

# Draw Lines Package

V 1.1.3

Indie Games Studio

[www.assetstore.unity3d.com/en/#!/publisher/9268](http://www.assetstore.unity3d.com/en/#!/publisher/9268)

[freelance.art2014@gmail.com](mailto:freelance.art2014@gmail.com)

## Table of Contents

1.0	- Overview.....	1
2.0	- Project Scenes.....	1
3.0	- The Grid.....	6
4.0	- Create Mission.....	7
5.0	- Create Level.....	9
6.0	- Create Pair of Elements.....	10
7.0	- Awesome & Confirm Dialog.....	11
8.0	- Hot Keys.....	12

## 1.0 - Overview

---

Draw Lines is a game where you have to connect every pair of elements of the same color or family without intersection between lines.

Connect the matching elements to create a line. Pair all elements, and cover the entire board to solve each puzzle. Lines will break if they cross.

## 2.0 - Project Scenes

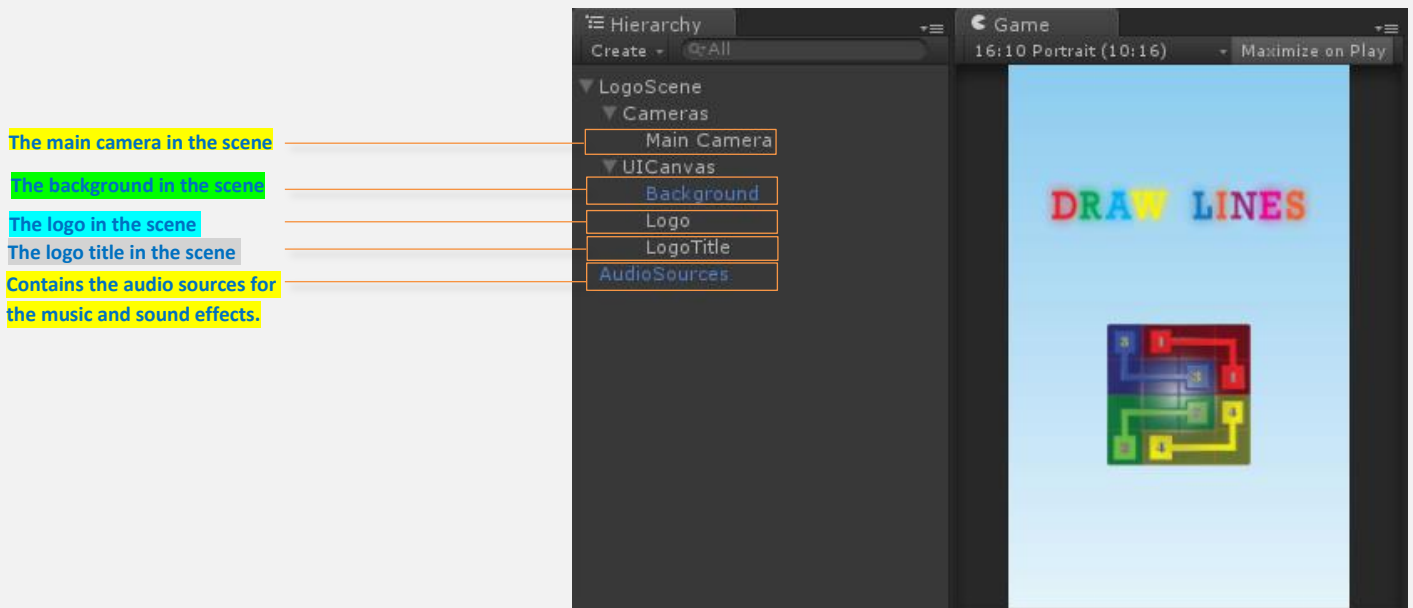
---

The project is divided into seven scenes:

