Virtual Reality Engine
Unity3D Theory

EPFL Immersive Interaction Group

Thibault Porssut

Dr. Ronan BOULIC

Outline

Theoretical Part:

- Available material for the VR Project
- One interface to rule them all (VR Game Engine)
- Unity: Current VR engine in IIG
- Interface (Unity Editor)
- Unity Elements
- Virtual Reality in Unity

Available material for the VR Project

- 3 Oculus Rifts
- 2 HTC Vives
- 1 Katwalk VR + 1 HTC Vive + 3 Vive Trackers
- 1 Hololens
- 1 GearVR + Samsung Galaxy S7
- Some Oculus Go
- Some Cardboards
- 6 Vive Trackers
- 2 Kinects V2
- 5 Leap Motions

Available material for the VR Project

- VR Headset

Oculus Rift

HTC Vive

- AR Headset

Hololens

- Cardboard Headset

Cardboard

GearVR + Samsung Galaxy S7

Oculus Go

Available material for the VR Project

- Tracking System

Vive Tracker

Kinect V2

Leap Motions

- Treadmill

Katwalk VR

Programming Assignment

- **1 programming assignment**: a real-time physically-based application with tracking done with Unity; this year the project is to be done in ***groups of 3*** (**40%**), during **weeks 7-12**, demo and project grading on **week 13**.
- Select among projects given or propose a project
- Material only available in the lab: KatwalkVR, HTC Vive, Oculus, Hololens
- The other material can be borrowed.
- Each device can be used by two groups.

One interface to rule them all (VR Game Engine)

- A system designed for development of VR scenes
- Provides a software framework that the users use to create interactive scenes / implement game scenarios
- Typical VR engines include:
  - 2D/3D graphics
  - Physics engine (collision detection)
  - Sound
  - Scripting
  - Animation
  - Networking
  - etc

One interface to rule them all (VR Game Engine)

- Open Source VR engine
  - OpenGL
  - Ogre3D
  - Panda3D
  - OpenSceneGraph
  - Godot Engine
  - Etc.

One interface to rule them all (VR Game Engine)

- VR engine with license
  - Unity3D
  - Unreal Engine
  - 3Dvia Studio (Virtools 5.1 last version)
  - Cryengine
  - Worldviz
  - Amazon Lumberyard
  - etc

One interface to rule them all (VR Game Engine)

- Comparison

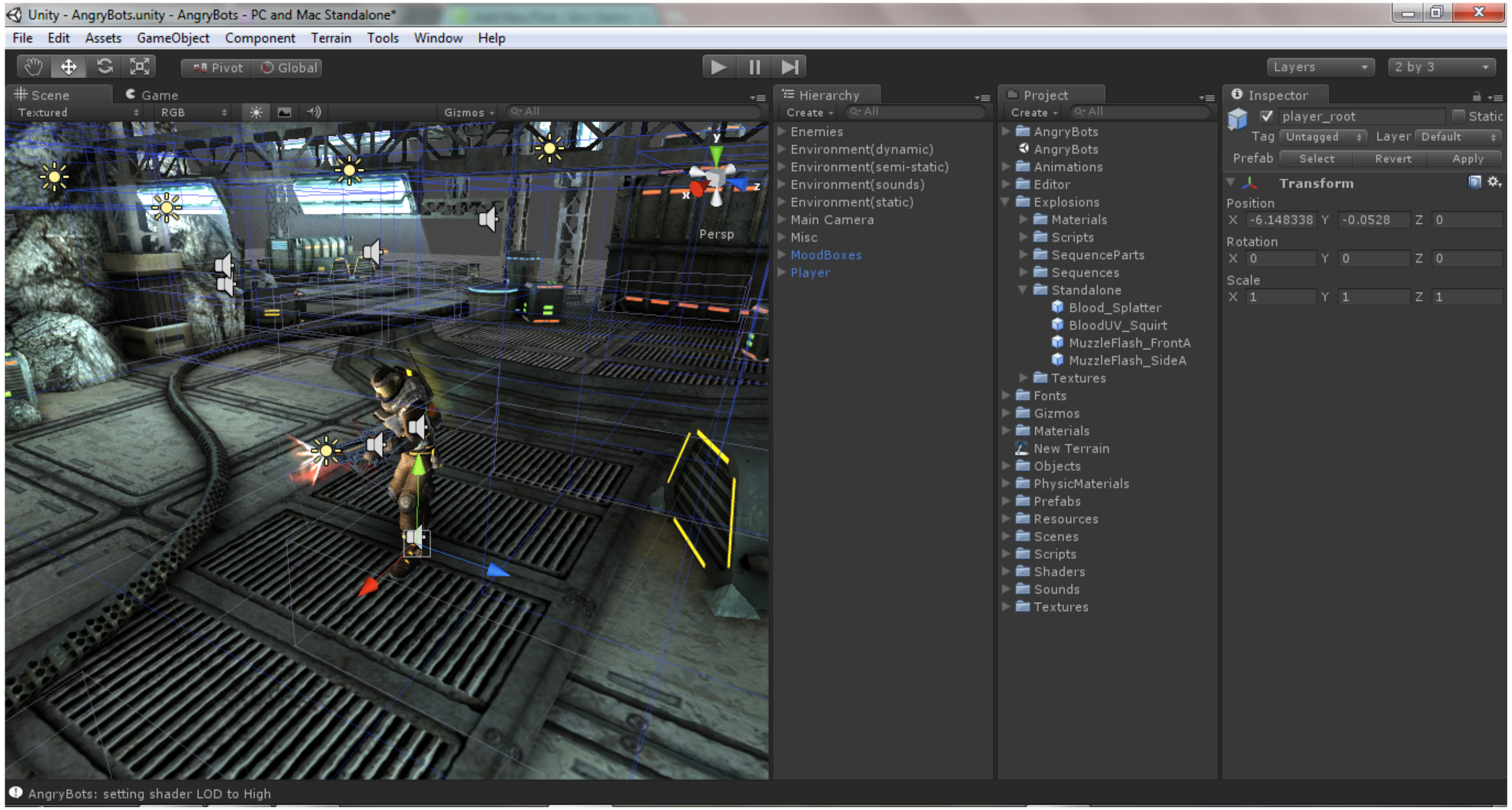
	Open-Source VR engine	VR engine with license
GUI (visual development)	No	Yes
Latest features availability (e.g. Geometry Shader)	Fast	Slow (Closed)
VR adaptation (e.g. HMD)	Libraries	Plug-ins (API)
Development	Time consuming	Time saving
Resources	Rich	Limited
Multi-platform development	Not limited	Limited
External device integration	Not easy (via APIs)	Easy (via Plug-ins)

Unity: Current VR engine in IIG

- Unity3D
  - Interface (GUI)
  - Powerful scripting
  - Multiple platforms supported
  - Features (animation, network sound, physics engine, etc...)
  - Support (forum)
  - Resources (asset store, demos, tutorial samples, etc…)
  - Not expensive and free for education/personal use

