*3. Assignment #3 - Checking for syntax and readability using Lint*

Using the code editor to your left, your task is to use the Linter on your Boggle solver and Boggle Solver Test. A linter is a tool that analyzes source code to flag programming errors, bugs, stylistic errors, and suspicious constructs.

Please provided the files for your solution (i.e. boggle\_solver.jsboggle\_solver.js) and your test program (i.e. boggle\_solver.test.jsboggle\_solver.test.js) in the code windows to the left.

NOTE: If you want to use eslint from the command line, click on the Tools->Terminal to bring up the terminal. You must change directories to /home/codio/workspace/Boggle\_Impl2\_After\_CodeReview.

Copy your Boggle solver code into the code window for boggle\_solver.jsboggle\_solver.js.

Make sure your solution runs by selecting TEST SOLVER below, or issue the CLI commands in the terminal (node boggle\_solver.js).

TEST SOLVER

Now lets move on to testing your solution. Copy your Boggle tester code into the code window for boggle\_solver.test.jsboggle\_solver.test.js.

Make sure your solution runs by selecting TEST TESTSUITE below, or issue the CLI commands in the terminal (jest boggle\_solver.test.js)

*Code Given*

const boggle\_solver = require('/home/codio/workspace/Boggle\_Impl2\_After\_CodeReview/boggle\_solver.js');

function ToGrid(...rows) {

return rows.map(row => row.split(""));

}

*/\*\* Lowercases a string array in-place. (Used for case-insensitive string array*

*\* matching).*

*\* @param {string[]}* stringArray *- String array to be lowercase.*

*\*/*

function lowercaseStringArray(stringArray) {

for (let i = 0; i < stringArray.length; i++)

stringArray[i] = stringArray[i].toLowerCase();

}

describe('Boggle Solver tests', () => {

describe('Normal input', () => {

test('Normal case 3x3', () => {

*// Tests a normal 3x3 grid.*

const grid = [['A', 'B', 'C'],

['D', 'E', 'F'],

['G', 'H', 'I']];

const dictionary = ['abc', 'abdhi', 'abi'];

const expected = ['abc', 'abdhi'];

let solutions = boggle\_solver.findAllSolutions(grid, dictionary);

*// Lowercasing for case-insensitive string array matching.*

lowercaseStringArray(solutions);

lowercaseStringArray(expected);

expect(solutions.sort()).toEqual(expected.sort());

});

});

});