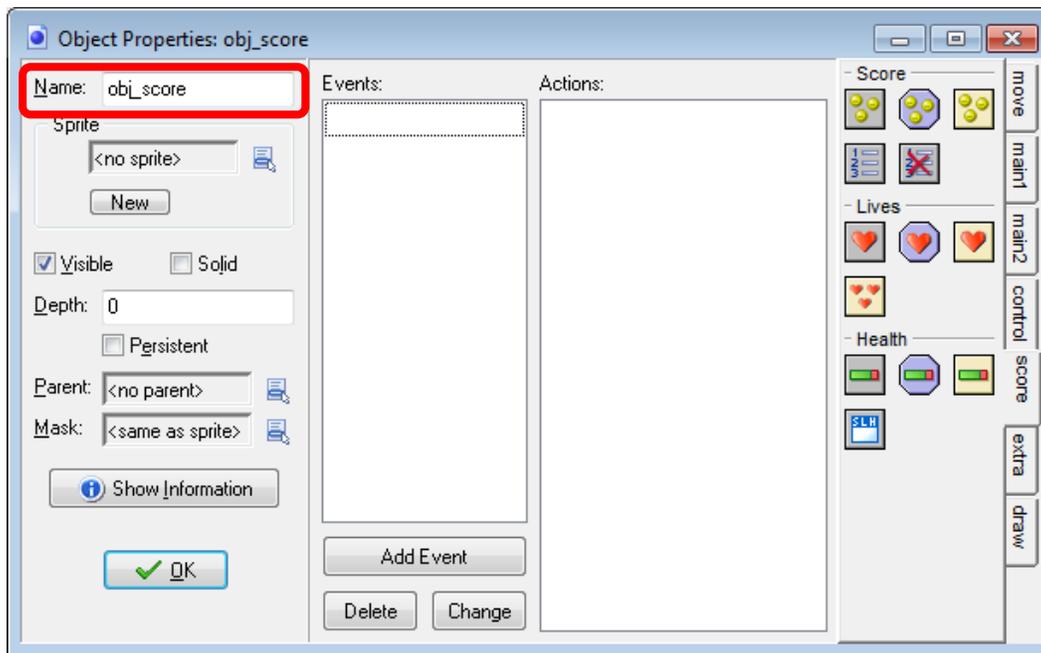


# KEEPING SCORE: HOW TO USE SCORES, LIVES AND HEALTH

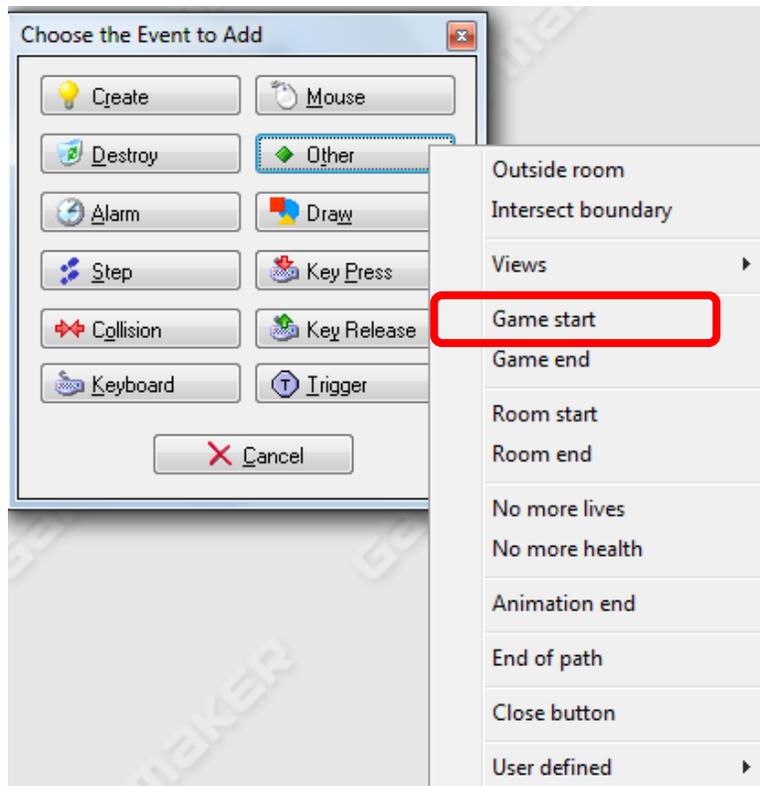
A game isn't much of a game unless you can measure how well you're doing. How well players are doing in a game is often measure by their score, how many lives they have and/or their health. GameMaker allows you to easily keep track of each of these things.

## KEEPING SCORE

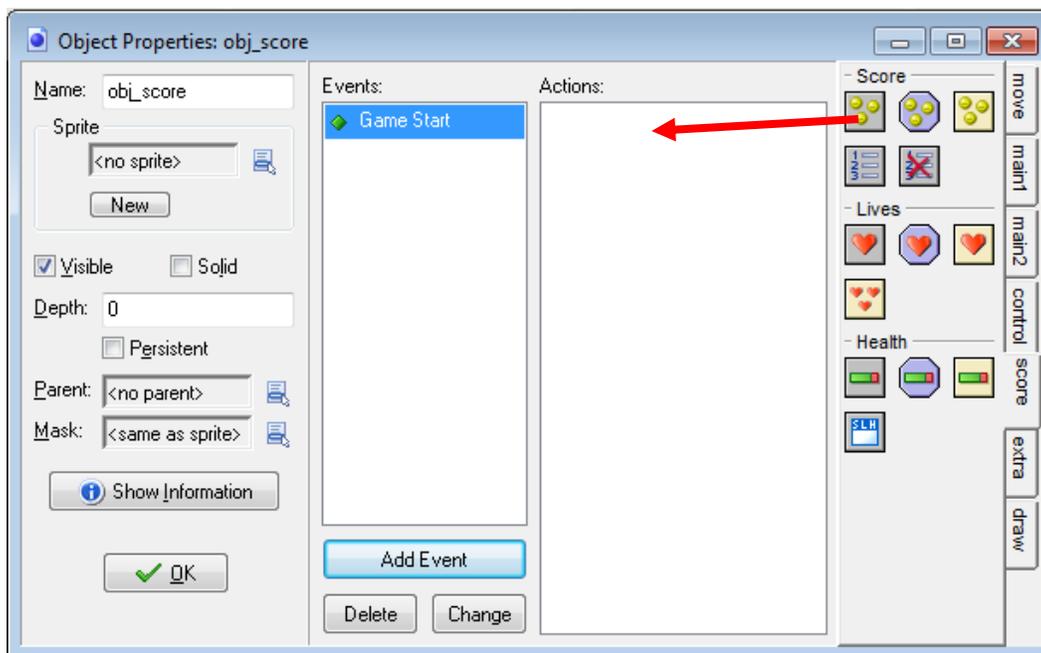
To keep score in our game, the first thing we're going to need to do is create an object. We will call this object **obj\_score**:



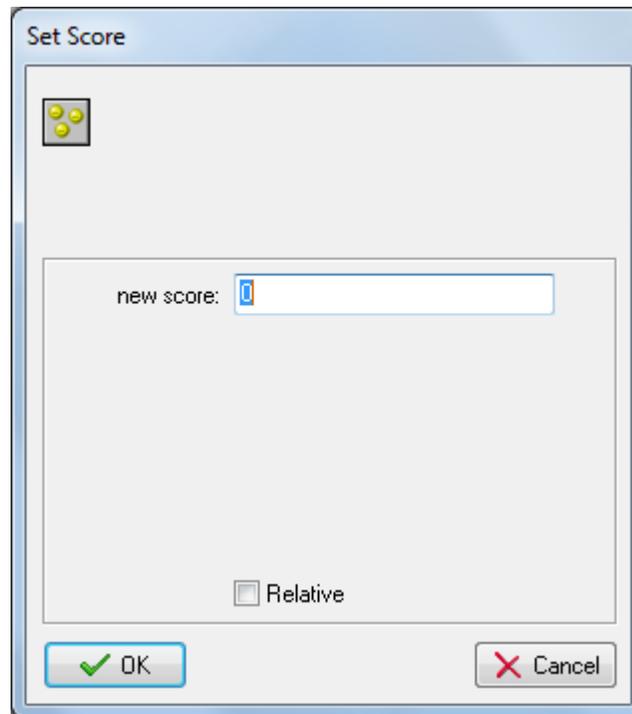
Next we will need to create an event so that the score is set and drawn at the start of the game. This will require us to create an event called **Game Start** which you can find by clicking **Add Event > Other**:



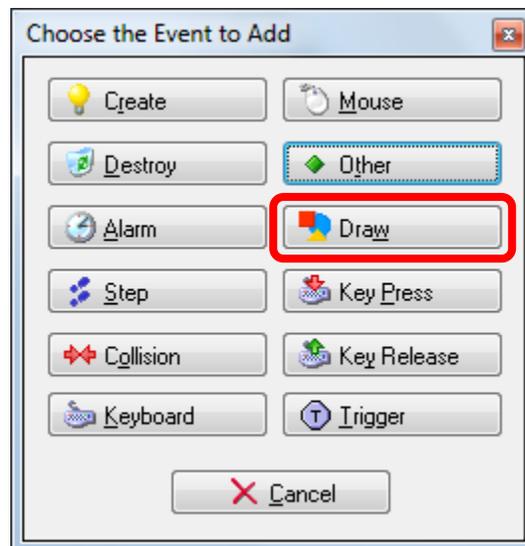
Next, we will need to set the initial score. To do so, we will need to click on the **score** tab and then drag the **Set Score** action to the Actions window:



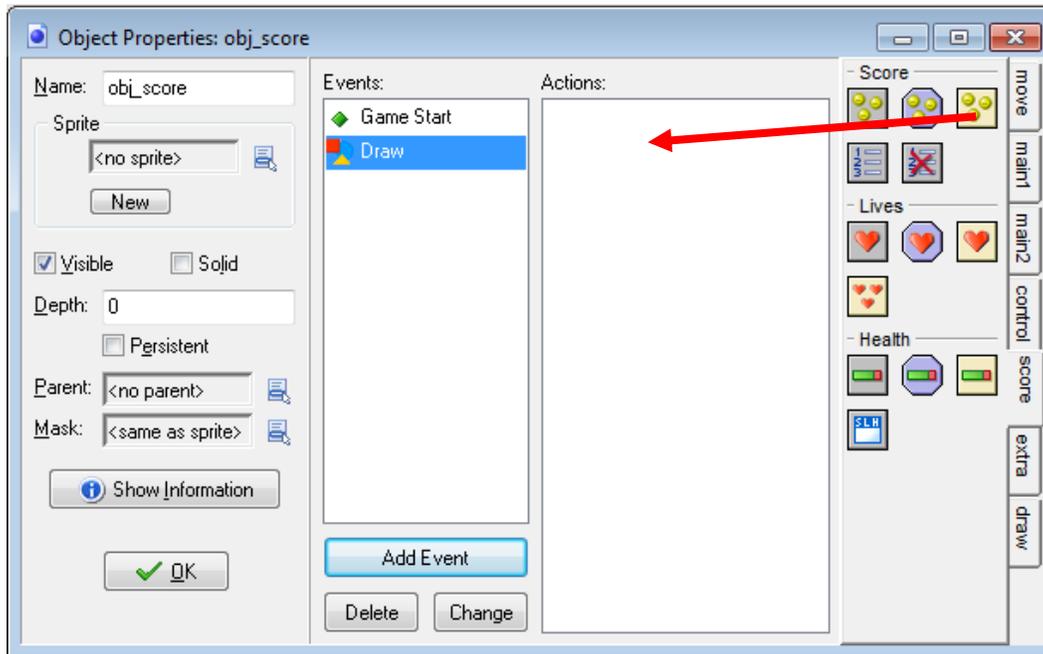
The **Set Score** properties window will appear allowing you to set whatever you want the score to be when the game starts. We will set the score to 0. Be sure to keep the **relative** checkbox unchecked because we do not want to set the score relative to whatever the score is when the game starts.



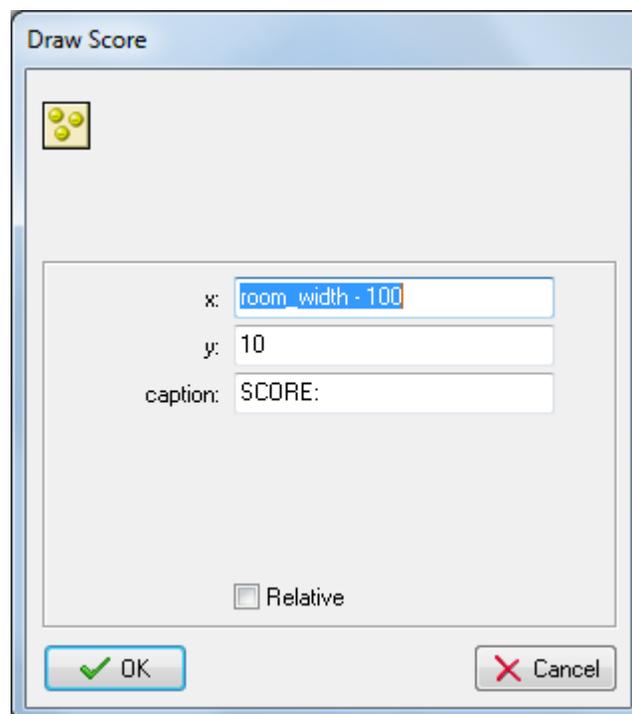
The next thing we will need to do is draw that score in the room. To do this, we will need to create a **Draw** event:



Now we will need to drag the **Draw Score** action to the Actions window:

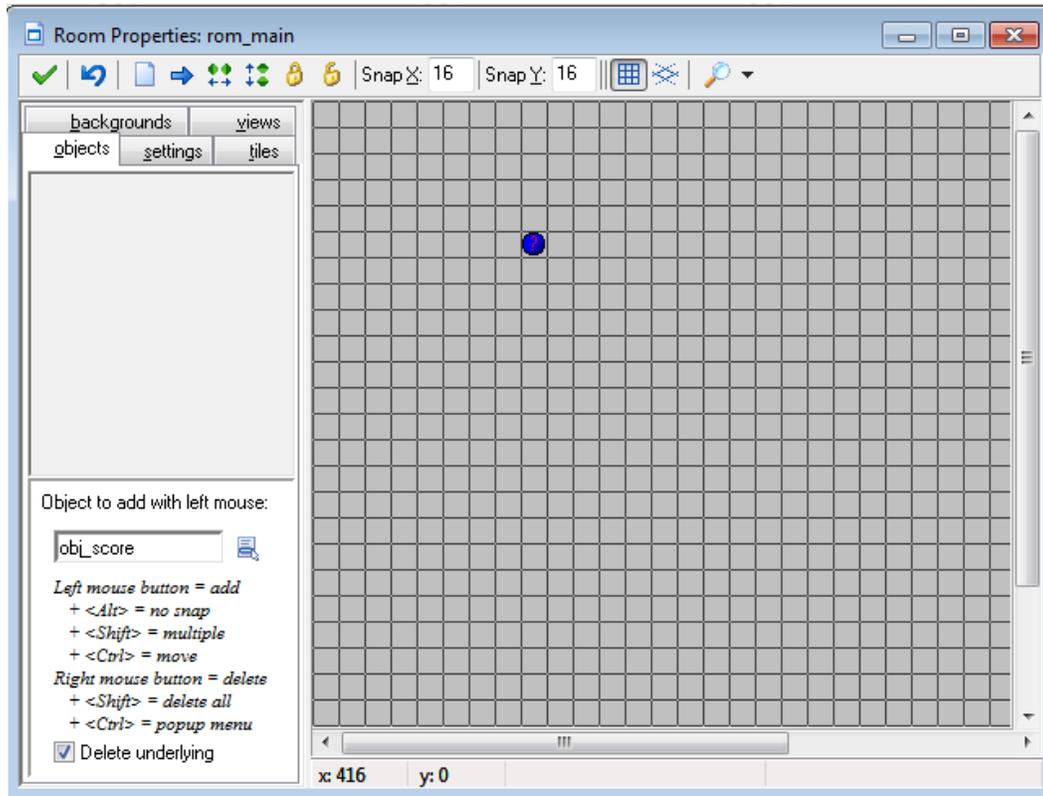


Next you will need to indicate where in the room you want the score to be drawn in the room (x and y) and what text you want to appear before the score is outputted. You can use the values **room\_width** and **room\_height** as part of your location. So, for example, if I want the score to be outputted to the far right of the room, I could enter **room\_width - 100**:

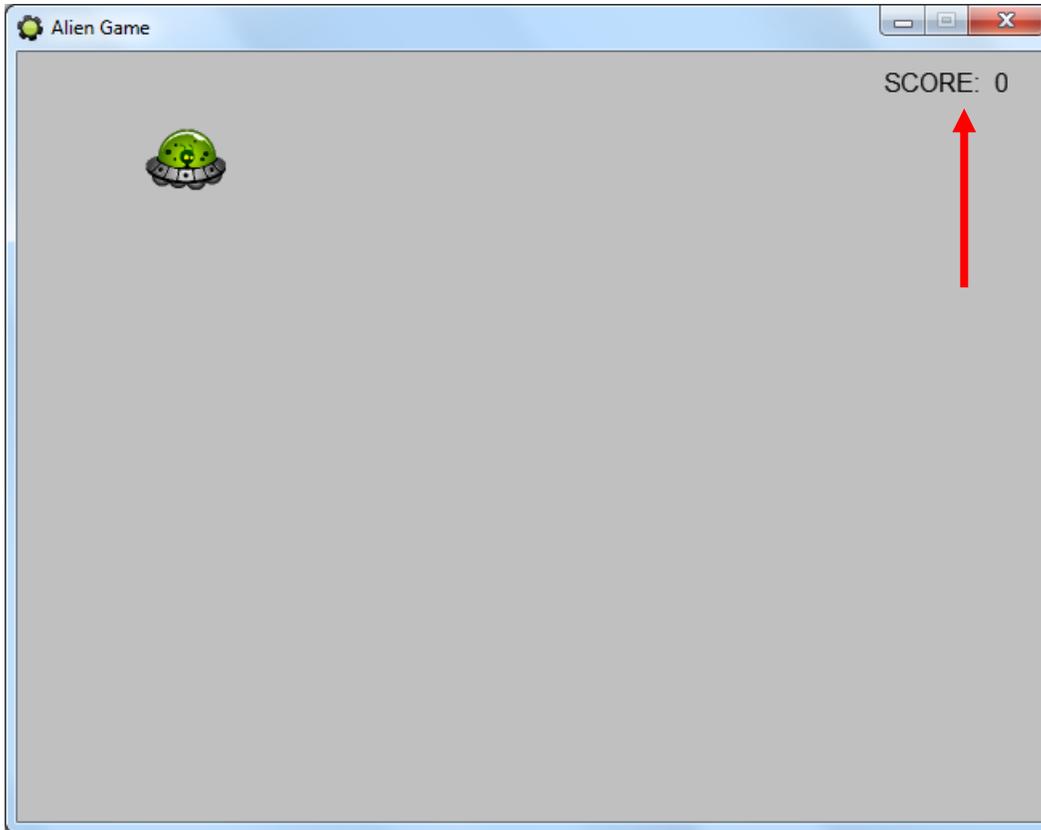


Our final step is to add the object to our room. You can add the object anywhere in the room. It doesn't matter where you add it because the location of the score will be based on the x- any y-values you entered when set the properties. You will also notice that when you add the object, all you will see is a

blue dot with a question mark in it. This will always be the case when you add an object without a sprite assigned to it.