Unity: Current VR engine in IIG

- Unity3D - GUI



Unity - AngryBots.unity - AngryBots - PC and Mac Standalone*

☑ player_root □ Static

Tag Untagged Layer Default

Prefab

Transform

Position X -6.148338 Y -0.0528 Z 0

Rotation X 0 Y 0 Z 0

Scale X 1 Y 1 Z 1

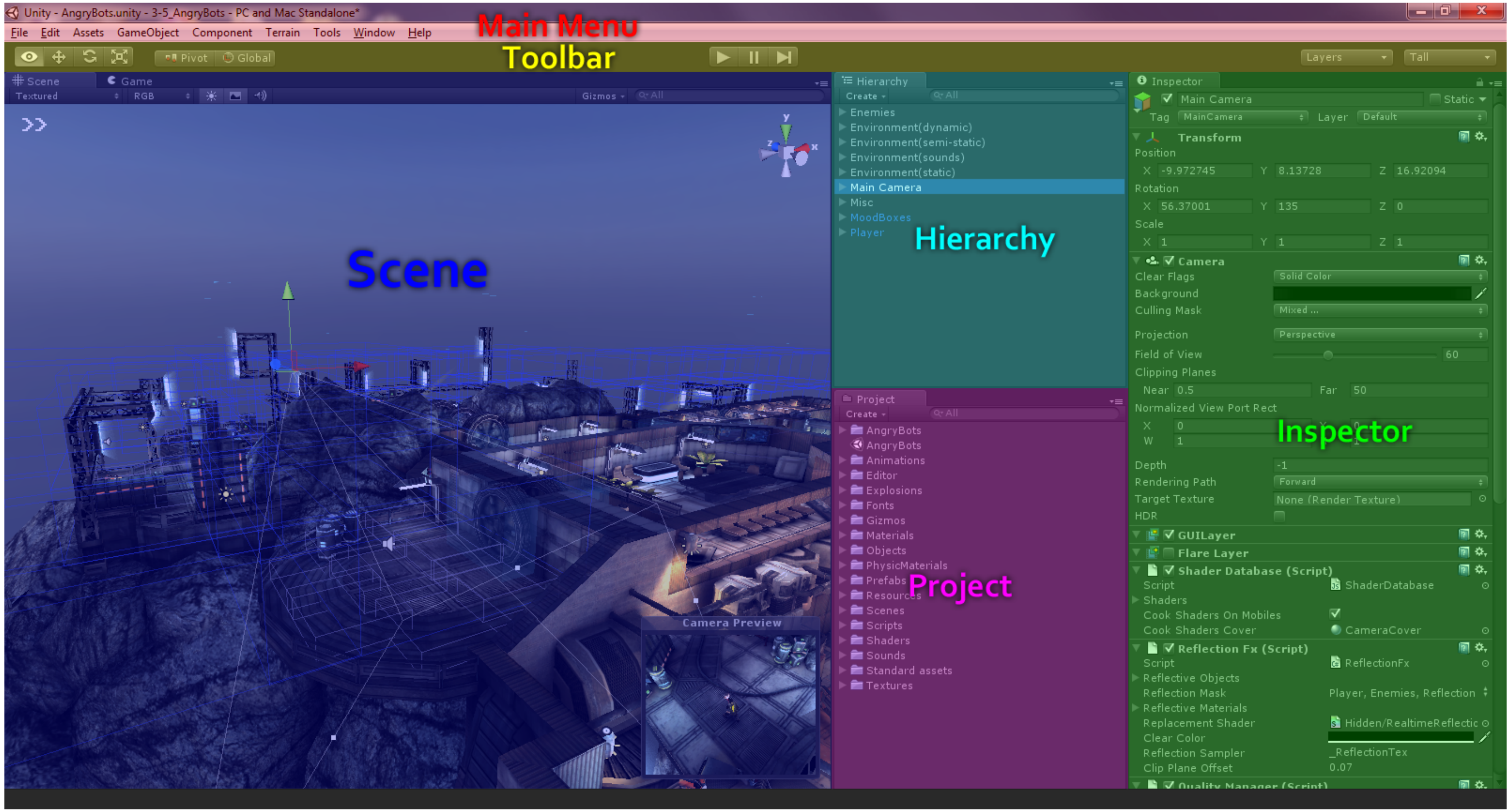
AngryBots: setting shader LOD to High

http://docs.unity3d.com/Documentation/Manual/UnityBasics.html

Unity: Current VR engine in IIG

- Intuitive tool
  - Low cost development system
  - 2D and 3D interactive content
  - Build-in physics engine
  - Multiplatform publishing
  - Quality control
  - Ready-made assets
  - Knowledge-sharing community

Interface (Unity Editor)



☑ Main Camera □ Static

Tag MainCamera Layer Default

Transform

Position X -9.972745 Y 8.13728 Z 16.92094

Rotation X 56.37001 Y 135 Z 0

Scale X 1 Y 1 Z 1

☑ **Camera**

Clear Flags: Solid Color

Background

Culling Mask: Mixed ...

Projection: Perspective

Field of View: 60

Clipping Planes Near 0.5 Far 50

Normalized View Port Rect X 0 Y 0 W 1 H 1

Depth: -1

Rendering Path: Forward

Target Texture: None (Render Texture)

HDR □

☑ **GUILayer**

□ **Flare Layer**

☑ **Shader Database (Script)**

Script: ShaderDatabase

Shaders

Cook Shaders On Mobiles ☑

Cook Shaders Cover: CameraCover

☑ **Reflection Fx (Script)**

Script: ReflectionFx

Reflective Objects

Reflection Mask: Player, Enemies, Reflection

Reflective Materials

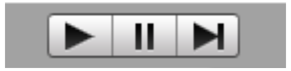
Replacement Shader: Hidden/RealtimeReflectic

Clear Color

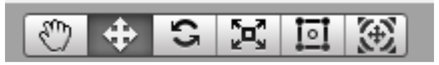
Reflection Sampler: _ReflectionTex

Clip Plane Offset: 0.07

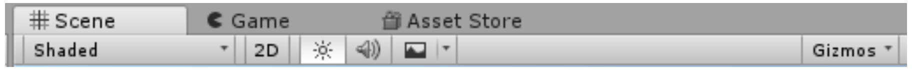
☑ **Quality Manager (Script)**



Play/Pause/Step (Test your game without compiling)



Grab/Translate/Rotate/Rescale your GameObject in your scene



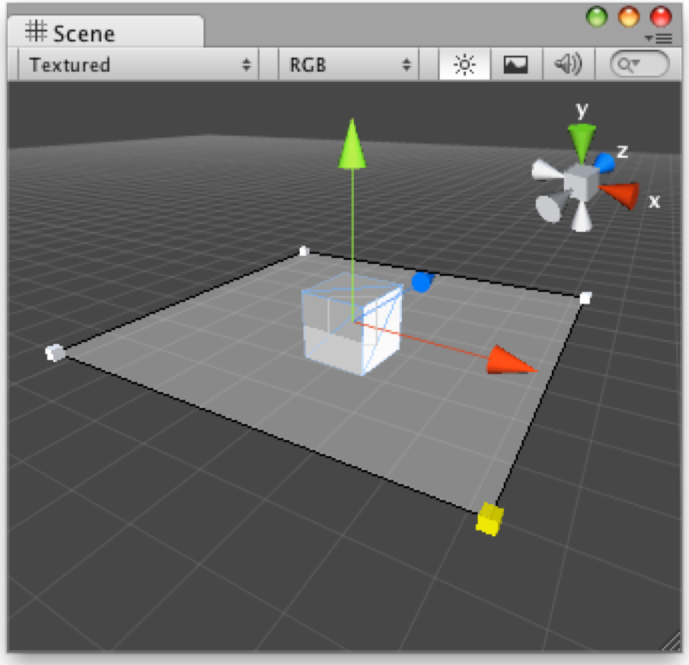
Several Tabs: Scene (Manipulate your objects)/Game (rendered on your camera)/Asset Store (download Unity packages)

Unity Elements: Overview

- Assets
- Scenes
- Game Objects
- Components
- Scripts
- Prefabs
- Interface

Unity Elements: Overview

- 3D coordinate system
- Local space/World space
- Camera and view port
- Polygons, edges, vertices, and meshes
- Materials, textures, and shaders
- Rigid Body physics
- Collision detection



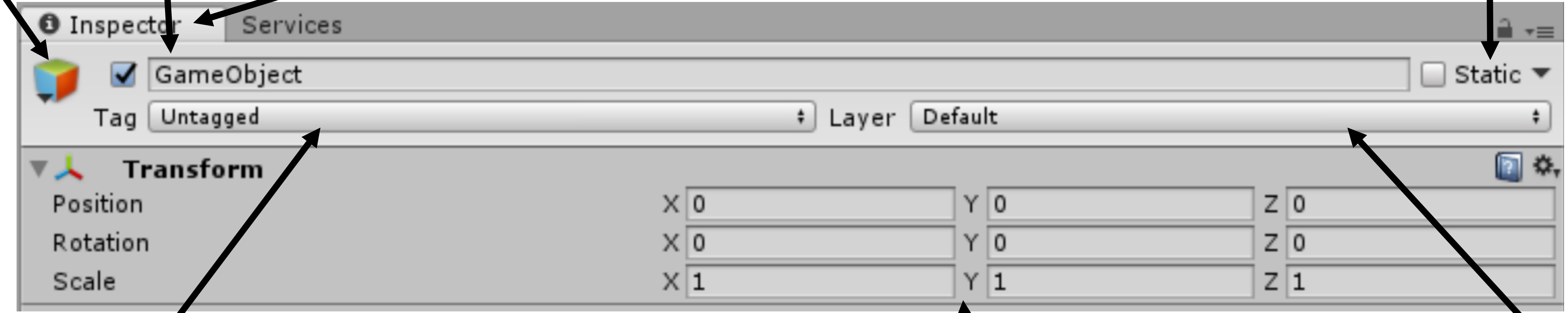
Unity Elements: GameObject

Name of the Object: Way to find it.

Static: Non-moving object (Useful for rendering optimization)

Icon to highlight your object in the scene.

Inspector: Access the GameObject properties.



☑ GameObject □ Static

Tag Untagged Layer Default

Transform

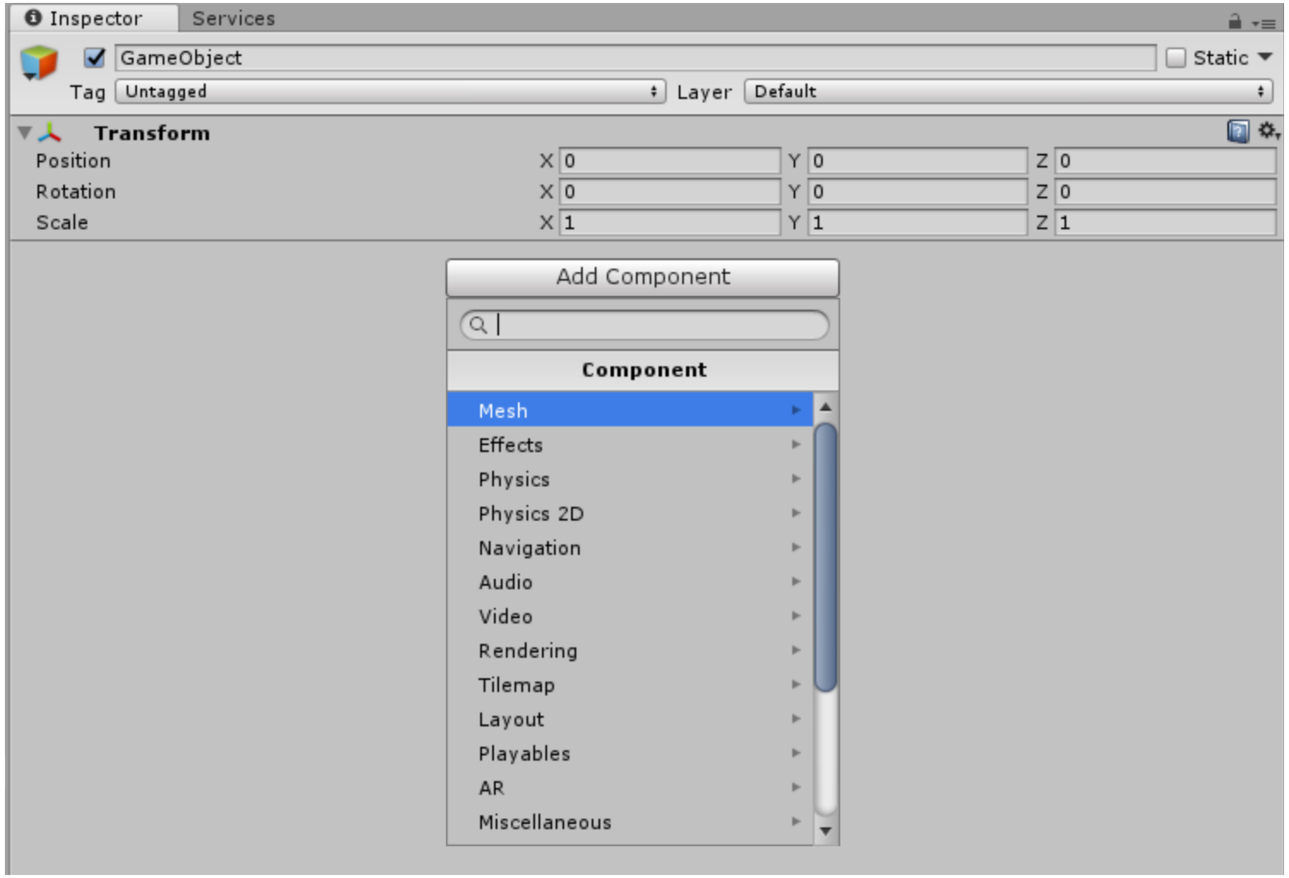
Position	X 0	Y 0	Z 0
Rotation	X 0	Y 0	Z 0
Scale	X 1	Y 1	Z 1

Tag: group membership (Useful for raycasting or finding a group of objects)

Layer: Access to the GameObject properties.

