Easy Drag and Drop

🌷 Easy Drag and Drop Documentation!

Welcome! Easy Drag and Drop is a unity customizable plugin that allows you to create your own drag and drop system easily in just a few minutes.

Related Link: Asset Store | Discord | Trello | Documentation

Demo Video: https://www.youtube.com/watch?v=Wo4dGSAtcHM

Youtube Demo Video

How To Use ?

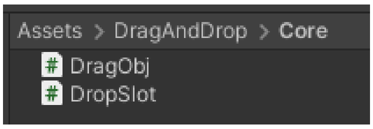
Quick Start

API Reference

Quick Start

Quick Start To Use Easy Drag and Drop Plugin

Drag and Drop Plugin Include 2 Main Core Script



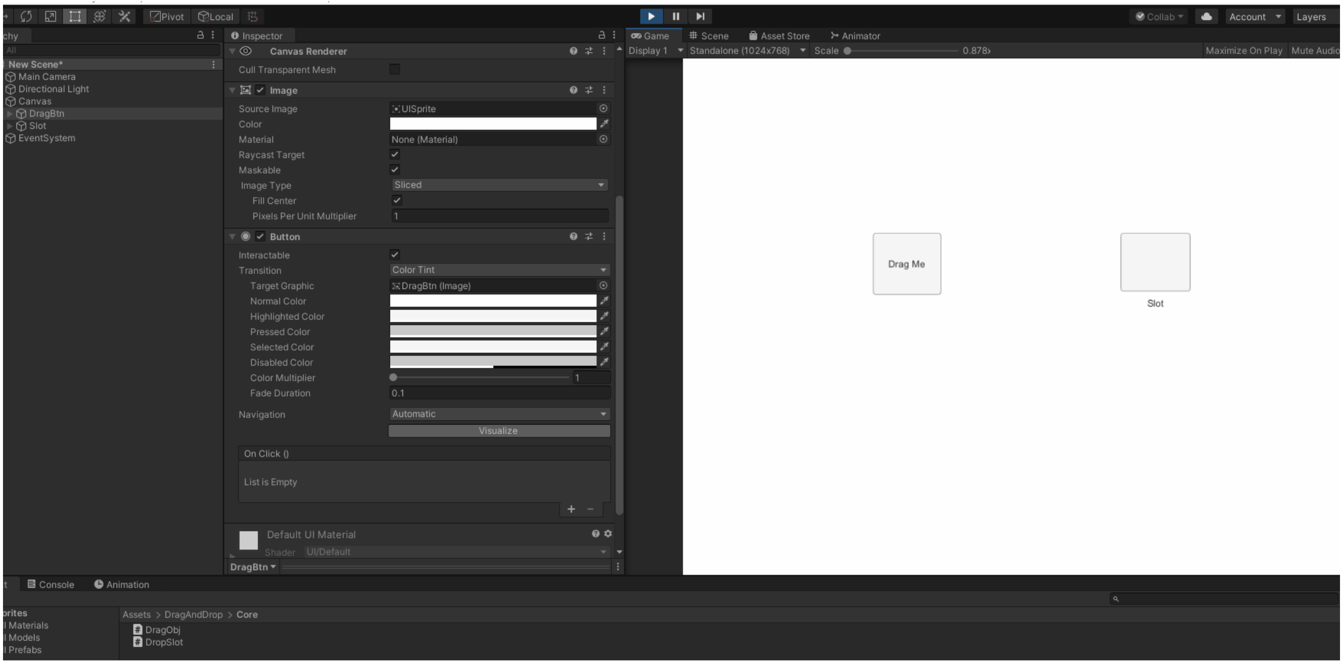
Assets > DragAndDrop > Core

DragObj

DropSlot

DragObj - Drag this c# script into the UI Element to drag the element.

DropSlot - Drag this c# script into the UI Element to drop the Drag object



Canvas Renderer

Cull Transparent Mesh □

Image

Source Image: UISprite

Material: None (Material)

Raycast Target ☑

Maskable ☑

Image Type: Sliced

Fill Center ☑

Pixels Per Unit Multiplier: 1

Button

Interactable ☑

Transition: Color Tint

Target Graphic: DragBtn (Image)

Color Multiplier: 1

Fade Duration: 0.1

Navigation: Automatic

Visualize

On Click ()

List is Empty

Default UI Material

Shader: UI/Default

DragBtn

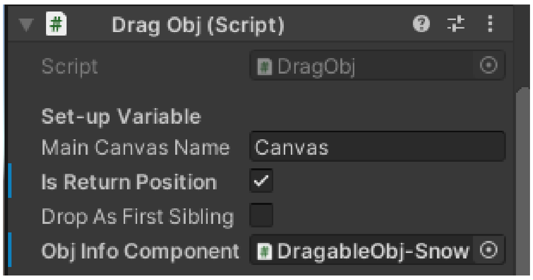
Assets > DragAndDrop > Core

DragObj

DropSlot

Example Drag and Drop GIf

- DragObj -



Drag Obj (Script)

Script: DragObj

Set-up Variable

Main Canvas Name: Canvas

Is Return Position ☑

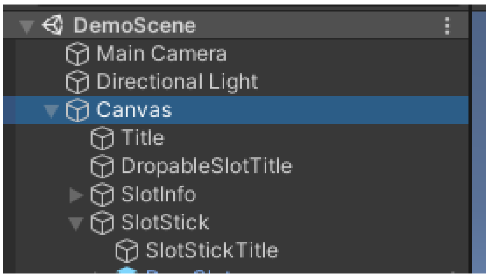
Drop As First Sibling □

Obj Info Component: DragableObj-Snow

DropObj Set-up Variable Description

- Main Canvas Name -

The main canvas name of the drag object, this is require for the raycast pointer.



