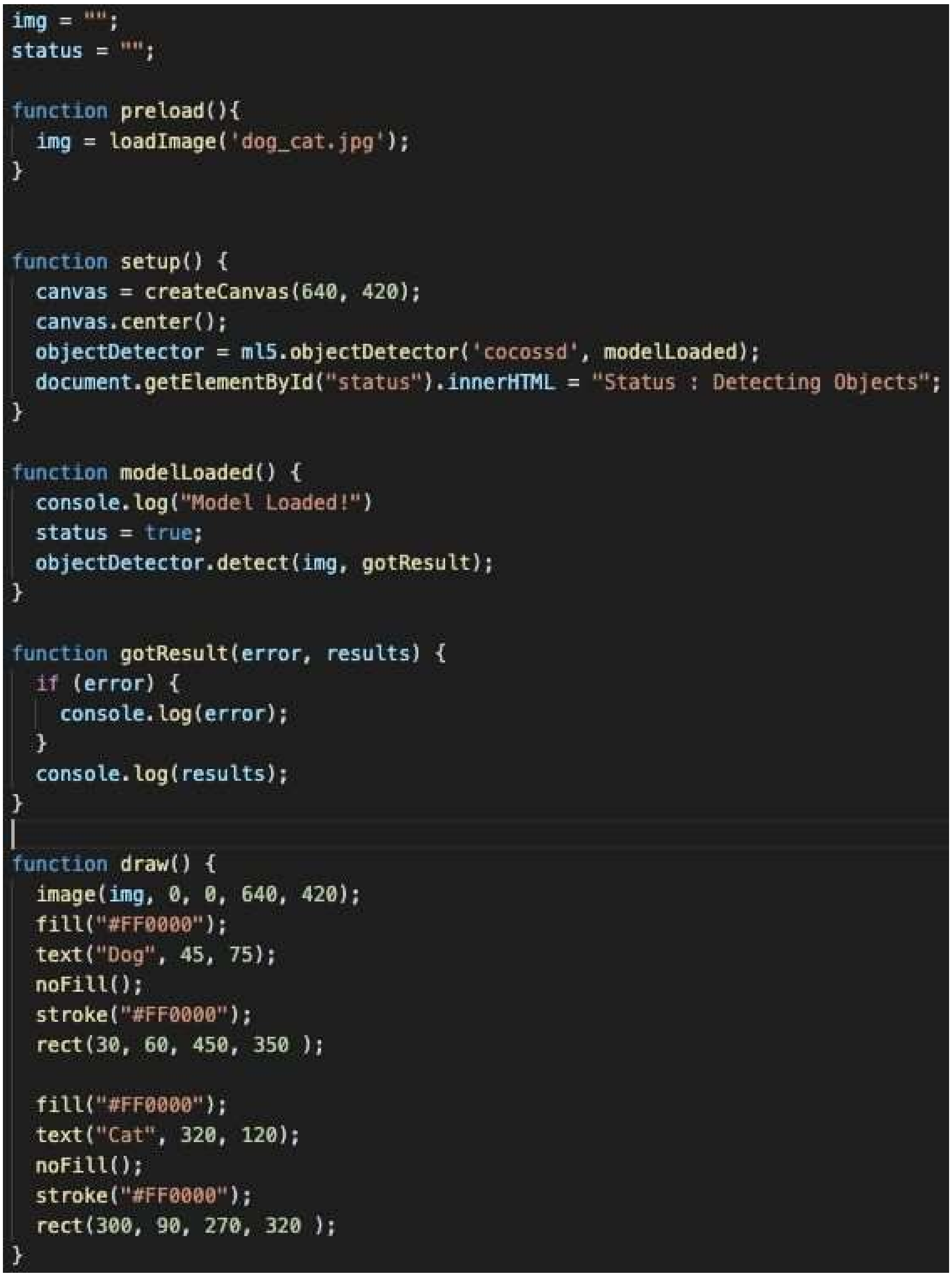
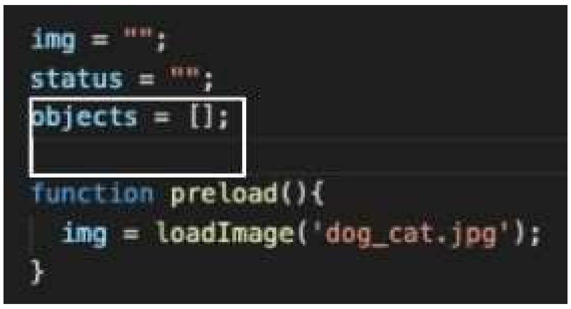
## Previous class JS code

