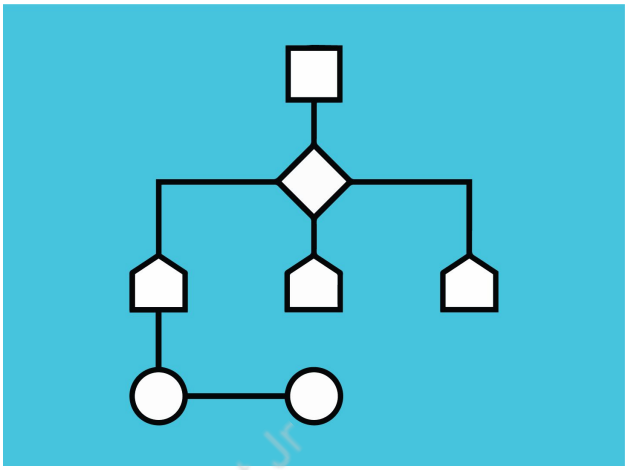
DESIGNING ALGORITHMS



What is our GOAL for this MODULE?

We used our knowledge about algorithms to write a collision detection algorithm and used a boilerplate code.

What did we ACHIEVE in the class TODAY?

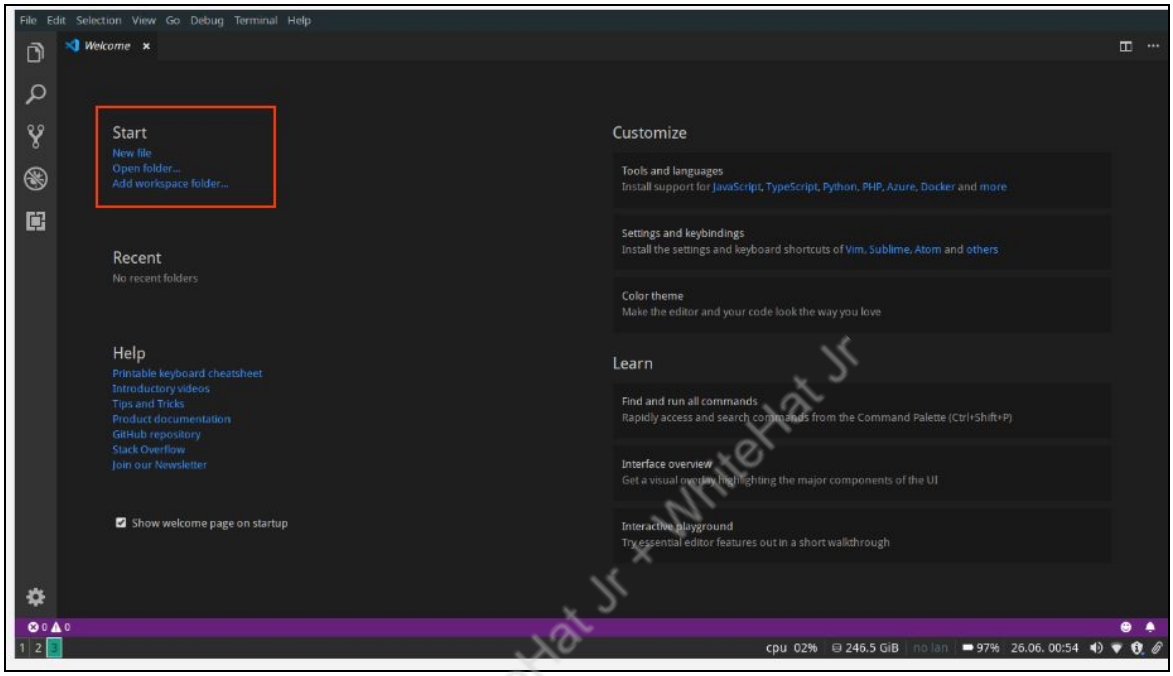
- Downloaded the Boilerplate code.
- Designed an algorithm for collision detection, wrote code, and tested the program.

Which CONCEPTS/ CODING BLOCKS did we cover today?

