DF

BOX

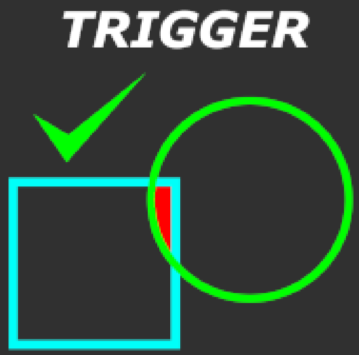
USER GUIDE

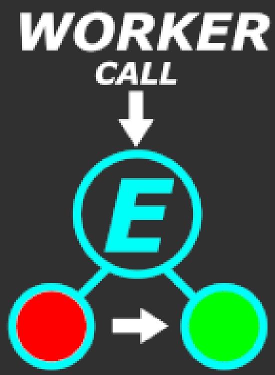
About

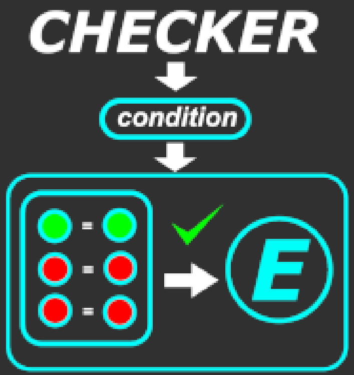
DF_BOX is for creating realtime interactions for game designers and developers on the Unity development platform without code.

This set of tools is designed to create interactive interactions in the scene. Working with animation, sound, and building logic between components will be easier and faster.

The set is suitable for both beginners and professionals. If necessary, you can expand the functionality and add new features. Nothing is limited, all the code is open and available to users.







CONTENTS

MAIN

COMPONENTS

GRAPHICS

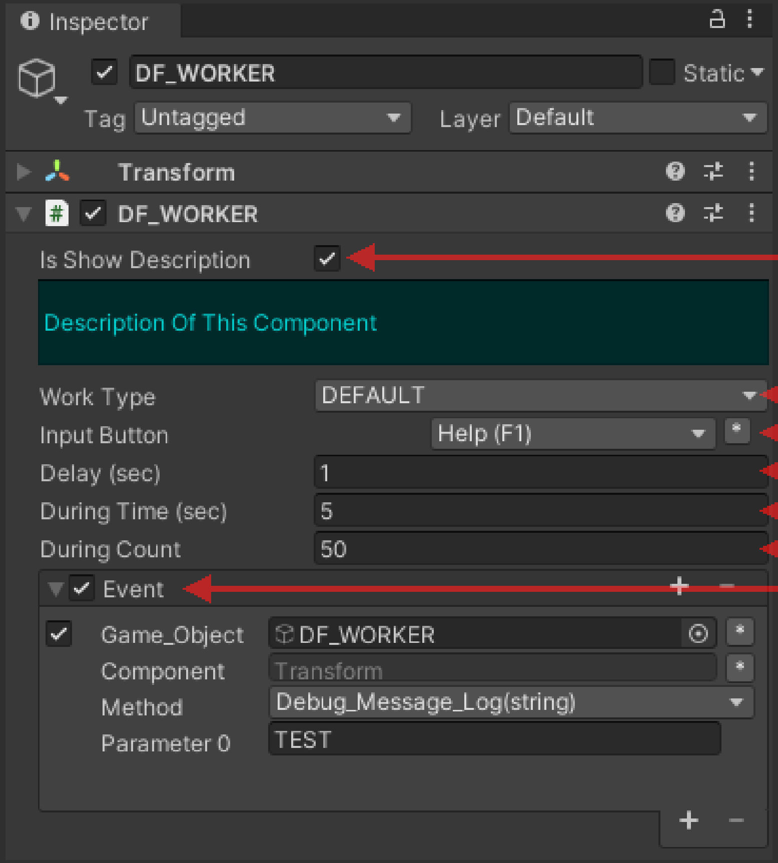
MISC

PHYSICS

SOUN

MANAGERS

TEMPORARY



☑ DF_WORKER □ Static

Tag Untagged   Layer Default

☑ DF_WORKER

Is Show Description ☑

Description Of This Component

Work Type DEFAULT

Input Button Help (F1)

Delay (sec) 1

During Time (sec) 5

During Count 50

☑ Game_Object DF_WORKER

Component Transform

Method Debug_Message_Log(string)

Parameter 0 TEST

===DF_WORKER=== VIDEO

Used to perform any actions, with or without delay, called from other events or controlled by the user, or for a specified time or number of repetitions

If the toggle is on - Show TextField to describe this component

Work Type (DEFAULT, INPUT, TIME, COUNT) ***

Button for call events (Work Type - INPUT)

Delay before calling events (Work Type - TIME or COUNT)

During this time value, events will be called (Work Type - TIME)

During this value event will be called (Work Type - COUNT)

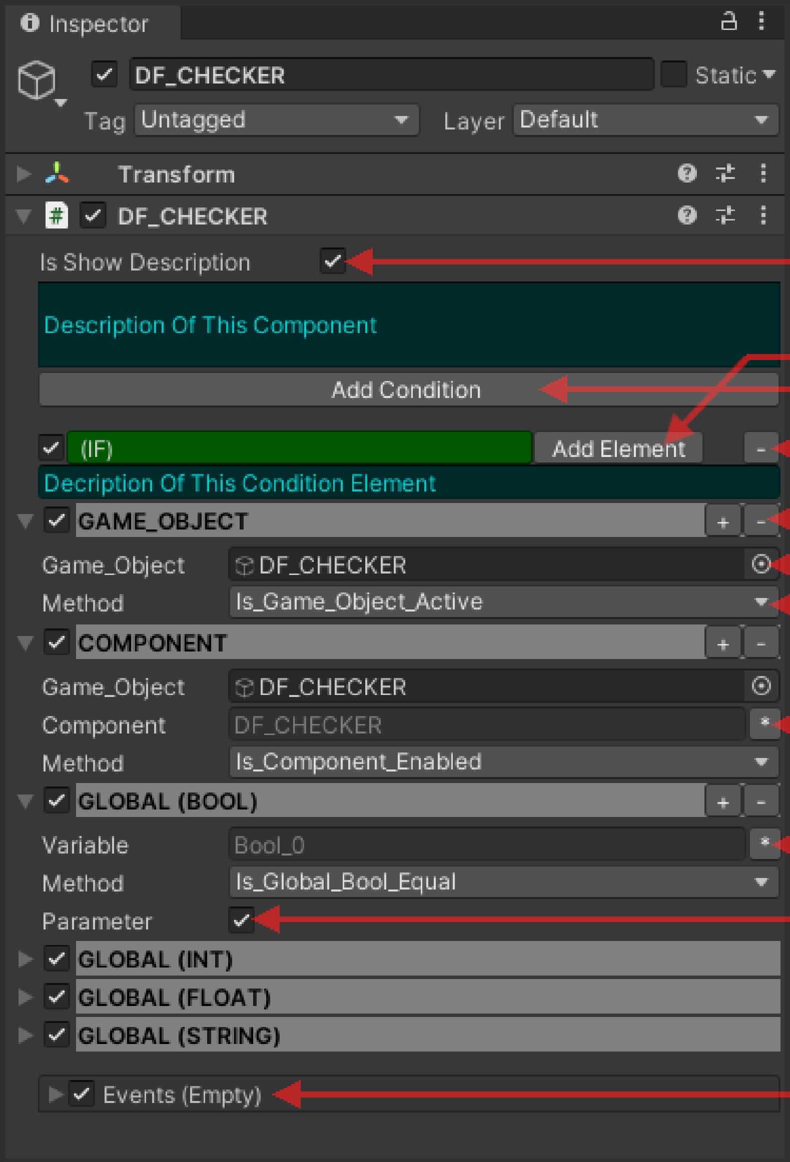
Events called when the current condition is done

