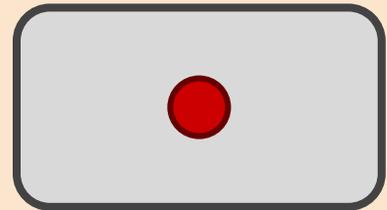


Coding in Scratch - Chapter 1

Created by thegamebox.ca

Coding is like giving instructions to a computer. Just like you tell your friends what to do, you can tell a computer what to do by writing special words called code. The computer then follows these instructions to do things like play games, make pictures, or even send emails. It's a fun and creative way to make technology work for you!

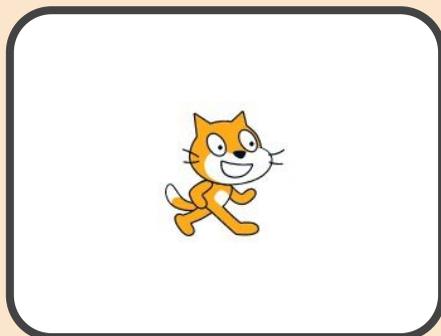
```
let ball = 0;  
ball.draw();
```



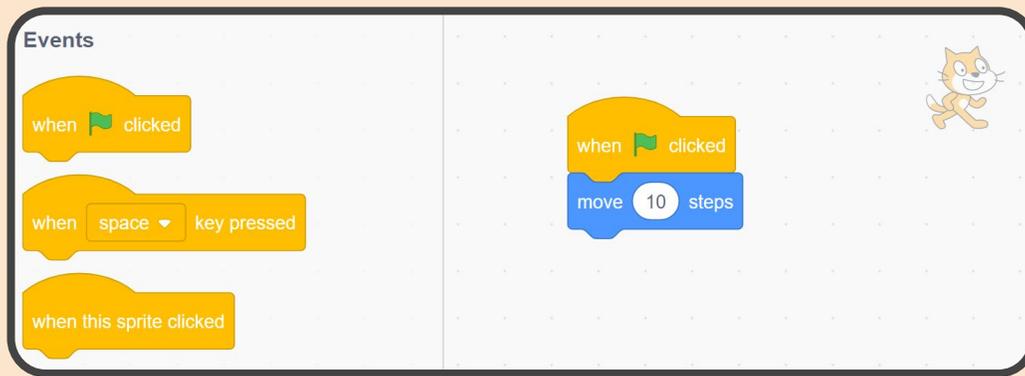
Before we can get into anything complicated, we have to understand the program that we are using to write our code: Scratch. Scratch allows us to create sprites, and add code to them, which tells them what to do, for example to move around the screen, or play a sound. To create a project in scratch go to the following link:

<https://scratch.mit.edu/projects/editor/>

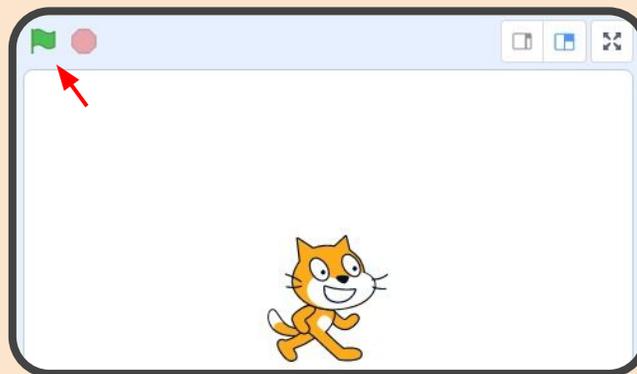
When you open the project, you will see a box on the right, this is called the stage, it's where you'll see all the sprites that you'll code.



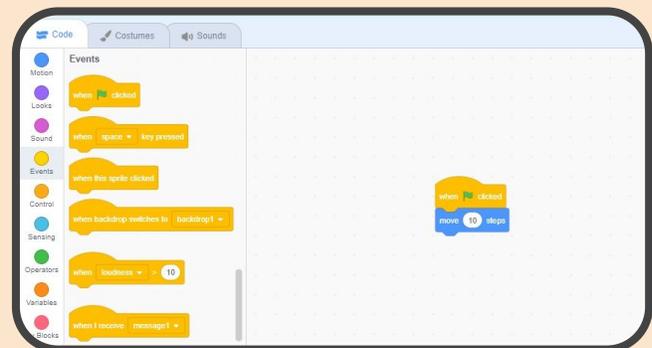
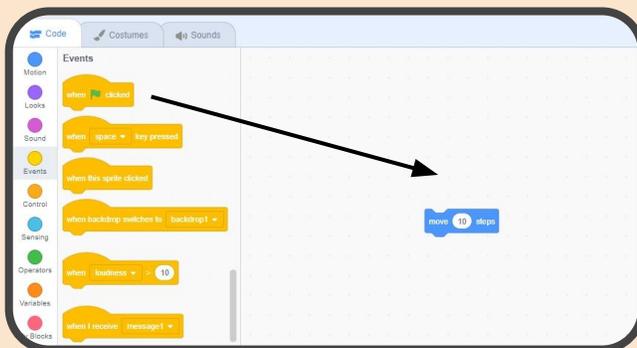
Scratch has added a sprite for you to start with: Scratch Cat! To add some code to Scratch Cat, you can drag the colored blocks from the side of the screen into the workspace. Let's start with the "move 10 steps" block.