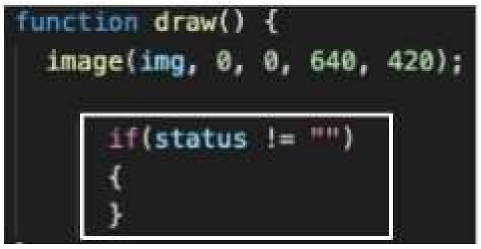


```
img = "";
status = "";

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);
}

function draw() {
  image(img, 0, 0, 640, 420);
  fill("#FF0000");
  text("Dog", 45, 75);
  noFill();
  stroke("#FF0000");
  rect(30, 60, 450, 350 );

  fill("#FF0000");
  text("Cat", 320, 120);
  noFill();
  stroke("#FF0000");
  rect(300, 90, 270, 320 );
}
```

1. **Define a empty array at the beginning of main.js file**



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

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



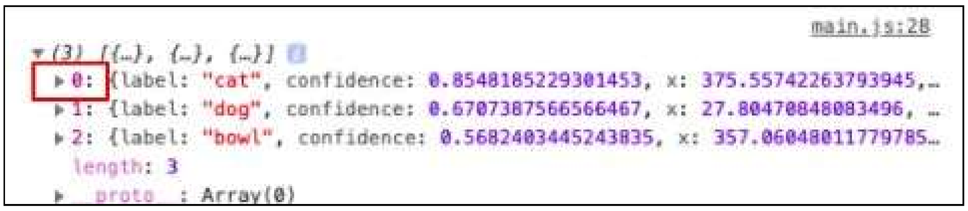
```
function draw() {
  image(img, 0, 0, 640, 420);

    if(status != "")
    {
    }
```

So in the “if condition” we are checking that “if”[`if(`] status variable[`status`] is no

empty[`"")`] then it should go inside this “if condition” and start drawing rectangles

**How to fetch the label of the first array from the objects ar**

1. We want to read the values of objects so first we will write `objects`
2. Inside objects array we want to get the first array, and first array is at 0 index, s clicking on the arrow next to `▸ 0:`

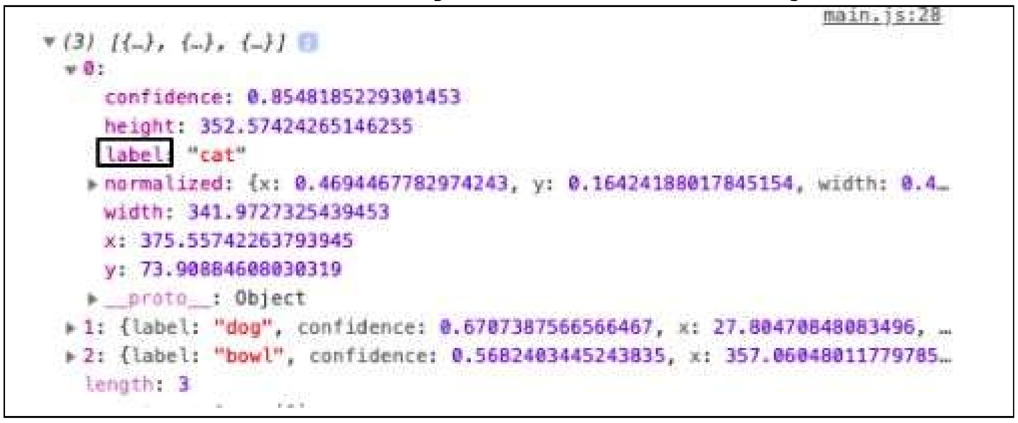


```
                                                               main.js:28
▼(3) [{…}, {…}, {…}]
  ▸0: {label: "cat", confidence: 0.8548185229301453, x: 375.55742263793945,…
  ▸1: {label: "dog", confidence: 0.6707387566566467, x: 27.80470848083496, …
  ▸2: {label: "bowl", confidence: 0.5682403445243835,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

This model keeps on updating, so it might not detect any one of the above things mentioned and i that’s fine. You continue with the code for the number of arrays returned

We have clicked on 0 index which is inside the objects array, so code will be `objects`

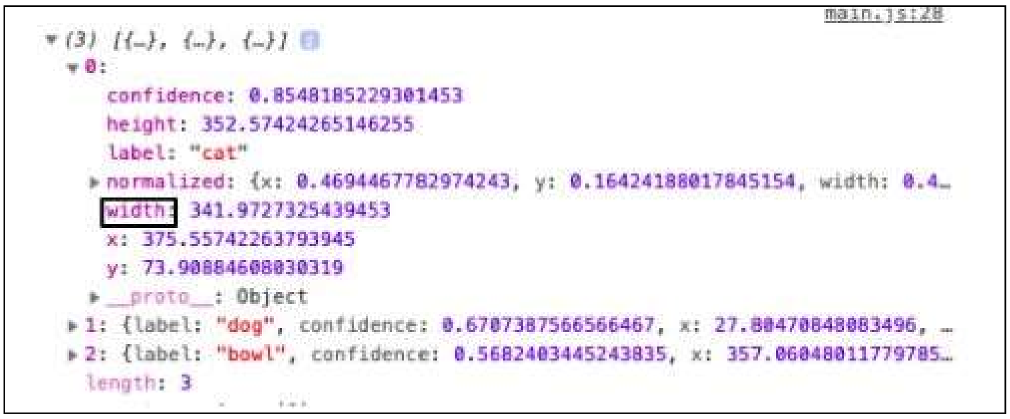
3. Inside the first array we have a label object, this label is of the first object whic



