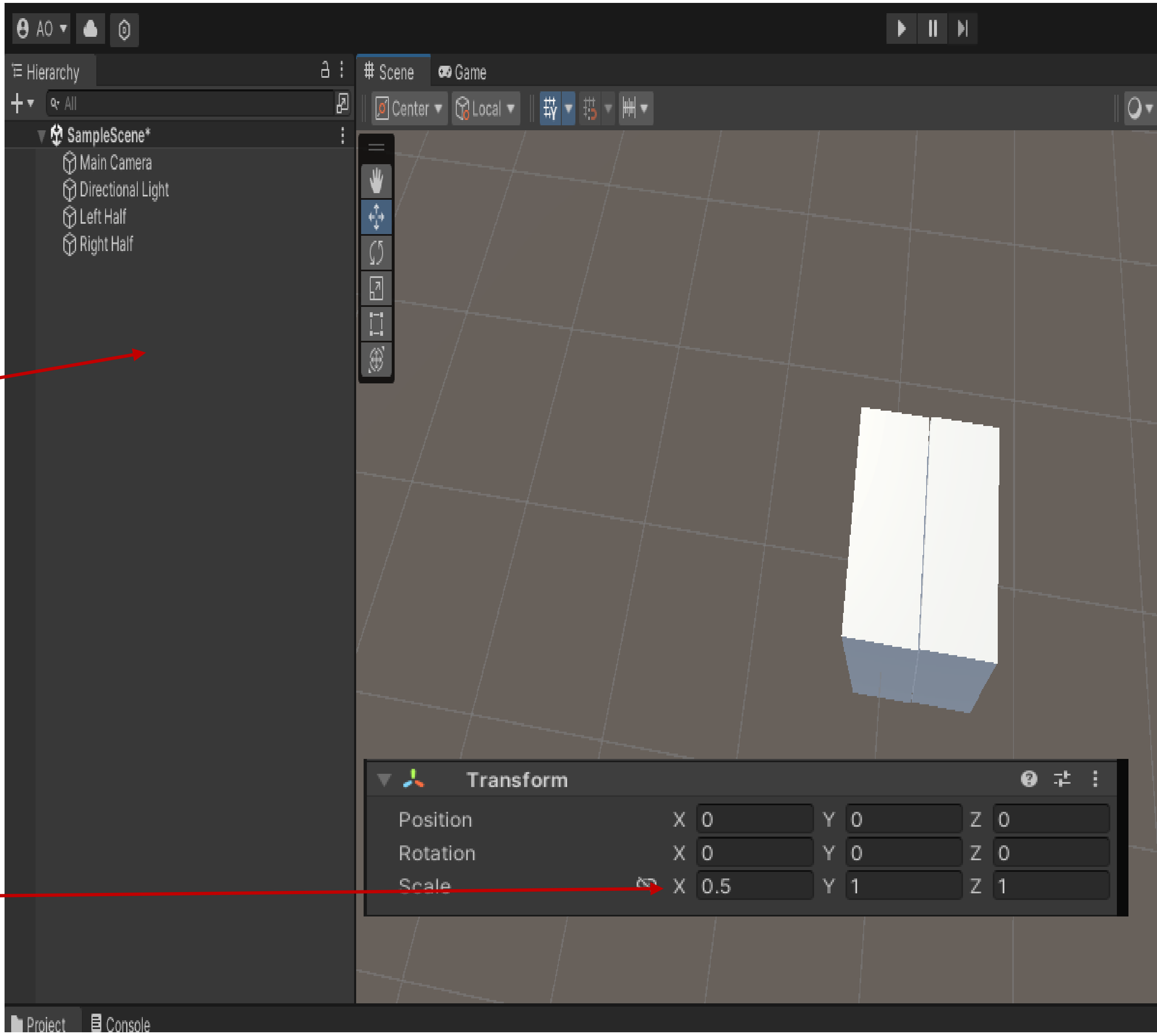
Object Destroy Tutorial

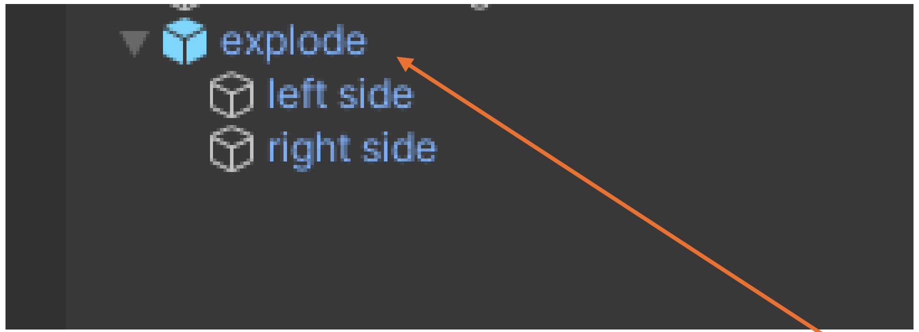
SETTING UP THE PIECES

- First you'll right click in the hierarchy.
- You'll then see [3D Object], youll then hover over to the drop down and you'll see [Cube]. Create two cubes.
- These two will be two halves to make a full cube [set the X scale on both squares to 0.5]

