

Origin

6 min

The origin of the shape determines which point the shape is transformed from. For example, some built-in p5.js drawing functions, such as `rect()`, are oriented around the upper left corner of the shape—but you can always change your shape's origin point.

The `rectMode()` and `ellipseMode()` functions allow you to modify the location from which the shape is drawn by changing how the x and y arguments passed to `rect()` and `ellipse()` functions are interpreted.

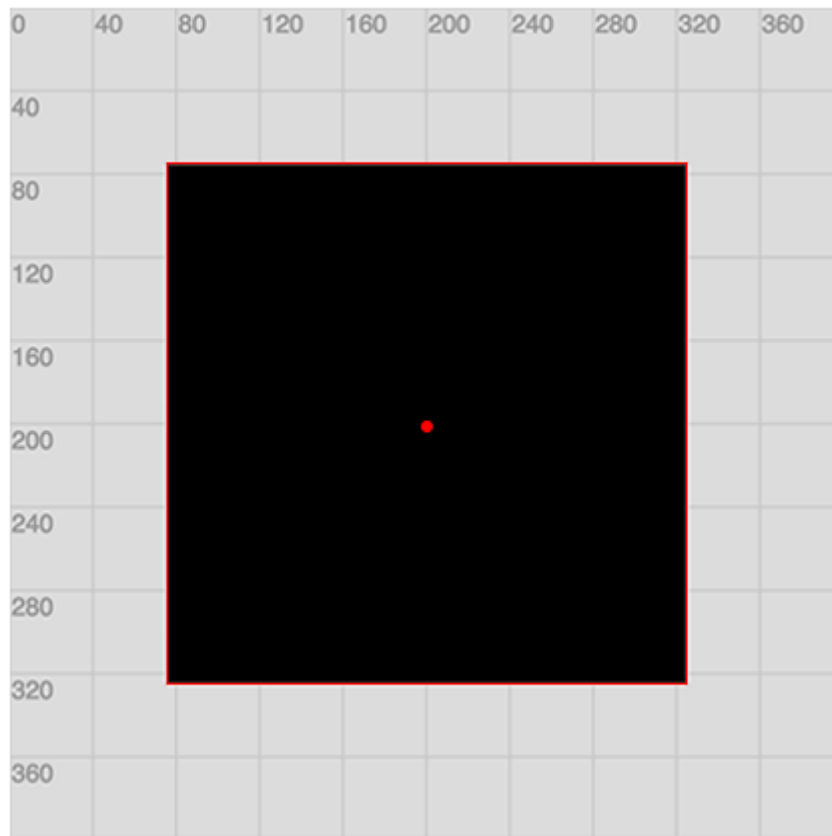
When the `rectMode(CENTER)` function is called, the x and y coordinates passed to the `rect()` function changes to represent the center of the rectangle, instead of the top left corner.

The code below shows how to specify a rectangle's origin point to be its center:

```
rectMode(CENTER);  
rect(200, 200, 250, 250);
```

Here, the first [argument](#) of the `rect()` function represents the x position of the origin point, which is now the horizontal center of the rectangle. The second argument, which represents the y position of the origin point, is now the vertical center of the rectangle. Since p5.js reads the code from top to bottom, the origin of a shape should be determined before the shape is drawn.

The diagram below shows that the rectangle's origin point is now in the center of the rectangle.