DemoScene
Main Camera
Directional Light
Canvas
Title
DropableSlotTitle
SlotInfo
SlotStick
SlotStickTitle

- Is Return Position -

True - the draggable object will return to it original position.

False - the draggable object will stay at the pointer drop position.

- Drop As First Sibling -

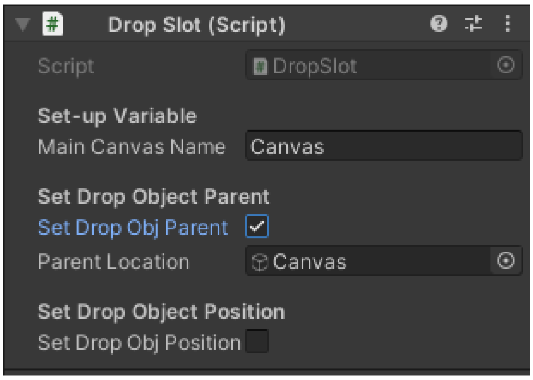
True - the draggable object will become a first sibling of the drop parent

False - the draggable object will be the last sibling of the drop parent.

- Obj Info Component -

A c# script component that containt the data structure of the draggable object. It will be store inside the DragObj and thus pass it to DropSlot for further usage.

- DropSlot -



Drop Slot (Script)

Script: DropSlot

Set-up Variable
Main Canvas Name: Canvas

Set Drop Object Parent
Set Drop Obj Parent ☑
Parent Location: Canvas

Set Drop Object Position
Set Drop Obj Position □

DropSlot Set-up Variable Description

- Main Canvas Name -

The main canvas name of the drag object, this is require for the raycast pointer.

- Set Drop Obj Parent -

True - the draggable object change it parent to **Parent Location**

False - the draggable object will remain it original parent when drop.

- Set Drop Object Position -

True - the draggable object change it position to drop slot transform position

False - the draggable object will return to its original position

DropSlot Inherient

Luckily, You can easily inherient the **DropSlot** script for further usage.

Example of inherient script in the demo scene:

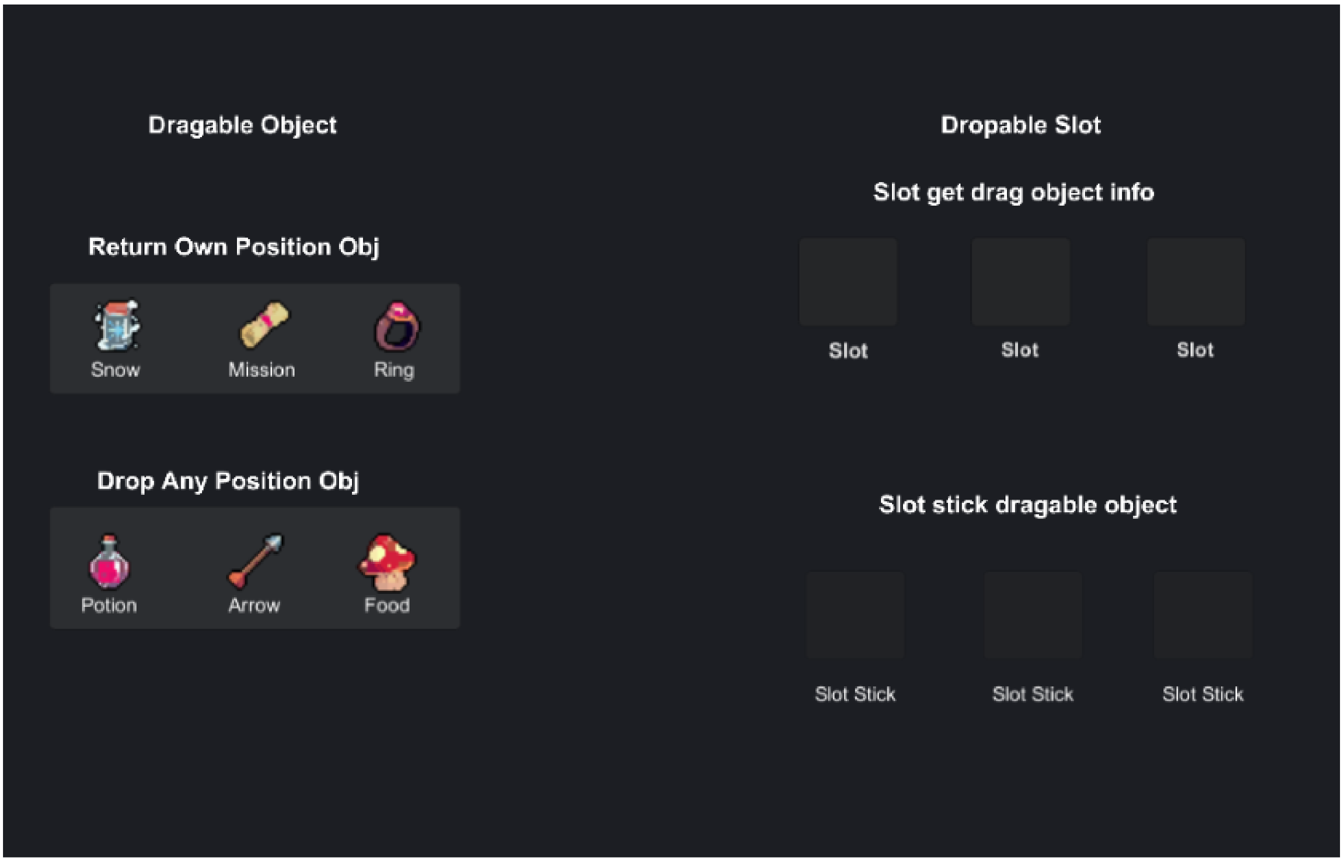
```
public class DropSlotDemo : DropSlot
    {
        public DataInfoDemo dataInfoDemo;
        public bool isOverrideOnDrop = true;
        public override void OnDrop(PointerEventData eventData)
        {
            base.OnDrop(eventData);
            if (!isOverrideOnDrop) return;
            var model = (DataInfoDemo) DropObj.m_ObjInfoComponent;
            dataInfoDemo.Initialize(model);
        }
    }

