This is a closed note, closed book exam. The exam will be taken in class, using paper and pencil.

Exam 1 has 100 points total.

Part A of the exam will be comprised of 35 multiple choice and short answer questions worth two points each. Students are advised to recall past quizzes in preparation for part A. Students will be asked to write very short jQuery statements. Students will be asked to read short portions of JavaScript and predict what it will do when it runs.

Part A is worth 70 points total.

Part B of the exam will instruct the student to write several very focused/specific portions of code. Part B is worth 30 points total.

In Part B, students may be asked to write a client-side Ajax call, a client-side error trap(s), and/or SQL statements. Students may be asked to define a simple client-side, supporting function using arrow notation.

Some topics to consider and review:

GENERAL CONCEPTS

1. URL Encoded data.
2. JSON Objects (plain/simple objects)
3. JavaScript Variable declaration (let, const, var)
4. JavaScript Data types (number, string, boolean, null, undefined, object)
5. Conditional statements.
    a. Checking for bad values: undefined, isNaN, empty string
    b. Comparison operators.
6. For-loops.
7. Function expressions and callback functions (with an emphasis on arrow notation.)

CLIENT-SIDE TECHNOLOGIES

1. Bootstrap classes
    a. Container
    b. Row
    c. Column
    d. alert, and alert contextual classes
    e. btn, and btn contextual classes
2. jQuery
    a. "#" (hashtag) vs. "." (dot)
    b. The ajax method
    c. show
    d. hide
    e. html
    f. val
    g. append
    h. addClass
    i. removeClass
    j. ready
    k. click
    l. the serialize method

SERVER-SIDE TECHNOLOGIES

1. SQL commands: CREATE DATABASE, USE, SELECT, INSERT, UPDATE, DELETE, SQL aggregate functions, and a JOIN of no more than two tables.
2. The mysql2 connection object used in Node.js (as seen in the endpoint template.) Especially:
    a. the use of placeholders to parameterize a query

    b. The purpose of the insertId attribute.
3. The correct use of GET, POST, PUT, PATCH and/or DELETE requests.
4. The correct use of HTTP Status codes: 200, 400, 500.

ADDITIONAL NOTES

Students are encouraged to review the lecture materials presented to date. They are a good indicator of what the instructor wants to prioritize.

Here's a rough overview of what has been covered:

1. A11y
2. JavaScript and jQuery
3. SQL Statements
4. What's an API? What's a Web Service? What's a Web Application?
5. REST (The REST concepts are what the entire course has been structured around!)
6. Client-side versus Server-side technologies

Finally, as we have worked on our assignments, the following concepts should be clear to each student by now:

- What does Url Encoded Data? What role does it play?
- What does it mean for form data to be *serialized*? What's the difference between an HTML *name* attributed, and an HTML *id* attribute?
- What are hidden input fields used for?
- What's a "self-closing" HTML tag?