2d Escape Engine User Manual

By Hitcode

Version 1.44

Introduction

An escape game are games like such as Rusty Lake series , Door & Room or Lost Island.
They are extremely famous game and now you can make your own like them with our engine.
The engine supply several ready made mini-games so theoretically you can made a game without any script by just click and choose.
But of course, for high commercial usage, it is better to have your own excellent creative.
The engine would supply Interfaces easy for you to add your own things.
As it was a script based engine, it has no conflict with any other engine or framework.
You can choose your prefer tools such as playmaker to enrich your game content.
And of course if you are a professional of C# script, you can do anything you want to add your own game function because the game engine is highly compatible with unity engine and it was even not rely on the engine itself.

Requirement

When you start to make a game with an engine. You at least require to know basic of unity.
And at least you would have got your own game logic thought. If you know nothing about making games. It would be a bit hard for you to start. And of course, it would not be a good question like about: how can I make this game, how can I make that game?
Engine would only helps to do the job. Thoughts are depending on you.

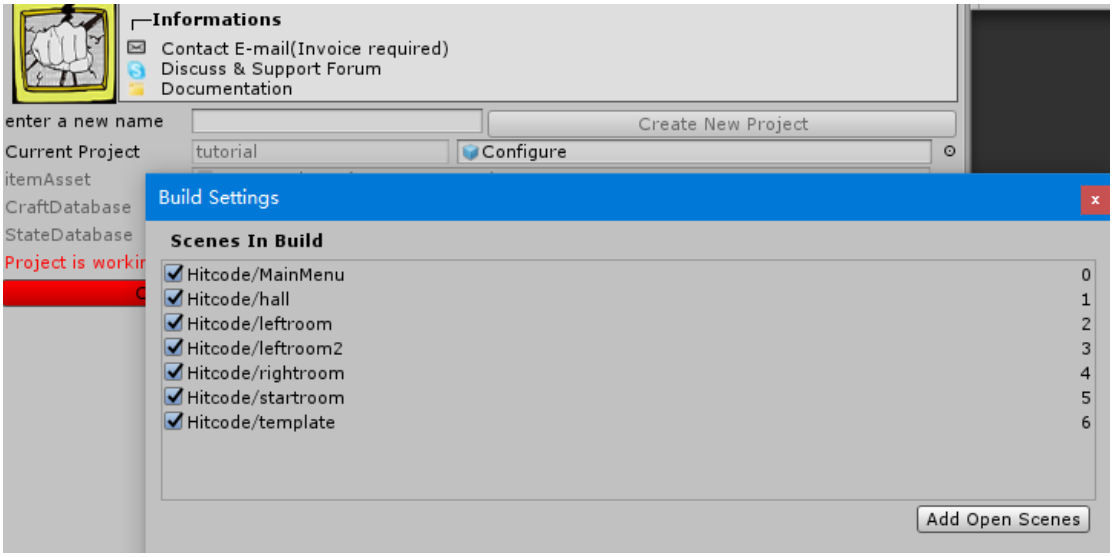
Install the package(updated)

1. Create a new and clean new project.
2. Assets->import package to import the project.
3. You should set up Dotween for current version.If you got error during an update.Delete Tweentool folder under asset/hitcode/script/tools folder.
4. Open Unity asset store or the package manager,download Dotween and finish the setup as below first.

5. The game is already being set up on android platform.

How to play

The game already have an example for you to familiar with the engine functions.
You can start the game anywhere by activate any ready made game scene.
But if you want to try the full function of a game flow.
You can start the game from the scene name “MainMenu”.
This would start the game from a menu.
And also, if you do not make your own start game menu.
You can set up the “MainMenu” to the first scene in File->build settings.
This would help you build a new complete game easily.



Do not forget to add scene into build settings.
When start from the main menu, if you click continue, you would start the game from scene when you exit the game last time.
If you do not start the game from the main menu. This function would not work.

Update Details

v1.1
Fix a problem caused by sprite construct being deprecate by unity 2018
Fix mini-game – unblock’s inaccuracy move bugs.
Add a new game to minigame folder – The Tangram
v1.2
Added function: Jounal/Game Note system
Added Save/load system.
(webgl not support file save, so the webgl still using a single autosave)

The following tutorials are all based on the first release version. So there maybe minor differences with your new versions. But,the places which requires to be focus would be pointed out so don't worry about for understanding.

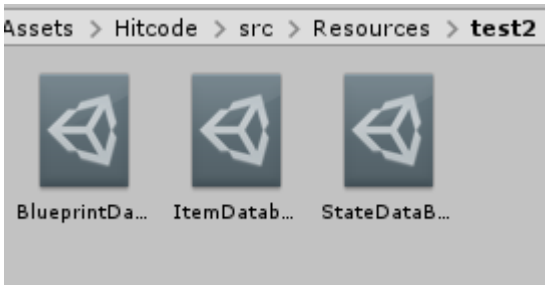
V1.4

Add Dialogue System.

Upgrade for old version customers

The 1.2 version got a big change so you should make a backup before overriding your old project. The version before 1.2 does not have a journal function. So doing the following to make the project working normally after importing.

1.Open your project folder. You should notice there are 3 config files. If you do not know this. You may follow the Chapter of Create New Projects to understand which and where was your project folder located.



2. Follow the Chapter of Create New Projects to create a new project by entering a new name. Under your new created project folder. There should be 4 new files like this now.

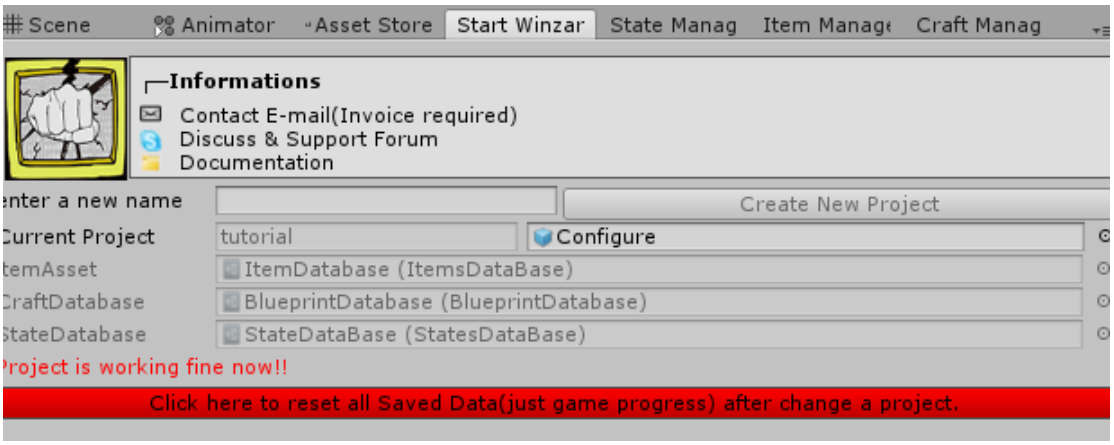
Copy the 3 files from step 1(from your old project configure folder)to this new created configure folder.

If there is no alert on start winzard panel. You should finish the job.

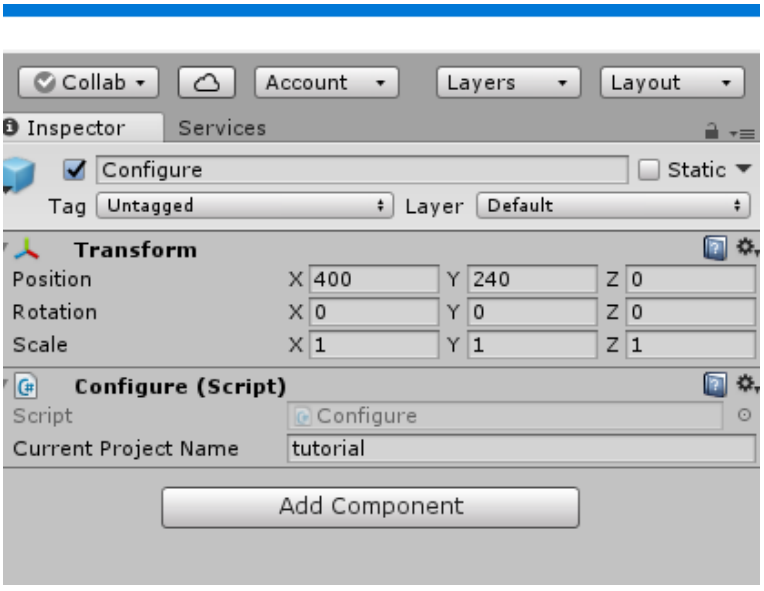
3.Remember the UI and graphic sources still would cover your old project. So make sure you did not lost your own picture sources after the override.

Create new projects

Activate the start wizard panel on Unity menu from Room Escape -> Create new Game



1. On "enter a new name" field. You enter an unique project name and press create new project button. The system would create a new folder under Assets/Hitcode/Src/Resources.
2. You can select the configure prefab in the project folder. And change projects on its inspector.



Change project just by input existed project folder name into "Current project Name" field.

3. If you have changed a project. You'd better click the long red button on the bottom of start wizard panel. This would clear the game saved data. And let you won't mess around non-existed saved data in a new project. Remember it only effects the gameplay but won't clear or delete any your settled work on settings. So be easily to click this button any time without any hesitation.

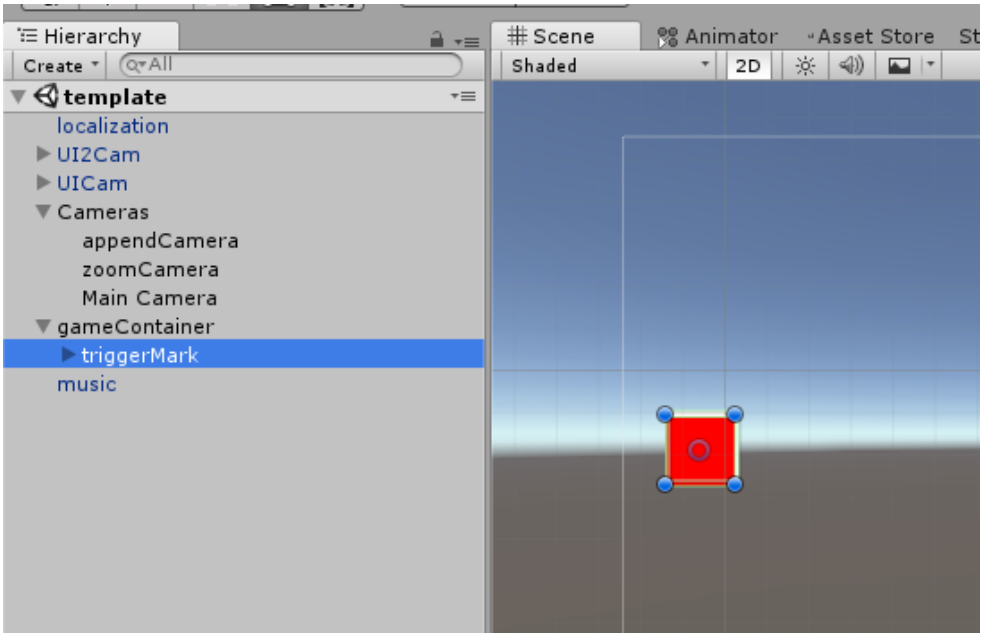
The template scene

As this is a very light weighted and none coupling engine. We don't like other products supply a lot menu and create shortcuts. We just supply template for create new scene. This would be extremely quick and easy to make your own level scene.

The template scene file is called the "template" just same place with other example scenes.

Under Assets/Hitcode

Let's open it for detail.



We see there are several blue gameobject in the hierarchy. It means they are prefabs and should always keep same to every scene.

If you modify one and apply the prefab. All other scene with the same prefab would be change automatically.

The localization: It is a localization manager for localize game text.

The UICam: It is the major UI manager for most of the game UI. Such as interactive panel, item panel. Please do not change their sort order sequence if you are not very familiar the engine.

The UI2Cam: When you activate a subscene(A scene which above your current scene view but not cover all.). The camera shows a black mask between the subscene and original scene to avoid misoperation. Please see details in "cameras" chapter.

The music: It is a sound manager for each scene to play sound or music.

We see under cameras there are 3 cameras. The main camera is necessary and another two are just ready made template cameras. The append camera always uses to add append scenes(subscene). And the zoomCamera always use to switch camera view in a single scene file.

We see a trigger mark inside the gameContainer. This thing uses for dynamic exit buttons. You can watch the introduction video to know what it is and see tutorial on chapter "dymatic exit mark".