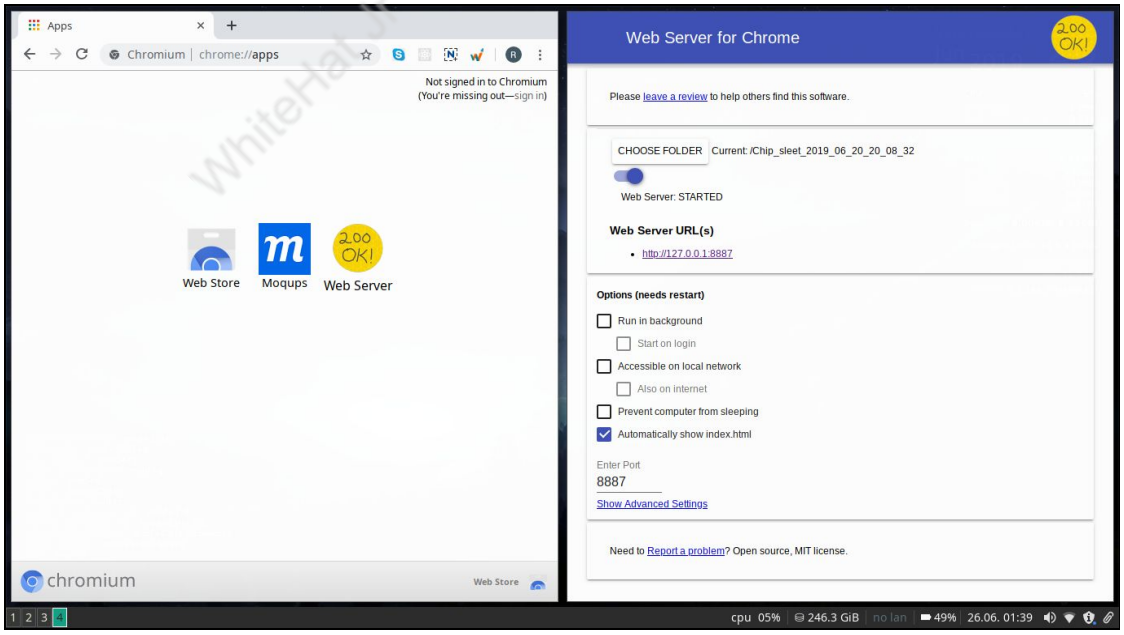
- boilerplate code
- touches
- && (Logical AND) operator

How did we DO the activities?

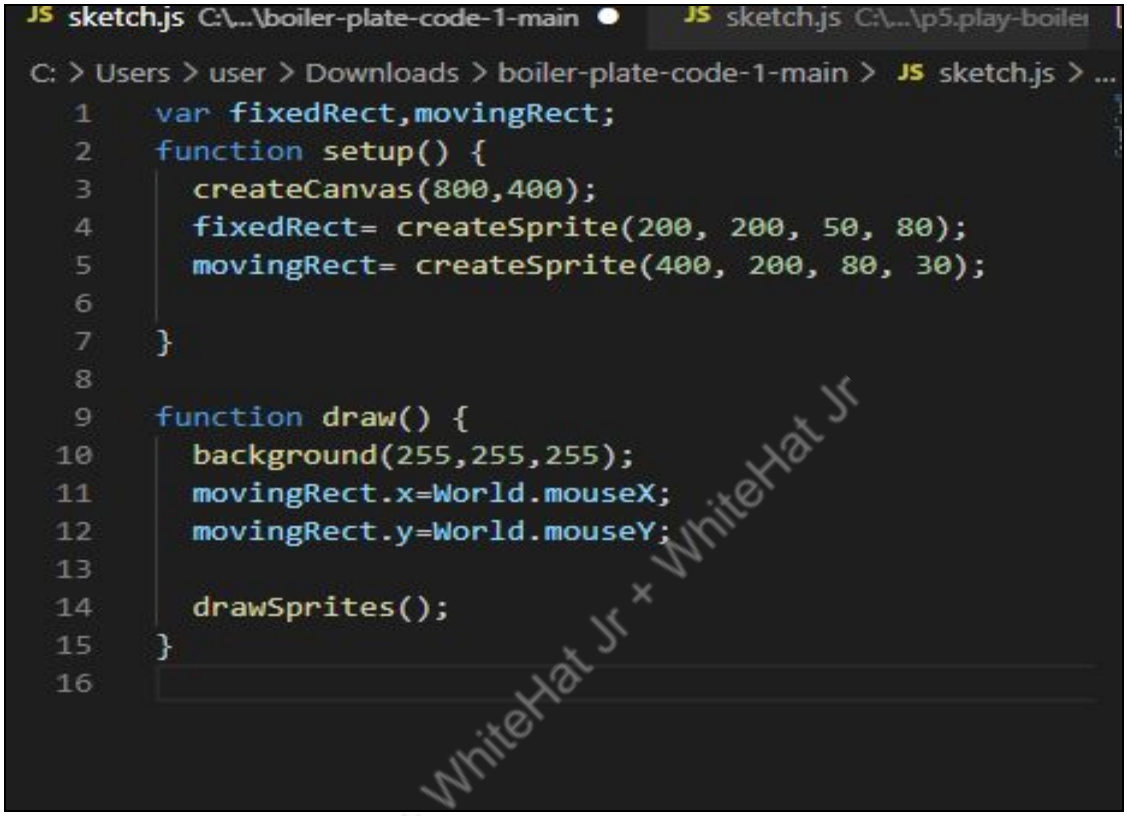
1. Download the boilerplate code and open it in Visual Studio Code.

