

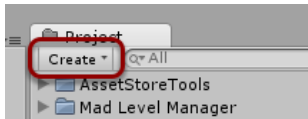
## Your First Level Select Screen

This tutorial is also available on YouTube: <http://youtu.be/GsMoQMWp23w>

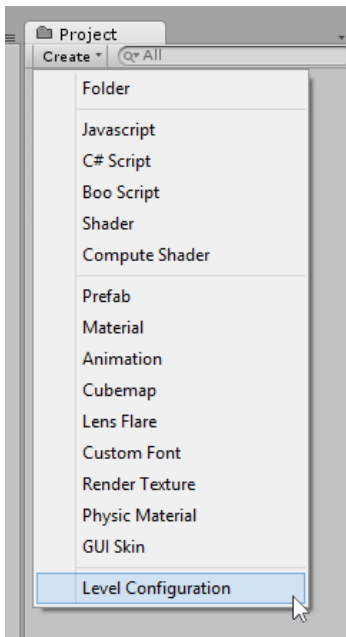
Welcome to the first Mad Level Manager tutorial. It explains how to create your first level select screen with the most basic configuration possible. This is the only small part of what Mad Level Manager can actually do. If you want to know more, please read the [overview](#) page or look at the other documentation pages.

## Creating Level Configuration

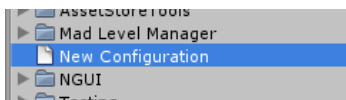
First thing you need to do is to create **Level Configuration**. This will be the work flow of your game. It will determine both the selection of the scenes which will be loaded first and the order of your playable levels. To create one, click on the *Create* button in your *Project* view...



...and then select the **Level Configuration** option.



The **New Configuration** asset will be created.



Let's rename it to "My First Level Configuration".

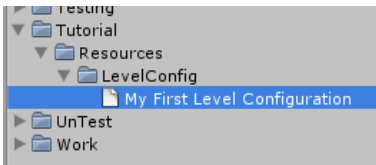


Now look at your *Inspector* view. You should see that there's one important error that should be immediately fixed.



This message means that the **Level Configuration** asset that we've created just now must be located under the declared directory. That's because Unity3D [treats differently](#) everything that lies in the *Resources* directory.

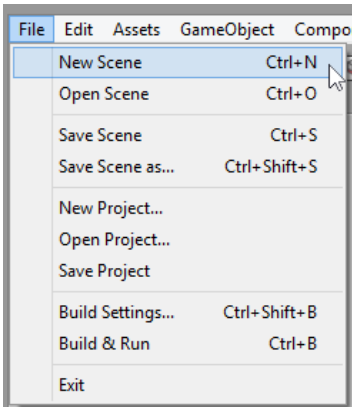
This is a tutorial, so let's make the directory structure look like this:



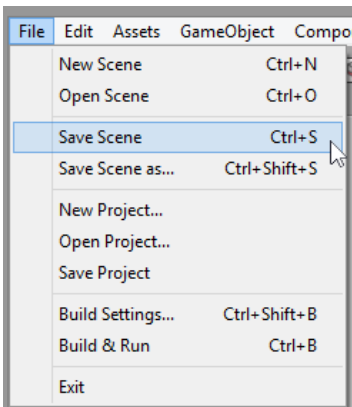
If you're still seeing this error after creating the directory structure, please make sure that there's no typo in your directory structure. Letter case matters!