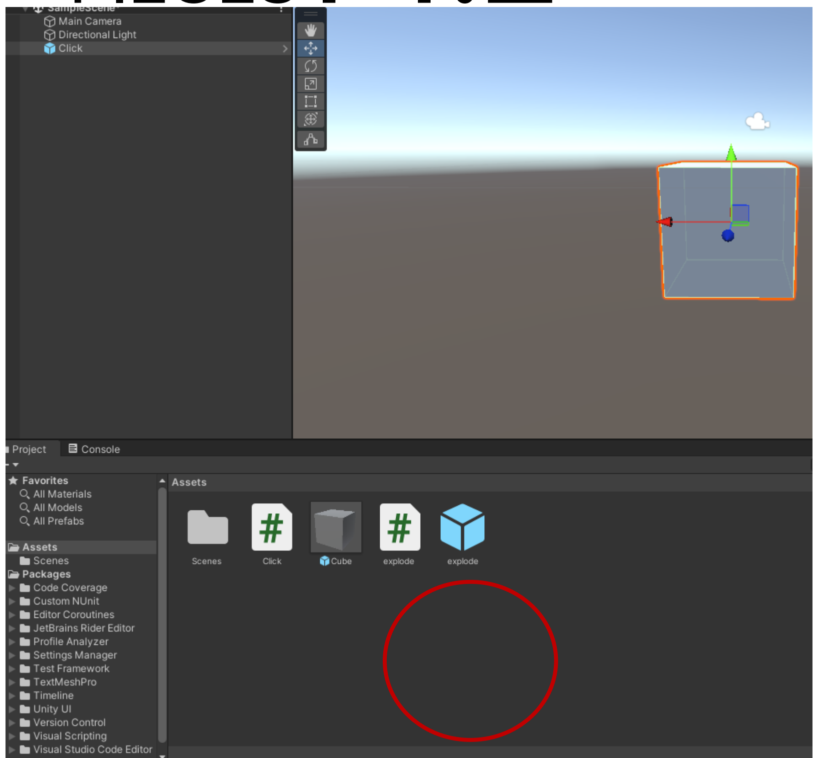


Transform

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



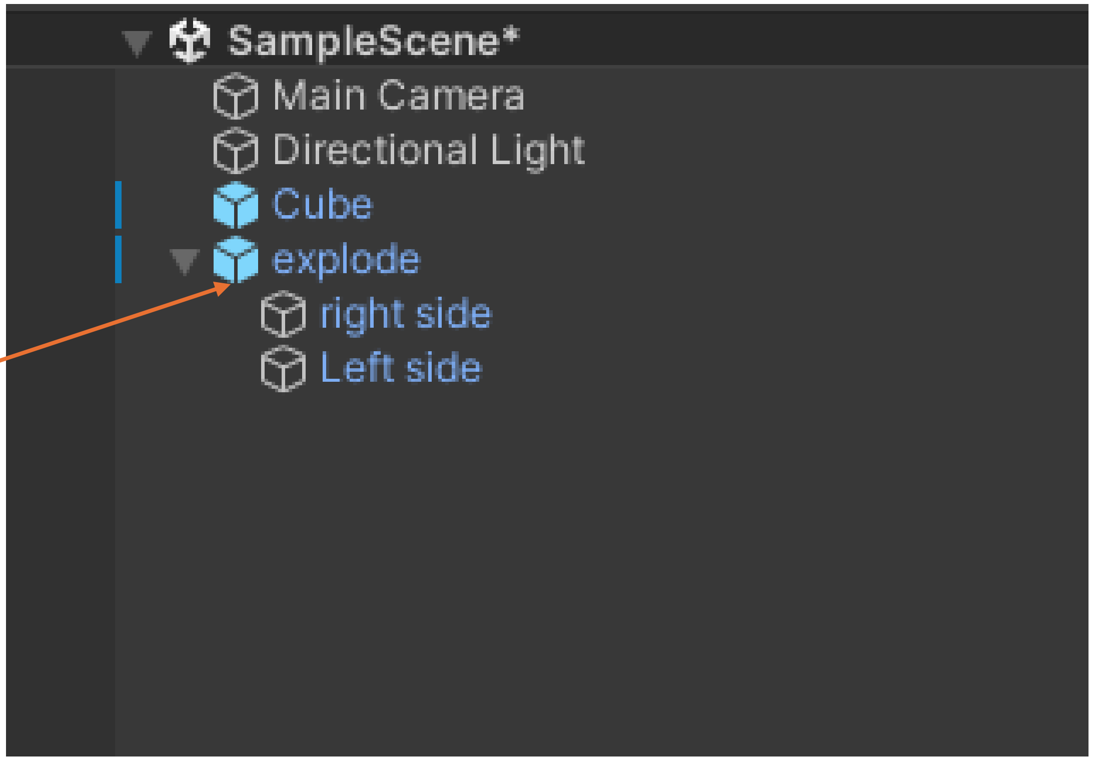
SETTING UP THE PIECES PT.2



- You'll then put in an [Empty Object] and make the two halves the children of that empty object (which will be renamed explode)
