Drag & Drop Pro

Version 1.95

YETALIT STUDIO

Contact me at dragdrop@pm.me

Documentation

First, we will learn how to use Drag & Drop Pro. Then, some additional tips about the plugin will be given.

Table of Contents

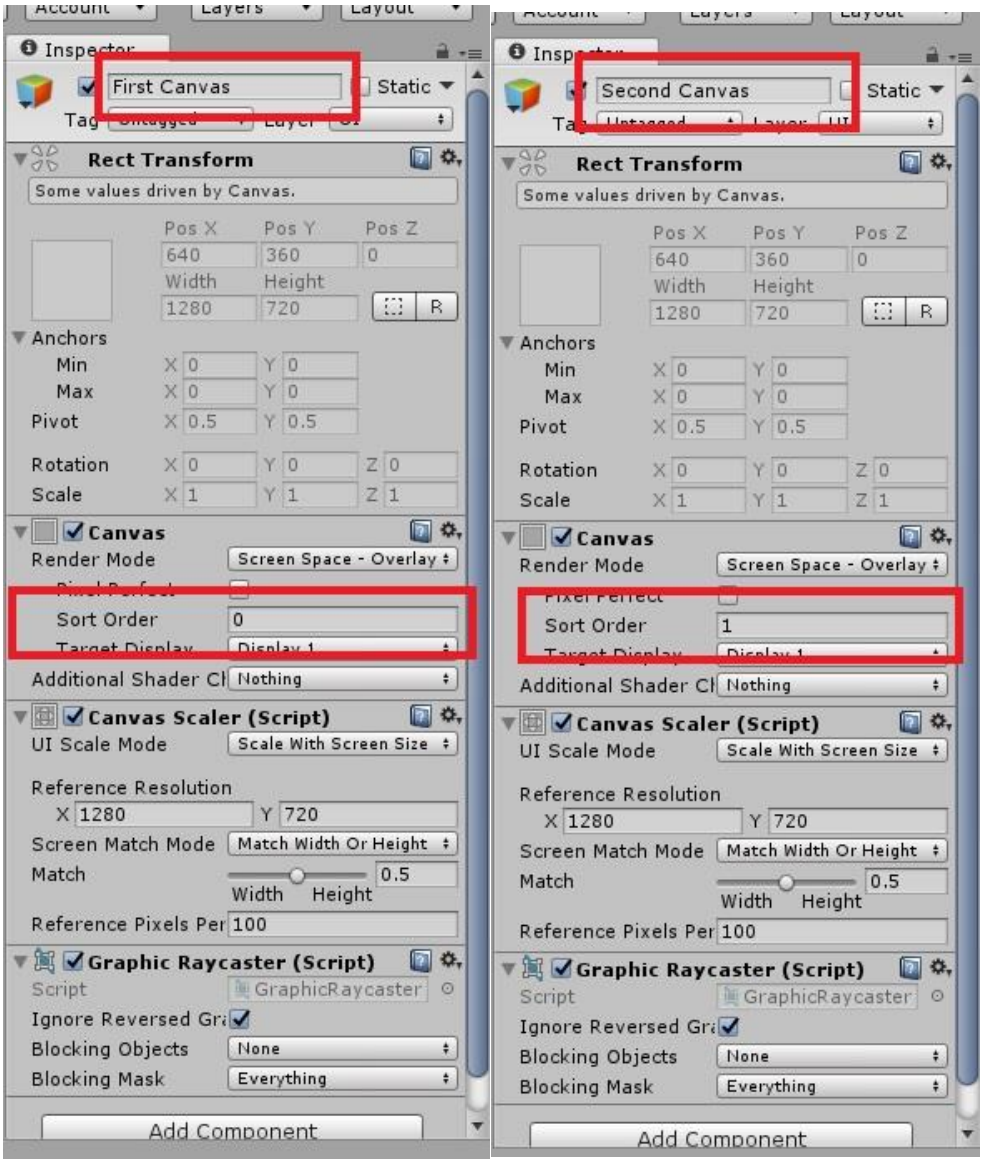
- 1- Canvas setup
- 2- Object setup
- 3- Panel setup
- 4- DDM setup
- 5- Additional tips

1- Canvas setup

Create two canvases.

- **First Canvas:** All objects & All Panels must be the children of the First Canvas and it must have less Sort Order or Order in Layer than Second Canvas.

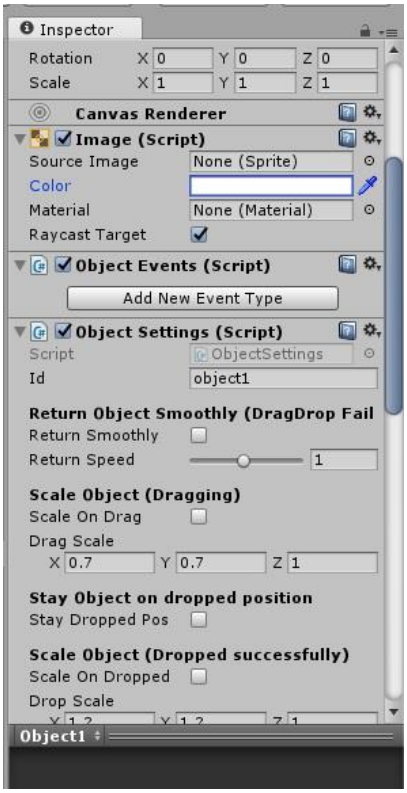
- **Second Canvas:** this canvas must not have any children and it must have more Sort Order or Order in Layer than First Canvas.



2- Object setup

We will use UI Image to create our object. You can use other UI Elements.

- Create an UI image.
- Assign ObjectSettings.cs script from Drag & Drop Pro/Scripts folder to it.



