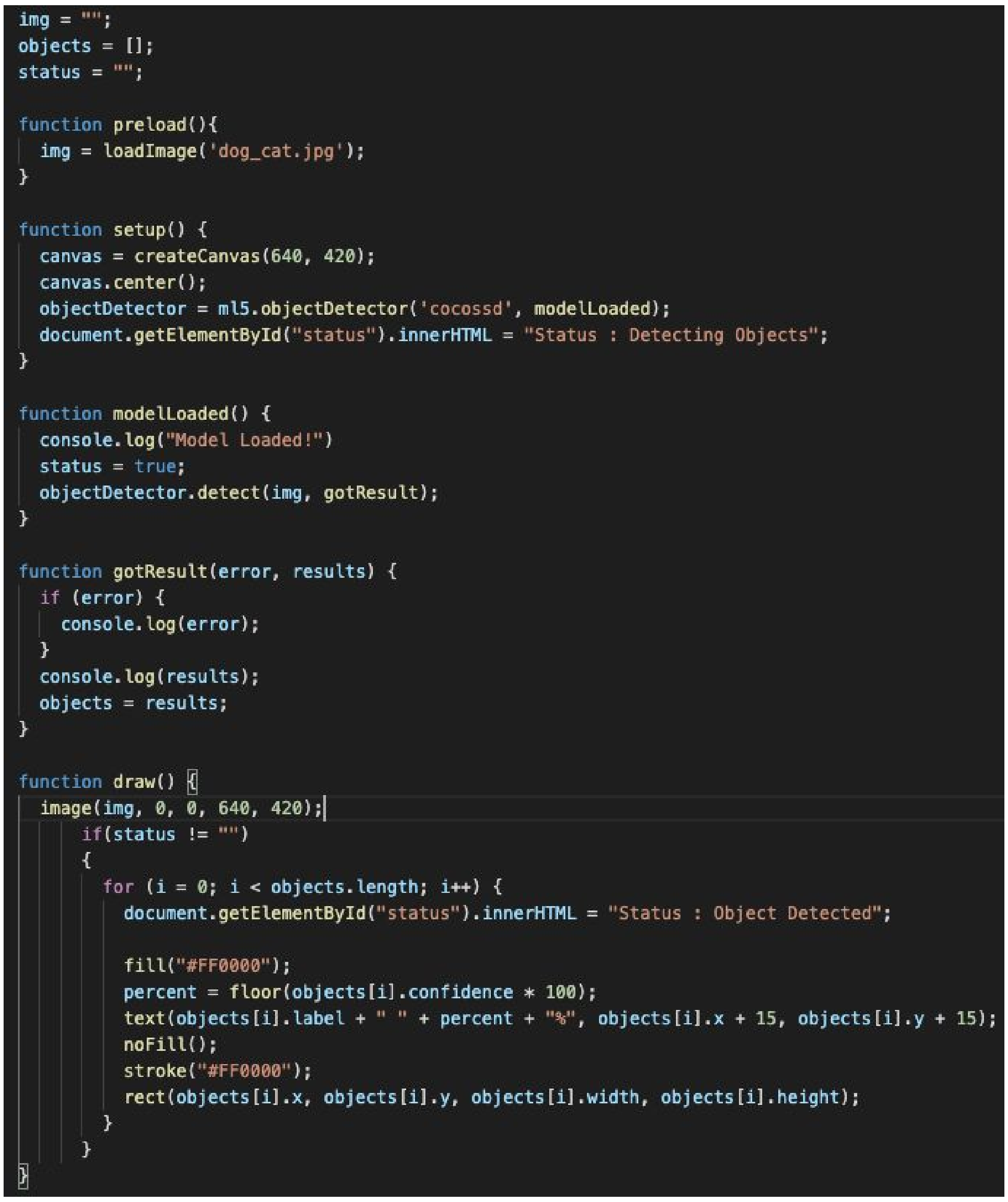
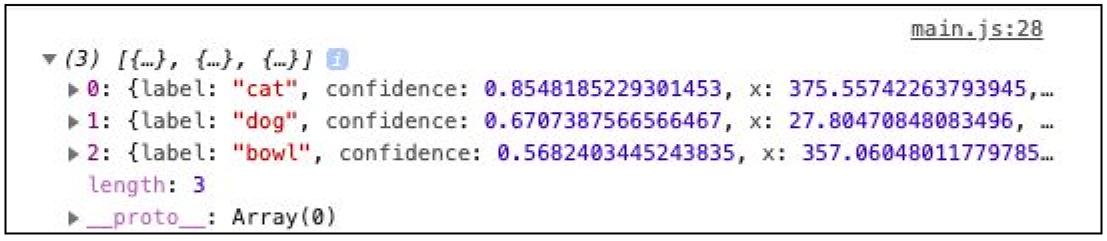
Previous class JS code



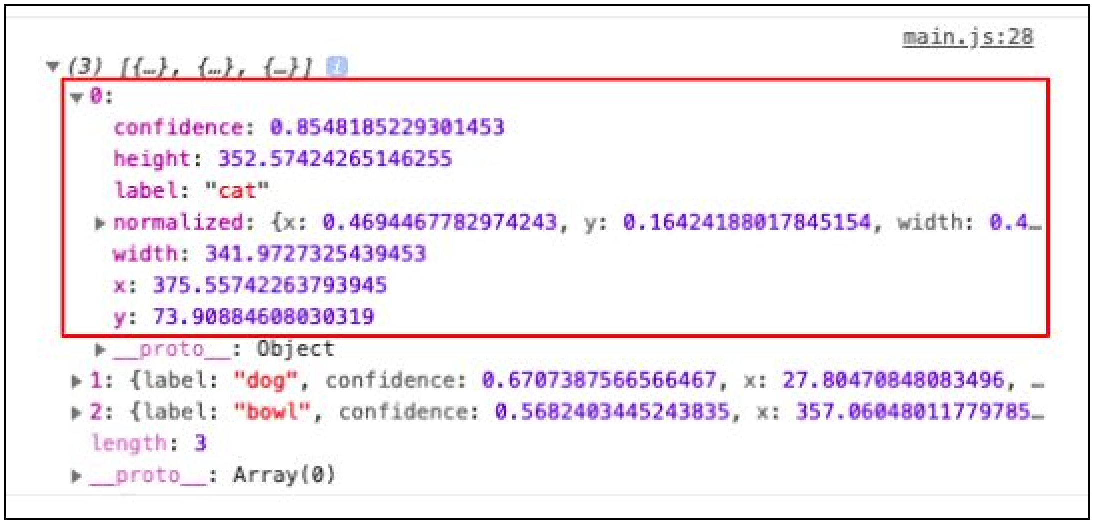
```
img = "";
objects = [];
status = "";

function preload(){
  img = loadImage('dog_cat.jpg');
}

function setup() {
  canvas = createCanvas(640, 420);
  canvas.center();
  objectDetector = ml5.objectDetector('cocossd', modelLoaded);
  document.getElementById("status").innerHTML = "Status : Detecting Objects";
}

function modelLoaded() {
  console.log("Model Loaded!")
  status = true;
  objectDetector.detect(img, gotResult);
}

function gotResult(error, results) {
  if (error) {
    console.log(error);
  }
  console.log(results);
  objects = results;
}

function draw() {
  image(img, 0, 0, 640, 420);
      if(status != "")
      {
        for (i = 0; i < objects.length; i++) {
          document.getElementById("status").innerHTML = "Status : Object Detected";

          fill("#FF0000");
          percent = floor(objects[i].confidence * 100);
          text(objects[i].label + " " + percent + "%", objects[i].x + 15, objects[i].y + 15);
          noFill();
          stroke("#FF0000");
          rect(objects[i].x, objects[i].y, objects[i].width, objects[i].height);
        }
      }
}
```

