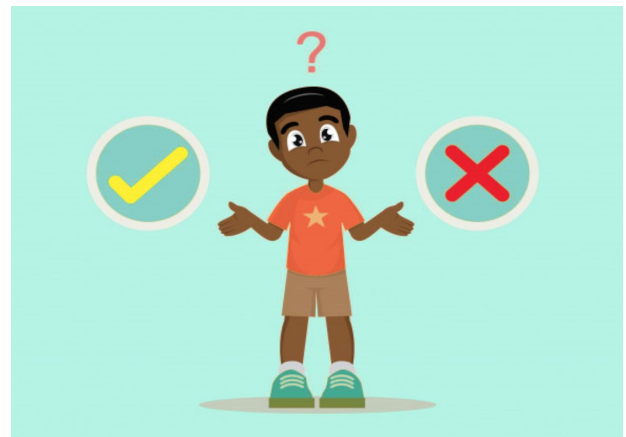


Writing functions which can take arguments

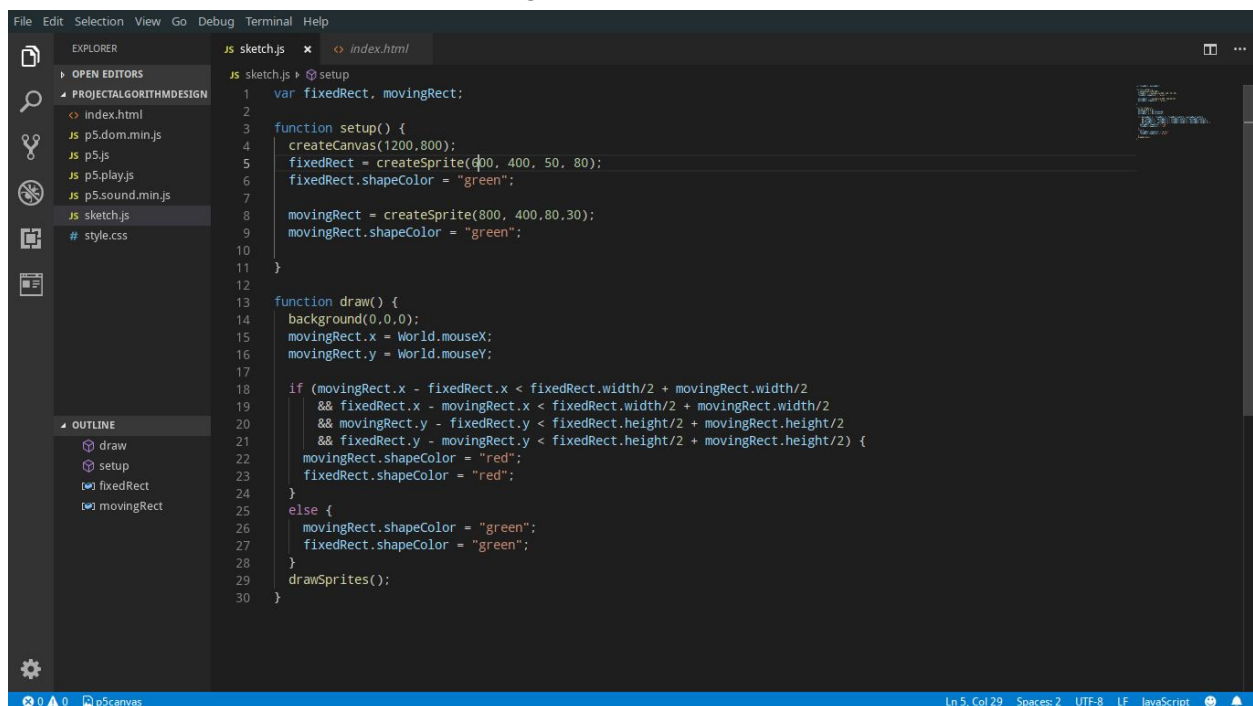


What we did:

- Learn "true" and "false" as the two boolean values
- Write a function which can accept arguments, return values and can be re-used for the different game objects.
- Create a little code library and use it within the code.

