



**leafletDemo
not provided**

js:leafletDemoJS

2022-08-29

目录

1. leafletDemo	Page 1
1.1. 概述	1
1.2. 问题分析	2
1.3. 问题详情	3
1.4. 质量配置	9

1. leafletDemo

报告提供了项目指标的概要，显示了与项目质量相关的最重要的指标。如果需要获取更详细的信息，请[登陆网站](#)进一步查询。

报告的项目为leafletDemo，生成时间为2022-08-29，使用的质量配置为 js:leafletDemoJS，共计207条规则。


1.1. 概述

编码问题

Bug	可靠性修复工作
0	0min
漏洞	安全修复工作
0	0min
坏味道	技术债务
84	7h17min
84	开启问题84
问题	重开问题0
	确认问题0
	误判问题0
	不修复的问题0
	已解决的问题223
	已删除的问题0
	阻断0
	严重49
	主要31
	次要4
	提示0

静态分析

项目规模

	leafletDemo	Sonar Report
--	-------------	--------------

594	行数	886
代码行数	方法	19
	类	0
	文件	5
	目录	N/A
	重复行(%)	0.0

复杂度

59	文件	11.8
复杂度		

注释(%)

20.2	注释行数	150
注释(%)		

1.2. 问题分析

违反最多的规则TOP10	
Variables should be declared with "let" or "const"	48
Sections of code should not be commented out	12
Variables should be used in the blocks where they are declared	8
"switch" statements should have "default" clauses	7
"for of" should be used with Iterables	4
"switch" statements should not be nested	2
CSS files should not be empty	1
Cognitive Complexity of functions should not be too high	1
Literals should not be thrown	1

违规最多的文件TOP5	
myJSver10.js	51
leafletDraw.vue	32


App.vue	1
---------	---

复杂度最高的文件TOP5	
myJSver10.js	32
leafletDraw.vue	25
App.vue	1
main.js	1

重复行最多的文件TOP5	
No duplications	

1.3. 问题详情

规则	Variables should be declared with "let" or "const"	
规则描述	<p>ECMAScript 2015 introduced the <code>let</code> and <code>const</code> keywords for block-scope variable declaration. Using <code>const</code> creates a read-only (constant) variable.</p> <p>The distinction between the variable types created by <code>var</code> and by <code>let</code> is significant, and a switch to <code>let</code> will help alleviate many of the variable scope issues which have caused confusion in the past.</p> <p>Because these new keywords create more precise variable types, they are preferred in environments that support ECMAScript 2015. However, some refactoring may be required by the switch from <code>var</code> to <code>let</code>, and you should be aware that they raise <code>SyntaxError</code>s in pre-ECMAScript 2015 environments.</p> <p>This rule raises an issue when <code>var</code> is used instead of <code>const</code> or <code>let</code>.</p> <p>Noncompliant Code Example</p> <pre>var color = "blue"; var size = 4;</pre> <p>Compliant Solution</p> <pre>const color = "blue"; let size = 4;</pre>	
文件名称	违规行	
myJSver10.js	17, 18, 22, 27, 34, 40, 52, 53, 54, 123, 124, 126, 127, 128, 141, 144, 152, 170, 172, 197, 201, 219, 227, 249, 274, 279, 305, 328, 338, 388, 392	

	leafletDemo	Sonar Report
--	-------------	--------------

leafletDraw.vue	75, 92, 109, 110, 111, 112, 161, 169, 170, 172, 173, 174, 187, 189, 210, 235, 245
-----------------	---

规则	Sections of code should not be commented out	
规则描述	Programmers should not comment out code as it bloats programs and reduces readability. Unused code should be deleted and can be retrieved from source control history if required.	
文件名称	违规行	
myJSver10.js	7, 38, 112, 133, 182, 198, 275	
leafletDraw.vue	14, 60, 80, 90, 105	

规则	Variables should be used in the blocks where they are declared
----	--

规则描述	<p>Variables that are declared inside a block but used outside of it (which is possible with a <code>var</code>-style declaration) should be declared outside the block.</p> <p>Noncompliant Code Example</p> <pre>function doSomething(a, b) { if (a > b) { var x = a - b; // Noncompliant } if (a > 4) { console.log(x); } for (var i = 0; i < m; i++) { // Noncompliant, both loops use same variable } for (var i = 0; i < n; i++) { } return a + b; }</pre> <p>Compliant Solution</p> <pre>function doSomething(a, b) { var x; if (a > b) { x = a - b; } if (a > 4) { console.log(x); } for (let i = 0; i < m; i++) { } for (let i = 0; i < n; i++) { } return a + b; }</pre>
文件名称	违规行
myJSver10.js	126, 127, 128, 172
leafletDraw.vue	172, 173, 174, 189