## Creating Level Select Scene

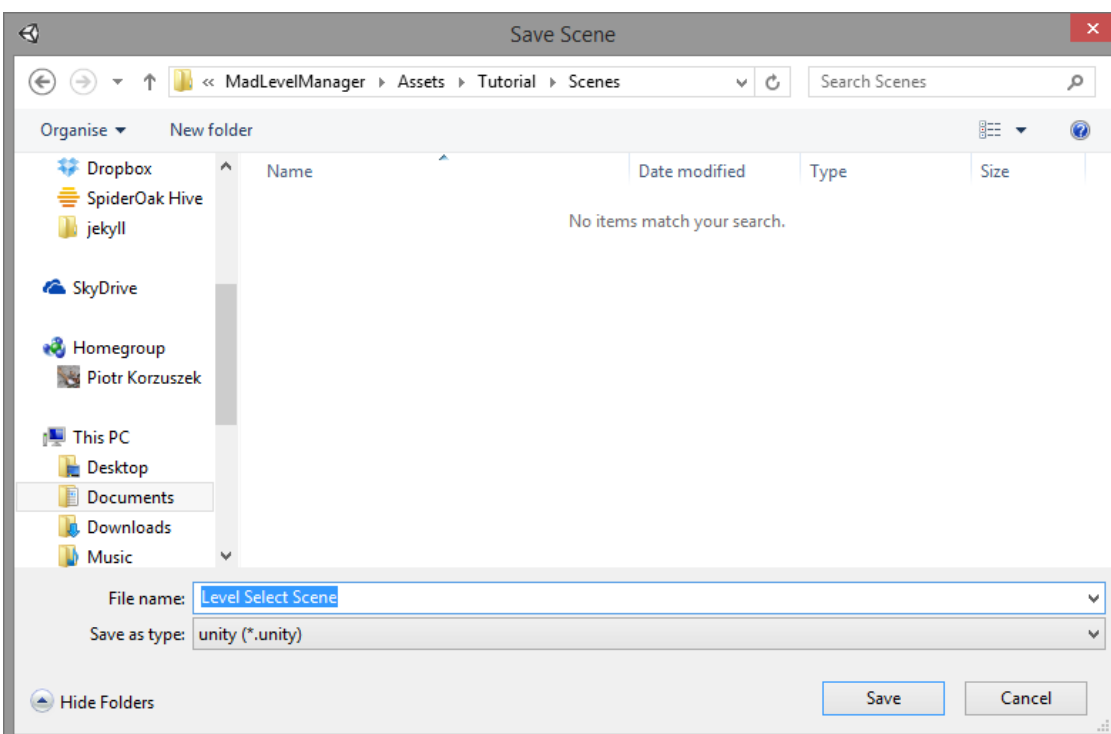
Before continuing with your levels configuration, we need to create and save a scene that will be used to select playable levels. Let's create it now:



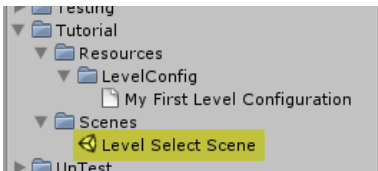
And save it without any modifications...



...in new *Tutorial/Scenes* directory as **Level Select Scene**.

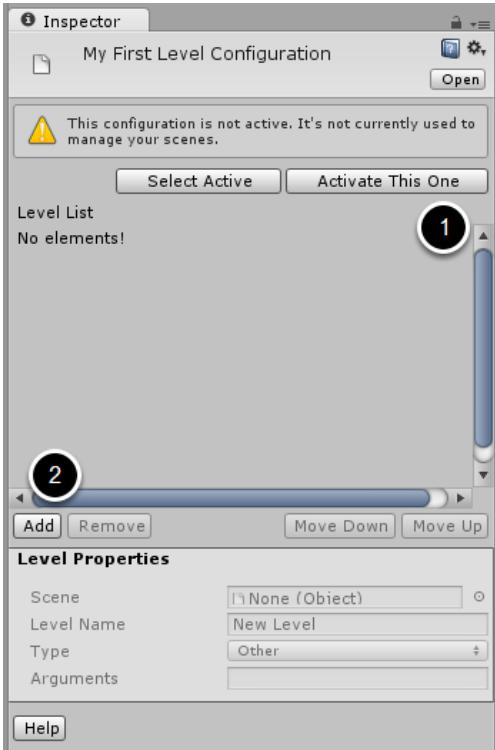


Now your *Project* should look like this:



## Configuring Levels

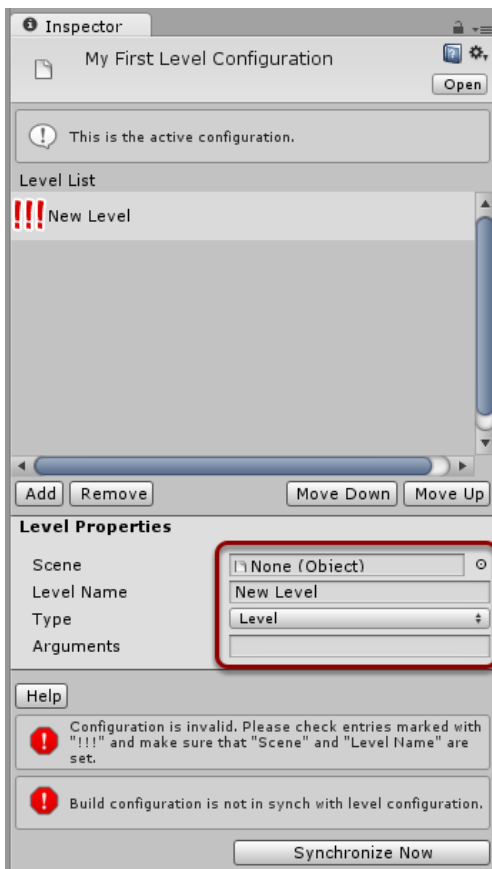
Make sure that you have *My First Level Configuration* selected and take a look at the *Inspector*.



On the top you can see that this newly created configuration is not currently active. This means that it won't be used to manage your level work flow at the current state. Let's make it active by clicking **Activate This One** (1).

You can have multiple level configurations in your project but only one should be active at the moment. Creating multiple configurations may help you to build demo and full versions of your game.

Let's add our first level by clicking on the **Add** button. The new level will appear:



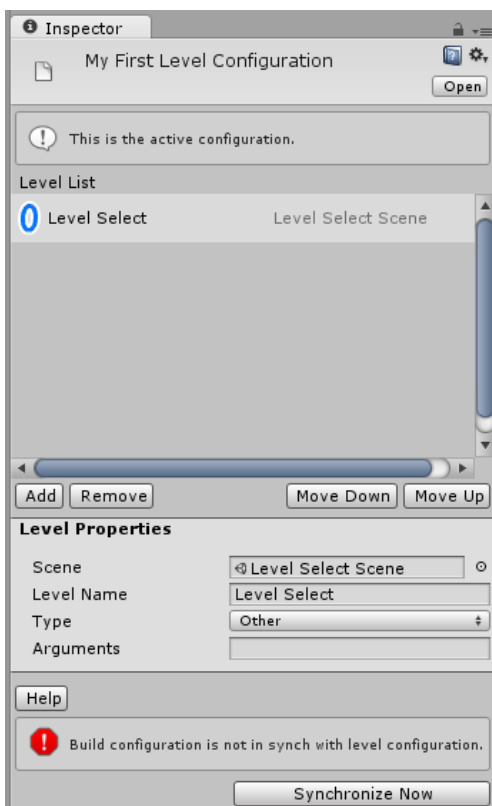
See three exclamation marks next to **New Level** on the list? This means that this level configuration is invalid and your attention is needed. It is a common thing because we've just created this level hence it needs some initial setup.

What we need to configure for the first level are these fields:

- **Scene** – here you should place your **Level Select Scene** that we've created earlier. This is a place for a scene object. Just drag & drop your scene here.
- **Level Name** – name your level "Level Select", so you can easily refer to it later.
- **Type** – set the type of your level. All levels that are not actual game levels should be set to **Other**.
- **Arguments** – if you want to pass arguments to your level (and read them later), you can do it here. We will leave this one empty.

Arguments can be read anytime by accessing *MadLevel.arguments*. This property will always contain the string set in the level configuration.

Your configuration should now look like this:



Have you noticed that some errors are gone? Don't worry about the synchronization error. This means that something in the level configuration has changed and the synchronization procedure is necessary to make everything work as expected. Unless we don't finish our configuration we won't need to synchronize it, because it will simply pop-up again.

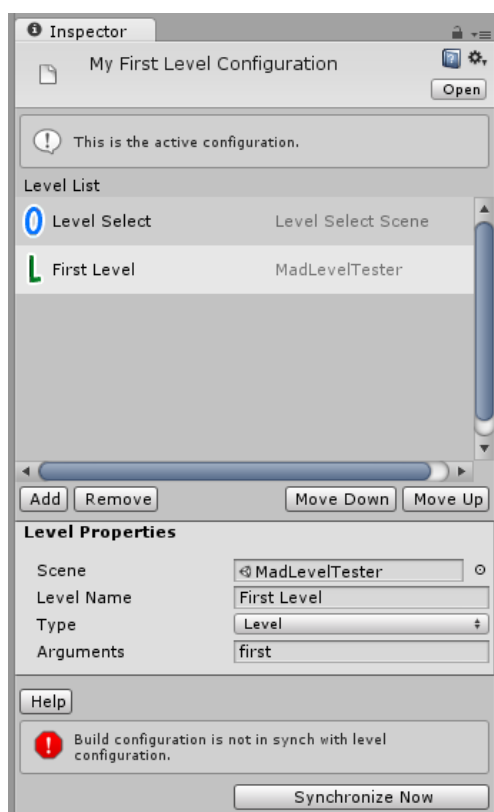
Now we will create two playable levels. As a scene we will use *MadLevelTester* that is placed in *Mad Level Manager/Examples/Scenes*. This scene imitates a playable level and allows you to proceed to the next one and return to level select screen.