- You'll then drag [explode] into the circled area to turn it into a prefab.

SETTING UP THE PIECES PT.3

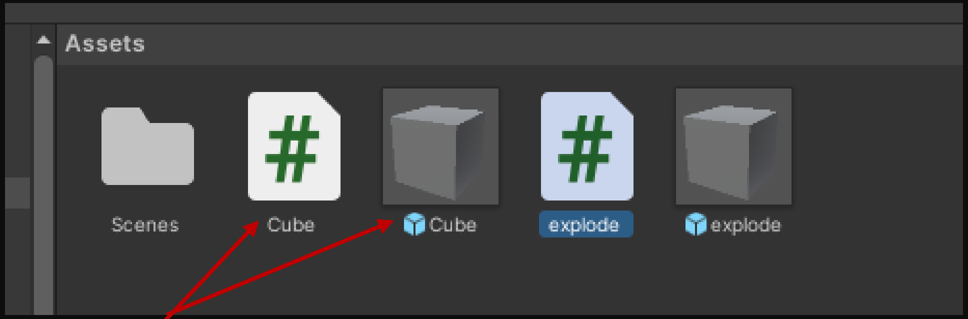
- After that you will then create another cube – this cube will be what you click on to initiate the explosion
- Don't forget to delete the explosion hierarchy or the explosion won't work properly