规则	"switch" statements should have "default" clauses
----	---

规则描述	<p>The requirement for a final default clause is defensive programming. The clause should either take appropriate action, or contain a suitable comment as to why no action is taken.</p> <p>Noncompliant Code Example</p> <pre>switch (param) { //missing default clause case 0: doSomething(); break; case 1: doSomethingElse(); break; }</pre> <pre>switch (param) { default: // default clause should be the last one error(); break; case 0: doSomething(); break; case 1: doSomethingElse(); break; }</pre> <p>Compliant Solution</p> <pre>switch (param) { case 0: doSomething(); break; case 1: doSomethingElse(); break; default: error(); break; }</pre> <p>See</p> <p>MITRE, CWE-478 - Missing Default Case in Switch Statement</p>
文件名称	违规行
myJSver10.js	56, 58, 84, 115
leafletDraw.vue	113, 162, 269

规则	"for of" should be used with Iterables
----	--

规则描述	<p>If you have an iterable, such as an array, set, or list, your best option for looping through its values is the <code>for of</code> syntax. Use a counter, and ... well you'll get the right behavior, but your code just isn't as clean or clear.</p> <p>Noncompliant Code Example</p> <pre>const arr = [4, 3, 2, 1]; for (let i = 0; i < arr.length; i++) { // Noncompliant console.log(arr[i]); }</pre> <p>Compliant Solution</p> <pre>const arr = [4, 3, 2, 1]; for (let value of arr) { console.log(value); }</pre>
文件名称	违规行
myJSver10.js	124, 172
leafletDraw.vue	170, 189

规则	"switch" statements should not be nested
----	--

规则描述	<p>Nested switch structures are difficult to understand because you can easily confuse the cases of an inner switch as belonging to an outer statement. Therefore nested switch statements should be avoided.</p> <p>Specifically, you should structure your code to avoid the need for nested switch statements, but if you cannot, then consider moving the inner switch to another function.</p> <p>Noncompliant Code Example</p> <pre>function foo(n, m) { switch (n) { case 0: switch (m) { // Noncompliant; nested switch // ... } case 1: // ... default: // ... } }</pre> <p>Compliant Solution</p> <pre>function foo(n, m) { switch (n) { case 0: bar(m); case 1: // ... default: // ... } } function bar(m) { switch(m) { // ... } }</pre>
文件名称	违规行
myJSver10.js	58, 84

规则	Literals should not be thrown
----	-------------------------------

规则描述	<p>It is a bad practice to throw something that's not derived at some level from <code>Error</code>. If you can't find an existing <code>Error</code> type that suitably conveys what you need to convey, then you should extend <code>Error</code> to create one.</p> <p>Specifically, part of the point of throwing <code>Errors</code> is to communicate about the conditions of the error, but literals have far less ability to communicate meaningfully than <code>Errors</code> because they don't include stacktraces.</p> <p>Noncompliant Code Example</p> <pre>throw 404; // Noncompliant throw "Invalid negative index."; // Noncompliant</pre> <p>Compliant Solution</p> <pre>throw new Error("Status: " + 404); throw new Error("Invalid negative index.");{code}</pre>
文件名称	违规行
leafletDraw.vue	318

规则	CSS files should not be empty
规则描述	This rule raises an issue when a CSS file is empty (ie: containing only spaces).
文件名称	违规行
App.vue	37

规则	Cognitive Complexity of functions should not be too high
规则描述	<p>Cognitive Complexity is a measure of how hard the control flow of a function is to understand. Functions with high Cognitive Complexity will be difficult to maintain.</p> <p>See</p> <p>Cognitive Complexity</p>
文件名称	违规行
myJSver10.js	114

1.4. 质量配置

质量配置	js:leafletDemoJS Bug:62 漏洞:11 坏味道:92	
规则	类型	违规级别
Callbacks of array methods should have return statements	Bug	阻断
Loops should not be infinite	Bug	阻断