```
                                                               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
```

So we had to click on objects- > then 0 index -> then there is label.

So the code will be - `objects[0].label`

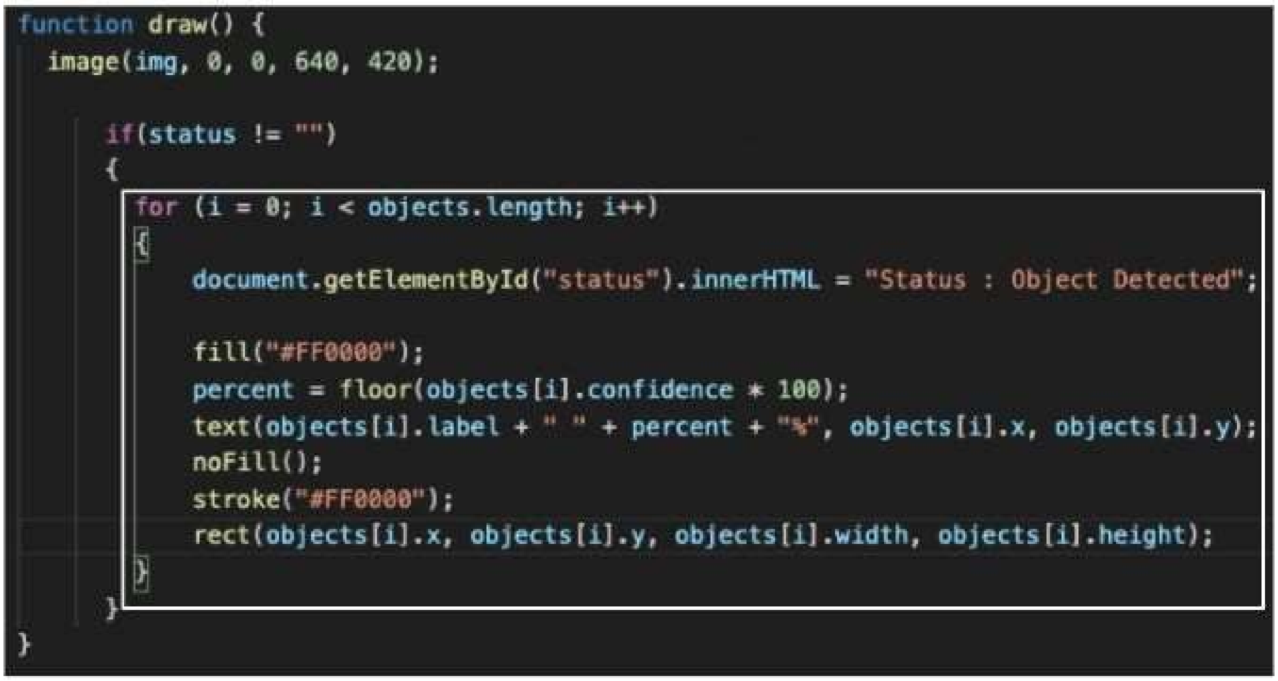


```
                                                                   main.js:28
▼(3) [{…}, {…}, {…}]
  ▼0:
     confidence: 0.8548185229301453
     height: 352.57424265146255
     label: "cat"
   ▶normalized: {x: 0.4694467782974243, y: 0.16424188017845154, width: 0.4…
     width: 341.9727325439453
     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
```

**So we had to click on objects > then 0 index -> then there is width.**

**So the code will be -** `objects[0].width`

**For-Loop Code -**

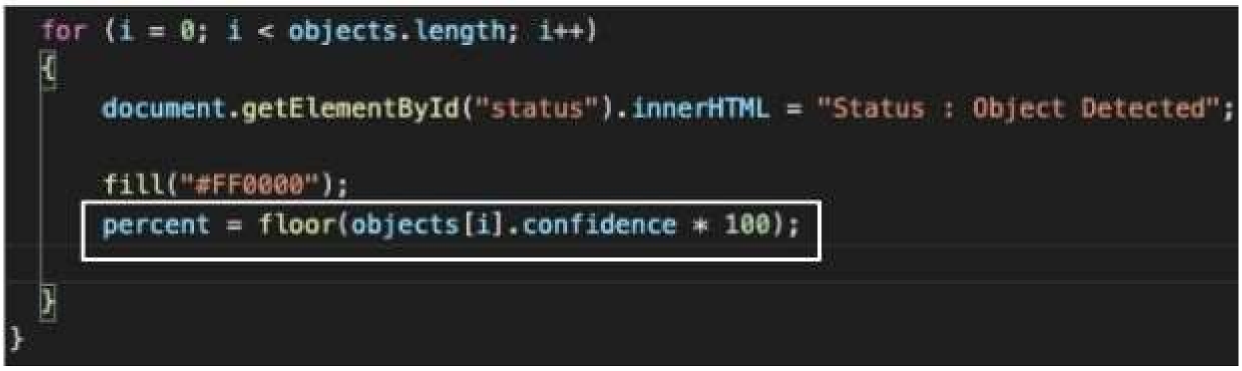


