Shapes

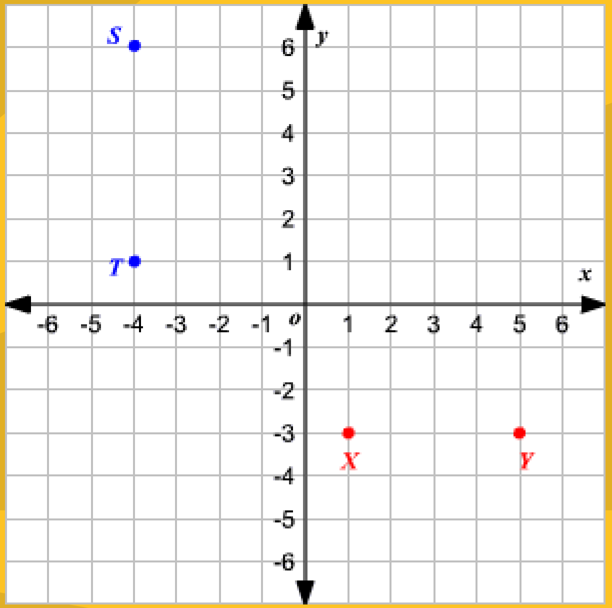
LO IWBAT Design background canvases on P5 editor in a variety of sizes.

Learning Objectives:

To:

- Grow in my understanding the basic structure of p5.js
- Design shapes on P5 editor in a variety of sizes.
- Input a variety of shapes to the background.
- Write lines of code (arguments) to reflect goals 2&3.

Quick Recap



What's needed to Create shapes?

Line of code/ Syntax

=

```
Function/Command( ) { Argument/parameters);
}
```

01

Vocab Words

What do we need to know to make input shapes?

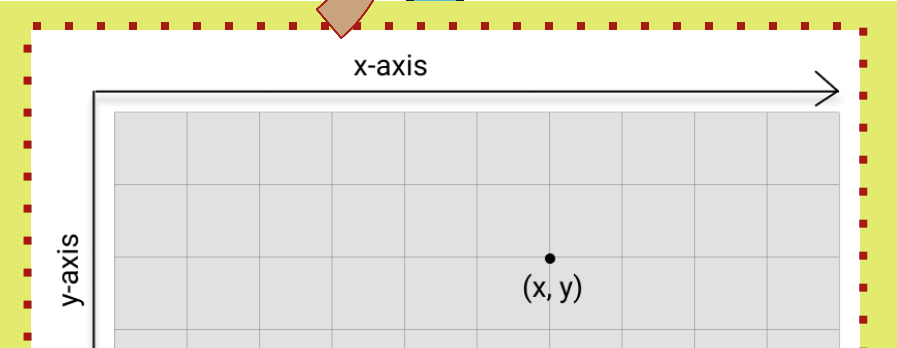
What's needed to Create shapes?

x-axis

Essentially, it is the horizontal line on a graph.

y-axis

Essentially, it is the vertical line on a graph.



Background Info:

X- & y- axis

Lines on a graph

What's the Function?

stroke() is the color of the point.
strokeWeight() size of the point. (thickness(

color/stroke()

Use grayscale values (v)

Use RGB values (v, v, v)

