

Learning to Code with SVG

Lesson Plan: Coding Rotating Ellipses in SVG on a 600 by 600 grid

Objective: Hands-on learning of SVG by drawing ellipses and rotating them.

Lab Time: Approximately 1/2 hour, not including Lecture time. Students should test each path command before adding each additional command.

Age range: 4-8th grades, or any age student unfamiliar with SVG

Requirements: Familiar with a simple text editor
Ability to save file with a .svg extension.
Understanding of concept of radius in both x and y directions.
Understanding of rotation of an object about a given point.
Familiar with the radius of ellipse & circle elements in SVG

Resources: <http://steamcoded.org/lessons/grid600x600.svg>
<http://steamcoded.org/images/ellipses.svg>
<http://steamcoded.org/images/ellipses.svg> with passed parameters
<http://www.w3schools.com/svg/default.asp>
<https://www.w3.org/TR/SVG11/>
Free eBook for iBooks: *STEAM Coded String Art*, by Jay Nick

Lecture: This lesson will draw 1 ellipse element and 2 circle elements on a 600x600 grid. The ellipse element will be reused and rotated every 5 degrees about it's center.

Demonstrate various images created by passing parameters to <http://steamcoded.org/images/ellipses.svg> Change the repeat value to rotate the ellipse every repeat degrees and change the x-radius by changing the radius value passed.

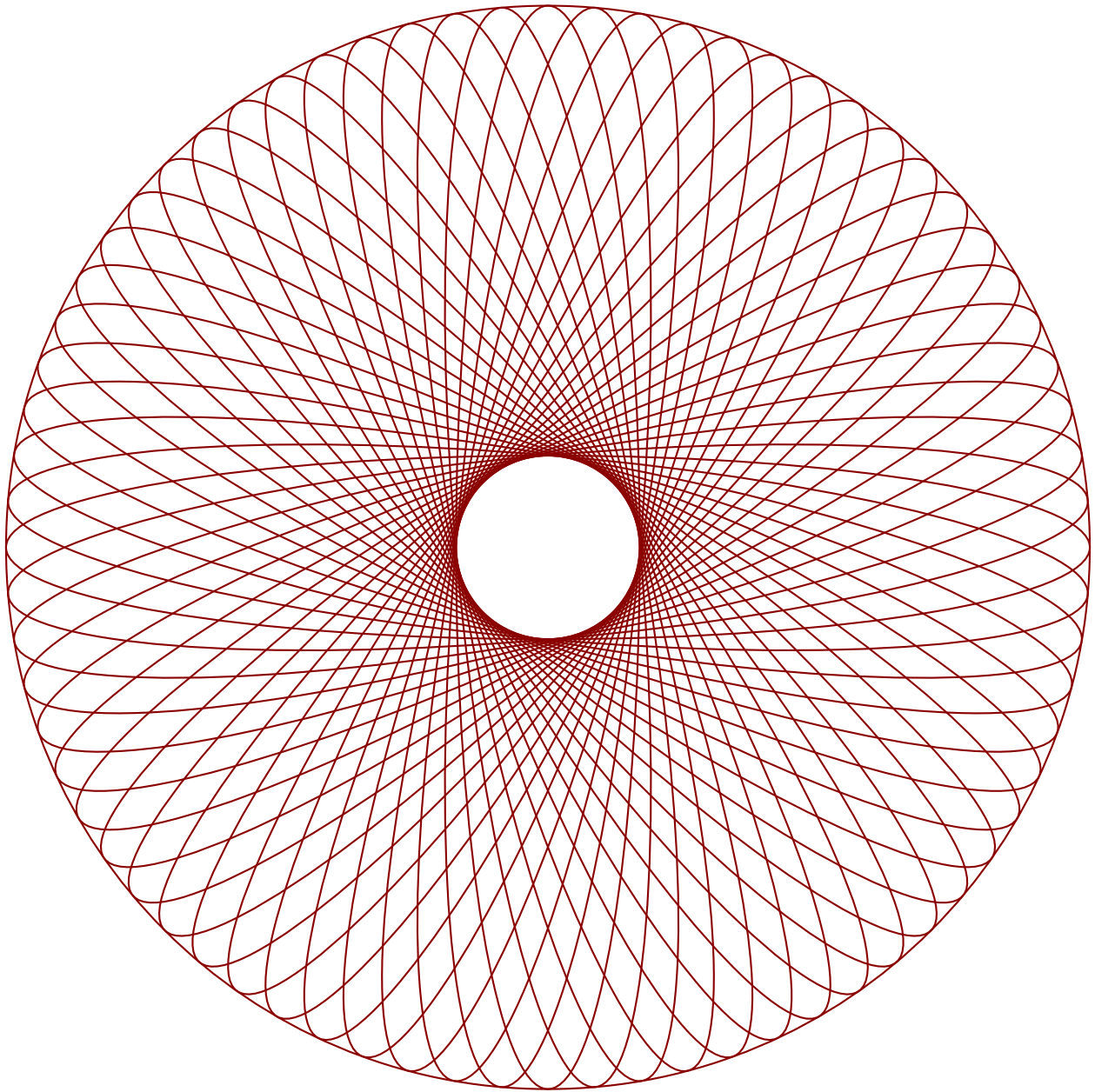
Procedure: Have students get an SVG template with 600x600 grid from:
<http://steamcoded.org/lessons/grid600x600.svg.txt>
Copy the code and paste it into a text editor.
Save the file as ellipses.svg then open in a browser. Keep the text editor and browser windows open.

