

Functions and Loops



What we did:

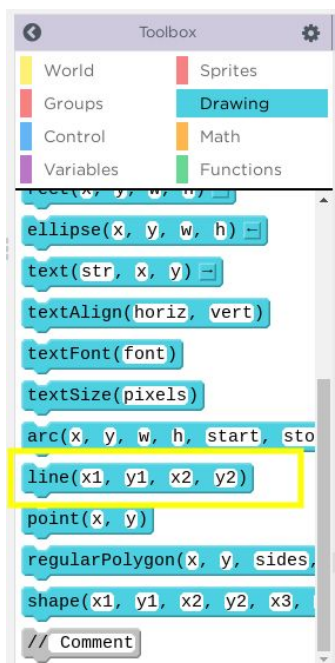
- Write custom functions to serve the ball, reset the ball and draw the net.
- Draw the net using line instruction and for-loop

Note: In coding, we have a principle D-R-Y : Don't Repeat Yourself.

Remember: Good Programmers don't like to repeat themselves while writing code

How we did it:

Step 1: Use a pre-defined instruction(a function) line()



Step 2: Draw a line by giving the start and the end coordinates

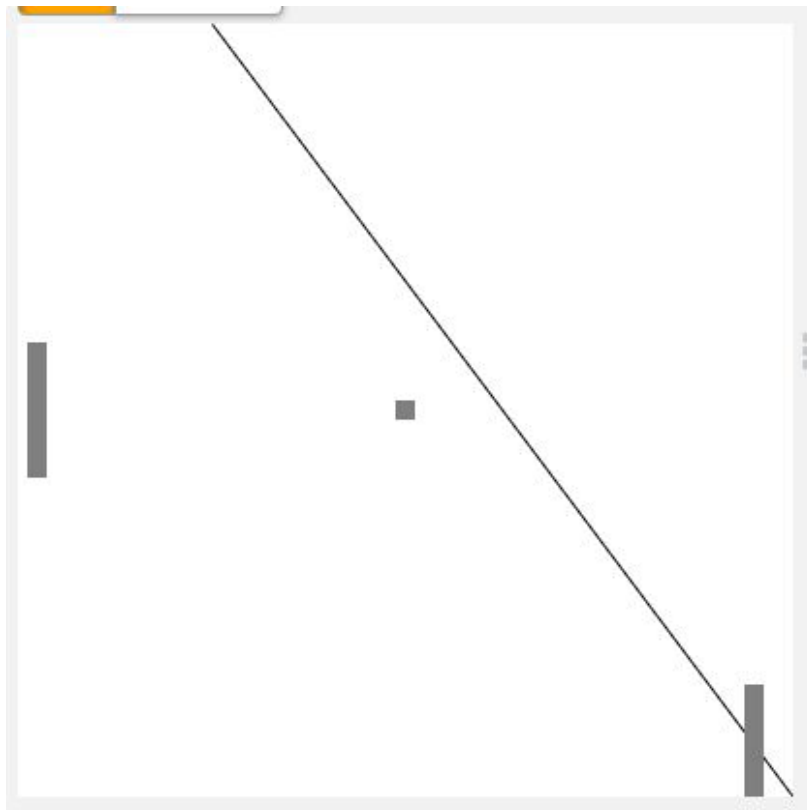
Code:

```

2 var ball = createSprite(200,200,10,10);
3 var playerPaddle = createSprite(380,200,10,70);
4 var computerPaddle = createSprite(10,200,10,70);
5
6
7
8 function draw() {
9   //clear the screen
10  background("white");
11
12  //make the player paddle move with the mouse's y position
13  playerPaddle.y = World.mouseY;
14
15  //AI for the computer paddle
16  //make it move with the ball's y position
17  computerPaddle.y = ball.y;
18
19
20  line(100, 0, 400, 400);
21
22  //create edge boundaries
23  //make the ball bounce with the top and the bottom edges
24  createEdgeSprites();
25  ball.bounceOff(topEdge);
26  ball.bounceOff(bottomEdge);
27
28  //make the ball bounce off the paddles
29  ball.bounceOff(playerPaddle);
30  ball.bounceOff(computerPaddle);
31

```

Output:



Step 3: Make several small (dashed) lines with height to be 10 and leave a gap of 10 after every dash.