We got the following array in the previous class from cococssd model



```
                                                                    main.js:28
▼(3) [{…}, {…}, {…}]
  ▶0: {label: "cat", confidence: 0.8548185229301453, x: 375.55742263793945,…
  ▶1: {label: "dog", confidence: 0.6707387566566467, x: 27.80470848083496, …
  ▶2: {label: "bowl", confidence: 0.5682403445243835, x: 357.06048011779785…
   length: 3
  ▶__proto__: Array(0)
```

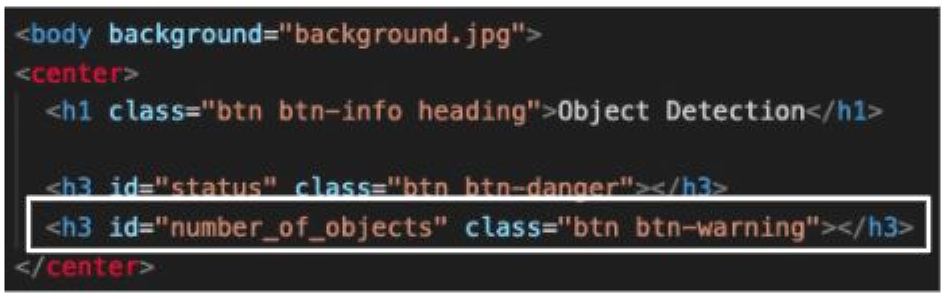
Inside the first array



```
                                                                    main.js:28
▼(3) [{…}, {…}, {…}]
  ▼0:
     confidence: 0.8548185229301453
     height: 352.57424265146255
     label: "cat"
    ▶normalized: {x: 0.4694467782974243, y: 0.16424188017845154, width: 0.4…
     width: 341.9727325439453
     x: 375.55742263793945
     y: 73.90884608030319
    ▶__proto__: Object
  ▶1: {label: "dog", confidence: 0.6707387566566467, x: 27.80470848083496, …
  ▶2: {label: "bowl", confidence: 0.5682403445243835, x: 357.06048011779785…
   length: 3
  ▶__proto__: Array(0)
```

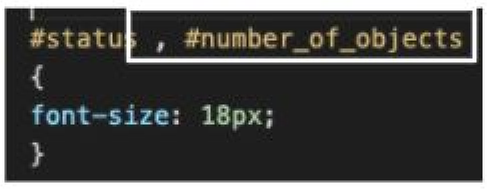
HTML and CSS code

1. **Adding a h3 tag, for holding the number of objects detected**



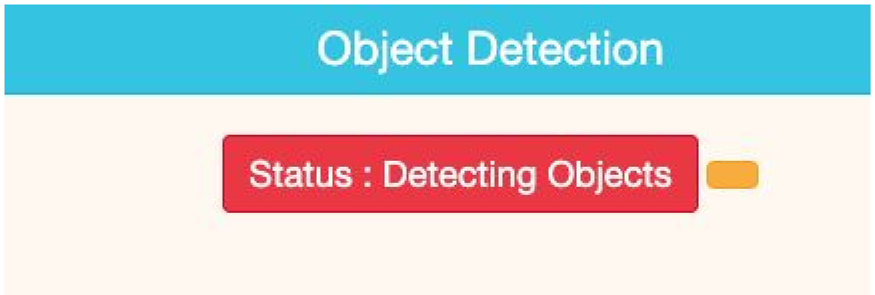
```
<body background="background.jpg">
<center>
  <h1 class="btn btn-info heading">Object Detection</h1>

  <h3 id="status" class="btn btn-danger"></h3>
  <h3 id="number_of_objects" class="btn btn-warning"></h3>
</center>
```