1. Logo Scene
2. Main Scene
3. How to Play Scene
4. Options Scene
5. Missions Scene
6. Levels Scene
7. Game Scene

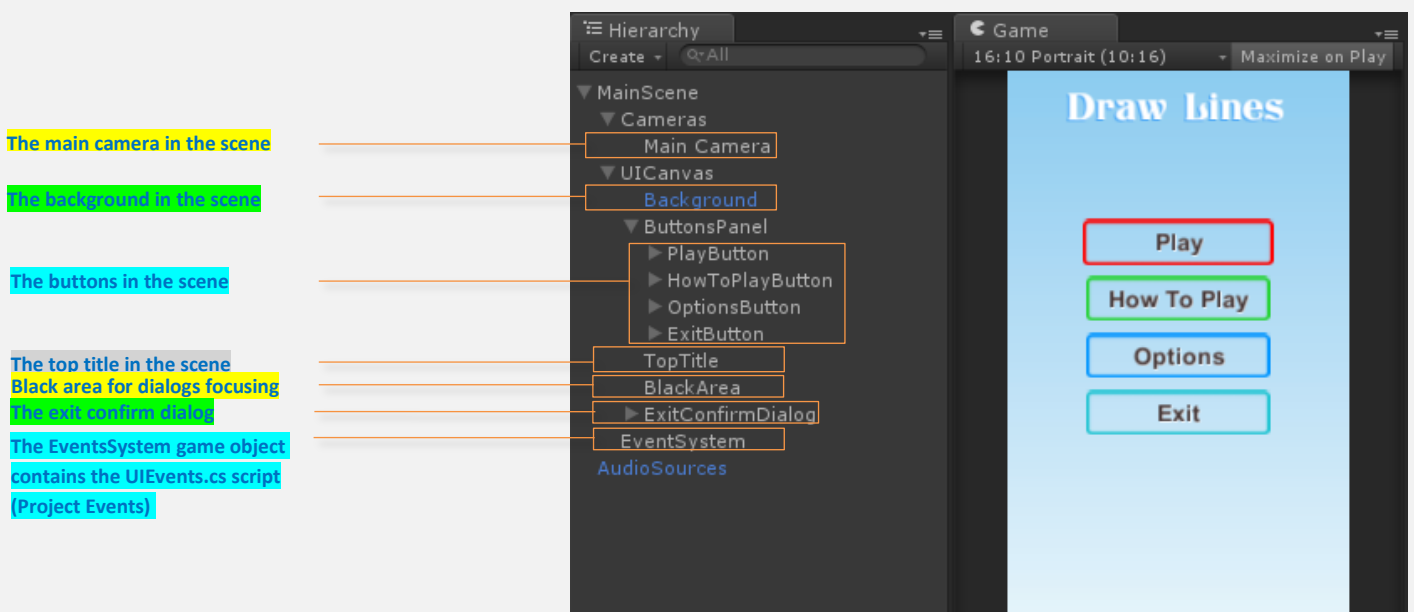
## - Logo Scene

Introduce your game here by adding the logo, title ... etc.



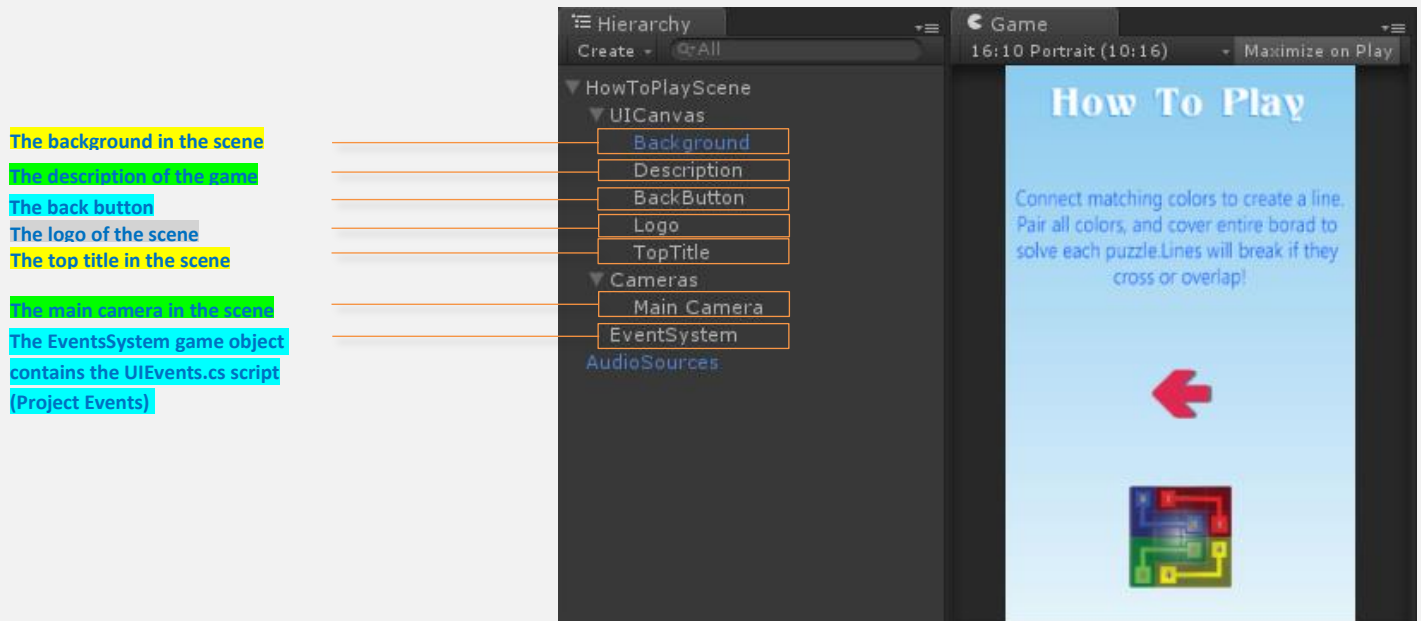
## - Main Scene

The main scene is the entry point to the game which includes three references to the other scenes such as HowToPlay, Options, Missions scenes and the Exit button.