Now what do you think will happen when we tell Scratch Cat to move ten steps? Of course! He'll move ten steps in the direction that he's looking. But Scratch Cat isn't going anywhere yet, the code doesn't run until you press the green flag at the top of the stage.

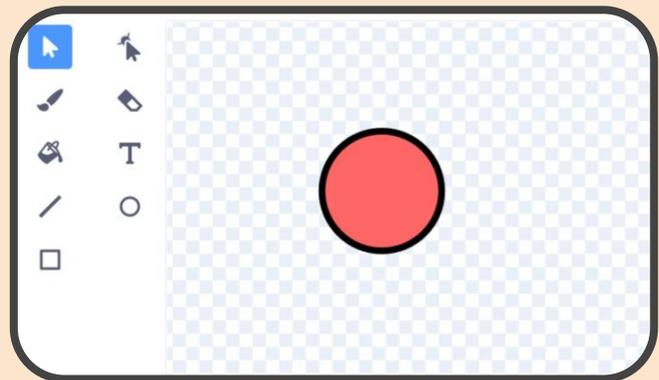
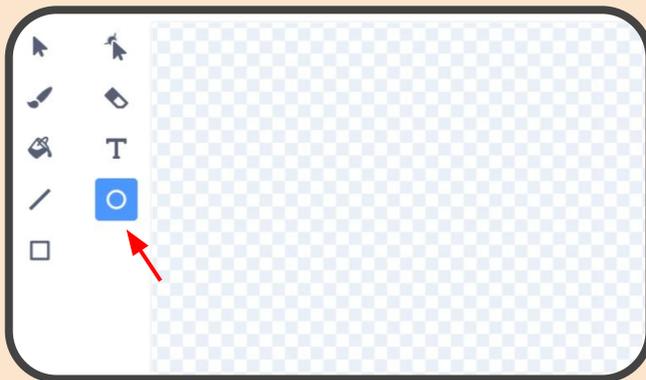
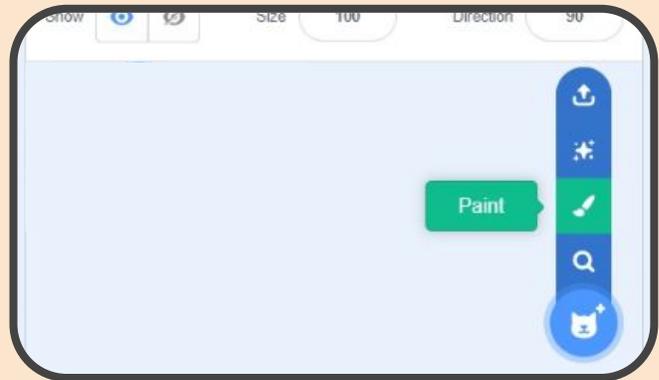
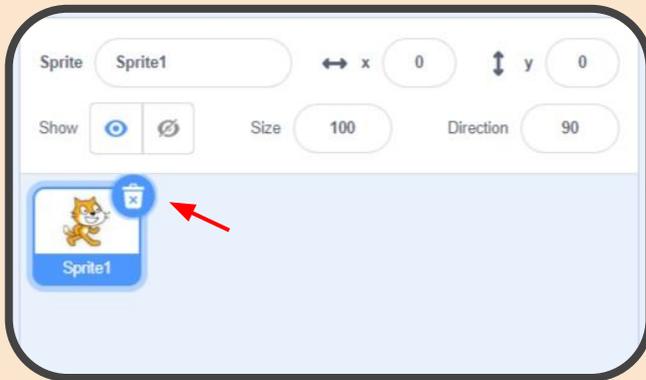


But Scratch Cat still isn't moving! This is because we told him to move ten steps, but we didn't say when to move. Go to the "Events" section of the blocks, and add the "When flag clicked" block above the "move 10 steps" block.