***

DEFAULT - Events can only be called from other events
INPUT - Events can only be called when the button is pressed
TIME - Events can only be called after the delay has expired and only during the specified time
COUNT - Events can only be called after the delay has expired and only during the specified countdown

For Use In Events (Called Methods)

Work_Start() - Activating the DF_WORKER component of the current GameObject
Work_Stop() - Deactivating the DF_WORKER component of the current GameObject



Inspector

☑ DF_CHECKER □ Static

Tag Untagged Layer Default

Transform

☑ DF_CHECKER

Is Show Description ☑

Description Of This Component

Add Condition

☑ (IF) Add Element -

Decription Of This Condition Element

☑ GAME_OBJECT + -

Game_Object DF_CHECKER

Method Is_Game_Object_Active

☑ COMPONENT + -

Game_Object DF_CHECKER

Component DF_CHECKER

Method Is_Component_Enabled

☑ GLOBAL (BOOL) + -

Variable Bool_0

Method Is_Global_Bool_Equal

Parameter ☑

☑ GLOBAL (INT)

☑ GLOBAL (FLOAT)

☑ GLOBAL (STRING)

☑ Events (Empty)

═══DF_CHECKER═══ VIDEO

Used to check conditions (if all the elements in the condition are true, then the event of this condition is executed). The checked condition elements can be of the following types (GameObject, Component, or Global variables (Bool, Int, Float, String)).

If the toggle is on - Show TextField to describe this component

Add condition element (Game_Object, Component or Global Variable (Bool, Int, Float or String))

Add another condition for the DF_CHECKER component

Delete the current condition

Clone or remove the current condition element

GameObject to be checked

Method for checking the current element of a condition

Component to be checked

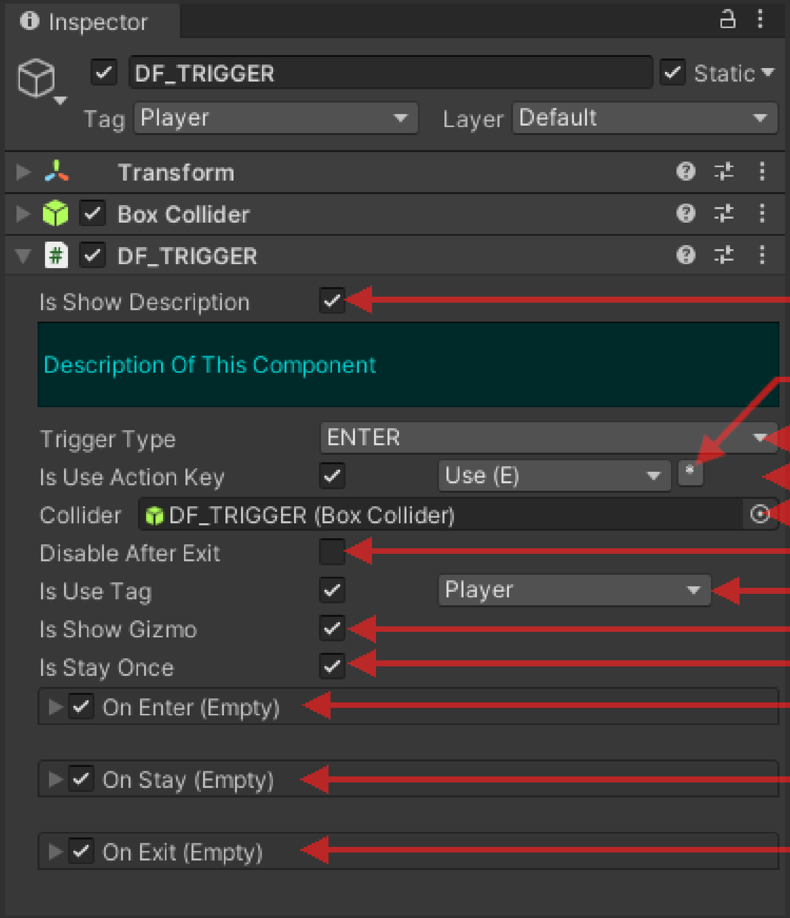
Global variable to be checked

Parameter to be checked

The event of this condition will be called when all the elements of this condition are true.

For Use In Events (Called Methods)

Check_Conditions() - Checking conditions of the DF_CHECKER component of the current GameObject (if all the elements in the condition are true, then the event of this condition is executed)



Inspector

☑ DF_TRIGGER ☑ Static

Tag Player Layer Default

Transform

☑ Box Collider

☑ DF_TRIGGER

Is Show Description ☑

Description Of This Component

Trigger Type ENTER

Is Use Action Key ☑ Use (E) *

Collider DF_TRIGGER (Box Collider)

Disable After Exit ☐

Is Use Tag ☑ Player

Is Show Gizmo ☑

Is Stay Once ☑

☑ On Enter (Empty)

☑ On Stay (Empty)

☑ On Exit (Empty)

===DF_TRIGGER=== VIDEO

Used to control events in a trigger when a GameObject enters, stands or exits the trigger collider (trigger type - ENTER). Or when a Player looks at a trigger (trigger type - LOOK) using a component DF_FPS_TRIGGER_ACTIVATOR

If the toggle is on - Show TextField to describe this component

Show DF_MANAGER_INPUT window to edit its elements

The trigger can be of two types (Enter, Look) ***

If toggle is on – The "On Stay" event will only be called when the button is pressed

The Collider for tracking collisions of GameObjects and this trigger

If toggle is on – Disable DF_TRIGGER when the object exits the collider

If toggle is on – Collision tracking will only be performed for GameObjects with this Tag

If toggle is on - Show collider gizmo

If toggle is on – The event "On Stay" will be called only once

The event will be called when the GameObject enters the collider (Trigger Type - ENTER)

The event will be called when the GameObject is in the collider (Trigger Type - ENTER)

The event will be called when the GameObject exits the collider (Trigger Type - ENTER)

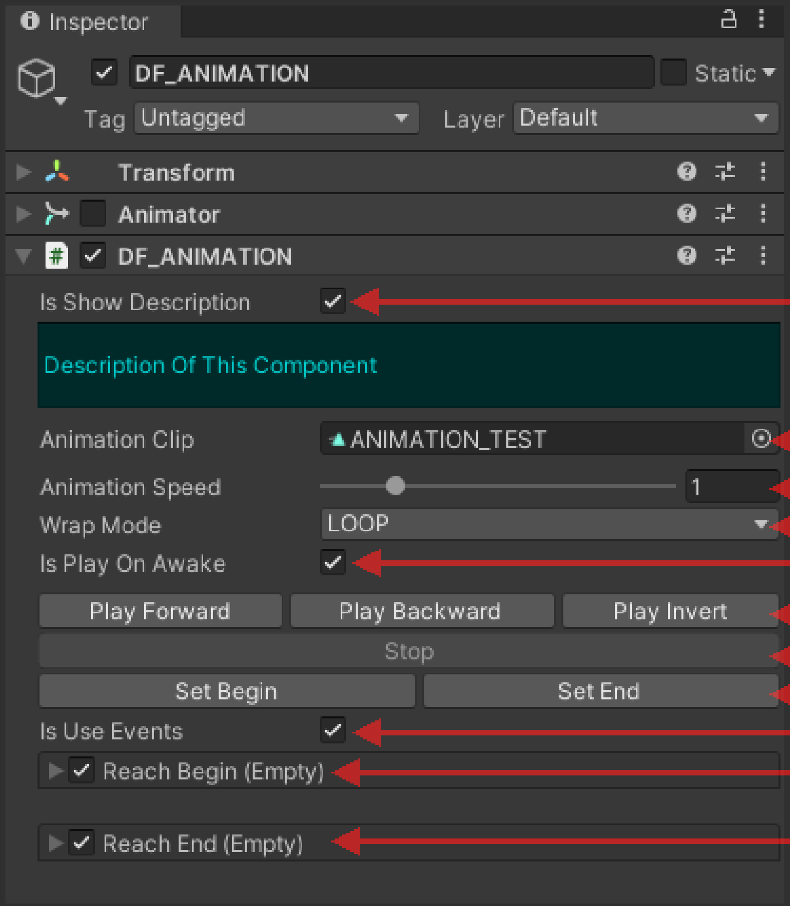
***

Trigger Type - ENTER (GameObject enters, stands or exits the trigger collider) (OnEnter, OnStay, OnExit)
Trigger Type - LOOK (Player looks at a trigger using a component DF_FPS_TRIGGER_ACTIVATOR) (OnLook)