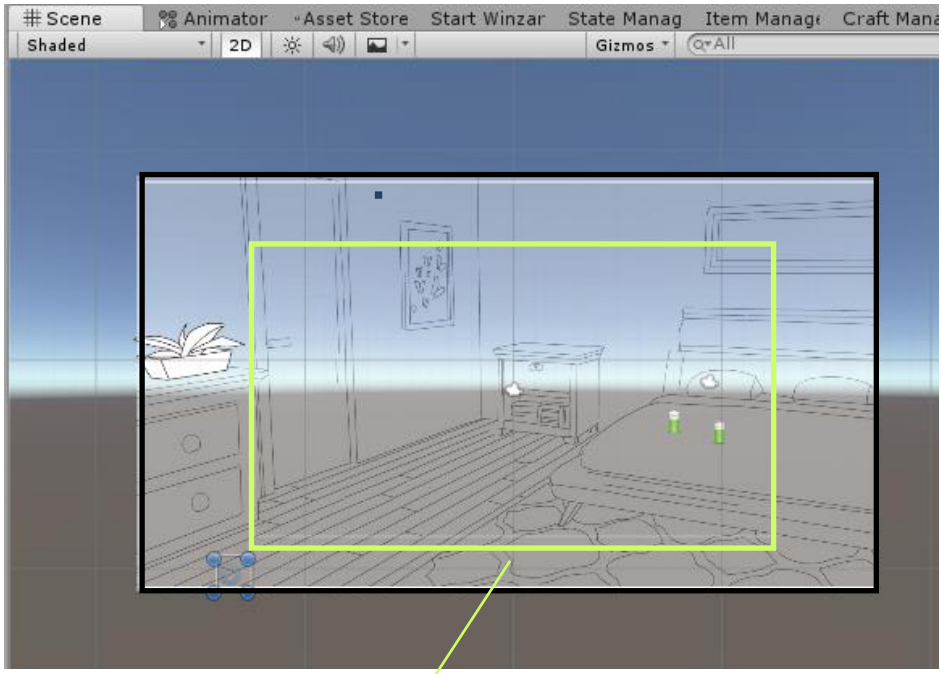
We always create a new scene just by drag and duplicate the template scene.
We can put our scene file pictures into gamecontainer folder so can be easy to understand and manager them. We would talk details later.
You can watch video lesson there for detail.

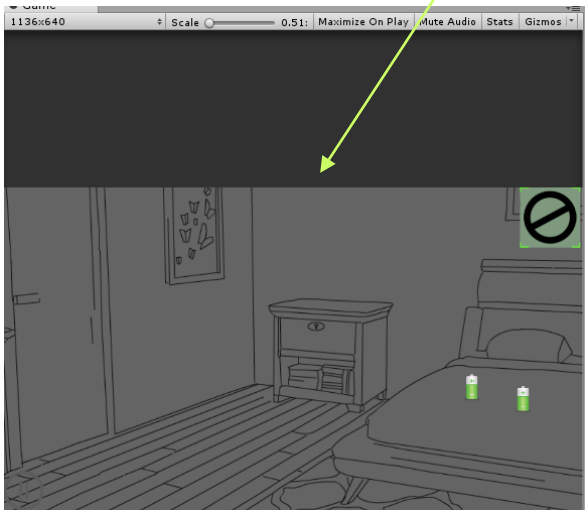
Tutorial: create a new game

Create a new room

After you duplicated a new template as a new scene file.

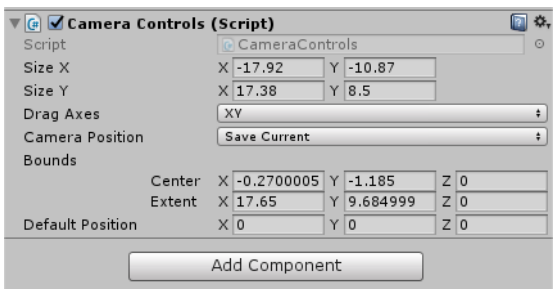
You can drag a map into the scene as a background. The map can be larger than the screen area, so you need add a script called “camera control” to make your scene dragable.





The green frame is your camera area. It refers to what you see in the game window. The black box is your dragable area set by camera control script. It is exactly the map size.

You must set the dragable area to be larger than the camera area and smaller than or equal to the map size.



See detail video tutorial here:

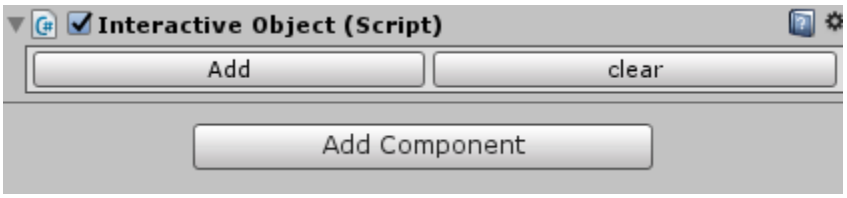
Tutorial: How to create a new room.

Interactive Object Component

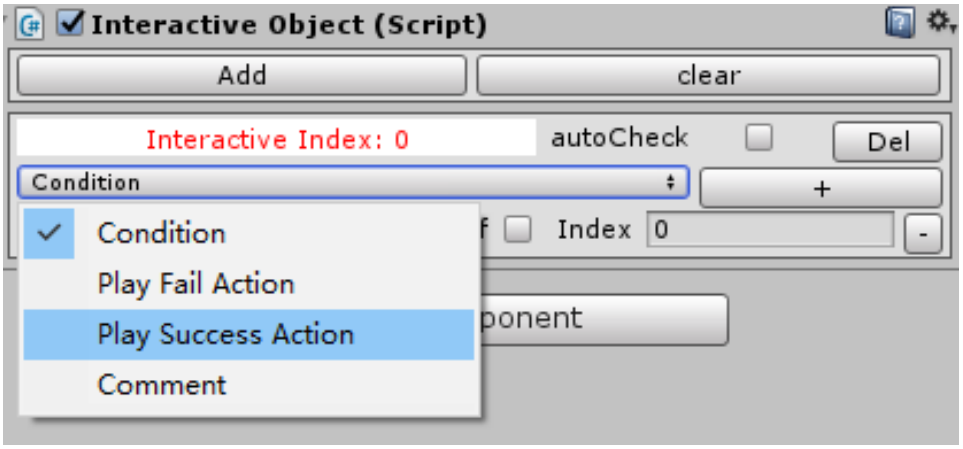
One of the most important script of this engine: the interactive object.
If you want to add interactive to object, which means if you want to click an object and want it do some action. You would require an interactive component.
An escape game actually is totally made with interactive objects.
To add an interactive component.

1. Select gameobject and see its inspector panel.
2. Press add component and input “interactive object”.
3. You would find the script and click “add component” confirm to add it.

A new added interactive object component is like this on inspector panel



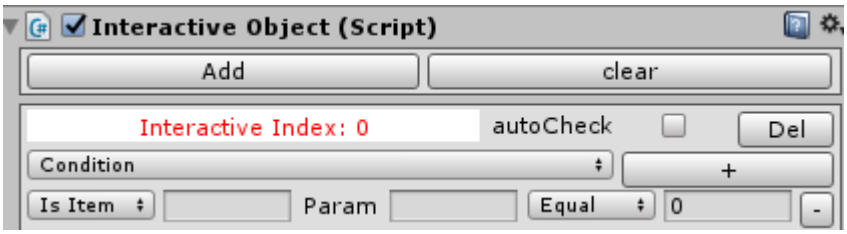
When we press the Add button. An interactive command would be created.
(There maybe something a bit different to video tutorial as we only upgrade the lastest to docs.
It is no need to tangle with the minor different between video and doc)



Each interactive you can add infinitive commands.
There are 4 kind of commands can be choose for a command.

Condition

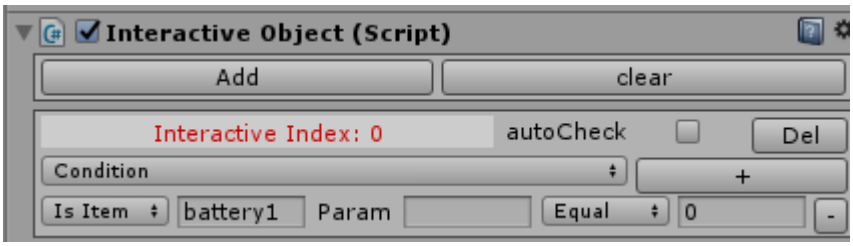
A condition is use for judge whether the interactive condition is fit or fail. Conditions can be infinitive. If one condition is not fit. It means the interactive condition is failed. But a fail condition does not mean not trigger the interactive. We would talk below.



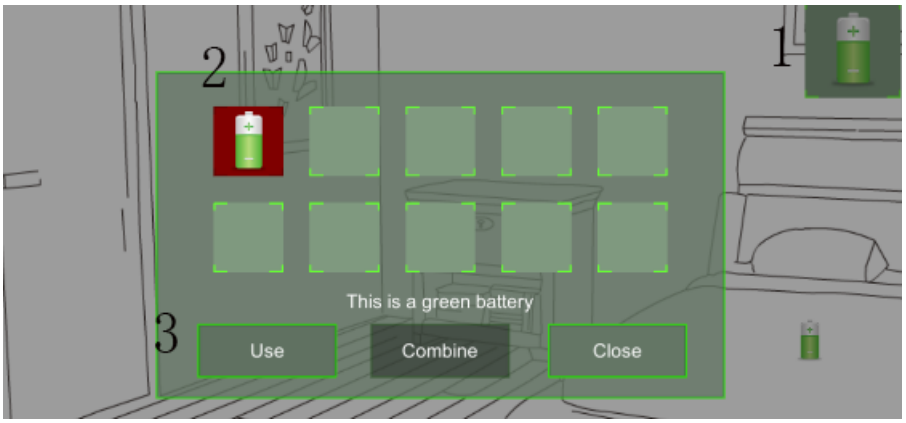
The upon picture is a normal condition command phase.
It contains 2 sections.

Is item or not item

This is the dropdown list which you can choose “is Item” or “not item”
When you choose “is item”. You can input into the field an item name.
This item must be exist in your item manager and exactly the name you set for it.
More about items, see chapter “item manager”.
If you do this. The condition means: when you interactive an object, you must have the item being used to fit the condition.
For example: you add battery1 in this field.



Then, If you want to fit the condition. You should pick up the item and using it. Like this:



And vise versa, if you choose “not item”, the condition would be fit only when you are holding an incorrect item. This situation usually being used for wrong operation tip in game.
If you do not require an object to start the interactive. Just leave this field blank.

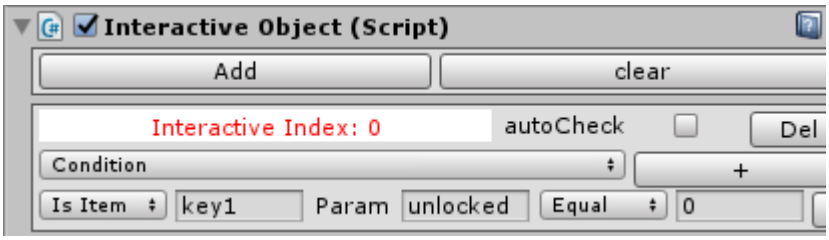
The parameter

Before you know parameter. You can watch video tutorial “Save the state”

Or read the chapter "state manager" first.

A param field is state variable. You can set the variable as a condition to be equal, less, great or not equal to a certain value.

For example, If you fill the fields like this.

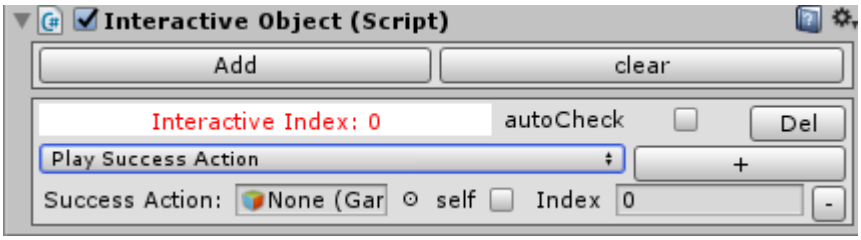


It means, if you want to fit the condition, you would both fit 2 conditions.

One is you must using an exist item named "key1", the other is that the "unlocked" variable in your system data should equal to zero.

Play Success Action

This is when you choose a "play success action" command in an interactive command.



A play success action means when you fit an condition, this command should be executed.

And if you do not add a condition(like this one),it would also play the success action.

We have talked upon about conditions. So what would this success action command do?

The success action field:

This is the field to assign action target. We would talk about actions later. Or you read the "actions" chapter.

The self checkbox:

As an action many times always combined with the interactive on the same object. It can be easily to assign the target by just check self. The target would be automatically assigned to the self gameobject.

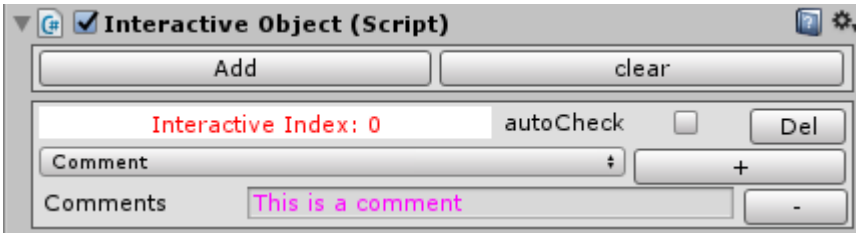
The Index:

As you see, an interactive object component can including more than one interactive command. Now we are focus on the interactive index 0. So the actions would be the same. It would has an index. So assign the index would relate one interactive to one action.

