

②ality – JavaScript and more

[About](#)

[Donate](#)

[Subscribe](#)

[ES2017](#)

[Books \(free online!\)](#)

Most popular
(last 30 days)

ECMAScript 6 modules:
the final syntax

ECMAScript 2017: the
final feature set

ES proposal: import()
– dynamically importing
ES modules

Making transpiled ES
modules more spec-
compliant

ES proposal: Shared
memory and atomics

Classes in ECMAScript
6 (final semantics)

Communicating
between Web Workers
via MessageChannel

Most popular
(all time)

ECMAScript 6 modules:
the final syntax

Classes in ECMAScript
6 (final semantics)

Iterating over arrays and
objects in JavaScript

ECMAScript 6's new
array methods

The final feature set of
ECMAScript 2016 (ES7)

WebAssembly: a binary
format for the web

Basic JavaScript for the
impatient programmer

Google Dart to
“ultimately ... replace
JavaScript”

Six nifty ES6 tricks

Google's Polymer and
the future of web UI
frameworks

Blog archive

► [2017](#) (4)

► [2016](#) (38)

Free email newsletter: “[ES.next News](#)”

2013-07-14

Trailing commas in object literals and array literals

Labels: [dev](#), [javascript](#), [jslang](#)

Quick reminder: trailing commas in object literals are legal in ECMAScript 5, trailing commas in arrays are ignored.

Trailing commas in object literals

Thus, if you can afford to ignore older JavaScript engines, you can write your object literals like this:

```
var obj = {  
    first: 'Jane',  
    last: 'Doe',  
    age: 40, // trailing comma  
};
```

The advantage of adding a trailing comma is that you can rearrange the innards of the literal without having to worry about commas being in the right places.

Trailing commas in array literals

Similarly, trailing commas in arrays are ignored:

```
> var arr = [ 'a', 'b', 'c', ];  
> arr  
[ 'a', 'b', 'c' ]  
> arr.length  
3
```

This goes so far that you need to write two trailing commas if you want to add a trailing hole [1]:

```
> var arr = [ 'a', 'b', , ];  
> arr.length  
3
```

Trailing commas in JSON

JSON [2] is based on JavaScript's syntax prior to ECMAScript 5, which means that trailing commas are illegal:

```
> JSON.parse('{"x":1,}')  
SyntaxError: Unexpected token }  
> JSON.parse('[1,]')  
SyntaxError: Unexpected token ]
```

What browsers support object literals with trailing commas?

Most browsers support object literals with trailing commas. You are only out of luck in Internet Explorer 8 and earlier ([compatibility table](#)).

References

[1] [Arrays in JavaScript](#)

(Ad, please don't block.)



90% Unlimited
Downloads Choose from
Over 300,000 Vectors,
Graphics & Photos.
ads via Carbon



Dr. Axel Rauschmayer

Free online books by Axel

[Speaking JavaScript
\[up to ES5\]](#)

[Exploring ES6](#)

[JavaScript training:
Ecmascript](#)

- 2015 (65)
- 2014 (55)
- ▼ 2013 (98)
 - December (6)
 - November (3)
 - October (4)
 - September (6)
 - August (6)
- ▼ July (8)
 - A meta style guide for JavaScript
- ECMAScript 6 modules: the future is now
- In defense of JavaScript's constructors
- Hello Polymer: Q&A with Google's Polymer team
- Trailing commas in object literals and array liter...
- Performance optimizations and for loops
- Array iteration and holes in JavaScript
- Feeds (RSS, Atom) in the time of social media
- June (11)
- May (11)
- April (12)
- March (8)
- February (10)
- January (13)
- 2012 (177)
- 2011 (380)
- 2010 (174)
- 2009 (68)
- 2008 (46)
- 2007 (12)
- 2006 (1)
- 2005 (2)

Labels

- [dev](#) (592)
- [javascript](#) (400)
- [computers](#) (316)
- [life](#) (194)
- [jslang](#) (179)
- [esnext](#) (156)
- [apple](#) (107)
- [webdev](#) (95)

[2] JavaScript's JSON API

12 Comments The 2ality blog

Login

Sort by Best



Join the discussion...



derp • 4 years ago

What about trailing commas in function parameters, are they allowed?

1 ▲ | ▼ • Reply • Share >

bfredit → derp • 7 months ago

There's a proposal for that: [http://www.2ality.com/2015/11/...](http://www.2ality.com/2015/11/)

^ | ▼ • Reply • Share >

Axel Rauschmayer Mod → derp • 4 years ago

No. But, IMO, less important, because one usually writes all parameters in a single line. Or, at the very least, the closing parenthesis immediately follows the last parameter.

^ | ▼ • Reply • Share >

Blago • 2 years ago

Is this allowed in ES6 ?