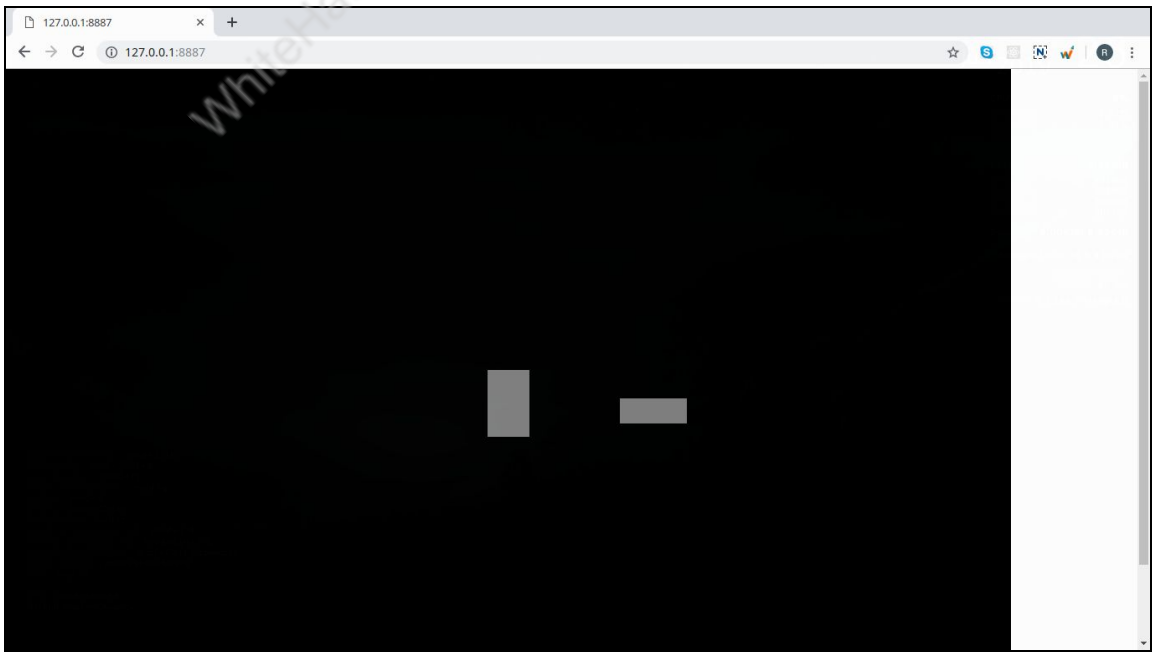


- Run the server and point it to the same folder.



