

## Fungus + Quarantine Framework Setup Instructions

Fungus + Quarantine Framework Setup Instructions .....	1
1. Step by Step .....	2
1.1. Create New Project .....	2
1.2. Import Framework (v10) .....	2
1.3. Check Console for Errors .....	3
1.4. Create basic test scene .....	6
1.5. Import Fungus .....	7
1.6. Framework Modifications .....	8
FpsPlayer Prefab 1 .....	8
PlayerControlManager Prefab.....	12
Test the new Player .....	15
Add a Test Flowchart .....	16
FpsPlayer Prefab 2 .....	19
Flowchart 2.....	21
Enjoy! .....	24

## 1. Step by Step

### 1.1. Create New Project

This section covers implementation of both Quarantine Framework and Fungus in a blank project, skip to Framework Modifications if you already have a project set up.

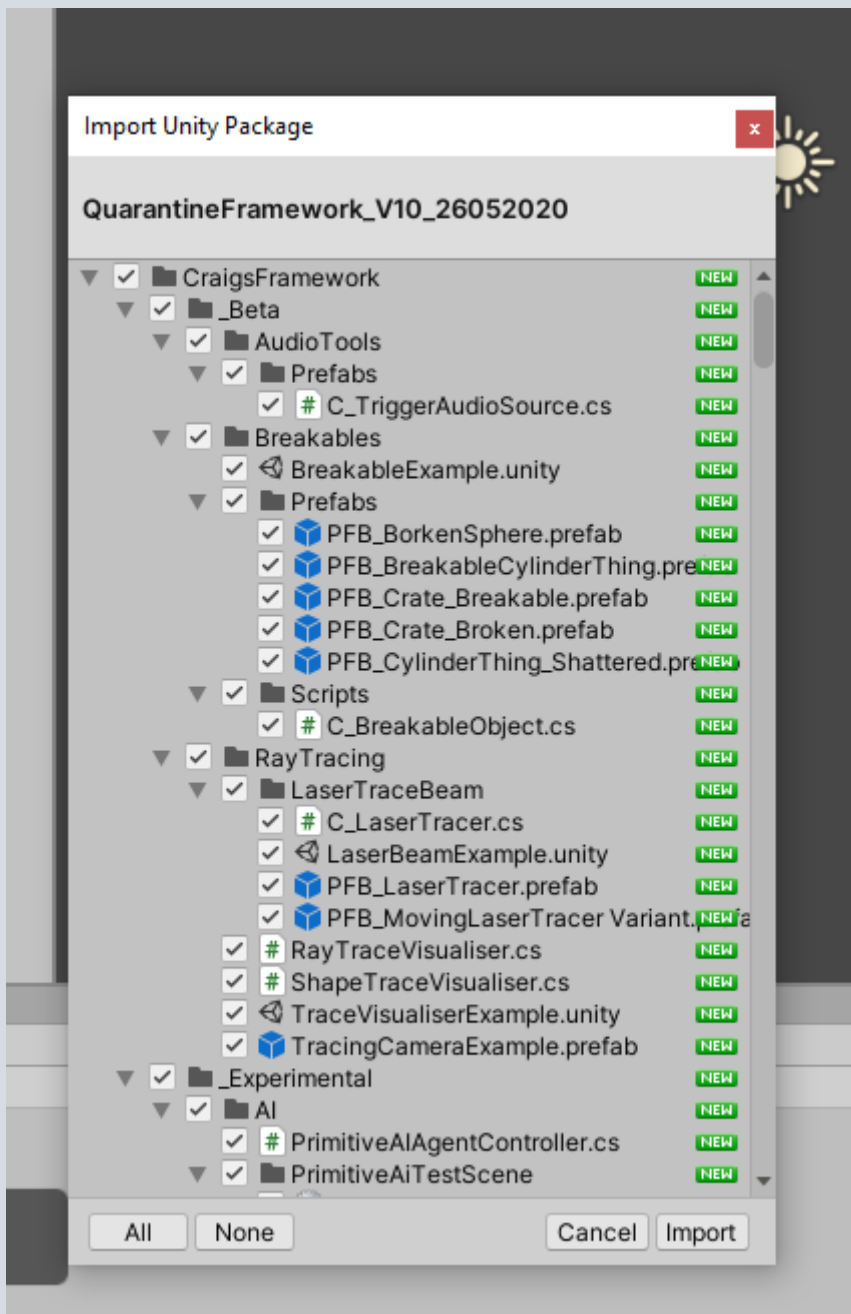
### 1.2. Import Framework (v10)

Framework V10

[Download Unity Package](#)

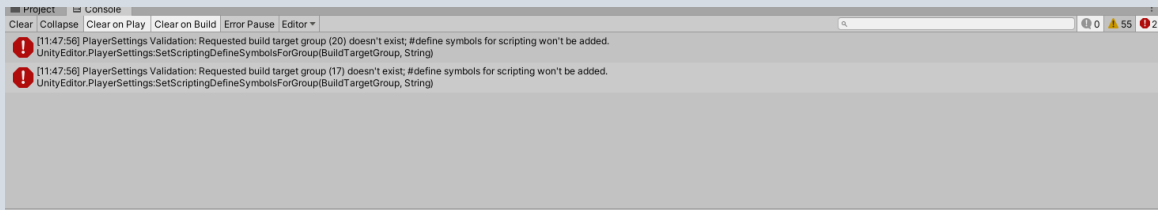
Framework Help Doc

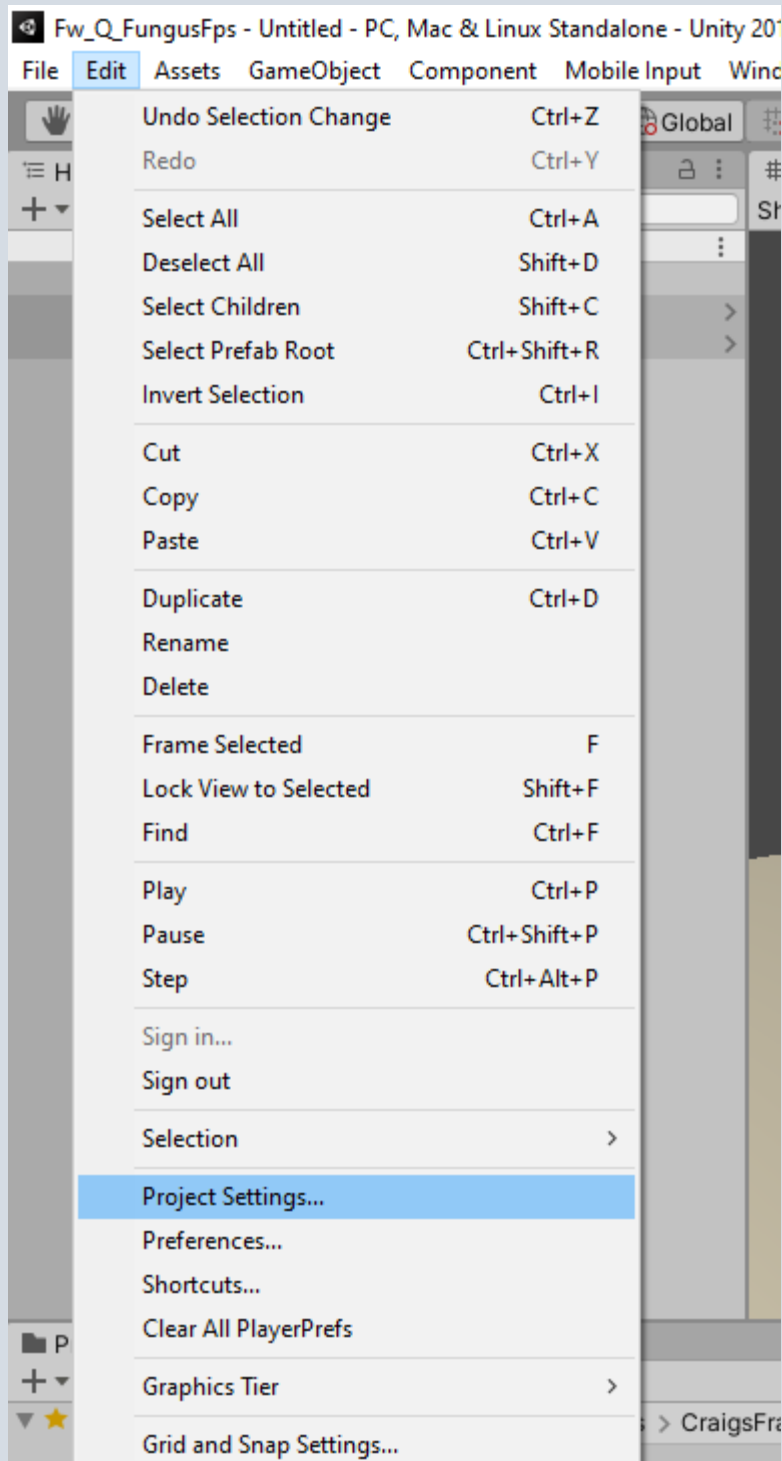
[View Online](#)

