

p5.js

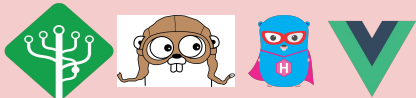
Eloisa Potrich

Estudante de Engenharia de Software

Maringaense mais gringa!



Comunidades



/eloisa.potrich



eloisapotrich1



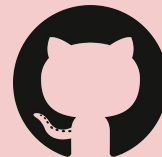
/elloo_p



/eloisa-potrich



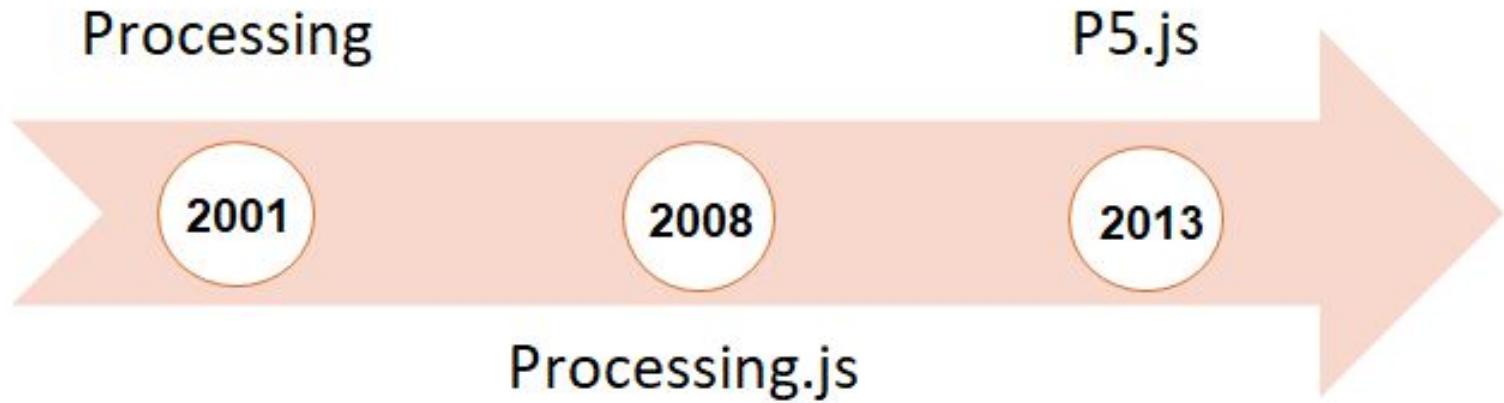
/ello_po



/EloPo

Enfim...