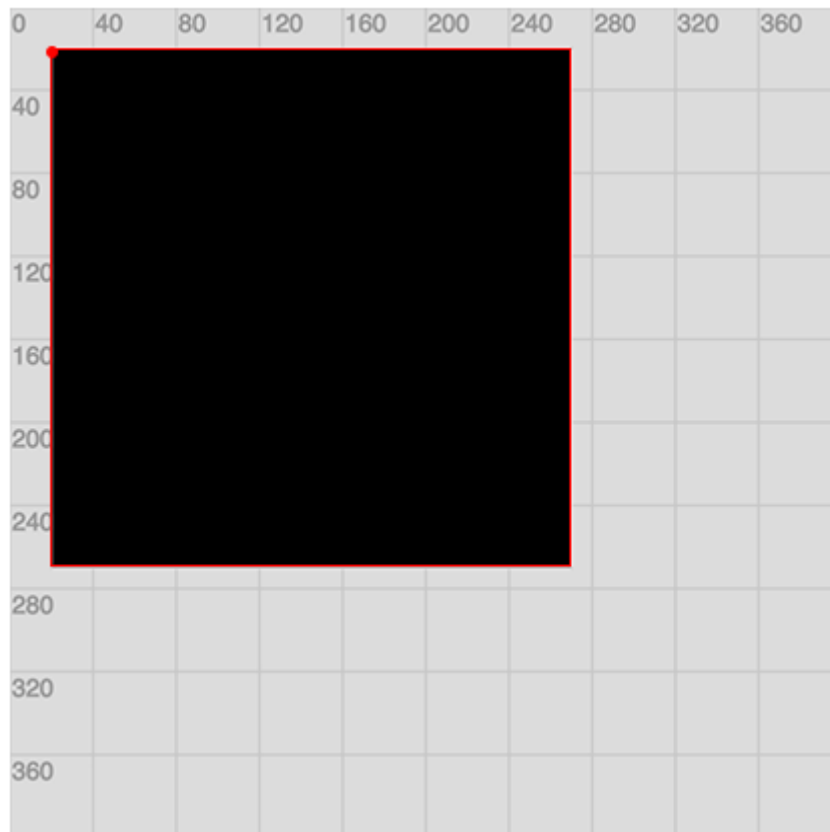


With `rectMode(CORNER)`, the x and y coordinates passed into the `rect()` function represents the rectangle's top-left corner. If you don't explicitly call `rectMode(CORNER)`, p5.js will automatically assign the origin to be the `CORNER`.

```
rectMode(CORNER);  
rect(20,20,250,250);
```

The above code will produce the same results as the code below, which draws the rectangle without specifying the `rectMode(CORNER)`. Both examples result in the rectangle having a top left corner orientation.

```
rect(20,20,250,250);
```



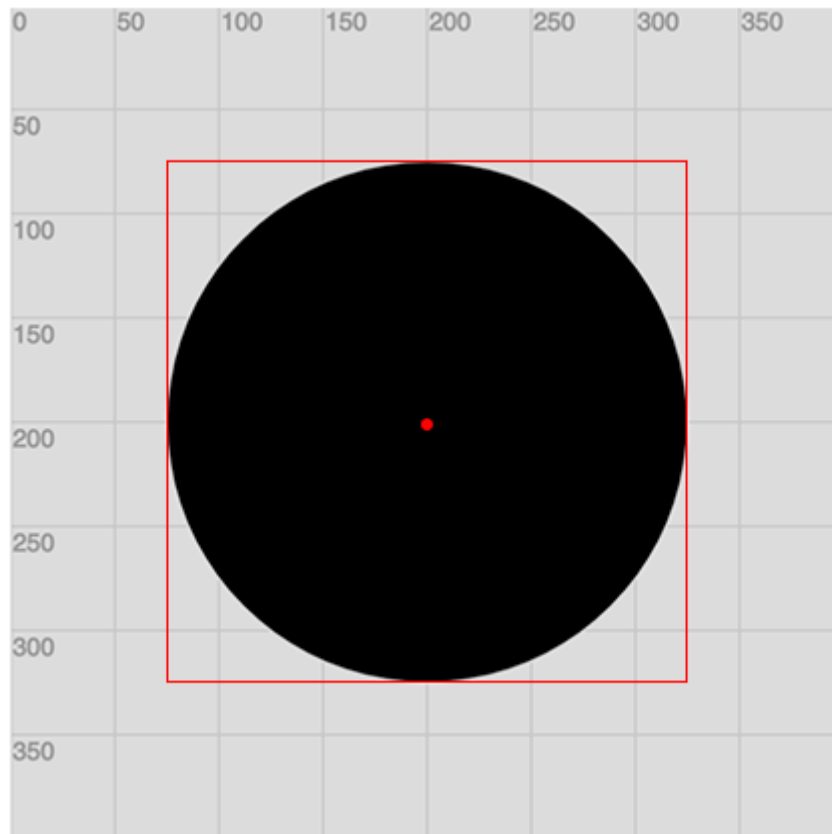
In p5.js, ellipses are automatically drawn from the shape's center. The first and second values that are passed to the `ellipse()` function represent the x and y coordinates of its center point.

