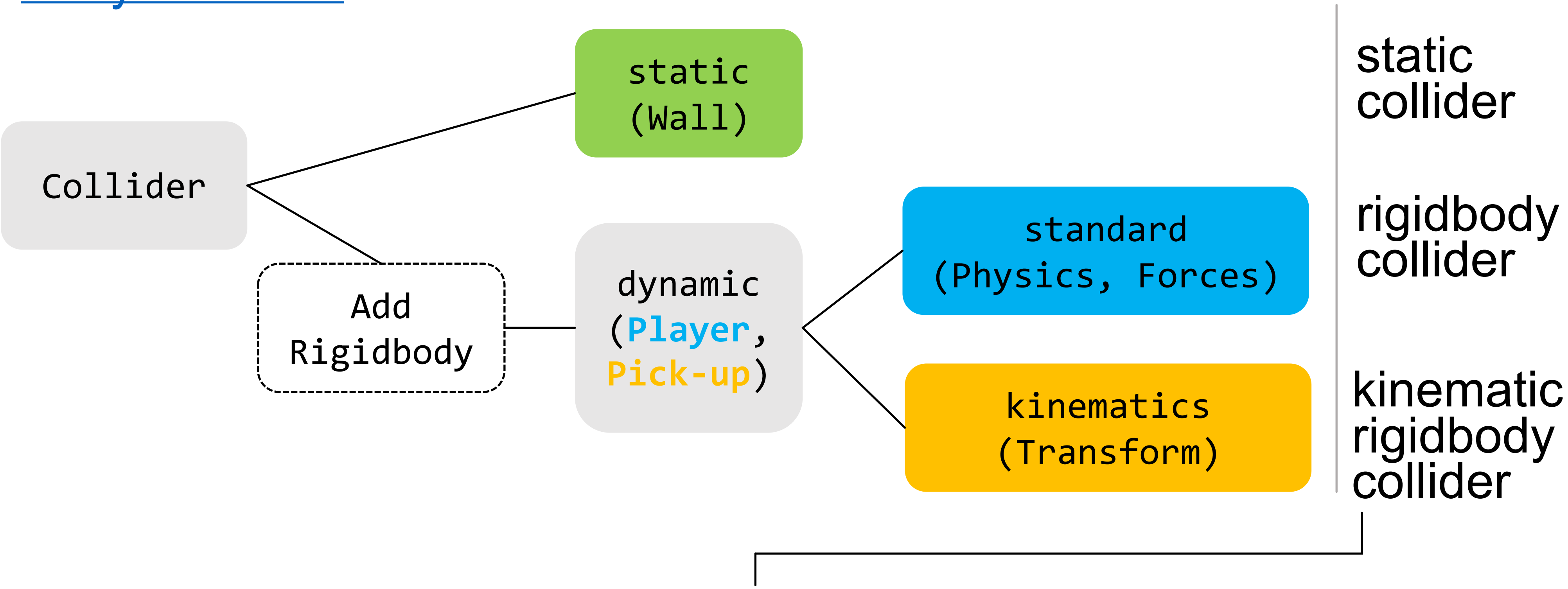
- OnCollisionEnter()
- OnTriggerEnter()

  detect when one collider enters the space of another without creating a collision

  In our example:
  Player has OnTriggerEnter
  Pick-up is triggered

Let’s have a look in our game

- Unity Colliders



Collider

static (Wall)

Add Rigidbody

dynamic (Player, Pick-up)

standard (Physics, Forces)

kinematics (Transform)

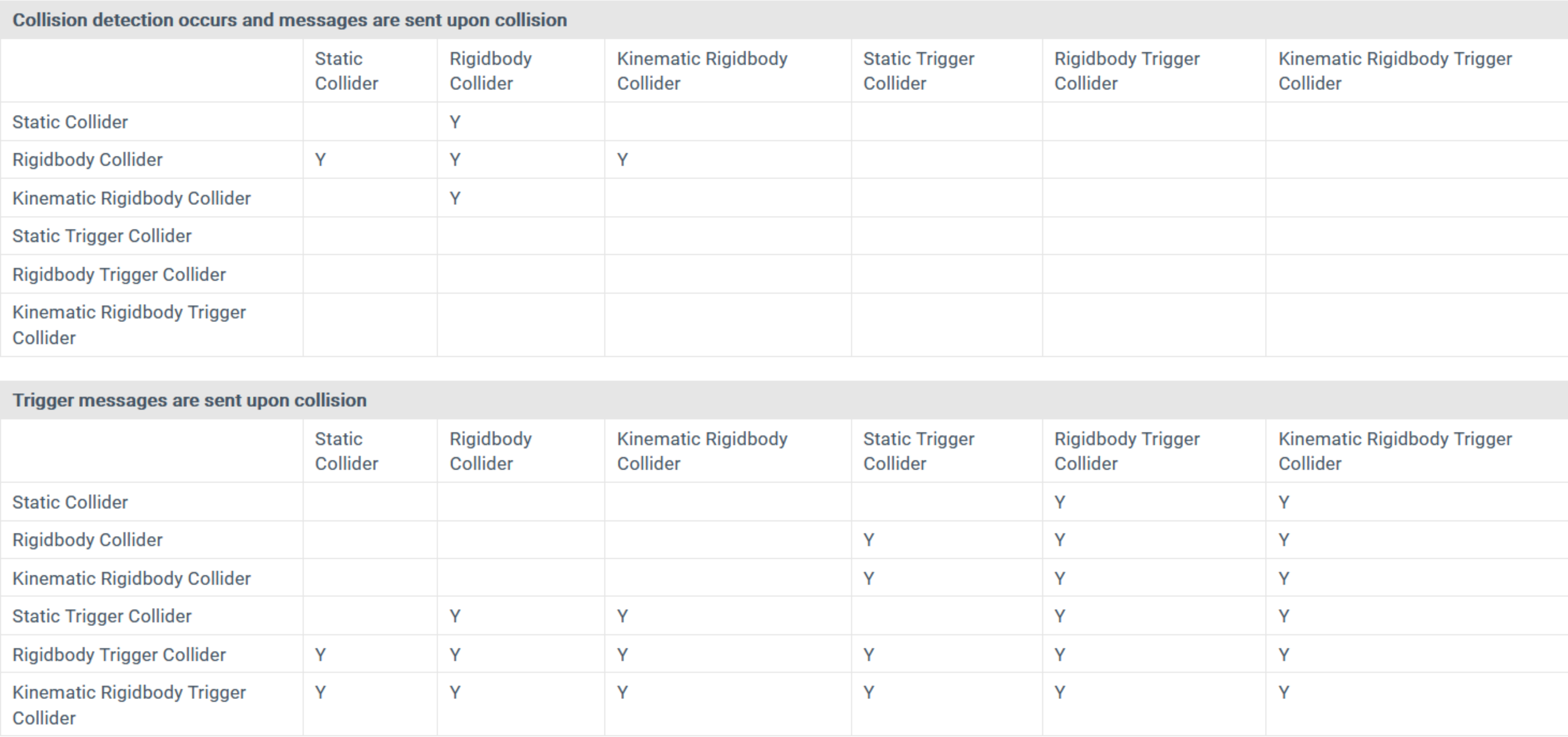
static collider

rigidbody collider

kinematic rigidbody collider

move a kinematic rigidbody object by modifying its **Transform Component** but it will not respond to collisions and forces.

Unity Colliders



Collision detection occurs and messages are sent upon collision

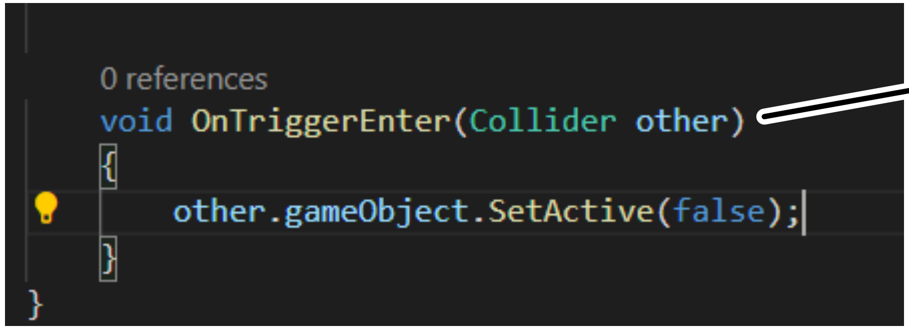
	Static Collider	Rigidbody Collider	Kinematic Rigidbody Collider	Static Trigger Collider	Rigidbody Trigger Collider	Kinematic Rigidbody Trigger Collider
Static Collider		Y				
Rigidbody Collider	Y	Y	Y			
Kinematic Rigidbody Collider		Y				
Static Trigger Collider						
Rigidbody Trigger Collider						
Kinematic Rigidbody Trigger Collider						

Trigger messages are sent upon collision

	Static Collider	Rigidbody Collider	Kinematic Rigidbody Collider	Static Trigger Collider	Rigidbody Trigger Collider	Kinematic Rigidbody Trigger Collider
Static Collider					Y	Y
Rigidbody Collider				Y	Y	Y
Kinematic Rigidbody Collider				Y	Y	Y
Static Trigger Collider		Y	Y		Y	Y
Rigidbody Trigger Collider	Y	Y	Y	Y	Y	Y
Kinematic Rigidbody Trigger Collider	Y	Y	Y	Y	Y	Y