```
function draw() {
  image(img, 0, 0, 640, 420);

      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, objects[i].y);
            noFill();
            stroke("#FF0000");
            rect(objects[i].x, objects[i].y, objects[i].width, objects[i].height);
        }
      }
}
```

**Breaking down the above code -**

1. **Define for-loop -** `for (i = 0; i < objects.length; i++)`
   - **Set the start point for for-loop -** `i = 0;`
   - **Set the stop point for the for loop -** `i < objects.length;`
   - **Set the interval between each loop -** `i++)`

2. **Update the h3 tag with “Status: Object Detected”**



```
for (i = 0; i < objects.length; i++)
{

    document.getElementById("status").innerHTML = "Status : Object Detected";

    fill("#FF0000");
    percent = floor(objects[i].confidence * 100);

}
}
```

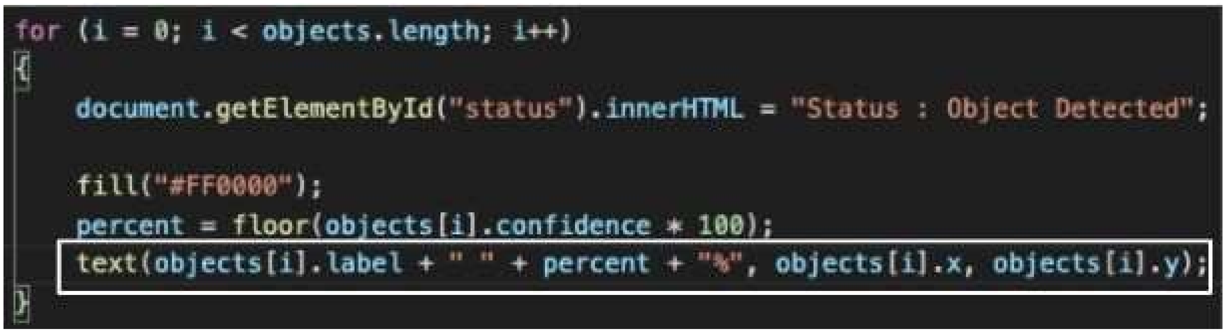
- **Code for fetching confidence from the objects array in the for loop** `(objects[i].c`

  **Explaining the above line -**
  **For eg - Let's consider there are 2 arrays inside the objects array. Means the le**
- **When the loop starts  i = 0**
  - **objects[i].confidence will become objects[0].confidence  // this means we first object.**
- **Then i is incremented and  i = 1**
  - **objects[i].confidence will become objects[1].confidence  // this means we second object.**

- **Then i is incremented and  i =2**
  - **The length of the array is 2, means the loop terminates.**

- **Converting confidence into percentage** `objects[0].confidence * 100`
- **Removing all the decimals -** `floor(objects[i].confidence * 100);`
- **Storing inside a variable** `percent = floor(objects[i].confidence * 100);`

5. **Fetch the label from objects array, and display the label and confidence for all function**



```
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, objects[i].y);
}
```

- **Code for fetching label from the objects array in the for loop** `(objects[i].label`

  **Explaining the above line -**

- **A space - the code will be -** `text(objects[i].label + " "`

- **Then percentage - the code will be - the code will be -** `text(objects[i].label + "`

- **The the symbol of the percentage - the code will be -**

```
text(objects[i].label + " " + percent + "%",
```

**Fetching x coordinates and passing inside text() function**

- **Code for fetching label from the objects array in the for loop** `objects[i].x`

  **Explaining the above line -**
  **For eg - Let's consider there are 2 arrays inside the objects array. Means the le**
  - **When the loop starts  i = 0**
    - **objects[i].x will become objects[0].x  // this means we got the x coor**
  - **Then i is incremented and  i = 1**
    - **objects[i].x will become objects[1].x  // this means we got the x coor object.**

  - **Then i is incremented and  i =2**
    - **The length of the array is 2, means the loop terminates.**

**Now let's pass this fetched x coordinate inside the text() function**

