p5.js Quick Reference

Shapes

p5.js provides many pre-written functions for drawing common shapes. We can control how and where the shape is drawn by providing different data values to the function parameters.

Points are drawn as single dots given as two parameters to the `point` function: X, then Y

```
// x = 4, y = 5
point(4, 5);
```

Lines are drawn between two coordinates given as four parameters to the `line` function: Start X, Start Y, End X, End Y

```
// Line between 0, 0 & 4, 5
line(0, 0, 4, 5);
```

Rectangles are drawn with their top left corner at a given coordinate using a given width and height.

```
// Rectangle starting at 1, 2
// with a width of 4 and height of 3
rect(1, 2, 4, 3);
```

Ellipses (ovals) are drawn with their *center* at a given coordinate using a given width and height. *Circles are ellipses with only a width (diameter).*

```
// Ellipse centered at 3, 3
// with a width of 4 and height of 6
ellipse(3, 3, 4, 6);

// Circle centered at 3, 3
// with a width (diameter) of 4
ellipse(3, 3, 4);
```

Triangles and **Quads** are drawn as connections between the given coordinates. (3 coordinates for triangles, 4 for quads.)

```
// Triangle with corners at:
// - 4, 2
// - 7, 8
// - 2, 7
triangle(4, 2, 7, 8, 2, 7);
```

```
// Quad with corners at:
// - 2, 1
// - 8, 2
// - 7, 7
// - 1, 4
quad(2, 1, 8, 2, 7, 7, 1, 4);
```

Colors

The simplest way to refer to colors when using p5.js is to use pre-defined colors names. For example: “Red”, “Yellow”, “Purple”, … To see a list of available color names, see https://www.w3schools.com/colors/colors_names.asp

To change the background color of the canvas, we use the command `background` with a color name:

```
background("Aqua");
```

In order to fill the various shapes in the scene with color, we use the command `fill` with a color name. **However**, once we set a fill color **ALL** shapes will become that color until we change the color again.

```
  // Sun
  fill("Yellow");
  ellipse(350, 20, 40, 40); // this is now Yellow inside.
```

Black outlines on each shape are distracting. To make them go away, we can add `noStroke();`. We can also change the outline color using `stroke`:

```
noStroke(); // Hide ALL outlines.
stroke("Purple"); // Make the outlines purple.
```

Variables & Expressions

A **variable** is a storage location in computer memory that contains a value.

- Variables can contain any type of value.
- They are called “variables” because the value that they store can change - vary - and be updated.

An **expression** lets us use common math **operators** (+, -, *, /) to tell the computer to perform calculations for us.

```
let canvasSize = 300;

// Make the sun size based on the canvas size AND start it down a bit from
// the top left corner.
let sunSize = canvasSize/3;
let sunX = 0;
let sunY = sunSize/4;
```

We can use both variables & expressions as parameters to functions!

```
// Make the top 3/4 of the canvas blue for the sky
// (so start at 0,0 and full canvas width but not the full height down)
fill("LightSkyBlue");
rect(0, 0, canvasSize, 3*canvasSize/4);
```

If we change a variable each time we draw, our picture changes too!

```
  ellipse(sunX, sunY, sunSize);
  sunX = sunX + 1; // ADD this line of code to slowly move the sun!
```

Global Variables Defined by p5.js

p5.js defines some variables for our use that store properties of the enviroment. A fun one is the current location of the mouse pointer!

- `mouseX` (number): x-coordinate of the mouse
- `mouseY` (number): y-coordinate of the mouse

```
  // Add a "bee" where the mouse is
  fill("Yellow");
  ellipse(mouseX, mouseY, 5);
```
