

# Drag & Drop Pro

Version 1.84



Contact me at [dragdrop@protonmail.com](mailto:dragdrop@protonmail.com)

## Documentation

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

### Table of Contents

[1- Canvas setup](#)

[2- Object setup](#)

[3- Panel setup](#)

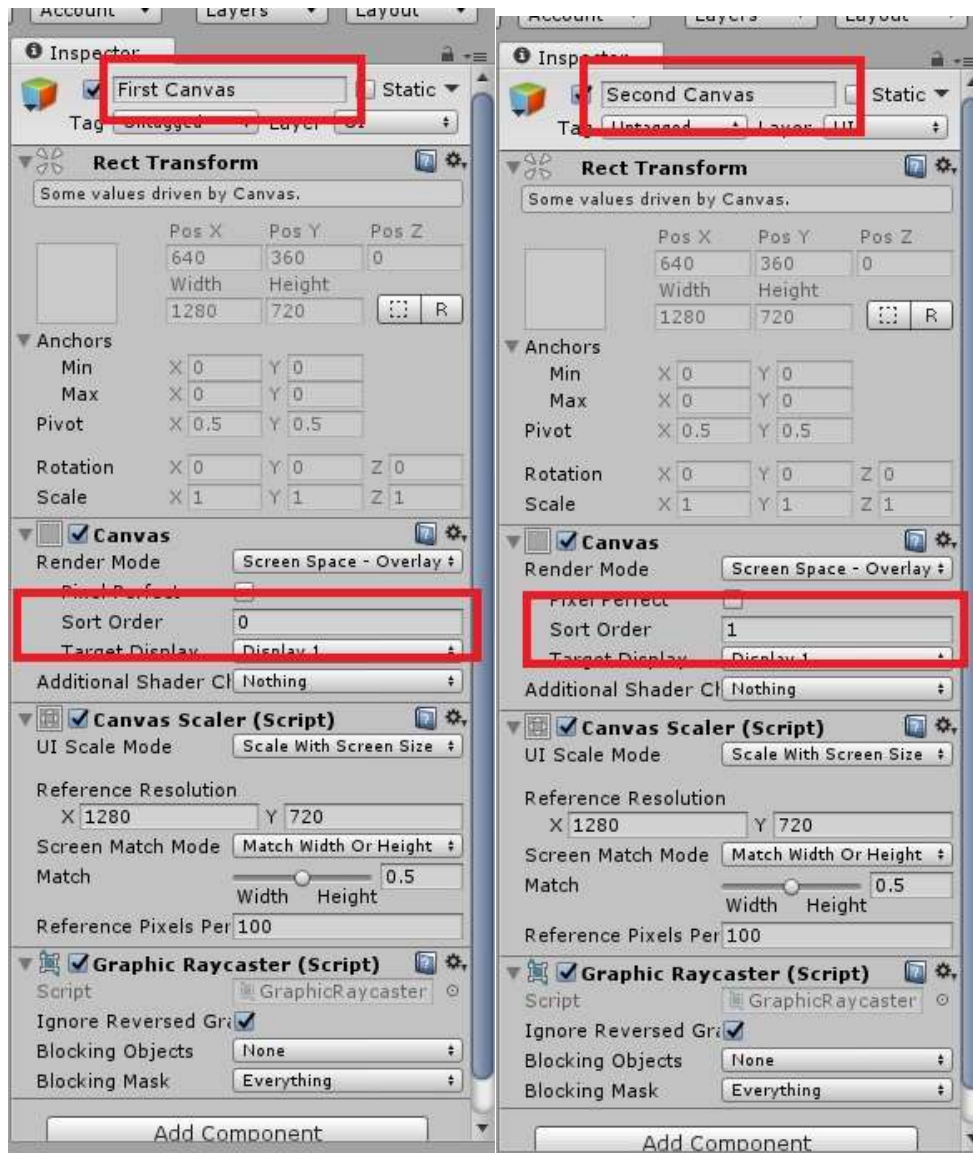
[4- DDM setup](#)

[5- Additional tips](#)

# 1- Canvas setup

Create two canvases.