```
text(objects[i].label + " " + percent + "%", objects[i].x
```

**Fetching y coordinates and passing inside text() function**

- **Code for fetching label from the objects array in the for loop** `objects[i].y`

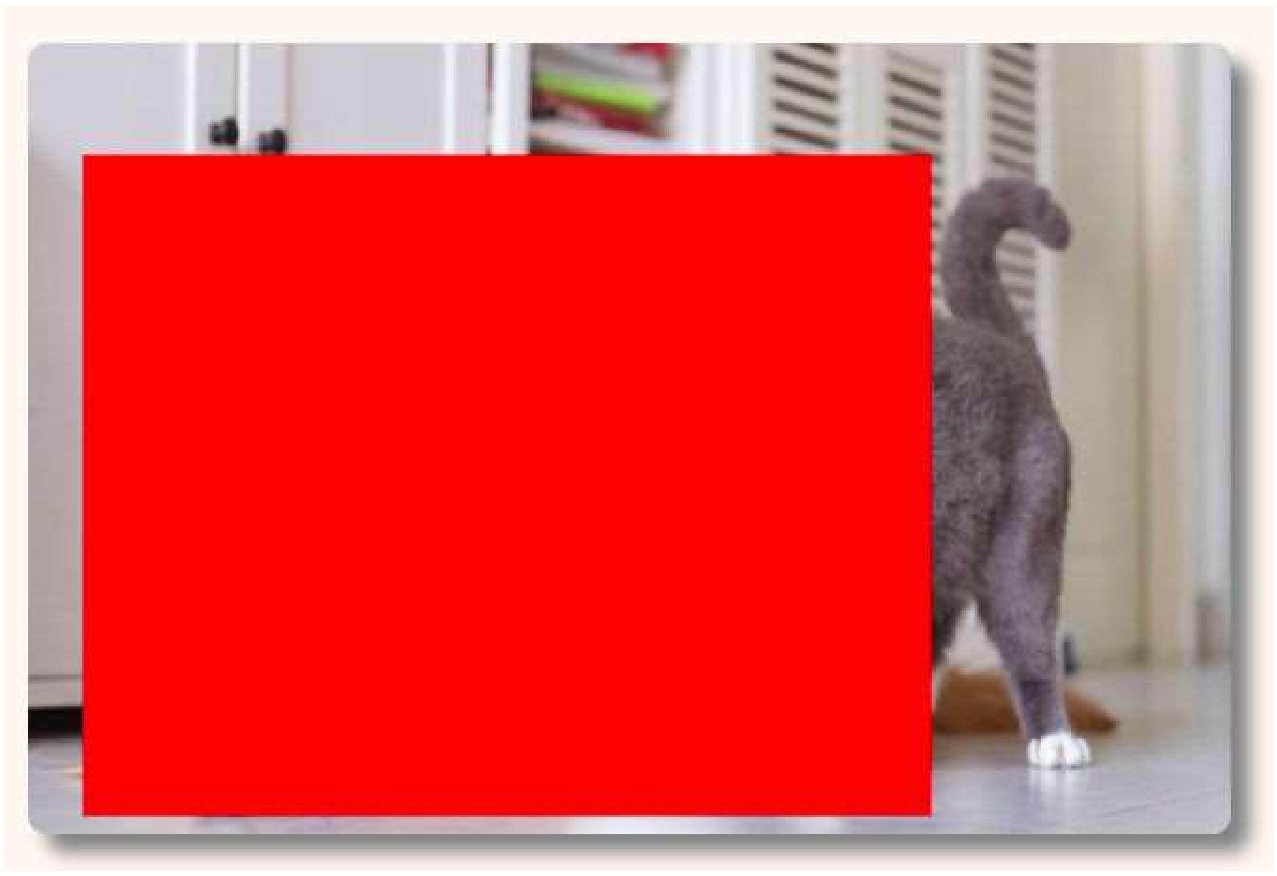
  **Explaining the above line -**

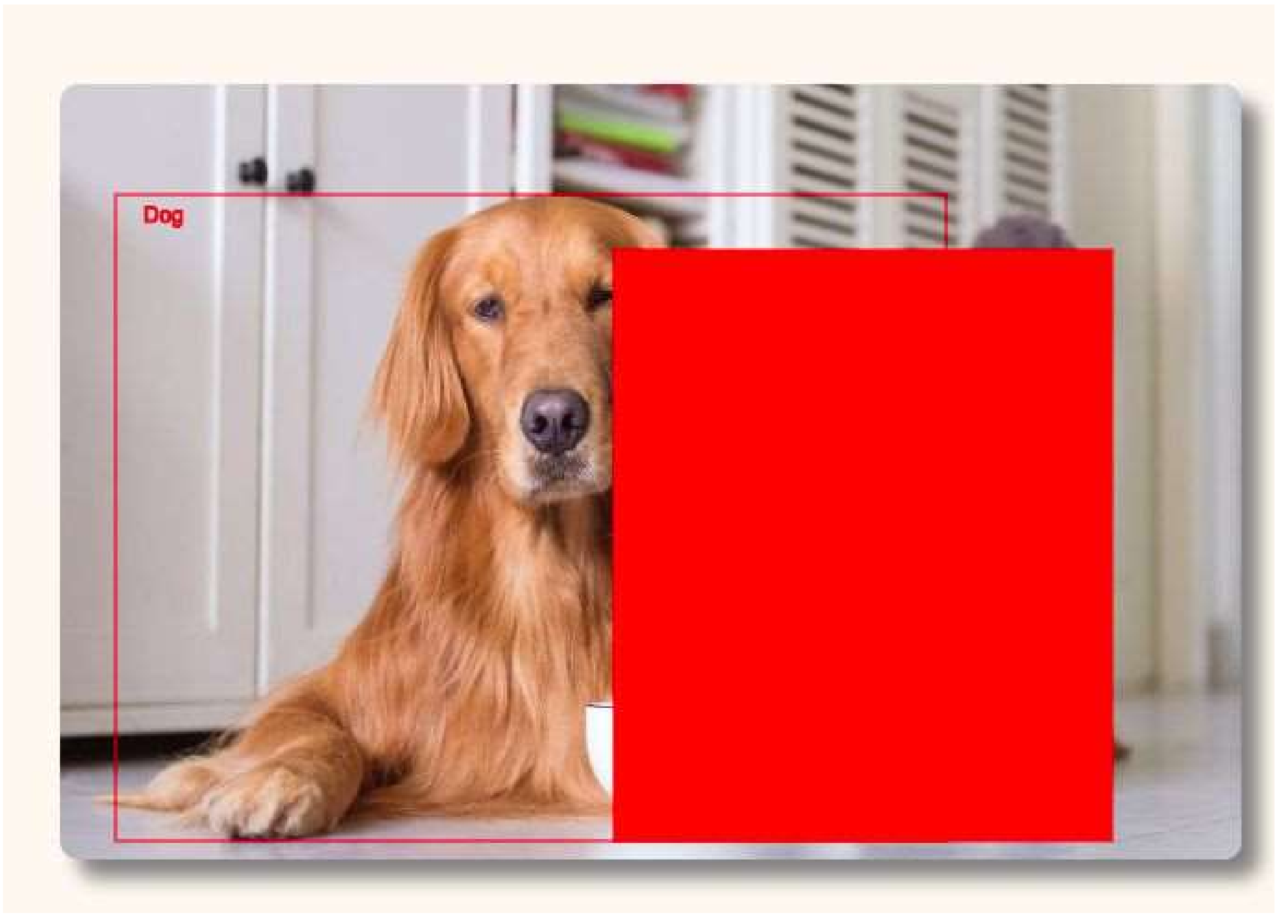
**Now let's pass this fetched y coordinate inside the text() function**