Adding style in style.css



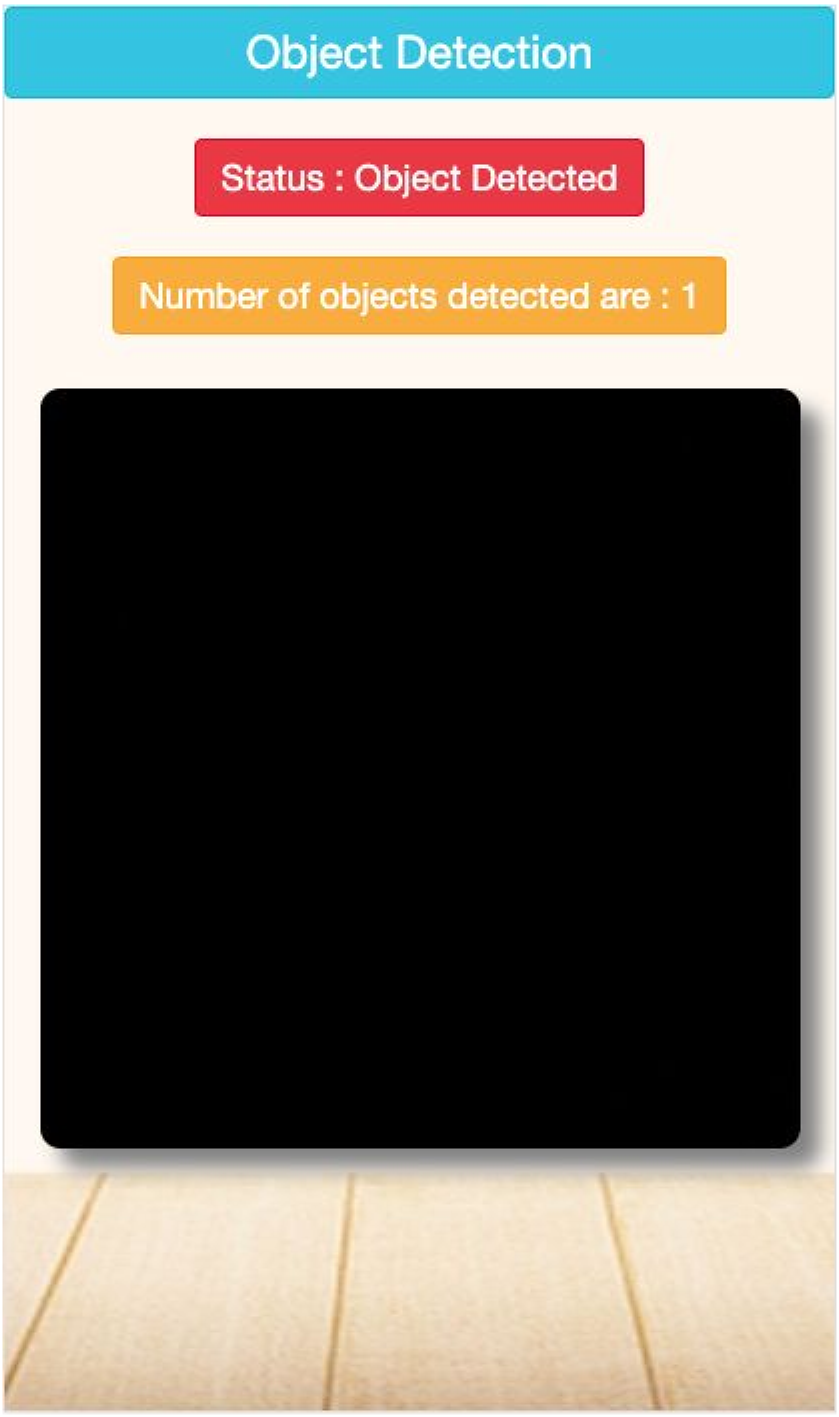
```
#status , #number_of_objects
{
font-size: 18px;
}
```

Output -



2. **Adding margin top to canvas**

On mobile the web app should look like this -

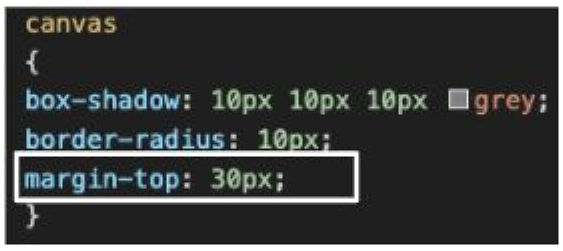


Object Detection

Status : Object Detected

Number of objects detected are : 1

That’s added margin top to canvas

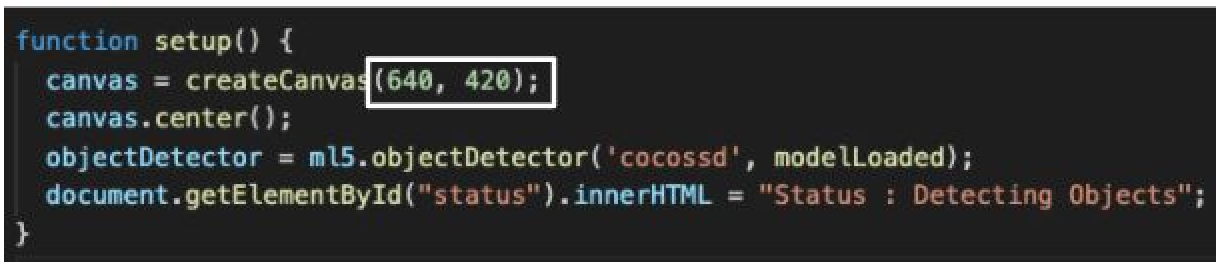


```
canvas
{
box-shadow: 10px 10px 10px grey;
border-radius: 10px;
margin-top: 30px;
}
```

JS code

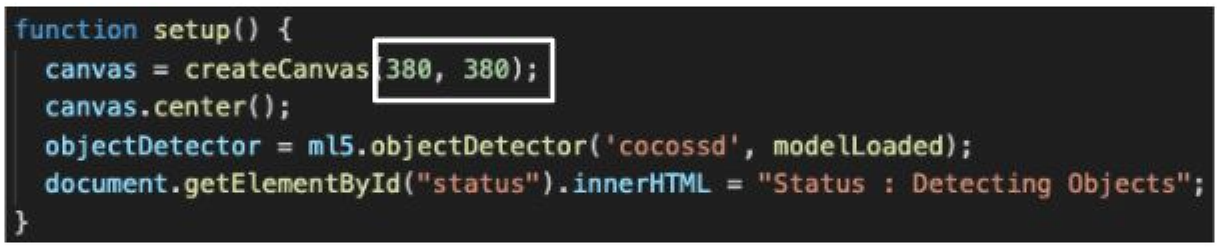
1. **We will reduce the size of the canvas**

From this -



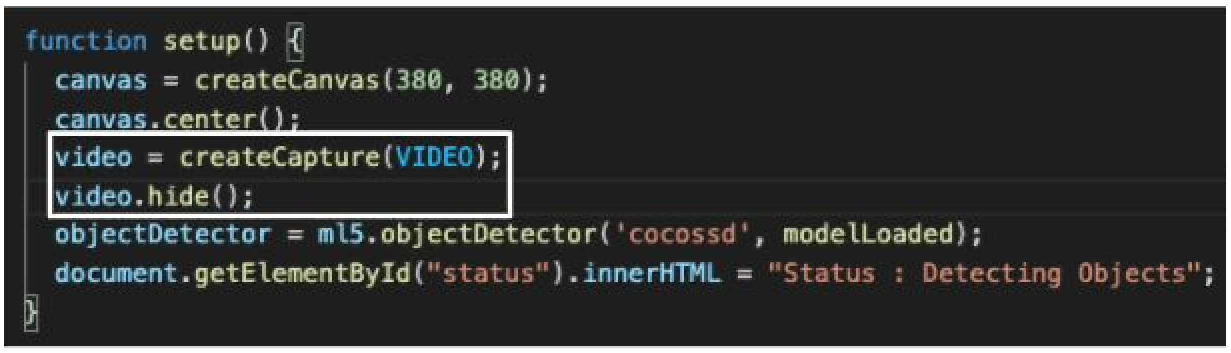
```
function setup() {
  canvas = createCanvas(640, 420);
  canvas.center();
  objectDetector = ml5.objectDetector('cocossd', modelLoaded);
  document.getElementById("status").innerHTML = "Status : Detecting Objects";
}
```