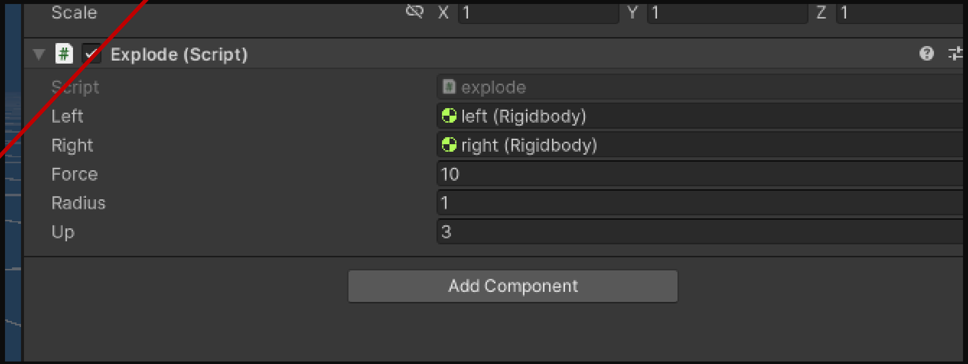


Implementing the actions

- Then you're going to make two c# scripts. One for explode and one for cube.
- Attach the scripts their correct prefab

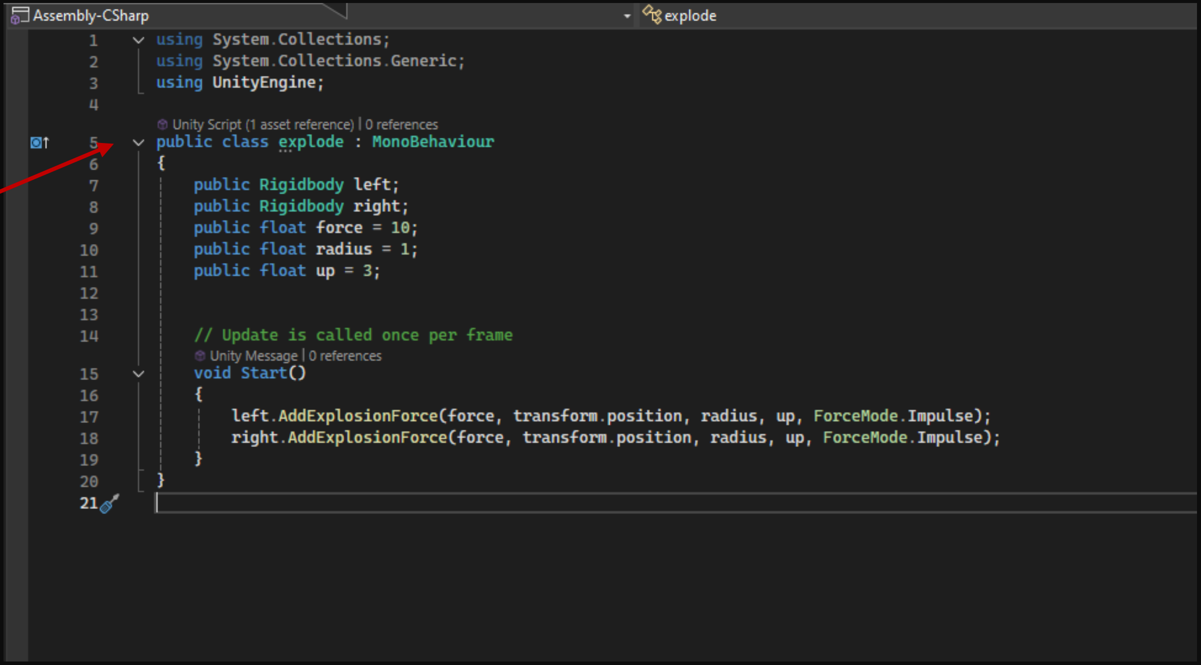
- We will start off with the explosion script.
- The first line in the script: [public class Explode: Monobehaviour]. This basically states that a public class named explode in Unity's Monobehaviour




Scale X 1 Y 1 Z 1
Explode (Script)

Script	explode
Left	left (Rigidbody)
Right	right (Rigidbody)
Force	10
Radius	1
Up	3

