



Đăng nhập vào medium.com bằng Google



Don't use Semicolons in TypeScript!

Eugen Kiss [Follow](#)

Jul 16, 2018 · 4 min read

336

“Just use semicolons. It's a little thing and makes code safer.” No one up about semicolons shows that it's not a trivial matter. **Semicolons make code safer.** They are unnecessary in TypeScript and here's why.



Hiện Doãn

toilati123vn@gmail.com

TIẾP TỤC VỚI HIỆN

Để tạo tài khoản của bạn, Google sẽ chia sẻ tên, địa chỉ email và ảnh hồ sơ của bạn với medium.com. Bằng cách tiếp tục, bạn đồng ý với [chính sách quyền riêng tư](#) và [điều khoản dịch vụ](#) của medium.com.



Đăng nhập vào medium.com bằng Google



SEMICOLONS IN JAVASCRIPT ARE OPTIONAL DUE TO AUTOMATIC SEMICOLON INSERTION (ASI). TYPESCRIPT ALSO FOLLOWS ASI, TOO. ASI IS NOT STRAIGHTFORWARD AND THERE ARE A FEW SITUATIONS WHERE ADDING A SEMICOLON WILL LEAD TO AN UNEXPECTED RUNTIME ERROR. **BUT THE FEW CASES WHERE YOU NEED TO ADD A SEMICOLON ARE FURTHER ELIMINATED BY TYPESCRIPT'S TYPE SYSTEM.** Consider the following code:

```

1  class Bike {
2    color
3
4    constructor(color) {
5      this.color = color
6    }
7
8    ring() {
9      console.log('ring ring')
10   }
11 }
12
13 function logWithTimestamp(data) {
14   console.log(`new Date(): ${data}`)
15 }
16
17 const log = logWithTimestamp //;
18 ['purple', 'red'].forEach(color => (new Bike(color)).ring())
19 log('bikes created')
```

semicolon1.ts hosted with ❤ by GitHub

view raw



Hiện Doãn

toilati123vn@gmail.com

TIẾP TỤC VỚI HIỆN

Để tạo tài khoản của bạn, Google sẽ chia sẻ tên, địa chỉ email và ảnh hồ sơ của bạn với medium.com. Bằng cách tiếp tục, bạn đồng ý với [chính sách quyền riêng tư](#) và [điều khoản dịch vụ](#) của medium.com.

The `['purple', 'red']` part (line 18) is interpreted as a comma expression. In JavaScript you'll get a run-time error. **In TypeScript you'll get a compile-time error (TS2695: Left side of comma operator is unused and has no side effects).**

The handful of similar cases where I needed to add a semicolon to correct the program, TypeScript complained as it couldn't type the program.



336

```

2   ring() {
3     console.log('ring ring')
4   }
5 }
6
7 function logWithTimestamp(data) {
8   console.log(`new Date(): ${data}`)
9   return data
10}
11
12 const log = logWithTimestamp //;
13 (new Bike()).ring()
14 log('path cleared')

```

semicolon2.ts hosted with ❤ by GitHub

[view raw](#)

Đăng nhập vào medium.com bằng Google



Hiện Doãn

toilati123vn@gmail.com

TIẾP TỤC VỚI HIỆN

Để tạo tài khoản của bạn, Google sẽ chia sẻ tên, địa chỉ email và ảnh hồ sơ của bạn với medium.com. Bằng cách tiếp tục, bạn đồng ý với [chính sách quyền riêng tư](#) và [điều khoản dịch vụ](#) của medium.com.

In both JavaScript and TypeScript you'll get a runtime error (`TypeError: log is not a function`).

Add no-unexpected-multiline to TSLint.

Then the previous example will fail at compile-time, too.

Just prefix lines starting with [, (, or ` with a semicolon. **But you don't even have to remember this since TypeScript and TSLint will save you.**

The examples have been taken from Olli Ahonen's "Should I use Semicolons in TypeScript?".



Olli Ahonen
@olliahonen

Should you use end-of-line `#semicolons` in `#TypeScript`? Read about it here: github.com/olliahonen/should-i-use-semicolons-ts ;

1 6:49 PM - Mar 26, 2018

[olliahonen/should-i-use-semicolons-ts](https://github.com/olliahonen/should-i-use-semicolons-ts)

Should I use semicolons in TypeScript?

Contribute to olliahonen/should-i-use-semicolons-ts development by creating an account on



Đăng nhập vào medium.com bằng Google

[See Olli Ahonen's other Tweets](#)

Advantages

Programming is a cognitive demanding task. It also requires more writing, editing and reading code, and how your tooling (IDE) handles it.

- Your IDE bothers you less ('semicolon missing'). Less distraction.
- Cursor navigation, mouse pointing, is easier as you don't have to type the semicolon.
- Less typing.

Common Counter Arguments

Here I list counter arguments and my thoughts on them.

Better safe than sorry!

