Staff Full Stack – Backend code challenge

1. Given two strings, determine if these are anagrams of each other. Two strings are anagrams of each other, if, and only if, they contain all the same characters the same amount of times.
    a. Examples: heart and earth, cars and scar, star and rats
    b. Additionally, the following test case MUST pass:
        i. “Tom Marvolo Riddle” -> “I am Lord Voldemort”

2. Balanced parenthesis: Given a string, determine if it is “balanced”, where “balanced” is defined by having a matching closing parenthesis for each open parenthesis (in the order in which it was opened). The set of all possible characters is: {“(“, ”{“, ”[“ ,“]”, ”}”, “)”}.
    a. Examples of balanced parenthesis:
        i. “(({{{{[[[]]]}}}}))”
        ii. “{{{[][][]}}}”
        iii. “()(){{{}}}[][]”
        iv. “()()()()”
    b. Examples of unbalanced parenthesis:
        i. “((((()}”
        ii. “)((()))()()”
        iii. “(((()()()))(()”

3. Create the code for a lambda function (or cloud-based equivalent) that returns a sha 256 hash for a given string if and only if said string consists of at least 8 characters, at least one number, and at least one special character. Include any yaml or json files necessary for deployment (without any secret keys or vulnerable data points), configuration files or diagrams as necessary to explain how this would be deployed and if any additional cloud-based resources are necessary to run said solution optimally. As an additional (bonus) point, callers to the lambda function should receive a meaningful message if and when things fail (for the corresponding reason). Consider using something like API Gateway (or equivalent) to call the lambda function as a RESTful endpoint.