QUESTIONS ACTIONS IN GAMEMAKER

Questions Actions are useful actions that come in handy when you need to check things, such as whether something exists, if there is a collision, if two values are the same as each other (or greater, or less), etc. For all Questions Actions there is a check box labeled **NOT**. If you check this, the result of the question is reversed so that if the result was true it becomes false and if it was false, it becomes true.

If you look at the action **Check Empty**, for example, you can have it without checking the "NOT" box, which means it is asking "if the place that I put the instance is empty", but if you have "NOT" checked then you are asking "if the place that I put the instance is not empty". This allows you to perform certain actions when a question is not true, or not equal to or even not greater or not less than!

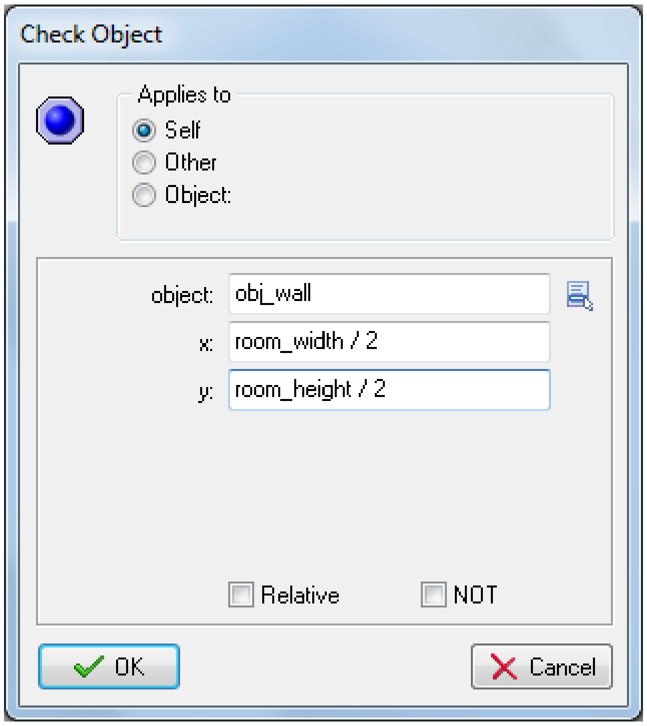
Another thing to watch when dealing with these actions is that you can indicate whether they should apply to all instances of a particular object or not. If you choose to apply them to all instances of an object, the result will be true (or whatever) only if it is the same for all instances of that object. For example, you can check all instances of an object to see if the position slightly to the right is collision free.

CHECK OBJECT

This action checks to see if there is an instance of a specified object at the indicated position. This question returns true if an object placed at the indicated position meets another object. These collisions will be precise if both instances have the mask index or sprite mask defined as being precise otherwise they will be based on whether their bounding box overlaps or not.

This action requires that both the object doing the check and the object being checked has a mask index or a sprite with a valid collision mask assigned to it.

In the following example, I am checking to see if there is a wall in the middle of the room:



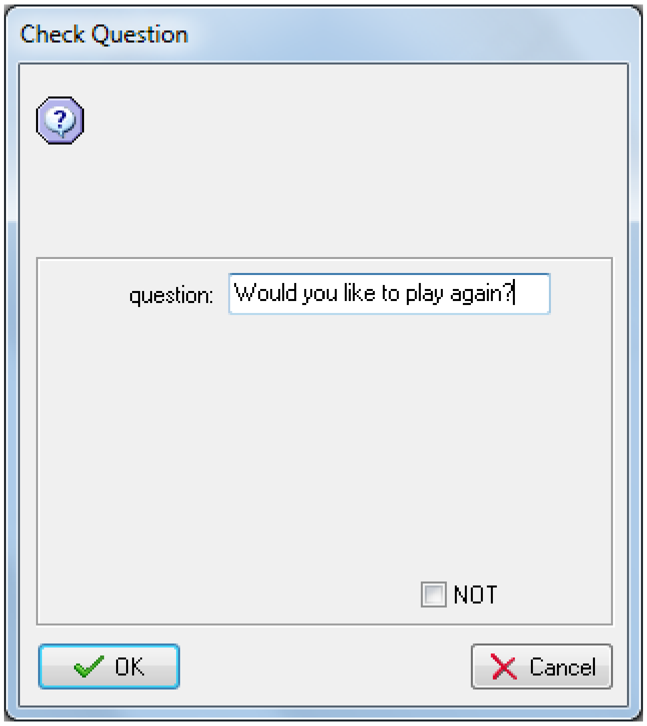
object: obj_wall
x: room_width / 2
y: room_height / 2

