How Do Assertions Work in Unit Testing?

freecodecamp.org/learn/front-end-development-libraries-v9/lecture-understanding-the-different-types-of-testing/how-do-assertions-work-in-unit-testing

You were first introduced to assertions and unit testing in a previous lesson. But as a review, unit testing is when you check the output of individual functions or components independently, in contrast with testing the final output of the entire program. Assertions are used to test that the code is behaving as expected.

In this lesson, we are going to take a closer look at assertions in unit tests and explore some best practices.

A commonly used assertion library is Chai. This is the same library used inside of the freeCodeCamp codebase. Whenever you go through a workshop or lab, there is an automated test suite that runs against your code to ensure that your code is behaving the way it is supposed to.

Let's take a look at a few examples that are from the Full Stack Developer Curriculum.

In Build a Calculator workshop, Step 1 is testing for a function called `addTwoAndSeven`:

```
assert.isFunction(addTwoAndSeven);
```

That assertion checks that the provided code contains a function named `addTwoAndSeven`. If the user creates an array, string, or anything else that's not a function and assigns it to `addTwoAndSeven`, then the test will fail.

Let's take a look at another assertion from later in the workshop:

```
assert.equal(addThreeAndFour(), 7);
```

This assertion checks that the return value from the `addThreeAndFour` function is equal to the number `7`. If the return value does not equal `7`, then the test will fail.

There are many more asserts built into Chai, like ones that check if a value is a boolean, if an object has a certain property name, or if a value is `null`. Other common JavaScript assertion libraries are should.js and expect.js.

Regardless of which assertion library you are working with, there are a few best practices to be aware of. The first is to write clear assert messages. Clear and meaningful assert messages should describe the intent behind it. Otherwise, it will be difficult to locate which tests are failing.

Another best practice is to write descriptive failure messages. A descriptive failure message should state which condition is true. This will help you figure out why a particular assert is failing.

As you continue learning to code, it's important to familiarize yourself with testing and start writing tests for your personal projects. Doing this will go a long way in helping you prepare for your first developer job.

Questions

Which of the following asserts is used in Chai to test that a value is a function?

`assert.functioning()`

`assert.isFunction()`

Correct!

`assert.function()`

`assert.isTrueFunction()`

Which of the following assert is used in Chai to test if two values are equal?

`assert.equality()`

`assert.equaling()`

`assert.equal()`

Correct!

`assert.equalize()`

Which of the following is NOT a commonly used assertion library?

expect.js

Chai

should.js

Gherkin

Correct!

Navigated to How Do Assertions Work in Unit Testing?