Add Component

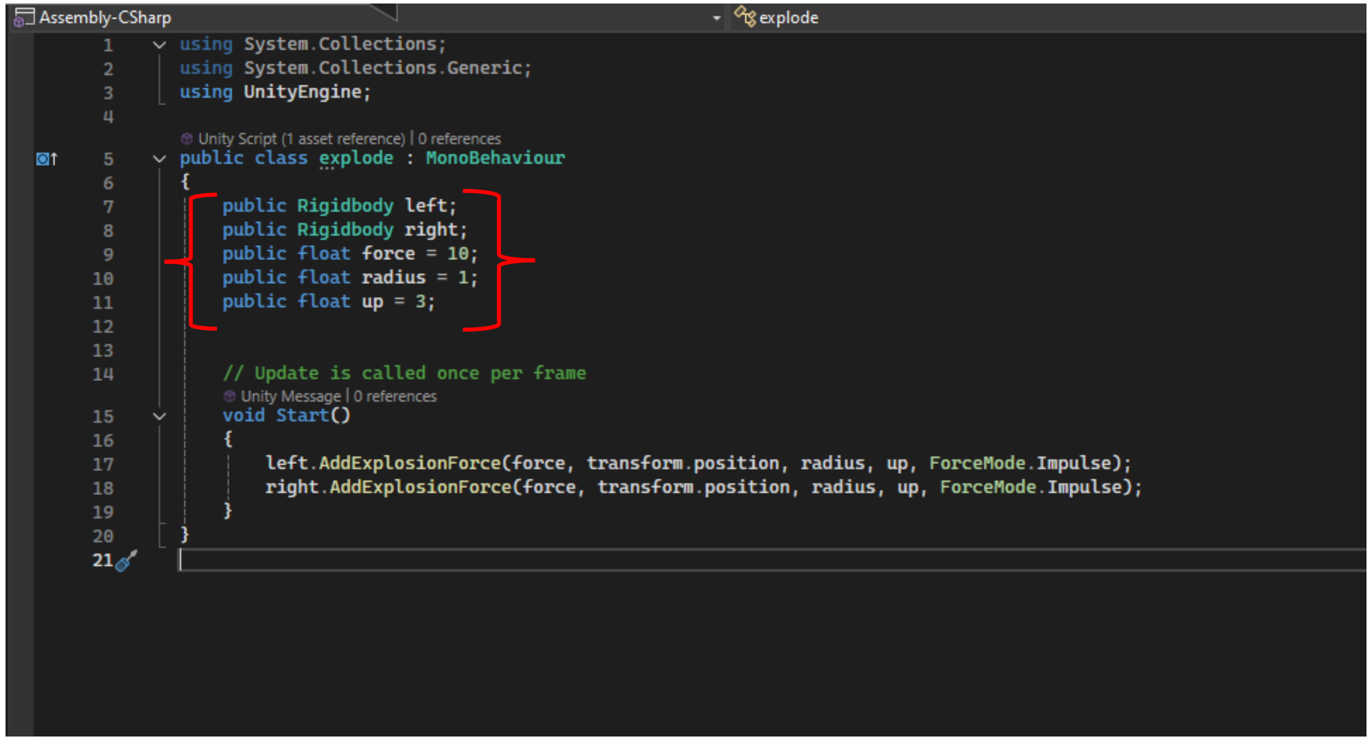

Assembly-CSharp explode

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

Unity Script (1 asset reference) | 0 references
public class explode : MonoBehaviour
{
    public Rigidbody left;
    public Rigidbody right;
    public float force = 10;
    public float radius = 1;
    public float up = 3;