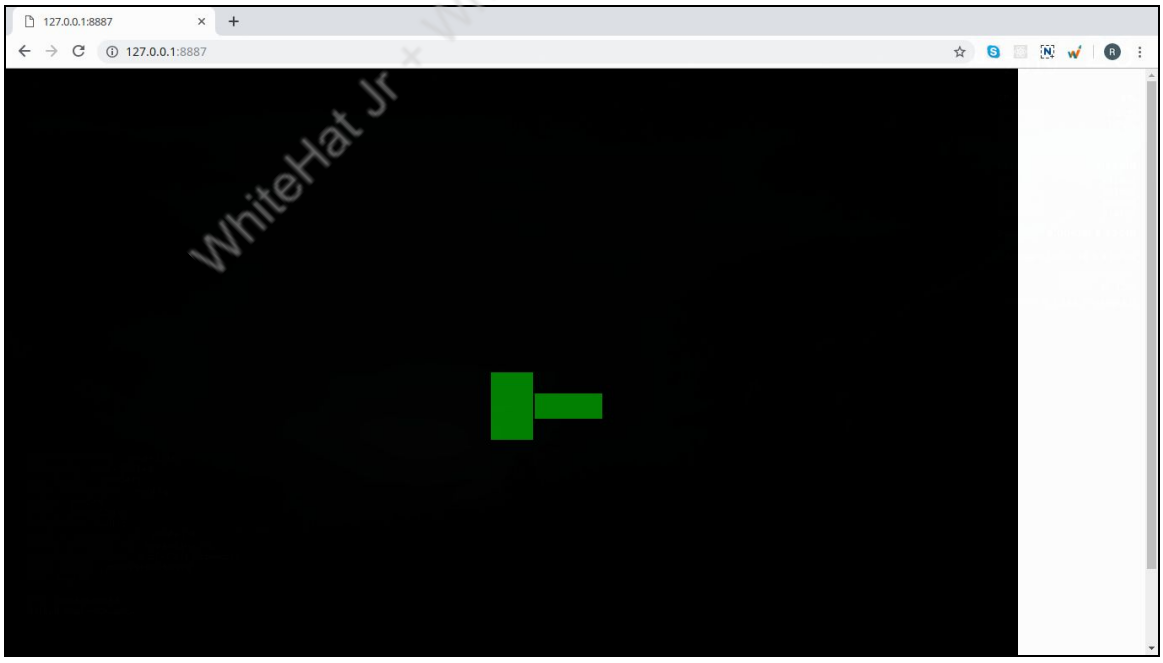
2. Create two rectangles with different widths and heights - one rectangle fixed, and the other moving; add controls to move the rectangle.





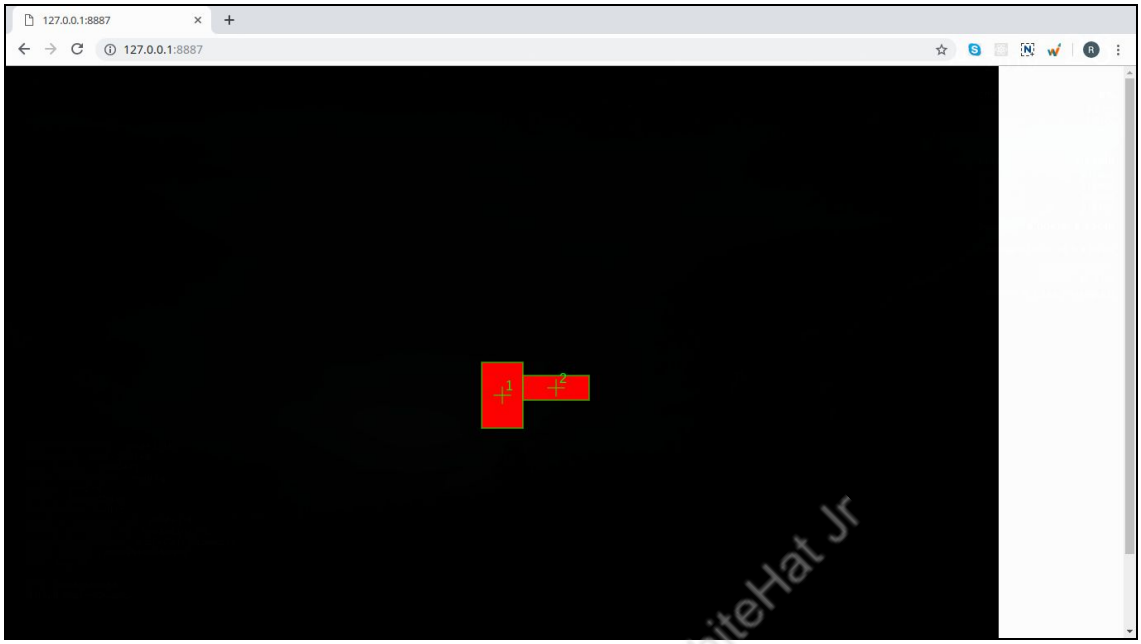
3. Create a collision detection algorithm so that when the two objects/rectangles collide, their color changes to green.

```
var fixedRect,movingRect;
function setup() {
  createCanvas(800,400);
  fixedRect= createSprite(200, 200, 50, 80);
  fixedRect.Shapecolor="green";
  movingRect= createSprite(400, 200, 80, 30);
  movingRect.ShapeColor="green";
}

function draw() {
  background(255,255,255);
  movingRect.x=World.mouseX;
  movingRect.y=World.mouseY;

  drawSprites();
}
```