To this -



```
function setup() {
  canvas = createCanvas(380, 380);
  canvas.center();
  objectDetector = ml5.objectDetector('cocossd', modelLoaded);
  document.getElementById("status").innerHTML = "Status : Detecting Objects";
}
```

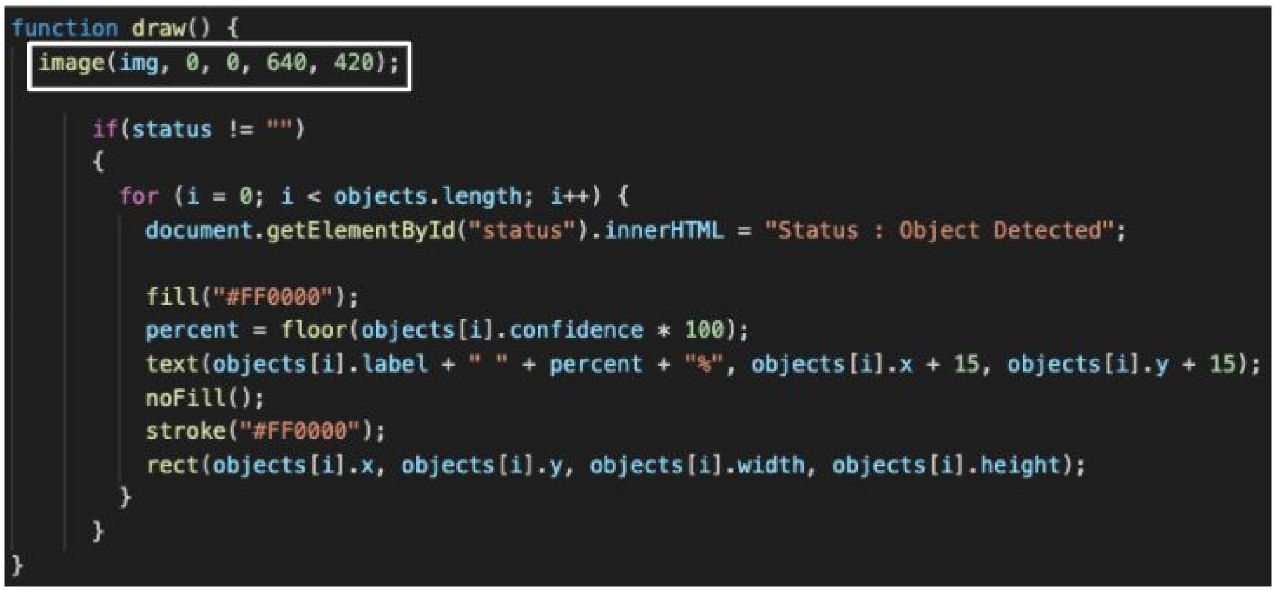
2. **Code for accessing webcam**



```
function setup() {
  canvas = createCanvas(380, 380);
  canvas.center();
  video = createCapture(VIDEO);
  video.hide();
  objectDetector = ml5.objectDetector('cocossd', modelLoaded);
  document.getElementById("status").innerHTML = "Status : Detecting Objects";
}
```

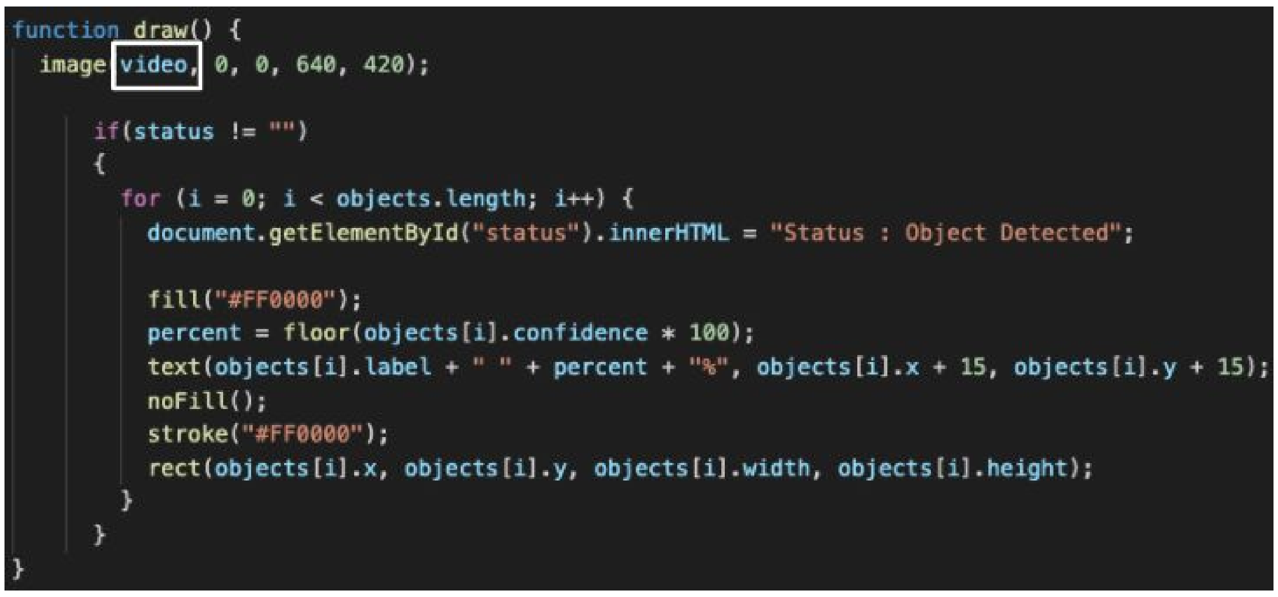
3. **Update JS for placing webcam live view on the canvas**

From this -



```
function draw() {
  image(img, 0, 0, 640, 420);

    if(status != "")
    {
      for (i = 0; i < objects.length; i++) {
        document.getElementById("status").innerHTML = "Status : Object Detected";

        fill("#FF0000");
        percent = floor(objects[i].confidence * 100);
        text(objects[i].label + " " + percent + "%", objects[i].x + 15, objects[i].y + 15);
        noFill();
        stroke("#FF0000");
        rect(objects[i].x, objects[i].y, objects[i].width, objects[i].height);
      }
    }
}
```