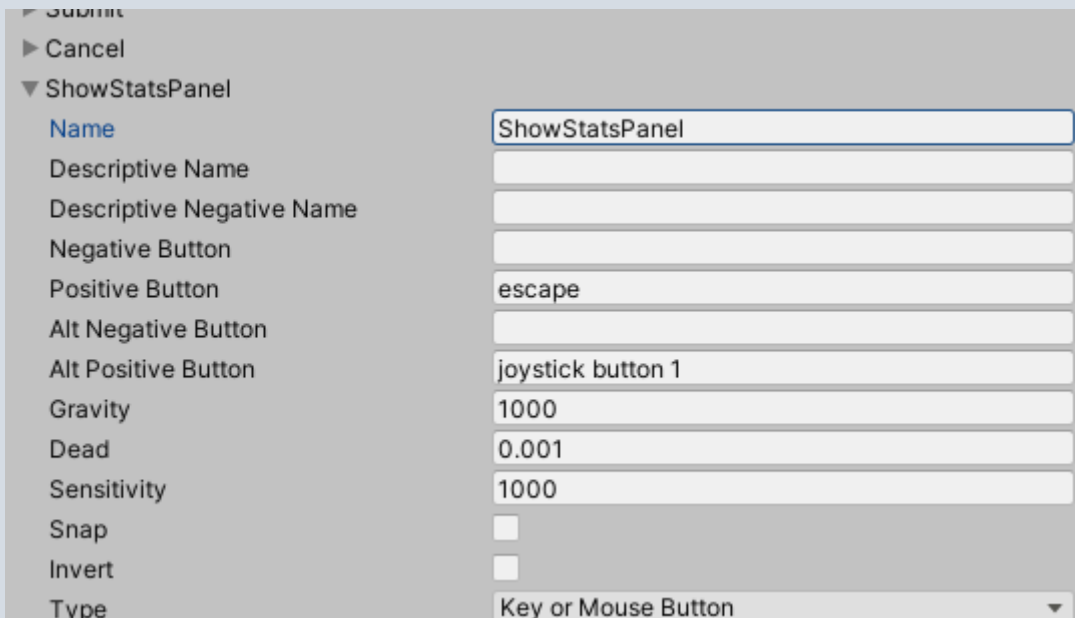
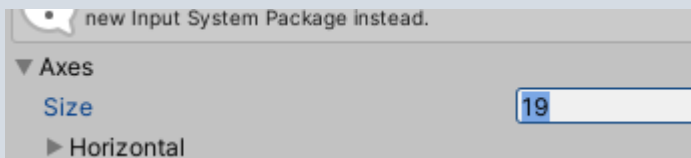
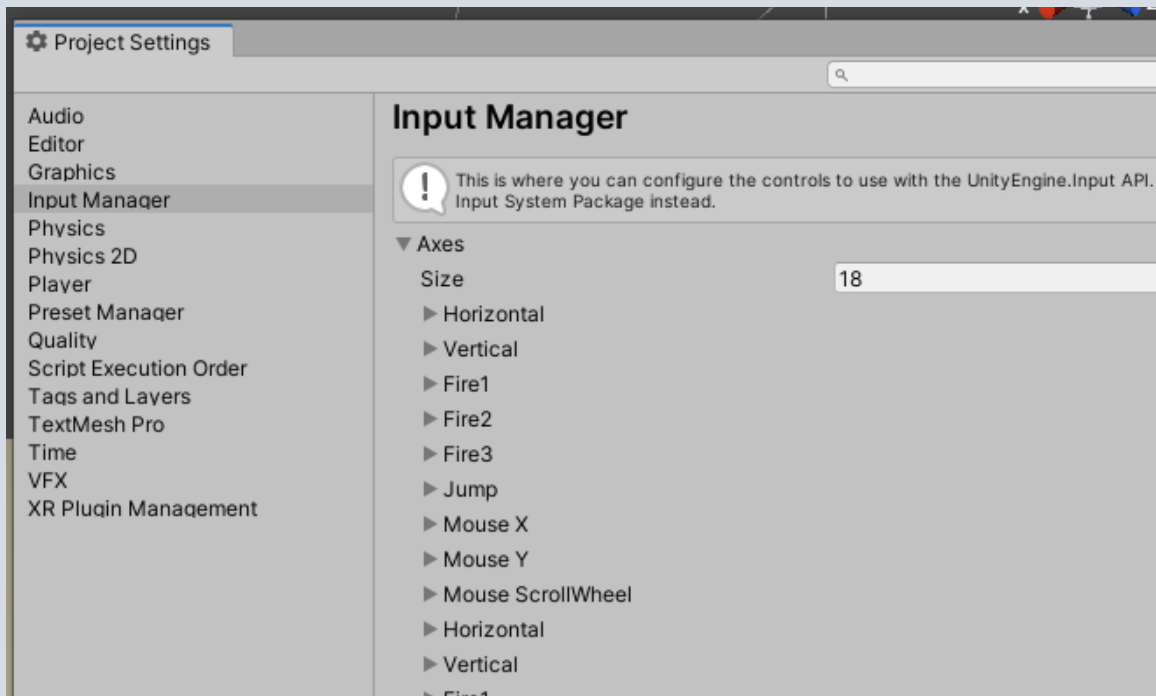


### 1.3. Check Console for Errors

You will receive an Error stating that the ShowStatsPanel input is missing, refer to the Framework Help documentation for detail, ensure you add an Input named ShowStatsPanel to your project, and map it to a suitable key, i use Tab.

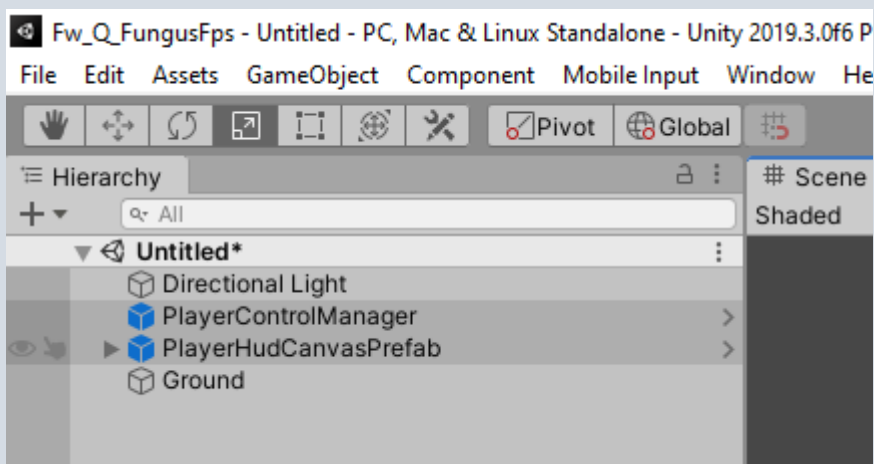
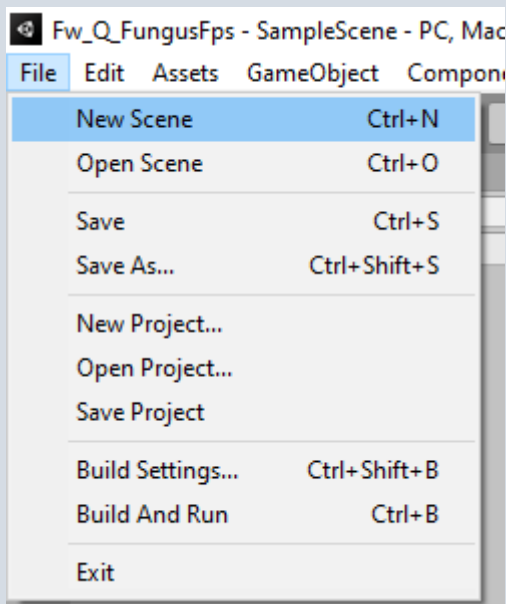