Linha do tempo







Jesse Reas
Artista Americano

Benjamin Fry
Especialista em Visualização de dados

mit
media
lab

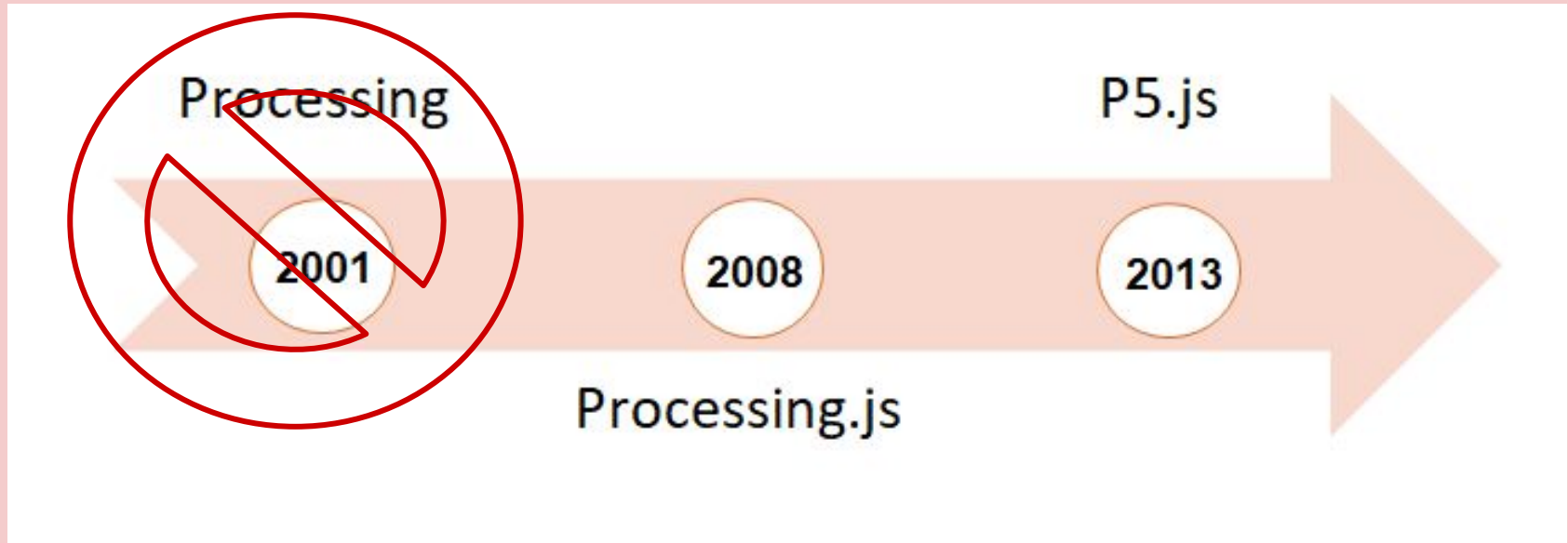


Processing

A linguagem para aprender a codificar dentro do contexto das artes visuais, sendo Open Source para que diversos artistas contribuam.

Tem por base as capacidades gráficas da linguagem de programação Java, simplificando características e criar alguns novos.

Linha do tempo





John Resig
Arquiteto Frontend



Processing.js

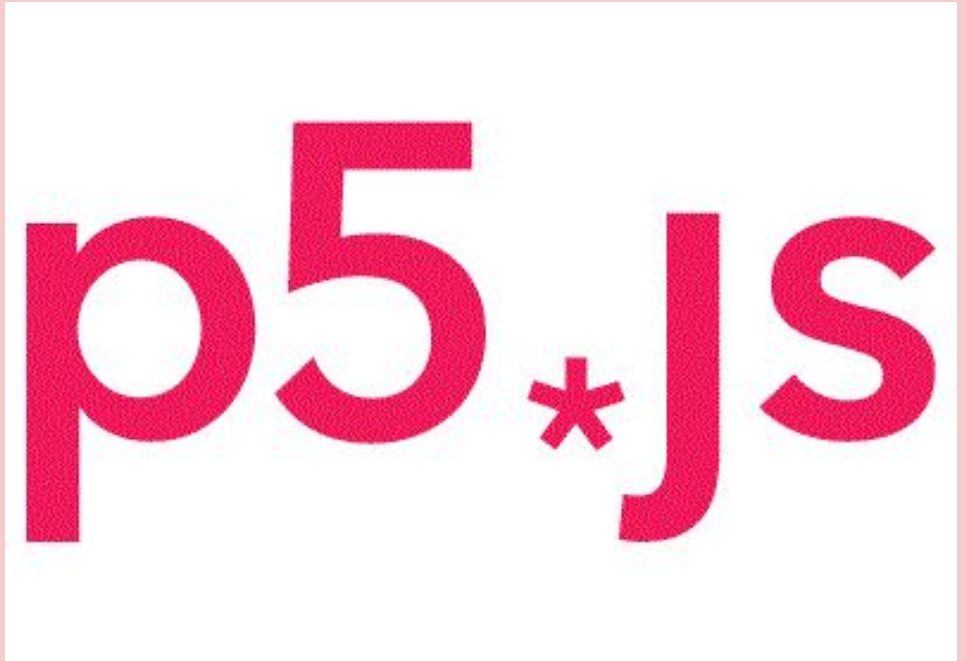
É uma porta JavaScript do Processing , uma linguagem de programação projetada para escrever visualizações, imagens e conteúdo interativo.