Play Fail Action

If a condition is not fit. We can play fail action. For example, you just give one condition and if your player interactives it. It can be success or fail. So you can deal with 2 different game situations.

The comment

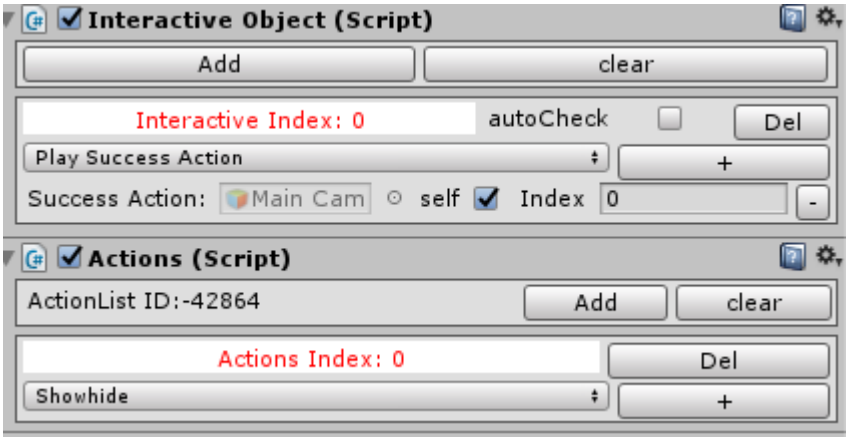


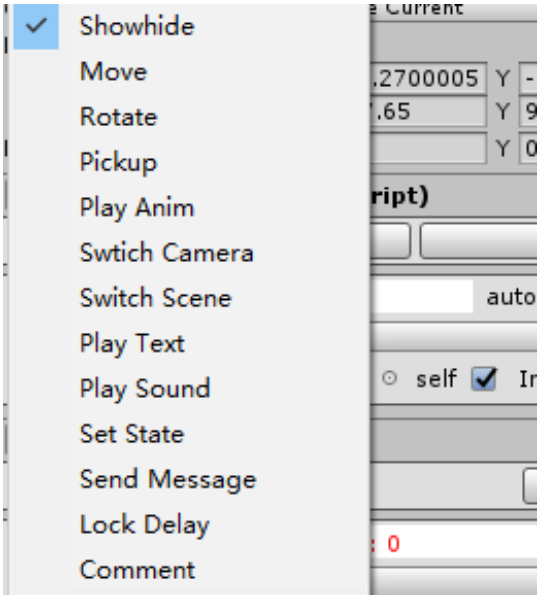
Comment uses for tip or mention yourself what you have done at this place.
It is not necessary but would be a good habit to always do that.

AutoCheck

Each index of an interactive phase would have an autocheck function.
When you leave it unchecked. The interactive would check the condition and doing the action by your touch or click to an object.
But if you check on the autocheck. This section of interactive would be ignored when you interactive with object by click or touch.
So when would this interactive trigger when we checked on the autoCheck?
When we check on the autoCheck. It means, the condition and action job would be played only when its attached gameobject being enabled or activated.
It is very useful for auto setup saved game states. For example, when you leave a room in the game with the door opened. And get back to the room. At this time you reenter the start room, all items and objects in this room is being activated, So then if they have an interactive with autoCheck, it would check the whether its condition fits or not fits. And do the play action command for the condition result.
The video tutorial here have detail example to teach you how to use the autocheck
Tutorial: Save the state

The actions component

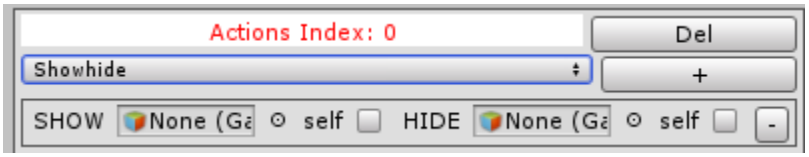




Here we add an action. Commonly, an interactive is always grouped with an action.
The above interactive here means: if you interactive with the object(click or touch on it), it would run the below action command which index is 0.
We have many choices for actions listed here. You can add many actions once as each action can include a lot of operation. You can show and hide objects, then rotate it ,then do other things.
Here we talk about details below of all these functions.

Showhide command

A showhide command means show or hide the gameobject. You can both show and hide 2 different gameobjects in a command.



When you add a showhide command. You can assign which object you want to show and which object you want to hide. If the field target is the script attached object, just check on self. The target would be fill automatically quickly.

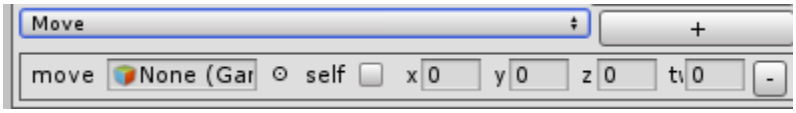
Show and object equals .setActive(true) by script. And hide equals .setActive(false) by script.

A showhide command always can be used in close/open operations like on door or closet.

Here is a detail video of how to use interactive and actions.

Tutorial: Interactive Basics

Move command



Move commands uses for position move.

The move filed requires a target for which object you need to move.

You can check on self to assign the self gameobject easily.

X,y,z refers to the relative move distance. The last field "tween" means whether you want the move to be an animation.

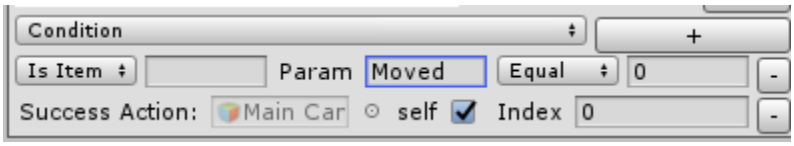
If you just want a state change just leave it to 0.But if you want a tween, assign it a float number. This would be the tween time for a move action.

One import thing! As after you interactive an object. The object would move a relative distance. So if you do nothing to tell the system that "you have already moved". It would keep move again everytime you click on the object.

The best way for forbidden this is add param limit to the condition.

Like this, once you moved, you add a param "Moved" and set it to 1.

So your interactive to trigger the move command should be like

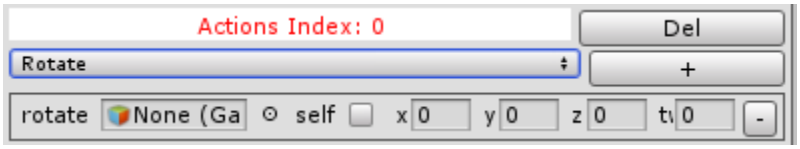


Because of you check the moved state, The move command would only run once.

This is a regular way to forbid unnecessary or too fast operations. You must understand it use it in many places of your game. For detail of state and interactive. See video tutorial below.

Tutorial: save the state.

Rotate command



A rotate command is very much similar as a move command.

The rotate field requires a target which you want to execute the rotation.

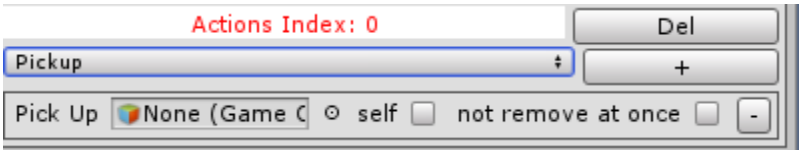
Self check apply self gameobject as target.

X,y,z refers to the value on three axis of rotation.

In a 2d game, normally we only set z for a rotation.

Tween is a value set to tween the rotation. If you do not require a tween. Leave it to 0.

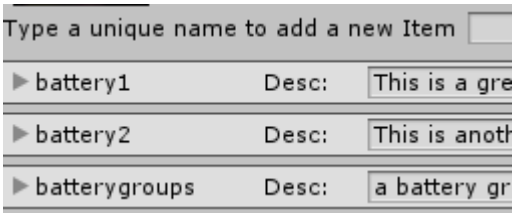
Pick up command



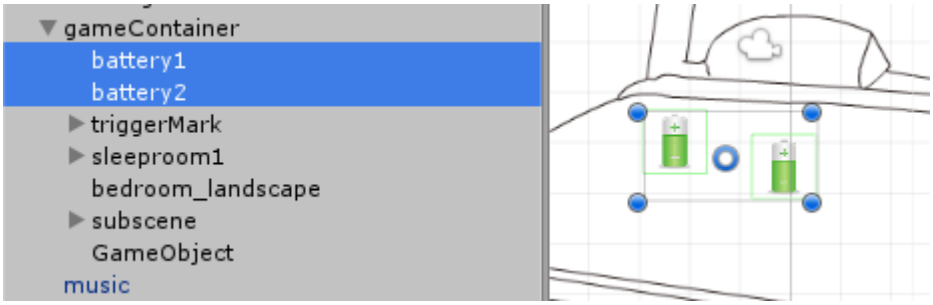
Before you pick up an item, you require set the item in the item database on item manager panel.

Watch the video tutorial of pick up item or the chapter item manger first.

For example:



This is in the item manager. You have add 3 items.

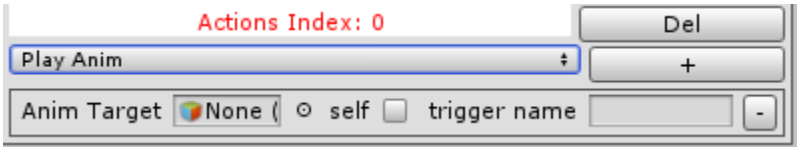


And this is the item which you want to pick up on the scene.

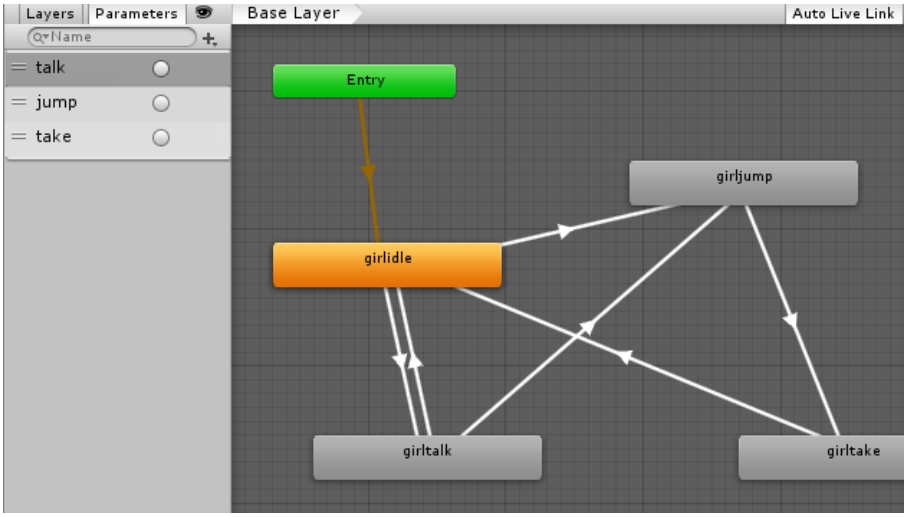
So, the item name requires exactly the same as what you set in the item manger

Let's back to talk about the pick up command action.
In the action, the pick up field requires a target to be picked.
Check on self to be easily assign self to be the target.
The not remove at once means ,if you check on it. The item won't disappear after your pick up. Like the items is infinitive. This can be used on the piles, stacks etc.
But remember. You always need parameters to save pick up state. Otherwise, when you restart the game, the picked up item would still existed on the scene.
You can refer to the tutorial: save the state.

Play Anim command

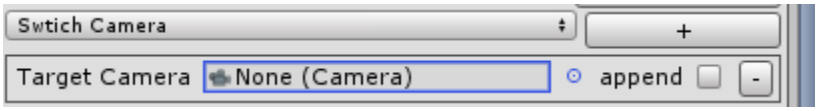


An play anim command uses for play animation on interactive objects.
The objects must contain a animator and a trigger parameter .
Such as



If you don't know what it is. You must learn unity tutorial about 2d animator animation first.
Tutorial: Unity 2d Animation
Once the interactive object has a animation which can be called by trigger. This command would work.
The anim Target is the field to assign which gameobject you want to play the anim. The gameobject must have the animator attached on its root.
Check on the self can quickly assign self gameobject to be the target.
The trigger name is the trigger parameter you set to trigger a animation in the animator controller.
As the upon image, the trigger parameter can be "talk","jump","take". They refer to 3 instant animation actions.

Switch Camera command

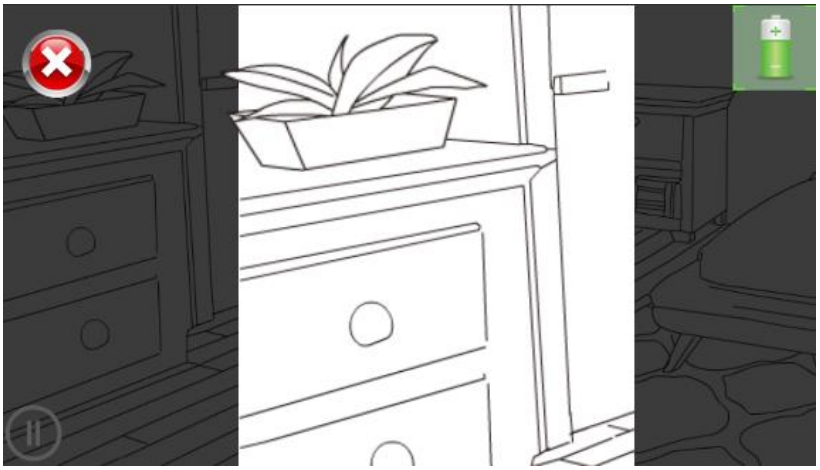


The command only have 2 filed.