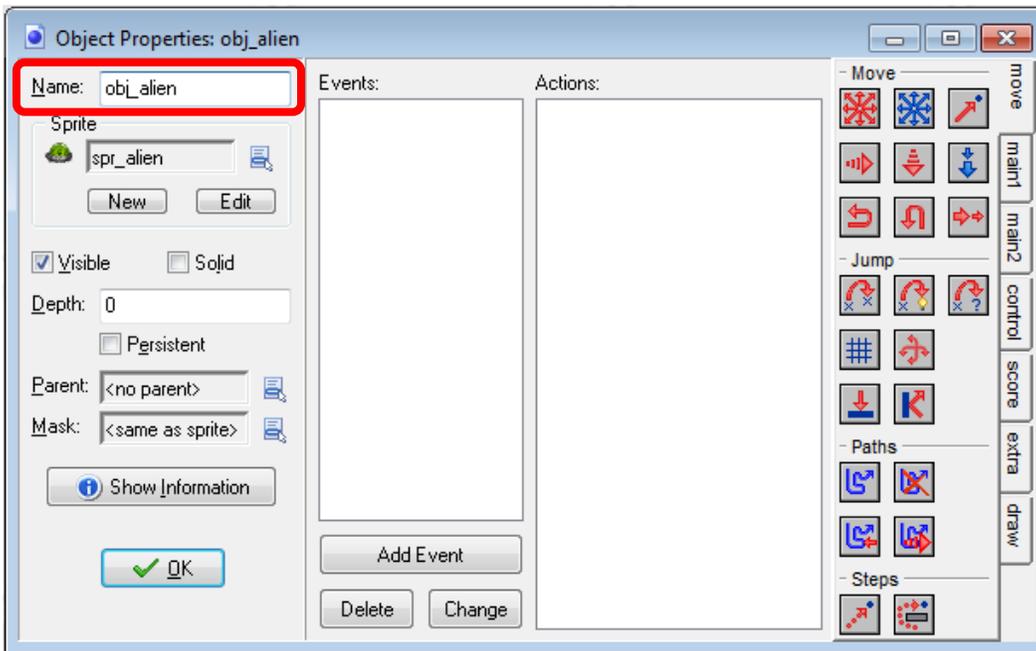
When you run the program, you will see the following output:



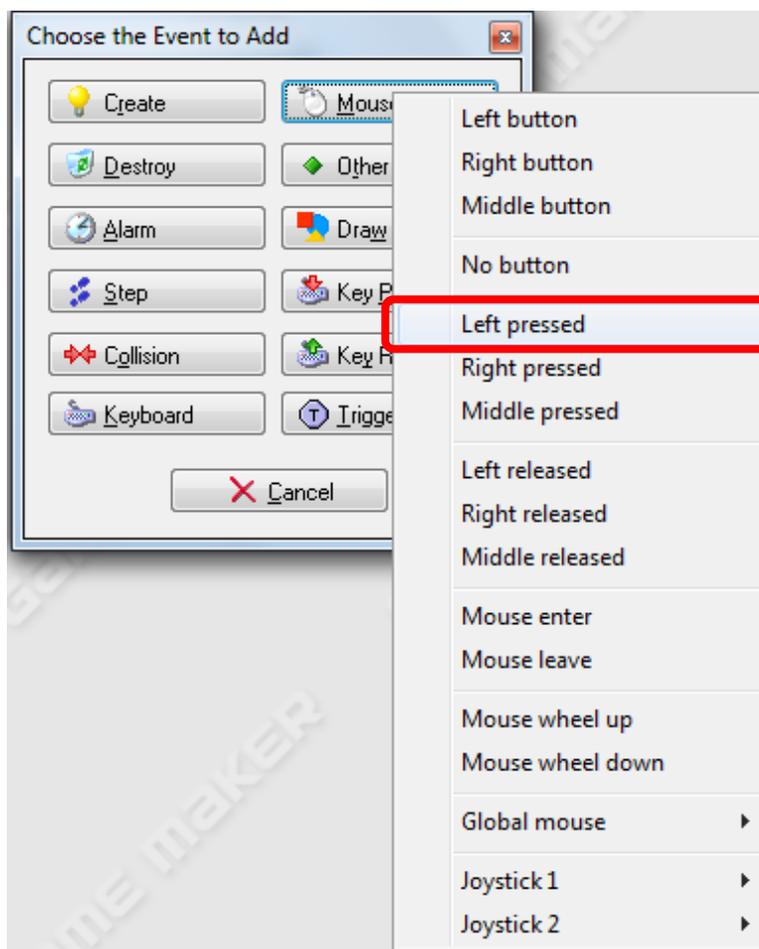
## UPDATING THE SCORE

Now that we have the score outputted, let's see how we can change the score. We're going to use a very simple example whereby the score will increase whenever the user clicks the left mouse button on an object that we will create and decrease when the user clicks the right mouse button on the object.

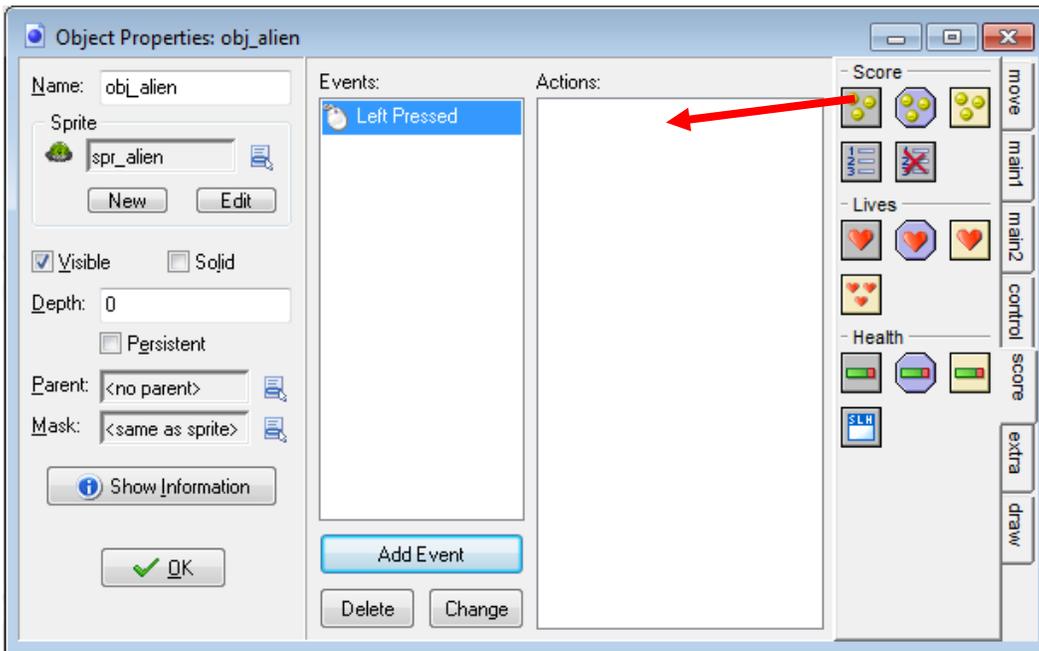
Let's start by creating an object called **obj\_alien** (feel free to use any sprite you want):



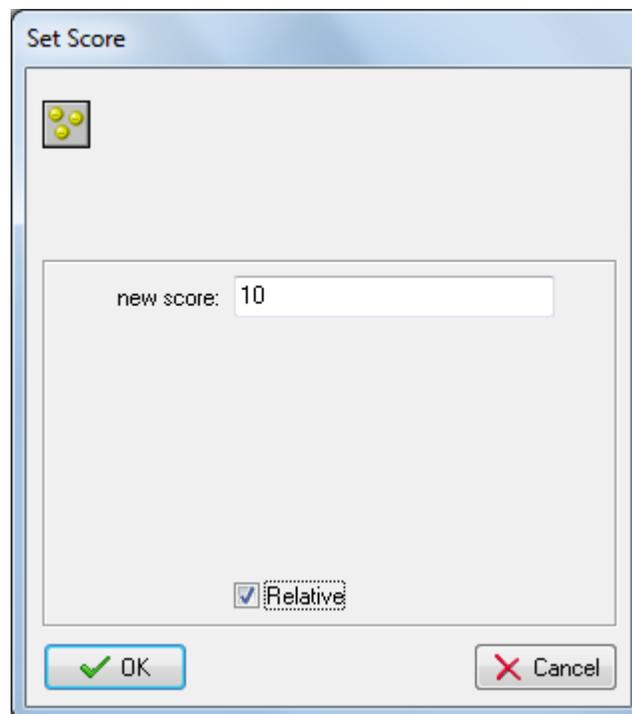
Now we will create a **Left Pressed** mouse event:



Now we're going to drag the **Set Score** action into the Actions window:

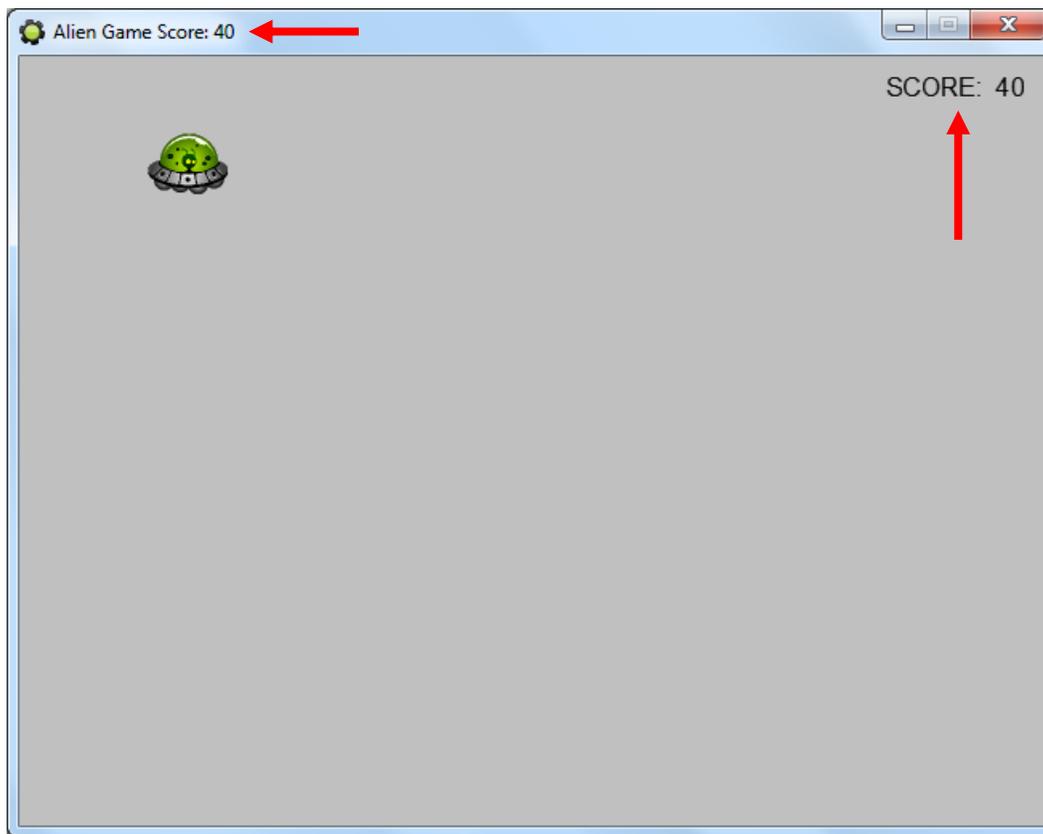


We will want the score to go up by 10 each time the user clicks the left mouse button, so we will need to set the score property to 10 and select the relative checkbox (so that the score goes up by 10 relative to what the score is):

