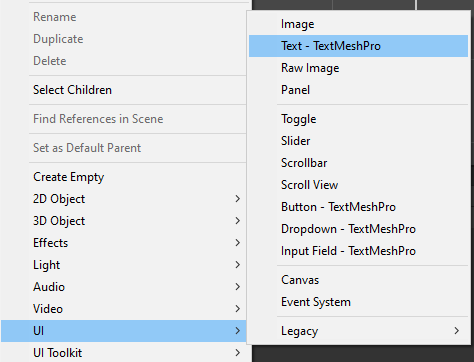
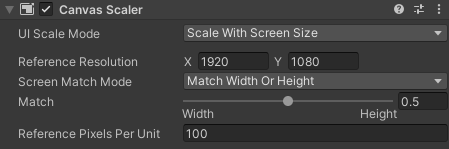
**Dominic Townsend’s Programming Tutorials**

**Pickup Counter**

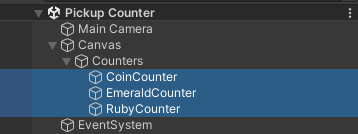
1. Create a TextMeshPro for every counter (in this case, a coin, emerald and ruby counter).



2. In the Canvas settings, set the UI Scale Mode to “Scale With Screen Size”, the Reference Resolution to 1080p and the Match value to 0.5.



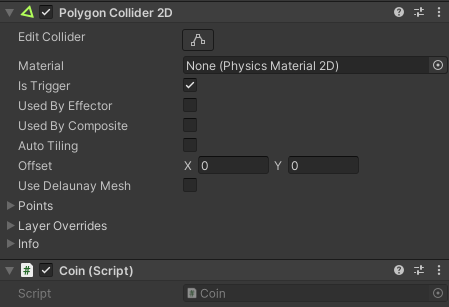
3. If you have multiple counters, group them with an empty GameObject.



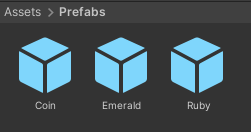
4. Create a sprite for each pickup and give them a texture or placeholder icon.



5. Add a polygon collider to each pickup and tick the “Is Trigger” box. Create an empty script for each pickup and add that too.



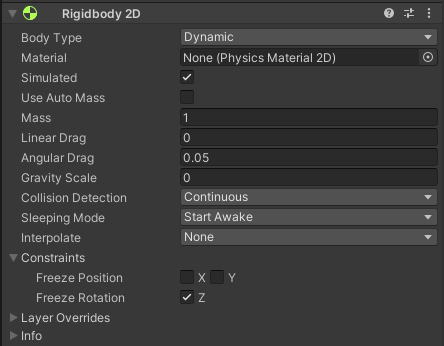
6. Set each pickup as a prefab by dragging them into the assets folder.



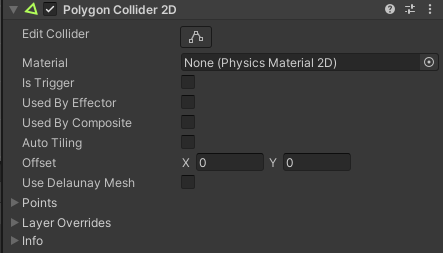
7. Delete the non-prefab pickups from the hierarchy and replace them with the newly made prefabs.