    // Update is called once per frame
    Unity Message | 0 references
    void Start()
    {
        left.AddExplosionForce(force, transform.position, radius, up, ForceMode.Impulse);
        right.AddExplosionForce(force, transform.position, radius, up, ForceMode.Impulse);
    }
}
```

Implementing the actions PT.2

- The codes in brackets are defined as public fields
- The [left] and [right] are references of two rigidbody components (as they both have a rigid body attached to them) that that are going to be affected by the explosion as they are parents of [explosion]

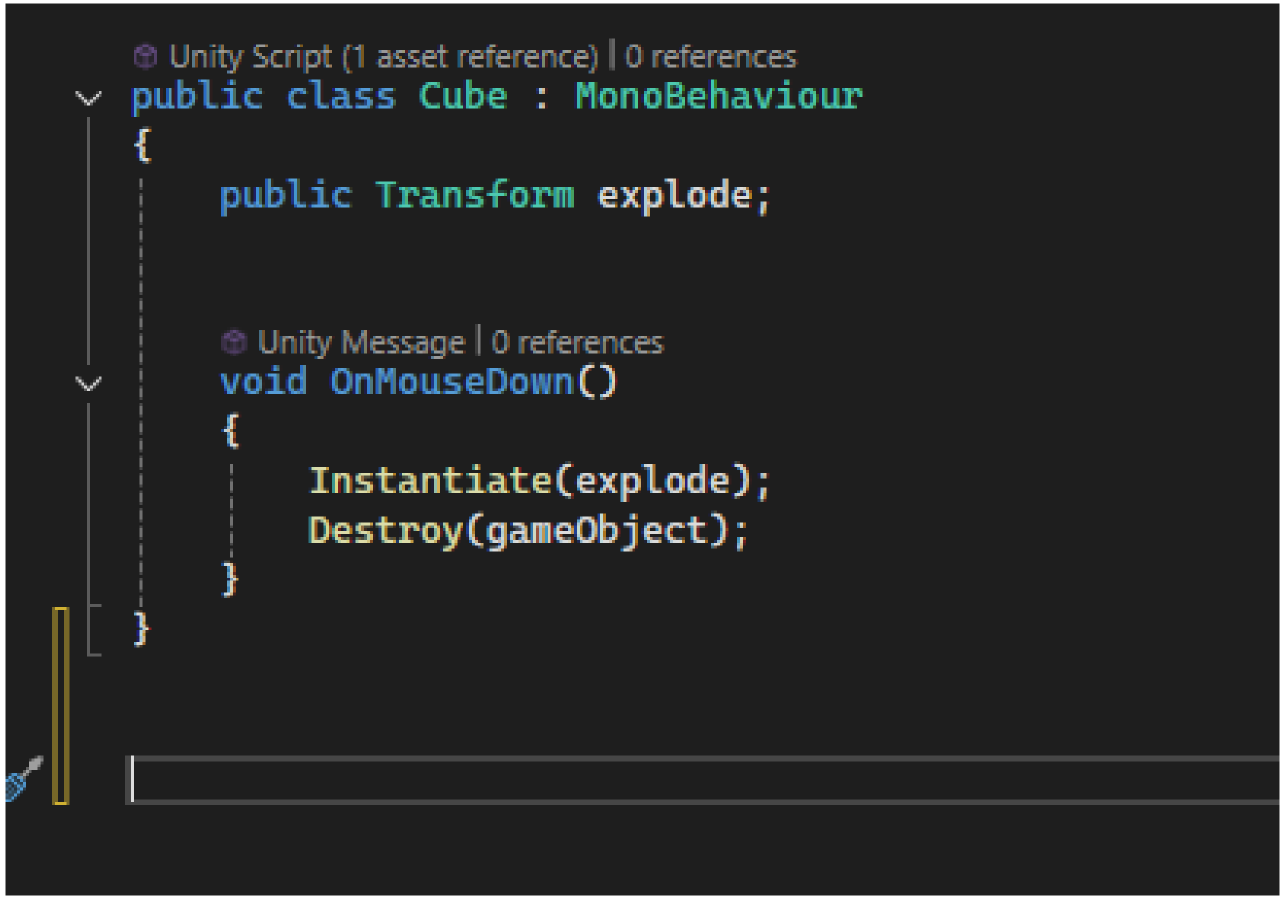
- [force] means the extent of the explosion. The value is set at 10
- [radius] means the radius of the explosion. Shows the area of where explosion will be applied
- [up] simply means how far the object will be pushed up (along Y axis) once explosion has been initiated.


Assembly-CSharp
explode
Unity Script (1 asset reference) | 0 references
Unity Message | 0 references

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

public class explode : MonoBehaviour
{
    public Rigidbody left;
    public Rigidbody right;
    public float force = 10;
    public float radius = 1;
    public float up = 3;


    // Update is called once per frame
    void Start()
    {
        left.AddExplosionForce(force, transform.position, radius, up, ForceMode.Impulse);
        right.AddExplosionForce(force, transform.position, radius, up, ForceMode.Impulse);
    }
}
```