Explanations of variables:

- **Id:** Write an Id for your object by using this variable.
  Warning: Before running your game, you must assign an Id to your object.
- **User Control:** Allow user to control this object.
- **Return Smoothly:** Return Object to its first Position Smoothly When Drag & Drop Failed.
- **Scale On Drag:** Scale Object When dragging gets begun.
- **Stay Dropped Pos:** Keep Object on dropped position When it gets dropped successfully.

- **Scale On Dropped:** Scale Object When it gets dropped successfully.
- **Lock Object:** Lock Object When it gets dropped successfully.
- **Return Object:** Return Object to its first position when it gets dropped successfully.
- **Replace Smoothly:** Replace Object smoothly when it gets dropped successfully.
- **Switch Objects:** Allow to switch Objects between panels.
- **Move Smoothly:** Move Object smoothly when it is switching.
  Note: When an object (first object) gets dropped successfully on a panel, if there is another object (second object) on that panel, normally, first object will replace with second object and the second object will return to its first position.
  In addition, if you make "Switch Objects" variable true and before beginning of dragging, first object was on any panel, objects will switch between their positions (or their panels).

- **Filter Panels:** Allow using Filter Panels tool.
- **Allowed Panels:** The Ids of the panels that object is allowed to drop on them.
- **OnReplaced:** When the object gets replaced by another object.

3- Panel setup

We will use UI Image to create our panel. You can use other UI Elements.

- Create an UI image.
- Assign PanelSettings.cs script from Drag & Drop Pro/Scripts folder to it.

Explanations of variables:

- **Id:** Write an Id for your panel by using this variable.
  Warning: Before running your game, you must assign an Id to your panel.
- **Object Position:** Customize the position of the object when it gets dropped on this panel.
- **Ignore:** Objects will ignore this panel.
- **Lock Object:** Customize Object Locking (Only for this panel).
- **Object Replacement:** Allow Object to Replace & Switch or use Multi Object Mode.
  - **Multi Object Mode:** Allow to drop more than one object on this panel.

  Note: Using Object Settings means that you do not want to customize the panel.

- **OnObjectDropped:** When any object gets dropped on this panel.

- **OnObjectExit:** When the object of this panel, gets dropped on another panel.

4- DDM setup

Drag and drop DDM prefab from Drag & Drop Pro/Prefabs folder to your game scene.

Explanations of variables:

-Target Platform: Choose the target platform that you want to use this plugin on.

-All Panels: Assign all panels to this list.

-All Objects: Assign all objects to this list.

-First Canvas: Assign First Canvas to this variable.

-Second Canvas: Assign Second Canvas to this variable.

-Save States: If you want to use save system, make this variable true.

Note: Save system is implemented using PlayerPrefs.

-Dragging Modes:

Change To Mouse Pos: When object begin dragging, its position will change to the current position of the mouse or touch.

-BeforeSetup & AfterSetup Events: If you have any codes which is related to the Drag & Drop System and you want to run them at the beginning of the scene, please use these Event Managements. Your codes will be called before or after the setting up of the DDM.

AI DragDrop:

By using this component, you can simulate the drag and drop system.

You just need to use the following command:

```
string ObjectId = "object1";

string PanelId = "panel1";

AIDragDrop.DragDrop (ObjectId, PanelId);
```

If you want an instant movement, call the method in this form:

```
AIDragDrop.DragDrop (ObjectId, PanelId, true);
```

Above codes will drag and drop `object1` onto `panel1.`

If you do not want to use AI system, you can remove this component.

5- Additional tips

- Default panel: If you want your object to be on a panel at the beginning, before running the game, change the position of your object to the position of the target panel like that your object is dropped on it.

-Panel Object detection: There are some methods that can help you to know which one of objects are dropped on which one of panels. Use the command below to get the Id of the object that has been dropped on the target panel:

```
string PanelId = "panel1";

string Panel1Object = DragDropManager.GetPanelObjectId
(PanelId);
```

You can check the value of the **Panel1Object variable** by using a **switch** command like this:

```
switch (Panel1Object) {
     case "object1":
          // do something
          break;
     case "object2":
          // do something
          break;
}
```

The method below returns the Id of the panel that target object has been dropped on:

```
string ObjectId = "object1";
string Object1Panel = DragDropManager.GetObjectPanelId (ObjectId);
```

You can check the value of the **Object1Panel variable** by using a **switch** command like this:

```
switch (Object1Panel) {
     case "panel1":
          // do something
          break;
     case "panel2":
          // do something
          break;
}
```

If you are using **Multi Object Mode**, you can get the Ids of the objects that have been dropped on the target panel by using the following command:

```
string PanelId = "panel1";

string[] Panel1Objects =
DragDropManager.GetPanelObjectsIds (PanelId);
```

The above code will put the Ids in **`Panel1Objects`** array.

-Add/Remove objects at runtime: To **add** an object dynamically, Give a variable of ObjectSettings type to the method below:

```
DragDropManager.AddObject (objectSettings);
```

To **remove** an object dynamically, Give a variable of ObjectSettings type to the method below:

```
DragDropManager.RemoveObject (objectSettings);
```

-Getting ObjectSettings by the id of the object: The following method returns the ObjectSettings of an object whose Id is known:

```
string ObjectId = "object1";

ObjectSettings obj = DragDropManager.GetObjectById
(ObjectId);
```

-Getting PanelSettings by the id of the panel: The method below returns the PanelSettings of a panel whose Id is known:

```
string PanelId = "panel1";

PanelSettings pnl = DragDropManager.GetPanelById
(PanelId);
```

-Reset Drag and Drop states: To reset drag and drop states of the scene to its initial status, just Call this method:

```
DragDropManager.ResetScene ();
```