For Use In Events (Called Methods)

Trigger_Enable(bool _is_enable) - Enable or disable the DF_TRIGGER component of the current GameObject (_is_enable is true – will be enabled and vice versa)

Trigger_Enable_Invert() - Enable or disable the DF_TRIGGER component of the current GameObject (was enabled – disable and vice versa)



Inspector

☑ DF_ANIMATION □ Static

Tag Untagged Layer Default

Transform

Animator

☑ DF_ANIMATION

Is Show Description ☑

Description Of This Component

Animation Clip ANIMATION_TEST

Animation Speed 1

Wrap Mode LOOP

Is Play On Awake ☑

Is Use Events ☑

☑ Reach Begin (Empty)

☑ Reach End (Empty)

══DF_ANIMATION══ VIDEO

Used to play an animation clip

If the toggle is on - Show TextField to describe this component

Animation Clip for using in this Component

Current Animation Speed

(ONCE, LOOP, PING_PONG) *

Play this animation clip when enter in Game Mode (or when Game has started)

Play this animation clip from start to end, from end to start or or play invert

Stop playing of this animation clip

Setting the starting or end position in this animation clip

If toggle is on, show and use events (On Reach Begin - On Reach End)

Events, called when time reaches the begin of the animation clip

Events, called when time reaches the end of the animation clip

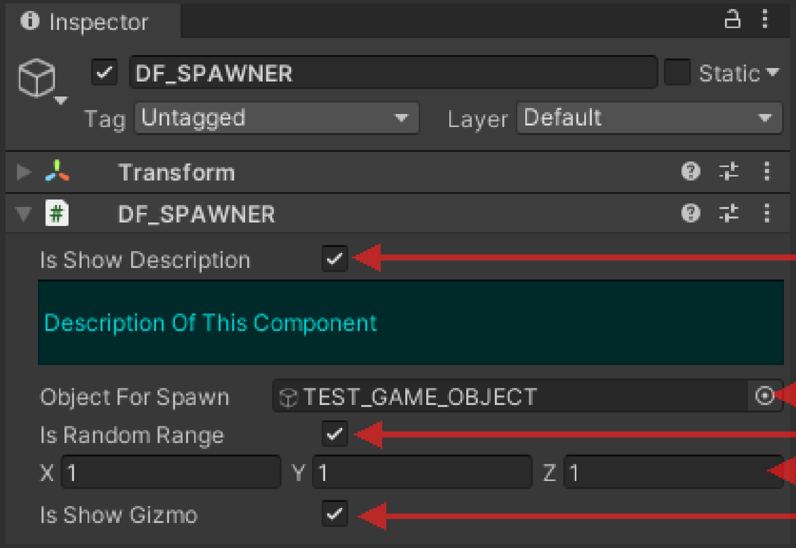
* ONCE –When time reaches the end of the animation clip, the clip will automatically stop playing and time will be reset to beginning of the clip

LOOP – When time reaches the end of the animation clip, time will continue at the beginning

PING_PONG – When time reaches the end of the animation clip, time will ping pong back between beginning and end

For Use In Events (Called Methods)

Play_Forward()	- Play the current animation clip from begin to end
Play_Backward()	- Play the current animation clip from end to begin
Play_Invert()	- Play this animation clip from start to end or vice versa (depends on where the animation is (start or end))
Pause()	- Pause the current animation clip
Stop()	- Stop the current animation clip
Set_Speed()	- Setting the speed of the current animation clip
Animation_Enable(bool _is_enable)	- Enable or disable the DF_ANIMATION component of the current GameObject (_is_enable is true – will be enabled and vice versa)
Animation_Enable_Invert()	- Enable or disable the ANIMATION component of the current GameObject (was enabled – disable and vice versa)



☑ DF_SPAWNER □ Static

Tag Untagged Layer Default

Is Show Description ☑

Description Of This Component

Object For Spawn TEST_GAME_OBJECT

Is Random Range ☑

X 1 Y 1 Z 1

Is Show Gizmo ☑

===DF_SPAWNER===

VIDEO

Used to create a GameObject in the scene with random or transform coordinates

If the toggle is on - Show TextField to describe this component

GameObject to be created in the scene

If toggle is on - GameObject will be created with random coordinates

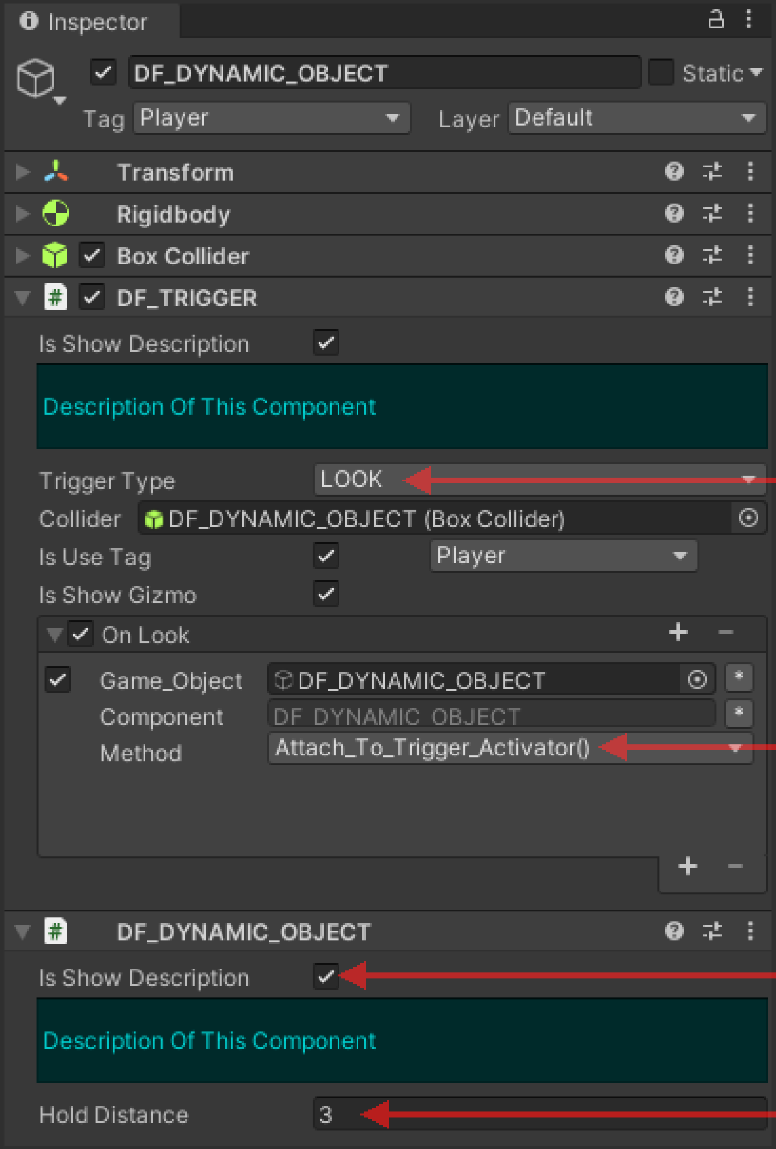
GameObject will be created with random coordinates in the given range (relative to the bounding box)

If toggle is on - Show collider gizmo

For Use In Events (Called Methods)

Spawn() - Creating a given GameObject in the scene

Spawn_And_Destroy(float _delay) - Creation of the given GameObject in the scene (this object will be destroyed after a delay)



DF_DYNAMIC_OBJECT

Tag: Player

Layer: Default

DF_TRIGGER

Is Show Description ☑

Description Of This Component

Trigger Type: LOOK

Collider: DF_DYNAMIC_OBJECT (Box Collider)

Is Use Tag ☑ Player

Is Show Gizmo ☑

On Look

☑ Game_Object: DF_DYNAMIC_OBJECT

Component: DF_DYNAMIC_OBJECT

Method: Attach_To_Trigger_Activator()

DF_DYNAMIC_OBJECT

Is Show Description ☑

Description Of This Component

Hold Distance: 3

===DF_DYNAMIC_OBJECT===

VIDEO

Used to control a dynamic object (take, put, throw) using the DF_FPS_TRIGGER_ACTIVATOR component

The trigger type must be of the LOOK type !