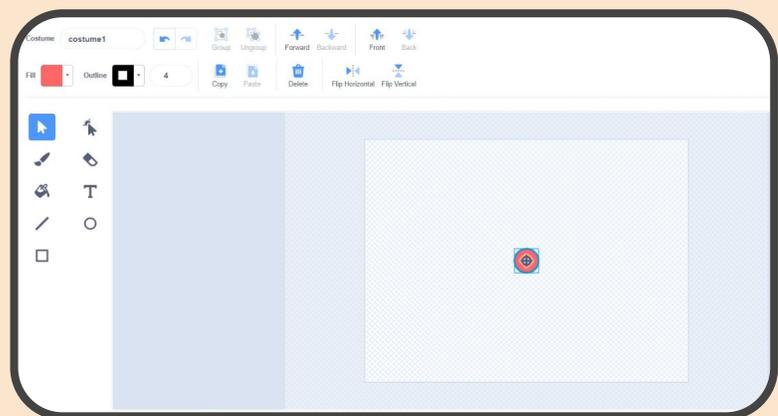


Now we can press the green flag, and see Scratch Cat move across the screen!

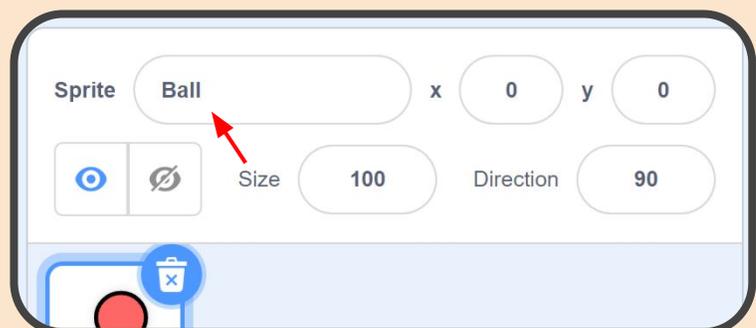
Now that you see how we can add code to a sprite to make it do things, let's try making a pong game. First delete Scratch Cat, we won't be needing him anymore. Then make a new sprite, let's use Scratch's sprite editor to make a ball for our game.



Also, you should make sure to center your circle so it doesn't look weird when it bounces around.



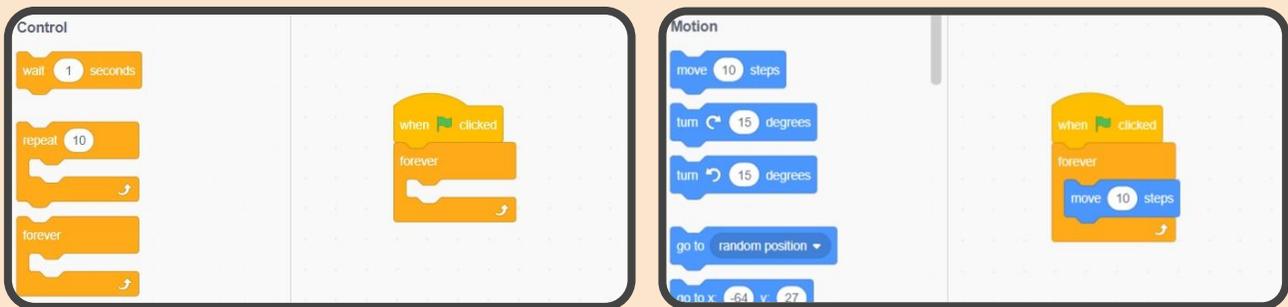
You should also change the name of the sprite to "Ball".



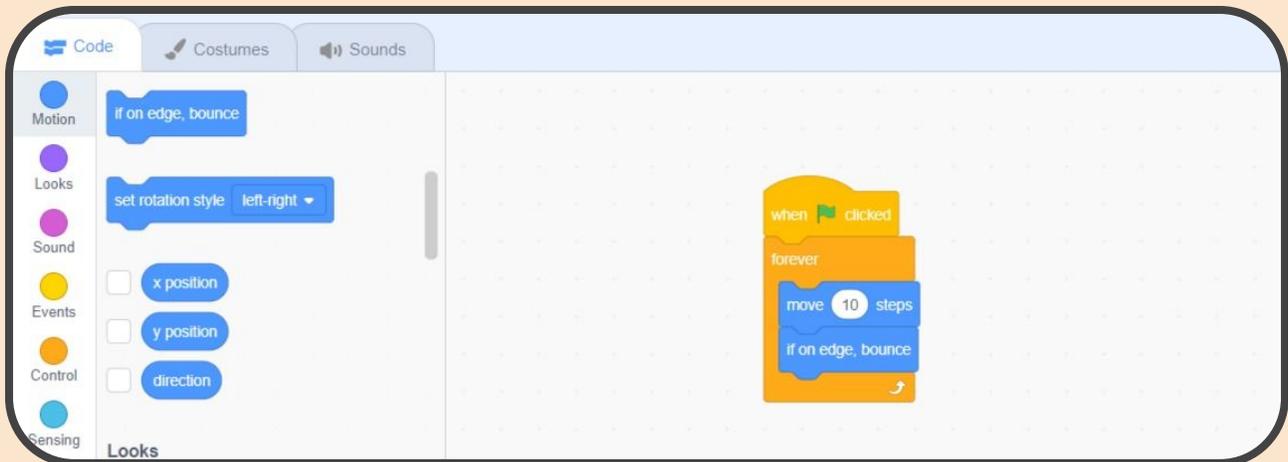
Now let's add some code to our ball, we want it to bounce around the stage forever. We can do this with the "forever" block, whatever you put inside will repeat until you stop the game.