*MadLevelTester* allows you to return to level select screen only if this level name is equal to *Level Select*. If you name your level select screen differently and then want to use the tester scene, you will simply receive an error about missing *Level Select* level.



Create the second level just like you've created the first one by pressing the **Add** button, and then set values to:

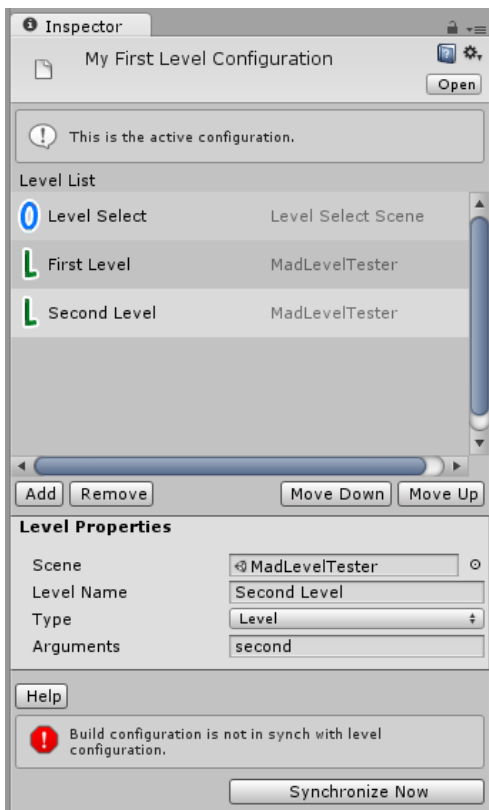
- **Scene** – *MadLevelTester*
- **Level Name** – First Level
- **Type** – Level
- **Arguments** – first



In a similar manner, create the third level that will be the second playable level with values:

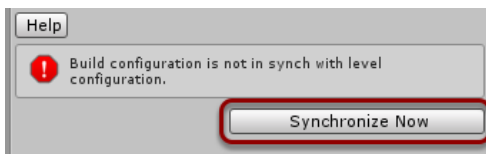
- **Scene** – *MadLevelTester*
- **Level Name** – Second Level
- **Type** – Level
- **Arguments** – second

We're setting arguments here just to make a proof that it can be read. *MadLevelTester* will display the read arguments on its screen.