```
text(objects[i].label + " " + percent + "%", objects[i].x, objects[i].y);
```

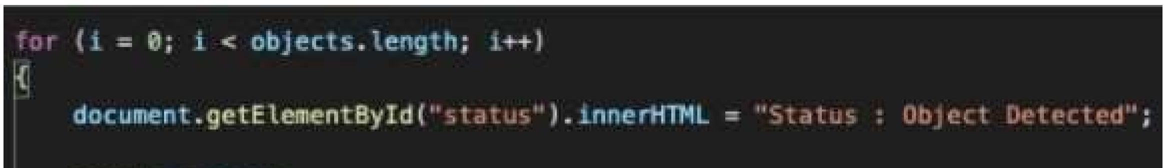
6. **Unset the set color by fill() function using p5.js onFill() function**

**If we don't do this the output will come like this -**

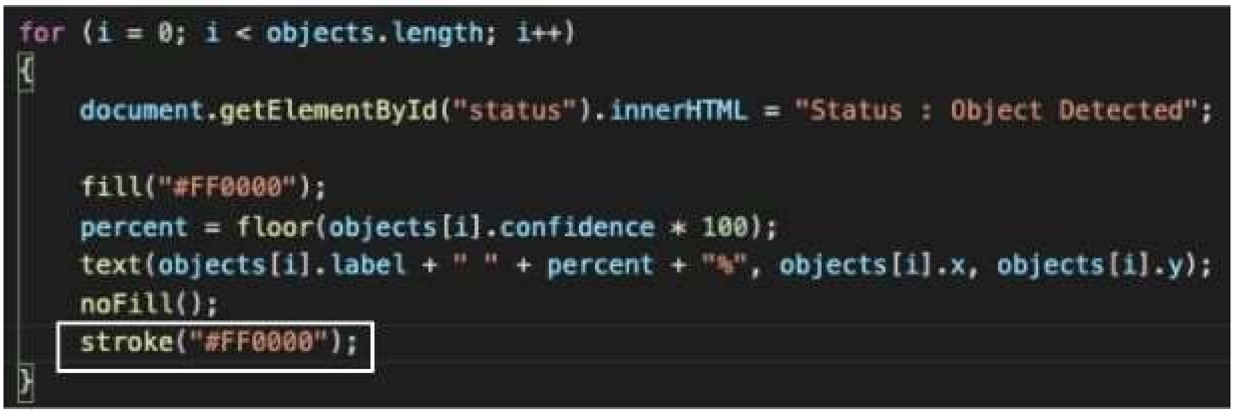




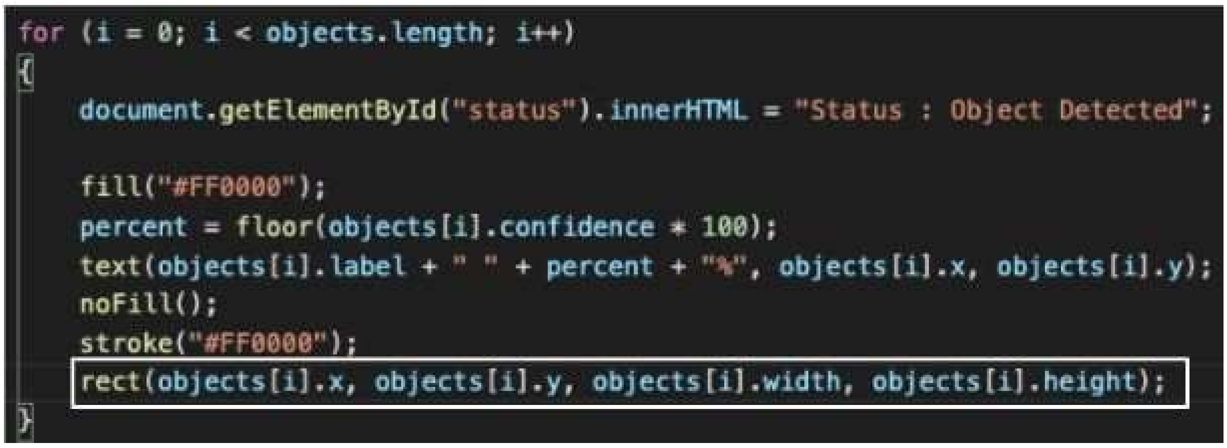
## Code for unsetting the color set by fill() function -



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