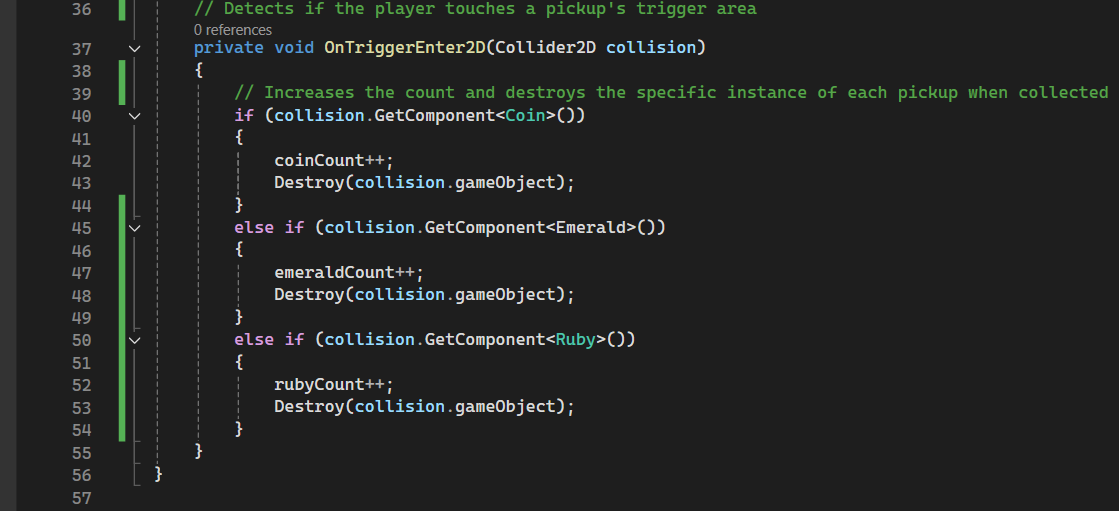
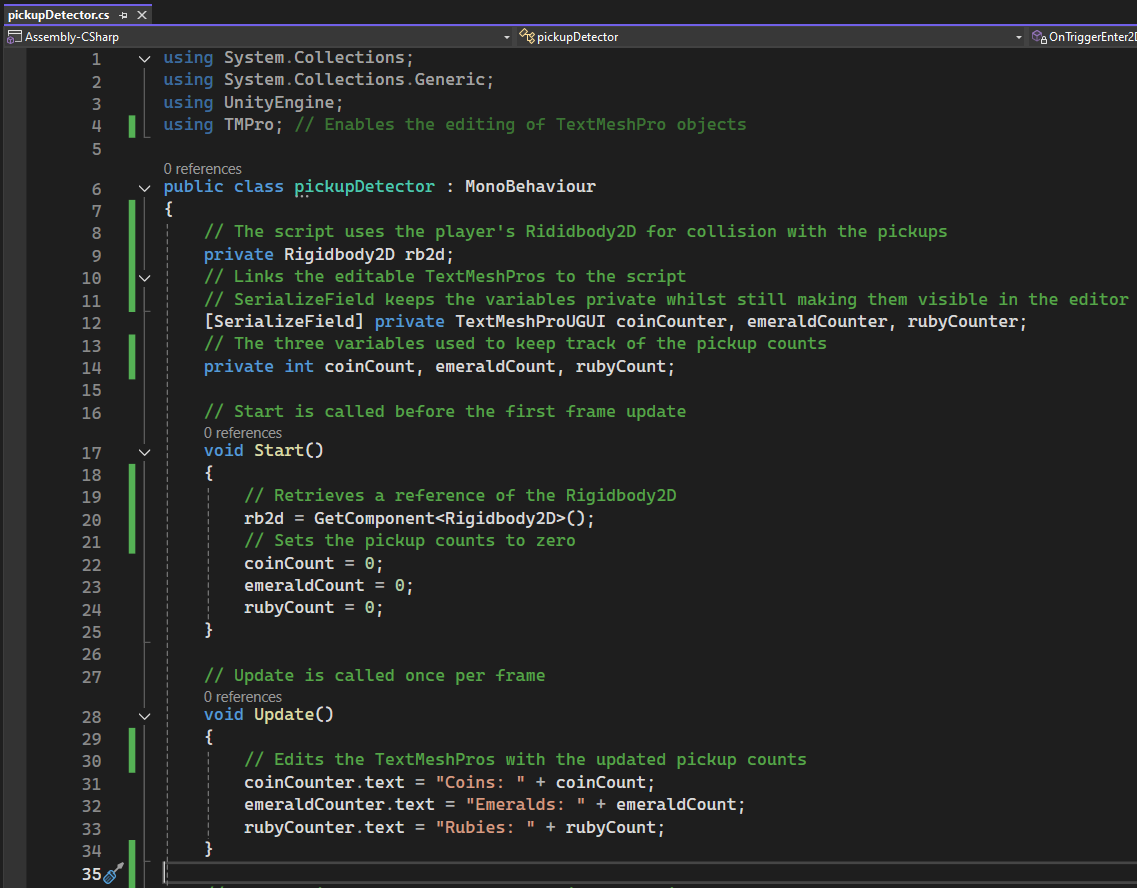
8. Create a player GameObject and attach a RigidBody2D. Set the Body Type to Dynamic and the Gravity Scale to 0.