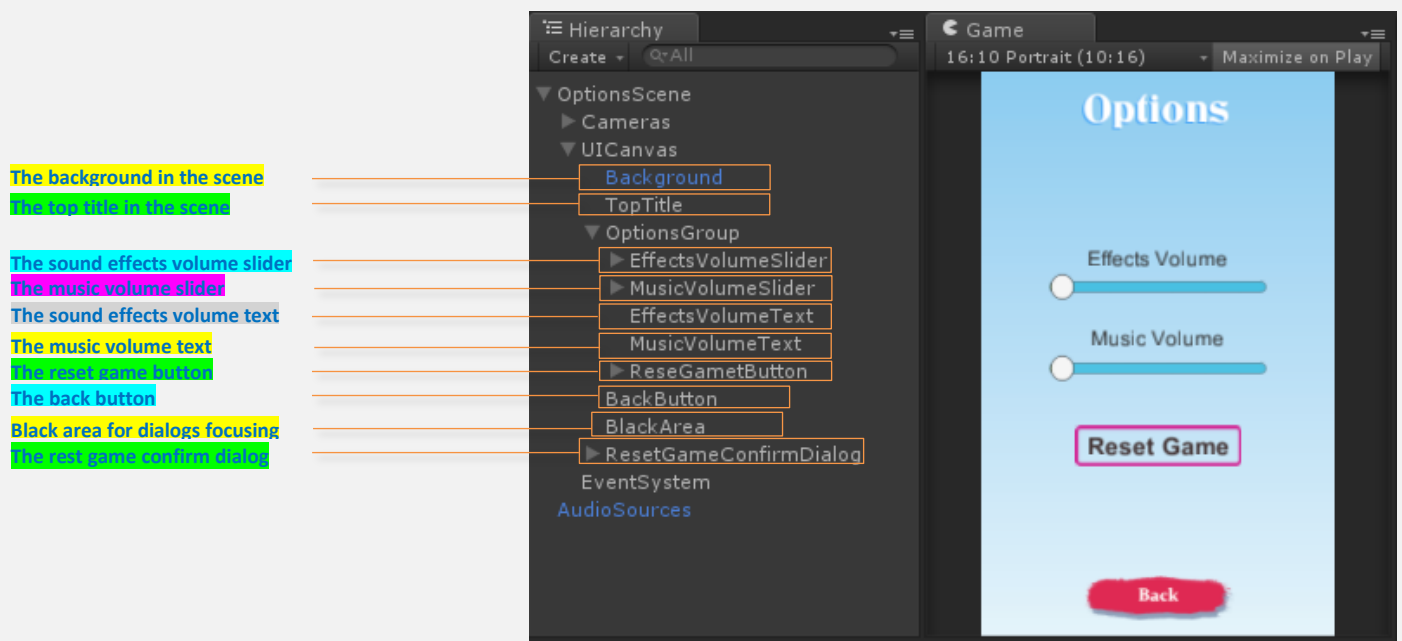
## - How to Play Scene

This scene is the guide of how to play the game.