Transform: 3D coordinates (Position, Rotation and scale) left-handed coordinate system.

Unity Elements: Components



☑ GameObject □ 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

Mesh: Vertex of your 3D object (3D shape)

Effects: Special renderer effect (particle effect)

Physics: Gravity (Rigidbody), Collision (Collider)

Navigation: Pathfinding Algorithm (Nav Mesh Agent)

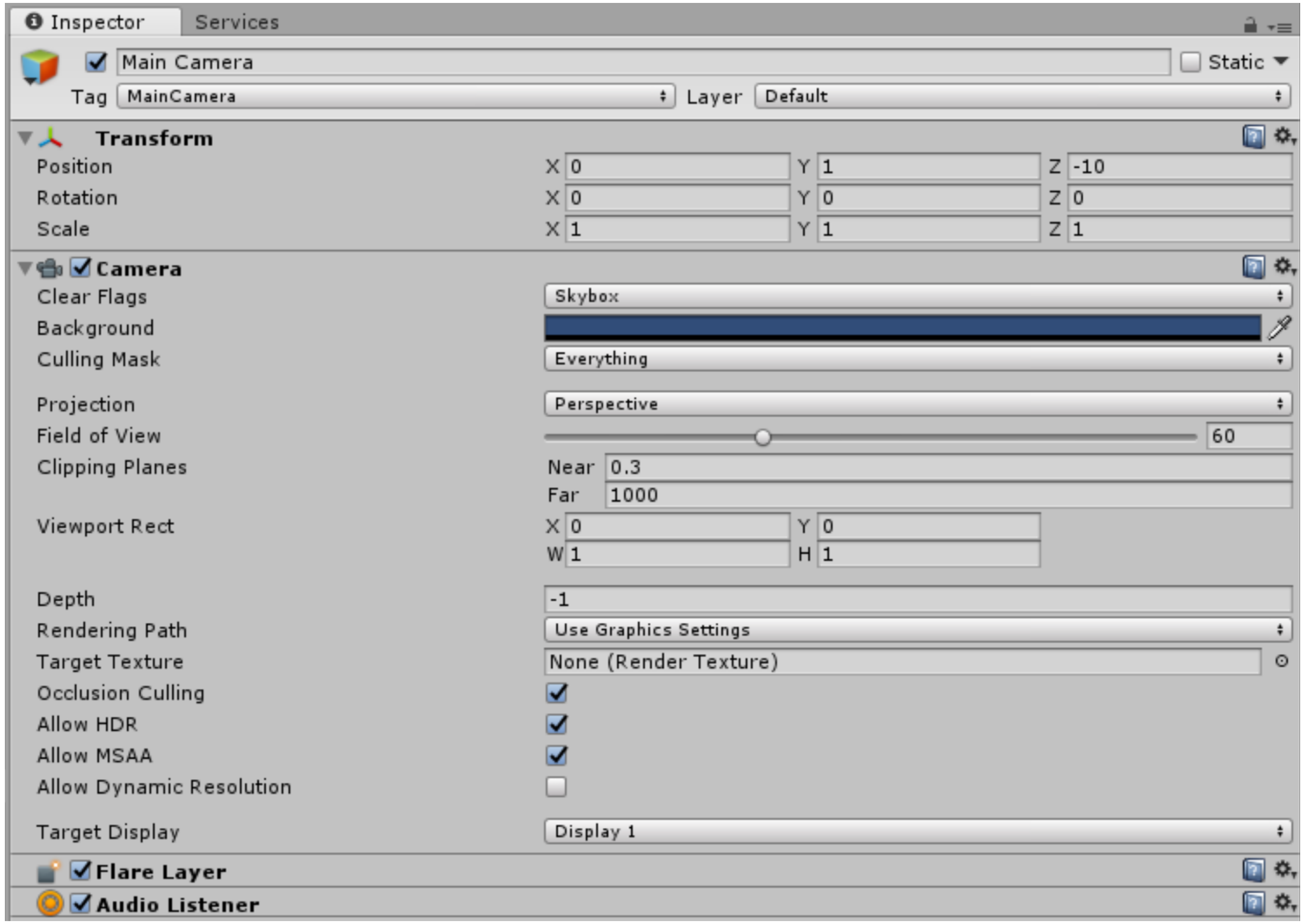
Audio: Audio Listener (your avatar), Audio Source (3D sound) and different Filters

Rendering: Camera and Skybox

Layout: Canvas (Menu in your game)

Miscellaneous: Terrain and Wind Zone.

Unity Elements: Main Camera



☑ 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**

Clear Flags	Skybox
Background	■
Culling Mask	Everything
Projection	Perspective
Field of View	60
Clipping Planes	Near 0.3 Far 1000
Viewport Rect	X 0 Y 0 W 1 H 1
Depth	-1
Rendering Path	Use Graphics Settings
Target Texture	None (Render Texture)
Occlusion Culling	☑
Allow HDR	☑
Allow MSAA	☑
Allow Dynamic Resolution	□
Target Display	Display 1

☑ **Flare Layer**

☑ **Audio Listener**

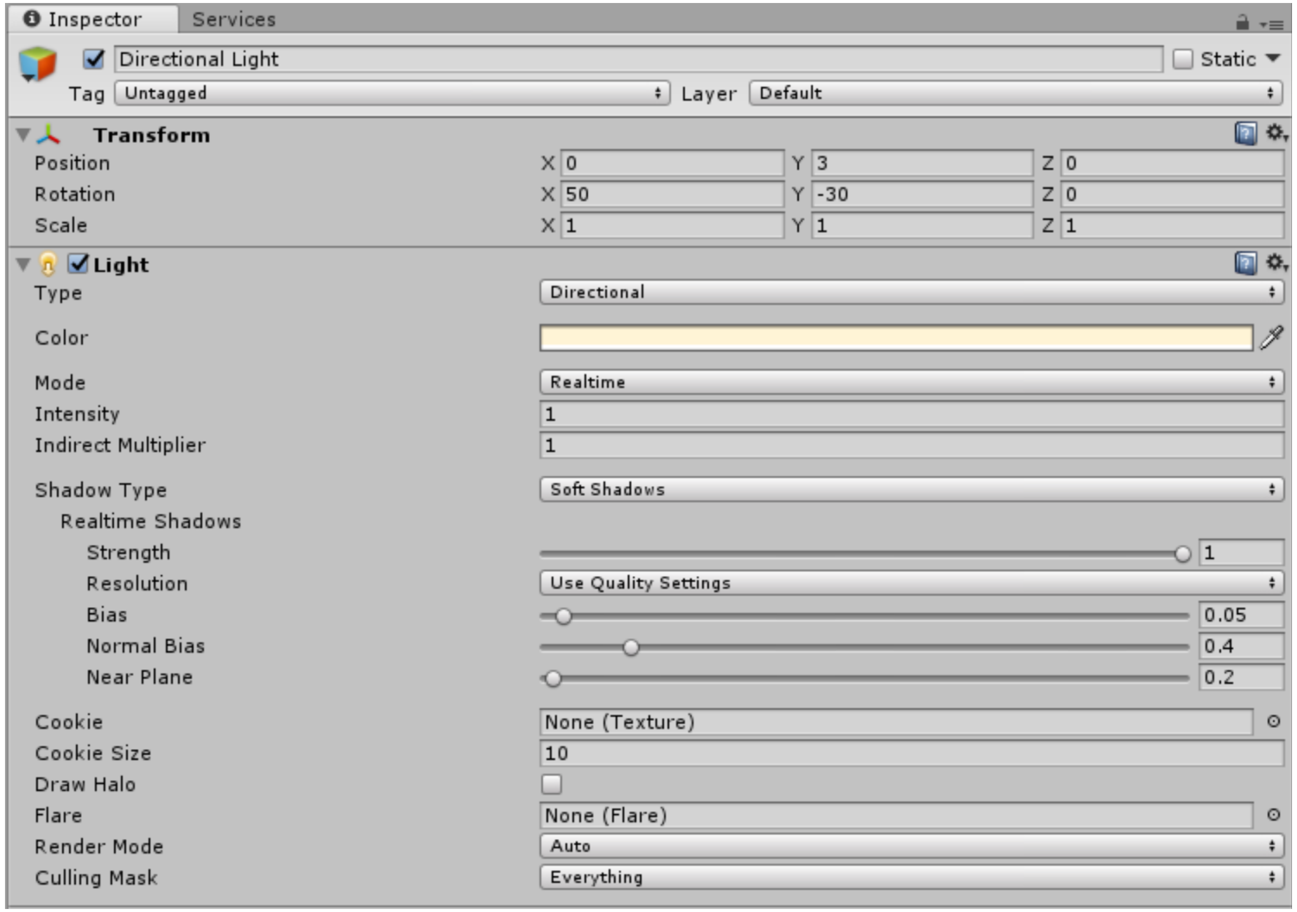
Tag: Main Camera (the default camera used by unity to render your game)

Background: The colour of the area when there is no Skybox

Field of View: The more important it is, the more you need to render.