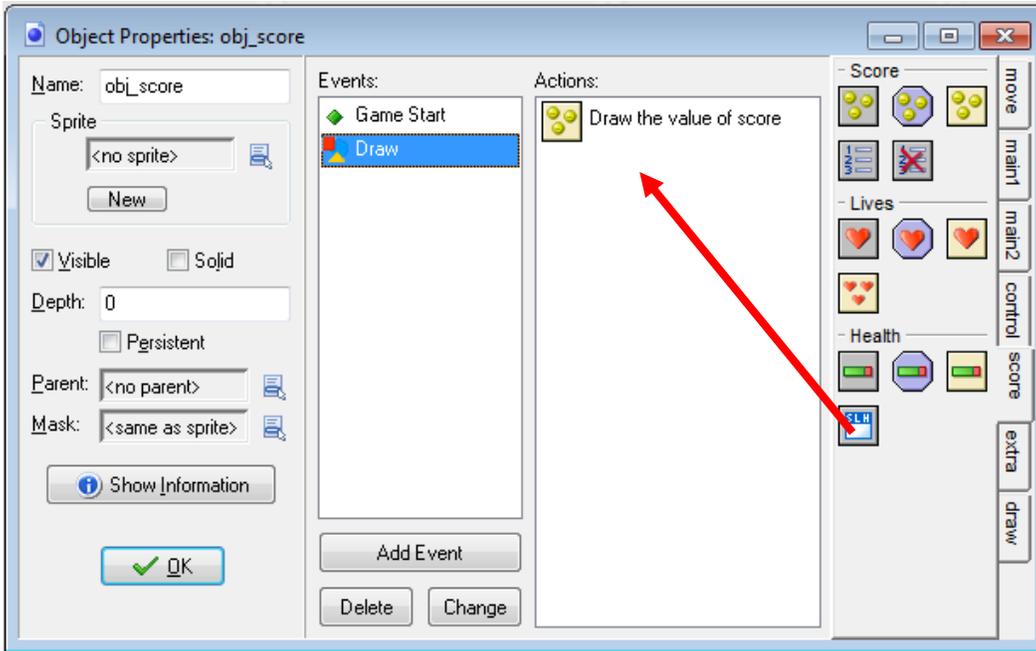


The last step is adding the object to the room and then running it. You will notice that each time you click on the object, the score increases by 10.

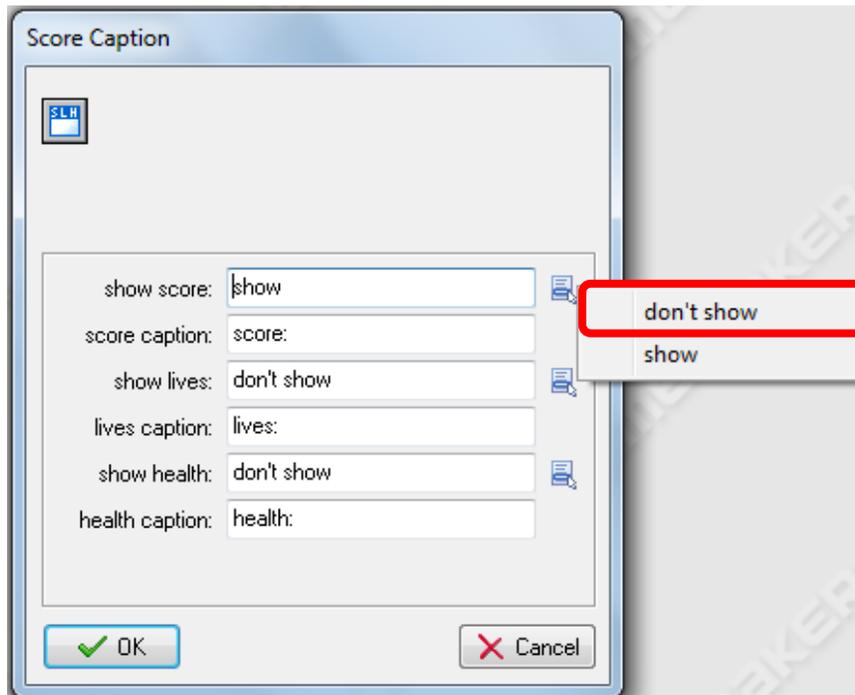
You'll notice when you run it, the score also appears in the title bar of your window as well as the location you set the score in your room:



To remove the score from the title bar, you would need to go to the **Draw** event and drag the **Score Caption** action to the Actions window:



This will bring up the Score Caption properties window, where you can select **Don't Show** in the **Show Score** property:



Now when you run the program, the score will not appear in the title bar of the window.

To decrease the score by 10 when the user clicks the right arrow key on the object, you would simply need to create a **Right Pressed** mouse event and set the relative score to **-10**:



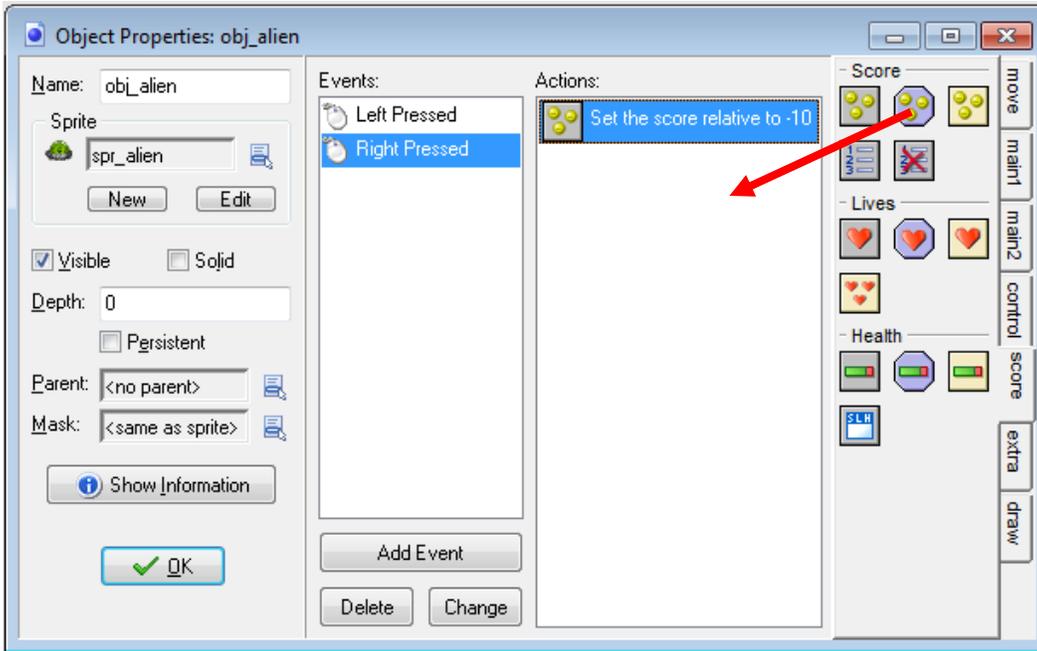
Now when you run the program and press the right mouse button on the object, the score should decrease by 10.

## HIGH SCORE TABLE

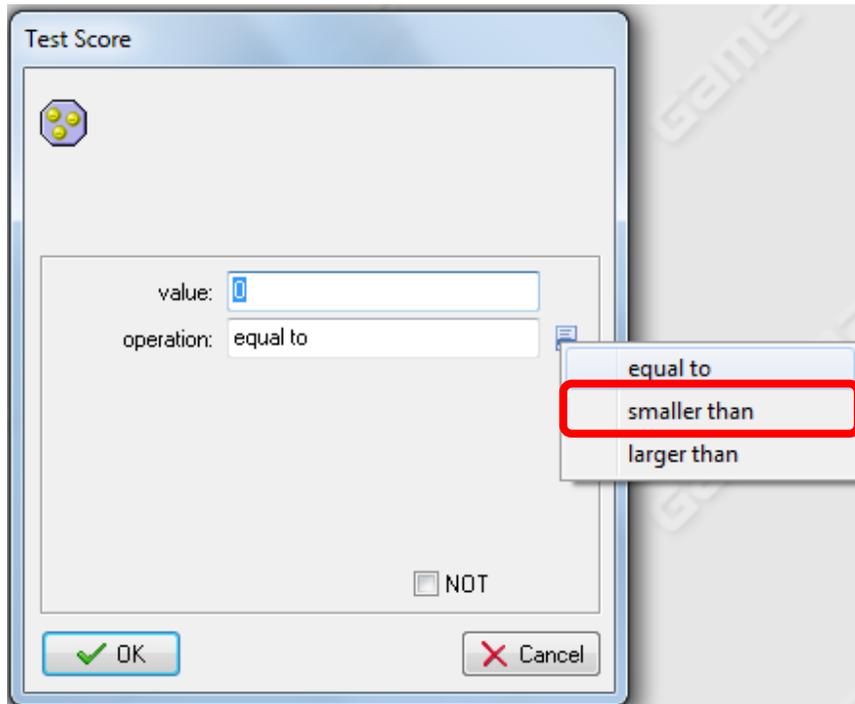
GameMaker helps you create and maintain high scores for your games by providing a **Show Highscore** action. Typically a high score table would be used and outputted at the end of a game so players can see how their score ranks among the best.

To incorporate a High Score table to our game, we will first need to determine when we want to display the high score table. In this example, we will output the high score table when the user has 0 points left. The first thing we will need to do is get the program to check when the user's score is 0. Obviously the only time the user's score will equal 0 (other than when the game starts) is when the user clicks the right button on the mouse (because this is the only time we are deducting points from the user's score).

