

Generative Art with JavaScript (p5.js) - Instructor Notes

Quick Links & Requirements

Software

p5Js

Online: <https://editor.p5js.org/>

Zoom (Video Conferencing Software)

Online version: zoom.us

Desktop version: zoom.us/download

Content

Slides: [EN](#) / [FR](#)

Solution Sheet: [EN](#) / [FR](#)

Example Project: [EN](#) / [FR](#)

Workshop Overview

Description

Why learn JavaScript? This workshop will focus on teaching the fundamentals of computer programming in a visual context, and highlight how simple mathematical equations can create beautiful and engaging outcomes. (Don't worry, you don't need to be a math wizard!)

We'll use p5.js, which is a JavaScript library and is often used in electronic arts, new media art, and visual design. The application itself allows anyone to code in something called a Sketchbook, where programs can be saved and previewed instantly. Learners will use geometrical shapes and lines to create interactivity with their mouse and keyboard, and use colours to create visually intricate designs!

Learning Goal

In this workshop, learners will use Javascript and p5js to build generative art.

Learning Objectives

- Develop the ability to read and write Javascript code
- Access reference documentation to help problem solve
- Learn how variables are used to store data
- Draw shapes using x,y coordinates
- Perform operations using various data types (strings, integers, booleans)
- Use computers to make decisions using conditional statements

Key Coding Concepts

Concept	Definition	Analogy/Real-World Example
Functions	A named section of a program that performs a specific task; can be used over and over again	Think of a function as the chorus of a song. When you are writing a song, you only have to compose the chorus once because it is repetitive. After composing and labeling it as the chorus, you can write "Chorus" anywhere you want it to be repeated within the song, and people will know what you mean! This saves time.
Variables	Allow us to store a single piece of information	We can think of a variable as our piggy bank or wallet, which stores our money. The amount of money we have can change depending on our actions. Did we pay rent or go on a shopping spree? If we want to know how much money we have at any given time, we just need to look inside our piggy bank or wallet.
Conditional Statements	Allow computers to make decisions based on certain conditions being met; if/else statements are commonly used in conditional statements	We use conditional statements when we make decisions about our lives all the time! For example: If it is raining, then I will use my umbrella. If it's not raining, I won't!
Operators	Mathematical and logical expressions Relational operators are used for comparison (e.g. == (equals), != (does not equal), >= (greater than or equal to), etc.) Arithmetic operators are used for calculations (e.g. + (addition), - (subtraction), etc.)	Symbols dictate our actions all the time! When we see a stop sign, we stop. We know how to leave buildings by looking for the EXIT sign. We know what the '+' or '-' symbols mean when we are calculating the answer to a math problem, and our computers use these same (or very similar!) symbols.
Stretch Concepts <i>Consider covering if learners have prior experience with p5js.</i>		

Events	When one act triggers another to occur	When the clock strikes 12 PM, we eat lunch! At 3 PM, we leave school!
X,Y Coordinates	X,Y coordinates refers to the horizontal and vertical position of any pixel on a computer screen.	When you walk away from your friend, you take 3 steps to the right and one step forward. Your X,Y coordinates are (3,1) relative to your friend.

Using P5JS in JavaScript

p5js is a JavaScript library built for artists and graphic designers to easily be able to create generative art. It is derived from a language called processing. This library allows for the creation of objects (shapes, colors etc.) using Javascript in the browser.

Below is a table outlining the differences between processing, processing.js and p5js.

Processing	Processing.js	p5.js
A raw programming language used to create generative art.	Takes processing code and converts (interprets) it to JavaScript on the fly	P5 is a direct Javascript derivative of processing
	This library does require you to know processing	This library does not require you to know processing

P5js Reference - <https://p5js.org/reference/>

P5js Example Projects - <https://showcase.p5js.org/>

P5js Video Tutorial - <http://youtube.com/user/shiffman>

Prep Work

Before the workshop...