The code below explicitly calls `ellipseMode(CENTER)` which keeps the ellipse's origin point at the shape's center.

```
ellipseMode(CENTER);  
ellipse(200,200,250,250);
```

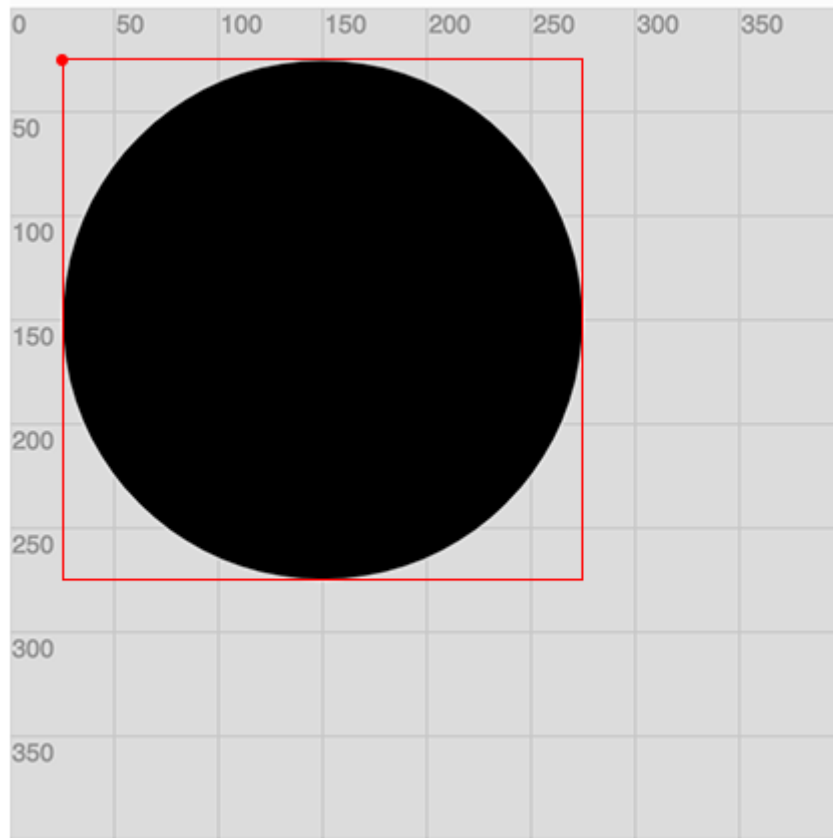
Above code will produce the same results as the one below, which draws the ellipse without specifying `ellipseMode(CENTER)`.

```
ellipse(200,200,250,250);
```



You can also change the ellipse's origin to be from its center to its top-left corner by giving `CORNER` as the argument of the `ellipseMode()` function. When you use `ellipseMode(CORNER)`, the arguments for the x and y positions given to the `ellipse()` function will represent the top-left corner of the bounding box of the ellipse.

```
ellipseMode(CORNER);  
ellipse(25,25,250,250);
```



Instructions

1.

In the line above the `ellipse()` function, specify the origin point of the ellipse to be from the `CORNER`.

Notice how the position of the ellipse changes.

Hint

You can use the `ellipseMode()` function to specify an ellipse's origin point.

2.

In the line above the `rect()` function, specify the origin point of the rectangle to be from the `CENTER`.

Notice how the position of the rectangle changes.

Hint

You can use the `rectMode()` function to specify a rectangle's origin point.

sketch.js

```
function setup() {  
  createCanvas(windowWidth, windowHeight);  
}  
  
function draw() {  
  background(0);  
  
  for (let i = 0; i < width; i += 75) {  
    for (let j = 0; j < height; j += 75) {  
      // Create red ellipse pattern  
      fill(255, 0, 0, 180);  
      // TODO: Specify a corner origin for the ellipse  
      ellipseMode(CORNER);  
      ellipse(i, j, 50, 50);  
  
      // Create blue square pattern  
      fill(0, 0, 255, 180);  
      // TODO: Specify a center origin for the rectangle  
      rectMode(CENTER);  
      rect(i, j, 50, 50);  
    }  
  }  
}
```