Programming tutorial 1: Basic movement scripts with WASD and arrow keys.

**Setting up the scene**

1. Open up a 3D scene in Unity and spawn 2 cubes by opening up the selection on Samplescene > GameObject > 3D Object > Cube, then duplicate the cube.
2. Stretch one cube to create a platform, the other cube will be used as the main player.
3. Name the objects “Platform” and “Player” accordingly.
4. To make my player more recognisable I change the colour of the player by right clicking inside my assets folder Create > Material and change the material colour to blue and assign it to my player.
5. Add a “Character Controller” component to the player.
6. Then add a “Script” to the player and label it “Movement”.

**Building the code**

1. After opening the script remove the start function as we do not need it.
2. We need a speed variable and a reference to our character controller since we’re using that to move our player.

public class Movement : MonoBehaviour

{

public CharacterController controller;

public float speed = 6f;

// Update is called once per frame

void Update()

{

}

}

1. To be able to move the player around using the WASD and arrow keys we need some Horizontal and Vertical Inputs. I’m going to use Input.GetAxisRaw to get the raw input with no smoothing. The horizontal axis will move our character left or right and the vertical axis will move our character up or down.

void Update()

{

float horizontal = Input.GetAxisRaw("Horizontal");

float vertical = Input.GetAxisRaw("Vertical");

}

1. Create a Vector3 to store the direction. For the y axis put in 0f because we want to move the player in the x and z axis but not on the y axis. If you hold down two movement keys to move diagonally the player may move faster, use “.normalized” to prevent that from happening.

void Update()

{

float horizontal = Input.GetAxisRaw("Horizontal");

float vertical = Input.GetAxisRaw("Vertical");

Vector3 direction = new Vector3(horizontal, 0f, vertical);

}

1. Use the following code to check if we are moving in any direction at a magnitude. “Time.deltaTime” is used to make the player move at a consistent velocity.

if (direction.magnitude >= 0.1f)

{

controller.Move(direction \* speed \* Time.deltaTime);

}

1. Drag the Character Controller into the player’s “controller”

The player should be able to move in any direction now using the WASD keys or arrow keys.

Reference used:

Brackeys tutorial “THIRD PERSON MOVEMENT in Unity”

<https://www.youtube.com/watch?v=4HpC--2iowE>