

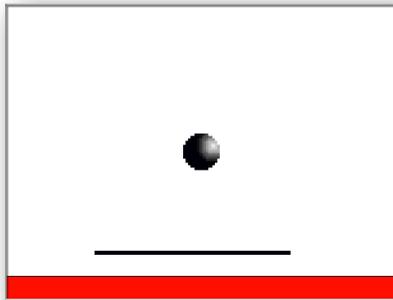
Pong:

A simple pong game written in Scratch.

Stage 1) Start a new project, right click to delete the cat.

Design your pong game with a red area at the bottom to show that if the ball enters this area, it will lose a life. Create a ball sprite using the "Paint New Sprite" option. To make the ball look more three-dimensional, you can use the paint bucket icon to choose a gradient. The circular gradient works well for curved or sphere shaped objects. Create a line-shaped sprite to be used as the pong paddle.

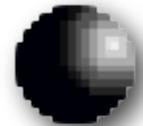
Tip: When drawing sprites, use the magnifying glass with the minus sign on it to zoom all the way out. This will make the drawing screen the same size as your world so that you can tell how large or small your objects actually are, and what size they will appear when you insert them into your world. Zooming in is helpful for adding detail, but when determining the relative sizes of objects, it is helpful to zoom out.



When your world has a ball, a paddle, and the red "dead area" on it, it should look like the picture at the left.

Tip: Name your sprites appropriately, so that the names indicate what the sprites are. This will be helpful when you are writing your program.

Now that your world is designed, you are ready to start your program.



The ball has a gradient to make it look more three-dimensional, but it will work just as well without the gradient if you prefer.

We don't want the program to be running all the time, just when we want to play Pong. Luckily, Scratch makes it easy to decide when your program will be running and when it will not. To start a new block of your program:

Select the ball by clicking on its icon in the bottom right-hand rectangle of the SCRATCH window.

Make sure that the **Scripts** tab is selected.

Click **Control**. Then, find the **when green flag clicked** icon. Drag this icon onto the gray area.



Staying in the control tab, experiment with some of the loops and other items to write a small program that works like the program on the left: when the green flag is clicked, the ball will turn 15° and forever move 10 - 20 steps and bounce if it is on an edge.

Before clicking on the green flag to test the program as shown at the left, first try this program without turning 15° at the beginning. What happens? Why?

After you have experimented with this program, add the “ turn 15° ” block back into the program.

You may notice that the ball turns frequently, making it look as though the light source, which is shining on the ball and creating the gradient on the ball, is moving all over the place. To make it look as though the light source is staying in one place, which is more realistic, select the ball, and at the top near the center of the screen, you will see three buttons that look like this:



Play with these buttons until the gradient stays pointing the same direction.

If you have trouble creating this program, please raise your hand and get assistance before you continue the tutorial.

Stage 2) The next step in the program is to make the player lose a life if they fall in the dead area. In order to do this, we will need to work with variables.

Click to  the tab. Click  on then, make sure the radio button

that says “For all sprites” is selected, and name the variable “livesLeft.” Add a block that gives the player 3 lives at the beginning of the game.

Hint: Giving a variable a particular value is usually called “setting” the variable. In this case, the value of the variable is always equal to the number of lives a player has left at any given time.

We don’t want our pong game to be too hard, but we don’t want it to be too easy either. If we set the number of lives left too high, the player can keep playing pong for a very long time without losing. If we set too few, the player will lose very quickly if (s)he is not a pong expert. When you have finished this stage, play around with the number of lives you give a player to start. You may consider giving a player a random number of lives to keep the game exciting.

Drag in another  add in a yellow  loop under the new “when

green flag is clicked” and use it to instruct the ball to bounce off the paddle if the ball is ever touching the paddle. Think carefully about what happens when something bounces off another thing. After you figure out how to make the ball bounce off the paddle using a number, try using a random number in place of the numbers you used to make the game more difficult. Be careful, though, that you don’t make the numbers so random that your game is unrealistic. Always use reasonable ranges for your random numbers.

Hint: If you can’t figure it out, try looking at the pieces that will fit in the  part of the “forever if”

loop. Look particularly at the pieces under the  tab, but remember, the sensing tools will

only be helpful when determining WHEN this part of the program should run. You still need to write the “bouncing” part of the program. Try using blocks under the Motion tab for this part.

Test your program to be sure it works as you expect. You will not be able to move the paddle, but just watch the ball to make sure that when the ball touches the paddle, it bounces off in a realistic manner. If it does not, try several other ideas before raising your hand to ask for help. Also, consider using random numbers wherever you must type in a number to make the game more exciting and less predictable.

Stage 3) Click on the paddle in the lower right hand rectangle of the SCRATCH window to select the paddle. Choose whether you want to have your user control the paddle with the mouse or with the arrow keys on the keyboard.

If you choose mouse:	If you choose arrow keys:
<p>Select the paddle. Make sure you are on the "Scripts" tab. Drag in a new green flag. Using a forever loop, search through the tabs to find a block that will allow you to set the x coordinate of the paddle to the x coordinate of the mouse.</p>	<p>Select the paddle. Drag in a new green flag. Using a forever loop, search through the tabs to find a block that will allow you to tell the paddle to move 10 steps when the right key is pressed. Add another green flag with a forever loop and a block that will allow you to tell the paddle to move -10 steps when the left arrow key is pressed. Make sure it bounces off the edges, otherwise the user could lose their paddle off the side of the screen.</p>

Challenge: Experiment with setting the x coordinates of the paddle. Can you make a pong game that will play itself and never lose? How can you do that? Try it and see if it works.