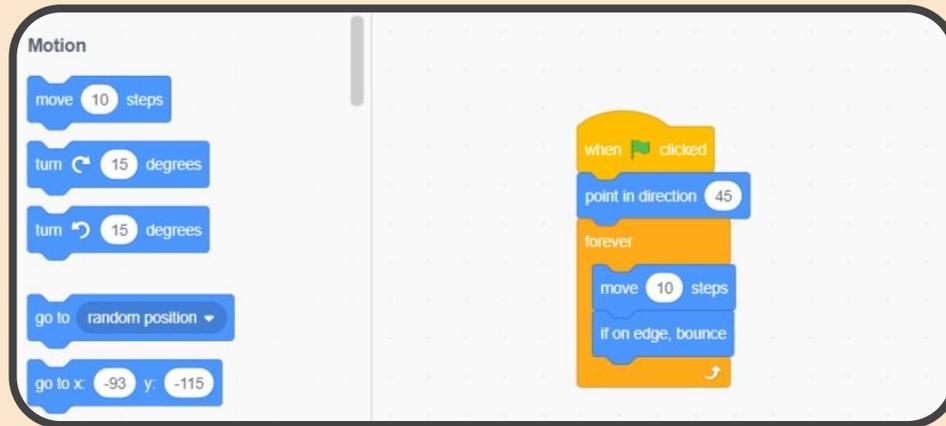
To make the ball move around the screen, we can add a "move 10 steps" block in the forever loop, which will keep it moving forever.



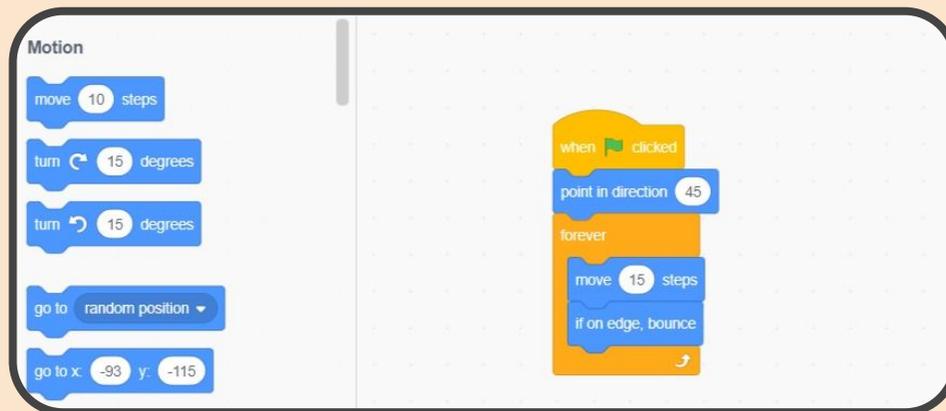
We also want the ball to bounce if it hits the edge of the stage, so we can add the "if one edge, bounce" block into our forever loop. We have to make sure to put it in the forever block, so that it check if it's on the edge everytime it moves, and not just once.



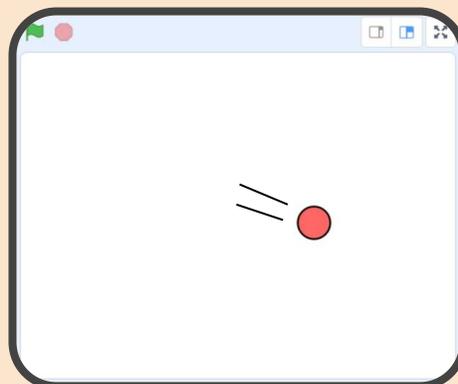
To make the ball start a little bit of an angle, let's put a "point in direction" block at the start of the code, and change the direction to 45.