Clipping Plane: The size of the area that needs to be rendered.

Unity Elements: Directional Light



Inspector Services

☑ Directional Light □ Static

Tag Untagged Layer Default

Transform

Position X 0 Y 3 Z 0

Rotation X 50 Y -30 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)

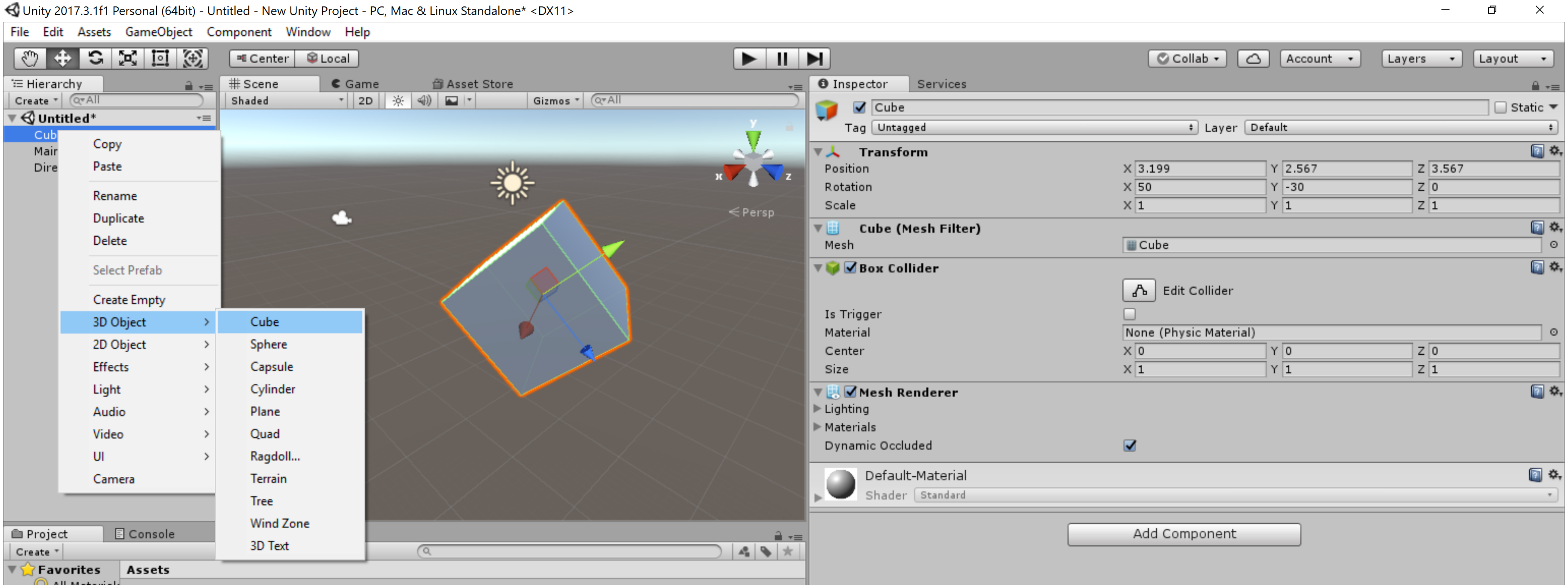
Render Mode Auto

Culling Mask Everything

By default, Unity creates one directional light but you can create several to make special effects:

- **Attach to an object**
- **Light some confined space**
- **Etc….**

Unity Elements: 3D object



☑ Cube □ Static

Tag Untagged Layer Default

Transform

Position	X 3.199	Y 2.567	Z 3.567
Rotation	X 50	Y -30	Z 0
Scale	X 1	Y 1	Z 1

Cube (Mesh Filter)

Mesh: Cube

☑ **Box Collider**

Is Trigger □

Material: None (Physic Material)

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

☑ **Mesh Renderer**

Lighting

Materials

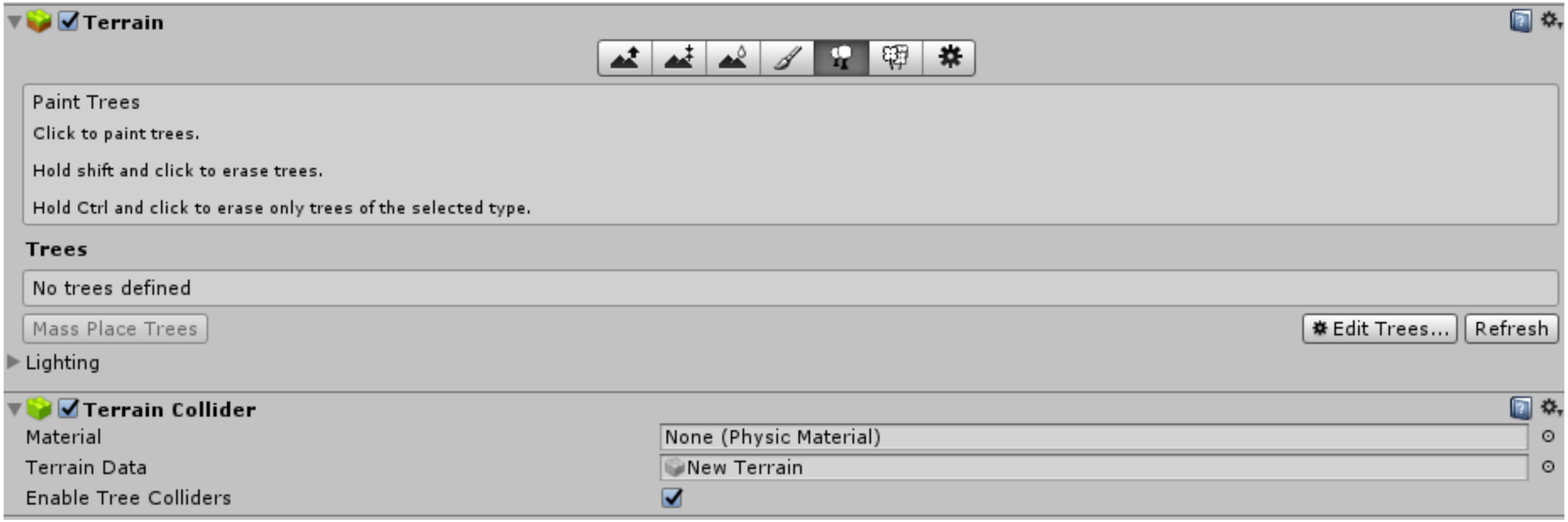
Dynamic Occluded ☑

Default-Material

Shader Standard

Unity Elements: Plane vs Terrain

- Plane : 3D object (Primitive) with a simple Collider (less computation)
- Terrain: 3D object with a terrain component with a specialized tool.



☑ Terrain

Paint Trees

Click to paint trees.

Hold shift and click to erase trees.

Hold Ctrl and click to erase only trees of the selected type.

Trees

No trees defined

☑ Terrain Collider

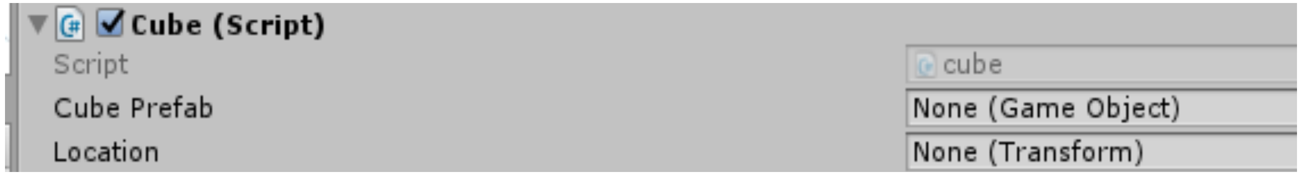
Material: None (Physic Material)

Terrain Data: New Terrain

Enable Tree Colliders: ☑

Unity Elements: Scripting

- Inherit from Monobehaviour
- Visual Studio Community C#
- Attach to a GameObject to change its behavior
- Instanciate Prefab on runtime

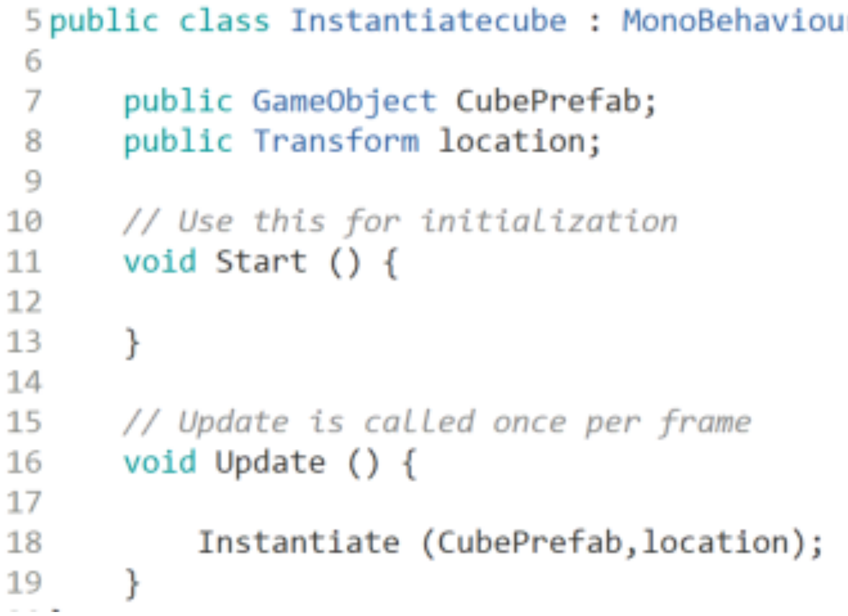


☑ **Cube (Script)**

Script: cube

Cube Prefab: None (Game Object)