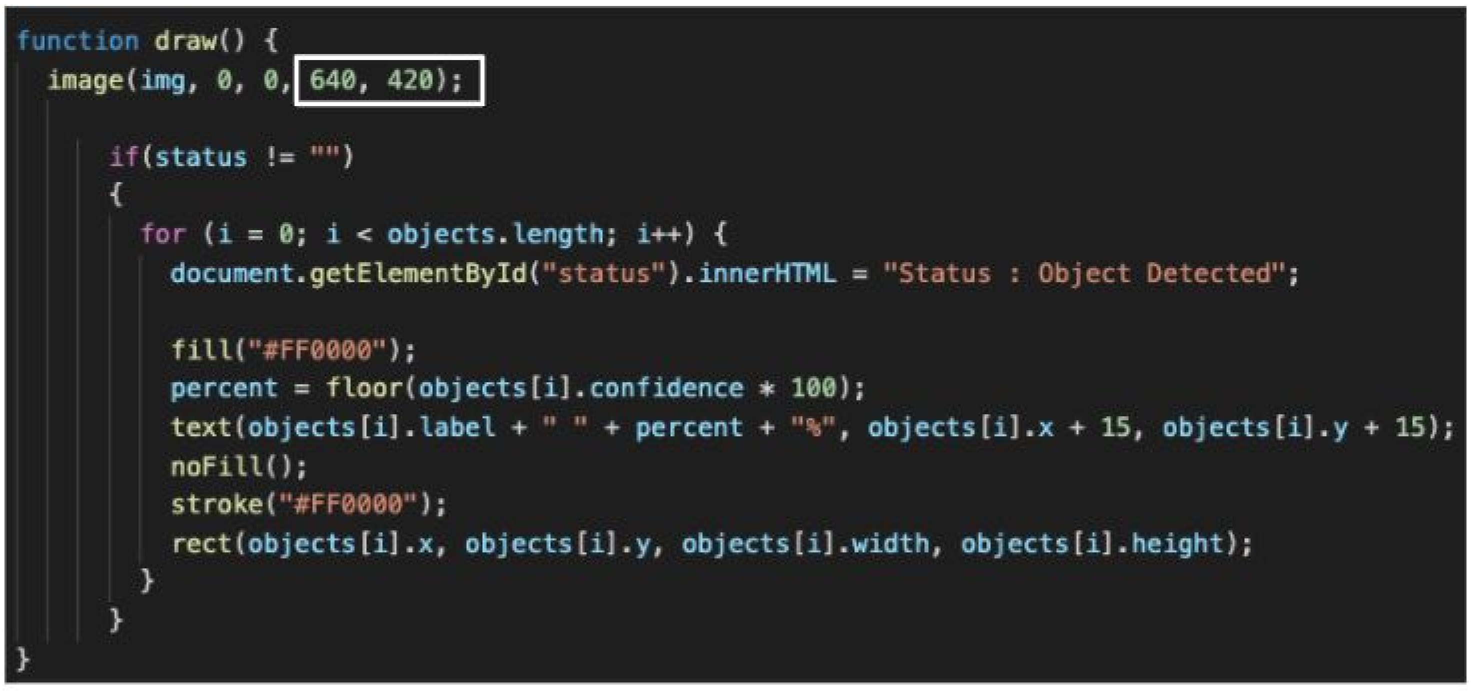
To this -



```
function draw() {
  image(video, 0, 0, 640, 420);

    if(status != "")
    {
      for (i = 0; i < objects.length; i++) {
        document.getElementById("status").innerHTML = "Status : Object Detected";

        fill("#FF0000");
        percent = floor(objects[i].confidence * 100);
        text(objects[i].label + " " + percent + "%", objects[i].x + 15, objects[i].y + 15);
        noFill();
        stroke("#FF0000");
        rect(objects[i].x, objects[i].y, objects[i].width, objects[i].height);
      }
    }
}
```

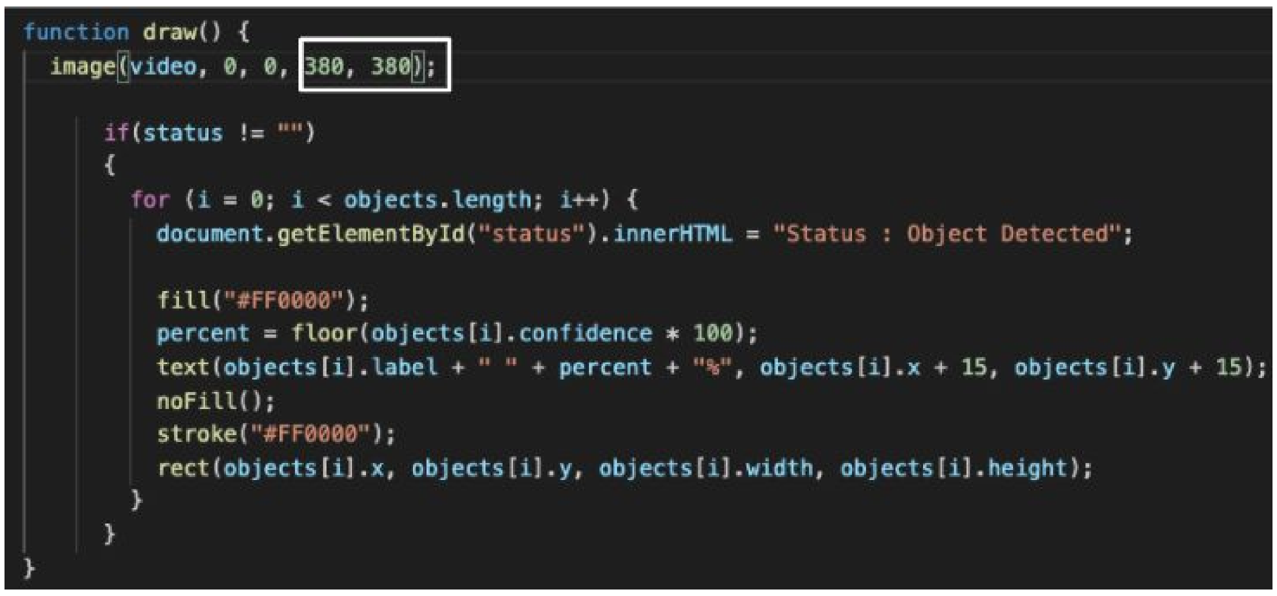
4. **Update the size of the webcam live view coming on the canvas**

