search   🔔 1   ⊞

objects? No... array, please!

👤 Fabio Russo 🐦 🐙 May 4

#javascript   **#arrays**   **#object**   #beginners

I don't like objects...that much!

This is an *object*:

```
const obj = {breed:"labrador",age:9}
```

But sometimes I prefer to work with arrays.

Why? Because they really look better **to me**... and there are really a lot of methods or loops that work *just* with [arrays]!

These are some tools used to *"convert"* objects to arrays.

```
//Object.values() will give you an array of all the object "values"

const obj = {breed:"labrador",age:9}

const values = Object.values(obj)

console.log(values)

//-> ["labrador", 9]
```

```
const obj = {breed:"labrador",age:9}

const keys = Object.keys(obj)

console.log(keys)

//-> ["breed", "age"]




//Object.entries()  will give you an arraysh version of the object.
//Where the key and the value will be paired into an array...
//and all of those arrays will be "pushed" into another array.

const obj = {breed:"labrador",age:9}

const entries = Object.entries(obj)

console.log(entries)

//->[["breed", "labrador"], ["age", 9]]
```
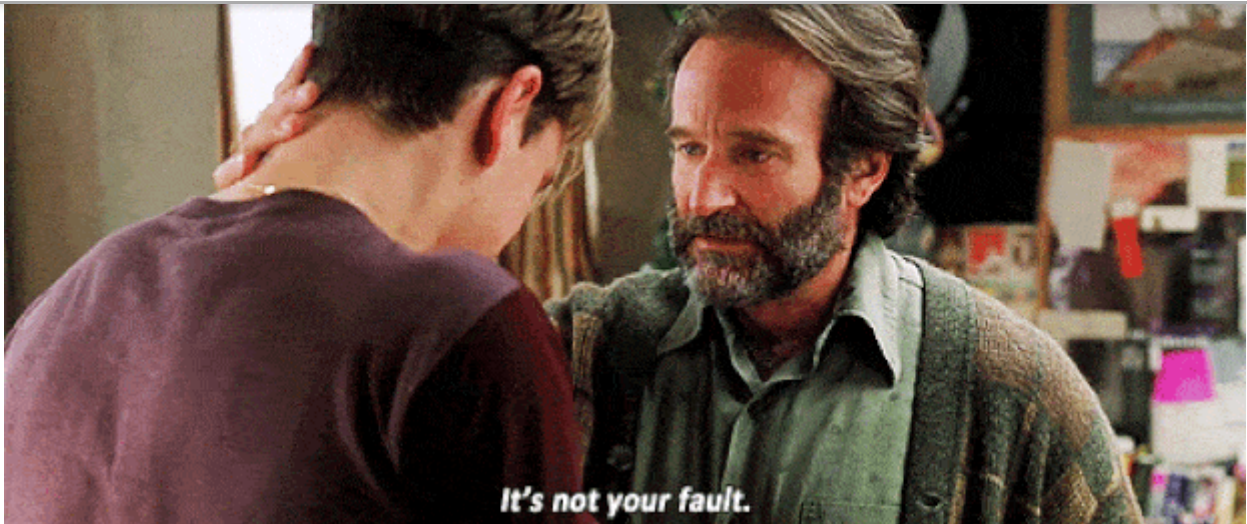
This is *easy peasy* stuff, but very often, at the beginning of my journey in *JS*, objects were very often a problem for me.

If only they had told me before ...

P.S: These tools are ok... if it's ok to work with arrays instead of objects.
Sometimes you've to use objects... because of performance or long term maintenance.

❤️ 10	🦄 6	⚡ 31	▪▪▪

dev.to is where software developers stay in the loop and avoid career stagnation.

Signing up (for free!) is the first step.

Fabio Russo + FOLLOW

Passionate with code... In love with Web Design, Videogames, Books and Jazz music. Years of experience in "document composition" and a Front-End Developer wannabe.

🐦 RussoFabio7   ⬡ Gentarozzo   🔗 codepen.io/Gentarozzo/full/OWZvEE

```
Add to the discussion
```

ⓘ   PREVIEW   SUBMIT

Be careful falling into what you live vs. what works best for the solution. Many times, using an array might make you fall into an anti-pattern you won't be able to support. For example, when your dataset requires a map data structure, or a graph data structure, you can't very easily replicate those in an array in most languages.

♡ 6   REPLY

Fabio Russo 🐦 🐙   May 5 ⌄

That's hm... obvious ? 😄

♡ 1   REPLY

Donald Merand 🐙   May 5 ⌄

Obvious to whom? I found these examples, which you didn't mention in your article, to be a helpful caveat.

♡ 2   THREAD

Fabio Russo 🐦 🐙   May 5 ⌄

They're... I can put those in the article if it's ok for him.

♡ 1   REPLY

Alain Van Hout 🐙   May 5 ⌄

Using arrays for everything will get you in the same place as using varchar columns for everything in an sql database: it might seem to save some effort and thinking in the short term, but it will be an enormous pain to maintain in the long run.

♡ 3   REPLY

Fabio Russo 🐦 🐙   May 5 ⌄

Just someway to do It.
I agree that sometimes you've to work with It.
Consider the DOM... most of the times it's easier to work with arrays.

