

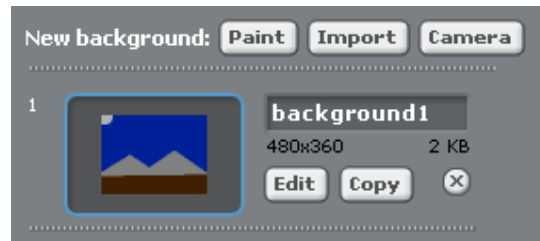
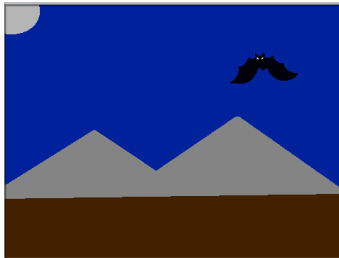
Scratch 1.4 Exercises

BY MR. REID

Our Fourth Program: Flying Bat Animation

In this exercise we will create a night time scene where our mouse will control the flight path of a bat. We will be using the two step process: Visual step, followed by Programming step. Remember that the visual step includes painting the stage, and making, selecting and changing sprites.

The Visual Stage I



Step1. Painting the stage requires you to select the stage and chose “edit”. Create a “night time” scene by using dark colors. In the example above I’ve suggested the pyramids of Egypt under a pail moon. You may draw any night time scene you wish.



Step2. Find the black bat and load bat2-a as the main sprite.

Step3. Choose Costumes and “import” bat2-b as the second costume.

Step4. Change the rotation of the Sprite to “don’t rotate”.

The animation in this program will come from flipping between the two bat costumes for sprite one. Each sprite can have multiple costumes. It is important to distinguish between a sprite and its costume. In this program we have only one sprite and the stage.

The Programming Stage I

In this section we will learn about animation and control for one sprite.



Above we have two blocks of code. Each block begins with a “When Flag Clicked” block. Each block of code contains a “forever” loop. This means the code blocks inside the loops will repeat until the program is stopped by pressing the red stop button above the stage.



When two independent blocks of code run, at the same time, this is called multi-threading.

- Step1: Drag some blocks onto the script area as shown above.
- Step2: Test your code as you go along. Observe what happens!
- Step3: Complete both blocks of code and test your program.

Questions

What happens to the Bat sprite if the rotate button is selected?

What happens if the wait block is missing or set to a different time?

What happens if the move block is missing or set to a different number?

Experiment

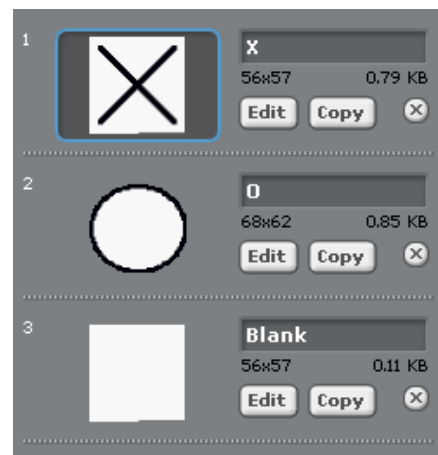
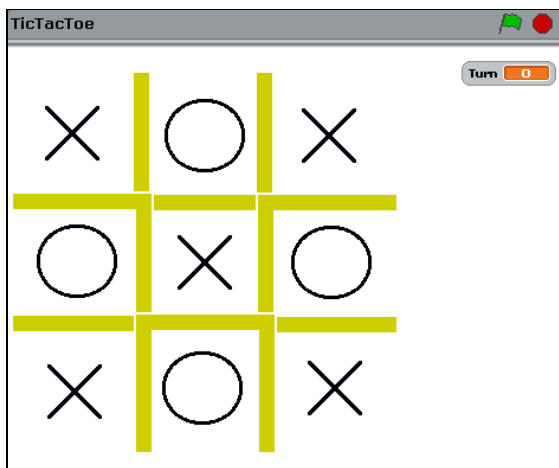
Add different blocks to the program and observe what happens.

Scratch 1.4 Exercises

BY MR. REID

Our Fifth Program: Tic Tac Toe

A classic western civilization children's game, "tic tac toe" is played with a grid of nine places divided by four lines as seen below. The game is played by placing X-tiles or O-tiles in empty squares in the grid. The game is won when a player gets three X-tiles or O-tiles in a row, column or diagonal. The stage is the golden colored grid.



There are nine sprites, one for each square space in the grid. Each sprite has three costumes: an X, an O and a blank (a white square). Each sprite may be called: xo1, xo2, xo3, xo4, xo5, xo6, xo7, xo8, and xo9.

The Visual Stage

Step1: Draw the stage grid. Choose any solid color you like for the lines of the grid. Remember you can change the thickness of the line.

Step2: Draw a single sprite with three costumes as illustrated above.

Step3: Copy (duplicate) the sprite eight (8) times for a total of nine sprites.

Step4: Place the nine (9) sprites, one per grid location.

Notes: The blank "white" square is a required costume. It allows the "when sprite clicked" block to work. A hidden sprite can not be clicked on using the mouse. If you want the background color of the stage to be blue, the color of the blank "white" square must also be blue.

The Programming Stage

In this example we are building a simple two player game. It requires the use of a variable. A variable is a storage location inside the computer's memory. In Scratch we can store numbers or characters inside a variable. In our program, we will store an X or an O (capital letter O, not the number zero). This will enable (allow) our program to remember whose turn is next.



(Fig 1)



(Fig 2)

Notice in figure-1 that the program uses an if-else-block. The “Turn = X” area is the decision area of the block. It checks the condition of variable's contents. If “X” is stored inside the variable called “Turn”, the result of “Turn = X” is true. The blocks between IF and ELSE are executed (run). If “O” is stored inside the variable called “Turn”, then false is the result. In the case of false, the blocks between ELSE and the End of the IF block are executed (run).

Step1: Create the variable called “Turn” (for all sprites), and place the global variable on the stage as shown in the image of the stage area.

Step2: Form the code block as given in figure-2 in the first sprite script area. This sets up the tic-tac-toe game so that it is ready to be played.

Step3: Drag the “when sprite clicked”, block into the script area.

Step4: Drag the if-else-block onto the script area. Set the decision code to evaluate the variable as seen above.

Step5: Drag the two blocks into the “true” section of the if-else-block.

Step6: Set the turn variable to “O”.

Step7: Drag the two blocks into the “false” section of the if-else-block

Step8: Set the turn variable to “X”.

Step9: Copy both blocks of code into each game piece sprite, and test your code, the basic game is complete.