One to assign a camera and another to set the camera mode to be whether is append.

If you check on the append checkbox

This command would be used for appending a sub scene. Like this:

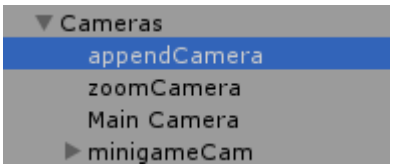


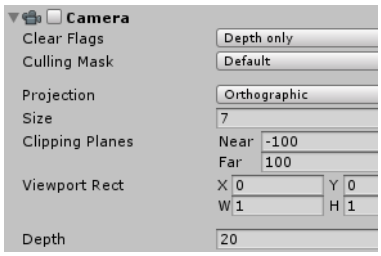
The sub scene mode(checked on the append checkbox)

As you see, this mode add a new scene on the origin scene.

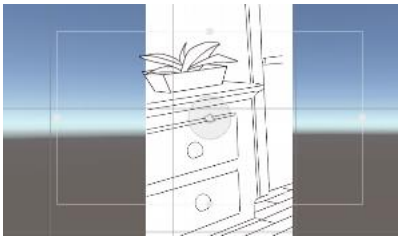
It did not cover the origin scene. But separated them by a black mask.

This is always useful for detail gameplay when you check on some place but do not want to interactive on the base scene below.



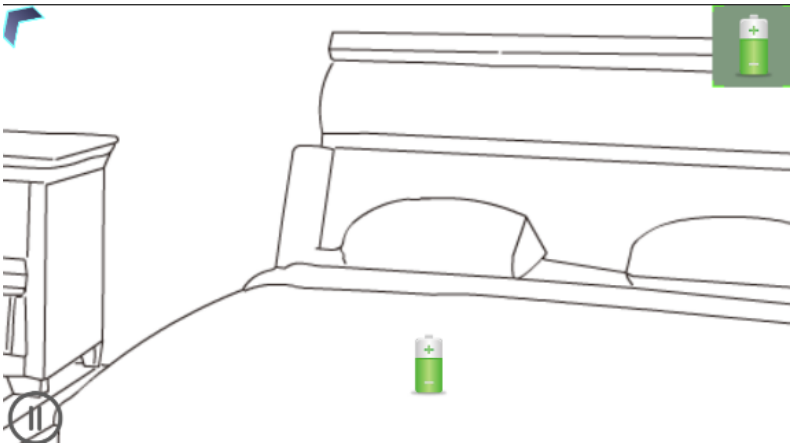


If you want to make an append scene. You require an appendCamera first, this camera has a very high depth so it can be keep on the top of the scene. Spot the camera to the additional sub scene in a empty place as I showed below. The camera default is not enabled because we use command to control is visibility.



If you do not check on the append checkbox

Should be like this when execute the command



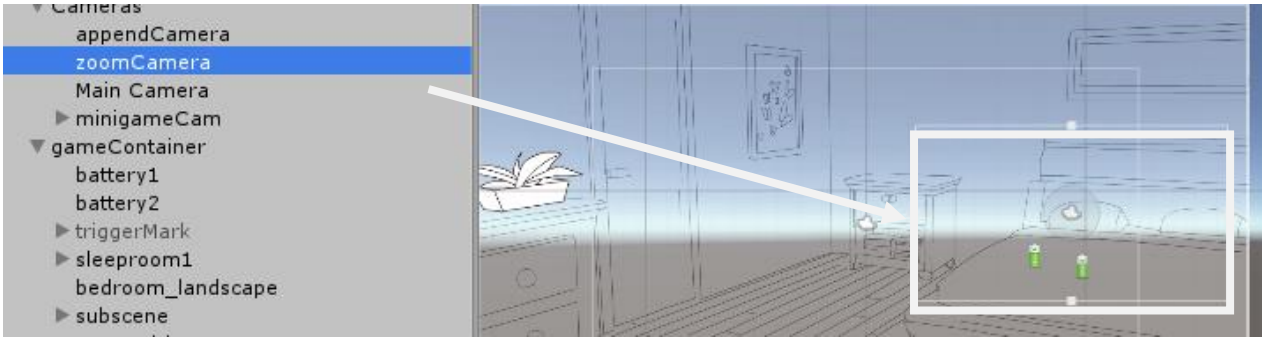
(The mode you do not check on the append checkbox.)

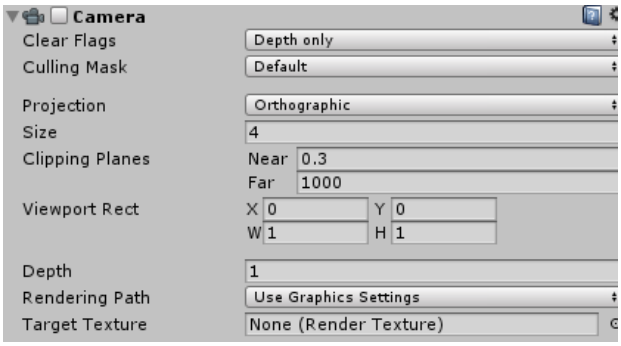
You see it is different from the append mode. There is a return button on the top left.

This always uses on quick switch within a room but not switch between scene files.

If you continue using an switch camera command in this new camera. The camera would be added into sequence. And you click the return button once, you exit one step to the previous camera, until you exit to the origin base scene camera.

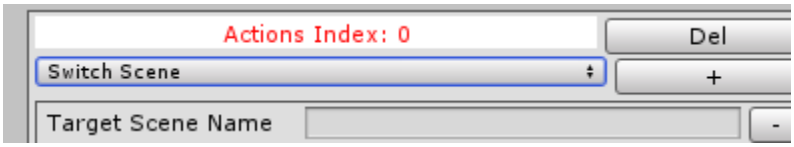
To make cameras like this. Duplicate the zoomCamera from the template.





Also, it is disabled and wait your command to enable it.
Remember, as the camera depth would effect the UI sequence. You'd better always duplicate what type of camera you want when using switch camera command.
For detail, watch the video tutorial: Switch Cameras

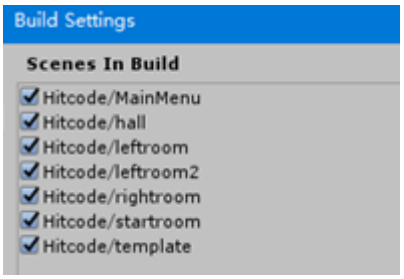
Switch Scenes command



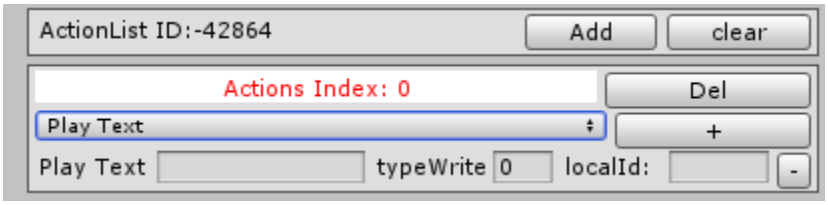
This is a very easy interactive action command which uses to fade in/out and change to a new scene file.

The only field "Target Scene Name" is the scene file name which you want switch for when interactive.

But be sure in scene file is listed in the unity build settings.



Play Text command



The play text field here, you can input your text into it.

The typewrite is the typewriter effect for when show a dialog. It has a sound effect also.

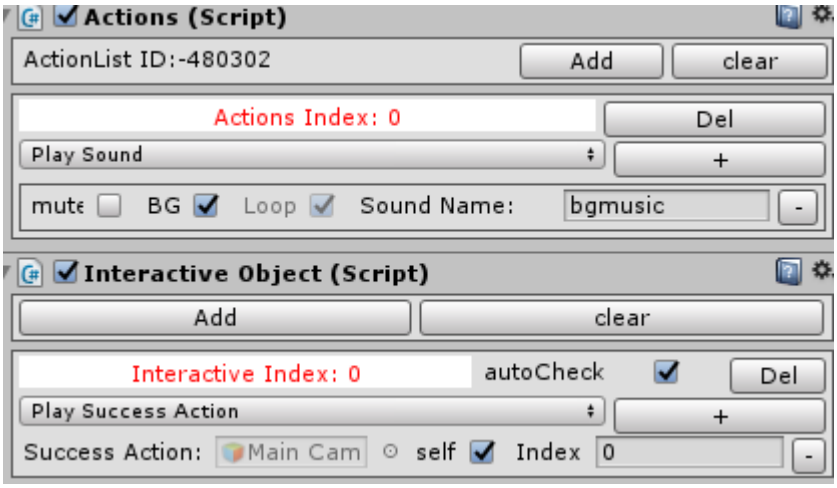
The localId is a localization id which you can set it for this text phase in your localization files.

See localization chapter for detail.

When text is being played. You will not be able to click menu or interactive other things. This is for game logic safety.

Play Sound command

In each scene file of example project, there is an actions/interactive group attached on the main camera like below.



We see the autocheck of this interactive is on.

So, it means no interactive is required by the system, it would check automatically on the start of loading the room scene. See detail about autocheck

It refer to the action 0. So we focus on Action index 0 and see there is only one command.

This is a command of playing music or sound effect.

Let's talk about this function below.

The mute: When you check on the mute, and if you assign a sound Name, You would stop a music or sound effect by this command.

BG checkbox: When you check the BG checkbox on, the loop checkbox would be check on automatically. Then this command would be ask to play a background music. You enter the background music name into soundName field.

The sound or music files must all under Assets/Hitcode/Resources/sound folder.

One thing must mention. The system only detect files with "bg" start to be a background music file. So, if you want the music file to be a background music. Name it to "bg"+xxxxx format.

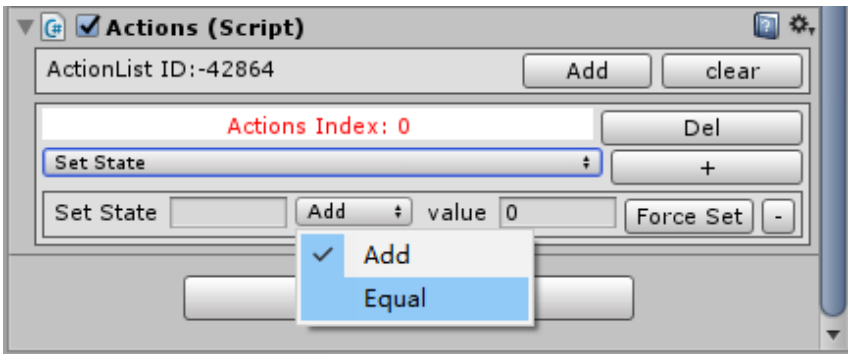
Such as bgMusic,bgSound,BgMusic2 etc. If you are not start a name with "bg", the system would not think of it was a background sound and would cause problem.

Why we do this because it is for you easy to change background sounds just by assign a new background music name but no need to stop it first.

Loop checkbox: when you check on this box, you define the game to be a loop sound/music. A loop sound won't stop until you mute it.

If you just want to play a sound effect. Input a sound name which existed as a file in the sound folder. It would play once you interactive the gameobject.

Set State command

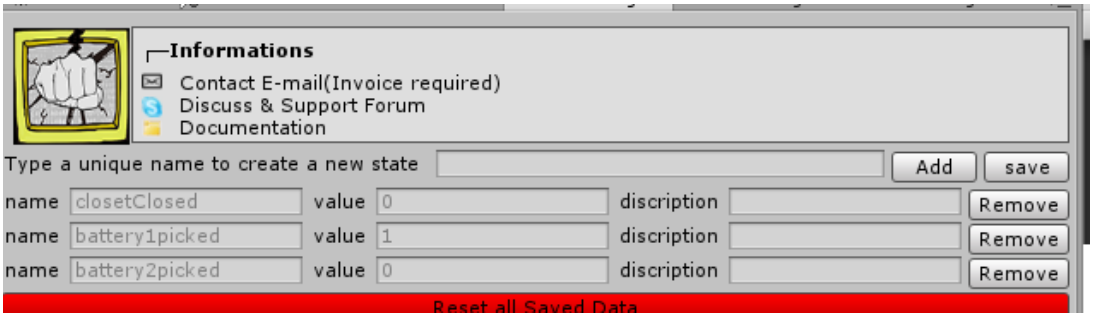


State actually is playerprefab integers. Their default value is 0.

Here you input the parameter name into "set state" field

And set add or equal to this value.

The force set is only use for test. When you click this button. The value would be exactly the value you want.



The state manager

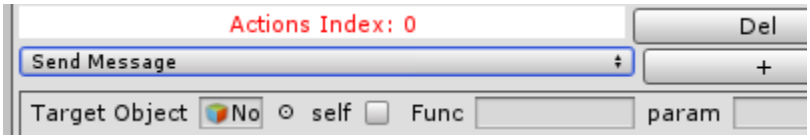
The state parameters are all listed in the state manager.

