

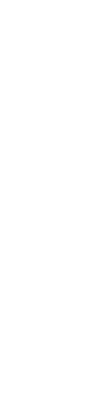
redraw()

Runs the code in `draw()` once.

By default, `draw()` tries to run 60 times per second. Calling `noLoop()` stops `draw()` from repeating. Calling `redraw()` will execute the code in the `draw()` function a set number of times.

The parameter, `n`, is optional. If a number is passed, as in `redraw(5)`, then the draw loop will run the given number of times. By default, `n` is 1.

Examples



```
// Double-click the canvas to move the circle.

let x = 0;

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

  // Turn off the draw loop.
  noLoop();

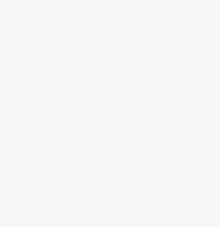
  describe(
    'A white half-circle on the left edge of a gray square.  
The circle moves a little to the right when the user double-clicks.'
  );
}

function draw() {
  background(200);

  // Draw the circle.
  circle(x, 50, 20);

  // Increment x.
  x += 5;
}

// Run the draw loop when the user double-clicks.
function doubleClicked() {
  redraw();
}
```



```
// Double-click the canvas to move the circle.

let x = 0;

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

  // Turn off the draw loop.
  noLoop();

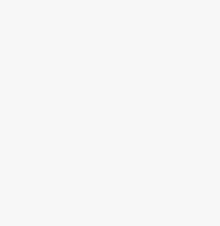
  describe(
    'A white half-circle on the left edge of a gray square.  
The circle hops to the right when the user double-clicks.'
  );
}

function draw() {
  background(200);

  // Draw the circle.
  circle(x, 50, 20);

  // Increment x.
  x += 5;
}

// Run the draw loop three times when the user double-clicks.
function doubleClicked() {
  redraw(3);
}
```



Syntax

```
redraw([n])
```



Parameters

`n` Integer: number of times to run `draw()`. Defaults to 1.

This page is generated from the comments in `src/core/structure.js`. Please feel free to edit it and submit a pull request!

Related References

`disableFriendlyErrors`
Turns off the parts of the Friendly Error System (FES) that impact performance.

`draw`
A function that's called repeatedly while the sketch runs.

`isLooping`
Returns true if the draw loop is running and false if not.

`loop`
Resumes the draw loop after `noLoop()` has been called.