Regular expressions should be syntactically valid	Bug	严重
Jump statements should not occur in "finally" blocks	Bug	严重
"in" should not be used with primitive types	Bug	严重
Function calls should not pass extra arguments	Bug	严重
"typeof" expressions should only be compared to valid values	Bug	严重
"Symbol" should not be used as a constructor	Bug	严重
A compare function should be provided when using "Array.prototype.sort()"	Bug	严重
Results of "in" and "instanceof" should be negated rather than operands	Bug	严重
"super()" should be invoked appropriately	Bug	严重
Getters and setters should access the expected fields	Bug	严重
Destructuring patterns should not be empty	Bug	主要
Alternatives in regular expressions should be grouped when used with anchors	Bug	主要
Assertions should not be given twice the same argument	Bug	主要
Unicode Grapheme Clusters should be avoided inside regex character classes	Bug	主要
"NaN" should not be used in comparisons	Bug	主要
Return values from functions without side effects should not be ignored	Bug	主要
Generators should "yield" something	Bug	主要
Related "if/else if" statements should not have the same condition	Bug	主要
All branches in a conditional structure should not have exactly the same implementation	Bug	主要
The output of functions that don't return anything should not be used	Bug	主要
Values should not be uselessly incremented	Bug	主要
Special identifiers should not be bound or assigned	Bug	主要
Properties of variables with "null" or "undefined" values should not be accessed	Bug	主要
Objects should not be created to be dropped immediately without being used	Bug	主要
A "for" loop update clause should move the counter in the right direction	Bug	主要
Exclusive tests should not be committed to version control	Bug	主要
Function argument names should be unique	Bug	主要
Variables should not be self-assigned	Bug	主要
Property names should not be duplicated within a class or object literal	Bug	主要
Non-empty statements should change control flow or have at least one side-effect	Bug	主要

React components should not render non-boolean condition values	Bug	主要
Comments inside JSX expressions should be enclosed in curly braces	Bug	主要
React "render" function should return a value	Bug	主要
Non-existent operators '+=', '=-' and '!=' should not be used	Bug	主要
"new" operators should be used with functions	Bug	主要
Loops with at most one iteration should be refactored	Bug	主要
Mocha timeout should be disabled by setting it to "0".	Bug	主要
Identical expressions should not be used on both sides of a binary operator	Bug	主要
All code should be reachable	Bug	主要
Bitwise operators should not be used in boolean contexts	Bug	主要
Strict equality operators should not be used with dissimilar types	Bug	主要
Setters should not return values	Bug	主要
Regular expressions with the global flag should be used with caution	Bug	主要
Promise rejections should not be caught by 'try' block	Bug	主要
Replacement strings should reference existing regular expression groups	Bug	主要
Alternation in regular expressions should not contain empty alternatives	Bug	主要
React state setter function should not be called with its matching state variable	Bug	主要
Regular expressions should not contain control characters	Bug	主要
Comma and logical OR operators should not be used in switch cases	Bug	主要
React Hooks should be properly called	Bug	主要
React's useState hook should only be used in the render function or body of a component	Bug	主要
Collection elements should not be replaced unconditionally	Bug	主要
Attempts should not be made to update "const" variables	Bug	主要
Errors should not be created without being thrown	Bug	主要
Collection sizes and array length comparisons should make sense	Bug	主要
Repeated patterns in regular expressions should not match the empty string	Bug	次要
"delete" should be used only with object properties	Bug	次要

Function parameters, caught exceptions and foreach variables' initial values should not be ignored	Bug	次要
"with" statements should not be used	Bug	次要
Empty collections should not be accessed or iterated	Bug	次要
XML parsers should not be vulnerable to XXE attacks	漏洞	阻断
File uploads should be restricted	漏洞	严重
JWT should be signed and verified with strong cipher algorithms	漏洞	严重
Origins should be verified during cross-origin communications	漏洞	严重
Cipher algorithms should be robust	漏洞	严重
Encryption algorithms should be used with secure mode and padding scheme	漏洞	严重
A new session should be created during user authentication	漏洞	严重
Weak SSL/TLS protocols should not be used	漏洞	严重
Cryptographic keys should be robust	漏洞	严重
Server hostnames should be verified during SSL/TLS connections	漏洞	严重
Server certificates should be verified during SSL/TLS connections	漏洞	严重
Tests should include assertions	坏味道	阻断
Variables should be declared explicitly	坏味道	阻断
Switch cases should end with an unconditional "break" statement	坏味道	阻断
Assertions should be complete	坏味道	阻断
"future reserved words" should not be used as identifiers	坏味道	阻断
"switch" statements should not contain non-case labels	坏味道	阻断
Function returns should not be invariant	坏味道	阻断
Octal values should not be used	坏味道	阻断
"default" clauses should be last	坏味道	严重
"switch" statements should not be nested	坏味道	严重
Cognitive Complexity of functions should not be too high	坏味道	严重
Functions should not be empty	坏味道	严重
Tests should not execute any code after "done()" is called	坏味道	严重
"switch" statements should have "default" clauses	坏味道	严重
"await" should only be used with promises	坏味道	严重
"for" loop increment clauses should modify the loops' counters	坏味道	严重
Loop counters should not be assigned to from within the loop body	坏味道	严重
Conditionals should start on new lines	坏味道	严重