You must copy the name from state manager to the "set state" field. If you make things wrong. You may not get the effect you want.

See how set state works, watch the video tutorial: Set the State

Or the Chapter State Manager.

Send message command



This command runs the sendmessage() function in game.

A send message command requires 3 main fields.

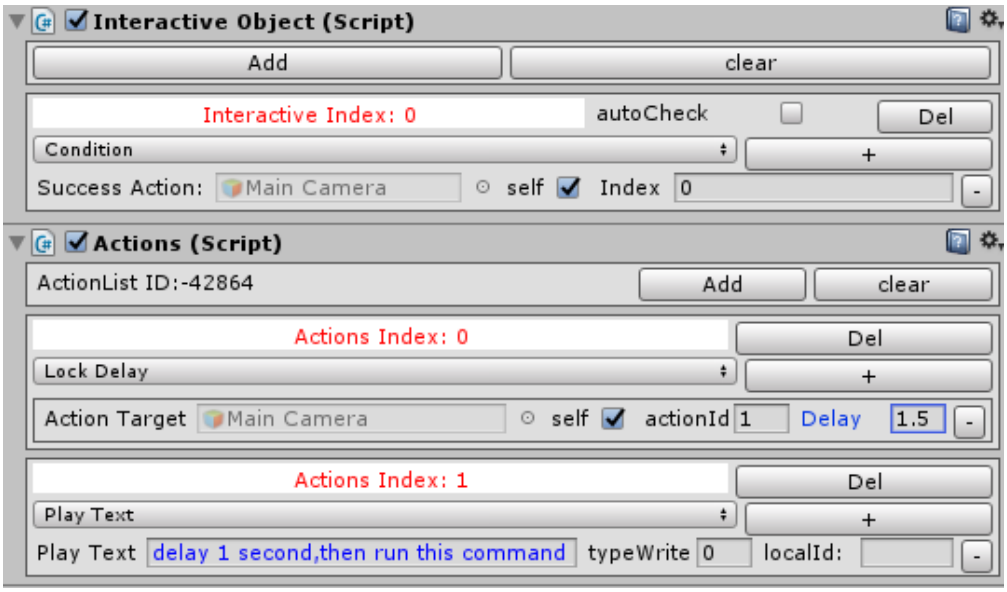
The target Object: It requires a field to fire the message. Check on self if want to assign the self gameobject.

Func: function of which you want to call the target. It is no need to be public.

Param: the parameter, if you want to send a variable with function as parameter, use this.

See detail of how to use send message. Watch video tutorial: minigames

The lock Delay command



The lock Delay is a very very useful function for sequence actions.

This picture upon is an easiest lock delay action example.

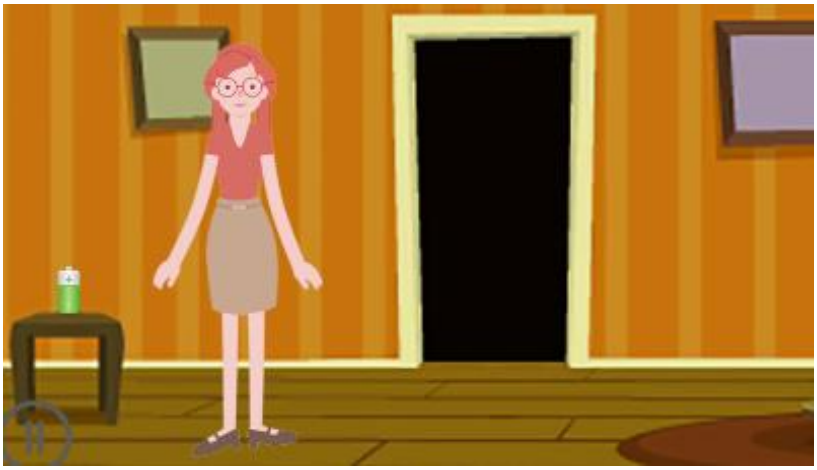
First, when you interactive an object, you target action zero. So you run action 0 at once.

Then after 1.5 seconds. The system would call action 1 itself. And export a text.

Why we need such function, because not all the action are played simultaneously but could be have a short time gap. During this gap. If you doing other things, You may cause vital harm to the game logical.

Lock Delay example:

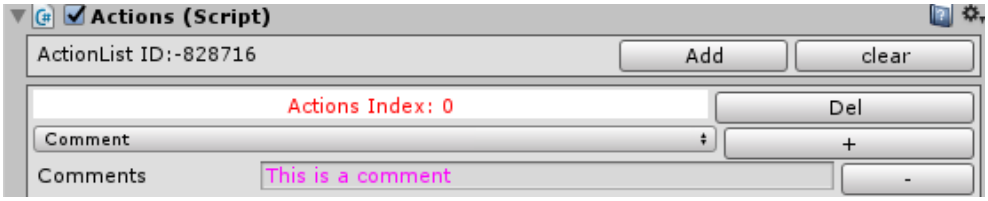
Here, first state: The girl stand far away from the left where can pick up a battery.



Step 2: You touched the battery. Many times, players won't obey the rule of the game, they would touch 100 times repeatedly on a object. So logical safety lock is necessary. Here the girl would jump for 3 seconds before she reach the battery. During this time, player won't be able to do any other operation before the animation finishes.

Step 3:Now the girl would pick up the item. After finishing this action, the game interaction would be unlocked. You can do other jobs then.

Comment command



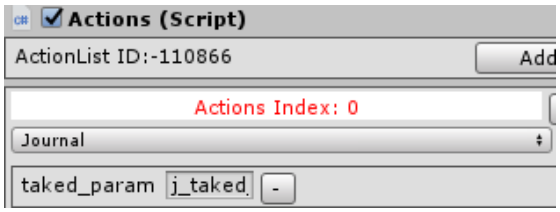
As the interactive component, each action can add comments to help remember what you do the job.

Journal command(V1.2)

First,you should create the journal data in journal manager.

View chapter journal manager for detail.

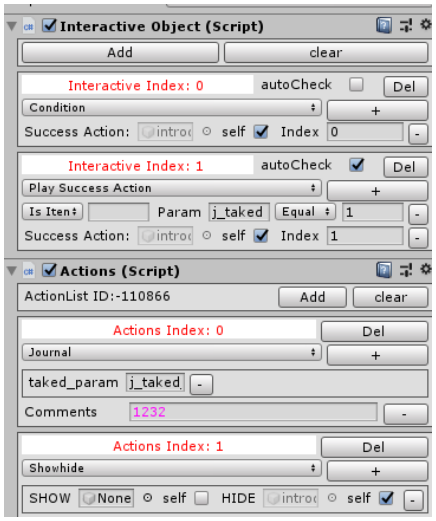
When your journal data is ready. You can add the journal action command then.



the taked_param refer to a parameter which created in the state manager.

Learn chapter state manager. It is used to record the state of a journal which whether been taken. If you leave it blank. This journal would not be pickup.

There requires a combo of actions and interactives for relieaze the being taken effect. S ee the example project secene for details.



For more information of why doing like this. Watch video

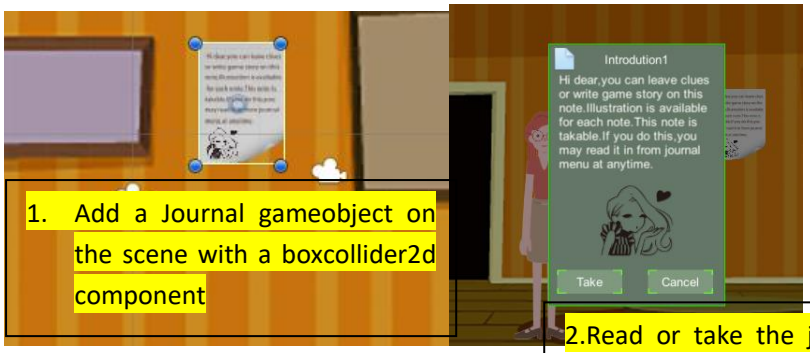
Interactive basics

Save the State

The journal book was included in the pause menu.

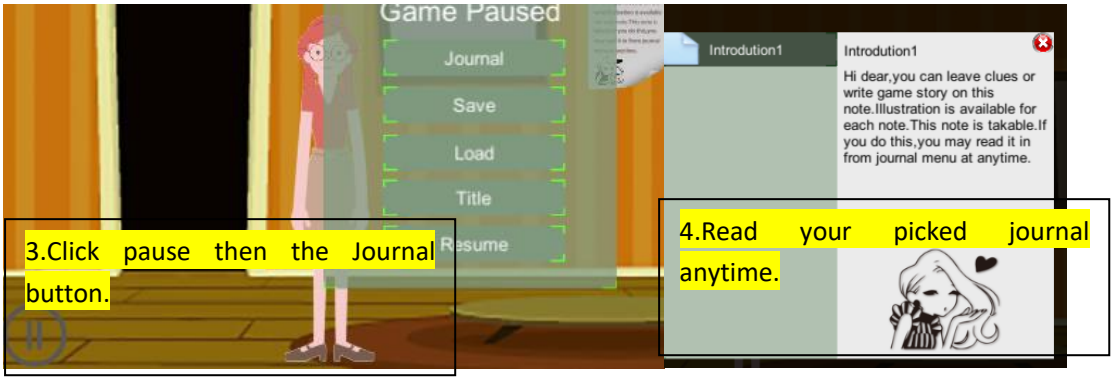
Click the pause button at bottom left of the map to find it.

Here is an example of the journal flow.



1. Add a Journal gameobject on the scene with a boxcollider2d component

2.Read or take the journal if it is takable during the game.(Depend on its takable parameter)



Trigger Marks(dymatic exit marks)

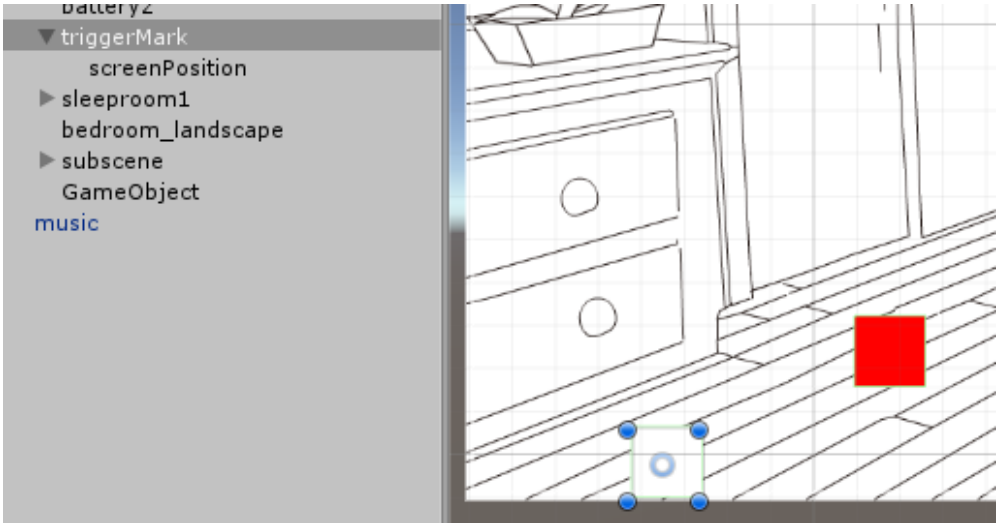
Many times the exit is not a door ,or a very clearly place. So you need to give players some tips when they drag the map to a place when there is exit area available.

For example, like upon, you just see the middle part of the whole map. So no exit mark is available for changing a scene.

But now you drag the map to its left, an arrow appeared to tell the player, there is an exit/entrance of another room, click on it and would fade into another scene file.

You can see triggerMark gameobject under gamecontainer in your template. Copy as many as you like to be used as dymatic exit marks.

Here is an example of how to add dymatic exit marks.

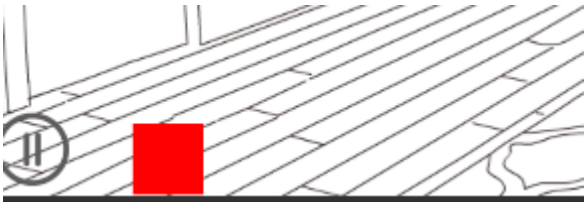


Here, each triggerMark has 2 objects.

