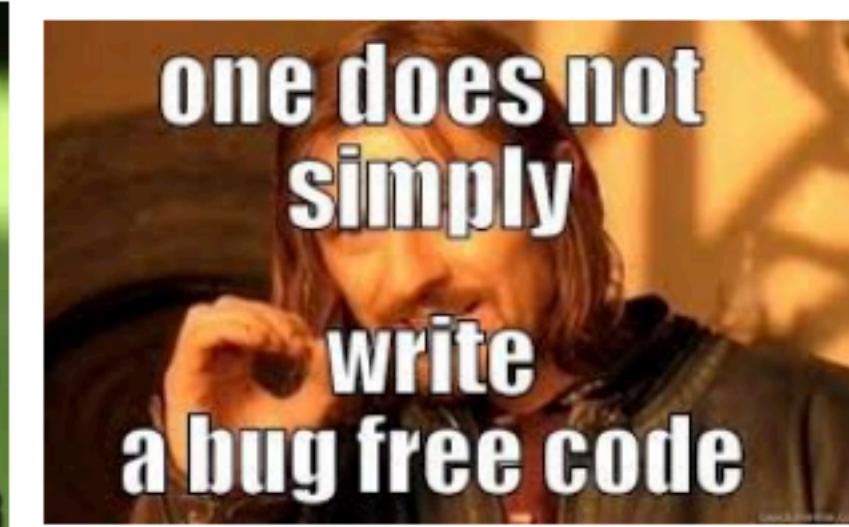
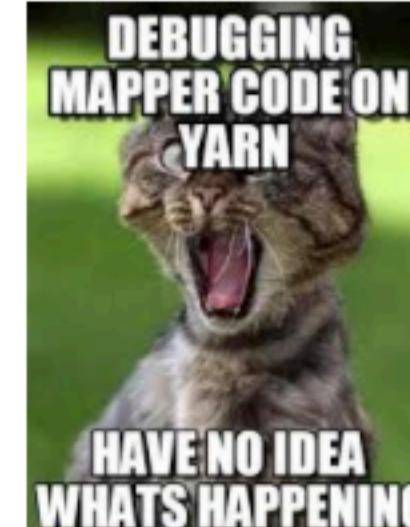
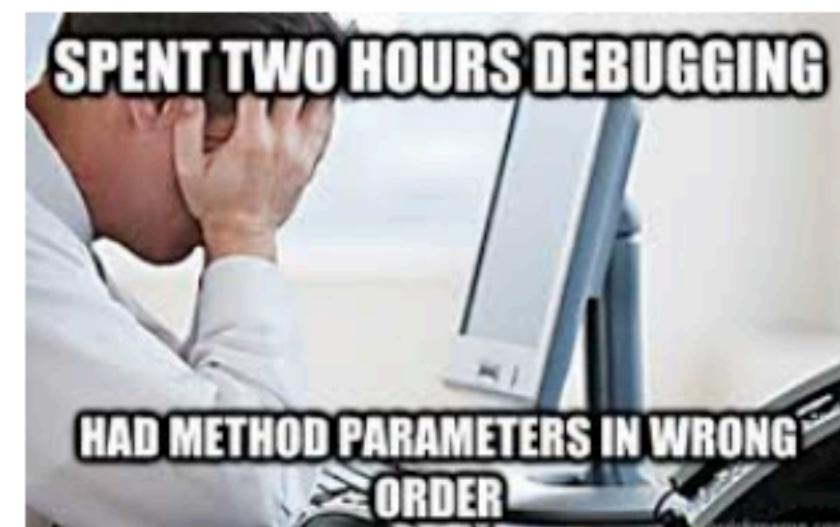
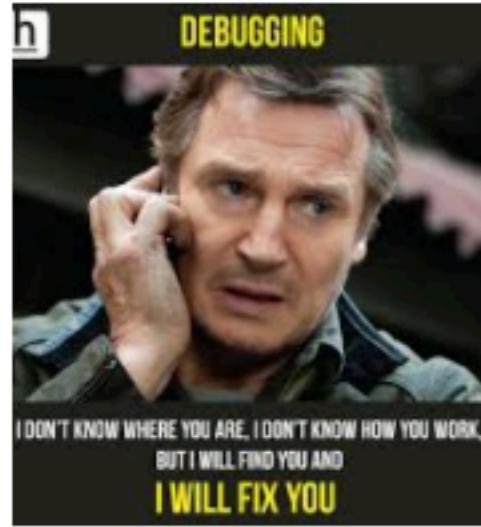