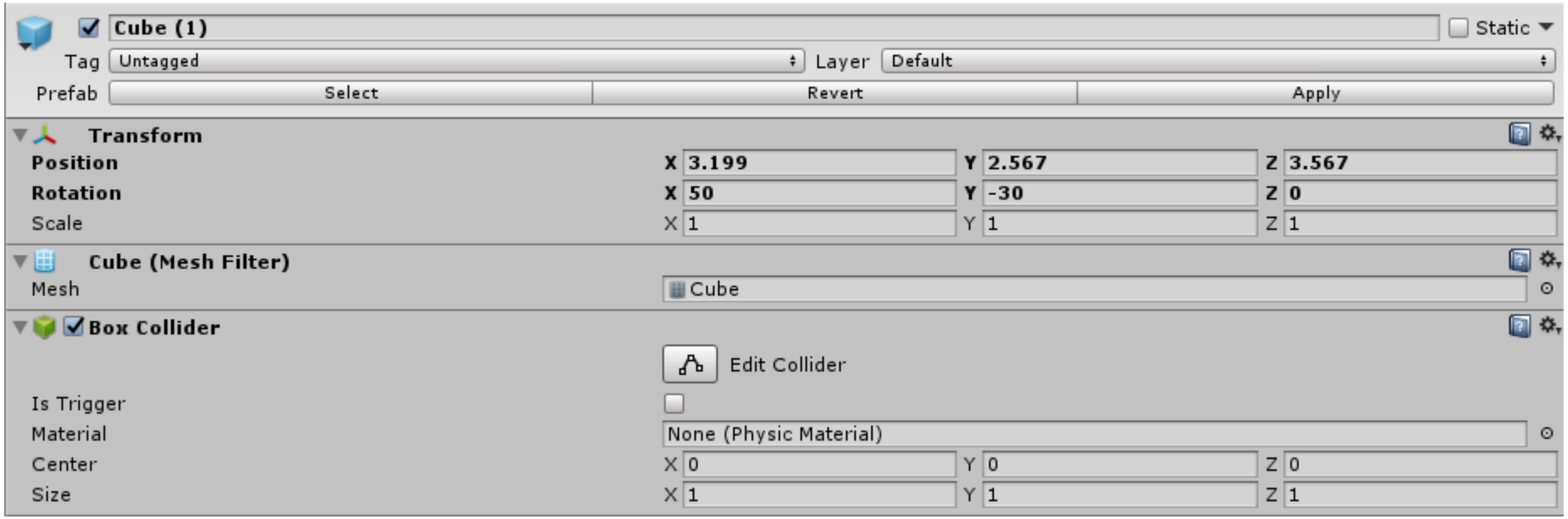
Location: None (Transform)



```
public class Instantiatecube : MonoBehaviour {

    public GameObject CubePrefab;
    public Transform location;

    // Use this for initialization
    void Start () {

    }

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

        Instantiate (CubePrefab,location);
    }
```

Unity Elements: Prefab

“Prefab acts as a template from which you can create new object instances in the scene.”*1



☑ Cube (1) □ Static

Tag Untagged Layer Default

Transform

Position **X** 3.199 **Y** 2.567 **Z** 3.567

Rotation **X** 50 **Y** -30 **Z** 0

Scale X 1 Y 1 Z 1

Cube (Mesh Filter)

Mesh Cube

☑ **Box Collider**

Is Trigger □

Material None (Physic Material)

Center X 0 Y 0 Z 0

Size X 1 Y 1 Z 1

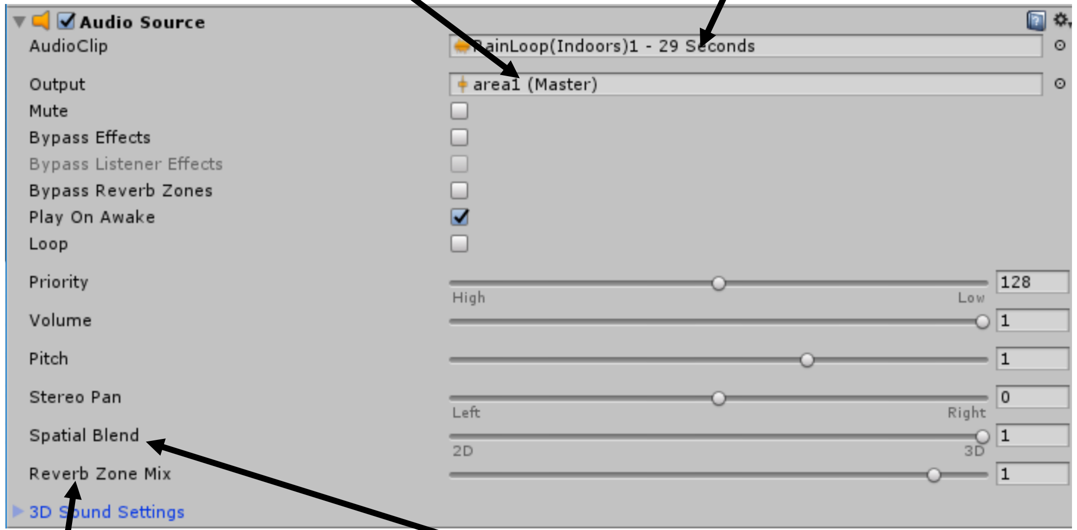
*1 https://docs.unity3d.com/Manual/Prefabs.html

Unity Elements: Audio

3D Sound = “all channels downmixed to mono and attenuated according to distance and direction.”*1

Original Sound Clip

The sound can be output through an audio listener or an audio mixer.



▼ ☑ **Audio Source**

AudioClip: RainLoop(Indoors)1 - 29 Seconds

Output: area1 (Master)

Mute ☐

Bypass Effects ☐

Bypass Listener Effects ☐

Bypass Reverb Zones ☐

Play On Awake ☑

Loop ☐

Priority: 128 (High – Low)

Volume: 1

Pitch: 1

Stereo Pan: 0 (Left – Right)

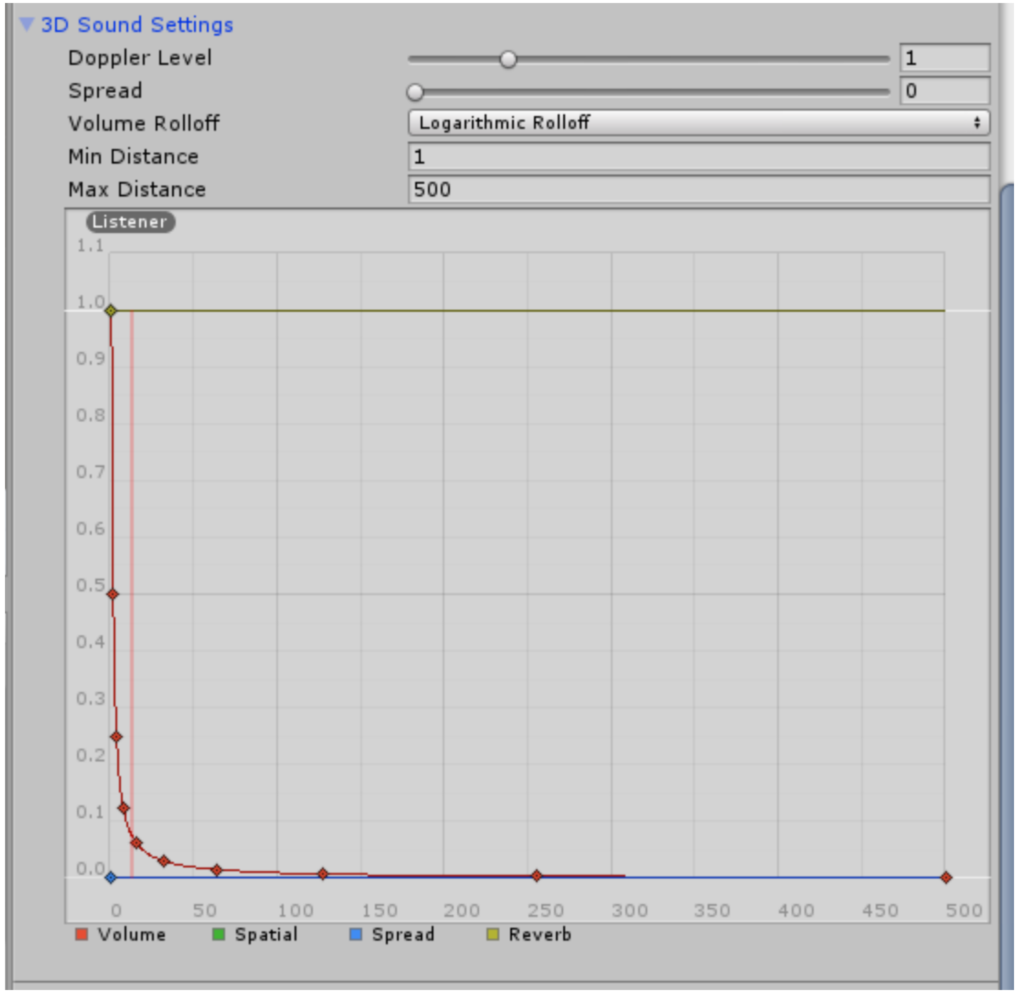
Spatial Blend: 1 (2D – 3D)

Reverb Zone Mix: 1

▶ 3D Sound Settings

Sound Effect to reverberate your sound (Cave like)