Applies to: [✓ Self] Other Object:

CHECK QUESTION

This action lets you specify a question for the player to answer either yes or no to, with the subsequent actions running (or block of actions) if the result is yes (true). The question is shown in an independent pop-up dialogue.

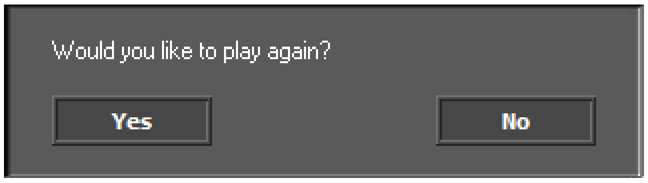
In the following example, I am asking the user if they would like to play again:


Check Question

question: Would you like to play again?

NOT

This produces the following dialog box:


Would you like to play again?

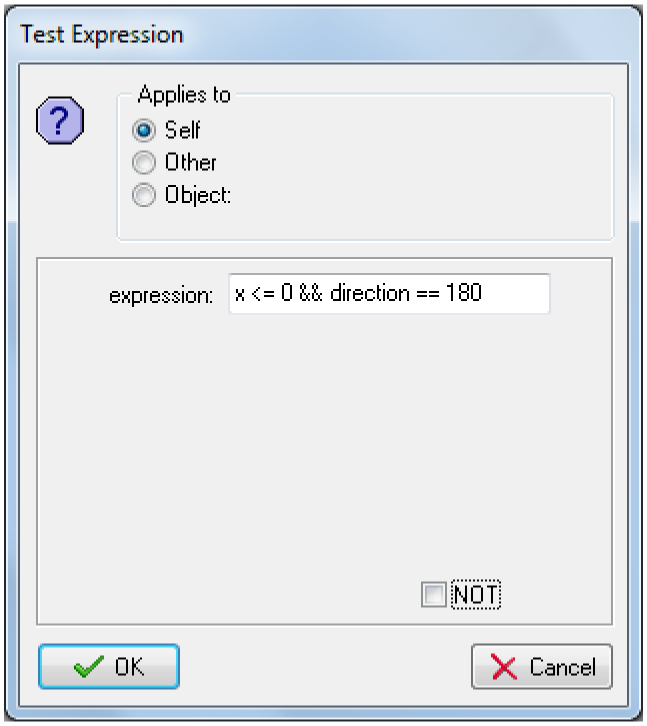
You can have the program do something when the user selects YES, and something else if the user selects NO.

TEST EXPRESSION

This is the most general question action whereby you can enter an arbitrary expression and evaluate it. If the expression evaluates to true (that is, a number larger or equal to 0.5) then the next action (or block of actions) is performed.

When using many of the actions in GameMaker, you are expected to input a value as one of the parameters, but sometimes that is not enough and you may find it necessary to input a formula or an expression to get the required result from the action. GameMaker allows the use of any of the in-built instance variables, constants or global variables as well as the use of your own (previously defined) variables and even mathematical formulas.

In the following example, I'm going to check if an object's x-position is less than or equal to 0 and its direction is 180° because if it is, I will reverse its direction.