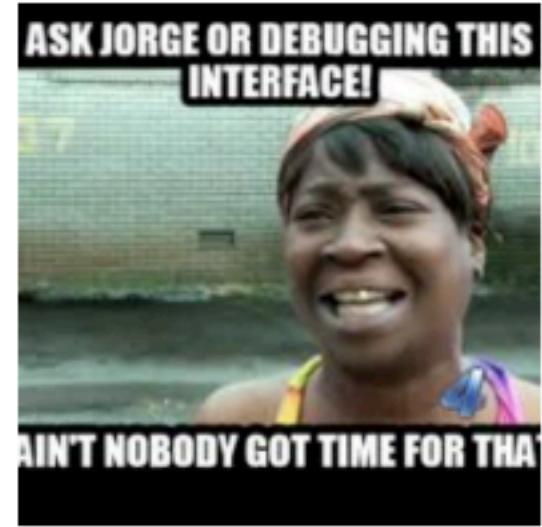
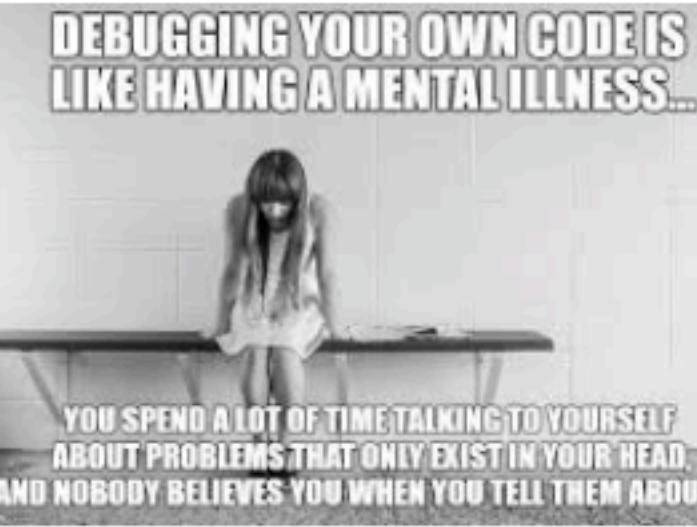
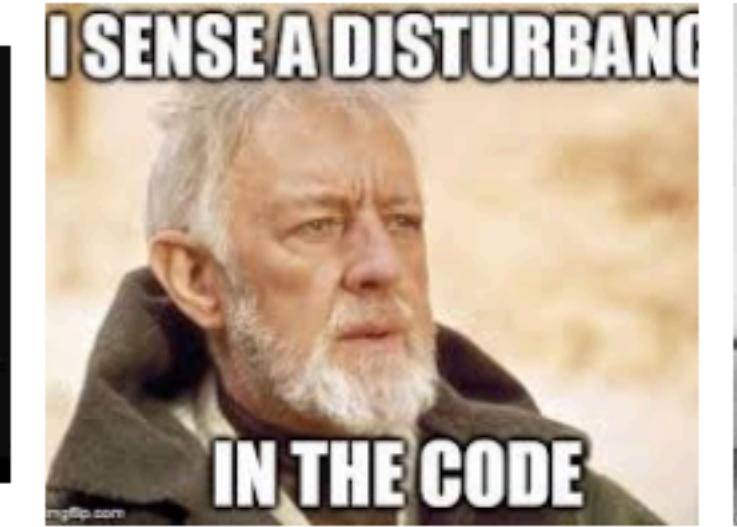
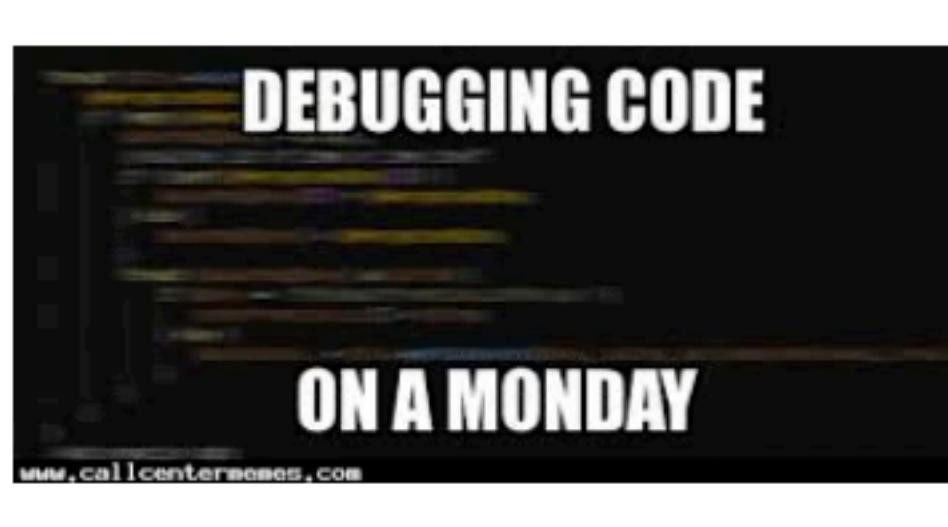