#### 1.4. Create basic test scene

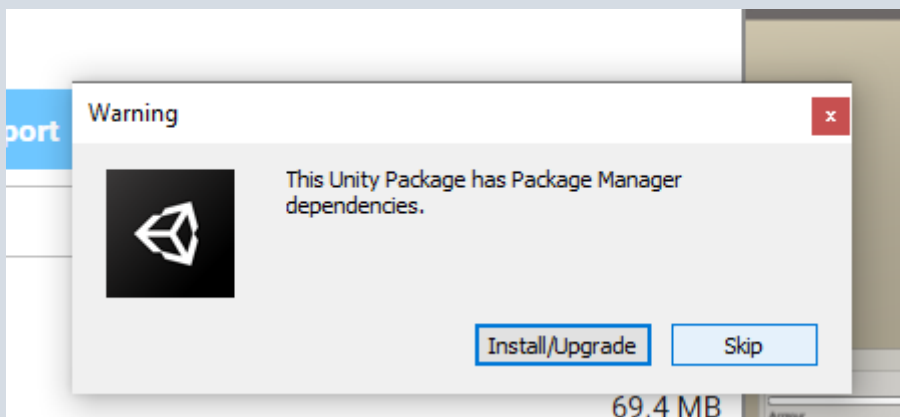
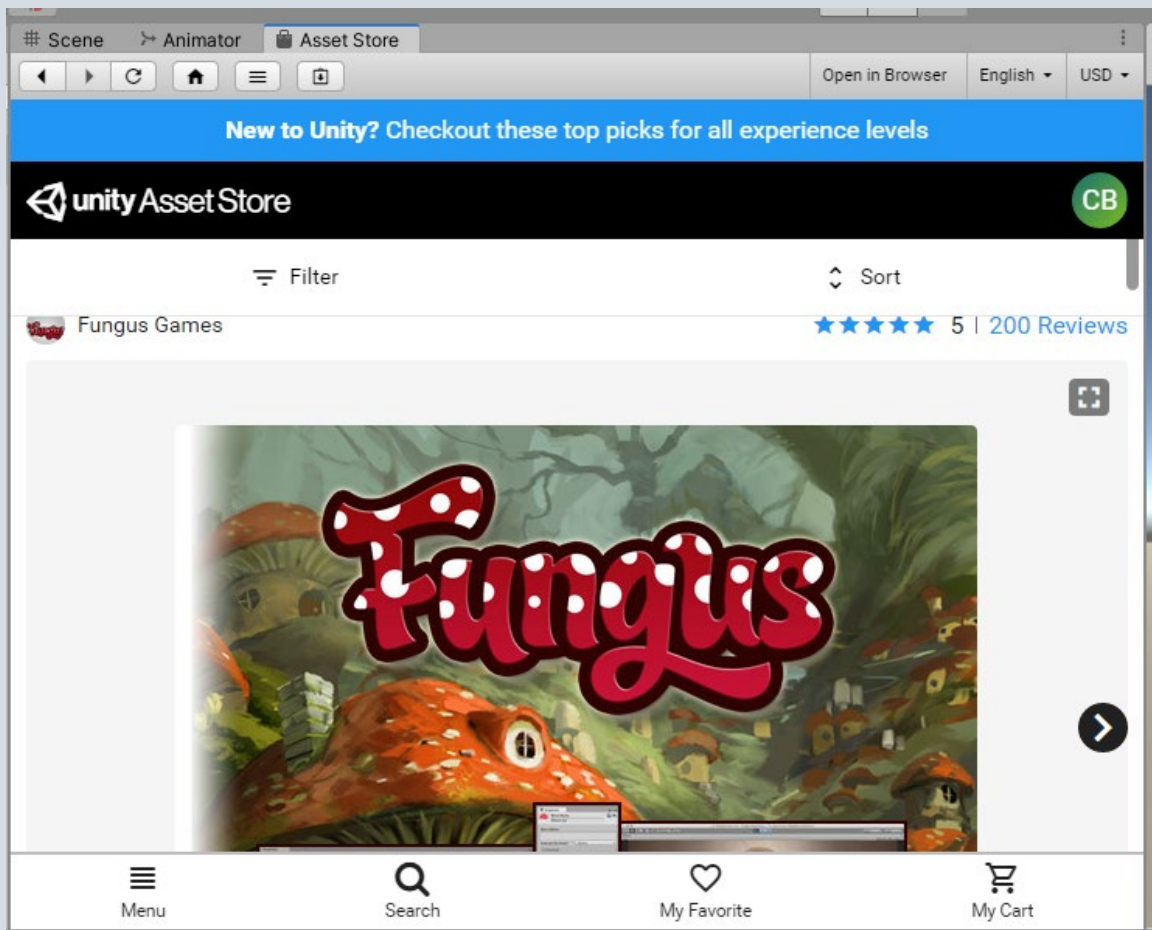
Refer to the Framework Instructions for detail, ensure you have both `PlayerControllManager` and `PlayerHubCanvasPrefab` added to the scene, and delete the existing `MainCamera`.



### 1.5. Import Fungus

Import the Fungus Unity Package through the Asset Store.

When asked choose 'Skip dependencies'.



## 1.6. Framework Modifications

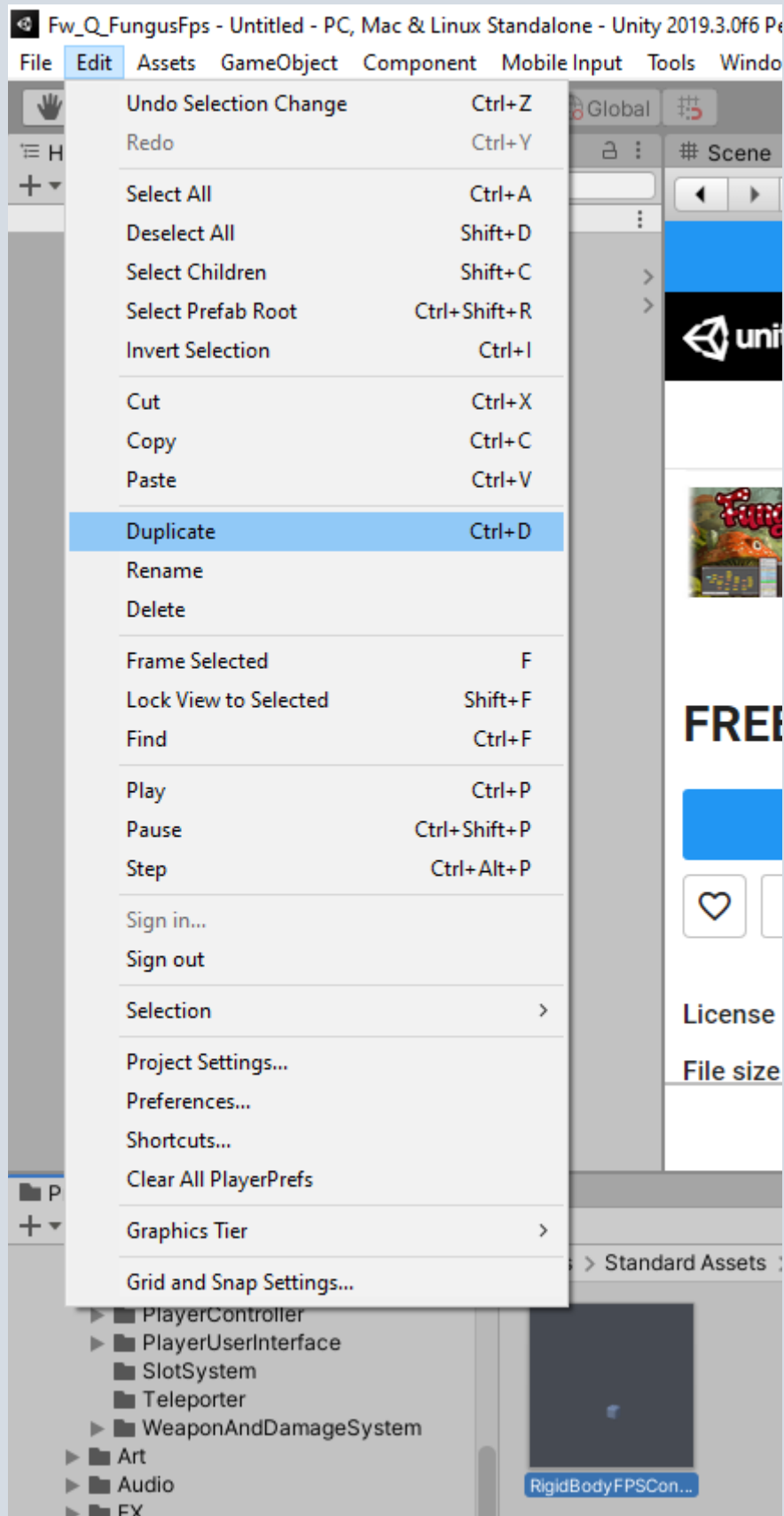
FpsPlayer Prefab 1



The first phase of modifications we will make to our existing Player Prefab object.

