

Visualize Programming and Create with JavaScript in p5.js

p5.js is a JavaScript library that makes it easy to create interactive and generative art and simulations. It's based on the Processing programming language, which is popular among artists and designers. p5.js provides a simple and intuitive API that makes it easy to get started with creative coding, even if you don't have any programming experience.



Generative Design: Visualize, Program, and Create with JavaScript in p5.js by Jerry Yarnell

★★★★☆ 4.5 out of 5

Language : English
File size : 89929 KB
Text-to-Speech : Enabled
Enhanced typesetting : Enabled
Print length : 502 pages
Lending : Enabled
Screen Reader : Supported



In this article, we'll dive into the basics of p5.js and build a fun interactive visualization. We'll cover the following topics:

- Setting up your p5.js project
- Creating basic shapes and animations
- Responding to user input
- Loading and manipulating images

- Creating interactive simulations

Setting up your p5.js project

To get started with p5.js, you can either download the library from the official website or use an online editor like CodePen or Glitch. Once you have p5.js set up, you can create a new sketch by creating a new HTML file and including the p5.js library.

```
<!DOCTYPE html> <html> <head> <title>My p5.js Sketch</title> <script src="p5.js"></script> </head> <body> </body> </html>
```

Once you have created a new sketch, you can start coding! The p5.js API is very well-documented, so you can easily find the functions that you need. You can also find many helpful tutorials and examples online.

Creating basic shapes and animations

One of the most basic things you can do with p5.js is create shapes and animations. To create a shape, you can use the following functions:

- `rect(x, y, width, height)`
- `circle(x, y, radius)`
- `ellipse(x, y, width, height)`
- `line(x1, y1, x2, y2)`
- `triangle(x1, y1, x2, y2, x3, y3)`

You can also use the following functions to animate your shapes:

- `translate(x, y)`

- `rotate(angle)`
- `scale(x, y)`

Here is an example of how to create a simple animation:

```
javascript function setup(){createCanvas(400, 400); }
```

```
function draw(){background(255); translate(width / 2, height / 2);  
rotate(frameCount / 100); rect(-50, -50, 100, 100); }
```

Responding to user input

p5.js makes it easy to respond to user input. You can use the following functions to detect mouse events:

- `mousePressed()`
- `mouseReleased()`
- `mouseMoved()`
- `mouseDragged()`

You can also use the following functions to detect keyboard events:

- `keyPressed()`
- `keyReleased()`
- `keyTyped()`

Here is an example of how to respond to mouse events:

javascript



Generative Design: Visualize, Program, and Create with JavaScript in p5.js by Jerry Yarnell

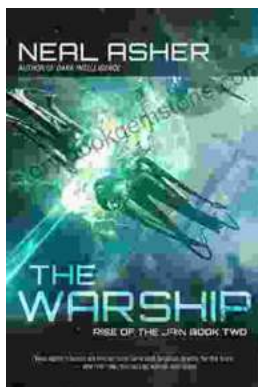
★★★★☆ 4.5 out of 5

Language : English
File size : 89929 KB
Text-to-Speech : Enabled
Enhanced typesetting : Enabled
Print length : 502 pages
Lending : Enabled
Screen Reader : Supported



The View From My Ordinary Resilient Disabled Body

In a world where normalcy is often defined by narrow and exclusionary standards, I stand as a testament to the boundless diversity and resilience of the...



The Rise of the Jain Two: A Monument to Naval Supremacy

In the vast expanse of the world's oceans, where the ebb and flow of tides dictate the rhythm of nations, a new era of maritime dominance is on...