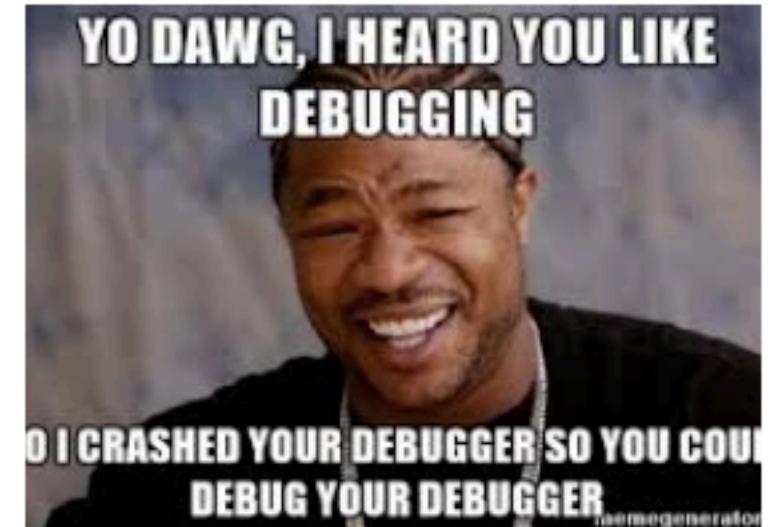
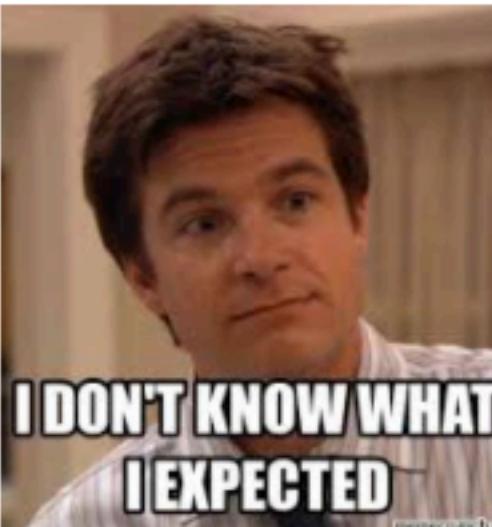
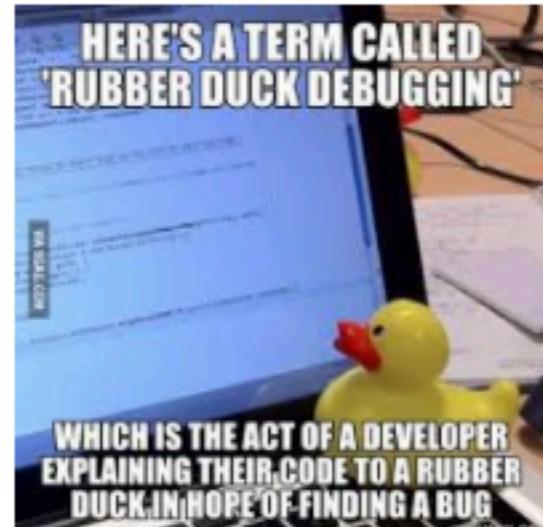