From this -



```
function draw() {
  image(img, 0, 0, 640, 420);

    if(status != "")
    {
      for (i = 0; i < objects.length; i++) {
        document.getElementById("status").innerHTML = "Status : Object Detected";

        fill("#FF0000");
        percent = floor(objects[i].confidence * 100);
        text(objects[i].label + " " + percent + "%", objects[i].x + 15, objects[i].y + 15);
        noFill();
        stroke("#FF0000");
        rect(objects[i].x, objects[i].y, objects[i].width, objects[i].height);
      }
    }
}
```

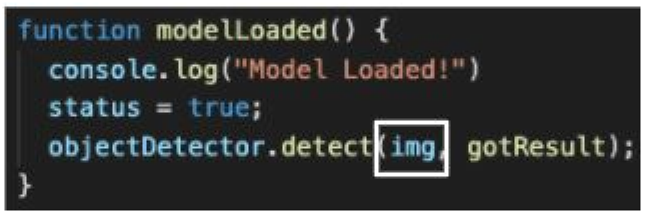
To this -



```
function draw() {
  image(video, 0, 0, 380, 380);

    if(status != "")
    {
      for (i = 0; i < objects.length; i++) {
        document.getElementById("status").innerHTML = "Status : Object Detected";

        fill("#FF0000");
        percent = floor(objects[i].confidence * 100);
        text(objects[i].label + " " + percent + "%", objects[i].x + 15, objects[i].y + 15);
        noFill();
        stroke("#FF0000");
        rect(objects[i].x, objects[i].y, objects[i].width, objects[i].height);
      }
    }
}
```

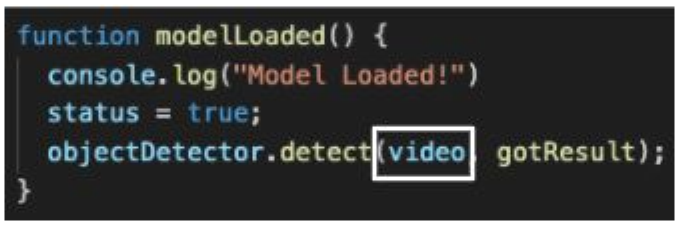
5. **Change the input given of the detect() function**

From this -



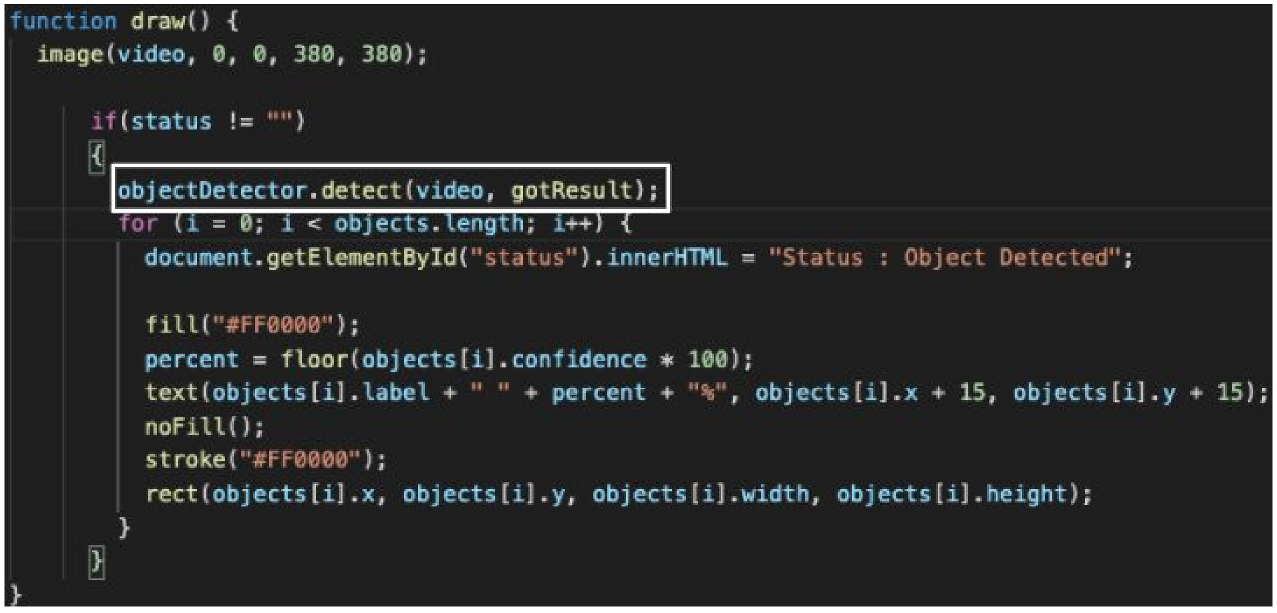
```
function modelLoaded() {
  console.log("Model Loaded!")
  status = true;
  objectDetector.detect(img, gotResult);
}
```

To this -



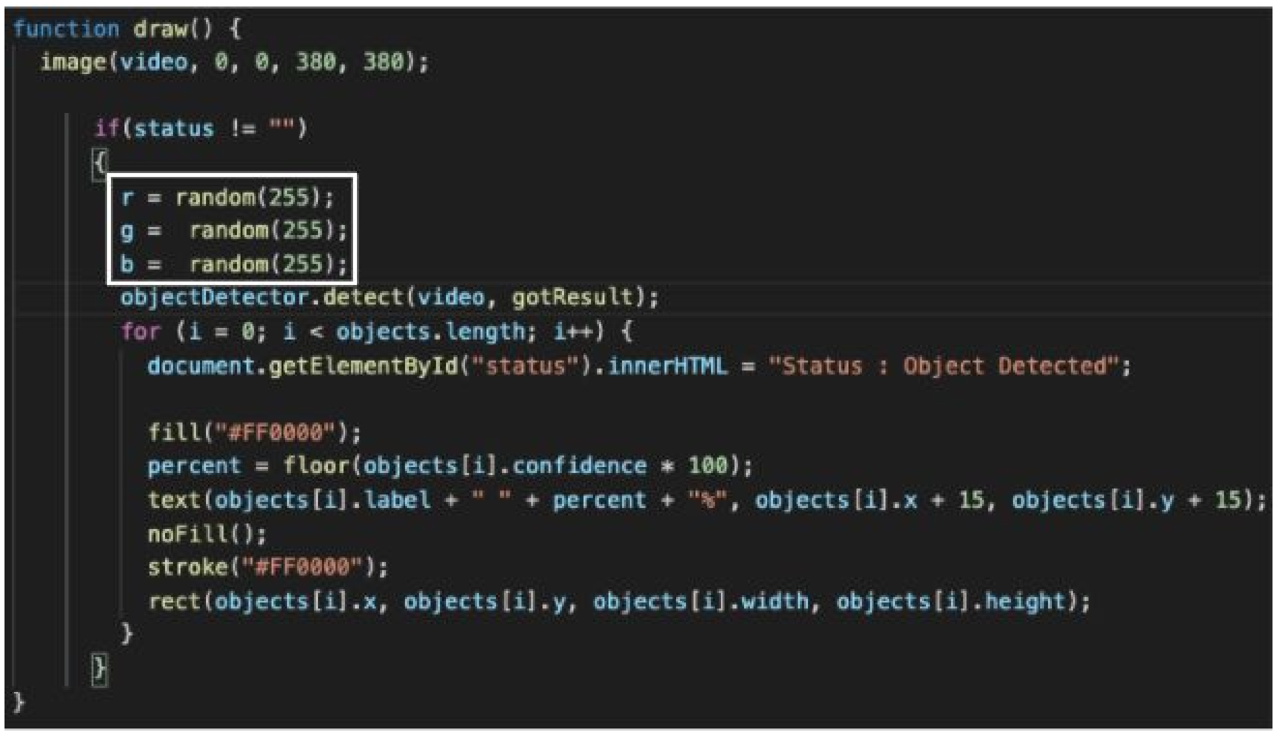
```
function modelLoaded() {
  console.log("Model Loaded!")
  status = true;
  objectDetector.detect(video, gotResult);
}
```

