

Neil's News

▶ [2023](#)

▼ [2022](#)

- [Webb Mirror](#)
- [Backwards Debugging](#)
- [Ingenook Shunting](#)
- [Blockly at 10](#)
- [FPS in JavaScript](#)
- [Polybahn](#)
- [Background Radiation](#)
- [Six Years](#)

▶ [2021](#)

▶ [2020](#)

▶ [2019](#)

▶ [2018](#)

▶ [2017](#)

▶ [2016](#)

▶ [2015](#)

▶ [2014](#)

▶ [2013](#)

▶ [2012](#)

▶ [2011](#)

▶ [2010](#)

▶ [2009](#)

▶ [2008](#)

▶ [2007](#)

▶ [2006](#)

▶ [2005](#)

▶ [2004](#)

▶ [2003](#)

▶ [2002](#)

FPS in JavaScript

12 May 2022

Some web pages have enough moving parts that running the page can create choppy animations or user interactions. In my case I'm dealing with Blockly when there are more than a thousand blocks on screen. As always, the first step in improving performance is to measure it. The modifications are beneficial or not.

Chrome's developer tools has a neat feature which [displays the frames per second](#) (FPS) that the page is rendering at. It's great, except the console completely changes the performance profile of the browser. So how can we measure FPS without invalidating the data?

Here's a simple JavaScript snippet which will display the FPS in the page title:

```
view plain print ?
01. var fpsHistory = [];
02. var then = Date.now();
03. (function render() {
04.     var now = Date.now();
05.     var elapsed = now - then;
06.     if (elapsed) {
07.         then = now;
08.         var fps = 1 / (elapsed / 1000);
09.         if (fpsHistory.length >= 10) fpsHistory.shift();
10.         fpsHistory.push(fps);
11.         var avg = 0;
12.         for (var i = 0; i < fpsHistory.length; i++) avg += fpsHistory[i];
13.         avg /= fpsHistory.length;
14.         document.title = "FPS: " + Math.round(avg);
15.     }
16.     requestAnimationFrame(render);
17. })();
```

Just copy this snippet into the web page, or paste it into the console (remembering to close the console afterwards). Here's a shorter version:

```
view plain print ?
01. var b=[],d=Date.now();(function e(){var a=Date.now(),c=a-d;if(c){d=a;a=1/(c/1E3);10<=b.length&&b.shift();b.push(a);for(c=a=0;c<b.length;c++)a+=b[c];a/=b.length;document.title="FPS: "+Math.round(a);}});
```

The script is currently running on this page, so you should see the FPS in the title of the current tab. To test it, press this to cause performance profiles of browsers vary wildly; Firefox and Safari slow down when they do work, whereas Chrome speeds up.

< [Previous](#) | [Next](#) >

Legal yada yada: My views do not necessarily represent those of my employer or my goldfish.

[Neil Fraser](#): [News](#): [FPS in JavaScript](#)