DOOR MANAGER SCRIPT FREE

DOCUMENTATION

===

INDEX

===

===

===

1. INTRODUCTION

===

This is the documentation file for the 'Door Manager Script Free' Unity3D asset available on the asset store. This guide will give you an overview of the asset and it's functions.

If you require an answer to a question that isn't included in this guide, contact me by using one of the methods listed below. (see 4. COPYRIGHT AND CONTACT)

===

===

2. SETUP GUIDE

===

2.1. WHAT'S IN THE BOX (AND WHAT'S NOT)

===

Included:

- ❖ Demo scene to show you what the asset is able to achieve
- ❖ 'Detection' script to detect objects that function as doors/windows in your scene
- ❖ 'Door' script to open/close the detected doors/windows
- ❖ 'MouseLook' and 'PlayerMovement' script to control the movement of the player in the demo scene
- ❖ Some textures, prefabs, models, skyboxes etc… to make the demo scene work

Not really included (but required):

- ❖ A character that is able to move and look around in the game world (you can use the 'MouseLook' and 'PlayerMovement' scripts included in this asset to achieve this, there is also a Player Prefab included in the tool)
- ❖ 3D objects in your scene that you want to use as doors/windows (there is also a Door Prefab included in the tool)

===

==

2.2. STEP-BY-STEP GUIDE

==

1) Import the asset
2) Create the tag 'Door' in the tag manager
   Edit>Project Settings>Tags and Layers
3) Click-and-drag the 'Detection' script onto your character/player (or just use the Player Prefab)
4) Click-and-drag the 'Door' script onto ALL of your doors/windows (or just use the Door Prefab)

ENJOY!

==

2.3. CHANGE KEY INPUT TUTORIAL

==

By default the script expects the user to click the key 'E' on their keyboard to open or close the door. You might want to change this and this is a quick tutorial that will show you how to do so.

Open the 'Detection.cs' script in your code editor of preference, and use the ctrl+f function to look for the line in the following picture. If your editor doesn't have this function, manually look for the line of code. It should be around line 84.

```
84        if (Input.GetKey(KeyCode.E))
```

Then change the piece of code 'KeyCode.E'. For regular letters it will be 'KeyCode.A' for the letter A or 'KeyCode.B' for the letter B, but for some more complex symbols, you should look the code up in the scripting reference on the Unity3D website.

Link: https://docs.unity3d.com/ScriptReference/KeyCode.html

Next you need to click on the text element prefab in your project files (or any other text element you wish to use). And then you click and drag it onto your hierarchy. Click on it and find the text element, it'll look like the picture below. Then you just change the 'Text' field with whatever you want.

==

===

2.4. TROUBLESHOOTING AND KNOWN BUGS

===

The only known 'bug' is that when a door rotates, it will choose the shortest

distance between InitialAngle and RotationAngle, and this might not always be what you

had in mind. (a workaround for this is in the making)

Also make sure your door has a collider component attached to it, otherwise the

script won't be able to detect the presence of a door/window.

===

===

2.4. COMPATABILITY WITH OTHER PACKAGES

===

The door script is currently known to be compatible with the following packages on the Unity
Asset Store.

-       UFPS (Ultimate FPS) https://www.assetstore.unity3d.com/en/#!/content/2943)

NOTE: UFPS might use the key 'E' for some of its own functions. You'll need to follow the tutorial in
2.3. in order to fix this.

The door script should work with every package on the asset store but this is not guaranteed of
course. If you run into any issues, feel free to contact me.

==

3. OVERVIEW SCRIPT

==

3.1. DETECTION SCRIPT

==

3.1.1. DETECTION SETTINGS

==

Detection Settings
Reach   4

Reach	Within this radius the player is able to open/close the door/window. Equals the length of the ray cast.