^ | ▼ • Reply • Share >

Axel Rauschmayer Mod → Blago • 2 years ago

ES6 is a strict superset of ES5, so anything that is allowed in ES5 is also allowed in ES6. ES6 introduces the spread operator (...), after which no trailing comma is allowed.

1 ▲ | ▼ • Reply • Share >

novakhun • 4 years ago

Yes, leave trailing commas if you do not want to support IE users :)

^ | ▼ • Reply • Share >

Axel Rauschmayer Mod → novakhun • 4 years ago

You are fine in Internet Explorer 9 and later; I've added a note.

^ | ▼ • Reply • Share >

novakhun → Axel Rauschmayer • 4 years ago

I'm just curious:

- What is the aim of this proposal?
- How does it improve anyone's code?
- Why should I leave a trailing comma?

^ | ▼ • Reply • Share >

mostwanted → novakhun • 2 years ago

when adding an array element, source code versioning tools will mark 1 instead of 2 lines as modified if you used a trailing comma for the previous last array element.

1 ▲ | ▼ • Reply • Share >

Danny → novakhun • 8 months ago

While arguably not the place of ECMAScript to worry about server scripting, this permissiveness is also beneficial when generating arrays/objects in a server scripting language not via JSON. Ordinarily this requires a loop and some form of string incantation

Tweets by @rauschma

Axel Rauschmayer @rauschma

Enjoying "Troll Hunters":

- "Let's call him 'Gnome Chomsky'"
- "Juliet dies in this? Nooo!"

10h

Axel Rauschmayer Retweeted

Jonathan Cramer @jcreamer898

A nice little shout out to @rauschma... infoworld.com/article/316483 ... #es2017 #async

JavaScr...
ECMASc...
infowor...

14h

Axel Rauschmayer @rauschma

Not a physics book! twitter.com/ManningBooks/s ...

02 Feb

Axel Rauschmayer Retweeted

Jordan Harband @ljharb

Making our React components forbid extra props has caught SO many bugs. I highly recommend it.

npmjs.com/airbnb-prop-ty...

npm: air...
Custom ...
npmjs.com

02 Feb

Axel Rauschmayer Retweeted

Seth Petry-Johnson @spetryjohnson

Open strong. Be bold. Tell a story. Do something to get me interested. Opening w/ the "obligatory 'about me' slide" puts me to sleep :)

02 Feb

Google

mobile (83)
scitech (50)
hack (49)
mac (47)
google (39)
java (37)
ios (33)
business (32)
video (32)
clientjs (31)
hci (27)
entertainment (26)
nodejs (26)
society (26)
browser (25)
firefox (25)
html5 (24)
ipad (24)
movie (23)
psychology (22)
2ality (18)
tv (18)
android (17)
social (17)
chrome (16)
fun (16)
jsmodules (16)
tablet (16)
humor (15)
politics (15)
web (15)
cloud (14)
hardware (14)
microsoft (14)
software engineering (13)
blogging (12)
gaming (12)
eclipse (11)
gwt (11)
numbers (11)
programming languages (11)
app store (10)
media (10)
nature (10)
security (10)
semantic web (10)
software (10)
twitter (10)
webos (10)
12quirks (9)
education (9)
jstools (9)
photo (9)

requires a loop and some form of string concatenation. Before this feature it was essential that the loop knew if it was the last iteration of the loop to decide if to add a comma at the end of the string. Now you can always add a comma and not worry about your iteration position in the loop.

^ | v • Reply • Share >

Axel Rauschmayer Mod → novakhun • 4 years ago

It is not a proposal, it is a feature of ECMAScript 5, the current version of JavaScript supported by all modern browsers.

There is only one advantage, you have to know whether it matters to you or not.. Quoting the blog post: "The advantage of adding a trailing comma is that you can rearrange the innards of the literal without having to worry about commas being in the right places."

^ | v • Reply • Share >

sheriffderek → Axel Rauschmayer • a year ago

I find that trailing commas are useful for git, because without them, you affect 2 lines when adding 1. (because you'll also have to add a comma to the previous line if the trailing comma was omitted.)