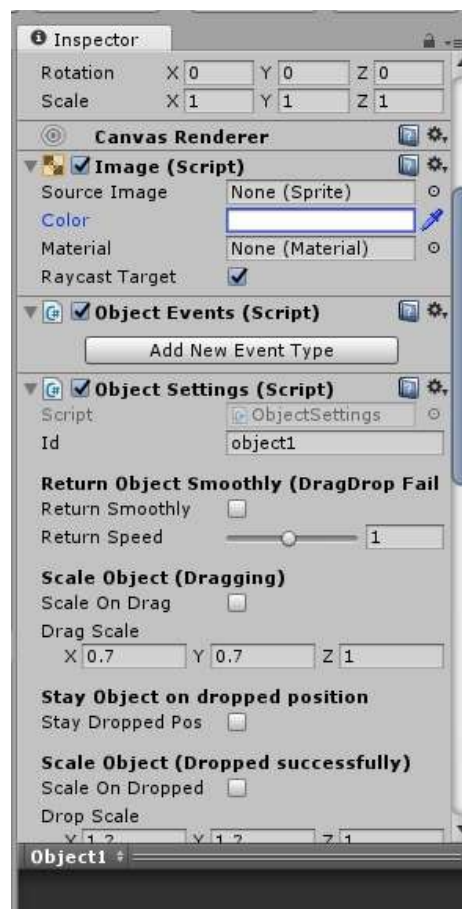
- **First Canvas:** All objects & All Panels should be the children of the First Canvas and it should have less **Sort Order** or **Order in Layer** than Second Canvas.
- **Second Canvas:** this canvas should not have any child and it should have the more **Sort Order** or **Order in Layer** than First Canvas.



## 2- Object setup

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

- Create an UI image
- Assign ObjectEvents.cs and ObjectSettings.cs scripts 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.
- **AI Control:** Allow AI to control this object.

- **Return Smoothly:** Return Object to its first Position Smoothly When Drag & Drop Failed.
- **Scale On Drag:** Scale Object When dragging was begun.
- **Stay Dropped Pos:** Stay Object on dropped position When the Object dropped successfully.
- **Scale On Dropped:** Scale Object When dropped successfully.
- **Lock Object:** Lock Object When dropped successfully.
- **Return Object:** Return Object to its first position when dropped successfully.
- **Replace Smoothly:** Replace Object smoothly when 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) dropped successfully on a panel, if there is another object (second object) on that panel, normally first object will replace with second object and second object will return to its first position.

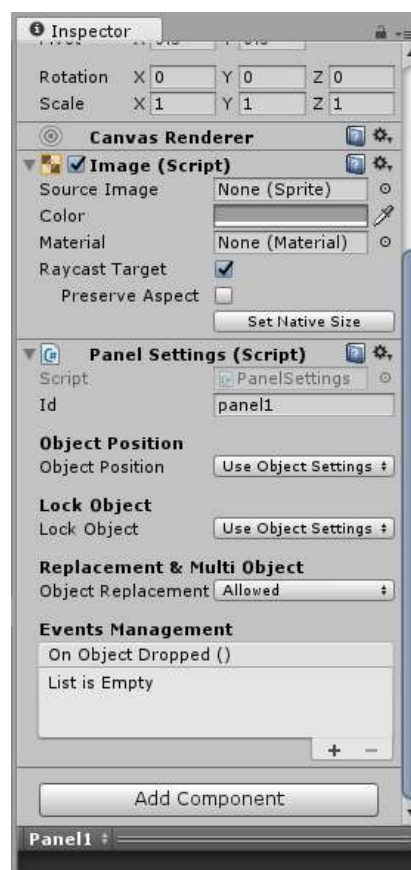
In addition, if you make "Switch Objects" variable **true** and before begin dragging, first object there was on any panel, objects will switch between their positions (or their panels).

- **Filter Panels:** Allow using Filter Panels tool.
- **Allowed Panels:** The Ids of the panels that object is allowed to drop on them.

### 3- Panel setup

We will use **UI Image** to create our panel. You can use another 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 dropped on this panel.
- **Lock Object:** Customize Object Locking (Only for this panel).
- **Object Replacement:** Allow Object to Replace & Switch or use Multi Object Mode.
  - o **Multi Object Mode:** Allow to drop more than one object on this panel.

**Note: Using Object Settings means that you will not to customize the panel.**

- **OnObjectDropped:** When any object dropped on the panel, the functions that you added to this event trigger will be called.

## 4- DDM setup

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

### Explanations of variables:

- Target Platform:** Choose your target platform that you want to use this plugin on its.
- All Panels:** Assign all panels to this array.

- All Objects:** Assign all objects to this array.
- 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 by using [PlayerPrefs](#).

#### **-Dragging Modes:**

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

## **AI Manager:**

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";
```

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

The above code will drag and drop object1 onto panel1.

**If you will not to use AI system, you can remove this component.**

## 5- Additional tips

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



**-Panel Object detection:** To know which one of objects dropped on which one of panels, you can use this command to get the Id of the object that dropped on target panel:

```
string PanelId = "panel1";  
  
string Panel1Object = DragDropManager.GetPanelObject  
(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;  
}
```



In addition, you can use this command to get the Id of the panel that target object dropped on it:

```
string ObjectId = "object1";  
  
string Object1Panel = DragDropManager.GetObjectPanel  
(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 dropped on target panel by using this command:

```
string PanelId = "panel1";  
  
string[] Panel10bjects = DragDropManager.GetPanel0bjects  
(PanelId);
```

The above code will put the Ids in Panel10bjects array.

**-Reset Drag and Drop states:** To reset drag and drop states to the initial status, you just need to call this method:

```
DragDropManager.Reset ();
```