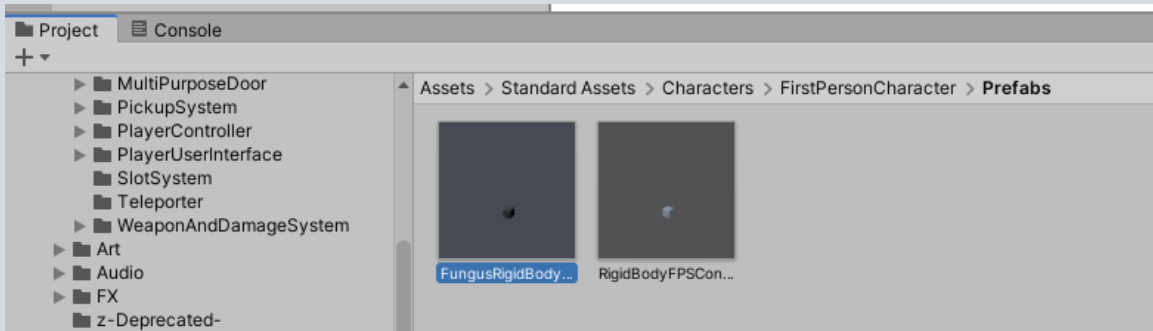
### Create a Duplicate of the RigidbodyFpsController Prefab

Select prefab in Project View, then Ctrl+D or Edit>Duplicate to clone the Prefab.



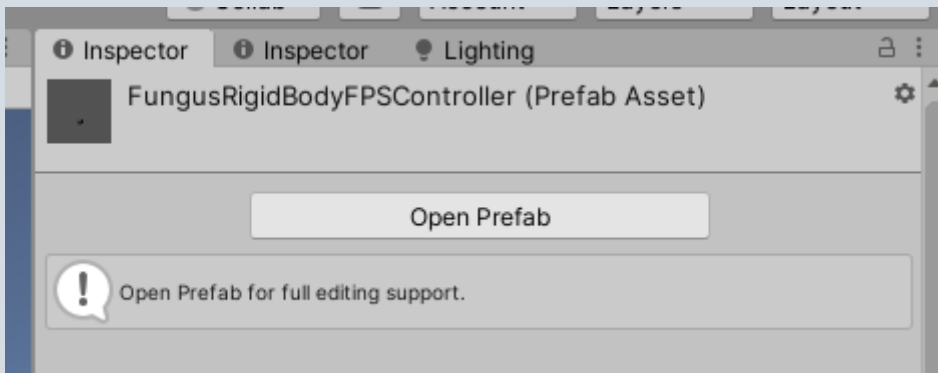
Rename to "FungusFpsRigidBodyFpsController"

This will help us differentiate this prefab from the original



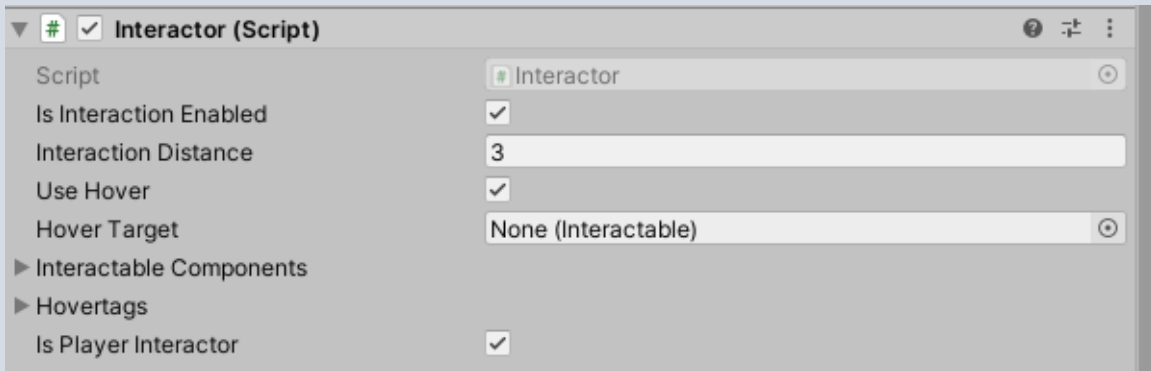
Open the newly duplicated prefab

Double-click asset or "Open Prefab"



Add an Interactor component to the FungusRigidBodyFpsController  
GameObject (the root of the Prefab)

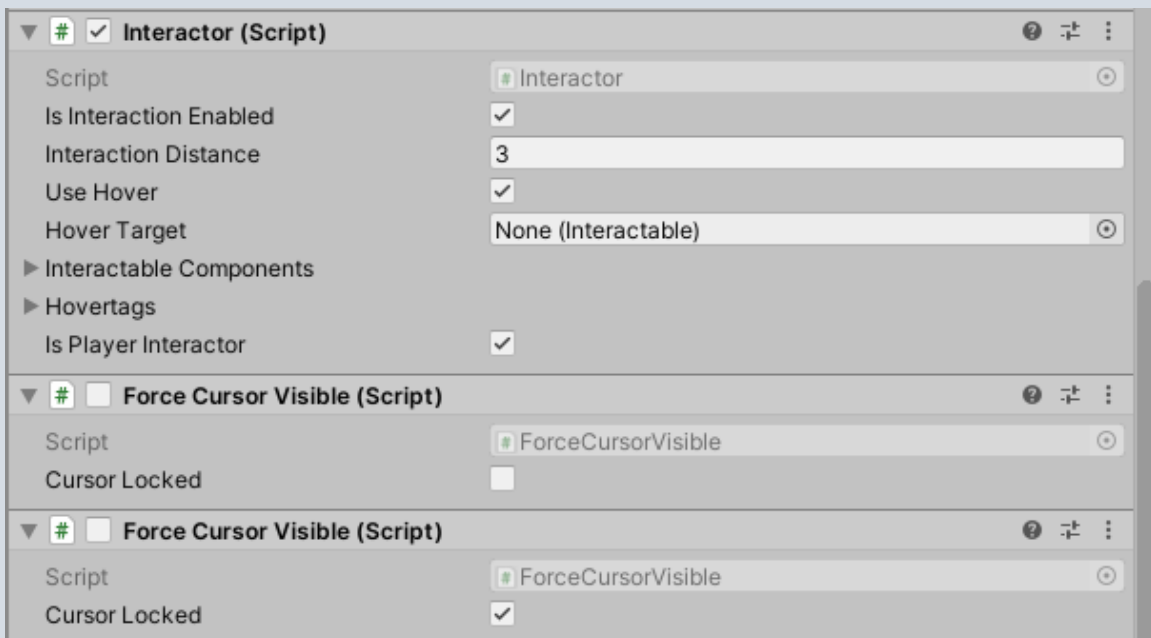
Tick "Is Player Interactor"



Add 2 "Force Cursor Visible" components also

Un-Tick "Cursor Locked" on Component 1

Tick "Cursor Locked" on Component 2



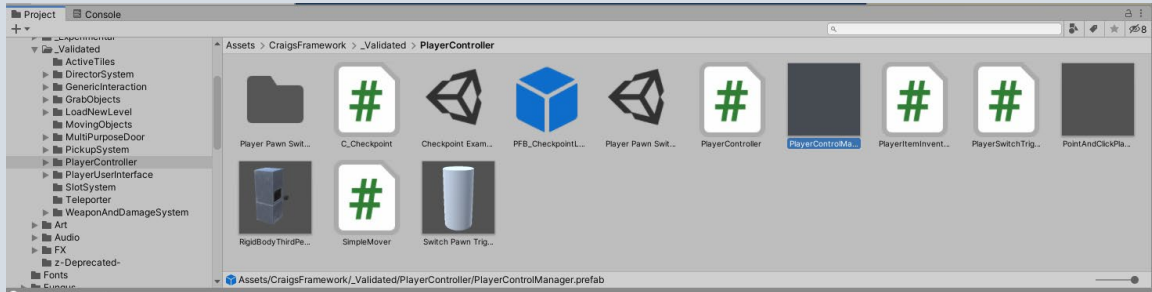
Disable both components

Fungus will handle turning these components on and off during gameplay.