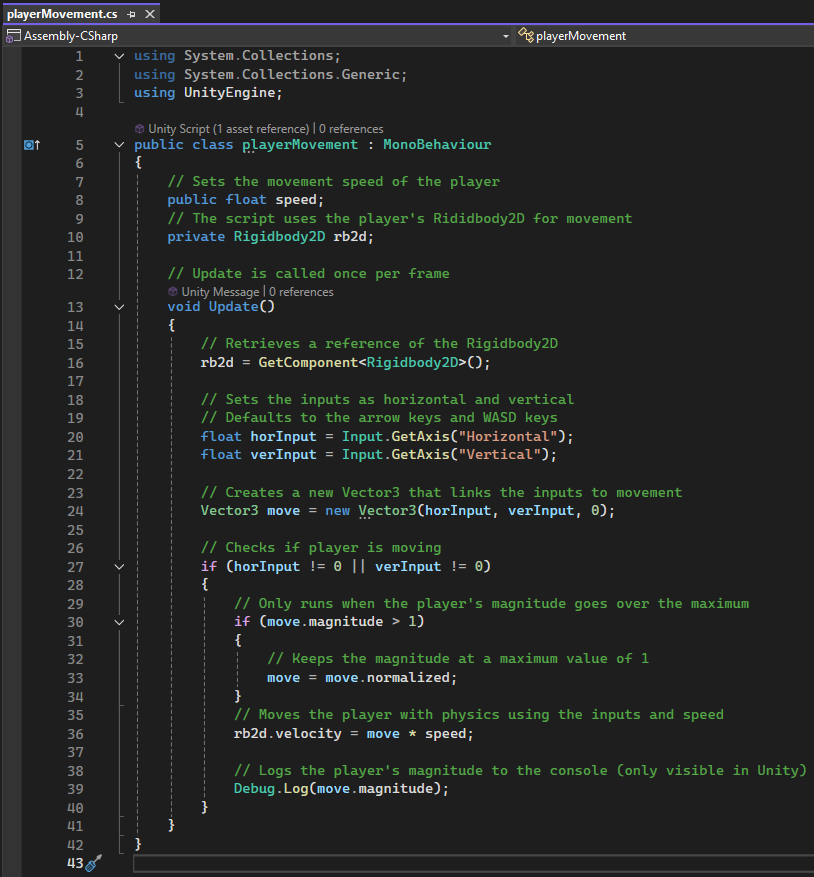
9. Attach a polygon collider, and ensure that the “Is Trigger” box is deselected.



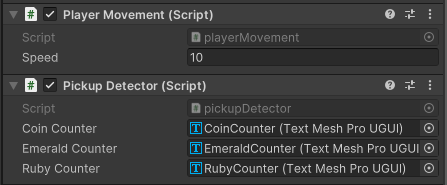
10. Write a script called “pickupDetector.cs”. This will give the player the ability to collect and count pickups.



11. Write another script called “playerMovement”. This movement script will give the player top-down movement, and account for diagonal movement speed issues by normalising the magnitude and capping the speed.



12. Attach both scripts to the player object. Set the speed, and attach the three counter TeshMeshPros to the pickup counter script.



13. When testing the project, the player should now be able to move around and collect each pickup. When this happens, each counter should increase by 1.