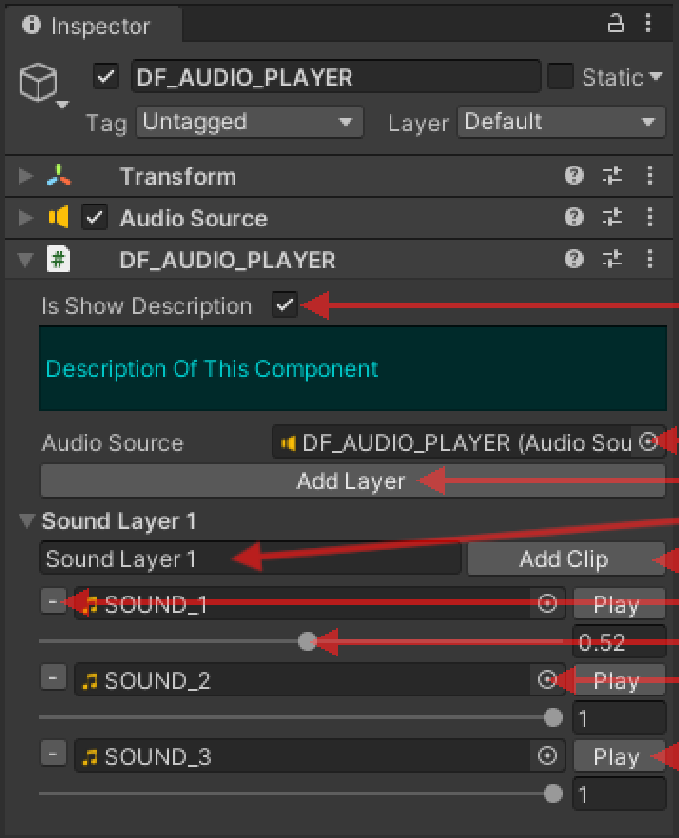
The "On Look" event must contain the Attach_To_Trigger_Activator () method !

If the toggle is on - Show TextField to describe this component

Dynamic object hold distance

For Use In Events (Called Methods)

Attach_To_Trigger_Activator() - Attaching this GameObject to DF_TRIGGER_ACTIVATOR (the player will hold this GameObject)



===DF_AUDIO_PLAYER=== VIDEO

Used to store and play sound clips

If the toggle is on - Show TextField to describe this component

Audio Source for playing sound clips

Add sound layer (array of sound clips)

Name of this sound layer

Add audio clip in this sound layer

Delete sound clip from this sound layer

The volume of this audio clip during playback

Object Field for audio clip

Play this audio clip

For Use In Events (Called Methods)

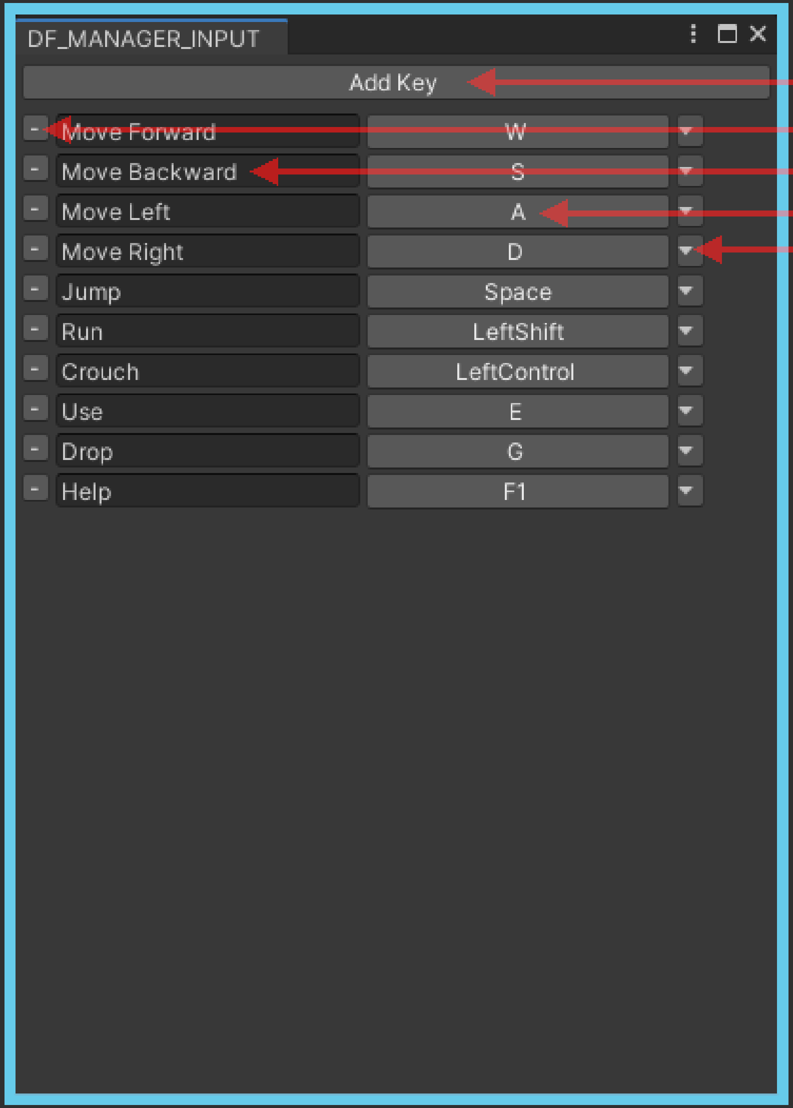
Play_Random(int _index_layer) - Play a random sound of the DF_AUDIO_PLAYER component (_index_layer is the index of the sound layer from which the sound will be played)

Play(int _index_layer, int _index_audio_clip) - Play a sound of the DF_AUDIO_PLAYER component (_index_audio_clip is the index of the audio clip that will be played)

Stop() - Stop playing any sound of the DF_AUDIO_PLAYER component of the current GameObject

===DF_MANAGER_INPUT=== VIDEO

Used to store, manage and use user input keys (Keyboard type) available in any scene of the current project



DF_MANAGER_INPUT

Add Key

-	Move Forward	W	▾
-	Move Backward	S	▾
-	Move Left	A	▾
-	Move Right	D	▾
-	Jump	Space	▾
-	Run	LeftShift	▾
-	Crouch	LeftControl	▾
-	Use	E	▾
-	Drop	G	▾
-	Help	F1	▾