How we did it:

Step 1: We have to write a big series of if-else condition inside the code— code can detect collision between fixedRect and movingRect



```

File Edit Selection View Go Debug Terminal Help
EXPLORER
PROJECTALGORITHMDESIGN
  index.html
  p5.dom.min.js
  p5.js
  p5.play.js
  p5.sound.min.js
  sketch.js
  style.css
OUTLINE
  draw
  setup
  fixedRect
  movingRect

Js sketch.js x index.html
Js sketch.js > @ setup
1 var fixedRect, movingRect;
2
3 function setup() {
4   createCanvas(1200,800);
5   fixedRect = createSprite(400, 400, 50, 80);
6   fixedRect.shapeColor = "green";
7
8   movingRect = createSprite(800, 400,80,30);
9   movingRect.shapeColor = "green";
10 }
11
12
13 function draw() {
14   background(0,0,0);
15   movingRect.x = World.mouseX;
16   movingRect.y = World.mouseY;
17
18   if (movingRect.x - fixedRect.x < fixedRect.width/2 + movingRect.width/2
19     && fixedRect.x - movingRect.x < fixedRect.width/2 + movingRect.width/2
20     && movingRect.y - fixedRect.y < fixedRect.height/2 + movingRect.height/2
21     && fixedRect.y - movingRect.y < fixedRect.height/2 + movingRect.height/2) {
22     movingRect.shapeColor = "red";
23     fixedRect.shapeColor = "red";
24   }
25   else {
26     movingRect.shapeColor = "green";
27     fixedRect.shapeColor = "green";
28   }
29   drawSprites();
30 }
  
