Ms. Boland
Software Engineering

Drawing with p5

SKILLS

```
ellipse()
rect()
line()
point()
triangle()
quad()
arc()
beginShape()
fill()
noFill()
stroke()
noStroke()
strokeWeight()

Sequencing
algorithms
```

Essential Question: How can code be used as a creative and expressive medium?

Part 1: Drawing in p5

Task: Create your own unique drawing/image using the p5 editor.

Key Objectives:

- I can use shapes that we discovered in the p5 reference in order to create your own unique image/symbol
- I can add comments to the different shapes/functions in order to identify/explain the sequencing of your code
- I can use fill and stroke to change the colors of the shape in order to enhance/create depth in the detail of your image/symbol
- I can center your image/symbol in the middle of the canvas as a means of demonstrating your knowledge of the coordinate plane in p5

Part 2: Reflection Questions

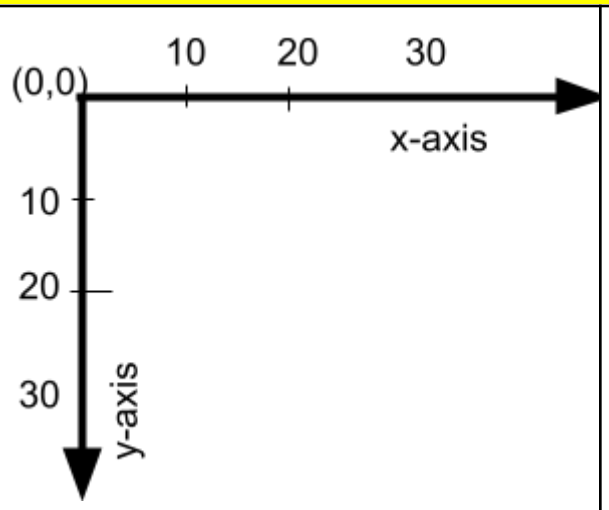
Directions: Create a google doc and answer the following questions using complete sentences.

1. How is coding in p5 similar/different from coding in pyret?
2. How can the p5 reference help with creating unique images/symbols?
3. Why is it important to pay attention to the order/sequencing of code in p5?
4. Describe the process you used to create your unique image/symbol.

p5 reference sheet | *Official documentation of the p5 library.*	**p5 web editor** | *Online IDE for p5 projects - it comes with the p5 library linked in the HTML and will give p5 specific errors.*

Ms. Boland
Software Engineering

p5.js Coordinate System && Major Callback Functions
This is the backbone of p5!



- Origin in the upper left corner
- X-axis runs left to right, increasing as you move right
- Y-axis runs up and down, increasing as you move down.

Size of canvas is determined by `createCanvas(x,y)` in setup function.

Using `windowWidth` and `windowHeight` in `createCanvas` makes the canvas fill the window.

System variables `width` and `height` hold any values set in createCanvas.

```
function setup(){
 //createCanvas here
 //anything else that should only happen once, or be true at start of program
}
```

MAJOR CALLBACK FUNCTION
Runs once, at the very start of the program. Anything you want to have happen or be true before the program starts should go here.

```
function draw(){
 //things you want to have appear on the canvas
//sometimes conditional logic to control decisions about your canvas
}
```

MAJOR CALLBACK FUNCTION
Runs in a repeated loop immediately after setup has run.

Generally displays what is on canvas, can sometimes also have conditional logic, loops, or whatever else controls the program or should run repeatedly.

There are lots of other callback functions you can use in p5 - look in the p5 reference sheet for an example of ones like preload(), mousePressed(), etc.

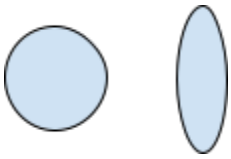
Ms. Boland
Software Engineering

Function Calls for Basic Shapes && Styling

Want to draw something else? Explore the p5 reference sheet!

```
//display coordinates of the mouse
at coordinate (20,20)
 text(mouseX + ", " + mouseY, 20, 20)
```

This is a helpful chunk of code that can help you find mouse coordinates when drawing shapes. Use it!



ELLIPSE/CIRCLE

```
//ellipse(x,y,width,[height])
//x and y are center of circle -
fourth height value is optional.
ellipse(50,70,100) //makes a circle
ellipse(50,70,100,75) //makes an
oval
```

COLORING SHAPES

```
//fill(r,g,b,[a])
fill(30,40,200)
rect(50,70,100,60)

//gets rid of the fill for
anything beneath it.
noFill()
```

- **`fill()`** changes the color of any shapes it is above.
- If each shape is a different color, each should have its own fill
- Accepts 3 values - red, green, blue - to mix colors.
- Optional 4th value controls opacity, or how see-through something is or is not.
- All values out of 255 by default, but we have other ways to control color!

COLORING STROKE

```
//stroke(r,g,b,[a])
stroke(30,40,200)
rect(50,70,100,60)

//gets rid of the stroke for
anything beneath it.
noStroke()
```

- **`stroke()`** changes the color of the **line around** any shapes it is above.
- Accepts 3 values - red, green, blue - to mix colors.
- Optional 4th value controls opacity, or how see-through something is or is not.
- All values out of 255 by default, but we have other ways to control color!

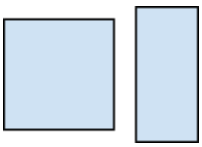
Ms. Boland
Software Engineering

<table>
<tr>
<td>CHANGING STROKEWEIGHT

```
//strokeWeight(x)
strokeWeight(10)
rect(50,70,100,60)
```

</td>
<td>

- <code>strokeWeight()</code> changes the color of the line thickness of any shapes it is above.
- Takes one number which represents thickness of the line in pixels
- If you want each shape to have a different strokeWeight, they each need the strokeWeight call above them.

</td>
</tr>
<tr>
<td> 
RECTANGLE</td>
<td>

```
//ellipse(x,y,width,height)
rect(50,70,100,60) //draws a
rectangle or square depending on
given values
```

</td>
</tr>
<tr>
<td> 
TRIANGLE</td>
<td>

```
//computer plays connect the dots
with points
//triangle(x1,y1,x2,y2,x3,y3)
triangle(30,40,100,120,200,230)
```

</td>
</tr>
<tr>
<td> 
LINE AND POINT</td>
<td>

```
//line(x1,y1,x2,y2)
//computer connects points in a
line
//lines have no fill - only a
stroke!
line(30,40,100,120)

//point(x,y)
//makes a point at the given
coordinate
//v small unless you increase
strokeWeight
point(30,40)
```

</td>
</tr>
<tr>
<td>Coding Train Videos for more Support</td>
<td>

- Drawing Shapes in p5.js
- Basics of Drawing in p5
- Color in p5.js
- Errors and Console

</td>
</tr>
</table>