**Project 1: Top-down Zombie game:**

Project requirements:

1. Top-down camera
2. Player movement script for X and Z axis
3. Player sprinting
4. Player rotate with mouse
5. Enemy with navigation (use nav mesh)
6. Enemies with proximity detection
7. Gun system with 3 different weapons
8. Interactable door
9. Lighting system to only show room you are in

Top-down Camera:

* Using Cinemachine virtual camera
* Want to get camera to have offset with the mouse (Future addition): Play around

Player movement script:

* Using raw input system for X and Z axis: Using strings
* Remembered we have the Input system
* Cleaned up the MovePlayer method by adding a currentSpeed variable

11/06/2024

* Watched a guide on the new input system. Making new script which can make use of the new system

Player Sprinting:

* Using Boolean to determine if sprinting
* Sprinting as separate function, is this good?
* Want to restrict sprinting to only going forwards? Might not work with current movement system: Expose a property that is facing forward: centralize the logic

Player rotate with mouse: They can be combined – do not keep separate

* Realized I needed to separate this from the player body or the movement script would fail
* Above was a mistake
* Went through the tutorial games I did and remembered that raycasting is the most likely way to detect mouse position
* Found a thread with a code that has the camera look where it is relative to the plane in the scene
* Need to alter the script to not have player movement dictated by the direction the camera is facing
* Believe the issue isn’t the rotation section but the movement section. Need to alter the input so they can only go in a specific direction
* Separated the rotation controller into it’s own script which is attached directly to the player body. Added a Player base object which controls movement (Used coding practice of compartmentalizing which helped me reach this conclusion)

Suggestion:

* Learn git
* can use git repo locally
* make feature then commit
* Can stage where you feel stable, then can experiment
* Git ignore(Allow to ignore files) file that works for unity
* Put code in appropriate spaces
* When moving files make sure to move their meta files
* Ensure you delete metas when removing because git reads in meta DO NOT RELY ON FOLDERS
* Make sure when you commit you are not accidentally making a meta for something that was deleted