

NMSU GDC Meeting 3/29/2018

Unity Tutorial Walkthrough:

- Open Unity and sign in or make an account
- Select “On Disk”
- “New”
 - Name the Project
 - 3D Template
 - Disable analytics to increase speed
- Unity Basics
 - Hierarchy of scenes– Left side of the screen
 - Scene view – center of the screen
 - Folders – bottom of screen
- Making Files
 - Right click in “File” area under “Assets” and create a folder called “Scripts”
 - Repeat for “Materials”
 - Repeat for “PreFabs”
- Making a Plane
 - Right click under “Hierarchy”
 - Selected “3D Object”
 - Select “Plane”
 - Look at the “Inspector” to the right
 - Make sure Position and Rotation are at 0 X, Y, and Z
- Getting the Assets
 - Select “Asset Store” tab beside “Scene” or search for it online
 - Search “Nature Starter Kit 2”
 - Select it and “Download” and then “Import”
 - A box with checkmarks should appear. Click “NatureStarterKit2” to uncheck everything
 - Select the “Nature” folder
 - Select the “Textures” folder
 - Click “Import”
 - You should now see a “NatureStarterKit” folder under “Assets”
- Applying Textures
 - Open the “NatureStarterKit” folder
 - Open “Textures”
 - Drag “Ground01” onto the Plane
- Adding a Character
 - Right click under “Hierarchy”
 - Selected “3D Object”
 - Select “Capsule”
 - Change Position Y to 1 under “Inspector”
 - Right Click the Capsule in Hierarchy and rename to “Player”

- Click the Plane
 - Under Scale, put 10 on X and 10 on Z
- Adjusting the Camera
 - Click “Main Camera”
 - Select the cog in the “Transform” box under “Inspector” and select “Reset”
 - Raise the camera to eye level by dragging it up
 - Drag “Main Camera” onto “Player” to make it a sub category
- Setting up the Mouse and Character Object in Script
 - Enter the “Scripts” folder
 - Right click to create “C# Script” and name it “PlayerController”
 - Open the “PlayerController” script
 - Write your script to look like the following:

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class PlayerController : MonoBehaviour {

    Vector2 mouseLook;
    Vector2 smoothV;
    public float sensitivity;
    public float smoothing;

    GameObject character;

    // Use this for initialization
    void Start () {
        character = this.transform.parent.gameObject;
    }

    // Update is called once per frame
    void Update () {
        var md = new Vector2 (Input.GetAxisRaw ("Mouse X"), Input.GetAxisRaw ("Mouse Y"));
        smoothV.x = Mathf.Lerp (smoothV.x, md.x, 1f / smoothing);
        smoothV.y = Mathf.Lerp (smoothV.y, md.y, 1f / smoothing);
        mouseLook += smoothV;

        transform.localRotation = Quaternion.AngleAxis (-
mouseLook.y, Vector3.right);
        character.transform.localRotation = Quaternion.AngleAxis (mouseLook.x,
character.transform.up);
    }
}

```

- Select “File” and “Save”
 - Return to Unity
- Attaching Script to Camera
 - Select “Main Camera”
 - Drag script from “Assets” to “Main Camera”
 - Under “Inspector,” find “Player Controller (Script)”
- Prettying Up
 - Click the “NatureStarterKit2” File
 - Click Nature
 - Drag 1 of any tree and 1 of any bush onto the Scene
 - Trees and Bushes
 - Shift click bush and tree in Hierarchy and right click
 - Select Duplicate
 - Use the moving tool to scoot trees next to the one another
 - Select tree and bush again
 - Select Duplicate
 - Move them next to the other tree and bush
 - Select tree and Bush again
 - Select Duplicate
 - Move them next to the other tree and bush so you now have a line of four bushes and four trees, alternating
 - Right click under “Hierarchy” and “Create Empty”
 - Right click it and rename it “TreesBehind”
 - CTRL click all bushes and trees
 - Drag them into “TreesBehind” axis
 - Click your “Main Camera” in “Hierarchy” to see where your camera is looking
 - Select “TreesBehind” and move the entire line at once to be behind the camera’s view
 - Click “TreesBehind” and Duplicate
 - Rename it “TreesLeft” and put it on the left of the field of view
 - In Inspector under Rotation, make Y 90 to Rotate
 - Click “TreesLeft” and Duplicate
 - Rename it “TreesRight” and put it on the right of the field of view
 - You should have 3 sides of a box of trees and bushes
- Lighting
 - Click “Directional Lighting”
 - Grab green arrow and drag forward a bit
 - Grab blue and drag forward a bit
 - Click the “Rotate” button above Hierarchy (the two arrows)
 - Grab the red axis and rotate until there is less shadow over the player character
- Lock the Cursor
 - Update the script code to the following:

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class PlayerController : MonoBehaviour {

    Vector2 mouseLook;
    Vector2 smoothV;
    public float sensitivity;
    public float smoothing;

    GameObject character;

    Quaternion in0;
    Vector2 curAng;

    CursorLockMode prevLock;
    bool wasCursorVis;

    // Use this for initialization
    void Start () {
        character = this.transform.parent.gameObject;

        in0 = transform.localRotation;

        prevLock = Cursor.lockState;
        wasCursorVis = Cursor.visible;

        Cursor.lockState = CursorLockMode.Locked;
        Cursor.visible = false;
    }

    // Update is called once per frame
    void Update () {
        var md = new Vector2 (Input.GetAxisRaw ("Mouse X"), Input.GetAxisRaw ("Mouse Y"));
        smoothV.x = Mathf.Lerp (smoothV.x, md.x, 1f / smoothing);
        smoothV.y = Mathf.Lerp (smoothV.y, md.y, 1f / smoothing);
        mouseLook += smoothV;

        transform.localRotation = Quaternion.AngleAxis (-
mouseLook.y, Vector3.right);
        character.transform.localRotation = Quaternion.AngleAxis (mouseLook.x, c
haracter.transform.up);
    }
}

```

- Click Save
 - If you now press play to test, it should no longer show your cursor and move more quickly
-