Add new input element

Delete current input element

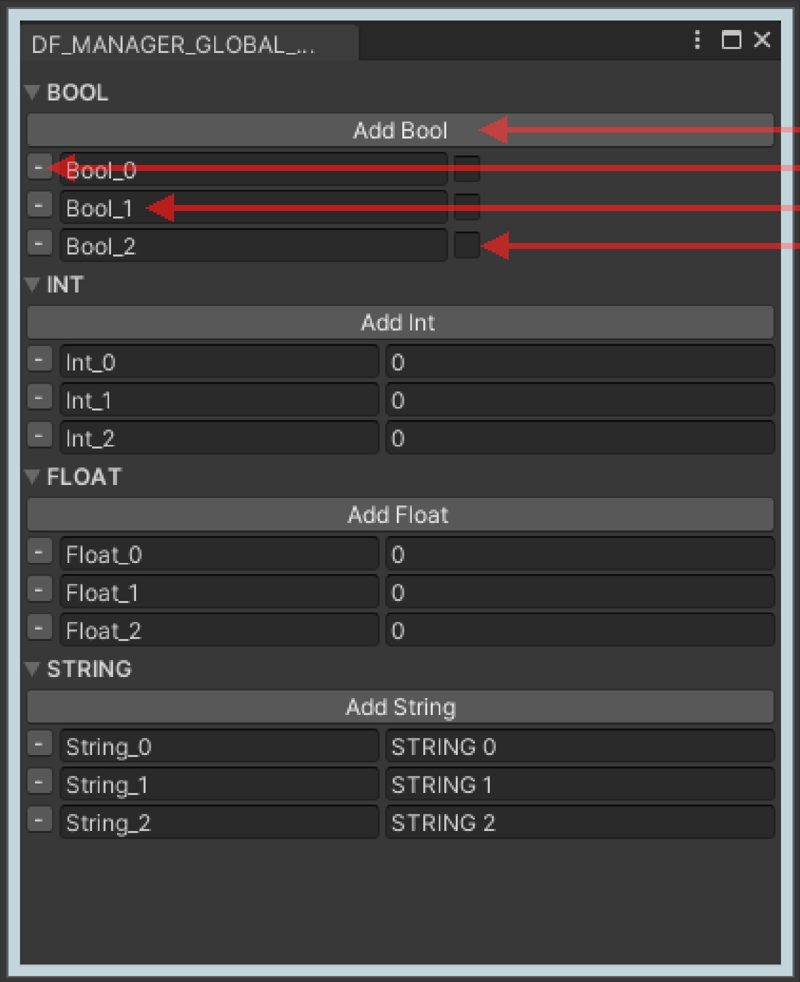
The name of the current input element

Button to select the current input element by pressing the button

Button for selecting the current input element in the standard way

═══DF_MANAGER_GLOBAL_VARIABLES═══ VIDEO

Used to store, manage and use global variables (such as Bool, Int, Float, String) available in any scene of the current project



DF_MANAGER_GLOBAL_...

▼ BOOL

Add Bool

- Bool_0 □
- Bool_1 □
- Bool_2 □

▼ INT

Add Int

Int_0	0
Int_1	0
Int_2	0

▼ FLOAT

Add Float

Float_0	0
Float_1	0
Float_2	0

▼ STRING

Add String

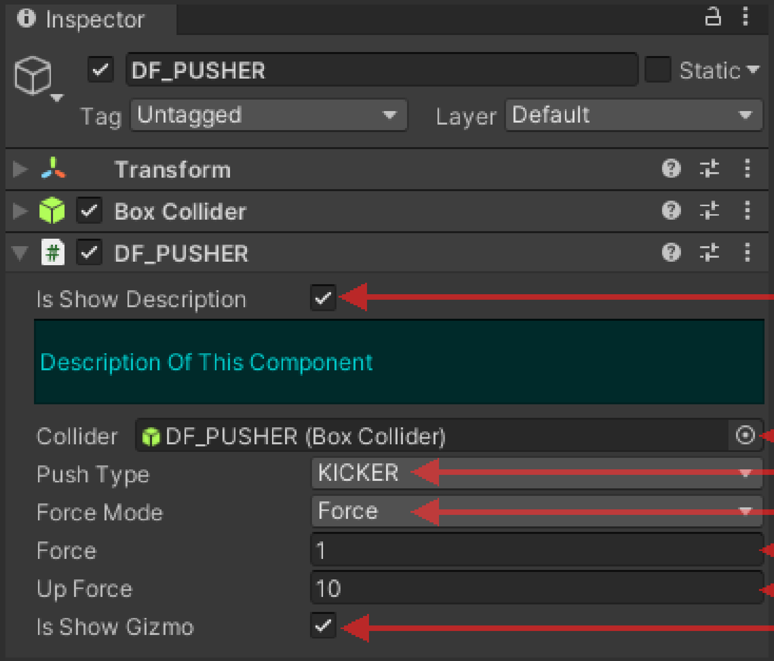
String_0	STRING 0
String_1	STRING 1
String_2	STRING 2

Add (bool, int, float, string) variable

Delete this (bool, int, float, string) variable

The Name of this (bool, int, float, string) variable

The parameter of this (bool, int, float, string) variable



DF_PUSHER

Tag Untagged

Layer Default

Is Show Description ☑

Description Of This Component

Collider DF_PUSHER (Box Collider)

Push Type KICKER

Force Mode Force

Force 1

Up Force 10

Is Show Gizmo ☑

==DF_PUSHER==

Used to physically impact a colliding GameObject (lifting or moving) - (FAN or KICKER)

If the toggle is on - Show TextField to describe this component

Collider, in the volume of which the force DF_PUSHER' acts
The type of this component (KICKER, FAN) ***
The type of force acting on the GameObject (Force, Acceleration, Impulse, Velocity Change) ***
The value of the force that will act on the colliding GameObject
Adjustment to the apparent position of the explosion to make it seem to lift objects (KICKER)
If toggle is on - Show collider gizmo

***
KICKER - Push the GameObject when entering the collider
FAN - Moving the GameObject in the collider

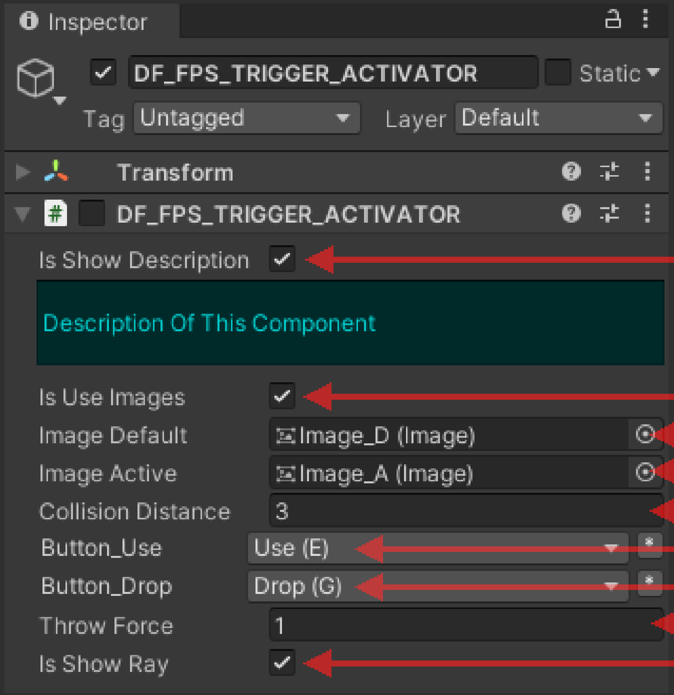
***
Force - Add a continuous force to the rigidbody, using its mass
Acceleration - Add a continuous acceleration to the rigidbody, ignoring its mass
Impulse - Add an instant force impulse to the rigidbody, using its mass
VelocityChange - Add an instant velocity change to the rigidbody, ignoring its mass

For Use In Events (Called Methods)

Pusher_Enable(bool _is_enable) - Enable or disable the DF_PUSHER component of the current GameObject (_is_enable is true – will be enabled and vice versa)
Pusher_Enable_Invert() - Enable or disable the DF_PUSHER component of the current GameObject (was enabled – disable and vice versa)

===DF_FPS_TRIGGER_ACTIVATOR=== VIDEO

Used to manage dynamic objects and triggers (trigger type is look)