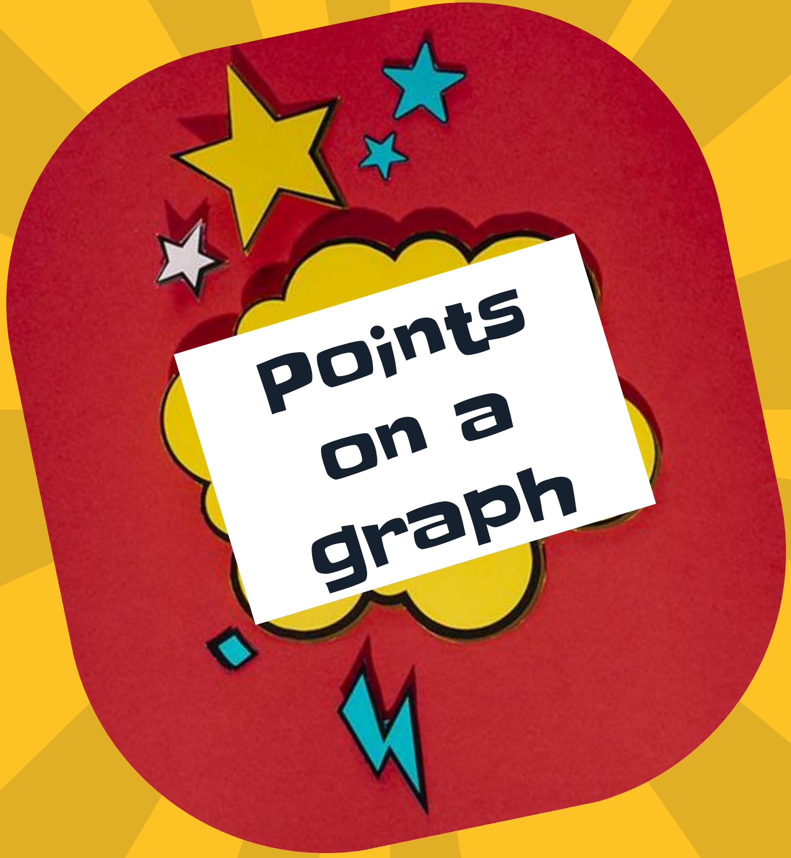
SVG & CSS colors ('color')

V= value

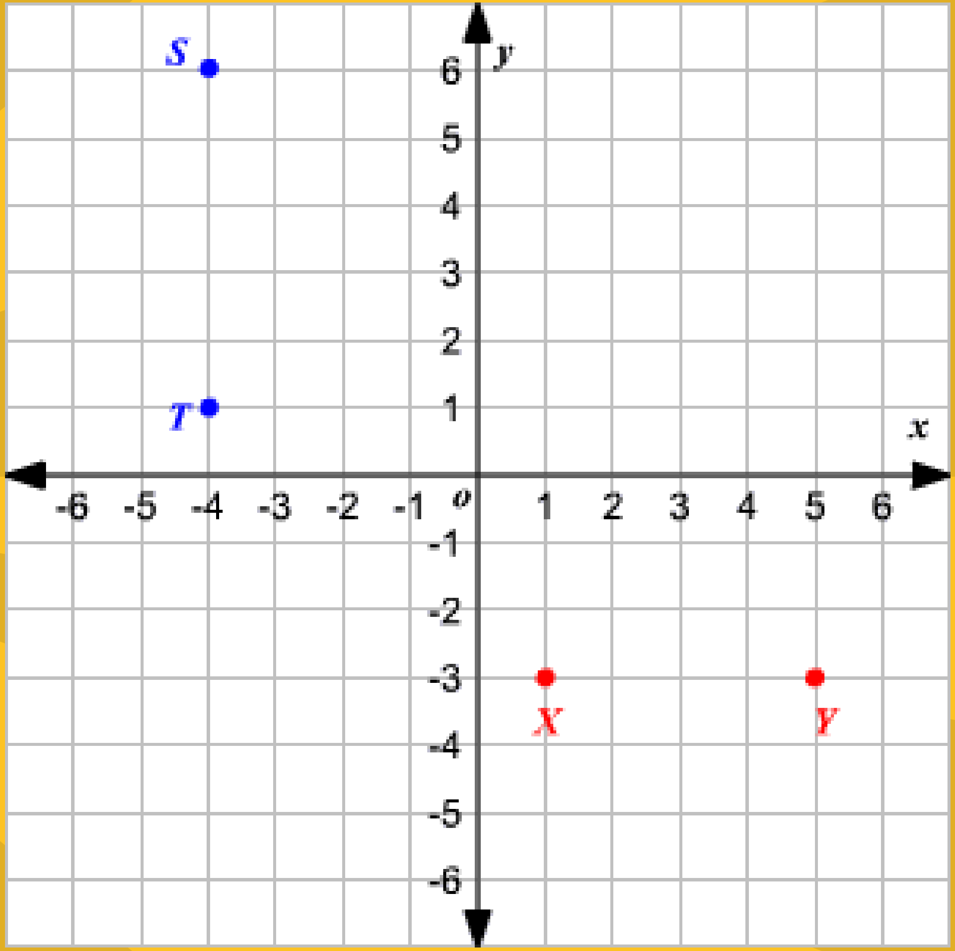
02

Variety of Shapes

What does math have to do with coding P5-JS?