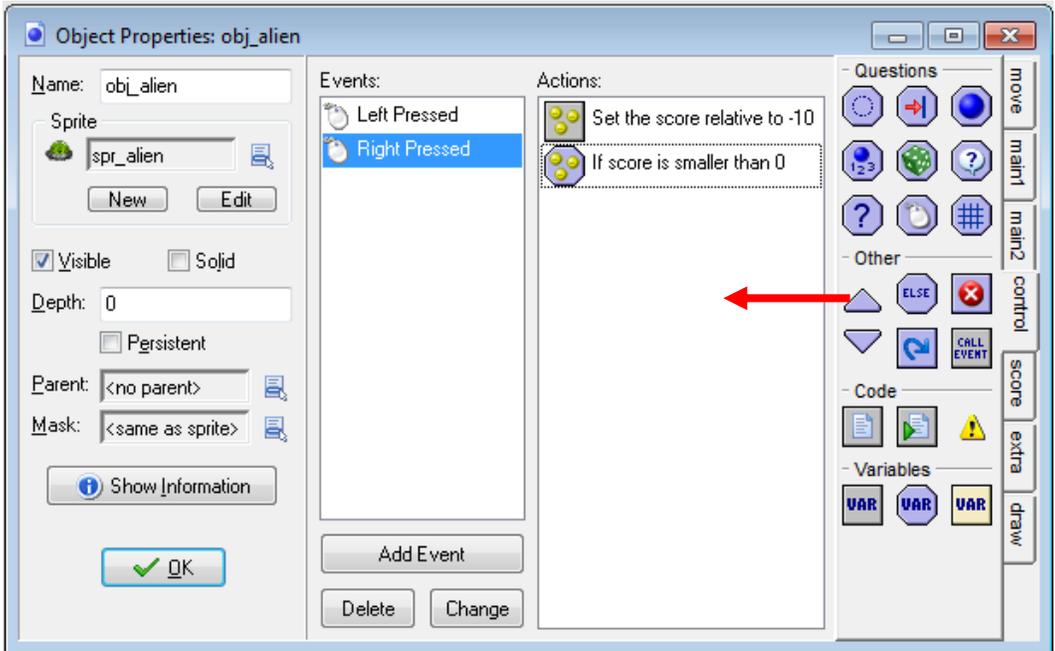
So let's add the **Test Score** action to the Actions window for the **Right Pressed** event that we had already created:



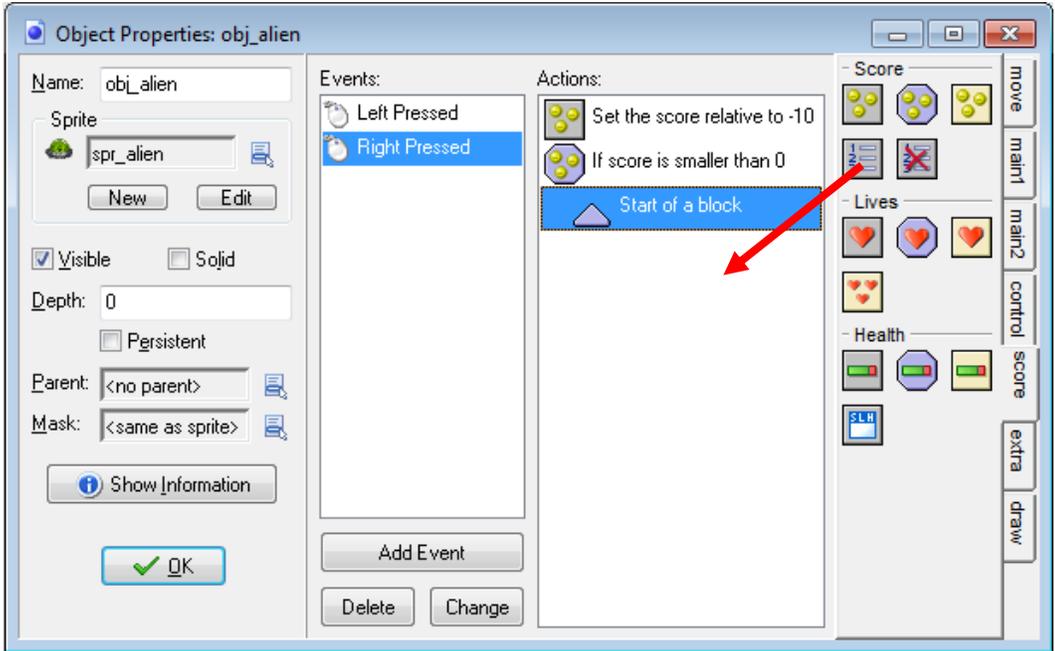
In the **Test Score** properties window, we can set what value we want to test the score. In this example, we want to test if the score is less than 0, so we will need to enter **0** in the **Value** property and **Smaller than** in the **Operation** property:



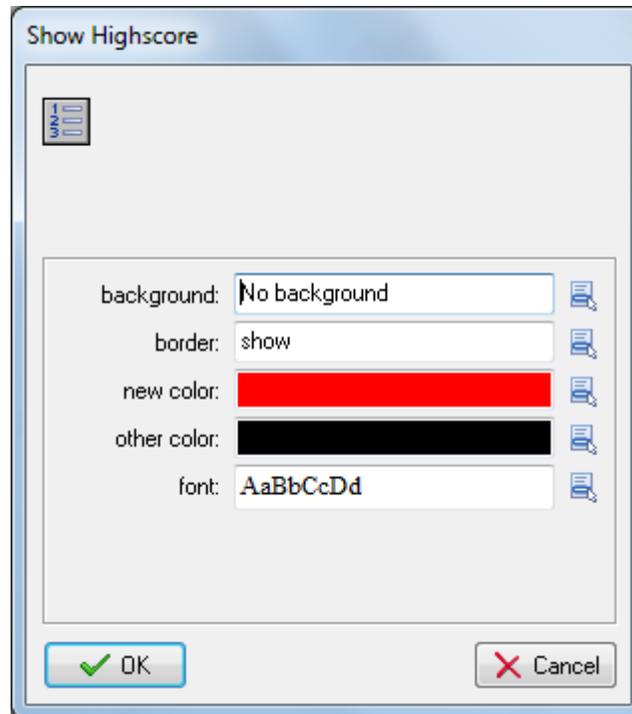
Next we will want to display the High Score table. Before we can do this, however, we will need add a **Start Block** which you can find in the **control** tab.



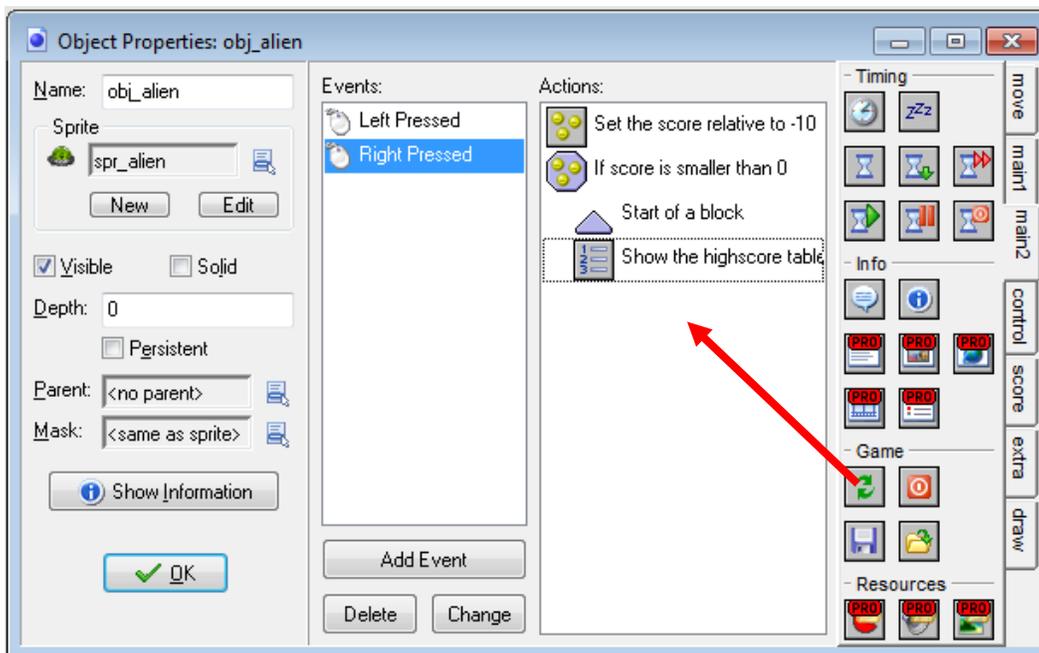
Once you have added a Start Block, you will need to drag the **Show Highscore** action to the Actions window:



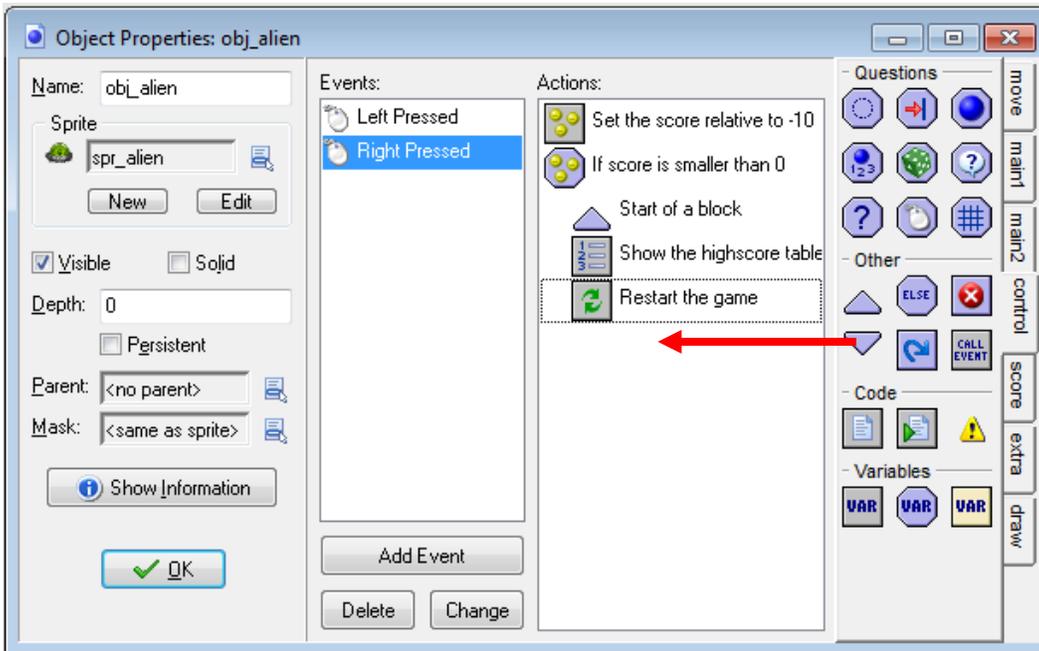
The **Show Highscore** properties window will appear allowing you to set certain things about the table (i.e. background, border, colours and fonts).



