# K2L Pong Game

## Today we are going to be making a “Pong” game. It is played by having two paddles on the screen, hitting a ball back and forth like a game of ping pong. The game is going to be 2-Player, although you can feel free to add a computer player if you would like. If the instructions are unclear, raise your hand and I will try to help!

**Steps:**

1. Open up the “mycode.js” file and find the spot labeled “Step 1”. Delete this line and replace it with 2 variables to represent the first players X and Y position
   1. Example:  
      var p1X = 0;  
      var p1Y = 0;
2. Now find “Step 2” and do the same, but for player 2.
3. Now where it says “Step 3” remove the line, and replace it with the X and Y coordinates for a ball, as well as direction.
   1. Example:  
      var ballX = 0;  
      var ballY = 0;  
      var ballYDirection = 0;  
      var ballXDirection = 0;
4. Now make some final variables for your players. We need the paddle widths and heights, as well as 2 colors for your players. You can change these if you want, or use the following code.
   1. Example:  
      var paddleWidth = 30;  
      var paddleHeight = 100;  
      var player1Color = “black”;  
      var player2Color = “black”;
5. Now we move on to drawing. Find “Step 5” and replace it with the following code
   1. c.fillRect(p1X, p1Y, paddleWidth, paddleHeight);
   2. The above code draws your player1 on the screen, at the x and y coordinates
6. Now do the same thing for step 6, but for player two. You shouldn’t notice any changes
7. Now make it so both paddles are on opposite sides of the board. Player one should be on the left, and Player 2 should be on the right. Change the X coordinates and Y coordinates until it looks right.
   1. I recommend using the following to get started:
   2. var p1X = 5;  
      var p1Y = 5;  
      var p2X = Canvas.width-30-5;  
      var p2Y = 5;
8. Refresh your game and make sure you can see both paddles on the screen. Now we are going to make them move.
9. Inside of your keyPressed methods (at the bottom of the page) add code to make the paddles move.
   1. Use W/A (Up/Down) For Player One, and I/K for Player Two (These are new methods at the bottom of “mycode.js”
   2. Example:  
      function upPressed(){  
       p1Y = p1Y – 10;   
      }
10. Now that we have the paddles moving, lets get the ball moving. Inside of your draw method, add code to draw a rectangle using the ball’s X and Y and a width and height of 15 where it says “Step 10”
    1. The ball should be in the top left corner. You should instead place it in the center by changing the values at the top for ballX and ballY
       1. Example  
          var ballX = Canvas.width/2-8;
11. Next, go into the update method. This is where the ball is going to move. For example, you could add 1 to the ballX every update, and the ball will “float” across the screen (try it!)
12. For this, we are going to want the ball to start by moving to the top right corner. To do this, start by setting the ballDirectionX to 1 and ballDirectionY to -1 at the top of your code
13. Now inside the “Update” method, where it says “step 13” add ballDirectionX and ballDirectionY to the ballX and ballY variables, example below:  
      
    ballX = ballX + ballDirectionX;  
    ballY = ballY + ballDirectionY;
14. Did it move?? If it didn’t make sure you have everything named correctly and ask me for help by raising your hand.
15. Now if you let the ball continue to move, it will move off the screen. In order to stop that, we have to add some **Collision Detection**. Basically you have to tell the game what to do when the ball hits the paddles, or leaves the board. This part is the hardest, so try to follow along as closely as you can.
16. Inside of the collisionDetection method, lets make it so that if the ball hits the borders of the game, it switches direction. Therefore never leaving the board!
    1. First, add code to check for the top of the board!  
       if(ballY < 0) ballDirectionY = 1;
    2. Next, code to check for the right of the board!  
       if(ballX > Canvas.width-15) ballDirectionX = -1;
    3. Next, for the Bottom!  
       if(ballY > Canvas.height-15) ballDirectionY = -1;
    4. Left!  
       if(ballX < 0) ballDirectionX = 1;
17. The ball should now move around the board and bounce off the walls!
18. Next we have to make it so they bounce of the paddles. To do this, use the following code:  
    if(ballX >= p2X-15 && ballY > p2Y && ballY < p2Y + paddleHeight) ballDirectionX = -1;  
    if(ballX <= p1X+paddleWidth && ballY > p1Y && ballY < p1Y + paddleHeight) ballDirectionX = 1;
19. What this does, is makes the balls bounce off the front of each paddle. It’s not perfect detection, but you should be able to play a rough game of pong! Look below for things you can add!
20. Things you can add to your game
    1. Score
    2. Points for whenever the ball hits the sides of the board
    3. Have the ball go a random direction each time!
    4. Increase the speed of the ball! (Hint: Change values in the update method!)
    5. Add colours!
    6. Add 2 more players on the top and bottom of the board!
    7. Add multiple balls!
    8. Have the ball reset when it hits a side of the board!