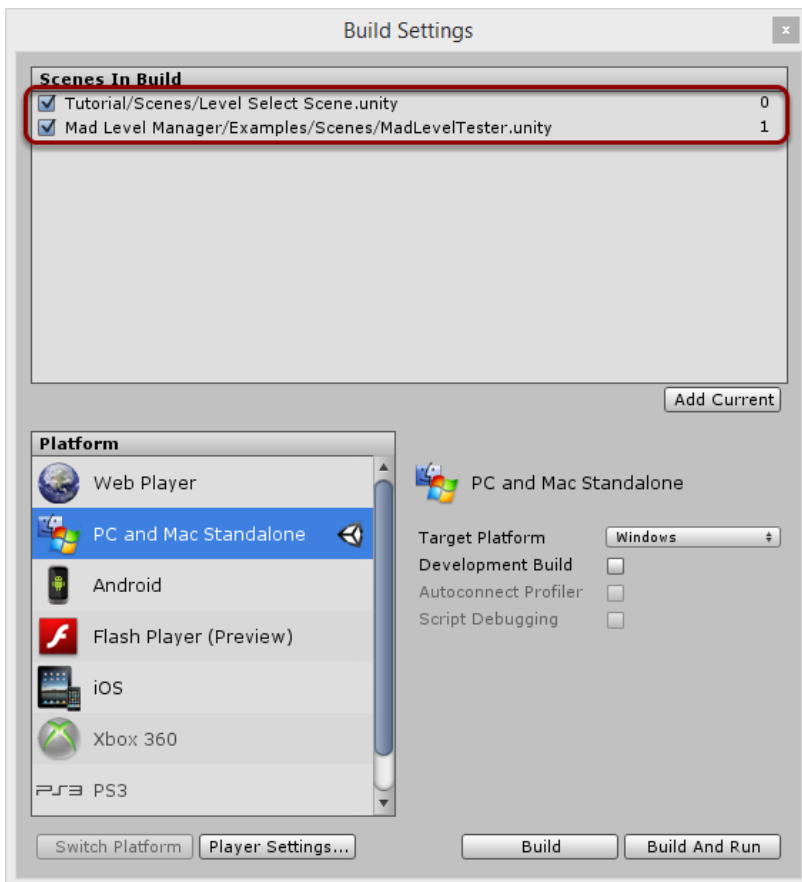


Note that *First Level* and *Second Level* are using the same scene. This is a common thing when working with Mad Level Manager. You can use one scene to represent many levels, and use an argument line to pass information about which data file should be loaded. Of course this is only up to you (or programmer in your team).

Now when you're done with configuring the levels, it's time to synchronize our build configuration. Click the **Synchronize Now** button:

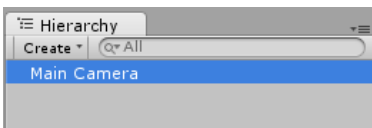


You can take a peek at what it did to your build configuration (*File* → *Build Settings* or *Ctrl + Shift + B*). **Level Select Scene** will be launched first, and *MadLevelTester* is placed as a second scene in the build.

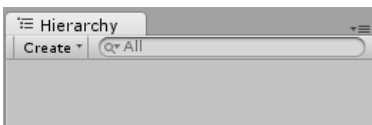


## Configuring Level Select Screen

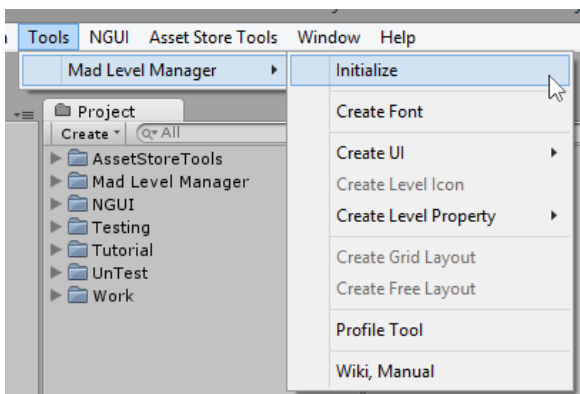
Be sure to have our **Level Select Scene** open. We need to make it completely clean, hence locate the *Main Camera* in the *Hierarchy*...



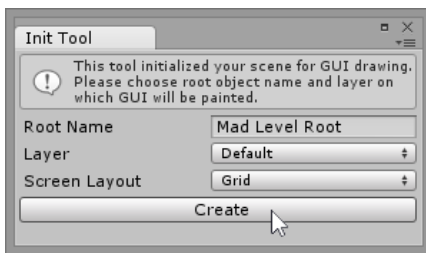
...and delete it.