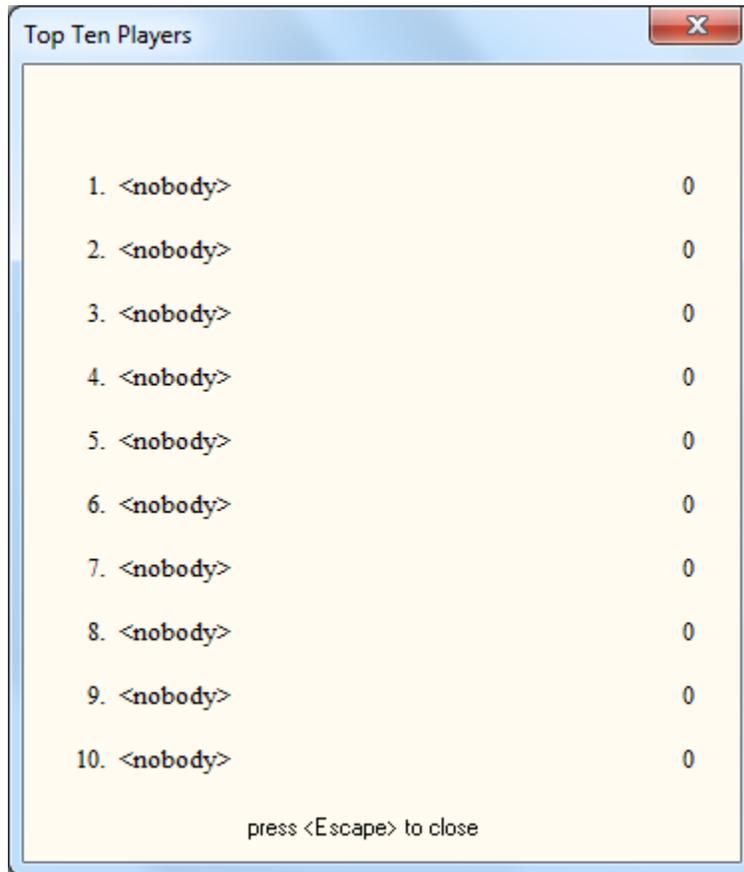
Let's also restart the game, after the High Score table appears. To do this, you will need to drag the **Restart Game** action to the Actions menu:



Now that we have included all the actions we want happening if the score is less than 0, we must end the block by dragging the **End Block** action to the Actions menu:

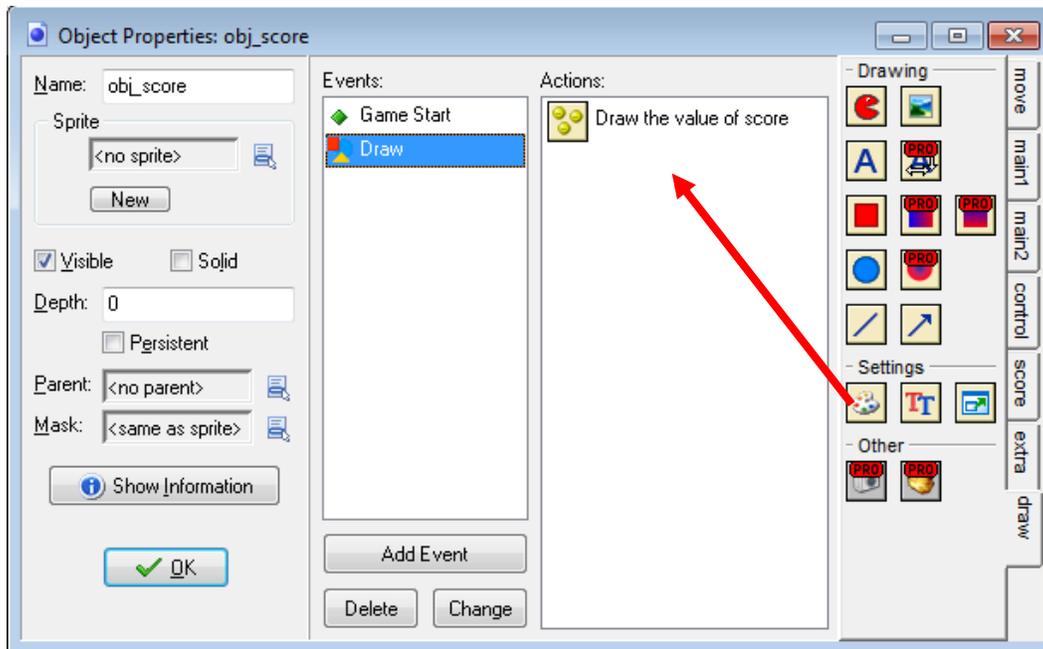


Now when you run the game, the High Score table should appear when the user's score is less than 0. Of course, users will not be able to enter their name since their score will not be among the top 10 scores.

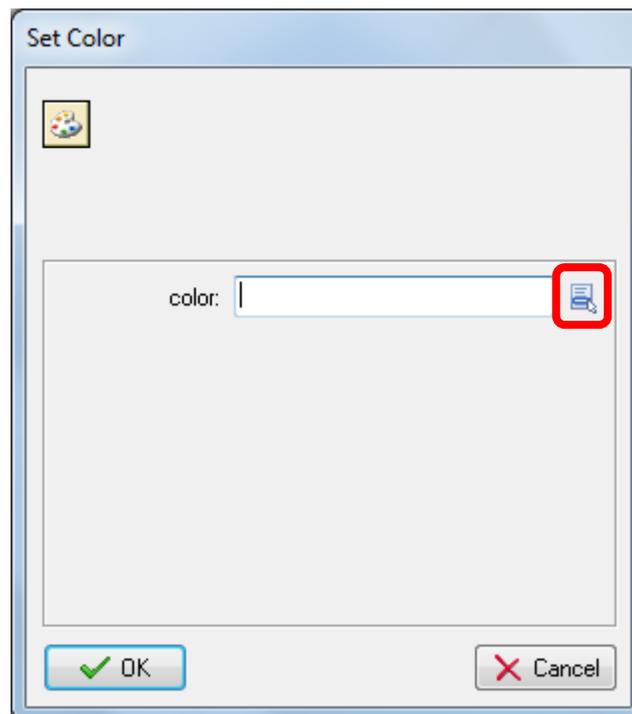


## CHANGING THE TEXT COLOUR

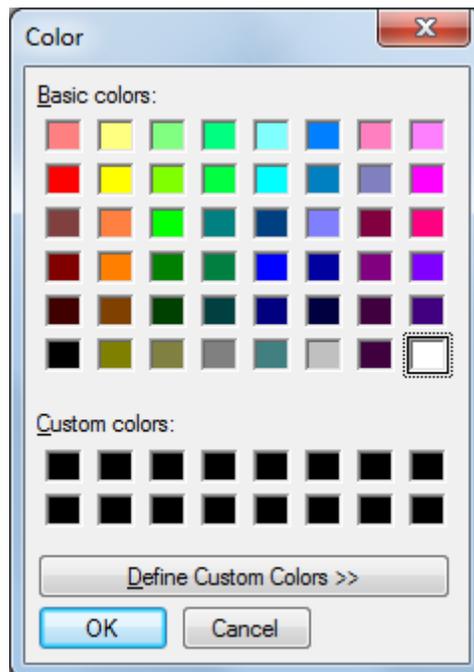
The default colour of text is black. However, you can change the colour to any colour you want. To do so you will need to open up the score object (**obj\_score**), go to the **Draw** event, click the **draw** tab and drag the **Set Color** action to the list of Actions:



The Set Color properties window will appear, where you will need to click the properties button:

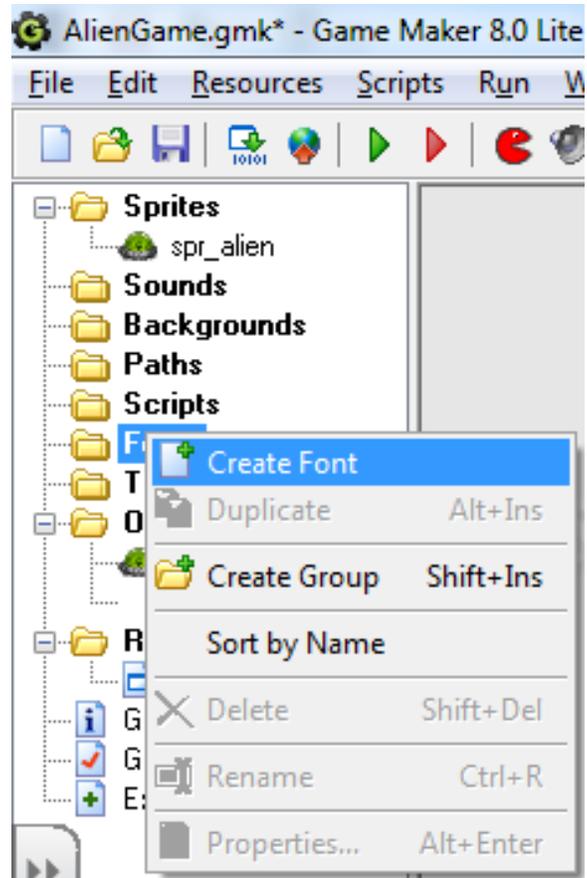


This will bring up the Color palette where you can select any colour you want for your text:

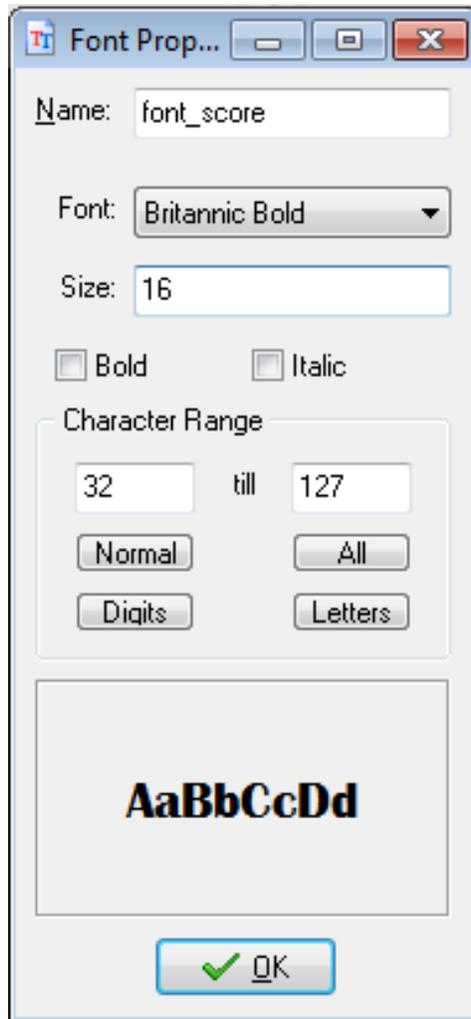


## SETTING THE FONT

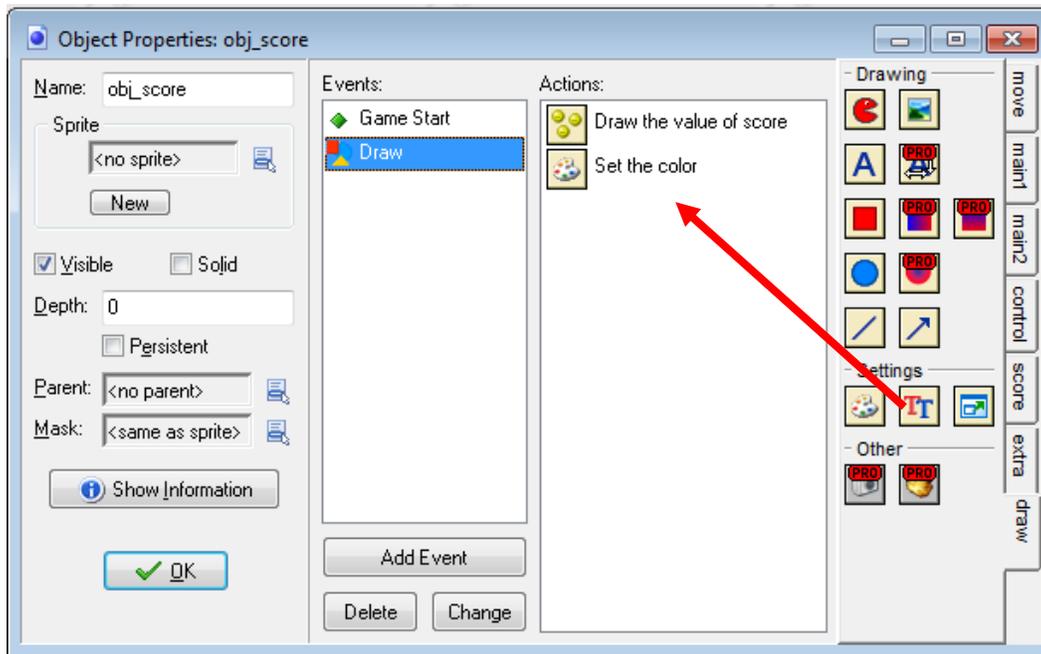
Just like you can change the colour of the text, you can also change the font that is used to output the score. The first thing you will need to do is create a font by right-clicking on the **Fonts** folder and selecting **Create Font**:



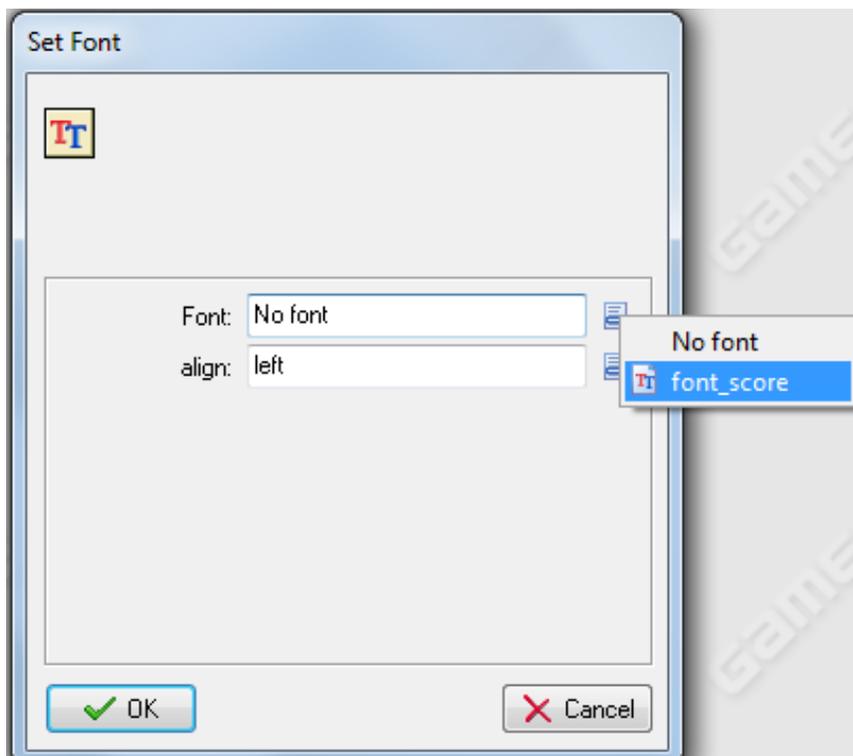
This will bring up the Font properties window, where you need to name your font object (e.g. **font\_score**) and set the font type, font size, etc.



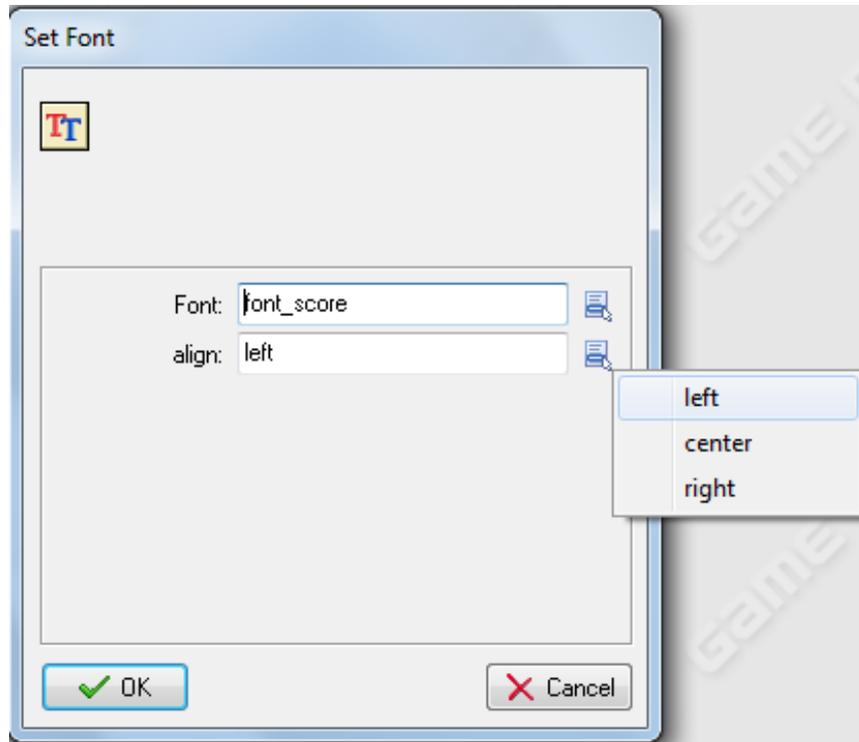
Once you have create your font, you will need to set the font by re-opening the score object (**obj\_score**), go to the **Draw** event, click the **draw** tab, and drag the **Set Font** action into the Actions menu:



This will bring up the Set Font properties window where you will need to select the font you created:



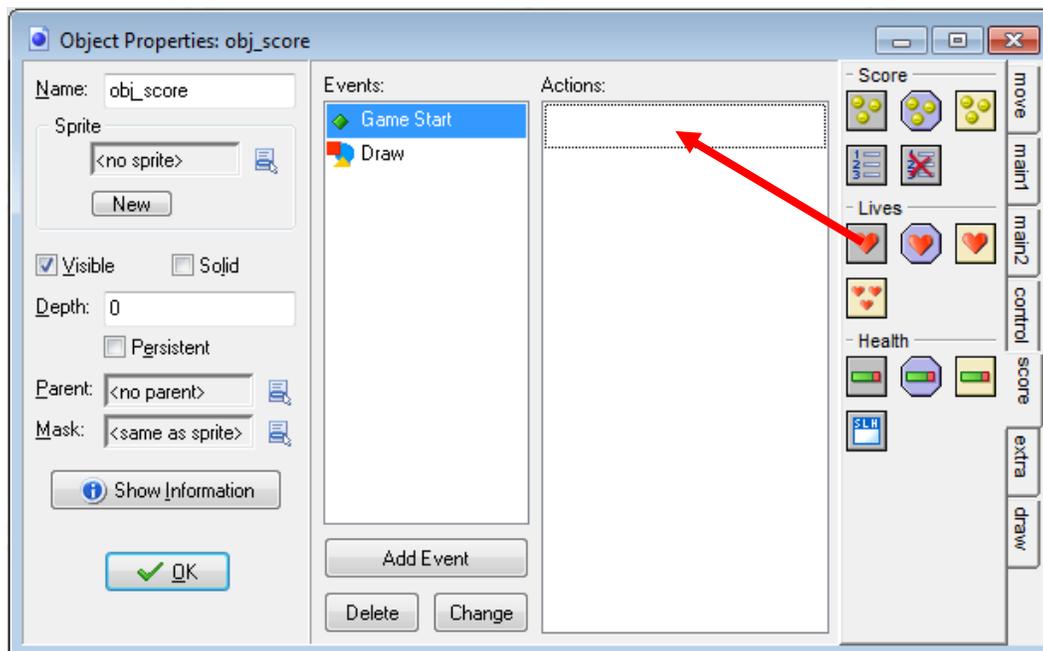
You also have the option to have the font left, centre or right aligned:

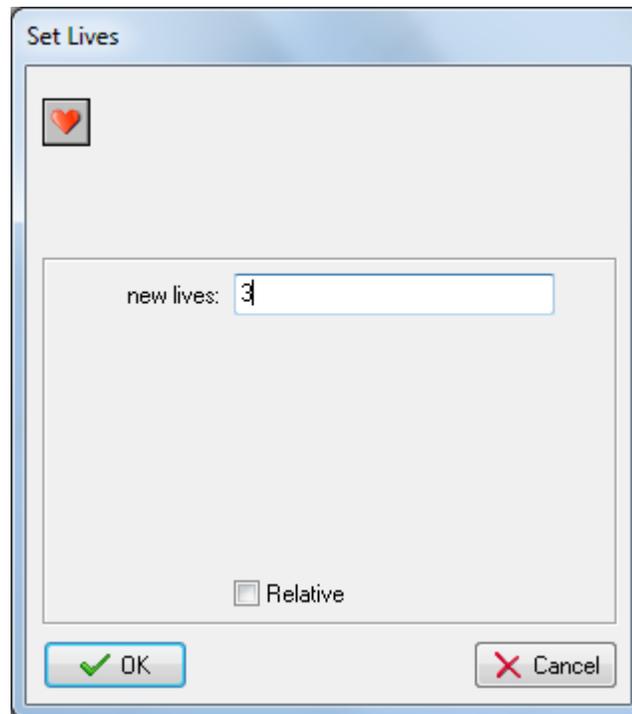


That's all you need to do to set your font.

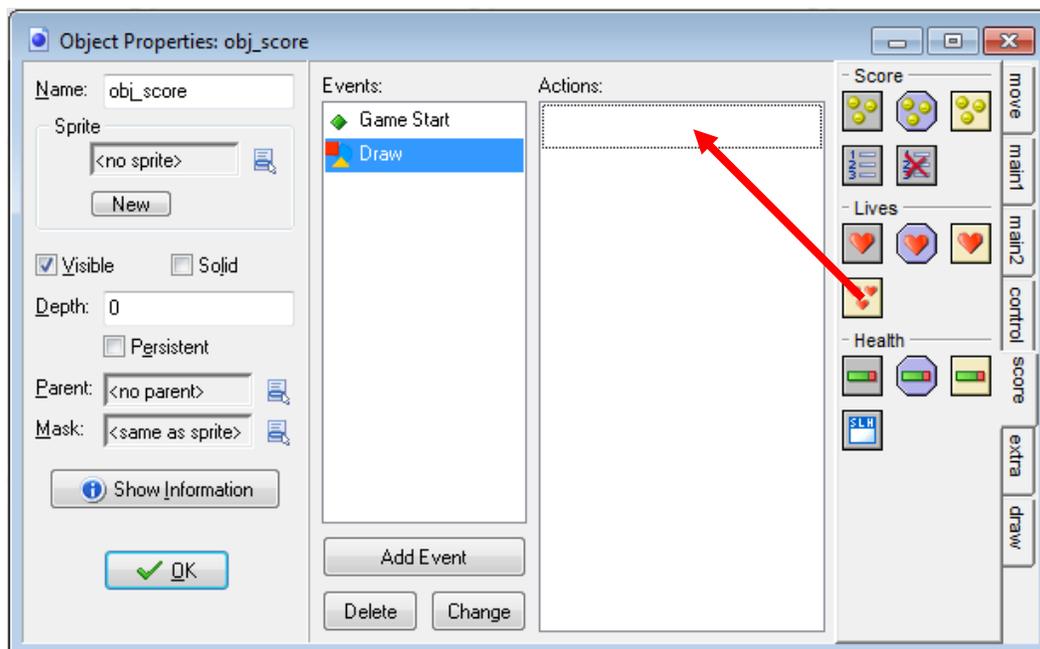
## WORKING WITH LIVES

Working with lives is very similar to working with scores. When you drag the **Set Lives** action to the Actions window, you will have the option to set an initial number of lives instead of an initial score:

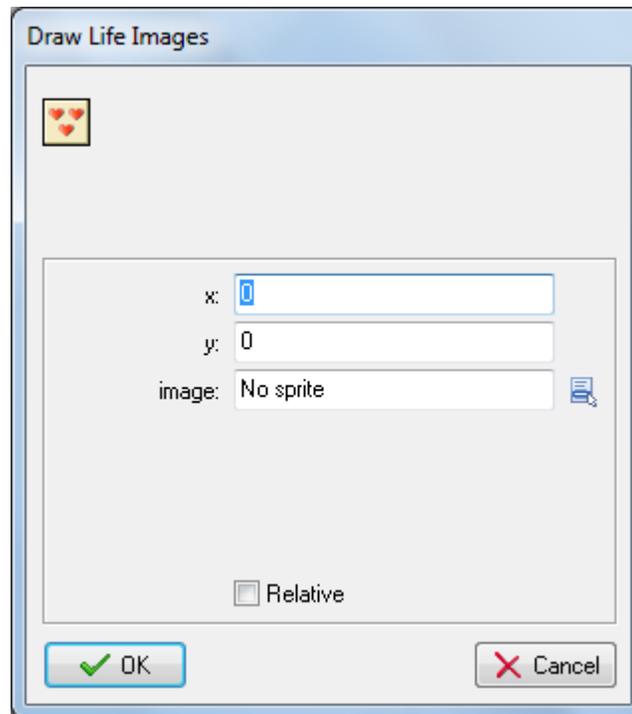




You can also draw images of lives instead of simply outputting a number by using the **Draw Life Images** action:

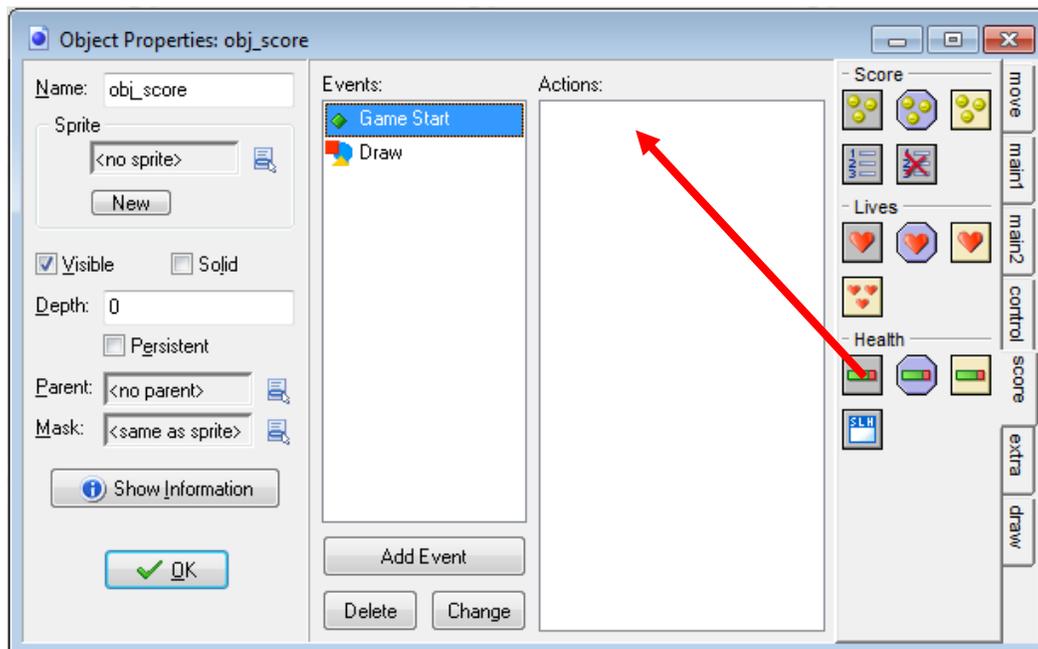


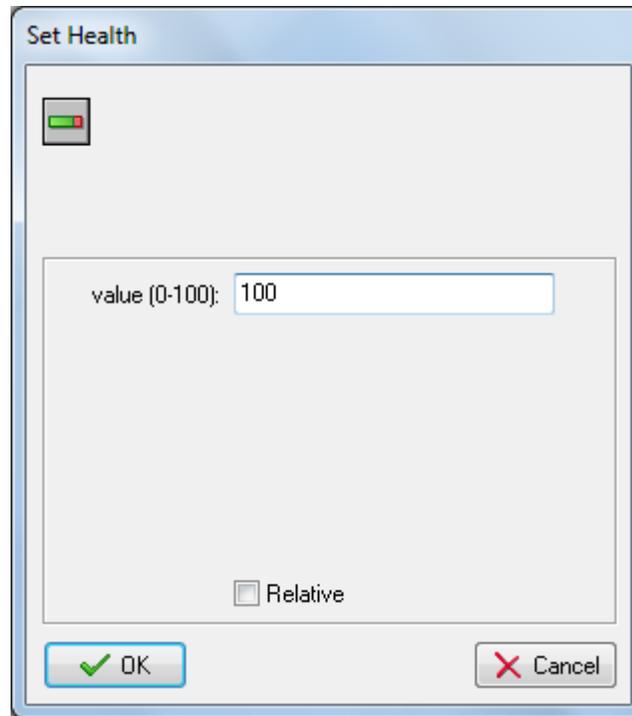
If you choose to draw life images, you will need to have created a sprite and then select the sprite you want to use. You will also need to indicate where in the room you want to draw the life images (x- and y-values).



## WORKING WITH HEALTH BARS

Health bars is another way of tracking a player's progress. Using the **Set Health** action, you can set the player's health by entering a value between 0 and 100:





Although you cannot use your own sprite to represent a player's health, you can set the position and size of the health bar (x1, y1, x2, y2), the colour of the bar and the colour of the background.