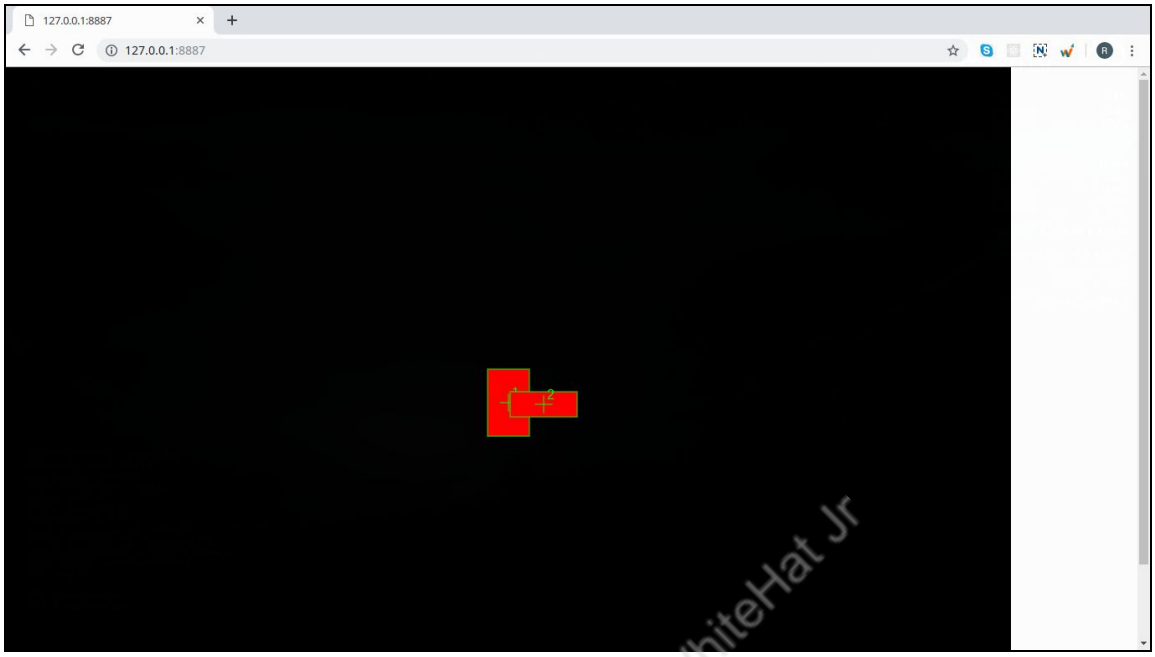
4. Add a condition to change the color of rectangles to red when the objects touch each other, else kept it green.

```
var fixedRect,movingRect;
function setup() {
  createCanvas(800,400);
  fixedRect= createSprite(200, 200, 50, 80);
  movingRect= createSprite(400, 200, 80, 30);

}

function draw() {
  background(255,255,255);
  movingRect.x=World.mouseX;
  movingRect.y=World.mouseY;
if(movingRect.x-fixedRect.x===fixedRect.width/2+movingRect.width/2 )
  {movingRect.shapeColor="red";
  fixedRect.shapeColor="red";

}
else{movingRect.shapeColor="green";
fixedRect.shapeColor="green";

}
  drawSprites();
}
```