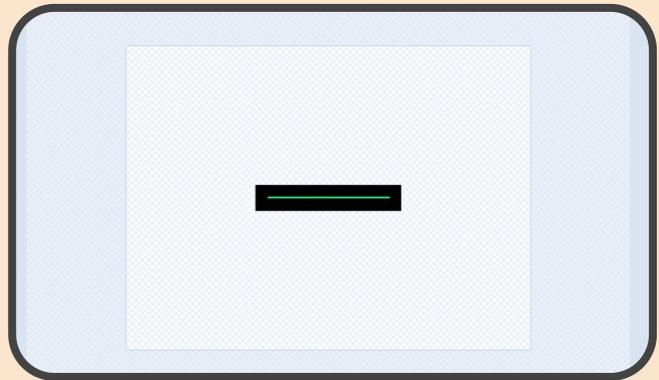
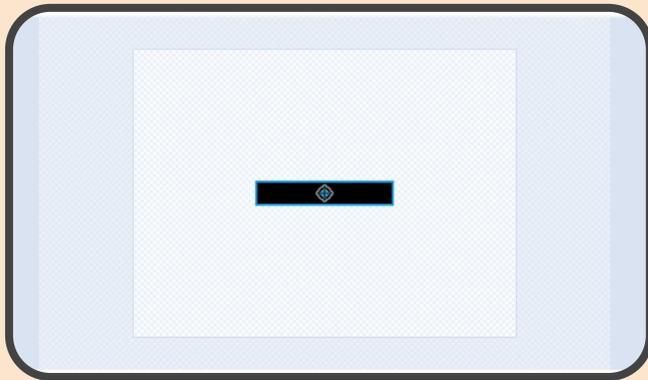
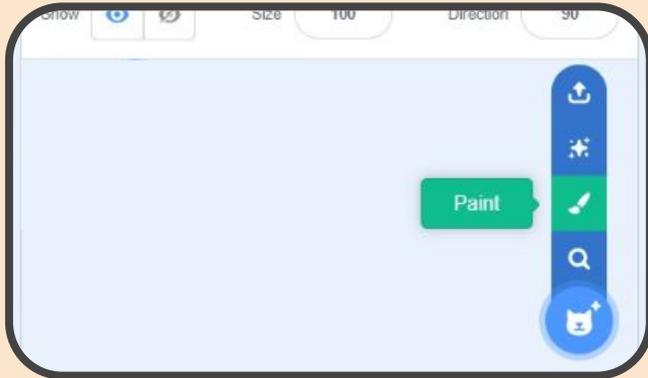
Finally, let's change our "move 10 steps" block to "move 15 steps", so that the ball goes a little faster.



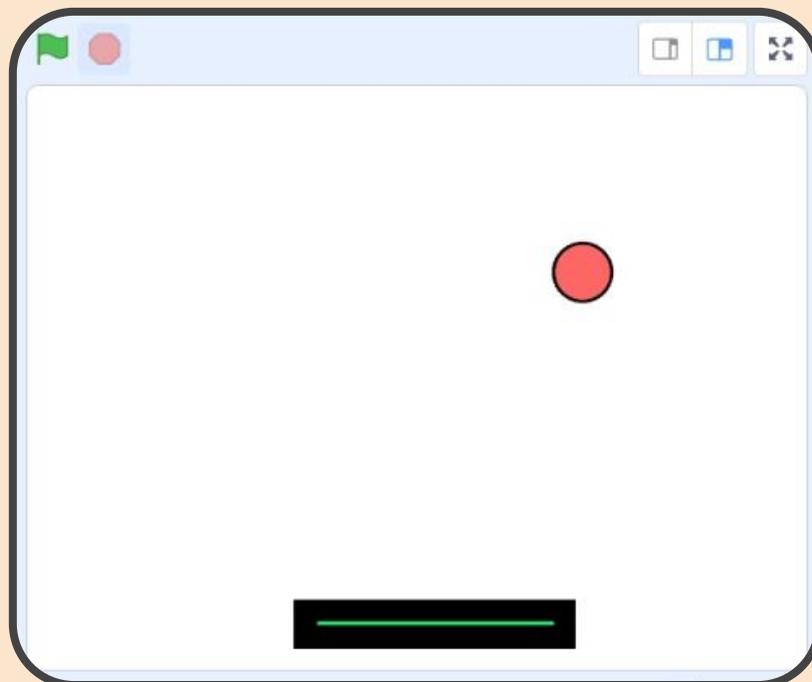
Great, now we can run our code, and see the ball bouncing around the stage!