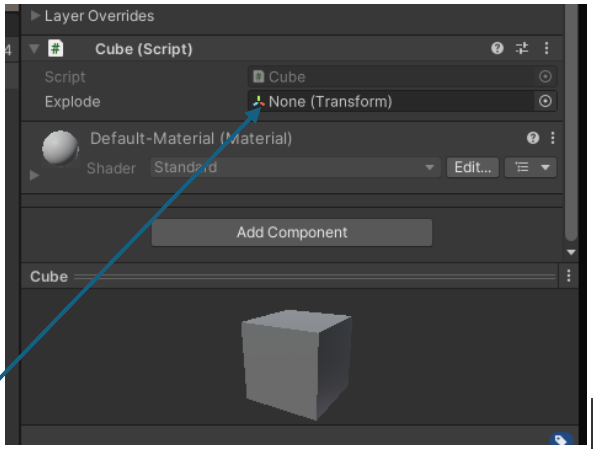
Click (Cube) Script

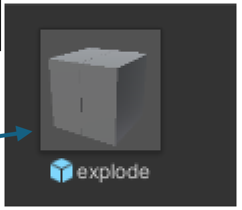
- Public means the variable is accessible in unity.
- A gaming objects position, rotation, and scale in three dimensions are represented by [Transform].
- OnMouseDown simply allow the player to click on the object to initiate the action.
- Instantiate (explode); refers the interaction with the cube to the explosion, when the cube is clicked the explosion will take place
- Destroy(gameObject); will destroy the game object, that being the cube when it's clicked on.