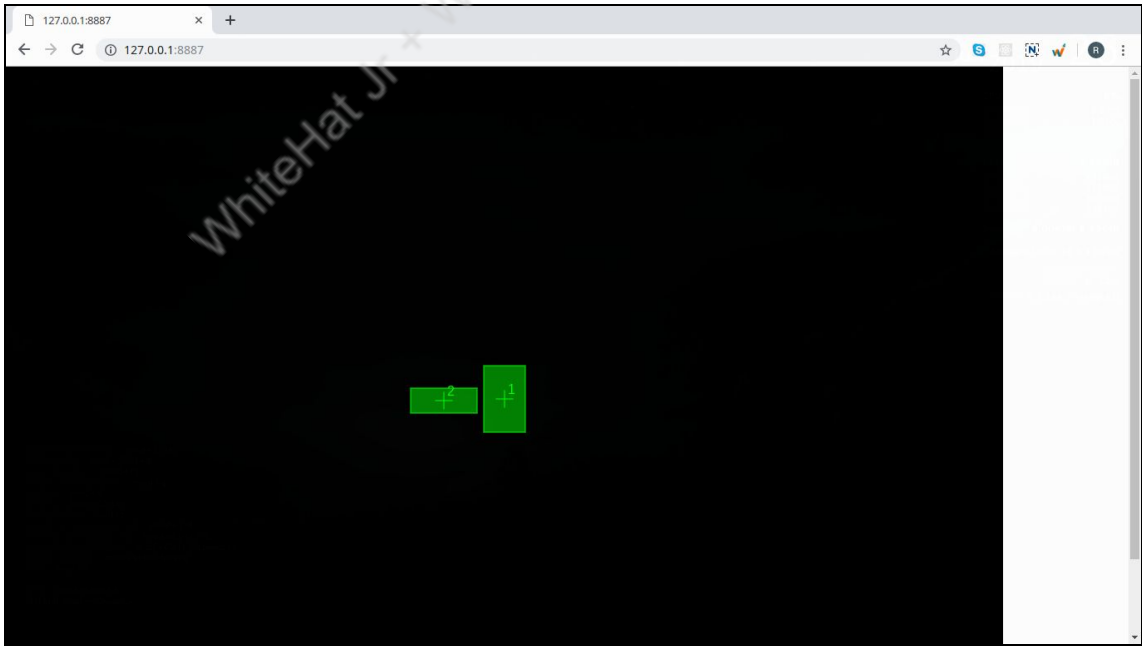
5. Change the "===" sign to "<", to detect when the moving rectangle has crossed over to the fixed rectangle.

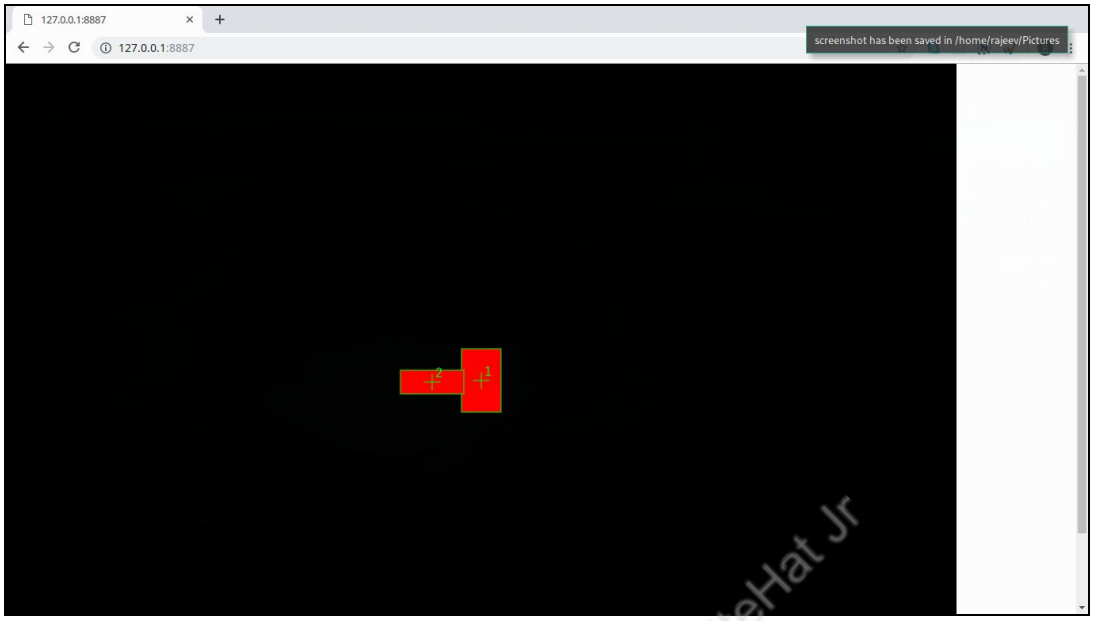
```
var fixedRect,movingRect;
function setup() {
  createCanvas(800,400);
  fixedRect= createSprite(200, 200, 50, 80);
  movingRect= createSprite(400, 200, 80, 30);

}

function draw() {
  background(255,255,255);
  movingRect.x=World.mouseX;
  movingRect.y=World.mouseY;
if(movingRect.x-fixedRect.x<fixedRect.width/2+movingRect.width/2 )
  {movingRect.shapeColor="red";
  fixedRect.shapeColor="red";

}
else{movingRect.shapeColor="green";
fixedRect.shapeColor="green";

}
  drawSprites();
}
```