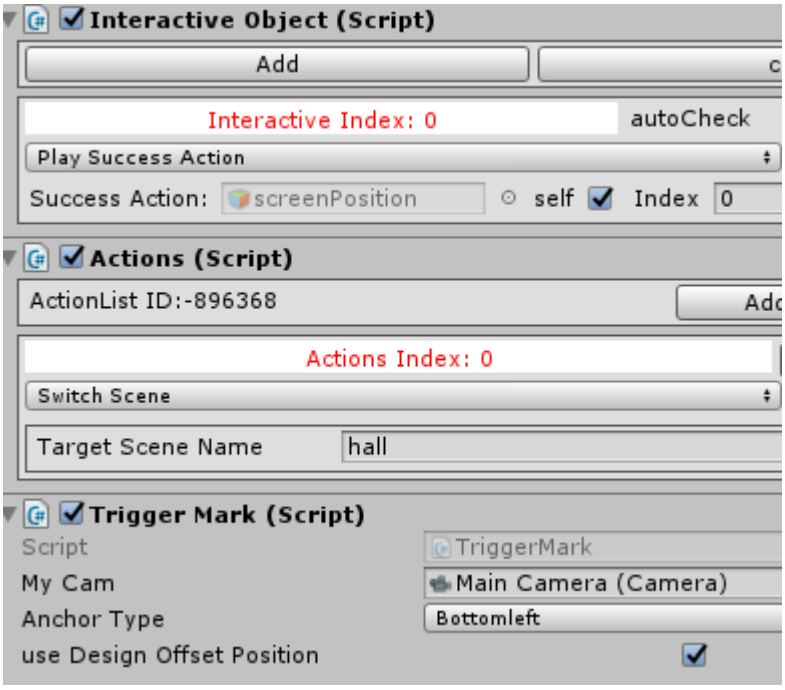
The parent object names triggerMark is a collision area which would never seen in gameplay. Where you put it? As the picture above, you put it on the way exit place of the whole map.

The child object of a triggerMark is the visible exit mark. You can change its sprite to be an arrow or other easy understand buttons.

As above situation, the red square currently in game window is at this place.



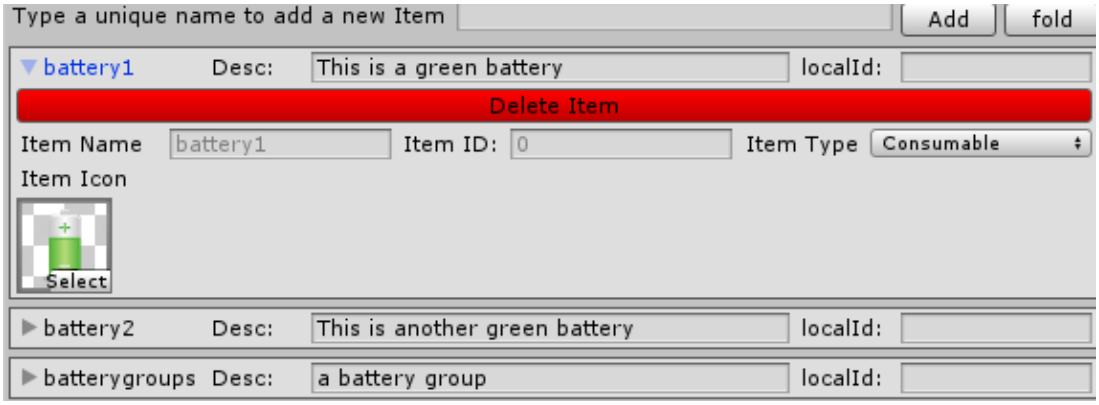
When you run the game, what you need is that the red square(you would change it to other image) should keep on this place.



So, first, you move the red square to the place where you want previewed in game window.
But you must remember, the place by this step you put the way mark, its relative position is base on your current design window aspect radio.(here is 1136x640). If you change the editor resolution, you should redo the work to put the way point to a current position in preview game window or you just change back to which resolution you use before.
But this does not mean it only support one resolution. No worry about that, it is because the system need get a relative value on screen board before compile. So you make sure you put the way mark right place when you compile(export) would be fine. The game would fit any resolution of screen and device.
So, to keep relative position on screen. We need set its relative anchor, just like unity UI.
On triggerMark's inspector, there is script named trigger Mark.
Set my Cam is the camera where your exit image belongs to.
The anchor type has 8 anchors on the screen which you can use to set relative positions.
But remember, use relative position to screen by your preview design. You should check on the Use Design Offset Position checkbox. Otherwise it would snap on anchors only.
To learn Detail of how to use trigger Mark, watch video tutorial: Dymatic exit mark

The item manager

Pick up item manager window from Room Escape ->item Manager



Item manager is the database for store item information.
Input unique name and press add button.
You set the item icon for the item. And you need not to worry about the size and aspect radio. The icon would show its best size and aspect radio automatically in game.
Desc: In item panel, when select one item, its description would be shown below.
localId: if you fill in a legal name, the description string would be replaced by the localized string in your localization files. See detail about localization.
Item ID: system required, no need to pay attention.
Item Type: You can choose consumable or un consumable. This only effect items which would be use in an interactive. The item uses only for combination would have no influence. When you

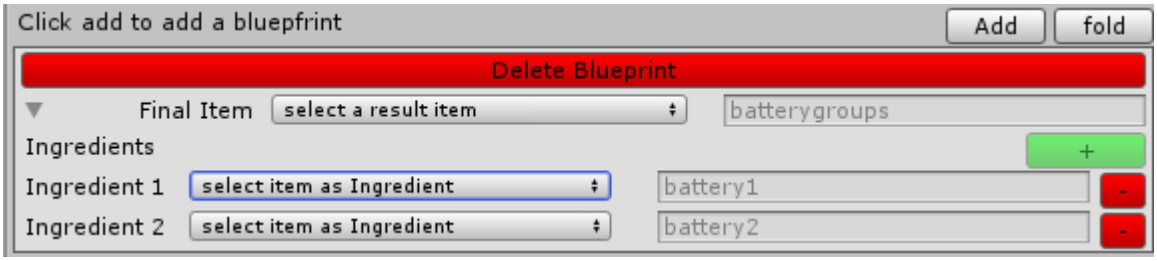
choose consumable and interactive an object. The item would be removed automatically if the interactive condition is successful.

And one more thing, some item manager depends on the item index when you use it for other places or managers(such as craft manager).This would cause mess if you delete one no more used item. But our engine do not have such problem. You can take it easy to add or remove any items which you not require anymore and no need to fix its side effect on other setting managers.

Learn detail about item manager, watch video: pick up items

The Craft manager

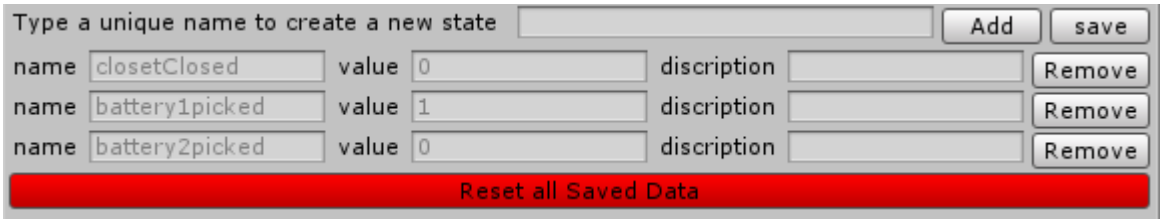
Pick up craft manager window from Room Escape ->craft Manager



Craft manager can not create items but only can choose items.
So the final item is also required to be created in the item manager first.
You press add to add one piece of craft data. Then you just pick 1 item to be the final item and 2 different items to be its ingredients. It is extremely easy just buy select from drop down list.
See more about craft manager. Watch video tutorial: Craft Item.

The State manager

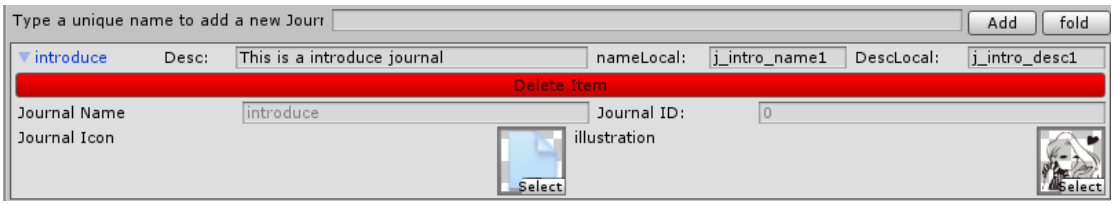
Pick up state manager window from Room Escape ->State Manager



State actually is playerprefab integers. Their default values are 0.
As the name suggests, state values uses for save different game state. Such as the door is opened or closed. Judge the state in condition param before run an action is the most common way in interactive commands.
See detail in chapter: Interactive object component
Or watch video tutorial: interactive basics.

The Journal manager(v1.2)

This is a new function for 1.2. You can add note/journal in the game to supply player c lues or used for tell your story.
To create a journal, open Journal manager by menu RoomEscape-journal manager.



a unique name is required as the joural name.

The desc param is the content/discription of the Journal. This parameter would be decre pated automatically if you have filled in the parameter in both nameLocal and declocal. nameLocal and declocal refers to the localization string of each journal's name/title and context/discription.

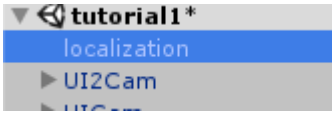
The journal icon is used for the icon. The illustration is used for insert a picture at the bottom of the journal.Use can leave them both blank.

About how to add a journal into your scene. See the chapter Journal command

Localization system

Set up localization string

In each scene, you would see a gameobject named localization. This is the localization manager which uses to control game language for different regions.



And here, it's his inspector

We see there are 2 files in the languages array here. Which means we only use 2 languages for the game. Each element here within the languages array refers to a file.

They are under Assets/Hitcode/src/Localizations

Each file was .txt file.

```
//system
btnStart = New Game
btnYes = YES
btnNo = NO
btnContinue = Continue
newGameTip = Start a new
```

English.txt

```
//system
btnStart = 新的游戏
btnYes = 好的
btnNo = 不要
btnContinue = 继续游戏
newGameTip = 开始新游戏么?\
```

Chinese.txt

As you see, each localization file include same localization string on left each line.

For example btnStart = xxx

The btnStart is a localization string.

We can set up localization string for any text in the game.

Detail as follows:

There are 3 places can set up a localization string.

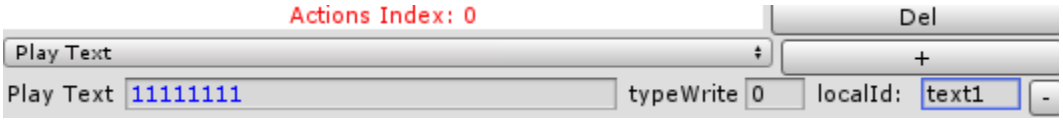
The origin system texts: UI texts are made within the system. And already reflect its localization string to the files.

The item description: It is set up in the item manager for each item

Here, you set up the localization string Id for battery1 as battery1.

If you not leave the localId blank, the default desc text would be overwritten.

The text action: It is set up when you add a play text action.



You set play Text field for a text string. But if you not leave localId blank.

The system would ignore texts inside play Text field, but search localization strings in files with this localId string.

When you set up a localId(localization id) for a text. You must add it manually into all your localization files. As the text command on inspector were all blue color, so it won't be hard to find them if not add the id as soon as you create the command.

In the game engine, to call a localization string, use the script as below:

Localization.Instance.GetString("xxxx");

the "xxxx" is the localization string we talk about above.

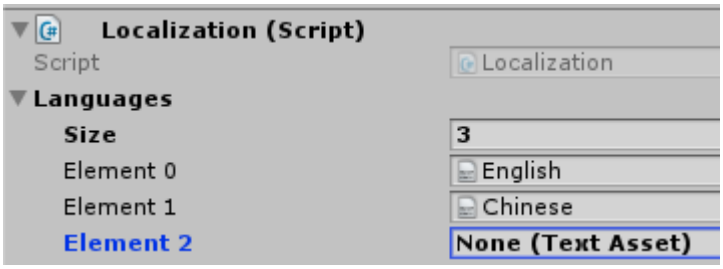
Add more languages

Drag and duplicate a new localization file first from

Assets/Hitcode/src/Localizations

If you not drag the text but create your self. Make sure 2 things

1. You must have all localization string exactly the same as default English.txt
2. The txt file must be UTF-8 format, you can set up the format by use save as with default windows or mac text editor.



To add a localization file. See any localization script in the scene or on localization prefab.

For example, here, you set the size from 2->3. Then you drag your new created language file into element 2. After doing this, you should click apply on inspector to apply your setting to the localization prefab so that all same gameobject in each scene would update the same time.

After doing this, you require to add a localization icon for player to choose on start menu.

Active the scene file Main Menu

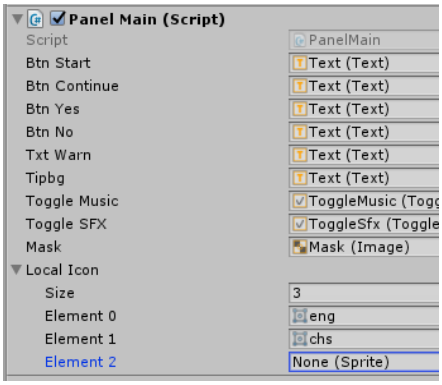
under Assets->Hitcode

Find Canvas gameobject in the hierarchy.

See its inspector， should be like the right.

Here, see Local Icon, originally have 2 element.