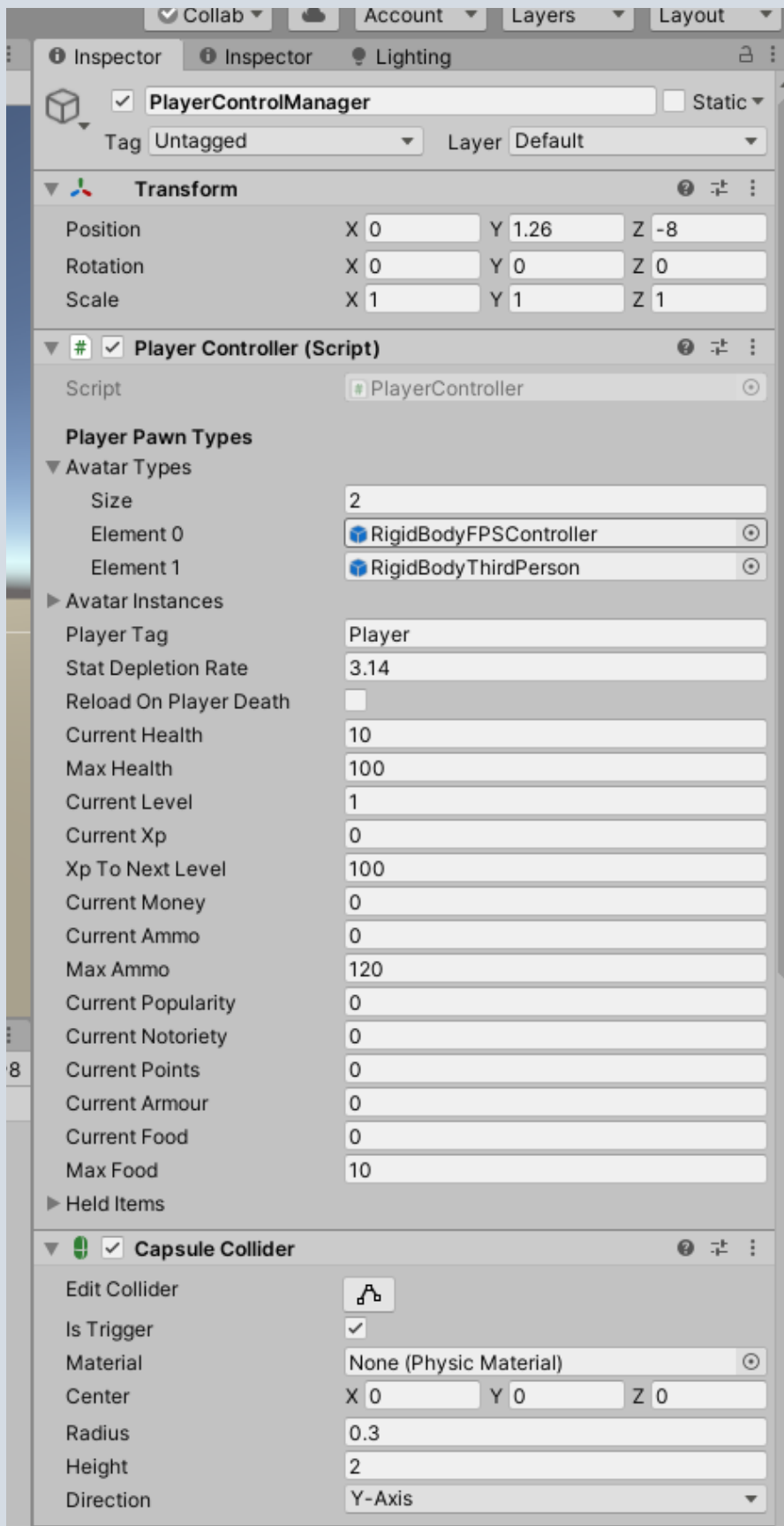
PlayerControlManager Prefab

We will modify the PlayerControlManager prefab to use the new FungusFpsPlayer prefab.

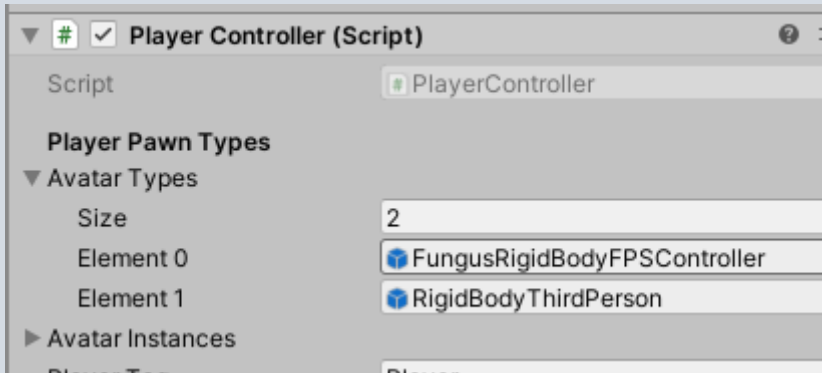
### Open the Prefab



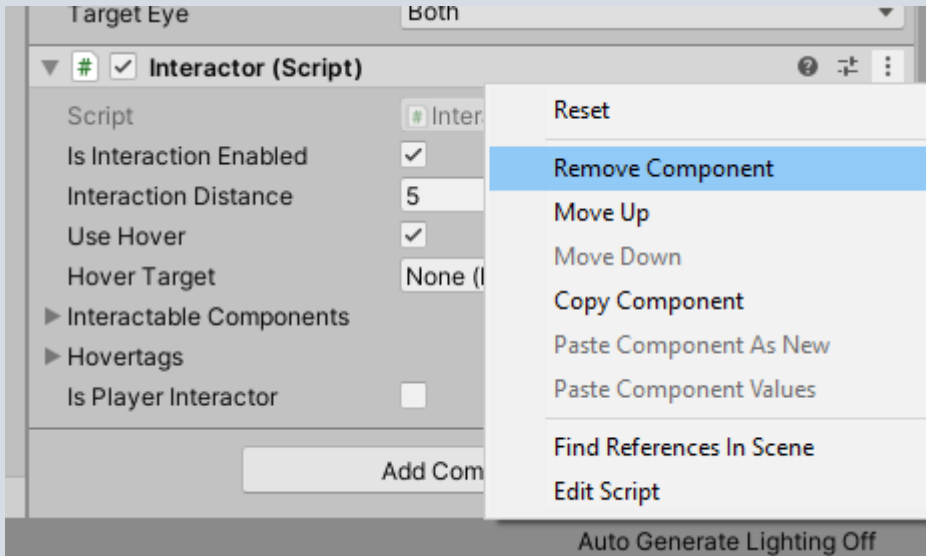
### Expand the "Avatar Types" array under Player Pawn Types



Replace the Element 0 with your new FungusFPS prefab



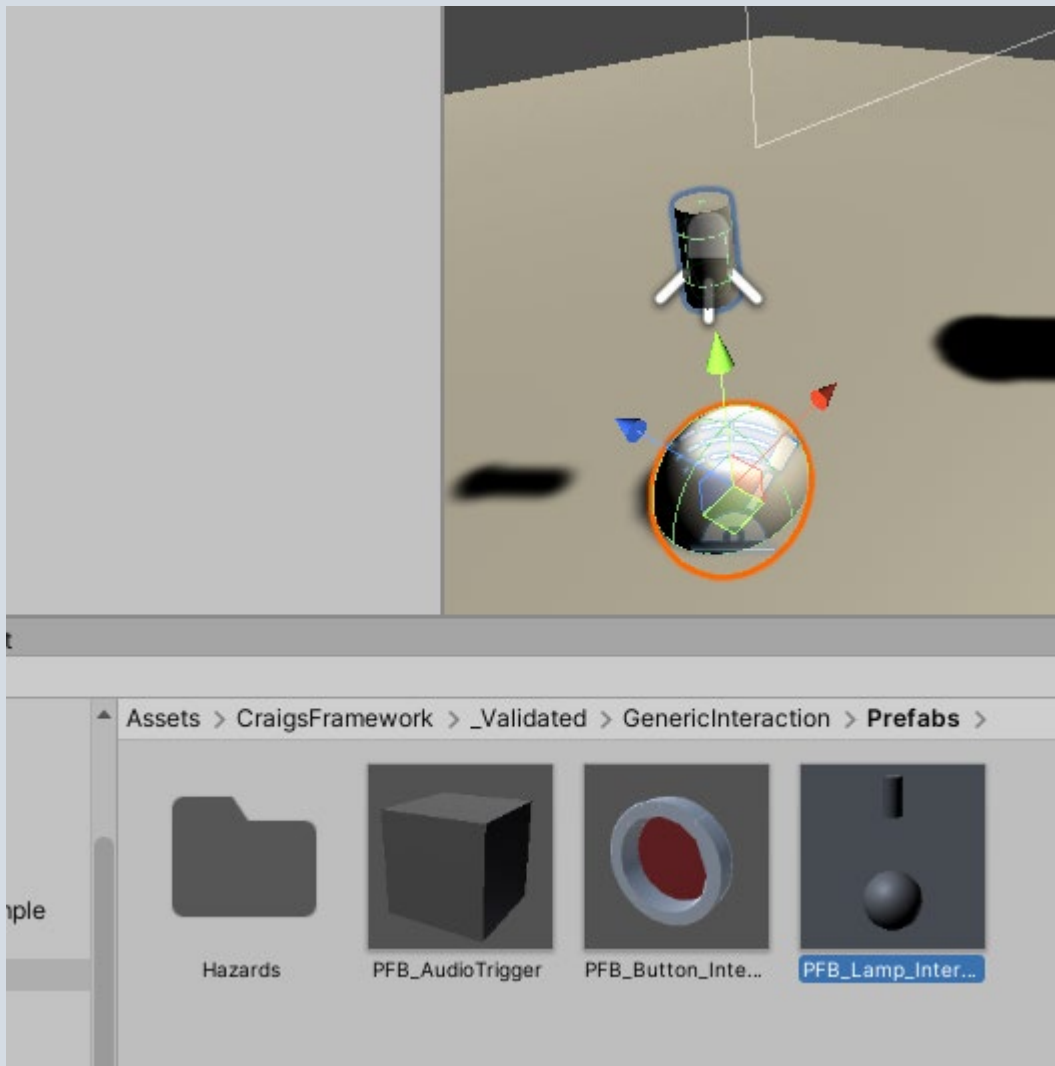
Remove the Interactor from the PlayerControlManger prefab