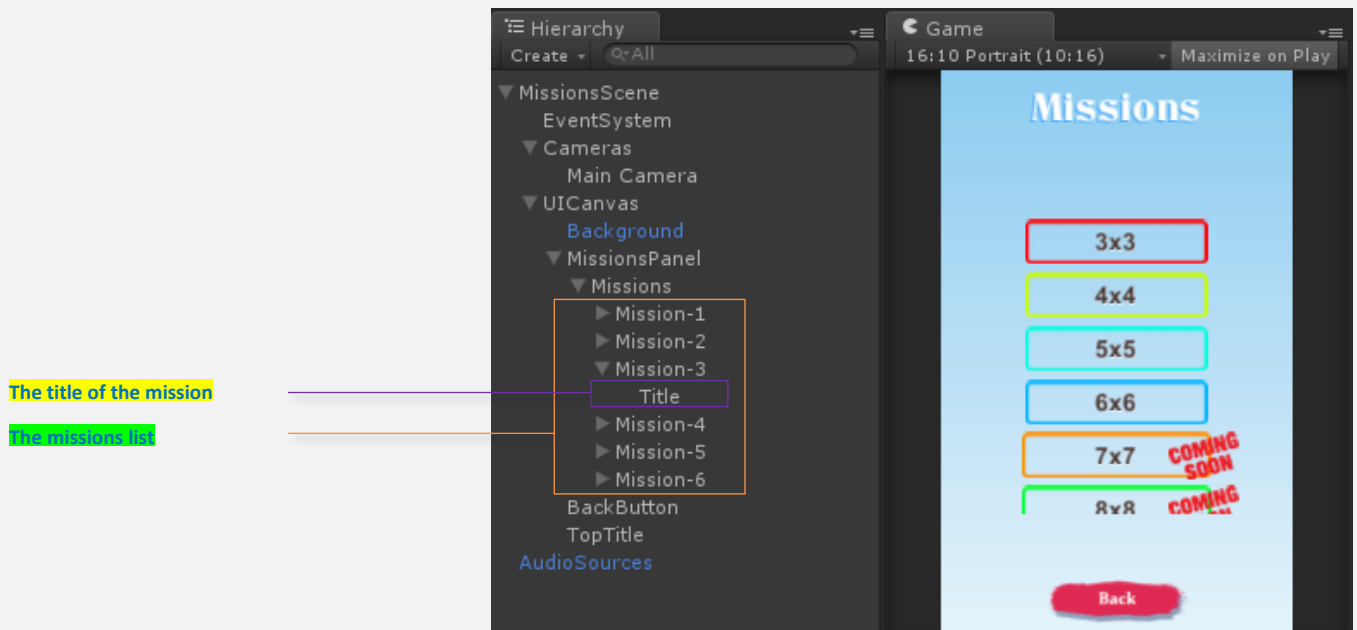
## - Options Scene

The Options contains the settings of your game such as effects/music volume, reset game option ...etc.



## - Missions Scene

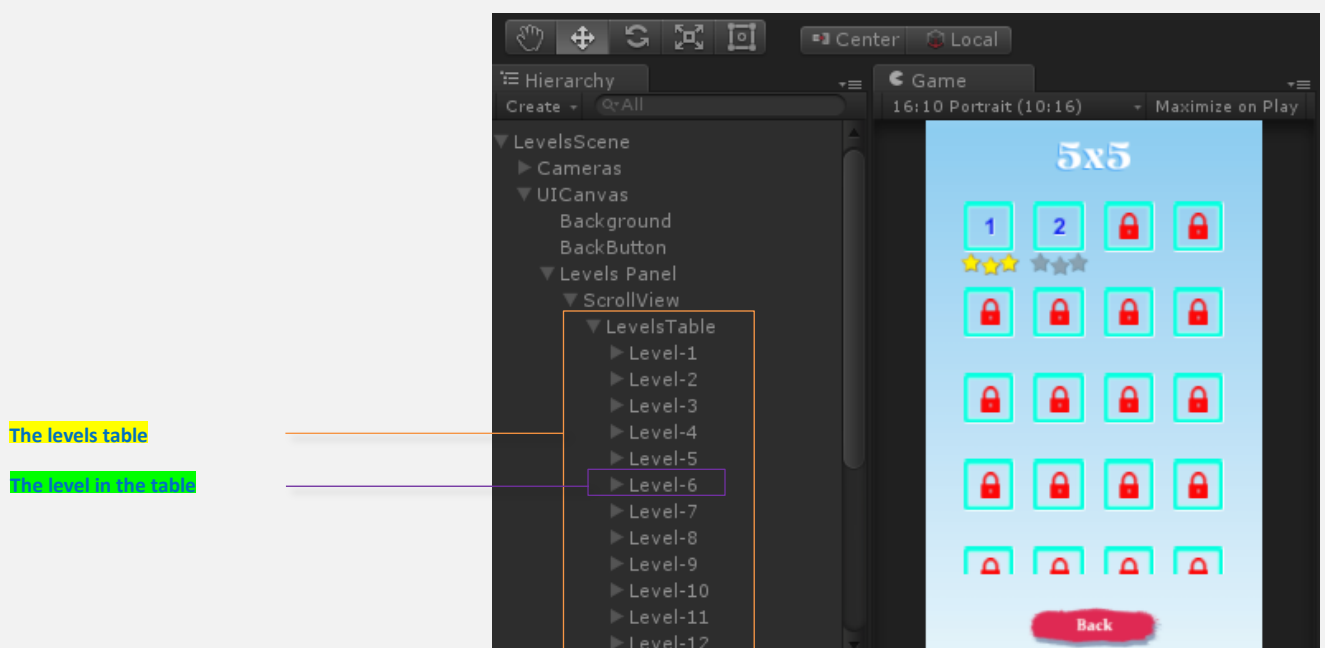
The Missions scene contains the missions of the game. Each mission is a child for the Missions GameObject.



