Potato Print Programming

Processing P5 in editor.p5js.org

Becca Glowacki, Steve Battle

Draw!


```javascript
// setup() is run once when the program starts
// the draw() is run repeatedly

// create a drawing area with
// createCanvas(screen width, screen height)

function setup() {
  createCanvas(400, 400);
}

// Draw a circle, specifying centre and diameter
// screen coordinates start at x=0,y=0 for the top-left
// They go up to x=width, y=height at the bottom-right
// circle(x, y, diameter)

function draw() {
  circle(200, 200, 100);
}
```

Painting in colour

Dare to be square


```
function setup() {
  createCanvas(400, 400);
}

// A square is a rectangle with equal width and height
// A square's x,y 'anchor point' is its top-left corner
// rect(x, y, width, height)

// You can draw many shapes

function draw() {
  circle(200,200,100);
  rect(0, 0, 100, 100);
}
```

Triangulate

Origin story


```
function setup() {
  createCanvas(400, 400);
}

// We can translate the origin 0,0 to the centre
// We can also move the x,y anchor point of the
// square to its CENTER - US spelling only :(
// (the default mode is CORNER)
// This draws a square dead centre 0,0

function draw() {
  translate(200,200);
  rectMode(CENTER);
  rect(0,0,100,100);
}
```

Turn squares into diamonds


```javascript
function setup() {
  createCanvas(400, 400);
}

// We can rotate the canvas before drawing a square
// Invisibly rotate the canvas 45 degrees anti-clockwise
// It rotates about the origin 0,0 (now in the centre)
// (Imagine rotating your head 45 degrees to the left)
// The square appears to rotate 45 degrees clockwise
// The angle mode is RADIANS by default
// A circle is 360 degrees, or 2 * PI = TAU radians

function draw() {
  translate(200,200);
  angleMode(DEGREES);
  rotate(45);
  rectMode(CENTER);
  rect(0,0,100,100);
}
```