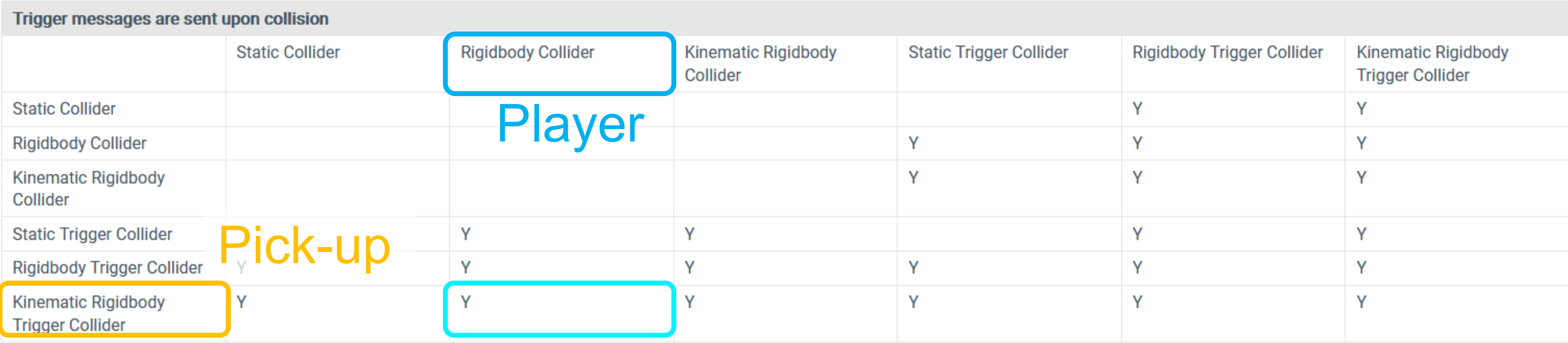
back to PlayerController.cs

- Unity Colliders



```
    0 references
    void OnTriggerEnter(Collider other)
    {
        other.gameObject.SetActive(false);
    }
}
```

means Pick-up



Trigger messages are sent upon collision

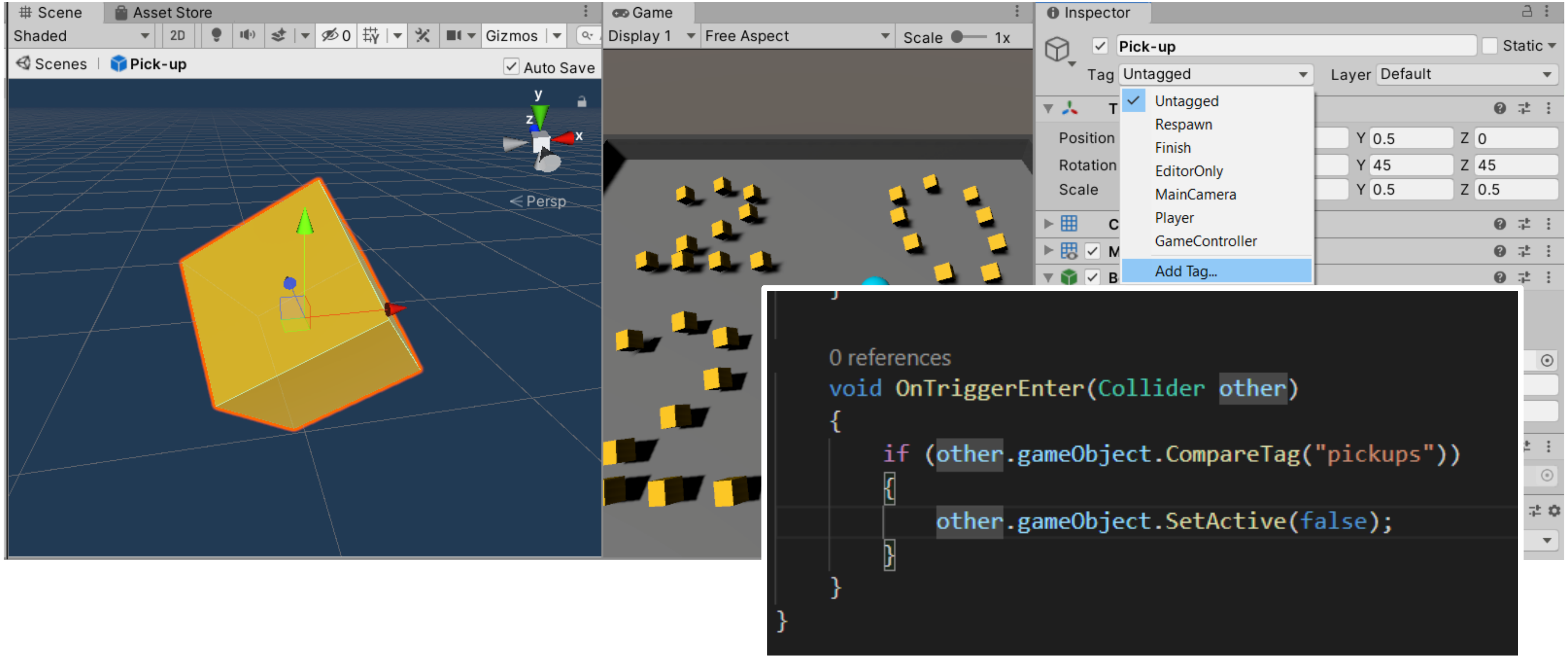
	Static Collider	Rigidbody Collider	Kinematic Rigidbody Collider	Static Trigger Collider	Rigidbody Trigger Collider	Kinematic Rigidbody Trigger Collider
Static Collider					Y	Y
Rigidbody Collider				Y	Y	Y
Kinematic Rigidbody Collider				Y	Y	Y
Static Trigger Collider		Y	Y		Y	Y
Rigidbody Trigger Collider	Y	Y	Y	Y	Y	Y
Kinematic Rigidbody Trigger Collider	Y	Y	Y	Y	Y	Y

Player

Pick-up

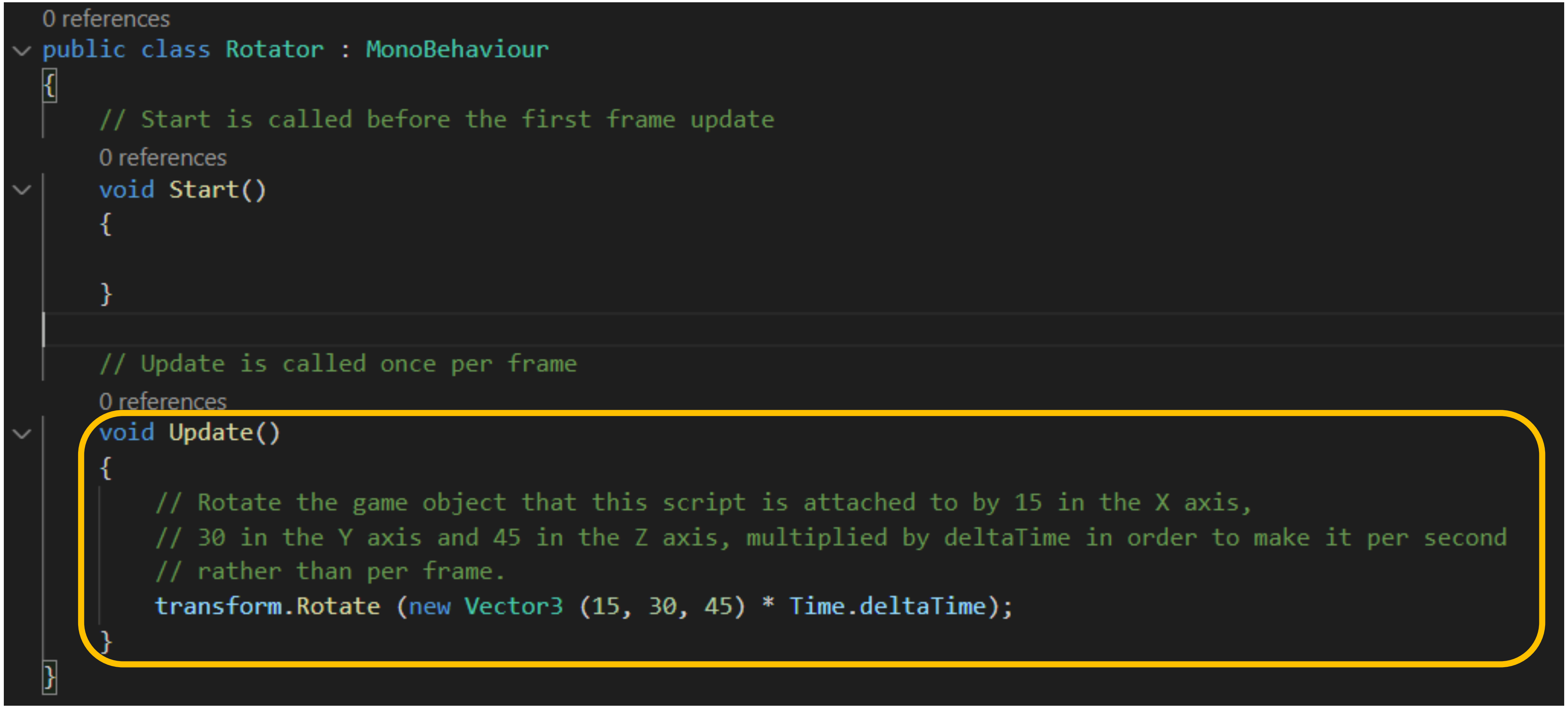
Add tag

- In the inspector of pick-up prefab, select Add tag and add “pickups”.



```
    }

    0 references
    void OnTriggerEnter(Collider other)
    {
        if (other.gameObject.CompareTag("pickups"))
        {
            other.gameObject.SetActive(false);
        }
    }
}
```

