

This is a closed note, closed book exam. The exam will be taken in class, using paper and pencil. Students will be assigned seats. ***Students must sit in their assigned seats*** or receive a grade penalty at the discretion of the instructor.

Exam 1 has 100 points total. **Part A** is worth 60 points total. **Part B** is worth 40 points total.

Part A of the exam will be comprised of 30 multiple choice questions worth two points each. Students are advised to recall past quizzes in preparation for part A. Students may be asked to read short portions of JavaScript and predict what it will do when it runs.

In Part B, students will be asked to write very short jQuery statements and SQL Statements.

You are expected to complete part A using the SCANTRON provided. You must indicate your first and last name on your SCANTRON. For Part B you are expected to write on the exam paper itself where indicated. You must indicate your first and last name on your exam paper. Students who fail to turn in **both** their SCANTRON and their exam paper will receive a grade of **zero**.

You are expected to work independently, without any external resources. Use of smartphones, earbuds, and/or other electronic devices ("smart" or otherwise) is prohibited.

Taking photographs of the exam, sharing any portion of the exam, and/or collaborating with another student on the exam, is prohibited. Any student doing so will receive a **zero** for the exam and face additional disciplinary action.

Students in the Tuesday / Thursday section of the class (MIS3502, Section 1) will take their exam in Speakman Lab 6.

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. Conditional statements.
 - a. Checking for bad values: undefined, isNaN, empty string
 - b. Comparison operators.
5. 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: USE, SELECT, INSERT, UPDATE, DELETE, SQL aggregate functions, and a JOIN of no more than two tables. LEFT / RIGHT joins are also in scope.
2. The mysql2 connection object used in Node.js (as seen in the endpoint template.) Especially the use of placeholders to parameterize a query
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 projects, the following concepts should be clear to each student by now:

- What are some common AI prompting strategies? (See Project 2)
- What is the basic composition of a web service?
- What are some of the visual elements of VS Code + GitHub Copilot? What is their purpose?
- What are some Best Practices for creating a UI? (See Shafer's Short List in Project 2)
- What is 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's a "self-closing" HTML tag?