You set it to 3 and drag a fit icon for your new added Language into the element 2 field



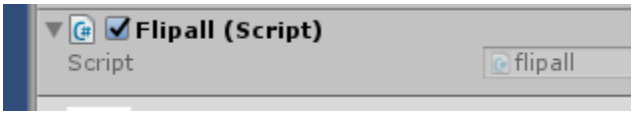
Remember, the language icon is related to the sequence you set language txt files in localization prefab. So 0 is English,1 is Chinese,3 is your new added language, and so on.
If you do not want to use main menu but want to set the language by your own.
You set the localization file first and apply the prefab. Then call below script to tell system which you language you want.
Localization.Instance.SetLanguage(index);
The index is your language file index in your localization prefab refers to its language txts.
But remember, set language won't update the already initialed text on the scene.
So you should reload the scene or update all localization text yourself.
So, only change language on the main menu is an easy way. Many game work by this way.

*If you wish to publish to webGL and use un-English language. You should assign the language font to each of your language and set all the text font in UI to this font. Web browser such as chrome not support default unity fonts. You must include language fonts in the project folder.

The Minigames

Although we supply some minigames, but still it is recommended that you make your own in-game games for personalized game play.
Here we take an example, and you may set up the interface for your own mini game by learning this chapter.
Find our example minigames prefabs under:
Assets/Hitcode/minigames/prefab

For example, let talk about one of them: the flipall
Select this prefab and focus on its inspector:

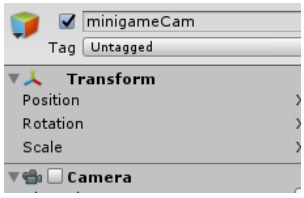


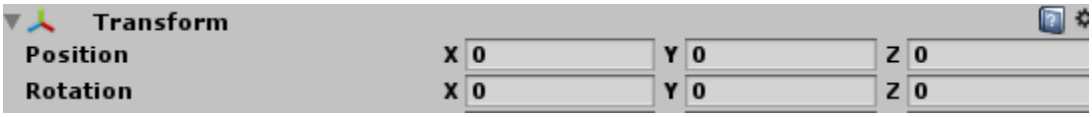
Open this Flipall.cs

```
public void init(){
    GameData.Instance.nActiveI
    Transform tcontainer = tra
```

We see there is an init function there.
So we use sendmessage command to start each minigame.
See detail in sendmessage chapter.

Add minigame into your scene





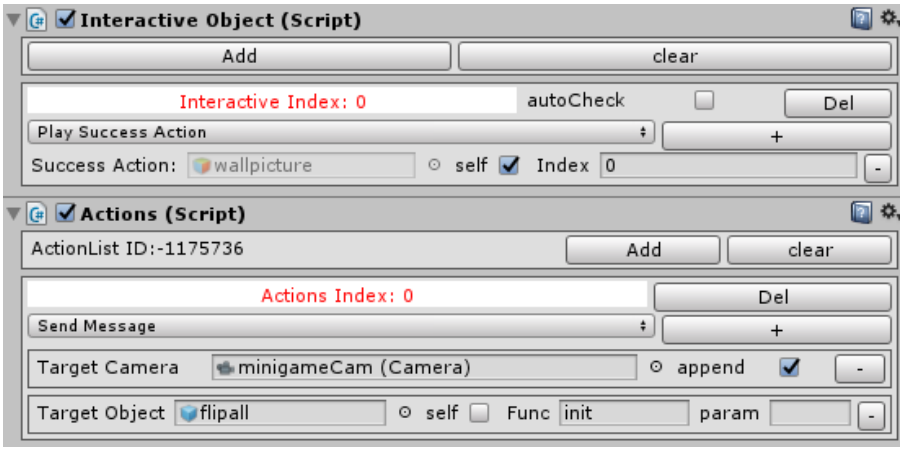
This minigameCam is the camera we duplicated by the appendCamera from the template.
Because minigame requires to be appended on the scene.
We just dragged the upon flipall prefab into this minigameCam's child and centered it.

The camera is not enabled by default. So the minigameCam would show this minigame on the top of the scene when you enable it.
Learn more about camera, read chapter: Switch Camera Command
Or watch video tutorial: Switch Cameras

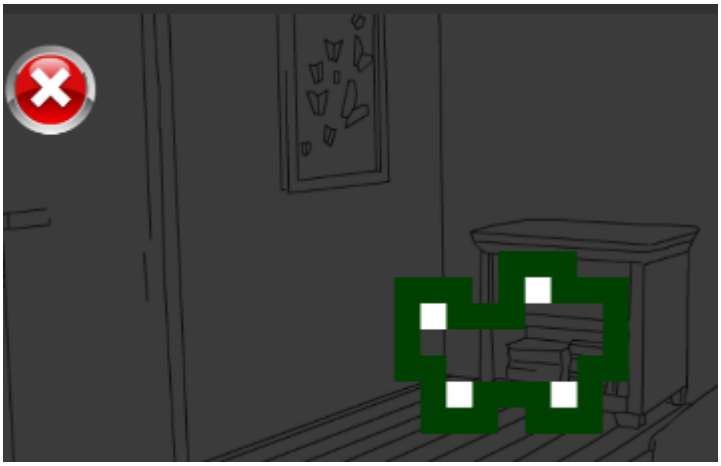
Start the minigame

When you put the minigame into the scene. It won't show unless you do some interactive on the scene to active it.
In our tutorial scene, we touch the photo on the wall to active the scene

Here, we attach 2d box collider, interactive component and actions component for this touching area. Let's see its inspector.



We see, we call self's action 0 by this interactive.
There are 2 commands inside action 0.
The switch camera command : We use append to enable the minigame camera and keep it on top. Learn more about camera, read chapter: Switch Camera Command
Or watch video tutorial: Switch Cameras
The sendMessage command : As we talked at this chapter's start, the minigame containers an init() on the root. So sendmessage would work. We used sendmessage to call the init function. So the minigame would finally show.
More about interactives, read chapter: interactive component.
Or watch video tutorial: interactive basics

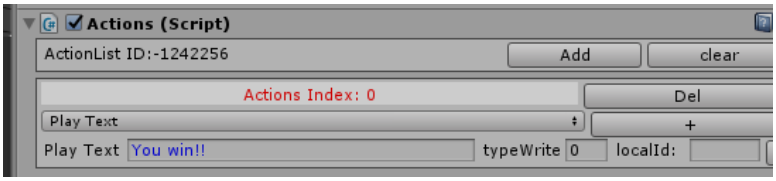


Like this, as a common appended scene. The minigame starts with a close button on its top left.

Win the minigame

Each time you won a mingame. We must do some action to push the progress of the main game. The example minigames were all already ready to call actions.

Here we just select the flipall on scene. And add an actions component on its inspector.



As an example, we just use the play text action for it.

So when you win the game. This command would be executed automatically to show a text on the bottom of screen. You can add action commands as many as you wish inside this action. Such as set state, open door, play sound, etc.

So you may want to know how this worked. This just because when you won the minigame, the game script find this action and called it by script.

See TouchBlock.cs under Assets/Hitcode/minigames/scripts/flipall

First, we found this actions command first on start by the script

actions = transform.parent.parent.GetComponents<Actions>();

Why we call these 2 parents, because the calling script is attaching on the actions gameobject's child's child.

When you make your own game, make sure you find your own action path correctly.

So, in this .cs script. We see a checkWin function. Here we use the script to call the action

```
foreach (Actions taction in actions)
{
    for (int i = 0; i < taction.actionSteps.Count; i++)
     {
          taction.playActionNow(i);
      }

  }
```

So,when you want to call the actions in your own minigames. Do it by this way.
See details about minigames. Watch video tutorial : minigames

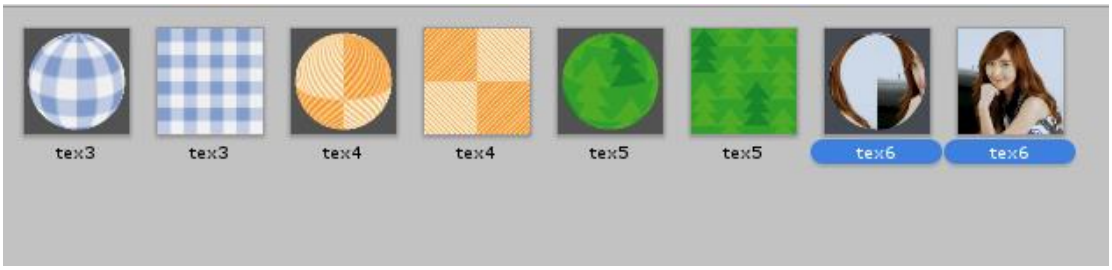
Detail of Mini-game specific settings

The polygon puzzle

The most important feature of the game was that you can easily add infinitive pictures for random puzzle. If you want the game not just a simple puzzle game, strongly recommend you to read this part.

Ready enough materials

Find under
Hitcode\minigames\Resources\tangram\Resources\Tangram\Materials
You could see many textures with a material ball for each inside this folder.



As you see, select a pair of a texture(an image file and its material ball) hold and drag to duplicate a new texture and a new material.
The system would give the new texture a sequenced new name ,in the upon example image, tex7 and material tex7 would be created if you doing so. Remember do not change the texture or material's name as the system requires a unique name format of "tex" +"number"

Setup in the scripts

Find the script under
\Hitcode\minigames\scripts\tangram\polypuzzle
Find Tangram.cs
Locate the script:

```
int tTextureNo = Random.Range(0, 3);//got 3 textures in materials folder total
```

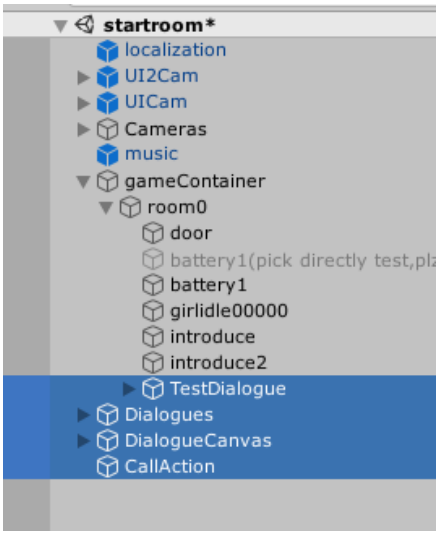
Here set the number 3 to the total pictures you just duplicated included in the materials folder.
If you got 5 pairs of pictures, set the number to 5 and so on.
So, in the game, system would find a random picture from your texture folder.
The other important thing is the UV size. If you don't want the finally completed picture the same of your origin texture image. You need to do the following:
Locate: `public int uvZoom = 1;` And set it to `public int uvZoom = 3 //or larger ;`
Then the game would auto color each block and make it as a loop mapped texture.

The Dialogue System(beta)

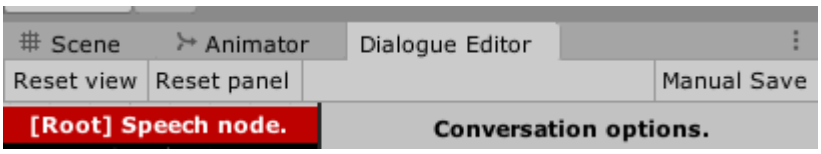
SomeTimes we not only need a npc to talk,but interactive conversation with you.
This dialogue system can enrich the game plot and make it more attractive.

Quick Starts

There is an example of dialogue in startroom scene.
To quickly add a dialogue for test.Activate startroom scene and do the following jobs.



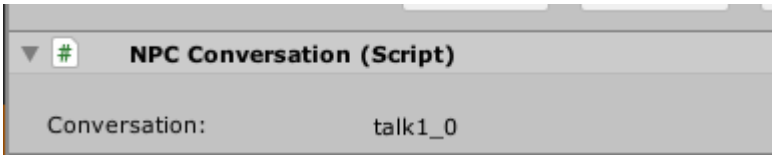
Copy these 4 gameobjects(include their children)from startroom hierarchy.
Then Paste them to any scene which you have not used for a dialogue.
The dialogue editor only support manual save currently.
So do not forget to click manual save every time before you press the run game.



Ready Dialogues

Lets see the Dialogues gameobject.We put the dialogues under this gameobject.
Every time you want to create your own dialogue. You should put them all under this folder.
To create a dialogue.Do the following.

1. create an empty gameobject under Dialogues gameobject
2. give it the component NPC Converstaion on its inspector



3.rename the gameobject to give it a dialogue name.

Let’s take a look at the Dialogue hierarchy

There are 2 dialogue gameobjects in it. But actually they are all refer to one dialogue which named talk1

Why being 2 gameobjects because we used localization for it. Add a suffix at the end of the dialogue name (from talk1 to talk1_0 and talk1_1)means it is for English and Chinese. The suffix _0 represent English language. This is due to your setting in Gamedata.

See localization section for detail.

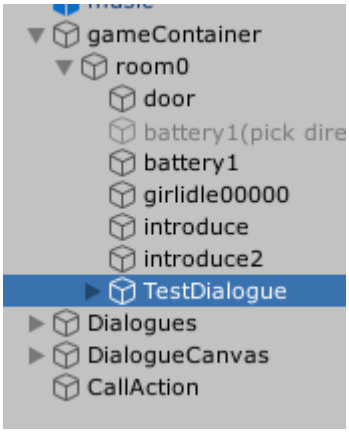
So if you want a localization,add _0,_1,_+any number for different languages.