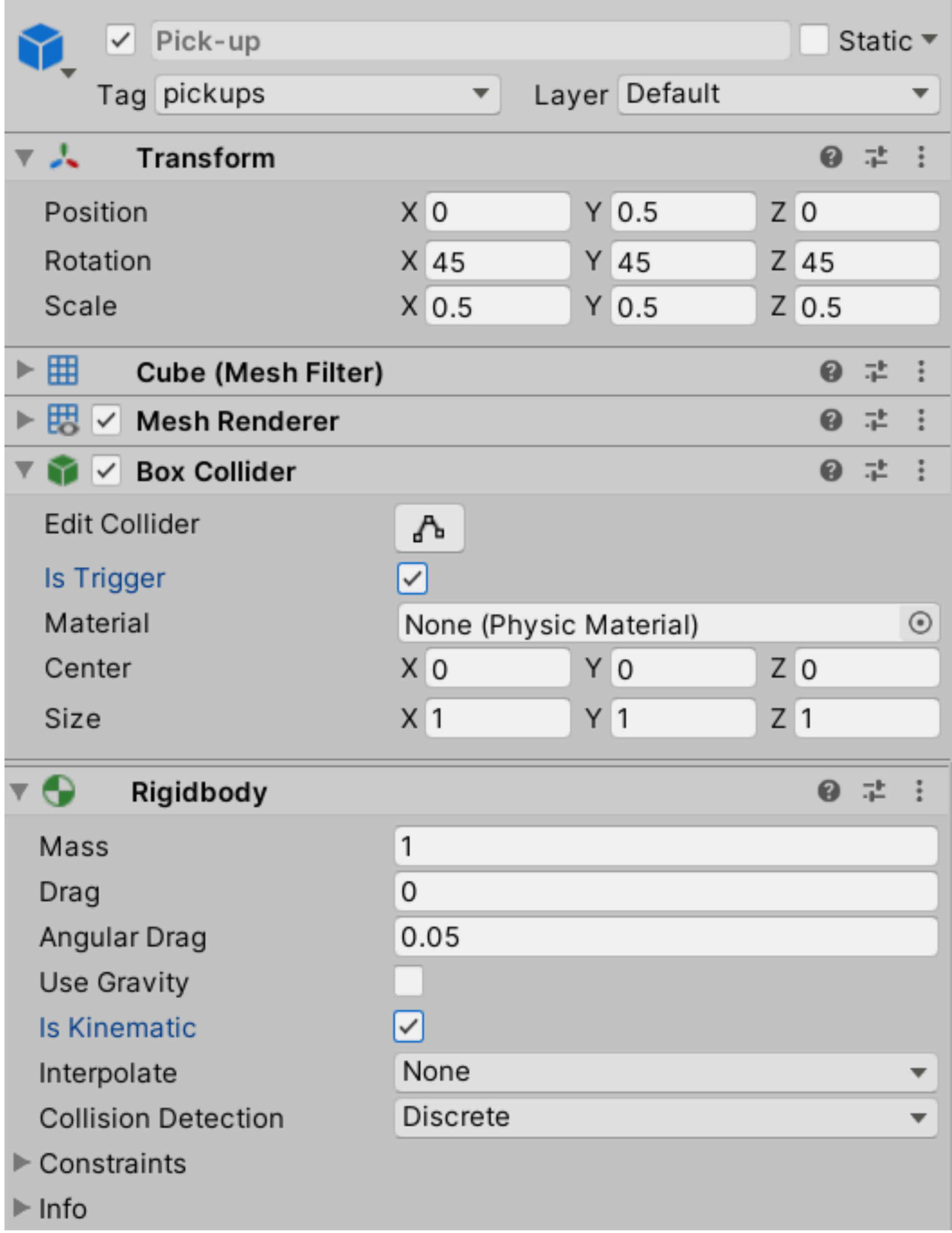
Rotator.cs



```
0 references
public class Rotator : MonoBehaviour
{
    // Start is called before the first frame update
    0 references
    void Start()
    {

    }

    // Update is called once per frame
    0 references
    void Update()
    {
        // Rotate the game object that this script is attached to by 15 in the X axis,
        // 30 in the Y axis and 45 in the Z axis, multiplied by deltaTime in order to make it per second
        // rather than per frame.
        transform.Rotate (new Vector3 (15, 30, 45) * Time.deltaTime);
    }
}
```

edit Pick-up prefab

- add Rotator.cs
- In Box Collider component
    - check **Is Trigger**
- In Rigidbody component
    - uncheck **Use Gravity**
    - check **Is Kinematic**



☑ Pick-up □ Static

Tag pickups Layer Default

Transform

Position	X 0	Y 0.5	Z 0
Rotation	X 45	Y 45	Z 45
Scale	X 0.5	Y 0.5	Z 0.5

Cube (Mesh Filter)

☑ **Mesh Renderer**

☑ **Box Collider**

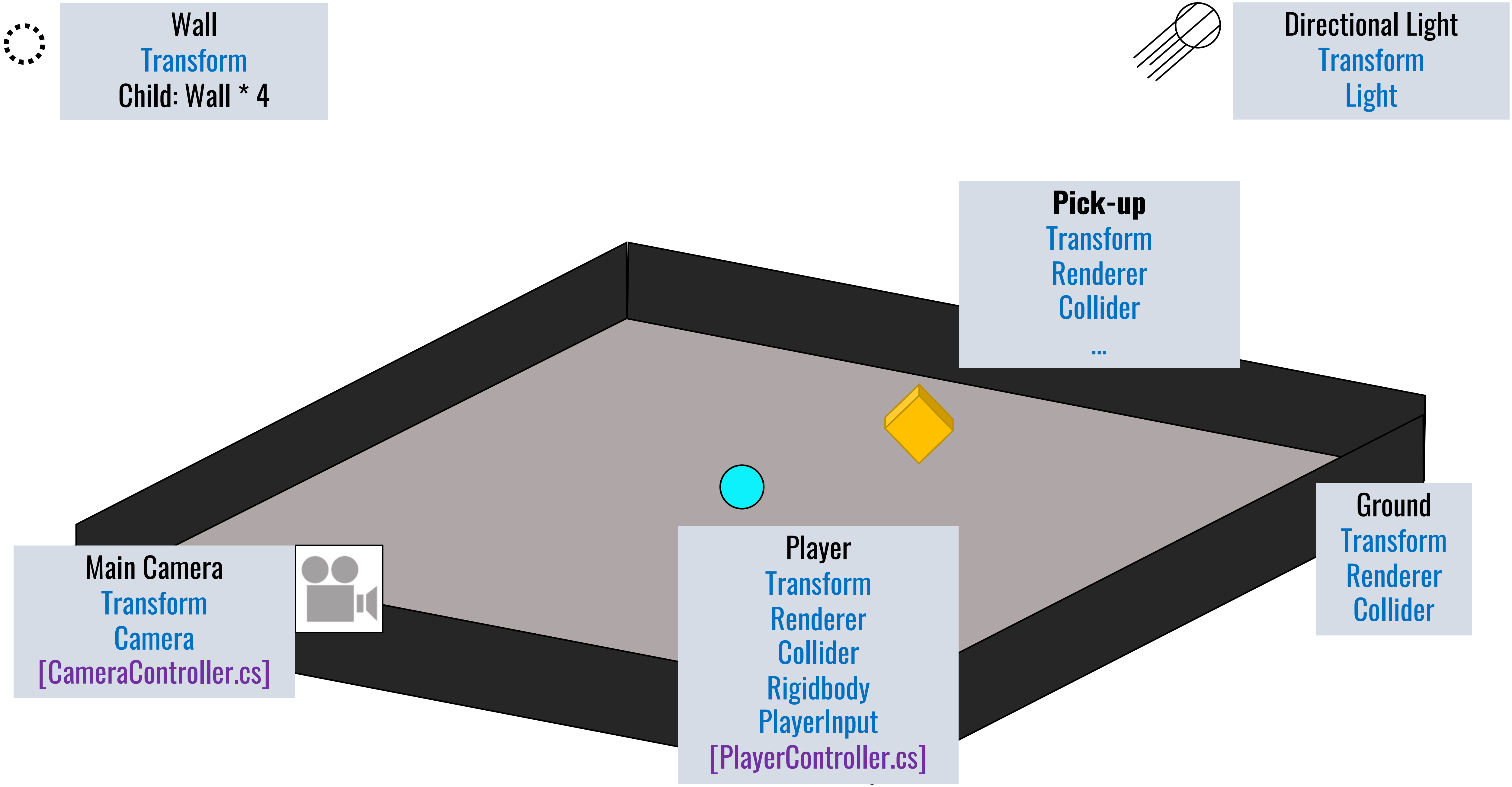
Edit Collider			
Is Trigger	☑		
Material	None (Physic Material)		
Center	X 0	Y 0	Z 0
Size	X 1	Y 1	Z 1

Rigidbody

Mass	1
Drag	0
Angular Drag	0.05
Use Gravity	□
Is Kinematic	☑
Interpolate	None
Collision Detection	Discrete