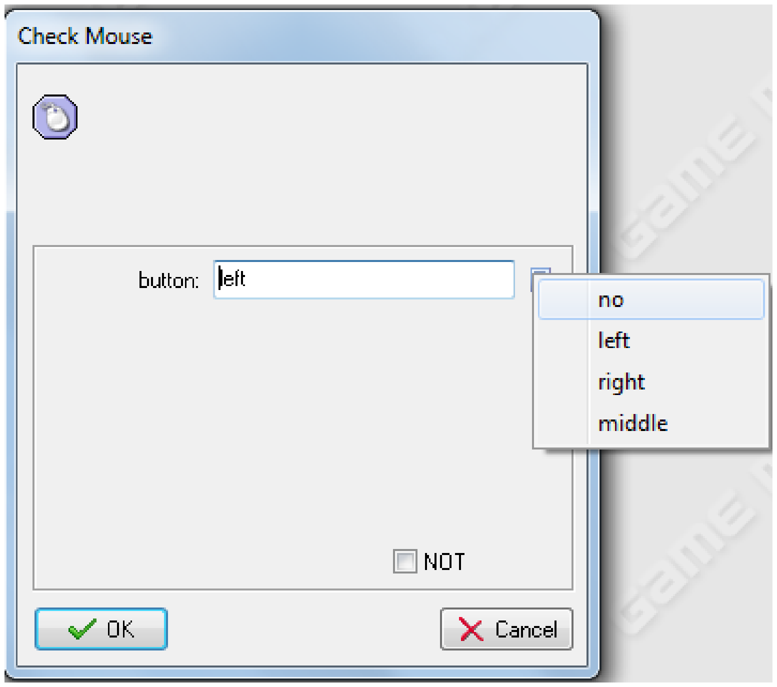
expression: x <= 0 && direction == 180

You can learn more about how to write expressions in GameMaker by clicking here: http://docs.yoyogames.com/source/dadiospice/002_reference/001_gml%20language%20overview/401_04_expressions.html.

CHECK MOUSE

This action will return true if the indicated mouse button is pressed. For example, in the step event of an object you can check whether a mouse button is pressed and, if so, move the instance to that position (you would use the jump to a point action with values **mouse_x** and **mouse_y**).

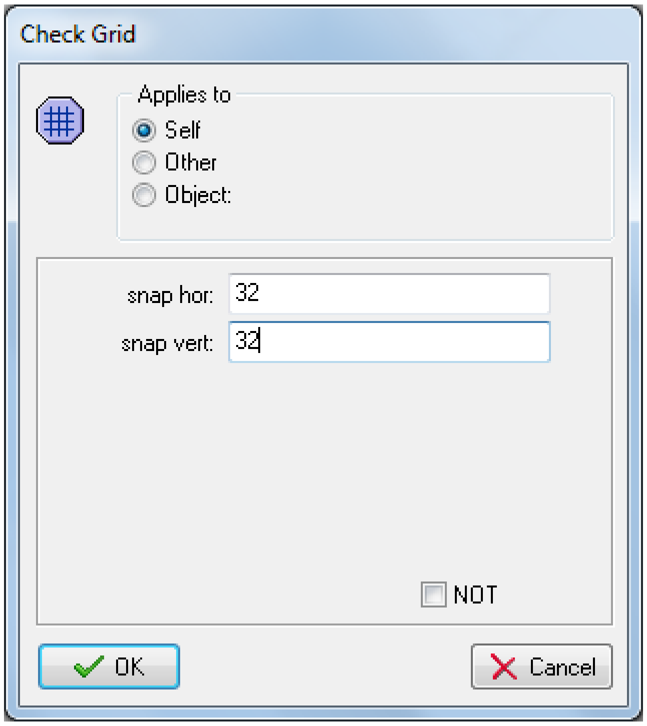
You can check if no button, the left button, the middle button or the right button is clicked.



CHECK GRID

The **Check Grid** action returns true if the position of an object lies on an imaginary grid, made up of spaces defined by you when you add values to the vertical and horizontal spacing parameters. A common use for this action is to do a check and only let the player move if his character is aligned to the grid (similar to what we did with Pac-Man).

For the **Snap Horizontal** and **Snap Vertical** parameters, you should indicate the snap X and snap Y values that you have set up in your room.



[✓ Self] Other Object:

snap hor: 32

snap vert: 32

NOT