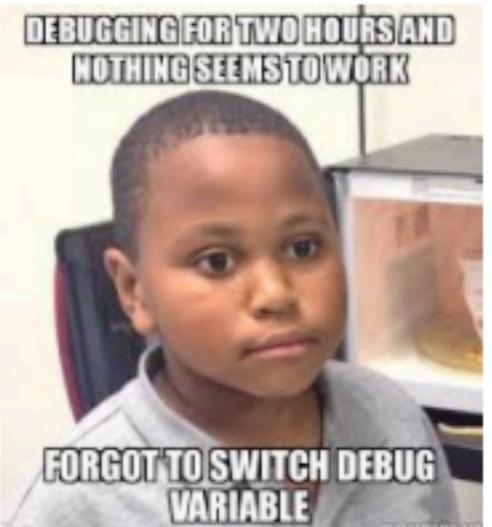
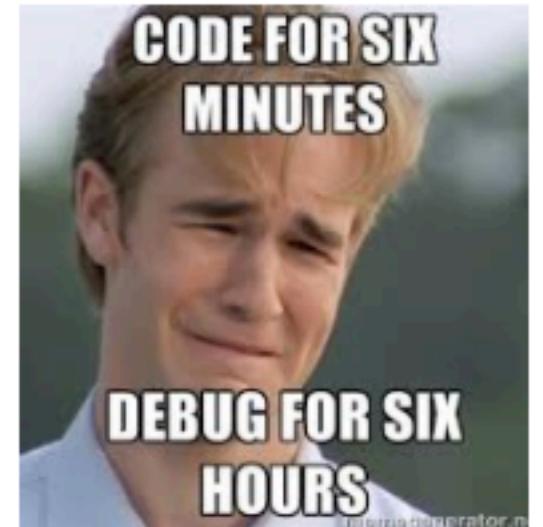
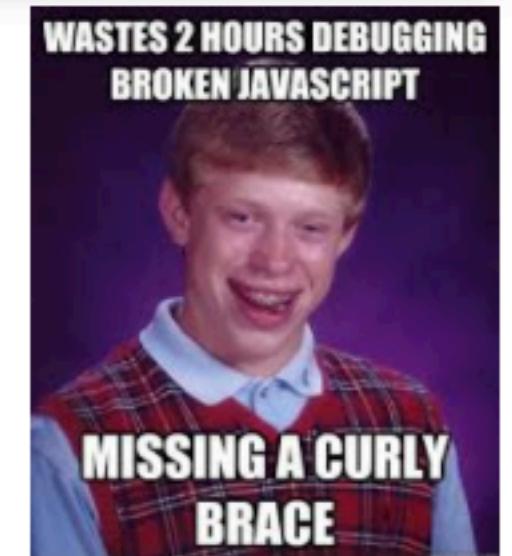
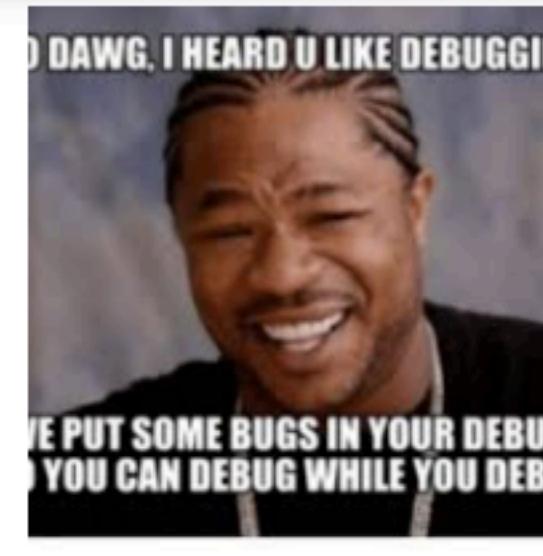