Constraints

Info



Wall
Transform
Child: Wall * 4

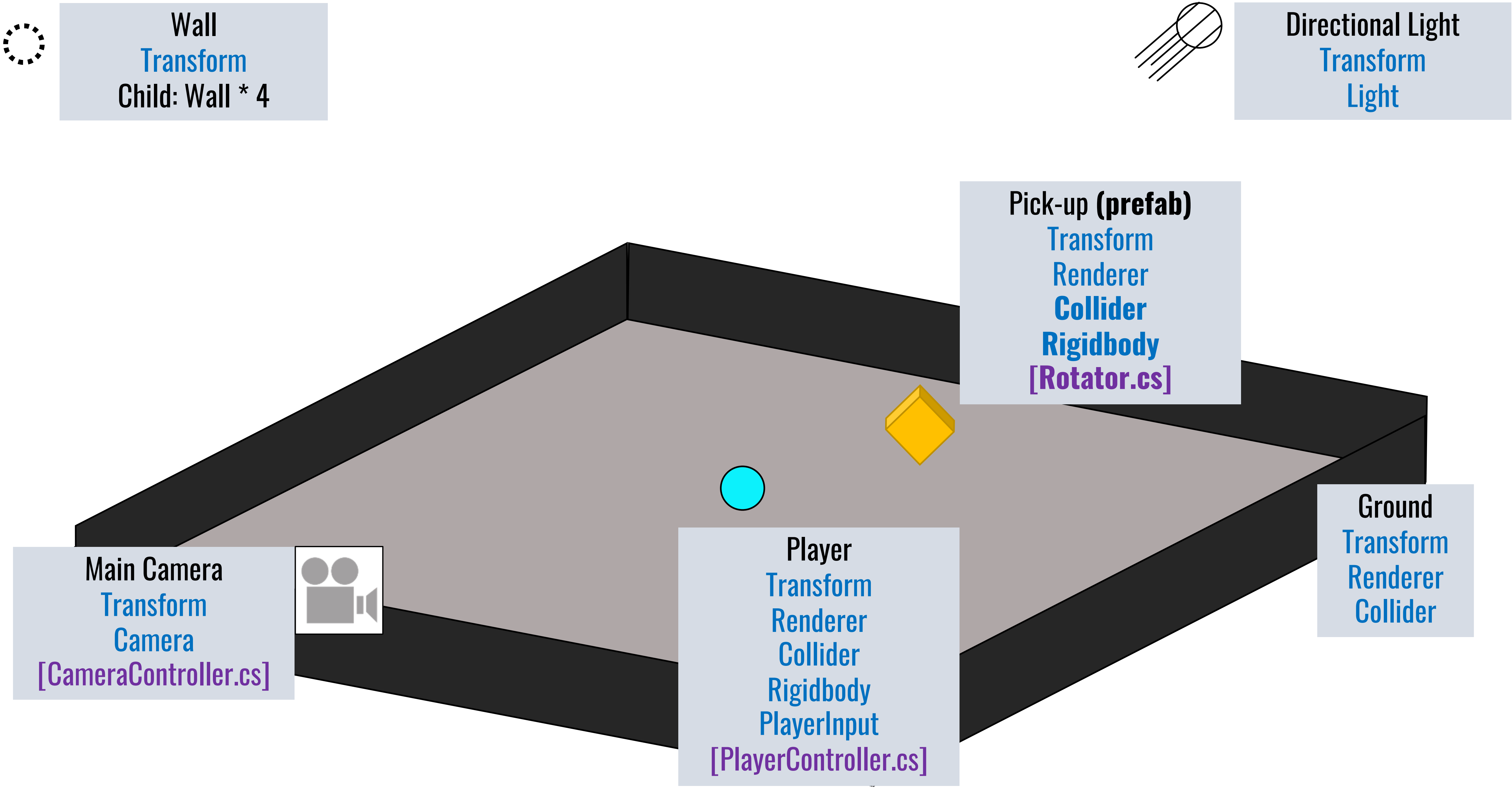
Directional Light
Transform
Light

Pick-up
Transform
Renderer
Collider
...

Ground
Transform
Renderer
Collider

Main Camera
Transform
Camera
[CameraController.cs]

Player
Transform
Renderer
Collider
Rigidbody
PlayerInput
[PlayerController.cs]



Wall
Transform
Child: Wall * 4

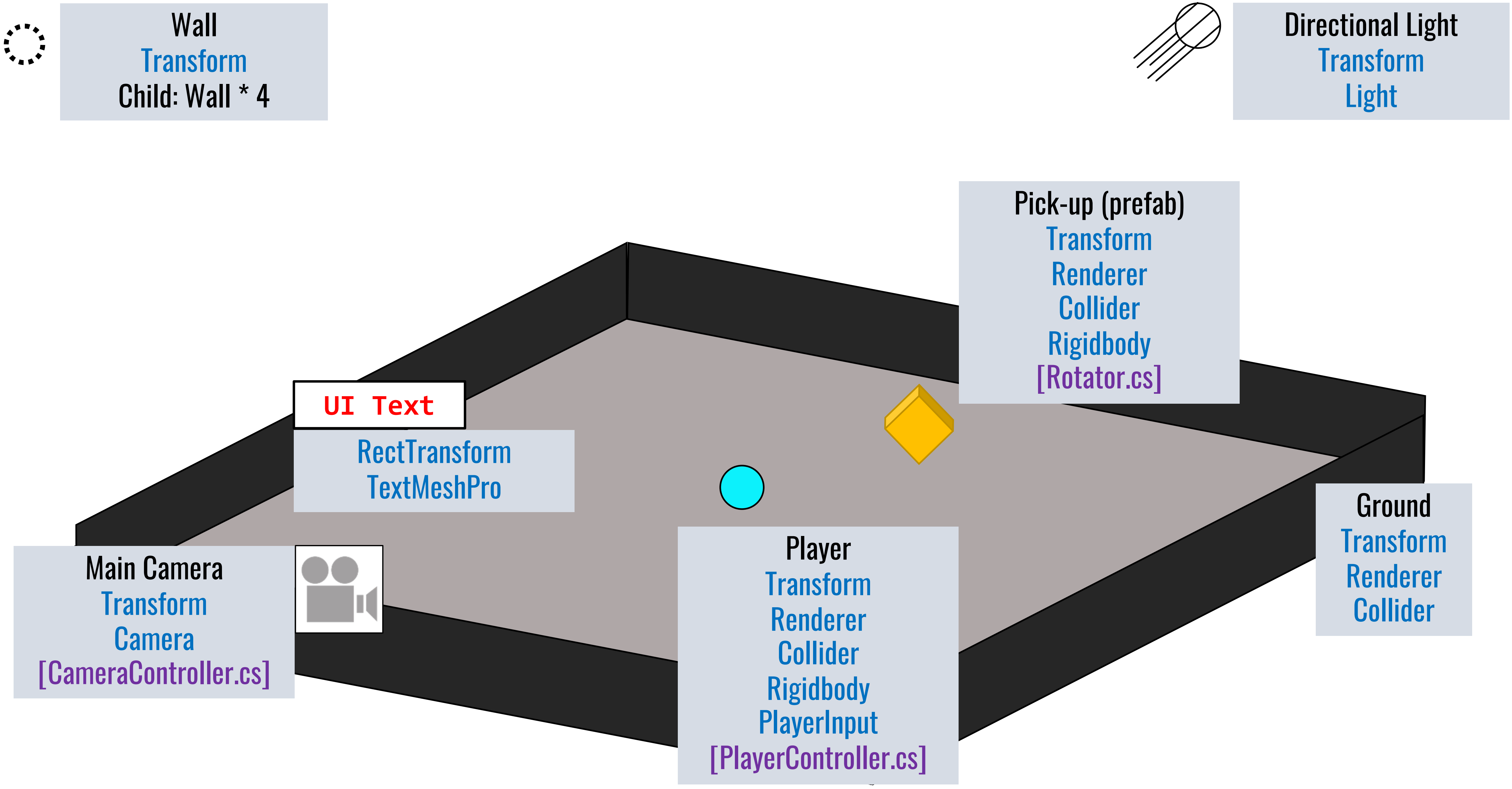
Directional Light
Transform
Light

Pick-up **(prefab)**
Transform
Renderer
Collider
Rigidbody
[Rotator.cs]

Main Camera
Transform
Camera
[CameraController.cs]

Player
Transform
Renderer
Collider
Rigidbody
PlayerInput
[PlayerController.cs]

Ground
Transform
Renderer
Collider



Wall
Transform
Child: Wall * 4

Directional Light
Transform
Light

Pick-up (prefab)
Transform
Renderer
Collider
Rigidbody
[Rotator.cs]

UI Text
RectTransform
TextMeshPro

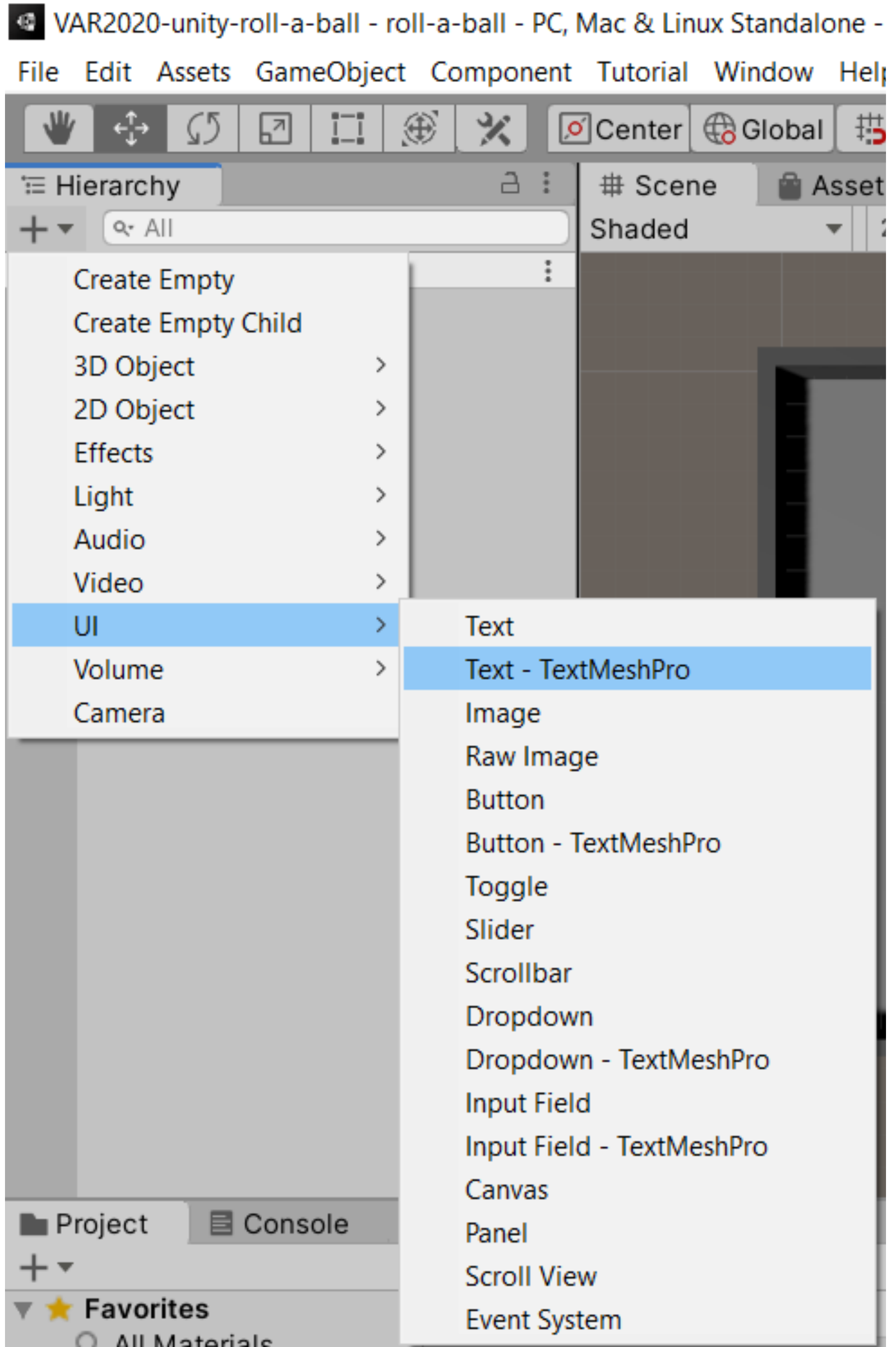
Main Camera
Transform
Camera
[CameraController.cs]