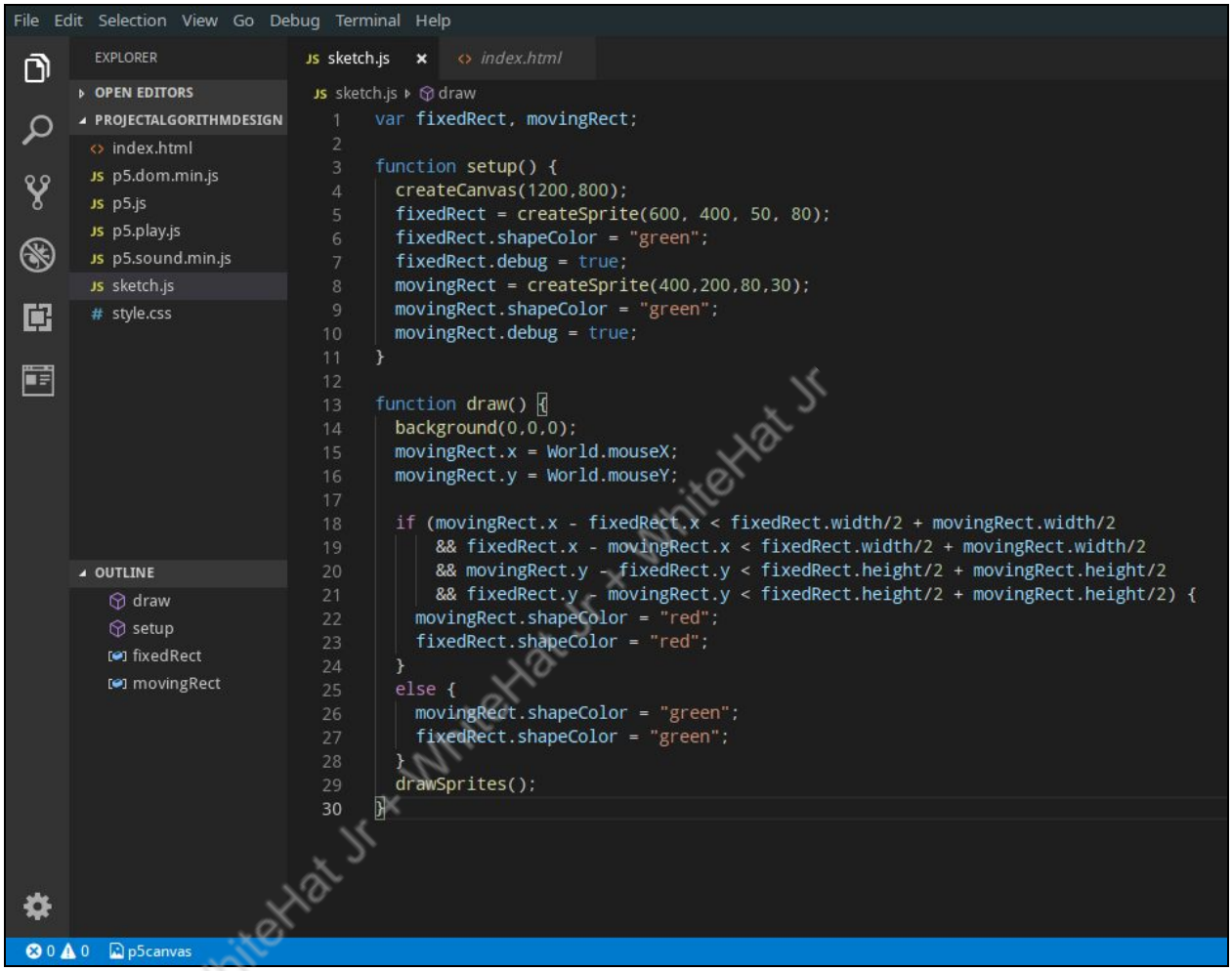
6. Make use of && operator to add two conditions.

