

imageMode()

Changes the location from which images are drawn when [image\(\)](#) is called.

By default, the first two parameters of [image\(\)](#) are the x- and y-coordinates of the image's upper-left corner. The next parameters are its width and height. This is the same as calling [imageMode\(CORNER\)](#).

[imageMode\(CORNERS\)](#) also uses the first two parameters of [image\(\)](#) as the x- and y-coordinates of the image's top-left corner. The third and fourth parameters are the coordinates of its bottom-right corner.

[imageMode\(CENTER\)](#) uses the first two parameters of [image\(\)](#) as the x- and y-coordinates of the image's center. The next parameters are its width and height.

Examples



```
let img;

// Load the image.
function preload() {
  img = loadImage('/assets/bricks.jpg');
}

function setup() {
  createCanvas(100, 100);

  background(200);

  // Use CORNER mode.
  imageMode(CORNER);

  // Display the image.
  image(img, 10, 10, 50, 50);

  describe('A square image of a brick wall is drawn at the top left of a gray square.');
}
```



```
let img;

// Load the image.
function preload() {
  img = loadImage('/assets/bricks.jpg');
}

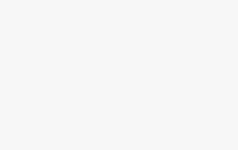
function setup() {
  createCanvas(100, 100);

  background(200);

  // Use CORNERS mode.
  imageMode(CORNERS);

  // Display the image.
  image(img, 10, 10, 90, 40);

  describe('An image of a brick wall is drawn on a gray square. The image is squeezed into a small rectangular area.');
}
```



```
let img;

// Load the image.
function preload() {
  img = loadImage('/assets/bricks.jpg');
}

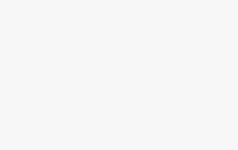
function setup() {
  createCanvas(100, 100);

  background(200);

  // Use CENTER mode.
  imageMode(CENTER);

  // Display the image.
  image(img, 50, 50, 80, 80);

  describe('A square image of a brick wall is drawn on a gray square.');
}
```



Syntax

```
imageMode(mode)
```



Parameters

mode Constant: either CORNER, CORNERS, or CENTER.

This page is generated from the comments in [src/image/loading_displaying.js](#). Please feel free to edit it and submit a pull request!

Related References

[font](#) The font's underlying opentype.js font object.

[textBounds](#) Returns the bounding box for a string of text written using the font.

[textToPoints](#) Returns an array of points outlining a string of text written using the font.

[image](#)

Draws an image to the canvas.