Change 2D sound to 3D Sound



▼ 3D Sound Settings

Doppler Level: 1

Spread: 0

Volume Rolloff: Logarithmic Rolloff

Min Distance: 1

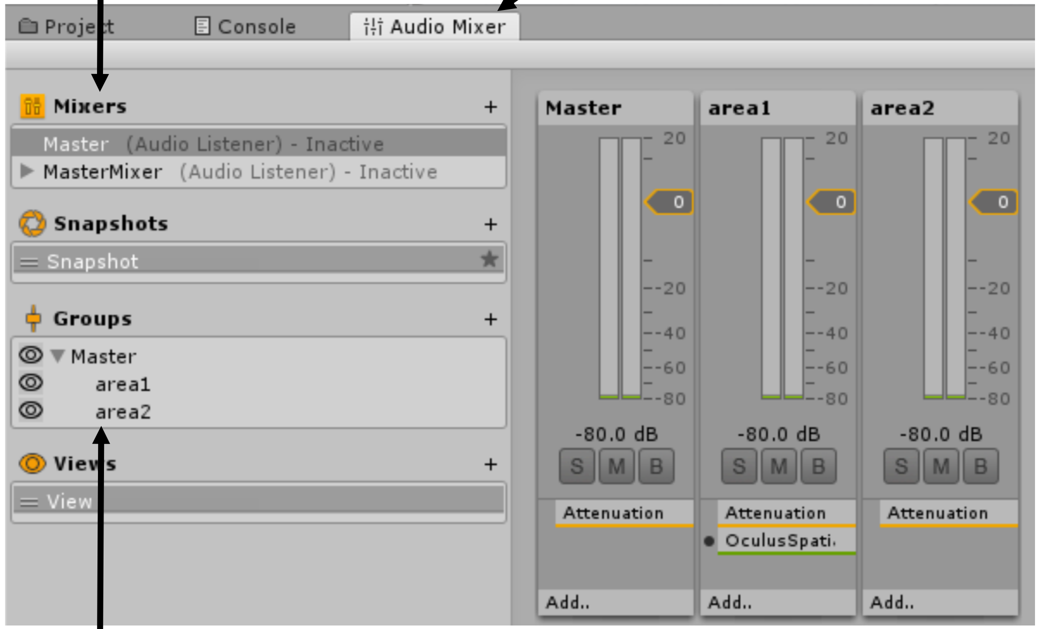
Max Distance: 500

*1 https://docs.unity3d.com/Manual/class-AudioSource.html

Unity Elements: Audio

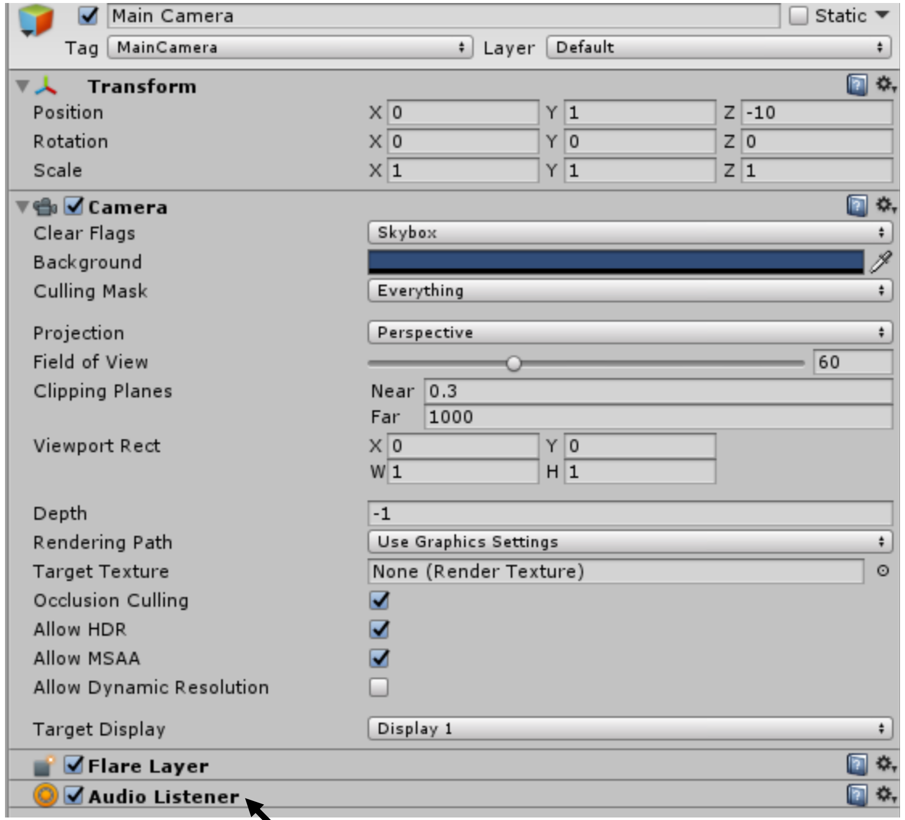
All Mixers Master

Tab to manage your different mixers



Manage the different mixers under different subgroups

Add affect to a group of sound



☑ Main Camera

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**

Clear Flags	Skybox
Background	
Culling Mask	Everything
Projection	Perspective
Field of View	60
Clipping Planes	Near 0.3 Far 1000
Viewport Rect	X 0 Y 0 W 1 H 1
Depth	-1
Rendering Path	Use Graphics Settings
Target Texture	None (Render Texture)
Occlusion Culling	☑
Allow HDR	☑
Allow MSAA	☑
Allow Dynamic Resolution	□
Target Display	Display 1

☑ **Flare Layer**

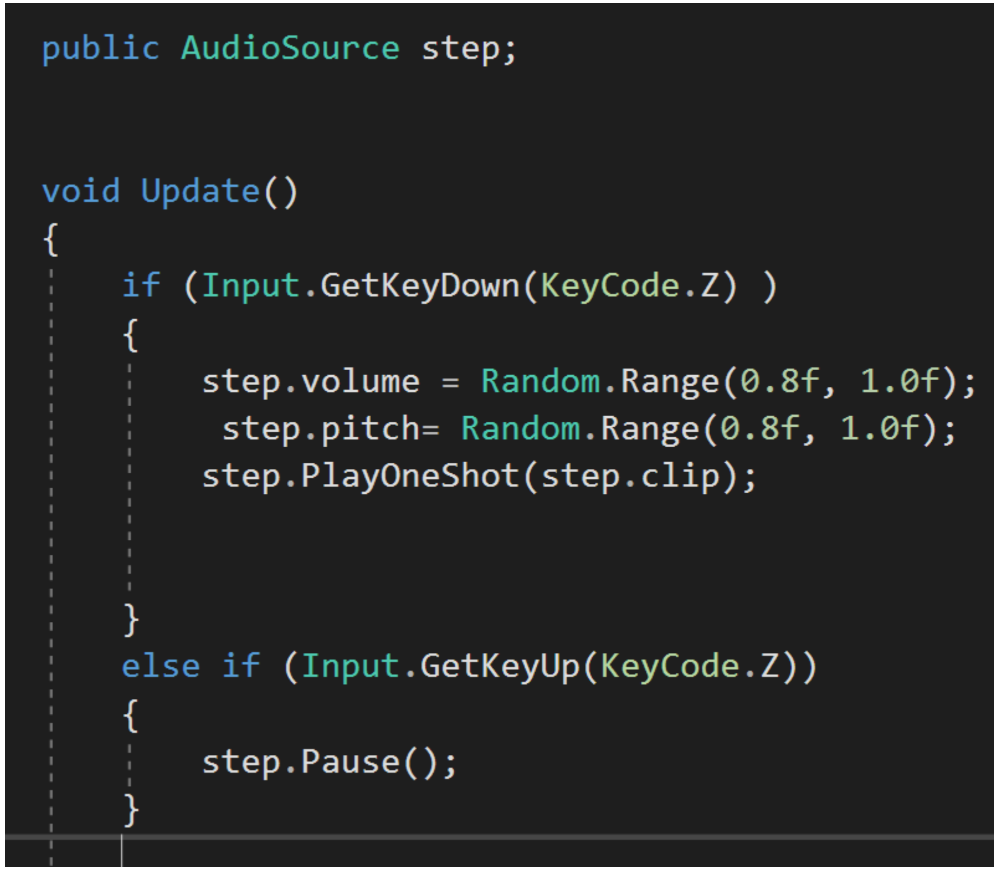
☑ **Audio Listener**

Simple output without any effect

Unity Elements: Audio

Volume How loud the sound is at a distance of one world unit (one meter) from the **Audio Listener**.*1

Pitch Amount of change in pitch due to slowdown/speed up of the **Audio Clip**. Value 1 is normal playback speed.*1



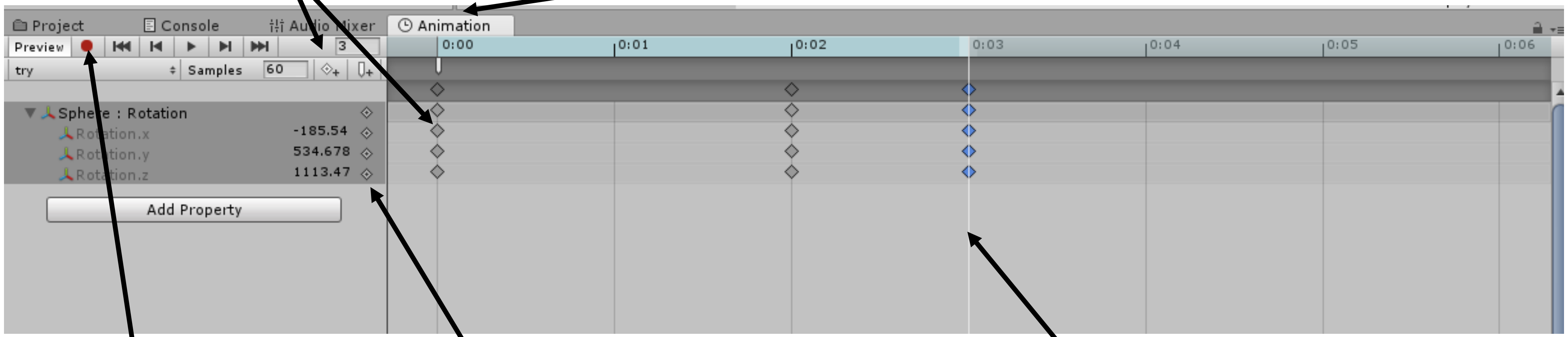
```
public AudioSource step;


void Update()
{
    if (Input.GetKeyDown(KeyCode.Z) )
    {
        step.volume = Random.Range(0.8f, 1.0f);
         step.pitch= Random.Range(0.8f, 1.0f);
        step.PlayOneShot(step.clip);



    }
    else if (Input.GetKeyUp(KeyCode.Z))
    {
        step.Pause();
    }
```