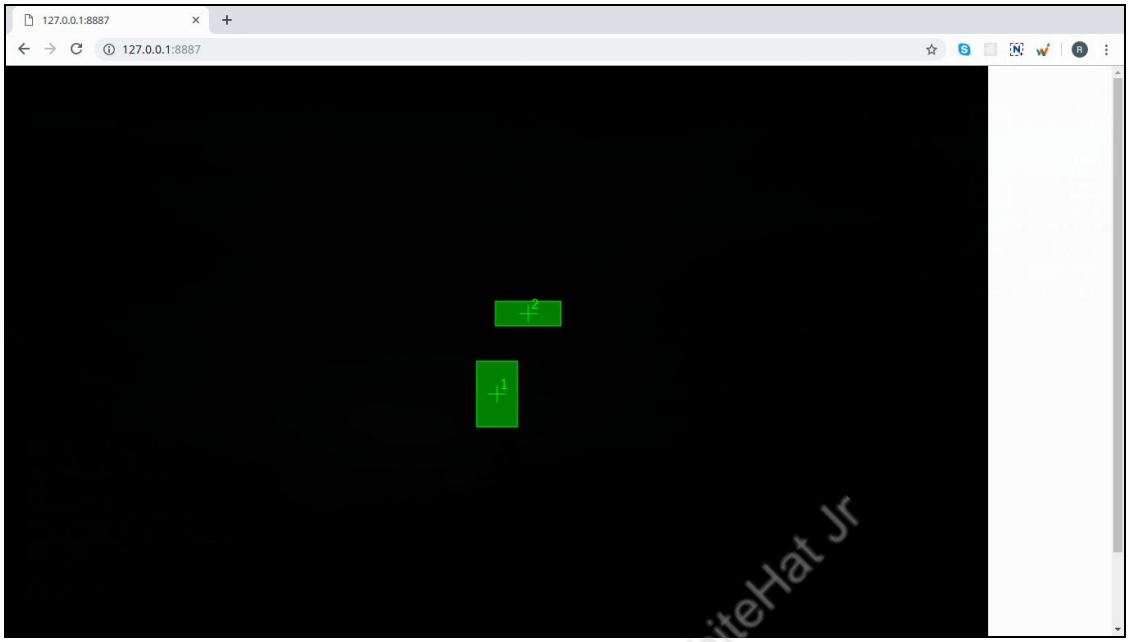
7. Check the condition when the rectangle is moving from left to right it is green and as it touches each other it becomes red.

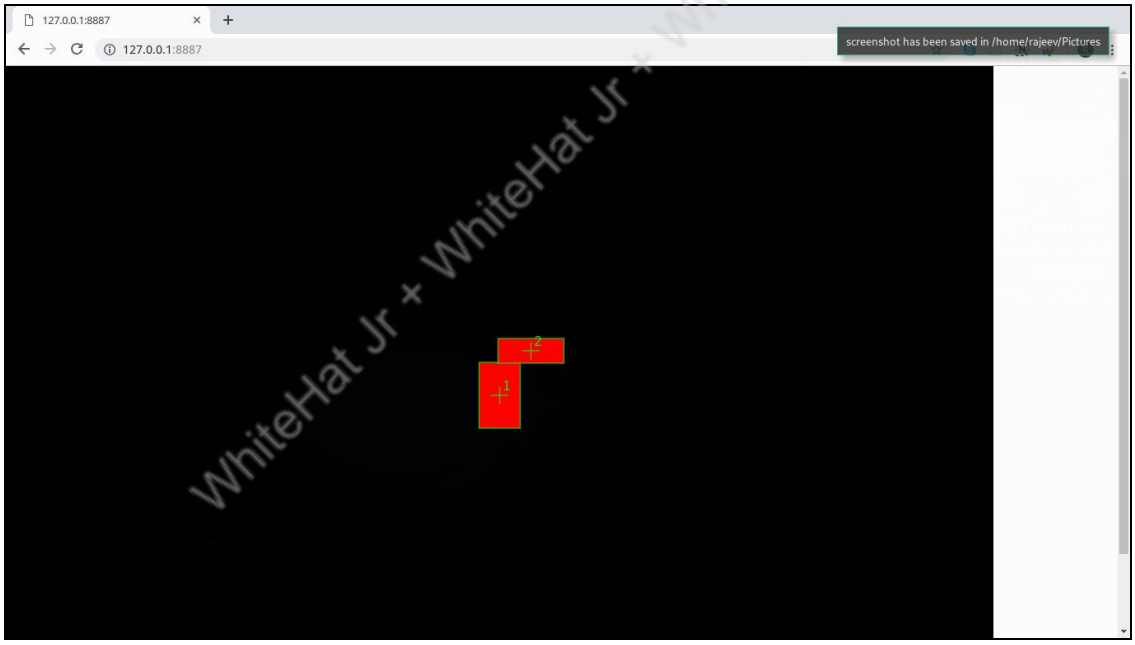




8. Check the vertical distance between the two rectangles and set the rectangle color to green and if the distance is 0 else to red.







What's next?

We will start working on the angry birds game.

Extend Your Knowledge:

1. Learn more about libraries in p5:https://p5js.org/libraries/