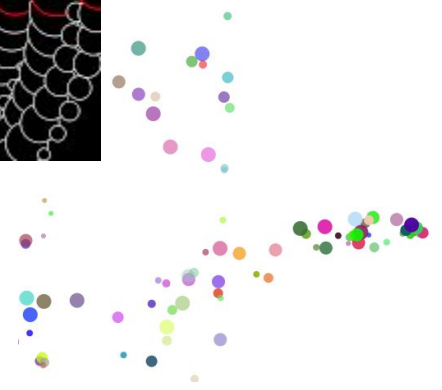
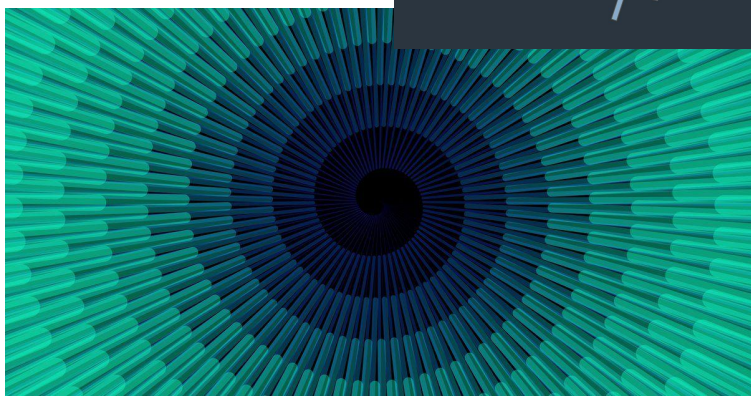
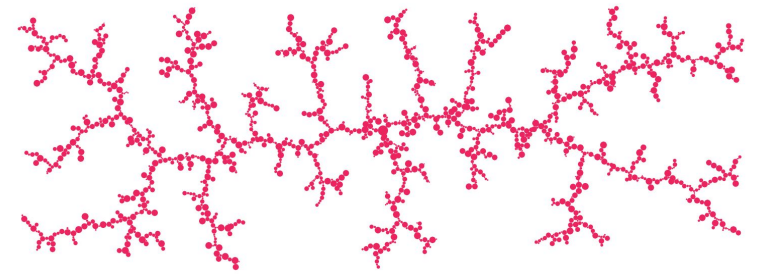
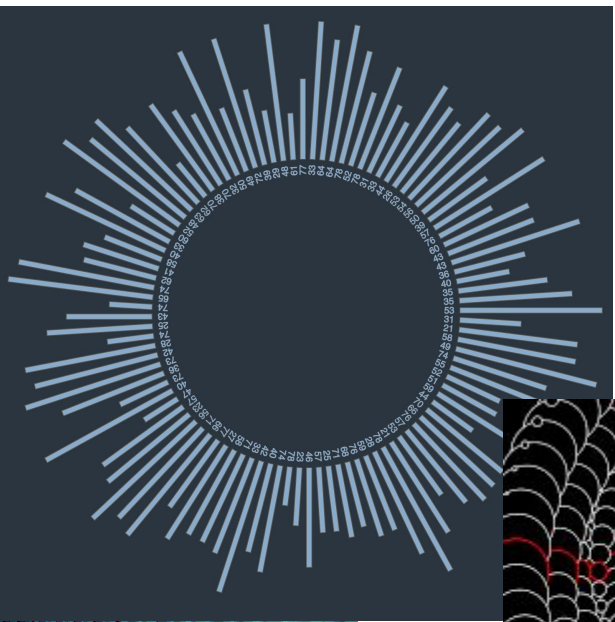
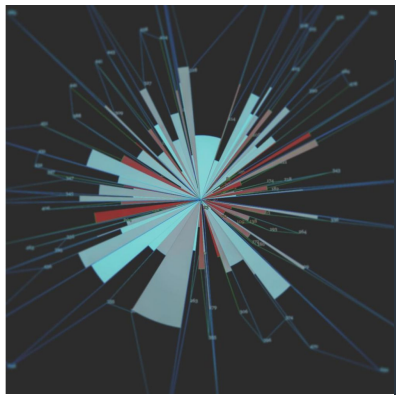
Linha do tempo