*1 https://docs.unity3d.com/Manual/class-AudioSource.html

Unity Elements: Mecanim

Add Key Frame to your animation

The Animation Tab allows you to create simple animations directly in Unity

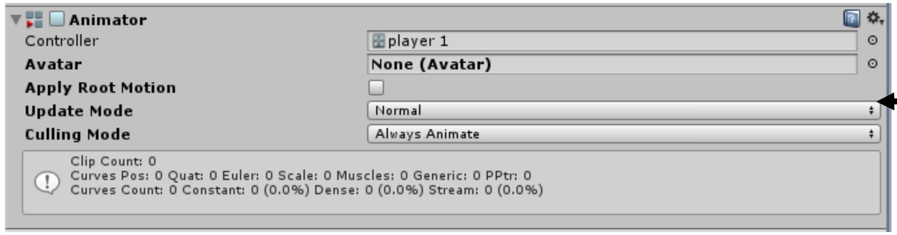


Sphere : Rotation
Rotation.x -185.54
Rotation.y 534.678
Rotation.z 1113.47

Record your animation when moving your object

Control your animation manually

Move in your animation



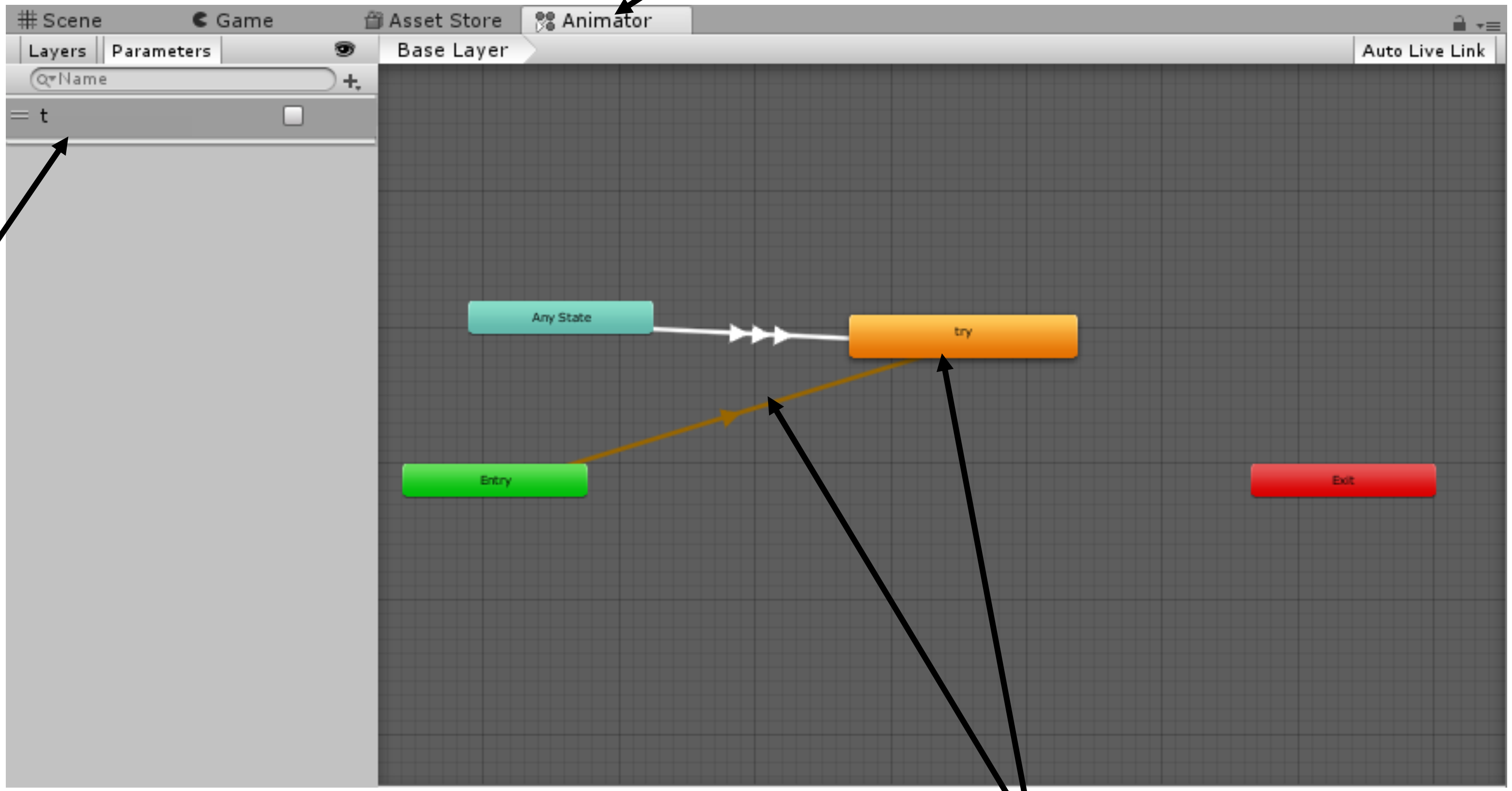
Controller: player 1
Avatar: None (Avatar)
Apply Root Motion ☐
Update Mode: Normal
Culling Mode: Always Animate

Clip Count: 0
Curves Pos: 0 Quat: 0 Euler: 0 Scale: 0 Muscles: 0 Generic: 0 PPtr: 0
Curves Count: 0 Constant: 0 (0.0%) Dense: 0 (0.0%) Stream: 0 (0.0%)

Created on the GameObject you have animated

Unity Elements: Mecanim

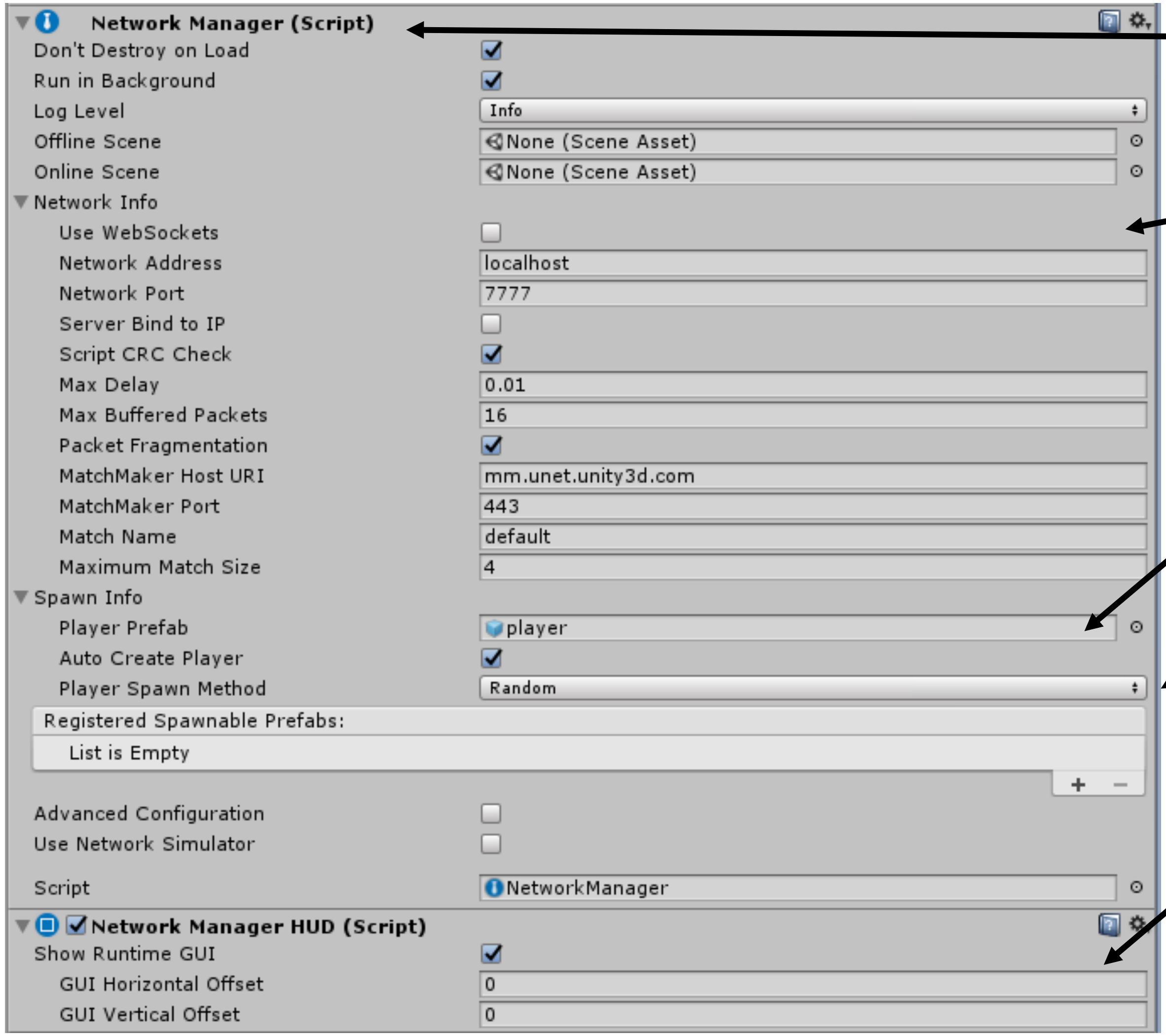
This tab opens automatically when an animator is created. One animator can control different animation clips thanks to the Animator tab.



Different variables to control different states.