# DEBUGGING

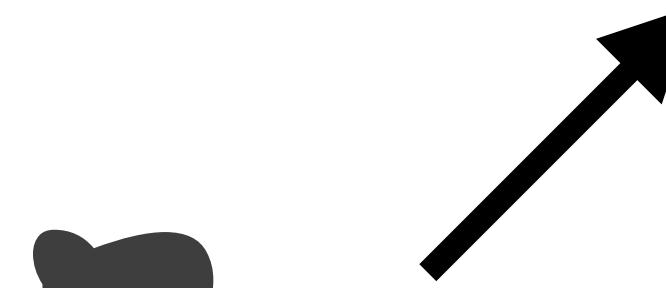




debugging meme



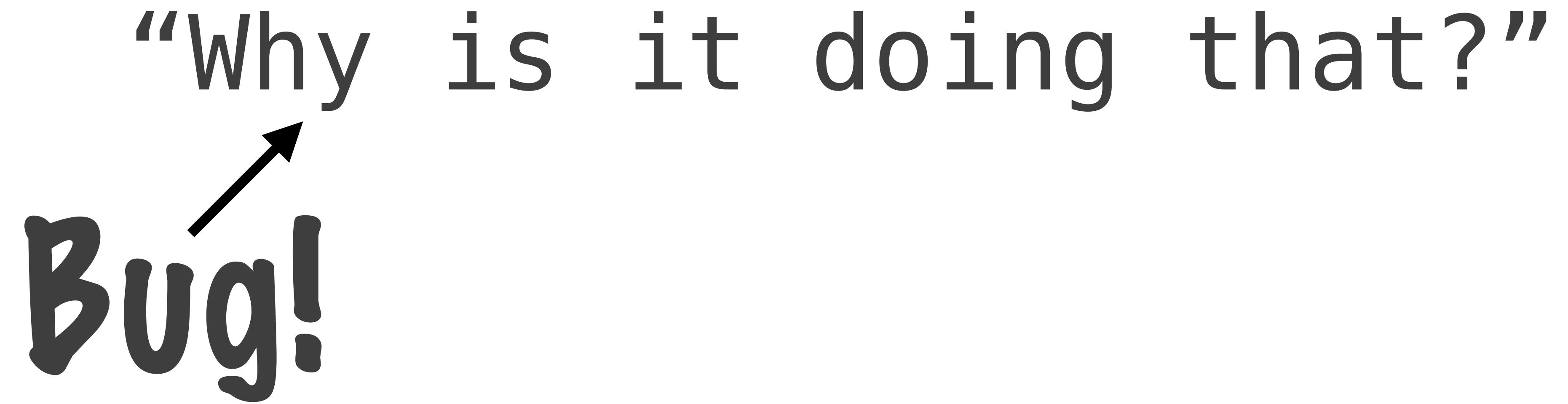
**Bug!**



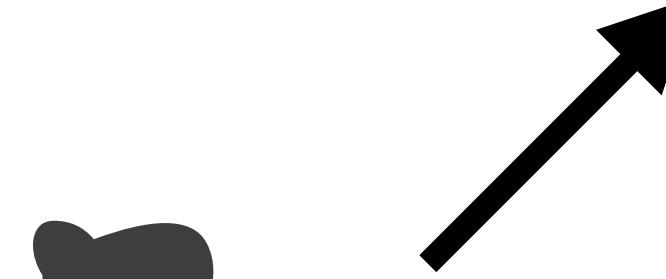
“It doesn’t work!”

**Bug!**

“Why is it doing that?”