If you have trouble with this part of the tutorial, keep trying for at least 10 minutes before raising your hand to ask for help-- you will find the solution eventually, but some of the information you need is intentionally left out of this handout because you will remember the things you discover on your own better than the things you just read on this handout. When you start writing your own programs with Scratch, you'll be better off if you have found these things yourself because you will remember where they are. Feel free to make notes on this handout once you figure things out because those notes will be helpful in the future. If you continue to have trouble, feel free to ask for help -- once you have completed the tutorial, the last page of this Pong handout will be given to you. It contains the key and you'll be able to check your work or fix any errors.

Stage 4) Click on the ball sprite so that the paddle sprite's Scripts tab shows up in the middle of the window. Drag in a new "when green flag is clicked" tile. Use the following program to decrease the number of lives left by one each time the ball is touching the dead area.



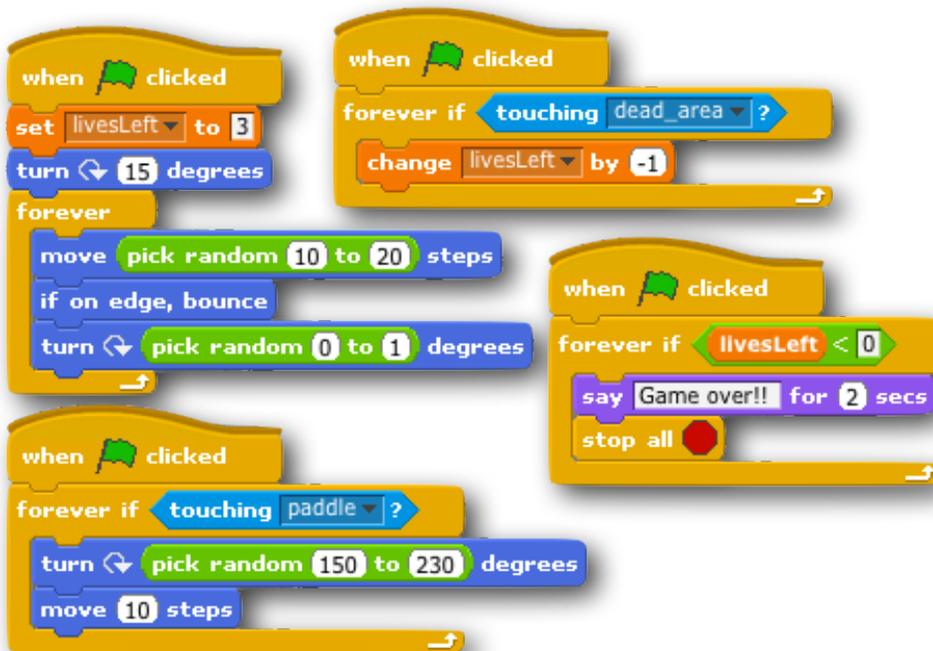
Write a program of your own to tell the game to end when there are no lives left.

Check your work:

On the paddle's scripts tab:



On the ball's scripts tab:



What I did differently and why: _____

What now?

Customize your Scratch game however you want. You can work alone or in a group if you like. Some ideas are:

- Add in “blocks” or rectangular shaped sprites to create a brick blaster where the object is to blast all the bricks using the ball before you run out of lives.
 - You may want to have the bricks disappear permanently when they disappear, or, to make the game more difficult, have them reappear after a few seconds. Then, make the player win when they have the screen totally clear of bricks at any given time.
 - Make a two player hockey game where one player uses the keypad arrows and the other uses the mouse to control separate paddles. Make a scoreboard which increments the points each player has whenever the ball touches the goal of the other player.
 - Make the ball change colors and make a noise every time it hits a wall. Then, create some other sprites in your world, which look like musical instruments. Using the “Sound” tab, tell the ball to make a sound like the musical instrument it hits as it hits each musical instrument. Use your paddle to create music. If you like, remove the variable “livesLeft” to allow for an endless game.
 - Make the ball draw a line everywhere it goes and make the line change colors gradually. Then use the paddle to create art. If you like, remove the variable “livesLeft” to allow for an endless game.
 - Or create your own ideas:
-
-
-

If you finish early:

- Create a Pac man game using scratch
- Create a car racing game where the player looks down on the track and uses the arrow keys to steer. Up arrow accelerates, down arrow brakes, right arrow turns right, and left arrow turns left. Outside the track, and in the middle of the track, make a green color to represent grass, then to make the car slow down when it runs on the grass.
- Create a game like Oregon Trail or another role-playing game.

You can download Scratch for your home computer. It is available for Mac, PC, and Linux. It is available for free at: <http://scratch.mit.edu>

Many video tutorials can be found through a simple Google search. SCRATCH is a great way to learn programming.

If you want to learn even more about programming, Carnegie Mellon’s ALICE is also a fun way to learn programming. Last year’s GEMS workshop attendee’s used ALICE to make three-dimensional animations, just like in Disney films like *Tangled*.

ALICE is available for free download from Carnegie Mellon at <http://www.alice.org>
