

Scorecard

Exercise: Band pass filter
No show?: No

Reviewer: Cheryl Fredericks
Language: JavaScript

I use an agile process

Rating: **Strong**

You began your information-gathering by first noting down the client brief. A top-down approach results in a better understanding of the core system that the client expects.

Your representation of the requirements of the system and acceptance criteria was sufficiently concise through using an input-output table.

You asked good questions that really cleared up any ambiguity that may exist within the system

I can model anything

Rating: **Strong**

You adopted a functional programming paradigm which worked well because your methods returned values based on input parameters and also because this was a stateless system.

You adhered to Javascript conventions naming conventions using camel_case for your method and variable names. You used nouns for variable names and they were descriptive of their purpose.

I can TDD anything

Rating: **Strong**

You are familiar with the Jest framework and testing syntax resulting in a smooth testing process.

You demonstrated a solid understanding of test driven and incremental development by writing only enough code in order to pass the failing test . You maintained a simple test progression and the result was a simple and efficient solution . More importantly it was one that could be built upon without causing previous tests to fail unless the entire codebase was rewritten from scratch.

I'm happy to see you using test triangulation as a means of driving the safe creation of generic code . Well done .

I can program fluently

Rating: **Strong**

You were fluent in your use of the terminal and local environment.

You were able to skillfully leverage built-in methods and functionality.

I can refactor anything

Rating: **Strong**

You did a great job of refactoring your codebase when a pattern emerged and none of the refactors overextended the functionality of the codebase.

I can debug anything

Rating: **Strong**

Your familiarity with the stack trace and commonplace errors resulted in a smooth debugging of the programme.

I write code that is easy to change

Rating: **Strong**

You were consistent with committing that after every passing test, you will be able to develop an extensive version history in case anything needs to be rolled back.

Your tests were appropriately decoupled with the implementation so you were able to add complexity without any tests failing. I didn't count trouble with method functionality which caused some tests to fail before you fixed the bug.

I have a methodological approach to problem solving

Rating: **Strong**

You started by addressing the core functionality that you noted down.

You did a fantastic job of adhering to the Red-Green-Refactor cycle and maintaining an incremental development of the solution.

I can justify the way I work

Rating: **Strong**

You were communicative and transparent in your approach to the problem.

General feedback

This was a great review and you demonstrated a great understanding of language conventions and programming principles.

Your information gathering was well handled and the system requirements were clearly and concisely represented through the use of an input-output table.

The RGR cycle was adhered to and you developed the programme incrementally without overextending functionality past getting the failing test to pass