



Denoは混沌としたJS界を救う

weblio study 開発チーム

鷹勇 @Takao

今もなお、JavaScript界は、混沌に満ちています



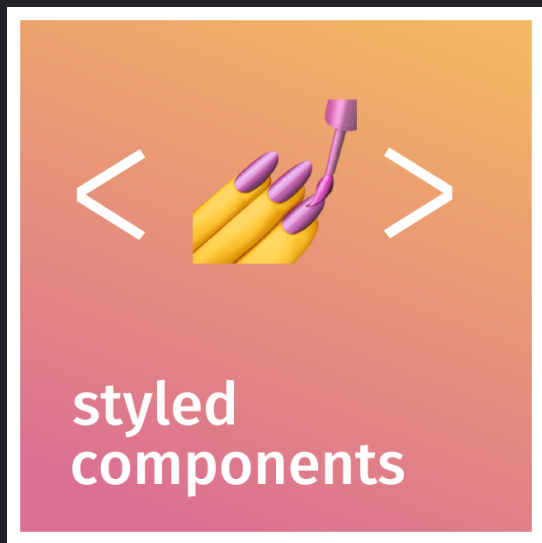
たしかにTypeScriptの普及により
JSによる開発は快適になりました。



Gulp や bower は死に絶え、
公式のツール（npm）で開発ができるようになりました



webpack の席卷も
Vite が覆そうとしています。



ReactやVueのエコシステムは
CSSの名前衝突を現実的に解決しました

しかし！

JS界には根本的な問題がある！！！！！！

- node.js,npm,webpack などによる**複雑なビルド**をしないと**TSが使えない**
- npm による依存関係のブラックボックス化により、node_modules下は魔境と化している
- ESM? commonJS? 意味不明！（require, import/export 方式が混在する）
- npm による依存関係解決が複雑

しかし、これは node.js が急成長を遂げた
一因になったとも考えます。
決して、無碍にはできない。

node.js に変わるランタイム



Deno

TypeScript

設定なしにTSを書ける！！！！

トップレベル `await`

`async` 関数の中でしか `await` は使えなかったが
トップレベルで `await` で書くことができる

パッケージマネージャ is dead...

```
export {  
  add,  
  multiply,  
} from "https://x.nest.land/ramda@0.27.0/source/index.js";
```

deno test

Jestをインストールしたりしなくていい！

```
import { assertEquals } from "https://deno.land/std@0.143.0/...";

// Compact form: name and function
Deno.test("hello world #1", () => {
  const x = 1 + 2;
  assertEquals(x, 3);
});
```

require撤廃

deno fmt, deno lint

ESLint? prettier?
必要ありません。

deno bundle

フロントエンドの開発に使える

公式が盛ん

deno deploy などのサービスもある

weblilio studyでも一部導入中👍

今のJSに疲れたのであれば、Denoをオススメします！

また、きっと、プログラミング（JS）を愛せるとおもいます