## - Levels Scene

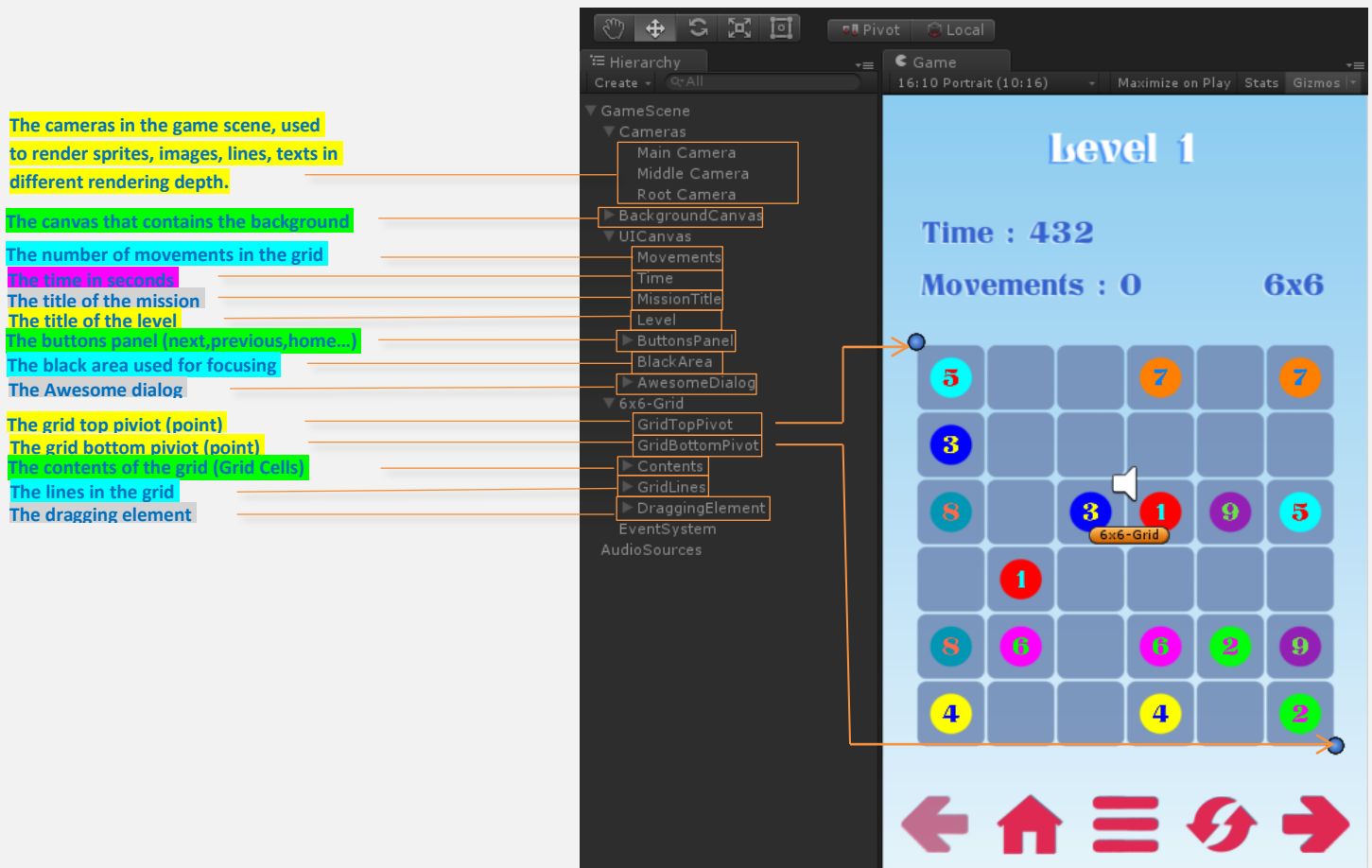
The levels scene contains the levels of the selected mission which are sorted in a table.

The levels are automatically generated using the LevelsTable script.



## - Game Scene

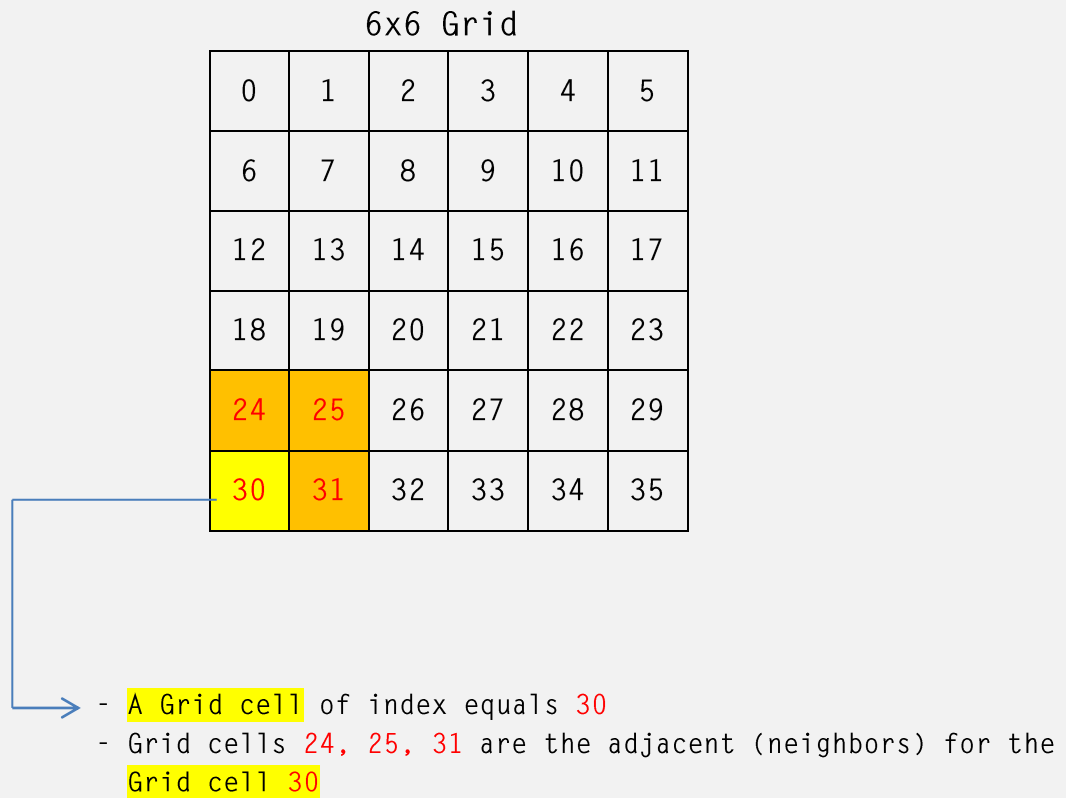
The game scene contains the main components for the game such as the grid of size equals number of columns \* number of rows.