**Bug!**



“That’s weird.”

```
const foo = `bar;
```

```
function bar () {  
}  
}
```

"We can't send [email]  
more than 500 miles"

<https://www.ibiblio.org/harris/500milemail.html>



*“Programmers Debugging Software”*

It is not so bad.

novice

20x

Time to debug

expert

1x



# What You'll Learn Today



# BUG

# BUG

# BUG

Prevention  
Detection  
Diagnosis  
Fixing

# PREVENTION

# PREVENTION

“Debugging is twice as hard as writing the code in the first place.

Therefore, if you write the code as cleverly as possible, you are,

by definition, not smart enough to debug it.”

– Brian W. Kernaghan

# PREVENTION

- 🐛 Readability > smallness
- 🐛 Simple > clever
- 🐛 git hygiene
- 🐛 Formatting matters!

# PREVENTION

```
1 async function main
2 () {
3 try { throws();
4 }catch (err) {console.log('caught')} }
```

indentation matters

# PREVENTION

```
1 async function main () {  
2   try {  
3     throws();  
4   }  
5   catch (err) {  
6     console.log('caught');  
7   }  
8 }
```

indentation matters

# PREVENTION

- 🐛 Readability > smallness
- 🐛 Simple > clever
- 🐛 git hygiene
- 🐛 Formatting matters!
- 🐛 Use functions in your favor
  - ▶ Prefer small, single-purposed and pure functions

# PREVENTION

```
1 let result
2 const add (a, b) => {
3   result = a + b
4 }
5 add(10, 20)
6 console.log(result)
```

avoid side-effects

# PREVENTION

```
1 const add = (a, b) => a + b  
2 console.log(add(10, 20))
```

avoid side-effects

# PREVENTION

```
1 const addOrSubtract = (a, b, operation) =>  
2   operation === 'add' ? a + b : a - b
```

single-purpose functions

# PREVENTION

```
1 const add = (a, b) => a + b  
2 const subtract = (a, b) => a - b
```

single-purpose functions

# PREVENTION

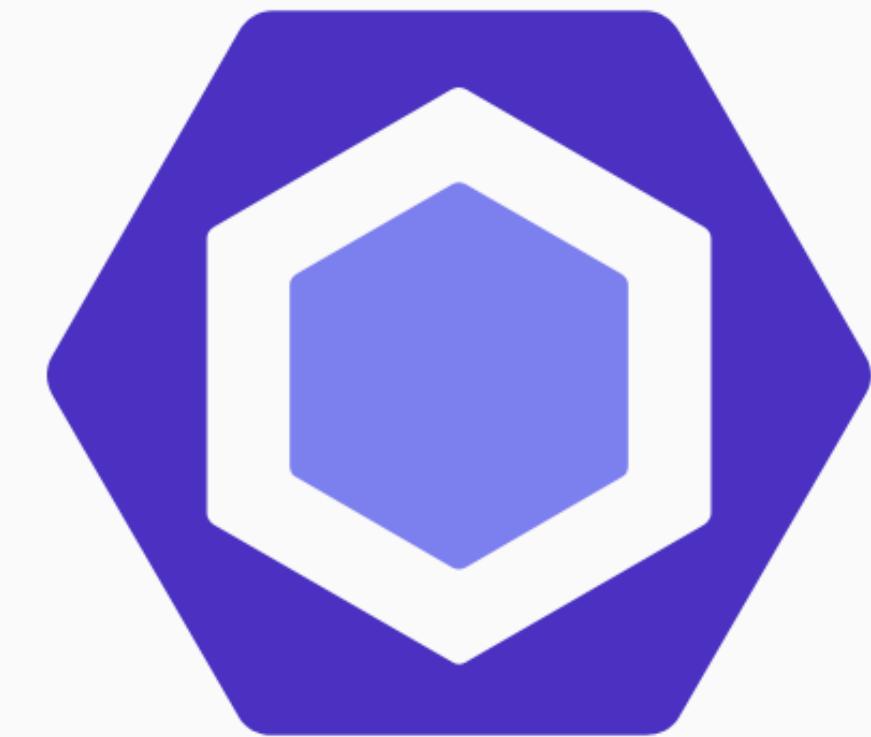
Tools can get help.

# PREVENTION



🐛 linter  
🐛 auto-formatter

# PREVENTION



ESLint

# PREVENTION