Player
Transform
Renderer
Collider
Rigidbody
PlayerInput
[PlayerController.cs]

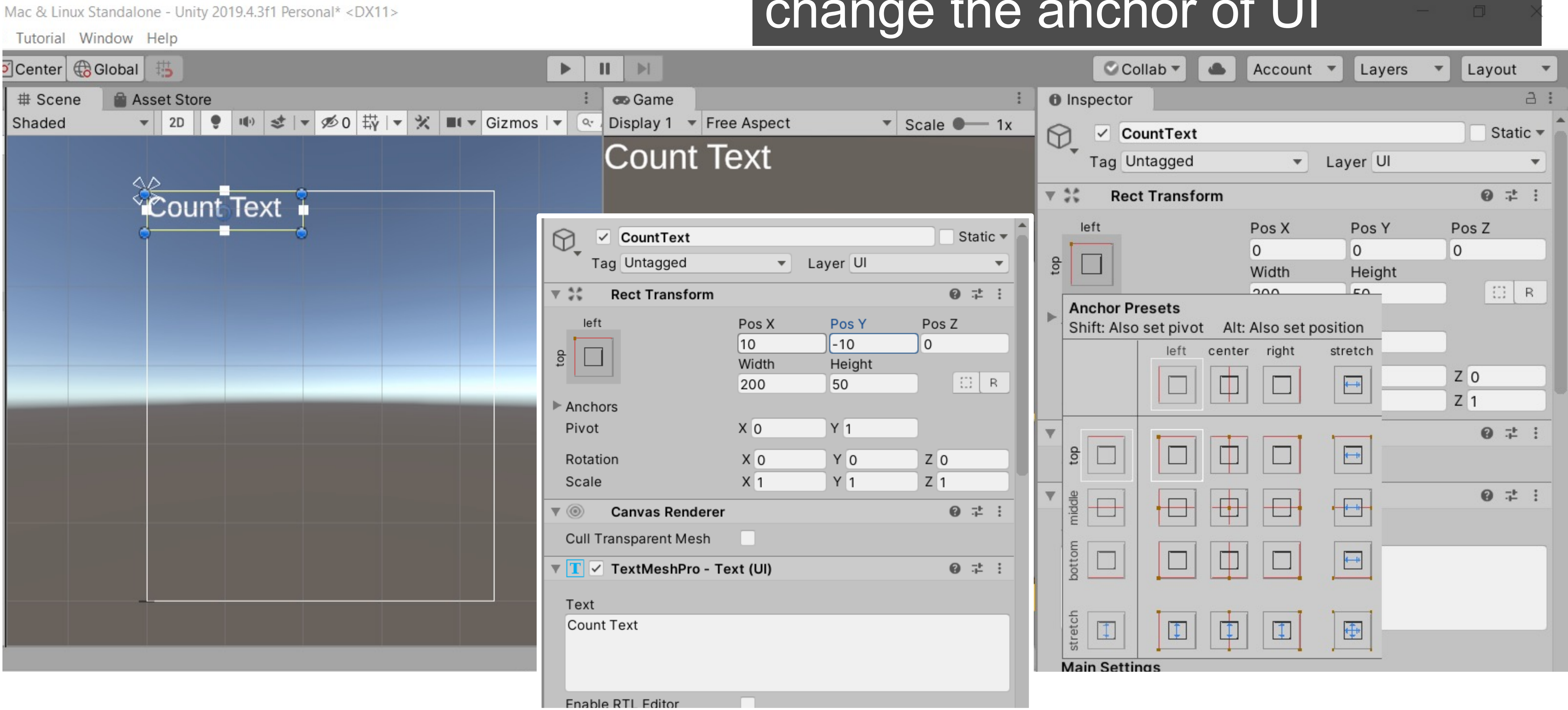
Ground
Transform
Renderer
Collider

UI: Count text

- Create UI
- Select Text - TextMeshPro



Shift + Alt and select upper left > change the anchor of UI



CountText Static
Tag Untagged Layer UI

Rect Transform

left / top

Pos X	Pos Y	Pos Z
10	-10	0
Width	Height	
200	50	

Anchors
Pivot X 0 Y 1
Rotation X 0 Y 0 Z 0
Scale X 1 Y 1 Z 1

Canvas Renderer
Cull Transparent Mesh □

TextMeshPro - Text (UI)
Text
Count Text
Enable RTL Editor □

CountText Static
Tag Untagged Layer UI

Rect Transform

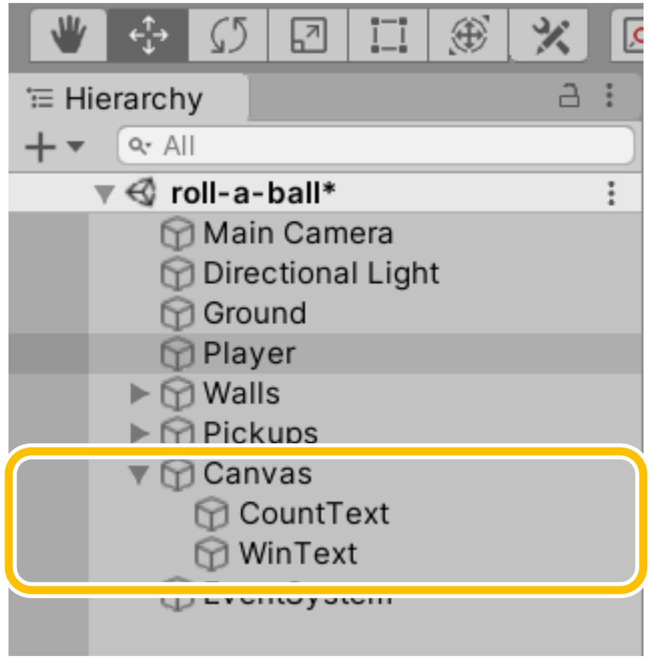
left / top

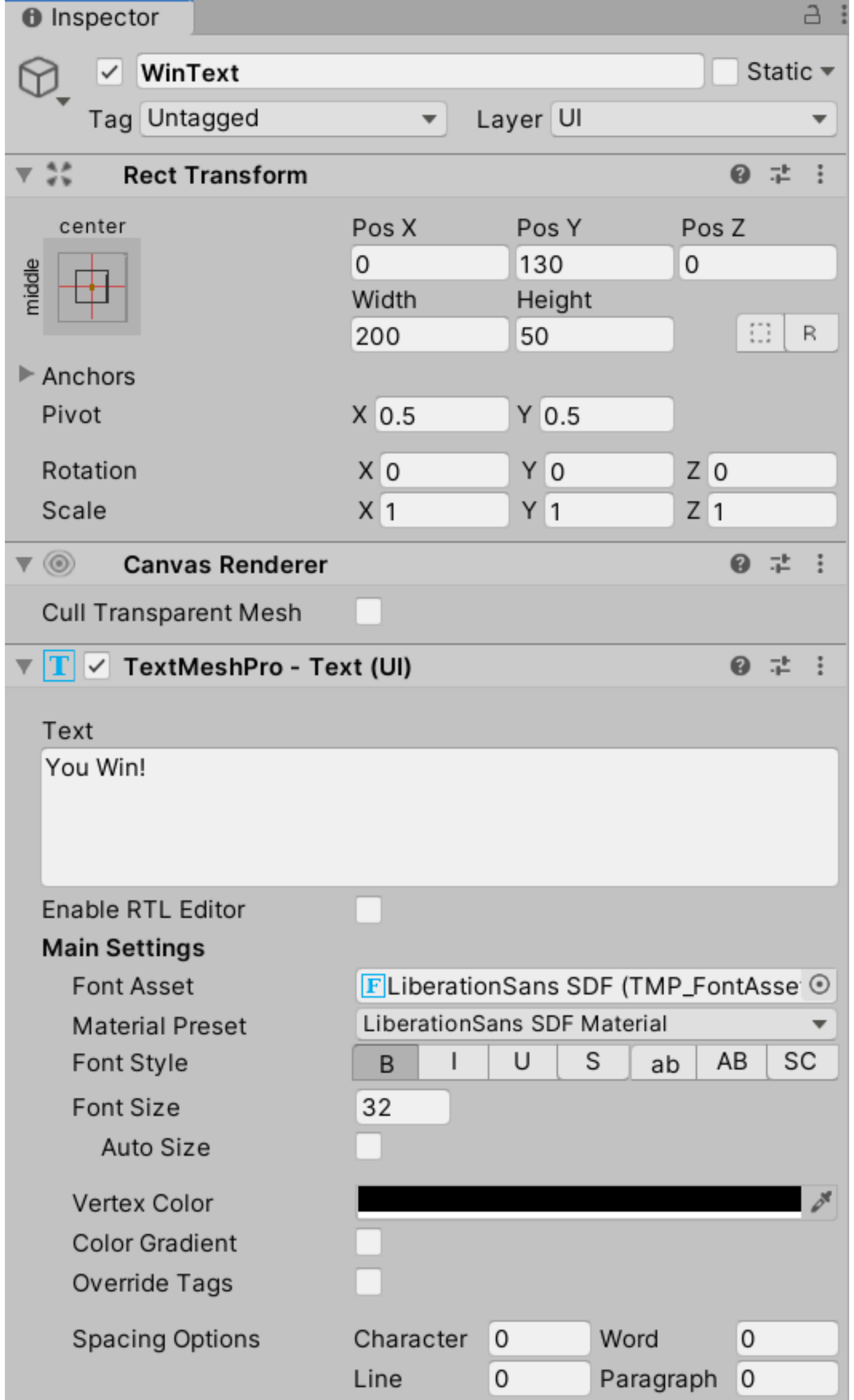
Pos X	Pos Y	Pos Z
0	0	0
Width	Height	
200	50	

