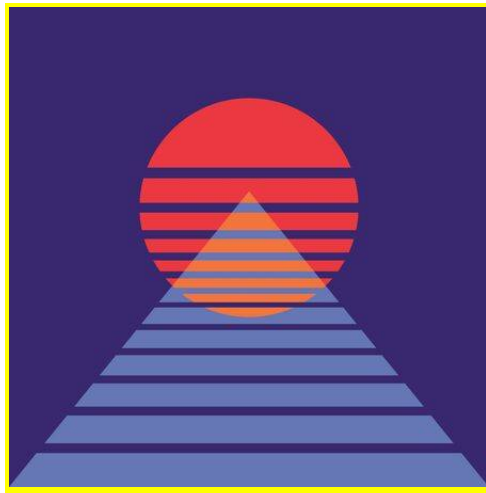


p5.js Tutorials

TUTORIAL 1:

Create an Animated Album Cover/Video



OBJECTIVE: (To learn the basics of the p5 editor and shapes)

Let's start at the beginning: how can we draw interesting shapes and colours on the computer screen?

First, [review this Interactive online tutorial by Allison Parrish](#), and follow along using the p5 editor at <http://p5js.org>. This will show you the very basics and get you oriented to using the p5 tool.

(OPTIONALLY, if you prefer learning via video tutorials rather than interactively online, review [Daniel Schiffman's introductory Video tutorials on p5.js](#) (Introduction 1.1 and 1.2 and Drawing 1.3 and 1.4. Both Parrish and Schiffman cover the same introductory material).

Afterward you will be ready to create some very simple, yet interesting, visual patterns on the computer using p5.

OK, after finishing Allison Parrish's tutorial (or watching and following along with the Schiffman videos), now it's your turn to code something that *you* like:

FOR YOUR FIRST TUTORIAL / ASSIGNMENT: Pick a piece of music that you really like. Listen to it a bit, then start a new p5 program in the p5 editor, and **then create an abstract design (cover art) that fits the piece**, using the simple shapes, colours, and other p5.js commands (noStroke, for example) you learned today, along with the **random()** function.

So ...

Open the [p5.js web editor](#)

- a. Create a new account (so that you can save your programs) OR you can just login with your google account.
- b. Set the p5.js canvas size to something like 1000 pixels wide, by 1000 pixels high. You will need the “createCanvas” function in the setup() function.
- c. Make the background colour something that you like - you’ll use the “background()” command in the setup() function.
- d. Create an abstract drawing by using the functions **ellipse**, **circle**, **triangle**, **quad**, **line**, **point** and **rectangle** on the computer screen. These can be of any size and position. Add at least 3 of these elements to the screen.
- e. Change the colours of one or more of the shape(s) you draw. You could make the ellipse red, for example. Use the fill() and stroke() commands.
- f. IMPORTANT: Finally, try using the **random()** function to define both the X and Y positions of

one or more of your shapes, rather than giving them a specific number/location.

We will use the “random()” function a lot. “Random” just generates a random number in a specific range that you set (e.g. “random(10)” gives you a random number between 0 and 9 inclusive).

Using random in p5 is really easy: just insert the word “random(and put a number to set an upper limit inside these brackets)” anywhere in a program where you use a number or variable.

So instead of:

`circle(125,175, 50);` (which draws a circle at x=125 and y=175)

You can do something like:

`circle(random(250), random(400), 50);`

You are basically saying to the computer: “you can decide the x and y position for this shape, within the range that I give you. Surprise me”. The great thing here is that this number will change each time the draw() loop runs, so the object appears to jump around (since the draw() loop runs over and over).

Random can also be used for size, colour, etc. - anything where we'd otherwise use a specific number, but instead want things to change in a surprising way.

We'll use it a lot in the future. Confused about random() (or just really keen!)? Watch this **OPTIONAL** Shiffman video [tutorial](#) 2.5 on the *random() function in p5.js.

DONE! :-)

Save the program (using a filename that includes your name like "BillyBoyd_Tutorial1") and show it to me.

GRADING: 3%

OPTIONAL Pro Tips: to make your animated (with random) images look even better, you can try:

1. adding the **noStroke()** command to the program (which gets rid of outlines for ellipses, rectangles, and other shapes). **Lines** and **points** become invisible though!

2. adding a fourth, low number like 50 to your **fill** (colour) definitions , like (255,255,255,50). It's called "alpha"; try it to see the "transparency" effect.

REALLY OPTIONAL, BUT INTERESTING ANIMATED GRAPHIC:

one quite cool effect is to generate random lines that all have a fixed point at the centre of the screen (so set the first X to $\frac{1}{2}$ of the canvas width and the first Y to $\frac{1}{2}$ of the canvas height. Make the second X and Y for the line anywhere on the screen - so x is random(your canvas width) and y is random (your canvas height). Try it and see. Also change the line thickness and colour using random. Wow!