Unity RPG Tutorial #9- Moving Between Areas/Levels

1. Go To Unity. Click on The Header, “File”, then click “New Scene”. The main camera is not the same as the last scene. Save the scene. Save it in the scene folder. Pretend you are making the inside of a house. Call it “House”
2. Create a folder in Assets. (by Right Clicking, and going to new folder). Call It “Prefabs”. Drag from the main scene the MainCamera, and the Player (into Prefab folder you just made). Save the scene. Now click on the “House” Scene from the scenes folder located in assets. Click and drag the Main Camera and Player from Prefabs into the Hierarchy (on the left).
3. Go back to the main scene. Create an empty object into the hierarchy. (right click, create empty). Call it “House Entry”. Now click on House Entry. Under the inspector on the right of unity, click add component (located at the bottom of inspector). Add “Box Collider 2D”. Click the “is Trigger” box under box collider 2d. Click and drag this empty object to right before the house door you have placed.
4. Go to your Scripts folder. Right Click, and create new c# script. Call this “LoadNewArea”.

Delete void update, and void start.

Enter this instead. Under the new script enter this

Using UnityEngine;

Using System.Collections;

Public class LoadNewArea : MonoBehaviour {

Public string levelToLoad;

void onTriggerEnter2D(Collider2D other) {

if(other.gameObject == “Player”)

{

Application.LoadLevel(levelToLoad);

}

}

}

Save the script.

1. Click back on the empty object. Under inspector on the right, click add Component, “LoadNewArea”.

Take the scene name “House” and write house in level to load.

Ex: LevelToLoad House

Now click on “File” under the header, and click “Build Settings”. Click and drag the scenes to the build settings.

Save the scene.

1. Go back to “House”. Delete the Player. Delete the Main Camera. Create an “Empty” and add Box Collider 2D (under add Compoinent). Apply the changes “isTrigger” as well. Move it to the exit of the house. Now Click on your PlayerController Script located under assets. Under void Start()

Add this

void Start() {

….

…..

DontDestroyOnLoad(transform.gameObject);

}

Make sure you paste DontDestroyonLoad(transform.gameObject); to your Camera controller script as well.

Save the scripts.

Watch out though, as this can duplicate the players and cameras.

In the next tutorial, we will go over where we can let the player spawn in the exact position of x and y, and making sure our player doesn’t duplicate everytime between scenes.