```

Step 2: Write a function called isTouching. Modify the code to create a function called isTouching()

```

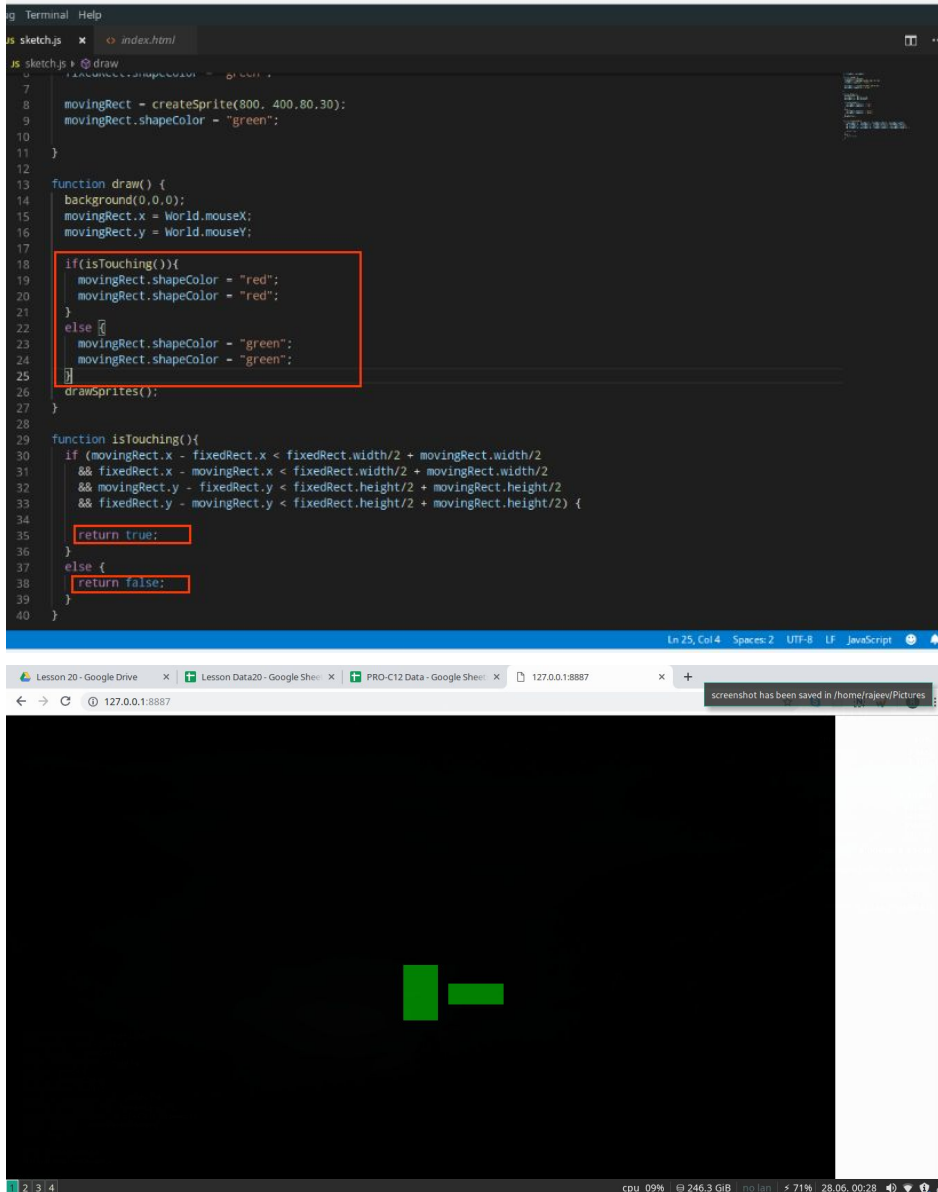
js sketch.js ▶ isTouching
9   movingRect.shapeColor = "green";
10
11 }
12
13 function draw() {
14   background(0,0,0);
15   movingRect.x = World.mouseX;
16   movingRect.y = World.mouseY;
17
18   if (movingRect.x - fixedRect.x < fixedRect.width/2 + movingRect.width/2
19       && fixedRect.x - movingRect.x < fixedRect.width/2 + movingRect.width/2
20       && movingRect.y - fixedRect.y < fixedRect.height/2 + movingRect.height/2
21       && fixedRect.y - movingRect.y < fixedRect.height/2 + movingRect.height/2) {
22     movingRect.shapeColor = "red";
23     fixedRect.shapeColor = "red";
24   }
25   else {
26     movingRect.shapeColor = "green";
27     fixedRect.shapeColor = "green";
28   }
29   drawSprites();
30 }
31
32 function isTouching(){
33
34 }
  
```

Step 3: Place our code from line 18 to 28 inside isTouching() function— call the isTouching function

```

1  var fixedRect, movingRect;
2
3  function setup() {
4    createCanvas(1200,800);
5    fixedRect = createSprite(600, 400, 50, 80);
6    fixedRect.shapeColor = "green";
7
8    movingRect = createSprite(800, 400, 80, 30);
9    movingRect.shapeColor = "green";
10
11 }
12
13 function draw() {
14   background(0,0,0);
15   movingRect.x = World.mouseX;
16   movingRect.y = World.mouseY;
17
18   isTouching();
19   drawSprites();
20 }
21
22 function isTouching(){
23   if (movingRect.x - fixedRect.x < fixedRect.width/2 + movingRect.width/2
24       && fixedRect.x - movingRect.x < fixedRect.width/2 + movingRect.width/2
25       && movingRect.y - fixedRect.y < fixedRect.height/2 + movingRect.height/2
26       && fixedRect.y - movingRect.y < fixedRect.height/2 + movingRect.height/2) {
27     movingRect.shapeColor = "red";
28     fixedRect.shapeColor = "red";
29   }
30   else {
31     movingRect.shapeColor = "green";
32     fixedRect.shapeColor = "green";
33   }
34 }
  