6. **Move the code of executing cocossd model inside draw() function**



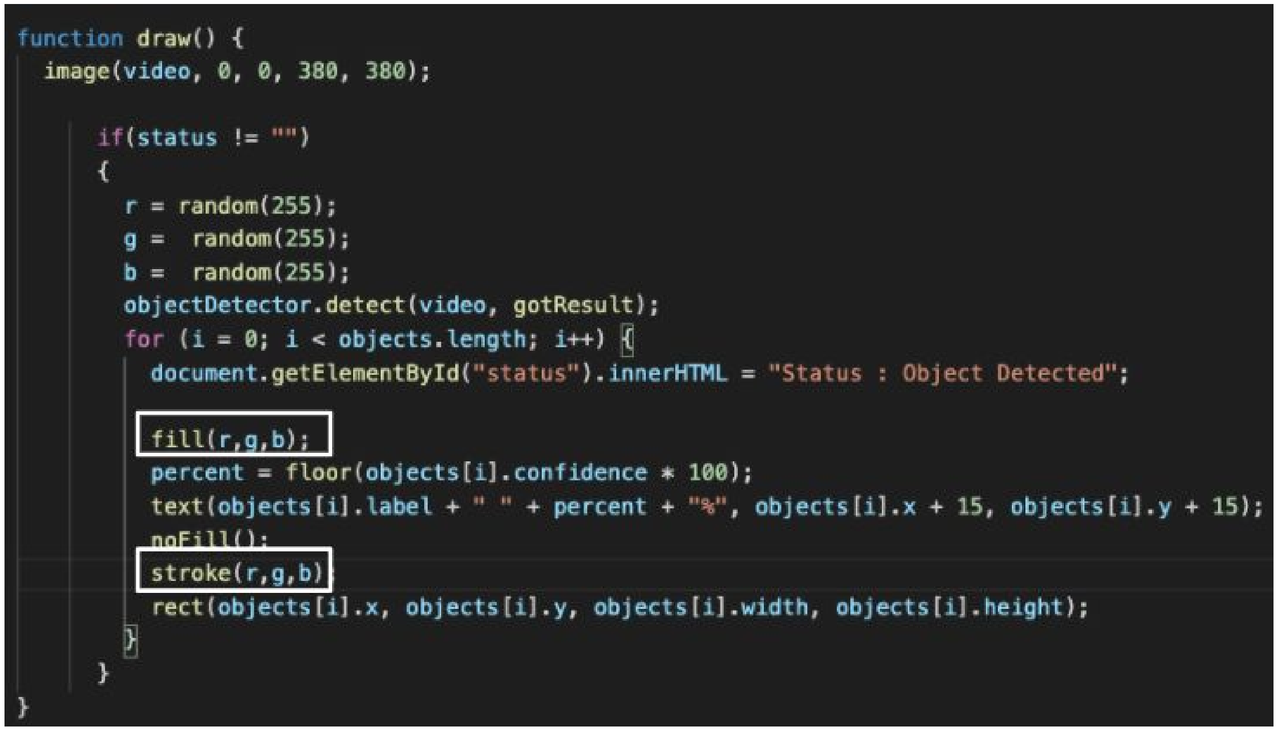
```
function draw() {
  image(video, 0, 0, 380, 380);

    if(status != "")
    {
      objectDetector.detect(video, gotResult);
      for (i = 0; i < objects.length; i++) {
        document.getElementById("status").innerHTML = "Status : Object Detected";

        fill("#FF0000");
        percent = floor(objects[i].confidence * 100);
        text(objects[i].label + " " + percent + "%", objects[i].x + 15, objects[i].y + 15);
        noFill();
        stroke("#FF0000");
        rect(objects[i].x, objects[i].y, objects[i].width, objects[i].height);
      }
    }
}
```

7. **Add code for generating random numbers for RGB and storing them in variables**