Inspector

☑ DF_FPS_TRIGGER_ACTIVATOR □ Static

Tag Untagged Layer Default

Transform

DF_FPS_TRIGGER_ACTIVATOR

Is Show Description ☑

Description Of This Component

Is Use Images ☑

Image Default: Image_D (Image)

Image Active: Image_A (Image)

Collision Distance: 3

Button_Use: Use (E)

Button_Drop: Drop (G)

Throw Force: 1

Is Show Ray ☑

If the toggle is on - Show TextField to describe this component

If toggle is on - Change cursor images when this component is looking at the trigger of the LOOK type

The displayed image when this component does not point at the trigger of type LOOK

The displayed image when this component points at the trigger of type LOOK

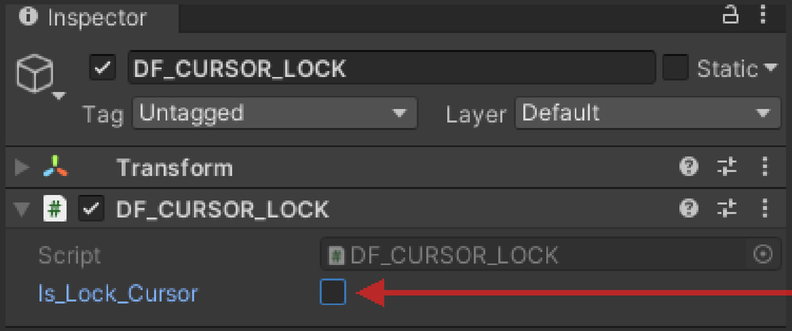
The Distance within which there is an intersection check with a trigger type of the LOOK type

The Input button to use a dynamic object

The Input button to place a dynamic object

The force with which the dynamic object will be thrown (by pressing the "Drop" button)

If toggle is on - Show pointing ray (ray length is equal to collision distance value)



☑ DF_CURSOR_LOCK □ Static

Tag Untagged Layer Default

☑ DF_CURSOR_LOCK

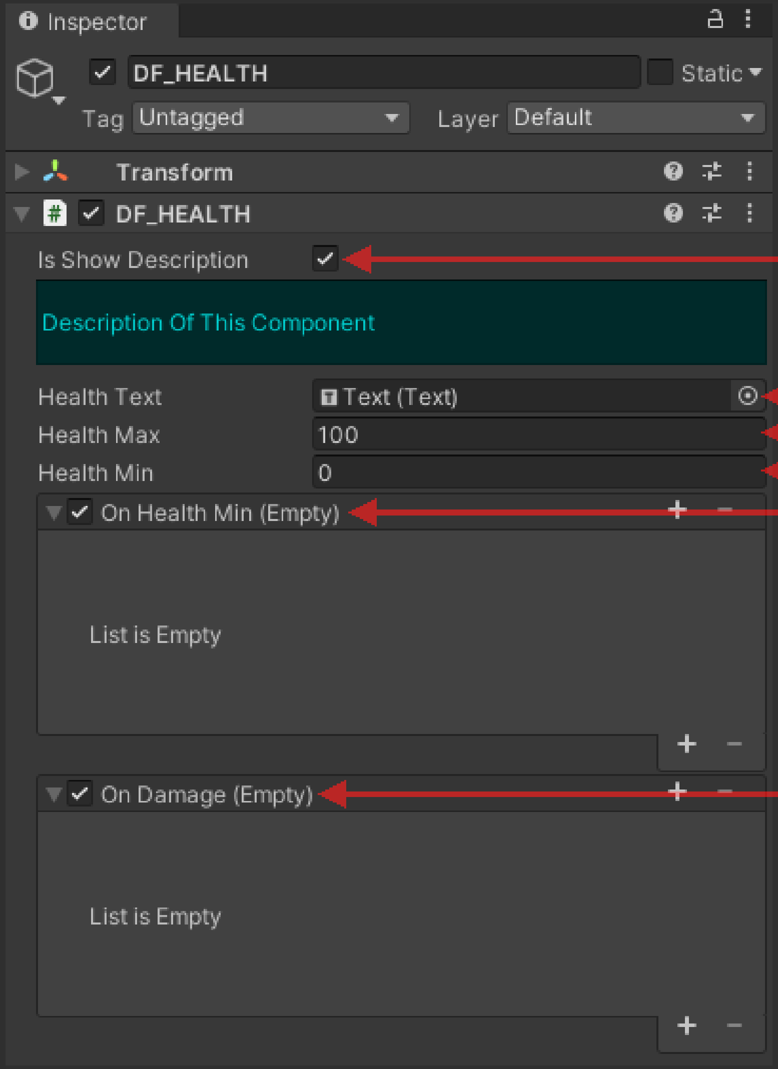
Script DF_CURSOR_LOCK

Is_Lock_Cursor □

===DF_CURSOR_LOCK===

Used to hide and lock the cursor in game (or game mode)

If toggle is on – Hide And lock the cursor in the game (or game mode)



☑ DF_HEALTH

Tag Untagged

Layer Default

Is Show Description ☑

Description Of This Component

Health Text Text (Text)

Health Max 100

Health Min 0

☑ On Health Min (Empty)

List is Empty

☑ On Damage (Empty)

List is Empty

===DF_HEALTH===

Used to store an integer health value and call events if the value has changed

If the toggle is on - Show TextField to describe this component

The text component to display the current health value

Maximum health value (the value must be greater than the minimum health value)

Min health value (the value must be greater than or equal to zero and less than the max health value)

Events called when the current health value reaches its minimum

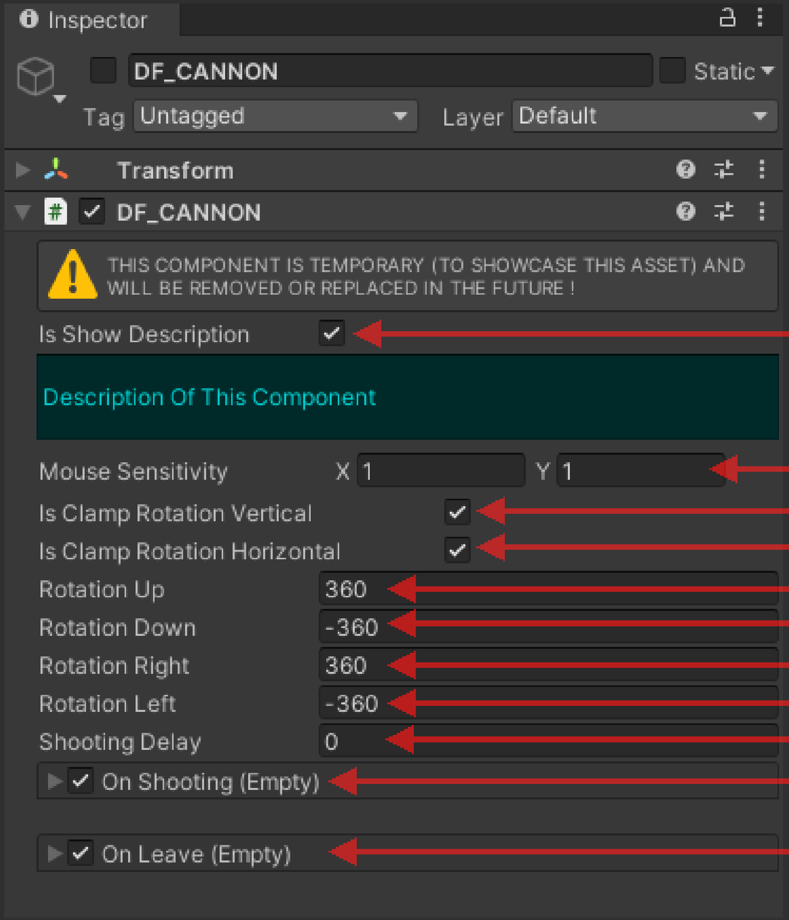
Events called when a player takes damage

For Use In Events (Called Methods)

Apply_Damage(float _value) - Setting the Damage of the current GameObject (current GameObject must have DF_HEALTH component)
Apply_Heal(float _value) - Setting the Heal of the current GameObject (current GameObject must have DF_HEALTH component)

===DF_CANNON===

Used to create and control a cannon



Inspector

☐ DF_CANNON ☐ Static

Tag Untagged Layer Default

Transform

☑ DF_CANNON

THIS COMPONENT IS TEMPORARY (TO SHOWCASE THIS ASSET) AND WILL BE REMOVED OR REPLACED IN THE FUTURE !