Test the new Player

Create a new Interactable or drop an existing Interactable Prefab into the scene

This example uses the PFB\_Lamp\_Interactable, it will switch an associated SpotLight on or off. It will be suitable for testing the new Interactor Setup.

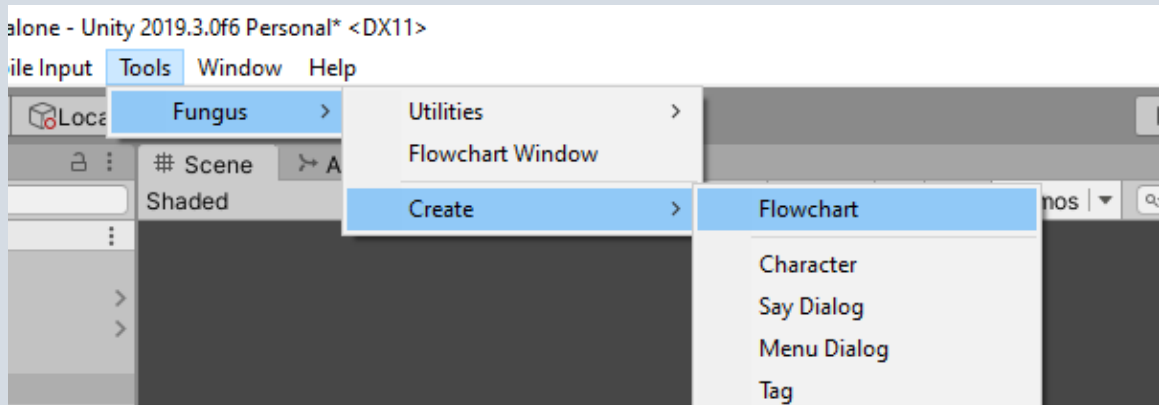


## Test Interaction

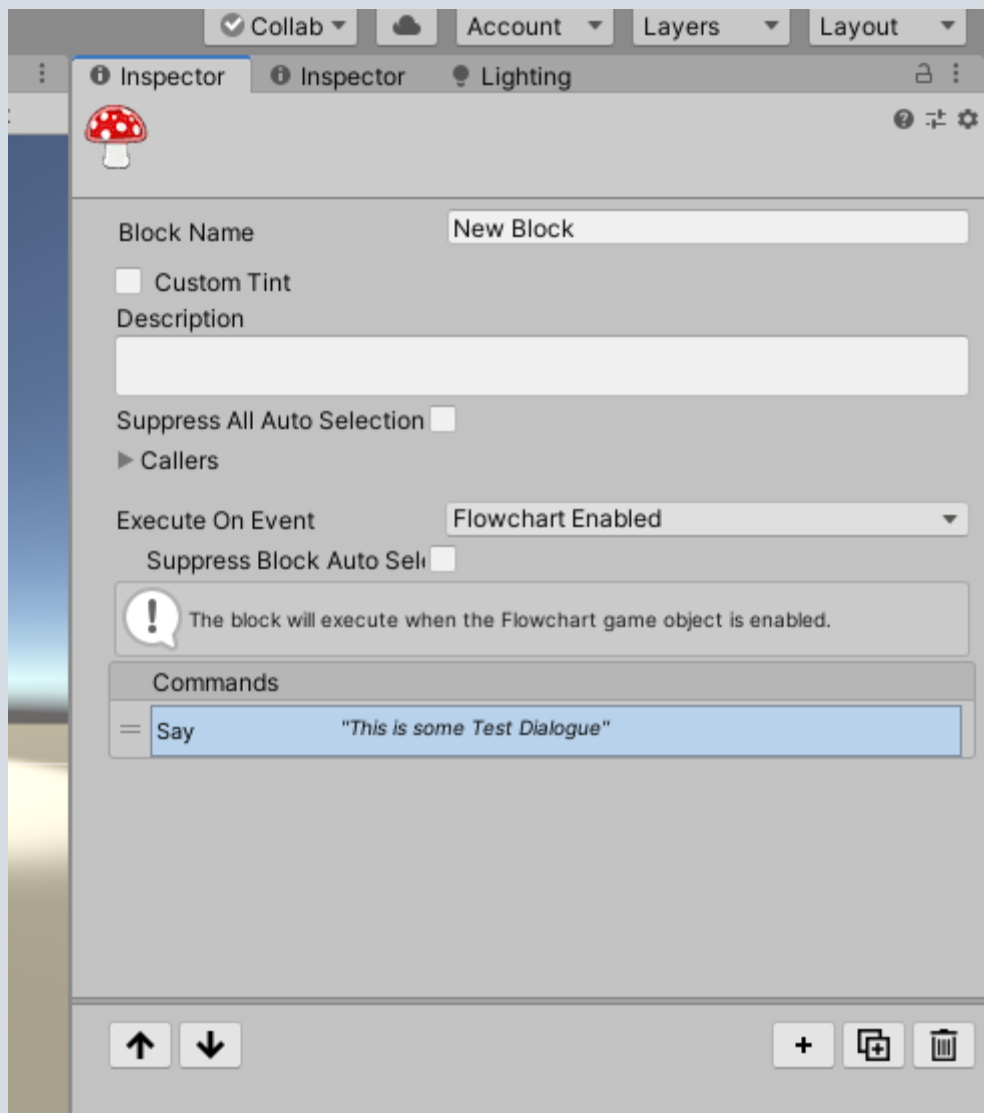
Playtest the scene to ensure you can interact normally.

