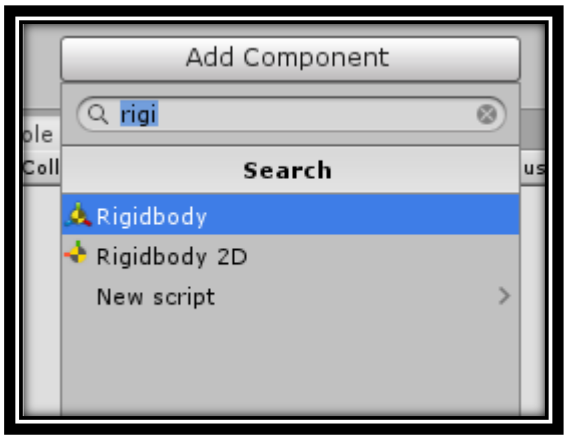
TUTORIAL TWO: JUMPING (3D)

Before beginning the Jumping tutorial you'll need a basic movement script attached to your player game object like the one created in this tutorial – TUTORIAL ONE - Movement.docx

Set up a new unity 3D project.
Select our player, either directly within the scene or via the hierarchy and move your attention to its inspector.
For this to work we'll need to add a new component titled 'Rigidbody'. This component enables physical behaviour for a GameObject. It adds real-world mass to our object and enables it to be affected by forces such as gravity. To add this component to our player, scroll to the bottom of inspector and click on 'Add Component' and then either navigate through the scroll list to Rigidbody or begin typing Rigidbody into the search bar.



Instead of creating a whole new script for our jump mechanic, we can simply type additional code into our Player script we've attached to our player GameObject as we are still just controlling the one object; and so, double left-click to open this script.
We want to begin by creating a couple variables and so within the same class section as moveSpeed, we'll type:

```
private Rigidbody rb;

public float jumpForce;

public SphereCollider;
```

After setting these variables, we want to make sure that in the start section unity knows which components to call upon so beneath moveSpeed, type:

```
rb = GetComponent<Rigidbody>();
```

We now need to write in our update which button we want unity to recognise as our jump button. We'll use space bar as our intended key so beneath our movement script type:

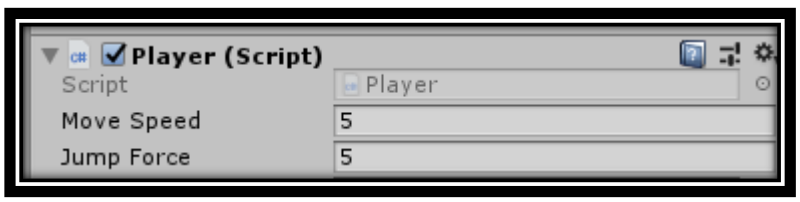
```
if (Input.GetKeyDown(Keycode. Space))
{
      rb.AddForce(Vector3.up * jumpForce, ForceMode.Impulse);
}
```

This translates directly to 'if the player presses the space bar down, once down, unity will apply upward force on our player object, making it jump'.

Your final script should look like this:

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

public class Player : MonoBehaviour
{

    public float moveSpeed;

    private Rigidbody rb;

    public float jumpForce;

    public SphereCollider col;

    // Use this for initialization
    void Start()
    {
        moveSpeed = 5f;

        rb = GetComponent<Rigidbody>();
        col = GetComponent<SphereCollider>();
    }

    // Update is called once per frame
    void Update()
    {
        transform.Translate(moveSpeed * Input.GetAxis("Horizontal") * Time.deltaTime, 0f, moveSpeed * Input.GetAxis("Vertical") * Time.deltaTime);

        if (Input.GetKeyDown(KeyCode.Space))
        {
            rb.AddForce(Vector3.up * jumpForce, ForceMode.Impulse);
        }
    }
}
```

Save and exit your script to unity; press play to load your game and you will now be able to press space bar to jump. Also notice in the inspector under our player script component we've added 'jumpForce' alongside our 'moveSpeed'. You'll be able to tweak this much like our 'moveSpeed' till you're satisfied with your jump.



Like in the previous tutorial, be sure to pay attention to capitalisation, spacing and ensuring that each open curly bracket has a closed one too.