Is Show Description ☑

Description Of This Component

Mouse Sensitivity X 1 Y 1

Is Clamp Rotation Vertical ☑

Is Clamp Rotation Horizontal ☑

Rotation Up 360

Rotation Down -360

Rotation Right 360

Rotation Left -360

Shooting Delay 0

☑ On Shooting (Empty)

☑ On Leave (Empty)

If the toggle is on - Show TextField to describe this component

The sensitivity of the mouse when rotating this component

If the toggle is on - Rotation limitation in the up-down direction

If the toggle is on - Rotation limitation in the left-right direction

Rotation limitation in the up direction

Rotation limitation in the down direction

Rotation limitation in the right direction

Rotation limitation in the left direction

Shooting delay (in seconds)

Events that are called when shooting

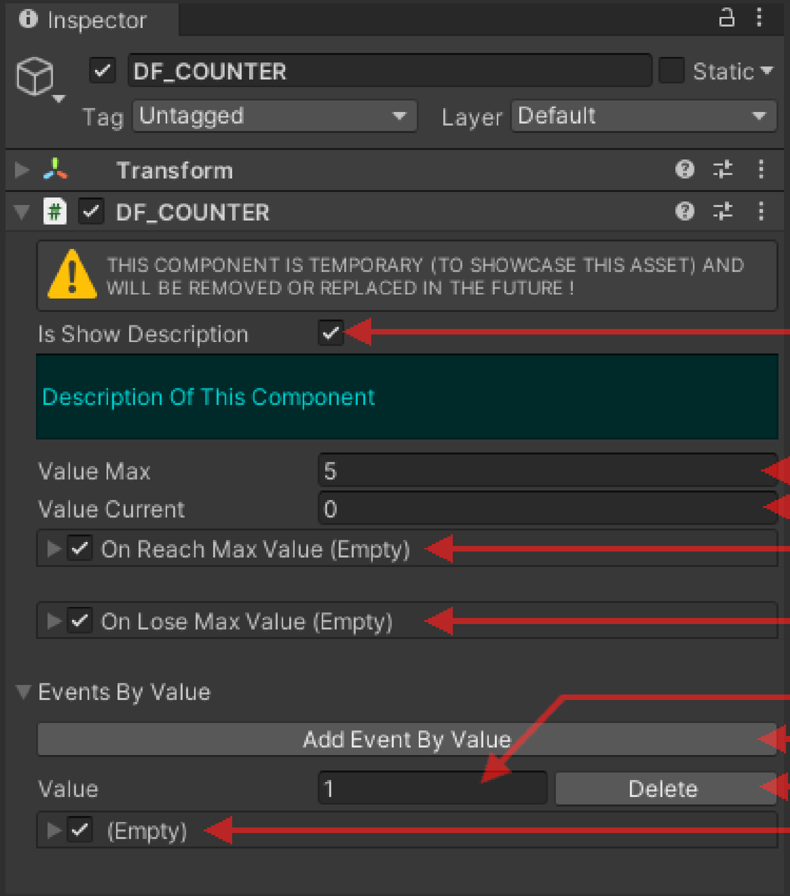
Events that are called when the player stops using the cannon

For Use In Events (Called Methods)

Cannon_Enable(bool _is_enable) - Enable or disable the DF_CANNON component of the current GameObject (_is_enable is true – will be enabled and vice versa)

Cannon_Enable_Invert() - Enable or disable the DF_CANNON component of the current GameObject (was enabled – disable and vice versa)

Cannon_Call_Events_On_Leave() - Calling the leaving event of the DF_CANNON component of the current GameObject



Inspector

☑ DF_COUNTER □ Static

Tag Untagged Layer Default

Transform

☑ DF_COUNTER

THIS COMPONENT IS TEMPORARY (TO SHOWCASE THIS ASSET) AND WILL BE REMOVED OR REPLACED IN THE FUTURE !

Is Show Description ☑

Description Of This Component

Value Max 5

Value Current 0

☑ On Reach Max Value (Empty)

☑ On Lose Max Value (Empty)

Events By Value

Add Event By Value

Value 1 Delete

☑ (Empty)

===DF_COUNTER===

Used to store an integer value and call events if the value has changed

If the toggle is on - Show TextField to describe this component

Maximum current value (the value must be greater than or equal to the min value and not equal to zero)

Current value (value must be greater than or equal to zero and less than or equal to the max value)

Events called when the current value reaches its maximum

Events called when the current value loses its maximum

The current value for this Event By Value

Add Event By Value

Delete current Event By Value

Events called when the current value becomes equal to the set value

For Use In Events (Called Methods)

Value_Set(int _value)	- Setting the current value of the DF_COUNTER
Value_Add(int _value)	- Adding the current value of the DF_COUNTER
Value_Delete(int _value)	- Deleting the current value of the DF_COUNTER
Check_Current()	- Checking the integer value of the DF_COUNTER and call events if the value has changed

=== FOR USING IN EVENTS (CALLE METHOS OF USER COMPONENTS) === OR ESCRIPTION OF EVENT ?

To call your methods from event, you need to place the F_ATTRIBUTE_METHO_FOR_CALL attribute before definition of your public method. The type of methods’ returned value is not important.

```
[F_ATTRIBUTE_METHO_FOR_CALL     ]
public void Method_For_Call_Name()
{
    ...
}

[F_ATTRIBUTE_METHO_FOR_CALL     ]
public void Method_For_Call_Name(bool value)
{
    ...
}


[F_ATTRIBUTE_METHO_FOR_CALL     ]
public void Method_For_Call_Name(string name, float value)
{
    ...
}
```

Limitations

-Your method must to be public
-Your method may to have one or two parameters or does not have any.

-If your method has one parameter, parameter must to be one of the types:

```
bool, int, float, string, Color, Vector2, Vector3, Vector4, Transform
```

If your method has two parameters, First parameter must to be int or string, Second parameter must to be one of the types:

```
bool, int, float, string, Color, Vector2, Vector3, Vector4, Transform
```

=== FOR USING IN EVENTS (CALLE METHOS OF UNITY COMPONENTS) ===

COLLIE_OBJECT

```
Collided_Object_isable   ()                              - isable the colliding GameObject
Collided_Object_estroy   (float _delay)                  - estroy the    colliding GameObject with delay
Collided_Object_Set_Position(Transform _transform)       - Setting the position of the colliding GameObject (the position value will be taken from the transform)
Collided_Object_Rigid_Body_Wake_Up()                     - Wake up the RigidBody component of the colliding GameObject
Collided_Object_Rigid_Body_Enable_Gravity(bool _is_enable) - Enable or disable the gravity in the RigidBody of the colliding GameObject (_is_enable is true – the gravity will be enabled and vice versa)
```

HEALTH

```
Collided_Object_Set_amage   (float _value) - Setting the amage of the colliding GameObject (colliding GameObject must have F_HEALTH component)
Collided_Object_Set_Heal(float _value)     - Setting the Heal      of the colliding GameObject  (colliding GameObject must have F_HEALTH component)
```