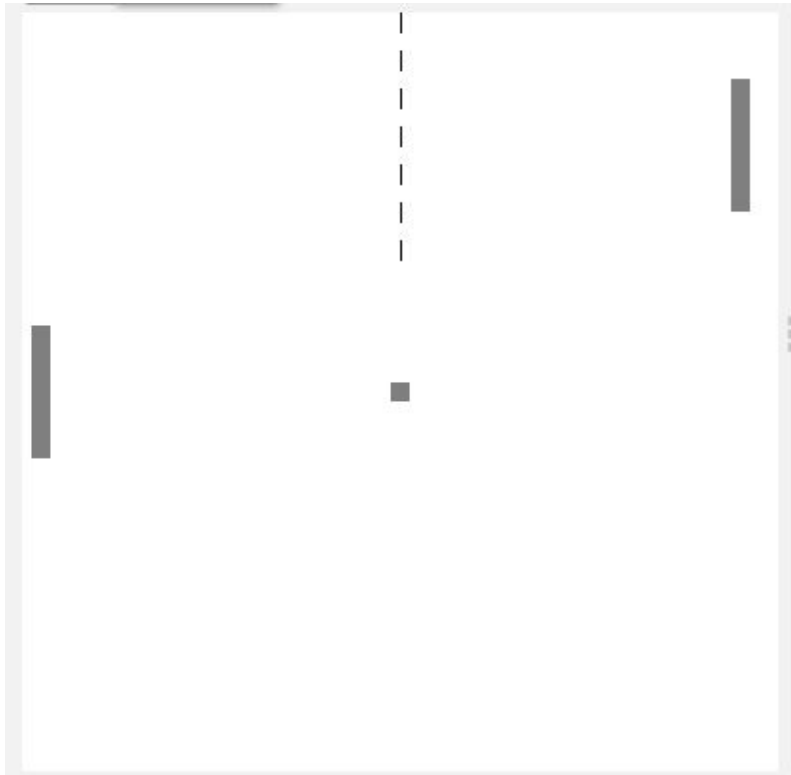
Code:

```

12 //make the player paddle move with the mouse's y position
13 playerPaddle.y = World.mouseY;
14
15 //AI for the computer paddle
16 //make it move with the ball's y position
17 computerPaddle.y = ball.y;
18
19
20
21 line(200,0,200,0+10);
22 line(200,0+20,200,0+20+10);
23 line(200,0+20+20,200,0+20+20+10);
24 line(200,0+20+20+20,200,0+20+20+20+10);
25 line(200,0+20+20+20+20,200,0+20+20+20+20+10);
26 line(200,0+20+20+20+20+20,200,0+20+20+20+20+20+10);
27 line(200,0+20+20+20+20+20+20,200,0+20+20+20+20+20+20+10);
28
29 //create edge boundaries
30 //make the ball bounce with the top and the bottom edges
31 createEdgeSprites();
32 bounceOff(topEdge, bottomEdge, playerPaddle, computerPaddle);
33
34
35 //serve the ball when space is pressed
36 if (keyDown("space")) {
37     ball.velocityY = 3;
38     ball.velocityX = 4;
39 }

```

Output:



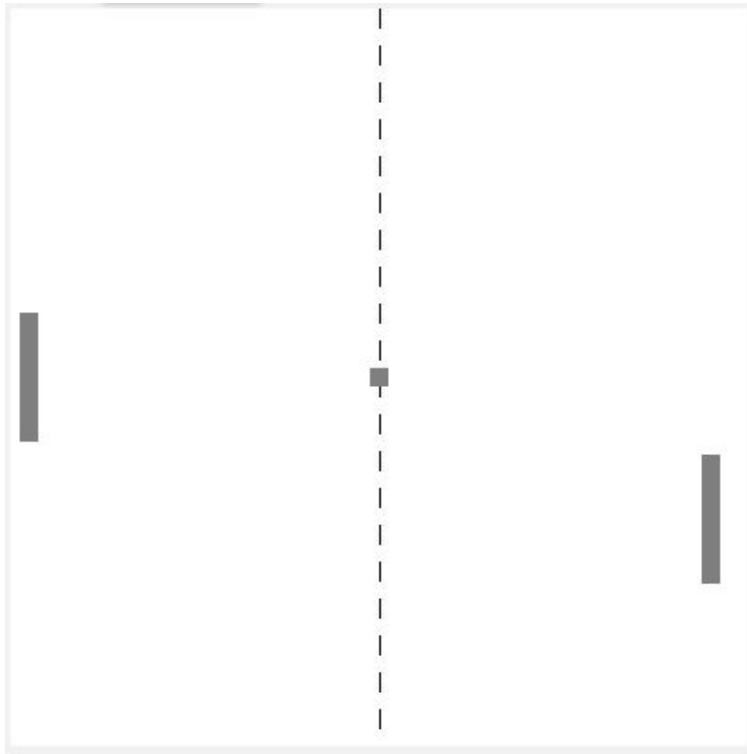
Step 4: Using For Loop to run the same instructions without repeating the code Code:

```

12 //make the player paddle move with the mouse's y position
13 playerPaddle.y = World.mouseY;
14
15 //AI for the computer paddle
16 //make it move with the ball's y position
17 computerPaddle.y = ball.y;
18
19 for (var num = 0; num < 400; num = num +20) {
20   line(200,num, 200, num+10);
21 }
22
23
24
25 //create edge boundaries
26 //make the ball bounce with the top and the bottom edges
27 createEdgeSprites();
28 bounceOff(topEdge, bottomEdge, playerPaddle,computerPaddle);
29
30
31 //serve the ball when space is pressed
32 if (keyDown("space")) {
33   ball.velocityY = 3;
34   ball.velocityX = 4;
35 }
36
37
38 //reset the ball to the centre if it crosses the screen
39 if(ball.x > 400 || ball.x <0) {
40   ball.x = 200;
41   ball.y = 200;
42 }

```

Output:



Step 5: Teaching the computer to draw the net using a custom-defined function `drawnet()`.
 Code:

```

8- function draw() {
9   //clear the screen
10  background("white");
11
12  //make the player paddle move with the mouse's y position
13  playerPaddle.y = World.mouseY;
14
15  //AI for the computer paddle
16  //make it move with the ball's y position
17  computerPaddle.y = ball.y;
18
19  drawnet();
20
21- for (var num = 0; num < 400; num = num +20) {
22   | line(200,num, 200, num+10);
23 }
24
25 |
26
27 //create edge boundaries
28 //make the ball bounce with the top and the bottom edges
29 createEdgeSprites();
30
31 //serve the ball when space is pressed
32- if (keyDown("space")) {
33   | ball.velocityY = 3;
34   | ball.velocityX = 4;

```

Step 6: Teaching the computer to serve the ball and reset the ball by writing custom-defined functions.

```
34     resetball();
35 }
36
37     ball.bounceOff(topEdge);
38     ball.bounceOff(bottomEdge);
39     ball.bounceOff(playerPaddle);
40     ball.bounceOff(computerPaddle);
41
42     drawSprites();
43 }
44
45 function drawnet() {
46     for (var num = 0; num < 400; num = num +20) {
47         line(200,num, 200, num+10);
48     }
49 }
50
51 }
52
53 function serveball() {
54     ball.velocityY = 3;
55     ball.velocityX = 4;
56 }
57
58 function resetball() {
59     ball.x = 200;
60     ball.y = 200;
61     ball.velocityX = 0;
62     ball.velocityY = 0;
63 }
```

What's next? :

Understanding the different states of a game and how to store information about the in-game states.