A conditionally executed single line should be denoted by indentation	坏味道	严重
"void" should not be used	坏味道	严重
Equality operators should not be used in "for" loop termination conditions	坏味道	严重
Variables should be declared with "let" or "const"	坏味道	严重
Boolean expressions should not be gratuitous	坏味道	主要
Track uses of "FIXME" tags	坏味道	主要
Parameters should be passed in the correct order	坏味道	主要
Functions should not be called both with and without "new"	坏味道	主要
Names of regular expressions named groups should be used	坏味道	主要
Character classes in regular expressions should not contain the same character twice	坏味道	主要
Single-character alternations in regular expressions should be replaced with character classes	坏味道	主要
Character classes in regular expressions should not contain only one character	坏味道	主要
Comma operator should not be used	坏味道	主要
Template literals should not be nested	坏味道	主要
"arguments.caller" and "arguments.callee" should not be used	坏味道	主要
Redundant pairs of parentheses should be removed	坏味道	主要
Multiline blocks should be enclosed in curly braces	坏味道	主要
Labels should not be used	坏味道	主要
Assertion arguments should be passed in the correct order	坏味道	主要
"switch" statements should not have too many "case" clauses	坏味道	主要
Tests should check which exception is thrown	坏味道	主要
Regular expressions should not be too complicated	坏味道	主要
Reluctant quantifiers in regular expressions should be followed by an expression that can't match the empty string	坏味道	主要
Assignments should not be made from within sub-expressions	坏味道	主要
Shorthand promises should be used	坏味道	主要
"indexOf" checks should not be for positive numbers	坏味道	主要
Arguments to built-in functions should match documented types	坏味道	主要
Ternary operators should not be nested	坏味道	主要
Unused assignments should be removed	坏味道	主要
Functions should always return the same type	坏味道	主要

Variables should be used in the blocks where they are declared	坏味道	主要
Array indexes should be numeric	坏味道	主要
Literals should not be thrown	坏味道	主要
Sections of code should not be commented out	坏味道	主要
Variables and functions should not be redeclared	坏味道	主要
Functions should not have too many parameters	坏味道	主要
Nested blocks of code should not be left empty	坏味道	主要
"delete" should not be used on arrays	坏味道	主要
Function parameters with default values should be last	坏味道	主要
Two branches in a conditional structure should not have exactly the same implementation	坏味道	主要
Functions should not be defined inside loops	坏味道	主要
Chai assertions should have only one reason to succeed	坏味道	主要
Array-mutating methods should not be used misleadingly	坏味道	主要
Assignments should not be redundant	坏味道	主要
"in" should not be used on arrays	坏味道	主要
Regular expressions should not contain multiple spaces	坏味道	主要
Functions should not have identical implementations	坏味道	主要
Only "while", "do", "for" and "switch" statements should be labelled	坏味道	主要
Sparse arrays should not be declared	坏味道	主要
Unused methods of React components should be removed	坏味道	主要
Collection and array contents should be used	坏味道	主要
Regular expressions should not contain empty groups	坏味道	主要
Boolean checks should not be inverted	坏味道	次要
A "while" loop should be used instead of a "for" loop	坏味道	次要
Function call arguments should not start on new lines	坏味道	次要
Extra semicolons should be removed	坏味道	次要
Return of boolean expressions should not be wrapped into an "if-then-else" statement	坏味道	次要
Boolean literals should not be used in comparisons	坏味道	次要
Unnecessary imports should be removed	坏味道	次要
Unused local variables and functions should be removed	坏味道	次要
"await" should not be used redundantly	坏味道	次要
Wrapper objects should not be used for primitive types	坏味道	次要
"for of" should be used with Iterables	坏味道	次要

"switch" statements should have at least 3 "case" clauses	坏味道	次要
The global "this" object should not be used	坏味道	次要
Deprecated APIs should not be used	坏味道	次要
Multiline string literals should not be used	坏味道	次要
Class names should comply with a naming convention	坏味道	次要
"catch" clauses should do more than rethrow	坏味道	次要
Regular expression quantifiers and character classes should be used concisely	坏味道	次要
Regular expression literals should be used when possible	坏味道	次要
Imports from the same modules should be merged	坏味道	次要
Jump statements should not be redundant	坏味道	次要
Track uses of "TODO" tags	坏味道	提示