The circle dance


```
editor.p5js.org

p5*   File ▼   Edit ▼   Sketch ▼   Help ▼                    Hello, stevebattle!  |  My Account ▼

▶  ■   ☐ Auto-refresh   seven potato ✎ by stevebattle                              ⚙

>  sketch.js                              Saved: 34 minutes ago    Preview

 1▽ function setup() {
 2    createCanvas(400, 400);
 3 }
 4
 5  // We can rotate the canvas around the centre
 6  // to draw shapes at a series of angles
 7  // Each square is offset x=0, y=-100 (100 upwards)
 8  // from the centre of rotation
 9
10▽ function draw() {
11    translate(200,200);
12    angleMode(DEGREES);
13    rectMode(CENTER);
14    circle(0,0,100);
15    rect(0,-100,50,50);
16    rotate(45);
17    rect(0,-100,50,50);
18    rotate(45);
19    rect(0,-100,50,50);
20 }
21

Console                                   Clear ▼
```
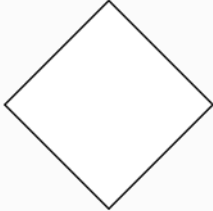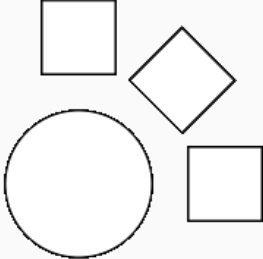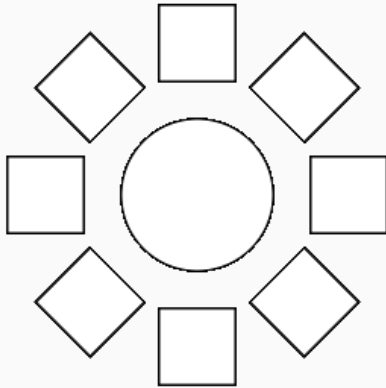
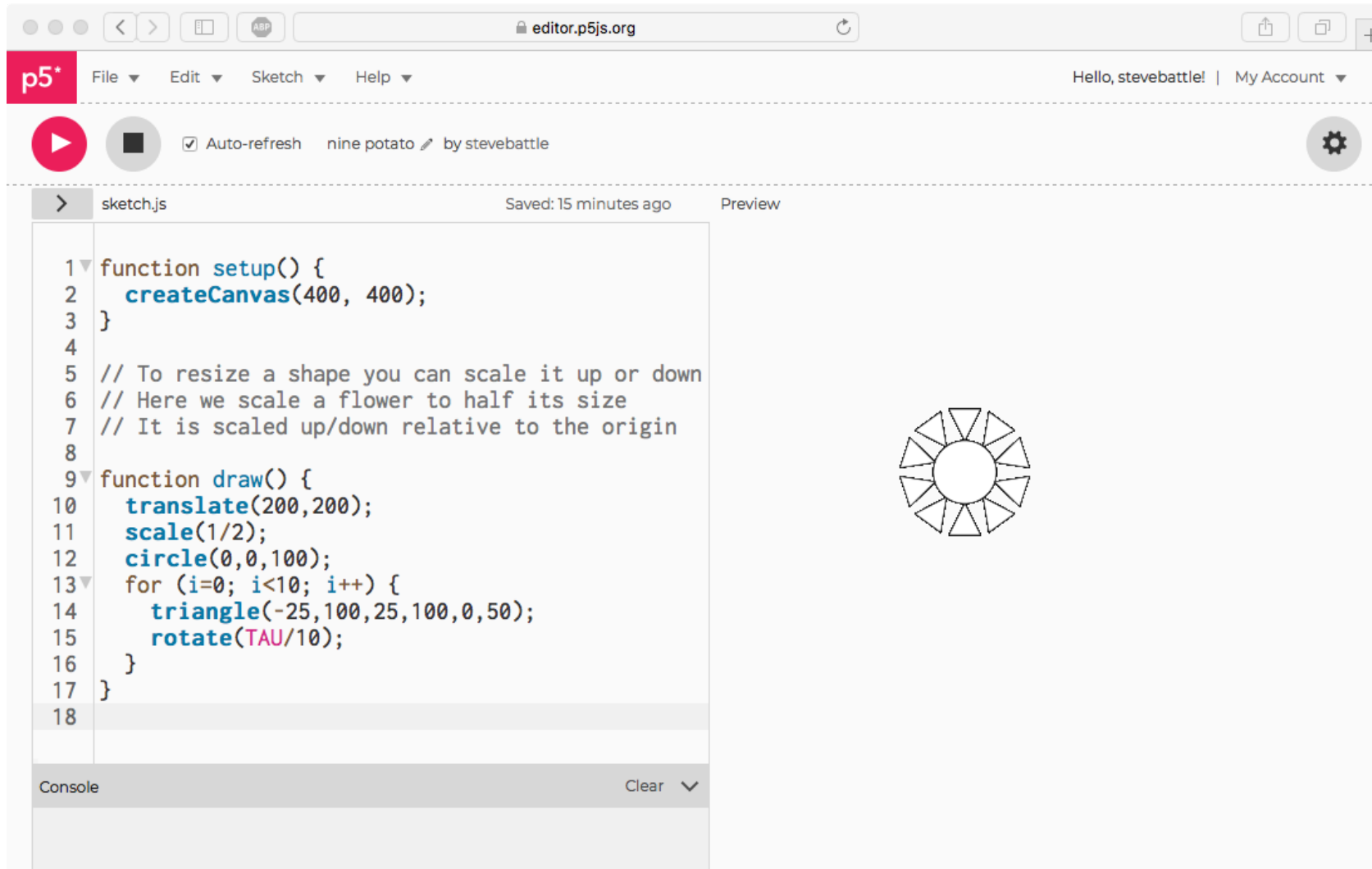# Going loopy


```
1 ▼ function setup() {
2     createCanvas(400, 400);
3 }
4
5 // Instead of repeating identical code
6 // you can make a (for) loop
7 // The loop variable, i, counts up from 0 to 7
8 // That's 8 squares in total, drawn in 45 degree steps
9
10 ▼ function draw() {
11     translate(200,200);
12     rectMode(CENTER);
13     circle(0,0,100);
14 ▼   for (i=0; i<8; i++) {
15         rect(0,100,50,50);
16         rotate(TAU/8);
17     }
18 }
19
```

Scale it back


```
1▾ function setup() {
2      createCanvas(400, 400);
3  }
4
5  // To resize a shape you can scale it up or down
6  // Here we scale a flower to half its size
7  // It is scaled up/down relative to the origin
8
9▾ function draw() {
10     translate(200,200);
11     scale(1/2);
12     circle(0,0,100);
13▾    for (i=0; i<10; i++) {
14         triangle(-25,100,25,100,0,50);
15         rotate(TAU/10);
16     }
17 }
18
```

Flower function


```javascript
function setup() {
  createCanvas(400, 400);
}

// push() and pop() keep the effects of translate(),
// scale(), and rotate() local.
// Make a flower 'stamp' function.

function flower(x,y) {
  push();
  translate(x,y);
  scale(0.5);
  fill('yellow');
  circle(0,0,100);
  fill('blue');
  rect(-5,50,10,150);
  for (i=0; i<10; i++) {
    triangle(-25,100,25,100,0,50);
    rotate(TAU/10);
  }
  pop();
}

function draw() {
  background('green');
  flower(100,150);
  flower(300,250);
}
```

Animated


```
var p = 10; // number of petals

function setup() {
  createCanvas(400, 400);
  frameRate(10); // frames per second
}

// frameCount counts frames displayed
// '%' is remainder after division (modulo)

function flower(x,y,a) {
  push();
  translate(x,y);
  scale(0.5);
  fill('yellow');
  circle(0,0,100);
  fill('blue');
  rect(-5,50,10,150);
  rotate(a);
  var n = frameCount % (2*p);
  for (i=0; i<p; i++) {
    if (n>i && i>=n-p) {
      triangle(-25,100,25,100,0,50);
    }
    rotate(TAU/p);
  }
  pop();
}

function draw() {
  background('green');
  flower(100,150,0);
  flower(300,250,PI);
}
```