```

Simple Demo Scene

Play around with the simple demo scene to quickly understand the drag and drop function.

1. Open The DemoScene and Play It~

Path = [Assets/DragAndDrop/DemoScene/Simple Drag and Drop/DemoScene.unity]



DemoScene.unity

2. Demo Video - How To Add a Dragable Object

https://www.youtube.com/watch?v=RapRP0iHV5I

3. Demo Video - How To Add a DropSlot Object

https://www.youtube.com/watch?v=1fZ-WWdhdV4

- DropSlotDemo Description -

For more custom and further usage, you can write your own drop function by simply inherient the **DropSlot** script.

For example in the demo scene, we write a **DropSlotDemo** script which inherent the **DropSlot**, then override the OnDrop function.

```
public class DropSlotDemo : DropSlot
    {
        public DataInfoDemo dataInfoDemo;
        public bool isOverrideOnDrop = true;
        public override void OnDrop(PointerEventData eventData)
        {
            base.OnDrop(eventData);
            if (!isOverrideOnDrop) return;
            var model = (DataInfoDemo) DropObj.m_ObjInfoComponent;
            dataInfoDemo.Initialize(model);
        }
    }
```

You can get the info of the drag object by casting the **DropObj.m_ObjInfoComponent.**

Then, **do whatever you want.**

For example, we call an Initialize function which replace the slot image to the draggable object image.

Reference

API Reference

DragObj

DropSlot

> **Good to know:** Using the 'Page Link' block lets you link directly to a page. If this page's name, URL or parent location changes, the reference will be kept up to date. You can also mention a page – like **DragObj** – if you don't want a block-level link.

DragObj

DragObj Public Function, you can inherent override the function or simply call it.

Public Variable

- **string** mainCanvasName
- **bool** isReturnPosition
- **bool** dropAsFirstSibling
- **Component** m_ObjInfoComponent

- Override Function -

OnBeginDrag()

The function call when the object on begin drag.

Parameter:

- **PointerEventData** eventData - pass by the pointer event.

OnDrag()

The function call when the object on drag.

Parameter:

- **PointerEventData** eventData - pass by the pointer event.

OnEndDrag()

The function call when the object on end drag.

Parameter:

- **PointerEventData** eventData - pass by the pointer event.

- Public Function -

Initialize()

Initialize the DragObj Info by script.

Parameter:

- **Component** component - data structure need to stored by dragobj.
- **string** canvasName - main canvas name

SetDropParent()

Change the drag object parent to given transform.

Parameter:

- **Transform** transform - transform of the drop parent.

SetRectPosition()

Change the drag object position to given Vector2 position.

Parameter:

- **Vector2** position - x-y position that want the drag object to be.

SetInitialPosition()

Set the drag object position to its own initial position.

DropSlot

DropSlot public function to call or inherent override with.

Public Variable

- **DragObj** DragObj
- **string** mainCanvasName
- **bool** setDropObjParent
- **GameObject** parentLocation
- **bool** setDropObjPosition

OnDrop() - Override Function

Trigger when the draggable object drop.

Parameter:

- **PointerEventData** eventData - pass by pointer event.