You can also only leave talk1 without suffix directly under Dialogue gameobject if you do not want a localization.

The way of localization to a dialogue is copy and paste the dialogue gameobject which is ready for default and then rename them with name + suffix. It will be not related to the txt local file which mentioned in localization section.

Call Dialogues

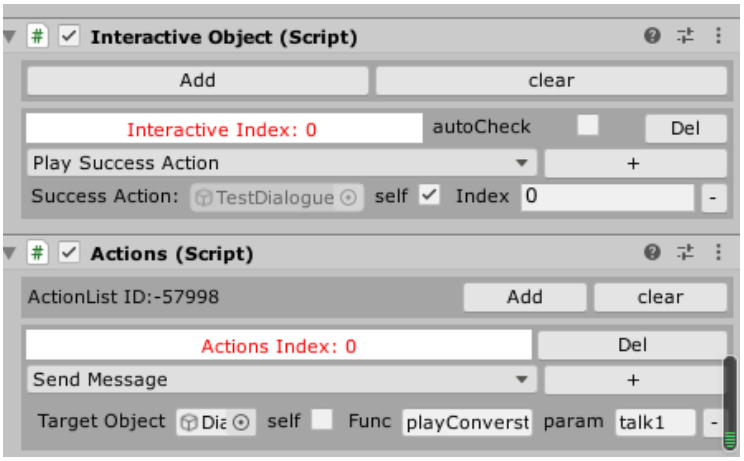
Unfold the catalogs and see this gameobject,



The TestDialouge,which is just a very common interactable item which you should already known how to create by the previous lessons.

Read the section if you are not quite familiar so.

See its inspector below



You should find it got only 1 action.

Which is send message.

This is because as the dialogue system is made as a widget,we use sendmessage to call it.

What we give the target object of sending message is DialogueCanvas which got a script binding on it. The func and param section means we would call playConverstion function and send a string which is talk1

This means you want to call the a dialogue in the Dialogues folder which named talk1

We don’t have a talk1 dialogue gameobject but 2 similar gameobjects for localization. We have explained this in the previous section.

You can open the script which attached on DialogueCanvas to see details.

Edit a Dialogue

Of course,it is the most import part. But also not too much to say. It is a node based visual editor which you can quickly understand without any tutorial.

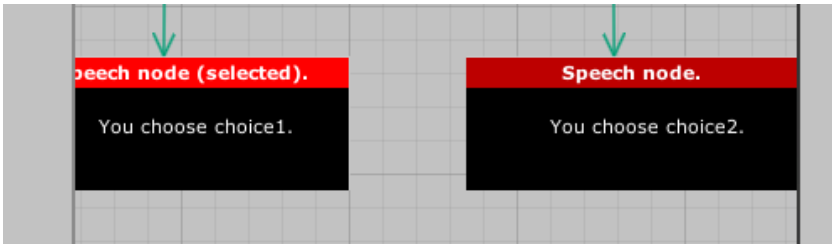
To edit a dialogue. Switch to the Dialogue editor panel or by Window->DialogueEditor

When you create a new dialogue gameobject. The root node should always being exist.

Move your mouse on any node section and right click on it.
You would have several options like:
Create Option: Create a green node which mean the an option for player to choose.
Create Speech: Directly create a linked node for next talk.
Connect: create a line with arrow for you to link from node to node.
Delete: Delete a talk or speech.
There are also some options on each node when you select them.
The Character name: Any name you want to appear on the Dialogue UI. Can be leave blank.
Dialogue: The talk content. Do not make this too long to exceed the UI section.
Icon: Give each talk a profile/avatar, you can edit The UI arrangement in dialogueCanvas to make different visions.
Audio: As beta version, just for special short sound/SFX. Do not use it for talk voice.
Event: Tell the engine to do something at the start of this node. We would talk it later.

Call Events

Each node of a dialogue can call different actions.
In our example, activate any dialogue and see its last nodes.



We see we set the node event to call a script.
Which is binding on the CallAction gameobject.

Let’s see its inspector.

There seems 2 actions included which each refers to one event.

When you choose node 1 action index 0 would be called so the testDialogue gameobject would keep being visible. If you choose speech node 2,the action 2 would be called so the gameobject would be hide.

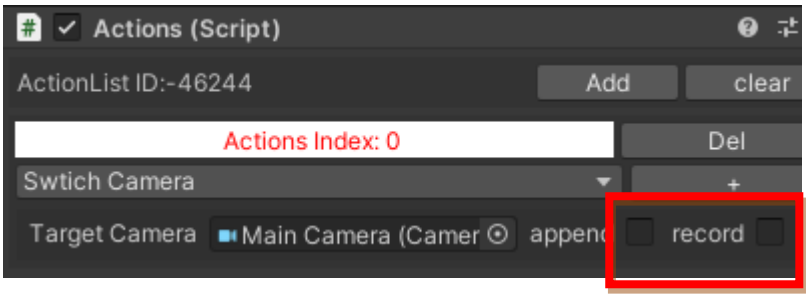
Now you can create your own actions for the game which happens during a conversation. Remember you should add separated index for each action. If you add several actions in one action. The command would keeping execute all its list.

See detail about this part in Actions.

New Updated Functions

V1.05

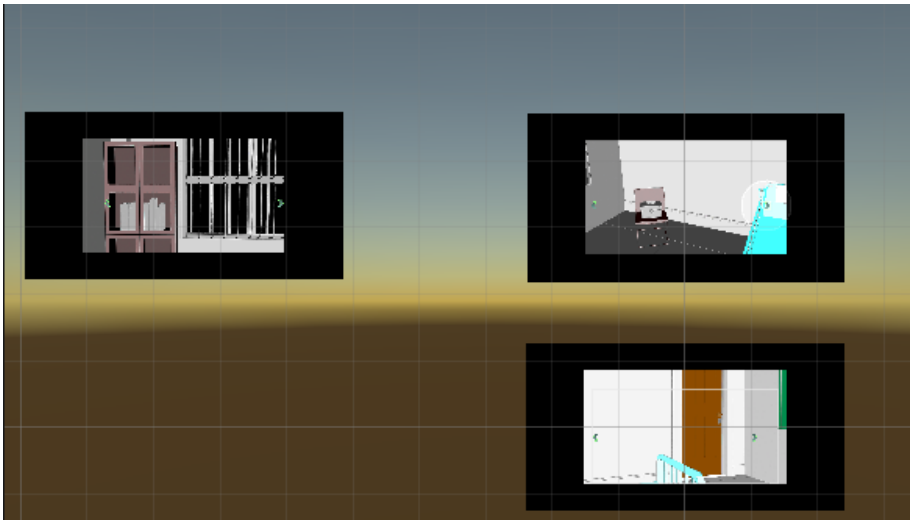
1.05 add a new checkbox for the switch camera action



The new parameter is record

This parameter is checked by default. Uncheck when you want to switch cameras within a scene.

Like the following.



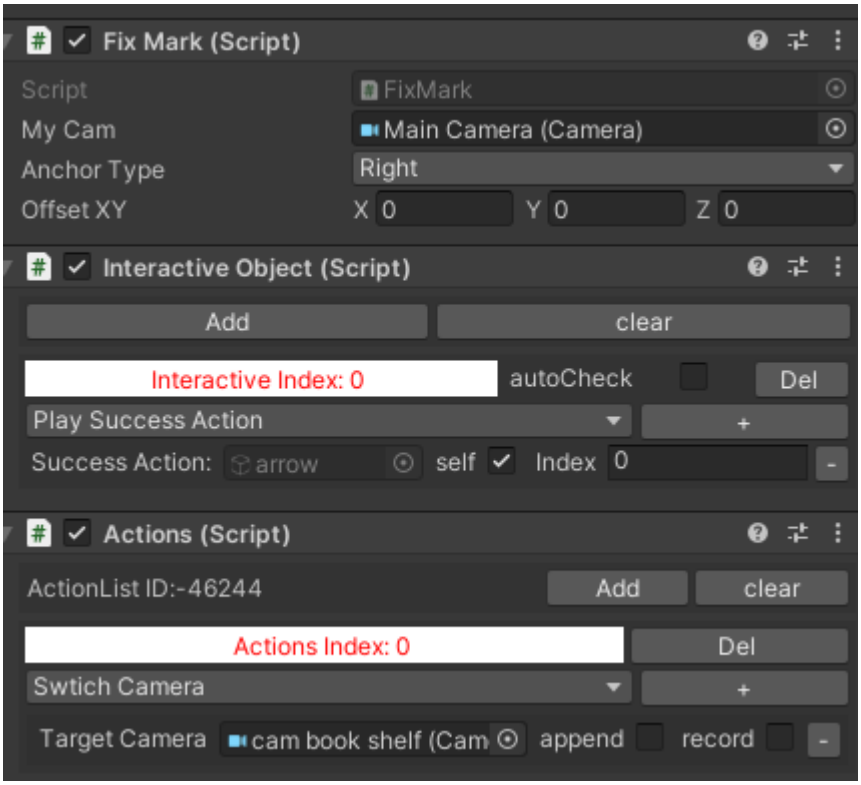
This scene got 3 cameras for each location.

Open the example quickswitch at unity/Hitcode for detail

To scene between camera within one scene. You should add a fixMark script for switch button.

Like this.

Attach fixMark.cs for any sprite which you want to use a button



Inspector detail of this arrow

Sprite attached with Fix Mark can be snapped and align automatically by the current camera.
My cam: this arrow shows in the main camera. Assign a proper camera for this section.
Anchor Type: Arrow buttons can would be snapped and arranged automatically.
Offset XY: Give offset position based on your Anchor setting.

For this example, As the record is not checked, when you click the arrow. The camera would switch instantly to another place but won’t leave the current scene or overlay the camera.
Open the example quickswitch at unity/Hitcode for detail

The

All Video Tutorials

Create a new game

Create a new room

Interactive basics

Save the state

Pick up items

Craft item

Switch camera

Dymatic Exit Mark

Minigames

Hold control + click to open the url. If it not works. Right click on the link to copy url links and pasted into your browser.

Publish to appstore or WebGL

Ready Dotween

As unity disallow uploading package including other package on store(even free). You should delete the minimal dotween dll and upload the full dotween package from official site manually.
Be easy. Just follow the steps
1.find …/scripts/tools/tweentool
there are 3 .dll files in it.
2.Delete this folder
3.download dotween from the official site or search dotween in asset store
4.Import dotween asset package
5.unity menu:"tools-dotween utility panel"
6 click setup dotween.

FAQ

build and submit a game on app store with unity.

link

I got error while building with ill2cpp

link

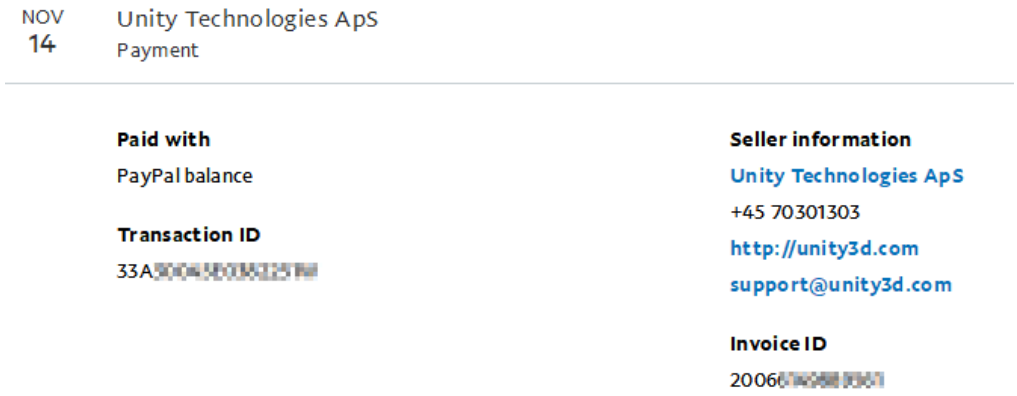
I got plugin error during import

Sometimes you may got errors like following
com.unity.multiplayer-hlapi@1.0.4 the multiplayer-hlapi
The red words can be anything like collab,textmesh pro or something.
Due to unity start to use package manager for in-build assets. Something would be happened unexpectly due to upgrade.
To solve this just remove/update the package in package manager.
In your case. the error happened on com.unity.multiplayer-hlapi@1.0.4 the multiplayer-hlapi is something seems effect your project.
My asset did not rely on these assets so you can just try to fix by removing them.
You can do the following
Window->package manager
select unity registry catalogy then search mutiplayer-hlapi
like the image

How to get support

Contact to us E-mail

Remember attach your invoice otherwise there would not be my reply.


NOV 14

Unity Technologies ApS

Payment

Paid with

PayPal balance

Transaction ID

33A

Seller information

Unity Technologies ApS

+45 70301303

http://unity3d.com

support@unity3d.com

Invoice ID

2006

The invoice you can get from your paypal account records.
If you do not have a invoice. Grab some screenshot to confirm your buy successful flow is also ok.

As we are busy to dealt with many emails. Reply maybe would last 1-2 days. Sorry for the inconvenience.

If you want support our work or feel interested in other assets, take a look at More Games