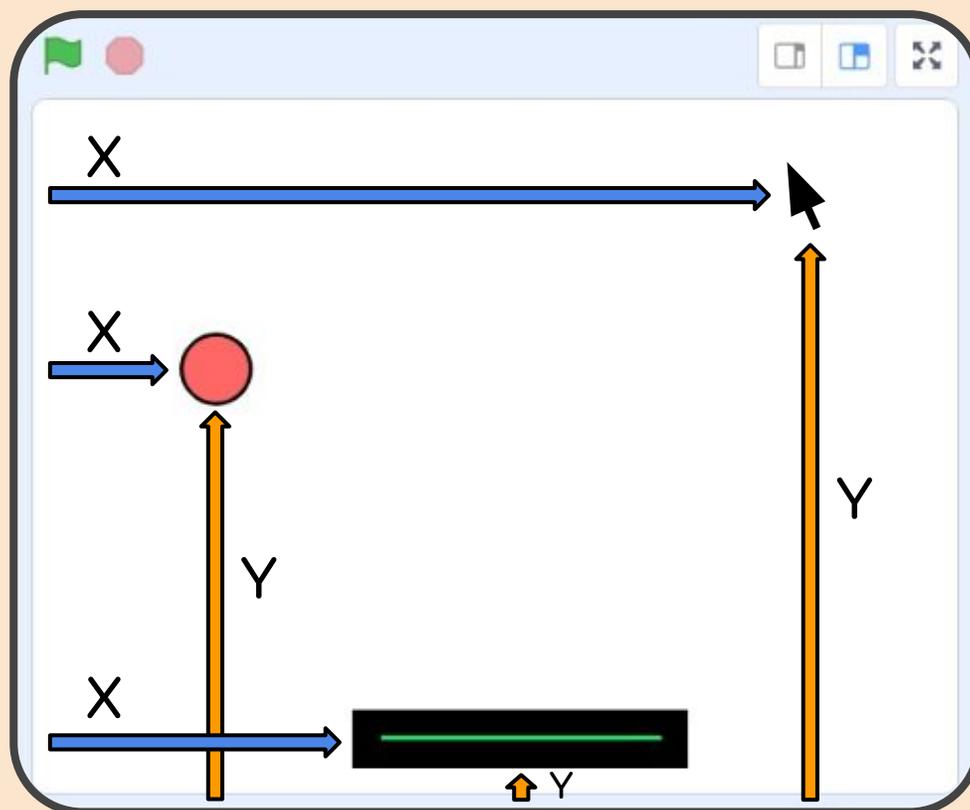
Now let's make a paddle that we can control to block the ball. Create a new sprite and call it "Paddle". Draw a rectangle for the sprite, you can also add a design if you want. Make sure to center this one too.



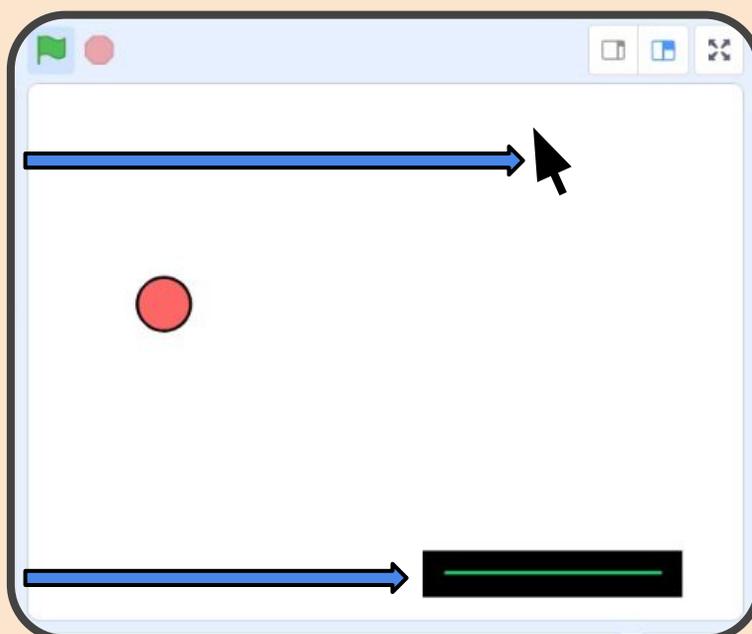
Now let's move the sprite to the bottom of the stage.



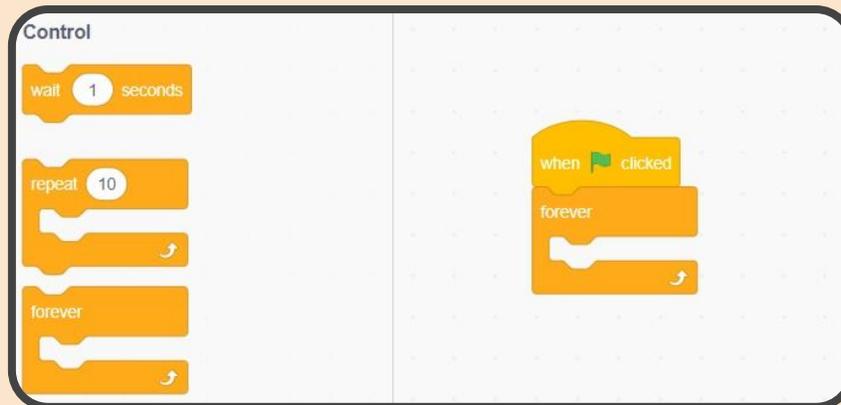
Now before we can make our paddle move around, we have to understand x position and y position. A sprite's x position is how far across the width of the screen, and it's y position is how far it is across the height of the screen.