Points on a graph

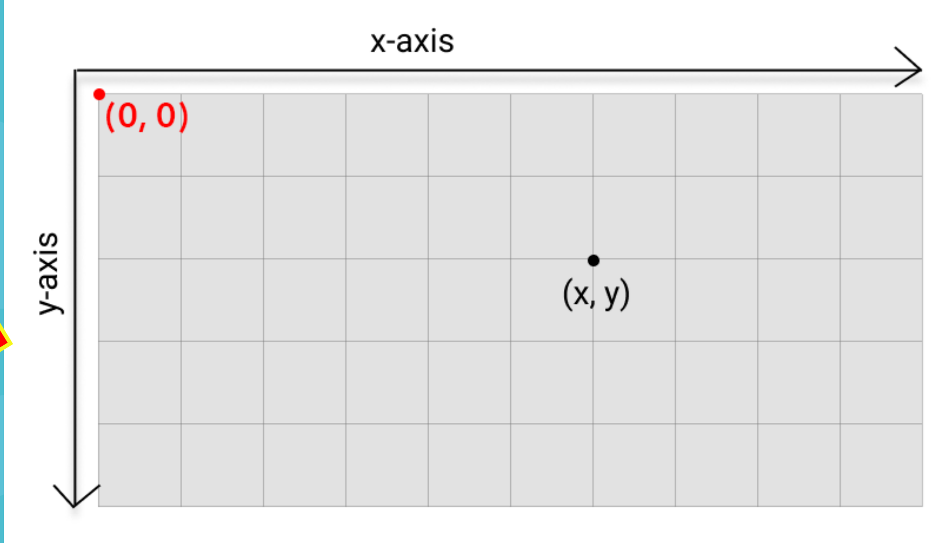


Point (v, v)

```
point(85, 75);
```

Syntax

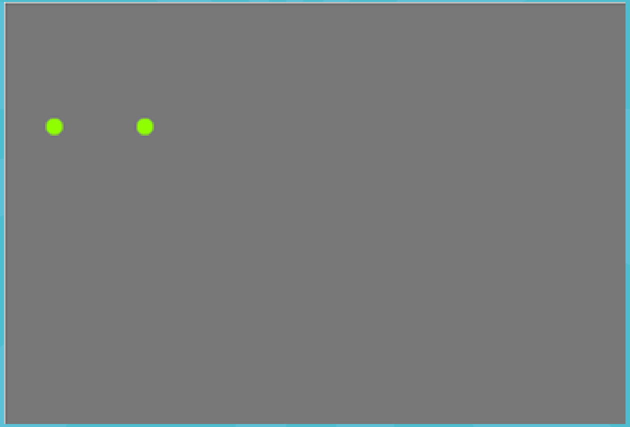
(x, y)



Point (v, v)

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

function draw() {
  background(120); optional
  point(85, 75);
  point(30, 75); optional
  stroke('greenyellow');
 strokeWeight(10);
}
```



Let's Explore points

With your partner:

Plot the following on a 400,400 canvas:

point(24, 58)

point(100, 200)

Point closest to the:

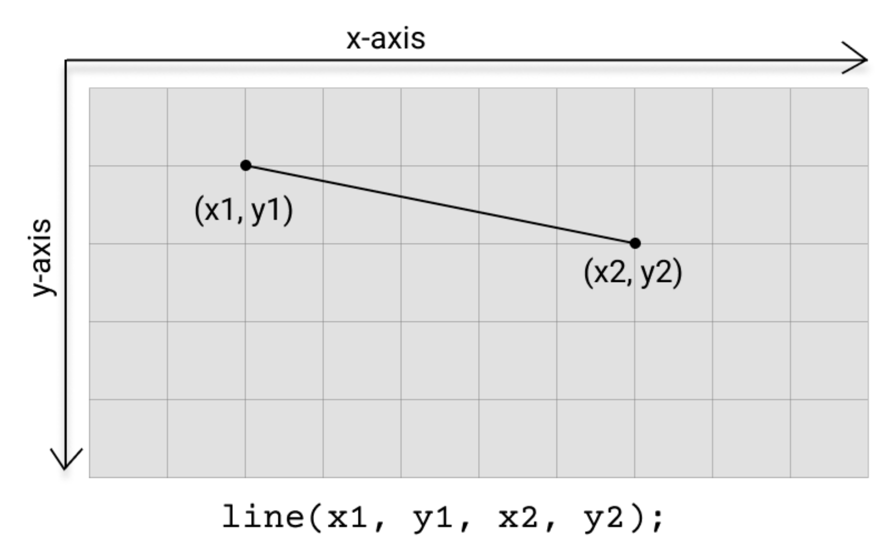
- Center
- bottom right
- Bottom left
- Top right
- Top left

line (v, v, v, v)