Example:
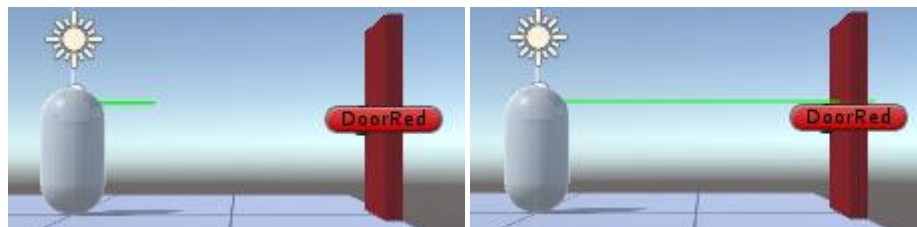
Reach = 1   Reach = 5

===

3.1.2. UI Settings

===

UI Settings
Text Prefab   TextPrefab
Crosshair Prefab   CrosshairPrefab
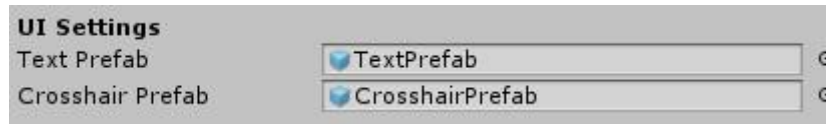
Text Prefab	The text element that gets displayed when the player is in reach of the door. Use 'TextPrefab' to see this in action.
Crosshair Prefab	The crosshair that gets displayed in the middle of the screen. Use 'CrosshairPrefab' to see this in action.

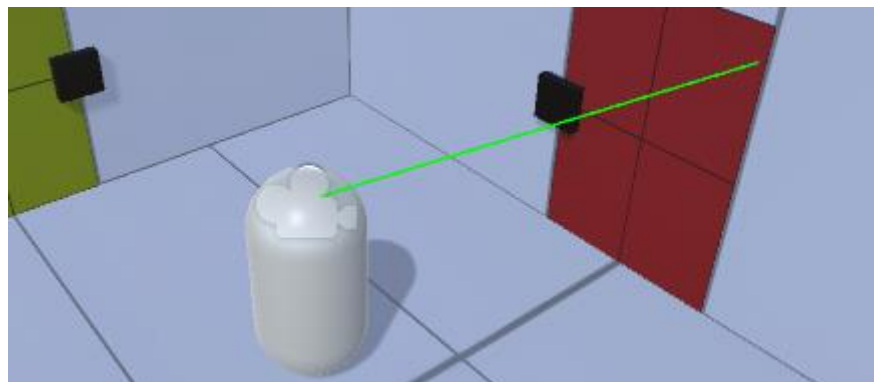===

===

3.1.3. DEBUG SETTINGS

===

Debug Settings
Debug Ray Color
Opacity   1

Debug Ray Color	The color of the line shown in scene-view that represents the ray cast.
Opacity	The opacity of the debug ray.

===

===

3.2. DOOR SCRIPT

===

3.2.1. DOOR SETTINGS

===

Door Settings

Initial Angle	0
Rotation Angle	90
Hinge Position	Right
Rotation Side	Right
Speed	3
Times Moveable	0

Initial Angle	The initial angle of the door/window.
Rotation Angle	The amount of degrees the door/window rotates.
Hinge Side	The hinge side determines which side of the door/window the door/window rotates around.
Speed	Rotating speed of the door/window.
Times Moveable	Determines how many times the player is able to move the door, if set to zero, the player will be able to open the door an infinite number of times.

===

3.2.2. DEBUG SETTINGS

===

Debug Settings

Visualize Hinge	☑
Hinge Color	

Visualize Hinge	Visualizes the position of the hinge in-game by a colored cube.
Hinge Color	The color of the visualization of the hinge.

===

4. COPYRIGHT AND CONTACT

===

Copyright:

All the scripts, scenes, 3D objects and materials in this asset are created by me except for these:

- ❖ The 'MouseLook' script
- ❖ The 'PlayerMovement' script

The two scripts are from this website:

> http://wiki.unity3d.com/index.php/Scripts

> The contents of this website are available under 'Creative Commons Attribution Share Alike'.

These scripts are included in the asset for the purpose of example only.

You are free to use my script for both commercial and non-commercial use.

Contact:

For support and feedback:

Email: alexanderameye@gmail.com