```

Step 4: Modify our code for the function `isTouching()` so that it tells "yes" if the two rectangles are touching and "no" if the two rectangles are not touching. In computer language, "yes" and "no" are written as `true` and `false`.



```

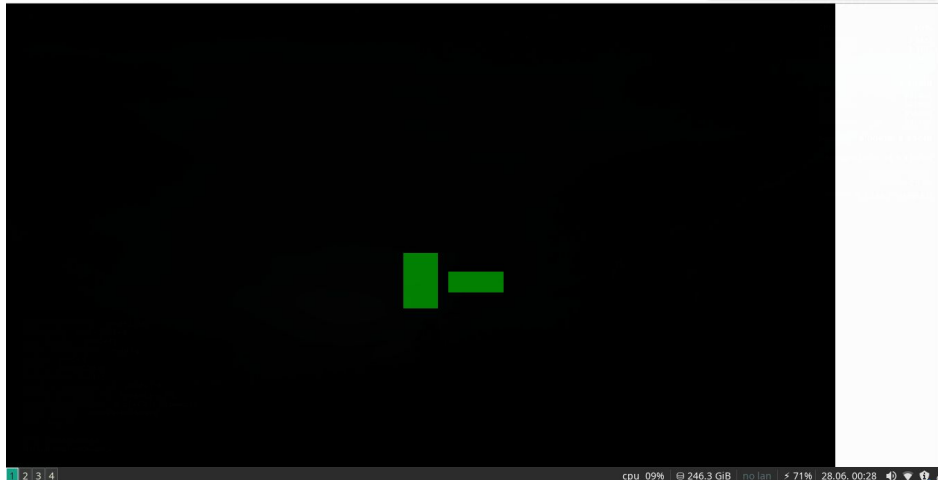
7
8  movingRect = createSprite(800, 400, 80, 30);
9  movingRect.shapeColor = "green";
10
11 }
12
13 function draw() {
14   background(0,0,0);
15   movingRect.x = World.mouseX;
16   movingRect.y = World.mouseY;
17
18   if(isTouching()){
19     movingRect.shapeColor = "red";
20     movingRect.shapeColor = "red";
21   }
22   else {
23     movingRect.shapeColor = "green";
24     movingRect.shapeColor = "green";
25   }
26   drawSprites();
27 }
28
29 function isTouching(){
30   if (movingRect.x - fixedRect.x < fixedRect.width/2 + movingRect.width/2
31     && fixedRect.x - movingRect.x < fixedRect.width/2 + movingRect.width/2
32     && movingRect.y - fixedRect.y < fixedRect.height/2 + movingRect.height/2
33     && fixedRect.y - movingRect.y < fixedRect.height/2 + movingRect.height/2) {
34
35     return true;
36   }
37   else {
38     return false;
39   }
40 }
  
```

Ln 25, Col 4 Spaces: 2 UTF-8 LF JavaScript

Lesson 20 - Google Drive Lesson Data20 - Google Sheet PRO-C12 Data - Google Sheet 127.0.0.1:8887

