Class PulsingLine

Class to make sure the line, shown when targets gets selected to navigate to, is pulsing.

Inheritance

↳ System.Object
  ↳ PulsingLine

Namespace: DocFx (DocFx.html)

Assembly: DocFx.dll

Syntax

```
public class PulsingLine : MonoBehaviour
```

Fields

endColor

Second color to be shown.

Declaration

```
public Color32 endColor
```

Field Value

Type	Description
Color32	

endPos

/// Current position of the selected target.

Declaration

```
public Vector3 endPos
```

Field Value

Type	Description
Vector3	

lastColorChangeTime

Time it took for the last color change.

Declaration

```
public float lastColorChangeTime
```

Field Value

Type	Description
System.Single	

material

/// Material to transform the color.

Declaration

```
public Material material
```

Field Value

Type	Description
Material	

myPointsInLine

/// Array of points frim the line.

Declaration

```
public Vector3[] myPointsInLine
```

Field Value

Type	Description
Vector3[]	

r

Object were changes should be applied.

Declaration

```
public LineRenderer r
```

Field Value

Type	Description
LineRenderer	

startColor

First color to be shown.

Declaration

```
public Color32 startColor
```

Field Value

Type	Description
Color32	

startPos

Current position of the player.

Declaration

```
public Vector3 startPos
```

Field Value

Type	Description
Vector3	

Methods

changecolor(Single)

This method used the lerp function to change between two color in different speeds so it shows the player how far he/she is away from the target.

Declaration

```
public void changecolor(float Fadeduration)
```

Parameters

Type	Name	Description
System.Single	*Fadeduration*	Gets the Fadeduration to know how fast the line should change its color

Start()

When the Game is started this method gets executed. It gets the linerenderer and his material.

Declaration

```
public void Start()
```

Update()

Method used to continously update the frequency the line changes is color so it can signal how far the target is. Gets to position of the player and the target and calculates the distance. The higher the distance the lower the frequency the line changes its color.

Declaration

```
public void Update()
```