Index.js

//Do NOT change the code in this file!!!

var Jasmine = require('jasmine');

var jasmine = new Jasmine();

jasmine.loadConfig({

spec\_dir: 'spec',

spec\_files: [

"\*\*/\*[sS]pec.js"

],

});

jasmine.execute();

launchCodeRocks.js

function launchOutput(num){

let result1 = num % 2;

let result2 = num % 3;

let result3 = num % 5;

if (result1 !== 0 && result2 !== 0 && result3 !== 0) {

return "Rutabagas! That doesn't work.";

} else {

if (result1 === 0 && result3 === 0) {

return 'Launch Rocks!';

} else {

if (result2 === 0 && result3 === 0) {

return 'Code Rocks!';

} else {

if (result1 === 0 && result2 === 0) {

return 'LaunchCode!';

} else {

if (result1 === 0) {

return "Launch!";

} else {

if (result2 === 0) {

return 'Code!';

} else {

if (result3 === 0) {

return 'Rocks!';

}}}}}}}

}

module.exports = {

launchOutput: launchOutput

}

launchCodeRocks.spec.js

const launchOutput = require('../launchCodeRocks');

const assert = require('assert');

describe("launchOutput", function(){

it("Testing if divisible by 2", function() {

let output = launchOutput.launchOutput(4);

assert.strictEqual(output, "Launch!");

});

it("Testing if divisible by 3", function() {

let output = launchOutput.launchOutput(3);

assert.strictEqual(output, "Code!");

});

it("Testing if divisible by 5", function() {

let output = launchOutput.launchOutput(5);

assert.strictEqual(output, "Rocks!");

});

it("Testing if divisible by 2 and 3", function() {

let output = launchOutput.launchOutput(6);

assert.strictEqual(output, "LaunchCode!");

});

it("Testing if divisible by 3 and 5", function() {

let output = launchOutput.launchOutput(15);

assert.strictEqual(output, "Code Rocks!");

});

it("Testing if divisible by 2 and 5", function() {

let output = launchOutput.launchOutput(10);

assert.strictEqual(output, "Launch Rocks!");

});

it("Testing if not divisible by 2, 3 or 5", function() {

let output = launchOutput.launchOutput(17);

assert.strictEqual(output, "Rutabagas! That doesn't work.");

});

});