Anchor Presets
Shift: Also set pivot Alt: Also set position

UI: Win text

- Create another TMP text
- Anchor to the center
- Change text





☑ WinText

Tag Untagged   Layer UI

Rect Transform

center / middle

Pos X	Pos Y	Pos Z
0	130	0
Width	Height	
200	50	

Anchors

Pivot X 0.5 Y 0.5

Rotation X 0 Y 0 Z 0

Scale X 1 Y 1 Z 1

Canvas Renderer

Cull Transparent Mesh □

☑ TextMeshPro - Text (UI)

Text: You Win!

Enable RTL Editor □

Main Settings

Font Asset: LiberationSans SDF (TMP_FontAsse

Material Preset: LiberationSans SDF Material

Font Style: B I U S ab AB SC

Font Size: 32

Auto Size □

Vertex Color

Color Gradient □

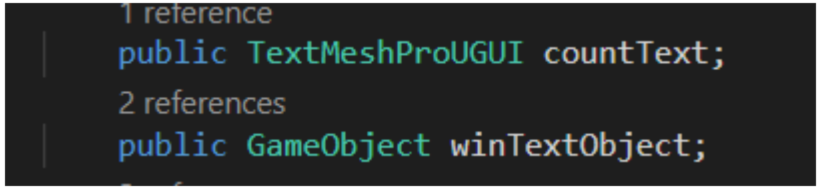
Override Tags □

Spacing Options: Character 0 Word 0 Line 0 Paragraph 0

back to PlayerController.cs

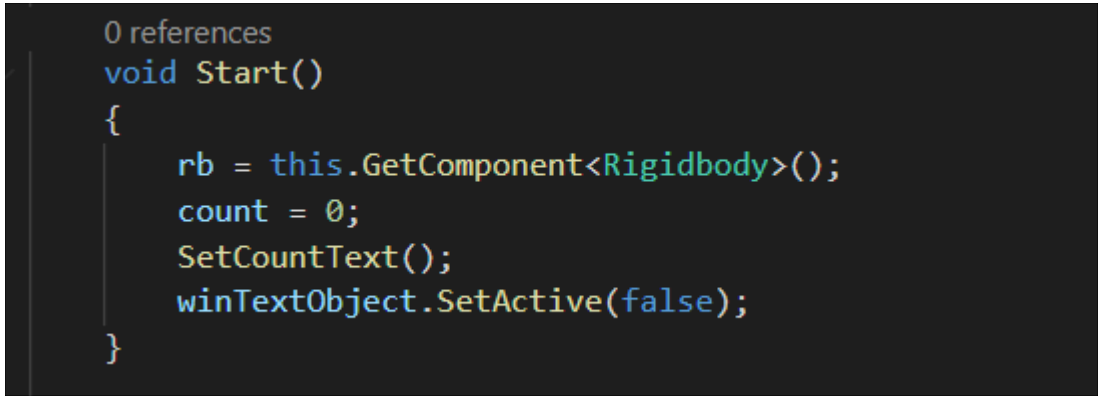
1. namespace: `using TMPro;` and global variables
2. Reset count and disable WinText in the beginning.
3. A function the update the CountText.

1



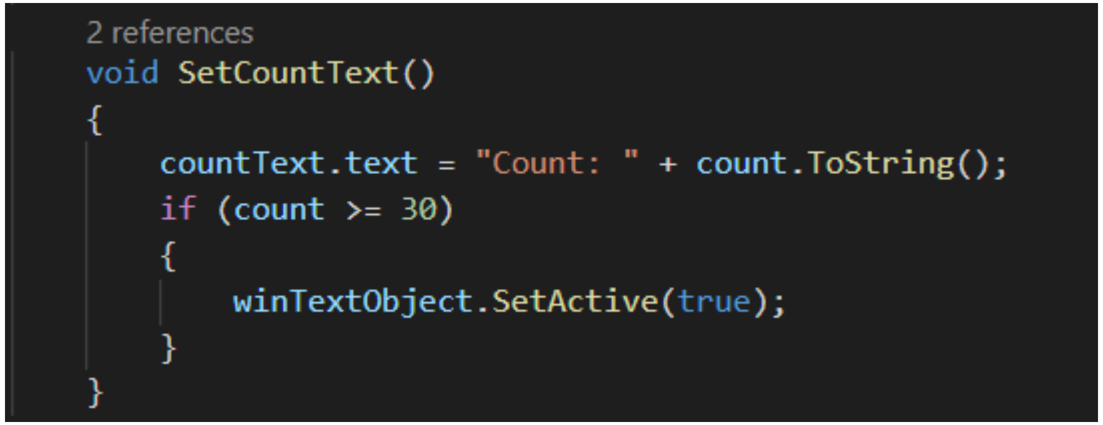
```
1 reference
public TextMeshProUGUI countText;
2 references
public GameObject winTextObject;
```

2



```
0 references
void Start()
{
    rb = this.GetComponent<Rigidbody>();
    count = 0;
    SetCountText();
    winTextObject.SetActive(false);
}
```

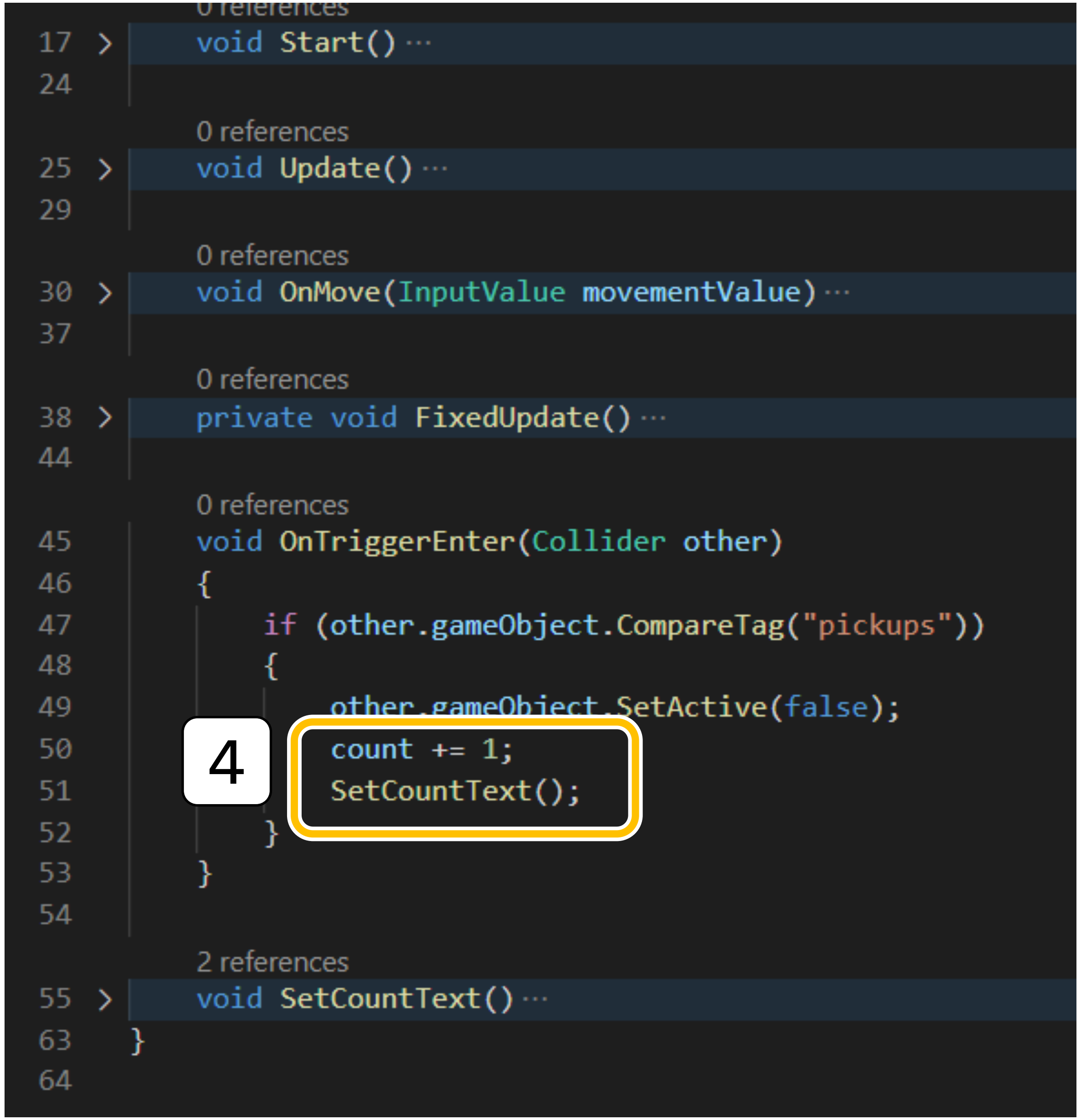
3



```
2 references
void SetCountText()
{
    countText.text = "Count: " + count.ToString();
    if (count >= 30)
    {
        winTextObject.SetActive(true);
    }
}
```

