Tutorial

Tutorial 1 player movement

For this tutorial we will be doing the player movement. To start off you will have to create a script and delete the void start because we want our player moving all the time and not at certain parts. We only need the update method.

void Update()

{

float xDirection = Input.GetAxis("Horizontal");

float yDirection = Input.GetAxis("Vertical");

what this does is the GetAxis is going to detected if you are typing with the WASD keys and will assign a number based on which button you click. It could be the arrows or the WASD.

Now we have to put these directions into a vector3. Vector3 is used for detecting directions.

Vector3 moveDirection = new Vector3(xDirection, 0.0f, yDirection);

Next will be where we want the direction of where the cube will go.

transform.position += moveDirection \* speed;

But we have to set a speed for the direction as the speed will be too quick. Above the void update enter this:

public float speed = .1f;

what this does is it gives a speed of the player so that the speed wont be too quick. So now once you save the script on unity attach the script to the player and you should see this.

Graphical user interface, text, application

Description automatically generated

This is what the speed will be and you can adjust this to however you like.

Tutorial 2 main menu

In this tutorial we will be making a main menu for you to navigate to the game or exit the game. To begin with click the plus, go to UI and then go to panel. You should a panel and next you will have to create a text that says play and quit. Next, we need to create a script for the play button and the quit button. To start you will need to create a public void for play and for quit.Text

Description automatically generated

The next bit of code will be loading the scene. The scene manager is what loads the next scene which in this case is the game. Once you have your two scenes go to file, build settings and at the top put both scenes. What this does it loads your main menu first but once you click play it would take you to your game.

A picture containing graphical user interface

Description automatically generated

Next go back to the script and this code loads your next scene for you to go to with a click of a button.

SceneManager.LoadScene(SceneManager.GetActiveScene().buildIndex + 1);

Next would be the quit option for your main menu. This part is pretty simple as you have this to write:

Application.Quit();

What this does is it quits the game as soon as you click quit it quits the game.

Tutorial 3 patrol

In this tutorial I will be showing the coding of a patrolling enemy.



This is our transform for the waypoints. This is for the enemy to know where to go and the speed is for how quick you want your enemy to move.



Next we have two private variables the first one is for knowing which waypoint youre on and the second private float is for the distance for the ai and the waypoint.

Text

Description automatically generated

In the void start this is for the first waypoint. This allows the waypoint to start from the first one. The the transform is for the enemy to be looking at the waypoint so that it faces to the next waypoint.

Text

Description automatically generated

Next we are going to be creating two functions. The is for the movement of the enemy. what this is doing is its using when you move forwards times the speed is what speed youre going to be at.

Graphical user interface, text, application

Description automatically generated

The second function is for increase index. What this does is it checks to see if the enemy is patrolling in a certain range and if it is it resets it back to zero.

Text

Description automatically generated

Next we are going to check to see if our distance is in range of the waypoint. This is so that to check if our ai is in one position of our waypoint and so if it is then we increase our index.

Next we go back to unity and attach the script to the enemy. you should see something that says size and make 3 cubes. When you make the three cubes rename them as waypoint 1 2 and 3. Then change the speed to however you want it to be.

Graphical user interface, text, application

Description automatically generated

Tutorial 4 enemy finds you

To start this tutoral create a script and at the top of the script add unityengine.ai



Next you will need a reference to your NavMeshAgent so you create a variable. The transform is to show where the enemy will go if he sees the player.

Text

Description automatically generated

Next create a variable to indicate what the ground is and what the player is.



Next this is for the patrolling. This is for walking towards where the player is and how far the player is.

Text

Description automatically generated

Next this state is for checking if the player is in range.

A screenshot of a computer

Description automatically generated with low confidence

What this does is it searches for the player so whatever you named the player has to be in the brackets.

Graphical user interface, text

Description automatically generated

Next we will have to get getcomponent to assign the NavMeshAgent.



Now we will be making the state functions which are patrolling and chase player

Text

Description automatically generated

What this does is it checks to see if the player is in sight range. If so, then it will understand that the player is close.

Text

Description automatically generated

Now this is for the walk point for the enemy. what it does it allows the walk range of the enemy to be configured when back in unity. It searches the map.

Text

Description automatically generated

What this does is it finds the distance for the enemy to walk.

Text

Description automatically generated

This allows the enemy to chase the player.

Graphical user interface, text, application

Description automatically generated

Next you will have to add a nav mesh agent which allows you to increase the radius of the enemy and the speed at which the enemy chases you.

Graphical user interface, text, application

Description automatically generated

Now you will have to tick navigation static which will allow you to go onto bake and bake the terrain. This allows the enemy to move around.

Graphical user interface, application

Description automatically generated

Then where the what is player change the player layer to target by making a new layer and change the what is player to target which will allow the enemy to know that youre the target.