sonar RULES

Products

- Secrets
- ABAP
- Apex
- C
- C++
- CloudFormation
- COBOL
- C#
- CSS
- Flex
- Go
- HTML
- Java
- JavaScript
- Kotlin
- Objective C
- PHP
- PL/I
- PL/SQL
- Python
- RPG
- Ruby
- Scala
- Swift
- Terraform
- Text
- TypeScript
- T-SQL
- VB.NET
- VB6
- XML

JavaScript static code analysis

Unique rules to find Bugs, Vulnerabilities, Security Hotspots, and Code Smells in your JAVASCRIPT code

All rules 285 | Vulnerability 29 | Bug 62 | Security Hotspot 43 | Code Smell 151 | Quick Fix 41

Tags | Search by name...

JavaScript parser failure
Code Smell

The ternary operator should not be used
Code Smell

"===" and "!==" should be used instead of "==" and "!="
Code Smell

Functions should not have too many lines of code
Code Smell

Track comments matching a regular expression
Code Smell

Statements should be on separate lines
Code Smell

Magic numbers should not be used
Code Smell

Collapsible "if" statements should be merged
Code Smell

Standard outputs should not be used directly to log anything
Code Smell

Files should not have too many lines of code
Code Smell

Lines should not be too long
Code Smell

Debugger statements should not be used

"delete" should not be used on arrays

Analyze your code

Code Smell Major

The `delete` operator can be used to remove a property from any object. Arrays are objects, so the `delete` operator can be used here too, but if it is, a hole will be left in the array because the indexes/keys won't be shifted to reflect the deletion.

The proper method for removing an element at a certain index would be:

- `Array.prototype.splice` - add/remove elements from the array
- `Array.prototype.pop` - add/remove elements from the end of the array
- `Array.prototype.shift` - add/remove elements from the beginning of the array

Noncompliant Code Example

```
var myArray = ['a', 'b', 'c', 'd'];

delete myArray[2];  // Noncompliant. myArray => ['a', 'b', u
console.log(myArray[2]); // expected value was 'd' but outpu
```

Compliant Solution

```
var myArray = ['a', 'b', 'c', 'd'];

// removes 1 element from index 2
removed = myArray.splice(2, 1);  // myArray => ['a', 'b', 'd
console.log(myArray[2]); // outputs 'd'
```

Available In:

sonarlint | sonarcloud | sonarqube

Vulnerability

"alert(...)" should not be used

Vulnerability

Regular expressions using Unicode character classes or property escapes should enable the unicode flag

Bug

The base should be provided to "parseInt"

Bug

Function declarations should not be made within blocks