

CS 253 Mark Holliday

Rubric for Project 3

Team: XXX Students: XXX

Category	Points Deducted
Correctness	
Style	
Documentation	
Total Score	

Grading Rubric: Correctness (84%)

General correctness.

	Comments
If this project is part of a project sequence, is all the functionality required by earlier projects working correctly?	
If example input (such as a data set) and the associated example output for that input is provided, then does the project when run on that example input provide the example output exactly as shown in the example output? In other words, is the answer the same, but also is the wording and formatting exactly the same?	
Does the project work correctly (correct answer, wording, and formatting) for other input besides the example input? This other input should be somewhat similar to the example output. If other input might be used that is quite different in terms of errors then the types of input possible will be specified by the instructor.	
When run does the project show as output anything that is not supposed to be in the output (that is, beyond the required output)? For example, is debugging information shown in the output?	
When a unit tests tests something that is supposed to throw an exception you should catch that exception.	
Does the project throw exceptions when run that are	

not supposed to occur?	
All of your code should be in function definitions except for your calls to the <code>main()</code> function. In other words, the only global code should be the calls to the <code>main()</code> function. This style is more modular and easier to understand.	
Do all the servers (that is, <code>railway_XXX.js</code>) run indefinitely (that is, not stop) so that each continues to respond to API calls?	

Unit Testing Correctness

	Comments
Each function must have at least one test that is just for that function. You cannot have one unit test of a top-level function (such as <code>getJourneys()</code>) and say that one test is your test for all the functions that top-level function calls.	
Is there at least 95% code coverage using the code coverage definition of every function, statement and branch? So every statement in every function needs to be executed in at least one test and every decision point needs to be reached with a true and false result.	
Each test should test one thing only; that is, be specific and informative. Each test needs to be specific in that you are checking whether a function returns the specific expected value or has the correct side effect or the correct exception is thrown when provided particular input. For example, given particular data structure does <code>getNetworkName(data)</code> return the correct network name? Also, what happens when you pass it null, undefined, or a number?	
Just checking that the return value or side effect or thrown exception is of a type such as <code>Object</code> or <code>String</code> is not acceptable. You must check for the exact required return value (or side effect or thrown exception), e.g. "Notional Railway Company". You may have to enhance existing functions to do this gracefully.	

Grading Rubric: Style (6%)

	Comments

Does the code conform to the Google JavaScript Style Guide, https://google.github.io/styleguide/jsguide.html ?	
Is the code well formatted and indented?	
Are variable and functions descriptive of purpose and assigned informative names?	
Are no lines (comments as well as code) over 100 characters long?	
Are function definitions not too long and each have one and only one purpose? Functions should rarely be more than 75 lines of actual code and ideally under 50 lines in length.	
Are function and class definitions and global code efficient; that is, they do not use more loops and code than is needed for a concise, readable, efficient solution?	
Except for very short function definitions, does the function body have clearly separate sections of code? In particular is there a blank line before a new code section and, if needed for readability, a one line comment explaining what the next section of code does.	
Is there at least one blank line but no more than approximately three blank lines above each function definition to separate the function definitions for readability? This implies that there are no parts of a source code file with large chunks of blank lines.	

Grading Rubric: Documentation (10%)

	Comments
Is there a README.md file at the top level of the project repository that is in Markdown and lists the author names, the date, the project title (such as CS 253 Project 1) how to run the program, any packages that need to be installed, a meaningful description of the functionality the program provides from the viewpoint of the user, and any known errors in your code?	
Does each source code file have a start-of-file comment at its start that includes author names, date software in the file was last modified, and a	

<p>description of the functionality of the code in the file? A description such as "The file is a bunch of functions needed for Project 1." is not a good description. You need to explain the functionality your specific functions are providing.</p>	
<p>For Jest test suite describe and test/it functions, JSDoc is not needed. But is the first argument of those functions have an accurate and detailed description of the purpose of the function?</p>	
<p>Does each function definition (besides Jest describe and test/it functions) have JSDoc? Is that JSDoc following the JSDoc standard including all the required parts such as specifying parameters, return values, and exceptions thrown? Does each REST API endpoint have JSDoc?</p>	
<p>If more than one person on the team, has each team member separately written an attestation commenting on the contributions by themselves and their teammate and how well they worked together?</p>	
<p>Have the attestations been added on the Canvas course site to the submission location for this project? The attestation can be put in a file that is submitted on Canvas or submitted a blank file and put as a comment in the Comments part of that submission location.</p>	