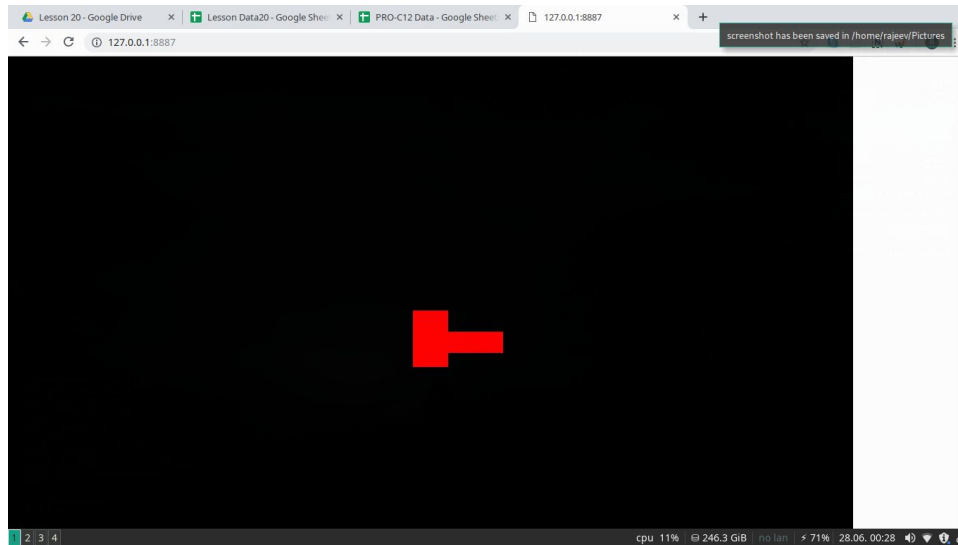
127.0.0.1:8887

screenshot has been saved in /home/rajeev/Pictures

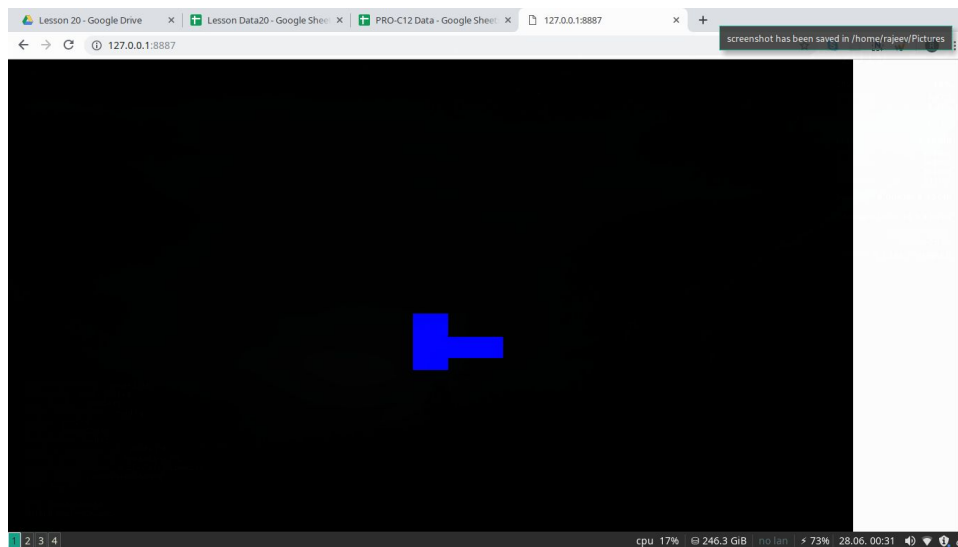


2 | 3 | 4

cpu: 09% 246.3 GiB no lan 71% 28.06.00:28



Step 5: Make the rectangles turn to blue instead of red when the two rectangles collide.



Step 6: Create some more game objects (sprites)

```

1  var fixedRect, movingRect;
2  var gameObject1, gameObject2, gameObject3, gameObject4;
3
4  function setup() {
5    createCanvas(1200,800);
6    fixedRect = createSprite(600, 400, 50, 80);
7    fixedRect.shapeColor = "green";
8
9    movingRect = createSprite(800, 400,80,30);
10   movingRect.shapeColor = "green";
11
12   gameObject1 = createSprite(100, 100, 50, 50);
13   gameObject1.shapeColor = "green";
14   gameObject2 = createSprite(200, 100, 50, 50);
15   gameObject2.shapeColor = "green";
16   gameObject3 = createSprite(300, 100, 50, 50);
17   gameObject3.shapeColor = "green";
18   gameObject4 = createSprite(400, 100, 50, 50);
19   gameObject4.shapeColor = "green";
20 }
21
22 function draw() {
23   background(0,0,0);
24   movingRect.x = World.mouseX;
25   movingRect.y = World.mouseY;
26
27   if(isTouching()){
28     movingRect.shapeColor = "blue";
29     fixedRect.shapeColor = "blue";
30   }
31   else {
32     movingRect.shapeColor = "green";
33     fixedRect.shapeColor = "green";
34   }
35   drawSprites();

