

Digital Art with p5.js

Middle School / High School - Unit (20 Contact Hours)

Course Overview and Goals

The Digital Art With p5.js course explores the intersection of art and computer science. Using p5.js, a free and open-source JavaScript library for creative coding, students create digital drawings, animations, and interactive sketches while also learning how to code.

Learning Environment

The course utilizes a blended classroom approach. The content is fully web-based, with students writing and running code in the browser. Teachers utilize tools and resources provided by CodeHS to leverage time in the classroom and give focused 1-on-1 attention to students. Each unit of the course is broken down into lessons. Lessons consist of video tutorials, example programs to explore, and written programming exercises.

Programming Environment

Students write and run p5.js sketches in the browser using the CodeHS editor.

More Information

Browse the content of this course at <https://codehs.com/course/17244>.

Prerequisites

There are no prerequisites for the Digital Art With p5.js course. This course is meant to be a first-time introduction to computer science and does not require students to come in with any computer programming experience. Students should be familiar with basic geometry and the Cartesian (x, y) coordinate system to represent points in a plane.

Course Breakdown

Unit 1: Getting Started With p5.js (1 week)

Students explore the intersection of art and coding. Students learn the fundamental concepts of p5.js and create their first sketch using p5.js shape-drawing and coloring functions.

Objectives / Topics Covered	<ul style="list-style-type: none">• What is Creative Coding• Drawing and Shapes• Color• Variables
-----------------------------	--

<p>Example Assignments / Labs</p>	<ul style="list-style-type: none"> ● Rectangle <ul style="list-style-type: none"> ○ Students practice using <code>rect()</code> and <code>square()</code> by recreating the following sketch that was inspired by <i>Composition in Gray</i> by Piet Mondrian (1919). ● Fill <ul style="list-style-type: none"> ○ Students practice using the <code>fill()</code> function by adding color to this sketch inspired by <i>Portrait of Woman</i> by Pablo Picasso (1936) using the provided color palette. ● Making Variables <ul style="list-style-type: none"> ○ Students practice using variables by defining variables to store RGB values and set the background color of the canvas.
---------------------------------------	---

Unit 2: Animation (1 - 2 weeks)

Students learn techniques to turn their static sketches into animations. Students learn how to change the value of variables to animate the position, color, and transformation of shapes.

<p>Objectives / Topics Covered</p>	<ul style="list-style-type: none"> ● The Draw Loop ● Color Transitions ● Shape Transformations ● Direction
<p>Example Assignments / Labs</p>	<ul style="list-style-type: none"> ● Frame Rate <ul style="list-style-type: none"> ○ Students practice using the <code>frameRate()</code> function by adjusting the speed of the load bar animation. ● Frame Count <ul style="list-style-type: none"> ○ Students practice using the <code>frameCount</code> variable by recreating this animation of street lines being painted. ● Grayscale to Color <ul style="list-style-type: none"> ○ Students recreate an animation where the color of the shapes transitions from grayscale to color. ● Animated Translation <ul style="list-style-type: none"> ○ Students practice using <code>translate()</code> by recreating a sketch of randomly colored translated squares. ● Animated Rotation <ul style="list-style-type: none"> ○ Students practice using the <code>rotate()</code> function by creating an animation of a clock. ● Diagonal <ul style="list-style-type: none"> ○ Students create an animation of a circle moving in the downright diagonal direction.

Unit 3: Interactivity (1 week)

Students learn about interactivity and how to create sketches that react to user input. Using system variables provided by p5.js, students add conditional logic to their code that modifies the composition when mouse and keyboard input is detected.

Objectives / Topics Covered	<ul style="list-style-type: none">● What is Interactivity● Mouse Data● Keyboard Data
Example Assignments / Labs	<ul style="list-style-type: none">● mouseY<ul style="list-style-type: none">○ Students practice using <code>mouseY</code> by turning a still sketch of a flag pole into an interactive sketch where the flag moves based on the position of the mouse.● Paint and Erase<ul style="list-style-type: none">○ Students practice using <code>mouseX</code>, <code>mouseY</code>, and <code>mouseButton</code> by creating a basic paint tool.● keyIsDown<ul style="list-style-type: none">○ Students practice using the <code>keyIsDown()</code> function by recreating a sketch where the position of a cloud moves up, down, left, or right based on which arrow key on your keyboard is pressed.