8/31/2021

CS305 Object Oriented and Functional Programming in JavaScript

W1D3 Code Quality Assignment

**PART I:** Complete the following tasks from The JavaScript Language book. You do not have to submit these to GitHub. Try to complete the answers before looking at the solutions.

Coding Style section: Bad style task

Ninja code section: (nothing to implement here, just write the answers)

Read the 'Ninja code' section on your own. Write the real rules implied by the irony examples. E.g.,

- Ninja irony: Make the code as short as possible. Show how smart you are
  - Meaning: do not sacrifice code clarity for brevity.
- Ninja irony: use single-letter variable names everywhere.
  - Meaning: *your answer here*
- Ninja irony: If the team rules forbid the use of one-letter and vague names – shorten them, make abbreviations
  - Meaning: *your answer here*
- Ninja irony: While choosing a name try to use the most abstract word
  - Meaning: *your answer here*

Etc etc

Automated testing with Mocha section: **What's wrong in the test?**

**Part II.** Implement the following in VSCode, and submit to your GitHub repository for this assignment. The name of the folder should be: d13codeQuality. Use esLint and JS Doc with your code. The eslint configuration will require you to write JS Doc for each function. See the example JSDoc comment in the instructions for installing esLint. (esLintSetupInstructions.pdf in Sakai > Resources > lab helpers)

1. The following is a unit test for an isVowel function. Implement the function so that the unit tests are satisfied. Try it first by writing your function and Mocha test in the same file, then put the function in a separate file and use Nodes' CommonJS modules to export and import into the different files.

```
"use strict";
/* global assert isVowel*/
/*  isVowel() that takes a character (i.e. a string of length 1) and returns true if it is a vowel, false otherwise. */
```

```
describe("isVowel", function () {

    it("a is vowel", function () {
        assert.equal(isVowel("a"), true);
    });

    it("e is vowel", function () {
        assert.equal(isVowel("e"), true);
    });
    it("i is vowel", function () {
        assert.equal(isVowel("i"), true);
    });
    it("o is vowel", function () {
        assert.equal(isVowel("o"), true);
    });
    it("u is vowel", function () {
        assert.equal(isVowel("u"), true);
    });

    it("z is not vowel", function () {
        assert.equal(isVowel("z"), false);
    });

    it("5 is not vowel", function () {
        assert.equal(isVowel("5"), false);
    });
});
```

2. Test your computeSalesCommission function from yesterday with the following Mocha test. First do the test by adding the following to the same file that contains your computeSalesCommission code. Then put it in a separate file and use CommonJS modules. Call this file d13mochaTests.js.

```
describe("test of ComputSalesCommission", function(){
   it("tests salaried and 200 sales", function(){
      assert.strictEqual(computeSalesCommission(true, 200), 0);
   });
   it("tests not salaried and 200 sales", function(){
      assert.strictEqual(computeSalesCommission(false, 200), 0);
   });
   it("tests salaried and 300 sales", function(){
```

```
    assert.strictEqual(computeSalesCommission(true, 300), 3);
  });
  it("tests not salaried and 300 sales", function(){
    assert.strictEqual(computeSalesCommission(false, 300), 6);
  });
  it("tests salaried and 3500 sales", function(){
    assert.strictEqual(computeSalesCommission(true, 3500), 65);
  });
  it("tests not salaried and 3500 sales", function(){
    assert.strictEqual(computeSalesCommission(false, 3500), 100);
  });
});
```

3. Add an additional Mocha “describe” method call with “it” method calls to d13mochaTests.js for each of the two test cases given for your compoundInterest function from the previous lesson. Import your compoundInterest code using CommonJS modules.

4. Do the same for the rest of the functions and test cases in yesterday’s assignment:
    a. calcDownpayment
    b. sumDigits
    c. multDigits
    d. convertFahrenheit
    e. calcDistance