**Learning Journal**

**15/02/22**

In the lesson I decided to develop a first person controller without the use of Unity’s Character Controller component to take advantage of custom physics and forces that the Rigidbody component offers.

While working on the mouse controls for look I encountered the problem of clamping the rotation values on the x axis as euler angles jump between 0 and 360 on the x plane. To solve this problem, I calculated a value for the look angle and clamp it to a range and assign said value, which works for the X axis but this ended up conflicting with the Y axis rotations as this would oppose the rotation of the Player object. To correct this problem I would use **Transform.LocalRotation** instead and set Y rotation to 0 so that the child camera would inherit the player’s rotation.

**22/02/22**

Continuing on from the initial movement system I made, I carried on by adding the ability to wall run from following a tutorial and then making heavy changes to some of the ways forces are applied to make the system feel fast paced.

Later on I then focused on developing a skeleton slide mechanic for the player.

**01/03/22**

Finally with my move system I redeveloped the sliding logic a couple times such that a slide would trigger properly and also introducing a system to change drag if either static or dynamic.

Once these systems were in place, I focused on tuning numbers from the inspector to give the controls a better feel.

**08/03/22**

Starting on a new project, I began developing a physics gun that operates quite similarly to the one used in the game Garry’s Mod.

Getting the system operational was simple enough, although I got sidetracked by trying to make a Line Renderer with variable divisions to curve towards the held object which took more time than expected.

**15/03/22**

In lesson, I was researching ways in which to interact with objects under a canvas for another students project, which resulted in me coming across the IPointerClickHander interface. This allowed me to create a system to click and drag an image and move it across the canvas.

**22/03/22**

This lesson I wanted to use a shader for another game in which I then discovered that it was incompatible with Unity’s Universal Render Pipeline. So I began researching methods to write a screen space shader for URP and found Scriptable Render Passes and began reading about all the different render methods that unity uses.

**29/03/22**

Carrying on from last week, I continued reading documents on HLSL for DX10+ co convert my current shader into one that supports URP.

**26/04/22**

After Easter break I needed some help fixing some of the errors in class. After they were solved I added some additional values to improve Depth Rendering mode and a debug view.

**03/05/22**

For my last package, I stripped down the fast movement parts of my first movement package to form a more simple base movement system. Then following a tutorial, I added viewmodel sway using the mouse then added separate logic using velocities to move the viewmodel while in motion.