Open Source Resume

Powered by lowlighter/metrics

Basic info

🗿 C. Lewis

- 🕐 Joined GitHub 9 years ago
- 🧑‍🤝‍🧑 Followed by 16 users

🗂 Contributed to 155 repositories

📈 Activity

- ⊶ 2231 Commits
- 11 Pull requests reviewed
- 30 Pull requests opened
- ⚠ 57 Issues opened
- 💬 255 issue comments

Community stats

- 🏢 Member of 2 organizations
- 🧑‍🤝‍🧑 Following 58 users
- ♡ Sponsoring 0 repositories
- ☆ Starred 76 repositories
- 👁 Watching 81 repositories

📖 58 Repositories (including 31 forks)

- ⚖ Prefers MIT license
- 🏷 0 Releases
- 📦 0 Packages
- 🗄 2.48 GB used
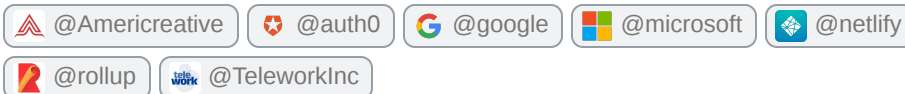- ± 75.2k added, 35.4k removed
- ♡ 0 Sponsors
- ☆ 8 Stargazers
- ⑂ 2 Forks
- 👁 26 Watchers

💬 Most used languages

- TypeScript
- JavaScript
- Java
- PHP
- CSS
- C
- HTML
- SCSS

🗂 Notable contributions

- ⛰ @Americreative
- 🛡 @auth0
- G @google
- ⊞ @microsoft
- ◈ @netlify
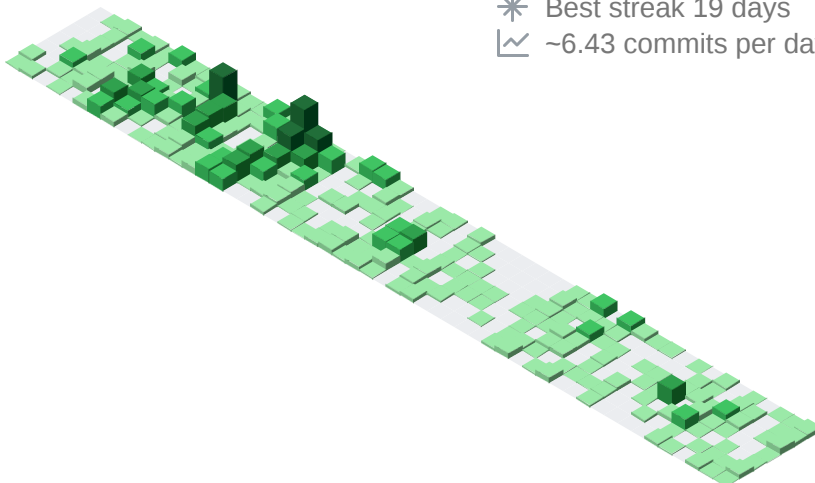- @rollup
- @TeleworkInc

📅 Contributions calendar

- 🔥 Current streak 9 days
- ✳ Best streak 19 days
- 📈 ~6.43 commits per day

🗐 Stackoverflow metrics

- ✳ 3053 reputation points
- ❓ 3 questions
- ♡ 47 badges
- ✓ 155 answers

💬 217 comments

☰ Top answers

↗ **Typescript: Index signature is missing in type**

🏷 casting, typescript ✅ Resolved

> TS wants us to define the type of the index. For example, to tell the
> compiler that you can index the object with any string, e.g.
> `myObj['anyString']` , change:
>
> ```
> interface myInterface {
>   myVal: string;
> (1 more line was trimmed)
> ```

👍 14   👎 0   💬 2   📅 28/12/2020

↗ **How get real char code from an emoji in JavaScript?**

🏷 javascript, emoji, fromcharcode ✅ Resolved

> `String.codePointAt()` and `String.fromCodePoint()` are made for
> this, though it is also possible to specify a UTF-16 `charCode` in
> legacy browsers using surrogate keypairs (see the discussion here).
>
> ```
> let emoji = "🙂",
>         charCode = emoji.charCodeAt(),
> (11 more lines were trimmed)
> ```

👍 6   👎 0   💬 1   📅 12/07/2020   ✅ Accepted

↗ **Flutter: Call a function on a child widget's state**

🏷 flutter ✅ Resolved

> State management is the one thing they dropped the ball on with
> respect to this framework. I've used static variables (if I need to share
> data between states) and GlobalKeys (for just one quick, dirty
> solution) to get this done. We're supposed to use InheritedWidgets,…

👍 5   👎 0   💬 1   📅 28/05/2018

☰ Recent answers

↗ **Typescript: Index signature is missing in type**

🏷 casting, typescript ✅ Resolved

> TS wants us to define the type of the index. For example, to tell the
> compiler that you can index the object with any string, e.g.
> `myObj['anyString']` , change:
>
> ```
> interface myInterface {
>   myVal: string;
> (1 more line was trimmed)
> ```

👍 14   👎 0   💬 2   📅 28/12/2020

↗ **Property 'value' does not exist on type 'EventTarget'**

🏷 angular, typescript, properties ✅ Resolved

> In my case, I had:
>
> ```
> const handleOnChange = (e: ChangeEvent) => {
>   doSomething(e.target.value);
> (1 more line was trimmed)
> ```
>
> And the issue was that I did not provide a type argument to
> ChangeEvent so that it knows `e.target` was an `HTMLInputElement` .

👍 2   👎 0   💬 1   📅 19/02/2021

↗ **JSC_REDECLARED_VARIABLE_ERROR when using gulp
closure compiler plugin**

🏷 javascript, typescript, gulp, google-closure-compiler

> Try `--env CUSTOM` or `--isolation_mode IIFE` . I ran into this when
> compiling code that conflicted with browser interfaces like `Node` .

👍 0   👎 0   💬 1   📅 25/12/2020