

Diversion: Generating a random color from JavaScript

 devblogs.microsoft.com/oldnewthing/20121023-00

October 23, 2012



Raymond Chen

A colleague posed a little puzzle for a fun little app he was writing in HTML: He wanted to generate a random color.

If you search around the intertubes, you can find several possible attempts at a solution, like [this collection](#), and [an interesting example that has trouble with the pigeonhole principle](#).

The original function to generate a random color went like this:

```
// Pad a string of up to two characters with a leading zero
// so the result is always exactly two characters long.
function padZero(v) {
    return (v.length == 1) ? '0' + v : v;
}
function randomColor() {
    return "#" + padZero(Math.floor(Math.random() * 256)).toString(16) +
                padZero(Math.floor(Math.random() * 256)).toString(16) +
                padZero(Math.floor(Math.random() * 256)).toString(16);
}
```

Can you do better? (My solution after the jump.)

That was a short jump.

My first simplification was recognizing that three random 8-bit values is the same as one random 24-bit value.

```
function padZeros6(v) {
    while (v.length < 6) v = "0" + v;
    return v;
}
function randomColor() {
    return "#" +
        padZeros6(Math.floor(Math.random() * 16777216).toString(16));
}
```

Next, I got rid of the `padZeros6` function by simply setting bit 25 to force a 7-digit result, then removing the leading 1.

```
function randomColor() {
  return "#" +
    (Math.floor(Math.random() * 16777216) +
     16777216).toString(16).substr(1);
}
```

Finally, I did some factoring.

```
function randomColor() {
  return "#" +
    Math.floor((1 + Math.random()) * 16777216).toString(16).substr(1);
}
```

That last bit was a bit dodgy due to the wonders of floating point arithmetic, but hey, it's a puzzle now.

Finally, I realized that CSS supports `#rgb` as shorthand for `#rrggbb`, so if you don't mind that your color palette is reduced to 4096 colors (and in the case of my colleague's little app, that was not an issue), you can shorten it a bit more:

```
function randomColor() {
  return "#" +
    Math.floor((1 + Math.random()) * 4096).toString(16).substr(1);
}
```

[Raymond Chen](#)

Follow