back to PlayerController.cs

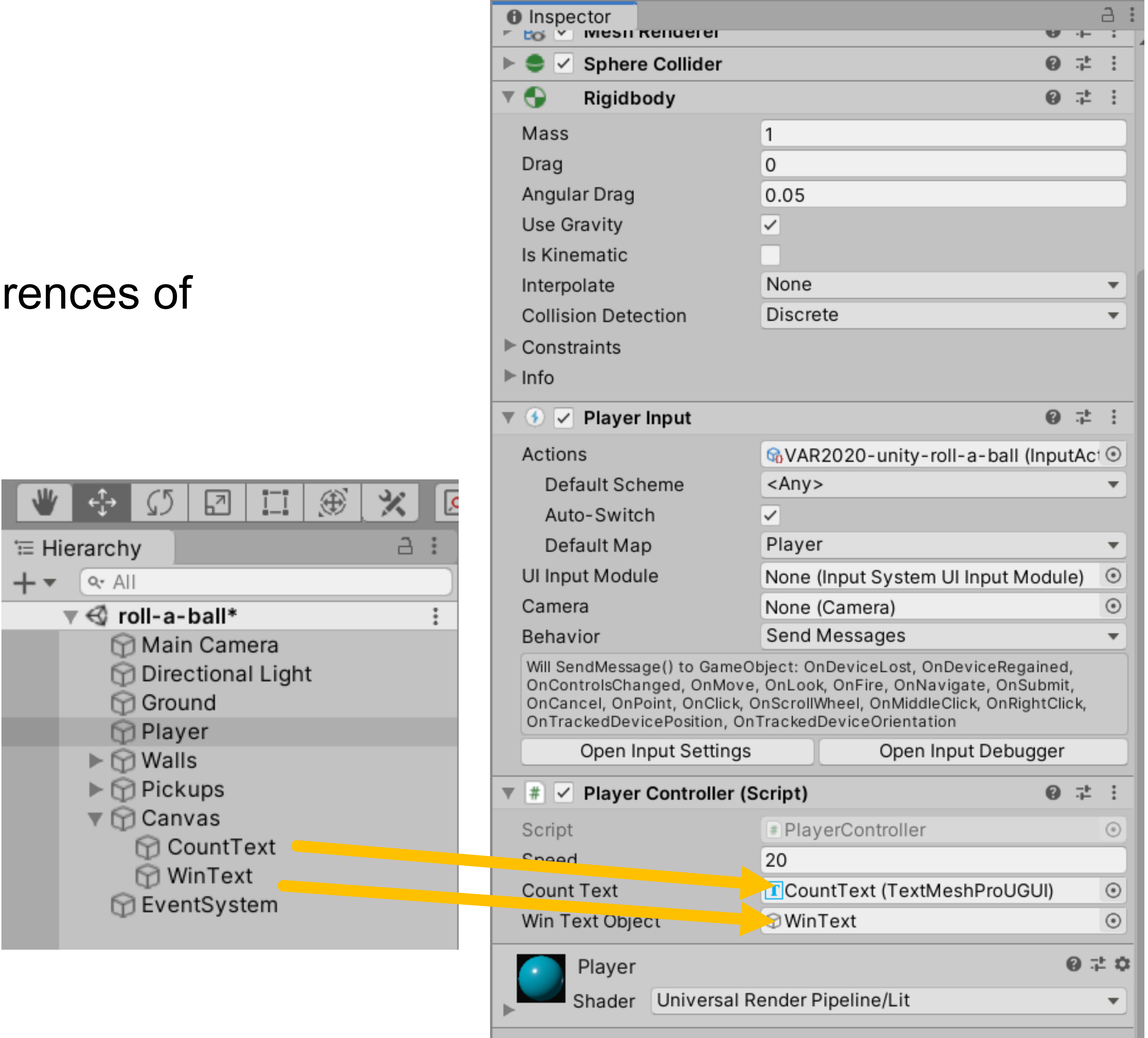
4. When Player hits a pick-up, increase the count and update the CountText



```
    0 references
    void Start() ···

    0 references
    void Update() ···

    0 references
    void OnMove(InputValue movementValue) ···

    0 references
    private void FixedUpdate() ···

    0 references
    void OnTriggerEnter(Collider other)
    {
        if (other.gameObject.CompareTag("pickups"))
        {
            other.gameObject.SetActive(false);
            count += 1;
            SetCountText();
        }
    }

    2 references
    void SetCountText() ···
}

```

In the scene

- Drag texts to the references of PlayerController



Sphere Collider ☑

Rigidbody

Mass: 1
Drag: 0
Angular Drag: 0.05
Use Gravity: ☑
Is Kinematic: □
Interpolate: None
Collision Detection: Discrete
Constraints
Info

Player Input ☑

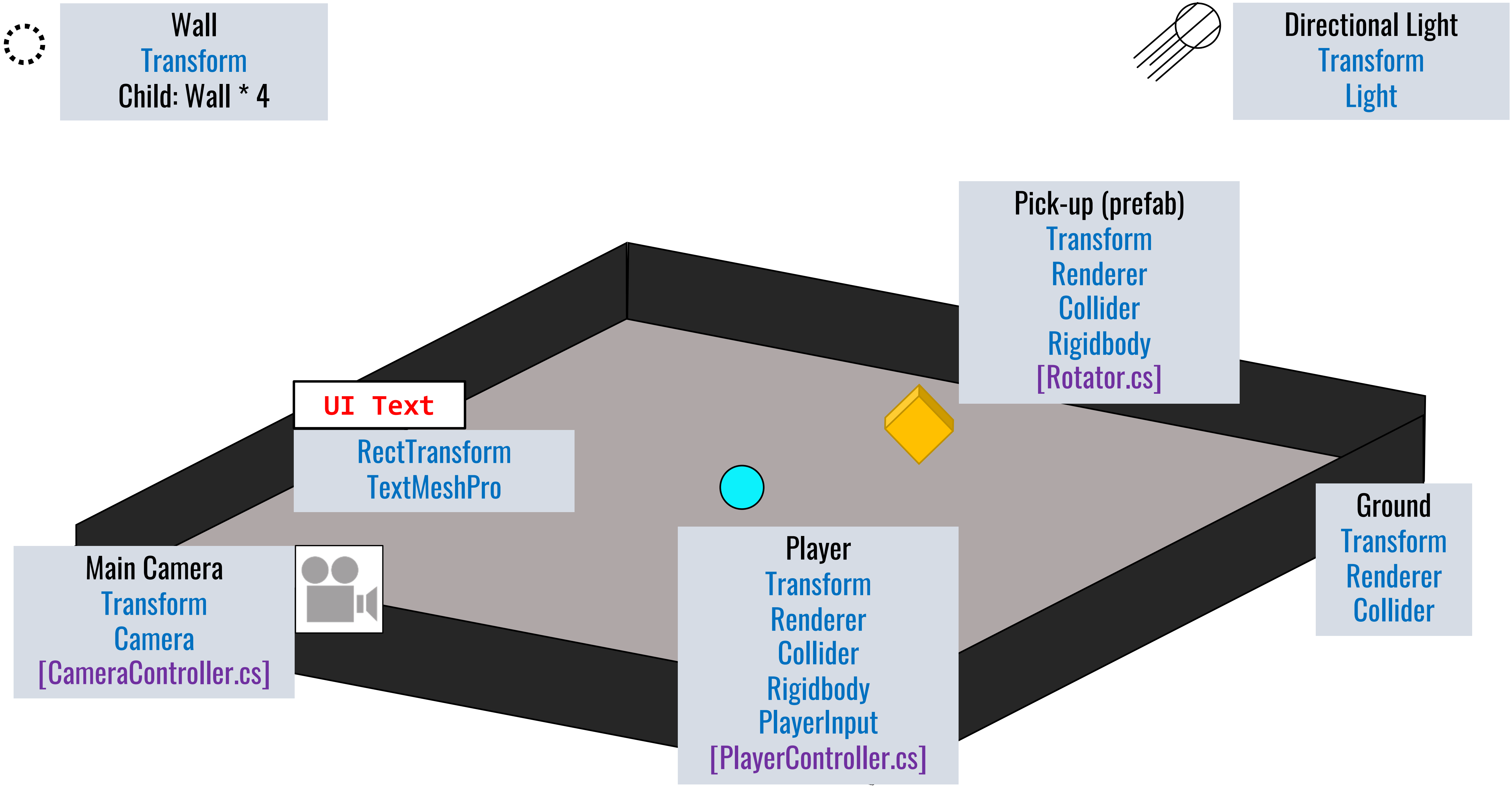
Actions: VAR2020-unity-roll-a-ball (InputAct
Default Scheme: <Any>
Auto-Switch: ☑
Default Map: Player
UI Input Module: None (Input System UI Input Module)
Camera: None (Camera)
Behavior: Send Messages

Will SendMessage() to GameObject: OnDeviceLost, OnDeviceRegained, OnControlsChanged, OnMove, OnLook, OnFire, OnNavigate, OnSubmit, OnCancel, OnPoint, OnClick, OnScrollWheel, OnMiddleClick, OnRightClick, OnTrackedDevicePosition, OnTrackedDeviceOrientation

