The following is a partial list of structural errors that JSLint will flag as being suspect coding practices. (You can find the complete list in JSLint's documentation.)

- JSLint expects that all lines end with a semicolon. While JavaScript does allow a line-feed character to act as a line termination character, this is considered ambiguous and poor coding style.

- Statements using `if` and `for` must use curly braces for the statement blocks.

- In JavaScript, unlike other programming languages, a block does not introduce variable scope. JavaScript supports only function-level scope. Thus, JSLint accepts only blocks that are part of function, `if`, `switch`, `while`, `for`, `do`, and `try` statements and will flag any other blocks as errors.

- A `var` may be declared only once, and it must be declared before it is used.

- JSLint flags code that comes immediately after a `return`, `break`, `continue`, or `throw` statement as unreachable code. These statements must be immediately followed by a closing curly brace.