Table of Contents

Prefacio

p5.j está inspirado y guiado por otro proyecto, que empezó hace 15 años. En el año 2001, Casey Reas y Ben Fry empezaron a trabajar en una nueva plataforma para hacer más fácil la programación de gráficas interactiva, la nombraron Processing. Ellos estaban frustrados por lo difícil que era escribir este tipo de software con los lenguajes que normalmente usaban (C++ y Java), y fueron inspirados por lo simple que era escribir programas interesantes con los lenguajes que usaban cuando niños (Logo y BASIC). Su mayor influencia fue Design by Numbers (DBN), un lenguaje que ellos estaban trabajando en mantenimiento y enseñando en ese tiempo (y que fue creado por su tutor de investigación, John Maeda). Con Processing, Ben y Casey estaban buscando una mejor manera de probar sus ideas en código, en vez de solo conversarlas o pasar demasiado tiempo programándolas en C++. Su otro objetivo era construir un lenguaje para enseñar cómo programar a estudiantes de diseño y de arte y también brindarles una manera más fácil de trabajar con gráficas a estudiantes más avanzados. Esta combinación es una desviación positiva de la manera en que comúnmente se enseña programación. Los nuevos usuarios empiezan concentrándose en gráficas e interacción en vez de estructuras de datos y resultados en forma de texto en la consola. A través de los años, Processing se ha transformado en una gran comunidad. Es usado en salas de clases en todo el mundo, en planes de estudios de artes, humanidades y ciencias de la computación, además de profesionales. Hace dos años, Ben y Casey se me acercaron con una pregunta: ¿cómo se vería Processing si funcionara en la web? p5.js empieza con el objetivo original de Processing, hacer que programar sea accesible para artistas, diseñadores, educadores y principiantes, y luego lo reinterpreta para la web actual usando Javascript y HTML.

Internal links

You can link to other files simply by making an internal link. The build process will automatically search for this ID in all files, and append the filename if needed for the particular format.

Images

You can insert images simply by adding an image tag with the name of the image. This will look for an image located in `images/bruce.jpg`, but you can easily change this location if you want.

Code examples

Code examples can be written using the markdown syntax, and they will automatically be converted to HTMLBook programlistings. Here's an example of using the `console.log` function.

```
console.log("hello");
```

Footnotes

You can also write footnotes using the Markdown syntax[1], or the HTMLBook syntax[2].

Then you use the following `footnotes` liquid tag to insert the footnote references on the page. This is often done at the bottom of the page.

1. They are great
2. They are great too

HTML Chapter

This is a chapter written in HTML