Add SVG elements where indicated using the instructions on page 3 (see below). **Important:** Students should save the file and refresh the browser after adding a few SVG element to their file to make sure they don't have errors.

When complete, change the style attribute of the first <g> element from "display:initial" to "display:none" which hides the grid Then change the style attribute of the second <g> element from "opacity:0.5" to "opacity:1"

Take Away: Students should gain an understanding of the artistic effect obtained by rotating ellipses.

Additional Activity Students can pass parameters to <http://steamcoded.org/images/ellipses.svg> to find one they like, then try to duplicate it in code.



STEAMCODED.ORG

Coding Rotating Ellipses in SVG on a 600 by 600 grid

To get started copy the code of this image into your editor:

<http://steamcoded.org/lessons/grid600x600.svg.txt> and save the file as ellipses.svg and open the file in a browser.

- 1: Append the following attributes to the <g> element with style="opacity:0.5;"
Add styles: "fill:none;stroke:darkred;"
- 2: Create a <ellipse> element with id="e1" centered at (300,300)
with x-radius: 50 and y-radius: 295
- 3: Reuse the ellipse with id="e1" by adding a <use> element with attributes:
xlink:href="#e1"
transform="rotate(5,300,300)"
- 4: Repeat step 3 rotating every 5 degrees until reaching 175 because rotating 180 degrees will be a duplicate of the original ellipse.
- 5: Add a <circle> element centered at (300,300) with a radius of 295
- 6: Add a <circle> element centered at (300,300) with a radius of 50

When complete, change the style attribute of the first element from "display:initial" to "display:none" which hides the grid. Then change the style attribute of the second element from "opacity:0.5" to "opacity:1"

Coding Rotating Ellipses in SVG on a 600 by 600 grid

Answer Sheet

Common mistakes are missing double quote marks around attribute values, missing space between attributes, missing the start < and ending /> tags, using a semicolon where a colon is required and vice-versa, and not putting elements in the correct order.

- 1: `<g style="opacity:0.5;fill:none;stroke:darkred;">`
- 2: `<ellipse id="e1" cx="300" cy="300" rx="50" ry="295" />`
- 3: `<use xlink:href="#e1" transform="rotate(5,300,300)" />`
- 4: `<use xlink:href="#e1" transform="rotate(10,300,300)" />`
`<use xlink:href="#e1" transform="rotate(15,300,300)" />`
`<use xlink:href="#e1" transform="rotate(20,300,300)" />`
`<use xlink:href="#e1" transform="rotate(25,300,300)" />`
`<use xlink:href="#e1" transform="rotate(30,300,300)" />`
`<use xlink:href="#e1" transform="rotate(35,300,300)" />`
`<use xlink:href="#e1" transform="rotate(40,300,300)" />`
`<use xlink:href="#e1" transform="rotate(45,300,300)" />`
`<use xlink:href="#e1" transform="rotate(50,300,300)" />`
`<use xlink:href="#e1" transform="rotate(55,300,300)" />`
`<use xlink:href="#e1" transform="rotate(60,300,300)" />`
`<use xlink:href="#e1" transform="rotate(65,300,300)" />`
`<use xlink:href="#e1" transform="rotate(70,300,300)" />`
`<use xlink:href="#e1" transform="rotate(75,300,300)" />`
`<use xlink:href="#e1" transform="rotate(80,300,300)" />`
`<use xlink:href="#e1" transform="rotate(85,300,300)" />`
`<use xlink:href="#e1" transform="rotate(90,300,300)" />`
`<use xlink:href="#e1" transform="rotate(95,300,300)" />`
`<use xlink:href="#e1" transform="rotate(100,300,300)" />`
`<use xlink:href="#e1" transform="rotate(105,300,300)" />`
`<use xlink:href="#e1" transform="rotate(110,300,300)" />`
`<use xlink:href="#e1" transform="rotate(115,300,300)" />`
`<use xlink:href="#e1" transform="rotate(120,300,300)" />`
`<use xlink:href="#e1" transform="rotate(125,300,300)" />`
`<use xlink:href="#e1" transform="rotate(130,300,300)" />`
`<use xlink:href="#e1" transform="rotate(135,300,300)" />`
`<use xlink:href="#e1" transform="rotate(140,300,300)" />`
`<use xlink:href="#e1" transform="rotate(145,300,300)" />`
`<use xlink:href="#e1" transform="rotate(150,300,300)" />`
`<use xlink:href="#e1" transform="rotate(155,300,300)" />`
`<use xlink:href="#e1" transform="rotate(160,300,300)" />`
`<use xlink:href="#e1" transform="rotate(165,300,300)" />`
`<use xlink:href="#e1" transform="rotate(170,300,300)" />`
`<use xlink:href="#e1" transform="rotate(175,300,300)" />`
- 5: `<circle cx="300" cy="300" r="295" />`
- 6: `<circle cx="300" cy="300" r="50" />`