## 3.0 - The Grid

---

The following figure is an example for a grid of size equals 6x6.

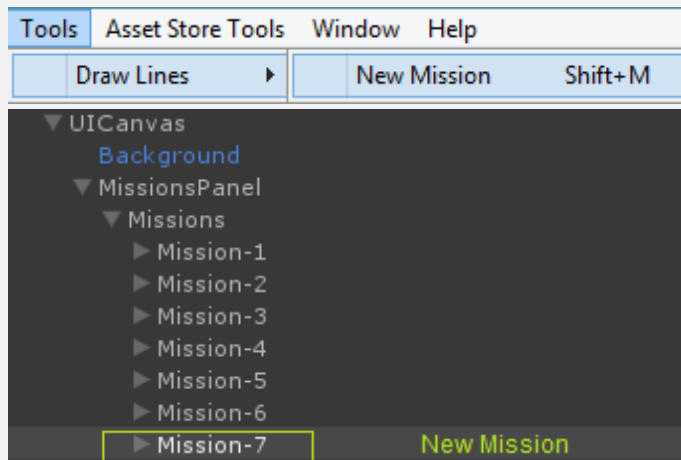




## 4.0 - Create Mission

---

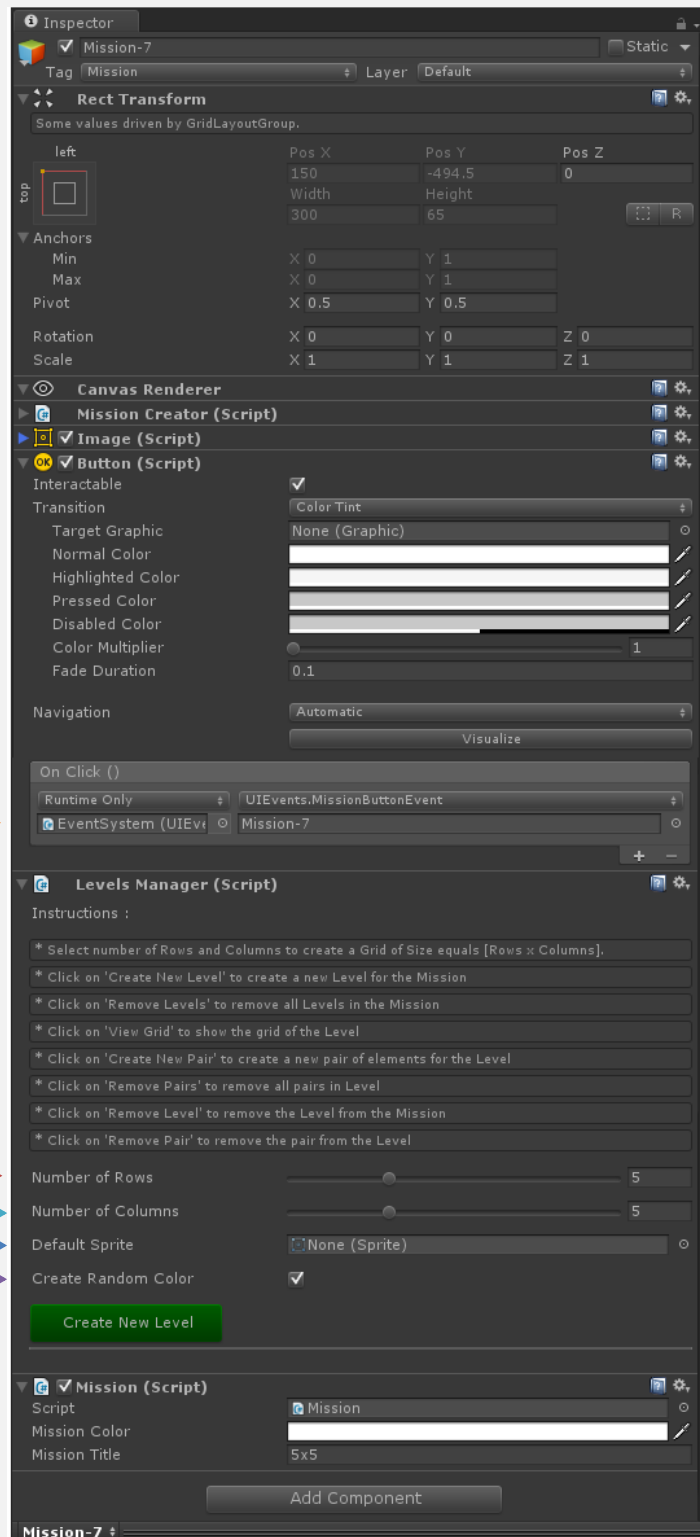
To create new mission select **Tools** -> **Draw Lines** -> **New Mission**.  
Check the following figures:



Congratulation, the new mission will be added automatically to the Missions GameObject with an **auto-generated name**.  
You will find the following components added to the new mission GameObject:

1. RectTransform
2. MissionCreator
3. CanvasRenderer
4. Button
5. Image
6. LevelsManager
7. Mission

Now edit the title and color of the new Mission using **Mission** component as you like, add the Mission button event (UIEvents.MissionButtonEvent, pass parameter: Mission GameObject) using **Button** component, create your Levels using **LevelsManager** component, customize mission as you like.

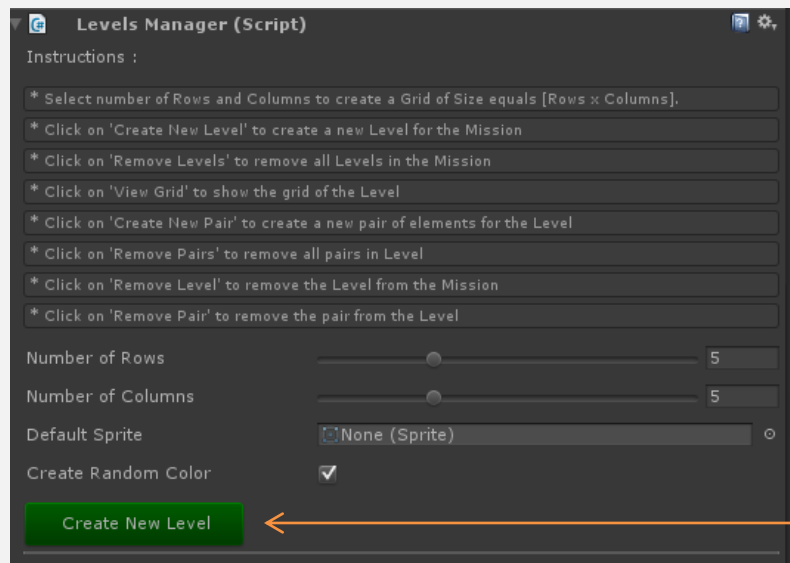


The Mission button event

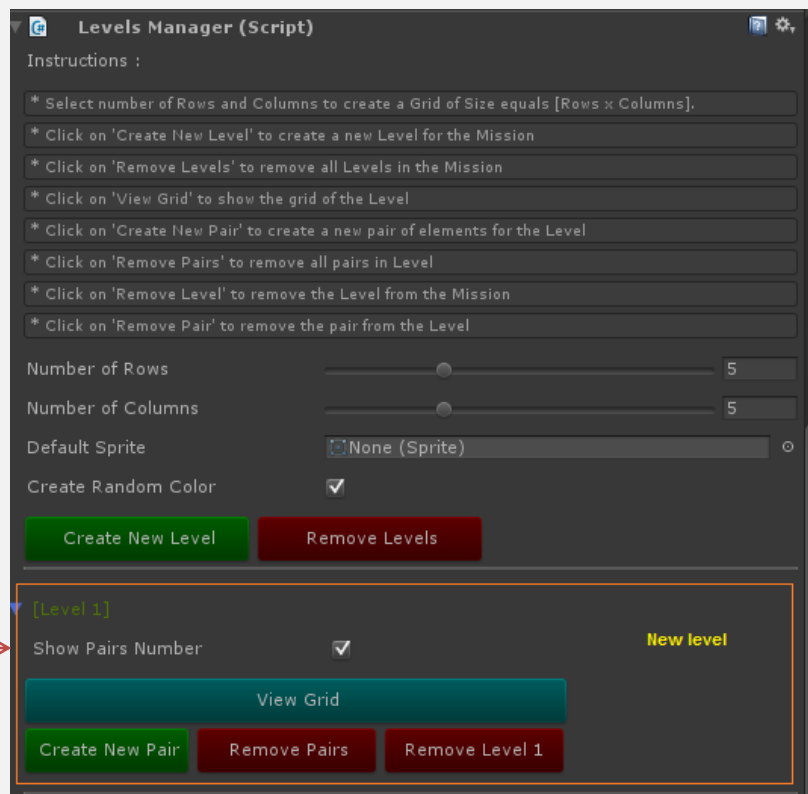
- \* Number of Rows: the number of rows in the grid.
- \* Number of Columns: the number of columns in the grid.
- \* Default sprite: the default texture for the pair of elements (Objects).
- \* Create Random Color: whether to create a random color for the sprite of the pair of elements and the line.