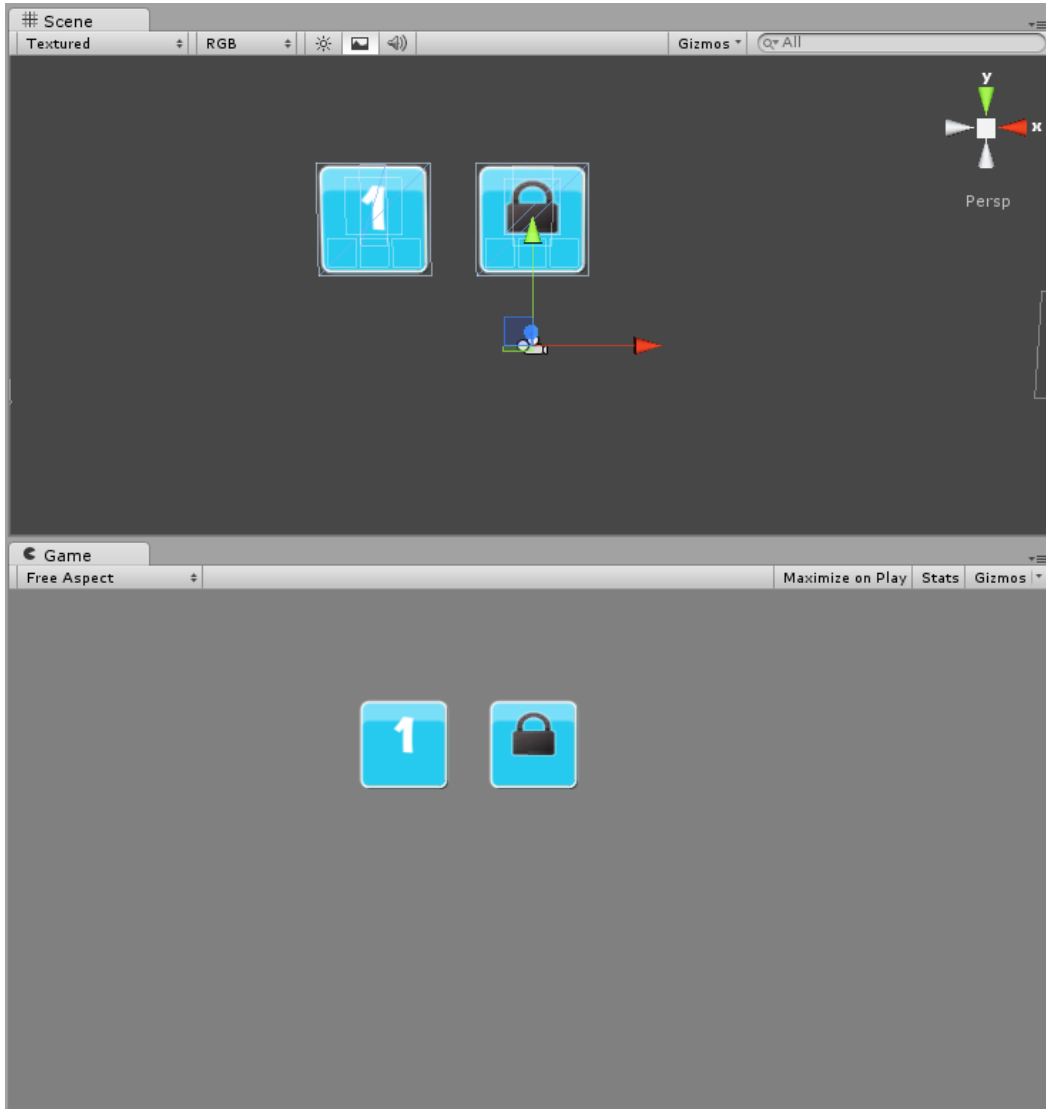
Next, from the main menu select *Tool* → *Mad Level Manager* → *Initialize*.



The **Init Tool** will open:



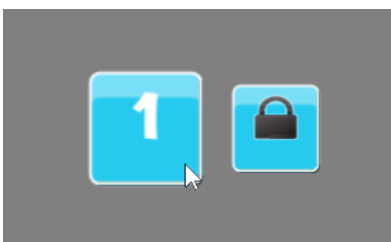
Here you can setup a few things but we won't change anything since we want to create the **Grid** layout. Click on the **Create** button to proceed, and your grid should appear on the screen:



And that's it! Now we can test if everything is working as expected! Hit the *Play* button:



Then select the first level in your *Game* view.



The **First Level** with the argument **first** will be loaded.



**Level Name: 'First Level'**

**Arguments: first**

**Choose number of points:**

A red circular button with the text "50 points" in white.

**50 points**

A green circular button with one yellow star at the top and the text "100 points" in white.

**100 points**

A green circular button with two yellow stars at the top and the text "150 points" in white.

**150 points**

A blue circular button with three yellow stars at the top and the text "200 points" in white.

**200 points**

**Back To Menu**

Now you can try to assign stars, so go back to the level select screen, and play with [Profile Tool](#).

This concludes this tutorial. Be aware that what you've learned here is just a small portion of what you can do with Mad Level Manager. Please look at the other documentation pages to discover how to customize the look of your level select screen, and how Mad Level Manager can help you manage your levels and game progress!