Transition and State (animation clip)

Unity Elements: Networking



Network Manager (Script)

Don't Destroy on Load ☑
Run in Background ☑
Log Level: Info
Offline Scene: None (Scene Asset)
Online Scene: None (Scene Asset)
Network Info
  Use WebSockets ☐
  Network Address: localhost
  Network Port: 7777
  Server Bind to IP ☐
  Script CRC Check ☑
  Max Delay: 0.01
  Max Buffered Packets: 16
  Packet Fragmentation ☑
  MatchMaker Host URI: mm.unet.unity3d.com
  MatchMaker Port: 443
  Match Name: default
  Maximum Match Size: 4
Spawn Info
  Player Prefab: player
  Auto Create Player ☑
  Player Spawn Method: Random
Registered Spawnable Prefabs:
  List is Empty
Advanced Configuration ☐
Use Network Simulator ☐
Script: NetworkManager

☑ Network Manager HUD (Script)

Show Runtime GUI ☑
  GUI Horizontal Offset: 0
  GUI Vertical Offset: 0

Add this component to an Empty GameObject named NetworkManager

Test your application on one machine

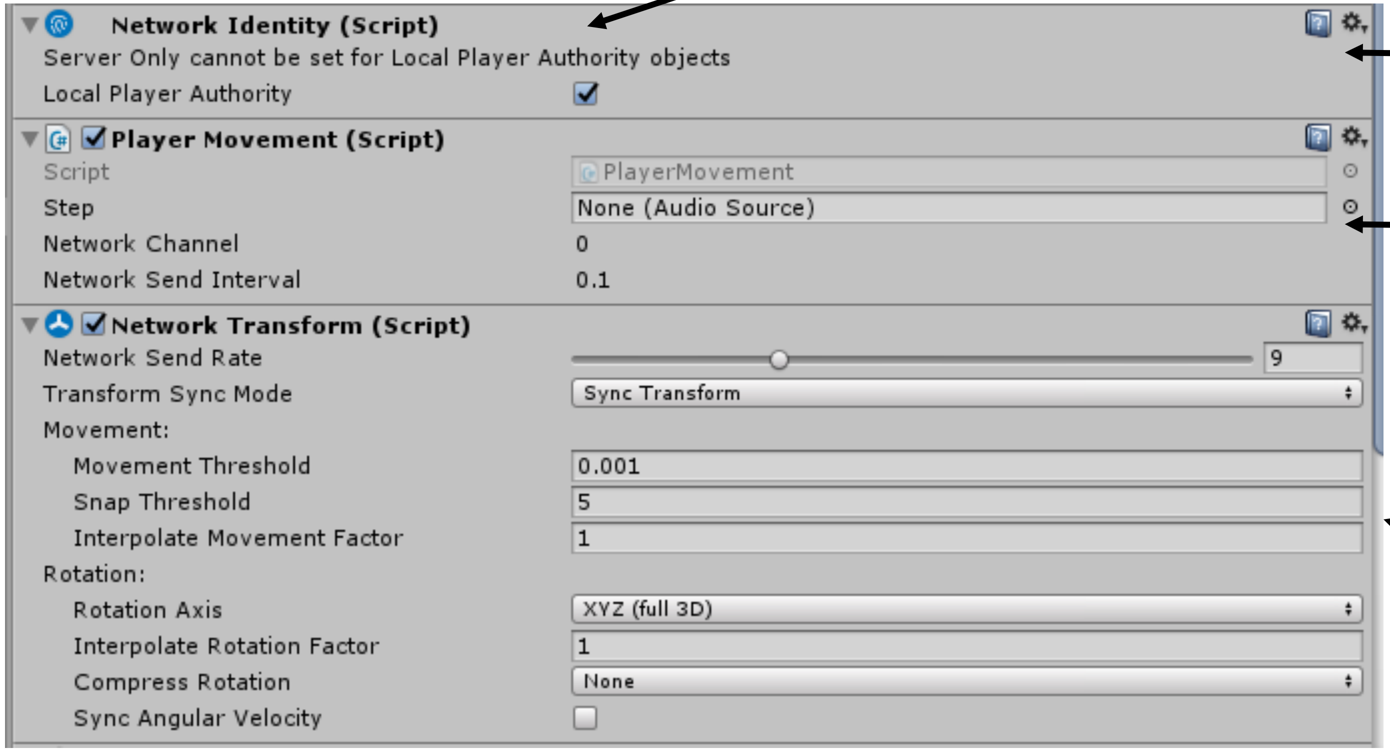
Add the player prefab to be instantiated in the scene

Randomly spawn your player in the different location implemented

HUD to control the Network Manager on runtime

Unity Elements: Networking

All these components need to be added on the player prefab



Network Identity (Script)

Server Only cannot be set for Local Player Authority objects

Local Player Authority ☑

☑ **Player Movement (Script)**

Script: PlayerMovement

Step: None (Audio Source)

Network Channel: 0

Network Send Interval: 0.1

☑ **Network Transform (Script)**

Network Send Rate: 9

Transform Sync Mode: Sync Transform

Movement:

- Movement Threshold: 0.001
- Snap Threshold: 5
- Interpolate Movement Factor: 1

Rotation:

- Rotation Axis: XYZ (full 3D)
- Interpolate Rotation Factor: 1
- Compress Rotation: None
- Sync Angular Velocity ☐

HUD to control the Network Manager on runtime

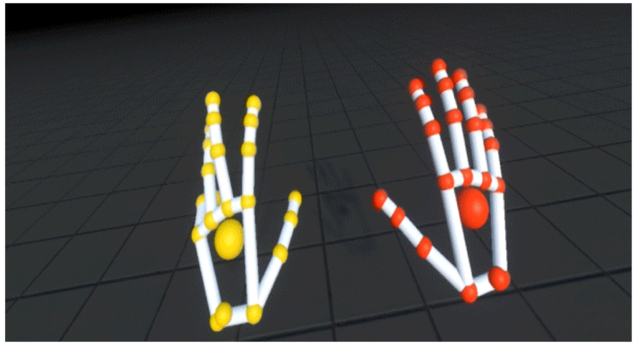
HUD to control the Network Manager on runtime

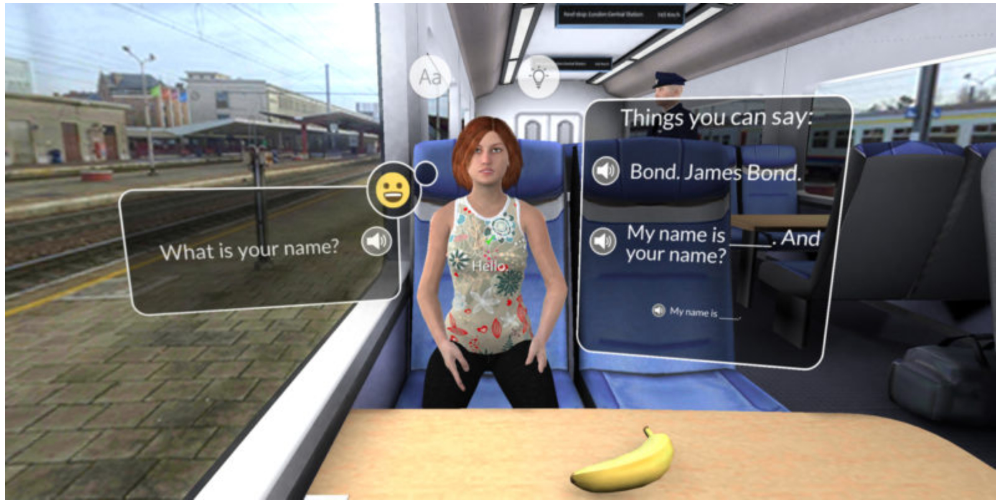
HUD to control the Network Manager on runtime

Virtual Reality in Unity

To have a good VR application you need :

- Good Interaction (Grabbing/UI/Controller)
- Good Physics
- Good AI (Social/Behavior)
- Locomotion (Motion Sickness)





What is your name?

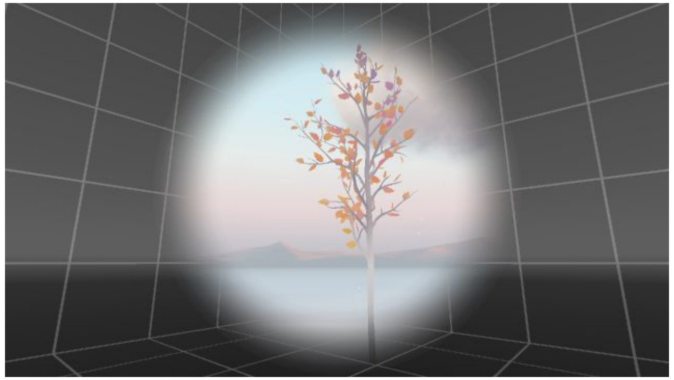
Hello

Things you can say:

Bond. James Bond.

My name is ___. And your name?

My name is ___.



Virtual Reality in Unity

- Edit -> Project Settings -> Player -> XR Settings-> Virtual Reality Supported

-> The camera tagged Main Camera will move according to your head

It is not enough to have a VR application. It is just the beginning of your journey.

Install the Last Unity Version for next course

- Link to download :

https://store.unity.com/fr/?_ga=2.187361902.794958388.1519564968-856357652.1519564968

Different use means different plans but one application:

- Personal License: Annual revenue or funds raised of $100k or less.
- Plus: Annual revenue or funds raised of $200k or less.
- Pro: No limits on revenue or funding.

=>Choose Personal License

Questions?

Useful Resources

- Unity3D:

https://unity3d.com/fr/learn/tutorials

https://unity3d.com/fr/learn/tutorials/topics/asset-store/merry-fragmas-multiplayer-fps-part-1

- VR:

https://assetstore.unity.com/packages/tools/vrtk-virtual-reality-toolkit-vr-toolkit-64131

https://unity3d.com/fr/learn/tutorials/s/virtual-reality

Useful Resources

- Mecanim:

https://unity3d.com/fr/learn/tutorials/modules/intermediate/live-training-archive/character-animation-setup?playlist=17099

https://unity3d.com/fr/learn/tutorials/topics/animation/animate-anything-mecanim

- Networking:

https://unity3d.com/fr/learn/tutorials/topics/multiplayer-networking/introduction-simple-multiplayer-example?playlist=29690