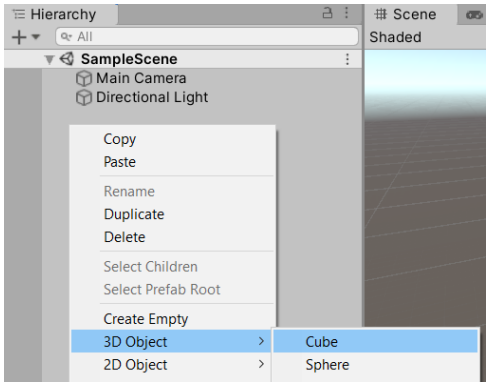
User input and basic sensor in Unity

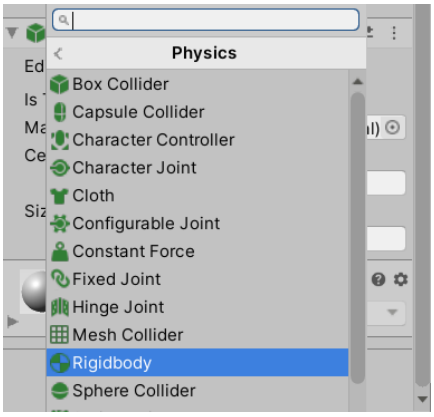
Step 1 – Open unity

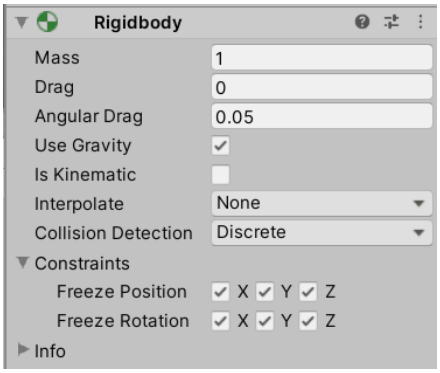
Step 2 - Create a new 3D project and save using an appropriate name

Step 3 – Create a cube in the hierarchy

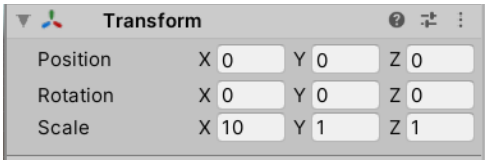


Step 4 - Add a Rigidbody component to the cube in the inspector window with the parameters outlined below:

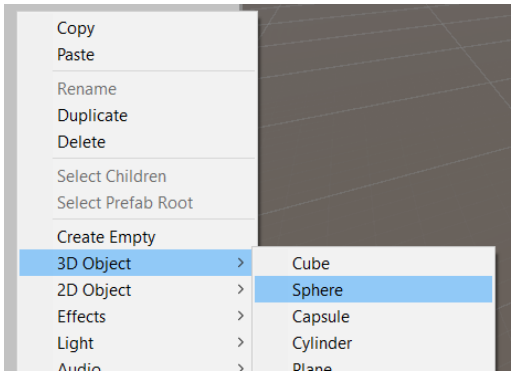




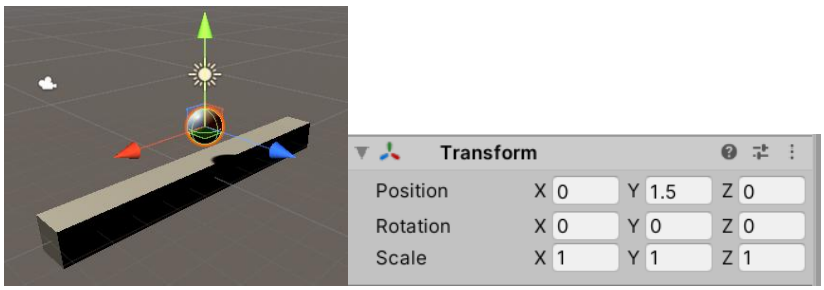
Step 5 - Increase the x scale of the cube in the inspector up to 10 as outlined below:



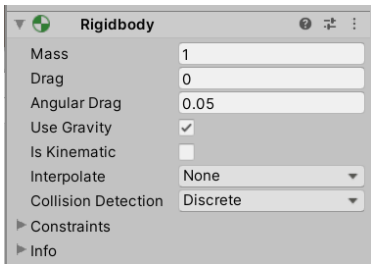
Step 6 – Add a sphere in the hierarchy



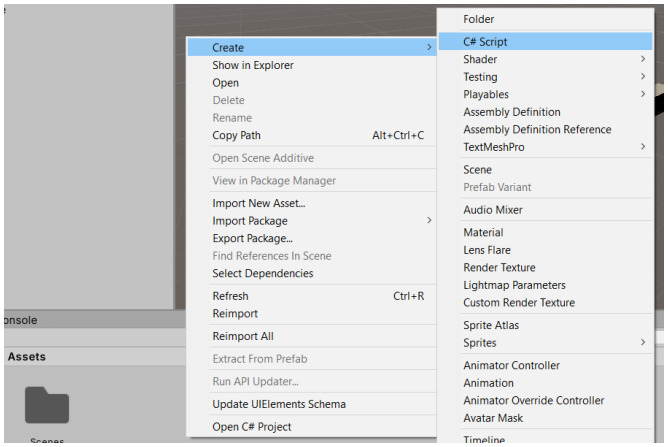
Step 7 – Change the y position of the sphere to 1.5



Step 8 – Add a Rigidbody component to the sphere



Step 9 – Create a C# Script in the assets folder and name it Controller



Step 10 – Apply the Controller script to the sphere

Step 11 – Open the C# script and entre the following code

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

Unity Script | 0 references
public class Controller : MonoBehaviour
{
    private Rigidbody myself;

    // Start is called before the first frame update
    Unity Message | 0 references
    void Start()
    {
        myself = GetComponent<Rigidbody>();
    }

    // Update is called once per frame
    Unity Message | 0 references
    void Update()
    {
        if (Input.GetKeyDown(KeyCode.LeftArrow))
        {
            print("Left key was pressed");
            myself.AddForce(transform.right * -20f);
        }

        if (Input.GetKeyDown(KeyCode.RightArrow))
        {
            print("Right key was pressed");
            myself.AddForce(transform.right * 20f);
        }
    }
}
```

Once you have entered this code run the game and try pressing the arrow keys the ball should move left and right on the cube.

Step 12 – Create a second cube, name it Block and move it to the following position:

Step 13 – Create another C# script named BlockController and apply it to the block you have just created, then enter the following code:

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

Unity Script (1 asset reference) | 0 references
public class BlockController : MonoBehaviour
{
    [SerializeField] private float DetectDistance = 2;
    private Rigidbody myself;

    // Start is called before the first frame update
    Unity Message | 0 references
    void Start()
    {

    }

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

    }

    Unity Message | 0 references
    private void FixedUpdate()
    {
        Vector3 fwd = transform.TransformDirection(Vector3.right);

        Debug.DrawRay(transform.position, fwd * DetectDistance, Color.red);

        if (Physics.Raycast(transform.position, fwd, DetectDistance))
            print("There is something in front of the Block!");
    }
}
```

Step 14 – Run the game and see how the objects react. In particular look at the message window at the bottom of the screen. You may notice that the sphere does not react as you would expect, try to establish why and how to resolve this.