```

Step 7: Change our function definition to make it accept arguments.

```

16   gameObject3 = createSprite(300, 100, 50, 50);
17   gameObject3.shapeColor = "green";
18   gameObject4 = createSprite(400, 100, 50, 50);
19   gameObject4.shapeColor = "green";
20 }
21
22 function draw() {
23   background(0,0,0);
24   movingRect.x = World.mouseX;
25   movingRect.y = World.mouseY;
26
27   if(isTouching()){
28     movingRect.shapeColor = "blue";
29     fixedRect.shapeColor = "blue";
30   }
31   else {
32     movingRect.shapeColor = "green";
33     fixedRect.shapeColor = "green";
34   }
35   drawSprites();
36 }
37
38 function isTouching(object1, object2){
39   if (movingRect.x - fixedRect.x < fixedRect.width/2 + movingRect.width/2
40     && fixedRect.x - movingRect.x < fixedRect.width/2 + movingRect.width/2
41     && movingRect.y - fixedRect.y < fixedRect.height/2 + movingRect.height/2
42     && fixedRect.y - movingRect.y < fixedRect.height/2 + movingRect.height/2) {
43
44     return true;
45   }
46   else {
47     return false;
48   }
49 }

```

Step 8:

Step 9: Modify the function isTouching() to change movingRect and fixedRect to object1 and object2

```

15  gameObject2.shapeColor = "green";
16  gameObject3 = createSprite(300, 100, 50, 50);
17  gameObject3.shapeColor = "green";
18  gameObject4 = createSprite(400, 100, 50, 50);
19  gameObject4.shapeColor = "green";
20  }
21
22  function draw() {
23    background(0,0,0);
24    movingRect.x = World.mouseX;
25    movingRect.y = World.mouseY;
26
27    if(isTouching(movingRect, gameObject1)){
28      movingRect.shapeColor = "blue";
29      gameObject1.shapeColor = "blue";
30    }
31    else {
32      movingRect.shapeColor = "green";
33      gameObject1.shapeColor = "green";
34    }
35    drawSprites();
36  }
37
38  function isTouching(object1,object2){
39    if (object1.x - object2.x < object2.width/2 + object1.width/2
40      && object2.x - object1.x < object2.width/2 + object1.width/2
41      && object1.y - object2.y < object2.height/2 + object1.height/2
42      && object2.y - object1.y < object2.height/2 + object1.height/2) {
43
44      return true;
45    }
46    else {
47      return false;
48    }
49  }

```

Step 10: Modify the code to check collisions between movingRect and other gameObjects

```

15  gameObject2.shapeColor = "green";
16  gameObject3 = createSprite(300, 100, 50, 50);
17  gameObject3.shapeColor = "green";
18  gameObject4 = createSprite(400, 100, 50, 50);
19  gameObject4.shapeColor = "green";
20  }
21
22  function draw() {
23    background(0,0,0);
24    movingRect.x = World.mouseX;
25    movingRect.y = World.mouseY;
26
27    if(isTouching(movingRect, gameObject1)){
28      movingRect.shapeColor = "blue";
29      gameObject1.shapeColor = "blue";
30    }
31    else {
32      movingRect.shapeColor = "green";
33      gameObject1.shapeColor = "green";
34    }
35    drawSprites();
36  }
37
38  function isTouching(object1,object2){
39    if (object1.x - object2.x < object2.width/2 + object1.width/2
40      && object2.x - object1.x < object2.width/2 + object1.width/2
41      && object1.y - object2.y < object2.height/2 + object1.height/2
42      && object2.y - object1.y < object2.height/2 + object1.height/2) {
43
44      return true;
45    }
46    else {
47      return false;
48    }
49  }