## Add a Test Flowchart

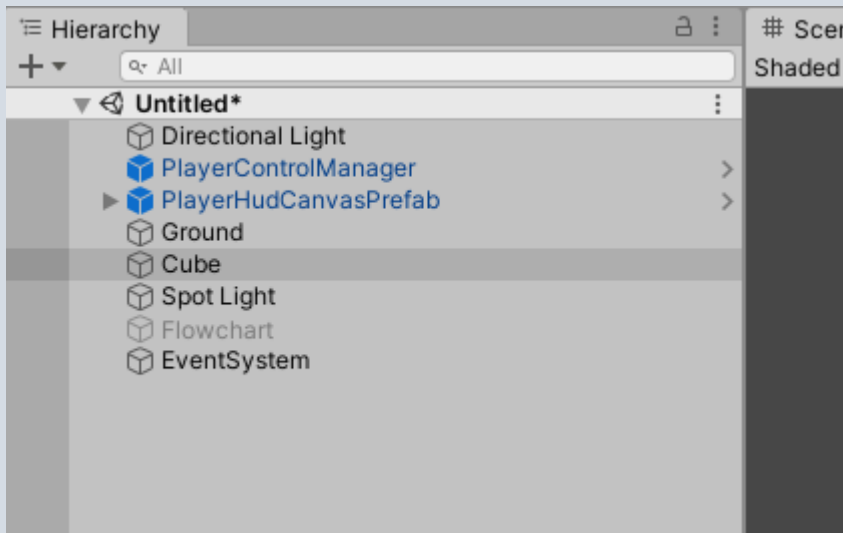




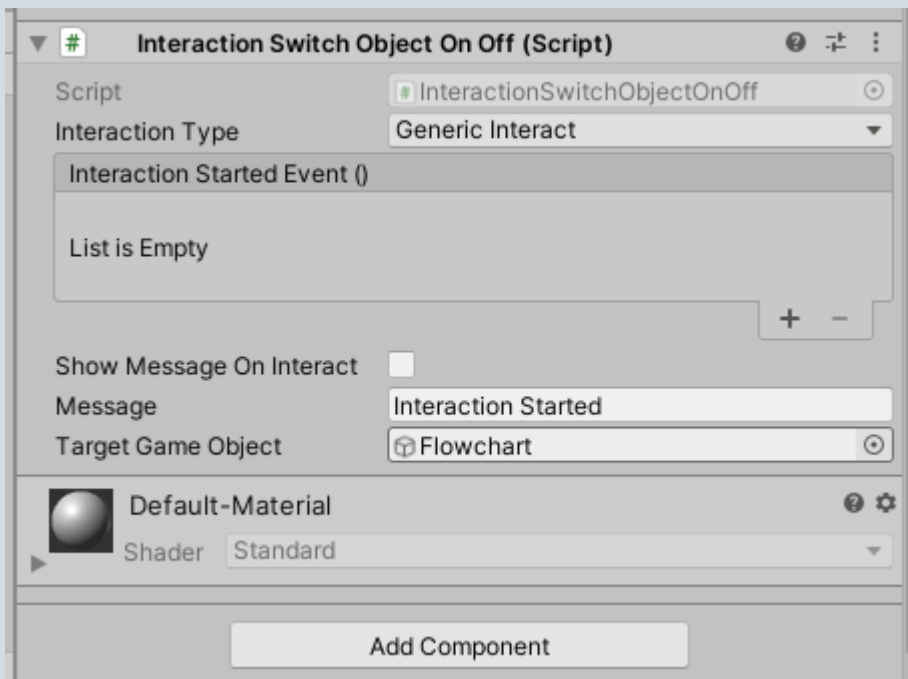
Set the Execute on Event to "Flowchart Enabled"



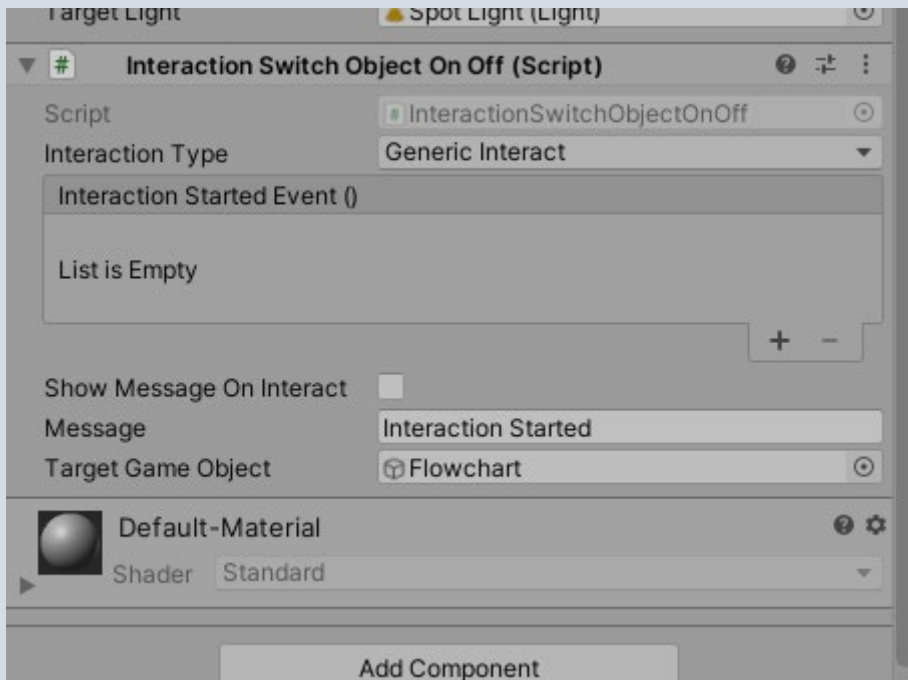
Disable the Flowchart GameObject



Add an InteractionSwitchObjectOnOff component to your Interactable (Box)



Drag the Flowchart from the Hierarchy into the Target Game Object slot



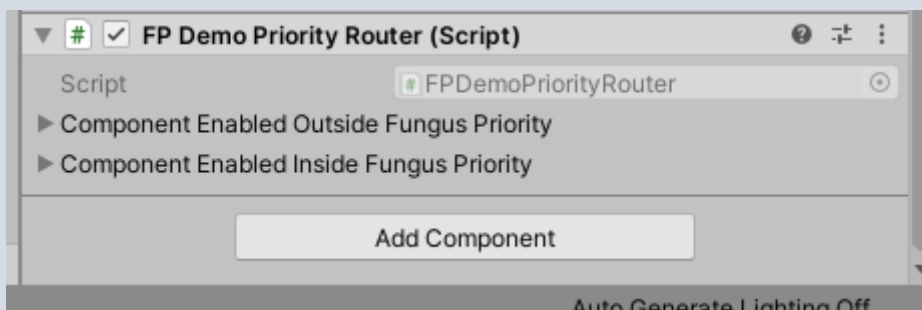
Test the Flowchart starts when you Interact with the Cube

FpsPlayer Prefab 2

Open the "FungusFpsRigidBodyFpsController" prefab again

Add Fp Demo Priority Router to the FPS Prefab

These allow us to interpret Fungus Priority changes, and turn particular Components on or off.



Expand the Component Enabled and Disabled arrays

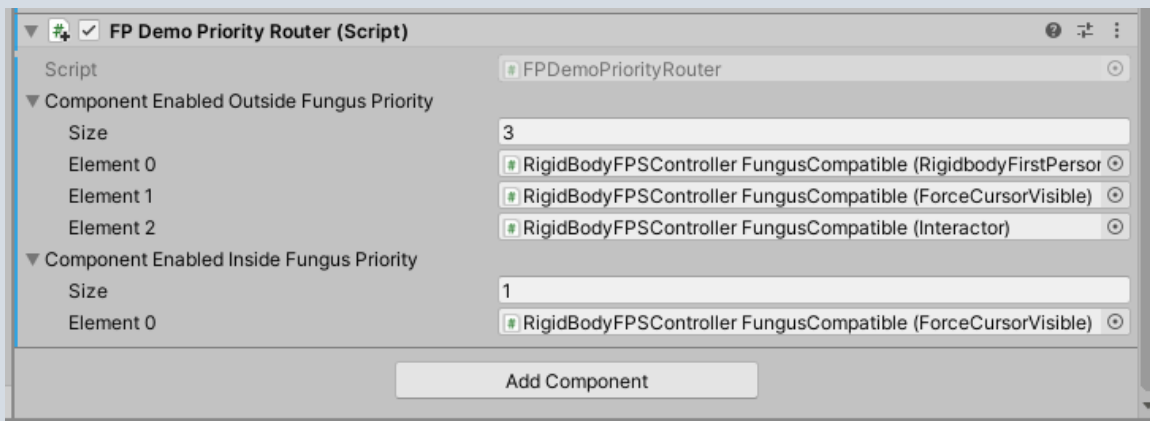
Set the sizes to 3, and 1

When Fungus Priority Increases; we will turn On the Cursor, Disable Player Movement, and Unlock the Cursor so the user can click on the UI.

We will switch all these changes back Off when the fungus Decreases Priority

Drag the relevant components to turn on/off when Fungus priority changes

You can drag the Components into the slots by dragging their Title Bar down onto the relevant spots in the Priority Router



When Fungus loses Priority

Enable Interactor

Enable Mouselook

Enable Interactor

Toggle Cursor Lock Mode and Visibility (Disable)