What risk are you preventing? **How often do you start a line with [**? You still have to think about special cases from ASI even with semicolons. And you pay with a redundant character for every line.

A small price for a single line but a big price for the whole code base. Plus, as shown here, you are safe with TypeScript and TSLint.

You need to remember complicated rules if you omit semicolons!

Even when you use semicolons **you still need to be aware of special cases**. It is easier to remember when a semicolon is required vs. when it is not. Plus, as shown here, you don't even have to remember anything if you use `no-unexpected-multiline`.

You need to remember complicated rules if you omit semicolons due to new experimental language

features whose ASI hazards are not yet properly known! Class Fields Example, ASI Hazard Clause

This only becomes an issue **when you write esoteric code**. It's a hypothetical risk. It's easier to not do it (because you have to write such code deliberately) than to use semicolons everywhere.

If you are not in an environment with TypeScript and TSLint (JSBin) you will be bitten by missing semicolons again!

The risk is already very low in pure JavaScript. Are you hosting your business critical code on JSBin? **Why should you optimize for environments that you use only rarely instead of**

optimizing for the environment you use all the time? The argument blows up a minuscule risk.

Semicolons provide a visual cue which makes code easier to read!

What about the new line? **Isn't the new line cue enough?** Admittedly, Python requires a technically unnecessary colon after if statements. The given reason is readability. But in that case



Hiện Doãn

toilati123vn@gmail.com

TIẾP TỤC VỚI HIỆN

Để tạo tài khoản của bạn, Google sẽ chia sẻ tên, địa chỉ email và ảnh hồ sơ của bạn với medium.com. Bằng cách tiếp tục, bạn đồng ý với [chính sách quyền riêng tư](#) và [điều khoản dịch vụ](#) của medium.com.



Đăng nhập vào medium.com bằng Google



Hiện Doãn

toilati123vn@gmail.com

TIẾP TỤC VỚI HIỆN

Để tạo tài khoản của bạn, Google sẽ chia sẻ tên, địa chỉ email và ảnh hồ sơ của bạn với medium.com. Bằng cách tiếp tục, bạn đồng ý với [chính sách quyền riêng tư](#) và [điều khoản dịch vụ](#) của medium.com.

336

Doesn't convince me. I claim code's clearer without semicolons. **without semicolons**. Many of their users even show genuine error in the burden of semicolons.

Not using semicolons is bad for tools like transpilers and minifiers. Code is for humans to read and write. If a tool parses the code then the tool is broken. Plus, nowadays this problem doesn't even exist.

Conclusion

Use TypeScript (with no-unexpected-multiline) and safely omit semicolons.

You don't have to remember any rules. The tooling will remind you in the rare case you need to prefix a line with semicolon. Semicolons are not free. Free yourself from their tyranny. //;

Update: Nowadays, I'm only using prettier with the `semi: false` option. Together with an IDE extension to auto-format on save it makes editing faster as you can be messier.

Sources

- <https://github.com/olliahonen/should-i-use-semicolons-ts/>
- <https://stackoverflow.com/a/41422345/283607>
- <https://mislav.net/2010/05/semicolons/>
- <https://medium.com/@goatslacker/no-you-dont-need-semicolons-148d936b9cf2>
- <https://blog.kentcdodds.com/semicolons-in-javascript-a-preference-dd8fc8b80895>
- <http://blog.izs.me/post/2353458699/an-open-letter-to-javascript-leaders-regarding>
- <https://standardjs.com/rules.html#semicolons>

. . .

If you enjoyed this article, please recommend and share it! 😊
Retweeting is a great way to share:



Đăng nhập vào medium.com bằng Google



Hiện Doãn

toilati123vn@gmail.com

TIẾP TỤC VỚI HIỆN

336

Để tạo tài khoản của bạn, Google sẽ chia sẻ tên, địa chỉ email và ảnh hồ sơ của bạn với medium.com. Bằng cách tiếp tục, bạn đồng ý với [chính sách quyền riêng tư](#) và [điều khoản dịch vụ](#) của medium.com.

JavaScript

TypeScript

Programming

Front End Development

Coding



336 claps

...
oo

WRITTEN BY

Eugen Kisseugenkiss.com

Follow

See responses (13)



More from Medium

More from Eugen Kiss

336

Lean Testing or Why Unit Tests are Worse than You Think



Jun 30, 2018 · 8 mi...



7.3K



Jun 13, 2018 · 7 m...



1.97K



Related reads

Sharing UI Components with Lerna and Yarn Workspaces



Hiện Doãn

toilati123vn@gmail.com

TIẾP TỤC VỚI HIỆN

Để tạo tài khoản của bạn, Google sẽ chia sẻ tên, địa chỉ email và ảnh hồ sơ của bạn với medium.com. Bằng cách tiếp tục, bạn đồng ý với [chính sách quyền riêng tư](#) và [điều khoản dịch vụ](#) của medium.com.

Functional Patterns with TypeScript



John Tucker in...
Jan 4 · 5 min read



1.3K