## 5.0 - Create Level

To create a new level click on **Create new Level** button



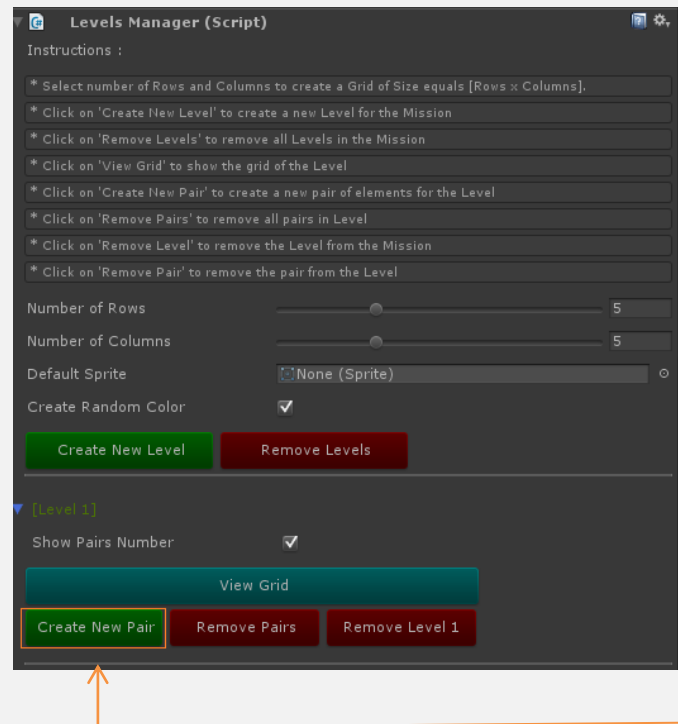
Congratulation, new level is created as the following figure:



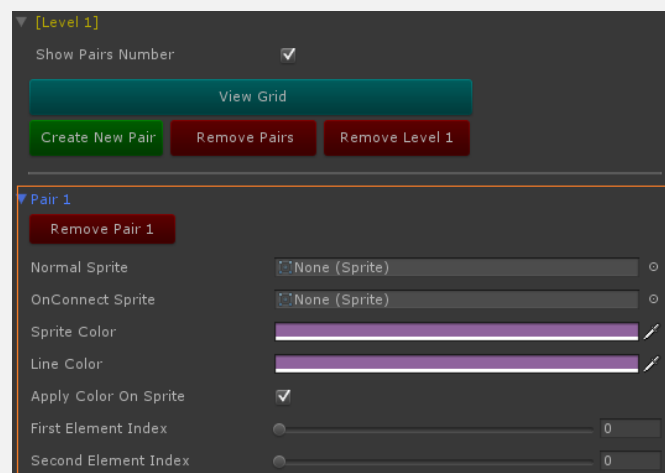
\* **Show Pairs Number:** Whether to show a number for the pairs in the grid.

## 6.0 - Create Pair of Elements

To create a new pair of elements click on **Create New Pair** button.



Congratulation, new pair elements is created as the following figure:



- \* **Normal Sprite:** the normal sprite of the pair of elements.
- \* **OnConnect Sprite:** the sprite of the pair of elements on line connected.
- \* **Sprite Color:** the color of the sprite of the pair of elements.
- \* **Line Color:** the color of the line between the pair of the elements.
- \* **Apply Color on Color:** whether to apply the sprite color.
- \* **First Element Index:** the index of the first element in the grid.
- \* **Second Element Index:** the index of the second element in the grid.

## 7.0 - Awesome & Confirm Dialog

---

### - Show Dialog

To show the Awesome/Confirm Dialog, you need to call the **Show()** Method which is in the AwesomeDialog/ConfirmDialog script.

Example: `GameObject.FindObjectOfType<AwesomeDialog> ().Show ();`

### - Hide Dialog

To hide the Awesome/Confirm Dialog, you need to call the **Hide()** Method which is in the Awesome Dialog/ConfirmDialog script.

Example: `GameObject.FindObjectOfType< AwesomeDialog > ().Hide ();`

## 8.0 - Hot Keys

---

<u>Press this</u>	<u>To do this</u>
<b>Shift + M</b>	Create new Mission