Player Controller (Script) ☑

Script: PlayerController
Speed: 20
Count Text: CountText (TextMeshProUGUI)
Win Text Object: WinText

Player
Shader: Universal Render Pipeline/Lit



Wall
Transform
Child: Wall * 4

Directional Light
Transform
Light

Pick-up (prefab)
Transform
Renderer
Collider
Rigidbody
[Rotator.cs]

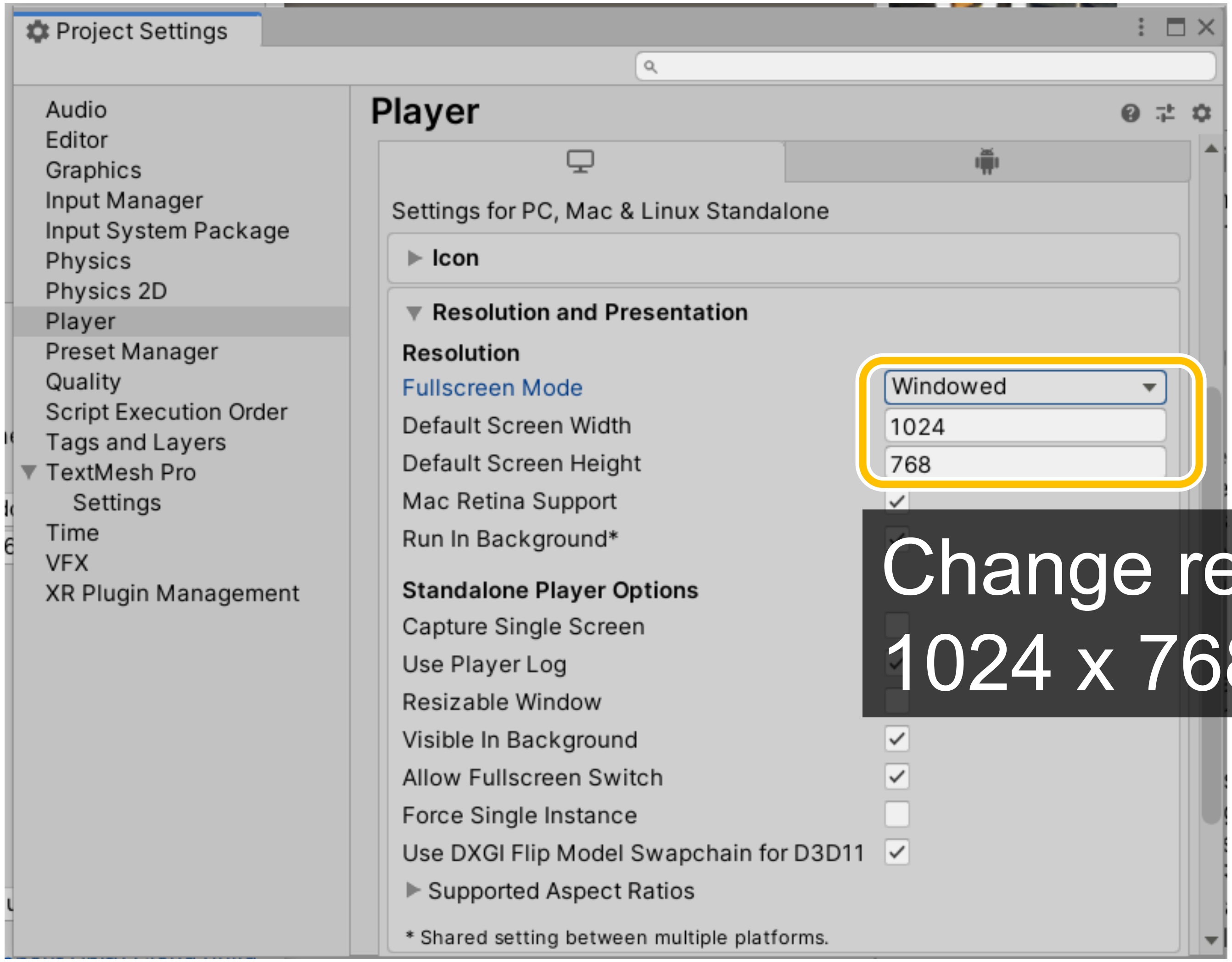
UI Text
RectTransform
TextMeshPro

Main Camera
Transform
Camera
[CameraController.cs]

Player
Transform
Renderer
Collider
Rigidbody
PlayerInput
[PlayerController.cs]

Ground
Transform
Renderer
Collider

Edit > Project Setting > Player



Player

Settings for PC, Mac & Linux Standalone

▶ Icon

▼ Resolution and Presentation

Resolution

Fullscreen Mode: Windowed

Default Screen Width: 1024

Default Screen Height: 768

Mac Retina Support ✓

Run In Background*

Standalone Player Options

Capture Single Screen

Use Player Log

Resizable Window

Visible In Background ✓

Allow Fullscreen Switch ✓

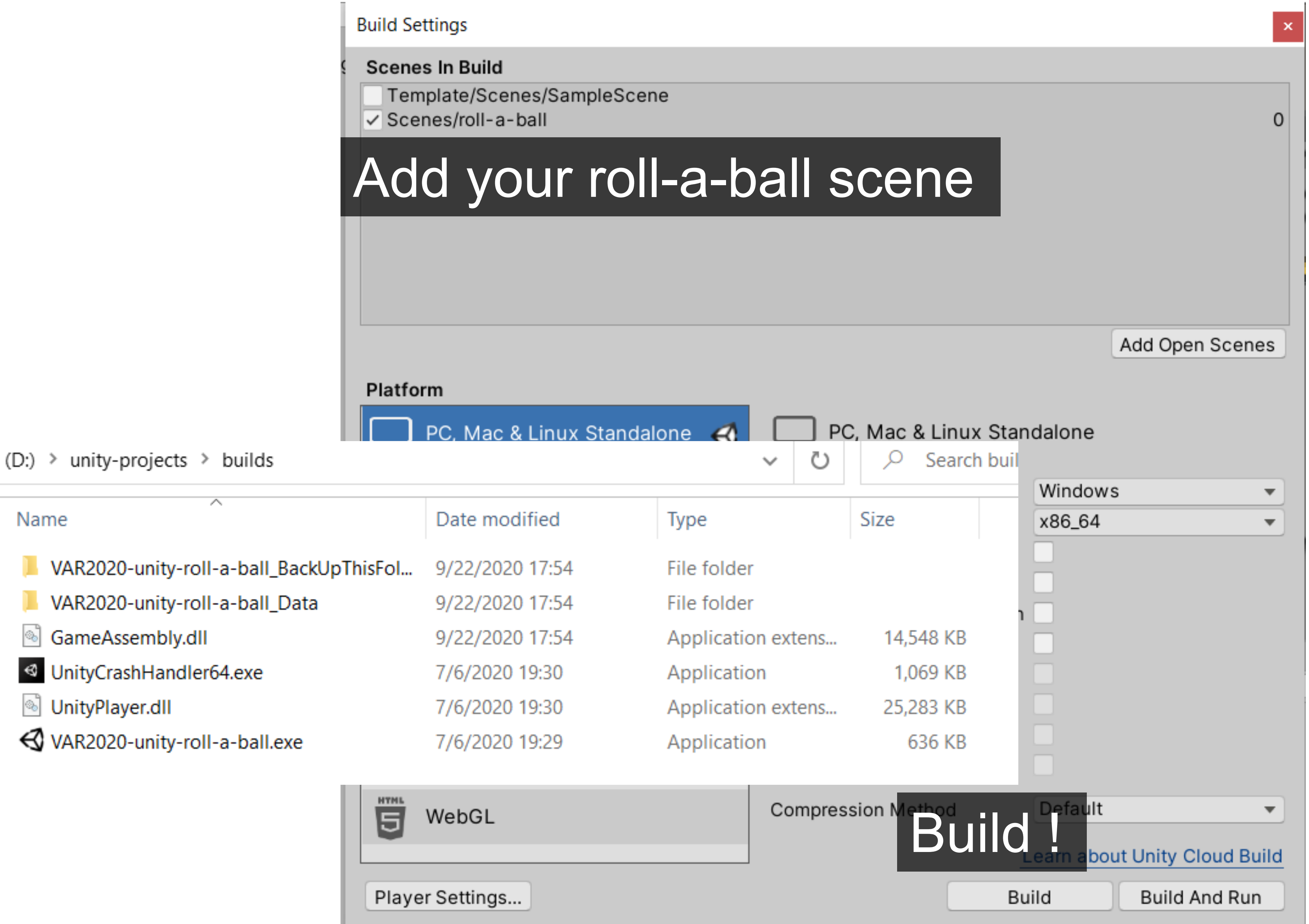
Force Single Instance □

Use DXGI Flip Model Swapchain for D3D11 ✓

▶ Supported Aspect Ratios

* Shared setting between multiple platforms.

Change resolution to 1024 x 768



Add your roll-a-ball scene

Scenes In Build

☐ Template/Scenes/SampleScene

☑ Scenes/roll-a-ball 0

Windows

x86_64

Compression Method: Default

(D:) › unity-projects › builds

Name	Date modified	Type	Size
VAR2020-unity-roll-a-ball_BackUpThisFol...	9/22/2020 17:54	File folder	
VAR2020-unity-roll-a-ball_Data	9/22/2020 17:54	File folder	
GameAssembly.dll	9/22/2020 17:54	Application extens...	14,548 KB
UnityCrashHandler64.exe	7/6/2020 19:30	Application	1,069 KB
UnityPlayer.dll	7/6/2020 19:30	Application extens...	25,283 KB
VAR2020-unity-roll-a-ball.exe	7/6/2020 19:29	Application	636 KB

Build !

Expected outcome

- A minimal roll-a-ball game.

References

https://gamedevbeginner.com/how-to-move-objects-in-unity/

Questions?