- ❑ Review instructor resources (notes, slides, solution sheet, example projects)
 - ❑ Review learning objectives and ensure you understand when and how they are accomplished within the content
 - ❑ Review coding concepts and ensure you are confident describing them to your group
 - ❑ Code and build (if applicable) the workshop's main project, ensuring you are comfortable with all steps outlined
- ❑ Decide which content the primary instructor and co-instructor are responsible for covering in the workshop

- ❑ Prepare for the [land acknowledgment](#)
- ❑ Ensure landing pages to links, videos, and/or examples are in the correct language
- ❑ OPTIONAL: Add any necessary slides for icebreaker instructions or audience-specific examples
- ❑ OPTIONAL: Print PDF of slides with instructor notes
- ❑ OPTIONAL: Conduct additional research on any concepts or tools

Additional Materials (Optional)

- ❑ Printed Zoom icons

Workshop Schedule

Introduction and Setup (30 mins)

Code-Along and Work Session (80 mins)

Wrap-up (10 mins)

Modifications & Extensions

How can the instructor or mentors support learners that are behind?

- Ask the producer to move learners who are having difficulty to a separate breakout room with a mentor

How can the instructor or mentors support learners that are ahead?

- Ask learners to complete the added challenges provided in the breakout rooms
- Provide free time for learners to remix (personalize!) their projects
- Have learners demo their projects (*Note: If all parties are comfortable, ask the learner to share their project link with the instructor (in the main room) or mentor (in the breakout room), and the instructor/mentor will share their screen with the group!*)
- Ask the producer to move learners who still have questions or who might want to talk to mentors about coding to a breakout room at the end of the workshop

Slide-Specific Notes

Today's Project (slide 13)

Demonstrate how to use the project and give learners time to try it too if possible.

Note: if the arrow keys no longer work for navigating the slides, click on an area of the slide outside of the p5js project

Generative Art (slide 18)

"We've probably all heard of painting and sculpting as forms of art, but generative art is art made with code! The computer will use our set of instructions to create that art, so we may need to do some simple math to have the artwork change over time. Since we're doing something creative, we'll find that experimenting with the code will often lead to some great results."

LTrees by Kate Compton (slide 19)

Play the video

"Kate's piece, LTrees, generates plants with code. The computer uses the same set of instructions each time, but with some simple math, is able to create wildly different results each time."

Wobbly Swarm by Konstantin Makhmutov (slide 20)

You will need to click the Refresh button (either the one built into the page or your browser refresh) in order to load the project. Once loaded, click to generate circles that interact with each other.

"Konstantin has used some slightly more complicated math to have the circles interact with each other, creating the wobbly swarm."

Quilts & Textiles by Libs Elliott (slide 21)

"Toronto-based artist, Libs Elliott, writes code that generates shapes on her computer screen. When she sees one that she's happy with, she turns the design into a hand-sewn quilt."

p5js (slide 23)

"Created in 1995, Processing was one of the first computer languages specifically intended to create art. It's open-source, which has led to several different variations of Processing over the years such as p5js. P5js is analogous to a tool box. It provides JavaScript with some additional information to allow us to create art. It's strength as a javascript flavour comes from how easily we can create exact shapes, that can then be combined to create pictures and even animations. It's also often used to represent data in a visual way - imagine each of the sun's rays in this animation represents the temperature each day over the course of a year."

Note: The rays of the sun do not actually represent any sort of real data ;)

How does p5js make art? (slide 24)

The words in bold are terms learners should become familiar with, so be sure to call out each of those.

As an illustrative example for the last point on animations, flipbook (and all traditional) animations work very similarly to p5js. It's just a series of pictures that we look at really quickly, and as long as parts of those pictures change our brains see the animation.

Keyboard Buttons Can Be Pressed Too! (slide 48)

Encourage learners to use what they've already learned to solve this problem themselves. The slide has 3 hints on it, each revealed by pressing a key, if learners need some help. They can also continue onto the next slide if they're stuck and want to see the suggested solution.

Almost Done! (slide 54)

If learners have been following the example code in the slide exactly, their project won't be quite the same as the example. The two main differences are the random function was only used as the blue value of the fill, and the width of the ellipse. By adding more random functions for both the red and green values of the fill, and the height of the ellipse, their project should be much more similar to the example.

Under "Other things to try" is a list of relatively simple challenges for learners who want to explore and get creative with their sketches. Example code for each can be found in the Solution Sheet as Add-Ons

Random Sizes and Colors (slide 52)

Note: Javascript's random function is exclusive to the last digit. ie) random(220,255) would result in the number 255 never being randomly selected. random(220, 256) will solve this issue.

Documentation:

https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Math/random

Stretch Goal: Let's Create Our Own Variables (slide 55)

Although we used a few variables in our project, this stretch goal walks through creating a variable to make the shapes grow. The example code can be found in the Solution Sheet as an Add-On.