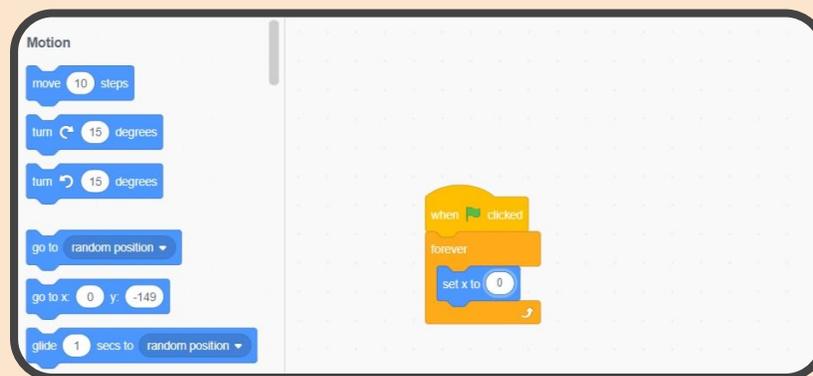
As you can see in the picture, the mouse also has x and y positions, so if we make the x position of the paddle the same as the x position of the mouse at all times, we can control the paddle with the mouse.



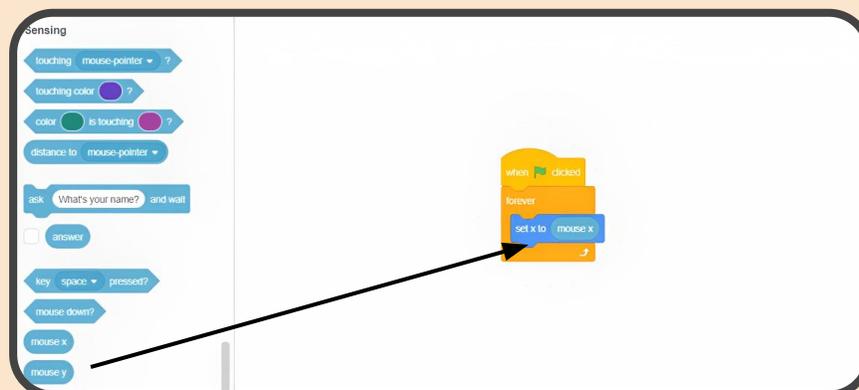
To do this, let's make a new forever loop in the paddles code, we want a forever loop because the paddle should *always* be where your mouse is.



Scratch has a block for setting the x position of a sprite, so let's add it into the forever loop.



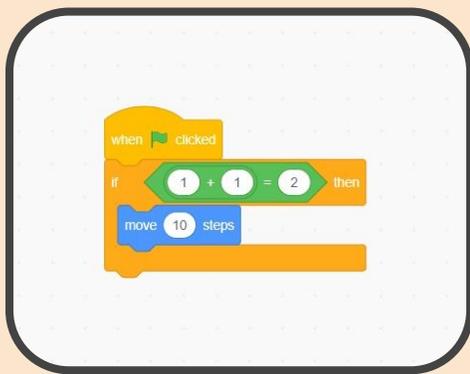
But we don't want the x position to be zero, we want it to be the mouse's x position. Luckily, Scratch has a small block that we can put inside the "set x to" block to replace the zero. It's in the sensing category.



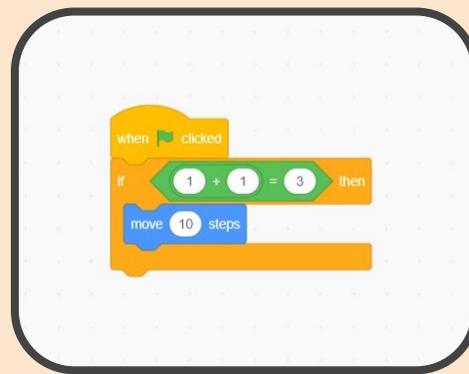
Perfect, now our paddle follows the mouse, all we have to do now is make the ball bounce when it hits the paddle. To do this, we have to learn about the if block.



When the if block is used, it checks if what is in the hexagon is true, and if it is, then it does whatever you put inside the if block. For example, in picture A, since one plus one equals two, it would move ten steps, but in picture B, it wouldn't move ten steps, because one plus one does not equal three.

