

Applause from you and 261 others



Tal Bereznikey

Follow

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

Dec 6, 2013 · 4 min read

Javascript Hacks for Hipsters

This post was originally posted on my personal blog and has lots of insights in the comments.

Javascript is so much fun, except when it's not.

There's always the fear of runtime errors that keeps us thinking all the time while writing code. It makes us better coders—we have no other option than to visualize every line of code as if it's running as we write it.

That's why it's so important to have tidy code. Small code. Pretty code. Code you just fall in love with. Otherwise, Javascript will scare you away.

I gathered some fun snippets I enjoy using instead of boring code that takes too much space. Some makes the code shorter, cleaner and more readable. Other are just plain hacks for debugging.

I learned all of this from open source code (until node.js all javascript code was open source, wasn't it?), but I'll write them here is if I invented them.

Hipster Hack #1—Method calling

I really hate if/else blocks and this is quite a useful trick to call the right function based on a boolean value.

```
// Boring  
  
if (success) {
```

```
obj.start();

} else {

  obj.stop();

}

// Hipster-fun

var method = (success ? 'start' : 'stop');

obj[method]();
```

Hipster Hack #2—String joins

It's a known fact that strings like other strings. Sooner or later you'd like to concatenate two or more of them. I don't really like +ing them together, so *join()* comes to the rescue.

```
['first', 'name'].join(' '); // = 'first name';

['milk', 'coffee', 'suger'].join(', '); // = 'milk, coffee,
suger'
```

Hipster Hack #3—Default Operator ||

Javascript is all about not knowing what an object holds. Sometime you get it as a function argument, other times you might read it from the network or a configuration file. Either way, you can use the || operator to use the second argument if the first is falsy.

```
// default to 'No name' when myName is empty (or null, or
undefined)

var name = myName || 'No name';

// make sure we have an options object
```

```
var doStuff = function(options) {  
  
    options = options || {};  
  
    // ...  
  
};
```

Hipster Hack #4—Guard Operator &&

Similar to the *Default Operator*, this one is super useful. It eliminates almost all *IF* calls and makes for a nicer code.

```
// Boring  
  
if (isThisAwesome) {  
  
    alert('yes'); // it's not  
  
}  
  
// Awesome  
  
isThisAwesome && alert('yes');  
  
// Also cool for guarding your code  
  
var aCoolFunction = undefined;  
  
aCoolFunction && aCoolFunction(); // won't run nor crash
```

Hipster Hack #5—XXX Operator

This one is totally copyrighted and also SFW. Whenever I write some code, but then have to consult the web, or a different part of the code, I add an `xxx` line to the code. This makes the code break so I can get back to the specific place and fix it later. Much easier to search for it (`xxx` usually never appears) and you don't have to think about a TODO comment.

```
var z = 15;

doSomeMath(z, 10);

xxx // Great placeholder. I'm the only one using xxx and
it's so easy to find in code instead of TODOs

doSomeMoreMath(z, 15);
```

Hipster Hack #6—Timing

Ever wonder what's faster: Looping with an `i++` or looping with an `i--`? Yeah, me neither. For those who does, you can use the console's timing methods to test for slow loops or any other event-loop blocking code.

```
var a = [1,2,3,4,5,6,7,8,9,10];

console.time('testing_forward');

for (var i = 0; i < a.length; i++);

console.timeEnd('testing_forward');

// output: testing_forward: 0.041ms

console.time('testing_backwards');

for (var i = a.length-1; i >= 0; i-);

console.timeEnd('testing_backwards');

// output: testing_backwards: 0.030ms
```

Hipster Hack #7—Debugging

I learned this one from a Java developer. I have absolutely no idea how he knew about it and I didn't, but I've been using it ever since. Just drop a *debugger* statement and the debugger will stop on that line.

```
var x = 1;

debugger; // Code execution stops here, happy debugging

x++;

var x = Math.random(2);

if (x > 0.5) {

  debugger; // Conditional breakpoint

}

x--;
```

Hipster Hack #8—Old School Debugging

I’ve always been a “printf debugger” more than a line-by-line-in-a-debugger kind of developer. If you’re like me, you’ll want to “export” some private vars into the global scope in order to examine them from time to time. Don’t forget to remove these before committing/pushing-to-production.

```
var deeplyNestedFunction = function() {

  var private_object = {

    year: '2013'

  };

  // Globalize it for debugging:

  pub = private_object;

};

// Now from the console (Chrome dev tools, firefox tools,
etc)

pub.year;
```

Hipster Hack #9—Ultra Light Templates

Are you still concatenating strings using the + operator? Here's a better way to combine a sentence with your data. It's called templating and here's a mini framework in 2.5 lines of code.

```
var firstName = 'Tal';

var screenName = 'ketacode'

// Ugly

'Hi, my name is ' + firstName + ' and my twitter screen name
is @' + screenName;

// Super

var template = 'Hi, my name is {first-name} and my twitter
screen name is @{screen-name}';

var txt = template.replace('{first-name}', firstName)

.replace('{screen-name}', screenName);
```

Already knew them all?

Even the copyrighted XXX Operator invented by me?! You're a true hipster hacker, let's talk more on [twitter](#).

[Hacker Noon](#) is how hackers start their afternoons. We're a part of the [@AMIfamily](#). We are now [accepting submissions](#) and happy to [discuss advertising & sponsorship opportunities](#).

To learn more, read our [about page](#), [like/message us on Facebook](#), or simply, [tweet/DM @HackerNoon](#).

If you enjoyed this story, we recommend reading our [latest tech stories](#) and trending tech stories. Until next time, don't take the realities of the world for granted!