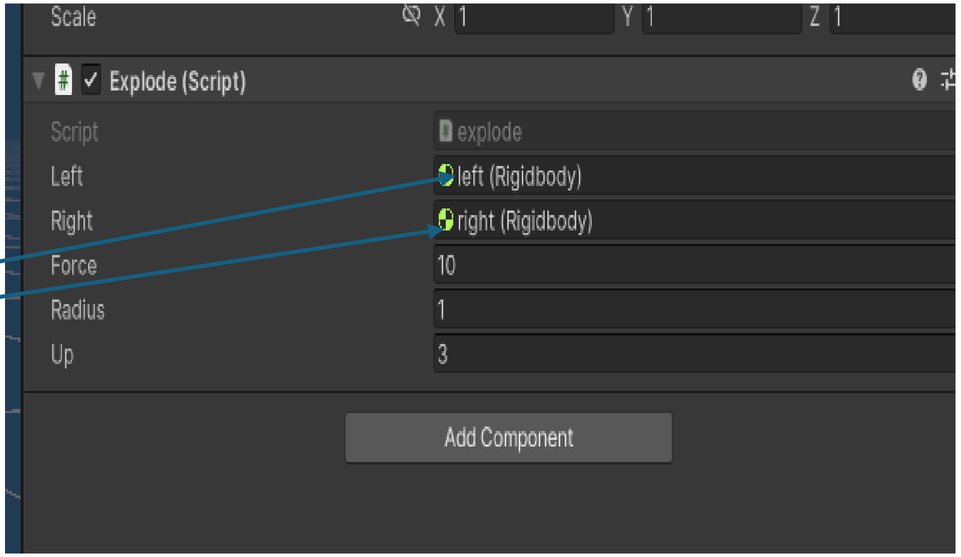


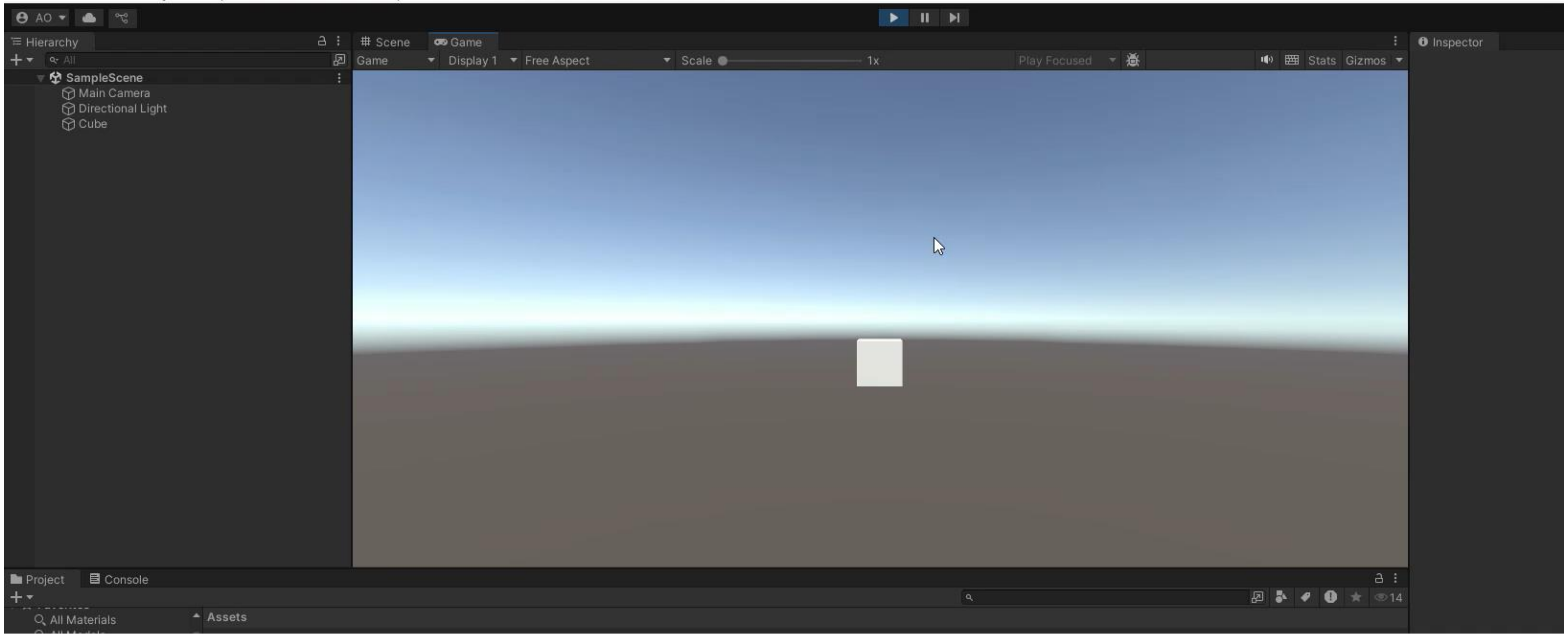
```
Unity Script (1 asset reference) | 0 references
public class Cube : MonoBehaviour
{
    public Transform explode;


    Unity Message | 0 references
    void OnMouseDown()
    {
        Instantiate(explode);
        Destroy(gameObject);
    }
}
```

- Once the code for both [Cube] and [explode] are done
- Youre then going to double click on the explosion prefab
- Attach rigidbodies to both halves
- Then drag your left and right half into the corresponding boxes.
- Lastly youll click on to the cube prefab, scroll down to the cube script attached to it then drag the explosion prefab into None (Transform)
- Then run the script and the cube shall explode. Shown on next slide


Layer Overrides
Cube (Script)
Script: Cube
Explode: None (Transform)
Default-Material (Material)
Shader: Standard




Scale X 1 Y 1 Z 1
Explode (Script)
Script: explode
Left: left (Rigidbody)
Right: right (Rigidbody)
Force: 10
Radius: 1
Up: 3


AO
Hierarchy
Scene
Game
Inspector
Game
Display 1
Free Aspect
Scale
1x
Play Focused
Stats
Gizmos
SampleScene
Main Camera
Directional Light
Cube
Project
Console
All Materials
Assets
14