

Laser ▶ No selection

```

1 using UnityEngine;
2 using System.Collections;
3
4 [RequireComponent (typeof(LineRenderer))]
5
6 public class Laser : MonoBehaviour
7 {
8     public int reflections = 1;
9
10    private LineRenderer laser;
11
12    private int points;
13
14    private Vector3 startPoint;
15    private Vector3 rayDir;
16
17    private RaycastHit hit;
18
19    private bool hittingSwitch = false;
20
21    void Awake ()
22    {
23        laser = GetComponent<LineRenderer> ();
24    }
25
26    void LateUpdate ()
27    {
28        drawLaser();
29    }
30
31    // Function to render the laser
32    void drawLaser()
33    {
34        reflections = Mathf.Clamp(reflections, 1, reflections);
35        points = reflections;
36
37        laser.SetVertexCount(points);
38        laser.SetPosition(0, transform.position);
39
40        startPoint = transform.position;
41        rayDir = transform.TransformDirection(Vector3.forward);
42
43        // Loop for Reflecting the laser
44        for(int i = 0; i < reflections; i++)
45        {
46            if(Physics.Raycast(startPoint, rayDir, out hit, 100))
47            {
48                if(hit.collider.tag == "Reflective" || hit.collider.tag == "PlayerReflector")
49                {
50                    //Debug.DrawLine (startPoint, hit.point, Color.red);
51                    rayDir = Vector3.Reflect((hit.point - startPoint), hit.normal);
52                    startPoint = hit.point;
53
54                    laser.SetVertexCount(++points);
55                    laser.SetPosition(i + 1, startPoint);
56
57                    reflections++;
58                }
59                else
60                {
61                    //Debug.DrawLine (startPoint, hit.point, Color.red);
62                    rayDir = Vector3.Reflect((hit.point - startPoint), hit.normal);
63                    startPoint = hit.point;
64
65                    laser.SetVertexCount(++points);
66                    laser.SetPosition(i + 1, startPoint);
67
68                    if(hit.collider.tag == "End")
69                    {
70                        Manager.Instance.OnLevelComplete();
71                    }
72
73                    if(hit.collider.tag == "Switch")
74                    {
75                        if(!hittingSwitch)
76                        {
77                            hit.collider.GetComponent<Switch>().ToggleSwitches();
78                            hittingSwitch = true;
79                        }
80                    }
81                    else
82                    {
83                        hittingSwitch = false;
84                    }
85                }
86            }
87        }
88
89        reflections = 1;
90    }
91 }

```