Tutorial 4 – Drawing

This tutorial will go through creating a script that allows you to draw with the mouse.

The first step is to create an empty game object in the scene, and then add a script to that object (I called mine DrawScript.cs.

In the script, create the following variables:

```
GameObject newDraw;
LineRenderer lr;

bool drawing = false;
int n = 1;
float t = 0;
public float timePerPoint = .1f;
```

“newDraw” is the object that will be created, and “lr” is the lineRenderer that will be attached to said object. “drawing” is set to true when the mouse is held, and false when released. “n” is the number to identify the object in the scene view, “t” is time passed and “timePerPoint” is the amount of time between points being added to the line.

The first step is to create two if statements, one to check if the mouse is pressed, and one for release.

```
if(Input.GetMouseButtonDown(0))
{

}
else if(Input.GetMouseButtonUp(0))
{

}
```

Inside the first, set drawing to true, and the second set drawing to false.

Now a lot of things must be done in the top if statement. First, newDraw must be set to a new gameObject. To this object, you must add a lineRenderer, and store that under the “lr” variable. Set the lineRenderer’s position count to 1 (this makes it so only 1 point is saved on the line), and set that point to the mouse position using lr.SetPosition(). Next, set the width of the line to (.1f, .1f) using lr.SetWidth, and finally, increase n by 1.This prepares the line to start drawing inside a future if statement.

```
newDraw = new GameObject("Line renderer " + n);
lr = newDraw.AddComponent<LineRenderer>();
lr.positionCount = 1;
lr.SetPosition(0, Camera.main.ScreenToWorldPoint(Input.mousePosition));
lr.SetWidth(.1f, .1f);
drawing = true;
n++;
```

Finally, create another if statement in update. This if statement checks if drawing == true. If it does, increase “t” by Time.deltaTime. Then, inside this if statement, add *another* if statement that checks if t >= timePerPoint.

```
t += Time.deltaTime;
if(t >= timePerPoint)
```

If this is true, set t to 0, increase the lineRenderer's position count by 1 and set the new position to the mouse position. You may also need to add 10 on the z axis of the new point, as mine was drawing too far back so it couldn't be seen.

```
if(t >= timePerPoint)
{
    t = 0;
    lr.positionCount++;
    lr.SetPosition(lr.positionCount-1, Camera.main.ScreenToWorldPoint(Input.mousePosition) + new Vector3(0,0,10));
}
```

And now, you should be able to draw!