

9/23/19

From Obinna /8/2019 (and my mods)

## Simplified step-by-step tutorial Guide on How to Setup/Use ESLint for Visual Studio Code

1. Launch-open VSCode and install the Extension named, ESLint (by Dirk Baeumer)
2. If not already installed, download and install node.js from <https://nodejs.org>
3. Install eslint by opening a command window (or terminal) and executing the command: `npm install --global eslint` (might take a few minutes)
4. Close and restart VSCode
5. In VSCode, create a new (or open an existing) project folder
6. Next, create a file named, **.eslintrc.json** (this is the eslint configuration file) and add the following JSON-formatted config settings into the **.eslintrc.json** file and save it:

```
{
  "env": {"browser": true, "es6": true, "mocha": true},
  "extends": "eslint:recommended",
  "parserOptions": {"sourceType": "script"},
  "rules": {
    "strict": ["error", "global"],
    "semi": ["warn", "always"],
    "quotes": ["error", "double"],
    "no-alert": "off",
    "no-console": "off",
    "no-debugger": "off",
    "no-unused-vars": "warn",
    "id-length": [2, {"min": 3, "max": 20, "exceptions": ["i", "j"]}],
    "camelcase": [2, {"properties": "always"}],
    "valid-jsdoc": ["error"],
    "require-jsdoc": ["error", {
      "require": { "FunctionDeclaration": true,
        "MethodDefinition": true,
        "ClassDeclaration": true,
        "ArrowFunctionExpression": false,
        "FunctionExpression": true } } ]
  }
}
```

Now, add your Javascript source code file(s) and start coding.

The ESLint server starts automatically and uses the config settings to indicate any issues found in your code, as errors or warning (as shown in the screen below). Open the vsCode terminal to see descriptions of problems and/or errors.

With this setup, eslint will work for any new JS source file. None of the initial setup steps need to be repeated.

**Note:** These are just a few basic settings for starters. More settings can be added for a more customized linting experience, as needed. See <https://eslint.org/docs/rules/> for the full listing/documentation of the ESLint rules.

Repetition of steps 6 can be eliminated, by creating the **.eslintrc.json** file once and making it a global eslint config file, by placing it in the User's home directory (e.g. For Windows, store the **.eslintrc.json** file in **C:\Users\{user's-windows-login-name}** folder).

Example js-doc for this eslint setup

```
/**
 *
 * @param {int} bottom - bottom of range
 * @param {int} top - top of range
 * @returns {function} that has test of whether a number is in bottom-top range
 */
function inBetween(bottom, top) {
    return function (num) { if (num > bottom && num < top) return true; else return false; };
}
```