```



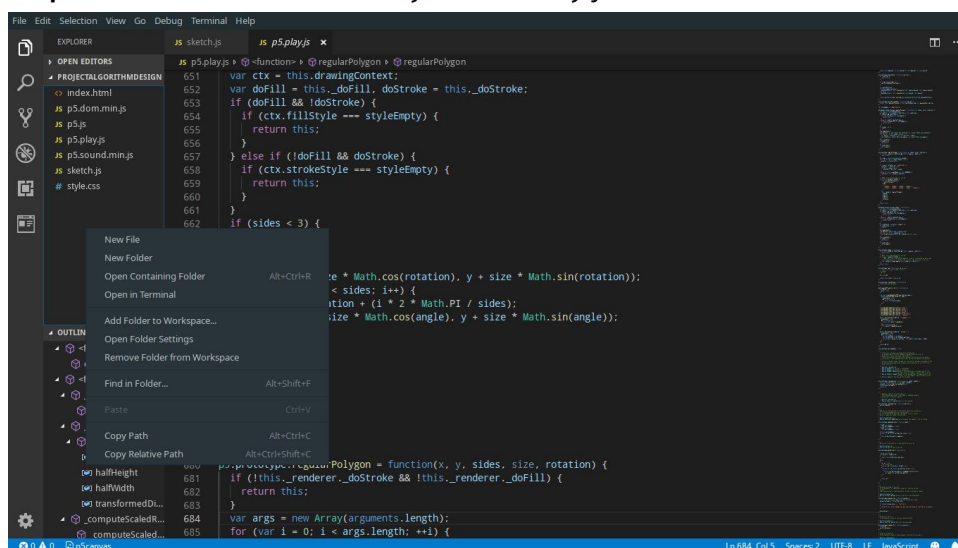
Step 11: Write the bounceOff function and test it

```

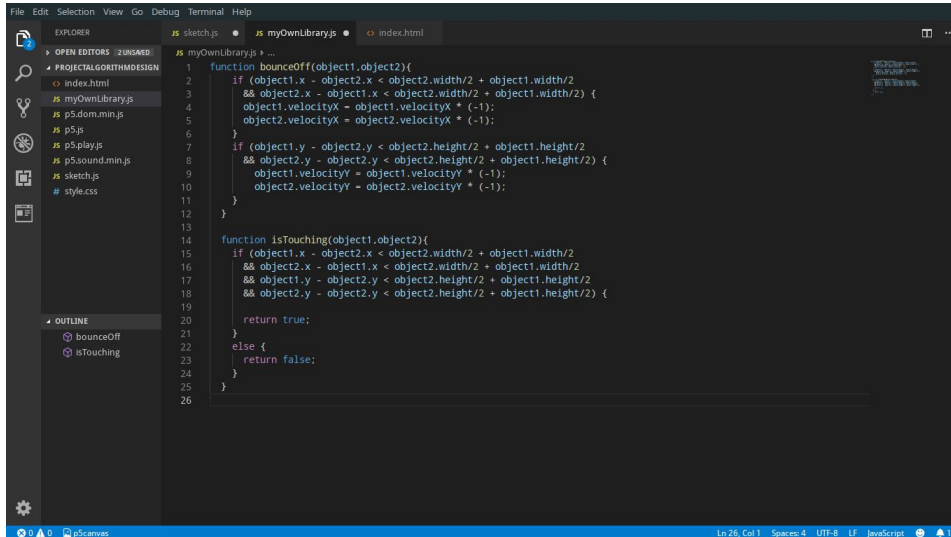
10  movingRect.debug = true;
11
12  movingRect.velocityY = -5;
13  fixedRect.velocityY = +5;
14  }
15
16  function draw() {
17    background(0,0,0);
18
19    bounceOff(movingRect,fixedRect);
20    drawSprites();
21  }
22
23  function bounceOff(object1,object2){
24    if (object1.x - object2.x < object2.width/2 + object1.width/2
25      && object2.x - object1.x < object2.width/2 + object1.width/2) {
26      object1.velocityX = object1.velocityX * (-1);
27      object2.velocityX = object2.velocityX * (-1);
28    }
29    if (object1.y - object2.y < object2.height/2 + object1.height/2
30      && object2.y - object1.y < object2.height/2 + object1.height/2) {
31      object1.velocityY = object1.velocityY * (-1);
32      object2.velocityY = object2.velocityY * (-1);
33    }
34  }

```

Step 12: Create a file called myOwnLibrary.js



Step 13: Copy the two functions— bounceOff and isTouching created inside myOwnLibrary.js

