Applause from you and 762 others

Tal Bereznitskey [Follow]

Making stuff with ASCII letters — CTO @ https://toriihq.com

Dec 21, 2017 · 2 min read
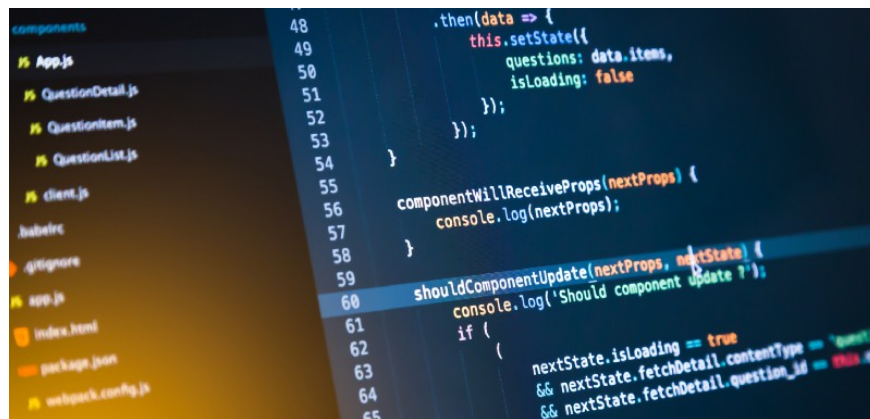
7 Hacks for ES6 Developers

Following the original JavaScript hacks, here's some new goodies.
Coding JavaScript in 2018 is actually fun again!

Hack #1 — Swap variables

Using `Array Destructuring` to swap values

```
let a = 'world', b = 'hello'
[a, b] = [b, a]
console.log(a) // -> hello
console.log(b) // -> world


// Yes, it's magic
```

Hack #2 — Async/Await with Destructuring

Once again, `Array Destructuring` is great. Combined with `async/await` and promises to make a complex flow—simple.

```
const [user, account] = await Promise.all([
  fetch('/user'),
  fetch('/account')
])
```

Hack #3—Debugging

For anyone who likes to debug using `console.log` s, here's something awesome (and yes, I heard of `console.table`):

```
const a = 5, b = 6, c = 7
console.log({ a, b, c })

// outputs this nice object:
// {
//    a: 5,
//    b: 6,
//    c: 7
// }
```

Hack #4—One liners

Syntax can be so much more compact for array operations

```
// Find max value
const max = (arr) => Math.max(...arr);
max([123, 321, 32]) // outputs: 321

// Sum array
const sum = (arr) => arr.reduce((a, b) => (a + b), 0)
sum([1, 2, 3, 4]) // output: 10
```

Hack #5—Array concatenation

The spread operator can be used instead of `concat` :

```
const one = ['a', 'b', 'c']
const two = ['d', 'e', 'f']
const three = ['g', 'h', 'i']

// Old way #1
const result = one.concat(two, three)

// Old way #2
const result = [].concat(one, two, three)

// New
const result = [...one, ...two, ...three]
```

Hack #6—Cloning

Clone arrays and objects with ease:

```
const obj = { ...oldObj }
const arr = [ ...oldArr ]
```

Note: This creates a shallow clone.

Hack #7—Named parameters

Making function and function calls more readable with destructuring:

```
const getStuffNotBad = (id, force, verbose) => {
  ...do stuff
}
const getStuffAwesome = ({ id, name, force, verbose }) => {
  ...do stuff
}

// Somewhere else in the codebase... WTF is true, true?
getStuffNotBad(150, true, true)

// Somewhere else in the codebase... I ♥ JS!!!
getStuffAwesome({ id: 150, force: true, verbose: true })
```

Already knew them all?

You're a true hacker, let's continue the conversation on **Twitter**.