```
line(30, 20, 85, 20);
```

Syntax

```
line(x1, y1, x2, y2)
```

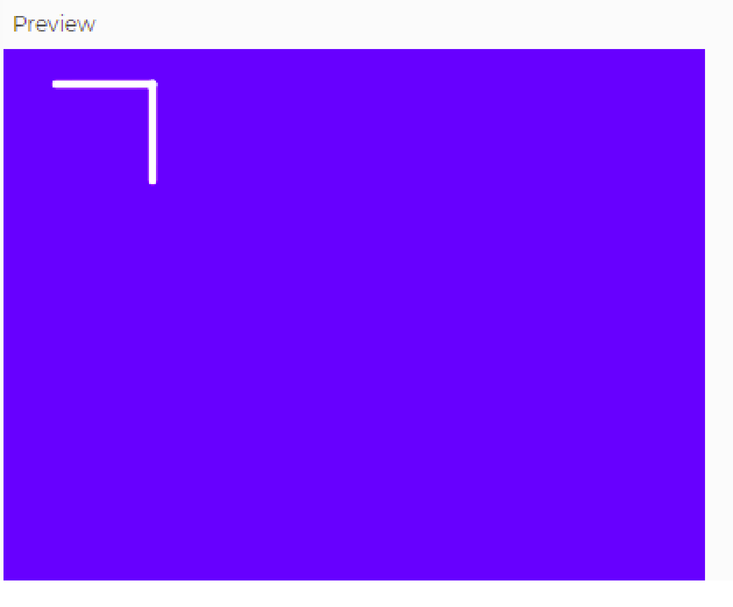


```
line(x1, y1, x2, y2);
```

line (v, v, v, v)

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

function draw() {
  background('blue');
  line(30, 20, 85, 20);
  stroke(255);
  line(85, 20, 85, 75);
  stroke(255);
 strokeWeight(5);
}
```



Let's Explore lines

With your partner:

Plot the following on a 400,400 canvas:

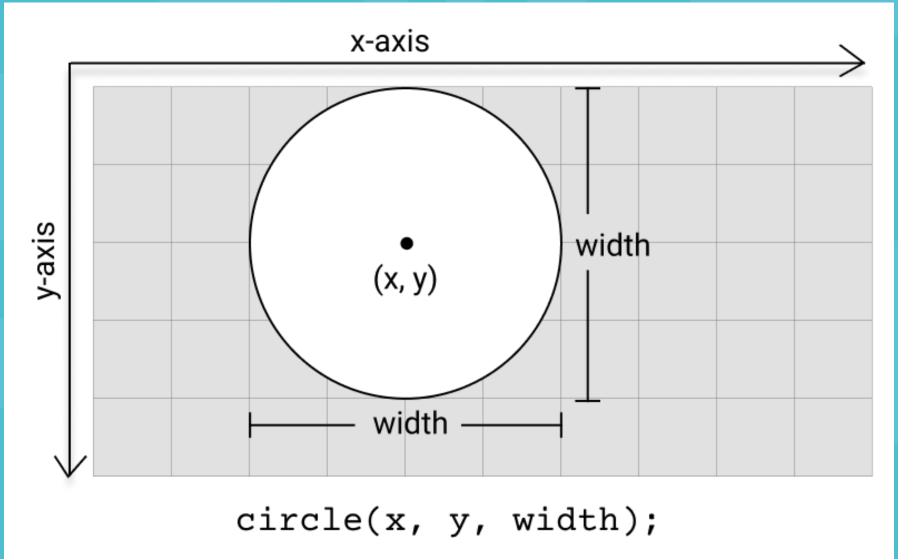
```
line(85, 75, 30, 75)
line(-30, -20, 85, 20)
line(30, 20, 85, 75)
```