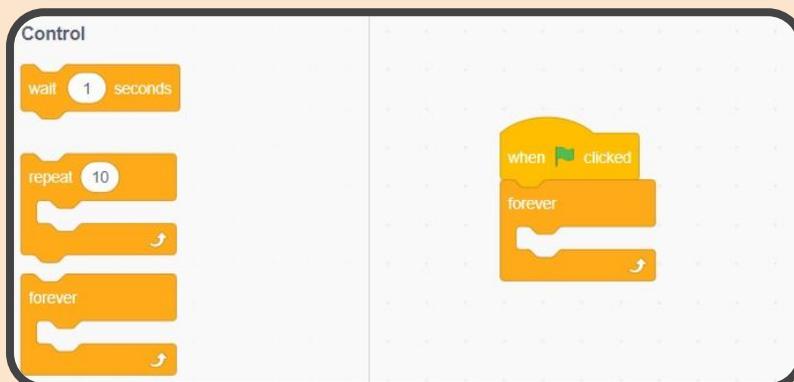


A

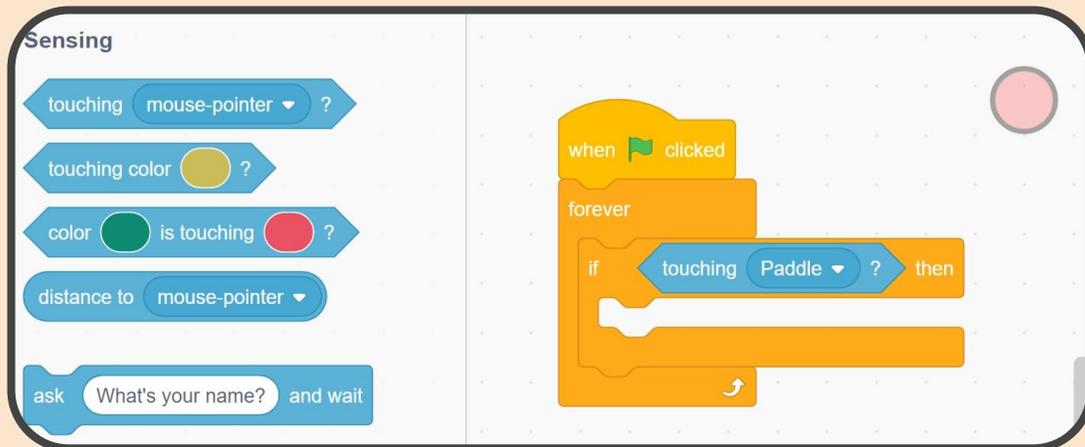


B

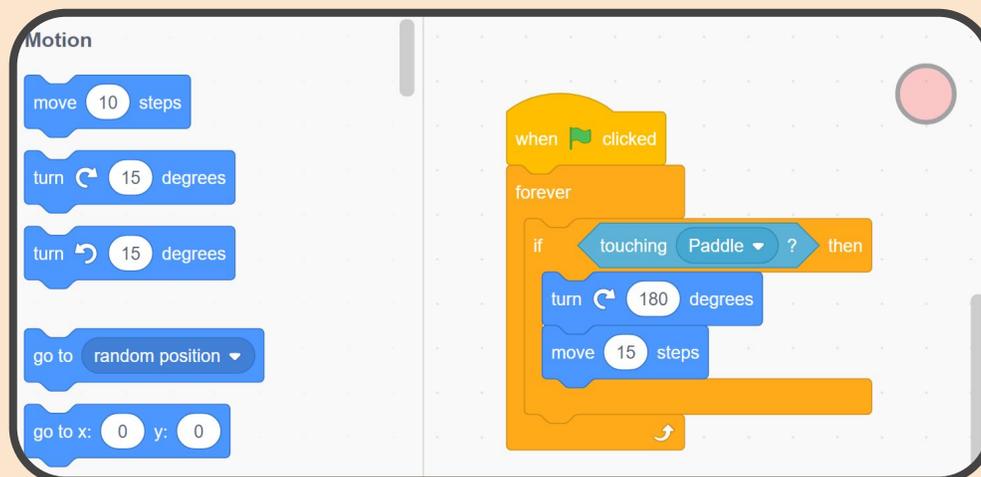
So in the code for the ball let's make a new forever loop, we want to use a forever loop so that it always checks if it's touching the paddle, and not just at the start of the game.



Inside the forever loop, put an if block, and inside the if block's little hexagon, we can put the "touching ____" block from the sensing category, and change whatever is in the blank to the name of your paddle sprite, mine is named Paddle. The entire if block reads out to be "if touching Paddle" which is exactly what we want to check.



So what should the ball do when it hits the paddle? Turn around and move away. To make something turn all the way around, you can turn it 180 degrees, so let's do that using the "turn ___ degrees" block, and change it to 180. Then let's add a "move 10 steps" block, and change it to 15 steps.



And just like that, we have created a working game in Scratch! Of course, there is much more that we could do with this, we could make the ball go faster as the game goes on, or add a score counter, but for now, it's a pretty fun game! See if you can last longer than your friends!