using System.Collections;

using System.Collections.Generic;

using UnityEngine;

using UnityEngine.AI; // Now able to use fucntions within unity for AI

public class Waypoint : MonoBehaviour {

NavMeshAgent NavPoint; // Declared the navMeshAgent to be called NavPoint

public Transform[] WayPoints;

public int current\_loc; // Current location

public float stop\_distance; // max distance from waypoint

void Start () {

NavPoint = GetComponent<NavMeshAgent>();

NavPoint.stoppingDistance = stop\_distance;

}

// Update is called once per frame

void Update () {

float distance\_between = Vector3.Distance(transform.position, WayPoints[current\_loc].position); // Finds the distance between the player location and waypoint

transform.LookAt(WayPoints[current\_loc]); // Face the waypoint selected

if (distance\_between <= stop\_distance) // once the player has arrived at the waypoint and the distance is less or equal to stop\_distance then...

{

current\_loc += 1; // add 1

}

if (current\_loc >= WayPoints.Length) // once the player has arrived at the waypoint but the distance is more than stop\_distance then...

{

current\_loc = 0; // reset it and make it equal to 0

}

if (current\_loc == 0) // if the current location of the waypoint is 0 then...

{

NavPoint.SetDestination(WayPoints[0].position); // use NavmeshAgent to set pathway 0 in the Waypoints array

}

if (current\_loc == 1) // // if the current location of the waypoint is 1 then...

{

NavPoint.SetDestination(WayPoints[1].position); // use NavmeshAgent to set pathway 1 in the Waypoints array

}

if (current\_loc == 2) // // if the current location of the waypoint is 2 then...

{

NavPoint.SetDestination(WayPoints[2].position); // use NavmeshAgent to set pathway 2 in the Waypoints array

}

}

}