Error Handling and Debugging | Lesson 5

Practice Solution

Instructions

You will need to compare your submission for the **Practice: Exception Handling and finally** activity to the solution below.

Solution

Corrected Code with Debugged Errors

```javascript
function processFile(fileName, fileData) {
    try {
        // Validate inputs
        if (!fileName) {
            throw new ReferenceError("File name is missing.");
        }
        if (typeof fileData !== "string") {
            throw new TypeError("File data must be a string.");
        }
        if (fileData.trim() === "") {
            throw new Error("File data cannot be empty.");
        }

        // Simulate file processing
        console.log(`Processing file: ${fileName}`);
        console.log(`File content: ${fileData}`);
    } catch (err) {
        console.error(`${err.name}: ${err.message}`);
    } finally {
        // Ensure resources are always released
        console.log("Closing file handle...");
    }
}
```

```
// Test Cases
processFile(); // Output: ReferenceError: File name is missing.
processFile("myFile.txt", 42); // Output: TypeError: File data must
be a string.
processFile("myFile.txt", ""); // Output: Error: File data cannot be
empty.
processFile("myFile.txt", "Hello, world!");
// Output:
// Processing file: myFile.txt
// File content: Hello, world!
```

Consider how your choices differ from the solution code and write a brief description of:

- How your code is different from the solution.
- What the solution code adds to your learning or understanding.

Submission

Enter your answer in the field below and then press submit.