Lauren McCarthy
Artista







Snake



Download da IDE

 Auto-refresh

Superficial island



sketch.js

Preview

```
1 function setup() {  
2   createCanvas(400, 400);  
3 }  
4  
5 function draw() {  
6   background(220);  
7 }
```

Console

Clear ▾

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

p5*



Untitled



sketch.js x +

```
1 function setup() {  
2  
3 }  
4  
5 function draw() {  
6  
7 }
```

> console

Blz! E se eu quiser fazer em um editor de texto?

```

<!DOCTYPE html>
<html lang="en">
<head>

  <meta charset="UTF-8">
  <title>Snake</title>
  <script src="libraries/p5.js" type="text/javascript"></script>
  <script src="libraries/p5.dom.js" type="text/javascript"></script>
  <script src="libraries/p5.sound.js" type="text/javascript"></script>

  <script src="sketch.js" type="text/javascript"></script>
  <script src="snake.js" type="text/javascript"></script>

</head>
<body>
</body>
</html>

```

```

<!DOCTYPE html>
<html lang="en">
<head>

  <meta charset="UTF-8">
  <title>Snake</title>
  <script src="libraries/p5.js" type="text/javascript"></script>
  <script src="libraries/p5.dom.js" type="text/javascript"></script>
  <script src="libraries/p5.sound.js" type="text/javascript"></script>

  <script src="sketch.js" type="text/javascript"></script>
  <script src="snake.js" type="text/javascript"></script>

</head>
<body>
</body>
</html>

```







```
function setup() {  
  
}  
  
function draw() {  
  
}
```



```
function setup() {  
  createCanvas(720, 400);  
}  
  
function draw(){  
}
```


Untitled




http://localhost:3000/





```
function checkForFruit() {
  point(xFruit, yFruit);
  if (xCor[xCor.length - 1] === xFruit && yCor[yCor.length - 1] === yFruit) {
    var prevScore = parseInt(scoreElem.html().substring(8));
    scoreElem.html('Score = ' + (prevScore + 1));
    xCor.unshift(xCor[0]);
    yCor.unshift(yCor[0]);
    numSegments++;
    updateFruitCoordinates();
  }
}
```

```
function keyPressed() {
  switch (keyCode) {
    case 74:
      if (direction !== 'right') {
        direction = 'left';
      }
      break;
    case 76:
      if (direction !== 'left') {
        direction = 'right';
      }
      break;
    case 73:
      if (direction !== 'down') {
        direction = 'up';
      }
      break;
    case 75:
      if (direction !== 'up') {
        direction = 'down';
      }
      break;
  }
}
```



```
function checkSnakeCollision() {  
  var snakeHeadX = xCor[xCor.length - 1];  
  var snakeHeadY = yCor[yCor.length - 1];  
  for (var i = 0; i < xCor.length - 1; i++) {  
    if (xCor[i] === snakeHeadX && yCor[i] === snakeHeadY) {  
      return true;  
    }  
  }  
}
```



```
function checkGameStatus() {  
  if (xCor[xCor.length - 1] > width ||  
      xCor[xCor.length - 1] < 0 ||  
      yCor[yCor.length - 1] > height ||  
      yCor[yCor.length - 1] < 0 ||  
      checkSnakeCollision()) {  
    noLoop();  
    var scoreVal = parseInt(scoreElem.html().substring(8));  
    scoreElem.html('Game ended! Your score was : ' + scoreVal);  
  }  
}
```

Referencias

<https://p5js.org/>

Comunidade



Libraries



p5.dom lets you interact with HTML5 objects beyond the canvas, including video, audio, webcam, input, and text.



p5.sound extends p5 with Web Audio functionality including audio input, playback, analysis and synthesis.



p5.accessibility makes the p5 canvas more accessible to people who are blind and visually impaired