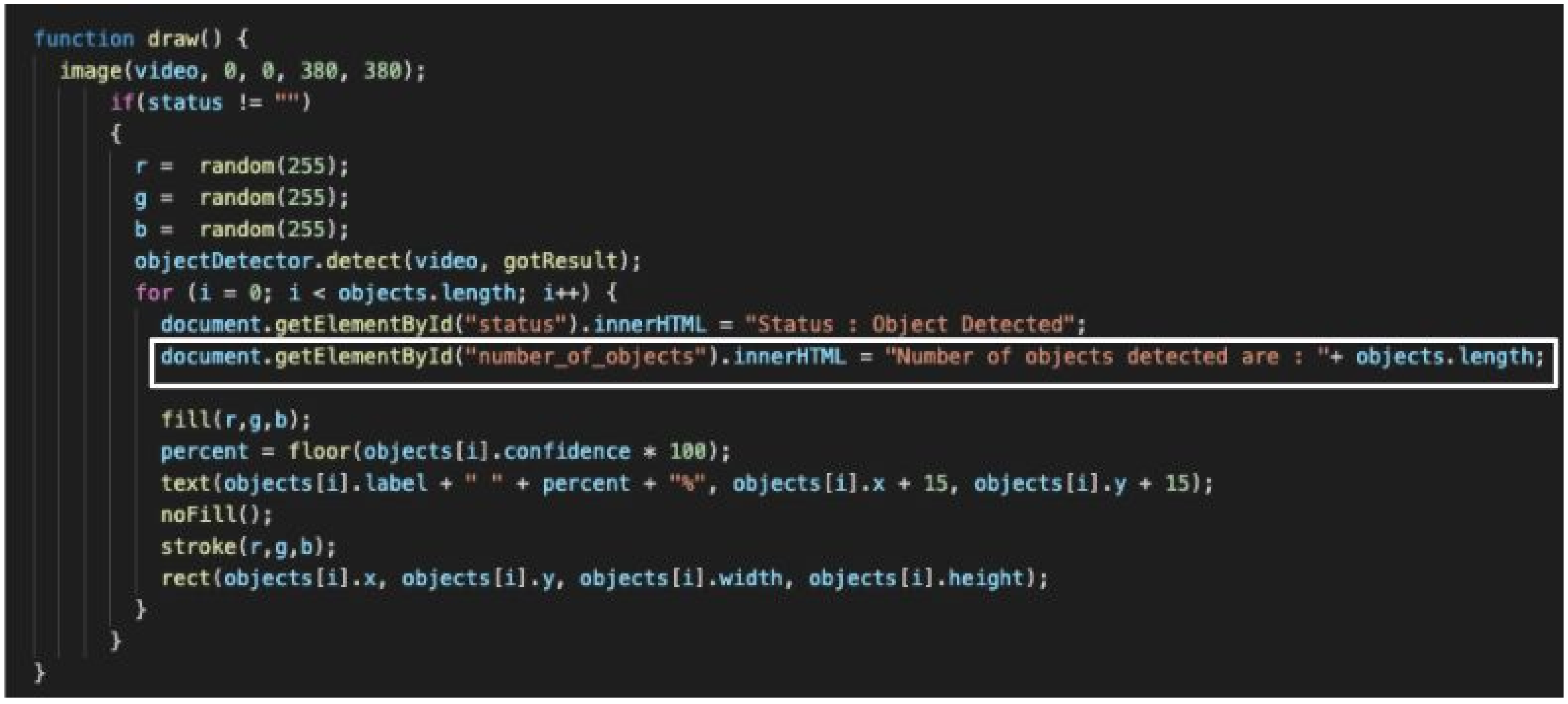
```
function draw() {
  image(video, 0, 0, 380, 380);

    if(status != "")
    {
      r = random(255);
      g =  random(255);
      b =  random(255);
      objectDetector.detect(video, gotResult);
      for (i = 0; i < objects.length; i++) {
        document.getElementById("status").innerHTML = "Status : Object Detected";

        fill("#FF0000");
        percent = floor(objects[i].confidence * 100);
        text(objects[i].label + " " + percent + "%", objects[i].x + 15, objects[i].y + 15);
        noFill();
        stroke("#FF0000");
        rect(objects[i].x, objects[i].y, objects[i].width, objects[i].height);
      }
    }
}
```

8. **Updating fill() and stroke() functions**



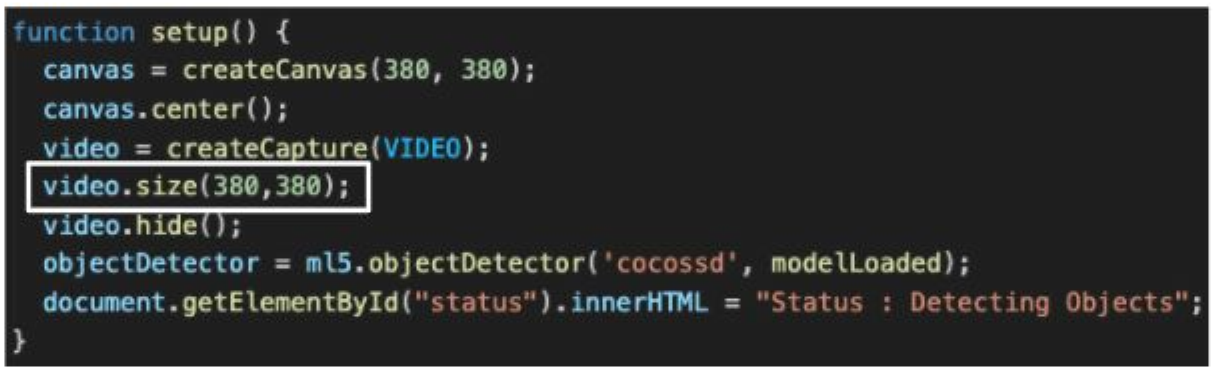
```
function draw() {
  image(video, 0, 0, 380, 380);

    if(status != "")
    {
      r = random(255);
      g =  random(255);
      b =  random(255);
      objectDetector.detect(video, gotResult);
      for (i = 0; i < objects.length; i++) {
        document.getElementById("status").innerHTML = "Status : Object Detected";

        fill(r,g,b);
        percent = floor(objects[i].confidence * 100);
        text(objects[i].label + " " + percent + "%", objects[i].x + 15, objects[i].y + 15);
        noFill();
        stroke(r,g,b);
        rect(objects[i].x, objects[i].y, objects[i].width, objects[i].height);
      }
    }
}
```

9. **Update h3 tag which is use to hold the number of objects**



```
function draw() {
  image(video, 0, 0, 380, 380);
    if(status != "")
    {
      r =  random(255);
      g =  random(255);
      b =  random(255);
      objectDetector.detect(video, gotResult);
      for (i = 0; i < objects.length; i++) {
        document.getElementById("status").innerHTML = "Status : Object Detected";
        document.getElementById("number_of_objects").innerHTML = "Number of objects detected are : "+ objects.length;

        fill(r,g,b);
        percent = floor(objects[i].confidence * 100);
        text(objects[i].label + " " + percent + "%", objects[i].x + 15, objects[i].y + 15);
        noFill();
        stroke(r,g,b);
        rect(objects[i].x, objects[i].y, objects[i].width, objects[i].height);
      }
    }
}
```

10. **Add size() function to get more accuracy in drawing the rectangle and placing the label**



```
function setup() {
  canvas = createCanvas(380, 380);
  canvas.center();
  video = createCapture(VIDEO);
  video.size(380,380);
  video.hide();
  objectDetector = ml5.objectDetector('cocossd', modelLoaded);
  document.getElementById("status").innerHTML = "Status : Detecting Objects";
}
```