When Fungus is Priority

Disable Interactor

Disable Mouselook

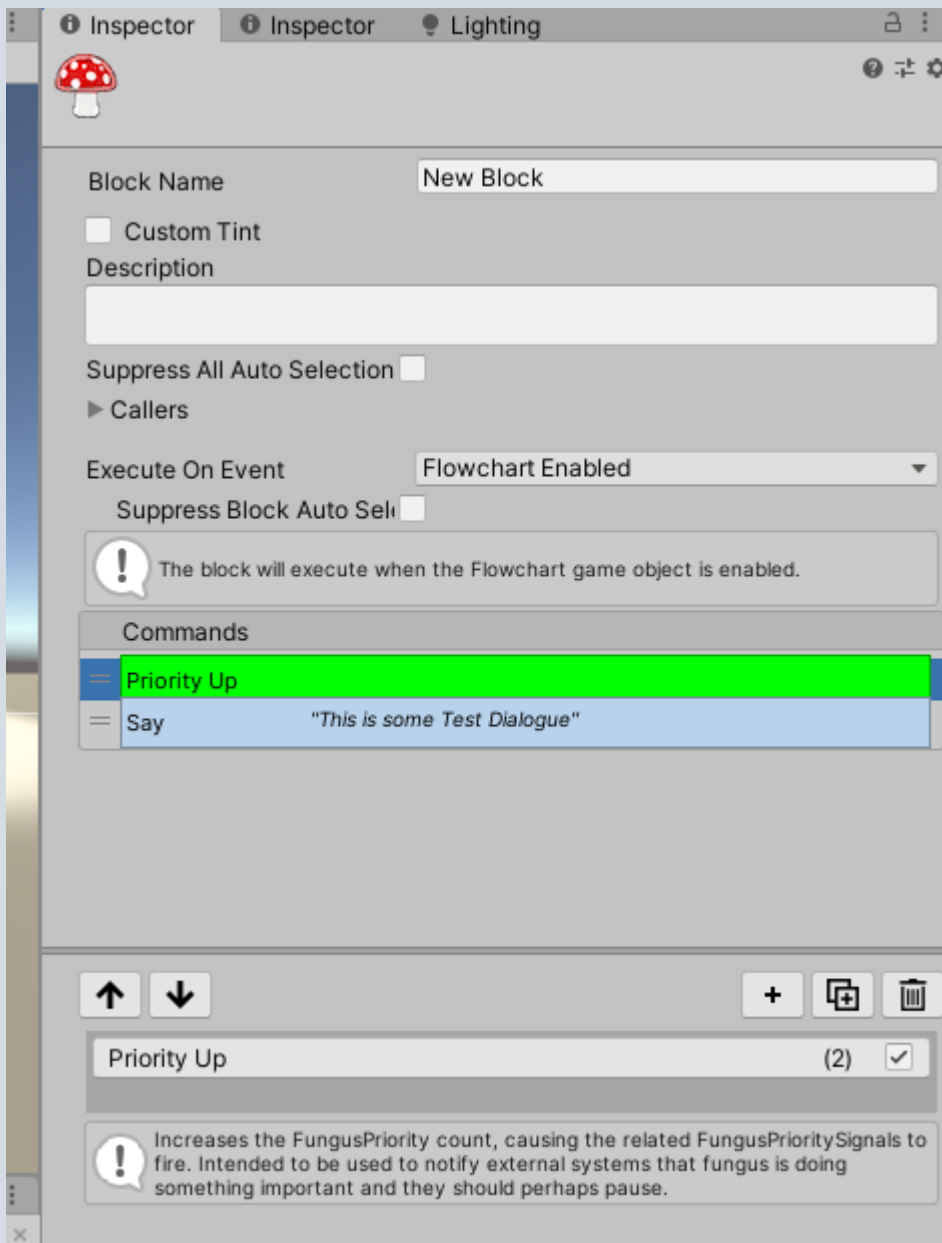
Disable Movement

Toggle Cursor Lock Mode and Visibility (Enable)

## Flowchart 2

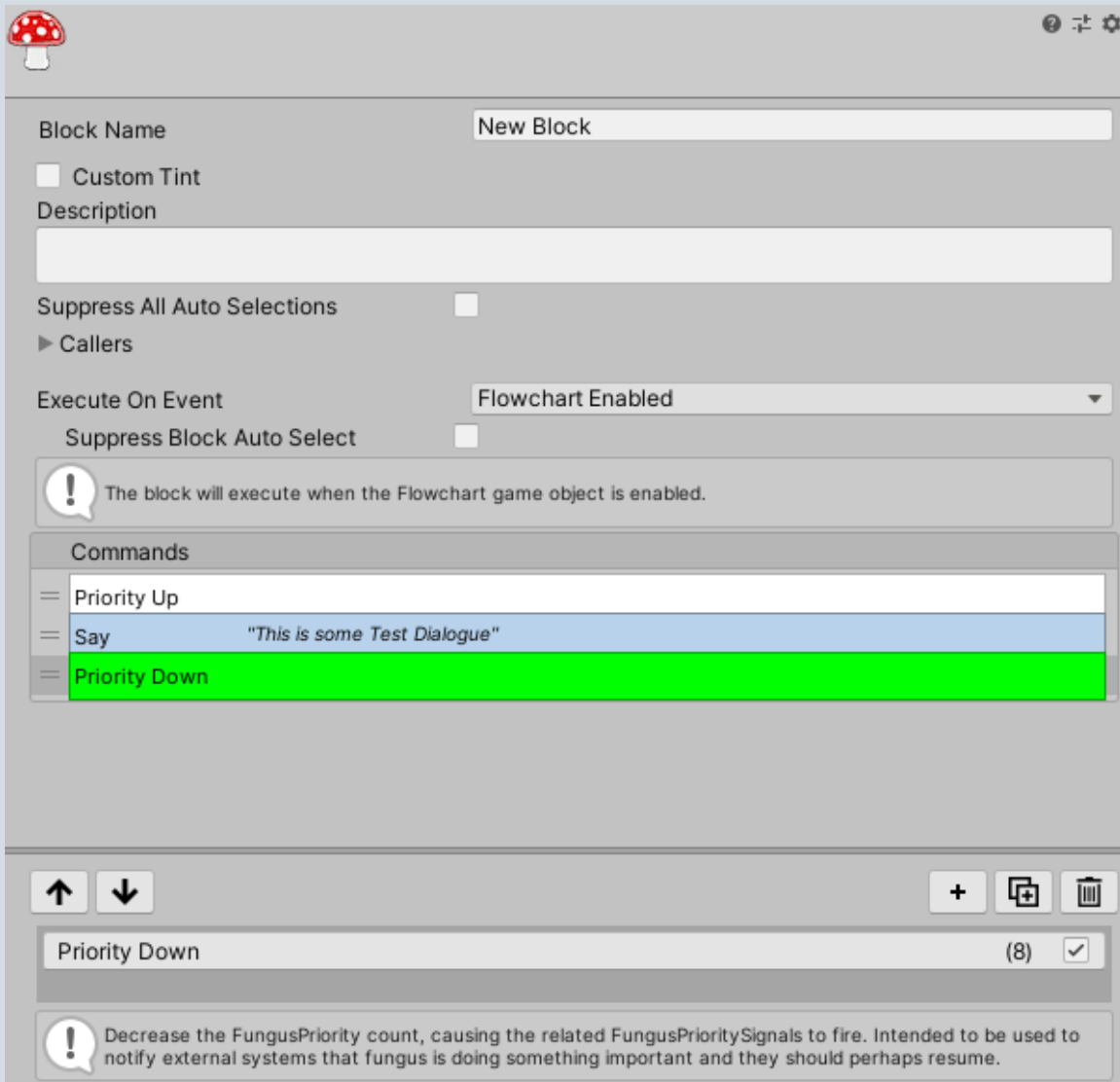
Add a "Fungus Priority UP" element to your Block as the first Element

This will cause Fungus to signal at our FungusPlayer that it should enable/disable the components we provided. This enabled the Cursor, UI Interaction and Disables movement so we can ensure the dialogue or interaction is completed before allowing direct interaction and movement again, when the FungusPlayer receives the Priority Down signal.



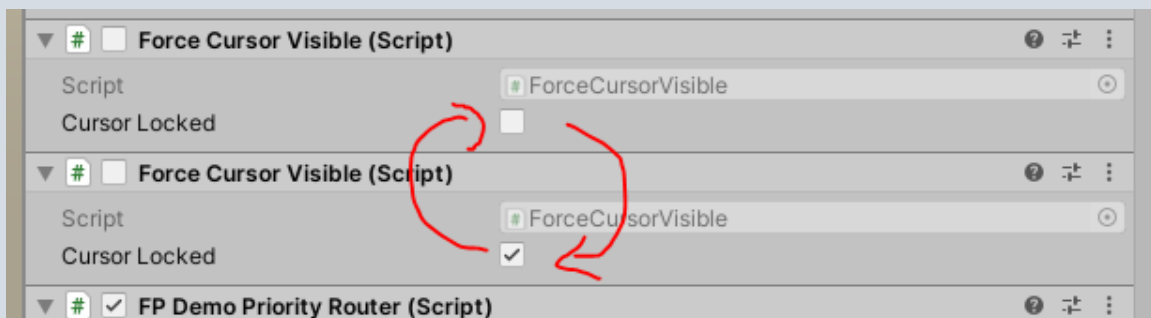
Ensure the Priority Up node precedes any Say or Menu element

Add a Priority Down node to return control to the Player



### Test level and Fungus Interaction

If your Cursor does not appear when in Fungus Mode, try swapping the Force Cursor Visible checkboxes



Enjoy!

You can now experiment with using fungus as a method for storing gameplay variables, comparing values, and providing the player with choices, this opens up your possibilities immensely, get creative!