^ | v • Reply • Share >

 [Subscribe](#)  [Add Disqus to your site](#) [Add Disqus Add](#)  [Privacy](#)

Newer Post

Home

Older Post

Subscribe to: [Post Comments \(Atom\)](#)

[webcomponents](#) (9)
[windows 8](#) (9)
[async](#) (8)
[idea](#) (8)
[iphone](#) (8)
[itunes](#) (8)
[scifi-fantasy](#) (8)
[app](#) (7)
[babel](#) (7)
[bookmarklet](#) (7)
[chromeos](#) (7)
[english](#) (7)
[es proposal](#) (7)
[fringe](#) (7)
[html](#) (7)
[jsint](#) (7)
[jsshell](#) (7)
[thunderbolt](#) (7)
[webapp](#) (7)
[advancedjs](#) (6)
[blogger](#) (6)
[crowdsourcing](#) (6)
[latex](#) (6)
[lion](#) (6)
[promises](#) (6)
[ted](#) (6)
[book](#) (5)
[environment](#) (5)
[gadget](#) (5)
[googleio](#) (5)
[intel](#) (5)
[jsarrays](#) (5)
[jshistory](#) (5)
[layout](#) (5)
[light peak](#) (5)
[michael j. fox](#) (5)
[music](#) (5)
[pdf](#) (5)
[polymer](#) (5)
[shell](#) (5)
[tc39](#) (5)
[template literals](#) (5)
[underscorejs](#) (5)
[vlc](#) (5)
[__proto__](#) (4)
[coffeescript](#) (4)
[concurrency](#) (4)
[dart](#) (4)
[facebook](#) (4)
[gimp](#) (4)
[googleplus](#) (4)
[health](#) (4)
[howto](#) (4)
[hp](#) (4)

javafx (4)

kindle (4)

leopard (4)

macbook (4)

motorola (4)

münchen (4)

occupy (4)

pi fundamentals (4)

presenting (4)

publishing (4)

series (4)

textbook (4)

web design (4)

amazon (3)

asmjs (3)

back to the future (3)

bitwise_ops (3)

css (3)

es2016 (3)

flattr (3)

fluentconf (3)

food (3)

foreign languages (3)

house (3)

icloud (3)

info mgmt (3)

jsfuture (3)

jsstyle (3)

linux (3)

mozilla (3)

python (3)

regexp (3)

samsung (3)

tizen (3)

traffic (3)

typedjs (3)

unix (3)

adobe (2)

angry birds (2)

angularjs (2)

astronomy (2)

audio (2)

comic (2)

design (2)

dom (2)

ecommerce (2)

eval (2)

exploring es6 (2)

facebook flow (2)

facets (2)

flash (2)

free (2)

futurama (2)

guide (2)

[history](#) (2)
[hyena](#) (2)
[internet explorer](#) (2)
[iteration](#) (2)
[journalism](#) (2)
[jquery](#) (2)
[jsengine](#) (2)
[jslib](#) (2)
[law](#) (2)
[lightning](#) (2)
[markdown](#) (2)
[math](#) (2)
[meego](#) (2)
[month](#) (2)
[nike](#) (2)
[nokia](#) (2)
[npm](#) (2)
[programming](#) (2)
[raffle](#) (2)
[repl](#) (2)
[servo](#) (2)
[sponsor](#) (2)
[steve jobs](#) (2)
[travel](#) (2)
[typescript](#) (2)
[usb](#) (2)
[winphone](#) (2)
[wwdc](#) (2)

[airbender](#) (1)
[amdefine](#) (1)
[aol](#) (1)
[app urls](#) (1)
[architecture](#) (1)
[atscript](#) (1)
[basic income](#) (1)
[biology](#) (1)
[blink](#) (1)
[bluetooth](#) (1)
[canada](#) (1)
[clip](#) (1)
[coding](#) (1)
[community](#) (1)
[cross-platform](#) (1)
[deutsch](#) (1)
[diaspora](#) (1)
[distributed-social-network](#) (1)
[dsl](#) (1)
[dvd](#) (1)
[dzone](#) (1)
[emacs](#) (1)
[emberjs](#) (1)
[energy](#) (1)
[esnext news](#) (1)
[esprop](#) (1)

example (1)

facetator (1)

feedback (1)

firefly (1)

firefoxos (1)

fritzbox (1)

german (1)

git (1)

guest (1)

guice (1)

h.264 (1)

home entertainment (1)

hosting (1)

htc (1)

ical (1)

jsdom (1)

jsmith (1)

library (1)

location (1)

marketing (1)

mars (1)

meta-data (1)

middle east (1)

mpaa (1)

msl (1)

mssurface (1)

netflix (1)

nsa (1)

obama (1)

openoffice (1)

opinion (1)

oracle (1)

organizing (1)

philosophy (1)

pixar (1)

pnacl (1)

prism (1)

privacy (1)

proxies (1)

puzzle (1)

raspberry pi (1)

read (1)

rodney (1)

rust (1)

safari (1)

sponsoring (1)

star trek (1)

static generation (1)

talk (1)

technique (1)

theora (1)

thunderbird (1)

typography (1)

unicode (1)

v8 (1)

voice control (1)

webassembly (1)

webkit (1)

webm (1)

webpack (1)

yahoo (1)

Powered by [Blogger](#).