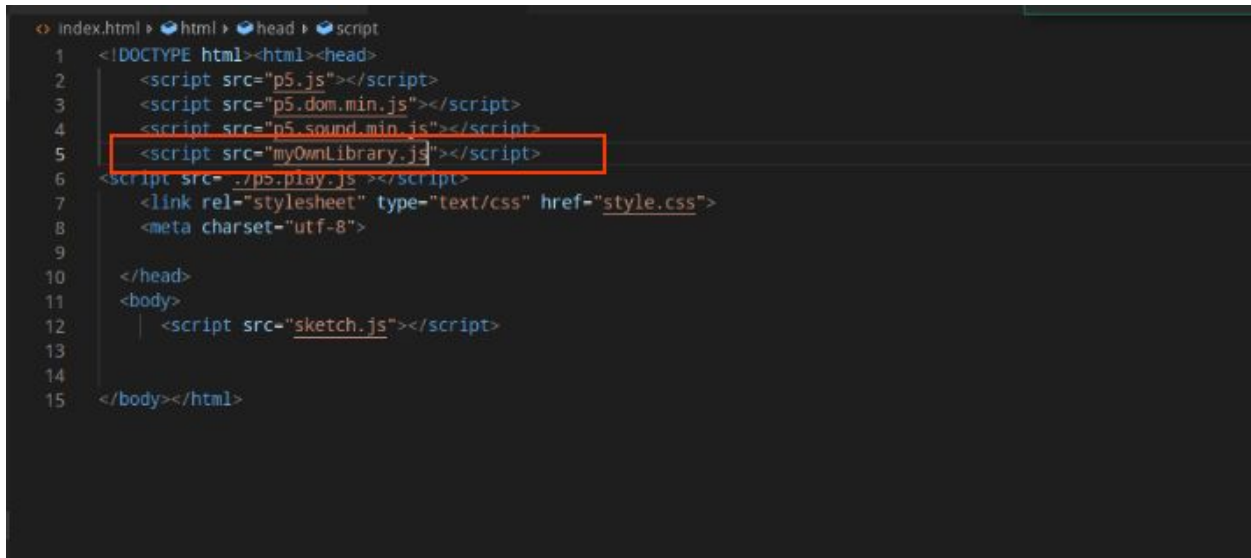


```

1 function bounceOff(object1, object2){
2   if (object1.x - object2.x < object2.width/2 + object1.width/2
3     && object2.x - object1.x < object2.width/2 + object1.width/2) {
4     object1.velocityx = object1.velocityx * (-1);
5     object2.velocityx = object2.velocityx * (-1);
6   }
7   if (object1.y - object2.y < object2.height/2 + object1.height/2
8     && object2.y - object1.y < object2.height/2 + object1.height/2) {
9     object1.velocityy = object1.velocityy * (-1);
10    object2.velocityy = object2.velocityy * (-1);
11  }
12 }
13
14 function isTouching(object1, object2){
15   if (object1.x - object2.x < object2.width/2 + object1.width/2
16     && object2.x - object1.x < object2.width/2 + object1.width/2
17     && object1.y - object2.y < object2.height/2 + object1.height/2
18     && object2.y - object1.y < object2.height/2 + object1.height/2) {
19     return true;
20   }
21   else {
22     return false;
23   }
24 }
25
26

```

Step 14: Include the myOwnLibrary.js in your index.html file. This file can now be included into any project and use the two functions - bounceOff and isTouching - in your code without writing any code!



```

1 <!DOCTYPE html><html><head>
2   <script src="p5.js"></script>
3   <script src="p5.dom.min.js"></script>
4   <script src="p5.sound.min.js"></script>
5   <script src="myOwnLibrary.js"></script>
6   <script src="./p5.play.js"></script>
7   <link rel="stylesheet" type="text/css" href="style.css">
8   <meta charset="utf-8">
9
10 </head>
11 <body>
12   <script src="sketch.js"></script>
13
14 </body></html>
15

```

What's next?: Get started on the Angry Birds Project