

# Rules availables in eslint and tslint

SONAR	TSLINT	ESLINT	Description
<a href="#">FunctionComplexity</a>	cyclomatic-complexity	complexity	Functions should not be too complex
<a href="#">DebuggerStatement</a>	no-debugger	no-debugger	Debugger statements should not be used
<a href="#">EqEqEq</a>	triple-equals	eqlreq	=== and !== should be used instead of == and !=
<a href="#">Eval</a>	no-eval	no-eval	Code should not be dynamically injected and executed
<a href="#">OneStatementPerLine</a>	one-variable-per-declaration	one-var	Statements should be on separate lines
<a href="#">Semicolon</a>	semicolon	semi	Statements should end with semicolons
<a href="#">CurlyBraces</a>	curly	curly	Control structures should use curly braces
<a href="#">BitwiseOperators</a>	no-bitwise	no-bitwise	Bitwise operators should not be used
<a href="#">PrimitiveWrappers</a>	no-construct	no-new-wrappers	Wrapper objects should not be used for primitive types
<a href="#">ForIn</a>	forin	guard-for-in	for...in loops should filter properties before acting on them
<a href="#">FunctionDeclarationsWithinBlocks</a>	no-inner-declarations	no-inner-declarations	Function declarations should not be made within blocks
<a href="#">TrailingComma</a>	trailing-comma	comma-dangle	Trailing commas should not be used
<a href="#">AssignmentWithinCondition</a>	no-conditional-assignment	no-cond-assign	Assignments should not be made from within sub-expressions
<a href="#">LabelPlacement</a>	label-position	no-labels	Only while, do and for statements should be labelled
<a href="#">SwitchWithoutDefault</a>	switch-default	default-case	switch statements should end with default clauses
<a href="#">NonEmptyCaseWithoutBreak</a>	no-switch-case-fall-through	no-fallthrough	Switch cases should end with an unconditional break statement
<a href="#">EmptyBlock</a>	no-empty	no-empty	Nested blocks of code should not be left empty
<a href="#">TabCharacter</a>	indent	indent	Tabulation characters should not be used
<a href="#">BoundOrAssignedEvalOrArguments</a>	no-eval	no-eval	eval and arguments should not be bound or assigned
<a href="#">VariableDeclarationAfterUsage</a>	no-use-before-declare	no-use-before-define	Variables should be declared before they are used
<a href="#">S1125</a>	no-extra-boolean-cast	no-extra-boolean-cast	Boolean literals should not be redundant
<a href="#">RedeclaredVariable</a>	no-duplicate-variable	no-redeclare	This rule checks that the var keyword is not used to declare a variable with a name that is already in use.
<a href="#">TrailingWhitespace</a>	no-trailing-whitespace	no-trailing-spaces	Lines should not end with trailing whitespaces
<a href="#">S1442</a>	ban	no-alert	alert(...) should not be used

## Rules availables in eslint but currently unavailables in tslint

SONAR	ESLINT	Description
<a href="#">WithStatement</a>	no-with	with statements should not be used
<a href="#">MultilineStringLiterals</a>	no-multi-str	Multiline string literals should not be used
<a href="#">ArrayAndObjectConstructors</a>	no-new-object && no-array-constructor	Array and Object constructors should not be used
<a href="#">ExcessiveParameterList</a>	max-params	Functions should not have too many parameters
<a href="#">FunctionDefinitionInsideLoop</a>	no-loop-func	Functions should not be defined inside loops
<a href="#">S1135</a>	no-warning-comments	Track uses of TODO tags
<a href="#">TrailingComment</a>	no-inline-comments	Comments should not be located at the end of lines of code
<a href="#">S1134</a>	no-warning-comments	Track uses of FIXME tags
<a href="#">S878</a>	no-sequences	Comma operator should not be used
<a href="#">NestedIfDepth</a>	max-depth	Control flow statements if, for, while, switch and try should not be nested too deeply

## Rules availables in eslint but not applicable to Typescript.

SONAR	ESLINT	Description
<a href="#">UnreachableCode</a>	no-unreachable	Jump statements should not be followed by other statements
<a href="#">DuplicateFunctionArgument</a>	no-dupe-args	Function argument names should be unique
<a href="#">DuplicatePropertyName</a>	no-dupe-keys	Property names should not be duplicated within a class or object literal
<a href="#">OctalNumber</a>	no-octal	Octal values should not be used
<a href="#">StrictMode</a>	strict	strict mode should be used with caution
<a href="#">UnusedVariable</a>	no-unused-vars [Deprecated]	Unused local variables and functions should be removed

## Rules not available neither in tslint nor eslint

SONAR	Description
<a href="#">HtmlComments</a>	HTML-style comments should not be used
<a href="#">CollapsibleIfStatements</a>	Collapsible "if" statements should be merged
<a href="#">ConstructorFunctionsForSideEffects</a>	Objects should not be created to be dropped immediately without being used
<a href="#">FutureReservedWords</a>	"future reserved words" should not be used as identifiers
<a href="#">ConditionalComment</a>	Internet Explorer's conditional comments should not be used
<a href="#">S1145</a>	Useless "if(true) {...}" and "if(false){...}" blocks should be removed
<a href="#">S138</a>	Functions should not have too many lines
<a href="#">RedeclaredFunction</a>	This rule checks that functions declared in same scope don't have identical names.

<a href="#">SameNameForFunctionAndVariable</a>	Declaring both a variable and a function with an identical name in same scope should be avoided.
<a href="#">NamedFunctionExpression</a>	Named function expressions should not be used
<a href="#">TooManyBreakOrContinueInLoop</a>	Loops should not contain more than a single "break" or "continue" statement
<a href="#">UnusedFunctionArgument</a>	Unused function parameters should be removed
<a href="#">S100</a>	Function names should comply with a naming convention
<a href="#">S1301</a>	"switch" statements should have at least 3 "case" clauses
<a href="#">S1126</a>	Return of boolean expressions should not be wrapped into an "if-then-else" statement
<a href="#">S1264</a>	A "while" loop should be used instead of a "for" loop
<a href="#">S1472</a>	Function call arguments should not start on new lines
<a href="#">S1219</a>	"switch" statements should not contain non-case labels
<a href="#">S1067</a>	Expressions should not be too complex

## Links

Sources:

- [Sonar Rules](#)
- [Sonar Rules Repository](#)
- [TSLint rules](#)
- [ESLint rules](#)

tslint-eslint-rules:

- [ESLint rules for TSLints - GitHub](#)
- [ESLint rules for TSLint - npm](#)