```
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, objects[i].y);
    noFill();
    stroke("#FF0000");
}
```

8. **Draw a rectangle for all the objects using rect() function**


```
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, objects[i].y);
    noFill();
    stroke("#FF0000");
    rect(objects[i].x, objects[i].y, objects[i].width, objects[i].height);
}
```

**Code for fetching x coordinate from objects array in the for-loop will be** `objects[i].`

**Code for fetching y coordinate from objects array in the for-loop will be** `objects[i].y`

**Code for fetching width from objects array in the for-loop will be** `objects[i].width` **b**

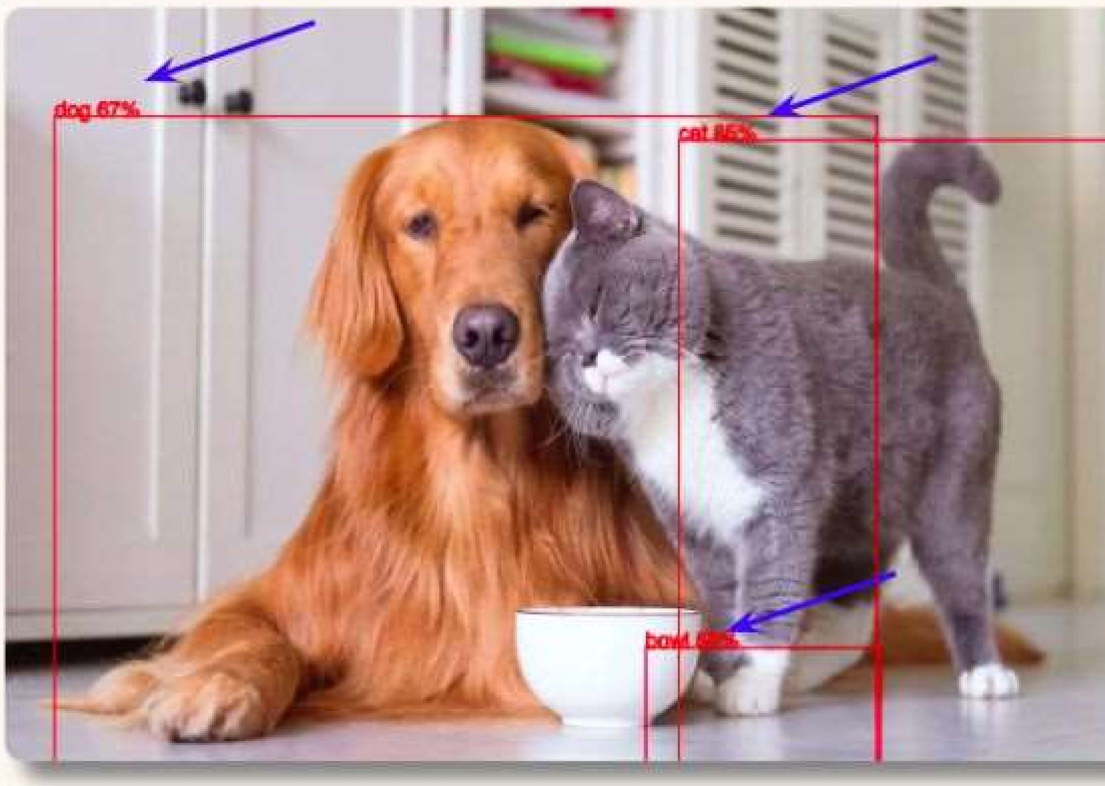
**For eg - Let's consider there are 2 arrays inside the objects array. Means the le**

- **When the loop starts  i = 0**
  - **objects[i].width will become objects[0].width  // this means we got th**
- **Then i is incremented and  i = 1**
  - **objects[i].width will become object[1].width  // this means we got the object.**
- **Then i is incremented and  i =2**
  - **The length of the array is 2, means the loop terminates.**

**Code for fetching height from objects array in the for-loop will be** `objects[i].height`

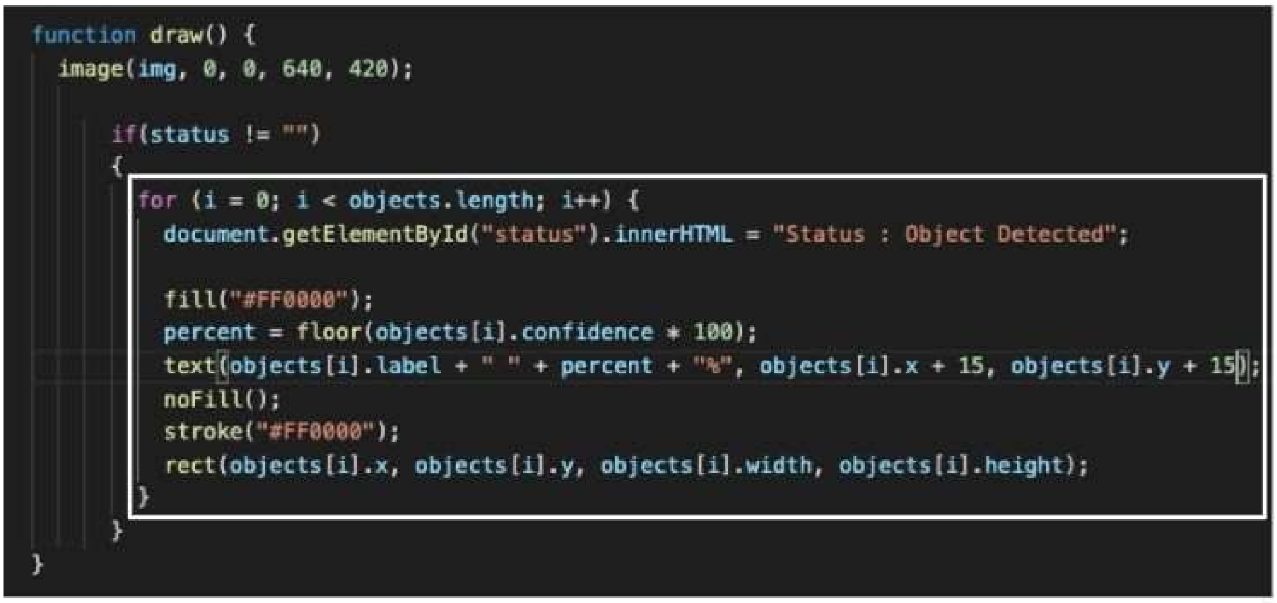
**For eg - Let's consider there are 2 arrays inside the objects array. Means the le**

- **When the loop starts  i = 0**
  - **objects[i].height will become objects[0].height  // this means we got**



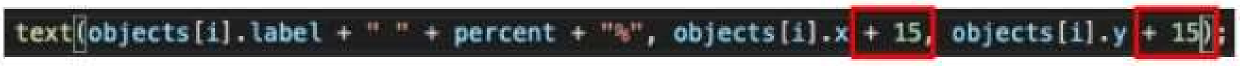
**This model keeps on updating, so it might not detect any one of the above things mentioned like it**
**some other object from the image, so that is fine. You continue with the code.**

## Code -



```
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);
     }
    }
}
```

## Change the x and y coordinates of text() by adding 15 pixels -



```
text(objects[i].label + " " + percent + "%", objects[i].x + 15, objects[i].y + 15);
```