Line closest to the:

- Center
- bottom right
- Bottom left
- Top right
- Top left
- Combine lines

circle (x, y, d)

```
circle(30, 30, 20);
```



Syntax

circle(x, y, d)

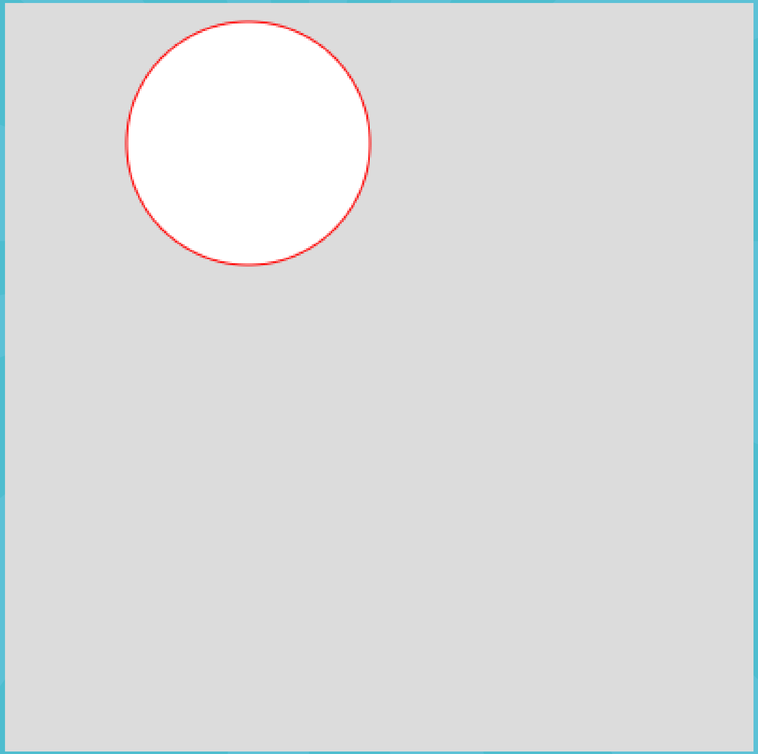
x-axis, y-axis, [location]

Diameter [width/height of circle]

circle (v, v, v)

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

function draw() {
  background(220);
  circle(130, 75, 130);
  stroke('red');
}
```



Let's Explore lines

With your partner:

Plot the following on a 400,400 canvas:

circle(30, 30, 20)

circle(130, 75, 130)

circle closest to the:

- Center
- bottom right
- Bottom left
- Top right
- Top left
- Combined circles

02

Tasks

Let's Try it OUt

Task 1: With your partner, Use slides 12-14 and the editor here to complete slide 21.

Task 2: With your partner, Use slides 15-17 and the editor here to complete slide 22.

Task 1: With your partner, Use slides 18-20 and the editor here to complete slide 22.

□ Spin the wheel to discover your task

Bonus: Use your knowledge of lines to create a rectangle. Use Use all the shapes to create a picture.

Task 1: Points

Task 1: With your partner, create points as specified in Slide 12.
Input values into the P5-editor link to verify.

		Value	Value
Shape	1	255	153
	2	#	#
	3	#	#
	4	#	#
	5	#	#
	6	#	#

Task 2: Lines

Task 2: With your partner, create lines as specified on Slide 15. Input values into the P5-editor link to verify.

		Value	Value	Value	Value
Shape	1	255	153		
	2	#	#	#	#
	3	#	#	#	#
	4	#	#	#	#
	5	#	#	#	#
	6	#	#	#	#

Task 3 Circles:

Task 3: With your partner, create circles as specified on Slide 15. Input values into the P5-editor link to verify.

		Value	Value	Value
Shape	1	255	153	204
	2	#	#	#
	3	#	#	#
	4	#	#	#
	5	#	#	#
	6			