EBUG (In UnityEditor Only !)

```
ebug_Message_Log    (string _message)     - Print log message in Unity console window                              (In UnityEditor Only )
ebug_Message_ialog      (string _message) - Show message dialog window                                             (In UnityEditor Only )
ebug_Collided_Object_Get_Name   ()        - Print the name of the colliding GameObject in Unity console window (In UnityEditor Only )
```

GLOBAL_VARIABLES

```
Set_Global_Bool (int _index, bool _value)     - Setting the value of a global variable of type Bool   by its index
Set_Global_Int (int _index, int _value)       - Setting the value of a global variable of type Int      by its index
Set_Global_Float (int _index, float _value)   - Setting the value of a global variable of type Float   by its index
Set_Global_String (int _index, string _value) - Setting the value of a global variable of type String by its index
```

SCENE

```
Scene_Current_Reload()  - Reload the currently loaded scene
Scene_Load(int _index)   - Load the scene by its index in the Build Settings
```

APPLICATION

```
Application_Quit() - Exit From Play Mode (Edit Mode in Unity Editor) or Exit From Game (in Application)
```

TRANSFORM - GAME_OBJECT

```
Game_Object_Enable(bool _is_enable)                - Enable or disable the current GameObject (_is_enable is true – will be enabled and vice versa)
Game_Object_Enable_Invert()                        - Enable or disable the current GameObject (was enabled – disable and vice versa)
Game_Object_estroy   (float _delay)                - estroy the current GameObject with delay
Game_Object_Set_Position(Transform _transform) - Setting the position of the current GameObject (the position value will be taken from the transform)
Game_Object_Set_Rotation(Transform _transform) - Setting the rotation of the current GameObject (the position value will be taken from the transform)
Game_Object_Set_Scale(Transform _transform)      - Setting the scaling of the current GameObject  (the position value will be taken from the transform)
Game_Object_Set_Position(Vector3 _position)       - Setting the position of the current GameObject (the position value will be taken from the position)
Game_Object_Set_Rotation(Vector3 _rotation)       - Setting the rotation of the current GameObject (the position value will be taken from the rotation)
Game_Object_Set_Scale(Vector3 _scale)               - Setting the scaling of the current GameObject  (the position value will be taken from the scale)
```

RECT_TRANSFORM - GAME_OBJECT

```
RT_Game_Object_Enable(bool _is_enable) - Enable or disable current GameObject (_is_enable is true – will be enabled and vice versa)
RT_Game_Object_Enable_Invert()             - Enable or disable current GameObject (was enabled – disable and vice versa)
```

TEXT

```
Text_Enable(bool _is_enable)                       - Enable or disable the Text component of the current GameObject (_is_enable is true – will be enabled and vice versa)
Text_Enable_Invert()                                - Enable or disable the Text component of the current GameObject (was enabled – disable and vice versa)
Text_Set_Content(string _content)                  - Setting content of the Text component of the current GameObject
Text_Set_Content_From_Global_String(int _index) - Setting content of the Text component of the current GameObject from the global variable of type String (_index – index of the global variable)
```

RIGI_BOY

```
Rigid_Body_3_Enable_Gravity   (bool _is_enable) - Enable or disable gravity in the RigidBody component of the current GameObject (_is_enable is true – will be enabled and vice versa)
Rigid_Body_3_Wake_Up   ()                        - Wake up the RigidBody component of the current GameObject
Rigid_Body_2_Wake_Up   ()                        - Wake up the RigidBody2 component of the current GameObject
```

PSYSTEM

```
PSystem_Play()          - Play the ParticleSystem component of the current GameObject
PSystem_Play_Invert() - Play or Stop the ParticleSystem component of the current GameObject (was Played – Stop and vice versa)
PSystem_Pause()         - Pause the ParticleSystem component of the current GameObject
PSystem_Stop()          - Stop the ParticleSystem component of the current GameObject
```

MESH_RENERER

```
Mesh_Renderer_Enable(bool _is_enable) - Enable or disable the MeshRenderer component of the current GameObject (_is_enable is true – will be enabled and vice versa)
Mesh_Renderer_Enable_Invert()           - Enable or disable the MeshRenderer component of the current GameObject (was enabled – disable and vice versa)
```

MATERIAL

```
Material_Set_Color_Main(Color _color)      - Set the main color of the MeshRenderer component of the current GameObject
Material_Set_Color_Emission(Color _color) - Set the emission color of the MeshRenderer component of the current GameObject
Material_Emission_Enable(bool _enable)     - Enable or disable the emission color of the MeshRenderer component of the current GameObject (_is_enable is true – will be enabled and vice versa)
Material_Emission_Enable_Invert()          - Enable or disable the emission color of the MeshRenderer component of the current GameObject (was enabled – disable and vice versa)
```

LIGHT

```
Light_Enable(bool _is_enable)         - Enable or disable the Light component of the current GameObject (_is_enable is true – will be enabled and vice versa)
Light_Enable_Invert()                 - Enable or disable the Light component of the current GameObject (was enabled – disable and vice versa)
Light_Set_Intensity(float _intensity) - Setting the intensity of the Light component of the current GameObject
Light_Set_Range(float _range)         - Setting the range of the Light component of the current GameObject (Point Light or Spot Light Only)
Light_Set_Color(Color _color)         - Setting the light color of the Light component of the current GameObject
```

COLLIER

```
Collider_3_Box_Enable   (bool _is_enable)      - Enable or disable the BoxCollider component of the current GameObject (_is_enable is true – the BoxCollider will be enabled and vice versa)
Collider_3_Capsule_Enable   (bool _is_enable) - Enable or disable the CapsuleCollider component of the current GameObject (_is_enable is true – will be enabled and vice versa)
Collider_3_Sphere_Enable   (bool _is_enable)   - Enable or disable the SphereCollider component of the current GameObject (_is_enable is true – will be enabled and vice versa)
Collider_2_Box_Enable   (bool _is_enable)      - Enable or disable the BoxCollider component of the current GameObject (_is_enable is true – will be enabled and vice versa)
Collider_2_Capsule_Enable   (bool _is_enable)  - Enable or disable the CapsuleCollider component of the current GameObject (_is_enable is true – will be enabled and vice versa)
Collider_2_Circle_Enable   (bool _is_enable)   - Enable or disable the CircleCollider component of the current GameObject (_is_enable is true – will be enabled and vice versa)
```

CAMERA

```
Camera_Enable(bool _is_enable) - Enable or disable the Camera component of the current GameObject (_is_enable is true – will be enabled and vice versa)
Camera_Enable_Invert()         - Enable or disable the Camera component of the current GameObject (was enabled – disable and vice versa)
```

AUIO_SOURCE

```
Audio_Source_Enable(bool _is_enable) - Enable or disable the AudioSource component of the current GameObject (_is_enable is true – will be enabled and vice versa)
Audio_Source_Enable_Invert()         - Enable or disable the AudioSource component of the current GameObject (was enabled – disable and vice versa)
Audio_Source_Play()                  - Play the AudioSource component of the current GameObject
Audio_Source_Pause()                 - Pause the AudioSource component of the current GameObject
Audio_Source_Stop()                  - Stop the AudioSource component of the current GameObject
```

ANIMATOR

```
Set_Bool(string _name, bool _value)   - Setting the Bool    variable of the Animator component of the current GameObject
Set_Int(string _name, int _value)     - Setting the Int     variable of the Animator component of the current GameObject
Set_Float(string _name, float _value) - Setting the Float   variable of the Animator component of the current GameObject
Set_Trigger(string _name)             - Setting the Trigger variable of the Animator component of the current GameObject
Animator_Enable(bool _is_enable)      - Enable or disable the Animator component of the current GameObject (_is_enable is true – will be enabled and vice versa)
Animator_Enable_Invert()              - Enable or disable the Animator component of the current GameObject (was enabled – disable and vice versa)
```

BUG, ERROR REPORT

If you found bug or error, please send a message to our email address: **dfbox.help@gmail.com**

CREITS

LINKS

Youtube Channel