♡ 1   REPLY

In the case of the DOM, you're already working with objects, which just happen to be composed into larger hierarchies via arrays. Imagine if a DOM object were an array, where you'd have to look up each property in the array rather than having direct access to it. It'd be cumbersome to work with, not to mention the inherently poor performance that it would entail.

♡ 1   THREAD

Fabio Russo 🐦 ⚫   May 5 ⌄

Most of the time with nodeList... I know what you're talking about but I use to convert them in arrays if there's not a HUGE amount of stuff to do.

♡ 1   THREAD

Alain Van Hout ⚫   May 5 ⌄

Yes, that allows you to leverage the extra methods (i.e. power) of javascript arrays (since a nodeList is just a poor-man's array anyway). But that's quite different from the approach you discuss above, about using arrays instead of objects with fields.

♡ 1   THREAD

Fabio Russo 🐦 ⚫   May 5 ⌄

Maybe It was not so clear ☹

♡ 1   REPLY

Jamie Bertram ⚫   May 5 ⌄

Hum, I find this odd... I also don't think this article gives a complelling reason to back up the claim (a claim which I do not agree with at all)... isn't this more or less unstructured programming? And how do you reliably work with arrays? Conventions for what index a given field should be present in? Seems dangerous and easy to screw up.

♡ 3   REPLY

Fabio Russo 🐦 ⚫   May 5 ⌄

These are tools...
You can use It... you can also chose not to use It.

♡ 1   REPLY

Jamie Bertram ◯   May 6 ⌄

Absolutely, arrays are tools, essential ones! I guess it isn't obvious what the point of the article is... it reads like you're advocating packing data into arrays rather than data structures with named fields.

♡ 2   THREAD

Fabio Russo 🐦 ◯   May 6 ⌄

That's why there's a **bold** P.S. at the end of It.
If you're conscious, you can chose to do It and use arrays.
If you dunno the differences, first you should learn those.

I use these tools when I work with the DOM or with very small amounts of data. Just to solve some problem I won't care in the future about.

♡ 1   THREAD

Jamie Bertram ◯   May 6 ⌄

Ok I understand, that makes sense. Obviously for experimentation anything goes, and if arrays help you visualize the data then that's a neat perspective on how to use them!

Like I said, the article doesnt make this idea very clear, which is probably what lead to all the confusion in the comments.

♡ 2   THREAD

Fabio Russo 🐦 ◯   May 6 ⌄

I've to agree... I'll do better next time.

♡ 1   REPLY

Ben Halpern

Hey there, we see you aren't signed in. (Yes you, the reader. This is a fake comment.)

Please consider creating an account on dev.to. It literally takes a few seconds and *we'd appreciate the support so much*. ♡

JOIN THE DEV COMMUNITY

Tiffany White 🐦 🔘   May 5 ⌄

There are times when you should use an object over an array for storing data. And if it is performance or efficiency that's your jam, using a `Set` is much more efficient than an array, again, depending on what you're looking to do. Swearing off objects because they don't *look* good is not an effective strategy in long-term development.

♡ 2   REPLY

> Fabio Russo 🐦 🔘   May 5 ⌄
>
> No one told... you've to... without considering It.
>
> ♡ 1   REPLY

Donald Merand 🔘   May 5 ⌄

Have you looked at object destructuring much? Seems like it can be a decent way to get array-like access to objects, without giving them up entirely.

It's also worth noting that you pay an object-traversal performance penalty every time you extract keys or values.

♡ 2   REPLY

> Fabio Russo 🐦 🔘   May 5 ⌄
>
> I'll check it out
>
> ♡ 2   REPLY

Avalander 🔘   May 5 ⌄

I'm confused about why you find objects problematic. You've stated that arrays look better to you and that objects were often a problem for you when you started with Javascript. Would you elaborate on what is it that you find problematic about objects and why you prefer working with arrays?

♡ 1   REPLY

More properties and during my work I do most of the stuff with data records... and arrays.
I started working a lot with katas... dunno why, most of them are about arrays.
It's just that I feel more comfortable with them.

♡ 1   REPLY

code of conduct - report abuse

Programmers who only code at work

Jan

What's your opinion on programmers who are not passionate about programming, ha...

❤ 172   💬 115

READ POST   SAVE FOR LATER

First Ever DEV Contest: Build a Realtime App with Pusher

Jess Lee

The theme of this contest is to create a real-time app or a hack that uses Pusher Channels real-time API.

❤ 281   💬 77

READ POST   SAVE FOR LATER

Another Post You Might Like

What is JAM Stack

Leo Melo

This post was based on a talk I gave at a local web development meetup in Jerse...

❤ 113　　〰 15

READ POST　　SAVE FOR LATER

Building a Pomodoro Timer with Vue.js on CodePen
Tori Pugh - May 31

Play With the React 🐘Router
Sai gowtham - Jun 1

Under the Hood of the Most Powerful Video JavaScript API
The JW Player Team - Jun 1

Think Like a Programmer??
Cat Carbonell - May 30

Home　About　Sustaining Membership　Privacy Policy　Terms of Use

Contact　Code of Conduct　The DEV Community copyright 2018 🔥